

BAB III

METODE PENELITIAN

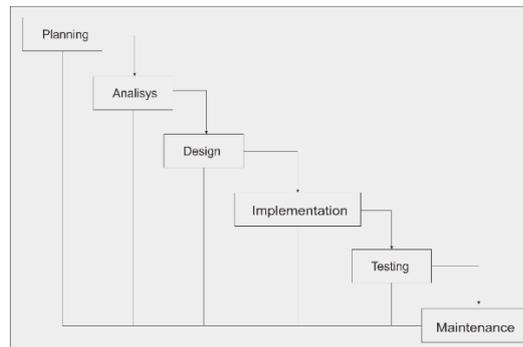
3.1 Metodologi Perancangan

Perancangan terhadap aplikasi *mobile learning* berbasis android dilakukan untuk mengetahui fitur-fitur yang dibutuhkan sehingga menghasilkan *mobile learning* yang layak dan efektif bagi mahasiswa.

Tahapan pembuatan aplikasi menggunakan model *system* pengembangan atau rekayasa *system* informasi *SDLC* (*Software Development Life Cycle*). Metode *SDLC* merupakan proses pengembangan *software*. Konsep ini biasa digunakan sebagai informasi pada perangkat lunak komputer. *SDLC* juga dapat digunakan untuk merencanakan langkah pembangunan perangkat lunak. Model-model yang digunakan pada *SDLC* yaitu:

1. Model *Waterfall*
2. Model Iterasi
3. Model *Rapid Application Development* (RAD)
4. Model *Prototyping*

Model yang akan digunakan pada penelitian ini yaitu model *waterfall*. *Waterfall* merupakan model pengembangan sistem yang paling mudah dan paling sering digunakan. Model pengembangan ini bersifat linear dari tahap awal pengembangan sistem yaitu tahap perencanaan sampai tahap akhir pengembangan sistem yaitu tahap pemeliharaan. Tahapan berikutnya tidak akan dilaksanakan sebelum tahapan sebelumnya selesai dilaksanakan dan tidak bisa kembali atau mengulang ke tahap sebelumnya. Tahapan-tahapan pada model *waterfall* yaitu: perencanaan (*planning*), analisa (*analysis*), desain (*design*), implementasi (*implementation*), pengujian (*testing*) dan pemeliharaan (*maintenance*). Seperti yang dipaparkan pada gambar 3.1.



Gambar 3. 1 Model *Waterfall*

3.1.1 Tahap Perencanaan

Tahap perencanaan merupakan studi kebutuhan pengembangan modul aplikasi. Pada tahapan ini juga di lakukan perencanaan tentang modul aplikasi yang akan di buat dan menentukan strategi yang akan digunakan dalam pengembangan modul aplikasi ini.

3.1.2 Tahap Analisis

Tahap analisis merupakan pemaparan dari sebuah sistem yang sesuai pada kinerja komponen-komponenya, sehingga dapat diidentifikasi dan dapat di evaluasi permasalahan atau hambatan dapat segera diperbaiki.

3.1.3 Tahap Desain

Tahapan desain merupakan tahapan untuk mempersiapkan rancangan sistem yang menggambarkan sistem baru dibentuk menjadi kumpulan modul, untuk menjelaskan proses dan kebutuhan sistem yang berkaitan dengan pembangunan aplikasi.

3.1.4 Tahap Implementasi

Tahapan implementasi merupakan proses penerapan langkah-langkah sebelumnya menggunakan bahasa pemrograman atau yang lebih dikenal dengan sebutan *coding* yang sudah ditentukan pada langkah sebelumnya. Pada tahapan ini pengembang membutuhkan sebuah perangkat lunak yang mendukung untuk melakukan pengembangan.

3.1.5 Tahap Pengujian

Tahapan pengujian merupakan pemeriksaan menyeluruh terhadap sistem. Ujicoba dilakukan untuk mengetahui bahwa produk tersebut benar-benar sesuai dengan kebutuhan yang telah ditentukan. Uji coba dan uji kasus digunakan untuk mengidentifikasi kesalahan dan mengidentifikasi bahwa sistem bekerja sesuai spesifikasi.

3.1.6 Tahap Pemeliharaan

Berikut ini beberapa hal perlunya pemeliharaan terhadap sistem, yaitu:

1. Untuk memperbaiki kesalahan atau kelemahan sistem yang tidak terdeteksi pada saat pengujian.
2. Untuk meningkatkan kinerja sistem

3.2 Alat dan Bahan

Adapun alat dan bahan yang digunakan pada penelitian ini adalah :

3.2.1 Perangkat Keras

Tabel 3. 1 Perangkat keras yang digunakan

<i>Personal Computer (PC) / Leptop</i>	Informasi
RAM	8GB
<i>Proseccor</i>	CPU AMD <i>Quad Core</i> A10. 64 bit
<i>System Operasi</i>	<i>Windows 10</i>

Tabel 3. 2 Perangkat keras yang digunakan

<i>Smartphone</i>	Informasi
<i>Processor</i>	<i>Qualcom Teknologi, version 5.1.1 Lollipop</i>
<i>Random Access Memory (RAM)</i>	4GB
<i>System Operasi</i>	<i>Android</i>

3.2.2 Perangkat Lunak

Tabel 3. 3 Perangkat lunak yang digunakan

no	Nama perangkat	Keterangan
1	<i>Courselab 2.7 dan Courselab 3.0</i>	Perangkat ini digunakan untuk mengembangkan aplikasi
2	<i>Internet Explorer</i>	Perangkat ini digunakan untuk pengujian aplikasi berjalan dengan baik atau tidaknya.
3	<i>Sublim text</i>	Perngakat ini digunakan untuk melakukan penulisan program (<i>coding</i>).
4	<i>Corell Draw 2017</i>	Perangkat ini digunakan untuk membuat desain aplikasi.
5	<i>Windows 10 Pro</i>	Sistem operasi yang digunakan untuk menjalankan perangkat keras.
6	<i>Microsoft Visio</i>	Perangkat ini digunakan dalam hal perancangan sistem menyediakan berbagai <i>template</i> untuk berbagai rancangan sistem.

3.3 Metode Pengumpulan Data

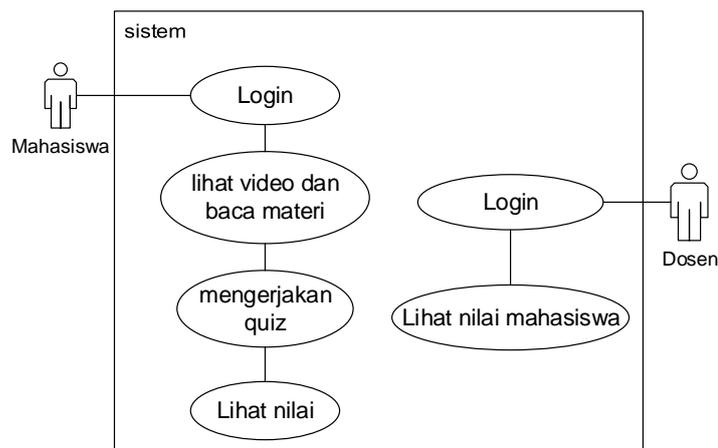
Pengumpulan data merupakan salah satu komponen yang penting dalam melakukan penelitian, untuk memperkuat hasil penelitian diperlukan alat ukur pengumpulan data. Alat ukur yang dapat digunakan untuk pengumpulan data yaitu berupa wawancara, kuesioner, observasi atau dapat juga menggunakan ketiga alat ukur tersebut. Pada penelitian ini penulis menggunakan metode keesioner terhadap pengguna untuk mengukur kelayakan aplikasi.

Kuesioner atau angket Menurut Suroyo anwar (2009:168) merupakan sejumlah pertanyaan atau pernyataan tertulis tentang data faktual atau opini yang berkaitan dengan diri responden, yang dianggap fakta atau kebenaran yang diketahui dan perlu dijawab oleh responden.

3.5 Perancangan Sistem

Perancangan sistem dilakukan untuk pengkajian pengembangan aplikasi dalam perancangan aplikasi mobile learning. Sistem yang dirancang harus mampu dimengerti oleh pengguna dan menyediakan arus data yang dapat masuk dan keluar dengan jelas.

3.5.1 Use Case Diagram



Gambar 3. 2 Diagram Use Case

Berikut adalah deskripsi pendefinisian *Use Case* pada aplikasi *mobile learning* Pemrograman Bahasa Java digunakan untuk menjelaskan setiap proses yang ada pada gambar 3.2.

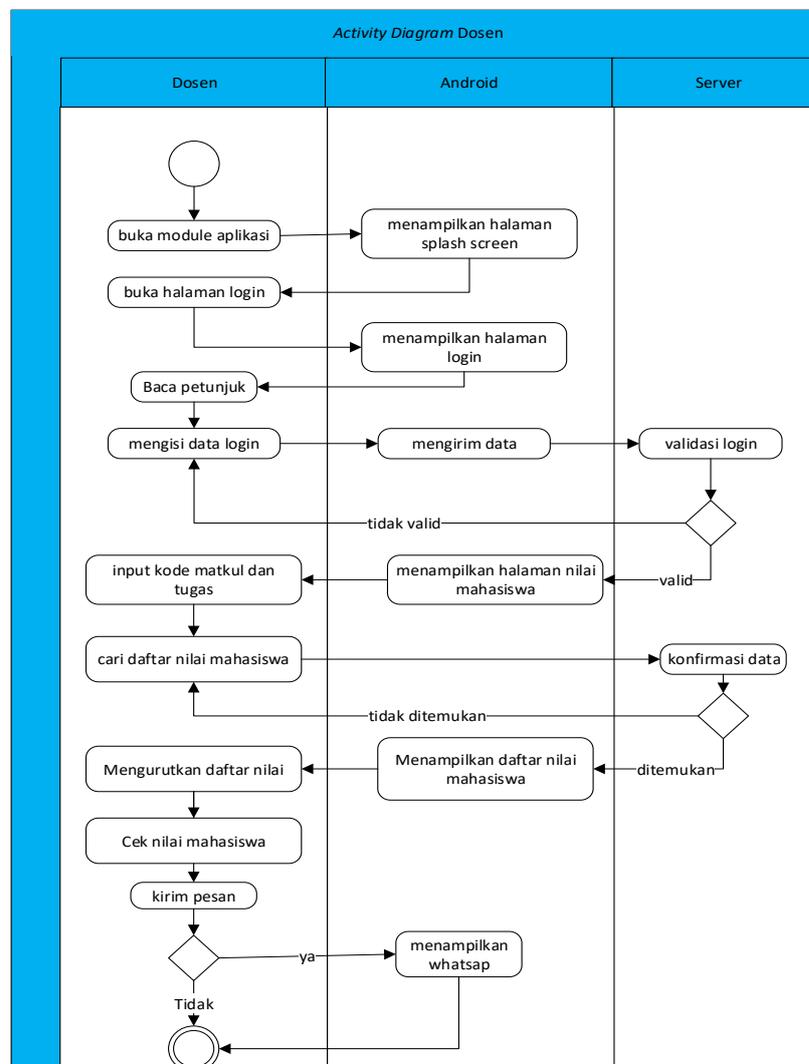
Tabel 3. 4 Penjelasan Use Case diagram

Use Case	Keterangan
Login	Menu tampilan untuk memvalidasi data user ke server apa bila user belum pernah melakukan login
Tampilan Video	Menu tampilan untuk mempelajari pemrograman bahasa java dengan cara melihat video penjelasan
Menu Materi	Menu tampilan untuk mempelajari pemrograman bahasa java dengan cara membaca materi yang disediakan

Use Case	Keterangan
Menu Quiz	Menu tampilan untuk mengetes kemampuan mahasiswa dalam memahami module yang telah diberikan
Halaman Nilai	Menu tampilan untuk melihat nilai mahasiswa yang telah didapat

3.3.2 Activity Diagram

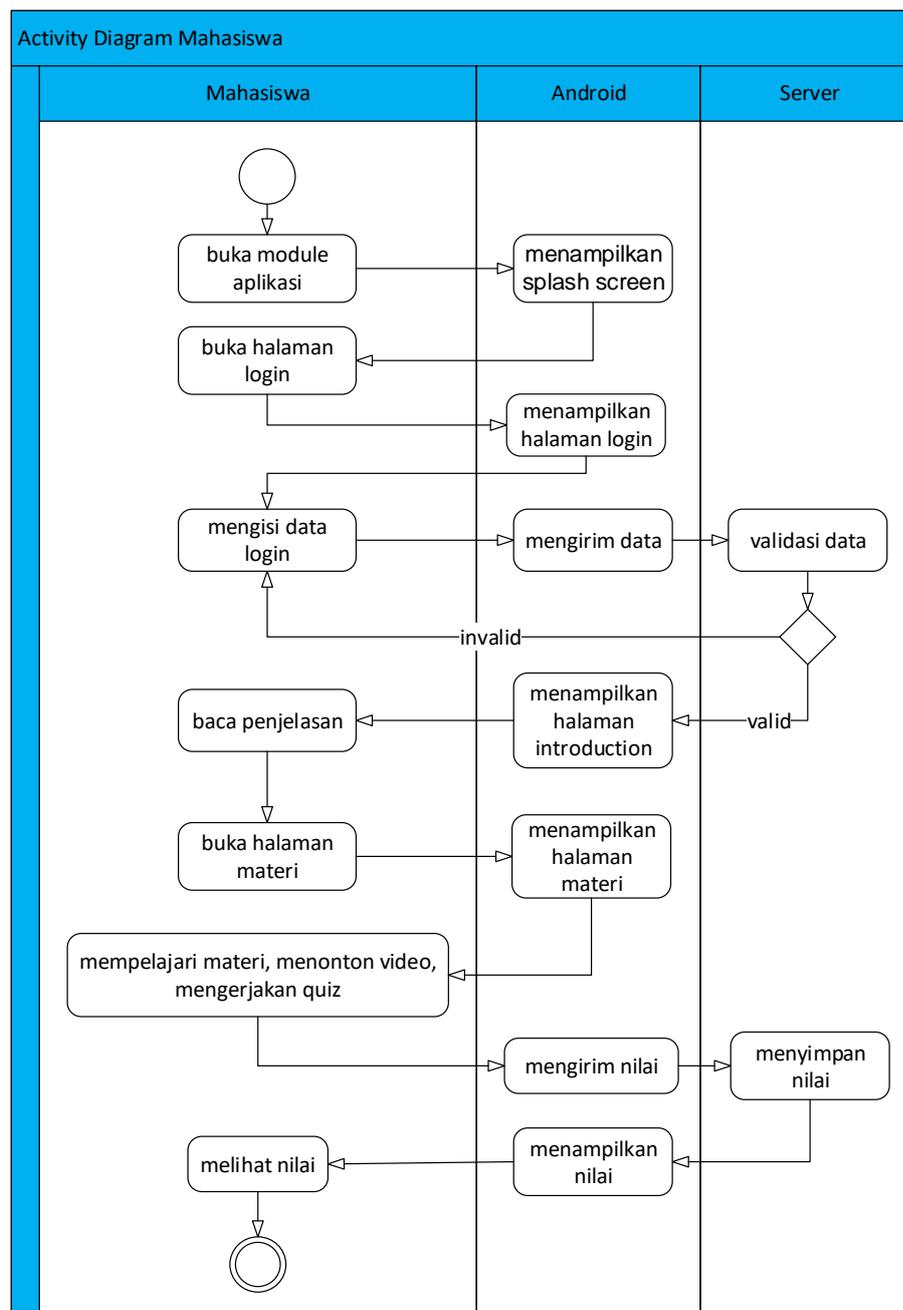
A. Diagram Activity Dosen



Gambar 3. 3 Activity diagram dosen

Seperti yang ditunjukkan pada gambar 3.3 dosen memiliki beberapa aktivitas akan tetapi harus melalui proses validasi supaya dapat membuka module aplikasi berikutnya. Setelah berhasil melalui proses validasi dosen dapat melihat nilai mahasiswa yang telah dikirim dari *server*.

B. Diagram Activity Mahasiswa

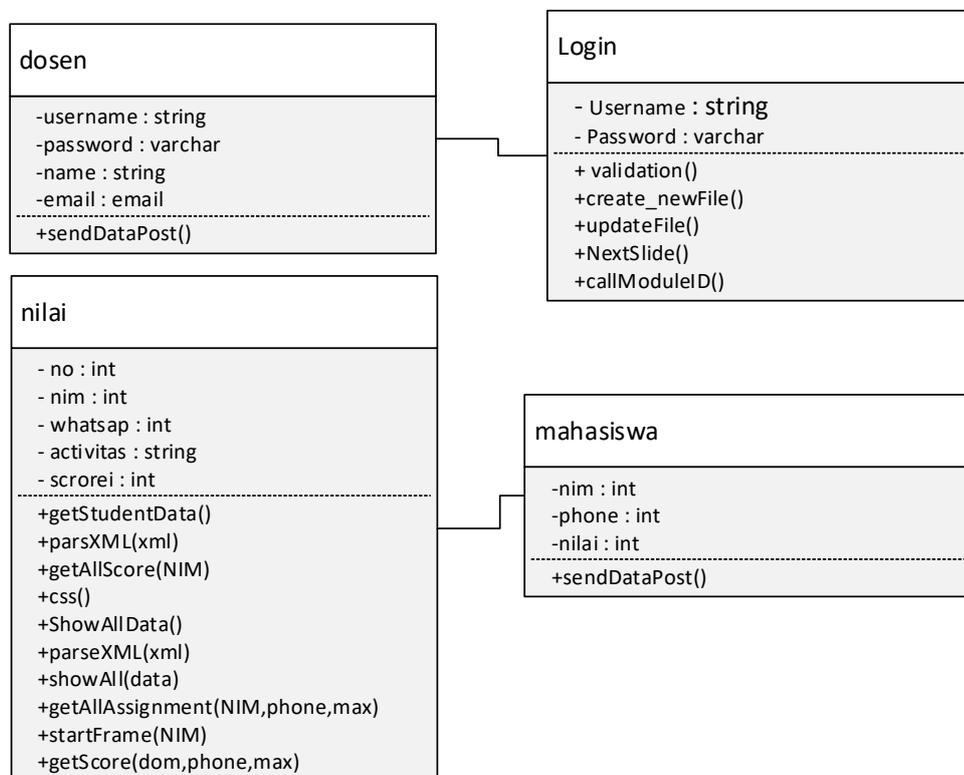


Gambar 3. 4 Activity diagram mahasiswa

Pada gambar 3.4 mahasiswa memiliki beberapa aktivitas tidak berbeda dengan dosen mahasiswa juga harus melalui proses validasi terlebih dahulu. Setelah validasi berhasil mahasiswa dapat melakukan aktivitas selanjutnya yaitu membaca materi, melihat video dan mengerjakan quiz yang disediakan. Setelah selesai mengerjakan quiz mahasiswa dapat melihat nilai yang telah didapat. mahasiswa dapat melihat kembali nilai yang telah didapat dari hasil mengerjakan quis.

3.3.3 Class Diagram

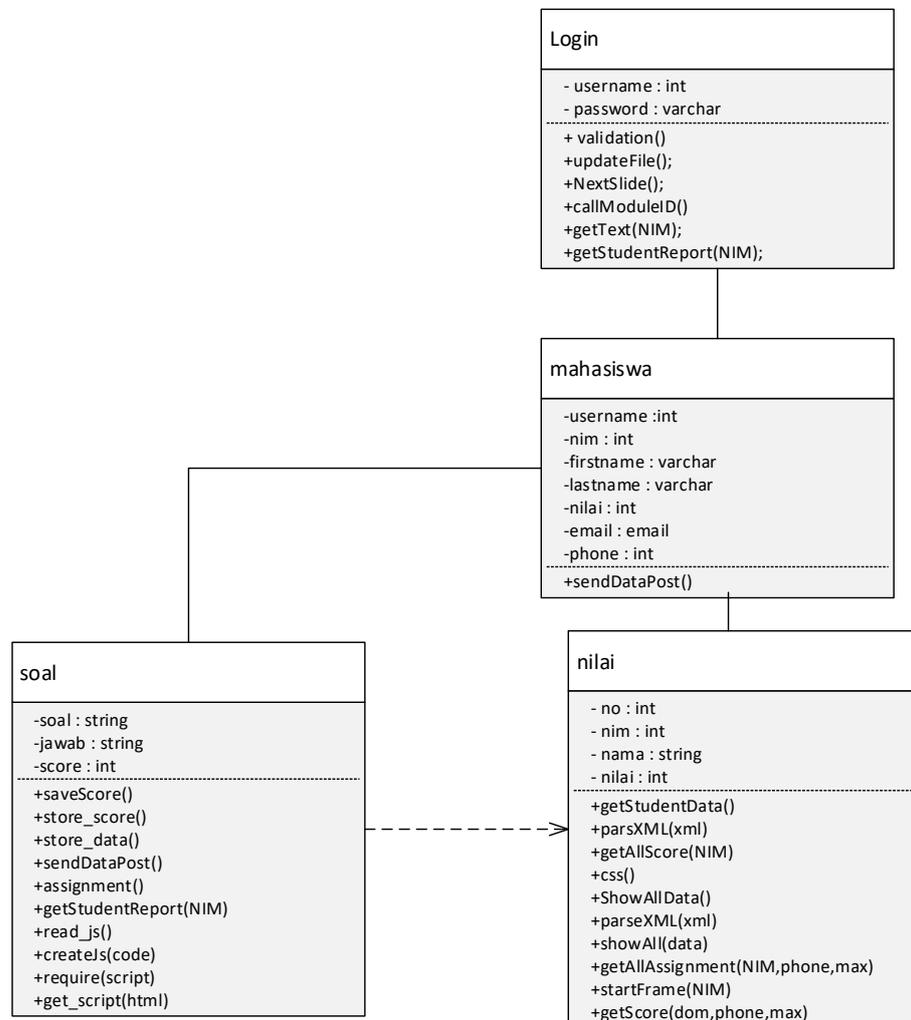
A. Class Diagram Dosen



Gambar 3. 5 Class Diagram Dosen

Pada gambar 3.5 diagram class dosen terdapat beberapa class diantaranya class dosen, login dan nilai. Class dosen digunakan untuk mengirim data yang ada pada class dosen sehingga class login dapat memvalidasi data untuk menuju slide berikutnya. class nilai di gunakan untuk menampilkan nilai mahasiswa yang membutuhkan dari class mahasiswa.

B. Class diagram mahasiswa



Gambar 3. 6 Class Diagram Mahasiswa

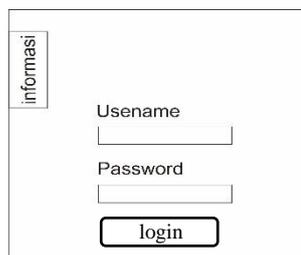
Gambar 3.6 menunjukkan beberapa *class* diagram diantaranya *class login*, *class mahasiswa*, *class soal* dan *class nilai*. *Class mahasiswa* merupakan *class* yang terdapat data yang digunakan pada semua *class* diagram mahasiswa. Pada *class login* membutuhkan data dari *class mahasiswa* untuk validasi. *Class soal* digunakan untuk menyimpan nilai mahasiswa dan *class* ini butuh data dari *class mahasiswa* untuk menyimpan nilai sesuai nama mahasiswa yang sedang mengerjakan soal. *Class nilai* digunakan untuk menampilkan nilai mahasiswa pada *class* ini butuh data dari *class*

mahasiswa, *class* ini bergantung pada *class* soal karena membutuhkan *score* yang ada pada *class* soal.

3.4 Rancangan Antarmuka

3.4.1 Aplikasi *Mobile Learning* Dosen

A. Halaman Login Dosen



The image shows a login form for a mobile application. On the left side, there is a vertical label 'informasi'. The main form area contains two input fields: 'Username' and 'Password'. Below these fields is a button labeled 'login'.

Gambar 3. 7 Rancangan Antarmuka Halaman *Login* Dosen

Pada halaman login seperti yang ditunjukkan pada gambar 3.7 diatas dosen harus memasukkan *username* pada kolom *username* dan *password* pada kolom *password*, setelah selesai maka dosen dapat menekan tombol *login* untuk menuju halaman selanjutnya. Halaman ini dibuat untuk memvalidasi data dosen ke *server* sehingga dosen dapat mengakses data seluruh mahasiswa. Pada halaman ini juga terdapat informasi penggunaan aplikasi.

B. Halaman Lihat Nilai Modul Dosen



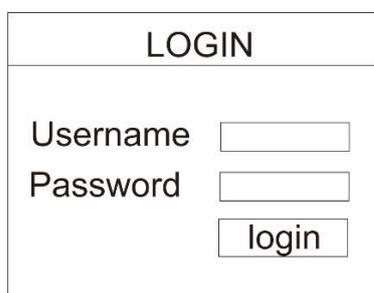
The image shows a screen for viewing student grades. At the top, the word 'nilai' is centered. Below it, there are two input fields: 'Matakuliah' and 'Tugas', each followed by a search button labeled 'Cari'. To the right of these fields is a search icon (magnifying glass). Below the search area, the text 'nilai seluruh mahasiswa' is displayed.

Gambar 3. 8 Rancangan Antarmuka Halaman Lihat Nilai

Setelah dosen selesai melakukan validasi pada halaman *login*, gambar 3.8 akan ditampilkan secara otomatis. Halaman ini dibuat agar dosen dapat melihat nilai seluruh nilai mahasiswa sehingga dosen dapat memonitoring mahasiswa yang telah mengerjakan soal.

3.4.2 Aplikasi *Mobile Learning* Mahasiswa

A. Halaman *Login* Mahasiswa

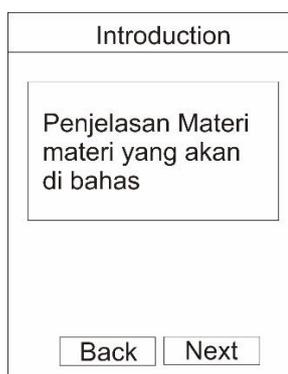


The image shows a login form with a title 'LOGIN' at the top. Below the title, there are two input fields: 'Username' and 'Password'. Below the 'Password' field, there is a 'login' button.

Gambar 3. 9 Rancangan Antarmuka Halaman *Login* Mahasiswa

Gambar 3.9 merupakan halaman login mahasiswa. Halaman ini dibuat untuk memvalidasi data mahasiswa ke server sehingga mahasiswa dapat menuju halaman selanjutnya. Halaman ini juga terdapat sinkronisasi sehingga data mahasiswa tidak terhapus saat mahasiswa melakukan login dengan modul atau perangkat yang berbeda.

B. Halaman *Introduction*



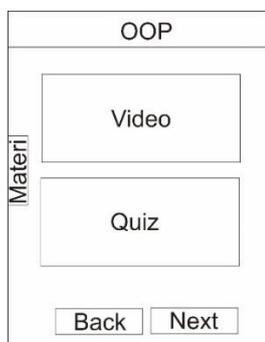
The image shows an introduction page with a title 'Introduction' at the top. Below the title, there is a text box containing the text 'Penjelasan Materi materi yang akan di bahas'. At the bottom of the page, there are two buttons: 'Back' and 'Next'.

Gambar 3. 10 Rancangan Antarmuka Halaman *Introduction*

Setelah mahasiswa selesai melakukan validasi pada halaman *login*, maka mahasiswa akan diarahkan ke halaman *chapter* seperti yang dipaparkan

pada gambar 3.9. pada halaman ini terdapat daftar materi yang akan dipelajari pada halaman selanjutnya.

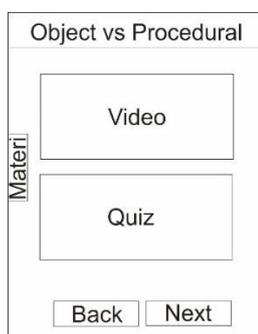
C. Halaman *Object Oriented Programming*



Gambar 3. 11 Perancangan Antarmuka Halaman OOP

Setelah mahasiswa menekan tombol *next* pada halaman *introduction* maka akan menuju kehalaman *Object Oriented Programming* seperti yang di paparkan pada gambar 3.11. Pada halaman ini mahasiswa dapat menonton video dan membaca materi tentang pengertian *Object Oriented Programming*. Pada halaman ini mahasiswa juga dapat mengerjakan quiz sesuai dengan video dan materi yang telah diberikan. Setelah mahasiswa selesai menjawab soal maka mahasiswa dapat menekan tombol *next* untuk menuju kehalaman berikutnya.

D. Halaman OOP vs *Procedural*

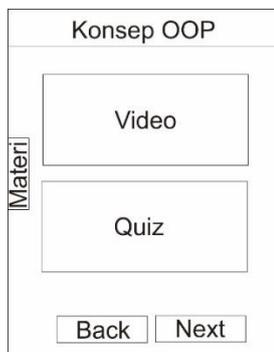


Gambar 3. 12 Perancangan Antarmuka Halaman Lihat Nilai

Pada gambar 3.12 menunjukkan bahwa mahasiswa dapat menonton video dan membaca materi tentang perbedaan OOP vs *Procedural*. Pada halaman ini mahasiswa juga dapat mengerjakan quiz sesuai dengan video dan materi yang

telah diberikan. Setelah mahasiswa selesai menjawab soal maka mahasiswa dapat menekan tombol *next* untuk menuju kehalaman berikutnya.

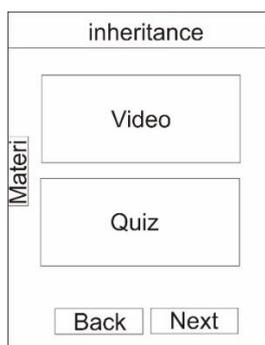
E. Halaman Konsep OOP



Gambar 3. 13 Perancangan Antarmuka Halaman Konsep OOP

Pada gambar 3.13 menunjukkan bahwa mahasiswa dapat menonton video dan membaca materi tentang konsep OOP. Pada halaman ini mahasiswa juga dapat mengerjakan quiz sesuai dengan video dan materi yang telah diberikan. Setelah mahasiswa selesai menjawab soal maka mahasiswa dapat menekan tombol *next* untuk menuju kehalaman berikutnya.

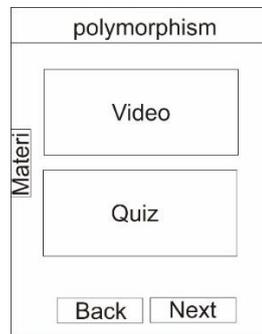
F. Halaman *Inheritance*



Gambar 3. 14 Perancangan Antarmuka Halaman *Inheritance*

Pada gambar 3.14 menunjukkan bahwa mahasiswa dapat menonton video dan membaca materi tentang pengertian *Inheritance*. Pada halaman ini mahasiswa juga dapat mengerjakan quiz sesuai dengan video dan materi yang telah diberikan. Setelah mahasiswa selesai menjawab soal maka mahasiswa dapat menekan tombol *next* untuk menuju kehalaman berikutnya.

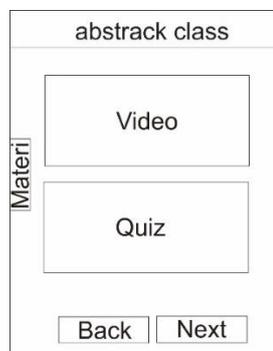
G. Halaman *Polymorphism*



Gambar 3. 15 Perancangan Antarmuka Halaman Polymorphism

Pada gambar 3.15 menunjukkan bahwa mahasiswa dapat menonton video dan membaca materi tentang pengertian *Polymorphism*. Pada halaman ini mahasiswa juga dapat mengerjakan quiz sesuai dengan video dan materi yang telah diberikan. Setelah mahasiswa selesai menjawab soal maka mahasiswa dapat menekan tombol *next* untuk menuju kehalaman berikutnya.

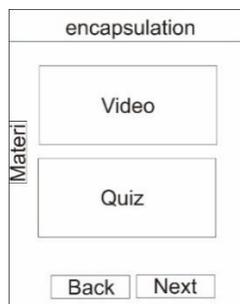
H. Halaman *Abstract Class*



Gambar 3. 16 Rancangan Antarmuka Halaman Abstract Class

Pada gambar 3.15 menunjukkan bahwa mahasiswa dapat menonton video dan membaca materi tentang pengertian *Abstract Class*. Pada halaman ini mahasiswa juga dapat mengerjakan quiz sesuai dengan video dan materi yang telah diberikan. Setelah mahasiswa selesai menjawab soal maka mahasiswa dapat menekan tombol *next* untuk menuju kehalaman berikutnya.

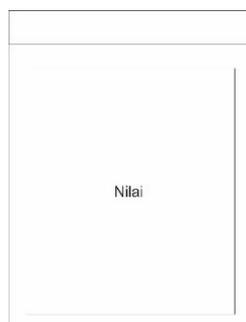
I. Halaman *Encapsulation*



Gambar 3. 17 Rancangan Antarmuka Halaman Encapsulation

Pada gambar 3.15 menunjukkan bahwa mahasiswa dapat menonton video dan membaca materi tentang pengertian *Encapsulation*. Pada halaman ini mahasiswa juga dapat mengerjakan quiz sesuai dengan video dan materi yang telah diberikan.

J. Halaman Lihat Nilai



Gambar 3. 18 Rancangan Antarmuka Halaman Lihat Nilai

Pada gambar 3.13 menunjukkan bahwa mahasiswa dapat melihat nilai yang telah didapatkan setelah mengerjakan semua soal yang terdapat pada halaman sebelumnya.

3.6 Analisis Data

Pada pengembangan aplikasi ini menggunakan metode deskriptif untuk menganalisa data penelitian. Metode deskriptif merupakan metode yang berfungsi untuk menjelaskan atau memberi gambaran terhadap objek yang sedang diteliti melalui data atau sampel yang telah terkumpul sebagaimana adanya tanpa melakukan analisis dan membuat kesimpulan yang berlaku untuk umum. Tujuan

dari penelitian deskriptif ini adalah untuk membuat penjelasan secara sistematis, real dan akurat mengenai hubungan antar fenomena yang sedang diteliti. Deskriptif kuantitatif penelitian yaitu menjelaskan hasil rekayasa perangkat lunak dan menguji tingkat kelayakan perangkat.

Teknik skala likert merupakan metode pengolahan data yang digunakan untuk mengetahui ukuran variable. Untuk memudahkan responden memilih jawaban yaitu dengan menggunakan pengukuran tersebut.

Untuk mengetahui hasil pengukuran skala likert yaitu menggunakan kuesioner yang dibagikan kepada responden. Terdapat 5 pilihan jawaban diantaranya yaitu:

Tabel 3. 5 Skala Likert

No	Jawaban	Poin
1	Sangat Setuju	5
2	Setuju	4
3	Cukup Setuju	3
4	Tidak Setuju	2
5	Sangat Tidak Setuju	1

Hasil tersebut digunakan untuk mengetahui kelayakan komponen-komponen penelitian. Menurut Arikunto (2010), Pembagian kategori kelayakan ada lima. Skala ini memperhatikan rentang dari bilangan persentase. Nilai maksimal yang diharapkan adalah 100% dan minimum 0%.

Berikut adalah tabel jenis kelayakan :

Tabel 3. 6 Kategori Kelayakan

No	Kategori	Persentase
1	Sangat Layak	81% - 100%
2	Layak	61% - 80%
3	Cukup layak	41% - 60%
4	Tidak Layak	21% - 40%
5	Sangat Tidak Layak	<20%