

## BAB III

### METODOLOGI PENELITIAN

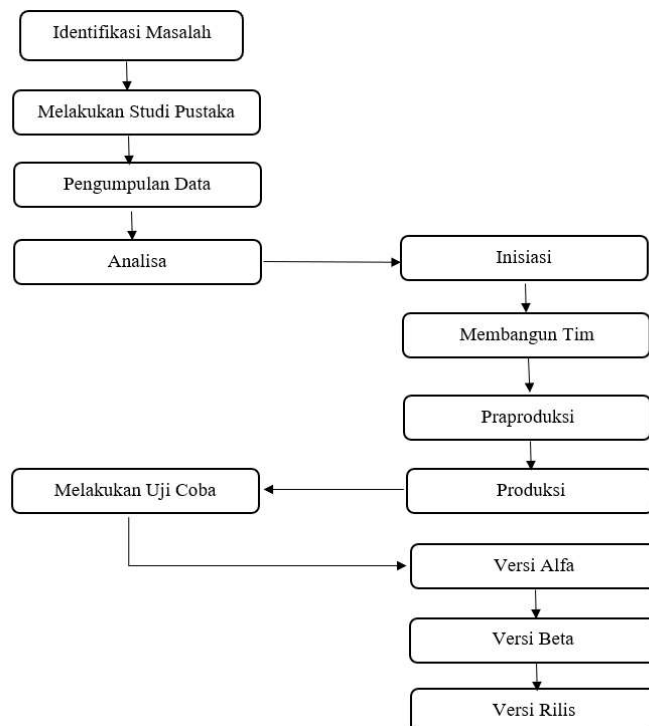
#### 3.1 Metode Penelitian

#### 3.2 Desain Penelitian

Desain penelitian pada penelitian ini digunakan untuk mempermudah dan menstrukturkan tahapan-tahapan perancangan *game* yang akan dibuat.

#### 3.3 Proses Perancangan Sistem

Dalam tahap penelitian yang dilakukan peneliti, mencakup beberapa langkah-langkah penelitian dari awal sampai selesai. Masing-masing langkah dari penelitian diuraikan secara spesifik dibawah ini, antara lain :



**Gambar 3.1** Tahapan Penelitian

Sumber : Data peneliti

Berdasarkan gambar 3.1 tahapan penelitian ini adalah :

1. Identifikasi Masalah

Dalam tahapan ini, peneliti mengidentifikasi sebuah permasalahan yang ada yaitu minimnya sumber belajar bahasa pemrograman C# dan minimnya aplikasi *game* yang mengajarkan tentang coding C# terutama yang berbahasa Indonesia.

2. Melakukan Studi Pustaka

Dalam tahapan ini, peneliti melakukan penelitian tentang studi pustaka, untuk mendapatkan teori-teori pendukung yang berkaitan dengan materi C#, metode, dan teori pendukung lainnya yang digunakan sebagai acuan dalam melaksanakan penelitian.

3. Pengumpulan Data

Dilakukan pengamatan terhadap *junior programmer* PT Inotek Mitra Indonesia tentang bagaimana mereka menggunakan C# dalam menyelesaikan pekerjaan sehari – hari.

4. Analisa

Setelah berhasil melakukan pengumpulan data, peneliti harus melakukan analisis terhadap data yang didapatkan guna merancang *game* edukasi *trivia* C# dengan metode *GDLC* berbasis *android*, alat bantu analisa *game* digunakan *UML*.

5. Inisiasi

Dilangkah ini peneliti memutuskan jenis *game* apa yang akan diciptakan seperti materi – materi yang akan digunakan. Adapun langkah-langkah inisiasi yang peneliti lakukan adalah :

a. Membangun Tim

Ditahapan ini dikarenakan peneliti mandiri maka tidak dibutuhkan untuk mengembangkan struktur tim dan pembagian tugas.

b. Praproduksi

Ditahapan ini peneliti memutuskan *gameplay* permainan, sistem *level*, sistem skor, dan mesin *game* yang akan digunakan. Ditahapan ini juga peneliti membangun prototipe.

c. Produksi

Pada tahap ini peneliti menggunakan prototipe yang dibuat pada saat praproduksi dan mengembangkan *game*. Disini peneliti mengubah rancangan menjadi pengkodean menggunakan aplikasi *unity* dan *visual studio code* dan akhirnya menghasilkan *game* yang dapat dimainkan tetapi belum lengkap sepenuhnya.

6. Melakukan Uji Coba

Ditahapan ini peneliti melakukan uji coba terhadap *game* yang dibuat pada saat produksi

a. Versi *Alfa*

Disini peneliti melakukan pengujian *alfa* dan memperbaiki *bug* yang ditemukan pada *game* versi *alfa*

b. Versi *Beta*

Disini peneliti melakukan pengujian *beta* terhadap *game* yang sudah selesai dan final. Disini peneliti memperbaiki *bug* yang ditemukan pada *game* versi *beta*

c. Versi Rilis

Ini merupakan tahap pengujian terakhir, dimana *game* hasil pengujian pada versi *beta* disempurnakan. Pengujian pada tahap ini menggunakan metode *black box*.

### 3.4 Peralatan Yang Digunakan

Dari rancangan yang akan dibangun, peneliti membedakan beberapa kategori dalam peralatan yang digunakan, dimana kriterianya adalah sebagai berikut :

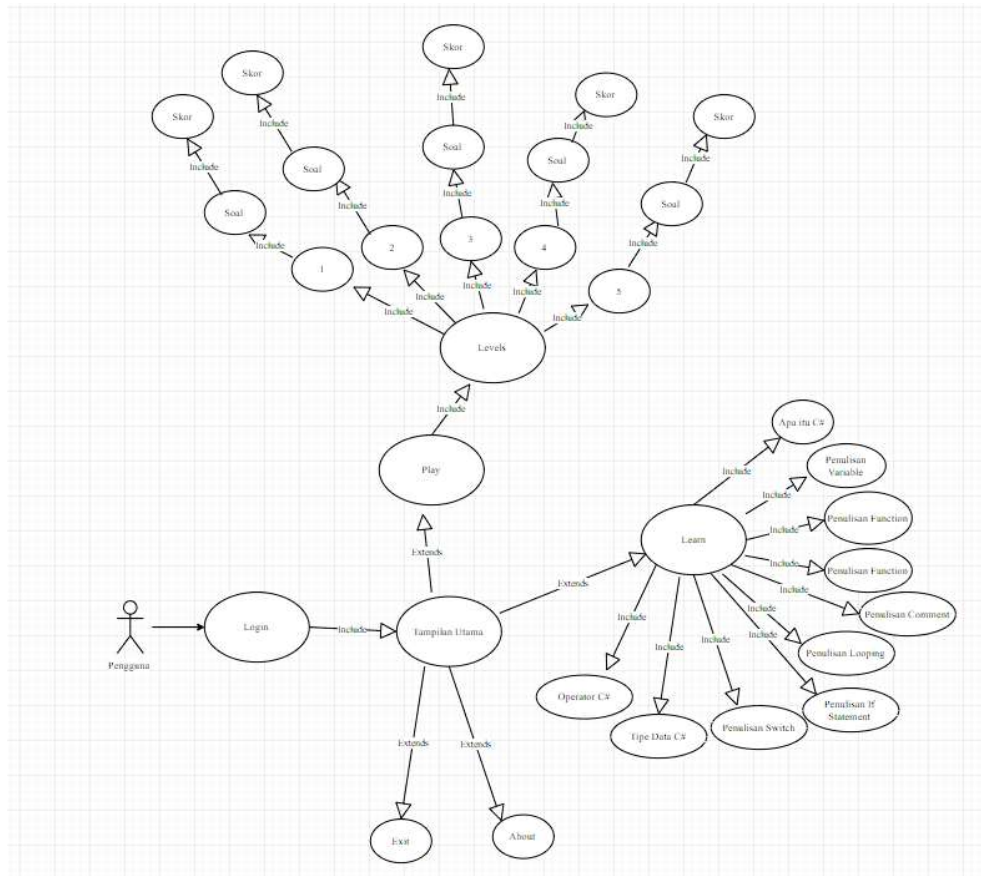
**Tabel 3.1** Data Peneliti

<b>Jenis Alat &amp; Bahan</b>	<b>Alat &amp; Bahan</b>
Hardware	Laptop <i>Asus</i>
	<i>Handphone Redmi Note 8 Pro</i>
Software	<i>Unity</i>
	<i>Visual Studio Code</i>

**Sumber :** Data peneliti

### 3.5 UML (Desain Unified Modeling Language)

#### 1. Use Case Diagram



**Gambar 3.2** Use Case Diagram  
Sumber : Data peneliti

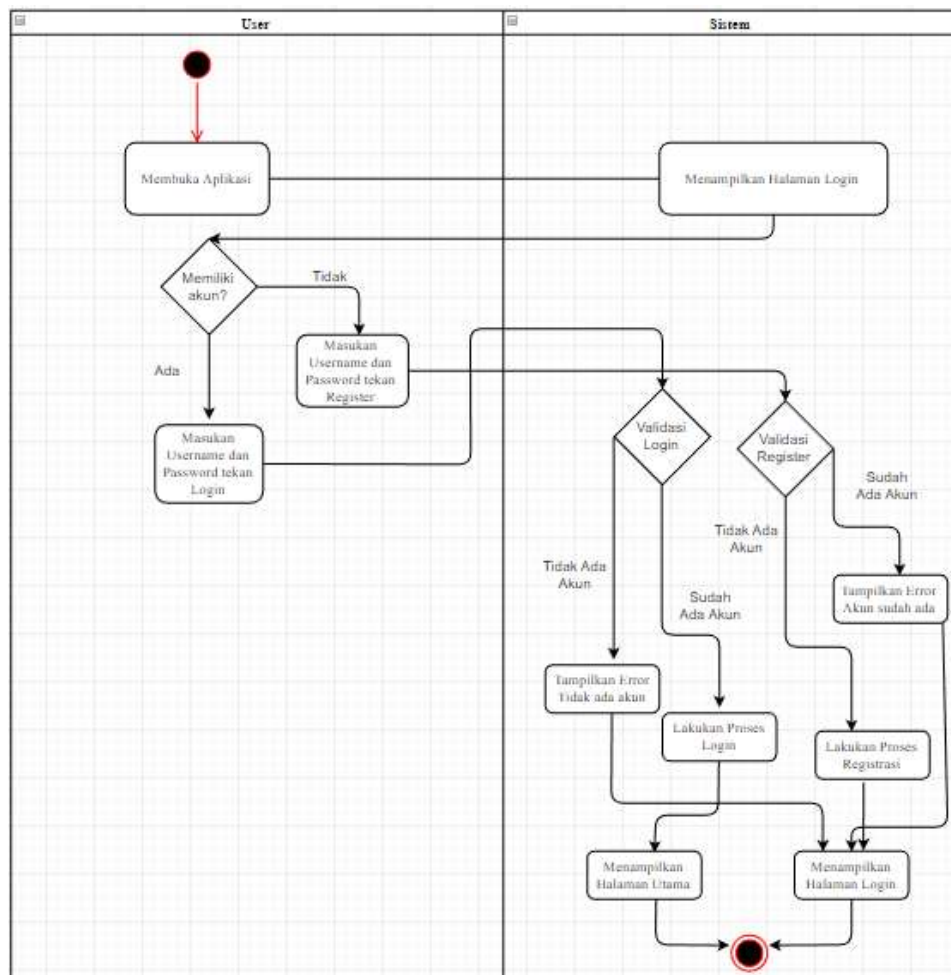
Aktor pada *game* ini adalah orang-orang yang sedang belajar atau akan belajar pemrograman C#. Setelah aplikasi *game* dijalankan pengguna akan di melihat tampilan *login* yang memiliki dua *input text field* yaitu *username* dan *password* setelah itu *user* dapat melakukan *login* atau *register*. Setelah itu *user* jika *login* sukses maka *user* akan diarahkan ke tampilan utama yang memiliki empat menu yaitu *play*, *learn*, *about* dan *exit*. Pada menu *play* pengguna dapat memilih level setelah itu permainan akan dimulai, menu *learn* pengguna dapat

belajar materi C#, sedangkan pada menu *about* pengguna dapat melihat informasi tentang aplikasi *game* trivia c#, dan pada menu *exit* pengguna dapat keluar dari aplikasi *game*.

## 2. Activity Diagram

Activity diagram digunakan untuk menjelaskan aktivitas dari sebuah sistem menu yang ada pada perangkat lunak. Activity diagram pada penelitian ini sebagai berikut:

### a. Activity Diagram Login



**Gambar 3.3** Activity Diagram Login

Sumber : Data peneliti

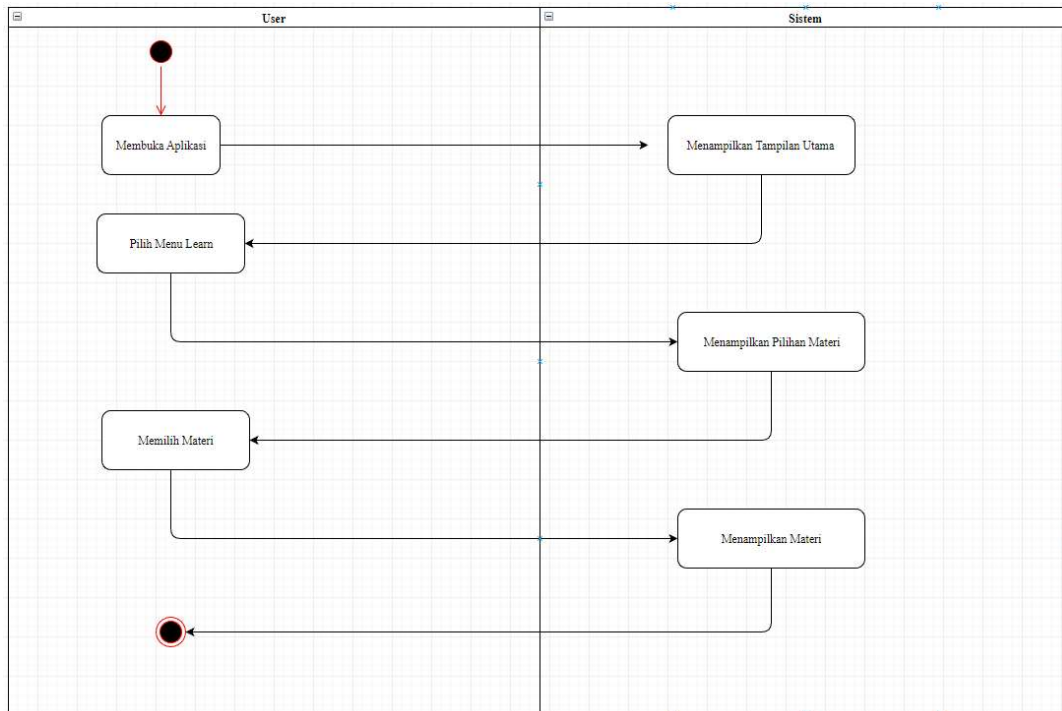
Berdasarkan gambar diatas, *user* menjalankan aplikasi dan kemudian aplikasi akan menampilkan halaman *login*. Jika *user* memiliki akun maka *user* memasukan *username* dan *password* serta menekan tombol *login* dan sistem akan melakukan validasi. Jika hasil validasi adalah akun sudah ada maka sistem akan menampilkan halaman utama. Jika hasil validasi adalah akun tidak ada maka sistem akan menampilkan *error* dan kembali ke halaman *login*. Jika *user* tidak memiliki akun maka *user* memasukan *username* dan *password* serta menekan tombol *register* dan sistem akan melakukan validasi. Jika hasil validasi adalah akun belum ada maka sistem akan melakukan registrasi akun dan menampilkan halaman *login*. Jika *user* sudah ada maka sistem akan menampilkan *error* dan kembali ke halaman *login*.





Berdasarkan gambar diatas, setelah *user* sukses melakukan login, sistem akan menampilkan halaman utama. Setelah itu *user* memilih menu *play* dan sistem menampilkan pilihan levels. *User* memilih salah satu *level* dan sistem menampilkan soal setelah itu *user* menjawab soal dan sistem melakukan validasi jawaban jika jawaban benar maka skor akan bertambah dan sistem akan menampilkan soal selanjutnya jika jawaban salah maka skor akan *reset* dan sistem akan menampilkan soal pertama *level* tersebut. Jika *level* sudah diselesaikan maka sistem akan menampilkan halaman *level cleared*. Setelah itu jika *user* memilih untuk lanjut maka sistem akan menampilkan soal dan melakukan perulangan yang sama seperti diatas. Jika semua *level* sudah berhasil diselesaikan maka sistem akan menampilkan tampilan *gameover*.

### c. Activity Diagram Menu Learn

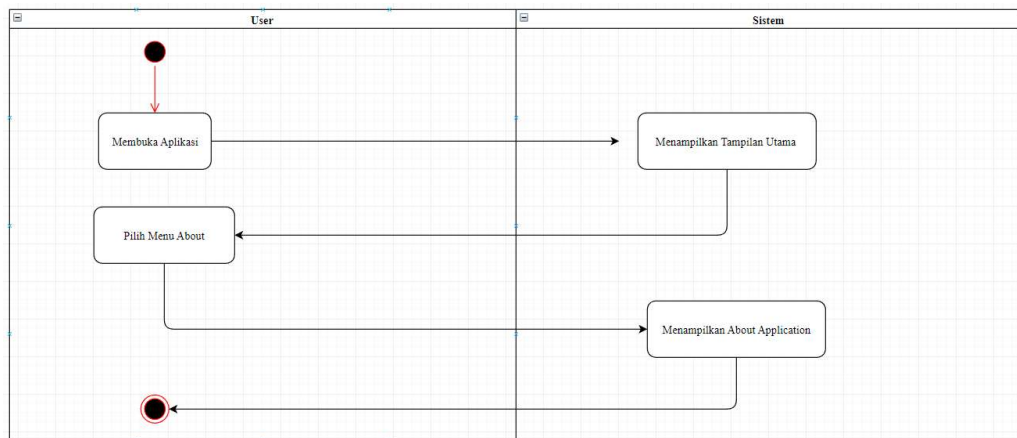


**Gambar 3.5** Activity Diagram Menu Learn

Sumber : Data peneliti

Berdasarkan gambar diatas, *user* menjalankan aplikasi dan kemudian aplikasi akan menampilkan halaman utama. Setelah itu *user* sudah berada di dalam *menu* utama, *user* mengklik *menu learn* dan aplikasi akan menampilkan halaman *learn* yang dimana *user* akan memilih materi tentang pemograman *C#*. Setelah itu *user* memilih salah satu materi di halaman *learn* dan aplikasi akan menampilkan materi yang dipilih *user*. Setelah *user* menyelesaikan materi tersebut *user* dapat menekan tombol *back* dan aplikasi akan kembali ke halaman *learn* setelah itu *user* dapat memilih materi selanjutnya atau kembali ke halaman utama.

#### d. Activity Diagram Menu About

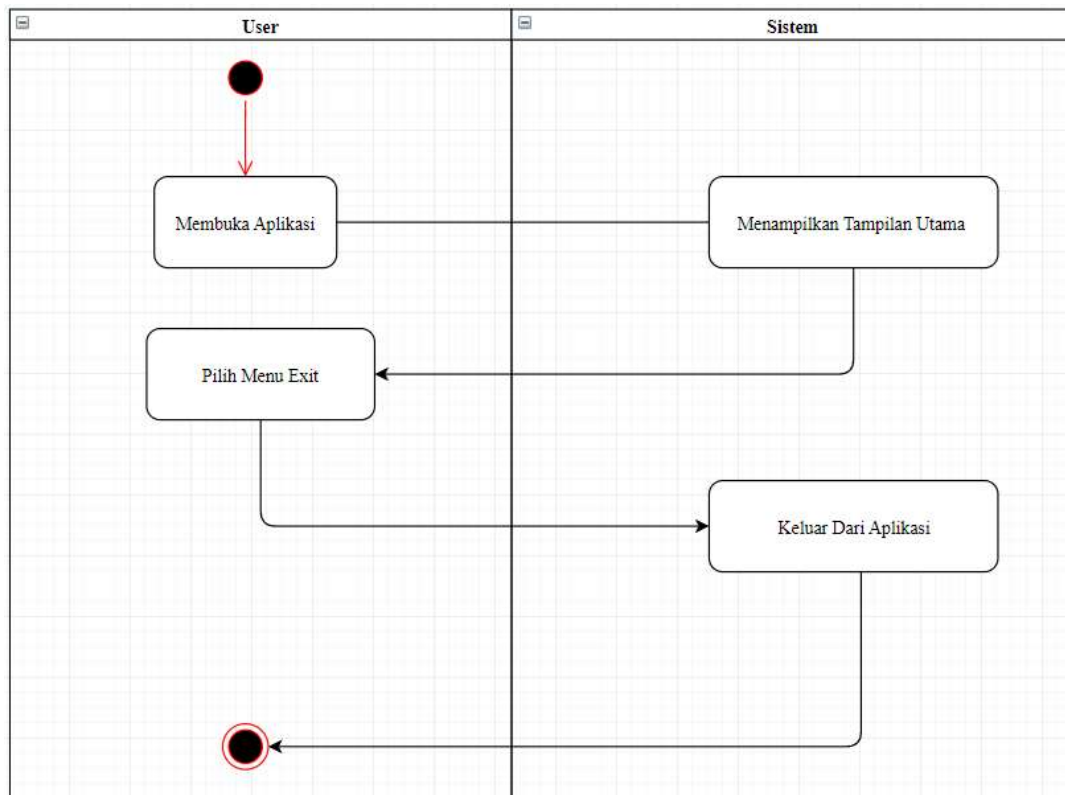


**Gambar 3.6** Activity Diagram Menu About

Sumber : Data peneliti

Berdasarkan gambar diatas, *user* menjalankan aplikasi dan kemudian aplikasi akan menampilkan halaman utama. Setelah itu *user* sudah berada di dalam *menu* utama, *user* mengklik *menu about* dan aplikasi akan menampilkan halaman *about* yang dimana *user* dapat melihat sedikit informasi tentang aplikasi seperti versi dan nama alias pengembang. Setelah itu *user* untuk kembali ke halaman *user* dapat menekan tombol *back* dan aplikasi akan kembali ke halaman utama.

### e. Activity Diagram Menu Exit



**Gambar 3.7** Activity Diagram Menu Exit

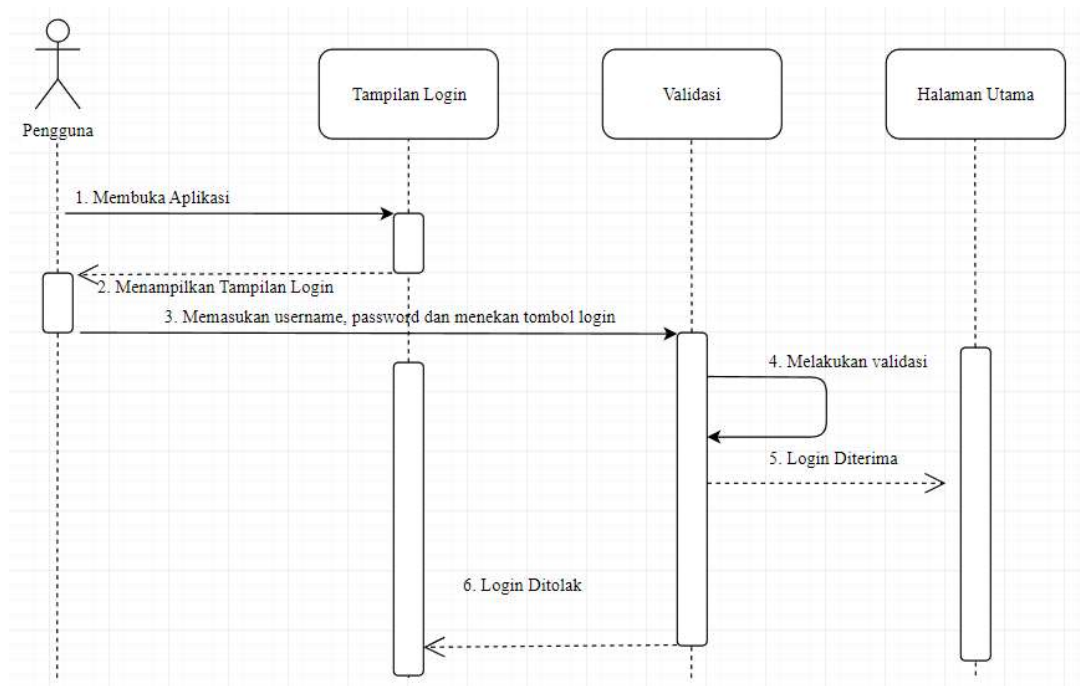
**Sumber :** Data peneliti

Berdasarkan gambar diatas, *user* menjalankan aplikasi dan kemudian aplikasi akan menampilkan halaman utama. Setelah itu *user* sudah berada di dalam *menu* utama, *user* mengklik *menu exit* dan aplikasi akan keluar.

### 3. Sequence Diagram

Sequence Diagram berguna untuk menjelaskan waktu hidup objek dan pesan yang dikirim maupun yang akan diterima.

#### a. Sequence Diagram Login

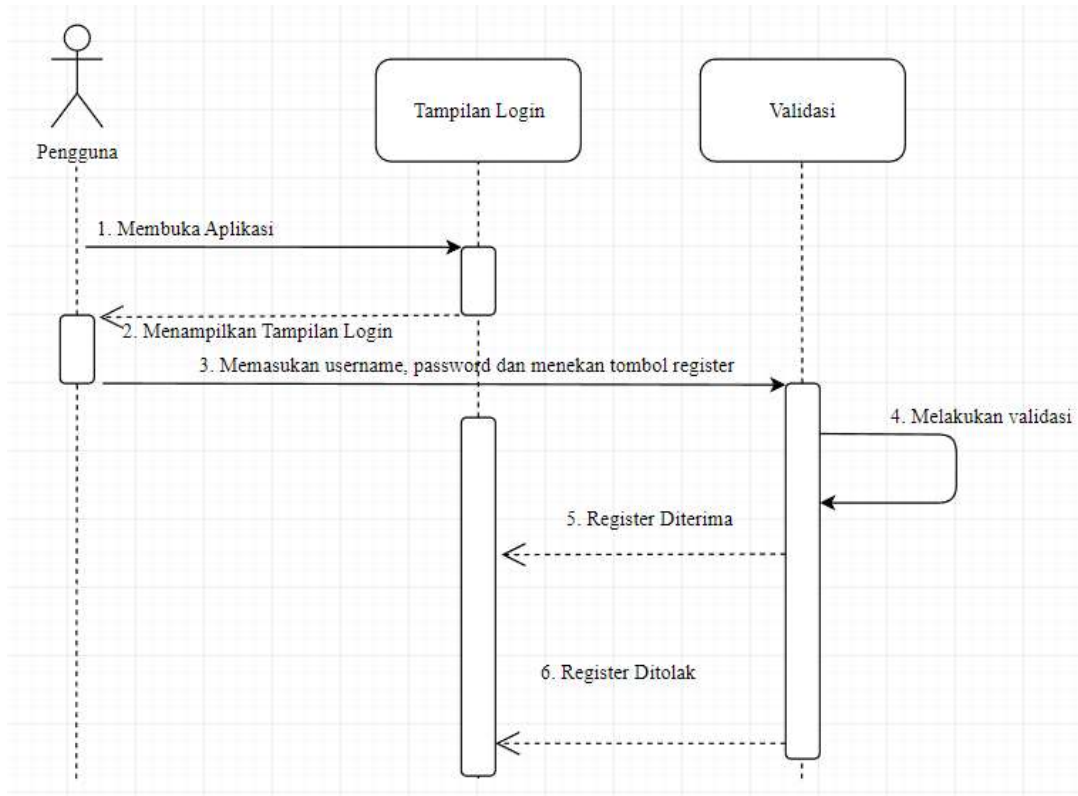


**Gambar 3.8** Sequence Diagram Login

Sumber : Data peneliti

Berdasarkan gambar diatas, setelah *user* menjalankan aplikasi sistem akan menampilkan halaman *login*, setelah itu *user* memasukan *username* dan *password* serta menekan tombol *login*, setelah itu sistem melakukan validasi apakah *username* dan *password* tersebut benar. Jika benar maka sistem akan menampilkan halaman utama. Jika salah maka sistem akan menampilkan kembali halaman *login*

## b. Sequence Diagram Register

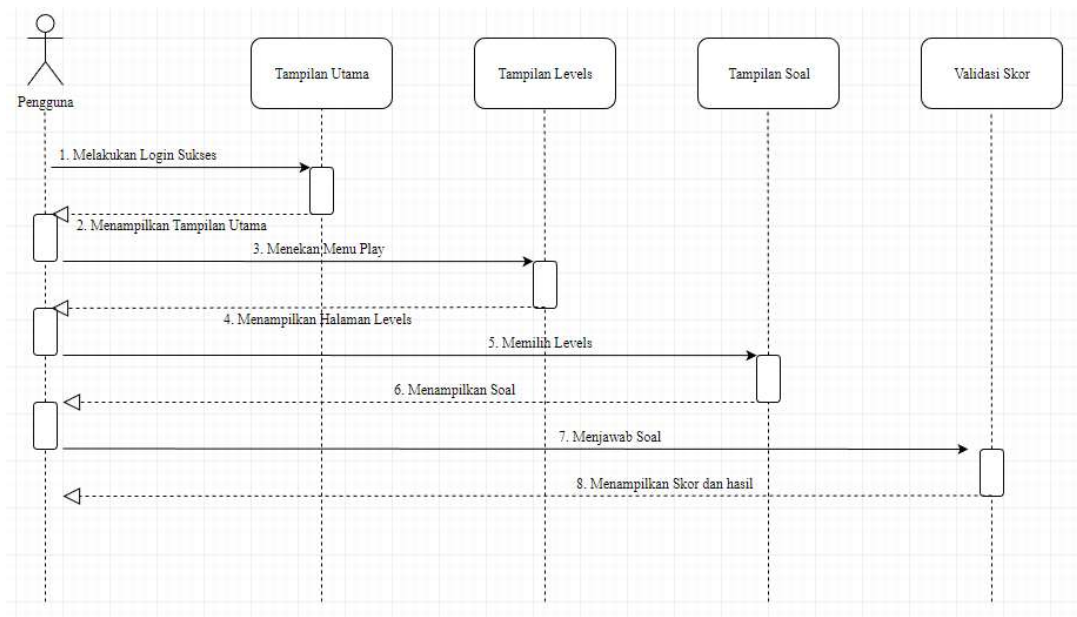


**Gambar 3.9** *Sequence Diagram Register*

**Sumber :** Data peneliti

Berdasarkan gambar diatas, setelah *user* menjalankan aplikasi sistem akan menampilkan halaman *login*, setelah itu *user* memasukkan *username* dan *password* serta menekan tombol *register*, setelah itu sistem melakukan validasi apakah *username* dan *password* tersebut sudah terdapat di sistem. Jika di sistem sudah ada maka sistem akan menampilkan halaman *login* dan pesan *error*. Jika di sistem belum ada maka sistem akan menampilkan kembali halaman *login* dan pesan *success*.

### c. Sequence Diagram Play

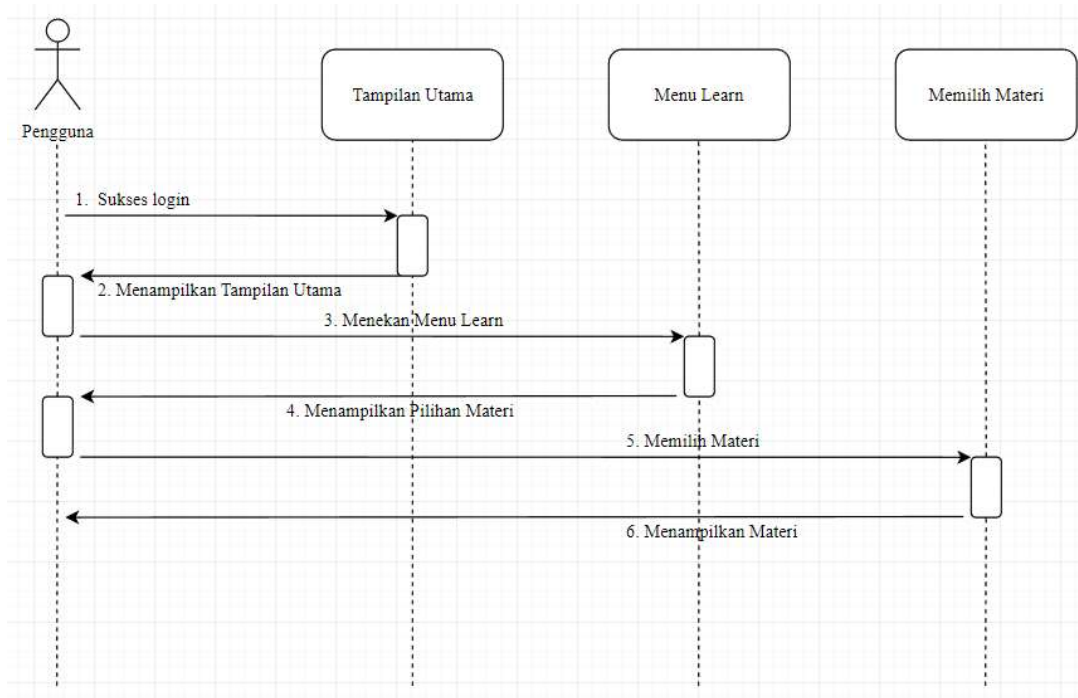


**Gambar 3.10** *Sequence Diagram Play*

**Sumber :** Data peneliti

Dapat dilihat dari gambar diatas, setelah *user* melakukan *login* dengan sukses maka sistem menampilkan halaman utama. Setelah itu *user* memilih menu *play* dan sistem menampilkan halaman *levels* yang berisi lima *level* dan tiap *level* memiliki lima butir soal objektif. Selanjutnya *user* menjawab soal dan aplikasi melakukan pengecekan apakah jawaban yang dipilih *user* benar, jika benar maka skor akan *diupdate* dan soal selanjutnya akan ditampilkan, jika salah maka skor dan soal akan *direset*.

#### d. Sequence Diagram Learn



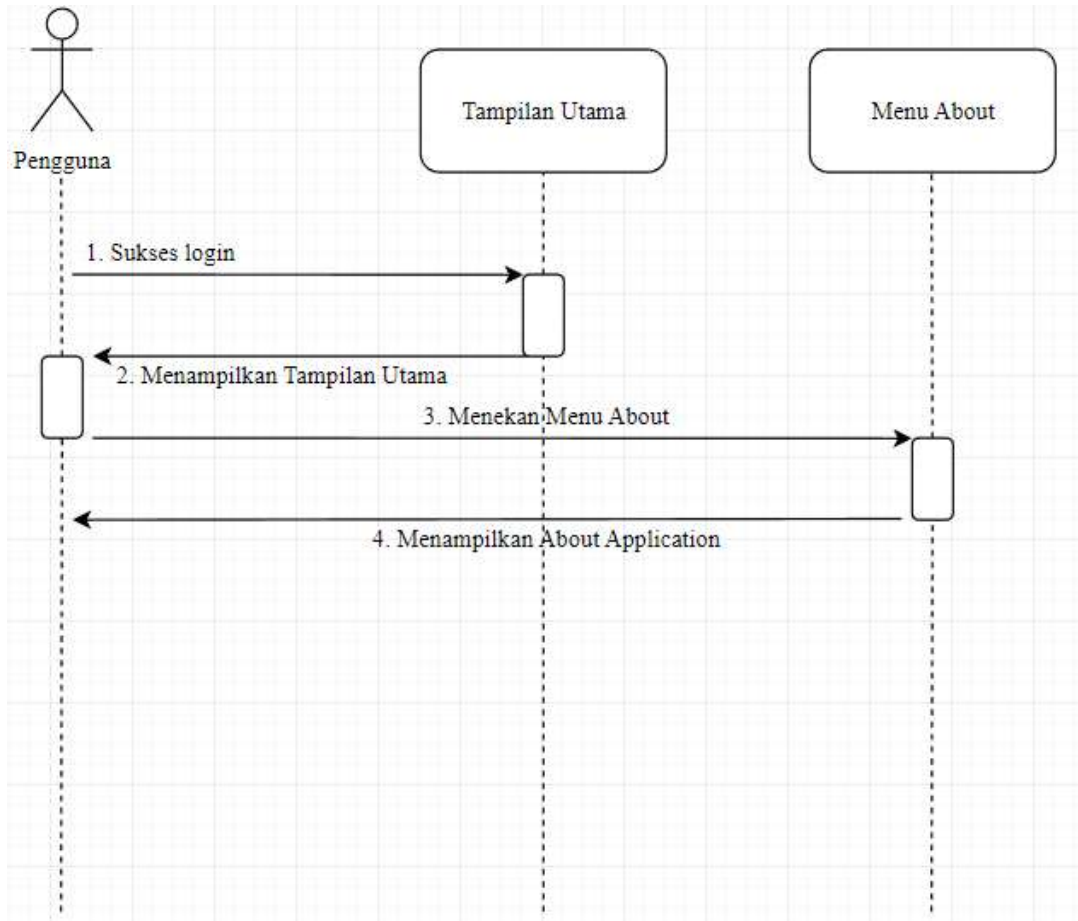
**Gambar 3.11** *Sequence Diagram Learn*

**Sumber :** Data peneliti

Berdasarkan gambar diatas, *user* menjalankan aplikasi dan kemudian aplikasi akan menampilkan halaman utama. Selanjutnya *user* menekan *menu learn*, aplikasi menampilkan halaman *learn* yang berisi materi yang berkaitan dengan pemograman *C#*. Selanjutnya *user* memilih materi yang diminati dan aplikasi akan menampilkan materi tersebut.



### e. Sequence Diagram About

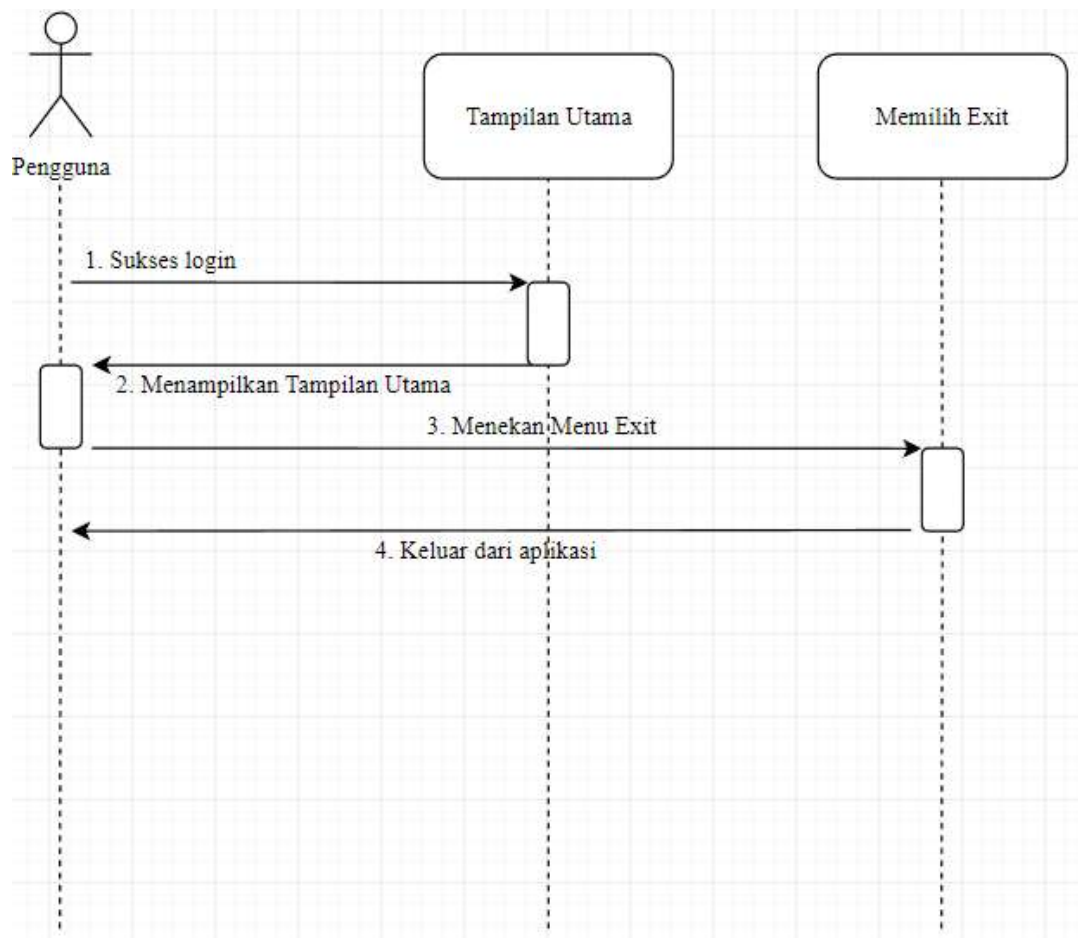


**Gambar 3.12** *Sequence Diagram About*

**Sumber :** Data peneliti

Berdasarkan gambar diatas, *user* menjalankan aplikasi dan kemudian aplikasi akan menampilkan halaman utama. Selanjutnya *user* menekan *menu about*, aplikasi menampilkan halaman *about* yang berisi informasi yang berkaitan dengan aplikasi, didalam menu about terdapat versi dan nama pengembang. Selanjutnya *user* dapat menekan tombol *back* untuk kembali ke halaman utama

### f. Sequence Diagram Exit



**Gambar 3.13** *Sequence Diagram Exit*

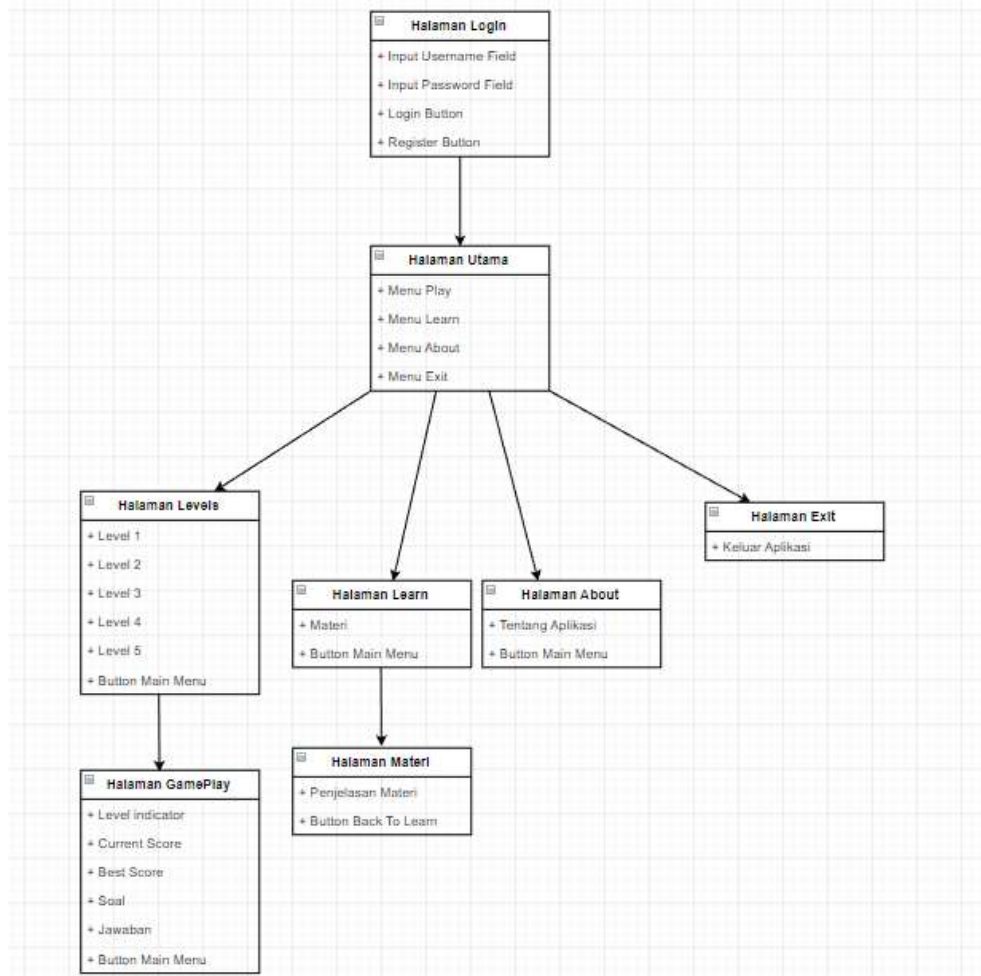
**Sumber :** Data peneliti

Berdasarkan gambar diatas, *user* menjalankan aplikasi dan kemudian aplikasi akan menampilkan halaman utama. Selanjutnya *user* menekan *menu exit*, setelah itu aplikasi akan keluar.

#### 4. Class Diagram

*Class Diagram* digunakan untuk menggambarkan kelas yang berorientasi objek dalam sistem yang saling berhubungan satu sama lain.

Diagram *class* pada penelitian ini sebagai berikut:



**Gambar 3.14** *Class Diagram*

Sumber : Data peneliti

### 3.6 Perancangan Tampilan

Perancangan tampilan ini digunakan untuk menjelaskan gambaran atau desain rancangan aplikasi *game* edukasi *coding c#* berbasis *android* yang akan dibuat.

#### 1. Rancangan Tampilan Login



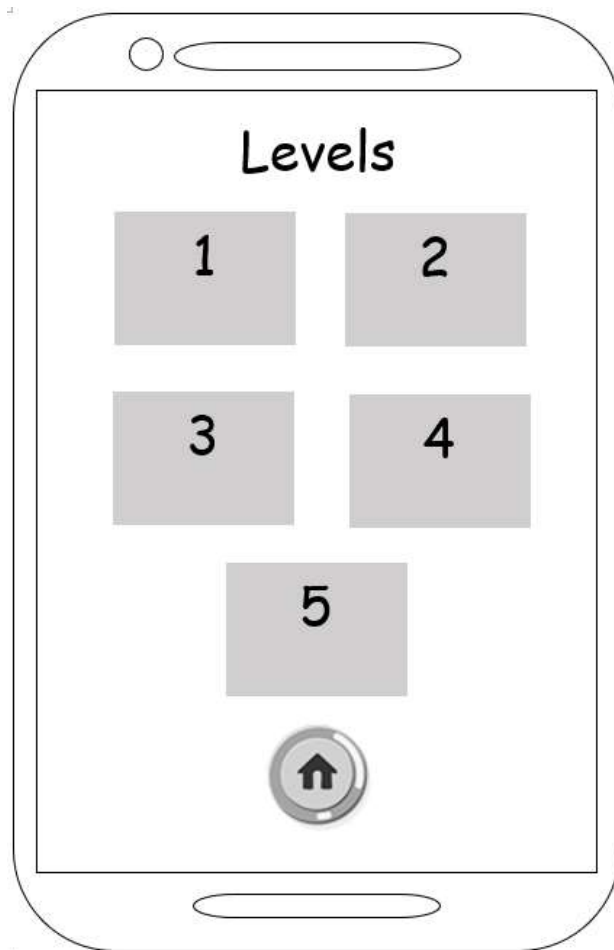
**Gambar 3.15** Rancangan Tampilan Login  
Sumber : Data peneliti

## 2. Rancangan Tampilan Utama

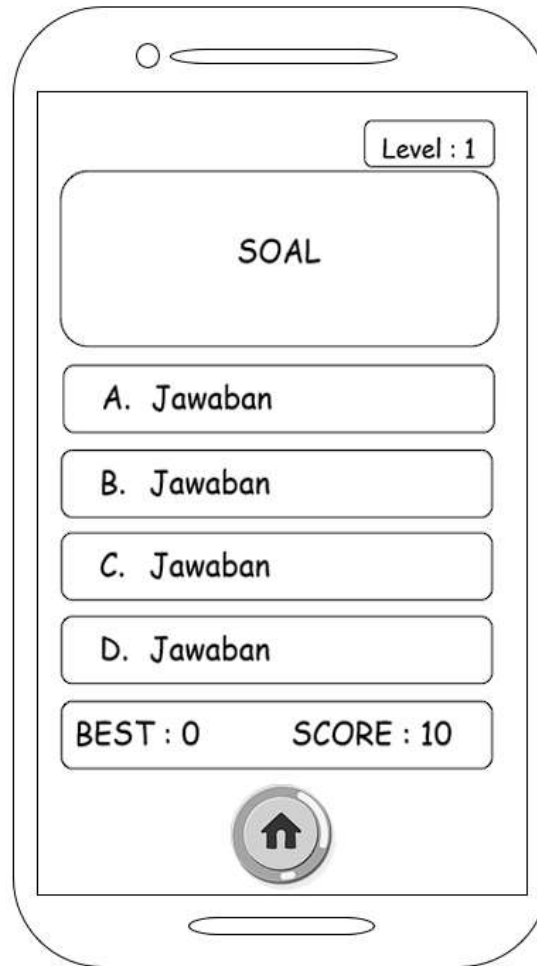


**Gambar 3.16** Rancangan Tampilan Utama  
**Sumber :** Data peneliti

### 3. Rancangan Tampilan *Levels*



**Gambar 3.17** Rancangan Tampilan *Levels*  
**Sumber :** Data peneliti

4. Rancangan Tampilan *GamePlay*

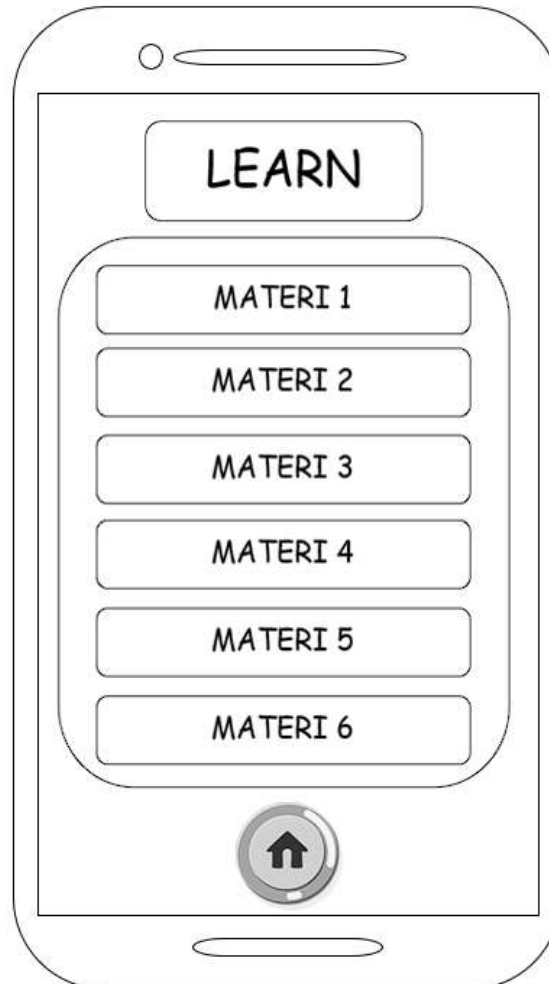
**Gambar 3.18** Rancangan Tampilan *GamePlay*  
**Sumber :** Data peneliti

5. Rancangan Tampilan *Level Cleared*



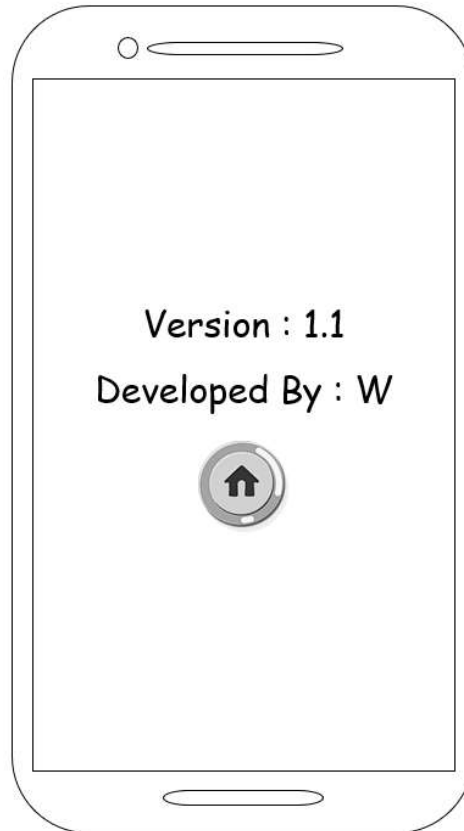
**Gambar 3.19** Rancangan Tampilan *Learn*  
**Sumber :** Data peneliti



6. Rancangan Tampilan *Learn*

**Gambar 3.20** Rancangan Tampilan *Learn*  
**Sumber :** Data peneliti

7. Rancangan Tampilan Jawaban *About*



**Gambar 3.21** Rancangan Tampilan *About*  
**Sumber :** Data peneliti

## 8. Rancangan Tampilan *Gameover*



**Gambar 3.22** Rancangan Tampilan *Gameover*  
**Sumber :** Data peneliti

### 3.7 Jadwal Penelitian

Penelitian ini di laksanakan oleh peneliti selama 5 bulan. Berikut adalah tabel jadwal penelitian yang dilakukan oleh peneliti:

**Tabel 3.2** Jadwal Penelitian

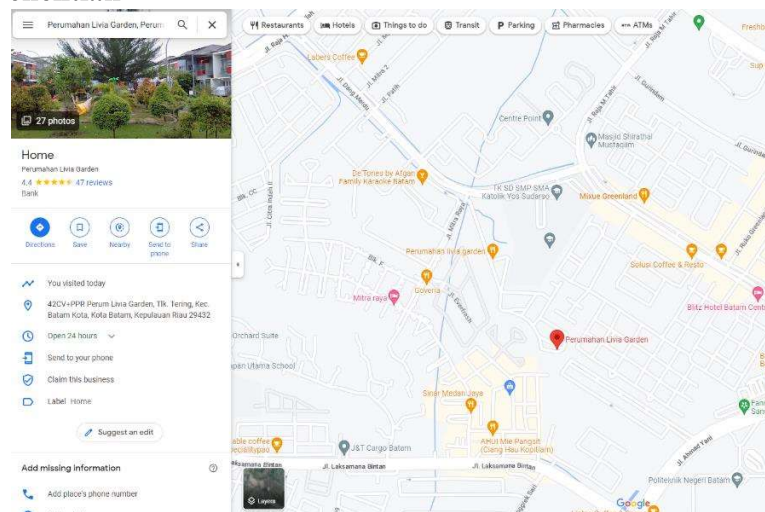
Kegiatan	Tahun 2022																Tahun 2023											
	September				Oktober				November				Desember				Januari			Februari			Maret					
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	1	2	3	4	1	2	3	4	
Pengajuan Judul			■																									
Penyusunan Bab I					■	■	■	■																				
Penyusunan Bab II									■	■	■	■																
Penyusunan Bab III													■	■	■	■												
Penyusunan Bab IV																■	■	■	■									
Penyusunan Bab V																			■	■	■	■						
Pengumpulan Skripsi																										■		

Sumber : Data peneliti

### 3.8 Metode Pengujian Sistem

Aplikasi *game* ini menggunakan metode *blackbox* untuk melakukan pengujian terhadap tampilan dan fungsinya. Agar dapat mengetahui apakah program sudah berfungsi sesuai dengan yang diinginkan.

### 3.9 Lokasi Penelitian



**Gambar 3.23** Lokasi Penelitian  
Sumber : Data Peneliti