

**INSTITUT TEKNOLOGI DEL**

**Pengembangan Aplikasi Manajemen Rapat berbasis Web (Studi Kasus: D3 Teknologi Informasi)**

**DOKUMEN TUGAS AKHIR**

|  |  |
| --- | --- |
| 11317049 | FEDRICK SULAIMAN SIAGIAN |
| 11318048 | TERESHA JESIKA TAMPUBOLON |
| 11318052 | VERONIKA OKTAFIA MARPAUNG |

FAKULTAS INFORMATIKA DAN ELEKTRO  
PROGRAM STUDI DIPLOMA III TEKNOLOGI INFORMASI  
INSTITUT TEKNOLOGI DEL  
NOVERMBER, 2021



**INSTITUT TEKNOLOGI DEL**

PENGEMBANGAN APLIKASI MANAJEMEN BERBASIS WEB(STUDI KASUS : D3 TEKNOLOGI INFORMASI)

DOKUMENTASI TUGAS AKHIR

Diajukan sebagai salah satu syarat untuk memperoleh gelar Ahli

Madya Komputer

|  |  |
| --- | --- |
| 11317049 | FEDRICK SULAIMAN SIAGIAN |
| 11318048 | TERESHA JESIKA TAMPUBOLON |
| 11318052 | VERONIKA OKTAFIA MARPAUNG |

FAKULTAS INFORMATIKA DAN TEKNIK ELEKTRO  
PROGRAM STUDI DIPLOMA III TEKNOLOGI INFORMASI

**DAFTAR ISI**

[DAFTAR TABEL 1](#_Toc77284659)

[DAFTAR GAMBAR 1](#_Toc77284660)

[BAB I PENDAHULUAN 1](#_Toc77284661)

[1.1 Latar Belakang 1](#_Toc77284662)

[1.2 Rumusan Masalah 1](#_Toc77284663)

[1.3 Tujuan 1](#_Toc77284664)

[1.4 Lingkup 1](#_Toc77284665)

[1.5 Hasil yang Diharapkan 1](#_Toc77284666)

[1.6 Pendekatan 1](#_Toc77284667)

[1.7 Sistematika Penyajian 1](#_Toc77284668)

[BAB II TINJAUAN PUSTAKA 1](#_Toc77284669)

[2.1 Pengertian Manajemen Rapat 1](#_Toc77284670)

[2.2 Konsep Manajemen Rapat 1](#_Toc77284671)

[2.3 Tujuan Rapat 1](#_Toc77284672)

[2.4 Metode Pengumpulan Data 1](#_Toc77284673)

[2.5 Metode Pengembangan Perangkat Lunak 1](#_Toc77284674)

[2.6 Website 1](#_Toc77284675)

[2.7 MySQL 1](#_Toc77284676)

[2.8 Bahasa Pemrograman PHP 1](#_Toc77284677)

[2.9 Laravel 1](#_Toc77284678)

[BAB III ANALISIS DAN DESAIN 1](#_Toc77284679)

[3.1 Gambaran Umum Sistem yang Berjalan Saat Ini 1](#_Toc77284680)

[3.2 Gambaran Umum Aplikasi Manajemen Rapat 1](#_Toc77284681)

[3.2.1 Proses Bisnis Mencari Jadwal Rapat **Error! Bookmark not defined.**](#_Toc77284682)

[3.2.2 Proses Bisnis Autentikasi **Error! Bookmark not defined.**](#_Toc77284683)

[3.2.3 Proses Bisnis Mengelola Jadwal Rapat **Error! Bookmark not defined.**](#_Toc77284684)

[3.2.4 Proses Bisnis Mengelola Hasil Rapat **Error! Bookmark not defined.**](#_Toc77284685)

[3.2.5 Proses Bisnis Mengelola Absensi Peserta Rapat **Error! Bookmark not defined.**](#_Toc77284686)

[3.2.6 Proses Bisnis Notifikasi **Error! Bookmark not defined.**](#_Toc77284687)

[3.2.7 Proses Bisnis Mengirim *Email* **Error! Bookmark not defined.**](#_Toc77284688)

[3.2.8 Menambahkan Akun *User* 1](#_Toc77284689)

[3.2.9 Autentikasi 1](#_Toc77284690)

[3.2.10 Mengelola Jadwal Rapat 1](#_Toc77284691)

[3.2.11 Mengelola Dokumen Hasil Rapat 1](#_Toc77284692)

[3.2.12 Mengelola Absensi Peserta Rapat **Error! Bookmark not defined.**](#_Toc77284693)

[3.2.13 Mencari Dokumen Hasil Rapat **Error! Bookmark not defined.**](#_Toc77284694)

[3.3 Modul Aplikasi Manajemen Rapat 1](#_Toc77284695)

[3.4 User Characteristic 1](#_Toc77284696)

[3.4.1 User Group 1](#_Toc77284697)

[3.4.1.1 *User*-Group-1 1](#_Toc77284698)

[3.4.1.2 *User*-Group-2 1](#_Toc77284699)

[3.4.1.3 *User*-Group-3 1](#_Toc77284700)

[3.4.1.4 *User*-Group-4 **Error! Bookmark not defined.**](#_Toc77284701)

[3.5 Use Case Diagram Pada Aplikasi Manajemen Rapat 1](#_Toc77284702)

[3.6 Use Case Scenario 1](#_Toc77284703)

[3.6.1 *Use Case Scenario* FR-01 1](#_Toc77284704)

[3.6.2 *Use Case Scenario* FR-02 **Error! Bookmark not defined.**](#_Toc77284705)

[*3.6.2.1* *Use Case Scenario Login* **Error! Bookmark not defined.**](#_Toc77284706)

[*3.6.2.2* *Use Case Scenario Logout* **Error! Bookmark not defined.**](#_Toc77284707)

[3.6.3 *Use Case Scenario* FR-03 **Error! Bookmark not defined.**](#_Toc77284708)

[3.6.3.1 *Use Case Scenario* Menambah Absensi Peserta Rapat **Error! Bookmark not defined.**](#_Toc77284709)

[3.6.3.2 *Use Case Scenario* Mengedit Absensi Peserta Rapat **Error! Bookmark not defined.**](#_Toc77284710)

[3.6.4 *Use Case Scenario* FR-04 **Error! Bookmark not defined.**](#_Toc77284711)

[3.6.4.1 *Use Case Scenario* Menambahkan Hasil Rapat **Error! Bookmark not defined.**](#_Toc77284712)

[3.6.4.2 *Use Case Scenario* Mengedit Dokumen Hasil Rapat **Error! Bookmark not defined.**](#_Toc77284713)

[3.6.4.3 *Use Case Scenario* Menghapus Dokumen Hasil Rapat **Error! Bookmark not defined.**](#_Toc77284714)

[3.6.5 *Use Case Scenario* FR-05 **Error! Bookmark not defined.**](#_Toc77284715)

[3.6.6 *Use Case Scenario* FR-06 **Error! Bookmark not defined.**](#_Toc77284716)

[3.6.6.1 *Use Case Scenario* Menambahkan Jadwal Rapat **Error! Bookmark not defined.**](#_Toc77284717)

[3.6.6.2 *Use Case Scenario* Mengedit Jadwal Rapat **Error! Bookmark not defined.**](#_Toc77284718)

[3.6.6.3 *Use Case Scenario* Menghapus Jadwal Rapat **Error! Bookmark not defined.**](#_Toc77284719)

[3.6.7 *Use Case Scenario* FR-07 **Error! Bookmark not defined.**](#_Toc77284720)

[3.6.8 *Use Case Scenario* FR-08 **Error! Bookmark not defined.**](#_Toc77284721)

[3.6.9 *Use Case Scenario* FR-09 **Error! Bookmark not defined.**](#_Toc77284722)

[3.6.10 *Use Case Scenario* FR-10 **Error! Bookmark not defined.**](#_Toc77284723)

[3.7 Desain *Interface* Aplikasi 1](#_Toc77284724)

[3.7.1 Desain *User Interface* Halaman Utama User 1](#_Toc77284725)

[3.7.2 Desain *User Interface* Autentikasi 1](#_Toc77284726)

[3.7.3 Desain *User Interface* Hasil Rapat 1](#_Toc77284727)

[3.7.4 Desain *User Interface* Jadwal Rapat 1](#_Toc77284728)

[3.7.5 Desain *User Interface* Absensi Peserta Rapat 1](#_Toc77284729)

[*3.8* *Entity Relationship Diagram* 1](#_Toc77284730)

[*3.9* *Conceptual Data Model* 1](#_Toc77284731)

[*3.10* *Physical Data Model* 1](#_Toc77284732)

[BAB IV IMPLENTASI 1](#_Toc77284733)

[4.1 Kebutuhan Implementasi 1](#_Toc77284734)

[4.2 Batasan Implementasi 1](#_Toc77284735)

[4.3 Implementasi 1](#_Toc77284736)

[4.3.1 Implementasi Kode Autentikasi 1](#_Toc77284737)

[4.3.2 Implementasi Kode Menambahkan Akun *User* 1](#_Toc77284738)

[4.3.3 Implementasi Kode Mengubah Akun *User* 1](#_Toc77284739)

[4.3.4 Implementasi Kode Melihat Akun *User* 1](#_Toc77284740)

[4.3.5 Implementasi Kode Mencari Jadwal Rapat 1](#_Toc77284741)

[4.3.6 Implementasi Kode Menambah Jadwal Rapat 1](#_Toc77284742)

[4.3.7 Implementasi Kode Melihat Jadwal Rapat 1](#_Toc77284743)

[4.3.8 Implementasi Kode Mengubah Jadwal Rapat 1](#_Toc77284744)

[4.3.9 Implementadi Kode Menghapus Jadwal Rapat 1](#_Toc77284745)

[4.3.10 Implementasi Kode Menambah Hasil Rapat 1](#_Toc77284746)

[4.3.11 Implementasi Kode Melihat Hasil Rapat 1](#_Toc77284747)

[4.3.12 Implementasi Kode Mengubah Hasil Rapat 1](#_Toc77284748)

[4.3.13 Implementasi Kode Mencari Hasil Rapat 1](#_Toc77284749)

[4.3.14 Implementasi Kode Menambahkan Absensi Peserta Rapat 1](#_Toc77284750)

[4.3.15 Implementasi Kode Melihat Absensi Peserta Rapat 1](#_Toc77284751)

[4.3.16 Implementasi Kode Mengubah Absensi Peserta Rapat 1](#_Toc77284752)

[4.3.17 Implementasi Kode Notifikasi 1](#_Toc77284753)

[4.3.18 Implementasi Kode Mengirim Email 1](#_Toc77284754)

[Daftar Pustaka dan Rujukan 1](#_Toc77284755)

[LAMPIRAN 1](#_Toc77284756)

# DAFTAR TABEL

[Tabel 1 Daftar Modul Aplikasi Manajemen Rapat 1](#_Toc67992512)

[Tabel 2 Fungsi Utama Aplikasi dan Deskripsi 1](#_Toc67992513)

[Tabel 3 User Group 1](#_Toc67992514)

[Tabel 4 Use Case Scenario Meregistrasi User **Error! Bookmark not defined.**](#_Toc67992515)

[Tabel 5 Use Case Scenario Login **Error! Bookmark not defined.**](#_Toc67992516)

[Tabel 6 Use Case Scenario Logout **Error! Bookmark not defined.**](#_Toc67992517)

[Tabel 7 Use Case Scenario Menambah Absensi Peserta Rapat **Error! Bookmark not defined.**](#_Toc67992518)

[Tabel 8 Use Case Scenario Mengedit Absensi Peserta Rapat **Error! Bookmark not defined.**](#_Toc67992519)

[Tabel 9 Use Case Scenario Menghapus Absensi Peserta Rapat **Error! Bookmark not defined.**](#_Toc67992520)

[Tabel 10 Use Case Scenario Menambahkan Dokumen Hasil Rapat **Error! Bookmark not defined.**](#_Toc67992521)

[Tabel 11 Use Case Scenario Mengedit Dokumen Hasil Rapat **Error! Bookmark not defined.**](#_Toc67992522)

[Tabel 12 Use Case Scenario Menghapus Dokumen Hasil Rapat **Error! Bookmark not defined.**](#_Toc67992523)

[Tabel 13 Use Case Mengexspor Daftar Hadir Peserta Rapat **Error! Bookmark not defined.**](#_Toc67992524)

[Tabel 14 Use Case Scenario Menambahkan Jadwal Rapat **Error! Bookmark not defined.**](#_Toc67992525)

[Tabel 15 Use Case Scenario Mengedit Jadwal Rapat **Error! Bookmark not defined.**](#_Toc67992526)

[Tabel 16 Use Case Scenario Menghapus Jadwal Rapat **Error! Bookmark not defined.**](#_Toc67992527)

[Tabel 17 Use Case Scenario Mengubah Password User **Error! Bookmark not defined.**](#_Toc67992528)

[Tabel 18 Use Case Scenario Import Data Absensi Rapat **Error! Bookmark not defined.**](#_Toc67992529)

[Tabel 19 Use Case Scenario Mencari Dokumen Hasil Rapat **Error! Bookmark not defined.**](#_Toc67992530)

[Tabel 20 Use Case Scenario Mencari Jadwal Rapat **Error! Bookmark not defined.**](#_Toc67992531)

[Tabel 21 Kebutuhan Perangkat Keras 1](#_Toc67992532)

[Tabel 22 Kebutuhan Perangkat Lunak 1](#_Toc67992533)

# DAFTAR GAMBAR

[Gambar 1 Konsep MVC(Model View Controller) Laravel Framework 1](#_Toc67992534)

[Gambar 2 Current System **Error! Bookmark not defined.**](#_Toc67992535)

[Gambar 3 Proses Bisnis Mencari Jadwal Rapat **Error! Bookmark not defined.**](#_Toc67992536)

[Gambar 4 Proses Bisnis Meregistrasi User **Error! Bookmark not defined.**](#_Toc67992537)

[Gambar 5 Proses Bisnis Autentikasi **Error! Bookmark not defined.**](#_Toc67992538)

[Gambar 6 Proses Bisnis Mengelola Jadwal Rapat **Error! Bookmark not defined.**](#_Toc67992539)

[Gambar 7 Proses Bisnis Mengelola Dokumen Hasil Rapat **Error! Bookmark not defined.**](#_Toc67992540)

[Gambar 8 Proses Bisnis Mengelola Absensi Peserta Rapat **Error! Bookmark not defined.**](#_Toc67992541)

[Gambar 9 Proses Bisnis Mencari Dokumen Hasil Rapat **Error! Bookmark not defined.**](#_Toc67992542)

[Gambar 10 Proses Bisnis Import Data Absensi Rapat **Error! Bookmark not defined.**](#_Toc67992543)

[Gambar 11 Proses Bisnis Mengubah Password User **Error! Bookmark not defined.**](#_Toc67992544)

[Gambar 12 Proses Bisnis Mengexspor Daftar Hadir Peserta Rapat **Error! Bookmark not defined.**](#_Toc67992545)

[Gambar 13 Use Case Diagram **Error! Bookmark not defined.**](#_Toc67992546)

[Gambar 14 Tampilan Halaman Utama Aplikasi pada User 1](#_Toc67992547)

[Gambar 15 Tampilan Halaman Login Aplikasi pada User 1](#_Toc67992548)

[Gambar 16 Tampilan Utama Halaman Menu Hasil Rapat 1](#_Toc67992549)

[Gambar 17 Tampilan Utama Halaman Menu Jadwal Rapat 1](#_Toc67992550)

[Gambar 18 Tampilan Halaman Absensi Peserta Rapat 1](#_Toc67992551)

[Gambar 19 Entity Relationship Diagram 1](#_Toc67992552)

[Gambar 20 Conseptual Data Model **Error! Bookmark not defined.**](#_Toc67992553)

[Gambar 21 Physical Data Model **Error! Bookmark not defined.**](#_Toc67992554)

[Gambar 22 Kode Autentikasi **Error! Bookmark not defined.**](#_Toc67992555)

[Gambar 23 Tampilan Autentikasi **Error! Bookmark not defined.**](#_Toc67992556)

[Gambar 24 Implementasi Kode Add User **Error! Bookmark not defined.**](#_Toc67992557)

[Gambar 25 Tampilan Add User **Error! Bookmark not defined.**](#_Toc67992558)

[Gambar 26 Implementasi Kode Membuat Jadwal Rapat **Error! Bookmark not defined.**](#_Toc67992559)

[Gambar 27 Tampilan Membuat Jadwal Rapat **Error! Bookmark not defined.**](#_Toc67992560)

[Gambar 28 Tampilan Mengedit Jadwal Rapat **Error! Bookmark not defined.**](#_Toc67992561)

[Gambar 29 Implementasi Kode Menghapus Jadwal Rapat **Error! Bookmark not defined.**](#_Toc67992562)

[Gambar 30 Tampilan Menghapus Jadwal Rapat **Error! Bookmark not defined.**](#_Toc67992563)

[Gambar 31 Implementasi Kode Detail Rapat **Error! Bookmark not defined.**](#_Toc67992564)

[Gambar 32 Tampilan Detail Rapat **Error! Bookmark not defined.**](#_Toc67992565)

[Gambar 33 Implementasi Kode Mencari Dokumen Hasil Rapat **Error! Bookmark not defined.**](#_Toc67992566)

[Gambar 34 Tampilan Mencari Dokumen Hasil Rapat **Error! Bookmark not defined.**](#_Toc67992567)

BAB I  
PENDAHULUAN

Pada bab ini dijelaskan mengenai latar belakang dari topik yang dijadikan tugas akhir, rumusan masalah yang menjadi acuan pengerjaan tugas akhir, tujuan dari dilaksanakannya tugas akhir, lingkup atau cakupan yang dibahas dan dikerjakan dalam tugas akhir, pendekatan yang digunakan dalam pengerjaan tugas akhir, dan sistematika penyajian dokumen tugas akhir.

Latar Belakang

Rapat di kampus Institut Teknologi Del merupakan salah satu kegiatan formal yang dihadiri oleh Ketua Prodi dan Dosen yang akan merundingkan suatu masalah atau kepentingan bersama untuk memberikan penjelasan, serta menyelesaikan suatu masalah, dengan tujuan memperoleh suatu hasil yang disepakati bersama. Rapat di Fakultas Informatika dan Elektro terdapat 2 jenis rapat, yaitu rapat prodi dan rapat bulanan. Rapat prodi dilaksanakan setiap 1 kali dalam sebulan, dan waktu pelaksanaan rapat tidak ditentukan diawal, penentuan jadwal rapat prodi ini ditentukan secara musyawarah dengan para dosen di setiap prodi. Rapat bulanan prodi D3 Teknologi Informasi dilaksanakan setiap hari rabu pada sore hari. Untuk waktu pelaksanaan rapat bulanan juga ditentukan secara musyawarah dengan dosen prodi, hal ini untuk memastikan dosen dapat menghadiri rapat tersebut. Rapat bulanan dilaksanakan apabila terdapat masalah pada prodi baik mahasiswa maupun akademik

Rapat pada masa *covid-19* dilaksanakan secara online, dimana Ketua Prodi dengan para dosen melakukan rapat dengan *online meeting* menggunakan aplikasi zoom. Ketua Prodi akan memilih notulis pada saat membuat undangan rapat. Notulis yang dipilih akan mencatat hasil rapat, mencatat daftar peserta rapat, dan akan membuat dokumentasi hasil rapat. Notulis akan menyimpan dokumen hasil rapat pada server simargala, hal ini bertujuan agar dosen dapat mengakses dokumen hasil rapat yang sudah berlansung.

Hal tersebut menimbulkan ketidakefisienan dalam waktu karena dokumen hasil rapat tidak terdokumentasi dengan teratur. Permasalahan tersebut akan menjadi topik permasalahan dalam Tugas Akhir ini. Dalam aplikasi web ini terdapat 16 fungsi yang mempermudah Ketua Prodi maupun notulis untuk melakukan manajemen rapat. Disediakan beberapa fungsi seperti mengelola jadwal rapat, mengirimkan undangan rapat, mengelola hasil dokumen rapat, mengelola daftar peserta rapat, pencarian dokumen rapat, *import* file rapat maupun dokumentasi rapat, dan notifikasi undangan rapat.

Dalam proses pengembangan perangkat lunak Aplikasi Manajemen Rapat yang akan dibangun oleh penulis akan menggunakan metode Agile Software Development. Agile software development merupakan metode pengembangan perangkat lunak yang bersifat cepat beradaptasi terhadap perubahan-perubahan sesuai dengan kebutuhan pengguna [6] [7]. Pada metode perangkat lunak *Agile Software Development* terdapat 6 tahap pengembangan, yaitu studi literatur, requirement gathering, perancangan, implementasi, testing dan pengumpulan laporan. Penggunaan metode ini melatih anggota tim untuk berkomunikasi yang baik kepada *User*, memungkinkan anggota tim dalam menentukan keputusan dengan cepat dan berkualitas, serta memiliki potensi yang baik dalam menangani setiap perubahan.

Berdasarkan latar belakang yang telah dipaparkan, maka perlu dibangun Aplikasi Manajemen Rapat berbasis web dengan menggunakan metode pengembangan perangkat lunak yaitu *Agile Software Development*. Aplikasi ini berguna bagi Ketua Prodi dan dosen prodi D3 Teknologi Informasi untuk mempermudah pengelolaan manajemen rapat. Oleh sebab itu, *User* dapat mendokumentasikan keperluan rapat seperti jadwal rapat, lokasi rapat, dokumen hasil rapat dan daftar peserta rapat.

Rumusan Masalah

Rumusan masalah yang menjadi acuan pengerjaan tugas akhir adalah bagaimana pengembangan aplikasi manajemen rapat?

Tujuan

Tujuan dari pengerjaan Tugas Akhir ini yaitu membangun Aplikasi Manajemen Rapat menggunakan metodologi *Agile* dan Framework Laravel sehingga dapat membantu prodi D3 Teknologi Informasi untuk mengelola rapat, mengelola hasil rapat, dan menyimpan hasil rapat.

Lingkup

Batasan atau lingkup dari pelaksanaan tugas akhir ini adalah:

1. Menggunakan *Framework Laravel*
2. Menggunakan metode pengembangan *Agile*
3. Studi kasus aplikasi manajemen rapat yaitu prodi D3 Teknologi Informasi

Hasil yang Diharapkan

Hasil yang diharapkan dari Tugas Akhir ini adalah Aplikasi Manajemen Rapat berbasis *web*, yaitu sebuah aplikasi yang dapat membantu Ketua Prodi, dan Dosen di Kampus Institut Teknologi Del dalam manajemen rapat, mengelola dokumen hasil rapat, dan melakukan pencarian khususnya pada prodi D3 Teknologi Informasi terhadap dokumen hasil rapat.

Pendekatan

Pada Tugas Akhir ini tahapan yang ditempuh dalam pelaksanaan tugas akhir ini adalah:

1. Studi Literatur

Studi literatur dapat digunakan untuk mendukung pembuatan Tugas Akhir dalam hal memperoleh informasi dan dasar teori dari buku, jurnal, dan internet. Hal ini dilakukan oleh setiap anggota kelompok dengan membaca informasi yang berhubungan dengan topik Tugas Akhir.

1. *Requirement Gathering*

*Requirement gathering* atau pengumpulan kebutuhan dilakukan untuk mengumpulkan kebutuhan-kebutuhan yang akan menjadi sarana bagi pengembang untuk membangun Aplikasi Manajemen Rapat studi kasus: D3 Teknologi Informasi, Fakultas FITE. Adapun requirement yang dilakukan adalah *User* requirement dan *software requirement*. *User* *requirement* adalah spesifikasi kebutuhan perangkat lunak yang didapat dari *User* atau pengguna. *Software requirement* adalah spesifikasi kebutuhan yang harus dipenuhi oleh perangkat lunak.

1. Perancangan Desain

Proses perancangan desain ini dibutuhkan untuk menjelaskan bagaimana sistem manajemen rapat yang akan dibangun. Pada tahap ini, dilakukan perancangan desain antarmuka pengguna dan perancangan struktur sistem aplikasi manajemen rapat di Kampus Institut Teknologi Del yang berbasis web dengan menerapkan *Continuous Integration* sebagai metode *testing*.

1. Implementasi Aplikasi

Pada tahap ini, hasil dari perancangan desain digunakan untuk membangun aplikasi manajemen rapat dengan menerapkan *Continuous Integration*. Hasil dari tahapan ini akan diuji agar memperoleh aplikasi sesuai dengan yang dibutuhkan.

1. Pengujian Aplikasi

Pada tahap ini, akan dilakukan proses pengujian terhadap aplikasi manajemen rapat yang telah diimplementasi. Hal ini dilakukan untuk menjamin aplikasi yang dibangun sesuai terhadap rancangan proses bisnis manajemen rapat yang telah dirancang sebelumnya atau berjalan sesuai dengan kebutuhan.

1. Penyusunan Laporan Akhir

Pada tahap ini, akan dilakukan penulisan laporan hasil akhir dari penelitian yang berisi mengenai hasil analisis dan kesimpulan yang dilakukan terhadap pengembangan aplikasi sesuai dengan tujuan dan rumusan masalah yang ada.

Sistematika Penyajian

Pada sub bab sistematika penyajian menjelaskan tentang uraian dari dokumen tugas akhir. Pada dokumen ini terdiri dari 6 bab yaitu:

**BAB I PENDAHULUAN**

Pada bab pendahuluan dijelaskan mengenai latar belakang, rumusan masalah, tujuan, lingkup, pendekatan, dan sistematika penyajian dalam tugas akhir.

**BAB II TINJAUAN PUSTAKA**

Pada bab tinjauan pustaka dijelaskan mengenai dasar-dasar teori yang berkaitan dengan topik tugas akhir.

**BAB III ANALISIS DAN DESAIN**

Pada bab analisis dijelaskan mengenai pengenalan proyek, perencanaan proyek, analisis yang dilakukan terhadap sistem yang sebelumnya ada (*current system*), dan rancangan sistem yang akan dikembangkan (*target system*). Pada bab ini juga dijelaskan mengenai perancangan dari sistem yang akan dikembangkan dalam bentuk data requirement, *conceptual data*, *physical data*, dan *class diagram*.

**BAB IV IMPLEMENTASI**

Pada bab implementasi dijelaskan mengenai implementasi dari aplikasi yang akan dikembangkan dengan metode pengembangan aplikasi *Agile*.

**BAB V HASIL DAN PEMBAHASAN**

Pada bab hasil dan pembahasan dijelaskan mengenai hasil yang diperoleh setelah melakukan tahap implementasi sistem.

**BAB VI KESIMPULAN DAN SARAN**

Institut Teknologi Del Pada bab kesimpulan dan saran dijelaskan mengenai kesimpulan yang diperoleh dari pengerjaan tugas akhir dan memberikan saran apabila memungkinkan penelitian ini untuk dilanjutkan atau dikembangkan lagi.

# 

BAB II  
TINJAUAN PUSTAKA

Pada bab tinjauan pustaka dijelaskan mengenai informasi teoritis atau dasar teori dan pustaka yang mendukung pengerjaan tugas akhir.

Pengertian Manajemen Rapat

Rapat merupakan pertemuan antara dua orang atau lebih sebagai alat komunikasi resmi dan formal untuk membahas dan memecahkan suatu masalah yang bersifat tatap muka untuk menyatukan pemikiran dalam menyelesaikan permasalahan dan pengambilan keputusan secara musyawarah untuk mufakat [8].

Pengertian Manajemen menurut Henry Fayol, merupakan proses perencanaan, penyusunan, dengan memanfaatkan sumber daya, termasuk sumber daya manusia untuk menentukan dan mencapai suatu tujuan tertentu secara efektif dan efisien.

Berdasarkan penjelasan diatas maka dapat dipahami bahwa, Manajemen Rapat adalah kegiatan untuk merencanakan, mengorganisasikan, memimpin, dan mengatur rapat yang merupakan suatu kegiatan formal tatap muka resmi yang telah diagendakan yang dilakukan oleh dua orang atau lebih yang bertujuan untuk membahas suatu permasalahan, mencari jalan keluarnya dan mengambil keputusannya agar mencapai tujuan dengan memanfaatkan sumber daya secara efektif dan efisien [8]. Jadi konsep manajemen rapat adalah suatu aktivitas mengatur dan mengetahui secara tepat apa yang ingin dikerjakan melalui proses perencanaan, pengorganisasian, pengarahan, dan pengendalian.

Konsep Manajemen Rapat

Dalam konsep manajemen rapat, terdapat beberapa aktivitas yang harus dipersiapkan:

1. Perancangan Rapat

Perancangan rapat merupakan suatu tahap untuk menetapkan tujuan rapat dan mempersiapkan segala kebutuhan selama rapat berlangsung. Hal tersebut biasanya dipersiapkan oleh ketua rapat dan notulis rapat, meliputi daftar hadir rapat, mempersiapkan bahan-bahan rapat, dan mengatur ruangan rapat yang akan digunakan serta perlengkapan yang lain.

1. Pengorganisasian Rapat

Pengorganisasian rapat merupakan suatu tahap dalam mempersiapkan kegiatan rapat yaitu pembagian tugas dalam rapat. Tujuannya agar setiap peserta rapat yang memiliki tugas seperti ketua rapat, notulis, pembaca bahan rapat, dan peserta rapat dapat mengetahui tugas masing-masing dan tidak menyimpang dari yang diharapkan.

1. Pengarahan Rapat

Pengarahan rapat merupakan suatu tahap memberikan suatu arahan kepada peserta rapat sebelum memulai rapat. Arahan meliputi topik yang akan dibahas dan target rapat yang harus dicapai. Ketua rapat diharapkan memberikan arahan terhadap peserta rapat, agar ikut berpartisipasi sesuai dengan tugas yang diemban. Sehingga peserta rapat dapat fokus pada materi yang dibahas dan tidak menyimpang dari tujuan rapat.

1. Pengendalian Rapat

Pengendalian rapat merupakan suatu tahap dimana ketua rapat mengendalikan kegiatan rapat agar sesuai dengan susunan rapat yang sudah di rencanakan. Tujuannya memperoleh hasil rapat, dan apabila belum memperoleh hasil rapat agar rapat dapat di tindak lanjut sehingga tidak menghabiskan banyak waktu.

Tujuan Rapat

Beberapa tujuan rapat diadakan diantaranya sebagai berikut:

1. Untuk memecahkan suatu permasalahan, hal ini bertujuan untuk membahas masalah dengan peserta rapat dan merundingkannya agar masalah tersebut dapat diselesaikan.
2. Menyampaikan beberapa informasi, tujuan dilakukannya hal ini untuk menyampaikan informasi secara langsung kepada peserta rapat agar seluruh peserta rapat memiliki persepsi yang sama terhadap informasi yang diterima.
3. Mempersiapkan suatu kegiatan, hal ini bertujuan untuk membahas suatu kegiatan yang akan diselenggarakan agar dapat terlaksana dengan baik dan tepat.
4. Mendapatkan saran dan kritik dari peserta rapat, hal ini bertujuan untuk mempertimbangkan seluruh saran dan kritik terhadap masalah yang sedang di bahas agar mendapatkan solusi yang terbaik.
5. Menjalin kerja sama di antara anggota, hal ini bertujuan untuk menciptakan rasa bertanggung jawab atas permasalahan yang sedang dibahas agar seluruh peserta rapat ambil bagian dalam rapat.

Metode Pengumpulan Data

Pengumpulan data pada penulisan tugas akhir ini dilakukan dengan metode sebagai berikut :

* 1. Wawancara

Pada pengumpulan data yang dilakukan oleh penulis menggunakan metode wawancara untuk mencari informasi tentang proses rapat yang dilakukan secara online oleh prodi D3 Teknologi Informasi. Wawancara dilaksanakan secara *online meeting* dengan Ketua Prodi D3 Teknologi Informasi dan dosen yang terkait dalam proses rapat yang dilakukan. Dosen yang menjadi narasumber merupakan dosen yang pernah berperan menjadi notulis dalam rapat online. Pada wawancara ini penulis mencari informasi mengenai proses manajemen rapat yang berlansung secara online, masalah yang dialami oleh dosen dengan mengadakan rapat online, dan tanggapan terhadap solusi yang diberikan penulis yaitu sistem manajamen rapat yang akan dibangun.

* 1. Studi Literatur

Penulis mencari dan mengumpulkan jurnal, skripsi beserta buku-buku yang berhubungan dengan pengembangan sistem manajemen rapat. Bahan tersebut digunakan oleh penulis sebagai bahan referensi dari pengembangan system dan penulisan tugas akhir yang akan dibuat.

Metode Pengembangan Perangkat Lunak

Pada pengembangan Aplikasi Manajemen Rapat berbasis *web* akan dilakukan dengan metode *Agile* *Software Development*. *Agile software Development* merupakan metode pengembangan perangkat lunak yang bersifat cepat beradaptasi terhadap perubahan-perubahan sesuai dengan kebutuhan pengguna [9]. Pada metode perangkat lunak *Agile Software Development* terdapat 6 tahap pengembangan, yaitu studi literatur, *requirement gathering*, perancangan, implementasi, *testing* dan pengumpulan laporan. Penggunaan metode ini melatih anggota tim untuk berkomunikasi yang baik kepada *User,* memungkinkan anggota tim dalam menentukan keputusan dengan cepat dan berkualitas, serta memiliki potensi yang baik dalam menangani setiap perubahan.

1. Perencanaan

Pada tahap perencanaan, *developer* akan menyusun perencanaan sistem yang akan dikembangkan dengan cara mengumpulkan data terhadap *client.* Pengumpulan data ini dapat diperoleh dengan melakukan wawancara yang bersifat tatap muka atau dengan cara pengumpulan data menggunakan kuesioner. Hal ini bertujuan untuk mendapatkan kebutuhan yang *client* inginkan.

1. Implementasi

Pada tahap implementasi, *developer* akan mengimplementasikan kode program untuk mengembangkan sistem, yaitu mengubah dokumentasi desain menjadi perangkat lunak yang sesuai dengan kebutuhan *client.*

1. Dokumentasi

Pada tahap dokumentasi, *developer* akan mencatat modul dan fungsi yang digunakan pada sistem. Hal ini bertujuan sebagai dokumentasi selama proses pengembangan perangkat lunak dan mempermudah *developer* untuk melanjutkan proses pengembangan sistem yang berkelanjutan.

1. *Deployment*

Pada tahap *deployment,* sistem sudah dapat digunakan oleh *client* tanpa memiliki kesalahan pada sistem.

1. *Testing*

Pada tahap ini *developer,* akan melakukan pengujian terhadap perangkat lunak agar *developer* dapat menemukan *bug* atau *error* dan segera memperbaiki. Hal ini bertujuan untuk menghasilkan perangkat lunak yang berkualitas dan tidak memiliki kegagalan pada perangkat lunak.

1. *Maintenance*

Pada tahap ini, pemeliharan sistem sangat diperlukan untuk menjaga sistem agar tetap aman dan bebas dari *bug,* oleh karena itu perangkat lunak wajib dipelihara secara berkala.

Website

Secara teknis, web adalah sebuah sistem dengan informasi yang disajikan dalam bentuk teks, gambar, suara dan lain-lain yang tersimpan dalam sebuah server web internet yang disajikan dalam bentuk hiperteks (Simarmata, 2010). Web menjadi sumber data yang sangat besar dan sangat berharga untuk setiap pengguna karena di dalam web kumpulan dokumen saling terhubung dan dapat diakses melalui koneksi internet (Antoniou G, Groth P, Harmelen, Hoekstra, 2012) [7]. Untuk mengakses website, pengguna mengetikkan nama domainnya di *address bar* pada browser dan browser akan menampilkan halaman utama *website* atau beranda. Menurut Rahman HM (2010), Website memuat kumpulan informasi berdasarkan pemilik situs dan dapat digunakan sebagai media tempat penyimpanan informasi, komunikasi maupun transaksi yang dilakukan [8].

MySQL

Database adalah data yang dikumpul dan saling berhubungan yang disimpan secara terstruktur di komputer digunakan untuk mengumpulkan, mengambil, mengatur dan mengelola data dalam menghasilkan informasi yang dibutuhkan. MySQL merupakan program database yang dapat menangani data yang besar, dan bersifat open source. MySQL menggunakan SQL (Structured Query Language) sebagai bahasa dasar untuk mengakses database. MySQL adalah RDBMS (*Relational Database Management System*) yang didistribusikan oleh *General Public License* secara gratis sehingga dapat digunakan secara bebas [9].

Ada beberapa kelebihan pada MySQL diantaranya adalah sebagai berikut.

1. Beberapa pengguna secara bersamaan dapat menggunakan MySQL tanpa mengalami masalah.

2. MySQL mendukung perintah *where* dan *select* untuk perintah *query.*

3. Beberapa lapisan keamanan seperti level akses subnetmask, nama host dan izin akses user dan sandi terenkripsi terdapat pada MySQL.

4. O*perating system,* seperti Windows, Linux, Mac OS X Server dan *operating system* lainnya dapat menjalankan MySQL dengan stabil.

5. Struktur tabel pada MySQL lebih fleksibel dibandingkan dengan database lainnya dalam menangani *ALTER TABLE* [9].

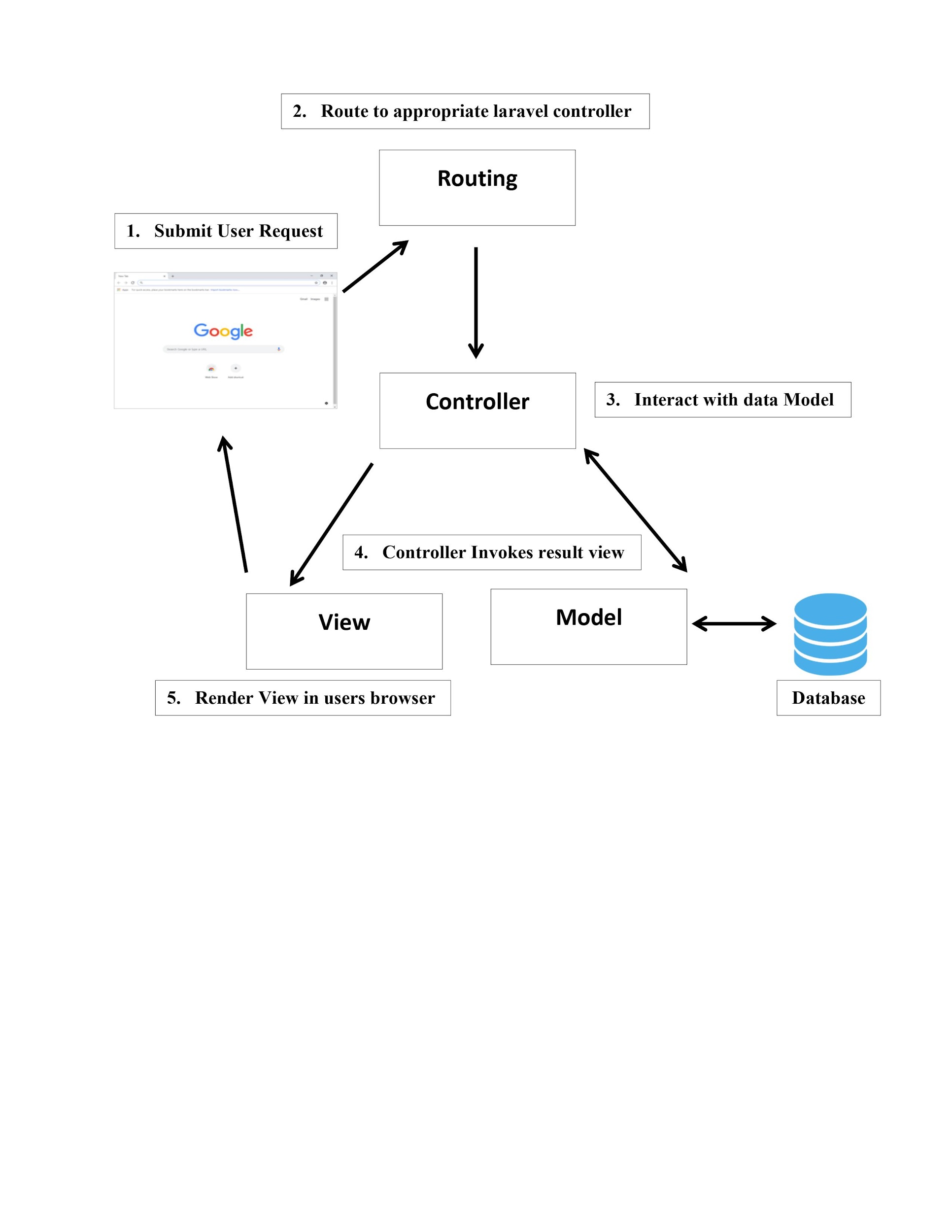
Bahasa Pemrograman PHP

PHP (singkatan dari PHP: Hypertext Preprocessor) adalah bahasa skrip yang dapat disematkan ke dalam skrip HTML yang umumnya digunakan dalam pengembangan website dan memungkinkan halaman web statis menjadi dinamis. Salah satu kekuatan utama PHP adalah kenyataan bahwa karena dapat disematkan langsung di samping kode HTML, tidak perlu menulis program yang memiliki banyak perintah hanya untuk mengeluarkan HTML. HTML dan PHP dapat digunakan secara bergantian sesuai kebutuhan, bekerja bersama satu sama lain secara serempak. Untuk menjalankan skrip PHP, terlebih dahulu harus menginstal dan mengkonfigurasi perangkat lunak PHP / Apache di server masing-masing. PHP secara dasar dapat mendapatkan data dari form, menghasilkan isi halaman web yang dinamik, serta dapat menerima cookies.

Hal yang paling utama terkait kemampuan PHP yang paling signifikan adalah dukungan kepada banyak jenis database. Membuat halaman web yang menggunakan data dari database dengan sangat mudah dapat dilakukan. Berikut ini adalah daftar database yang didukung oleh PHP: Hyperwave, IBM DB2, Informix , Ingres, Interbase, MSQL, Direct MS SQL, MySQL, ODBC, Oracle. PHP juga mendukung komunikasi dengan layanan yang lain menggunakan protokol seperti IMAP, SNMP, NNTP,POP3, HTTP.

Laravel

*Laravel* adalah sebuah framework PHP yang dibangun dengan konsep MVC (Model View Controller). *Laravel* adalah pengembangan aplikasi berbasis *website* berbasis MVP yang ditulis dalam bahasa pemrograman PHP yang dirancang untuk meningkatkan kualitas dari aplikasi dengan mengurangi biaya pengembangan awal dan biaya maintance aplikasi, dan untuk meningkatkan pengalaman bekerja dengan aplikasi dengan menyediakan sintaks yang ekspresif. Konsep *laravel* yaitu MVC (Model View Controller) *Laravel* *Framework* disajikan pada Gambar 1.



**Gambar 1 Konsep MVC(Model View Controller) Laravel Framework [9]**

(Sumber: F. Abdussalam and S. Anggieta Saputra, “PERANCANGAN SISTEM INFORMASI COMPLAINT MANAGEMENT DENGAN METODE RAD MENGGUNAKAN FRAMEWORK LARAVEL,” 2018.)

Laravel menyediakan fitur-fitur yang canggih seperti *dependency injection, database abstraction layer* yang ekspresif, *queue* dan *scheduled jobs, unit* dan *integration testing* dan lainnya [10].

Beberapa fitur yang terdapat di Laravel: Bundles, Eloquent ORM, Application Logic, Reverse Routing, Restful controllers, Class Auto Loading, View Composer, IoC Container, Migration, Unit testing, dan Automatic Pagination.

Ada beberapa keuntungan menggunakan Laravel, dua diantaranya framework yang progresif, dimana terdapat perpustakaan dokumentasi, panduan, video tutorial yang luas, sehingga pengguna dapat mempelajari tanpa kewalahan serta *community framework,* ribuan pengembang telah berkontribusi pada framework Laravel [10].

BAB III  
ANALISIS DAN DESAIN

Pada bab analisis dan desain dijelaskan mengenai pengenalan proyek, perencanaan proyek, analisis yang dilakukan terhadap sistem yang sebelumnya ada (*current system*), dan rancangan sistem yang akan dikembangkan (*target system*). Pada bab ini juga dijelaskan mengenai perancangan dari sistem yang akan dikembangkan dalam bentuk Business *Process Modeling Notation* (BPMN), *mockup, Entity Relational Diagram* (ERD), *class diagram, use case diagram dan use case scenario*.

* 1. Gambaran Umum Sistem yang Berjalan Saat Ini

Gambaran umum proses manajemen rapat di prodi D3 Teknologi Informasi Kampus Institut Teknologi Del yang berjalan saat ini dilakukan secara *online.* Proses manajemen rapat dilakukan oleh Ketua Prodi dan dosen untuk mengadakan rapat.

Ketua Prodi akan menentukan jadwal rapat yang terdiri dari beberapa *component* rapat yaitu nama rapat, hari/tanggal, waktu, tempat, pimpinan rapat, notulen rapat, dan peserta rapat*.* Ketua prodi memilih notulis rapat secara bergilir berdasarkan abjad nama dosen. Setelah jadwal rapat telah ditentukan maka Ketua prodi akan mengirimkan undangan rapat kepada notulis dan dosen sebagai peserta rapat melalui *email*. Notulis akan mengirimkan agenda rapat kepada peserta rapat melalui *group* *chat online* menggunakan aplikasi *Whatsapps* sehari sebelum rapat berlansung. Penentuan lokasi rapat ada 2 jenis yaitu, ruangan untuk rapat yang bersifat *offline* dan *link zoom* untuk rapat *online.* Untuk rapat *offline* ruangan yang dipakai terdiri dari ruangan 512, 421, 422, dan ruangan RMK. Untuk rapat secara *online* penentuan ruangan rapat dilakukan 2 jam sebelum rapat dimulai dengan mengirimkan *link meeting* aplikasi zoom melalui *group* *chat online* menggunakan aplikasi *Whatsapps.*

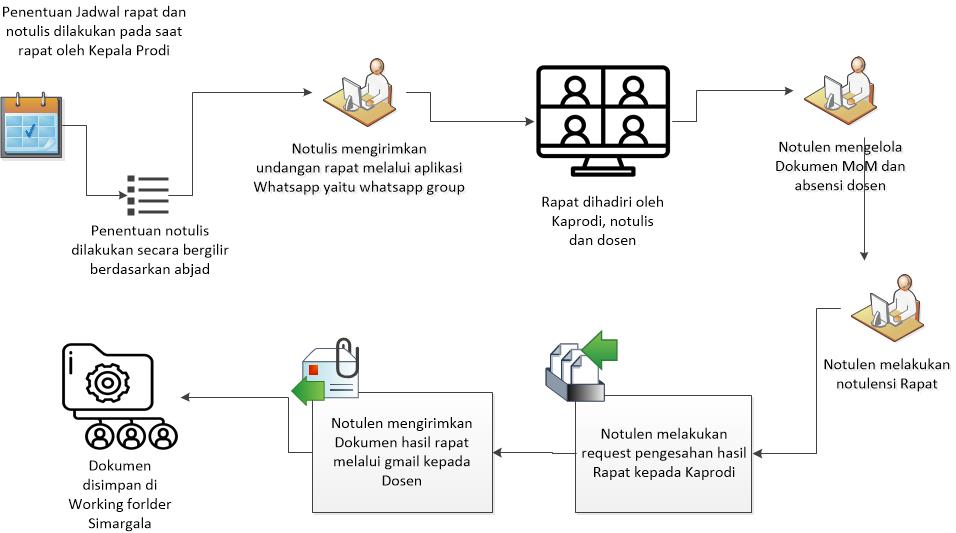
Notulis akan meng*input* hasil rapat ke dalam dokumen *template* MoM*.* Dokumen *template* MoM yang di*input* oleh notulis berisikan hal-hal yang akan ditindaklanjuti untuk rapat selanjutnya dan menuliskan penanggung jawab untuk setiap topik. Data rapat yang telah di*input* ke dalam dokumen *template* MoM akan disimpan sebagai bentuk laporan pertanggungjawaban notulis kepada ketua rapat. Hasil rapat yang diperoleh akan disimpan dalam bentuk dokumen oleh notulis rapat ke *working folder* simargala. Simargala merupakan *server* yang berperan sebagai *working folder* yang berfungsi untuk menyimpan dokumen-dokumen Kampus Institut Teknologi Del dan bisa diakses oleh semua dosen dan staff.

Pendataan daftar hadir peserta rapat yang hadir akan di*input* oleh notulis di form daftar hadir peserta rapat. Pendataan daftar hadir peserta rapat yang tidak dapat hadir dilakukan dengan memberitahukan kepada ketua rapat melalui *group* *chat online* menggunakan aplikasi *Whatsapps*, dengan memberikan konfirmasi ketidakhadiran terhadap rapat dan alasannya. Dengan cara tersebut memerlukan lebih banyak waktu, sehingga arsip daftar hadir peserta rapat tidak akan terdokumentasi dengan cepat karena notulis harus melakukan pengecekkan data dosen yang tidak dapat menghadiri rapat satu persatu melalui aplikasi *Whatsapp* yaitu *whatsapp group.*

Dokumen hasil rapat MoM akan konfirmasi terlebih dahulu kepada ketua rapat yaitu Ketua Prodi untuk memastikan kesesuaian terhadap hasil rapat. Setelah notulis selesai melakukan konfirmasi terhadap hasil rapat MoM maka notulis akan menyimpan dokumen hasil rapat ke *working folder* simargala agar dapat diakses oleh dosen.

Penyimpanan dokumen hasil rapat disimpan secara *softcopy* di *working folder* simargala, dilakukan dengan cara meng*-upload* dokumen ke *working folder* berdasarkan prodi. Hal ini memerlukan waktu bagi ketua rapat dan peserta rapat untuk menerima hasil rapat yaitu dokumen MoM dari notulis. Hasil rapat yaitu dokumen MoM akan dikirimkan oleh notulis ke *email* dosen masing-masing.

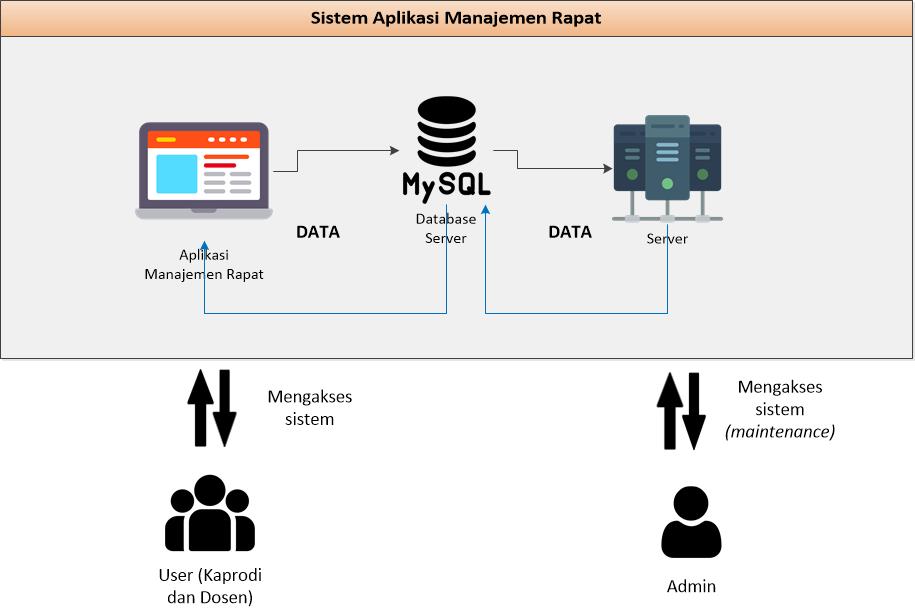
Proses pencarian dokumen MoM rapat sebelumnya dilakukan dengan mencari *folder* berdasarkan prodi kemudian memasukkan *keywords* pada kolom pencarian. Hal ini akan membutuhkan banyak waktu bagi dosen dalam mencari hasil rapat karena dosen harus mencari folder prodi, kemudian mencari dokumen MoM didalam folder prodi. Gambaran sistem dapat digambarkan pada gambar dibawah ini.



* 1. Gambaran Umum Aplikasi Manajemen Rapat

Gambaran umum Aplikasi Manajemen Rapat seperti yang sudah dijelaskan pada latarbelakang akan dibangun berdasarkan dengan proses rapat online yang berlansung di kampus. Aplikasi Manajemen Rapat berbasis Website terdiri dari beberapa fungsi yang saling berhubungan yaitu fungsi mengelola akun *user,* mencari jadwal rapat, autentikasi, mengelola jadwal rapat, mengelola hasil rapat, mengelolah absensi peserta rapat, mencari hasil rapat, notifikasi, dan mengirim email. Aplikasi Manajemen Rapat berbasis *Website* akan dikembangkan menggunakan *framework* Laravel, dan bahasa pemrograman PHP.

Ketua Prodidapat mengakses Aplikasi Manajemen Rapat berbasis *Website* untuk membuat jadwal rapat apabila melakukan pengisian data pada form login sesuai ketentuan, data yang di­-*input user* akan disesuaikan dengan data yang terdapat didalam *database.* *Server* akan menerima *request* dari *user,* dan akan memberikan hak akses kepada *user* apabila data yang di-*input* sesuai. *User* mengakses sistem, dan dapat menggunakan Aplikasi Manajemen Rapat berbasis Website sesuai dengan kebutuhan *user.* Admin akan melakukan *maintenance* apabila ditemukan *error* pada sistem. Gambaran sistem Aplikasi Manajemen Rapat berbasis *Website* dapat dilihat pada gambar.



**Gambar 2 Gambaran umum Aplikasi Manajemen Rapat**

Aplikasi Manajemen Rapat berbasis *Website* terdapat 18 fungsi utama, yaitu mendaftarkan akun *user,* mengubah akun *user,* melihat akun *user,* mencari jadwal rapat, autentikasi, menambahkan jadwal rapat, mengubah jadwal rapat, menghapus jadwal rapat, melihat jadwal rapat, menambahkan hasil rapat, melihat hasil rapat, mengubah hasil rapat, mencari hasil rapat, menambahkan absensi peserta rapat, mengubah absensi peserta rapat, notifikasi dan mengirim email.

* 1. Fungsi Menambahkan *User*

Fungsi menambahkan akun *user* dapat dilakukan oleh admin. Fungsi ini digunakan untuk menambahkan data *user* yaitu ketua prodi dan dosen ke sistem. Data *user* yang akan ditambahkan terdiri dari nama, *email,* dan *password.*

* 1. Fungsi Melihat Akun *User*

Fungsi melihat akun *user* merupakan fungsi yang dilakukan oleh admin untuk melihat akun *user* yang terdaftar didalam sistem. Fungsi ini digunakan untuk melihat detail data dari *user.*

* 1. Fungsi Mencari Akun *User*

Fungsi mencari akun *user* merupakan fungsi yang dilakukan oleh admin, untuk mencari data *user.* Fungsi ini dapat digunakan oleh admin didalam menu user, pada kolom pencarian admin memasukkan *identifier* untuk mencari data *user* sesuai dengan kebutuhan.

* 1. Fungsi Mengubah Akun *User*

Fungsi mengubah akun *user* merupakan fungsi yang dilakukan oleh admin, ketua prodi dan dosen, yaitu mengubah nama, *email,* foto dan *password.* Ketua Prodi dan dosenakan didaftarkan oleh admin dengan membuat *default account,* dimana akun ini terdiri dari nama, *email,* foto dan *password.*

* 1. Fungsi Mencari Jadwal Rapat

Fungsi mencari jadwal rapat merupakan fungsi yang digunakan untuk mencari jadwal rapat pada kolom pencarian jadwal rapat dengan meng*input keyword* sesuai dengan kebutuhan. Fungsi ini digunakan oleh ketua prodi dan dosen yang terdapat pada menu Jadwal.

* 1. Fungsi Autentikasi

Fungsi autentikasi merupakan fungsi yang terdiri dari *login* dan *logout* yang dapat digunakanoleh admin, ketua prodi dan dosen*.* Fungsi *login* digunakan untuk dapat mengakses sistem. Admin, ketua prodi dan dosen akan meng*input username* dan *password* pada form login sistem sesuai dengan ketentuan. Fungsi *logout* digunakan untuk dapat keluar dari sistem.

* 1. Fungsi Menambahkan Jadwal Rapat

Fungsi menambahkan jadwal rapat dilakukan oleh ketua prodi untuk menambahkan jadwal rapat baru.

* 1. Fungsi Melihat Jadwal Rapat

Fungsi melihat jadwal rapat dilakukan oleh ketua prodi dan dosen untuk melihat jadwal rapat yang telah di*upload* oleh Ketua Prodi.

* 1. Fungsi Mengubah Jadwal Rapat

Fungsi mengubah jadwal rapat digunakan oleh Ketua Prodi untuk mengubah jadwal rapat apabila terdapat perubahan pada jadwal yang telah dibuat sebelumnya.

* 1. Fungsi Menghapus Jadwal Rapat

Fungsi menghapus jadwal rapat digunakan oleh ketua prodi untuk menghapus jadwal rapat apabila terdapat jadwal rapat yang tidak perlu untuk dilaksanakan.

* 1. Fungsi Menambahkan Hasil Rapat

Fungsi menambahkan hasil rapat dapat dilakukan oleh dosen yang terpilih menjadi notulis untuk menambahkan data hasil rapat meliputi nama rapat, keterangan waktu, hari/tanggal, tempat, pimpinan rapat, notulis rapat, peserta rapat, agenda rapat, notulen rapat, dokumentasi, dan tindak lanjut.

* 1. Fungsi Melihat Hasil Rapat

Fungsi melihat hasil rapat digunakan untuk melihat hasil rapat yang telah di*upload* oleh dosen yang terpilih menjadi notulis kedalam sistem, agar peserta rapat dapat melihat detail hasil rapat yang telah selesai. Fungsi ini dapat dilakukan oleh ketua prodi dan dosen.

* 1. Fungsi Mengubah Hasil Rapat

Fungsi mengubah hasil rapat digunakan untuk melakukan perubahaan pada data yang tidak sesuai dengan hasil rapat.

* 1. Fungsi Mencari hasil rapat

Fungsi mencari hasil rapat merupakan fungsi yang digunakan untuk mencari hasil rapat pada menu hasil rapat. Fungsi mencari hasil rapat dapat digunakan oleh ketua prodi dan dosen agar lebih cepat menemukan hasil rapat yang dibutuhkan, yaitu dengan memasukkan *keyword* pada kolom pencarian.

* 1. Fungsi Menambahkan Absensi Peserta Rapat

Fungsi menambahkan absensi peserta rapat dilakukan oleh dosen yang terpilih menjadi notulis, yang digunakan untuk menambahkan absensi. Notulis menambahkan daftar peserta rapat yang hadir, pada halaman absensi peserta rapat.

* 1. Fungsi Mengubah Absensi Peserta Rapat

Fungsi mengubah absensi peserta rapat dilakukan oleh dosen yang terpilih menjadi notulis, yang digunakan untuk melakukan perubahanpada data absensi peserta rapat yang tidak sesuai.

* 1. Fungsi Melihat Absensi Peserta Rapat

Fungsi melihat absensi peserta rapat dilakukan oleh dosen yang terpilih menjadi notulis, yang digunakan untuk melihat daftar peserta rapat yang hadir pada saat rapat.

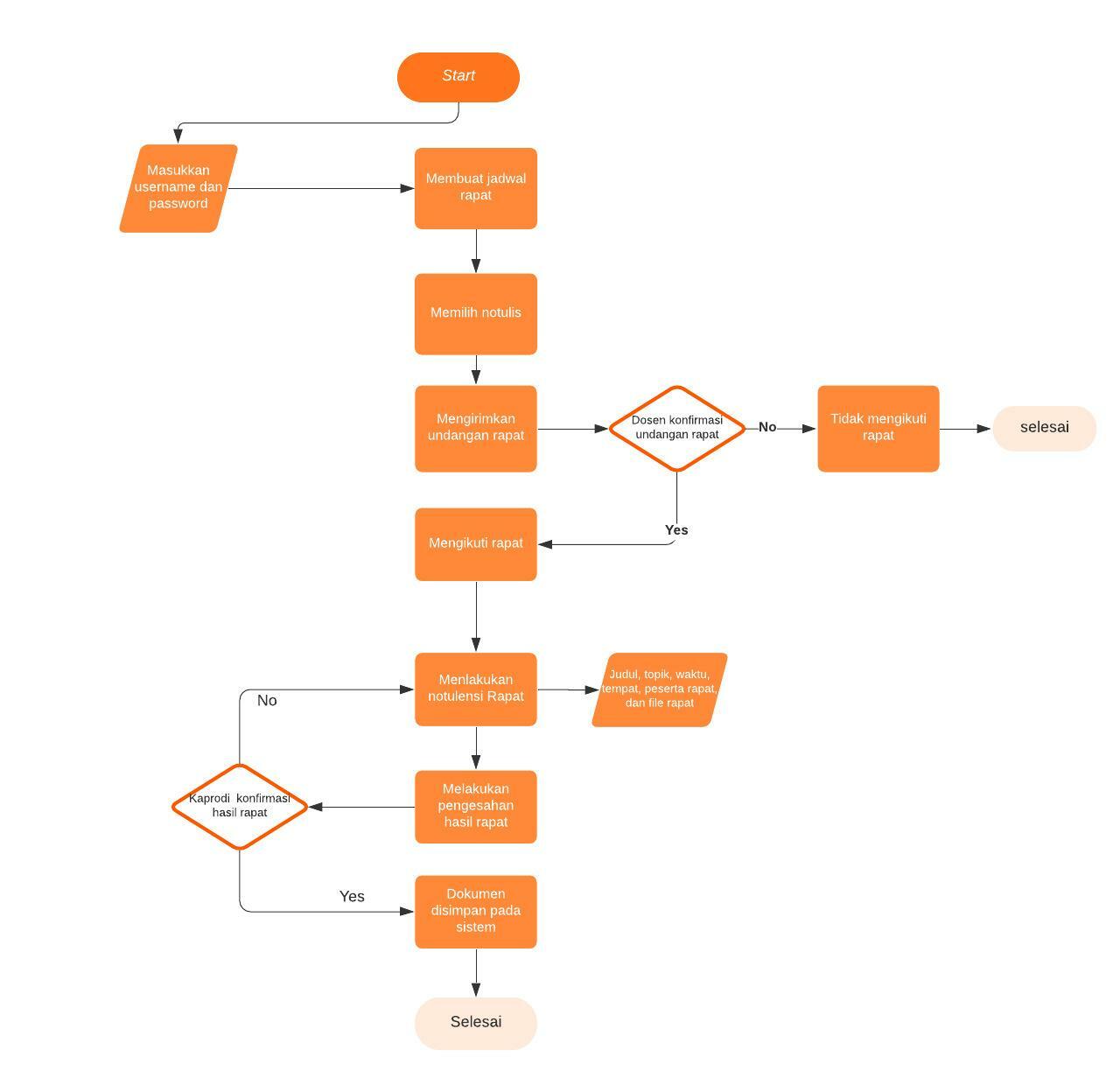
* 1. Fungsi Notifikasi

Fungsi notifikasi merupakan fungsi yang memberikan pemberitahuan dan dilakukan oleh ketua prodi dan dosen. Fungsi notifikasi digunakan untuk memberikan pemberitahuan jadwal rapat kepada dosen, pemberitahuan dosen tidak dapat menghadiri rapat kepada ketua prodi, *request* pengesahan hasil rapat kepada ketua prodi, dan pemberitahuan hasil rapat telah berhasil di*upload* kepada dosen*.*

* 1. Fungsi Mengirim *Email*

Fungsi mengirim *email* merupakan fungsi untuk mengirimkan *email* undangan rapat yang dilakukan oleh ketua prodi. Fungsi mengirim email bertujuan agar dosen dapat mengetahui undangan rapat yang telah dikirim oleh ketua prodi apabila dosen dalam kondisi sedang tidak mengakses Aplikasi Manajemen Rapat.

Aplikasi Manajemen Rapat berbasis *Website* dapat diakses oleh ketua prodi dan dosen apabila sudah memiliki akun yang sudah terdaftar pada sistem oleh admin. Ketua Prodi dapat menambahkan jadwal rapat dengan memilih menu jadwal rapat, kemudian melakukan pengisian pada form jadwal rapat meliputi nama rapat, waktu rapat, hari/tanggal rapat, tempat, pimpinan rapat, notulis rapat, peserta rapat, topik rapat, file rapat. Jadwal rapat yang telah ditambahkan oleh ketua prodi akan dikirim kepada dosen, kemudian dosen akan menerima notifikasi undangan rapat. Dosen yang menerima undangan rapat akan melihat undangan rapat pada menu *dashboard,* kemudian jika dosen menerima undangan rapat maka pada menu jadwal rapat akan menampilkan jadwal rapat terbaru. Jika Notulis tidak dapat menghadiri rapat, maka akan muncul notifikasi kepada ketua prodi bahwa dosen tidak dapat menghadiri rapat. Notulis akan membuat hasil rapat dan absensi peserta rapat pada menu hasil rapat. Notulis akan mengirimkan hasil rapat kepada ketua prodi untuk mendapatkan *approve* terhadap hasil rapat. Hasil rapat yang telah di*approve* oleh ketua prodi akan disimpan oleh notulis kedalam menu hasil rapat sebagai dokumentasi hasil rapat. Berikut gambar flowchart dari proses manajemen rapat pada Aplikasi Manajemen Rapat berbasis *Website.*



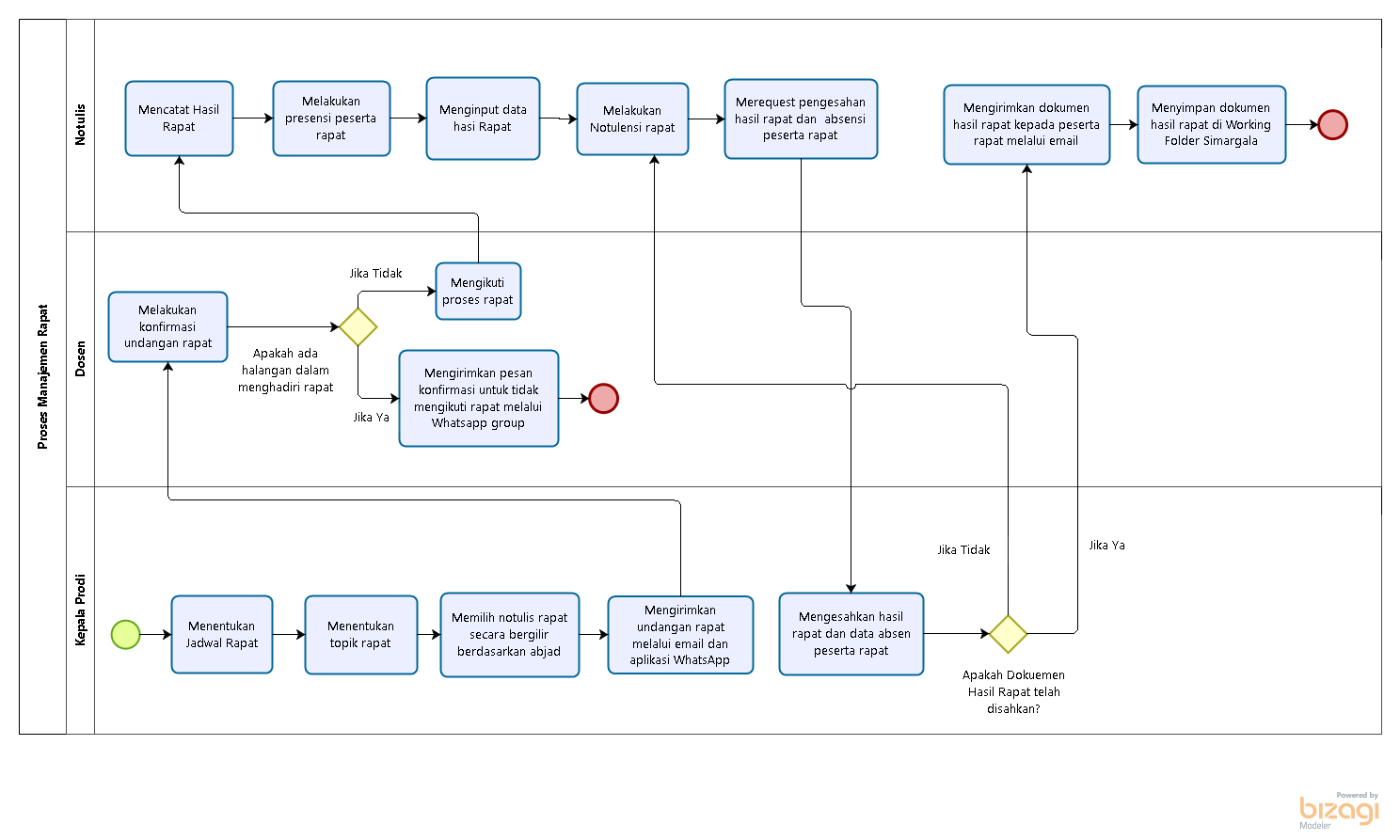
* 1. Analisis Rapat

Rapat di kampus Institut Teknologi Del merupakan salah satu kegiatan formal yang dihadiri oleh ketua prodi dan dosen yang akan merundingkan suatu masalah atau kepentingan bersama untuk memberikan penjelasan, serta menyelesaikan suatu masalah, dengan tujuan memperoleh suatu hasil yang disepakati bersama. Ketua prodi akan mengadakan rapat dengan membuat jadwal rapat dan mengirimkan undangan rapat terlebih dahulu kepada peserta rapat melalui email. Dosen yang menerima undangan rapat akan menghadiri rapat sesuai dengan jadwal yang ditentukan. Ketua prodi yang menjadi ketua rapat akan memilih dosen menjadi notulis untuk rapat secara bergilir berdasarkan abjad. Dosen yang terpilih menjadi notulis akan mendata keperluan rapat sebagai hasil rapat dari rapat. Cara kerjanya yaitu pada setiap rapat diadakan, maka dosen akan mengisi form notulen yaitu dokumen *template* MoM.

* 1. Proses Bisnis *Current System*

Proses bisnis manajemen rapat yang berjalan saat ini yaitu ketua prodi mengadakan rapat dengan menentukan jadwal rapat. Ketua prodi mengirimkan undangan rapat secara *online* kepada dosen melalui *email*. Ketua prodi akan memilih dosen secara bergilir berdasarkan abjad menjadi notulis. Notulis yang terpilih akan mencatat keperluan data rapat yang sedang berlansung, meliputi nama rapat, hari/tanggal, waktu, tempat, pimpinana rapat, notulis rapat, absensi peserta rapat, dan agenda rapat. Notulis menambahkan absensi peserta rapat dengan cara melihat peserta rapat yang hadir dan yang tidak hadir ke dalam file. Hasil rapat yang telah dicatat ke dokumen *template* MoM akan disimpan sebagai bentuk laporan pertanggungjawaban notulis. Dalam proses pendataan hasil rapat dan absensi peserta rapat terjadi ketidakefisienan karena notulis akan membutuhkan waktu yang cukup lama.

Pada bagian ini dijelaskan mengenai BPMN (*Business Process Modelling Notation*), *User* dan *service time* dari proses pendataan peserta rapat yang hadir, mencatat data rapat hingga melakukan proses penyimpanan dokumen rapat.



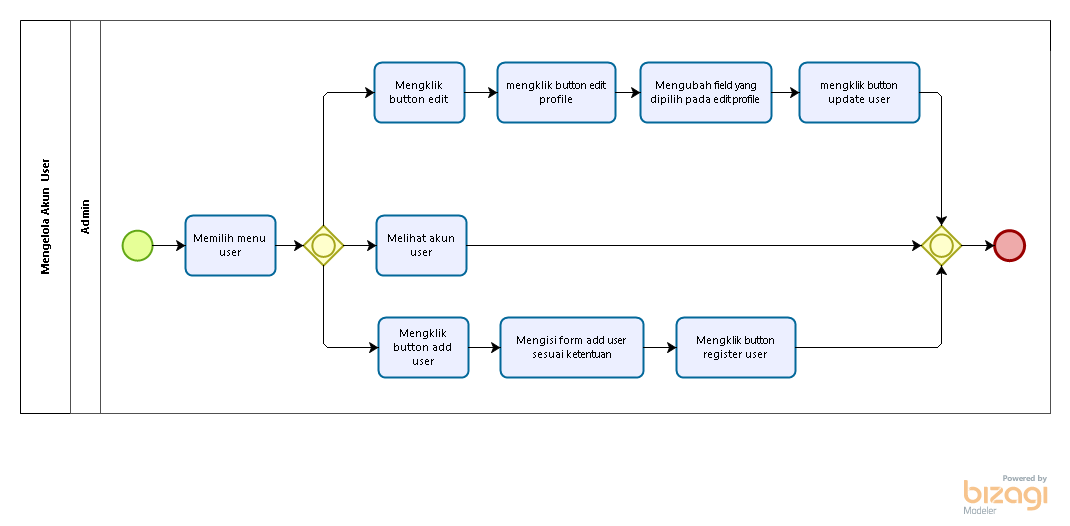
* 1. Proses Bisnis *Target System*

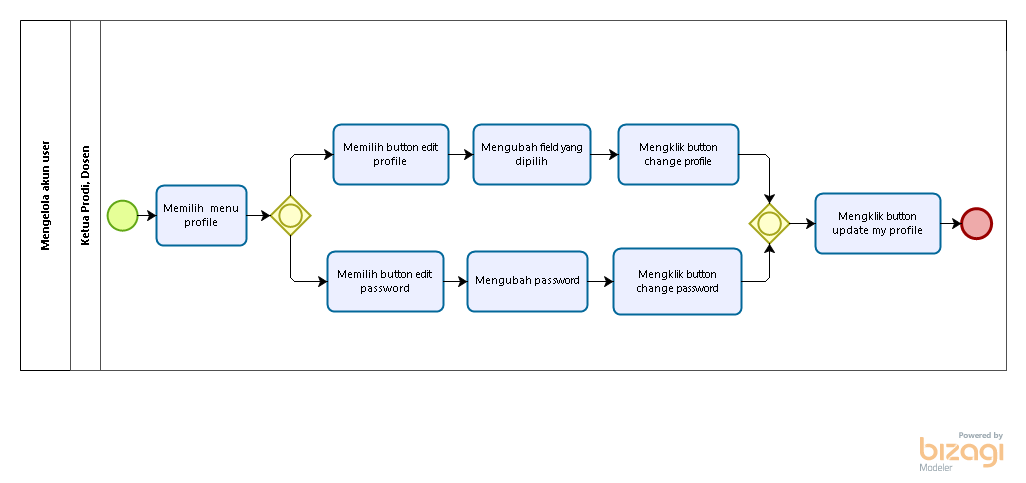
Aplikasi Manajemen Rapat berbasis *Website* merupakan aplikasi yang digunakan oleh prodi D3 Teknologi Informasi untuk memanajemen rapat. Pada sistem ini terdapat 3 *user roles* yaitu admin, ketua prodi dan dosen.

Admin dalam Aplikasi Manajemen Rapat adalah BAAK yang bertugas untuk mengelola akun *user* yaitu menambah akun *user*, melihat akun *user*, dan mengubah akun *user.* Admin akan mendaftarkan *user* kedalam sistem dengan menggunakan fungsi menambahkan akun *user.* Hal ini bertujuan agar *user* dapat mengakses sistem. Ketua Prodi akan membuat jadwal rapat dan mengirim undangan rapat melalui sistem yang akan dikirim ke akun masing-masing dosen dan melalui *email*. Ketua prodi akan memilih dosen yang berperan sebagai notulis secara bergilir berdasarkan abjad. Dosen yang terpilih menjadi notulis untuk rapat tidak boleh menjadi notulis lagi dirapat selanjutnya. Notulis memiliki tugas yaitu mengelola hasil rapat dan mengelola absensi peserta rapat. Dengan adanya fungsi mengelola hasil rapat dan mengelola absensi peserta rapat akan memudahkan notulis dalam mencatat hasil rapat dan absensi peserta rapat. Dalam *target system* disediakan fungsi mencari untuk mempermudah ketua prodi dan dosen dalam mencari data. Notulis dimudahkan meng*export* hasil rapat kedalam bentuk file .pdf, excel, dan csv. Proses bisnis dari target sistem yang dibangun dapat dilihat dalam sub bab di bawah.

* + 1. Proses Bisnis Mengelola Akun *User*

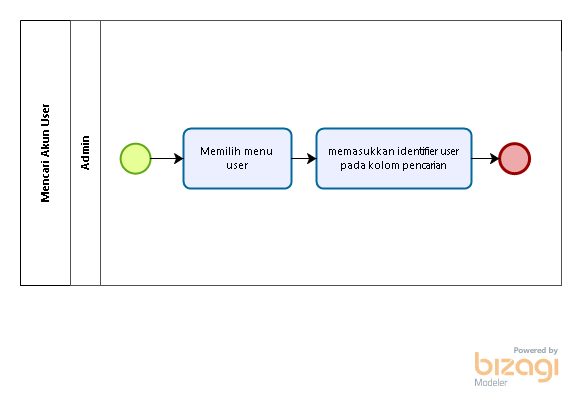
Aplikasi Manajemen Rapat berbasis *Website* memiliki fungsi mengelola akun *user* yang dilakukan oleh admin, ketua prodi, dan dosen. Fungsi mengelola akun *user* terdiri dari 3 fungsi yaitu, menambahkan akun *user,* melihat akun *user,* dan mengubah akun *user.* Proses bisnis menambahkan akun *user* dan melihat akun *user* hanya dapat dilakukan oleh admin. Proses bisnis menambahkan akun *user* oleh admin, dimulai dengan memilih menu *user* kemudian menambahkan akun *user.* Proses bisnis melihat akun *user* oleh admin, dimulai dengan menambahkan akun *user* kemudian melihat list *user* yang telah terdaftar. Proses mengubah akun *user* pada *role* admin dimulai dengan memilih menu *user,* kemudian memilih *user* untuk mengubah data *user.* Ketua Prodi dan dosen dapat mengubah akun *user* dengan memilih menu *profile*. Ketua prodi dan dosen dapat mengubah *profile* dan mengubah *password.* Ketua prodi dan dosen mengubah akun *user* dengan memilih *button edit profile* kemudian memilih *field* data yang akan diubah sesuai ketentuan. Ketua prodi dan dosen mengubah *password* dengan memilih *button edit password.* Proses bisnis mengelola akun *user* dapat dilihat pada gambar.





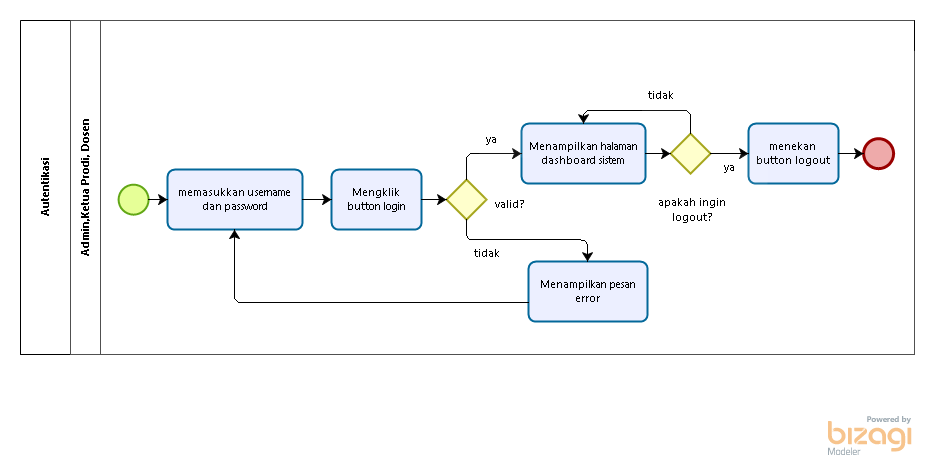
* + 1. Proses Bisnis Mencari Akun *User*

Aplikasi manajemen rapat memiliki fungsi mencari akun *user*, dimana fungsi ini dapat dilakukan oleh admin. Proses bisnis mencari akun *user* dimulai dengan admin memilih menu *user*, kemudian memasukkan *identifier* pada kolom pencarian. Proses bisnis mencari *user* dapat dilihat pada gambar.

****

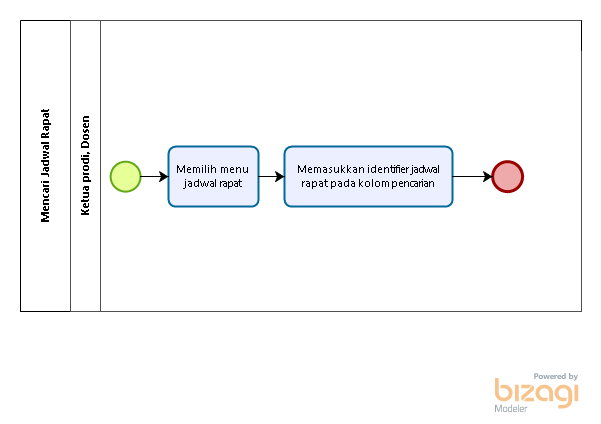
* + 1. Proses Bisnis Autentikasi

Aplikasi manajemen rapat memiliki fungsi Autentikasi, dimana fungsi ini dapat dilakukan oleh admin, ketua prodi, dan dosen. Admin, ketua prodi dan dosen melakukan *login* menggunakan *username* dan *password* *default* yang telah didaftarkan sebelumnya oleh admin ke dalam sistem, agar dapat mengakses aplikasi. Admin, ketua prodi, dan dosen mengisi *username* dan *password* sesuai dengan akun masing-masing. Proses bisnis *logout* pada admin, ketua prodi dan dosen dengan memilih *button logout* pada profile. Proses bisnis Autentikasi dapat dilihat pada Gambar.

****

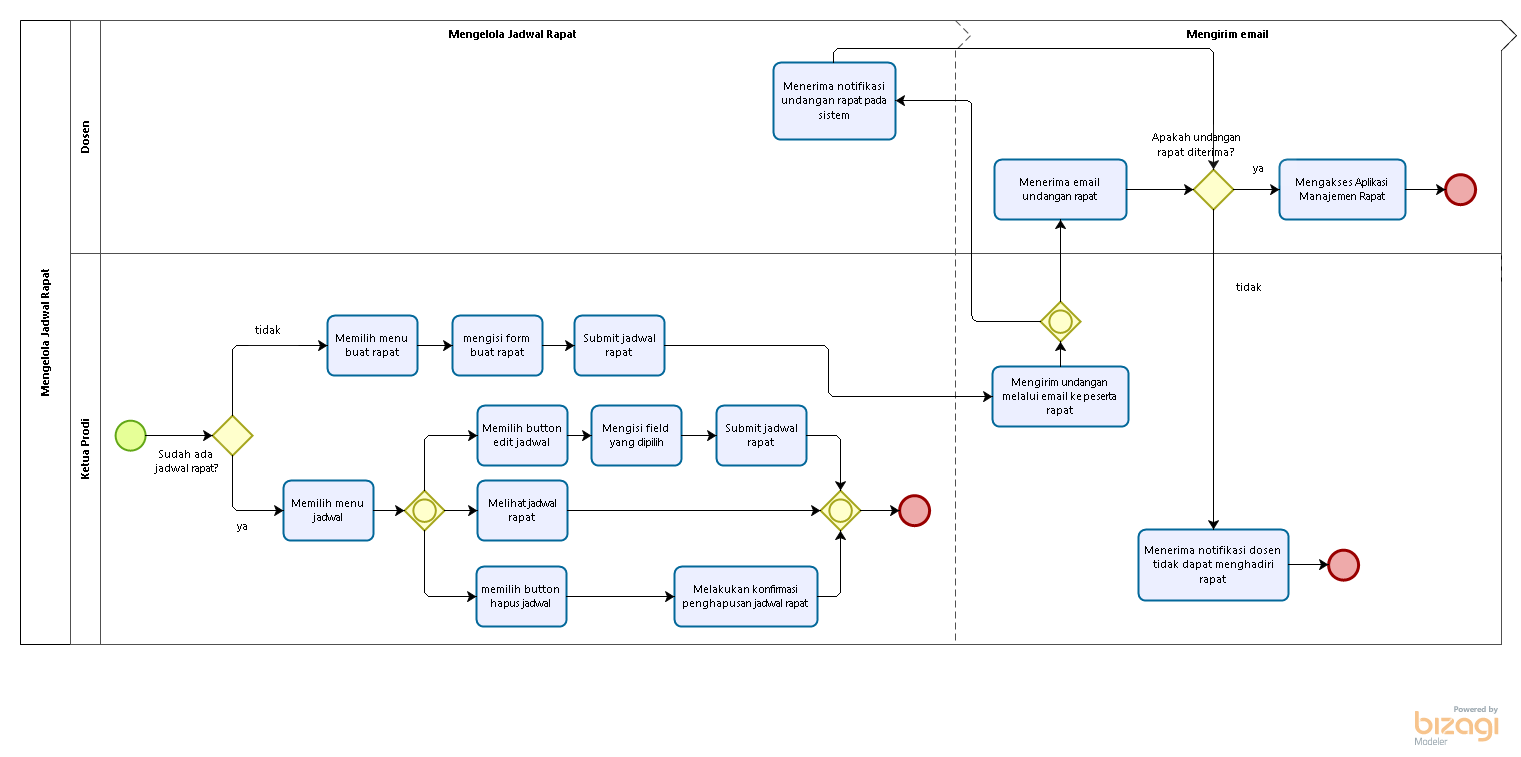
* + 1. Prose Bisnis Mencari Jadwal Rapat

Aplikasi manajemen rapat memiliki fungsi mencari jadwal rapat, dimana fungsi ini dapat dilakukan oleh ketua prodi dan dosen. Proses bisnis mencari jadwal rapat dimulai dengan ketua prodi dan dosen memilih menu jadwal, kemudian memasukkan *identifier* pada kolom pencarian. Proses bisnis mencari jadwal rapat dapat dilihat pada gambar.

****

* + 1. Proses Bisnis Mengelola Jadwal Rapat

Aplikasi manajemen rapat memiliki fungsi mengelola jadwal rapat, dimana fungsi ini dilakukan oleh ketua prodi dan dosen. Proses bisnis mengelola jadwal rapat yang dapat dilakukan oleh ketua prodi yaitu menambahkan jadwal rapat, melihat jadwal, mengubah jadwal, menghapus jadwal rapat dan mengirimkan jadwal rapat melalui email pada menu jadwal. Dosen dapat melihat jadwal rapat yang telah dikirimkan oleh ketua prodi melalui *email* dan notifikasi pada aplikasi. Dosen dapat melihat jadwal yang telah ditambahkan oleh ketua prodi pada aplikasi melalui menu jadwal. Dosen yang tidak dapat menghadiri rapat dapat memberikan konfirmasi untuk tidak dapat menghadiri rapat dengan memilih *button reject.* Ketua prodi menerima notifikasi dari dosen pada sistem, bahwa dosen tidak dapat menghadiri rapat. Bisnis proses mengelola jadwal rapat dapat dilihat pada gambar.



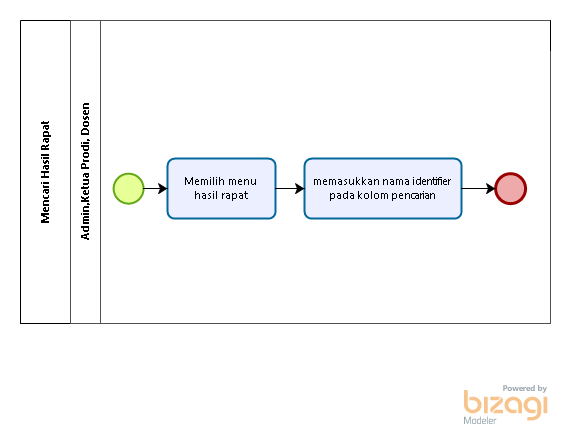
* + 1. Proses Bisnis Mengelola Hasil Rapat

Aplikasi manajemen rapat memiliki fungsi mengelola hasil rapat, dimana fungsi ini dapat dilakukan oleh ketua prodi dan dosen. Proses bisnis mengelola hasil rapat menjelaskan bahwa dosen sebagai notulis dapat menambahkan, melihat dan mengubah hasil rapat pada menu hasil rapat dan absensi peserta rapat. Dosen menambahkan absensi peserta rapat dengan memilih terlebih dahulu rapat, kemudian memilih *button* absensi untuk menambahkan data absensi. Dosen sebagai peserta rapat dapat melihat absensi peserta rapat dengan mengklik *button* anggota rapat. Dosen mengubah absensi peserta rapat dengan mengklik *button* absensi, sistem akan menampilkan daftar peserta rapat. Absensi peserta rapat yang telah di­*update* dapat dilihat pada *button* anggota rapat. Bisnis proses mengelola absensi peserta rapat dapat dilihat pada bisnis proses mengelola hasil rapat.

Ketua prodi akan melihat notifikasi hasil rapat yang telah di-*upload* oleh dosen, jika ketua prodi menyetujui hasil rapat maka dosen akan mengirimkan hasil rapatke menu hasil rapat, jika hasil rapat ditolak akan diperbaharui oleh dosen. Ketua prodi dan dosen sebagai peserta rapat hanya dapat melihat hasil rapat pada menu hasil rapat. Bisnis proses mengelola hasil rapat terdapat pada gambar.

* + 1. Proses Bisnis Mencari Hasil Rapat

Aplikasi manajemen rapat memiliki fungsi mencari hasil rapat, dimana fungsi ini dapat dilakukan oleh ketua prodi dan dosen. Proses bisnis mencari hasil rapat dimulai dengan ketua prodi dan dosen memilih menu hasil rapat, kemudian memasukkan *identifier* pada kolom pencarian. Proses bisnis mencari hasil rapat dapat dilihat pada gambar.

****

* 1. Modul Aplikasi Manajemen Rapat

Pada sub bab ini dijelaskan mengenai modul pembangunan Aplikasi Manajemen Rapat. Aplikasi manajemen rapat dibangun dengan menggunakan framework Laravel dan bahasa pemrograman PHP yang berbasis *website*. Pada Tabel 1 diberikan pembagian fungsi berdasarkan modul yang dikembangkan oleh tim pengembang.

**Tabel 1 Daftar Modul Aplikasi Manajemen Rapat**

| **Nomor** | **Daftar Modul** | **Pengembang** |
| --- | --- | --- |
| 1 | Fungsi menambahkan akun *user* | Veronika Oktafia Marpaung |
| 2 | Fungsi mengubah akun *user* | Veronika Oktafia Marpaung |
| 3 | Fungsi melihat akun *user* | Teresha Jesika Tampubolon |
| 4 | Fungsi mencari akun *user* | Teresha Jesika Tampubolon |
| 5 | Fungsi mencari jadwal rapat | Fedrick Sulaiman Siagian |
| 6 | Fungsi autentikasi | Fedrick Sulaiman Siagian |
| 7 | Fungsi menambahkan jadwal rapat | Fedrick Sulaiman Siagian |
| 8 | Fungsi mengubah jadwal rapat | Veronika Oktafia Marpaung |
| 9 | Fungsi menghapus jadwal rapat | Teresha Jesika Tampubolon |
| 10 | Fungsi melihat jadwal rapat | Fedrick Sulaiman Siagian |
| 11 | Fungsi menambahkan hasil rapat | Fedrick Sulaiman Siagian |
| 12 | Fungsi melihat hasil rapat | Fedrick Sulaiman Siagian |
| 13 | Fungsi mengubah hasil rapat | Fedrick Sulaiman Siagian |
| 14 | Fungsi mencari hasil rapat | Fedrick Sulaiman Siagian |
| 15 | Fungsi menambahkan absensi peserta rapat | Fedrick Sulaiman Siagian |
| 16 | Fungsi melihat absensi peserta rapat | Fedrick Sulaiman Siagian |
| 17 | Fungsi mengubah absensi peserta rapat | Fedrick Sulaiman Siagian |
| 18 | Fungsi notifikasi | Fedrick Sulaiman Siagian |
| 19 | Fungsi mengirim email | Fedrick Sulaiman Siagian |

Pada Tabel 2 diberikan pembagian fungsi berdasarkan modul yang dikembangkan oleh tim pengembang.

**Tabel 2 Fungsi Utama Aplikasi dan Deskripsi**

| **Nomor**  **Fungsi** | **Nama Fungsi** | **Deskripsi** |
| --- | --- | --- |
| FR-01 | Fungsi menambahkan akun *user* | Fungsi menambahkan akun *user* dapat dilakukan oleh admin. Fungsi ini digunakan untuk menambahkan data *user* yaitu ketua prodi dan dosen ke sistem. Data *user* yang akan ditambahkan terdiri dari nama, *email,* dan *password.* |
| FR-02 | Fungsi mengubah akun *user* | Fungsi mengubah akun *user* merupakan fungsi yang dilakukan oleh admin, ketua prodi dan dosen, yaitu mengubah nama, *email,* foto dan *password.* Ketua Prodi dan dosenakan didaftarkan oleh admin dengan membuat *default account,* dimana akun ini terdiri dari nama, *email,* foto dan *password.* |
| FR-03 | Fungsi melihat akun *user* | Fungsi melihat akun *user* merupakan fungsi yang dilakukan oleh admin untuk melihat akun *user* yang terdaftar didalam sistem. Fungsi ini digunakan untuk melihat detail data dari *user.* |
| FR-04 | Fungsi mencari akun *user* | Fungsi mencari akun *user* merupakan fungsi yang dilakukan oleh admin, untuk mencari data *user.* Fungsi ini dapat digunakan oleh admin didalam menu user, pada kolom pencarian admin memasukkan *identifier* untuk mencari data *user* sesuai dengan kebutuhan. |
| FR-05 | Fungsi mencari jadwal rapat | Fungsi mencari jadwal rapat merupakan fungsi yang digunakan untuk mencari jadwal rapat pada kolom pencarian jadwal rapat dengan menginput keyword sesuai dengan kebutuhan. Fungsi ini digunakan oleh ketua prodi dan dosen yang terdapat pada menu Jadwal. |
| FR-06 | Fungsi autentikasi | Fungsi autentikasi merupakan fungsi yang terdiri dari login dan logout yang dapat digunakan oleh admin, ketua prodi dan dosen. Fungsi login digunakan untuk dapat mengakses sistem. Admin, ketua prodi dan dosen akan meng*input* username dan password pada form login sistem sesuai dengan ketentuan. Fungsi logout digunakan untuk dapat keluar dari sistem dengan memilih *button logout* pada profile. |
| FR-07 | Fungsi menambahkan jadwal rapat | Fungsi menambahkan jadwal rapat dilakukan oleh ketua prodi untuk menambahkan jadwal rapat baru. |
| FR-08 | Fungsi mengubah jadwal rapat | Fungsi mengubah jadwal rapat digunakan oleh ketua prodi untuk mengubah jadwal rapat apabila terdapat perubahan pada jadwal yang telah dibuat sebelumnya. |
| FR-09 | Fungsi menghapus jadwal rapat | Fungsi menghapus jadwal rapat digunakan oleh ketua prodi untuk menghapus jadwal rapat apabila terdapat jadwal rapat yang tidak perlu untuk dilaksanakan. |
| FR-10 | Fungsi melihat jadwal rapat | Fungsi melihat jadwal rapat dilakukan oleh ketua prodi dan dosen untuk melihat jadwal rapat yang telah diupload oleh Ketua Prodi. |
| FR-11 | Fungsi menambahkan hasil rapat | Fungsi menambahkan hasil rapat dapat dilakukan oleh dosen yang terpilih menjadi notulis untuk menambahkan data hasil rapat meliputi nama rapat, keterangan waktu, hari/tanggal, tempat, pimpinan rapat, notulis rapat, peserta rapat, agenda rapat, notulis rapat, dokumentasi, dan tindak lanjut. |
| FR-12 | Fungsi melihat hasil rapat | Fungsi melihat hasil rapat digunakan untuk melihat hasil rapat yang telah di*upload* oleh dosen yang terpilih menjadi notulis kedalam sistem, agar peserta rapat dapat melihat detail hasil rapat yang telah selesai. Fungsi ini dapat dilakukan oleh ketua prodi dan dosen. |
| FR-13 | Fungsi mengubah hasil rapat | Fungsi mengubah hasil rapat digunakan untuk melakukan perubahaan pada data yang tidak sesuai dengan hasil rapat. |
| FR-14 | Fungsi mencari hasil rapat | Fungsi mencari hasil rapat merupakan fungsi yang digunakan untuk mencari hasil rapat pada menu hasil rapat. Fungsi mencari hasil rapat dapat digunakan oleh ketua prodi dan dosen agar lebih cepat menemukan hasil rapat yang dibutuhkan, yaitu dengan memasukkan *keyword* pada kolom pencarian. |
| FR-15 | Fungsi menambahkan absensi peserta rapat | Fungsi menambahkan absensi peserta rapat dilakukan oleh dosen yang terpilih menjadi notulis, yang digunakan untuk menambahkan absensi. Notulis menambahkan daftar peserta rapat yang hadir, pada halaman absensi peserta rapat. |
| FR-16 | Fungsi melihat absensi peserta rapat | Fungsi melihat absensi peserta rapat dilakukan oleh dosen yang terpilih menjadi notulis, yang digunakan untuk melihat daftar peserta rapat yang hadir pada saat rapat. |
| FR-17 | Fungsi mengubah absensi peserta rapat | Fungsi mengubah absensi peserta rapat dilakukan oleh dosen yang terpilih menjadi notulis, yang digunakan untuk melakukan perubahanpada data absensi peserta rapat yang tidak sesuai. |
| FR-18 | Fungsi notifikasi | Fungsi notifikasi merupakan fungsi yang memberikan pemberitahuan dan dilakukan oleh ketua prodi dan dosen. Fungsi notifikasi digunakan untuk memberikan pemberitahuan jadwal rapat kepada dosen, pemberitahuan dosen tidak dapat menghadiri rapat kepada ketua prodi, *request* pengesahan hasil rapat kepada ketua prodi, dan pemberitahuan hasil rapat telah berhasil di *upload* kepada dosen*.* |
| FR-19 | Fungsi mengirim email | Fungsi mengirim *email* merupakan fungsi untuk mengirimkan *email* undangan rapat yang dilakukan oleh ketua prodi. Fungsi mengirim email bertujuan agar dosen dapat mengetahui undangan rapat yang telah dikirim oleh ketua prodi apabila dosen dalam kondisi sedang tidak mengakses Aplikasi Manajemen Rapat. |

* 1. User Characteristic

Pada bagian ini menjelaskan karakteristik dari pengguna Aplikasi Manajemen Rapat berbasis *web*.

* + 1. User Group

*User* group pada Tabel 3 merupakan pengelompokan pengguna dalam mengakses aplikasi.

**Tabel 3 User Group**

|  |  |  |
| --- | --- | --- |
| ***User* Group/Role** | **Kepentingan Akses** | **Hak Akses** |
| *Admin* | Mengelola data pada sistem dan mengelola akun *user.* | *User* yang berhak mengelola data dan sistem, seperti menyetujui dan menghapus data pada sistem. |
| Ketua Prodi | Autentikasi, mengelola jadwal rapat, mencari jadwal rapat, mencari dokumen hasil rapat dan mengubah *profile.* | *User* yang berhak mengirim undangan rapat dan mengelola jadwal rapat seperti menentukan jadwal rapat yang akan dilaksanakan dan memberikan pengesahan terhadap hasil rapat yang di *upload* oleh notulis. |
| Dosen | Autentikasi, mengelola hasil rapat, mengelola absensi peserta rapat, mengubah *profile,* mencari jadwal rapat, mencari hasil rapat, | *User* yang berhak mengelola hasil rapat selama rapat berlansung, *user* dapat mencari dokumen hasil rapat yang dibutuhkan, mengelolaabsensi peserta rapat, dapat melakukan perubahan pada profile yang sudah didaftarkan oleh *admin* sebelumnya. |

#### *User*-Group-1

Description of *User* : *User* memiliki hak akses untuk mengelola sistem dan mengelola akun *User.*

Role : Admin

Prerequisite : Admin telah *login* ke sistem.

Task description : Admin mampu mengelola akun *User*, mengelola data pada sistem.

#### *User*-Group-2

Description of *User* : *User* berperan mengatur jadwal rapat, mengirimkan undangan rapat, mengesahkan hasil rapat dan mencari hasil rapat pada aplikasi.

Role : Ketua Prodi

Prerequisite : Ketua Prodi telah *login* ke sistem.

Task description :

* Ketua Prodi mampu menambah jadwal rapat, melihat jadwal rapat, mengubah jadwal rapat dan menghapus jadwal rapat, yang diolah pada aplikasi.
* Mengirimkan undangan rapat kepada dosen.
* Mencari jadwal rapat.
* Mencari hasil rapat pada aplikasi.
* Mengubah *profile*.

#### *User*-Group-3

Description of *User* : *User* berperan mengelola hasil rapat selama rapat berlansung, dan mengelola absensi peserta rapat.

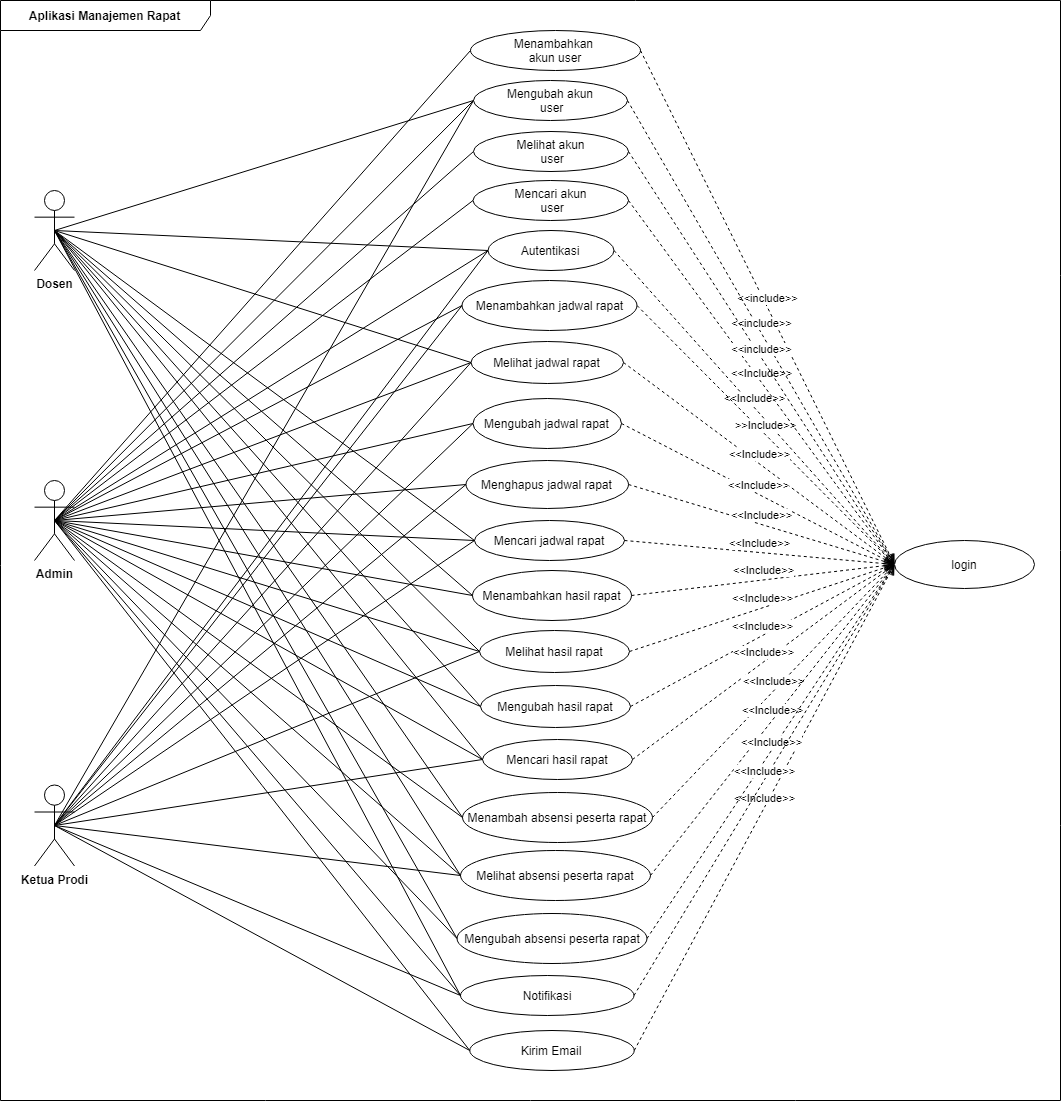
Role : Dosen

Prerequisite : Dosen telah *login* ke sistem.

Task description :

* Dosen mampu menambahkan hasil rapat, melihat hasil rapat dan mengubah hasil rapat.
* Dosen mampu menambahkan absensi peserta rapat, melihat absensi peserta rapat, dan mengubah absensi peserta rapat.
* Dosen mampu melakukan perubahan pada profile.
  1. Use Case Diagram Pada Aplikasi Manajemen Rapat

Pada subbab ini akan dijelaskan mengenai *use case* diagram Aplikasi Manajemen Rapat. *Use case* merupakan hubungan antara aktor dengan sistem. Hubungan antara aktor dengan sistem dalam aplikasi manajemen rapat dapat dilihat pada Gambar.



Dalam penelitian ini, yang menjadi aktor adalah Ketua prodi, dosen sebagai pengguna biasa dan BAAK sebagai admin. Pada Gambar 13, dapat dilihat kegiatan yang dilakukan oleh ketua prodi menggunakan aplikasi manajemen rapat untuk mengelola jadwal rapat, mencari jadwal rapat, mengirimkan undangan rapat, dan mencari dokumen hasil rapat. Dosen yang terpilih menjadi notulis menggunakan aplikasi manajemen rapat untuk mengelola hasil rapat, mengelola absensi peserta rapat, mencari hasil rapat, dan mencari jadwal rapat. Admin akan mengelola akun *user* dan mengelola seluruh data pada sistem*.*

* 1. Use Case Scenario

Pada subbab ini dijelaskan *scenario* dari *use case* diagram Aplikasi Manajemen Rapat yang diterapkan pada desain aplikasi. *Use case scenario* merupakan *scenario* atau alur dari suatu proses *use case* dari sisi *user* dan sistem.

### *Use Case Scenario* FR-01

Pada bagian ini dipaparkan *use case scenario* dari fungsi menambahkan akun *user* oleh admin.

|  |  |  |
| --- | --- | --- |
| Use Case Name | Menambahkan akun *user* | |
| Use Case Description | Use case ini untuk menambahkan data diri agar terdaftar di dalam sistem. | |
| Actor | Admin | |
| Precondition | PRE-1. Admin sudah mengakses dan berhasil *login* kedalam sistem. | |
| Primary Flow of Events | *User* Action | System Response |
| 1. Admin membuka Aplikasi Manajemen Rapat |  |
| 1. Admin memilih menu users |  |
|  | 1. Sistem menampilkan halaman “List User” yang sudah memiliki akun |
| 1. *User* meng-klik button “Add User” |  |
|  | 1. Sistem menampilkan form add user |
| 1. *User* mengisi data pada field sesuai ketentuan dan memilih role *user* yang akan ditambahkan |  |
| 1. *User* mengklik *button* “Register” |  |
|  | 1. Sistem akan menyimpan data *user* kedalam database sistem. |
|  | 1. Sistem akan menampilkan halaman dashboard. |
| Error Flow of Events | User Action | Sistem Response |
| 6a. Admin tidak mengisi semua *field* pada form add user yang kosong. | 6b. Sistem menampilkan required bahwa data harus diisi sesuai ketentuan. |
| Post Condition | POST-1. Admin berhasil menambahkan akun *user* baru. | |

### *Use Case Scenario* FR-02

Pada bagian ini dipaparkan *use case scenario* dari fungsi mengubah akun *user* oleh admin, ketua prodi dan dosen.

#### *Use Case Scenario* Mengubah akun user oleh Admin

Pada bagian ini dipaparkan *use case scenario* dari fungsi mengubah akun *user* oleh admin.

|  |  |  |
| --- | --- | --- |
| Use Case Name | Mengubah akun *user* | |
| Use Case Description | Use case ini untuk mengubah data diri *user.* | |
| Actor | Admin, ketua prodi dan dosen. | |
| Precondition | PRE-1. Admin, ketua prodi dan dosensudah berhasil *login* kedalam sistem | |
| Primary Flow of Events | *User* Action | System Response |
| 1. Admin memilih menu users |  |
|  | 1. Sistem menampilkan halaman “List User” yang sudah memiliki akun |
| 1. Admin memilih akun *user* yang akan diubah. |  |
|  | 1. Sistem menampilkan halaman “User Detail” dari akun *user* yang dipilih. |
| 1. Admin mengklik *button* edit |  |
|  | 1. Sistem menampilkan halaman “Edit Profile” |
| 1. Admin mengisi *field* yang akan diubah |  |
| 1. Admin mengklik *button save* |  |
| Admin mengklik *button* Edit Password untuk mengubah password *user* |  |
|  | 1. Sistem menampilkan halaman edit password |
| Admin mengubah pada *field* password |  |
| Admin mengklik button change password |  |
|  | 1. Sistem menampilkan halaman Edit Profile |
| Error Flow of Events | *User* Action | System Response |
| 7a. Admin tidak mengisi semua *field*  pada halaman edit profile | 7b. Sistem menampilkan required bahwa semua data harus diisi sesuai ketentuan. |
| 11a. Admin tidak mengisi semua *field* password pada halaman edit password | 11b. Sistem menampilkan required bahwa semua data harus diisi sesuai ketentuan. |
| Post Condition | POST-1. Admin berhasil mengubah akun *user.* | |

#### *Use Case Scenario* Mengubah akun *user* oleh Ketua prodi dan dosen.

Pada bagian ini dipaparkan *use case scenario* dari fungsi mengubah akun *user* oleh ketua prodi dan dosen.

|  |  |  |
| --- | --- | --- |
| Use Case Name | Mengubah akun *user* | |
| Use Case Description | Use case ini untuk mengubah data diri *user.* | |
| Actor | Ketua prodi dan dosen. | |
| Precondition | PRE-1. Ketua prodi dan dosensudah berhasil *login* kedalam sistem | |
| Primary Flow of Events | *User* Action | System Response |
| 1. Ketua prodi dan dosen mengklik profile. |  |
|  | 1. Sistem menampilkan 2 button yaitu edit profile dan *logout*. |
| 1. Ketua prodi dan dosen mengklik button edit profile |  |
|  | 1. Sistem menampilkan halaman “Edit Profile” |
| 1. Ketua prodi dan dosen mengisi semua *field* yang akan diubah |  |
| 1. Ketua prodi dan dosen mengklik *button save* |  |
| Ketua prodi dan dosen mengklik *button* Edit Password untuk mengubah password *user* |  |
|  | 1. Sistem menampilkan halaman edit password |
| Ketua prodi dan dosen mengubah semua *field* password |  |
| Ketua prodi dan dosen mengklik button change password |  |
|  | 1. Sistem menampilkan halaman Edit Profile |
| Error Flow of Events | *User* Action | System Response |
| 5a. Ketua prodi dan dosen tidak mengisi semua *field*  pada halaman edit profile | 5b. Sistem menampilkan required bahwa semua data harus diisi sesuai ketentuan. |
| 9a. Ketua prodi dan dosen tidak mengisi semua *field* password pada halaman edit password | 9b. Sistem menampilkan required bahwa semua data harus diisi sesuai ketentuan. |
| Post Condition | POST-1. Ketua prodi dan dosen berhasil mengubah akun. | |

### *Use Case Scenario* FR-03

Pada bagian ini dipaparkan *use case scenario* dari fungsi melihat akun *user* oleh admin.

|  |  |  |
| --- | --- | --- |
| Use Case Name | Melihat akun *user* | |
| Use Case Description | Use case ini untuk melihat daftar dari akun *user* yang telah ditambahkan oleh admin. | |
| Actor | Admin | |
| Precondition | PRE-1. Adminberhasil *login* kedalam sistem. | |
| Primary Flow of Events | *User* Action | System Response |
| 1. Admin memilih menu users |  |
|  | 1. Sistem menampilkan halaman “List User” yang sudah memiliki akun |
| 1. Admin melihat daftar list akun *user* yang sudah terdaftar |  |
| 1. Admin mengklik button view untuk melihat akun *user* |  |
|  | 1. Sistem menampilkan detail akun *user* |
| Error Flow of Events | *User* Action | System Response |
|  |  |
| Post Condition | POST-1. Admin berhasil melihat akun *user* | |

### *Use Case Scenario* FR-04

Pada bagian ini dipaparkan *use case scenario* dari fungsi mencari akun *user* oleh admin.

|  |  |  |
| --- | --- | --- |
| Use Case Name | Mencari akun *user* | |
| Use Case Description | Use case ini untuk mencari akun *user* yang dibutuhkan oleh admin. | |
| Actor | Admin | |
| Precondition | PRE-1. Adminsudah berhasil *login* kedalam sistem | |
| Primary Flow of Events | *User* Action | System Response |
| 1. Admin memilih menu users |  |
|  | 1. Sistem menampilkan halaman “List User” yang sudah memiliki akun |
| 1. Admin memasukkan keywordpada kolom pencarian |  |
|  | 1. Sistem menampilkan hasil pencarian data *user* |
| Error Flow of Events | *User* Action | Sistem Response |
| 3a. Admin mengganti keyword akun yang dicari | 3b. Sistem menampilkan alert akun tidak ditemukan |
| Post Condition | POST-1. Admin berhasil mencari data *user.* | |

### *Use Case Scenario* FR-05

Pada bagian ini dipaparkan *use case scenario* dari fungsi mencari jadwal rapatoleh ketua prodi dan dosen.

|  |  |  |
| --- | --- | --- |
| Use Case Name | Mencari jadwal rapat | |
| Use Case Description | Use case ini untuk mencari jadwal rapat yang telah ditambahkan oleh ketua prodi. | |
| Actor | Admin, ketua prodi, dan dosen. | |
| Precondition | PRE-1. Admin*,*ketua prodi, dan dosen sudah membuka sistem.  PRE-2. Ketua prodi sudah menambahkan jadwal rapat kedalam sistem. | |
| Primary Flow of Events | *User* Action | System Response |
| 1. Admin, ketua prodi dan dosen memilih menu jadwal |  |
|  | 1. Sistem menampilkan halaman “Rapat” yang terdiri dari rapat yang sudah ditambahkan. |
| 1. Admin, ketua prodi dan dosen memasukkan keywordpada kolom pencarian |  |
|  | 1. Sistem menampilkan data jadwal rapatsesuai yang di*input* |
| Error Flow of Events | *User* Action | System Response |
| 3a. Admin, ketua prodi dan dosen mengganti keyword jadwal rapat yang dicari | 3b. Sistem menampilkan alert jadwal rapat tidak ditemukan |
| Post Condition | POST-1. Admin, ketua prodi dan dosen berhasil mencari jadwal rapat dan menemukan jadwal rapat yang dicari. | |

### *Use Case Scenario* FR-06

Pada bagian ini dipaparkan *use case scenario* dari fungsi autentikasioleh admin, ketua prodi, dan dosen yang terdiri dari *login* dan *logout.*

#### *Use Case Scenario Login*

Use Case Scenario login yang dipakai dalam sistem ini dapat dilihat pada tabel di bawah ini.

|  |  |  |
| --- | --- | --- |
| Use Case Name | Login | |
| Use Case Description | Use case ini untuk mengakses sistem. | |
| Actor | Admin*,* ketua prodi dan dosen. | |
| Precondition | PRE-1. Admin*,* ketua prodi dan dosen belum *login* kedalam sistem. | |
| Primary Flow of Events | *User* Action | System Response |
| 1. Admin, ketua rapat dan dosen mengakses Aplikasi Manajemen Rapat |  |
|  | 1. Sistem menampilkan halaman form *login.* |
| 1. Admin, ketua prodi dan dosen memasukkan *email* dan *password* ke form login sesuai dengan ketentuan. |  |
|  | 1. Sistem menampilkan halaman dashboard aplikasi. |
| Error Flow of Events | *User* Action | System Response |
| 3a. Admin, ketua prodi dan dosen memasukkan kembali *email* dan *password* pada form *login* | 3b. Sistem menampilkan pesan error pada sistem. |
| Post Condition | POST-1. Admin, ketua prodi dan dosen berhasil login kedalam sistem. | |

#### *Use Case Scenario Logout*

Use Case Scenario *logout* yang dipakai dalam sistem ini dapat dilihat pada tabel di bawah ini

|  |  |  |
| --- | --- | --- |
| Use Case Name | Autentikasi | |
| Use Case Description | Use case ini untuk logout dari sistem | |
| Actor | Admin*,* ketua prodi, dan dosen | |
| Precondition | PRE-1. Admin*,* ketua prodi dan dosen berhasil *login* kedalam sistem. | |
| Primary Flow of Events | *User* Action | System Response |
| 1. Admin, ketua prodi dan dosen mengklik profile |  |
|  | 1. Sistem menampilkan 2 *button* yaitu profile dan *logout.* |
| 1. Admin, ketua prodi dan dosen mengklik *button logout.* |  |
|  | 1. Sistem menampilkan halaman form *login.* |
| Error Flow of Events | *User* Action | System Response |
|  |  |
| Post Condition | POST-1. Admin, ketua prodi dan dosen berhasil keluar dari sistem. | |

### *Use Case Scenario* FR-07

Pada bagian ini dipaparkan *use case scenario* dari fungsi menambahkan jadwal rapat oleh ketua prodi.

|  |  |  |
| --- | --- | --- |
| Use Case Name | Menambahkan jadwal rapat. | |
| Use Case Description | Use case ini untuk menambahkan jadwal rapat yang baru ke dalam sistem. | |
| Actor | Ketua prodi | |
| Precondition | PRE-1. Ketua prodisudah login kedalam sistem dan ingin menambahkan jadwal rapat. | |
| Primary Flow of Events | *User* Action | System Response |
| 1. Ketua prodi mengklik menu Buat Rapat. |  |
|  | 1. Sistem menampilkan halaman Buat Rapat |
| 1. Ketua prodi mengisi semua *field* pada form Buat Rapat |  |
| 1. Ketua prodi mengklik *button* Create Meeting |  |
|  | 1. Sistem menyimpan data jadwal rapat pada database sistem. |
|  | 1. Sistem mengirimkan undangan rapat melalui sistem dan *email.* |
|  | 1. Sistem menampilkan halaman “Rapat” yang berisikan jadwal rapat terbaru pada menu Jadwal. |
| Error Flow of Events | *User* Action | System Response |
| 3a. Ketua prodi tidak mengisi semua *field* pada form Buat Rapat. | 3b. Sistem akan menampilkan required bahwa semua *field* harus diisi. |
| Post Condition | POST-1. Ketua prodi berhasil menambahkan jadwal rapat. | |

### *Use Case Scenario* FR-08

Pada bagian ini dipaparkan *use case scenario* dari fungsi mengubah jadwal rapat oleh ketua prodi.

|  |  |  |
| --- | --- | --- |
| Use Case Name | Mengubah jadwal rapat | |
| Use Case Description | Use case ini untuk mengubah jadwal rapat | |
| Actor | Ketua prodi | |
| Precondition | PRE-1. Ketua prodi sudah membuka sistem dan memilih menu Jadwal untuk mengubah jadwal rapat. | |
| Primary Flow of Events | *User* Action | System Response |
| 1. Ketua prodi mengklik button edit pada jadwal rapat yang dipilih |  |
|  | 1. Sistem menampilkan halaman Edit Rapat sesuai dengan jadwal rapat yang dipilih. |
| 1. Ketua prodi mengisi *field* yang akan diubah. |  |
|  | 1. Sistem menyimpan data jadwal rapat pada database sistem. |
|  | 1. Sistem menampilkan halaman Rapat |
| Error Flow of Events | *User* Action | System Response |
| 3a. Ketua prodi tidak mengisi semua *field* pada form Rapat. | 3b. Sistem akan menampilkan required bahwa semua *field* harus diisi. |
| Post Condition | POST-1. Ketua prodi berhasil mengubah jadwal rapat. | |

### *Use Case Scenario* FR-09

Pada bagian ini dipaparkan *use case scenario* dari fungsi menghapus jadwal rapat oleh ketua prodi.

|  |  |  |
| --- | --- | --- |
| Use Case Name | Menghapusjadwal rapat | |
| Use Case Description | Use case ini untuk menghapus jadwal rapat | |
| Actor | Ketua prodi | |
| Precondition | PRE-1. Ketua prodisudah *login* kedalam sistem dan melihat daftar rapat namun ingin menghapus jadwal rapat yang tidak diperlukan. | |
| Primary Flow of Events | *User* Action | System Response |
| 1. Ketua prodi mengklik *button delete* pada salah satu jadwal rapat |  |
|  | 1. Sistem memberikan *alert delete* “Hapus jadwal rapat: nama rapat?” terhadap rapat yang akan dihapus. |
| 1. Ketua prodi mengklik *button delete.* |  |
|  | 1. Sistem menghapus jadwal rapat |
|  | 1. Sistem menampilkan daftar list jadwal rapat lainnya. |
| Error Flow of Events | *User* Action | System Response |
|  |  |
| Post Condition | POST-1. Ketua prodi berhasil menghapus jadwal rapat. | |

### *Use Case Scenario* FR-10

Pada bagian ini dipaparkan *use case scenario* dari fungsi melihat jadwal rapatoleh ketua prodi dan dosen.

|  |  |  |
| --- | --- | --- |
| Use Case Name | Melihat jadwal rapat | |
| Use Case Description | Use case ini untuk melihat jadwal rapat | |
| Actor | Ketua prodi dan dosen | |
| Precondition | PRE-1. Ketua prodi dan dosen sudah *login* kedalam sistem, dan sudah menerima notifikasi pemberitahuan undangan rapat. | |
| Primary Flow of Events | *User* Action | System Response |
| 1. Ketua prodi dan dosen mengklik menu Jadwal. |  |
|  | 1. Sistem menampilkan seluruh jadwal rapat. |
| 1. Ketua prodi dan dosen mengklik *button view* di salah satu jadwal rapat |  |
|  | 1. Sistem menampilkan detail dari jadwal rapat. |
| Error Flow of Events | *User* Action | System Response |
|  |  |
| Post Condition | POST-1. Ketua prodi dan dosen berhasil melihat jadwal rapat, dan berhasil melihat detail jadwal rapat yang dipilih. | |

### *Use Case Scenario* FR-11

Pada bagian ini dipaparkan *use case scenario* dari fungsi menambahkan hasil rapatoleh dosen.

|  |  |  |
| --- | --- | --- |
| Use Case Name | Menambahkan hasil rapat | |
| Use Case Description | Use case ini untuk menambahkan hasil rapat agar ketua prodi dan peserta rapat dapat melihat hasil rapat. | |
| Actor | Dosen | |
| Precondition | PRE-1. Dosen sudah mengakses/membuka sistem dan mengikuti rapat. | |
| Primary Flow of Events | *User* Action | System Response |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| Error Flow of Events | *User* Action | System Response |
|  |  |
| Post Condition | POST-1. *User* berhasil membuat akun baru dan mendapat *User*name dan password yang sudah terdaftar dalam sistem. | |

### *Use Case Scenario* FR-12

Pada bagian ini dipaparkan *use case scenario* dari fungsi melihat hasil rapatoleh ketua prodi dan dosen.

|  |  |  |
| --- | --- | --- |
| Use Case Name | Melihat hasil rapat | |
| Use Case Description | Use case ini untuk melihat hasil rapat yang telah di*upload* oleh dosen. | |
| Actor | Ketua prodi dan dosen | |
| Precondition | PRE-1. Ketua prodi dan dosen sudah berhasil *login* kedalam sistem | |
| Primary Flow of Events | *User* Action | System Response |
| 1. Ketua prodi dan dosen mengklik menu “Hasil Rapat” |  |
|  | 1. Sistem menampilkan halaman Rapat dan menampilkan daftar hasil rapat |
| 1. Ketua prodi dan dosen mengklik *button view* di salah satu hasil rapat |  |
|  | 1. Sistem menampilkan hasil pencarian dari hasil rapat pada halaman Detail Rapat |
| Error Flow of Events | *User* Action | System Response |
|  |  |
| Post Condition | POST-1. Ketua prodi dan dosen berhasil melihat hasil rapat yang telah di*upload* oleh dosen sebagai notulis. | |

### *Use Case Scenario* FR-13

Pada bagian ini dipaparkan *use case scenario* dari fungsi mengubah hasil rapatoleh dosen.

|  |  |  |
| --- | --- | --- |
| Use Case Name | Mengubah hasil rapat | |
| Use Case Description | Use case ini untuk mengubah hasil rapat | |
| Actor | Dosen | |
| Precondition | PRE-1. Dosenberhasil *login* kedalam sistem, dan melakukan notulensi hasil rapat didalam sistem. | |
| Primry Flow of Events | *User* Action | System Response |
| 1. Dosen mengklik menu Hasil Rapat. |  |
|  | 1. Sistem menampilkan daftar list hasil rapat yang dibuat oleh dosen. |
| 1. Dosen mengklik *button view* disalah satu hasil rapat yang akan diubah. |  |
|  | 1. Sistem menampilkan detail hasil rapat pada halaman Rapat. |
| 1. Dosen mengisi *field* yang akan diperbaharui dengan data yang baru. |  |
|  | 1. Sistem menyimpan data hasil rapat pada database sistem. |
|  | 1. Sistem menampilkan halaman detail rapat dari rapat yang sudah diperbaharui. |
| Error Flow of Events | *User* Action | System Response |
| 5a. Dosen tidak mengisi semua *field* yang akan diperbaharui dengan data yang baru. | 5b. Sistem akan menampilkan required bahwa semua *field* harus diisi. |
| Post Condition | POST-1. Dosen berhasil mengubah hasil rapat | |

### *Use Case Scenario* FR-14

Pada bagian ini dipaparkan *use case scenario* dari fungsi mencari hasil rapat oleh ketua prodi dan dosen.

|  |  |  |
| --- | --- | --- |
| Use Case Name | Mencari hasil rapat | |
| Use Case Description | Use case ini untuk mencari hasil rapat | |
| Actor | Ketua prodi dan dosen. | |
| Precondition | PRE-1. Ketua prodi dan dosen berhasil *login* kedalam sistem. | |
| Primary Flow of Events | *User* Action | System Response |
| 1. Admin, ketua prodi dan dosen memilih menu Hasil Rapat |  |
|  | 1. Sistem menampilkan halaman “Rapat” yang terdiri dari hasil rapat yang sudah ditambahkan. |
| 1. Ketua prodi dan dosen memasukkan keyword pada kolom pencarian hasil rapat |  |
|  | 1. Sistem menampilkan hasil pencarian hasil rapat |
| Error Flow of Events | *User* Action | System Response |
| 3a. User mengganti keyword hasil rapat yang dicari | 3b. Sistem menampilkan alert hasil rapat tidak ditemukan |
| Post Condition | POST-1. *User* berhasil membuat akun baru dan mendapat *User*name dan password yang sudah terdaftar dalam sistem. | |

### *Use Case Scenario* FR-15

Pada bagian ini dipaparkan *use case scenario* dari fungsi menambahkan absensi peserta rapat oleh dosen.

|  |  |  |
| --- | --- | --- |
| Use Case Name | Menambahkan absensi peserta rapat | |
| Use Case Description | Use case ini untuk menambahkan daftar hadir dari peserta rapat pada rapat. | |
| Actor | Dosen | |
| Precondition | PRE-1. Dosenberhasil *login* kedalam sistem  PRE-2. Dosen mengklik menu Jadwal Rapat dan melihat detail rapat tetapi belum melakukan absensi peserta rapat. | |
| Primary Flow of Events | *User* Action | System Response |
| 1. Dosen mengklik *button* Absensi. |  |
|  | 1. Sistem menampilkan halaman Absen dan daftar anggota rapat. |
| 1. Dosen men*checklist* anggota rapat yang hadir pada rapat. |  |
|  | 1. Sistem menyimpan data absensi peserta rapat yang hadir pada database sistem. |
| Error Flow of Events | *User* Action | System Response |
|  |  |
| Post Condition | POST-1. Dosen berhasil menambahkan absensi peserta rapat. | |

### *Use Case Scenario* FR-16

Pada bagian ini dipaparkan *use case scenario* dari fungsi melihat absensi peserta rapat oleh ketua prodi dan dosen.

|  |  |  |
| --- | --- | --- |
| Use Case Name | Melihat absensi peserta rapat | |
| Use Case Description | Use case ini untuk melihat daftar hadir peserta rapat. | |
| Actor | Ketua prodi dan dosen | |
| Precondition | PRE-1. Ketua prodi dan dosen berhasil *login* kedalam sistem. | |
| Primary Flow of Events | *User* Action | System Response |
| 1. Ketua prodi dan dosen mengklik menu hasil rapat |  |
|  | 1. Sistem menampilkan daftar hasil rapat pada halaman “Rapat”. |
| 1. Ketua prodi dan dosen mengklik *button view*di salah satu hasil rapat |  |
|  | 1. Sistem menampilkan detail hasil rapat pada halaman rapat. |
| 1. Ketua prodi dan dosen mengklik *button* Anggota Rapat |  |
|  | 1. Sistem menampilkan daftar anggota rapat. |
| Error Flow of Events | *User* Action | System Response |
|  |  |
| Post Condition | POST-1. Ketua prodi dan dosen berhasil melihat daftar anggota rapat. | |

### *Use Case Scenario* FR-17

Pada bagian ini dipaparkan *use case scenario* dari fungsi mengubah absensi peserta rapatoleh dosen.

|  |  |  |
| --- | --- | --- |
| Use Case Name | Mengubah absensi peserta rapat | |
| Use Case Description | Use case ini untuk mengubah daftar hadir anggota peserta rapat | |
| Actor | Dosen | |
| Precondition | PRE-1. Dosenberhasil *login* kedalam sistem  PRE-2. Dosen mengklik menu Jadwal Rapat dan melihat detail hasil rapat | |
| Primary Flow of Events | *User* Action | System Response |
| 1. Dosen mengklik *button* Absensi. |  |
|  | 1. Sistem menampilkan halaman Absen dan daftar anggota rapat. |
| 1. Dosen mengubah data absensipeserta rapat |  |
|  | 1. Sistem menyimpan data absensi peserta rapat yang hadir pada database sistem. |
| Error Flow of Events | *User* Action | System Response |
| 3a. Dosen tidak men*-checklist* data absensi peserta rapat. | 3b. Sistem menampilkan pesan error |
| Post Condition | POST-1. Dosen berhasil mengubah absensi peserta rapat | |

### *Use Case Scenario* FR-18

Pada bagian ini dipaparkan *use case scenario* dari fungsi mengirim *email* oleh ketua prodi.

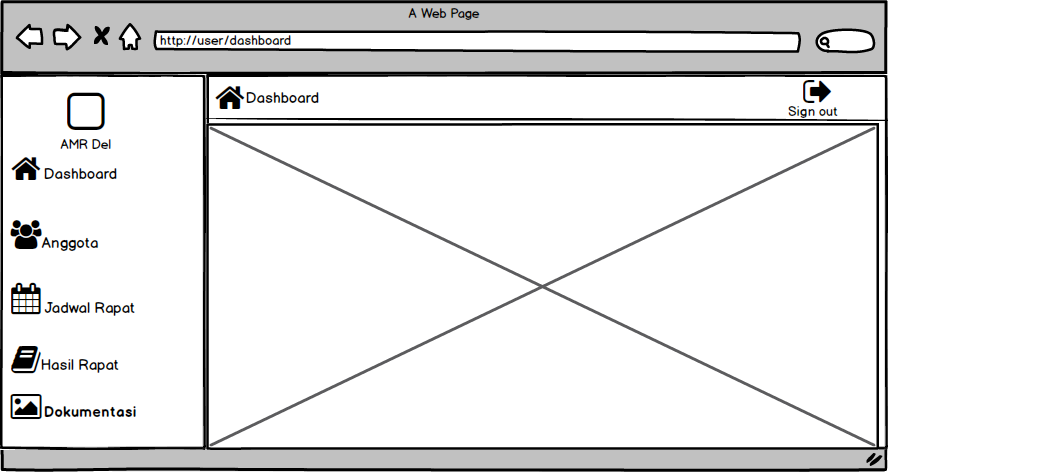
|  |  |  |
| --- | --- | --- |
| Use Case Name | Mengirim *email* | |
| Use Case Description | Use case ini untuk mengirimkan *email* undangan rapat kepada dosen. | |
| Actor | Ketua prodi | |
| Precondition | PRE-1. Ketua prodi berhasil *login* kedalam sistem | |
| Basic Flow of Events | *User* Action | System Response |
| 1. Ketua prodi mengklik menu Buat Rapat. |  |
|  | 1. Sistem menampilkan halaman Buat Rapat |
| 1. Ketua prodi mengisi semua *field* pada form Buat Rapat |  |
| 1. Ketua prodi mengklik *button* Create Meeting |  |
|  | 1. Sistem menyimpan data jadwal rapat pada database sistem. |
|  | 1. Sistem mengirimkan undangan rapat melalui sistem dan *email.* |
|  | 1. Sistem menampilkan halaman “Rapat” yang berisikan jadwal rapat terbaru pada menu Jadwal. |
| Error Flow of Events | *User* Action | System Response |
| 3a. Ketua prodi tidak mengisi semua *field* pada form Buat Rapat | 3b. Sistem akan menampilkan required bahwa semua *field* harus diisi. |
| Post Condition | POST-1. Ketua prodi berhasil mengirimkan undangan rapat melalui *email.* | |

## Desain *Interface* Aplikasi

Pada subbab ini akan menjelaskan rancangan antarmuka aplikasi manajemen rapat yakni mulai dari rancangan antarmuka halaman utama, mencari jadwal rapat, mengelola jadwal rapat, mengelola dokumen hasil rapat, mencari dokumen hasil rapat dan mengelola absensi rapat.

### Desain *User Interface* Halaman Utama User

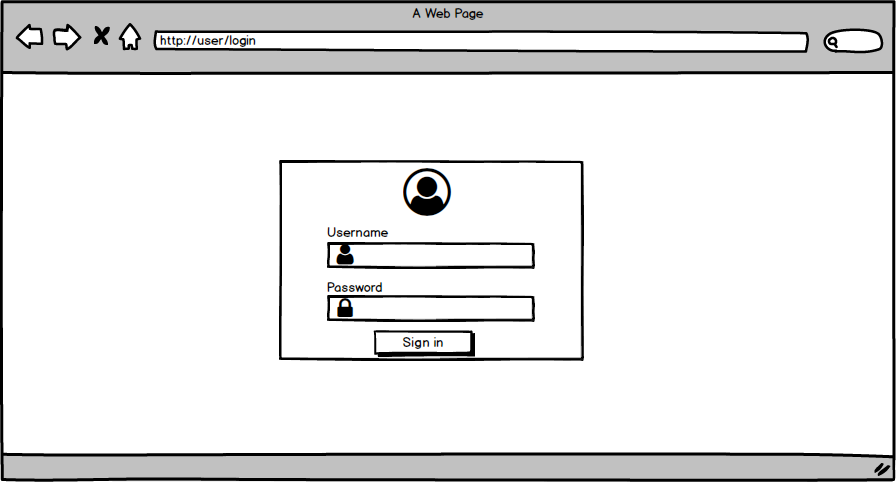
Pada bagian ini ditampilkan desain *user interface* halaman utama Ketua Prodi, Dosen, dan Notulis pada aplikasi manajemen rapat yang dijelaskan pada Gambar 16. Pada halaman utama menampilkan *icon* dari *website* aplikasi manajemen rapat, menu anggota, jadwal rapat, hasil rapat dan menu dokumentasi. Pada halaman awal akan menampilkan informasi-informasi yang terdapat pada sistem. Pada menu anggota, notulis dapat menambahkan daftar absensi peserta rapat yang mengikuti rapat sesuai dengan jadwal rapat. Pada menu jadwal rapat, Ketua Prodi dapat menambahkan jadwal rapat yang baru, dan mengirimkan undangan rapat. Pada menu hasil rapat, notulis dapat menambahkan dokumen hasil rapat yang sudah disetujui oleh Ketua Prodi. Pada menu dokumentasi, Ketua Prodi, dosen, dan notulis dapat mencari dokumen hasil rapat yang sudah selesai dan disetuji oleh Ketua Prodi. Pada halaman utama *user* terdapat *icon setting, icon* ini bertujuan untuk mengarahkan *user* ke halaman profile. Hal profil berguna untuk mengubah *password default* Ketua Prodi,dosen dan notulis yang didaftarkan oleh *admin.*



**Gambar 2 Tampilan Halaman Utama Aplikasi pada User**

### Desain *User Interface* Autentikasi

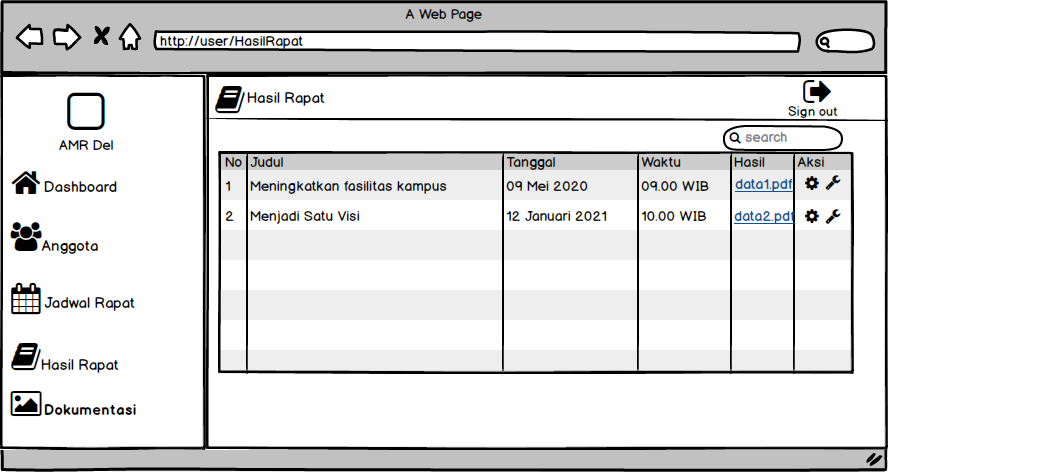
Pada bagian ini ditampilkan desain *user interface* halaman autentikasi oleh Ketua Prodi, Dosen, dan Notulis pada aplikasi manajemen rapat yang dijelaskan pada Gambar 17. Pada halaman *login,*terdapat *field username* dan *password.* User dapat mengakses sistem apabila sudah *login* ke dalam aplikasi manajemen rapat, dengan memasukkan *username* dan *password* yang diberikan oleh *admin* sebagai data *default.* Setelah Ketua Prodi, dosen dan notulis berhasil *login*, maka sistem akan menampilkan halaman utama yaitu *dashboard*.



**Gambar 3 Tampilan Halaman Login Aplikasi pada User**

### Desain *User Interface* Hasil Rapat

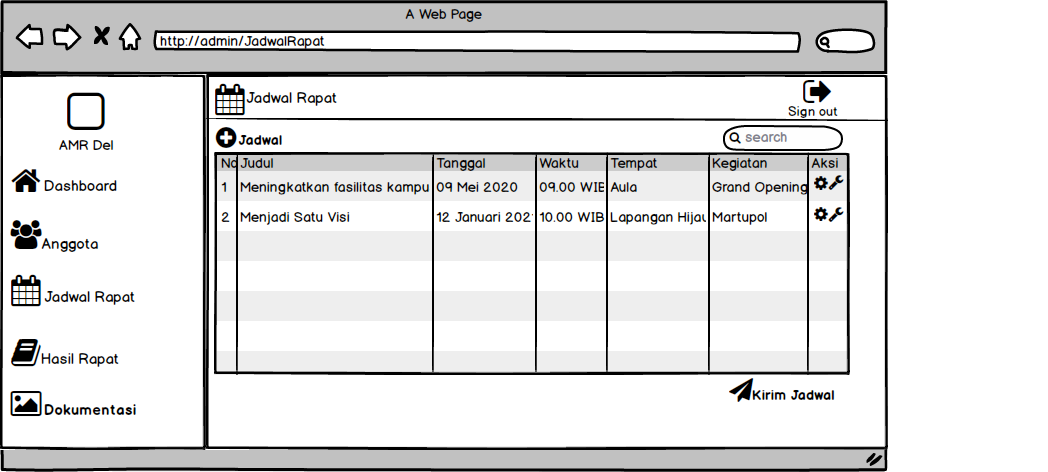
Pada bagian ini ditampilkan desain *user interface* hasil rapat oleh Notulis pada aplikasi manajemen rapat yang dijelaskan pada Gambar 18. Pada halaman ini, notulis akan meng*input* data rapat yang sedang berlansung seperti judul, tanggal, waktu, hasil rapat, dan lokasi rapat. Notulis dapat mengelola hasil rapat apabila terdapat kesalahan dalam penulisan hasil rapat. Hasil rapat yang sudah disetujui oleh ketua rapat akan di *upload* ke sistem, sehingga dosen dan ketua rapat dapat melihat hasil rapat melalui sistem.



**Gambar 4 Tampilan Utama Halaman Menu Hasil Rapat**

### Desain *User Interface* Jadwal Rapat

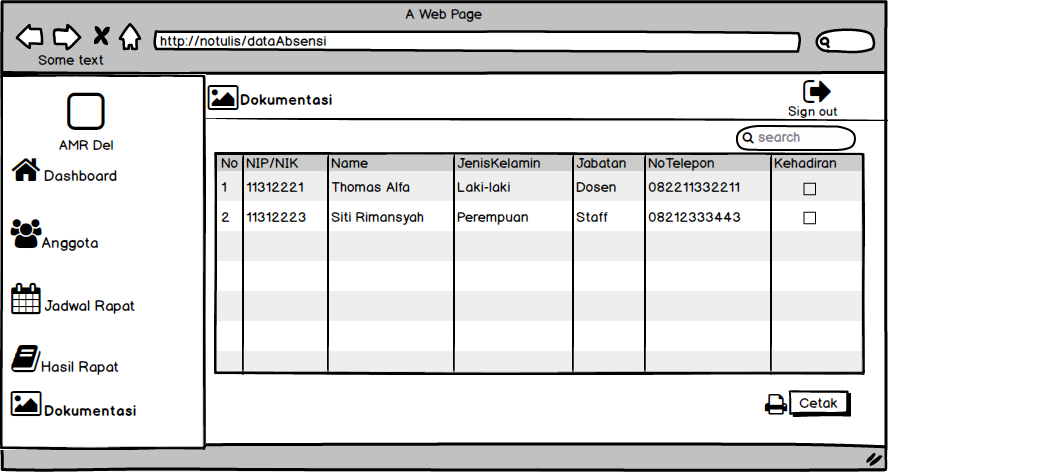
Pada bagian ini ditampilkan desain *user interface* jadwal rapat oleh Ketua Prodi, dan Dosen pada aplikasi manajemen rapat yang dijelaskan pada Gambar 19. Pada halaman menu jadwal rapat, Ketua Prodi dapat menambahkan jadwal rapat yang baru dan mengirimkannya kepada peserta rapat. Jadwal rapat yang diisi oleh Ketua Prodi berupa keterangan judul, tanggal, waktu, tempat, dan kegiatan. Ketua Prodi juga dapat mengelola jadwal rapat pada menu jadwal rapat, apabila pada jadwal rapat ditemukan kesalahan. Dosen akan mendapatkan notifikasi pada navbar, dan untuk detail jadwal rapat akan dikirimkan ke menu jadwal rapat setelah Ketua Prodi mengirimkan undangan rapat.



**Gambar 5 Tampilan Utama Halaman Menu Jadwal Rapat**

### Desain *User Interface* Absensi Peserta Rapat

Pada bagian ini ditampilkan desain *user interface* absensi peserta rapat oleh Ketua Prodi, Dosen dan Notulis pada aplikasi manajemen rapat yang dijelaskan pada Gambar 20. Pada halaman menu anggota, Ketua Prodi, dosen dan notulis dapat melakukan absensi pada sistem. Proses pengabsenan dengan sistem, dimulai dengan Ketua Prodi, dan dosen mengakses sistem kemudian memilih menu anggota. Ketua Prodi dan Dosen akan memilih rapat yang sedang berlansung, pada halaman absensi rapat Ketua Prodi dan dosen akan meng*input* nomor NIP kedalam sistem, maka otomatis sistem akan menyimpan dan memproses absensi Ketua Prodi dan dosen. Setelah berhasil melakukan absensi, maka keterangan seperti NIP, nama, jenis kelamin, jabatan, no telepon, dan status kehadiran akan terisi pada sistem.



**Gambar 6 Tampilan Halaman Absensi Peserta Rapat**

## *Entity Relationship Diagram*

Pada subbab in dijelaskan mengenai kebutuhan data dari aplikasi yang digunakan untuk proses pengembangan aplikasi yaitu dalam bentuk *Entity Relationship Diagram* (ERD*). Entity Relationship Diagram* adalah suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. Pada Gambar dijelaskan mengenai ERD aplikasi manajemen rapat.

**Gambar SEQ Gambar \\* ARABIC 22 *Entity Relationship Diagram***

## *Conceptual Data Model*

*Conceptual Data Model* (CDM) merupakan model yang dibuat berdasarkan seluruh fakta dari dunia nyata serta hubungan antara model tersebut. *Conceptual Data Model* dalam aplikasi pengadaan barang terdapat pada gambar.

## *Physical Data Model*

*Physical Data Model* merupakan model yang menggunakan tabel untuk menggambarkan data serta hubungan antara data tersebut. *Physical Data* *Model* merupakan rancangan fisik *database* yang akan dibangun. Berikut ini merupakan gambar *Physical Data Model* yang dibuat berdasarkan CDM (*Conceptual Data Model*) dapat dilihat pada gambar

# BAB IV IMPLENTASI

Pada bab ini akan dijelaskan mengenai aktivitas yang dilakukan dalam proses implementasi beserta penjelasan yang telah dilakukan dalam pembangunan aplikasi. Tahapan implementasi terdiri atas beberapa bagian yang akan dijelaskan pada bab ini.

* 1. Kebutuhan Implementasi

Pada tahap ini, analisis dilakukan untuk mengidentifikasi kebutuhan *tools* yang digunakan dalam pengerjaan Tugas Akhir ini. Dalam penembangan Aplikasi Manajemen Rapat berbasis *web* diperlukan beberapa *software* dan *hardware* yang digunakan untuk membantu serta mempermudah proses pengerjaan Tugas Akhir, seperti *compile* aplikasi, pengkodean program, dan lain sebagainya. Adapun spesifikasi *software* dan *hardware* yang digunakan adalah sebagai berikut.

**Tabel 4 Kebutuhan Perangkat Keras**

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Perangkat Keras | Spesifikasi | |
| 1. | Asus | *Processor* | Intel® Core™ i5-8250U CPU @ 1.60GHz  1.80 GHz |
|  |  | Memori | 8.00 GB |
| Sistem Operasi | Windows 10 Pro |
| 2. | Lenovo | *Processor* | Intel® Core™ i5-8250U CPU @ 1.60GHz  1.80 GHz |
|  |  | Memori | 8.00 GB |
| Sistem Operasi | Windows 10 Pro |
| 3. | Asus | *Processor* | Intel® Core™ i5-5200U CPU @ 2.20GHz  2.2 GHz |
|  |  | Memori | 8.00 GB |
| Sistem Operasi | Windows 10 Pro |

Spesifikasi perangkat lunak yang digunakan dalam implementasi terdapat pada Tabel 10 berikut.

**Tabel 5 Kebutuhan Perangkat Lunak**

|  |  |  |
| --- | --- | --- |
| No. | Perangkat Lunak | Spesifikasi |
| 1. | *Operating System* | Windows 10 Pro |
| 2. | *Development Tools* | Visual Studio |
| 3. | *Programming Language* | PHP 7.4.6 |
| 4. | *Database* | MySQL |
| 5. | *Framework* | Laravel 8 |
| 6. | *Web Server* | Apache |

* 1. Batasan Implementasi

Hal-hal yang menjadi batasan saat melakukan Pengembangan Aplikasi Manajemen Rapat adalah sebagai berikut :

1. Bahasa Pemrograman yang digunakan adalah PHP.
2. Aplikasi manajemen rapat merupakan hasil implementasi tahap pengembangan yang berbasis *website.*
3. Sumber *project* *Website* berasal dari *project* aplikasi pada mata kuliah Pengembangan Situs Web (PSW) dan juga Github yang tersebar di internet.
4. Dalam pengerjaan Tugas Akhir ini, dilakukan pengembangan sumber data dalam proses implementasi berdasakan pada data rapat program studi D3 Teknologi Informasi.

## Implementasi

Pada subbab ini dijelaskan mengenai implementasi pengembangan aplikasi manajemen rapat dan implementasi fitur tambahan pada aplikasi*.*

* 1. Implementasi Kode Autentikasi

Berikut ini akan dijelaskan subbab yang akan menggambarkan implementasi kode dari fungsi Autentikasi yang terdiri dari login dan logout yang dibangun pada aplikasi manajemen rapat dalam bentuk pseudocode.

* 1. Login

INPUT "Username & Password"

IF "Module exist"

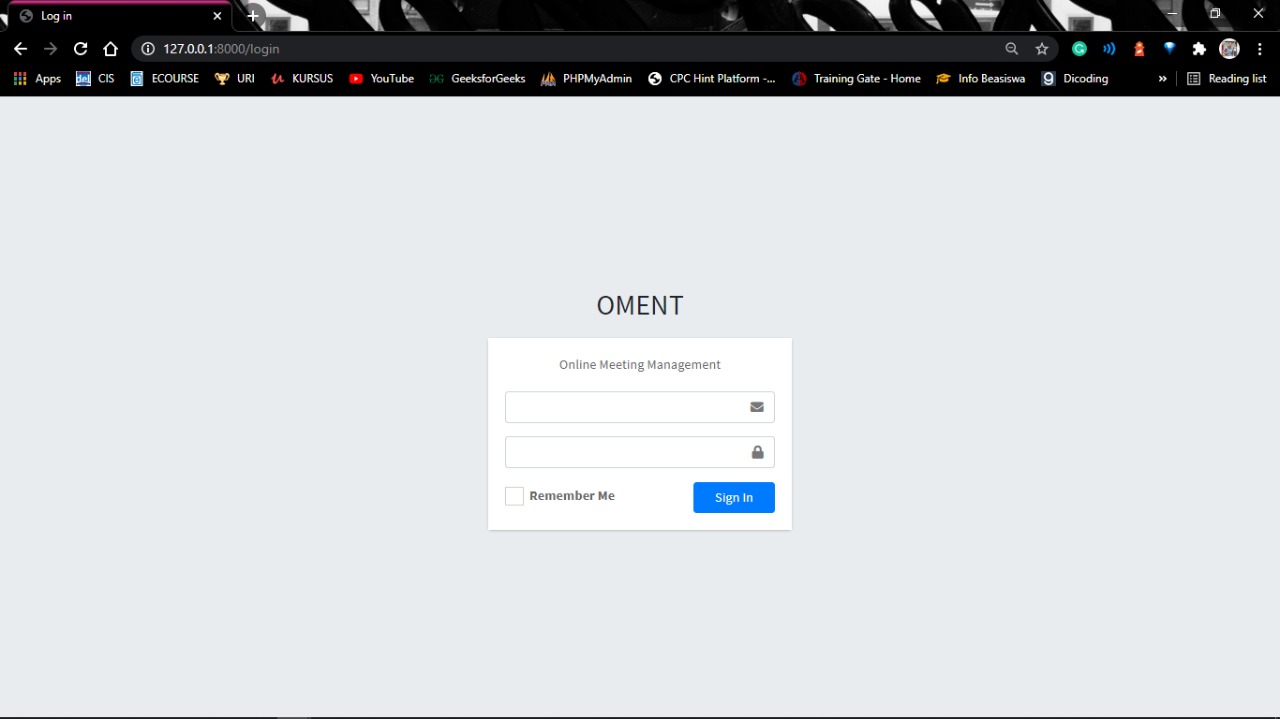
IF "User exists in Role"

THEN "To Dashboard"

ELSE "Back to login"

ELSE "Back to login"

1. Logout



### Implementasi Kode Menambahkan Akun *User*

Berikut ini akan dijelaskan subbab yang akan menggambarkan implementasi kode dari fungsi menambahkan akun user pada aplikasi manajemen rapat dalam bentuk pseudocode.

### Implementasi Kode Mengubah Akun *User*

Berikut ini akan dijelaskan subbab yang akan menggambarkan implementasi kode dari fungsi mengubah akun user pada aplikasi manajemen rapat dalam bentuk pseudocode.

### Implementasi Kode Melihat Akun *User*

Berikut ini akan dijelaskan subbab yang akan menggambarkan implementasi kode dari fungsi melihat akun user pada aplikasi manajemen rapat dalam bentuk pseudocode.

### Implementasi Kode Mencari Jadwal Rapat

Berikut ini akan dijelaskan subbab yang akan menggambarkan implementasi kode dari fungsi mencari jadwal rapat pada aplikasi manajemen rapat dalam bentuk pseudocode.

### Implementasi Kode Menambah Jadwal Rapat

Berikut ini akan dijelaskan subbab yang akan menggambarkan implementasi kode dari fungsi menambah jadwal rapat pada aplikasi manajemen rapat dalam bentuk pseudocode.

Prosedure createRapat( request )

INPUT "Judul,Tanggal,Mulai,Berakhir,Tempat,Id,Notulis,User\_id"

Meeting = []

Meeting.insert(Judul,Tanggal,Mulai,Berakhir,Tempat,Id,Notulis,User\_id)

topic = []

for i = 0: i< length(request.field\_name) do:

row = []

row['judul'] = request.field\_name[i]

row['meeting\_id'] = Meeting.id

topic[] = $row

end for

if "request.lampiran exists",then

for each file in request->file('lampiran'),do

"move file to public path /files"

end for each

file = []

for i = 0; i < length(data) do:

row = []

row['Path'] = data[i]

row['meetings\_id'] = meetings.id

file [] = $row

end for

absence = []

for each item in users,do

row = []

row[users\_id] = item.id

row[meetings\_id] = Meeting.id

absence[] = $row

end for each

end prosedure

### Implementasi Kode Melihat Jadwal Rapat

Berikut ini akan dijelaskan subbab yang akan menggambarkan implementasi kode dari fungsi melihat jadwal rapat pada aplikasi manajemen rapat dalam bentuk pseudocode.

Procedure detailRapat (id)

if "meeting id tidak sama dengan exist " do

404 = Page not found

If

“meeting=meeting\_id”

“lampiran = lampiran\_id”

“topik = topik\_id”

“Notulis = user\_id.minuter”

“leaders = user\_id.leader”

View " Id,meetings,lampiran,Topik,Notulis,Leaders "

endif

End procedure

### Implementasi Kode Mengubah Jadwal Rapat

Berikut ini akan dijelaskan subbab yang akan menggambarkan implementasi kode dari fungsi mengubah jadwal rapat pada aplikasi manajemen rapat dalam bentuk pseudocode.

Procedure editRapat (id)

if" Meeting exist in id"

do editRapat

Where

“meeting = meeting.id”,

“users = role”,

“topik = topik.id”

View editRapat (Meeting, user)

Endif

End procedure

### Implementadi Kode Menghapus Jadwal Rapat

Berikut ini akan dijelaskan subbab yang akan menggambarkan implementasi kode dari fungsi menghapus jadwal rapat pada aplikasi manajemen rapat dalam bentuk pseudocode.

Procedure deleteRapat (id)

if "Meeting id tidak sama dengan exist"

404 = Page not found

Id meeting = database.table “meeting”

deleteRapat

View “hasilRapat”

Endif

End procedure

### Implementasi Kode Menambah Hasil Rapat

Berikut ini akan dijelaskan subbab yang akan menggambarkan implementasi kode dari fungsi menambah hasil rapat pada aplikasi manajemen rapat dalam bentuk pseudocode.

Procedure buatCatatan(request)

INPUT "meetings\_id, Users\_id,request.isi"

Lampiran=[]

if "request.lampiran exists",then

for each file in request->file('lampiran'),do

"move file to public path /files"

end for each

file = []

for i = 0; i < length(data) do:

row = []

row['Path'] = data[i]

row['meetings\_id'] = meetings.id

file [] = $row

end for

endif

end prosedure

### Implementasi Kode Melihat Hasil Rapat

Berikut ini akan dijelaskan subbab yang akan menggambarkan implementasi kode dari fungsi melihat hasil rapat pada aplikasi manajemen rapat dalam bentuk pseudocode.

Procedure lihatCatatan (id)

View=v\_note

v\_note = meeting\_id

Read v\_note

End procedure

### Implementasi Kode Mengubah Hasil Rapat

Berikut ini akan dijelaskan subbab yang akan menggambarkan implementasi kode dari fungsi mengubah hasil rapat pada aplikasi manajemen rapat dalam bentuk pseudocode.

### Implementasi Kode Mencari Hasil Rapat

Berikut ini akan dijelaskan subbab yang akan menggambarkan implementasi kode dari fungsi mencari hasil rapat pada aplikasi manajemen rapat dalam bentuk pseudocode.

### Implementasi Kode Menambahkan Absensi Peserta Rapat

Berikut ini akan dijelaskan subbab yang akan menggambarkan implementasi kode dari fungsi menambah absensi peserta rapat pada aplikasi manajemen rapat dalam bentuk pseudocode.

### Implementasi Kode Melihat Absensi Peserta Rapat

Berikut ini akan dijelaskan subbab yang akan menggambarkan implementasi kode dari fungsi melihat absensi peserta rapat pada aplikasi manajemen rapat dalam bentuk pseudocode.

### Implementasi Kode Mengubah Absensi Peserta Rapat

Berikut ini akan dijelaskan subbab yang akan menggambarkan implementasi kode dari fungsi mengubah absensi peserta rapat pada aplikasi manajemen rapat dalam bentuk pseudocode.

### Implementasi Kode Notifikasi

Berikut ini akan dijelaskan subbab yang akan menggambarkan implementasi kode dari fungsi notifikasi pada aplikasi manajemen rapat dalam bentuk pseudocode.

### Implementasi Kode Mengirim Email

Berikut ini akan dijelaskan subbab yang akan menggambarkan implementasi kode dari fungsi mengirim email undangan rapat pada aplikasi manajemen rapat dalam bentuk pseudocode.

### 

Daftar Pustaka dan Rujukan

|  |  |
| --- | --- |
| [1] | D. Ståhl and J. Bosch, “Modeling continuous integration practice differences in industry software development,” *The Journal of System and Software,* pp. 48-59, 2013. |
| [2] | J. Humble and D. Farley, Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation, Fowler: Addison-Wesley Signature Series, 2010. |
| [3] | M. Fowler, “Continuous Integration,” *Thought Works,* pp. 1-9, 10 September 2000. |
| [4] | S. Elbaum, G. Rothermel and J. Penix, “Techniques for improving regression testing in continuous integration development environments,” *In Proceedings of the International Symposium on Foundations of Software Engineering (FSE),* p. 235–245, 2014. |
| [5] | M. Beller, G. Gousios, A. Panichella and A. Zaidman , “When, how, and why developers (do not) test in their IDEs,” *In Proceedings of the European Software Engineering Conference held jointly with the Symposium on Foundations of Software Engineering (ESEC/FSE),* p. 179–190, 2015. |
| [6] | A. I. G. Prakasa and F. Ardiansyah, “Perancangan User Experience Aplikasi Marketplace Paket Wisata Indonesia untuk Wisatawan Lokal,” *Jurnal Ilmu Komputer dan Agri-informatika,* vol. V, no. 1, pp. 51-60, 2016. |
| [7] | H. S. Samosir, “PENGGUNAAN AUTOMATED TOOLS DALAM PRAKTIK AUTOMATED CONTINUOUS INTEGRATION PADA STUDI KASUS APLIKASI REKAM MEDIS,” *Laporan Tugas Akhir,* p. 51, 28 Mei 2014. |
| [8] | N. T. Bestari and Asmunin, “RANCANG BANGUN APLIKASI AGENDA RAPAT BERBASIS ANDROID,” *Jurnal Manajemen Informatika,* vol. X, no. 01, pp. 49-57, 2020. |
| [9] | J. Highsmith, “What Is Agile Software Development?,” Cross Talk, Alaska, 2002. |
| [11] | B. S. Chacon, “Pro Git,” 15 January 2021. [Online]. Available: https://git-scm.com/book/en/v2;. |
| [12] | A. Labuschagne, “Continuous Integration Build,” The University of Waterloo in fulfillment of the thesis requirement for the degree of Master of Mathematics, Ontario, 2016. |
| [13] | B. Vasilescu, S. v. Schuylenburg, J. Wulms, A. Serebrenik and M. G. v. d. Brand, “Continuous Integration in a Social-Coding World: Empirical Evidence from GitHub,” in *IEEE International Conference on Software Maintenance and Evolution*, Victoria, 2014. |
| [14] | N. Seth and R. Khare, “ACI (automated Continuous Integration) using Jenkins: Key for successful embedded Software development,” 2nd Int. Conf. Recent Adv. Eng. Comput. Sci. RAECS, English, 2016. |
| [15] | [Online]. Available: http://searchsoftwarequality.techtarget.com/definition/continuousintegration. [Accessed 15 January 2021]. |
| [16] | J. d. D. F. Humble, “Continuous Delivery: Reliable Software Releases through Build, Test and Deployment Automation,” Addison-Wesley Signature Series, USA, 2010. |
| [17] | [Online]. Available: http://www.cognifide.com/blogs/quality-assurance/automated-testscontinous-integration-environment-1/. [Accessed 15 January 2021]. |
| [18] | T. Molter, “What is Jenkins?,” Jenkins, 2012. [Online]. Available: https://wiki.jenkins-ci.org/display/JENKINS/Meet+Jenkins. [Accessed 16 January 2021]. |
| [19] | S. Ravishankar, “Git: Version Control for Everyone,” Packt Publishing Ltd, Birmingham B3 2PB, 2013. |
| [20] | T. B. C. S. Glenford J. Myers, “The Art of Software Testing 3rd Edition,” JohnWiley & Sons, Hoboken, 2012. |
| [21] | P. R. S, “Software Engineering: A Practitioner’s Approach Fifth Edition,” The McGraw Hill, New York, 2001. |
| [22] | Patrick Oladimeji, “Level Of Testing,” 2007. [Online]. Available: http://www.cs.swan.ac.uk/~csmarkus/CS339/dissertations/OladimejiP.pdf. [Accessed 15 January 2021]. |
| [23] | S. M. a. a. G. P. Duvall, “Continuous integration: improving software quality and reducing risk,” 2007. |
| [24] | P. M. S. M. d. A. G. Duvall, “Continuous Integration: Improving software quality and reducing risk,” in *Addison-Wesley Signature Series*, USA, 2007. |
| [25] | E. T. Samosir, H. Samosir, I. Liem and Y. Dwi, “The utilization of automated tools in the automated continuous integration practice case study: Medical record application,” *J. Telecommun. Electron. Comput. Eng,* vol. 9, no. 2-5, pp. 123-127, 2017. |
| [26] | J. Baumeister and J. Reutelshoefer, “Developing knowledge systems with continuous integration,” in *Conference Proceeding Ser*, ACM Int, 2011. |
| [27] | “The Apache Ant Project,” 2013. [Online]. Available: http://ant.apache.org/. [Accessed 15 January 2021]. |

LAMPIRAN

**Lampiran 1 – Hasil Wawancara**

Hari/Tanggal : Sabtu, 14 November 2020

Waktu : 10.00 – 11.15

Narasumber : Ike Fitriyaningsih, S.Si., M.Si

Ruang : Zoom (Online)

Pewawancara : Tim D3TI-10

Tujuan : Mengetahui proses manajemen rapat

|  |  |  |
| --- | --- | --- |
| Pewancara | : | Untuk meng-*upload* hasil dokumentasi rapat yang sudah fix kemana bu? Ke *email* kah bu? |
| Narasumber | : | Bukan, kita punya *email list* dosen prodi kan. Yang .del.ac.id, nah kesitu lah biasanya kita kirim form rapat itu. |
| Pewancara |  | Kalau untuk penyimpanan hasil dokumen rapat nya, bagaimana prosesnya bu? |
| Narasumber |  | Kan pertama disahkan dulu kan oleh notulis sama ketuaa rapat kemudian dikirimkan *email*, nah dikirimkan *email* itu kita punya disimargala ada apa namanya lokasi penyimpanan rapat, nah tapi misalnya kalian sudah bangun system ini, buat terpisah juga boleh misalnya ada folder yang menampung MoM hasil rapat, spserti itu. |
| Pewancara |  | Maaf bu memotong, simargala itu apa bu? |
| Narasumber |  | Simargala itu working folder ya, jadi *server*-nya sebenarnya ada didalam kampus, nah disitu la semua dokumen-dokumen IT Del disimpan dan bisa diakses oleh semua dosen staff. Jadi sebenarnya dia tempat penyimpanan file tapi bisa diakses oleh semua orang, maksudnya semua dosen staff ya, kalo kalian kan yang untuk mahasiswa juga. |
| Pewancara |  | Jadi yang mengirim hasilnya itu notulis nya bu? |
| Narasumber |  | Iya notulis |
| Pewancara |  | Jadi kalau misalnya bu ada sanggahan atau tambahan yang diberikan oleh peserta rapat ke hasil rapat yang sudah final kan bu, itu biasanya dibuat perubahan lagi atau dimasukkan ke pembahasan rapat selanjutnya bu? |
| Narasumber |  | Dibuat ke meeting selanjutnya saja, kalau memang sudah disetujui sama notulis sama ketua rapat ya |
| Pewancara |  | Berarti tidak perlu dirubah kan bu? |
| Narasumber |  | Tidak usah, itu menjadi isu di meeting selanjutnya saja |
| Pewancara |  | Untuk proses pencarian dokumen hasil rapatnya itu bu biasanya dalam benuk keyword seperti apa bu? Semisal ada dosen yang mau mencari hasil rapat bu |
| Narasumber |  | Jadi penyimpanannya itu ada aturannya, yang pertama itu tanggal, kemudian underscore MoM, kemudian rapat apa itu, ada judul rapatnya da nada aturan penyimpanan file rapatnya |
| Pewancara |  | Misalkan ada dosen yang sudah pernah menjadi notulis dan sudah tidak menjadi dosen di FITE lagi, semisal ada dosen yang mencari hasil rapatnya apakah data dosennya masih tetap ditampilkan bu? |
| Narasumber |  | Nah berarti kalian ini user nya bisa di-*update* dan di-*delete* ya? |
| Pewancara |  | Iya bisa bu |
| Narasumber |  | Tapi kalau dokumennya itu sudah menjadi sebuah dokumen kan, nah masalahnya ini kan pada saat kita mengirim undangan atau pada saat kita menentukan notulis lagi? |
| Pewancara |  | Maksud saya bu, misalkan ada dokumen hasil rapat yang sudah selesai, jadi untuk di lain waktu ada dosen lain yang mencari dokumen tersebut tetapi notulis nya sudah tidak menjadi dosen lagi, itu data dari notulis nya tetap ada bu? |
| Narasumber |  | Tetap lah, karna itu kan sebuah dokumen yang di-*generate* sebelum dia keluar, nah jadi MoM bukan lagi hak notulis dan Ketua Prodi ketika sudah di *share*, itu sudah menjadi hak dari semua anggota, jadi semua bisa *search* |
| Pewancara |  | Berarti dari *current system* sekarang masalah-masalah yang dihadapi apa aja sih bu? |
| Narasumber |  | Masalahnya dalam pencatatan itu sebenarnya, memang kadang-kadang notulis itu tidak bekerja ya, maksudnya lama sekali kita dapat hasil rapatnya, mungkin dengan adanya system kalian itu bisa ada pengingatnya, oh ini belum ditulis, sehingga bisa terorganisir dengan baik, maksudnya Mo Mini isu nya apa dan hasilnya apa. Kalau yang sekarang memang agak susah, karena ketika rapat kita sudah dapat nih solusinya, tetapi terkadang kita lupa untuk menulisnya |
| Pewancara |  | Biasaya setelah rapat MoM bisa didapatkan setelah beberapa hari setelah selesai rapat? |
| Narasumber |  | Biasanya 2-3 hari, ituoun tidak pasti ya, kadang beberapa bula pun MoM nya belum tentu ada, tapi kita sudah tau oh hasinya seperti apa, Cuma pencatatannya memang sangat kurang untuk rapat |
| Pewancara |  | Misalnya MoM sudah selesai 3 hari dosen lain itu bisa nge cek MoM nya untuk task yang harus dilakukan ya bu? Karena dosen tidak memiliki catatan satu-satu kan bu |
| Narasumber |  | Tidak, lebih ke yasudah 1 MoM itu untuk kita semua, ya walaupun saya tidak menulis tetapi saya tau siapa yang disuruh |
| Pewancara |  | Berarti bu di simargala itu ada beberapa folder sesuai jenis rapat atau hanya satu folder menampung semua hasil rapat? |
| Narasumber |  | Kalau untuk prodi langsung semua rapat, yang dibedakan itu justru prodi a, prodi b, prodi c, fakultas sama institute |
| Pewancara |  | Berarti untuk rekapitulasi per bulan itu adanya bu? Karena ibu bilang rapat itu dilakukan sekali per bulan |
| Narasumber |  | itu rapat rutin ya, karena biasanya rapat prodi itu yang dibicarakan tidak terlalu banyak, tapi kalau memang ada sesuatu yang harus dibicarakan itu boleh tetapi harus dibicarakan dulu dengan Ketua Prodinya |
| Pewancara |  | perlunya email untuk mengirim hasil rapat? |
| Narasumber |  | perlu, biasanya pakai email si notulis |
| **Kesimpulan** | : | 1. Proses rapat di program studi D3 Teknologi Informasi diawali dengan mengirim undangan oleh Ketua Prodi kepada peserta rapat yang berisi tempat, waktu, notulis rapat yang akan diadakan beserta agenda. Lalu, setelah rapat selesai peserta membuat minute of meeting guna menindaklanjuti hasil rapat 2. Bukan hanya dosen yang dapat mengikuti rapat, namun staff terkadang juga diperlukan dalam mengikuti rapat 3. Sistem dapat mengingatkan beberapa kali kepada peserta rapat sebelum rapat berlangsung 4. Website diharapkan mampu menyimpan hasil rapat juga membuat hasil tindak lanjut rapat berupa tugas yang harus dilakukan peserta rapat setelah tugas diberikan pada saat rapat berlangsung |

**Lampiran 2 – Hasil Wawancara**

Hari/Tanggal : Selasa, 2 Februari 2021

Waktu : 10.00 – 10.30

Narasumber : Ike Fitriyaningsih, S.Si., M.Si

Ruang : Zoom (Online)

Pewawancara : Tim D3TI-10

Tujuan : Mengetahui proses *current system* manajemen rapat di Fakultas Informatika dan Elektro, prodi D3 Teknologi Informasi

| No | Hasil Wawancara | | |
| --- | --- | --- | --- |
| 1 | Pewawancara | : | Pada saat Ketua Prodi mengirimkan undangan rapat bu, apakah sudah termasuk mengirimkan nama notulisnya bu? |
| Narasumber | : | Ya, sudah termasuk. |
| 2 | Pewawancara | : | Untuk jadwal dan tempat rapat bagaimana yah bu untuk kondisi pada saat ini? |
| Narasumber | : | Untuk saat ini, penentuan tempat yang mungkin berbeda yah. Untuk tempatnya itu diisi oleh ketua rapat atau notulen. Pengisian tempat rapat oleh notulen dapat diperpanjang sampai pagi hari sebelum rapat dimulai. |
| 3 | Pewawancara | : | Perpanjangan maksudnya bagaimana yah bu? |
| Narasumber | : | Untuk tempatnya itu tidak lansung dilampirkan di undangan rapat atau tidak lansung dikirimkan melalui email oleh notulen. Jadi pengiriman tempat rapat biasanya dikirimkan 2 jam sebelum rapat dihari yang sama untuk pelaksanaan rapat. Tetapi baiknya dikirimkan sehari sebelum rapat diadakan. |
| 4 | Pewawancara | : | Apakah pengiriman undangan rapat dilakukan dihari pada saat rapat akan berlansung bu? Atau H-beberapa hari? |
| Narasumber | : | Idealnya sih dikirimkan sehari sebelum rapat berlansung yah, hal ini juga untuk memastikan dosen dapat menghadiri rapat. Karena tidak mungkin mengundang dosen pada hari rapat berlansung. |
| 5 | Pewawancara | : | Jadi bu, untuk pemilihan hari rapat bu tidak ada pemilihan khusus yah bu? |
| Narasumber | : | Ya, bukan ada jadwal tertentu kita. |
| 6 | Pewawancara | : | Bagaimana proses pemilihan hari untuk rapat bu? apakah terlebih dahulu menanyakan kepada peserta rapat atau mencari jadwal kosong peserta rapat? |
| Narasumber | : | Sebenarnya untuk di FITE, pada hari rabu sore itu sudah dikosongkan untuk rapat. Hari rabu tersebut sudah disetting untuk tidak membuat dosen bekerja, atau dikosongkan jadwalnya. Nah cuman, pada kenyataannya jadwal kita banyak dan sumber daya kita sedikit untuk tenaga kerja maka kadang ada beberapa dosen yang memplot untuk rabu sore itu mengajar. |
| 7 | Pewawancara | : | Jadi bu, jadwalnya diganti lagi bu seperti itu? |
| Narasumber | : | Sebenarnya itu, untuk rapat prodi aturannya itu diadakan sebulan sekali harus rapat dan rutin. Tetapi untuk penentuan waktunya, harinya itu tidak ditentukan diawal tanggal berapa, hari apa begitu. Asalkan ada *issue* maka itu dimasukkan kedalam rapat bulanan. |
| 8 | Pewawancara | : | Jadikan bu untuk masa pada saat ini kan rapat diadakan melalui zoom bu, jadi jika ada peserta rapat yang tidak dapat hadir, bagaimana peserta rapat mengkonfirmasi ketidakhadiran kepada ketua rapat bu? apakah ada batas waktu yang ditentukan? |
| Narasumber | : | Kalau biasanya sih ini lewat *Whatsapp.* Tapi kalau bisa kalian masukkan saja ini ke sistem kalian. Kan disitu ada daftar hadir rapat kan, nah itu akan diurus oleh notulis. Tetapi alangkah lebih baiknya dosen juga dapat melakukan konfirmasi kehadiran melalui sistem kalian, jadi ada notifikasi kepada notulis. Tujuannya agar ter-*record* di sistem kalian. |
| 9 | Pewawancara | : | Jadi bu, untuk pemilihan peserta rapat adakah perubahan? atau untuk peserta rapat tidak mengalami perubahan setiap rapat diadakan? |
| Narasumber | : | Kalau rapat prodi sebernarnya tetap. Tapi mungkin kadang-kadang kita butuh mengudang pihak luar kali yah. |
| 10 | Pewawancara | : | Contoh kasusnya mengundang pihak luar pada kondisi seperti apa ya bu kalau boleh tau? |
| Narasumber | : | Mungkin kita butuh undang BAAK, tapi sebenarnya nih sejauh ini kita belum pernah sampai mengundang BAAK. Karena BAAK itu kan resource nya fakultas. Nah jadi kalau kita rapat prodi yang hadir itu hanya dosen-dosen prodi saja begitu. |
| 11 | Pewawancara | : | Jadikan bu setiap rapat kan bu ada notulis, jadi untuk pemilihan notulis ini bagaimana yah bu? Dan jika notulis tidak dapat hadir solusinya bagaimana yah bu? |
| Narasumber | : | Tapi dia sudah izin yah. Kalau dia sudah izin harus biasanya sih mencari ganti yah. |
| 12 | Pewawancara | : | Itu untuk dilanjutkan kegiliran atau bagaimana yah bu? Atau notulis sendiri yang mencari gantinya bu? |
| Narasumber | : | Kita akan membuat untuk ke giliran selanjutnya yah. Jadi kita akan melanjutkan ke notulis berikutnya sesuai abjad, tetapi untuk rapat selajutnya maka notulis yang sebelumnya tidak dapat hadir akan mengisi jadwalnya di rapat selanjutnya begitu, supaya adil yah. Diurutkan abjad saja dek. |
| 13 | Pewawancara | : | Jadi bu tidak perlu diurutkan dengan NIDN? |
| Narasumber | : | Janganlah, tidak perlu. Karena ada dosen yang belum memiliki NIDN saat ini. |
| 14 | Pewawancara | : | Jadi bu, selain Ketua Prodi ada tidak bu yang diperbolehkan mengadakan rapat bu? Misalnya dosen lain bu, tetapi meminta mengadakan rapat bu? |
| Narasumber | : | Kalau rapat setahu saya hanya Ketua Prodi saja yah rapat dalam prodi. Sebenarnya boleh, cuman harus mengkonfirmasi kepada Ketua Prodi, harus lewat Ketua Prodi untuk mengadakan rapat. Jadi jangan dosen mengadakan rapat, diluar rapat yang diketahui Ketua Prodi begitu. Contohnya semisal masalahnya tidak terlalu penting, dapat di pending dan akan dibahas di rapat selanjutnya begitu. |
| 15 | Pewawancara | : | Jadi bu kalau rapat dipending ini apakah tetap membahas masalah yang sama kah bu? Dan kalau rapat dipending ini bu siapa ya bu kalau boleh tau yang melakukan pending? |
| Narasumber | : | Ya, Ketua Prodi. |
| 16 | Pewawancara | : | Jadi bu, terkait task-task untuk dosen setelah rapat itu bu maksudnya bagaimana yah bu? Maksud saya Minute of Meeting bu. |
| Narasumber | : | Betul, jadi itu sudah termasuk kedalam *template* sebenarnya yah. Jadi dibagian bawah itu ada hal-hal apa saja yang perlu ditindak lanjut beserta PIC atau penanggungjawab nya. Nah itu sebenarnya itu yang kita perlukan, apakah bisa solved atau tidak. Misalnya kita ada rapat, ada masalah seperti kasus mahasiswa, nah siapa nih yang bertanggung jawab menghubungi, pasti walinya kan. Nah itu menjadi status pending, tetapi jika sudah berhasil dihubungi oleh wali maka masalah tersebut akan menjadi status solved seperti itu. Dan hasil dari menghubunginya tidak perlu ditindaklanjuti, begitu. |
| 17 | Pewawancara | : | Jadi bu, untuk sistem pendataan daftar hadirnya bagaimana yah bu? Untuk sistem online saat ini bu? |
| Narasumber | : | Kalau sekarang yah, yang hadir di rapat itu ada dilist dia hadir. Jadi kalau tidak ada berarti dia sudah izin atau misalnya diingatkan kan di wa group, jadi kalau izin mereka berarti tidak hadir. |
| 18 | Pewawancara | : | Jadi bu untuk rekaptulasi daftar hadir bu bagaimana bu jika sistem pendataan daftar hadirnya begitu bu? |
| Narasumber | : | Nah itu sebernarnya masuk ke notulen rapat. |
| 19 | Pewawancara | : | Jadikan bu, semisal rapat sudah selesai untuk dokumen hasil rapatnya bu apakah perlu melakukan konfirmasi kepada ketua rapat dahulu bu? Atau bagaimana bu? |
| Narasumber | : | Jadi, di form notulen oleh si notulis itu ada 2 tanda tangan yaitu tanda tangan ketua rapat dan tanda tangan di notulis. Jadi sebelum disebarkan MoM nya itu melalui email, sebenarnya ada pengecekan lagi dari 2 legalitas tersebut. |
| 20 | Pewawancara | : | Untuk meng-*upload* hasil dokumentasi rapat yang sudah fix kemana bu? Ke *email* kah bu? |
| Narasumber | : | Bukan, kita punya *email list* dosen prodi kan. Yang .del.ac.id, nah kesitu lah biasanya kita kirim form rapat itu. |
| 21 | Pewawancara | : | Kalau untuk penyimpanan hasil dokumen rapat nya, bagaimana prosesnya bu? |
| Narasumber | : | Kan pertama disahkan dulu kan oleh notulen sama ketuaa rapat kemudian dikirimkan *email*, nah dikirimkan *email* itu kita punya disimargala ada apa namanya lokasi penyimpanan rapat, nah tapi misalnya kalian sudah bangun system ini, buat terpisah juga boleh misalnya ada folder yang menampung MoM hasil rapat, spserti itu. |
| 22 | Pewawancara | : | Maaf bu memotong, simargala itu apa bu? |
| Narasumber | : | Simargala itu working folder ya, jadi servernya sebenarnya ada didalam kampus, nah disitu la semua dokumen-dokumen IT Del disimpan dan bisa diakses oleh semua dosen staff. Jadi sebenarnya dia tempat penyimpanan file tapi bisa diakses oleh semua orang, maksudnya semua dosen staff ya, kalo kalian kan yang untuk mahasiswa juga. |
| 23 | Pewawancara | : | Jadi yang mengirim hasilnya itu notulen nya bu? |
| Narasumber | : | Iya notulen |
| 24 | Pewawancara | : | Jadi kalau misalnya bu ada sanggahan atau tambahan yang diberikan oleh peserta rapat ke hasil rapat yang sudah final kan bu, itu biasanya dibuat perubahan lagi atau dimasukkan ke oembahasan rapat selanjutnya bu? |
| Narasumber | : | Dibuat ke meeting selanjutnya saja, kalau memang sudah disetujui sama notulen sama ketua rapat ya. |
| 25 | Pewawancara | : | Berarti tidak perlu dirubah kan bu? |
| Narasumber | : | Tidak usah, itu menjadi isu di meeting selanjutnya saja. |
| 26 | Pewawancara | : | Untuk proses pencarian dokumen hasil rapatnya itu bu biasanya dalam benuk keyword seperti apa bu? Semisal ada dosen yang mau mencari hasil rapat bu |
| Narasumber | : | Jadi penyimpanannya itu ada aturannya, yang pertama itu tanggal, kemudian underscore MoM, kemudian rapat apa itu, ada judul rapatnya dan ada aturan penyimpanan file rapatnya |
| 27 | Pewawancara | : | Misalkan ada dosen yang sudah pernah menjadi notulen dan sudah tidak menjadi dosen di FITE lagi, semisal ada dosen yang mencari hasil rapatnya apakah data dosennya masih tetap ditampilkan bu? |
| Narasumber | : | Nah berarti kalian ini user nya bisa di-*update* dan di-*delete* ya? |
| 28 | Pewawancara | : | Iya bisa bu |
| Narasumber | : | Tapi kalau dokumennya itu sudah menjadi sebuah dokumen kan, nah masalahnya ini kan pada saat kita mengirim undangan atau pada saat kita menentukan notulen lagi? |
| 29 | Pewawancara | : | Maksud saya bu, misalkan ada dokumen hasil rapat yang sudah selesai, jadi untuk di lain waktu ada dosen lain yang mencari dokumen tersebut tetapi notulis nya sudah tidak menjadi dosen lagi, itu data dari notulis nya tetap ada bu? |
| Narasumber | : | Tetap lah, karna itu kan sebuah dokumen yang di­-*generate* sebelum dia keluar, nah jadi MoM bukan lagi hak notulen dan Ketua Prodi ketika sudah di *share*, itu sudah menjadi hak dari semua anggota, jadi semua bisa *search.* |
| 30 | Pewawancara | : | Berarti dari *current system* sekarang masalah-masalah yang dihadapi apa aja sih bu? |
| Narasumber | : | Masalahnya dalam pencatatan itu sebenarnya, memang kadang-kadang notulen itu tidak bekerja ya, maksudnya lama sekali kita dapat hasil rapatnya, mungkin dengan adanya system kalian itu bisa ada pengingatnya, oh ini belum ditulis, sehingga bisa terorganisir dengan baik, maksudnya Mo Mini isu nya apa dan hasilnya apa. Kalau yang sekarang memang agak susah, karena ketika rapat kita sudah dapat nih solusinya, tetapi terkadang kita lupa untuk menulisnya. |
| 31 | Pewawancara | : | Biasaya setelah rapat MoM bisa didapatkan setelah beberapa hari setelah selesai rapat? |
| Narasumber | : | Biasanya 2-3 hari, ituoun tidak pasti ya, kadang beberapa bula pun MoM nya belum tentu ada, tapi kita sudah tau oh hasinya seperti apa, Cuma pencatatannya memang sangat kurang untuk rapat. |
| 32 | Pewawancara | : | Misalnya MoM sudah selesai 3 hari dosen lain itu bisa nge cek MoM nya untuk task yang harus dilakukan ya bu? Karena dosen tidak memiliki catatan satu-satu kan bu. |
| Narasumber | : | Tidak, lebih ke yasudah 1 MoM itu untuk kita semua, ya walaupun saya tidak menulis tetapi saya tau siapa yang disuruh. |
| 33 | Pewawancara | : | Berarti bu di simargala itu ada beberapa folder sesuai jenis rapat atau hanya satu folder menampung semua hasil rapat? |
| Narasumber | : | Kalau untuk prodi langsung semua rapat, yang dibedakan itu justru prodi a, prodi b, prodi c, fakultas sama institute. |
| 34 | Pewawancara | : | Berarti untuk rekapitulasi per bulan itu adanya bu? Karena ibu bilang rapat itu dilakukan sekali per bulan. |
| Narasumber | : | itu rapat rutin ya, karena biasanya rapat prodi itu yang dibicarakan tidak terlalu banyak, tapi kalau memang ada sesuatu yang harus dibicarakan itu boleh tetapi harus dibicarakan dulu dengan Ketua Prodinya. |
| 35 | Pewawancara | : | Perlunya email untuk mengirim hasil rapat? |
| Narasumber | : | Perlu, biasanya pakai email si notulen. |