



KERJA PRAKTIK – IF184801

**Pembangunan Aplikasi Monitoring Program Kerja Bidang
Teknologi Informasi PT PJB Menggunakan Konsep *Software
Development Life Cycle* dengan Metode *Waterfall***

**PT Pembangkit Jawa Bali (PJB)
Kantor Pusat PT PJB Surabaya**

Periode 30 November 2020 – 21 Mei 2021

Oleh:

Hisam Widi Prayoga 0511174000026
Hendra Ramadani 0511174000055

Pembimbing Departemen

Wahyu Suadi, S.Kom., M.M., M.Kom.

Pembimbing Lapangan

Habib Amaluddin Mahfudz
Egga Bahartyan

DEPARTEMEN TEKNIK INFORMATIKA
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember
Surabaya 2021

[Halaman ini sengaja dikosongkan]



KERJA PRAKTIK – IF184801

Pembangunan Aplikasi Monitoring Program Kerja Bidang Teknologi Informasi PT PJB Menggunakan Konsep *Software Development Life Cycle* dengan Metode *Waterfall*

PT Pembangkit Jawa Bali (PJB)
Kantor Pusat PT PJB Surabaya

Oleh:

Hisam Widi Prayoga

05111740000026

Hendra Ramadani

05111740000055

Pembimbing Departemen

Wahyu Suadi, S.Kom., M.M., M.Kom.

Pembimbing Lapangan

Habib Amaluddin Mahfudz

Egga Bahartyan

DEPARTEMEN TEKNIK INFORMATIKA

Fakultas Teknologi Elektro dan Informatika Cerdas

Institut Teknologi Sepuluh Nopember

Surabaya 2021

[Halaman ini sengaja dikosongkan]

**LEMBAR PENGESAHAN
KERJA PRAKTIK**

**Pembangunan Aplikasi Monitoring Program Kerja
Bidang Teknologi Informasi PT PJB Menggunakan
Konsep *Software Development Life Cycle* dengan Metode
*Waterfall***

Oleh:

Hisam Widi Prayoga 05111740000026

Hendra Ramadani 05111740000055

Mengetahui,
Manajer Pengembangan
Teknologi Informasi & Analisa
Bisnis,
Pembimbing Kerja Praktik



Habib Amaluddin Mahfudz
NID 8308132JA

Mengetahui,
Assistant Analyst Pengembangan
Teknologi Informasi & Analisa
Bisnis,
Pembimbing Kerja Praktik



Egga Bahartyan
NID 9015051ZJY

Menyetujui,
Dosen Pembimbing,
Kerja Praktik



Wahyu Suadi
Wahyu Suadi, S.Kom., M.M., M.Kom.
NIP. 197110302002121001

SURABAYA, JULI 2021

[Halaman ini sengaja dikosongkan]

Pembangunan Aplikasi Monitoring Program Kerja Bidang Teknologi Informasi PT PJB Menggunakan Konsep *Software Development Life Cycle* dengan Metode *Waterfall*

Nama Mahasiswa : Hisam Widi Prayoga
Hendra Ramadani
NRP : 0511174000026
0511174000055
Departemen : Teknik Informatika FTEIC-ITS
Pembimbing Jurusan : Wahyu Suadi, S.Kom, M. Kom.
Pembimbing Lapangan : Habib Amaluddin Mahfudz
Egga Bahartyan

ABSTRAK

Monitoring program kerja di bidang BTIF PT PJB masih menggunakan cara lama, yaitu menggunakan *spreadsheet* dan belum ada sistem yang dapat mengelola dan memonitor program kerja di bidang BTIF. Selain pengelolaan program kerja, pengelolaan terkait permintaan juga masih dikelola secara manual melalui sebuah *spreadsheet*, sehingga dibutuhkan sebuah sistem yang dapat mengelola dan memonitor program kerja dan permintaan secara keseluruhan yang terdapat di bidang BTIF. Selain itu, sistem juga dapat melakukan *change request* dari sebuah permintaan menjadi program kerja.

Oleh sebab itu, penulis mengusulkan sebuah solusi, yaitu pembuatan sistem *monitoring* di bidang BTIF dengan menggunakan konsep *Software Development Life Cycle*. Konsep tersebut akan diterapkan menggunakan metode *waterfall*, dimana merupakan metode pembuatan suatu aplikasi dengan beberapa tahapan, yaitu dimulai dari analisis kebutuhan, desain aplikasi, pengembangan aplikasi, *deployment* aplikasi, dan *testing* aplikasi. Tahapan tersebut akan dilakukan secara berurutan, dari awal hingga akhir tahapan.

Sistem *monitoring* yang dibuat akan menggunakan arsitektur berbasis kontainer pada *deployment*-nya, dimana akan membagi sistem menjadi beberapa bagian sesuai fungsionalitasnya. Sistem *monitoring* juga akan diuji coba pada tiap fungsional yang dibuat pada sistemnya, sehingga hasil yang didapatkan adalah terbuatnya aplikasi BTIF Monitoring yang berjalan dengan baik dan sesuai dengan kebutuhan pengguna.

Kata kunci: Pengembangan Perangkat Lunak, Software Development Life Cycle, Container.

KATA PENGANTAR

Puji syukur penulis haturkan kepada Allah SWT, berkat anugrahnya penulis dapat melaksanakan salah satu kewajiban penulis sebagai mahasiswa Departemen Informatika, yakni Kerja Praktik (KP).

Penulis menyadari masih ada kekurangan baik dalam pelaksanaan kerja praktik maupun penyusunan buku laporan ini. Namun, saya berharap buku laporan ini dapat menambah wawasan pembaca dan dapat menjadi sumber referensi. Penulis mengharapkan kritik dan saran yang membangun untuk kesempurnaan buku laporan kerja praktik ini.

Melalui buku ini, penulis juga ingin menyampaikan rasa terima kasih kepada orang-orang yang telah membantu, baik langsung maupun tidak langsung, dalam pelaksanaan kerja praktik hingga penyusunan laporan. Orang-orang tersebut antara lain adalah:

1. Kedua Orang Tua penulis.
2. Bapak Wahyu Suadi, S.Kom., M.M., M.Kom. selaku dosen pembimbing kerja praktik.
3. Bapak Ary Mazharuddin Shiddiqi, S. Kom., M. Comp. Sc., Ph. D. selaku koordinator Kerja Praktik.
4. Bapak Sapto Nugroho, selaku Kepala Divisi Bidang Teknologi Informasi.
5. Bapak Habib Amaluddin Mahfudz selaku mentor lapangan serta manajer dari divisi Bidang Teknologi Informasi bagian Pengembangan Teknologi Informasi & Analisa Bisnis.
6. Bapak Egga Bahartyan selaku co-mentor lapangan serta Assistant Analyst dari divisi Bidang Teknologi Informasi bagian Pengembangan Teknologi Informasi & Analisa Bisnis.
7. Seluruh karyawan di Bidang Teknologi Informasi pada umumnya dan khususnya Mbak Nurul Isnaini yang sangat membantu penulis dalam proses magang.

8. Seluruh karyawan Divisi SDM dan Talenta khususnya ibu Partini dan mas Yoga Satrio atas arahan dan bantuan yang telah diberikan kepada penulis.
9. Semua rekan Program Magang Mahasiswa Bersertifikat angkatan ketiga yang sudah menemani penulis dan berbagi pengalaman selama magang.
10. Dan semua pihak yang telah membantu penulis dalam menyelesaikan quest secara keseluruhan.

Surabaya, 25 Juli 2021

Penulis

DAFTAR ISI

LEMBAR PENGESAHAN	v
ABSTRAK.....	vii
KATA PENGANTAR	ix
DAFTAR ISI.....	xi
DAFTAR TABEL.....	xiii
DAFTAR GAMBAR	xvii
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2. Tujuan	1
1.3. Manfaat	2
1.4. Rumusan masalah	2
1.5. Lokasi dan Waktu Kerja Praktik.....	3
1.6. Metodologi Kerja Praktik.....	3
1.7. Sistematika Laporan.....	4
BAB II PROFIL PERUSAHAAN	7
2.1. Profil PT. Pembangkitan Jawa Bali	7
2.2. Profil Bidang Teknologi Informasi PT PJB.....	8
BAB III TINJAUAN PUSTAKA	13
3.1 HTML	13
3.2 PHP	13
3.3 <i>Framework</i> Codeigniter	13
3.4 REST API	14
3.5 MySQL	14
3.6 Postman.....	15

3.7	Container.....	15
3.8	Docker.....	15
3.9	Docker Compose.....	15
3.10	Microservice	15
BAB IV Pengerjaan Kerja Praktik.....		17
4.1.	Analisis Kebutuhan Pengguna	18
4.2.	Desain Aplikasi	20
4.3.	Pengembangan Aplikasi.....	24
4.4.	<i>Deployment</i> Aplikasi.....	35
4.5.	<i>Testing</i> Aplikasi	42
BAB V KESIMPULAN.....		119
5.1.	Kesimpulan	119
5.2.	Saran	119
DAFTAR PUSTAKA		121
BIODATA PENULIS I.....		123
BIODATA PENULIS II		124

DAFTAR TABEL

Tabel 1. Daftar <i>Virtual Machine</i> yang tidak efisien.....	40
Tabel 2. Aplikasi dengan arsitektur <i>Monolith</i>	41
Tabel 3. Spesifikasi pengujian autentikasi aplikasi.....	42
Tabel 4. Spesifikasi pengujian melihat permintaan aplikasi	43
Tabel 5. Spesifikasi pengujian menambah permintaan aplikasi	44
Tabel 6. Spesifikasi pengujian mengubah permintaan aplikasi	44
Tabel 7. Spesifikasi pengujian menghapus permintaan aplikasi....	45
Tabel 8. Spesifikasi pengujian konversi permintaan menjadi program kerja.....	46
Tabel 9. Spesifikasi pengujian melihat melihat data helpdesk	48
Tabel 10. Spesifikasi pengujian menyimpan data helpdesk sebagai permintaan.....	48
Tabel 11. Spesifikasi pengujian melihat program kerja aplikasi ...	50
Tabel 12. Spesifikasi pengujian mengubah data satu program kerja	50
Tabel 13. Spesifikasi pengujian melihat aktivitas pada satu program kerja	52
Tabel 14. Spesifikasi pengujian menambah aktivitas pada satu program kerja.....	53
Tabel 15. Spesifikasi pengujian mengubah aktivitas pada satu program kerja.....	54
Tabel 16. Spesifikasi pengujian menghapus aktivitas pada satu program kerja.....	56
Tabel 17. Spesifikasi pengujian mengubah urutan aktivitas pada satu program kerja.....	58

Tabel 18. Spesifikasi pengujian mengubah project team pada satu program kerja.....	59
Tabel 19. Spesifikasi pengujian melihat data <i>client</i>	60
Tabel 20. Spesifikasi pengujian menambah data <i>client</i>	61
Tabel 21. Spesifikasi pengujian mengubah data <i>client</i>	62
Tabel 22. Spesifikasi pengujian menghapus data <i>client</i>	63
Tabel 23. Spesifikasi pengujian melihat data dokumen.....	64
Tabel 24. Spesifikasi pengujian menambah data dokumen.....	65
Tabel 25. Spesifikasi pengujian mengubah data dokumen.....	66
Tabel 26. Spesifikasi pengujian menghapus data dokumen.....	67
Tabel 27. Spesifikasi pengujian melihat data jenis program.....	68
Tabel 28. Spesifikasi pengujian menambah data jenis program....	69
Tabel 29. Spesifikasi pengujian mengubah data jenis program.....	70
Tabel 30. Spesifikasi pengujian menghapus data jenis program ...	71
Tabel 31. Spesifikasi pengujian melihat data kategori program....	72
Tabel 32. Spesifikasi pengujian menambah data kategori program.....	73
Tabel 33. Spesifikasi pengujian mengubah data kategori program	74
Tabel 34. Spesifikasi pengujian menghapus data kategori program.....	75
Tabel 35. Spesifikasi pengujian melihat data kelompok.....	76
Tabel 36. Spesifikasi pengujian menambah data kelompok.....	77
Tabel 37. Spesifikasi pengujian mengubah data kelompok.....	78
Tabel 38. Spesifikasi pengujian menghapus data kelompok.....	79
Tabel 39. Spesifikasi pengujian melihat data media.....	80
Tabel 40. Spesifikasi pengujian menambah data media.....	81

Tabel 41. Spesifikasi pengujian mengubah data media	82
Tabel 42. Spesifikasi pengujian menghapus data media.....	83
Tabel 43. Spesifikasi pengujian melihat data prioritas	84
Tabel 44. Spesifikasi pengujian menambah data prioritas.....	85
Tabel 45. Spesifikasi pengujian mengubah data prioritas.....	86
Tabel 46. Spesifikasi pengujian menghapus data prioritas	87
Tabel 47. Spesifikasi pengujian melihat data <i>resource</i>	88
Tabel 48. Spesifikasi pengujian menambah data <i>resource</i>	89
Tabel 49. Spesifikasi pengujian mengubah data <i>resource</i>	90
Tabel 50. Spesifikasi pengujian menghapus data <i>resource</i>	91
Tabel 51. Spesifikasi pengujian melihat data RJPP	92
Tabel 52. Spesifikasi pengujian menambah data RJPP	93
Tabel 53. Spesifikasi pengujian mengubah data RJPP	93
Tabel 54. Spesifikasi pengujian menghapus data RJPP.....	94
Tabel 55. Spesifikasi pengujian melihat data <i>roles</i>	95
Tabel 56. Spesifikasi pengujian menambah data <i>roles</i>	96
Tabel 57. Spesifikasi pengujian mengubah data <i>roles</i>	97
Tabel 58. Spesifikasi pengujian menghapus data <i>roles</i>	98
Tabel 59. Spesifikasi pengujian melihat data RSTI.....	99
Tabel 60. Spesifikasi pengujian menambah data RSTI	100
Tabel 61. Spesifikasi pengujian mengubah data RSTI	101
Tabel 62. Spesifikasi pengujian menghapus data RSTI.....	102
Tabel 63. Spesifikasi pengujian melihat data <i>service catalog</i>	103
Tabel 64. Spesifikasi pengujian menambah data <i>service catalog</i>	104
Tabel 65. Spesifikasi pengujian mengubah data <i>service catalog</i> .	105

Tabel 66. Spesifikasi pengujian menghapus data <i>service catalog</i>	106
Tabel 67. Spesifikasi pengujian melihat data status.....	107
Tabel 68. Spesifikasi pengujian menambah data status	108
Tabel 69. Spesifikasi pengujian mengubah data status	109
Tabel 70. Spesifikasi pengujian menghapus data status	110
Tabel 71. Spesifikasi pengujian melihat data <i>stream</i>	111
Tabel 72. Spesifikasi pengujian menambah data <i>stream</i>	112
Tabel 73. Spesifikasi pengujian mengubah data <i>stream</i>	112
Tabel 74. Spesifikasi pengujian menghapus data <i>stream</i>	113
Tabel 75. Spesifikasi pengujian melihat data sumber program ...	114
Tabel 76. Spesifikasi pengujian menambah data sumber program	115
Tabel 77. Spesifikasi pengujian mengubah data sumber program	116
Tabel 78. Spesifikasi pengujian menghapus data sumber program	117

DAFTAR GAMBAR

Gambar 1. Struktur Organisasi BTIF	8
Gambar 2. Proses Bisnis BTIF di setiap bagian.....	9
Gambar 3. Bagan Visi BTIF	10
Gambar 4. <i>Framework</i> Manajemen TI BTIF	11
Gambar 5. Tahapan SDLC.....	17
Gambar 6. Kasus Penggunaan Aplikasi BTIF Monitoring	19
Gambar 7. Halaman Login Aplikasi BTIF Monitoring	20
Gambar 8. Halaman List Permintaan Aplikasi BTIF Monitoring..	20
Gambar 9. Halaman Buat Permintaan Aplikasi BTIF Monitoring	21
Gambar 10. Halaman Detail Permintaan Aplikasi BTIF Monitoring	21
Gambar 12. <i>Class Diagram Frontend</i>	23
Gambar 13. <i>Class Diagram Backend</i>	24
Gambar 14. Dokumentasi API aplikasi BTIF Monitoring	26
Gambar 15. Tampilan Program dan detail program.....	28
Gambar 16. Tampilan ganttchart untuk <i>monitoring</i> program kerja	29
Gambar 17. Tampilan helpdesk	29
Gambar 18. Tampilan permintaan	29
Gambar 19. Tampilan buat permintaan baru	30
Gambar 20. Tampilan detail permintaan.....	30
Gambar 21. Tampilan data <i>master client</i>	30
Gambar 22. Tampilan data <i>master</i> dokumen	31
Gambar 23. Tampilan data <i>master</i> jenis program.....	31
Gambar 24. Tampilan data <i>master</i> kategori program	31

Gambar 25. Tampilan data <i>master</i> kelompok.....	32
Gambar 26. Tampilan data <i>master</i> media	32
Gambar 27. Tampilan data <i>master</i> prioritas.....	32
Gambar 28. Tampilan data <i>master resource</i>	33
Gambar 29. Tampilan data <i>master</i> RJPP	33
Gambar 30. Tampilan data <i>master roles</i>	33
Gambar 31. Tampilan data <i>master</i> RSTI	34
Gambar 32. Tampilan data <i>master service catalog</i>	34
Gambar 33. Tampilan data <i>master</i> status.....	34
Gambar 34. Tampilan data <i>master stream</i>	35
Gambar 35. Tampilan data <i>master</i> sumber program.....	35

BAB I

PENDAHULUAN

1.1. Latar Belakang

PT PJB merupakan BUMN yang sudah lama berdiri, dimana sudah ada sejak tahun 1995. Lamanya sebuah perusahaan berdiri, maka banyak program kerja yang sudah dibuat. Pada bidang BTIF sendiri, terdapat banyak program kerja yang terdaftar. Program kerja tersebut meliputi pembuatan aplikasi, pembaruan aplikasi, *maintenance*, dan lain sebagainya. Saat ini, belum ada sebuah sistem yang dapat mencatat, mengelola, dan memonitor program kerja yang ada di bidang BTIF sendiri. Pengelolaan masih dilakukan dengan menggunakan *spreadsheet*, sehingga masih dilakukan secara manual. Diperlukan sebuah sistem yang dapat melakukan pencatatan, pengelolaan, dan *monitoring* program kerja yang ada di bidang BTIF sendiri.

Selain hal tersebut, terdapat juga sebuah sistem pengelolaan permintaan, dimana permintaan merupakan *request* yang diajukan admin sebelum menjadi program kerja. Permintaan ini sudah dikelola pada suatu sistem, yaitu sistem helpdesk, namun pada sistem tersebut hanya mengelola permintaan tertentu saja, belum mengelola permintaan yang berasal dari RSTI bidang BTIF, serta belum menangani *change* dari permintaan ke program kerja. Dimana dalam melakukan *change* tersebut, diperlukan integrasi dengan data-data lainnya. Saat ini, *change* permintaan tersebut masih dilakukan manual, menggunakan sebuah *spreadsheet*. Sehingga muncul sebuah gagasan untuk membuat sebuah sistem *monitoring* yang dapat mencatat dan mengelola permintaan serta program kerja dalam sebuah sistem, dimana dalam sistem tersebut juga dapat melakukan *change request* dari permintaan ke program kerja.

1.2. Tujuan

Tujuan kerja praktik ini adalah untuk menyelesaikan kewajiban kuliah kerja praktik di Institut Teknologi Sepuluh Nopember dengan

beban dua SKS. Selain itu juga turut serta membantu PT. Pembangunan Jawa Bali khususnya divisi Bidang Teknologi Informasi dalam mengembangkan aplikasi web BTIF Monitoring menggunakan konsep *Software Development Life Cycle* untuk kemudahan *monitoring* program kerja divisi Bidang Teknologi Informasi.

1.3. Manfaat

Manfaat yang dapat diperoleh penulis dari kegiatan pengembangan web divisi Bidang Teknologi Informasi bagian Pengembangan Teknologi Informasi & Analisa Bisnis antara lain adalah:

1. Penulis dapat menerapkan ilmu pengetahuan tentang pengembangan aplikasi web yang selama ini diterima di bangku perkuliahan pada dunia kerja sesungguhnya.
2. Penulis mendapatkan ilmu terkait proses bisnis pengembangan teknologi Informasi di dalam internal PT Pembangunan Jawa Bali
3. Penulis mendapatkan ilmu terkait pengembangan aplikasi web menggunakan konsep *Software Development Life Cycle* yang dikombinasikan dengan *framework* Codeigniter 4.
4. Penulis mendapatkan ilmu baru terkait implementasi aplikasi web menggunakan teknologi *container* dan menerapkan arsitektur *microservices* dari proses implementasi yang tidak didapatkan di bangku perkuliahan.

1.4. Rumusan masalah

Berikut ini rumusan masalah pada kerja praktik di tim pengembangan teknologi informasi dan analisa bisnis divisi Bidang Teknologi Informasi:

1. Bagaimana membuat sebuah sistem *monitoring* yang dapat memudahkan *user* dalam melakukan *monitoring* program?
2. Bagaimana aplikasi BTIF Monitoring dibuat dengan konsep *Software Development Life Cycle*?

3. Bagaimana melakukan *deployment* aplikasi BTIF Monitoring menggunakan *docker* dengan konsep *container* untuk mengintegrasikan aplikasi *frontend* dan aplikasi *backend*?

1.5. Lokasi dan Waktu Kerja Praktik

Kerja praktik ini dilaksanakan pada waktu dan tempat sebagai berikut:

Lokasi : PT. Pembangunan Jawa Bali (Kantor Pusat)
Alamat : Jl. Ketintang Baru No.11, Ketintang, Kec. Gayungan,
Kota Surabaya, Jawa Timur 60231
Waktu : 30 Nopember 2020 - 21 Mei 2021
Hari Kerja : Senin - Jumat
Jam Kerja : 07.30 - 16.00

1.6. Metodologi Kerja Praktik

Tahapan pengerjaan kerja praktik dijabarkan sebagai berikut:

1. Perumusan Masalah

Untuk mengetahui bagaimana menyelesaikan permasalahan pada tiap pekerjaan, pembimbing lapangan bersama tim menjelaskan mengenai alur pengembangan web yang dilakukan di Bidang Teknologi Informasi. Pada tiap fitur yang dikembangkan juga dijelaskan mengenai proses bisnis pada fitur tersebut dan apa saja kebutuhan yang diberikan oleh klien. Dari penjelasan tersebut dihasilkan catatan-catatan penting mengenai gambaran sistem. Dengan begitu proses pengerjaan proyek pengembangan web dapat dilakukan sesuai dengan alur kerja di Bidang Teknologi Informasi dan juga dapat memenuhi kebutuhan klien secara keseluruhan.

2. Studi Literatur

Setelah ditentukan pembagian tugas pengembangan web oleh Bidang Teknologi Informasi, dilakukan studi literatur mengenai cara implementasinya sesuai kebutuhan *user* dan teknologi yang digunakan Bidang Teknologi Informasi. Pada tahap ini dilakukan proses pencarian, pembelajaran, pengumpulan dan pemahaman informasi serta literatur yang berkaitan untuk membantu dalam

merumuskan solusi dan implementasinya. Informasi bisa didapat dari internet untuk istilah-istilah umum yang digunakan dalam pengimplementasian suatu sistem atau aplikasi.

3. Solusi dan Implementasi Sistem

Tahap ini meliputi penjelasan mengenai solusi yang diberikan serta implementasi sistem yang didasarkan pada tahap perancangan dan analisis. Jika ada masukan atau perbaikan dari pembimbing lapangan maka akan segera diaplikasikan

4. Pengujian dan Evaluasi

Pengujian dilakukan dengan menguji fitur-fitur yang telah dibuat. Kesesuaian sistem dengan kebutuhan menentukan keberhasilan dalam pengujian. Hal ini menghasilkan hasil evaluasi apakah sistem sudah sesuai dengan tujuan dan kebutuhan atau belum.

1.7. Sistematika Laporan

Laporan kerja praktik ini terdiri dari tujuh bab dengan rincian sebagai berikut:

1. BAB I Pendahuluan

Pada bab ini dijelaskan tentang latar belakang permasalahan, tujuan, waktu pelaksanaan, serta sistematika pengerjaan kerja praktik dan juga penulisan laporan kerja praktik.

2. BAB II Profil Perusahaan

Pada bab ini, dijelaskan secara rinci tentang profil, perusahaan tempat kami melaksanakan kerja praktik, yakni PT. Pembangunan Jawa Bali divisi Bidang Teknologi Informasi bagian Pengembangan Teknologi Informasi dan Analisa Bisnis.

3. BAB III Tinjauan Pustaka

Pada bab ini, dijelaskan mengenai tinjauan pustaka dan literatur yang digunakan dalam penyelesaian kerja praktik.

4. BAB IV Pengerjaan Kerja Praktik

Pada bab ini, dijelaskan mengenai kegiatan yang dilakukan selama kerja praktik di divisi Bidang Teknologi Informasi bagian Pengembangan Teknologi Informasi & Analisa Bisnis.

5. Kesimpulan dan Saran

Pada bab ini, dipaparkan kesimpulan yang dapat diambil dan juga saran selama pengerjaan kerja praktik.

[Halaman ini sengaja dikosongkan]

BAB II

PROFIL PERUSAHAAN

2.1. Profil PT. Pembangkitan Jawa Bali

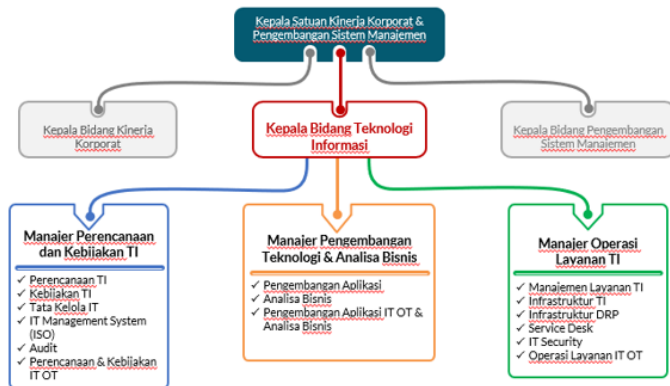
PT Pembangkitan Jawa–Bali (disingkat PT PJB) adalah sebuah anak perusahaan PLN. PT Pembangkitan Jawa-Bali (PJB) sejak berdiri tahun 1995 senantiasa mengabdikan diri untuk bangsa dan negara Indonesia, serta mendorong perkembangan perekonomian nasional dengan menyediakan energi listrik yang bermutu tinggi, andal dan ramah lingkungan. Dengan visi menjadi perusahaan pembangkit tenaga listrik Indonesia yang terkemuka dengan standar kelas dunia, PJB tiada henti berbenah dan melakukan inovasi dengan tetap berpegang pada kaidah tata pengelolaan perusahaan yang baik (Good Corporate Governance/GCG). Berkat dukungan shareholders dan stakeholders, PJB tumbuh dan berkembang dengan berbagai bidang usaha, tanpa meninggalkan tanggung jawab sosial perusahaan demi terwujudnya kemandirian masyarakat dan kelestarian lingkungan hidup. Saat ini PT PJB mengelola 6 Pembangkit Tenaga Listrik di Pulau Jawa, dengan kapasitas total 6.511 Megawatt. PT PJB juga mengelola sejumlah unit bisnis, termasuk unit pengelolaan, teknologi informasi, dan pengembangan. Kantor pusat PT PJB berada di Surabaya. PT. PJB Adalah perusahaan pertama di asia pasifik yang memiliki sertifikasi ISO 55001. PT PJB memiliki 6 (enam) unit pembangkitan (UP) yang tersebar di Jawa Timur, Jawa Barat dan DKI Jakarta, yaitu UP Gresik, UP Paiton, UP Muara Karang, UP Muara Tawar, UP Cirata dan UP Brantas. Total kapasitas terpasang mencapai 6.977 MW, yang terdiri dari Pembangkit Listrik Tenaga Uap (PLTU), Pembangkit Listrik Tenaga Gas (PLTG), Pembangkit Listrik Tenaga Gas dan Uap (PLTGU) dan Pembangkit Listrik Tenaga Air (PLTA) [1].

2.2. Profil Bidang Teknologi Informasi PT PJB

Bidang Teknologi Informasi (BTIF) adalah salah satu bidang di PT Pembangunan Jawa-Bali (PJB) yang mana memiliki visi menjadi partner strategis dalam mewujudkan PJB menjadi perusahaan terdepan dan terpercaya dalam bisnis energi berkelanjutan di Asia Tenggara. Berdasarkan SK No 0003.P/019/DIR/2021 Pasal 3 Ayat 1, Fungsi Utama BTIF adalah “Memastikan kesiapan pengelolaan Strategi dan Kebijakan Sistem Informasi Terpadu Perusahaan secara optimal dan terintegrasi dalam mendukung kinerja Perusahaan”.

A. Struktur Organisasi BTIF

Berikut merupakan struktur organisasi BTIF secara umum:



Gambar 1. Struktur Organisasi BTIF

BTIF merupakan bidang berada pada Satuan Kinerja Korporat & Pengembangan Sistem Manajemen. Di BTIF terdapat tiga bagian yaitu Perencanaan dan Kebijakan Teknologi Informasi, Pengembangan Teknologi dan Analisa Bisnis serta Operasi Layanan Teknologi Informasi. Di setiap bagian, terdapat fokus utama masing-masing yaitu:

1. Perencanaan dan Kebijakan TI, memiliki peran yaitu Perencanaan TI, kebijakan TI, tata kelola TI, *IT Management System* (ISO), Audit, serta Perencanaan dan Kebijakan IT OT.
2. Pengembangan Teknologi dan Analisa Bisnis, memiliki peran untuk melakukan pengembangan aplikasi, analisa bisnis, serta pengembangan aplikasi IT OT dan analisa bisnis.
3. Operasi Layanan Teknologi Informasi, memiliki peran untuk mengelola manajemen layanan TI, infrastruktur TI dan DRP, Service Desk, *IT Security*, dan operasi layanan IT OT.

B. Proses Bisnis BTIF

Berikut merupakan Tata Kelola yang menjadi proses bisnis BTIF berdasarkan SK NO : 067.K/010/DIR/2017 :



Gambar 2. Proses Bisnis BTIF di setiap bagian

Setiap bagian dari BTIF memiliki proses bisnis masing-masing dan saling memiliki keterikatan satu sama lain. Bagian Perencanaan dan Kebijakan TI mengelola bisnis TI; Mengelola rencana kerja tahunan dan portofolio TI;

Mengelola desain layanan TI. Lalu, bagian Pengembangan TI dan Analisa Bisnis mengelola transisi dan pengembangan layanan Teknologi Informasi. Sedangkan, bagian Operasi dan Layanan TI mengelola operasional layanan TI; Memantau kinerja TI, kepatuhan dan *Continual Service Improvement*. Adapun tujuan dari Tata Kelola TI adalah :

1. Sebagai pedoman untuk mewujudkan pola standardisasi kerangka pelaksanaan pengembangan, penerapan dan operasi Teknologi Informasi yang selaras dengan kebutuhan strategis Perusahaan.
2. Sebagai pedoman untuk memantau dan mengevaluasi unjuk kerja Penyelenggaraan Teknologi Informasi
3. Sebagai pedoman untuk meningkatkan kapabilitas Perusahaan dalam memberikan kontribusi bagi penciptaan nilai tambah serta meningkatkan efektivitas dan efisiensi kegiatan operasional.

Dalam mendukung proses bisnis tersebut, sebuah *Strategy Map* TI yang menjadi visi dalam BTIF. Hal ini tertuangkan pada Gambar 3 :



Gambar 3. Bagan Visi BTIF

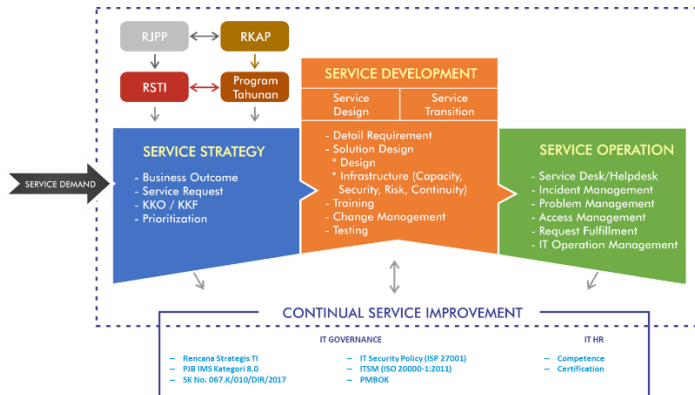
Berdasarkan gambar tersebut, Pada sisi *stakeholder*, BTIF berusaha memastikan kepuasan pengguna terhadap layanan dan tingkat layanan yang diberikan, memastikan

layanan TI yang andal, dan pemenuhan kebutuhan layanan TI. Lalu pada sisi internal proses, terdiri dari :

1. Perencanaan, Pengembangan, Operasional dan Pemantauan TI. Dimana pada Perencanaan yaitu sebagai *IT Business Alignment* dan Perencanaan TI yang efektif.
2. Pengembangan melakukan penyediaan *High Quality Information* dan Pengembangan IT yang efisien.
3. Pada operasional memiliki peran untuk melakukan pengelolaan TI secara aman dan andal
4. Pemantauan Layanan TI, yaitu melakukan pemantauan dan perbaikan secara terus menerus.

Dan pada sisi *learning & growth* BTIF berusaha meningkatkan pengetahuan dan kompetensi SDM nya, mengimplementasi tata kelola TI, RSTI, dan *Standar Management System*, serta memastikan *tools* dan teknologi yang mencukupi.

Untuk menjalankan proses bisnis dengan baik, BTIF menggunakan *Framework* Manajemen TI sehingga prosesnya lebih terstruktur. *Framework* ini dijelaskan pada Gambar 4.



Gambar 4. *Framework* Manajemen TI BTIF

Service demand merupakan suatu permintaan layanan dari *user/pengguna*. *User* merupakan bidang/divisi yang membutuhkan layanan TI. Lalu bagian Perencanaan dan Kebijakan TI akan melakukan serangkaian proses seperti memastikan apakah *service demand* sesuai dengan 4 faktor yaitu Rencana Jangka Panjang Perusahaan (RJPP), Rencana Kerja dan Anggaran Perusahaan (RKAP), Rencana Strategis Teknologi Informasi (RSTI) dan Program Tahunan BTIF. Setelah sesuai, *service demand* akan diproses dan dikalkulasi dari sisi *bussiness outcome*, *service request*, Kajian Kelayakan Operasional (KKO)/Kajian Kelayakan Finansial (KKF), prioritas layanan.

Lalu berlanjut ke bagian Pengembangan TI dan Analisa Bisnis. Di bagian ini, kebutuhan TI atas layanan tersebut diperinci, lalu layanan didesain yang menjadi solusi dari layanan, *training* terhadap layanan aplikasi, *change management* dan *testing* layanan aplikasi. Semua proses tersebut merupakan *rincian* dari proses *service design* dan *service transition*. Di dalam *service transition*, dibutuhkan bagian Operasi Layanan Teknologi Informasi.

Di bagian Operasi Layanan Teknologi Informasi, layanan yang telah “*go live*” akan *dihandle* sehingga layanan tetap berjalan sebagaimana mestinya. Terdapat beberapa proses yang digunakan pada bagian ini yaitu *service desk/helpdesk*, *incident management*, *problem management*, *access management*, *request fulfilment* dan *IT Operation Management*.

BAB III

TINJAUAN PUSTAKA

Pada bab ini, akan dijelaskan mengenai dasar teori yang digunakan selama proses pengembangan aplikasi web di PT Pembangunan Jawa Bali.

3.1 HTML

HTML merupakan singkatan dari *Hypertext Markup Language*, dimana merupakan *standard markup language* untuk pembuatan sebuah pemrograman web. HTML biasanya mendeskripsikan struktur sebuah halaman web, dimana terdiri dari beberapa elemen tampilan web. Elemen dalam HTML tersebut yang akan berkomunikasi dengan *browser* berkaitan dengan data yang akan ditampilkan dalam sebuah *website* [2].

3.2 PHP

PHP adalah sebuah bahasa pemrograman di sisi *server* (*server side scripting*) yang bersifat *open source*. Sebagai sebuah *scripting language*, PHP menjalankan instruksi pemrograman saat proses *runtime* [3]. Hasil dari instruksi tentu akan berbeda tergantung data yang diproses. PHP merupakan bahasa pemrograman *server-side*, sehingga *script* dari PHP nantinya akan diproses di sisi *server*.

Keunggulan menggunakan bahasa pemrograman PHP adalah bahasa PHP merupakan bahasa yang bersifat *open source* serta mempunyai banyak *library*, sehingga dapat memudahkan pengembangan dan mempercepat waktu pengembangan aplikasi.

3.3 Framework Codeigniter

CodeIgniter adalah sebuah *web application network* yang bersifat *open source* yang digunakan untuk membangun aplikasi php dinamis [4].

CodeIgniter menjadi sebuah *framework* PHP dengan model MVC (*Model, View, Controller*) untuk membangun website dinamis

dengan menggunakan PHP yang dapat mempercepat pengembang untuk membuat sebuah aplikasi web. Selain ringan dan cepat, CodeIgniter juga memiliki dokumentasi yang super lengkap disertai dengan contoh implementasi kodenya. Dokumentasi yang lengkap inilah yang menjadi salah satu alasan kuat mengapa banyak orang memilih CodeIgniter sebagai *framework* pilihannya. Karena kelebihan-kelebihan yang dimiliki oleh CodeIgniter, pembuat PHP Rasmus Lerdorf memuji CodeIgniter di frOSCon (Agustus 2008) dengan mengatakan bahwa dia menyukai CodeIgniter karena “*it is faster, lighter and the least like a framework.*” [4]

CodeIgniter pertamakali dikembangkan pada tahun 2006 oleh Rick Ellis. Dengan logo api yang menyala, CodeIgniter dengan cepat “membakar” semangat para web *developer* untuk mengembangkan web dinamis dengan cepat dan mudah menggunakan *framework* PHP yang satu ini [4].

3.4 REST API

REST (*Representational State Transfer*) adalah *web service* yang bersifat *stateless*, jadi setiap kali *request* harus menyertakan semua data dan parameter dengan lengkap. REST ini bersifat *client* dan *server*. *Client* REST akan meminta sesuatu ke REST *server*, REST *server* kemudian akan memberikan *response*, *client* REST ini kemudian akan menampilkan hasilnya atau melakukan pemrosesan yang lain [5]. Pada kasus ini, penulis akan menggunakan konsep ini untuk menghubungkan antara aplikasi *frontend* BTIF Monitoring dengan aplikasi *backend* BTIF Monitoring.

3.5 MySQL

MySQL adalah *open source relational database management system* (RDBMS) yang berbasis *Structured Query Language* (SQL). MySQL juga merupakan *Database management system* (DBMS) multi pengguna dan multi alur. Dengan MySQL, pengaturan database dilakukan dengan menggunakan *Structured Query Language* (SQL) dengan basis *client-server model* [6].

3.6 Postman

Postman merupakan sebuah platform yang digunakan dalam pengembangan API. Postman bertujuan untuk menyederhanakan langkah pengembangan API sehingga mempercepat waktu pengembangan aplikasinya [7].

3.7 Container

Container merupakan suatu paket atau kemasan yang berisi aplikasi beserta dependensinya seperti *library*. Dengan adanya *container* ini aplikasi bisa berjalan di atas sistem operasi kernel tanpa memerlukan *Virtual Machine*. Dalam implementasinya *container* berkaitan erat dengan *tools* yang bernama Docker [8].

3.8 Docker

Docker adalah sebuah *tools* aplikasi yang digunakan untuk memudahkan dalam membuat, mendeploy, dan menjalankan aplikasi dengan menggunakan sebuah *container*. Docker sendiri berfungsi untuk menyatukan *file-file* yang dibutuhkan untuk membangun sebuah *image* docker. Docker juga digunakan sebagai alat yang digunakan untuk menjalankan *image* yang selanjutnya akan terbentuk menjadi *container* [9].

3.9 Docker Compose

Docker Compose merupakan suatu alat yang digunakan untuk melakukan konfigurasi suatu *image* docker yang selanjutnya dijalankan untuk membentuk sebuah *container* [10]. Dengan menggunakan Docker Compose kita bisa melakukan konfigurasi *port*, *enviromtent*, *networking*, nama *container*, *image* docker yang digunakan, dll.

3.10 Microservice

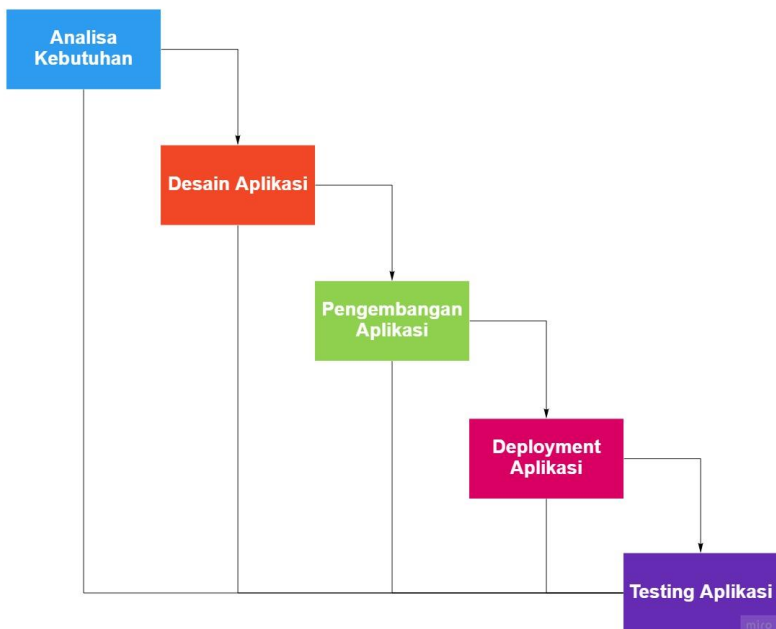
Microservice adalah arsitektur yang menekankan kepada banyak *service* atau aplikasi yang tidak terikat erat satu dengan lainnya (*loosely coupled*). Setiap *service* memiliki tujuan yang sangat terfokus dan memiliki metode komunikasi antar *service* yang

tidak berat. Arsitektur ini berkaitan erat dengan teknologi *Container* [11].

BAB IV

PENGERJAAN KERJA PRAKTIK

Pengerjaan kerja praktik penulis akan menggunakan konsep *Software Development Life Cycle*, dimana konsep ini akan menjelaskan tahapan pembuatan dari suatu aplikasi. Konsep SDLC yang akan penulis terapkan akan disesuaikan dengan metode pelaksanaannya, yaitu metode *waterfall*. Dimana metode ini akan membuat sebuah aplikasi secara berurutan, mulai dari menganalisa kebutuhan hingga melakukan *deployment* aplikasi.



Gambar 5. Tahapan SDLC

Penjelasan mengenai pembuatan aplikasi menggunakan tahapan seperti pada gambar di atas adalah sebagai berikut.

4.1. Analisis Kebutuhan Pengguna

Tahapan awal dalam pembuatan aplikasi BTIF Monitoring adalah melakukan analisa kebutuhan. Analisa ini dapat dilakukan berdasarkan hasil diskusi yang telah dilakukan dengan klien dari aplikasi BTIF Monitoring. Berdasarkan diskusi dengan klien, penulis mendapatkan beberapa kebutuhan pada aplikasi BTIF Monitoring.

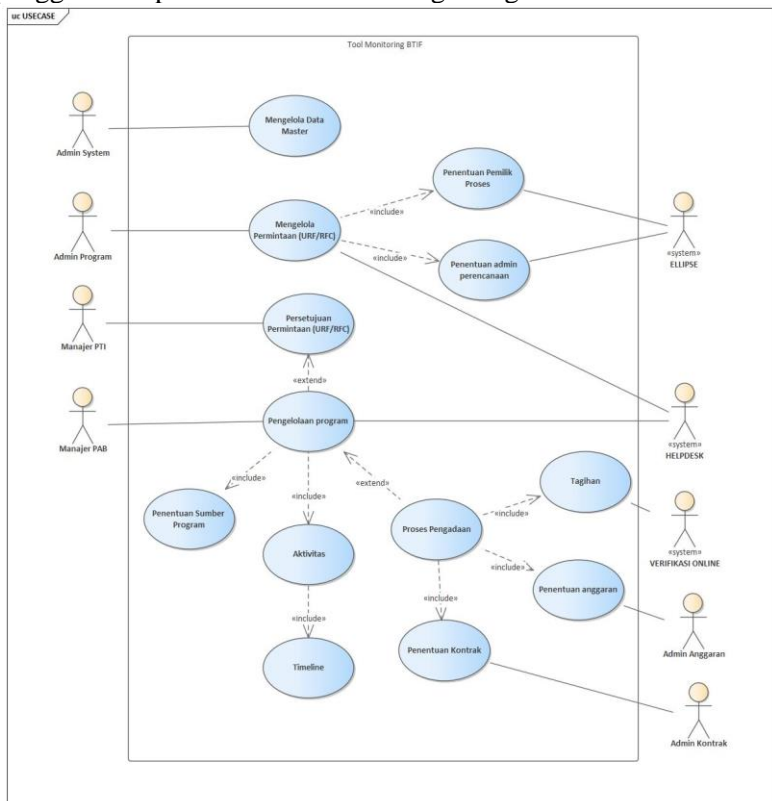
Klien meminta sebuah aplikasi yang dapat melakukan *monitoring* program kerja pada suatu divisi. Selain melakukan *monitoring*, aplikasi juga dapat mengelola permintaan untuk dapat dijalankan menjadi program kerja. Aplikasi juga dapat terintegrasi dengan sistem lain yang sudah ada, yaitu sistem Helpdesk, dimana sistem tersebut juga merupakan sistem pengelolaan permintaan, namun dengan cakupan yang lebih luas. Klien juga meminta aplikasi mempunyai fungsi autentikasi yang terintegrasi dengan sistem LDAP, dimana merupakan sistem autentikasi yang sudah ada pada PT PJB. Klien juga menyatakan akan terdapat tiga aktor pada pengelolaan *monitoring* program, yaitu Manajer PAB, Manajer PTI dan Admin.

Penjelasan terkait permintaan klien tersebut, akan dianalisis terkait kebutuhan apa saja yang dibutuhkan dalam pembuatan aplikasi BTIF Monitoring. Pada aplikasi BTIF Monitoring, terdapat dua aktor, yaitu Admin dan Manajer PTI. Penjelasan apa saja yang dapat dilakukan oleh dua aktor tersebut adalah sebagai berikut.

- 1) *User* sebagai admin sistem dapat mengelola data *master*, dimana data *master* adalah data yang mendukung data utama, yaitu data permintaan dan data program.
- 2) *User* sebagai admin program dapat mengelola permintaan, dimana permintaan yang dikelola sudah diintegrasikan dengan sistem helpdesk.

- 3) *User* sebagai manajer PTI dapat melakukan *change* permintaan, dimana dapat mengubah permintaan menjadi program kerja
- 4) *User* sebagai manajer PAB dapat melakukan pengelolaan program kerja, dimana dapat menentukan sumber program, melihat aktivitas dan *timeline* program, dan melakukan proses pengadaan.

Berdasarkan penjelasan tersebut, dibuat sebuah diagram kasus penggunaan aplikasi BTIF Monitoring sebagai berikut.



Gambar 6. Kasus Penggunaan Aplikasi BTIF Monitoring

4.2. Desain Aplikasi

Setelah melakukan analisa kebutuhan dari aplikasi yang dibuat dan memvisualisasikan hasilnya menjadi sebuah diagram kasus penggunaan, selanjutnya penulis akan membuat protipe dari antarmuka aplikasinya. Prototipe antarmuka aplikasi yang akan dibuat penulis adalah sebagai berikut:

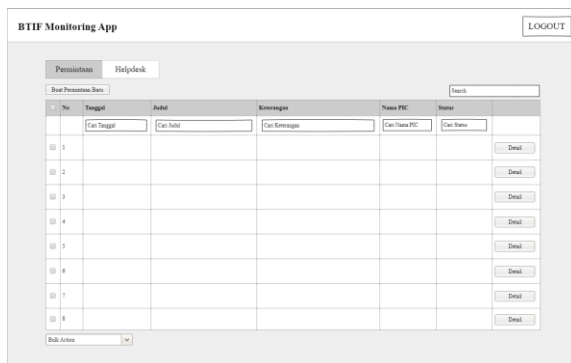


LOGIN PJB

Email

Password

Gambar 7. Halaman *Login* Aplikasi BTIF Monitoring



BTIF Monitoring App

Permintaan Helpdesk

Dftar Permintaan Baru Search

No	Tanggal	Judul	Keterangan	Nama PIC	Status	
1	Cat Tanggal	Cat Judul	Cat Keterangan	Cat Nama PIC	Cat Status	<input type="button" value="Detail"/>
2						<input type="button" value="Detail"/>
3						<input type="button" value="Detail"/>
4						<input type="button" value="Detail"/>
5						<input type="button" value="Detail"/>
6						<input type="button" value="Detail"/>
7						<input type="button" value="Detail"/>
8						<input type="button" value="Detail"/>

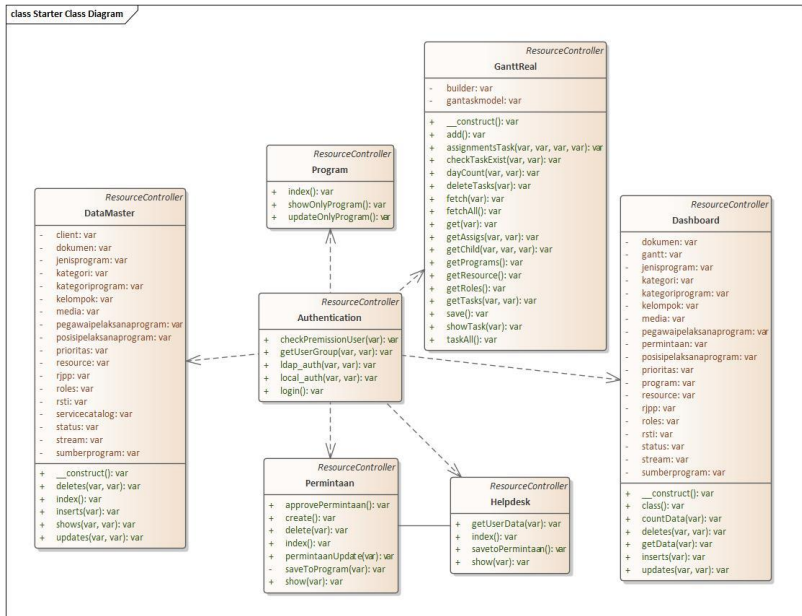
Dibuat Admin

Gambar 8. Halaman List Permintaan Aplikasi BTIF Monitoring

Gambar 9. Halaman Buat Permintaan Aplikasi BTIF Monitoring

Gambar 10. Halaman Detail Permintaan Aplikasi BTIF Monitoring

Setelah membuat desain prototipe dari aplikasinya, selanjutnya penulis melakukan desain dari basis data yang akan dibuat. Desain tersebut akan divisualisasikan dalam sebuah *Conceptual Data Model* (CDM). CDM dari aplikasi BTIF Monitoring adalah sebagai berikut.



Gambar 13. Class Diagram Backend

Berdasarkan *class diagram* diatas, bahwasannya aplikasi yang akan penulis buat terdiri dua aplikasi, yaitu aplikasi yang berkaitan dengan pengelolaan *backend*-nya dan aplikasi yang berkaitan dengan *frontend*-nya. Pengembangan ini untuk mengimplementasikan konsep dasar *microservice*, namun karena penulis mempunyai instance yang terbatas, maka implementasi ini dilakukan sesederhana mungkin, dengan tetap menggunakan konsep dan teknologi dari pengembangan *microservice*.

4.3. Pengembangan Aplikasi

Proses implementasi dari aplikasi yang penulis buat akan menggunakan konsep dasar dari *microservice*. Dimana konsep *microservice* adalah sebuah konsep yang membagi satu aplikasi menjadi beberapa service. Konsep tersebut penulis aplikasikan, dan membagi aplikasi yang dibuat menjadi dua aplikasi, yaitu aplikasi

yang berkaitan dengan *backend*, dimana akan berisi konfigurasi yang terhubung basis datanya, dan dapat diakses melalui sebuah API. Aplikasi kedua berkaitan dengan *frontend*, dimana berkaitan dengan *user interface* dari aplikasinya, data yang ditampilkan adalah data yang diambil dari aplikasi backendnya melalui sebuah API.

Dalam melakukan implementasi aplikasi *frontend*, penulis mencoba menggunakan konsep *component-based architecture*, karena penulis melakukan analisis dari prototipe yang telah dibuat. Dari prototipe, banyak sekali repetitif yang dilakukan di tiap halaman, hanya berbeda beberapa *property* saja dan sedikit penambahan *method*, untuk sebagian besar hampir sama. Berdasarkan hal tersebut, maka implementasi *frontend* yang dilakukan penulis adalah membuat beberapa component, seperti *form*, *modal*, *table*, *header*, *footer*, *navigation bar*, dan sebagainya. Nantinya component tersebut akan saling terkoneksi pada *controller framework*-nya, dimana nantinya dapat di-*reuseable* dan mempercepat pembuatan sebuah *frontend* dari aplikasi.

Implementasi *backend* sendiri, penulis akan membuat beberapa API, dimana nantinya API ini akan diakses oleh aplikasi *frontend* untuk ditampilkan kepada pengguna. Konsep yang penulis gunakan dalam membuat aplikasi *backend* adalah dengan menggunakan konsep MVC, namun hanya mengimplementasikan *Model* dan *Controller*, karena bagian *View* sudah diaplikasikan pada aplikasi *frontend* yang dibuat. Penulis akan menghubungkan aplikasi *backend* dengan basis datanya melalui *Model*, lalu akan mengolah data tersebut pada *Controller*. Nantinya data yang diolah tersebut akan menjadi respon dari API yang dibuat pada aplikasi *backend*-nya. Berkaitan dengan API yang dibuat, berikut adalah daftar API yang dibuat dan diimplementasikan pada aplikasinya.

API		Status	URL API (dev)	method	Params
Authentication	-login	OK	http://192.168.18.33:8888/login	POST	NID, password
Permintaan	-show all	OK	http://192.168.18.33:8888/permintaan	GET	-
	-show per id	OK	http://192.168.18.33:8888/permintaan	GET	ID
	- create	OK	http://192.168.18.33:8888/permintaan	POST	ID_Jabatan, ID_Distrik, ID_Divbid, ID_Subdivsbbid, ID_Media, Tanggal, Judul, Keterangan, Link_URF, PIC_Perencanaan
	- update per id	OK	http://192.168.18.33:8888/permintaan	PUT	ID, ID_Jabatan, ID_Distrik, ID_Divbid, ID_Subdivsbbid, ID_Media, Tanggal, Judul, Keterangan, Link_URF, PIC_Perencanaan
	- delete per id	OK	http://192.168.18.33:8888/permintaan	DELETE	ID
	- delete kolektif	OK	http://192.168.18.33:8888/permintaan	DELETE	Array of Object
	- change permintaan	OK	http://192.168.18.33:8888/permintaan	POST	Array of Object (id permintaan)
Program	-show all	OK	http://192.168.18.33:8888/program	GET	-
	-show per id	OK	http://192.168.18.33:8888/program	GET	ID
	- update per id	OK	http://192.168.18.33:8888/program	POST	ID, Data program yang diupdate
Helpdesk	-show all	OK	http://192.168.18.33:8888/helpdesk	GET	-
	-show per id	OK	http://192.168.18.33:8888/helpdesk	GET	ID
	-helpdesk to permintaan	OK	http://192.168.18.33:8888/helpdesk	POST	User ID, Array of Object (Object Permintaan)
Gantt	- Show gantt per id	OK	http://192.168.18.33:8888/ganttreal/fetch	GET	ID
	- Show tasks per id	OK	http://192.168.18.33:8888/ganttreal/task	GET	ID
	- Save	OK	http://192.168.18.33:8888/ganttReal/save	POST	Array of Object
Data Master	- show class	OK	http://192.168.18.33:8888/list/class	GET	-
	- show detail of 1 class	OK	http://192.168.18.33:8888/list/find	GET	nama data master
	- show detail from 1 id of a class	OK	http://192.168.18.33:8888/list/find	GET	nama data master, ID data master
	- insert data to a class	OK	http://192.168.18.33:8888/list/insert	POST	nama data master
	- update data to a class	OK	http://192.168.18.33:8888/list/update	PUT	nama data master, ID data master yang ingin diupdate
	- delete data to a class	OK	http://192.168.18.33:8888/list/delete	DELETE	nama data master, ID data master yang ingin didelete

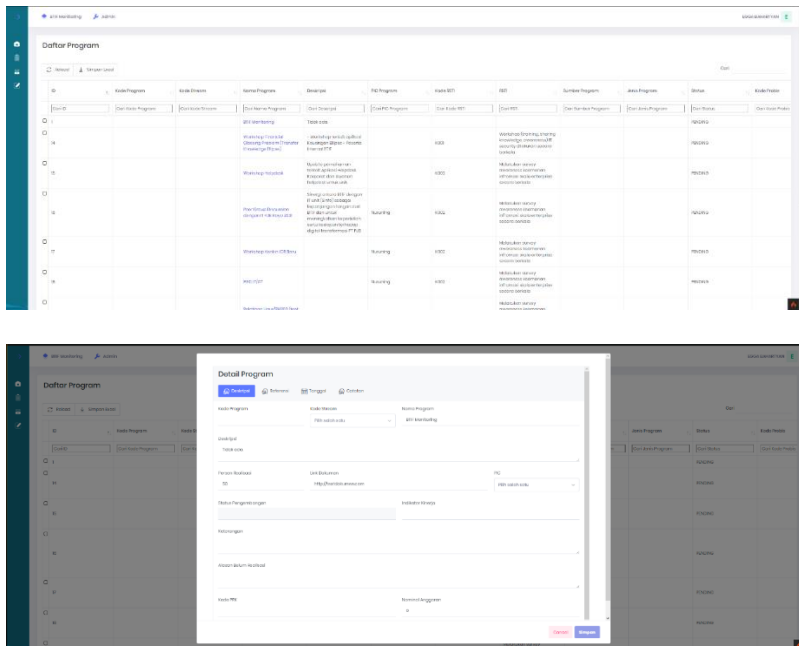
Gambar 14. Dokumentasi API aplikasi BTIF Monitoring

Berdasarkan gambar dokumentasi diatas, terdapat beberapa modul di aplikasi yang akan penulis buat. Tiap modul tersebut nantinya akan mempunyai fungsionalitas yang berbeda-beda, berkaitan dengan data dari aplikasinya. Penjelasan fungsionalitas tiap modul adalah sebagai berikut:

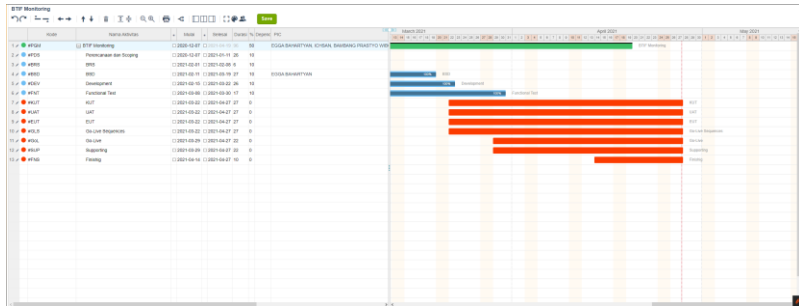
1. *Authentication*: modul yang menangani autentikasi pada aplikasi yang penulis buat.
2. *Permintaan*: modul yang menangani pengelolaan data permintaan, berkaitan dengan fungsi menambah permintaan baru, melihat permintaan, mengubah satu permintaan, menghapus permintaan, dan melakukan konversi dari permintaan menjadi program kerja.
3. *Helpdesk*: modul yang menangani pengelolaan data yang didapatkan dari sistem lain, yaitu sistem helpdesk. Pada modul ini akan berkaitan dengan fungsi melihat data helpdesk dan konversi dari helpdesk menjadi sebuah permintaan.
4. *Program*: modul yang menangani pengelolaan data program, berkaitan dengan fungsi melihat program dan mengubah data satu program, dimana perubahan ini juga berkaitan dengan penambahan catatan pada satu program kerja.
5. *Gantt Chart*: modul yang menangani perubahan progress dari satu program, dimana berkaitan dengan judul program, tanggal rencana mulai dan tanggal rencana selesai, tanggal realisasi mulai dan tanggal realisasi selesai, deskripsi, dan fungsionalitas pengelolaan aktivitas pada satu program. Aktivitas adalah kegiatan yang dijalankan pada satu program kerja, dimana tiap aktivitas akan mempunyai tanggal mulai dan tanggal selesai yang berbeda dengan program kerja induknya, namun berada dalam cakupan tanggal mulai dan tanggal selesai program kerja induknya.
6. *Data master*: modul yang menangani pengelolaan data *master* dari aplikasinya, dimana pengelolaan data ini berkaitan dengan melihat, menambah, mengubah, dan menghapus data *master* aplikasi. *Data master* dari aplikasi yang penulis buat ada banyak, dimana merupakan data yang mendukung data utama dari

aplikasinya, yaitu data permintaan dan data program kerja. Data *master* di aplikasi yang penulis buat diantaranya adalah *client*, dokumen, jenis program, kategori, kategori program, kelompok, media, pegawai pelaksana program, *roles*, posisi pelaksana program, prioritas, RSTI, *resource*, RJPP, service katalog, status, *stream*, dan sumber program.

Hasil implementasi pada aplikasi *frontend* BTIF Monitoring adalah sebagai berikut.



Gambar 15. Tampilan Program dan detail program



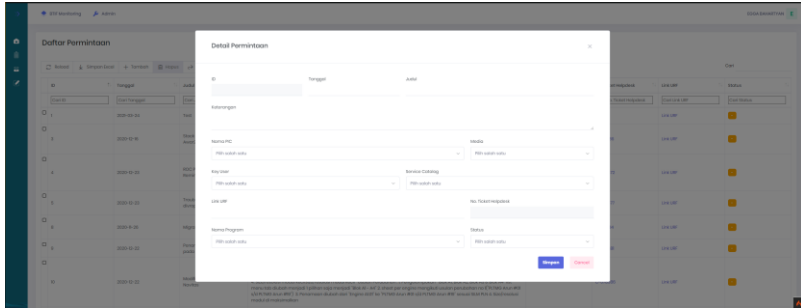
Gambar 16. Tampilan ganttchart untuk monitoring program kerja

ID	Judul	Tipe	Organisasi	Start Date	End Date	Status	Agent	Last Update	Closed Date	Creation Date	Reporter
C-07588	Membuka aplikasi	Normal Change	Kantor/Kantor Subdit	2023-02-28 09:07	2023-02-28 09:07	Penyelesaian	Agus	2023-02-28 09:07	2023-02-28 09:07	2023-02-28 09:07	Agus
C-07589	Membuka dan Penutupan aplikasi	Normal Change	Kantor/Kantor Subdit	2023-02-28 09:07	2023-02-28 09:07	Penyelesaian	Agus	2023-02-28 09:07	2023-02-28 09:07	2023-02-28 09:07	Agus
C-07590	Tidak bisa login	Normal Change	Kantor/Kantor Subdit	2023-02-28 09:07	2023-02-28 09:07	Penyelesaian	Agus	2023-02-28 09:07	2023-02-28 09:07	2023-02-28 09:07	Agus

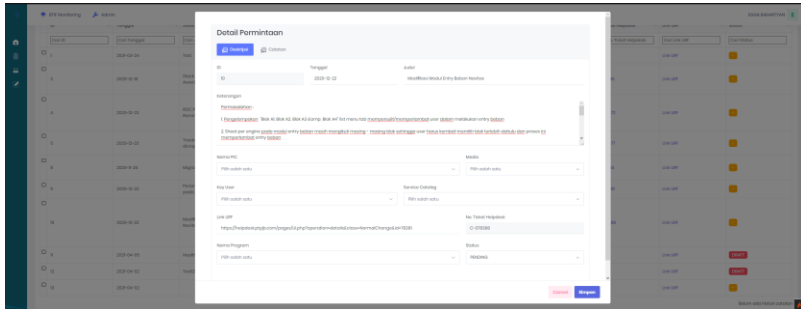
Gambar 17. Tampilan helpdesk

ID	Requester	Judul	Deskripsi	Status	No. Ticket Requested	No. Ticket Resolved	Status
1	Agus	Tidak bisa login	Tidak bisa login ke aplikasi	Penyelesaian	07588	07588	Penyelesaian
2	Agus	Tidak bisa akses aplikasi	Tidak bisa akses aplikasi	Penyelesaian	07589	07589	Penyelesaian
3	Agus	Tidak bisa akses aplikasi	Tidak bisa akses aplikasi	Penyelesaian	07590	07590	Penyelesaian

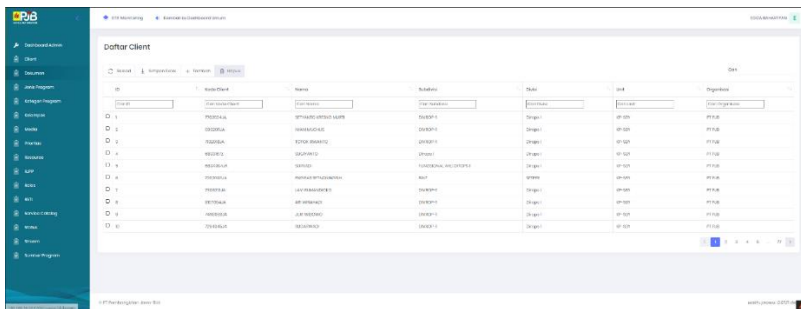
Gambar 18. Tampilan permintaan



Gambar 19. Tampilan buat permintaan baru



Gambar 20. Tampilan detail permintaan



Gambar 21. Tampilan data master client

Kategori Dokumen	Nama Dokumen	Nama	Status Dokumen
01	Kategori Dokumen	Kategori Dokumen	✓
02	Kategori Dokumen	Kategori Dokumen	✓
03	Kategori Dokumen	Kategori Dokumen	✓
04	Kategori Dokumen	Kategori Dokumen	✓
05	Kategori Dokumen	Kategori Dokumen	✓
06	Kategori Dokumen	Kategori Dokumen	✓
07	Kategori Dokumen	Kategori Dokumen	✓
08	Kategori Dokumen	Kategori Dokumen	✓
09	Kategori Dokumen	Kategori Dokumen	✓
10	Kategori Dokumen	Kategori Dokumen	✓
11	Kategori Dokumen	Kategori Dokumen	✓
12	Kategori Dokumen	Kategori Dokumen	✓
13	Kategori Dokumen	Kategori Dokumen	✓
14	Kategori Dokumen	Kategori Dokumen	✓
15	Kategori Dokumen	Kategori Dokumen	✓
16	Kategori Dokumen	Kategori Dokumen	✓
17	Kategori Dokumen	Kategori Dokumen	✓
18	Kategori Dokumen	Kategori Dokumen	✓
19	Kategori Dokumen	Kategori Dokumen	✓
20	Kategori Dokumen	Kategori Dokumen	✓

Gambar 22. Tampilan data *master* dokumen

Kategori Program	Nama Program	Nama	Kategori Program	Status Program	Status Dokumen
01	Kategori Program	Kategori Program	Kategori Program	✓	✓
02	Kategori Program	Kategori Program	Kategori Program	✓	✓
03	Kategori Program	Kategori Program	Kategori Program	✓	✓

Gambar 23. Tampilan data *master* jenis program

Kategori Program	Kategori Program	Kategori Program	Nama	Nama	Kategori Program	Status Program	Status Dokumen	Status Dokumen
01	01	01	Kategori Program	Kategori Program	Kategori Program	✓	✓	✓
02	02	02	Kategori Program	Kategori Program	Kategori Program	✓	✓	✓
03	03	03	Kategori Program	Kategori Program	Kategori Program	✓	✓	✓
04	04	04	Kategori Program	Kategori Program	Kategori Program	✓	✓	✓
05	05	05	Kategori Program	Kategori Program	Kategori Program	✓	✓	✓
06	06	06	Kategori Program	Kategori Program	Kategori Program	✓	✓	✓
07	07	07	Kategori Program	Kategori Program	Kategori Program	✓	✓	✓
08	08	08	Kategori Program	Kategori Program	Kategori Program	✓	✓	✓
09	09	09	Kategori Program	Kategori Program	Kategori Program	✓	✓	✓
10	10	10	Kategori Program	Kategori Program	Kategori Program	✓	✓	✓
11	11	11	Kategori Program	Kategori Program	Kategori Program	✓	✓	✓
12	12	12	Kategori Program	Kategori Program	Kategori Program	✓	✓	✓
13	13	13	Kategori Program	Kategori Program	Kategori Program	✓	✓	✓
14	14	14	Kategori Program	Kategori Program	Kategori Program	✓	✓	✓
15	15	15	Kategori Program	Kategori Program	Kategori Program	✓	✓	✓

Gambar 24. Tampilan data *master* kategori program

id	nama	status_loker
1	WISATA BUNDAWATI	1
2	WISATA BUNDAWATI	1
3	WISATA BUNDAWATI	1
4	WISATA BUNDAWATI	1

Gambar 25. Tampilan data *master* kelompok

id	nama	status_loker
1	WISATA BUNDAWATI	1
2	WISATA BUNDAWATI	1
3	WISATA BUNDAWATI	1
4	WISATA BUNDAWATI	1

Gambar 26. Tampilan data *master* media

id	nama	status_loker
1	WISATA BUNDAWATI	1
2	WISATA BUNDAWATI	1
3	WISATA BUNDAWATI	1

Gambar 27. Tampilan data *master* prioritas

ID	Nama	Deskripsi	Status	Kelembagaan	Kelembagaan	Kelembagaan	Kelembagaan	Kelembagaan
01	01	01	01	01	01	01	01	01
02	02	02	02	02	02	02	02	02
03	03	03	03	03	03	03	03	03
04	04	04	04	04	04	04	04	04
05	05	05	05	05	05	05	05	05
06	06	06	06	06	06	06	06	06
07	07	07	07	07	07	07	07	07
08	08	08	08	08	08	08	08	08
09	09	09	09	09	09	09	09	09
10	10	10	10	10	10	10	10	10

Gambar 31. Tampilan data master RSTI

ID	Nama	Deskripsi
01	01	01
02	02	02
03	03	03
04	04	04
05	05	05
06	06	06
07	07	07
08	08	08
09	09	09
10	10	10

Gambar 32. Tampilan data master service catalog

ID	Nama	Deskripsi
01	01	01
02	02	02
03	03	03
04	04	04
05	05	05
06	06	06
07	07	07
08	08	08
09	09	09
10	10	10

Gambar 33. Tampilan data master status

Stream ID	Stream Name	Stream Description
D 1	Kantor Pusat	Pusat Kegiatan, Komunikasi, Koordinasi, dan Administrasi
D 2	Kantor Cabang	Operasi Administrasi, Engineering, Safety, dan Environment
D 3	Kantor Perwakilan	Keuangan, Finance, Budgeting, Accounting, Legal, and HRD
D 4	Kantor Perwakilan	Operasi Program Development
D 5	Kantor Perwakilan	Operasi Marketing dan Promosi/Hubungan Masyarakat
D 6	Kantor Perwakilan	Data Management, Warehouse, Reporting, Dashboard, Security, and Risk Management

Gambar 34. Tampilan data *master stream*

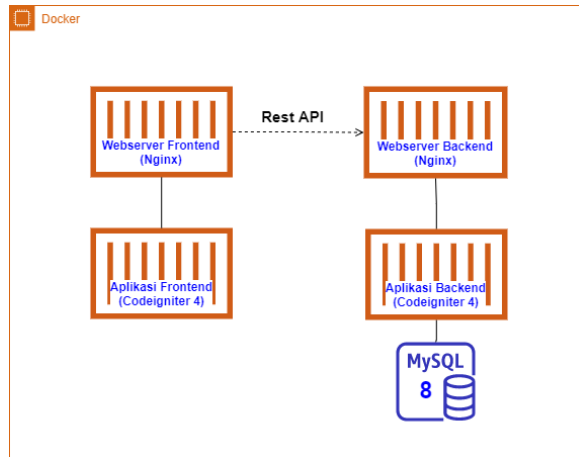
Sumber Program	Sumber	Subsumber	Sumber Aspek
D 1	Kantor Pusat	Kantor Pusat	Kantor Pusat
D 2	Kantor Cabang	Kantor Cabang	Kantor Cabang
D 3	Kantor Perwakilan	Kantor Perwakilan	Kantor Perwakilan
D 4	Kantor Perwakilan	Kantor Perwakilan	Kantor Perwakilan
D 5	Kantor Perwakilan	Kantor Perwakilan	Kantor Perwakilan
D 6	Kantor Perwakilan	Kantor Perwakilan	Kantor Perwakilan

Gambar 35. Tampilan data *master sumber program*

4.4. Deployment Aplikasi

Setelah pengembangan aplikasi diselesaikan dan semua fungsionalitas dari aplikasi berjalan, maka selanjutnya dilakukan tahap *deployment* aplikasi ke dalam *development server (virtual machine)*, dalam tahapan disini penulis menggunakan ubuntu server versi 20.04, dengan spesifikasi 12 Core CPU, 16 GB, dan 70 GB penyimpanan.

yang disediakan aplikasi *Backend*. Gambar berikut merupakan ilustrasi *deployment* containernya.



Gambar 13. Ilustrasi container.

Untuk membuat container tersebut (Gambar 13), penulis mengerjakannya dengan memanfaatkan Docker Compose. Karena dalam pengerjaannya dibagi menjadi dua bagian, maka terdapat dua file Docker Compose, yang pertama file *docker-compose.yml* untuk *frontend* dan yang kedua file *dokcer-compose.yml* untuk *backend*. Penjelasan sebagai berikut.

1. Docker Compose *Frontend*.

```
1  version: '3.8'
2
3  services:
4    frontend-app:
5      build:
6        context: ../
7        dockerfile: ../docker/php/Dockerfile
8      working_dir: /var/www/
9      container_name: frontend-app
10     volumes:
11       - ../www:/var/www
12     networks:
13       - btif-monitor-network
14
15     frontend-webserver:
16       image: nginx:latest
17       container_name: frontend-webserver
18       volumes:
19         - ../docker/nginx/nginx.conf:/etc/nginx/conf.d/default.conf
20       depends_on:
21         - frontend-app
22       ports:
23         - 1200:80
24       networks:
25         - btif-monitor-network
26
27     networks:
28       btif-monitor-network:
29         driver: bridge
30         external: true
```

Gambar 14. *docker-compose.yml frontend.*

2. Docker Compose *Backend.*


```

3 services:
4   backend-db:
5     image: mysql:8.0
6     container_name: backend-db
7     restart: unless-stopped
8     environment:
9       MYSQL_DATABASE: btif-monitoring
10      MYSQL_USER: root
11      MYSQL_PASSWORD: root
12      MYSQL_ROOT_PASSWORD: root
13     volumes:
14       - ./docker/mysql/data:/var/lib/mysql/
15       - ./docker/mysql/logs:/var/log/mysql/
16     ports:
17       - 3307:3306
18     networks:
19       - btif-monitor-network
20
21   backend-app:
22     build:
23       context: ./
24       dockerfile: ./docker/php/Dockerfile
25     working_dir: /var/www/
26     container_name: backend-app
27     volumes:
28       - ./btif-monitor-backend:/var/www
29     depends_on:
30       - backend-db
31     networks:
32       - btif-monitor-network
33
34   backend-webserver:
35     image: nginx:latest
36     container_name: backend-webserver
37     volumes:
38       - ./docker/nginx/nginx.conf:/etc/nginx/conf.d/default.conf
39       - ./btif-monitor-backend:/var/www/
40     depends_on:
41       - backend-app
42       - backend-db
43     ports:
44       - 8888:80
45     networks:
46       - btif-monitor-network
47
48 networks:
49   btif-monitor-network:
50     driver: bridge
51     external: true

```

Gambar 15. *docker-compose.yml backend.*

Penggunaan teknologi container untuk pengembangan aplikasi dilingkungan divisi Bidang Teknologi Informasi PT Pembangunan Jawa Bali sifatnya masih uji coba, dan belum pernah diimplementasikan sebelumnya. Beberapa alasan yang menjadi latar belakang *deployment* BTIF Monitoring menggunakan teknologi *container*.

1. Efisiensi *Resource*

Tabel 1. Daftar *Virtual Machine* yang tidak efisien

Nama Aplikasi	Alokasi penyimpanan	Penyimpanan digunakan	Penyimpanan tersisa
Presensi PT.PJB	87 GB	33 GB	54 GB
SINATRA	50 GB	18 GB	30 GB
OA(Office Automation)	550 GB	21 GB	529 GB
MyPJB	87 GB	33 GB	54 GB

BTIF selaku pengelola layanan aplikasi, mengelola lebih dari 50 aplikasi yang digunakan untuk menunjang proses bisnis di PT.PJB. Dimana rata-rata setiap aplikasi memiliki satu *Virtual Machine* tersendiri dan minimal penyimpanan yang diberikan adalah 50 GB, karena menurut data yang penulis terima rata-rata penyimpanan yang dialokasikan terlalu banyak dibandingkan dengan penyimpanan yang digunakan, penyimpanan yang tersisa atau tidak terpakai kurang lebih sebesar 70 persen. Hal ini sangatlah tidak efisien untuk *resource* karena satu aplikasi menggunakan satu IP Address / *Virtual Machine* sendiri. Tabel diatas merupakan beberapa aplikasi yang dikembangkan BTIF.

2. Arsitektur *Monolith*

Rata-rata aplikasi yang dikembangkan BTIF masih menggunakan arsitektur *monolith* dimana aplikasi

dikembangkan dalam satu sistem besar didalam satu tempat. Hal ini sangat tidak baik untuk pengembangan kedepannya karena semakin berkembang aplikasi tersebut maka kompleksitasnya akan bertambah pula dengan seiring waktu, sehingga menyebabkan akan sangat sulit dipahami oleh developer baru. Setiap ada perubahan harus melakukan *deployment* keseluruhan sistem yang menjadikan aplikasi harus *down* sepenuhnya dengan jangka waktu yang lumayan lama. Apabila server *down* ataupun terjadi kegagalan aplikasi maka proses *recovery*-nya akan membutuhkan waktu yang lama. Sedikit contoh aplikasi yang dikembangkan menggunakan arsitektur *monolith*.

Tabel 2. Aplikasi dengan arsitektur *Monolith*

Nama Aplikasi	Alamat Web
Web Kehadiran	hadir.ptpjb.com
Employee Self Service	amio.ptpjb.com

3. Proses *Deployment* ke Environment baru

Pada pengembanagn aplikasi di BTIF, proses *deployment* harus melakukan setup environment terlebih dahulu ke dalam Sistem Operasi *virtual machine* dan harus konfigurasi secara manual. Proses *setup environment* akan membutuhkan waktu belum lagi kalau terjadi eror pada proses konfigurasi. Apabila ingin memindahkan aplikasi ke environment baru juga akan memakan waktu karena harus *setup* ulang *environment deployment*. Contohnya pada *deployment* aplikasi pada table diatas, membutuhkan beberapa komponen untuk diinstal langsung ke *virtual machine* terlebih dahulu, yaitu perlu melakukan instalasi

webservice Apache, PHP, dan Mysql. Selanjutnya perlu mengkonfigurasinya terlebih dahulu secara manual sebelum digunakan untuk mendeploy aplikasinya.

4.5. Testing Aplikasi

Pada bagian *testing*, nantinya penulis akan mencoba semua kasus penggunaan yang terdapat di aplikasi. Berikut adalah kasus penggunaan yang akan dijelaskan melalui sebuah tabel sebagai berikut.

4.5.1. Modul Authentication

Tabel 3. Spesifikasi pengujian autentikasi aplikasi

Autentikasi aplikasi	
Kode	UJ-001
Skenario	Pengguna mengakses aplikasi melalui sebuah web browser, dan akan diarahkan ke halaman <i>login</i>
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna mengakses aplikasi melalui sebuah url 2. Pengguna mempunyai akun yang terdaftar untuk masuk ke aplikasi
Hasil yang diharapkan	Pengguna dapat melakukan autentikasi dan dapat mengakses aplikasi sesuai <i>role</i> yang dipunyai
Langkah pengujian	<ol style="list-style-type: none"> 1. Mengakses aplikasi melalui sebuah url dengan menggunakan web browser 2. Mengisikan data-data yang diperlukan pada halaman

	<i>login</i> aplikasi, yaitu data NID dan password 3. Menekan tombol <i>login</i>
Hasil yang diperoleh	Pengguna dapat melakukan autentikasi dan dapat mengakses aplikasi sesuai <i>role</i> yang dipunyai
Hasil pengujian	Berhasil

4.5.2. Modul Permintaan

Tabel 4. Spesifikasi pengujian melihat permintaan aplikasi

Melihat Permintaan	
Kode	UJ-002
Skenario	Menekan tombol permintaan pada menu bagian kiri aplikasi
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin program 2. Terdapat setidaknya satu permintaan pada aplikasi
Hasil yang diharapkan	Data permintaan pada aplikasi terlihat sebagai tabel
Langkah pengujian	<ol style="list-style-type: none"> 1. <i>Login</i> dengan menggunakan NID dan password 2. Menekan tombol permintaan pada menu sebelah kiri aplikasi
Hasil yang diperoleh	Data permintaan pada aplikasi terlihat sebagai tabel
Hasil pengujian	Berhasil

Tabel 5. Spesifikasi pengujian menambah permintaan aplikasi

Menambah Permintaan	
Kode	UJ-003
Skenario	Menekan tombol tambah pada halaman permintaan, menambahkan data-data permintaan, dan melakukan simpan data permintaan baru tersebut.
Kondisi awal	1. Pengguna sudah melakukan <i>login</i> sebagai admin program
Hasil yang diharapkan	Data permintaan telah berhasil ditambahkan
Langkah pengujian	<ol style="list-style-type: none"> 1. Mengakses halaman permintaan melalui menu aplikasi dengan menekan tombol permintaan pada menu 2. Menekan tombol tambah pada halaman permintaan 3. Mengisikan data-data yang dibutuhkan dalam membuat satu permintaan baru 4. Menekan tombol simpan
Hasil yang diperoleh	Data permintaan baru telah berhasil ditambahkan
Hasil pengujian	Berhasil

Tabel 6. Spesifikasi pengujian mengubah permintaan aplikasi

Mengubah Permintaan	
Kode	UJ-004

Skenario	Menekan tombol permintaan pada menu bagian kiri aplikasi dan memilih salah satu permintaan untuk diubah datanya.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin program 2. Terdapat setidaknya satu permintaan pada aplikasi
Hasil yang diharapkan	Data-data pada satu permintaan telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol permintaan pada menu sebelah kiri aplikasi 2. Memilih salah satu permintaan 3. Mengubah satu dan atau banyak data dari permintaan tersebut 4. Menekan tombol simpan
Hasil yang diperoleh	Data-data pada satu permintaan telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Hasil pengujian	Berhasil

Tabel 7. Spesifikasi pengujian menghapus permintaan aplikasi

Menghapus Permintaan	
Kode	UJ-005

Skenario	Menekan tombol permintaan pada menu bagian kiri aplikasi, memilih satu atau banyak permintaan dengan menekan <i>checkbox</i> pada kolom paling kiri tabel, dan menekan tombol hapus di bagian atas tabel permintaan.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin program 2. Terdapat setidaknya satu permintaan pada aplikasi
Hasil yang diharapkan	Permintaan yang dipilih telah berhasil dihapus dari aplikasi.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol permintaan pada menu sebelah kiri aplikasi 2. Memilih satu atau banyak permintaan yang ingin dihapus dengan menekan <i>checkbox</i> pada kolom paling kiri dari tabel. 3. Menekan tombol hapus di bagian atas tabel permintaan
Hasil yang diperoleh	Permintaan yang dipilih telah berhasil dihapus dari aplikasi.
Hasil pengujian	Berhasil

Tabel 8. Spesifikasi pengujian konversi permintaan menjadi program kerja

Konversi Permintaan menjadi Program Kerja

Kode	UJ-006
Skenario	Menekan tombol permintaan pada menu bagian kiri aplikasi, memilih satu atau banyak permintaan dengan menekan <i>checkbox</i> pada kolom paling kiri tabel, dan menekan tombol Tindaklanjuti sebagai program pada bagian atas tabel permintaan.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai Manajer PTI 2. Terdapat setidaknya satu permintaan pada aplikasi
Hasil yang diharapkan	Permintaan yang dipilih telah berhasil dikonversi menjadi sebuah program kerja.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol permintaan pada menu sebelah kiri aplikasi 2. Memilih satu atau banyak permintaan yang ingin dikonversi dengan menekan <i>checkbox</i> pada kolom paling kiri dari tabel. 3. Menekan tombol Tindaklanjuti sebagai program di bagian atas tabel permintaan
Hasil yang diperoleh	Permintaan yang dipilih telah berhasil dikonversi menjadi sebuah program kerja.
Hasil pengujian	Berhasil

4.5.3. Modul Helpdesk

Tabel 9. Spesifikasi pengujian melihat melihat data helpdesk

Melihat Data Helpdesk	
Kode	UJ-007
Skenario	Menekan tombol helpdesk pada menu bagian kiri aplikasi
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin program 2. Terdapat setidaknya satu data helpdesk pada aplikasi
Hasil yang diharapkan	Data helpdesk pada aplikasi terlihat sebagai tabel
Langkah pengujian	<ol style="list-style-type: none"> 1. <i>Login</i> dengan menggunakan NID dan password 2. Menekan tombol helpdesk pada menu sebelah kiri aplikasi
Hasil yang diperoleh	Data helpdesk pada aplikasi terlihat sebagai tabel
Hasil pengujian	Berhasil

Tabel 10. Spesifikasi pengujian menyimpan data helpdesk sebagai permintaan

Menyimpan Data Helpdesk sebagai Permintaan	
Kode	UJ-008
Skenario	Menekan tombol helpdesk pada menu bagian kiri aplikasi, memilih satu atau

	banyak data helpdesk dengan menekan <i>checkbox</i> pada kolom paling kiri tabel, dan menekan tombol Simpan ke permintaan pada bagian atas tabel helpdesk.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin program 2. Terdapat setidaknya satu data helpdesk pada aplikasi
Hasil yang diharapkan	Data helpdesk yang dipilih telah berhasil disimpan sebagai permintaan pada aplikasi.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol helpdesk pada menu sebelah kiri aplikasi 2. Memilih satu atau banyak data helpdesk yang ingin disimpan sebagai permintaan dengan menekan <i>checkbox</i> pada kolom paling kiri dari tabel. 3. Menekan tombol Simpan ke permintaan di bagian atas tabel helpdesk.
Hasil yang diperoleh	Data helpdesk yang dipilih telah berhasil disimpan sebagai permintaan pada aplikasi.
Hasil pengujian	Berhasil

4.5.4. Modul Program

Tabel 11. Spesifikasi pengujian melihat program kerja aplikasi

Melihat Program Kerja	
Kode	UJ-009
Skenario	Menekan tombol program pada menu bagian kiri aplikasi
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai Manajer PTI, Manajer PAB, dan karyawan yang ikut serta dalam suatu program kerja 2. Terdapat setidaknya satu program kerja pada aplikasi
Hasil yang diharapkan	Program kerja pada aplikasi terlihat sebagai tabel
Langkah pengujian	<ol style="list-style-type: none"> 1. <i>Login</i> dengan menggunakan NID dan password 2. Menekan tombol program pada menu sebelah kiri aplikasi
Hasil yang diperoleh	Program kerja pada aplikasi terlihat sebagai tabel
Hasil pengujian	Berhasil

Tabel 12. Spesifikasi pengujian mengubah data satu program kerja

Mengubah Data Satu Program Kerja	
Kode	UJ-010
Skenario	Menekan tombol program pada menu bagian kiri

	aplikasi, memilih salah satu program kerja, mengubah data program kerja, menekan tombol simpan
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai Manajer PTI, Manajer PAB, dan karyawan yang ikut serta dalam suatu program kerja 2. Terdapat setidaknya satu program kerja pada aplikasi
Hasil yang diharapkan	Data-data pada satu program kerja pada aplikasi telah diubah dan perubahan telah disimpan dalam aplikasi
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol program pada menu sebelah kiri aplikasi 2. Memilih salah satu program kerja pada aplikasi 3. Sebagai Manajer PAB atau Manajer PTI: dapat mengubah semua data yang berkaitan dengan satu program kerja 4. Sebagai karyawan yang terlibat dalam program kerja: dapat mengubah data yang berkaitan dengan progress program kerja dan catatan perubahan dari program kerja tersebut 5. Setelah mengubah data program kerja, menekan

	tombol simpan untuk menyimpan data perubahan tersebut.
Hasil yang diperoleh	Data-data pada satu program kerja pada aplikasi telah diubah dan perubahan telah disimpan dalam aplikasi
Hasil pengujian	Berhasil

4.5.5. Modul Gantt Chart

Tabel 13. Spesifikasi pengujian melihat aktivitas pada satu program kerja

Melihat Aktivitas pada satu Program Kerja	
Kode	UJ-011
Skenario	Membuka halaman program kerja, lalu membuka detail aktivitas satu program kerja
Kondisi awal	1. Pengguna sudah melakukan <i>login</i> sebagai Manajer PAB dan admin.
Hasil yang diharapkan	Data aktivitas dari satu program kerja dapat ditampilkan dalam bentuk <i>gantt chart</i>
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol program pada menu sebelah kiri aplikasi. 2. Memilih salah satu program yang ingin ditampilkan aktivitasnya.

	3. Menekan judul dari program kerja yang dipilih
Hasil yang diperoleh	Data aktivitas dari satu program kerja dapat ditampilkan dalam bentuk <i>gant chart</i>
Hasil pengujian	Berhasil

Tabel 14. Spesifikasi pengujian menambah aktivitas pada satu program kerja

Menambah Aktivitas pada satu Program Kerja	
Kode	UJ-012
Skenario	<p>Membuka halaman program kerja, lalu membuka detail aktivitas satu program kerja. Terdapat dua skenario, yaitu:</p> <ol style="list-style-type: none"> 1. Menekan baris terbawah di bawah judul program kerja dan menamai aktivitas tersebut 2. Menekan baris salah satu aktivitas lalu menekan <i>icon insert above</i> atau <i>insert below</i> untuk menambahkan baris baru di atas atau di bawah baris yang dipilih, dan menamai baris baru tersebut dengan aktivitas baru
Kondisi awal	1. Pengguna sudah melakukan <i>login</i> sebagai Manajer PAB dan admin.

Hasil yang diharapkan	Data aktivitas baru ditambahkan ke program kerja yang dipilih
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol program pada menu sebelah kiri aplikasi. 2. Memilih salah satu program yang ingin ditampilkan aktivitasnya. 3. Menekan judul dari program kerja yang dipilih 4. Menambah aktivitas dengan menekan baris terbawah di bawah judul program kerja atau menekan baris salah satu aktivitas dan menekan <i>icon insert above</i> atau <i>insert below</i> untuk menambahkan baris baru di atas atau di bawah baris yang dipilih. 5. Mengisikan nama aktivitas, kode aktivitas, tanggal mulai, dan tanggal selesai 6. Menekan tombol <i>save</i>
Hasil yang diperoleh	Data aktivitas baru ditambahkan ke program kerja yang dipilih
Hasil pengujian	Berhasil

Tabel 15. Spesifikasi pengujian mengubah aktivitas pada satu program kerja

Mengubah Aktivitas pada satu Program Kerja

Kode	UJ-013
Skenario	<p>Membuka halaman program kerja, lalu membuka detail aktivitas satu program kerja, terdapat dua skenario, yaitu:</p> <ol style="list-style-type: none"> 1. Memilih salah satu aktivitas dan mengganti data-data dari aktivitas tersebut 2. Menekan <i>timeline bar</i> salah satu aktivitas pada tampilan sebelah kanan aplikasi dan mengganti data-data dari aktivitas tersebut
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai Manajer PAB dan admin. 2. Setidaknya terdapat satu aktivitas di program kerja yang dipilih
Hasil yang diharapkan	Data-data pada satu aktivitas yang berada di satu program kerja berhasil diubah dan disimpan oleh sistem.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol program pada menu sebelah kiri aplikasi. 2. Memilih salah satu program yang ingin ditampilkan aktivitasnya. 3. Menekan judul dari program kerja yang dipilih 4. Memilih satu aktivitas dengan dua skenario, yaitu

	<p>menekan baris aktivitas tersebut, atau menekan <i>timeline bar</i> dari aktivitas tersebut</p> <p>5. Mengubah data-data dari aktivitas tersebut</p> <p>6. Menekan tombol <i>save</i></p>
Hasil yang diperoleh	Data-data pada satu aktivitas yang berada di satu program kerja berhasil diubah dan disimpan oleh sistem.
Hasil pengujian	Berhasil

Tabel 16. Spesifikasi pengujian menghapus aktivitas pada satu program kerja

Menghapus Aktivitas pada satu Program Kerja	
Kode	UJ-014
Skenario	<p>Membuka halaman program kerja, lalu membuka detail aktivitas satu program kerja, terdapat dua skenario, yaitu:</p> <ol style="list-style-type: none"> 1. Memilih salah satu aktivitas dengan menekan baris dari aktivitasnya dan menekan <i>icon</i> sampah yang berada pada bagian atas tampilan aplikasi 2. Menekan <i>timeline bar</i> salah satu aktivitas pada tampilan sebelah kanan aplikasi dan menekan <i>icon</i> sampah yang berada pada bagian atas tampilan aplikasi atau

	menekan tombol <i>delete</i> pada <i>keyboard</i>
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai Manajer PAB dan admin. 2. Setidaknya terdapat satu aktivitas di program kerja yang dipilih
Hasil yang diharapkan	Satu aktivitas yang dipilih berhasil dihapus dari satu program kerja
L.angkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol program pada menu sebelah kiri aplikasi. 2. Memilih salah satu program yang ingin ditampilkan aktivitasnya. 3. Menekan judul dari program kerja yang dipilih 4. Memilih satu aktivitas dengan dua skenario, yaitu menekan baris aktivitas tersebut, atau menekan <i>timeline bar</i> dari aktivitas tersebut 5. Menghapus aktivitas yang dipilih dengan menekan <i>icon</i> sampah yang berada pada bagian atas tampilan aplikasi atau menghapus dengan menekan tombol <i>delete</i> pada <i>keyboard</i> apabila memilih aktivitas

	dengan menekan <i>timeline bar</i> -nya.
Hasil yang diperoleh	Satu aktivitas yang dipilih berhasil dihapus dari satu program kerja
Hasil pengujian	Berhasil

Tabel 17. Spesifikasi pengujian mengubah urutan aktivitas pada satu program kerja

Mengubah Urutan Aktivitas pada satu Program Kerja	
Kode	UJ-015
Skenario	Membuka halaman program kerja, lalu membuka detail aktivitas satu program kerja, memilih salah satu aktivitas dan menekan tombol <i>move up</i> atau <i>move down</i>
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai Manajer PAB dan admin. 2. Setidaknya terdapat dua aktivitas di program kerja yang dipilih
Hasil yang diharapkan	Aktivitas yang dipilih berubah urutannya dan tersimpan urutannya dalam aplikasi
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol program pada menu sebelah kiri aplikasi. 2. Memilih salah satu program yang ingin ditampilkan aktivitasnya.

	<ol style="list-style-type: none"> 3. Menekan judul dari program kerja yang dipilih 4. Memilih salah satu aktivitas dengan menekan baris aktivitas tersebut 5. menekan tombol <i>move up</i> atau <i>move down</i> pada tampilan bagian atas aplikasi.
Hasil yang diperoleh	Aktivitas yang dipilih berubah urutannya dan tersimpan urutannya dalam aplikasi
Hasil pengujian	Berhasil

Tabel 18. Spesifikasi pengujian mengubah project team pada satu program kerja

Mengubah Project Team pada satu Program Kerja	
Kode	UJ-016
Skenario	Membuka halaman program kerja, lalu membuka detail aktivitas satu program kerja, menekan <i>icon edit resources</i> pada bagian atas tampilan aplikasi, dan mengubah serta menghapus data <i>Project Team</i> satu program kerja
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai Manajer PAB dan admin.
Hasil yang diharapkan	Data <i>Project Team</i> pada satu program kerja telah berubah dan telah disimpan dalam aplikasi.

Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol program pada menu sebelah kiri aplikasi. 2. Memilih salah satu program yang ingin ditampilkan aktivitasnya. 3. Menekan <i>icon edit resources</i> pada bagian atas tampilan aplikasi 4. Mengubah serta menghapus data <i>Project Team</i> satu program kerja
Hasil yang diperoleh	Data <i>Project Team</i> pada satu program kerja telah berubah dan telah disimpan dalam aplikasi.
Hasil pengujian	Berhasil

4.5.6. Modul Data *Master*

Tabel 19. Spesifikasi pengujian melihat data *client*

Melihat Data <i>Client</i>	
Kode	UJ-017
Skenario	Menekan tombol admin pada bagian atas aplikasi, lalu menekan tombol <i>client</i> pada menu bagian kiri halaman admin.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem.

	2. Terdapat setidaknya satu data <i>client</i> pada aplikasi.
Hasil yang diharapkan	Data <i>client</i> pada aplikasi terlihat sebagai tabel.
Langkah pengujian	<ol style="list-style-type: none"> 1. <i>Login</i> dengan menggunakan NID dan password. 2. Menekan tombol admin pada bagian atas aplikasi. 3. Menekan tombol <i>Client</i> di bagian kiri aplikasi.
Hasil yang diperoleh	Data <i>client</i> pada aplikasi terlihat sebagai tabel
Hasil pengujian	Berhasil

Tabel 20. Spesifikasi pengujian menambah data *client*

Menambah Data Client	
Kode	UJ-018
Skenario	Menekan tombol tambah pada data <i>client</i> , menambahkan data-data <i>client</i> , dan melakukan simpan data <i>client</i> baru tersebut.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem.
Hasil yang diharapkan	Data <i>client</i> telah berhasil ditambahkan.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Mengakses halaman data <i>client</i> melalui menu aplikasi dengan menekan

	<p>tombol <i>client</i> pada menu dibagian kiri halaman.</p> <p>3. Menekan tombol tambah pada halaman data <i>client</i>.</p> <p>4. Mengisikan data-data yang dibutuhkan dalam membuat satu data <i>client</i> baru.</p> <p>5. Menekan tombol simpan.</p>
Hasil yang diperoleh	Data <i>client</i> baru telah berhasil ditambahkan
Hasil pengujian	Berhasil

Tabel 21. Spesifikasi pengujian mengubah data *client*

Mengubah Data Client	
Kode	UJ-019
Skenario	Menekan tombol <i>client</i> pada menu bagian kiri aplikasi dan memilih salah satu baris data <i>client</i> untuk diubah datanya.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data <i>client</i> pada aplikasi
Hasil yang diharapkan	Data-data pada satu data <i>client</i> telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol <i>client</i> pada menu sebelah kiri aplikasi.

	<ol style="list-style-type: none"> 3. Memilih salah satu data <i>client</i>. 4. Mengubah satu dan atau banyak data dari data <i>client</i> tersebut. 5. Menekan tombol simpan.
Hasil yang diperoleh	Data pada satu data <i>client</i> telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Hasil pengujian	Berhasil

Tabel 22. Spesifikasi pengujian menghapus data *client*

Menghapus Data Client	
Kode	UJ-020
Skenario	Menekan tombol <i>client</i> pada menu bagian kiri aplikasi, memilih satu atau banyak data <i>client</i> dengan menekan <i>checkbox</i> pada kolom paling kiri tabel, dan menekan tombol hapus di bagian atas tabel data <i>client</i> .
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data <i>client</i> pada aplikasi
Hasil yang diharapkan	Data <i>client</i> yang dipilih telah berhasil dihapus dari aplikasi.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi.

	<ol style="list-style-type: none"> 2. Menekan tombol <i>client</i> pada menu sebelah kiri aplikasi 3. Memilih satu atau banyak data <i>client</i> yang ingin dihapus dengan menekan <i>checkbox</i> pada kolom paling kiri tabel. 4. Menekan tombol hapus di bagian atas tabel data <i>client</i>.
Hasil yang diperoleh	Data <i>client</i> yang dipilih telah berhasil dihapus dari aplikasi.
Hasil pengujian	Berhasil

Tabel 23. Spesifikasi pengujian melihat data dokumen

Melihat Data Dokumen	
Kode	UJ-021
Skenario	Menekan tombol admin pada bagian atas aplikasi, lalu menekan tombol dokumen pada menu bagian kiri halaman admin.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data dokumen pada aplikasi.
Hasil yang diharapkan	Data dokumen pada aplikasi terlihat sebagai tabel.
Langkah pengujian	<ol style="list-style-type: none"> 1. <i>Login</i> dengan menggunakan NID dan password.

	<ol style="list-style-type: none"> 2. Menekan tombol admin pada bagian atas aplikasi. 3. Menekan tombol dokumen di bagian kiri aplikasi.
Hasil yang diperoleh	Data dokumen pada aplikasi terlihat sebagai tabel.
Hasil pengujian	Berhasil

Tabel 24. Spesifikasi pengujian menambah data dokumen

Menambah Data Dokumen	
Kode	UJ-022
Skenario	Menekan tombol tambah pada data dokumen, menambahkan data-data dokumen, dan melakukan simpan data <i>client</i> baru tersebut.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem.
Hasil yang diharapkan	Data dokumen telah berhasil ditambahkan.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Mengakses halaman data dokumen melalui menu aplikasi dengan menekan tombol dokumen pada menu dibagian kiri halaman. 3. Menekan tombol tambah pada halaman data data dokumen.

	<ol style="list-style-type: none"> 4. Mengisikan data-data yang dibutuhkan dalam membuat satu data dokumen baru. 5. Menekan tombol simpan.
Hasil yang diperoleh	Data dokumen baru telah berhasil ditambahkan.
Hasil pengujian	Berhasil

Tabel 25. Spesifikasi pengujian mengubah data dokumen

Mengubah Data Dokumen	
Kode	UJ-023
Skenario	Menekan tombol dokumen pada menu bagian kiri aplikasi dan memilih salah satu baris data dokumen untuk diubah datanya.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data dokumen pada aplikasi
Hasil yang diharapkan	Data-data pada satu data dokumen telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol dokumen pada menu sebelah kiri aplikasi. 3. Memilih salah satu data dokumen.

	<p>4. Mengubah satu dan atau banyak data dari data dokumen tersebut.</p> <p>5. Menekan tombol simpan.</p>
Hasil yang diperoleh	Data pada satu data dokumen telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Hasil pengujian	Berhasil

Tabel 26. Spesifikasi pengujian menghapus data dokumen

Menghapus Data Dokumen	
Kode	UJ-024
Skenario	Menekan tombol dokumen pada menu bagian kiri aplikasi, memilih satu atau banyak data dokumen dengan menekan <i>checkbox</i> pada kolom paling kiri tabel, dan menekan tombol hapus di bagian atas tabel data dokumen.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data dokumen pada aplikasi
Hasil yang diharapkan	Data dokumen yang dipilih telah berhasil dihapus dari aplikasi.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi.

	<ol style="list-style-type: none"> 2. Menekan tombol dokumen pada menu sebelah kiri aplikasi 3. Memilih satu atau banyak data dokumen yang ingin dihapus dengan menekan <i>checkbox</i> pada kolom paling kiri tabel. 4. Menekan tombol hapus di bagian atas tabel data dokumen.
Hasil yang diperoleh	Data dokumen yang dipilih telah berhasil dihapus dari aplikasi.
Hasil pengujian	Berhasil

Tabel 27. Spesifikasi pengujian melihat data jenis program

Melihat Data Jenis Program	
Kode	UJ-025
Skenario	Menekan tombol admin pada bagian atas aplikasi, lalu menekan tombol jenis program pada menu bagian kiri halaman admin.
Kondisi awal	<ol style="list-style-type: none"> 3. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 4. Terdapat setidaknya satu data jenis program pada aplikasi.
Hasil yang diharapkan	Data jenis program pada aplikasi terlihat sebagai tabel.

Langkah pengujian	<p>4. <i>Login</i> dengan menggunakan NID dan password.</p> <p>5. Menekan tombol admin pada bagian atas aplikasi.</p> <p>6. Menekan tombol jenis program di bagian kiri aplikasi.</p>
Hasil yang diperoleh	Data jenis program pada aplikasi terlihat sebagai tabel.
Hasil pengujian	Berhasil

Tabel 28. Spesifikasi pengujian menambah data jenis program

Menambah Data Jenis Program	
Kode	UJ-026
Skenario	Menekan tombol tambah pada data jenis program, menambahkan data-data jenis program, dan melakukan simpan data jenis program baru tersebut.
Kondisi awal	2. Pengguna sudah melakukan <i>login</i> sebagai admin sistem.
Hasil yang diharapkan	Data jenis program telah berhasil ditambahkan.
Langkah pengujian	<p>5. Menekan tombol admin pada bagian atas aplikasi.</p> <p>6. Mengakses halaman data jenis program melalui menu aplikasi dengan menekan tombol jenis program pada menu dibagian kiri halaman.</p>

	<ol style="list-style-type: none"> 7. Menekan tombol tambah pada halaman data jenis program. 8. Mengisikan data-data yang dibutuhkan dalam membuat satu data jenis program baru. 9. Menekan tombol simpan.
Hasil yang diperoleh	Data jenis program baru telah berhasil ditambahkan.
Hasil pengujian	Berhasil

Tabel 29. Spesifikasi pengujian mengubah data jenis program

Mengubah Data Jenis Program	
Kode	UJ-027
Skenario	Menekan tombol jenis program pada menu bagian kiri aplikasi dan memilih salah satu baris data jenis program untuk diubah datanya.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data jenis program pada aplikasi.
Hasil yang diharapkan	Data-data pada satu data jenis program telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi.

	<ol style="list-style-type: none"> 2. Menekan tombol jenis program pada menu sebelah kiri aplikasi. 3. Memilih salah satu data jenis program. 4. Mengubah satu dan atau banyak data dari data jenis program tersebut. 5. Menekan tombol simpan.
Hasil yang diperoleh	Data pada satu data jenis program telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Hasil pengujian	Berhasil

Tabel 30. Spesifikasi pengujian menghapus data jenis program

Menghapus Data Jenis Program	
Kode	UJ-028
Skenario	Menekan tombol jenis program pada menu bagian kiri aplikasi, memilih satu atau banyak data jenis program dengan menekan <i>checkbox</i> pada kolom paling kiri tabel, dan menekan tombol hapus di bagian atas tabel data <i>client</i> .
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data <i>client</i> pada aplikasi
Hasil yang diharapkan	Data jenis program yang dipilih telah berhasil dihapus dari aplikasi.

L.angkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol jenis program pada menu sebelah kiri aplikasi 3. Memilih satu atau banyak data jenis program yang ingin dihapus dengan menekan <i>checkbox</i> pada kolom paling kiri tabel. 4. Menekan tombol hapus di bagian atas tabel data jenis program.
Hasil yang diperoleh	Data jenis program yang dipilih telah berhasil dihapus dari aplikasi.
Hasil pengujian	Berhasil

Tabel 31. Spesifikasi pengujian melihat data kategori program

Melihat Data Kategori Program	
Kode	UJ-029
Skenario	Menekan tombol admin pada bagian atas aplikasi, lalu menekan tombol kategori program pada menu bagian kiri halaman admin.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data kategori program pada aplikasi.
Hasil yang diharapkan	Data kategori program pada aplikasi terlihat sebagai tabel.

Langkah pengujian	<ol style="list-style-type: none"> 1. <i>Login</i> dengan menggunakan NID dan password. 2. Menekan tombol admin pada bagian atas aplikasi. 3. Menekan tombol kategori program di bagian kiri aplikasi.
Hasil yang diperoleh	Data kategori program pada aplikasi terlihat sebagai tabel
Hasil pengujian	Berhasil

Tabel 32. Spesifikasi pengujian menambah data kategori program

Menambah Data Kategori Program	
Kode	UJ-030
Skenario	Menekan tombol tambah pada data kategori program, menambahkan data-data kategori program, dan melakukan simpan data kategori program baru tersebut.
Kondisi awal	1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem.
Hasil yang diharapkan	Data kategori program telah berhasil ditambahkan.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Mengakses halaman data kategori program melalui menu aplikasi dengan menekan tombol kategori

	<p>program pada menu dibagian kiri halaman.</p> <ol style="list-style-type: none"> 3. Menekan tombol tambah pada halaman data kategori program. 4. Mengisikan data-data yang dibutuhkan dalam membuat satu data kategori program baru. 5. Menekan tombol simpan.
Hasil yang diperoleh	Data kategori program baru telah berhasil ditambahkan
Hasil pengujian	Berhasil

Tabel 33. Spesifikasi pengujian mengubah data kategori program

Mengubah Data Kategori Program	
Kode	UJ-031
Skenario	Menekan tombol kategori program pada menu bagian kiri aplikasi dan memilih salah satu baris data kategori program untuk diubah datanya.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data kategori program pada aplikasi
Hasil yang diharapkan	Data-data pada satu data kategori program telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.

Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol kategori program pada menu sebelah kiri aplikasi. 3. Memilih salah satu data kategori program. 4. Mengubah satu dan atau banyak data dari data kategori program tersebut. 5. Menekan tombol simpan.
Hasil yang diperoleh	Data pada satu data kategori program telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Hasil pengujian	Berhasil

Tabel 34. Spesifikasi pengujian menghapus data kategori program

Menghapus Data Kategori Program	
Kode	UJ-032
Skenario	Menekan tombol kategori program pada menu bagian kiri aplikasi, memilih satu atau banyak data kategori program dengan menekan <i>checkbox</i> pada kolom paling kiri tabel, dan menekan tombol hapus di bagian atas tabel data kategori program.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem.

	2. Terdapat setidaknya satu data kategori program pada aplikasi
Hasil yang diharapkan	Data kategori program yang dipilih telah berhasil dihapus dari aplikasi.
L.angkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol kategori program pada menu sebelah kiri aplikasi 3. Memilih satu atau banyak data kategori program yang ingin dihapus dengan menekan <i>checkbox</i> pada kolom paling kiri tabel. 4. Menekan tombol hapus di bagian atas tabel data kategori program.
Hasil yang diperoleh	Data kategori program yang dipilih telah berhasil dihapus dari aplikasi.
Hasil pengujian	Berhasil

Tabel 35. Spesifikasi pengujian melihat data kelompok

Melihat Data Kelompok	
Kode	UJ-033
Skenario	Menekan tombol admin pada bagian atas aplikasi, lalu menekan tombol kelompok pada menu bagian kiri halaman admin.

Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data kelompok pada aplikasi.
Hasil yang diharapkan	Data kelompok pada aplikasi terlihat sebagai tabel.
Langkah pengujian	<ol style="list-style-type: none"> 1. <i>Login</i> dengan menggunakan NID dan password. 2. Menekan tombol admin pada bagian atas aplikasi. 3. Menekan tombol Kelompok di bagian kiri aplikasi.
Hasil yang diperoleh	Data kelompok pada aplikasi terlihat sebagai tabel
Hasil pengujian	Berhasil

Tabel 36. Spesifikasi pengujian menambah data kelompok

Menambah Data Kelompok	
Kode	UJ-034
Skenario	Menekan tombol tambah pada data kelompok, menambahkan data-data kelompok, dan melakukan simpan data kelompok baru tersebut.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem.
Hasil yang diharapkan	Data kelompok telah berhasil ditambahkan.

Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Mengakses halaman data kelompok melalui menu aplikasi dengan menekan tombol kelompok pada menu dibagian kiri halaman. 3. Menekan tombol tambah pada halaman data kelompok. 4. Mengisikan data-data yang dibutuhkan dalam membuat satu data kelompok baru. 5. Menekan tombol simpan.
Hasil yang diperoleh	Data kelompok baru telah berhasil ditambahkan
Hasil pengujian	Berhasil

Tabel 37. Spesifikasi pengujian mengubah data kelompok

Mengubah Data Kelompok	
Kode	UJ-035
Skenario	Menekan tombol kelompok pada menu bagian kiri aplikasi dan memilih salah satu baris data kelompok untuk diubah datanya.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data kelompok pada aplikasi

Hasil yang diharapkan	Data-data pada satu data kelompok telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol kelompok pada menu sebelah kiri aplikasi. 3. Memilih salah satu data kelompok. 4. Mengubah satu dan atau banyak data dari data kelompok tersebut. 5. Menekan tombol simpan.
Hasil yang diperoleh	Data pada satu data kelompok telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Hasil pengujian	Berhasil

Tabel 38. Spesifikasi pengujian menghapus data kelompok

Menghapus Data Kelompok	
Kode	UJ-036
Skenario	Menekan tombol kelompok pada menu bagian kiri aplikasi, memilih satu atau banyak data kelompok dengan menekan <i>checkbox</i> pada kolom paling kiri tabel, dan menekan tombol hapus di bagian atas tabel data kelompok.

Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data kelompok pada aplikasi
Hasil yang diharapkan	Data kelompok yang dipilih telah berhasil dihapus dari aplikasi.
L.angkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol kelompok pada menu sebelah kiri aplikasi 3. Memilih satu atau banyak data kelompok yang ingin dihapus dengan menekan <i>checkbox</i> pada kolom paling kiri tabel. 4. Menekan tombol hapus di bagian atas tabel data kelompok.
Hasil yang diperoleh	Data kelompok yang dipilih telah berhasil dihapus dari aplikasi.
Hasil pengujian	Berhasil

Tabel 39. Spesifikasi pengujian melihat data media

Melihat Data Media	
Kode	UJ-037
Skenario	Menekan tombol admin pada bagian atas aplikasi, lalu menekan tombol media pada

	menu bagian kiri halaman admin.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data media pada aplikasi.
Hasil yang diharapkan	Data media pada aplikasi terlihat sebagai tabel.
Langkah pengujian	<ol style="list-style-type: none"> 1. <i>Login</i> dengan menggunakan NID dan password. 2. Menekan tombol admin pada bagian atas aplikasi. 3. Menekan tombol Media di bagian kiri aplikasi.
Hasil yang diperoleh	Data media pada aplikasi terlihat sebagai tabel
Hasil pengujian	Berhasil

Tabel 40. Spesifikasi pengujian menambah data media

Menambah Data Media	
Kode	UJ-038
Skenario	Menekan tombol tambah pada data media, menambahkan data-data media, dan melakukan simpan data media baru tersebut.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem.
Hasil yang diharapkan	Data media telah berhasil ditambahkan.

Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Mengakses halaman data media melalui menu aplikasi dengan menekan tombol media pada menu dibagian kiri halaman. 3. Menekan tombol tambah pada halaman data media. 4. Mengisikan data-data yang dibutuhkan dalam membuat satu data media baru. 5. Menekan tombol simpan.
Hasil yang diperoleh	Data media baru telah berhasil ditambahkan
Hasil pengujian	Berhasil

Tabel 41. Spesifikasi pengujian mengubah data media

Mengubah Data Media	
Kode	UJ-039
Skenario	Menekan tombol media pada menu bagian kiri aplikasi dan memilih salah satu baris data media untuk diubah datanya.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data media pada aplikasi
Hasil yang diharapkan	Data-data pada satu data media telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.

Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol media pada menu sebelah kiri aplikasi. 3. Memilih salah satu data media. 4. Mengubah satu dan atau banyak data dari data media tersebut. 5. Menekan tombol simpan.
Hasil yang diperoleh	Data pada satu data media telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Hasil pengujian	Berhasil

Tabel 42. Spesifikasi pengujian menghapus data media

Menghapus Data Media	
Kode	UJ-040
Skenario	Menekan tombol media pada menu bagian kiri aplikasi, memilih satu atau banyak data media dengan menekan <i>checkbox</i> pada kolom paling kiri tabel, dan menekan tombol hapus di bagian atas tabel data media.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data media pada aplikasi

Hasil yang diharapkan	Data media yang dipilih telah berhasil dihapus dari aplikasi.
L.angkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol media pada menu sebelah kiri aplikasi 3. Memilih satu atau banyak data media yang ingin dihapus dengan menekan <i>checkbox</i> pada kolom paling kiri tabel. 4. Menekan tombol hapus di bagian atas tabel data media.
Hasil yang diperoleh	Data media yang dipilih telah berhasil dihapus dari aplikasi.
Hasil pengujian	Berhasil

Tabel 43. Spesifikasi pengujian melihat data prioritas

Melihat Data Prioritas	
Kode	UJ-041
Skenario	Menekan tombol admin pada bagian atas aplikasi, lalu menekan tombol prioritas pada menu bagian kiri halaman admin.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data prioritas pada aplikasi.
Hasil yang diharapkan	Data prioritas pada aplikasi terlihat sebagai tabel.

Langkah pengujian	<ol style="list-style-type: none"> 1. <i>Login</i> dengan menggunakan NID dan password. 2. Menekan tombol admin pada bagian atas aplikasi. 3. Menekan tombol Prioritas di bagian kiri aplikasi.
Hasil yang diperoleh	Data prioritas pada aplikasi terlihat sebagai tabel
Hasil pengujian	Berhasil

Tabel 44. Spesifikasi pengujian menambah data prioritas

Menambah Data Prioritas	
Kode	UJ-042
Skenario	Menekan tombol tambah pada data prioritas, menambahkan data-data prioritas, dan melakukan simpan data prioritas baru tersebut.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem.
Hasil yang diharapkan	Data prioritas telah berhasil ditambahkan.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Mengakses halaman data prioritas melalui menu aplikasi dengan menekan tombol prioritas pada menu dibagian kiri halaman. 3. Menekan tombol tambah pada halaman data prioritas.

	<ol style="list-style-type: none"> 4. Mengisikan data-data yang dibutuhkan dalam membuat satu data prioritas baru. 5. Menekan tombol simpan.
Hasil yang diperoleh	Data prioritas baru telah berhasil ditambahkan
Hasil pengujian	Berhasil

Tabel 45. Spesifikasi pengujian mengubah data prioritas

Mengubah Data Prioritas	
Kode	UJ-043
Skenario	Menekan tombol prioritas pada menu bagian kiri aplikasi dan memilih salah satu baris data prioritas untuk diubah datanya.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data prioritas pada aplikasi
Hasil yang diharapkan	Data-data pada satu data prioritas telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol prioritas pada menu sebelah kiri aplikasi. 3. Memilih salah satu data prioritas.

	<p>4. Mengubah satu dan atau banyak data dari data prioritas tersebut.</p> <p>5. Menekan tombol simpan.</p>
Hasil yang diperoleh	Data pada satu data prioritas telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Hasil pengujian	Berhasil

Tabel 46. Spesifikasi pengujian menghapus data prioritas

Menghapus Data Prioritas	
Kode	UJ-044
Skenario	Menekan tombol prioritas pada menu bagian kiri aplikasi, memilih satu atau banyak data prioritas dengan menekan <i>checkbox</i> pada kolom paling kiri tabel, dan menekan tombol hapus di bagian atas tabel data prioritas.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data prioritas pada aplikasi
Hasil yang diharapkan	Data prioritas yang dipilih telah berhasil dihapus dari aplikasi.
L.angkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol prioritas pada menu sebelah kiri aplikasi

	<ol style="list-style-type: none"> 3. Memilih satu atau banyak data prioritas yang ingin dihapus dengan menekan <i>checkbox</i> pada kolom paling kiri tabel. 4. Menekan tombol hapus di bagian atas tabel data prioritas.
Hasil yang diperoleh	Data prioritas yang dipilih telah berhasil dihapus dari aplikasi.
Hasil pengujian	Berhasil

Tabel 47. Spesifikasi pengujian melihat data *resource*

Melihat Data Resource	
Kode	UJ-045
Skenario	Menekan tombol admin pada bagian atas aplikasi, lalu menekan tombol <i>resource</i> pada menu bagian kiri halaman admin.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data <i>resource</i> pada aplikasi.
Hasil yang diharapkan	Data <i>resource</i> pada aplikasi terlihat sebagai tabel.
Langkah pengujian	<ol style="list-style-type: none"> 1. <i>Login</i> dengan menggunakan NID dan password. 2. Menekan tombol admin pada bagian atas aplikasi.

	3. Menekan tombol <i>Resource</i> di bagian kiri aplikasi.
Hasil yang diperoleh	Data <i>resource</i> pada aplikasi terlihat sebagai tabel.
Hasil pengujian	Berhasil

Tabel 48. Spesifikasi pengujian menambah data *resource*

Menambah Data Resource	
Kode	UJ-046
Skenario	Menekan tombol tambah pada data <i>resource</i> , menambahkan data-data <i>resource</i> , dan melakukan simpan data <i>resource</i> baru tersebut.
Kondisi awal	1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem.
Hasil yang diharapkan	Data <i>resource</i> telah berhasil ditambahkan.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Mengakses halaman data <i>resource</i> melalui menu aplikasi dengan menekan tombol <i>resource</i> pada menu dibagian kiri halaman. 3. Menekan tombol tambah pada halaman data <i>resource</i>. 4. Mengisikan data-data yang dibutuhkan dalam membuat satu data <i>resource</i> baru. 5. Menekan tombol simpan.

Hasil yang diperoleh	Data <i>resource</i> baru telah berhasil ditambahkan
Hasil pengujian	Berhasil

Tabel 49. Spesifikasi pengujian mengubah data *resource*

Mengubah Data Resource	
Kode	UJ-047
Skenario	Menekan tombol <i>resource</i> pada menu bagian kiri aplikasi dan memilih salah satu baris data <i>resource</i> untuk diubah datanya.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data <i>resource</i> pada aplikasi
Hasil yang diharapkan	Data-data pada satu data <i>resource</i> telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol <i>resource</i> pada menu sebelah kiri aplikasi. 3. Memilih salah satu data <i>resource</i>. 4. Mengubah satu dan atau banyak data dari data <i>resource</i> tersebut. 5. Menekan tombol simpan.
Hasil yang diperoleh	Data pada satu data <i>resource</i> telah berhasil diubah dan

	perubahan tersebut telah disimpan oleh aplikasi.
Hasil pengujian	Berhasil

Tabel 50. Spesifikasi pengujian menghapus data *resource*

Menghapus Data Resource	
Kode	UJ-048
Skenario	Menekan tombol <i>resource</i> pada menu bagian kiri aplikasi, memilih satu atau banyak data <i>resource</i> dengan menekan <i>checkbox</i> pada kolom paling kiri tabel, dan menekan tombol hapus di bagian atas tabel data <i>resource</i> .
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data <i>resource</i> pada aplikasi
Hasil yang diharapkan	Data <i>resource</i> yang dipilih telah berhasil dihapus dari aplikasi.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol <i>resource</i> pada menu sebelah kiri aplikasi 3. Memilih satu atau banyak data <i>resource</i> yang ingin dihapus dengan menekan <i>checkbox</i> pada kolom paling kiri tabel.

	4. Menekan tombol hapus di bagian atas tabel data <i>resource</i> .
Hasil yang diperoleh	Data <i>resource</i> yang dipilih telah berhasil dihapus dari aplikasi.
Hasil pengujian	Berhasil

Tabel 51. Spesifikasi pengujian melihat data RJPP

Melihat Data RJPP	
Kode	UJ-049
Skenario	Menekan tombol admin pada bagian atas aplikasi, lalu menekan tombol RJPP pada menu bagian kiri halaman admin.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data RJPP pada aplikasi.
Hasil yang diharapkan	Data RJPP pada aplikasi terlihat sebagai tabel.
Langkah pengujian	<ol style="list-style-type: none"> 1. <i>Login</i> dengan menggunakan NID dan password. 2. Menekan tombol admin pada bagian atas aplikasi. 3. Menekan tombol RJPP di bagian kiri aplikasi.
Hasil yang diperoleh	Data RJPP pada aplikasi terlihat sebagai tabel
Hasil pengujian	Berhasil

Tabel 52. Spesifikasi pengujian menambah data RJPP

Menambah Data RJPP	
Kode	UJ-050
Skenario	Menekan tombol tambah pada data RJPP, menambahkan data-data RJPP, dan melakukan simpan data RJPP baru tersebut.
Kondisi awal	1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem.
Hasil yang diharapkan	Data RJPP telah berhasil ditambahkan.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Mengakses halaman data RJPP melalui menu aplikasi dengan menekan tombol RJPP pada menu dibagian kiri halaman. 3. Menekan tombol tambah pada halaman data RJPP. 4. Mengisikan data-data yang dibutuhkan dalam membuat satu data RJPP baru. 5. Menekan tombol simpan.
Hasil yang diperoleh	Data RJPP baru telah berhasil ditambahkan
Hasil pengujian	Berhasil

Tabel 53. Spesifikasi pengujian mengubah data RJPP

Mengubah Data RJPP	
Kode	UJ-051

Skenario	Menekan tombol RJPP pada menu bagian kiri aplikasi dan memilih salah satu baris data RJPP untuk diubah datanya.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data RJPP pada aplikasi
Hasil yang diharapkan	Data-data pada satu data RJPP telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol RJPP pada menu sebelah kiri aplikasi. 3. Memilih salah satu data RJPP. 4. Mengubah satu dan atau banyak data dari data RJPP tersebut. 5. Menekan tombol simpan.
Hasil yang diperoleh	Data pada satu data RJPP telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Hasil pengujian	Berhasil

Tabel 54. Spesifikasi pengujian menghapus data RJPP

Menghapus Data RJPP	
Kode	UJ-052

Skenario	Menekan tombol RJPP pada menu bagian kiri aplikasi, memilih satu atau banyak data RJPP dengan menekan <i>checkbox</i> pada kolom paling kiri tabel, dan menekan tombol hapus di bagian atas tabel data RJPP.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data RJPP pada aplikasi
Hasil yang diharapkan	Data RJPP yang dipilih telah berhasil dihapus dari aplikasi.
L.angkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol RJPP pada menu sebelah kiri aplikasi 3. Memilih satu atau banyak data RJPP yang ingin dihapus dengan menekan <i>checkbox</i> pada kolom paling kiri tabel. 4. Menekan tombol hapus di bagian atas tabel data RJPP.
Hasil yang diperoleh	Data RJPP yang dipilih telah berhasil dihapus dari aplikasi.
Hasil pengujian	Berhasil

Tabel 55. Spesifikasi pengujian melihat data *roles*

Melihat Data Roles	
Kode	UJ-053

Skenario	Menekan tombol admin pada bagian atas aplikasi, lalu menekan tombol <i>Roles</i> pada menu bagian kiri halaman admin.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data <i>Roles</i> pada aplikasi.
Hasil yang diharapkan	Data <i>Roles</i> pada aplikasi terlihat sebagai tabel.
Langkah pengujian	<ol style="list-style-type: none"> 1. <i>Login</i> dengan menggunakan NID dan password. 2. Menekan tombol admin pada bagian atas aplikasi. 3. Menekan tombol <i>Roles</i> di bagian kiri aplikasi.
Hasil yang diperoleh	Data <i>Roles</i> pada aplikasi terlihat sebagai tabel
Hasil pengujian	Berhasil

Tabel 56. Spesifikasi pengujian menambah data *roles*

Menambah Data Roles	
Kode	UJ-054
Skenario	Menekan tombol tambah pada data <i>Roles</i> , menambahkan data-data <i>Roles</i> , dan melakukan simpan data <i>Roles</i> baru tersebut.
Kondisi awal	<ol style="list-style-type: none"> 2. Pengguna sudah melakukan <i>login</i> sebagai admin sistem.

Hasil yang diharapkan	Data <i>Roles</i> telah berhasil ditambahkan.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Mengakses halaman data <i>Roles</i> melalui menu aplikasi dengan menekan tombol <i>Roles</i> pada menu dibagian kiri halaman. 3. Menekan tombol tambah pada halaman data <i>Roles</i>. 4. Mengisikan data-data yang dibutuhkan dalam membuat satu data <i>Roles</i> baru. 5. Menekan tombol simpan.
Hasil yang diperoleh	Data <i>Roles</i> baru telah berhasil ditambahkan
Hasil pengujian	Berhasil

Tabel 57. Spesifikasi pengujian mengubah data *roles*

Mengubah Data Roles	
Kode	UJ-055
Skenario	Menekan tombol <i>Roles</i> pada menu bagian kiri aplikasi dan memilih salah satu baris data <i>Roles</i> untuk diubah datanya.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data <i>Roles</i> pada aplikasi
Hasil yang diharapkan	Data-data pada satu data <i>Roles</i> telah berhasil diubah dan

	perubahan tersebut telah disimpan oleh aplikasi.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol <i>Roles</i> pada menu sebelah kiri aplikasi. 3. Memilih salah satu data <i>Roles</i>. 4. Mengubah satu dan atau banyak data dari data <i>Roles</i> tersebut. 5. Menekan tombol simpan.
Hasil yang diperoleh	Data pada satu data <i>Roles</i> telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Hasil pengujian	Berhasil

Tabel 58. Spesifikasi pengujian menghapus data *roles*

Menghapus Data Roles	
Kode	UJ-056
Skenario	Menekan tombol <i>Roles</i> pada menu bagian kiri aplikasi, memilih satu atau banyak data <i>Roles</i> dengan menekan <i>checkbox</i> pada kolom paling kiri tabel, dan menekan tombol hapus di bagian atas tabel data <i>Roles</i> .
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem.

	2. Terdapat setidaknya satu data <i>Roles</i> pada aplikasi
Hasil yang diharapkan	Data <i>Roles</i> yang dipilih telah berhasil dihapus dari aplikasi.
L.angkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol <i>Roles</i> pada menu sebelah kiri aplikasi. 3. Memilih satu atau banyak data <i>Roles</i> yang ingin dihapus dengan menekan <i>checkbox</i> pada kolom paling kiri tabel. 4. Menekan tombol hapus di bagian atas tabel data <i>Roles</i>.
Hasil yang diperoleh	Data <i>Roles</i> yang dipilih telah berhasil dihapus dari aplikasi.
Hasil pengujian	Berhasil

Tabel 59. Spesifikasi pengujian melihat data RSTI

Melihat Data RSTI	
Kode	UJ-057
Skenario	Menekan tombol admin pada bagian atas aplikasi, lalu menekan tombol RSTI pada menu bagian kiri halaman admin.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data RSTI pada aplikasi.

Hasil yang diharapkan	Data RSTI pada aplikasi terlihat sebagai tabel.
Langkah pengujian	<ol style="list-style-type: none"> 1. <i>Login</i> dengan menggunakan NID dan password. 2. Menekan tombol admin pada bagian atas aplikasi. 3. Menekan tombol RSTI di bagian kiri aplikasi.
Hasil yang diperoleh	Data RSTI pada aplikasi terlihat sebagai tabel
Hasil pengujian	Berhasil

Tabel 60. Spesifikasi pengujian menambah data RSTI

Menambah Data RSTI	
Kode	UJ-058
Skenario	Menekan tombol tambah pada data RSTI, menambahkan data-data RSTI, dan melakukan simpan data RSTI baru tersebut.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem.
Hasil yang diharapkan	Data RSTI telah berhasil ditambahkan.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Mengakses halaman data RSTI melalui menu aplikasi dengan menekan tombol RSTI pada menu dibagian kiri halaman.

	<ol style="list-style-type: none"> 3. Menekan tombol tambah pada halaman data RSTI. 4. Mengisikan data-data yang dibutuhkan dalam membuat satu data RSTI baru. 5. Menekan tombol simpan.
Hasil yang diperoleh	Data RSTI baru telah berhasil ditambahkan
Hasil pengujian	Berhasil

Tabel 61. Spesifikasi pengujian mengubah data RSTI

Mengubah Data RSTI	
Kode	UJ-059
Skenario	Menekan tombol RSTI pada menu bagian kiri aplikasi dan memilih salah satu baris data RSTI untuk diubah datanya.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data RSTI pada aplikasi
Hasil yang diharapkan	Data-data pada satu data RSTI telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol RSTI pada menu sebelah kiri aplikasi. 3. Memilih salah satu data RSTI.

	<p>4. Mengubah satu dan atau banyak data dari data RSTI tersebut.</p> <p>5. Menekan tombol simpan.</p>
Hasil yang diperoleh	Data pada satu data RSTI telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Hasil pengujian	Berhasil

Tabel 62. Spesifikasi pengujian menghapus data RSTI

Menghapus Data RSTI	
Kode	UJ-060
Skenario	Menekan tombol RSTI pada menu bagian kiri aplikasi, memilih satu atau banyak data RSTI dengan menekan <i>checkbox</i> pada kolom paling kiri tabel, dan menekan tombol hapus di bagian atas tabel data RSTI.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data RSTI pada aplikasi
Hasil yang diharapkan	Data RSTI yang dipilih telah berhasil dihapus dari aplikasi.
L.angkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol RSTI pada menu sebelah kiri aplikasi

	<ol style="list-style-type: none"> 3. Memilih satu atau banyak data RSTI yang ingin dihapus dengan menekan <i>checkbox</i> pada kolom paling kiri tabel. 4. Menekan tombol hapus di bagian atas tabel data RSTI.
Hasil yang diperoleh	Data RSTI yang dipilih telah berhasil dihapus dari aplikasi.
Hasil pengujian	Berhasil

Tabel 63. Spesifikasi pengujian melihat data *service catalog*

Melihat Data <i>Service Catalog</i>	
Kode	UJ-061
Skenario	Menekan tombol admin pada bagian atas aplikasi, lalu menekan tombol <i>service catalog</i> pada menu bagian kiri halaman admin.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data <i>service catalog</i> pada aplikasi.
Hasil yang diharapkan	Data <i>service catalog</i> pada aplikasi terlihat sebagai tabel.
Langkah pengujian	<ol style="list-style-type: none"> 1. <i>Login</i> dengan menggunakan NID dan password. 2. Menekan tombol admin pada bagian atas aplikasi.

	3. Menekan tombol <i>service catalog</i> di bagian kiri aplikasi.
Hasil yang diperoleh	Data <i>service catalog</i> pada aplikasi terlihat sebagai tabel
Hasil pengujian	Berhasil

Tabel 64. Spesifikasi pengujian menambah data *service catalog*

Menambah Data <i>Service Catalog</i>	
Kode	UJ-062
Skenario	Menekan tombol tambah pada data <i>service catalog</i> , menambahkan data-data <i>service catalog</i> , dan melakukan simpan data <i>service catalog</i> baru tersebut.
Kondisi awal	1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem.
Hasil yang diharapkan	Data <i>service catalog</i> telah berhasil ditambahkan.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Mengakses halaman data <i>service catalog</i> melalui menu aplikasi dengan menekan tombol <i>service catalog</i> pada menu dibagian kiri halaman. 3. Menekan tombol tambah pada halaman data <i>service catalog</i>. 4. Mengisikan data-data yang dibutuhkan dalam membuat

	satu data <i>service catalog</i> baru. 5. Menekan tombol simpan.
Hasil yang diperoleh	Data <i>service catalog</i> baru telah berhasil ditambahkan
Hasil pengujian	Berhasil

Tabel 65. Spesifikasi pengujian mengubah data *service catalog*

Mengubah Data <i>Service Catalog</i>	
Kode	UJ-063
Skenario	Menekan tombol <i>service catalog</i> pada menu bagian kiri aplikasi dan memilih salah satu baris data <i>service catalog</i> untuk diubah datanya.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data <i>service catalog</i> pada aplikasi
Hasil yang diharapkan	Data-data pada satu data <i>service catalog</i> telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol <i>service catalog</i> pada menu sebelah kiri aplikasi. 3. Memilih salah satu data <i>service catalog</i>.

	<p>4. Mengubah satu dan atau banyak data dari data <i>service catalog</i> tersebut.</p> <p>5. Menekan tombol simpan.</p>
Hasil yang diperoleh	Data pada satu data <i>service catalog</i> telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Hasil pengujian	Berhasil

Tabel 66. Spesifikasi pengujian menghapus data *service catalog*

Menghapus Data <i>Service Catalog</i>	
Kode	UJ-064
Skenario	Menekan tombol <i>service catalog</i> pada menu bagian kiri aplikasi, memilih satu atau banyak data <i>service catalog</i> dengan menekan <i>checkbox</i> pada kolom paling kiri tabel, dan menekan tombol hapus di bagian atas tabel data <i>service catalog</i> .
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data <i>service catalog</i> pada aplikasi
Hasil yang diharapkan	Data <i>service catalog</i> yang dipilih telah berhasil dihapus dari aplikasi.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi.

	<ol style="list-style-type: none"> 2. Menekan tombol <i>service catalog</i> pada menu sebelah kiri aplikasi 3. Memilih satu atau banyak data <i>service catalog</i> yang ingin dihapus dengan menekan <i>checkbox</i> pada kolom paling kiri tabel. 4. Menekan tombol hapus di bagian atas tabel data <i>service catalog</i>.
Hasil yang diperoleh	Data <i>service catalog</i> yang dipilih telah berhasil dihapus dari aplikasi.
Hasil pengujian	Berhasil

Tabel 67. Spesifikasi pengujian melihat data status

Melihat Data Status	
Kode	UJ-065
Skenario	Menekan tombol admin pada bagian atas aplikasi, lalu menekan tombol status pada menu bagian kiri halaman admin.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data status pada aplikasi.
Hasil yang diharapkan	Data status pada aplikasi terlihat sebagai tabel.
Langkah pengujian	<ol style="list-style-type: none"> 1. <i>Login</i> dengan menggunakan NID dan password.

	<ol style="list-style-type: none"> 2. Menekan tombol admin pada bagian atas aplikasi. 3. Menekan tombol status di bagian kiri aplikasi.
Hasil yang diperoleh	Data status pada aplikasi terlihat sebagai tabel
Hasil pengujian	Berhasil

Tabel 68. Spesifikasi pengujian menambah data status

Menambah Data Status	
Kode	UJ-066
Skenario	Menekan tombol tambah pada data status, menambahkan data-data status, dan melakukan simpan data status baru tersebut.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem.
Hasil yang diharapkan	Data status telah berhasil ditambahkan.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Mengakses halaman data status melalui menu aplikasi dengan menekan tombol status pada menu dibagian kiri halaman. 3. Menekan tombol tambah pada halaman data status. 4. Mengisikan data-data yang dibutuhkan dalam membuat satu data status baru. 5. Menekan tombol simpan.

Hasil yang diperoleh	Data status baru telah berhasil ditambahkan
Hasil pengujian	Berhasil

Tabel 69. Spesifikasi pengujian mengubah data status

Mengubah Data Status	
Kode	UJ-067
Skenario	Menekan tombol status pada menu bagian kiri aplikasi dan memilih salah satu baris data status untuk diubah datanya.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data status pada aplikasi
Hasil yang diharapkan	Data-data pada satu data status telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol status pada menu sebelah kiri aplikasi. 3. Memilih salah satu data status. 4. Mengubah satu dan atau banyak data dari data status tersebut. 5. Menekan tombol simpan.

Hasil yang diperoleh	Data pada satu data status telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Hasil pengujian	Berhasil

Tabel 70. Spesifikasi pengujian menghapus data status

Menghapus Data Status	
Kode	UJ-068
Skenario	Menekan tombol status pada menu bagian kiri aplikasi, memilih satu atau banyak data status dengan menekan <i>checkbox</i> pada kolom paling kiri tabel, dan menekan tombol hapus di bagian atas tabel data status.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data status pada aplikasi
Hasil yang diharapkan	Data status yang dipilih telah berhasil dihapus dari aplikasi.
L.angkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol status pada menu sebelah kiri aplikasi 3. Memilih satu atau banyak data status yang ingin dihapus dengan menekan <i>checkbox</i> pada kolom paling kiri tabel.

	4. Menekan tombol hapus di bagian atas tabel data status.
Hasil yang diperoleh	Data status yang dipilih telah berhasil dihapus dari aplikasi.
Hasil pengujian	Berhasil

Tabel 71. Spesifikasi pengujian melihat data *stream*

Melihat Data Stream	
Kode	UJ-069
Skenario	Menekan tombol admin pada bagian atas aplikasi, lalu menekan tombol <i>stream</i> pada menu bagian kiri halaman admin.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data <i>stream</i> pada aplikasi.
Hasil yang diharapkan	Data <i>stream</i> pada aplikasi terlihat sebagai tabel.
Langkah pengujian	<ol style="list-style-type: none"> 1. <i>Login</i> dengan menggunakan NID dan password. 2. Menekan tombol admin pada bagian atas aplikasi. 3. Menekan tombol <i>stream</i> di bagian kiri aplikasi.
Hasil yang diperoleh	Data <i>stream</i> pada aplikasi terlihat sebagai tabel
Hasil pengujian	Berhasil

Tabel 72. Spesifikasi pengujian menambah data *stream*

Menambah Data Stream	
Kode	UJ-070
Skenario	Menekan tombol tambah pada data <i>stream</i> , menambahkan data-data <i>stream</i> , dan melakukan simpan data <i>stream</i> baru tersebut.
Kondisi awal	1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem.
Hasil yang diharapkan	Data <i>stream</i> telah berhasil ditambahkan.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Mengakses halaman data <i>stream</i> melalui menu aplikasi dengan menekan tombol <i>stream</i> pada menu dibagian kiri halaman. 3. Menekan tombol tambah pada halaman data <i>stream</i>. 4. Mengisikan data-data yang dibutuhkan dalam membuat satu data <i>stream</i> baru. 5. Menekan tombol simpan.
Hasil yang diperoleh	Data <i>stream</i> baru telah berhasil ditambahkan
Hasil pengujian	Berhasil

Tabel 73. Spesifikasi pengujian mengubah data *stream*

Mengubah Data Stream	
Kode	UJ-071

Skenario	Menekan tombol <i>stream</i> pada menu bagian kiri aplikasi dan memilih salah satu baris data <i>stream</i> untuk diubah datanya.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data <i>stream</i> pada aplikasi
Hasil yang diharapkan	Data-data pada satu data <i>stream</i> telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol <i>stream</i> pada menu sebelah kiri aplikasi. 3. Memilih salah satu data <i>stream</i>. 4. Mengubah satu dan atau banyak data dari data <i>stream</i> tersebut. 5. Menekan tombol simpan.
Hasil yang diperoleh	Data pada satu data <i>stream</i> telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Hasil pengujian	Berhasil

Tabel 74. Spesifikasi pengujian menghapus data *stream*

Menghapus Data Stream	
Kode	UJ-072

Skenario	Menekan tombol <i>stream</i> pada menu bagian kiri aplikasi, memilih satu atau banyak data <i>stream</i> dengan menekan <i>checkbox</i> pada kolom paling kiri tabel, dan menekan tombol hapus di bagian atas tabel data <i>stream</i> .
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data <i>stream</i> pada aplikasi
Hasil yang diharapkan	Data <i>stream</i> yang dipilih telah berhasil dihapus dari aplikasi.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol <i>stream</i> pada menu sebelah kiri aplikasi 3. Memilih satu atau banyak data <i>stream</i> yang ingin dihapus dengan menekan <i>checkbox</i> pada kolom paling kiri tabel. 4. Menekan tombol hapus di bagian atas tabel data <i>stream</i>.
Hasil yang diperoleh	Data <i>stream</i> yang dipilih telah berhasil dihapus dari aplikasi.
Hasil pengujian	Berhasil

Tabel 75. Spesifikasi pengujian melihat data sumber program

Melihat Data Sumber Program

Kode	UJ-073
Skenario	Menekan tombol admin pada bagian atas aplikasi, lalu menekan tombol sumber program pada menu bagian kiri halaman admin.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data sumber program pada aplikasi.
Hasil yang diharapkan	Data sumber program pada aplikasi terlihat sebagai tabel.
Langkah pengujian	<ol style="list-style-type: none"> 1. <i>Login</i> dengan menggunakan NID dan password. 2. Menekan tombol admin pada bagian atas aplikasi. 3. Menekan tombol sumber program di bagian kiri aplikasi.
Hasil yang diperoleh	Data sumber program pada aplikasi terlihat sebagai tabel
Hasil pengujian	Berhasil

Tabel 76. Spesifikasi pengujian menambah data sumber program

Menambah Data Sumber Program	
Kode	UJ-074
Skenario	Menekan tombol tambah pada data sumber program, menambahkan data-data sumber program, dan

	melakukan simpan data sumber program baru tersebut.
Kondisi awal	1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem.
Hasil yang diharapkan	Data sumber program telah berhasil ditambahkan.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Mengakses halaman data sumber program melalui menu aplikasi dengan menekan tombol sumber program pada menu dibagian kiri halaman. 3. Menekan tombol tambah pada halaman data sumber program. 4. Mengisikan data-data yang dibutuhkan dalam membuat satu data sumber program baru. 5. Menekan tombol simpan.
Hasil yang diperoleh	Data sumber program baru telah berhasil ditambahkan
Hasil pengujian	Berhasil

Tabel 77. Spesifikasi pengujian mengubah data sumber program

Mengubah Data Sumber Program	
Kode	UJ-075
Skenario	Menekan tombol sumber program pada menu bagian kiri aplikasi dan memilih salah satu baris data sumber

	program untuk diubah datanya.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data sumber program pada aplikasi
Hasil yang diharapkan	Data-data pada satu data sumber program telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Langkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol sumber program pada menu sebelah kiri aplikasi. 3. Memilih salah satu data sumber program. 4. Mengubah satu dan atau banyak data dari data sumber program tersebut. 5. Menekan tombol simpan.
Hasil yang diperoleh	Data pada satu data sumber program telah berhasil diubah dan perubahan tersebut telah disimpan oleh aplikasi.
Hasil pengujian	Berhasil

Tabel 78. Spesifikasi pengujian menghapus data sumber program

Menghapus Data Sumber Program	
Kode	UJ-076
Skenario	Menekan tombol sumber program pada menu bagian

	kiri aplikasi, memilih satu atau banyak data sumber program dengan menekan <i>checkbox</i> pada kolom paling kiri tabel, dan menekan tombol hapus di bagian atas tabel data sumber program.
Kondisi awal	<ol style="list-style-type: none"> 1. Pengguna sudah melakukan <i>login</i> sebagai admin sistem. 2. Terdapat setidaknya satu data sumber program pada aplikasi
Hasil yang diharapkan	Data sumber program yang dipilih telah berhasil dihapus dari aplikasi.
L.angkah pengujian	<ol style="list-style-type: none"> 1. Menekan tombol admin pada bagian atas aplikasi. 2. Menekan tombol sumber program pada menu sebelah kiri aplikasi 3. Memilih satu atau banyak data sumber program yang ingin dihapus dengan menekan <i>checkbox</i> pada kolom paling kiri tabel. 4. Menekan tombol hapus di bagian atas tabel data sumber program.
Hasil yang diperoleh	Data sumber program yang dipilih telah berhasil dihapus dari aplikasi.
Hasil pengujian	Berhasil

BAB V

KESIMPULAN

Setelah melakukan kerja praktik selama enam bulan, didapatkan beberapa kesimpulan dan saran untuk pengembangan aplikasi BTIF Monitoring.

5.1. Kesimpulan

Kesimpulan yang didapat dari pengembangan aplikasi BTIF Monitoring selama kerja praktik ini adalah sebagai berikut.

1. Aplikasi yang penulis buat telah dilakukan *testing*, dan berdasarkan *testing* tersebut, pengguna sudah cukup terpenuhi kebutuhannya, sesuai dengan yang dibutuhkan dari aplikasi yang dibuat.
2. Aplikasi dibuat sudah sesuai dengan tahapan dari *Software Development Life Cycle*, dimana dimulai dari perencanaan, analisis, desain, implementasi, *deployment*, dan *testing*. Tahapan tersebut dapat dilihat pada bab 4.
3. Konsep penggunaan arsitektur yang berbasis container mempercepat proses penyiapan *environment development*, karena sudah tidak melakukan instalasi secara manual, melainkan menggunakan docker sebagai alat yang digunakan untuk mengatur keseluruhan proses saat dari kode sampai menjadi *container*. Dengan menggunakan docker penulis sangat terbantu dalam pembuatan aplikasi BTIF Monitoring dan mempercepat waktu pengembangan aplikasi dan *deployment* aplikasi.

5.2. Saran

Berikut adalah saran yang dapat diberikan penulis setelah menjalani kerja praktik ini.

1. Aplikasi BTIF Monitoring masih berada dalam tahapan *development*, dimana hanya sampai pada tahapan *testing*, dikarenakan kebutuhan yang masih belum tergali secara sempurna dari pengguna aplikasinya. Untuk kedepannya, aplikasi BTIF Monitoring dapat dikembangkan lebih baik lagi sehingga

- fungsionalitasnya semakin lengkap dalam memenuhi kebutuhan pengguna aplikasinya.
2. Aplikasi BTIF Monitoring belum digunakan oleh bidang BTIF PT PJB, karena masih sampai pada tahapan *testing* aplikasinya. Kedepannya, penulis berharap aplikasi BTIF Monitoring dapat segera digunakan untuk membantu *monitoring* program kerja di bidang BTIF PT PJB.

DAFTAR PUSTAKA

- [1] “TENTANG KAMI,” *PT Pembangunan Jawa-Bali*.
[//www.ptpjb.com/tentang-kami/](http://www.ptpjb.com/tentang-kami/) (accessed Jul. 25, 2021).
- [2] “Introduction to HTML.”
https://www.w3schools.com/html/html_intro.asp (accessed Jul. 25, 2021).
- [3] “Pengertian PHP, Fungsi dan Sintaks Dasarnya,” *Niagahoster Blog*, Nov. 02, 2020.
<https://www.niagahoster.co.id/blog/pengertian-php/> (accessed Jul. 08, 2021).
- [4] admin, “Apa itu *Framework* Codeigniter? | Program Studi Teknologi Informasi.”
<https://psti.unisayogya.ac.id/2020/10/14/apa-itu-framework-codeigniter/> (accessed Jul. 25, 2021).
- [5] arisg, “Pengertian REST API – PT Proweb Indonesia.”
<https://www.proweb.co.id/articles/restful/restful-api.html>
(accessed Jul. 25, 2021).
- [6] “Pengertian MySQL, Fungsi, dan Cara Kerjanya (Lengkap),” *Niagahoster Blog*, Jul. 24, 2019.
<https://www.niagahoster.co.id/blog/mysql-adalah/> (accessed Jul. 25, 2021).
- [7] “Postman | The Collaboration Platform for API Development,” *Postman*. <https://www.postman.com/> (accessed Jul. 25, 2021).
- [8] “What is a Container? | App Containerization | Docker.”
<https://www.docker.com/resources/what-container> (accessed Jul. 25, 2021).
- [9] “What is Docker?,” *Opensource.com*.
<https://opensource.com/resources/what-docker> (accessed Jul. 25, 2021).
- [10] “How To Install and Use Docker Compose on Ubuntu 20.04,” *DigitalOcean*.
<https://www.digitalocean.com/community/tutorials/how-to-install-and-use-docker-compose-on-ubuntu-20-04> (accessed Jul. 25, 2021).

- [11] “What are microservices?” *microservices.io*.
<http://microservices.io/index.html> (accessed Jul. 25, 2021).

BIODATA PENULIS I

Nama : Hisam Widi Prayoga
Tempat, Tanggal Lahir : Kudus, 3 Agustus 1999
Jenis Kelamin : Laki-laki
Agama : Islam
Status : Belum Menikah
Alamat Asal : Jln. Madukoro No.13, Kec. Bae, Kab.
Kudus, Jawa Tengah
Alamat Surabaya : Keputih 2A
Telepon : +62 823-9914-6886
Email : hwprayoga@gmail.com

PENDIDIKAN FORMAL

2017 – sekarang : Mahasiswa S1 Informatika ITS
2014 – 2017 : SMA Negeri 1 Kudus
2011 – 2014 : SMP Negeri 1 Kudus
2005 – 2011 : SD Negeri 1 Melati Lor

KEMAMPUAN

- *Web Programming* (HTML, PHP, CSS, Javascript, Codeigniter, Laravel, ReactJS)
- *Mobile Programming* (React Native, Java, Kotlin)
- *Programming* (Python, Javascript, C++)
- *Machine Learning*
- Bahasa (Indonesia, Inggris)

AKADEMIS

Kuliah : Departemen Informatika, Fakultas Teknologi
Elektro dan Informatika Cerdas, Institut
Teknologi Sepuluh Nopember Surabaya
Angkatan : 2017
Semester : 8 (Delapan)
IPK : 3,56 (Semester 8)

BIODATA PENULIS II

Nama : Hendra Ramadani
 Tempat, Tanggal Lahir : Tulungagung, 29 Oktober 1998
 Jenis Kelamin : Laki-laki
 Agama : Islam
 Status : Belum Menikah
 Alamat Asal : Dsn. Batan Kec. Kauman RT 4 RW 1,
 Kab. Tulungagung, Jawa Timur
 Alamat Surabaya : Keputih Makam IX no. 20
 Telepon : +62 812-3027-5840
 Email : hendraramadani14@gmail.com

PENDIDIKAN FORMAL

2017 – sekarang : Mahasiswa S1 Informatika ITS
 2014 – 2017 : SMK Negeri 3 Boyolangu
 2011 – 2014 : SMP Negeri 1 Kauman
 2005 – 2011 : SD Negeri Kalangbret

KEMAMPUAN

- *Web Programming* (HTML, PHP, CSS, Javasript, Codeigniter, Laravel, ReactJS)
- *Mobile Programming* (React Native)
- *Programming* (Python, Javascript, C++)
- Desain Interface (*UI/UX*)
- Bahasa (Indonesia, Inggris)

AKADEMIS

Kuliah : Departemen Informatika, Fakultas Teknologi
 Elektro dan Informatika Cerdas, Institut
 Teknologi Sepuluh Nopember Surabaya
 Angkatan : 2017
 Semester : 8 (Delapan)
 IPK : 3,55 (Semester 8)