

BAB III METODOLOGI PENELITIAN

3.1 Identifikasi Masalah

3.1.1 Tujuan Penyelesaian Masalah

Tujuan penyelesaian masalah dalam tugas akhir ini adalah untuk memudahkan dosen dan karyawan di UII untuk melakukan presensi. Sistem presensi yang dibangun memungkinkan dosen untuk melakukan presensi secara seamless atau terotomasi dengan cara memanfaatkan *log wifi* dari UII *Connect*. Selain itu, dengan sistem ini juga memungkinkan top management untuk melihat performa karyawan baik dari segi kuantitas kehadiran hingga ketepatan waktu. Dengan fitur tersebut, diharapkan dapat dijadikan sebagai acuan untuk pengambilan keputusan khususnya dalam lingkup SDM.

3.1.2 Pengumpulan Data

Setelah merumuskan tujuan penyelesaian masalah, selanjutnya diperlukan pengumpulan beberapa data pendukung yang didapatkan dari berbagai sumber. Pada Tabel 3.1 merupakan prosedur yang digunakan pada saat pengumpulan data.

Tabel 3.1 Tabel Pengumpulan Data

Teknik	Data	Sumber
Observasi	- Proses melakukan perekaman kehadiran	Lingkungan UII
Studi Pustaka	<ul style="list-style-type: none"> - Teori Seamless - Teori Business intelligence dan Pemanfaatannya - Teori efektifitas sistem presensi 	Jurnal, Artikel dan Internet

3.2 Analisis

Sistem pada penelitian ini dibuat berdasarkan proses identifikasi masalah yang telah dijelaskan pada bab sebelumnya. Berdasarkan data tersebut maka penulis menyimpulkan bahwa terdapat dua sistem yang dibutuhkan, yaitu Sistem *Cron Job* dan Sistem Manajemen Presensi. Sistem *Cron Job* nantinya berfungsi untuk mengambil data pengguna yang terhubung ke jaringan UII *Connect* kemudian diolah menjadi data presensi sedangkan Sistem Manajemen Presensi berfungsi untuk mengelola data presensi.

3.2.1 Alat Penelitian

Alat penelitian disini adalah komponen hardware dan software yang digunakan sebagai alat dalam mendukung penelitian yang dilakukan, alat ini yang akan dijadikan sebagai alat pengolahan data dan pembuatan sistem yang akan dirancang.

- Hardware

Spesifikasi hardware yang digunakan adalah sebagai berikut:

Processor : Intel Core i5

Memory : 8GB

SSD : 256 GB

- Software

Spesifikasi software yang digunakan adalah sebagai berikut:

Sistem Operasi : MacOS Mojave

Bahasa Program : Elixir, Typescript

Tools : Visual Studio Code

DBMS : MySQL

Web Server : Nginx

Web Browser : Google Chrome

3.3 Perancangan

3.3.1 Flowchart Diagram

Sistem *Cron Job* yang dirancang dengan menggunakan *flowchart* yang menampilkan berbagai macam proses yang ada pada program secara keseluruhan sehingga bisa dilihat proses – proses apa saja yang ada di dalam program.

Cron Job adalah kumpulan perintah yang dijalankan pada waktu tertentu setiap hari, minggu, bulan, bahkan bisa pada jam tertentu. Dengan kata lain pengertian *cron job* adalah

sistem yang bertugas untuk penjadwalan yang memungkinkan melakukan perintah tertentu secara terjadwal. Penjadwalan dilakukan berdasarkan pada tabel yang disebut *crontab*. *Crontab* merupakan kependekan dari *Cron Table* yang berisi kumpulan perintah yang akan dijalankan oleh cron. Struktur crontab adalah sebagai berikut:

Struktur Perintah File Crontab

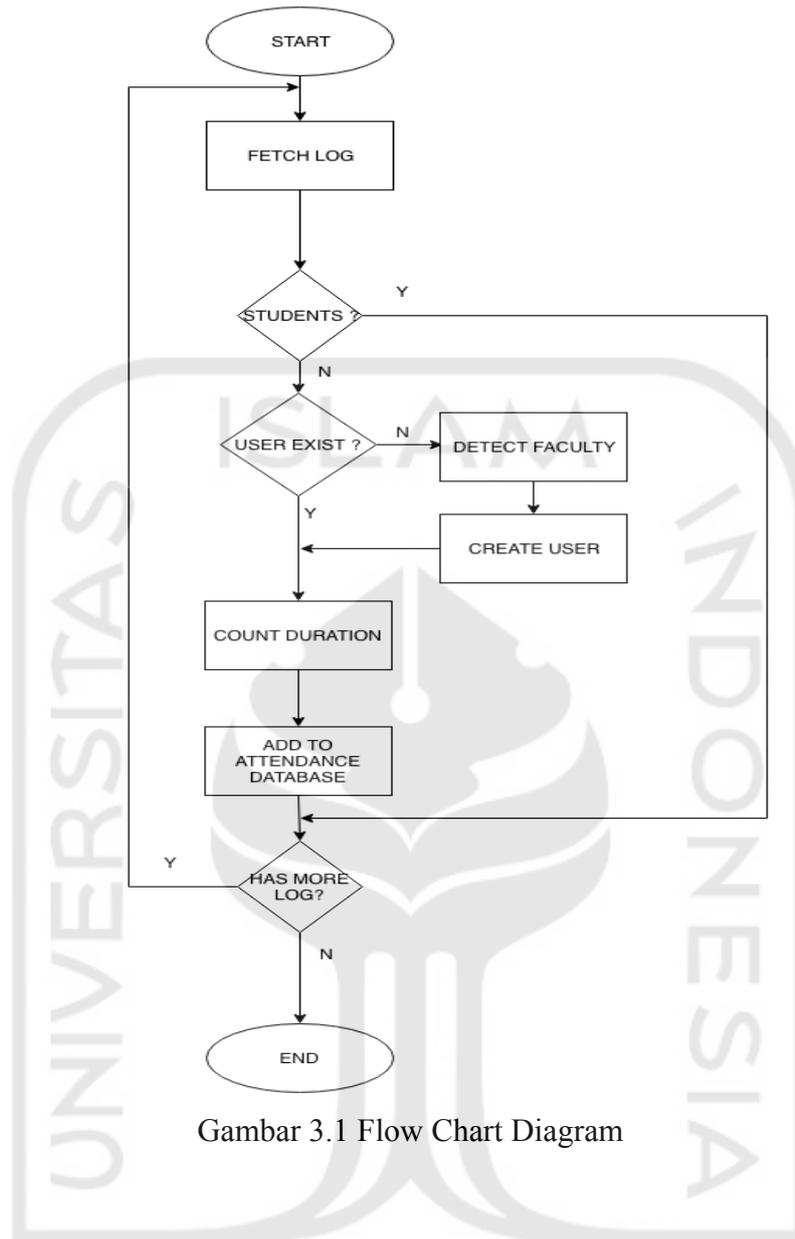
* * * * *

Maksudnya adalah, untuk tanda bintang yang ke:

1. Menit (0-59)
2. Jam (0-23)
3. Hari (1-31)
4. Bulan (1-12)
5. Mingguan (0-6)
6. User (optional)
7. Command/Perintah

Proses-proses yang dilakukan pada sistem ini seperti yang ditunjukkan pada Gambar 3.1 adalah sebagai berikut:

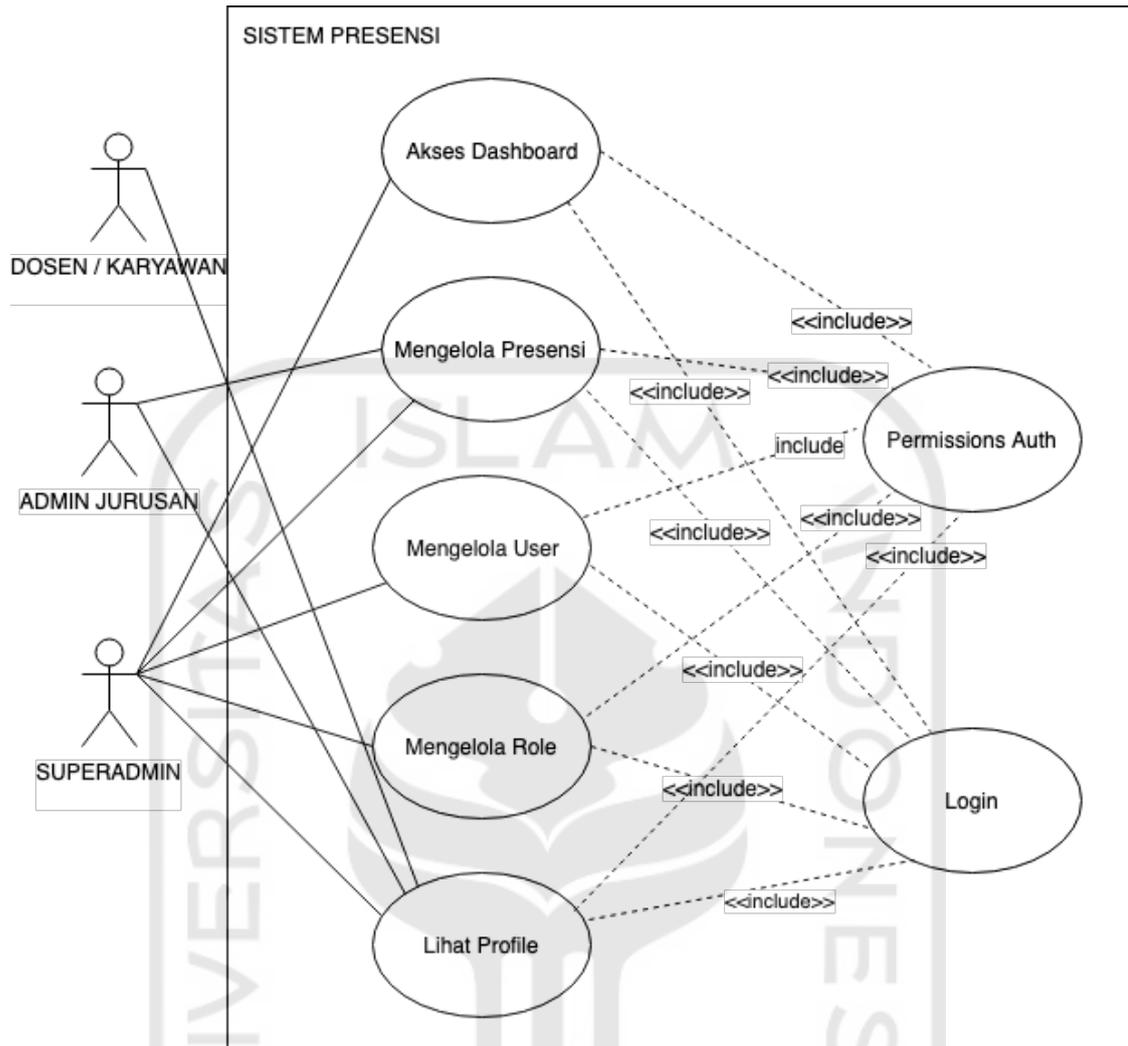
1. Sistem mengambil data dari *database log wifi*.
2. Sistem melakukan cek apakah pengguna mahasiswa atau bukan
3. Jika Mahasiswa, maka data tidak akan diproses dan langsung melakukan proses pada data selanjutnya
4. Jika bukan Mahasiswa, maka sistem akan melanjutkan pengecekan apakah data user sudah ada pada database
5. Jika belum ada, data user akan ditambahkan ke database sistem
6. Jika sudah ada, sistem melanjutkan pada proses menghitung durasi user tersebut pada suatu *access point*
7. Setelah menghitung durasi, data *log wifi* tersebut akan ditambahkan ke database yang nantinya akan dihitung sebagai presensi.
8. Jika masih terdapat data, proses diulang dari nomor satu



Gambar 3.1 Flow Chart Diagram

3.3.2 Use Case Diagram

Sistem Manajemen Presensi dirancang dengan menggunakan *Use Case Diagram*. *Use case Diagram* bertujuan untuk mengetahui fungsi pada sebuah sistem dan interaksinya. Untuk itu maka dibuat *Use Case Diagram* untuk memudahkan dalam perancangan yang ditunjukkan pada Gambar 3.2.



Gambar 3.2 Use Case Diagram

Gambar 3.2 menjelaskan rancangan fungsi sebuah sistem yang akan diimplementasikan pada bab berikutnya. Dari gambar diatas hanya terdapat tiga aktor yaitu dosen atau karyawan, admin jurusan, dan superadmin. Untuk aktor dosen atau karyawan hanya mampu melihat profil, aktor admin jurusan mampu mengelola presensi dan melihat profil, sedangkan aktor superadmin mampu mengakses semua proses. Use case diagram pada gambar 3.2 memiliki 6 use case yaitu:

- a. Fungsi Akses Dashboard

- b. Fungsi Mengelola Presensi
- c. Fungsi Mengelola User
- d. Fungsi Mengelola Role
- e. Fungsi Lihat Profile
- f. Fungsi Login

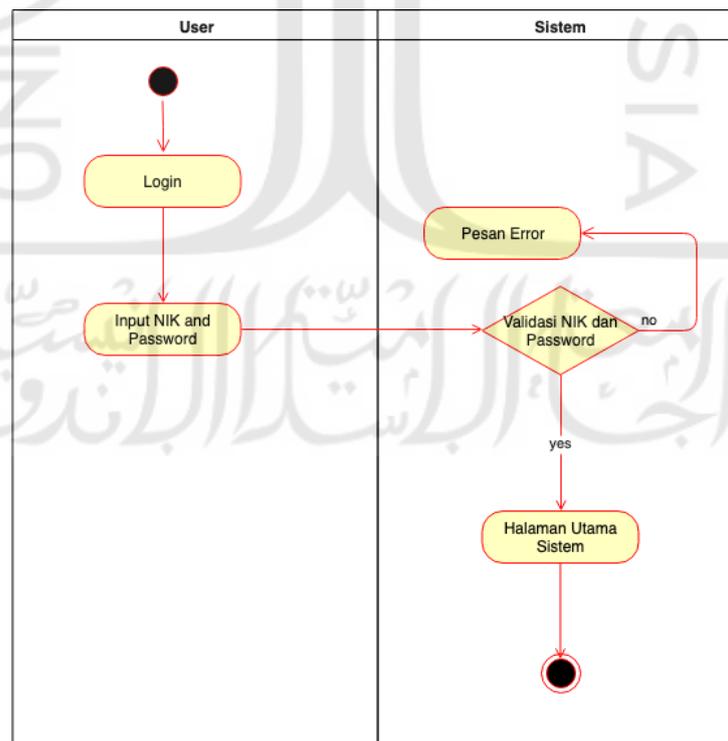
3.3.3 Activity Diagram

Activity Diagram adalah penjelasan secara garis besar proses bisnis dari sebuah sistem. Activity diagram menjelaskan langkah-langkah dari setiap alur menu yang tersedia pada sistem presensi. Diagram ini dapat digunakan untuk menganalisa setiap proses yang terjadi di dalam sistem secara lebih spesifik.

Terdapat enam activity diagram yang dihasilkan dengan menilik pada *Use Case Diagram* yang sudah dibuat sebelumnya, dibuat desain *activity diagram* sebagai berikut:

1. Halaman Login

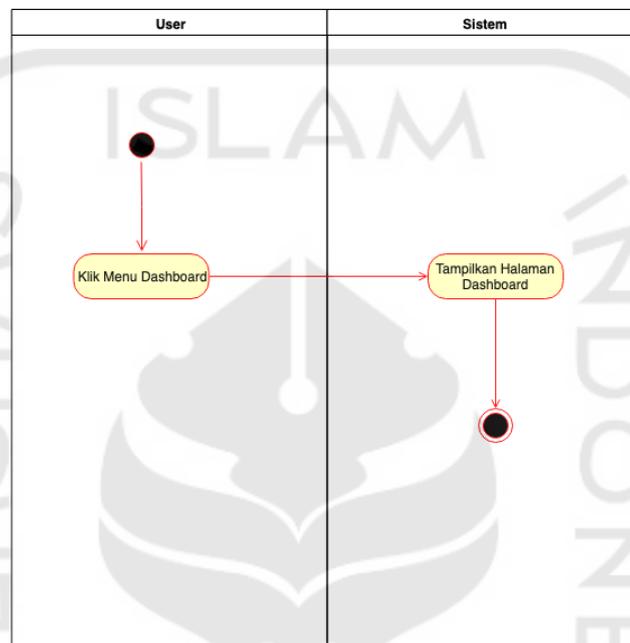
Terdapat activity diagram pada halaman login. Sebelum mengakses fitur didalam sistem pengguna harus melakukan validasi terlebih dahulu dengan cara menginputkan NIK dan Password. Pada Gambar 3.3 merupakan activity diagram untuk halaman login.



Gambar 3.3 *Activity Diagram* Halaman Login

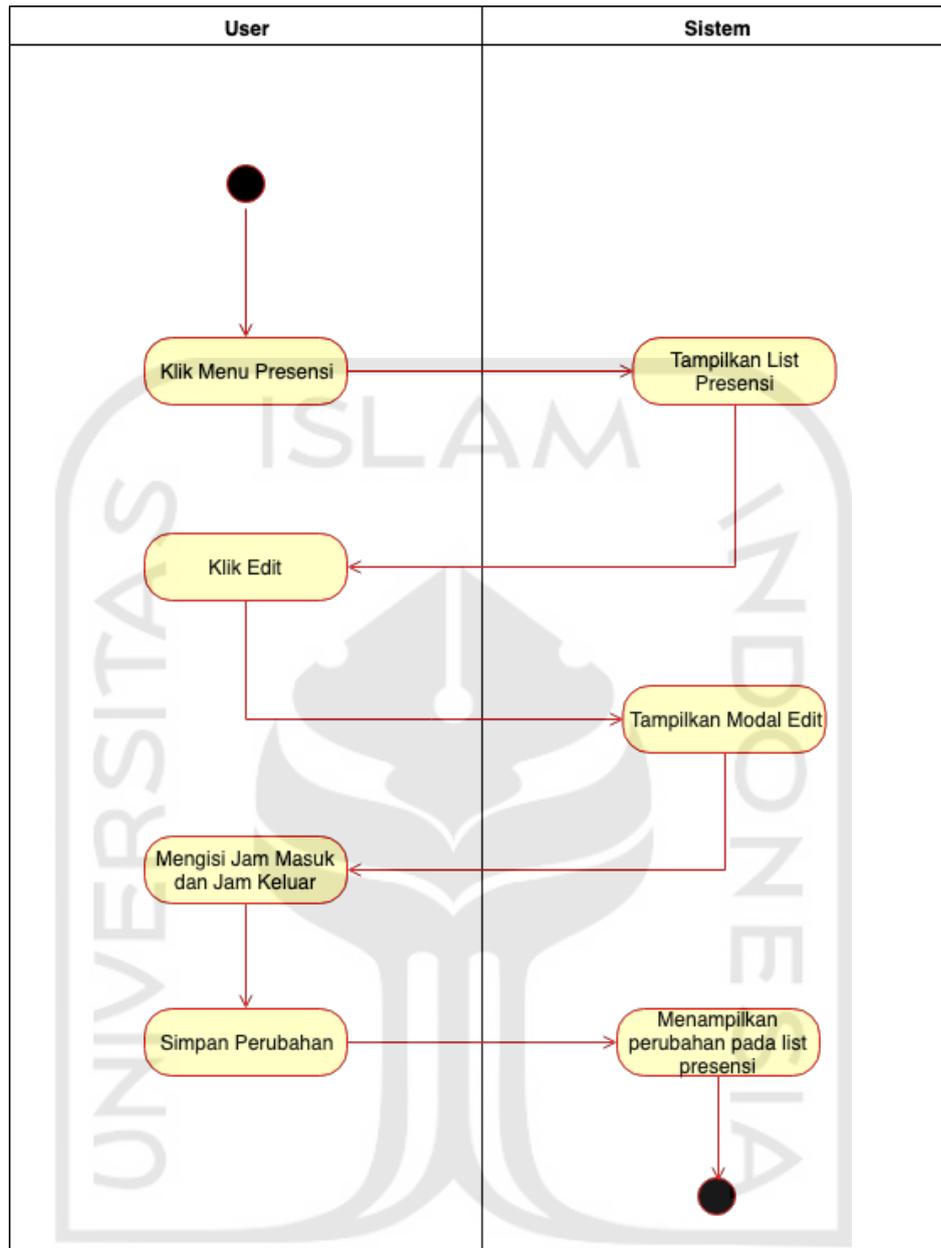
2. Halaman Dasbor

Gambar 3.4 menggambarkan fungsi untuk menampilkan halaman dasbor. Didalam halaman dasbor terdapat informasi terkait jumlah kehadiran hari ini dan juga grafiik kehadiran secara global.

Gambar 3.4 *Activity Diagram* Halaman Dasbor

3. Halaman Mengelola Presensi

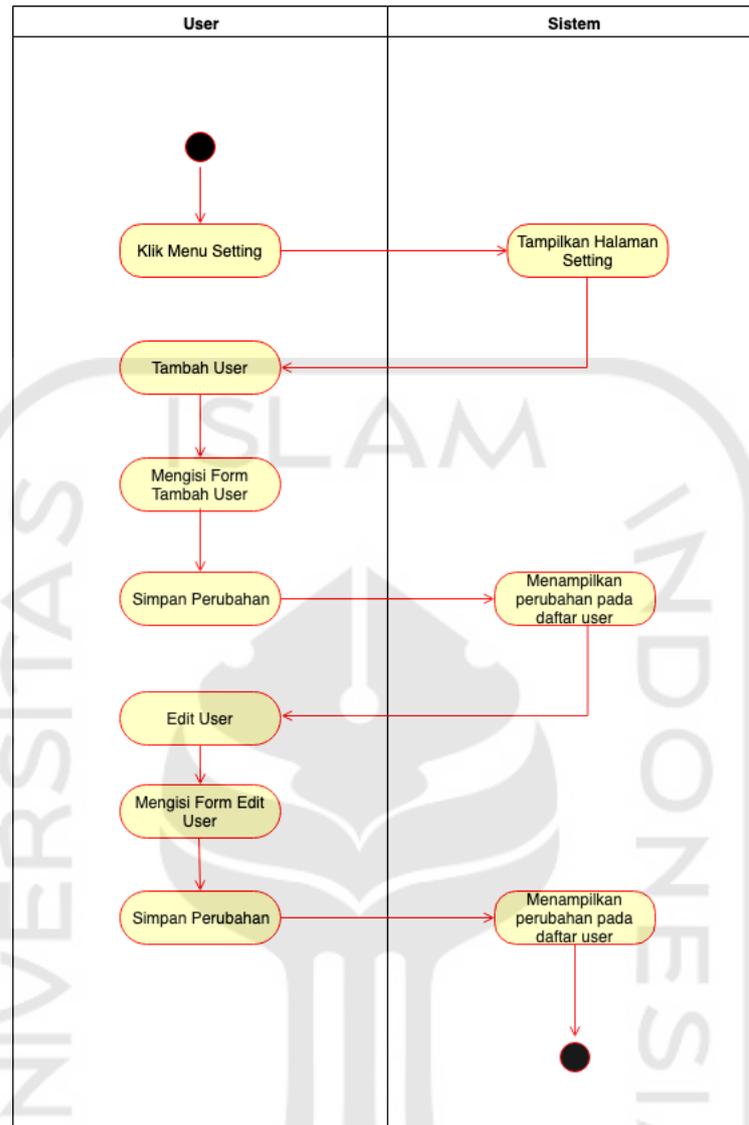
Gambar 3.5 menggambarkan fungsi untuk melakukan *override* pada presensi. Langkah pertama yang dilakukan adalah dengan cara membuka halaman presensi dan memilih presensi yang ingin diubah, selanjutnya klik tombol edit lalu akan tampil form edit untuk mengisikan form perubahan kemudian klik tombol simpan perubahan. Apabila presensi berhasil disimpan nantinya akan terlihat terjadi perubahan data pada halaman presensi ketika berhasil diupdate.



Gambar 3.5 *Activity Diagram* Halaman Presensi

4. Halaman Mengelola User

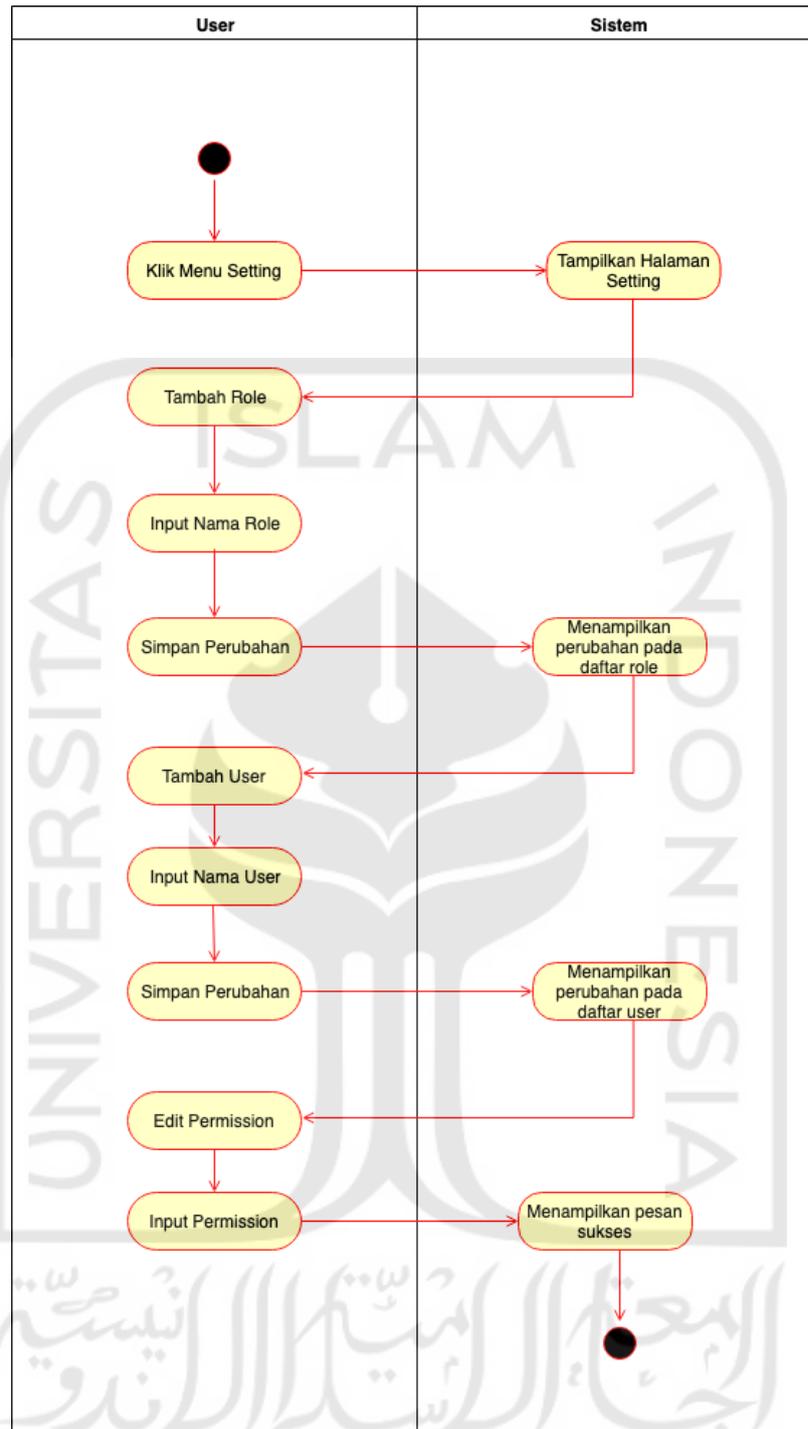
Gambar 3.6 merupakan *activity diagram* yang menunjukkan fungsi mengelola user. Pada halaman mengelola user, pengguna dapat menambahkan maupun mengedit data user.



Gambar 3.6 Activity Diagram Halaman Mengelola User

5. Halaman Mengelola Role

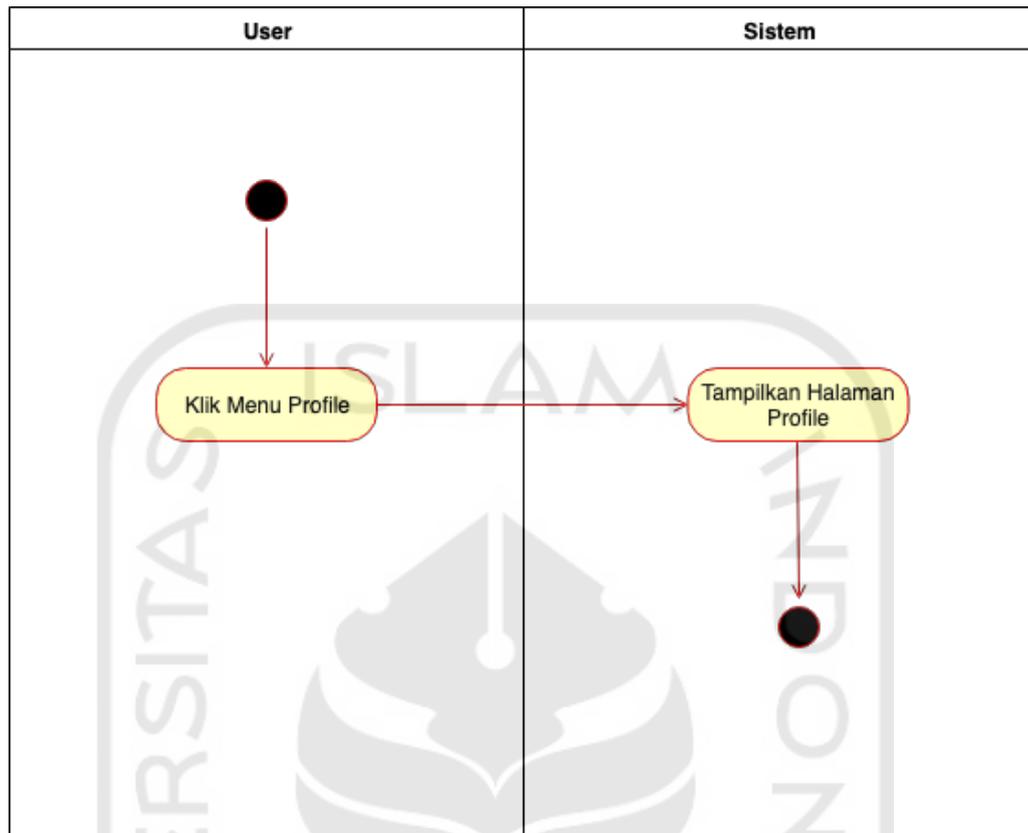
Gambar 3.7 merupakan *activity diagram* yang menunjukkan fungsi mengelola *role* pada sistem ini. Langkah pertama yang dilakukan adalah dengan cara klik tombol tambah *role* terlebih dahulu lalu kemudian pengguna akan memasukkan nama *role* yang akan dibuat. Setelah itu, pengguna dapat menambahkan user kedalam *role* tersebut. Terakhir, pengguna dapat mengedit *permission* yang ingin diaplikasikan ke *role* tersebut.



Gambar 3.7 Activity Diagram Halaman Mengelola Role

6. Halaman Lihat Profil

Fungsi Lihat Profil ditunjukkan pada Gambar 3.8 . Pada halaman profil terdapat kalender dimana terdapat gambaran terkait status kehadiran pada setiap tanggal. Selain itu, terdapat informasi terkait kemana dan kapan saja kita berada pada suatu tempat untuk tanggal tertentu.

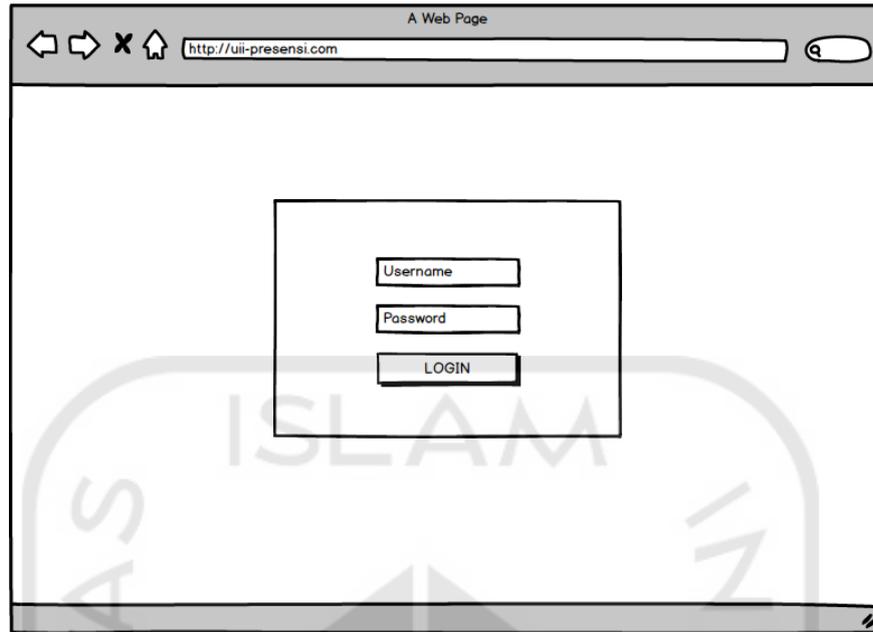


Gambar 3.8 *Activity Diagram* Halaman Profil

3.4 Prototyping

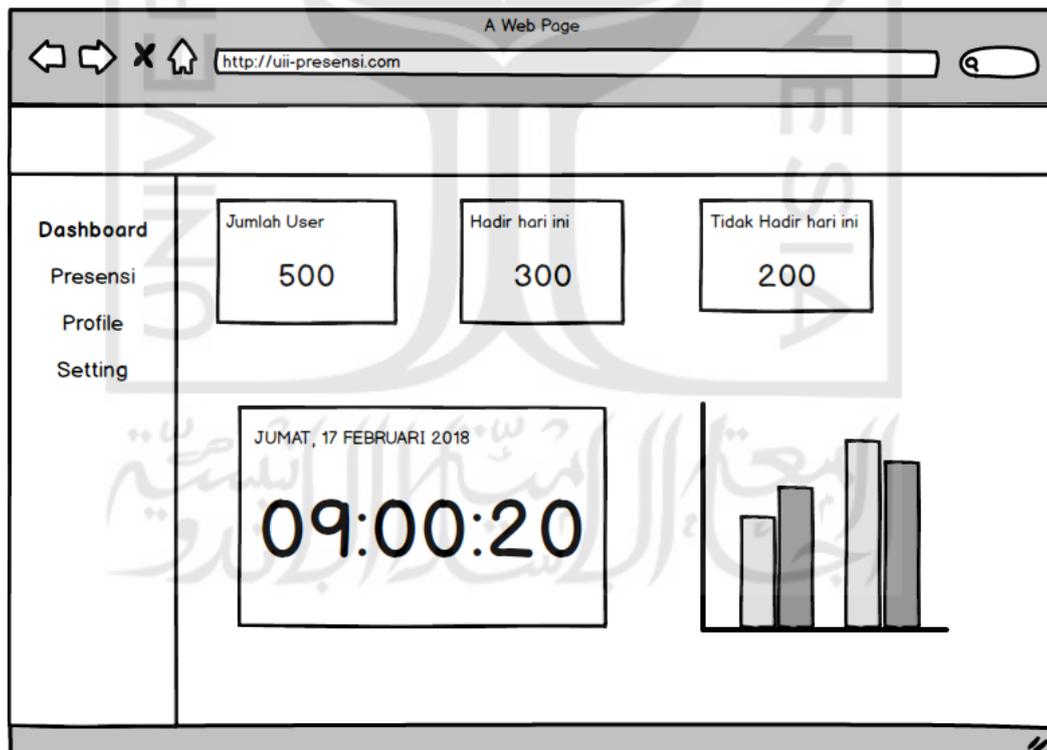
a. Desain Halaman Login

Gambar 3.9 merupakan tampilan form login untuk masuk kedalam sistem. User harus memasukkan username dan password untuk dapat melakukan pinjaman. Terdapat tombol Login untuk masuk ke dalam sistem.



Gambar 3.9 Desain Halaman Login

b. Desain Halaman Dasbor

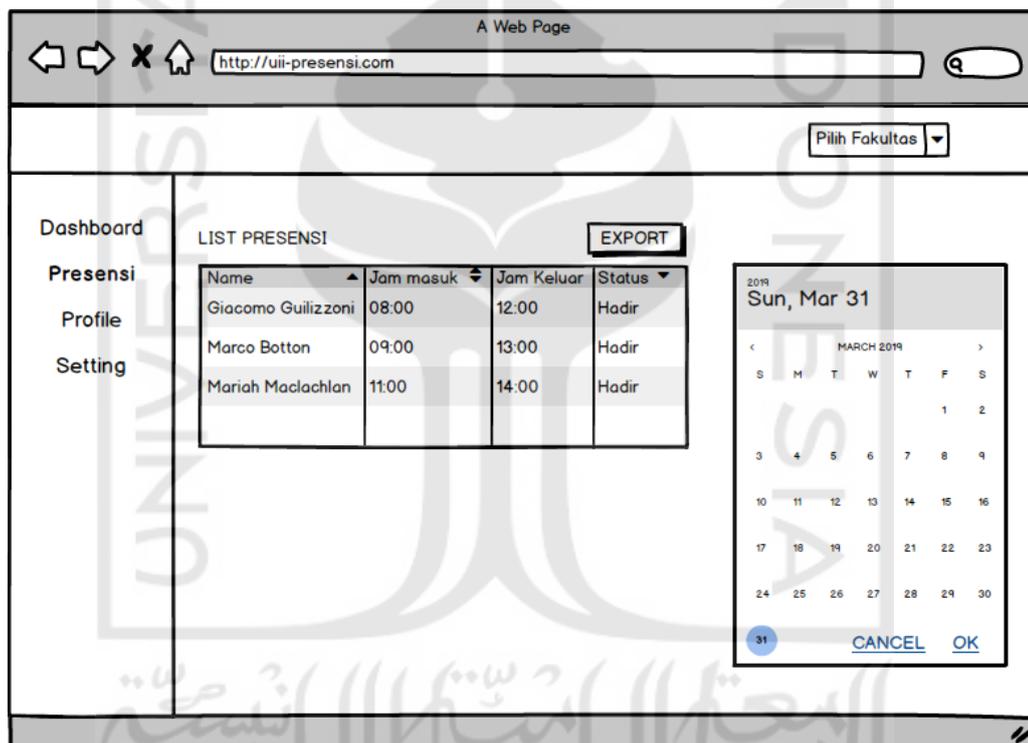


Gambar 3.10 Desain Halaman Dasbor

Halaman ini dapat diakses ketika user berhasil login. Pada gambar 3.10 merupakan desain tampilan halaman dashboard. Pada halaman dashboard memungkinkan untuk melihat jumlah user yang terdapat disistem baik user yang hadir maupun tidak. Selain itu ada juga grafik yang menunjukkan rata rata kehadiran dalam satu organisasi.

c. Desain Halaman Presensi

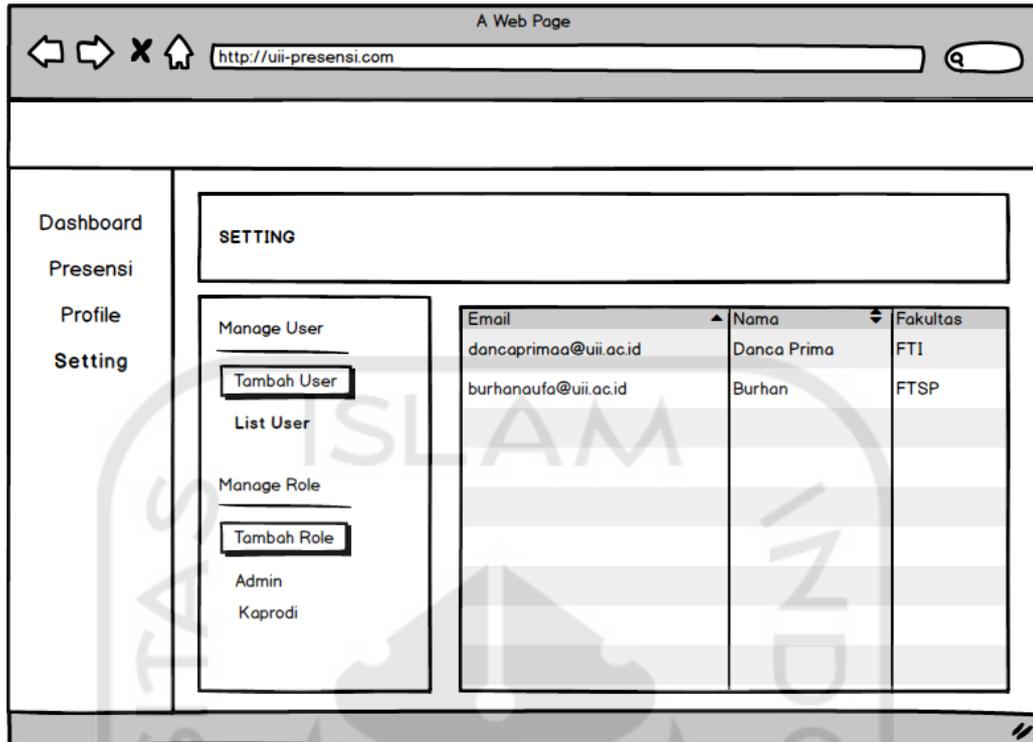
Pada Gambar 3.11 menunjukkan desain halaman presensi. Halaman ini menampilkan daftar pengguna yang hadir maupun tidak dan dapat disaring berdasarkan tanggal dan fakultas. Pada halaman ini terdapat informasi terkait nama karyawan atau dosen, jam masuk, jam keluar, dan status kehadiran.



Gambar 3.11 Desain Halaman Presensi

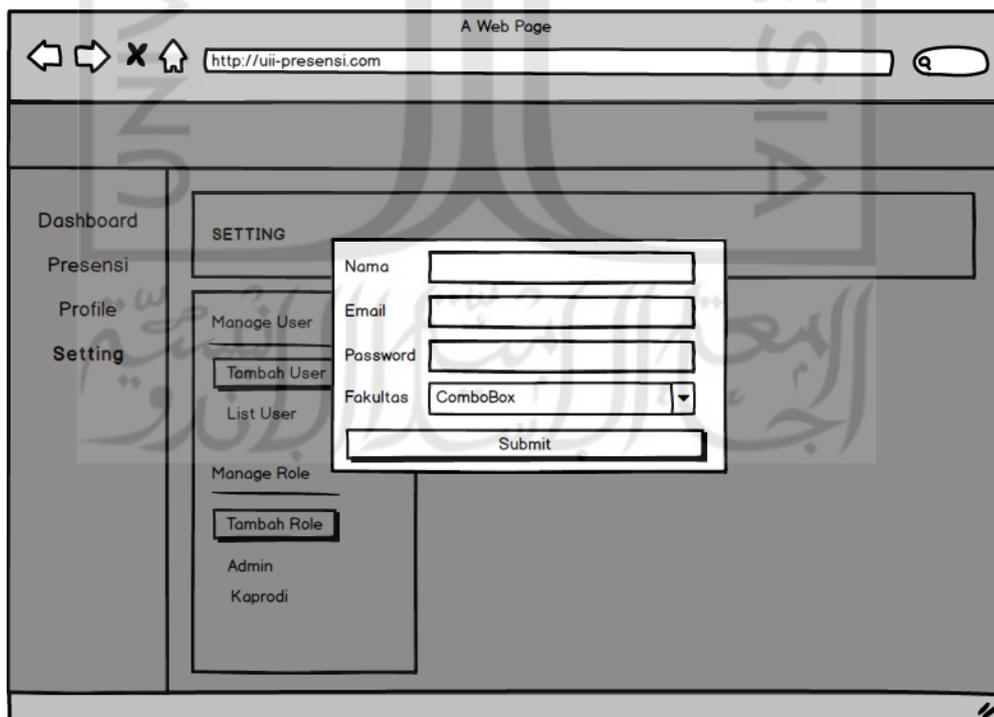
d. Desain Halaman User

Halaman User akan muncul ketika masuk pada menu setting. Pada halaman user terdapat informasi terkait email, nama, dan fakultas yang ditunjukkan pada Gambar 3.12.



Gambar 3.12 Desain Halaman User

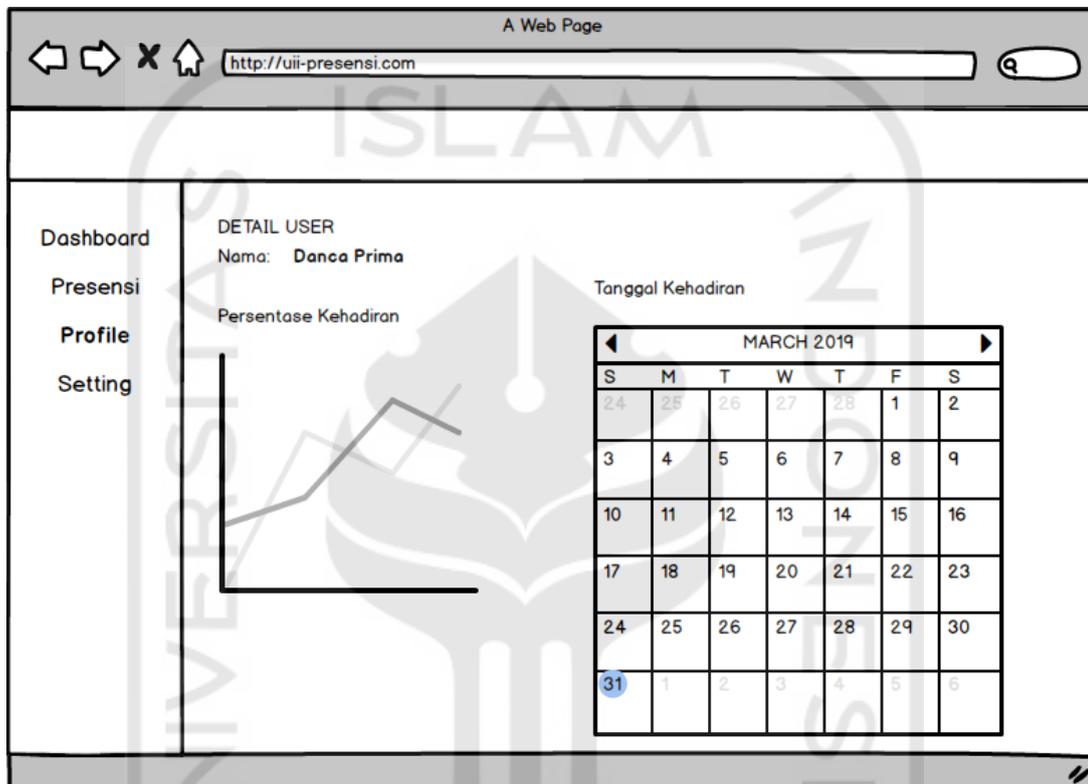
e. Desain Halaman Tambah User



Gambar 3.13 Desain Halaman Tambah User

Pada Gambar 3.13 merupakan desain halaman tambah user. Halaman ini merupakan halaman *pop up* dalam bentuk *form*. Terdapat beberapa yang harus diisi pada form tambah user yaitu, nama, email, password, dan fakultas. Setelah mengisi form, klik tombol submit.

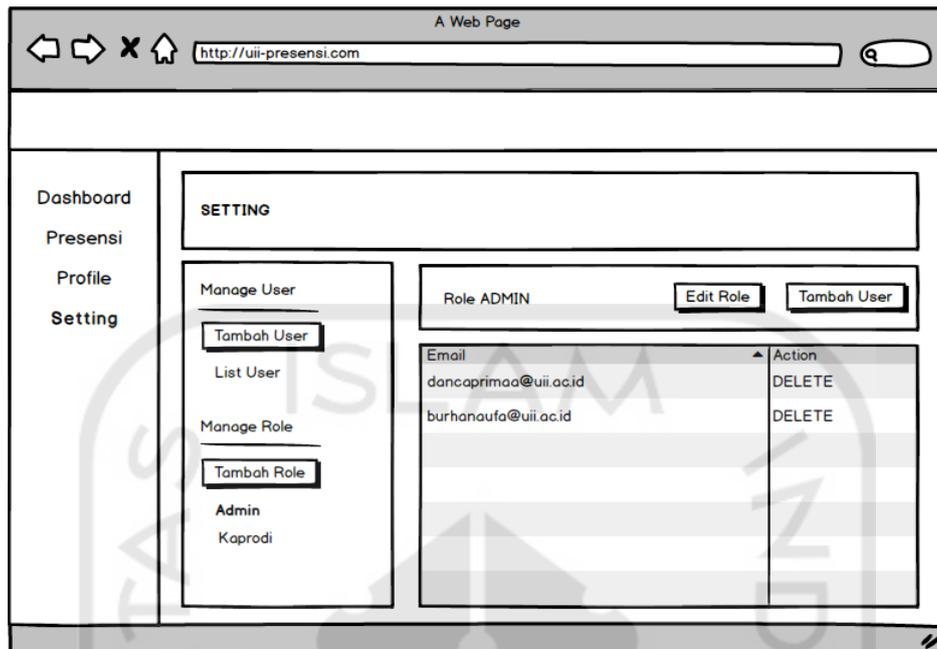
f. Desain Halaman Profil



Gambar 3.14 Desain Halaman Profil

Pada gambar 3.14 merupakan desain tampilan untuk halaman profil. Pada halaman ini terdapat grafik yang menunjukkan perbandingan persentase kehadiran suatu user dengan rata rata persentase kehadiran seluruh universitas. Selain itu, juga terdapat kalender yang nantinya akan merepresentasikan status kehadiran. Status kehadiran yang dimaksud meliputi hadir tepat waktu, tidak hadir, dan hadir terlambat.

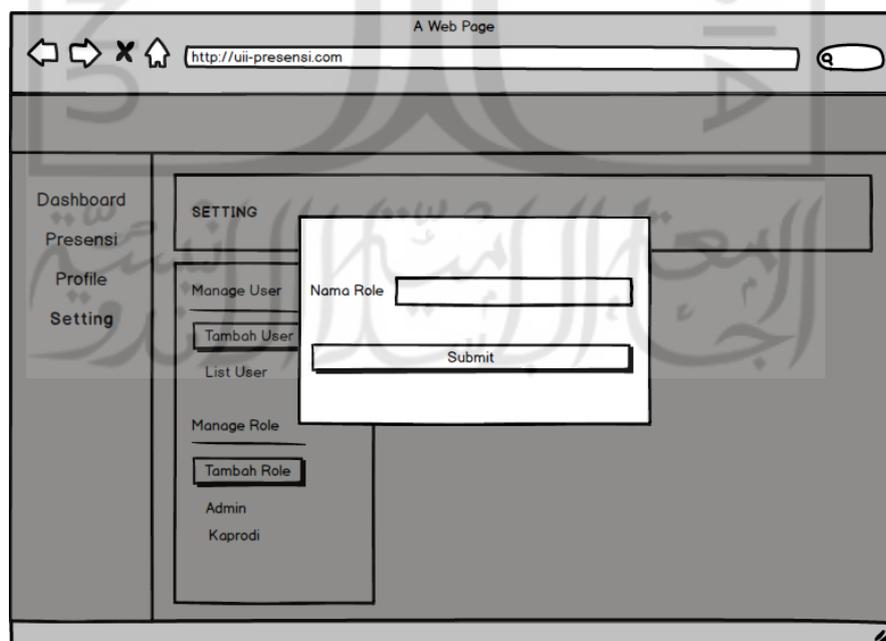
g. Desain Halaman Role



Gambar 3.15 Desain Halaman Role

Gambar 3.15 merupakan desain *prototype* untuk halaman role. Halaman role terdapat pada menu setting. Terdapat informasi terkait nama role dan daftar user yang memiliki role tersebut.

h. Desain Halaman Tambah Role

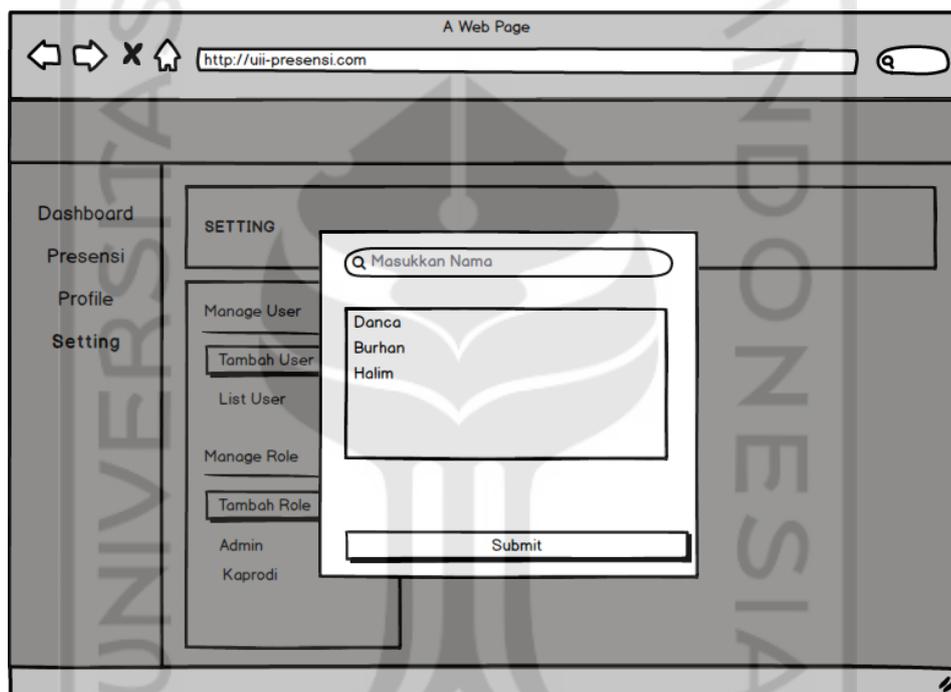


Gambar 3.16 Desain Halaman Tambah Role

Gambar 3.16 merupakan desain *prototype* untuk tampilan *form* tambah role. *Form* tersebut akan muncul ketika tombol tambah role pada halaman setting di klik. Setelah itu, kita diharuskan untuk mengisi nama role kemudian klik submit.

i. Desain Halaman Tambah *User Role*

Gambar 3.17 merupakan desain *prototype* tampilan *form* untuk menambahkan user pada role. Pada halaman ini memungkinkan kita untuk mencari user berdasarkan email dan menambahkannya pada role tertentu.



Gambar 3.17 Desain Halaman Tambah User Role

j. Desain Halaman *Edit Permission*

Gambar 3.18 merupakan desain *prototype* tampilan *form* untuk mengedit *permission* yang terdapat pada halaman role. Pada *form edit permission* terdapat daftar *menu* yang dapat dipilih sesuai dengan *permission* yang kita inginkan.



Gambar 3.18 Desain Halaman Edit Permission

3.5 Desain Pengujian

Pengujian adalah proses melakukan validasi, verifikasi, dan mendeteksi terjadinya error pada suatu sistem. Proses tersebut diharapkan dapat menemukan kesalahan pada sebuah sistem yang nantinya dapat diperbaiki. Proses-proses yang telah dijelaskan diatas memiliki pengertian sebagai berikut:

- Verifikasi yaitu proses untuk menentukan sistem tersebut sudah berjalan dengan benar atau tidak.
- Validasi yaitu proses untuk menentukan sistem telah sesuai dengan tujuan yang diharapkan.
- Deteksi yaitu proses untuk menentukan sistem mampu mendeteksi error atau kesalahan pada sistem.

Selain itu, pengujian ini dilakukan dengan metode *Blackbox*. Pada metode *Blackbox* pengujian hanya berfokus pada fungsional sistem. Dengan menerapkan metode ini terdapat beberapa kategori *error* yang akan diketahui, yaitu:

- Eksekusi fungsi yang salah.
- *Error* dari antar-muka.
- *Error* dari akses *database*.
- *Error* dari kinerja atau tingkah laku.
- *Error* dari inisiasi dan terminasi.

Dalam sebuah pengujian dibutuhkan *test case*. *Test case* atau juga bisa disebut dengan uji kasus adalah suatu rancangan atau rangkaian mengenai tindakan yang dilakukan oleh user untuk melakukan verifikasi terhadap fitur atau fungsi tertentu dari sebuah perangkat lunak. Rancangan tersebut nantinya digambarkan dalam bentuk dokumen. Dokumen uji kasus yang digunakan pada penelitian ini ditunjukkan pada Tabel 3.2. Pada uji kasus ini peneliti mengambil sampel pengguna dengan *role Admin*.

Tabel 3.2 Tabel Uji Kasus

Uji Kasus	Deskripsi Uji Kasus	Hasil yang diharapkan
Login	Pengguna login dengan memasukkan Nomor Induk Pegawai dan Password	Setelah login sukses pengguna diarahkan menuju halaman dasbor
Lihat Halaman Presensi	Pengguna pindah menuju halaman presensi dengan cara klik menu presensi pada sidebar lalu akan muncul daftar karyawan/dosen.	Pengguna dapat melihat data yang benar sesuai dengan penyaringan yang dilakukan dan juga pada saat terjadi perpindahan halaman.
Edit Rekam Hadir	Pengguna dapat melakukan edit terhadap rekam hadir dengan cara memasukkan tanggal, jam masuk, dan jam keluar.	Setelah sukses mengedit pengguna kembali pada halaman presensi dan dapat langsung melihat data rekam hadir yang telah diperbarui.

Lihat Ranking	Pengguna dapat melihat daftar urutan karyawan/dosen berdasarkan performa kerja yang dihitung dari jumlah hadir, terlambat, dan tidak hadir.	Pada halaman ini akan menampilkan urutan karyawan/dosen berdasarkan terbaik atau terburuk sesuai dengan fakultas dan bulan tertentu.
Unduh data presensi	Pengguna dapat mengunduh data rekap presensi dengan format excel berdasarkan fakultas, bulan, dan tahun.	Pengguna sukses mengunduh file dengan format excel dengan data yang benar.
Lihat Detail/Profil	Pengguna dapat melihat statistik terkait perbandingan rata rata kehadiran dosen atau karyawan tertentu dibandingkan seluruh Universitas. Selain itu pengguna juga dapat melihat data kehadiran dalam bentuk kalender.	Sukses menampilkan status kehadiran dan statistik dengan benar .
Lihat Halaman User	Pengguna dapat melihat daftar karyawan atau dosen yang terdaftar pada sistem.	Menampilkan daftar pengguna yang terdaftar oleh sistem yang diurutkan berdasarkan email.
Tambah User	Pengguna dapat menambahkan karyawan atau dosen pada sistem.	Sukses menambahkan data karyawan atau dosen pada sistem.

Edit User	Pengguna dapat mengedit data nama, password, dan fakultas karyawan atau dosen pada sistem.	Sukses mengedit semua data karyawan atau dosen mulai dari nama, password, fakultas dan membatasi pengguna untuk mengubah data email pada sistem.
Tambah Role	Pengguna dapat menambahkan role pada sistem.	Sukses menambahkan role pada sistem yang ditandai dengan bertambahnya role pada sidebar setting.
Edit Permission pada Role	Pengguna dapat mengedit permission pada role yang sudah ditambahkan.	Sukses menambahkan permission pada role tertentu.
Tambah Role pada Karyawan atau Dosen	Pengguna dapat menambahkan role pada dosen atau karyawan.	Sukses menambahkan role pada user yang ditandai dengan munculnya email pada tabel user di halaman role ketua yayasan.
Hapus Role pada Karyawan atau dosen	Pengguna dapat menghapus role pada dosen atau karyawan.	Sukses menghapus role pada user ditandai dengan hilangnya email yang dihapus pada tabel user di halaman role ketua yayasan.