



TESIS - IS185401

**METODE ESTIMASI BIAYA MENGGUNAKAN  
FUNCTION POINT DAN STORY POINT UNTUK  
PROYEK PENGEMBANGAN PERMAINAN DIGITAL**

**MOCHAMMAD FAJAR HILLMAN  
05211750012022**

Dosen Pembimbing  
Dr. Apol Pribadi Subriadi, S.T., M.T.  
NIP. 197002252009121001

Departemen Sistem Informasi  
Fakultas Teknologi Informasi dan Komunikasi  
Institut Teknologi Sepuluh Nopember  
2019

# LEMBAR PENGESAHAN TESIS

Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar  
**Magister Ilmu Komputer (M.Kom)**

di

**Institut Teknologi Sepuluh Nopember**

Oleh:

**MOCHAMMAD FAJAR HILLMAN**

**NRP: 05211750012022**

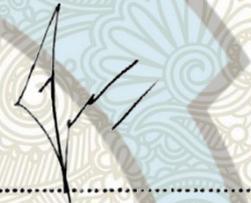
Tanggal Ujian: 26 Desember 2019

Periode Wisuda: Maret 2020

Disetujui oleh:

Pembimbing:

1. Dr. Apol Pribadi Subriadi, S.T., M.T.  
NIP: 197002252009121001



Penguji:

1. Nur Aini Rachmawati, S.Kom., M.Sc.Eng., Ph.D.  
NIP: 198201202005012001



2. Faizal Mahananto, S.Kom., M.Eng., Ph.D.  
NIP: 198510312019031009



Kepala Departemen Sistem Informasi  
Fakultas Teknologi Informasi dan Komunikasi



**Mahendrawati ER, S.T., M.Sc., Ph.D.**

**NIP: 197610112006042001**

# **METODE ESTIMASI BIAYA MENGGUNAKAN FUNCTION POINT DAN STORY POINT UNTUK PROYEK PENGEMBANGAN PERMAINAN DIGITAL**

Nama Mahasiswa : Mochammad Fajar Hillman  
NRP : 05211750012022  
Dosen Pembimbing : Dr. Apol Pribadi Subriadi, S.T., M.T.

## **ABSTRAK**

Industri permainan digital sedang berada di puncak popularitasnya. Berdasarkan Superdata, revenue global dari industri permainan digital di tahun 2017 mencapai USD 108.4 milyar, meningkat USD 17.4 milyar dari tahun 2016. Revenue yang dihasilkan pada tahun 2016 bahkan melampaui capaian pendapatan kombinasi dua segmen industri entertainment lainnya yaitu industri film dan musik. Hasil pendapatan yang menggiurkan dari industri digital membuat meningkatnya daya saing di industri permainan digital yang mengakibatkan banyak perusahaan yang mengalami kegagalan. Meningkatnya angka kegagalan membuat metode estimasi biaya untuk perangkat lunak permainan digital semakin dibutuhkan karena faktor terbesar yang mengakibatkan kegagalan dalam pembangunan perangkat lunak adalah biaya.

Metode Function Point Analysis (FPA) yang merupakan metode standar industri dinilai gagal karena tidak sesuai dengan karakteristik permainan digital yang 58.96% dari kebutuhannya merupakan kebutuhan yang berkaitan dengan unsur seni. Namun disisi lain metode FPA merupakan standar inputan utama pada metode estimasi biaya. Oleh karena itu, penelitian ini bertujuan untuk mengkombinasikan metode FPA dengan metode Story Point (MSP). Metode story point diharapkan dapat memberikan sebuah nilai pada unsur seni yang diabaikan oleh perhitungan FPA.

Berdasarkan hasil penelitian ini, metode FPA-MSP memiliki akurasi estimasi yang lebih baik dengan nilai rata-rata gap sebesar 29.57% dibandingkan estimasi FPA murni yang memiliki rata-rata gap sebesar 53.51%. Metode FPA-MSP membuktikan bahwa unsur seni menjadi faktor utama yang harus dipertimbangkan dalam upaya estimasi biaya perangkat lunak permainan digital.

**Kata Kunci:** estimasi biaya perangkat lunak, Function Point Analysis, permainan digital, Story Point, unsur seni.

[HALAMAN INI SENGAJA DIKOSONGKAN]

# **A COST ESTIMATION METHOD FOR GAME DEVELOPMENT PROJECT USING FUNCTION POINT AND STORY POINT**

By : Mochammad Fajar Hillman  
Student Identity Number : 05211750012022  
Supervisor : Dr. Apol Pribadi Subriadi, S.T., M.T.

## **ABSTRACT**

The digital game industry is reaching the peak of its popularity. Based on Superdata, global revenue from the digital game industry in 2017 reached USD 108.4 billion, an increase of USD 17.4 billion from 2016. Even in 2016, the revenue generated by the industry exceeded the combined achievements of the two other entertainment industry segments, namely the film and music industry. The lucrative income from the digital industry has made competitiveness increased in the digital game industry, which has resulted in many companies failing in the development process. The increasing failure rate makes the cost estimation method for digital game software increasingly needed because the most significant factor that causes failure in software development is cost.

The Function Point Analysis (FPA) method, which is an industry-standard method, is deemed a failure because it is incompatible with the characteristics of digital games, which 58.96% of their needs are needs related to aesthetic elements such as art and sound. However, on the other hand, the FPA method is still relevant because used as the primary input for the cost estimation method. Therefore, this study aims to combine the FPA method with the Story Point (MSP) method which expected to give value to the aesthetic elements that are ignored by FPA calculations.

Based on the results of this study, the FPA-MSP method has proven to be a better estimation with an accuracy value of 29.57% than the pure FPA which has 53.51%. The FPA-MSP method proves that the element of aesthetic is a significant factor that must be considered in the culture of estimating the cost of the digital game software.

**Keywords:** aesthetic, digital game industry, function point, software cost estimation, story point.

[HALAMAN INI SENGAJA DIKOSONGKAN]

## KATA PENGANTAR

Puji syukur penulis panjatkan kepada Allah SWT, yang telah memberikan Ridho, Rahmat, dan Hidayah-Nya sehingga Tesis yang berjudul “Metode Estimasi Biaya Menggunakan Function Point dan Story Point untuk Proyek Pengembangan Permainan Digital” dapat disusun dengan baik. Tesis ini disusun sebagai salah satu syarat menyelesaikan pendidikan pada Program Magister Sistem Informasi, Jurusan Sistem Informasi, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember.

Dalam proses penyelesaian tesis ini, penulis mendapatkan banyak bantuan, baik bantuan moral maupun materiil dari berbagai pihak. Oleh karena itu, penulis mengucapkan banyak terimakasih kepada :

1. Orang tua penulis tersayang, Ir. Zainal Muludi, M.T. dan Dra. Siti Chopsih, M.Pd., yang senantiasa memberikan kasih sayang, motivasi doa, dan dukungan baik secara moral dan materiil dalam perjalanan penulis menempuh pendidikan magister hingga penyelesaian buku tesis ini.
2. Kakak tersayang penulis, Maya Auristika Rizhati yang memberikan semangat, dukungan dan do'a dari awal hingga akhir pengerjaan buku tesis ini.
3. Paman dan Bibi penulis tersayang, Drs. Marsudi M.S., dan Dra. Siti Zahrok, M.Pd., yang telah menjadi orang tua kedua penulis yang memberikan kasih sayang, doa, dan dukungan baik secara moral dan materiil serta meluangkan waktu, tenaga dan pikiran selama penulis menempuh pendidikan magister hingga penyelesaian studi tesis ini.
4. Bapak Dr. Apol Pribadi Subriadi, S.T., M.T., selaku Dosen Wali Akademik yang telah memberikan kesempatan penulis untuk dapat menempuh pendidikan magister, sebagai target motivasi penulis untuk memberikan hasil yang terbaik dalam keseluruhan perjalanan pendidikan magister ini. Selaku Dosen Pembimbing yang telah meluangkan waktu, tenaga dan pikiran serta memberikan ilmu,

kesabaran, dukungan materiil selama membimbing penulis dari awal hingga tesis ini selesai.

5. Ibu Nur Aini Rakhmawati, S. Kom., M.Sc.Eng., Ph.D., selaku Dosen Penguji I yang telah bersedia menguji dan memberikan arahan serta masukan untuk menjadikan penelitian ini lebih baik dan membuka wawasan penulis lebih jauh.
6. Bapak Faizal Mahananto, S. Kom., M. Eng., Ph.D., selaku Dosen Penguji II yang telah bersedia menguji dan memberikan kritikan membangun yang telah membuat peneliti dapat melihat sudut pandang baru dari penelitian ini.
7. Teman-teman terbaik penulis, M. Syaqiq Azzahid dan Abudzar Ali, serta seluruh keluarga besar Re:Tale Game Studio yang telah bersedia menjadi narasumber utama penelitian ini.
8. Bapak dan Ibu dosen yang telah mendidik serta memberikan ilmu selama Penulis menempuh pendidikan di Jurusan Sistem Informasi, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember.
9. Segenap staf dan karyawan di Jurusan Sistem Informasi, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember yang membantu Penulis dalam pelaksanaan tesis ini. Khususnya mbak Vian yang telah banyak membantu penulis dalam melancarkan administrasi baik tesis maupun selama penulis menempuh studi magister.
10. Teman-teman seperjuangan penulis, Asih Nur Fadhillah, Silmie Vidiya Fani, Anggraeni Widya Purwita, Alexander Wirapraja, Mirza Arie Andini Permana, dan teman-teman keluarga besar S2 Sistem Informasi ITS yang telah memberikan dukungan selama pengerjaan tesis ini dan menjadi motivasi penulis untuk memberikan hasil yang terbaik.
11. Semua pihak yang tidak dapat disebutkan satu per satu, yang telah membantu dan terlibat baik secara langsung maupun tidak langsung dalam penulisan tesis ini.

Penulis menyadari bahwa tesis ini tidak lepas dari kesalahan dan kekurangan, serta masih sangat jauh dari kata sempurna. Oleh karena itu, Penulis

bersedia menerima kritik dan saran yang membangun melalui email penulis *mhfajar8@gmail.com* untuk memperbaiki diri. Penulis berharap tesis ini dapat memberi manfaat bagi kemajuan dunia pendidikan di Indonesia.

Surabaya, Januari 2020

Mochammad Fajar Hillman

[HALAMAN INI SENGAJA DIKOSONGKAN]

# DAFTAR ISI

LEMBAR PENGESAHAN .....	ii
ABSTRAK.....	iii
ABSTRACT.....	v
KATA PENGANTAR .....	vii
DAFTAR ISI.....	xi
DAFTAR GAMBAR .....	xiii
DAFTAR TABEL.....	xv
BAB 1 PENDAHULUAN .....	1
1.1. Latar belakang .....	1
1.2. Rumusan masalah.....	5
1.3. Tujuan penelitian.....	5
1.4. Lingkup penelitian.....	6
1.5. Kontribusi penelitian .....	6
1.5.1. Kontribusi bidang keilmuan.....	6
1.5.2. Kontribusi praktis.....	6
1.6. Sistematika penulisan .....	7
BAB 2 TINJAUAN PUSTAKA .....	9
2.1. Kajian teori.....	9
2.1.1. Industri permainan digital .....	9
2.1.2. Pengukuran perangkat lunak.....	11
2.1.3. Function Point Analysis (FPA).....	14
2.1.4. Story Point.....	36
2.1.5. Estimasi biaya perangkat lunak.....	39
2.2. Kajian peneliti terdahulu .....	44
BAB 3 METODOLOGI PENELITIAN.....	49
9.1. Tahapan penelitian .....	49
3.1.1. Identifikasi topik penelitian .....	50
3.1.2. Studi literatur.....	50
3.1.3. Penyusunan latar belakang, permasalahan, tujuan, dan batasan penelitian .....	50
3.1.4. Rancangan sistematika metode penelitian .....	51
3.1.5. Rancangan konseptual model.....	52
3.1.6. Pengumpulan data .....	52
3.1.7. Analisis data .....	53
3.1.8. Implementasi .....	54
3.1.9. Pengujian.....	54
3.1.10. Hasil dan pembahasan penelitian.....	54
3.1.11. Penyusunan kesimpulan dan saran .....	54
BAB 4 MODEL KONSEPTUAL .....	57

4.1. Skema model Konseptual .....	57
<b>BAB 5 HASIL DAN PEMBAHASAN .....</b>	<b>59</b>
5.1. Analisa kebutuhan proyek permainan digital .....	59
5.1.1. Kebutuhan pada proyek P-MCP .....	60
5.1.2. Kebutuhan pada proyek P-MCE .....	106
5.1.3. Kebutuhan pada proyek P-MVN .....	159
5.1.4. Elemen aesthetic mendominasi permainan digital .....	175
5.2. Analisa biaya asli proyek permainan digital .....	176
5.3. Perhitungan estimasi biaya .....	177
5.3.1. Perhitungan Function point .....	178
5.3.2. Perhitungan effort .....	186
5.3.3. Perhitungan effort per aktivitas .....	187
5.3.4. Perhitungan biaya per aktivitas .....	188
5.3.5. Analisa gap hasil estimasi metode FPA .....	194
5.4. Estimasi biaya menggunakan metode FPA-MSP .....	196
5.4.1. Metode Story Point (MSP) .....	197
5.4.2. Memodifikasi besaran nilai effort dengan MSP .....	214
5.4.3. Perhitungan effort per aktivitas (FPA-MSP) .....	220
5.4.4. Perhitungan biaya per aktivitas (FPA-MSP) .....	221
5.4.5. Analisa gap hasil estimasi metode FPA-MSP .....	222
5.4.6. Hasil FPA-MSP vs Harga Asli .....	225
5.4.7. FPA-MSP sebagai solusi metode estimasi biaya .....	226
<b>BAB 6 KESIMPULAN DAN SARAN .....</b>	<b>229</b>
6.1. Kesimpulan .....	229
6.2. Saran .....	230
<b>DAFTAR PUSTAKA .....</b>	<b>231</b>
<b>LAMPIRAN .....</b>	<b>237</b>
Lampiran 1 - perhitungan FPA .....	237
Lampiran 2 – penilaian kompleksitas dengan MSP .....	263
Lampiran 3 – rancangan pengerjaan proyek .....	274

## DAFTAR GAMBAR

Gambar 2.1. Project management triagle.....	12
Gambar 2.2. Level Abstraksi Function Point dalam Aplikasi .....	15
Gambar 2.3. Tahapan metode FPA .....	17
Gambar 2.4. Konsep transaction function.....	20
Gambar 2.5. Kosep data function.....	22
Gambar 3.1. Tahapan penelitian .....	49
Gambar 4.1. Model konseptual .....	57
Gambar 5.1. Tujuan permainan P-MCP.....	61
Gambar 5.2. Alur permainan P-MCP.....	62
Gambar 5.3. Pengembangan konsep permainan .....	64
Gambar 5.4. Elemen dynamic pada interface Pilihan Level.....	73
Gambar 5.5. Fitur Pause pada Interface Gameplay.....	75
Gambar 5.6. Kondisi pathfinding kereta biru pada objek rel.....	75
Gambar 5.7. Elemen <i>Aesthetic</i> pada interface Home.....	81
Gambar 5.8. Elemen visual dalam interface Pilihan Level.....	84
Gambar 5.9. Tampilan tingkatan level pada interface Gameplay .....	86
Gambar 5.10. Perbandingan konsep tingkatan level.....	87
Gambar 5.11. Hubungan state dengan fase permainan .....	88
Gambar 5.12. Elemen visual interface Gameplay di <i>state playing</i> .....	89
Gambar 5.13. Elemen visual interface Gameplay di <i>state paused</i> .....	90
Gambar 5.14. Elemen visual interface Gameplay di <i>state finished</i> .....	91
Gambar 5.15. Indikator nilai pada permainan Fruit Ninja.....	99
Gambar 5.16. Indikator nilai pada permainan Stack.....	100
Gambar 5.17. Objektif pertama permainan digital P-MCE .....	106
Gambar 5.18. Objektif kedua permainan digital P-MCE.....	107
Gambar 5.19. Alur permainan P-MCE .....	109
Gambar 5.20. Kategori barang yang dilarang .....	110
Gambar 5.21. Peraturan perilaku penumpang.....	110
Gambar 5.22. Fungsi untuk mematikan/menghidupkan BGM .....	123
Gambar 5.23. Tombol navigasi pada peraturan penumpang.....	123
Gambar 5.24. Tombol navigasi tutorial .....	125
Gambar 5.25. Fitur pause di interface Gameplay SBB.....	127
Gambar 5.26. Elemen visual interface Pilihan Permainan.....	132
Gambar 5.27. Elemen visual interface Pilihan Level SBB .....	135
Gambar 5.28. Tampilan awal animasi interface Pilihan Level .....	135
Gambar 5.29. Animasi pada objek awan .....	136
Gambar 5.30. Animasi awal permainan SBB .....	138
Gambar 5.31. Tampilan saat permainan SBB berjalan.....	139
Gambar 5.32. Tampilan saat permainan SBB berhenti sementara.....	140
Gambar 5.33. Tampilan saat permainan SBB selesai .....	141
Gambar 5.34. Tampilan fase start Gameplay PP .....	145
Gambar 5.35. Fase pertama Gameplay PP .....	145
Gambar 5.36. Tampilan peringatan permainan PP .....	146

Gambar 5.37. Fase kedua Gameplay PP.....	146
Gambar 5.38. Hasil interaksi dengan meja lipat.....	147
Gambar 5.39. Pilihan opsi untuk sabuk pengaman .....	147
Gambar 5.40. Tampilan permainan saat sabuk pengaman benar .....	148
Gambar 5.41. Cuplikan permainan digital P-MVN.....	160
Gambar 5.42. Alur Proyek P-MVN.....	161
Gambar 5.43. Unsur dinamis interface Home .....	163
Gambar 5.44. Opsi pilihan yang merubah alur cerita.....	165
Gambar 5.45. Ilustrasi yang menggantikan tombol.....	167
Gambar 5.46. Contoh alur cerita pada VN Steins:Gate.....	169
Gambar 5.47. Contoh visual dalam cerita VN Steins;Gate .....	172
Gambar 5.48. Contoh visual dalam cerita pada Proyek P-MVN.....	172
Gambar 5.49. Presentase komponen dalam permainan digital.....	176
Gambar 5.50. Perbandingan harga pada proyek P-MCP dan P-MCE.....	194
Gambar 5.51. Perbandingan harga pada proyek P-MVN.....	195
Gambar 5.52. Rentang bilangan fibonacci yang digunakan .....	198
Gambar 5.53. Perbandingan dua kebutuhan berdasarkan resiko.....	201
Gambar 5.54. Perbandingan kebutuhan berdasarkan tugas .....	203
Gambar 5.55. Rencana pengembangan proyek P-MVN .....	212
Gambar 5.56. Karakteristik proyek vs produktivitas.....	214
Gambar 5.57. Perbandingan estimasi FPA, FPA-MSP, dan harga asli .....	223
Gambar 5.58. Perbandingan hasil estimasi FPA, FPA-MSP dan harga asli .....	224
Gambar 5.59. FPA-MSP sebagai varian metode FPA .....	226

## DAFTAR TABEL

Tabel 2.1. Tahapan dalam fase pengembangan permainan digital .....	10
Tabel 2.2. Nilai kompleksitas EI.....	19
Tabel 2.3. Nilai kompleksitas EI dan EQ.....	19
Tabel 2.4. Nilai bobot EI, EO, dan EQ berdasarkan kompleksitasnya .....	20
Tabel 2.5. Nilai kompleksitas ILF dan EIF.....	21
Tabel 2.6. Nilai bobot ILF dan EIF berdasarkan kompleksitasnya.....	22
Tabel 2.7. Perhitungan nilai FP.....	23
Tabel 2.8. General System Characteristic .....	24
Tabel 2.9. Degree of influence dari data communication .....	25
Tabel 2.10. Degree of influence dari Distributed data processing.....	26
Tabel 2.11. Degree of influence dari performance .....	27
Tabel 2.12. Degree of influence dari heavily used configuration .....	27
Tabel 2.13. Degree of influence dari transaction rate .....	28
Tabel 2.14. Degree of influence dari online data entry.....	29
Tabel 2.15. Degree of influence dari enduser efficiency .....	30
Tabel 2.16. Degree of influence dari online update .....	31
Tabel 2.17. Degree of influence dari complex processing.....	32
Tabel 2.18. Degree of influence dari reusability .....	32
Tabel 2.19. Degree of influence dari installation ease.....	33
Tabel 2.20. Degree of influence dari operational ease.....	34
Tabel 2.21. Degree of influence dari multiple sites .....	34
Tabel 2.22. Degree of influence dari facilitate change .....	36
Tabel 2.23. Contoh Alokasi Story Point .....	38
Tabel 2.24. Contoh perhitungan Velocity .....	39
Tabel 2.25. Distribusi Effort per Aktivitas .....	42
Tabel 3.1. Detail Re: tale game studio .....	52
Tabel 5.1. Overview proyek P-MCP.....	61
Tabel 5.2. Kebutuhan pada interface Home.....	63
Tabel 5.3. Kebutuhan pada interface Pilihan Level .....	63
Tabel 5.4. Pengembangan konsep berdasarkan tiga fase permainan .....	65
Tabel 5.5. Kebutuhan pada Interface Gameplay .....	66
Tabel 5.6. Daftar kebutuhan proyek P-MCP.....	69
Tabel 5.7. Update kebutuhan interface Home (Dynamic) .....	72
Tabel 5.8. Update kebutuhan interface Pilihan Level (Dynamic).....	74
Tabel 5.9. Update kebutuhan inferface Gameplay (Dynamic) .....	76
Tabel 5.10. Update daftar kebutuhan proyek P-MCP (Dynamic).....	77
Tabel 5.11. Update daftar kebutuhan interface Home (Aesthetic).....	82
Tabel 5.12. Update kebutuhan Interface Pilihan Level (Aesthetic).....	85
Tabel 5.13. Update kebutuhan interface Gameplay (Aesthetic) .....	92
Tabel 5.14. Update daftar kebutuhan proyek P-MCP (Aesthetic) .....	94
Tabel 5.15. Daftar kebutuhan final proyek P-MCP .....	102
Tabel 5.16. Overview proyek P-MCE.....	107
Tabel 5.17. Kebutuhan pada Interface Pilihan Permainan.....	111

Tabel 5.18. kebutuhan pada Interface Pilihan Level SBB.....	112
Tabel 5.19. Pengembangan konsep berdasarkan fase permainan.....	113
Tabel 5.20. Kebutuhan pada Interface Gameplay SBB.....	114
Tabel 5.21. Kebutuhan pada Interface Gameplay PP.....	117
Tabel 5.22. Daftar kebutuhan proyek P-MCE.....	119
Tabel 5.23. Update kebutuhan Pilihan Permainan (Dynamic).....	124
Tabel 5.24. Update kebutuhan Pilihan Level SBB (Dynamic).....	125
Tabel 5.25. Update kebutuhan Gameplay SBB (Dynamic).....	127
Tabel 5.26. Update kebutuhan Gameplay PP (Dynamic).....	129
Tabel 5.27. Update kebutuhan Pilihan Permainan (Aesthetic).....	132
Tabel 5.28. Update kebutuhan Pilihan Level SBB (Aesthetic).....	136
Tabel 5.29. Update kebutuhan Gameplay SBB (Aesthetic).....	141
Tabel 5.30. Update kebutuhan Gameplay PP (Aesthetic).....	148
Tabel 5.31. Daftar kebutuhan final proyek P-MCE.....	151
Tabel 5.32. Daftar kebutuhan Interface Home.....	161
Tabel 5.33. Daftar kebutuhan interface Story.....	162
Tabel 5.34. Update kebutuhan interface Home (Dynamic).....	164
Tabel 5.35. Update kebutuhan interface Story (Dynamic).....	165
Tabel 5.36. Update kebutuhan Interface Home (Aesthetic).....	167
Tabel 5.37. Update kebutuhan interface Story (Aesthetic).....	170
Tabel 5.38. Daftar kebutuhan final proyek P-MVN.....	173
Tabel 5.39. Detail proyek permainan digital.....	177
Tabel 5.40. Klasifikasi fungsi dan bobot proyek P-MCP.....	179
Tabel 5.41. Perhitungan nilai FP Proyek P-MCP.....	185
Tabel 5.42. Nilai FP ketiga proyek permainan digital.....	186
Tabel 5.43. Effort per proyek permainan digital.....	187
Tabel 5.44. Nilai usaha pada setiap aktifitas proyek.....	187
Tabel 5.45. Daftar posisi pada studio re:tale.....	189
Tabel 5.46. Referensi gaji pengembang permainan digital.....	190
Tabel 5.47. Perkiraan gaji pada studio re:tale.....	190
Tabel 5.48. Pay rate untuk setiap posisi.....	192
Tabel 5.49. Pay rate per aktivitas permainan digital.....	193
Tabel 5.50. Estimasi biaya menggunakan FPA murni.....	194
Tabel 5.51. Gap estimasi dengan harga asli pada genre casual.....	195
Tabel 5.52. Tugas yang dihitung diabaikan pada interface Home.....	199
Tabel 5.53. Contoh draf pertanyaan kepada pengembang.....	203
Tabel 5.54. Nilai SP proyek P-MCP.....	204
Tabel 5.55. Penilaian FPA vs MSP.....	210
Tabel 5.56. Akumulasi nilai SP pada ketiga proyek.....	211
Tabel 5.57. Rancangan pengerjaan proyek P-MVN.....	212
Tabel 5.58. Produktivitas tim proyek.....	213
Tabel 5.59. Produktivitas per hari tim pengembang.....	216
Tabel 5.60. Produktivitas per jam tim pengembang.....	217
Tabel 5.61. Referensi kebutuhan sebagai medium perhitungan.....	218
Tabel 5.62. Nilai PDR pada ketiga proyek.....	219
Tabel 5.63. Nilai effort baru.....	219
Tabel 5.64. Nilai usaha pada setiap aktivitas proyek (FPA-MSP).....	220

Tabel 5.65. Pay rate per aktivitas permainan digital (FPA-MSP).....	221
Tabel 5.66. Estimasi biaya menggunakan FPA-MSP .....	222
Tabel 5.67. Gap estimasi metode FPA-MSP dengan harga asli .....	223

[HALAMAN INI SENGAJA DIKOSONGKAN]

# **BAB 1**

## **PENDAHULUAN**

Bab pendahuluan mendefinisikan beberapa poin, yaitu latar belakang penelitian, perumusan masalah, tujuan penelitian, batas penelitian dan sistematika penelitian.

### **1.1. Latar belakang**

Industri permainan digital sedang berada pada periode popularnya. Pernyataan tersebut berdasarkan pada fakta yang dihimpun oleh Superdata bahwa revenue global dari industri permainan digital di tahun 2017 mencapai USD 108.4 milyar (Batchelor, 2018). Revenue industri permainan digital tahun 2017 mengalami peningkatan dibandingkan tahun 2016 yang hanya berkisar USD 91 milyar (Sinclair, 2016). Revenue yang dihasilkan pada tahun 2016 bahkan melampaui capaian pendapatan kombinasi dua segmen industri entertainment lainnya yaitu industri film dan musik (Malim, 2018). Fakta tersebut selaras dengan pernyataan Callele et al. (2005) bahwa segmen permainan digital menjadi bagian terpenting dalam industri entertainment.

Salah satu faktor pendorong terbesar dalam popularitas industri permainan digital adalah antusiasme. Survey yang dilakukan Newzoo (2017) di Indonesia membuktikan dari jumlah total 255.7 juta jiwa masyarakat Indonesia, 66 juta diantaranya termasuk dalam kategori sadar teknologi dan sekitar 64.8% atau 42.8 juta dari kategori sadar teknologi tersebut merupakan penggemar permainan digital (*gamer*). Antusiasme masyarakat yang besar tersebut menempatkan Indonesia pada peringkat 16 dalam daftar pasar game terbesar didunia dengan potensi penghasilan sebesar USD 879.7 juta (Newzoo, 2017), Indonesia naik 8 peringkat dari tahun 2015 dengan total pendapatan saat itu sebesar USD 321 juta (Newzoo, 2015).

Namun, diantara ratusan proyek permainan digital, sebagian besar proyek tersebut gagal mencapai pasar dan hanya beberapa yang sampai ketangan konsumen (Callele et al., 2005). Selain karena faktor persaingan yang semakin ketat, mayoritas kegagalan ratusan proyek permainan digital terjadi sebelum mencapai pasar atau pada tahap pengembangan proyek permainan digital (Callele et al., 2005; Kanode

and Haddad, 2009). Salah satu faktor yang berperan atas kegagalan pada tahap pengembangan proyek perangkat lunak adalah biaya. Mengutip dari statista.com (2019) sebesar 63.16% dari keseluruhan proyek permainan digital yang telah didanai penuh melalui portal website *kickstarter.com* mengalami kegagalan. Website *kickstarter.com* sendiri merupakan medium yang memungkinkan pengembang mendapatkan sumber dana dari dari calon investor secara langsung. Pengembang dapat menentukan target biaya yang ingin didapatkan untuk mendanai sebuah proyek pengembangan secara langsung. Akan tetapi hanya sebesar 36.84% proyek yang berhasil didanai mengalami kesuksesan (statista.com, 2019). Hal ini mengindikasikan pentingnya memiliki estimasi biaya yang tepat sebelum melanjutkan proyek permainan digital ke tahap pengembangan.

Perencanaan proyek yang kurang tepat menjadi faktor utama penyebab kegagalan pada sebuah pengembangan proyek perangkat lunak (Kurniawan et al., 2013; Primandari A and Sholiq, 2015; Saleh, 2011). *Software Cost Estimation* dapat dijadikan solusi untuk menyelesaikan permasalahan tersebut. *Software Cost Estimation* adalah usaha untuk memprediksi biaya yang diperlukan dalam mengembangkan suatu perangkat lunak. Li et al. (2009) menyatakan bahwa software cost estimation berperan penting pada keberhasilan proyek perangkat lunak.

Variabel utama yang yang wajib diketahui untuk melakukan perhitungan *Software Cost Estimation* adalah *Effort*. *Effort* merupakan nilai ukur yang merepresentasikan total usaha yang dilakukan untuk sebuah proyek pengembangan perangkat lunak. Besaran *Effort* dipengaruhi ukuran perangkat lunak dan *Project Delivery Rate* (PDR) atau seberapa cepat team dalam mengerjakan per satuan size (International Software Benchmarking Standards Group (ISBSG), 2011).

Pengukuran perangkat lunak merupakan aktivitas dalam rekayasa perangkat lunak yang digunakan untuk mengestimasi ukuran perangkat lunak sehingga memudahkan aktivitas manajemen proyek perangkat lunak (Ng'ang'a and Tonui, 2015). Apabila ukuran diestimasi dengan benar maka estimasi *effort* dan biaya juga akan menjadi realistis (Galorath and Evans, 2006). Metode yang paling populer dalam upaya estimasi ukuran perangkat lunak adalah metode *Function*

*Point Analysis* (FPA) (Bagriyanik and Karahoca, 2016; Kumar et al., 2019; Symons, 2001).

Function point pertamakali diperkenalkan oleh Alan Albrecht (1979) dan telah digunakan pada 22 proyek pada rentang tahun 1974 hingga 1978 di IBM. Albrecht menyatakan bahwa dengan penggunaan function point dapat meningkatkan produktivitas hingga 50 persen per tahun (Albrecht, 1979). Pada tahun 1986, *International Function Point User Group* (IFPUG) didirikan untuk menyebarkan penggunaan FPA dan menjadi badan penanggung jawab untuk mengimprovisasi dan melakukan standarisasi peraturan terkait FPA (2014). Peraturan dan panduan perhitungan FPA disusun dalam *Counting Practice Manual* (CPM). Metode FPA semula dirancang untuk mengukur produktivitas pada pengembangan perangkat lunak (Albrecht, 1979; Albrecht and Gaffney, 1983; Symons, 1988). namun seiring dengan perkembangannya, metode FPA kemudian digunakan untuk mengestimasi *effort* dan biaya (Dewi et al., 2018; Di Martino et al., 2016; Kumar et al., 2019; Putri and Subriadi, 2018; Vijay, 2018).

Metode FPA mengukur ukuran dari perangkat lunak berdasarkan pada input dan output spesifik yang disediakan untuk pengguna (Verner et al., 1989), metode FPA juga menggunakan sudut pandang pengguna pada sistem sehingga penilaian ukuran software bergantung pada terpenuhinya atau tidak value bisnis dari pengguna (Albrecht and Gaffney, 1983; Meli, 1997; Symons, 1988). Selain menjadi sebuah metode untuk mengukur ukuran perangkat lunak, nama Function Point (FP) juga digunakan sebagai satuan ukuran perangkat lunak yang dihitung menggunakan metode FPA (Albrecht and Gaffney, 1983; Pradani, 2014). FP merupakan nilai *dimensionless* yang merepresentasikan ukuran fungsionalitas sistem untuk pengguna (Bock and Klepper, 1992).

Berdasarkan penelitian untuk mengestimasi ukuran, *effort*, atau biaya dari permainan digital didapatkan bahwa Penelitian yang dilakukan oleh pada medium permainan digital membuktikan bahwa estimasi *effort* yang dihasilkan menggunakan perhitungan *Function Point* memiliki deviasi sebesar 376% dari *effort* sesungguhnya (2018). Dalam penelitian lain yang dilakukan oleh Abdullah et al. (2014) dengan menggunakan “anak” dari metode FPA yaitu COSMIC FP untuk mengestimasi ukuran permainan digital apabila dibangun dengan menggunakan

game engine Unity 3D juga membuktikan estimasi ukuran didapatkan hanya berdasarkan pergerakan data (data movement) yang berarti dilihat dari sisi fungsional saja dan tidak mempertimbangkan unsur non-fungsional yang lain.

Berdasarkan karakteristik perangkat lunak permainan digital, faktor non-fungsional memiliki porsi yang lebih besar daripada faktor fungsional. Faktor - faktor non-fungsional tersebut antara lain adalah orientasi terhadap teknologi (Blow, 2004; Zackariasson and Wilson, 2010), faktor teknis pelaksanaan seperti pembuatan model 3D, tekstur, animasi, efek, suara, musik, dialog, penggunaan teknologi third-party hingga penyusunan kombinasi team (Kanode and Haddad, 2009). Tingginya dependensi kepada faktor non-fungsional tersebut menjadikan proyek permainan digital atau secara umum proyek multimedia berbeda dengan proyek-proyek lainnya (Kumar et al., 2019). Dependensi terhadap faktor non-fungsional tersebut menimbulkan permasalahan. FPA ternyata memiliki kelemahan dalam komponen non-fungsional sistem. Komponen non-fungsional dalam metode FPA yang diwakili dalam 14 komponen *General System Characteristics* (GSC) (Albrecht and Gaffney, 1983) ternyata tidak memberikan dampak signifikan, dinilai telah kadaluarsa, dan jarang sekali digunakan (Heemstra and Kusters, 1991; Jeffery et al., 1993; Lokan, 2000; Symons, 2001). Hal tersebut menyebabkan komponen non-fungsional yang lebih dominan pada proyek multimedia (Kumar et al., 2019) tidak dapat diukur dengan benar.

Kapabilitas individu (skill dan pengetahuan) berperan penting dalam pengembangan sebuah proyek perangkat lunak (Symons, 1988). Dalam proyek permainan digital, kemampuan individu memiliki pengaruh besar dalam faktor non-fungsional seperti gambar, naskah cerita, desain karakter, dan faktor non-fungsional lainnya. Dengan pengaruh faktor non-fungsional yang besar dalam permainan digital, maka kapabilitas sekumpulan individu (team) juga menjadi sangat penting untuk keberhasilan pengembangan proyek permainan digital. Solusi yang diajukan untuk menghitung faktor non-fungsional pada penelitian ini adalah dengan mengukur kapabilitas unik tiap individu untuk menyelesaikan sebuah tugas (*task*). Metode *Story Point* (MSP) dipilih peneliti karena memiliki kapabilitas tersebut.

MSP merupakan metode yang biasa dipakai dalam metodologi Agile, konsep kerja metode ini adalah memecah sistem menjadi *task* terkecil dan menilai

masing masing tugas sebuah nilai subjektif yaitu *Story Point* (SP) yang mencerminkan tingkat kesulitan dari *task* tersebut berdasarkan sudut pandang team pengembang proyek. Karena menggunakan sudut pandang team, maka nilai SP pada setiap *task* telah mengandung tidak hanya faktor fungsionalitasnya (fitur fungsional yang diberikan kepada user), namun juga faktor non-fungsional (faktor penunjang seperti availability, continuity dan security), dan teknis pelaksanaannya (bahasa pemrograman yang digunakan, tools yang dipakai) dan disesuaikan dengan kemampuan individu tersebut. Sehingga MSP dapat digunakan untuk mengestimasi lama pengerjaan masing-masing *task*. Estimasi lama pengerjaan kemudian akan menghasilkan keluaran berupa produktivitas team (*Velocity*) yang digunakan untuk memodifikasi besaran effort sehingga bisa didapatkan estimasi biaya yang mengikutsertakan dan mempertimbangkan faktor non fungsional pada proyek permainan digital.

Berdasarkan uraian diatas, peneliti bertujuan menggunakan metode Story Point (MSP) untuk membantu dalam proses mendapatkan besaran biaya pada proyek permainan digital. Metode Story Point akan digunakan untuk mendapatkan nilai *Project Delivery Rate* (PDR) yang kemudian digunakan untuk mengkalkulasi nilai *Effort*. Hasil besaran Effort tersebut kemudian digunakan untuk mendapatkan besaran estimasi biaya pada proyek perangkat lunak permainan digital.

## **1.2. Rumusan masalah**

Berdasarkan latar belakang yang telah dijabarkan, perumusan masalah pada penelitian ini adalah

1. Bagaimana cara mengestimasi biaya perangkat lunak menggunakan metode gabungan function point dengan story point.
2. Berapakah hasil evaluasi gap antara estimasi harga perangkat lunak permainan digital menggunakan metode Function Point dan Story Point dan harga sesungguhnya.

## **1.3. Tujuan penelitian**

Penelitian ini memiliki dua tujuan utama. Pertama, untuk menentukan bagaimana cara menghitung besaran Effort menggunakan metode gabungan FPA

dan MSP. Kedua, untuk menilai akurasi dari estimasi biaya perangkat lunak game yang dihasilkan menggunakan kombinasi metode FPA dan MSP.

#### **1.4. Lingkup penelitian**

Batasan-batasan dan ruang lingkup dalam penelitian yang dilakukan adalah sebagai berikut:

1. Pengukuran perangkat lunak yang menjadi obyek penelitian berada pada fase awal pengembangan (*Early Phase Development*).
2. Rumus FPA yang digunakan adalah rumus yang dipakai pada fase pengembangan.
3. Jenis obyek penelitian yang digunakan merupakan perangkat lunak permainan digital (*video game*) skala kecil-medium.
4. Tim developer memiliki referensi atau pengalaman sebelumnya dalam mengembangkan permainan digital.
5. Skala perusahaan adalah small-medium.
6. Project yang digunakan sebagai sample untuk menilai akurasi adalah project yang telah siap rilis di pasar.

#### **1.5. Kontribusi penelitian**

Kontribusi penelitian dibagi menjadi dua yaitu kontribusi di bidang keilmuan dan kontribusi praktis.

##### **1.5.1. Kontribusi bidang keilmuan**

Kontribusi yang dapat diberikan pada penelitian yang dilakukan adalah konsep perhitungan *Effort* menggunakan kombinasi metode *Function Point* dan *Story Point* sehingga dapat digunakan untuk melakukan estimasi biaya terutama pada perangkat lunak permainan digital.

##### **1.5.2. Kontribusi praktis**

Kontribusi yang dapat diberikan pada penelitian ini diharapkan dapat membantu developer dan Project Manager dalam melakukan estimasi pengukuran biaya perangkat lunak permainan digital agar dapat memenuhi target waktu dan biaya yang ditentukan.

## **1.6. Sistematika penulisan**

Sistematika penulisan dari penelitian ini adalah sebagai berikut:

### **1.6.1. Bab 1 Pendahuluan**

Bab ini terdiri dari latar belakang penelitian, perumusan masalah, tujuan penelitian, kontribusi penelitian, batasan penelitian, dan sistematika penulisan.

### **1.6.2. Bab 2 Kajian Pustaka**

Bab ini berisi kajian yang meliputi teori-teori dan penelitian yang sudah ada terkait dengan topik penelitian.

### **1.6.3. Bab 3 Metodologi Penelitian**

Bab ini membahas mengenai rancangan penelitian dan juga tahapan-tahapan sistematis yang digunakan selama melakukan penelitian.

### **1.6.4. Bab 4 Konseptual Model**

Bab ini menjelaskan rancangan model konseptual yang peneliti ajukan dalam penelitian ini.

### **1.6.5. Bab 5 Hasil Dan Pembahasan**

Bab ini menjelaskan implementasi dari rancangan model konseptual yang dirancang pada bab 4, kemudian menyusun hasil atau pembahasan, hasil dari implementasi dan pengujian yang telah tervalidasi digunakan untuk menjawab pertanyaan penelitian.

### **1.6.6. Bab 5 Kesimpulan Dan Saran**

Bab ini berisi kesimpulan dan saran untuk penelitian yang akan datang. Sehingga penelitian ini dapat terus dikembangkan kedepannya

[HALAMAN INI SENGAJA DIKOSONGKAN]

## **BAB 2**

### **TINJAUAN PUSTAKA**

Bab tinjauan pustaka menjelaskan kajian pustaka untuk menunjang penelitian. Kajian pustaka terdiri dari kajian teori dan penelitian-penelitian yang telah dilakukan sebelumnya.

#### **2.1. Kajian teori**

Kajian teori membahas mengenai teori dasar dan konsep yang terkait dengan penelitian guna mendapatkan landasan konstruksi teoritis sebagai pedoman dan tolak ukur penelitian. Kajian teori pada penelitian ini meliputi Industri permainan digital, Pengukuran Perangkat Lunak, metode Function Point Analysis (FPA), metode Story Point (MSP), dan Estimasi Biaya Perangkat Lunak.

##### **2.1.1. Industri permainan digital**

Sebelum menjadi konsep permainan digital atau video game yang kita kenal sekarang, ternyata permulaan gagasan tersebut berawal dari permainan kartu tradisional sederhana yang diciptakan oleh Fusajiro Yamauchi di tahun 1889. Kemudian pada tahun 1961, konsep permainan digital interaktif pada platform komputer yang bernama “*Spacewar*” diciptakan oleh Steve Russell, seorang mahasiswa Massachusetts Institute of Technology (MIT) (Kent, 2010). *Spacewar* menjadi permainan pertama yang terdigitalisasi.

Setelah hampir 60 tahun sejak pertamakali diluncurkan, permainan digital telah menjadi sektor penting dalam industri entertainment (Callele et al., 2005). Revenue di industry permainan digital merupakan yang terbaik jika dibandingkan bidang entertainment lainnya seperti Musik dan film ditahun 2016 (Malim, 2018). Lebih menariknya lagi, dalam industri permainan digital bukan para inventor ataupun pengembang yang mendapatkan keuntungan paling besar, melainkan individu atau kelompok yang menjadikannya sebuah peluang bisnis (Zackariasson and Wilson, 2010). Oleh karena itu, tidak heran industri game juga mendapat perhatian lebih dari para investor (Maulana and Kurniawan, 2018).

Dalam proses untuk menciptakan sebuah permainan digital, tentunya memiliki tahapan-tahapan yang tidak jauh berbeda dengan pengembangan perangkat lunak lainnya. Pada Tabel 2.1 disajikan tahapan-tahapan yang dilakukan dalam proses pengembangan permainan digital.

Tabel 2.1. Tahapan dalam fase pengembangan permainan digital

Tahapan	Aktivitas	Sumber
Pre-produksi	<ul style="list-style-type: none"> <li>• Overview : goal/ objektif dari permainan</li> <li>• Mechanic : elemen game seperti karakter, mekanisme permainan, daftar aset</li> <li>• Dynamic : interaksi game seperti interface, level atau artificial intelligence</li> <li>• Aesthetic : apa yang pemain dapat lihat, dengar, atau yang didapatkan dari panca indranya (gambar, suara, animasi)</li> <li>• Experience : ekspektasi dari pemain</li> <li>• Assuptions and constrain : batasan teknis yang mempengaruhi desain game.</li> </ul>	(Blow, 2004) (Callele et al., 2005) (Kanode and Haddad, 2009) (Keith and Cohn, 2010) (Salazar et al., 2012)
produksi	<ul style="list-style-type: none"> <li>• Organisasi team/ kombinasi team</li> <li>• Teknologi third-party yang digunakan</li> <li>• Spesifikasi teknis</li> <li>• Implementasi/konstruksi</li> <li>• Qulity test</li> </ul>	(Callele et al., 2005) (Kanode and Haddad, 2009) (Petrillo and Pimenta, 2010) (Keith and Cohn, 2010)

	<ul style="list-style-type: none"> <li>• Gameplay test</li> <li>• Alpha dan beta test</li> <li>• Final version release</li> </ul>	
--	---	--

Dari ratusan proyek permainan digital, sebagian besar proyek gagal mencapai pasar dan hanya beberapa yang sampai tangan konsumen (Callele et al., 2005). Mayoritas kegagalan ratusan proyek permainan digital terjadi pada tahapan pengembangan permainan digital (Callele et al., 2005; Kanode and Haddad, 2009). Dalam pengembangan perangkat lunak permainan digital, selain aspek fungsional pada permainan digital seperti bagaimana konsep kerja sebuah permainan digital (*gameplay*), berbagai hal lainnya yang harus diperhitungkan mulai dari perencanaan proyek, seperti besar lingkup proyek game yang akan dibuat dan tingkat kompleksitasnya (Blow, 2004), depedensi dan orientasi teknologi yang akan digunakan (Blow, 2004; Zackariasson and Wilson, 2010). Hingga hal teknis pelaksanaan, seperti keberagaman aset yang diperlukan seperti model 3D, tekstur, animasi, efek suara, musik, dialog, dsb, penyusunan anggota team, dan penggunaan teknologi third-party (Kanode and Haddad, 2009).

### **2.1.2. Pengukuran perangkat lunak**

Untuk mencapai keberhasilan proyek perangkat lunak yang dapat diandalkan dalam pengembangan perangkat lunak, perencana perangkat lunak memerlukan rencana yang dapat dicapai yaitu perkiraan jadwal, ukuran sumber daya, dan risiko yang layak (Galorath and Evans, 2006).

Permasalahan yang sering terjadi dalam proses pengembangan perangkat lunak adalah Proyek membutuhkan waktu yang lebih lama dari yang diperkirakan (*time*), membutuhkan biaya lebih dari rencana awal, dan ketidaksesuaian fungsi yang dibuat dengan apa diminta oleh stakeholder (*scope*). Hubungan *Cost*, *Scope*, dan *Time* ini disebut dengan Project Management Triangle (Cole and Scotcher, 2016). Gambar 2.1 merupakan ilustrasi dari Project Management Triangle.



Gambar 2.1. Project management triagle

(sumber: <https://www.smartsheet.com/triple-constraint-triangle-theory>)

Keterkaitan ketiga hubungan tersebut mengakibatkan apabila salah satu parameter mengalami perubahan maka akan mempengaruhi parameter yang lain. Dalam prakteknya yang sering terjadi pada proyek seperti ketika ada penambahan requirement (scope), maka akan mengakibatkan terganggunya penjadwalan (time) dan menyebabkan bertambahnya biaya (cost). Selama bertahun-tahun team proyek berusaha untuk menyeimbangkan ketiga constrain tersebut (Cole and Scotcher, 2016). Berdasarkan Putri dan Subriadi (2018), alasan utama dibalik kegagalan proyek dan masalah perangkat lunak terjadi karena hal berikut:

1. Ketidakmampuan mengukur proyek perangkat lunak dengan akurat
2. Ketidakmampuan menentukan pengembangan perangkat lunak dan lingkungan yang sesuai secara akurat
3. Penentuan level kepegawaian dan keterampilan yang tidak tepat
4. Kebutuhan yang kurang terdefinisi dengan baik untuk perkiraan aktivitas perangkat lunak

Menurut Gencel dan Demirors (2008) alasan pentingnya melakukan pengukuran perangkat lunak adalah:

1. Ukuran perangkat lunak adalah ukuran utama yang digunakan sebagai masukan dalam memperkirakan biaya dan usaha
2. Ukuran yang diestimasi dan ukuran sebenarnya dibandingkan untuk memantau pencapaian proyek
3. Kontrak juga dibentuk dengan menggunakan ukuran pengukuran perangkat lunak
4. Ukuran perangkat lunak juga digunakan dalam normalisasi tindakan lainnya

Pengukuran perangkat lunak merupakan aktivitas dalam rekayasa perangkat lunak yang digunakan untuk mengestimasi ukuran komponen perangkat lunak agar dapat mengimplementasikan aktivitas manajemen proyek perangkat lunak (Ng'ang'a and Tonui, 2015). Apabila ukuran yang diestimasi benar maka estimasi *effort* akan realistis dan menjadi perkiraan biaya yang realistis (Galorath and Evans, 2006).

#### **2.1.2.1. Metode pengukuran perangkat lunak**

*Engineering Research Community* telah banyak mengusulkan pendekatan dan teknik pengukuran perangkat lunak. Mengutip dari Putri dan Subriadi (2018) Perencana proyek perangkat lunak menggunakan pendekatan ini untuk memperkirakan biaya proyek yang tepat untuk mengembangkan produk perangkat lunak. Teknik pengukuran perangkat lunak dapat dikategorikan menjadi dua, yaitu metrik pengukuran berbasis kode dan pengukuran ukuran fungsional (FSM). Metrik pengukuran berbasis kode mengukur ukuran perangkat lunak menggunakan program source code meliputi LOC (Lines of Code), Halstead metric (Halstead, 1977), dan McCabe's Cyclometric Complexity (McCabe, 1976). Pengukuran ukuran fungsional (FSM) merupakan metode pengukuran berbasis kode yang memperhitungkan aspek statis dan dinamis dari sistem (Nguyen, 2010), metode Function Point merupakan salah satu metode FSM. Selain Function Point, metode-metode yang termasuk FSM adalah Use Case Point, COCOMO II, Test Point, Object Point dan lain-lain (Borade and Khalkar, 2013).

### 2.1.3. Function Point Analysis (FPA)

Function Point adalah metode yang memecah sistem menjadi komponen yang lebih kecil, sehingga dapat dipahami dan dianalisis lebih baik (Longstreet, 2002). Metode Function Point menggunakan pendekatan berorientasi pada fungsionalitas dan kompleksitas untuk memperkirakan ukuran perangkat lunak pada masa pengembangan perangkat lunak (Pratiwi, 2013). Function Point Analysis (FPA) merupakan bagian dari FSM (Pengukuran Ukuran Fungsional) yang merupakan alternatif untuk metode pengukuran berbasis kode (LOC) yang memperhitungkan aspek statis dan dinamis dari sistem (Nguyen, 2010). Metode Function Point pertama kali diperkenalkan oleh Allan Albrecht (1979) dengan judul “*Measuring Application Development Productivity*” di IBM. Pada tahun 1986 No. Function Point User Group (IFPUG) dibentuk dan hingga sekarang telah menerbitkan beberapa versi petunjuk manual Penghitungan Function Point.

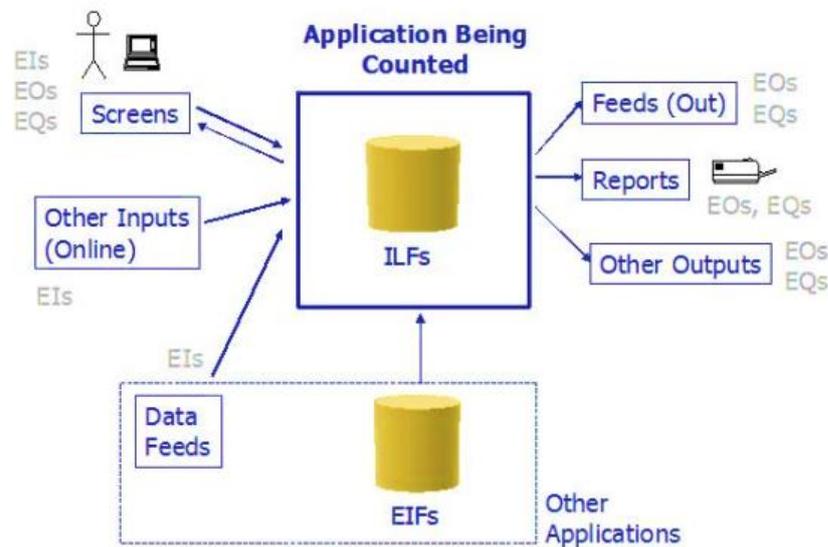
Menurut Longstreet (2002), metode *Function Point Analysis* membantu mendefinisikan dua level abstrak data pada level konseptual (Gambar 2.2), yaitu *data at rest* dan *data in motion*.

#### a. *Data In Motion*

*Data in motion* merupakan jenis data yang merupakan fungsi transaksional atau transaksi sederhana. Transaksi (*elementary processes*) yang membawa data dari luar domain aplikasi (*application boundary*) kedalam *application boundary* yang disebut sebagai *external input* (EI). Sedangkan transaksi Transaksi (atau *elementary processes*) yang mengambil data dari *resting position* (biasanya pada *file*) keluar domain aplikasi (*application boundary*) disebut juga sebagai *external inquiries* (EQ).

#### b. *Data At Rest*

*Data at rest* merupakan jenis data dikelola. Apabila data dikelola oleh aplikasi utama, maka data tersebut diklasifikasikan sebagai *internal logic file* (ILF). Sedangkan data yang dikelola oleh aplikasi lain diluar batasan aplikasi utama diklasifikasikan sebagai *external interface files* (EIF).



Gambar 2.2. Level Abstraksi Function Point dalam Aplikasi  
(Sumber:alvinalexander.com/FunctionPoints/FunctionPoints.shtml)

### 2.1.3.1. Kegunaan function point analysis

Function Point ideal dalam estimasi biaya perangkat lunak karena selain telah memiliki standarisasi ISO, metode FPA dapat digunakan pada fase awal pengembangan (Xia et al., 2008) seperti pada fase requirement (International Function Point Users Group (IFPUG), 2010), lebih lanjut berdasarkan Longstreet (2002), kegunaan *Function Point* adalah :

1. Function Points dapat digunakan untuk berkomunikasi lebih efektif dengan kelompok pengguna bisnis.
2. Function Points dapat digunakan untuk mengurangi waktu lembur.
3. Function Points dapat digunakan untuk membuat inventory semua transaksi dan file dari proyek atau aplikasi saat ini. Inventory ini bisa dijadikan alat evaluasi finansial suatu aplikasi. Jika inventory dilakukan untuk development project atau enhancement project, inventory yang sama dapat digunakan untuk membantu menjaga scope creep dan untuk

membantu mengendalikan pertumbuhan proyek. Yang lebih penting lagi, inventory ini membantu memahami besarnya masalah.

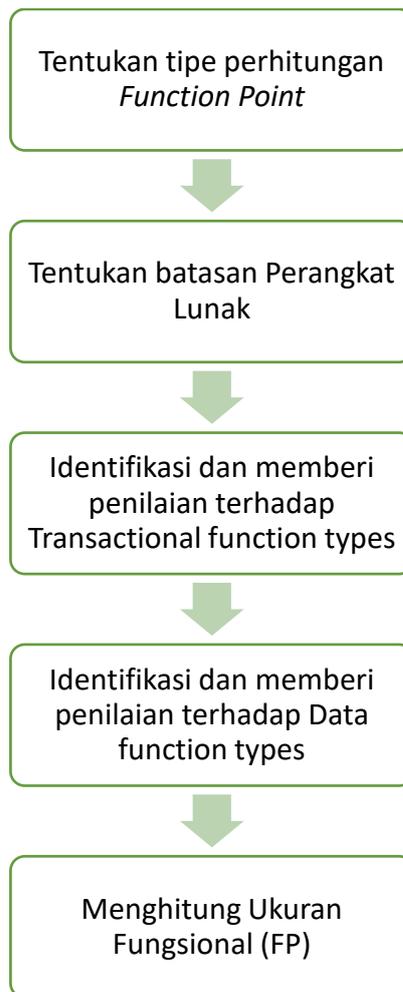
4. Function Points dapat digunakan untuk mengukur aplikasi software. Ukuran merupakan komponen penting dalam menentukan produktivitas (output / input), memprediksi usaha, memahami biaya unit, dan sebagainya.
5. Tidak seperti beberapa metrik perangkat lunak lainnya, orang yang berbeda dapat menghitung Function Points pada waktu yang berbeda, untuk mendapatkan ukuran yang sama dalam selisih kesalahan yang wajar. Artinya, kesimpulan yang sama akan diambil dari hasilnya.
6. FPA dapat membantu organisasi memahami biaya unit dari aplikasi perangkat lunak atau proyek. Begitu biaya unit dipahami alat, bahasa, platform dapat dibandingkan secara kuantitatif daripada secara subyektif. Jenis analisis ini jauh lebih mudah dipahami daripada informasi teknis. Jadi pengguna non teknis dapat memahami Function Points

#### **2.1.3.2. Tahap perhitungan function point**

Berdasarkan manual IFPUG (International Function Point Users Group (IFPUG), 2010), langkah-langkah untuk mendapatkan ukuran perangkat lunak (ukuran fungsional) diilustrasikan dalam Gambar 2.3.

#### **Tahap 1 – Tentukan tipe perhitungan Function Point**

*Function Point* memiliki tiga jenis perhitungan yaitu : *Development Function Point Count*, *Enhancement Function Point Count*, dan *Application Function Point Count* yang sejalan dengan jenis proyek perangkat lunak utama yaitu *Development*, *Enhancements*, dan *Maintenance* (International Function Point Users Group (IFPUG), 2010; Longstreet, 2002).



Gambar 2.3. Tahapan metode FPA

**a. Development Function Point Count**

*Development Function Point Count* atau *Baseline Function Point Count* merupakan jenis perhitungan *Function Point* yang dilakukan pada fase pembuatan perangkat lunak. pada tipe ini perhitungan *Function Point* dapat dilakukan pada fase Requirement hingga tahap implementasi.

**b. Enhancement Function Point Count**

*Enhancement Function Point Count* merupakan tipe perhitungan *Function Point* yang digunakan pada kasus penambahan satu atau lebih fitur baru pada perangkat lunak yang telah selesai diproduksi.

### c. **Application Function Point Count**

*Application Function Point Count* merupakan tipe perhitungan *Function Point* yang digunakan untuk mendapatkan besaran perangkat lunak yang telah diproduksi sebelumnya.

Jenis *Function Point* yang digunakan pada penelitian ini adalah *Development Function Point*.

### **Tahap 2 – Tentukan batas perangkat lunak**

Batasan perangkat lunak dapat ditentukan dengan menetapkan *logical boundary* antara perangkat lunak yang diukur, pengguna, dan sistem perangkat lunak lainnya yang berkaitan (International Function Point Users Group (IFPUG), 2010). Perlu diperhatikan bahwa penentuan batasan bergantung pada pandangan bisnis pengguna. Karena keputusan didasarkan kepada pandangan bisnis pengguna, maka batasan bisa menjadi subyektif dan menyebabkan kesulitan untuk menggambarkan bagaimana atau di mana perangkat lunak dimulai dan berakhir (Vazquez et al., 2003).

### **Tahap 3 – Menghitung Fungsi transaksi (transaction function)**

*Transaction function* atau fungsi transaksi menggambarkan sejumlah fungsionalitas sistem yang disediakan untuk *end user* yang digunakan untuk memproses data oleh aplikasi (International Function Point Users Group (IFPUG), 2010). *Transaction Function* dibagi menjadi tiga tipe, yaitu:

- a. **External Input** – merupakan proses dasar dimana data melintasi batasan aplikasi dari luar menuju kedalam aplikasi. Data mungkin berasal dari layar input maupun berasal dari aplikasi lain. Data digunakan untuk mengelola satu atau lebih Internal Logic File atau No. perilaku perangkat lunak (International Function Point Users Group (IFPUG), 2010).
- b. **External Output** – merupakan proses dasar dimana data melewati boundary aplikasi dari dalam ke luar dan berfungsi untuk menampilkan informasi kepada pengguna. Informasi yang dihasilkan

merupakan hasil dari sebuah algoritma atau perhitungan tertentu (International Function Point Users Group (IFPUG), 2010).

- c. **External Inquiries** – merupakan proses dasar untuk menampilkan informasi dari satu atau lebih internal logic file atau external interface file (International Function Point Users Group (IFPUG), 2010).

Berdasarkan CPM manual (International Function Point Users Group (IFPUG), 2010), langkah-langkah untuk mengukur nilai dari *Transaction Function* adalah :

- 1) Mendefinisikan aktivitas menjadi unit aktivitas terkecil.
- 2) Melakukan klasifikasi unit aktivitas terkecil menjadi EI, EO, atau EQ.
- 3) Menghitung jumlah *Data Element Type* (DET) dan *File Type Reference* (FTR) pada setiap unit EI, EO, dan EQ.
- 4) Menentukan nilai kompleksitas masing-masing EI, EO, dan EQ menggunakan tabel referensi.

Tabel 2.2. Nilai kompleksitas EI

File Type Reference (FTR)	Data Element Type (DET)		
	1-4	5-15	>15
<2	Low	Low	Average
2	Low	Average	High
>2	Average	High	High

Sumber : IFPUG, 2010

Penentuan tingkat kompleksitas EI disajikan pada Tabel 2.2, sedangkan nilai kompleksitas EO dan EQ disajikan pada Tabel 2.3.

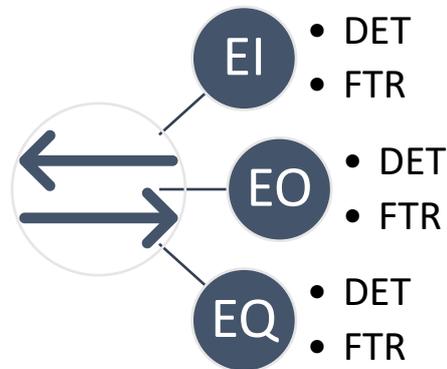
Tabel 2.3. Nilai kompleksitas EI dan EQ.

File Type Reference (FTR)	Data Element Type (DET)		
	1-5	6-19	>19
<2	Low	Low	Average
2 – 3	Low	Average	High

File Type Reference (FTR)	Data Element Type (DET)		
	1-5	6-19	>19
>3	Average	High	High

Sumber : IFPUG, 2010

Nilai kompleksitas dari semua *Transaction Function* sangat dipengaruhi oleh jumlah *Data Element Type* (DET) yang merupakan jumlah field dalam database yang dirujuk oleh fungsi dan *File Type Reference* (FTR) yang merupakan tabel yang direferensi oleh sebuah fungsi seperti yang diilustrasikan pada Gambar 2.4.



Gambar 2.4. Konsep transaction function

Representasi nilai bobot dari nilai kompleksitas masing-masing EI, EO, dan EQ disajikan dalam Tabel 2.4.

Tabel 2.4. Nilai bobot EI, EO, dan EQ berdasarkan kompleksitasnya

Function Type	Kompleksitas		
	Low	Medium	High
EI	3	4	6
EO	4	5	7
EQ	3	4	6

Sumber : IFPUG, 2010

#### Tahap 4 – Menghitung fungsi data (*data function*)

*Data function* atau fungsi data merepresentasikan kumpulan data logic yang dibutuhkan *end user*. (International Function Point Users Group (IFPUG), 2010). *Data Function* terdiri dari dua tipe, yaitu :

- a. **Internal Logical File** – *Internal Logical File* (ILF) merupakan data ataupun kontrol informasi yang seluruhnya berada dalam batasan perangkat lunak dan dikelola oleh satu atau lebih proses dasar (International Function Point Users Group (IFPUG), 2010; Longstreet, 2002).
- b. **External Interface File** – *External Internal File* (EIF) merupakan data atau kontrol informasi yang direferensi oleh satu atau lebih proses dasar dan berada diluar batasan perangkat lunak.(International Function Point Users Group (IFPUG), 2010; Longstreet, 2002).

Berdasarkan CPM IFPUG, langkah-langkah untuk menentukan *Data Function* adalah:

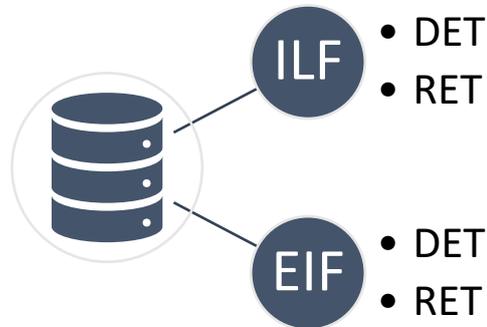
- 1) Mengelompokkan tabel-tabel data berdasarkan user logic.
- 2) Melakukan klasifikasi terhadap kelompok-kelompok tabel apakah sebagai ILF atau EIF.
- 3) Menentukan jumlah *Data Element Type* (DET) dan *Record Element Type* (RET) untuk setiap ILF dan EIF.
- 4) Menentukan nilai kompleksitas masing-masing ILF dan EIF menggunakan tabel referensi yang disajikan pada Tabel 2.5.

Tabel 2.5. Nilai kompleksitas ILF dan EIF

Record Element Types (RET)	Data Element Type (DET)		
	1-19	20-50	>50
1	Low	Low	Average
2-5	Low	Average	High
>5	Average	High	High

Sumber : IFPUG, 2010

Nilai data function ( ILF dan EIF) sangat bergantung kepada nilai DET dan RET yang merupakan sebuah primary key yang terdapat pada tabel yang digunakan pada sebuah fungsi seperti yang diilustrasikan pada Gambar 2.5.



Gambar 2.5. Kosep data function

Representasi nilai bobot dari nilai kompleksitas masing-masing ILF dan EIF direpresentasikan dalam Tabel 2.6.

Tabel 2.6. Nilai bobot ILF dan EIF berdasarkan kompleksitasnya.

Function Type	Kompleksitas		
	Low	Medium	High
ILF	7	10	15
EIF	5	7	10

Sumber : IFPUG, 2010

### Tahap 5 – Menghitung ukuran fungsional (nilai function point)

Nilai fungsional dari perangkat lunak didapatkan dengan menjumlahkan seluruh nilai bobot dari Transaction function (EI,EO, dan EQ) dan Data Function (ILF dan EIF). Tabel 2.7 menyajikan contoh template pembobotan FP yang masih kosong untuk mempermudah perhitungan nilai function point.

Tabel 2.7. Perhitungan nilai FP

Function Type	Kompleksitas						Total
	Low		Average		High		
	(Val) x Count	Sum	(Val) x Count	Sum	(Val) x Count	Sum	
EI	(3) x _	_	(4) x _	_	(6) x _	_	_
EO	(4) x _	_	(5) x _	_	(7) x _	_	_
EQ	(3) x _	_	(4) x _	_	(6) x _	_	_
ILF	(7) x _	_	(10) x _	_	(15) x _	_	_
EIF	(5) x _	_	(7) x _	_	(10) x _	_	_
	<b>Function Point</b>						-

Sumber : IFPUG, 2010

Persamaan untuk mendapatkan nilai Function Point (FP) pada perangkat lunak adalah :

$$FP = EI + EO + EQ + ILF + EIF \dots\dots\dots(1)$$

**2.1.3.3. Value Adjustment Factor (Opsional)**

IFPUG juga menyediakan langkah opsional untuk melakukan proses penyesuaian (*adjusting*) dengan menambahkan perhitungan *Value Adjustment Factor* (VAF). Perhitungan VAF ini bertujuan untuk menghitung faktor nonfungsional dalam perangkat lunak (International Software Benchmarking Standards Group (ISBSG), 2011). VAF dihitung berdasarkan nilai dari 14 *General System Characteristics* (GSC). Persamaan untuk mendapatkan nilai VAF adalah :

$$VAF = (TDI * 0.01) + 0.65 \dots\dots\dots(2)$$

Nilai TDI (Total Degree of Influence) merupakan tingkat pengaruh indikator-indikator GSC dalam perangkat lunak. tingkat pengaruh tersebut direpresentasikan secara numerik, dari angka 0 (tidak berpengaruh) hingga 5

(sangat berpengaruh). Berdasarkan IFPUG, GSC terdiri dari 14 item seperti yang disajikan pada Tabel 2.8.

Tabel 2.8. General System Characteristic

No	GSC	Deskripsi
1	Data Communication	Berapa banyak fasilitas komunikasi yang ada untuk membantu mentransfer atau bertukar informasi dengan aplikasi atau sistem?
2	Distributed data processing	Bagaimana data terdistribusi dan fungsi pemrosesan ditangani?
3	Performance	Apakah pengguna memerlukan <i>respose time</i> atau <i>throughput</i> ?
4	Heavily used configuration	Seberapa banyak penggunaan platform perangkat keras saat ini dimana sistem akan dieksekusi?
5	Transaction rate	Seberapa sering transaksi dieksekusi setiap hari, setiap minggu, setiap bulan, dan lainlain?
6	On-line data entry	Berapakah persentase informasi yang dimasukkan secara online?
7	End-user efficiency	Apakah aplikasi dirancang untuk efisiensi end user?
8	On-line update	Berapa banyak ILF yang diperbaharui oleh transaksi online?
9	Complex processing	Apakah aplikasi memiliki extensive logical atau mathematical processing?
10	Reusability	Apakah aplikasi dikembangkan untuk memenuhi satu atau lebih kebutuhan pengguna?
11	Installation ease	Seberapa sulit konversi dan instalasi?

No	GSC	Deskripsi
12	Operational ease	Seberapa efektif dan/atau otomatis <i>start-up</i> , <i>back up</i> , dan prosedur <i>recovery</i> ?
13	Multiple sites	Apakah aplikasi dirancang, dikembangkan dan didukung terutama untuk diinstal pada <i>multiple sites</i> untuk beberapa organisasi?
14	Facilitate change	Apakah aplikasi dirancang, dikembangkan dan didukung untuk memudahkan perubahan?

Selanjutnya, peneliti akan menjelaskan untuk masing-masing komponen pada General System Characteristic (GSC) berdasarkan manual yang diterbitkan oleh IFPUG (2002) yaitu sebagai berikut:

#### **Data communication**

Tingkat kebutuhan transfer data antara aplikasi dan processor. Tabel 2.9 menunjukkan deskripsi masing-masing bobot untuk menentukan degree of influence dari data communication.

Tabel 2.9. Degree of influence dari data communication

<b>Bobot</b>	<b>Deskripsi untuk menentukan besarnya pengaruh</b>
0	Aplikasi adalah pemrosesan batch murni atau standalone PC.
1	Aplikasi bersifat batch namun memiliki remote entri data atau remote printing.
2	Aplikasi bersifat batch namun memiliki remote entri data dan remote printing.
3	Aplikasi meliputi pengumpulan data online atau TP (teleprocessing) front end ke proses batch atau sistem query.
4	Aplikasi lebih dari sekedar front-end, tapi hanya mendukung satu jenis protokol komunikasi TP.

<b>Bobot</b>	<b>Deskripsi untuk menentukan besarnya pengaruh</b>
5	Aplikasi lebih dari sekedar front-end, dan mendukung lebih dari satu jenis protokol komunikasi TP.

### **Distributed data processing**

Data terdistribusi atau fungsi pemrosesan merupakan karakteristik aplikasi dalam boundary / lingkup dalam aplikasi. Tabel 2.10 menunjukkan deskripsi masing-masing bobot untuk menentukan degree of influence dari distributed data processing.

Tabel 2.10. Degree of influence dari Distributed data processing

<b>Bobot</b>	<b>Deskripsi untuk menentukan besarnya pengaruh</b>
0	Aplikasi tidak membantu transfer data atau pengolahan fungsi antar komponen sistem.
1	Aplikasi menyiapkan data untuk pemrosesan enduser pada komponen yang lain dari sistem seperti PC spreadsheets dan PC DBMS.
2	Data disiapkan untuk transfer, kemudian ditransfer dan diproses pada komponen lain dari sistem (bukan untuk Pengolahan enduser).
3	Pengolahan data dan transfer data secara online dan satu arah Saja.
4	Pengolahan dan transfer data terdistribusi secara online dan masuk kedua arah.
5	Fungsi pemrosesan secara dinamis ditampilkan pada komponen paling banyak yang sesuai dari sistem.

### **Performance**

Tujuan kinerja aplikasi, dinyatakan atau disetujui oleh pengguna, baik dalam respon maupun throughput, pengaruh (atau akan mempengaruhi) desain, pengembangan, pemasangan, dan dukungan aplikasi. Tabel 2.11 menunjukkan deskripsi masing-masing bobot untuk menentukan degree of influence dari performance.

Tabel 2.11. Degree of influence dari performance

<b>Bobot</b>	<b>Deskripsi untuk menentukan besarnya pengaruh</b>
0	Tidak ada kebutuhan kinerja khusus yang dinyatakan oleh pengguna.
1	Kebutuhan kinerja dan rancangan dinyatakan serta ditinjau tapi tidak ada tindakan khusus yang diperlukan.
2	Waktu respon atau throughput sangat penting selama jam sibuk. Tidak ada rancangan khusus untuk penggunaan CPU yang dibutuhkan. Batas waktu proses adalah untuk hari kerja berikutnya.
3	Waktu respon atau throughput sangat penting selama semua business hours. Tidak ada rancangan khusus untuk penggunaan CPU yang dibutuhkan. Memproses kebutuhan deadline dengan interfacing sistem merupakan menghambat.
4	Selain itu, kebutuhan kinerja pengguna yang disebutkan cukup ketat untuk membutuhkan performance analysis task di fase desain.
5	Selain itu, alat analisis kinerja digunakan di tahap desain, pengembangan, dan/atau implementasi untuk mencapai kebutuhan kinerja pengguna yang disebutkan.

### **Heavily used configuration**

Konfigurasi operasional yang banyak digunakan, yang memerlukan pertimbangan desain khusus, yaitu karakteristik aplikasi. Misalnya, pengguna ingin menjalankan aplikasi yang ada atau committed equipment yang akan banyak digunakan. Tabel 2.12 menunjukkan deskripsi masing-masing bobot untuk menentukan degree of influence dari heavily used configuration.

Tabel 2.12. Degree of influence dari heavily used configuration

<b>Bobot</b>	<b>Deskripsi untuk menentukan besarnya pengaruh</b>
0	Tidak ada batasan operasional eksplisit atau implisit yang disertakan.
1	Pembatasan operasional memang ada, tapi kurang ketat dari pada sebuah aplikasi tertentu. Tidak ada upaya khusus yang diperlukan untuk memenuhi pembatasan.

<b>Bobot</b>	<b>Deskripsi untuk menentukan besarnya pengaruh</b>
2	Beberapa pertimbangan keamanan atau waktu disertakan.
3	Kebutuhan prosesor khusus untuk bagian tertentu dari aplikasi disertakan.
4	Batasan operasi yang diatur memerlukan batasan khusus aplikasi di central processor atau dedicated prosesor.
5	Selain itu, ada kendala khusus pada aplikasi dalam komponen terdistribusi dari sistem.

### **Transaction rate**

Tingkat transaksi tinggi dan mempengaruhi desain, pengembangan, pemasangan, dan dukungan dari aplikasi Tabel 2.13 menunjukkan deskripsi masing-masing bobot untuk menentukan degree of influence dari transaction rate.

Tabel 2.13. Degree of influence dari transaction rate

<b>Bobot</b>	<b>Deskripsi untuk menentukan besarnya pengaruh</b>
0	Tidak ada periode transaksi puncak yang diantisipasi.
1	Periode transaksi puncak (mis., Bulanan, kuartalan, musiman, setiap tahun) diantisipasi
2	Periode transaksi puncak mingguan diantisipasi
3	Periode transaksi puncak harian diantisipasi.
4	Tarif transaksi tinggi yang dinyatakan oleh pengguna dalam kebutuhan aplikasi atau perjanjian tingkat layanan cukup tinggi memerlukan tugas analisis kinerja pada tahap perancangan.
5	Tarif transaksi tinggi yang dinyatakan oleh pengguna dalam kebutuhan aplikasi atau perjanjian tingkat layanan cukup tinggi memerlukan tugas analisis kinerja dan memerlukan penggunaan alat analisis kinerja dalam desain, tahap pengembangan, dan / atau instalasi.

### **On-line data entry**

Online data entry dan control functions disediakan dalam aplikasi. Tabel 2.14 menunjukkan deskripsi masing-masing bobot untuk menentukan degree of influence dari online data entry.

Tabel 2.14. Degree of influence dari online data entry

<b>Bobot</b>	<b>Deskripsi untuk menentukan besarnya pengaruh</b>
0	Semua transaksi diproses dalam mode batch.
1	1% sampai 7% dari transaksi adalah entri data interaktif.
2	8% sampai 15% transaksi adalah entri data interaktif
3	16% sampai 23% transaksi adalah entri data interaktif.
4	24% sampai 30% transaksi adalah entri data interaktif
5	Lebih dari 30% transaksi adalah entri data interaktif

### **End-user efficiency**

Fungsi online yang diberikan menekankan desain untuk efisiensi pengguna akhir. Desainnya meliputi:

- a) Bantuan navigasi (misalnya, tombol fungsi, lompatan, menu yang dibuat secara dinamis)
- b) Menu
- c) Bantuan dan dokumen online
- d) Gerakan kursor otomatis
- e) Scrolling
- f) Remote printing (melalui transaksi online)
- g) Tombol fungsi yang telah ditentukan sebelumnya
- h) Batch jobs dikirim dari transaksi online
- i) Pemilihan kursor data layar
- j) Penggunaan reverse video, highlight, color underlining, dan indikator lainnya
- k) Dokumentasi pengguna hard copy transaksi online
- l) Antarmuka mouse

- m) Jendela pop-up.
- n) Layar sedikit mungkin untuk menyelesaikan fungsi bisnis
- o) Dukungan bilingual (mendukung dua bahasa; dihitung sebagai empat item)
- p) Dukungan multi bahasa (mendukung lebih dari dua bahasa; dihitung sebagai enam item)

Tabel 2.15 menunjukkan deskripsi masing-masing bobot untuk menentukan degree of influence dari enduser efficiency.

Tabel 2.15. Degree of influence dari enduser efficiency

<b>Bobot</b>	<b>Deskripsi untuk menentukan besarnya pengaruh</b>
0	Bukan dari salah satu dari kriteria desain yang disebutkan.
1	Satu sampai tiga dari kriteria desain yang disebutkan.
2	Empat sampai lima dari kriteria desain yang disebutkan.
3	Enam atau lebih hal dari kriteria desain yang disebutkan, tapi tidak ada kebutuhan pengguna khusus yang berkaitan dengan efisiensi.
4	Enam atau lebih kebutuhan dari kriteria desain yang disebutkan, dan pernyataan kebutuhan untuk efisiensi pengguna akhir cukup kuat untuk membutuhkan tugas desain Human tasks yang harus disertakan (misalnya meminimalkan keystrokes, memaksimalkan default, penggunaan template).
5	Enam atau lebih dari kriteria desain yang disebutkan, dan menyatakan kebutuhan untuk efisiensi pengguna akhir yang cukup kuat membutuhkan pemakaian alat dan proses khusus untuk menunjukkan bahwa tujuan telah tercapai.

### **On-line update**

Aplikasi menyediakan online update untuk internal logic file (ILF). Tabel 2.16 menunjukkan deskripsi masing-masing bobot untuk menentukan degree of influence dari online update.

Tabel 2.16. Degree of influence dari online update

<b>Bobot</b>	<b>Deskripsi untuk menentukan besarnya pengaruh</b>
0	Tidak ada.
1	Online update satu sampai tiga file kontrol disertakan. Volume update rendah dan recovery mudah dilakukan.
2	Online update dari empat atau lebih file kontrol disertakan. Volume update rendah dan recovery mudah.
3	Pembaruan online internal logic file utama disertakan.
4	Selain itu, perlindungan terhadap data yang hilang sangat penting dan telah dirancang serta diprogram secara khusus dalam sistem.
5	Selain itu, volume tinggi membawa pertimbangan biaya ke dalam proses recovery. Prosedur recovery yang sangat otomatis dengan intervensi operator minimal disertakan

### **Complex processing**

Complex processing merupakan ciri khas dari aplikasi. Komponen yang disajikan adalah sebagai berikut:

- a) Kontrol sensitif (misalnya, pemrosesan audit khusus) dan / atau pengolahan keamanan khusus aplikasi.
- b) Proses logis yang ekstensif.
- c) Pengolahan matematika secara ekstensif.
- d) Banyak proses pengecualian yang mengakibatkan transaksi tidak lengkap yang harus diolah kembali, misalnya transaksi ATM yang tidak lengkap yang disebabkan oleh gangguan TP, kehilangan nilai data, atau gagal mengedit.
- e) Pengolahan kompleks untuk menangani beberapa kemungkinan input/output, misalnya multimedia, atau independensi perangkat

Tabel 2.17 menunjukkan deskripsi masing-masing bobot untuk menentukan degree of influence dari complex processing.

Tabel 2.17. Degree of influence dari complex processing

<b>Bobot</b>	<b>Deskripsi untuk menentukan besarnya pengaruh</b>
0	Bukan dari salah satu di atas.
1	Salah satu di atas.
2	Dua hal di atas.
3	Tiga hal di atas.
4	Keempat hal di atas.
5	Kelima hal di atas.

### **Reusability**

Aplikasi dan kode dalam aplikasi telah dirancang, dikembangkan, dan didukung untuk dapat digunakan dalam aplikasi lain. Tabel 2.18 menunjukkan deskripsi masing-masing bobot untuk menentukan degree of influence dari reusability.

Tabel 2.18. Degree of influence dari reusability

<b>Bobot</b>	<b>Deskripsi untuk menentukan besarnya pengaruh</b>
0	Tidak ada kode yang dapat digunakan kembali.
1	Kode dapat digunakan kembali digunakan dalam aplikasi.
2	Kurang dari 10% aplikasi dianggap lebih dari satu kebutuhan Pengguna.
3	Sepuluh persen (10%) atau lebih dari aplikasi dipertimbangkan lebih dari satu kebutuhan pengguna.
4	Aplikasi dikemas secara khusus dan / atau didokumentasikan untuk memudahkan penggunaan ulang, dan aplikasi disesuaikan oleh pengguna pada tingkat kode sumber.
5	Aplikasi secara khusus dikemas dan / atau didokumentasikan untuk memudahkan penggunaan ulang, dan aplikasi disesuaikan untuk digunakan dengan cara pemeliharaan parameter pengguna

### **Installation ease**

Kemudahan konversi dan pemasangan adalah karakteristik dari aplikasi. Konversi dan rencana instalasi atau alat konversi disediakan serta diuji selama tahap uji sistem. Tabel 2.19 menunjukkan deskripsi masing-masing bobot untuk menentukan degree of influence dari installation ease.

Tabel 2.19. Degree of influence dari installation ease

<b>Bobot</b>	<b>Deskripsi untuk menentukan besarnya pengaruh</b>
0	Tidak ada pertimbangan khusus yang dikemukakan oleh pengguna, dan tidak ada pengaturan khusus yang diperlukan untuk instalasi.
1	Tidak ada pertimbangan khusus yang dikemukakan oleh pengguna namun pengaturan khusus diperlukan untuk instalasi.
2	Kebutuhan konversi dan pemasangan dinyatakan oleh pengguna, panduan konversi dan pemasangan disediakan serta diuji. Dampak konversi pada proyek tidak dianggap penting.
3	Kebutuhan konversi dan pemasangan dinyatakan oleh pengguna, dan panduan konversi dan pemasangan disediakan dan diuji. Dampak konversi pada proyek ini adalah dianggap penting.
4	Selain 2 di atas, konversi dan pemasangan otomatis alat disediakan dan diuji.
5	Selain 3 di atas, konversi dan pemasangan otomatis alat disediakan dan diuji.

### **Operational ease**

Kemudahan operasional adalah karakteristik dari aplikasi. Prosedur start-up, back up, dan recovery yang efektif diberikan dan diuji selama fase uji sistem. Aplikasi meminimalkan kebutuhan akan kegiatan manual, seperti tape mount, penanganan kertas, dan intervensi manual langsung di lokasi. Tabel 2.20 menunjukkan deskripsi masing-masing bobot untuk menentukan degree of influence dari operational ease.

Tabel 2.20. Degree of influence dari operational ease.

<b>Bobot</b>	<b>Deskripsi untuk menentukan besarnya pengaruh</b>
0	Tidak ada pertimbangan operasional khusus selain prosedur backup yang normal dinyatakan oleh pengguna
1- 4	<p>Satu, beberapa, atau semua hal berikut berlaku untuk aplikasi. Pilih semua yang berlaku. Setiap item ada benarnya nilai satu, kecuali seperti yang dinyatakan lain.</p> <ul style="list-style-type: none"> <li>• Proses start-up, back-up, dan recovery yang efektif disediakan, tapi intervensi operator diperlukan.</li> <li>• Proses start-up, back-up, dan recovery yang efektif disediakan, tapi tidak ada intervensi operator yang diperlukan (dihitung sebagai dua item).</li> <li>• Aplikasi ini meminimalkan kebutuhan akan tape mount</li> <li>• Aplikasi ini meminimalkan kebutuhan penanganan kertas</li> </ul>
5	Aplikasi ini dirancang untuk operasi yang tidak dijaga. Operasi tanpa pengawasan berarti tidak ada intervensi operator diperlukan untuk mengoperasikan sistem selain untuk start up atau shut down aplikasi. Pemulihan error otomatis adalah fitur dari aplikasi.

### Multiple sites

Aplikasi ini telah dirancang, dikembangkan, dan didukung secara khusus untuk dipasang di beberapa situs untuk beberapa organisasi. Tabel 2.21 menunjukkan deskripsi masing-masing bobot untuk menentukan degree of influence dari multiple sites.

Tabel 2.21. Degree of influence dari multiple sites

<b>Bobot</b>	<b>Deskripsi untuk menentukan besarnya pengaruh</b>
0	Kebutuhan pengguna tidak perlu mempertimbangkan kebutuhan lebih dari satu pengguna / situs instalasi.

<b>Bobot</b>	<b>Deskripsi untuk menentukan besarnya pengaruh</b>
1	Kebutuhan beberapa situs dipertimbangkan dalam desain, dan aplikasi dirancang untuk beroperasi hanya dibawah lingkungan perangkat keras dan perangkat lunak yang identik.
2	Kebutuhan beberapa situs dipertimbangkan dalam desain, dan aplikasi dirancang untuk beroperasi hanya dibawah lingkungan perangkat keras dan / atau perangkat lunak yang serupa.
3	Kebutuhan beberapa situs dipertimbangkan dalam perancangan, dan aplikasi ini dirancang untuk beroperasi di bawah lingkungan perangkat keras dan / atau perangkat lunak yang berbeda
4	Rencana dokumentasi dan dukungan disediakan serta diuji untuk mendukung aplikasi di beberapa situs dan aplikasinya seperti yang dijelaskan pada 1 atau 2.
5	Rencana dokumentasi dan dukungan disediakan serta diuji untuk mendukung aplikasi di beberapa situs dan aplikasinya seperti yang dijelaskan oleh 3.

### **Facilitate change**

Aplikasi ini telah dirancang secara khusus, dikembangkan, dan didukung untuk memudahkan perubahan. Karakteristik berikut dapat diterapkan untuk aplikasi:

- a) Fasilitas query dan laporan fleksibel yang disediakan dapat menangani permintaan sederhana; sebagai contoh, dan / atau logika hanya diterapkan pada satu internal logic file (dihitung sebagai satu item).
- b) Fasilitas query dan laporan fleksibel yang disediakan dapat menangani permintaan dengan kompleksitas rata-rata, misalnya, dan / atau logika yang diterapkan pada lebih dari satu internal logic file (dihitung sebagai dua item).
- c) Fasilitas query dan laporan fleksibel yang disediakan dapat menangani permintaan yang rumit, misalnya, dan / atau kombinasi logika pada satu atau lebih internal logic file (dihitung sebagai tiga item).

- d) Data kontrol bisnis disimpan dalam tabel yang dikelola oleh pengguna dengan proses interaktif secara online, tapi perubahan hanya berlaku pada hari kerja berikutnya.
- e) Data kontrol bisnis disimpan dalam tabel yang dikelola oleh pengguna dengan proses interaktif secara online, dan perubahan berlaku segera (dihitung sebagai dua item).

Tabel 2.22 menunjukkan deskripsi masing-masing bobot untuk menentukan degree of influence dari facilitate change.

Tabel 2.22. Degree of influence dari facilitate change

<b>Bobot</b>	<b>Deskripsi untuk menentukan besarnya pengaruh</b>
0	Bukan dari salah satu di atas
1	Salah satu di atas.
2	Dua hal di atas.
3	Tiga hal di atas.
4	Keempat hal di atas.
5	Kelima hal di atas.

Namun perhitungan dengan menggunakan VAF saat ini sudah ditinggalkan karena IFPUG sedang mengembangkan metode baru untuk menggantikan fungsi VAF pada perhitungan perangkat lunak yaitu metode *Software Non-Functional Assessment Process* (SNAP) (International Function Point Users Group (IFPUG), 2019).

Dalam penelitian ini, peneliti tidak menggunakan perhitungan VAF untuk mengestimasi ukuran perangkat lunak (nilai function point). Peneliti melakukan alternatif untuk mendapatkan nilai non-fungsional dan teknis dengan menggunakan metode Story Point.

#### **2.1.4. Story Point**

Story point merupakan metode yang digunakan pada metodologi *SCRUM*. Metodologi SCRUM memandang perangkat lunak merupakan kumpulan sejumlah

fitur (International Software Benchmarking Standards Group (ISBSG), 2011) atau disebut *Project Backlog* (PB) dan memiliki metode pengembangan secara berulang dan bertahap (Tipu and Zia, 2012). SCRUM membagi produk perangkat lunak menjadi beberapa iterasi yang disebut dengan *sprint*. Setiap sprint memiliki daftar pekerjaan (Task) yang disebut *sprint backlog* yang harus diselesaikan sehingga sprint tersebut dinyatakan selesai (Tipu and Zia, 2012). Pekerjaan (task) merupakan fungsi terkecil dari fitur perangkat lunak. Sebagai contoh apabila terdapat fitur untuk mengelola akun pengguna maka task-nya terdiri atas menambah akun pengguna (add), memodifikasi akun pengguna (modify), dan menghapus akun pengguna (delete). Setiap task memiliki nilai kompleksitas menggunakan nilai story point.

Pada subbab selanjutnya akan dibahas secara detail langkah-langkah pada metodologi SCRUM berdasarkan ISBSG (2011).

#### **2.1.4.1. Menentukan skala story point**

Tahapan ini merupakan inisiasi nilai numerik yang akan digunakan untuk menjadi skala tingkat kesulitan setiap task pada proyek, nilai skala yang umumnya digunakan adalah bilangan *fibonacci* (Jørgensen, 2013), power of two (International Software Benchmarking Standards Group (ISBSG), 2011), dan ukuran T-shirt (Mistry, 2017). Pada penelitian ini peneliti menggunakan bilangan fibonacci sebagai skala kompleksitas task-task pada proyek permainan digital karena dinilai memiliki range yang lebih lebar dan dinilai lebih jelas bagi anggota tim pengembang.

#### **2.1.4.2. Kalibrasi skala story point**

Tahapan ini bertujuan untuk mendiskusikan dan menganalisa desain kebutuhan pada Proyek berdasarkan pada dokumen SRS (*software requirements specification*) atau pada proyek permainan digital disebut GDD (*game design document*). Tahapan ini menghasilkan daftar sprint backlog yang memiliki nilai Story Point (SP) sebagai indikator kompleksitas setiap task.

Pemberian nilai dilakukan dengan metode *planning poker* (Raith et al., 2013) yang terdiri dari empat langkah. Pertama, estimator menuliskan bilangan

fibonnaci dari bilangan terkecil yaitu 1 hingga bilangan terbesar pada set kartu; bilangan terbesar yang digunakan dalam penelitian ini adalah 32. Kedua, set kartu tersebut dibagikan kepada seluruh tim yang terlibat dalam pengembangan proyek perangkat lunak. Ketiga, setiap task dari proyek akan dipresentasikan satu-persatu dan masing-masing individu akan membobot task tersebut menggunakan set kartu, hal tersebut mengimplikasikan kompleksitas task berdasarkan sudut pandang individu. Terakhir. Apabila telah terjadi kesepakatan pembobotan pada sebuah task maka estimator akan mendokumentasikan pembobotan tersebut dan akan dilanjutkan ke task selanjutnya hingga seluruh task pada sebuah proyek perangkat lunak mendapat nilai SP. Tabel 2.23 merupakan contoh alokasi story point.

Tabel 2.23. Contoh Alokasi Story Point

No.	Story	Task	Story Point
1	Sebagai pemain saya ingin data tersinkronisasi dengan akun google saya	Membangun integrasi dengan google service	8
2	Sebagai pemain saya ingin memberi inisial kepada karakter saya	Membuat tampilan untuk memasukkan nama pemain	3
3	Sebagai pemain saya ingin menyimpan data permainan pada device saya	Membangun database untuk menyimpan data pemain	5

Story point adalah pengukuran yang relatif dengan tim pengembangan proyek, sehingga perhitungan antara tim A yang terdiri dari individu dengan jam terbang tinggi dengan tim B dengan jam terbang medium pada proyek perangkat lunak tertentu dapat menjadi berbeda (International Software Benchmarking Standards Group (ISBSG), 2011).

Dikarenakan nilai SP merupakan nilai yang relatif, maka potensi terjadi perbedaan dalam penentuan nilai SP pada setiap task proyek perangkat lunak oleh masing-masing individu pada satu tim cukup tinggi. Apabila terjadi perbedaan

tersebut maka solusi penyelesaiannya adalah individu yang memberikan nilai tertinggi dan terendah akan menjelaskan alasannya mengapa memberikan nilai tersebut. Langkah selanjutnya adalah memberi nilai ulang hingga tercapai kesepakatan.

### 2.1.4.3. Menghitung velocity

Velocity adalah rasio upaya penyelesaian PB pada setiap sprintnya (International Software Benchmarking Standards Group (ISBSG), 2011) atau jumlah *Story Point* per iterasinya. Iterasi yang digunakan dalam proyek perangkat lunak umumnya adalah setiap 2 minggu atau 2 bulan. Tabel 2.24 menyajikan contoh perhitungan velocity pada sebuah proyek perangkat lunak.

Tabel 2.24. Contoh perhitungan Velocity

Iterasi	Jumlah Story Per Iteration	Jumlah Story Point	
		setiap Story	Setiap iterasi
1	2	13,13	26
2	3	13, 8, 2	23
3	6	5, 5, 5, 3, 3, 2	22
4	6	8, 3, 2, 2, 2, 2	19
5	8	8, 3, 3, 3, 1, 1, 1, 1	21
<b>Total</b>	25		111
<b>Rerata</b>	5		22.2

Sumber : ISBSG

Berdasarkan contoh perhitungan yang dilakukan pada Tabel 2.24 mengindikasikan bahwa pada iterasi pertama (sprint 1) tim mengerjakan 2 story (PB) yang masing- masing memiliki nilai kompleksitas 13. Sehingga dalam sprint tersebut maka nilai *Velocity* pada tim adalah 22.2

### 2.1.5. Estimasi biaya perangkat lunak

Estimasi biaya perangkat lunak berperan penting dalam kesuksesan manajemen proyek perangkat lunak (Li et al., 2009). Pada dasarnya estimasi biaya perangkat lunak dapat memprediksi kebutuhan satu atau lebih faktor yaitu effort,

durasi proyek dan harga dalam pengembangan perangkat lunak (Leung and Fan, 2002). Tahapan yang dilakukan untuk melakukan estimasi biaya perangkat lunak pada penelitian ini mengacu pada langkah yang digunakan oleh Putri dan Subriadi (2018) dengan modifikasi pada bagian menentukan effort proyek.

**2.1.5.1. Menentukan effort proyek**

Perhitungan effort yang digunakan dalam penelitian ini menngkombinasikan persamaan ISBSG dengan persamaan yang digunakan Putri dan Subriadi (2018) yaitu :

$$Effort = Size * Project Delivery Rate * 8.2.....(3)$$

Dimana nilai Size merupakan nilai Function Point yang dihasilkan pada tahapan metode Function Point. Sedangkan PDR merupakan durasi penyelesaian per size Function Point (fp/hour). Nilai konstanta 8.2 merupakan nilai effort rate yang menyatakan nilai usaha untuk setiap satu function point (Subriadi and Ningrum, 2014).

Dalam penelitian ini, nilai Velocity digunakan untuk mendapatkan nilai PDR. Oleh karena PDR memiliki satuan fp/hour sedangkan Velocity merupakan rata-rata nilai story point yang diselesaikan dalam satu kali iterasi, maka peneliti mengajukan tiga langkah persamaan untuk mengkonversi nilai velocity menjadi nilai PDR.

Langkah pertama, mendapatkan nilai  $D_{VEL}$  atau nilai *Story Point per harinya* dengan persamaan sebagai berikut:

$$VEL_d = \frac{Velocity}{i} .....(4)$$

Dimana:

$VEL_d$  : Story Point / hari

Velocity: rata-rata story point

i : total hari dalam sekali iterasi

Langkah kedua, mendapatkan nilai story point per jamnya. Sesuai dengan peraturan dalam undang-undang Depnaker no. 13 tahun 2003 pasal 77 ayat 1 yang mengatur jam kerja dengan ketentuan 40 jam per minggu atau 8 jam per hari untuk 5 hari kerja. Oleh karena itu, persamaan dirumuskan sebagai berikut:

$$VEL_h = \frac{VEL_d}{8} \dots\dots\dots(5)$$

Dimana :

- VEL<sub>h</sub> : Story Point / jam
- VEL<sub>d</sub> : Story Point / hari

Setelah mendapatkan nilai Story Point per jam, tahap terakhir adalah mendapatkan nilai Project Delivery Rate menggunakan persamaan sebagai berikut:

$$X_H = \frac{X_S}{VEL_H} \dots\dots\dots(6.1)$$

$$PDR = \frac{X_H}{X_{FP}} \dots\dots\dots(6.2)$$

Dimana :

- PDR : Project Delivery Rate (fp/jam)
- X<sub>H</sub> : Total estimasi penyelesaian fitur x (jam)
- X<sub>FP</sub> : Function point pada fitur x (fp)
- X<sub>H</sub> : Total estimasi penyelesaian fitur x (jam)
- X<sub>S</sub> : Story point pada fitur x (SP)
- VEL<sub>H</sub> : Story Point / jam

**2.1.5.2. Menentukan distribusi effort per aktivitas**

Tahap ini bertujuan untuk mengidentifikasi aktivitas-aktivitas yang telah ditetapkan oleh manajer proyek. Daftar aktivitas tersebut digunakan untuk memperoleh distribusi effort pada setiap aktivitasnya. Beberapa penelitian telah melakukan penelitian tentang presentasi effort per aktivitas berdasarkan kapasitas

perusahaan tempat pengembangan proyek perangkat lunak. Tabel 2.25. menyajikan distribusi *Effort* pada setiap aktivitas pengembangan perangkat lunak.

Tabel 2.25. Distribusi Effort per Aktivitas

No.	Activity	Effort distribution per project size (%)	
		Medium-large (Saleh, 2011)	Small (Primandari A and Sholih, 2015)
Software development phase			
1	Requirements	7.5%	1.17%
2	Specifications	7.5%	6.75%
3	Design	10%	5.57%
4	Implementation	10%	55.65%
5	Integration Testing	7.5%	6.42%
6	Acceptance & deployment	7.5%	5.6%
Ongoing activities & quality and testing			
7	Project management	8.34%	2.55%
8	Configuration management	4.16%	3.58%
9	Quality assurance	8.34%	0.66%
10	Documentation	4.16%	9.76%
11	Training & support	4.16%	0.6%
12	Evaluation & testing	20.84%	1.67%

Sumber : Saleh, 2011; Primandari & Sholih, 2015

Untuk memperoleh effort per aktivitas, persamaan yang digunakan mengacu pada Putri dan Subriadi (2018) adalah sebagai berikut:

$$Effort\ per\ Activity = Effort\ Distribution(\%) * Effort \dots\dots\dots(7)$$

### 2.1.5.3. Menentukan pay-rate per aktivitas

Pay-rate (pay/jam) dapat ditentukan melalui gaji personil yang terlibat pada proyek pengembangan perangkat lunak. nilai pay-rate (pay/jam) berbeda tiap personil bergantung pada posisi personil dalam team seperti manajer proyek, analis sistem, programmer, dokumenter, tim penguji dan lainnya. Tahap memperoleh pay-rate di menggunakan rumus yang digunakan oleh Putri dan Subriadi (2018) sebagai berikut:

$$PWR = \frac{PMR}{4.1} \dots\dots\dots(8)$$

$$PDR = \frac{PWR}{22} * 1.1 \dots\dots\dots(9)$$

$$PHR = \frac{PDR}{8} * 1.3 \dots\dots\dots(10)$$

Dimana:

PWR : Gaji per-minggu

PMR : Gaji per-bulan

PDR : Gaji per-hari

PHR : Gaji per-jam

Tujuan dari tahapan ini adalah mendapatkan gaji per-jam dari masing masing individu yang terlibat pada proses pengembangan perangkat lunak. Apabila individu memiliki standar gaji/bulan maka dapat dikonversi menjadi gaji/jam menggunakan persamaan (9) dan (10).

### 2.1.5.4. Menentukan biaya per aktivitas

Biaya per aktivitas perangkat lunak dapat dihitung menggunakan persamaan yang digunakan oleh Putri dan Subriadi (2018) sebagai berikut :

$$Cost\ in\ activity_i = Effort\ x\ percentage_i\ x\ pay\ Rate_i \dots\dots\dots(11)$$

Dimana :

cost in activity<sub>*i*</sub> : Biaya yang dibutuhkan aktivitas *i* dalam pengembangan proyek perangkat lunak

Effort : Usaha yang dibutuhkan untuk menyelesaikan proyek perangkat lunak

Percentage<sub>*i*</sub> : Persentase distribusi effort pada aktivitas *i*

Pay Rate<sub>*i*</sub> : Merupakan pay-rate untuk aktivitas *i*

Akumulasi dari nilai biaya per aktivitas yang didapatkan merupakan hasil akhir atau keluaran dari proses estimasi biaya perangkat lunak.

## 2.2. Kajian peneliti terdahulu

Subbab ini menjabarkan penelitian terdahulu yang berkaitan dengan penelitian yang akan dilakukan. Penelitian yang akan dibahas adalah kajian teori yang telah dilakukan peneliti sebelumnya sehingga bisa ditemukan celah yang nantinya akan diteliti lebih lanjut dan diharapkan dapat dilakukan penggalan lebih mendalam dari penelitian-penelitian yang disesuaikan dengan kebutuhan penelitian ini.

### 1. Limitations of Function Point Analysis in Multimedia Software/Application Estimation – (Kumar et al., 2019)

Penelitian yang dilakukan Kumar et al. ini bertujuan untuk menjelaskan batasan-batasan yang dapat dilakukan oleh metode FPA terhadap jenis aplikasi multimedia. Hasilnya, secara keseluruhan metode FPA hanya mampu melingkupi are fungsional dari sistem dan mengabaikan area non-fungsional seperti animasi, keamanan, dan User Interface yang merupakan bagian penting dari aplikasi multimedia.

### 2. Software Cost Estimation Using Function Point Analysis – (Putri and Subriadi, 2019)

Penelitian yang dilakukan oleh Purti dan Subriadi bertujuan untuk memodifikasi metode Function Point dengan menambahkan indikator kompleksitas bisnis kedalam General System Characteristics (GSC).

Modifikasi pada metode Function Point tersebut digunakan untuk melakukan estimasi biaya perangkat lunak pada 4 aplikasi layanan publik. Hasil estimasi dari metode function point sebelum ataupun sesudah modifikasi kemudian dibandingkan dengan harga sebenarnya. Hasil penelitian yang didapatkan adalah metode FPA tanpa modifikasi memiliki deviasi 5.54%, sedangkan FPA yang dimodifikasi memiliki deviasi yang lebih kecil yaitu 4.05%.

**3. Function Points Method in Game Casual Context – (Dewi et al., 2018)**

Penelitian yang dilakukan oleh Dewi et al. bertujuan untuk menilai keakuratan metode FPA dalam industri game, pengujian dilakukan dengan melakukan estimasi effort pada 8 casual game. Hasil dari penelitian yang didapat adalah effort yang dihitung menggunakan metode function point memiliki deviasi yang besar yaitu 376 % dari effort sesungguhnya.

**4. A Size Estimation Model for Board-Based Desktop Games – (Sabahat et al., 2017)**

Penelitian yang dilakukan Sabahat et al. bertujuan untuk menciptakan sebuah estimasi ukuran permainan digital bertipe board-game berdasarkan proyek terdahulu. Dalam penelitiannya, Sabahat et al. menggunakan 65 permainan digital berbasis board-game dan menggunakan multiple regresi linier untuk memprediksi ukuran pada sebuah permainan digital. Hasilnya, penelitian ini menghasilkan metode pengukuran size permainan digital menggunakan regresi linier dengan inputan parameter berupa banyaknya aturan permainan (*rules*), opsi pilihan lainnya (*miscellaneous game options*), jumlah pemain dan animasi.

**5. Understanding and Improving Effort Estimation in Agile Software Development- An Industrial Case Study – (Tanveer et al., 2016)**

Tanveer et al. melakukan penelitian yang bertujuan untuk meningkatkan efektivitas dari hasil estimasi effort pada perangkat

lunak yang menerapkan agile development. Berdasarkan hasil dari dua observasi dan 11 interview pada tiga tim agile development dari perusahaan multinasional di Jerman, penelitian ini menyimpulkan bahwa faktor-faktor seperti pengetahuan dan pengalaman pengembang dan kompleksitas dan dampak perubahan pada sistem yang mendasarinya mempengaruhi besarnya serta keakuratan estimasi.

**6. Effort, Duration and Cost Estimation in Agile Software Development – (Owais and Ramakishore, 2016)**

Penelitian yang dilakukan oleh Owais dan Ramakishore bertujuan untuk merumuskan metode untuk mengestimasi biaya dari perangkat lunak pada perusahaan IT di india. Dalam penelitian ini, Owais dan Ramakishore menggunakan pendekatan perhitungan Story Point. Hasilnya menunjukkan bahwa nilai durasi dan biaya proyek akan meningkat jika effort semakin tinggi. Namun jika nilai effort tinggi dan dilakukan penambahan sumber daya maka durasi dan biaya hampir tetap sama. Penelitian ini juga menggarisbawahi faktor-faktor seperti Komunikasi dalam tim di zona waktu berbeda yang berbicara bahasa yang berbeda, keakraban dalam tim juga meningkatkan kualitas hasil yang didapatkan.

**7. A Review of Software Cost Estimation in Agile Software Development Using Soft Computing Techniques – (Bilgaiyan et al., 2016)**

Penelitian yang dilakukan oleh Bilgaiyan et al. bertujuan untuk melakukan literatur review model estimasi biaya pada perangkat lunak yang dikembangkan menggunakan agile methodology. Hasilnya dua metode estimasi biaya terbaik dalam mengukur biaya adalah menggunakan metode Bayesian Network dengan akurasi 62.8% dan metode Planning Poker (Story point) dengan akurasi 57%.

**8. Mobile game size estimation: COSMIC FSM rules, UML mapping model and Unity3D game engine – (Abdullah et al., 2014)**

Penelitian ini dilakukan oleh Abdullah et al. bertujuan untuk mengestimasi besaran ukuran permainan digital dengan menggunakan

metode COSMIC FSM. Target permainan digital yang digunakan dalam penelitian ini merupakan mobile game dan dibangun menggunakan Unity 3D engine. Hasilnya, ukuran perangkat lunak permainan digital didapatkan hanya berdasarkan data movement saja yang berarti hanya mempertimbangkan unsur fungsional saja.

#### **9. A Case Study in COSMIC Functional Size Measurement: Angry Bird Mobile Application – (Abdullah et al., 2013)**

Penelitian yang dilakukan Abdullah et al. merumuskan penggunaan UML dengan COSMIC FSM untuk studi kasus di game mobile. Penelitian ini memberikan opsi kepada pengembang perangkat lunak untuk memperkirakan ukuran aplikasi game mobile dengan menggunakan metode FSM COSMIC berbasis UML. Penelitian ini berkontribusi pada pembuatan model UML sesuai dengan aturan pengukuran COSMIC sehingga memudahkan prosedur pengukuran. Perlu digarisbawahi bahwa dalam COSMIC yang merupakan turunan dari FPA, perangkat lunak diukur berdasarkan pergerakan data (*data movement*) atau dari unsur fungsional sistem.

Penelitian terdahulu mengindikasikan bahwa metode Function point membutuhkan sebuah modifikasi dalam menghadapi perubahan jaman. Secara spesifik, metode FPA dan metode turunannya (COSMIC) belum secara sempurna menghitung besarnya faktor non-fungsional, padahal faktor non-fungsional dapat mempengaruhi besaran sistem hingga 85 persen (Verner et al., 1989). Kurangnya kemampuan FPA dalam non-fungsional sistem menyebabkan ketidaksesuaian dalam menghasilkan ukuran perangkat lunak permainan digital – yang sebagian besar merupakan non-fungsional – dan mengakibatkan estimasi *effort* yang berbeda dengan *effort* aslinya sehingga berujung pada nilai estimasi biaya yang tidak sesuai dengan usaha asli yang dikeluarkan oleh tim pengembang. Selain itu, dapat disimpulkan dari penelitian terdahulu bahwa belum ada metode estimasi biaya yang secara spesifik ditujukan untuk permainan digital.

Peneliti bertujuan untuk melibatkan faktor non-fungsional dalam perhitungan untuk mendapatkan besaran *Effort* dengan menggunakan metode *Story*

*Point.* Metode *Story Point* akan diarahkan untuk mendapatkan nilai PDR sehingga akan mengkalibrasi nilai dari effort dengan mempertimbangkan kompleksitas non-fungsional proyek perangkat lunak permainan digital.

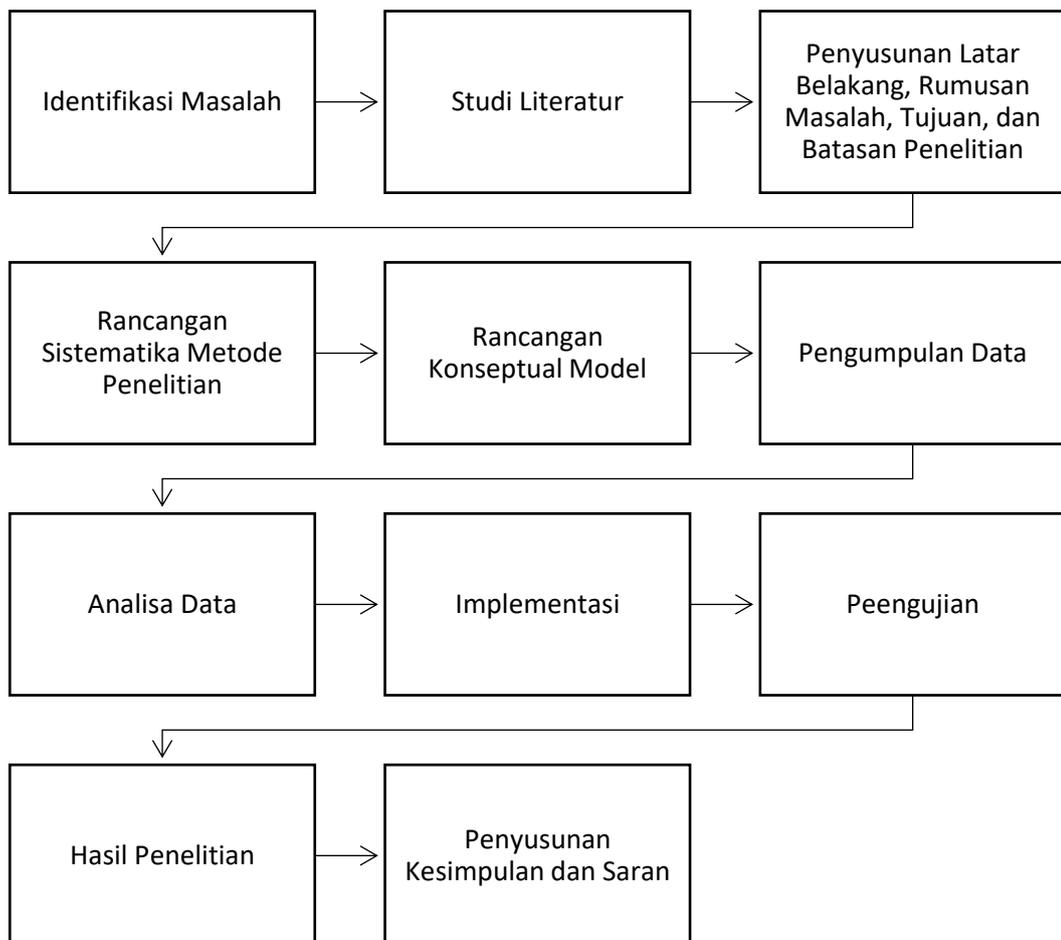
## BAB 3

### METODOLOGI PENELITIAN

Bab ini menyajikan tahapan penelitian yang dilakukan dari awal hingga penelitian selesai dan dijadikan sebagai acuan dalam proses pengerjaan tesis, sehingga rangkaian pengerjaan dapat dilakukan secara terarah, teratur, dan sistematis.

#### 9.1. Tahapan penelitian

Berikut ini merupakan tahapan yang akan dilakukan dalam penelitian ini seperti pada Gambar 3.1.



Gambar 3.1. Tahapan penelitian

### **3.1.1. Identifikasi topik penelitian**

Topik pada penelitian ini didapatkan dari identifikasi isu yang terjadi dan studi literatur terdahulu. Peneliti menemukan fenomena semakin dibutuhkannya estimasi biaya perangkat lunak di industri permainan digital karena popularitas yang semakin meningkat, namun belum ada metode yang dapat melakukan estimasi biaya pada perangkat lunak terutama pada perangkat lunak permainan digital yang akurat.

### **3.1.2. Studi literatur**

Studi literature dilakukan dengan mengumpulkan data, informasi dan teori-teori yang mendukung penelitian. Selain itu juga mengulas penelitian terkait dan metode yang dijadikan acuan dalam penelitian. Penjelasan mendetail dari studi literature yang digunakan pada penelitian ini dijabarkan dalam bab dua.

Pada tahap ini peneliti menemukan bahwa metode Function Point merupakan metode standar dan paling banyak digunakan untuk menghitung estimasi effort yang kemudian dapat digunakan untuk mengestimasi biaya. Namun kelemahan pada metode ini yaitu tidak dapat menghitung kebutuhan non-fungsional pada permainan digital, yang justru merupakan depedensi terbesar menyebabkan perlunya solusi pendukung untuk menyelesaikan masalah tersebut. Metode Story point dinilai peneliti memiliki kualitas untuk membantu menghitung faktor non-fungsional tersebut karena dalam metode Story Point tidak melihat dari fungsional sistem, tetapi melihat secara keseluruhan sistem dengan sudut pandang tim pengembang.

### **3.1.3. Penyusunan latar belakang, permasalahan, tujuan, dan batasan penelitian**

Latar belakang disusun setelah penulis mengidentifikasi fenomena dalam pengembangan perangkat lunak dan berdasarkan penelitian sebelumnya. Rumusan masalah didapatkan berdasarkan permasalahan menarik dari fenomena yang muncul. Tujuan penelitian ditetapkan untuk menentukan arah dari penelitian ini. Kontribusi penelitian ditetapkan dengan tujuan pengembangan ilmu pengetahuan bagi masyarakat dan bisnis dan dapat digunakan dalam penelitian selanjutnya.

Batasan penelitian agar penelitian lebih terfokus dan sesuai dengan kebutuhan. Pembahasan dari keempat poin ini pada bab satu.

#### **3.1.4. Rancangan sistematika metode penelitian**

Pengerjaan thesis ini memiliki sistematika tahapan yaitu sebagai berikut:

1. Mendapatkan requirement yang digunakan untuk mengembangkan sistem permainan digital berupa GDD (*game design document*) yang berisi kurang lebih seperti aktivitas pada tahapan pre-produksi pada Tabel 2.1.
2. Requirement yang berhasil dikumpulkan kemudian dihitung nilai Function Point berdasarkan penilaian data functional types dan data transactional types (Tabel 2.7). sehingga didapat nilai keseluruhan FP dari proyek permainan digital.
3. Selanjutnya dalam sebuah grup diskusi, tim pengembang melakukan pembagian fitur menjadi task terkecil dan memberi nilai Story Point pada setiap *task* seperti yang dicontohkan pada Tabel 2.8. Apabila terdapat permasalahan perbedaan nilai dalam satu team pada sebuah task, maka dilakukan penyelesaian seperti yang dijelaskan pada subbab Kalibrasi skala story point.
4. tim pengembang melakukan penjadwalan pengerjaan masing-masing task, sehingga didapatkan team velocity seperti pada Tabel 2.9.
5. Selanjutnya adalah melakukan konversi nilai Velocity menjadi PDR dengan menggunakan rumus (4), (5), dan (6). Setelah didapatkan nilai PDR, nilai effort dapat dihitung dengan nilai FP dan nilai PDR menggunakan rumus (3). Nilai effort tersebut selanjutnya akan digunakan untuk menghitung estimasi biaya menggunakan rumus (7), (8), (9), (10) dan (11).
6. Tahap terakhir adalah membandingkan hasil estimasi biaya setiap proyek permainan digital menggunakan metode gabungan FPA dan metode story point dengan biaya sesungguhnya dengan menggunakan nilai selisih antara estimasi biaya dengan nilai biaya sesungguhnya

untuk mengetahui akurasi dari metode gabungan ini dalam upaya estimasi biaya perangkat lunak permainan digital.

### 3.1.5. Rancangan konseptual model

Pada tahap perancangan konseptual model, peneliti menyusun kerangka berpikir dengan memodelkan modifikasi metode function point dengan metode Story Point. Metode function point fokus pada pengukuran fungsional perangkat lunak, namun perhitungan tersebut belum mewakili non-fungsional yang terdapat pada perangkat lunak. Oleh karena peneliti tidak dapat merubah dan memodifikasi nilai FP secara langsung, maka solusi peneliti adalah memodifikasi PDR yang menggambarkan kecepatan team dalam menyelesaikan task dengan menggunakan metode Story Point. Perancangan konseptual model dituangkan dalam bentuk bagan konseptual model yang dijabarkan pada bab empat.

### 3.1.6. Pengumpulan data

Pengumpulan data dilakukan di sebuah game studio dan spesifikasi data merupakan projek permainan digital berbasis mobile. Untuk penjelasan lebih spesifik akan disajikan pada sub-bab profil perusahaan dan sub-bab spesifikasi data yang digunakan.

#### 3.1.6.1. Profil perusahaan

Perusahaan yang akan digunakan pada penelitian ini adalah sebuah game studio indie bernama Re: tale. Lebih lanjut Detail dari perusahaan ini disajikan pada Tabel 3.1.

Tabel 3.1. Detail Re: tale game studio

<b>Nama Perusahaan</b>	Re: tale Game Studios
<b>Motto</b>	<ul style="list-style-type: none"><li>- Bring Story to the Game</li><li>- Work Fast and High Spirit</li><li>- Make Award Quality Game</li></ul>
<b>CEO</b>	M. Syaqiq Azzahid

<b>Member</b>	
<b>Game designer and programmer</b>	Abudzar Ali Ibnu Raziq R. Surahutomo
<b>Artist (2d)</b>	M. Syaqq Azzahid Dimmy Maulana
<b>Artist (3d)</b>	M. Fajar Hillman
<b>Website</b>	<a href="https://retale.github.io/">https://retale.github.io/</a>

Re: tale game studios didirikan pada November 2016 oleh M. Syaqq Azzahid bersama 4 orang lain dan bertujuan untuk membuat permainan digital yang berfokus pada story dan gameplay yang *enjoyable*. Mayoritas permainan digital yang dibangun berfokus dalam pengembangan permainan digital berbasis mobile dan pada platform Android maupun IOS. Kedepannya, Re: tale juga akan menarget membangun permainan digital berbasis PC dengan medium STEAM.

### **3.1.6.2. Spesifikasi data yang digunakan**

Data yang akan dikumpulkan dan digunakan pada penelitian ini meliputi *Game Design Document* (GDD) termasuk jenis perangkat lunak permainan digital (mobile, PC, atau konsol), aplikasi permainan digital yang dapat beroperasi, gaji team yang terlibat langsung dalam pengembangan permainan digital, durasi pengerjaan proyek, dan biaya sesungguhnya pengerjaan proyek.

### **3.1.7. Analisis data**

Data yang didapat dari tahap sebelumnya dianalisis dan diproses berdasarkan pada jenis datanya. Data-data tersebut kemudian diklasifikasi menjadi dua jenis data. Pertama, data permainan digital yang akan digunakan dalam perhitungan FPA dan SP. Data tersebut berdasarkan pada dokumen spesifikasi kebutuhan perangkat lunak (SRS atau GDD), apabila tidak tersedia maka peneliti akan menyusun SRS/GDD dengan menganalisa fitur-fitur dari aplikasi yang telah jadi dan menjabarkannya sesuai dengan template pada Tabel 2.1 tahapan pre-produksi. Kedua, data nilai proyek berdasarkan dari harga asli proyek yang

diberikan dari project owner atau apabila tidak tersedia maka data nilai proyek akan didapatkan berdasarkan nilai gaji seluruh tim dikalikan dengan durasi pengerjaan proyek.

### **3.1.8. Implementasi**

Tahap implementasi adalah proses implementasi dari sistematika metode penelitian yang telah dirancang pada penelitian ini. Lebih spesifik, data kebutuhan proyek atau GDD digunakan untuk mendapatkan nilai FP, selain itu GDD juga digunakan untuk mendapatkan nilai Velocity team pengembang. Data gaji digunakan untuk proses menghitung estimasi biaya, proses estimasi biaya dilakukan dengan menggunakan persamaan (8), (9), (10) dan (11). Terakhir, data gaji dan durasi digunakan untuk mendapatkan harga asli proyek. Harga asli proyek tersebut digunakan sebagai pengukur keakuratan dari dua perhitungan yaitu menggunakan metode FPA dan gabungan metode FPA-MSP.

### **3.1.9. Pengujian**

Pengujian dilakukan dengan menguji validitas yaitu membandingkan estimasi biaya yang dihitung berdasarkan metode FPA dan FPA-MSP dengan harga perangkat lunak permainan digital yang sebenarnya dengan menggunakan nilai selisih dari biaya asli dengan biaya estimasi.

### **3.1.10. Hasil dan pembahasan penelitian**

Tahap penyusunan hasil dan pembahasan menyajikan hasil dari implementasi dan pengujian, seberapa akurat metode gabungan FPA-MSP dengan harga asli permainan digital.

### **3.1.11. Penyusunan kesimpulan dan saran**

Tahap akhir dalam penelitian ini yaitu menganalisis dan membahas temuan keseluruhan dalam penelitian, terkait dengan hasil implementasi dan pengujian yang diperoleh. Tahap penyusunan kesimpulan dilakukan dengan menelaah secara keseluruhan terhadap apa yang telah dilakukan pada penelitian ini. Kesimpulan dibuat berdasarkan hasil studi literatur, sistematika metode penelitian,

implementasi, pengujian dan penyusunan hasil yang diperoleh. Selain itu pada tahapan ini, peneliti memberikan saran untuk peluang penelitian yang akan datang.

[HALAMAN INI SENGAJA DIKOSONGKAN]

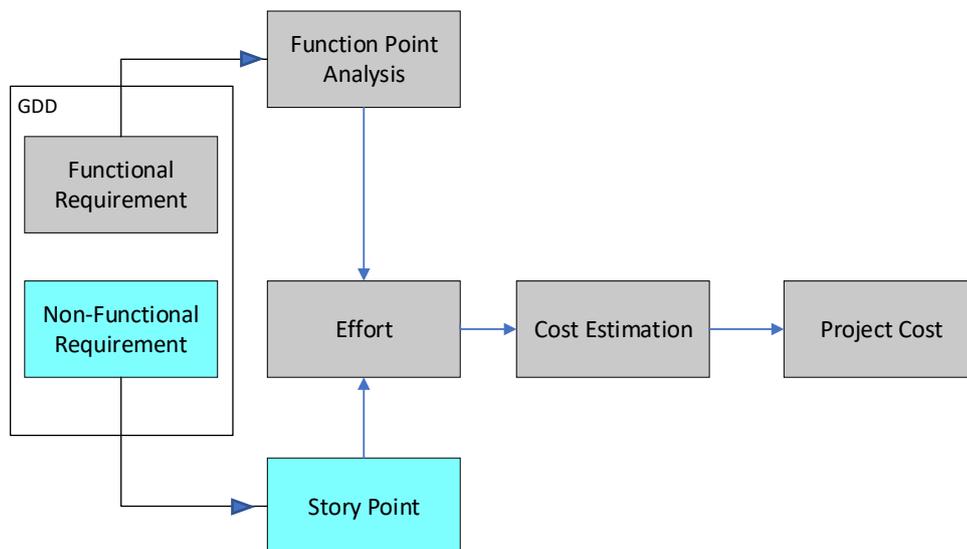
## BAB 4

### MODEL KONSEPTUAL

Bab ini menjelaskan model konseptual perhitungan estimasi biaya dengan menggunakan gabungan metode Function Point dengan Story Point. Function Point digunakan untuk menghitung fungsional dari sistem, kemudian metode Story Point digunakan untuk melengkapi kelemahan dari metode Function Point yaitu kelemahan dalam perhitungan non-fungsional. Oleh karena tidak ditemukan cara untuk mempengaruhi nilai functional poin secara langsung, maka story point akan diarahkan untuk memodifikasi nilai PDR yang merupakan value untuk menilai tingkat produktivitas team.

#### 4.1. Skema model Konseptual

Model konseptual pada penelitian ini berawal dari kerangka pikir yang disajikan pada Gambar 4.1.



Gambar 4.1. Model konseptual

Nilai function point didapatkan melalui perhitungan nilai FP (Function Point) berdasarkan hasil pembobotan lima komponen fungsional dari sistem yaitu EI, EO, EQ, ILF, dan EIF. Pada penelitian ini nilai FP tersebut berperan sebagai nilai kebutuhan fungsional dari sebuah sistem permainan digital. Untuk kebutuhan

non fungsional seperti kompleksitas pembuatan gambar, komposisi dan desain level, dan keterlibatan AI, Peneliti akan menggunakan metode penilaian story point (MSP). Namun metode penilaian story point tidak bisa secara langsung mengubah besaran dari nilai function point. Tidak adanya petunjuk untuk konversi nilai SP ke FP membuat peneliti merumuskan persamaan (4), (5), dan (6) untuk mengubah nilai tingkat produktivitas team (velocity) pada metode Story Point ke nilai PDR yang memiliki satuan fp/jam. Nilai PDR yang merupakan hasil dari metode Story Point akan digunakan melalui persamaan (3) untuk mengkalibrasi nilai effort yang didapatkan menggunakan metode FPA. Selanjutnya, nilai effort digunakan untuk mendapatkan nilai estimasi biaya pada proyek permainan digital.

Metode story point menggunakan pertimbangan tim pengembang sistem untuk menentukan kompleksitas masing-masing task pada permainan digital berbeda dengan metode function point yang hanya menggunakan input dan output spesifik dan telah memiliki nilai pembobotan khusus. Sebagai contoh, untuk membangun task login pada permainan digital, FPA hanya akan dipertimbangkan input yang akan dimasukkan oleh user karena tujuan metode function point berorientasi pada data. Namun pada kenyataannya dalam permainan digital, fitur tersebut juga dilengkapi oleh ilustrasi yang mungkin 2D bahkan 3D, sound background, beserta narasi-narasi dialog yang tidak mungkin dihitung menggunakan FPA. Oleh karena itu, apabila sebuah perangkat lunak multimedia termasuk permainan digital diestimasi ukurannya dengan FPA maka berpotensi tinggi memiliki tingkat akurasi yang rendah karena terdapat komponen-komponen non-fungsional yang diabaikan. MSP mengisi kelemahan tersebut dengan mengikutsertakan faktor-faktor non-fungsional tersebut dengan menggunakan sudut pandang pengembang dalam menentukan kompleksitas task login. Sebagai contoh, untuk menentukan nilai dari task login metode FPA akan memberikan nilai stagnan sesuai dengan jumlah data yang digunakan. Dalam penilaian menggunakan SP task tersebut dihitung tidak hanya pada kompleksitas data saja namun juga dari sisi teknis lain seperti desain interfacenya, BGM, dan pemilihan font.

## **BAB 5**

### **HASIL DAN PEMBAHASAN**

Bab ini menjelaskan tentang implementasi dari konseptual model yang dirancang pada Gambar 4.1 dan hasil yang didapatkan dari proses tersebut. Lebih rinci bab ini akan menjelaskan tentang cara mendapatkan kebutuhan permainan digital pada setiap proyek, realitas proyek permainan digital, perhitungan nilai function point dan perhitungan nilai story point berdasarkan daftar kebutuhan yang didapat pada masing-masing proyek, menghitung estimasi biaya proyek permainan digital dengan metode function point, menghitung estimasi biaya dengan menggunakan metode gabungan function point dan story point, dan terakhir membandingkan nilai estimasi dari hasil perhitungan metode function point dan metode gabungan function point dan story point dengan harga asli proyek.

#### **5.1. Analisa kebutuhan proyek permainan digital**

Sebuah perangkat lunak memiliki daftar kebutuhan yang berbeda satu dengan yang lain. Daftar kebutuhan merupakan variasi fitur atau fungsi yang akan ditanamkan ke dalam perangkat lunak. Semakin banyak fitur atau fungsi yang ada pada perangkat lunak akan semakin meningkatkan kompleksitas perangkat lunak. Kompleksitas perangkat lunak akan mempengaruhi besar usaha yang dilakukan tim pengembang dan akan berefek pada biaya dan waktu yang dibutuhkan untuk menyelesaikan perangkat lunak. Dalam permainan digital, daftar kebutuhan dapat disusun menggunakan sebuah *Game Design Document* (GDD). Yang memiliki komponen analisis sebagai berikut:

- *Overview* - Menjelaskan goal/ objektif dari permainan, termasuk konsep permainan secara umum, dan biaya proyek.
- *Mechanic* - Menjelaskan tentang mekanisme permainan termasuk karakter, dan daftar aset.
- *Dynamic* – Menjelaskan interaksi yang digunakan pada permainan termasuk antarmuka, level dan kecerdasan buatan.

- *Aesthetic* – Menjelaskan apa yang pemain dapat lihat, dengar, atau yang didapatkan dari panca inderanya (gambar, suara, animasi)
- *Experience* – Ekspektasi yang diharapkan dari pemain
- *Assuptions and constrain* – Batasan teknis yang mempengaruhi desain game.

Dengan menggunakan komponen analisis diatas, kebutuhan ketiga proyek permainan digital (P-MCP, PMCE, dan P-MVN) dapat didapatkan secara menyeluruh sehingga mempermudah untuk menghitung estimasi biaya masing-masing proyek. Berikut ini peneliti akan menjelaskan cara mendapatkan daftar kebutuhan beserta tugas yang harus dilakukan untuk memenuhi kebutuhan tersebut.

### **5.1.1. Kebutuhan pada proyek P-MCP**

Sub bab ini akan menjelaskan proses mendapatkan daftar kebutuhan pada proyek P-MCP.

#### **5.1.1.1. Overview proyek**

Proyek P-MCP merupakan sebuah proyek permainan digital yang ditujukan pada platform mobile. Proyek P-MCP memiliki genre *Casual*, pengertian casual disini adalah sebuah permainan digital yang dapat dimainkan secara santai, mudah dan diilustrasikan secara menarik atau lucu. Contoh permainan digital bergenre casual antara lain *Golfinity*, *Stack*, *That Level Again 2*, dan *Clay Jam*. Proyek permainan digital P-MCP memiliki konsep atau tema tentang kereta api dimana tujuan permainannya adalah mengarahkan kereta api pada rute yang seharusnya. Kesesuaian rute diimplikasikan dengan tujuan akhir stasiun yang memiliki warna sesuai dengan warna kereta api seperti yang diilustrasikan pada Gambar 5.1. Tugas pemain adalah mengubah rute rel kereta supaya kereta api dapat menuju pada stasiun yang sesuai.



Gambar 5.1. Tujuan permainan P-MCP

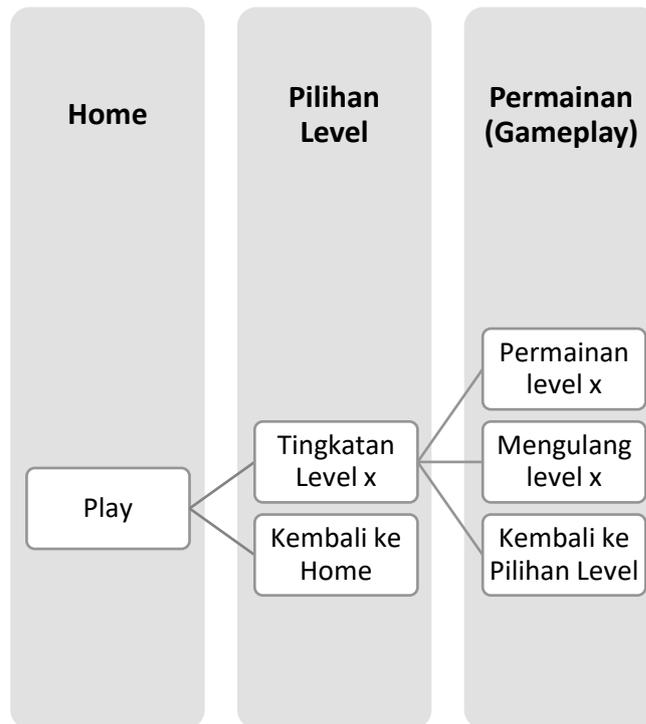
Berdasarkan penjabaran tersebut, peneliti merangkum poin-poin kunci proyek P-MCP pada Tabel 5.1.

Tabel 5.1. Overview proyek P-MCP

<b>Nama Proyek</b>	P-MCP
<b>Genre</b>	Casual
<b>Tema</b>	Kereta Api
<b>Konsep permainan</b>	Permainan bertujuan untuk mengarahkan kereta api menuju stasiun yang sesuai dengan warna kereta api dengan cara mengubah arah rel kereta.

#### 5.1.1.2. *Mechanic* proyek

Mekanisme (*Mechanic*) pada proyek permainan digital merupakan penjelasan dimana sebuah perangkat lunak itu dimulai dan berakhir. Gambar 5.2 mengilustrasikan mekanisme dari proyek permainan digital P-MCP.



Gambar 5.2. Alur permainan P-MCP

Berdasarkan hasil analisa mekanisme dalam Proyek P-MCP, proyek ini memiliki tiga tampilan utama (*interface*) yaitu Home, Pilihan Level, dan Permainan (Gameplay).

### A. Interface Home

Interface Home adalah tampilan pertama kali saat aplikasi P-MCP dibuka. interface Home tidak memiliki mekanisme yang kompleks; mekanisme yang ada pada interface ini hanya bertujuan untuk menghubungkan interface Home dengan interface Pilihan Level. Penghubungan tersebut diimplementasikan dengan fungsi untuk menampilkan interface Pilihan Level seperti yang disajikan pada Tabel 5.2.

Tabel 5.2. Kebutuhan pada interface Home

No	Daftar kebutuhan (fitur)	Tugas
1	Berpindah ke Interface Pilihan Level	Membuat fungsi untuk berpindah ke Tampilan Pilihan Level

## B. Interface Pilihan Level

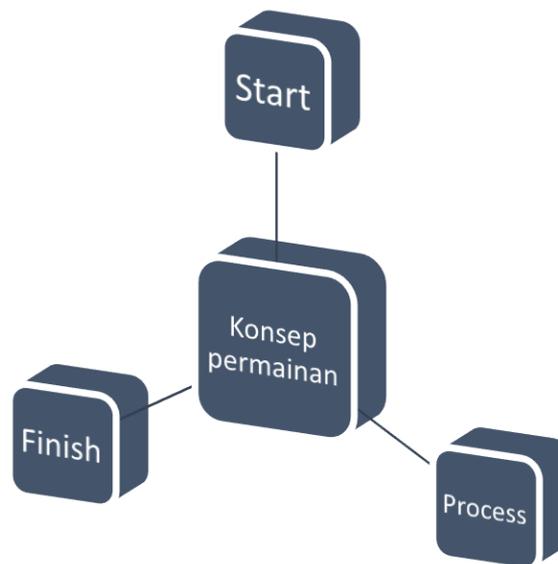
Interface Pilihan level merupakan interface yang menjadi perantara dari kedua Interface lainnya. Secara teknis interface ini memiliki fungsi utama yaitu (1) fungsi untuk kembali ke interface Home, dan (2) fungsi untuk menuju ke interface Gameplay. Daftar kebutuhan pada Interface Pilihan Level ditampilkan pada Tabel 5.3. Secara mekanisme teknis, opsi level biasa dan level endless diidentifikasi menjadi fungsi (2) karena sama-sama berfungsi untuk berpindah ke tampilan gameplay. Namun perlu digarisbawahi bahwa Proyek P-MCP memiliki mekanisme penguncian level, artinya bahwa pemain tidak bisa memainkan tingkatan level semaunya sehingga pemain diharuskan untuk menyelesaikan tingkatan level terendah sebelum memainkan tingkatan level yang lebih tinggi.

Tabel 5.3. Kebutuhan pada interface Pilihan Level

No	Daftar kebutuhan (fitur)	Tugas
1	Berpindah ke Interface Home	Membuat fungsi untuk berpindah ke interface Home
2	Berpindah ke Interface Gameplay sesuai pilihan level	Membuat fungsi untuk berpindah ke interface Gameplay
3	Pemain dapat memainkan level tertentu apabila level sebelumnya telah diselesaikan	Membuat fungsi untuk mengunci dan membuka level.

### C. Interface Gameplay

Interface Gameplay merupakan interface inti permainan digital P-MCP karena merupakan implementasi dari konsep permainan yang telah didefinisikan pada Tabel 5.1. Selanjutnya, konsep permainan tersebut diperjelas sehingga dapat dikembangkan pada game engine. Untuk tujuan tersebut, peneliti menggunakan pendekatan fase permainan seperti yang diilustrasikan pada Gambar 5.3.



Gambar 5.3. Pengembangan konsep permainan

Secara umum fase yang terdapat dalam permainan terdiri dari tiga yaitu:

- 1) *Start* merupakan bagaimana fase kondisi awal permainan
- 2) *Process* merupakan fase kondisi saat permainan berlangsung, dan
- 3) *Finish* merupakan fase kondisi pada saat permainan berakhir.

Dengan mengacu pada aplikasi P-MCP yang diberikan oleh tim pengembang, kemudian peneliti melakukan penjabaran mekanisme permainan berdasarkan ketiga fase tersebut seperti yang disajikan pada Tabel 5.4

Tabel 5.4. Pengembangan konsep berdasarkan tiga fase permainan

Konsep permainan	Fase	Detail mekanisme permainan
<p>Bertujuan untuk mengarahkan kereta api menuju stasiun yang sesuai dengan warna kereta api dengan cara mengubah arah rel kereta</p>	Start	<ul style="list-style-type: none"> <li>- Kereta api akan muncul pada titik tertentu dengan waktu tertentu.</li> <li>- Menampilkan durasi waktu permainan</li> <li>- Menampilkan nyawa yang dimiliki pemain.</li> </ul>
	Process	<ul style="list-style-type: none"> <li>- Kereta api akan berjalan dengan kecepatan tertentu.</li> <li>- Menjalankan hitung mundur waktu permainan.</li> <li>- User dapat merubah arah rel kereta pada titik tertentu.</li> <li>- Apabila pemain salah dalam mengarahkan kereta api sesuai dengan stasiunnya maka dihitung satu kesalahan dan mengurangi jumlah nyawa.</li> </ul>
	Finish	<ul style="list-style-type: none"> <li>- Pemain dinyatakan kalah apabila jumlah nyawa sama dengan 0.</li> <li>- Apabila nyawa pemain masih ada hingga waktu habis pemain dinyatakan menyelesaikan level.</li> <li>- Semakin banyak pemain melakukan kesalahan maka semakin kecil nilai yang</li> </ul>

Konsep permainan	Fase	Detail mekanisme permainan
		<p>didapat. Nilai tersebut diinterpretasikan sebagai bintang.</p> <ul style="list-style-type: none"> <li>- Apabila persyaratan bintang terpenuhi pemain dapat bermain ditingkatkan level selanjutnya</li> <li>- Pemain dapat mengulang permainan</li> <li>- Pemain dapat kembali ke interface Pilihan Level dengan melihat tampilan iklan terlebih dahulu.</li> </ul>

Setelah mendapatkan mekanisme permainan, kemudian peneliti merumuskan kebutuhan untuk mendukung mekanisme permainan supaya dapat berjalan. Tabel 5.5 menampilkan hasil analisa kebutuhan berdasarkan mekanisme permainan pada Interface Gameplay.

Tabel 5.5. Kebutuhan pada Interface Gameplay

No	Daftar kebutuhan	Tugas	Referensi Mekanisme permainan
1	Memunculkan (spawn) objek kereta pada posisi dan waktu tertentu.	Membuat fungsi untuk memunculkan kereta api.	Kereta api akan start pada titik tertentu.

No	Daftar kebutuhan	Tugas	Referensi Mekanisme permainan
2	Mendefinisikan dan menampilkan waktu permainan.	Mendefinisikan dan membuat fungsi untuk menampilkan dan kalkulasi waktu permainan.	<ul style="list-style-type: none"> <li>- Menampilkan durasi waktu permainan.</li> <li>- Menjalankan hitung mundur waktu permainan.</li> <li>- Apabila nyawa pemain masih ada hingga waktu habis pemain dinyatakan menyelesaikan level.</li> </ul>
3	Mendefinisikan dan menampilkan nyawa pemain.	Mendefinisikan dan membuat fungsi untuk menampilkan nyawa.	<ul style="list-style-type: none"> <li>- Menampilkan nyawa yang dimiliki pemain.</li> <li>- Apabila pemain salah dalam mengarahkan</li> </ul>
4	User mendapatkan pengurangan nyawa untuk setiap kereta yang masuk ke stasiun yang salah (tidak sesuai dengan warna).	Membuat fungsi untuk kalkulasi nyawa setiap kesalahan terjadi.	<ul style="list-style-type: none"> <li>kereta api sesuai dengan stasiunnya maka dihitung satu kesalahan dan mengurangi jumlah nyawa.</li> <li>- Pemain dinyatakan kalah apabila jumlah nyawa sama dengan 0.</li> </ul>

No	Daftar kebutuhan	Tugas	Referensi Mekanisme permainan
5	Menentukan kecepatan pergerakan kereta.	Membuat fungsi untuk menjalankan kereta berdasarkan kecepatan tertentu.	- Kereta api akan berjalan dengan kecepatan tertentu.
6	Merubah arah rel kereta berdasarkan input pemain.	Membuat fungsi untuk menerima input pemain yang berguna untuk merubah arah rel kereta.	- Pemain dapat merubah arah rel kereta pada titik tertentu.
7	Menampilkan jumlah bintang yang menandakan nilai dari pemain.	Membuat fungsi untuk mengakumulasi nilai yang didapat pemain berdasarkan jumlah kereta yang masuk ke stasiun yang benar.	Semakin banyak pemain melakukan kesalahan maka semakin kecil nilai yang didapat. Nilai tersebut diinterpretasikan sebagai bintang.
8	Memungkinkan pemain menuju level selanjutnya apabila	Membuat fungsi menuju tingkatan level selanjutnya	Apabila persyaratan bintang terpenuhi pemain dapat bermain ditingkatan level selanjutnya

No	Daftar kebutuhan	Tugas	Referensi Mekanisme permainan
	memenangkan permainan		
9	Mengulang permainan.	Membuat fungsi untuk mengulang permainan.	Pemain dapat mengulang permainan.
10	Berpindah ke Interface Pilihan Level dengan menampilkan iklan terlebih dahulu.	Membuat fungsi untuk menampilkan iklan yang kemudian berpindah ke interface Pilihan Level.	Pemain dapat kembali ke interface Pilihan Level dengan melihat tampilan iklan terlebih dahulu.

Secara keseluruhan, tahapan analisa mekanisme proyek P-MCP ini mendapatkan kebutuhan dan tugas seperti yang disajikan pada Tabel 5.6.

Tabel 5.6. Daftar kebutuhan proyek P-MCP

No	Daftar kebutuhan (fungsi)	Tugas
<b><i>Interface Home</i></b>		
1	Berpindah ke Interface Pilihan Level	Membuat fungsi untuk berpindah ke Tampilan Pilihan Level
<b><i>Interface Pilihan Level</i></b>		
2	Berpindah ke Interface Home	Membuat fungsi untuk berpindah ke interface Home
3	Berpindah ke Interface Gameplay sesuai pilihan level	Membuat fungsi untuk berpindah ke interface Gameplay

No	Daftar kebutuhan (fungsi)	Tugas
4	Pemain dapat memainkan level tertentu apabila level sebelumnya telah diselesaikan	Membuat fungsi untuk mengunci dan membuka level.
<b><i>Interface Gameplay</i></b>		
5	Memunculkan (spawn) objek kereta pada posisi dan waktu tertentu.	Membuat fungsi untuk memunculkan kereta api.
6	Mendefinisikan dan menampilkan waktu permainan.	Mendefinisikan dan membuat fungsi untuk menampilkan dan kalkulasi waktu permainan.
7	Mendefinisikan dan menampilkan nyawa pemain.	Mendefinisikan dan membuat fungsi untuk menampilkan dan kalkulasi nyawa.
8	Pemain mendapatkan pengurangan nyawa untuk setiap kereta yang masuk ke stasiun yang salah (tidak sesuai dengan warna).	Membuat fungsi untuk kalkulasi nyawa setiap kesalahan terjadi.
9	Menentukan kecepatan pergerakan kereta.	Membuat fungsi untuk menjalankan kereta berdasarkan kecepatan tertentu.
10	Merubah arah rel kereta berdasarkan input pemain.	Membuat fungsi untuk menerima input pemain yang berguna untuk merubah arah rel kereta.
11	Menampilkan jumlah bintang yang menandakan nilai dari pemain.	Membuat fungsi untuk mengakumulasi nilai yang didapat pemain berdasarkan jumlah kereta yang masuk ke

No	Daftar kebutuhan (fungsi)	Tugas
		stasiun yang benar dan menampikannya.
12	Memungkinkan pemain menuju level selanjutnya apabila memenangkan permainan	Membuat fungsi menuju tingkatan level selanjutnya
13	Mengulang permainan.	Membuat fungsi untuk mengulang permainan.
14	Berpindah ke Interface Pilihan Level dengan menampilkan iklan terlebih dahulu.	Membuat fungsi untuk menampilkan iklan yang kemudian berpindah ke interface Pilihan Level.

Berdasarkan Tabel 5.6 terlihat bahwa terdapat duplikasi kebutuhan pada Interface Home dan Interface Gameplay (no 1 dan 14). Hal ini kemudian menggarisbawahi kelemahan function point karena dalam function point kedua kebutuhan tersebut dianggap menjadi satu kebutuhan karena memiliki fungsi yang sama. Namun hal tersebut tidak selalu baik untuk jenis proyek bertipe multimedia khususnya permainan digital dikarenakan kedua fungsi tersebut dikemas dengan tujuan yang sedikit berbeda, kebutuhan no 1 akan berfungsi berpindah ke interface Pilihan Level sedangkan kebutuhan no 13 akan menampilkan iklan terlebih dahulu.

### 5.1.1.3. *Dynamic* proyek

Komponen dinamis merupakan sisi yang mempengaruhi *look and feel* dari permainan digital namun tidak merubah mekanisme permainan. Contoh untuk komponen dinamis adalah tampilan opsi grafis (biasanya terdapat settingan low-medium-high-ultra) yang mempengaruhi tampilan permainan namun secara mekanisme permainan tidak berubah. Selain itu, music sebagai pengiring

permainan digital pada beberapa jenis permainan merupakan contoh lain dari komponen dinamis. Namun tidak semua music merupakan komponen dinamis, seperti contoh permainan digital *Guitar Hero* yang memang berfokus terhadap music. Apabila komponen music dihilangkan maka permainan tersebut akan terganggu.

Berikut kami akan menjelaskan analisa sisi dinamis pada proyek P-MCP pada Interface Home, Pilihan Level, dan Gameplay.

### A. Interface Home

Dalam interface Home, peneliti mengidentifikasi empat kebutuhan dinamis pada permainan digital P-MCP yaitu fungsi untuk menghidupkan atau mematikan suara BGM, fungsi untuk menampilkan iklan, fungsi menghilangkan iklan dengan mengaktifkan pembelian menggunakan aplikasi pihak ketiga (eksternal) serta fungsi untuk menampilkan profil developer dengan membuka website developer pada sebuah browser atau memerintahkan aplikasi untuk membuka browser dengan website developer sebagai alamat url-nya. Berdasarkan olahan informasi tersebut, peneliti kemudian mengupdate kebutuhan pada interface Home seperti yang ditampilkan pada Tabel 5.7.

Tabel 5.7. Update kebutuhan interface Home (Dynamic)

No	Daftar kebutuhan (fitur)	Tugas
1	Berpindah ke Interface Pilihan Level	Membuat fungsi untuk berpindah ke Tampilan Pilihan Level
2	Pemain dapat mengaktifkan atau mematikan efek suara	Membuat fungsi untuk mengaktifkan atau mematikan efek suara
3	Menampilkan iklan pada aplikasi	Membuat fungsi untuk menampilkan iklan

No	Daftar kebutuhan (fitur)	Tugas
4	Menghilangkan iklan dengan mengaktifkan pembelian dalam aplikasi	Membuat fungsi untuk menghilangkan iklan
5	Menampilkan website developer	Membuat fungsi untuk membuka browser dengan url website developer.
<b>Keterangan</b>		
<p> <span style="display: inline-block; width: 15px; height: 15px; background-color: yellow; border: 1px solid black; margin-right: 5px;"></span> : Kebutuhan / tugas yang didapat pada tahapan Mechanic  <span style="display: inline-block; width: 15px; height: 15px; background-color: blue; border: 1px solid black; margin-right: 5px;"></span> : Kebutuhan / tugas yang didapat pada tahapan Dynamic </p>		

## B. Interface Pilihan Level

Interface Pilihan Level dirancang untuk tidak menampilkan seluruh level dalam satu penampilan. Hal tersebut menyebabkan aplikasi P-MCP memiliki mekanisme navigasi tingkatan level (diilustrasikan pada Gambar 5.4 yang dilingkari dengan warna biru) untuk berpindah dari satu daftar tingkatan level ke daftar tingkatan level lain.



Gambar 5.4. Elemen dynamic pada interface Pilihan Level

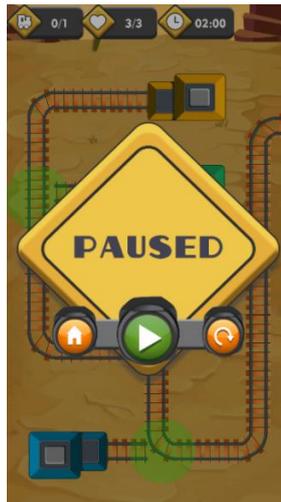
Oleh karena itu, peneliti kemudian mengupdate daftar kebutuhan pada interface Pilihan level seperti pada Tabel 5.8.

Tabel 5.8. Update kebutuhan interface Pilihan Level (Dynamic)

No	Daftar kebutuhan (fitur)	Tugas
1	Berpindah ke Interface Home	Membuat fungsi untuk berpindah ke interface Home
2	Berpindah ke Interface Gameplay sesuai pilihan level	Membuat fungsi untuk berpindah ke interface Gameplay
3	Navigasi daftar tingkatan level	Membuat fungsi untuk menampilkan daftar tingkatan level dengan range tertentu
4	Pemain dapat memainkan level tertentu apabila level sebelumnya telah diselesaikan	Membuat fungsi untuk mengunci dan membuka level.
<b>Keterangan</b>		
 : Kebutuhan / tugas yang didapat pada tahapan Mechanic  : Kebutuhan / tugas yang didapat pada tahapan Dynamic		

### C. Interface Gameplay

Peneliti mendapatkan dua elemen dinamis dalam interface Gameplay proyek P-MCP. Pertama, fitur untuk menghentikan sementara permainan (*pause*) dan melanjutkannya kembali (*resume*). Hal tersebut dikarenakan kemungkinan pemain untuk melakukan kegiatan lain ditengah proses permainan. Selain itu, fitur pause & resume juga membantu pemain untuk tidak kehilangan poin yang bukan akibat kesalahannya. Gambar 5.5 mengilustrasikan fitur pause yang ada pada interface Gameplay.



Gambar 5.5. Fitur Pause pada Interface Gameplay

Elemen dinamis yang kedua adalah kecerdasan buatan (AI) pada kereta. Kegunaan AI yang ditanamkan pada objek kereta memungkinkan kereta berjalan mengikuti arah rel kereta (*pathfinding*) dimana kereta akan menjadikan rel kereta sebagai *pointer* arah kereta bergerak. Gambar 5.6 mengilustrasikan pathfinding yang digunakan dalam interface Gameplay.



Gambar 5.6. Kondisi pathfinding kereta biru pada objek rel

Berdasarkan hasil indentifikasi tersebut daftar kebutuhan pada interface Gameplay hingga tahap dynamic adalah seperti pada Tabel 5.9.

Tabel 5.9. Update kebutuhan inferface Gameplay (Dynamic)

No	Daftar kebutuhan	Tugas
1	Memunculkan (spawn) objek kereta pada posisi dan waktu tertentu.	Membuat fungsi untuk memunculkan kereta api.
2	Mendefinisikan dan menampilkan waktu permainan.	Mendefinisikan dan membuat fungsi untuk menampilkan dan kalkulasi waktu permainan.
3	Mendefinisikan dan menampilkan nyawa pemain.	Mendefinisikan dan membuat fungsi untuk menampilkan nyawa.
4	Pemain mendapatkan pengurangan nyawa untuk setiap kereta yang masuk ke stasiun yang salah (tidak sesuai dengan warna).	Membuat fungsi untuk kalkulasi nyawa setiap kesalahan terjadi.
5	Menentukan kecepatan pergerakan kereta.	Membuat fungsi untuk menjalankan kereta berdasarkan kecepatan tertentu.
6	Merubah arah rel kereta berdasarkan input pemain.	Membuat fungsi untuk menerima input pemain yang berguna untuk merubah arah rel kereta.
7	Kereta dapat berjalan sesuai dengan alur rel kereta	Membuat fungsi pathfinding untuk objek kereta
8	Pemain dapat menghentikan permainan sementara dan melanjutkannya kembali	Membuat fungsi untuk menghentikan permainan sementara.
9	Menampilakan jumlah bintang yang menandakan nilai dari pemain.	Membuat fungsi untuk mengakumulasi nilai yang didapat pemain berdasarkan jumlah kereta yang masuk ke stasiun yang benar.

No	Daftar kebutuhan	Tugas
10	Mengulang permainan.	Membuat fungsi untuk mengulang permainan.
11	Berpindah ke Interface Pilihan Level dengan menampilkan iklan terlebih dahulu.	Membuat fungsi untuk menampilkan iklan yang kemudian berpindah ke interface Pilihan Level.
<b>Keterangan</b>		
 : Kebutuhan / tugas yang didapat pada tahapan Mechanic  : Kebutuhan / tugas yang didapat pada tahapan Dynamic		

Hasil analisa pada komponen dinamis yang dilakukan peneliti mengakibatkan kebutuhan proyek P-MCP menjadi bertambah. Tabel 5.10 menyajikan perubahan daftar kebutuhan yang telah didapatkan hingga tahapan Analisa komponen dinamis yang didapatkan pada ketiga interface yaitu Home, Pilihan Level, dan Permainan / Gameplay.

Tabel 5.10. Update daftar kebutuhan proyek P-MCP (Dynamic)

No	Daftar kebutuhan (fungsi)	Tugas
<i>Interface Home</i>		
1	Berpindah ke Interface Pilihan Level	Membuat fungsi untuk berpindah ke Tampilan Pilihan Level
2	Pemain dapat mengaktifkan atau mematikan efek suara	Membuat fungsi untuk mengaktifkan atau mematikan efek suara

No	Daftar kebutuhan (fungsi)	Tugas
3	Menampilkan iklan pada aplikasi	Membuat fungsi untuk menampilkan iklan
4	Menghilangkan iklan dengan mengaktifkan pembelian dalam aplikasi	Membuat fungsi untuk menghilangkan iklan
5	Menampilkan website developer	Membuat fungsi untuk membuka browser dengan url website developer.
<b><i>Interface Pilihan Level</i></b>		
6	Berpindah ke Interface Home	Membuat fungsi untuk berpindah ke interface Home
7	Berpindah ke Interface Gameplay sesuai pilihan level	Membuat fungsi untuk berpindah ke interface Gameplay
8	Navigasi daftar tingkatan level	Membuat fungsi untuk menampilkan daftar tingkatan level dengan range tertentu
9	Pemain dapat memainkan level tertentu apabila level sebelumnya telah diselesaikan	Membuat fungsi untuk mengunci dan membuka level.
<b><i>Interface Gameplay</i></b>		
10	Memunculkan (spawn) objek kereta pada posisi dan waktu tertentu.	Membuat fungsi untuk memunculkan kereta api.

No	Daftar kebutuhan (fungsi)	Tugas
11	Mendefinisikan dan menampilkan waktu permainan.	Mendefinisikan dan membuat fungsi untuk menampilkan dan kalkulasi waktu permainan.
12	Mendefinisikan dan menampilkan nyawa pemain.	Mendefinisikan dan membuat fungsi untuk menampilkan dan kalkulasi nyawa.
13	Pemain mendapatkan pengurangan nyawa untuk setiap kereta yang masuk ke stasiun yang salah (tidak sesuai dengan warna).	Membuat fungsi untuk kalkulasi nyawa setiap kesalahan terjadi.
14	Menentukan kecepatan pergerakan kereta.	Membuat fungsi untuk menjalankan kereta berdasarkan kecepatan tertentu.
15	Merubah arah rel kereta berdasarkan input pemain.	Membuat fungsi untuk menerima input pemain yang berguna untuk merubah arah rel kereta.
16	Kereta dapat berjalan sesuai dengan alur rel kereta	Membuat fungsi pathfinding untuk objek kereta
17	Pemain dapat menghentikan permainan sementara dan	Membuat fungsi untuk menghentikan permainan sementara.

No	Daftar kebutuhan (fungsi)	Tugas
	melanjutkannya kembali	
18	Menampilakan jumlah bintang yang menandakan nilai dari pemain.	Membuat fungsi untuk mengakumulasi nilai yang didapat pemain berdasarkan jumlah kereta yang masuk ke stasiun yang benar dan menampikannya.
19	Memungkinkan pemain menuju level selanjutnya apabila memenangkan permainan	Membuat fungsi menuju tingkatan level selanjutnya
20	Mengulang permainan.	Membuat fungsi untuk mengulang permainan.
21	Berpindah ke Interface Pilihan Level dengan menampilkan iklan terlebih dahulu.	Membuat fungsi untuk menampilkan iklan yang kemudian berpindah ke interface Pilihan Level.
<b>Keterangan</b>		
 : Kebutuhan / tugas yang didapat pada tahapan Mechanic  : Kebutuhan / tugas yang didapat pada tahapan Dynamic		

#### 5.1.1.4. Aesthetic proyek

Komponen Aesthetic berhubungan dengan apa yang dilihat dan didengar oleh pemain. Komponen ini menjadi kunci pada permainan digital karena aesthetic

terutama visualisasi merupakan elemen pertama yang berinteraksi dengan pemain. Selain itu, visualisasi juga menjadi faktor pembeda satu proyek permainan digital dengan permainan digital lainnya. Usaha yang dilakukan dalam membangun elemen-elemen aesthetic berbeda tergantung pada jenisnya. Sebagai contoh, untuk menciptakan elemen visual 3D yang realistis tentunya lebih memakan usaha yang lebih besar dibanding dengan visual 2D, untuk menciptakan BGM yang terdiri dari komposisi beragam alat musik juga tentunya akan lebih besar usahanya daripada BGM dengan satu alat musik saja.

Pada proyek P-MCP, komponen Aesthetic yang utamanya digunakan adalah visual. Komponen audio tidak digunakan karena Proyek P-MCP menggunakan *sound effect* dan *BGM* dari pihak ketiga sehingga peneliti memutuskan untuk mengabaikannya karena bukan merupakan ciptaan dari tim pengembang itu sendiri. Selanjutnya, peneliti akan melakukan analisa pada interface-interface proyek P-MCP untuk mendapatkan potensi kebutuhan baru berdasarkan analisis pada komponen aesthetic.

#### A. Interface Home

Interface Home pada proyek merupakan wajah dari sebuah permainan digital. Istilah wajah mengindikasikan bahwa interface tersebut adalah yang pertama kali terlihat saat pemain menjalankan permainan digital P-MCP. Interface Home P-MCP memiliki tampilan seperti pada Gambar 5.7.



Gambar 5.7. Elemen *Aesthetic* pada interface Home

Visual pada Interface Home harus memiliki elemen yang memperkenalkan identitas dan tema dari proyek P-MCP. Peneliti mengidentifikasi elemen visual pada interface Home proyek permainan digital P-MCP terdiri atas:

- 1) Ilustrasi background
- 2) Desain judul permainan digital
- 3) Desain tombol yang mewakili fungsi berpindah ke interface Pilihan Level
- 4) Desain tombol yang mewakili fungsi untuk enable / disable BGM
- 5) Desain tombol yang mewakili fungsi menampilkan website developer pada browser.
- 6) Desain tombol yang mewakili fungsi untuk menghilangkan iklan

Pentingnya visual pada Interface Home tersebut membuat daftar kebutuhan menjadi bertambah. Dalam interface Home, update daftar kebutuhan disajikan pada Tabel 5.11.

Tabel 5.11. Update daftar kebutuhan interface Home (Aesthetic)

No	Daftar kebutuhan (fitur)	Tugas
1	Berpindah ke Interface Pilihan Level	Membuat fungsi untuk berpindah ke Tampilan Pilihan Level
		Membuat desain tombol “play” yang mengindikasikan memulai permainan
2	Pemain dapat mengaktifkan atau mematikan efek suara	Membuat fungsi untuk mengaktifkan atau mematikan efek suara
		Membuat desain tombol efek suara “aktif” dan “nonaktif”

No	Daftar kebutuhan (fitur)	Tugas
3	Menampilkan iklan pada aplikasi	Membuat fungsi untuk menampilkan iklan
4	Menghilangkan iklan dengan mengaktifkan pembelian dalam aplikasi	Membuat fungsi untuk menghilangkan iklan
		Membuat desain tombol “no ads”
5	Menampilkan website developer	Membuat fungsi untuk membuka browser dengan url website developer.
		Membuat desain tombol Info
6	Memiliki visualisasi Interface Home yang menarik	Membuat desain background untuk interface Home
		Membuat desain logo untuk identitas aplikasi
		Merancang komposisi elemen-elemen tampilan visual.
<b>Keterangan</b>		
 <ul style="list-style-type: none"> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: yellow; border: 1px solid black; margin-right: 5px;"></span> : Kebutuhan / tugas yang didapat pada tahapan Mechanic</li> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: blue; border: 1px solid black; margin-right: 5px;"></span> : Kebutuhan / tugas yang didapat pada tahapan Dynamic</li> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: green; border: 1px solid black; margin-right: 5px;"></span> : Kebutuhan / tugas yang didapat pada tahapan Aesthetic</li> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: orange; border: 1px solid black; margin-right: 5px;"></span> : Kebutuhan yang memiliki tugas teknis program dan visual</li> </ul>		

Berdasarkan Tabel 5.11, dapat diketahui bahwa beberapa kebutuhan berkembang dengan memiliki tugas baru di sisi visual. Seperti contoh, kebutuhan “**Pemain dapat mengaktifkan atau mematikan efek suara**” yang semula diidentifikasi memiliki tugas teknis program (dynamic) saja (berwarna biru), namun pada tahapan Aesthetic diidentifikasi tugas baru yaitu menciptakan ilustrasi visual tombol *on / off*

suara yang merepresentasikan kondisi BGM dari permainan digital P-MCP sehingga kebutuhan tersebut menjadi memiliki dua tugas yang berkarakteristik berbeda yaitu teknis pemrograman dan desain visualisasi (berwarna orange).

## B. Interface Pilihan Level

Seperti halnya pada Interface Home, elemen visual yang digunakan pada interface Pilihan Level juga merupakan visualisasi dari fungsi-fungsi mekanis dan dinamis dan tampilan antarmuka yang memiliki komposisi menarik seperti yang ditampilkan pada Gambar 5.8.



Gambar 5.8. Elemen visual dalam interface Pilihan Level

Elemen visual pada interface Pilihan Level terdiri atas:

- 1) Background interface Pilihan Level
- 2) Visualisasi teks tulisan “*Level Selection*”
- 3) Visualisasi tombol fungsi lock/unlock tingkatan level
- 4) Visualisasi tombol fungsi navigasi daftar tingkatan level
- 5) Visualisasi tombol fungsi kembali ke Interface Home

Elemen - elemen visual tersebut beserta desain komposisi peletakan masing-masing elemen visual merupakan aspek penting dalam permainan digital. Oleh karena itu, dalam analisa komponen Aesthetic di

Interface Pilihan Level didapatkan kebutuhan baru seperti yang disajikan pada Tabel 5.12.

Tabel 5.12. Update kebutuhan Interface Pilihan Level (Aesthetic)

No	Daftar kebutuhan (fitur)	Tugas
1	Berpindah ke Interface Home	Membuat fungsi untuk berpindah ke interface Home
		Membuat desain tombol untuk fungsi kembali ke interface Home
2	Berpindah ke Interface Gameplay sesuai pilihan level	Membuat fungsi untuk berpindah ke interface Gameplay
3	Navigasi daftar tingkatan level	Membuat fungsi untuk menampilkan daftar tingkatan level dengan range tertentu
		Membuat desain tombol navigasi daftar tingkatan level
4	Pemain dapat memainkan level tertentu apabila level sebelumnya telah diselesaikan	Membuat fungsi untuk mengunci dan membuka level
		Membuat desain tombol untuk tingkatan level dengan kondisi terbuka dan terkunci
5	Memiliki visualisasi interface Pilihan Level yang menarik	Merancang konsep dan komposisi visual
		Mendesain background untuk interface Pilihan Level
		Mendesain label "Level Selection" pada interface Pilihan Level

## Keterangan

-  : Kebutuhan dan tugas yang didapat pada tahapan Mechanic
-  : Kebutuhan dan tugas yang didapat pada tahapan Dynamic
-  : Kebutuhan dan tugas yang didapat pada tahapan Aesthetic
-  : Kebutuhan yang memiliki tugas teknis program dan visual

### C. Interface Gameplay

Interface Gameplay merupakan interface yang memiliki porsi interaksi yang besar dengan pemain. Oleh karena itu, diperlukan desain visualisasi yang nyaman dan tidak membosankan di mata pemain. Visualisasi yang menarik dan unik sudah pasti menjadi bahan pertimbangan tim pengembang. Hasilnya, visualisasi yang digunakan pada permainan proyek P-MCP menggunakan sudut pandang atau prespektif *bird-eye* atau dari sudut pandang tampak atas seperti yang ditampilkan pada Gambar 5.9.



Gambar 5.9. Tampilan tingkatan level pada interface Gameplay

Penentuan sudut pandang tersebut mempengaruhi desain dari objek kereta dan objek-objek lainnya yang digunakan sebagai aset permainan. Terlebih, pada permainan digital P-MCP memiliki tingkatan 15 Level sehingga tim harus mendesain variasi susunan yang berbeda di setiap levelnya seperti perbandingan yang diilustrasikan pada Gambar 5.10. Hal tersebut cukup menjadi tantangan tersendiri.



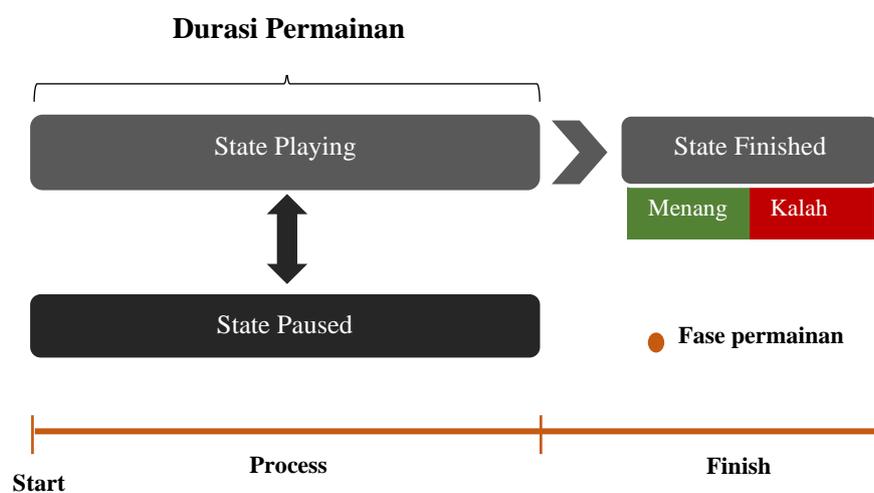
Gambar 5.10. Perbandingan konsep tingkatan level

Peneliti kemudian mengelompokkan Interface Gameplay kedalam tiga kondisi (*state*) tampilan Aesthetic yang berbeda yaitu:

1. ***State Playing*** atau kondisi tampilan saat permainan dimana user menjalankan permainan.
2. ***State Paused*** atau kondisi tampilan saat permainan berhenti dengan menekan tombol *pause*.
3. ***State Finished*** yang merupakan kondisi tampilan saat permainan berakhir. State ini akan dipanggil saat pemain memenuhi kondisi kemenangan atau kondisi kekalahan.

Tampilan pada state playing merupakan visualisasi dari fase awal (*start*) hingga proses (*prosess*) permainan. Dalam fase proses permainan, aplikasi permainan digital P-MCP memiliki kemampuan untuk menghentikan jalannya permainan (*pause*) sehingga tampilan *State*

*Playing* memiliki interaksi timbal balik dengan tampilan *State Paused* selama fase proses permainan berlangsung. Tampilan pada *State Finished* merupakan visualisasi dari fase selesainya permainan (*finished*). Fase *finished* sendiri merupakan hasil dari pencapaian pemain dalam fase proses permainan. Apabila pemain dapat memenuhi persyaratan memenangkan permainan maka *State Finished* akan menampilkan visual tampilan kemenangan dan begitu pula sebaliknya. Hubungan antara ketiga state tampilan diatas dengan fase permainan diilustrasikan pada Gambar 5.11.

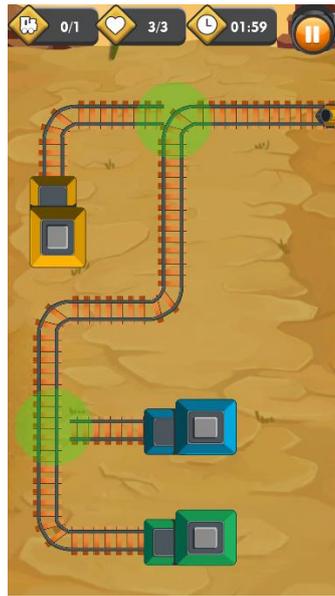


Gambar 5.11. Hubungan state dengan fase permainan

### C.1. State Playing

*State Playing* merupakan kondisi tampilan dimana pemain berinteraksi dengan permainan secara *real-time* dengan objek-objek visual seperti yang telah diilustrasikan pada Gambar 5.12. Peneliti mengidentifikasi elemen Aesthetic yang terdapat pada *state playing* terdiri atas:

- 1) Desain tombol pause untuk fungsi menghentikan permainan sementara
- 2) Desain indikator waktu
- 3) Desain indikator nyawa pemain

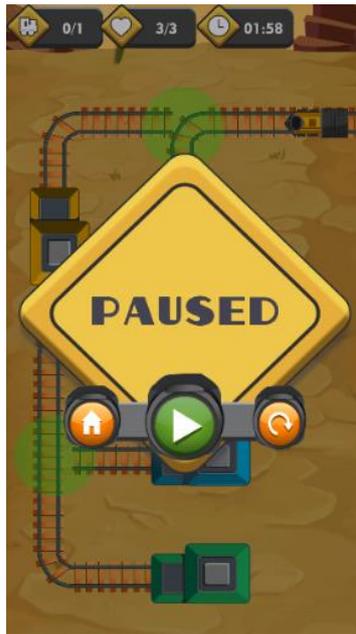


Gambar 5.12. Elemen visual interface Gameplay di *state playing*

- 4) Desain indikator jumlah kereta
- 5) Desain background
- 6) Desain objek stasiun kereta api, 14 jenis warna yaitu biru, biru & merah, hijau, hijau & merah muda, kuning, kuning & merah muda, merah, merah & putih, merah muda, merah muda & biru, putih, ungu, ungu & hijau, dan ungu & kuning.
- 7) Desain indikator sebagai fungsi mengubah arah rel kereta berdasarkan input pemain
- 8) Desain rel kereta api, dua tipe yaitu lurus dan berbelok
- 9) Desain objek kereta, 14 jenis warna sesuai dengan jenis warna stasiun kereta api.

## C.2. State Paused

*State Paused* merupakan kondisi tampilan dimana pemain menjalankan fungsi untuk menghentikan permainan sementara. Contoh state paused telah diilustrasikan pada Gambar 5.13.



Gambar 5.13. Elemen visual interface Gameplay di *state paused*

Elemen visual pada state paused ini terdiri atas:

- 1) Desain frame untuk kondisi pause
- 2) Desain tulisan “pause”
- 3) Desain tombol yang berfungsi untuk melanjutkan permainan
- 4) Desain tombol yang berfungsi untuk menuju ke interface Pilihan Level
- 5) Desain tombol yang berfungsi untuk mengulang permainan

### **C.3. State Finished**

*State Finished* merupakan kondisi tampilan yang menandakan apakah pemain berhasil menyelesaikan permainan atau sebaliknya. Kondisi tampilan ini akan dipanggil dan ditampilkan apabila pemain telah memenuhi persyaratan tertentu. Kondisi kekalahan dan kemenangan pemain bergantung kepada indikator yaitu (1) *nyawa pemain* dan (2) *waktu permainan*. Apabila jumlah nyawa menjadi “0” saat permainan berlangsung seperti yang telah dijelaskan di detail permainan pada Tabel 5.4, maka tampilan state finished yang dipanggil merupakan tampilan

kondisi kekalahan pemain seperti yang ditampilkan pada Gambar 5.14 sisi kiri.



Gambar 5.14. Elemen visual interface Gameplay di *state finished*

Sebaliknya apabila pemain berhasil menjaga nilai nyawa tidak sama dengan 0 hingga waktu permainan habis maka tampilan *state finished* yang dipanggil merupakan tampilan kemenangan seperti yang ditampilkan pada Gambar 5.14 sisi kanan. Berdasarkan tampilan kedua kondisi tersebut, elemen visual pada *state finished* terdiri atas:

- 1) Desain label yang menandakan pemain memenangkan ("*stage cleared*") atau kalah ("*stage ended...*") dalam permainan.
- 2) Desain frame sebagai pelat penanda.
- 3) Desain tombol yang mewakili fungsi untuk kembali ke interface Pilihan Level dengan menampilkan iklan terlebih dahulu
- 4) Desain bintang sebagai visualisasi nilai.
- 5) Desain tombol yang mewakili fungsi untuk mengulang permainan
- 6) Desain tombol yang mewakili fungsi untuk memainkan permainan di tingkatan level selanjutnya.

Berdasarkan kebutuhan aesthetic berjenis visual pada ketiga state tersebut, peneliti kemudian mengupdate daftar kebutuhan dan tugas seperti yang disajikan pada Tabel 5.13.

Tabel 5.13. Update kebutuhan interface Gameplay (Aesthetic)

No	Daftar kebutuhan	Tugas
1	Konsep level yang berbeda dalam 14 tingkatan level	Mengonsep mapping permainan (termasuk posisi kereta dimulai, alur rel, lokasi stasiun kereta)
		Mendesain ilustrasi background
2	Mendefinisikan dan menampilkan waktu permainan.	Mendefinisikan dan membuat fungsi untuk menampilkan dan kalkulasi waktu permainan.
		Desain indikator waktu
3	Mendefinisikan dan menampilkan nyawa pemain.	Mendefinisikan dan membuat fungsi untuk menampilkan nyawa.
		Desain indikator nyawa pemain
4	Memunculkan (spawn) objek kereta pada posisi dan waktu tertentu.	Membuat fungsi untuk memunculkan kereta api.
		Desain objek kereta (14 jenis warna)
5	Menentukan kecepatan pergerakan kereta.	Membuat fungsi untuk menjalankan kereta berdasarkan kecepatan tertentu.
6	Merubah arah rel kereta berdasarkan input pemain.	Membuat fungsi untuk menerima input pemain yang berguna untuk merubah arah rel kereta.
		Desain indikator sebagai fungsi merubah arah rel kereta berdasarkan input pemain
7	Kereta dapat berjalan sesuai dengan alur rel kereta	Membuat fungsi pathfinding untuk objek kereta

No	Daftar kebutuhan	Tugas
		Desain rel kereta api
8	Pemain mendapatkan pengurangan nyawa untuk setiap kereta yang masuk ke stasiun yang salah (tidak sesuai dengan warna).	Membuat fungsi untuk kalkulasi nyawa setiap kesalahan terjadi.
		Desain stasiun kereta api
9	Pemain dapat menghentikan permainan sementara dan melanjutkannya kembali	Membuat fungsi untuk menghentikan permainan sementara (pause).
		Desain tombol pause untuk fungsi menghentikan permainan sementara
		Desain tulisan "pause"
		Desain frame untuk kondisi pause
10	Menampilkan jumlah bintang yang menandakan nilai dari pemain.	Membuat fungsi untuk mengakumulasi nilai yang didapat pemain berdasarkan jumlah kereta yang masuk ke stasiun yang benar.
		Membuat desain bintang untuk visualisasi nilai
		Desain frame sebagai pelat penanda.
		Desain label
11	Memungkinkan pemain menuju level selanjutnya apabila memenangkan permainan	Membuat fungsi menuju tingkatan level selanjutnya
		Desain tombol yang mewakili fungsi untuk memainkan permainan di tingkatan level selanjutnya.
12	Mengulang permainan.	Membuat fungsi untuk mengulang permainan.
		Desain tombol yang mewakili fungsi untuk mengulang permainan

No	Daftar kebutuhan	Tugas
13	Berpindah ke Interface Pilihan Level dengan menampilkan iklan terlebih dahulu.	Membuat fungsi untuk menampilkan iklan yang kemudian berpindah ke interface Pilihan Level.
		Desain tombol yang mewakili fungsi untuk kembali ke interface Pilihan Level
<b>Keterangan</b>		
 <ul style="list-style-type: none"> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: yellow; margin-right: 5px;"></span> : Kebutuhan / tugas yang didapat pada tahapan Mechanic</li> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: blue; margin-right: 5px;"></span> : Kebutuhan / tugas yang didapat pada tahapan Dynamic</li> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: green; margin-right: 5px;"></span> : Kebutuhan dan tugas yang didapat pada tahapan Aesthetic</li> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: orange; margin-right: 5px;"></span> : Kebutuhan yang memiliki tugas teknis program dan visual</li> </ul>		

Selanjutnya, berdasarkan ketiga interface yang telah dianalisa yaitu Home, Pilihan Level, dan Gameplay kemudian peneliti merangkum update daftar kebutuhan dan tugas sehingga daftar kebutuhan akan menjadi seperti pada Tabel 5.14.

Tabel 5.14. Update daftar kebutuhan proyek P-MCP (Aesthetic)

No	Daftar kebutuhan (fungsi)	Tugas
<i>Interface Home</i>		
1	Berpindah ke Interface Pilihan Level	Membuat fungsi untuk berpindah ke Tampilan Pilihan Level
		Membuat desain tombol “play” yang mengindikasikan memulai permainan

No	Daftar kebutuhan (fungsi)	Tugas
2	Pemain dapat mengaktifkan atau mematikan efek suara	Membuat fungsi untuk mengaktifkan atau mematikan efek suara
		Membuat desain tombol efek suara “aktif” dan “nonaktif”
3	Menampilkan iklan pada aplikasi	Membuat fungsi untuk menampilkan iklan
4	Menghilangkan iklan dengan mengaktifkan pembelian dalam aplikasi	Membuat fungsi untuk menghilangkan iklan
		Membuat desain tombol “no ads”
5	Menampilkan website developer	Membuat fungsi untuk membuka browser dengan url website developer.
		Membuat desain tombol Info
<b><i>Interface Pilihan Level</i></b>		
6	Memiliki visualisasi interface Pilihan Level yang menarik	Merancang konsep dan komposisi visual
		Mendesain background untuk interface Pilihan Level
		Mendesain label “Level Selection” pada interface Pilihan Level
7	Berpindah ke Interface Home	Membuat fungsi untuk berpindah ke interface Home
		Membuat desain tombol untuk fungsi kembali ke interface Home
8	Berpindah ke Interface Gameplay sesuai pilihan level	Membuat fungsi untuk berpindah ke interface Gameplay

No	Daftar kebutuhan (fungsi)	Tugas
9	Navigasi daftar tingkatan level	Membuat fungsi untuk menampilkan daftar tingkatan level dengan range tertentu
		Membuat desain tombol navigasi daftar tingkatan level
10	Pemain dapat memainkan level tertentu apabila level sebelumnya telah diselesaikan	Membuat fungsi untuk mengunci dan membuka level.
		Membuat desain tombol untuk tingkatan level dengan kondisi terbuka dan terkunci
<b><i>Interface Gameplay</i></b>		
11	Konsep level yang berbeda dalam 14 tingkatan level	Mengonsep mapping permainan (termasuk posisi kereta dimulai, alur rel, lokasi stasiun kereta)
		Mendesain ilustrasi background
12	Memunculkan (spawn) objek kereta pada posisi dan waktu tertentu.	Membuat fungsi untuk memunculkan kereta api.
		Desain objek kereta (14 jenis warna)
13	Mendefinisikan dan menampilkan waktu permainan.	Mendefinisikan dan membuat fungsi untuk menampilkan dan kalkulasi waktu permainan.
		Desain indikator waktu
14	Mendefinisikan dan menampilkan nyawa pemain.	Mendefinisikan dan membuat fungsi untuk menampilkan dan kalkulasi nyawa.
		Desain indikator nyawa pemain

No	Daftar kebutuhan (fungsi)	Tugas
14	Pemain mendapatkan pengurangan nyawa untuk setiap kereta yang masuk ke stasiun yang salah (tidak sesuai dengan warna).	Membuat fungsi untuk kalkulasi nyawa setiap kesalahan terjadi.
		Desain stasiun kereta api
16	Menentukan kecepatan pergerakan kereta.	Membuat fungsi untuk menjalankan kereta berdasarkan kecepatan tertentu.
17	Merubah arah rel kereta berdasarkan input pemain.	Membuat fungsi untuk menerima input pemain yang berguna untuk merubah arah rel kereta.
		Desain indikator sebagai fungsi merubah arah rel kereta berdasarkan input pemain
18	Kereta dapat berjalan sesuai dengan alur rel kereta	Membuat fungsi pathfinding untuk objek kereta
		Desain rel kereta api
19	Pemain dapat menghentikan permainan sementara dan melanjutkannya kembali	Membuat fungsi untuk menghentikan permainan sementara.
		Desain tombol pause untuk fungsi menghentikan permainan sementara
		Desain tulisan "pause"
		Desain frame untuk kondisi pause
20	Menampilkan jumlah bintang yang menandakan nilai dari pemain.	Membuat fungsi untuk mengakumulasi nilai yang didapat pemain berdasarkan jumlah kereta

No	Daftar kebutuhan (fungsi)	Tugas
		yang masuk ke stasiun yang benar dan menampikannya.
		Membuat desain bintang untuk visualisasi nilai
		Desain frame sebagai pelat penanda.
		Desain label
21	Memungkinkan pemain menuju level selanjutnya apabila memenangkan permainan	Membuat fungsi menuju tingkatan level selanjutnya
		Desain tombol yang mewakili fungsi untuk memainkan permainan di tingkatan level selanjutnya.
22	Mengulang permainan.	Membuat fungsi untuk mengulang permainan.
		Desain tombol yang mewakili fungsi untuk mengulang permainan
23	Berpindah ke Interface Pilihan Level dengan menampilkan iklan terlebih dahulu.	Membuat fungsi untuk menampilkan iklan yang kemudian berpindah ke interface Pilihan Level.
		Desain tombol yang mewakili fungsi untuk kembali ke interface Pilihan Level

### Keterangan

-  : Kebutuhan / tugas yang didapat pada tahapan Mechanic
-  : Kebutuhan / tugas yang didapat pada tahapan Dynamic
-  : Kebutuhan dan tugas yang didapat pada tahapan Aesthetic
-  : Kebutuhan yang memiliki tugas teknis program dan visual

#### 5.1.1.5. *Experience* proyek

Experience atau pengalaman pemain merupakan faktor yang mempengaruhi bagaimana seharusnya permainan digital berinteraksi dengan pemain. Jenis interaksi yang ditekankan pada permainan digital umumnya adalah pada visualisasi permainan. Contohnya, pada indikator permainan seperti nilai/score dalam permainan digital selalu terletak pada area yang mudah dilihat oleh pemain seperti bagian sudut atau pojok dari layar, Lebih spesifik di bagian pojok kiri (area merah) pada permainan digital “Fruit Ninja” di Gambar 5.15



Gambar 5.15. Indikator nilai pada permainan Fruit Ninja

Atau seperti pada permainan digital “Stack” dengan indikator yang terletak di sebelah kanan (area merah) pada Gambar 5.16.



Gambar 5.16. Indikator nilai pada permainan Stack

Kedua gambar diatas merupakan contoh user experience pada visualisasi komponen score atau nilai dalam permainan digital. Tentunya dalam merancang visualisasi dan komposisi peletakan komponen-komponen lainnya memiliki aturan tersendiri sehingga menambah kompleksitas dalam merancang sisi Aesthetic dalam permainan digital. Dalam proyek P-MCP secara umum konsep Aesthetic yang digunakan menggunakan tampilan permainan secara vertikal (seperti Gambar 5.16) sehingga seluruh daftar kebutuhan dan tugas yang berhubungan dengan Aesthetic harus menyesuaikan dengan ketentuan tersebut supaya pemain dapat secara optimal menikmati dan tidak terganggu dengan visualisasi yang tidak sesuai.

#### **5.1.1.6. Assuptions dan Constrain proyek**

Proyek P-MCP merupakan proyek permainan digital yang ditarget ke pasar mobile baik android maupun IOS. Oleh karena itu, hal teknis yang harus diperhatikan antara lain:

- 1) **Versi android dan IOS yang ditarget** – Hal ini akan berpengaruh terhadap fitur tertentu yang dirancang, karena terdapat kemungkinan versi android atau IOS tidak support terhadap fungsi tertentu sehingga fitur pada permainan digital P-MCP tidak dapat berjalan dengan semestinya. Pada proyek P-MCP target android adalah versi 7.0 (Nougat) dan untuk target IOS adalah versi IOS 11.
- 2) **Resolusi maksimal layar pada target android dan IOS** – Tim pengembang harus memperhatikan resolusi maksimal dari setiap versi android dan IOS. Lalai dalam memperhatikan resolusi maksimal akan mengakibatkan tampilan akan menjadi buram atau bahkan terpecah-pecah dikarenakan resolusi gambar yang diciptakan terlalu kecil.
- 3) **Game engine yang dipakai** – Tim pengembang harus memperhatikan game engine yang dipakai dikarenakan setiap game engine membutuhkan komponen tertentu untuk membangun permainan digital pada target yang ditentukan (contoh target: *PS4, STEAM, Android, IOS, Nitendo Switch*, dsb.).
- 4) **Interaksi permainan** – tim pengembang harus memperhatikan cara pemain berinteraksi dengan permainan. Interaksi dengan cara *touch* tentu berbeda rasanya dengan cara interaksi dengan *swipe* atau *drag & drop*. Tim pengembang perlu menentukan pendekatan interaksi terbaik dengan menyesuaikan dengan konsep permainan. dalam proyek P-MCP pendekatan yang dipakai adalah *touch* karena konsep permainan yang berfokus untuk merubah arah laju kereta api.

#### **5.1.1.7. Daftar kebutuhan final proyek P-MCP**

Bedasarkan langkah-langkah analisis komponen mechanic, dynamic, aesthetic, experience, dan assumption and constrain pada proyek P-MCP didapatkan daftar kebutuhan final yang akan digunakan sebagai masukan ditahap selanjutnya. Daftar kebutuhan dan tugas final pada proyek P-MCP disajikan pada Tabel 5.15.

Tabel 5.15. Daftar kebutuhan final proyek P-MCP

No	Daftar kebutuhan (fungsi)	Tugas
<b><i>Interface Home</i></b>		
1	Berpindah ke Interface Pilihan Level	Membuat fungsi untuk berpindah ke Tampilan Pilihan Level
		Membuat desain tombol “play” yang mengindikasikan memulai permainan
2	Pemain dapat mengaktifkan atau mematikan efek suara	Membuat fungsi untuk mengaktifkan atau mematikan efek suara
		Membuat desain tombol efek suara “aktif” dan “nonaktif”
3	Menampilkan iklan pada aplikasi	Membuat fungsi untuk menampilkan iklan
4	Menghilangkan iklan dengan mengaktifkan pembelian dalam aplikasi	Membuat fungsi untuk menghilangkan iklan
		Membuat desain tombol “no ads”
5	Menampilkan website developer	Membuat fungsi untuk membuka browser dengan url website developer.
		Membuat desain tombol Info
6	Memiliki visualisasi Interface Home yang menarik	Membuat desain background untuk interface Home
		Membuat desain logo untuk identitas aplikasi
		Merancang komposisi elemen-elemen tampilan visual.
<b><i>Interface Pilihan Level</i></b>		
7	Memiliki visualisasi interface Pilihan Level yang menarik	Merancang konsep dan komposisi visual

No	Daftar kebutuhan (fungsi)	Tugas
		Mendesain background untuk interface Pilihan Level
		Mendesain label "Level Selection" pada interface Pilihan Level
8	Berpindah ke Interface Home	Membuat fungsi untuk berpindah ke interface Home
		Membuat desain tombol untuk fungsi kembali ke interface Home
9	Berpindah ke Interface Gameplay sesuai pilihan level	Membuat fungsi untuk berpindah ke interface Gameplay
10	Navigasi daftar tingkatan level	Membuat fungsi untuk menampilkan daftar tingkatan level dengan range tertentu
		Membuat desain tombol navigasi daftar tingkatan level
11	Pemain dapat memainkan level tertentu apabila level sebelumnya telah diselesaikan	Membuat fungsi untuk mengunci dan membuka level.
		Membuat desain tombol untuk tingkatan level dengan kondisi terbuka dan terkunci
<b><i>Interface Gameplay</i></b>		
12	Konsep level yang berbeda dalam 14 tingkatan level	Mengonsep mapping permainan (termasuk posisi kereta dimulai, alur rel, lokasi stasiun kereta)
		Mendesain ilustrasi background
13	Memunculkan (spawn) objek kereta pada posisi dan waktu tertentu.	Membuat fungsi untuk memunculkan kereta api.
		Desain objek kereta (14 jenis warna)

No	Daftar kebutuhan (fungsi)	Tugas
14	Mendefinisikan dan menampilkan waktu permainan.	Mendefinisikan dan membuat fungsi untuk menampilkan dan kalkulasi waktu permainan.
		Desain indikator waktu
15	Mendefinisikan dan menampilkan nyawa pemain.	Mendefinisikan dan membuat fungsi untuk menampilkan dan kalkulasi nyawa.
		Desain indikator nyawa pemain
16	Pemain mendapatkan pengurangan nyawa untuk setiap kereta yang masuk ke stasiun yang salah (tidak sesuai dengan warna).	Membuat fungsi untuk kalkulasi nyawa setiap kesalahan terjadi.
		Desain stasiun kereta api
17	Menentukan kecepatan pergerakan kereta.	Membuat fungsi untuk menjalankan kereta berdasarkan kecepatan tertentu.
18	Merubah arah rel kereta berdasarkan input pemain.	Membuat fungsi untuk menerima input pemain yang berguna untuk merubah arah rel kereta.
		Desain indikator sebagai fungsi merubah arah rel kereta berdasarkan input pemain
19	Kereta dapat berjalan sesuai dengan alur rel kereta	Membuat fungsi pathfinding untuk objek kereta
		Desain rel kereta api
20	Pemain dapat menghentikan permainan sementara dan melanjutkannya kembali	Membuat fungsi untuk menghentikan permainan sementara.
		Desain tombol pause untuk fungsi menghentikan permainan sementara

No	Daftar kebutuhan (fungsi)	Tugas
		Desain tulisan "pause"
		Desain frame untuk kondisi pause
21	Menampilkan jumlah bintang yang menandakan nilai dari pemain.	Membuat fungsi untuk mengakumulasi nilai yang didapat pemain berdasarkan jumlah kereta yang masuk ke stasiun yang benar dan menampikannya.
		Membuat desain bintang untuk visualisasi nilai
		Desain frame sebagai pelat penanda.
		Desain label
22	Memungkinkan pemain menuju level selanjutnya apabila memenangkan permainan	Membuat fungsi menuju tingkatan level selanjutnya
		Desain tombol yang mewakili fungsi untuk memainkan permainan di tingkatan level selanjutnya.
23	Mengulang permainan.	Membuat fungsi untuk mengulang permainan.
		Desain tombol yang mewakili fungsi untuk mengulang permainan
24	Berpindah ke Interface Pilihan Level dengan menampilkan iklan terlebih dahulu.	Membuat fungsi untuk menampilkan iklan yang kemudian berpindah ke interface Pilihan Level.
		Desain tombol yang mewakili fungsi untuk kembali ke interface Pilihan Level

Keterangan	
	: Kebutuhan / tugas yang didapat pada tahapan Mechanic
	: Kebutuhan / tugas yang didapat pada tahapan Dynamic
	: Kebutuhan dan tugas yang didapat pada tahapan Aesthetic
	: Kebutuhan yang memiliki tugas teknis program dan visual

Berdasarkan daftar kebutuhan final proyek P-MCP pada Tabel 5.15, terlihat bahwa elemen aesthetic merupakan faktor non fungsional penting. Hal tersebut dibuktikan dengan 30 dari 51 daftar tugas (58.82%) merupakan pekerjaan yang berhubungan dengan Aesthetic.

### 5.1.2. Kebutuhan pada proyek P-MCE

Sub bab ini akan menjelaskan proses mendapatkan daftar kebutuhan pada proyek P-MCE.

#### 5.1.2.1. Overview proyek

Proyek permainan digital P-MCE bertujuan untuk mengedukasi pemain tentang keselamatan dalam penerbangan. Lingkup pada proyek permainan digital ini adalah mencakup edukasi memilah barang bawaan legal dan ilegal dibawa oleh penumpang pesawat terbang dengan mode kontrol *drag&drop*. Instruksi untuk memilah barang bawaan diilustrasikan pada Gambar 5.17.



Gambar 5.17. Objektif pertama permainan digital P-MCE

Selain itu, proyek ini juga memberikan edukasi kepada pemain tentang prosedur dan perilaku penumpang saat pesawat terbang akan lepas landas antara

lain yaitu penumpang dalam posisi duduk, menutup meja, mematikan smartphone, dan mengencangkan sabuk pengaman. Gambar 5.18 mengilustrasikan tujuan tersebut dalam model permainan.



Gambar 5.18. Objektif kedua permainan digital P-MCE

Proyek P-MCE ditargetkan untuk platform mobile berbasis android dan memiliki genre casual. Poin-poin kunci dari proyek P-MCE dijabarkan pada Tabel 5.16.

Tabel 5.16. Overview proyek P-MCE

<b>Nama Proyek</b>	P-MCE
<b>Genre</b>	Casual
<b>Tema</b>	Pesawat terbang
<b>Konsep permainan</b>	<p>Pemmainan bertujuan mengedukasi pemain dengan dua objektif yaitu:</p> <ol style="list-style-type: none"> <li>1) Memisahkan barang-barang yang dibawa menjadi barang yang boleh dibawa dan yang tidak boleh dibawa (ilegal pada transportasi penerbangan atau harus berada di</li> </ol>

	<p>bagasi) dengan menggunakan interaksi drag&amp;drop.</p> <p>2) Menunjukkan Perilaku penumpang yang benar saat pesawat akan lepas landas dengan mode interaksi <i>touch</i>, antara lain penumpang dalam posisi duduk, menutup meja, mematikan smartphone, dan mengencangkan sabuk pengaman.</p>
--	---

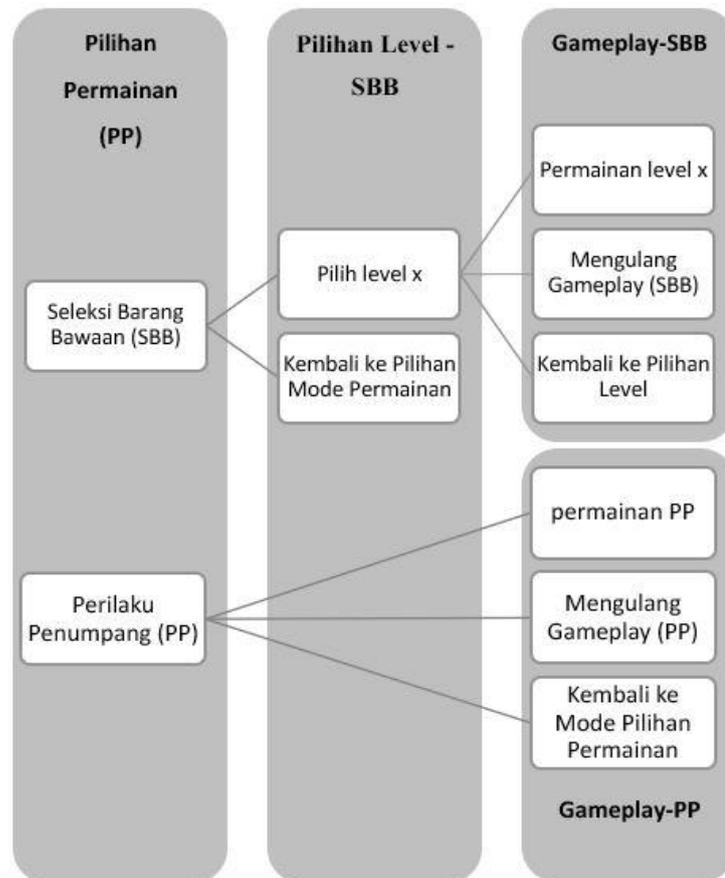
#### 5.1.2.2. *Mechanic* proyek

Permainan digital P-MCE dirancang memiliki tampilan untuk memilih mode permainan. Mode permainan dibagi menjadi dua yaitu mode Seleksi Barang Bawaan (SBB) yaitu mode permainan yang sesuai dengan objektif pertama yaitu “mendidik pemain tentang barang-barang yang boleh di kabin pesawat atau yang dilarang (harus berada di bagasi pesawat atau dilarang sepenuhnya dalam transportasi penerbangan)”. Seperti pada proyek P-MCP, proyek P-MCE juga memiliki tingkatan level dengan prosedur yang serupa dengan proyek permainan digital P-MCP yaitu pemain tidak bisa memainkan sembarang tingkatan level secara langsung. Pemain dapat memainkan tingkatan level yang lebih tinggi apabila pemain telah menyelesaikan tingkatan level yang lebih rendah terlebih dahulu. Dalam mode ini juga terdapat fungsi untuk mengulang permainan dan fungsi untuk kembali ke pilihan level.

Mode permainan kedua yaitu Perilaku Penumpang (PP) bertujuan untuk memenuhi objektif kedua proyek yaitu “mendidik pemain tentang perilaku penumpang saat pesawat akan lepas landas”. Mode ini juga memiliki fungsi untuk mengulang permainan dan kembali ke Pilihan Mode Permainan (PMP). Berdasarkan penjelasan tersebut, mekanisme permainan dari proyek P-MCE dapat diilustrasikan seperti Gambar 5.19.

Proyek P-MCE terbagi menjadi empat interface yaitu Pilihan Permainan, Pilihan Level SBB, Gameplay SBB, dan Gameplay PP. Selanjutnya peneliti akan

menganalisa mekanisme masing-masing interface untuk mendapatkan daftar kebutuhan dan tugas yang dikerjakan pada masing-masing kebutuhan.



Gambar 5.19. Alur permainan P-MCE

### A. Interface Pilihan Permainan

Mekanisme dalam interface Pilihan Permainan hanya berfungsi sebagai interface yang menyajikan pilihan mode permainan yaitu mode permainan seleksi barang bawaan (SBB) dan mode permainan Perilaku Penumpang (PP). Apabila pemain memilih mode SBB maka interface Pilihan Permainan akan memanggil interface Pilihan Level SBB sedangkan Gameplay PP akan dipanggil jika pemain memilih mode permainan PP. Dalam interface Pilihan Permainan juga terdapat fungsi untuk menampilkan kategori barang bawaan sebagai petunjuk permainan

pada seleksi barang bawaan (SBB). Gambar 5.20 merupakan salah satu tampilan kategori barang dilarang sepenuhnya dari kategori barang bawaan.



Gambar 5.20. Kategori barang yang dilarang

Selain itu, pada interface ini juga menampilkan peraturan perilaku penumpang yang dapat digunakan sebagai petunjuk pada permainan perilaku penumpang. Gambar 5.21 mengilustrasikan salah satu perilaku penumpang yang harus dihindari saat menggunakan jasa transportasi pesawat terbang.



Gambar 5.21. Peraturan perilaku penumpang

Berdasarkan mekanisme tersebut, peneliti mendapatkan kebutuhan mekanis pada Interface ini yaitu seperti yang disajikan pada Tabel 5.17.

Tabel 5.17. Kebutuhan pada Interface Pilihan Permainan

No	Daftar kebutuhan (fitur)	Tugas
1	Berpindah ke Interface Pilihan Level SBB	Membuat fungsi untuk memanggil interface Pilihan Level SBB
2	Berpindah ke Interface Gameplay PP	Membuat fungsi untuk memanggil interface Gameplay PP
3	Menampilkan contoh barang bawaan berdasarkan kategori	Membuat fungsi untuk menampilkan aset -aset barang berdasarkan kategori
4	Menampilkan contoh perilaku penumpang yang harus ditaati.	Membuat fungsi untuk menampilkan peraturan perilaku penumpang.

## B. Interface Pilihan Level SBB

Proyek P-MCE memiliki mekanisme penguncian level pada Interface Pilihan Level SBB seperti pada permainan digital P-MCP. Selain itu, dalam interface ini juga terdapat tiga mekanisme lain yaitu:

- 1) Fungsi kembali dengan cara memanggil Interface Pilihan Permainan
- 2) Fungsi untuk menampilkan waktu terbaik pada setiap tingkatan level
- 3) Fungsi untuk memainkan permainan pada level tertentu dengan cara memanggil Interface Gameplay SBB sesuai dengan tingkatan level yang dipilih.

Berdasarkan rancangan mekanisme tersebut, maka kebutuhan dan tugas yang didapat berdasarkan mekanisme di Interface Pilihan Level SBB yaitu seperti yang disajikan pada Tabel 5.18.

Tabel 5.18. kebutuhan pada Interface Pilihan Level SBB

No	Daftar kebutuhan (fitur)	Tugas
1	Berpindah ke interface Pilihan Permainan	Membuat fungsi untuk memanggil interface Pilihan Permainan
2	Berpindah ke Interface Gameplay SBB sesuai pilihan level	Membuat fungsi untuk memanggil interface Gameplay SBB dengan mengirimkan informasi level yang dipilih
3	Pemain dapat memainkan level tertentu apabila level sebelumnya telah diselesaikan	Membuat fungsi untuk mengunci dan membuka level.
4	Setiap tingkatan level memiliki catatan waktu terbaik	Membuat fungsi untuk menampilkan waktu terbaik pada setiap tingkatan level.

### C. Interface Gameplay SBB

Untuk menganalisa kebutuhan pada interface Gameplay, peneliti menggunakan pendekatan tiga fase seperti pada ilustrasi 5.3 yang terdiri dari fase Inisialisasi (*Start*), proses (*process*), dan selesai (*Finish*). Ketiga fase tersebut akan digunakan untuk mengembangkan konsep permainan pada mode SBB seperti yang telah dijabarkan pada Tabel 5.16 yaitu:

“Memisahkan barang-barang yang dibawa menjadi barang yang boleh dibawa dan yang tidak boleh dibawa (ilegal pada transportasi penerbangan atau harus berada di bagasi)”

Tabel 5.19 menyajikan pengembangan konsep permainan SBB kedalam tiga fase permainan tersebut.

Tabel 5.19. Pengembangan konsep berdasarkan fase permainan

Fase	No	Detail mekanisme permainan
Start	1	Memunculkan barang-barang dalam kondisi random. Barang yang dimunculkan memiliki tiga label yaitu (1) barang berlabel legal (yang dapat dibawa), (2) barang berlabel bagasi (barang yang harus diletakkan dalam bagasi), dan (3) barang berlabel terlarang (barang yang dilarang dibawa saat penggunaan transportasi pesawat terbang)
	2	Menginisialisasi timer permainan
	3	Menginisialisasi area <i>drag</i> pada masing-masing barang supaya barang dapat di drag oleh pemain.
	4	Menginisialisasi area <i>drop</i> supaya barang dapat di drop oleh pemain di area ini.
	5	Menginisialisasi jumlah barang yang harus dibuang ke area drop.
Process	6	Menjalankan timer
	7	Melakukan pendeteksian area <i>drop</i> barang apabila barang di drag masuk dalam jangkauan area <i>drop</i> . Pendeteksian dilakukan dengan penyesuaian label pada barang dengan label yang ada pada area drop (contoh barang berlabel terlarang harus di drop pada area berlabel terlarang)
	8	Melakukan kalkulasi pengurangan jumlah barang yang harus dibuang pada area drop apabila barang berlabel telah drop pada area yang berlabel sama.
Finish	9	Mencatat waktu berdasarkan timer
	10	Mengupdate waktu yang tersimpan apabila waktu yang didapat lebih kecil.
	11	Pemain dapat mengulang permainan
	12	Pemain dapat memanggil interface Pilihan Level

Berdasarkan penjabaran ketiga fase permainan tersebut, selanjutnya peneliti melakukan proses analisa kebutuhan dan tugas yang harus dikerjakan supaya mekanisme permainan SBB dapat berjalan dengan baik dan dapat berjalan seperti konsep yang dirancang . Tabel 5.20 merupakan hasil kebutuhan dan tugas yang dibutuhkan dan dikerjakan berdasarkan mekanisme permainan terkait.

Tabel 5.20. Kebutuhan pada Interface Gameplay SBB

No	Daftar kebutuhan	Tugas	Referensi Mekanisme permainan
1	Menginisialisasi dan menjalankan timer permainan	Membuat fungsi untuk menampilkan timer dan menjalankan timer	Menginisialisasi timer permainan
			Menjalankan timer
2	Memunculkan barang berlabel legal, terlarang, dan bagasi secara random dengan komposisi tertentu	Membuat fungsi untuk menampilkan barang secara random.	Memunculkan barang-barang dalam kondisi random. Barang yang dimunculkan memiliki tiga label yaitu (1) barang berlabel legal (yang dapat dibawa), (2) barang berlabel bagasi (barang yang harus diletakkan dalam bagasi), dan (3) barang berlabel terlarang (barang yang dilarang dibawa saat penggunaan
		Menginisiasi label pada barang	

No	Daftar kebutuhan	Tugas	Referensi Mekanisme permainan
			transportasi pesawat terbang)
3	Kemampuan drag & drop pada masing-masing barang	Membuat fungsi <i>drag&amp;drop</i> dan meng-attach fungsi tersebut pada setiap barang	Menginisialisasi area <i>drag</i> pada masing-masing barang supaya barang dapat di drag oleh pemain.
4	Membuat area drop barang berlabel terlarang, dan bagasi	Membuat fungsi untuk menginisiasi area drop berlabel.	Menginisialisasi area <i>drop</i> supaya barang dapat di drop oleh pemain di area ini.
5	Terdapat inisialisasi jumlah target barang yang harus di filter setiap levelnya	Menginisialisasi nilai jumlah target barang.	Menginisialisasi jumlah barang yang harus dibuang ke area drop.
6	Barang yang berlabel sama dengan area drop dapat di drop pada area drop.	Membuat fungsi pengecekan untuk mendeteksi kesesuaian label pada barang dan area drop	Melakukan pendeteksian area <i>drop</i> barang apabila barang di drag masuk dalam jangkauan area <i>drop</i> . Pendeteksian dilakukan dengan penyesuaian label pada barang dengan label yang ada pada area drop (contoh barang berlabel terlarang harus di drop

No	Daftar kebutuhan	Tugas	Referensi Mekanisme permainan
			pada area berlabel terlarang)
7	Mengurangi nilai jumlah barang apabila barang telah di drop pada area yang sesuai	Membuat fungsi untuk mengurangi nilai pada jumlah target barang	Melakukan kalkulasi pengurangan jumlah barang yang harus dibuang pada area drop apabila barang berlabel telah drop pada area yang berlabel sama.
8	Apabila jumlah targer barang = 0, maka sistem akan mencatat waktu timer	Membuat fungsi untuk mencatat waktu timer saat nilai jumlah target barang = 0.	Mencatat waktu berdasarkan timer
9	Mengupdate catatan waktu terbaik	Membuat fungsi untuk membandingkan catatan waktu dengan catatan waktu permainan Mengupdate catatan waktu dengan catatan waktu tercepat	Mengupdate waktu yang tersimpan apabila waktu yang didapat lebih kecil.
10	Pemain dapat mengulang permainan pada	Membuat fungsi untuk mengulang permainan	Pemain dapat mengulang permainan

No	Daftar kebutuhan	Tugas	Referensi Mekanisme permainan
	level yang dimainkannya		
11	Pemain dapat kembali ke tampilan pilihan level	Membuat fungsi untuk memanggil interface Pilihan Level SBB	Pemain dapat memanggil interface Pilihan Level

#### D. Interface Gameplay PP

Dengan menggunakan tahapan yang sama dengan Gameplay SBB, yaitu menjabarkan konsep permainan PP kedalam tiga fase permainan. konsep permainan PP seperti yang telah dijabarkan pada Tabel 5.16 adalah Menunjukkan Perilaku penumpang yang benar saat pesawat akan lepas landas dengan mode interaksi *touch*, antara lain penumpang dalam posisi duduk, menutup meja, mematikan smartphone, dan mengencangkan sabuk pengaman.

Dari hasil penjabaran tersebut peneliti mendapatkan detail mekanisme permainan PP. Selanjutnya peneliti menganalisa kebutuhan beserta tugas yang harus dilakukan supaya mekanisme dapat berjalan sesuai konsep permainan. Hasilnya, pada Tabel 5.21 merupakan daftar kebutuhan pada Interface Gameplay PP berdasarkan mekanisme permainan pada Gameplay PP.

Tabel 5.21. Kebutuhan pada Interface Gameplay PP

No	Daftar kebutuhan	Tugas	Referensi Mekanisme permainan
1	Menampilkan aset – aset yang digunakan sebagai persyaratan	Membuat fungsi untuk menampilkan aset	Menampilkan aset-aset yang pada koordinat tertentu

No	Daftar kebutuhan	Tugas	Referensi Mekanisme permainan
	perilaku penumpang pada posisi tertentu	pada koordinat tertentu	
2	Setiap asset dapat berinteraksi dengan pemain apabila disentuh ( <i>touch</i> ) dengan menampilkan opsi pilihan	Membuat fungsi interaksi touch pada setiap aset dengan hasil interaksi berupa opsi pilihan	Menampilkan opsi pilihan apabila aset di touch oleh pemain
3	Setiap pilihan opsi yang benar akan mengakibatkan satu persyaratan terpenuhi dan merubah tampilan permainan	Membuat fungsi untuk hide dan unhide pada asset sebagai akibat dari pilihan opsi yang benar	Apabila pemain memilih opsi yang benar maka satu persyaratan terpenuhi.
4	Menampilkan instruksi persyaratan apa yang belum terpenuhi apabila pemain menekan tombol "OK" dan membuka tampilan kemenangan apabila semua persyaratan telah terpenuhi.	Membuat fungsi untuk pengecekan persyaratan pada tombol "OK"	Menekan tombol tertentu saat persyaratan belum terpenuhi akan mengakibatkan tampilan instruksi syarat yang belum terpenuhi Menekan tombol "ok" saat semua persyaratan terpenuhi akan menampilkan akhir permainan

No	Daftar kebutuhan	Tugas	Referensi Mekanisme permainan
5	Pemain dapat mengulang kembali permainan PP	Membuat fungsi untuk mengulang permainan PP	Pemain dapat mengulang permainan PP
6	Pemain dapat kembali ke tampilan pilihan mode permainan	Membuat fungsi untuk memanggil interface Pilihan Permainan.	Pemain dapat memanggil Interface Pilihan Permainan

Selanjutnya, peneliti merangkum hasil daftar kebutuhan yang didapat pada fase analisa mekanisme di empat interface yaitu Pilihan Permainan, Pilihan Level SBB, Gameplay SBB, dan Gameplay PP sehingga menghasilkan daftar kebutuhan dan tugas seperti pada Tabel 5.22.

Tabel 5.22. Daftar kebutuhan proyek P-MCE

No	Daftar kebutuhan (fungsi)	Tugas
<i>Interface Pilihan Permainan</i>		
1	Berpindah ke Interface Pilihan Level SBB	Membuat fungsi untuk memanggil interface Pilihan Level SBB
2	Berpindah ke Interface Gameplay PP	Membuat fungsi untuk memanggil interface Gameplay PP
3	Menampilkan contoh barang bawaan berdasarkan kategori	Membuat fungsi untuk menampilkan aset -aset barang berdasarkan kategori
4	Menampilkan contoh perilaku penumpang yang harus ditaati.	Membuat fungsi untuk menampilkan peraturan perilaku penumpang.

No	Daftar kebutuhan (fungsi)	Tugas
<b><i>Interface Pilihan Level SBB</i></b>		
5	Berpindah ke interface Pilihan Permainan	Membuat fungsi untuk memanggil interface Pilihan Permainan
6	Berpindah ke Interface Gameplay SBB sesuai pilihan level	Membuat fungsi untuk memanggil interface Gameplay SBB dengan mengirimkan informasi level yang dipilih
7	Pemain dapat memainkan level tertentu apabila level sebelumnya telah diselesaikan	Membuat fungsi untuk mengunci dan membuka level.
8	Setiap tingkatan level memiliki catatan waktu terbaik	Membuat fungsi untuk menampilkan waktu terbaik pada setiap tingkatan level.
<b><i>Interface Gameplay SBB</i></b>		
9	Menginisialisasi dan menjalankan timer permainan	Membuat fungsi untuk menampilkan timer dan menjalankan timer
10	Memunculkan barang berlabel legal, terlarang, dan bagasi secara random dengan komposisi tertentu	Membuat fungsi untuk menampilkan barang secara random.
		Menginisiasi label pada barang
11	Kemampuan drag & drop pada masing-masing barang	Membuat fungsi <i>drag&amp;drop</i> dan meng-attach fungsi tersebut pada setiap barang
12	Membuat area drop barang berlabel terlarang, dan bagasi	Membuat fungsi untuk menginisiasi area drop berlabel.

No	Daftar kebutuhan (fungsi)	Tugas
13	Terdapat inisialisasi jumlah target barang yang harus di filter setiap levelnya	Menginisialisasi nilai jumlah target barang.
14	Barang yang berlabel sama dengan area drop dapat di drop pada area drop.	Membuat fungsi pengecekan untuk mendeteksi kesesuaian label pada barang dan area drop
15	Mengurangi nilai jumlah barang apabila barang telah di drop pada area yang sesuai	Membuat fungsi untuk mengurangi nilai pada jumlah target barang
16	Apabila jumlah targer barang = 0, maka sistem akan mencatat waktu timer	Membuat fungsi untuk mencatat waktu timer saat nilai jumlah target barang = 0.
17	Mengupdate catatan waktu terbaik	Membuat fungsi untuk membandingkan catatan waktu dengan catatan waktu permainan Mengupdate catatan waktu dengan catatan waktu tercepat
18	Pemain dapat mengulang permainan pada level yang dimainkannya	Membuat fungsi untuk mengulang permainan
19	Pemain dapat kembali ke tampilan pilihan level	Membuat fungsi untuk memanggil interface Pilihan Level SBB

No	Daftar kebutuhan (fungsi)	Tugas
<i>Interface Gameplay PP</i>		
20	Menampilkan aset – aset yang digunakan sebagai persyaratan perilaku penumpang pada posisi tertentu	Membuat fungsi untuk menampilkan aset pada koordinat tertentu
21	Setiap aset dapat berinteraksi dengan pemain apabila disentuh ( <i>touch</i> ) dengan menampilkan opsi pilihan	Membuat fungsi interaksi touch pada setiap aset dengan hasil interaksi berupa opsi pilihan
22	Setiap pilihan opsi yang benar akan mengakibatkan satu persyaratan terpenuhi dan merubah tampilan permainan	Membuat fungsi untuk hide dan unhide pada asset sebagai akibat dari pilihan opsi yang benar
23	Menampilkan instruksi persyaratan apa yang belum terpenuhi apabila pemain menekan tombol “OK” dan membuka tampilan kemenangan apabila semua persyaratan terpenuhi	Membuat fungsi untuk pengecekan persyaratan.
24	Pemain dapat mengulang kembali permainan PP	Membuat fungsi untuk mengulang permainan PP
25	Pemain dapat kembali ke tampilan pilihan mode permainan	Membuat fungsi untuk memanggil interface Pilihan Permainan.

### 5.1.2.3. *Dynamic proyek*

Pada sub bab ini peneliti akan menganalisa elemen dinamis pada proyek P-MCE untuk mendapatkan potensi kebutuhan baru. Lebih detail, peneliti akan menganalisa semua interface yang telah diidentifikasi pada tahapan sebelumnya

yaitu Interface Pilihan Permainan, Interface Pilihan Level SBB, Interface Gameplay SBB, dan Interface Gameplay PP.

### A. Interface Pilihan Permainan

Elemen dinamis pada Interface Pilihan Permainan berdasarkan analisa peneliti terhadap permainan digital P-MCE terdiri atas dua komponen. Pertama, tombol untuk menghidupkan dan mematikan suara BGM seperti yang ditandai dengan lingkaran biru pada Gambar 5.22.



Gambar 5.22. Fungsi untuk mematikan/menghidupkan BGM

Kedua, dalam menampilkan peraturan penumpang, sistem akan menampilkan satu-persatu peraturan secara slideshow dan dikontrol dengan tombol navigasi seperti pada Gambar 5.23 yang ditandai dengan warna biru.



Gambar 5.23. Tombol navigasi pada peraturan penumpang

Berdasarkan hasil analisa tersebut, peneliti mengupdate kebutuhan dan tugas pada Interface Pilihan Permainan seperti yang disajikan pada Tabel 5.23.

Tabel 5.23. Update kebutuhan Pilihan Permainan (Dynamic)

No	Daftar kebutuhan (fitur)	Tugas
1	Berpindah ke Interface Pilihan Level SBB	Membuat fungsi untuk memanggil interface Pilihan Level SBB
2	Berpindah ke Interface Gameplay PP	Membuat fungsi untuk memanggil interface Gameplay PP
3	Menampilkan contoh barang bawaan berdasarkan kategori	Membuat fungsi untuk menampilkan aset -aset barang berdasarkan kategori
4	Menampilkan contoh perilaku penumpang yang harus ditaati.	Membuat fungsi untuk menampilkan peraturan perilaku penumpang.
5	Menampilkan peraturan perilaku penumpang satu persatu dan dapat berubah dengan menggunakan tombol navigasi	Membuat fungsi untuk menampilkan peraturan perilaku penumpang satu-persatu dan fungsi navigasinya.
6	Mematikan dan menghidupkan suara BGM	Membuat fungsi untuk menyalakan dan mematikan efek BGM pada permainan
<b>Keterangan</b>		
 : Kebutuhan / tugas yang didapat pada tahapan Mechanic  : Kebutuhan / tugas yang didapat pada tahapan Dynamic		

## B. Interface Pilihan Level SBB

Dalam interface Pilihan Level terdapat elemen dinamis yang ditemukan peneliti yaitu berupa penampilan tutorial permainan SBB. Tutorial ditampilkan dalam bentuk *slideshow* seperti yang diilustrasikan pada Gambar 5.24 dengan tombol kiri dan kanan untuk menavigasi teks tutorial ditandai dengan area berwarna biru.



Gambar 5.24. Tombol navigasi tutorial

Selain itu, dalam interface ini juga terdapat fungsi untuk menampilkan ilustrasi bandara berdasarkan tingkatan level yang dipilih. Berdasarkan penemuan tersebut, peneliti mengupdate daftar kebutuhan Interface Pilihan Level SBB seperti yang disajikan pada Tabel 5.24.

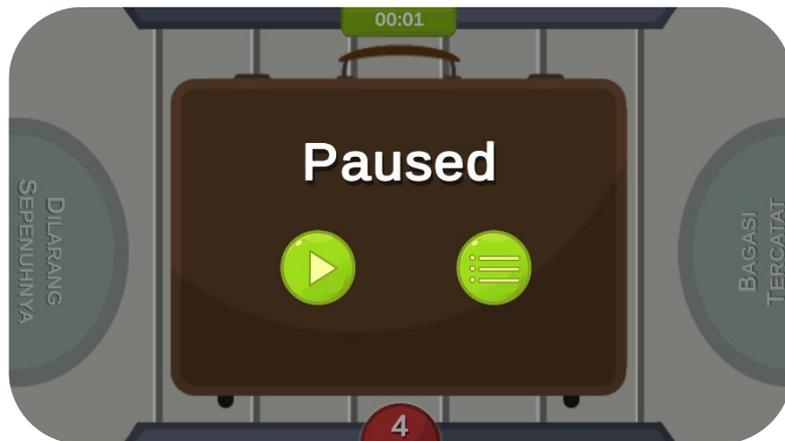
Tabel 5.24. Update kebutuhan Pilihan Level SBB (Dynamic)

No	Daftar kebutuhan (fitur)	Tugas
1	Berpindah ke interface Pilihan Permainan	Membuat fungsi untuk memanggil interface Pilihan Permainan
2	Berpindah ke Interface Gameplay SBB sesuai pilihan level	Membuat fungsi untuk memanggil interface Gameplay SBB dengan mengirimkan informasi level yang dipilih

No	Daftar kebutuhan (fitur)	Tugas
3	Pemain dapat memainkan level tertentu apabila level sebelumnya telah diselesaikan	Membuat fungsi untuk mengunci dan membuka level.
4	Setiap tingkatan level memiliki catatan waktu terbaik	Membuat fungsi untuk menampilkan waktu terbaik pada setiap tingkatan level.
5	Pemain dapat melihat dan menutup tutorial mode permainan SBB	Membuat fungsi untuk menampilkan dan menutup jendela tutorial SBB
6	tutorial berbentuk barisan teks dan ditampilkan secara bertahap dengan bantuan tombol navigasi untuk menavigasi teks tutorial	Membuat fungsi navigasi untuk tutorial SBB
7	Pemain dapat melihat keterangan bandara pada setiap tingkatan level	Membuat fungsi untuk menampilkan keterangan bandara pada setiap tingkatan level.
<b>Keterangan</b>		
 : Kebutuhan / tugas yang didapat pada tahapan Mechanic  : Kebutuhan / tugas yang didapat pada tahapan Dynamic		

### C. Interface Gameplay SBB

Dalam interface Gameplay mode Seleksi Barang Bawaan atau SBB pada permainan digital P-MCE, elemen dinamis yang ditemukan oleh peneliti adalah kemampuan sistem untuk menghentikan sementara dan melanjutkan kembali permainan. Fitur pause diilustrasikan seperti pada Gambar 5.25.



Gambar 5.25. Fitur pause di interface Gameplay SBB

Fitur pause dan resume ini menjadi penting untuk permainan digital P-MCE dikarenakan pada mode SBB berkaitan dengan pencapaian waktu terbaik. Oleh karena itu, fitur pause menjadi sangat penting supaya waktu permainan tidak terus berjalan secara tidak sengaja apabila pemain meninggalkan permainan sementara. Berdasarkan deskripsi tersebut, daftar kebutuhan pada interface Gameplay SBB akan bertambah dan menjadi seperti pada Tabel 5.25.

Tabel 5.25. Update kebutuhan Gameplay SBB (Dynamic)

No	Daftar kebutuhan (fitur)	Tugas
1	Menginisialisasi dan menjalankan timer permainan	Membuat fungsi untuk menampilkan timer dan menjalankan timer
2	Memunculkan barang berlabel legal, terlarang, dan bagasi secara random dengan komposisi tertentu	Membuat fungsi untuk menampilkan barang secara random. Menginisiasi label pada barang
3	Kemampuan drag & drop pada masing-masing barang	Membuat fungsi <i>drag&amp;drop</i> dan meng-attach fungsi tersebut pada setiap barang

No	Daftar kebutuhan (fitur)	Tugas
4	Membuat area drop barang berlabel terlarang, dan bagasi	Membuat fungsi untuk menginisiasi area drop berlabel.
5	Terdapat inisialisasi jumlah target barang yang harus di filter setiap levelnya	Menginisialisasi nilai jumlah target barang.
6	Barang yang berlabel sama dengan area drop dapat di drop pada area drop.	Membuat fungsi pengecekan untuk mendeteksi kesesuaian label pada barang dan area drop
7	Mengurangi nilai jumlah barang apabila barang telah di drop pada area yang sesuai	Membuat fungsi untuk mengurangi nilai pada jumlah target barang
8	Apabila jumlah targer barang = 0, maka sistem akan mencatat waktu timer	Membuat fungsi untuk mencatat waktu timer saat nilai jumlah target barang = 0.
9	Mengupdate catatan waktu terbaik	Membuat fungsi untuk membandingkan catatan waktu yg tersimpan dengan catatan waktu permainan. Apabila catatan waktu lebih cepat dari waktu yang tersimpan maka update catatan waktu, dan tampilkan “Anda mencatat waktu terbaik”, apabila tidak maka tampilkan waktu terbaik dari waktu yang tersimpan
10	Pemain dapat mengulang permainan pada level yang dimainkannya	Membuat fungsi untuk mengulang permainan

No	Daftar kebutuhan (fitur)	Tugas
11	Pemain dapat kembali ke tampilan pilihan level	Membuat fungsi untuk memanggil interface Pilihan Level SBB
12	Pemain dapat menghentikan sementara dan melanjutkan permainan dan timer	Membuat fungsi untuk menghentikan sementara dan melanjutkan permainan.
<b>Keterangan</b>		
 : Kebutuhan / tugas yang didapat pada tahapan Mechanic  : Kebutuhan / tugas yang didapat pada tahapan Dynamic		

#### D. Interface Gameplay PP

Interface Gameplay mode Perilaku Penumpang (PP) memiliki elemen dinamis yaitu tutorial permainan mode PP yang memiliki cara kerja sama dengan tutorial permainan SBB pada Interface Pilihan Level SBB. Permainan mode PP tidak memiliki tingkatan level seperti permainan mode SBB sehingga tutorial permainan PP diimplementasikan kedalam Gameplay PP.

Berdasarkan hasil penemuan elemen dinamis pada interface Gameplay PP, daftar kebutuhan pada interface ini juga akan bertambah seperti yang ditampilkan pada Tabel 5.26.

Tabel 5.26. Update kebutuhan Gameplay PP (Dynamic)

No	Daftar kebutuhan (fitur)	Tugas
1	Menampilkan aset – aset yang digunakan sebagai persyaratan perilaku penumpang pada posisi tertentu	Membuat fungsi untuk menampilkan aset pada koordinat tertentu

No	Daftar kebutuhan (fitur)	Tugas
2	Setiap asset dapat berinteraksi dengan pemain apabila disentuh ( <i>touch</i> ) dengan menampilkan opsi pilihan	Membuat fungsi interaksi touch pada setiap aset dengan hasil interaksi berupa opsi pilihan
3	Setiap pilihan opsi yang benar akan mengakibatkan satu persyaratan terpenuhi dan merubah tampilan permainan	Membuat fungsi untuk hide dan unhide pada asset sebagai akibat dari pilihan opsi yang benar
4	Menampilkan instruksi persyaratan apa yang belum terpenuhi apabila pemain menekan tombol "OK" dan membuka tampilan kemenangan apabila semua persyaratan telah terpenuhi.	Membuat fungsi untuk pengecekan persyaratan pada tombol "OK"
5	Pemain dapat mengulang kembali permainan PP	Membuat fungsi untuk mengulang permainan PP
6	Pemain dapat kembali ke tampilan pilihan mode permainan	Membuat fungsi untuk memanggil interface Pilihan Permainan.
7	Pemain dapat melihat dan menutup tutorial mode permainan PP	Membuat fungsi untuk menampilkan dan menutup jendela tutorial PP
8	tutorial berbentuk barisan teks dan ditampilkan secara bertahap dengan bantuan tombol navigasi untuk menavigasi teks tutorial	Membuat fungsi navigasi untuk tutorial PP

Keterangan	
	: Kebutuhan / tugas yang didapat pada tahapan Mechanic
	: Kebutuhan / tugas yang didapat pada tahapan Dynamic

#### 5.1.2.4. *Aesthetic* proyek

Sebagai perangkat lunak bertipe permainan digital, visualisasi menjadi bobot utama karena berperan sebagai faktor untuk menarik minat pemain. Proyek P-MCE tidak terlepas terhadap hal tersebut. Visualisasi yang menjadi perhatian utama dalam proyek P-MCE adalah aset barang-barang yang digunakan pada mode permainan SBB dan juga item-item interaksi yang digunakan pada mode permainan PP. Untuk lebih jelasnya, peneliti akan menganalisa keseluruhan interface yang ada pada permainan digital P-MCE yaitu Interface Pilihan permainan, Pilihan Level SBB, Gameplay SBB, dan Gameplay PP untuk mendapatkan daftar kebutuhan baru yang berkaitan dengan elemen *Aesthetic*.

##### A. **Interface Pilihan Permainan**

Elemen *Aesthetic* yang nampak pada interface Pilihan level lebih condong pada visualisasi dan animasi. Elemen visual yang terdapat pada interface ini meliputi:

- 1) Background berilustrasi langit
- 2) Logo permainan digital
- 3) Desain tombol untuk memvisualisasikan fungsi utama permainan
- 4) Frame sebagai tempat untuk menyusun tombol

Elemen yang telah disebutkan peneliti tersebut dapat dilihat pada Gambar 5.26 dengan rincian area yang berwarna “jingga” merupakan ilustrasi background, “merah” merupakan desain logo permainan digital, “kuning” merupakan desain tombol, dan warna “biru” merupakan frame untuk menyusun komposisi tombol-tombol.



Gambar 5.26. Elemen visual interface Pilihan Permainan

Selain itu, pada interface ini juga memiliki elemen visual pada dokumentasi contoh barang-barang kategori bagasi dan dilarang dan visualisasi perilaku penumpang yang harus dihindari beserta akibatnya jika tidak dihindari. Berdasarkan hasil analisa elemen visual pada interface Pilihan Permainan maka daftar kebutuhan akan berubah menjadi seperti Tabel 5.27.

Tabel 5.27. Update kebutuhan Pilihan Permainan (Aesthetic)

No	Daftar kebutuhan (fitur)	Tugas
1	Berpindah ke Interface Pilihan Level SBB	Membuat fungsi untuk memanggil interface Pilihan Level SBB
		Menciptakan desain tombol sebagai tampilan visual untuk fungsi memanggil interface Pilihan Level SBB
2	Berpindah ke Interface Gameplay PP	Membuat fungsi untuk memanggil interface Gameplay PP

No	Daftar kebutuhan (fitur)	Tugas
		Menciptakan desain tombol sebagai tampilan visual untuk fungsi memanggil interface Gameplay PP
3	Menampilkan contoh barang bawaan berdasarkan kategori	Membuat fungsi untuk menampilkan aset barang-barang berdasarkan kategori
		Menciptakan desain tombol sebagai tampilan visual untuk fungsi menampilkan barang-barang berdasarkan kategori
		Mengonsep tata letak aset barang pada setiap kategori
4	Menampilkan contoh perilaku penumpang yang harus ditaati.	Membuat fungsi untuk menampilkan peraturan perilaku penumpang.
		Membuat desain ilustrasi tombol sebagai tampilan visual untuk fungsi menampilkan contoh perilaku penumpang
5	Menampilkan peraturan perilaku penumpang satu persatu dan dapat berubah dengan menggunakan tombol navigasi	Membuat fungsi untuk menampilkan peraturan perilaku penumpang satu-persatu dan fungsi navigasinya.
		Membuat tombol sebagai tampilan visual untuk fungsi navigasi
		Membuat ilustrasi perilaku penumpang dan akibatnya

No	Daftar kebutuhan (fitur)	Tugas
6	Mematikan dan menghidupkan suara BGM	Membuat fungsi untuk menyalakan dan mematikan efek BGM pada permainan
		Menciptakan desain tombol sebagai tampilan visual untuk fungsi mematikan dan menghidupkan efek BGM
7	Membuat tampilan interface Pilihan permainan	Menciptakan desain ilustrasi background
		Menciptakan desain ilustrasi logo permainan digital
		Menciptakan desain frame
		Mengonsep komposisi dan pengaturan tata letak komponen visual
<b>Keterangan</b>		
 <ul style="list-style-type: none"> <li> : Kebutuhan / tugas yang didapat pada tahapan Mechanic</li> <li> : Kebutuhan / tugas yang didapat pada tahapan Dynamic</li> <li> : Kebutuhan / tugas yang didapat pada tahapan Aesthetic</li> <li> : Kebutuhan yang memiliki tugas teknis program dan visual</li> </ul>		

## B. Interface Pilihan Level SBB

Elemen visual yang terdapat pada interface Pilihan Level SBB yang ditampilkan pada Gambar 5.27 adalah sebagai berikut:

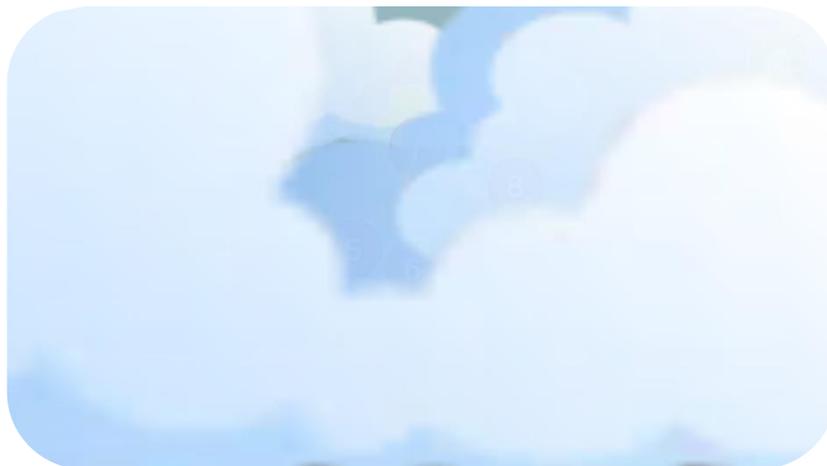
- 1) Background ilustrasi peta Indonesia
- 2) Desain frame

- 3) Desain tombol sebagai tampilan visual untuk fungsi fungsi yang ada pada interface Pilihan Level SBB antara lain fungsi untuk mengunci dan membuka tingkatan level, memainkan tingkatan level tertentu, menampilkan tutorial, dan kembali ke pilihan permainan.



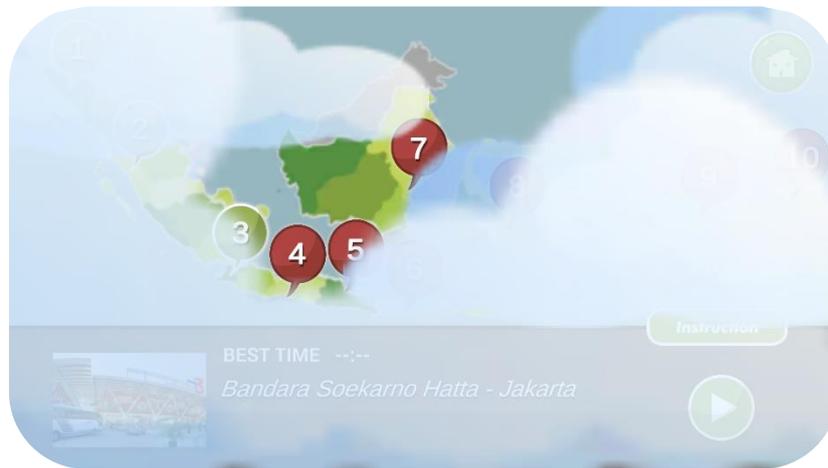
Gambar 5.27. Elemen visual interface Pilihan Level SBB

Pada interface ini juga terdapat elemen animasi saat hendak menampilkan peta pilihan level dengan pergerakan awan yang awalnya menutup tampilan pilhan level seperti Gambar 5.28.



Gambar 5.28. Tampilan awal animasi interface Pilihan Level

Secara perlahan awan tersebut akan bergerak dan menampilkan tampilan Pilihan. Gambar 5.29 mengilustrasikan pergerakan awan yang secara perlahan.



Gambar 5.29. Animasi pada objek awan

Awan tersebut akan bergerak dan menghilang sehingga menampilkan visual pilihan level secara jelas seperti Gambar 5.27. Berdasarkan penjelasan mengenai elemen visual dan animasi diatas daftar kebutuhan pada interface Pilihan Level SBB berubah dan bertambah menjadi seperti pada Tabel 5.28.

Tabel 5.28. Update kebutuhan Pilihan Level SBB (Aesthetic)

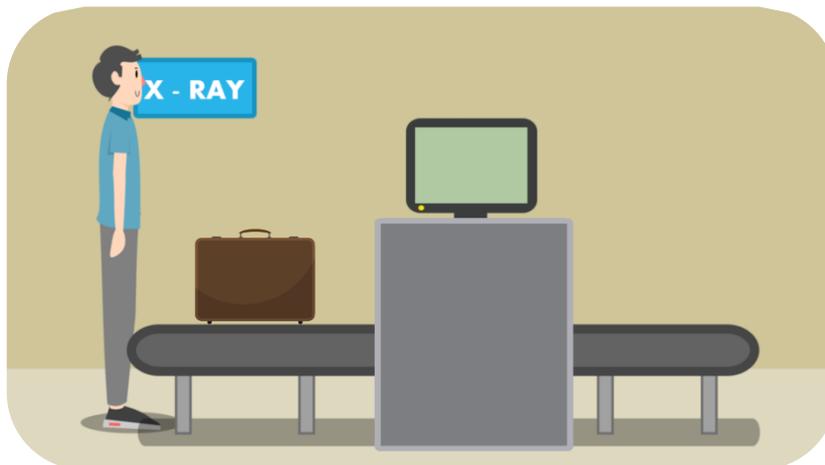
No	Daftar kebutuhan (fitur)	Tugas
1	Berpindah ke interface Pilihan Permainan	Membuat fungsi untuk memanggil interface Pilihan Permainan
		Menciptakan desain tombol sebagai tampilan visual untuk fungsi memanggil interface Pilihan Permainan
2	Berpindah ke Interface Gameplay SBB sesuai pilihan level	Membuat fungsi untuk memanggil interface Gameplay SBB dengan

No	Daftar kebutuhan (fitur)	Tugas
		mengirimkan informasi level yang dipilih
		Menciptakan desain tombol untuk menuju interface Gameplay SBB (dua kondisi terkunci atau terbuka)
3	Pemain dapat memainkan level tertentu apabila level sebelumnya telah diselesaikan	Membuat fungsi untuk mengunci dan membuka level.
		Menciptakan tombol level dengan dua desain (terbuka dan terkunci)
4	Setiap tingkatan level memiliki catatan waktu terbaik	Membuat fungsi untuk menampilkan waktu terbaik pada setiap tingkatan level.
5	Pemain dapat melihat dan menutup tutorial mode permainan SBB	Membuat fungsi untuk menampilkan dan menutup jendela tutorial SBB
		Menciptakan desain tutorial
6	tutorial berbentuk barisan teks dan ditampilkan secara bertahap dengan bantuan tombol navigasi untuk menavigasi teks tutorial	Membuat fungsi navigasi untuk tutorial SBB
		Membuat desain tombol sebagai tampilan untuk fungsi navigasi.
7	Pemain dapat melihat keterangan bandara pada setiap tingkatan level	Membuat fungsi untuk menampilkan keterangan bandara pada setiap tingkatan level.
		Membuat ilustrasi bandara untuk setiap level
8		Mengonsep tata letak visual

No	Daftar kebutuhan (fitur)	Tugas
	Memiliki tampilan Pilihan Level bertema indonesia	Menciptakan desain ilustrasi background peta indonesia
		Menciptakan desain frame
		Menciptakan desain awan
		Menciptakan dan animasi terhadap objek awan
Keterangan		
	 : Kebutuhan / tugas yang didapat pada tahapan Mechanic  : Kebutuhan / tugas yang didapat pada tahapan Dynamic  : Kebutuhan / tugas yang didapat pada tahapan Aesthetic  : Kebutuhan yang memiliki tugas teknis program dan visual	

### C. Interface Gameplay SBB

Elemen aesthetic dalam interface Gameplay SBB didominasi di sektor visual dan animasi. Dimulai pada fase awal permainan (*start*), pada fase ini terdapat animasi pemain yang memasukkan koper kedalam mesin x-ray seperti yang diilustrasikan pada Gambar 5.30.



Gambar 5.30. Animasi awal permainan SBB

Selanjutnya, pemain akan dihadapkan pada tampilan permainan seperti pada Gambar 5.31.

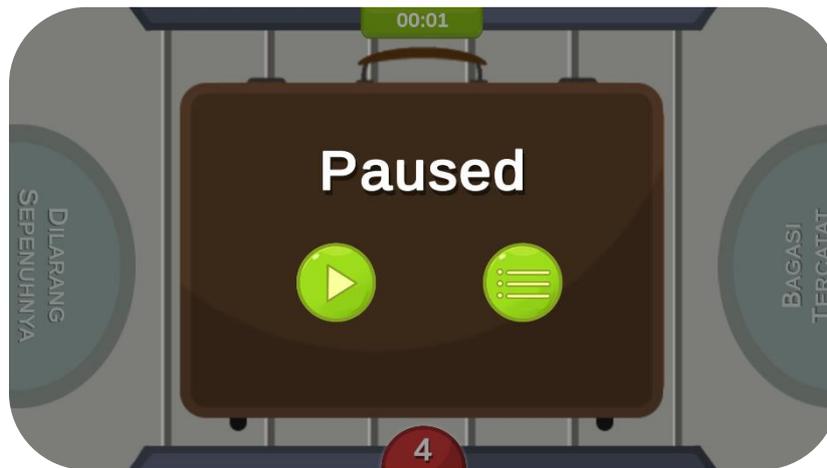


Gambar 5.31. Tampilan saat permainan SBB berjalan

Pada tampilan permainan yang diilustrasikan pada Gambar 5.31 tersebut, peneliti mendapatkan komponen visual yang terdiri atas:

- 1) Ilustrasi background berupa konveyor mesin x-ray
- 2) Ilustrasi desain koper
- 3) Ilustrasi frame untuk indikator waktu
- 4) Ilustrasi tombol untuk fungsi *pause*
- 5) Ilustrasi alarm untuk indikator jumlah barang yang harus difilter.
- 6) Ilustrasi area untuk drop barang (dilarang sepenuhnya dan bagasi tercatat)
- 7) Ilustrasi untuk aset barang-barang yang digunakan pada permainan SBB.
- 8) Animasi *swing* (rotasi kekiri dan kekanan) pada semua barang.

Selanjutnya, apabila pemain menjalankan fungsi menghentikan sementara (*pause*) dengan menekan tombol *pause*, maka tampilan permainan akan menjadi seperti Gambar 5.32.



Gambar 5.32. Tampilan saat permainan SBB berhenti sementara

Berdasarkan gambar 5.32 tersebut, elemen visual yang terlihat terdiri atas:

- 1) Label “paused” untuk menandakan kondisi permainan dalam keadaan berhenti
- 2) Desain tombol untuk mewakili fungsi melanjutkan kembali permainan (*resume*)
- 3) Desain tombol untuk mewakili fungsi kembali ke interface Pilihan Level SBB.

Terakhir, apabila pemain telah berhasil melakukan filter terhadap barang yang ada di dalam koper dengan cara drag&drop barang ke kategori yang sesuai (dilarang sepenuhnya atau bagasi tercatat) hingga jumlah barang yang harus difilter menjadi 0, maka tampilan permainan akan menjadi seperti Gambar 5.33 yang menandakan pemain telah menyelesaikan permainan di tingkatan level tersebut.



Gambar 5.33. Tampilan saat permainan SBB selesai

Elemen visual yang terdapat pada Gambar 5.33 terdiri atas:

- 1) Label “Selesai” yang menandakan pemain telah menyelesaikan permainan
- 2) Desain alarm berwarna hijau
- 3) Desain sinar berwarna hijau
- 4) Desain tombol yang mewakili fungsi untuk mengulang permainan
- 5) Desain tombol yang mewakili fungsi untuk kembali ke Pilihan Level SBB.

Berdasarkan uraian diatas, peneliti mengupdate daftar kebutuhan pada interface Gameplay SBB seperti yang disajikan pada Tabel 5.29.

Tabel 5.29. Update kebutuhan Gameplay SBB (Aesthetic)

No	Daftar kebutuhan (fitur)	Tugas
1	Menginisialisasi dan menjalankan timer permainan	Membuat fungsi untuk menampilkan timer dan menjalankan timer
		Membuat desain frame untuk tempat indikator timer

No	Daftar kebutuhan (fitur)	Tugas
2	Memunculkan barang berlabel legal, terlarang, dan bagasi secara random dengan komposisi tertentu	Membuat fungsi untuk menampilkan barang secara random.
		Menginisiasi label pada barang
		Membuat desain setiap aset barang
		Menciptakan animasi <i>swing</i> pada setiap barang
3	Kemampuan drag & drop pada masing-masing barang	Membuat fungsi <i>drag&amp;drop</i> dan meng-attach fungsi tersebut pada setiap barang
4	Membuat area drop barang berlabel (dilarang sepenuhnya dan bagasi tercatat)	Membuat fungsi untuk menginisiasi area drop berlabel.
		Membuat desain area drop barang
5	Terdapat inisialisasi jumlah target barang yang harus di filter setiap levelnya	Menginisialisasi nilai jumlah target barang.
		Membuat desain alarm berwarna merah yang menandakan koper berisi barang berbahaya
6	Barang yang berlabel sama dengan area drop dapat di drop pada area drop.	Membuat fungsi pengecekan untuk mendeteksi kesesuaian label pada barang dan area drop
7	Mengurangi nilai jumlah barang apabila barang telah di drop pada area yang sesuai	Membuat fungsi untuk mengurangi nilai pada jumlah target barang
8	Apabila jumlah targer barang = 0, maka sistem akan mencatat waktu timer	Membuat fungsi untuk mencatat waktu timer saat nilai jumlah target barang = 0.

No	Daftar kebutuhan (fitur)	Tugas
9	Mengupdate catatan waktu terbaik	Membuat fungsi untuk membandingkan catatan waktu yg tersimpan dengan catatan waktu permainan. Apabila catatan waktu lebih cepat dari waktu yang tersimpan maka update catatan waktu, dan tampilkan “Anda mencatat waktu terbaik”, apabila tidak maka tampilkan waktu terbaik dari waktu yang tersimpan
		Mendesain tampilan saat permainan berhasil dimenangkan
10	Pemain dapat mengulang permainan pada level yang dimainkannya	Membuat fungsi untuk mengulang permainan
		Mendesain ilustrasi tombol yang berfungsi untuk mengulang permainan
11	Pemain dapat kembali ke tampilan pilihan level	Membuat fungsi untuk memanggil interface Pilihan Level SBB
		Mendesain ilustrasi tombol yang berfungsi untuk memanggil interface Pilihan Level SBB
12	Pemain dapat menghentikan sementara dan melanjutkan permainan dan timer	Membuat fungsi untuk menghentikan sementara dan melanjutkan permainan.

No	Daftar kebutuhan (fitur)	Tugas
		Mendesain ilustrasi tombol yang berfungsi untuk menghentikan dan melanjutkan permainan
		Mendesain kondisi tampilan saat permainan berhenti
13	Tampilan permainan SBB seperti kondisi koper dalam mesin x-ray	Ilustrasi desain koper
		Animasi koper memasuki mesin x-ray
		Ilustrasi background berupa konveyor mesin x-ray
<b>Keterangan</b>		
 <ul style="list-style-type: none"> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: yellow; margin-right: 5px;"></span> : Kebutuhan / tugas yang didapat pada tahapan Mechanic</li> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: blue; margin-right: 5px;"></span> : Kebutuhan / tugas yang didapat pada tahapan Dynamic</li> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: green; margin-right: 5px;"></span> : Kebutuhan / tugas yang didapat pada tahapan Aesthetic</li> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: orange; margin-right: 5px;"></span> : Kebutuhan yang memiliki tugas teknis program dan visual</li> </ul>		

#### D. Interface Gameplay PP

Tujuan dari permainan PP ini adalah untuk menyiapkan seorang penumpang yang berada dalam pesawat saat kondisi akan lepas landas. Apabila pemain telah yakin kondisi penumpang telah siap dan sesuai dengan prosedur penumpang saat lepas landas, pemain selanjutnya menekan tombol OK untuk menyelesaikan permainan. Namun apabila kondisi penumpang belum sepenuhnya sesuai, maka akan terdapat peringatan dari pramugari tentang prosedur yang belum dilakukan oleh penumpang.

Untuk fase awal (start) permainan, kondisi Gameplay PP memiliki tampilan visual yang diilustrasikan pada Gambar 5.33



Gambar 5.34. Tampilan fase start Gameplay PP

Tampilan instruksi awal ini adalah sebagai petunjuk untuk pemain kira-kira aset visual apa yang nantinya dapat berinteraksi dengan pemain. Fase awal permainan Gameplay PP ini memiliki elemen visual yang terdiri atas:

- 1) Ilustrasi pramugari
- 2) Ilustrasi frame sebagai tempat untuk teks keterangan
- 3) Ilustrasi tombol yang berfungsi untuk memulai permainan.

Selanjutnya, apabila pemain menekan tombol *play* (warna hijau) maka pemain akan dibawa menuju tampilan permainan seperti pada Gambar 5.34.



Gambar 5.35. Fase pertama Gameplay PP

Tampilan Gambar 5.34 merupakan fase pertama yang hanya memiliki satu aset yang dapat berinteraksi dengan pemain apabila pemain menyentuhnya. Aset tersebut adalah penumpang dalam kondisi berdiri. Apabila pemain menekan tombol “OK” saat penumpang masih dalam kondisi berdiri maka akan ditampilkan warning seperti pada Gambar 5.35 yang menandakan persyaratan untuk fase pertama permainan PP belum terpenuhi.



Gambar 5.36. Tampilan peringatan permainan PP

Interaksi touch pada penumpang berdiri ini juga telah memenuhi persyaratan pertama dan aset penumpang berdiri akan digantikan dengan set penumpang duduk seperti Gambar 5.36 dan dimulailah fase kedua permainan PP.



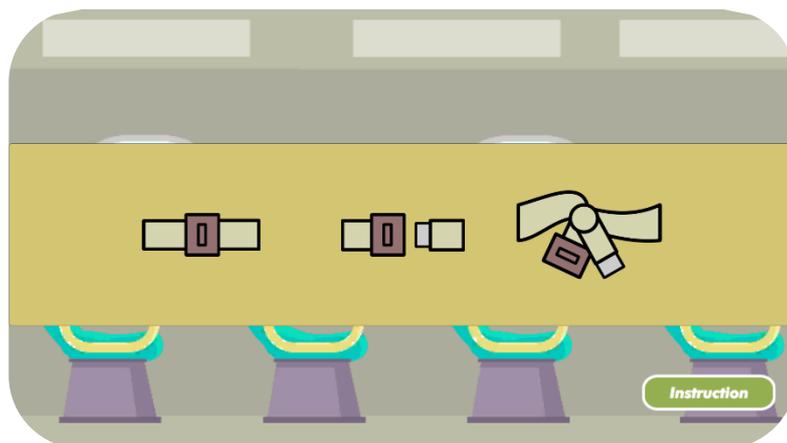
Gambar 5.37. Fase kedua Gameplay PP

Dalam fase kedua ini, terdapat tiga interaksi yang harus dilakukan oleh pemain yaitu menutup meja, mematikan smartphone dan mengencangkan sabuk pengaman. Setiap interaksi ada yang memiliki opsi ada yang dieksekusi secara langsung. Sebagai contoh interaksi dengan meja menghasilkan meja akan ditutup seperti Gambar 5.37.



Gambar 5.38. Hasil interaksi dengan meja lipat

Contoh interaksi yang memiliki opsi adalah interaksi pemain dengan sabuk pengaman. Apabila pemain melakukan touch kepada sabuk pengaman maka pilihan opsi seperti Gambar 5.39 akan muncul.



Gambar 5.39. Pilihan opsi untuk sabuk pengaman

Jika opsi yang dipilih pemain benar, maka tampilan akan berubah seperti Gambar 5.38. Apabila pemain memilih opsi yang salah maka tampilan tidak akan berubah.



Gambar 5.40. Tampilan permainan saat sabuk pengaman benar

Apabila pemain telah menyelesaikan semua persyaratan kondisi penumpang saat pesawat akan lepas landas dan menekan tombol “OK” maka tampilan kemenangan akan muncul. Tampilan kemenangan pada permainan ini berupa instruksi lebih lanjut mengenai keselamatan penerbangan serta memiliki opsi untuk mengulang permainan dan kembali ke pilihan permainan. Gambar 5.39 merupakan ilustrasi tampilan kemenangan pada Gameplay PP.

Berdasarkan penjelasan tersebut dapat dipahami bahwa elemen visual berperan aktif dalam permainan ini dan terdapat beberapa aset memiliki tampilan kondisi yang berbeda. Pentingnya elemen visual mengakibatkan daftar kebutuhan pada interface Gameplay PP menjadi berubah seperti pada Tabel 5.30.

Tabel 5.30. Update kebutuhan Gameplay PP (Aesthetic)

No	Daftar kebutuhan (fitur)	Tugas
1	Menampilkan aset – aset yang digunakan sebagai persyaratan perilaku	Membuat fungsi untuk menampilkan aset pada koordinat tertentu

No	Daftar kebutuhan (fitur)	Tugas
	penumpang pada posisi tertentu	Merancang komposisi aset-aset yang digunakan, antara lain (kursi, background interior pesawat, penumpang, meja, sabuk pengaman, smartphone)
2	Setiap asset dapat berinteraksi dengan pemain apabila disentuh ( <i>touch</i> ) dengan menampilkan opsi pilihan	Membuat fungsi interaksi touch pada setiap aset dengan hasil interaksi berupa opsi pilihan
		Mendesain ilustrasi setiap opsi pilihan
3	Setiap pilihan opsi yang benar akan mengakibatkan satu persyaratan terpenuhi dan merubah tampilan permainan	Membuat fungsi untuk hide dan unhide pada asset sebagai akibat dari pilihan opsi yang benar
		Mendesain asset dengan dua kondisi yang berbeda, seperti penumpang dengan kondisi berdiri dan duduk dan penumpang dengan posisi memegang HP dengan posisi duduk sempurna.
4	Menampilkan instruksi persyaratan apa yang belum terpenuhi apabila pemain menekan tombol “OK” dan membuka tampilan kemenangan apabila semua persyaratan telah terpenuhi.	Membuat fungsi untuk pengecekan persyaratan pada tombol “OK”
		Mendesain ilustrasi tombol untuk pengecekan persyaratan penumpang dengan label “OK”
		Mendesain tampilan kemenangan Gameplay PP
5	Pemain dapat mengulang kembali permainan PP	Membuat fungsi untuk mengulang permainan PP

No	Daftar kebutuhan (fitur)	Tugas
		Mendesain ilustrasi tombol yang berfungsi untuk mengulang permainan
6	Pemain dapat kembali ke tampilan pilihan mode permainan	Membuat fungsi untuk memanggil interface Pilihan Permainan.
		Mendesain ilustrasi tombol yang berfungsi memanggil interface Pilihan Permainan
7	Pemain dapat melihat dan menutup tutorial mode permainan PP	Membuat fungsi untuk menampilkan dan menutup jendela tutorial PP
		Mendesain tampilan tutorial PP
8	Tutorial berbentuk barisan teks dan ditampilkan secara bertahap dengan bantuan tombol navigasi untuk menavigasi teks tutorial	Membuat fungsi navigasi untuk tutorial PP
		Mendesain ilustrasi tombol yang berfungsi untuk navigasi tutorial PP
9	Memiliki Tampilan awal permainan yang menjelaskan petunjuk permainan PP	Mendesain tampilan awal permainan
<b>Keterangan</b>		
 : Kebutuhan / tugas yang didapat pada tahapan Mechanic  : Kebutuhan / tugas yang didapat pada tahapan Dynamic  : Kebutuhan / tugas yang didapat pada tahapan Aesthetic  : Kebutuhan yang memiliki tugas teknis program dan visual		

#### 5.1.2.5. *Experience* proyek

Proyek P-MCE menerapkan tampilan permainan yang memanfaatkan sisi horizontal layar sehingga menggunakan konsep pendekatan permainan yang berbasis horizontal (contoh Gambar 5.15). Hal tersebut dikarenakan proyek P-MCE menggunakan interaksi bertipe *touch* dan *drag&drop*. Kedua interaksi tersebut memerlukan ruang yang luas dan pendekatan horizontal merupakan yang terbaik karena pemain dapat dengan leluasa untuk memilih objek tanpa takut terkena objek lain yang tidak sesuai keinginan pemain.

#### 5.1.2.6. *Assumption and Constrain* proyek

Konsep permainan yang menggunakan interaksi *touch* dan *drag&drop* memiliki beberapa hal yang harus diperhatikan oleh tim pengembang yaitu:

- 1) Delay saat melakan fungsi drag pada objek perlu diperhatikan karena apabila delay terlalu besar maka terdapat kemungkinan objek akan terlepas dari tangan pemain.
- 2) Ukuran objek untuk fungsi *drag&drop* maupun fungsi *touch* tidak boleh terlalu kecil karena akan mempengaruhi kedua fungsi tersebut untuk mendeteksi sentuhan tangan pemain terhadap benda.

#### 5.1.2.7. Daftar kebutuhan final proyek P-MCE

Bedasarkan langkah-langkah analisis komponen mechanic, dynamic, aesthetic, experience, dan assumption and constrain pada proyek permainan digital P-MCE. Peneliti mendapatkan daftar kebutuhan final yang akan digunakan sebagai masukan ditahap selanjutnya. Daftar kebutuhan dan tugas final pada proyek P-MCE disajikan pada Tabel 5.31 berikut.

Tabel 5.31. Daftar kebutuhan final proyek P-MCE

No	Daftar kebutuhan (fitur)	Tugas
<i>Interface Pilihan Permainan</i>		
1	Berpindah ke Interface Pilihan Level SBB	Membuat fungsi untuk memanggil interface Pilihan Level SBB

No	Daftar kebutuhan (fitur)	Tugas
		Menciptakan desain tombol sebagai tampilan visual untuk fungsi memanggil interface Pilihan Level SBB
2	Berpindah ke Interface Gameplay PP	Membuat fungsi untuk memanggil interface Gameplay PP
		Menciptakan desain tombol sebagai tampilan visual untuk fungsi memanggil interface Gameplay PP
3	Menampilkan contoh barang bawaan berdasarkan kategori	Membuat fungsi untuk menampilkan aset barang-barang berdasarkan kategori
		Menciptakan desain tombol sebagai tampilan visual untuk fungsi menampilkan barang-barang berdasarkan kategori
		Mengonsep tata letak aset barang pada setiap kategori
4	Menampilkan contoh perilaku penumpang yang harus ditaati.	Membuat fungsi untuk menampilkan peraturan perilaku penumpang.
		Membuat desain ilustrasi tombol sebagai tampilan visual untuk fungsi menampilkan contoh perilaku penumpang
5	Menampilkan peraturan perilaku penumpang satu persatu dan dapat berubah dengan menggunakan tombol navigasi	Membuat fungsi untuk menampilkan peraturan perilaku penumpang satu-persatu dan fungsi navigasinya.

No	Daftar kebutuhan (fitur)	Tugas
		Membuat tombol sebagai tampilan visual untuk fungsi navigasi
		Membuat ilustrasi perilaku penumpang dan akibatnya
6	Mematikan dan menghidupkan suara BGM	Membuat fungsi untuk menyalakan dan mematikan efek BGM pada permainan
		Menciptakan desain tombol sebagai tampilan visual untuk fungsi mematikan dan menghidupkan efek BGM
7	Membuat tampilan interface Pilihan permainan	Menciptakan desain ilustrasi background
		Menciptakan desain ilustrasi logo permainan digital
		Menciptakan desain frame
		Mengonsep komposisi dan pengaturan tata letak komponen visual
<b><i>Interface Pilihan Level SBB</i></b>		
8	Berpindah ke interface Pilihan Permainan	Membuat fungsi untuk memanggil interface Pilihan Permainan
		Menciptakan desain tombol sebagai tampilan visual untuk fungsi memanggil interface Pilihan Permainan

No	Daftar kebutuhan (fitur)	Tugas
9	Berpindah ke Interface Gameplay SBB sesuai pilihan level.	Membuat fungsi untuk memanggil interface Gameplay SBB dengan mengirimkan informasi level yang dipilih
		Menciptakan desain tombol untuk menuju interface Gameplay SBB (dua kondisi terkunci atau terbuka)
10	Pemain dapat memainkan level tertentu apabila level sebelumnya telah diselesaikan	Membuat fungsi untuk mengunci dan membuka level.
		Menciptakan tombol level dengan dua desain (terbuka dan terkunci)
11	Setiap tingkatan level memiliki catatan waktu terbaik	Membuat fungsi untuk menampilkan waktu terbaik pada setiap tingkatan level.
12	Pemain dapat melihat dan menutup tutorial mode permainan SBB	Membuat fungsi untuk menampilkan dan menutup jendela tutorial SBB
		Menciptakan desain tutorial
13	Tutorial berbentuk barisan teks dan ditampilkan secara bertahap dengan bantuan tombol navigasi untuk menavigasi teks tutorial	Membuat fungsi navigasi untuk tutorial SBB
		Membuat desain tombol sebagai tampilan untuk fungsi navigasi.
14	Pemain dapat melihat keterangan bandara pada setiap tingkatan level	Membuat fungsi untuk menampilkan keterangan bandara pada setiap tingkatan level.
		Membuat ilustrasi bandara untuk setiap level
15	Memiliki tampilan Pilihan Level bertema indonesia	Mengonsep tata letak visual

No	Daftar kebutuhan (fitur)	Tugas
		Menciptakan desain ilustrasi background peta indonesia
		Menciptakan desain frame
		Menciptakan desain awan Menciptakan dan animasi terhadap objek awan
<b>Interface Gameplay SBB</b>		
16	Menginisialisasi dan menjalankan timer permainan	Membuat fungsi untuk menampilkan timer dan menjalankan timer
		Membuat desain frame untuk tempat indikator timer
17	Memunculkan barang berlabel legal, terlarang, dan bagasi secara random dengan komposisi tertentu	Membuat fungsi untuk menampilkan barang secara random.
		Menginisiasi label pada barang
		Membuat desain setiap aset barang
		Menciptakan animasi <i>swing</i> pada setiap barang
18	Kemampuan drag & drop pada masing-masing barang	Membuat fungsi <i>drag&amp;drop</i> dan meng-attach fungsi tersebut pada setiap barang
19	Membuat area drop barang berlabel (dilarang sepenuhnya dan bagasi tercatat)	Membuat fungsi untuk menginisiasi area drop berlabel.
		Membuat desain area drop barang
20		Menginisialisasi nilai jumlah target barang.

No	Daftar kebutuhan (fitur)	Tugas
	Terdapat inisialisasi jumlah target barang yang harus di filter setiap levelnya	Membuat desain alarm berwarna merah yang menandakan koper berisi barang berbahaya
21	Barang yang berlabel sama dengan area drop dapat di drop pada area drop.	Membuat fungsi pengecekan untuk mendeteksi kesesuaian label pada barang dan area drop
22	Mengurangi nilai jumlah barang apabila barang telah di drop pada area yang sesuai	Membuat fungsi untuk mengurangi nilai pada jumlah target barang
23	Apabila jumlah targer barang = 0, maka sistem akan mencatat waktu timer	Membuat fungsi untuk mencatat waktu timer saat nilai jumlah target barang = 0.
24	Mengupdate catatan waktu terbaik	Membuat fungsi untuk membandingkan catatan waktu yg tersimpan dengan catatan waktu permainan. Apabila catatan waktu lebih cepat dari waktu yang tersimpan maka update catatan waktu, dan tampilkan “Anda mencatat waktu terbaik”, apabila tidak maka tampilkan waktu terbaik dari waktu yang tersimpan
		Mendesain tampilan saat permainan berhasil dimenangkan
25	Pemain dapat mengulang permainan pada level yang dimainkannya	Membuat fungsi untuk mengulang permainan
		Mendesain ilustrasi tombol yang berfungsi untuk mengulang permainan

No	Daftar kebutuhan (fitur)	Tugas
26	Pemain dapat kembali ke tampilan pilihan level	Membuat fungsi untuk memanggil interface Pilihan Level SBB
		Mendesain ilustrasi tombol yang berfungsi untuk memanggil interface Pilihan Level SBB
27	Pemain dapat menghentikan sementara dan melanjutkan permainan dan timer	Membuat fungsi untuk menghentikan sementara dan melanjutkan permainan.
		Mendesain ilustrasi tombol yang berfungsi untuk menghentikan dan melanjutkan permainan
		Mendesain kondisi tampilan saat permainan berhenti
28	Tampilan permainan SBB seperti kondisi koper dalam mesin x-ray	Ilustrasi desain koper
		Animasi koper memasuki mesin x-ray
		Ilustrasi background berupa konveyor mesin x-ray
<b><i>Interface Gameplay PP</i></b>		
29	Menampilkan aset – aset yang digunakan sebagai persyaratan perilaku penumpang pada posisi tertentu	Membuat fungsi untuk menampilkan aset pada koordinat tertentu
		Merancang komposisi aset-aset yang digunakan, antara lain (kursi, background interior pesawat, penumpang, meja, sabuk pengaman, smartphone)
30	Setiap asset dapat berinteraksi dengan pemain apabila disentuh	Membuat fungsi interaksi touch pada setiap aset dengan hasil interaksi berupa opsi pilihan

No	Daftar kebutuhan (fitur)	Tugas
	<i>(touch)</i> dengan menampilkan opsi pilihan	Mendesain ilustrasi setiap opsi pilihan
31	Setiap pilihan opsi yang benar akan mengakibatkan satu persyaratan terpenuhi dan merubah tampilan permainan	<p>Membuat fungsi untuk hide dan unhide pada asset sebagai akibat dari pilihan opsi yang benar</p> <p>Mendesain asset dengan dua kondisi yang berbeda, seperti penumpang dengan kondisi berdiri dan duduk dan penumpang dengan posisi memegang HP dengan posisi duduk sempurna.</p>
32	Menampilkan instruksi persyaratan apa yang belum terpenuhi apabila pemain menekan tombol “OK” dan membuka tampilan kemenangan apabila semua persyaratan telah terpenuhi.	<p>Membuat fungsi untuk pengecekan persyaratan pada tombol “OK”</p> <p>Mendesain ilustrasi tombol untuk pengecekan persyaratan penumpang dengan label “OK”</p> <p>Mendesain tampilan kemenangan Gameplay PP</p>
33	Pemain dapat mengulang kembali permainan PP	<p>Membuat fungsi untuk mengulang permainan PP</p> <p>Mendesain ilustrasi tombol yang berfungsi untuk mengulang permainan</p>
34	Pemain dapat kembali ke tampilan pilihan mode permainan	<p>Membuat fungsi untuk memanggil interface Pilihan Permainan.</p> <p>Mendesain ilustrasi tombol yang berfungsi memanggil interface Pilihan Permainan</p>

No	Daftar kebutuhan (fitur)	Tugas
35	Pemain dapat melihat dan menutup tutorial mode permainan PP	Membuat fungsi untuk menampilkan dan menutup jendela tutorial PP
		Mendesain tampilan tutorial PP
36	Tutorial berbentuk barisan teks dan ditampilkan secara bertahap dengan bantuan tombol navigasi untuk menavigasi teks tutorial	Membuat fungsi navigasi untuk tutorial PP
		Mendesain ilustrasi tombol yang berfungsi untuk navigasi tutorial PP
37	Memiliki Tampilan awal permainan yang menjelaskan petunjuk permainan PP	Mendesain tampilan awal permainan
<b>Keterangan</b>		
 <ul style="list-style-type: none"> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: yellow; margin-right: 5px;"></span> : Kebutuhan / tugas yang didapat pada tahapan Mechanic</li> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: blue; margin-right: 5px;"></span> : Kebutuhan / tugas yang didapat pada tahapan Dynamic</li> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: green; margin-right: 5px;"></span> : Kebutuhan / tugas yang didapat pada tahapan Aesthetic</li> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: orange; margin-right: 5px;"></span> : Kebutuhan yang memiliki tugas teknis program dan visual</li> </ul>		

Berdasarkan Tabel 5.31, tugas yang berkaitan dengan elemen aesthetic pada proyek P-MCE berjumlah 45 dari total 79 tugas atau sekitar 56.96%. Hal ini menandakan elemen Aesthetic lebih mendominasi pekerjaan keseluruhan untuk membangun perangkat lunak permainan digital P-MCE.

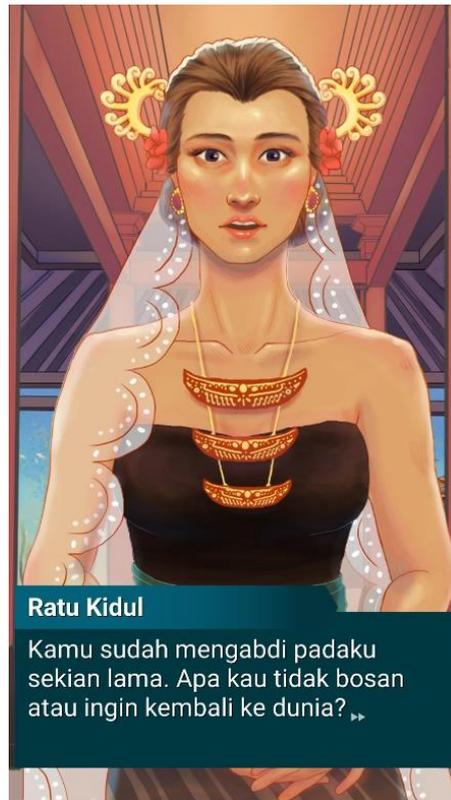
### 5.1.3. Kebutuhan pada proyek P-MVN

Sub bab ini akan menjelaskan proses mendapatkan daftar kebutuhan pada proyek P-MVN.

#### 5.1.3.1. Overview Proyek

Proyek P-MVN merupakan sebuah permainan digital bergenre *visual novel*. Secara harfiah, visual novel berarti sebuah cerita yang disajikan dalam

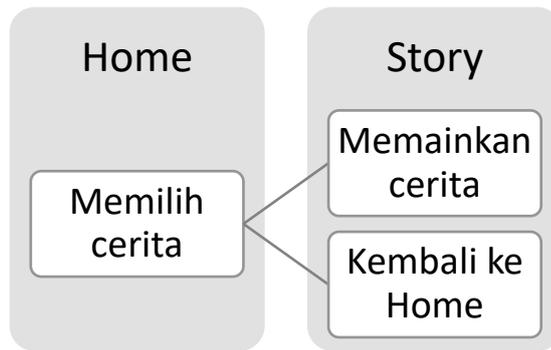
bentuk tidak hanya dengan teks, namun juga didampingi dengan visual, animasi, serta BGM. Proyek P-MVN menceritakan tentang seorang individu yang masuk ke dalam dunia visual novel, dimana visual yang dimasuki oleh individu merupakan cerita fiksi yang mengadopsi rakyat Ratu kidul. Gambar 5.40 merupakan contoh dari potongan permainan dalam permainan digital P-MVN.



Gambar 5.41. Cuplikan permainan digital P-MVN

### 5.1.3.2. *Mechanic* Proyek

Mekanisme dalam permainan digital P-MVN versi pertama ini masih sangat sederhana seperti yang dapat dilihat pada Gambar 5.41.



Gambar 5.42. Alur Proyek P-MVN

Dalam alur proyek P-MVN, pertama kali pemain menjalankan aplikasi akan dihadapkan pada interface home, disini pemain memiliki opsi untuk memilih cerita (namun versi pertama ini hanya memiliki 1 cerita). Cerita yang dipilih oleh pemain akan mengantarkan pemain ke interface story dimana dalam interface ini cerita tersebut dinarasikan, dan berinteraksi dengan pemain. Setelah keseluruhan cerita ini telah mencapai titik akhir, pemain memiliki opsi untuk kembali ke interface Home untuk memilih cerita lain. Berdasarkan alur permainan tersebut, proyek P-MVN memiliki dua interface yaitu interface Home, dan interface Story.

#### A. Interface Home

Seperti yang dijelaskan pada Gambar 5.41, Mekanisme pada Interface Home dalam proyek P-MVN hanya bertujuan untuk sebagai pustaka bagi pemain untuk memilih cerita yang akan dimainkan. Untuk versi pertama hanya satu cerita saja yang tersedia dari empat cerita yang direncanakan. Tabel 5.32 menyajikan daftar kebutuhan dan tugas berdasarkan mekanisme yang tersedia pada interface Home.

Tabel 5.32. Daftar kebutuhan Interface Home

No	Daftar kebutuhan (fitur)	Tugas
1	Berpindah ke Interface Story sesuai dengan pilihan cerita	Membuat fungsi untuk memanggil interface Story

## B. Interface Story

Permainan digital bergenre visual novel pada umumnya memiliki mekanisme berupa “*Next*” yaitu untuk menampilkan narasi cerita berikutnya. “*Previous*” untuk menampilkan narasi cerita sebelumnya, “*Save*” untuk menyimpan progres cerita, “*Load*” untuk melanjutkan cerita berdasarkan progres cerita yang disimpan dan “*return*” untuk kembali ke tampilan home. Mekanisme yang telah tertanam pada versi pertama dari proyek P-MVN adalah mekanisme “*Next*” dan “*Return*” saja. Selain itu interface story juga memiliki fungsi input nama pemain yang nantinya akan menjadi panggilan karakter kita saat berdialog dengan *NPC* lain. Berdasarkan deskripsi tersebut, daftar kebutuhan pada interface Story menjadi seperti Tabel 5.33.

Tabel 5.33. Daftar kebutuhan interface Story

No	Daftar kebutuhan (fitur)	Tugas
1	Pemain dapat melanjutkan ke narasi cerita berikutnya dengan menyentuh layar	Membuat fungsi mendeteksi sentuhan pemain yang berguna untuk menampilkan narasi cerita selanjutnya
2	Pemain dapat menginputkan nama untuk menjadi subjek dalam narasi cerita yang berkaitan dengan pemain atau pada dialog antara NPC dengan pemain	Membuat fungsi input nama pemain
3	Pemain dapat kembali ke tampilan Home setelah menyelesaikan cerita	Membuat fungsi untuk memanggil interface Home

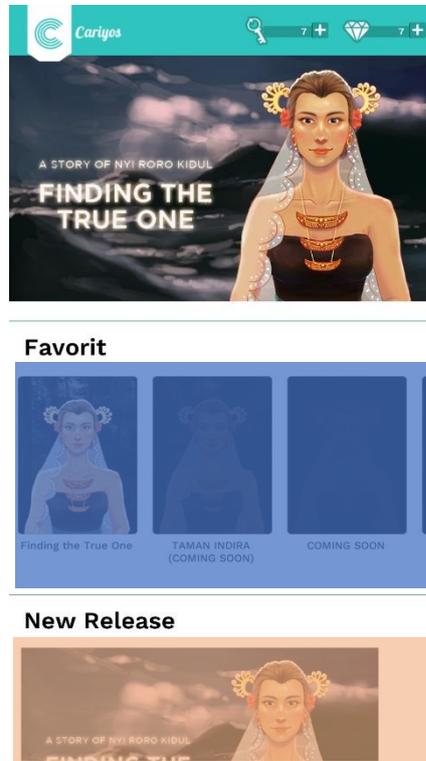
Perlu digarisbawahi fitur return pada versi pertama hanya tersedia saat pemain menamatkan cerita yang dipilih saja.

### 5.1.3.3. *Dynamic* Proyek

Sub bab ini akan membahas elemen dinamis proyek P-MVN pada interface Home dan Story.

#### A. **Interface Home**

Unsur dinamis yang memperkaya interaksi pada interface Home adalah kemampuan untuk scroll pilihan cerita secara horizontal seperti yang ditampilkan pada Gambar 5.42 pada area berwarna biru. Selain itu, terdapat juga area untuk menampilkan update cerita terbaru seperti pada berwarna area jingga.



Gambar 5.43. Unsur dinamis interface Home

Unsur-unsur dinamis pada interface Home membuat daftar kebutuhan untuk interface Home menjadi seperti Tabel 5.34.

Tabel 5.34. Update kebutuhan interface Home (Dynamic)

No	Daftar kebutuhan (fitur)	Tugas
1	Berpindah ke Interface Story sesuai dengan pilihan cerita	Membuat fungsi untuk memanggil interface Story
2	Pemain dapat menavigasi cerita dengan swipe ke kiri dan kekanan pada area pilihan cerita	Membuat fungsi yang mendeteksi inputan <i>swipe</i> pada area cerita
3	Pemain dapat mengetahui update cerita terbaru	Membuat fungsi untuk menampilkan thumbnail cerita terbaru yang ditambahkan ke dalam aplikasi
Keterangan		
 : Kebutuhan / tugas yang didapat pada tahapan Mechanic  : Kebutuhan / tugas yang didapat pada tahapan Dynamic		

## B. Interface Story

Elemen dinamis yang ditanamkan pada interface Story proyek P-MVN adalah kemampuan untuk memilih opsi pilihan pada saat cerita berlangsung. Opsi pilihan merupakan elemen dinamis yang sangat vital dalam permainan berjenis visual novel. Fitur opsi ini nantinya akan dikombinasikan dengan konsep *what if* yang mana pemilihan satu opsi akan mengarahkan cerita dari visual novel ke alur yang berbeda dan akan menghasilkan narasi yang berbeda pula. Gambar 5.43 merupakan contoh pilihan opsi pada proyek P-MVN yang mempengaruhi alur cerita kedepannya.



Gambar 5.44. Opsi pilihan yang merubah alur cerita

Berdasarkan deskripsi tersebut, daftar kebutuhan interface story hingga pada tahapan dynamic ini adalah seperti pada Tabel 5.35.

Tabel 5.35. Update kebutuhan interface Story (Dynamic)

No	Daftar kebutuhan (fitur)	Tugas
1	Pemain dapat melanjutkan ke narasi cerita berikutnya dengan menyentuh layar	Membuat fungsi mendeteksi sentuhan pemain yang berguna untuk menampilkan narasi cerita selanjutnya
2	Pemain dapat menginputkan nama untuk menjadi subjek dalam narasi cerita yang berkaitan dengan pemain atau pada dialog antara NPC dengan pemain	Membuat fungsi input nama pemain
3	Pemain dapat kembali ke tampilan Home setelah menyelesaikan cerita	Membuat fungsi untuk memanggil interface Home

No	Daftar kebutuhan (fitur)	Tugas
4	Alur cerita memiliki opsi yang dapat dipilih pemain	Membuat fungsi untuk menampilkan opsi
<b>Keterangan</b>		
 : Kebutuhan / tugas yang didapat pada tahapan Mechanic  : Kebutuhan / tugas yang didapat pada tahapan Dynamic		

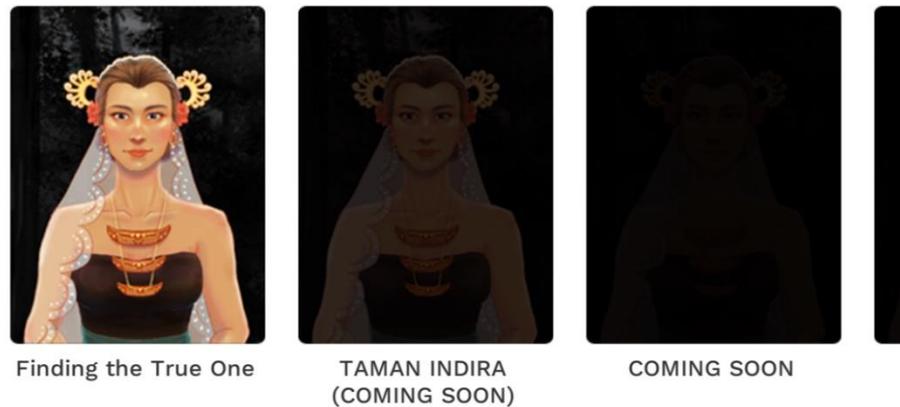
#### 5.1.3.4. *Aesthetic* Proyek

Pada sub bab ini peneliti akan menjelaskan elemen *Aesthetic* proyek P-MVN. Elemen *Aesthetic* pada genre visual novel memegang peranan utama untuk menarik perhatian pemain. Pada permainan P-MVN ini ilustrasi karakter dan background memiliki jenis berupa *painting-realistic* karena konsep cerita merupakan adaptasi dari cerita legenda daerah. Selanjutnya, peneliti akan membahas elemen *Aesthetic* pada dua interface yang tersedia pada proyek P-MVN yaitu interface Home dan Story.

##### A. Interface Home

Konsep tampilan yang dimiliki oleh P-MVN pada interface Home memiliki tipe minimalis. Tipe minimalis mengutamakan interaksi menggunakan gestur sentuhan seperti *swipe* dan *touch* namun tidak ditandai oleh sebuah tombol. Sebagai contoh untuk memilih sebuah cerita dalam proyek P-MVN pemain hanya perlu memilih berdasarkan ilustrasi thumbnail atau sampul dari cerita seperti yang diilustrasikan pada Gambar 5.44 sebagai pengganti dari ilustrasi tombol.

## Favorit



Gambar 5.45. Ilustrasi yang menggantikan tombol

Berdasarkan penjelasan diatas, maka kebutuhan pada interface Home menjadi bertambah karena masuknya unsur Aesthetic menjadi seperti pada Tabel 5.36.

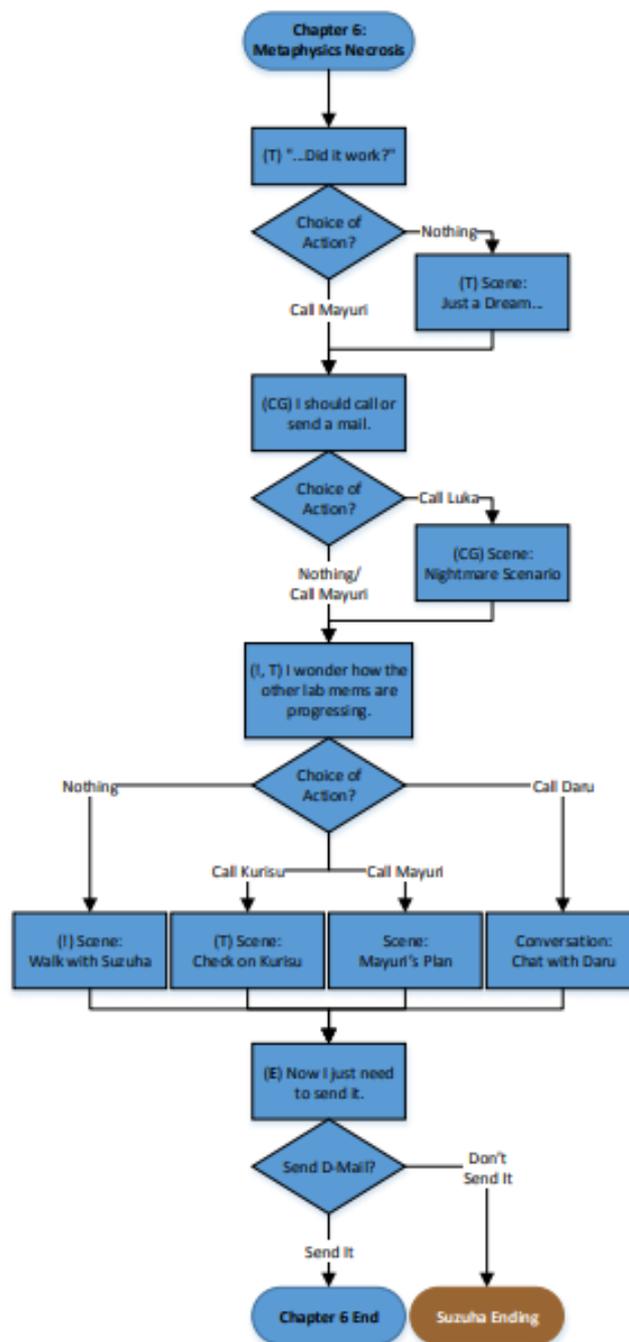
Tabel 5.36. Update kebutuhan Interface Home (Aesthetic)

No	Daftar kebutuhan (fitur)	Tugas
1	Berpindah ke Interface Story sesuai dengan pilihan cerita	Membuat fungsi untuk memanggil interface Story
		Membuat Ilustrasi Thumbnail yang mewakili karakter di cerita
2	Pemain dapat menavigasi cerita dengan swipe ke kiri dan kekanan pada area pilihan cerita	Membuat fungsi yang mendeteksi inputan <i>swipe</i> pada area cerita
3	Pemain dapat mengetahui update cerita terbaru	Membuat fungsi untuk menampilkan thumbnail cerita terbaru yang ditambahkan ke dalam aplikasi

No	Daftar kebutuhan (fitur)	Tugas
4	Tampilan minimalis pada tampilan home	Mengonsep tampilan pada interface Home
<b>Keterangan</b>		
 <ul style="list-style-type: none"> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: yellow; margin-right: 5px;"></span> : Kebutuhan / tugas yang didapat pada tahapan Mechanic</li> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: blue; margin-right: 5px;"></span> : Kebutuhan / tugas yang didapat pada tahapan Dynamic</li> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: green; margin-right: 5px;"></span> : Kebutuhan / tugas yang didapat pada tahapan Aesthetic</li> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: orange; margin-right: 5px;"></span> : Kebutuhan yang memiliki tugas teknis program dan visual</li> </ul>		

## B. Interface Story

Elemen aesthetic yang berperan penting pada interface Story adalah cerita, visual, dan audio. Elemen visual merupakan faktor penting karena menampilkan ilustrasi background, karakter, dan animasi bersamaan dengan narasi teks. Selain itu, juga terdapat elemen BGM yang mempercantik penyampaian cerita dengan iringan musik tertentu sesuai dengan situasi dalam cerita. Unsur visual dan audio BGM ini tentunya bergantung kepada alur cerita yang dikonsep oleh tim pengembang. Alur cerita merupakan daya jual utama bersama dengan visual dalam genre visual novel. Alur cerita pada visual novel umumnya bercabang, kompleks, dan memiliki multiple ending atau akhir cerita yang berbeda. Gambar 5.45 merupakan contoh alur pada visual novel Steins;Gate chapter 6.



Gambar 5.46. Contoh alur cerita pada VN Steins:Gate

Akhir cerita dapat berupa “*Bad end*” atau akhir cerita yang buruk yang diakibatkan pemain memilih opsi yang kurang tepat atau “*Good end*” saat pemain berhasil memilih opsi yang tepat di situasi tertentu. Namun terkadang sebuah cerita juga memiliki alur yang bernama “*True end*” atau

akhir yang sesungguhnya saat pemain dapat memilih opsi yang paling optimum pada setiap opsi dan sejalan dengan pattern atau pola yang dikonsept oleh tim pengembang. Setiap cerita dalam memiliki cabang yang mengakibatkan jalan cerita berbeda apabila memilih cabang yang satu dengan cabang lain. Semakin rumit cerita mengakibatkan semakin banyak cabang pada sebuah jalan cerita dan meningkatkan jumlah kondisi (*event*) yang dapat terjadi. Semakin banyak event akan mengakibatkan pekerjaan pengonsept cerita, ilustrator background dan karakter, animator, audio engineer semakin berat.

Kerumitan cerita dan pentingnya unsur visual dan audio membuat kebutuhan dan tugas pada interface Story menjadi bertambah dalam Tabel 5.37.

Tabel 5.37. Update kebutuhan interface Story (Aesthetic)

No	Daftar kebutuhan (fitur)	Tugas
1	Pemain dapat membaca narasi cerita atau dialog dengan menyentuh layar	Membuat fungsi mendeteksi sentuhan pemain yang berguna untuk menampilkan narasi cerita selanjutnya
		Mengonsept alur cerita
		Membuat ilustrasi untuk background / latar belakang
		Membuat desain ilustrasi karakter cerita
		Membuat animasi pada situasi dan kondisi tertentu (contoh saat situasi hujan, maka terdapat animasi petir dan rintik hujan)
		Membuat frame untuk tempat narasi cerita dan dialog

No	Daftar kebutuhan (fitur)	Tugas
2	Pemain dapat menginputkan nama untuk menjadi subjek dalam narasi cerita yang berkaitan dengan pemain atau pada dialog antara NPC dengan pemain	Membuat fungsi input nama pemain
		Membuat ilustrasi frame untuk input nama pemain
3	Pemain dapat kembali ke tampilan Home setelah menyelesaikan cerita	Membuat fungsi untuk memanggil interface Home
		Membuat ilustrasi tombol untuk fungsi kembali ke tampilan home
4	Alur cerita memiliki opsi yang dapat dipilih pemain	Membuat fungsi untuk menampilkan opsi
		Membuat ilustrasi tombol untuk tampilan opsi pilihan
		Merancang event untuk setiap cabang opsi
<b>Keterangan</b>		
 <ul style="list-style-type: none"> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: yellow; margin-right: 5px;"></span> : Kebutuhan / tugas yang didapat pada tahapan Mechanic</li> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: blue; margin-right: 5px;"></span> : Kebutuhan / tugas yang didapat pada tahapan Dynamic</li> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: green; margin-right: 5px;"></span> : Kebutuhan / tugas yang didapat pada tahapan Aesthetic</li> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: orange; margin-right: 5px;"></span> : Kebutuhan yang memiliki tugas teknis program dan visual</li> </ul>		

### 5.1.3.5. *Experience Proyek*

Dalam menciptakan proyek berbasis visual novel, tim pengembang harus mempelajari aturan untuk menampilkan sebuah narasi cerita. Umumnya genre visual novel akan memiliki dialog box pada sepertiga bawah tampilan yang digunakan untuk menampilkan narasi cerita atau dialog pada tokoh. Contoh Gambar 5.46 merupakan tampilan dialog box pada visual novel Steins;Gate.



Gambar 5.47. Contoh visual dalam cerita VN Steins;Gate

Sehingga proyek P-MVN juga mengikuti kaidah tersebut seperti pada Gambar 5.47.



Gambar 5.48. Contoh visual dalam cerita pada Proyek P-MVN

Berdasarkan Gambar 5.46 dan 5.47 juga dapat dilihat bahwa apabila sebuah dialog dilakukan oleh karakter tertentu, maka ilustrasi karakter juga ikut

ditampilkan untuk meingkatkan pemahaman pemain siapa karakter yang sedang berbicara saat dialog berlangsung.

### 5.1.3.6. *Assumption and constraint* Proyek

Pada visual novel, umumnya cerita dibagi menjadi tiga durasi yaitu short, yang berkisar 2-10 jam untuk menamatkan cerita, medium yang berkisar 10-30 jam, dan “long” yang berkisar 30-50 jam. Lamanya durasi permainan dipengaruhi oleh kompleksitas rute dan alur cerita. Semakin kompleks cerita semakin banyak scene yang berakibat semakin banyaknya ilustrasi yang harus dibuat. Dalam proyek P-MVN asumsi tim pengembang untuk masing-masing cerita merupakan cerita yang berdurasi pendek (2-10 jam).

### 5.1.3.7. Daftar kebutuhan final proyek P-MVN

Setelah langkah-langkah analisis komponen mechanic, dynamic, aesthetic, experience, dan assumption and constrain dilakukan pada proyek permainan digital P-MVN. Peneliti mendapatkan daftar kebutuhan final yang akan digunakan sebagai masukan ditahap selanjutnya. Daftar kebutuhan dan tugas final pada proyek P-MVN ditampilkan pada Tabel 5.38.

Tabel 5.38. Daftar kebutuhan final proyek P-MVN

No	Daftar kebutuhan (fitur)	Tugas
<i>Interface Home</i>		
1	Berpindah ke Interface Story sesuai dengan pilihan cerita	Membuat fungsi untuk memanggil interface Story Membuat Ilustrasi Thumbnail yang mewakili karakter di cerita
2	Pemain dapat menavigasi cerita dengan swipe ke kiri dan kekanan pada area pilihan cerita	Membuat fungsi yang mendeteksi inputan <i>swipe</i> pada area cerita
3	Pemain dapat mengetahui update cerita terbaru	Membuat fungsi untuk menampilkan thumbnail cerita

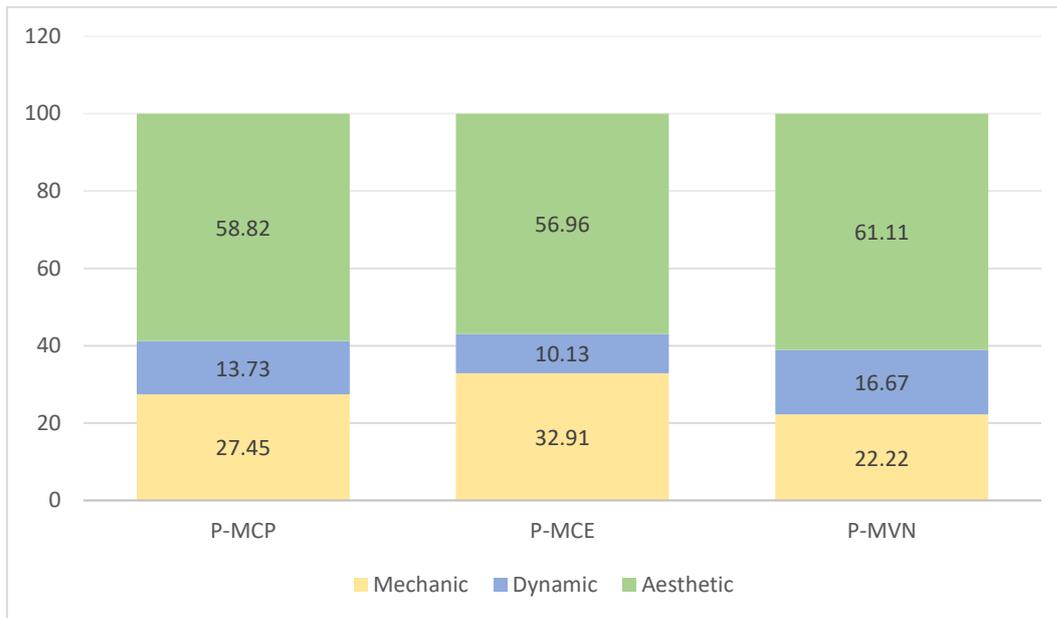
No	Daftar kebutuhan (fitur)	Tugas
		terbaru yang ditambahkan ke dalam aplikasi
4	Tampilan minimalis pada tampilan home	Mengonsep tampilan pada interface Home
<i>Interface Story</i>		
5	Pemain dapat membaca narasi cerita atau dialog dengan menyentuh layar	Membuat fungsi mendeteksi sentuhan pemain yang berguna untuk menampilkan narasi cerita selanjutnya
		Mengonsep alur cerita
		Membuat ilustrasi untuk background / latar belakang
		Membuat desain ilustrasi karakter cerita
		Membuat animasi pada situasi dan kondisi tertentu (contoh saat situasi hujan, maka terdapat animasi petir dan rintik hujan)
	Membuat frame untuk tempat narasi cerita dan dialog	
6	Pemain dapat menginputkan nama untuk menjadi subjek dalam narasi cerita yang berkaitan dengan pemain atau pada dialog antara NPC dengan pemain	Membuat fungsi input nama pemain
		Membuat ilustrasi frame untuk input nama pemain
7	Pemain dapat kembali ke tampilan Home setelah menyelesaikan cerita	Membuat fungsi untuk memanggil interface Home
		Membuat ilustrasi tombol untuk fungsi kembali ke tampilan home

No	Daftar kebutuhan (fitur)	Tugas
8	Alur cerita memiliki opsi yang dapat dipilih pemain	Membuat fungsi untuk menampilkan opsi
		Membuat ilustrasi tombol untuk tampilan opsi pilihan
		Merancang event untuk setiap cabang opsi
<b>Keterangan</b>		
 <ul style="list-style-type: none"> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: yellow; margin-right: 5px;"></span> : Kebutuhan / tugas yang didapat pada tahapan Mechanic</li> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: blue; margin-right: 5px;"></span> : Kebutuhan / tugas yang didapat pada tahapan Dynamic</li> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: green; margin-right: 5px;"></span> : Kebutuhan / tugas yang didapat pada tahapan Aesthetic</li> <li><span style="display: inline-block; width: 15px; height: 15px; background-color: orange; margin-right: 5px;"></span> : Kebutuhan yang memiliki tugas teknis program dan visual</li> </ul>		

Berdasarkan hasil dari kebutuhan yang didapat dari proyek P-MVN, 61.11% dari pekerjaan yang harus dilakukan merupakan pekerjaan yang berubungan dengan elemen Aesthetic.

#### 5.1.4. Elemen aesthetic mendominasi permainan digital

Proyek permainan digital yang digunakan dalam penelitian ini terdiri atas dua permainan bergenre casual (P-MCP dan P-MCE) dan satu bergenre visual novel (P-MVN). Berdasarkan hasil dari daftar kebutuhan final yang didapatkan dari ketiga proyek, peneliti menemukan bahwa faktor Aesthetic terutama elemen visual menjadi tugas yang paling banyak dikerjakan dari seluruh tugas perangkat lunak permainan digital. Gambar 5.48 mengilustrasikan presentase elemen mechanic, dynamic dan aesthetic pada ketiga proyek.



Gambar 5.49. Presentase komponen dalam permainan digital

Dengan mendominasinya elemen aesthetic (visual, cerita, audio) hingga rata-rata 58.96% dari proyek permainan digital, secara konseptual telah terbukti bahwa apabila sebuah metode estimasi biaya tidak mempertimbangkan faktor aesthetic dalam perhitungan perangkat lunak permainan digital maka besar kemungkinannya hasil estimasi biaya akan jauh dengan biaya aslinya.

## 5.2. Analisa biaya asli proyek permainan digital

Dalam upaya membangun ketiga proyek permainan digital, tentunya biaya, durasi, dan banyaknya anggota tim pengembang menjadi informasi yang sangat penting untuk didapatkan. Informasi tersebut dapat dimanfaatkan sebagai acuan untuk proyek selanjutnya yang sejenis sehingga mempermudah proses negosiasi harga. Tabel 5.39 menyajikan informasi data real pada ketiga proyek yaitu P-MCP, P-MCE, dan P-MVN.

Tabel 5.39. Detail proyek permainan digital

No	Nama Proyek	Platform	Anggota Tim	Target waktu	Realisasi waktu	Biaya Asli
1.	P-MCP	Mobile	1 Programmer; 1 Illustrator	1 bulan	1.5 bulan	7.000.000
2.	P-MCE	Mobile	1 Programmer; 1 Illustrator	1 bulan	1 bulan	5.000.000
3.	P-MVN	Mobile	2 Programmer; 6 Illustrator	3 bulan	3 bulan	40.000.000

Dalam penelitian ini informasi biaya asli digunakan peneliti untuk mendapatkan nilai akurasi dari metode estimasi biaya perangkat lunak yang digunakan. Berdasarkan data pada Tabel 5.39 juga didapatkan rata-rata gaji seorang pengembang permainan digital berdasarkan studio Re:tale adalah Rp. 3.500.000 / bulan. Perlu digaris bawahi bahwa biaya untuk seluruh proyek yang digunakan dalam penelitian ini merupakan biaya yang dibayarkan secara langsung saat awal negosiasi proyek atau di akhir saat proyek telah selesai dengan durasi target waktu pengerjaan untuk setiap proyek dinegosiasikan di awal. Apabila nominal biaya dan durasi telah disepakati, maka tim pengembang harus dapat menyelesaikan proyek sesuai target waktu dan bila mengalami keterlambatan maka biaya keterlambatan di tanggung oleh tim pengembang itu sendiri, atau bahkan dalam satu kasus tim pengembang malah mendapatkan denda dari pemilik proyek.

### 5.3. Perhitungan estimasi biaya

Dalam setiap proses atau upaya dalam menciptakan perangkat lunak, biaya merupakan faktor penentu keberhasilan terciptanya perangkat lunak. Apabila biaya yang dianggarkan tidak mencukupi kemungkinan pengembangan perangkat lunak dapat tertunda atau bahkan ditinggalkan; termasuk permainan digital. Oleh karena itu, peran estimasi biaya menjadi penting dan sangat diharapkan akurasinya sehingga apabila biaya diukur dengan benar maka proyek akan dapat berjalan sesuai rencana. Untuk mendapatkan atau menghitung estimasi biaya, nilai utama yang

harus didapatkan adalah nilai ukuran perangkat lunak dan umumnya nilai tersebut didapatkan menggunakan metode FPA.

### **5.3.1. Perhitungan Function point**

Ukuran perangkat lunak merupakan elemen vital untuk proses estimasi biaya tidak terkecuali proyek permainan digital. Untuk mendapatkan ukuran perangkat lunak permainan digital pada Metode FPA adalah mendapatkan nilai function point. Nilai function point merepresentasikan ukuran perangkat lunak yang diukur. Berikut ini adalah langkah-langkah yang diperlukan untuk mendapatkan nilai function point.

#### **5.3.1.1. Identifikasi fungsi terkecil (primary function)**

Tahapan ini bertujuan untuk memecah fungsi-fungsi yang ada pada aplikasi menjadi satuan fungsi terkecil. Dalam penelitian ini, untuk menentukan fungsi terkecil, peneliti menggunakan daftar kebutuhan dari ketiga proyek pada Tabel 5.15, Tabel 5.31, dan Tabel 5.38.

#### **5.3.1.2. Klasifikasi tipe fungsi**

Fungsi-fungsi terkecil yang telah teridentifikasi kemudian diklasifikasikan menjadi tipe fungsi yang terdiri atas Internal Logical File (ILF), External Interface File (EIF), External Input (EI), External Output (EO) dan External Inquiry (EQ). Untuk penjelasan karakteristik masing-masing tipe fungsi dapat merujuk pada sub 2.1.3.2 tahap 3 dan 4.

#### **5.3.1.3. Tentukan tingkat kompleksitas tipe fungsi**

Langkah ini bertujuan untuk mendapatkan bobot pada semua tipe fungsi berdasarkan nilai RET, FTR, dan DET (mengacu pada ketentuan di Tabel 2.2, Tabel 2.3, Tabel 2.4, Tabel 2.5, dan Tabel 2.6).

Contoh pengklasifikasian tipe fungsi dan penentuan tingkat bobot tipe fungsi pada proyek P-MCP disajikan pada Tabel 5.40 (perhitungan lebih detail pada Lampiran 1.A).

Tabel 5.40. Klasifikasi fungsi dan bobot proyek P-MCP

No	Daftar kebutuhan (fungsi)	Tugas	Function Point (FP)	
			Type	Weight
<b>Interface Home</b>				
1	Berpindah ke Interface Pilihan Level	Membuat fungsi untuk berpindah ke Tampilan Pilihan Level	EQ	3
		Membuat desain tombol “play” yang mengindikasikan memulai permainan	Diabaikan	
2	Pemain dapat mengaktifkan atau mematikan efek suara	Membuat fungsi untuk mengaktifkan atau mematikan efek suara	EI ILF	3 7
		Membuat desain tombol efek suara “aktif” dan “nonaktif”	Diabaikan	
3	Menampilkan iklan pada aplikasi	Membuat fungsi untuk menampilkan iklan	EQ	3
4	Menghilangkan iklan dengan mengaktifkan pembelian dalam aplikasi	Membuat fungsi untuk menghilangkan iklan	EIF	5
		Membuat desain tombol “no ads”	Diabaikan	
5	Menampilkan website developer	Membuat fungsi untuk membuka browser dengan url website developer.	EQ	3
		Membuat desain tombol Info	Diabaikan	

No	Daftar kebutuhan (fungsi)	Tugas	Function Point (FP)	
			Type	Weight
6	Memiliki visualisasi Interface Home yang menarik	Membuat desain background untuk interface Home	Diabaikan	
		Membuat desain logo untuk identitas aplikasi	Diabaikan	
		Merancang komposisi elemen-elemen tampilan visual.	Diabaikan	
<b>Interface Pilihan Level</b>				
7	Memiliki visualisasi interface Pilihan Level yang menarik	Merancang konsep dan komposisi visual	Diabaikan	
		Mendesain background untuk interface Pilihan Level	Diabaikan	
		Mendesain label “Level Selection” pada interface Pilihan Level	Diabaikan	
8	Berpindah ke Interface Home	Membuat fungsi untuk berpindah ke interface Home	EQ	3
		Membuat desain tombol untuk fungsi kembali ke interface Home	Diabaikan	
9	Berpindah ke Interface Gameplay sesuai pilihan level	Membuat fungsi untuk berpindah ke interface Gameplay	EQ	3
10	Navigasi daftar tingkatan level	Membuat fungsi untuk menampilkan daftar tingkatan level dengan range tertentu	EQ	3

No	Daftar kebutuhan (fungsi)	Tugas	Function Point (FP)	
			Type	Weight
		Membuat desain tombol navigasi daftar tingkatan level	Diabaikan	
11	Pemain dapat memainkan level tertentu apabila level sebelumnya telah diselesaikan	Membuat fungsi untuk mengunci dan membuka level.	ILF	7
		Membuat desain tombol untuk tingkatan level dengan kondisi terbuka dan terkunci	Diabaikan	
<b>Interface Gameplay</b>				
12	Konsep level yang berbeda dalam 14 tingkatan level	Mengonsep mapping permainan (termasuk posisi kereta dimulai, alur rel, lokasi stasiun kereta)	Diabaikan	
		Mendesain ilustrasi background		
13	Memunculkan (spawn) objek kereta pada posisi dan waktu tertentu.	Membuat fungsi untuk memunculkan kereta api.	EQ	3
		Desain objek kereta (14 jenis warna)	Diabaikan	
14	Mendefinisikan dan menampilkan waktu permainan.	Mendefinisikan dan membuat fungsi untuk menampilkan dan kalkulasi waktu permainan.	EO	4
		Desain indikator waktu	Diabaikan	
15	Mendefinisikan dan menampilkan nyawa pemain.	Mendefinisikan dan membuat fungsi untuk menampilkan dan kalkulasi nyawa.	EO	4
		Desain indikator nyawa pemain	Diabaikan	

No	Daftar kebutuhan (fungsi)	Tugas	Function Point (FP)	
			Type	Weight
16	Pemain mendapatkan pengurangan nyawa untuk setiap kereta yang masuk ke stasiun yang salah (tidak sesuai dengan warna).	Membuat fungsi untuk kalkulasi nyawa setiap kesalahan terjadi.	Diabaikan	
		Desain stasiun kereta api	Diabaikan	
17	Menentukan kecepatan pergerakan kereta.	Membuat fungsi untuk menjalankan kereta berdasarkan kecepatan tertentu.	Diabaikan	
18	Merubah arah rel kereta berdasarkan input pemain.	Membuat fungsi untuk menerima input pemain yang berguna untuk merubah arah rel kereta.	EI ILF	3 7
		Desain indikator sebagai fungsi merubah arah rel kereta berdasarkan input pemain	Diabaikan	
19	Kereta dapat berjalan sesuai dengan alur rel kereta	Membuat fungsi pathfinding untuk objek kereta	Diabaikan	
		Desain rel kereta api	Diabaikan	
20	Pemain dapat menghentikan permainan	Membuat fungsi untuk menghentikan permainan sementara.	EI	3

No	Daftar kebutuhan (fungsi)	Tugas	Function Point (FP)	
			Type	Weight
	sementara dan melanjutkannya kembali	Desain tombol pause untuk fungsi menghentikan permainan sementara	Diabaikan	
		Desain tulisan "pause"	Diabaikan	
		Desain frame untuk kondisi pause	Diabaikan	
21	Menampilkan jumlah bintang yang menandakan nilai dari pemain.	Membuat fungsi untuk mengakumulasi nilai yang didapat pemain berdasarkan jumlah kereta yang masuk ke stasiun yang benar dan menampikannya.	EO	4
			ILF	7
		Membuat desain bintang untuk visualisasi nilai	Diabaikan	
		Desain frame sebagai pelat penanda.	Diabaikan	
	Desain label	Diabaikan		
22	Memungkinkan pemain menuju level selanjutnya apabila memenangkan permainan	Membuat fungsi menuju tingkatan level selanjutnya	EQ	3
		Desain tombol yang mewakili fungsi untuk memainkan permainan di tingkatan level selanjutnya.	Diabaikan	
23	Mengulang permainan.	Membuat fungsi untuk mengulang permainan.	EQ	3

No	Daftar kebutuhan (fungsi)	Tugas	Function Point (FP)	
			Type	Weight
		Desain tombol yang mewakili fungsi untuk mengulang permainan	Diabaikan	
24	Berpindah ke Interface Pilihan Level dengan menampilkan iklan terlebih dahulu.	Membuat fungsi untuk menampilkan iklan yang kemudian berpindah ke interface Pilihan Level.	EQ	3
		Desain tombol yang mewakili fungsi untuk kembali ke interface Pilihan Level	Diabaikan	
Keterangan				
 : Kebutuhan / tugas yang didapat pada tahapan Mechanic  : Kebutuhan / tugas yang didapat pada tahapan Dynamic  : Kebutuhan dan tugas yang didapat pada tahapan Aesthetic  : Kebutuhan yang memiliki tugas teknis program dan visual				

Metode FPA hanya dapat menilai tugas yang bersifat fungsional dari keseluruhan tugas pada permainan digital. Tugas fungsional yang dapat dihitung menggunakan metode FPA juga hanya melingkupi fungsi unik (fungsi yang sama dihitung satu kali) dari input dan output. Oleh karena itu, tugas yang merupakan internal proses seperti yang disajikan pada Tabel 5.40 yaitu *membuat fungsi untuk mengakumulasi nilai yang didapat pemain berdasarkan jumlah kereta yang masuk ke stasiun yang benar* atau *membuat fungsi untuk menjalankan kereta berdasarkan kecepatan tertentu* akan diabaikan karena merupakan sebuah mekanisme proses.

Selain itu, keseluruhan tugas Aesthetic (tugas yang bersifat seni) juga dihiraukan karena bukan merupakan sebuah fungsi melainkan hanya faktor estetika.

Hal tersebut jelas mempengaruhi akan mempengaruhi besaran perangkat lunak yang dihasilkan apabila menghiraukan tugas aesthetic karena pekerjaan aesthetic sendiri mencakup 30 dari 51 (58.82%) daftar tugas keseluruhan pada proyek P-MCP.

#### 5.3.1.4. Perhitungan total function point

Tahapan terakhir ini bertujuan untuk mengakumulasi jumlah masing-masing tipe fungsi beserta setiap nilai kompleksitasnya. Tabel perhitungan function point yang disajikan pada Tabel 2.7 dapat digunakan untuk mempermudah perhitungan total nilai function point. Tabel 5.41 contoh perhitungan nilai total function point untuk proyek P-MCP.

Tabel 5.41. Perhitungan nilai FP Proyek P-MCP

Function Type	Kompleksitas						Total
	Low		Average		High		
	(Val) x Count	Sum	(Val) x Count	Sum	(Val) x Count	Sum	
EI	(3) x 2	6	(4) x 0	0	(6) x 0	0	6
EO	(4) x 3	12	(5) x 0	0	(7) x 0	0	12
EQ	(3) x 10	30	(4) x 0	0	(6) x 0	0	30
ILF	(7) x 4	28	(10) x 0	0	(15) x 0	0	28
EIF	(5) x 1	5	(7) x 0	0	(10) x 0	0	5
	<b>Function Point</b>						<b>81</b>

Berdasarkan hasil perhitungan pada Tabel 5.41, ukuran perangkat lunak permainan digital P-MCP adalah sebesar 81 FP. Tahapan perhitungan function point yang meliputi identifikasi fungsi terkecil, klasifikasi tipe fungsi, menentukan nilai kompleksitas, dan perhitungan total nilai function point juga dilakukan proyek P-MCE dan P-MVN (perhitungan dapat dilihat di Lampiran 1.B dan Lampiran 1.C) sehingga menghasilkan besaran ukuran perangkat lunak pada masing-masing perangkat lunak permainan digital seperti pada Tabel 5.42.

Tabel 5.42. Nilai FP ketiga proyek permainan digital

No	Nama Proyek	Presentase ukuran fungsional	Besaran ukuran perangkat lunak (FP)
1	P-MCP	27.45 %	81
2	P-MCE	32.91%	110
3	P-MVN	23.53%	26

Berdasarkan Tabel 5.42 diketahui bahwa permainan digital P-MVN yang sebagian besar bobot pekerjaannya pada unsur Aesthetic (58.82%) mendapatkan nilai ukuran perangkat lunak yang paling kecil berdasarkan perhitungan metode FPA karena memiliki elemen fungsional yang sedikit yaitu 23.53% dibanding proyek P-MCP yang memiliki presentase 27.45% dan terbesar secara fungsional adalah proyek P-MCE dengan 32.91%.

### 5.3.2. Perhitungan effort

Nilai effort merupakan interpretasi seberapa besar usaha yang diperlukan oleh tim pengembang untuk membangun perangkat lunak. Dalam penelitian ini, nilai effort didapatkan dengan menggunakan persamaan (3) yaitu:

$$Effort = Size * Project Delivery Rate * 8.2$$

Dimana nilai Size merupakan hasil tahapan perhitungan metode FPA yaitu nilai function point seperti yang dihasilkan pada Tabel 5.42. Nilai konstanta 8.2 merupakan nilai effort rate yang menyatakan nilai usaha manusia per jamnya (man/hour) untuk setiap satu function point (Subriadi and Ningrum, 2014). Sedangkan PDR merupakan durasi penyelesaian per size Function Point (fp/hour).

Dalam tahapan ini peneliti ingin menguji seberapa besar akurasi metode FPA murni dalam mengestimasi biaya perangkat lunak permainan digital. Oleh karena Nilai PDR merupakan nilai konstanta yang dihasilkan oleh metode usulan peneliti, pada langkah ini nilai PDR didefinisikan sebagai “1” supaya tidak mempengaruhi hasil dari perhitungan estimasi biaya menggunakan metode FPA murni sehingga nilai usaha untuk proyek P-MCP adalah:

$$\begin{aligned}
 \text{Effort} &= \text{Size} * \text{Project Delivery Rate} * 8.2 \\
 &= 81 * 1 * 8.2 \\
 &= 664.2
 \end{aligned}$$

Hasil dari perhitungan usaha (*Effort*) untuk membangun ketiga proyek permainan digital disajikan pada Tabel 5.43.

Tabel 5.43. Effort per proyek permainan digital

No	Nama Proyek	FP	Effort Rate * PDR	Effort Proyek
1	P-MCP	81	8.2	664.2
2	P-MCE	110	8.2	902
3	P-MVN	26	8.2	213.2

### 5.3.3. Perhitungan effort per aktivitas

Setiap usaha pembangunan perangkat lunak terdiri dari berbagai aktifitas. Setiap aktivitasnya tentu memiliki tingkat usaha masing-masing tergantung tingkat intensitasnya. Dalam penelitian ini peneliti menggunakan daftar aktivitas yang digagas oleh Primandari dan Sholiq (2015) yang merupakan pengembangan penelitian dari Saleh (2011). Tabel 5.44 berikut merupakan implementasi hasil dari nilai effort per proyek pada Tabel 5.42 ke daftar aktifitas tersebut.

Tabel 5.44. Nilai usaha pada setiap aktifitas proyek

No	Activity	Effort distribution (%)	Project Name		
			P-MCP	P-MCE	P-MVN
<i>Software development phase</i>					
1	Requirements	1.17%	7.771	10.553	2.494
2	Specifications	6.75%	44.834	60.885	14.391
3	Design	5.57%	36.996	50.241	11.875
4	Implementation	55.65%	369.627	501.963	118.646
5	Integration Testing	6.42%	42.642	57.908	13.687

No	Activity	Effort distribution (%)	Project Name		
			P-MCP	P-MCE	P-MVN
6	Acceptance & deployment	5.6%	37.195	50.512	11.939
<b><i>Ongoing activities &amp; quality and testing</i></b>					
7	Project management	2.55%	16.937	23.001	5.437
8	Configuration management	3.58%	23.778	32.292	7.633
9	Quality assurance	0.66%	4.384	5.953	1.407
10	Documentation	9.76%	64.826	88.035	20.808
11	Training & support	0.6%	3.985	5.412	1.279
12	Evaluation & testing	1.67%	11.092	15.063	3.560
<b>Total</b>		100%	664.2	902	213.2

#### 5.3.4. Perhitungan biaya per aktivitas

Dalam membangun perangkat lunak, setiap individu dalam satu kelompok tim pengembang memiliki peran masing-masing tidak terkecuali dalam pengembangan permainan digital. Peran dalam sebuah studio yang memiliki jumlah karyawan yang besar berbeda dengan peran pada studio *indie*. Dalam studio besar setiap peran dimungkinkan dikerjakan oleh satu divisi yang terdiri dari puluhan orang sehingga peran pada studio permainan digital besar memiliki variasi yang besar dan spesifik.

Studio *indie* yang memiliki jumlah personil terbatas, sehingga studio harus memaksimalkan jumlah personil yang ada untuk mengerjakan berbagai macam tugas sehingga satu individu dapat bertanggung jawab dalam berbagai peran. Pada penelitian ini peneliti memilih studi kasus pada studio permainan skala kecil (*indie*) sehingga penentuan peran ditentukan oleh studio yang bersangkutan. Peran yang ada dalam studio *re:tale* beserta aktivitas yang dilakukan dapat dilihat pada Tabel 5.45.

Tabel 5.45. Daftar posisi pada studio re:tale

<b>Peran (role)</b>	<b>Deskripsi pekerjaan</b>	<b>Aktivitas terkait</b>
Game designer	Bertanggung jawab untuk menentukan konsep, alur cerita, dan gameplay pada permainan digital serta menentukan kebutuhan. Terkadang dalam grup kecil game designer juga membantu untuk manajemen proyek.	<ul style="list-style-type: none"> <li>- Requirement</li> <li>- Specification</li> <li>- Design</li> <li>- Project Management</li> <li>- Design</li> <li>- Quality assurance</li> <li>- Evaluation &amp; Testing</li> </ul>
Programmer	Bertanggung jawab terhadap implementasi konsep game pada game engine.	<ul style="list-style-type: none"> <li>- Implementation</li> <li>- Integration Testing</li> <li>- Acceptance &amp; deployment</li> <li>- Configuration management</li> <li>- Training &amp; support</li> </ul>
Illustrator / Artist	Bertanggung jawab terhadap tampilan keseluruhan pada permainan digital (seperti karakter, font, tampilan interface, pemilihan warna dan tema) berdasarkan konsep yang ada.	<ul style="list-style-type: none"> <li>- Implementation</li> <li>- Documentation</li> </ul>

Untuk mendapatkan hitungan estimasi biaya, referensi gaji merupakan faktor penting yang harus didapatkan. Sayangnya, tidak ada yang badan resmi yang membahas standar gaji untuk pengembang permainan digital. Oleh karena itu, peneliti berusaha mencari referensi di berbagai situs kerja untuk mendapatkan referensi gaji permainan digital. Tabel 5.46 merupakan referensi gaji yang akan diproses pada langkah selanjutnya.

Tabel 5.46. Referensi gaji pengembang permainan digital.

No	Sumber	Posisi	Skala Perusahaan	Rata-rata gaji per bulan
1	Job Planet.com	Tidak spesifik	Tidak spesifik	6,505,000
		Tidak spesifik	Tidak spesifik	3,205,000
2	Indeed.com	Tidak terdefinisi	Tidak spesifik	4,547,673
3	Qerja.com	Programmer	Tidak spesifik	5,000,000
		Software engineer	Tidak spesifik	6,500,000
		System Analyst	Tidak spesifik	9,250,000
		Software developer	Tidak spesifik	6,500,000

Dalam kenyataan dilapangan, ternyata gaji pengembang permainan digital berdasarkan situs kerja yang peneliti pakai memiliki variasi yang luar biasa. hal tersebut karena setiap studio permainan menetapkan preferensi standar gaji tersendiri. Oleh karena itu, peneliti harus mencari perkiraan gaji yang sesuai atau mendekati preferensi gaji pada setiap posisi di studio re:tale. Proses pencarian perkiraan gaji yang sesuai dilakukan dengan proses wawancara pada salah satu perwakilan anggota re:tale. Lebih spesifik, saat wawancara berlangsung, peneliti menunjukkan referensi gaji pada Tabel 5.46 untuk selanjutnya informan memilih rata-rata gaji mana yang terdekat pada setiap posisi di studio re:tale. Tabel 5.47 menyajikan nilai gaji yang mendekati pada masing-masing posisi / peran dalam studio re:tale.

Tabel 5.47. Perkiraan gaji pada studio re:tale

No	Posisi	Gaji yang mendekati (Rupiah)
1	Game Designer	Rp. 6,505,000
2	Programmer	Rp. 3,205,000
3	Illustrator	Rp. 3,205,000

Dapat dilihat pada tabel 5.47 ternyata referensi gaji yang paling mendekati adalah referensi gaji semuanya berasal dari situs Jobplanet.com. Peneliti merasa situs tersebut dapat digunakan sebagai acuan walaupun tidak terspesifikasi peran untuk setiap nilai gajinya namun dapat menjadi acuan mengingat nilai gaji tersebut didapatkan berdasarkan hasil survey dari 6.500 responden dalam riset yang dilakukan oleh Jobplanet (Jobplanet.com, 2016) dan ternyata relevan dengan gaji pada setiap peran tim pengembang di studio re:tale .

Selanjutnya, untuk mendapatkan nilai biaya per aktifitas, langkah pertama yang harus dilakukan adalah memperoleh nilai rate gaji per jam atau *Person Hours Rate* (PHR). Proses mendapatkan PHR terdiri dari tiga tahapan. Pertama, nilai gaji perbulan (PMR) untuk setiap posisi dirubah menjadi nilai per minggunya (PWR). Proses pengubahan nilai PMR menjadi PWR dilakukan dengan menggunakan persamaan (8) sehingga apabila perkiraan gaji seorang *Game Designer* per bulan adalah Rp. 6,505,000, maka nilai gaji per minggunya adalah:

$$\begin{aligned}
 PWR &= \frac{PMR}{4.1} \\
 &= \frac{6,505,000}{4.1} \\
 &= 1,585,585
 \end{aligned}$$

Tahapan kedua adalah mendapatkan nilai gaji per harinya (PDR) dari nilai gaji PWR dengan menggunakan persamaan (9) sehingga nilai perkiraan gaji seorang *Game Designer* per harinya adalah:

$$\begin{aligned}
 PDR &= \frac{PWR}{22} * 1.1 \\
 &= \frac{1,585,585}{22} * 1.1 \\
 &= 79,279
 \end{aligned}$$

Tahapan terakhir adalah mendapatkan nilai gaji per jamnya atau (PHR) menggunakan persamaan (10), sehingga apabila gaji Game Designer per harinya adalah Rp. 79,279 maka gaji per jamnya adalah:

$$\begin{aligned}
 PHR &= \frac{PDR}{8} * 1.3 \\
 &= \frac{79,279}{8} * 1.3 \\
 &= 12.883
 \end{aligned}$$

Tabel 5.48 berikut menjabarkan nilai dari gaji per bulan (PMR), gaji per minggu (PWR), gaji per hari (PDR), dan gaji per jam (PHR) dari masing-masing posisi pada studio re:tale.

Tabel 5.48. Pay rate untuk setiap posisi

No	Posisi	PMR (Rp)	PWR (Rp)	PDR (Rp)	PHR (Rp)
1	Game Designer	6,505,000	1,585,585	79,279	12,883
2	Programmer	3,205,000	781,707	39,085	6,351
3	Illustrator	3,205,000	781,707	39,085	6,351

Untuk mendapatkan biaya per aktivitiitas, langkah selanjutnya adalah mengalikan setiap effort peraktivitas yang nilainya telah didapatkan pada Tabel 5.44 dengan nilai pay rate dari hasil perhitungan yang ditampilkan pada Tabel 5.48 sehingga biaya aktifitas kebutuhan (requirement) pada proyek P-MCP berdasarkan persamaan (11) adalah:

$$\begin{aligned}
 Cost\ in\ Req &= Effort * percentage_{Req} * pay\ Rate_{req} \\
 &= 664.2 * 1.17\% * 12,883 \\
 &= 7.771 * 12,883 \\
 &= 100,116
 \end{aligned}$$

Selanjutnya peneliti melakukan tahapan yang sama untuk seluruh aktivitas pada ketiga proyek permainan digital yaitu P-MCP, P,MCE, dan P-MVN sehingga hasil perhitungan biaya setiap aktivitas tersaji pada Tabel 5.49 berikut.

Tabel 5.49. Pay rate per aktivitas permainan digital

No	Aktivitas	Pay rate	Project Name		
			P-MCP	P-MCE	P-MVN
<i>Software development phase</i>					
1	Requirements	12,883	100,116	135,959	32,136
2	Specifications	12,883	577,590	784,381	185,399
3	Design	12,883	476,619	647,260	152,989
4	Implementation	6,351	2,347,503	3,187,967	753,519
5	Integration Testing	6,351	270,817	367,776	86,929
6	Acceptance & deployment	6,351	236,227	320,802	75,826
<i>Ongoing activities &amp; quality and testing</i>					
7	Project management	12,883	218,201	296,322	70,040
8	Configuration management	6,351	151,016	205,084	48,474
9	Quality assurance	12,883	56,475	76,695	18,128
10	Documentation	6,351	411,709	559,112	132,154
11	Training & support	6,351	25,310	34,372	8,124
12	Evaluation & testing	12,883	142,900	194,062	45,869
<b>Total</b>			5,014,483	6,809,792	1,609,587

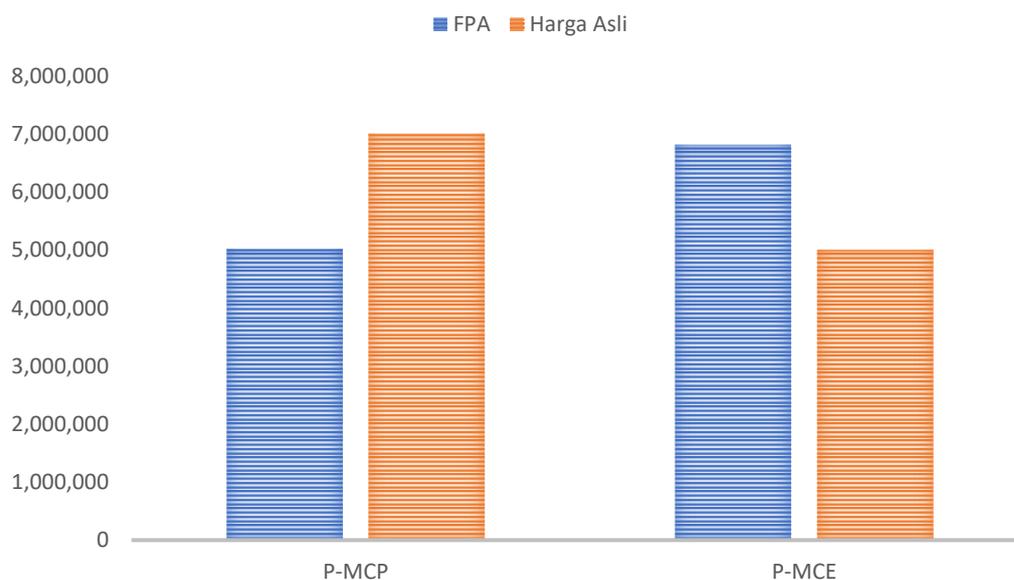
### 5.3.5. Analisa gap hasil estimasi metode FPA

Berdasarkan hasil pada Tabel 5.49 didapatkan nilai estimasi harga untuk ketiga proyek yaitu total dari biaya per aktivitas. Hasil estimasi biaya pada ketiga proyek yaitu P-MCP, P-MCE dan P-MVN disajikan pada Tabel 5.50.

Tabel 5.50. Estimasi biaya menggunakan FPA murni

No	Nama Proyek	Estimasi Biaya
1	P-MCP	Rp. 5,014,483
2	P-MCE	Rp. 6,809,792
3	P-MVN	Rp. 1,609,587

Nilai yang dihasilkan oleh perhitungan FPA kemudian dibandingkan dengan harga asli dari ketiga proyek. Gambar 5.49 mengilustrasikan perbandingan antara harga estimasi dan harga asli proyek P-MCP dan P-MCE.



Gambar 5.50. Perbandingan harga pada proyek P-MCP dan P-MCE

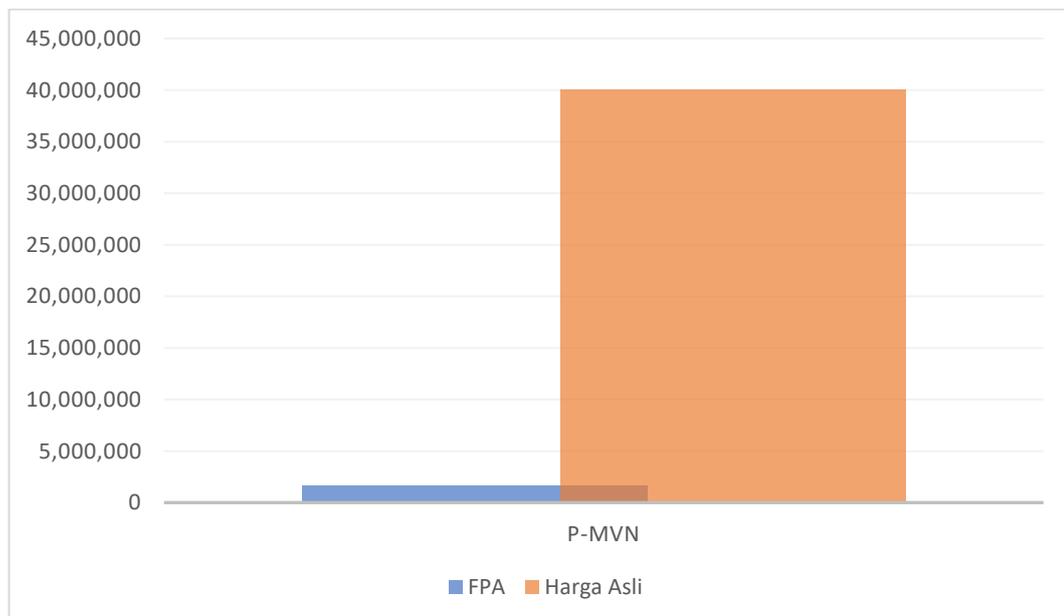
Peneliti mengelompokkan kedua proyek P-MCP dan P-MCE karena kedua proyek tersebut memiliki genre yang sama yaitu *casual*. Performa hasil estimasi menggunakan metode FPA pada proyek permainan digital bergenre casual disajikan pada Tabel 5.51.

Tabel 5.51. Gap estimasi dengan harga asli pada genre casual

No	Nama Proyek	Harga Asli (Rp)	Estimasi Harga (Rp)	Gap (%)
1	P-MCP	7,000,000	5,014,483	28.36%
2	P-MCE	5,000,000	6,809,792	36.20%

Hasil pada Tabel 5.51 menjelaskan bahwa menurut metode FPA, proyek P-MCE lebih sulit daripada proyek P-MCP karena proyek P-MCE memiliki lebih banyak tugas fungsional yaitu sebanyak 24 tugas dibanding proyek P-MCP yang hanya memiliki 17 tugas saja. Hal tersebut sangat berseberangan dengan testimoni dari tim pengembang yang mengatakan bahwa proyek P-MCP lebih sulit dari proyek P-MCE dan terbukti dari harga proyek P-MCP yang lebih tinggi.

Hasil menarik terjadi pada estimasi biaya proyek P-MVN yang bergenre *Visual Novel*. Harga estimasi yang dihasilkan metode FPA yang sebesar Rp. 1,609,587 memiliki selisih 38,390,413 (95.98 %) dari harga aslinya yang memiliki nilai sebesar Rp. 40.000.000. Perbedaan yang besar seperti yang diilustrasikan pada Gambar 5.50 membuktikan bahwa metode FPA sangat tidak cocok dengan permainan digital yang memiliki unsur *Aesthetic* yang dominan seperti pada permainan digital genre *Visual Novel*.



Gambar 5.51. Perbandingan harga pada proyek P-MVN

Perbedaan yang sangat besar tersebut tentunya sangat berbahaya apabila dilakukan pada praktik di lapangan. Berdasarkan Gambar 5.50 dapat diambil kesimpulan bahwa apabila sebuah tim menggunakan metode FPA untuk melakukan estimasi biaya permainan digital yang memiliki elemen Aesthetic yang dominan dan menggunakan hasilnya sebagai landasan, maka kemungkinan kecil proyek tersebut akan berhasil.

Hasil estimasi menggunakan metode FPA secara keseluruhan ternyata menghasilkan selisih gap yang cukup signifikan pada proyek permainan digital dengan rata-rata gap 32.28% untuk proyek bergenre casual dan bahkan hingga 95.98% untuk permainan digital bergenre visual novel. Hal tersebut dikarenakan tugas-tugas non fungsional terutama tugas yang berkaitan dengan elemen Aesthetic tidak ikut diperhitungkan. Selain itu, terdapat kasus pada proyek P-MCE yang didefinisikan lebih sulit (memiliki nilai usaha yang lebih besar) dari pada proyek P-MCP karena memiliki faktor fungsional lebih banyak. Namun pada kenyataannya, berdasarkan pendapat tim pengembang proyek tugas-tugas pada proyek P-MCP lebih sulit dari proyek P-MCE.

Berdasarkan hasil diskusi diatas, peneliti yakin bahwa mengkalibrasi nilai ukuran perangkat lunak atau *size* yang dihasilkan oleh metode FPA berdasarkan sudut pandang tim pengembang perlu dilakukan untuk dapat menghasilkan nilai estimasi dengan akurasi yang lebih baik. Dalam penelitian ini, usaha untuk mengkalibrasi nilai *size* dilakukan dengan menggunakan Metode Story Point (MSP).

#### **5.4. Estimasi biaya menggunakan metode FPA-MSP**

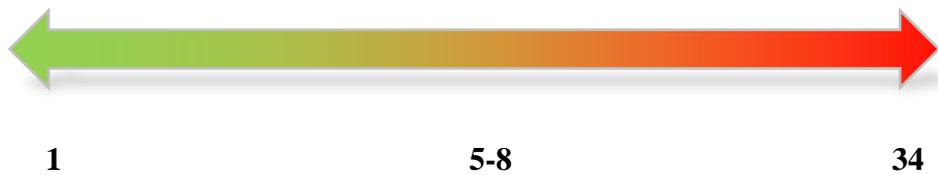
Metode Story Point (MSP) merupakan metode yang melakukan pengukuran dari sisi developer dan berbeda dengan pengukuran dengan metode FPA yang berdasarkan sudut pandang pengguna. Oleh karena penggunaan sudut pandang yang berbeda, maka MSP tidak bisa secara langsung memodifikasi perhitungan yang dilakukan dengan metode FPA. Sebelum membahas lebih lanjut tentang prosedur transformasi nilai MSP ke nilai FP terlebih dahulu peneliti bahas tentang cakupan yang dapat dilakukan MSP dan tidak dapat dilakukan oleh metode FPA.

#### **5.4.1. Metode Story Point (MSP)**

Metode Story Point (MSP) merupakan metode yang menggunakan pendekatan dari sisi developer. Hal tersebut berarti pihak developer secara subjektif menentukan tingkat kesulitan pada setiap tugas yang dikerjakan dalam membangun sebuah aplikasi. Penentuan kesulitan setiap tugas dilakukan dengan membandingkan satu tugas dengan tugas lainnya dalam proyek berdasarkan beberapa kriteria antara lain (1) prioritas tugas, (2) resiko tugas apabila terjadi kesalahan, dan (3) kompleksitas teknis yang dibutuhkan untuk menyelesaikan tugas.

##### **5.4.1.1. Nilai skala story point**

Penentuan tingkat kesulitan pada setiap tugas direpresentasikan dengan sebuah nilai. Pada penelitian ini, peneliti menggunakan nilai fibonacci sesuai dengan anjuran ISBSG (2011). Dalam beberapa penelitian (Hamouda, 2014; Raslan et al., 2015) skala fibonacci yang digunakan adalah 1, 2, 3, 5, 8, 13, dan 21. Namun dalam penelitian ini, peneliti menambahkan satu nilai skala tambahan karena terdapat unsur seni yang merupakan satu kesatuan yang tidak dapat dipecah menjadi lebih kecil lagi seperti menciptakan ilustrasi tombol dengan efek enable dan disable dengan ilustrasi tombol dengan satu tipe saja yang menurut masukan dari pengembang merupakan pekerjaan dengan skala medium tapi dengan tingkat kerumitan yang berbeda. Oleh karena itu, apabila sebelumnya nilai skala medium berdasarkan Hamouda (2014) dan Raslan et al. (2015) adalah lima, maka penelitian ini menggunakan nilai lima dan delapan sebagai skala nilai untuk mengilustrasikan kebutuhan dengan skala medium sehingga skala yang digunakan pada penelitian ini adalah 1, 2, 3, 5, 8, 13, 21, dan 34 (Meyer, 2014) supaya nilai lima dan delapan dapat menjadi nilai tengah pada skala fibonacci pada penelitian ini. Gambar 5.51 mengilustrasikan nilai skala berdasarkan warna kompleksitas (semakin merah menandakan semakin kompleks).



Gambar 5.52. Rentang bilangan fibonacci yang digunakan

#### 5.4.1.2. Depedensi dalam proses penilaian tugas

Metode MSP menggunakan preferensi dari tim pengembang dalam menentukan kompleksitas dalam setiap tugas pada sebuah proyek. Hal tersebut memungkinkan terdapat perbedaan penilaian dari satu tim pengembang dengan tim pengembang lainnya pada proyek yang sama. Perbedaan penilaian tersebut sangat dipengaruhi oleh tingkat kemampuan individu. Sebagai contoh tim pengembang A yang memiliki jam terbang selama lima tahun kemungkinan besar dapat menyelesaikan sebuah proyek X lebih cepat daripada tim B dengan jam terbang hanya dua tahun sehingga nilai kompleksitas yang diberikan oleh tim A dapat lebih kecil dari nilai kompleksitas yang diberikan oleh tim B pada proyek X.

Disisi lain, pemberian nilai berdasarkan tim yang mengembangkan perangkat lunak secara langsung dapat merepresentasikan kesulitan yang akan dihadapi tim tersebut kedepannya. Oleh karena itu, pada penelitian ini untuk mengukur nilai Story Point (SP) peneliti sangat bergantung kepada individu yang terlibat langsung pada proses pengembangan.

Dalam penelitian ini, proyek P-MCP dan P-MVN menggunakan referensi nilai dari pengembang asli proyek, sedangkan proyek P-MCE dikarenakan pengembang asli yang sulit dikontak maka peneliti memutuskan untuk menggunakan referensi orang ketiga yang memiliki pengalaman membangun proyek yang mirip secara tipe permainan dengan proyek P-MCE.

#### 5.4.1.3. Menilai kompleksitas kebutuhan menggunakan MSP

Pada sub bab 5.3.1.3 dalam Tabel 5.40 (lebih detail pada Lampiran 1A, Lampiran 1B dan Lampiran 1C) ditampilkan bahwa perhitungan menggunakan metode FPA hanya mencakup perhitungan fungsional dari sudut pandang user.

Sudut pandang user yang dimaksud adalah berdasarkan fungsi input dan output yang disediakan untuk user pada sebuah aplikasi. Lebih spesifik, metode FPA hanya berorientasi terhadap data sehingga sebuah fungsi dapat dihitung menggunakan metode FPA apabila terdapat data yang mengalir atau diolah. Hal tersebut kurang relevan dengan permainan digital secara umum karena selain sisi fungsi, permainan digital juga mementingkan sisi seni (*Aesthetic*) sebagai daya jual aplikasi. Usaha untuk melakukan pekerjaan aesthetic yang seharusnya memiliki nilai dihiraukan oleh metode FPA dan secara keseluruhan mempengaruhi nilai estimasi yang dihasilkan. Contoh tugas-tugas yang dihitung dan diabaikan pada Interface Home proyek P-MCP apabila dihitung menggunakan metode FPA dapat dilihat pada Tabel 5.52.

Tabel 5.52. Tugas yang dihitung diabaikan pada interface Home

No	Daftar kebutuhan (fungsi)	Tugas	Function Point (FP)	
			Type	Weight
1	Berpindah ke Interface Pilihan Level	Membuat fungsi untuk berpindah ke Tampilan Pilihan Level	EQ	3
		Membuat desain tombol “play” yang mengindikasikan memulai permainan	Diabaikan	
2	Pemain dapat mengaktifkan atau mematikan efek suara	Membuat fungsi untuk mengaktifkan atau mematikan efek suara	EI ILF	3 7
		Membuat desain tombol efek suara “aktif” dan “nonaktif”	Diabaikan	
3	Menampilkan iklan pada aplikasi	Membuat fungsi untuk menampilkan iklan	EQ EIF	3 5

No	Daftar kebutuhan (fungsi)	Tugas	Function Point (FP)	
			Type	Weight
4	Menghilangkan iklan dengan mengaktifkan pembelian dalam aplikasi	Membuat fungsi untuk menghilangkan iklan		
		Membuat desain tombol “no ads”	Diabaikan	
5	Menampilkan website developer	Membuat fungsi untuk membuka browser dengan url website developer.	EQ	3
		Membuat desain tombol Info	Diabaikan	
6	Memiliki visualisasi Interface Home yang menarik	Membuat desain background untuk interface Home	Diabaikan	
		Membuat desain logo untuk identitas aplikasi	Diabaikan	
		Merancang komposisi elemen-elemen tampilan visual.	Diabaikan	

Secara keseluruhan elemen *Aesthetic* sepenuhnya diabaikan apabila dihitung menggunakan metode FPA. Pengabaian itulah yang peneliti coba perbaiki menggunakan MSP karena seharusnya faktor yang dihiraukan tersebut memiliki nilai usaha yang seharusnya diperhitungkan. Untuk menghitung nilai kompleksitas setiap tugas dalam interface Home proyek P-MCP menggunakan metode MSP, langkah yang harus dilakukan adalah sebagai berikut:

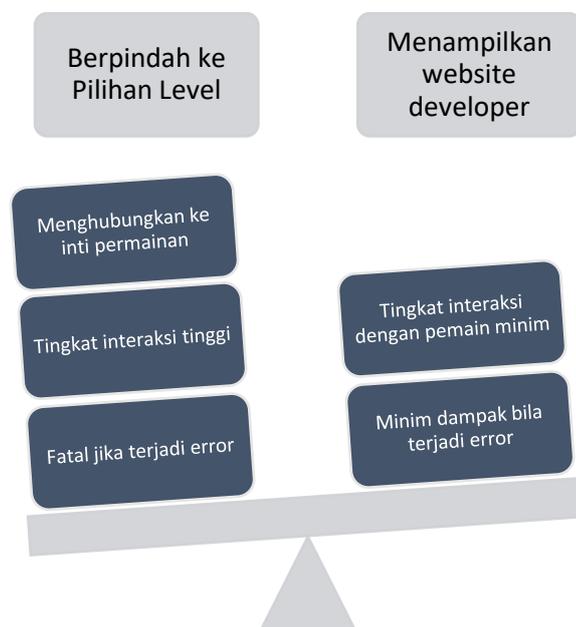
### 1. Tentukan satu kebutuhan sebagai landasan

Kebutuhan landasan merupakan kebutuhan yang digunakan sebagai alat pembandingan dengan kebutuhan lain sebagai upaya untuk memberikan nilai SP pada kebutuhan lain. Kebutuhan landasan sebaiknya dipilih dari kebutuhan dengan tingkat kesulitan medium (Jørgensen, 2013).

Sebagai contoh dalam interface Home proyek P-MCP, kebutuhan “*Berpindah ke Interface Pilihan Level*” dapat dipilih sebagai kebutuhan landasan karena memiliki tingkat resiko yang berada diantara kebutuhan “*Menampilkan website developer*” dan kebutuhan “*Visualisasi Interface Home yang menarik*”.

Gambar 5.52 merupakan contoh perbandingan tingkat kepentingan fungsi dan risiko pada kedua kebutuhan yaitu kebutuhan “*Berpindah ke Interface Pilihan Level*” dan “*Menampilkan website developer*”.

Kebutuhan “*Berpindah ke Interface Pilihan Level*” dianggap lebih penting karena memiliki rasio interaksi dengan pemain yang tinggi, setiap pengguna yang menjalankan aplikasi permainan digital P-MCP tentunya tujuan utamanya adalah memainkan permainan sehingga kebutuhan ini merupakan kebutuhan dengan tingkat interaksi dengan pengguna yang tinggi.



Gambar 5.53. Perbandingan dua kebutuhan berdasarkan resiko

Kebutuhan ini juga memiliki fungsi utama yaitu menghubungkan interface Home dengan interface Pilihan Level, yang selanjutnya interface Pilihan Level tersebut akan mengantarkan pemain ke interface Gameplay

yang merupakan inti permainan sehingga kebutuhan “*Berpindah ke Interface Pilihan Level*” menghubungkan pemain ke inti permainan. Terakhir kebutuhan ini merupakan kebutuhan kunci karena apabila mengalami masalah, pengguna tidak akan bisa mengakses interface Pilihan Level dan pada akhirnya tidak bisa mengakses inti permainan pada interface Gameplay sehingga aplikasi permainan P-MCP akan menjadi tidak berfungsi.

## **2. Tentukan nilai SP pada kebutuhan landasan**

Dalam menentukan nilai besaran SP yang subjektif, peneliti mengikuti pedoman yang dianjurkan oleh Jorgensen (2013) yaitu:

“*Perhatikan urutan estimasi, mulai dari estimasi task level medium*”.

Dengan kebutuhan landasan yang merupakan kebutuhan tingkat medium, peneliti akan membobot kebutuhan landasan dengan menggunakan skala fibonacci pada Gambar 5.51 dengan pilihan 5 atau 8 yang mengindikasikan kompleksitas medium. Peneliti memutuskan kompleksitasnya bernilai 5.

## **3. Tentukan nilai SP pada semua kebutuhan**

Dengan berbekal kebutuhan landasan dan nilainya, peneliti kemudian meminta pertimbangan dari developer yang terlibat pada proyek P-MCP untuk memberikan nilai pada semua kebutuhan. Cara untuk memberikan setiap kebutuhan sebuah nilai adalah dengan membandingkan beban tugas yang dilakukan dalam kebutuhan landasan dengan beban tugas yang dilakukan di kebutuhan lainnya.

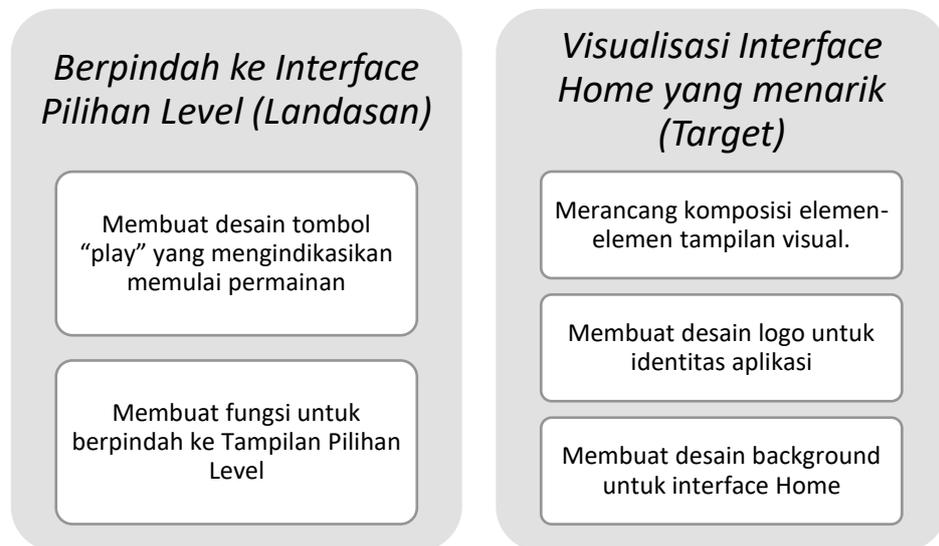
Sebagai contoh, peneliti melakukan perbandingan kebutuhan landasan dengan kebutuhan target yaitu “*Visualisasi Interface Home yang menarik*”. Untuk mendapatkan nilai SP pada kebutuhan target tersebut, tindakan yang harus dilakukan adalah bertanya kepada pengembang mengenai seberapa berat kebutuhan target dari sisi pengembang secara individu apabila dibandingkan dengan kebutuhan landasan. Tabel 5.53

merupakan contoh pertanyaan yang dapat digunakan untuk mendapatkan nilai SP pada suatu kebutuhan.

Tabel 5.53. Contoh draf pertanyaan kepada pengembang

No	Draft pertanyaan
1	Apabila kebutuhan __ (yang merupakan kebutuhan landasan) bernilai __ , maka kira-kira berapa nilai kebutuhan __ (kebutuhan target) dengan mempertimbangkan segala tugas yang dikerjakan untuk memenuhi kebutuhan tersebut?
2	Seberapa sulit pekerjaan kebutuhan __ (kebutuhan landasan) dibandingkan kebutuhan __ (kebutuhan target) ?, apabila kebutuhan __ (kebutuhan landasan) saya nilai __, maka berapa menurut Anda nilai dari kebutuhan __?

Perbandingan tugas yang dilakukan pada kebutuhan landasan dengan yang dilakukan pada kebutuhan target diilustrasikan pada Gambar 5.53.



Gambar 5.54. Perbandingan kebutuhan berdasarkan tugas

Faktor yang membuat nilai kebutuhan target lebih besar dari kebutuhan landasan adalah tugas atau pekerjaan yang dilakukan dalam kebutuhan tersebut dinilai pengembang lebih sulit seperti desain

background dan desain logo yang dianggap sebagai tantangan cukup besar bagi pengembang. Berdasarkan pertimbangan tersebut, pengembang memberikan nilai 21 SP pada kebutuhan target “*Visualisasi Interface Home yang menarik*” sehingga dapat diartikan kebutuhan tersebut memiliki kompleksitas 4.2 kali lipat (420%) dari kebutuhan landasan.

Selanjutnya, kebutuhan landasan digunakan untuk menghitung setiap kebutuhan pada Proyek P-MCP hingga semua kebutuhan proyek P-MCP memiliki nilai SP seperti yang disajikan pada Tabel 5.54.

Tabel 5.54. Nilai SP proyek P-MCP

No	Daftar kebutuhan	Referensi Tugas	SP
<b><i>Interface Home</i></b>			
1	Berpindah ke Interface Pilihan Level	Membuat fungsi untuk berpindah ke Tampilan Pilihan Level	5
		Membuat desain tombol “play” yang mengindikasikan memulai permainan	
2	Pemain dapat mengaktifkan atau mematikan efek suara	Membuat fungsi untuk mengaktifkan atau mematikan efek suara	8
		Membuat desain tombol efek suara “aktif” dan “nonaktif”	
3	Menampilkan iklan pada aplikasi	Membuat fungsi untuk menampilkan iklan	5
4	Menghilangkan iklan dengan mengaktifkan pembelian dalam aplikasi	Membuat fungsi untuk menghilangkan iklan	13
		Membuat desain tombol “no ads”	

No	Daftar kebutuhan	Referensi Tugas	SP
5	Menampilkan website developer	Membuat fungsi untuk membuka browser dengan url website developer.	5
		Membuat desain tombol Info	
6	Memiliki visualisasi Interface Home yang menarik	Membuat desain background untuk interface Home	21
		Membuat desain logo untuk identitas aplikasi	
		Merancang komposisi elemen-elemen tampilan visual.	
<b><i>Interface Pilihan Level</i></b>			
7	Memiliki visualisasi interface Pilihan Level yang menarik	Merancang konsep dan komposisi visual	21
		Mendesain background untuk interface Pilihan Level	
		Mendesain label “Level Selection” pada interface Pilihan Level	
8	Berpindah ke Interface Home	Membuat fungsi untuk berpindah ke interface Home	5
		Membuat desain tombol untuk fungsi kembali ke interface Home	

No	Daftar kebutuhan	Referensi Tugas	SP
9	Berpindah ke Interface Gameplay sesuai pilihan level	Membuat fungsi untuk berpindah ke interface Gameplay	8
10	Navigasi daftar tingkatan level	Membuat fungsi untuk menampilkan daftar tingkatan level dengan range tertentu	5
		Membuat desain tombol navigasi daftar tingkatan level	
11	Pemain dapat memainkan level tertentu apabila level sebelumnya telah diselesaikan	Membuat fungsi untuk mengunci dan membuka level.	13
		Membuat desain tombol untuk tingkatan level dengan kondisi terbuka dan terkunci	
<b><i>Interface Gameplay</i></b>			
12	Konsep level yang berbeda dalam 14 tingkatan level	Mengonsep mapping permainan (termasuk posisi kereta dimulai, alur rel, lokasi stasiun kereta)	21
		Mendesain ilustrasi background	
13	Memunculkan (spawn) objek kereta pada posisi dan waktu tertentu.	Membuat fungsi untuk memunculkan kereta api.	5
		Desain objek kereta (14 jenis warna)	

No	Daftar kebutuhan	Referensi Tugas	SP
14	Mendefinisikan dan menampilkan waktu permainan.	Mendefinisikan dan membuat fungsi untuk menampilkan dan kalkulasi waktu permainan.	13
		Desain indikator waktu	
15	Mendefinisikan dan menampilkan nyawa pemain.	Mendefinisikan dan membuat fungsi untuk menampilkan dan kalkulasi nyawa.	13
		Desain indikator nyawa pemain	
16	Pemain mendapatkan pengurangan nyawa untuk setiap kereta yang masuk ke stasiun yang salah (tidak sesuai dengan warna).	Membuat fungsi untuk kalkulasi nyawa setiap kesalahan terjadi.	13
		Desain stasiun kereta api	
17	Menentukan kecepatan pergerakan kereta.	Membuat fungsi untuk menjalankan kereta berdasarkan kecepatan tertentu.	13
18	Merubah arah rel kereta berdasarkan input pemain.	Membuat fungsi untuk menerima input pemain yang berguna untuk merubah arah rel kereta.	34
		Desain indikator sebagai fungsi merubah arah rel kereta berdasarkan input pemain	

No	Daftar kebutuhan	Referensi Tugas	SP
19	Kereta dapat berjalan sesuai dengan alur rel kereta	Membuat fungsi pathfinding untuk objek kereta	34
		Desain rel kereta api	
20	Pemain dapat menghentikan permainan sementara dan melanjutkannya kembali	Membuat fungsi untuk menghentikan permainan sementara.	13
		Desain tombol pause untuk fungsi menghentikan permainan sementara	
		Desain tulisan "pause"	
		Desain frame untuk kondisi pause	
21	Menampilkan jumlah bintang yang menandakan nilai dari pemain.	Membuat fungsi untuk mengakumulasi nilai yang didapat pemain berdasarkan jumlah kereta yang masuk ke stasiun yang benar dan menampikannya.	13
		Membuat desain bintang untuk visualisasi nilai	
		Desain frame sebagai pelat penanda.	
		Desain label	
22	Memungkinkan pemain menuju level selanjutnya	Membuat fungsi menuju tingkatan level selanjutnya	8

No	Daftar kebutuhan	Referensi Tugas	SP
	apabila memenangkan permainan	Desain tombol yang mewakili fungsi untuk memainkan permainan di tingkatan level selanjutnya.	
23	Mengulang permainan.	Membuat fungsi untuk mengulang permainan. Desain tombol yang mewakili fungsi untuk mengulang permainan	5
24	Berpindah ke Interface Pilihan Level dengan menampilkan iklan terlebih dahulu.	Membuat fungsi untuk menampilkan iklan yang kemudian berpindah ke interface Pilihan Level. Desain tombol yang mewakili fungsi untuk kembali ke interface Pilihan Level	8

#### Keterangan

 : Kebutuhan Landasan

#### Tingkat kompleksitas

	: 1 ( <i>easiest</i> )		: 8 ( <i>Intermediate</i> )
	: 2		: 13
	: 3		: 21
	: 5 ( <i>Intermediate</i> )		: 34 ( <i>Hardest</i> )

Perhitungan kompleksitas menggunakan nilai Story Point (SP) tentunya memiliki perbedaan dengan perhitungan kompleksitas

menggunakan metode FPA. Dalam metode FPA, fungsi yang diukur merupakan fungsi terkecil atau tugas nya, seperti contoh salah satu daftar kebutuhan pada interface Home proyek P-MCP pada Tabel 5.55 dimana tugas yang diukur merupakan tugas yang sesuai dengan kriteria tipe fungsi dalam metode FPA yaitu EI, EO, EQ, ILF, dan EIF. Sedangkan untuk tugas yang tidak dapat diklasifikasikan ke lima tipe fungsi tersebut akan diabaikan.

Tabel 5.55. Penilaian FPA vs MSP

Daftar kebutuhan	Tugas	FP	SP
Pemain dapat mengaktifkan atau mematikan efek suara	Membuat fungsi untuk mengaktifkan atau mematikan efek suara	EI = 3 ILF = 7	8
	Membuat desain tombol efek suara “aktif” dan “nonaktif”	Diabaikan	
<b>Total</b>		<b>10</b>	<b>8</b>

Dalam penilaian menggunakan MSP, kebutuhan diukur kompleksitasnya dengan membandingkan satu kebutuhan dengan kebutuhan lain sehingga pada Tabel 5.55 besaran kebutuhan “*Pemain dapat mengaktifkan atau mematikan efek suara*” bernilai yang bernilai 8 memiliki artinya kebutuhan tersebut memiliki kompleksitas medium bila dibandingkan dengan kebutuhan lainnya dalam proyek P-MCP.

Namun yang perlu diperhatikan, untuk menghasilkan nilai kompleksitas 8 pada MSP dalam sebuah kebutuhan, seluruh tugas yang terkait dalam kebutuhan tersebut akan ikut dipertimbangkan sebelum menghasilkan sebuah keluaran nilai SP. Sedangkan apabila kebutuhan tersebut diukur menggunakan metode FPA, maka FPA secara tegas melakukan filter pada setiap tugas dalam kebutuhan tersebut berdasarkan pada tipe fungsi yang didefinisikan oleh metode FPA sehingga tugas yang

sesuai dengan kelima tipe fungsi tersebut yang diberi nilai dan yang tidak sesuai akan diabaikan.

Proses pemberian nilai SP kepada seluruh kebutuhan dilakukan juga pada kedua proyek lainnya (lihat pada Lampiran 2A dan Lampiran 2B). Tabel 5.56 menyajikan nilai akumulasi SP pada ketiga proyek yang digunakan dalam penelitian ini yaitu P-MCP, P-MCE dan P-MVN.

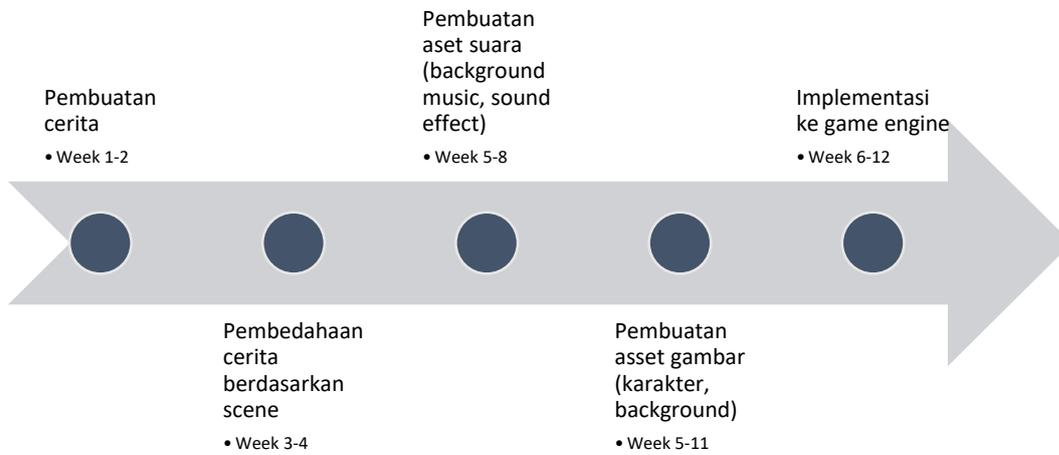
Tabel 5.56. Akumulasi nilai SP pada ketiga proyek

No	Nama Proyek	Jumlah kebutuhan	Total Story Point (SP)
1	P-MCP	24	302
2	P-MCE	37	285
3	P-MVN	8	155

#### 5.4.1.4. Mendapatkan rasio produktivitas

Kemampuan individu dalam sebuah tim berperan penting dalam seberapa cepat tim membangun perangkat lunak karena semakin berpengalaman dan ahli seseorang maka semakin cepat individu tersebut dapat menyelesaikan tugas. Dalam metode MSP tingkat produktivitas disebut dengan nilai *velocity*.

Nilai *velocity* masing-masing tim yang menangani proyek P-MCP, P-MCE, dan P-MVN didapatkan berdasarkan perkiraan setiap tim untuk menyelesaikan proyek terkait. Sebagai contoh, tim proyek P-MVN memiliki rencana untuk menyelesaikan proyek P-MVN dalam kurun waktu tiga bulan seperti yang diilustrasikan pada Gambar 5.55.



Gambar 5.55. Rencana pengembangan proyek P-MVN

Tingkat produktivitas tim proyek P-MVN dapat diketahui menggunakan metode MSP dapat dengan cara mendapatkan nilai rasio rata-rata story point setiap iterasinya, dalam kasus proyek P-MVN iterasi yang digunakan adalah setiap dua minggu sekali. Informasi iterasi tersebut digunakan peneliti untuk menyusun rancangan pengerjaan untuk setiap kebutuhan (lihat di Tabel 5.31) proyek P-MVN dengan mengacu kepada roadmap rencana pengembangan proyek pada Gambar 5.55 dan menghasilkan rancangan seperti pada Tabel 5.57.

Tabel 5.57. Rancangan pengerjaan proyek P-MVN

Iterasi (2 weeks)	Jumlah Story Per Iteration	Jumlah Story Point	
		per Story	per literasi
1	1 (kebutuhan 5)	34	34
2	1 (kebutuhan 8)	34	34
3	1 (kebutuhan 1)	21	21
4	1 (kebutuhan 3)	21	21
5	1 (kebutuhan 4)	21	21
6	3 (kebutuhan 2,6,7)	8,8,8	24
<b>Total</b>	8		155

Iterasi (2 weeks)	Jumlah Story Per Iteration	Jumlah Story Point	
		per Story	per literasi
Rerata	1.33		25.83

Nilai velocity didapatkan berdasarkan jumlah akumulasi nilai SP pada seluruh proyek dibagi dengan total iterasi, dalam hal ini nilai velocity atau produktivitas tim pengembang P-MVN adalah:

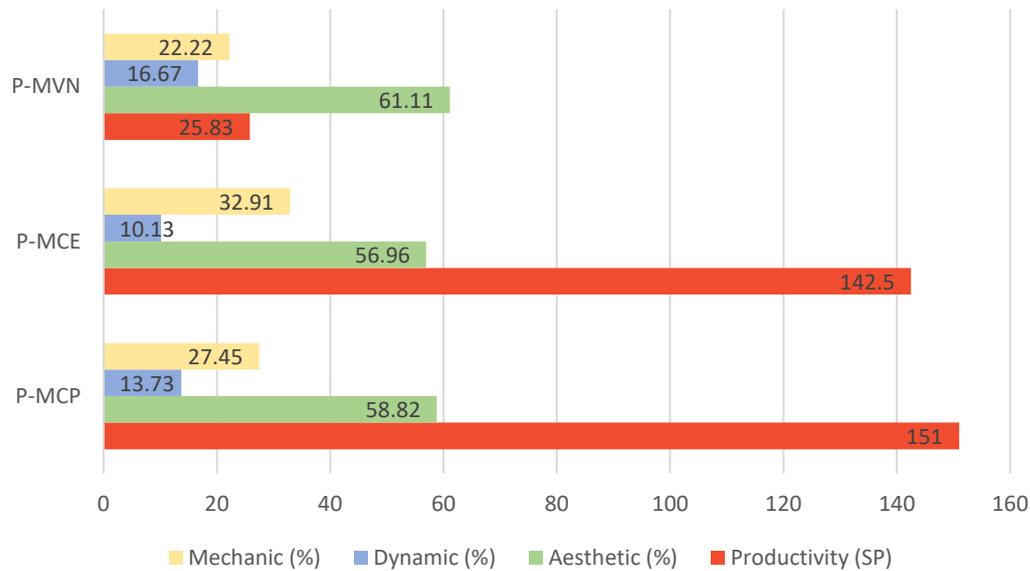
$$\begin{aligned}
 Velocity &= \frac{Total\ SP}{\sum\ iteration} \\
 &= \frac{155}{6} \\
 &= 25.83
 \end{aligned}$$

Berdasarkan perhitungan tersebut maka didapatkan bahwa nilai produktivitas dari tim proyek P-MVN adalah sebesar 25.83 SP atau bila diartikan adalah dalam dua minggu, tim proyek P-MVN mengerjakan kebutuhan dengan bobot rata-rata sebesar 25.83 SP. Dengan menggunakan langkah yang sama peneliti melakukan rancangan pengerjaan pada kedua proyek lainnya yang dapat dilihat pada Lampiran 3.A dan Lampiran 3.B sehingga diketahui produktivitas masing-masing tim proyek seperti pada Tabel 5.58.

Tabel 5.58. Produktivitas tim proyek

No	Nama Proyek	Jumlah Anggota tim	Produktivitas
1	P-MCP	2 orang	151
2	P-MCE	2 orang	142.5
3	P-MVN	8 orang	25.83

Apabila produktivitas ketiga proyek P-MCP, P-MCE, dan P-MVN dibandingkan dengan komponen penyusun permainan digital yaitu mechanic, dynamic dan aesthetic maka akan menjadi seperti pada Gambar 5.56.



Gambar 5.56. Karakteristik proyek vs produktivitas

Berdasarkan Tabel 5.58 dan Gambar 5.56 dapat dilihat bahwa proyek P-MVN yang memiliki faktor Aesthetic yang lebih banyak dari proyek P-MCP dan P-MCE memiliki tingkat produktivitas yang sangat kecil. Disamping itu, hal menarik lainnya adalah nilai produktivitas 25.83 (5.68 kali lipat lebih kecil dari rata-rata produktivitas kedua proyek) didapatkan dengan jumlah anggota tim yang empat kali lipat lebih besar. Peneliti memiliki dugaan tentang bagaimana hal tersebut dapat terjadi yaitu:

1. Elemen aesthetic pada proyek P-MVN memiliki kompleksitas yang sangat jauh berbeda dibandingkan kedua proyek lainnya dan belum dapat dieksplorasi lebih jauh dalam penelitian ini, atau
2. Jumlah tim yang terlalu besar dapat menyebabkan penurunan performa tim secara keseluruhan sesuai dengan pendapat ISBSG (2011).

#### 5.4.2. Memodifikasi besaran nilai effort dengan MSP

Seperti yang telah disinggung pada sub bab 5.4 , metode Story Point (MSP) tidak bisa secara langsung memodifikasi perhitungan yang dilakukan dengan metode FPA karena perbedaan cara perhitungan. Namun disini lain peneliti yakin

bahwa MSP dapat menjadi komplementer bagi metode FPA yang lemah dari sisi non fungsional dan proses teknis. Lebih khusus, dalam permainan digital, peneliti meyakini bahwa metode MSP dapat digunakan untuk mengukur elemen *aesthetic* yang 100% diabaikan oleh FPA.

Dalam penelitian ini, peneliti menggunakan pendekatan melalui durasi pengerjaan yang memiliki satuan FP/jam sebagai cara untuk mengarahkan nilai MSP supaya dapat memodifikasi ukuran besaran perangkat lunak permainan digital. Langkah-langkah yang dilakukan peneliti adalah (1) mengkonversi nilai produktivitas per iterasi (SP/iterasi) menjadi produktivitas per jam (SP/jam), sehingga dalam tahapan ini peneliti memiliki nilai produktivitas dengan satuan SP/jam dan (2) melakukan persamaan linier dengan acuan salah satu kebutuhan untuk mendapatkan nilai FP/jam.

#### **5.4.2.1. Mendapatkan produktivitas per jam**

Dalam proses ini, nilai produktivitas setiap proyek yang telah dihitung pada Tabel 5.58 yang memiliki satuan story point per iterasi akan dipersempit skalanya menjadi story point per harinya dengan menggunakan persamaan (4) yaitu

$$VEL_d = \frac{Velocity}{i}$$

Dimana  $VEL_d$  merupakan nilai yang memiliki satuan story point per hari, sedangkan  $Velocity$  merupakan nilai produktivitas atau rata-rata story point per iterasi dan  $i$  merupakan total hari dalam sekali iterasi. Dengan menggunakan persamaan (4) maka nilai story point per hari atau  $VEL_d$  untuk proyek P-MCP adalah:

$$\begin{aligned}VEL_d &= \frac{Velocity}{i} \\ &= \frac{151}{10} \\ &= 15.1\end{aligned}$$

Berdasarkan Lampiran 3.A, tim proyek P-MCP menentukan durasi iterasi yaitu dua minggu sekali maka semula didapat nilai variabel  $i$  adalah 14, akan tetapi berdasarkan peraturan dalam undang-undang Depnaker no. 13 tahun 2003 pasal 77 ayat 1 yang mengatur jam kerja bahwa maksimal kerja dalam satu minggu adalah 40 jam yang dibagi dalam lima atau enam hari kerja. Dalam penelitian ini, peneliti memilih lima hari jam kerja karena dinilai lebih relevan dengan hari kerja studio game saat ini. Hasilnya nilai variabel  $i$  yang semula 14 peneliti revisi menjadi 10 dan menghasilkan nilai produktivitas per harinya adalah 15.1.

Perhitungan produktivitas per hari menggunakan persamaan (4) juga dilakukan pada kedua proyek lainnya yaitu P-MCE dan P-MVN sehingga diperoleh hasil seperti pada Tabel 5.59

Tabel 5.59. Produktivitas per hari tim pengembang

No	Nama Proyek	Produktivitas per hari (SP/hari)
1	P-MCP	15.1
2	P-MCE	14.25
3	P-MVN	2.583

Selanjutnya, cakupan nilai story point per hari akan kembali dipersempit menjadi story point per hari menggunakan persamaan (5) yaitu:

$$VEL_h = \frac{VEL_d}{8}$$

Dimana variabel  $VEL_h$  merupakan nilai story Point per jam,  $VEL_d$  merupakan nilai story point per hari dimana pada penelitian ini telah didapatkan pada Tabel 5.59. Nilai konstanta 8 pada persamaan (5) merupakan nilai yang mengacu kepada aturan dari Depnaker no. 13 tahun 2003 pasal 77 ayat 1 yang mengatur jam kerja 8 jam per hari untuk 5 hari kerja. Dengan menggunakan persamaan (5) maka nilai produktivitas per hari atau  $VEL_h$  dari proyek P-MCP adalah:

$$\begin{aligned}
VEL_h &= \frac{VEL_d}{8} \\
&= \frac{15.1}{8} \\
&= 1.888
\end{aligned}$$

Langkah Persamaan (5) juga diterapkan pada kedua proyek lainnya sehingga menghasilkan nilai story point per jam pada ketiga proyek yaitu P-MCP, P-MCE dan P-MVN seperti yang disajikan pada Tabel 5.60.

Tabel 5.60. Produktivitas per jam tim pengembang

No	Nama Proyek	Produktivitas per jam (SP/jam)
1	P-MCP	1.888
2	P-MCE	1.781
3	P-MVN	0.323

#### 5.4.2.2. Mendapatkan nilai PDR

Berdasarkan pengertian ISBSG (2011), Product Delivery Rate (PDR) adalah seberapa sepat tim harus menyelesaikan sebuah fitur yang memiliki satuan function point per jam (FP/jam) sehingga nilai PDR yang besar mengindikasikan usaha yang besar. Oleh karena peneliti hanya memiliki nilai yang memiliki satuan story point per jam (SP/jam), maka peneliti menggunakan persamaan linear (6.1) dan (6.2) yaitu:

$$\begin{aligned}
X_H &= \frac{X_S}{VEL_H} \\
PDR &= \frac{X_H}{X_{FP}}
\end{aligned}$$

Dimana :

PDR : Project Delivery Rate (fp/jam)

$X_H$  : Total estimasi penyelesaian kebutuhan x (jam)

$X_{FP}$  : Function point pada kebutuhan x (fp)

$X_H$  : Total estimasi penyelesaian kebutuhan x (jam)

$X_S$  : Story point pada kebutuhan x (SP)

$VEL_H$  : Story Point / jam

Persamaan linear (6.1 dan 6.2) menggunakan salah satu kebutuhan dalam proyek permainan digital sebagai medium. Pada penelitian ini, ketiga medium yang digunakan peneliti merupakan kebutuhan yang tugasnya merupakan kebutuhan fungsional supaya dari kedua sisi perhitungan FPA dan MSP memiliki nilai. Tabel 5.61 merupakan kebutuhan yang digunakan sebagai medium untuk persamaan (6.1) dan (6.2).

Tabel 5.61. Referensi kebutuhan sebagai medium perhitungan

No	Nama Proyek	Medium kebutuhan	Nilai FP	Nilai SP
1	P-MCP	Berpindah ke Interface Gameplay sesuai pilihan level (kebutuhan no 9)	3	8
2	P-MCE	Setiap tingkatan level memiliki catatan waktu terbaik (kebutuhan no 11)	4	8
3	P-MVN	Pemain dapat mengetahui update cerita terbaru (kebutuhan no 3)	4	21

Berdasarkan Tabel 5.61, maka untuk mendapatkan nilai PDR proyek P-MCP adalah:

$$\begin{aligned} X_H &= \frac{X_S}{VEL_H} \\ &= \frac{8}{1.888} \\ &= 4.237 \end{aligned}$$

Kemudian dilanjutkan dengan;

$$PDR = \frac{X_H}{X_{FP}}$$

$$= \frac{4.237}{3}$$

$$= 1.412$$

Sehingga didapat nilai PDR pada proyek P-MCP adalah 1.412. Langkah persamaan (6.1) dan persamaan (6.2) juga dilakukan untuk proyek P-MCE dan P-MVN sehingga menghasilkan nilai PDR untuk ketiga proyek seperti Tabel 5.62.

Tabel 5.62. Nilai PDR pada ketiga proyek

No	Nama Proyek	Product Delivery Rate (PDR)
1	P-MCP	1.412
2	P-MCE	1.123
3	P-MVN	16.254

#### 5.4.2.3. Mendapatkan nilai effort baru

Untuk mendapatkan nilai effort baru, peneliti menggunakan persamaan (3) yang telah digunakan pada sub bab 5.3.2. Perbedaannya adalah nilai PDR yang pada sub bab 5.3.2 diinisialisasi sebagai 1 namun pada proses ini nilai PDR diinisialisasi menggunakan nilai pada Tabel 5.62 yang didapat melalui proses perhitungan dengan persamaan (6.1) dan (6.2). Oleh karena itu nilai effort baru pada proyek P-MCP adalah:

$$Effort = Size * PDR * 8.2$$

$$= 81 * 1.412 * 8.2$$

$$= 937.85$$

Perhitungan effort baru dengan menggunakan nilai PDR digunakan kepada dua proyek lainnya sehingga nilai effort baru untuk ketiga proyek menjadi seperti Tabel .

Tabel 5.63. Nilai effort baru

No	Nama Proyek	FP	Effort Rate	PDR	Effort Proyek
1	P-MCP	81	8.2	1.412	937.85

No	Nama Proyek	FP	Effort Rate	PDR	Effort Proyek
2	P-MCE	110	8.2	1.123	1,012.946
3	P-MVN	26	8.2	16.254	3,465.353

Terlihat pada Tabel 5.63 nilai usaha pada proyek P-MVN yang semula 213.2 apabila menggunakan perhitungan metode FPA saja menjadi meningkat drastis hingga 3,465.353 apabila dihitung menggunakan metode FPA-MSP. Hal tersebut membuktikan bahwa metode MSP berhasil memasukkan faktor non-fungsional khususnya aesthetic pada permainan digital ke nilai usaha.

#### 5.4.3. Perhitungan effort per aktivitas (FPA-MSP)

Tahapan Perhitungan effort per aktivitas (FPA-MSP) memiliki aktivitas yang sama dengan sub bab 5.3.3. Hal yang membedakannya adalah nilai effort yang digunakan adalah nilai effort hasil modifikasi dengan MSP pada Tabel 5.63 sehingga hasilnya seperti yang disajikan pada Tabel 5.64.

Tabel 5.64. Nilai usaha pada setiap aktivitas proyek (FPA-MSP)

No	Activity	Effort distribution (%)	Project Name		
			P-MCP	P-MCE	P-MVN
<i>Software development phase</i>					
1	Requirements	1.17%	10.973	11.851	40.545
2	Specifications	6.75%	63.305	68.374	233.911
3	Design	5.57%	52.238	56.421	193.020
4	Implementation	55.65%	521.914	563.704	1928.469
5	Integration Testing	6.42%	60.210	65.031	222.476
6	Acceptance & deployment	5.6%	52.520	56.725	194.060
<i>Ongoing activities &amp; quality and testing</i>					
7	Project management	2.55%	23.915	25.830	88.367
8	Configuration management	3.58%	33.575	36.263	124.060

No	Activity	Effort distribution (%)	Project Name		
			P-MCP	P-MCE	P-MVN
9	Quality assurance	0.66%	6.190	6.685	22.871
10	Documentation	9.76%	91.534	98.864	338.218
11	Training & support	0.6%	5.627	6.078	20.792
12	Evaluation & testing	1.67%	15.662	16.916	57.871
<b>Total</b>		100%	937.85	1,012.946	3,465.353

#### 5.4.4. Perhitungan biaya per aktivitas (FPA-MSP)

Tahapan perhitungan biaya per aktivitas (FPA-MSP) memiliki proses yang identik dengan sub bab 5.3.4. Hal yang membedakan adalah pada tahapan ini adalah proses perhitungan nilai biaya per aktivitas menggunakan hasil nilai usaha per aktivitas pada ketiga proyek yang didapatkan pada Tabel 5.64 sehingga hasil dari proses perhitungan biaya per aktivitas untuk ketiga proyek memiliki nilai seperti pada Tabel 5.65.

Tabel 5.65. Pay rate per aktivitas permainan digital (FPA-MSP)

No	Aktivitas	Pay rate	Project Name		
			P-MCP	P-MCE	P-MVN
<i>Software development phase</i>					
1	Requirements	12,883	141,363	152,682	522,336
2	Specifications	12,883	815,557	880,860	3,013,480
3	Design	12,883	672,985	726,873	2,486,679
4	Implementation	6,351	3,314,673	3,580,087	12,247,706
5	Integration Testing	6,351	382,394	413,013	1,412,943
6	Acceptance & deployment	6,351	333,552	360,260	1,232,474
<i>Ongoing activities &amp; quality and testing</i>					

No	Aktivitas	Pay rate	Project Name		
			P-MCP	P-MCE	P-MVN
7	Project management	12,883	308,099	332,769	1,138,426
8	Configuration management	6,351	213,235	230,309	787,903
9	Quality assurance	12,883	79,743	86,129	294,651
10	Documentation	6,351	581,333	627,882	2,148,025
11	Training & support	6,351	35,738	38,599	132,051
12	Evaluation & testing	12,883	201,775	217,931	745,557
<b>Total</b>			7,080,447	7,647,396	26,162,231

#### 5.4.5. Analisa gap hasil estimasi metode FPA-MSP

Berdasarkan hasil dari Tabel 5.65, apabila semua biaya per aktivitas pada suatu perangkat lunak diakumulasi maka nilai akumulasi tersebut merupakan nilai estimasi biaya untuk perangkat lunak terkait seperti yang disajikan pada Tabel 5.66. Sehingga nilai estimasi biaya untuk proyek P-MCP adalah Rp. 7,080,447, P-MCE dengan nilai estimasi Rp. 7,647,396, dan estimasi P-MVN bernilai Rp. 26,162,231.

Tabel 5.66. Estimasi biaya menggunakan FPA-MSP

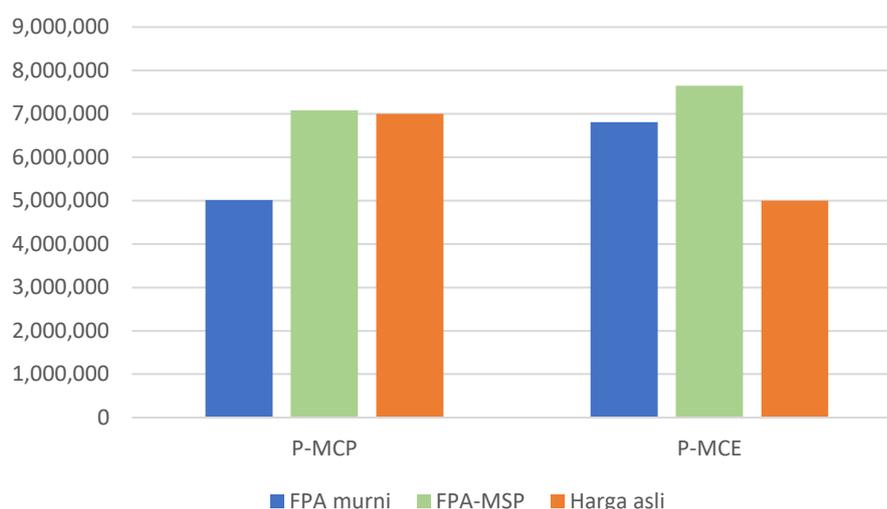
No	Nama Proyek	Estimasi Biaya (Rp)
1	P-MCP	7,080,447
2	P-MCE	7,647,396
3	P-MVN	26,162,231

Selanjutnya peneliti akan menganalisa selisih gap harga estimasi menggunakan metode FPA-MSP dengan harga asli pada ketiga proyek seperti yang disajikan pada Tabel 5.67.

Tabel 5.67. Gap estimasi metode FPA-MSP dengan harga asli

No	Nama Proyek	Harga Asli (Rp)	Estimasi Harga (Rp)	Gap (%)
1	P-MCP	7,000,000	7,080,447	1.15%
2	P-MCE	5,000,000	7,647,396	52.95%

Berdasarkan Tabel 5.67, estimasi menggunakan metode FPA-MSP pada genre permainan casual menghasilkan keluaran yang menarik. Disatu sisi pada proyek P-MCP, estimasi menggunakan metode FPA-MSP meningkatkan akurasi dengan selisih gap estimasi senilai 1.15% dibandingkan dengan estimasi menggunakan FPA murni yang memiliki selisih gap sebesar 28.36% seperti yang diilustrasikan pada Gambar 5.57.



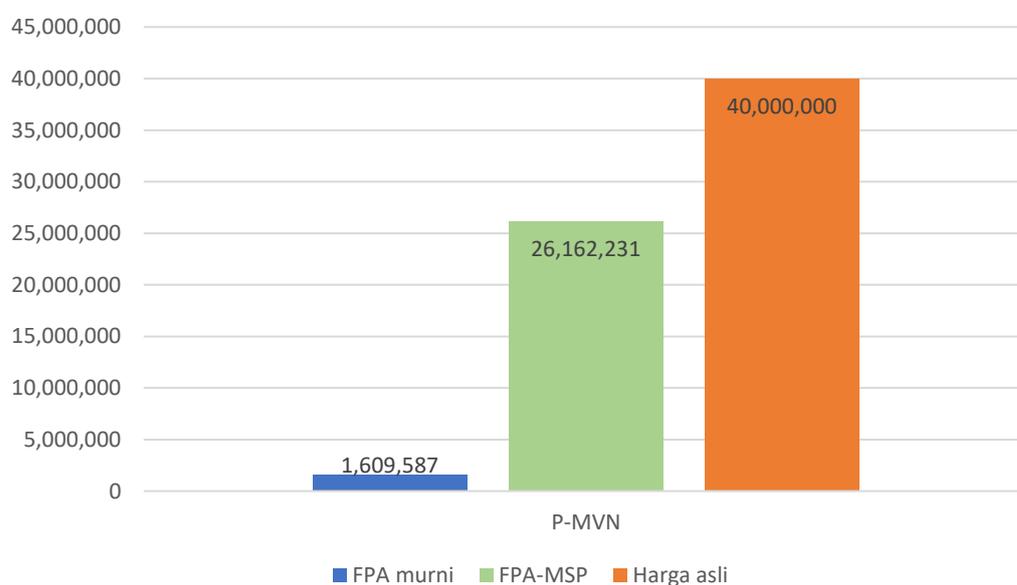
Gambar 5.57. Perbandingan estimasi FPA, FPA-MSP, dan harga asli

Disisi lain nilai estimasi yang dihasilkan metode FPA-MSP pada proyek P-MCE menghasilkan gap yang justru lebih besar yaitu 52.95% dibanding dengan metode FPA murni yang “hanya” memiliki gap sebesar 36.20%. Mengapa hal tersebut dapat terjadi ?

Disini peneliti jabarkan kembali bahwa metode MSP merupakan metode yang subjektif, sehingga nilai yang diberikan oleh satu individu dengan individu lain akan memiliki kemungkinan perbedaan yang besar. Pada proyek P-MCP, penilaian kebutuhan menggunakan metode MSP dilakukan oleh individu yang

terlibat langsung pada proyek tersebut sedangkan pada proyek P-MCE nilai kebutuhan didapatkan dari individu yang memiliki pengalaman mengerjakan aplikasi yang serupa namun bukan anggota tim yang membangun proyek P-MCE (seperti yang telah dijelaskan pada sub bab 5.4.1.2). Berdasarkan hasil yang didapat sesuai Gambar 5.57 maka didapatkan kesimpulan bahwa metode estimasi menggunakan FPA-MSP akan relevan jika estimasi dilakukan oleh tim yang terlibat langsung dalam pembangunan sebuah perangkat lunak.

Pada permainan digital yang memiliki genre visual novel, hasil keluaran metode FPA-MSP juga memiliki hasil yang jauh lebih baik dari estimasi yang dihasilkan oleh metode FPA murni dibuktikan pada Gambar 5.58.



Gambar 5.58. Perbandingan hasil estimasi FPA, FPA-MSP dan harga asli

Secara presentase, metode FPA-MSP memiliki tingkat akurasi yang jauh lebih baik dibandingkan metode FPA murni dengan selisih gap sebesar 34.6 % dibanding 95.98% dan membuktikan bahwa pentingnya peran faktor aesthetic yang mendominasi kebutuhan permainan digital hingga 58.96 % untuk dipertimbangkan nilai effortnya.

Secara keseluruhan pada tiga permainan digital yang digunakan pada penelitian ini yaitu P-MCP,P-MCE,dan P-MVN, akurasi estimasi metode FPA-MSP memiliki selisih gap sebesar 29.57% , jauh lebih baik dari estimasi yang

dihasilkan metode FPA murni dengan selisih gap sebesar 53.51%. hasil tersebut membuktikan bahwa faktor non-fungsional terutama elemen aesthetic perlu ikut dipertimbangkan ketika menghitung nilai effort pada permainan digital.

#### **5.4.6. Hasil FPA-MSP vs Harga Asli**

Nilai hasil dari metode FPA-MSP memiliki rata-rata selisih gap sebesar 29.57% jika dibandingkan harga asli meskipun telah mempertimbangkan faktor aesthetic yang dominan pada permainan digital, peneliti akan menjelaskan mengapa hal tersebut dapat terjadi.

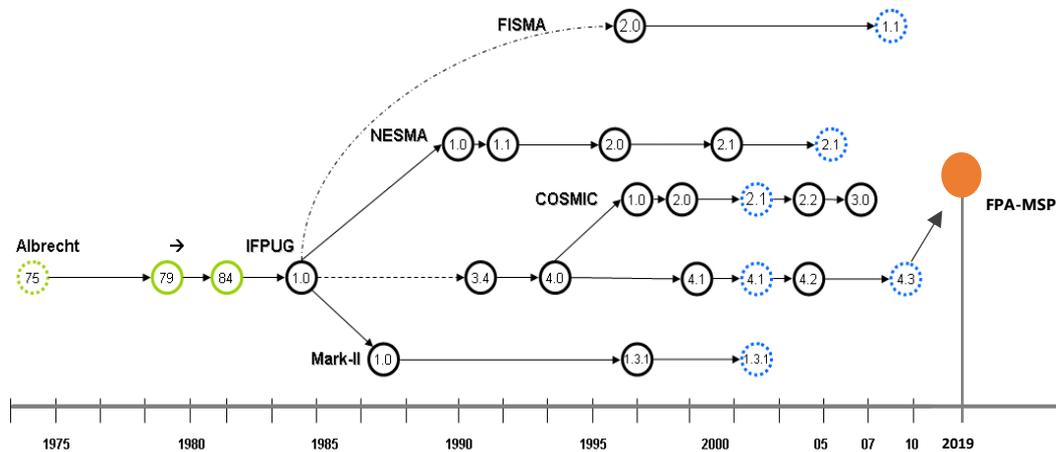
Metode FPA-MSP menggunakan distribusi effort menggunakan penelitian dari Saleh (2011) serta Primandari dan Sholiq (2015). Distribusi effort mengindikasikan seberapa besar nilai effort kontribusi oleh individu pada fase pengembangan dalam sebuah proyek permainan digital. Sehingga terdapat kemungkinan pemotongan gaji berdasarkan besar kecilnya kontribusi yang dilakukan oleh individu dan kebutuhan dalam aplikasi terkait.

Hal tersebut berbanding terbalik dengan fakta lapangan untuk gaji para pengembang perangkat lunak yang dihitung berdasarkan masa bakti sebuah individu, semakin lama individu bekerja maka semakin besar gaji individu walaupun sebenarnya effort yang dilakukan individu tersebut minim.

Metode FPA-MSP mempertimbangkan unsur fungsional, kemampuan tim, dan kontribusi yang dilakukan untuk mendapatkan hasil estimasi. Dengan mempertimbangkan ketiga unsur tersebut nilai hasil estimasi metode FPA-MSP merupakan nilai standar terkecil supaya sebuah tim pengembang tidak rugi apabila mengerjakan proyek perangkat lunak. Keuntungan utama yang didapatkan apabila menggunakan metode FPA-MSP untuk mengestimasi biaya adalah tim pengembang dapat menjustifikasi biaya tersebut berdasarkan pekerjaan mereka dengan jelas dan terstruktur serta dapat menyusun seberapa besar tim pengembang yang terlibat, selain itu tim pengembang juga meningkatkan daya saing mereka dengan kompetitor dengan menawarkan biaya yang lebih rendah namun tidak sampau merugikan tim pengembang dalam proyek permainan digital terkait.

### 5.4.7. FPA-MSP sebagai solusi metode estimasi biaya

Sudah hampir 40 tahun lamanya sejak Allan J. Albrecht (1979) menemukan metode FPA sebagai solusi perhitungan usaha. Pada 10 tahun awal sejak ditemukan metode FPA menjadi sangat populer dan menjadi standar pengukuran perangkat lunak di berbagai industri. Namun perlu disadari bahwa jaman berubah, dan metodologi serta teknologi juga ikut berkembang. FPA sebagai sebuah metode pelopor ukuran perangkat lunak ternyata tidak dapat mengikuti perkembangan tersebut. Ketidakmampuan metode FPA mengikuti jaman mengakibatkan munculnya banyak varian dari FPA yang bertujuan untuk menyempurnakan metode FPA asli seperti yang diilustrasikan pada Gambar 5.59.



Gambar 5.59. FPA-MSP sebagai varian metode FPA

Tahun 2019, dengan semakin meningkatnya kebutuhan pada estimasi biaya pada permainan digital di kalangan investor. Peneliti melihat bahwa metode FPA saja belum cukup mampu untuk menjadi solusi akan kebutuhan tersebut. Terbukti dengan rata-rata hasil selisih gap yang lebih dari 50%, mengakibatkan estimasi biaya menggunakan metode FPA memiliki resiko biaya yang jauh dengan aslinya. Hal tersebut disebabkan metode FPA secara aktif mengabaikan kebutuhan non-fungsional lebih khususnya kepada kebutuhan seni (Aesthetic) pada aplikasi permainan digital. Oleh karena itu, peneliti memberikan alternatif solusi yaitu

menggunakan metode gabungan FPA dengan metode Story Point (MSP) yang dapat mengukur sisi seni yang memiliki dominasi rata-rata 58.96% dari keseluruhan pembangunan aplikasi permainan digital.

[HALAMAN INI SENGAJA DIKOSONGKAN]

## **BAB 6**

### **KESIMPULAN DAN SARAN**

Bab ini berisi kesimpulan dan saran untuk penelitian yang akan datang. Sehingga diharapkan penelitian ini dapat terus dikembangkan kedepannya.

#### **6.1. Kesimpulan**

Kesimpulan yang didapat dari penelitian ini adalah sebagai berikut:

1. Perangkat lunak permainan digital memiliki porsi non-fungsional yang besar terutama elemen aesthetic rata-rata 58.96% dari keseluruhan kebutuhan perangkat lunak permainan digital.
2. Metode FPA sangat tidak cocok digunakan untuk mengukur estimasi biaya permainan digital, terbukti dengan terdapatnya selisih gap sebesar 95.98% pada permainan digital yang memiliki 61.11% unsur aesthetic pada kebutuhan pengembangannya.
3. Metode Function Point memerlukan perombakan dalam menghadapi tantangan pengembangan perangkat lunak pada masa ini. Pada kasus permainan digital, pernyataan itu terbukti karena hasil modifikasi FPA dengan MSP yang memiliki rata-rata gap sebesar 29.57%, jauh lebih baik dibandingkan hasil estimasi FPA murni yang memiliki rata-rata gap sebesar 53.51%. Hal ini sejalan dengan penelitian terdahulu yang memiliki fokus sama yaitu memodifikasi metode FPA (Dewi et al., 2017; Putri and Subriadi, 2018; Xia et al., 2008).
4. MSP terbukti dapat mengangkat performa akurasi estimasi biaya hingga 1.15% apabila nilai story point dinilai oleh individu yang terlibat langsung dalam proses pengembangan perangkat lunak.
5. Namun sebaliknya, MSP sangat tidak dianjurkan apabila menggunakan referensi dari individu yang tidak terlibat langsung dalam pengembangan proyek. Hal tersebut terbukti pada proyek P-MCE yang justru memiliki hasil estimasi biaya yang lebih buruk dengan gap sebesar 52.95% dibanding metode FPA murni yang hanya 36.20%

## **6.2. Saran**

Saran yang dapat peneliti berikan dalam penelitian ini adalah sebagai berikut:

1. Perlu adanya standarisasi gaji bagi individu yang bekerja dalam lingkup aplikasi permainan digital supaya nilai estimasi bisa menjadi relevan apabila dibandingkan dari satu studio permainan digital dengan studio lainnya.
2. Metode FPA-MSP perlu diuji lagi pada varian permainan digital untuk melihat seberapa besar konsistensinya dalam menghasilkan estimasi biaya yang akurat.
3. Pemilihan informan pada saat menggunakan metode MSP perlu diperhatikan karena dapat mengakibatkan nilai estimasi menjadi tidak akurat. Pilih informan yang terlibat langsung pada pengembangan proyek terkait.

## DAFTAR PUSTAKA

- Abdullah, N.A.S., Rusli, N.I.A., Ibrahim, M.F., 2014. Mobile game size estimation: COSMIC FSM rules, UML mapping model and Unity3D game engine, in: 2014 IEEE Conference on Open Systems (ICOS). Presented at the 2014 IEEE Conference on Open Systems (ICOS), IEEE, Subang, Malaysia, pp. 42–47. <https://doi.org/10.1109/ICOS.2014.7042407>
- Abdullah, N.A.S., Rusli, N.I.A., Ibrahim, M.F., 2013. A case study in COSMIC functional size measurement: Angry Bird Mobile Application, in: 2013 IEEE Conference on Open Systems (ICOS). Presented at the 2013 IEEE Conference on Open Systems (ICOS), IEEE, Sarawak, Malaysia, pp. 139–144. <https://doi.org/10.1109/ICOS.2013.6735063>
- Albrecht, A.J., 1979. Measuring application development productivity, in: Proc. Joint Share, Guide, and IBM Application Development Symposium. IBM Corporation, Montara, CA, USA, pp. 83–92.
- Albrecht, A.J., Gaffney, J.E., 1983. Software function, source lines of code, and development effort prediction: a software science validation. *IEEE Trans. Softw. Eng.* 639–648.
- Bagriyanik, S., Karahoca, A., 2016. Automated COSMIC Function Point measurement using a requirements engineering ontology. *Inf. Softw. Technol.* 72, 189–203. <https://doi.org/10.1016/j.infsof.2015.12.011>
- Batchelor, J., 2018. Games industry generated \$108.4bn in revenues in 2017 [WWW Document]. GamesIndustry.biz. URL <https://www.gamesindustry.biz/articles/2018-01-31-games-industry-generated-usd108-4bn-in-revenues-in-2017> (accessed 11.17.18).
- Bilgaiyan, S., Mishra, S., Das, M., 2016. A Review of Software Cost Estimation in Agile Software Development Using Soft Computing Techniques, in: 2016 2nd International Conference on Computational Intelligence and Networks (CINE). Presented at the 2016 2nd International Conference on Computational Intelligence and Networks (CINE), pp. 112–117. <https://doi.org/10.1109/CINE.2016.27>
- Blow, J., 2004. Game Development: Harder Than You Think. *Queue* 1, 28–37. <https://doi.org/10.1145/971564.971590>
- Bock, D.B., Klepper, R., 1992. FP-S: a simplified function point counting method. *J. Syst. Softw.* 18, 245–254.
- Borade, J.G., Khalkar, V.R., 2013. Software project effort and cost estimation techniques. *Int. J. Adv. Res. Comput. Sci. Softw. Eng.* 3.
- Callele, D., Neufeld, E., Schneider, K., 2005. Requirements engineering and the creative process in the video game industry, in: 13th IEEE International Conference on Requirements Engineering (RE'05). Presented at the 13th IEEE International Conference on Requirements Engineering (RE'05), pp. 240–250. <https://doi.org/10.1109/RE.2005.58>
- Cole, R., Scotcher, E., 2016. Brilliant Agile project management: a practical guide to using Agile, Scrum and Kanban. Pearson UK.

- Dewi, R.S., Andari, T.W., Subriadi, A.P., Sholiq, 2018. Function Points Method in Game Casual Context, in: 2018 International Conference on Electrical Engineering and Computer Science (ICECOS). Presented at the 2018 International Conference on Electrical Engineering and Computer Science (ICECOS), pp. 367–372. <https://doi.org/10.1109/ICECOS.2018.8605188>
- Dewi, R.S., Subriadi, A.P., Sholiq, 2017. A Modification Complexity Factor in Function Points Method for Software Cost Estimation Towards Public Service Application. *Procedia Comput. Sci.*, 4th Information Systems International Conference 2017, ISICO 2017, 6-8 November 2017, Bali, Indonesia 124, 415–422. <https://doi.org/10.1016/j.procs.2017.12.172>
- Di Martino, S., Ferrucci, F., Gravino, C., Sarro, F., 2016. Web Effort Estimation: Function Point Analysis vs. COSMIC. *Inf. Softw. Technol.* 72, 90–109. <https://doi.org/10.1016/j.infsof.2015.12.001>
- Galorath, D.D., Evans, M.W., 2006. Software sizing, estimation, and risk management: when performance is measured performance improves, 1st ed. Auerbach Publications, New York.
- Gencel, C., Demirors, O., 2008. Measuring the Software Functional Size as a Vector of Measures. ACM Publishers.
- Halstead, M.H., 1977. Elements of Software Science (Operating and programming systems series). Elsevier Science Inc., New York, NY.
- Hamouda, A.E.D., 2014. Using Agile Story Points as an Estimation Technique in CMMI Organizations, in: 2014 Agile Conference. Presented at the 2014 Agile Conference, pp. 16–23. <https://doi.org/10.1109/AGILE.2014.11>
- Heemstra, F.J., Kusters, R.J., 1991. Function point analysis: Evaluation of a software cost estimation model. *Eur. J. Inf. Syst.* 1, 229–237.
- International Function Point Users Group (IFPUG), 2019. About SNAP - IFPUG [WWW Document]. URL <http://www.ifpug.org/about-snap/> (accessed 2.20.19).
- International Function Point Users Group (IFPUG), 2014. International Function Point Users Group (IFPUG) [WWW Document]. URL <http://www.ifpug.org/> (accessed 12.20.18).
- International Function Point Users Group (IFPUG), 2010. Function Point Counting Practices Manual Version 4.3.1. Int. Funct. Point Users Group Princet. Junction NJ.
- International Software Benchmarking Standards Group (ISBSG), 2011. Practical Software Project Estimation: A Toolkit for Estimating Software Development Effort & Duration. McGraw-Hill.
- Jeffery, D.R., Low, G.C., Barnes, M., 1993. A comparison of function point counting techniques. *IEEE Trans. Softw. Eng.* 19, 529–532. <https://doi.org/10.1109/32.232016>
- Jobplanet.com, 2016. Ragam Profesi TI dengan Gaji Rata-rata Paling Tinggi di Indonesia. Jobplanet Blog. URL <http://blog.id.jobplanet.com/ragam-profesi-ti-dengan-gaji-rata-rata-paling-tinggi-di-indonesia/> (accessed 11.12.19).
- Jørgensen, M., 2013. Relative Estimation of Software Development Effort: It Matters with What and How You Compare. *IEEE Softw.* 30, 74–79. <https://doi.org/10.1109/MS.2012.70>

- Kanode, C.M., Haddad, H.M., 2009. Software Engineering Challenges in Game Development, in: 2009 Sixth International Conference on Information Technology: New Generations. Presented at the 2009 Sixth International Conference on Information Technology: New Generations, pp. 260–265. <https://doi.org/10.1109/ITNG.2009.74>
- Keith, C., Cohn, M., 2010. Agile game development with Scrum. Addison-Wesley Upper Saddle River, NJ.
- Kent, S.L., 2010. The Ultimate History of Video Games: from Pong to Pokemon and beyond... the story behind the craze that touched our lives and changed the world. Three Rivers Press.
- Kumar, S., Rastogi, R., Nag, R., 2019. Limitations of Function Point Analysis in Multimedia Software/Application Estimation, in: Hoda, M.N., Chauhan, N., Quadri, S.M.K., Srivastava, P.R. (Eds.), Software Engineering, Advances in Intelligent Systems and Computing. Springer Singapore, pp. 383–392.
- Kurniawan, W., Sholih, S., Sutanto, T., 2013. Penentuan Effort Rate Pada Estimasi Effort Menggunakan Metode Use Case Point Untuk Pengembangan Perangkat Lunak Website Pemerintahan. *J. JSIKA* 2, 61–71.
- Leung, H., Fan, Z., 2002. Software cost estimation, in: Handbook of Software Engineering and Knowledge Engineering. World Scientific Publishing Company, pp. 307–324. [https://doi.org/10.1142/9789812389701\\_0014](https://doi.org/10.1142/9789812389701_0014)
- Li, Y.F., Xie, M., Goh, T.N., 2009. A study of project selection and feature weighting for analogy based software cost estimation. *J. Syst. Softw.* 82, 241–252. <https://doi.org/10.1016/j.jss.2008.06.001>
- Lokan, C.J., 2000. An empirical analysis of function point adjustment factors. *Inf. Softw. Technol.* 42, 649–659. [https://doi.org/10.1016/S0950-5849\(00\)00108-7](https://doi.org/10.1016/S0950-5849(00)00108-7)
- Longstreet, D., 2002. Fundamentals of function point analysis. Longstreet Consulting, Inc.
- Malim, G., 2018. Video games market is worth more than music and movies combined so why aren't CSPs launching games services? - VanillaPlus - The global voice of Telecoms IT [WWW Document]. VanillaPlus - Glob. Voice Telecoms IT. URL <https://www.vanillaplus.com/2018/07/05/40093-video-games-market-worth-music-movies-combined-arent-csps-launching-games-services/> (accessed 11.17.18).
- Maulana, R.F., Kurniawan, I., 2018. Tantangan dan Peluang Startup Game di Indonesia pada Tahun 2018 [WWW Document]. Tech Asia Indones. URL <https://id.techinasia.com/tantangan-peluang-startup-game-indonesia-tahun-2018> (accessed 11.17.18).
- McCabe, T.J., 1976. A complexity measure. *IEEE Trans. Softw. Eng.* 308–320.
- Meli, R., 1997. Early and Extended Function Point: a new method for Function Points estimation, in: IFPUG-Fall Conference. pp. 15–19.
- Meyer, B., 2014. Agile practices: managerial, in: Meyer, B. (Ed.), Agile!: The Good, the Hype and the Ugly. Springer International Publishing, Cham, pp. 89–102. [https://doi.org/10.1007/978-3-319-05155-0\\_6](https://doi.org/10.1007/978-3-319-05155-0_6)
- Mistry, A., 2017. Agile Story Point Estimation Techniques - T-Shirt Sizing [WWW Document]. URL <https://www.c-sharpcorner.com/article/agile-story-point-estimation-techniques-t-shirt-sizing/> (accessed 11.20.19).

- Newzoo, 2017. The Indonesian Gamer | 2017. Newzoo. URL <https://newzoo.com/insights/infographics/the-indonesian-gamer-2017/> (accessed 11.17.18).
- Newzoo, 2015. Indonesian Games Market 2015. Newzoo. URL <https://newzoo.com/insights/infographics/newzoo-summer-series-21-indonesian-games-market/> (accessed 11.17.18).
- Ng'ang'a, E., Tonui, I., 2015. A Survey on Software Sizing for Project Estimation. *Int. J. Softw. Eng. Knowl. Eng.* 5, 56.
- Nguyen, V., 2010. Improved size and effort estimation models for software maintenance, in: 2010 IEEE International Conference on Software Maintenance. Presented at the 2010 IEEE International Conference on Software Maintenance, pp. 1–2. <https://doi.org/10.1109/ICSM.2010.5609554>
- Owais, Mohd., Ramakishore, R., 2016. Effort, duration and cost estimation in agile software development, in: 2016 Ninth International Conference on Contemporary Computing (IC3). Presented at the 2016 Ninth International Conference on Contemporary Computing (IC3), pp. 1–5. <https://doi.org/10.1109/IC3.2016.7880216>
- Petrillo, F., Pimenta, M., 2010. Is Agility out There?: Agile Practices in Game Development, in: Proceedings of the 28th ACM International Conference on Design of Communication, SIGDOC '10. ACM, New York, NY, USA, pp. 9–15. <https://doi.org/10.1145/1878450.1878453>
- Pradani, W., 2014. Kajian Metode Perhitungan Metrik Function-Point dan Penerapannya pada Dua Perangkat Lunak yang Dipilih. *J. Al-Azhar Indones. Seri Sains Dan Teknol.* 2, 28–34.
- Pratiwi, D., 2013. Implementation of function point analysis in measuring the volume estimation of software system in object oriented and structural model of academic system. *ArXiv Prepr. ArXiv13092404*.
- Primandari A, P.L., Sholih, 2015. Effort Distribution to Estimate Cost in Small to Medium Software Development Project with Use Case Points. *Procedia Comput. Sci., The Third Information Systems International Conference 2015* 72, 78–85. <https://doi.org/10.1016/j.procs.2015.12.107>
- Putri, A.Y.P., Subriadi, A.P., 2019. Software Cost Estimation Using Function Point Analysis. *IPTEK J. Proc. Ser.* 79–83.
- Putri, A.Y.P., Subriadi, A.P., 2018. Modifikasi Metode Function Point Dengan Menambahkan Kompleksitas Proses Bisnis Pada General System Characteristics Untuk Estimasi Biaya Perangkat Lunak (Information Systems Department). Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia.
- Raith, F., Richter, I., Lindermeier, R., Klinker, G., 2013. Identification of Inaccurate Effort Estimates in Agile Software Development, in: 2013 20th Asia-Pacific Software Engineering Conference (APSEC). Presented at the 2013 20th Asia-Pacific Software Engineering Conference (APSEC), pp. 67–72. <https://doi.org/10.1109/APSEC.2013.114>
- Raslan, A.T., Darwish, N.R., Hefny, H.A., 2015. Towards a fuzzy based framework for effort estimation in agile software development. *Int. J. Comput. Sci. Inf. Secur.* 13, 37.

- Sabahat, N., Malik, A.A., Azam, F., 2017. A Size Estimation Model for Board-Based Desktop Games. *IEEE Access* 5, 4980–4990. <https://doi.org/10.1109/ACCESS.2017.2678459>
- Salazar, M.G., Mitre, H.A., Olalde, C.L., Sánchez, J.L.G., 2012. Proposal of Game Design Document from software engineering requirements perspective, in: 2012 17th International Conference on Computer Games (CGAMES). Presented at the 2012 17th International Conference on Computer Games (CGAMES), pp. 81–85. <https://doi.org/10.1109/CGames.2012.6314556>
- Saleh, K., 2011. Effort and Cost Allocation in Medium to Large Software Development Projects. *Int. J. Comput.* 5, 74–79.
- Sinclair, B., 2016. 2016 games industry brings in \$94 billion - Superdata [WWW Document]. *GamesIndustry.biz*. URL <https://www.gamesindustry.biz/articles/2016-12-21-2016-games-industry-brings-in-usd94-billion-superdata> (accessed 11.17.18).
- statista.com, 2019. Kickstarter: project funding success rate 2019 [WWW Document]. *Statista*. URL <https://www.statista.com/statistics/235405/kickstarter-project-funding-success-rate/> (accessed 7.17.19).
- Subriadi, A.P., Ningrum, P.A., 2014. Critical review of the effort rate value in use case point method for estimating software development effort. *J. Theoretical Appl. Inf. Technol.* 59, 735–44.
- Symons, C., 2001. Come back function point analysis (modernized)—all is forgiven! *Proc 4th Eur. Conf. Softw. Meas. ICT Control FESMA-DASMA* 413–426.
- Symons, C.R., 1988. Function point analysis: difficulties and improvements. *IEEE Trans. Softw. Eng.* 14, 2–11. <https://doi.org/10.1109/32.4618>
- Tanveer, B., Guzmán, L., Engel, U.M., 2016. Understanding and Improving Effort Estimation in Agile Software Development—An Industrial Case Study, in: 2016 IEEE/ACM International Conference on Software and System Processes (ICSSP). Presented at the 2016 IEEE/ACM International Conference on Software and System Processes (ICSSP), pp. 41–50. <https://doi.org/10.1109/ICSSP.2016.014>
- Tipu, S.K., Zia, S., 2012. An Effort Estimation Model for Agile Software Development 2, 12.
- Vazquez, C.E., SIMÕES, G.S., Albert, R.M., 2003. *Análise de Pontos de Função: medição, estimativas e gerenciamento de projetos de software*. Ed. Érica São Paulo 3.
- Verner, J.M., Tate, G., Jackson, B., Hayward, R.G., 1989. Technology dependence in function point analysis: a case study and critical review, in: 11th International Conference on Software Engineering. *IEEE*, pp. 375–382.
- Vijay, J.F., 2018. Enrichment of accurate software effort estimation using fuzzy-based function point analysis in business data analytics. *Neural Comput. Appl.* 1–7. <https://doi.org/10.1007/s00521-018-3565-3>
- Xia, W., Capretz, L.F., Ho, D., Ahmed, F., 2008. A new calibration for Function Point complexity weights. *Inf. Softw. Technol.* 50, 670–683. <https://doi.org/10.1016/j.infsof.2007.07.004>
- Zackariasson, P., Wilson, T.L., 2010. Paradigm shifts in the video game industry. *Compet. Rev.* 20, 139–151. <https://doi.org/10.1108/10595421011029857>

[HALAMAN INI SENGAJA DIKOSONGKAN]

# LAMPIRAN

## Lampiran 1 - perhitungan FPA

Semua data yang diperoleh merupakan hasil dari pengamatan peneliti dari antarmuka aplikasi permainan digital ketiga jenis proyek (P-MCP, P-MCE, P-MVN). Pengamatan berdasarkan antarmuka sendiri karena tidak tersedianya dokumentasi pada saat aplikasi dibangun sehingga peneliti melakukan langkah proaktif dengan menyusun kebutuhan berdasarkan antarmuka setiap aplikasi dan melakukan perhitungan FPA berdasarkan kebutuhan tersebut.

Definisi dari istilah yang akan ditemui pada lampiran 1:

Daftar kebutuhan	: susunan daftar kebutuhan dari sebuah proyek
Tugas	: pekerjaan yang harus dikerjakan oleh tim pengembang supaya kebutuhan dapat terpenuhi
Record Elemen Types	: tabel yang digunakan untuk menampilkan data
File Type References	: tabel yang dirujuk oleh fungsi
Data Element Type	: data yang ditunjuk oleh fungsi
Function point	: metode perhitungan function point
Type	: tipe fungsi sebuah pekerjaan (diabaikan apabila tidak sesuai dengan definisi kelima tipe fungsi FPA)
Weight	: nilai bobot berdasarkan jumlah kombinasi RET,DET dan DET pada setiap tipe fungsi sesuai ketentuan metode FPA

**A. Proyek P-MCP**

No	Daftar kebutuhan (fungsi)	Tugas	R E T	F T R	D E T	Function Point (FP)	
						Type	Weight
<i>Interface Home</i>							
1	Berpindah ke Interface Pilihan Level	Membuat fungsi untuk berpindah ke Tampilan Pilihan Level	-	1	1	EQ	3
		Membuat desain tombol “play” yang mengindikasikan memulai permainan	Diabaikan				
2	Pemain dapat mengaktifkan atau mematikan efek suara	Membuat fungsi untuk mengaktifkan atau mematikan efek suara	- 1	1 -	1 1	EI ILF	3 7
		Membuat desain tombol efek suara “aktif” dan “nonaktif”	Diabaikan				
3	Menampilkan iklan pada aplikasi	Membuat fungsi untuk menampilkan iklan	- 1	1 -	2 2	EQ EIF	3 5
		Membuat fungsi untuk menghilangkan iklan					
4	Menghilangkan iklan dengan mengaktifkan pembelian dalam aplikasi	Membuat fungsi untuk menghilangkan iklan					
		Membuat desain tombol “no ads”	Diabaikan				

No	Daftar kebutuhan (fungsi)	Tugas	R E T	F T R	D E T	Function Point (FP)	
						Type	Weight
5	Menampilkan website developer	Membuat fungsi untuk membuka browser dengan url website developer.	-	1	1	EQ	3
		Membuat desain tombol Info	Diabaikan				
<b><i>Interface Pilihan Level</i></b>							
6	Memiliki visualisasi interface Pilihan Level yang menarik	Merancang konsep dan komposisi visual	Diabaikan				
		Mendesain background untuk interface Pilihan Level	Diabaikan				
		Mendesain label "Level Selection" pada interface Pilihan Level	Diabaikan				
7	Berpindah ke Interface Home	Membuat fungsi untuk berpindah ke interface Home	-	1	1	EQ	3
		Membuat desain tombol untuk fungsi kembali ke interface Home	Diabaikan				
8	Berpindah ke Interface Gameplay	Membuat fungsi untuk berpindah ke interface Gameplay	-	1	1	EQ	3

No	Daftar kebutuhan (fungsi)	Tugas	R E T	F T R	D E T	Function Point (FP)	
						Type	Weight
	sesuai pilihan level						
9	Navigasi daftar tingkatan level	Membuat fungsi untuk menampilkan daftar tingkatan level dengan range tertentu	-	1	1	EQ	3
		Membuat desain tombol navigasi daftar tingkatan level	Diabaikan				
10	Pemain dapat memainkan level tertentu apabila level sebelumnya telah diselesaikan	Membuat fungsi untuk mengunci dan membuka level.	1	-	2	ILF	7
		Membuat desain tombol untuk tingkatan level dengan kondisi terbuka dan terkunci	Diabaikan				
<b>Interface Gameplay</b>							
11	Konsep level yang berbeda dalam 14 tingkatan level	Mengonsep mapping permainan (termasuk posisi kereta dimulai, alur rel, lokasi stasiun kereta)	Diabaikan				
		Mendesain ilustrasi background					

No	Daftar kebutuhan (fungsi)	Tugas	R E T	F T R	D E T	Function Point (FP)	
						Type	Weight
12	Memunculkan (spawn) objek kereta pada posisi dan waktu tertentu.	Membuat fungsi untuk memunculkan kereta api.	-	1	1	EQ	3
		Desain objek kereta (14 jenis warna)	Diabaikan				
13	Mendefinisikan dan menampilkan waktu permainan.	Mendefinisikan dan membuat fungsi untuk menampilkan dan kalkulasi waktu permainan.	-	1	1	EO	4
		Desain indikator waktu	Diabaikan				
14	Mendefinisikan dan menampilkan nyawa pemain.	Mendefinisikan dan membuat fungsi untuk menampilkan dan kalkulasi nyawa.	-	1	1	EO	4
		Desain indikator nyawa pemain	Diabaikan				
15	Pemain mendapatkan pengurangan nyawa untuk setiap kereta yang masuk ke stasiun yang salah (tidak sesuai dengan warna).	Membuat fungsi untuk kalkulasi nyawa setiap kesalahan terjadi.	Diabaikan				
		Desain stasiun kereta api	Diabaikan				

No	Daftar kebutuhan (fungsi)	Tugas	R E T	F T R	D E T	Function Point (FP)		
						Type	Weight	
16	Menentukan kecepatan pergerakan kereta.	Membuat fungsi untuk menjalankan kereta berdasarkan kecepatan tertentu.	Diabaikan					
17	Merubah arah rel kereta berdasarkan input pemain.	Membuat fungsi untuk menerima input pemain yang berguna untuk merubah arah rel kereta.	- 1	1 -	1 1	EI ILF	3 7	
		Desain indikator sebagai fungsi merubah arah rel kereta berdasarkan input pemain	Diabaikan					
18	Kereta dapat berjalan sesuai dengan alur rel kereta	Membuat fungsi pathfinding untuk objek kereta	Diabaikan					
		Desain rel kereta api	Diabaikan					
19	Pemain dapat menghentikan permainan sementara dan melanjutkannya kembali	Membuat fungsi untuk menghentikan permainan sementara.	-	1	1	EI	3	
		Desain tombol pause untuk fungsi menghentikan permainan sementara	Diabaikan					
		Desain tulisan "pause"	Diabaikan					

No	Daftar kebutuhan (fungsi)	Tugas	R E T T	F T R	D E T	Function Point (FP)	
						Type	Weight
		Desain frame untuk kondisi pause	Diabaikan				
20	Menampilkan jumlah bintang yang menandakan nilai dari pemain.	Membuat fungsi untuk mengakumulasi nilai yang didapat pemain berdasarkan jumlah kereta yang masuk ke stasiun yang benar dan menampikannya.	-	1	1	EO	4
			1	-	1	ILF	7
		Membuat desain bintang untuk visualisasi nilai	Diabaikan				
		Desain frame sebagai pelat penanda.	Diabaikan				
		Desain label	Diabaikan				
21	Memungkinkan pemain menuju level selanjutnya apabila memenangkan permainan	Membuat fungsi menuju tingkatan level selanjutnya	-	1	1	EQ	3
		Desain tombol yang mewakili fungsi untuk memainkan permainan di tingkatan level selanjutnya.	Diabaikan				
22	Mengulang permainan.	Membuat fungsi untuk mengulang permainan.	-	1	1	EQ	3

No	Daftar kebutuhan (fungsi)	Tugas	R E T	F T R	D E T	Function Point (FP)		
						Type	Weight	
		Desain tombol yang mewakili fungsi untuk mengulang permainan	Diabaikan					
23	Berpindah ke Interface Pilihan Level dengan menampilkan iklan terlebih dahulu.	Membuat fungsi untuk menampilkan iklan yang kemudian berpindah ke interface Pilihan Level.	-	1	1	EQ	3	
		Desain tombol yang mewakili fungsi untuk kembali ke interface Pilihan Level	Diabaikan					
<b>Total nilai function point (FP)</b>							<b>81</b>	
<b>Keterangan</b>								
 : Kebutuhan / tugas yang didapat pada tahapan Mechanic  : Kebutuhan / tugas yang didapat pada tahapan Dynamic  : Kebutuhan dan tugas yang didapat pada tahapan Aesthetic  : Kebutuhan yang memiliki tugas teknis program dan visual								

## B. Proyek P-MCE

No	Daftar kebutuhan (fitur)	Tugas	R E T	F T R	D E T	Function Point (FP)	
						Type	Weight
<i>Interface Pilihan Permainan</i>							
1	Berpindah ke Interface Pilihan Level SBB	Membuat fungsi untuk memanggil interface Pilihan Level SBB	-	1	1	EQ	3
		Menciptakan desain tombol sebagai tampilan visual untuk fungsi memanggil interface Pilihan Level SBB	Diabaikan				
2	Berpindah ke Interface Gameplay PP	Membuat fungsi untuk memanggil interface Gameplay PP	-	1	1	EQ	3
		Menciptakan desain tombol sebagai tampilan visual untuk fungsi memanggil interface Gameplay PP	Diabaikan				
3	Menampilkan contoh barang bawaan	Membuat fungsi untuk menampilkan aset barang-barang berdasarkan kategori	-	1	2	EQ	3

No	Daftar kebutuhan (fitur)	Tugas	R E T	F T R	D E T	Function Point (FP)	
						Type	Weight
	berdasarkan kategori	Menciptakan desain tombol sebagai tampilan visual untuk fungsi menampilkan barang-barang berdasarkan kategori	Diabaikan				
		Mengonsep tata letak aset barang pada setiap kategori	Diabaikan				
4	Menampilkan contoh perilaku penumpang yang harus ditaati.	Membuat fungsi untuk menampilkan peraturan perilaku penumpang.	-	1	1	EQ	3
		Membuat desain ilustrasi tombol sebagai tampilan visual untuk fungsi menampilkan contoh perilaku penumpang	Diabaikan				
5	Menampilkan peraturan perilaku penumpang satu-persatu dan dapat berubah dengan	Membuat fungsi untuk menampilkan peraturan perilaku penumpang satu-persatu dan fungsi navigasinya.	-	1	1	EQ	3

No	Daftar kebutuhan (fitur)	Tugas	R E T	F T R	D E T	Function Point (FP)	
						Type	Weight
	menggunakan tombol navigasi	Membuat tombol sebagai tampilan visual untuk fungsi navigasi	Diabaikan				
		Membuat ilustrasi perilaku penumpang dan akibatnya	Diabaikan				
6	Mematikan dan menghidupkan suara BGM	Membuat fungsi untuk menyalakan dan mematikan efek BGM pada permainan	- 1	1 -	1 1	EI ILF	3 7
		Menciptakan desain tombol sebagai tampilan visual untuk fungsi mematikan dan menghidupkan efek BGM	Diabaikan				
		Membuat tampilan interface Pilihan permainan	Diabaikan				
7	Membuat tampilan interface Pilihan permainan	Menciptakan desain ilustrasi background	Diabaikan				
		Menciptakan desain ilustrasi logo permainan digital	Diabaikan				
		Menciptakan desain frame	Diabaikan				
		Mengonsep komposisi dan	Diabaikan				

No	Daftar kebutuhan (fitur)	Tugas	R E T	F T R	D E T	Function Point (FP)		
						Type	Weight	
		pengaturan tata letak komponen visual						
<b><i>Interface Pilihan Level SBB</i></b>								
8	Berpindah ke interface Pilihan Permainan	Membuat fungsi untuk memanggil interface Pilihan Permainan	-	1	1	EQ	3	
		Menciptakan desain tombol sebagai tampilan visual untuk fungsi memanggil interface Pilihan Permainan	Diabaikan					
9	Berpindah ke Interface Gameplay SBB sesuai pilihan level.	Membuat fungsi untuk memanggil interface Gameplay SBB dengan mengirimkan informasi level yang dipilih	-	1	1	EQ	3	
		Menciptakan desain tombol untuk menuju interface Gameplay SBB (dua kondisi terkunci atau terbuka)	Diabaikan					

No	Daftar kebutuhan (fitur)	Tugas	R E T	F T R	D E T	Function Point (FP)	
						Type	Weight
10	Pemain dapat memainkan level tertentu apabila level sebelumnya telah diselesaikan	Membuat fungsi untuk mengunci dan membuka level.	1	-	2	ILF	7
		Menciptakan tombol level dengan dua desain (terbuka dan terkunci)	Diabaikan				
11	Setiap tingkatan level memiliki catatan waktu terbaik	Membuat fungsi untuk menampilkan waktu terbaik pada setiap tingkatan level.	-	1	1	EO	4
12	Pemain dapat melihat dan menutup tutorial mode permainan SBB	Membuat fungsi untuk menampilkan dan menutup jendela tutorial SBB	-	1	1	EQ	3
		Menciptakan desain tutorial	Diabaikan				
13	Tutorial berbentuk barisan teks dan ditampilkan secara bertahap dengan bantuan tombol navigasi untuk menavigasi teks tutorial	Membuat fungsi navigasi untuk tutorial SBB	Diabaikan (fungsi proses)				
		Membuat desain tombol sebagai tampilan untuk fungsi navigasi.	Diabaikan				

No	Daftar kebutuhan (fitur)	Tugas	R E T	F T R	D E T	Function Point (FP)	
						Type	Weight
14	Pemain dapat melihat keterangan bandara pada setiap tingkatan level	Membuat fungsi untuk menampilkan keterangan bandara pada setiap tingkatan level.	-	1	2	EO	4
		Membuat ilustrasi bandara untuk setiap level	Diabaikan				
15	Memiliki tampilan Pilihan Level bertema indonesia	Mengonsep tata letak visual	Diabaikan				
		Menciptakan desain ilustrasi background peta indonesia	Diabaikan				
		Menciptakan desain frame	Diabaikan				
		Menciptakan desain awan Menciptakan dan animasi terhadap objek awan	Diabaikan				
<b>Interface Gameplay SBB</b>							
16	Menginisialisasi dan menjalankan timer permainan	Membuat fungsi untuk menampilkan timer dan menjalankan timer	-	1	1	EO	4
		Membuat desain frame untuk tempat indikator timer	Diabaikan				

No	Daftar kebutuhan (fitur)	Tugas	R E T	F T R	D E T	Function Point (FP)	
						Type	Weight
17	Memunculkan barang berlabel legal, terlarang, dan bagasi secara random dengan komposisi tertentu	Membuat fungsi untuk menampilkan barang secara random.	-	1	2	EO	4
		Menginisiasi label pada barang	Diabaikan (Fungsi proses)				
		Membuat desain setiap aset barang	Diabaikan				
		Menciptakan animasi <i>swing</i> pada setiap barang	Diabaikan				
18	Kemampuan drag & drop pada masing-masing barang	Membuat fungsi <i>drag&amp;drop</i> dan meng-attach fungsi tersebut pada setiap barang	- 2	1 -	1 3	EI ILF	3 7
		Membuat area drop barang berlabel (dilarang sepenuhnya dan bagasi tercatat)	Diabaikan (fungsi teknis proses)				
19	Membuat area drop barang berlabel (dilarang sepenuhnya dan bagasi tercatat)	Membuat fungsi untuk menginisiasi area drop berlabel.	Diabaikan (fungsi teknis proses)				
		Membuat desain area drop barang	Diabaikan				
20	Terdapat inisialisasi jumlah target barang yang harus di filter setiap levelnya	Menginisialisasi nilai jumlah target barang.	-	1	2	EO	4
		Membuat desain alarm berwarna merah yang menandakan koper berisi barang berbahaya	Diabaikan				

No	Daftar kebutuhan (fitur)	Tugas	R E T	F T R	D E T	Function Point (FP)		
						Type	Weight	
21	Barang yang berlabel sama dengan area drop dapat di drop pada area drop.	Membuat fungsi pengecekan untuk mendeteksi kesesuaian label pada barang dan area drop	Diabaikan (Fungsi teknis)					
22	Mengurangi nilai jumlah barang apabila barang telah di drop pada area yang sesuai	Membuat fungsi untuk mengurangi nilai pada jumlah target barang	-	1	2	EO	4	
23	Apabila jumlah targer barang = 0, maka sistem akan mencatat waktu timer	Membuat fungsi untuk mencatat waktu timer saat nilai jumlah target barang = 0.	Diabaikan (Fungsi teknis)					
24	Mengupdate catatan waktu terbaik	Membuat fungsi untuk membandingkan catatan waktu yg tersimpan dengan catatan waktu permainan. Apabila catatan waktu lebih cepat dari waktu yang tersimpan maka update catatan waktu,	-	1	2	EI	3	

No	Daftar kebutuhan (fitur)	Tugas	R E T	F T R	D E T	Function Point (FP)	
						Type	Weight
		dan tampilkan “Anda mencatat waktu terbaik”, apabila tidak maka tampilkan waktu terbaik dari waktu yang tersimpan					
		Mendesain tampilan saat permainan berhasil dimenangkan	Diabaikan				
25	Pemain dapat mengulang permainan pada level yang dimainkannya	Membuat fungsi untuk mengulang permainan	-	1	1	EQ	3
		Mendesain ilustrasi tombol yang berfungsi untuk mengulang permainan	Diabaikan				
26	Pemain dapat kembali ke tampilan pilihan level	Membuat fungsi untuk memanggil interface Pilihan Level SBB	Diabaikan (fungsi tidak unik, sama seperti no 1)				
		Mendesain ilustrasi tombol yang berfungsi untuk memanggil	Diabaikan				

No	Daftar kebutuhan (fitur)	Tugas	R E T	F T R	D E T	Function Point (FP)	
						Type	Weight
		interface Pilihan Level SBB					
27	Pemain dapat menghentikan sementara dan melanjutkan permainan dan timer	Membuat fungsi untuk menghentikan sementara dan melanjutkan permainan.	-	1	1	EI	3
		Mendesain ilustrasi tombol yang berfungsi untuk menghentikan dan melanjutkan permainan				Diabaikan	
		Mendesain kondisi tampilan saat permainan berhenti				Diabaikan	
28	Tampilan permainan SBB seperti kondisi koper dalam mesin x-ray	Ilustrasi desain koper				Diabaikan	
		Animasi koper memasuki mesin x-ray				Diabaikan	
		Ilustrasi background berupa konveyor mesin x-ray				Diabaikan	
<b><i>Interface Gameplay PP</i></b>							
29	Menampilkan aset – aset yang	Membuat fungsi untuk menampilkan				Diabaikan (merupakan fungsi proses)	

No	Daftar kebutuhan (fitur)	Tugas	R E T	F T R	D E T	Function Point (FP)	
						Type	Weight
	digunakan sebagai persyaratan perilaku penumpang pada posisi tertentu	aset pada koordinat tertentu	Diabaikan				
		Merancang komposisi aset-aset yang digunakan, antara lain (kursi, background interior pesawat, penumpang, meja, sabuk pengaman, smartphone)					
30	Setiap asset dapat berinteraksi dengan pemain apabila disentuh ( <i>touch</i> ) dengan menampilkan opsi pilihan	Membuat fungsi interaksi touch pada setiap aset dengan hasil interaksi berupa opsi pilihan	-	2	2	EI	3
			2	-	2	ILF	7
		Mendesain ilustrasi setiap opsi pilihan	Diabaikan				
31	Setiap pilihan opsi yang benar akan mengakibatkan satu persyaratan	Membuat fungsi untuk hide dan unhide pada asset sebagai akibat dari pilihan opsi yang benar	-	2	2	EI	3
			2	-	2	ILF	7

No	Daftar kebutuhan (fitur)	Tugas	R E T	F T R	D E T	Function Point (FP)		
						Type	Weight	
	terpenuhi dan merubah tampilan permainan	Mendesain asset dengan dua kondisi yang berbeda, seperti penumpang dengan kondisi berdiri dan duduk dan penumpang dengan posisi memegang HP dengan posisi duduk sempurna.	Diabaikan					
32	Menampilkan instruksi persyaratan apa yang belum terpenuhi apabila pemain menekan tombol "OK" dan membuka tampilan kemenangan apabila semua persyaratan telah terpenuhi.	Membuat fungsi untuk pengecekan persyaratan pada tombol "OK"	-	1	1	EQ	3	
		Mendesain ilustrasi tombol untuk pengecekan persyaratan penumpang dengan label "OK"	Diabaikan					
		Mendesain tampilan kemenangan Gameplay PP	Diabaikan					
33	Pemain dapat mengulang	Membuat fungsi untuk mengulang permainan PP	-	1	1	EQ	3	

No	Daftar kebutuhan (fitur)	Tugas	R E T	F T R	D E T	Function Point (FP)	
						Type	Weight
	kembali permainan PP	Mendesain ilustrasi tombol yang berfungsi untuk mengulang permainan	Diabaikan				
34	Pemain dapat kembali ke tampilan pilihan mode permainan	Membuat fungsi untuk memanggil interface Pilihan Permainan.	Diabaikan (fungsi sama dengan no 8)				
		Mendesain ilustrasi tombol yang berfungsi memanggil interface Pilihan Permainan	Diabaikan				
35	Pemain dapat melihat dan menutup tutorial mode permainan PP	Membuat fungsi untuk menampilkan dan menutup jendela tutorial PP	Diabaikan (fungsi sama dengan no. 12)				
		Mendesain tampilan tutorial PP	Diabaikan				
36	Tutorial berbentuk barisan teks dan ditampilkan secara bertahap dengan bantuan tombol navigasi untuk	Membuat fungsi navigasi untuk tutorial PP	Diabaikan (fungsi proses)				
		Mendesain ilustrasi tombol yang berfungsi untuk navigasi tutorial PP	Diabaikan				

No	Daftar kebutuhan (fitur)	Tugas	R E T	F T R	D E T	Function Point (FP)		
						Type	Weight	
	menavigasi teks tutorial							
37	Memiliki Tampilan awal permainan yang menjelaskan petunjuk permainan PP	Mendesain tampilan awal permainan				Diabaikan		
<b>Total Function Point</b>							<b>110</b>	
<b>Keterangan</b>								
 : Kebutuhan / tugas yang didapat pada tahapan Mechanic  : Kebutuhan / tugas yang didapat pada tahapan Dynamic  : Kebutuhan / tugas yang didapat pada tahapan Aesthetic  : Kebutuhan yang memiliki tugas teknis program dan visual								

### C. Proyek P-MVN

No	Daftar kebutuhan (fitur)	Tugas	R E T	F T R	D E T	Function Point (FP)	
						Type	Wight
<i>Interface Home</i>							
1	Berpindah ke Interface Story sesuai dengan pilihan cerita	Membuat fungsi untuk memanggil interface Story	-	1	1	EQ	3
		Membuat Ilustrasi Thumbnail yang mewakili karakter di cerita	Diabaikan				
2	Pemain dapat menavigasi cerita dengan swipe ke kiri dan kekanan pada area pilihan cerita	Membuat fungsi yang mendeteksi inputan <i>swipe</i> pada area cerita	Diabaikan (fungsi proses)				
3	Pemain dapat mengetahui update cerita terbaru	Membuat fungsi untuk menampilkan thumbnail cerita terbaru yang ditambahkan ke dalam aplikasi	-	1	2	EO	4

No	Daftar kebutuhan (fitur)	Tugas	R E T	F T R	D E T	Function Point (FP)		
						Type	Wight	
4	Tampilan minimalis pada tampilan home	Mengonsep tampilan pada interface Home	Diabaikan					
<i>Interface Story</i>								
5	Pemain dapat membaca narasi cerita atau dialog dengan menyentuh layar	Membuat fungsi mendeteksi sentuhan pemain yang berguna untuk menampilkan narasi cerita selanjutnya	-	1	4	EQ	3	
		Mengonsep alur cerita	Diabaikan					
		Membuat ilustrasi untuk background / latar belakang	Diabaikan					
		Membuat desain ilustrasi karakter cerita	Diabaikan					
		Membuat animasi pada situasi dan kondisi tertentu (contoh saat situasi hujan,	Diabaikan					

No	Daftar kebutuhan (fitur)	Tugas	R E T	F T R	D E T	Function Point (FP)	
						Type	Wight
		maka terdapat animasi petir dan rintik hujan)	Diabaikan				
		Membuat frame untuk tempat narasi cerita dan dialog					
6	Pemain dapat menginputkan nama untuk menjadi subjek dalam narasi cerita yang berkaitan dengan pemain atau pada dialog antara NPC dengan pemain	Membuat fungsi input nama pemain	- 1	1 -	2 2	EI ILF	3 7
		Membuat ilustrasi frame untuk input nama pemain	Diabaikan				
7	Pemain dapat kembali ke tampilan Home setelah menyelesaikan cerita	Membuat fungsi untuk memanggil interface Home	-	1	2	EQ	3
		Membuat ilustrasi tombol untuk fungsi kembali ke tampilan home	Diabaikan				

No	Daftar kebutuhan (fitur)	Tugas	R E T	F T R	D E T	Function Point (FP)	
						Type	Wight
8	Alur cerita memiliki opsi yang dapat dipilih pemain	Membuat fungsi untuk menampilkan opsi	-	1	2	EQ	3
		Membuat ilustrasi tombol untuk tampilan opsi pilihan	Diabaikan				
		Merancang event untuk setiap cabang opsi	Diabaikan				
<b>Total Function Point</b>							<b>26</b>
<b>Keterangan</b>							
 : Kebutuhan / tugas yang didapat pada tahapan Mechanic  : Kebutuhan / tugas yang didapat pada tahapan Dynamic  : Kebutuhan / tugas yang didapat pada tahapan Aesthetic  : Kebutuhan yang memiliki tugas teknis program dan visual							

## Lampiran 2 – penilaian kompleksitas dengan MSP

Dalam menilai kebutuhan menggunakan MSP, informasi yang akan ditemui adalah:

Keterangan	
	: Kebutuhan Landasan
<b>Tingkat Kompleksitas</b>	
	: 1 ( <i>Easiest</i> )
	: 2
	: 3
	: 5 ( <i>Intermediate</i> )
	: 8 ( <i>Intermediate</i> )
	: 13
	: 21
	: 34 ( <i>Hardest</i> )

Pada penelitian ini, nilai indikator yang digunakan untuk menilai kompleksitas pada kebutuhan menggunakan MSP adalah nilai kelipatan bilangan fibonacci. Sedangkan kebutuhan landasan merupakan kebutuhan yang dibandingkan dengan kebutuhan lainnya berdasarkan kompleksitas tugas dalam kebutuhan tersebut untuk mendapatkan nilai kompleksitas dari kebutuhan-kebutuhan lainnya,

### A. Proyek P-MCE

No	Daftar kebutuhan (fitur)	Tugas	Nilai SP
<i>Interface Pilihan Permainan</i>			
1	Berpindah ke Interface Pilihan Level SBB	Membuat fungsi untuk memanggil interface Pilihan Level SBB	2
		Menciptakan desain tombol sebagai tampilan visual untuk fungsi memanggil interface Pilihan Level SBB	

No	Daftar kebutuhan (fitur)	Tugas	Nilai SP
2	Berpindah ke Interface Gameplay PP	Membuat fungsi untuk memanggil interface Gameplay PP	2
		Menciptakan desain tombol sebagai tampilan visual untuk fungsi memanggil interface Gameplay PP	
3	Menampilkan contoh barang bawaan berdasarkan kategori	Membuat fungsi untuk menampilkan aset barang-barang berdasarkan kategori	8
		Menciptakan desain tombol sebagai tampilan visual untuk fungsi menampilkan barang-barang berdasarkan kategori	
		Mengonsep tata letak aset barang pada setiap kategori	
4	Menampilkan contoh perilaku penumpang yang harus ditaati.	Membuat fungsi untuk menampilkan peraturan perilaku penumpang.	8
		Membuat desain ilustrasi tombol sebagai tampilan visual untuk fungsi menampilkan contoh perilaku penumpang	
5	Menampilkan peraturan perilaku penumpang satu persatu dan dapat berubah dengan menggunakan tombol navigasi	Membuat fungsi untuk menampilkan peraturan perilaku penumpang satu-persatu dan fungsi navigasinya.	8
		Membuat tombol sebagai tampilan visual untuk fungsi navigasi	
		Membuat ilustrasi perilaku penumpang dan akibatnya	

No	Daftar kebutuhan (fitur)	Tugas	Nilai SP
6	Mematikan dan menghidupkan suara BGM	Membuat fungsi untuk menyalakan dan mematikan efek BGM pada permainan	2
		Menciptakan desain tombol sebagai tampilan visual untuk fungsi mematikan dan menghidupkan efek BGM	
7	Membuat tampilan interface Pilihan permainan	Menciptakan desain ilustrasi background	13
		Menciptakan desain ilustrasi logo permainan digital	
		Menciptakan desain frame	
		Mengonsep komposisi dan pengaturan tata letak komponen visual	
<b><i>Interface Pilihan Level SBB</i></b>			
8	Berpindah ke interface Pilihan Permainan	Membuat fungsi untuk memanggil interface Pilihan Permainan	2
		Menciptakan desain tombol sebagai tampilan visual untuk fungsi memanggil interface Pilihan Permainan	
9	Berpindah ke Interface Gameplay SBB sesuai pilihan level.	Membuat fungsi untuk memanggil interface Gameplay SBB dengan mengirimkan informasi level yang dipilih	2

No	Daftar kebutuhan (fitur)	Tugas	Nilai SP
		Menciptakan desain tombol untuk menuju interface Gameplay SBB (dua kondisi terkunci atau terbuka)	
10	Pemain dapat memainkan level tertentu apabila level sebelumnya telah diselesaikan	Membuat fungsi untuk mengunci dan membuka level. Menciptakan tombol level dengan dua desain (terbuka dan terkunci)	8
11	Setiap tingkatan level memiliki catatan waktu terbaik	Membuat fungsi untuk menampilkan waktu terbaik pada setiap tingkatan level.	8
12	Pemain dapat melihat dan menutup tutorial mode permainan SBB	Membuat fungsi untuk menampilkan dan menutup jendela tutorial SBB Menciptakan desain tutorial	8
13	Tutorial berbentuk barisan teks dan ditampilkan secara bertahap dengan bantuan tombol navigasi untuk menavigasi teks tutorial	Membuat fungsi navigasi untuk tutorial SBB Membuat desain tombol sebagai tampilan untuk fungsi navigasi.	2
14	Pemain dapat melihat keterangan bandara pada setiap tingkatan level	Membuat fungsi untuk menampilkan keterangan bandara pada setiap tingkatan level. Membuat ilustrasi bandara untuk setiap level	2
15	Memiliki tampilan Pilihan Level bertema indonesia	Mengonsep tata letak visual Menciptakan desain ilustrasi background peta indonesia Menciptakan desain frame	21

No	Daftar kebutuhan (fitur)	Tugas	Nilai SP
		Menciptakan desain awan Menciptakan dan animasi terhadap objek awan	
<b>Interface Gameplay SBB</b>			
16	Menginialisasi dan menjalankan timer permainan	Membuat fungsi untuk menampilkan timer dan menjalankan timer	5
		Membuat desain frame untuk tempat indikator timer	
17	Memunculkan barang berlabel legal, terlarang, dan bagasi secara random dengan komposisi tertentu	Membuat fungsi untuk menampilkan barang secara random.	34
		Menginisiasi label pada barang	
		Membuat desain setiap aset barang	
		Menciptakan animasi <i>swing</i> pada setiap barang	
18	Kemampuan drag & drop pada masing-masing barang	Membuat fungsi <i>drag&amp;drop</i> dan meng-attach fungsi tersebut pada setiap barang	13
19	Membuat area drop barang berlabel (dilarang sepenuhnya dan bagasi tercatat)	Membuat fungsi untuk menginisiasi area drop berlabel.	13
		Membuat desain area drop barang	
20		Menginisialisasi nilai jumlah target barang.	5

No	Daftar kebutuhan (fitur)	Tugas	Nilai SP
	Terdapat inisialisasi jumlah target barang yang harus di filter setiap levelnya	Membuat desain alarm berwarna merah yang menandakan koper berisi barang berbahaya	
21	Barang yang berlabel sama dengan area drop dapat di drop pada area drop.	Membuat fungsi pengecekan untuk mendeteksi kesesuaian label pada barang dan area drop	5
22	Mengurangi nilai jumlah barang apabila barang telah di drop pada area yang sesuai	Membuat fungsi untuk mengurangi nilai pada jumlah target barang	2
23	Apabila jumlah targer barang = 0, maka sistem akan mencatat waktu timer	Membuat fungsi untuk mencatat waktu timer saat nilai jumlah target barang = 0.	5
24	Mengupdate catatan waktu terbaik	Membuat fungsi untuk membandingkan catatan waktu yg tersimpan dengan catatan waktu permainan. Apabila catatan waktu lebih cepat dari waktu yang tersimpan maka update catatan waktu, dan tampilkan “Anda mencatat waktu terbaik”, apabila tidak maka tampilkan waktu terbaik dari waktu yang tersimpan	8
		Mendesain tampilan saat permainan berhasil dimenangkan	
25		Membuat fungsi untuk mengulang permainan	2

No	Daftar kebutuhan (fitur)	Tugas	Nilai SP
	Pemain dapat mengulang permainan pada level yang dimainkannya	Mendesain ilustrasi tombol yang berfungsi untuk mengulang permainan	
26	Pemain dapat kembali ke tampilan pilihan level	Membuat fungsi untuk memanggil interface Pilihan Level SBB	2
		Mendesain ilustrasi tombol yang berfungsi untuk memanggil interface Pilihan Level SBB	
27	Pemain dapat menghentikan sementara dan melanjutkan permainan dan timer	Membuat fungsi untuk menghentikan sementara dan melanjutkan permainan.	13
		Mendesain ilustrasi tombol yang berfungsi untuk menghentikan dan melanjutkan permainan	
		Mendesain kondisi tampilan saat permainan berhenti	
28	Tampilan permainan SBB seperti kondisi koper dalam mesin x-ray	Ilustrasi desain koper	13
		Animasi koper memasuki mesin x-ray	
		Ilustrasi background berupa konveyor mesin x-ray	
<b><i>Interface Gameplay PP</i></b>			
29	Menampilkan aset – aset yang digunakan sebagai persyaratan perilaku	Membuat fungsi untuk menampilkan aset pada koordinat tertentu	13

No	Daftar kebutuhan (fitur)	Tugas	Nilai SP
	penumpang pada posisi tertentu	Merancang komposisi aset-aset yang digunakan, antara lain (kursi, background interior pesawat, penumpang, meja, sabuk pengaman, smartphone)	
30	Setiap asset dapat berinteraksi dengan pemain apabila disentuh ( <i>touch</i> ) dengan menampilkan opsi pilihan	Membuat fungsi interaksi touch pada setiap aset dengan hasil interaksi berupa opsi pilihan Mendesain ilustrasi setiap opsi pilihan	8
31	Setiap pilihan opsi yang benar akan mengakibatkan satu persyaratan terpenuhi dan merubah tampilan permainan	Membuat fungsi untuk hide dan unhide pada asset sebagai akibat dari pilihan opsi yang benar Mendesain asset dengan dua kondisi yang berbeda, seperti penumpang dengan kondisi berdiri dan duduk dan penumpang dengan posisi memegang HP dengan posisi duduk sempurna.	8
32	Menampilkan instruksi persyaratan apa yang belum terpenuhi apabila pemain menekan tombol "OK" dan membuka tampilan kemenangan apabila semua persyaratan telah terpenuhi.	Membuat fungsi untuk pengecekan persyaratan pada tombol "OK" Mendesain ilustrasi tombol untuk pengecekan persyaratan penumpang dengan label "OK" Mendesain tampilan kemenangan Gameplay PP	13
33	Pemain dapat mengulang kembali permainan PP	Membuat fungsi untuk mengulang permainan PP	2

No	Daftar kebutuhan (fitur)	Tugas	Nilai SP
		Mendesain ilustrasi tombol yang berfungsi untuk mengulang permainan	
34	Pemain dapat kembali ke tampilan pilihan mode permainan	Membuat fungsi untuk memanggil interface Pilihan Permainan. Mendesain ilustrasi tombol yang berfungsi memanggil interface Pilihan Permainan	2
35	Pemain dapat melihat dan menutup tutorial mode permainan PP	Membuat fungsi untuk menampilkan dan menutup jendela tutorial PP Mendesain tampilan tutorial PP	5
36	Tutorial berbentuk barisan teks dan ditampilkan secara bertahap dengan bantuan tombol navigasi untuk menavigasi teks tutorial	Membuat fungsi navigasi untuk tutorial PP Mendesain ilustrasi tombol yang berfungsi untuk navigasi tutorial PP	5
37	Memiliki Tampilan awal permainan yang menjelaskan petunjuk permainan PP	Mendesain tampilan awal permainan	13

#### B. Proyek P-MVN

No	Daftar kebutuhan (fitur)	Tugas	Nilai SP
<i>Interface Home</i>			
1	Berpindah ke Interface Story sesuai dengan pilihan cerita	Membuat fungsi untuk memanggil interface Story	21

No	Daftar kebutuhan (fitur)	Tugas	Nilai SP
		Membuat Ilustrasi Thumbnail yang mewakili karakter di cerita	
2	Pemain dapat menavigasi cerita dengan swipe ke kiri dan kekanan pada area pilihan cerita	Membuat fungsi yang mendeteksi inputan <i>swipe</i> pada area cerita	8
3	Pemain dapat mengetahui update cerita terbaru	Membuat fungsi untuk menampilkan thumbnail cerita terbaru yang ditambahkan ke dalam aplikasi	21
4	Tampilan minimalis pada tampilan home	Mengonsep tampilan pada interface Home	21
<b><i>Interface Story</i></b>			
5	Pemain dapat membaca narasi cerita atau dialog dengan menyentuh layar	Membuat fungsi mendeteksi sentuhan pemain yang berguna untuk menampilkan narasi cerita selanjutnya	34
		Mengonsep alur cerita	
		Membuat ilustrasi untuk background / latar belakang	
		Membuat desain ilustrasi karakter cerita	
		Membuat animasi pada situasi dan kondisi tertentu (contoh saat situasi hujan, maka terdapat animasi petir dan rintik hujan)	

No	Daftar kebutuhan (fitur)	Tugas	Nilai SP
		Membuat frame untuk tempat narasi cerita dan dialog	
6	Pemain dapat menginputkan nama untuk menjadi subjek dalam narasi cerita yang berkaitan dengan pemain atau pada dialog antara NPC dengan pemain	Membuat fungsi input nama pemain	8
		Membuat ilustrasi frame untuk input nama pemain	
7	Pemain dapat kembali ke tampilan Home setelah menyelesaikan cerita	Membuat fungsi untuk memanggil interface Home	8
		Membuat ilustrasi tombol untuk fungsi kembali ke tampilan home	
8	Alur cerita memiliki opsi yang dapat dipilih pemain	Membuat fungsi untuk menampilkan opsi	34
		Membuat ilustrasi tombol untuk tampilan opsi pilihan	
		Merancang event untuk setiap cabang opsi	

### Lampiran 3 – rancangan pengerjaan proyek

Dalam lampiran ini disertakan draf rancangan pengerjaan proyek berdasarkan kebutuhan yang telah disusun sebelumnya. Input atau masukan dari tim pengembang sangat dominan dalam menyusun rancangan pengerjaan proyek seperti estimasi durasi pengerjaan dan total hari dalam satu kali iterasi.

#### A. Proyek P-MCP

Dalam proyek P-MCP, durasi pengerjaan yang diperkirakan oleh salah seorang tim pengembang adalah selama satu bulan dan dengan rasio iterasi adalah setiap dua minggu. Dalam perkiraan satu bulan tersebut, peneliti mendapatkan informasi bahwa pekerjaan masing-masing interface diperkirakan seperti berikut: Interface Home memiliki ekspektasi pengerjaan selama tiga hari, Interface Pilihan Level memiliki ekspektasi pengerjaan juga selama satu minggu, dan Interface Gameplay diekspektasi memakan waktu selama 2.4 minggu dan menjadi prioritas utama. Berdasarkan masukan tersebut maka rancangan pengerjaannya adalah:

Iterasi (2 W)	Jumlah Story Per Iteration	Jumlah Story Point	
		per Story	per Iterasi
1	12 (kebutuhan 12-23)	5, 5, 8, 8, 13, 13, 13, 13, 13, 13, 21, 34 ,34	193
2	12 (kebutuhan 1-6, 7-11, 24)	5, 5, 5, 5, 5, 8, 8, 13, 13, 21, 21	109
<b>Total</b>	24		302
<b>Rerata</b>	12		151

#### B. Proyek P-MCE

Berdasarkan informasi yang didapatkan dari tim pengembang P-MCE, durasi yang diperkirakan untuk mengembangkan proyek P-MCE adalah selama 1 bulan. Prioritas pengerjaan terletak di interface permainan SBB dan PP sehingga,

dalam durasi 3 minggu tim pengembang berfokus untuk mengerjakan kedua interface tersebut. Dan minggu terakhir digunakan pengembang untuk mengerjakan interface Pilihan permainan dan Pilihan Level. Berdasarkan keterangan tersebut maka rancangan pengerjaan untuk proyek P-MCE adalah

Iterasi (2 W)	Jumlah Story Per Iteration	Jumlah Story Point	
		per Story	per Iterasi
1	20 (Kebutuhan 16-32, 35-37)	2, 2, 2, 2, 2, 5, 5, 5, 5, 5, 5, 8, 8, 8, 13, 13, 13, 13, 13, 13, 13, 34	185
2	17 (Kebutuhan 1-15, 33, 34)	2, 2, 2, 2, 2, 2, 2, 2, 2, 8, 8, 8, 8, 8, 8, 13, 21	100
<b>Total</b>	37		285
<b>Rerata</b>	18.5		142.5