

PENGEMBANGAN APLIKASI MOBILE LEARNING PEMROGRAMAN BAHASA JAVA

Hermawan

Teknik Informatika

Universitas Muhammadiyah Yogyakarta

Jalan Brawijaya, Kasihan, Geblagan, Tamantirto, Kasihan, Bantul, Daerah Istimewa Yogyakarta
55183

hermawans710@gmail.com

Abstract— Saat ini java merupakan bahasa pemrograman yang paling populer digunakan, dan secara luas dimanfaatkan dalam pengembangan berbagai jenis perangkat lunak. Pemrograman bahasa java juga sering digunakan sebagai uji kompetensi profesi programmer oleh Badan Nasional Sertifikasi Profesi (BNSP).

Banyak materi yang harus dipelajari mengenai pemrograman bahasa java, sehingga menyulitkan mahasiswa dalam mempelajari pemrograman bahasa java. Sedikitnya aplikasi berbasis android tentang pembelajaran pemrograman bahasa java yang dapat diakses secara online maupun offline merupakan salah satu penyebab kurangnya pemahaman mahasiswa dalam mempelajari materi pemrograman bahasa java.

Metode yang digunakan adalah metode Software Development Life Cycle (SDLC) dengan model waterfall. Hasil dari pengembangan ini adalah aplikasi pembelajaran pemrograman bahasa java pada perangkat mobile berbasis android. Dengan dibuatnya aplikasi Mobile Learning diharapkan dapat mempermudah sistem pembelajaran dan meningkatkan keefektifan belajar bagi mahasiswa.

Kata Kunci: *Android, Aplikasi Mobile Learning, Pemrograman Bahasa Java*

1 Pendahuluan

Java adalah bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam. Saat ini java merupakan bahasa pemrograman yang paling populer digunakan, dan secara luas dimanfaatkan dalam pengembangan berbagai jenis perangkat lunak. Pemrograman bahasa java juga sering digunakan sebagai uji kompetensi profesi programmer oleh Badan Nasional Sertifikasi Profesi (BNSP).

Banyak materi yang harus dipelajari mengenai pemrograman bahasa java, sehingga menyulitkan mahasiswa dalam mempelajari pemrograman bahasa java. Sedikitnya aplikasi berbasis android tentang pemrograman bahasa java yang dapat diakses secara online maupun offline merupakan salah satu penyebab kurangnya pemahaman mahasiswa dalam mempelajari materi pemrograman bahasa java.

Permasalahan tersebut harus segera dipecahkan dengan membangun modul aplikasi yang memanfaatkan smartphone sebagai media pembelajaran agar memudahkan mahasiswa

ataupun dosen dalam menggunakannya, juga dapat diunggah pada LMS dengan menggunakan standar SCORM. Pengembangan aplikasi yang memanfaatkan smartphone biasanya disebut dengan Mobile Learning.

Mobile Learning adalah salah satu model pembelajaran yang memanfaatkan teknologi informasi dan komunikasi. *Mobile Learning* memberi manfaat menyediakan materi ajar yang dapat diakses kapan saja dan visualisasi materi yang lebih menarik pada konsep pembelajaran tersebut. *Mobile learning* yang sering didefinisikan sebagai *e-learning* melalui perangkat komputasi *mobile*. *Mobile Learning* juga merupakan cara menyampaikan materi pembelajaran elektronik pada alat komputasi *mobile* agar dapat diakses oleh pengguna dimana saja dan kapan saja.

Namun pemanfaatan *smartphone* sebagai media pembelajaran masih sangat minim. Oleh karena itu penulis mengembangkan modul aplikasi *mobile learning* pemrograman bahasa *java* berbasis android agar memudahkan mahasiswa ataupun dosen dalam menggunakannya, juga dapat diunggah pada LMS dengan menggunakan standar SCORM, aplikasi *mobile learning* dapat digunakan secara *online* maupun *offline*, untuk mengaksesnya koneksi internet hanya dibutuhkan ketika *login* dan *upload score* hasil belajar.

2 Metodologi Perancangan

2.1 Metodologi Perancangan

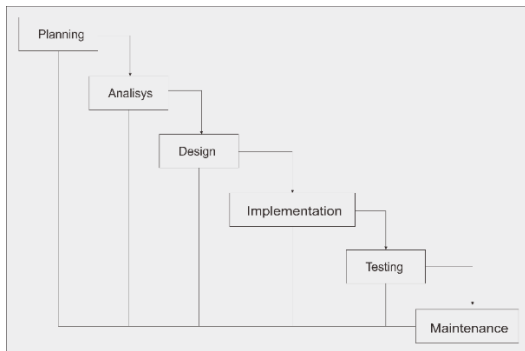
Perancangan terhadap aplikasi *mobile learning* berbasis android dilakukan untuk mengetahui fitur-fitur yang dibutuhkan sehingga menghasilkan *mobile learning* yang layak dan efektif bagi mahasiswa.

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:

- Model *Waterfall*
- Model Iterasi

- Model *Rapid Application Development (RAD)*
- 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 1.1.



Gambar 2. 1 Model Waterfall

2.2 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.

Berikut adalah deskripsi pendefinisian kebutuhan fungsional aplikasi *mobile learning* Pemrograman Bahasa Java pada modul mahasiswa yang digunakan untuk menjelaskan setiap proses yang ada sistem.

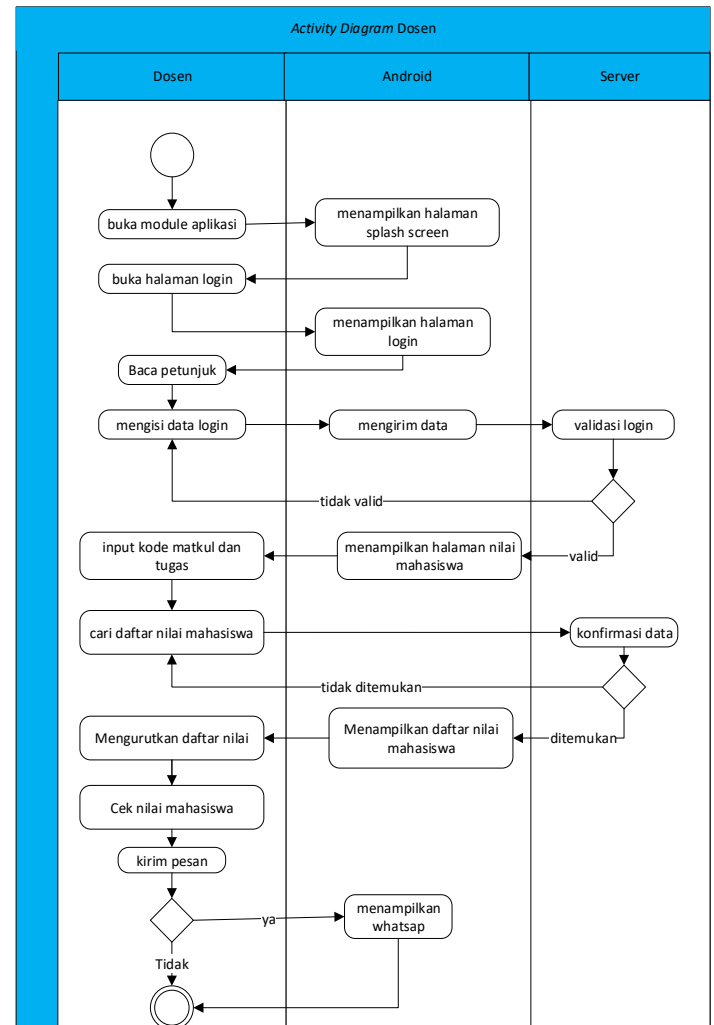
Tabel 1 Tabel Kebutuhan Fungsional

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

Usecase	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

Pada Tabel 1. menggambarkan kebutuhan fungsional pada sistem dimana sistem yang akan dibuat memiliki fungsional seperti *login*, video tutorial, menu materi, menu quiz dan halaman lihat nilai.

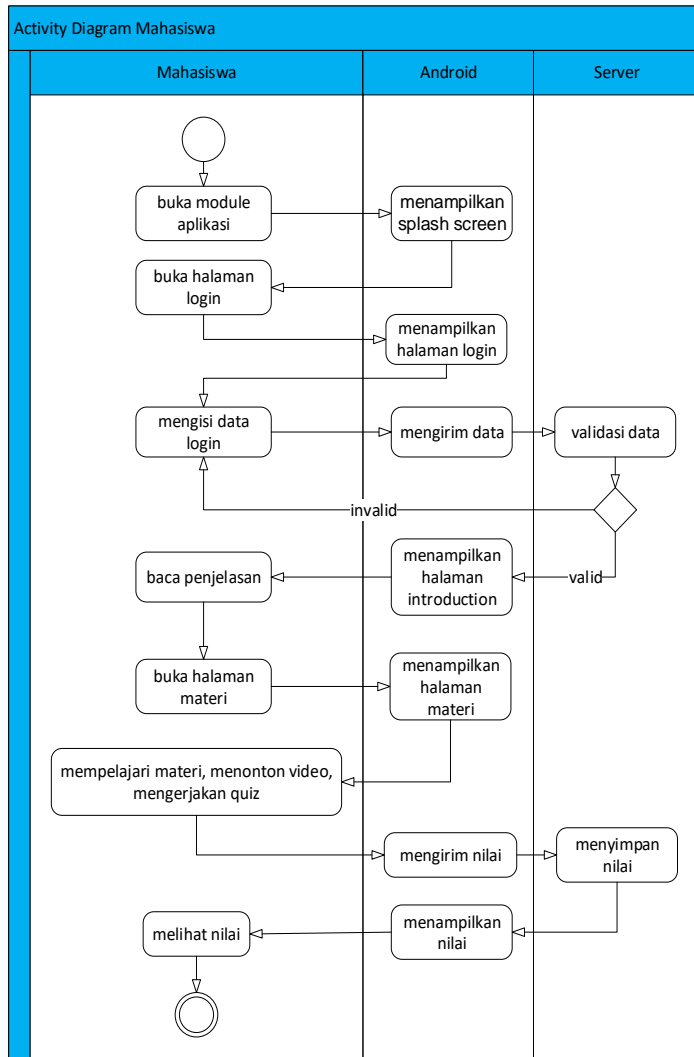
A. Activity Diagram Modul Dosen



Gambar 2. 2 Class Diagram Dosen

Seperti yang ditunjukkan pada gambar 2.2 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. Modul Dosen



Gambar 2. 3 Class Diagram Mahasiswa

Pada gambar 2.3 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 quiz.

C. Perancangan Halaman Login Dosen

The wireframe shows a login page with a vertical label "informasi" on the left. It contains two input fields: "Username" and "Password".

Gambar 2. 4 Rancangan Antarmuka Halaman Login Modul Dosen

Pada halaman login seperti yang ditunjukkan pada gambar 2.4 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.

D. Perancangan Halaman Lihat Nilai Modul Dosen

The wireframe shows a page titled "nilai". At the top, there are two tabs: "Matakuliah" and "Tugas". Below the tabs are two input fields and a "Cari" button. To the right of the search area is a dropdown menu icon. The main content area displays "nilai seluruh mahasiswa".

Gambar 2. 5 Rancangan Antarmuka Halaman Lihat Nilai Modul Dosen

Setelah dosen selesai melakukan validasi pada halaman *login*, gambar 2.5 akan ditampilkan secara otomatis. Halaman ini dibuat agar dosen dapat melihat nilai seluruh nilai mahasiswa sehingga dosen dapat memonitoring mahasiswa yang telah mengerjakan soal.

E. Perancangan Halaman Login Modul Mahasiswa

```
graph TD
    subgraph LOGIN
        U[Username] --- U_in[ ]
        P>Password --- P_in[ ]
        L[login]
    end
```

Gambar 2. 6 Rancangan Antarmuka Halaman Login Mahasiswa

Gambar 2.6 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.

F. Perancangan Halaman Introduction Modul Mahasiswa

```
graph TD
    subgraph Introduction
        T[Penjelasan Materi materi yang akan di bahas]
        B[Back]
        N[Next]
    end
```

Gambar 2. 7 Rancangan Antarmuka Halaman Introduction

Setelah mahasiswa selesai melakukan validasi pada halaman *login*, maka mahasiswa akan diarahkan ke halaman *chapter* seperti yang dipaparkan pada gambar 2.7. pada halaman ini terdapat daftar materi yang akan dipelajari pada halaman selanjutnya.

G. Perancangan Antarmuka Halaman OOP

```
graph TD
    subgraph OOP
        M[Materi]
        V[Video]
        Q[Quiz]
        B[Back]
        N[Next]
    end
```

Gambar 2. 8 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 2.8. 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.

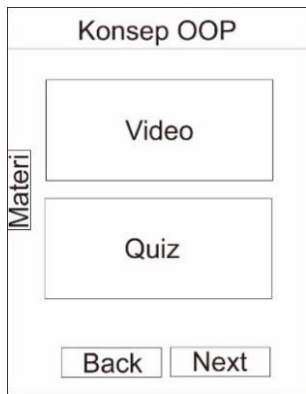
H. Perancangan Antarmuka Halaman Lihat Nilai

```
graph TD
    subgraph Object_vs_Procedural
        M[Materi]
        V[Video]
        Q[Quiz]
        B[Back]
        N[Next]
    end
```

Gambar 2. 9 Perancangan Antarmuka Halaman Lihat Nilai

Pada gambar 2.9 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.

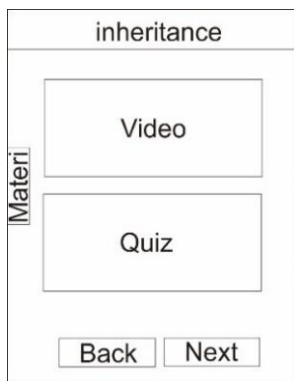
I. Perancangan Antarmuka Halaman Konsep OOP



Gambar 2. 10 Perancangan Antarmuka Halaman Konsep OOP

Pada gambar 2.10 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.

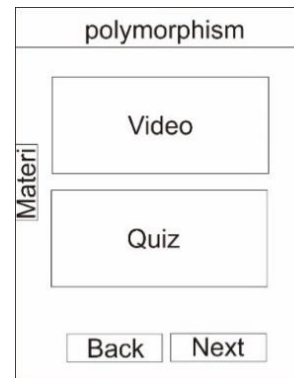
J. Perancangan Antarmuka Halaman *Inheritance*



Gambar 2. 11 Perancangan Antarmuka Halaman *Inheritance*

Pada gambar 2.11 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.

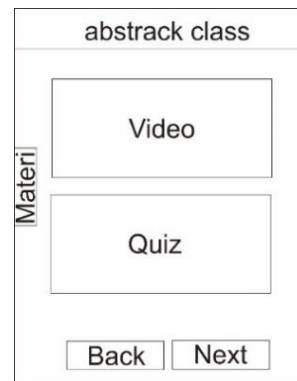
K. Perancangan Antarmuka Halaman Polymorphism



Gambar 2. 12 Perancangan Antarmuka Halaman Polymorphism

Pada gambar 2.12 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.

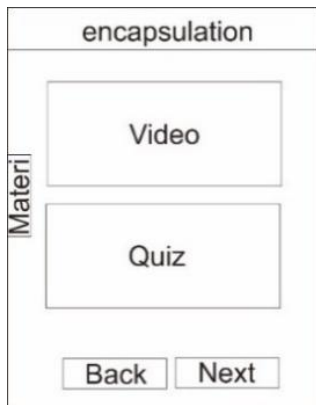
L. Rancangan Antarmuka Halaman Abstract Class



Gambar 2. 13 Rancangan Antarmuka Halaman Abstract Class

Pada gambar 2.13 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.

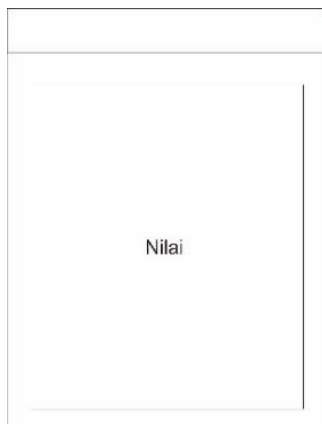
M. Rancangan Antarmuka Halaman Encapsulation



Gambar 2. 14 Rancangan Antarmuka Halaman Encapsulation

Pada gambar 2.14 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.

N. Rancangan Antarmuka Halaman Lihat Nilai



Gambar 2. 15 Rancangan Antarmuka Halaman Lihat Nilai

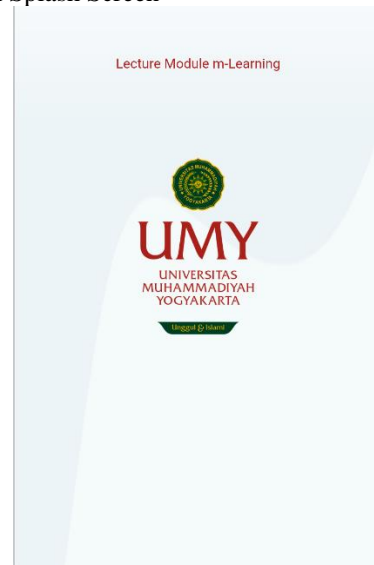
Pada gambar 2.15 menunjukkan bahwa mahasiswa dapat melihat nilai yang telah didapatkan setelah mengerjakan semua soal yang terdapat pada halaman sebelumnya.

3 Hasil

Hasil dari penelitian ini adalah modul aplikasi untuk pembelajaran pemrograman bahasa java yang berisi paket SCORM dan berbasis android. Modul aplikasi itu berupa modul dosen dan modul mahasiswa. Modul dosen berisi nilai seluruh mahasiswa, sedangkan modul mahasiswa berisi modul materi, quiz dan nilai.

3.1 Modul Dosen

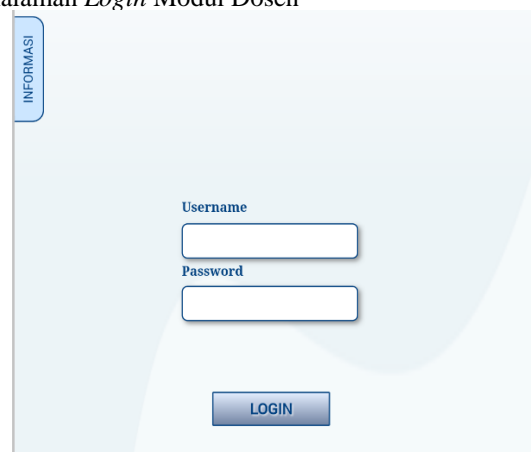
A. Halaman Splash Screen



Gambar 3. 1 Tampilan Halaman *Splash Screen* Modul Dosen

Pada gambar 3.1 merupakan halaman yang akan muncul pertama kali saat aplikasi modul dosen dibuka. Pada proses ini user harus menekan layar agar dapat menuju halaman berikutnya.

B. Halaman *Login* Modul Dosen



Gambar 3. 2 Tampilan Halaman *Login* Modul Dosen

Pada gambar 3.2 merupakan halaman login modul dosen untuk melakukan validasi *username* dan *password*. Pada halaman ini juga terdapat label informasi tentang modul, tatacara pemakaian aplikasi dan alamat/kontak pengaduan apabila terjadi kesalahan pada isi aplikasi, seperti yang di tampilkan pada gambar 4.3.

C. Label Informasi Modul Dosen



Gambar 3. 3 Tampilan Label Informasi Modul Dosen

Pada gambar 3.3 merupakan label informasi untuk mempelajari penggunaan modul dosen dan informasi pengaduan aplikasi.

D. Halaman Lihat Nilai Modul Dosen

No	Nim	WhatsApp	Aktivitas	Score
1	20180140105	081380128515	Class	90
2	20180140098	087886883665	Class	90
3	20180140087	089665851565	Class	90
4	20180140069	081258348963	Class	90
5	20180140082	082139527186	Class	90
6	20180140065	081347509154	Class	90
7	20180140123	082198874753	Class	80
8	20180140091	082158565173	Class	80
9	20180140055	082143578882	Class	80
10	20180140052	082242112183	Class	80
11	20180140008	082133922316	Class	80
12	20180140029	082277215266	Class	80
13	20180140096	085357929545	Class	70
14	20180140104	081326711418	Class	70
15	20180140084	085267664015	Class	70
16	20180140054	081326855427	Class	70
17	20180140030	081914578684	Class	70
18	20180140042	085228664552	Class	70
19	20180140116	081335415422	Class	60
20	20180140101	081558857762	Class	60
21	20180140090	085959455704	Class	60
22	20180140115	082227013191	Class	50
23	20180140114	081242171125	Class	100
24	20180140079	085210332132	Class	100
25	20180140093	082289803751	Class	100
26	20180140037	087888451559	Class	100
27	20180140067	082353133841	Class	100
28	20180140113	085365918702	Class	100
29	20180140099	081350505151	Class	0
30	20180140094	082269311563	Class	0

Gambar 3. 4 Tampilan Halaman Lihat Nilai Modul Dosen

Pada gambar 3.4 diatas menunjukkan halaman lihat nilai ini dibuat supaya dosen dapat mengetahui nilai masing-masing mahasiswanya. Pada halaman ini terdapat kolom kode matakuliah, tugas dan tombol cari untuk mencari nilai mahasiswa sesuai kode matakuliah dan tugas yang diberikan.

3.2 Modul Mahasiswa

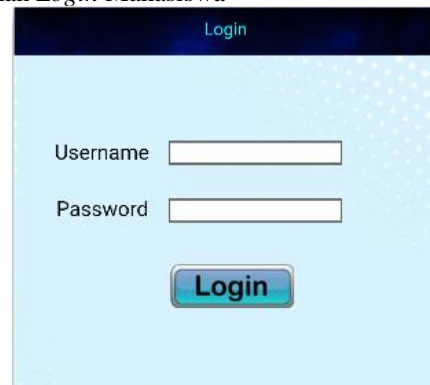
A. Halaman Splash Screen



Gambar 3. 5 Tampilan Halaman Splash Screen Modul Mahasiswa

Pada gambar 3.5 diatas merupakan halaman yang akan muncul pertama kali saat aplikasi modul mahasiswa dibuka. Pada proses ini user harus menekan layar agar dapat menuju halaman berikutnya.

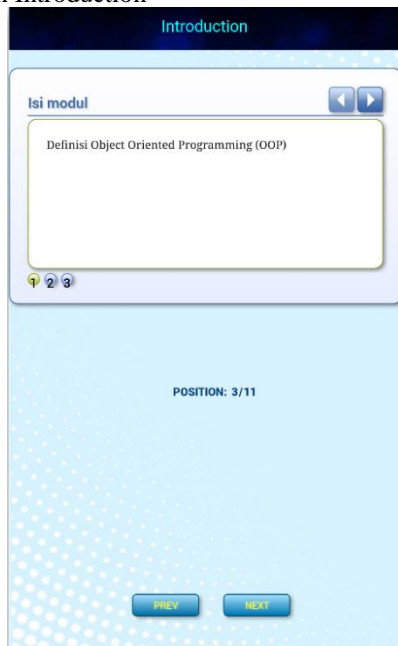
B. Halaman Login Mahasiswa



Gambar 3. 6 Tampilan Halaman Login Modul Mahasiswa

Pada gambar 4.6 merupakan halaman login modul mahasiswa untuk melakukan validasi *username* dan *password*.

C. Halaman Introduction



Gambar 3. 7 Tampilan Halaman Introduction

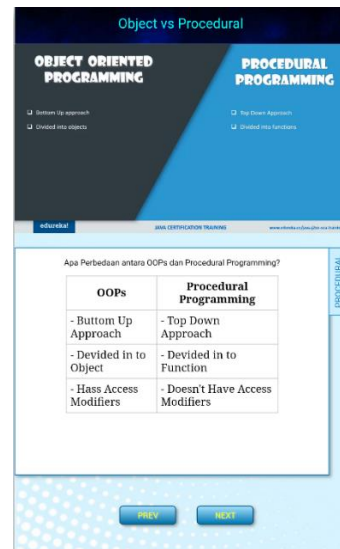
Pada gambar 3.7 merupakan halaman judul materi yang akan dipelajari pada halaman selanjutnya.

D. Halaman Object Oriented Programming



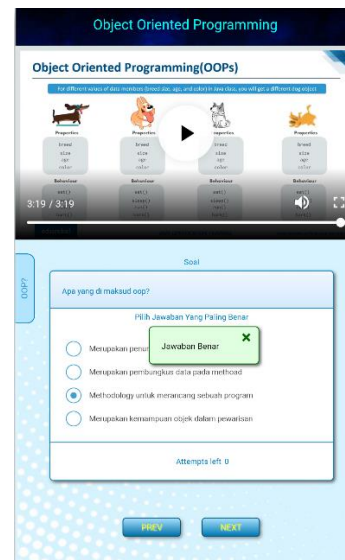
Gambar 3. 8 Halaman OOP Tampilan Video Materi

Pada gambar 3.8 merupakan tampilan halaman oop untuk mempelajari materi melalui video. Setelah video selesai diputar maka *quiz* akan di tampilkan secara otomatis.



Gambar 3. 9 Tampilan Label Ringkasan Materi Halaman OOP

Pada gambar 3.9 merupakan tampilan yang berisi materi ringkasan pada video. Tampilan ini akan muncul apabila tombol menu oop ditekan.



Gambar 3. 10 Tampilan Label Quiz Halaman OOP

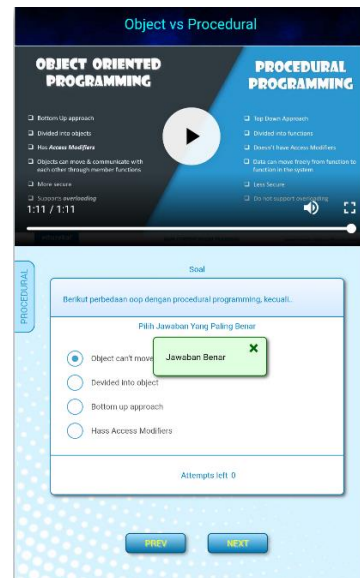
Pada gambar 3.10 merupakan tampilan yang digunakan mahasiswa untuk mengerjakan quiz pada halaman oop. Pada halaman ini juga terdapat *feedback* apakah jawaban benar atau salah.

E. Halaman OOP vs Procedural



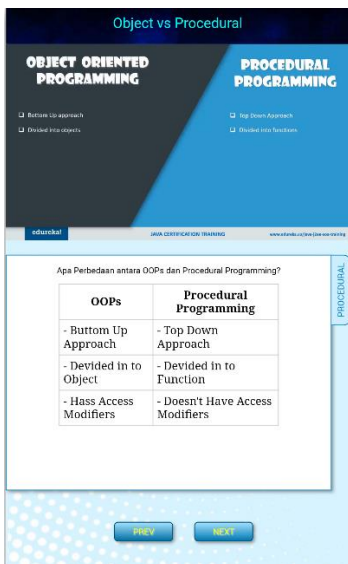
Gambar 3. 11 Halaman Object vs Procedural Tampilan Video Materi

Pada gambar 3.11 merupakan tampilan halaman *object vs procedural* untuk mempelajari materi melalui video. Setelah video selesai diputar maka quiz akan di tampilkan secara otomatis.



Gambar 3. 13 Tampilan Label Quiz Halaman *Object vs Procedural*

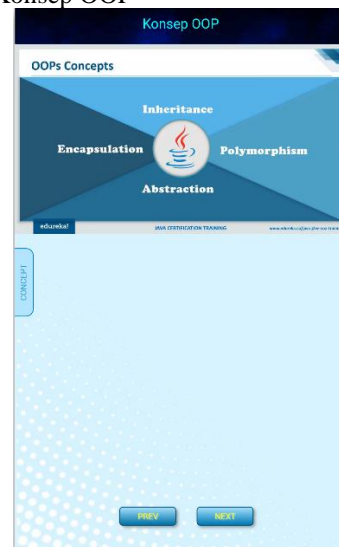
Pada gambar 3.13 merupakan tampilan yang digunakan mahasiswa untuk mengerjakan quiz pada halaman *object vs procedural*. Pada halaman ini juga terdapat *feedback* apakah jawaban benar atau salah.



Gambar 3. 12 Tampilan Label Ringkasan Materi *Object vs Procedural*

Pada gambar 4.12 merupakan tampilan yang berisi materi ringkasan pada video. Tampilan ini akan muncul apabila tombol menu *object vs procedural* ditekan.

F. Halaman Konsep OOP



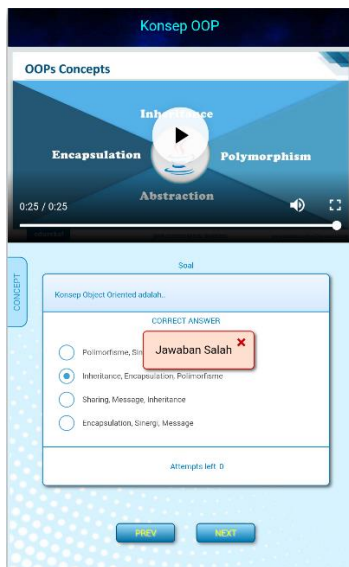
Gambar 3. 14 Halaman Konsep OOP Tampilan Video Materi

Pada gambar 3.14 merupakan tampilan halaman konsep oop untuk mempelajari materi melalui video. Setelah video selesai diputar maka quiz akan di tampilkan secara otomatis.



Gambar 3. 15 Tampilan Label Ringkasan Materi Halaman Konsep OOP

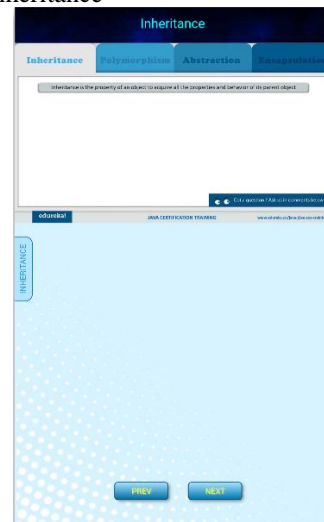
Pada gambar 3.15 merupakan tampilan yang berisi materi ringkasan pada video. Tampilan ini akan muncul apabila tombol menu konsep oop ditekan.



Gambar 3. 16 Tampilan Label Quiz Halaman Konsep OOP

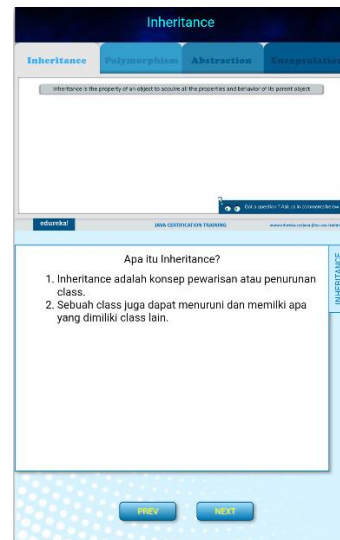
Pada gambar 3.16 merupakan tampilan yang digunakan mahasiswa untuk mengerjakan quiz pada halaman konsep oop. Pada halaman ini juga terdapat *feedback* apakah jawaban benar atau salah.

G. Halaman Inheritance



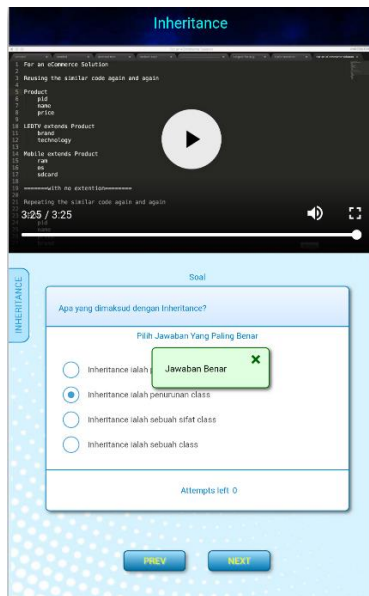
Gambar 3. 17 Halaman *Inheritance* Tampilan Video Materi

Pada gambar 3.17 merupakan tampilan halaman *inheritance* untuk mempelajari materi melalui video. Setelah video selesai diputar maka quiz akan ditampilkan secara otomatis.



Gambar 3. 18 Tampilan Label Ringkasan Materi Halaman *Inheritance*

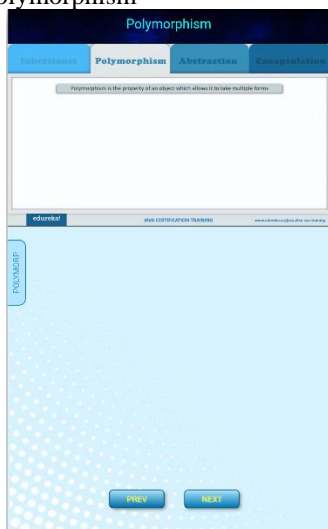
Pada gambar 3.18 merupakan tampilan yang berisi materi ringkasan pada video. Tampilan ini akan muncul apabila tombol menu *inheritance* ditekan.



Gambar 3. 19 Tampilan Label Quiz Halaman *Inheritance*

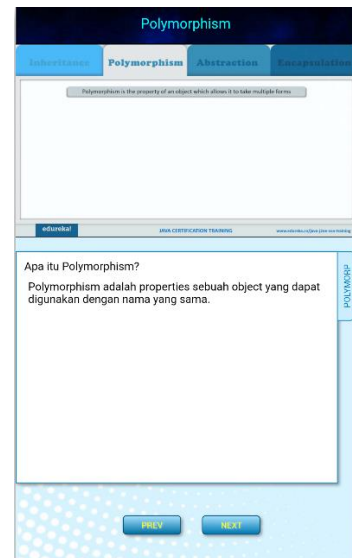
Pada gambar 3.19 merupakan tampilan yang digunakan mahasiswa untuk mengerjakan quiz pada halaman *inheritance*. Pada halaman ini juga terdapat *feedback* apakah jawaban benar atau salah.

H. Halaman Polymorphism



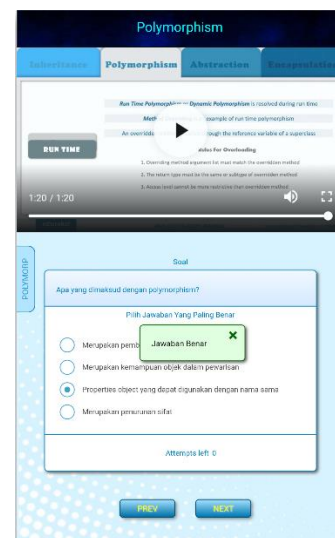
Gambar 3. 20 Halaman *Polymorphism* Tampilan Video Materi

Pada gambar 3.20 merupakan tampilan halaman *polymorphism* untuk mempelajari materi melalui video. Setelah video selesai diputar maka quiz akan ditampilkan secara otomatis.



Gambar 3. 21 Tampilan Label Ringkasan Materi Halaman *Polymorphism*

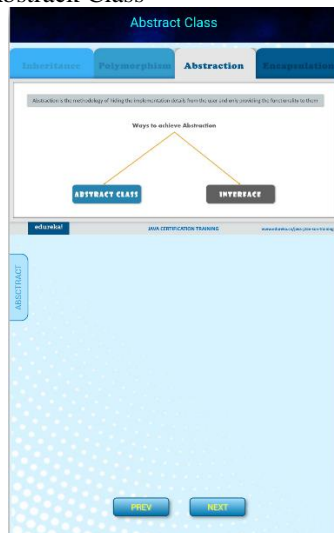
Pada gambar 3.21 merupakan tampilan yang berisi materi ringkasan pada video. Tampilan ini akan muncul apabila tombol menu *polymorphism* ditekan.



Gambar 3. 22 Tampilan Label Quiz Halaman *Polymorphism*

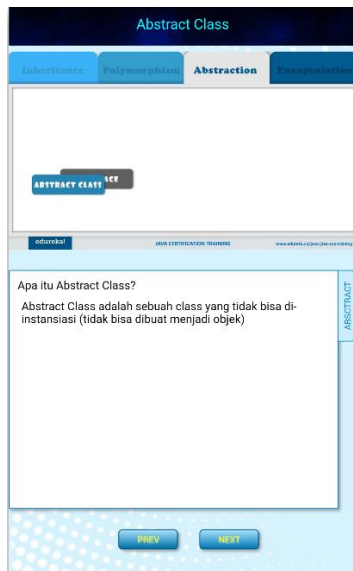
Pada gambar 3.22 merupakan tampilan yang digunakan mahasiswa untuk mengerjakan quiz pada halaman *polymorphism*. Pada halaman ini juga terdapat *feedback* apakah jawaban benar atau salah.

I. Halaman Abstract Class



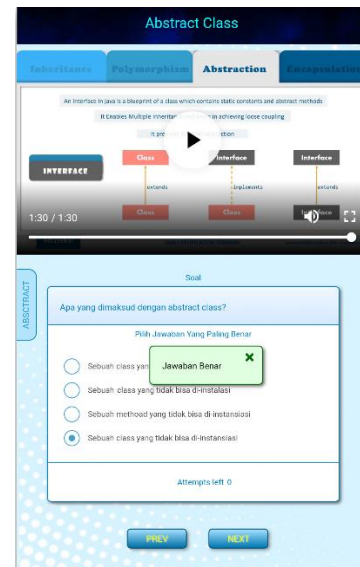
Gambar 3. 23 Halaman *Abstract Class* Tampilan Video Materi

Pada gambar 3.23 merupakan tampilan halaman *abstract class* untuk mempelajari materi melalui video. Setelah video selesai diputar maka quiz akan ditampilkan secara otomatis.



Gambar 3. 24 Tampilan Label Ringkasan Materi Halaman *Abstract Class*

Pada gambar 3.24 merupakan tampilan yang berisi materi ringkasan pada video. Tampilan ini akan muncul apabila tombol menu *abstract class* ditekan.



Gambar 3. 25 Tampilan Label Quiz Halaman *Abstract Class*

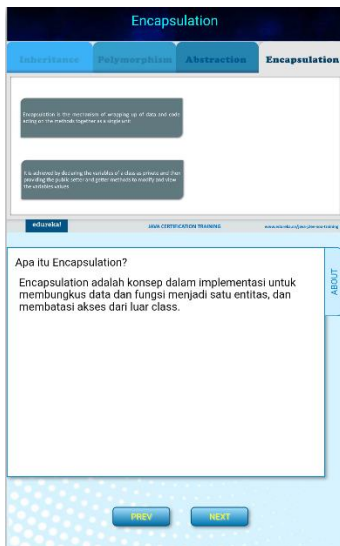
Pada gambar 3.25 merupakan tampilan yang digunakan mahasiswa untuk mengerjakan quiz pada halaman *abstract class*. Pada halaman ini juga terdapat *feedback* apakah jawaban benar atau salah.

J. Halaman Encapsulation



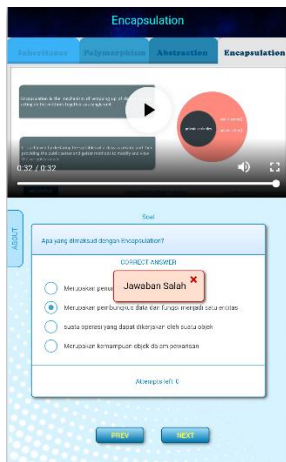
Gambar 3. 26 Halaman *Encapsulation* Tampilan Video Materi

Pada gambar 3.24 merupakan tampilan halaman *abstract class* untuk mempelajari materi melalui video. Setelah video selesai diputar maka quiz akan ditampilkan secara otomatis.



Gambar 3. 27 Tampilan Label Ringkasan Materi Halaman *Encapsulation*

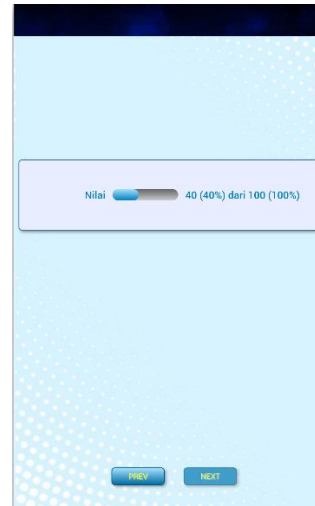
Pada gambar 3.27 merupakan tampilan yang berisi materi ringkasan pada video. Tampilan ini akan muncul apabila tombol menu *abstract class* ditekan.



Gambar 3. 28 Tampilan Label Quiz Halaman *Encapsulation*

Pada gambar 3.28 merupakan tampilan yang digunakan mahasiswa untuk mengerjakan quiz pada halaman *abstract class*. Pada halaman ini juga terdapat *feedback* apakah jawaban benar atau salah.

K. Halaman Lihat Nilai



Gambar 3. 29 Tampilan Halaman Lihat Nilai Modul Mahasiswa

Pada gambar 3.29 merupakan tampilan halaman yang digunakan untuk melihat nilai mahasiswa yang telah mengerjakan quiz.

4 Pembahasan

Setelah pengujian aplikasi dan pengujian kelayakan dilakukan, dapat dilihat bahwa aplikasi *mobile learning* ini dapat membantu mahasiswa dalam kegiatan belajar. Hal ini terlihat dari hasil pengolahan data yang penulis lakukan menggunakan kuesioner. Hasil pengolahan data menunjukkan kelayakan aplikasi yang signifikan.

Dengan hasil tersebut, dapat dikatakan bahwa tujuan dari penelitian ini sudah tercapai, yaitu membantu dan memudahkan mahasiswa jurusan Teknik Informatika untuk mempelajari pemrograman bahasa java. Setelah tujuan penelitian tercapai, diharapkan manfaat penelitian juga tercapai yaitu meningkatkan pengetahuan dalam pembuatan aplikasi *mobile learning* dan membantu mahasiswa dalam kegiatan belajar.

Penelitian ini juga masih bisa dikembangkan dikemudian hari, mengingat aplikasi yang telah dibangun ini masih memiliki jangkauan materi yang terbatas. Aplikasi dapat dikembangkan lebih lanjut, seperti penambahan video tutorial, materi, soal dan masih banyak lagi.