



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - KI141502
**PERANCANGAN DAN IMPLEMENTASI
PEMBANGKIT *PUZZLE* OTOMATIS UNTUK
PERMAINAN EDUKASI MENYUSUN KALIMAT
MATEMATIKA**

RIFQI MAULA IQBAL
NRP 5113100018

Dosen Pembimbing
Imam Kuswardayan, S.Kom., M.T.
Anny Yuniarti, S.Kom, M.Comp.Sc.

JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2017



TUGAS AKHIR - KI141502

**PERANCANGAN DAN IMPLEMENTASI
PEMBANGKIT *PUZZLE* OTOMATIS UNTUK
PERMAINAN EDUKASI MENYUSUN KALIMAT
MATEMATIKA**

**RIFQI MAULA IQBAL
NRP 5113100018**

**Dosen Pembimbing
Imam Kuswardayan, S.Kom., M.T.
Anny Yuniarti, S.Kom, M.Comp.Sc.**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2017**

(Halaman Ini Sengaja dikosongkan)



FINAL PROJECT- KI141502

**DESIGN AND IMPLEMENTATION OF
AUTOMATIC PUZZLE GENERATOR FOR
EDUCATION MODULE IN GENERATING
MATHEMATICS EXPRESSION**

**RIFQI MAULA IQBAL
NRP 5113100018**

Advisor

**Imam Kuswardayan, S.Kom., M.T.
Anny Yuniarti, S.Kom, M.Comp.Sc.**

**DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY
SEPULUH NOPEMBER INSTITUTE OF TECHNOLOGY
SURABAYA
2017**

(Halaman Ini Sengaja dikosongkan)

LEMBAR PENGESAHAN

PERANCANGAN DAN IMPLEMENTASI PEMBANGKIT PUZZLE OTOMATIS UNTUK PERMAINAN EDUKASI MENYUSUN KALIMAT MATEMATIKA

Tugas Akhir

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Rumpun Mata Kuliah Interaksi, Grafika, dan Seni
Program Studi S-1 Jurusan Teknik Informatika
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh:

RIFQI MAULA IQBAL

NRP. 5113 100 018

Disetujui oleh Dosen Pembimbing Tugas Akhir
Imam Kuswardayan, S.Kom., M.T.
NIP: 19761215 200312 1 001



Anny Yuniarti, S.Kom, M.Comp.Sc.
NIP: 198106222005012002

**SURABAYA
JUNI, 2017**

PERANCANGAN DAN IMPLEMENTASI PEMBANGKIT PUZZLE OTOMATIS UNTUK PERMAINAN EDUKASI MENYUSUN KALIMAT MATEMATIKA

Nama Mahasiswa : Rifqi Maula Iqbal
NRP : 5113100018
Jurusan : Teknik Informatika FTIf-ITS
Dosen Pembimbing I : Imam Kuswardayan, S.Kom., M.T.
Dosen Pembimbing II : Anny Yuniarti, S.Kom, M.Comp.Sc.

ABSTRAK

Saat ini game memiliki perkembangan yang sangat pesat. Game saat ini juga tidak hanya pada perangkat komputer namun juga telah merambah ke perangkat mobile khususnya dalam sistem operasi Android. Selain itu pula saat ini banyak game yang hanya memprioritaskan unsur adiktif sehingga hanya memiliki unsur adiktif saja dan kurangnya unsur edukatif di dalam game. Padahal pada kenyataannya pengetahuan juga sangatlah penting daripada hanya bermain game. Game yang menggabungkan antara permainan dan unsur edukatif didalamnya disebut dengan game edukasi.

Dalam Tugas Akhir ini dibangun suatu permainan yang bergenre edukasi dengan menerapkan metode pembangkit puzzle otomatis di dalam sistem permainan supaya dapat lebih menambah unsur variatif. Metode ini digunakan untuk membangun suatu puzzle secara otomatis dan variatif baik pada hanya pada materi edukasi. Pembangkitan puzzle secara otomatis ini dilakukan dengan mengacak urutan pattern teka teki silang yang telah diinisialisasi. Kalimat matematika diperoleh secara acak berdasarkan bobot level yang disusun secara berpola dari level minimum hingga level maksimum. Dari hasil uji coba diketahui bahwa game dapat membangkitkan puzzle dengan beragam dan nyaris tanpa ada susunan pattern, soal dan piece

puzzle yang sama dalam, dua puluh kalikali percobaan. Dengan ini diketahui bahwa fungsionalitas game dapat berjalan dengan lancar dan cepat sehingga membuat pemain tidak mudah bosan dalam bermain. Serta pengimplementasian materi edukasi di dalam game dapat membuat wawasan edukasi pemain menjadi bertambah.

Kata kunci: Pembangkit Puzzle Otomatis, Edukatif, Perangkat Mobile, Sistem Operasi Android.

DESIGN AND IMPLEMENTATION OF AUTOMATIC PUZZLE GENERATOR FOR EDUCATION MODULE IN GENERATING MATHEMATICS EXPRESSION

Student Name : Rifqi Maula Iqbal
NRP : 5113100018
Major : Teknik Informatika FTIf-ITS
Advisor I : Imam Kuswardayan, S.Kom., M.T.
Advisor II : Anny Yuniarti, S.Kom, M.Comp.Sc.

ABSTRACT

Today the game is not limited deploy on console device. Games today are not only on the computer but also has deployed on mobile devices, especially in the Android operating system. Beside that, today many games are just so more addictive and less of education in the game. Whereas in fact, knowledge is also very important rather than just playing games. Game that combines games and educational elements in one called educational games.

this Final Project built a game of educational genre by applying the method of automatic puzzle generator in game system in order to more menaqmbah variatif element. This method is used to build a puzzle automatically and varied well on only on educational materials. This automatic puzzle generation is done by randoming the initialized pattern. Mathematical sentences are obtained randomly based on the patterned level weights from the minimum to the maximum. From the results of the trial is known that the game can evoke puzzles with a variety and almost no pattern, the same puzzle problem and piece in a twenty-thousand-year experiment. With this is known that the functionality of the game can run smoothly and quickly so as to make players not easily bored in playing. As well as implementing educational materials in the game can make educational insights into players to grow.

Keywords: Automatic Puzzle Generator, Education, Mobile Device, Android OS.

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Segala puji dan syukur, kehadiran Allah Subhanahu wa ta'ala yang telah memberikan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan tugas akhir yang berjudul “PERANCANGAN DAN IMPLEMENTASI PEMBANGKIT *PUZZLE* OTOMATIS UNTUK PERMAINAN EDUKASI MENYUSUN KALIMAT MATEMATIKA”.

Pengerjaan tugas akhir ini adalah momen bagi penulis untuk mengeluarkan seluruh kemampuan, hasrat, dan keinginan yang terpendam di dalam hati mulai dari masuk kuliah hingga lulus sekarang ini, lebih tepatnya di kampus Teknik Informatika Institut Teknologi Sepuluh Nopember Surabaya.

Dalam pelaksanaan dan pembuatan tugas akhir ini tentunya sangat banyak bantuan-bantuan yang penulis terima dari berbagai pihak. Melalui lembar ini, penulis ingin secara khusus menyampaikan ucapan terima kasih kepada:

1. Allah SWT atas limpahan rahmat dan rezeki-Nya sehingga penulis dapat menyelesaikan Tugas Akhir.
2. Ayah penulis, Gatot Hari Adi, dan Ibu penulis, Mubarroh, yang selalu memberikan dukungan, doa, perhatian, dan kasih sayang.
3. Kakak penulis, Alvina Fitri Amalia, yang selalu memberikan motivasi dan bimbingan selama menyelesaikan studi penulis.
4. Adik tingkat, Miftahul Akhyar, yang telah membantu memfasilitasi sarana teknis yang sulit saya dapatkan.
5. Bapak Imam Kuswardayan selaku dosen pembimbing Tugas Akhir pertama dan yang telah memberikan arahan dalam pengerjaan Tugas Akhir ini.
6. Ibu Anny Yuniarti selaku dosen pembimbing Tugas Akhir kedua yang dengan sabar membimbing penulis dalam pengerjaan Tugas Akhir ini.

7. Ibu Anny Yuniarti selaku dosen wali yang berkenan memberi saran dan arahan selama penulis menjalani studi S1.
8. Bapak Imam Kuswardayan selaku dosen wali semester satu sampai TUJUH yang berkenan memberi motivasi dan arahan selama penulis menjalani studi S1.
9. Bapak Radityo Anggoro selaku dosen koordinator Tugas Akhir yang telah membantu penulis atas segala sesuatu mengenai syarat-syarat dan terlaksananya sidang Tugas Akhir.
10. Bapak Darlis Herumurti selaku Ketua Jurusan Teknik Informatika ITS yang selama ini memberikan bantuan kepada penulis.
11. Dosen-dosen Teknik Informatika yang dengan sabar mendidik dan memberikan pengalaman baru kepada penulis selama di Teknik Informatika.
12. Staf TU Teknik Informatika ITS yang senantiasa memudahkan segala urusan penulis di jurusan.
13. Rekan-rekan dan pengelola Laboratorium Interaksi, Grafik, dan Seni yang telah memberikan fasilitas dan kesempatan melakukan riset atas Tugas Akhir yang dikerjakan penulis.
14. Rekan-rekan dan sahabat-sahabat penulis angkatan 2013 yang memberikan dorongan motivasi dan bantuan kepada penulis.
15. Pihak-pihak lain yang tidak sengaja terlewat dan tidak dapat penulis sebutkan satu per satu.

Penulis telah berusaha sebaik mungkin dalam menyusun tugas akhir ini, namun penulis mohon maaf apabila terdapat kekurangan, kesalahan maupun kelalaian yang telah penulis lakukan. Kritik dan saran yang membangun dapat disampaikan sebagai bahan perbaikan selanjutnya.

Surabaya, Mei 2017
Penulis

Rifqi Maula Iqbal

DAFTAR ISI

LEMBAR PENGESAHAN.....	v
ABSTRAK	vi
ABSTRACT	viii
KATA PENGANTAR.....	x
DAFTAR ISI	xii
DAFTAR GAMBAR.....	xvi
DAFTAR TABEL	xviii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	2
1.4 Tujuan.....	2
1.5 Manfaat.....	3
1.6 Metodologi.....	3
1.7 Sistematika Penulisan	5
BAB II TINJAUAN PUSTAKA.....	7
2.1 Game Maker	7
2.2 Permainan	7
2.3 Game Maker Language.....	8
2.4 <i>Android</i>	8
2.5 <i>Rancang Bangun Perangkat Lunak</i>	9
2.6 <i>Puzzle</i>	10
2.7 <i>Pattern</i>	10
2.8 <i>Kalimat Matematika</i>	10
2.9 <i>Education Games</i>	11
BAB III ANALISIS DAN PERANCANGAN	13
3.1 Sistem	13
3.2 Perancangan Permainan.....	14
3.2.1 Deskripsi Umum Perangkat Lunak	14
3.2.2 Spesifikasi Kebutuhan Fungsional	14
3.2.3 Spesifikasi Kebutuhan Non-Fungsional.....	15
3.2.4 Karakteristik Pengguna	15
3.3 Perancangan Sistem	16

3.3.1	Perancangan Diagram Kasus Penggunaan.....	16
3.3.2	Perancangan Skenario Kasus Penggunaan	17
3.3.3	Perancangan Antarmuka Pengguna	29
3.3.4	Perancangan Kontrol Permainan	35
3.3.5	Perancangan Aturan Permainan.....	35
3.3.6	Perancangan Data	36
	BAB IV IMPLEMENTASI	51
4.1	Lingkungan Implementasi	51
4.2	Implementasi Permainan	51
4.2.1	Implementasi Tampilan Judul Game	52
4.2.2	Implementasi Tampilan Main Menu.....	52
4.2.3	Implementasi Tampilan Play Mode.....	53
4.2.4	Implementasi Tampilan Main Game	53
4.2.5	Implementasi Tampilan Level Complete.....	54
4.2.6	Implementasi Tampilan Level Selector	54
4.2.7	Implementasi Tampilan Failed	55
4.2.8	Implementasi Tampilan Out Of Time.....	55
4.2.9	Implementasi Kasus Penggunaan	56
	BAB V PENGUJIAN DAN EVALUASI	61
5.1	Lingkungan Uji Coba	61
5.2	Pengujian Performa Pembangkitan <i>puzzle</i>	62
5.2.1	Skenario Uji Pembangkitan Puzzle.....	62
5.2.2	Hasil Pengujian Performa	63
5.3	Pengujian Aturan Main Math Scrabble	64
5.3.1	Skenario Uji Aturan Main Math Scrabble.	64
5.3.2	Hasil Pengujian Aturan Main Math Scrabble Pembangkitan <i>Puzzle</i>	65
5.4	Pengujian pengaturan tingkat kesulitan Math Scrabble..	67
5.4.1	Skenario Uji pengaturan tingkat kesulitan Math Scrabble. 67	
5.4.2	Hasil Uji pengaturan tingkat kesulitan Math Scrabble. 67	
5.5	Pengujian Fungsionalitas dengan metode black-box.....	70
5.5.1	Skenario Pengujian Fungsionalitas.....	70
5.6	Pengujian Pengguna	75

5.6.1	Skenario Uji Coba Pengguna	75
5.6.2	Daftar Penguji Perangkat Lunak	76
5.6.3	Hasil Uji Coba Pengguna	76
5.6.4	Hasil Pengujian Pengguna	78
	BAB VI KESIMPULAN DAN SARAN.....	82
6.1.	Kesimpulan.....	82
6.2.	Saran	83
	DAFTAR PUSTAKA	84
	LAMPIRAN 1	86
	LAMPIRAN 2	98
	LAMPIRAN 3.....	93
	BIODATA PENULIS.....	101

(Halaman Ini Sengaja Dkosongkan)

DAFTAR GAMBAR

Gambar 3.1 Diagram kasus aplikasi	17
Gambar 3.2 Diagram aktivitas Memilih Level selector mode.....	24
Gambar 3.3 Diagram aktivitas Memilih Native Mode	25
Gambar 3.4 Diagram aktivitas Menyelesaikan level.....	26
Gambar 3.5 Diagram aktivitas Mengulangi level.....	27
Gambar 3.6 Diagram aktivitas Kembali ke menu	28
Gambar 3.7 Diagram aktivitas Menandai level terselesaikan	29
Gambar 3.8 Tampilan Judul <i>game</i>	32
Gambar 3.9 Tampilan Menu Permainan.....	32
Gambar 3.10 Tampilan Play Mode.....	33
Gambar 3.11 Tampilan Main game	33
Gambar 3.12 Tampilan Level Complete	34
Gambar 3.13 Tampilan Level Selector(1) dan Level Selector (2)	34
Gambar 3.14 Tampilan Failed.....	35
Gambar 3.15 Tampilan Out of time.....	35
Gambar 3.16 <i>Pattern</i> untuk satu soal, digunakan pada level satu sampai lima, terdiri dari dua macam <i>pattern</i>	36
Gambar 3.17 <i>Pattern</i> untuk dua soal, digunakan pada level enam sampai 15, terdiri dari 13 macam <i>pattern</i>	37
Gambar 3.18 <i>Pattern</i> untuk tiga soal, digunakan pada level 16 sampai 28, terdiri dari 26 macam <i>pattern</i>	38
Gambar 3.19 <i>Pattern</i> untuk empat soal, digunakan pada level 29 sampai 40, terdiri dari 14 macam <i>pattern</i>	39
Gambar 3.20 <i>Pattern</i> untuk lima soal, digunakan pada level 41 sampai 60, terdiri dari 20 macam <i>pattern</i>	40
Gambar 3.21 <i>Pattern</i> untuk enam soal, digunakan pada level 61 sampai 85, terdiri dari 20 macam <i>pattern</i>	41
Gambar 3.22 <i>Pattern</i> untuk tujuh soal, digunakan pada level 86 sampai 100, terdiri dari 12 macam <i>pattern</i>	42

Gambar 5.1 Tampilan di mana pemain dinyatakan kalah dengan status ‘Failed’ apabila ada <i>piece puzzle</i> yang letaknya kurang tepat yakni <i>piece puzzle</i> 27 dan = yang posisinya tertukar, sehingga kalimat matematika salah.....	65
Gambar 5.2 Tampilan di mana pemain dinyatakan kalah dengan status out of time dikarenakan waktu habis (Lihat time left)	65
Gambar 5.3 Tampilan di mana pemain memenangkan permainan atau berhasil menyelesaikan level, terbukti dengan letak <i>piece puzzle</i> yang letaknya tepat dan diselesaikan dalam waktu yang mencukupi atau tidak melebihi batas waktu.....	66
Gambar 5.4 Tampilan pengujian kasus penggunaan memilih level selector	71
Gambar 5.5 Tampilan pengujian kasus penggunaan memilih native mode	72
Gambar 5.6 Tampilan Pengujian Kasus Penggunaan Menyelesaikan Level.....	72
Gambar 5.7 Tampilan pengujian kasus penggunaan Mengulangi Level.....	73
Gambar 5.8 Tampilan pengujian kasus penggunaan Kembali ke Menu.....	73
Gambar 5.9 Tampilan Pengujian Kasus Penggunaan Menandai Level terselesaikan.....	74

DAFTAR TABEL

Tabel 3.1 Karakteristik Pengguna	16
Tabel 3.2 Skenario Kasus Penggunaan.....	18
Tabel 3.3 Skenario Kasus Penggunaan Memilih Level selector mode	19
Tabel 3.4 Skenario Kasus Penggunaan Memilih Native Mode ...	20
Tabel 3.5 Skenario Kasus Penggunaan Menyelesaikan level.....	21
Tabel 3.6 Skenario Kasus Penggunaan Mengulangi level.....	21
Tabel 3.7 Skenario Kasus Penggunaan Kembali ke menu	22
Tabel 3.8 Skenario Kasus Penggunaan Menandai level terselesaikan	23
Tabel 3.9 Detil Perancangan Data untuk materi edukasi tiap level untuk selengkapnya bisa dilihat pada Lampiran 3.....	43
Tabel 3.10 Detil Perancangan Data untuk permainan tiap level.	43
Tabel 4.1 Lingkungan Implementasi Perangkat Lunak	51
Tabel 5.1 Lingkungan Ujicoba Perangkat Lunak (bagian 1).....	61
Tabel 5.2 Lingkungan Ujicoba Perangkat Lunak (bagian 2).....	61
Tabel 5.3 Skenario Pengujian Performa	62
Tabel 5.4 Hasil Uji Coba Performa Pembangkitan <i>Puzzle</i>	63
Tabel 5.5 Skenario Pengujian Aturan Main	64
Tabel 5.6 Hasil Pengujian pengaturan tingkat kesulitan	67
Tabel 5.7 Pengujian Permainan.....	70
Tabel 5.8 Hasil Pengujian Fungsionalitas	74
Tabel 5.9 Daftar Nama Penguji Coba Aplikasi	76
Tabel 5.10 Penilaian Antarmuka	77
Tabel 5.11 Penilaian Performa Sistem	77
Tabel 5.12 Penilaian Materi Edukasi.....	78
Tabel 5.13 Rekapitulasi Hasil Uji Coba Pengguna	79

(Halaman Ini Sengaja dikosongkan)

BAB I

PENDAHULUAN

Bab ini memaparkan garis besar Tugas Akhir yang meliputi latar belakang, tujuan dan manfaat pembuatan, rumusan dan batasan permasalahan, metodologi pembuatan Tugas Akhir, dan sistematika penulisan.

1.1 Latar Belakang

Teknologi *smartphone* yang kini sedang berkembang pesat di tengah masyarakat, membuat interaksi antar sesama manusia menjadi lebih mudah dan sangat cepat. Interaksi tersebut dapat melalui media telepon, pesan singkat, dan surat elektronik. Pengguna perangkat *smartphone* juga menggunakannya sebagai media hiburan seperti berselancar di internet atau memainkan *game*.

Permainan adalah sebuah aktivitas yang memerlukan paling sedikit satu pemain, mempunyai aturan, dan sebuah kondisi menang. Serta komponen terpenting yang harus ada pada permainan adalah level. Desain level permainan secara manual memakan waktu yang cukup lama, sehingga dibutuhkan pembangkitan level secara dinamis dimana berarti level harus otomatis dan variatif. Pendekatan ini dapat menghemat waktu pembuatan dari sebuah level.

Selain itu pula saat ini banyak permainan yang hanya memprioritaskan unsur adiktif sehingga membuat kurangnya unsur edukatif di dalam permainan. Pengaruh dari permainan tersebut membuat pemain menjadi memiliki rasa yang kurang mementingkan edukasi sehingga banyak yang masih anak-anak sudah menghabiskan uang dan waktu demi sebuah *game* daripada belajar. Padahal pada kenyataannya pengetahuan juga sangatlah penting daripada hanya bermain *game*. Permainan yang menggabungkan antara permainan dan unsur edukatif didalamnya disebut dengan permainan edukasi.

Tugas akhir ini membangun permainan yang mampu membangkitkan *puzzle* secara dinamis. Permainan yang dibuat mengandung unsur edukasi bagi para pengguna yang bermain. Serta

permainan ini akan dijalankan dan dimainkan pada perangkat *smartphone* dengan sistem operasi *Android*. Permainan ini diberi nama Math Scrabble.

1.2 Rumusan Masalah

Rumusan masalah yang diangkat dalam tugas akhir ini adalah sebagai berikut:

1. Bagaimana merancang aturan main pada Math Scrabble?
2. Bagaimana skenario *puzzle* dan kalimat matematika yang dibangkitkan secara otomatis pada permainan edukasi berbasis android?
3. Bagaimana skenario permainan Math Scrabble?
4. Bagaimana mengatur tingkat kesulitan permainan Math Scrabble?

1.3 Batasan Masalah

Permasalahan yang dibahas dalam tugas akhir ini memiliki beberapa batasan, di antaranya sebagai berikut:

1. Soal pada aplikasi ini hanya berisi operasi matematika yang berupa perkalian, penjumlahan, pembagian, dan pengurangan
2. Angka baik itu yang digunakan pada soal maupun jawaban berkisar antara 1 sampai dengan 99.
3. Bagian yang dilakukan *random generate* pada permainan ini hanyalah, pola *puzzle*, posisi *puzzle*, soal, jawaban dan letak kotak kosong.

1.4 Tujuan

Tujuan dari pembuatan tugas akhir ini adalah untuk membangun suatu sistem *game* yang mengimplementasikan pembangkit *puzzle* otomatis untuk materi edukasi. Serta meningkatkan pemikiran yang strategis, cermat terhadap pengguna, dan memberikan sensasi *animonoton*.

1.5 Manfaat

Manfaat dari hasil pembuatan tugas akhir ini antara lain:

1. Mendapatkan pengetahuan dan wawasan edukasi di dalam *game Math Scrabble* bagi pengguna.
2. Meningkatkan pemikiran strategis dan cermat kepada pengguna.
3. Memberikan media hiburan bagi para pengguna.
4. Memberikan sensasi antimonoton pada permainan edukasi

1.6 Metodologi

Pembuatan tugas akhir dilakukan menggunakan metodologi sebagai berikut:

A. Studi literatur

Tahap Studi Literatur merupakan tahap pembelajaran dan pengumpulan informasi yang digunakan untuk mengimplementasikan Tugas Akhir. Tahap ini diawali dengan pengumpulan literatur, diskusi, eksplorasi teknologi dan pustaka, serta pemahaman dasar teori yang digunakan pada topik tugas akhir. Literatur-literatur yang dimaksud disebutkan sebagai berikut:

1. Pembangkit *Puzzle* Dinamis;
2. Game Maker Language(GML);
3. Instalasi pada *Platform* Perangkat Mobile Android;
4. Kalimat Matematika;
5. Math Pieces (*Game* serupa).

B. Perancangan perangkat lunak

Pada tahap ini dilakukan analisa awal dan pendefinisian kebutuhan sistem untuk mengetahui permasalahan yang sedang dihadapi. Selanjutnya, dirumuskan rancangan sistem yang dapat memberi solusi terhadap permasalahan tersebut. Langkah yang akan digunakan pada tahap ini adalah sebagai berikut:

1. Pencarian dan pendataan materi yang akan digunakan dalam *game Math Scrabble*.
2. Perancangan sistem dan mekanisme *game Math Scrabble*.
3. Analisis kebutuhan non fungsional.

4. Perancangan pembangkit *puzzle* dinamis untuk karakter pada *game Math Scrabble*.

C. Implementasi dan pembuatan sistem

Tahap implementasi merupakan tahap untuk membangun aplikasi permainan beserta sistem yang terkait. Aplikasi ini akan dibangun dengan bahasa pemrograman GML(Game Maker Language) dengan IDE Game Maker Standard 1.41763. Aplikasi yang akan dibangun berbasis perangkat mobile *Android*.

D. Uji coba dan evaluasi

Pada tahap ini akan dilakukan pengujian terhadap perangkat lunak menggunakan data atau skenario yang telah dipersiapkan sebelumnya yakni sebagai berikut:

1. Pengujian blackbox

Pengujian blackbox adalah pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, tester dapat mendefinisikan kumpulan kondisi input dan melakukan pengetestan pada spesifikasi fungsional program. Pengujian ini dilakukan untuk menguji apakah proses kinerja aplikasi *game* ini sudah sesuai dengan kebutuhan pengguna atau tidak.

2. Pengujian usabilitas

Pengujian usabilitas dilakukan dengan cara melakukan survei ke pengguna yaitu beberapa pengguna yang suka bermain *game* khususnya yang berusia dibawah 13 tahun. Survei dilakukan untuk mengukur tingkat dinamisasi dan *antimonoton* dari aplikasi yang dibuat, sehingga pengguna dalam menyusun kalimat matematika tidak ada unsur menghafal dan benar benar berhitung.

E. Penyusunan laporan tugas akhir

Pada tahap ini dilakukan penyusunan laporan yang berisi dasar teori, dokumentasi dari perangkat lunak, dan hasil-hasil yang diperoleh selama pengerjaan tugas akhir.

1.7 Sistematika Penulisan

Buku tugas akhir ini terdiri dari beberapa bab, yang dijelaskan sebagai berikut.

BAB I PENDAHULUAN

Bab ini berisi latar belakang masalah, rumusan dan batasan permasalahan, tujuan dan manfaat pembuatan tugas akhir, metodologi yang digunakan, dan sistematika penyusunan tugas akhir.

BAB II TINJAUAN PUSTAKA

Bab ini membahas dasar pembuatan dan beberapa teori penunjang yang berhubungan dengan pokok pembahasan yang mendasari pembuatan tugas akhir ini.

BAB III ANALISIS DAN PERANCANGAN

Bab ini membahas analisis dari sistem yang dibuat meliputi analisis permasalahan, deskripsi umum perangkat lunak, spesifikasi kebutuhan, dan identifikasi pengguna. Kemudian membahas rancangan dari sistem yang dibuat meliputi rancangan skenario kasus penggunaan, arsitektur, data, dan antarmuka.

BAB IV IMPLEMENTASI

Bab ini membahas implementasi dari rancangan sistem yang dilakukan pada tahap perancangan. Penjelasan implementasi meliputi implementasi pembangkitan area permainan, dan antarmuka permainan.

BAB V PENGUJIAN DAN EVALUASI

Bab ini membahas pengujian dari aplikasi yang dibuat dengan melihat keluaran yang dihasilkan oleh aplikasi dan evaluasi untuk mengetahui kemampuan aplikasi.

BAB VI PENUTUP

Bab ini berisi kesimpulan dari hasil pengujian yang dilakukan serta saran untuk pengembangan aplikasi selanjutnya.

BAB II

TINJAUAN PUSTAKA

Pada bab ini akan dibahas mengenai teori-teori yang menjadi dasar dari pembuatan tugas akhir. Teori-teori tersebut adalah *Game Maker Language*, *Game Maker*, permainan, *Android*, rancang bangun perangkat lunak, *puzzle*, *pattern*, kalimat matematika, dan *education games*.

2.1 Game Maker

Game Maker Studio adalah *game engine* untuk pengembangan game 2D yang dibuat oleh YoYoGame. Game Maker menggunakan konsep *visual coding interface* berbasis *drag and drop*, sehingga mudah untuk digunakan oleh pemula. Game Maker menyediakan bahasa pemrograman untuk user yang memiliki kemampuan *coding*, yaitu GameMaker Language (GML). Versi Master dari Game Maker dapat digunakan dalam membuat aplikasi untuk berbagai macam *platform* seperti Android, HTML5, iOS, Playstation Vita, Playstation 3, Playstation 4, Xbox ONE, Ubuntu, Windows dan platform-platform lain.outputnya.

2.2 Permainan

Menurut Scott Rogers permainan adalah sebuah aktivitas yang memerlukan paling sedikit satu pemain, mempunyai aturan, dan kondisi menang. Bermain dengan bola tangan hanya membuang waktu, namun dapat menjadi sebuah permainan jika terdapat peraturan dan tujuan. Si pemain hanya dapat melempar dengan tangan kanan dan menangkap dengan tangan kiri, dan tidak boleh menjatuhkan bola. Kondisi menang terjadi ketika si pemain berhasil menangkap bola sebanyak sepuluh kali tanpa jatuh. Kondisi gagal terjadi ketika si pemain melakukan pelanggaran terhadap peraturan atau kondisi menang. Sebuah aktivitas permainan tercipta ketika semua kriteria terpenuhi.

Hal yang pertama kali harus dilakukan oleh pengembang untuk membangun sebuah permainan yaitu menemukan ide. Dari

penemuan ide tersebut, pengembang harus mengembangkannya menjadi cerita atau alur. Alur permainan dapat membantu pengembang melakukan perancangan sistem. Sistem tersebut meliputi “Tiga C” yaitu *character*, *camera* dan *control*. Selain “Tiga C”, pengembang juga harus melakukan perancangan HUD dan *icon*. Semua aspek tersebut harus dilakukan secara sistematis agar tercipta suatu permainan yang baik dan menyenangkan.

2.3 Game Maker Language

Game Maker Language adalah merupakan bahasa pemrograman built-in yang digunakan GameMaker: Studio. Ini berguna agar pengguna menjadi lebih akrab dengan GameMaker: Studio dan ingin menggunakannya sepenuhnya, disarankan untuk mulai belajar menggunakan bahasa ini, karena ini akan memperluas kemungkinan pengguna saat membuat game.

Sebuah skrip (seperti fungsi built-in) dapat mengambil variabel masukan yang berbeda, dengan nilai angka, string, boolean dan khusus seperti semua atau tidak ada yang diterima, dan variabel input ini umumnya disebut argumen, meskipun juga disebut parameter. Untuk menjalankan skrip dari objek atau *timeline* pengguna dapat menggunakan tindakan *Execute Script*, namun pengguna juga dapat menyebutnya dalam kotak kode dengan menggunakan nama skrip yang sebenarnya seolah-olah itu adalah fungsi atau bahkan dengan menggunakan skrip fungsi *GML_execute*. Jika pengguna menggunakan tindakan skrip pengguna dapat menentukan skrip yang ingin pengguna jalankan, bersama dengan hingga lima argumen, namun jika pengguna menggunakan *editor* kode untuk memanggil skrip, pengguna dapat mengirimkan hingga enam belas argumen.

2.4 Android

Android adalah sistem operasi berbasis Linux yang dirancang untuk perangkat seluler layar sentuh seperti telepon pintar dan komputer tablet. *Android* awalnya dikembangkan oleh *Android, Inc.*,

dengan dukungan finansial dari Google, yang kemudian membelinya pada tahun 2005. Sistem operasi ini dirilis secara resmi pada tahun 2007, bersamaan dengan didirikannya Open Handset Alliance, konsorsium dari perusahaan-perusahaan perangkat keras, perangkat lunak, dan telekomunikasi yang bertujuan untuk memajukan standar terbuka perangkat seluler. Ponsel *Android* pertama mulai dijual pada bulan Oktober 2008.

Antarmuka pengguna *Android* didasarkan pada manipulasi langsung, menggunakan masukan sentuh yang serupa dengan tindakan di dunia nyata, seperti menggesek, mengetuk, mencubit, dan membalikkan cubitan untuk memanipulasi obyek di layar. *Android* adalah sistem operasi dengan sumber terbuka, dan Google merilis kodenya di bawah Lisensi Apache. Kode dengan sumber terbuka dan lisensi perizinan pada *Android* memungkinkan perangkat lunak untuk dimodifikasi secara bebas dan didistribusikan oleh para pembuat perangkat, operator nirkabel, dan pengembang aplikasi.

Selain itu, *Android* memiliki sejumlah besar komunitas pengembang aplikasi (apps) yang memperluas fungsionalitas perangkat, umumnya ditulis dalam versi kustomisasi bahasa pemrograman Java. Pada bulan Oktober 2012, ada sekitar 700.000 aplikasi yang tersedia untuk *Android*, dan sekitar 25 juta aplikasi telah diunduh dari Google Play, toko aplikasi utama *Android*. Sebuah survei pada bulan April-Mei 2013 menemukan bahwa *Android* adalah platform paling populer bagi para pengembang, digunakan oleh 71% pengembang aplikasi seluler. Sehingga *Android* sangat digunakan dalam era saat ini.

2.5 Rancang Bangun Perangkat Lunak

Rancang bangun perangkat lunak merupakan tahap-tahap teknis untuk membangun perangkat lunak yang melingkupi analisis permasalahan dan kebutuhan, perencanaan, analisis sistem, implementasi, serta aktivitas pengujian dan pemeliharaan perangkat lunak. Rancang bangun perangkat lunak diperlukan untuk menentukan konsep, strategi, dan praktik yang baik diterapkan untuk

menciptakan perangkat lunak yang berkualitas tinggi, sesuai anggaran biaya, mudah dalam pemeliharannya, serta tidak membutuhkan waktu yang lama dalam pembangunannya. Beberapa model rancang bangun perangkat lunak yang terkenal dan banyak dipakai antara lain model air terjun dan model iterasi.

2.6 *Puzzle*

Puzzle (dalam *game* ini) adalah objek yang berisikan angka, *operator* , ataupun konjungsi yang berbentuk kotak baik itu yang dapat dipindahkan maupun tidak.

2.7 *Pattern*

Pattern adalah posisi utama dari *puzzle puzzle* setelah terbentuk kalimat matematika, pattern mirip dengan bentuk dari kotak kotak yang tersusun pada permainan teka teki silang pattern di sini terbentuk oleh sejumlah n kotak mendatar terurut dan m kotak menurun yang di mana $|m-n| < 2$. Pattern inilah yang nantinya diacak sejumlah n kemungkinan, yang mana setiap jumlah x kalimat matematika memiliki banyaknya macam pattern masing masing.

2.8 *Kalimat Matematika*

Kalimat Matematika adalah sebuah pernyataan yang tersusun oleh lambang bilangan dan sebuah konjungsi. Berdasarkan adanya variable kalimat matematika dibedakan menjadi dua yakni.

1. Kalimat Matematika terbuka

Kalimat matematika terbuka adalah kalimat matematika yang terdapat variable di dalamnya, sehingga untuk menentukan nilai kebenarannya perlu adanya nilai yang diinisialisasikan untuk mengganti variable., contoh $2+x=4$, apabila x diganti dengan angka dua maka kalimat matematika tersebut bernilai benar, selain yang senilai dengan 2 maka akan menimbulkan nilai salah pada kalimat matematika tersebut

2. Kalimat Matematika tertutup

Kalimat matematika tertutup adalah kalimat matematika yang tidak ada variable di dalamnya sehingga bersifat tertutup untuk

semua nilai contoh $2 + 2 = 4$, di sini kalimat matematika sudah bernilai benar dikarenakan nilai di ruas kiri sama dengan nilai di ruas kanan.

2.9 *Education Games*

Education game adalah game secara eksplisit dirancang dengan tujuan pendidikan, atau yang memiliki nilai pendidikan insidental atau sekunder . Semua jenis permainan dapat digunakan di lingkungan pendidikan. *Education game* juga bisa didefinisikan sebagai permainan yang dirancang untuk membantu orang untuk belajar tentang mata pelajaran tertentu, memperluas konsep, memperkuat pengembangan, memahami sebuah peristiwa historis atau budaya, atau membantu mereka dalam mempelajari keterampilan saat mereka bermain.

(Halaman Ini Sengaja dikosongkan)

BAB III ANALISIS DAN PERANCANGAN

Pada bab ini akan dibahas mengenai analisis dan perancangan yang akan digunakan untuk menyelesaikan tugas akhir.

3.1 Sistem

Permainan video berkembang sangat pesat akibat tuntutan perkembangan zaman. Masyarakat berminat terhadap permainan video yang tentunya menyenangkan. Salah satu faktor yang mempengaruhi tingkat kesenangan pemain dari permainan digital yaitu *map* atau *room* permainan. *map* permainan yang beragam tentunya membutuhkan variasi tingkat kesulitan pada permainan. Pembuatan *map* permainan secara manual dengan *modeling* memakan waktu cukup lama. Dibutuhkan suatu sistem yang dapat membangkitkan *map* permainan secara dinamis setiap kali suatu *map* ingin diciptakan.

Aplikasi ini dibangun dengan tujuan membantu para pemain dapat menikmati permainan yang variatif dan otomatis sekaligus dapat menambah ilmu pembelajaran yang edukatif. *Map* yang dalam permainan ini adalah *puzzle* dibangun secara dinamis berdasarkan tingkat kesulitan masing-masing. Data yang digunakan untuk membangun adalah pembobotan setiap level. Semakin tinggi level yang dipilih maka semakin tinggi pula pembobotan level. Pembobotan level merepresentasikan tingkat kesulitan level. Aplikasi ini diharapkan dapat memberikan pembelajaran yang menyenangkan kepada para pengguna yaitu mengenai pengerjaan persoalan yang matematis dan terarah tanpa meninggalkan unsur Fun pada suatu game.

Penulis menggunakan teknologi Game Maker 1.4 dengan Bahasa pemrograman Game Maker Language untuk memfasilitasi pengembangan permainan.. Aplikasi yang dibangun dengan pembangkit *puzzle* otomatis permainan yang sudah dibangkitkan sebelumnya yang membuat penulis tidak perlu lagi mendesain secara manual.

3.2 Perancangan Permainan

3.2.1 Deskripsi Umum Perangkat Lunak

Tugas akhir yang akan dikembangkan adalah sebuah permainan 2D bergenre edukasi. *Map (puzzle)* permainan dibangkitkan secara dinamis dimana meliputi karakteristik otomatis dan variatif dengan menggunakan pembobotan yang terarah dan meningkat. *Map* yang otomatis dimaksudkan bahwa *map* dibangun dengan mengacak pattern pertama sampai pattern terakhir sesuai dengan jumlah soal dan pembangkitan *Map* tidak dilakukan secara manual atau secara satu per satu. Serta variatif dimaksudkan agar komponen setiap level dibangun secara acak namun tetap mengikuti aturan pembobotan yang meningkat seiring meningkatnya level. Permainan ini serta akan dijalankan dan dimainkan pada perangkat *smartphone* Android agar dapat dimainkan secara fleksibel.

Pengguna utama dari permainan ini adalah anak usia dibawah tigabelas tahun.. Pemain dapat memilih dua cara bermain yakni **Native Mode** dan **Level Selector**. Pada **Native Mode** pemain bermain mulai dari level satu. Pada level ini pemain berinteraksi dengan materi edukasi yakni persoalan matematis, yang mana soal soal berupa puzzle dibangkitkan secara otomatis sehingga apabila diakses ulang akan berbeda letak posisi dan bentuknya . Selanjutnya pada **Level Selector** juga dibangkitkan puzzle dinamis. Namun pada menu ini pemain dapat langsung memilih untuk memulai pada level tertentu.

3.2.2 Spesifikasi Kebutuhan Fungsional

Berdasarkan deskripsi umum sistem, maka disimpulkan bahwa kebutuhan fungsional dari aplikasi ini hanya ada satu yaitu bermain *game*.

3.2.3 Spesifikasi Kebutuhan Non-Fungsional

Terdapat beberapa kebutuhan non-fungsional yang apabila dipenuhi, dapat meningkatkan kualitas dari permainan ini. Berikut daftar kebutuhan non-fungsional:

1. *FrameRate*

Permainan ini harus mampu dimainkan secara lancar, tidak ada *lag* dan nyaman di mata. Sebagian permainan 2D biasanya optimal pada *Frame Rate* 24-30 fps (*Frame per Second*). Namun untuk pembangkitan level secara dinamis dapat mempengaruhi kelancaran *Frame Rate* di awal pembentukan. Serta kelancaran *Frame Rate* dipengaruhi dan tergantung oleh spesifikasi *smartphone* yang akan digunakan.

2. Kebutuhan Grafis

Kenyamanan bermain berbanding lurus dengan kualitas grafis yang disajikan dalam permainan. Efek seperti animasi merupakan salah satu daya tarik dalam suatu permainan. Efek-efek ini bisa membuat *drop rate fps* dan permainan melambat (*lag*), karena membutuhkan tambahan komputasi.

3.2.4 Karakteristik Pengguna

Berdasarkan deskripsi umum diatas, maka dapat diketahui bahwa pengguna yang akan menggunakan aplikasi ini ada dua orang, yaitu pemain yang memainkan permainan, dan pengembang. Karakteristik pengguna tercantum dalam Tabel 3.1.

Tabel 3.1 Karakteristik Pengguna

Nama Aktor	Tugas	Hak Akses Aplikasi	Kemampuan yang harus dimiliki
Pemain	Pihak luar yang memainkan permainan.	Memainkan permainan	Tidak ada

3.3 Perancangan Sistem

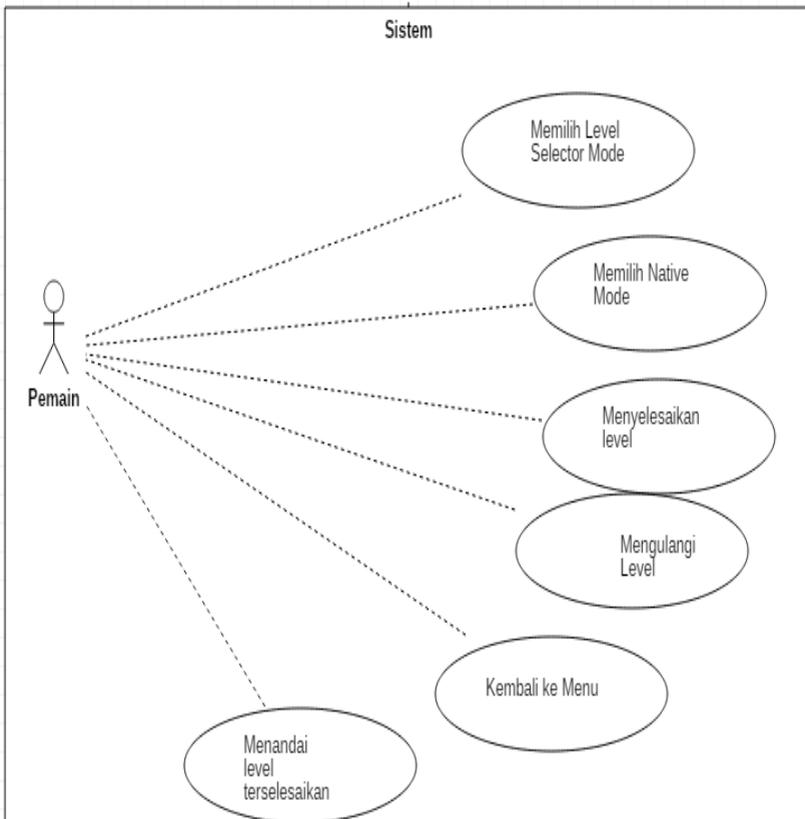
Tahap perancangan dalam subbab ini dibagi menjadi beberapa bagian yaitu perancangan diagram kasus penggunaan, perancangan skenario kasus penggunaan, perancangan arsitektur, perancangan antarmuka, perancangan kelas diagram, perancangan penyimpanan, perancangan, *storyboard*, perancangan data, dan perancangan diagram alur permainan.

3.3.1 Perancangan Diagram Kasus Penggunaan

Dalam aplikasi tugas akhir ini, terdapat enam kasus penggunaan. enam kasus penggunaan ini diantaranya adalah memilih level selector mode, memilih native mode, menyelesaikan level, menguallangi level, Kembali ke menu, dan menandai level terselesaikan. Pembangkitan *puzzle* dinamis permainan dijadikan satu dengan beberapa kasus penggunaan karena sebagian besar perancangan perangkat lunak ini menggunakan pembangkit *puzzle* dinamis yang diterapkan pada layar-layar tertentu. Pengguna atau entitas luar dari sistem adalah pemain yang memainkan permainan.

3.3.2 Perancangan Skenario Kasus Penggunaan

Kasus penggunaan yang terdapat didalam sistem dicantumkan pada Gambar 3.1.



Gambar 3.1 Diagram kasus aplikasi

Penjelasan dari masing-masing kasus penggunaan dicantumkan pada Tabel 3.2. Tabel tersebut berisi penjelasan skenario yang akan dilakukan ketika pengujian.

Tabel 3.2 Skenario Kasus Penggunaan

No	Kode Kasus Penggunaan	Nama Kasus Penggunaan	Keterangan
1	UC-001	Memilih Level selector mode	Untuk bermain dengan memilih level tertentu atau loncat ke level tertentu.
2	UC-002	Memilih Native Mode	Untuk bermain dengan memulai dari level satu.
3	UC-003	Menyelesaikan level	Untuk menyelesaikan semua kalimat matematika yang ada di level tertentu , sehingga bisa melaju ke level selanjutnya.
4	UC-004	Mengulangi level	Untuk mengulangi level yang gagal diselesaikan.
5	UC-005	Kembali ke menu	Untuk mengembalikan ke tampilan menu
6	UC-006	Menandai level terselesaikan	Untuk mengubah kotak pemilih level menjadi putih apabila level pernah diselesaikan.

3.3.2.1 Kasus Penggunaan Permainan

Penjelasan kasus penggunaan permainan untuk skenario UC-001 yakni Memilih Level selector mode dijelaskan pada Tabel 3.3.

Tabel 3.3 Skenario Kasus Penggunaan Memilih Level selector mode

Nama Kasus Penggunaan	Memilih Level selector mode
Kode	UC-001
Deskripsi	Kasus penggunaan dimana aktor memilih level tertentu atau loncat ke level tertentu.
Aktor	Pemain.
Kondisi Awal	Pemain sudah masuk ke aplikasi dan muncul layar Judul . Lalu sistem menampilkan tampilan Main menu
Alur Normal	<ol style="list-style-type: none"> 1. Pemain mengklik tombol play. 2. Sistem menampilkan tampilan play mode. 3. Pemain mengklik tombol level selector. 4. Sistem menampilkan pilihan level berupa kotak kotak yang berisi angka 5. Pemain mengklik salah satu kotak berisikan angka 6. Sistem menampilkan tampilan Main game pada level terpilih

Selanjutnya penjelasan kasus penggunaan permainan untuk skenario UC-002 yakni Memilih Native Mode dijelaskan pada Tabel 3.4.

Tabel 3.4 Skenario Kasus Penggunaan Memilih Native Mode

Nama Kasus Penggunaan	Memilih Native Mode
Kode	UC-002
Deskripsi	Kasus penggunaan dimana aktor bermain dengan memulai dari level satu
Aktor	Pemain.
Kondisi Awal	Pemain sudah masuk ke aplikasi dan muncul layar Judul . Lalu sistem menampilkan tampilan Main menu.
Alr Normal	<ol style="list-style-type: none"> 1. Pemain mengklik tombol play. 2. Sistem menampilkan tampilan play mode. 3. Pemain mengklik tombol native mode. 4. Sistem menampilkan tampilan Main game pada level satu

Kemudian penjelasan kasus penggunaan permainan untuk skenario UC-003 yakni Menjawab Soal Battle dijelaskan pada Tabel 3.5

Tabel 3.5 Skenario Kasus Penggunaan Menyelesaikan level

Nama Kasus Penggunaan	Menyelesaikan level
Kode	UC-003
Deskripsi	Kasus penggunaan dimana aktor menyelesaikan semua kalimat matematika yang ada di level tertentu , sehingga bisa melaju ke level selanjutnya.
Aktor	Pemain.
Kondisi Awal	Pemain sudah masuk ke aplikasi dan muncul tampilan Level .
Alur Normal	<ol style="list-style-type: none"> 1. Pemain melakukan <i>drag and drop puzzle</i> tersebut ke petak yang dianggap tepat sehingga membentuk kalimat matematika yang benar. 2. Pemain menekan tombol submit 3. Sistem menampilkan tulisan Level Complete jika benar, danj tulisan Failed jika salah

Selanjutnya penjelasan kasus penggunaan permainan untuk skenario UC-004 yakni Mengulangi level dijelaskan pada Tabel 3.6.

Tabel 3.6 Skenario Kasus Penggunaan Mengulangi level

Nama Kasus Penggunaan	Mengulangi level
Kode	UC-004
Deskripsi	Kasus penggunaan dimana aktor mengulangi level yang gagal diselesaikan
Aktor	Pemain.
Kondisi Awal	Sistem menampilkan tulisan Failed di layar.
Alur Normal	<ol style="list-style-type: none"> 1 Pemain memilih tombol ‘RESTART LEVEL’ dan menekan klik. 2 Sistem kembali ke tampilan Main Menu dan mengulangi level tersebut

Kemudian penjelasan kasus penggunaan permainan untuk skenario UC-005 yakni Kembali ke menu dijelaskan pada Tabel 3.7.

Tabel 3.7 Skenario Kasus Penggunaan Kembali ke menu

Nama Kasus Penggunaan	Kembali ke menu
Kode	UC-005
Deskripsi	Kasus penggunaan dimana aktor Untuk mengembalikan ke tampilan menu
Aktor	Pemain.
Kondisi Awal	Sistem menampilkan level tertentu
Alur Normal	<ol style="list-style-type: none"> 1 Pemain memilih tombol 'BACK TO MENU' dan menekan klik. 2 Sistem menampilkan tampilan Main menu

Selanjutnya penjelasan kasus penggunaan permainan untuk skenario UC-006 yakni Menandai level terselesaikan dijelaskan pada Tabel 3.8.

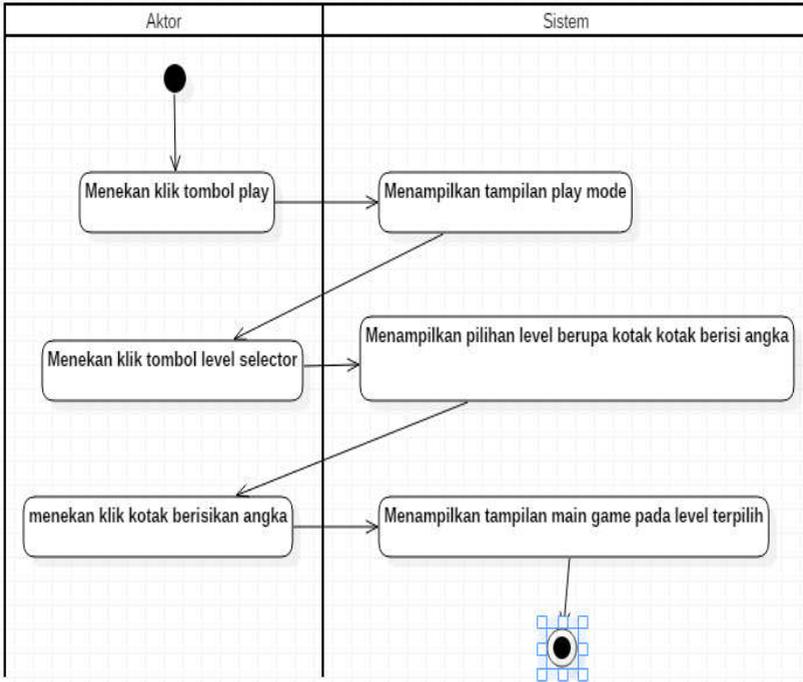
Tabel 3.8 Skenario Kasus Penggunaan Menandai level terselesaikan

Nama Kasus Penggunaan	Menandai level terselesaikan
Kode	UC-006
Deskripsi	Kasus penggunaan dimana sistem mengubah kotak pemilih level menjadi putih apabila level pernah diselesaikan.
Aktor	Sistem.
Kondisi Awal	Sistem menampilkan tulisan level complete pada suatu level.
Alur Normal	<ol style="list-style-type: none"> 1. Pemain memilih tombol 'BACK TO MENU' dan menekan klik. 2. Sistem menampilkan tampilan Main menu 3. Pemain memilih tombol 'PLAY' dan menekan klik 4. Sistem menampilkan tampilan play mode 5. Pemain memilih tombol 'Level Selector' dan menekan klik 6. Sistem menampilkan tampilan kotak kotak angka dengan mengganti kotak putih apabila level dengan urutan tersebut pernah diselesaikan

3.3.2.2 Diagram Aktivitas

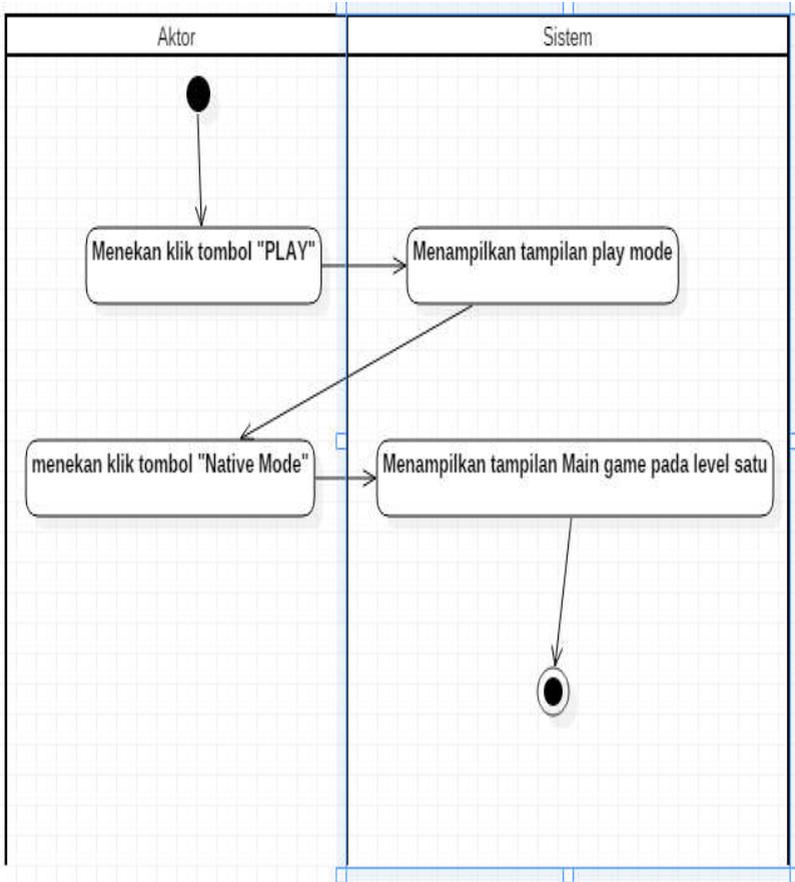
Diagram aktivitas menampilkan langkah-langkah normal yang harus dilakukan pemain untuk menjalankan studi kasus permainan dimulai dari awal permainan hingga kondisi akhir.

Diagram aktivitas untuk dari kasus penggunaan UC-001 yakni Memilih Level selector mode dijelaskan pada Gambar 3.2.



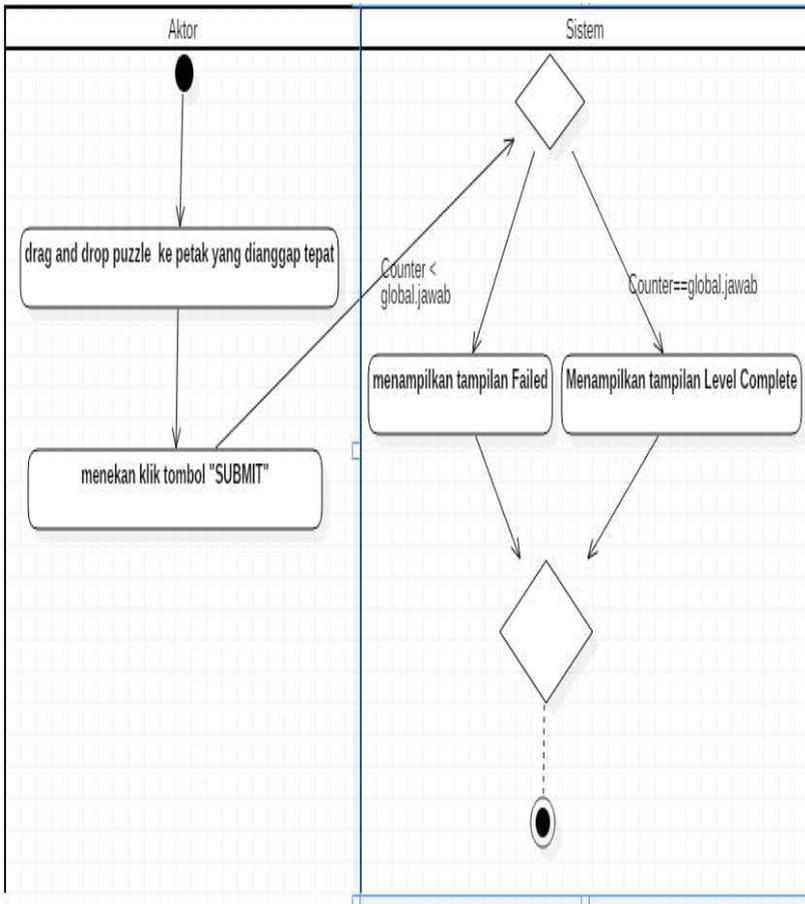
Gambar 3.2 Diagram aktivitas Memilih Level selector mode

Kemudian diagram aktivitas untuk dari kasus penggunaan UC-002 yakni Memilih Native Mode dijelaskan pada Gambar 3.3. Diagram aktivitas menyatakan bahwa aktivitas dimulai dari mengklik tombol “PLAY” yang dilakukan oleh pengguna dan diakhiri dengan menampilkan layar menu Main game pada level satu yang dilakukan oleh sistem.



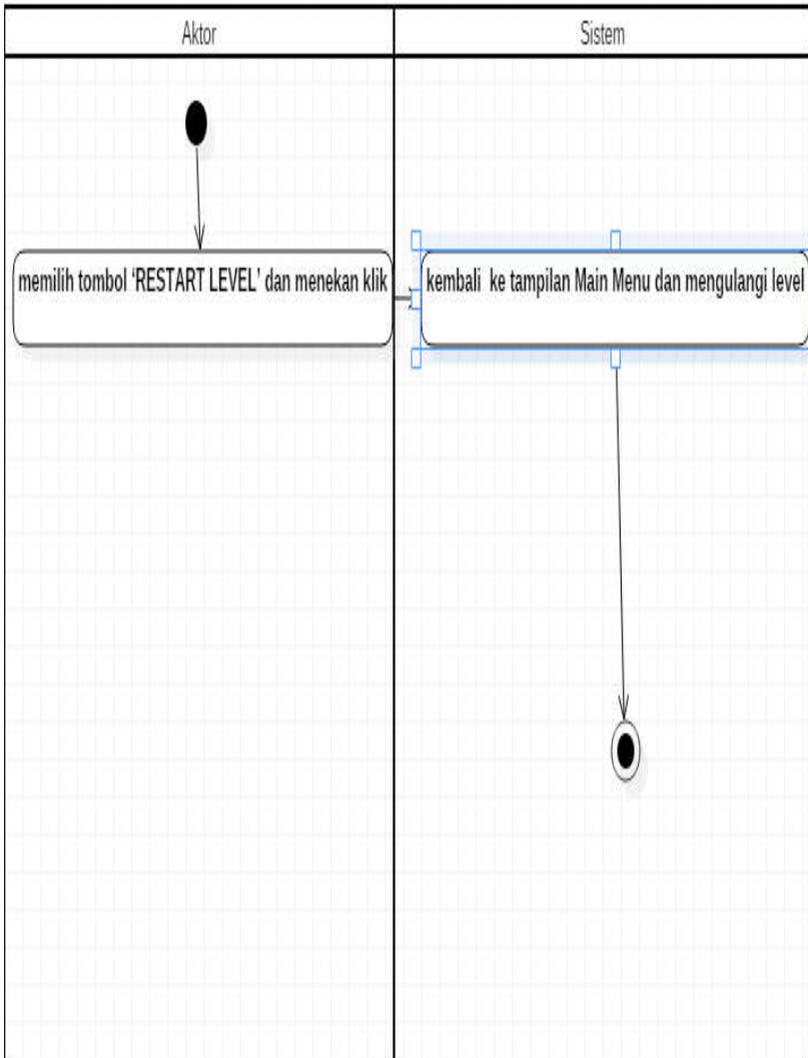
Gambar 3.3 Diagram aktivitas Memilih Native Mode

Kemudian diagram aktivitas untuk dari kasus penggunaan UC-003 yakni Menyelesaikan level dijelaskan pada Gambar 3.4. Diagram aktivitas menyatakan bahwa aktivitas dimulai dari melakukan drag and drop puzzle tersebut ke petak yang dianggap tepat sehingga membentuk kalimat matematika yang benar oleh pengguna dan diakhiri menampilkan tulisan Level Complete jika benar, danj tulisan Failed jika salah yang dilakukan oleh sistem



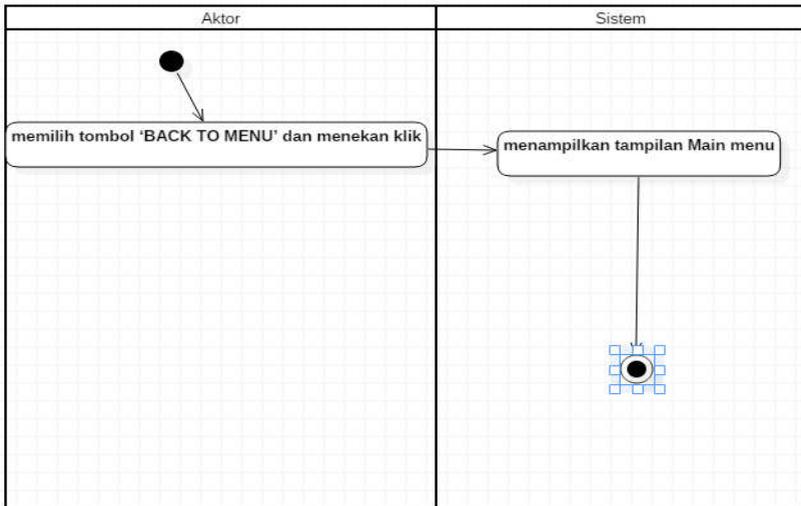
Gambar 3.4 Diagram aktivitas Menyelesaikan level

Kemudian diagram aktivitas untuk dari kasus penggunaan UC-004 yakni Mengulangi level dijelaskan pada Gambar 3.5. Diagram aktivitas menyatakan bahwa aktivitas dimulai dari memilih tombol 'RESTART LEVEL' dan menekan klik yang dilakukan oleh pengguna dan diakhiri dengan kembali ke tampilan Main Menu dan mengulangi level yang dilakukan oleh sistem



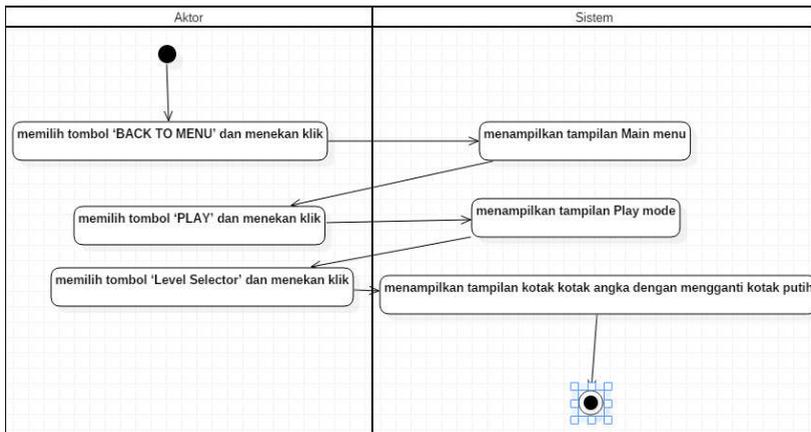
Gambar 3.5 Diagram aktivitas Mengulangi level

Selanjutnya diagram aktivitas untuk dari kasus penggunaan UC-005 yakni Kembali ke menu dijelaskan pada Gambar 3.6.



Gambar 3.6 Diagram aktivitas Kembali ke menu

Kemudian diagram aktivitas untuk dari kasus penggunaan UC-006 yakni Menandai level terselesaikan dijelaskan pada Gambar 3.7.



Gambar 3.7 Diagram aktivitas Menandai level terselesaikan

3.3.3 Perancangan Antarmuka Pengguna

Subbab ini membahas bagaimana rancangan antarmuka pengguna yang akan digunakan untuk tugas akhir. Rancangan antarmuka yang dibahas meliputi ketentuan masukan dan rancangan jendela tampilan. Dalam aplikasi ini terdapat beberapa tampilan, yaitu **Judul game**, **Main Menu**, **Menu Training**, **Menu Battle**, **Profile**, dan **Menu Buy**.

3.3.3.1 Tampilan Judul *Game*

Tampilan judul *game* merupakan tampilan yang pertama kali muncul ketika aplikasi dijalankan untuk yang pertama kalinya.. Tampilan ini akan berganti ke Main Menu sekitar 5 detik kemudian. Tampilan ini ditunjukkan pada Gambar 3.8.

3.3.3.2 Tampilan Main Menu

Tampilan Main Menu merupakan tampilan yang muncul setelah tampilan Judul *game*. Tampilan ini merupakan salah satu tampilan untuk interaksi pengguna yakni diminta untuk memilih salah satu tombol dimana ‘PLAY’ untuk bermain dan menuju main game, ‘INSTRUCTIONS’ untuk mengarah ke petunjuk cara bermain, dan ‘EXIT’ untuk menutup permainan. Tampilan ini akan berganti ke Main Menu sekitar 5 detik kemudian. Tampilan ini ditunjukkan pada Gambar 3.9.

3.3.3.3 Tampilan Play Mode

Tampilan ini merupakan tampilan untuk menuju ke menu-menu penting. Dari tampilan ini pengguna dapat memilih di antara dua tombol diantaranya adalah Native Mode dan Level Selector., Native mode untuk memulai dari level satu, sedangkan level selector untuk memilih level tertentu Tampilan ini ditunjukkan pada Gambar 3.10

3.3.3.4 Tampilan Main Game

Tampilan ini merupakan tampilan inti dari game. Puzzle berwarna aqua yang nantinya dipindah ke kotak berwarna putih polos sehingga membentuk kalimat matematika yang benar. Tombol “SUBMIT” digunakan untuk finalisasi dari letak puzzle yang telah kita letakkan sedemikian rupa sehingga dapat mengarah ke tampilan level complete atau failed, tombol “BACK TO MENU” untuk mengembalikan ke menu. Tampilan **Main Game** ditunjukkan seperti pada Gambar 3.11.

3.3.3.5 Tampilan Level Complete

Tampilan ini merupakan tampilan di mana pemain berhasil menyelesaikan suatu level, tombol “BACK TO MENU” digunakan untuk kembali ke Main menu, tombol “NEXT LEVEL” digunakan untuk berpindah ke Main game dengan level selanjutnya atau satu tingkat di atasnya. Tampilan **Level Complete** ditunjukkan seperti pada Gambar 3.12.

3.3.3.6 Tampilan Level Selector

Tampilan ini merupakan tampilan dari **Level Selector**. Pengguna dapat memilih kotak-kotak angka untuk memilih level dengan urutan angka tersebut, “BACK TO MENU” untuk kembali ke Menu Utama, tombol berbentuk segitiga kecil di kanan bawah digunakan untuk beralih ke tampilan Level Selector Lainnya. Tampilan **Level Selector** ditunjukkan seperti pada Gambar 3.13.

3.3.3.7 Tampilan Failed

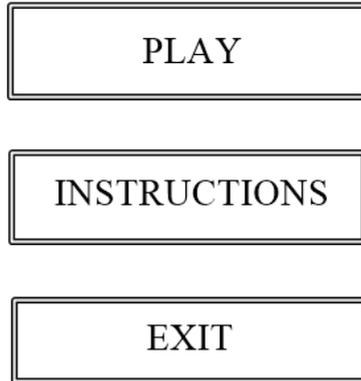
Tampilan ini merupakan tampilan di mana pemain dinyatakan kalah, hal ini dikarenakan menyusun kalimat matematika yang tidak sesuai, tombol “BACK TO MENU” digunakan untuk kembali ke Menu utama, tombol “RESTART LEVEL” digunakan untuk kembali ke tampilan Main game dengan level yang sama. Tampilan **Failed** ditunjukkan seperti pada Gambar 3.14

3.3.3.8 Tampilan Out of Time

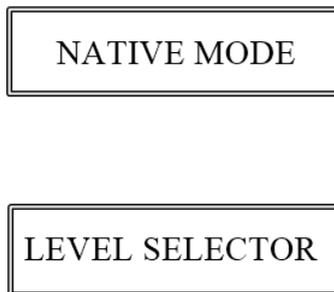
Tampilan ini merupakan tampilan di mana pemain dinyatakan kalah, hal ini dikarenakan pemain kehabisan waktu saat menyelesaikan level, tombol “BACK TO MENU” digunakan untuk kembali ke Menu utama, tombol “RESTART LEVEL” digunakan untuk kembali ketampilan Main main game dengan level yang sama. Tampilan **Out of time** ditunjukkan seperti pada Gambar 3.15.

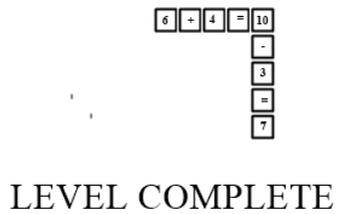
MATH
SCRABBLE

Gambar 3.8 Tampilan Judul *game*

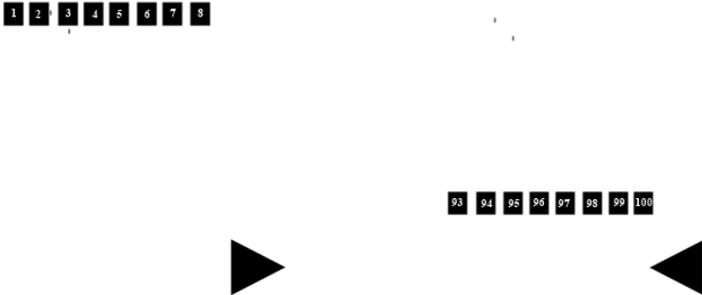


Gambar 3.9 Tampilan Menu Permainan

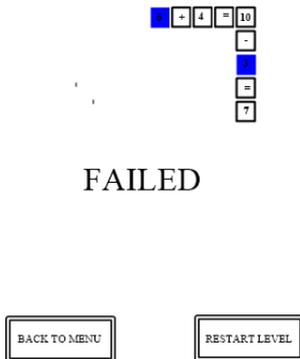


Gambar 3.10 Tampilan Play Mode**Gambar 3.11 Tampilan Main game**

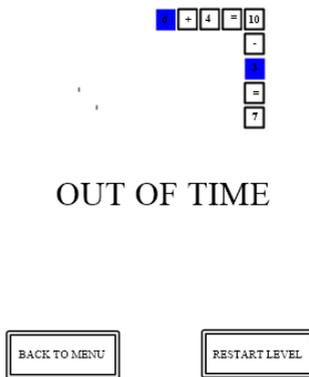
Gambar 3.12 Tampilan Level Complete



Gambar 3.13 Tampilan Level Selector(1) dan Level Selector (2)



Gambar 3.14 Tampilan Failed



Gambar 3.15 Tampilan Out of time

3.3.4 Perancangan Kontrol Permainan

Terdapat satu jenis kontrol dalam permainan yaitu menggunakan *mouse* dengan klik kiri apabila dijalankan pada desktop dan tap apabila dijalankan pada *smartphone* Android. Semua interaksi dari user hanya menggunakan satu kontrol tersebut.

3.3.5 Perancangan Aturan Permainan

Alur permainan merupakan serangkaian proses yang harus diikuti pemain untuk memperoleh kemenangan. Untuk memperoleh kemenangan pemain harus memindahkan *puzzle puzzle* ke dalam kotak putih polos sehingga membentuk kalimat matematika yang benar, selain itu pemain juga harus memperhatikan waktu yang tersisa sehingga tidak terjadi *out of time* jika semua *puzzle* dianggap

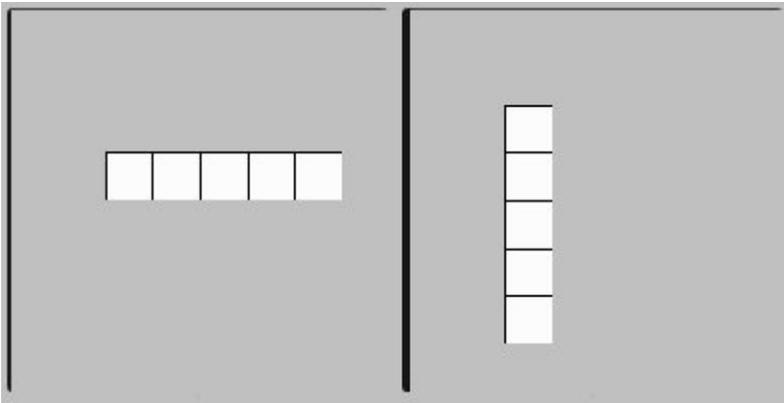
sudah diletakkan di kotak putih yang tepat maka pemain menekan tombol “SUBMIT” untuk dinilai kebenaran dan ketepatannya.

3.3.6 Perancangan Data

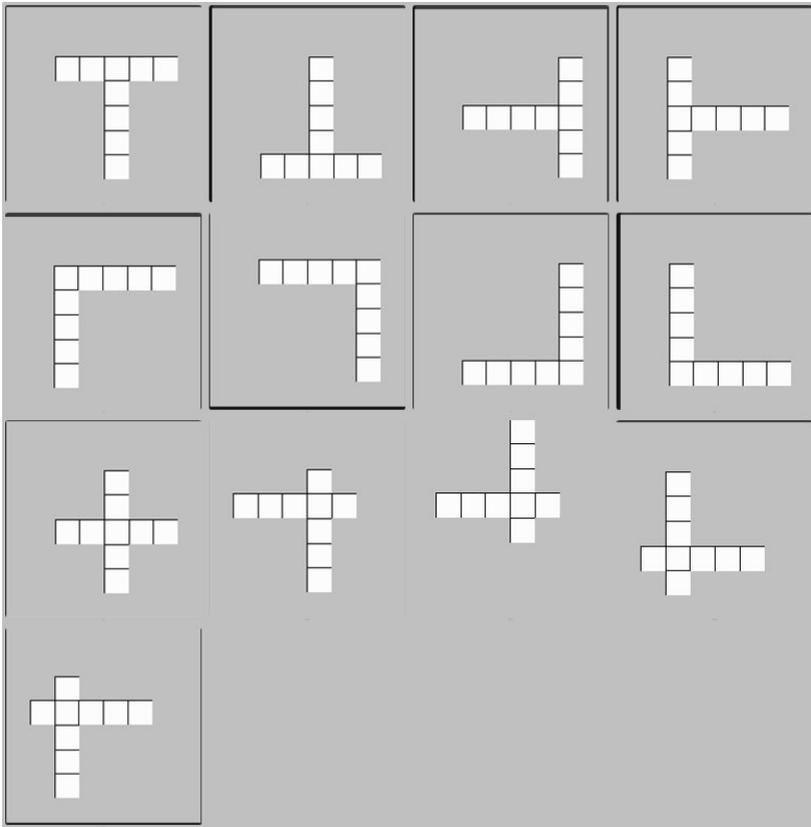
Perancangan data merupakan hal penting untuk diperhatikan karena diperlukan data yang tepat agar sistem beroperasi secara benar. Perancangan data ini dibagi menjadi dua yakni data untuk materi edukasi dan data untuk permainan.

3.3.6.1 Perancangan Data Materi Edukasi

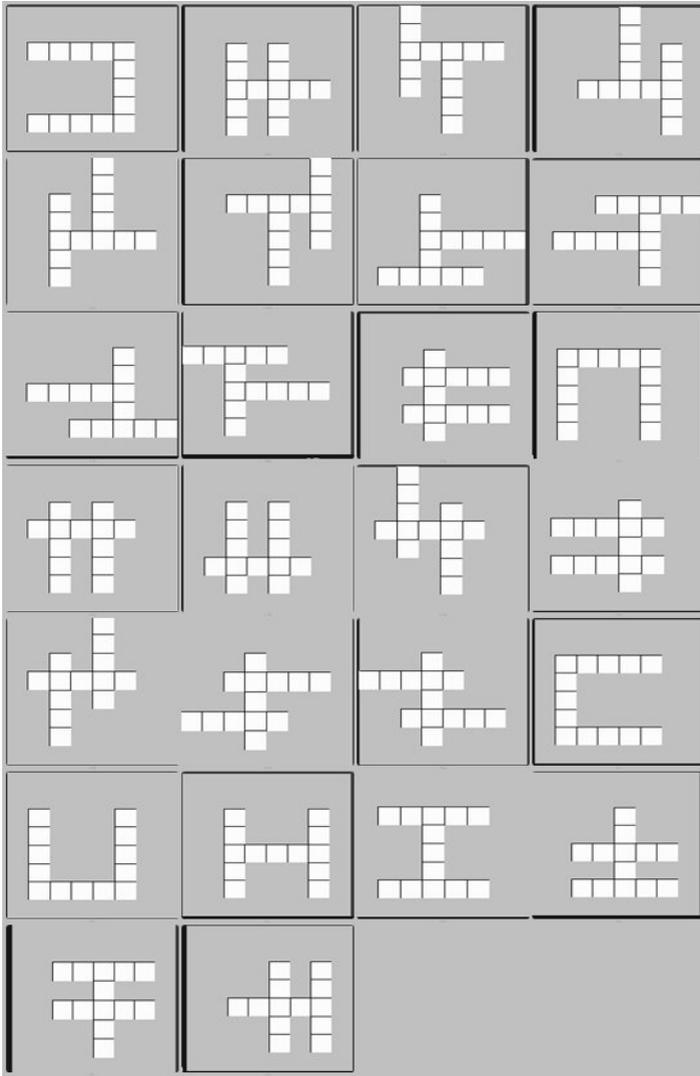
Perancangan Materi edukasi diawali dengan mendesain *pattern* sebanyak mungkin, berikut adalah gambar untuk masing masing kumpulan *pattern* sesuai dengan jumlah soal yang dijelaskan *pattern* diinisialisai lalu diacak sesuai urutannya mulai gambar 3.16 sampai dengan 3.22



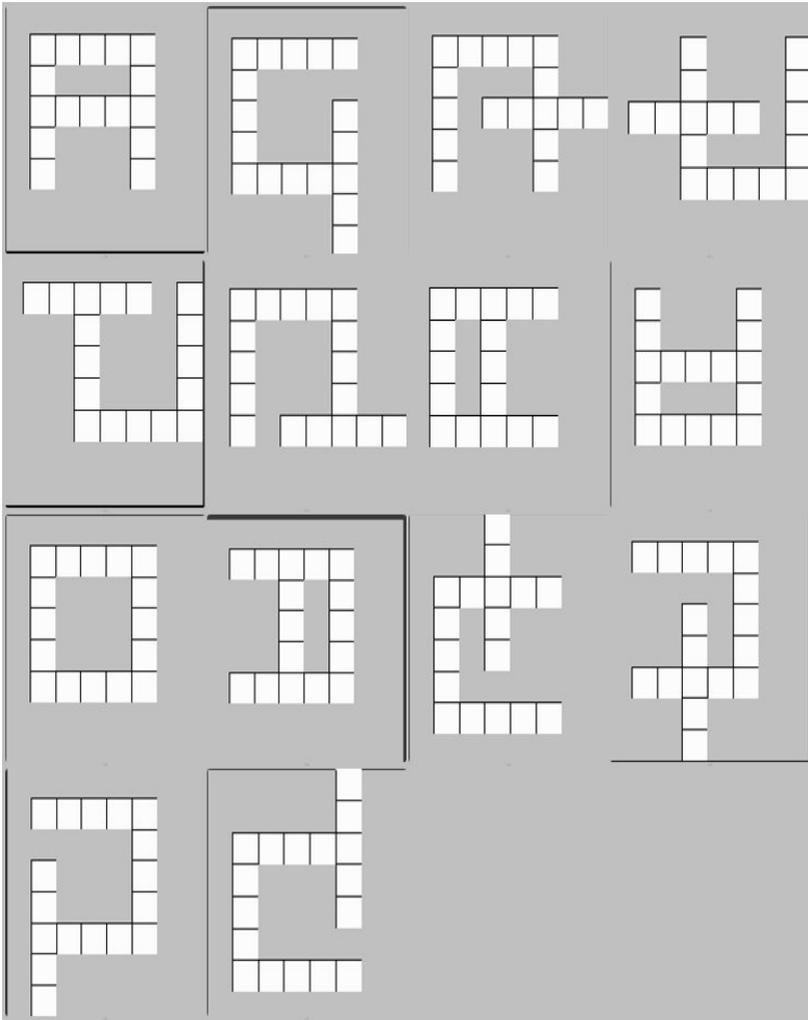
Gambar 3.16 *Pattern* untuk satu soal, digunakan pada level satu sampai lima, terdiri dari dua macam *pattern*



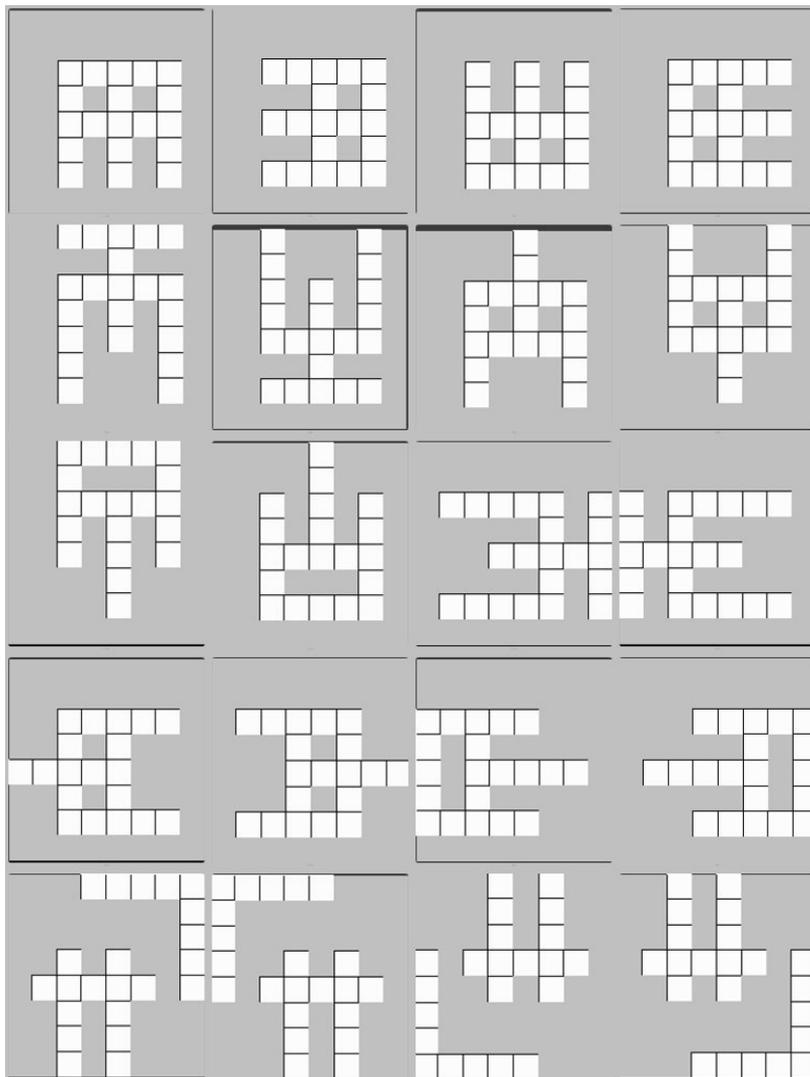
Gambar 3.17 *Pattern* untuk dua soal, digunakan pada level enam sampai 15, terdiri dari 13 macam *pattern*



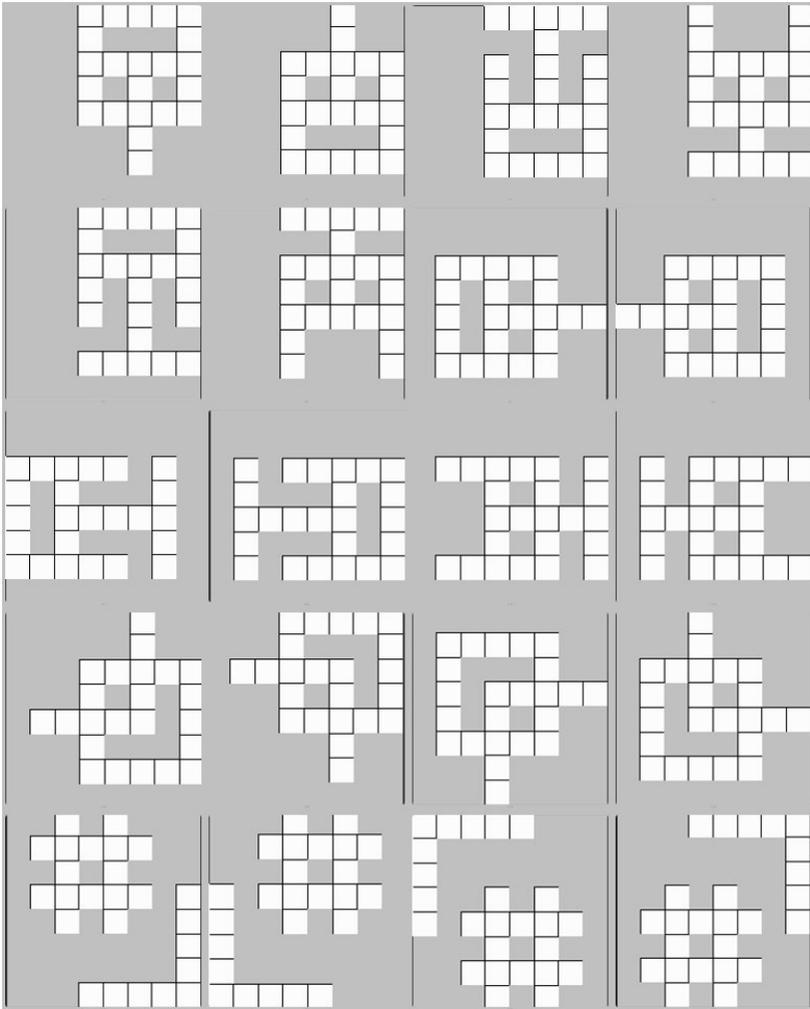
Gambar 3.18 *Pattern* untuk tiga soal, digunakan pada level 16 sampai 28, terdiri dari 26 macam *pattern*



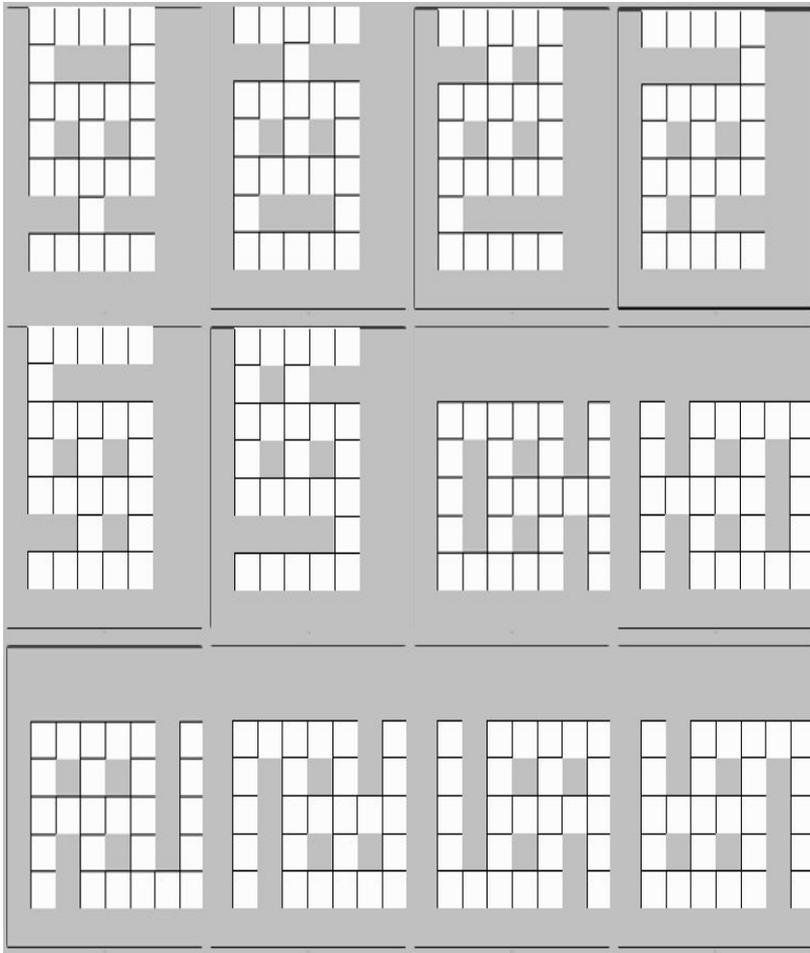
Gambar 3.19 *Pattern* untuk empat soal, digunakan pada level 29 sampai 40, terdiri dari 14 macam *pattern*



Gambar 3.20 *Pattern* untuk lima soal, digunakan pada level 41 sampai 60, terdiri dari 20 macam *pattern*



Gambar 3.21 *Pattern* untuk enam soal, digunakan pada level 61 sampai 85, terdiri dari 20 macam *pattern*



Gambar 3.22 *Pattern* untuk tujuh soal, digunakan pada level 86 sampai 100, terdiri dari 12 macam *pattern*

Data untuk perancangan materi edukasi didapat dari sumber yang terkait untuk materi pelajaran pada umumnya. Yakni berisi soal penjumlahan (+), pengurangan (-), perkalian (*) dan pembagian (/)

Tabel 3.9 Detil Perancangan Data untuk materi edukasi tiap level untuk selengkapnya bisa dilihat pada Lampiran 3.

Level	Angka Max	Macam Soal	Jumlah Soal
1	20	+, -	1
10	20	+, -	2
20	20	+, -, *	3
30	40	+, -, *, /	4
50	60	+, -, *, /	5
61	80	+, -, *, /	6
95	99	+, -, *, /	7

3.3.6.2 Perancangan Data Permainan dan Skenario tingkat kesulitan

Data untuk perancangan permainan terdiri dari Rata rata total petal per *pattern*, Jumlah petak kosong tiap levelnya , dan waktu untuk menyelesaikan level tersebut, yang mana akan dijelaskan di tabel 3.10, untuk selengkapnya bisa dilihat pada lampiran 3

Tabel 3.10 Detil Perancangan Data untuk permainan tiap level.

Level	Rata rata total petak per pattern	Jumlah Petak Kosong	Waktu
1	5	1	348
10	9	3	330

20	12	5	310
35	16	10	280
50	20	12	250
61	24	8	228
95	26	16	160

3.3.6.1 Perancangan Inisialisasi *Piece Puzzle* dan evaluasi input pemain

Dimisalkan ada sejumlah n petak putih kosong maka agar terhindar dari dualisme kalimat matematika maka untuk satu sampai setengah dari n adalah jatah *slot* untuk angka dan sisanya adalah jatah *slot* untuk sama dengan dan operator, sehingga selisih banyaknya angka dan bukan angka dalam membuat *piece puzzle* tidak berselisih jauh.

Untuk evaluasi *input* dari pemain dilakukan dengan cara mendeteksi apakah di posisi petak kosong tertentu terdapat *piece puzzle* yang sesuai dengan bobot nilainya, jika semua iya (ada) maka dianggap benar (Level Complete), jika ada stu saja yang tidak maka dianggap salah (Failed).

3.3.6.2 Perancangan Pembangkita Kalimat Matematika

1. Pembangkitan Kalimat Matematika perkalian

2	x	8	=	16
---	---	---	---	----

Gambar 3.23 Contoh Kalimat matematika perkalian

Pada gambar di atas terdapat lima kotak yakni kotak pertama berisi *operand* bernilai dua pada petak ke dua terdapat operator kali dan di kotak ke tiga terdapat *operand* dengan nilai delapan lalu di kotak keempat terdapat konjungsi yakni sama dengan, dan hasil terletak di petak ke lima.

Untuk pembangkitan kalimat matematika perkalian mula mula nilai terbesar yang diacak terlebih dahulu. karena di sini hasil menjadi nilai terbesar maka hasil diacak dengan *range* tiga per empat dari angka maksimal(angka yang terbesar yang mungkin keluar dalam suatu level) hingga angka maksimal, lalu yang diacak setelah hasil adalah satu dari dua angka sisa paling kiri untuk kalimat matematika mendatar dan paling atas untuk kalimat matematika menurun, sehingga yang diacak berikutnya adalah *operand* pertama yang diacak dengan *range* dari angka satu hingga setengah kali angka maksimal yang mana merupakan faktor dari angka terbesar, dan untuk *operand* ke dua adalah hasil pembagian dari angka terbesar (hasil) dan *operand* pertama.

2. Pembangkitan Kalimat Matematika pembagian

20	/	4	=	5
----	---	---	---	---

Gambar 3.24 Contoh Kalimat matematika pembagian

Pada gambar di atas terdapat lima kotak yakni kotak pertama berisi *operand* bernilai dua puluh pada petak ke dua terdapat operator bagi dan di kotak ke tiga terdapat *operand* dengan nilai empat lalu di kotak keempat terdapat konjungsi yakni sama dengan, dan hasil terletak di petak ke lima.

Untuk pembangkitan kalimat matematika pembagian mula mula nilai terbesar yang diacak terlebih dahulu. karena di sini *operand* pertama menjadi nilai terbesar maka *operand* pertama diacak dengan *range* tiga per empat dari angka maksimal(angka yang terbesar yang mungkin keluar dalam suatu level) hingga angka maksimal, lalu yang diacak setelah hasil adalah satu dari dua angka sisa paling kiri untuk kalimat matematika mendatar dan paling atas untuk kalimat matematika menurun, sehingga yang diacak berikutnya adalah *operand* kedua yang diacak dengan *range* dari angka satu hingga setengah kali angka maksimal yang mana merupakan faktor dari angka terbesar, dan untuk hasil adalah hasil pembagian dari angka terbesar (*operand* pertama) dan *operand* kedua.

3. Pembangkitan Kalimat Matematika penjumlahan

5	+	10	=	15
---	---	----	---	----

Gambar 3.25 Contoh Kalimat matematika penjumlahan

Pada gambar di atas terdapat lima kotak yakni kotak pertama berisi *operand* bernilai lima pada petak ke dua terdapat operator tambah dan di kotak ke tiga terdapat *operand* dengan nilai sepuluh lalu di kotak keempat terdapat konjungsi yakni sama dengan, dan hasil terletak di petak ke lima.

Untuk pembangkitan kalimat matematika penjumlahan mula mula nilai terbesar yang diacak terlebih dahulu. karena di sini hasil menjadi nilai terbesar maka hasil diacak dengan *range* tiga per empat dari angka maksimal(angka yang terbesar yang mungkin

keluar dalam suatu level) hingga angka maksimal, lalu yang diacak setelah hasil adalah satu dari dua angka sisa paling kiri untuk kalimat matematika mendatar dan paling atas untuk kalimat matematika menurun, sehingga yang diacak berikutnya adalah *operand* pertama yang diacak dengan *range* dari angka satu hingga setengah kali angka maksimal yang mana merupakan nilai lebih kecil dari angka terbesar, dan untuk *operand* kedua adalah hasil pengurangan dari angka terbesar (*operand* pertama) dan *operand* pertama.

15	=	10	+	5
----	---	----	---	---

Gambar 3.26 Contoh Kalimat matematika penjumlahan terbalik

Pada gambar di atas terdapat lima kotak yakni kotak pertama berisi hasil bernilai lima belas pada petak ke dua terdapat konjungsi sama dengan dan di kotak ke tiga terdapat *operand* dengan nilai sepuluh lalu di kotak keempat terdapat operator tambah dan *operand* kedua terletak di petak ke lima.

Untuk pembangkitan kalimat matematika penjumlahan terbalik mula mula nilai terbesarliah yang diacak terlebih dahulu. karena di sini hasil menjadi nilai terbesar maka hasil diacak dengan *range* tiga per empat dari angka maksimal(angka yang terbesar yang mungkin keluar dalam suatu level) hingga angka maksimal, lalu yang diacak setelah hasil adalah satu dari dua angka sisa paling kiri untuk kalimat matematika mendatar dan paling atas untuk kalimat matematika menurun, sehingga yang diacak berikutnya adalah *operand* pertama yang diacak dengan *range* dari angka satu hingga

setengah kali angka maksimal yang mana merupakan nilai lebih kecil dari angka terbesar, dan untuk *operand* kedua adalah hasil pengurangan dari angka terbesar (*operand* pertama) dan *operand* pertama.

4. Pembangkitan Kalimat Matematika pengurangan

19	-	11	=	8
----	---	----	---	---

Gambar 3.27 Contoh Kalimat matematika pengurangan

Pada gambar di atas terdapat lima kotak yakni kotak pertama berisi *operand* bernilai sembilan belas pada petak ke dua terdapat operator kurang dan di kotak ke tiga terdapat *operand* dengan nilai sebelas lalu di kotak keempat terdapat konjungsi yakni sama dengan, dan hasil terletak di petak ke lima.

Untuk pembangkitan kalimat matematika pengurangan mula mula nilai terbesar lah yang diacak terlebih dahulu. karena di sini *operand* pertama menjadi nilai terbesar maka *operand* pertama diacak dengan *range* tiga per empat dari angka maksimal(angka yang terbesar yang mungkin keluar dalam suatu level) hingga angka maksimal, lalu yang diacak setelah hasil adalah satu dari dua angka

sisanya paling kiri untuk kalimat matematika mendatar dan paling atas untuk kalimat matematika menurun, sehingga yang diacak berikutnya adalah *operand* kedua yang diacak dengan *range* dari angka satu hingga setengah kali angka maksimal yang mana merupakan nilai lebih kecil dari angka terbesar, dan untuk hasil adalah hasil pengurangan dari angka terbesar (*operand* pertama) dan *operand* kedua.

11	=	19	-	8
----	---	----	---	---

Gambar 3.28 Contoh Kalimat matematika pengurangan terbalik

Pada gambar di atas terdapat lima kotak yakni kotak pertama berisi hasil bernilai sebelas pada petak ke dua terdapat konjungsi sama dengan dan di kotak ke tiga terdapat *operand* dengan nilai sembilan belas lalu di kotak keempat terdapat operator kurang dan *operand* kedua terletak di petak ke lima.

Untuk pembangkitan kalimat matematika pengurangan terbalik mula mula nilai terbesar yang diacak terlebih dahulu. karena di sini *operand* pertama menjadi nilai terbesar maka *operand* pertama diacak dengan *range* tiga per empat dari angka maksimal(angka yang terbesar yang mungkin keluar dalam suatu level) hingga angka maksimal, lalu yang diacak setelah hasil adalah satu dari dua angka sisanya paling kiri untuk kalimat matematika mendatar dan paling atas untuk kalimat matematika menurun, sehingga yang diacak berikutnya adalah hasil yang diacak dengan

range dari angka satu hingga setengah kali angka maksimal yang mana merupakan nilai lebih kecil dari angka terbesar, dan untuk *operand* kedua adalah hasil pengurangan dari angka terbesar (*operand* pertama) dan hasil.

BAB IV IMPLEMENTASI

Pada bab ini akan dibahas mengenai implementasi dari perancangan perangkat lunak. Di dalamnya mencakup proses penerapan dan pengimplementasian algoritma, dan antar muka yang mengacu pada rancangan yang telah dibahas sebelumnya.

4.1 Lingkungan Implementasi

Lingkungan implementasi dari tugas akhir dijelaskan pada Tabel 4.1.

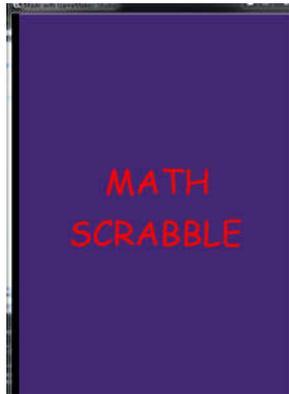
Tabel 4.1 Lingkungan Implementasi Perangkat Lunak

Perangkat Keras	Prosesor: Pentium(R) Dual-Core CPU T4400 @ 2.20GHz Memori: 4 GB
Perangkat Lunak	Sistem Operasi: Microsoft Windows 8 64-bit Perangkat Pengembang: Adobe Profesional Flash CC

4.2 Implementasi Permainan

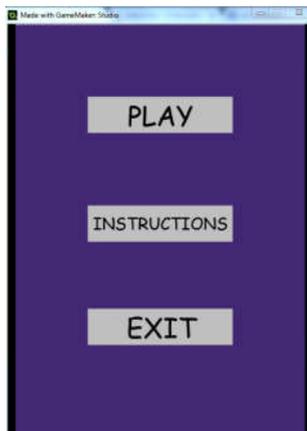
Implementasi terdapat dua macam yakni implementasi kasus penggunaan dan antarmuka, untuk antarmuka langsung dapat dilakukan *screenshooting*, sedangkan untuk kasus penggunaan hanya diambil beberapa source code yang dianggap mewakili dari keseluruhan yang mana terdapat pada **Lampiran 1**

4.2.1 Implementasi Tampilan Judul Game



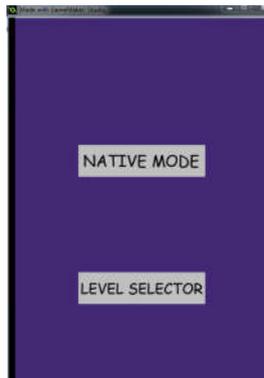
Gambar 4.1 Tampilan Judul Game

4.2.2 Implementasi Tampilan Main Menu



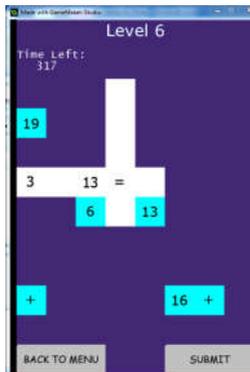
Gambar 4.2 Tampilan Main Menu

4.2.3 Implementasi Tampilan Play Mode



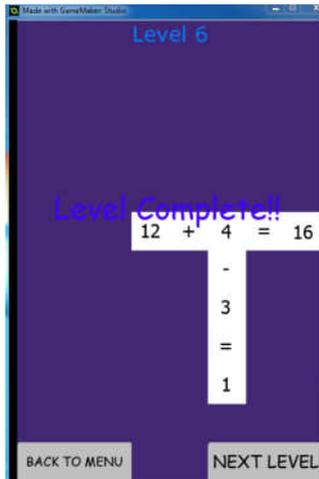
Gambar 4.3 Tampilan Play Mode

4.2.4 Implementasi Tampilan Main Game



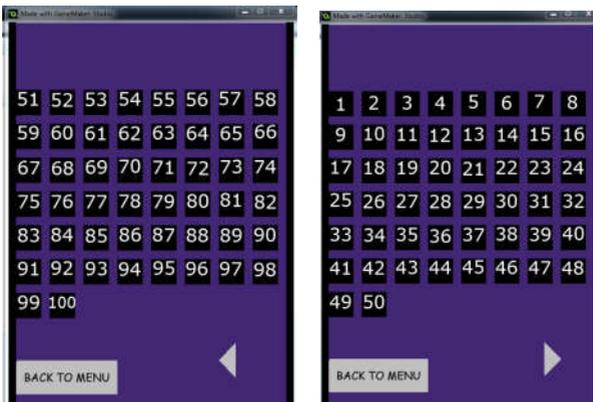
Gambar 4.4 Tampilan Play Mode

4.2.5 Implementasi Tampilan Level Complete



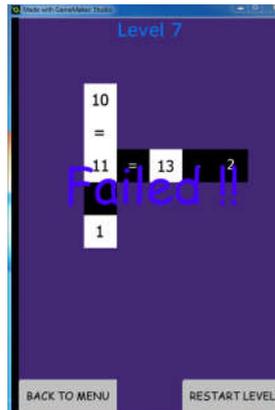
Gambar 4.5 Tampilan Level Complete

4.2.6 Implementasi Tampilan Level Selector



Gambar 4.6 Tampilan Level Selector 1 dan 2

4.2.7 Implementasi Tampilan Failed



Gambar 4.7 Tampilan Failed

4.2.8 Implementasi Tampilan Out Of Time



Gambar 4.8 Tampilan Out of time

4.2.9 Implementasi Kasus Penggunaan

Dari enam kasus penggunaan implementasi disajikan dalam *Source code*, yang mana selengkapanya terdapat pada lampiran 1 **Lampiran 1**

1. Implementasi Kasus Penggunaan Level Selector

```
instance_create(var_mod*64, (var_bagi*64)+64, global.obj_
link2_level[i])
}
else{

instance_create(var_mod*64, (var_bagi*64)+64, global.obj_
link_level[i])
}

}
//Source code pada tombol angka (misal tombol angka
95), tombol angka akan berisi nama level yg di load
yakni Level_X

room_goto(Level_95)
```

2. Implementasi Kasus Penggunaan Memilih Native Mode

```
//Source code pada tombol PLAY

room_goto(room107);
//Source code pada tombol NATIVE MODE

room_goto(Level_1)

//Source code pada main game (Level 1)

global.level=1;
global.waktu = 348;
global.angka_max =20;
//petak kosong =1
global.jumlah_soal =1;
i=0;
```

3. Implementasi Kasus Penggunaan Menyelesaikan Level

```

/*--MENYELESAIKAN LEVEL--*/
/*action piece puzzle*/

//Create
for(i=0;i<=104;i++)
{
    object_set_depth(global.obj_piece[i], -100);
}
//Destroy

instance_create(global.cursor_x,global.cursor_y,global.obj_moving[global.cursor_value])
//step
if(mouse_check_button_pressed(mb_left)){
    for(i=1;i<=global.jwb;i++)
    {
        for(j=0;j<=104;j++)
        {
            if(mouse_x==global.pos_Jawab_X_fix[i]&&
mouse_y==global.pos_Jawab_Y_fix[i]                               &&
position_meeting(mouse_x,mouse_y,global.obj_piece[j]))
            {
                global.cursor_x=global.pos_Jawab_X_fix[i]
                global.cursor_y=global.pos_Jawab_Y_fix[i]
                global.cursor_value=j;
                var
inst=instance_position(mouse_x,mouse_y,global.obj_piece[j])
                if inst{
                    with obj_piece_
                    {
                        instance_destroy()
                    }
                }
            }
        }
    }
}
//Mouse pressed
if(global.counter_click==1){
    for(i=0;i<=104;i++)
    {

```

```

    obj=  global.obj_moving[i];
        var  inst  =  instance_position(global.cursor_x,
global.cursor_y, obj);

//instance_change(global.obj_moving[global.cursor_value],
true);
        if inst {
            with inst instance_destroy();
        }
    }
}

for(i=0;i<=104;i++)
{
    if(position_meeting(mouse_x, mouse_y, global.obj_piece[i]))
    {
        with obj_moving_parent
        { instance_destroy();
        }
        xx=mouse_x%64
        yy=mouse_y%64
        global.cursor_x=mouse_x-xx;
        global.cursor_y=mouse_y-yy;
        global.counter_click=1;
        global.cursor_value=i;

//instance_create(global.obj_moving[global.cursor_value],global
.cursor_x,global.cursor_y)
        obj=  global.obj_piece[i];
        var  inst  =  instance_position(global.cursor_x,
global.cursor_y, obj);

//instance_change(global.obj_moving[global.cursor_value],
true);

        if inst {
            with inst instance_destroy();
        }

    }
}

```

4. Implementasi Kasus Penggunann Mengulangi Level

```

/*--MENYELESAIKAN LEVEL--*/
/*action tombol RESTART LEVEL */
room_restart()

```

5. Implementasi Kasus Penggunann Kembali Ke Menu

```
/*--KEMBALI KE MENU--*/
//Sourcode pada tombol BACK TO MENU
room_goto(menu_awal);
```

6. Implementasi Kasus Penggunaan Menandai Level Terselesaikan

```
/*--MENANDAI LEVEL TERSELESAIKAN--*/
//Action pada level complete
for(i=1;i<=100;i++){
    if(room==global.room_level[i])
    {
        global.berhasil[i]=1
    }
}
//Source code pada room level selector 1
a=1
for(i=a;i<=50;i++)
{
    var_mod=(i-a)%8;
    var_bagi=floor((i-a+8)/8)
    if(global.berhasil[i]==1)
    {

instance_create(var_mod*64,(var_bagi*64)+64,global.obj_
link2_level[i])
    }
    else{

instance_create(var_mod*64,(var_bagi*64)+64,global.obj_
link_level[i])
    }

}
//Source code pada room level selector 2
a=51
for(i=a;i<=100;i++)
{
    var_mod=(i-a)%8;
    var_bagi=floor((i-a+8)/8)
    if(global.berhasil[i]==1)
    {

instance_create(var_mod*64,(var_bagi*64)+64,global.obj_
link2_level[i])
    }
    else{
```

```
instance_create(var_mod*64, (var_bagi*64)+64, global.obj_  
link_level[i])  
    }  
}
```

BAB V

PENGUJIAN DAN EVALUASI

Pada bab ini akan dijelaskan mengenai rangkaian uji coba dan evaluasi yang dilakukan. Proses pengujian dilakukan menggunakan metode kotak hitam berdasarkan skenario yang telah ditentukan dan pengujian dilakukan dengan survei langsung kepada pengguna.

5.1 Lingkungan Uji Coba

Lingkungan pelaksanaan uji coba meliputi perangkat keras dan perangkat lunak yang akan digunakan pada sistem ini. Spesifikasi perangkat keras dan perangkat lunak yang digunakan dalam rangka uji coba perangkat lunak ini dicantumkan pada Tabel 5.1 dan Tabel 5.2.

Tabel 5.1 Lingkungan Ujicoba Perangkat Lunak (bagian 1).

Perangkat Keras	Prosesor: Intel(R) Core(TM) i3-5005U CPU @ 2.00GHz (4 CPUs), ~2.0GHz Memori: 4096MB RAM
Perangkat Lunak	Sistem Operasi: Windows 10 Professional 64-bit Perangkat Pengembang: Game Maker

Tabel 5.2 Lingkungan Ujicoba Perangkat Lunak (bagian 2).

Perangkat Keras	Alat : <i>Smarthphone</i> Memori: 512 MB
Perangkat Lunak	Sistem Operasi: Android Perangkat Pengembang: Game Maker

5.2 Pengujian Performa Pembangkitan *puzzle*.

Pengujian dilakukan untuk menguji seberapa variatif sistem membangkitkan *puzzle* secara otomatis yang hanya berkaitan dengan edukasi. Sistem yang akan diuji adalah pada performa **Main Game**, karena di situlah *puzzle* dibangkitkan.

5.2.1 Skenario Uji Pembangkitan *Puzzle*.

Skenario Pengujian kecepatan performa digunakan untuk memberikan tahap-tahap dalam pengujian sistem dalam hal mendapatkan keberagaman *puzzle*. Skenario ini tertera pada Tabel 5.3.

Tabel 5.3 Skenario Pengujian Performa

Deskripsi	Bertujuan untuk mengetahui performa sistem dalam membangkitkan area permainan
Langkah pengujian	<ol style="list-style-type: none"> 1. pilih salah satu level melalui level selector mode. 2. Tekan tombol ‘SUBMIT’ lalu tekan tombol ‘RESTART LEVEL’ . 3. Catat bentuk dan soal yang telah dibangkitkan 4. Ulangi langkah ini sebanyak banyaknya, (di sini ditetapkan 20 kali). 5. Hitung berapa kali terjadi pengulangan bentuk.

Pengujian dilakukan dengan cara memilih satu persatu level yang ingin dibangkitkan pada layar **Main Game**. Lalu langsung ditekan tombol ‘SUBMIT’ , dan selanjutnya ditekan tombol ‘RESTART LEVEL’ dan dicatat pattern , soal, dan *puzzle* yang dibangkitkan, ini dilakukan sampai 20 kali sehingga memungkinkan ada 20 bentuk untuk level yang berbeda. Setelah itu dicatat berapa kali bentuk,soal, dan letak *piece puzzle* yang sama akan terulang.

5.2.2 Hasil Pengujian Performa

Hasil pengujian performa dikhususkan untuk menghitung terjadinya *pattern*, soal, letak *piece puzzle*, dan letak petak kosong yang sama benar benar terulang pada 20 kali percobaan pengulangan level. pada Tabel 5.4. akan dihitung frekuensinya bila terjadi.

No	Level	Frekuensi	Jumlah Soal
1	1	Tidak pernah	1 soal
2	10	Tidak Pernah	2 soal
3	20	Tidak Pernah	3 soal
4	35	Tidak Pernah	4 soal
5	50	Tidak Pernah	5 soal
6	61	Tidak Pernah	6 soal
7	95	Tidak Pernah	7 soal

Tabel 5.4 Hasil Uji Coba Performa Pembangkitan *Puzzle*

Berdasarkan Tabel 5.4 pemain nantinya akan memiliki kemungkinan kecil untuk menemui kombinasi *pattern*, soal, dan letak *puzzle* yang sama dengan kombinasi yang pernah dicoba sebelumnya. Dengan ini pengujian pembangkitan *puzzle* secara otomatis dinyatakan **berhasil**, di mana screenshot pembangkitan *puzzle* dapat diamati dalam **Lampiran 2**. Hasil pembangkitan *puzzle* merupakan kombinasi antara pengacakan *pattern*, pengacaan *piece puzzle*, serta pengacakan soal

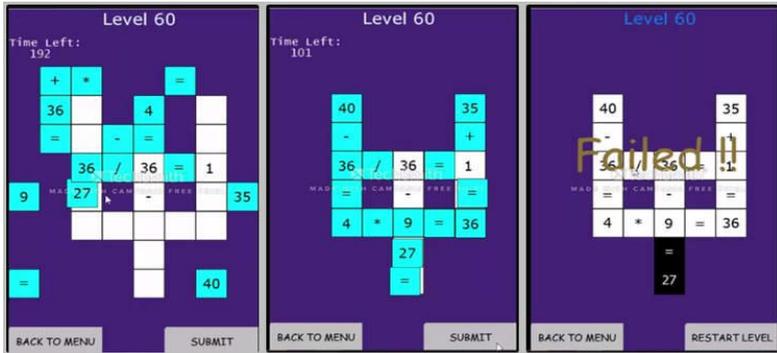
5.3 Pengujian Aturan Main Math Scrabble

5.3.1 Skenario Uji Aturan Main Math Scrabble.

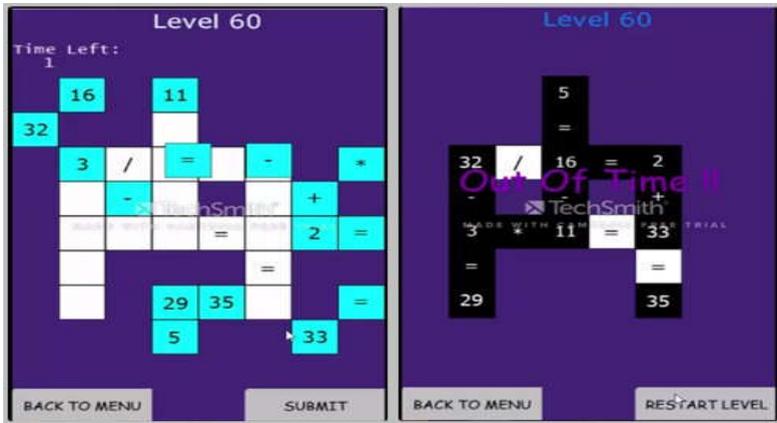
Tabel 5.5 Skenario Pengujian Aturan Main

Deskripsi	Bertujuan untuk mengetahui performa sistem dalam menerapkan aturan permainan
Langkah pengujian	<ol style="list-style-type: none"> 1. Pemain memainkan level tertentu, dengan memindahkan piece puzzle ke sembarang petak kosong 2. Pemain menmekan tombol 'SUBMIT' 3. Sistem memunculkan tampilan Failed dengan solusi jawaban tertampilkan 4. Pemain menekan tombol 'RESTART LEVEL'. 5. Pemain memainkan LEVEL tertentu namun dikondisikan tidak selesai sampai waktu habis 6. Sistem memunculkan tampilan Out Of Time dengan solusi jawaban tertampilkan 7. Pemain menekan tombol 'RESTART LEVEL' 8. Pemain memainkan level tertentu, dengan memindahkan piece puzzle ke petak kosong ke petak kosong yang tepat sebelum waktu habis 9. Pemain menmekan tombol 'SUBMIT' 10. Sistem memunculkan tampilan Level Complete

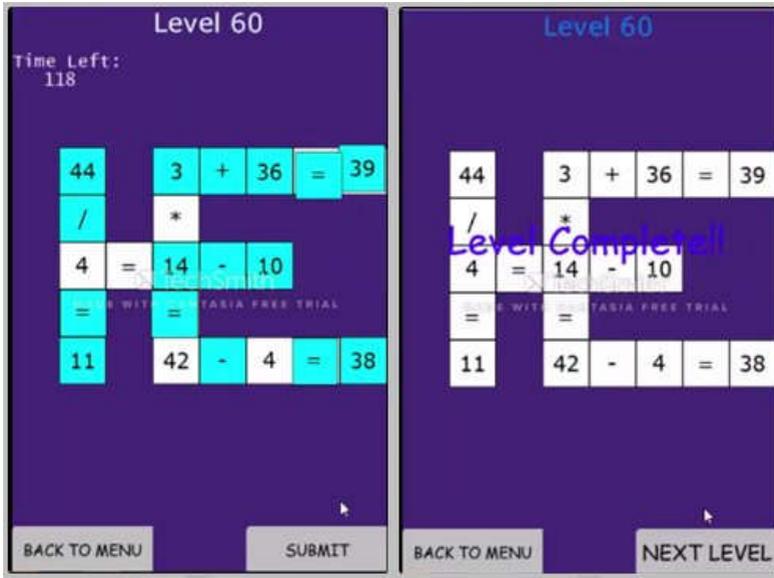
5.3.2 Hasil Pengujian Aturan Main Math Scrabble Pembangkitan Puzzle.



Gambar 5.1 Tampilan di mana pemain dinyatakan kalah dengan status 'Failed'



Gambar 5.2 Tampilan di mana pemain dinyatakan kalah s out of time karenawaktu habis (Lihat time left)



Gambar 5.3 Tampilan di mana pemain memenangkan permainan atau berhasil menyelesaikan level

Aturan main pada math scrabble yakni pemain memindahkan dengan double klik *piece puzzle* ke petak putih yang kosong sampai membentuk soal matematika yang benar. Jika sudah dirasa selesai pemain menekan tombol 'SUBMIT'. Jika ada *piece puzzle* yang letaknya tidak sesuai muncul Failed (Gambar 5.1) jika pemain waktunya habis akan muncul tampilan Out of Time (Gambar 5.2), jika semuanya tepat maka muncul tampilan Level Complete (Gambar 5.3) Untuk video aturan main dapat diakses di alamat www.youtube.com/watch?v=ZzbNwvxlN4Y

5.4 Pengujian pengaturan tingkat kesulitan Math Scrabble

5.4.1 Skenario Uji pengaturan tingkat kesulitan Math Scrabble.

Skenario dilakukan dengan cara membandingkan atribut pada level yang ada pada permainan, dengan yang ada di perancangan, jika benar benar sama persis maka pengujian dikatakan berhasil.

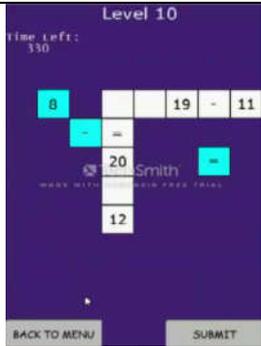
5.4.2 Hasil Uji pengaturan tingkat kesulitan Math Scrabble.

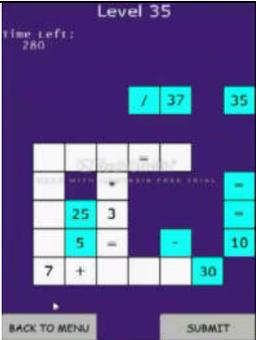
Hasil hanya yang diorekap di sini hanya beberapa saja yang cukup mewakili beberapa jumlah soal, untuk selengkapnya bisa dikunjungi di situs <https://youtu.be/iHG63h3AABg>.

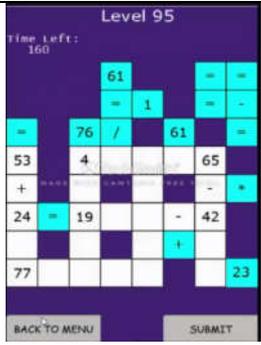
Jadi pada math scrabble tingkat kesulitan disusun dengan cara menaikkan jumlah petak kosong pada level yang memiliki jumlah soal yang sama, memperbanyak macam soal seperti perkalian dan pembagian pada kenaikan level tertentu, menaikkan angka maksimum yang keluar pada level tertentu serta menurunkan waktu penyelesaian level dua detik untuk setiap kenaikan satu level.

Tabel 5.6 Hasil Pengujian pengaturan tingkat kesulitan

Level	Jumlah soal	Jumlah petak kosong	Macam Soal	Waktu (detik)	Angka Maksimum	ScreenShoot

1	1 soal	1	+, -	348	20	
10	2 soal	3	+, -	330	20	
20	3 soal	5	+, -, *	310	20	

35	4 soal	10	+ , - , * , /	280	40	
50	5 soal	12	+ , - , * , /	250	60	
61	6 soal	8	+ , - , * , /	228	80	

95	7 soal	16	+, -, *, /	160	99	
----	--------	----	---------------	-----	----	---

5.5 Pengujian Fungsionalitas dengan metode black-box

Pengujian ini dilakukan untuk menguji apakah fungsionalitas yang diidentifikasi pada tahap kebutuhan benar-benar diimplementasikan dan bekerja semestinya. Selain itu juga untuk mengetahui kesesuaian keluaran dari setiap tahapan atau langkah penggunaan fitur terhadap skenario yang dipersiapkan. Pengujian dilakukan dengan menggunakan metode *black-box*.

5.5.1 Skenario Pengujian Fungsionalitas

Skenario Pengujian fungsionalitas digunakan untuk memberikan tahap-tahap dalam pengujian sistem. Skenario ini tertera pada Tabel 5.5.

Tabel 5.7 Pengujian Permainan

	UFG01
Kondisi Awal	Pengguna berada pada layar Judul permainan.
Prosedur Pengujian	Pengguna memainkan permainan hingga selesai dari beberapa fungsionalitas yang ingin diuji
Hasil yang diharapkan	Pengguna berhasil menyelesaikan permainan dan fungsionalitas

	permainan berjalan dengan lancar.
Hasil yang diperoleh	Pengguna berhasil menyelesaikan permainan dan fungsionalitas berjalan lancar.
Kesimpulan.	Pengujian berhasil.

5.5.1.1 PF01: Pengujian Kasus Penggunaan Memilih Level Selector

Pengujian dimulai ketika pemain telah masuk ke layar **Main Menu** sampai pemain mendapatkan tampilan Main Game seperti yang tercantum pada Gambar 5.4 .Lalu pemain menekan tombol 'PLAY', lalu sistem menampilkan tampilan Play Mode selanjutnya pemain menekan tombol LEVEL SELECTOR, lalu sistem menampilkan tampilan Level Selector, lalu pemain menekan salah satu tombol angka, dan munculah tampilan Main Game

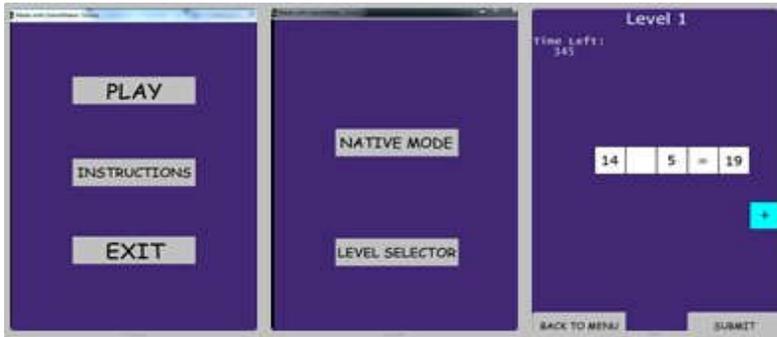


Gambar 5.4 Tampilan pengujian kasus penggunaan memilih level selector

5.5.1.2 PF02: Pengujian Kasus Penggunaan Memilih Native Mode

Pengujian dimulai ketika pemain telah masuk ke layar **Main Menu** sampai pemain mendapatkan tampilan Main Game seperti yang tercantum pada Gambar 5.4 .Lalu pemain menekan

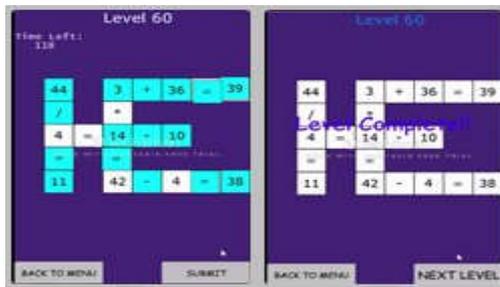
tombol ‘PLAY’, lalu sistem menampilkan tampilan Play Mode selanjutnya pemain menekan tombol LEVEL SELECTOR, lalu sistem menampilkan tampilan Main Game(level 1).



Gambar 5.5 Tampilan pengujian kasus penggunaan memilih native mode

5.5.1.3 PF03: Pengujian Kasus Penggunaan Menyelesaikan Level

Pengujian dimulai ketika pemain sudah masuk ke dalam Main Game lalu pemain memindahkan *piece puzzle* sehingga membentuk kalimat matematika yang benar. Sehingga ketika menekan tombol ‘SUBMIT’ akan muncul tampilan Level Complete



Gambar 5.6 Tampilan Pengujian Kasus Penggunaan Menyelesaikan Level

5.5.1.4 PF04: Pengujian Kasus Penggunaan Mengulangi Level

Pengujian dimulai ketika pemain sudah masuk ke dalam Main Game lalu pemain menekan tombol 'SUBMIT' dalam keadaan *piece puzzle* ada yang belum diposisi yang tepat sehingga akan muncul tampilan Failed. Dari tampilan failed pemain menekan tombol 'SUBMIT' untuk mengulangi level tersebut



Gambar 5.7 Tampilan pengujian kasus penggunaan Mengulangi Level

5.5.1.5 PF05: Pengujian Kasus Penggunaan Kembali Ke Menu

Pengujian dimulai ketika pemain berada pada tampilan yang terdapat tombol 'BACK TO MENU', lalu pemain menekan tombol tersebut dan munculah tampilan Main Menu.



Gambar 5.8 Tampilan pengujian kasus penggunaan Kembali ke Menu

5.5.1.6 PF06: Pengujian Kasus Penggunaan Menandai Level Terselesaikan

Pengujian berhasil jika pemain pernah menyelesaikan suatu level maka akan ada kotak angka berwarna putih pada level selector.



Gambar 5.9 Tampilan Pengujian Kasus Penggunaan Menandai Level Terselesaikan

5.5.1.7 Hasil Pengujian Fungsional

Hasil uji fungsionalitas yang sudah dilakukan berdasarkan pada PF01 hingga PF06 . Enam uji coba yang telah dilakukan menunjukkan bahwa semua fungsionalitas permainan berjalan dengan baik dan sesuai dengan sebagaimana mestinya skenario dan alur yang telah dibuat pada perancangan. Hasil pengujian fungsionalitas dicantumkan pada Tabel 5.6.

Tabel 5.8 Hasil Pengujian Fungsionalitas

No	Kode Pengujian	Hasil Pengujian
1	PF01	Berhasil
2	PF02	Berhasil
3	PF03	Berhasil
4	PF04	Berhasil
5	PF05	Berhasil
6	PF06	Berhasail

5.6 Pengujian Pengguna

Pengujian pada perangkat lunak yang dibangun tidak hanya dilakukan pada fungsionalitas yang dimiliki, tetapi juga pada pengguna untuk mencoba secara langsung. Pengujian ini berfungsi sebagai pengujian subjektif yang bertujuan untuk mengetahui tingkat keberhasilan aplikasi yang dibangun dari sisi pengguna. Hal ini dapat dicapai dengan meminta penilaian dan tanggapan dari pengguna terhadap sejumlah aspek perangkat lunak yang ada.

5.6.1 Skenario Uji Coba Pengguna

Dalam melakukan pengujian perangkat lunak, penguji diminta mencoba menggunakan perangkat lunak untuk mencoba semua fungsionalitas dan fitur yang ada. Serta untuk mencoba materi-materi dan isi dari soal-soal edukasi yang ada pada permainan.

Pengujian aplikasi oleh pengguna dilakukan dengan sebelumnya memberikan informasi seputar aplikasi, kegunaan, dan fitur-fitur yang dimiliki. Setelah informasi tersampaikan, pengguna kemudian diarahkan untuk langsung mencoba aplikasi dengan spesifikasi lingkungan yang sama dengan yang telah diuraikan pada uji coba fungsionalitas.

Jumlah pengguna yang terlibat dalam pengujian perangkat lunak sebanyak sepuluh orang untuk penilaian materi edukasi dan lima orang untuk pengujian performa dan antarmuka. Dalam melakukan pengujian, pengguna melakukan percobaan lebih dari satu kali penggunaan dengan level yang sama untuk masing-masing pengguna.

Dalam memberikan penilaian dan tanggapan, penguji diberikan formulir pengujian perangkat lunak. Formulir pengujian perangkat lunak ini memiliki beberapa aspek penilaian dan pada bagian akhir terdapat saran untuk perbaikan fitur.

5.6.2 Daftar Penguji Perangkat Lunak

Pada subbab ini ditunjukkan daftar pengguna yang bertindak sebagai penguji coba aplikasi yang dibangun. Daftar nama penguji aplikasi ini dapat dilihat pada Tabel 5.7.

Tabel 5.9 Daftar Nama Penguji Coba Aplikasi

No	Nama	Pekerjaan
1	Gilang Ramadhan.	Mahasiswa Teknik Mesin ITS
2	Adli Nibras Pramudia.	Mahasiswa Arsitektur ITS
3	Adib Wahyu Kuncoro	Mahasiswa Teknik Informatika ITS
4	Ken Genesisius	Mahasiswa Teknik Informatika ITS
5	Mariya Kusuma Dewi	Mahasiswa PPPNS
6	Lani Siswanto	Siswa SDI Al Falah Surabaya
7	Erika Octaviani Arianti	Siswa SDI Al Falah Surabaya
8	Salsabila Q	Siswa SDI Al Falah Surabaya
9	Selvi Julia Putri	Siswa SDI Al Falah Surabaya
10	Sheva	Siswa SDI Sunan Giri Surabaya

5.6.3 Hasil Uji Coba Pengguna

Uji coba yang dilakukan terhadap beberapa pengguna memiliki beberapa aspek yang dipisahkan berdasarkan antarmuka dan fungsionalitas yang dimiliki. Sistem penilaian didasarkan pada skala penghitungan satu sampai empat di mana skala satu menunjukkan nilai terendah dan skala empat menunjukkan skala tertinggi. Penilaian akhir kemudian dilakukan dengan menghitung berapa banyak penguji yang memilih suata skala tertentu dan kemudian dicari nilai rata-ratanya. Hasil uji coba dipaparkan secara lengkap dengan disertai tabel yang dapat dilihat pada subbab.

5.6.3.1 Hasil Penilaian Antarmuka

Penilaian antarmuka difokuskan pada penilaian pengguna terhadap kemudahan penggunaan antarmuka dan sifat-sifat lain yang perlu dimiliki. Hasil penilaian pengguna terhadap antarmuka aplikasi dapat dilihat pada Tabel 5.8.

Tabel 5.10 Penilaian Antarmuka

No.	Antarmuka	Penilaian				Rata-Rata
		1	2	3	4	
1	Kemudahan Penggunaan	0	0	2	3	3,60
2	Kelengkapan Menu	0	1	3	0	2,75
3	Keindahan Tampilan	0	1	1	2	3,25
4	Kecepatan Pemilihan Menu/Fitur	0	1	1	3	3,40
5	Kesesuaian tema	0	2	1	2	3,00
6	Ketertarikan Bermain	0	2	2	1	2,80
Nilai Akhir						3,13

5.6.3.2 Hasil Penilaian Performa Sistem

Penilaian performa sistem difokuskan pada penilaian pengguna terhadap kemampuan aplikasi dalam menghasilkan performa dari interaksi pengguna. Penilaian ini juga ditujukan untuk mendapatkan tingkat kecepatan dan kelancaran sistem atas interaksi yang dibuat oleh pengguna. Hasil penilaian performa sistem dapat dilihat pada Tabel 5.9.

Tabel 5.11 Penilaian Performa Sistem

No.	Performa Sistem	Penilaian				Rata-Rata
		1	2	3	4	
1	Performa atau kinerja pada permainan	0	0	0	5	4,00
2	Nilai pembangkit <i>puzzle</i> pada game	0	0	3	2	3,40
3	Kesesuaian interaksi dengan hasil yang seharusnya	0	0	2	3	3,60
Nilai Akhir						3,67

5.6.3.3 Hasil Penilaian Materi Edukasi

Penilaian materi edukasi difokuskan pada penilaian isi data didalam permainan yang berkaitan dengan edukasi. Penilaian ini juga difokuskan pada penilaian kecepatan dan ketepatan sistem dalam mengatasi semua yang berkaitan edukasi. Terdapat Hasil penilaian materi edukasi dapat dilihat pada Tabel 5.10.

Tabel 5.12 Penilaian Materi Edukasi

No.	Materi Edukasi	Penilaian				Rata-Rata
		1	2	3	4	
1	Nilai Edukasi dalam game	0	2	5	3	3,10
2	Pengaruh Melatih kecepatan hitung	2	4	2	2	2,40
3	Pengaruh Melatih ketepatan menjawab	2	4	2	2	2,40
4	Membantu dalam memahami pelajaran matematika	0	5	3	2	2,70
5	Ketertarikan untuk mempelajari pelajaran Matematika	1	2	4	3	2,90
6	kecocokan permainan untuk target pengguna	0	0	0	10	4,00
7	Nilai Fun (<i>Antimonoton</i>) untuk materi edukasi pada game	1	0	1	8	3,60
8	Kesesuaian materi edukasi pada tiap level	1	0	7	2	3,00
Nilai Akhir						3,01

5.6.4 Hasil Pengujian Pengguna

Evaluasi pengujian pengguna dilakukan dengan menampilkan data rekapitulasi perangkat lunak yang telah dipaparkan. Dalam hal ini, rekapitulasi disusun dalam bentuk tabel yang dapat dilihat pada Tabel 5.11. Dari data diketahui bahwa aplikasi telah memenuhi unsur yang seharusnya seperti perancangan dimana nilai prosentase memiliki lebih dari 80%.

Arti dari nilai yang memiliki lebih dari 80% ini merupakan penilaian yang sangat baik. Permainan yang dibuat telah mendapatkan apresiasi oleh pengguna sebenarnya. Sehingga pula dapat disimpulkan bahwa pengguna secara tidak langsung cukup menyetujui dan memberikan komentar baik kepada aplikasi permainan yang dibuat.

Tabel 5.13 Rekapitulasi Hasil Uji Coba Pengguna

No.	Nama Pengujian		Rata-Rata	Nilai	Nilai (%)
1	Penilaian Antarmuka	Kemudahan Penggunaan	3,60	3,13	78
		Kelengkapan Menu pada Permainan	2,75		
		Keindahan Tampilan	3,25		
		Kecepatan Pemilihan Menu/Fitur	3,40		
		Kesesuaian tema pada Permainan	3,00		
		Ketertarikan Bermain	2,80		
2	Performa Sistem permainan	Performa atau kinerja pada permainan	4,00	3,67	92
		Nilai dinamis level pada game	3,40		
		Kesesuaian interaksi dengan hasil yang seharusnya	3,60		

3	Penilaian Materi Edukasi	Nilai Edukasi dalam game	3,10	3,01	73
		Pengaruh Melatih kecepatan hitung	2,40		
		Pengaruh Melatih ketepatan menjawab	2,40		
		Membantu dalam memahami pelajaran matematika	2,70		
		Ketertarikan untuk mempelajari pelajaran matematika	2,90		
		kecocokan permainan untuk target pengguna	4,00		
		Nilai Fun (<i>antimonoton</i>) untuk materi edukasi pada game	3,60		
		Kesesuaian materi edukasi pada tiap level	3,00		
Rata Rata					82

(Halaman ini sengaja dikosongkan)

BAB VI KESIMPULAN DAN SARAN

Bab ini membahas mengenai kesimpulan yang dapat diambil dari tujuan pembuatan perangkat lunak dan hasil uji coba yang telah dilakukan sebagai jawaban dari rumusan masalah yang dikemukakan. Selain kesimpulan, terdapat pula saran yang ditujukan untuk pengembangan perangkat lunak lebih lanjut.

6.1. Kesimpulan

Dalam proses pengerjaan tugas akhir mulai dari tahap analisis, desain, implementasi, hingga pengujian didapatkan kesimpulan sebagai berikut:

1. Berdasarkan uji pembangkit *puzzle*, aplikasi berhasil membangkitkan *puzzle* dengan berbagai macam kombinasi dari *pattern*, soal, letak *piece puzzle*, dan letak petak kosong yang memiliki kemungkinan yang relatif kecil untuk ditemukan bentuk yang sama pada banyak percobaan, hasil pembangkitan *puzzle* dapat dilihat pada **Lampiran 2**.
2. Aplikasi berhasil membangun permainan menggunakan pembangkit *puzzle* dinamis khususnya untuk materi edukasi. Keandalan permainan diuji menggunakan kebutuhan fungsional dan kebutuhan pengguna yang bernilai lebih dari 80%. Berdasarkan uraian yang telah disampaikan dapat disimpulkan semua fungsionalitas permainan berfungsi dengan baik.
3. Rancangan aturan main Math Scrabble dapat tersusun secara sistematis yakni Pemain meletakkan *piece puzzle* di petak kosong putih dengan tepat agar dapat meraih kemenangan, tidak lupa pula pemain juga harus menyelesaikan hal itu dalam waktu yang ditentukan. Jika peletakan *piece puzzle* salah maka akan muncul tampilan Failed, jika waktu habis maka akan muncul tampilan Out of Time.

4. Skenario Math Scrabble yakni permainan disajikan dalam kotak besar berukuran 8x8 , yang di dalamnya terdapat teka teki silang kalimat matematika, dari satu soal hingga tujuh soal. Di sana terdapat waktu bermain yang mana berkurang dua detik setiap kenaikan satu levelnya. Kalimat matematika terdiri dari penjumlahan , pengurangan, perkalian , dan atau pembagian.
5. Tingkat kesulitan diatur dengan cara mengurangi dua detik waktu total penyelesaian level untuk setiap kenaikan satu levelnya, adapun jenis soal yanmg semakin beragam seiring dengan naiknya level dan jumlah soal yang semakin meningkat, lalu ditambah lagi dengan piece puzzle yang semakin banyak sehingga usaha untuk menyelesaikan level tentunya juga semakin besar. Perancangan level dapat dilihat pada **Lampiran 3** dan keseluruhan implementasinya dapat dilihat di youtu.be/iHG63h3AABg.

6.2. Saran

Berikut merupakan beberapa saran untuk pengembangan sistem di masa yang akan datang, berdasarkan pada hasil perancangan, implementasi dan uji coba yang telah dilakukan.

1. Game play, sebaiknya diubah menjadi *double click* yakni menekan klik objek dan menekan klik tempat yang dituju objek

DAFTAR PUSTAKA

- [1] YoYo Games, "Data Types," YoYo Games, 2015. [Online]. Available: https://docs.yoyogames.com/source/dadiospice/002_reference/001_gml%20language%20overview/data%20types.html. [Accessed 29 03 2017].
- [2] YoYo Games, "Global Variables," YoYo Games, 2015. [Online]. Available: https://docs.yoyogames.com/source/dadiospice/002_reference/001_gml%20language%20overview/variables/global%20variables.html. [Accessed 29 03 2017].
- [3] YoYo Games, "Rooms," YoYo Games, 2015. [Online]. Available: https://docs.yoyogames.com/source/dadiospice/002_reference/rooms/index.html. [Accessed 29 03 2017].
- [4] YoYo Games, "Mouse Input," YoYo Games, 2015. [Online]. Available: https://docs.yoyogames.com/source/dadiospice/002_reference/mouse,%20keyboard%20and%20other%20controls/mouse%20input/index.html. [Accessed 29 03 2017].
- [5] YoYo Games, "Arrays," YoYo Games, 2015. [Online]. Available: https://docs.yoyogames.com/source/dadiospice/002_reference/001_gml%20language%20overview/401_06_arrays.html. [Accessed 29 03 2017].
- [6] YoYo Games, "Expressions," YoYo Games, 2015. [Online]. Available: https://docs.yoyogames.com/source/dadiospice/002_reference/001_gml%20language%20overview/401_04_expressions.html.

- [Accessed 29 03 2017].
- [7] YoYo Games, "Real Number Functions," YoYo Games, 2015. [Online]. Available: https://docs.yoyogames.com/source/dadiospice/002_reference/maths/real%20valued%20functions/index.html. [Accessed 29 03 2017].
- [8] YoYo Games, " Drawing," YoYo Games, 2015. [Online]. Available: https://docs.yoyogames.com/source/dadiospice/002_reference/drawing/index.html. [Accessed 29 03 2017].
- [9] YoYo Games, "Miscellaneous Functions," YoYo Games, 2015. [Online]. Available: https://docs.yoyogames.com/source/dadiospice/002_reference/miscellaneous/index.html. [Accessed 29 03 2017].
- [10] Wikipedia, "Scrabble," Wikipedia, 2017. [Online]. Available: <https://id.wikipedia.org/wiki/Scrabble>. [Accessed 17 06 2017].
- [11] Matematika Kelompok Teknologi, Kesehatan, dan Pertanian, "Kalimat Matematika," Grafindo Media Pratama., 2016. [Online]. Available: https://books.google.co.id/books?id=l4SUX1M7W8gC&pg=PA54-IA5&lpg=PA54-IA5&dq=kalimat+matematika&source=bl&ots=JbEHPw-XaY&sig=6MVfRM1anGvFVkcAgccDUJq3H5Y&hl=id&sa=X&redir_esc=y#v=onepage&q=kalimat%20matematika&f=false. [Accessed 17 06 2017].
- [12] YoYo Games, " More About Objects," YoYo Games, 2015. [Online]. Available: https://docs.yoyogames.com/source/dadiospice/001_advanced%20use/more%20about%20objects/index.html. [Accessed 29 03 2017].

LAMPIRAN 1

Berikut Source code dari masing masing kasus penggunaan yang dapat dilampirkan

1. Memilih Level selector Mode

```

/*--MEMILIH LEVEL SELECTOR--*/
//Source code pada tombol play

room_goto(room107);
//Source code pada tombol Level Selector

room_goto(room100)
//Source code pada room level selector 1
global.obj_link_level[1]=obj_link_level_1;
global.obj_link_level[2]=obj_link_level_2;
global.obj_link_level[3]=obj_link_level_3;
global.obj_link_level[4]=obj_link_level_4;
global.obj_link_level[5]=obj_link_level_5;
global.obj_link_level[6]=obj_link_level_6;
global.obj_link_level[7]=obj_link_level_7;
global.obj_link_level[8]=obj_link_level_8;
global.obj_link_level[9]=obj_link_level_9;
global.obj_link_level[10]=obj_link_level_10;
global.obj_link_level[11]=obj_link_level_11;
global.obj_link_level[12]=obj_link_level_12;
global.obj_link_level[13]=obj_link_level_13;
global.obj_link_level[14]=obj_link_level_14;
global.obj_link_level[15]=obj_link_level_15;
global.obj_link_level[16]=obj_link_level_16;
global.obj_link_level[17]=obj_link_level_17;
global.obj_link_level[18]=obj_link_level_18;
global.obj_link_level[19]=obj_link_level_19;
global.obj_link_level[20]=obj_link_level_20;
global.obj_link_level[21]=obj_link_level_21;
global.obj_link_level[22]=obj_link_level_22;
global.obj_link_level[23]=obj_link_level_23;
global.obj_link_level[24]=obj_link_level_24;
global.obj_link_level[25]=obj_link_level_25;
global.obj_link_level[26]=obj_link_level_26;
global.obj_link_level[27]=obj_link_level_27;
global.obj_link_level[28]=obj_link_level_28;
global.obj_link_level[29]=obj_link_level_29;
global.obj_link_level[30]=obj_link_level_30;

```

```

global.obj_link_level[31]=obj_link_level_31;
global.obj_link_level[32]=obj_link_level_32;
global.obj_link_level[33]=obj_link_level_33;
global.obj_link_level[34]=obj_link_level_34;
global.obj_link_level[35]=obj_link_level_35;
global.obj_link_level[36]=obj_link_level_36;
global.obj_link_level[37]=obj_link_level_37;
global.obj_link_level[38]=obj_link_level_38;
global.obj_link_level[39]=obj_link_level_39;
global.obj_link_level[40]=obj_link_level_40;
global.obj_link_level[41]=obj_link_level_41;
global.obj_link_level[42]=obj_link_level_42;
global.obj_link_level[43]=obj_link_level_43;
global.obj_link_level[44]=obj_link_level_44;
global.obj_link_level[45]=obj_link_level_45;
global.obj_link_level[46]=obj_link_level_46;
global.obj_link_level[47]=obj_link_level_47;
global.obj_link_level[48]=obj_link_level_48;
global.obj_link_level[49]=obj_link_level_49;
global.obj_link_level[50]=obj_link_level_50;
a=1
for (i=a;i<=50;i++)
{
    var_mod=(i-a)%8;
    var_bagi=floor((i-a+8)/8)
    if(global.berhasil[i]==1)
    {

instance_create(var_mod*64,(var_bagi*64)+64,global.obj
j_link2_level[i])
    }
    else{

instance_create(var_mod*64,(var_bagi*64)+64,global.obj
j_link_level[i])
    }
}
//Source code pada tombol angka (misal tombol angka
95), tombol angka akan berisi nama level yg di load
yakni Level_X

room_goto(Level_95)

//Source code pada main game (Level 95)
global.level=86;
global.waktu = 160
global.angka_max =84;
//petak kosong =1
global.jumlah_soal =7;
i=0;

```

```

/*inisialisai_pattern*/
randomize();
rand_1=irandom_range(1,12);
instance_create(buat_objek_7_soal);

for(i=1;i<=8;i++)
{
    for(j=1;j<=8;j++)
    {
        koor[i,j]=0;
    }
}
for(i=1;i<=7;i++)
{
    for(j=1;j<=5;j++)

    {

        XX=Soal_X[i,j]
        YY=Soal_Y[i,j]
        koor[XX,YY]=1;
    }
}
n_size=26;
for(i=1;i<=n_size;i++)
{
    role[i]=0;
}
global.jwb=16
for(i=1;i<=global.jwb;i++)
{
    randomize();
    n=irandom_range(1,n_size);
    while(role[n]!=0)
    {
        randomize();
        n=irandom_range(1,n_size);
    }
    global.pos_Jawab_X_fix[i]=global.pos_Soal_X_fix[n]+32
;
    global.pos_Jawab_Y_fix[i]=global.pos_Soal_Y_fix[n]+32
;
    global.jawaban[i]=global.val_fix[n];
    role[n]=1;
}
for(i=1;i<=n_size;i++)
{

```

```

n=global.val_fix[i]
if(n>104 || n<0)
{

instance_create(global.pos_Soal_X_fix[i],global.pos_S
oal_Y_fix[i],global.obj_soal[0]);
}
else{
if(role[i]==0)
{
n=global.val_fix[i];

instance_create(global.pos_Soal_X_fix[i],global.pos_S
oal_Y_fix[i],global.obj_soal[n]);
}
if(role[i]==1)
{
n=global.val_fix[i];

instance_create(global.pos_Soal_X_fix[i],global.pos_S
oal_Y_fix[i],obj_soal_);
}
}

}
for(i=1;i<=global.jwb;i++)
{
randomize();
XXX=irandom_range(1,8)
YYY=irandom_range(1,8)
n=global.jawaban[i]
while(koor[XXX,YYY]!=0)
{
randomize();
XXX=irandom_range(1,8)
YYY=irandom_range(1,8)
}
koor[XXX,YYY]=2;

//instance_create(global.pos_Soal_X_fix[i],global.pos
_Soal_Y_fix[i],global.obj_piece[n]);
instance_create((XXX*64)-
64,(YYY*64)+64,global.obj_piece[n]);
}
instance_create(0,64,timer);
instance_create(0,704,obj_back_to_menu);

```

2. Memilih Native Mode

```

/*--MEMILIH NATIVE MODE--*/
//Source code pada tombol PLAY

room_goto(room107);
//Source code pada tombol NATIVE MODE

room_goto(Level_1)
//Source code pada main game (Level 1)
global.level=1;
global.waktu = 348;
global.angka_max =20;
//petak kosong =1
global.jumlah_soal =1;
i=0;

/*inisialisai pattern*/
randomize();
rand_1=irandom_range(1,2);
instance_create(buat_objek_1_soal);
n_size=5;

for(i=1;i<=8;i++)
{
  for(j=1;j<=8;j++)
  {
    koor[i,j]=0;
  }
}
for(i=1;i<=1;i++)
{
  for(j=1;j<=5;j++)

  {

    XX=Soal_X[i,j]
    YY=Soal_Y[i,j]
    koor[XX,YY]=1;
  }
}
for(i=1;i<=n_size;i++)
{
  role[i]=0;
}
global.jwb=1
for(i=1;i<=global.jwb;i++)

```

```

{
    randomize();
    n=irandom_range(1,n_size);
while(role[n]!=0)
{
    randomize();
    n=irandom_range(1,n_size);
}
global.pos_Jawab_X_fix[i]=global.pos_Soal_X_fix[n]+32;
global.pos_Jawab_Y_fix[i]=global.pos_Soal_Y_fix[n]+32;
global.jawaban[i]=global.val_fix[n];
    role[n]=1;

}
for(i=1;i<=n_size;i++)
{
    n=global.val_fix[i]
    if(n>104 || n<0)
    {

instance_create(global.pos_Soal_X_fix[i],global.pos_Soal_Y_f
ix[i],global.obj_soal[0]);
    }
    else{
        if(role[i]==0)
        {
            n=global.val_fix[i];

instance_create(global.pos_Soal_X_fix[i],global.pos_Soal_Y_f
ix[i],global.obj_soal[n]);
        }
        if(role[i]==1)
        {
            n=global.val_fix[i];

instance_create(global.pos_Soal_X_fix[i],global.pos_Soal_Y_f
ix[i],obj_soal_);
        }
    }

}
for(i=1;i<=global.jwb;i++)
{
randomize();
XXX=irandom_range(1,8)
YYY=irandom_range(1,8)
n=global.jawaban[i]
while(koor[XXX,YYY]!=0)
{
    randomize();

```

```

    XXX=irandom_range(1,8)
    YYY=irandom_range(1,8)
    }
    koor[XXX,YYY]=2;

//instance_create(global.pos_Soal_X_fix[i],global.pos_Soal_Y
_fix[i],global.obj_piece[n]);
    instance_create((XXX*64)-
64,(YYY*64)+64,global.obj_piece[n]);
}
instance_create(0,64,timer);
instance_create(0,704,obj_back_to_menu);

```

3. Menyelesaikan level

```

/*--MENYELESAIKAN LEVEL--*/
/*action piece puzzle*/

//Create
for(i=0;i<=104;i++)
{
    object_set_depth(global.obj_piece[i], -100);
}
//Destroy

instance_create(global.cursor_x,global.cursor_y,global.obj_m
oving[global.cursor_value])
//step
if(mouse_check_button_pressed(mb_left)){
    for(i=1;i<=global.jwb;i++)
    {
        for(j=0;j<=104;j++)
        {
            if(mouse_x==global.pos_Jawab_X_fix[i]&&
mouse_y==global.pos_Jawab_Y_fix[i] &&
position_meeting(mouse_x,mouse_y,global.obj_piece[j]))
            {
                global.cursor_x=global.pos_Jawab_X_fix[i]
                global.cursor_x=global.pos_Jawab_Y_fix[i]
                global.cursor_value=j;
                var
inst=instance_position(mouse_x,mouse_y,global.obj_piece[j])
                if inst{
                    with obj_piece_
                    {
                        instance_destroy()
                    }
                }
            }
        }
    }
}

```

```

        }
    }
}
//Mouse pressed
if(global.counter_click==1){
    for(i=0;i<=104;i++)
    {
        obj=    global.obj_moving[i];
                var inst =    instance_position(global.cursor_x,
global.cursor_y, obj);

//instance_change(global.obj_moving[global.cursor_value],
true);
        if inst {
            with inst instance_destroy();
        }
    }
}

for(i=0;i<=104;i++)
{
    if(position_meeting(mouse_x,                mouse_y,
global.obj_piece[i]))
    {
        with obj_moving_parent
        { instance_destroy();
        }
        xx=mouse_x%64
        yy=mouse_y%64
        global.cursor_x=mouse_x-xx;
        global.cursor_y=mouse_y-yy;
        global.counter_click=1;
        global.cursor_value=i;

//instance_create(global.obj_moving[global.cursor_value],glo
bal.cursor_x,global.cursor_y)
        obj=    global.obj_piece[i];
                var inst =    instance_position(global.cursor_x,
global.cursor_y, obj);

//instance_change(global.obj_moving[global.cursor_value],
true);

        if inst {
            with inst instance_destroy();
        }
    }
}
}

```

```

//action pada obj_moving_parent
//Destroy
instance_create(global.cursor_x,global.cursor_y,global.obj_p
iece[global.cursor_value])

global.counter_click=0;
//Step
if mouse_check_button_pressed(mb_left)
{
    for(i=0;i<=104;i++)
    {

if(position_meeting(mouse_x,mouse_y,global.obj_soal[i]))
        {
            var inst = instance_position(global.cursor_x,
global.cursor_y, obj_moving_parent);

                if inst {
                    with inst instance_destroy();
                }
            }
        if(mouse_x>0    &&  mouse_x<512    &&  mouse_y>128    &&
mouse_y<640)
        {
            for(i=1;i<=20;i++)
            {

if(position_empty(mouse_x,mouse_y)||position_meeting(mouse_x
,mouse_y,global.obj_kosong[i]))
                {
                    xx=mouse_x%64
                    yy=mouse_y%64
                    global.cursor_x=mouse_x-xx;
                    global.cursor_y=mouse_y-yy;
                    action_move_to(
global.cursor_x,global.cursor_y)
                    with obj_moving_parent instance_destroy();
                    var          inst          =
instance_position(global.cursor_x,
global.cursor_y,global.obj_kosong[i]);

                        if inst {
                            with inst instance_destroy();
                        }
                    }
                }

if(position_meeting(mouse_x,mouse_y,obj_soal_)
        {

```

```

        xx=mouse_x%64
        yy=mouse_y%64
        global.cursor_x=mouse_x-xx;
        global.cursor_y=mouse_y-yy;
        action_move_to(
global.cursor_x,global.cursor_y)
            with obj_moving_parent instance_destroy();
            var inst =
instance_position(global.cursor_x,
global.cursor_y,obj_soal_);

                if inst {
                    with inst instance_destroy();
                }
            }
        }
        global.counter_click=0;
    }
}
//Left Pressed
for(i=0;i<=104;i++)
{
    if(position_meeting(mouse_x, mouse_y,
global.obj_moving[i]))
    {
        xx=mouse_x%64
        yy=mouse_y%64
        global.cursor_x=mouse_x-xx;
        global.cursor_y=mouse_y-yy;
        global.cursor_value=i;

        obj= global.obj_moving[i];
        var inst = instance_position(global.cursor_x,
global.cursor_y, obj);

            if inst {
                with inst instance_destroy();
            }
        }
    }
}
//Actiojn pada tombol submit
//left_ppressed
global.submit_click=1
global.nilai=0;
for(i=1;i<=global.jwb;i++)
{
    n=global.jawaban[i]
    x1=global.pos_Jawab_X_fix[i]
    y1=global.pos_Jawab_Y_fix[i]
    x2=x1-32

```

```

y2=y1-32

if(position_meeting(x1,y1,global.obj_piece[n])           ||
position_meeting(x1,y1,global.obj_moving[n] ))
{
global.nilai++;
object_set_depth(global.obj_soal[n], -100);
instance_create(x2,y2,global.obj_soal[n])
}
if(!position_meeting(x1,y1,global.obj_piece[n])){

object_set_depth(global.obj_wrong[n], -100);
instance_create(x2,y2,global.obj_wrong[n])

}
}

with (timer)
{
instance_destroy();
}
with (obj_piece_)
{
instance_destroy();
}

if(global.nilai==global.jwb)
{
instance_create(320,704,obj_next_level)
instance_create(64,256,obj_level_complete)

//room_goto_next();
}
else
{
instance_create(320,704,obj_restart_level)
instance_create(64,256,obj_failed)
}
//room_goto(room1);

```

4. Mengulangi Level

```

/*--MENYELESAIKAN LEVEL--*/
/*action tombol RESTART LEVEL */
room_restart()

```

5. Kembali ke Main Menu

```

/*--KEMBALI KE MENU--*/

```

```
//Sourcode pada tombol BACK TO MENU
room_goto(menu_awal);
```

6. Menandai Level terselesaikan

```
/*--MENANDAI LEVEL TERSELESAIKAN--*/
//Action pada level complete
for(i=1;i<=100;i++){
    if(room==global.room_level[i])
    {
        global.berhasil[i]=1
    }
}
//Source code pada room level selector 1
a=1
for(i=a;i<=50;i++)
{
    var_mod=(i-a)%8;
    var_bagi=floor((i-a+8)/8)
    if(global.berhasil[i]==1)
    {

instance_create(var_mod*64,(var_bagi*64)+64,global.ob
j_link2_level[i])
    }
    else{

instance_create(var_mod*64,(var_bagi*64)+64,global.ob
j_link_level[i])
    }

}
//Source code pada room level selector 2
a=51
for(i=a;i<=100;i++)
{
    var_mod=(i-a)%8;
    var_bagi=floor((i-a+8)/8)
    if(global.berhasil[i]==1)
    {

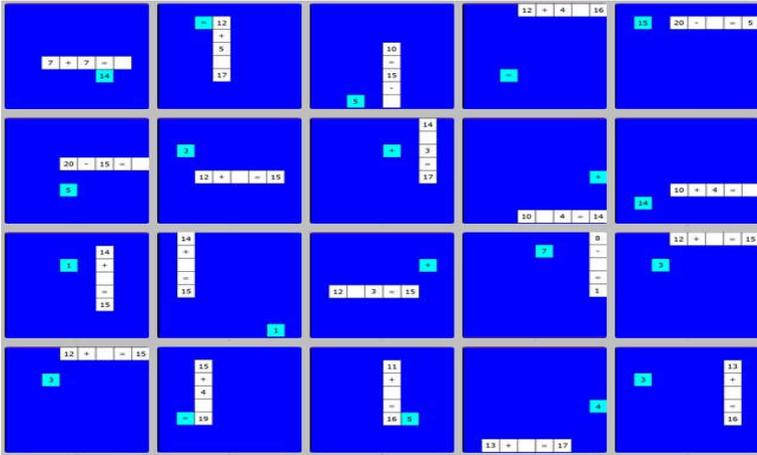
instance_create(var_mod*64,(var_bagi*64)+64,global.ob
j_link2_level[i])
    }
    else{

instance_create(var_mod*64,(var_bagi*64)+64,global.ob
j_link_level[i])
    }

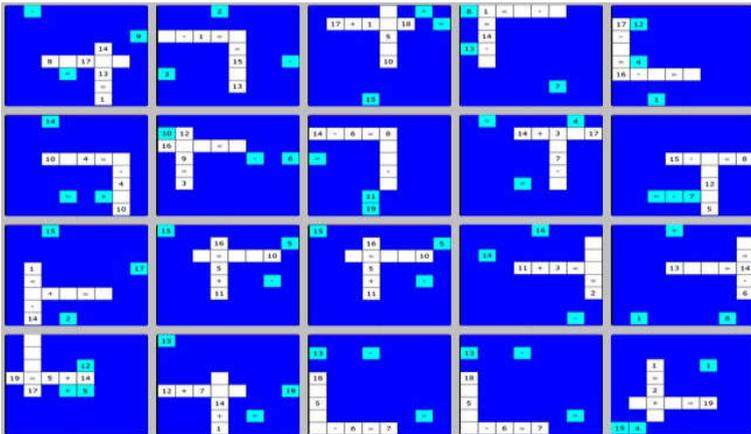
}
}
```

LAMPIRAN 2

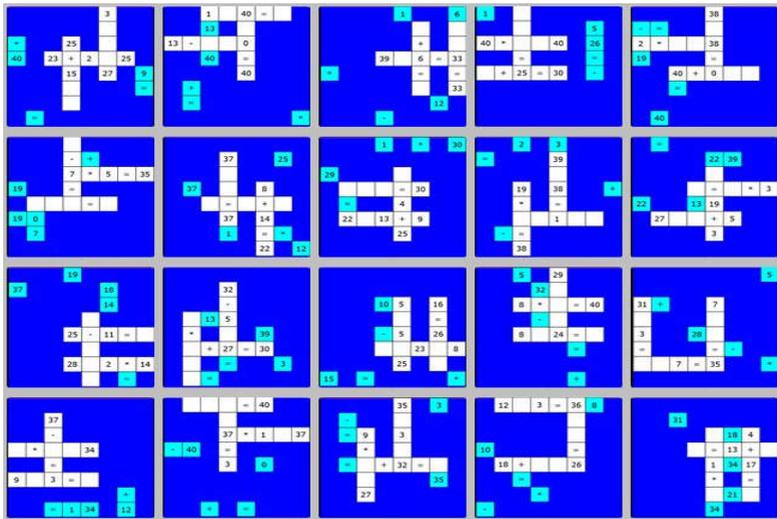
1. 20 percobaan pembangkit level 1



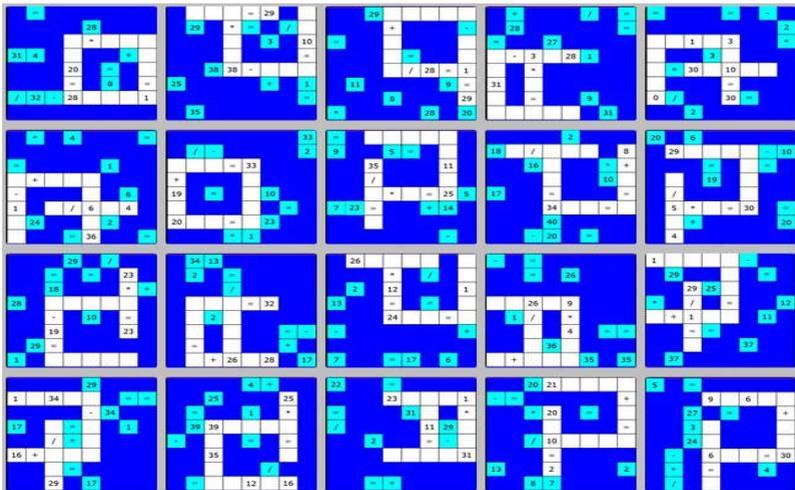
2. 20 percobaan pembangkit level 10



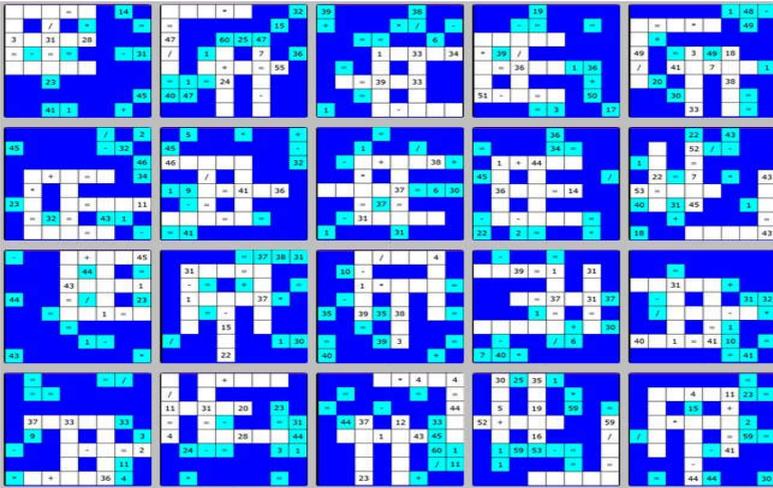
3. 20 percobaan pembangkit level 20



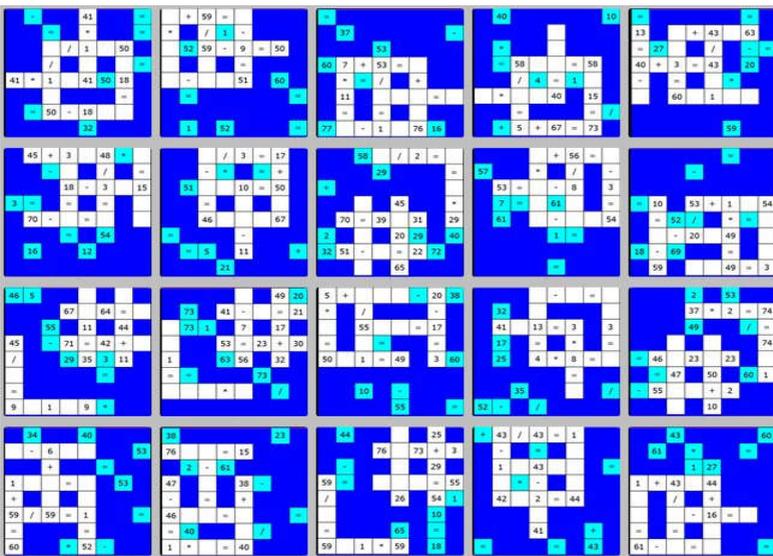
4. 20 percobaan pembangkit level 35



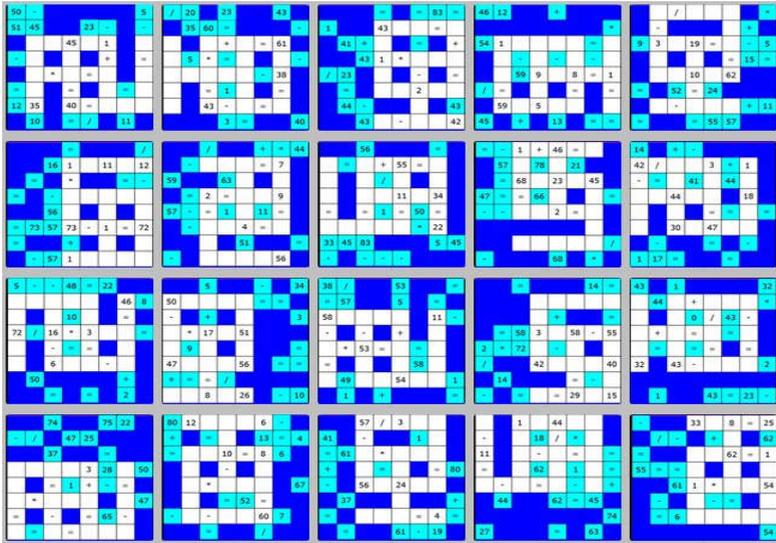
5. 20 percobaan pembangkit level 50



6. 20 percobaan pembangkit level 61



7. 20 percobaan pembangkit level 95



LAMPIRAN 3

1. Tabel Perancangan Level keseluruhan

Level	Rata rata total petak per pattern	Jumlah Petak Kosong	Waktu
1	5	1	348
2	5	1	346
3	5	2	344
4	5	2	342
5	5	2	340
6	9	2	338
7	9	2	336
8	9	2	334
9	9	3	332
10	9	3	330
11	9	3	328
12	9	3	326
13	9	4	324
14	9	4	322
15	9	4	320
16	12	4	318
17	12	4	316
18	12	4	314

19	12	5	312
20	12	5	310
21	12	5	308
22	12	6	306
23	12	6	304
24	12	6	302
25	12	6	300
26	12	6	298
27	12	7	296
28	12	7	294
29	16	7	292
30	16	7	290
31	16	8	288
32	16	8	286
33	16	9	284
34	16	9	282
35	16	10	280
36	16	10	278
37	16	11	276
38	16	11	274
39	16	12	272
40	16	12	270
41	20	8	268
42	20	8	266
43	20	9	264
44	20	9	262
45	20	10	260
46	20	10	258
47	20	11	256

48	20	11	254
49	20	12	252
50	20	12	250
51	20	13	248
52	20	13	246
53	20	14	244
54	20	14	242
55	20	15	240
56	20	15	238
57	20	16	236
58	20	16	234
59	20	16	232
60	24	16	230
61	24	8	228
62	24	8	226
63	24	9	224
64	24	9	222
65	24	10	220
66	24	10	218
67	24	11	216
68	24	11	214
69	24	12	212
70	24	12	210
71	24	13	208
72	24	13	206
73	24	14	204
74	24	14	202
75	24	15	200
76	24	15	198

77	24	16	196
78	24	16	194
79	24	17	192
80	24	17	190
81	24	18	188
82	24	18	186
83	24	19	184
84	24	19	182
85	24	20	180
86	26	12	178
87	26	12	176
88	26	13	174
89	26	13	172
90	26	14	170
91	26	14	168
92	26	15	166
93	26	15	164
94	26	16	162
95	26	16	160
96	26	17	158
97	26	17	156
98	26	18	154
99	26	18	152
100	26	20	150

Level	Angka Max	Macam Soal	Jumlah Soal
1	20	+, -	1

2	20	+, -	1
3	20	+, -	1
4	20	+, -	1
5	20	+, -	1
6	20	+, -	2
7	20	+, -	2
8	20	+, -	2
9	20	+, -	2
10	20	+, -	2
11	20	+, -	2
12	20	+, -	2
13	20	+, -	2
14	20	+, -	2
15	20	+, -	2
16	20	+, -, *	3
17	20	+, -, *	3
18	20	+, -, *	3
19	20	+, -, *	3
20	20	+, -, *	3
21	40	+, -, *	3
22	40	+, -, *	3
23	40	+, -, *	3
24	40	+, -, *	3
25	40	+, -, *	3
26	40	+, -, *	3
27	40	+, -, *	3
28	40	+, -, *	3
29	40	+, -, *, /	4
30	40	+, -, *, /	4

31	40	+, -, *, /	4
32	40	+, -, *, /	4
33	40	+, -, *, /	4
34	40	+, -, *, /	4
35	40	+, -, *, /	4
36	40	+, -, *, /	4
37	40	+, -, *, /	4
38	40	+, -, *, /	4
39	40	+, -, *, /	4
40	40	+, -, *, /	4
41	60	+, -, *, /	5
42	60	+, -, *, /	5
43	60	+, -, *, /	5
44	60	+, -, *, /	5
45	60	+, -, *, /	5
46	60	+, -, *, /	5
47	60	+, -, *, /	5
48	60	+, -, *, /	5
49	60	+, -, *, /	5
50	60	+, -, *, /	5
51	60	+, -, *, /	5
52	60	+, -, *, /	5
53	60	+, -, *, /	5
54	60	+, -, *, /	5
55	60	+, -, *, /	5
56	60	+, -, *, /	5
57	60	+, -, *, /	5
58	60	+, -, *, /	5
59	60	+, -, *, /	5

60	60	+, -, */	5
61	80	+, -, */	6
62	80	+, -, */	6
63	80	+, -, */	6
64	80	+, -, */	6
65	80	+, -, */	6
66	80	+, -, */	6
67	80	+, -, */	6
68	80	+, -, */	6
69	80	+, -, */	6
70	80	+, -, */	6
71	80	+, -, */	6
72	80	+, -, */	6
73	80	+, -, */	6
74	80	+, -, */	6
75	80	+, -, */	6
76	80	+, -, */	6
77	80	+, -, */	6
78	80	+, -, */	6
79	80	+, -, */	6
80	80	+, -, */	6
81	99	+, -, */	6
82	99	+, -, */	6
83	99	+, -, */	6
84	99	+, -, */	6
85	99	+, -, */	6
86	99	+, -, */	7
87	99	+, -, */	7
88	99	+, -, */	7

89	99	+, -, *, /	7
90	99	+, -, *, /	7
91	99	+, -, *, /	7
92	99	+, -, *, /	7
93	99	+, -, *, /	7
94	99	+, -, *, /	7
95	99	+, -, *, /	7
96	99	+, -, *, /	7
97	99	+, -, *, /	7
98	99	+, -, *, /	7
99	99	+, -, *, /	7
100	99	+, -, *, /	7

BIODATA PENULIS



Penulis dilahirkan di Madiun, 2 Agustus 1994, merupakan anak kedua dari 2 bersaudara. Penulis telah menempuh pendidikan formal yaitu TK Dharma Wanita Sogaten (1999-2001), SD Negeri 1 Kartoharjo Madiun (2001-2007), SMP Negeri 1 Madiun (2007-2010), SMA Negeri 2 Madiun (2010-2013), dan mahasiswa S1 Jurusan Teknik Informatika Institut Teknologi Sepuluh Nopember Surabaya rumpun mata kuliah Interaksi, Grafika, dan Seni (2013-2017).

Selama menjadi mahasiswa . Penulis yang memiliki hobi bermain *game* ,travelling, olahraga khususnya catur, bereksplorasi diri merupakan mahasiswa yang aktif dalam organisasi diantaranya, LMB ITS (staff ahli 2016-2017), UKM CATUR ITS (kadek eksternal 2014-2015, ketua umum 2015-2016) dan KMI ITS (Staff kaderisasi 2014-2015). Penulis dapat dihubungi melalui *surel* rifqimaula@gmail.com.,dan id line [maula_igbal](https://www.whatsapp.com/channel/00291111111111111111)