**PROPOSAL PROYEK AKHIR**

**PENGEMBANGAN SISTEM INFORMASI WORK ORDER BERBASIS WEBSITE MENGGUNAKAN WATERFALL**

**(STUDI KASUS: SARANA DAN UMUM**

**POLITEKNIK CALTEX RIAU)**

****

**Disusun oleh:**

**Mario Dimas Putra Prabowo**

**NIM. 2055301077**

**PROGRAM STUDI TE­KNIK INFORMATIKA**

**POLITEKNIK CALTEX RIAU**

**2024**

# RINGKASAN

Politeknik Caltex Riau merupakan salah satu perguruan tinggi yang berlokasi di Provinsi Riau, tepatnya di kota Pekanbaru. Saat ini Politeknik Caltex Riau memiliki beberapa bagian yang menangani kebutuhan Politeknik Caltex Riau, salah satunya adalah Sarana dan Umum atau yang lebih sering disebut sebagai Sarana dan Umum yang saat ini tidak memiliki media untuk mencatat segala pekerjaan yang telah dilakukannya. Dengan adanya permasalahan tersebut, diperlukannya sebuah sistem informasi *work order* untuk mencatat pekerjaan yang akan dan sudah dilakukan oleh pihak Sarana dan Umum. Metode penelitian yang digunakan adalah metode *User Centered Design* dimana metode ini mencakup analisis kebutuhan pengguna, perancangan sistem, implementasi dan pengujian. Diharapkan penelitian ini dapat memberikan dampak positif pada pihak Sarana dan Umum serta meningkatkan efesiensi dalam bekerja.

Kata kunci: Politeknik Caltex Riau, Sistem Informasi, *User Centered Design, Work Order.*

# DAFTAR ISI

[RINGKASAN i](#_Toc160433448)

[DAFTAR ISI ii](#_Toc160433449)

[DAFTAR GAMBAR iv](#_Toc160433450)

[DAFTAR TABEL v](#_Toc160433451)

[I. PENDAHULUAN 1](#_Toc160433452)

[1.1 Latar Belakang 1](#_Toc160433453)

[1.2 Rumusan Masalah 2](#_Toc160433454)

[1.3 Batasan Masalah 2](#_Toc160433455)

[1.4 Tujuan dan Manfaat 3](#_Toc160433456)

[1.4.1 Tujuan 3](#_Toc160433457)

[1.4.2 Manfaat 3](#_Toc160433458)

[1.5 Metodologi Penelitian 3](#_Toc160433459)

[1.6 Sistematika Penulisan 4](#_Toc160433460)

[II. TINJAUAN PUSTAKA 5](#_Toc160433461)

[2.1 Penelitian Terdahulu 5](#_Toc160433462)

[2.2 Landasan Teori 6](#_Toc160433463)

[2.2.1 Metode *Waterfall* 6](#_Toc160433464)

[2.2.2 Sistem Informasi 7](#_Toc160433465)

[2.2.3 *Work Order* 8](#_Toc160433466)

[2.2.4 *Website* 8](#_Toc160433467)

[2.2.5 *Hypertext PreProcessor* PHP 9](#_Toc160433468)

[2.2.6 *Framework Laravel* 9](#_Toc160433469)

[2.2.7 *Database* 9](#_Toc160433470)

[2.2.9 *Black Box Testing* 10](#_Toc160433471)

[2.2.10 *User Acceptance Testing (UAT)* 11](#_Toc160433472)

[2.2.11 *Usabilty Testing* 11](#_Toc160433473)

[III. PERANCANGAN 13](#_Toc160433474)

[3.1 Metodologi Penelitian 13](#_Toc160433476)

[3.2 Analisis Kebutuhan Sistem 13](#_Toc160433477)

[3.2.1 Hasil Wawancara 14](#_Toc160433478)

[3.2.2 Proses Bisnis 14](#_Toc160433479)

[3.3 Desain Sistem 19](#_Toc160433480)

[3.3.1 Arsitektur Sistem 19](#_Toc160433481)

[3.3.5 *Entity Relationship Diagram* (ERD) 24](#_Toc160433482)

[3.3.6 Perancangan Tabel 25](#_Toc160433483)

[3.3.7 *Wireframe* 26](#_Toc160433484)

[IV. JADWAL DAN ANGGARAN BIAYA 35](#_Toc160433485)

[4.1 Jadwal 35](#_Toc160433487)

[4.2 Perkiraan Biaya 35](#_Toc160433488)

[DAFTAR PUSTAKA 36](#_Toc160433489)

[LAMPIRAN I 1](#_Toc160433490)

[LAMPIRAN II 1](#_Toc160433491)

[LAMPIRAN III 1](#_Toc160433492)

# DAFTAR GAMBAR

[Gambar 2.1 Tahapan Metode Waterfall (Hermansyah dkk., 2023) 6](#_Toc160433284)

[Gambar 3.1 Metodologi Penelitian 13](#_Toc160421766)

[Gambar 3.2 Grafik Kendala 14](#_Toc160421767)

[Gambar 3. 3 Arsitektur Sistem 19](#_Toc160421768)

[Gambar 3.4 Use Case Diagram 20](#_Toc160421769)

[Gambar 3.5 Entity Relationship Diagram 24](#_Toc160421770)

[Gambar 3.6 Wireframe Login 27](#_Toc160421771)

[Gambar 3.7 Wireframe Daftar Permintaan Staf PCR 27](#_Toc160421772)

[Gambar 3.8 Wireframe Lapor Kerusakan 28](#_Toc160421773)

[Gambar 3.9 Wireframe Order Kegiatan 28](#_Toc160421774)

# DAFTAR TABEL

[Tabel 3.1 Tabel Identifikasi Aktor 19](#_Toc160428941)

[Tabel 3.2 Use Case Login 20](#_Toc160428942)

[Tabel 3.3 Use Case Scenario Mengelola Laporan Kerusakan 21](#_Toc160428943)

[Tabel 3.4 Use Case Scenario Mengelola Order Kegiatan 23](#_Toc160428944)

# I. PENDAHULUAN

## Latar Belakang

Sistem Informasi adalah adalah sebuah media yang dapat memberikan informasi mengenai sebuah data untuk mempermudahkan kinerja manusia dalam bekerja. Saat ini, perkembangan teknologi yang semakin pesat memaksa kita untuk menerima data dalam jumlah besar sehingga kita membutuhkan sistem untuk mengolah data tersebut menjadi informasi. Salah satu bentuk dari sistem informasi adalah *website* yang digunakan untuk memberikan informasi mengenai suatu kebutuhan untuk organisasi bahkan khalayak umum.

Sarana dan Umum adalah sebuah divisi yang menaungi kebutuhan material dari perguruan tinggi Politeknik Caltex Riau. Sarana dan Umum bertanggung jawab atas segala properti yang digunakan dalam Politeknik Caltex Riau, baik bertanggung jawab atas kerusakan yang terjadi maupun kebutuhan lainnya. Berdasarkan hasil wawancara yang telah dilakukan, Pihak Sarana dan Umum masih menggunakan media *WhatsApp* untuk menerima pekerjaan dari staf PCR karena tidak ada sistem yang dapat mencatat riwayat pekerjaan yang telah dilakukan oleh pihak Sarana dan Umum. Hal itu menyebabkan pihak Sarana dan Umum kesulitan untuk mencatat pekerjaan apa saja yang telah dilakukan.

Dari permasalahan yang telah ada, maka dirancanglah sebuah sistem berbentuk *website* dengan tujuan mencatat serta meminta pekerjaan untuk staf Politeknik Caltex Riau. Dengan dibangunnya *website* ini, diharapkan *website* ini akan sesuai dengan kebutuhan serta kenyamanan pengguna yaitu pihak Sarana dan Umum yang akan menerima pekerjaan serta melakukan pelaporan dan staf PCR yang akan meminta pekerjaan pada pihak Sarana dan Umum. Dengan ini, pihak Sarana dan Umum dapat melihat riwayat pekerjaan yang telah dilakukan ataupun pekerjaan yang akan dilakukan.

Sistem dikembangkan menggunakan metode *waterfall* dimana metode ini menerapkan pendekatan secara sistematis dan urut, mulai dari analisa kebutuhan, pemodelan sistem, implementasi, pengujian dan pemeliharaan. Metode *waterfall* memiliki tahap yang bersifat sekuensial dimana tahapan tidak dapat berlanjut apabila tahap sebelumnya belum selesai.

Metode *waterfall* dipilih karena metode *waterfall* memiliki kelebihan yang terletak pada penggunaan-nya dimana tahapan tidak dapat berlanjut apabila tahapan sebelumnya belum tuntas. Hal tersebut meminimalisir terjadinya kesalahan dan inkonsistensi antara pengembang dan pengguna. Dengan mengikuti tahapan dari metode *waterfall*, pengembang dapat bekerja secara terstruktur dan memberikan hasil yang optimal. Oleh karena itu, pada proyek akhir ini diajukan judul “Rancang Bangun Sistem Informasi & Manajemen Proyek (SIMPRO) menggunakan Metode Waterfall (Studi Kasus : Divisi TSI PT Bank Riau Kepri Syariah)”.

## Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, rumusan masalah dari proyek akhir ini adalah:

1. Bagaimana perancangan dan pembangunan Sistem Informasi *Work* *Order* pada bagian Sarana dan Umum?
2. Bagaimana penerapan metode *Waterfall* dalam perancangan dan pembangunan Sistem Informasi *Work* *Order* pada bagian Sarana dan Umum?

## Batasan Masalah

Adapun batasan masalah dalam pembuatan proyek akhir ini adalah:

1. Pengguna yang akan menggunakan sistem ini adalah staf PCR dan bagian Sarana dan Umum.
2. Sistem ini akan digunakan dalam lingkup Politeknik Caltex Riau.
3. Sistem akan dibangun berbasis *website* menggunakan *PHP Framework Laravel 10* dengan database *MySQL.*

## Tujuan dan Manfaat

### Tujuan

Adapun tujuan dari penelitian ini adalah:

1. Merancang dan membangun Sistem Informasi *Work Order* pada bagian Sarana dan Umum.
2. Mengimplementasikan metode *Waterfall* dalam perancangan dan Pembangunan Sistem Informasi *Work Order.*

### Manfaat

Adapun Manfaat yang diharapkan dari penelitian ini adalah:

1. Memudahkan pihak Sarana dan Umum dalam mengelola data pekerjaan masuk.
2. Memudahkan pihak Sarana dan Umum dalam mengelola data riwayat pekerjaan yang telah dilaksanakan.
3. Memudahkan pihak Staf Politeknik Caltex Riau dalam melihat transparasi pekerjaan yang dilakukan.

## Metodologi Penelitian

Metodologi penelitian yang dipakai dalam pembuatan proyek akhir ini adalah:

1. Studi Literatur

Pada tahap ini, dilakukan riset mengenai penelitian dengan mencari referensi berupa jurnal, buku, makalah, *paper*, dan situs web untuk mendukung pembuatan sistem.

1. Wawancara

Pada tahap ini dilakukan wawancara dengan pihak Sarana dan Umum untuk mendapatkan informasi mengenai masalah dan kebutuhan yang digunakan untuk membangun sistem.

1. Perancangan

Pada tahap ini, penelitin menyusun beberapa dokumen sebagai acuan dalam pembuatan sistem seperti perancangan *use case diagram*, *entity relational* *diagram* serta perancangan *wireframe* sistem yang akan dikembangkan.

1. Pengkodean Sistem

Pada tahap ini sistem dibangun bedasarkan desain *wireframe* yang telah dibuat. Sistem akan dibuat berbasis *website* menggunakan *PHP* *Framewrok* *Laravel 10* dan *database* server *MySQL*.

1. Pengujian

Sistem yang telah dibangun akan dilakukan pengujian untuk melihat apakah sistem dapat berjalan dengan baik. Selain itu pengujian dilakukan untuk menilai apakah sistem dapat diterima olehpengguna.

1. Penerapan dan Perawatan

Sistem yang telah dilakukan pengujian akan diterapkan dan dapat diakses oleh pengguna. Peningkatan fungsionalitas mungkin terjadi pada tahap ini.

## Sistematika Penulisan

Proposal proyek akhir ini terdiri dari empat bab, masing-masing terdiri dari beberapa sub bab. Adapun pokok pembahasan dari masing-masing bab tersebut secara garis besar adalah sebagai berikut :

**BAB I PENDAHULUAN**

Bab pertama akan membahas latar belakang masalah, perumusan masalah dan batasannya, tujuan dan manfaat, metodologi penelitian serta sistematika penulisan.

**BAB II TINJAUAN PUSTAKA**

Bab kedua akan menguraikan beberapa hasil penelitian terdahulu dan landasan teori yang diperlukan dalam perancangan sistem.

**BAB III PERANCANGAN**

Bab ketiga akan menjelaskan tentang perancangan sistem terdiri dari analisis kebutuhan sistem dan desain sistem.

**BAB IV JADWAL DAN PERKIRAAN BIAYA**

Bab keempat akan berisi informasi mengenai jadwal pengerjaan proyek akhir dan perkiraan biaya yang dibutuhkan untuk pengerjaan proyek akhir.

# II. TINJAUAN PUSTAKA

## Penelitian Terdahulu

Penelitian terdahulu dapat memberikan gambaran mengenai penelitian yang akan dijalankan. Selain itu, penelitian terdahulu dapat menjadi perbandingan atas penelitian yang akan dilaksanakan. Beberapa penelitian terdahulu yang menjadi gambaran dalam proyek ini adalah:

Penelitian yang dilakukan oleh Bagaskoro dkk. (2023) tentang Sistem Informasi Penyeleksian Calon Karyawan Menggunakan Metode Waterfall pada Alawuri Kebuli. Penelitian ini menggunakan metode *waterfall* dengan tahapan yang urut dan sistematis. Sistem dibangun dengan tujuan untuk mempermudah proses penyeleksian calon karyawan pada Alawuri Kebuli.

Penelitian yang dilakukan oleh Handayani & Salam (2023)tentang Aplikasi Sistem Informasi Simpan Pinjam Koperasi Berbasis Website Menggunakan Metode Waterfall. Penelitian dilakukan menggunakan metode *waterfall* dengan menggunakan tahapan yang urut dan sistematis dengan platform *website*. Sistem ini bertujuan untuk memudahkan proses transaksi dan pencatatan yang dilakukan oleh koperasi simpan pinjam BMT Al-Bayan.

Penelitian yang dilakukan oleh Hudhoifah dkk. (2022)tentang Sistem Informasi Pengingat Jadwal Kajian Menggunakan Metode Waterfall Berbasis Android pada Masjid Nurul Qalbi Duren Sawit. Penelitian dilakukan menggunakan metode *waterfall* dengan menggunakan tahapan yang urut dan sistematis dengan platform aplikasi *mobile.* Sistem bertujuan untuk memberikan informasi mengenai Masjid Nurul Qalbi Duren Sawit terkait jadwal kajian pada Masjid Nurul Qalbi Duren Sawit.

Penelitian yang dilakukan oleh Agung Izulhaq dkk.( 2023)tentang Sistem Informasi Pemesanan Produk Percetakan Berbasis Web Menggunakan Metode Waterfall. Penelitian dilakukan menggunakan metode *waterfall* dengan tahapan yang urut dan sistematis dengan platform *website*. Sistem bertujuan untuk memudahkan dengan cara memfasilitasi proses pemesanan produk percetakan melalui situs *website* pada Jalanpintas.art.

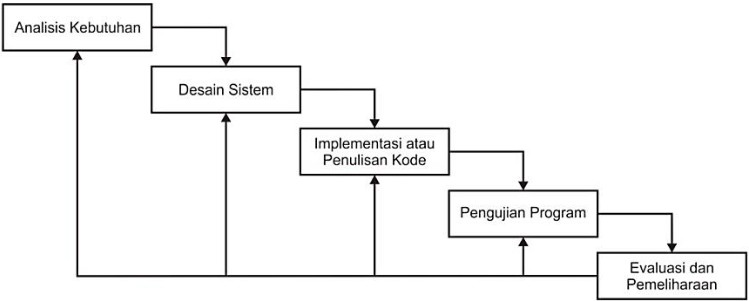
Penelitian yang dilakukan oleh Kaligis & Fatri (2020) mengenai Pengembangan Tampilan Antarmuka Aplikasi Survei Berbasis Web Dengan Metode *User Centered Design*. Menggunakan metode *user centered design,* tahapan menggunakan beberapa iterasi untuk penyesuaian kebutuhan pengguna. Penelitian ini bertujuan melakukan Pengembangan *Web* Aplikasi Survei untuk membantu efektivitas dalam pengambilan survei.

| **Peneliti (Tahun)** | Bagaskoro dkk. (2023) | Handayani & Salam (2023) | Hudhoifah dkk. (2022) | Agung Izulhaq dkk.( 2023) | Kaligis & Fatri (2020) |
| --- | --- | --- | --- | --- | --- |
| **Metodologi** | *Waterfall* | *Waterfall* | *Waterfall* | *Waterfall* | *User Centered Design* |
| **Platform** | *Website* | *Website* | Aplikasi *Mobile* | *Website* | *Website* |
| **Teknik pengumpul-an data** | Metode wawancara dilakukan secara langsung di restoran Alawuri Kebuli. | Mengajukan pertanyaan terhadap karyawan BMT Al-Bayan terkait sistem yang sedang berjalan. | Dilakukan proses observasi dan wawancara yang dilakukan oleh pihak-pihak yang berkaitan dengan Pengelola kegiatan masjid. | Dilakukan analisis PIECE, analisis permasala-han dan kebutuhan. | Dilakukan wawancara dengan responden penelitian. |

## Landasan Teori

### Metode *Waterfall*

Menurut Fachri & Surbakti (2021) metode waterfall adalah sebuah metode pengembangan sistem dimana antar satu fase ke fase yang lain dilakukan secara berurutan. Dalam proses implementasi metode Waterfall ini, sebuah langkah akan diselesaikan terlebih dahulu dimulai dari tahapan yang pertama sebelum melanjutkan ke tahapan yang berikutnya. Dalam pengembangannya metode *waterfall* terdiri dari 5 tahapan sebagai berikut:



Gambar 2.1 Tahapan Metode Waterfall (Hermansyah dkk., 2023)

Beberapa tahapan dari metode *waterfall* yaitu:

1. Analisis Kebutuhan

Tahap ini bertujuan untuk mengumpulkan data kebutuhan dari pengguna terkait sistem yang akan dibangun. Pengembang akan melakukan wawancara dengan pengguna serta membuat dokumen yang harus dipenuhi.

1. Desain Sistem

Setelah kebutuhan dipahami, maka pengembang akan merancang desain dari sistem yang akan dibangun. Perancangan desain yang akan dibangun antara lain desain arsitektur sistem, desain struktur data dan desain antar muka.

1. Implementasi atau Penulisan Kode

Setelah desain dibangun dan disepakati, maka tahap selanjutnya adalah implementasi. Hasil dari tahap ini adalah sebuah sistem yang disesuaikan dengan desain yang telah disepakati.

1. Pengujian Program

Pengujian fokus pada sistem dari segi fungsional serta memastikan bahwa sistem yang telah diuji berjalan baik dan sudah memenuhi kebutuhan pengguna.

1. Evaluasi dan Pemeliharaan

Pada tahap ini sistem sudah dapat digunakan oleh pengguna. Sistem akan dilakukan pemeliharaan serta peningkatan sistem mungkin terjadi pada tahap ini.

### Sistem Informasi

Menurut Rasefta & Esabella (2020) sistem informasi adalah cara-cara mengorganisasi untuk mengumpulkan, memasukkan dan mengolah serta menyimpan data untuk menyimpan dan melaporkan informasi sehingga sebuah organisasi dapat mencapai tujuan yang ingin dicapai. Sementara menurut Maydianto & Ridho (2021) sistem informasi adalah kumpulan beberapa komponen yang digunakan untuk mengelola data agar data yang dikelola dapat menjadi informasi berguna demi mencapai tujuan organisasi.

Dari pengertian diatas, dapat disimpulkan bahwa sistem informasi adalah sebuah sekumpulan komponen yang dikumpulkan dengan tujuan mengolah data sehingga nantinya data tersebut dapat menjadi sebuah informasi yang dapat digunakan suatu organisasi. Dengan adanya sistem informasi, data yang sebelumnya masih bersifat mentah dapat diolah menjadi suatu informasi ataupun pengetahuan yang dapat berguna untuk mencapai tujuan bersama.

### *Work Order*

Menurut Subyantoro (2009), pekerjaan merupakan sifat dari tugas yang dapat diartikan juga sebagai tanggung jawab atau tugas yang diperoleh dari pekerjaan itu sendiri. Pekerjaan biasa berupa aktivitas yang dilakukan oleh manusia dengan baik dan benar, dan dalam pengerjaannya memiliki tujuan tertentu. Sementara permintaan menurut Fitri dkk. (2022) merupakan bermacam tipe serta jumlah yang dapat berupa benda maupun jasa yang diinginkan seseorang pada periode tertentu.

Dari pengertian yang telah dijabarkan dapat disimpulkan bahwa permintaan akan sebuah pekerjaan atau *work order* adalah permintaan untuk melakukan atau menyelesaikan tugas atau aktivitas tertentu sesuai dengan tujuan yang diinginkan. Permintaan semacam ini bisa datang dari berbagai sumber, seperti individu, perusahaan, atau organisasi yang membutuhkan bantuan dalam menyelesaikan tugas tertentu.

### *Website*

Menurut Ramadha (2022) *website* merupakan sekumpulan halaman yang berhubungan serta pada umumnya berada pada *server* yang berisikan informasi oleh kelompok, orgnanisasi maupun perorangan. Biasanya untuk dapat mengakses sebuah *website* diperlukan jaringan baik jaringan internet ataupun jaringan lokal menggunakan *URL* sebagai alamat internet. Sementara menurut Suryandaru & Setyaningtyas (2021) *website* adalah perangkat lunak yang berisikan dokumen seperti teks atau gambar yang saling berhubungan didalam sebuah internet serta dibuat dalam bentuk *HTML* dan digunakan dengan bantuan navigasi untuk mengarahkan dari satu halaman ke halaman lainnya.

Dari pengertian diatas, *website* adalah sebuah perangkat atau media yang berisikan dokumen yang saling terhubung pada beberapa halaman *web* untuk memberikan informasi berupa gambar, teks, suara, dan *file* lainnya. Sebuah *website* dapat dijalankan dimana saja dan oleh siapa saja di segala waktu dengan ketentuan bahwa sebuah *website* membutuhkan adanya jaringan internet untuk dapat diakses. *Website* sangat berguna pada saat ini dimana segala informasi bergerak secara digital sehingga orang-orang memerlukan akses informasi yang dapat diakses setiap saat.

### *Hypertext PreProcessor* PHP

PHP merupakan bahasa pemrograman yang umumnya digunakan dalam pembuatan sebuah situs web. PHP pertama kali diciptakan oleh Rasmus Ledorf dengan wujud sekumpulan *script* untuk mengolah data form dari web. Awalnya, PHP digunakan untuk menghitung jumlah pengunjung di sebuah *website* (Sudrajat & Ahirudin, 2022).

PHP adalah bahasa script yang dijalankan pada sisi server SSS (*ServerSide Scripting*). PHP adalah *opensource software,* bebas untuk diunduh dan digunakan. File PHP dapat berisi teks, script dan tag HTML.

### *Framework Laravel*

*Framework* dapat diartikan sebagai kerangka kerja yang berisikan *script* seperti kelas dan fungsi dengan tujuan membantu *developer*/*programmer* untuk membangun aplikasi. Kerangka *framework* dapat digunakan ulang sehingga *programmer* tidak perlu membuat *script* untuk tugas atau keperluan yang sama. *Framework* berisi kumpulan fungsi yang nantinya fungsi tersebut bisa disimpan lalu digunakan kembali dengan cara memanggil kembali fungsi tersebut dengan kegunaan-nya masing masing (Yudhanto & Prasetyo, 2019).

Salah satu framework yang umum digunakan untuk membangun sebuah aplikasi adalah *Laravel.* Laravel adalah framework PHP bersifat *open source* menggunakan desain program MVC (*Model-View-Controller*) yang digunakan untuk membangun *website. Framework Laravel* menawarkan tools untuk pengelolaan, notifikasi, serta fitur lain yang dapat mempercepat serta mengoptimalkan kinerja aplikasi *web* (Thyo Priandika dkk. 2023).

### *Database*

Menurut Yustika (2023) *database* atau basis data adalah informasi yang dapat mengintegrasikan data serta menawarkan beberapa informasi yang saling berhubungan satu sama lain dengan berbagai tujuan organisasi. Sementara menurut Syahputri dkk. (2023) *database* adalah sebuah struktur koleksi dari data yang memiliki keterkaitan, disimpan dalam sebuah media penyimpanan komputer serta dapat diakses untuk dikelola menggunakan *software* khusus. Sistem *database* akan dikelola oleh sistem manajemen *database* (DBMS).

Menurut pengertian diatas, database atau basis data informasi merupakan kumpulan informasi yang berisikan data mengenai suatu keterkaitan dengan tujuan membantu organisasi dalam menyusun data. Data tersebut diolah oleh sistem manajemen *database* sehingga data ataupun informasi dapat disimpan dalam penyimpanan komputer. Pemilik *database* tentu saja dapat melihat serta mengelola data menggunakan perangkat lunak khusus agar datanya dapat berguna bagi organisasi.

#### *MySQL*

Menurut Ramadha (2022) MySQL adalah program *database server* yang dapat menerima serta mengirim data oleh multi user dengan sangat cepat menggunakan perintah standar SQL (*Structure Query Language*). MySQL banyak digunakan oleh perorangan maupun perusahaan dikarenakan MySQL adalah basis data yang dapat digunakan oleh semua kalangan karena tidak memungut biaya. MySQL berbasis *open source* sehingga terkenal di kalangan pengembang sistem seperti aplikasi, *website*, dan sistem lainnya

Pengertian lain mengenai MySQL menurut Sudrajat & Ahirudin (2022) merupakan sebuah *software* manajemen basis data SQL yang multiuser dan multithread dengan implementasi dari sistem manajemen basis data relasional (RDBMS). Penggunaan MySQL dapat dilakukan secara bebas dibawah lisensi GPL (*General Public License*). Namun MySQL tidak dapat dijadikan produk turunan berisfat komersial.

### *Black Box Testing*

Metode *Black Box Testing* merupakan pengujian untuk menunjukkan kesalahan pada sistem aplikasi seperti kesalahan pada fungsi system aplikasi, serta menu aplikasi yang hilang. *Black Box testing* merupakan metode uji fungsionalitas sistem aplikasi. Dalam melakukan pengujian menggunakan masukan data acak dengan tujuan untuk mendapatkan hasil yang pasti. Dikatakan pasti artinya bila salah, maka ditolak oleh sistem informasi atau data input tersebut tidak dapat disimpan dalam *database*, sedangkan bila data input benar maka dapat diterima/masuk di database sistem informasi. Metode pengujian *Black Box Testing* umum digunakan karena mudah digunakan yaitu tidak mengharuskan untuk menguasai bahasa pemrograman tertentu (Uminingsih dkk. 2022).

### *User Acceptance Testing (UAT)*

*User Acceptance Testing* merupakan pengujian yang dilakukan oleh enduser yang langsung berinteraksi dengan sistem dan dilakukan verifikasi apakah fungsi yang ada telah berjalan sesuai dengan kebutuhan/fungsinya.

*User Acceptance Testing* merupakan pengujian yang dilakukan oleh pengguna sistem. Hasil dari pengujian dapat dijadikan bukti bahwa sistem dapat membantu para pengguna. *User Acceptance Testing* dilakukan pada pengembangan perangkat lunak yang bertujuan untuk memastikan sistem memenuhi kebutuhan sebenarnya dari pengguna, bukan hanya sekedar memenuhi spesifikasi sistem (Wahyudi dkk., 2023).

### *Usabilty Testing*

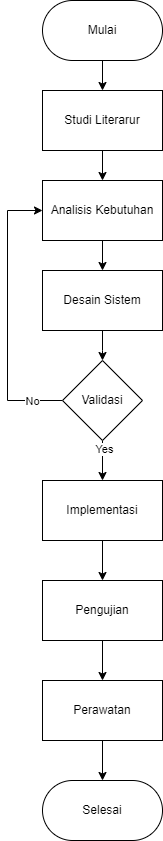
Menurut Rahadi (2014) *Usability* berasal dari kata *Usable* yang berarti dapat digunakan dengan baik. Sesuatu dapat dikatakan berguna dengan baik apabila kegagalan dalam penggunaanya dapat diminimalkan serta memberi manfaat yang dapat dirasakan oleh pengguna. Berdasarkan definisi tersebut *usability* diukur berdasarkan komponen:

1. Kemudahan (*learnability*) didefinisikan seberapa cepat pengguna mahir dalam menggunakan sistem serta kemudahan dalam penggunaan menjalankan suatu fungsi serta apa yang pengguna inginkan dapat meraka dapatkan.
2. Efisiensi (*efficiency*) didefenisikan sebagai sumber daya yang dikeluarkan guna mencapai ketepatan dan kelengkapan tujuan.
3. Mudah diingat (*memorability*) didefinisikan bagaimana kemapuan pengguna mempertahankan pengetahuannya setelah jangka waktu tertentu, kemampuan mengngat didapatkan dari peletakkan menu yang selalu tetap.
4. Kesalahan dan keamanan (*errors*) didefinisikan berapa banyak kesalahan-kesalahan apa saja yang dibuat pengguna, kesalahan yang dibuat pengguna mencangkup ketidaksesuaian apa yang pengguna pikirkan dengan apa yang sebenarnya disajikan oleh sistem.
5. Kepuasan (*satisfaction*) didefinisikan kebebasan dari ketidaknyamanan, dan sikap positif terhadap penggunaan produk atau ukuran subjektif sebagaimana pengguna merasa tentang penggunaan sistem

# III. PERANCANGAN



## Metodologi Penelitian



Gambar 3.1 Metodologi Penelitian

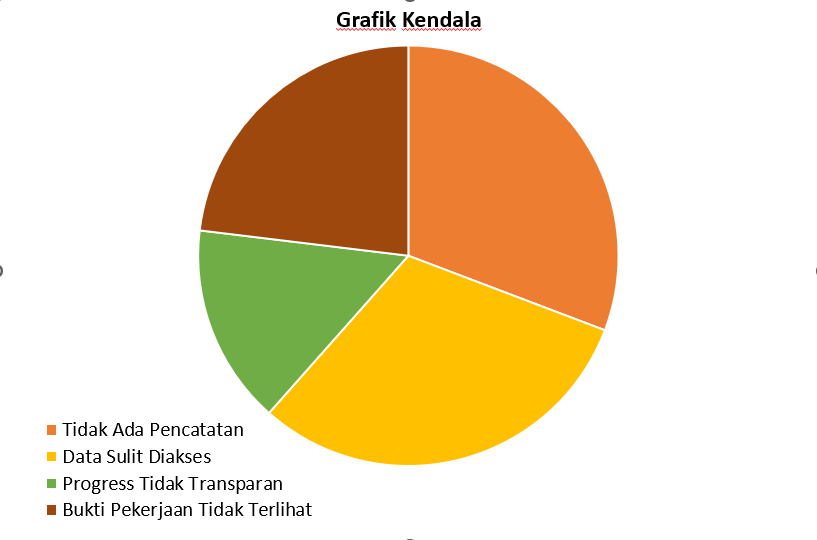
Pada Gambar 3.1 terdapat alur mengenai metode penelitian dimana pada awalnya diperlukan studi literatur mengenai sistem yang akan dibangun. Setelah itu dilakukan analisis kebutuhan berdasarkan wawancara yang telah dilakukan. Lalu rancangan sistem dibangun dan divalidasi apakah sesuai dengan pengguna. Selanjutnya desain yang telah dibangun diimplementasikan dan hasil akan diuji. Hasil yang telah diuji dapat diakses serta dilakukan perawatan.

## Analisis Kebutuhan Sistem

Pada tahap ini dilakukan proses identifikasi masalah, pengumpulan data serta dokumentasi terkait kebutuhan pengguna yang perlu ada di dalam sistem. Analisis kebutuhan sistem sangat diperlukan supaya tidak terjadi kesalahpahaman maupun kekurangan pada sistem sebelum tahapan dapat berlanjut dan tidak mengganggu tahapan selanjutnya.

### Hasil Wawancara

Wawancara merupakan tahapan krusial dalam melakukan analisis kebutuhan sistem oleh pengguna. Wawancara dilakukan dengan bagian sarana dan umum serta beberapa staf Politeknik Caltex Riau yang dapat dilihat pada Lampiran IV. Berdasarkan hasil wawancara didapatkan data terkait kendala serta kebutuhan mengenai sistem yang akan dibangun dapat dilihat pada Gambar 3.2.



Gambar 3.2 Grafik Kendala

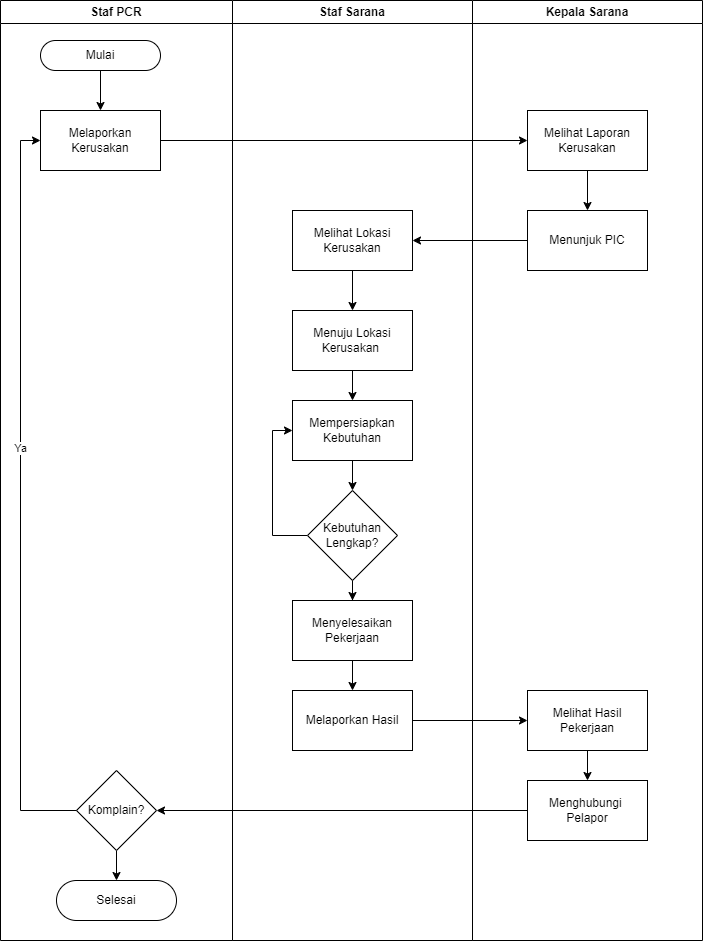
Gambar 3.2 merupakan grafik mengenai hasil wawancara yang telah dilakukan terkait dengan kendala yang dirasakan tanpa adanya sistem. Dapat dilihat bahwa kendala yang terjadi selama ini yaitu tidak ada pencatatan, data sulit diakses, progress tidak transparan dan bukti pekerjaan tidak terlihat.

### Proses Bisnis

#### Sebelum ada sistem (Lapor Kerusakan)

Sebelum adanya sistem, staf Politeknik Caltex Riau menghubungi pihak pihak sarana dan umum menggunakan media *whatsapp* dan *email* mengenai kerusakan yang dialami. Lalu kepala sarana dan umum akan mengerahkan salah satu staf sarana dan umum untuk melihat kerusakan yang terjadi. Lalu staf sarana tersebut menilai apakah kerusakan dapat diperbaiki saat itu juga dengan maksud staf sarana dan umum perlu melihat bahan dan alat yang akan digunakan apakah sanggup untuk melakukan perbaikan. Apabila kerusakan tersebut dapat diselesaikan saat itu maka staf sarana dan umum akan langsung memperbaiki kerusakan tersebut. Namun apabila kerusakan tersebut tidak dapat ditangani pada saat itu maka kerusakan tersebut harus di *pending* terlebih dahulu untuk mempersiapkan kebutuhan yang diperlukan agar kerusakan dapat diperbaiki.

Setelah kerusakan selesai ditangani, staf Politeknik Caltex Riau akan melihat kembali hasil pekerjaan yang telah dilakukan. Apabila menurut staf Politeknik Caltex Riau kerusakan tersebut sudah ditangani dengan baik maka pekerjaan tersebut sudah selesai. Namun apabila staf Politeknik Caltex Riau menilai kerusakan tersebut masih belum cukup, maka staf Politeknik Caltex Riau dapat mengajukan complain sehingga pihak sarana dan umum akan melihat kembali pekerjaan tersebut. Alur dari proses sebelum ada sistem dapat dilihat pada Gambar 3.3



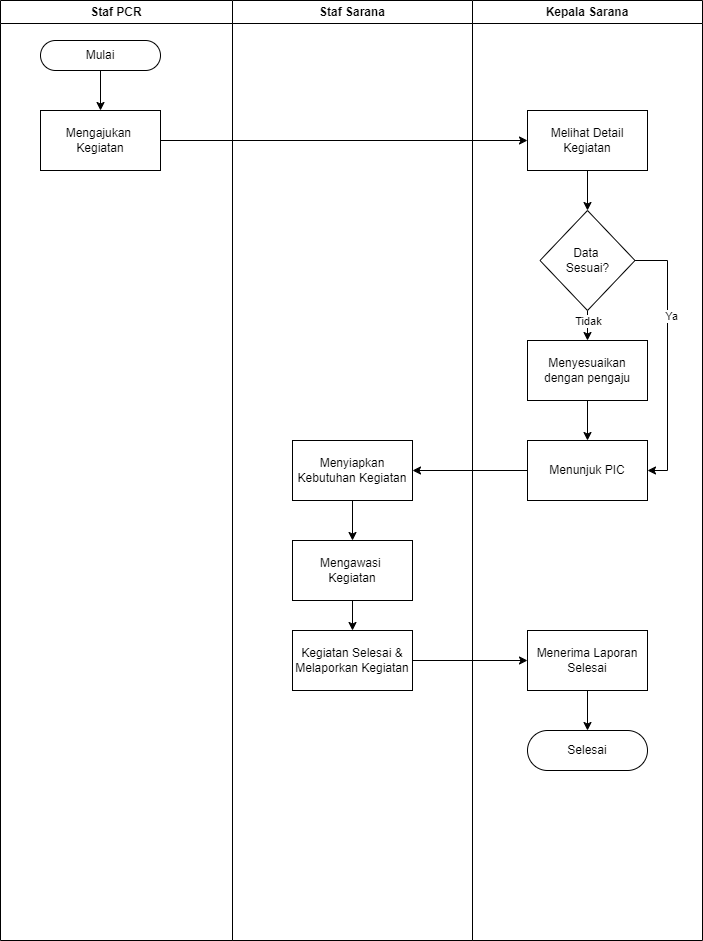
Gambar 3.3 Flowchart alur sebelum ada sistem (lapor kerusakan)

#### Sebelum ada sistem (Order Kegiatan)

Saat pihak staf Politeknik Caltex Riau ingin mengajukan kegiatan pada pihak sarana dan umum, staf Politeknik Caltex Riau akan mempersiapkan data mengenai kebutuhan seperti nama acara, ruangan, dan detail lainnya. Kemudian pihak staf Politeknik Caltex Riau akan menghubungi pihak sarana dan umum dan memberitahukan detail kegiatan yang ingin dilakukan.

Selanjutnya kepala sarana dan umum akan melihat detail kegiatan tersebut untuk memeriksa bahwa kegiatan tersebut dapat terlaksana. Apabila terdapat kebutuhan yang tidak bisa digunakan seperti ruangan ataupun kebutuhan lainnya maka pihak sarana dan umum akan menghubungi kembali staf Politeknik Caltex Riau untuk berdiskusi kembali agar kegiatan tetap dapat dilakukan.

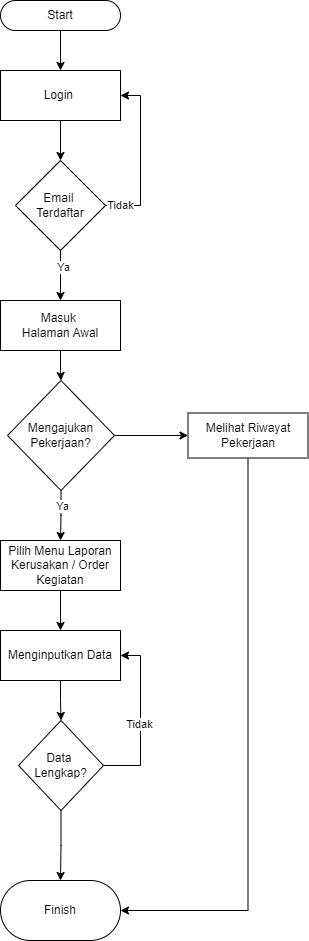
Setelah penyesuaian telah dilakukan maka kepala sarana dan umum akan menunjuk salah satu staf-nya untuk menjadi penanggung jawab pada kegiatan tersebut. Penanggung jawab tersebut akan mempersiapkan segala kebutuhan untuk kegiatan yang akan berlangsung serta bertanggung jawab untuk hari pelaksanaan kegiatan. Alur dari proses sebelum ada sistem dapat dilihat pada Gambar 3.4



Gambar 3.4 Flowchart alur sebelum ada sistem (order kegiatan)

#### Setelah ada sistem

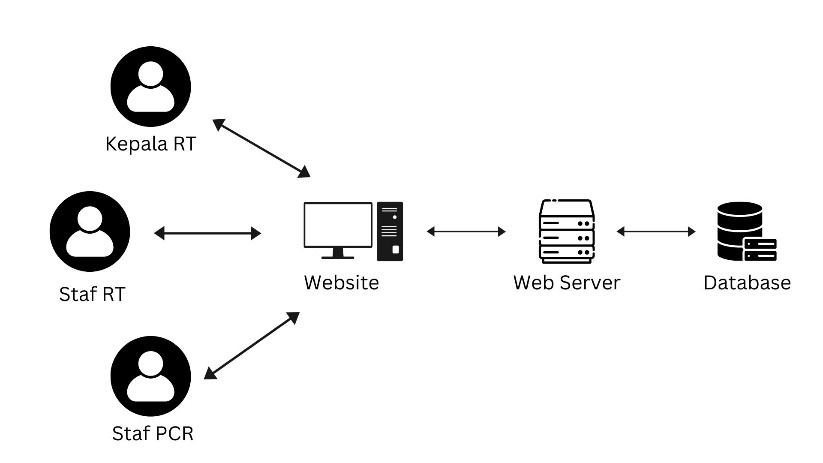
Pada sistem yang akan dibangun, segala aktifitas yang terjadi antara staf Politeknik Caltex Riau dengan pihak sarana dan umum mengenai proses pengajuan kegiatan dan pelaporan kerusakan akan terekam. Alur dari setiap user dapat dilihat pada *flowchart* Gambar 3.5 dan Lampiran IV.



Gambar 3.5 Flowchart Staf PCR

## Desain Sistem

### Arsitektur Sistem



Gambar 3.6 Arsitektur Sistem

Arsitektur sistem pada Gambar 3.6 menjelaskan tentang bagaimana arsitektur dari sistem informasi *work order*. Terdapat beberapa komponen yang membangun sistem informasi ini.

* + 1. Identifikasi Aktor

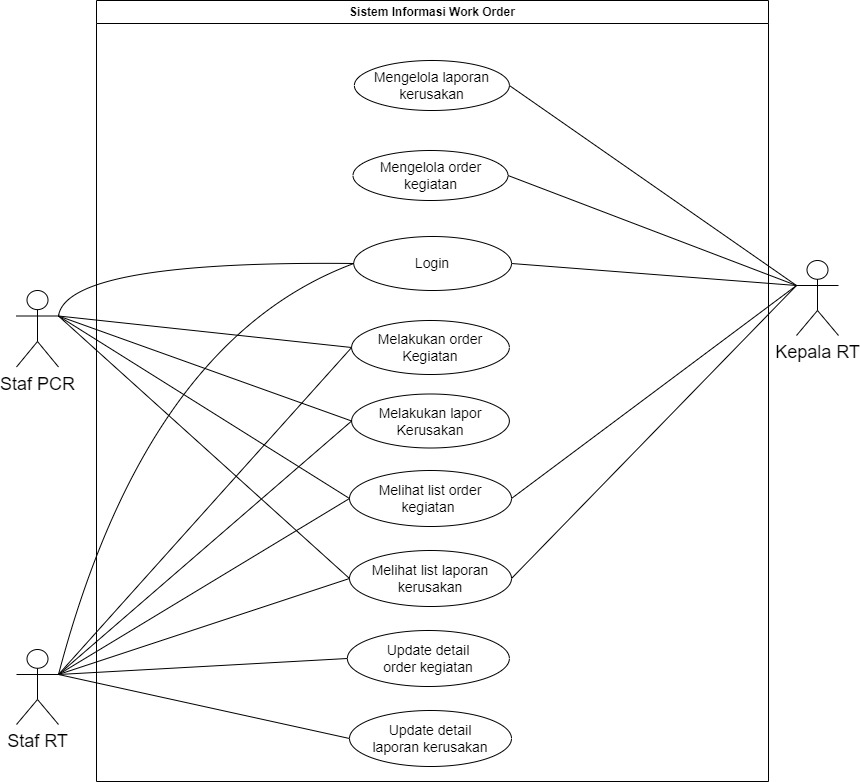
Identifikasi aktor adalah pengenalan pihak-pihak yang akan berinteraksi dengan sistem. Identifikasi aktor bertujuan untuk memahami siapa yang terlibat dalam sistem serta perannannya dalam sistem. Berikut merupakan identifikasi aktor pada sistem informasi *work order*:

Tabel 3.1 Tabel Identifikasi Aktor

| **No** | **Aktor** | **Deskripsi** |
| --- | --- | --- |
| 1 | Kepala Sarana | Aktor dengan *role* ini mempunyai wewenang untuk melihat pekerjaan yang telah diminta lalu memilih staf sarana dan umum untuk melakukan pekerjaan tersebut. Aktor dengan *role* ini juga dapat mengubah status dan mengatur pic. |
| 2 | StafSarana | Aktor dengan *role* ini mempunyai wewenang untuk membuat dokumentasi mengenai pekerjaan yang telah dilakukan. Aktor dengan *role* ini juga dapat mengajukan kegiatan atau melapor kerusakan. |
| 3 | Staf PCR | Aktor dengan *role* ini mempunyai wewenang untuk merequest pekerjaan kepada pihak Rumah Tangga PCR serta memberikan komentar kepada Rumah Tangga atas hasil pekerjaan nya. |

* + 1. *Use Case Diagram*

Perancangan *Use Case Diagram* berdasarkan kebutuhan sistem sesuai dengan aktornya dapat dilihat pada Gambar 3.4.



Gambar 3.7 Use Case Diagram

* + 1. *Use Case Scenario*

*Use Case Scenario* digunakan untuk memudahkan dalam menganalisis skenario yang ada. Berikut tahapan-tahapan *use case scenario* pada sistem informasi *work order.*

Tabel 3.2 Use Case Login

|  |  |
| --- | --- |
| **Use Case** | *Login* |
| **Aktor** | Kepala RT, Staf RT, Staf PCR |
| **Kondisi Awal** | Aktor belum melakukan *login* |
| **Kondisi Akhir** | Aktor berhasil melakukan *login* |
| **Skenario Normal 1 (Aktor *Login*)** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Menekan tombol *sign in* |  |
|  | 1. Menampilkan daftar *email* yang tersedia |
| 1. Memilih *email* |  |
|  | 1. Memverifikasi email |
|  | 1. Menampilkan halaman awal |
| **Skenario Alternatif** | |
| 1. Jika email yang dipilih tidak terdaftar maka akan muncul pesan *error* | |

Tabel 3.3 Use Case Scenario Mengelola Laporan Kerusakan

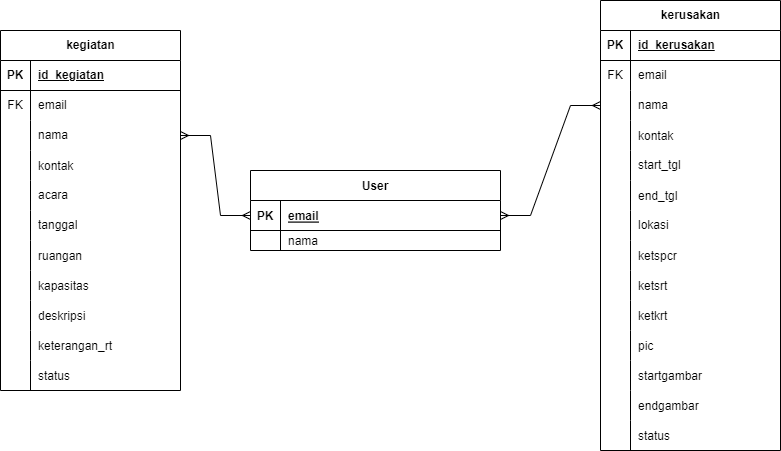
|  |  |
| --- | --- |
| **Use Case** | Mengelola Laporan Kerusakan |
| **Aktor** | Kepala RT |
| **Kondisi Awal** | Aktor sudah melakukan *login* |
| **Kondisi Akhir** | Aktor berhasil mengubah data laporan kerusakan |
| **Skenario Normal 1 (Menentukan PIC)** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Memilih menu ”Laporan Kerusakan” |  |
|  | 1. Menampilkan data laporan kerusakan |
| 1. Menekan tombol detail |  |
|  | 1. Menampilkan detail laporan kerusakan |
| 1. Menekan *dropdown* nama |  |
|  | 1. Menampilkan nama-nama staf rumah tangga |
| 1. Memilih sesuai dengan opsi *dropdown* yang tersedia |  |
| 1. Menekan tombol *save* |  |
|  | 1. Menyimpan peruabahan data laporan kerusakan |
|  | 1. Mengirimkan *email* kepada pic yang dipilih |
|  | 1. Menampilkan kembali halaman laporan kerusakan dengan data terbaru |
| **Skenario Normal 2 (Mengatur Status)** | |
| * + - 1. Menekan tombol detail |  |
|  | * + - 1. Menampilkan data laporan kerusakan |
| * + - 1. Menekan *dropdown* status |  |
|  | * + - 1. Menampilkan pilihan status |
| * + - 1. Memilih status selesai |  |
| * + - 1. Mengisi keterangan |  |
| * + - 1. Menekan tombol *save* |  |
|  | * + - 1. Menyimpan perubahan data laporan kerusakan |
|  | * + - 1. Mengirimkan email kepada staf PCR |
|  | * + - 1. Menampilkan kembali halaman laporan kerusakan dengan data terbaru |
| **Skenario Alternatif** | |
| Jika pekerjaan tidak dapat diselesaikan, maka aktor memilih status *pending* | |

Tabel 3.4 Use Case Scenario Mengelola Order Kegiatan

|  |  |
| --- | --- |
| **Use Case** | Mengelola *Order* Kegiatan |
| **Aktor** | Kepala RT |
| **Kondisi Awal** | Aktor sudah melakukan *login* |
| **Kondisi Akhir** | Aktor berhasil mengubah data order kegiatan |
| **Skenario Normal 1 (Menentukan PIC)** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Memilih menu ”*Order* Kegiatan” |  |
|  | 1. Menampilkan data *order* kegiatan |
| 1. Menekan tombol detail |  |
|  | 1. Menampilkan detail *order* kegiatan |
| 1. Menekan *dropdown* nama |  |
|  | 1. Menampilkan nama-nama staf rumah tangga |
| 1. Memilih sesuai dengan opsi *dropdown* yang tersedia |  |
| 1. Menekan tombol *save* |  |
|  | 1. Menyimpan peruabahan data order kegiatan |
|  | 1. Mengirimkan *email* kepada pic yang dipilih |
|  | 1. Menampilkan kembali halaman order kegiatandengan data terbaru |
| **Skenario Normal 2 (Mengatur Status)** | |
| 1. Menekan tombol detail |  |
|  | 1. Menampilkan data order kegiatan |
| 1. Menekan *dropdown* status |  |
|  | 1. Menampilkan pilihan status |
| 1. Memilih status selesai |  |
| 1. Mengisi keterangan |  |
| 1. Menekan tombol *save* |  |
|  | 1. Menyimpan perubahan data order kegiatan |
|  | 1. Mengirimkan *email* kepada staf PCR |
|  | 1. Menampilkan kembali halaman order kegiatan dengan data terbaru |
| **Skenario Alternatif** | |
| 1. Jika ruangan tidak tersedia, aktor mengubah data ruangan dengan ruangan yang tersedia | |

Untuk *use case scenario* selanjutnya dapat dilihat pada Lampiran I.

### *Entity Relationship Diagram* (ERD)



Gambar 3.8 Entity Relationship Diagram

Pada Gambar 3.8 terdapat 5 tabel yaitu tabel kegiatan, tabel kerusakan, tabel staf\_rt, tabel kepala\_rt dan tabel staf\_pcr. Tabel memuat data yang akan disimpan kedalam *database*.

### Perancangan Tabel

Perancangan tabel digunakan sebagai penjelasan mengenai detail *database* yang akan dibangun. Terdapat lima tabel yang terdiri dari tabel Staf RT, tabel Kepala RT, tabel Staf PCR, tabel Kerusakan dan tabel Kegiatan.

1. Tabel User

Nama tabel : user

Primary Key : email

Foreign Key : -

|  |  |  |
| --- | --- | --- |
| ***Field*** | ***Data Type*** | ***Description*** |
| email | *Varchar(100)* | *Primary Key* |
| nama | *Varchar(100)* | Nama lengkap dari *user*. |

1. Tabel Kegiatan

Nama tabel : kegiatan

Primary Key : id\_kegiatan

Foreign Key : -

|  |  |  |
| --- | --- | --- |
| ***Field*** | ***Data Type*** | ***Description*** |
| id\_kegiatan | *Integer(11)* | *Primary Key* |
| email | *Varchar(100)* | *Foreign Key* |
| nama | *Varchar(100)* | Nama dari pemohon |
| kontak | *Varchar(100)* | Kontak dari pemohon |
| acara | *Varchar(100)* | Nama acara yang akan dilangsungkan |
| tanggal | *Date* | Tanggal berlangsung acara |
| ruangan | *Integer(11)* | Ruangan yang diminta |
| kapasitas | *Integer(11)* | Kapasitas tamu acara |
| deskripsi | *Text* | Deskripsi lanjutan mengenai permohonan |
| keterangan\_rt | *Text* | Umpan balik dari rumah tangga jika terdapat kendala (opsional) |
| status | *Varchar(100)* | Status akhir dari permintaan kegiatan |

1. Tabel Kerusakan

Nama tabel : kerusakan

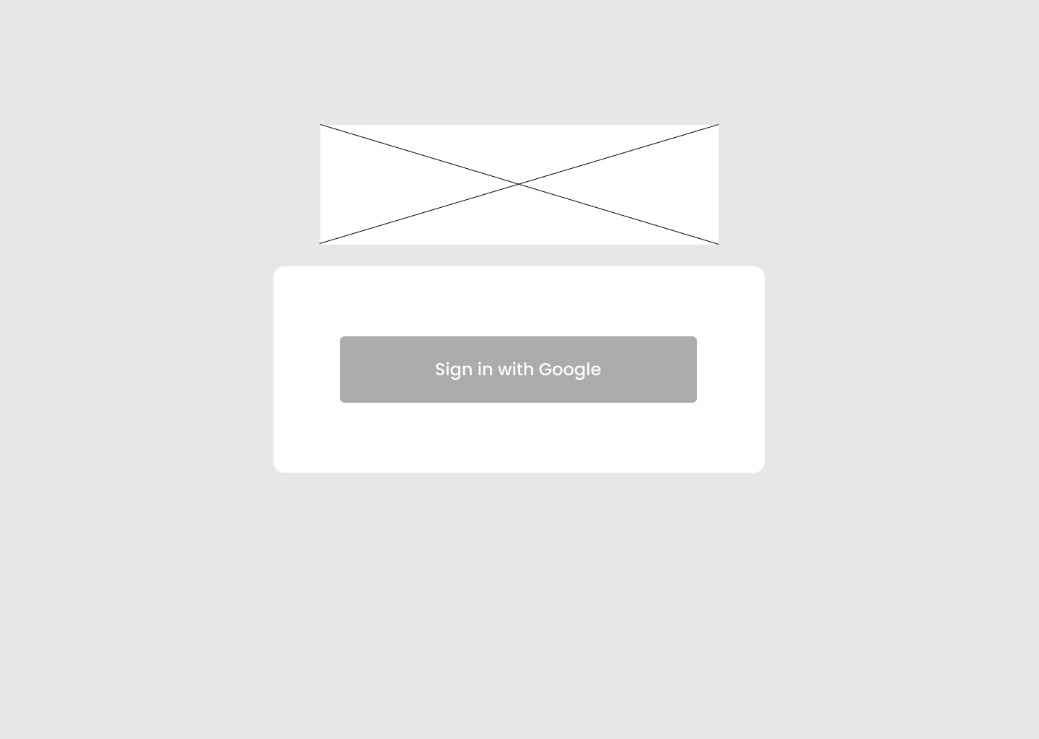
Primary Key : id\_kerusakan

Foreign Key : -

|  |  |  |
| --- | --- | --- |
| ***Field*** | ***Data Type*** | ***Description*** |
| id\_kerusakan | *Integer(11)* | *Primary Key* |
| email | *Varchar(100)* | *Foreign Key* |
| nama | *Varchar(100)* | Nama dari pemohon |
| kontak | *Varchar(100)* | Kontak dari pemohon |
| start\_tgl | *Date* | Tanggal mulai perbaikan |
| end\_tgl | *Date* | Tanggal selesai perbaikan |
| lokasi | *Varchar(100)* | Ruangan yang diminta |
| ketspcr | *Text* | Kapasitas tamu acara |
| ketsrt | *Text* | Deskripsi lanjutan mengenai permohonan |
| ketkrt | *Text* | Umpan balik dari rumah tangga jika terdapat kendala (opsional) |
| pic | *Varchar(100)* | PIC (staf RT) yang ditugaskan mengerjakan perbaikan |
| startgambar | *Varchar(100)* | Gambar kerusakan (oleh pemohon) |
| endgambar | *Varchar(100)* | Gambar laporan selesai (oleh staf RT) |
| status | *Varchar(100)* | Status akhir dari perbaikan kerusakan |

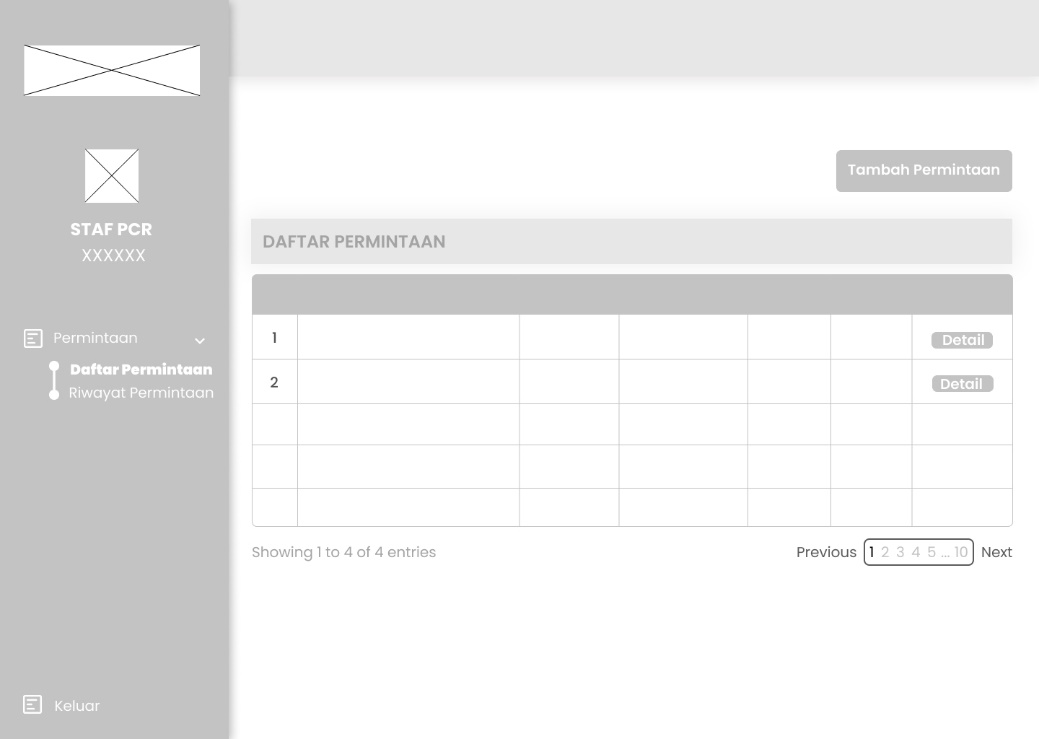
### *Wireframe*

*Wireframe* digunakan untuk pada tahap perancangan sistem. *Wireframe* digunakan untuk menggambarkan fitur, konten dan data yang ingin ditampilkan pada halaman *web*. Pada tahap ini dibuat 13 rancangan tampilan untuk ketiga *role* yang akan menggunakan sistem.



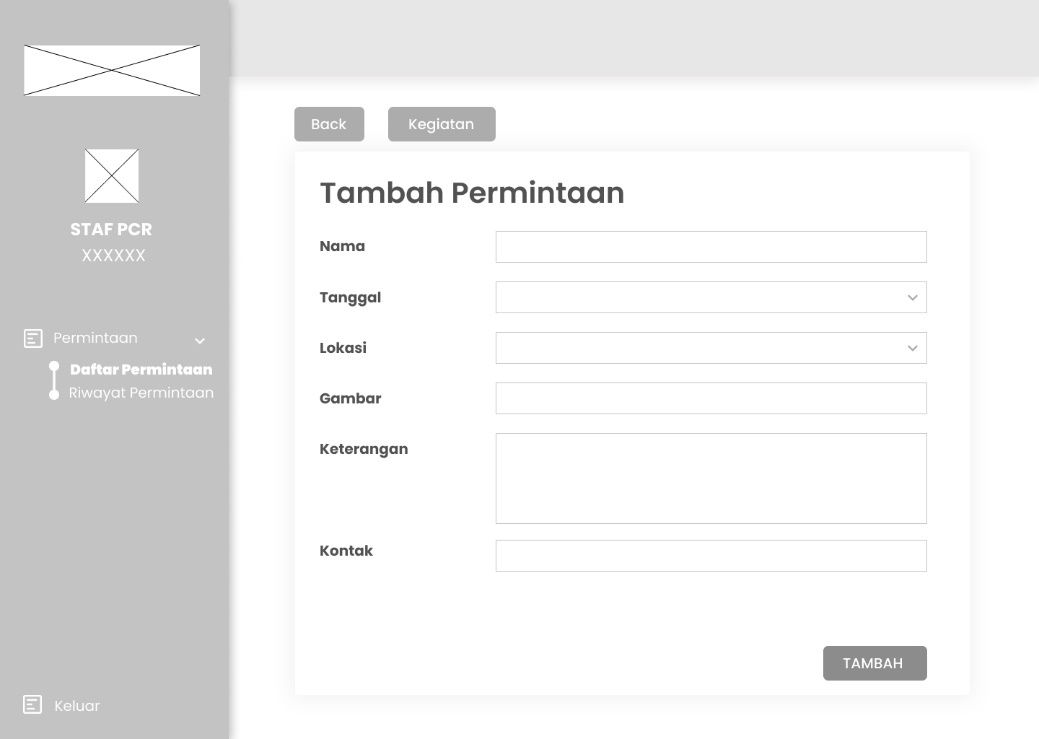
Gambar 3.9 Wireframe Login

Gambar 3.9 meruapakan rancangan tampilan login dimana terdapat logo dan *button* untuk *login* menggunakan email. *User* perlu melakukan login untuk dapat mengakses sistem.



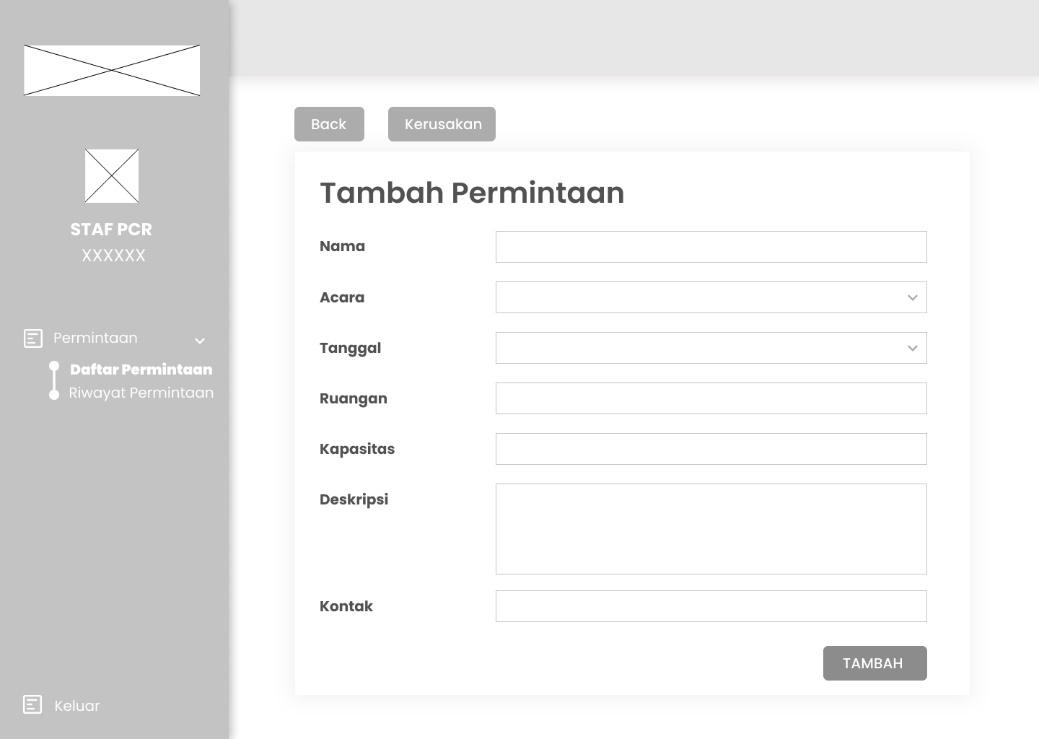
Gambar 3.10 Wireframe Daftar Permintaan Staf PCR

Gambar 3.10 merupakan rancangan tampilan untuk *user* staf pcr dimana *user* dapat melihat permintaan yang telah *user* ajukan. Staf pcr dapat melihat detail dari permintaan yang diajukan dan menambah permintaan kepada pihak rumah tangga pcr.



Gambar 3.11 Wireframe Lapor Kerusakan

Gambar 3.11 merupakan rancangan tampilan lapor kerusakan. *User* perlu mengisi beberapa data supaya pihak rumah tangga pcr mendapatkan informasi mengenai kebutuhan untuk mengerjakan laporan tersebut.



Gambar 3.12 Wireframe Order Kegiatan

Gambar 3.12 merupakan rancangan tampilan *order* kegiatan untuk kebutuhan pelaksanaan acara. *User* perlu mengisi beberapa data untuk kebutuhan acara yang akan berlangsung. Untuk *Wireframe* selanjutnya dapat dilihat pada Lampiran II.

## Metode Pengujian

### *Blackbox Testing*

*Blackbox testing* bertujuan untuk melakukan pengujian terhadap fungsional sistem. Sistem akan diuji berdasarkan keperluan dari pengguna sehingga mendapatkan hasil apakah sistem dapat betfungsi dengan baik. Pengujian dilakukan oleh Staf PCR, Staf RT dan Kepala RT.

Tabel 3.5 Tabel Blackbox Testing

| **No** | **Aktor** | **Fungsi** | **Hasil yang diharapkan** | **Hasil Pengujian** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| 1 | Staf PCR | *Login* | Menampilkan *form login* dan berhasil masuk ke halaman staf pcr | [ ] Berhasil  [ ] Gagal |  |
| 2 | Staf PCR | Melihat laporan kerusakan | Menampilkan tabel daftar laporan kerusakan | [ ] Berhasil  [ ] Gagal |  |
| 3 | Staf PCR | Melihat order kegiatan | Menampilkan tabel daftar *order* kegiatan | [ ] Berhasil  [ ] Gagal |  |
| 4 | Staf PCR | Melakukan lapor kerusakan | Menampilkan form dan melakukan tambah data laporan kerusakan | [ ] Berhasil  [ ] Gagal |  |
| 5 | Staf PCR | Melakukan *order* kegiatan | Menampilkan form dan melakukan tambah data *order* kegiatan | [ ] Berhasil  [ ] Gagal |  |
| 6 | Staf RT | *Login* | Menampilkan form login dan berhasil masuk ke halaman staf RT | [ ] Berhasil  [ ] Gagal |  |
| 7 | Staf RT | Melihat laporan kerusakan | Menampilkan tabel daftar laporan kerusakan | [ ] Berhasil  [ ] Gagal |  |
| 8 | Staf RT | Melihat *order* kegiatan | Menampilkan tabel daftar *order* kegiatan | [ ] Berhasil  [ ] Gagal |  |
| 9 | Staf RT | Melakukan lapor kerusakan | Menampilkan form dan melakukan tambah data laporan kerusakan | [ ] Berhasil  [ ] Gagal |  |
| 10 | Staf RT | Melakukan *order* kegiatan | Menampilkan form dan melakukan tambah data *order* kegiatan | [ ] Berhasil  [ ] Gagal |  |
| 11 | Staf RT | *Update* detail laporan kerusakan | Menampilkan halaman detail laporan kerusakan dan melakukan *edit* data | [ ] Berhasil  [ ] Gagal |  |
| 12 | Staf RT | *Update* detail *order* kegiatan | Menampilkan halaman detail *order* kegiatan dan melakukan *edit* data | [ ] Berhasil  [ ] Gagal |  |
| 13 | Kepala RT | *Login* | Menampilkan form login dan berhasil masuk ke halaman kepala rt | [ ] Berhasil  [ ] Gagal |  |
| 14 | Kepala RT | Melihat laporan kerusakan | Menampilkan tabel daftar laporan kerusakan | [ ] Berhasil  [ ] Gagal |  |
| 15 | Kepala RT | Melihat order kegiatan | Menampilkan tabel daftar *order* kegiatan | [ ] Berhasil  [ ] Gagal |  |
| 16 | Kepala RT | Mengelola laporan kerusakan | Menampilkan halaman detail laporan kerusakan dan melakukan *edit* data | [ ] Berhasil  [ ] Gagal |  |
| 17 | Kepala RT | Mengelola detail *order* kegiatan | Menampilkan halaman detail *order* kegiatan dan melakukan *edit* data | [ ] Berhasil  [ ] Gagal |  |

### *Usability Testing*

*Usability Testing* digunakan dengan tujuan menilai lima aspek yaitu *learnability, efficiency, memorability, errors* dan *helpfulness.* Menggunakan skala *likert,* *user* akan memilih antara Sangat Setuju, Setuju, Netral, Tidak Setuju, dan Sangan Tidak Setuju. Pengujian akan dilakukan oleh staf pcr.

Tabel 3.6 Tabel Usability Testing

| **No** | **Pernyataan** | **Pilihan Jawaban** | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **SS** | **S** | **N** | **TS** | **STS** |
| **LEARNABILITY** | | | | | | |
| 1 | Tampilan antarmuka Sistem Informasi Work Order mudah dipahami |  |  |  |  |  |
| 2 | Sistem Informasi Work Order mudah untuk dioperasikan |  |  |  |  |  |
| 3 | Alur dari Sistem Informasi Work Order mudah untuk dipahami |  |  |  |  |  |
| **EFFICIENCY** | | | | | | |
| 4 | Pengguna mudah melakukan perpindahan dari halaman yang satu ke halaman lainnya |  |  |  |  |  |
| 5 | Pengguna tidak merasa kesulitan menggunakan fitur yang ada |  |  |  |  |  |
| 6 | Pengguna mendapatkan hasil yang diharapkan dari aktivitas yang dilakukan |  |  |  |  |  |
| **MEMORABILITY** | | | | | | |
| 7 | Tata letak menu dalam sistem dapat dipahami dengan baik |  |  |  |  |  |
| 8 | Pemilihan nama menu yang ada mudah untuk dipahami |  |  |  |  |  |
| 9 | Fitur/fungsi yang ada di dalam sistem mudah diingat ketika diakses kembali |  |  |  |  |  |
| **ERRORS** | | | | | | |
| 10 | Ketika mengoperasikan sistem, tidak terdapat error yang mengganggu kinerja sistem |  |  |  |  |  |
| 11 | Terdapat peringatan jika sistem mengalami error yang berkaitan dengan alur kerja |  |  |  |  |  |
| **SATISFACTION** | | | | | | |
| 12 | Fitur dan fungsi yang ada sangat membantu pengguna |  |  |  |  |  |
| 13 | Informasi yang disajikan pada Sistem Informasi Work Order mudah dimengerti |  |  |  |  |  |
| 14 | Secara keseluruhan, saya cukup puas dengan kinerja sistem |  |  |  |  |  |

### 

### *User Acceptance Testing*

*User Acceptance Testing* digunakan dengan tujuan memperoleh kesimpulan mengenai fitur yang ada. Pengujian akan dilakukan oleh Staf RT dan Kepala RT dengan maksud menilai apakah fitur telah memenuhi kriteria dari Rumah Tangga. Terdapat enam fitur yang akan dilakukan pengujian.

Tabel 3.7 Pengujian login

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Deskripsi Pengujian** | **Prosedur Pengujian** | **Masukan Pengujian** | **Hasil Yang Diharapkan** | **Hasil Yang Diperoleh** | **Kesimpulan** |
| Menguji apakah user dapat melakukan login menggunakan *email* | Memilih *email* yang akan digunakan | *Email user* | Sistem menampilkan halaman daftar permintaan |  |  |

Tabel 3.8 Tambah laporan kerusakan staf Rumah Tangga

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Deskripsi Pengujian** | **Prosedur Pengujian** | **Masukan Pengujian** | **Hasil Yang Diharapkan** | **Hasil Yang Diperoleh** | **Kesimpulan** |
| Menguji apakah *user* dapat menginputkan laporan kerusakan | *User* masuk kedalam halaman tambah laporan kerusakan kemudian memasukkan data laporan kerusakan | Nama, Tanggal, Lokasi, Gambar, Keterangan dan Kontak | Sistem dapat menyimpan data kedalam *database* dan menampilkan kembali halaman daftar laporan kerusakan seRumah Tanggaa menerima notifikasi *email* |  |  |

Tabel 3.9 Tambah order kegiatan staf Rumah Tangga

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Deskripsi Pengujian** | **Prosedur Pengujian** | **Masukan Pengujian** | **Hasil Yang Diharapkan** | **Hasil Yang Diperoleh** | **Kesimpulan** |
| Menguji apakah *user* dapat menginputkan *order* kegiatan | *User* masuk kedalam halaman tambah laporan kerusakan kemudian memasukkan data *order* kegiatan | Nama, Acara, Tanggal, Ruangan, Kapasitas, Deskripsi, dan Kontak | Sistem dapat menyimpan data kedalam *database* dan menampilkan kembali halaman daftar *order* kegiatan serta menerima notifikasi *email* |  |  |

Tabel 3.10 Edit data laporan kerusakan kepala Rumah Tangga

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Deskripsi Pengujian** | **Prosedur Pengujian** | **Masukan Pengujian** | **Hasil Yang Diharapkan** | **Hasil Yang Diperoleh** | **Kesimpulan** |
| Menguji apakah kepala rt dapat mengubah data laporan kerusakan | Kepala rt masuk kedalam halaman detail laporan kerusakan lalu mengubah data | Status, PIC dan Keterangan | Sistem dapat menyimpan data kedalam *database* serta status dan PIC dapat berubah pada halaman laporan kerusakan |  |  |

Tabel 3.11 Edit data order kegiatan kepala Rumah Tangga

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Deskripsi Pengujian** | **Prosedur Pengujian** | **Masukan Pengujian** | **Hasil Yang Diharapkan** | **Hasil Yang Diperoleh** | **Kesimpulan** |
| Menguji apakah kepala rt dapat mengubah data *order* kegiatan | Kepala rt masuk kedalam halaman detail *order* kegiatan lalu mengubah data | Status, PIC, Ruangan, dan Keterangan | Sistem dapat menyimpan data kedalam *database* serta status dan PIC dapat berubah pada halaman *order* kegiatan |  |  |

Tabel 3.12 Edit laporan kerusakan staf Rumah Tangga

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Deskripsi Pengujian** | **Prosedur Pengujian** | **Masukan Pengujian** | **Hasil Yang Diharapkan** | **Hasil Yang Diperoleh** | **Kesimpulan** |
| Menguji apakah staf rt dapat mengubah data laporan kerusakan | Kepala rt masuk kedalam halaman detail laporan kerusakan lalu mengubah data | Tanggal, Gambar, Keterangan | Sistem dapat menyimpan data kedalam *database* serta data yang diinputkan dapat tersimpan kedalam database |  |  |

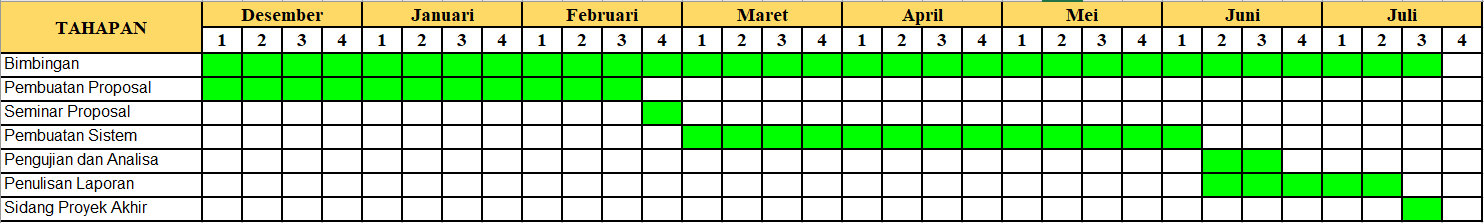
# IV. JADWAL DAN ANGGARAN BIAYA



## Jadwal

Jadwal pengerjaan proyek akhir bertujuan agar penggunaan waktu lebih efektif dan sesuai dengan metodologi yang digunakan. Adapun jadwal pelaksanaan dapat dilihat pada Tabel 4.1 berikut.

Tabel 4.1 Jadwal Pelaksanaan Proyek Akhir



## Perkiraan Biaya

Pada perkiraan biaya agar proyek akhir dapat berjalan dengan lancar dan sesuai dengan metodologi yang digunakan, dibutuhkan biaya untuk peralatan yang dibutuhkan dalam mengerjakan proyek akhir ini. Adapun peralatan serta biaya yang digunakan sebagai berikut.

Tabel 4.2 Perkiraan Biaya

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No** | **Aktivitas/**  **Komponen Biaya** | **Satuan** | **Jumlah** | **Harga Satuan** | **Total** | **Keterangan** |
| 1 | Laptop | Unit | 1 | 13.000.000 | 13.000.000 | Milik Pribadi |
| 2 | Biaya *Programmer* | Hari | 120 | 20.000 | 2.400.000 | Durasi 4 bulan |
| 3 | Langganan Internet | Bulan | 3 | 100.000 | 300.000 | Berlangga-nan |
| Total Biaya Keseluruhan | | | | | 15.700.000 | Total Biaya |

# DAFTAR PUSTAKA

Agung Izulhaq, Uce Indahyanti, & Ika Ratna Indra Astutik. (2023). Sistem Informasi Pemesanan Produk Percetakan Berbasis Web Menggunakan Metode Waterfall Agung Izulhaq \* , Uce Indahyanti, Ika Ratna Indra Astutik. *Kajian Ilmiah Informatika Dan Komputer*, *4*(1), 486–496. https://doi.org/10.30865/klik.v4i1.1146

Bagaskoro, H., Ardhitian, F., Waluya Baskara, A., & Budi Setiawan, A. (2023). Sistem Informasi Penyeleksian Calon Karyawan Menggunakan Metode Waterfall pada Alawuri Kebuli. *Jurnal Informasi Dan Teknologi*, *5*(2), 73–84. https://doi.org/10.37034/jidt.v5i2.337

Fachri, B., & Surbakti, R. W. (2021). Perancangan Sistem Dan Desain Undangan Digital Menggunakan Metode Waterfall Berbasis Website (Studi Kasus: Asco Jaya). *Journal of Science and Social Research*, *4*(3), 263. https://doi.org/10.54314/jssr.v4i3.692

Fitri, N., Kamilah, K., & Rahma, T. I. F. (2022). Analisis Faktor-Faktor Yang Mempengaruhi Permintaan Konsumen Depot Air Minum Semuril Marelan. *SIBATIK JOURNAL: Jurnal Ilmiah Bidang Sosial, Ekonomi, Budaya, Teknologi, Dan Pendidikan*, *1*(3), 151–162. https://doi.org/10.54443/sibatik.v1i3.21

Handayani, D., & Salam, M. (2023). Aplikasi Sistem Informasi Simpan Pinjam Koperasi Berbasis Website Menggunakan Metode Waterfall. *Media Online)*, *3*(5), 425–434. https://djournals.com/klik

Hermansyah, H., Wijaya, R. F., & Utomo, R. B. (2023). Metode Waterfall Dalam Rancang Bangun Sistem Informasi Manajemen Kegiatan Masjid Berbasis Web. *KLIK: Kajian Ilmiah Informatika Dan Komputer*, *3*(5), 563–571. https://djournals.com/klik/article/view/756

Hudhoifah, M. A., Tri, W., Fatonah, & Anissa Pramudyah Heryani. (2022). Sistem Informasi Pengingat Jadwal Kajian MenggunakanMetode Waterfall Berbasis Android pada Masjid NurulQalbi Duren Sawit. *Jurnal Pengabdian Kepada Masyarakat Nusantara (JPkMN)*, *3*(2), 1294–1302.

Maydianto, & Ridho, M. R. (2021). Rancang Bangun Sistem Informasi Point of Sale Dengan Framework Codeigniter Pada Cv Powershop. *Jurnal Comasie*, *02*, 50–59.

Rahadi, D. R. (2014). Pengukuran Usability Sistem Menggunakan Use Questionnaire Pada Aplikasi Android. *Jurnal Sistem Informasi (JSI)*, *6*(1), 661–671. http://ejournal.unsri.ac.id/index.php/jsi/index

Ramadha, K. N. (2022). *WEBSITE NOVEL KOMEDI WEBNOVEL MENGGUNAKAN PHP DAN*. *1*(4), 64–79.

Rasefta, R. S., & Esabella, S. (2020). Sistem Informasi Akademik Smk Negeri 3 Sumbawa Besar Berbasis Web. *Jurnal Informatika, Teknologi Dan Sains*, *2*(1), 50–58. https://doi.org/10.51401/jinteks.v2i1.558

Subyantoro, A. (2009). Karakteristik Individu, Karakteristik Pekerjaan, Karakteristik Organisasi dan Kepuasan Kerja Pengurus yang Dimediasi oleh Motivasi Kerja (Studi pada Pengurus KUD di Kabupaten Sleman). *Jurnal Manajemen Dan Kewirausahaan*, *11*(1), pp.11-19. http://puslit2.petra.ac.id/ejournal/index.php/man/article/view/17740

Sudrajat, R., & Ahirudin, R. (2022). Pembangunan Sistem Informasi Pondok Pesantren Sa ’ adatuddaroin Berbasis Web. *Jurnal Ilmu Komputer Universitas Nasional Pasim*, 1–13.

Suryandaru, N. A., & Setyaningtyas, E. W. (2021). Pengembangan Media Pembelajaran Berbasis Website Pada Muatan Pembelajaran Matematika Kelas IV. *Jurnal Basicedu*, *5*(6), 6040–6048. https://doi.org/10.31004/basicedu.v5i6.1803

Syahputri, K., Irwan, M., & Nasution, P. (2023). *Peran Database Dalam Sistem Informasi Manajemen*. *01*(02), 54–58.

Thyo Priandika, A., Korespondensi, P., Alfarisi, I. A., & Puspaningrum, A. S. (2023). Penerapan Framework Laravel Pada Sistem Pelayanan Kesehatan (Studi Kasus: Klinik Berkah Medical Center). *Jurnal Ilmiah Computer Science*, *2*(1), 1–9. https://ejurnal.snn-media.com/index.php/jics/article/view/11

Uminingsih, Nur Ichsanudin, M., Yusuf, M., & Suraya, S. (2022). Pengujian Fungsional Perangkat Lunak Sistem Informasi Perpustakaan Dengan Metode Black Box Testing Bagi Pemula. *STORAGE: Jurnal Ilmiah Teknik Dan Ilmu Komputer*, *1*(2), 1–8. https://doi.org/10.55123/storage.v1i2.270

Wahyudi, I., Fahrullah, Alameka, F., & Haerullah. (2023). Analisis Blackbox Testing Dan User Acceptance Testing Terhadap Sistem Informasi Solusimedsosku. *Jurnal Teknosains Kodepena |*, *04*(01), 1–9.

Yudhanto, Y., & Prasetyo, H. A. (2019). *Mudah Menguasai Framework Laravel*. PT Elex Media Komputindo, Jakarta. https://books.google.co.id/books?id=8tKdDwAAQBAJ&printsec=copyright&hl=id&source=gbs\_pub\_info\_r#v=onepage&q&f=false

Yustika, dkk. (2023). Peranan Sistem Database Di Dalam Sistem Informasi Manajemen Pada UINSU (Universitas Islam Negeri Sumatera Utara). *… : Jurnal Ekonomi Dan …*, *1*(5), 188–196. http://qjurnal.my.id/index.php/sur/article/view/428%0Ahttps://qjurnal.my.id/index.php/sur/article/download/428/322

# LAMPIRAN I

|  |  |
| --- | --- |
| **Use Case** | *Melakukan Order Kegiatan* |
| **Aktor** | Staf RT, Staf PCR |
| **Kondisi Awal** | Aktor sudah melakukan *login* |
| **Kondisi Akhir** | Aktor berhasil menambah *order* kegiatan |
| **Skenario Normal 1 (Menambah Order Kegiatan)** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Memilih menu ”*Order* Kegiatan” |  |
|  | 1. Menampilkan halaman order kegiatan |
| 1. Menekan tombol tambah |  |
|  | 1. Menampilkan halaman tambah kegiatan |
| 1. Mengisi form dengan data yang sesuai |  |
| 1. Menekan tombol *save* |  |
|  | 1. Memeriksa kelengkapan form |
|  | 1. Menyimpan data terbaru |
|  | 1. Mengirimkan *email* ke kepala Sarana dan Umum |
|  | 1. Menampilkan kembali halaman order kegiatan |
| **Skenario Alternatif** | |
| 1. Jika terdapat form yang kosong maka muncul pesan error dan mengisi ulang form | |

|  |  |
| --- | --- |
| **Use Case** | MelakukanLaporan Kerusakan |
| **Aktor** | Staf RT, Staf PCR |
| **Kondisi Awal** | Aktor sudah melakukan *login* |
| **Kondisi Akhir** | Aktor berhasil menambah *order* kegiatan |
| **Skenario Normal 1 (Menambah Laporan Kerusakan)** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Memilih menu ”Lapor Kerusakan” |  |
|  | 1. Menampilkan halaman lapor kerusakan |
| 1. Menekan tombol tambah |  |
|  | 1. Menampilkan halaman lapor kerusakan |
| 1. Mengisi *form* dengan data yang sesuai |  |
| 1. Menekan tombol *save* |  |
|  | 1. Memeriksa kelengkapan *form* |
|  | 1. Menyimpan data terbaru |
|  | 1. Mengirimkan *email* ke kepala Sarana dan Umum |
|  | 1. Menampilkan kembali halaman laporan kerusakan |
| **Skenario Alternatif** | |
| 1. Jika terdapat form yang kosong maka muncul pesan error dan mengisi ulang form | |

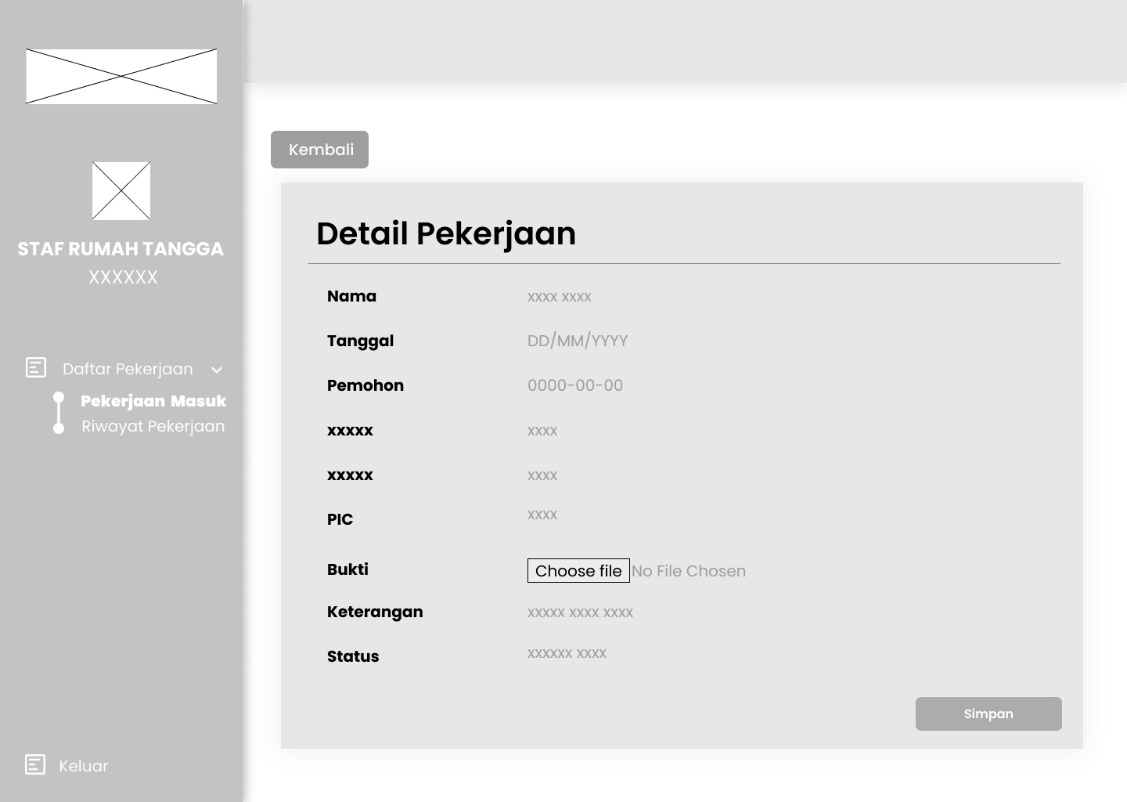
|  |  |
| --- | --- |
| **Use Case** | Melihat *list order* kegiatan |
| **Aktor** | Kepala RT, Staf RT, Staf PCR |
| **Kondisi Awal** | Aktor sudah melakukan *login* |
| **Kondisi Akhir** | Aktor berhasil mengakses data *order* kegiatan |
| **Skenario Normal 1 (Melihat *Order* Kegiatan)** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Memilih menu ”*Order* Kegiatan” |  |
|  | 1. Menampilkan halaman order kegiatan |
| **Skenario Alternatif** | |
| - | |

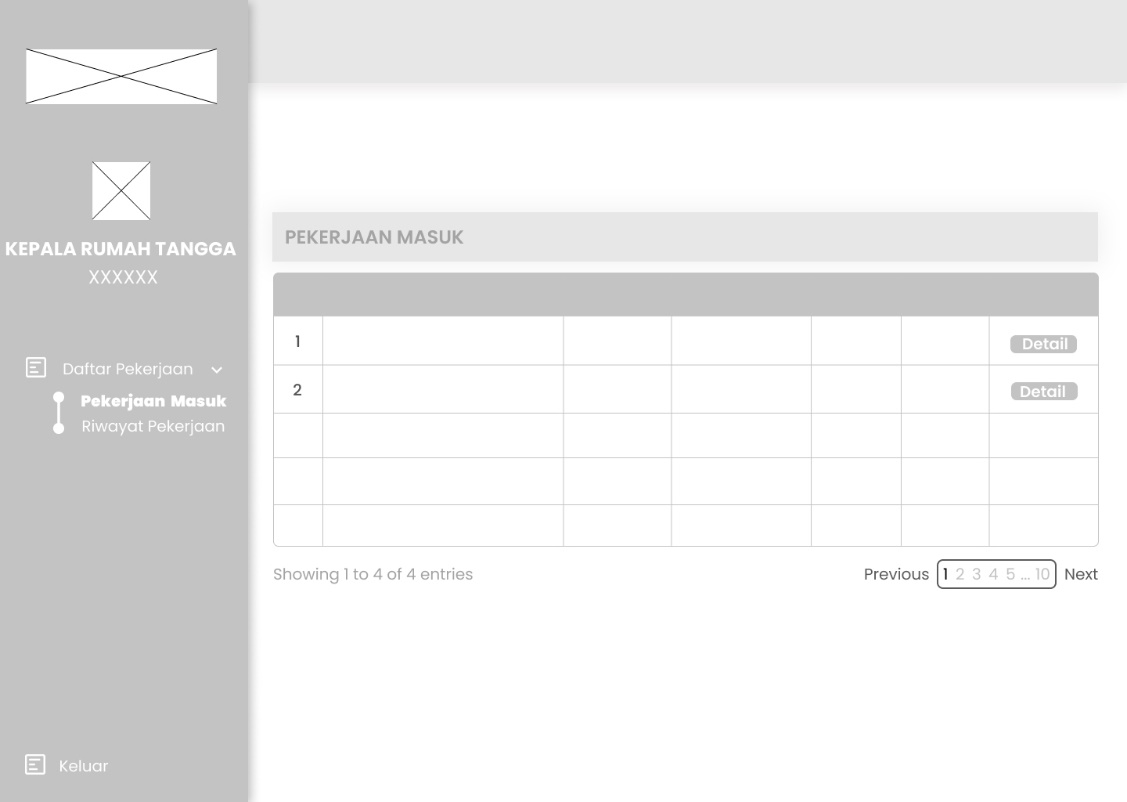
|  |  |
| --- | --- |
| **Use Case** | Melihat *list* laporan kerusakan |
| **Aktor** | Kepala RT, Staf RT, Staf PCR |
| **Kondisi Awal** | Aktor sudah melakukan *login* |
| **Kondisi Akhir** | Aktor berhasil mengakses data laporan kerusakan |
| **Skenario Normal 1 (Melihat *Order* Kegiatan)** | |

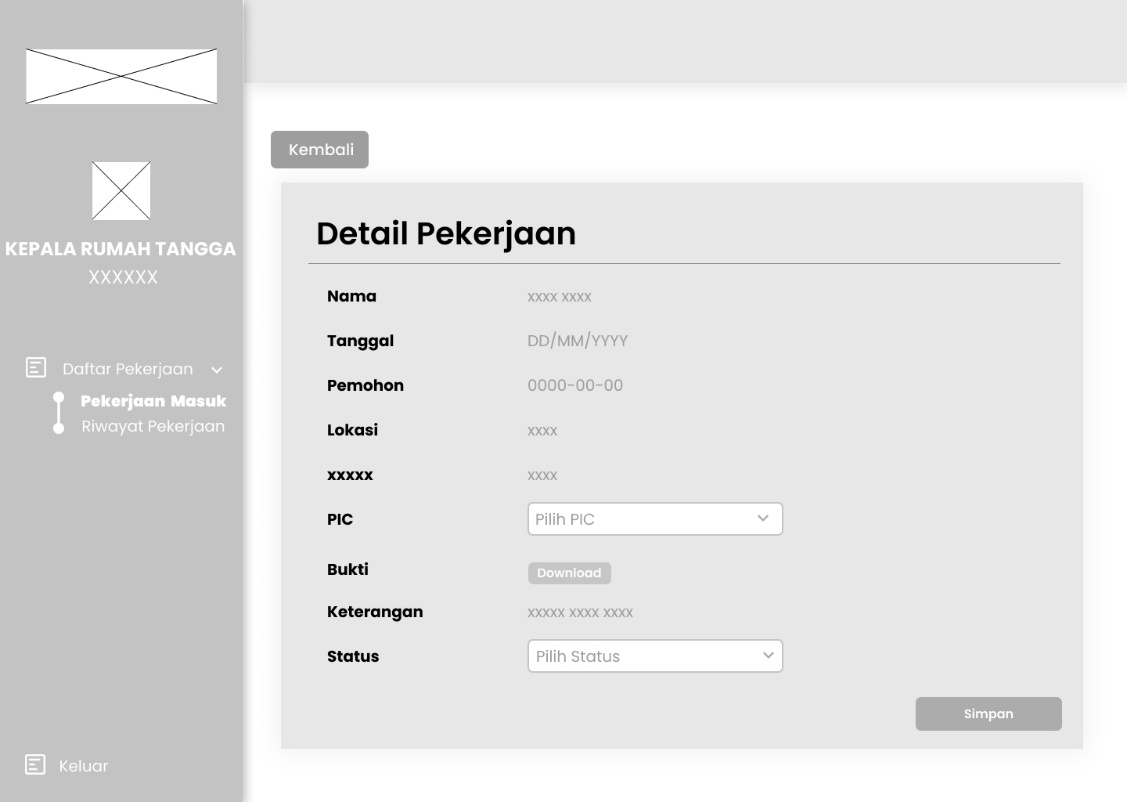
|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Memilih menu ”Lapor Kerusakan” |  |
|  | 1. Menampilkan halaman laporan kerusakan |
| **Skenario Alternatif** | |
| - | |

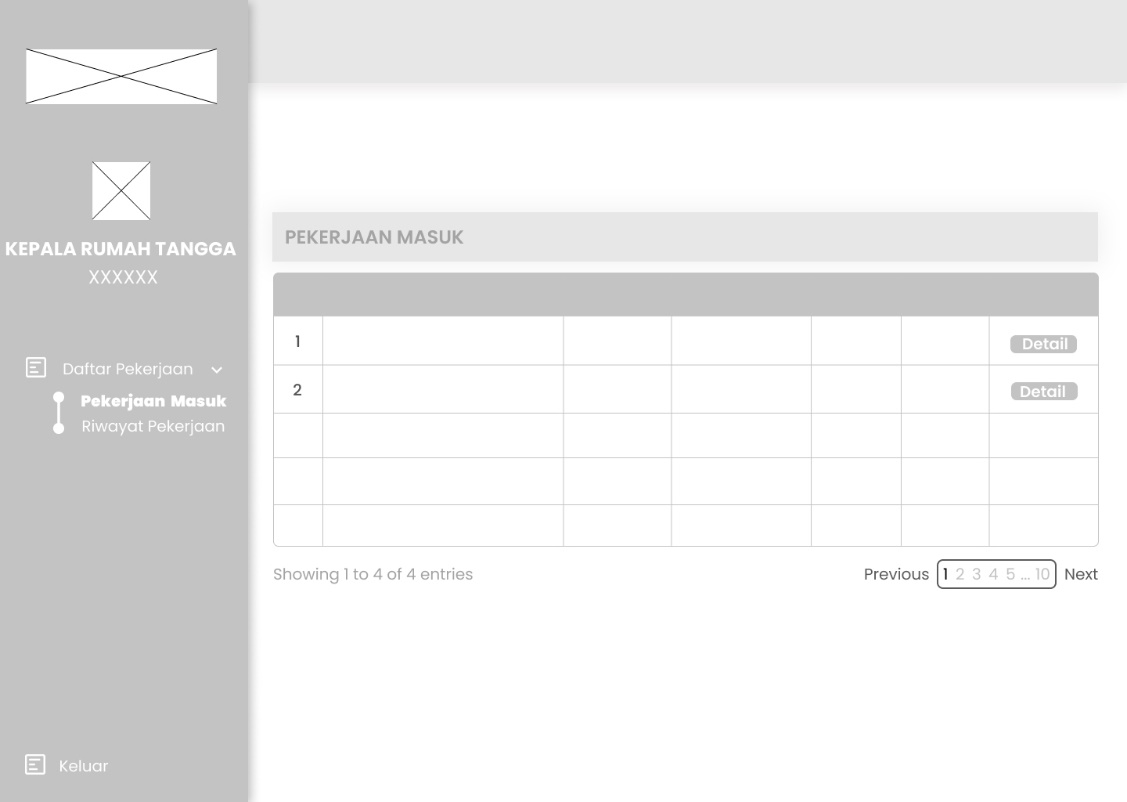
|  |  |
| --- | --- |
| **Use Case** | *Update* laporan kerusakan |
| **Aktor** | Staf RT |
| **Kondisi Awal** | Aktor sudah melakukan *login* |
| **Kondisi Akhir** | Aktor berhasil mengubah data laporan kerusakan |
| **Skenario Normal 1 (Menambah Order Kegiatan)** | |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Memilih menu ”Laporan Kerusakan” |  |
|  | 1. Menampilkan halaman laporan kerusakan |
| 1. Menekan tombol detail |  |
|  | 1. Menampilkan halaman detail laporan kerusakan |
| 1. Mengubah data yang ingin diubah |  |
| 1. Menekan tombol *save* |  |
|  | 1. Menyimpan data terbaru |
|  | 1. Menampilkan kembali halaman laporan kerusakan |
| **Skenario Alternatif** | |
| - | |

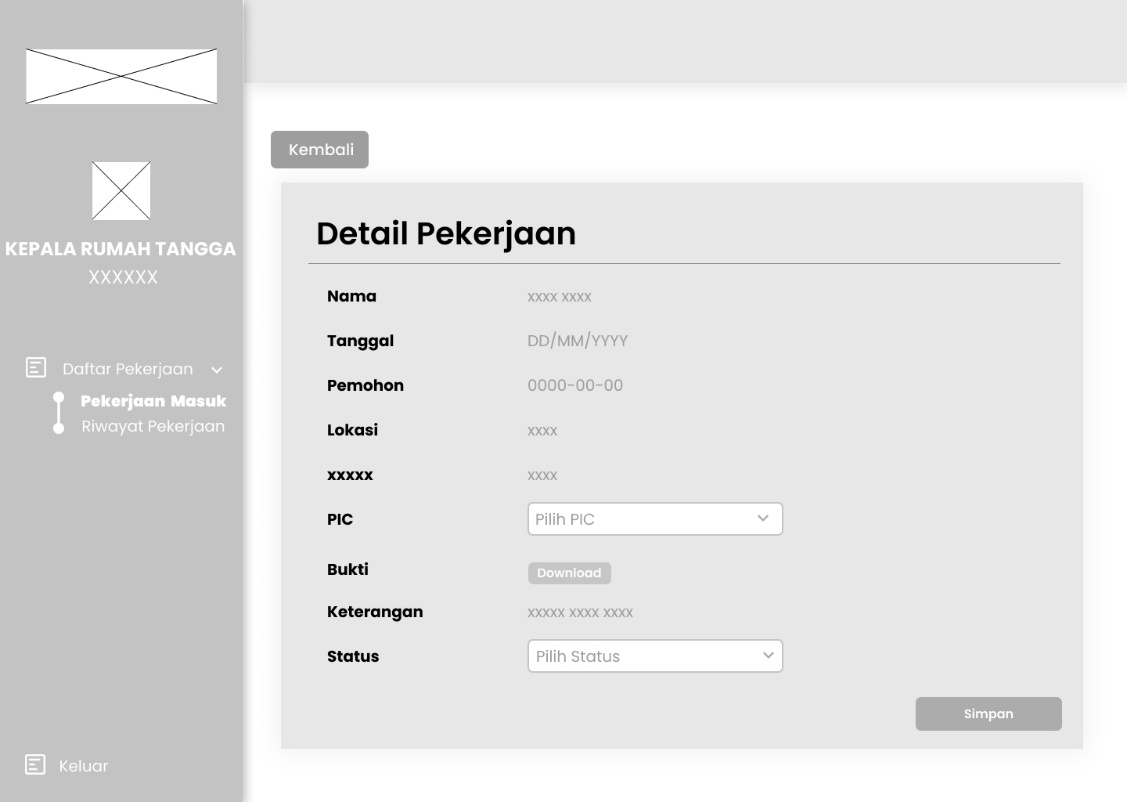
# LAMPIRAN II

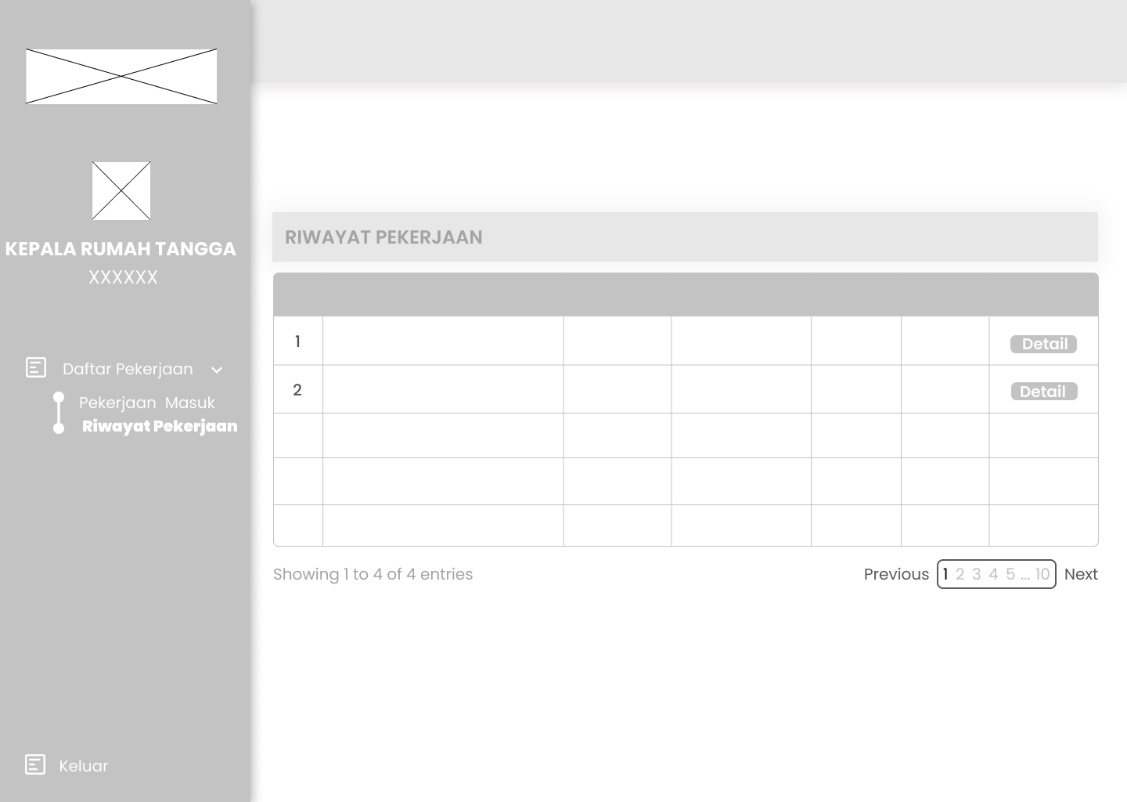




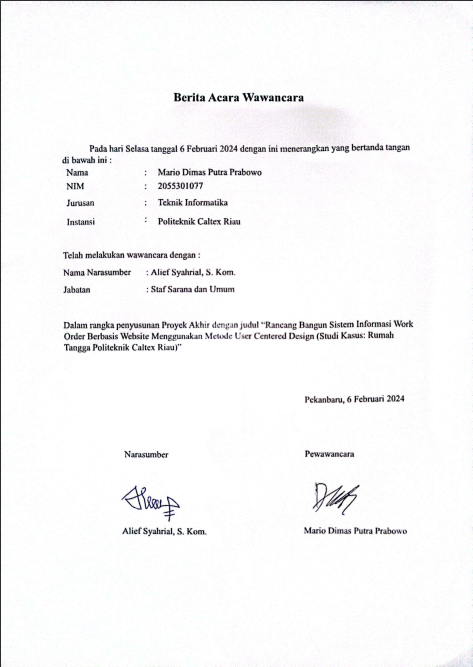


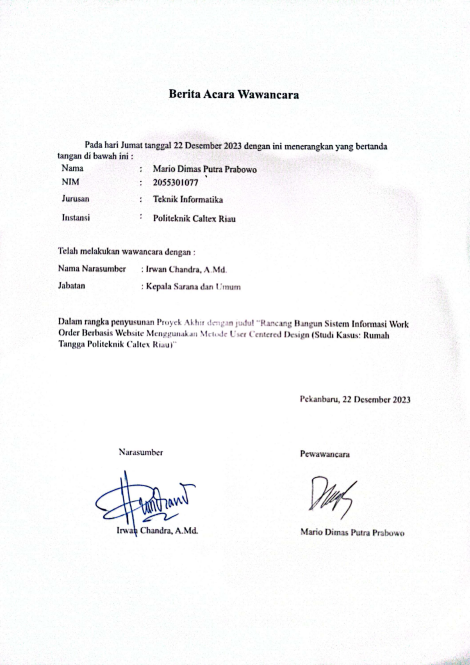






# LAMPIRAN III





# LAMPIRAN IV

