

BAB I

PENDAHULUAN

A. Latar Belakang

Perkembangan teknologi begitu pesat saat ini, semua hal yang memiliki hubungan dengan manusia dan memberikan kemudahan untuk menghasilkan dan/atau menggunakan sesuatu. Hal ini mengakibatkan ikut berkembangnya suatu pemanfaatan teknologi di kehidupan sehari-hari terutama pada penggunaan *smartphone android* baik pada orang tua, anak muda, maupun anak-anak. Sudah tidak heran jika kelompok usia anak-anak hingga orang tua telah menggunakan *smartphone*. Sedangkan bagi anak-anak, *smartphone* sebagian besar digunakan untuk mengakses sosial media dan game.

Game merupakan salah satu media audio visual yang dibuat untuk tujuan tertentu dan dirancang agar pengguna secara lebih aktif terlibat dengan sistem didalamnya. Perkembangan game saat ini sangat berkembang pesat sehingga menghasilkan berbagai macam aplikasi pembelajaran yang bisa dinikmati mulai dari anak-anak hingga dewasa. Salah satu aplikasi pembelajaran yang cocok untuk saat ini adalah aplikasi *education game* dalam pembelajaran edukasi. Dengan adanya aplikasi pembelajaran edukasi, pelajaran yang awalnya dianggap sulit dan membosankan bisa dituangkan dalam aplikasi pembelajaran edukasi sehingga lebih menyenangkan sekaligus menambah pengetahuan.

Rambu lalu lintas adalah bagian dari perlengkapan jalan yang berupa, lambang, huruf, angka, kalimat atau perpaduan yang diantaranya berfungsi

sebagai peringatan, larangan, perintah, dan petunjuk bagi pengguna jalan. Rambu lalu lintas merupakan salah satu pengetahuan yang sangat penting bagi pengguna jalan dalam menertibkan lalu lintas, hal ini dikarenakan dapat mengurangi permasalahan lalu lintas seperti macet, kecelakaan lalu lintas dan permasalahan lalu lintas lainnya yang dapat merugikan orang lain.

Peraturan berkendara bermotor telah diatur dalam Undang-Undang Republik Indonesia nomor 22 tahun 2009 tentang Lalu Lintas dan Angkutan Jalan, bahwa pengetahuan dalam berkendara harus diutamakan untuk keselamatan dalam berkendara. Korlantas Polri mencatat angka kecelakaan lalu lintas pada 2022 masih cukup tinggi. Korban tewas akibat kecelakaan lalu lintas mencapai ratusan jiwa. Tingginya kecelakaan kendaraan bermotor di Indonesia bisa dilihat dari data Korlantas Polri bahwa jumlah kejadian kecelakaan di tahun 2021 lebih tinggi dibandingkan tahun 2022, pada tahun 2021 tercatat 25.266 jiwa dan selama 2022 (bulan Mei) sudah ada 11.183 jiwa tak terselamatkan. Hingga data terakhir di 2022 terdapat 15.885 kecelakaan lalu lintas karena pengemudi tidak waspada, gagal menjaga jarak aman dan selebihnya dikarenakan ceroboh saat berbelok (Korlantas Polri, 2022). Faktor dominan penyebab kecelakaan lalu lintas, karena hampir semua kecelakaan lalu lintas yang terjadi disebabkan oleh pelanggaran lalu lintas sehingga pada hal tersebut terdapat permasalahan dalam hal kendaraan bermotor di jalan raya, salah satu langkah alternatif untuk meminimalisir kejadian kecelakaan di jalan raya adalah dengan pendidikan usia dini terkait dengan peraturan berlalu lintas. Pentingnya pendidikan anak usia dini adalah sebagai pembentuk karakter anak, dengan pembentukan karakter yang baik diharapkan

sebagai jembatan penghubung anak di masa depan. Selain itu, hasil dari penelitian (Wulandari, 2015) menyebutkan bahwa kurangnya sosialisasi dan kesadaran untuk belajar mengenai aturan berlalu lintas menjadi salah satu penyebab minimnya pengetahuan serta pemahaman pelajar.

Menurut Undang-Undang No 20 Tahun 2003 pasal 8 tentang hak dan kewajiban masyarakat menyatakan bahwa, masyarakat berhak berperan serta dalam perencanaan, pelaksanaan, pengawasan, dan evaluasi program pendidikan. dan Bab III, pada pasal 4 bahwa pendidikan diselenggarakan dengan memberi keteladanan, membangun kemauan, dan mengembangkan kreativitas peserta didik dalam proses pembelajaran (www.kelembagaan.ristekdikti.go.id, 2016). Mengacu dengan uraian tersebut, memacu penulis untuk berkarya terkait dengan bagaimana mengurangi kejadian kecelakaan berlalu lintas di masa yang akan datang, salah satu upaya yang dilakukan adalah melalui pengenalan atau sosialisai rambu dan aturan lalu lintas dalam bentuk game edukasi.

Berdasarkan latar belakang permasalahan yang telah diuraikan diatas maka diidentifikasi masalahnya untuk meminimalisir pelanggaran terhadap rambu lalu lintas dan meningkatkan pemahaman tentang rambu-rambu lalu lintas, bagi anak usia dini yang merupakan generasi penerus bangsa dijadikanlah suatu sistem baru sebagai solusi permasalahan ini yang kemudian dituangkan dalam bentuk tugas akhir. Maka penulis mengusulkan dengan judul **“Aplikasi *Education Game* Dalam Pembelajaran Interaktif Pengenalan Rambu Lalu Lintas Pada Anak SD Berbasis *Mobile Learning*”**. Penelitian ini bertujuan untuk dapat membantu anak dalam mengenali, mempelajari dan menghafal arti dari rambu lalu lintas.

B. Rumusan Masalah

Berdasarkan uraian dari latar belakang diatas maka dirumuskan masalah sebagai berikut :

1. Bagaimana menganalisis, merancang dan membangun suatu Aplikasi *Education Game* Dalam Pembelajaran Interaktif Pengenalan Rambu Lalu Lintas Pada Anak SD Berbasis *Mobile Learning*?
2. Bagaimana mengimplementasikan suatu aplikasi *education game* rambu lalu lintas dengan menerapkan konsep gamifikasi untuk meningkatkan kemampuan pemahaman dan motivasi belajar?

C. Batasan Masalah

Adapun agar pembahasan menjadi lebih terarah dan tidak menyimpang dan juga sesuai dengan latar belakang yang sudah diuraikan sebelumnya, maka penulis membatasi masalah sebagai berikut :

1. Ruang lingkup dan uji coba implementasi penelitian ini difokuskan atau terpusat pada Anak SD umur 6 – 12 Tahun, dikarenakan anak usia dini akan lebih mudah paham dan memahami apabila pembelajarannya dengan game edukasi.
2. Aplikasi ini hanya memberikan pembelajaran mengenai rambu-rambu lalu lintas, dengan menggunakan gambar simbol rambu lalu lintas yang sudah ditetapkan oleh Pemerintah Indonesia.
3. Antarmuka pemrograman aplikasi dibangun dengan menggunakan bahasa pemrograman *Javascript* dan *Android Studio* serta database *MySQL*.

D. Tujuan Penelitian

Rumusan masalah yang telah didefinisikan sebelumnya tujuan dari penelitian dan perancangan sistem dalam penelitian ini adalah :

1. Untuk merancang dan membangun suatu Aplikasi *Education Game* Dalam Pembelajaran Interaktif Pengenalan Rambu Lalu Lintas Pada Anak SD Berbasis *Mobile Learning*.
2. Untuk mengimplementasikan suatu aplikasi *education game* rambu lalu lintas dengan menerapkan konsep gamifikasi untuk meningkatkan kemampuan pemahaman dan motivasi belajar.

E. Manfaat Penelitian

Salah satu fungsi dari penelitian adalah bisa memberikan manfaat dan kegunaan yang secara nyata akan diterima oleh berbagai pihak yang terkait dengan obyek penelitian secara luas. Dalam penelitian kali ini peneliti berharap bisa memberikan kegunaan. Penulisan tugas akhir ini memberikan manfaat ke beberapa pihak, antara lain :

1. Aspek Akademis
 - a. Untuk mengaplikasikan teori yang di dapat di bangku perkuliahan dengan kondisi real di lapangan, khususnya terkait dengan permasalahan yang dibahas terkait penelitian dan *tools* atau metode penyelesaian masalah.
 - b. Diharapkan sebagai bahan masukan bagi para praktisi pendidikan dalam mengembangkan suatu proses pembelajaran yang menyenangkan melalui *education game*.

2. Aspek Praktis

- a. Memiliki sistem yang baru sebagai salah satu investasi jangka panjang terkait dengan sadar berlalu lintas pada masyarakat dengan melalui pendidikan usia dini karena pada usia dini adalah masa pembentukan karakter.
- b. Menghemat dari segi biaya, waktu dan tenaga dalam proses penyuluhan, sosialisasi dan kesadaran untuk belajar mengenai aturan berlalu lintas.
- c. Sebagai tugas akhir untuk mendapatkan gelar sarjana strata 1 (S.1) pada Fakultas Teknik, Program Studi Informatika, Universitas Muhammadiyah Parepare.

F. Sistematika Penulisan

Sistematika penulisan skripsi ini dibagi ke dalam lima bab yang masing-masing menjelaskan bagian-bagian yang berbeda namun tetap memiliki korelasi satu dengan yang lain. Setiap bab-nya terdiri dari beberapa sub tersendiri. Diawali dengan bab pendahuluan dan diakhiri dengan bab penutup yang berisi kesimpulan dan saran-saran. Dalam menyusun sistematika penulisan, penulis menguraikan ke dalam lima bab sebagai berikut :

BAB I : PENDAHULUAN

Pada bab awal ini akan dijelaskan tentang latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

BAB II : TINJAUAN PUSTAKA

Pada bab ini berisi uraian tentang teori-teori yang digunakan sebagai referensi dalam suatu system “**Aplikasi *Education Game* Dalam Pembelajaran Interaktif Pengenalan Rambu Lalu Lintas Pada Anak SD Berbasis *Mobile Learning***”.

BAB III : METODE PENELITIAN

Pada bab ini berisi mengenai metode perancangan yang digunakan dalam membangun sistem ini. Pada bab ini juga membahas tentang identifikasi masalah, analisis kebutuhan aplikasi yang meliputi kebutuhan masukan, proses dan keluaran suatu aplikasi yang akan dibangun menggunakan UML.

BAB IV : HASIL DAN PEMBAHASAN

Pada bab ini membahas mengenai hasil dari tahap analisis dan perancangan sesuai dengan metode yang digunakan pada alat dan sistem yang dibuat serta hasil dari pengujian serta bentuk-bentuk implementasi aplikasi beserta penerapannya.

BAB V : PENUTUP

Pada bab akhir ini berisi tentang kesimpulan dari penelitian yang dilakukan dan saran-saran yang diberikan untuk pengembangan lebih lanjut agar tercapai hasil yang lebih maksimal.

DAFTAR PUSTAKA

LAMPIRAN

BAB II

TINJAUAN PUSTAKA

A. Penelitian Terdahulu

Pada penelitian terkait ini adalah beberapa hasil dari yang dilakukan dari penelitian sebelumnya sebagai dasar acuan untuk menyelesaikan skripsi. Dari penelitian tersebut, peneliti menemukan beberapa penelitian yang mendukung untuk mengangkat topik yang berkaitan dengan penelitian penulis. Oleh karena itu, peneliti melakukan kajian terhadap beberapa hasil penelitian berupa tugas akhir/ skripsi atau jurnal melalui internet. Berikut adalah tabulasi dari beberapa penelitian yang terkait.

Tabel 2.1 Penelitian Terdahulu

No	Nama Penulis	Judul Penelitian	Gambaran Umum

1.	Setiadi, Lydia Ignacia (2020)	“Pembangunan Aplikasi Pembelajaran Rambu Lalu Lintas Berbasis Multimedia Interaktif”	Penelitian ini bertujuan untuk merancang sistem sebagai sarana untuk menyampaikan materi pelajaran dengan cara yang lebih menarik, interaktif, dan mudah penggunaannya. Aplikasi ini berbasis desktop dan dibuat menggunakan <i>Adobe Flash CS3</i>, <i>Adobe Premiere 1.5</i>, <i>Corel Draw X3</i> dan <i>Any Video Converter</i>. Aplikasi RaTas ini telah berhasil dibangun dan telah diuji cobakan terhadap 40 siswa dari kelas 3 SD. Dari hasil uji coba melalui kuisioner, diperoleh persentase penilaian sangat baik sebanyak 40%, persentase penilaian baik sebesar 51.75% dan persentase penilaian kurang sebesar 4.25%.
No	Nama Penulis (Tahun)	Judul Penelitian	Gambaran Umum
2.	Fitri Marisa	“Aplikasi Pengenalan	Aplikasi ini mampu memberikan

	(2021)	Rambu Lalu Lintas Menggunakan Metode <i>Fuzzy</i> Mamdani Berbasis <i>Android</i>”	pemahaman tentang peraturan pasal rambu lalu lintas serta dapat memberikan pembelajaran terhadap masyarakat rambu- rambu yang ada di setiap perlintasan jalan. Logika <i>Fuzzy</i> merupakan teknik yang bisa membantu menaikkan kualitas interaksi pemain pengenalan rambu lalu lintas. Berdasarkan hasil pengujian menggunakan <i>Black Box</i>, sistem dikatakan berhasil dan berjalan sesuai apa yang diharapkan.
--	--------	---	---

3.	Cecep Supriatna, Sutono (2022)	“Media Sosialisasi Rambu-Rambu Lalu lintas Dengan Metode <i>Augmented Reality</i> Berbasis <i>Android</i> ”	Aplikasi ini bertujuan untuk mempermudah pihak kepolisian dalam menyampaikan informasi rambu-rambu lalulintas kepada masyarakat dan anak-anak yang kurang paham. Dengan teknologi augmented reality masyarakat akan melihat langsung arti dari rambu-rambu yang ada dari layar ponsel yang dimilikinya. Hasil dari penelitian ini dapat membantu masyarakat dan anakanak yang belum paham dalam mendapatkan informasi yang menarik, efektif karena media sosialilasi dapat digunakan dimana saja dan efisien karena tidak memerlukan biaya yang besar.
----	---	---	--

Ada banyak aplikasi tentang rambu-rambu lalu lintas yang telah digunakan di dunia. Akan tetapi metode dan media yang digunakan tentunya tidak sama, pengaruh budaya, kebiasaan, perilaku masyarakat yang berbeda-beda dan juga perkembangan

teknologi yang pesat membuat banyak sarana yang memungkinkan untuk digunakan sebagai media pengingat.

Pertama, ada penelitian Setiadi, Lydia Ignacia, (2020) yang merancang Pembangunan Aplikasi Pembelajaran Rambu Lalu Lintas Berbasis Multimedia Interaktif. Persamaan penelitian kali ini adalah sama-sama memberikan informasi tentang rambu-rambu lalu lintas. Dalam aplikasi ini, juga memberikan penjelasan rambu lalu lintas yang ada. Perbedaan dari penelitian yang dilakukan oleh penulis adalah media yang digunakan berupa *smartphone* berbasis *mobile*. Sedangkan peneliti terdahulu, media yang digunakan berupa *desktop* dan hanya menampilkan rambu lalu lintas.

Kedua, penelitian yang dilakukan oleh Fitri Marisa, (2021) dari segi rambu lalu lintas dapat dilihat kesamaan dari penelitian tersebut dengan penelitian yang akan dilakukan. Perbedaaan dengan peneliti sekarang adalah teknologi dan metode yang digunakan. Jika dibandingkan dengan penelitian terdahulu tentunya lebih unggul dikarenakan penulis melakukan sebuah penelitian pengenalan rambu-rambu lalu lintas pada anak dengan menerapkan konsep gamifikasi untuk meningkatkan kemampuan pemahaman dan motivasi belajar.

Ketiga, penelitian yang dilakukan oleh Cecep Supriatna, Sutono (2022), terdapat kesamaan dengan penelitian di atas, yaitu sama sama sebagai media sosialisasi peraturan pengenalan rambu-rambu lalu lintas. Adapun yang menjadi perbedaannya yaitu teknologi yang digunakan, fitur yang disediakan, serta studi kasus pada penelitian diatas sedangkan sistem yang akan dibuat Aplikasi game edukasi, yang ber-genre *multiple choice*.

B. Tinjauan Teori

1. Aplikasi

Aplikasi berasal dari kata *application* yaitu bentuk benda dari kata kerja *to apply* yang dalam Bahasa Indonesia berarti pengolah. Secara istilah, aplikasi komputer adalah suatu sub kelas perangkat lunak komputer yang menggunakan komputer langsung untuk melakukan suatu tugas yang diinginkan pemakai. Menurut Kamus Besar Bahasa Indonesia (KBBI, 2015) Aplikasi adalah penerapan dari rancang sistem untuk mengolah data yang menggunakan aturan atau ketentuan bahasa pemrograman tertentu. Aplikasi adalah suatu program komputer yang dibuat untuk mengerjakan dan melaksanakan tugas khusus dari pengguna. Sementara menurut Hartono (2020) aplikasi adalah penggunaan dalam suatu komputer, instruksi (*instruction*) atau pernyataan (*statement*) yang disusun sedemikian rupa sehingga komputer dapat memproses *input* menjadi *output*.

2. Game Edukasi (*Education Game*)

Game edukasi adalah salah satu genre game yang digunakan untuk memberikan pengajaran/ menambah pengetahuan penggunanya melalui suatu media unik dan menarik. Ada beberapa kriteria game edukasi yang ideal yaitu: (Nurrun Muchammad, 2018)

- a. Rasa ingin tahu, fantasi dan control pengguna
 - 1) Motivasi intrinsik, menyenangkan untuk digunakan. Pengguna memiliki kontrol terhadap permainan.
 - 2) Integrasi antara materi edukasi dan aspek fantasi permainan.

- 3) Mendorong keingintahuan, pengguna dapat melakukan eksplorasi bebas, permainan mengandung rahasia tersembunyi.
- 4) Keberhasilan ditentukan oleh pengetahuan, bukan kebetulan.
- 5) Simulasi realistik dunia.
- 6) Materi edukasi disesuaikan dengan materi dunia nyata.

b. Tantangan

- 1) Tantangan diberikan secara terus menerus tapi disesuaikan dengan tingkat pemahaman pemain.
- 2) Menyediakan hint dan instruksi untuk membantu pengguna.

c. Pengguna anak-anak dan yang berkebutuhan khusus

- 1) Gambar, objek, layar tertata rapi.
- 2) Permainannya sendiri merupakan aktivitas yang penting (*play for the sake of play*).
- 3) Menginspirasi anak, bahkan setelah komputer dimatikan.

Game berasal dari bahasa Inggris yang berarti permainan. Menurut Greg Costikyan (2016 : 20), game adalah sebarang karya seni di mana peserta, yang disebut pemain, membuat keputusan untuk mengelola sumber daya yang dimilikinya melalui benda di dalam game demi mencapai tujuan. Joan Freeman dan Utami Munandar (dalam Andang Ismail, 2019: 27) mendefinisikan permainan sebagai suatu aktifitas yang membantu anak mencapai perkembangan yang utuh, baik fisik, intelektual, sosial, moral, dan emosional.

Berdasarkan uraian di atas maka dapat disimpulkan bahwa Game ataupun permainan adalah sebuah aktifitas yang dilakukan satu atau lebih pemain dengan

aturan tertentu sehingga ada yang dapat menang ataupun kalah dengan tujuan bersenang-senang, mengisi waktu luang atau *refreshing*. game dapat mengasah kecerdasan dan keterampilan otak dalam mengatasi konflik atau permasalahan buatan yang ada dalam permainan. Pemain dihadapkan dengan sistem dan konflik buatan. Konflik atau masalah dalam setiap game berbeda-beda. Konflik dalam setiap game menuntut pemain untuk menyelesaikannya dengan cepat dan tepat sehingga dapat meningkatkan konsentrasi dan melatih otak untuk memecahkan masalah dengan tepat dan cepat.

Kata edukasi berasal dari bahasa Inggris yaitu *education* yang berarti pendidikan. Menurut Marc Prensky (2021 : 90), game edukasi adalah game yang didesain untuk belajar, tapi juga bisa menawarkan bermain dan bersenang-senang. Game edukasi adalah gabungan dari konten edukasi, prinsip pembelajaran, dan game *computer*. Adapun dampak game edukasi Menurut Edward, game edukasi memberikan banyak dampak positif dalam penerapannya antara lain :

- a. Game banyak digunakan orang untuk mengajarkan suatu pengetahuan dan membangun keterampilan baik dibidang edukasi, bisnis maupun militer.
- b. Game efektif digunakan untuk membangun kemampun matematika dan membaca pada anak.
- c. Game terbukti efektif untuk membantu anak-anak penderita asma dan diabetes mengelola kebiasaan hidup sehat.

Berdasarkan uraian di atas maka dapat disimpulkan game edukasi merupakan salah satu jenis game yang tidak hanya bersifat menghibur tetapi didalamnya mengandung pengetahuan yang disampaikan kepada penggunanya.

Game edukasi dapat digunakan sebagai salah satu media pendidikan yang bisa digunakan sebagai media pembelajaran. Game jenis ini biasa digunakan untuk mengajak penggunanya belajar sambil bermain. Melalui proses belajar ini maka penggunanya dapat memperoleh ilmu pengetahuan, sehingga game edukasi merupakan terobosan baru yang digunakan dalam dunia pendidikan. Selain dikarenakan game jenis ini memadukan antara sisi belajar dan bermain, game jenis ini juga dapat digunakan untuk menarik pendidik untuk ikut belajar.

3. Rambu Lalu Lintas

Rambu lalu lintas merupakan peralatan yang dipergunakan agar kendaraan dapat berjalan dengan cepat, aman, selamat, lancar, tertib, dan teratur. Rambu lalu lintas yang terdiri dari empat jenis diatas digunakan agar pemakai jalan merasa nyaman dan aman saat sedang berkendara atau berjalan di jalan raya. Rambu lalu lintas biasa ditempatkan di tempat-tempat yang sering terjadi kecelakaan atau ketidaktertiban pengguna jalan. Contoh dari pemasangan rambu lalu lintas antar lain adalah lampu lalu lintas yang dipasang di persimpangan jalan untuk mengatur lalu lintas kendaraan bermotor, rambu peringatan kereta api ditempatkan pada perlintasan jalan dengan jalur rel kereta api, dan lain-lain. Zulfiar (2020:51) menyebutkan bahwa penempatan rambu lalu lintas ditentukan setelah dipelajari kondisi jalan dan lalu lintas, umumnya diletakkan agar mudah dilihat oleh pengemudi. Marka jalan biasanya berbentuk garis yang bisa berupa petunjuk atau menunjukkan kondisi suatu tempat atau pembatas.

Pasal 21(1) UU No. 14/1992 dalam buku yang berjudul “Rekayasa Lalu Lintas” karya Leksmono Suryo Putranto (2008:117) tentang Lalu Lintas dan Angkutan Jalan, tata cara berlalu lintas di jalan (*right of away*) adalah dengan mengambil jalur sebelah kiri.

Batas kecepatan diatur dalam Pasal 80 PP No. 43/1993, marka jalan dijelaskan dalam Pasal 19 (1) PP No. 43/1993 dan untuk rambu-rambu lalu lintas diatur dalam pasal 17 PP No. 43/1993 yang terdiri atas empat golongan (Lekmono, 2008:117):

- a. Rambu peringatan (sebagian besar berwarna dasar kuning), digunakan untuk menyatakan peringatan bahaya atau tempat berbahaya.
- b. Rambu larangan (sebagian besar berwarna dasar putih dan bergaris tepi merah), digunakan untuk menyatakan perbuatan yang dilarang dilakukan oleh pemakai jalan.
- c. Rambu perintah (sebagian besar berwarna dasar biru), digunakan untuk menyatakan perintah yang wajib dilakukan oleh pemakai jalan.
- d. Rambu petunjuk (sebagian berwarna dasar putih bergaris tepi biru, berwarna dasar hijau atau cokelat), digunakan untuk menyatakan petunjuk mengenai jurusan, jalan, situasi, kota, tempat, pengaturan, fasilitas dan lain-lain.



Gambar 2.1 Contoh Rambu Lalu Lintas

Macam-macam rambu peringatan antara lain adalah rambu lampu lalu lintas, rambu persimpangan jalan, rambu jalan menanjak dan menurun, dan lainlain. Contoh dari rambu larangan adalah rambu larangan berhenti, rambu larangan parkir, rambu larangan melewati jalan, dan lan-lain. Contoh rambu perintah adalah rambu untuk berbelok kanan/kiri, rambu untuk melewati jalan berputar, dan lain-lain. Sedangkan

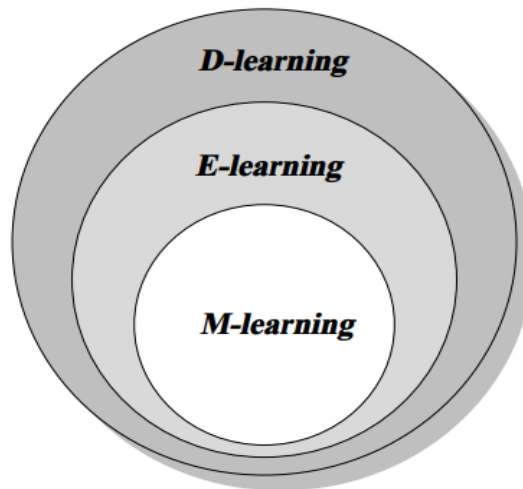
untuk rambu petunjuk, contohnya adalah rambu petunjuk jalan, rambu petunjuk jalan buntu, rambu awal daerah/wilayah kota atau akhir daerah kota, dan lain-lain.

Jadi dapat disimpulkan rambu lalu lintas adalah bagian dari perlengkapan jalan yang dimana memuat lambang, huruf, angka, kalimat dan/atau perpaduan diantaranya, yang digunakan untuk memberikan peringatan, larangan, perintah dan petunjuk bagi pemakai jalan. Agar rambu dapat terlihat baik siang ataupun malam atau pada waktu hujan maka bahan harus terbuat dari material retro-reflektif.

4. *Mobile Learning*

Mobile learning merupakan istilah untuk materi pembelajaran maupun aktifitas belajar yang disampaikan dengan menggunakan media perangkat bergerak yang mengakomodasi keterbatasan penyampaian multimedia, terutama dalam bentuk suara, gambar, animasi dan teks (Susan, 2019).

Mobile Learning (m-learning) memiliki suatu arti yang berbeda-beda, tergantung bagaimana setiap komunitas tersebut mendefinisikannya. Istilah dari *m-learning* sendiri mengacu pada penggunaan perangkat teknologi informasi (TI) genggam dan bergerak, seperti *Personal Digital Assistant* (PDA), telepon genggam, laptop, dan tablet PC dalam pengajaran dan pembelajaran. *M-learning* merupakan bagian dari *electronic learning (e-learning)* yang juga merupakan bagian dari *distance learning (d-learning)*, seperti yang dapat dilihat pada gambar berikut:



Gambar 2.1 Skema *m-Learning*

Seperti yang terlihat pada gambar 2.2, karena konsep dari *m-learning* merupakan bagian dari *e-learning*, maka metode pembelajarannya ini dibagi menjadi 2 tipe, yaitu : (Efendi dan Zhuang, 2015)

a. *Synchronous Training*

Synchronous berarti pada waktu yang sama. Jadi, *synchronous training* adalah tipe pembelajaran, dimana proses pembelajaran terjadi pada saat yang sama ketika pengajar sedang mengajar dan pelajar sedang belajar. Tipe ini lebih banyak digunakan untuk seminar atau konferensi dan sering digunakan untuk kuliah universitas *online*.

b. *Asynchronous Training*

Asynchronous berarti tidak pada waktu yang sama. Jadi, *asynchronous Training* adalah tipe pembelajaran, dimana pelajar dapat mengambil pelajaran pada waktu yang berbeda dengan pengajar memberikan pelatihan. Pelatihan ini lebih

dikenal di dunia mobile learning, karena memberikan keuntungan lebih bagi peserta pelatihan, karena dapat mengakses pelatihan kapanpun dan dimanapun.

Namun secara umum, *mobile learning* merupakan pembelajaran bergerak yang memanfaatkan teknologi *mobile smartphone* sehingga para pembelajar dapat belajar dimana pun tanpa adanya batasan waktu dan tempat. Karakteristik dari *mobile learning* yang perlu diperhatikan adalah :

- a. *Ubiquitous*, yang artinya materi *mobile learning* bisa diakses di mana saja, berkaitan dengan lokasi belajar.
- b. *Bite sized*, yang artinya ukuran dari materi *mobile learning* yang diakses harus disampaikan dalam durasi yang singkat.
- c. *On demand*, yang artinya *mobile learning* harus sanggup menyampaikan materi dikala dibutuhkan oleh siswa. Memaksimalkan penyampaian materi pembelajaran saat dibutuhkan oleh siswa.
- d. *Typically blended*, yang artinya *mobile learning* sudah biasa digunakan bersama metode pembelajaran yang lain. *Mobile Learning* cenderung digunakan sebagai alat bantu untuk meningkatkan pemahaman siswa selain yang telah disampaikan dengan metode lain, misal belajar di kelas.

C. Tinjauan Perangkat Lunak

1. *Android*



Gambar 2.3 Logo *Android*

(Sumber : <https://2.bp.blogspot.com/>)

Android merupakan OS (*Operating System*) *mobile* yang tumbuh ditengah OS lainnnya yang berkembang pada zaman sekarang (Hermawan, 2019). OS lainnnya seperti *windows mobile*, *Iphone OS*, *Symbian*, dan masih banyak lagi. Akan tetapi, OS ini berjalan dengan memprioritaskan aplikasi inti yang dibangun sendiri tanpa melihat potensi yang cukup besar dari aplikasi pihak ketiga. Oleh karena itu, adanya keterbatasan dari aplikasi pihak ketiga untuk mendapatkan data asli ponsel, berkomunikasi antar proses serta keterbatasan distribusi aplikasi pihak ketiga untuk *platform* mereka. Selain itu *android* juga merupakan sistem operasi yang dirancang oleh *Google* dengan basis *kernel linux* untuk mendukung kinerja perangkat elektronik layar sentuh (*touch screen*), seperti tablet atau *smartphone*. *Android* memiliki sifat *open source* atau bebas jika digunakan, dimodifikasi, diperbaiki dan didistribusikan oleh para pembuat ataupun pengembang perangkat lunak.

2. *Flutter*



Gambar 2.4 Flutter

(Sumber : <https://www.cloudsavvyit.com/>)

Flutter merupakan salah satu *framework* atau *Software Development Kit* (SDK) untuk pengembangan aplikasi *mobile* yang dapat berjalan di sistem operasi *ios* dan *android*. *Flutter* diciptakan oleh *Google* dan dirilis pertama kali pada bulan Mei 2017 (versi alpha). Pada Desember 2018, *Google* meluncurkan *flutter* 1.0.0 (versi stabil). *Flutter* menggunakan bahasa pemrograman *dart*, yang juga diciptakan oleh *Google* dan dirilis pertama kali pada bulan Oktober 2013. *Flutter* juga merupakan *framework* baru yang dirancang untuk mempermudah dan mempercepat proses pembuatan aplikasi *mobile*, yang dulunya harus ditulis menggunakan *objective-c* atau *swift* (untuk *ios*) dan *java* atau *kotlin* (untuk *android*). Dengan *flutter*, kita hanya perlu mempelajari satu bahasa pemrograman (*dart*) untuk membuat aplikasi *mobile* yang dapat berjalan di dalam dua sistem operasi *ios* dan *android*.

3. *Android Studio*



Gambar 2.5 *Android Studio*

(Sumber : <https://trainingeltasa.com/>)

Android Studio merupakan sebuah *Integrated Development Environment* (IDE) khusus untuk membangun aplikasi yang berjalan pada *platform android*. *Android studio* ini berbasis pada *IntelliJ IDEA*, sebuah IDE untuk bahasa pemrograman *Java*. Bahasa pemrograman utama yang digunakan adalah *Java*, sedangkan untuk membuat tampilan atau *layout*, digunakan bahasa XML. *Android studio* juga terintegrasi dengan *Android Software Development Kit* (SDK) untuk *deploy* ke perangkat *android*. *Android Studio* juga merupakan pengembangan dari *eclipse*, dikembangkan menjadi lebih kompleks dan lebih profesional yang telah tersedia didalamnya *Android Studio IDE*, *Android SDK tools* (Smith, 2015).

Setiap proyek di *Android Studio* berisi satu atau beberapa modul dengan file kode sumber dan file sumber daya. Jenis-jenis modul mencakup suatu modul aplikasi *android*, modul pustaka dan modul *google APP engine*.

4. *Java Development Kit (JDK)*



Gambar 2.6 *Java Development Kit (JDK)*
(Sumber : <https://cdn.educba.com/>)

Java Development Kit (JDK) merupakan perangkat lunak yang digunakan untuk manajemen dan membangun berbagai aplikasi *Java*. *JDK* merupakan *superset* dari *JRE*, berisikan segala sesuatu di *JRE* ditambahkan *compiler* dan *debugger* yang diperlukan untuk membangun aplikasi. *Android Studio* SDK dikembangkan dengan menggunakan bahasa pemrograman *java*. Demikian aplikasi *android* juga dikembangkan *Java*. Sehingga *Java Development Kit (JDK)* adalah komponen yang harus diinstal. Pengembangan *android* membutuhkan instalasi baik versi 6 atau 7 Edisi Standar *Kit Java Platform* Pembangunan. *Java* disediakan di kedua paket pengembangan (*JDK*) dan *runtime (JRE)*.

5. *Software Development Kit*



Gambar 2.7 *Software Development Kit (SDK)*
(Sumber : <https://cdn.educba.com/>)

Java Development Kit (JDK) adalah distribusi Teknologi *Java* oleh *Oracle Corporation*, pada proses pengembangan dan pembuatan sebuah aplikasi *Android*, terdapat bantuan untuk membuatnya yaitu dengan menggunakan suatu perangkat bantuan berbasis perangkat lunak yang dikenal sebagai *Android SDK*. *Android SDK* merupakan suatu perangkat API (*Application Programming Interface*) yang dibutuhkan untuk memulai pengembangan dan pembuatan aplikasi pada *platform Android*, dalam proses tersebut digunakan bahasa pemrograman *Java*. Dengan adanya *Android SDK* dapat membantu *developer* untuk mengembangkan aplikasi yang bukan merupakan bawaan dari *mobilephone* atau *smartphone* tersebut.

6. *Visual Studio Code*



Gambar 2.8 *Visual Studio Code*
(Sumber : <https://trainingeltasa.com/>)

Visual Studio Code adalah suatu editor *source code* yang dikembangkan oleh Microsoft untuk *Windows*, *Linux* dan *MacOS*. Ini termasuk dukungan untuk *debugging*, *GIT Control* yang disematkan, penyorotan *sintaks*, penyelesaian kode cerdas, cuplikan, dan kode *refactoring*. Hal ini juga dapat disesuaikan, sehingga pengguna dapat mengubah tema editor, *shortcut keyboard*, dan preferensi. *Visual Studio Code* gratis dan *open-source*, meskipun unduhan resmi berada di bawah lisensi *proprietary* (Lardinois, 2021).

D. Tahapan Perancangan Sistem

1. Use Case

Use case diagram adalah abstraksi dari interaksi antara sistem dan *actor*.

Use case bekerja mendeskripsikan tipe interaksi antara *user* sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai.

Use Case Diagram merupakan pemodelan untuk melakukan (*behavior*) sistem informai yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu (Rosa dan Shalahuddin, 2018).

2. Kamus Data

Tabel 2.2 Simbol-Simbol Dalam Kamus Data

No	Simbol	Arti
1	=	Disusun atau terdiri atas
2	+	Dan
3	[I]	Baik ...atau...
4	{ <i>n</i> }	n kali diulang/ bernilai banyak
5	()	Data operasional
6	*...*	Batas komentar

Kamus data (*data dictionary*) adalah katalog fakta tentang data dan kebutuhan informasi yang mengalir pada sistem dengan lengkap dari suatu sistem informasi. Kamus data di buat berdasarkan arus data yang mengalir pada *Data*

Flow Diagram (DFD). Sukamto dan Shalahuddin, (2019), "Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar cara penulisan)".

Kamus data merupakan tempat penyimpanan dari aliran-aliran data, *file-file* dan proses-proses dalam sebuah sistem. Bagian ini menjelaskan secara detail proses-proses yang terjadi disetiap proses, *file* dan struktur data mengenai model sistem yang digambarkan. Kamus data digunakan untuk menjelaskan semua data yang mengalir atau digunakan dalam sistem, yaitu mengenai arus data yang masuk ke dalam sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem.

Kamus data adalah repositori (*manual atau computed-based*) yang berisi informasi tentang berbagai objek data muncul pada setiap UML. tidak ada format standar untuk kamus data. Notasi menyimpan data dalam UML menunjukkan adanya salah satu item data yang lebih dari data yang tersimpan. Sebuah kamus data hanya dapat memberikan gambaran sederhana dari data. (Sharma, 2020)

Ada berbagai jenis data, masing-masing mampu memegang hanya jenis tertentu dari informasi :

- a. *Integer*: mampu menahan hanya seluruh nomor tunggal.
- b. *Real* : mampu menahan 1 angka dengan *decimal*.
- c. *Date*: hanya bisa menampung tanggal.
- d. *String*: bisa menahan teks, termasuk nomor dan tanda baca.

- e. *Boolean*: hanya bisa memegang nilai ‘*true*’ or ‘*false*’. Sebuah kamus data yang khas terlihat seperti ini.

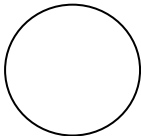
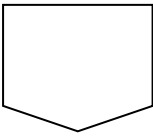
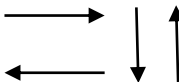
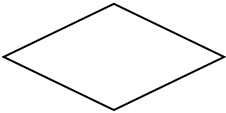
3. Bagan Alir Program

Bagan alir program merupakan bagan yang menjelaskan rinci langkah-langkah dari proses program. Bagan alir program dibuat dari derifikasi bagan alir sistem. Menurut Jogiyanto (2015) “Bagan alir program (*program flowchart*) merupakan bagan alir yang mirip dengan suatu bagan alir sistem, yaitu untuk menggambarkan prosedur di dalam sistem”

Supardi (2017), “*flowchart* merupakan bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika. Bagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi. Pada waktu akan menggambar suatu bagan alir”. Adapun simbol-simbol dari *Flowchart* adalah sebagai berikut :

Tabel 2.3 Simbol *Flowchart*

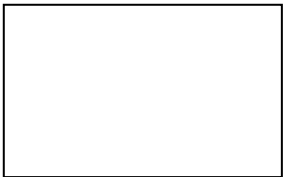
SIMBOL	KETERANGAN
	Simbol <i>input/output</i> digunakan untuk mewakili data <i>input/output</i> .
	Simbol proses, digunakan untuk mewakili suatu proses.
	Simbol proses terdefinisi, digunakan untuk menunjukkan suatu operasi yang rinciannya ditunjukkan ditempat lain.

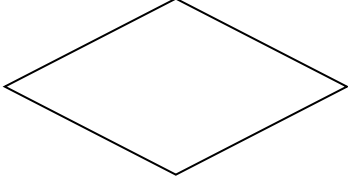
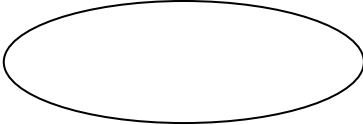
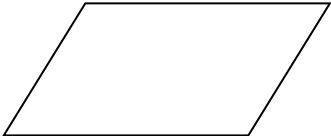
SIMBOL	KETERANGAN
	Simbol titik terminal, digunakan untuk awal dan akhir suatu proses.
	Simbol penghubung, digunakan untuk menunjukkan sambungan dari alir yang terputus di halaman yang masih sama.
	Simbol penghubung, digunakan untuk menunjukkan sambungan dari alir yang terputus di halaman yang berbeda.
	Simbol garis alir, menunjukkan arus dari proses.
	Simbol keputusan, digunakan untuk suatu penyelesaian kondisi didalam program.

4. *Entity Relationship Diagram*

Entity Relationship Diagram (ERD) merupakan komponen-komponen himpunan entitas dan himpunan relasi yang masing-masing dilengkapi atribut-atribut yang merepresentasikan seluruh fakta dari dunia nyata yang kita tinjau. Simbol-simbol yang sering digunakan dalam *entity relationship diagram* dapat dilihat pada tabel 2.4.

Tabel 2.4 *Simbol Entity Relationship Diagram*

NOTASI	KETERANGAN
	<i>Entitas</i> , yaitu kumpulan dari objek yang diidentifikasi secara unik.

	Relasi, yaitu hubungan yang terjadi antara satu atau lebih entitas. Jenis hubungan antara lain : satu ke satu, satu ke banyak dan banyak ke banyak.
NOTASI	KETERANGAN
	Atribut, yaitu karakteristik dari <i>entity</i> atau relasi yang merupakan penjelasan detail tentang <i>entitas</i> .
	Garis, hubungan antara <i>entity</i> dengan atributnya dan himpunan <i>entitas</i> dengan himpunan relasi.
	<i>Input/ Output</i> data, yaitu proses <i>input/ output</i> data, parameter, informasi.

Entity Relationship Diagram (ERD) adalah suatu pemodelan konseptual yang didesain secara khusus untuk mengidentifikasi *entitas* yang menjelaskan data dan hubungan antar data. *ERD* untuk memodelkan struktur data dan hubungan antar data, untuk menggambarkannya digunakan beberapa notasi dan simbol. Pada dasarnya ada tiga komponen yang digunakan, yaitu :

a. *Entity*

Entity merupakan objek yang mewakili sesuatu yang nyata dan dapat dibedakan dari sesuatu yang lain.

b. Atribut

Setiap *entitas* pasti mempunyai elemen yang disebut atribut yang berfungsi untuk mendeskripsikan karakteristik dari *entitas* tersebut.

c. Hubungan/ Relasi

Hubungan antara sejumlah entitas yang berasal dari himpunan *entitas* yang berbeda. Relasi dapat digambarkan sebagai berikut:

Relasi yang terjadi diantara dua himpunan entitas (misalnya A dan B) dalam satu basis data yaitu:

1) Satu ke satu (*One to one*)

Hubungan relasi satu ke satu yaitu setiap *entitas* pada himpunan entitas A berhubungan paling banyak dengan satu *entitas* pada himpunan *entitas* B.

2) Satu ke banyak (*One to many*)

Setiap entitas pada himpunan *entitas* A dapat berhubungan dengan banyak entitas pada himpunan *entitas* B, tetapi setiap *entitas* pada *entitas* B dapat berhubungan dengan satu entitas pada himpunan entitas A.

3) Banyak ke banyak (*Many to many*)

Setiap *entitas* pada himpunan *entitas* A dapat berhubungan dengan banyak *entitas* pada himpunan *entitas* B.

5. UML (*Unified Modelling Language*)

UML (*Unified Modelling Language*) menurut Munawar dalam buku “Pemodelan Visual dengan UML” (2015 : 17) menyatakan UML adalah salah satu alat bantu yang sangat handal didunia pengembangan sistem yang berorientasi obyek. Hal ini disebabkan karena UML menyediakan bahasa pemodelan *visual* yang memungkinkan bagi pengembang sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi (*sharing*) dan mengkomunikasikan rancangan mereka dengan yang lain.

Sedangkan menurut menurut Adi Nugroho (2019:6), “*Unified Modelling Language* adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berpradigma berorientasi objek”. Pemodelan (*modeling*) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami. Jurnal CCIT (2018:70) *Unified Modelling Language* (UML) didefinisikan oleh *Good Corporate* sebagai berikut :

Metode Unified Modelling Language (UML) menggunakan tiga bangunan dasar untuk mendeskripsikan suatu sistem atau perangkat lunak yang akan dikembangkan, yaitu :

a. Sesuatu (*things*)

Ada empat *things* dalam *Unified Modelling Language* (UML) :

- 1) *Structural things*, bagian yang relative statis dapat berupa elemen-elemen yang bersifat fisik maupun konseptual.

- 2) *Behaviorial things*, bagian dinamis biasanya merupakan kata kerja dari model UML yang mencerminkan perilaku sepanjang ruang dan waktu.
- 3) *Grouping things*, bagian pengorganisasian dalam UML. Dalam penggambaran model UML yang rumit diperlukan penggambaran paket yang menyederhanakan model. Paket-paket ini kemudian dapat didekomposisi lebih lanjut. Paket berguna bagi pengelompokan sesuatu, misalnya model-model serta *subsitem-subsistem*.
- 4) *An notational things*, merupakan bagian yang meperjelas model UML. Dapat berisi komentar yang menjelaskan fungsi serta ciri-ciri tiap element dalam model UML.

b. Relasi (*relationship*)

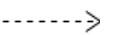
Untuk upaya tersebut UML menyediakan 9 jenis diagram yang dapat dikelompokkan berdasarkan sifatnya *statis* atau *dinamis*. Ke 9 jenis diagram dalam UML itu adalah :

- 1) Diagram kelas bersifat statis Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi-kolaborasi, serta relasi-relasi. Diagram ini umum dijumpai pada pemodelan sistem berorientasi objek. Meskipun bersifat statis, sering pula diagram kelas memuat kelas-kelas aktif.
- 2) Diagram objek bersifat statis. Diagram ini memperlihatkan objek serta relasi-relasi antar objek. Diagram objek memperlihatkan instalasi statis dari segala sesuatu yang dijumpai pada diagram kelas.

- 3) Diagram objek bersifat statis. Diagram ini memperlihatkan objek serta relasi-relasi antar objek. Diagram objek memperlihatkan instalasi statis dari segala sesuatu yang dijumpai pada diagram kelas.
- 4) *Use Case diagram* bersifat statis. Diagram ini memperlihatkan himpunan *use case* dan aktor-aktor. Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku dari suatu sistem yang dibutuhkan serta diharapkan pengguna.
- 5) *Sequence diagram* bersifat dinamis. Diagram urutan adalah diagram interaksi yang menekankan pada pengiriman pesan dalam suatu waktu tertentu.
- 6) *Statechart diagram* bersifat dinamis. Diagram ini memperlihatkan *state-state* pada sistem memuat *state*, *transisi*, *event*, serta aktifitas. Diagram ini terutama penting untuk memperlihatkan sifat dinamis dari antarmuka, kelas, kolaborasi dan terutama penting pada pemodelan sistem-sistem yang reaktif.
- 7) *Activity diagram* Bersifat dinamis. Diagram aktivitas ini adalah tipe khusus dari diagram *state* yang memperlihatkan aliran dari suatu aktivitas ke aktivitas lainnya dalam suatu sistem.
- 8) *Component diagram* bersifat statis. Komponen ini memperlihatkan organisasi serta kebergantungan sistem atau perangkat lunak pada komponen-komponen yang telah ada sebelumnya.
- 9) *Deployment diagram* bersifat statis. Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan. Diagram ini membuat simpul-

simpul beserta komponen-komponen yang ada di dalamnya. Diagram ini sangat berguna pada banyak mesin. (Sholih, 2016).

Tabel 2.5 *Table Use Case Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
3		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor

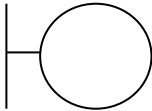
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

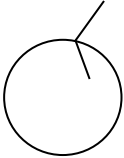
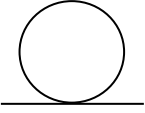
Tabel 2.6 *Symbol Activity Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi.
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

Tabel 2.7 *Symbol Sequence Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Objekct</i> <i>(Partisipan)</i>	<i>Object</i> atau biasa juga disebut partisipan merupakan <i>instance</i> dari sebuah <i>class</i> dan dituliskan tersusun secara <i>horizontal</i> . Digambarkan sebagai sebuah <i>class</i> (kotak) dengan

			nama objek di dalamnya yang diawali dengan sebuah titik koma.
2		<i>Actor</i>	<i>Actor</i> juga dapat berkomunikasi dengan <i>object</i> , maka <i>actor</i> juga dapat diurutkan sebagai kolom.
3		<i>Life line</i>	<i>Life line</i> mengidentifikasi keberadaan sebuah object dalam basis waktu. Notasi untuk <i>lifeline</i> adalah garis putus-putus <i>vertical</i> yang di tarik oleh sebuah <i>object</i>
4		<i>Collaborations</i>	<i>Activation</i> dinotasikan sebagai kotak segi empat yang digambar pada sebuah <i>lifeline action</i> mengidentifikasi sebuah <i>object</i> yang akan beraksi.
NO	GAMBAR	NAMA	KETERANGAN
5		<i>Boundary</i>	<i>Boundary</i> terletak diantara system dengan dunia di sekelilingnya. Semua <i>form</i> , laporan-laporan, antara muka ke perangkat keras seperti <i>printer</i> atau <i>scenner</i> dan antara muka

			ke system lainnya adalah termasuk dalam kategori.
6		<i>Control</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		<i>Entity</i>	<i>Entity</i> digunakan untuk menengani informasi yang akan disimpan secara permanen.

E. Metode Pengujian Perangkat Lunak

Untuk metode pengujian yang dilakukan pada penelitian ini adalah metode pengujian langsung yaitu dengan menggunakan pengujian *White Box* dan *Black Box* *White box testing* adalah pengujian yang didasarkan pada pengecekan terhadap detail perancangan, menggunakan struktur kontrol dari desain program secara *procedural* untuk membagi pengujian ke dalam beberapa kasus pengujian. *Black Box* Digunakan untuk menguji fungsi-fungsi khusus dari perangkat lunak yang dirancang.

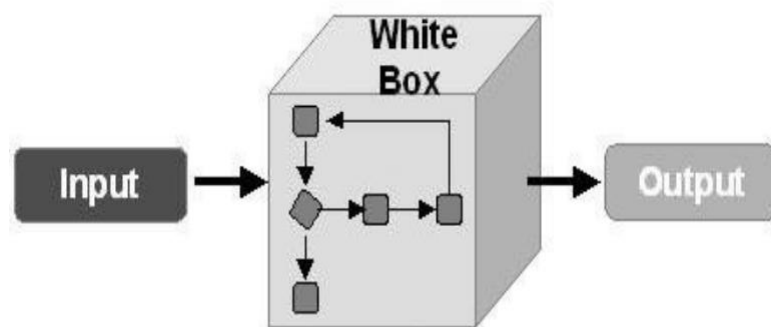
Beberapa *test-case* harus dilaksanakan dengan beberapa perbedaan strategi, *query*, atau jalur navigasi yang mewakili penggunaan sistem yang *tipical*, kritis atau abnormal. Pengujian harus mencakup *unit testing*, yang mengecek *validasi* dari *prosedur* dan fungsi-fungsi secara *independen* dari komponen sistem

yang lain. Kemudian modul *testing* harus menyusul dilakukan untuk mengetahui apakah penggabungan beberapa unit dalam satu modul sudah berjalan dengan baik, termasuk eksekusi dari beberapa modul yang saling berelasi, apakah sudah berjalan sesuai karakteristik sistem yang diinginkan.

Jika struktur kendali antar modul sudah terbukti bagus, maka pengujian yang tak kalah pentingnya adalah pengujian unit. Pengujian unit digunakan untuk menguji setiap modul untuk menjamin setiap modul menjalankan fungsinya dengan baik. Ada 2 metode untuk melakukan unit *testing*, yaitu :

1. *White Box Testing*

Uji coba *white box testing* merupakan metode perancangan *test case* yang menggunakan struktural untuk mendapatkan *test case*, *test* ini digunakan untuk meramal cara kerja perangkat lunak secara rinci kepada *logic path* (jalur logika), perangkat lunak di *test* dengan kondisi dan perulangan secara fisik.

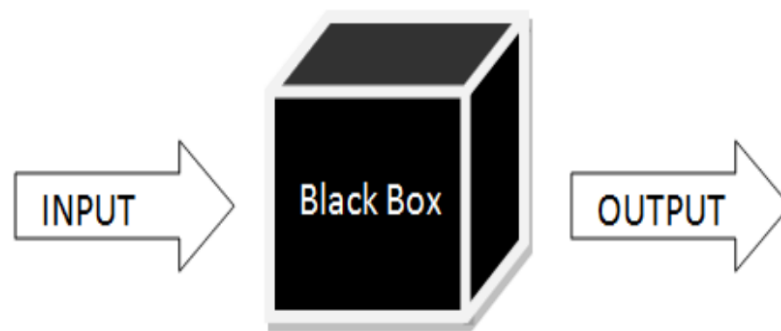


Gambar 2.9 Pengujian *White Box*
(Sumber : <https://www.codester.com/>)

Contoh pengujian *white box testing* ini merupakan peringatan ketika *user* menginputkan *password user* yang salah, untuk kesalahan semacam ini akan memberikan suatu informasi kepada *user* mengenai kesalahan yang dilakukan.

2. *Black Box Testing*

Black box testing adalah pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. Jadi dianalogikan seperti kita melihat suatu kotak hitam, kita hanya bisa melihat penampilan luarnya saja, tanpa tau ada apa dibalik bungkus hitamnya. Sama seperti pengujian *black box testing*, mengevaluasi hanya dari tampilan luarnya (*interfacenya*), fungsionalitas tanpa mengetahui apa sesungguhnya yang terjadi dalam proses detilnya (hanya mengetahui *input* dan *output*).



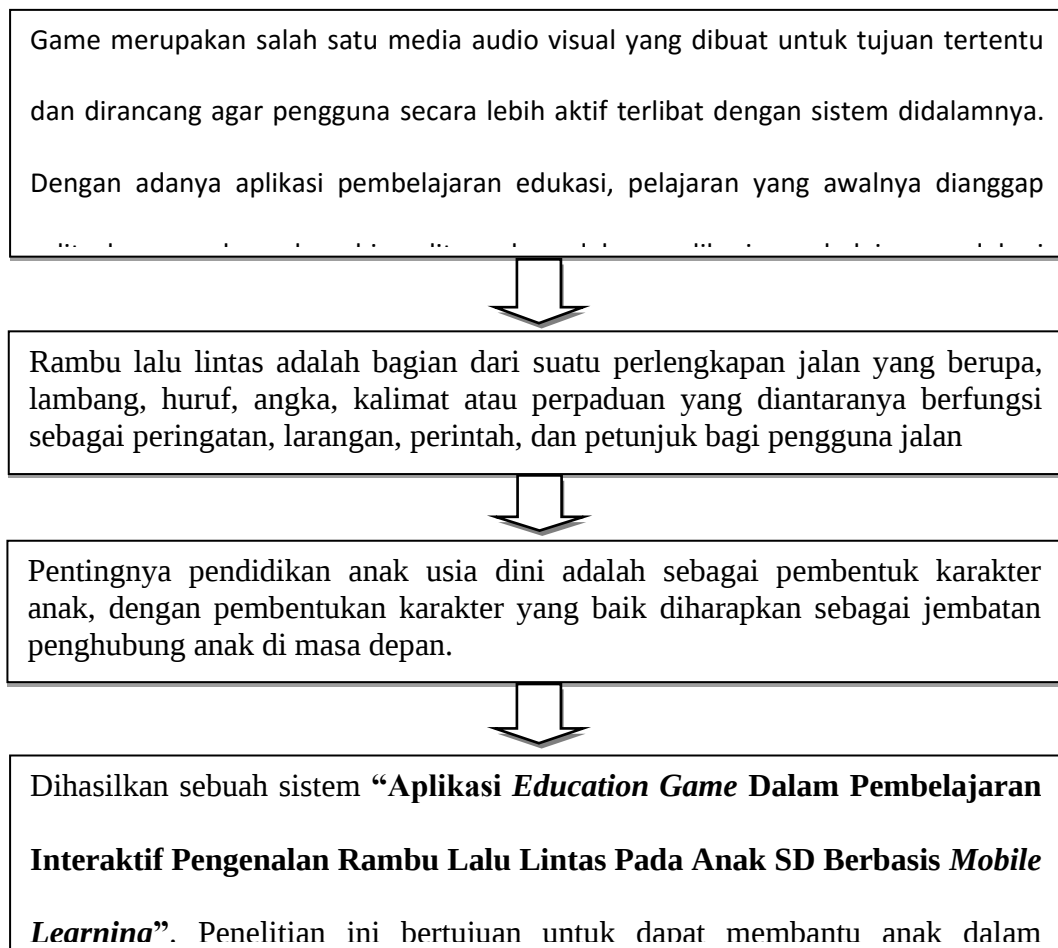
Gambar 2.10 Pengujian *Black Box*
(Sumber : <https://www.codester.com/>)

Teknik yang digunakan dalam *Black Box Testing* antara lain:

- a. Digunakan untuk menguji fungsi-fungsi khusus dari perangkat lunak.
- b. Kebenaran perangkat lunak yang diuji hanya dilihat berdasarkan keluaran (*output*) yang dihasilkan.
- c. Kemampuan program dalam memenuhi kebutuhan pemakai dapat diukur sekaligus dapat diketahui kesalahan-kesalahannya.

F. Kerangka Pikir

Untuk memperjelas suatu kerangka berpikir yang akan dituliskan dalam penelitian akhir ini. Maka berikut digambarkan kerangka pikir tersebut.



Gambar 2.11 Kerangka Pikir

BAB III

METODOLOGI PENELITIAN

A. Tempat dan Waktu Penelitian

1. Lokasi penelitian berada di Kota Parepare, Sulawesi Selatan 91112.
Kota Parepare terletak di sebuah teluk yang menghadap ke Selat Makassar. Kota Parepare merupakan salah satu daerah di Sulawesi Selatan yang memiliki posisi strategis dengan luas 99,33 km².
2. Lokasi penelitian kedua dilakukan oleh penulis bertempat pada UPTD SD Negeri 15 Kota Parepare, Jl. Kesuma Utara, Kp. Baru, Kec. Bacukiki Barat, Kota Parepare, Sulawesi Selatan 91111.
3. Pelaksanaan penelitian selama ($\pm$) 2 bulan, terhitung pasca seminar proposal.

B. Metode Penelitian

Metode yang digunakan dalam penelitian ini terdiri dari tahapan metode pengumpulan data, metode pengembangan sistem dan metode pengujian *black box testing*. Metode pengumpulan data yang digunakan adalah teknik wawancara, observasi dan studi pustaka. Pengembangan sistem dilakukan menggunakan metode *prototipe* dan untuk pengujian sistem digunakan *black box testing* dengan uji

fungsional sistem. Untuk membantu suatu kelancaran pengumpulan data, maka penulis menggunakan beberapa metode antara lain :

1. Pengumpulan Data

Untuk memperoleh data yang diperlukan dalam penulisan skripsi ini, penulis menggunakan beberapa metode antara lain :

- a. Metode Pengamatan (*Observation*) Metode pengumpulan data dengan cara pengamatan secara langsung pada objek-objek permasalahan yang akan di teliti untuk memperoleh informasi.
- b. Metode Wawancara (*Interview*) Metode pengumpulan data melalui tanya jawab langsung kepada target pengguna aplikasi mengenai masalah yang erat hubungannya dengan penelitian ini.
- c. Metode Kepustakaan (*Library*) Metode pengumpulan data ini dengan membaca buku-buku atau literatur-literatur yang berhubungan dengan permasalahan sebagai referensi dan bahan pembanding.

2. Analisis

Tahap analisa dilakukan dengan melakukan analisa pada beberapa hal, adapun proses analisisnya dijelaskan sebagai berikut :

a. Analisa Masalah

Tahap ini menjelaskan tentang apa saja masalah yang terjadi dalam proses perancangan aplikasi.

b. Penyelesaian Masalah

Tahap ini merupakan tahap lanjutan dari proses analisa masalah. Dalam tahap ini dijabarkan solusi penyelesaian untuk mengatasi masalah.

c. Analisa Sistem Berjalan

Pada tahap ini dijelaskan mengenai suatu proses awal bagaimana cara anak usia dini dalam hal ini anak SD memperoleh informasi umum mengenai jenis-jenis rambu-rambu lalu lintas.

d. Analisa Sistem Usulan

Pada tahap ini dijelaskan mengenai sebuah sistem yang diusulkan untuk menyelesaikan masalah sesuai dengan masalah yang dijabarkan pada sub bab analisa masalah, tahapan ini digambarkan dalam bentuk UML.

e. Analisa Kebutuhan Aplikasi

Pada tahap ini dijelaskan hal-hal apa saja yang dibutuhkan oleh aplikasi seperti alat perangkat keras, perangkat lunak dan perangkat-perangkat pendukung lainnya. Dan juga pada tahap ini dijelaskan apa saja yang menjadi kebutuhan fungsional dan *non* fungsional dari Aplikasi *Education Game* Dalam Pembelajaran Interaktif Pengenalan Rambu Lalu Lintas Pada Anak SD Berbasis *Mobile Learning*.

f. Analisa Data

- 1) *Identify* (Mengidentifikasi masalah) merupakan suatu langkah pertama yang dilakukan dalam tahap analisis data. Masalah yang dapat diidentifikasi sebagai suatu pertanyaan yang ingin dipecahkan.
- 2) *Understand* (Memahami kerja sistem yang ada), langkah ini dapat dilakukan dengan mempelajari secara terinci bagaimana sistem yang sudah ada. Dengan begitu kita dapat mengetahui beberapa kelemahan atau masalah yang ada pada sistem tersebut.
- 3) *Analyze* (Menganalisis hasil) Pada tahap ini dilakukan hasil penelitian yang telah dilakukan. Sehingga dapat mengetahui sistem seperti apa yang seharusnya dibuat.

3. Perancangan Sistem

Proses perancangan sistem dibagi dalam 5 tahap, yaitu :

- a. Perancangan *Input*, menentukan *input* atau masukan sistem, yaitu *input* atau masukan seperti apa yang nantinya dapat dimasukkan ke dalam sistem untuk dapat diproses.
- b. Perancangan Proses, sistem merancang proses yang akan terjadi di dalam sistem. Hal ini dilakukan agar proses sistem yang terjadi dapat memproses data dengan lancar dan teratur sehingga menghasilkan informasi yang benar.
- c. Perancangan *Output*, menentukan *output* atau keluaran yang akan dihasilkan oleh sistem, agar *output* yang dihasilkan dapat memudahkan setiap unsur manusia yang membutuhkannya.

- d. Perancangan Database, langkah yang dilakukan dalam perancangan database yaitu menentukan kebutuhan file database untuk sistem yang akan dibuat dan menentukan parameter dari file database itu.
- e. Menulis Program (Pengcodingan program), Penulisan program ini harus sesuai dengan desain database dan desain proses serta *input* sistem yang telah ditentukan sebelumnya. Sehingga sistem yang dihasilkan (dibuat) sesuai dengan ketentuan.

4. Implementasi Sistem

Implementasi adalah suatu proses untuk memastikan terlaksananya suatu

kegiatan dan tercapainya tujuan dari kegiatan tersebut. Keberhasilan implementasi dapat dilihat dari program tersebut dirancang dengan landasan yang jelas, dengan kelompok sasaran dan tujuan yang jelas.

- a. Instalasi Perangkat Lunak (*software*) yang telah dibuat oleh penulis akan dibahas di bab 4, seperti melakukan penginstalan aplikasi yang sudah dibuat ke *smartphone*.
- b. Pengujian (menggunakan metode *blackbox*) Pengujian dilakukan dengan cara uji coba *blackbox testing* yaitu uji coba yang dilakukan pada *interface software*, dan digunakan untuk mendemonstrasikan fungsi *software* yang dioperasikan, apakah *input* yang diterima dengan benar dan *output* yang dihasilkan benar.

5. Pengujian Program

Meliputi program dan sistem secara keseluruhan, apakah sistem yang dihasilkan sesuai dengan yang diinginkan atau belum, sehingga program tersebut benar-benar siap untuk digunakan.

C. Metode Pengumpulan Data

Pada tahap ini peneliti melakukan wawancara dengan pihak yang terkait dengan penelitian, serta melakukan pencatatan dan pengamatan (*observation*). Tahapan suatu penelitian yang dimaksud dalam penelitian ini ada beberapa tahapan. Adapun uraian dari tahapan tersebut adalah sebagai berikut :

1. Persiapan Penelitian

Pada tahapan ini peneliti melakukan persiapan penelitian. Persiapan penelitian yang dimaksud adalah menyiapkan buku-buku, artikel-artikel, serta *software* yang digunakan selama penelitian.

2. Studi Literature

Pada tahapan ini peneliti melakukan apa yang disebut dengan kajian pustaka, yaitu dengan mempelajari buku-buku referensi dan hasil penelitian sejenis sebelumnya yang pernah dilakukan oleh orang lain.

3. Pengumpulan Data

Pada tahap ini peneliti melakukan wawancara dengan pihak yang terkait dengan penelitian, serta melakukan pencatatan dan pengamatan langsung di tempat penelitian.

4. Analisis

Pada tahap analisis, peneliti melakukan analisa terhadap sistem yang di terapkan sekarang berdasarkan kemudian merumuskan masalah yang menjadi pokok penelitian sehingga dapat dibuat alternatif pemecahan masalah.

5. Perancangan

Peneliti kemudian merancang aplikasi yang ingin dibuat berdasarkan alternatif pemecahan masalah.

6. Pengujian

Setelah melakukan perancangan, peneliti kemudian menguji hasil perancangan yang telah dibuat. Jika hasil perancangan terdapat kekurangan atau kelemahan maka kembali ke tahap analisis.

D. Sumber Data Penelitian

Menurut Suharsimi Arikunto (2021:265), instrumen pengumpulan data adalah alat bantu yang dipilih dan digunakan oleh peneliti dalam kegiatannya mengumpulkan data agar kegiatan tersebut menjadi sistematis dan dipermudah olehnya. Adapun jenis dan sumber data yang dibutuhkan dalam penelitian ini adalah sebagai berikut :

1. Data Primer merupakan suatu data yang diperoleh melalui wawancara yang dilakukan langsung dengan responden yang dapat mewakili beberapa sumber data. Didapatkan secara langsung dari informan yang memiliki data dan informasi mengenai penelitian ini oleh peneliti, yaitu oleh Satlantas Polres Parepare dan Dinas Perhubungan Kota Parepare melalui wawancara disertai dengan observasi (pengamatan). Dengan demikian dari

data dan informasi yang diperoleh adalah data yang validitasnya dapat dipertanggungjawabkan. Penulis menggunakan suatu istilah *social situation* atau situasi sosial sebagai objek peneitian yang terdiri dari tiga elemen, yaitu tempat (*place*), pelaku (*actors*) dan aktivitas (*activity*), yang berintraksi secara sinergi.

2. Data sekunder merupakan data yang dihasilkan oleh peneliti melalui sumber yang sudah tersedia. Data yang diperoleh melalui studi kepustakaan atau dari berbagai literatur dengan menelaah buku-buku dan tulisan-tulisan atau internet, jurnal hukum, serta peraturan perundang-undangan yang relevan dengan permasalahan yang diteliti dan tidak terlepas dari pengamatan suatu objek oleh peneliti.

E. Alat dan Bahan

Tahap desain dilakukan untuk menentukan ruang lingkup pembuatan aplikasi yang ditangani, yang selanjutnya dijabarkan dalam bentuk diagram alir. Dalam pembuatan program aplikasi tersebut penulis membutuhkan perangkat keras (*Hardware*) dan perangkat lunak (*Software*). Adapun *Hardware* dan *Software* yang digunakan adalah :

1. Perangkat keras (*hardware*)

Perangkat keras yang digunakan untuk membangun aplikasi menggunakan PC (*Personal Komputer*) atau laptop dengan spesifikasi minimal sebagai berikut :

- a. *Processor Core i5.*
- b. *Hardisk 500 GB.*
- c. *RAM 8 GB DDR 4.*
- d. *Smartphone Android versi 9 (Pie).*

2. Perangkat Lunak (*software*)

Perangkat lunak yang digunakan merancang dan membangun aplikasi ini dapat dilihat pada tabel 3.2 berikut :

Table 3.1 Spesifikasi Perangkat Lunak

Jenis	Spesifikasi
Sistem Operasi	<i>Windows 10 (64 bit)</i>
Bahasa Pemrograman	<i>Javascript</i>
Database	<i>MySQL</i>
<i>Tools</i>	<i>XAMPP, Visual Studio Code</i>
<i>Interface</i>	<i>Miscrosoft Visio</i>

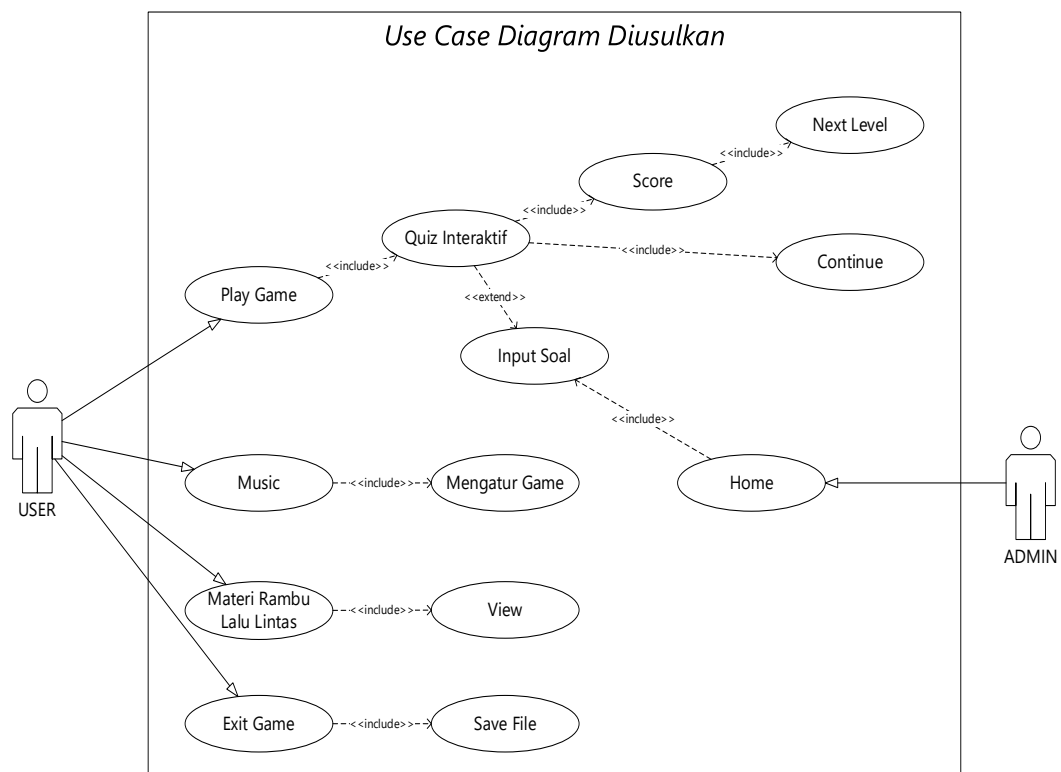
BAB IV

HASIL DAN PEMBAHASAN

A. Rancangan Sistem

1. Sistem Diusulkan

Sistem yang diusulkan ini menggunakan *Use Case Diagram*. *Use case diagram* menggambarkan interaksi antara *actor* dengan proses atau sistem yang dibuat. *Use case* dan *actor* menggambarkan ruang lingkup sistem yang sedang dibangun. *Use case* meliputi semua hal yang ada pada sistem, sedangkan *actor* meliputi semua hal yang ada diluar sistem. *Actor* termasuk seseorang atau apa saja yang berhubungan dengan sistem yang dibangun.

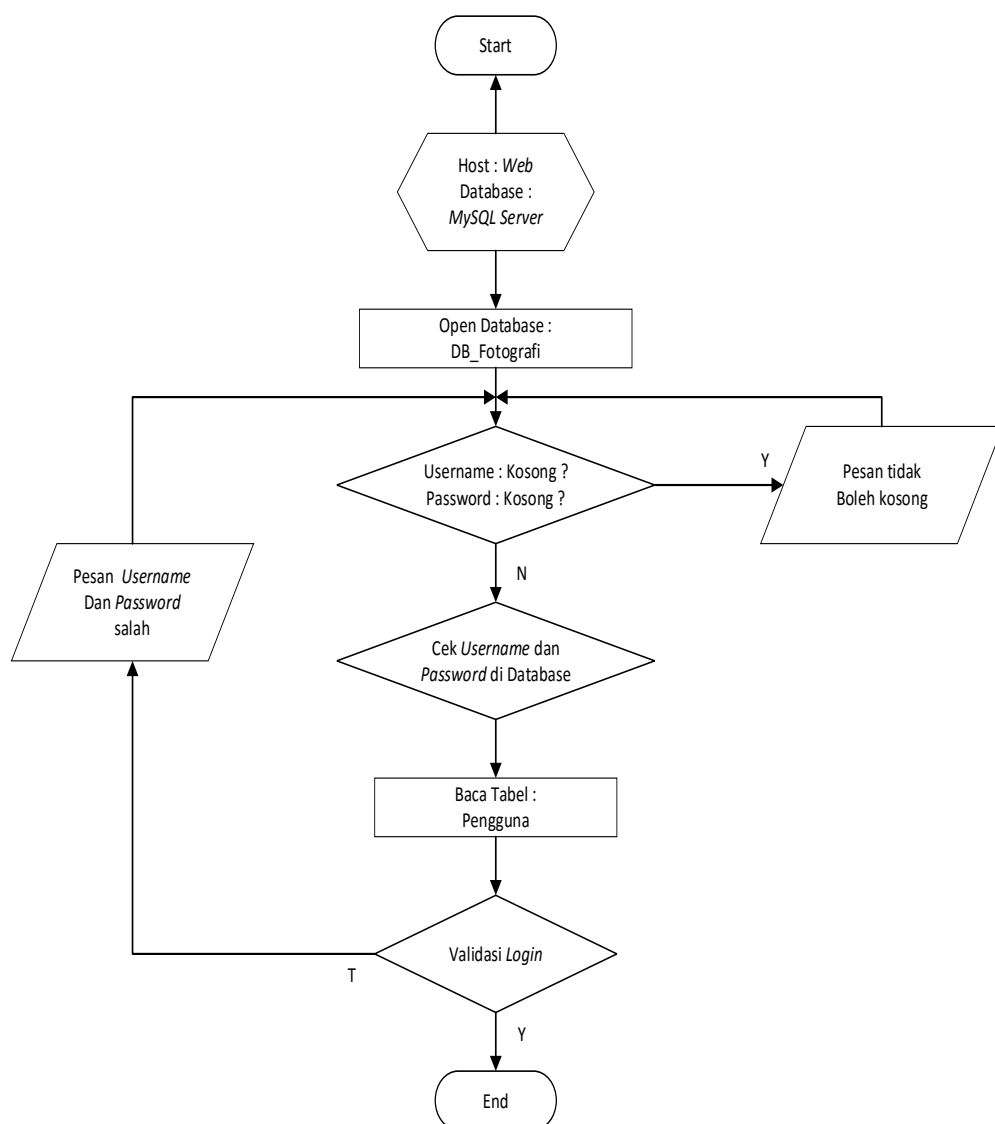


Gambar 4.1 *Use Case* Sistem Diusulkan

2. Flowchart

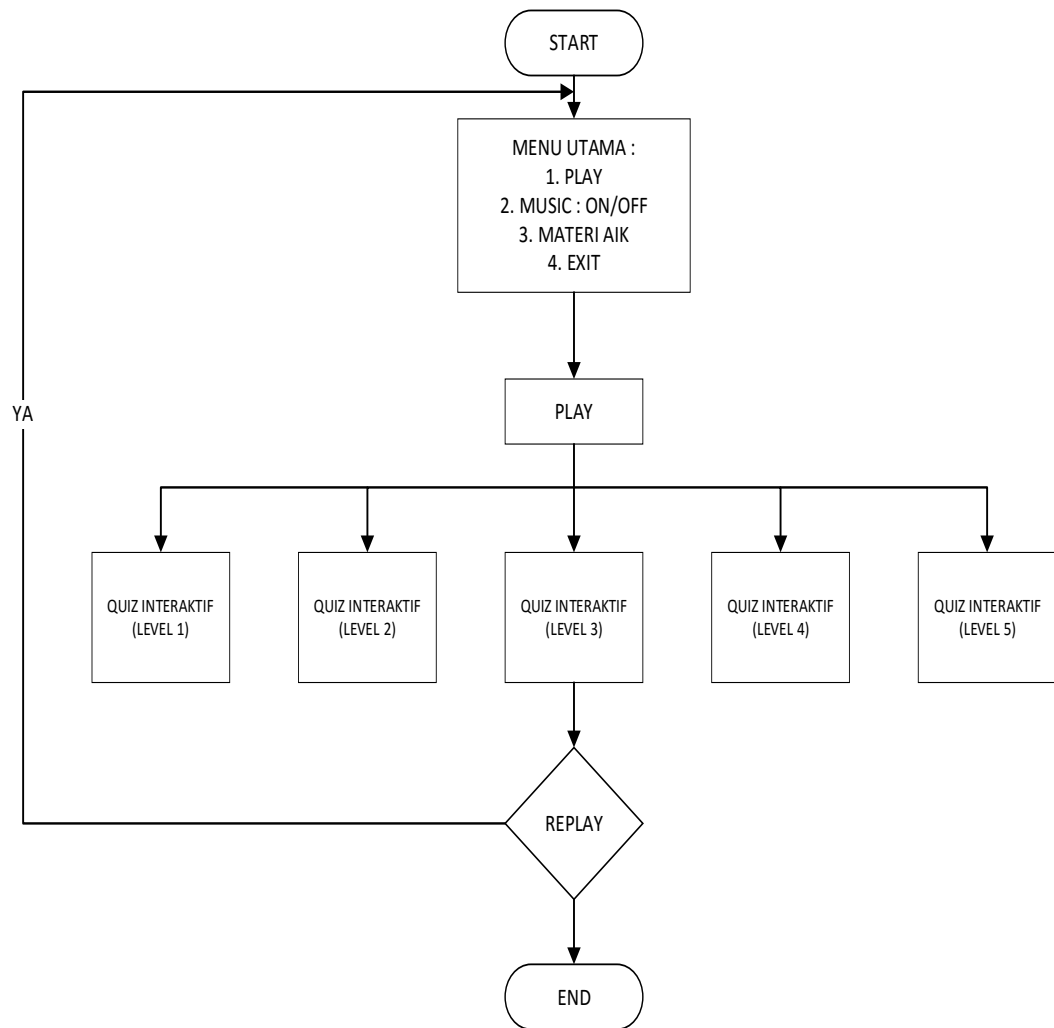
Flowchart merupakan bagan yang menunjukkan alur kerja atau apa yang sedang dikerjakan di dalam sistem secara keseluruhan dan menjelaskan urutan dari prosedur-prosedur yang ada di dalam alur penelitian. Berikut gambaran *flowchart login* dan *system* dapat dilihat pada gambar sebagai berikut.

a. Flowchart Login



Gambar 4.2 Flowchart Login

b. *Flowchart System*



Gambar 4.3 *Flowchart System*

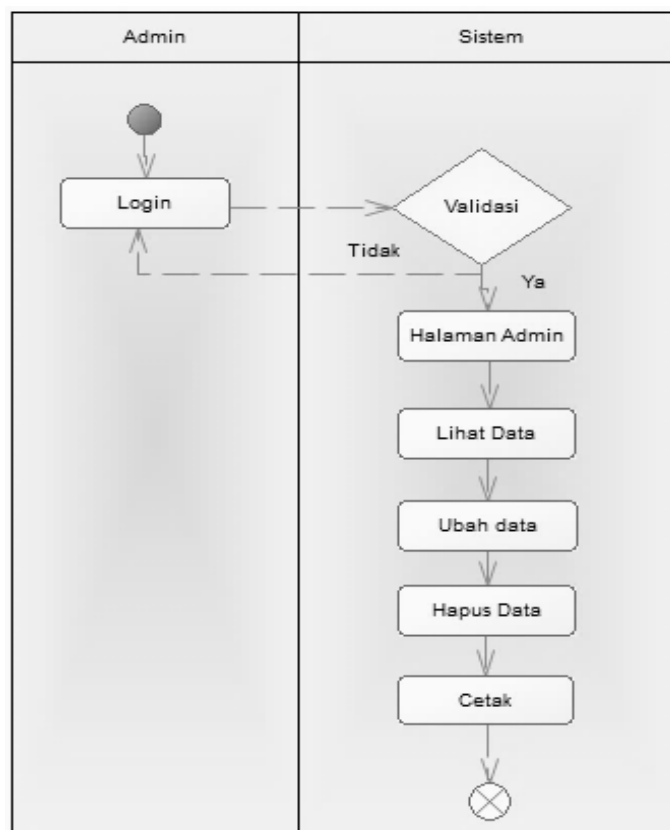
B. Analisis Aliran Data UML

Analisis aliran data yang diusulkan ini bertujuan mengetahui proses informasi, dalam analisis sistem penulis menggunakan pengembangan orientasi objek sehingga menggunakan *Use Case Diagram*, *Activity Diagram*, dan *Sequence diagram*.

1. Activity Diagram

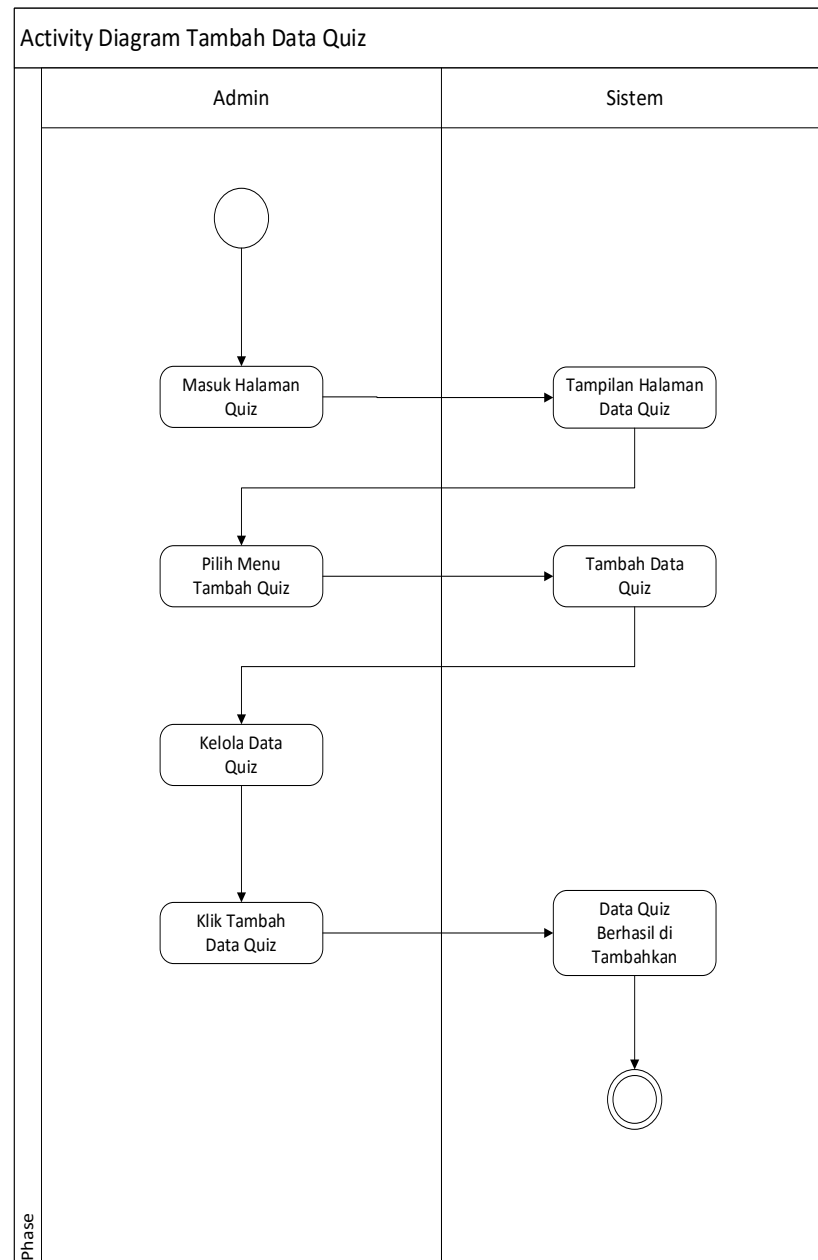
Activity diagram adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. *Activity diagram* ini menjelaskan tentang aktifitas-aktifitas yang terjadi dalam sebuah aliran proses pada sebuah *system*.

a. Activity Diagram Admin



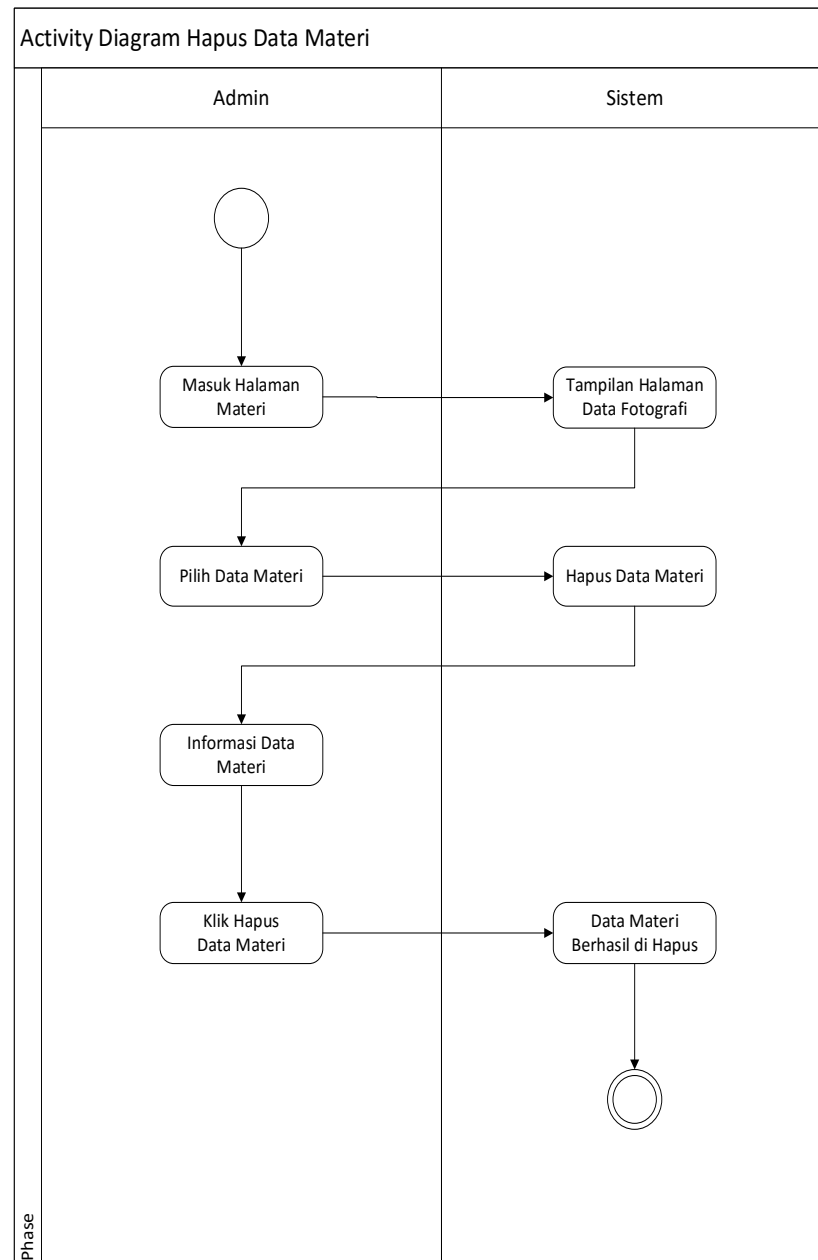
Gambar 4.4 Activity Diagram Admin

b. *Activity Diagram* Tambah Data Materi



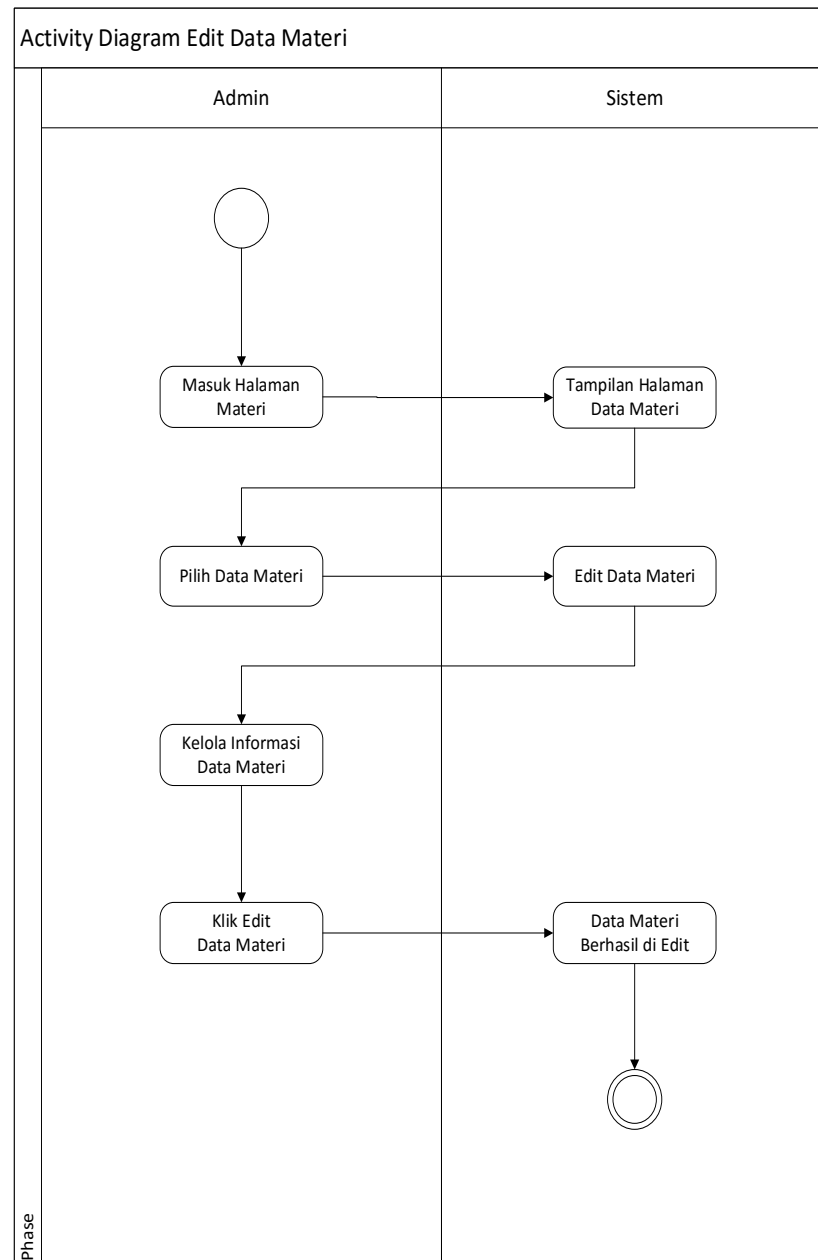
Gambar 4.5 *Activity Diagram* Tambah Data Materi

c. *Activity Diagram* Hapus Data Materi



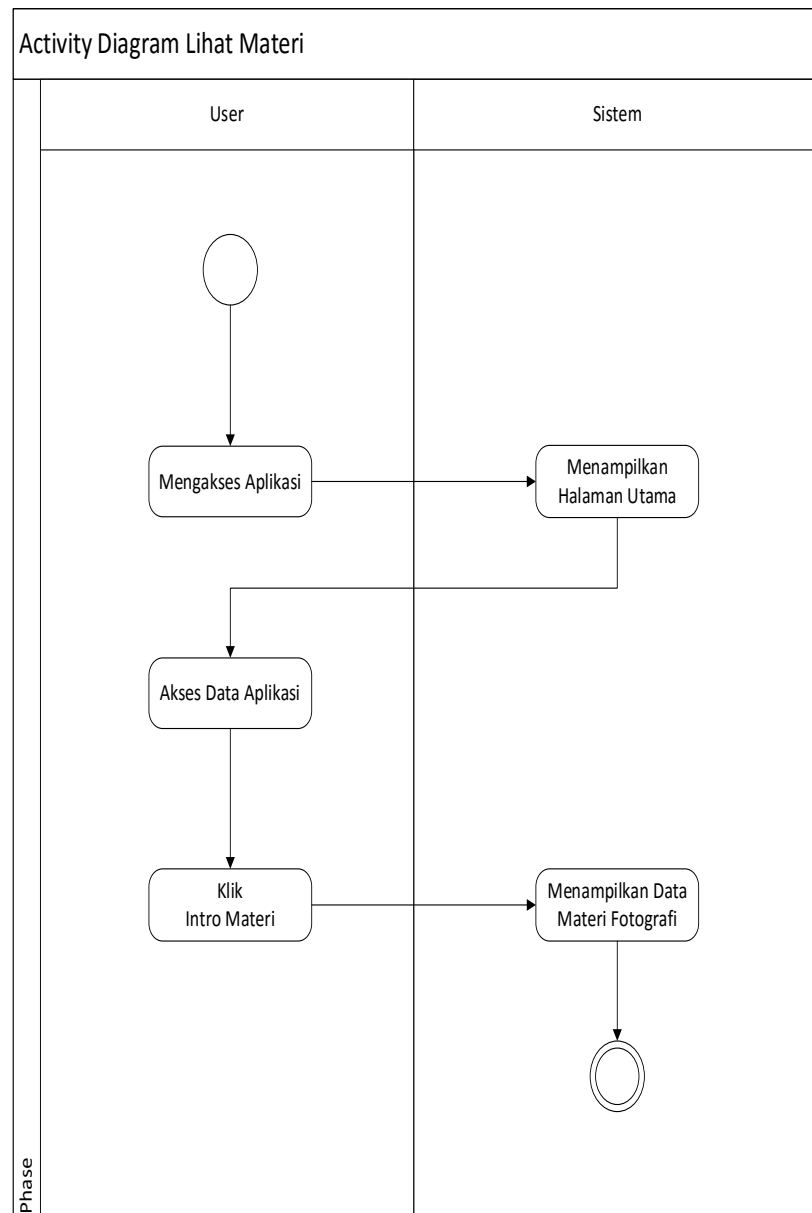
Gambar 4.6 *Activity Diagram* Hapus Data Materi

d. *Activity Diagram Edit Data Materi*



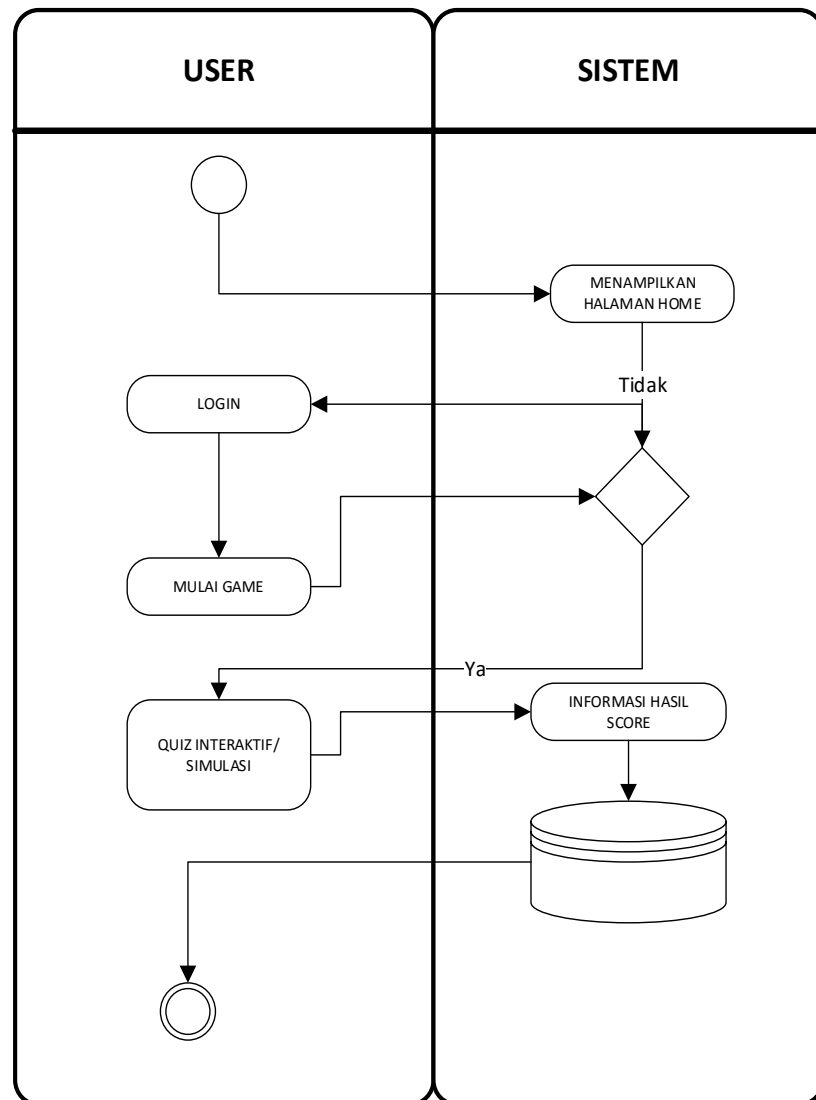
Gambar 4.7 *Activity Diagram Edit Data Materi*

e. *Activity Diagram* Lihat Materi



Gambar 4.8 *Activity Diagram* Lihat Materi

f. *Activity Diagram User*

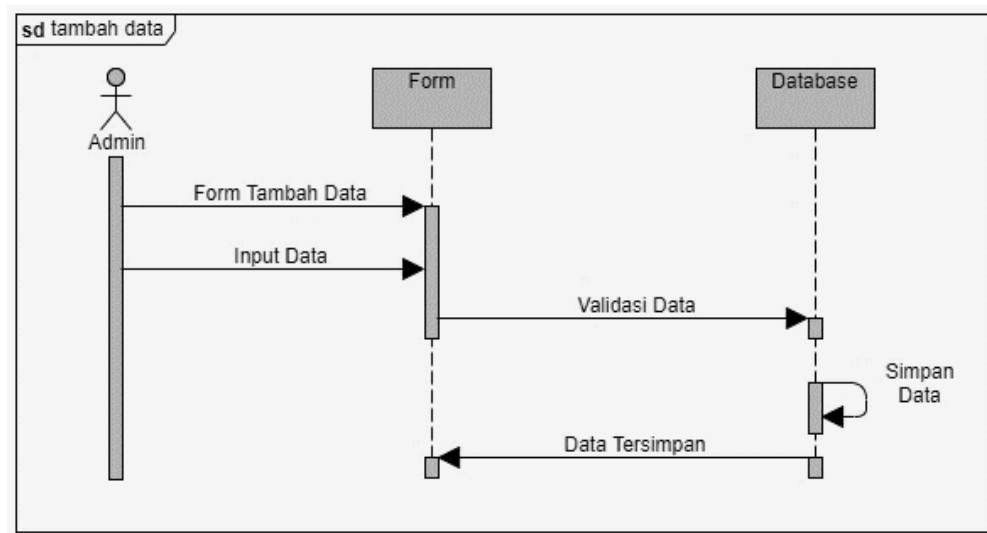


Gambar 4.9 *Activity Diagram User*

2. Sequence Diagram

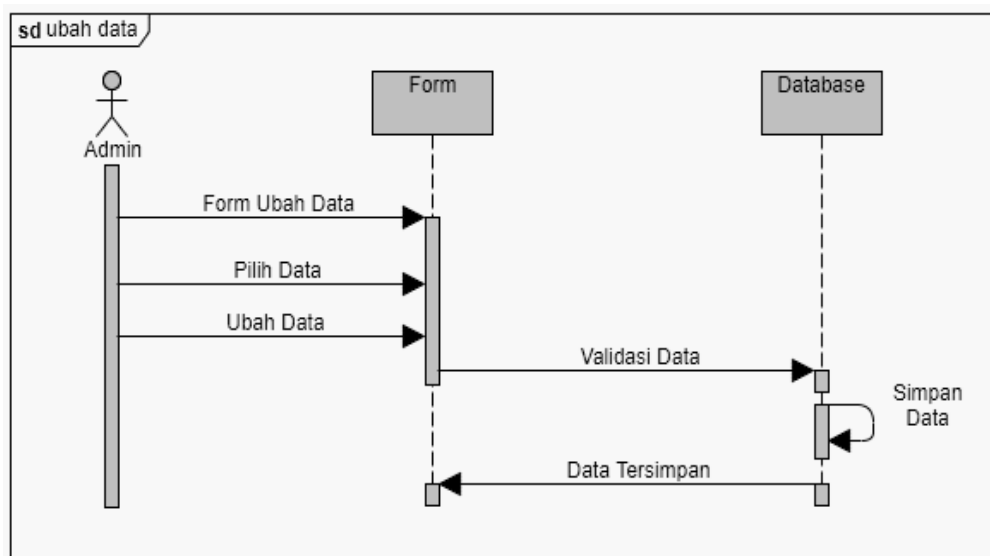
Sequence Diagram merupakan salah satu diagram *Interaction* yang menjelaskan bagaimana suatu operasi itu dilakukan *message* (pesan) apa yang dikirim dan kapan pelaksanaannya.

a. Sequence Diagram Tambah Data Admin



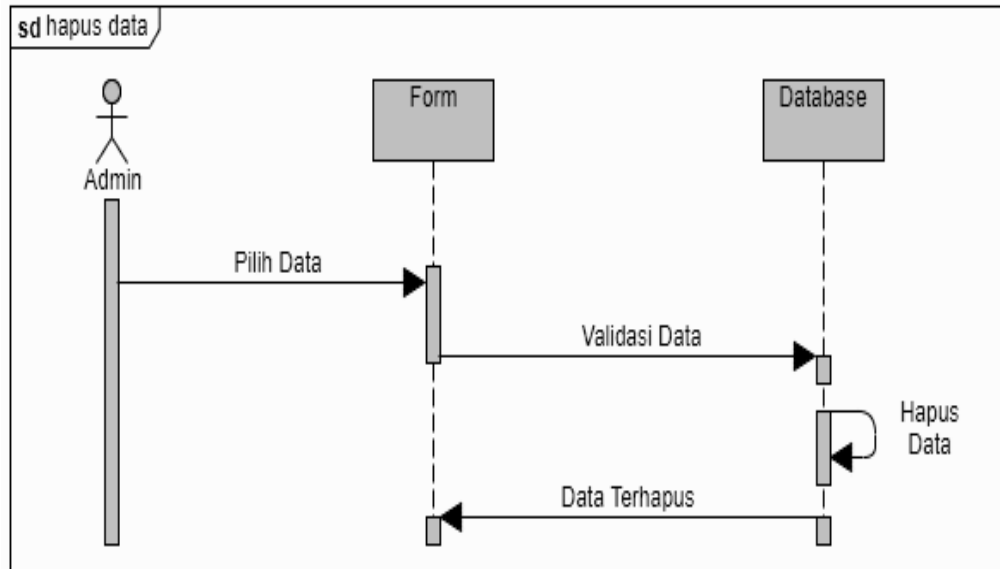
Gambar 4.10 Sequence Diagram Tambah Data Admin

b. Sequence Diagram Edit Data Admin



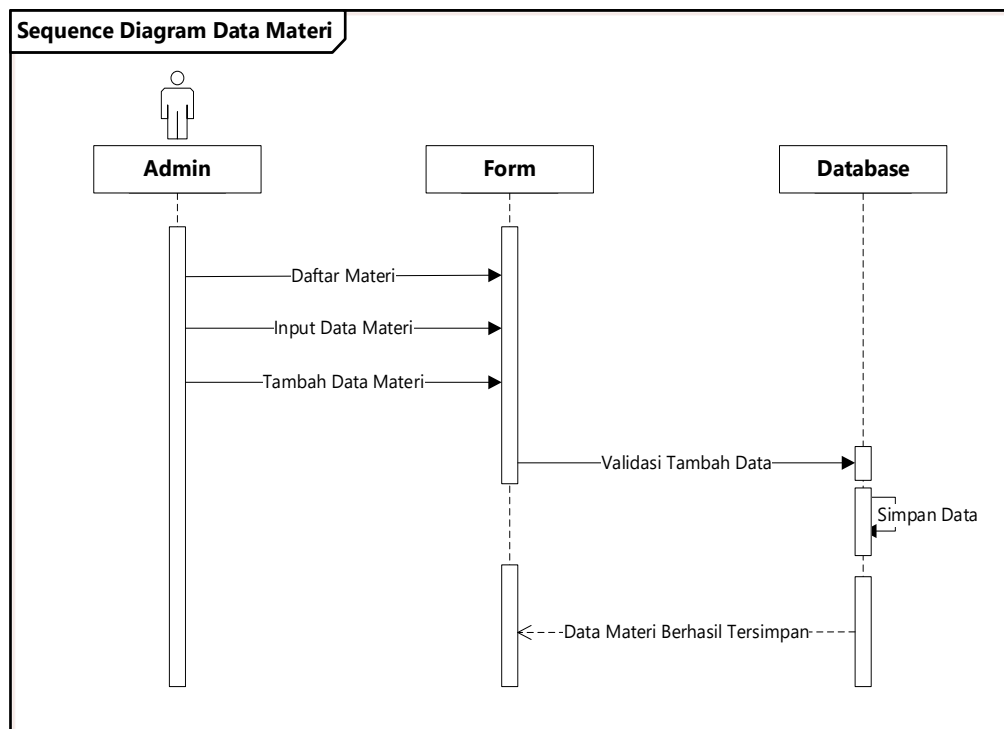
Gambar 4.11 *Sequence Diagram Edit Data Admin*

c. *Sequence Diagram Hapus Data Admin*



Gambar 4.12 *Sequence Diagram Hapus Data Admin*

d. *Sequence Diagram Data Materi*

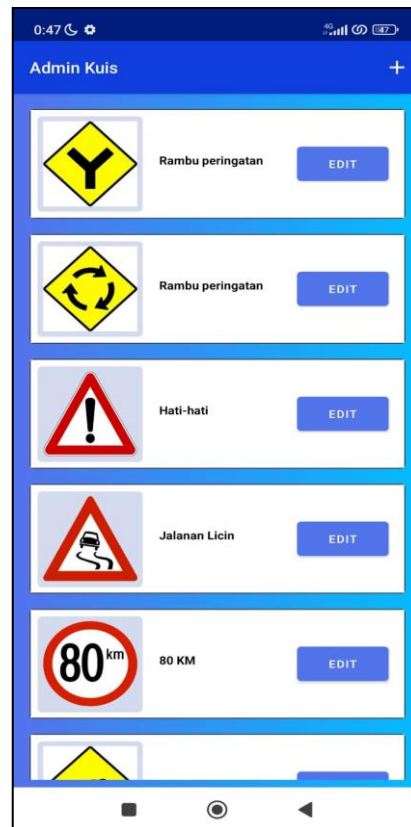


Gambar 4.13 *Sequence Diagram Data Mater*

C. Detail Aplikasi

1. Admin

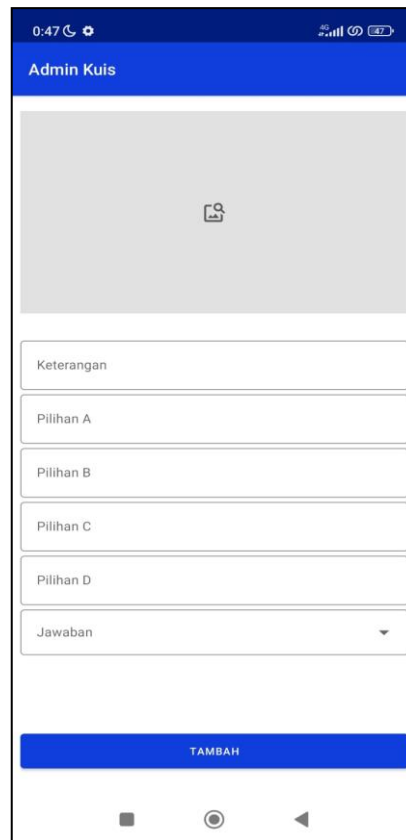
a. Halaman Utama Admin



Gambar 4.14 Halaman Utama Admin

Halaman utama merupakan halaman beranda *admin*, halaman ini digunakan *admin* untuk mengelola data-data materi rambu lalu lintas.

a. Halaman Tambah Data Materi

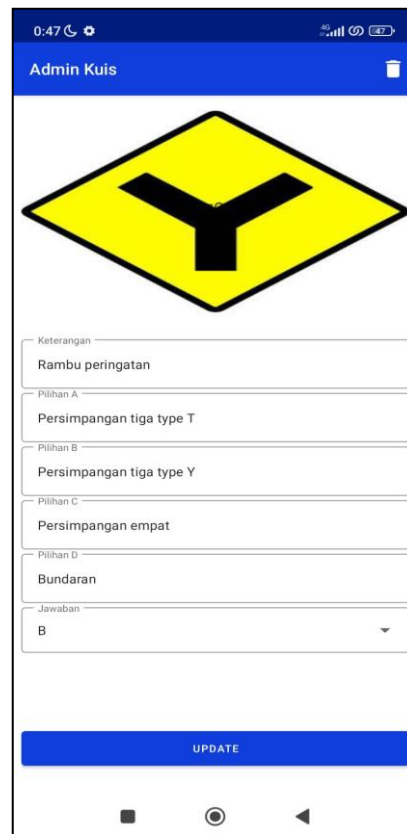


The screenshot displays the 'Admin Kuis' mobile application interface. At the top, there is a status bar showing the time as 0:47 and various system icons. Below the status bar is a blue header with the text 'Admin Kuis'. The main content area features a large grey rectangle with a question mark icon. Below this, there are several input fields: 'Keterangan', 'Pilihan A', 'Pilihan B', 'Pilihan C', 'Pilihan D', and 'Jawaban' (a dropdown menu). At the bottom of the form is a blue button labeled 'TAMBAH'. The bottom of the screen shows the standard Android navigation bar with back, home, and recent apps icons.

Gambar 4.15 Halaman Menu Tambah Data Materi

Pada gambar diatas ini merupakan tampilan menu tambah data materi yang hanya bisa diakses oleh *admin*. Halaman menu ini digunakan *admin* untuk memasukkan data-data materi rambu lalu lintas.

b. Halaman *Edit* Data Materi



0:47

Admin Kuis

Rambu peringatan

Persimpangan tiga type T

Persimpangan tiga type Y

Persimpangan empat

Bundaran

B

UPDATE

Gambar 4.16 Halaman Menu *Edit* Data Materi

Pada gambar diatas ini merupakan tampilan menu *edit* data materi yang hanya bisa diakses oleh *admin*. Halaman menu ini digunakan *admin* untuk mengubah data-data materi rambu lalu lintas.

c. Halaman Hapus Data Materi



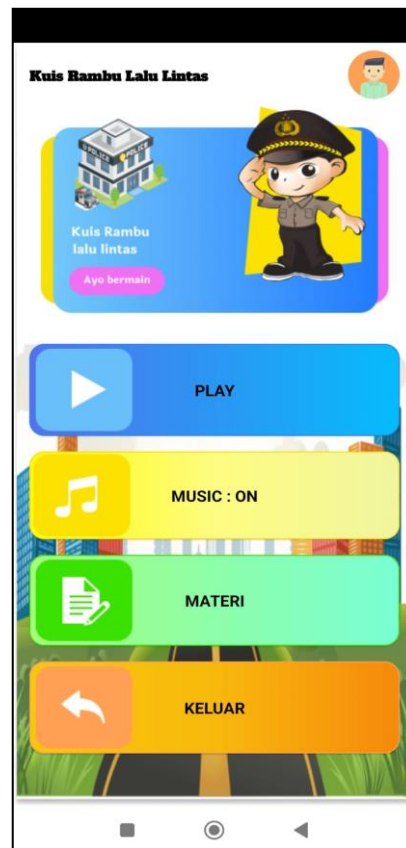
The screenshot shows a mobile application interface titled "Admin Kuis". At the top, there is a status bar with the time "0:47", signal strength, and battery level. Below the title bar, there is a large yellow diamond-shaped traffic sign with a black Y-junction symbol. Below the sign, there is a form with several input fields: "Keterangan" (containing "Rambu peringatan"), "Pilihan A" (containing "Persimpangan tiga type T"), "Pilihan B" (containing "Persimpangan tiga type Y"), "Pilihan C" (containing "Persimpangan empat"), and "Pilihan D" (containing "Bundaran"). Below these fields is a "Jawaban" (Answer) dropdown menu currently set to "B". At the bottom of the form is a blue button labeled "UPDATE". The bottom of the screen shows the standard Android navigation bar with back, home, and recent apps buttons.

Gambar 4.17 Halaman Menu Hapus Data Materi

Pada gambar diatas ini merupakan tampilan menu hapus data materi yang hanya bisa diakses oleh *admin*. Halaman menu ini digunakan *admin* untuk menghapus data-data materi rambu lalu lintas.

1. *User*

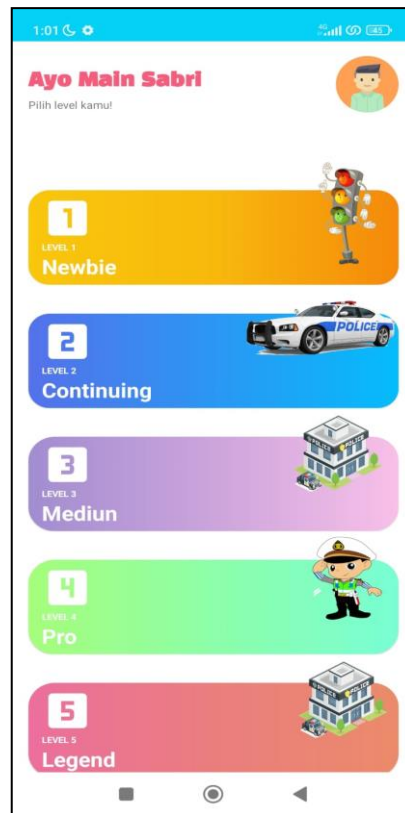
a. Halaman *Home User*



Gambar 4.18 Halaman *Home User*

Halaman *home* merupakan halaman beranda *user*, halaman ini digunakan *user* untuk memulai materi pembelajaran rambu-rambu lalu lintas. Pada halaman ini terdapat menu *play*, *music*, materi dan keluar.

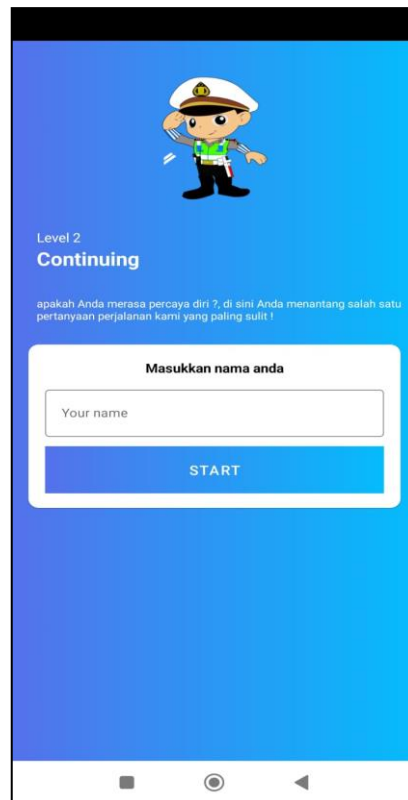
b. Halaman Pilih Level *User*



Gambar 4.19 Halaman Menu Pilih Level *User*

Halaman pilih user merupakan halaman level permainan *user*, halaman ini digunakan *user* untuk memulai pembelajaran rambu-rambu lalu lintas. Halaman menu pilih level permainan memiliki daftar level yang tersedia, termasuk ikon atau nomor yang mewakili tingkat kesulitan atau urutan level dalam permainan. Pemain dapat memilih level yang diinginkan dengan mengklik tombol yang sesuai.

c. Halaman *Form* Masukkan *User*



apakah Anda merasa percaya diri ?, di sini Anda menantang salah satu pertanyaan perjalanan kami yang paling sulit !

Level 2
Continuing

Masukkan nama anda

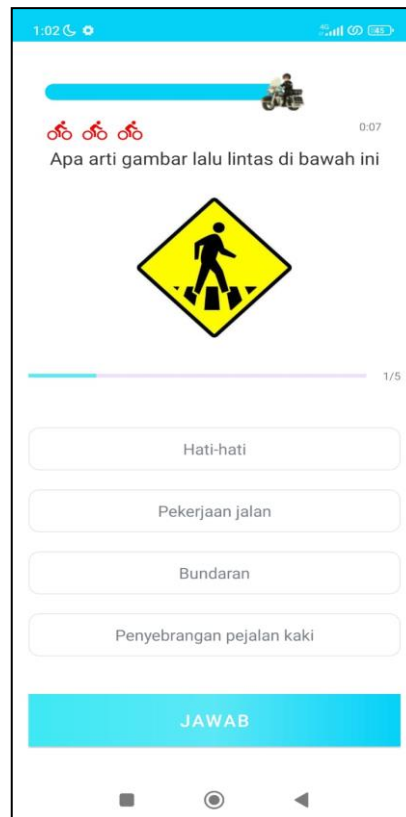
Your name

START

Gambar 4.21 Halaman Menu Masukkan *User*

Halaman masukkan materi merupakan halaman menu yang berada pada halaman utama *User*, halaman ini terdapat pada awal sistem yang digunakan *user* sebagai langkah awal memulai permainan *Quiz*.

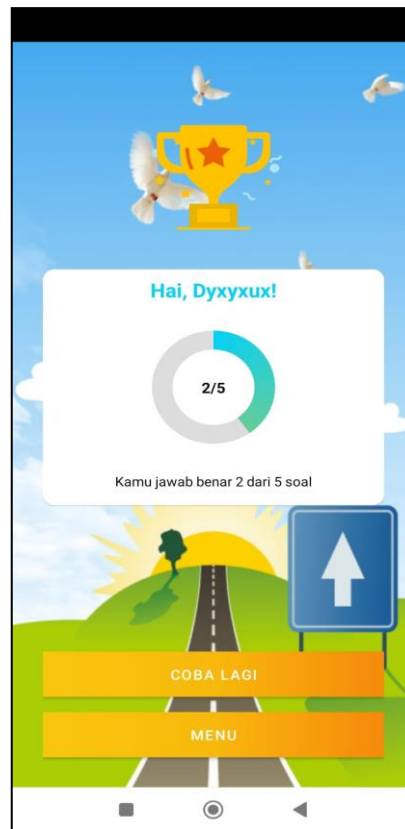
d. Halaman *Form* Evaluasi (*Quiz Interaktif*)



Gambar 4.22 Halaman *Form* Evaluasi (*Quiz Interaktif*)

Halaman *form* evaluasi merupakan halaman menu yang berada pada halaman utama *User*, halaman ini digunakan *user* untuk memulai permainan game *quiz* interaktif dalam pembelajaran rambu lalu lintas. Pada halaman ini berfungsi sebagai bahan evaluasi dan simulasi anak-anak dalam materi pembelajaran rambu lalu lintas.

e. Halaman Menu *Score*



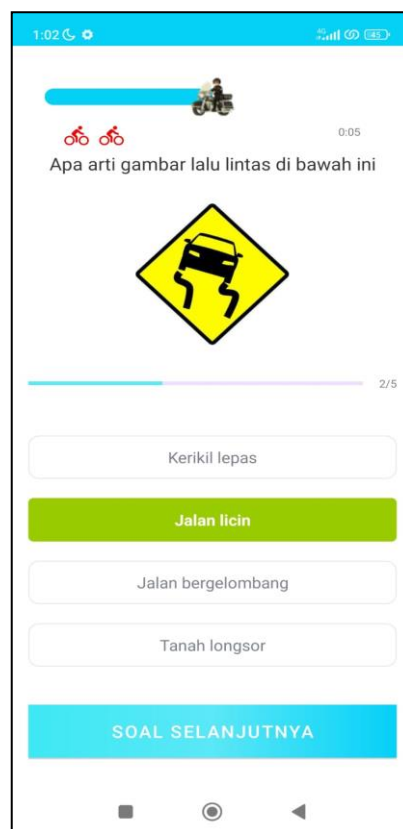
Gambar 4.23 Halaman Menu *Score*

Halaman menu *popup score* terdapat dalam halaman evaluasi/ *quiz*, halaman ini digunakan *user* untuk menampilkan hasil *score* permainan setelah melakukan permainan *game*. Pada menu ini terdapat fitur kembali ke halaman menu dan *continue game*.

D. Rancangan *Output* Program

Perancangan *output* program merupakan perancangan yang dilakukan untuk mengetahui informasi dan data-data pengguna berupa Aplikasi *Education Game* Dalam Pembelajaran Rambu Lalu Lintas Pada Anak SD Berbasis *Mobile Learning*.

1. Tampilan Halaman Jawaban Benar



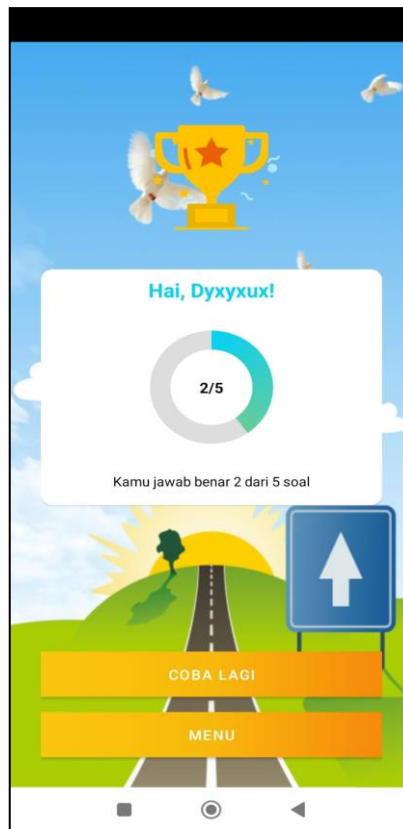
Gambar 4.24 Halaman Jawaban Benar

2. Tampilan Halaman Jawaban Salah



Gambar 4.24 Halaman Jawaban Salah

3. Tampilan Halaman *Popup Score*



Gambar 4.24 Halaman *Popup Score*

E. Implementasi

Adapun kebutuhan yang digunakan untuk Membangun sebuah sistem dengan judul “*E-Commerce Vape Store Sebagai Media Pemasaran Aksesoris dan Liquid Berbasis Mobile Smartphone*”.

1. Kebutuhan Perangkat Keras (*Hardware*)

Spesifikasi *hardware* pengguna aplikasi ini adalah sebagai berikut :

Tabel 4.1 Spesifikasi *Hardware* Komputer

Jenis	Spesifikasi
<i>Notebook/Komputer</i>	<i>Acer</i>
<i>Processor</i>	<i>Prosesor Intel(R) Core (TM) i5-3210M</i>
<i>Ram</i>	8 GB DDR 4
<i>Hardisk</i>	500 GB

Tabel 4.2 Spesifikasi *Smartphone Admin*

Jenis	Spesifikasi
<i>Jenis Smartphone</i>	Xiaomi Redmi 11
Layar	5,2 inch
Ram	4 gb

Tabel 4.3 Spesifikasi *Smartphone User*

Jenis	Spesifikasi
<i>Jenis Smartphone</i>	Vivo V25e
Layar	6 inch
Ram	4 gb

2. Kebutuhan Perangkat Lunak (*Software*)

Spesifikasi *software* pengguna aplikasi ini sebagai berikut :

Tabel 4.4 Spesifikasi *Software Komputer*

Jenis	Spesifikasi
Sistem Operasi	<i>Windows 10 (64 bit)</i>

Tabel 4.5 Spesifikasi *Software Smartphone Admin*

Jenis	Spesifikasi
Versi <i>Android</i>	<i>7.1 Nougat</i>
Versi perangkat lunak	PD1705F_EX_A_1.18.0

Tabel 4.6 Spesifikasi *Software User*

Jenis	Spesifikasi
Versi <i>Android</i>	<i>Android 9 Pie</i>
Versi <i>MIUI</i>	<i>MIUI 10 Global 9.6.27 Beta</i>

F. Metode pengujian

Pengujian sistem merupakan proses eksekusi suatu program atau sistem secara keseluruhan. Pengujian menunjukkan bahwa fungsi dari perangkat lunak bekerja sesuai dengan spesifikasi dan bahwa persyaratan kinerja telah dipenuhi.

Metode pengujian yang digunakan ada *black-box*. Pengujian *blackbox* perangkat lunak dilakukan untuk memperlihatkan bahwa masing-masing fungsi telah beroperasi sepenuhnya dan secara bersamaan pula mencari kesalahan pada setiap fungsi.

G. Teknik Pengujian

Pengujian *black-box* berkaitan dengan pengujian yang dilakukan pada interface perangkat lunak. Meskipun didesain untuk mengungkapkan kesalahan, pengujian *black-box* digunakan untuk memperlihatkan bahwa fungsi-fungsi perangkat lunak adalah operasional, bahwa *input* diterima dengan baik dan *output* dihasilkan dengan tepat, dan integritas informasi *external* (seperti file data) dipelihara. Pengujian *black-box* menguji beberapa aspek dasar suatu sistem dengan sedikit memperhatikan struktur logika *internal* perangkat lunak tersebut.

Pengujian sistem dilakukan untuk memeriksa kekompakan antara komponen sistem yang diimplementasikan. Tujuan utama dari pengujian sistem ini adalah untuk memastikan bahwa elemen-elemen atau komponen-komponen dari sistem telah berfungsi sesuai dengan yang diharapkan. Pengujian perlu dilakukan untuk mencari kesalahan-kesalahan atau kelemahan-kelemahan yang

mungkin terjadi. Pengetesan sistem merupakan pengetesan program secara keseluruhan, adapun tahap yang di lakukan sebagai berikut :

1. Mekanisme pengujian :

- a. Pengetesan program aplikasi dengan cara menjalankan aplikasi.
- b. Pengetesan pemasukan data, perubahan data, dan penghapusan data.
- c. Pengetesan terhadap fungsi tombol yang terdapat pada masing-masing halaman apakah berfungsi dengan baik.

2. Hasil yang diperoleh :

Aplikasi *Education Game* Dalam Pembelajaran Interaktif Pengenalan Rambu Lalu Lintas Pada Anak SD Berbasis *Mobile Learning* sesuai usulan dapat berjalan dengan lancar sesuai dengan yang diharapkan.

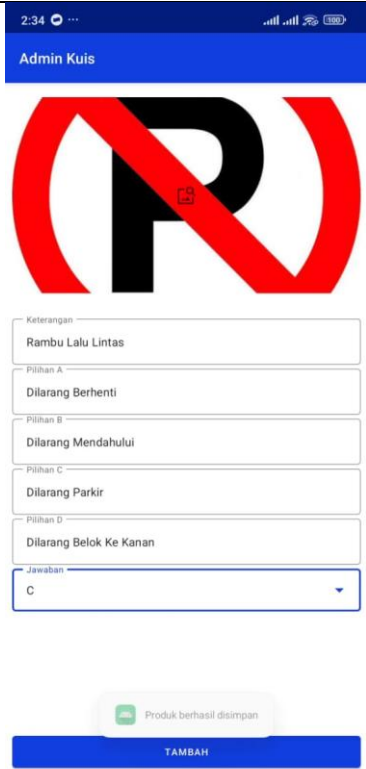
- a. Peningkatan pemahaman anak – anak dapat memperoleh pemahaman yang lebih baik tentang rambu lalu lintas dan artinya dalam keselamatan berkendara.
- b. Motivasi yang tinggi dengan pendekatan yang disajikan melalui game cenderung lebih menarik bagi anak – anak, meningkatkan motivasi mereka untuk belajar.
- c. Pengalaman pembelajaran yang menyenangkan akan menjadi lebih menarik dan interaktif, mengurangi kemungkinan anak – anak merasa bosan atau lelah.
- d. Peningkatan kesadaran anak – anak dengan mengetahui akan pentingnya keselamatan di jalan dan lebih mampu mengidentifikasi situasi berbahaya.

3. Pengujian *Black Box*

Pengujian merupakan suatu keharusan dalam membuat aplikasi untuk mendapatkan informasi mengenai kualitas dari aplikasi yang telah dibuat dan mengetahui apakah fungsi dari aplikasi tersebut telah berjalan sesuai tujuan. Berdasarkan rencana pengujian, maka dapat dilakukan pengujian sebagai berikut.

a. Hasil Pengujian Input Data Materi

Tabel 4.7 Hasil Pengujian Input Data Materi

Teks Faktor	Hasil	Keterangan
Menginput/menambahkan data materi	✓	Berhasil menambahkan data materi
Screenshot		
		

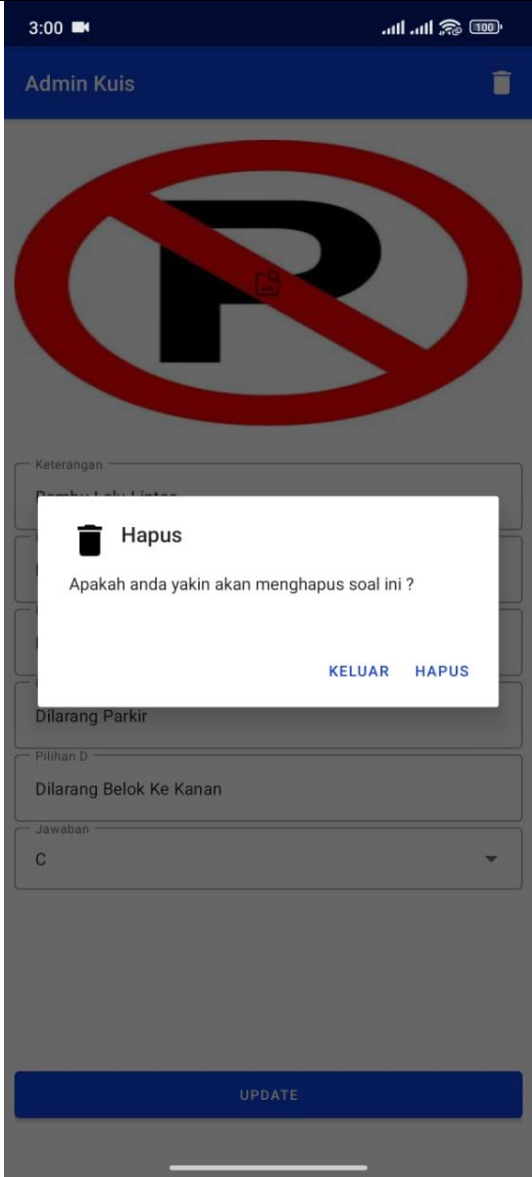
b. Hasil Pengujian Edit Data Materi

Tabel 4.8 Hasil Pengujian Edit Data Materi

Teks Faktor	Hasil	Keterangan
Mengedit/mengubah data materi	✓	Berhasil mengedit/mengubah data materi
Screenshot		
 The screenshot shows the 'Admin Kuis' interface. At the top, there's a blue header with the title 'Admin Kuis' and a trash icon. Below the header is a large image of a red triangular warning sign with a black exclamation mark. Underneath the image is a form with several input fields: 'Keterangan' (containing 'Hati-hati'), 'Pilihan A' (containing 'Dilarang Belok kiri'), 'Pilihan B' (containing 'Dilarang Putar Balik'), 'Pilihan C' (containing 'Dilarang berhenti'), 'Pilihan D' (containing 'Hati-hati'), and 'Jawaban' (a dropdown menu set to 'D'). At the bottom of the form is a blue button labeled 'UPDATE'.		

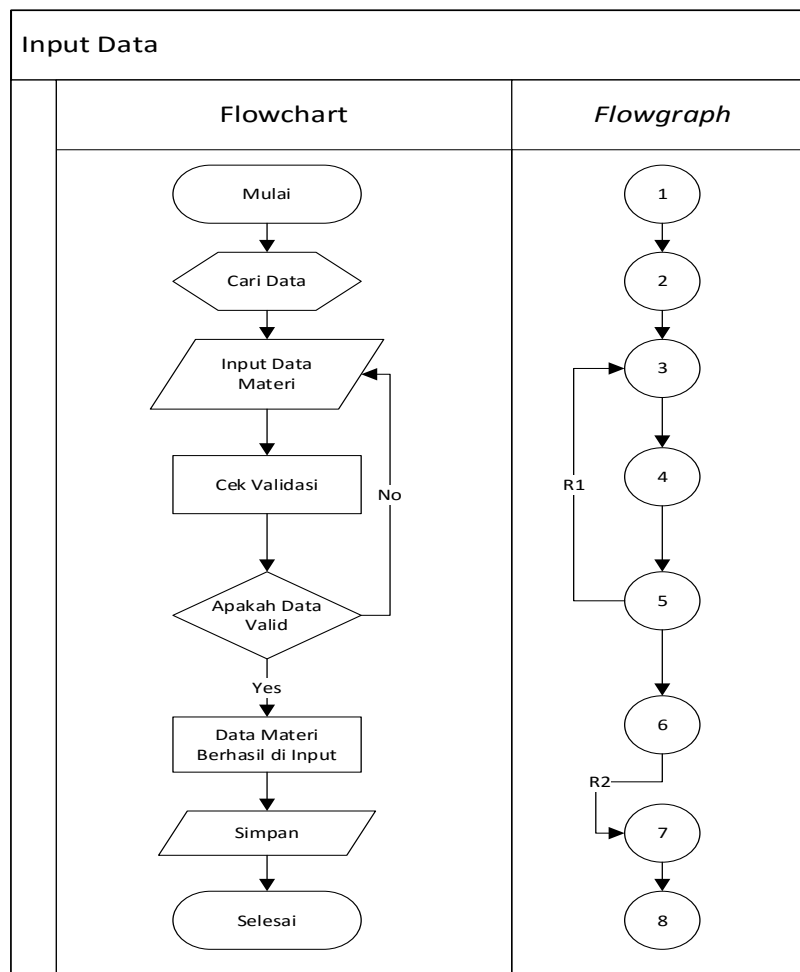
c. Hasil Pengujian Hapus Data Materi

Tabel 4.9 Hasil Pengujian Hapus Data Materi

Teks Faktor	Hasil	Keterangan
Menghapus data materi	✓	Berhasil menghapus data materi
Screenshot		
		

4. Pengujian *White Box*

a. *Input Data*

Tabel 4.10 Pengujian *Input Data Materi*

Dari gambar *flowgraph* *Input Data Materi* diatas dapat dilakukan proses perhitungan sebagai berikut :

- 1) Menghitung *Cyclomatic Complexity* $V(G)$ dari *Egde* dan *Node*:

Dengan Rumus : $V(G) = E - N + 2$

Dengan E (*edge*) = 8

N (*node*) = 8

Predikat *Node* (P) = 1

Penyelesaian : $V(G) = E - N + 2$

$$= 8 - 8 + 2 = 2$$

$$\begin{aligned}\text{Predikat (P)} &= P + 1 \\ &= 1 + 1 = 2\end{aligned}$$

2) Berdasarkan perhitungan *Cyclomatic Complexity* dari *Flowgraph* diatas memiliki *Region* = 2

3) Independent *Path* pada *flowgraph* diatas adalah :

$$\text{Path 1} = 1 - 2 - 3 - 4 - 2 - 3 - 4 - 5 - 6 - 7 - 8$$

$$\text{Path 2} = 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8$$

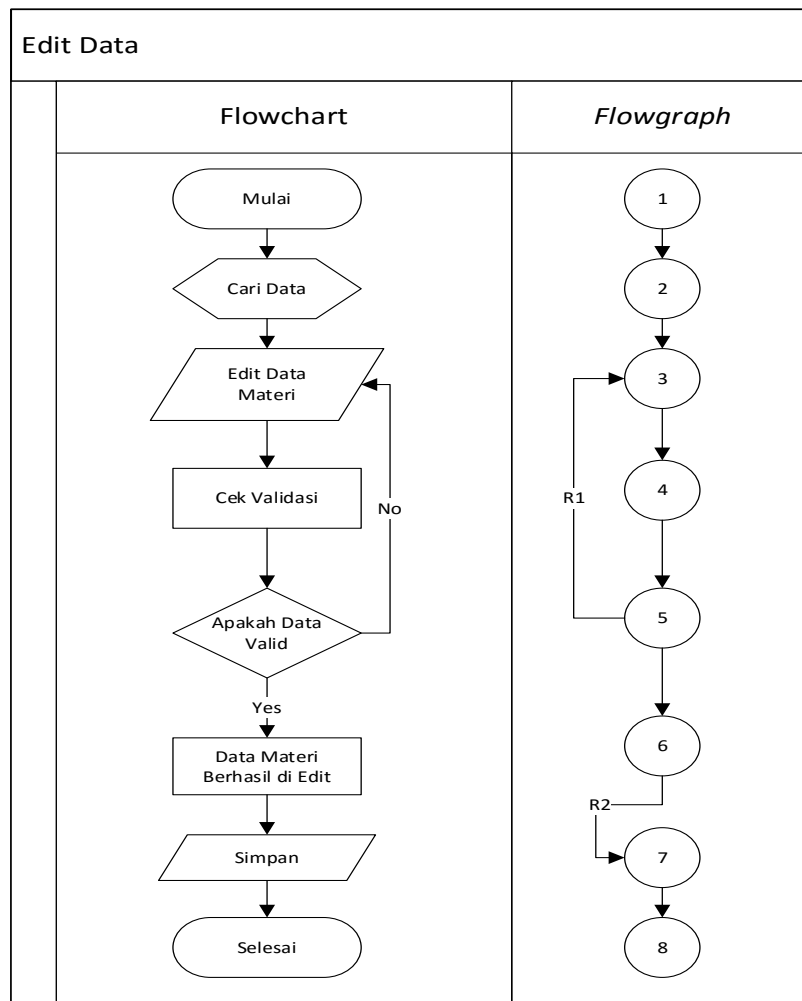
4) Grafik *Matriks*

Tabel 4.11 Grafik *Matriks* pada *Flowgraph Input Data Materi*

	1	2	3	4	5	6	7	$E - 1 = 0$
1		1						$1 - 1 = 0$
2			1					$1 - 1 = 0$
3				1				$1 - 1 = 0$
4		1			1			$2 - 1 = 1$
5						1		$1 - 1 = 0$
6							1	$1 - 1 = 0$
7								0
Sum (E) + 1								$1 + 1 = 2$

b. *Edit Data*

Tabel 4.12 Pengujian *Edit Data Materi*



Dari gambar *flowgraph* Edit Data Materi diatas dapat dilakukan proses perhitungan sebagai berikut:

- 1) Menghitung *Cyclomatic Complexity* $V(G)$ dari Egde dan Node:

Dengan Rumus : $V(G) = E - N + 2$

Dengan E (edge) = 8

N (node) = 8

Predikat Node (P) = 1

Penyelesaian : $V(G) = E - N + 2$

$$= 8 - 8 + 2 = 2$$

$$\begin{aligned}\text{Predikat (P)} &= P + 1 \\ &= 1 + 1 = 2\end{aligned}$$

2) Berdasarkan perhitungan *Cyclomatic Complexity* dari *Flowgraph* diatas memiliki *Region* = 2

3) Independent *Path* pada *flowgraph* diatas adalah :

$$\text{Path 1} = 1 - 2 - 3 - 4 - 2 - 3 - 4 - 5 - 6 - 7 - 8$$

$$\text{Path 2} = 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8$$

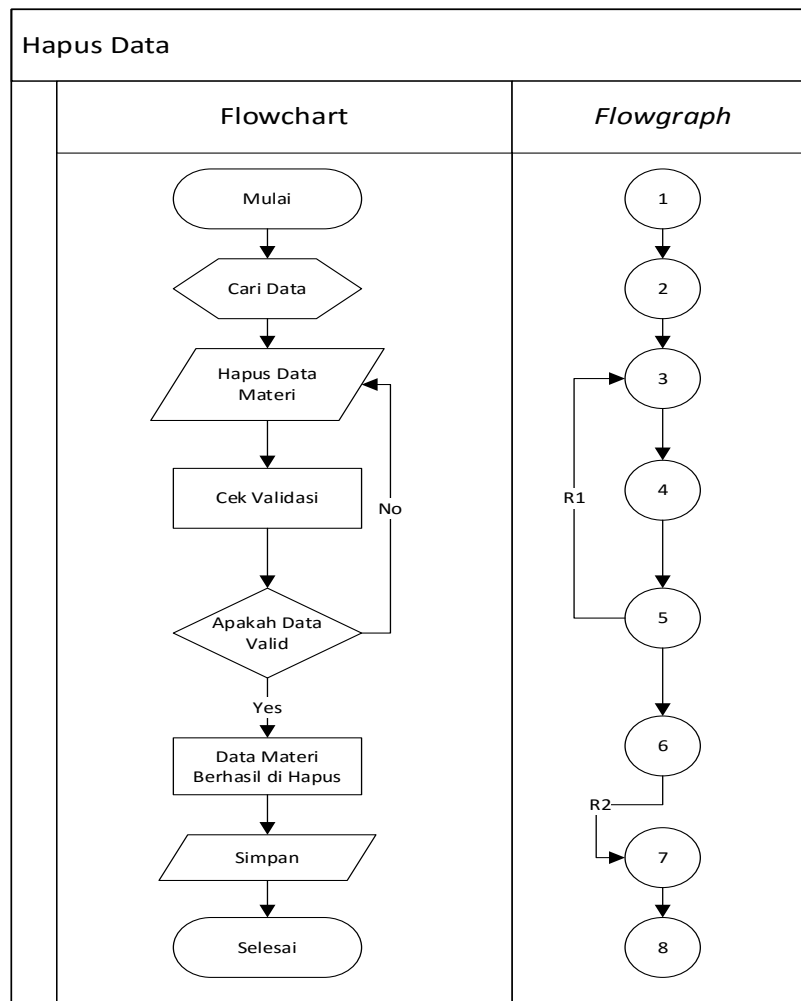
4) Grafik *Matriks*

Tabel 4.13 Grafik *Matriks* pada *Flowgraph Edit Data Materi*

	1	2	3	4	5	6	7	8	$E - 1 = 0$
1		1							$1 - 1 = 0$
2			1						$1 - 1 = 0$
3				1					$1 - 1 = 0$
4					1				$1 - 1 = 0$
5			1			1			$2 - 1 = 1$
6							1		$1 - 1 = 0$
7								1	$1 - 1 = 0$
8									0
Sum (E) + 1									$1 + 1 = 2$

c. Hapus Data

Tabel 4.14 Pengujian Hapus Data Materi



Dari gambar *flowgraph* Hapus Data Materi diatas dapat dilakukan proses perhitungan sebagai berikut :

- 1) Menghitung *Cyclomatic Complexity* $V(G)$ dari Egde dan *Node*:

Dengan Rumus : $V(G) = E - N + 2$

Dengan E (edge) = 8

N (node) = 8

Predikat *Node* (P) = 1

Penyelesaian : $V(G) = E - N + 2$

$$= 8 - 8 + 2 = 2$$

$$= 1 + 1 = 2$$

diatas memiliki $Region = 2$

Independent *Path* pada *flowgraph* diatas adalah :

Path 1 = 1 – 2 – 3 – 4 – 2 – 3 – 4 – 5 – 6 – 7 – 8

$$\text{Path 2} = 1-2-3-4-5-6-7-8$$

Grafik Matriks

Tabel 4.15 Grafik *Matriks* pada *Flowgraph* Hapus Data Materi

[illegible]

BAB V

PENUTUP

A. Kesimpulan

Berdasarkan hasil penelitian dan pembahasan yang telah diuraikan pada bab-bab sebelumnya, Dihasilkannya sebuah Aplikasi *Education Game* Dalam Pembelajaran Interaktif Pengenalan Rambu Lalu Lintas Pada Anak SD Berbasis *Mobile Learning*. Berdasarkan implementasi dan hasil pengujian dengan dua teknik pengujian yaitu pengujian *Black-Box* dan pengujian *White-Box* dapat disimpulkan bahwa aplikasi ini berjalan baik sesuai dengan fungsinya. Penulis mendapatkan data-data yang kemudian dianalisis dan diuraikan, maka penulis mengambil kesimpulan sebagai berikut :

1. Dalam menganalisis, merancang, dan membangun aplikasi education game untuk pembelajaran interaktif pengenalan rambu lalu lintas pada anak SD berbasis *mobile learning*, langkah pertama adalah melakukan analisis kebutuhan pengguna dengan fokus pada suatu fitur-fitur yang mendukung pengenalan rambu lalu lintas secara menyenangkan. Desain aplikasi harus melibatkan antarmuka yang berwarna dan mudah dipahami (*user friendly*), menggunakan *HTML*, *CSS*, dan *JavaScript* untuk *front-end* serta *PHP* dan *MySQL* untuk *back-end*. Selanjutnya, struktur database harus dirancang untuk menyimpan data mengenai rambu lalu lintas serta progres pengguna.

2. Aplikasi *education game* ini dengan menerapkan suatu konsep gamifikasi, penting untuk memasukkan elemen permainan seperti level, tantangan, dan sistem penghargaan guna meningkatkan motivasi dan pemahaman anak-anak SD. Aplikasi harus menyediakan berbagai tantangan yang relevan dengan materi pembelajaran serta sistem poin dan lencana untuk memberikan umpan balik positif dan mendorong kemajuan. Penggunaan elemen interaktif dan animasi dapat membuat pembelajaran lebih menarik, sementara pengujian serta umpan balik dari pengguna memastikan suatu aplikasi efektif dan menyenangkan. Dengan pendekatan ini, aplikasi tidak hanya mengajarkan tentang rambu lalu lintas tetapi juga menjadikan proses pembelajaran lebih motivatif dan menyenangkan.

B. Saran-saran

Aplikasi *Education Game* Dalam Pembelajaran Interaktif Pengenalan Rambu

Lalu Lintas Pada Anak SD Berbasis *Mobile Learning* masih terdapat beberapa kekurangan. Berdasarkan kesimpulan yang dijabarkan pada kesimpulan diatas, aplikasi ini sudah tentu masih jauh dari kata sempurna dan masih memiliki banyak kekurangan. Apabila ada yang berniat untuk mengembangkan sistem ini, Adapun saran agar aplikasi ini bisa berjalan dengan lebih optimal dan lebih menarik sebagai berikut.

1. Aplikasi pembelajaran ini hanya dapat berjalan pada satu *platform* yaitu *Android*. Kelemahan ini menjadi acuan untuk dapat dikembangkan lagi agar dapat digunakan di beberapa *platform*.
2. Penelitian lebih lanjut dapat dilakukan untuk mengukur dampak jangka panjang dari penggunaan aplikasi ini terhadap pengetahuan dan perilaku anak-anak terkait keselamatan lalu lintas.
3. Dalam penelitian ini, aplikasi dibangun dengan hanya menggunakan model 2D bisa dikembangkan model kadaster 3D dan dibuat versi *Android* versi *hybrid*.
4. Sistem aplikasi yang telah dikembangkan penulis sebaiknya dipelihara dengan sebaik-baiknya dan di perbaharui sesuai dengan kebutuhan yang ada serta melakukan *backup* data pada jangka waktu tertentu untuk menghindari kemungkinan data hilang atau rusak.

Demikian saran yang dapat penulis berikan, semoga saran tersebut bisa dijadikan sebagai bahan masukan yang dapat bermanfaat bagi penulis khususnya dan bagi pengembang pada umumnya.