

BAB IV. ANALISIS DAN PERANCANGAN SISTEM

4.1.1 Analisis Sistem

Pada penelitian ini menghasilkan aplikasi yang disebut VIAT-MAP yang menerapkan model argumentasi Toulmin untuk membentuk pemahaman yang mendalam akan sebuah teks/kalimat yang di sajikan dengan menggunakan strategi grafis dengan menambahkan umpan balik (*feedback*) di dalamnya guna membantu penalaran siswa dalam membangun pemahaman yang kuat dari sebuah kalimat.

4.1.2 Analisis Kebutuhan Pengguna

Analisa digunakan untuk mengetahui aktor yang terdapat pada Aplikasi ini. Pada aplikasi ini memiliki 2 aktor dimana setiap aktor memiliki peran nya masing-masing yang terbagi menjadi:

Tabel 4.1 Tabel Analisis Aktor

No.	Aktor	Keterangan
1.	Dosen	Dosen memiliki peran untuk membuat soal latihan beserta umpan balik (<i>feedback</i>) dari soal tersebut, membuat akun untuk mahasiswa serta melihat hasil dari latihan para siswa.
2.	Mahasiswa	Mahasiswa memiliki peran untuk melakukan pengerjaan soal serta melihat hasil dari pengerjaan yang telah dilakukan.

4.1.3 Analisis Kebutuhan Sistem

Kebutuhan pada perangkat untuk membangun sistem pada penelitian kali ini akan diuraikan pada tabel di bawah ini:

A. Kebutuhan Perangkat Keras

Analisa kebutuhan perangkat keras (*Hardware*) yang digunakan pada pembuatan Aplikasi VIAT-MAP ini disajikan pada tabel dibawah ini:

Tabel 4.2 Kebutuhan Perangkat Keras

No	Perangkat
1	Processor (Intel Core i7)
2	RAM (8 GB)
3	SSD (256 GB)
4	Keyboard
5	Monitor
6	Mouse

B. Kebutuhan Perangkat Lunak

Analisis kebutuhan perangkat lunak (*Software*) pada Aplikasi VIAT-MAP yang digunakan dalam pembuatan aplikasi ini disajikan pada tabel dibawah ini:

Tabel 4.3 Kebutuhan Perangkat Lunak

No	Perangkat
1	Windows
2	PHPStrom
3	Browser (Chrome)
4	Laragon
5	Unity
6	Visual Studio

4.2 Perancangan Sistem

4.2.1 Kebutuhan Fungsional

Analisa Kebutuhan Fungsional merupakan uraian akan fungsionalitas dari fitur – fitur yang terdapat pada system. Berikut uraian kebutuhan fungsional yang terdapat pada sistem ini pada masing-masing aktor:

Tabel 4.4 Tabel Kebutuhan Fungsional Dosen

No.	Fitur	Keterangan
1.	Membuat Soal Latihan	Dosen dapat membuat soal latihan dari materi pembelajaran yang konstruksikan menjadi model argumentasi Toulmin.
2.	Membuat Feedback dari soal	Dosen dapat menambahkan feedback pada setiap kemungkinan pilihan jawaban salah yang dibuat oleh mahasiswa.
3.	Membuat Akun Mahasiswa	Dosen dapat menambahkan data Mahasiswa agar terdaftar ke dalam sistem.
4.	Melihat hasil latihan	Dosen dapat melihat hasil latihan dari mahasiswa yang telah selesai mengerjakan latihan yang diberikan.

Tabel 4.5 Tabel Kebutuhan Fungsional Mahasiswa

No.	Fitur	Keterangan
1.	Mengerjakan Latihan	Mahasiswa dapat mengerjakan latihan yang telah dibuat sebelumnya oleh dosen.
2.	Mendapaptkan <i>Feedback</i>	Mahasiswa mendapatkan <i>feedback</i> dari pilihan jawaban mereka pada saat salah dalam menjawab pertanyaan.
3.	Melihat hasil latihan	Mahasiswa dapat melihat hasil dari pekerjaan mereka.

4.2.2 Kebutuhan Non-Fungsional

Kebutuhan Non-Fungsional adalah kemampuan yang dimiliki oleh aplikasi pada penelitian ini, antar lain sebagai berikut:

Tabel 4.6 Tabel Kebutuhahan Non-Fungsional

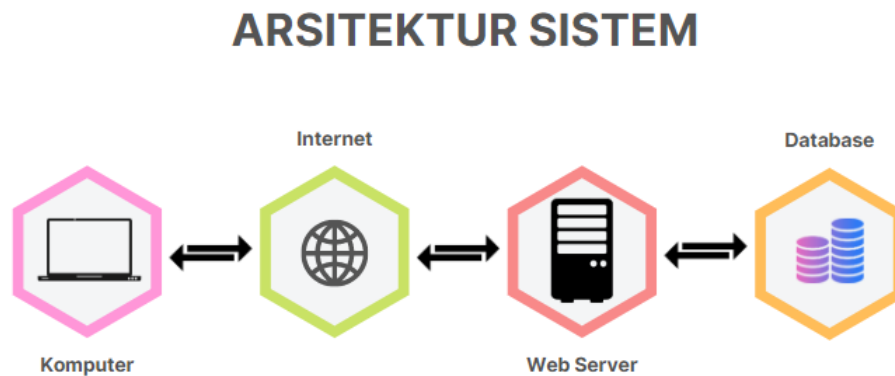
No.	Fitur	Keterangan
1.	Usability	Sistem dapat digunakan pada lingkungan kampus guna membantu proses pembelajaran Bahasa Inggris khususnya materi Reading Comprehension
2.	Portability	Sistem berbasis <i>Web Application</i> yang dijalankan dengan menggunakan browser.
3.	Supportbility	Sistem ini dapat digunakan dengan koneksi internet.
4.	Reliability	Dengan sistem ini diharapkan dapat meningkatkan <i>Reading Comprehension</i> pada pembelajaran Bahasa Inggris.

Pengertian dari setiap jenis kebutuhan adalah sebagai berikut:

1. *Usability* merupakan kebutuhan non fungsional yang berhubungan dengan pemakaian sistem oleh pengguna.
2. *Portability* merupakan bagaimana sistem diakses secara mudah oleh pengguna, khususnya dalam faktor lokasi pengaksesan, perangkat, waktu dan teknologi yang digunakan.
3. *Supportability* merupakan kebutuhan yang berfokus pada dukungan pemakaian sistem.
4. *Reliability* merupakan kebutuhan yang berfokus pada keandalan suatu sistem.

4.2.3 Arsitektur Sistem

Pada bagian ini akan menjelaskan secara umum arsitektur akan proses yang berjalan didalam Aplikasi VIAT-MAP:



Gambar 4.1 Arsitektur Sistem

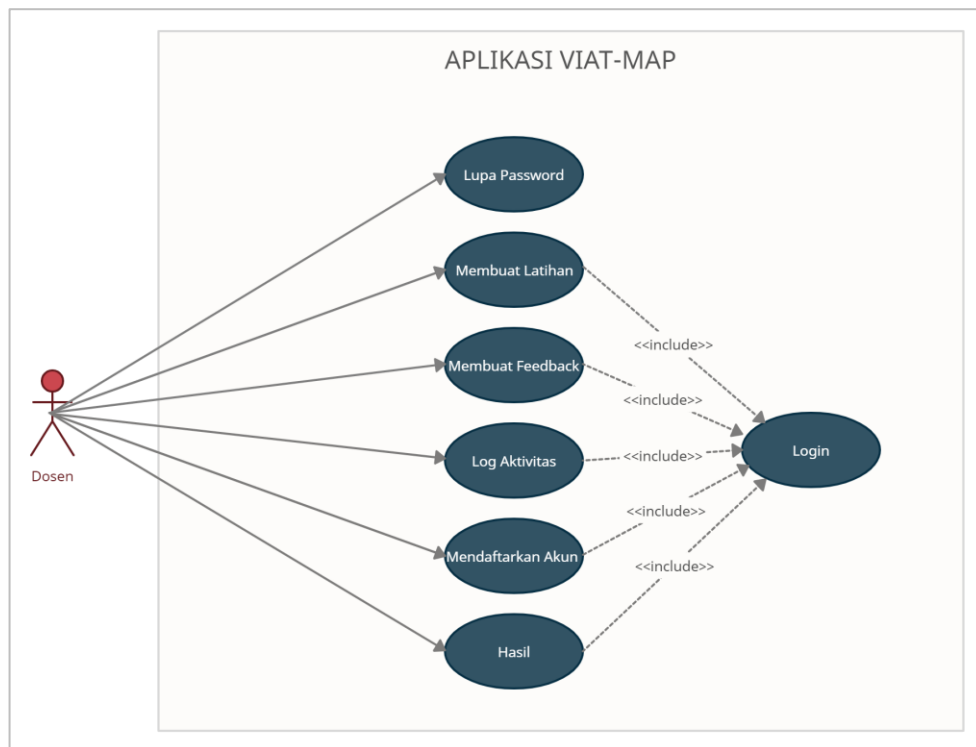
Pada Penelitian ini pada aplikasi yang digunakan dapat diakses dengan perangkat komputer yang terkoneksi pada internet sehingga terhubung dengan web service, setelah terhubung maka DB Server dapat diakses dan digunakan untuk menjalankan aplikasi ini.

4.2.4 Use Case Diagram

Use Case Diagram merupakan sebuah pemodelan yang akan memaparkan alur pada alur akan sistem yang dibuat serta dapat mengetahui interaksi apa saja yang terdapat pada aktor – aktor yang ada di dalam sistem.

Berikut definisi dari use case diagram di atas:

1. Dosen



Gambar 4.2 Use Case Diagram Dosen

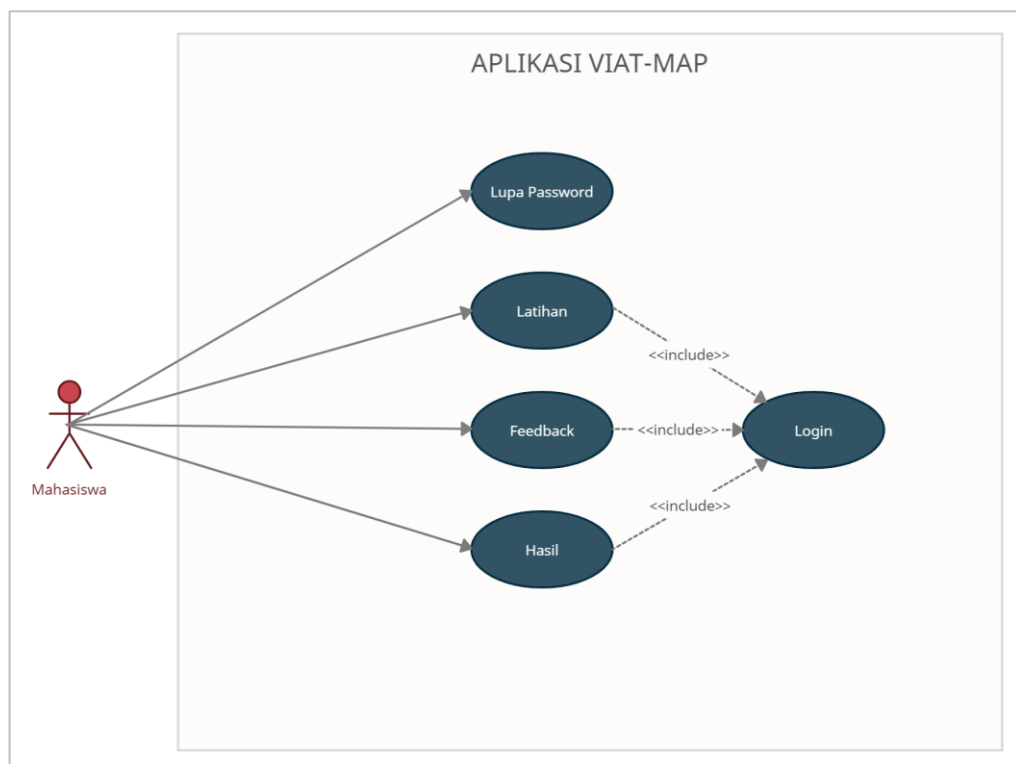
Penejelasan mengenai Use Case Diagram pada gambar di atas diuraikan pada tabel dibawah ini:

Tabel 4.7 Definisi Use Case Dosen

No.	Use Case	Deskripsi
1.	Login	Menginputkan NIM/Email dan Password
2.	Lupa Password	Menginputkan email yang sudah terlebih dahulu terdaftar ke dalam sistem kemudian sistem akan menampilkan email dan password pengguna untuk melakukan autentikasi.
3.	Melihat Hasil Latihan	Mahasiswa dapat melihat hasil latihan dan berapa lama waktu yang mereka butuhkan untuk menjawab keseluruhan soal.

4.	Mendaftarkan siswa	Dosen dapat menambahkan data siswa dengan melengkapi form yang tersedia.
5.	Membuat Soal dan Jawaban	Dosen membuat soal (claim) serta membuat pilihan jawaban akan peta bacaan yakni warrant dan ground.
6.	Membuat Feedback	Setelah membuat soal dosen akan membuat feedback akan kemungkinan pilihan yang salah.
7.	Aktivitas Log	Dosen dapat melihat aktivitas dari siswa, berapa kali siswa membuat percobaan menjawab sebelum akhirnya benar dalam menjawab

2. Mahasiswa



Gambar 4.3 Use Case Diagram Mahasiswa

Penejelasan mengenai Use Case Diagram pada gambar di atas diuraikan pada tabel dibawah ini:

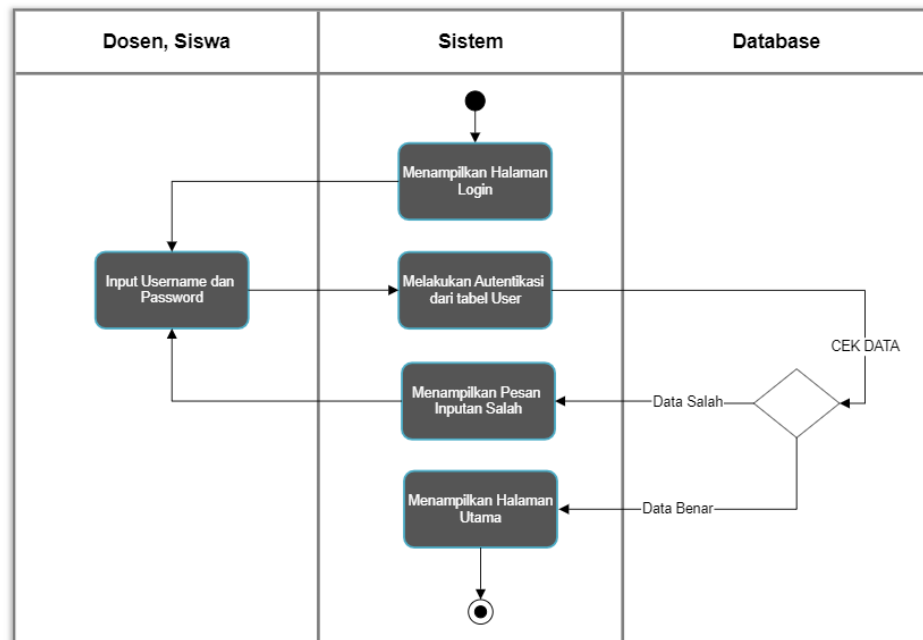
Tabel 4.8 Definisi Use Case Mahasiswa

No.	Use Case	Deskripsi
1.	Login	Menginputkan NIM/Email dan Password
2.	Lupa Password	Menginputkan email yang sudah terlebih dahulu terdaftar ke dalam sistem kemudian sistem akan menampilkan email dan password pengguna untuk melakukan autentikasi.
3.	Mengerjakan Latihan	Menjawab soal dengan menyusun peta bacaan dari claim yang disediakan yang dilakukan dengan melakukan <i>drag and drop</i> pilihan ground dan warrant ke bidang yang disediakan.
4.	Feedback	Panel feedback akan muncul jika susunan jawaban yang dibuat salah.
5.	Melihat hasil latihan	Mahasiswa dapat melihat hasil latihan dan berapa lama waktu yang mereka butuhkan untuk menjawab keseluruhan soal.

4.2.5 Activity Diagram

1. Login

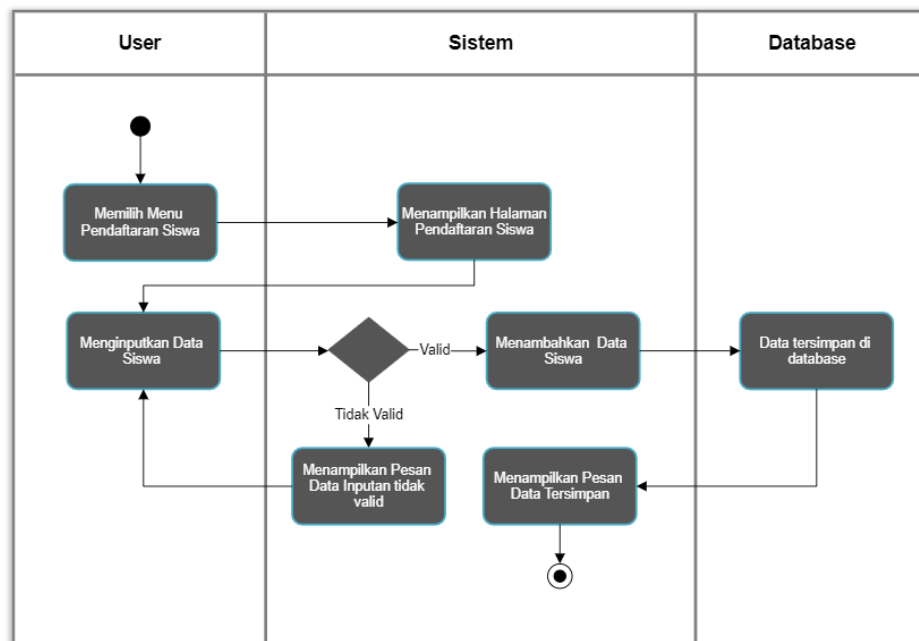
Dalam menjalankan aplikasi ini, halaman yang akan dimuat pertama yakni halaman login, disini pengguna diminta menginputkan *username* dan *password* yang kemudian sistem akan mengirim *request* untuk mengecek data dalam database. Ketika data yang diinputkan salah / tidak valid maka akan menampilkan pesan *username* atau *password* salah dan ketika data yang diinputkan benar atau terdaftar dalam database maka sistem akan menampilkan masing – masing halaman utama sesuai dengan role yang ada.



Gambar 4.4 Activity Diagram Login

2. Membuat akun

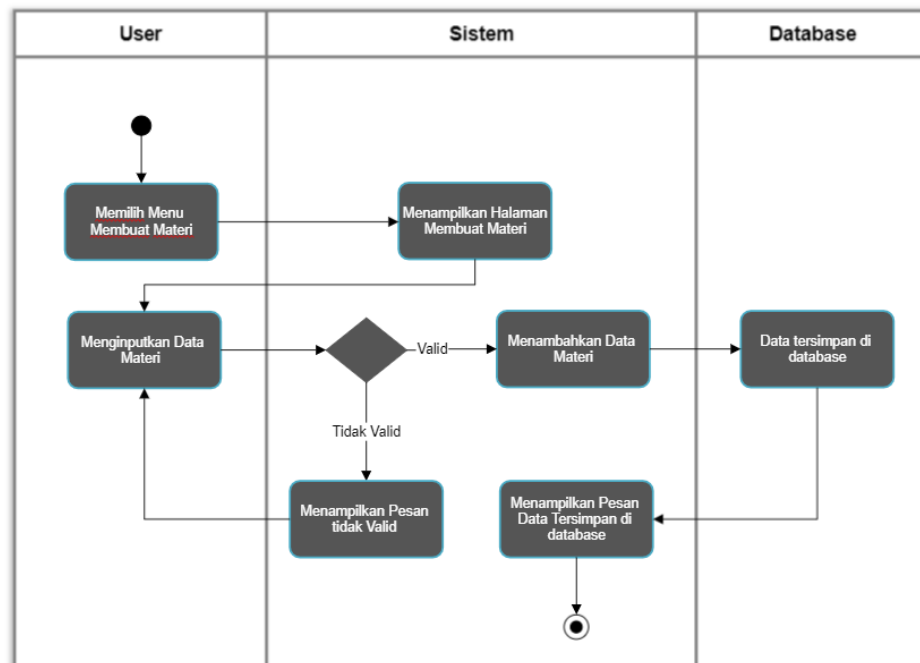
Untuk mendaftarkan siswa dosen akan memilih menu pendaftaran siswa (*Enroll student*) pada halaman utama, kemudian sistem akan menampilkan halaman pendaftaran siswa dimana disana terdapat bidang-bidang informasi siswa yang harus diinputkan, pada saat inputan tidak valid maka akan muncul pesan peringatan dan pengguna diminta untuk menginputkan data dengan benar dan lengkap, inputan yang dimasukkan dinilai valid apabila telah memasukkan keseluruhan data pada form yang tersedia. Jika data yang diinputkan telah valid maka sistem akan menambahkan data mahasiswa dimana data tersebut akan di kirim kedalam database, selanjutnya sistem akan menampilkan data mahasiswa telah dibuat.



Gambar 4.5 *Activity Diagram* Membuat Materi Latihan

3. Membuat Materi Latihan

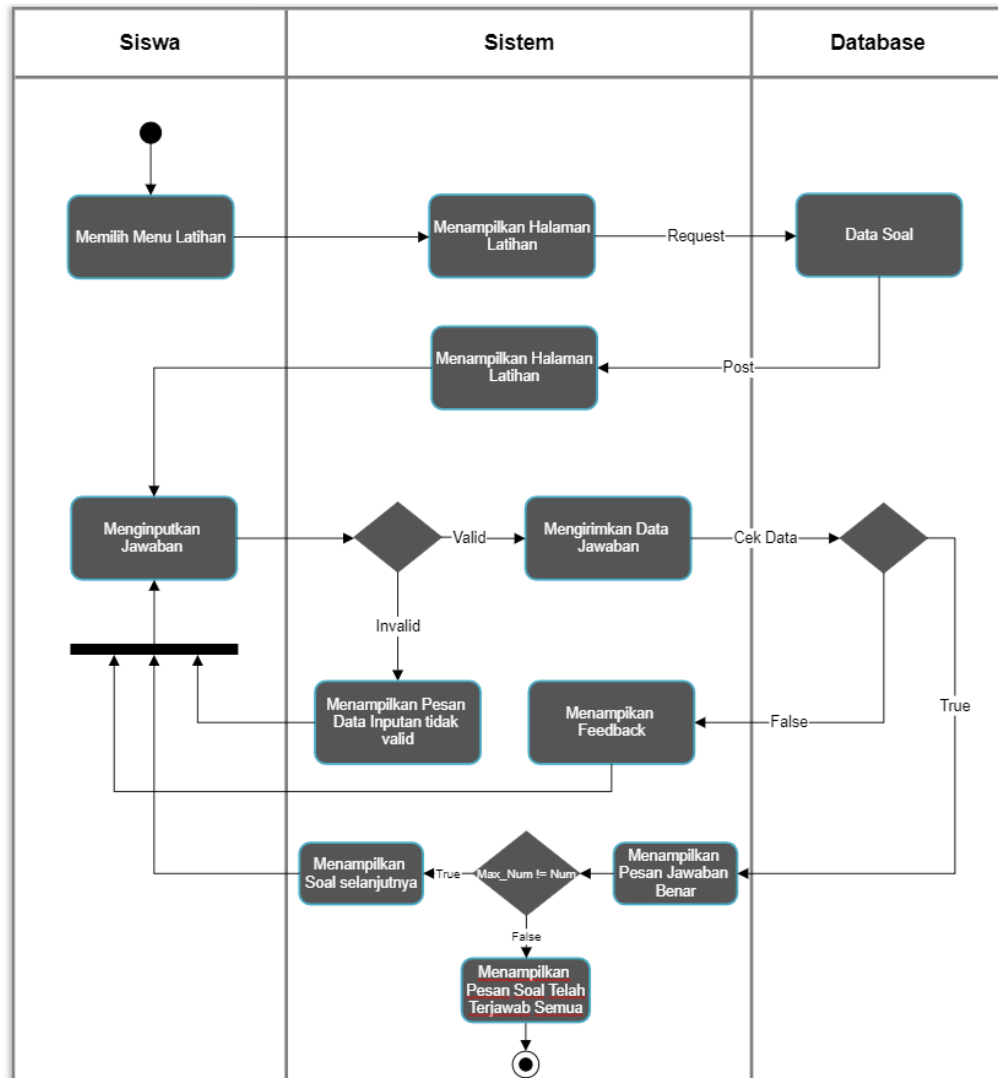
Untuk membuat materi maka disini memilih menu membuat materi pada halaman utama dosen, kemudian sistem akan menampilkan halaman membuat materi yang dimana terdapat beberapa bidang yang harus diinputkan yakni soal, pilihan jawaban serta umpan balik yang akan muncul, data akan dihitung valid jika inputan telah sepenuhnya diisi dengan lengkap, apabila data yang diinputkan tidak lengkap maka sistem akan menampilkan pesan bahwa inputan tidak valid, jika inputan telah lengkap maka sistem akan menambahkan material dimana data tersebut di kirim kedalam database dan sistem akan merespon dengan menampilkan pesan bahwa data telah dibuat.



Gambar 4.6 Activity Diagram Membuat Materi Latihan

4. Proses latihan

Untuk mengerjakan Latihan, mahasiswa akan memilih menu latihan pada halaman utama mahasiswa, kemudian sistem akan menampilkan halaman latihan dimana sistem akan melakukan request data soal ke dalam yang akan ditampilkan di dalam halaman latihan kemudian mahasiswa akan menginputkan jawaban dengan melakukan *drag* jawaban dan meletakkannya pada bidang jawaban yang disediakan, kemudian sistem akan melakukan pengecekan apabila inputan mahasiswa valid, jika inputan mahasiswa tidak valid maka sistem akan menampilkan pesan bahwa inputan tidak valid dan mahasiswa diminta melakukan pengisian jawaban kembali, jika telah valid maka sistem akan melakukan pengecekan data jawaban yang terdapat pada database, jika jawaban siswa salah maka sistem akan menampilkan *feedback* kemudian siswa akan diminta untuk menginputkan jawaban kembali hingga benar, pada saat jawaban siswa benar maka sistem akan merespon dengan menampilkan pesan jawaban benar lalu akan lanjut ke soal selanjutnya jika soal belum terjawab semua dan jika soal telah terjawab maka sistem akan menampilkan bahwa siswa telah menyelesaikan latihan.

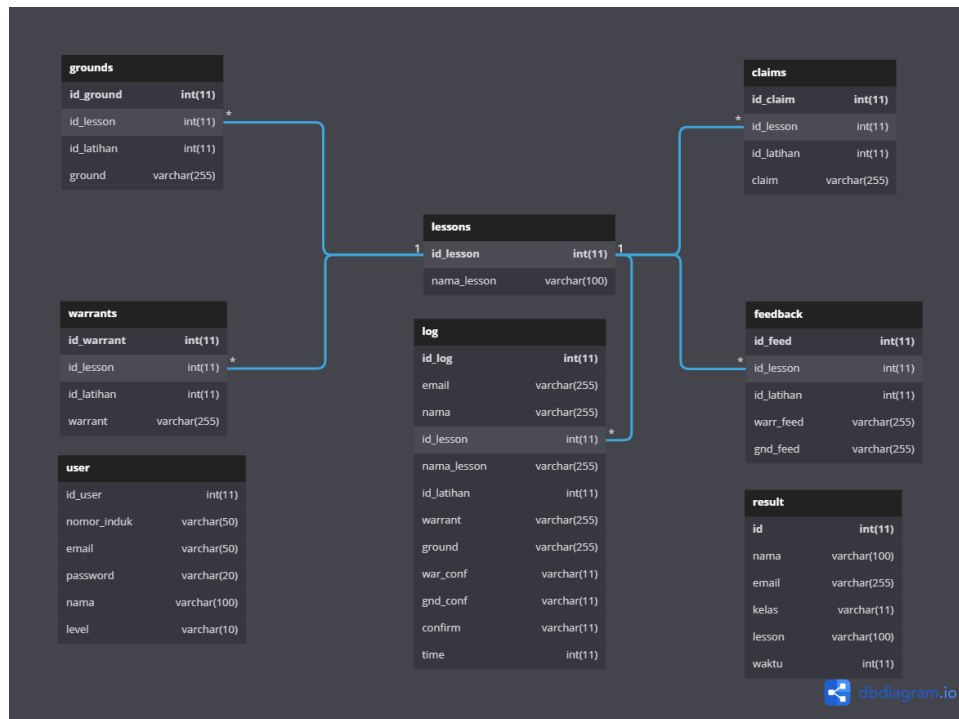


Gambar 4.7 Activity Diagram Proses Latihan

4.3 Perancangan Database

4.3.1 Entity Relation Diagram (ERD)

Pada gambar 4. Dibawah merupakan Entity Relation Diagram perancangan Aplikasi VIAT-MAP.



Gambar 4.8 Entity Relation Diagram Aplikasi

Berdasarkan *Entity Relation Database* tabel-tabel akan digunakan untuk mendukung proses dari pengolahan data yang terdapat pada sistem. Berikut penjelasan secara umum akan penggunaan tabel yang digunakan:

1. Tabel user

Tabel yang menyimpan data *user* seperti nomor induk, email, nama, password, serta level yang digunakan menjadi identitas pembeda antar aktor.

2. Tabel lessons

Tabel yang digunakan untuk menyimpan data lessons yang telah dibuat.

3. Tabel claims

Tabel yang digunakan untuk menyimpan data claims yang telah dibuat, seperti id_claim, id_lessons, id_latihan, serta claim yang menampung isi dari claim itu sendiri.

4. Tabel grounds

Tabel yang digunakan untuk menyimpan data ground yang telah dibuat, seperti id_ ground, id_lessons, id_latihan, serta claim yang menampung isi dari ground itu sendiri.

5. Tabel warrants

Tabel yang digunakan untuk menyimpan data warrants yang telah dibuat, seperti id_ warrant, id_lessons, id_latihan, serta claim yang menampung isi dari warrant itu sendiri.

6. Tabel feedback

Tabel yang digunakan untuk menampung feedback dari setiap soal (*claims*) yang dibuat.

7. Tabel result

Tabel yang digunakan untuk menampung informasi hasil yang diperoleh oleh mahasiswa.

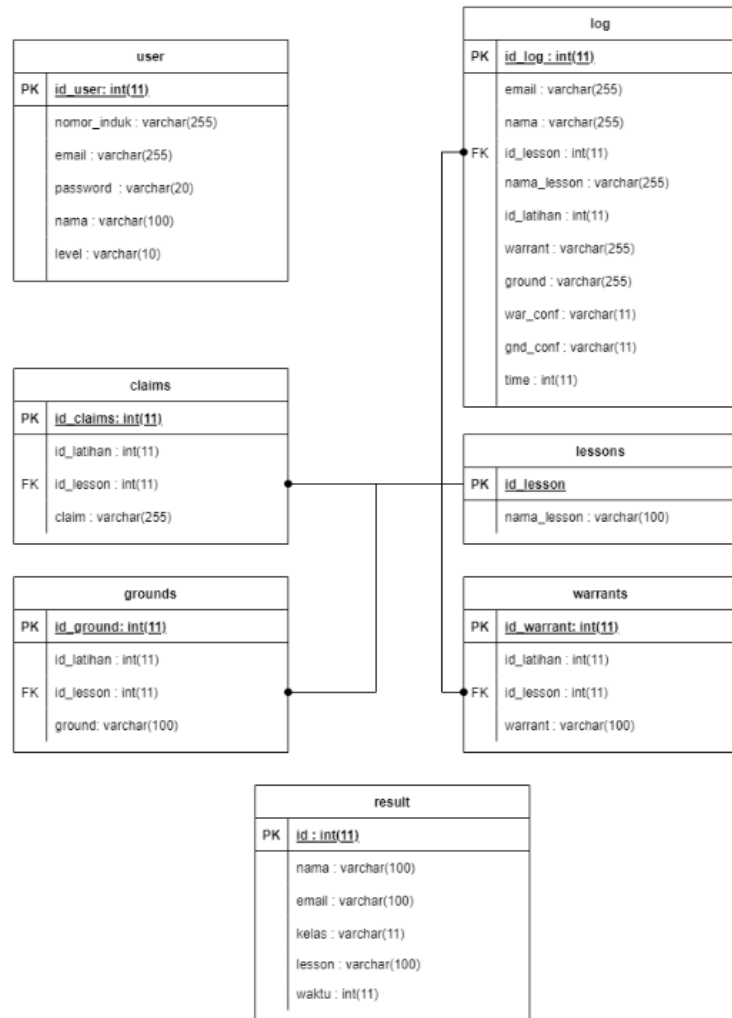
8. Tabel log

Tabel yang digunakan untuk menampung data aktivitas yang dilakukan mahasiswa pada saat menjawab setiap soal yang diberikan.

Pada gambar diatas juga terdapat hubungan antar tabel pada *Entity Relation Diagram* (ERD) yang memiliki penjelasan sebagai berikut:

1. Entity Lesson mempunyai Relasi One to Many terhadap entity grounds
2. Entity Lesson mempunyai Relasi One to Many terhadap entity warrants
3. Entity Lesson mempunyai Relasi One to Many terhadap entity claims
4. Entity Lesson mempunyai Relasi One to Many terhadap entity feedback
5. Entity Lesson mempunyai Relasi One to Many terhadap entity log

4.3.2 Class Diagram



Gambar 4.9 *Class Diagram* Aplikasi VIAT-MAP

4.3.3 Perancangan Database

1. Tabel User

Tabel User digunakan untuk menampung informasi Pengguna.

Tabel 4.9 Tabel User

Nama Field	Tipe	Keterangan
id_user	Int (11)	PrimaryKey pada tabel user yang menjadi identitas pada data user.
nomor_induk	Varchar (50)	Kolom untuk menyimpan data nomor induk

email	Varchar (50)	Kolom untuk menyimpan data email
password	Varchar (20)	Kolom untuk menyimpan data password
nama	Varchar (100)	Kolom untuk menyimpan data nama pengguna
level	Varchar (10)	Kolom untuk menyimpan tingkatan level pengguna.

2. Tabel Lessons

Tabel *Lessons* digunakan untuk menampung data akan *lessons* yang telah dibuat.

Tabel 4.10 Tabel *Lessons*

Nama Field	Tipe	Keterangan
id_lesson	Int (11)	PrimaryKey pada tabel lessons yang menjadi identitas pada tabel lessons.
nama_lesson	Varchar (100)	Kolom untuk menyimpan data nama lessons yang dibuat.

3. Tabel Claims

Tabel *Claims* digunakan untuk menampung data akan *claims* yang telah dibuat yang terhubung oleh tabel *lessons*.

Tabel 4.11 Tabel *Claims*

Nama Field	Tipe	Keterangan
id_claim	Int (11)	PrimaryKey pada tabel claim yang menjadi identitas pada tabel claim.
id_lesson	Int (11)	ForeignKey dari tabel lessons,
id_latihan	Int (11)	Kolom untuk menyimpan data id_latihan
claim	Varchar (255)	Kolom untuk menyimpan data claim

4. Tabel Grounds

Tabel grounds digunakan untuk menampung data akan grounds yang telah dibuat yang terhubung oleh tabel grounds.

Tabel 4.12 Tabel *Grounds*

Nama Field	Tipe	Keterangan
id_ground	Int (11)	PrimaryKey pada tabel grounds yang menjadi identitas pada tabel grounds.
id_lesson	Int (11)	ForeignKey dari tabel lessons,
id_latihan	Int (11)	Kolom untuk menyimpan data id_latihan
grounds	Varchar (255)	Kolom untuk menyimpan data grounds

5. Tabel Warrants

Tabel warrants digunakan untuk menampung data akan warrants yang telah dibuat yang terhubung oleh tabel warrants.

Tabel 4.13 Tabel *Warrants*

Nama Field	Tipe	Keterangan
id_warrant	Int (11)	PrimaryKey pada tabel warrants yang menjadi identitas pada tabel warrants.
id_lesson	Int (11)	ForeignKey dari tabel lessons.
id_latihan	Int (11)	Kolom untuk menyimpan data id_latihan
warrants	Varchar (255)	Kolom untuk menyimpan data warrants

6. Tabel Feedback

Tabel yang digunakan untuk menampung feedback dari setiap soal (claims) yang dibuat.

Tabel 4.14 Tabel *Feedback*

Nama Field	Tipe	Keterangan
id_feed	Int (11)	PrimaryKey pada tabel feedback yang menjadi identitas pada tabel feedback.
id_lesson	Int (11)	ForeignKey dari tabel lessons,
id_latihan	Int (11)	Kolom untuk menyimpan data id_latihan

war_feed	Varchar (255)	Kolom untuk menyimpan data dari feedback pada warrant
Gnd_feed	Varchar (255)	Kolom untuk menyimpan data dari feedback pada grounds

7. Tabel Result

Tabel yang digunakan untuk menampung informasi hasil yang diperoleh oleh mahasiswa.

Tabel 4.15 Tabel *Result*

Nama Field	Tipe	Keterangan
id	Int (11)	Primary Key pada tabel Result yang menjadi identitas pada tabel Result.
email	Varchar (100)	Kolom untuk menyimpan data email dari user.
nama	varchar (255)	Kolom untuk menyimpan data nama dari user.
kelas	Int (11)	Kolom untuk menyimpan data kelas dari user.
lesson	Varchar (100)	Kolom untuk menyimpan data lessons yang dikerjakan oleh user.
waktu	Int (11)	Kolom untuk menyimpan data waktu pengerjaan yang ditempuh untuk menyelesaikan lessons yang diberikan.

8. Tabel Log

Tabel yang digunakan untuk menampung data aktivitas yang dilakukan mahasiswa pada saat menjawab setiap soal yang diberikan

Tabel 4.16 Tabel *Log*

Nama Field	Tipe	Keterangan
id_log	Int (11)	Primary Key pada tabel Log yang menjadi identitas pada tabel Log.

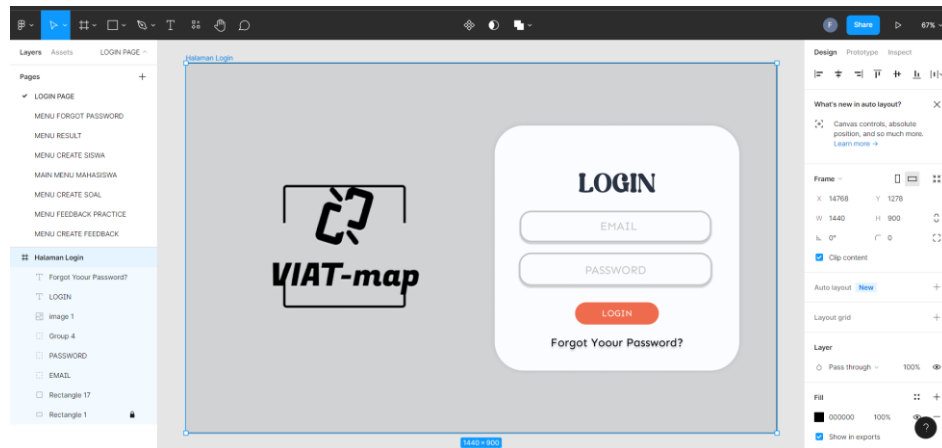
email	Varchar (255)	Kolom untuk menyimpan data email dari user.
nama	Varchar (255)	Kolom untuk menyimpan data nama dari user.
id_lesson	Int (11)	Kolom untuk menyimpan data id_lessons yang dikerjakan user.
nama_lesson	Varchar (255)	Kolom untuk menyimpan data nama_lessons yang dikerjakan user.
id_latihan	Int (11)	Kolom untuk menyimpan data id_latihan yang dikerjakan user.
warrant	Varchar (255)	Kolom untuk menyimpan data pilihan warrant yang dipilih oleh user.
ground	Varchar (255)	Kolom untuk menyimpan data pilihan warrant yang dipilih oleh user.
time	Int (11)	Kolom untuk menyimpan data waktu yang dibutuhkan user dalam menjawab 1 soal.

4.4 Perancangan Design Antarmuka

Design Interface atau desain antar muka terfokus pada penampilan Aplikasi yang dilihat oleh pengguna. Tujuan dari desain antarmuka sendiri yakni dapat meningkatkan pengalaman pengguna terhadap perangkat serta perangkat dapat digunakan dengan mudah.

4.4.1 Tampilan Menu Login

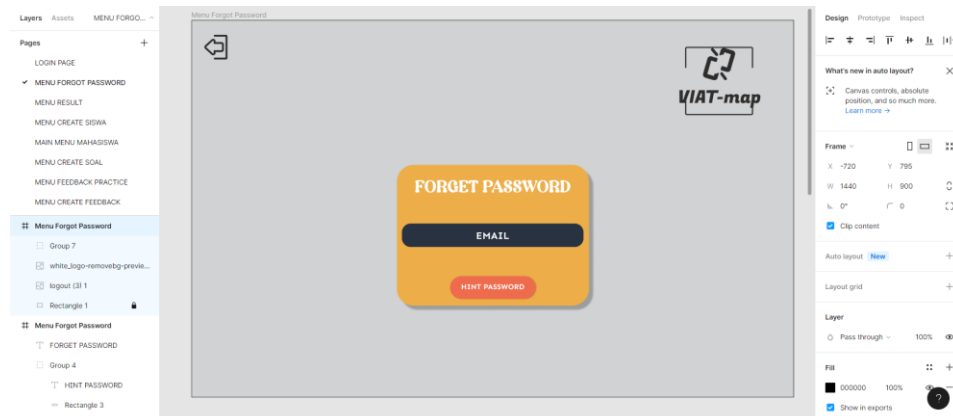
Halaman ini merupakan halaman pertama yang akan dijumpai oleh pengguna. Pada halaman ini pengguna harus menginputkan *email* dan *password* yang telah terdaftar agar dapat memasuki menu utama pada masing-masing pengguna. Berikut desain halaman *login* aplikasi:



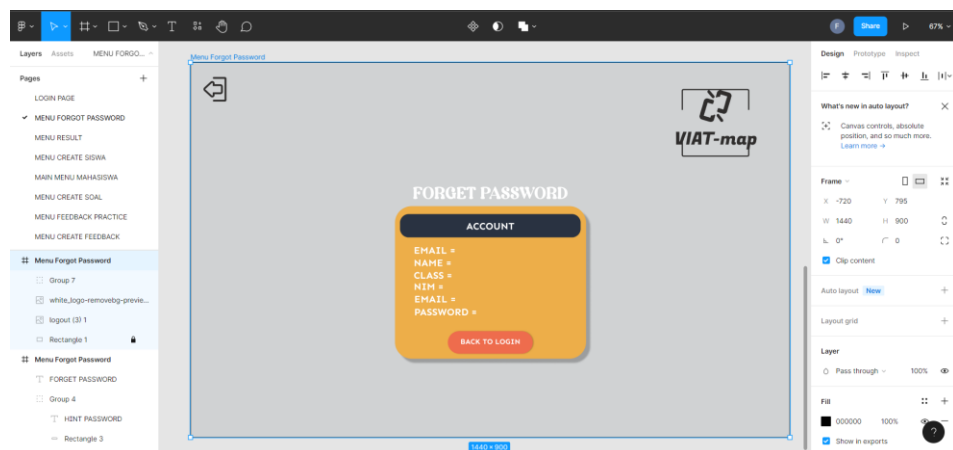
Gambar 4.10 Desain Tampilan Halaman *Login*

4.4.2 Tampilan Menu Lupa Password

Apabila pengguna lupa akan password yang dimilikinya maka pengguna dapat menekan tombol “*Forget Your Password?*” dan akan diarahkan ke halaman “*Forgot Password*”. Pada halaman ini pengguna diminta memasukkan email yang dimilikinya untuk melihat informasi dari akun pengguna.



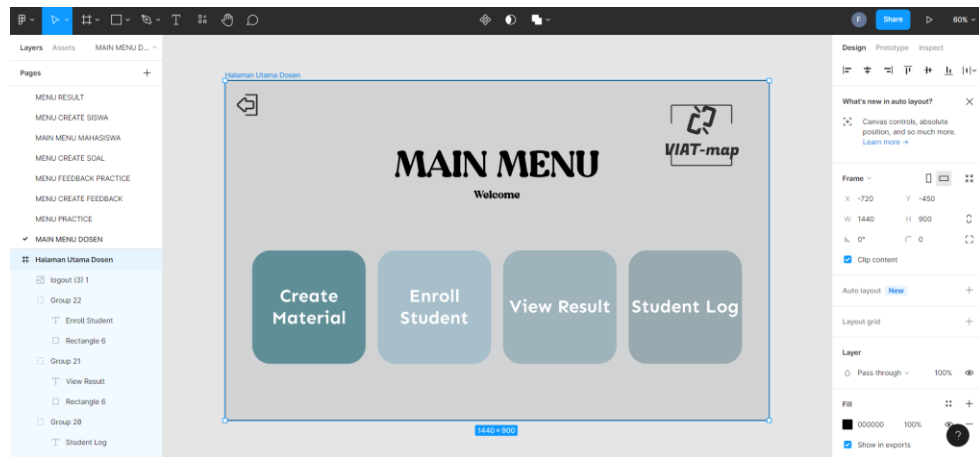
Gambar 4.11 Desain Halaman Form Lupa Password



Gambar 4.12 Desain Tampilan Halaman Output Lupa *Password*

4.4.3 Tampilan Menu Utama Dosen

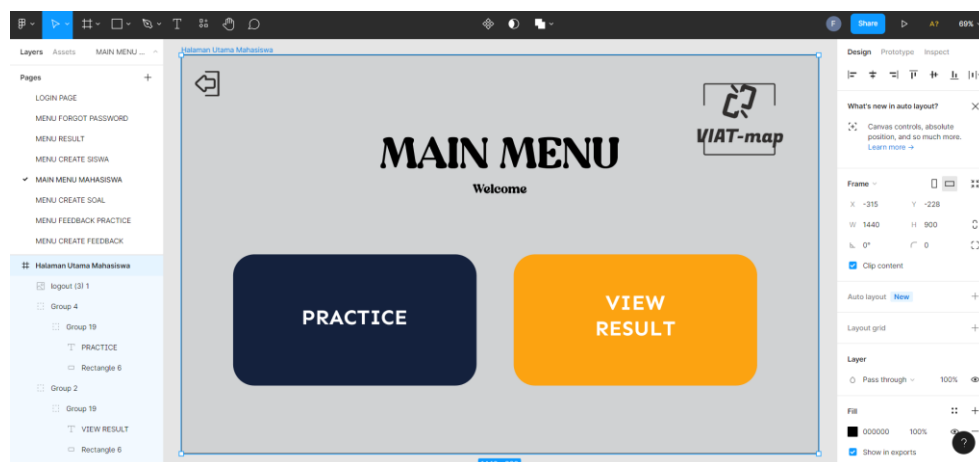
Setelah pengguna dengan level “Dosen” berhasil login ke aplikasi maka akan menuju ke halaman utama dari menu dosen. Halaman ini terdapat beberapa menu yang dapat dipilih. Adapun desain dari halaman utama dosen dapat dilihat pada gambar 4.12 berikut ini:



Gambar 4.13 Desain Tampilan Halaman Menu Utama Dosen

4.4.4 Tampilan Menu Utama Mahasiswa

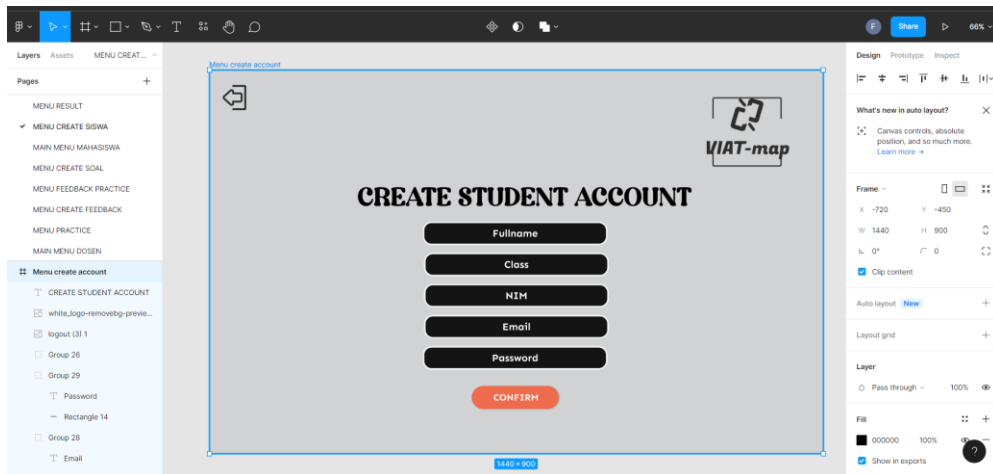
Setelah pengguna dengan level “Mahasiswa” berhasil login ke aplikasi maka akan menuju ke halaman utama dari menu mahasiswa. Halaman ini terdapat 2 menu yang dapat dipilih. Adapun desain dari halaman utama dosen dapat dilihat pada gambar 4.13 berikut ini:



Gambar 4.14 Desain Tampilan Halaman Menu Utama Mahasiswa

4.4.5 Tampilan Menu Daftar Mahasiswa

Pada saat Dosen menekan tombol “*Enroll Student*” pada halaman utama dosen maka akan menuju kepada halaman pendaftaran akun mahasiswa, yang mana dosen akan menginputkan informasi – informasi oleh mahasiswa agar dapat terdaftar ke dalam sistem. Adapun desain halaman pendaftaran akun Mahasiswa dapat dilihat pada gambar 4.14 dibawah ini:



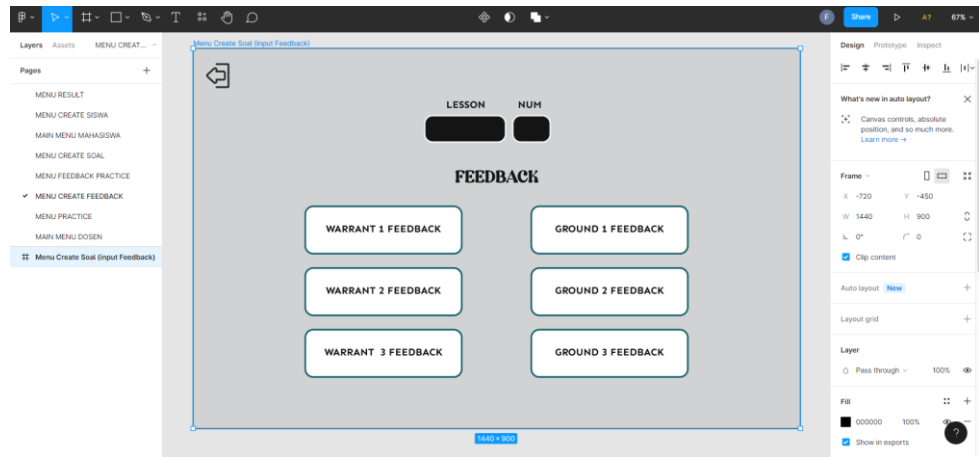
Gambar 4.15 Desain Tampilan Halaman Daftar Mahasiswa

4.4.6 Tampilan Menu Membuat Materi

Pada saat Dosen menekan tombol “*Create Material*” pada halaman utama dosen maka akan menuju kepada halaman untuk membuat materi yang akan diujikan, yang mana dosen akan menginputkan bahan bacaan, serta pilihan-pilihan jawaban serta *feedback* dari pilihan-pilihan jawaban tersebut apabila jawaban tersebut salah. Adapun desain halaman untuk membuat materi dapat dilihat pada gambar 4.15 dan 4.16 dibawah ini:



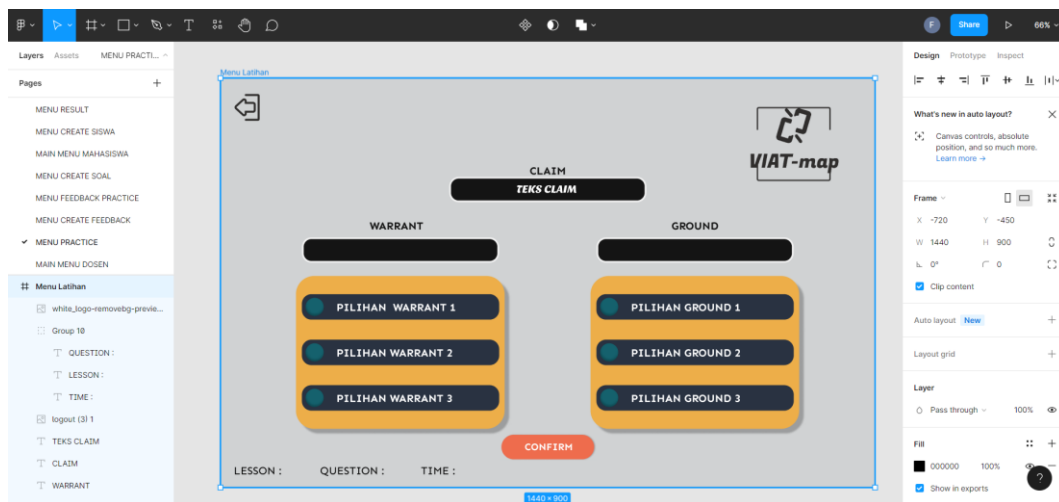
Gambar 4.16 Desain Tampilan Halaman Membuat Materi



Gambar 4.17 Desain Tampilan Halaman Input *Feedback*

4.4.7 Tampilan Menu Latihan

Pada saat Mahasiswa menekan tombol “Practice” pada halaman utama mahasiswa maka akan menuju kepada halaman latihan, yang mana mahasiswa akan disuguhkan bahan bacaan beberapa pilihan jawaban yang harus dibangun dengan cara melakukan drag pilihan dan drop ke bidang yang disediakan. Apabila pilihan jawaban yang di bangun oleh mahasiswa salah maka akan menampilkan pop-up yang berisi umpan balik. Adapun desain halaman untuk membuat materi dapat dilihat pada gambar 4.17 dibawah ini:



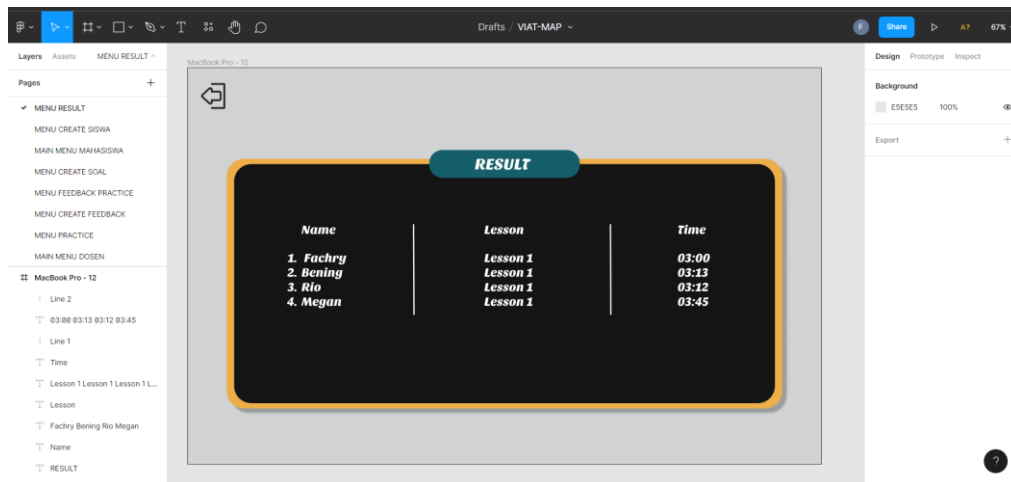
Gambar 4.18 Desain Tampilan Menu Latihan



Gambar 4.19 Desain Tampilan Pop-Up *Feedback*

4.4.8 Tampilan Menu Hasil

Pada saat Dosen & Mahasiswa menekan tombol “View Result” pada halaman utama maka akan menampilkan hasil dari mahasiswa yang telah selesai mengerjakan latihan. Adapun desain Hasil Latihan dapat dilihat pada gambar 4.19 dibawah ini:



Gambar 4.20 Desain Tampilan menu Hasil