

BAB II

TINJAUAN PUSTAKA, LANDASAN TEORI DAN PERANCANGAN

2.1 Tinjauan Pustaka

Pembelajaran yang disusun dengan tujuan menggunakan sistem elektronik atau komputer sehingga mampu mendukung proses pembelajaran dan untuk memenuhi kebutuhan belajar mengajar, sekarang telah ada media pembelajaran di internet yang biasa disebut "*e-learning*", di *e-learning* terdapat beberapa model salah satu modelnya yaitu CAL (*Computer Aided Learning*).

CAL (*Computer Aided Learning*) adalah sistem komputer yang dapat menghantarkan pembelajaran secara langsung pada mahasiswa dan memungkinkan lebih interaktif dengan program pembelajaran yang telah tersedia. Adapun jenis pembelajaran dapat menjadi lebih interaktif dan efektif dengan menggunakan CAL antara lain: tutorial (32%), latihan (22%), penemuan (20%), simulasi (13%), permainan (3%) dan lain-lain (10%). Pengembangan media komputer khususnya CAL sudah pernah dikembangkan sebelumnya namun masih banyak persamaan dalam penggunaan perangkat lunak, yaitu *macromedia flash*, tetapi berbeda pada materi (*content*) dan bentuk tampilan (*interface*) (Susanti, Purwoko, & Zulkardi, 2008).

Sedangkan dalam penelitian Adipranata, Liliana, dan Dewi (2005) yang berjudul pembuatan *computer aided learning* (CAL) mengenai hewan vertebrata bagi siswa SMP, perangkat lunak ini dapat membantu siswa dalam mempelajari hewan vertebrata. Perangkat lunak yang digunakan sebagai perangkat penunjang dalam pembuatannya yaitu *Macromedia Flash MX 2004* sebagai *interface* dan *Microsoft Access* sebagai *database*.

Dari tinjauan pustaka di atas telah banyak dilakukan penelitian tentang pembuatan CAL berbasis dekstop yang dapat membantu pengguna (*user*) dalam proses pembelajaran, namun masih banyak juga kekurangan dari pembuatan CAL berbasis dekstop ini karena belum berbasiskan *windows phone* dan belum menggunakan *SCORM Module*.

Dengan demikian untuk memudahkan penggunaan dibangun modul aplikasi secara *offline* juga fleksibel karena berbasis *windows phone* dan menarik karena menurut (Hendri, 2014) SCORM mengandung konten media digital seperti teks, gambar, suara, animasi yang membuat menariknya suatu konten.

Maka penulis membuat aplikasi modul yang berbasis *windows phone* dan menggunakan SCORM *Module* agar proses pembelajaran dapat dilakukan secara *offline*, fleksibel dan juga menarik.

2.2 Landasan Teori

2.2.1 E-Learning

E-Learning adalah suatu sistem pembelajaran yang berbasis teknologi dan tentunya internet sebagai sarana pembelajaran *e-learning* merupakan singkatan dari *Electronic Learning*, dengan menggunakan media elektronik khususnya internet sebagai sistem inti dalam kegiatan pembelajaran.

Menurut (Jaya, 2012) karakteristik *E-Learning* adalah:

1. Memanfaatkan jasa teknologi elektronik.
2. Memanfaatkan keunggulan komputer (digital media dan komputer *networks*).
3. Menggunakan bahan ajar yang bersifat mandiri (*self learning materials*) kemudian disimpan di komputer, sehingga dapat diakses oleh dosen dan mahasiswa kapan saja dan dimana saja.

2.2.2 LMS (*Learning Management System*)

LMS (*Learning Management System*) adalah pengelolaan interaksi proses pembelajaran berbasis TI melalui website. Sedangkan (Munir, 2010) menyebutkan bahwa LMS berisi materi dalam kompetensi pedagogik dan profesional, dibuat dengan kemasan multimedia (teks, animasi, video, *sound*, *FX*), diberikan sebagai *supplement dan enrichment* bagi pengembangan kompetensi pembelajar. LMS membuat inovasi dalam dunia TI, yang berbasis *web online learning*, multimedia dan *video conference*. LMS pembelajaran berbasis *web* dikembangkan secara *dynamic e-learning*

2.2.3 SCORM

Sharable Content Object Reference Model (SCORM) adalah standar *e-learning* yang dikeluarkan ADL dalam upayanya untuk mulai menyeragamkan pengembangan sistem *e-learning* berbasis teknologi web yang disebut *Learning Management Systems (LMS)*. SCORM menggunakan pendekatan object oriented dan memandang setiap *learning object* atau *content object* sebagai sekumpulan objek yang dapat disatukan untuk membangun suatu sistem yang lebih besar (Pujianto & Ahmad, 2012).

Proses SCORM dimulai dari pembuatan SCO (*Shareable Content Object*). SCO disusun dari *assets* (media digital seperti teks, gambar, suara, animasi dan data lain yang dapat di-render oleh *web browser*). Setelah itu ditambahkan metadata yang berisi informasi dari SCO tersebut. Untuk mengatur proses sekuensial dari materi pembelajaran dan alur navigasi antar SCO, dalam *manifest* dituliskan daftar isi dan sekuensial pembelajaran. Pada saat pembelajaran berlangsung, proses dilakukan oleh RTE. Kunci pada proses ini adalah kemampuan SCO untuk berkomunikasi dengan LMS. Pada sisi LMS, proses komunikasi itu dilakukan oleh *API Adapter*. Saat mahasiswa melakukan *request* pada SCO, SCO tersebut mencari *API Adapter* pada LMS. Disinilah letak keunggulan SCORM yaitu SCO bisa berkomunikasi dengan LMS tentang informasi mahasiswa, seperti: nama, *preference*, dan nilai. Selain itu komunikasi juga dilakukan untuk merekam aktivitas mahasiswa (*user tracking*).

2.2.4 CAL (*Computer Aided Learning*)

CAL merupakan kemajuan dari metode pembelajaran berbasis komputer, dimana mahasiswa bisa lebih mudah, cepat dalam memahami dan menerapkan informasi apa yang dipelajari melalui suara dan gambar yang menarik (audio visual).

Menurut Adipranata, Liliana, dan Dewi (2005) ada beberapa keuntungan menggunakan CAL, antara lain:

1. Pembelajaran secara individu. Dengan adanya CAL, bisa membantu mereka untuk belajar secara mandiri.

2. Minat masyarakat yang tinggi dalam menggunakan komputer. Masyarakat telah memiliki minat yang tinggi dalam menggunakan komputer. Jika komputer dilengkapi dengan aplikasi pengajaran dengan penyampaian yang menarik, diharapkan dapat menarik minat siswa dalam belajar.
3. Pembelajaran yang cepat. Keistimewaan penggunaan komputer sebagai proses belajar mengajar adalah waktu yang diperlukan untuk mempelajari sesuatu pelajaran lebih cepat dan fleksibel.
4. Audio Visual. Dengan menggunakan audio visual menyebabkan materi yang dipelajari menjadi lebih menarik.

2.3 Perangkat Lunak Pembangun Modul

2.3.1 JavaScript

JavaScript merupakan bahasa pemrograman *web* yang bersifat *Client Side Programming Language*. Menurut (Yatini, 2014) *JavaScript* adalah bahasa scripting kecil, ringan, berorientasi objek yang ditempelkan pada kode HTML dan di proses di sisi *client*. Aplikasi *client* yang dimaksud merujuk kepada *web browser*. Untuk menjalankan *JavaScript*, membutuhkan aplikasi *text editor*, dan *web browser*. *JavaScript* memiliki fitur: *high-level programming language*, *client-side*, *loosely typed*, dan berorientasi objek.

2.3.2 CourseLab 2.4

CourseLab merupakan *software* yang digunakan untuk menyusun bahan ajar multimedia berbasis *e-Learning* yang *powerfull* dan mudah digunakan. *CourseLab* menawarkan lingkungan WYSIWYG (*What You See Is What You Get*) yang bebas dari pemrograman untuk menghasilkan bahan ajar interaktif yang dapat di publikasikan di Internet, *Learning Management Systems (LMS)*, serta *CD-ROM*.

Aplikasi *Courselab 2.4* merupakan aplikasi gratis yang dapat digunakan untuk membuat bahan ajar. Dengan menggunakan aplikasi ini kita dapat membuat bahan ajar yang nantinya dapat di integrasikan dengan *Learning Management System* berbasis *web*.

Menurut (Raharjo, 2011) keunggulan *software CourseLab* sebagai pembuat media interaktif dibandingkan dengan *software* yang sejenis adalah:

1. Bersifat *freeware*.
2. Dapat digunakan untuk membuat soal evaluasi.
3. Kompatibel dengan *software* LMS yang dipakai untuk mendesain web pembelajaran.
4. Penggunaannya relatif praktis, tidak terlalu banyak menggunakan *script* pemrograman.

2.3.3 HTMLExecutable 4.8

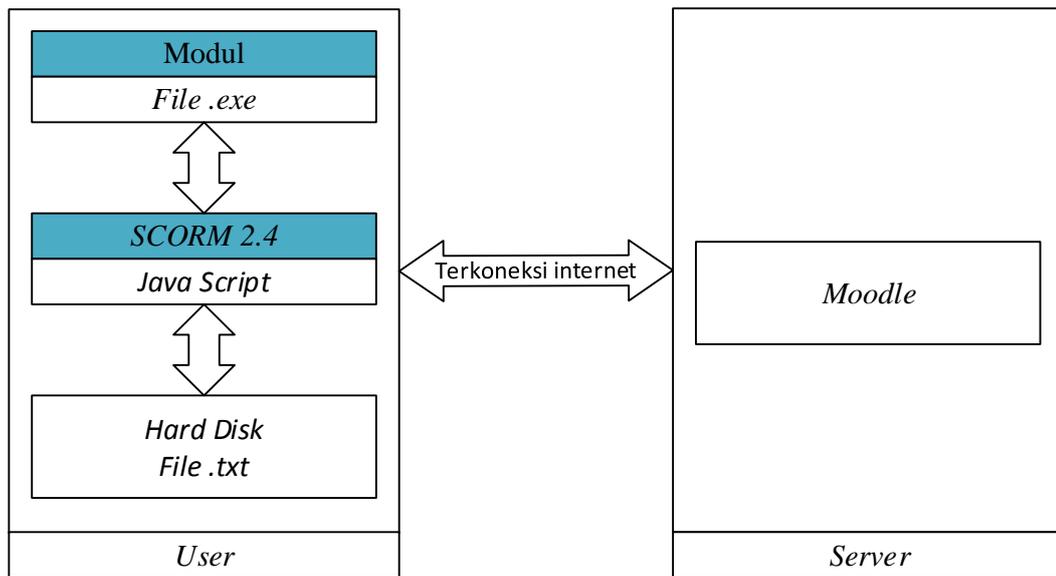
HTMLExecutable 4.8 adalah perangkat lunak yang digunakan untuk *convert* file HTML yang berisi gambar, *audio*, *video* dan lain-lain untuk menjadi *file .exe* yang bisa di jalankan di semua *device windows* khususnya *windows phone*.

2.4 Perancangan

2.4.1 Arsitektur Software

Arsitektur *software* adalah gambaran dari berbagai komponen atau elemen perangkat lunak yang memiliki hubungan atau batasan. Dalam modul aplikasi dibutuhkan sebuah rancangan yang tepat dan matang, agar semua tahapan pembangunan bisa berjalan dengan lancar.

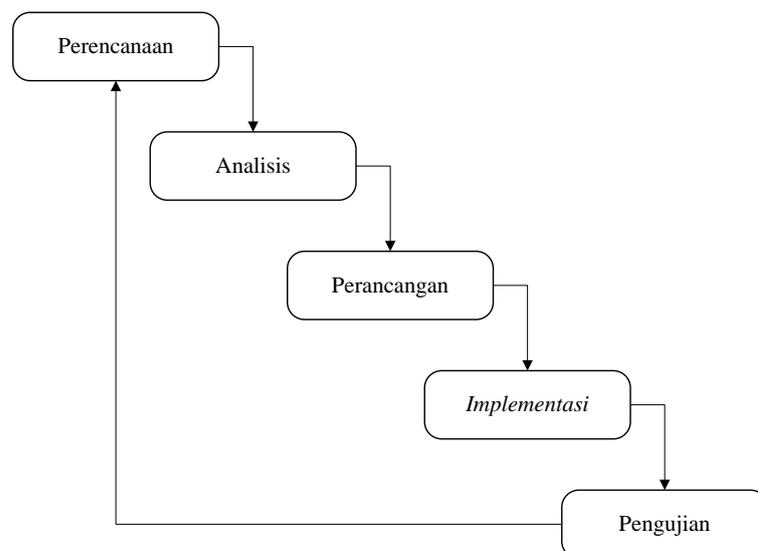
Gambar 2.1 menjelaskan hubungan antara *user* dan *server* termasuk elemen yang ada di dalamnya. Pada sisi *client* aplikasi yang digunakan adalah modul pembelajaran berformat *.exe* (berbasis *windows*). *File* tersebut merupakan hasil publishing dari paket SCORM yang telah dibuat. Di dalam paket SCORM terdapat *script* bahasa pemrograman menggunakan bahasa *Javascript*, *script* tersebut berfungsi untuk mengirimkan data nilai mahasiswa pada modul yang digunakan mahasiswa dan melihat data nilai mahasiswa dari modul yang digunakan oleh dosen. Modul yang digunakan mahasiswa menyimpan data nilai hasil mengerjakan soal di dalam *hardisk* dalam format *file .txt*, sehingga ketika memiliki koneksi internet mahasiswa bisa mengunggah nilai tersebut.



Gambar 2.1 Arsitektur software

2.4.2 Langkah Penelitian

Penelitian ini menggunakan model *system* pengembangan atau rekayasa *system* informasi SDLC (*Software Development Life Cycle*). Metode SDLC adalah metode yang menggunakan pendekatan sistem yang disebut pendekatan air terjun (*waterfall approach*) dimana setiap tahapan sistem akan dikerjakan secara berurutan menurun dimulai dari perencanaan, analisa, perancangan, *implementasi*, dan pengujian. SDLC dilakukan secara *top-down*, oleh karena itu SDLC disebut pendekatan air terjun. Diagram model SDLC di jelaskan pada gambar 2.2.



Gambar 2.2 Software Development Life Cycle

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.

2. Tahap Analisis

Tahap analisis dilakukan dengan melakukan analisa kebutuhan *software* yang diharapkan oleh mahasiswa (*user*). Informasi kegunaan *software* diperoleh melalui wawancara dan diskusi dari beberapa mahasiswa. Dari proses analisa disimpulkan beberapa perangkat keras dan perangkat lunak untuk mendukung berjalannya pembangunan dan implementasi modul aplikasi, antara lain:

A. Perangkat keras (*Hardware*)

- *Personal Computer* (PC) / Laptop.
- 64 bit *Architecture Processor*.
- 4GB *Random Access Memory* (RAM).
- Sistem Operasi Windows 8.1.

B. Perangkat lunak (*Software*)

- *Courselab 2.4*
- *HTMLExecutable 4.8*

3. Tahap Perancangan

Tahap perancangan ini merupakan tahapan dalam mencari solusi permasalahan yang didapat dari tahapan analisis. Perancangan sistem dibutuhkan untuk membantu proses pembangunan modul aplikasi.

4. Tahap Implementasi

Tahap *implementasi* adalah tahap dimana perencanaan, analisis dan rancangan diwujudkan atau di implementasikan ke situasi nyata dengan pemilihan perangkat serta *design (coding)*, untuk implementasinya yaitu memasukan *design* yang sudah dibuat.

5. Tahap Pengujian

Tahap pengujian ini merupakan tahapan yang sangat menentukan apakah modul aplikasi yang dibuat sudah sesuai dengan kebutuhan pengguna (*user*) atau belum. Tujuan tahap pengujian ini adalah meminimalisir kemungkinan aplikasi *error* sehingga modul aplikasi yang telah di buat bisa berjalan sebagai mana mestinya.

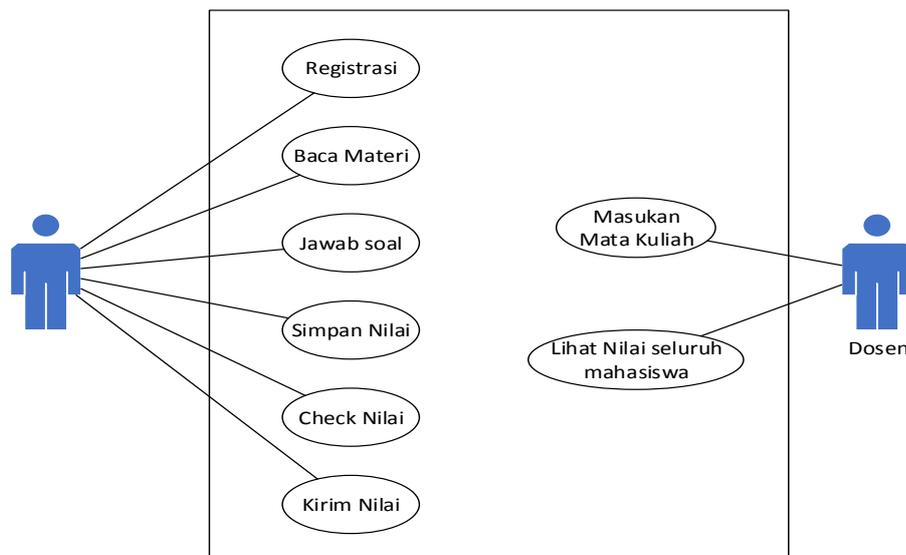
2.4.3 Rancangan

Dengan modul aplikasi ini mahasiswa dapat belajar dan menjawab soal tanpa harus selalu terhubung / terkoneksi dengan internet atau dapat dikerjakan secara *offline*. Hasil menjawab soal pada modul aplikasi dapat disimpan di memori internal *windows phone* terlebih dahulu lalu dapat di *upload* saat terkoneksi dengan internet. Rancangan yang digunakan dalam modul aplikasi adalah *Unified Markup Language (UML)*. Model UML yang dipakai dalam pengembangan aplikasi yaitu *use case diagram* dan *activity diagram* dilanjutkan pembuatan antar muka.

2.4.4 Use Case Diagram

Use case diagram merupakan permodelan untuk modul aplikasi yang akan dibuat. *Use case diagram* digunakan untuk mengetahui fungsi yang ada dalam aplikasi dan siapa saja yang berhak menggunakan fungsi tersebut.

Use case diagram gambar 2.5 pada sisi Mahasiswa dan Dosen.



Gambar 2.3 Use Case Diagram

Pada sisi Mahasiswa:

1. Mahasiswa membuka modul aplikasi yang telah dimiliki.
2. Registrasi, masukan NIM & mata kuliah agar sesuai dengan Nim mahasiswa tersebut dan sesuai dengan mata kuliah yang ingin di kerjakan.
3. Membaca materi kuliah yang telah di buat oleh dosen.
4. Menjawab soal yang telah dibuat oleh dosen berdasarkan *materi* pada *slide* sebelumnya.
5. Selesai mengerjakan soal mahasiswa menyimpan nilai pada internal memori *windows phone*.
6. Mahasiswa dapat melihat (*Check*) nilai sebelum dikirim ke *server*.
7. Kirim nilai adalah proses pengiriman nilai ke *server* dari hasil mahasiswa mengerjakan soal.

Pada sisi Dosen:

1. Dosen membuka modul aplikasi yang telah dimiliki.
2. Dosen memasukan mata kuliah yang ingin dilihat nilainya.
3. Melihat nilai dari hasil dari mahasiswa mengerjakan soal yang telah dikirim ke *server*.

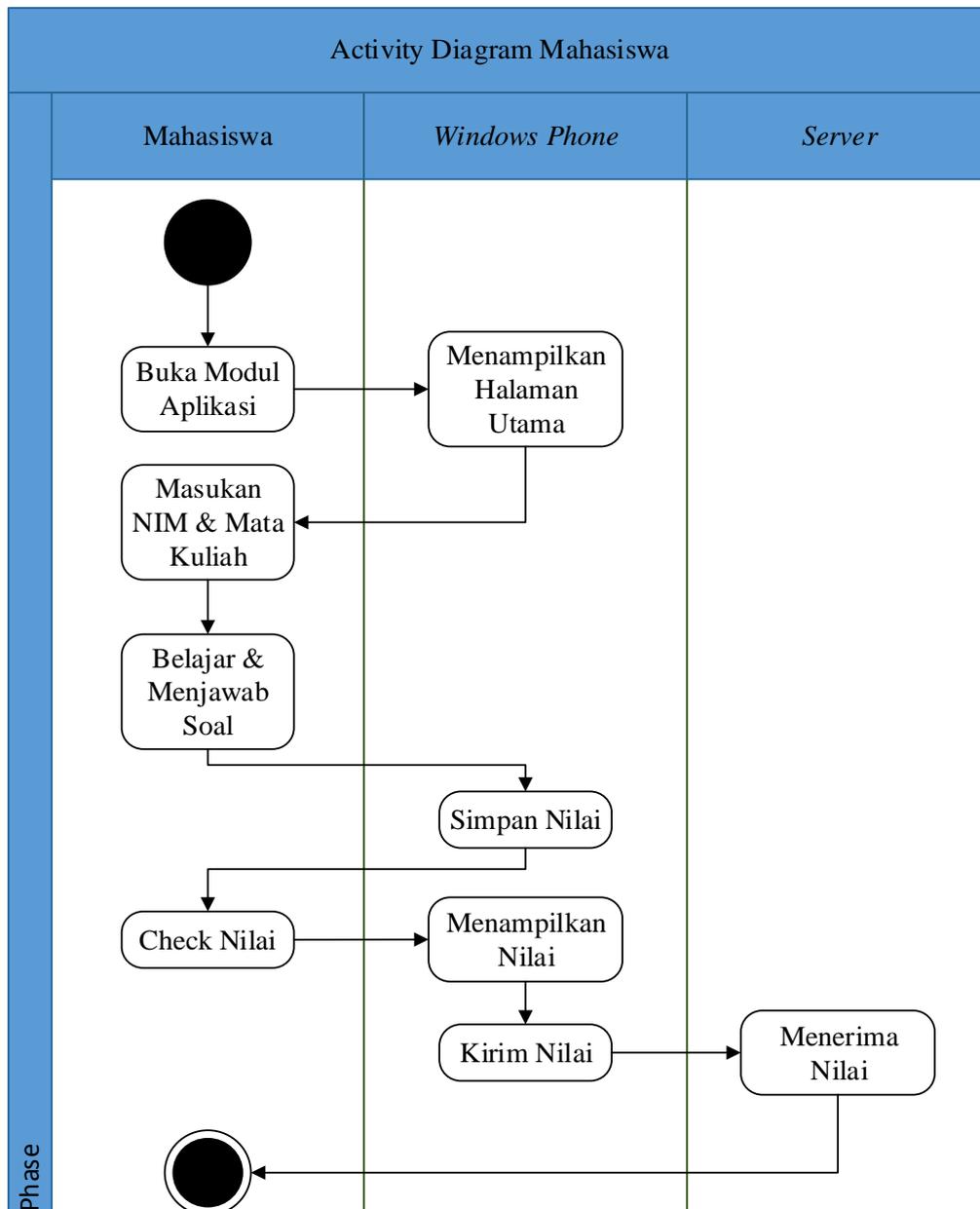
2.4.5 Activity Diagram

Activity diagram menggambarkan berbagai aktivitas dalam modul aplikasi yang sedang dirancang, berawal dari menggunakan modul aplikasi hingga berakhirnya menggunakan modul aplikasi. *Activity* diagram ini terdapat *swimline* yang berfungsi sebagai garis pembatas antara aktivitas satu dengan aktivitas lainnya, *activity* diagram mahasiswa dan *activity* diagram dosen dapat di lihat pada gambar 2.3 dan gambar 2.4.

1. Activity diagram Mahasiswa.

Penjelasan dari gambar 2.3 *Activity Diagram* Mahasiswa sebagai berikut:

- a. Mahasiswa membuka modul aplikasi yang telah dimiliki.
- b. Masuk halaman yang berisi judul modul aplikasi.
- c. Mahasiswa memasukan nim dan memilih Mata kuliah yang ingin di kerjakan, lalu mahasiswa membaca *summary* untuk menjawab soal yang telah tersedia.
- d. Setelah menjawab soal mahasiswa menyimpan hasil berupa *score* di internal memori *windows phone* lalu bisa *check* nilai yang telah tersimpan.
- e. Ketika nilai sudah sesuai dengan apa yang diinginkan mahasiswa dapat kirim nilai ke *server* yang nantinya di gunakan sebagai bahan penilaian oleh dosen.

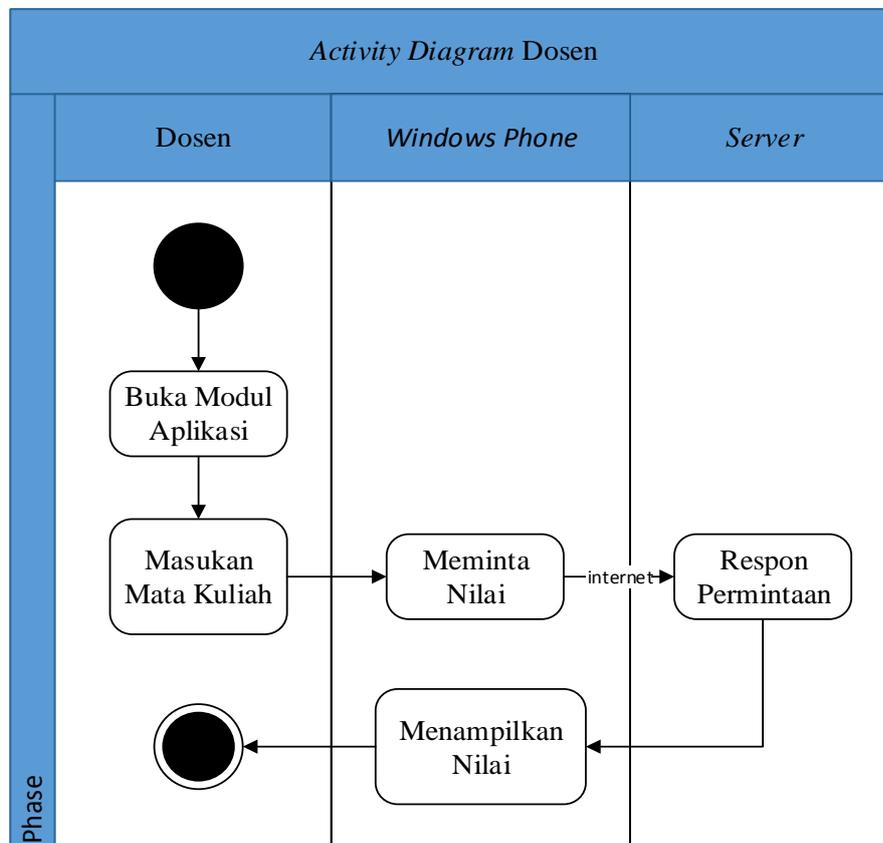


Gambar 2.4 Activity Diagram Mahasiswa.

1. Activity Diagram Dosen

Penjelasan dari Gambar 2.3 *Activity Diagram Dosen* sebagai berikut:

1. Dosen membuka modul aplikasi yang telah dimiliki.
2. Dosen memasukkan mata kuliah untuk melihat nilai dari mahasiswa yang mengerjakan.
3. Ketika terkoneksi internet *server* merespon permintaan.
4. *Server* menampilkan seluruh *score* modul yang telah di *upload* oleh mahasiswa.



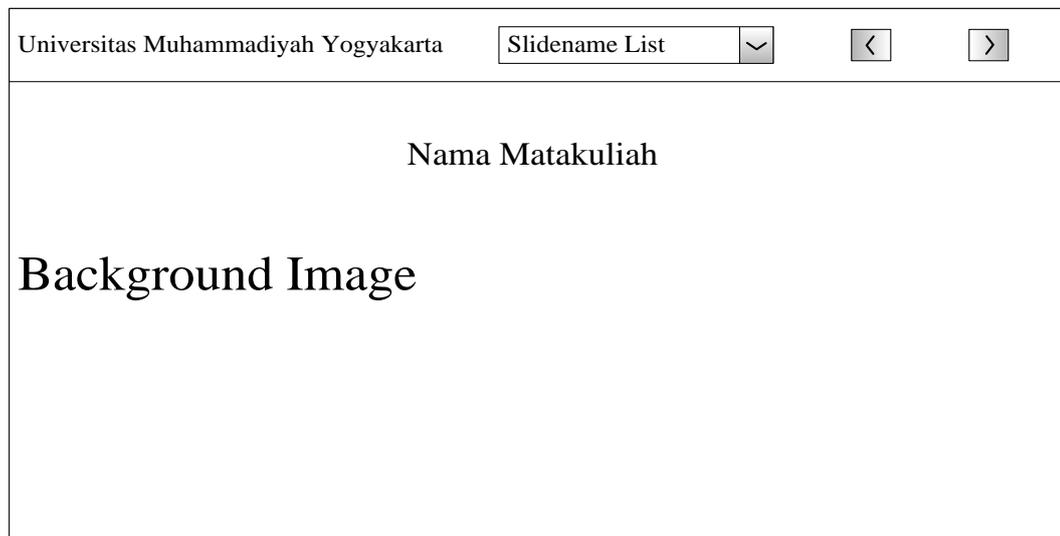
Gambar 2.5 *Activity Diagram Dosen*

2.5 Rancangan Antar Muka

Rancangan antarmuka 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.

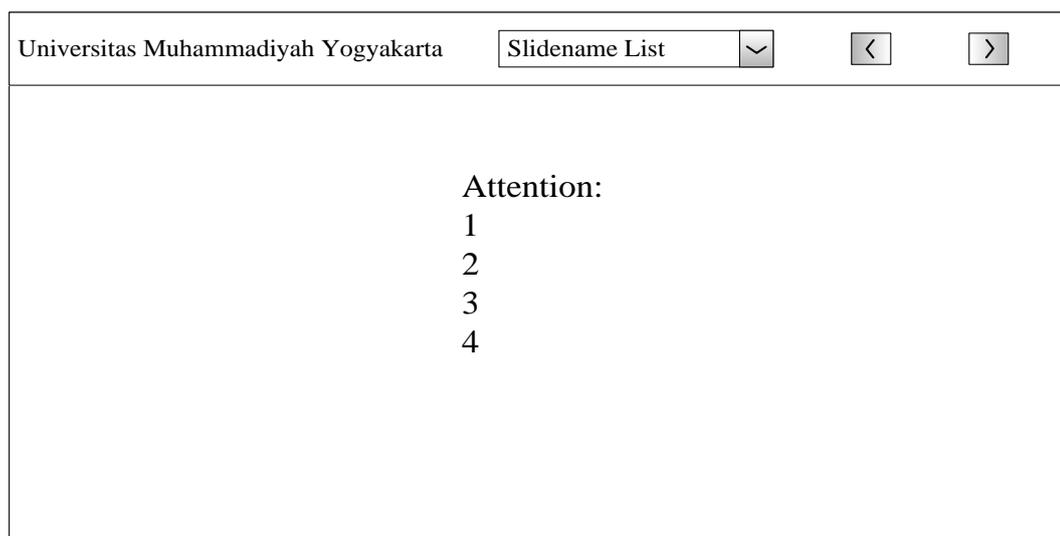
2.5.1 Rancangan Antar Muka Mahasiswa

Halaman awal



Gambar 2.6 Halaman awal saat mahasiswa membuka modul aplikasi.

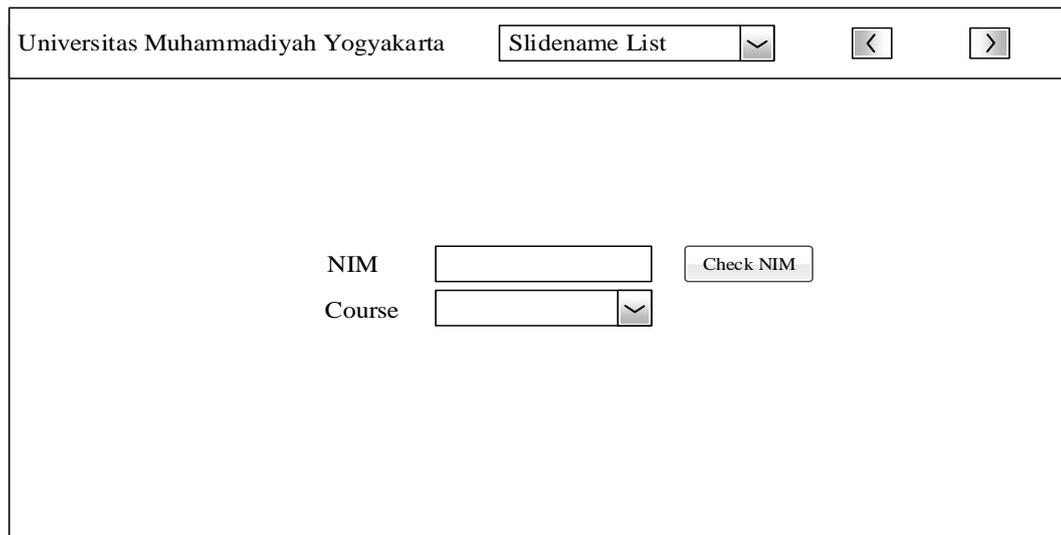
Halaman Attention



Gambar 2.7 Halaman *attention* yang berfungsi sebagai petunjuk mahasiswa dalam mengerjakan modul aplikasi.

Halaman *input* NIM dan *Course* (Mata kuliah)

Merupakan proses mahasiswa untuk input NIM dan *Course* (Mata Kuliah) yang dikerjakan, disitu terdapat button check nim yang nantinya berfungsi untuk check nim mahasiswa apakah sudah enroll atau belum enroll.



The screenshot shows a web browser window with the title 'Universitas Muhammadiyah Yogyakarta'. The address bar contains 'Slidename List' and navigation buttons. The main content area features a form with two input fields: 'NIM' (a text box) and 'Course' (a dropdown menu). To the right of these fields is a button labeled 'Check NIM'.

Gambar 2.8 Halaman *input* NIM dan *Course* (Mata kuliah)

Halaman ringkasan materi (*summary*)

Halaman dimana mahasiswa membaca ringkasan materi (*summary*) yang telah di buat oleh dosen untuk mengerjakan soal di halaman berikutnya.

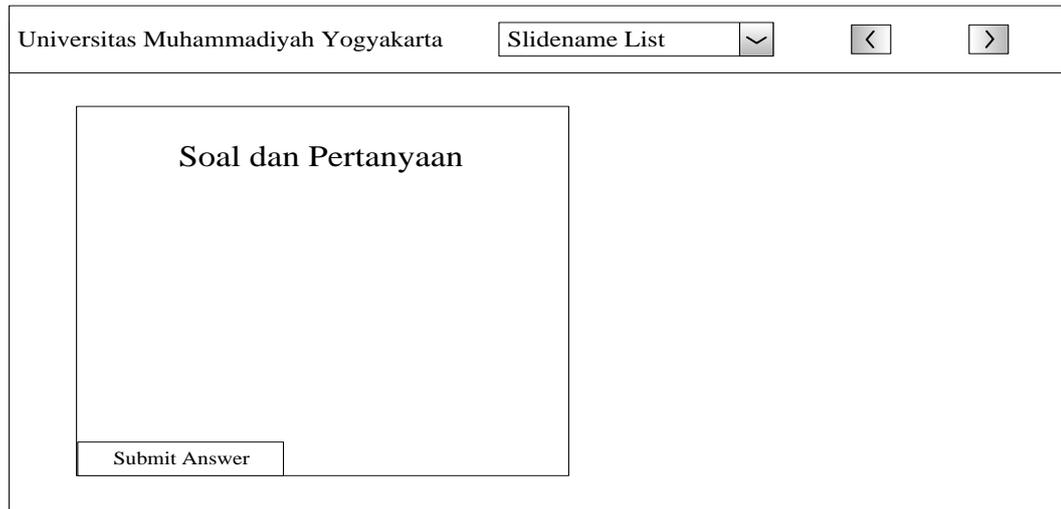


The screenshot shows a web browser window with the title 'Universitas Muhammadiyah Yogyakarta'. The address bar contains 'Slidename List' and navigation buttons. The main content area displays the text 'Ringkasan Materi/ Summary' in a large, bold font. A vertical scrollbar is visible on the right side of the content area.

Gambar 2.9 Halaman ringkasan materi (*summary*)

Halaman Soal dan Pertanyaan

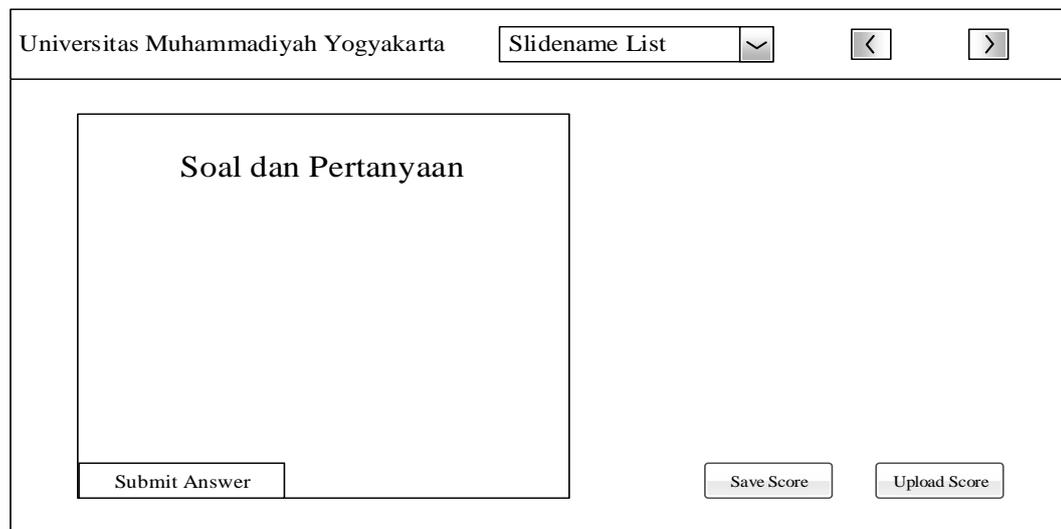
Merupakan soal yang diberikan oleh dosen untuk mahasiswa dan *button submit answer* itu untuk menyimpan *score* dari setiap bagian soal.



Gambar 2.10 Halaman Soal dan Pertanyaan.

Halaman *save score* dan *upload score*

Secara keseluruhan fungsi sama dengan halaman soal dan pertanyaan namun di gambar ini ada *button save score* dan *upload score*. *Button save score* berfungsi untuk *save file score* pada *internal memory windows phone* dan *button upload score* berfungsi untuk *upload score* ke *server* ketika mahasiswa sudah terkoneksi ke internet.



Gambar 2.11 Halaman *save score* dan *upload score*.

Halaman Check score dan upload score

Ini terdapat *button check score* dan *upload score*. *Check score* berfungsi untuk melihat nilai yang akan dikirim ke *server* dan *upload score* berfungsi untuk *upload score* ke *server* ketika terkoneksi internet.



The screenshot shows a web browser window with the title 'Universitas Muhammadiyah Yogyakarta'. The address bar contains 'Slidename List' and navigation buttons. The main content area features two buttons: 'Check Score' and 'Upload Score'.

Gambar 2.12 Check score dan upload score.

2.5.2 Rancangan Antar Muka Dosen

Halaman awal

Di situ terdapat text box yang berfungsi untuk dosen memasukan nama mata kuliah yang diampu untuk melihat nilai mahasiswa yang telah mengerjakan modul.



The screenshot shows a web browser window with the title 'Universitas Muhammadiyah Yogyakarta'. The address bar contains 'Slidename List'. The main content area features a text input field labeled 'Course Name' and a 'Search' button.

Gambar 2.13 Merupakan awal tampilan modul dosen.

Dosen Lihat Nilai Mahasiswa

Untuk melihat *score* mahasiswa yang telah *upload* ke *server*, dosen memasukkan mata kuliah dan *click search* nantinya muncul modul, nama, nim dan nilai mahasiswa yang telah *upload* nilai ke *server*.

Universitas Muhammadiyah Yogyakarta		Slidename List	▼
Course Name	<input type="text"/>	<input type="button" value="Search"/>	
Modul	Nama	NIM	Nilai

Gambar 2.14 Dosen Lihat Nilai Mahasiswa.

