

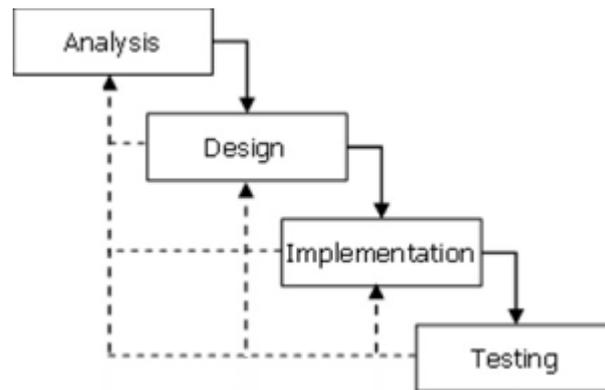
BAB III

METODE PENELITIAN

3.1 Metode Pengembangan Sistem

Untuk pengembangan sistem, penelitian ini menggunakan model SDLC (*Software Development Life Cycle*). Selain untuk proses pembuatan, SDLC juga penting untuk proses *maintenance software* itu sendiri. SDLC juga merupakan pola yang diambil untuk mengembangkan sistem perangkat lunak, yang terdiri dari tahap – tahap: rencana (*planning*), analisis (*analysis*), desain (*design*), implementasi (*implementation*), dan uji coba (*testing*).

Model SDLC yang dipakai dalam penelitian ini adalah model *Waterfall* yang mengadopsi dari model *waterfall* yang dikemukakan oleh Youssef Bassil dalam jurnalnya yang berjudul “*A Simulation Model for the Waterfall Software Development Life Cycle*”. Akan tetapi tahapan model SDLC ini hanya sampai pada tahap *testing*, dikarenakan pada penelitian ini tidak sampai pada tahap *maintenance*. *Waterfall Model* atau *Classic Life Cycle* merupakan model yang paling banyak dipakai dalam *Software Engineering* (SE). Disebut *waterfall* karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan (Bassil, 2012). Untuk lebih jelasnya dapat dilihat pada gambar 3.1.



Gambar 3.1 Metode Waterfall menurut Bassil (2012)

3.1.1 Analysis

Sebelum melakukan analisis, penulis terlebih dahulu melakukan observasi dan wawancara terhadap beberapa mahasiswa Universitas Muhammadiyah Yogyakarta. Berdasarkan hasil observasi dan wawancara, peneliti menganalisis data dari permasalahan *user*. Apakah kebutuhan fungsi *software* untuk memenuhi kendala yang dialami oleh *user*.

3.1.2 Design

Setelah peneliti mendapatkan dokumen hasil dari analisis, maka peneliti mengubah kebutuhan-kebutuhan fungsi *software* diatas menjadi sebuah bentuk “*blueprint*” *software*. Sehingga hasil desain akan digunakan oleh peneliti untuk membangun sebuah aplikasi.

3.1.3 Implementation

Untuk dapat dimengerti oleh mesin, dalam hal ini adalah komputer, maka desain tadi harus diubah bentuknya menjadi bentuk yang dapat dimengerti oleh

mesin, yaitu ke dalam bahasa pemrograman melalui proses *coding*. Bahasa pemrograman yang digunakan adalah Bahasa pemrograman *java* dan *xml*.

Pada tahap ini, peneliti membangun sebuah aplikasi berdasarkan desain “*blueprint*” yang telah dibuat. Pengembangan aplikasi ini dilakukan dari awal hingga aplikasi siap dijalankan. Dari fungsi-fungsi yang dibutuhkan hingga tampilan untuk pengguna (mahasiswa).

3.1.4 Testing

Dalam tahap testing dilakukan pengujian *software* yang sudah dibuat. Semua fungsi-fungsi *software* harus diuji coba agar *software* bebas dari *error* atau kesalahan dan hasilnya harus benar-benar sesuai dengan kebutuhan yang sudah didefinisikan sebelumnya. Aplikasi diuji berdasarkan metode *black box* untuk mengetahui tingkat keberhasilan dari bagian sistem. Selain itu, peneliti juga melakukan pengujian secara langsung pada setiap sesi perkuliahan di UMY (khususnya di Jurusan Teknik Informatika).

3.2 Alat dan Bahan

3.2.1 Alat

Alat yang digunakan dalam penelitian ini mencakup perangkat keras dan perangkat lunak..

a. Perangkat Keras Untuk Pengembang

Spesifikasi laptop yang digunakan untuk pengembang dapat dilihat pada tabel

3.1.

Tabel 3.1 Spesifikasi Laptop Pengembang

Sistem Operasi	Windows 10 Pro
Processor	Intel® Core™ i5-2450M Processor 2.50 GHz with Turbo Boost up to 3.10 GHz
RAM	4 GB
Hard Disk	500 GB

b. Perangkat Keras Untuk User (Mahasiswa)

Spesifikasi minimum perangkat *smartphone* berbasis *android* yang dapat digunakan oleh mahasiswa untuk menjalankan aplikasi ini dapat dilihat pada tabel 3.2.

Tabel 3.2 Spesifikasi Minimum Perangkat *Smartphone User* (Mahasiswa)

Sistem Operasi	Android 4.1 Jelly Bean
RAM	512 MB
Internal Memori	8 GB

Spesifikasi minimum untuk *user* diharuskan seperti yang terdapat pada tabel 3.2 dikarenakan terdapat beberapa interface yang hanya bisa berjalan pada sistem operasi *android 4.1 jelly bean* keatas, dan untuk dapat menjalankan sistem operasi *android 4.1 jelly bean* diperlukan minimum *ram* sebesar 512 *mb* dan internal memori sebesar 8 *gb*.

c. Perangkat Lunak Untuk Pengembang

Daftar perangkat lunak yang digunakan pengembang dapat dilihat pada tabel 3.3.

Tabel 3.3 Daftar Perangkat Lunak

No	Perangkat Lunak	Fungsi
1.	Android Developer Tools, Build: v21.1.0-569685	Digunakan untuk pembuatan aplikasi presensi berbasis <i>android</i> .
2.	SQL Server Management Studio 2014	Digunakan untuk pengolahan <i>database</i> yang akan digunakan dalam aplikasi presensi.
3.	Balsamiq Mockup	Digunakan untuk pembuatan perancangan antarmuka aplikasi.
4.	Microsoft Visio 2013	Digunakan untuk pembuatan perancangan sistem.

3.2.2 Bahan

Bahan – bahan yang digunakan dalam penelitian ini adalah:

1. Data yang diperoleh melalui studi literatur berdasarkan penelitian sebelumnya yang masih memiliki keterkaitan dengan aplikasi yang dikembangkan.

2. *Database UMY* (data yang digunakan adalah data – data yang mendukung untuk proses presensi) dari Biro Sistem Informasi (BSI) Universitas Muhammadiyah Yogyakarta.

3.3 Metode Pengumpulan Data

3.3.1 Observasi

Tujuan dari observasi ini adalah untuk mengetahui sistem yang digunakan untuk presensi mahasiswa yang sedang berjalan sebelumnya. Observasi dapat dibedakan menjadi *participant observation* (observasi berperan serta) dan *non-participant observation* (observasi nonpartisipan).

Observasi yang dilakukan oleh peneliti adalah jenis observasi berperan serta (*participant observation*), dalam observasi ini, peneliti terlibat langsung dalam objek yang diamati atau yang digunakan sebagai sumber data penelitian. Pada penelitian ini peneliti mendapatkan data dengan mengamati objek yang akan diteliti secara langsung di tempat penelitian yaitu Biro Sistem Informasi (BSI) Universitas Muhammadiyah Yogyakarta dan pada saat mengikuti perkuliahan didalam kelas. Pengumpulan data meliputi sistem pengumpulan data presensi yang digunakan di Universitas Muhammadiyah Yogyakarta.

3.3.2 Wawancara

Wawancara merupakan salah satu metode pengumpulan data dengan cara tanya jawab sambil bertatap muka antara peneliti dengan narasumber. Proses wawancara ini dilakukan di Biro Sistem Informasi (BSI) Universitas Muhammadiyah Yogyakarta untuk mendapatkan data dengan melakukan tanya

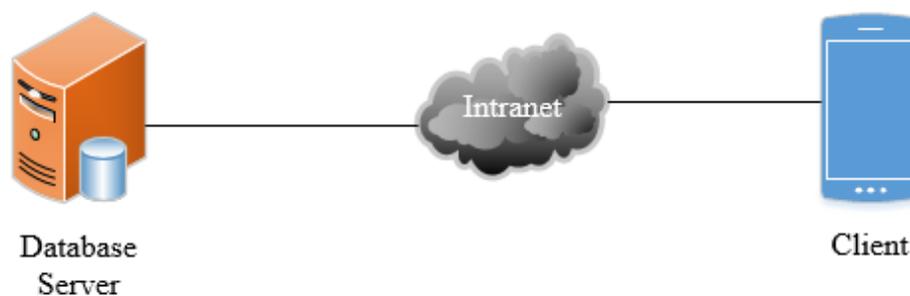
jawab langsung dengan Kepala Bagian Urusan Aplikasi (BSI) Universitas Muhammadiyah Yogyakarta Bapak Miftahul Yaum, S.Kom., yang dianggap dapat memberikan informasi yang dibutuhkan oleh peneliti.

Tujuan melakukan wawancara ini adalah untuk mengetahui sistem presensi mahasiswa yang sedang berjalan saat ini dan bagaimana proses mahasiswa melakukan presensi pada Universitas Muhammadiyah Yogyakarta dari segi arsitektur aplikasi dan teknologi yang digunakan serta pemanfaatannya.

3.4 Analisis Data

3.4.1 Arsitektur

Dari hasil observasi, diketahui bahwa Universitas Muhammadiyah Yogyakarta mempunyai banyak *server*. Setiap *server* mempunyai data-data dan peruntukkan yang berbeda dengan *server* yang lain. Setiap sistem informasi yang ada mempunyai *server* sendiri, sehingga peneliti mendapatkan sebuah analisis rancangan arsitektur seperti pada gambar 3.2.



Gambar 3.2 Arsitektur aplikasi

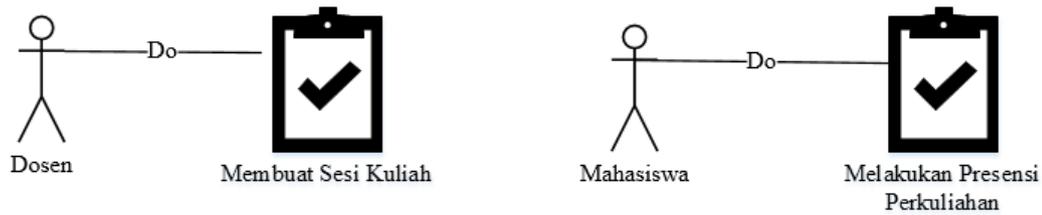
Database yang digunakan pada *database server* adalah Microsoft SQL Server 2014, kemudian dihubungkan melalui *intranet*. Aplikasi *mobile* yang digunakan oleh *user* adalah aplikasi *android*. Saat *user* mengakses aplikasi *android*, aplikasi akan memuat antarmuka aplikasi *android* dan melakukan pengambilan data dari *database server*. Melalui antarmuka aplikasi *android*, *user* dapat menambahkan data dan disimpan kedalam *database*.

3.4.2 Analisis Pengguna

Berdasarkan hasil wawancara dengan bagian BSI (Biro Sistem Informasi) dan observasi di Universitas Muhammadiyah Yogyakarta, proses presensi mahasiswa yang diterapkan sebelumnya adalah sebagai berikut:

- 1) Staf membuat berita acara sesi perkuliahan.
- 2) Mahasiswa melakukan presensi dengan memberi tanda tangan pada lembar berita acara sesi perkuliahan.
- 3) Staf merekap data presensi berdasarkan lembar berita acara sesi perkuliahan pada aplikasi presensi.

Dari beberapa tahapan proses presensi yang ada, peneliti dapat menyimpulkan bahwa kegiatan presensi dapat dilakukan dengan mengurangi proses yang sebenarnya dapat dilakukan sekaligus. Sehingga dalam analisis ini peneliti memperoleh rancangan proses presensi seperti pada gambar 3.3.



Gambar 3.3 Analisis pengguna

3.4.3 Analisis Pembuatan Sesi Perkuliahan

Pada proses presensi yang sedang berjalan saat ini, pembuatan berita acara perkuliahan dilakukan oleh staf pada masing-masing jurusan. Sebenarnya proses ini dapat diperbaiki dengan cara dosen yang mengisi perkuliahan yang membuat sesi perkuliahan, sehingga dapat mengurangi kerja staf yang dapat memperlambat proses presensi sampai data terekap. Peneliti dalam hal ini membuat sebuah rancangan seperti pada gambar 3.4.



Gambar 3.4 Analisis Pembuatan Sesi Perkuliahan

Penjelasan tentang Gambar 3.4 adalah sebagai berikut:

1. Pertama dosen mengakses website presensi mahasiswa *online* UMY, dan *login* berdasarkan email.
2. Dosen membuat sesi perkuliahan berdasarkan tahun ajaran, jurusan, matakuliah yang diampu dan kelas yang diampu.

3. Setelah membuat sesi perkuliahan, dosen akan masuk ke halaman detail sesi perkuliahan. Pada halaman detail sesi perkuliahan terdapat beberapa data tentang sesi perkuliahan yang telah dibuat seperti: tahun ajaran, kelas program, mata kuliah, kelas, ruangan, waktu mulai, waktu selesai, maksimal keterlambatan, isi perkuliahan, deskripsi dan *token*.
4. Kemudian dosen akan memberi kode *token* kepada mahasiswa untuk melakukan proses presensi.

3.4.4 Analisis Presensi Mahasiswa

Proses presensi mahasiswa yang dijalankan saat ini adalah dengan memberikan tanda tangan pada lembar berita acara perkuliahan yang diberikan oleh dosen pada saat proses kegiatan belajar mengajar berlangsung. Akan tetapi data mahasiswa yang hadir dan memberikan tanda tangan pada lembar berita acara perkuliahan tersebut tidak langsung terekap ke dalam *database*, melainkan harus direkap terlebih dahulu oleh staf dan kemudian dimasukkan kedalam aplikasi presensi yang hanya dimiliki oleh staf. Pada kondisi ini, proses presensi masih memakan waktu yang cukup lama. Sehingga peneliti membuat sebuah rancangan yang data mahasiswa yang melakukan presensi langsung terekap pada *database* setelah mahasiswa melakukan presensi seperti pada gambar 3.5.



Gambar 3.5 Analisis presensi mahasiswa

Penjelasan tentang Gambar 3.5. adalah sebagai berikut:

1. Mahasiswa memperoleh sebuah *token* dari dosen berdasarkan sesi yang telah dibuat.
2. Kemudian mahasiswa akan masuk ke aplikasi *android* untuk melakukan *login* berdasarkan nim dan password sesuai dengan *KRS-Online*
3. Mahasiswa melakukan validasi *token* yang sudah diberikan oleh dosen sebelumnya.
4. Setelah validasi berhasil aplikasi akan menampilkan detail dari perkuliahan yang akan atau sedang berjalan.
5. Setelah itu mahasiswa melakukan presensi terhadap mata kuliah yang akan atau sedang berjalan.

3.4.5 Analisis Kebutuhan Sistem

Analisis kebutuhan didapat dari studi *literature* dan diskusi bersama staf BSI UMY. Berikut ini merupakan analisis kebutuhan pada aplikasi presensi UMY:

1. Sistem harus terjamin keamanannya. Pada aplikasi ini menggunakan *Two Factor Authentication* untuk menjaga keamanannya, dimana faktor pertama adalah *login* menggunakan nim dan *password* mahasiswa sesuai dengan krs dan yang kedua adalah dengan menggunakan *security token*. Proses pembuatan *security token* telah dibuat dalam penelitian sebelumnya oleh Wicaksono, A. (2016) dengan judul “APLIKASI PENGELOLAAN PRESENSI MAHASISWA UMY BERBASIS *TOKEN*”. Hal ini diperlukan

agar sistem hanya dapat digunakan oleh mahasiswa UMY dan telah melakukan *key-in* sebelumnya.

2. Sistem harus dapat menerapkan satu *device* untuk satu *user*. Hal ini diperlukan agar mahasiswa tidak dapat melakukan titip absen.
3. Sistem hanya dapat digunakan ketika sedang berada pada jaringan UMY. Ini diperlukan untuk mempersulit mahasiswa yang ingin melakukan kecurangan dalam hal presensi (bagi mahasiswa yang ingin melakukan titip absen).
4. Halaman *login* mahasiswa yang akan melakukan presensi. Halaman login diperlukan untuk mengetahui mahasiswa yang menggunakan aplikasi adalah benar mahasiswa UMY atau bukan.
5. Halaman validasi *token* untuk memasukkan *token* yang diberikan oleh dosen.
6. Halaman tampilan detail sesi perkuliahan untuk memastikan bahwa sesi perkuliahan adalah benar dan untuk melakukan presensi.

3.5 Rancangan Sistem

Rancangan sistem adalah suatu tahapan kegiatan yang dilakukan seseorang atau kelompok dalam merancang atau membuat sistem sebelum sistem dibuat dengan tujuan sistem yang dibangun sesuai dengan kebutuhan dalam memecahkan atau dengan kebutuhan pengguna berkaitan dengan pengolahan, pengelolaan dan perolehan informasi yang diinginkan. Rancangan yang dibuat pada penelitian ini adalah rancangan sistem dan rancangan antarmuka pengguna. Berikut beberapa rancangan pada penelitian ini:

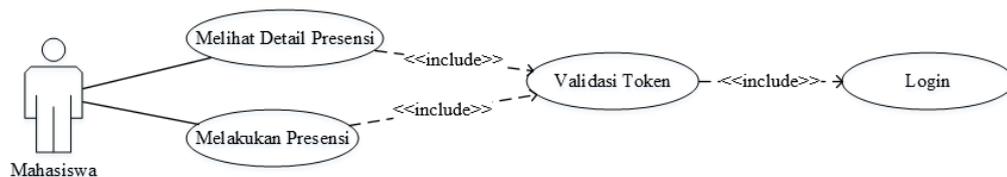
3.5.1 Rancangan Sistem

Dalam pembuatan aplikasi *android* dilakukan perancangan sistem menggunakan UML (*Unified Modeling Language*). Jenis UML yang digunakan dalam perancangan aplikasi android ini antara lain *Use Case Diagram* dan *Activity Diagram*. Perancangan dari jenis UML tersebut dapat dijelaskan sebagai berikut:

a. *Use Case Diagram*

Use Case Diagram merupakan salah satu jenis diagram UML yang menggambarkan interaksi yang mungkin terjadi antara sistem dan aktor. Berdasarkan analisis kebutuhan dibuatlah *use case diagram* untuk membantu perancangan aplikasi.

Gambaran *Use Case Diagram* yang digunakan pada aplikasi dapat dilihat pada gambar 3.6.

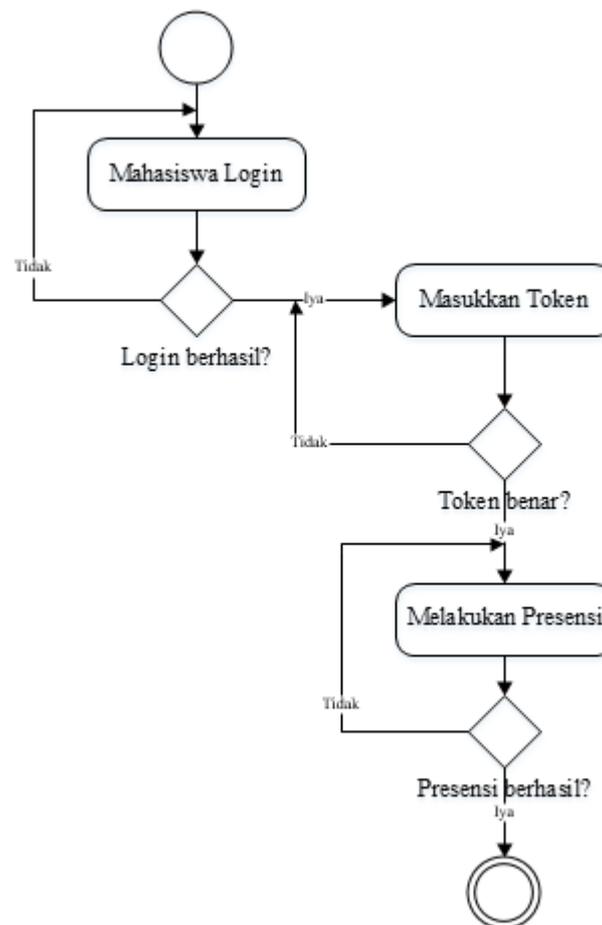


Gambar 3.6 *Use case diagram*

Pada gambar 3.6 menjelaskan bahwa aplikasi hanya digunakan oleh mahasiswa. Mahasiswa dapat melakukan validasi *token* setelah melakukan proses *login*. Kemudian setelah melakukan validasi *login*, mahasiswa dapat melihat detail presensi, dan yang yang terakhir mahasiswa dapat melakukan presensi setelah melakukan validasi *token*.

b. Activity Diagram

Berdasarkan *use case* yang telah dibuat maka dapat diperoleh *activity diagram* berdasarkan aktor yang terlibat dalam *use case diagram*. Untuk lebih jelasnya dapat dilihat pada gambar 3.7.



Gambar 3.7 Activity diagram

Berikut penjelasan tentang langkah-langkah pada Gambar 3.7:

- 1) Terdapat beberapa proses ketika Mahasiswa membuka aplikasi presensi *online*. Mahasiswa diharuskan untuk melakukan *login* terlebih dahulu.

- 2) Jika *nim* dan *password* benar maka akan berpindah ke halaman validasi *token*, jika salah maka sistem akan meminta untuk memasukkan *username* dan *password* kembali
- 3) Setelah masuk ke halaman validasi *token*, mahasiswa diharuskan untuk memasukkan *token* yang sudah diberikan oleh dosen.
- 4) Jika *token* yang dimasukkan benar, maka aplikasi akan menampilkan detail presensi, dan jika *token* salah maka mahasiswa diharuskan untuk memasukkan *token* kembali.
- 5) Setelah aplikasi menampilkan detail presensi maka mahasiswa tinggal melakukan presensi, jika presensi gagal maka dalam kondisi ini mahasiswa hanya perlu melakukan presensi ulang.

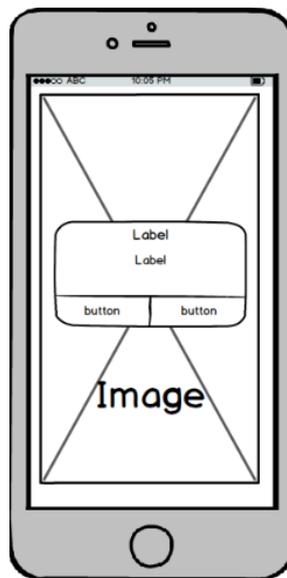
3.5.2 Rancangan Antarmuka (*User Interface*)

User Interface sangatlah penting dalam suatu aplikasi karena merupakan bagian dari perangkat lunak yang menjadi sarana komunikasi antar pengguna dengan sistem serta dapat memberikan kemudahan bagi pengguna dalam melakukan aktivitasnya.

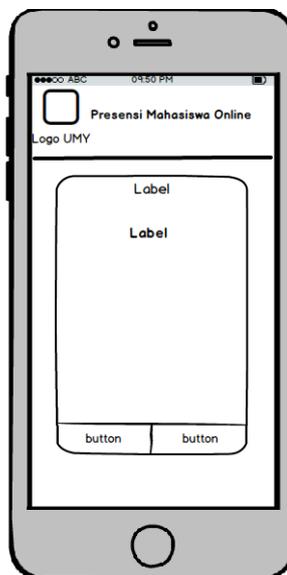
a. Rancangan Antarmuka Halaman Pembuka

Rancangan antar muka ini adalah rancangan halaman yang pertama kali muncul saat aplikasi dijalankan, pada halaman ini terdapat sebuah konfirmasi untuk memastikan bahwa *user* telah tersambung dengan jaringan lokal UMY. Jika ternyata *user* belum tersambung dengan jaringan lokal UMY maka akan menampilkan pesan bahwa terdapat masalah dengan koneksi. Pada rancangan ini terdapat dua buah tombol dimana berfungsi untuk memastikan bahwa *user* sudah

tersambung dengan jaringan lokal UMY atau belum, dan tombol yang lainnya berfungsi untuk keluar dari aplikasi. Gambaran rancangan antarmuka halaman pembuka ini dapat dilihat pada gambar 3.8 dan gambaran rancangan antarmuka untuk pesan kesalahan dengan jaringan dapat dilihat pada gambar 3.9.



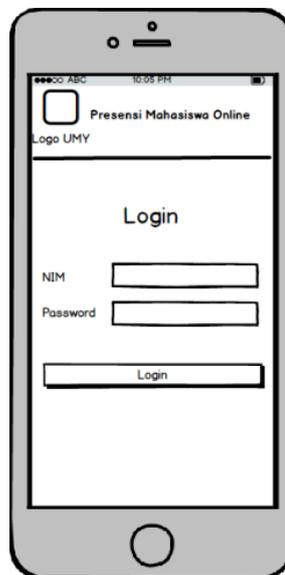
Gambar 3.8 Rancangan antarmuka halaman pembuka



Gambar 3.9 Rancangan tampilan pesan kesalahan dengan jaringan

b. Rancangan Antarmuka Halaman *Login*

Rancangan antarmuka ini merupakan halaman validasi *token* yang berfungsi untuk melakukan *login* kedalam aplikasi. Mahasiswa melakukan *login* sebelum masuk pada halaman validasi *token*. Setiap satu *smartphone* berbasis *android* hanya dapat digunakan untuk satu kali *login*, sehingga ketika *smartphone* berbasis *android* pernah digunakan untuk melakukan *login* untuk salah satu mahasiswa maka mahasiswa yang lain tidak dapat melakukan *login* lagi dengan *device* yang sama. Gambaran rancangan antarmuka halaman ini dapat dilihat pada gambar 3.10.

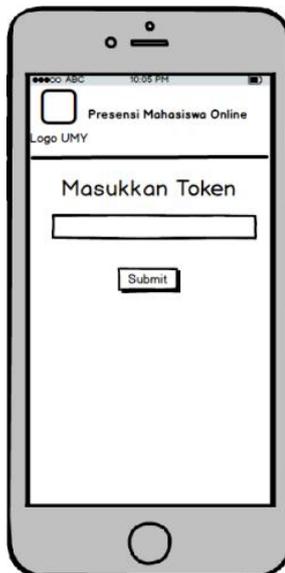


Gambar 3.10 Rancangan antarmuka halaman *login*.

c. Rancangan Antarmuka Halaman validasi *token*

Rancangan antarmuka ini merupakan halaman validasi *token* yang berfungsi untuk melakukan validasi *token*. Didalam tampilan ini terdapat satu buah tombol yang berfungsi sebagai validasi *token*, jika *token* yang dimasukkan valid maka

sistem akan menampilkan halaman detail sesi perkuliahan dan jika *token* yang dimasukkan tidak valid maka akan muncul pesan bahwa *token* tidak valid. Gambaran rancangan antarmuka halaman ini dapat dilihat pada gambar 3.11.



Gambar 3.11 Rancangan antarmuka halaman validasi *token*

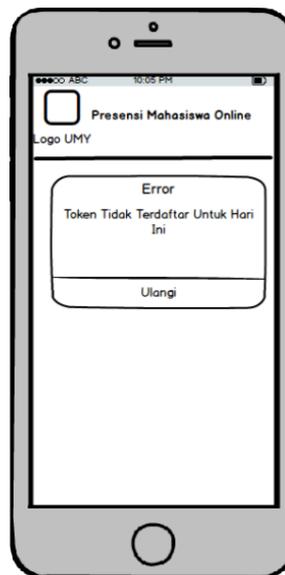
d. Rancangan Antarmuka Tampilan Pesan *Token* Tidak Valid

Rancangan ini merupakan rancangan untuk menampilkan pesan bahwa *token* yang dimasukkan tidak benar atau *token* salah. Didalam tampilan ini terdapat satu tombol yang berfungsi untuk kembali kehalaman validasi *token*. Gambaran rancangan antarmuka tampilan ini dapat dilihat pada gambar 3.12.

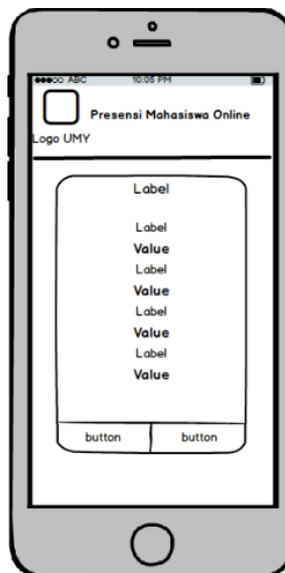
e. Rancangan Antarmuka Halaman *Detail* Sesi Perkuliahan

Rancangan antarmuka ini merupakan Halaman detail sesi perkuliahan yang akan atau sedang berlangsung. Halaman ini menampilkan beberapa data dari sesi perkuliahan. Pada rancangan ini terdapat dua buah tombol dimana satu tombol

berfungsi untuk melakukan presensi dan tombol yang lain adalah untuk membatalkan presensi jika mahasiswa merasa detail yang tampil bukan merupakan perkuliahan yang ingin diikuti. Gambaran rancangan antarmuka halaman ini dapat dilihat pada gambar 3.13.



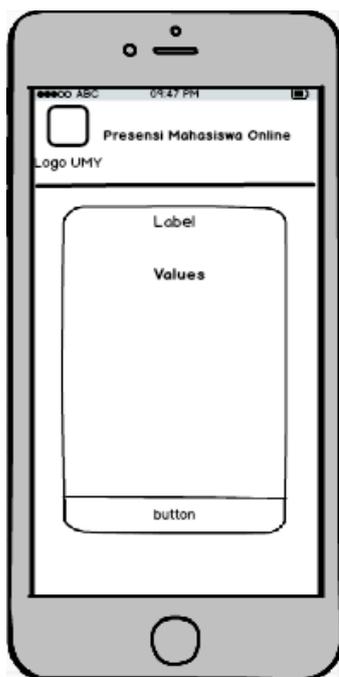
Gambar 3.12 Rancangan antarmuka tampilan pesan *token* tidak valid



Gambar 3.13 Rancangan antarmuka halaman detail sesi perkuliahan

f. Rancangan Antarmuka Tampilan Presensi Gagal

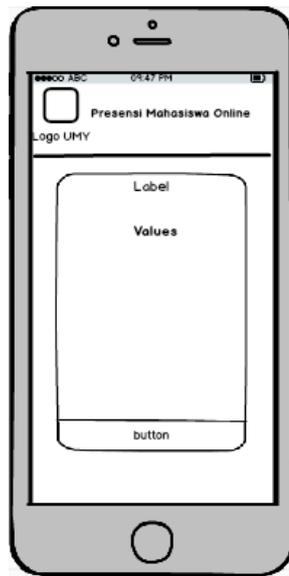
Rancangan ini merupakan rancangan untuk menampilkan pesan bahwa presensi gagal. Didalam tampilan ini terdapat pesan alasan kenapa presensi gagal dilakukan dan satu tombol yang berfungsi untuk kembali kehalaman validasi *token*. Gambaran rancangan antarmuka tampilan ini dapat dilihat pada gambar 3.14.



Gambar 3.14 Rancangan antarmuka tampilan presensi gagal

g. Rancangan Antarmuka Tampilan Presensi Berhasil

Rancangan ini merupakan rancangan untuk menampilkan pesan bahwa presensi berhasil dilakukan. Didalam tampilan ini terdapat pesan bahwa presensi telah berhasil dilakukan serta status keterlambatan presensi, selain itu pada tampilan ini juga terdapat satu buah tombol yang berfungsi untuk keluar dari aplikasi. Gambaran rancangan antarmuka tampilan ini dapat dilihat pada gambar 3.15.



Gambar 3.15 Rancangan antarmuka tampilan presensi berhasil