LAPORAN SKRIPSI.pdf

by Eka Aprilia Irawan

Submission date: 10-Jul-2019 02:00PM (UTC+0700)
Submission ID: 1150695535
File name: LAPORAN_SKRIPSI.pdf (3.4M)
Word count: 20131
Character count: 123505

PERANCANGAN APLIKASI UJIAN *ONLINE* BERBASIS *ANDROID* DI JURUSAN TEKNIK ELEKTRO UNIVERSITAS SRIWIJAYA

SKRIPSI

Dibuat Untuk Memenuhi Syarat Mendapatkan Gelar Sarjana Teknik
Pada Jurusan Teknik Elektro Fakultas Teknik
Universitas Sriwijaya

Oleh:
EKA APRILIA IRAWAN
03041381720026

JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS SRIWIJAYA
2019

BAB I
PENDAHULUAN

1.1 Latar Belakang

Sistem informasi berperan penting bagi suatu organisasi atau suatu instansi termasuk Universitas Sriwijaya. Dimana sistem informasi dilakukan dengan baik, terorganisasi dan benar. Universitas Sriwijaya adalah salah satu perguruan tinggi yang memiliki beberapa program studi, salah satunya adalah program studi Teknik Elektro. Pemanfaat sistem informasi digunakan untuk pelaksanaa akademik yaitu pengelolaan dan manajemen hasil studi mahasiswa yang dapat diakases melalui *website akademik.unsri.ac.id.* Hasil studi yang didapat tidak terlepas dari proses evaluasi untuk mengukur atau menilai hasil belajar mengajar yaitu dengan ujian. Ujian dapat mengukur taraf pencapaian belajar dan pemahaman dari ilmu yang didapat oleh peserta didik.

Sebagian besar ujian di Jurusan Teknik Elektro Universitas Sriwijaya masih dilaksanakan secara konvesional atau manual, dimana pelaksanaan ujian masih menggunakan kertas dan alat tulis dan hasil ujian diperiksa satu persatu secara manual. Tentu memiliki kekurangan dalam hal efektifitas, waktu, dan kinerja. Ujian memiliki bentuk pilihan ganda, uraian singkat dan esai. Namun ujian dengan sistem esai dapat melatih dalam menyampaikan suatu informasi dan opini yang ada, ujian *online* juga menuntut untuk berfikir secara kritis tentang pemahaman suatu ilmu secara mendalam.

Berdasarkan latar belakang yang telah dijelaskan, ada sistem ujian yang lebih efektif untuk menggantikan sistem ujian yang ada saat ini. Yang mana sistem ini akan memudahkan dalam pelaksanaan ujian. Maka dalam tugas akhir ini penulis akan merancang sebuah sistem ujian online menggunkan *mobile* aplikasi berbasis *android* di Jurusan Teknik Elektro Universitas Sriwijaya. Dimana pelaksanaaan ujian dapat dilakukan secara *online*. Diharapkan sistem ujian *online* ini mampu memberikan metode ujian yang efektif dan efesien bagi pihak yang berkepentingan dalam hal ini dosen dan mahasiswa. Hal tersebutlah yang melatarbelakangi penulis untuk membuat tugas akhir dengan judul ***"PERANCANGAN APLIKASI UJIAN***

ONLINE BERBASIS ANDROID DI JURUSAN TEKNIK ELEKTRO UNIVERSITAS SRIWIJAYA"

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dibahas maka rumusan masalah dari tugas akhir ini adalah :

Bagaiman merancang dan membangun sistem Ujian *Online* Berbasis *Android* di Jurusan Teknik Elektro Universitas Sriwijaya dengan kriteria sebagai berikut :

1. Sistem sebagai solusi pelaksanaan ujian secara *online* di Jurusan Teknik Elektro Universitas Sriwijaya
2. Sistem dapat bermanfaat secara efektif serta mudah digunakan oleh mahasiswa dan dosen.
3. Aplikasi dapat diakses kapan saja dan dimana saja
4. Menu dan fitur pada aplikasi sesuai dengan kebutuhan

1.3 Batasan Masalah

Adapun batasan masalah dalam penulisan tugas akhir ini adalah sebagai berikut :

1. Sasaran pada sistem adalah mahasiswa dan dosen di lingkungan Jurusan Teknik Elektro Universitas Sriwijaya
2. Metode perancangan sistem informasi yang digunakan mengacu pada metode RAD *(Rappid Application Development).*
3. Pada tahapan perancangan dari sistem informasi ini menggunakan tools:
   a. Metode Kipling 5W+1H untuk mendapatkan gambaran secara detail mengenai sistem yang akan dirancang.
   b. Metode analisis yang digunakan yaitu metode SOAR (*Strenght, Opportunities, Aspiration, and Result*). Metode SOAR digunakan untuk perancangan strategis suatu sistem yang berfokus pada kekuatan pada sistem yang akan dibuat, keuntungan apa yang akan didapat dari sistem yang dibuat, aspirasi dari berbagai pihak, dan tujuan yang akan dicapai.

4. Sistem menggunakan tools diagram UML Versi 2.0 meliputi *usecase diagram*, *activity diagram*, *class diagram*, *sequence diagram*, dan *deployment diagram*
5. Pada tahapan melakukan pengkodingan perancangan aplikasi ini menggunakan :
   a. *Framework* yang digunakan yaitu *CodeIgniter*, *Ionic AngularJS, Cordova,* dan *Bootstrap*
   b. Media *coding* menggunakan *Netbeans* IDE dan *Visual Studio Code*.
6. Metode *testing* yang digunakan yaitu metode *Black Box.*

1.4 Tujuan Penelitian

Tujuan penulisan dari tugas akhir ini adalah :

1. Terbangunnya suatu sistem ujian *online* yang dapat membantu mahasiswa di Jurusan Teknik Elektro Universitas Sriwijaya dan dioperasikan pada *mobile android* serta dapat diakses dimana saja dan kapan saja.
2. Terbangunnya suatu sistem yang dapat memanajemen ujian dan memudahkan dosen dalam pelaksanaan ujian di Jurusan Teknik Elektro Universitas Sriwijaya.
3. Terbangunnya sistem ujian yang lebih efektif dibanding pelaksanaan ujian secara konvensional di Jurusan Teknik Elektro Universitas Sriwijaya.

1.5 Manfaat Penelitian

Manfaat yang diharapkan dari penyusunan Tugas Akhir ini adalah sebagai berikut :

Manfaat yang diharapkan dari penyusunan Tugas Akhir ini adalah sebagai berikut:

a. Bagi Penulis

- Sebagai sarana untuk menerapkan pengetahuan yang diperoleh selama menempuh studi.

- Sebagai media pembelajaran dalam menyelesaikan masalah yang sesuai dengan kondisi lapangan atau masyarakat dengan merancang suatu sistem informasi yang tepat guna.

b. Bagi Akademik

Laporan Tugas Akhir ini dapat dijadikan sebagai sarana tambahan referensi di perpustakaan Universitas Sriwijaya mengenai permasalahan yang terkait dengan penulisan Tugas Akhir ini.

c. Bagi pengguna

Hasil Penulisan Tugas Akhir ini diharapkan dapat mempermudah proses belajar mengajar bagi dosen dan mahasiswa terutama pada pelaksanaan ujian.

1.6 Sistematika Penelitian

Sistematika penulisan yang digunakan dalam penyusunan tugas akhir ini terdiri atas lima bab yaitu :

BAB I : PENDAHULUAN

Pada bab ini diuraikan mengenai latar belakang, rumusan masalah, batasan masalah, tujuan dan sistematika penulisan dalam tugas akhir ini.

BAB II : TINJAUAN PUSTAKA

Pada bab ini diuraikan teori-teori yang dibutuhkan dan mendukung penulisan tugas akhir ini.

BAB III : METODOLOGI

Dalam bab ini menjelaskan metode, pengumpulan data pengembangan system dan metode penulisan.

BAB IV : PERANCANGAN

Pada bab ini berisikan uraian dan penjelasan proses perancangan serta menguji aplikasi hasil dari perancangan tersebut.

BAB V : KESIMPULAN DAN SARAN

Pada bab ini berisi kesimpulan dari tugas akhir ini, dan saran sebagai acuan untuk pengembangan dan perbaikan sistem kedepannya.

BAB II
TINJAUAN PUSTAKA

2.1 Sistem

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu, sistem juga dinyatakan abstrak atau fisis[1]. Sistem yang abstrak yaitu sistem tersusun secara teratur dari gagasan-gagasan yang saling berhubungan atau saling bergantung. Sedangkan sistem yang bersifat fisis adalah bagaimana cara suatu sistem bekerjasama secara bersama-sama untuk mrncapai suatu tujuan.

2.2 Informasi

Berikut ini beberapa pendapat mengenai pengertian informasi:

a. Informasi adalah sekumpulan fakta (data) yang diorganisasikan dengan cara tertentu sehingga mereka mempunyai arti bagi si penerima[2].
b. Informasi adalah data yang diolah menjadi bentuk yang lebih berguna atau lebih berarti bagi yang menerimanya[3].
c. Informasi adalah data yang sudah mengalami pemrosesan sedemikian rupa sehingga dapat digunakan oleh pengunanya dalam membuat keputusan[4].

Berdasarkan beberapa pendapat yang dikemukakan di atas, maka dapat disimpulkan bahwa Informasi adalah suatu data yang diorganisaskan dan juga diolah yang mengalami pemrosesan menjadi bentuk data yang lebih berguna untuk penerimanya dan digunakan untuk membuat suatu keputusan

2.3 Sistem Informasi

Sistem Informasi adalah suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan data transaksi harian, mendukung operasi yang bersifat manajerial dan kegiatan strategi dari suatu organisasi yang menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan[5] .

Sistem informasi adalah sistem yang dibuat oleh manusia yang digunakan sebagai penyedia informasi, menyimpan, mengelola data dalam suatu himpunan yang berbasis komputer komputer dan manual untuk ditujukan kepada pemakai. Dalam artian luas sistem informasi adalah cara berinteraksi antara pengguna, proses algoritmik, data, dan informasi dengan teknologi yang dapat mempermudah dalam kegiatan bisnis.

2.4 Sistem Informasi Akademik

Sistem Informasi Akademik adalah perangkat lunak yang digunakan untuk menyajikan informasi dan menata administrasi yang berhubungan dengan kegiatan akademis. Dengan menggunakan perangkat lunak seperti ini diharapkan kegiatan administrasi akademis dapat dikelola dengan baik dan informasi yang diperlukan dapat diperoleh dengan mudah dan cepat[6]. Akademik adalah suatu bidang pembelajaran yang dikelola oleh suatu lembaga pendidikan untuk peningkatan pengetahuan dalam bidang pendidikan yang mempelajari kurikulum. Sedangkan Sistem informasi merupakan aplikasi komputer untuk mendukung operasi dari suatu organisasi: operasi, instalasi, perangkat lunak, data dan perawatan komputer[7]. Komponen sistem informasi terdiri dari:

a. *Hardware*, terdiri dari perangkat keras komputer dan perangkat pendukung lainnya.
b. *Software*, merupakan kumpulan dari perintah atau fungsi yang ditulis dengan aturan tertentu untuk memerintahkan komputer melaksanakan tugas tertentu.
c. Data, merupakan komponen dasar yang menyediakan dan mempunyai nilai informasi.
d. Sumber Daya Manusia, merupakan komponen lingkungan yang berusaha memecahkan masalah dan mengolah sistem informasi.

Jadi dapat disimpulkan bahwa sistem informasi akademik adalah sistem yang menangani hal dalam bidang akademik guna untuk mengetahui bagaimana pencapaian prestasi seorang siswa atau mahasiswa dalam proses pendidikan. Sistem akademik mulai dari pendaftaran, nilai, dan juga proses belajar-mengajar.

2.5 Akademik

Akademik adalah suatu bidang pembelajaran yang dikelola oleh suatu lembaga pendidikan untuk peningkatan pengetahuan dalam bidang pendidikan yang mempelajari kurikulum[6].

Sedangkan mengajar adalah perbuatan yang dilakukan seseorang dengan tujuan untuk membantu atau memudahkan orang lain melakukan kegiatan belajar. Berdasarkan dari pendapat ahli tersebut, dapat penulis tarik benang merah bahwa proses kegitatan akademik baik itu belajar dan mengajar adalah memiliki suatu tujuan yang dapat membawa perubahan dalam pengetahuan, pemahaman, dan keterampilan pembelajar tersebut[8].

2.6 Konsep Dasar Ujian

2.6.1 Pengertian Ujian

Ujian merupakan suatu cara dan juga alat ukur evaluasi untuk mengukur taraf pencapaian belajar dan pemahaman dari ilmu yang didapat oleh peserta didik. Ujian juga dijadikan sebagai evaluasi nilai dalam proses belajar-mengajar seberapa jauh ilmu yang dikuasai dan juga dapat mendorong wawasan dan pemikiran seseorang dalam kegiatan pembelajaran.

Dengan ujian kita dapat mengetahui seberapa jauh ilmu yang kita kuasai dan juga kita dapat selama ini. Dalam ujian kita dituntut untuk berfikir secara kritis, benar dan tepat.

2.6.2 Jenis-Jenis Ujian

Berikut adalah jenis-jenis ujian :

a. Ujian Konvensional

   Ujian konvensional atau ujian manual adalah jenis ujian yang sudah ada dan diterapkan puluhan tahun yang lalu, ujian manual ini masih menggunakan alat tulis berupa kertas, pena, pensil dan alat tulis umum lainnya untuk pelaksanaan ujian dan dilakukan dengan tulis tangan baik untuk distribusi soal maupun jawaban, pengumpulan hasil ujian juga masih manual yaitu memberikan langsung kepada pengawan ujian ditempat[9]

b. Ujian Modern

Ujian modern adalah ujian yang hampir sama dengan ujian konvensional atau ujian manuak. Namun perbedaannya ujian modern sudah menggunakan alat ketik untuk penulisan soal dan untuk memperbanyak soal menggunakan mesin *fotocopy*. Pemeriksaan ujian juga sudah dipermudah dengan adanya alat yang dapat memeriksa hasil ujian secara komputerisasi. Bentuk ujian ini biasanya bersifat objektif atau pilihan ganda[9].

c. Ujian *Online*

Ujian *online* merupakan jenis ujian dimana penditribusian soal sampai pengkoreksian hasil ujian semuanya dilakukan secara *online* menggunakan perangkat elektronik yang terkoneksi internet. Ujian *online* dapat membantu pengajar dalam proses pengkoresian sehingga tidak perlu pengumpulan lembar jawaban satu persatu. Dengan menggunaka ujian *online* (*online test*) ini dapat mengukur kemampuan seseorang dalam menggunakan komputer. Penilaian pada ujian *online* ini dapat digunakan untuk meminimalisir terjadinya kecurangan dan *human error*[9].

2.7 Metode Kipling 5W+1H

Pada tugas akhir ini dalam penulisan tahapan planning menggunakan metode analisis 5W+1H, untuk mendapatkan gambaran secara detail mengenai sistem informasi yang dirancang. Berikut penjelasan tetang 5W+1H sebagai metode analisis pada sistem yaitu sebagai berikut[10]:

a. *What* (Apa), Bentuk pertanyaan yang bertujuan untuk mencari tahu mengenai sesuatu perangkat lunak dan sistem informasi yang akan dirancang atau dikembangkan.
b. *Who* (Siapa), Bentuk pertanyaan yang bertujuan mencari tahu setiap subjek atau aktor yang terlibat dalam penggunaan dan perancangan perangkat lunak dan sistem informasi.
c. *Where* (Dimana), Bentuk pertanyaan yang bertujuan untuk menjelaskan tempat atau lokasi yang berhubungan dengan pemakaian perangkat lunak tersebut.

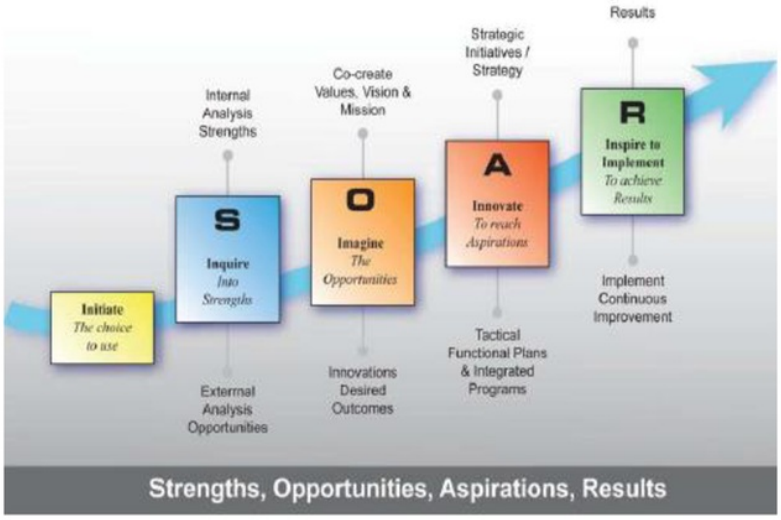
d. *Where* (Kapan), Bentuk pertanyaan yang bertujuan mencari tahu tempat kejadian suatu peristiwa.

e. *Why* (Mengapa), Bentuk pertanyan yang bertujuan menjelaskan alasan dan sebab akibat dari adanya perancangan perangkat lunak dan sistem informasi.

f. *How* (Bagaimana), Bentuk pertanyaan dengan tujuan mencari tahu proses peristiwa yang terjadi dalam perancangan perangkat lunak dan sistem informasi.

2.8 Metode SOAR

SOAR adalah kerangka perencanaan strategis dengan pendekatan yang berfokus pada kekuatan dan mencari untuk mengerti keseluruhan sistem dengan memasukkan suara dari stakeholders yang relevan. Pendekatan ini memungkinkan organisasi untuk membangun masa depan melalui kolaborasi, pemahaman bersama, dan komitmen untuk bertindak[11].



Gambar 2.1 SOAR *framework*
(Sumber: Stavros , Hinrichs. 2009. *The Thin Book of Appreciative Inquiry 3rd Edition.* Thin Book Series. 1-4)

Berikut ini merupakan unsur-unsur dari analisis SOAR :

a. *Strength* (S)

   Strength (S) merupakan segala hal yang menjadi kekuatan pada kemampuan terbesar yang dimiliki. Bentuk dari kekuatan tersebut dapat berupa aset yang berwujud maupun yang tidak berwujud. Kekuatan dapat terus dikembangan agar terjadinya kemajuan.

b. *Opportunities* (O)

   Opportunities (O) merupakan segala bentuk peluang yang ada pada lingkungan eksternal dari suatu usaha. Peluang tersebut harus dapat dianalisis dengan baik, kemampuan dalam memaksimalkan peluang akan mempengaruhi keberhasilan usaha.

c. *Aspiration* (A)

   Keberhasilan juga memerlukan aspirasi yang dapat diperoleh dari berbagai pihak yang terlibat pada suatu usaha tersebut. Untuk memperoleh aspirasi dapat dilakukan dengan cara diskusi dan bertukar pendapat.

d. *Results* (R)

   Results (R) merupakan hasil dan tujuan yang akan dicapai dalam suatu usaha, mengidentifikasikan results di awal suatu proses dapat menjadikan acuan dan motivasi dalam mencapai tujuan yang telah .

2.9 UML Versi 2.0

Unified Modelling Language (*UML*) adalah bahasa standar untuk penulisan cetak biru (*blue print*) perangkat lunak. *UML* dapat digunakan untuk memvisualisasikan, menentukan, mengkonstruksi, dan mendokumentasikan artifak-artifak suatu sistem software. UML 2.0 memberikan 13 diagram yang berbeda-beda untuk digunakan pada pemodelan perangkat lunak[12].

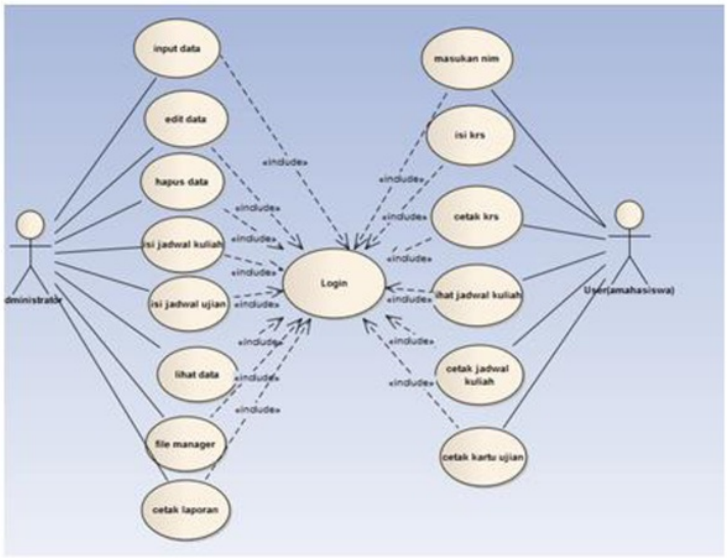
Pada tugas akhir ini penulis menggunakan UML Vversi 2.0 dan menggunakan 4 diagram. Keempat diagram tersebut adalah *usecase diagram*, *activity diagram*, *class diagram*, dan *sequence diagram*. Dengan menggunakan 4 diagram tersebut sudah cukup untuk menggambarkan keseluruhan dari perangkat

lunak dan sistem informasi yang akan dirancang. Berikut penjelasan dar 4 diagram tersebut :

a. *Usecase Diagrams*

Usecase menggambarkan bagaimana pengguna berinteraksi dengan sistem dengan cara menentukan langkah-langkah yang diperlukan untuk mencapai suatu tujuan tertentu. Tujuan utama dari *use case* terhadap proses pengembangan perangkat lunak adalah deskripsi tekstual dari masing-masing *usecase.*

Pada diagram *usecase*, *use case-use case* tersebut ditampilkan dengan bentuk oval. Aktor-aktor dihubungkan dengan garis-garis ke *usecase*. *Use case* ditempatkan pada kotak, namun aktor-aktor berada diluar kotak tersebut. Kotak tersebut merupakan pengingat visual tentang batasan-batasan antara sistem dan aktor[12].

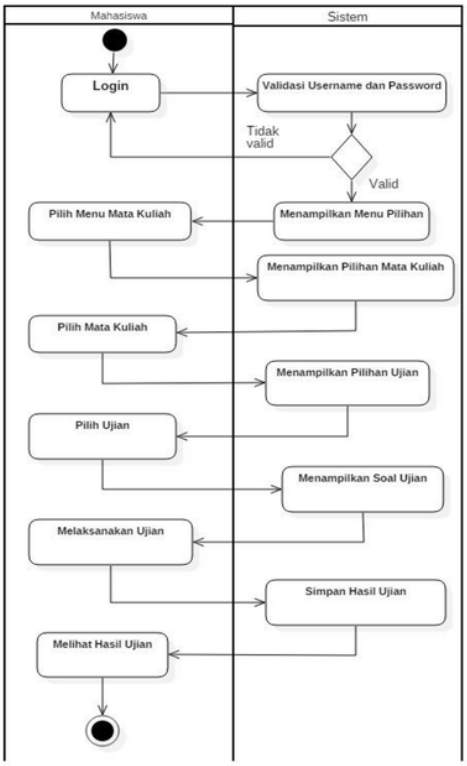


Gambar 2.2 Contoh *usecase diagram*
(Sumber : http://nurhayatiizz.blogspot.com/2012/08/use-case-sistem-informasi-pelaksanaan_13.html)

b. *Activity Diagrams*

Acrivity diagrams menggambarkan perilaku dinamis dari suatu sistem atau suatu komponen sistem melalui aliran kontrol atau alir kerja di antara aksi-aksi yang dilakukan sistem yang berisi aktivitas[12]. Komponen utama suatu diagram aktivitas

adalah *node* aksi, direpresentasikan dengan suatu persegi panjang dengan sudut bulat, yang berhubungan dengan suatu tugas yang dilakukan oleh sistem perangkat lunak. Tanda panah ke *node* lain mengartikan bahwa ada aliran kerja aktivitas. Titik hitam pada awal adalah *initial node* sedangan titik hitam dengan lingaran luar menandakan *final node*.

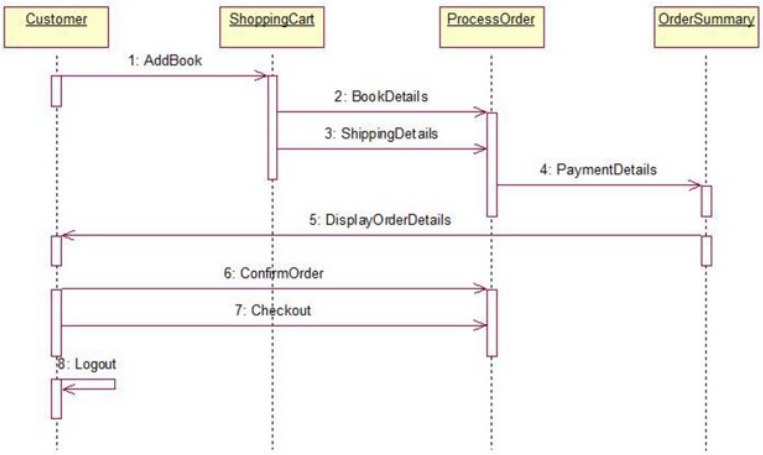


Gambar 2.3 Contoh *activity diagram*
(Sumber: http://rasalibre.co/draw-a-uml-sequence-diagram-for-online-bookshop.html)

c. *Squence Diagrams*

Squence diagram menggambarkan urutan sementara dimana pesan-pesan dikirim diantara objek-objek untuk menyelesaikan pekerjaan tersebut. Diagram urutan juga digunakan untuk menunjukan interaksi-interaksi pada suatu *use case* atau pada suatu skenario dari suatu sistem perangkat lunak[12]. Komponen utama yang ada pada squence diagram adalah *object*, *activation boxes*, *actors* dan *lifeline*.

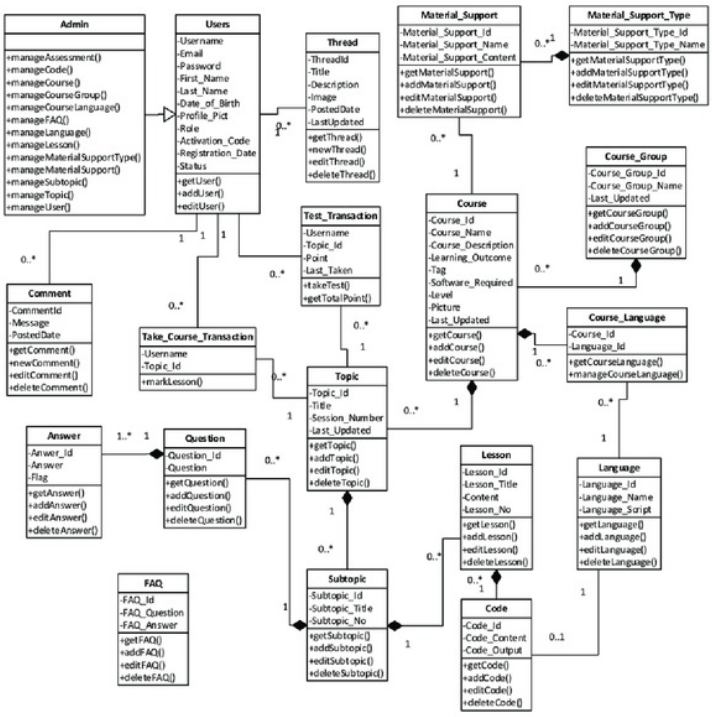
Object adalah komponen yang berbentuk kotak yang mendeskripsikan bagaimana sebuah *object* bertindak kepada sistem. *Activation boxes* adalah komponen persegi panjang yang menggambarkan waktu yang diperlukan *object* menyelesaikan tugas. *Actors* adalah yang mewakili seorang pengguna. Dan *lifeline* merupakan komponen berbentuk garis putus-putus yang menggambarkan aktifitas *object.*



Gambar 2.4 Contoh *squence diagram*
(Sumber: http://rasalibre.co/draw-a-uml-sequence-diagram-for-online-bookshop.html)

d. *Class Diagrams*

Class diagrams atau dapat disebut juga diagram kelas digunakan untuk memodelkan kelas-kelas, termaksud atribut-atributnya, operasi-operasi, dan hubungan serta asosiasinya dengan kelas-kelas lain[11]. Terdapat 4 elemen pada class diagram yaitu *classes name, attributee, operations,* dan *relationships.*



Gambar 2.5 Contoh *class diagram*
(Sumber: https://www.researchgate.net/figure/Class-Diagram-of-Online-Learning-System_fig4_320211937)

2.10 *Mobile*

Kata *mobile* menunjuk kepada perangkat dengan teknologi tinggi yang dapat bergerak digunakan sebagai perangkat komunikasi tanpa menggunakan kabel. Aplikasi yang akan dirancang ini berupa perangkat lunak yang ada pada perangkat mobile[13]. Dengan berkembanganya ajringan internet dengan jangkauan yang luas kini memberikan kemudahan mencari dan mendapatkan informasi juga berkomunikasi dengan orang lain yang berada jauh semakin mudah dan cepat. Untuk mengakses informasi serta berkomunikasi tersebut kita menggunakan sebuah perangkat *mobile* seperti handphone atau *smartphone*. Maka penggunaan *mobile* aplikasi dan *mobile web* tidak bisa kita hindari lagi.

a. *Mobile Web*

Mobile web adalah halaman HTML yang dapat diakses dengan menggunakan perangkat mobile (smartphone atau tablet) melalui jaringan internet 3G, 4G atau WiFi. *Mobile* web ini akan menampilkan konten-konten berupa text, gamabr, video maupun audio.

Mobile web memiliki prinsip "*develop once run everywhere*", yang berarti dapat beroperasi pada lintas *platform* dalam sekali pengembangan. Jadi, *mobile web* relatif tidak membutuhkan banyak biaya untuk pengembangan dibandingkan *mobile apps* karena dapat berjalan dengan baik di semua *browser* dan platform *mobile*. *Mobile web* dapat didistribusikan tanpa harus mendapatkan *approval* dari pihak tertentu dan pengguna tidak perlu harus mencari di toko aplikasi. *Mobile web* hanya tersedia hanya jika pengguna dalam keadaan *online*, performa relatif lambat, dan tidak semua perangkat memiliki fitur tertentu yang bisa dimiliki dalam *mobile web*[13].

b. *Mobile Aplikasi*

Mobile aplikasi adalah aplikasi yang dirancang untuk digunakan pada perangkat smartphone yang dapat dibeli atau di*download* melalui toko aplikasi seperti *Play Store, App Store,* maupun *Blackberry App* kemudian di*install* untuk dapat dioperasikan.

Mobile apps memiliki performa lebih cepat, hanya memiliki 1 domain, dan lebih menarik dari segi visual. Pengguna memiliki akses penuh ke perangkat mobile, seperti kamera, gesture, pemberitahuan, *speaker*, dan fitur lainnya. Kualitas dan keamanan *mobile apps* terjamin karena dikontrol oleh vendor masing-masing. *Mobile apps* tidak selalu dapat dioperasikan ketika pengguna *online*, tetapi dalam keadaan *offline* juga bisa. Hanya saja, ada beberapa jenis aplikasi yang tidak digunakan di semua *platform* IOS lain. Biaya pengembangan dan *maintenance* aplikasi lebih mahal. *Mobile apps* juga mengharuskan adanya proses persetujuan dan pembatasan konten yang berlaku di toko aplikasi[13].

2.11 *Website*

Website atau disingkat *web*, dapat diartikan sekumpulan halaman yang terdiri dari beberapa laman yang berisi informasi dalam bentuk data digital baik berupa text, gambar, video, audio, dan animasi lainnyayang disediakan melalui jalur internet. Lebih jelasnya, *website* merupakan halaman-halaman yang berisi informasi yang ditampilkan oleh *browser* seperti Mozila Firefox, Google Chrome atau yang lainnya[14].

Untuk menyediakan sebuah website, maka harus tersedia unsur-unsur penunjangnya, adalah sebagai berikut:

a. Nama Domain (*domain name*) : Nama domain adalah alamat unik di dinia internet yang digunakan untuk mengidentifikasi sebuah *website,* atau dengan kata lain alamat yang digunakan untuk menemukan sebuah *website* pada dunia internet. Nama domain diperjualbelikan secara bebas di internet dengan status sewa tahunan.

b. Rumah tempat website (*web hosting*) : *Web Hosting* dapat diartikan sebagai ruangan yang terdapat menyimpan berbagai data, *file-file*, gambar, video, data *email,* statistik, *database*, dan lain sebagainya yang akan ditampilkan di *website*. *Web Hosting* juga diperoleh dengan menyewa. Pengguna akan memperoleh kontrol panel yang terproteksi dengan *username* dan *password* untuk administrasi *website*nya.

c. Bahasa Program (*Scripts Program*) : Bahasa yang digunakan untuk menerjemahkan setiap perintah dalam website yang pada saat diakses. Jenis bahasa program sangat menentukan statis, dinamis, atau interaktifnya sebuah *website*. Semakin banyak bahasa program yang digunakan maka akan terlihat website semakin dinamis dan interaktif. Jenis-jenis bahasa program yang banyak dipakai antara lain: HTML, ASP, PHP, JSP, Java Scripts, Java Applets, XML, Ajax dan sebagainya.

d. Desain *Website* : Setelah melakukan penyewaan *domain name* dan *web hosting* serta penguasaan bahasa program, unsur *website* yang penting dan utama adalah desain. Desain *web* menentukan kualitas dan keindahan sebuah *website.* Desain

akan berpengaruh kepada penilaian pengunjung akan bagus tidaknya sebuah *website*.

e. Program transfer data ke pusat data : FTP (*File Transfer Protocol*) merupakan akses yang diberikan pada saat kita memesan *web hosting*, FTP berguna untuk memindahkan *file-file website* yang ada pada komputer kita ke pusat *web hosting* agar dapat terakses ke seluruh dunia.

2.12 RAD (*Rapid Application Development*)

Merupakan metode proses pengembangan perangkat lunak secara linear sequential yang menekankan pada siklus pengembangan yang sangat singkat. Jika kebutuhan dipahami dengan baik, proses RAD memungkinkan tim pengembangan menciptakan "sistem fungsional yang utuh" dalam periode waktu yang sangat pendek (kira-kira 60-90 hari). Pendekatan metode RAD menekankan cakupan[15]:

a. Pemodelan bisnis (*Bussiness Modelling*)

   Aliran informasi diantara fungsi-fungsi bisnis dimodelkan dengan suatu cara untuk menjawab pertanyaan-pertanyaan berikut : Informasi apa yang mengendalikan proses bisnis ? Kemana informasi itu pergi? Siapa yang memprosesnya ?

b. Pemodelan data (*Data Modelling*)

   Aliran informasi yang didefinisikan sebagai bagian dari fase pemodelan bisnis disaring ke dalam serangkaian objek data yang dibutuhkan untuk menopang bisnis tersebut. Karakteristik/atribut dari masing-masing objek diidentifikasi dan hubungan antara objek-objek tersebut didefinisikan.

c. Pemodelan proses (*Process Modelling*)

   Aliran informasi yang didefinisikan dalam fase pemodelan data ditransformasikan untuk mencapai aliran informasi yang perlu bagi implementasi sebuah fungsi bisnis. Gambaran pemrosesan diciptakan untuk menambah, memodifikasi, menghapus atau mendapatkan kembali sebuah objek data.

d. Pembuatan aplikasi (*Application generation*)

Selain menciptakan perangkat lunak dengan menggunakan bahasa pemrograman generasi ketiga yang konvensional, RAD lebih banyak memproses kerja untuk memakai lagi komponen program yang telah ada atau menciftakan komponen yang bias dipakai lagi. Pada semua kasus, alat-alat Bantu otomatis dipakai untuk memfasilitasi kontruksi perangkat lunak.

e. Pengujian dan pergantian (*Testing and turnover*)

Karena proses RAD menekankan pada pemakaian kembali, banyak komponen yang telah diuji. Hal ini mengurangi keseluruhan waktu pengujian. Tapi komponen baru harus diuji.

2.13 PHP

PHP singkatan dari PHP *Hypertext Preprocessing*. Merupakan bahasa *scripting* untuk *web* yang cukup populer. Dengan PHP, kita bisa membuat *web* dinamis dimana kode PHP diselipkan di antara *script* kode-kode HTML yang merupakan bahasa markup standar untuk dunia web. PHP adalah bahasa *script,* artinya ditanamkan atau disisipkan ke dalam HTML. PHP adalah bahasa pemprograman yang populer dikalagan web programmer untuk membuat sebuah website yang dinamis dan interaktif.Untuk membedakan kode PHP dan kode HTML sebagai wadahnya, digunakan tag-tag PHP [16].

2.14 MySQL

MySQL adalah *software database* yang paling banyak dipakai untuk membuat aplikasi dinamis. *Software* paketan AMP (Apache, MySQL, dan PHP) biasanya sudah mengandung MySQL. Indikasinya adalah dengan terlihatnya halaman PHPMyAdmin. MySQL adalah sebagai wadah untuk tempat penyimpana text based data. Data berupa file, gambar, audio dan lain-lain diletakkan pada server dalam folder tanpa harus disimpan pada database, disimpan dengan referensi nama.MySQL ini adalah konstruksi database yang memungkinkan Apache dan PHP bekerja sama untuk mengakses dan menampilkan data dalam format yang terbaca di *browser*. MySQL adalah *server database* yang dapat diakses dengan

Sructured Query Language. MySQL didesain untuk bisa menangani beban pemrosesan yang berat dan pemrosesan yang cepat[16].

2.15 CSS

CSS (*Cascading Style Sheets*) adalah suatu bahasa *stylesheet* yang digunakan untuk mengatur *style* suatu dokumen. Pada umumnya CSS dipakai untuk memformat tampilan halaman *web* yang dibuat dengan bahasa HTML dan XHTML. CSS merupakan sekumpulan kode-kode program untuk mempercantik tampilan halaman HTML atau dengan kata lain CSS digunakan untuk memisahkan tampilan dari konte. CSS dapat mengubah besar kecilnya text, menjadi warna background pada sebuah halaman, atau dapat pula mengubah warna border pada tabel, dan masih banyak lagi hal yang dapat dilakukan oleh CSS. Singkatnya, CSS digunakan untuk mengatur susunan tampilan halaman HTML[17].

2.16 XAMPP

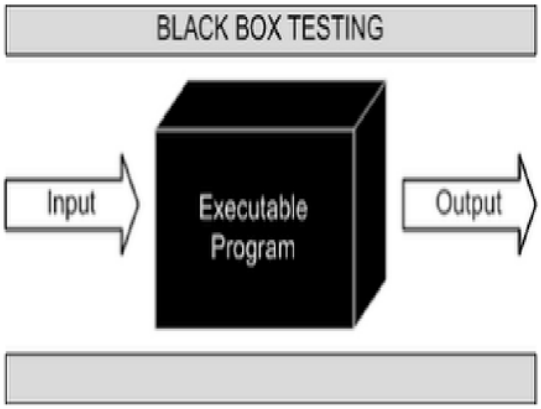
Salah satu paket *web server* yang praktis karena menginstal langsung Apache, PHP, dan MySQL adalah XAMPP. XAMPP adalah aplikasi pengembang yang berbasis PHP dan MySQL dan digunakan untuk menggembangkan sebuag *website.* XAMPP dilengkapi dengan fitur PHPMyAdmin sehingga penggembang dapat mengembangkan database dengan mudah. XAMPP tersedia untuk *multisystem* 19 operasi, seperti Windows dan Linux. Sebenarnya *server* untuk PHP tidak hanya XAMPP, ada banyak sekali *web server*[16].

2.17 *Codeigniter*

Codeigniter dalah sebuah *framework* PHP yang dapat membantu mempercepat *developer* dalam pengembangan aplikasi *web* berbasis php dibandingkan jika menulis semua kode program dari awal. *Codeigniter* menyediakan banyak *library* untuk mengerjakan tugas-tugas yang umumnya ada pada sebuah aplikasi berbasis *web*. Selain itu, struktur dan susunan logis dari *codeigniter* membuat aplikasi menjadi semakin teratur dan rapi[18].

2.18 Metode Pengujian *Black-Box*

Black Box Testing adalah pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan cek fungsional perangkat lunak. Jadi analogi seperti yang kita lihat kotak hitam, kita dapat melihat hanya penampilan luar, tanpa mengetahui apa di balik bungkus hitamnya. Sama seperti pengujian kotak hitam, mengevaluasi hanya penampilan *eksternal* (antarmuka), fungsional tanpa mengetahui apa yang sebenarnya terjadi dalam codingan[19].



Gambar 2.6 Proses pengujian *black-box*
(Sumber : Rekayasa Perangkat Lunak Pendekatan Praktisi (Buku Satu), (Roger S. Pressman, Ph.D. 2002 : 536)

***Black Box Testing* dapat menemukan *error* seperti:**

a. Fungsi atau logika yang tidak benar
b. *Error interface*
c. *Error performance*
d. Kesalahan dalam struktur data atau akses database eksternal

Tujuan *Black Box Testing* adalah untuk mencari kesalahan/kegagalan dalam operasi tingkat tinggi, yang mencakup kemampuan dari perangkat lunak, operasional/tata laksana, skenario pemakai. Fungsi dari pengujian ini berdasarkan kepada apa yang dapat dilakukan oleh sistem. Untuk melakukan pengujian perilaku

seseorang harus mengerti lingkup dari aplikasi, solusi bisnis yang diberikan oleh aplikasi, dan tujuan sistem dibuat[19].

2.19 *Netbeans*

NetBeans adalah *Integrated Development Environment* (IDE) berbasiskan *Java* dari *Sun Microsystems* yang berjalan di atas *Swing. Swing* adalah sebuah teknologi Java untuk pengembangan aplikasi *desktop* yang dapat bejalan di berbagai macam *platforms* seperti Windows, Linux, Mac OS X dan Solaris [20].

Suatu IDE adalah lingkup pemrograman yang diintegrasikan kedalam suatu aplikasi perangkat lunak yang menyediakan pembangun *Graphic User Interface* (GUI), suatu text atau kode editor, suatu *compiler* atau *interpreter* dan suatu *debugger*. *NetBeans* merupakan *software development* yang *Open Source*, dengan kata lain *software* ini di bawah pengembangan bersama dan bebas biaya. *NetBeans* IDE adalah sebuah lingkungan pengembangan sebuah *tools* untuk pemrogram menulis, mengompilasi, mencari kesalahan dan menyebarkan program. *NetBeans* IDE ditulis dalam *Java,* namun dapat mendukung bahasa pemrograman lain. Terdapat banyak modul untuk memperluas Netbeans IDE[20]

Gambar 2.7 Logo netbeans
(Sumber: https://netbeans.org/)

2.20 *Visual Studio Code*

Visual Studio merupakan salah satu perangkat lunak yang digunakan dalam pengembangan aplikasi. Visual Studio Code adalah salah satu teks editor yang dibuat Microsoft dapat digunakan pada sistem operasi multi platform seperti Linux, Windows, dan Mac OS X.Visual Studio Code mendukung bahasa pemograman Java Script, Type Script, Node.js serta bahasa pemograman lainnya dengan bantuan

plugin yang dapat dipasang pada marketplace Visual Studio Code ini seperti C++, C#, Python, PHP, Go, dan Java[21].

Visual Studio terdiri dari kompiler, SDK, *Integrated Development Environment* (IDE), dan dokumentasi (umumnya berupa MSDN Library). Kompiler yang dimasukkan ke dalam paket *Visual Studio* antara lain *Visual* C++, *Visual* C#, *Visual Basic*, NET, *Visual InterDev*, *Visu*al J++, *Visual* J#, *Visual FoxPro*, dan *Visual SourceSafe*. *Visual Studio Code* dalam perancangan ini digunakan untuk editor pembuatan aplikasi berbasis *mobile*[21].

Gambar 2.8 Logo visual studio code
(Sumber : https://medium.com/edgefund/c-development-with-visual-studio-code-b860cc71a5ec)

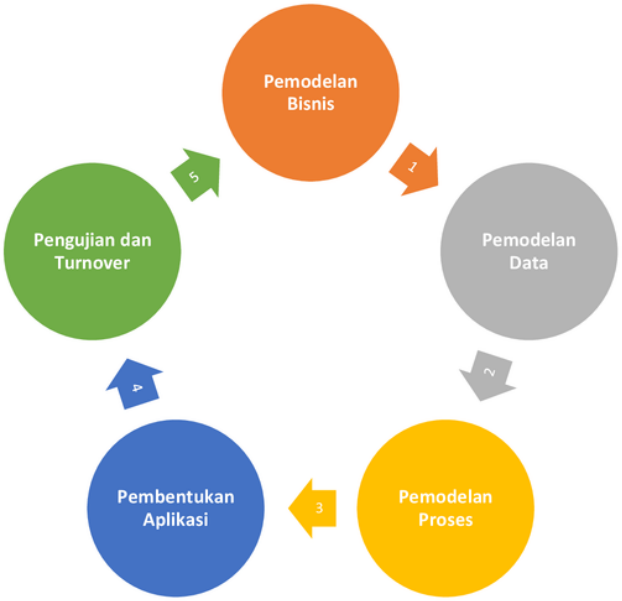
2.21 *Bootstrap*

Bootstrap merupakakan salah satu jenis *framework* untuk CSS (*Cascading Style Sheet*) yang sangat membantu bagi *programmer* untuk perancangan tampilan situs *website*. Penggunaan *Bootstrap* bisa untuk memilih fitur CSS dan *JavaScript* sesuai dengan kebutuhan atau menyesuaikan dengan kebutuhan *website.* Contohnya CSS pada *Bootstrap* juga menyediakan fitur *form*, tombol, navigasi dan komponen-komponen lainnya dan *JavaScript* yang membantu pembuatan antarmuka lebih mudah dan stabil. *Boothstrap* saat ini menjadi salah satu *front-end framework* yang banyak digunakan oleh *programmmer* karena *boothstrap* juga menyediakan banyak *class-class* CSS dan *plug ini*, namun harus dikombinasikan dengan penggunaan *stylesheet*. *Stylesheet* digunakan sebagai pembeda tampilan *websitte* supaya tidak ada kemiripan antar tampilan *website*[22].

BAB III
METODOLOGI

3.1 Metodologi Pengembangan Sistem

Metode pengembangan pada perancangan aplikasi ujian *online* berbasis *android* di Jurusan Teknik Elektro Universitas Sriwijaya ini mengacu pada metode RAD (*Rapid Application Development)* karena sistem pengembangannya cepat dan terarah. Berikut tahapan-tahapan pengembangan sistem menggunakan metode RAD (*Rapid Application Development)*ditunjukkan pada gambar 3.1 dibawah ini:



Gambar 3.1 Metode pengembangan RAD (*Rapid Application Development*)

3.2 Tahapan-Tahapan Metodelogi Pengembangan Sistem

Berdasarkan gambar 3.1 terdapat 5 tahapan yaitu tahapan pemodelan bisnis, pemodelan data, pemodelan proses, perancangan aplikasi, pengujian dan perbaikan. Berikut ini penjelasan dari ke 5 tahapan tersebut:

3.2.1 Pemodelan Bisnis

Pemodelan bisnis merupakan tahap awal untuk memulai suatu perancangan dan pengembangan sistem informasi pada metode RAD (*Rapid Application Development*). Tahapan awal untuk membuat pemodelan rancangan agar mudah dipahami. Pada tahapan ini dilakukan analisa dan pengumpulan data untuk membangun aplikasi yang sesuai dengan yang diharapkan. Proses yang dilakukan yaitu dengan melakukan obeservasi dan penyebaran kuesioner kepada beberapa responden, kemudian dilakukan analisa untuk menentukan bagaimana aplikasi yang sesuai dengan yang diharapkan. Pada pemodalan bisnis ini dijelaskan alasan mengapa sistem dibuat, dan bagaiman sistem akan berjalan, menentukan waktu dan lokasi untuk sistem dapat beroperasi dan menentukan bagaimana perancangan sistem sendiri. Untuk membantu menidentifikasi hal-hal tersebut, maka digunakan batuan *tools* 5W+1H Pada tahapan ini juga mengidentifikasi hubungan dengan eksternal sistem menggunakan *tools* metode analisis SOAR (*Strengths, Opportunities, Aspirations, Result*) yang bertujuan menentukan kelebihan dan kekurangan pada sistem, kesempatan, keuntungan dan peluang apa yang akan didapat, menentukan aspirasi dari berbagai pihak dalam perancangan dan pengembangan sistem, dan tujuan yang akan dicapai untuk sistem ini. Pada tahapan ini juga dilakukan analisis kebutuhan fungsional dan non-fungsional. Dimana kebutuhan fungsional adalah kebutuhan yang berhubungan dengan proses-proses pada sistem sedangkan kebutuhan non fungsional adalah kebutuhan yang mengacu kepada perilaku sistem. Dilakukan perancangan

3.2.2 Pemodelan Data

Pada tahap pemodelan data adalah bagian dari pemodelan bisnis dari serangkaian objek-objek data yang dibutuhkan untuk menunjang bisnis yang kemudian akan disusun dan dikumpulkan menjadi sekumpulan objek data yang akan digunakan untuk perancangan *database* dengan atribut-atributnya yang digambarkan dengan bantuan *tools class diagram* pada UML 2.0. *Class diagram* akan memodelkan hubungan antara kelas-kelas, atribut-atributnya dan juga operasi-

operasi, serta asosiasi hubungan dengan kelas lain. Sehingga mempermudah untuk penyusunan struktur pada *database*.

3.2.3 Pemodelan Proses

Pada tahap pemodelan proses akan dianalisis bagaimana proses sistem berjalan. *Input* dari pemodelan bisnis adalah gabungan dari hasil pemodelan bisnis dengan pemodelan data. Pada tahap ini dilakukan proses untuk menentukan fitur atau konten yang akan disematkan pada sistem, aktor atau pengguna yang terlibat pada sistem digambarkan dengan bantuan *tools use case diagram* digunakan untuk menggambarkan tindakan antar pengguna dan admin di dalam sistem yang diharapkan. Tahapan ini juga akan dilakukan analisa mengenai proses sistem usulan, dimana akan dilakukan modifikasi, penambahan, penghapusan objek data dan dilihat apakah sistem berjalan sesuai dengan skenario perancangan. Tahapam ini menggambarkan bagaimana prilaku dan aksi-aksi dari pengguna terhadap sistem melalui aliran kontrol dengan bantuan *tools activity diagram,* diagram yang menunjukkan urutan antara objek-objek selama eksekusi suatu pekerjaan berlangsung yang digambarkan dengan bantuan *tools squence diagrams*. Untuk mempermudah perancangan tersebut maka penulis menggunakan bantuan tools UML 2.0.

3.2.4 Perancangan Aplikasi

Tahap ini merupakan tahapan pengkodean atau coding menggunakan bahasa pemprograman seperti HTML5, PHP, CSS, Javascript dan MySQL. Bahasa pemrograman HTML5, PHP dan CSS digunakan untuk merancang *website* pada ujian *online*. Javascript adalah bahasa pemprograman yang digunakan pada perancangan aplikasi ujian *online* yang berbasis *mobile android*. Sedangkan MySQL adalah *server database* pada aplikasi ujian *online*. Selain itu pada tahapan ini juga dilakukan desain untuk tampilan (*user interface*). *Output* yang dihasilkan pada tahap ini berupa perangkat lunak (*software*) pada aplikasi ujian *online* berbasis *android* di Jurusan Teknik Elektro Universitas Sriwijaya. Perangkat lunak yang

akan dihasilkan berbentuk aplikasi untuk operasi pada *android* dan *website* untuk sistem admin.

3.2.5 Pengujian dan *Turnover*

Tahap pengujian dan *turnover* adalah tahapan terakhir pada metode pengembangan sistem ini, bertujuan untuk menguji semua elemen dari perangkat lunak atau aplikasi yang sudah dihasilkan apakah sesuai dengan yang diharapakan. Input pada tahap ini adalah output secara keseluruhan dari semua tahapan-tahapan yang telah dilakukan Pada tahapan ini, aplikasi atau perangkat lunak yang sudah dirancang akan dilakukan pengujian fungsional sistem menggunakan metode pengujian *black-box*. Apabila hasil akhir yang didapat sesuai dengan yang diharapkan dan tidak ada lagi perubahan maka tidak akan terjadi proses pengulangan. *Output* yang dihasilkan pada tahap ini adalah dokumen hasil pengujian *black-box test*.

Tabel 3.1 Tahapan metode pengembangan sistem

No.	Tahapan	Input	Proses	Output
1.	Pemodelan Bisnis	Melakukan pengamatan terhadap lingkungan untuk mencari data informasi yang diperlukan.	Pencarian data informasi yang diperlukan.	Dokumen/data hasil observasi berurpa data kuesioner
2.	Pemodelan Data	Output dari proses pemodelan bisnis	Pemodelan objek-objek dengan bantuan *class diagram*.	Dokumen/data dari proses tahapan pemodelan data dengan bantuan *class diagram*.

3.	Pemodelan Proses	Dokumen/data dari proses perencanaan bisnis dan pemodelan data.	Memodelkan alur aktivitas dan interaksi pada sistem dengan bantuan *usecase diagram, activity diagram* dan *sequence diagram*	Dokumen/data aktivitas sistem dengan bantuan *use case diagram, activity diagram* dan *sequence diagram*.
4.	Pembuatan Aplikasi	Dokumen/data hasil dari tahap keseluruhan perancangan dan juga desain *interface*.	Penyusunan dengan menggunakan bahasa pemrograman.	Hasil *interface* pada sistem. Mahasiswa (*mobile*), Dosen (*website*), Admin (*website*).
5.	Pengujian dan *Turnover*	Sistem yang sudah siap untuk diuji	Menguji sistem dengan pengujian *Black box* dan kuesioner hasil pegujian sistem	Dokumen/data hasil pengujian *Black box* dan kuesioner pada sistem

BAB IV
PERANCANGAN

4.1 Pemodelan Bisnis

Tahapan pemodelan bisnis adalah tahapan awal dalam Perancangan Aplikasi Ujian *Online* Berbasis *Android* di Teknik Elektro Universitas Sriwijya. Dimana dilakukan analisa dan pengumpulan data untuk membangun aplikasi yang sesuai dengan yang diharapkan. Proses yang dilakukan yaitu dengan melakukan obeservasi dan penyebaran kuesioner kepada beberapa responden, kemudian dilakukan analisa untuk menentukan bagaimana aplikasi yang sesuai dengan yang diharapkan. Pada tahap ini juga dilakukan identifikasi hal-hal yang berkaitan dengan aplikasi secara internal maupun secara eksternal. Untuk membantu tahapan ini digunakan *tools* 5W +1H dan SOAR (*Strength, Opportunities, Aspiration, Result*).

Hasil observasi yang didapatkan adalah sistem ujian yang berjalan selama ini masih menggunakan sistem konvensional atau manual. Dimana pelaksanaan ujian masih menggunakan kertas dan alat tulis dan hasil ujian diperiksa satu persatu secara manual. Tentu memiliki kekurangan dalam hal efektifitas, waktu, dan kinerja dan juga mengurangi kecurangan pada saat melakukan ujian. Sehingga dibuat sistem ujian berbasis *android* yang efektif dimana mahasiswa dapat melakukan ujian dalam bentuk esai yang dapat melatih dalam menyampaikan suatu informasi dan opini yang ada, ujian *online* juga menuntut untuk berfikir secara kritis tentang pemahaman suatu ilmu secara mendalam.

Proses selanjutnya yang dilakukan adalah pengumpulan data dengan dilakukan penyebaran kuesioner dan analisa dengan berbagai pihak antara lain mahasiswa dan dosen sebagai calon pengguna aplikasi yaitu dengan mengajukan beberapa pertanyaaan/kuisioner yang nantinya akan menentukan kelayakan aplikasi yang akan dibuat.

Berdasarkan hasil data kuesioner terbuka dengan menggunakan *google form* yang dilakukan dari tanggal 24 Januari 2019-2 Februari 2019 dengan responden adalah mahasiswa jurusan Teknik Elektro Universitas Palembang Angkatan 2016

didapat bahwa hampir semua mahasiswa adalah pengguna perangkat *smartphone android* dan sudah sangat familiar dengan aplikasi-aplikasi yang ada pada perangkat *android.* Hasil kuesioner masing-masing responden akan dilampirkan pada **lampiran**. Adapun pertanyaan-pertanyaan dari kuesioner tersebut terdiri dari sebagai berikut:

1. Apakah Anda pengguna *smartphone* ?

   - ○ Ya
   - ○ Tidak

2. Anda biasa menggunakan *Smartphone* untuk keperluan apa saja?

   - ○ *Youtube*
   - ○ Mencari Materi Pelajaran
   - ○ Membaca Berita
   - ○ Bermain *Games*
   - ○ Berkomunikasi (*chatting, email*, dll)

3. Bagaimana sistem ujian yang dipakai selama ini ?

   - ○ Ujian Manual
   - ○ Ujian *Online*

4. Jenis soal ujian yang biasa Anda temui pada saat ujian?

   - ○ Pilihan Ganda
   - ○ Esai

5. Bagaimana penilaian Anda mengenai sistem ujian yang ada di Teknik Elektrro?

- ○ Kurang
- ○ Cukup
- ○ Baik

6. Apakah Anda setuju jika kegiatan ujian dilakukan secara *online* menggunakan *smartphone* ?
   - ○ Setuju
   - ○ Tidak Setuju

7. Harapan apa saja kedepannya yang Anda inginkan pada Aplikasi Ujian *Online* ini ?

Analisa yang didapat dari pengumpulan data dari kuesioner dimana terdapat 7 soal pertanyaan yang terdiri dari 6 soal objektif dan 1 soal esai yang diisi oleh mahasiswa jurusan Teknik Elektro Universitas Sriwijya.

1. Pada pertanyaan pertama yang diajukan penulis, membahas seberapa banyak mahasiswa teknik elektro yang sudah menggunakan perangkat *smartphone android*. Dari hasil data kuesioner yang didapat dari 32 responden 100% adalah pengguna perangkat *smartphone android*. Hal ini bisa kita lihat bahwa penggunaan *smartphone android* dikalangan mahasiswa teknik elektro sudah banyak dan familiar.

2. Pada pertanyaan kedua yang diajukan penulis, membahas bagaimana penggunaan perangkat *smartphone android* sehari-hari. Dari pilihan yang diberikan 31 responden menjawab untuk *youtube* 90.3 %, mencari materi kuliah 93.5 %, membaca berita 77.4 %, bermain *games* 77.4 % dan berkomunikasi sebanyak 96.8 %. Dari hasil data yang didapat tersebut, bisa

disimpulkan bahwa penggunaan *smartphone* paling tinggi adalah untuk berkomunikasi dan yang kedua untuk mencari materi kuliah.

3. Pada pertanyaan ketiga yang diajukan penulis, membahas mengenai bagaimana sistem ujian yang berlangsung di Teknik Elektro Universitas Sriwijaya. Dari 34 responden yang menjawab, sebanyak 91.2 % mengatakan ujian yang dilakukan selama ini masih bersifat ujian manual, dan yang yang menjawab sebanyak 14.7%. Ujian manual adalah ujian yang masih dilakukan dengan menggunakan kertas dan alat tulis dengan sistem penggumpulan dan pengkoreksian masih bersifat manual.

4. Pada pertanyaan keempat yang diajukan penulis, membahas mengenai jenis ujian yang biasa ditemui di Teknik Elektro Universitas Sriwijaya. Dari 12 responden yang menjawab sebanyak 91.7 % soal ujian yang biasa ditemui yaitu esai, dan 8.3 % pilihan ganda. Dari data yang didapat dapat disimpulkan bahwa di jurusan teknik elektro lebih banyak menggunakan soal berjenis esai dibanding dengan pilihan ganda. Karena mahasiswa dituntut untuk bisa berfikir secara kritis dan bisa menganalisa suatu masalah.

5. Pada pertanyaan kelima yang diajukan penulis, membahas mengenai pendapat mahasiswa Teknik Elektro Universitas Sriwijaya apabila kegiatan ujian manual digantikan dengan ujian *online* menggunakan *smartphone android.* Dari 32 responden menjawab setuju dengan persentase sebesar 65.6 %, dan tidak setuju sebanyak 32.4 %. Dari data yang didapat, cukup banyak mahasiswa teknik elektro tertarik dengan rencana mengganti sistem ujian manual dengan ujian *online.*

6. Pada pertanyaan keenam yang diajukan penulis, membahas mengenai pendapat mahasiswa Teknik Elektro Universitas Sriwijaya mengenai penilaian sistem ujian yang selama ini digunakan di Teknik Elektro. Dari 30

responden yang menjawab pertanyaan dengan pilihan yang telah disediakan, sebanyak 6.7 % menjawab kurang, 56.7 % menjawab cukup, dan 36.7 % menjawab baik.

7. Pertanyaan terakhir merupakan pertanyaan esai yang mengenai tentang harapan apa saja kedepannya untuk aplikasi ujian *online* yang akan dibuat. Dimana responden menyampaikan masukkan dan saran berdasarkan apa yang mereka inginkan nantinya. Berdasarkan jawaban-jawaban yang diberikan oleh para responden dapat disimpulkan, untuk rencana perancangan aplikasi ujian *online* berbasis *android* ini harapan dari responden adalah sebuah aplikasi yang dapat memudahkan dalam pengerjaan ujian dan tentunya lebih baik dari sistem ujian manual, memiliki sistem keamanan yang baik, dan serta memiliki koneksi yang bagus.

Berdasarkan hasil kesimpulan dari kuesioner yang telah dilakukan, maka perancangan aplikasi ujian *online* berbasis *android* di jurusan teknik elektro universitas sriwijaya ini layak dilanjutkan ketahap berikutnya, yaitu tahapan analisa sistem, baik secara internal maupun secara exsternal menggunakan bantuan *tools* 5W+1H dan analisis SOAR (*Strength, Opportunities, Aspiration, Result*)

4.1.1 Tahap Pemodelan Bisnis secara Internal

Pada tahapan ini dilakukan identifikasi hal-hal yang berhubungan dengan sistem secara internal menggunakan bantuan analisis metode kipling 5W + 1H (*What, Where, When, Who, Why,* dan *How*), yaitu sebagai berikut :

1. ***What***

   Sistem yang akan dirancang adalah aplikasi ujian *online* yang berbasis *mobile android*.. Aplikasi ini melayani ujian secara *online* dan akan menampilkan hasil ujian pada aplikasi berupa nilai.

2. ***Why***

   Aplikasi ini lebih mempermudah baik mahasiswa maupun dosen dalam melaksanakan ujian secara *online* yang dapat di akses pada perangkat *mobile*

berbasis *Android* dibanding metode konvensional yang dipakai saaat ini. Ujian dengan metode konvensional masih banyak memiliki kekurangan antara lain pemborosan kertas juga terbatas ruang dan waktu. Aplikasi ini dibuat untuk membuat sistem ujian menjadi lebih efektif, lebih menghemat waktu dan dapat mengurangi penggunaan kertas. Aplikasi ini diharapkan supaya kendala yang biasanya ditemui pada saat pelaksanaan ujian secara konvensional, antara lain kekurangan kertas soal, kertas jawaban, ataupun tulisan mahasiswa yang tidak terbaca pada saat pengoreksian jawaban .

3. *Who* (Siapa Menggunakan Aplikasi dan Siapa Saja Yang Terlibat)

Pengguna yang terlibat dalam implementasi dari perancangan aplikasi ujian online berbasis *Android* di Jurusan Teknik Elektro Universitas Sriwijaya yaitu :

1. Mahasiswa

   Merupakan mahasiswa yang memiliki *smartphone android* yang akan melakukan ujian dan yang telah terdaftar akunnya.

2. Dosen

   Merupakan dosen/pengajar yang membutuhkan sistem untuk mengelola ujian dan nilai *online* mahasiswa di Jurusan Teknik Elektro Universitas Siwijaya

3. Admin

   Admin bertugas untuk mengawasi dan mengatur sistem yang dibuat. Admin juga bertugas menampilkan soal ujian, menampilkan nilai ujian dll.

4. *When* (Kapan aplikasi ini dapat digunakan?)

Aplikasi ini dapat digunakan pada perangkat *mobile android*. Pengguna harus men*download* dan meng*install* aplikasi. Aplikasi ini dapat diakses kapan dan dimana saja selama terkoneksi dengan internet dan sudah terdaftar pada sistem.

5. ***Where* (Dimana aplikasi ini digunakan)**

   Aplikasi ini akan digunakan di lingkungan Jurusan Teknik Elektro Universitas Sriwijaya.

6. ***How* (Bagaimana aplikasi dirancang)**

   Secara teknis aplikasi ini akan dirancang untuk aplikasi *mobile* yang akan digunakan oleh *user* pada *flatform android* sedangkan perancangan aplikasi *website* akan digunakan oleh admin. Data pada sistem ini akan tersimpan *database* admin. Sedangkan untuk rancangan aplikasi pada sisi pengguna, user (mahasiswa dan dosen) dapat men*download* dan *install* aplikasi pada perangkat *mobile android*, dan dapat digunakan dimana saja selama terhubung dengan internet dengan jaringan yang baik. Mahasiswa dapat mengakses ujian *online* sesuai dengan mata kuliah yang sudah dijadwalkan. Sedangkan dosen yang mengolah mata kuliah, mengelola ujian, mengelola hasil ujian dan mengelola nilai.

4.1.2 Tahap Pemodelan Bisnis secara Eksternal

Pada tahapan ini dilakukan identifikasi hal-hal yang berhubungan dengan sistem secara eksternal menggunakan bantuan *tools* analisis SOAR yaitu *Strength* (kekuatan), *Opportunities* (peluang), *Aspiration* (aspirasi), *Result* (hasil). Berikut tabel analisis SOAR untuk perancangan aplikasi ujian online berbasis *android* di teknik elektro universitas sriwijaya

Tabel 4.1 Tabel SOAR

***Strenghts* (Kekuatan)**	***Oppertunities* (Kesempatan)**
1. Seiring perkembangan zaman banyak mahasiswa yang sudah familiar dalam penggunakan perangkat *smartphone android.* 2. Di era sekarang kemajuan teknologi sudah semakin pesat yang menuntut *high technology* diberbagai bidang. 3. Aplikasi ini mudah digunakan untuk pelaksanaan ujian.	1. Sebagai media alternatif pengganti ujian konvensional atau ujian *online* 2. Banyaknya mahasiswa dan dosen yang sudah menggunakan perangkat *mobile android* yang menjadikan aplikasi ini diminati. Karena selain praktis, aplikasi ini juga lebih menghemat menghemat waktu pelaksanaan ujian.
***Aspiration* (Aspirasi)**	***Result* (Hasil)**
1. Adanya sistem ujian secara *online* yang dapat digunakan dengan mudah/*userfriendly* 2. Sistem ujian yang memiliki keamanan yang baik. 3. Memberikan kemudahan baik dosen maupun mahasiswa dalam pelaksanaan ujian dan pengelolahan hasil ujian.	Sebuah aplikasi ujian *online* berbasis *android* di jurusan teknik elektro universitas sriwijaya.

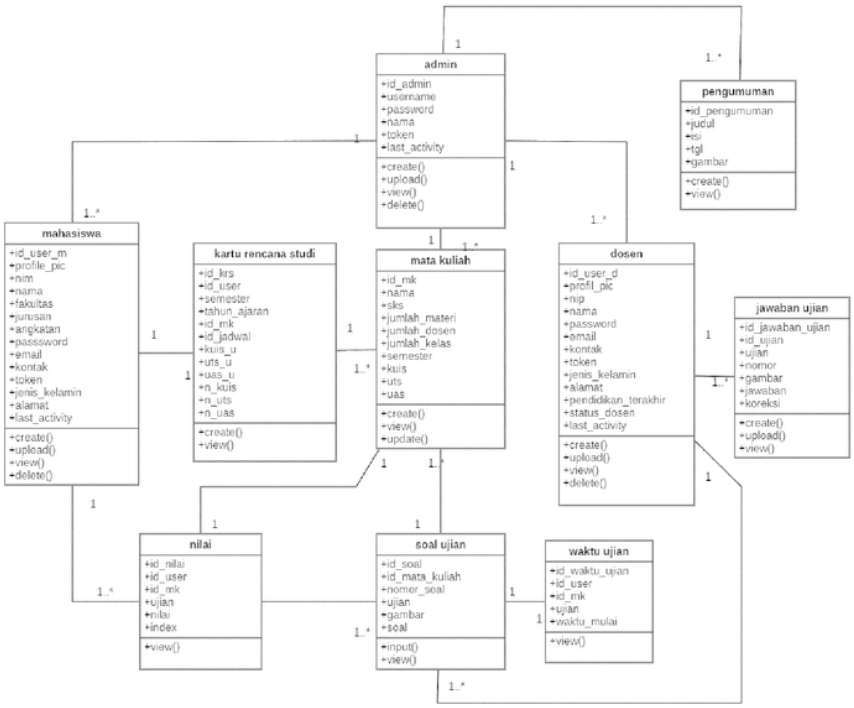
Berdasarkan tabel dengan menggunakan bantuan analisis SOAR diatas, maka diperoleh hasil berupa hubungan antara keempat analisis SOAR tersebut sehingga menghasilkan penyataan bahwa sistem ini berguna untuk masyarakat

Tabel 4.2 Matriks SOAR

	Strength	***Opportunities***
Aspiration	1. Kemajuan teknologi saat ini membuat penggunaan perangkat *smartphone android* sangat familiar. 2. Merancang dan membangun sistem aplikasi ujian *online* yang dapat digunakan dengan mudah bagi mahasiswa maupun dosen juga memiliki tampilan yang menarik dengan sistem keamanan yang baik.	1. Menjadi alternatif pengganti ujian konvensional yang memberikan kemudahan bagi para mahasiswa dosen, seperti pelaksanaan ujian dan pengelolahan hasil ujian dilakukan secara *online*. 2. Merancang aplikasi ujian online yang dapat menghemat waktu dan efisien.
Result	Merancang sebuah sistem aplikasi ujian *online* yang dapat dapat diakses dengan perangkat *android*, karena perkembangan zaman sangat pesat yang menuntut teknologi tinggi diberbagai bidang.	1. Sebagai media pengganti ujian konvensional yang dapat diakses dengan menggunakan perangkat *android* 2. Banyaknya mahasiswa yang sudah familiar dengan penggunaan perangkat *android* sehingga memberikan kemudahan untuk akses aplikasi yang telah dibuat. 3. meningkatnya masyarakat yang menggunakan *smartphone* sehingga memudahkan pengoperasian aplikasi dari sistem yang telah dihasilkan.

4.2 Pemodelan Data

Tahap pemodelan data adalah bagian dari pemodelan bisnis dari serangkaian objek-objek data yang dibutuhkan untuk menunjang bisnis yang kemudian akan disusun dan dikumpulkan menjadi sekumpulan objek data yang akan digunakan untuk perancangan *database* dengan atribut-atributnya yang digambarkan dengan bantuan *tools class diagram* pada UML 2.0. *Class diagram* akan memodelkan hubungan antara kelas-kelas, atribut-atributnya dan juga operasi-operasi, serta asosiasi hubungan dengan kelas lain. Sehingga mempermudah untuk penyusunan struktur pada *database*.



Gambar 4.1 ***Class diagram***

Berdasarkan *class diagram* pada Gambar 4.1 diatas, maka dibuat struktur tabel yang akan diimplementasikan menjadi *database* yaitu sebagai berikut :

Tabel 4.3 Tabel mahasiswa

No	Field	Type	Length
1.	id_user_m	Int	11
2.	profil_pic	Varchar	40
3.	nim	Varchar	14
4.	nama	Varchar	60
5.	fakultas	Varchar	40
6.	jurusan	Varchar	40
7.	angkatan	Int	4
8.	password	Varchar	40
9.	email	Varchar	40
10.	kontak	Varchar	13
11.	token	Varchar	40
12.	jenis_kelamin	Varchar	1
13.	alamat	Varchar	200
14.	last_activity	Varchar	30

Tabel 4.4 Tabel dosen

No	Field	Type	Length
1.	id_user_d	Int	11
2.	profil_pic	Varchar	40
3.	nip	Varchar	14
4.	nama	Varchar	60
5.	password	Varchar	40
6.	email	Varchar	40
7.	kontak	Varchar	13
8.	token	Varchar	40
9.	jenis_kelamin	Varchar	1
10.	alamat	Varchar	100
11.	pendidikan_terakhir	Varchar	30

12.	status_dosen	Varchar	40
13	last_activity	varchar	30

Tabel 4.5 Tabel admin

No	Field	Type	Length
1.	id_admin	INT	11
2.	username	VAR	30
3.	password	VAR	40
4.	nama	VAR	50
5.	token	VAR	40
6.	last_activity	VAR	40

Tabel 4.6 Tabel mata kuliah

No	Field	Type	Length
1.	id_mk	Int	11
2.	nama	Varchar	60
3.	sks	Int	1
4.	jumlah_materi	Int	3
5.	jumlah_dosen	Int	1
6.	jumlah_kelas	Int	2
7.	semester	Varchar	8
8.	kuis	Int	11
9.	uts	Int	11
10.	uas	Int	11

Tabel 4.7 Tabel kartu rencana studi

No	Field	Type	Length
1.	id_krs	Int	11

2.	id_user	Int	11
3.	semester	Varchar	8
4.	tahun_ajaran	Varchar	11
5.	Id_mk	Int	11
6.	id_jadwal	Int	11
7.	kuis_u	Int	11
8.	uts_u	Int	11
9.	uas_u	Int	11
10.	n_kuis	Int	3
11.	n_uts	Int	3
12.	n_uas	Int	3

Tabel 4.8 Tabel soal ujian

No	Field	Type	Length
1.	id_soal	Int	11
2.	id_mata_kuliah	Int	11
3.	nomor_soal	Int	11
4.	ujian	Varchar	4
5.	gambar	Varchar	50
6.	soal	Varchar	200

Tabel 4.9 Nilai

No	Field	Type	Length
1.	id_nilai	Int	11
2.	id_user	Int	11
3.	id_mk	Int	11
4.	ujian	Int	11
5.	nilai	Int	11

6.	index	Varchar	1

Tabel 4.10 Tabel pengumuman

No	Field	Type	Length
1.	id_pengumuman	Int	11
2.	judul	Varchar	40
3.	isi	Varchar	200
4.	tgl	Varchae	30
5.	gambar	Varchar	40

Tabel 4.11 Jawaban ujian

No	Field	Type	Length
1.	id_jawaban_ujian	Int	11
2.	id_ujian	Int	11
3.	ujian	Int	11
4.	nomor	Int	11
5.	gambar	Int	11
6.	jawaban	Int	11
7.	koreksi	Int	11

Tabel 4.12 Waktu ujian

No	Field	Type	Length
1.	id_waktu_ujian	Int	11
2.	id_user	Int	11
3.	id_mk	Int	11
4.	ujian	Int	11

5.	waktu_mulai	Int	11
6.	waktu_selesai	Int	11

4.3 Pemodelan Proses

Pada tahap pemodelan proses akan dianalisis bagaimana proses sistem berjalan. *Input* dari pemodelan bisnis adalah gabungan dari hasil pemodelan bisnis dengan pemodelan data. Tahapan ini akan diilakukan analisa mengenai proses sistem usulan, dimana akan dilakukan modifikasi, penambahan, penghapusan objek data dan dilihat apakah sistem berjalan sesuai dengan skenario perancangan. Pada tahapan ini dilakukan *requirements planning* yang bertujuan untuk menganalisisi kebutuhan dari sistem yang akan dibuat, aktor atau pengguna yang terlibat pada sistem digambarkan dengan bantuan *tools use case diagram*, menggambarkan bagaimana prilaku dan aksi-aksi dari pengguna terhadap sistem melalui aliran kontrol dengan bantuan *tools activity diagram,* dan urutan antara objek-objek selama eksekusi suatu pekerjaan berlangsung yang digambarkan dengan bantuan *tools squence diagrams*.

4.3.1 Tindakan Yang Dilakukan Pengguna terhadap Sistem

Tindakan yang dilakukan menggambarkan aktifitas oleh pengguna didalam sistem dimodelkan dengan *usecase diagram* menggunakan *tool* UML 2.0 seperti gambar dibawah ini :

4.3.1.1 Interaksi Yang Dilakukan Mahasiswa Terhadap Sistem.

Gambar 4.2 Diagram interaksi mahasiswa terhadap sistem

Pada gambar 4.2 menjelaskan ineraksi antara mahasiswa terhadap sistem. Hal pertama yang dilakukan mahasiswa untuk dapat mengakses aplikasi yaitu dengan melakukan *login username* dan *password* yang *valid* terlebih dahulu. Setelah *login* maka akan menampilkan halaman pilihan. Mahasiswa memilih "Ujian" maka sistem akan masuk ke halaman utama ujian mahasiswa. Pada halaman utama. Pada halaman utama mahasiswa, mahasiswa dapat mengganti *password* dan *logout* pada menu *profile.* Pada menu ujian mahasiswa dapat melakukan ujian secara *online* terjadwal seperti kuis, ujian tengah semester (UTS) dan ujian akhir semester (UAS) dan juga bisa melihat nilai hasil ujian. Serta mahasiswa dapat mengetahui berita atau pengumuman yang berhubungan dengan akademik pada menu pengumuman.

Tabel 4. 13 Deskripsi diagram interaksi *user* terhadap sistem

No.	Usecase	Deskripsi
1.	*Login*	Kegiatan *user* untuk masuk ke menu utama *user* dengan cara menginputkan *username* dan *password* dengan benar.
2.	Menu Pilihan	Halaman setelah *user* berhasil *login* ke sistem, dimana *user* dapat memilih menu seperti Absensi, Materi Pembelajaran, Ujian dan Bimbingan Tugas Akhir.
3.	Menu *Profile*	Kegiatan yang dilakukan *user* untuk masuk ke halaman profile dan untuk melakukan ganti *password* dan *logout.*
4.	Ganti *Password*	Kegiatan user mengubah *password* atau kata sandi.
5.	*Logout*	Kegiatan untuk keluar dari aplikasi dengan memilih menu *logout* yang dilakukan oleh *user*

6.	Masuk Halaman Utama Mahasiswa	Halaman utama mahasiswa untuk melakukan segala kegiatan atasu aktifitas pada sistem aplikasi.
7.	Masuk Menu *Profile*	Kegiatan mahasiswa untuk masuk ke halaman menu *profile*.
8.	Masuk Menu Kelola Ujian	Kegiatan untuk masuk ke halaman menu ujian
9.	Melihat Ujian	Kegiatan yang dilakukan untuk melihat ujian yang akan dilaksanakan
10.	Mengerjakan Ujian	Kegiatan yang dilakukan untuk mengerjakan ujian secara *online* terjadwal seperti kuis, ujian tengah semester (UTS) dan ujian akhir semester (UAS).
11.	Masuk Menu Kelola Nilai	Kegiatan untuk masuk ke halaman menu nilai
12.	Melihat Nilai	Kegiatan untuk melihat nilai dari hasil ujian yang sudah dilakukan secara *online*.
13.	Masuk Menu Pengumanan	Kegiatan untuk masuk ke halaman menu pengumuman.
14.	Melihat Pengumuman	Kegiatan untuk melihata pengumuman yang berhubungan dengan ujian online.

4.3.1.2 Interaksi yang dilakukan dosen terhadap sistem.

Gambar 4.3 Diagram interaksi dosen terhadap sistem

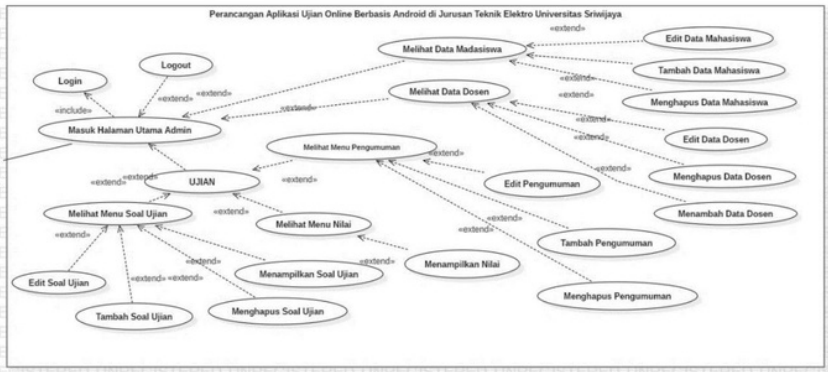
Pada gambar 4.3 menjelaskan ineraksi antara mahasiswa terhadap sistem. Hal pertama yang dilakukan mahasiswa untuk dapat mengakses aplikasi yaitu dengan melakukan *login username* dan *password* yang *valid* terlebih dahulu. Setelah *login* maka akan menampilkan halaman pilihan. Dosen memilih "Ujian" maka sistem akan masuk ke halaman utama dosen. Pada halaman utama dosen, dosen dapat mengganti *password* dan *logout* pada menu *profile.* Dosen bisa menginput soal ujian dan koreksi soal ujian dengan memilih "Kelola Soal Ujian" kemudian sistem akan menampilkan halaman input soal ujian dan koreksi soal ujian. Dosen juga bisa input nilai hasil ujian dengan memilih "Kelola Nilai" maka sistem akan menampilkan halaman input nilai.

Tabel 4. 14 Deskripsi diagram interaksi dosen terhadap sistem

No.	Usecase	Deskripsi
1.	*Login*	Kegiatan *user* untuk masuk ke menu utama *user* dengan cara menginputkan *username* dan *password* dengan benar.
2.	Menu Pilihan	Halaman setelah *user* berhasil *login* ke sistem, dimana *user* dapat memilih menu seperti Absensi, Materi Pembelajaran, Ujian dan Bimbingan Tugas Akhir.
3.	Menu *Profile*	Kegiatan yang dilakukan *user* untuk masuk ke halaman profile dan untuk melakukan ganti *password* dan *logout.*
4.	Ganti *Password*	Kegiatan user mengubah *password* atau kata sandi.
5.	*Logout*	Kegiatan untuk keluar dari aplikasi dengan memilih menu *logout* yang dilakukan oleh *user*

6.	Masuk Halaman Utama Dosen	Halaman utama dosen untuk melakukan segala kegiatan atau aktifitas pada sistem aplikasi.
7.	Masuk Menu Ujian	Kegiatan untuk masuk ke halaman menu ujian.
8.	Input Soal Ujian	Kegiatan menginput soal ujian ke sistem secara online.
9.	Koreksi Soal Ujian	Kegiatan mengkoreksi soal ujian oleh dosen setelah mahasiswa melakukan ujian secara online.
10.	Masuk Menu Nilai	Kegiatan untuk masuk ke halaman menu nilai
11.	Input Nilai	Kegiatan menginput nilai ujian ke sistem secara online.

4.3.1.3 Interaksi yang dilakukan admin terhadap sistem



Gambar 4.4 Diagram interaksi admin terhadap sistem

Pada gambar diatas menjelaskan interaksi tindakan antara admin terhadap sistem. Hal pertama yang dilakukan admin adalah *login* dengan *username* dan *password* untuk dapat menggunakan *web* admin. Setelah *login* maka akan menampilkan halaman utama admin. Admin dapat melihat menu-menu yang ada di dalam *web* admin. Salah satunya menu "Ujian". Pada halaman ujian admin dapat mengedit, menambah, menghapus dan menampilkan soal ujian, juga dapat

menampilkan nilai. Pada halaman ujian ini admin juga dapat mengedit, menambah dan menghapus pengumuman.

Tabel 4. 15 Deskripsi diagram interaksi admin terhadap sistem

No.	Usecase	Deskripsi
1.	*Login*	Kegiatan admin untuk masuk ke menu utama admin dengan cara menginputkan *username* dan *password* dengan benar.
2.	*Logout*	Kegiatan untuk keluar dari aplikasi dengan memilih menu *logout* yang dilakukan oleh admin
3.	Masuk Halaman Utama Admin	Halaman utama untuk melakukan segala kegiatan atasu aktifitas.
4.	Masuk Menu Ujian	Kegiatan yang dilakukan admin untuk masuk ke halaman menu ujian
5.	Melihat Soal Ujian	Kegiatan yang dilakukan admin untuk melihat halaman menu soal ujian.
6.	Edit Soal Ujian	Kegiatan yang dilakukan admin apabila ingin mengedit soal ujian
7.	Tambah Soal Ujian	Kegiatan yang dilakukan admin apabila ingin menambah soal ujian
8.	Menghapus Soal Ujian	Kegiatan yang dilakukan admin apabila ingin menghapus soal ujian
9.	Menampilkan Soal Ujian	Kegiatan approval soal ujian yang dilakukan admin dan kemudian akan ditampilkan pada aplikasi mahasiswa.
10.	Melihat Data Mahasiswa	Kegiatan yang dilakukan admin untuk melihat data mahasiswa
11.	Edit Data Mahasiswa	Kegiatan yang dilakukan admin apabila ingin mengedit data mahasiswa

12.	Tambah Data Mahasiswa	Kegiatan yang dilakukan admin apabila ingin menambah data mahasiswa
13.	Menghapus Data Mahasiswa	Kegiatan yang dilakukan admin apabila ingin menghapus data mahasiswa
14.	Melihat Data Dosen	Kegiatan yang dilakukan admin untuk melihat halaman menu data dosen
15.	Edit Data Dosen	Kegiatan yang dilakukan admin apabila ingin mengedit data dosen
16.	Tambah Data Dosen	Kegiatan yang dilakukan admin apabila ingin menambah data dosen
17.	Menghapus Data Dosen	Kegiatan yang dilakukan admin apabila ingin menghapus data dosen
18.	Melihat Pengumuman	Kegiatan yang dilakukan admin untuk melihat halaman menu pengumuman
19.	Edit Pengumuman	Kegiatan yang dilakukan admin untuk mengedit pengumuman
20.	Tambah Pengumuman	Kegiatan yang dilakukan admin untuk menambah pengumuman
21.	Menghapus Pengumuman	Kegiatan yang dilakukan admin untuk menghapus pengumuman
22.	Menampilkan Nilai	Kegiatan approval nilai yang dilakukan admin untuk kemudian ditampilkan aplikasi mahasiswa.

4.3.2 Alur Kerja Aktor Terhadap Sistem (*Activity Diagram)*

4.3.2.1 *Activity Diagram* Mahasiswa Terhadap Sistem

a. *Activity Diagram Login*

Gambar 4.5 *Activity* diagram *login*

Gambar 4.5 adalah *activity diagram login* dimana tahap pertama yang harus dilakukan oleh mahasiswa adalah dengan melakukan *login username* dan *password* yang *valid* terlebih dahulu. Jika *username* dan *password* yang dimasukkan salah maka *user* tidak bisa mengakses aplikasi. Sedangkan jika *username* dan *password* yang dimasukkan benar, maka akan menampilkan halaman pilihan. Selanjutnya mahasiswa dapat memilih menu ujian untuk mengakses menu-menu yang tersedia pada halaman utama ujian.

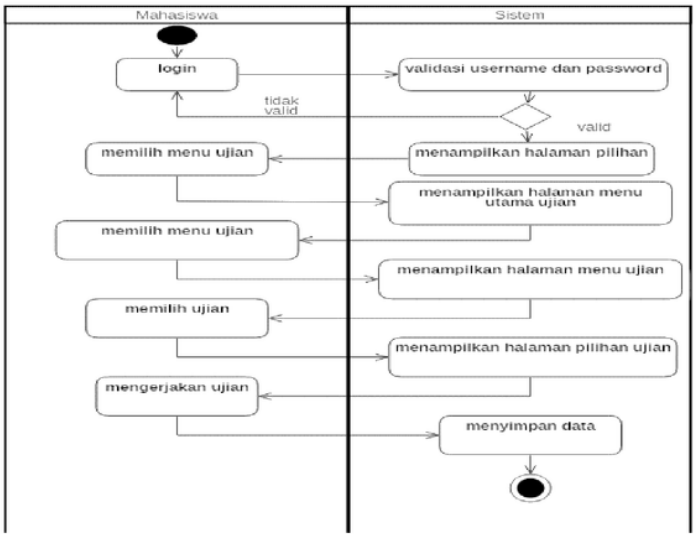
b. *Activity* Diagram Menu Profile

Gambar 4.6 *Activity* diagram menu *profile*

Gambar 4.6 adalah *activity diagram profile* dimana tahap pertama yang harus dilakukan oleh mahasiswa adalah dengan melakukan *login username* dan *password* yang *valid* terlebih dahulu. Jika *username* dan *password* yang dimasukkan salah maka *user* tidak bisa mengakses aplikasi. Sedangkan jika *username* dan *password* yang dimasukkan benar, maka akan menampilkan halaman

pilihan. Sedangkan jika *username* dan *password* yang dimasukkan benar, maka akan menampilkan halaman pilihan. Selanjutnya mahasiswa dapat memilih menu ujian untuk mengakses menu-menu yang tersedia pada halaman utama ujian. Kemudian mahasiswa memilih menu "*Profile*", pada menu *profile* mahasiswa dapat melakukan aktivitas mengedit *profile* dan mengganti *password*. Untuk mengedit *profile,* mahasiswa memilih tombol "Edit *Profile*" kemudian sistem akan menampilkan halaman edit *profile*. Pada halaman *profile* ini mahasiswa bisa mengganti *password* akun dengan mengklik tombol "Ganti *Password*" maka sistem akan menampilkan halaman ganti *password*. Mahasiswa bisa mengganti *password* dengan cara memasukkan *password* lama dan *password* baru, kemudian sistem akan menampilkan informasi *password* terganti

c. *Activity* Diagram Menu Ujian

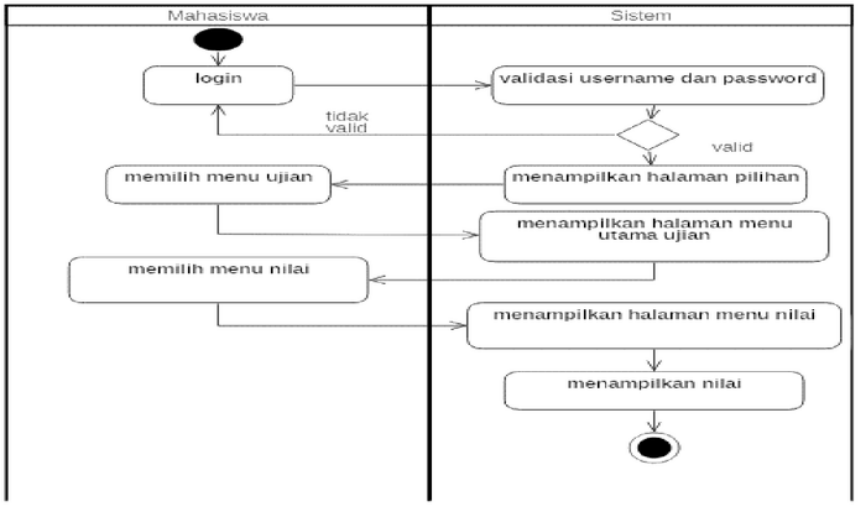


Gambar 4.7 *Activity* diagram menu ujian

Gambar 4.7 merupakan *activity diagram* ujian dimana tahap pertama yang harus dilakukan oleh mahasiswa adalah dengan melakukan *login username* dan *password* yang *valid* terlebih dahulu. Selanjutnya adalah masuk ke menu pilihan dan memilih menu "Ujian" maka sistem akan menampilkan halaman utama ujian

dan menampilkan halaman utama dengan menu-menu yang disediakan sistem aplikasi. Mahasiswa bisa memilih menu "Ujian" maka sistem akan menampilkan menu pilihan berupa ujian yang terjadwal seperti kuis, ujian tengah semester (uts), dan ujian akhir semester (uas). Lalu pilih "Kerjakan Tugas" maka sistem akan menampilkan soal ujian. Untuk pengerjaan ujian, masing-masing soal akan diberikan waktu pengerjaan yang berbeda tergantung dengan tingkat kesulitan soal ujian. Setelah menyelesaikan ujian mahasiswa dapat menyimpan hasil ujian dengan menekan tombol "Simpan". Maka jawaban akan tersimpan pada sistem

d. *Activity* Diagram Menu Nilai

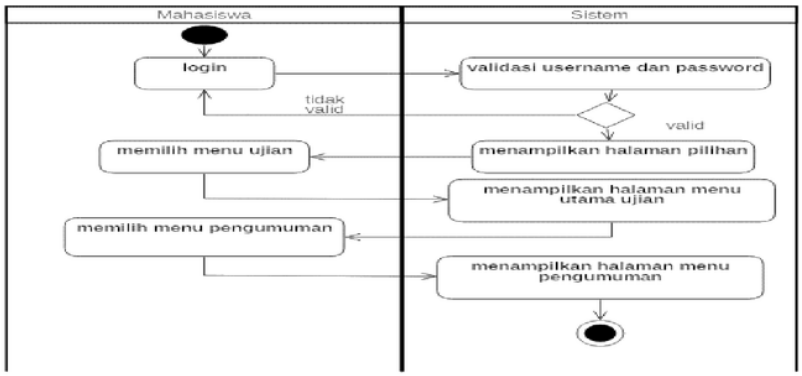


Gambar 4.8 *Activity* diagram menu nilai

Gambar 4.8 adalah *activity* diagram menu nilai dimana tahapan pertama yang harus dilakukan oleh mahasiswa adalah dengan melakukan *login username* dan *password* yang *valid* terlebih dahulu. Jika *username* dan *password* yang dimasukkan salah maka *user* tidak bisa mengakses aplikasi. Sedangkan jika *username* dan *password* yang dimasukkan benar, maka akan menampilkan halaman pilihan. Selanjutnya mahasiswa dapat memilih menu ujian untuk mengakses menu-menu yang tersedia pada halaman utama ujian. Mahasiswa dapat meihat hasil ujian

dengan memilih menu "Nilai" maka sistem akan menampilkan nilai-nilai hasil ujian.

e. *Activity* Diagram Menu Pengumuman



Gambar 4.9 *Activity* diagram menu pengumuman

Gambar 4.9 merupakan *activity* diagram menu pegumuman dimana tahapan pertama yang harus dilakukan oleh mahasiswa adalah dengan melakukan *login username* dan *password* yang *valid* terlebih dahulu. Jika *username* dan *password* yang dimasukkan salah maka *user* tidak bisa mengakses aplikasi. Sedangkan jika *username* dan *password* yang dimasukkan benar, maka akan menampilkan halaman pilihan. Selanjutnya mahasiswa dapat memilih menu ujian untuk mengakses menu-menu yang tersedia pada halaman utama ujian. Mahasiswa dapat melihat pengumuman jadwal ujian dan penguman lainnya dengan memilih menu "Pengumuman" maka sistem akan menampilkan nilai-nilai hasil ujian.

4.3.2.2 *Activity Diagram* Dosen Terhadap Sistem

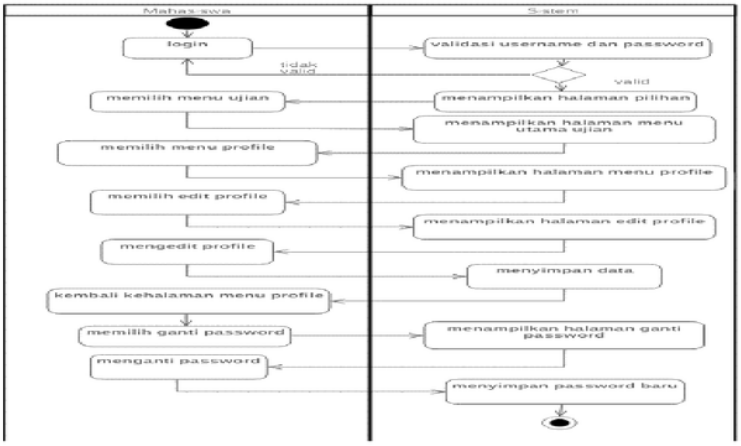
a. *Activity Diagram Login*

Gambar 4.10 *Activity* diagram *login*

Gambar 4.10 merupakan *activity diagram login* dimana tahapan pertama yang harus dilakukan oleh dosen adalah dengan melakukan *login username* dan *password* yang *valid* terlebih dahulu. Jika *username* dan *password* yang dimasukkan salah maka *user* tidak bisa mengakses aplikasi. Sedangkan jika *username* dan *password* yang dimasukkan benar, maka akan menampilkan halaman

pilihan. Selanjutnya mahasiswa dapat memilih menu ujian untuk mengakses menu-menu yang tersedia pada halaman utama ujian.

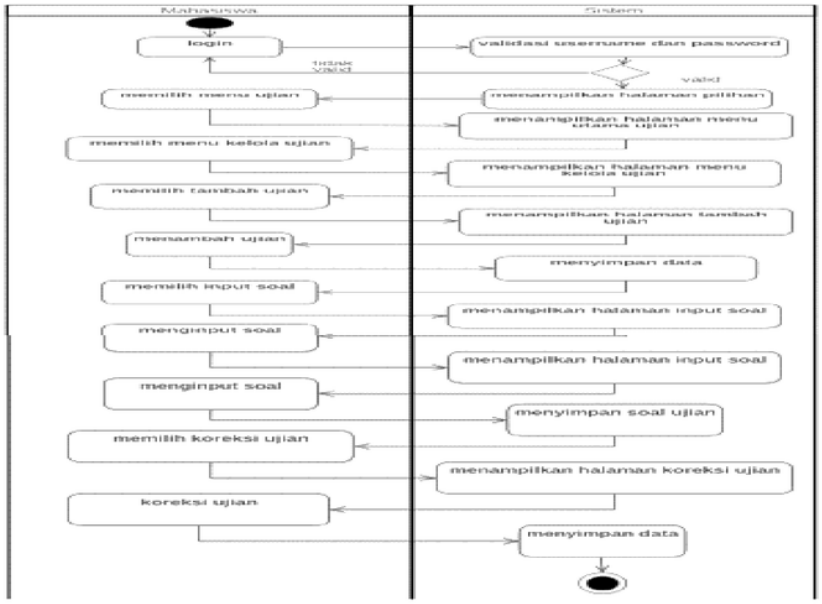
b. *Activity* Diagram Menu Profile



Gambar 4.11 *Activity* diagram menu *profile*

Gambar 4.11 merupakan *activity* diagram menu *profile* dimana tahapan pertama yang harus dilakukan oleh dosen adalah dengan melakukan *login username* dan *password* yang *valid* terlebih dahulu. Jika *username* dan *password* yang dimasukkan salah maka *user* tidak bisa mengakses aplikasi. Sedangkan jika *username* dan *password* yang dimasukkan benar, maka akan menampilkan halaman pilihan. Selanjutnya dosen dapat memilih menu ujian untuk mengakses menu-menu yang tersedia pada halaman utama ujian. Kemudian dosen memilih menu "*Profile*", pada menu *profile* dosen dapat melakukan aktivitas mengedit *profile* dan mengganti *password*. Untuk mengedit *profile*, dosen memilih tombol "Edit *Profile*" kemudian sistem akan menampilkan halaman edit *profile.* Pada halaman *profile* ini dosen bisa mengganti *password* akun dengan mengklik tombol "Ganti *Password*" maka sistem akan menampilkan halaman ganti *password.* dosen bisa mengganti *password* dengan cara memasukkan *password* lama dan *password* baru, kemudian sistem akan menampilkan informasi *password* terganti

c. *Activity* Diagram Menu Kelola Ujian

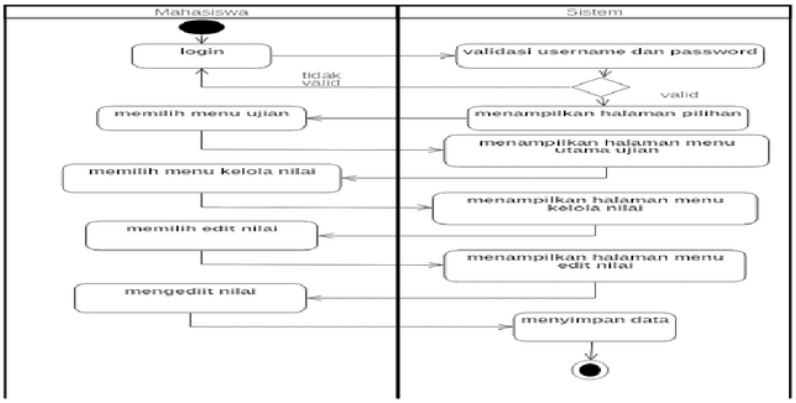


Gambar 4.12 *Activity* diagram menu kelola ujian

Gambar 4.12 merupakan *activity* diagram menu kelola ujian dimana tahapan pertama yang harus dilakukan oleh dosen adalah dengan melakukan *login username* dan *password* yang *valid* terlebih dahulu. Jika *username* dan *password* yang dimasukkan salah maka *user* tidak bisa mengakses aplikasi. Sedangkan jika *username* dan *password* yang dimasukkan benar, maka akan menampilkan halaman pilihan. Selanjutnya dosen dapat memilih menu ujian untuk mengakses menu-menu yang tersedia pada halaman utama ujian. Pada halaman menu ujian, dosen bisa melakukan aktivitas kelola ujian seperti, menambah ujian,menginput soal ujian, dan mengkoreksi soal ujian. Dosen dapat menambah mata kuliah yang akan diujikan dengan mengklik tombol "Tambah Ujian" maka sistem akan menampilkan halaman tambah ujian. Kemudian dosen bisa menginput soal ujian dengan mengklik "*Input* Soal" pada tabel aksi maka sistem akan menampilkan halaman input soal. Setelah menyelesaikan input soal ujian dosen dapat menyimpan soal ujian dengan menekan tombol "Simpan". Maka jawaban akan tersimpan pada sistem. Pada halaman menu ini dosen bisa mengkoreksi hasil ujian mahasiswa dengn mengklik "Koreksi" maka

halaman koreksi akan tampil. Pada halaman koreksi ujian ini soal dan jawaban dari mahasiswa akan ditampilkan, dosen dapat menilai langsung pada halaman tersebut, kemudian akan dihitung rata-rata secara otomatis oleh sistem.

d. *Activity* Diagram Menu Kelola Nilai

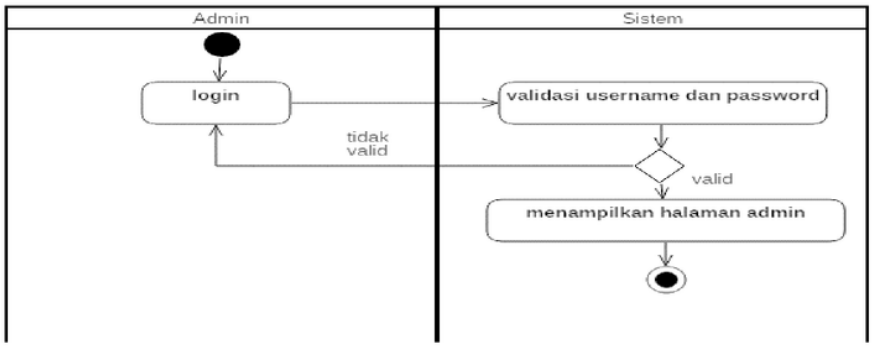


Gambar 4.13 Activity diagram menu kelola nilai

Gambar 4.13 merupakan *activity* diagram menu kelola nilai dimana tahapan pertama yang harus dilakukan oleh dosen adalah dengan melakukan *login username* dan *password* yang *valid* terlebih dahulu. Jika *username* dan *password* yang dimasukkan salah maka *user* tidak bisa mengakses aplikasi. Sedangkan jika *username* dan *password* yang dimasukkan benar, maka akan menampilkan halaman pilihan. Selanjutnya dosen dapat memilih menu ujian untuk mengakses menu-menu yang tersedia pada halaman utama ujian. Pada menu kelola nilai ini, dosen dapat mengedit nilai dari hasil koreksi nilai yang sudah dilakukan sebelumnya dengan cara mengklik tombol "Edit" kemudian dosen ddapat mengedit nilai ujian.

4.3.2.3 Activity Diagram Admin Terhadap Sistem

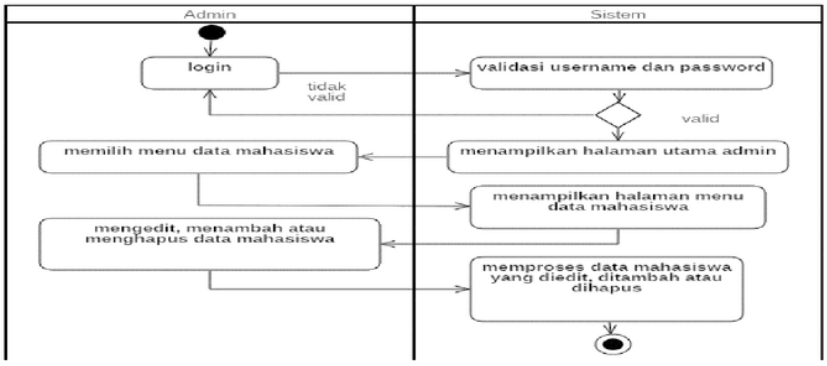
a. Activity Diaagram Login Admin



Gambar 4.14 Activity diagram login admin

Gambar 4.14 adalah *activity* diagram *login* dimana tahap pertama yang harus dilakukan oleh admin adalah dengan melakukan *login username* dan *password* yang *valid* terlebih dahulu. Jika *username* dan *password* yang dimasukkan salah maka admin tidak bisa mengakses aplikasi. Sedangkan jika *username* dan *password* yang dimasukkan benar, maka akan menampilkan halaman utama admin.

b. Activity Diagram Halaman Data Mahasiswa

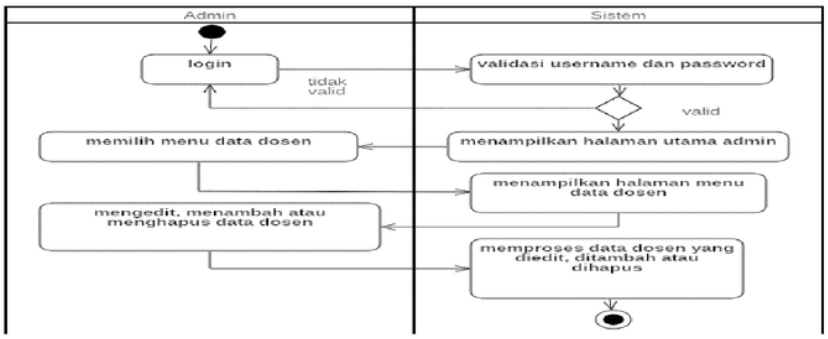


Gambar 4.15 Activity diagram menu data mahasiswa

Gambar 4.15 adalah *activity* diagram menu data mahasiswa dimana tahapan pertama yang harus dilakukan oleh admin adalah dengan melakukan *login username* dan *password* yang *valid* terlebih dahulu. Jika *username* dan *password*

yang dimasukkan salah maka admin tidak bisa mengakses aplikasi. Sedangkan jika *username* dan *password* yang dimasukkan benar, maka akan menampilkan halaman utama admin. Kemudian admin bisa memilih menu "Data Mahasiswa" maka sistem akan menampilkan halaman data mahasiswa. Pada halaman data mahasiswa admin dapat mengedit, menambah dan menghapus data mahasiswa.

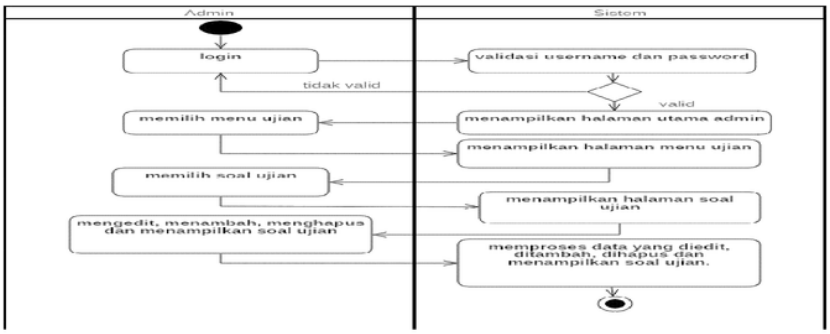
c. Activity Diagram Halaman Data Dosen



Gambar 4.16 Activity diagram menu data dcosen

Gambar 4.16 adalah *activity* diagram menu data dosen dimana tahapan pertama yang harus dilakukan oleh admin adalah dengan melakukan *login username* dan *password* yang *valid* terlebih dahulu. Jika *username* dan *password* yang dimasukkan salah maka admin tidak bisa mengakses aplikasi. Sedangkan jika *username* dan *password* yang dimasukkan benar, maka akan menampilkan halaman utama admin. Kemudian admin bisa memilih menu "Data Dosen" maka sistem akan menampilkan halaman data dosen. Pada halaman data mahasiswa admin dapat mengedit, menambah dan menghapus data dosen.

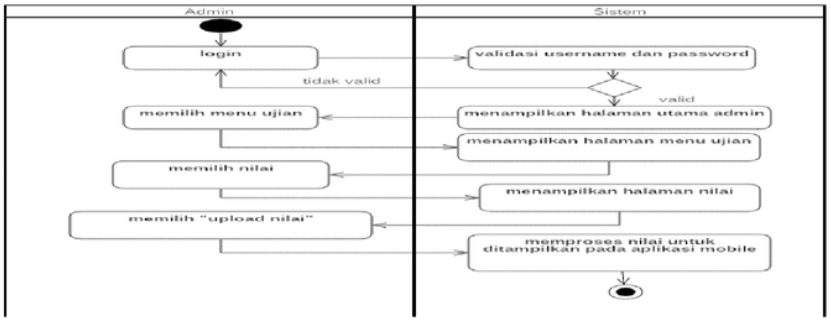
d. Activity Diagram Halaman Soal Ujian



Gambar 4.17 Activity diagram menu soal ujian

Gambar 4.17 adalah *activity* diagram halaman soal ujian dimana tahapan pertama yang harus dilakukan oleh admin adalah dengan melakukan *login username* dan *password* yang *valid* terlebih dahulu. Jika *username* dan *password* yang dimasukkan salah maka admin tidak bisa mengakses aplikasi. Sedangkan jika *username* dan *password* yang dimasukkan benar, maka akan menampilkan halaman utama admin. Kemudian admin dapat memilih menu "Ujian" lalu sistem akan menampilkan halaman ujian. Pada halaman ujian admin dapat mengklik "Soal Ujian" maka sistem akan menampilkan halaman soal ujian. Pada halaman soal ujian ini, admin bisa mengedit, menambah, menghapus dan menampilkan soal ujian.

f. Activity Diagram Halaman Nilai

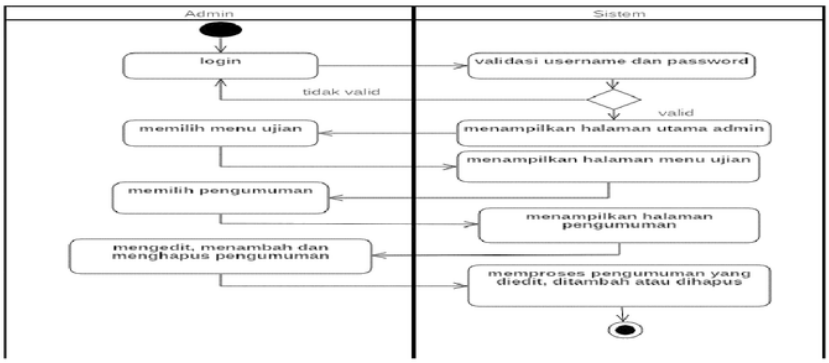


Gambar 4.18 Activity diagram menu nilai

Gambar 4.18 adalah *activity* diagram halaman nilai dimana tahapan pertama yang harus dilakukan oleh admin adalah dengan melakukan *login username* dan

password yang *valid* terlebih dahulu. Jika *username* dan *password* yang dimasukkan salah maka admin tidak bisa mengakses aplikasi. Sedangkan jika *username* dan *password* yang dimasukkan benar, maka akan menampilkan halaman utama admin. Kemudian admin dapat memilih menu "Ujian" lalu sistem akan menampilkan halaman ujian. Selanjutnya admin bisa mengklik "Nilai" maka sistem akan menampilkan halaman nilai. Pada halaman nilai admin dapat menampilkan nilai ujian dengan memilih tombol "Upload Nilai" maka nilai ujian akan tampil pada aplikasi android mahasiswa

g. *Activity Diagram* Menu Halaman Pengumuman



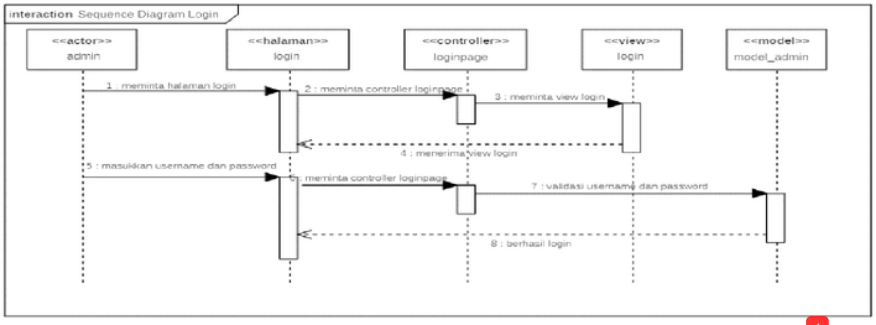
Gambar 4.19 Activity diagram menu halaman nilai

Gambar 4.19 adalah *activity* diagram Menu halaman pegumuman dimana tahapan pertama yang harus dilakukan oleh admin adalah dengan melakukan *login username* dan *password* yang *valid* terlebih dahulu. Jika *username* dan *password* yang dimasukkan salah maka admin tidak bisa mengakses aplikasi. Sedangkan jika *username* dan *password* yang dimasukkan benar, maka akan menampilkan halaman utama admin. Kemudian admin dapat memilih menu "Ujian" lalu sistem akan menampilkan halaman ujian. Selanjutnya admin bisa mengklik "Pengumuman" maka sistem akan menampilkan halaman nilai. Pada halaman pengumuman, admin dapat mengedit, menambah dan menghapus pengumuman.

4.3.3 Urutan Pengguna dalam Setiap Tindakan (*Sequence Diagram*)

4.3.3.1 Urutan Penggunaan Admin pada Sistem

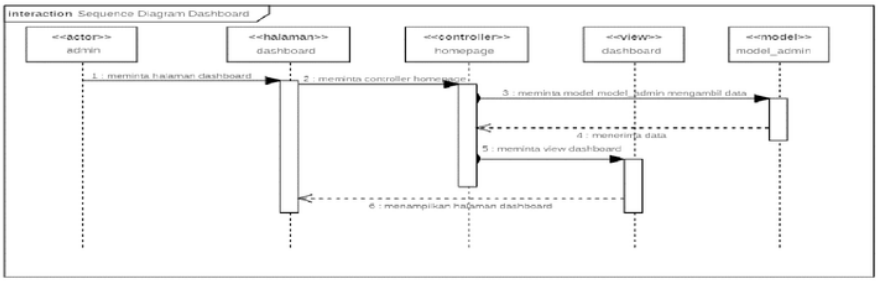
a. *Login* Admin



Gambar 4.20 Diagram urutan komunikasi antar objek untuk melihat halaman *login*

Pada gambar 4.20 *sequence* diagram diatas menggambarkan interaksi admin dalam melakukan *login* pada *web* admin. Admin login dengan menggunkan *browser* dengan cara memasukkan url *http://jteunsri.siatek.com/admin*. Halaman *login* tampil dengan memanggil *controller loginpage*. Pada *controller loginpage* terdapat proses pemanggilan *view login* untuk menampilkan halaman *login*. Setelah halaman *login* admin tampil. Admin harus *login* dengan memasukkan *username* dan *password* dan mengklik tombol *login*. Kemudian *controller loginpage* memanggil model model_admin untuk mengambil dan validasi data admin, jika data benar dan sesuai dengan *data base* maka admin berhasil *login*.

b. Masuk Halaman Utama

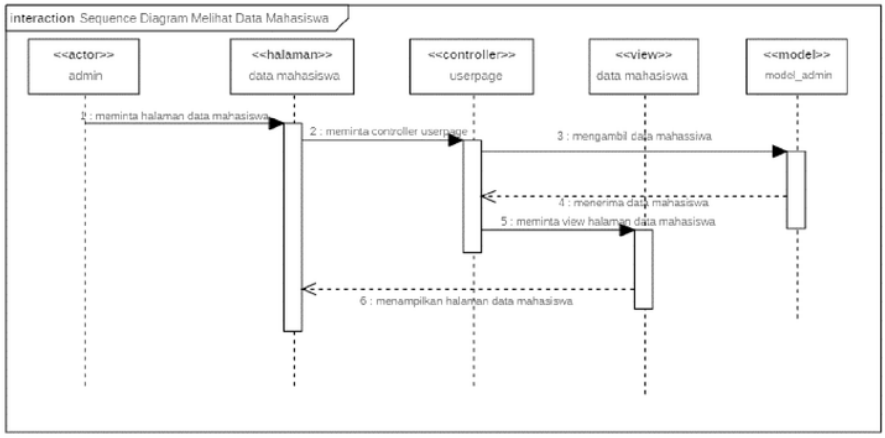


Gambar 4.21 Diagram urutan komunikasi antar objek untuk masuk halaman utama admin

Gambar 4.21 diatas menunjukkan *sequence* diagram interaksi admin dalam proses melihat halaman *dashboard* pada *web* admin. Halaman *dashboard* akan

tampil dengan memanggil *controller homepage*. Didalam *controller homepage* admin melakukan pengambilan data dengan memanggil model model_admin. Kemudian setelah data diterima, *controller* meminta *view dashboard* untuk menampilkan halaman *dashboard* admin.

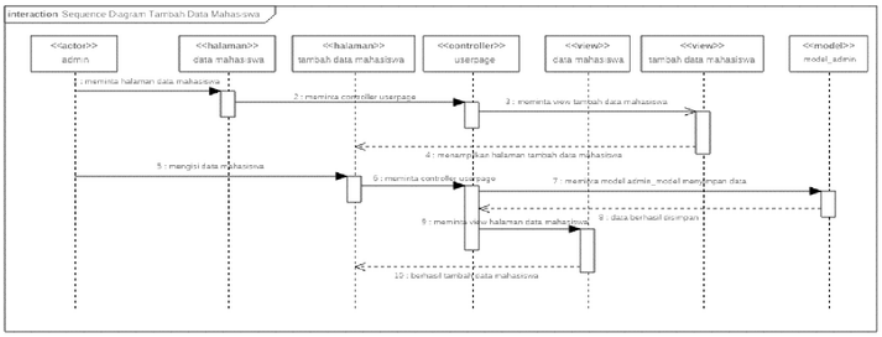
c. Melihat Data Mahasiswa



Gambar 4.22 Diagram urutan komunikasi antar objek untuk halaman melihat data mahasiswa

Gambar 4.22 diatas adalah *sequence* diagram interaksi admin yang menggambarkan prses menampilkan halaman melihat data mahasiswa. Halaman data mahasiswa akan muncul dengan memanggil *controller userpage*. Pada *controller userpage* akan memanggil model model_admin untuk mengambil data mahasiswa pada *database*. Setelah menerima data mahasiswa, *controller userpage* akan meminta *view* data mahasiswa. Lalu Admin masuk ke halaman data mahasiswa yang berisi data-data mahasiswa.

d. Menambah Data Mahasiswa



Gambar 4.23 Diagram urutan komunikasi antar objek untuk tambah data mahasiswa

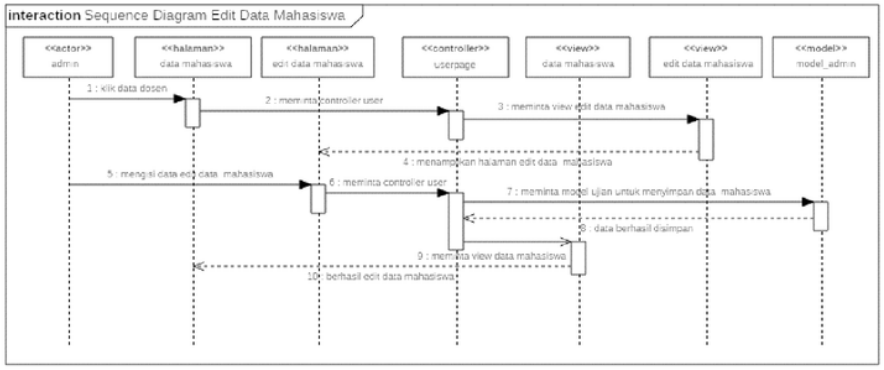
Gambar 4.23 diatas adalah *sequence* diagram interaksi admin yang menggambarkan prses menampilkan proses tambah data mahasiswa. Admin masuk ke halaman data mahasiswa kemudian pilih tambah data siswa. Halaman data masiswa kemudian melakukan proses pemanggilan *controller userpage* untuk menampilkan halaman tambah data mahasiswa. Setelah Admin masuk ke halaman tambah data mahasiswa, Admin dapat menambahkan data mahasiswa kemudian klik simpan. Kemudian *controller userpage* akan meminta model model_admin untuk menyimpan data mahasiswa. Kemudian setelah tersimpan *controller userpage* meminta *view* data mahasiswa untuk menampilkan data mahasiswa dan menampilkan keterangan berhasil menambah data mahasiswa.

e. Menghapus Data Mahasiswa

Gambar 4.24 Diagram urutan komunikasi antar objek untuk hapus data mahasiswa

Gambar 4.24 diatas adalah *sequence* diagram interaksi admin yang menggambarkan proses hapus data mahasiswa. Admin masuk ke halaman data mahasiswa kemudian mengklik tombol *delete*. Kemudian *controller userpage* akan meminta model model_admin untuk menghapus data mahasiswa dari *database*. Kemudian setelah itu *controller userpage* meminta *view* data mahasiswa dan menampilkan keterangan data mahasiswa berhasil dihapus.

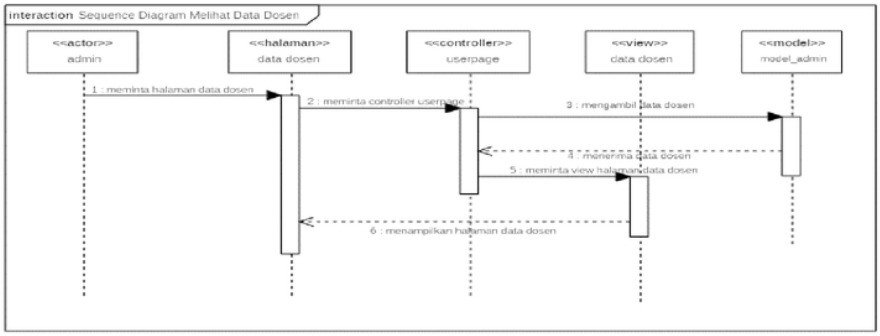
f. Mengedit Data Mahasiswa



Gambar 4.25 Diagram urutan komunikasi antar objek untuk halaman edit data mahasiswa

Gambar 4.25 diatas adalah *sequence* diagram interaksi admin yang menggambarkan prses menampilkan proses edit data mahasiswa. Admin masuk ke halaman data mahasiswa kemudian pilih edit data siswa. Halaman data masiswa kemudian melakukan proses pemanggilan *controller userpage* untuk menampilkan halaman edit data mahasiswa. Setelah Admin masuk ke halaman edit data mahasiswa, Admin dapat mengedit data mahasiswa. Kemudian *controller userpage* akan meminta model model_admin untuk menyimpan data mahasiswa. Kemudian setelah tersimpan *controller userpage* meminta *view* data mahasiswa untuk menampilkan data mahasiswa dan menampilkan keterangan berhasil mengedit data mahasiswa.

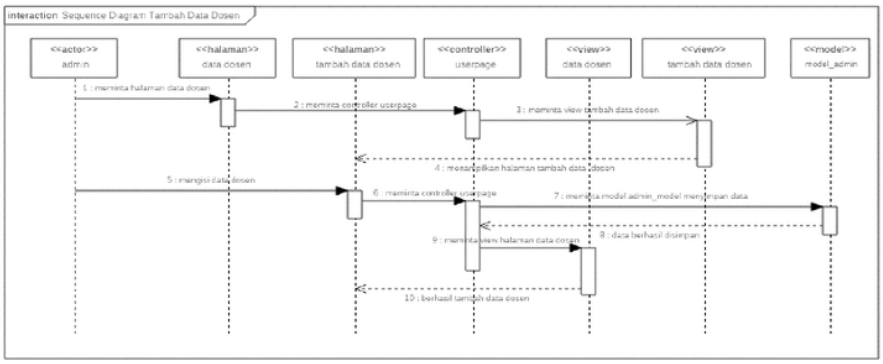
g. Melihat Data Dosen



Gambar 4.26 Diagram urutan komunikasi antar objek untuk halaman melihat data dosen

Gambar 4.26 diatas adalah *sequence* diagram interaksi admin yang menggambarkan prses menampilkan halaman melihat data dosen. Halaman data dosen akan muncul dengan memanggil *controller userpage*. Pada *controller userpage* akan memanggil model model_admin untuk mengambil data dosen pada *database*. Setelah menerima data dosen, *controller userpage* akan meminta *view* data dsoen. Lalu Admin masuk ke halaman data dosen yang berisi data-data dosen.

h. Menambah Data Dosen

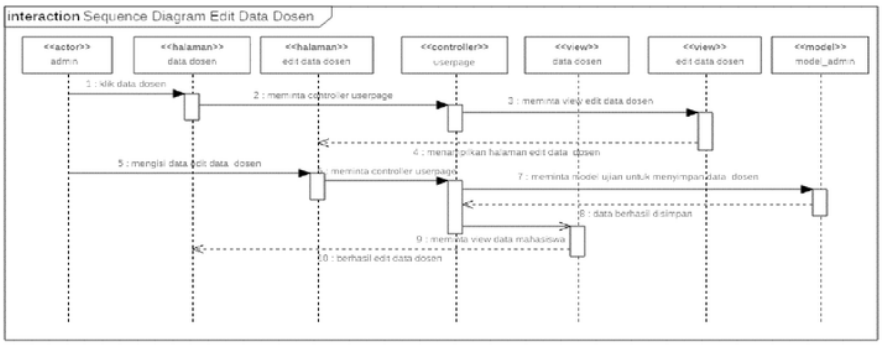


Gambar 4.27 Diagram urutan komunikasi antar objek untuk halaman tambah data dosen

Gambar 4.27 diatas adalah *sequence* diagram interaksi admin yang menggambarkan prses menampilkan proses tambah data dosen. Admin masuk ke halaman data dosen kemudian pilih tambah data dosen. Halaman data dosen

kemudian melakukan proses pemanggilan *controller userpage* untuk menampilkan halaman tambah data dosen. Setelah Admin masuk ke halaman tambah data dosen, Admin dapat menambahkan data dosen kemudian klik simpan. Kemudian *controller userpage* akan meminta model model_admin untuk menyimpan data dosen. Kemudian setelah tersimpan *controller userpage* meminta *view* data dosen untuk menampilkan data dosen dan menampilkan keterangan berhasil menambah data dosen.

i. Mengedit Data Dosen



Gambar 4.28 Diagram urutan komunikasi antar objek untuk halaman edit data dosen

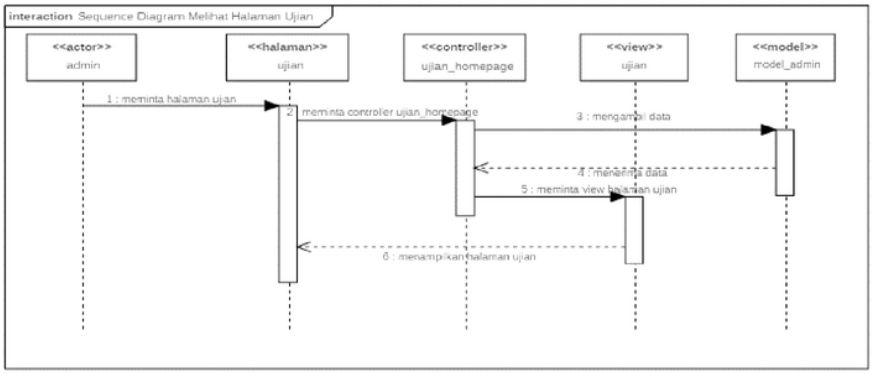
Gambar 4.28 diatas adalah *sequence* diagram interaksi admin yang menggambarkan prses menampilkan proses edit data dosen. Admin masuk ke halaman data mahasiswa kemudian pilih edit data dosen. Halaman data dosen kemudian melakukan proses pemanggilan *controller userpage* untuk menampilkan halaman edit data dosen. Setelah Admin masuk ke halaman edit data mahasiswa, Admin dapat mengedit data mahasiswa. Kemudian *controller userpage* akan meminta model model_admin untuk menyimpan data dosen Kemudian setelah tersimpan *controller userpage* meminta *view* data data untuk menampilkan data dosen dan menampilkan keterangan berhasil mengedit data dosen.

j. Menghapus Data Dosen

Gambar 4.29 Diagram urutan komunikasi antar objek untuk hapus data dosen

Gambar 4.29 diatas adalah *sequence* diagram interaksi admin yang menggambarkan proses menghapus data dosen. Admin masuk ke halaman data dosen kemudian mengklik tombol *delete*. Kemudian *controller userpage* akan meminta model model_admin untuk menghapus data dosen dari *database*. Kemudian setelah itu *controller userpage* meminta *view* data dosen dan menampilkan keterangan data dosen berhasil dihapus.

k. Melihat Soal Ujian

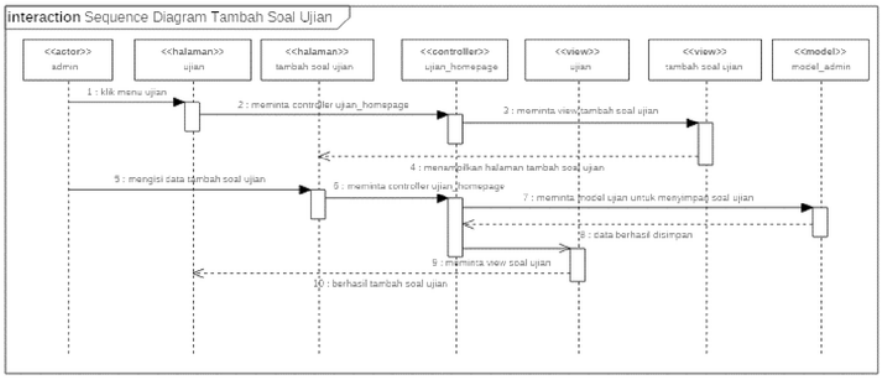


Gambar 4.30 Diagram urutan komunikasi antar objek untuk melihat halaman soal ujian

Gambar 4.30 diatas adalah *sequence* diagram interaksi admin yang menggambarkan prses menampilkan halaman ujian. Halaman ujian akan muncul

dengan memanggil *controller* ujian_homepage. Pada *controller* ujian_homepage akan memanggil model model_admin untuk mengambil data ujian pada *database*. Setelah menerima data, *controller* ujian_homepage akan meminta *view* halaman ujian.

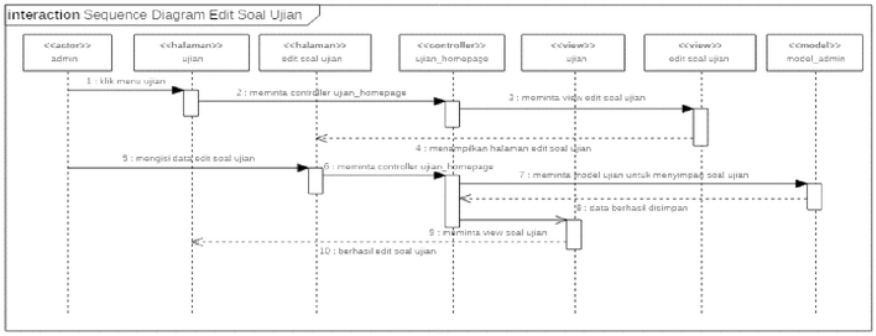
l. Menambah Soal Ujian



Gambar 4.31 Diagram urutan komunikasi antar objek untuk halaman tambah soal ujian

Gambar 4.31 diatas adalah *sequence* diagram interaksi admin yang menggambarkan proses menampilkan proses tambah soal ujian. Admin masuk ke halaman ujian kemudian pilih tambah soal ujian. Halaman data dosen kemudian melakukan proses pemanggilan *controller* ujian_homepage untuk menampilkan halaman tambah soal ujian. Setelah Admin masuk ke halaman tambah soal ujian, Admin dapat menambahkan soal ujian lalu klik simpan. Kemudian *controller* ujian_homepage akan meminta model model_admin untuk menyimpan soal. Kemudian setelah tersimpan *controller* ujian_homepage meminta *view* soal ujian untuk menampilkan soal ujian dan menampilkan keterangan berhasil menambah soal ujian.

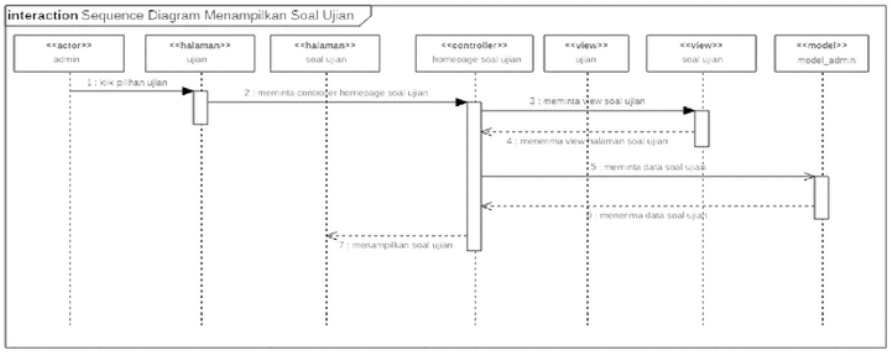
m. Mengedit Soal Ujian



Gambar 4.32 Diagram urutan komunikasi antar objek untuk halaman edit soal ujian

Gambar 4.32 diatas adalah *sequence* diagram interaksi admin yang menggambarkan proses menampilkan proses edit data dosen. Admin masuk ke halaman data mahasiswa kemudian pilih edit data dosen. Halaman data dosen kemudian melakukan proses pemanggilan *controller userpage* untuk menampilkan halaman edit data dosen. Setelah Admin masuk ke halaman edit data mahasiswa, Admin dapat mengedit data mahasiswa. Kemudian *controller userpage* akan meminta model model_admin untuk menyimpan data dosen Kemudian setelah tersimpan *controller userpage* meminta *view* data data untuk menampilkan data dosen dan menampilkan keterangan berhasil mengedit data dosen.

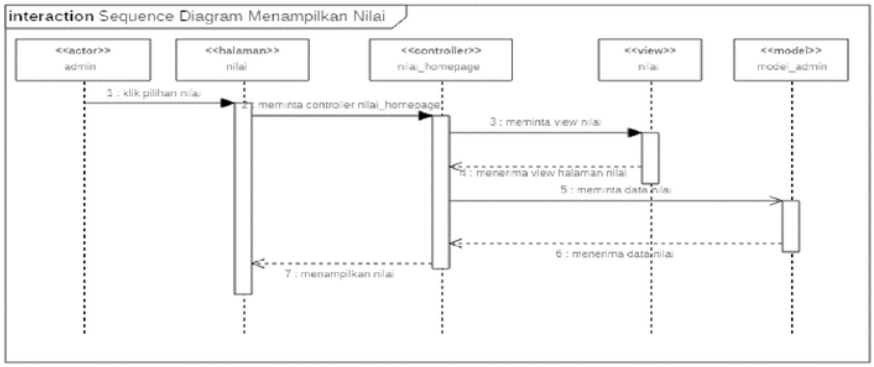
n. Menampilkan Soal Ujian



Gambar 4.33 Diagram urutan komunikasi antar objek untuk halaman menampilkan soal ujian

Gambar 4.33 diatas adalah sequence diagram interaksi admin yang menggambarkan prses menampilkan soal ujian. Halaman soal ujian akan muncul dengan memanggil controller homepage_soalujian. Pada controller homepage_soalujian akan memanggil view soal ujian. Selanjutnya controller homepage_soalujian meminta model model_admin untuk mengambil data soal ujian pada data base. Setelah menerima data, controller homepage_soal ujian akan menampilkan halaman soal ujian.

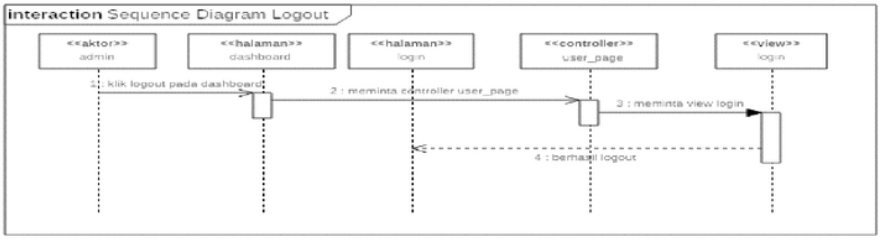
o. Menampilkan Nilai



Gambar 4.34 Diagram urutan komunikasi antar objek untuk halaman menampilkan nilai

Gambar 4.34 diatas adalah sequence diagram interaksi admin yang menggambarkan prses menampilkan nilai ujian. Halaman nilai akan muncul dengan memanggil controller nilai_homepage. Pada controller nilai_homepage memanggil view nilai. Selanjutnya controller nilai_homepage meminta model model_admin untuk mengambil data nilai pada data base. Setelah menerima data, controller nilai_homepage akan menampilkan halaman nilai ujian.

p. *Logout* Admin

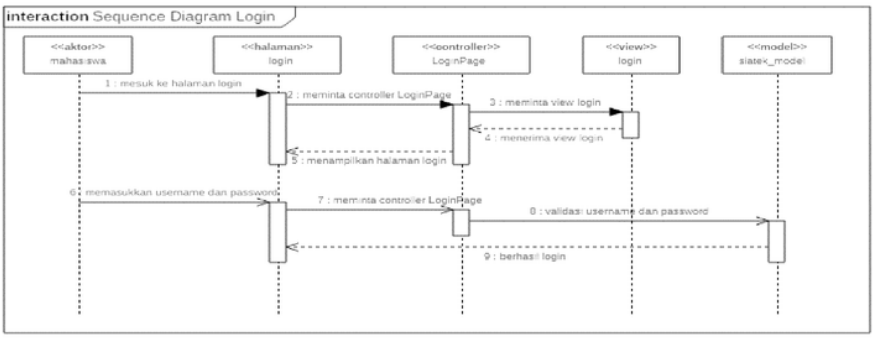


Gambar 4.35 Diagram urutan komunikasi antar objek untuk halaman *logout* admin

Gambar 4.35 *sequence diagram* diatas menggambarkan interaksi antar objek dalam melakukan proses logout dari web admin. Admin klik logout pada halaman dashboard web admin kemudian proses selanjutnya adalah *controller* user_page meminta view login untuk menampilkan halaman login.

4.3.3.2 Urutan Penggunaan User (Mahasiswa) pada Sistem

a. Login Mahasiswa

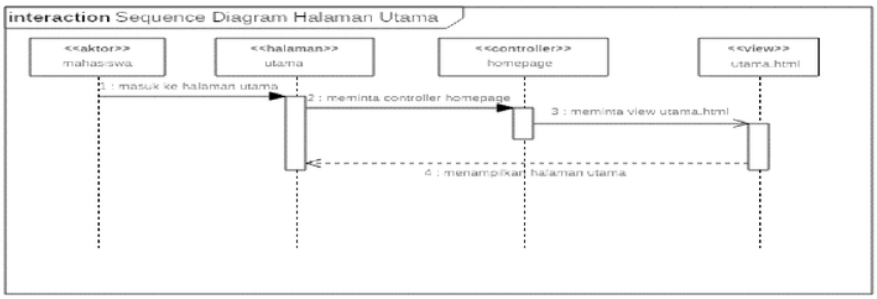


Gambar 4.36 Diagram urutan komunikasi antar objek untuk halaman *login* mahasiswa

Gambar 4.36 *sequence* diagram diatas menggambarkan interaksi mahasiswa dalam melakukan *login* pada perangkat android. Mahasiswa login dengan menggunkan aplikasi SIATEK UNSRI (Sistem Informasi Teknik Elektro Universitas Sriwijaya) yang telah didownload di Google Play. Halaman *login* tampil dengan memanggil *controller LoginPage*. Pada *controller LoginPage*

terdapat proses pemanggilan *view login* untuk menampilkan halaman *login*. Setelah halaman login mahasisa tampil. Mahasiswa harus *login* dengan memasukkan *username* dan *password* dan mengklik tombol login. Kemudian *controller LoginPage* memanggil model siatek_model untuk mengambil dan validasi data admin, jika data benar dan sesuai dengan *data base* maka mahasiswa berhasil *login*.

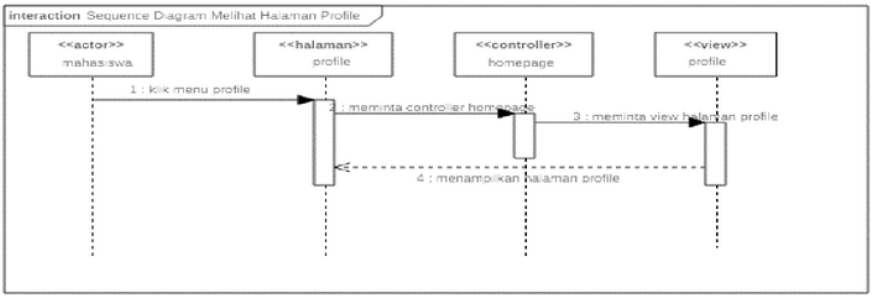
b. Masuk Halaman Utama Mahasiswa



Gambar 4.37 Diagram urutan komunikasi antar objek untuk halaman utama

Gambar 4.37 diatas menunjukkan *sequence* diagram interaksi objek dalam melakukan proses melihat halaman utama mahasiswa pada aplikasi. Halaman utama mahasiswa akan tampil dengan memanggil *controller homepage*. Kemudian *controller* meminta *view* utama.html untuk menampilkan halaman utama mahasiswa.

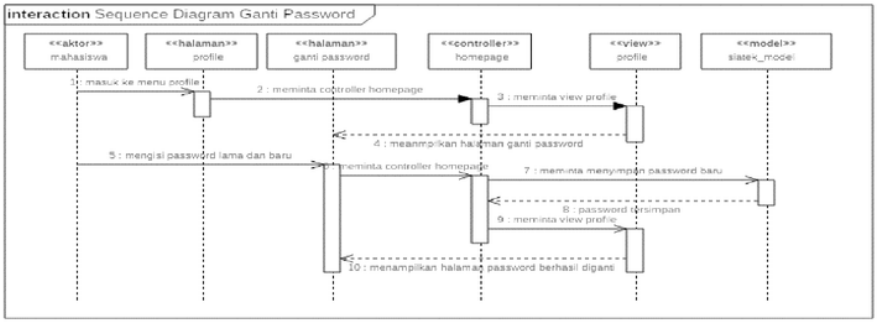
c. Melihat Menu Profile



Gambar 4.38 Diagram urutan komunikasi antar objek untuk halaman menu profile

Gambar 4.38 diatas menjelaskan interaksi antar objek dalam proses melihat halaman menu profile mahasiswa pada aplikasi. Mahasiswa masuk ke halaman profile dengan klik menu profile pada halaman utama. Halaman profile muncul melalui proses pemanggilan *controller homepage*. Didalam *controller homepage* terdapat proses pemanggilan *view* profile untuk menampilkan halaman profile, maka halaman menu profile akan tampil.

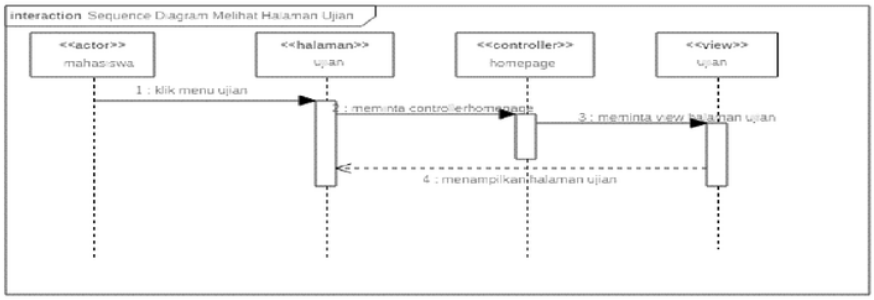
d. Mengganti Password



Gambar 4.39 Diagram urutan komunikasi antar objek untuk ganti password

Gambar 4.39 diatas menunjukkan sequence diagram interaksi antar objek dalam melakukan proses mengganti *password* pada aplikasi. Mahasiswa masuk ke halaman ganti *password* dengan cara masuk ke menu profile dan klik tombol ganti *password* pada halaman profile. Halaman ganti *password* muncul melalui proses pemanggilan *controller homepage*. Didalam *controller homepage* terdapat proses pemanggilan *view* profile untuk menampilkan halaman ganti *password*. Setelah muncul halaman ganti *password*, mahasiswa dapat mengisi data *password* lama dan *password* baru. Selanjutnya *controller homepage* meminta *model* siatek_model untuk menyimpan data *password* baru pada *database*. Kemudian *controller homepage* meminta *view* profile untuk menampilkan hasil bahwa *password* berhasil diganti.

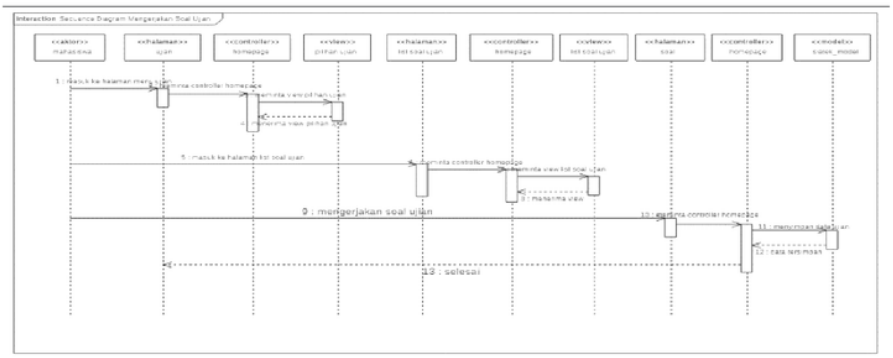
e. Melihat Halaman Menu Ujian



Gamabr 4.40 Diagram urutan komunikasi antar objek untuk melihat halaman menu ujian

Gambar 4.40 diatas menunjukkan sequence diagram interaksi antar objek dalam melakukan proses melihat halaman ujian mahasiswa pada aplikasi. Mahasiswa masuk ke halaman ujian dengan klik menu ujian pada halaman utama. Halaman ujian muncul melalui proses pemanggilan *controller homepage*. Didalam *controller homepage* terdapat proses pemanggilan *view* ujian untuk menampilkan halaman ujian, maka halaman menu ujian akan tampil.

f. Mengerjakan Soal Ujian

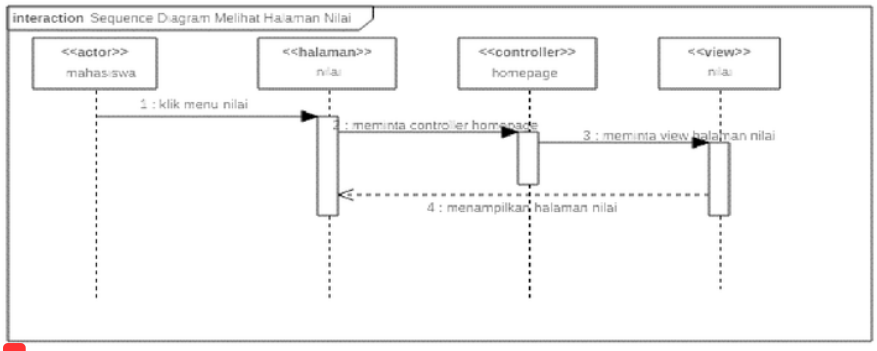


Gambar 4.41 Diagram urutan komunikasi antar objek untuk mengerjakan soal ujian

Gambar 4.41 diatas menunjukkan sequence diagrama interaksi antar objek dalam melakukan proses mengerjakan soal ujian bagi mahasiswa pada aplikasi. Mahasiswa masuk ke halaman menu ujian dengan cara masuk ke halaman utama

mahasiswa terlebih dahulu. Halaman ujian muncul melalui proses pemanggilan *controller homepage*. Didalam *controller homepage* terdapat proses pemanggilan *view* pilihan ujian untuk menampilkan halaman pilihan ujian (kuis, uts, dan uas). Setelah muncul halaman pilihan ujian, mahasiswa dapat memilih mata kuliah yang akan diuji pada list soal ujian melalui proses pemanggilan *controller homepage*. Didalam *controller homepage* terdapat proses pemanggilan *view* list soal ujian untuk menampilkan list soal ujian. Setelah memilih, mahasiswa dapat melakukan proses mengerjakan ujian sesuai dengan waktu yang telah ditentukan. Selanjutnya *controller homepage* meminta *model* siatek_model untuk menyimpan data jawaban pada *database*, dan aplikasi akan kembali ke halaman ujian.

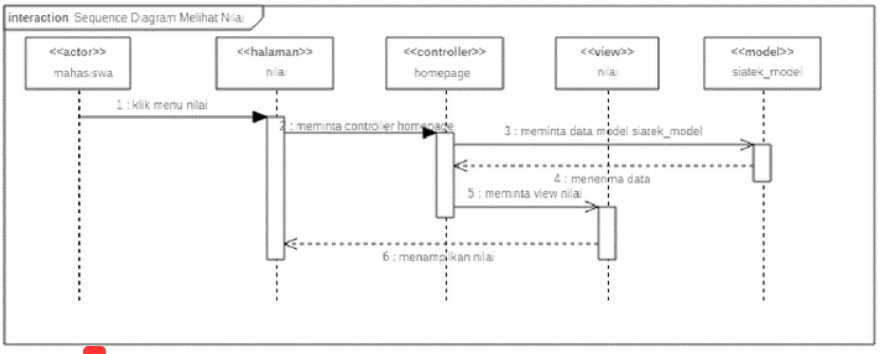
g. Melihat Halaman Menu Nilai



Gambar 4.42 Diagram urutan komunikasi antar objek untuk melihat halaman menu nilai

Gambar 4.42 diatas menunjukkan sequence diagram interaksi antar objek dalam melakukan proses melihat halaman menu nilai mahasiswa pada aplikasi. Mahasiswa masuk ke halaman ujian dengan klik menu ujian pada halaman utama. Halaman nilai muncul melalui proses pemanggilan *controller homepage*. Didalam *controller homepage* terdapat proses pemanggilan *view* nlai untuk menampilkan halaman nilai, maka halaman menu ujian akan tampil.

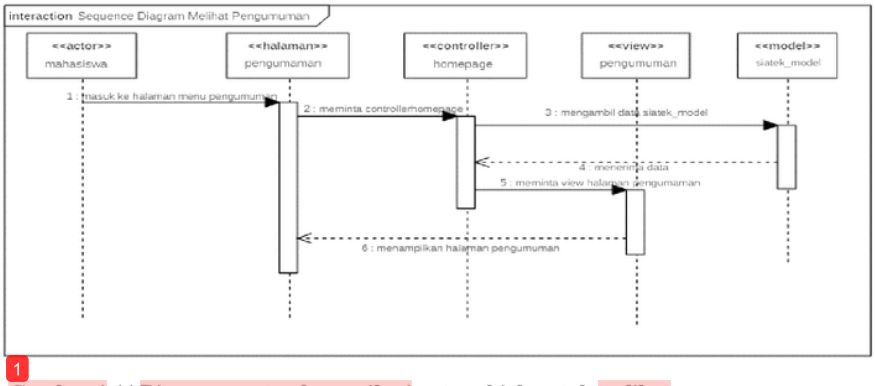
h. Melihat Nilai



Gambar 4.43 Diagram urutan komunikasi antar objek untuk melihat nilai

Gambar 4.43 diatas menunjukkan sequence diagram interaksi antar objek dalam melakukan proses melihat nilai mahasiswa pada aplikasi. Mahasiswa klik menu nilai pada halaman utama ujian. Halaman melihat nilai mahasiswa muncul melalui proses pemanggilan *controller homepage*. Didalam *controller homepage* terdapat proses pemanggilan model siatek_model untuk meminta data-data nilai mahasiswa pada database. Selanjutnya *controller homepage* meminta *view* nilai untuk menampilkan hasil nilai ujian mahasiswa.

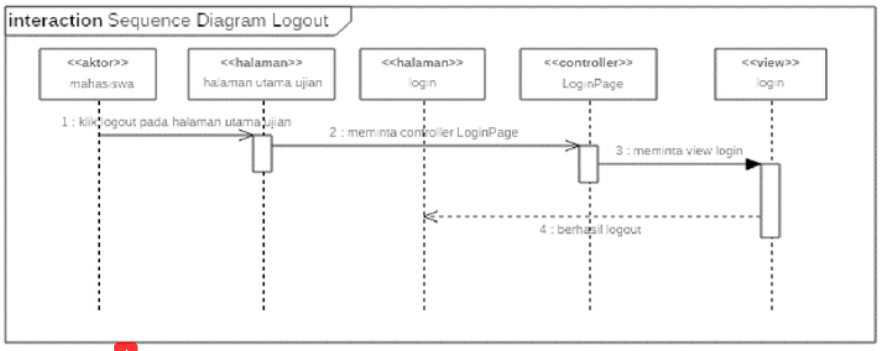
i. Melihat Halaman Pengumuman



Gambar 4.44 Diagram urutan komunikasi antar objek untuk melihat menu pengumaman

Gambar 4.44 diatas menunjukkan sequence diagram interaksi antar objek dalam melakukan proses melihat pengumaman bagi mahasiswa pada aplikasi. Mahasiswa masuk ke halaman menu pengumuman. Halaman melihat pengumuman muncul melalui proses pemanggilan *controller homepage*. Didalam *controller homepage* terdapat proses pemanggilan model siatek_model untuk meminta data pada database. Selanjutnya *controller homepage* meminta *view* pengumuman untuk menampilkan halaman menu pengumuman.

j. Logout Mahasiswa

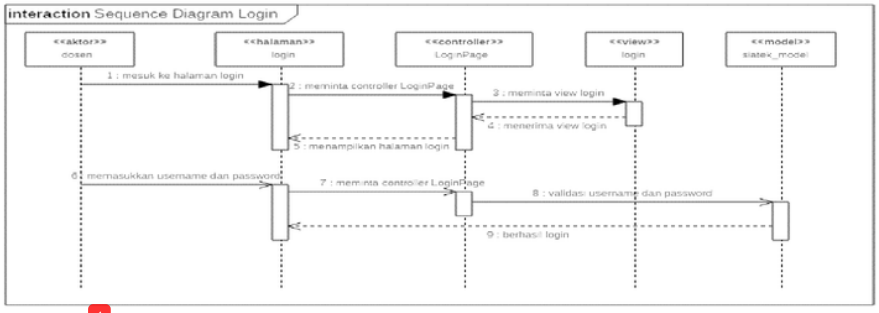


Gambar 4.45 Diagram urutan komunikasi antar objek untuk logout

Gambar 4.45 *sequence diagram* diatas menggambarkan interaksi antar objek dalam melakukan proses logout mahasiswa pada aplikasi. Mahasiswa klik tombol *logout* pada halaman utama ujian di aplikasi. Halaman utama ujian melalui proses pemanggilan *controller LoginPage*. Didalam *controller LoginPage* terdapat proses pemanggilan *view login* maka mahasiswa berhasil *logout* dari aplikasi.

4.3.3.3 Urutan Penggunaan User (Dosen) pada Sistem

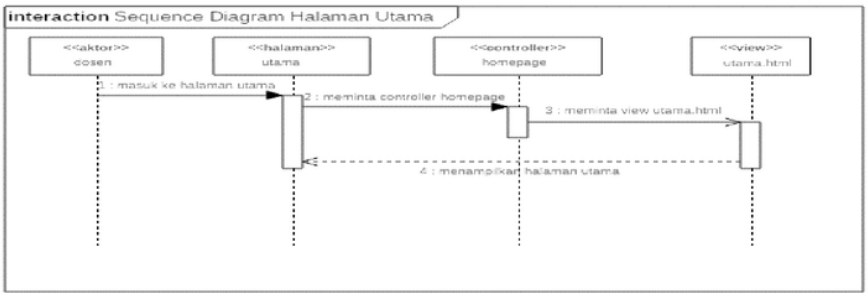
a. *Login* Dosen



Gambar 4.46 Diagram urutan komunikasi antar objek untuk login dosen

Gambar 4.46 *sequence* diagram diatas menggambarkan interaksi dosen dalam melakukan *login* pada *web*. Dosen login dengan menggunkan *browser* dengan cara memasukkan url url *http://siatek.jteundro.com* Halaman *login* tampil dengan memanggil *controller LoginPage*. Pada *controller LoginPage* terdapat proses pemanggilan *view login* untuk menampilkan halaman *login*. Setelah halaman login dosen tampil. Dosen harus *login* dengan memasukkan *username* dan *password* dan mengklik tombol login. Kemudian *controller LoginPage* memanggil model siatek_model untuk mengambil dan validasi username dan password, jika data benar dan sesuai dengan *data base* maka dosen berhasil *login*.

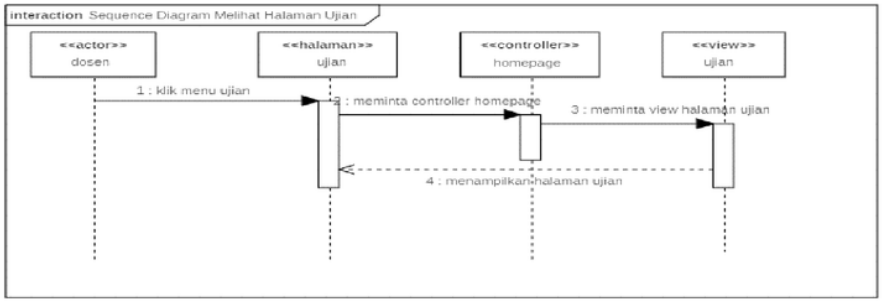
b. Masuk Halaman Utama



Gambar 4.47 Diagram urutan komunikasi antar objek untuk masuk halaman utama dosen

Gambar 4.47 diatas menunjukkan *sequence* diagram interaksi objek dalam melakukan proses melihat halaman utama dosen pada web. Halaman utama dosen akan tampil dengan memanggil *controller homepage*. Kemudian *controller* homepage meminta *view* utama.html untuk menampilkan halaman utama dosen.

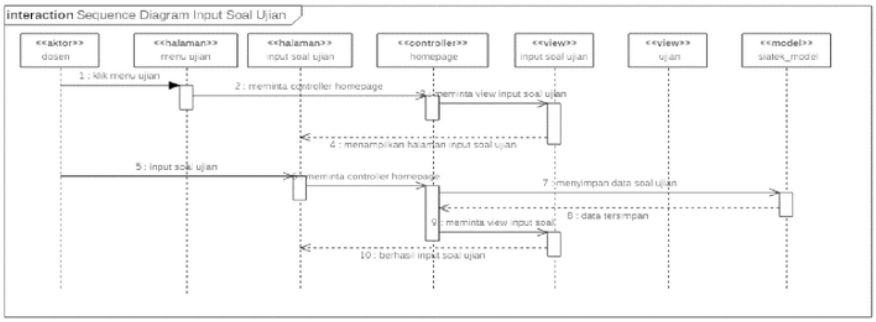
c. Masuk Menu Ujian



Gambar 4.48 Diagram urutan komunikasi antar objek untuk masuk menu ujian

Gambar 4.48 diatas menunjukkan sequence diagram interaksi antar objek dalam melakukan proses melihat halaman menu ujian pada web. Mahasiswa masuk ke halaman ujian dengan klik menu ujian pada halaman utama. Halaman menu ujian muncul melalui proses pemanggilan *controller homepage*. Didalam *controller homepage* terdapat proses pemanggilan *view* ujian untuk menampilkan halaman ujian, maka halaman menu ujian akan tampil

d. Meng*input Soal* Ujian



Gambar 4.49 Diagram urutan komunikasi antar objek untuk input soal ujian

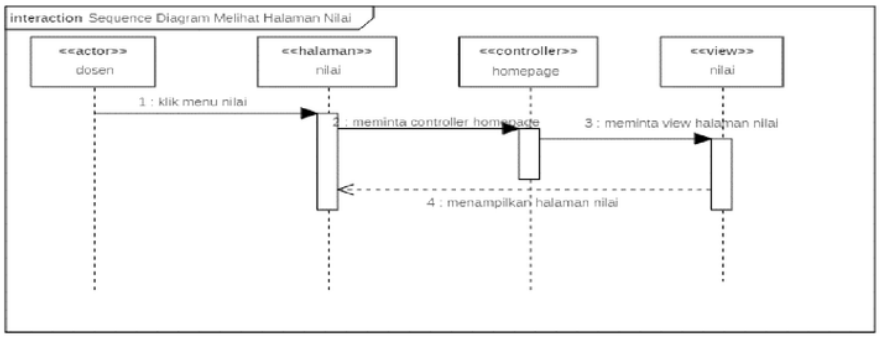
Gambar 4.49 diatas menunjukkan *sequence* diagram interaksi antar objek dalam melakukan proses input soal ujian. Dosen klik menu ujian, kemudian halaman halaman menu ujian memanggil controller homepage. Didalam controller homepage terdapt proses pemanggilan view input soal ujian untuk menampilkan halaman input soal ujian. Setelah muncul halaman input soal ujian, dosen dapat menginputkan soal ujian sesuai mata kuliah yang diajar. Selanjutnya controller homepage meminta model siatek_model untuk menyimpan data soal ujian pada database. Kemudian controller homepage meminta view input soal ujian untuk menampilkan informasi bawha soal ujian berhasil diinput.

e. Mengkoreksi Soal Ujian

Gambar 4.50 Diagram urutan komunikasi antar objek untuk koreksi soal ujian

Gambar 4.50 diatas menunjukkan sequence diagram interaksi antar objek dalam melakukan proses koreksi soal ujian. Dosen klik menu ujian, kemudian halaman halaman menu ujian memanggil controller homepage. Didalam controller homepage terdapt proses pemanggilan view koreksi untuk menampilkan halaman koreksi. Setelah muncul halaman koreksi, dosen dapat mengkoreksi hasil ujian mahasiswa. Selanjutnya controller homepage meminta model siatek_model untuk menyimpan data koreksi nilai pada database dan mengkalkulasi hasil ujian. Kemudian controller homepage meminta view koreksi untuk menampilkan informasi bawha hasil ujian berhasil dikoreksi.

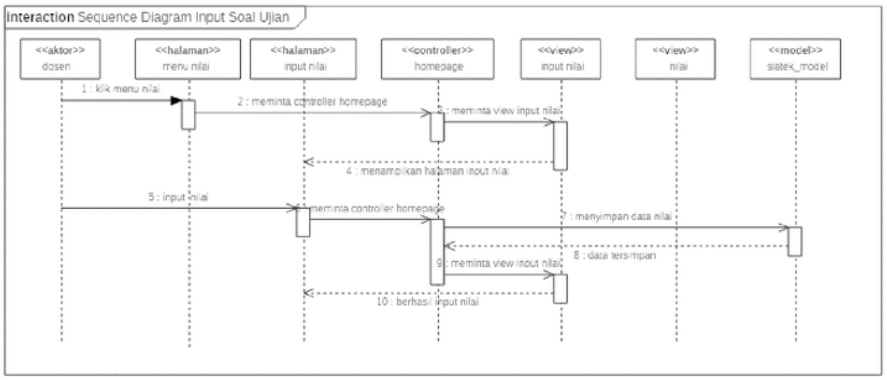
f. Masuk Menu Nilai



Gambar 4.51 Diagram urutan komunikasi antar objek untuk masuk menu nilai

Gambar 4.51 diatas menunjukkan sequence diagram interaksi antar objek dalam melakukan proses melihat halaman menu nilai pada web. Mahasiswa masuk ke halaman utama ujian dengan klik menu nilai. Halaman menu nilai muncul melalui proses pemanggilan *controller homepage*. Didalam *controller homepage* terdapat proses pemanggilan *view* nilai untuk menampilkan halaman nilai, maka halaman menu ujian akan tampil

g. *Menginput Nilai*

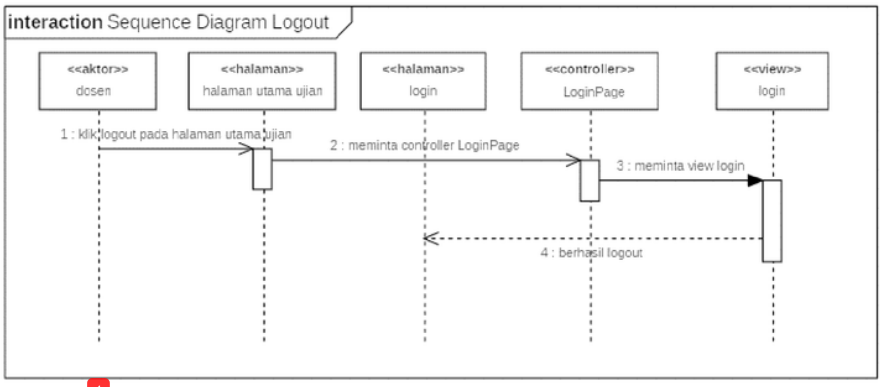


Gambar 4.52 Diagram urutan komunikasi antar objek untuk input nilai

Gambar 4.52 diatas menunjukkan sequence diagram interaksi antar objek dalam melakukan proses input nilai. Dosen klik menu nilai, kemudian halaman

halaman menu nilai memanggil controller homepage. Didalam controller homepage terdapt proses pemanggilan view input nilai untuk menampilkan halaman input nilai. Setelah muncul halaman input nilai, dosen dapat menginputkan nilai. Selanjutnya controller homepage meminta model siatek_model untuk menyimpan data nilai yang diinput pada database. Kemudian controller homepage meminta view nilai untuk menampilkan informasi bahwa nilai berhasil diinput.

h. *Logout* Dosen



Gambar 4.53 Diagram urutan komunikasi antar objek untuk logout dosen

Gambar 4.53 *sequence diagram* diatas menggambarkan interaksi antar objek dalam melakukan proses logout dari web dosen. Dosen klik logout pada halaman web dosen kemudian proses selanjutnya adalah *controller* LoginPage meminta view login untuk menampilkan informasi logout berhasil

4.4 Tahap Pembuatan Aplikasi

Tahap pembuatan aplikasi ini aadalah tahapan penting dalam perancangan sistem yang akan dibangun. Adapun pada tahapan ini dilakukan identifikasi kebutuhan *software* dan *hardware,* proses pengkodean (*coding*), *hosting* dan *domain*, perancangaan *interface* dan implementasi sistem.

4.4.1 Spesifikasi Perangkat Lunak (*Software*) dan Perangkat Keras (*Hardware*)

4.4.1.1 *Software*

Software yang digunakan pada perancangan Aplikasi Ujian *Online* Berbasis *Android* Di Jurusan Teknik Elektro Universitas Sriwijaya adalah sebagai berikut:

Tabel 4.16 Tabel spesifikasi perangkat lunak (*software*) *mobile*

No	Bahasa Pemrograman	Deskripsi
1.	HTML5	HTML sebagai bahasa pemrograman yang mengatur posisi atau tata letak tampilan halaman *login*, halaman *profile*, halaman menu pilihan, halaman *home* ujian, halaman soal ujian, halaman nilai, dan halaman pengumuman pada aplikasi ujian *online* berbasis *android*.
2.	*TypeScript*	*Typescript* merupakan bahasa pemrograman yang digunakan untuk membuat aplikasi ujian *online* berbasis *android* yang berfungsi mengolah data sebelum ditampilkan pada aplikasi di *mobile*.
3.	CSS	CSS adalah bahasa pemrograman yang berfungsi untuk mendesain tampilan seperti memberi warna ungu untuk judul aplikasi pada halaman home, warna hitam untuk nama menu pilihan, profile mahasiswa, dan nama menu untuk halaman home ujian pada aplikasi ujian *online* berbasis *android*.
4.	*Ionic*	*Ionic* adalah *framework* utama yang digunakan untuk *mobile development application* agar aplikasi ujian *online* berbasis *android* dapat berjalan diperangkat *android*.

5.	*Cordova*	*Cordova* merupakan *framework* pendukung yang menyediakan *library* untuk mengakses aplikasi lain di dalam android.
6.	*AngularJS*	*AngularJS* merupakan *framework* pendukung yang menyediakan *library* untuk konversi ke bahasa pemrograman *java* agar aplikasi ujian *online* dapat bekerja pada sistem operasi *mobile*.

Tabel 4.17 Tabel spesifikasi perangkat lunak (*software*) *website*

No	Bahasa Pemrograman	Deskripsi
1.	HTML	HTML sebagai bahasa pemrograman yang dapat mengatur posisi layout atau tata letak tampilan halaman *website* dosen dan admin aplikasi ujian *online*.
2.	PHP	PHP sebagai bahasa pemograman untuk *website* yang berfungsi untuk mengolah data dari *database* yang telah dibuat untuk membangun aplikasi ujian *online*.
3.	CSS	CSS sebagai bahasa pemrograman yang berfungsi untuk mengatur desain tampilan tulisan atau gaya tulisan pada *website* dosen dan admin aplikasi ujian *online*.
4.	*JavaScript*	*JavaScript* sebagai bahasa pemrograman yang digunakan untuk memberikan efek tambahan pada tampilan *website* dosen dan admin aplikasi ujian *online* menjadi lebih dinamis.

5.	*CodeIgniter*	*CodeIgniter* adalah *framework* utama yang digunakan untuk mempercepat desain *website* dosen dan admin pada sistem aplikasi ujian *online*
6.	*BootStrap*	*BootStrap* merupakan *framework* pendukung yang digunakan unutk mempercepat desain halaman *website* dosen dan admin pada sistem aplikasi ujian *online,* dimana template yang sudah tersedia dapat disesuaikan dengan kebutuhan pembuatan *website* aplikasi ujian *online*.

4.4.1.2 *Hardware*

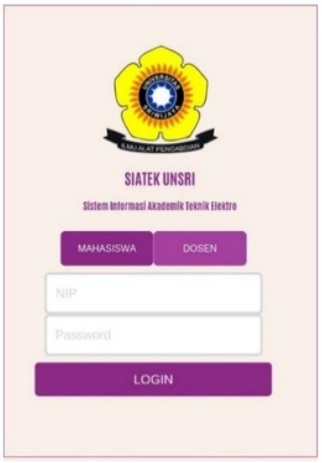
Hardware yang digunakan pada perancangan Aplikasi Ujian *Online* Berbasis *Android* Di Jurusan Teknik Elektro Universitas Sriwijaya adalah sebagai berikut:

Tabel 4.18 Tabel spesifikasi perangkat keras (*hardware*)

NO	Nama Perangkat Keras	Spesifikasi	Jumlah
1.	Laptop	Lenovo Processor : Intel® Core™ i3-5005U CPU @ 2.00GHz 2.00 GHz Installed RAM : 2 GB System Type : 32-bit operating system. X64-based processor	1
2.	*Smartphone Android*	Samsung A6 2018 OS : v5.0.x lollipop Memori internal : 32GB RAM : 3GB Jaringan : 3G/4G	1

4.4.2 Perancangan Antarmuka (*Interface*)

4.4.2.1 Perancangan Antarmuka pada *Android* Mahasiswa



Gambar 4.54 Tampilan halaman *login*

Pada gambar 4.54 merupakan tampilan halaman login mahasiswa pada perangkat android. Untuk masuk ke aplikasi, mahasiswa harus melakukan login dengan memasukkan username dan password . Kemudian memilih level user akses bagi mahasiwa.

Gambar 4.55 Tampilan halaman pilihan

Pada gambar 4.55 merupakan tampilan halaman pilihan. Terdapat menu pilihan absensi, ujian, materi pelajaran, dan bimbingan tugas akhir.

Mahasiswa dapat memilih menu "Ujian" untuk melakukan ujian online pada perangkat android. Pada halaman pilihan, mahasiswa bisa mengedit dan mengganti password dengan mengklik "Profile" pada pojok kanan bawah aplikasi.

Gambar 4.56 Tampilan halaman utama ujian

Pada gambar 4.56 merupakan tampilan halaman utama ujian. Pada halaman utama ujian terdapat beberapa menu pilihan yaitu profile, ujian, nilai, dan pengumuman.

Gambar 4.57 Tampilan halaman menu profile

Pada gambar 4.57 merupakan tampilan halaman menu “Profile”. Pada halaman menu profiel, mahasiswa dapat mengedit dan mengganti passsword.

Gambar 4.58 Tampilan halaman ganti password

Pada gambar 4.58 merupakan tampilan halaman menu “Ganti Password”. Untuk mengganti password baru mahasiswa perlu memasukkan password lama dan password baru, kemudian klik tombol ganti password

4

Gambar 4.59 Tampilan halaman menu ujian

Pada gambar 4.59 merupakan tampilan halaman menu “Ujian”. Pada halaman menu ujian ini terdapat pilihan ujian berupa kuis, ujian tengah semester (UTS), dan Ujian Akhir Semester (UAS). Aplikasi ini hanya untuk ujian yang terjadwal. Tidak untuk semester pendek (SP) maupun ujian susulan.

Gambar 4.60 Tampilan halaman kuis

Pada gambar 4.60 merupakan tampilan halaman menu “Kuis”. Pada halaman ini mhasiswa dapat mengerjakan kuis berdasarkan mata kuliah yang diambil dengan mengklik tombol “Kerjakan” pada tabel aksi.

Gambar 4.61 Tampilan halaman uts

Pada gambar 4.61 merupakan tampilan halaman menu "Ujian Tengah Semester (UTS)". Pada halaman ini mhasiswa dapat mengerjakan kuis berdasarkan mata kuliah yang diambil dengan mengklik tombol "Kerjakan" pada tabel aksi.

Gambar 4.62 Tampilan halaman uas

Pada gambar 4.62 merupakan tampilan halaman menu "Ujian Akhir Semester (UAS)". Pada halaman ini mhasiswa dapat mengerjakan kuis berdasarkan mata kuliah yang diambil dengan mengklik tombol "Kerjakan" pada tabel aksi

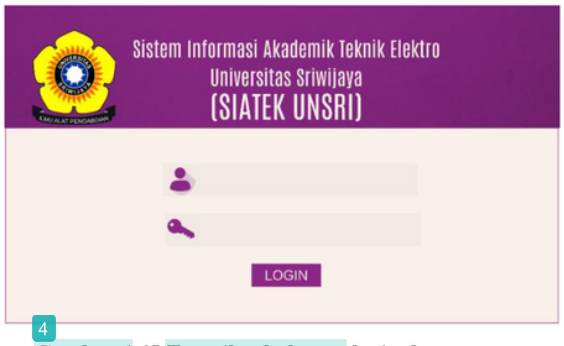
Gambar 4.63 Tampilan Halaman Soal Ujian

Pada gambar 4.63 merupakan tampilan halaman menu "Soal Ujian". Pada halaman soal ujian ini tiap soal akan ditampilkan satu persatu. Dimana mahasiswa dapat mengerjakan soal ujian pada kolom jawaban yang disediakan. Pada halaman ujian ini juga bisa menginput gambar dengan mengambil gambar dari kamera handphone lalu disimpan kemudian memilih file gambar dari galeri handphone.

Gambar 4.64 Tampilan halaman menu nilai

Pada gambar 4.64 merupakan tampilan halaman menu "Nilai". Pada halaman nilai, mahasiswa dapat melihat hasil ujian yang sudah diakumulasi berupa angka dan huruf. Pada halaman menu nilai ini mahasiswa dapat mendownload dan mencetak hasil ujian dengana format word.

b. Perancangan Antarmuka pada *Website* Dosen

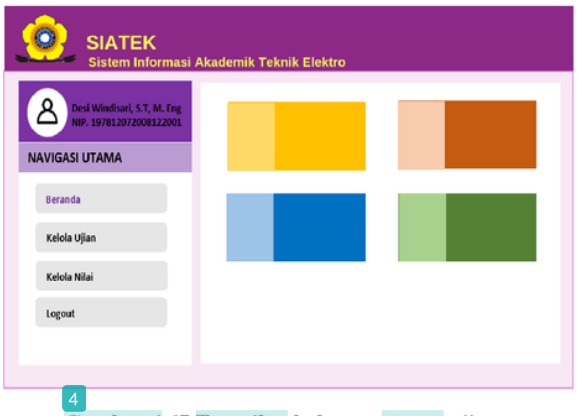


Gambar 4.65 Tampilan halaman login dosen

Pada gambar 4.65 merupakan tampilan halaman login mahasiswa pada perangkat android. Untuk masuk ke aplikasi, mahasiswa harus melakukan login dengan memasukkan username dan password . Kemudian memilih level user akses bagi dosen.

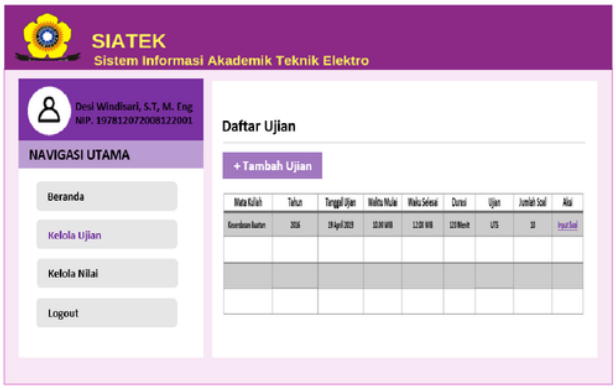
Gambar 4.66 Tampilan halaman pilihan dosen

Pada gambar 4.66 merupakan tampilan halaman pilihan untuk dosen. Terdapat menu pilihan absensi, ujian, materi pelajaran, dan bimbingan tugas akhir. Dosen dapat memilih menu "Ujian" untuk melakukan mengelolah ujian secara online.



Gambar 4.67 Tampilan halaman utama ujian

Pada gambar 4.67 merupakan tampilan halaman utama ujian dosen pada laman website. Pada halaman utama ini terdapat menu pilihan berupa beranda, kelola ujian, kelola nilai dan logout.

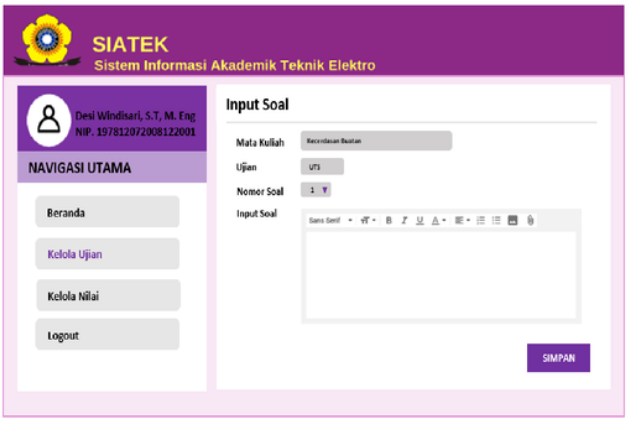


Gambar 4.68 Tampilan halaman menu kelola ujian

Pada gambar 4.68 merupakan tampilan halaman menu "Kelola Ujian". Pada halaman ini dosen telebih dahulu menambahkan mata kuliah yang akan diujikan dengan mengklik "Tambah Ujian". Setelah menambah ujian beserta keterangan waktu ujian dan durasi ujian, dosen dapat memilih aksi input soal atau edit soal.

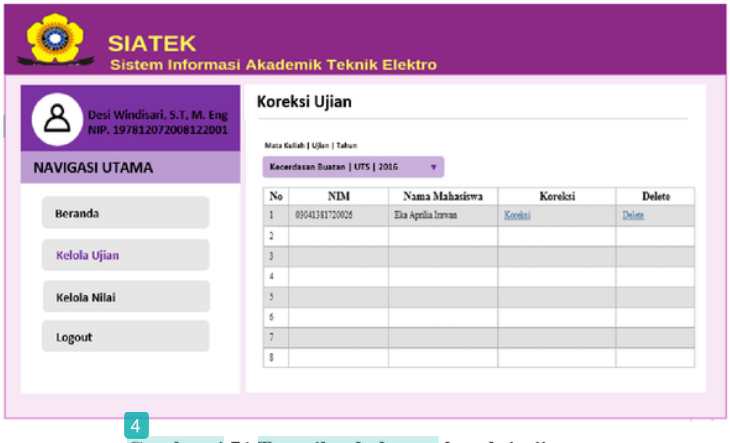
Gambar 4.73 Tampilan halaman tambah ujian

Pada gambar 4.69 merupakan tampilan halaman "Tambah Ujian". Pada halaman tambah ujian, dosen dapat mengatur jadwal ujian, waktu ujian, durasi ujian, dan jenis ujian. Sehingga kegiatan ujian menjadi teratur dan terjadwal.



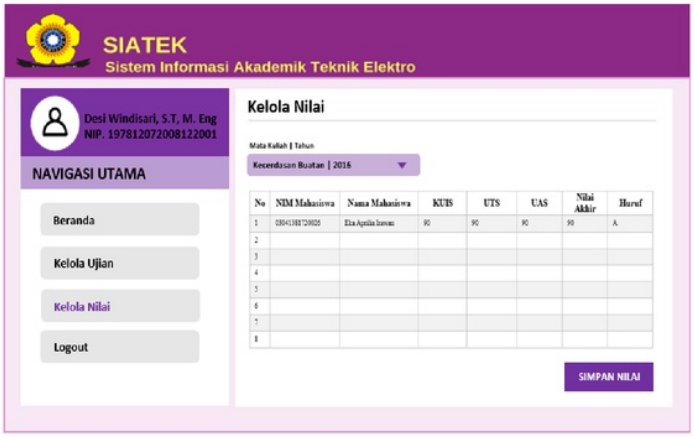
Gambar 4.70 Tampilan halaman input soal ujian

Pada gambar 4.70 merupakan tampilan halaman "Input Ujian". Pada halaman input soal ujian. Pada halaman input soal, dosen dapat menginputkan soal secara manual maupun *attached file* soal.



Gambar 4.71 Tampilan halaman koreksi ujian

Pada gambar 4.71 merupakan tampilan halaman "Koreksi Ujian". Untuk mengkoreksi ujian mahasiswa, dosen bisa mengklik "Koreksi" pada tabel koreksi.



Gambar 4.72 Tampilan halaman menu kelola nilai

Pada gambar 4.72 merupakan tampilan halaman "Nilai". Pada halaman nilai ini dosen dapat melihat nilai ujian mahasiswa yang mengambil mata kuliah tersebut.

4.4.3 Membuat *Database*

Pada tahap pemodelan data terdapat class diagram yang akan digunakan untuk pembuatan database, kemudian dilakukan *Implementasi DataBase Management System* (DBMS) sesuai yang dipilih. Untuk pengolahan *database MySQL* perancangan aplikasi ujian online berbasis android di teknik elektro universitas sriwijaya ini, penulis menggunakan *phpMyAdmin*. Berikut database MySQL yang digunakan untuk menjalankan aplikasi ujian online ini:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_admin	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	username	varchar(30)	latin1_swedish_ci		No	None			Change Drop More
3	password	varchar(40)	latin1_swedish_ci		No	None			Change Drop More
4	nama	varchar(50)	latin1_swedish_ci		No	None			Change Drop More
5	token	varchar(40)	latin1_swedish_ci		No	None			Change Drop More
6	last_activity	varchar(40)	latin1_swedish_ci		No	None			Change Drop More

Gambar 4.73 Tabel database admin

Pada gambar 4.73 diatas merupakan gambar tabel database admin pada phpMyAdmin yang terdiri id_admin menggunakan tipe data int atau interger dengan panjang length 11. Tipe interger dugunakan untuk mempresentasikan angka atau bilangan bulat. Pada username mempunyai tipe data varchar memiliki panjang length 30, password mempunyai tipe data varchar memiliki panjang length 40, dan nama memiliki tipe data varchar dengan panjang length 50. Pada token dan last_activity mempunyai tipe data varchar dengan panjang length masing-masing 40. Tipe data varchar digunakan untuk mempresentasikan karakter dan angka.

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	**id_user_m**	int(11)			No	*None*			Change Drop More
☐	2	**profil_pic**	varchar(40)	latin1_swedish_ci		Yes	*NULL*			Change Drop More
☐	3	**nim**	varchar(14)	latin1_swedish_ci		No	*None*			Change Drop More
☐	4	**nama**	varchar(60)	latin1_swedish_ci		No	*None*			Change Drop More
☐	5	**fakultas**	varchar(40)	latin1_swedish_ci		No	*None*			Change Drop More
☐	6	**jurusan**	varchar(40)	latin1_swedish_ci		No	*None*			Change Drop More
☐	7	**angkatan**	int(4)			No	*None*			Change Drop More
☐	8	**password**	varchar(40)	latin1_swedish_ci		No	*None*			Change Drop More
☐	9	**email**	varchar(40)	latin1_swedish_ci		No	*None*			Change Drop More
☐	10	**kontak**	varchar(13)	latin1_swedish_ci		No	*None*			Change Drop More
☐	11	**token**	varchar(40)	latin1_swedish_ci		Yes	*NULL*			Change Drop More
☐	12	**jenis_kelamin**	varchar(1)	latin1_swedish_ci		No	*None*			Change Drop More
☐	13	**alamat**	varchar(200)	latin1_swedish_ci		No	*None*			Change Drop More
☐	14	**last_activity**	varchar(30)	latin1_swedish_ci		Yes	*NULL*			Change Drop More

Gambar 4.74 Tabel database mahasiswa

Pada gambar 4.74 diatas merupakan gambar tabel database mahasiswa pada phpMyAdmin yang terdiri dari id_user_m dan angkatan menggunakan tipe data int atau interger dengan panjang length 11 untuk id_user dan panjang length 4 untuk angkatan. Tipe interger dugunakan untuk mempresentasikan angka atau bilangan bulat. Pada profil_pc, fakultas, jurusan, password, email, token menggunakan tipe data varchar dengan panjang length 40. Tipe data varchar digunakan untuk mempresentasikan karakter. Pada nim, nama, kontak, jenis_kelamin, dan last_activity meenggunakan tipe data varchar dengan panjang length 14 untuk nim, panjang length 60 untuk nama, panjang length 13 untuk kontak, dan panjang length 30 untuk last_activity. Sedangan untuk alamat menggunakan tipe data varchar dengan panjang length 200.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_user_d	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	profil_pic	varchar(40)	latin1_swedish_ci		Yes	NULL			Change Drop More
3	nip	varchar(14)	latin1_swedish_ci		No	None			Change Drop More
4	nama	varchar(60)	latin1_swedish_ci		No	None			Change Drop More
5	password	varchar(40)	latin1_swedish_ci		No	None			Change Drop More
6	email	varchar(40)	latin1_swedish_ci		No	None			Change Drop More
7	kontak	varchar(13)	latin1_swedish_ci		No	None			Change Drop More
8	token	varchar(40)	latin1_swedish_ci		Yes	NULL			Change Drop More
9	jenis_kelamin	varchar(1)	latin1_swedish_ci		No	None			Change Drop More
10	alamat	varchar(100)	latin1_swedish_ci		No	None			Change Drop More
11	pendidikian_terakhir	varchar(30)	latin1_swedish_ci		No	None			Change Drop More
12	status_dosen	varchar(40)	latin1_swedish_ci		No	None			Change Drop More
13	last_activity	varchar(30)	latin1_swedish_ci		Yes	NULL			Change Drop More

Gambar 4.75 Tabel database dosen

Pada gambar 4.75 diatas merupakan gambar tabel database dosen pada phpMyAdmin yang terdiri id_user_d menggunakan tipe data int atau interger dengan panjang length 11. Pada profil_pic, password, email, token, dan status_dosen memiliki panjang length 40. pada nip, nama,, kontak, jenis_kelamin menggunakan tipe data varchar dan memiliki panjang length masing-masing 14 untuk nip, 60 untuk panjang length nama, panjang length 13 untuk kontak, panjang length 1 untuk jenis_kelamin. Sedangkan pada pendidikan_terakhir dan last_activity mempunyai tipe data varchar dengan panjang length 30. Tipe data varchar digunakan untuk mempresentasikan karakter. Sedangkan tipe interger dugunakan untuk mempresentasikan angka atau bilangan bulat.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_mk	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	nama	varchar(60)	latin1_swedish_ci		No	None			Change Drop More
3	sks	int(1)			No	None			Change Drop More
4	jumlah_materi	int(3)			No	None			Change Drop More
5	jumlah_dosen	int(1)			No	None			Change Drop More
6	jumlah_kelas	int(2)			No	None			Change Drop More
7	semester	varchar(8)	latin1_swedish_ci		No	None			Change Drop More
8	kuis	int(11)			No	None			Change Drop More
9	uts	int(11)			No	None			Change Drop More
10	uas	int(11)			No	None			Change Drop More

Gambar 4.76 Tabel database mata kuliah

Pada gambar 4.76 diatas merupakan gambar tabel database mata kuliah pada phpMyAdmin yang terdiri id_mk, kuis, uts, dan uas menggunakan tipe data int atau interger dengan panjang length 11. Pada sks, jumlah_dosen mempunyai panjang length 1 dengan tipe data int atau interger. Pada jumlah_materi menggunkan tipe data interger atau int dengan panjang length 3. Pada jumlah_kelas menggunakan tipe data int dengan panjang length 2. Pada nama dan semester menggunakan tipe data varchar dengan panjang length 60 untuk nama dan panjang length 8 untuk semester. Tipe data varchar digunakan untuk mempresentasikan karakter. Sedangkan tipe data interger dugunakan untuk mempresentasikan angka atau bilangan bulat

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	**id_krs**	int(11)			No	*None*		AUTO_INCREMENT	Change Drop More
☐	2	**id_user**	int(11)			No	*None*			Change Drop More
☐	3	**semester**	varchar(8)	latin1_swedish_ci		No	*None*			Change Drop More
☐	4	**tahun_ajaran**	varchar(11)	latin1_swedish_ci		No	*None*			Change Drop More
☐	5	**id_mk**	int(11)			No	*None*			Change Drop More
☐	6	**id_jadwal**	int(11)			No	*None*			Change Drop More
☐	7	**kuis_u**	int(11)			No	*None*			Change Drop More
☐	8	**uts_u**	int(11)			No	*None*			Change Drop More
☐	9	**uas_u**	int(11)			No	*None*			Change Drop More
☐	10	**n_kuis**	int(3)			No	*None*			Change Drop More
☐	11	**n_uts**	int(3)			No	*None*			Change Drop More
☐	12	**n_uas**	int(3)			No	*None*			Change Drop More

Gambar 4.77 Tabel database kartu rencana studi

Pada gambar 4.77 diatas merupakan gambar tabel database kartu rencana studi pada phpMyAdmin yang terdiri id_krs, id_user, id_mk, id_jadwal, kuis_u, uts_u, dan uas_u. menggunakan tipe data int atau interger dengan panjang length 11. Pada n_kuis, n_uts, dan n_uas menggunakan tipe data int dengan panjang length 3. Pada semester dan tahun_ajaran menggunakan tipe data varchar dengan panjang length 8 untuk semester dan panjang length 11 untuk tahun ajaran. Tipe data varchar digunakan untuk mempresentasikan karakter atau numerik. Sedangkan tipe data interger dugunakan untuk mempresentasikan angka atau bilangan bulat

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_soal	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	id_mata_kuliah	int(11)			No	None			Change Drop More
3	nomor_soal	int(11)			No	None			Change Drop More
4	ujian	varchar(4)	latin1_swedish_ci		No	None			Change Drop More
5	gambar	varchar(50)	latin1_swedish_ci		No	None			Change Drop More
6	soal	varchar(200)	latin1_swedish_ci		No	None			Change Drop More

Gambar 4.78 Tabel database soal ujian

Pada gambar 4.78 diatas merupakan gambar tabel database soal ujian pada phpMyAdmin yang terdiri id_soal, id_mata_kuliah, dan nomor_soal. menggunakan tipe data int atau interger dengan panjang length 11. Tipe data interger atau int digunakan untuk mempresentasikan angka atau bilangan bulat. Pada ujian menggunakan tipe data varchar dengan panjang length 50, dan pada soal menggunaka tipe varchar dengan panjang length 200. Tipe data varchar digunakan untuk mempresentasikan alfanumerik.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_nilai	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	id_user	int(11)			No	None			Change Drop More
3	id_mk	int(11)			No	None			Change Drop More
4	ujian	int(11)			No	None			Change Drop More
5	nilai	int(11)			No	None			Change Drop More
6	index	varchar(1)	latin1_swedish_ci		No	None			Change Drop More

Gambar 4.79 Tabel database nilai

Pada gambar 4.79 diatas merupakan gambar tabel database nilai pada phpMyAdmin yang terdiri id_nilai, id_user, id_mk, ujian, dan nilai. menggunakan tipe data int atau interger dengan panjang length 11. Tipe data interger atau int digunakan untuk mempresentasikan angka atau bilangan bulat. Pada tipe data index menggunakan tipe data varchar ddengan panjang length 1. Tipe data varchar digunakan untuk mempresentasikan alfanumerik.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_pengumuman	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	judul	varchar(40)	latin1_swedish_ci		No	None			Change Drop More
3	isi	varchar(200)	latin1_swedish_ci		No	None			Change Drop More
4	tgl	varchar(30)	latin1_swedish_ci		No	None			Change Drop More
5	gambar	varchar(40)	latin1_swedish_ci		No	None			Change Drop More

Gambar 4.80 Tabel database pengumuman

Pada gambar 4.80 diatas merupakan gambar tabel database jawaban ujian pada phpMyAdmin yang terdiri id_pengumuman menggunakan tipe data int atau interger dengan panjang length 11. Tipe data interger atau int digunakan untuk mempresentasikan angka atau bilangan bulat. Pada judul, isi, tgl, dan gambar menggunakan tipe data varchar dengan panjang length 40 untuk judul, panjang length 200 untuk isi, panjang length 30 untuk tgl, dan panjang length 40 untuk gambar. Tipe data varchar digunakan untuk mempresentasikan alfanumerik.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_jawaban_ujian	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	id_ujian	int(11)			No	None			Change Drop More
3	ujian	int(11)			No	None			Change Drop More
4	nomor	int(11)			No	None			Change Drop More
5	gambar	int(11)			No	None			Change Drop More
6	jawaban	int(11)			No	None			Change Drop More
7	koreksi	int(11)			No	None			Change Drop More

Gambar 4.81 Tabel database jawaban ujian

Pada gambar 4.81 diatas merupakan gambar tabel database jawaban ujian pada phpMyAdmin yang terdiri id_jawaban_ujian, id_ujian, ujian, nomor, gamabar, jawaban, dan koreksi menggunakan tipe data int atau interger dengan panjang length 11. Tipe data interger atau int digunakan untuk mempresentasikan angka atau bilangan bulat

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_waktu_ujian	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	id_user	int(11)			No	None			Change Drop More
3	id_mk	int(11)			No	None			Change Drop More
4	ujian	int(11)			No	None			Change Drop More
5	waktu_mulai	int(11)			No	None			Change Drop More
6	waktu_selesai	int(11)			No	None			Change Drop More

Gambar 4.82 Tabel database waktu ujian

Pada gambar 4.82 diatas merupakan gambar tabel database waktu ujian pada phpMyAdmin yang terdiri id_waktu_ujian, id_user, id_mk, ujian, waktu_mulai, dan waktu_selesai menggunakan tipe data int atau interger dengan panjang length 11. Tipe data interger atau int digunakan untuk mempresentasikan angka atau bilangan bulat.

4.4.4 Pengkodean Data (*Coding*)

Pengkoden data (coding) adalah proses penerjemahan atau pengimplementasian persyaratan logika dari diagram-diagram atau sekumpulan intruksi kedalam suatu bahasa pemrograman dengan bantuan text editor yang sesuai dengan tujuan agar tercipta suatu sistem. Pengkodean data pada aplikasi ujian online, codingan terlampir pada lampiran.

4.4.5 Web Hosting dan Domain

Web Hosting dan Domain pada perancangan aplikasi ujian online berbasis android di Jurusan Teknik Elektro Universitas Sriwijaya digunakan untuk dosesn dan admin untuk akses website.

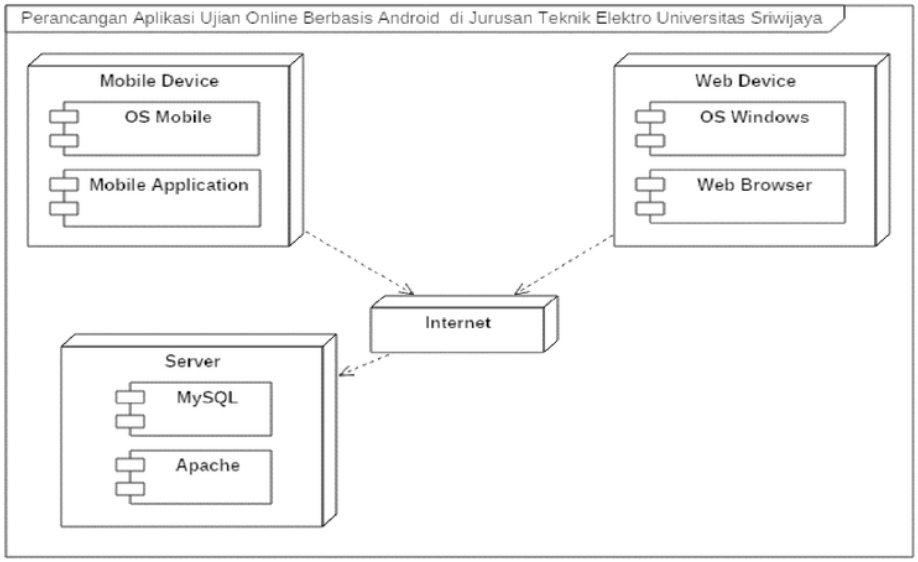
Web Hosting yang digunakan pada aplikasi ujian online berbasis android di jurusan elektro universitas sriwijaya adalah menggunakan hosting ardetamedia menggunakan cPanel sesuai dengan kebutuhan sistem yang diinginkan. cPanel merupakan control panel online sebuah hosting untuk membuat dan mengupload database, mengelola website dan lain-lain.

Domain digunakan untuk mengidentifikasi suatu alamat website berupa nama yang unik sebagai pengganti alamat IP supaya mudah untuk dihafal. Pada sistem ini domain digunakan untuk mengaskses sistem dosen dan sistem admin yaitu dengan menggunakan domain *Generic Top Level Domain* (gTLD) dengan akhiran .com pada sistem. Domain sistem pada website yang penulis gunakan adalah http://jteunsri.com/

4.4.6 Proses *build* sistem menjadi aplikasi (apk)

Adapun proses untuk *build* aplikasi ujian online menjadi aplikasi (apk) yaitu harus mempunyai *framework ionic*, android SDK untuk membantu proses *build*, aplikasi XAMPP untuk mengaktifkan *localhost*. Selanjutnya buka *Command Prompt* pada *windows* untuk memanggil *framework ionic*. Kemudian panggil folder tempat menyimpan kodingan yang telah dibuat dan mengetikkan *"ionic cordova build android"* pada *Command Prompt* lalu tunggu sampai proses *result* selesai. Lalu apk yang telah selesai di build akan tersimpan dalam folder build.

4.4.7 Proses Menjalankan *Software* dan *Hardware*



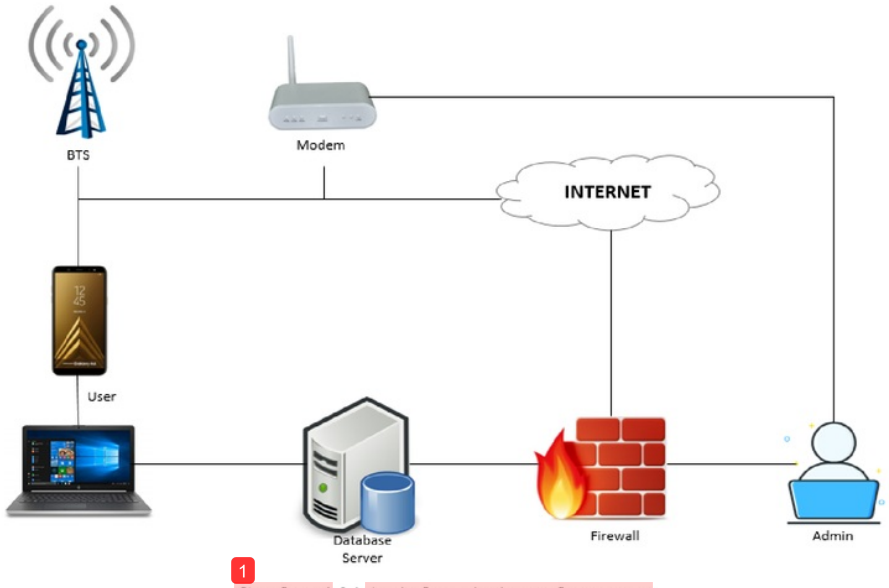
4.83 *Deployment* diagram aplikasi ujian *online* berbasis *android*

Gambar 4.83 merupakan gambar *deployment* diagram aplikasi ujian online berbabasis *android* di jurusan teknik elektro universitas sriwijaya. Dari gambar diatas dapat dijelaskan bahwa untuk *node Mobile Device* berisikan komponen pendukung dalam pembuatan aplikasi *mobile* sistem aplikasi ujian *online* yang terdiri dari *OS Mobile* dan *Mobile Application*. *OS* digunakan untuk mengontrol kinerja pada sistem aplikasi ujian oline berbasis androidm, dan *OS* juga yang akan menampilkan aplikasi pada layar. OS yang digunakan pada sistem ini adalah OS versi 8.0.0 oreo. Komponen kedua yaitu *Mobile Application*, dimana *Mobile Application* adalah sebuah *software* yang dijalankan pada perangkat *mobile*.

Pada n*ode Web Device* berisikan komponen pendukung dalam pembuatan *website* di sistem aplikasi ujian *online* yang terdiri dari *OS Windows* dan *Web Browser*. Kedia komponen digunakan untuk menjalankan sistem pada *website* yang digunakan dosen dan admin. Untuk menajalankan dan menampilkan sistem yang dibangun pada website maka digunakan *OS Windows*. Windows yang digunakan pada sistem yaitu *Windows 32-bit Operating System*. Sedangkan *Web Browser* adalah adalah sebuah media untuk membuka *website* dosen dan admin juga digunakan sebagai media simulasi dari sistem.

Kedua *device* tersebut terintegrasi melalui internet dengan menggunakan koneksi paket data sehingga dapat terhubung ke *server*. Sedangkan *node server* berfungsi untuk menyimpan data dari sistem ujian *online* yang terdiring dari *MySQL* dan *Apache*. *MySQL* merupakan *database server* yang dapat menambah, mengedit, dan menghapus data pada *database*. Sedangkan *Apache* adalah *application server* yang berfungsi untuk membuat koneksi antara *server* dengan *browser website*.

4.4.8 Jaringan Komputer



Gambar 4.84 Arsitektur jaringan komputer

Gambar 4.84 merupakan arsitektur jaringan komputer pada perancangan ujian online. Pada sistem ini jaringan komputer harus terkoneksi dengan internet. Pengguna (*user*) pada perancangan aplikasi sistem ujian online terdiri dari 3 aktor, dimana 2 aktor sebagai pengguna yaitu mahasiswa dan dosen dan 1 aktor sebagai admin yang mengontrol dan mengawasi sistem. Untuk pengguna mahasiswa aplikasi dapat diakses menggunakan perangkat mobile android, sedangkan dosen bisa mengakses website menggunakan perangkat PC atau laptop. Untuk mengakses aplikasi ujian, setiap perangkat haruslah terhubung dengan internet melalui koneksi seluler atau modem router. Kemudian data-data yang diperlukan akan dioleah oleh sitem dan kemudian disimpan dalam database dan server. Firewall digunakan untuk keaman data jaringan.

4.4.9 Keamanan Sistem

Pada perancangan aplikasi ujian online berbasis android di Jurusan Teknik Elektro Universitas Sriwijaya menggunakan kode enkripsi MD5 dengan sistem token. Sistem token berupa kode-kode unik yang akan didapatkan berbeda-beda

untuk setiap kali user melakukan sesi login atau masuk ke aplikasi. Dilakukan pencocokan data token pada perangkat user dengan data yang ada pada database. Sebuah token bisa digunakan di satu perangkat saja, jadi ketika user berpindah ke perangkat lain maka akan mendapatkan token baru dan akan logout dari sebuah aplikasi

4.4.10 Implementasi *User Interface* (Antar Muka)

a. *User Interface* Mahasiswa

Berikut ini adalah tampilan aplikasi mahasiswa berdasarkan hasil setelah dilakukan proses pengkodingan :

Gambar 4.85 Tampilan logo

Gambar 4.85 merupakan tampilan logo aplikasi pada perangkat android. Untuk masuk ke aplikasi, mahasiswa harus melakukan login dengan memilih tipe user login yaitu mahasiswa.

Gambar 4.86 Tampilan login

Gambar 4.86 merupakan tampilan halaman login mahasiswa pada perangkat android. Untuk masuk ke aplikasi, mahasiswa harus melakukan login dengan memasukkan username berupa Nomor Induk Mahasiswa dan password . Kemudian memilih pilhan tipe user login yaitu mahasiswa.

Gambar 4.87 Tampilan halaman pilihan

Gambar 4.87 merupakan tampilan pilihan pada perangkat android. Untuk masuk ke halaman utama ujian, mahasiswa dengan mengklik icon ujian pada halaman ujian

Gambar 4.88 Tampilan profile

Gambar 4.88 merupakan tampilan profile pada perangkat android. Pada halaman profile ini, mahasiswa dapat mengganti password akun dan juga logout.

Gambar 4.89 Tampilan ganti password

Gambar 4.89 merupakan tampilan halaman ganti password pada perangkat android. Pada halaman ganti password, mahasiswa harus mengisikan username lama dengan username baru yang akan diganti.

Gambar 4.90 Tampilan Utama

Gambar 4.90 merupakan tampilan utama pada perangkat android. Pada halaman utama terdapat menu profile, ujian, nilai, dan pengumuman.

Gambar 4.91 Tampilan Menu Ujian

Gambar 4.91 merupakan tampilan menu ujian, dimana akan menampilkan halaman Mata Kuliah yang diambil.

Gambar 4.92 Tampilan Pilihan Ujian

Gambar 4.92 merupakan tampilan yang akan muncul ketika mahasiswa memilih mata kuliah pada menu ujian.

Gambar 4.93 Tampilan list soal

Gambar 4.93 merupakan tampilan yang akan muncul ketika mahasiswa memilih mata kuliah pada menu ujian

Gambar 4.94 Tampilan soal ujian

Gambar 4.94 merupakan tampilan menu Soal Ujian setelah mengklik kerjakan pada setiap nomor soal. Pada halaman soal ujian ini tiap soal akan ditampilkan satu persatu. Dimana mahasiswa dapat mengerjakan soal ujian pada kolom jawaban yang disediakan, dan juga dapat menguploade gambar jika dibutuhkan.

Gambar 4.95 Tampilan nilai

Gambar 4.66 merupakan tampilan menu Nilai. Mahasiwa dapat melihat index nilai dari seluruh ujian.

Gambar 4.96 Tampilan pengumuman

Gambar 4.96 merupakan tampilan menu pengumuman. Mahasiswa dapat melihat pengumuman tentang akademik, tanggal ujian dan lainnya.

b. *User Interface* Pada *Website* Dosen

Gambar 4.97 Tampilan halaman login dosen

Gambar 4.97 merupakan tampilan halaman *login* dosen pada *website*. Untuk masuk ke halaman utama dosen, dosen harus melakukan *login* dengan memasukkan NIP dan *password* . Kemudian memilih pilhan tipe *user login* yaitu dosen.

Gambar 4.98 Tampilan halaman utama dosen

Gambar 4.98 merupakan tampilan halaman utama dosen pada perangkat wevsite. Pada halaman ini dosen memilih menu "Ujian", maka halaman menu ujian akan muncul. Pada halaman menu ujian, terdapat pilihan soal ujian dan koreksi. Untuk menginput soal ujian, dosen mengklik pilihan "soal ujian" maka sistem akan menampilkan form input soal. Pada pilihan "koreksi", sistem akan menampilkan halaman koreksi ujian.

Gambar 4.99 Tampilan halaman input soal ujian

Gambar 4.99 merupakan tampilan halaman input soal ujian. Untuk menginput soal ujian dosen dapat mengisi form input soal yang sudah disediakan.

Gambar 4.100 Tampilan halaman koreksi soal ujian

Gambar 4.100 merupakan tampilan halaman koreksi soal ujian. Pada halaman koreksi nilai dosen bisa memilih jenis ujian, kemudian memilih nama mahasiswa, dan klik tombol "pilih mahasiswa" maka akan muncul halaman jawaban mahasiswa yang dipilih.

Gambar 4.101 Tampilan halaman input nilai

Gambar 4.101 merupakan tampilan halaman input nilai. Pada halaman input nilai, dosen bisa melakukan input nilai kuis, uts, dan uas.

Gambar 4.102 Tampilan halaman data nilai

c. *User Interface* Pada *Website* Admin

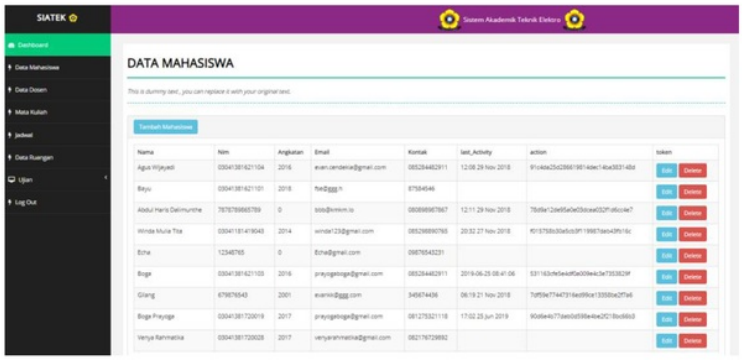
Gambar 4.100 Tampilan halaman *login* admin

Gambar 4.100 merupakan tampilan halaman *login* admin pada *website*. Untuk masuk ke halaman utama admin admin harus melakukan *login* dengan memasukkan *Username* dan *password* .

Gambar 4.71 Tampilan Halaman Login Admin

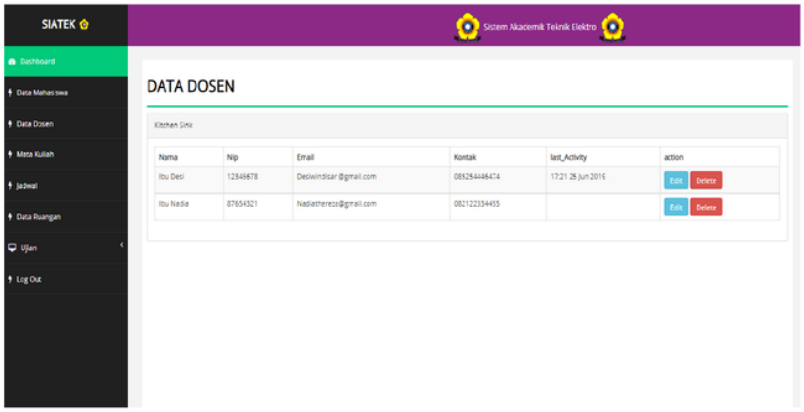
Gambar 4.101 Tampilan halaman utama admin

Gambar 4.101 merupakan tampilan halaman utama admin pada perangkat *website*. Pada halaman utama ini, admin dapat melakukan tambah, edit, dan menghapus data.



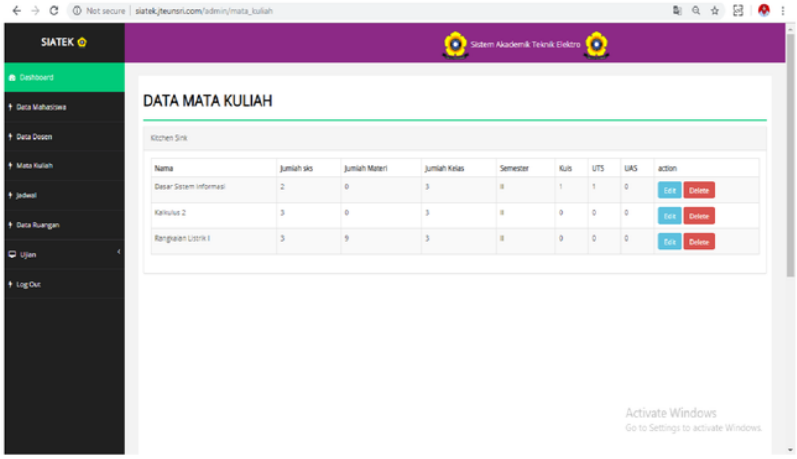
Gambar 4.102 Tampilan halaman data mahasiswa

Gambar 4.102 merupakan tampilan halaman Data Mahasiswa. Pada halaman data mahasiswa admin bisa menambah, mengedit, dan menghapus data mahasiswa.



Gambar 4.103 Tampilan halaman data dosen

Gambar 4.103 merupakan tampilan halaman Data Dosen. Pada halaman data dosen, admin bisa menambah, mengedit, dan menghapus data dosen.



Gambar 4.104 Tampilan halaman data mata kuliah

Gambar 4.104 merupakan tampilan halaman Mata Kuliah. Pada halaman mata kuliah, admin bisa menambah, mengedit, dan menghapus data mata kuliah.

4.5 Tahap Pengujian dan *Turnover*

Setelah tahapan pembentukan aplikasi maka dilakukan pengujian terhadap sistem ujian online yang sudah dibuat. Tahapan pembentukan aplikasi menghasilkan aplikasi mobile berbasis android untuk mahasiswa, sedangkan untuk dosen dan admin berupa web. Pengujian dilakukan untuk mengetahui apakah sistem yang diingin sudah sesuai dengan yang diharapkan, masih terdapat kesalahan atau tidak dengan metode *Blackbox Testing* sebagai berikut :

4.5.1 Tahap Pengujian dengan *Black Box Testing*

4.5.1.1 *Black box testing* untuk aplikasi *mobile* mahasiswa

Tabel 4.19 *Black box* fungsionalitas login mahasiswa

No.	Tanggal Pengujian	Skenario	Test Case	Harapan	Hasil
1.	1-06-2019	Tidak memilih level user	Mahasiswa : - Dosen : -	Sistem akan menolak akses login dan tidak akan menampilkan halaman login	Valid
2.	1-06-2019	Tidak mengisi semua *field* pada halaman *login*	NIM : - Password : - Klik Login	Sistem akan menolak akses login dan menmpilkan pemberitahuan "Silahkan Lengkapi NIM dan Password Anda!"	Valid
3	1-06-2019	Mengisi salah satu *field* pada halaman *login*, dengan menggisikan NIM.	NIM: 03041381720 26 Password : - Klik Login	Sistem akan menolak akses login dan menmpilkan pemberitahuan "Silahkan Lengkapi NIM dan Password Anda!"	Valid
4	1-06-2019	Mengisi seluruh *field* pada halaman *login*, dengan menggunakan NIM dan *password* yang akan	NIM: 03041381720 026 Password : nana12 Klik Login	Sistem akan menerima akses login dan menampilkan pemberitahuan "Berhasil Login"	Valid

		digunakan setiap *login.*			

Tabel 4.20 *Black box* fungsionalitas ujian

No.	Tanggal Pengujian	Skenario	Test Case	Harapan	Hasil
1.	1-06-2019	Masuk halaman utama	Mengklik icon ujian pada halaman pilihan setelah berhasil login	Berhasil masuk ke halaman utama mahasiswa	Valid
2.	1-06-2019	Memilih menu ujian	Mengklik menu ujian pada halaman mahasiswa	Berhasil masuk ke halaman menu ujian	Valid
3.	1-06-2019	Masuk ke menu profile pada halaman utama mahasiswa, untuk melihat profile mahasiswa	Mengklik menu profile	Berhasil masuk ke halaman menu profile	Valid
4.	1-06-2019	Masuk ke menu ujian untuk melakukan ujian sesuai dengan memilih mata kuliah terlebih dahulu	Mengklik salah satu mata kuliah	Berhasil masuk halaman pilihan ujian	Valid
5.	1-06-2019	Mahasiswa memilih Ujian Tengah Semester (UTS)	Mengklik pilihan UTS	Berhasil masuk halaman UTS	Valid
6.	1-06-2019	Mahasiswa memilih Ujian Akhir Semester (UAS)	Mengklik pilihan UAS	Berhasil masuk halaman UAS	Valid
7.	1-06-2019	Mahasiswa mengerjakan soal ujian	Mengklik tombol kerjakan	Berhasil masuk halaman ujian	Valid

8.	1-06-2019	Mahasiswa menginput gambar yang diperlukan pada saat ujian	Mengklik icon kamera pada halaman ujian	Berhasil menginput gamabar yang dibutuhkan	Valid
9.	1-06-2019	Pada saat ujian berlangsung, mahasiswa berpindah ke soal berikutnya	Mengklik tombol next pada halaman ujian untuk berpindah ke soal berikutnya	Berhasil berpindah kesoal selanjutnya	Valid
10.	1-06-2019	Mahasiswa mensubmit semua jawaban ketika selesai mengerjakan ujian	Mengklik tombol submit	Behasil mensubmit seluruh jawaban ujian	Valid
11.	1-06-2019	Mahasiswa memilih menu nilai	Mengklik menu nilai pada halaman utama mahasiswa	Berhasil masuk ke halaman menu nilai	Valid
12.	1-06-2019	Mahasiswa masuk ke menu pengumuman	Mengklik menu pengumuman pada halaman mahasiswa	Berhasil masuk ke halaman menu pengumuman	Valid
13.	1-06-2019	Mahasiswa mencetak nilai dalam format word.	Mengklik tombol ms.word	Berhasil mencetak hasil ujian	Valid

Tabel 4.21 *Black box testing* fungsionalitas ganti password

No	Tanggal Pengujian	Skenario	Test Case	Harapan	Hasil
1.	1-06-2019	Tidak mengisi seluruh *field* pada halaman *Ganti Password.*	Password Lama: - Password Baru: - Konfirmasi Password Baru: -	Sistem akan menolak akses *Ganti Password* dan menampilkan pemberitahuan "Silahkan Lengkapi Data Password Anda!"	Valid

			Klik Update	pada halaman ganti password.	
2.	1-06-2019	Mengisi salah satu *field* pada halaman *Ganti Password*.	Password Lama: venyarahmatika Password Baru: - Konfirmasi Password Baru: - Klik Update	Sistem akan menolak akses *Ganti Password* dan menampilkan pemberitahuan "Silahkan Lengkapi Data Password Anda!" pada halaman ganti password.	Valid
3.		Mengosongkan salah satu *field* pada halaman *Ganti Password*.	Password Lama: - Password Baru: siatekunsri Konfirmasi Password Baru: siatekunsri Klik Update	Sistem akan menolak akses *Ganti Password* dan menampilkan pemberitahuan "Silahkan Lengkapi Data Password Anda!" pada halaman ganti password.	Valid
4.	1-06-2019	Mengisi seluruh field pada halaman *Ganti Password* namun dengan mengisikan password lama yang salah.	Password Lama: venyarahmatikaaa Password Baru: siatekunsri Konfirmasi Password Baru: siatekunsri Klik Update	Sistem akan menolak akses *Ganti Password* dan menampilkan pemberitahuan "Maaf,Password yang anda masukan salah! Silahkan coba lagi!" pada halaman ganti password.	Valid
5.	1-06-2019	Mengisi seluruh *field* pada halaman *Ganti Password* dengan data yang benar.	Password Lama: venyarahmatika Password Baru: siatekunsri Konfirmasi Password Baru: siatekunsri Klik Update	Sistem akan menerima akses *Ganti Password* dan menampilkan pemberitahuan " Update Berhasil!" pada halaman ganti password.	Valid

Tabel 4.22 *Black box testing* fungsionalitas logout mahasiswa

No	Tanggal Pengujian	Skenario	Test Case	Harapan	Hasil

1.	1-06-2019	Melakukan *Logout*	-Klik Icon Home -Klik Profile -Klik Logout	Sistem akan melakukan melakukan *logout*. Dan kembali ke halaman logn awal	Valid

4.5.1.2 *Black box testing* untuk *website* dosen

Tabel 4.23 *Black box testing* fungsionalitas login dosen

No	Tanggal Pengujian	Skenario	Test Case	Harapan	Hasil
1.	3-06-2019	Tidak mengisi seluruh *field* pada halaman *login*.	NIP : - Password : - Klik Login	Sistem akan menolak akses *login* dan menampilkan pemberitahuan "Silahkan Lengkapi Username dan Password Anda!".	Valid
2.	3-06-2019	Mengisi salah satu *field* pada halaman *login*.	NIP : 197812072008122001 Password : - Klik Login	Sistem akan menolak akses *login* dan menampilkan pemberitahuan "Silahkan Lengkapi Username dan Password Anda!."	Valid
3.	3-06-2019	Mengisi seluruh *field* pada halaman *login*, namun dengan NIP yang belum pernah mendaftar.	NIP : 197812072008122002 Password : - Klik Login	Sistem akan menolak akses *login* dan menampilkan pemberitahuan "Maaf Username dan Password yang Anda masukan Salah!"	Valid
4.	3-06-2019	Mengisi seluruh *field* pada halaman *login*, dengan data yang benar.	NIP : 197812072008122001 Password : desiwindi Klik Login	Sistem menerima akses *login* kemudian menampilkan halaman beranda media pembelajarn	Valid

Tabel 4.24 *Black box* fungsionalitas ujian

No.	Tanggal Pengujian	Skenario	Test Case	Harapan	Hasil
1.	3-06-2019	Dosen masuk ke halaman pilihan	Mengklik icon ujian pada halaman	Berhasil masuk ke halaman utama dosen	Valid

			pilihan setelah berhasil login		
2.	3-06-2019	Dosen masuk menu kelola ujian	Mengklik icon menu kelola ujian dalam menu ujian	Berhasil masuk ke halaman kelola ujian	Valid
3.	3-06-2019	Dosen menginput soal ujian	Mengklik input soal pada tabel aksi kelola ujian	Berhasil masuk ke halaman input soal	Valid
4.	3-06-2019	Dosen mengkoreksi soal ujia	Mengklik koreksi ujian pada tabel aksi kelola ujian	Berhasil masuk ke halaman koreksi ujian	Valid
5.	3-06-2019	Dosen masuk menu kelola nilai	Mengklik menu kelola nilai pada menu ujian	Berhasil masuk ke halaman kelola ujian	Valid

Tabel 4.25 *Black box* fungsionalitas ujian

No	Tanggal Pengujian	Skenario	Test Case	Harapan	Hasil
1.	3-06-2019	Tidak mengisi seluruh *field* pada halaman *Ganti Password*.	Password Lama: - Password Baru: - Konfirmasi Password Baru: - Klik Update	Sistem akan menolak akses *Ganti Password* dan menampilkan pemberitahuan "Silahkan Lengkapi Data Password Anda!" pada halaman ganti password.	Valid
2.	3-06-2019	Mengisi salah satu *field* pada halaman *Ganti Password*.	Password Lama: desiwindi Password Baru: - Konfirmasi Password Baru: - Klik Update	Sistem akan menolak akses *Ganti Password* dan menampilkan pemberitahuan "Silahkan Lengkapi Data Password Anda!" pada halaman ganti password.	Valid
3.	3-06-2019	Mengosongkan salah satu *field* pada halaman *Ganti Password*.	Password Lama: - Password Baru: siatekunsri Konfirmasi Password Baru: siatekunsri Klik Update	Sistem akan menolak akses *Ganti Password* dan menampilkan pemberitahuan "Silahkan Lengkapi Data Password Anda!" pada halaman ganti password.	Valid

4.	3-06-2019	Mengisi seluruh field pada halaman *Ganti Password* namun dengan mengisikan password lama yang salah.	Password Lama: desiwindiii Password Baru: siatekunsri Konfirmasi Password Baru: siatekunsri Klik Update	Sistem akan menolak akses *Ganti Password* dan menampilkan pemberitahuan "Maaf,Password yang anda masukan salah! Silahkan coba lagi!" pada halaman ganti password.	Valid
5.	3-06-2019	Mengisi seluruh *field* pada halaman *Ganti Password* dengan data yang benar.	Password Lama: venyarahmatika Password Baru: siatekunsri Konfirmasi Password Baru: siatekunsri Klik Update	Sistem akan menerima akses *Ganti Password* dan menampilkan pemberitahuan " Update Berhasil!" pada halaman ganti password.	Valid

Tabel 4.26 *Black box testing* fungsionalitas logout dosen

No	Tanggal Pengujian	Skenario	Test Case	Harapan	Hasil
1.	3-06-2019	Melakukan *Logout*	-Klik Menu -Klik Profile -Klik Logout	Sistem akan melakukan melakukan *logout.* Dan kembali ke halaman login.	Valid

4.5.1.3 *Black box testing* untuk web Admin

Tabel 4.27 *Black box testing* fungsionalitas login admin

No	Tanggal Pengujian	Skenario	Test Case	Harapan	Hasil
1.	3-06-2019	Tidak mengisi seluruh *field* pada halaman *login*.	NIP : - Password : - Klik Login	Sistem akan menolak akses *login* dan menampilkan pemberitahuan "Silahkan Lengkapi Username dan Password Anda!".	Valid
2.	3-06-2019	Mengisi salah satu *field* pada halaman *login*.	NIP : 197812072008122001 Password : - Klik Login	Sistem akan menolak akses *login* dan menampilkan pemberitahuan "Silahkan Lengkapi Username dan Password Anda!."	Valid

3.	3-06-2019	Mengisi seluruh *field* pada halaman *login*, namun dengan NIP yang belum pernah mendaftar.	NIP : 197812072008112002 Password : - Klik Login	Sistem akan menolak akses *login* dan menampilkan pemberitahuan "Maaf Username dan Password yang Anda masukan Salah!"	Valid
4.	3-06-2019	Mengisi seluruh *field* pada halaman *login*, dengan data yang benar.	NIP : 197812072008122001 Password : resiwindi Klik Login	Sistem menerima akses *login* kemudian menampilkan halaman beranda media pembelajarn	Valid

Tabel 4.28 *Black box testing* fungsionalitas data mahasiswa pada web admin

No	Tanggal Pengujian	Skenario	Test Case	Harapan	Hasil
1.	3-06-2019	Melihat data mahasiswa	-Klik tombol Mahasiswa.	Sistem akan menerima akses data mahasiswa dan menampilkan halaman mahasiswa berupa nama, nim, jenis kelamin, agaman, tempat lahir, alamat dan jurusan	Valid
2.	3-06-2019	Menambah data mahasiswa.	-Klik tombol Mahasiswa. -Klik tombol tambah pada Mahasiswa.	Sistem akan menerima akses data mahasiswa dan menampilkan halaman tambah untuk mahasiswa	Valid
3.	3-06-2019	Mengedit data mahasiswa.	- Klik tombol Mahasiswa. -Klik tombol edit pada Mahasiswa.	Sistem akan menerima akses data mahasiswa dan menampilkan halaman edit untuk mahasiswa	Valid
4.	3-06-2019	Menghapus data mahasiswa.	- Klik tombol Mahasiswa. -Klik tombol hapus pada Mahasiswa.	Sistem akan menerima akses data mahasiswa dan menampilkan halaman hapus untuk mahasiswa	Valid

Tabel 4.29 *Black box testing* fungsionalitas data dosen pada web admin

No	Tanggal Pengujian	Skenario	Test Case	Harapan	Hasil
1.	3-06-2019	Melihat data dosen	-Klik tombol dosen.	Sistem akan menerima akses data mahasiswa dan menampilkan halaman dosen berupa nama, nip, jenis kelamin, agama, tempat lahir, alamat pendidikan terkhir, jabatan, tahun kerja dan status	Valid
2.	3-06-2019	Menambah data dosen.	-Klik tombol Dosen. -Klik tombol tambah pada Dosen.	Sistem akan menerima akses data mahasiswa dan menampilkan halaman tambah untuk dosen	Valid
3.	3-06-2019	Mengedit data dosen	- Klik tombol Dosen. -Klik tombol edit pada Dosen.	Sistem akan menerima akses data mahasiswa dan menampilkan halaman edit untuk dosen	Valid
4.	3-06-2019	Menghapus data dosen.	- Klik tombol dosen. -Klik tombol hapus pada dosen.	Sistem akan menerima akses data mahasiswa dan menampilkan halaman hapus untuk dosen	Valid

Tabel 4.30 *Black box* fungsionalitas ujian

No.	Tanggal Pengujian	Skenario	Test Case	Harapan	Hasil
1.	3-06-2019	Masuk menu ujian	Mengklik menu ujian pada halaman admin	Berhasil masuk ke halaman menu ujian	Valid
2.	3-06-2019	Melihat menu soal ujian	Mengklik pilihan soal ujian setelah masuk ke halaman menu ujian	Berhasil melihat pilihan menu soal ujian	Valid
3.	3-06-2019	Mengedit soal ujian	Mengklik tombol edit	Berhasil mengedit soal ujian	Valid

4.	3-06-2019	Menambah soal ujian	Mengklik tombol tambah soal	Berhasil menambah soal ujian	Valid
5.	3-06-2019	Menghapus soal ujian	Mengklik tombol delete	Berhasil menghapus soal ujian	Valid
6.	3-06-2019	Menampilkan soal ujian	Mengklik upload soal	Berhasil menampilkan soal ujian pada aplikasi ujian online	Vald
7.	3-06-2019	Melihat menu nilai	Mengklik pilihan nilai setelah masuk ke halaman menu ujian	Berhasil melihat dan menampilkan pilihan menu nilai	Valid
8.	3-06-2019	Menampilkankan nilai	Mengklik upload nilai	Berhasil menampilkan nilai pada aplikasi mahasiswa	Valid
9.	3-06-2019	Melihat menu pengumuman	Mnegklik pilihan pengumuman setelah masuk ke halamn menu ujian	Berhasil melihat dan masuk ke menu pilihan pengumuman	Valid
10.	3-06-2019	Mengedit menu pengumuman	Mengklik tombol edit	Berhsail mengedit pengumuman	valid
11.	3-06-2019	Menambah menu pengumuman	Mengklik tombol tambah pengumuman	Berhasil menambah pengumuman	Valid
12	3-06-2019	Menghapus menu pengumuman	Mengklik tombol hapus	Berhsil menghapus pengumuman	Valid

Setelah dilakukan dilakukan pengujian berbagai fungsionalitas dapat diketahui bahwa Aplikasi Ujian *Online* Berbasis *Android* ini layak untuk di *publish* dan pakai di Jurusan Teknik Elektro Sriwijaya. Aplikasi ini dapat bermanfaat bagi mahasiswa dalam pelaksanaan ujian secara *online* untuk menggurangi kecurangan pada saat pengerjaan ujian, juga dapat membantu dosen dalam hal mengelola soal ujian dan pengkoreksian hasil ujian. Pada sisi *user* (mahasiswa) perangkat lunak yang digunakan yaitu berbasis *mobile*, sedangkan *user* (dosen) berbasis *website*. Aplikasi *user* (mahasiswa) di*publish* melalui *Play Store*. Sebelum memulai mem*publish* aplikasi (apk) , terlebih dahulu membuat akun pengembang baru

dengan mengunjungi *Google PlayStore Developer Console*. Setelah memiliki akun tersebut klik *"Publish an Android App on Google Play"*. Selanjutnya upload aplikasi ke *google play,* lalu isi keterangan tentang aplikasi, kemudian klik *review and publish* untuk proses *review* dan otomatis *publish* setelah di *review* oleh *google play*. Aplikasi Ujian *Online* ini merupakan aplikasi gabungan yang terdiri dari Absensi, Media Pembelajaran, dan Bimbingan Tugas Akhir yang tergabung di satu sistem aplikasi Sistem Informasi Akademik Teknik Elektro Universitas Sriwijaya yang disingkat SIATEK UNSRI.

4.5.2 Tahap Pengujian dengan Kuesioner Pengujian Sistem

Untuk menguji kelayakan dan kinerja sistem aplikasi ini, penulis mengajukan pertanyaan melalui kuesioner pada *Google Form* dengan mengambil responden mahasiswa dan dosen. Diambil 20 responden dari mahasiswa dan dosen untuk mengisi kuesioner yang telah disediakan, adapun hasil kuesioner masing-masing responden akan dilampirkan pada **lampiran**. Berikut pertanyaan-pertanyaan dari kuesioner tersebut terdiri dari :

1. Dengan adanya aplikasi ujian online berbasis android dapat membantu Anda dalam pengerjaan ujian?

   ○ Sangat Setuju

   ○ Setuju

   ○ Kurang Setuju

   ○ Tidak Setuju

   ○ Sangat Tidak Setuju

2. Apakah aplikasi ujian online berbasi android berjalan sesuai fungsinya dengan tepat?

   ○ Sangat Setuju

   ○ Setuju

- ○ Kurang Setuju
- ○ Tidak Setuju
- ○ Sangat Tidak Setuju

3. Proses pengisian jawaban menjadi lebih mudah, baik dalam pengisian jawaban esai maupun input gambar ?
   - ○ Sangat Setuju
   - ○ Setuju
   - ○ Kurang Setuju
   - ○ Tidak Setuju
   - ○ Sangat Tidak Setuju

4. Dalam implementasi aplikasi ujian online berbasis android mudah digunakan?
   - ○ Sangat Setuju
   - ○ Setuju
   - ○ Kurang Setuju
   - ○ Tidak Setuju
   - ○ Sangat Tidak Setuju

5. Tampilan aplikasi ujian online berbasis android menarik dan sesuai dengan harapan Anda?
   - ○ Sangat Setuju
   - ○ Setuju
   - ○ Kurang Setuju
   - ○ Tidak Setuju

- ○ Sangat Tidak Setuju

6. Secara kesuluruhan Aplikasi jian Online Berbasis Android di Jurusan Teknik Elektro Universitas Sriwijay sudah efektif?
   - ○ Sangat Setuju
   - ○ Setuju
   - ○ Kurang Setuju
   - ○ Tidak Setuju
   - ○ Sangat Tidak Setuju

Analisa yang didapat dari pengumpulan data dari kuesioner dimana terdapat 6 soal pertanyaan objektif yang diisi oleh mahasiswa jurusan Teknik Elektro Universitas Sriwijya.

1. Pada pertanyaan pertama yang diajukan penulis, membahas bagaimana pendapat responden dengan adanya aplikasi ujian online berbasis android apakah membantu pada saat pengerjaan atau pelaksanaan ujian. Hasil data kuesioner yang didapat dari 21 responden 42,9 % menyatakan sangat setuju, 52,4 % menyatakan setuju, dan 4,7 % menyatakan tidak setuju. Dari persentase yang diatas, dapat kita simpulkan bahwa aplikasi ujian online berbasis android ini membantu dalam pelaksanaan ujian.

2. Pada pertanyaan kedua yang diajukan penulis, membahas apakah aplikasi ujian online berbasis android berjalan sesuai fungsinya dengan tepat. Hasil data kuesioner yang didapat dari 21 responden 85,7 % menyatakan setuju, 9,5 % menyatakan sangat setuju, dan 4,8 % kurang setuju. Dapat kita simpulkan bahwa aplikasi ujian online ini sudah berjalan sesuai dengan fungsinya.

3. Pada pertanyaan kelima yang diajukan penulis, membahas apakah proses pengisian jawaban maupun input gambar pada saat ujian online menjadi lebih mudah. Hasil data kuesioner yang didapat dari 21 responden 75% menyatakan setuju, 15% menyatakan sangat setuju, 5% menyatakan sangat kurang setuju, dan 5% menyatakan tidak setuju. Dapat disimpulkan dari hasil kuesioner bahwa pada saat pengisian jawaban baik jawaban esai maupun input gambar berjalan lebih mudah.

4. Pada pertanyaan ketujuh yang diajukan penulisan, membahas apakah aplikasi ujian online berbasis android pada implementasinya mudah digunakan. Hasil data kuesioner yang didapat dari 21 responden 66.7 % menyatakan setuju, 28.6% menyatakan sangat setuju, dan 4.8% menyatakan kurang setuju. Dapat disimpulkan dari hasil kuesioner bahwa aplikasi ujian online berbasis android dalam implementasinya sudah dapat dikatakan

5. Pada pertanyaan kedelapan yang diajukan penulis, memahas apakah tampilan pada aplikasi ujian online berbasis android menarik dan sesuai dengan harapan. Dari hasil kuesioner yang didapat dari 21 responden 66.7% menyatakan setuju, 23.8% menyatakan sangat setuju, dan 9.5% menyatakan kurang setuju. Dapat disimpulkan bahwa tampilan aplikasi sudah cukup menarik dan sesuai dengan yang diharapkan.

6. Pada pertanyaan kesembilan yang diajukan penulis, membahas aplikasi secara keseluruhan sudah efektif. Dari hasil kuesioner yang didapat dari 21 responden 66.7% menyatakan setuju dan 33.3% menyatan sangat setuju. Dapat disimpulkan bahwa aplikasi ujian online berbasis android di Jurusan Teknik Elektro Universitas Sriwijaya sudah cukup efektif.

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Adapun kesimpulan yang didapat dari penulisan penelitian ini adalah sebagai berikut :

1. Aplikasi ujian *online* berbasis *android* di Teknik Elektro Universitas Sriwijaya efektif digunakan untuk mempermudah mahasiswa maupun dosen. Ujian *online* yang dapat diakses dengan mudah melalui perangkat *mobile* berbasis *android* bagi mahasiswa dan dapat digunakan dimana saja yang memiliki akses jaringan internet. Serta terbanggunnya sebuah sistem untuk mengontrol dan memanajemen pelaksanaan ujian yang dapat diakses menggunakan *website*.
2. Aplikasi ujian *online* berbasis *android* ini digunakan untuk menggantikan metode ujian yang selama ini digunakan yaitu metode konvensional. Metode ujian konvensial masih banyak memiliki kekurangan antara lain pemborosan kertas juga terbatas ruang dan waktu.
3. Jenis soal ujian pada aplikasi ujian *online* berbasis *android* di Teknik Elektro Universitas Sriwijaya ini adalah esai. Penggunaan soal berbentuk esai dapat melatih mahasiswa untuk berfikir lebih kritis, logis dan cerdas.
4. Dalam pengembangan aplikasi ujian *online* berbasis *android,* penulis menggunakan metode pengembangan RAD (*Rapid Application Development*), semua tahapan dari metode RAD ini telah berhasil dalam proses pembangunan aplikasi ujian *online* berbasis *android* di Teknik Elektro Universitas Sriwijaya. Pada dosen dan admin sistem yang dihasilkan berupa *website.* Penggunaan *website* untuk dosen dinilai lebih efektif, seperti kegiatan meng*upload* dan mengkoreksi soal ujian.
5. Hasil akhir dari perancangan tersebut sudah terbangun sebuah aplikasi yang merupakan aplikasi gabungan yang terdiri dari Absensi, Media

Pembelajaran, dan Bimbingan Tugas Akhir yang tergabung di satu sistem aplikasi Sistem Informasi Akademik Teknik Elektro Universitas Sriwijaya yang disingkat SIATEK UNSRI

6. Sistem aplikasi *mobile* sudah dapat di*download* melalui *Google Play Store*.

5.2 Saran

Adapun saran yang dapat diberikan berdasarkan kesimpulan di atas adalah sebagai berikut :

1. Untuk pengembangan aplikasi ujian *online* berbasis *android* di Teknik Elektro Universitas Sriwijaya disarankan dapat menambah beberapa *fiture*, seperti : tugas, pilihan soal pilihan ganda, Kartu Rencana Studi (KRS), dan transkrip nilai.
2. Diharapkan untuk kedepannya aplikasi ujian *online* berbasis *android* ini dapat digunakan pada perangkat *Windows* dan *IOS*.

LAPORAN SKRIPSI.pdf

ORIGINALITY REPORT

14%	5%	2%	13%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Sriwijaya University Student Paper	9%
2	sir.stikom.edu Internet Source	2%
3	repository.usu.ac.id Internet Source	1%
4	Submitted to Universitas Brawijaya Student Paper	1%
5	kancilsquad.blogspot.com Internet Source	1%
6	rizqiaawulansari.wordpress.com Internet Source	1%

Exclude quotes	On	Exclude matches	< 1%
Exclude bibliography	On		