



**PENGEMBANGAN DASHBOARD DAN TRACKER PADA APLIKASI
BEASISWA UNTUK PEMANTAUAN PERKEMBANGAN MAHASISWA
PENERIMA BEASISWA DI INSTITUSI PENDIDIKAN XYZ**

TUGAS AKHIR

Mochammad Yoga Prasetya

2020081071

**PROGRAM STUDI SISTEM INFORMASI
INSTITUSI PENDIDIKAN XYZ
FAKULTAS TEKNOLOGI DAN DESAIN
SISTEM INFORMASI
2022/2023**

DAFTAR ISI

| | |
|--|-------------------------------------|
| DAFTAR ISI..... | 2 |
| BAB I PENDAHULUAN..... | 4 |
| 1.1 Latar Belakang Masalah | 4 |
| 1.2 Identifikasi dan Rumus Masalah | 6 |
| 1.2.1 Identifikasi Masalah..... | 6 |
| 1.2.2 Rumusan Masalah..... | 6 |
| 1.3 Ruang Lingkup dan Batas Masalah..... | 6 |
| 1.3.1 Ruang Lingkup..... | 7 |
| 1.3.2 Batasan Masalah | 7 |
| 1.4 Tujuan Penelitian | 7 |
| 1.5 Manfaat Penelitian | 8 |
| 1.6 Sistematika Penulisan | 9 |
| BAB II TINJAUAN PUSTAKA..... | 10 |
| 2.1 Teori Dasar | 10 |
| 2.2.1 Aplikasi | 10 |
| 2.2.2 Aplikasi Beasiswa | 11 |
| 2.2.3 Beasiswa | 11 |
| 2.2.4 Tracker | 12 |
| 2.2.5 Dashboard | 13 |
| 2.2.6 Javascript..... | 13 |
| 2.2.7 PHP..... | 14 |
| 2.2.8 Database | 15 |
| 2.2.9 MySql..... | 15 |
| 2.2.10 Laravel | 16 |
| 2.2.11 Software Development Life Cycle (SDLC)..... | 16 |
| 2.2.12 Rapid Application Development (RAD) | 16 |
| 2.2.13 Unified Modelling Language (UML). | 17 |
| 2.2.14 Black Box Testing..... | 22 |
| 2.1 Tinjauan Studi | 23 |
| BAB III METODE PENELITIAN | 26 |
| 3.1 Objek Penelitian | 26 |
| 4.4.1 Latar Belakang Objek Penelitian..... | 26 |
| 4.4.2 Visi & Misi Objek Penelitian | Error! Bookmark not defined. |
| 4.4.3 Struktur Organisasi..... | 27 |

| | | |
|--------------------------|--------------------------------------|-------------------------------------|
| 3.2 | Metode Pengembangan Sistem | 27 |
| 3.2.1 | Requirements Planning | Error! Bookmark not defined. |
| 3.2.2 | User Design | Error! Bookmark not defined. |
| 3.2.3 | Rapid Construction | Error! Bookmark not defined. |
| 3.2.4 | Cutover | Error! Bookmark not defined. |
| 3.2.5 | Implementasi | Error! Bookmark not defined. |
| 3.3 | Analisis Sistem Yang Berjalan | 30 |
| 3.4 | Analisis Dokumen | Error! Bookmark not defined. |
| 3.5 | Analisis Kebutuhan | 34 |
| 3.4.1 | Analisis Kebutuhan Sistem | 34 |
| 3.4.2 | Kebutuhan Perangkat | Error! Bookmark not defined. |
| BAB IV | BIAYA DAN JADWAL KEGIATAN | 43 |
| 4.1 | Tahap Perencanaan | 43 |
| 4.2 | Tahap Analisis | Error! Bookmark not defined. |
| 4.2.1 | Analisis Sistem Berjalan | Error! Bookmark not defined. |
| 4.2.2 | Analisis Kebutuhan Sistem | Error! Bookmark not defined. |
| 4.2.3 | Analisis Perancangan Sistem | Error! Bookmark not defined. |
| 4.3 | Tahap Perencanaan Sistem | 43 |
| 4.3.1 | Use Case Diagram | 44 |
| 4.3.2 | Use Case Description | 44 |
| 4.3.3 | Activity Diagram | 48 |
| 4.3.4 | Sequence Diagram | 48 |
| 4.3.5 | Class Diagram | 48 |
| 4.3.6 | Spesifikasi Basis Data | Error! Bookmark not defined. |
| 4.4 | Tahap Perencanaan Sistem | 49 |
| 4.4.1 | Implementasi Sistem | 55 |
| 4.4.2 | Skenario Pengujian | 56 |
| BAB V | PENUTUP | 58 |
| DAFTAR PUSTAKA | | 59 |
| LAMPIRAN-LAMPIRAN | | 61 |

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Perkembangan teknologi telah meresap ke dalam berbagai aspek kehidupan modern, termasuk dalam dunia pendidikan tinggi. Institusi pendidikan, khususnya universitas, telah mengadopsi dan memanfaatkan teknologi secara luas untuk meningkatkan kualitas pendidikan, penelitian, dan administrasi. Salah satu bidang yang terpengaruh secara signifikan adalah dunia pendidikan tinggi. Institusi-institusi pendidikan, terutama universitas, telah mengambil langkah maju dengan memanfaatkan kemajuan teknologi untuk meningkatkan kualitas pendidikan dan layanan yang mereka tawarkan. Salah satu contoh penerapan teknologi yang menonjol adalah dalam pengelolaan program beasiswa. Transformasi ini tidak hanya memungkinkan akses yang lebih mudah, tetapi juga meningkatkan efisiensi proses seleksi serta memberikan pengalaman pendidikan yang lebih beragam dan kaya. Dalam tulisan ini, kita akan mengeksplorasi bagaimana institusi pendidikan, khususnya universitas, telah beradaptasi dengan perkembangan teknologi melalui penerapan teknologi dalam program beasiswa mereka.

Beasiswa adalah bentuk bantuan keuangan yang diberikan kepada individu seperti mahasiswa atau pelajar untuk membantu mereka menanggung biaya pendidikan atau riset. Biasanya, beasiswa diberikan oleh lembaga pendidikan, organisasi, pemerintah, perusahaan, atau individu dengan tujuan mendukung perkembangan akademik atau non-akademik individu yang memiliki potensi namun mengalami keterbatasan finansial. Tujuan utama beasiswa adalah untuk membantu mengurangi beban biaya pendidikan, buku, peralatan, biaya hidup, dan berbagai kegiatan pendidikan lainnya.

Institusi Pendidikan XYZ memiliki beragam program beasiswa, termasuk beasiswa prestasi jaya, beasiswa ayo sekolah ayo kuliah (ASAK), XYZ untuk negeri, dan beasiswa unggulan. Semua mahasiswa yang menunjukkan prestasi atau potensi baik dalam bidang akademik maupun non-akademik berhak untuk mendapatkan beasiswa ini. Pengelolaan seluruh program beasiswa di XYZ ditangani oleh BKAL (Biro Kemahasiswaan dan Alumni), yang bertugas mengelola dan mengurus berbagai urusan yang berkaitan dengan mahasiswa selama masa studi mereka dan setelah lulus.

Pengembangan dashboard dan tracker pada aplikasi beasiswa di Institusi Pendidikan XYZ dimulai terhadap kebutuhan dan tujuan pemantauan. Pengembang mengidentifikasi data yang relevan seperti jumlah beasiswa, status (berjalan, tertunda, selesai), serta merancang antarmuka dashboard untuk memudahkan stakeholder memantau data. Tracker memungkinkan BKAL mengikuti status beasiswa secara sistematis, memantau progres mahasiswa, dan memberikan dukungan yang tepat waktu. Integrasi sistem memungkinkan pengumpulan dan penyajian data secara real-time, meningkatkan responsivitas dalam memantau perkembangan mahasiswa. Dengan ini, diharapkan BKAL dapat lebih efektif dalam mendukung kesuksesan akademis dan karir mahasiswa penerima beasiswa.

Terdapat beberapa kendala yang dihadapi oleh Biro Kemahasiswaan dan Alumni (BKAL), pengelolaan dan pelaporan merupakan dua aspek terpisah dari proses aplikasi beasiswa di Biro Kemahasiswaan dan Alumni (BKAL). Pengelolaan merujuk pada proses mengumpulkan, menyimpan, dan mengelola data terkait mahasiswa penerima beasiswa, sementara Pelaporan merujuk pada proses menyusun dan menyajikan informasi tersebut dalam bentuk laporan yang dapat digunakan untuk analisis dan pengambilan keputusan.

Saat ini, kedua proses tersebut menggunakan sistem atau aplikasi yang berbeda. Misalnya, pengelolaan data mahasiswa penerima beasiswa dapat dilakukan melalui basis data atau aplikasi pengelolaan mahasiswa yang terpisah dari sistem pelaporan yang digunakan untuk menyusun laporan kemajuan akademik atau analisis keuangan. Integrasi yang diinginkan adalah menggabungkan kedua proses tersebut menjadi satu kesatuan dalam aplikasi beasiswa yang terinci. Ini berarti bahwa aplikasi beasiswa akan mencakup fitur-fitur untuk pengelolaan data mahasiswa penerima beasiswa (misalnya, mencatat informasi pribadi, riwayat akademik, dan detail penerimaan beasiswa) serta kemampuan untuk menghasilkan laporan-laporan yang diperlukan (misalnya, laporan kemajuan akademik, laporan partisipasi kegiatan, dan laporan keuangan). Dengan integrasi ini, BKAL akan memiliki satu platform yang menyatukan semua aspek administrasi beasiswa dalam satu tempat. Ini akan memungkinkan proses pengelolaan data menjadi lebih efisien dan akurat karena informasi dapat diakses dan dikelola dalam satu sistem. Selain itu, proses pelaporan akan menjadi lebih terstruktur dan konsisten karena data yang sama dapat digunakan untuk berbagai keperluan pelaporan. Dengan demikian, integrasi ini akan menghasilkan satu proses yang terintegrasi untuk pengelolaan dan pelaporan data beasiswa,

Selain itu, BKAL membutuhkan suatu tool pelacakan untuk memantau perkembangan mahasiswa penerima beasiswa. Kemudian, diperlukan pula pembuatan dashboard yang berisi informasi tentang jumlah beasiswa, status beasiswa (berjalan, tertunda, selesai), sebagai gambaran umum perkembangan beasiswa. dengan mengatasi kendala-kendala ini, diharapkan BKAL dapat meningkatkan efisiensi operasional mereka dan memberikan layanan yang lebih baik kepada mahasiswa penerima beasiswa. Dengan latar belakang tersebut maka penulis menuangkannya dalam bentuk riset yang dijadikan bahan untuk penelitian yang berjudul **“PENGEMBANGAN DASHBOARD DAN TRACKER PADA APLIKASI BEASISWA UNTUK PEMANTAUAN PERKEMBANGAN MAHASISWA PENERIMA BEASISWA DI INSTITUSI PENDIDIKAN XYZ”**

1.2 Identifikasi dan Rumus Masalah

1.2.1 Identifikasi Masalah

Berdasarkan Latar Belakang yang ada maka masalah-masalah yang dapat di identifikasikan yaitu:

- 1) Pengelolaan dan Pelaporan yang belum menjadi satu proses yang menyebabkan ketidakakuratan data atau tidak terupdate dapat muncul akibat kurangnya integrasi, mempengaruhi keputusan beasiswa.
- 2) Untuk BKAL membutuhkan suatu tools pelacakan untuk mengetahui tentang perkembangan mahasiswa penerima beasiswa
- 3) Membutuhkan dashboard yang berisi jumlah beasiswa, beasiswa berjalan, beasiswa tertunda, beasiswa selesai sebagai gambaran umum perkembangan beasiswa.

1.2.2 Rumusan Masalah

Dari identifikasi masalah tersebut, maka dapat diambil rumusan masalah yaitu “Bagaimana mengembangkan dashboard dan tracker pada aplikasi beasiswa untuk pemantauan perkembangan mahasiswa penerima beasiswa di institusi pendidikan xyz?”

1.3 Ruang Lingkup dan Batas Masalah

Pengembangan aplikasi ini melibatkan beberapa ruang lingkup dan batasan yang perlu dipertimbangkan agar tidak melenceng dari sasaran pengembangan.

1.3.1 Ruang Lingkup

Berikut merupakan ruang lingkup dari penelitian:

- Pengembangan aplikasi beasiswa terdapat modul tracer yang berfungsi memvisualisasikan data progres laporan yang sudah dilakukan mahasiswa dalam bentuk grafik linear.
- Pengembangan dashboard pelaporan beasiswa yang berfungsi sebagai visualisasi jumlah beasiswa, beasiswa berjalan, beasiswa tertunda, beasiswa selesai dan dijadikan menjadi bentuk grafik batang.
- Pengembangan dapat melakukan chat real-time untuk memfasilitasi interaksi antara mahasiswa penerima beasiswa dengan assessor yang bertanggung jawab dalam mendukung perkembangan mereka.
- Terdapat rekap laporan jumlah beasiswa, beasiswa berjalan, beasiswa tertunda, beasiswa selesai untuk memudahkan pengguna, terutama BKAL, dalam memantau laporan-laporan yang masih tertunda dan yang lain.

1.3.2 Batasan Masalah

- Aplikasi yang dirancang hanya berfokus pada pemantauan dan *tracker* (pelacak) mahasiswa penerima beasiswa yang ada di Institusi Pendidikan XYZ.
- Aplikasi tidak dapat manajemen akun mahasiswa penerima beasiswa.
- Aplikasi tidak dapat atau kemampuan memberikan penilaian terhadap mahasiswa penerima beasiswa.
- Aplikasi ini hanya bisa diakses oleh Biro Kemahasiswaan dan Alumni (BKAL)

1.4 Tujuan Penelitian

Maksud dan tujuan penelitian ini adalah untuk merancang dan membangun sebuah sistem integrasi yang bertujuan utama untuk menjaga keakuratan data pada sistem informasi beasiswa di Institusi Pendidikan XYZ. Penelitian ini berfokus untuk meningkatkan efisiensi operasional universitas dengan mengotomatisasi dan mengintegrasikan proses pengelolaan dan pelaporan data beasiswa, sehingga

mengurangi potensi kesalahan administratif dan waktu yang diperlukan dalam pengelolaan data tersebut.

1.5 Manfaat Penelitian

Manfaat yang diperoleh dari penelitian dashboard dan tracker mahasiswa penerima beasiswa di Institusi Pendidikan XYZ adalah:

a) Bagi Peneliti:

Penelitian tentang dashboard dan tracker mahasiswa penerima beasiswa di Institusi Pendidikan XYZ memberikan kesempatan bagi peneliti untuk mendalami efektivitas program beasiswa. Mereka dapat memperoleh pemahaman yang mendalam tentang faktor-faktor yang memengaruhi kinerja akademik dan kesejahteraan mahasiswa penerima beasiswa, serta memberikan rekomendasi perbaikan program.

b) Bagi Mahasiswa:

Dashboard dan tracker memberikan mahasiswa kemampuan untuk memantau kemajuan akademik mereka dengan lebih baik. Mereka dapat melihat progres laporan mereka dalam bentuk grafik linear dan memahami status beasiswa mereka melalui visualisasi grafik batang. Ini membantu mahasiswa dalam memantau pencapaian mereka dan membuat keputusan yang lebih baik terkait pendidikan mereka.

c) Bagi Biro Kemahasiswaan dan Alumni:

Dashboard dan tracker memungkinkan biro kemahasiswaan untuk memberikan dukungan yang lebih efektif kepada mahasiswa. Dengan memantau progres laporan mahasiswa dan status beasiswa secara visual, biro kemahasiswaan dapat lebih mudah mengidentifikasi mahasiswa yang memerlukan bantuan tambahan atau dukungan khusus. Hal ini memungkinkan mereka untuk menyediakan layanan yang lebih tepat dan efisien kepada mahasiswa.

d) Bagi Institusi Pendidikan XYZ:

Penelitian ini membantu institusi meningkatkan transparansi dan akuntabilitas dalam pengelolaan program beasiswa, serta mengevaluasi dan meningkatkan efektivitasnya. Ini juga membantu dalam pengembangan kebijakan

yang lebih efektif dalam manajemen beasiswa, memperkuat reputasi institusi sebagai lembaga yang peduli terhadap kesejahteraan mahasiswa.

1.6 Sistematika Penulisan

Sistematika penulisan yang diterapkan di penelitian ini menjelaskan serta menggambarkan terkait penulisan pada penelitian yang akan dilaksanakan. Materi pada penulisan dalam penelitian ini terbagi menjadi 5 Bab, diantaranya:

BAB I

BAB 1 membahas mengenai permasalahan-permasalahan yang melatarbelakangi penelitian yang dilakukan, sehingga dapat memahami tujuan dan dasar dilakukannya suatu penelitian.

BAB II

Bab 2 ini memberikan berbagai macam penjelasan mengenai pemahaman dan ilmu yang memiliki keterkaitan terhadap penelitian. Teori-teori tersebut dijelaskan secara sistematis dengan memuat fakta, serta hasil penelitian sebelumnya, yang berasal dari pustaka.

BAB III

Bab 3 ini membahas tentang objek penelitian, metode penelitian, metode pengumpulan data, analisis sistem yang berjalan, analisis dokumen, metode pengembangan sistem dan analisis kebutuhan.

BAB IV

Bab 4 ini membahas mengenai hasil dari penelitian yang dilakukan untuk diungkapkan dalam bentuk narasi yang menggambarkan penelitian secara lengkap.

BAB V

Bab 5 ini membahas mengenai kesimpulan dari penelitian serta saran untuk proses pengembangan penelitian lebih lanjut.

BAB II

TINJAUAN PUSTAKA

2.1 Teori Dasar

2.2.1 Aplikasi

Aplikasi adalah suatu perangkat lunak yang dibuat khusus untuk memenuhi kebutuhan berbagai aktivitas dan pekerjaan, misalnya; pelayanan masyarakat, aktivitas niaga, periklanan, game, dan berbagai aktivitas lainnya Pramana (2021). Aplikasi secara umum adalah suatu proses dari cara manual yang ditransformasikan ke komputer dengan membuat sistem atau program agar data diolah lebih berdaya guna secara optimal. Menurut Dhanta (2009:32), aplikasi (application) adalah software yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu, misalnya Microsoft Word, dan Microsoft Excel. Sedangkan menurut Anisyah (2000:30), aplikasi adalah penerapan, penggunaan atau penambahan data.

Aplikasi, juga disebut sebagai program aplikasi atau perangkat lunak aplikasi, adalah paket perangkat lunak komputer yang melakukan fungsi tertentu secara langsung untuk pengguna akhir atau, dalam beberapa kasus, untuk aplikasi lain. Sebuah aplikasi dapat berdiri sendiri atau sekelompok program.

Frasa aplikasi berasal dari kata bahasa Inggris "application", yang berarti penggunaan, peringatan, atau penerapan. Sedangkan aplikasi adalah perangkat lunak siap pakai yang dirancang untuk melakukan suatu fungsi bagi pengguna layanan aplikasi dan penggunaan aplikasi lain oleh suatu sasaran yang akan dituju. Menurut leksikon komputer eksekutif. Aplikasi mengacu pada strategi pemecahan masalah yang memanfaatkan salah satu teknik pemrosesan data aplikasi dan biasanya berpacu dengan waktu untuk menyelesaikan perhitungan atau pemrosesan data yang diinginkan atau diharapkan (Huda, 2012). Selain itu aplikasi didefinisikan sebagai komponen yang berguna untuk pemrosesan data dan tugas lainnya. Contohnya termasuk pembuatan dokumen dan pemrosesan data (Mulyadi, 2010). Aplikasi adalah komponen komputer yang berinteraksi langsung dengan pengguna. Program beroperasi pada sistem operasi, dan untuk mengaktifkannya, kita harus menginstal sistem operasi terlebih dahulu (Wahyuni, T. D. , 2023).

2.2.2 Aplikasi Beasiswa

Aplikasi beasiswa adalah sebuah sistem atau platform digital yang dirancang untuk memudahkan proses pengelolaan dan penyaluran beasiswa kepada calon penerima. Aplikasi ini umumnya menyediakan berbagai fitur, seperti pengumpulan dan verifikasi data calon penerima, pencocokan kriteria beasiswa dengan profil pelamar, serta pelaporan dan pemantauan kemajuan penerima beasiswa. Dengan menggunakan aplikasi beasiswa, lembaga atau instansi yang menyelenggarakan program beasiswa dapat lebih efisien dalam mengelola proses seleksi dan pendistribusian dana beasiswa, sementara para pelamar dapat dengan mudah mengakses informasi tentang berbagai program beasiswa yang tersedia dan mengajukan permohonan secara online. Aplikasi ini juga membantu dalam meningkatkan transparansi dan akuntabilitas dalam penyaluran dana beasiswa serta memungkinkan penerima beasiswa untuk memanfaatkan kesempatan pendidikan dengan lebih baik (Hasanudin, 2020).

2.2.3 Beasiswa

Menurut Dicky Reynaldo dalam bukunya "Beasiswa: Panduan Mendapatkan Beasiswa Pendidikan Dalam dan Luar Negeri," beasiswa adalah bentuk bantuan keuangan dari pihak tertentu kepada individu atau kelompok yang memenuhi persyaratan tertentu untuk mendukung pendidikan. dalam bukunya "Kiat Meraih Beasiswa Pendidikan Dalam dan Luar Negeri," juga menggambarkan beasiswa sebagai bantuan keuangan dari lembaga tertentu kepada mereka yang memenuhi syarat untuk mendukung pendidikan atau penelitian akademik. Menurut Magdalena H. Lim dan Windya Sari (2018), beasiswa adalah dukungan finansial bagi individu atau kelompok yang memenuhi kriteria tertentu, dengan tujuan memperluas kesempatan pendidikan, memperkaya pengalaman akademik, dan mendukung pencapaian cita-cita atau tujuan karir.

Dari penjelasan tersebut dapat disimpulkan bahwa beasiswa adalah bentuk dukungan keuangan dari lembaga atau pihak lain kepada individu yang memenuhi syarat, seperti prestasi akademik, bakat, atau kebutuhan finansial, untuk membiayai pendidikan atau riset. Jenjang pendidikan yang dapat didukung oleh beasiswa mencakup dari pendidikan dasar hingga tinggi, dengan tujuan membantu individu mencapai tujuan akademik dan karir, serta mendorong penelitian dan pengembangan di berbagai bidang ilmu pengetahuan.

2.2.4 Tracker

Dalam konteks aplikasi, istilah "tracker" biasanya merujuk kepada suatu sistem atau komponen yang digunakan untuk melacak atau memantau aktivitas atau informasi tertentu. Penggunaan "tracker" dapat bervariasi tergantung pada jenis aplikasi dan kebutuhan spesifiknya. Berikut beberapa pengertian umum dari "tracker" dalam aplikasi

1. Tracker Pengguna (User Tracker)

Ini adalah jenis tracker yang digunakan untuk melacak perilaku pengguna dalam aplikasi. Ini dapat mencakup informasi seperti aktivitas browsing, tindakan yang diambil oleh pengguna, waktu yang dihabiskan di aplikasi, dan preferensi pengguna.

2. Tracker Kinerja (Performance Tracker)

Ini adalah jenis tracker yang digunakan untuk memantau kinerja aplikasi. Ini dapat melacak berbagai metrik kinerja, seperti waktu respons server, waktu pemuatan halaman, penggunaan memori, dan lain-lain. Data ini membantu pengembang dalam mengidentifikasi dan memperbaiki masalah kinerja.

3. Tracker Analitik (Analytics Tracker)

Ini adalah jenis tracker yang digunakan untuk mengumpulkan data analitik tentang penggunaan aplikasi. Ini dapat mencakup informasi seperti jumlah pengguna unik, frekuensi penggunaan, tingkat retensi pengguna, konversi, dan lain-lain. Data ini membantu pengembang dan pemilik aplikasi dalam memahami perilaku pengguna dan membuat keputusan berdasarkan data.

4. Tracker Lokasi (Location Tracker)

Ini adalah jenis tracker yang digunakan untuk melacak lokasi pengguna dalam aplikasi. Ini dapat digunakan dalam aplikasi yang memerlukan informasi lokasi, seperti aplikasi navigasi, layanan pengiriman, dan lain-lain.

Penggunaan "tracker" dalam aplikasi sangat bervariasi tergantung pada kebutuhan dan tujuan aplikasi tersebut. Dalam banyak kasus, "tracker" membantu dalam meningkatkan pengalaman pengguna, mengoptimalkan kinerja, meningkatkan keamanan, dan membuat keputusan berdasarkan data (Pane, 2020).

2.2.5 Dashboard

Dashboard adalah sebuah antarmuka visual yang menyajikan informasi penting dalam bentuk grafik, tabel, dan metrik lainnya secara terpadu dan mudah dipahami. Dashboard digunakan untuk memonitor, menganalisis, dan mengelola kinerja suatu sistem, proses, atau aktivitas dengan cepat dan efektif. Biasanya, informasi yang ditampilkan di dalam dashboard berasal dari berbagai sumber data yang telah diolah dan disajikan secara ringkas dan intuitif. Tujuan utama dari dashboard adalah untuk menyediakan gambaran keseluruhan tentang situasi atau performa suatu entitas secara real-time atau berdasarkan periode waktu tertentu, sehingga memungkinkan pengguna untuk mengambil keputusan yang tepat dan strategis. Dalam konteks bisnis, dashboard sering digunakan untuk memantau kinerja keuangan, pemasaran, sumber daya manusia, dan operasional perusahaan.

Komponen yang umumnya ada di dalam sebuah dashboard dapat bervariasi tergantung pada tujuan dan konteks penggunaannya, namun beberapa komponen yang umumnya ditemui di dalam dashboard adalah sebagai berikut:

1. Grafik dan Diagram digunakan untuk menyajikan data secara visual, seperti grafik batang, grafik garis, grafik lingkaran, dan diagram lainnya. Ini membantu pengguna untuk dengan cepat memahami pola dan tren dalam data.

2. Tabel dan Metrik memberikan tampilan yang rinci tentang data, termasuk angka-angka spesifik dan nilai-nilai kunci. Metrik ini seringkali disajikan dalam bentuk angka atau persentase untuk menggambarkan kinerja atau pencapaian tertentu.

3. KPI (Key Performance Indicators) adalah indikator kinerja kunci yang digunakan untuk mengukur kesuksesan atau pencapaian tujuan tertentu. KPI ini dapat disajikan dalam bentuk grafik, angka, atau indikator visual lainnya untuk memudahkan pemantauan.

4. Filter dan Pemfilteran Data Filter memungkinkan pengguna untuk memilih dan menyesuaikan data yang ingin mereka lihat di dalam dashboard. Ini memungkinkan pengguna untuk fokus pada informasi yang relevan dan penting bagi mereka.

2.2.6 Javascript

JavaScript adalah bahasa pemrograman yang digunakan untuk mengembangkan aplikasi web interaktif dan dinamis. Ini adalah salah satu dari tiga teknologi inti web

bersama dengan HTML dan CSS. JavaScript dapat dijalankan di sisi klien (browser) untuk membuat halaman web responsif dan berinteraksi dengan pengguna. Selain itu, dengan penggunaan Node.js, JavaScript juga dapat dijalankan di sisi server untuk mengelola logika server (Devie Rosa Anamisa at al., 2022).

JavaScript merupakan bahasa pemrograman web yang pemrosesan ya dilakukan di sisi client. Karena berjalan di sisi client, JavaScript dapat dijalankan hanya dengan menggunakan browser. Berbeda dengan PHP yang bekerja di sisi server, untuk menjalankan skrip JavaScript tidak memerlukan refresh pada browser. JavaScript biasanya dijalankan ketika ada event tertentu yang terjadi pada halaman web. Baik event yang dilakukan oleh user, maupun event yang terjadi karena adanya perubahan pada halaman website. JavaScript mulai sangat populer sejak munculnya query sebagai salah satu library yang memudahkan penggunaan JavaScript. Namun seiring munculnya library baru dan framework baru dengan berbagai kelebihanya, popularitas jQuery mulai tergeser (Rohi Abdulloh., 2020).

2.2.7 PHP

PHP singkatan dari PHP: Hypertext Preprocessor yaitu bahasa pemrograman web server-side yang bersifat open source. PHP merupakan script yang terintegrasi dengan HTML dan berada pada server (server side HTML embedded scripting). PHP adalah script yang digunakan untuk membuat halaman website yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh client. Mekanisme ini menyebabkan informasi yang diterima client selalu yang terbaru/up to date. Semua script PHP dieksekusi pada server di mana script tersebut dijalankan (Anhar., 2020).

PHP (Hypertext Preprocessor) adalah bahasa pemrograman yang banyak digunakan untuk pengembangan web. PHP bersifat server-side, yang berarti kode PHP dijalankan di server web, menghasilkan HTML yang dikirimkan ke browser pengguna. Ini sering digunakan untuk membuat halaman web dinamis dan interaktif, mengakses database, mengelola formulir, dan berbagai tugas pengembangan web lainnya. Dalam konteks "Aplikasi Kehadiran Dosen menggunakan PHP OOP," mungkin PHP OOP (Object-Oriented Programming) digunakan untuk menyusun kode PHP dalam struktur yang lebih terorganisir, modular, dan dapat dipelihara. Paradigma OOP memungkinkan

pengelolaan lebih baik terhadap kompleksitas aplikasi dan meningkatkan keterbacaan serta skalabilitas kode. (Roni Habibi., 2020).

2.2.8 Database

Database adalah kumpulan data terstruktur yang disimpan secara terpadu dalam suatu sistem yang dapat diakses dan dikelola oleh pengguna atau aplikasi. Data yang disimpan dalam database biasanya terorganisir dalam tabel atau relasi, dan dapat diakses dan diubah dengan menggunakan bahasa query seperti SQL (Drs. Afrizal zein et al., 2023). Komponen Database Database terdiri dari beberapa komponen utama, antara lain:

- 1) Data: informasi yang disimpan dalam database.
- 2) Struktur Data: cara data disimpan dan diorganisir dalam database, seperti tabel, indeks, relasi, dan lain-lain.
- 3) Metadata: informasi tentang struktur dan isi database, seperti definisi tabel, kolom, indeks, dan lain-lain.
- 4) Database Management System (DBMS): perangkat lunak yang digunakan untuk mengelola database, termasuk mengelola akses pengguna, menjaga integritas data, dan melakukan backup dan recovery.

2.2.9 MySql

MySQL adalah DBMS yang open source dengan dua bentuk lisensi, yaitu Free Software (perangkat lunak bebas) dan Shareware (perangkat lunak berpemilik yang penggunaannya terbatas). Jadi, MySQL adalah database server yang gratis dengan lisensi GNU General Public License (GPL) sehingga dapat dipakai untuk keperluan pribadi atau komersil tanpa harus membayar lisensi yang ada. Seperti yang sudah disinggung di atas, MySQL masuk ke dalam jenis RDBMS (Relational Database Management System). Maka dari itu, istilah semacam baris, kolom, tabel, dipakai pada aplikasi database ini. Contohnya di dalam MySQL sebuah database terdapat satu atau beberapa tabel (Respaty Namruddin at al., 2023).

SQL sendiri merupakan suatu bahasa yang dipakai di dalam pengambilan data pada relational database atau database yang terstruktur. Jadi MySQL adalah database management system yang menggunakan bahasa SQL sebagai bahasa penghubung antara perangkat lunak aplikasi dengan database server.

2.2.10 Laravel

Laravel merupakan salah satu dari sekian banyak framework PHP yang dapat digunakan secara gratis. Laravel dikembangkan oleh programmer keren asal Amerika yaitu Taylor Otwell pada tahun 2011. Sejak dirilis ke publik, secara perlahan Laravel mulai merebut perhatian para programmer dunia. Hingga buku ini ditulis, Laravel telah menjadi salah satu framework favorit programmer dunia, mengalahkan framework-framework pendahulunya yang sudah lebih dulu lahir (Rohi Abdulloh., 2022).

2.2.11 Software Development Life Cycle (SDLC)

SDLC adalah siklus perancangan atau pengembangan perangkat lunak yang bertujuan untuk menyelesaikan masalah secara efektif dan menghasilkan sistem yang berkualitas sesuai dengan keinginan pengguna atau sesuai tujuan perancangan sistem. SDLC dalam rekayasa perangkat lunak adalah proses membuat dan memodifikasi perangkat lunak dengan menggunakan model atau metode pengembangan perangkat lunak tersebut. Konsep SDLC mendasari banyak jenis metodologi pengembangan perangkat lunak. Ada 3 jenis metode siklus hidup sistem yang paling banyak digunakan, yaitu: SDLC tradisional, SDLC prototyping, dan SDLC berorientasi pada objek (Nazaruddin Ahmad et al., 2022).

2.2.12 Rapid Application Development (RAD)

Rapid Application Development (RAD) adalah istilah awalnya digunakan untuk menggambarkan proses pengembangan perangkat lunak pertama kali dikembangkan dan berhasil digunakan selama pertengahan 1970-an oleh Sistem Pusat Pengembangan New York Telephone Co di bawah arahan Dan Gielan. Setelah serangkaian implementasi sangat berhasil dari proses ini, Gielan kuliah secara ekstensif di berbagai forum pada metodologi, praktek, dan manfaat dari proses ini (Roni Habibi et al., 2020). Model RAD mengadopsi model waterfall dan pembangunan dalam waktu singkat yang dicapai dengan menerapkan:

1) Perencanaan Kebutuhan (Requirement Planning)

Pada tahap awal ini, pengembang bersama dengan pemangku kepentingan melakukan pertemuan untuk mengumpulkan dan memahami kebutuhan bisnis serta teknis proyek. Diskusi ini bertujuan untuk mengidentifikasi kebutuhan utama dan menetapkan tujuan proyek secara jelas. Hasil akhir dari tahap ini adalah sebuah

dokumen persyaratan yang komprehensif yang akan menjadi panduan selama proses pengembangan.

2) Desain Pengguna (User Design)

Tahap ini melibatkan kolaborasi yang intens antara pengguna dan pengembang untuk menciptakan prototipe, model, dan arsitektur awal aplikasi. Prototipe ini dibuat dengan cepat untuk mendapatkan umpan balik langsung dari pengguna. Tujuan utamanya adalah memastikan aplikasi yang dikembangkan benar-benar memenuhi kebutuhan dan harapan pengguna. Hasil dari tahap ini adalah prototipe yang dapat dengan mudah diubah dan diperbaiki berdasarkan masukan yang diterima dari pengguna.

3) Pembangunan dan Konstruksi (Construction)

Pada tahap ini, pengembang mulai membangun sistem akhir menggunakan alat dan metode RAD. Prototipe yang telah dibuat sebelumnya diubah menjadi aplikasi yang berfungsi penuh. Pengembang menggunakan komponen yang dapat digunakan kembali dan alat pengembangan visual untuk mempercepat proses pembangunan. Tujuan dari tahap ini adalah untuk menciptakan sistem dengan cepat namun tetap berkualitas tinggi. Hasilnya adalah aplikasi yang hampir selesai, yang mungkin masih memerlukan beberapa perbaikan kecil sebelum bisa diterapkan.

4) Implementasi (Cutover)

Tahap implementasi adalah fase akhir di mana aplikasi yang telah dikembangkan diuji secara menyeluruh dan kemudian dipindahkan ke lingkungan produksi. Pengujian dilakukan untuk memastikan bahwa aplikasi berfungsi dengan baik dan bebas dari kesalahan. Pengguna juga diberikan pelatihan yang diperlukan agar mereka dapat menggunakan aplikasi dengan efektif. Tujuan dari tahap ini adalah memastikan aplikasi siap untuk digunakan dalam lingkungan bisnis sebenarnya. Hasil akhirnya adalah aplikasi yang sudah siap digunakan dan diimplementasikan dalam lingkungan bisnis yang sebenarnya.

2.2.13 Unified Modelling Language (UML).

Unified Modeling Language atau biasa disebut UML dikembangkan oleh Object Management Group (OMG). UML digunakan untuk menggambarkan dan merepresentasikan berbagai aspek dalam sebuah sistem, seperti struktur kelas, diagram

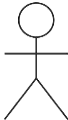

use case, activity diagram, dan lain-lain. Pengembang perangkat lunak menggunakan UML untuk membantu proses analisis, desain, dan dokumentasi sistem. UML didefinisikan sebagai bahasa pemodelan yang terstandarisasi dan dapat mengilustrasikan berbagai jenis sistem, seperti sistem perangkat lunak dan sistem bisnis.

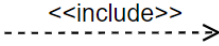
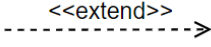

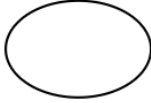


UML termasuk sebagai bahasa pemodelan yang sederhana dan fleksibel, sehingga dapat digunakan dalam berbagai konteks dan dapat disesuaikan dengan kebutuhan pengguna. UML juga dapat digunakan oleh berbagai stakeholder dalam pengembangan sistem, termasuk pengembang, analis, manajer proyek, dan pengguna akhir. Ada banyak diagram didalam UML diantaranya.

1. Use Case Diagram

Jenis diagram dalam UML (*Unified Modeling Language*) yang disebut *use case* diagram digunakan untuk menunjukkan interaksi antara sistem dan pengguna atau antara sistem dengan sistem lain. *Use case* menggambarkan dengan jelas terkait fungsi-fungsi sistem dan bagaimana sistem berinteraksi dengan pengguna atau aktor lain.

Tahap analisis dan desain biasanya menggunakan diagram ini untuk membantu dalam memahami kebutuhan fungsional sistem dan untuk memfasilitasi komunikasi antara para pemangku kepentingan. Diagram ini juga dapat membantu dalam mengidentifikasi kebutuhan bisnis dan dalam memastikan bahwa sistem memenuhi kebutuhan tersebut. berikut adalah simbol yang terdapat pada diagram tersebut.

| Simbol | Nama | Keterangan |
|---|----------------|--|
|  | Actor | Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan use case. |
|  | Generalization | Hubungan dimana objek anak (descendent) berbagi perilaku dan struktur daya dari objek yang ada diatasnya objek induk (ancestor). |







| | | |
|---|---------------|---|
|  | Include | Menspesifikasikan bahwa use case sumber secara eksplisit. |
|  | Extend | Menspesifikasikan bahwa use case target memperluas perilaku dari use case sumber pada suatu titik yang diberikan |
|  | Association | yang menghubungkan antara objek satu dengan objek lainnya. |
|  | Use Case | Deskripsi dari uruan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor. |
|  | Collaboration | Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi). |
|  | Note | Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi. |

2. Activity Diagram

Salah satu jenis diagram yang tersedia dalam UML (Unified Modeling Language) adalah activity diagram yang umumnya dipakai untuk memberikan gambaran urutan tindakan atau kegiatan pada sebuah proses atau aliran kerja (*workflow*). Diagram ini memberikan gambaran yang jelas mengenai alur kerja sistem dan interaksi antara objek atau entitas dalam sistem.

Diagram ini berguna dalam menggambarkan proses dan sistem kompleks. Diagram ini bisa membantu untuk mengidentifikasi masalah dalam alur kerja yang ada dan berguna dalam merancang sebuah solusi yang lebih efektif. *Activity*

diagram juga sangat berguna sebagai perantara komunikasi antara para pemangku kepentingan atau *client* dengan pengembang dan memastikan bahwa sistem telah memenuhi kebutuhan mereka. berikut merupakan simbol pada diagram tersebut.


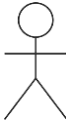

| Simbol | Nama | Keterangan |
|---|------------|---|
|  | Start | Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal. |
|  | Activity | Aktivitas yang dilakukan sistem dan biasanya diawali dengan kata kerja. |
|  | Decision | Asosiasi atau percabangan merupakan kondisi dimana ada pilihan aktivitas lebih dari satu |
|  | Join | Join merupakan penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu. |
|  | End | Status akhir yang dilakukan sistem, sebuah sistem diharuskan untuk memiliki sebuah status akhir |
|  | Flow final | Simbol Flow Final menandakan akhir dari aliran proses atau aktivitas, menunjukkan bahwa proses telah selesai dan tidak ada langkah lagi yang perlu dilakukan. |

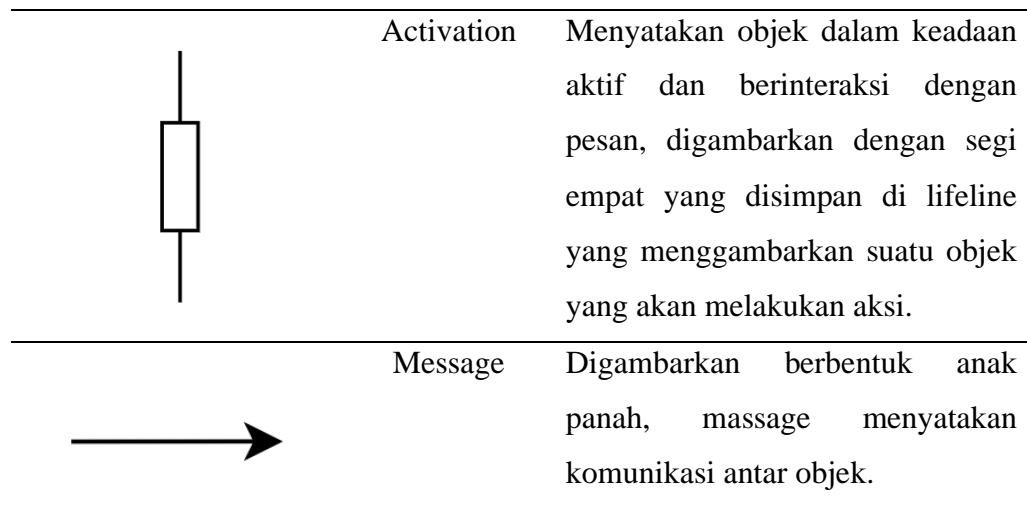
3. Sequence Diagram

Diagram pada UML yang digunakan untuk memodelkan interaksi antar *object* pada sebuah sistem adalah sequence diagram. *Sequence* terdiri dari objek-objek yang direpresentasikan oleh kotak vertikal dan pesan-pesan atau panggilan metode yang direpresentasikan oleh panah horizontal antara objek-objek tersebut.

Urutan pesan atau panggilan metode ditunjukkan oleh urutan panah, dan waktu dieksekusinya ditunjukkan oleh sumbu waktu vertikal.

Sequence diagram sangat berguna dalam memodelkan interaksi antara objek dalam sistem, terutama dalam kasus yang melibatkan banyak objek yang berinteraksi secara kompleks. Dengan menggunakan *sequence* diagram, pengembang dapat memvisualisasikan urutan pesan atau panggilan metode antar objek dan menemukan bug atau kelemahan dalam desain sistem sebelum diimplementasikan. Diagram ini juga dapat digunakan sebagai dokumen referensi atau dokumentasi untuk membantu pemeliharaan dan pengembangan sistem di masa depan. Berikut ini merupakan beberapa simbol yang terdapat di *sequence* diagram:

| Simbol | Nama | Keterangan |
|---|----------|---|
|  | Objek | Menyatakan objek yang berinteraksi atau mengirimkan pesan dapat juga menjadi sebuah instance dari class dengan nama objek di dalamnya yang diawali titik koma. |
|  | Aktor | Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang dibuat sendiri jadi walaupun aktor adalah gambar atau orang tetapi belum tentu merupakan orang. |
|  | Lifeline | Menyatakan kehidupan suatu objek atau keberadaan sebuah object. Lifeline biasanya garis putus vertikal yang ditarik dari sebuah objek. |



2.2.14 Black Box Testing

Black Box Testing adalah proses pengujian perangkat lunak yang fokus pada evaluasi fungsionalitas sistem tanpa memperhatikan rincian struktur internal atau kode programnya. Dalam pengujian ini, pengujian dilakukan tanpa pengetahuan tentang bagaimana perangkat lunak mengolah data di dalamnya. Tujuannya adalah untuk memastikan bahwa perangkat lunak berperilaku sesuai dengan spesifikasi yang telah ditetapkan dan memenuhi kebutuhan pengguna, tanpa memperhatikan detail implementasi di dalamnya. Metode ini memungkinkan identifikasi bug atau kesalahan dalam fungsionalitas aplikasi tanpa memerlukan pengetahuan mendalam tentang struktur atau logika internal perangkat lunak tersebut.

Klasifikasi *Black Box Testing* meliputi beberapa jenis pengujian, di antaranya:

1. Pengujian *Fungsional*

Ini melibatkan pengujian perangkat lunak terhadap persyaratan fungsionalnya. Pengujian dilakukan dalam bentuk tertulis untuk memeriksa apakah aplikasi berperilaku seperti yang diharapkan, termasuk kemampuan dalam menangani perintah pengguna, manipulasi data, pencarian, proses bisnis, dan integrasi.

2. Pengujian Tegangan

Melibatkan penilaian kualitas aplikasi dalam lingkungan yang menuntut. Tujuannya adalah untuk menciptakan kondisi lingkungan yang menuntut untuk mengevaluasi bagaimana aplikasi berperilaku di bawah tekanan, yang dapat membantu mengidentifikasi masalah atau kelemahan kinerja.

3. Pengujian Beban

Melibatkan pengujian aplikasi dengan beban atau input yang berat untuk menilai kinerjanya saat diberikan beban yang tinggi. Ini membantu dalam menentukan apakah aplikasi mampu menangani beban kerja yang besar tanpa mengalami kegagalan atau penurunan kinerja.

Pengujian Black Box Testing dan klasifikasinya bertujuan untuk meningkatkan kepercayaan pengembang perangkat lunak terhadap fungsi-fungsi aplikasi, serta memastikan bahwa aplikasi dapat beroperasi dengan baik dalam berbagai kondisi dan lingkungan (Manuaba, 2023).

2.1 Tinjauan Studi

Penyusunan laporan tugas akhir ini didukung oleh beberapa referensi dari jurnal terdahulu yang berkaitan dengan objek penelitian, diantaranya:

- 1) Penelitian pertama yaitu pengembangan dashboard yang dilakukan oleh Dede Bagus Setiawan, dengan judul **“PENGEMBANGAN DASHBOARD INTERAKTIF COVID-19 SEBAGAI ALAT INFORMASI MASYARAKAT DALAM MELIHAT PANDEMI DI INDONESIA”** yang dimuat dalam jurnal Technologia: Jurnal Ilmiah Vol.2 No.1 Tahun 2023. Jurnal ini membahas tentang pengembangan Dashboard Interaktif Covid-19 sebagai alat informasi masyarakat dalam melihat pandemi di Indonesia. Dilakukan oleh Dede Bagus Setiawan dari Universitas Islam 45 Bekasi, penelitian ini bertujuan untuk menciptakan alat yang memberikan informasi aktual tentang perkembangan pandemi Covid-19 serta memberdayakan masyarakat dengan pemahaman yang lebih baik tentang situasi tersebut. Dashboard ini menggabungkan data dari berbagai sumber terpercaya dan mengubahnya menjadi visualisasi informatif, memberikan informasi tentang jumlah kasus, vaksinasi, serta kebijakan yang diterapkan di tingkat lokal, nasional, dan global. Selain itu, dashboard ini juga memberikan pemahaman tentang dampak pandemi pada sektor kesehatan, ekonomi, pendidikan, dan sosial melalui visualisasi data yang jelas.
- 2) Pembahasan kedua yaitu membahas tentang pengembangan dashboard di Universitas Ibnu Sina Batam yang di buat Lastri Yanti. (2019). **“PENGEMBANGAN DASHBOARD UNIVERSITAS IBNU SINA BATAM BERBASIS WEB MOBILE”** (UIS), yang dimuat dalam jurnal Technologia: Jurnal Ilmiah Vol.3 No.2

Tahun 2019. Jurnal ini membahas tentang pengembangan Dashboard Universitas Ibnu Sina Batam berbasis web mobile. Penelitian dilakukan untuk mendukung pengembangan kampus digital Universitas Ibnu Sina dengan merancang dan mengembangkan dashboard yang terintegrasi. Objek penelitian berfokus pada bagian Biro Administrasi dan Akademik serta Kemahasiswaan (BAAK) di kampus tersebut. Metode yang digunakan adalah Object Oriented Analysis and Design (OOAD) yang meliputi kebutuhan, analisis, desain, implementasi, pengujian, dan deployment, dengan menggunakan permodelan UML. Sistem informasi yang dihasilkan diharapkan mampu memberikan informasi secara real-time dan akurat, serta dapat diakses oleh seluruh civitas akademika di Universitas Ibnu Sina, mempermudah akses terhadap informasi melalui internet.

- 3) Penelitian ketiga yaitu membahas tentang Perancangan Aplikasi Tracking Sales di PT. Infinity yang dilakukan oleh Yusup Nur Abdurrahman, Sari Susanti. (2021). ” **PERANCANGAN APLIKASI TRACKING SALES BERBASIS ANDROID PADA PT. INFINITY**”. E-PROSIDING SISTEM INFORMASI Vol. 2, No. 2, Desember 2021. Jurnal ini membahas tentang pengembangan aplikasi tracking sales berbasis Android pada PT. Infinity, yang merupakan sebuah perusahaan yang berfokus pada pengembangan sumber daya manusia. Dalam konteks kerjasama dengan perusahaan besar seperti Akulaku, sistem pemantauan dan pengawasan terhadap salesman menjadi sangat penting. Saat ini, proses pemantauan masih dilakukan melalui GroupChat WhatsApp, yang memiliki risiko pemalsuan informasi dan kesulitan dalam pemantauan secara efektif. Oleh karena itu, aplikasi yang dikembangkan dengan menggunakan metode Waterfall dan Java sebagai bahasa pemrograman berbasis Android ini, dilengkapi dengan teknologi Google Maps API untuk memberikan informasi lokasi posisi salesman secara langsung. Diharapkan bahwa aplikasi ini dapat mempermudah proses pemantauan dan pengawasan bagi supervisor atau team leader terhadap sales.
- 4) Penelitian keempat yaitu penelitian yang dilakukan oleh Anastasia Meyliana, Lina Ayu Safitri, Anik Andriani dengan judul “**APLIKASI METODE RAPID APPLICATION DEVELOPMENT (RAD) DALAM PERANCANGAN WEBSITE PT SOVVA KREASI INDONESIA** “. Indonesian Journal on Networking and Security - Volume 11 No 3 – 2022. Jurnal ini membahas tentang pengembangan

website untuk PT Sovva Kreasi Indonesia, perusahaan yang bergerak dalam bidang perawatan wajah dan tubuh dengan merek Sovva Beauty. Dalam penelitian ini, dilakukan perancangan website menggunakan metode Rapid Application Development (RAD) untuk membantu mempromosikan dan meningkatkan penjualan produk perusahaan. Metode analisis PIECES digunakan untuk menganalisis permasalahan, sementara metode black box digunakan untuk pengujian. Melalui observasi, studi pustaka, dan wawancara, penelitian ini menghasilkan sebuah website yang dapat membantu dalam memperluas jangkauan pemasaran, mempromosikan produk, meningkatkan penjualan, serta memudahkan pengelolaan data produk dan penyajian laporan bagi divisi marketing.

- 5) Jurnal oleh Mohammad Reza Fahlevi, Dwi Rizky Rahmawati, Binti Mamluatul Karomah. (2020). **”RANCANG BANGUN SISTEM INFORMASI PEMBAYARAN SPPBERBASIS WEB MENGGUNAKAN FRAMEWORKLARAVEL 9”**. Jurnal Ilmu Komputer dan Sistem Informasi (JIKOMSI) Volume 6, Nomor 3 Oktober-Desember 2023. Jurnal ini membahas tentang pengembangan sistem informasi pembayaran Sumbangan Pembinaan Pendidikan (SPP) berbasis web menggunakan Framework Laravel 9. Penelitian ini dilakukan oleh Mohammad Reza Fahlevi, Dwi Rizky Rahmawati, dan Binti Mamluatul Karomah dari Program Studi Teknik Informatika, Universitas Nahdlatul Ulama Indonesia, Jakarta, Indonesia. Mereka merancang dan mengembangkan sistem informasi yang bertujuan membantu sekolah beradaptasi dengan perubahan zaman dalam pengelolaan dan rekapan data pembayaran SPP. Dengan memanfaatkan teknologi ini, sekolah dapat mengoptimalkan proses transaksi serta pengarsipan data pembayaran SPP, memberikan kemudahan dan keamanan bagi siswa dan orang tua dalam melakukan pembayaran melalui platform online. Selain itu, penerapan sistem informasi berbasis web ini juga membawa berbagai manfaat lainnya, seperti kemudahan pengarsipan, akses data yang cepat dan akurat, serta peningkatan dalam pengelolaan keuangan sekolah.

BAB III

METODE PENELITIAN

3.1 Objek Penelitian

3.1.1 Latar Belakang Objek Penelitian

Institusi Pendidikan XYZ didirikan pada tahun 2011 oleh Yayasan Pendidikan Jaya, yang merupakan bagian dari Grup Perusahaan Pembangunan Jaya. Lokasinya terletak di Kawasan Bintaro Jaya, Tangerang Selatan. Seperti halnya kebanyakan perguruan tinggi, Institusi Pendidikan XYZ juga menyediakan sistem beasiswa untuk mahasiswanya. Penanganan beasiswa dilakukan oleh Biro Kemahasiswaan dan Alumni (BKAL), sebuah entitas khusus yang bertanggung jawab atas semua aspek terkait beasiswa.

Dalam operasionalnya, Biro Kemahasiswaan memiliki beberapa fungsi utama, termasuk pengelolaan minat dan bakat di tingkat universitas, pembinaan lembaga kemahasiswaan seperti Badan Perwakilan Mahasiswa (BPM), Badan Eksekutif Mahasiswa, serta organisasi di bidang minat dan bakat seperti Unit Kemahasiswaan dan Klub. Biro ini juga berperan dalam meningkatkan prestasi mahasiswa dalam berbagai kegiatan dan lomba di tingkat lokal, nasional, dan internasional.

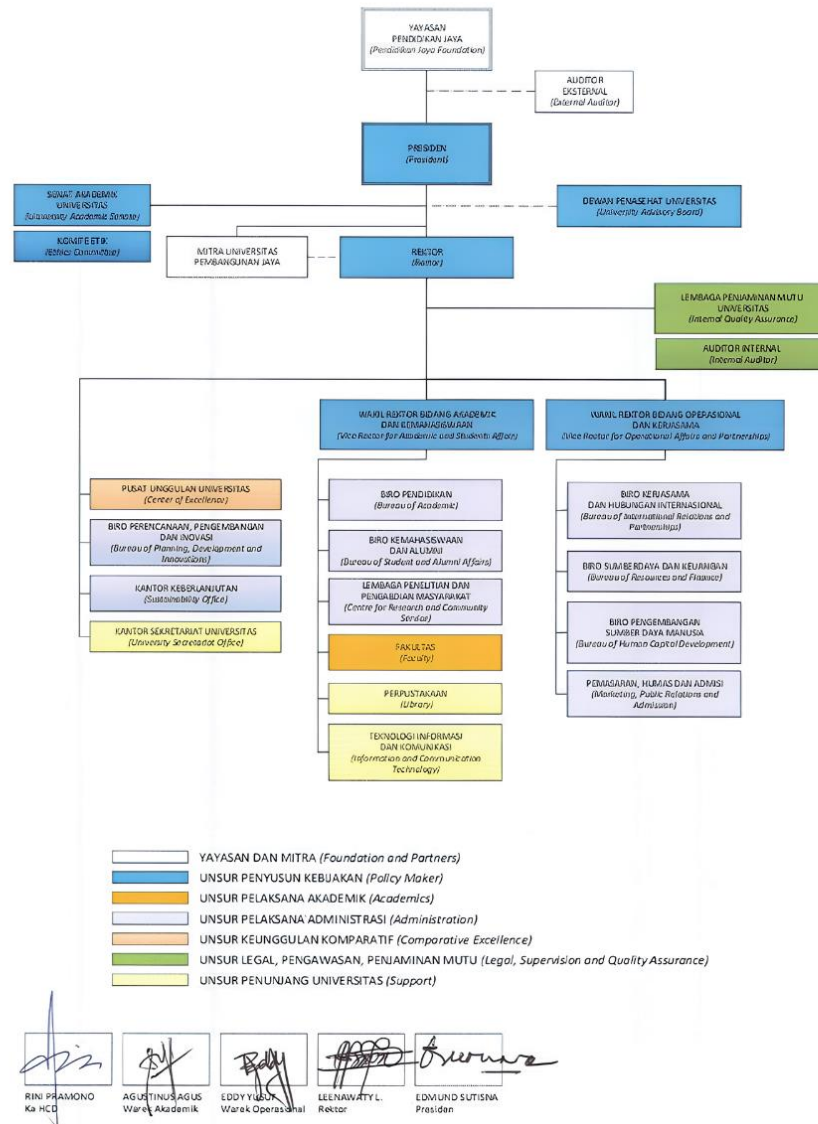
Selain itu, Biro Kemahasiswaan juga bertanggung jawab dalam meningkatkan kesejahteraan mahasiswa, termasuk pengelolaan beasiswa untuk mahasiswa berprestasi dan yang berasal dari keluarga kurang mampu. Fungsi lainnya melibatkan pengelolaan karir, di mana Biro ini mendorong mahasiswa untuk mempersiapkan diri menghadapi dunia kerja setelah lulus, melalui kegiatan pelatihan, seminar, penyaluran kerja, job fair, dan campus hiring.

Terakhir, Biro Kemahasiswaan memiliki tanggung jawab terhadap pengelolaan alumni sebagai salah satu stakeholder universitas. Mereka terlibat dalam kegiatan seperti temu alumni, tracer study, sharing session, dan monitoring keberadaan alumni untuk memastikan kontribusi berkelanjutan mereka terhadap komunitas universitas.

3.1.2 Struktur Organisasi

Berikut merupakan struktur organisasi dari Institusi Pendidikan XYZ.

Struktur Organisasi Universitas Pembangunan Jaya



3.2 Metode Penelitian

Metode penelitian adalah cara yang digunakan untuk mencapai tujuan penelitian, yaitu untuk mendapatkan data yang valid dan reliabel. Metode penelitian merupakan suatu cara yang sistematis, terencana, dan terarah untuk mencapai tujuan penelitian (Jurnal Sosiologi Universitas Islam Indonesia Vol. 25 No. 1 Tahun 2020).

Dari beberapa definisi di atas, dapat disimpulkan bahwa metode penelitian adalah cara ilmiah yang sistematis, terencana, dan terarah untuk mencapai tujuan penelitian, yaitu untuk mendapatkan data yang valid dan reliabel. Data yang diperoleh kemudian dianalisis dan diinterpretasikan untuk menjawab pertanyaan penelitian dan menghasilkan kesimpulan.

Metode penelitian sangat penting dalam penelitian karena membantu peneliti untuk:

- Memperoleh data yang valid dan reliabel, data yang valid adalah data yang benar dan sesuai dengan kenyataan, sedangkan data yang reliabel adalah data yang konsisten dan dapat dipercaya.
- Menjawab pertanyaan penelitian, pertanyaan penelitian adalah pertanyaan yang ingin dijawab oleh peneliti melalui penelitian. Pertanyaan penelitian harus jelas, terukur, dan dapat diuji.
- Menghasilkan kesimpulan, kesimpulan adalah jawaban atas pertanyaan penelitian. Kesimpulan harus didasarkan pada data yang telah dianalisis dan diinterpretasikan.

3.3 Metode Pengumpulan Data

Metode pengumpulan data yang dilakukan peneliti untuk mengumpulkan data dalam melakukan penelitian menyatakan bahwa secara umum terdapat 2 (dua) macam teknik pengumpulan data wawancara, dan observasi.

a) Wawancara

Tahap wawancara yang melibatkan Mahasiswa dan Biro Kemahasiswaan Alumni di Institusi Pendidikan xyz bertujuan untuk mengidentifikasi kebutuhan data, informasi, dan permasalahan dalam melacak dan memantau progres pelaporan beasiswa. Aplikasi ini diharapkan mampu meningkatkan pemantauan dengan menyediakan dashboard intuitif dan tracker yang memberikan informasi real-time mengenai kinerja akademik mahasiswa. Fitur utama yang diinginkan dalam gambaran menyeluruh tentang status perkembangan akademik, notifikasi otomatis, dan kemampuan rekapitulasi terkait mahasiswa penerima beasiswa.

b) Observasi

Observasi dalam konteks pengumpulan data pada aplikasi beasiswa adalah proses pengamatan langsung terhadap aktivitas, perilaku, dan kinerja mahasiswa

penerima beasiswa yang dilakukan oleh pihak BKAL (Biro Kemahasiswaan dan Alumni).

3.4 Metode Pengembangan Sistem

Metode pengembangan sistem yang diterapkan dalam penelitian ini adalah metodologi pengembangan RAD (Rapid Application Development) yang merupakan salah satu metode pengembangan yang ada dalam Software Development Life Cycle (SDLC). RAD merupakan model proses pengembangan perangkat lunak secara linear sequential yang berfokus pada siklus pengembangan dalam waktu singkat serta menggunakan metode iteratif (berulang) dimana model kerjanya dikonstruksikan pada awal tahap pengembangan untuk menentukan kebutuhan pengguna. Dalam menjalankan metode ini dibutuhkan kolaborasi antara pengembang dan pengguna untuk menemukan kebutuhan bisnis, membuat desain konsep, dan pengujian prototipe. RAD merupakan metodologi pengembangan sistem yang menekankan iterasi cepat, umpan balik pengguna berkelanjutan, dan prototipe untuk membangun aplikasi secara efisien. Prinsip utamanya adalah melibatkan pengguna secara aktif sejak awal untuk menghasilkan sistem yang memenuhi kebutuhan mereka (Jurnal Teknoinfo Vol. 3 No. 1 Tahun 2020).

3.2.1 Perencanaan Kebutuhan (Requirement Planning)

Tahap ini merupakan fase pertama dalam pengembangan sistem, di mana identifikasi masalah dan pengumpulan data dilakukan dengan melibatkan pengguna atau pemangku kepentingan. Tujuan utamanya adalah untuk mengetahui tujuan akhir sistem dan kebutuhan informasi yang diinginkan. Keterlibatan aktif dari kedua pihak sangat penting pada tahap ini untuk mengidentifikasi kebutuhan pengembangan sistem dengan baik.

3.2.2 Desain Pengguna (User Design)

Pada fase ini, dilakukan perancangan sistem yang mencakup UML, UI/UX, dan prototype aplikasi. Proses perancangan sistem dapat diulang jika terdapat ketidaksesuaian antara desain dengan kebutuhan pengguna yang telah diidentifikasi sebelumnya. Tahap ini juga melibatkan pembuatan prototype sebagai representasi visual bagi pengguna mengenai tampilan UI/UX aplikasi sesuai harapan. Prototype yang telah dirancang kemudian diujikan kepada pengguna dengan mengumpulkan umpan balik untuk penyempurnaan dan penambahan fitur yang diperlukan. Selain itu,

tahap ini melibatkan pengembangan model data dan proses, serta model arsitektur teknis.

3.2.3 Pembangunan dan Konstruksi (Construction)

Tahap ini merupakan fase di mana perangkat lunak sebenarnya dibuat atau dimulainya proses coding. Pada tahap ini, pengembangan perangkat lunak berlangsung secara iteratif dengan penerapan teknologi pemrograman berorientasi objek. Proses ini secara konsisten memperhitungkan umpan balik dari pengguna untuk mencapai hasil yang sesuai dengan preferensi mereka. Iterasi ini dilakukan berulang kali hingga mencapai versi akhir dari aplikasi, bertujuan untuk menjamin kualitas dan kesesuaian sistem. Setelah berbagai komponen perangkat lunak selesai dibangun, dilakukan integrasi dan pengujian menyeluruh untuk memverifikasi bahwa sistem beroperasi dengan baik.

3.2.4 Implementasi (Cutover)

Tahap implementasi adalah fase akhir di mana aplikasi yang telah dikembangkan diuji secara menyeluruh dan kemudian dipindahkan ke lingkungan produksi. Pengujian dilakukan untuk memastikan bahwa aplikasi berfungsi dengan baik dan bebas dari kesalahan. Pengguna juga diberikan pelatihan yang diperlukan agar mereka dapat menggunakan aplikasi dengan efektif. Tujuan dari tahap ini adalah memastikan aplikasi siap untuk digunakan dalam lingkungan bisnis sebenarnya. Hasil akhirnya adalah aplikasi yang sudah siap digunakan dan diimplementasikan dalam lingkungan bisnis yang sebenarnya.

3.5 Analisis Sistem Yang Berjalan

Analisis ini bermanfaat untuk mendapatkan data tentang implementasi sistem di objek penelitian. Untuk memahami bagaimana sistem beroperasi saat ini, penulis melakukan pengamatan langsung terhadap proses bisnis serta melakukan wawancara dengan individu yang terlibat dalam penelitian. Hasil dari serangkaian wawancara yang telah dilakukan oleh penulis akan dijabarkan selanjutnya.

| | |
|-------------------------|--|
| Tempat Wawancara | Universitas Pembangunan Jaya |
| Divisi | Biro Kemahasiswaan dan Alumni (BKAL) |
| Narasumber | BKAL <i>Officer</i> |
| Tujuan Wawancara | Tujuan wawancara adalah untuk mendapatkan pemahaman yang lebih |

| | |
|--|--|
| | mendalam tentang kebutuhan terkait pengembangan aplikasi beasiswa. |
|--|--|

List Pertanyaan:

1. Apa tujuan utama dari pengembangan aplikasi dashboard dan tracker untuk pemantauan perkembangan mahasiswa penerima beasiswa?
 2. Bagaimana aplikasi ini akan membantu dalam meningkatkan pemantauan perkembangan mahasiswa penerima beasiswa?
 3. Apa saja fitur utama yang akan dimiliki oleh aplikasi ini?
-

Jawaban Narasumber:

1. Tujuannya untuk memberikan alat yang efisien bagi staf administrasi, BKAL untuk melacak dan memantau progres pelaporan mereka selama masa penerimaan beasiswa.
 2. Aplikasi ini dapat membantu dalam meningkatkan pemantauan dengan menyediakan dashboard yang intuitif dan tracker yang memberikan informasi real-time tentang kinerja akademik, partisipasi dalam kegiatan pelaporan, dan faktor-faktor lain yang relevan.
 3. Fitur utama yang diharapkan meliputi:
 - Dashboard yang memberikan gambaran menyeluruh tentang perkembangan akademik dan kegiatan mahasiswa penerima beasiswa.
 - Tracker untuk memantau kemajuan akademik, partisipasi dalam kegiatan pelaporan, dan pencapaian lainnya.
 - Notifikasi otomatis untuk mengingatkan mahasiswa dan pihak terkait tentang tenggat waktu penting, pertemuan dengan dosen pembimbing, atau acara penting lainnya.
 - Kemampuan untuk melakukan rekapitulasi terkait perkembangan setiap mahasiswa mencakup pemantauan detail tentang kemajuan akademik mereka, ini juga mencakup rekapitulasi mengenai status beasiswa, termasuk beasiswa yang tertunda karena belum memenuhi persyaratan tertentu, serta beasiswa yang sudah selesai ketika mahasiswa telah memenuhi semua kriteria yang ditetapkan.
-

3.3.1 Analisis Dokumen

Analisis dokumen merupakan suatu proses mendalam yang melibatkan penelaahan terhadap berbagai informasi yang terdapat dalam suatu dokumen spesifik. Tujuannya adalah untuk menggali makna dan konteks dari dokumen tersebut, serta untuk mengekstrak informasi yang relevan. Dalam proses ini, para analis akan

membaca dokumen secara teliti, mengidentifikasi pola atau tema yang muncul, dan membuat kesimpulan atau rekomendasi berdasarkan temuan mereka. Analisis dokumen dapat dilakukan pada berbagai jenis dokumen, seperti laporan, artikel, kebijakan, atau panduan. Metode ini umumnya digunakan dalam berbagai konteks, seperti riset, manajemen, pendidikan, dan administrasi, untuk mendapatkan pemahaman yang lebih dalam tentang suatu topik atau masalah tertentu.

I. PENDAHULUAN

Dengan ini saya melaporkan hasil prestasi akademik dan non akademik dalam proses pembelajaran saya selama satu semester di Universitas Pembangunan Jaya Program Studi xxxxxxxxxxxx pada semester (ganjil/genap) tahun ajaran xxxxxxxx

II. LAPORAN PRESTASI AKADEMIK

| No. | Semester | IPS | SKS | IPK | Keterangan |
|-----|----------|-----|-----|-----|------------|
| 1 | I | | | | |
| 2 | II | | | | |
| 3 | III | | | | |
| 4 | IV | | | | |
| 5 | V | | | | |
| 6 | VI | | | | |
| 7 | VII | | | | |
| 8 | VIII | | | | |

*) IP (melampirkan KHS semester terakhir yang dilegalisir oleh Jurusan/Program Studi)

*) Bagi mahasiswa lama, laporkan juga IPK pada semester sebelumnya

III. LAPORAN CAPAIAN NON AKADEMIK

- a) Capaian non akademik yang diraih **selama menjadi mahasiswa** Universitas Pembangunan Jaya (isikan capaian sejak semester 1 hingga sekarang) :

| No | Kegiatan | Tingkat | Waktu Pelaksanaan | Hasil |
|----|----------|---------|-------------------|-------|
| 1 | 2 | 3 | 4 | 5 |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

*) Kolom 3 diisi dengan pilihan kota/propinsi/nasional/internasional

*) melampirkan sertifikat/piagam atas prestasi yang diraih

- b) Keikutsetaan pada kegiatan organisasi kemahasiswaan intra kampus **selama menjadi mahasiswa** Universitas Pembangunan Jaya (*isikan aktifitas organisasi sejak semester 1 hingga sekarang*) :

| No. | Nama Organisasi | Periode Aktif | Jabatan |
|-----|-----------------|---------------|---------|
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

- c) Keikutsertaan pada kegiatan kepanitiaan yang diikuti **selama menjadi mahasiswa** Universitas Pembangunan Jaya (*isikan kegiatan sejak semester 1 hingga sekarang*) :

| No. | Kegiatan | Waktu Pelaksanaan |
|-----|----------|-------------------|
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |

*) melampirkan sertifikat/surat keterangan dari ketua panitia

- d) Publikasi Ilmiah/Karya Tulis/PKM yang dibuat **selama menjadi mahasiswa** Universitas Pembangunan Jaya (*isikan capaian sejak semester 1 hingga sekarang*) :

| No. | Judul Karya Tulis/Karya Ilmiah |
|-----|--------------------------------|
| | |
| | |
| | |
| | |
| | |
| | |
| | |

*) melampirkan hasil Karya Ilmiah/Karya Tulis/PKM yang telah dibuat *hardcopy* dan *softcopy*

Berdasarkan ilustrasi pada Gambar 3.3 dan Gambar 3.4, terlihat format laporan yang dipergunakan oleh mahasiswa saat melakukan pelaporan. Dokumen tersebut mencakup lima tabel yang harus diisi oleh mahasiswa. Tabel pertama memuat laporan mengenai pencapaian akademik selama masa kuliah di Institusi Pendidikan XYZ, sementara tabel-tabel berikutnya memuat pencapaian non-akademik selama menjadi mahasiswa di institusi tersebut, serta informasi mengenai kegiatan organisasi, kepanitiaan, dan karya yang diperoleh oleh mahasiswa selama masa kuliah di Institusi Pendidikan XYZ.

| A | B | C | D | E | F | O | P | Q | R | S | T |
|---------|------------|-----------------|------|-----|--------|------|------|-------|-------|--------|--------|
| No. | NIM | NAMA | AN | PRG | JENIS | 2020 | 2020 | 2021 | 2021 | 2022-1 | 2022-2 |
| #VALUE! | 2017011029 | GAHENDAH NANNDA | 2017 | AKT | ASAK B | 3,45 | 3,46 | lulus | lulus | lulus | lulus |
| 1 | 2017011030 | GAHENDAH NANNDA | 2017 | AKT | ASAK A | 3,56 | 3,56 | lulus | lulus | lulus | lulus |

Selanjutnya, pada Gambar 3.5 terdapat dokumen data identitas mahasiswa penerima beasiswa yang telah dikumpulkan. Dokumen ini berisi beberapa dokumen dan data pendukung yang dapat dianalisis lebih lanjut. Data tersebut mencakup informasi pribadi mahasiswa, seperti nama, angkatan, prodi, ipk, ips, dan nim, serta informasi terkait status penerimaan beasiswa, jenis beasiswa yang diterima, dan periode penerimaan. Selain itu, dokumen ini juga mungkin mencakup data tambahan, seperti transkrip nilai. Semua data ini dikelola dan diolah oleh BKAL melalui file Excel atau sistem manajemen data yang disesuaikan dengan kebutuhan mereka.

3.3.2 Alur Sistem Berjalan

Berdasarkan hasil wawancara dengan BKAL Staff di institusi pendidikan xyz, sistem yang berjalan saat ini masih mengandalkan proses yang melibatkan pencatatan dan pemantauan progres pelaporan mahasiswa penerima beasiswa secara langsung oleh staf administrasi BKAL. Proses ini melibatkan penggunaan berbagai dokumen fisik, seperti laporan kemajuan akademik dan partisipasi dalam kegiatan pelaporan, yang perlu diperiksa, direkam, dan dikelola secara manual. Ketergantungan pada metode ini seringkali memunculkan beberapa tantangan, antara lain keterbatasan dalam aksesibilitas dan kemudahan dalam mengelola data, serta risiko kesalahan manusia dalam pencatatan dan interpretasi informasi.

3.6 Analisis Kebutuhan

3.4.1 Analisis Kebutuhan Sistem

Analisa kebutuhan sistem berisikan konsep dari pengembangan sistem berdasarkan dari hasil observasi yang dilakukan. Secara garis besar Analisa kebutuhan dibagi menjadi dua jenis yaitu analisa kebutuhan fungsional dan non-fungsional. Berikut merupakan analisa kebutuhan fungsional yang akan diterapkan didalam aplikasi:

Elisitasi Tahap I

| Analisis Kebutuhan |
|-------------------------|
| Kebutuhan Functional |
| Saya ingin Sistem Dapat |

| | |
|-----|--|
| 1. | Seluruh user melakukan login |
| 2. | Seluruh user dapat mengelola data profil |
| 3. | Seluruh user melakukan logout |
| 4. | Mahasiswa dapat mengisi form laporan beasiswa |
| 5. | Mahasiswa dapat melihat proses permohonan pengajuan laporan |
| 6. | Mahasiswa dapat menerima feedback |
| 7. | Mahasiswa dapat merubah/merevisi laporan yang telah disubmit |
| 8. | Mahasiswa dapat melakukan chat real-time |
| 9. | Mahasiswa dapat menerima notifikasi laporan & chat telah disetujui |
| 10. | Pembimbing Akademik dapat melihat data Mahasiswa berdasarkan mahasiswa bimbingannya |
| 11. | Kepala Program Studi dapat melihat data mahasiswa berdasarkan mahasiswa bimbingannya |
| 12. | BKAL dapat mengelola data Mahasiswa |
| 13. | BKAL dapat mengelola jadwal deadline |
| 14. | BKAL dapat mengelola data kategori beasiswa |
| 15. | BKAL dapat mengelola data jenis beasiswa |
| 16. | BKAL dapat melihat laporan beasiswa yang telah diupload |
| 17. | BKAL dapat mengirimkan feedback |
| 18. | BKAL dapat menerima laporan beasiswa yang telah direvisi |
| 19. | BKAL dapat menyetujui laporan beasiswa yang telah diupload |
| 20. | BKAL dapat melihat prestasi mahasiswa yang telah di upload |
| 21. | BKAL dapat melihat tracker terhadap mahasiswa laporan tertunda |
| 22. | BKAL dapat melihat detail tracker terhadap mahasiswa laporan tertunda |
| 23. | BKAL dapat melakukan chat real-time kepada mahasiswa |
| 24. | BKAL dapat melihat perkembangan jumlah, berjalan, tertunda, selesai (beasiswa) |
| 25. | BKAL dapat melihat detail perkembangan beasiswa |
| 26. | BKAL dapat melakukan rekap perkembangan beasiswa |
| 27. | BKAL dapat menerima notifikasi laporan & chat |
| 28. | Wakil Rektor, Kepala Program Studi, dan Pembimbing Akademik dapat melihat laporan beasiswa yang telah diupload |
| 29. | Wakil Rektor, Kepala Program Studi, dan Pembimbing Akademik dapat |

| | |
|--------------------------------|---|
| | mengirimkan feedback |
| 30. | Wakil Rektor, Kepala Program Studi, dan Pembimbing Akademik dapat menyetujui laporan beasiswa yang telah diupload |
| 31. | Wakil Rektor, Kepala Program Studi, dan Pembimbing Akademik dapat menerima notifikasi laporan & chat |
| Non Functional | |
| Saya Ingin Sistem Dapat | |
| 1. | Sistem memiliki UI/UX yang ramah <i>user</i> |
| 2. | <i>Password</i> minimal 8 karakter yang terdiri dari kombinasi huruf, simbol dan angka |
| 3. | Sistem harus bisa memberikan respon yang cepat |

Elisitasi Tahap II

Keterangan:

M = *Mandatory* (wajib)

D = *Desirable* (diinginkan)

I = *Inessential* (tidak penting)

| Kebutuhan Functional | | | | |
|--------------------------------|---|----------|----------|----------|
| Analisis Kebutuhan | | | | |
| Saya ingin Sistem Dapat | | M | D | I |
| 1. | Seluruh user melakukan login | ✓ | | |
| 2. | Seluruh user dapat mengelola data profil | ✓ | | |
| 3. | Seluruh user melakukan logout | | ✓ | |
| 4. | Mahasiswa dapat mengisi form laporan beasiswa | ✓ | | |
| 5. | Mahasiswa dapat melihat proses permohonan pengajuan laporan | ✓ | | |
| 6. | Mahasiswa dapat menerima feedback | | ✓ | |
| 7. | Mahasiswa dapat merubah/merevisi laporan yang telah disubmit | ✓ | | |
| 8. | Mahasiswa dapat melakukan chat real-time | ✓ | | |
| 9. | Mahasiswa dapat menerima notifikasi laporan & chat telah disetujui | ✓ | | |
| 10. | Pembimbing Akademik dapat melihat data Mahasiswa berdasarkan mahasiswa bimbingannya | ✓ | | |

| | | | | |
|--------------------------------|---|----------|----------|----------|
| 11. | Kepala Program Studi dapat melihat data mahasiswa berdasarkan mahasiswa bimbingannya | ✓ | | |
| 12. | BKAL dapat mengelola data Mahasiswa | ✓ | | |
| 13. | BKAL dapat mengelola jadwal deadline | | ✓ | |
| 14. | BKAL dapat mengelola data kategori beasiswa | ✓ | | |
| 15. | BKAL dapat mengelola data jenis beasiswa | ✓ | | |
| 16. | BKAL dapat melihat laporan beasiswa yang telah diupload | | ✓ | |
| 17. | BKAL dapat mengirimkan feedback | ✓ | | |
| 18. | BKAL dapat menerima laporan beasiswa yang telah direvisi | | ✓ | |
| 19. | BKAL dapat menyetujui laporan beasiswa yang telah diupload | ✓ | | |
| 20. | BKAL dapat melihat prestasi mahasiswa yang telah di upload | | ✓ | |
| 21. | BKAL dapat melihat tracker terhadap mahasiswa laporan tertunda | ✓ | | |
| 22. | BKAL dapat melihat detail tracker terhadap mahasiswa laporan tertunda | | ✓ | |
| 23. | BKAL dapat melakukan chat real-time kepada mahasiswa | ✓ | | |
| 24. | BKAL dapat melihat perkembangan jumlah, berjalan, tertunda, selesai (beasiswa) | ✓ | | |
| 25. | BKAL dapat melihat detail perkembangan beasiswa | ✓ | | |
| 26. | BKAL dapat melakukan rekap perkembangan beasiswa | ✓ | | |
| 27. | BKAL dapat menerima notifikasi laporan & chat | ✓ | | |
| 28. | Wakil Rektor, Kepala Program Studi, dan Pembimbing Akademik dapat melihat laporan beasiswa yang telah diupload | ✓ | | |
| 29. | Wakil Rektor, Kepala Program Studi, dan Pembimbing Akademik dapat mengirimkan feedback | ✓ | | |
| 30. | Wakil Rektor, Kepala Program Studi, dan Pembimbing Akademik dapat menyetujui laporan beasiswa yang telah diupload | ✓ | | |
| 31. | Wakil Rektor, Kepala Program Studi, dan Pembimbing Akademik dapat menerima notifikasi laporan & chat | ✓ | | |
| Non Functional | | | | |
| Saya Ingin Sistem Dapat | | M | D | I |
| 1. | Sistem memiliki UI/UX yang ramah <i>user</i> | ✓ | | |
| 2. | <i>Password</i> minimal 8 karakter yang terdiri dari kombinasi huruf, | | ✓ | |

| | | | | |
|----|--|--|---|--|
| | simboldan angka | | | |
| 3. | Sistem harus bisa memberikan respon yang cepat | | ✓ | |

Elisitasi Tahap III

Keterangan:

T = Technical (Teknisi)

O = Operation (Operasional)

E = Economic (Ekonomis)

H = High (Tinggi)

M = Middle (Sedang)

L = Low (Rendah)

| Kebutuhan Functional | | | | | | | | | | |
|-------------------------|--|---|---|---|---|---|---|---|---|---|
| Analisis Kebutuhan | | T | | | O | | | E | | |
| Saya ingin Sistem Dapat | | H | M | L | H | M | L | H | M | L |
| 1. | Seluruh user melakukan login | | ✓ | | | | ✓ | | | ✓ |
| 2. | Seluruh user dapat mengelola data profil | | | ✓ | | ✓ | | | ✓ | |
| 3. | Seluruh user melakukan logout | | | ✓ | | | ✓ | | | ✓ |
| 4. | Mahasiswa dapat mengisi form laporan beasiswa | | ✓ | | | ✓ | | | ✓ | |
| 5. | Mahasiswa dapat melihat proses permohonan pengajuan laporan | | ✓ | | | ✓ | | | ✓ | |
| 6. | Mahasiswa dapat menerima feedback | | | ✓ | | ✓ | | | ✓ | |
| 7. | Mahasiswa dapat merubah/merevisi laporan yang telah disubmit | | ✓ | | | ✓ | | | ✓ | |
| 8. | Mahasiswa dapat melakukan chat real-time | | ✓ | | | ✓ | | | ✓ | |

| | | | | | | | | | |
|-----|--|--|---|---|--|---|---|---|---|
| 9. | Mahasiswa dapat menerima notifikasi laporan & chat telah disetujui | | ✓ | | | ✓ | | | ✓ |
| 10. | Pembimbing Akademik dapat melihat data Mahasiswa berdasarkan mahasiswa bimbingannya | | ✓ | | | ✓ | | ✓ | |
| 11. | Kepala Program Studi dapat melihat data mahasiswa berdasarkan mahasiswa bimbingannya | | ✓ | | | ✓ | | ✓ | |
| 12. | BKAL dapat mengelola data Mahasiswa | | ✓ | | | ✓ | | ✓ | |
| 13. | BKAL dapat mengelola jadwal deadline | | | ✓ | | | ✓ | | ✓ |
| 14. | BKAL dapat mengelola data kategori beasiswa | | | ✓ | | ✓ | | | ✓ |
| 15. | BKAL dapat mengelola data jenis beasiswa | | | ✓ | | | ✓ | | ✓ |
| 16. | BKAL dapat melihat laporan beasiswa yang telah diupload | | | ✓ | | ✓ | | | ✓ |
| 17. | BKAL dapat mengirimkan feedback | | | ✓ | | ✓ | | | ✓ |
| 18. | BKAL dapat menerima laporan beasiswa yang telah direvisi | | ✓ | | | ✓ | | | ✓ |
| 19. | BKAL dapat menyetujui laporan beasiswa yang telah diupload | | | ✓ | | ✓ | | | ✓ |
| 20. | BKAL dapat melihat prestasi mahasiswa yang telah di upload | | | ✓ | | ✓ | | ✓ | |
| 21. | BKAL dapat melihat tracker terhadap mahasiswa laporan tertunda | | ✓ | | | ✓ | | ✓ | |
| 22. | BKAL dapat melihat detail tracker terhadap mahasiswa laporan tertunda | | ✓ | | | ✓ | | ✓ | |
| 23. | BKAL dapat melakukan chat real-time kepada mahasiswa | | ✓ | | | ✓ | | ✓ | |
| 24. | BKAL dapat melihat perkembangan | | | ✓ | | ✓ | | | ✓ |

| | | | | | | | | | | |
|--------------------------------|---|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| | jumlah, berjalan, tertunda, selesai (beasiswa) | | | | | | | | | |
| 25. | BKAL dapat melihat detail perkembangan beasiswa | | | ✓ | | ✓ | | | | ✓ |
| 26. | BKAL dapat melakukan rekap perkembangan beasiswa | | ✓ | | | ✓ | | | ✓ | |
| 27. | BKAL dapat menerima notifikasi laporan & chat | | | ✓ | | | ✓ | | ✓ | |
| 28. | Wakil Rektor, Kepala Program Studi, dan Pembimbing Akademik dapat melihat laporan beasiswa yang telah diupload | | ✓ | | | ✓ | | | | ✓ |
| 29. | Wakil Rektor, Kepala Program Studi, dan Pembimbing Akademik dapat mengirimkan feedback | | ✓ | | | | | | | |
| 30. | Wakil Rektor, Kepala Program Studi, dan Pembimbing Akademik dapat menyetujui laporan beasiswa yang telah diupload | | ✓ | | | | ✓ | | ✓ | |
| 31. | Wakil Rektor, Kepala Program Studi, dan Pembimbing Akademik dapat menerima notifikasi laporan & chat | | ✓ | | | ✓ | | | ✓ | |
| Non Functional | | | | | | | | | | |
| Saya Ingin Sistem Dapat | | M | D | I | M | D | I | M | D | I |
| 1. | Sistem memiliki UI/UX yang ramah <i>user</i> | | ✓ | | | ✓ | | | ✓ | |
| 2. | <i>Password</i> minimal 8 karakter yang terdiri dari kombinasi huruf, simbol dan angka | | ✓ | | | ✓ | | | ✓ | |
| 3. | Sistem harus bisa memberikan respon yang cepat | | ✓ | | | ✓ | | | ✓ | |

| Analisis Kebutuhan | |
|-------------------------|--|
| Kebutuhan Functional | |
| Saya ingin Sistem Dapat | |
| 1. | Seluruh user melakukan login |
| 2. | Seluruh user dapat mengelola data profil |
| 3. | Seluruh user melakukan logout |
| 4. | Mahasiswa dapat mengisi form laporan beasiswa |
| 5. | Mahasiswa dapat melihat proses permohonan pengajuan laporan |
| 6. | Mahasiswa dapat menerima feedback |
| 7. | Mahasiswa dapat merubah/merevisi laporan yang telah disubmit |
| 8. | Mahasiswa dapat melakukan chat real-time |
| 9. | Mahasiswa dapat menerima notifikasi laporan & chat telah disetujui |
| 10. | Pembimbing Akademik dapat melihat data Mahasiswa berdasarkan mahasiswa bimbingannya |
| 11. | Kepala Program Studi dapat melihat data mahasiswa berdasarkan mahasiswa bimbingannya |
| 12. | BKAL dapat mengelola data Mahasiswa |
| 13. | BKAL dapat mengelola jadwal deadline |
| 14. | BKAL dapat mengelola data kategori beasiswa |
| 15. | BKAL dapat mengelola data jenis beasiswa |
| 16. | BKAL dapat melihat laporan beasiswa yang telah diupload |
| 17. | BKAL dapat mengirimkan feedback |
| 18. | BKAL dapat menerima laporan beasiswa yang telah direvisi |
| 19. | BKAL dapat menyetujui laporan beasiswa yang telah diupload |
| 20. | BKAL dapat melihat prestasi mahasiswa yang telah di upload |
| 21. | BKAL dapat melihat tracker terhadap mahasiswa laporan tertunda |
| 22. | BKAL dapat melihat detail tracker terhadap mahasiswa laporan tertunda |
| 23. | BKAL dapat melakukan chat real-time kepada mahasiswa |
| 24. | BKAL dapat melihat perkembangan jumlah, berjalan, tertunda, selesai (beasiswa) |
| 25. | BKAL dapat melihat detail perkembangan beasiswa |
| 26. | BKAL dapat melakukan rekap perkembangan beasiswa |

| | |
|--------------------------------|---|
| 27. | BKAL dapat menerima notifikasi laporan & chat |
| 28. | Wakil Rektor, Kepala Program Studi, dan Pembimbing Akademik dapat melihat laporan beasiswa yang telah diupload |
| 29. | Wakil Rektor, Kepala Program Studi, dan Pembimbing Akademik dapat mengirimkan feedback |
| 30. | Wakil Rektor, Kepala Program Studi, dan Pembimbing Akademik dapat menyetujui laporan beasiswa yang telah diupload |
| 31. | Wakil Rektor, Kepala Program Studi, dan Pembimbing Akademik dapat menerima notifikasi laporan & chat |
| Non Functional | |
| Saya Ingin Sistem Dapat | |
| 1. | Sistem memiliki UI/UX yang ramah <i>user</i> |
| 2. | <i>Password</i> minimal 8 karakter yang terdiri dari kombinasi huruf, simbol dan angka |
| 3. | Sistem harus bisa memberikan respon yang cepat |

BAB IV

BIAYA DAN JADWAL KEGIATAN

4.1 Analisa Perencanaan Sistem

Pelaksanaan kegiatan akan dilakukan dengan mengacu kepada tahapan pengembangan metodologi *Rapid Application Development* (RAD) yang sudah dijelaskan pada BAB III. Seluruh kegiatan akan dilakukan sesuai dengan setiap tahapannya dari awal hingga akhir supaya bisa mendapatkan hasil yang diharapkan.

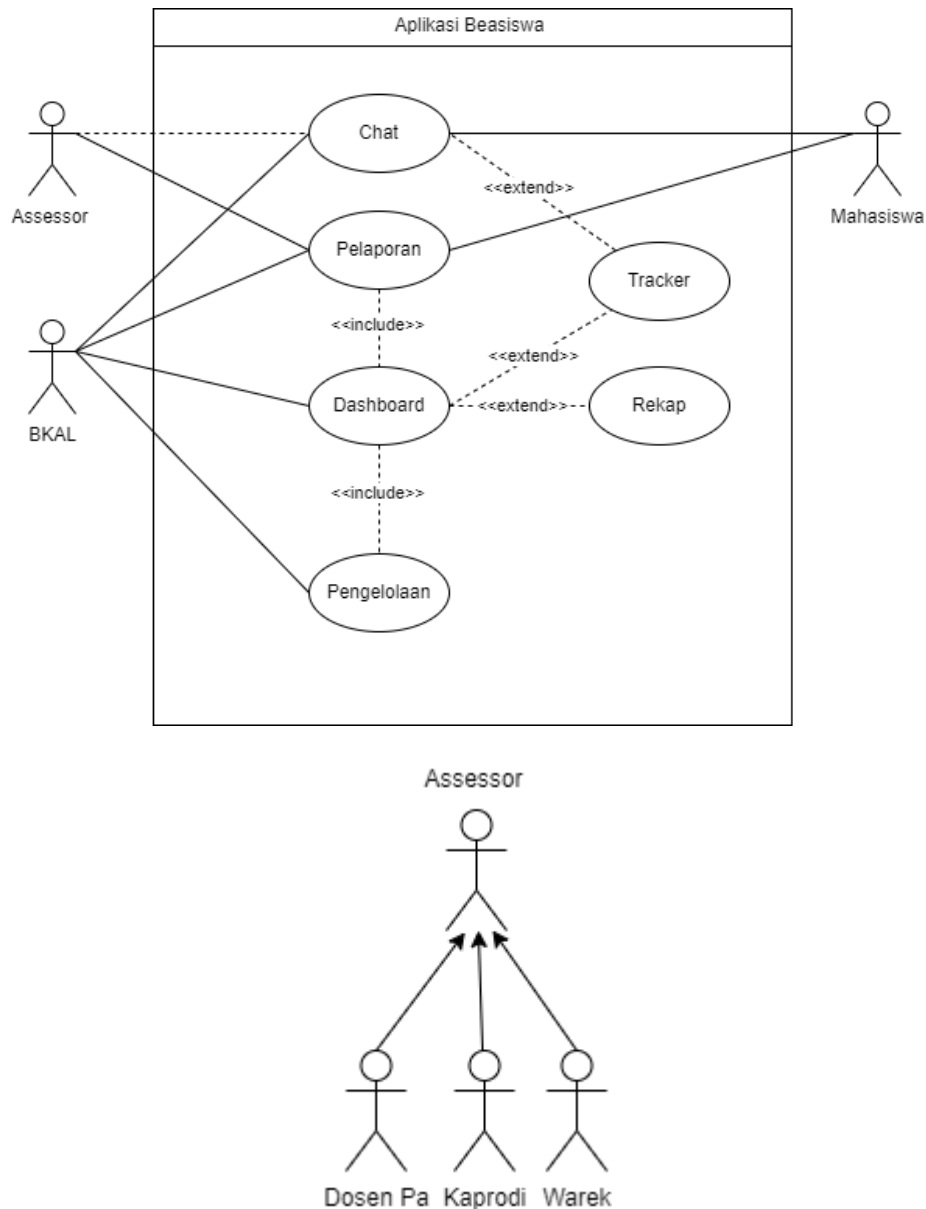
| Tahap | Kegiatan | Bulan 1 | | Bulan 2 | | | | Bulan 3 | | | | Bulan 4 | | | |
|------------------------------------|----------------------------|---------|---|---------|---|---|---|---------|---|---|---|---------|---|---|---|
| | | 1 | 2 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 |
| Tahap <i>Requirements Planning</i> | | | | | | | | | | | | | | | |
| | Identifikasi Kebutuhan | | | | | | | | | | | | | | |
| | Pengumpulan Data | | | | | | | | | | | | | | |
| Tahap <i>User Design</i> | | | | | | | | | | | | | | | |
| | Perancangan UML | | | | | | | | | | | | | | |
| | <i>Design UI/UX</i> | | | | | | | | | | | | | | |
| | Pembuatan <i>Prototype</i> | | | | | | | | | | | | | | |
| | <i>Feedback User</i> | | | | | | | | | | | | | | |
| Tahap <i>Rapid construction</i> | | | | | | | | | | | | | | | |
| | Pembuatan <i>Database</i> | | | | | | | | | | | | | | |
| | Pemrograman Sistem | | | | | | | | | | | | | | |
| | <i>Feedback User</i> | | | | | | | | | | | | | | |
| Tahap <i>Implementation</i> | | | | | | | | | | | | | | | |
| | Implementasi | | | | | | | | | | | | | | |
| | <i>Black Box Testing</i> | | | | | | | | | | | | | | |

4.2 Perancangan Diagram Sistem Usulan

Perancangan sistem yang diusulkan berkaitan dengan permasalahan yang terjadi pada saat pelaporan beasiswa unggulan dilakukan, seperti pada saat proses pembuatan laporan, approval laporan, revisi laporan dan pengelolaan data laporan. Analisa mengenai sistem pelaporan beasiswa akan digambarkan melalui diagram Unified Modelling Language (UML) seperti Use Case Diagram, Spesifikasi Use Case, Activity Diagram, Sequence Diagram dan Class Diagram.

4.2.1 Perancangan Use Case Diagram

Deskripsi use case bertujuan untuk memberikan gambaran terperinci mengenai cara sebuah use case dalam sistem perangkat lunak berinteraksi dengan sistem atau aktor. Misalnya, dalam use case aplikasi pelaporan beasiswa, deskripsi ini akan menjelaskan bagaimana sistem tersebut berperilaku ketika digunakan oleh pengguna untuk melaporkan atau mengelola informasi terkait beasiswa.



4.2.2 Perancangan Use Case Description

Deskripsi use case digunakan untuk secara detail menggambarkan bagaimana sebuah use case di dalam sistem seharusnya berinteraksi dengan sistem atau pengguna lainnya. Sebagai contoh, mari kita jelaskan use case untuk aplikasi pelaporan beasiswa:

| | | |
|--------------------------|--|--|
| Use Case Name | Pelaporan Mahasiswa | |
| Actor | Mahasiswa, Asesor, dan BKAL | |
| Deskripsi | Mahasiswa dapat membuat laporan beasiswa kemudian dapat disetujui oleh asesor dan BKAL | |
| Pre-Condition | User melakukan login lalu memilih menu data laporan | |
| Post-Condition | Mahasiswa dapat membuat laporan kemudian BKAL dan asesor dapat melihat detail serta menyetujui laporan-laporan yang telah diajukan oleh mahasiswa penerima beasiswa. | |
| Secenario | User | System |
| Normal Course | 1. Akses menu pelaporan | |
| | | 2. Menampilkan halaman laporan sesuai <i>role user</i> |
| | 3. Mahasiswa mengisi form pelaporan beasiswa | |
| | 4. Mahasiswa menekan tombol submit | |
| | | 5. Sistem menyimpan data laporan mahasiswa |
| | 6. Asesor dan BKAL mengecek laporan mahasiswa | |
| | 7. Asesor dan BKAL menekan tombol approve | |
| | | 8. Mengirimkan notifikasi laporan diterima |
| | 9. Mahasiswa menerima notifikasi | |
| Alternative Flows | Mahasiswa harus melakukan revisi apabila laporan tidak di “approve” oleh asesor atau BKAL. | |

| | |
|----------------------|-------------|
| Use Case Name | Pengelolaan |
| Actor | BKAL |

| | | |
|--------------------------|---|------------------------------------|
| Deskripsi | Mengelola data mahasiswa penerima beasiswa | |
| Pre-Condition | BKAL melakukan login lalu memilih menu data mahasiswa | |
| Post-Condition | BKAL dapat tambah, edit, dan hapus data mahasiswa | |
| Secenario | User | System |
| Normal Course | 1. Akses data mahasiswa | |
| | | 2. Menampilkan data mahasiswa |
| | | |
| | 2. BKAL melakukan import data mahasiswa | |
| | 3. Melakukan input edit data mahasiswa | |
| | | 4. Sistem menyimpan data mahasiswa |
| | 5. BKAL melakukan hapus data mahasiswa | |
| | | 6. Sistem menghapus data mahasiswa |
| | 7. Melihat data mahasiswa | |
| Alternative Flows | Terdapat pesan saat selesai melakukan kelola data mahasiswa | |

| | | |
|-----------------------|---|---------------|
| Use Case Name | Dashboard - Tracker | |
| Actor | BKAL | |
| Deskripsi | Melacak perkembangan data mahasiswa penerima beasiswa | |
| Pre-Condition | BKAL melakukan login lalu memilih menu data laporan | |
| Post-Condition | Mahasiswa membuat laporan kemudian BKAL dapat melacak / tracket data mahasiswa penerima beasiswa. | |
| Secenario | User | System |

| | | |
|--------------------------|---|---|
| Normal Course | 1. Akses data tracker | |
| | | 2. Menampilkan data tracker |
| | 3. BKAL melihat detail data tracker berdasarkan bulan | |
| | | 3. Menampilkan data tracker berdasarkan bulan |
| | 4. BKAL akses detail data mahasiswa | |
| | | 5. Menampilkan status dan feedback |
| | 6. BKAL akses “hubungi mahasiswa” | |
| | | 7. Menampilkan halaman Chat |
| | 8. BKAL dapat Chat mahasiswa | |
| Alternative Flows | Apabila data pada bulan tidak ada, maka grafik linear tidak ada dan tidak dapat diakses | |

| | | |
|-----------------------|---|---------------|
| Use Case Name | Dashboard - Rekap | |
| Actor | BKAL | |
| Deskripsi | Melihat perkembangan data mahasiswa penerima beasiswa | |
| Pre-Condition | BKAL melakukan login lalu memilih menu data laporan | |
| Post-Condition | Mahasiswa membuat laporan kemudian BKAL dapat melihat perkembang seluruh mahasiswa penerima beasiswa. | |
| Secenario | User | System |
| Normal Course | | |
| | | |
| | | |
| | | |
| | | |

| | | |
|--------------------------|--|--|
| | | |
| | | |
| Alternative Flows | | |

| | | |
|--------------------------|-------------|---------------|
| Use Case Name | | |
| Actor | | |
| Deskripsi | | |
| Pre-Condition | | |
| Post-Condition | | |
| Secenario | User | System |
| Normal Course | | |
| Alternative Flows | | |

4.2.3 Perancangan Activity Diagram

Deskripsi use case bertujuan untuk memberikan gambaran terperinci mengenai cara sebuah use case dalam sistem perangkat lunak berinteraksi dengan sistem atau aktor. Misalnya, dalam use case aplikasi pelaporan beasiswa, deskripsi ini akan menjelaskan bagaimana sistem tersebut berperilaku ketika digunakan oleh pengguna untuk melaporkan atau mengelola informasi terkait beasiswa.

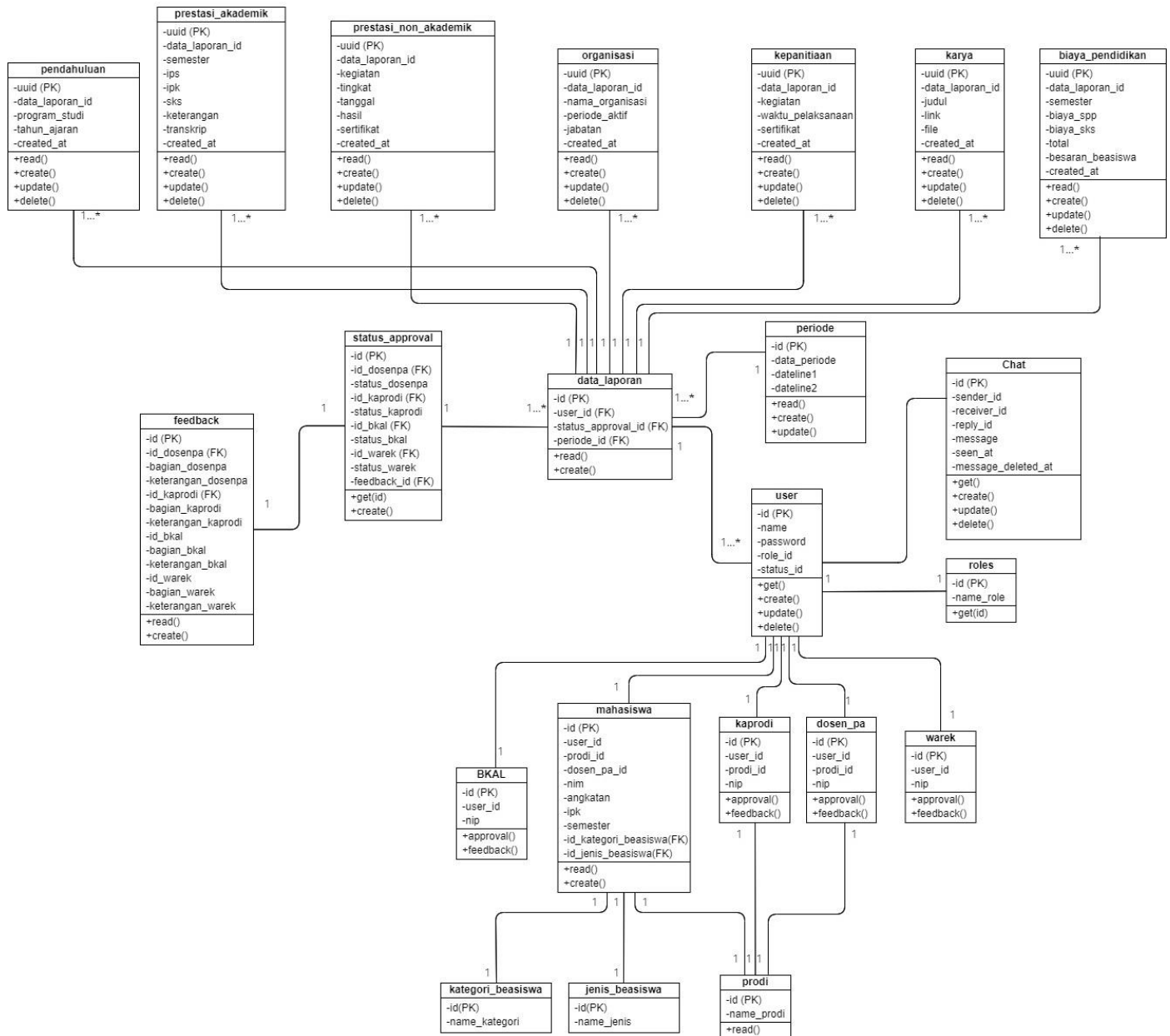
4.2.4 Perancangan Sequence Diagram

asd

4.2.5 Perancangan Class Diagram

Deskripsi use case bertujuan untuk memberikan gambaran terperinci mengenai cara sebuah use case dalam sistem perangkat lunak berinteraksi dengan sistem atau aktor. Misalnya, dalam use case aplikasi pelaporan beasiswa, deskripsi ini akan menjelaskan

bagaimana sistem tersebut berperilaku ketika digunakan oleh pengguna untuk melaporkan atau mengelola informasi terkait beasiswa.



4.3 Perancangan Basis Data

Deskripsi use case bertujuan untuk memberikan gambaran terperinci mengenai cara sebuah use case dalam sistem perangkat lunak berinteraksi dengan sistem atau aktor. Misalnya, dalam use case aplikasi pelaporan beasiswa, deskripsi ini akan menjelaskan

bagaimana sistem tersebut berperilaku ketika digunakan oleh pengguna untuk melaporkan atau mengelola informasi terkait beasiswa.

Struktur Database Data Laporan

| Nama Field | Type | Size | Keterangan |
|--------------------|------|------|--|
| id | int | 11 | Primary key data_laporan |
| user_id | int | 11 | Foreign key dari table user |
| status_approval_id | int | 11 | Foreign key dari table status_approval |
| periode_id | int | 11 | Foreign key dari table periode |

Struktur Database Periode

| Nama Field | Type | Size | Keterangan |
|--------------|---------|------|-------------------------------|
| id | int | 11 | Primary key periode |
| data_periode | varchar | 255 | Data pada periode gasal/genap |
| dateline1 | date | - | Dateline pembukaan |
| dateline2 | date | - | Dateline akhir |

Struktur Status Approval

| Nama Field | Type | Size | Keterangan |
|----------------|---------|------|---------------------------------|
| id | int | 11 | Primary key data_laporan |
| id_dosenpa | int | 11 | Foreign key dari table dosenpa |
| status_dosenpa | varchar | 50 | Status dosenpa |
| id_kapordi | int | 11 | Foreign key dari table kaprodi |
| status_kaprodi | varchar | 50 | Status kaprodi |
| id_bkal | int | 11 | Foreign key dari table bkal |
| status_bkal | varchar | 50 | Status bkal |
| id_warek | int | 11 | Foreign key dari table warek |
| status_warek | varchar | 50 | Status warek |
| feedback_id | int | 11 | Foreign key dari table feedback |

Struktur Status Approval

| Nama Field | Type | Size | Keterangan |
|----------------|---------|------|---------------------------------|
| id | int | 11 | Primary key status_approval |
| id_dosenpa | int | 11 | Foreign key dari table dosenpa |
| status_dosenpa | varchar | 50 | Status dosenpa |
| id_kapordi | int | 11 | Foreign key dari table kaprodi |
| status_kaprodi | varchar | 50 | Status kaprodi |
| id_bkal | int | 11 | Foreign key dari table bkal |
| status_bkal | varchar | 50 | Status bkal |
| id_warek | int | 11 | Foreign key dari table warek |
| status_warek | varchar | 50 | Status warek |
| feedback_id | int | 11 | Foreign key dari table feedback |

Struktur Feedback

| Nama Field | Type | Size | Keterangan |
|--------------------|---------|------|--------------------------------|
| id | int | 11 | Primary key feedback |
| id_dosenpa | int | 11 | Foreign key dari table dosenpa |
| bagian_dosenpa | varchar | 50 | Bagian dosenpa |
| keterangan_dosenpa | varchar | 255 | Keterangan dosenpa |
| id_kapordi | int | 11 | Foreign key dari table kaprodi |
| bagian_kaprodi | varchar | 50 | Bagian kaprodi |
| keterangan_kaprodi | varchar | 255 | Keterangan kaprodi |
| id_bkal | int | 11 | Foreign key dari table bkal |
| bagian_bkal | varchar | 50 | Bagian bkal |
| keterangan_bkal | varchar | 255 | Keterangan bkal |
| id_warek | int | 11 | Foreign key dari table warek |
| bagian_warek | varchar | 50 | Bagian warek |

| | | | |
|------------------|---------|-----|------------------|
| keterangan_warek | varchar | 255 | Keterangan warek |
|------------------|---------|-----|------------------|

Struktur Database Pendahuluan

| Nama Field | Type | Size | Keterangan |
|-----------------|---------|------|-------------------------------------|
| uuid | varchar | 255 | Unik ID pendahulaun |
| data_laporan_id | int | 11 | Foreign key dari table data_laporan |
| program_studi | varchar | 50 | Nama program studi |
| tahun_ajaran | varchar | 20 | Tahun ajaran saat pelaporan dibuat |

Struktur Database Prestasi Akademik

| Nama Field | Type | Size | Keterangan |
|-----------------|---------|------|-------------------------------------|
| uuid | varchar | 255 | Unik ID prestasi_akademik |
| data_laporan_id | int | 11 | Foreign key dari table data_laporan |
| semester | varchar | 20 | Semester dibuatnya laporan |
| ips | varchar | 20 | Jumlah IPS semester |
| ipk | varchar | 20 | Jumlah IPK semester |
| sks | varchar | 20 | Jumlah SKS semester |
| keterangan | varchar | 255 | Keterangan |
| transkrip | varchar | 255 | Link transkrip |

Struktur Database Prestasi Non Akademik

| Nama Field | Type | Size | Keterangan |
|-----------------|---------|------|-------------------------------------|
| uuid | varchar | 255 | Unik ID prestasi_non_akademik |
| data_laporan_id | int | 11 | Foreign key dari table data_laporan |
| kegiatan | varchar | 255 | Nama kegiatan |
| tingkat | varchar | 100 | Tingkat prestasi |
| tanggal | date | - | Tanggal prestasi didapatkan |
| hasil | varchar | 255 | Hasil |
| sertifikat | varchar | 255 | Link sertifikat |

Struktur Database Organisasi

| Nama Field | Type | Size | Keterangan |
|-----------------|---------|------|-------------------------------------|
| uuid | varchar | 255 | Unik ID organisasi |
| data_laporan_id | int | 11 | Foreign key dari table data_laporan |
| nama_organisasi | varchar | 255 | Nama organisasi yang diikuti |
| periode_aktif | date | - | Periode aktif organisasi |
| jabatan | varchar | 100 | Jabatan saat menjalani organisasi |

Struktur Database Kepanitiaan

| Nama Field | Type | Size | Keterangan |
|-------------------|---------|------|-------------------------------------|
| uuid | varchar | 255 | Unik ID kepanitiaan |
| data_laporan_id | int | 11 | Foreign key dari table data_laporan |
| kegiatan | varchar | 255 | Nama kegiatan |
| waktu_pelaksanaan | varchar | 255 | Waktu pelaksanaan acara |
| sertifikat | varchar | 255 | Link sertifikat |

Struktur Database Karya

| Nama Field | Type | Size | Keterangan |
|-----------------|---------|------|-------------------------------------|
| uuid | varchar | 255 | Unik ID karya |
| data_laporan_id | int | 11 | Foreign key dari table data_laporan |
| judul | varchar | 255 | Nama judul karya |
| link | varchar | 255 | Link karya |
| file | varchar | 255 | File karya |

Struktur Database User

| Nama Field | Type | Size | Keterangan |
|------------|------|------|-----------------------------|
| id_user | int | 11 | Primary key user |
| id_role | int | 11 | Foreign key dari table role |

| Nama Field | Type | Size | Keterangan |
|------------|---------|------|---------------|
| username | varchar | 255 | username |
| password | varchar | 255 | Password user |

Struktur Database Roles

| Nama Field | Type | Size | Keterangan |
|------------|---------|------|------------------|
| id_role | int | 11 | Primary key role |
| name_role | varchar | 255 | Nama role |

Struktur Database User

| Nama Field | Type | Size | Keterangan |
|------------|---------|------|------------------------------|
| id_user | int | 11 | Primary key User |
| username | varchar | 255 | Bagian yang terkena feedback |
| password | varchar | 255 | Isi feedback |
| role_id | varchar | 255 | Foregnkey dari tabel roles |
| status_id | varchar | 255 | Status keaktifan |

Struktur Database BKAL

| Nama Field | Type | Size | Keterangan |
|------------|---------|------|---------------------------|
| id_bkal | int | 11 | Primary key BKAL |
| user_id | int | 11 | Foregnkey dari tabel user |
| nama | varchar | 255 | Nama user Bkal |
| nip | varchar | 255 | Nip dari user BKAL |
| email | varchar | 255 | Email User |
| no_hp | varchar | 255 | No hp User |

Sturktur Database Kaprodi

| Nama Field | Type | Size | Keterangan |
|------------|------|------|---------------------------|
| id_kaprodi | int | 11 | Primary key kaprodi |
| user_id | int | 11 | Foregnkey dari tabel user |

| | | | |
|-------|---------|-----|------------------------|
| nama | varchar | 255 | nama dari user kaprodi |
| nip | varchar | 255 | Nip User Kaprodi |
| no_hp | varchar | 255 | No hp User |

Struktur Database Warek

| Nama Field | Type | Size | Keterangan |
|------------|---------|------|---------------------------|
| id_warek | int | 11 | Primary key warek |
| user_id | int | 11 | Foregnkey dari tabel user |
| nama | varchar | 255 | nama dari user warek |
| nip | varchar | 255 | Nip User warek |
| no_hp | varchar | 255 | No hp User |

Struktur Database dosen_pa

| Nama Field | Type | Size | Keterangan |
|------------|---------|------|---------------------------|
| id_dosen | int | 11 | Primary key dosen |
| user_id | int | 11 | Foregnkey dari tabel user |
| nama | varchar | 255 | nama dari user dosen |
| nip | varchar | 255 | Nip User dosen |
| no_hp | varchar | 255 | No hp User |

Struktur Database Roles

| Nama Field | Type | Size | Keterangan |
|------------|---------|------|------------------|
| id_role | int | 11 | Primary key role |
| name_role | varchar | 255 | Nama role |

4.4 Perancangan Antarmuka

4.5 Perancangan Implementasi

4.5.1 Implementasi Sistem

Asd

4.5.2 Skenario Pengujian

Dalam tahap implementasi dilakukan testing yang berguna dalam menentukan bahwa aplikasi yang dibangun telah memenuhi syarat dan tidak terdapat bug yang dapat mengganggu jalannya sistem. Metode yang digunakan dalam melakukan testing untuk aplikasi adalah black box testing. Dalam pengujian dalam black box yang menjadi fokus dari pengujian adalah input dan output sistem serta memeriksa apakah respons sistem sudah memenuhi kriteria yang telah ditentukan. Berikut ini merupakan rancangan testing dari aplikasi pelaporan beasiswa unggulan.

[illegible]

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and extend across the width of the page. There are no margins, text, or other markings on the paper.

BAB V

PENUTUP

Awdawd

Awd

Awd

awd

DAFTAR PUSTAKA

- Setiawan, D. B. (2023). Implementasi Program Studi Independ Implementasi Program Studi Independen MBKM: Pengembangan Dashboard Interaktif Covid-19 sebagai Alat Informasi dan Pendidikan Masyarakat dalam Penanganan Pandemi Studi Kasus di PT. Inovasi Lentera Cipta Kreasi. *Inisiatif: Jurnal Dedikasi Pengabdian Masyarakat*, 2(2), 117-132.
- Yanti, L. (2019). Pengembangan Dashboard Universitas Ibnu Sina Batam Berbasis Web Mobile. *JR: Jurnal Responsive Teknik Informatika*, 3(02), 80-94.
- Wahyuni, T. D. (2023). Aplikasi “Si Pintar” untuk Pencegahan Covid-19. Penerbit NEM. Reynaldo, D. (2019). *Beasiswa: Panduan Mendapatkan Beasiswa Pendidikan Dalam dan Luar Negeri*. Prenada Media Group.
- Pane, S. F., Hardy, I. H., & Sujadi, E. C. (2020). Pengembangan Smart Conveyor Pada Tracking Barang Berbasis IOT (Vol. 1). Kreatif.
- Kom, E. S. E. S., Kom, M., Pratomo, A. B., Kom, S., Chrisantus Trisianto, S. T., Kom, M., ... & Kom, M. (2023). *Konsep Dasar Pengenalan Database Rumpun Ilmu Komputer*. Cendikia Mulia Mandiri.
- Hasanudin, D., Andarsyah, R., Prianto, C., Awangga, R. M., & Habibi, R. (2020). Sistem pendukung keputusan tentang beasiswa menggunakan metode Fuzzy MADM dan SAW. Kreatif.
- Anamisa, D. R., & Mufarroha, F. A. (2022). *Dasar Pemrograman WEB Teori dan Implementasi: HTML, CSS, Javascript, Bootstrap, CodeIgniter*. Media Nusa Creative (MNC Publishing).
- Abdulloh, R. (2020). *Menguasai React JS Untuk Pemula: Panduan belajar JavaScript dari dasar hingga membuat aplikasi web modern* (Vol. 1). Rohi Abdulloh.
- Habibi, R., Putra, F. B., & Putri, I. F. (2020). *Aplikasi kehadiran dosen menggunakan PHP OOP* (Vol. 1). Kreatif.
- Rupiah, J., Nagari, D. S. S. A., Barung, P. K. M. D. K., & Balantai, B. Adi, AP 2020. *Panduan Cepat Belajar HTML, PHP, & MYSQL*. Jakarta: PT Elex Media Komputindo. Anhar. 2010. *Panduan Menguasai PHP & MySQL Secara Otodidak*. Jakarta Selatan: Mediakita. Anhar. 2016. *Panduan Belajar Bijak Internet Untuk Anak*. Adamssein Media.
- Namruddin, R., Basalamah, A., Ali, M. Z. A., Syarifuddin, A., Alam, S., Wardhani, N., & Abdurrahman, T. S. D. (2023). *BELAJAR DATABASE DENGAN MUDAH MENGGUNAKAN MYSQL*. TOHAR MEDIA.
- Abdulloh, R. (2022). *7 Materi Pemrograman Web untuk Pemula 1: HTML, CSS, & MariaDB*. PT Elex Media Komputindo.
- Ahmad, N., Krisnanik, E., Rupilele, F. G. J., Muliawati, A., Syamsiyah, N., Kraugusteeliana, K., ... & Guntoro, G. (2022). *Analisa & Perancangan Sistem Informasi Berorientasi Objek*. Penerbit Widina.
- Habibi, R., & Aprilian, R. (2020). *Tutorial dan penjelasan aplikasi e-office berbasis web menggunakan metode RAD* (Vol. 1). Kreatif.
- Darman, R. (2018). Pembangunan Dashboard Lokasi Rawan Tanah Longsor di Indonesia Menggunakan Tableau. *Jurnal Teknik Informatika dan Sistem Informasi*, 4(2), 256-269.

Umar, M. K. G., Sabtu, J., & Sukur, R. S. (2022). *Implementasi Metode Rapid Application Development (Rad) Dalam Rancangan Sistem Informasi Pelayanan Administrasi Di Kelurahan Tabam Kota Ternate. Jurnal Teknoinfo*, 16(2), 277-290.

Sugiyono, P. D. (2019). *Metode penelitian bisnis: pendekatan kuantitatif, kualitatif, kombinasi, dan R&D. Penerbit CV. Alfabeta: Bandung*, 225(87), 48-61.

LAMPIRAN-LAMPIRAN

Lampiran 1. Justifikasi Anggaran Kegiatan

| NO | Material | Volume | | Harga Satuan (Rp) | Total (Rp) | Keterangan |
|----|------------------------------|--------|-----|-------------------|------------|--|
| | | JML | SAT | | | |
| 1. | Barang Habis Pakai | | | | | |
| | a. Sewa hosting | 1 | Thn | 1.500.000 | 1.500.000 | Server website |
| | b. Sewa domain | 1 | Thn | 250.000 | 250.000 | Domain website |
| | c. Kuota Internet | 3 | Bln | 100.000 | 300.000 | |
| | d. Kertas HVS | 1 | Rim | 60.000 | 60.000 | Laporan aplikasi dan dokumen pendukung lainnya |
| | e. Tinta Printer | 4 | Pcs | 70.000 | 280.000 | Cetak laporan pendukung |
| | f. Materai | 6 | Pcs | 10.000 | 60.000 | Legalitas proposal dan dokumen lainnya |
| | SUB TOTAL | | | | 2.310.000 | |
| 2. | Perjalanan | | | | | |
| | a. Transportasi | 6 | | 50.000 | 300.000 | Melakukan survei dan pengumpulan berkas |
| | SUB TOTAL | | | | 300.000 | |
| 3. | Peralatan penunjang | | | | | |
| | a. Upgrade komponen komputer | 1 | | 800.000 | 800.000 | Peningkatan performa untuk pembuatan aplikasi |
| | SUB TOTAL | | | | 800.000 | |