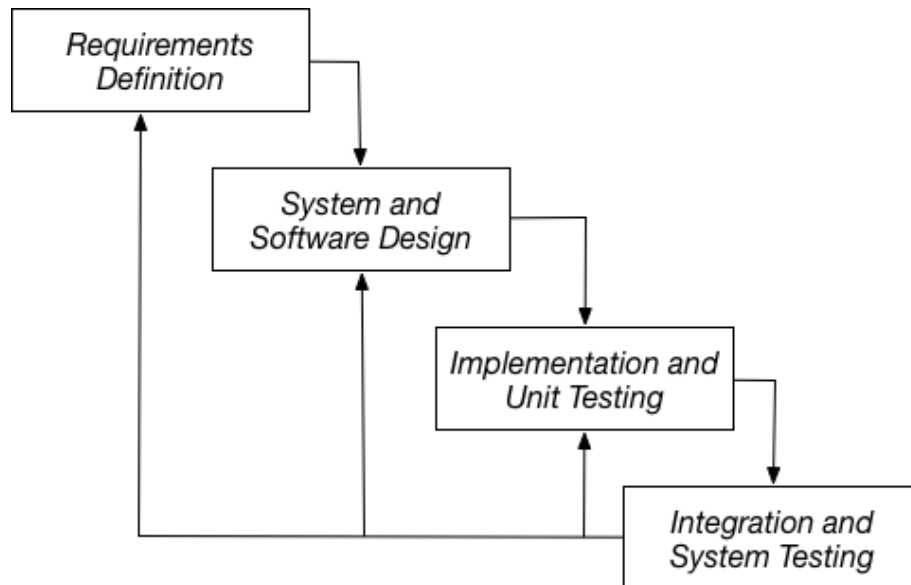


BAB III

METODE PENELITIAN

3.1 Metode Pengembangan Sistem

Dalam penelitian ini penulis menggunakan metode *Waterfall* sebagai model pengembangan sistem. Model SDLC yang dipakai dalam penelitian ini mengadopsi model *waterfall* yang dikemukakan oleh Govardhan .dkk (2010) dalam jurnalnya yang berjudul “*A Comparison Between Five Models Of Software Engineering*”. Dalam jurnal tersebut dikemukakan Model dimulai dengan membangun *system requirements* dan *software requirements* dan berlanjut dengan *architectural design*, *detailed design*, *coding*, *testing*, dan *maintenance*.



Gambar 3.1 Model *Waterfall*

Berdasarkan gambar 3.1 menunjukkan *Model Waterfall* yang dipakai yaitu menggunakan *Model Waterfall* yang dikemukakan Govardan .dkk (2010) dengan menghilangkan tahap *Operation* dan *Maintenance*. Karena penelitian ini tidak

melakukan tahap *Operation* dan *Maintenance*. Tetapi dalam penelitian ini hanya sampai tahap *System testing*,

3.1.1 *Requirements Definition*

Pada tahap ini peneliti melakukan observasi dan wawancara langsung dengan pihak Biro Sitem Informasi (BSI) di Universitas Muhammadiyah Yogyakarta. Penjelasan lebih lengkap dari aktifitas tersebut terdapat pada bab 3.3. Pada Observasi ini terfokus pada permasalahan yang menjadi keluhan dari pihak user. Apakah kebutuhan fungsi *software* untuk memenuhi kendala yang dialami oleh pengguna. Sehingga mendapatkan dokumentasi kebutuhan pengguna yang digunakan sebagai bahan untuk ditunjukkan kepada pengguna. Penjelasan lebih lengkap dari aktifitas tersebut terdapat pada bab 3.4.

3.1.2 *System & Software Design*

Setelah peneliti mendapatkan dokumentasi dari hasil *Requirements Definition*, maka peneliti mengubah kebutuhan-kebutuhan fungsi software diatas menjadi modul-modul bentuk "*blueprint*" *software* untuk digunakan dalam mengembangkan sistem.

3.1.3 *Implementation*

Pada tahap ini, peneliti membangun sebuah aplikasi berdasarkan modul-modul 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). Penjelasan lebih lengkap dari aktifitas ini terdapat pada lampiran 1.

3.1.4 *Integration & System Testing*

Setelah proses pembangunan aplikasi selesai, peneliti melakukan pengujian pada tahap ini. Aplikasi diuji berdasarkan metode *black box* untuk mengetahui tingkat keberhasilan dari bagian sistem. Penjelasan lebih lengkap dari aktifitas ini terdapat pada bab 4.2.

3.2 **Metode Pengumpulan Data**

3.2.1 **Observasi**

Observasi adalah pengamatan dan pencatatan secara sistematis terhadap unsur-unsur yang tampak dalam suatu gejala atau gejala-gejala pada obyek penelitian (Nawawi dan Martini 1992, p.74). Dengan adanya observasi peneliti dapat mengamati dan mengidentifikasi sistem yang digunakan untuk presensi mahasiswa yang sedang berjalan sebelumnya.

Observasi yang dilakukan penulis adalah pengamatan secara langsung (*participant observation*). Penelitian ini melibatkan peneliti atau *observer* secara langsung dalam kegiatan pengamatan objek yang akan diteliti. Pada penelitian ini peneliti mendapatkan data dengan mengamati objek yang akan diteliti secara langsung di tempat penelitian yaitu Biro Sistem Informasi (BSI) UMY. Pengumpulan data meliputi sistem pengumpulan data presensi yang digunakan di Universitas Muhammadiyah Yogyakarta.

3.2.2 **Wawancara**

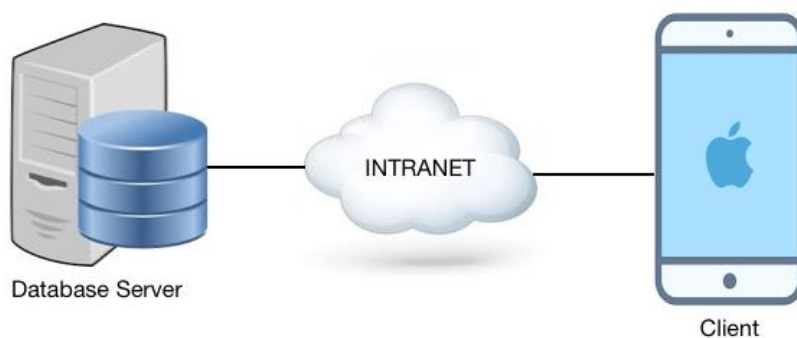
Wawancara yang digunakan penulis dalam penelitian ini adalah dengan wawancara yang terjadi secara spontan bergantung dengan suasana dan keadaan ketika kegiatan wawancara berlangsung. Karena peneliti menggunakan jenis

wawancara bebas atau tidak terstruktur berdasarkan daftar pertanyaan-pertanyaan untuk mengumpulkan data yang dicari. Proses wawancara ini dilakukan di Biro Sistem Informasi (BSI) UMY untuk mendapatkan data dengan melakukan dialog langsung dengan Kepala *Techno Creative* UMY Bapak Ir. Eko Prasetyo, M.Eng yang dianggap dapat memberikan informasi yang dibutuhkan oleh peneliti.

3.3 Analisis Data

3.3.1 Arsitektur

Berdasarkan hasil observasi yang telah dilakukan dapat diketahui bahwa Universitas Muhammadiyah Yogyakarta mempunyai banyak *server*. Setiap server mempunyai data-data yang diperuntukan untuk kebutuhan yang berbeda-beda. Arsitektur yang digunakan di presensi perkuliahan sebelumnya menggunakan arsitektur *two-tier*. Arsitektur *two-tier* yang digunakan dapat dilihat pada gambar 3.2.



Gambar 3. 2 Arsitektur *two-tier* Aplikasi

Untuk memenuhi kebutuhan pengembangan sistem presensi ini, Maka peneliti menggunakan rancangan arsitektur *three-tier* seperti pada gambar 3.3.



Gambar 3.3 Arsitektur *three-tier* Aplikasi

Dalam arsitektur *three-tier* aplikasi yang telah digambarkan pada gambar 3.3 terdapat 3 bagian. Bagian pertama dalam gambar tersebut terdapat *Database Server* dengan menggunakan *Database Microsoft SQL Server 2014*. Kemudian dihubungkan melalui intranet dengan *web server* yang menggunakan *Apache Web Server*. Dalam *Apache Web Server* terdapat *Web service* yang digunakan sebagai perantara antara *database* dengan *client*. *Web service* dalam pengembangannya menggunakan teknik *REST(Representational State Transfer)* yang dikenal sebagai *RESTful web service*. Dalam membangun *RESTful web service* peneliti menggunakan *slim framework*.

Selanjutnya *RESTful web service* dihubungkan dengan *client* melalui *intranet*. *Client* pada gambar 3.3 hanya menggunakan sistem operasi *iOS* untuk mengakses sistem presensi Mahasiswa. Karena peneliti tidak melakukan perubahan untuk aplikasi *Android* supaya dapat terhubung ke *web service*.

3.3.2 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 penerapan proses presensi yang telah dilakukan sebelumnya, peneliti memperoleh kesimpulan bahwa proses presensi dapat dilakukan secara sekaligus. Maka penulis mendapat rancangan proses melakukan presensi mahasiswa dan juga dapat mengantisipasi kecurangan dalam presensi seperti pada gambar 3.4.

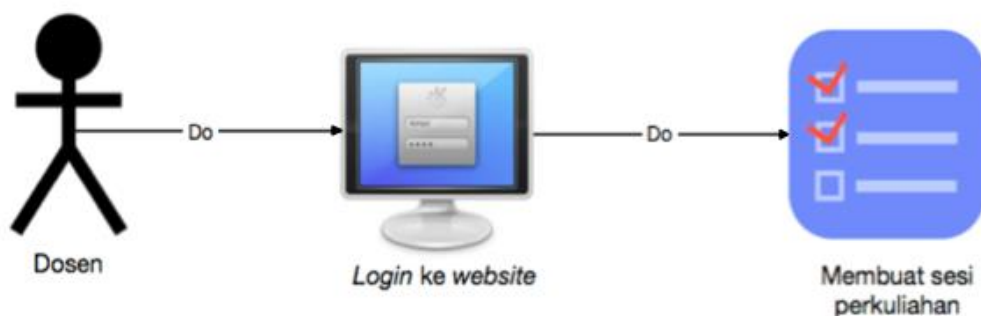


Gambar 3.4 Analisis Pengguna

3.3.3 Analisis Pembuatan Sesi Perkuliahan

Proses presensi mahasiswa yang saat ini masih berjalan di Universitas Muhammadiyah Yogyakarta dalam pembuatan sesi perkuliahan, pada masing-masing prodi atau jurusan dilakukan oleh staf yang berbeda-beda. Dari pernyataan tersebut penulis menyimpulkan bahwa proses pembuatan sesi perkuliahan melalui staf dapat memperlambat proses presensi sampai terekap, padahal proses presensi mahasiswa tersebut dapat dilakukan oleh dosen pada saat mengisi perkuliahan.

Peneliti dalam hal ini memperoleh sebuah rancangan seperti pada Gambar 3.5.



Gambar 3.5 Analisis Pembuatan Sesi Perkuliahan

Penjelasan gambar 3.5 tentang analisis pembuatan sesi perkuliahan adalah sebagai berikut:

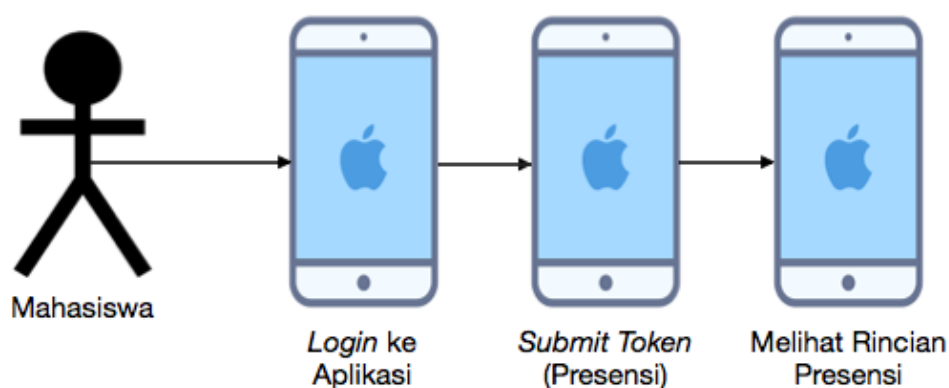
1. Dosen mengakses *website* presensi *online* UMY, dan *login* berdasarkan *email* dari kampus.
2. Dosen melakukan pembuatan sesi perkuliahan berdasarkan tahun ajaran, jurusan, matakuliah dan kelas.
3. Selanjutnya dosen akan masuk ke halaman detail sesi perkuliahan. Pada halaman detail sesi perkuliahan terdapat beberapa data tentang sesi perkuliahan yang telah dibuat seperti: tahun ajaran, kelas program, matakuliah, kelas, ruangan, waktu mulai, waktu selesai, maksimal keterlambatan, isi perkuliahan, deskripsi dan token. Selain itu dosen juga bisa melihat data mahasiswa yang telah melakukan presensi berdasarkan token.
4. Kemudian dosen akan memberi kode *token* kepada mahasiswa untuk melakukan proses presensi.

3.3.4 Analisis Presensi Mahasiswa

Universitas Muhammadiyah Yogyakarta saat ini dalam proses presensi mahasiswa masih menggunakan lembar berita acara. Lembar berita acara diisi oleh mahasiswa ketika sesi perkuliahan berlangsung. Kemudian lembar berita acara akan diserahkan ke staf pada setiap prodi atau jurusan untuk dilakukan rekap data pada aplikasi presensi.

Peneliti mendapat kesimpulan bahwa proses presensi masih memakan waktu yang lama. Dari kesimpulan tersebut penulis membuat sebuah rancangan dengan proses presensi tidak lagi menggunakan lembar berita acara. Namun presensi menggunakan aplikasi *mobile*, Sehingga setelah mahasiswa melakukan presensi data langsung terekap pada database. Rancangan yang dibuat mempunyai dua kondisi yang berbeda untuk mengurangi kecurangan saat presensi.

Rancangan analisis presensi mahasiswa pada kondisi ketika Mahasiswa belum melakukan *login* menggunakan aplikasi presensi mahasiswa *iOS* seperti pada Gambar 3.6.

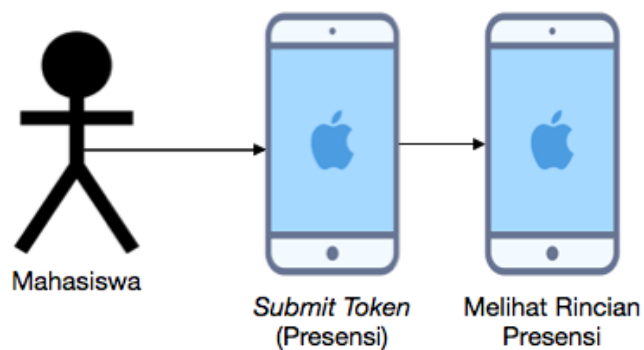


Gambar 3.6 Analisis presensi mahasiswa kondisi belum *login*

Penjelasan gambar 3.6 tentang analisis presensi Mahasiswa adalah sebagai berikut:

- 1) Mahasiswa akan memperoleh *token* dari dosen berdasarkan sesi perkuliahan yang telah dibuat.
- 2) Mahasiswa akan melakukan *login* ke aplikasi presensi *iOS* berdasarkan NIM(Nomor Induk Mahasiswa) dari kampus beserta *password*.
- 3) Mahasiswa melakukan *submit token* sesuai dengan *token* yang diberikan dosen pada sesi perkuliahan tertentu.
- 4) Jika proses presensi berhasil maka akan tampil pemberitahuan bahwa presensi berhasil. Jika presensi gagal maka akan tampil pemberitahuan bergantung pada kegagalan persensi.
- 5) Mahasiswa dapat melihat rincian presensi dari presensi yang telah dilakukannya.

Rancangan analisis presensi mahasiswa pada kondisi ketika mahasiswa sudah melakukan *login* menggunakan aplikasi presensi mahasiswa *iOS* seperti pada Gambar 3.7.



Gambar 3.7 Analisis presensi mahasiswa kondisi sudah login

Penjelasan proses presensi pada gambar 3.7 hampir sama dengan proses presensi pada gambar 3.6. Perbedaanya dari kedua kondisi tersebut yaitu ketika mahasiswa sudah melakukan *login* menggunakan aplikasi presensi *iOS*, Mahasiswa

tidak melakukan *login* atau tidak melakukan proses pada penjelasan gambar 3.6 kondisi ketika mahasiswa belum melakukan *login*.

Menghilangkan proses *login* ketika mahasiswa sudah melakukan *login* ini dimaksudkan untuk mengatasi kebutuhan dimana sistem harus dapat menerapkan satu *device* untuk *satu user* untuk menanggulangi kecurangan dalam presensi. Karena ada suatu kondisi ketika mahasiswa dapat melakukan presensi dengan akun mahasiswa lainnya.

3.3.5 Analisis Kebutuhan Sistem

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

1. Aplikasi harus menerapkan sistem keamanan dalam hal presensi. Aplikasi ini masih seperti aplikasi presensi berbasis *Android* sebelumnya yang menggunakan *TWO Factor Authentication* untuk menjaga keamanan aplikasi. 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 yang telah melakukan *key-in* sebelumnya.

2. Sistem harus dapat menerapkan satu *device* untuk satu *user*. Untuk mengatasi hal tersebut peneliti sudah membuat rancangan pada gambar 3.7 beserta penjelasannya.
3. Sistem hanya dapat digunakan ketika sedang berada pada jaringan UMY. Hal tersebut diperlukan supaya mahasiswa tidak dapat melakukan presensi dari diluar area kampus UMY.
4. Halaman *login* mahasiswa yang akan melakukan presensi. Halaman login diperlukan untuk mengetahui status pengguna adalah mahasiswa UMY atau bukan.
5. Halaman validasi *token* untuk memasukkan *token* yang diberikan oleh dosen.
6. Fitur rincian daftar presensi untuk menampilkan daftar rincian presensi dari presensi yang telah dilakukan mahasiswa menggunakan aplikasi ini maupun aplikasi *Android* sebelumnya.
7. Aplikasi diajukan ke *iOS App Store*.

3.4 Rancangan

Rancangan merupakan kegiatan menerjemahkan hasil analisa ke dalam bentuk modul-modul perangkat lunak kemudian menggabungkan modul-modul tersebut untuk menciptakan sistem ataupun memperbaiki sistem dengan yang sudah ada agar sesuai kebutuhan. Rancangan pada penelitian ini terdapat beberapa rancangan, yaitu rancangan sistem dan rancangan antarmuka pengguna.

3.4.1 Rancangan Sistem

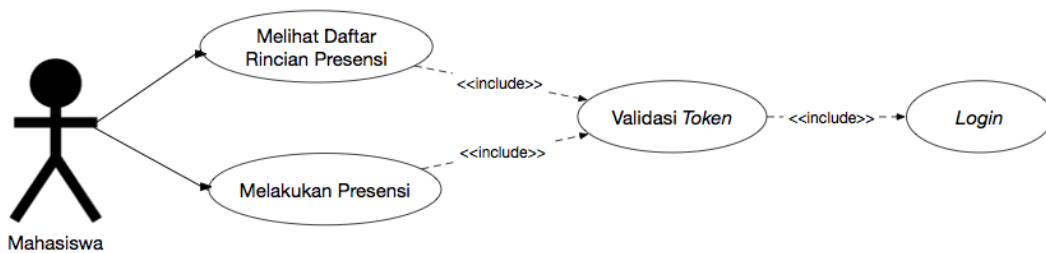
Dalam pembuatan aplikasi web dilakukan perancangan sistem menggunakan UML. Metode UML yang digunakan dalam perancangan aplikasi

web ini antara lain *Use Case Diagram*, *Activity Diagram* dan *Sequence Diagram*. Perancangan tersebut dijelaskan sebagai berikut:

a. ***Use Case Diagram***

Use Case Diagram digunakan untuk menggambarkan interaksi antara pengguna sistem (*actor*) dengan kasus (*use case*) yang disesuaikan dengan langkah-langkah (*scenario*) yang telah ditentukan.

Gambaran *Use Case Diagram* yang digunakan pada aplikasi presensi mahasiswa berbasis iOS dapat dilihat pada gambar 3.8.

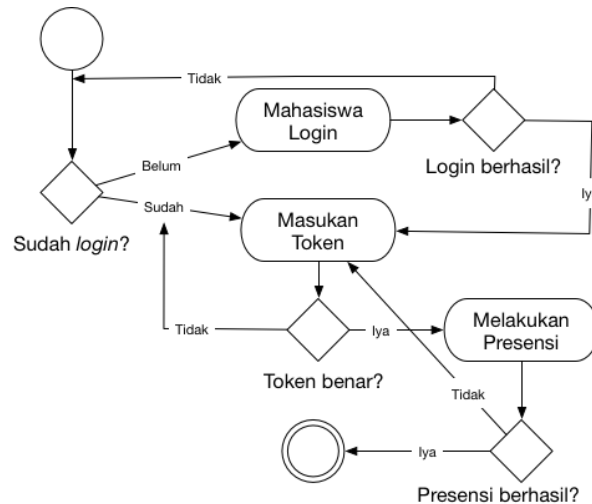


Gambar 3.8 *Use Case Diagram*

Pada Gambar 3.8 dijelaskan bahwa mahasiswa hanya dapat melakukan *validasi token* setelah berhasil melalui proses *login*. Kemudian setelah itu Mahasiswa dapat melihat daftar rincian presensi dan melakukan presensi hanya saat proses *validasi token* berhasil.

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 3.9.



Gambar 3.9 Activity Diagram

Penjelasan Gambar 3.9 tentang langkah-langkah dalam melakukan presensi di aplikasi ini adalah sebagai berikut:

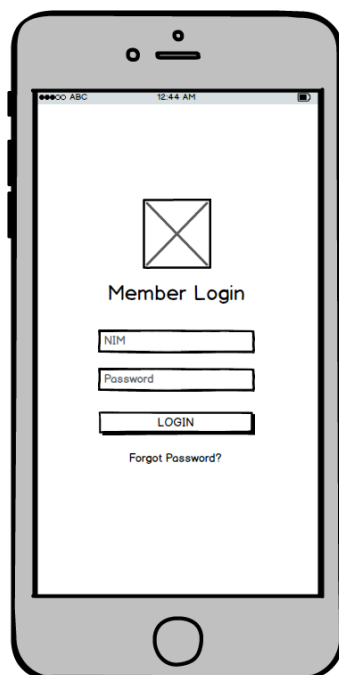
- a) Ketika pertama kali menjalankan sistem ini, sistem akan memeriksa pengguna sudah melakukan *login* atau belum. Jika belum melakukan *login* maka akan berpindah ke halaman *login* dan jika sudah melakukan *login* maka akan berpindah ke *menu* utama untuk memasukkan *token*.
- b) Jika *nim* dan *password* benar maka berpindah ke *menu* utama untuk memasukkan *token*. Jika *nim* dan *password* benar maka akan menampilkan pemberitahuan untuk memasukkan *password* yang benar.
- c) Setelah masuk ke halaman utama, pada halaman utama disediakan fitur untuk memasukkan *token*. *Token* yang dimasukkan sesuai dengan *token* yang diberikan dosen.
- d) Jika *token* benar maka sistem akan menampilkan pemberitahuan bahwa presensi berhasil dan kembali ke *menu* utama. Jika *token* salah maka sistem akan menampilkan pemberitahuan sesuai kesalahan *validasi token* dan kembali ke *menu* utama.

3.4.2 Rancangan Antarmuka (*User Interface*)

User interface merupakan bentuk tampilan yang menghubungkan atau penerjemah informasi antara pengguna dengan *software*, sehingga *software* dapat dioperasikan. Karena dalam aplikasi presensi mahasiswa ini terdapat Mahasiswa sebagai pengguna atau aktor untuk mengoperasikan aplikasi tersebut, maka aplikasi ini membutuhkan *user interface*.

a. Rancangan Antarmuka Halaman *Login Mahasiswa*

Halaman ini akan muncul ketika pertama kali aplikasi dijalankan. Pada halaman ini Mahasiswa memasukan NIM beserta *password* pengguna sebagai syarat untuk login. Setelah pengguna melakukan login maka sistem akan menampilkan halaman menu utama. Halaman login ini ditampilkan sistem hanya pada kondisi pertama, kondisi ini dijelaskan pada gambar 3.6. Gambaran rancangan antarmuka halaman ini dapat dilihat pada Gambar 3.10.

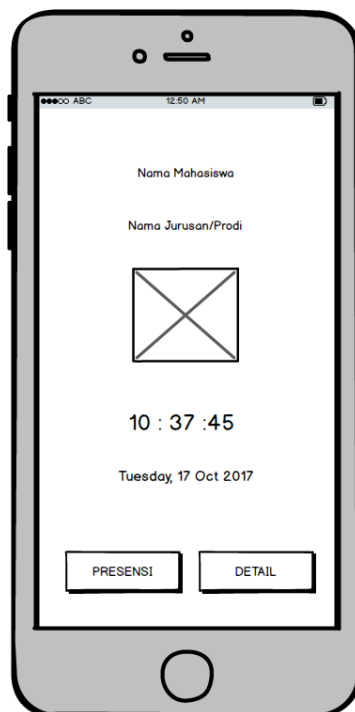


Gambar 3.10 Rancangan Antarmuka Halaman *Login*

b. **Rancangan Antarmuka Halaman *Menu Utama***

Halaman ini merupakan halaman yang berfungsi untuk menampilkan informasi pengguna seperti nama , jurusan dan foto pengguna. Halaman ini juga menampilkan waktu dan tanggal yang berfungsi untuk memudahkan pengguna dalam memeriksa waktu dan tanggal.

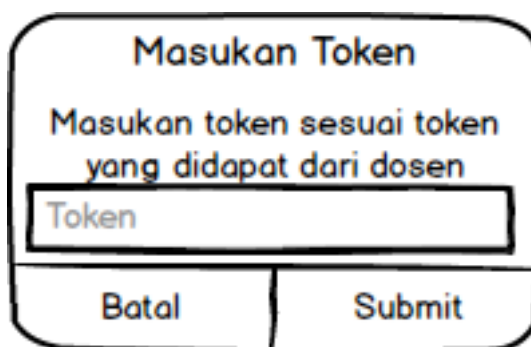
Pada halaman ini terdapat dua tombol yaitu tombol presensi dan tombol detail. Tombol presensi berfungsi untuk menampilkan antarmuka *input token* yang dijelaskan pada rancangan antarmuka *input token* serta digambarkan pada gambar 3.12. Sedangkan tombol detail berfungsi untuk menampilkan halaman rincian presensi yang dijelaskan pada rancangan antarmuka halaman rincian presensi serta digambarkan pada gambar 3.13. Gambaran rancangan antarmuka halaman ini dapat dilihat pada Gambar 3.11.



Gambar 3.11 Rancangan Antarmuka Halaman Menu Utama

c. **Rancangan Antarmuka Proses *Input Token***

Rancangan antarmuka ini menggunakan *popup layout* sebagai tampilan untuk proses *input token* mahasiswa. Pada rancangan antarmuka ini terdapat dua buah tombol yaitu tombol batal dan tombol *submit*. Tombol batal digunakan untuk membatalkan proses *input token* dan menyembunyikan tampilan proses input token. Sedangkan tombol *submit* berfungsi untuk validasi *token*, jika *token* yang dimasukkan *valid* dan maka sistem akan menampilkan pemberitahuan bahwa proses presensi berhasil dan jika *token* yang dimasukkan tidak *valid* maka akan muncul pesan bahwa *token* tidak *valid*. Gambaran rancangan antarmuka halaman ini dapat dilihat pada gambar 3.12.

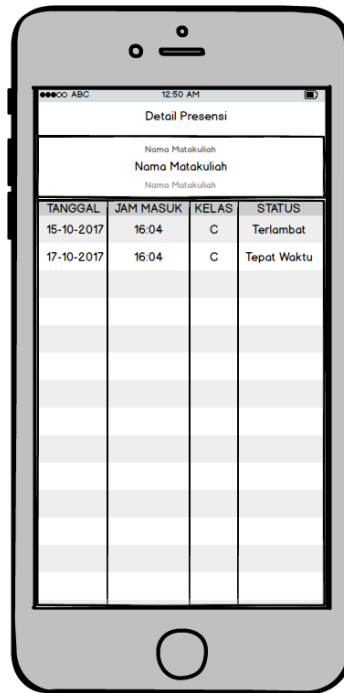


The image shows a screenshot of a web form titled "Masukan Token". The form is enclosed in a rounded rectangular border. At the top, the title "Masukan Token" is displayed in a bold, black font. Below the title, there is a subtitle "Masukan token sesuai token yang didapat dari dosen" in a smaller, regular black font. Underneath the subtitle is a text input field with the placeholder text "Token". At the bottom of the form, there are two buttons: "Batal" on the left and "Submit" on the right, both in a regular black font.

Gambar 3.12 Rancangan Antarmuka Proses *Input Token*

d. **Rancangan Antarmuka Halaman Rincian Presensi**

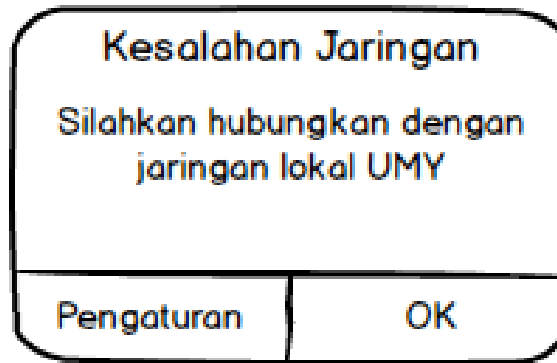
Rancangan ini merupakan rancangan untuk menampilkan tabel presensi yang berisi daftar presensi yang pernah dilakukan mahasiswa. Tabel presensi ditampilkan berdasarkan matakuliah yang dipilih pada pilihan matakuliah. Jika Pilihan Matakuliah diganti pilihannya maka secara otomatis sistem akan mengganti daftar presensi sesuai dengan matakuliah yang dipilih. Gambaran rancangan antarmuka halaman ini dapat dilihat pada gambar 3.13.



Gambar 3.13 Rancangan Antarmuka Halaman Rincian Presensi

e. **Rancangan Antarmuka Pesan Kesalahan Jaringan**

Rancangan ini merupakan rancangan untuk menampilkan pesan kesalahan jaringan. Tampilan ini akan tampil pada halaman yang aktif atau yang sedang digunakan, jika pengguna tidak tersambung dengan jaringan lokal UMY. Pada rancangan ini terdapat dua tombol yaitu tombol pengaturan dan tombol OK. Jika tombol pengaturan ditekan maka sistem operasi akan menampilkan pengaturan jaringan. Sedangkan jika tombol OK ditekan maka pesan akan dihilangkan, tetapi aplikasi tetap akan menampilkan kembali jika pengguna belum terhubung dengan jaringan lokal UMY. Gambaran rancangan antarmuka pesan ini dapat dilihat pada gambar 3.14.

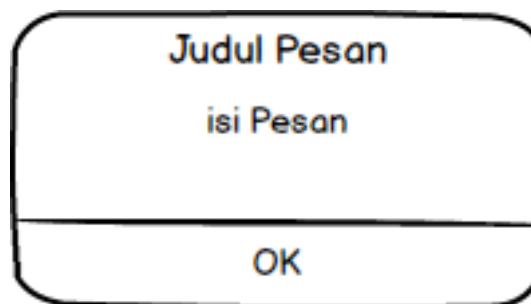


Gambar 3.14 Rancangan Antarmuka Pesan Kesalahan Jaringan

f. **Rancangan Antarmuka Pesan Status Presensi**

Rancangan ini merupakan rancangan untuk menampilkan pesan status keberhasilan presensi, Jika presensi berhasil maka judul dan isi pesan akan berisi pesan bahwa presensi berhasil dilakukan. Sedangkan , jika presensi gagal maka judul dan isi pesan akan berisi alasan kenapa presensi gagal dilakukan.

Dalam rancangan ini terdapat tombol OK yang digunakan untuk kembali ke menu utama jika pesan tersebut merupakan pesan bahwa presensi berhasil. Tetapi jika presensi gagal maka akan kembali menampilkan Antarmuka proses input token. Gambaran rancangan antarmuka pesan ini dapat dilihat pada gambar 3.15.



Gambar 3.15 Rancangan Antarmuka Pesan Status Presensi

3.5 *Software dan Hardware*

3.5.1 *Hardware*

Hardware yang digunakan dalam penelitian ini terdiri dari perangkat keras untuk *pengembang, user* dan *server*.

a. *Hardware untuk Pengembang*

Spesifikasi perangkat keras yang digunakan pengembang dalam pengembangan aplikasi ini dapat dilihat pada tabel 3.1.

Tabel 3.1 Spesifikasi Hardware Pengembang

Sistem Operasi	MacOS High Sierra 10.13
Prosesor	1.8 GHz Intel Core i5
RAM	8 GB 1600 MHz DDR3
Hardisk	500GB

b. *Hardware untuk User (Mahasiswa)*

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

3.2.

Tabel 3.2 Spesifikasi Minimum *Hardware* Mahasiswa

Sistem Operasi	iOS 9
Ram	512 MB
Internal Memori	8GB

c. ***Hardware untuk Server***

Spesifikasi perangkat keras yang digunakan untuk menjalankan *web service* pada aplikasi ini dapat dilihat pada tabel 3.3.

Tabel 3.3 Spesifikasi Server

Sistem Operasi	Windows Server 2012
Prosesor	Intel(R) i3
RAM	2GB
Hardisk	60GB

3.5.2 ***Software***

Software yang digunakan dalam penelitian ini terdiri dari perangkat lunak untuk *pengembang* dan *server*.

a. ***Software untuk Pengembang***

Daftar Perangkat Lunak yang digunakan pengembang untuk mengembangkan Aplikasi ini dapat dilihat pada table 3.4.

Tabel 3.4 Daftar *Software* untuk Pengembang

Software	Versi	Fungsi
Xcode	Versi 9.0	Digunakan untuk pengembangan aplikasi presensi untuk User(Mahasiswa) berbasis <i>iOS</i> .
PhpStrom	Versi 2017.2.4	Digunakan untuk pembuatan <i>web service</i> pada aplikasi presensi berbasis <i>iOS</i> .
Paw	Versi 3.1.4	Digunakan untuk melakukan pengujian webservice.
DataGrip	Versi 2017.1.4	Digunakan untuk pengolahan database yang akan digunakan dalam aplikasi presensi.
Balsamiq Mockups 3	Versi 3.5.14	Digunakan untuk pembuatan Perancangan Antarmuka Aplikasi.
OmniGraffle	Versi 7.4.3	Digunakan untuk pembuatan Perancangan Sistem

b. ***Software untuk Server***

Daftar Perangkat Lunak yang digunakan pengembang untuk menjalankan *web service* dapat dilihat pada table 3.5.

Tabel 3.5 Daftar *Software* untuk Server

No.	Perangkat Lunak
1	Browser Internet (Mozilla Firefox atau Google Chrome)
2	Apache dengan Php versi 5.4
3	SQL Server Management Studio 2014