**PENERAPAN METODE *FINITE STATE MACHINE***

**PADA *GAME ADVENTURE* THE MUSLIM**

**SKRIPSI**

Program Studi Informatika

Jurusan Informatika

Oleh:

**NASYIYA ULFA**

D1041151027



FAKULTAS TEKNIK

UNIVERSITAS TANJUNGPURA

PONTIANAK

2020

**Halaman Pernyataan**

Yang bertanda tangan di bawah ini:

Nama : Nasyiya Ulfa

NIM : D1041151027

menyatakan bahwa dalam skripsi yang berjudul “Penerapan Metode *Finite State Machine* pada *Game Adventure* The Muslim” tidak terdapat karya yang pernah diajukan untuk memperoleh gelar sarjana di suatu perguruan tinggi manapun. Sepanjang pengetahuan Saya, tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam Daftar Pustaka.

Demikian pernyataan ini dibuat dengan sebenar-benarnya. Saya sanggup menerima konsekuensi akademis dan hukum di kemudian hari apabila pernyataan yang dibuat ini tidak benar.

Pontianak, Juni 2020

Nasyiya Ulfa

NIM D1041151027

###### 

###### **HALAMAN PERSETUJUAN**

**PENERAPAN METODE *FINITE STATE MACHINE***

**PADA *GAME ADVENTURE* THE MUSLIM**

Skripsi

Program Studi Informatika

Jurusan Informatika

Oleh:

Nasyiya Ulfa

NIM D1041151027

Disetujui untuk diajukan melakukan sidang terbuka

Pembimbing Utama,



H.Hengky Anra, S.T., M.Kom.

NIP 197503251999031005

Pembimbing Pendamping,



Yulianti, S.Kom., MMSI.

NIP 197210162008012005

Puji dan syukur saya ucapkan kepada **ALLAH SWT**

atas segala limpahan berkat dan rahmat yang telah diberikan kepada saya

dalam menyelesaikan skrpisi ini.

Skripsi ini saya persembahkan untuk kedua orang tua saya

**Bapak Kaharudin DL** & **Ibu Ella Nurlaila**

yang selalu mendukung dan mendoakan saya.

Serta tak lupa pula saya ucapkan terima kasih kepada teman-teman saya yang telah memberikan dukungan dan doa dalam menyelesaikan skripsi ini.

“Nasyiya Ulfa”

**Kata Pengantar**

Segala puji dan syukur kehadirat Allah SWT atas berkah, rahamat dan hidyah-Nya yang senantiasa dilimpahakan kepada penulis, sehingga bisa menyelasaikan skripsi dengan judul “Penerapan Metode *Finite State Machine* pada *Game Adventure* The Muslim”.

Penulis dalam menyelesaikan penulisan tugas akhir ini banyak mendapat bantuan dari berbagai pihak, maka dari itu dalam kesempatan ini penulis ingin menyampaikan terima kasih yang sebesar-besarnya kepada Bapak **Hengky Anra, S.T., M.Kom.,** dan Ibu **Yulianti, S.Kom., MMSI.,** selaku pembimbing utama dan pembimbing pendamping yang telah memberikan bimbingan, masukan serta saran dalam pembuatan skripsi ini. Terima kasih disampaikan kepada Bapak **Novi Safriadi, S.T., M.T.,** dan Ibu **Anggi Perwitasari, S.T., M.T., s**elaku penguji utama dan penguji pendamping yang telah banyak membantu dalam memberikan saran serta masukan untuk perbaikan skripsi ini.

Penulis menyadari bahwa dalam penyusunan skripsi ini masih banyak terdapat kekurangan. Oleh karena itu kritik dan saran yang bersifat membangun sangat penulis harapkan guna penyempurnaan penyusunan dan penulisan skripsi ini. Akhir kata, penulis berharap penelitian ini dapat memberikan manfaat bagi pihak-pihak yang membutuhkan.

Pontianak, Juni 2020

Penulis

**Abstrak**

Pendidikan dasar Islam merupakan pengetahuan yang wajib di perkenalkan kepada anak yang berdampak pada aqidah dan akhlak seorang anak. Pada era globalisasi saat ini masih banyak dijumpai kenakalan remaja maupun anak-anak yang jauh dari akhlak mulia. Faktor yang menyebabkan anak-anak berkepribadian buruk salah satunya adalah kurangnya pendidikan agama pada anak. Berdasarkan permasalahan tersebut serta untuk meningkatkan efisiensi penyediaan aplikasi yang mengandung unsur pendidikan, maka dalam penelitian kali ini dibangun sebuah *game* berkonsep edukasi Islam dengan judul “Penerapan Metode *Finite State Machine* Pada *Game Adventure* The Muslim”.Game ini dibangun menggunakan metode *Finite State Machine* untuk mengatur tingkah laku Non Playable Character (NPC) pada game berdasarkan *state* (keadaan), *event* (kejadian) dan *action* (aksi). Dari hasil pengujian dengan menggunakan metode *black box*, aplikasi ini dapat digunakan dan berjalan baik pada perangkat *mobile* android.Berdasarkan hasil kuesioner menggunakan skala Likert Summarted Rating (LSR) diperoleh nilai total 884 yang berarti aplikasi ini dirancang dan dibangun dengan baik dan layak digunakan sebagai sarana dalam mempelajari ilmu dasar islam.

Kata kunci: *Game, Finite State Machine, Non Playable Character,* Pendidikan Dasar Islam

**Abstract**

Islamic basic education is the knowledge that must be introduced to children that have an impact on the faith and character of a child. In the current era of globalization, there are still many delinquents found in adolescents and children who are far from noble morals. One of the factors that cause children with bad personalities is the lack of religious education in children. Based on these problems as well as to improve the efficiency of providing applications that contain educational elements, in this study a game developed with the concept of Islamic education with the title "Application of the Finite State Machine Method in The Muslim Adventure Game". This game was built using the Finite State Machine method to regulate Non-Playable Character (NPC) behavior in the game is based on state, event, and action. From the results of testing using the black box method, this application can be used and runs well on Android mobile devices. Based on the results of the questionnaire using the Likert Summarted Rating (LSR) scale, a total score of 884 is obtained, which means that this application is well designed and built and is fit to be used as a means of learning the basic knowledge of Islam.

Keywords : Game, Finite State Machine, Non Playable Character, Basic knowledge of Islam

**Daftar Isi**

[**Halaman Pernyataan.**](#_heading=h.3ep43zb) **ii**

[**Halaman Persetujuan**](#_heading=h.3ep43zb) **iii**

[**Halaman Persembahan.**](#_heading=h.3ep43zb) **iv**

[**Kata Pengantar.**](#_heading=h.3ep43zb) **v**

[**Abstrak.**](#_heading=h.3ep43zb) **vi**

[***Abstract*.**](#_heading=h.3ep43zb) **vii**

[**Daftar Isi**](#_heading=h.3ep43zb) **viii**

[**Daftar Tabel.**](#_heading=h.3ep43zb) **xi**

[**Daftar Gambar.. .**](#_heading=h.1tuee74)**xiii**

[**Daftar Lampiran x**](#_heading=h.tyjcwt)**iv**

[**Bab I**](#_heading=h.3dy6vkm) **Pendahuluan 1**

[1.1](#_heading=h.1t3h5sf) Latar Belakang 1

[1.2](#_heading=h.4d34og8) Perumusan Masalah 3

[1.3](#_heading=h.2s8eyo1) Tujuan Penelitian 3

[1.4](#_heading=h.17dp8vu) Pembatasan Masalah 3

[1.5](#_heading=h.3rdcrjn) Sistematika Penulisan 4

[**Bab II**](#_heading=h.26in1rg) **Tinjauan Pustaka 5**

21. *Game*.... ...... 5

[2.2](#_heading=h.lnxbz9) *Adventure Game*/Game Petualangan 7

[2.3](#_heading=h.4du1wux) *Game* Edukasi 8

[2.4](#_heading=h.35nkun2) *Finite State Machine* 9

[2.4.1](#_heading=h.3whwml4) Kelebihan FSM 12

[2.4.2](#_heading=h.3whwml4) Kelemahan FSM 12

[2.5](#_heading=h.2szc72q) Kecerdasan Buatan 12

[2.6](#_heading=h.2bn6wsx) Pendidikan Islam 13

[2.6.1](#_heading=h.3whwml4) Tujuan Pendidikan Islam 13

[2.6.2](#_heading=h.3whwml4) Macam Pendidikan Islam 13

[2.7](#_heading=h.44sinio) *Unifed Modelling Language* (UML) 16

[2.7.1](#_heading=h.z337ya) *Use Case* Diagram 17

[2.7.2](#_heading=h.3j2qqm3) *Class* Diagram 18

[2.7.3](#_heading=h.4i7ojhp) *Activity* Diagram 19

[2.8](#_heading=h.44sinio) *Storyboard* 20

[2.8.1](#_heading=h.z337ya) Pengertian *Storyboard* 20

[2.8.1](#_heading=h.z337ya) *Membuat Storyboard* 21

[2.9](#_heading=h.1ci93xb) Android 22

[2.10](#_heading=h.1ci93xb) Apache Cordova 21

[2.11](#_heading=h.184mhaj) Construct 2 23

[2.12](#_heading=h.3s49zyc) *Black Box Testing* 23

[2.13 Skala Likert 26](#_heading=h.147n2zr)

[2.13.1](#_heading=h.3o7alnk) Pengertian Skala Likert 26

[2.13.2](#_heading=h.23ckvvd) Penentuan Skor Jawaban 26

[2.13.3](#_heading=h.32hioqz) Skor Ideal 27

[2.13.4](#_heading=h.1hmsyys) Raiting Scale Skala Likert 27

[2.13.5 Persentase Persetujuan 27](#_heading=h.2grqrue)

[2.14](#_heading=h.147n2zr) Penelitian Terkait 28

[**Bab III**](#_heading=h.3fwokq0) **[Metodologi Penelitian 30](#_heading=h.3fwokq0)**

[3.1](#_heading=h.1v1yuxt) Metodologi Penelitian 30

[3.1.1 Pengumpulan Data/Informasi 30](#_heading=h.4f1mdlm)

[3.2 Alat yang dipergunakan 30](#_heading=h.2u6wntf)

[3.2.1](#_heading=h.19c6y18) Alat Penelitian 30

[3.2.2](#_heading=h.3tbugp1) Perangkat Keras 30

[3.2.3](#_heading=h.28h4qwu) Perangkat Lunak 30

[3.3](#_heading=h.nmf14n) Langkah penelitian 31

[3.4](#_heading=h.1mrcu09) Analisis Kebutuhan Aplikasi 32

[3.5](#_heading=h.2lwamvv) Perancangan *Finite State Machine* NPC 34

[3.5.1](#_heading=h.3l18frh) Skenario FSM Pada NPC *Friendly* 39

3.5.1Skenario FSM Pada NPC *Non-Friendly* 35

[3.6](#_heading=h.279ka65) Perancangan Sistem 39

[3.7](#_heading=h.111kx3o) Perancangan *Unified Modeling Language (UML)* 39

[3.7.1](#_heading=h.3l18frh) *Use Case* Diagram 39

[3.7.2](#_heading=h.1egqt2p) *Activity* Diagram 44

[3.7.3](#_heading=h.sqyw64) *Class* Diagram 48

[3.8](#_heading=h.3cqmetx) Perancangan Aplikasi 50

[3.8.1](#_heading=h.1rvwp1q) Perancangan Flowchart 50

[3.8.2](#_heading=h.3q5sasy) Perancangan Struktur Navigasi 51

[3.8.3](#_heading=h.2iq8gzs) Perancangan Tampilan *Layout* dan *Storyboard* Aplikasi 52

[3.9](#_heading=h.kgcv8k) Perancangan *Aset Game* 57

[3.10](#_heading=h.43ky6rz) Pengujian Aplikasi 60

[3.10.1](#_heading=h.2iq8gzs) Pengujian *Black Box* 60

[3.10.2](#_heading=h.xvir7l) Pengujian Kuesioner 61

[**Bab IV**](#_heading=h.4h042r0) **HASIL DAN ANALISIS 64**

[4.1](#_heading=h.2w5ecyt) Implementasi Aplikasi 64

[4.2](#_heading=h.meukdy) Implementasi Antar Muka Aplikasi 64

[4.2.1 Tampilan](#_heading=h.pkwqa1) *Menu* Utama 64

4.2.2 Tampilan *Menu* Petunjuk 65

[4.2.3 Tampilan](#_heading=h.48pi1tg) *Menu* Pilih *Level* 66

[4.2.4 Tampilan](#_heading=h.pkwqa1) *Menu* *Level* 1 67

4.2.5 Tampilan *Menu* *Level* 2 68

[4.2.6 Tampilan](#_heading=h.48pi1tg) *Menu* *Level* 3 69

[4.2.7 Tampilan](#_heading=h.pkwqa1) *Menu* *Level* 4 70

[4.3](#_heading=h.1302m92) *Export* Aplikasi 71

4.4Pengujian Metode *Finite State Machine* 72

[4.5](#_heading=h.3mzq4wv) Pengujian Aplikasi 74

[4.5.1](#_heading=h.2250f4o) Pengujian *Black Box* 74

[4.5.2](#_heading=h.haapch) Hasil PengujianKuesioner 78

[4.5.3](#_heading=h.haapch) *User Acceptence Test* 86

[4.6](#_heading=h.3mzq4wv) Analisis Hasil Pengujian 88

[**Bab V**](#_heading=h.1gf8i83) **Penutup 89**

[5.1](#_heading=h.40ew0vw) Kesimpulan 89

[5.2 Saran 89](#_heading=h.2fk6b3p)

[**Daftar Pustaka 89**](#_heading=h.1gf8i83)

**DAFTAR TABEL**

[**Tabel 2. 1** Deskripsi Notasi Pada *Use Case* Diagram 18](#_heading=h.1y810tw)

[**Tabel 2. 2** Deskripsi Notasi Pada *Class* Diagram 18](#_heading=h.1y810tw)

[**Tabel 2. 3** Deskripsi Notasi Pada *Activity* Diagram 19](#_heading=h.2xcytpi)

[**Tabel 2. 4** Skala Nilai Likert 26](#_heading=h.ihv636)

[**Tabel 2. 5** Rumus Skala Likert 2](#_heading=h.36ei31r)7

[**Tabel 2. 6** Hasil Penelitian Skala Likert 27](#_heading=h.41mghml)

[**Tabel 2. 7** Kajian Terkait 27](#_heading=h.41mghml)

[**Tabel 3. 1**Spesifikasi Kebutuhan 3](#_heading=h.206ipza)3

[**Tabel 3. 2** *Use Case* Aktor 39](#_heading=h.206ipza)

[**Tabel 3. 3** Definisi *Use Case* 4](#_heading=h.4k668n3)0

[**Tabel 3. 4** Skenario *Use Case* Melihat Petunjuk 4](#_heading=h.1ljsd9k)1

[**Tabel 3. 5** Skenario *Use Case* Memilih *Play* 4](#_heading=h.206ipza)1

[**Tabel 3. 6** Skenario *Use Case* Memilih *Level* 4](#_heading=h.206ipza)1

[**Tabel 3. 7** Skenario *Use Case* Memainkan *Game* 4](#_heading=h.4k668n3)2

[**Tabel 3. 8** Skenario *Use Case Game Over* 4](#_heading=h.206ipza)3

[**Tabel 3. 9** Skenario *Use Case* Menutup Aplikasi 4](#_heading=h.4k668n3)4

[**Tabel 3. 10** Perancangan Tampilan *Layout* dan *Storyboard* Aplikasi 53](#_heading=h.25b2l0r)

[**Tabel 3. 11** *Aset Images* 5](#_heading=h.34g0dwd)7

[**Tabel 3. 12** *Aset Souds* 5](#_heading=h.1jlao46)9

[**Tabel 3. 13** Rencana Pengujian Aplikasi 6](#_heading=h.3hv69ve)0

[**Tabel 3. 14** Rencana Kasus dan Hasil Uji Aplikasi...............................................6](#_heading=h.3hv69ve)0

[**Tabel 3. 15** Rancangan Kuesioner 6](#_heading=h.34g0dwd)2

[**Tabel 3. 16** Skor Hasil Kuesioner 6](#_heading=h.1jlao46)3

[**Tabel 3. 17** Table Kelayakan.................................................................................6](#_heading=h.3hv69ve)3

[**Tabel 4. 1** Hasil Uji Coba Metode FSM pada NPC *Friendly*.](#_heading=h.45jfvxd)...............................72

[**Tabel 4. 2** Hasil Uji Coba Metode FSM pada Musuh Lebah](#_heading=h.2koq656) 73

[**Tabel 4. 3** Hasil Uji Coba Metode FSM pada Musuh Tikus](#_heading=h.2koq656) 73

[**Tabel 4. 4** Hasil Uji Coba Metode FSM pada Musuh Siput](#_heading=h.2koq656) 74

[**Tabel 4. 5** Rencana Pengujian Aplikasi](#_heading=h.zu0gcz) 75

[**Tabel 4. 6** Pengujian Menu Melihat Petunjuk](#_heading=h.3jtnz0s) 75

[**Tabel 4. 7** Pengujian Menu Memilih *Play*](#_heading=h.zu0gcz) 76

[**Tabel 4. 8** Pengujian Menu Memilih *Level*](#_heading=h.3jtnz0s) 76

[**Tabel 4. 9** Pengujian Menu Memainkan *Game*](#_heading=h.3jtnz0s) 76

[**Tabel 4.10** Pengujian Menu *Game Over*](#_heading=h.zu0gcz) 77

[**Tabel 4.11** Pengujian Menu Menutup Aplikasi](#_heading=h.3jtnz0s) 78

[**Tabel 4.12** Hasil Pengisian Kuisioner](#_heading=h.3jtnz0s) 79

[**Tabel 4.13** Hasil Persentase Kuisioner Nomor 1](#_heading=h.zu0gcz) 79

[**Tabel 4.14** Hasil Persentase Kuisioner Nomor 2](#_heading=h.3jtnz0s) 80

[**Tabel 4.15** Hasil Persentase Kuisioner Nomor 3](#_heading=h.zu0gcz) 81

[**Tabel 4.16** Hasil Persentase Kuisioner Nomor 4](#_heading=h.zu0gcz) 82

[**Tabel 4.17** Hasil Persentase Kuisioner Nomor 5](#_heading=h.3jtnz0s) 82

[**Tabel 4.18** Hasil Persentase Kuisioner Nomor 6](#_heading=h.zu0gcz) 83

[**Tabel 4.19** Hasil Persentase Kuisioner Nomor 7](#_heading=h.zu0gcz) 84

[**Tabel 4.20** Hasil Persentase Kuisioner Nomor 8](#_heading=h.3jtnz0s) 84

[**Tabel 4.21** Hasil Persentase Kuisioner Nomor 9](#_heading=h.zu0gcz) 85

[**Tabel 4.22** Hasil Persentase Kuisioner Nomor 10](#_heading=h.3jtnz0s) 86

[**Tabel 4.23** Hasil Total Skor Responden](#_heading=h.zu0gcz) 86

**DAFTAR GAMBAR**

[**Gambar 2. 1** A](#_heading=h.2jxsxqh)lur dari Metode *Finite State Machine*...........................................10

[**Gambar 2. 2***M*](#_heading=h.2jxsxqh)*oore State Machine.*......................................................................11

[**Gambar 2. 3***M*](#_heading=h.2jxsxqh)*early State Machine*......................................................................11

[**Gambar 2. 4** Jenis Kategori Diagram UML 17](#_heading=h.2jxsxqh)

[**Gambar 3. 1** Langkah-Langkah Penelitian...........................................................32](#_heading=h.37m2jsg)

[**Gambar 3. 2** Alur FSM pada Karakter Pemberi Pertanyaan](#_heading=h.1yyy98l) 35

[**Gambar 3. 3** Alur FSM pada Musuh Lebah 3](#_heading=h.2zbgiuw)6

[**Gambar 3. 4** Alur FSM pada Musuh Tikus 3](#_heading=h.2zbgiuw)7

[**Gambar 3. 5** Alur FSM pada Musuh Siput 3](#_heading=h.2zbgiuw)8

[**Gambar 3. 6** *Use Case Game* Edukasi The Muslim 4](#_heading=h.2dlolyb)0

[**Gambar 3. 7** *Activity* Diagram Petunjuk 4](#_heading=h.4iylrwe)4

[**Gambar 3. 8** *Activity* Diagram Memilih *Play* 4](#_heading=h.2dlolyb)5

[**Gambar 3. 9** *Activity* Diagram Memilih *Level* 4](#_heading=h.4iylrwe)5

[**Gambar 3.10** *Activity* Diagram Memainkan *Game* 4](#_heading=h.4iylrwe)6

[**Gambar 3.11** *Activity* Diagram *Game Over* 4](#_heading=h.4iylrwe)7

[**Gambar 3.12** *Activity* Diagram Keluar Aplikasi 4](#_heading=h.4iylrwe)8

[**Gambar 3.13** *Class Diagram* Game Edukasi The Muslim 4](#_heading=h.4iylrwe)9

[**Gambar 3.14** *Flowchart* Game Edukasi The Muslim...........................................5](#_heading=h.4iylrwe)0

[**Gambar 3.15** Perancangan Struktur Navigasi Game Edukasi The Muslim 51](#_heading=h.2r0uhxc)

[**Gambar 4. 1** Tampilan Menu Utama....................................................................](#_heading=h.2y3w247)65

[**Gambar 4. 2** Tampilan Menu Petunjuk Slide 1 6](#_heading=h.2afmg28)5

[**Gambar 4. 3** Tampilan Menu Petunjuk Slide 2 6](#_heading=h.39kk8xu)6

[**Gambar 4. 4** Tampilan *Menu* Pilih Level](#_heading=h.1d96cc0) 66

[**Gambar 4. 5** Tampilan *Menu* Level 1 6](#_heading=h.2nusc19)7

[**Gambar 4. 6** Tampilan Pertanyaan Level 1 6](#_heading=h.3x8tuzt)8

[**Gambar 4. 7** Tampilan *Menu* Level 2 6](#_heading=h.2nusc19)8

[**Gambar 4. 8** Tampilan Pertanyaan Level 2 6](#_heading=h.3x8tuzt)9

[**Gambar 4. 9** Tampilan *Menu* Level 3 6](#_heading=h.2nusc19)9

[**Gambar 4.10** Tampilan Pertanyaan Level 3 7](#_heading=h.3x8tuzt)0

[**Gambar 4.11** Tampilan *Menu* Level 4 7](#_heading=h.2nusc19)0

[**Gambar 4.12** Tampilan Pertanyaan Level 4 7](#_heading=h.3x8tuzt)1

[**Gambar 4.13** Proses Export ke Format *Cordova*](#_heading=h.2ce457m) 71

[**Gambar 4.14** Proses Export ke Format *.apk*](#_heading=h.rjefff) 72

**Daftar Lampiran**

Lampiran

A Dokumentasi Pengujian Di Bimbel Inawa Education...………….A-1

[Lampiran B](#_heading=h.3bj1y38) Lampiran Kuesioner Game Edukasi The Muslim.........................B-1

[Lampiran C](#_heading=h.3bj1y38) Hasil Kuesioner Game Edukasi The Muslim................................C-1

# Pendahuluan

## Latar Belakang

Pendidikan agama yang baik dalam keluarga adalah salah satu contoh perhatian orang tua terhadap anak agar dapat tumbuh dan berkembang menjadi manusia yang bermoral. Pendidikan dasar Islam merupakan pengetahuan yang wajib di perkenalkan kepada anak yang berdampak pada aqidah dan akhlak seorang anak. Pada era globalisasi saat ini masih banyak dijumpai kenakalan remaja maupun anak-anak yang jauh dari akhlak mulia seperti tidak patuh kepada orang tua dan guru, sering berkelahi, mencuri dan sebagainya. Faktor yang menyebabkan anak-anak berkepribadian buruk salah satunya adalah kurang tertanamnya jiwa agama pada tiap-tiap orang dalam masyarakat (Zakiah Daradjat, 1971).

Dalam upaya meningkatkan efisiensi penyediaan aplikasi yang mengandung unsur pendidikan diperlukan berbagai alternatif dan inovasi baru dalam hal pemrograman untuk bisa diterapkan sebagai alat untuk mempermudah proses pembelajaran. Salah satu media pembelajaran bisa berupa *game*, karena *game* (permainan) adalah bagian mutlak dari kehidupan anak dan permainan merupakan bagian integral dari proses pembentukan kepribadian anak (Andang Ismail, 2009).

*Game* yang memiliki konten pendidikan lebih dikenal dengan istilah *game* edukasi.  *Game* berjenis edukasi ini bertujuan untuk memancing minat belajar anak terhadap materi pelajaran sambil bermain, sehingga dengan perasaan senang diharapkan siswa bisa lebih mudah memahami materi pelajaran yang disajikan, jenis ini sebenarnya lebih mengacu kepada isi dan tujuan game, bukan jenis yang sesungguhnya, game edukasi dapat berpengaruh bagi pelajar tingkat sekolah dasar (SD) sampai sekolah menengah atas (SMA), apabila siswa/i dalam proses pembelajaran mengalami kesulitan dalam mengerti atau mencerna ajaran yang diberikan oleh guru, siswa cenderung mengalami kejenuhan dan akhirnya tidak ada minat akan pelajaran tersebut. Akibat dari kejenuhan tersebut menimbulkan dampak yang sangat besar

dikemudian hari, seperti anak tersebut tidak akan pernah senang akan pelajaran tersebut dan setiap ujian tidak mendapat hasil yang baik (Cristianto Tjandra, 2011).

Berdasarkan permasalahan yang telah dikemukakan oleh (Zakiah Daradjat, 1971) tentang faktor kenakalan anak-anak pada saat ini serta untuk meningkatkan efisiensi penyediaan aplikasi yang mengandung unsur pendidikan yang bisa diterapkan sebagai alat untuk mempermudah proses pembelajaran, maka dalam penelitian kali ini dibangun sebuah *game* berkonsep edukasi Islam dengan judul “Penerapan Metode *Finite State Machine* Pada *Game Adventure* The Muslim”.

The Muslim merupakan *game* edukasi untuk anak yang dirancang sebagai media pembelajaran dalam memberikan ilmu dasar Islam berbasis android. Adapun jenis *game* yang dibuat yaitu bergenre *adventure* (petualangan) dimana pemain akan disajikan beberapa pertanyaan yang akan muncul pada setiap *stage* mulai dari Tauhid, Aqidah & Akhlak, Al-Quran & Hadist, Sejarah Islam & Fiqih serta pemain wajib menjawab pertanyaan-pertanyaan tersebut. Selain itu terdapat pula *enemy* (musuh) yang akan muncul di masing-masing *stage*.

*Game* ini dibangun dengan menggunakan metode Finite State Machine. *Finite State Machine* (FSM) adalah salah satu bidang *Artificial Intelligence* (AI) yang bisa diaplikasikan pada game AI. Prinsip kerjanya dengan menggunakan tiga hal berikut : *state* (keadaan), *event* (kejadian) dan *action* (aksi). (Setiawan, 2006) Implementasi FSM pada game ini yaitu pada *Non Playable Character* atau NPC.

*Non-Playable Character* merupakan jenis autonomus agent yang ditunjukan untuk penggunaan komputer animasi dan media interaktif seperti games dan virtual reality. Agen ini mewakili tokoh dalam cerita atau permainan dan memiliki kemampuan untuk improvisasi tindakan mereka. Ini adalah kebalikan dari seorang tokoh dalam sebuah film animasi, yang tindakannya ditulis di muka, dan untuk “avatar” dalam sebuah game atau virtual reality, tindakan yang diarahkan secara real time oleh player. (Reynolds, 1999).

Penelitian ini dilakukan untuk membangun game berjudul “The Muslim” dengan metode *Finite State Machine*. Metode FSM tersebut akan diterapkan pada karakter NPC yang muncul dalam game baik NPC yang bersifat *Friendly* dan *Non*

*Friendly*. NPC *Friendly* yaitu NPC yang bertugas memberi pertanyaan seputar ilmu dasar islam, apabila player mampu menjawab pertanyaan dengan benar maka NPC *Friendly* akan memberikan *reward* atau hadiah kepada *player*. Sedangkan pada NPC *Non Friendly* adalah musuh yang akan muncul pada setiap *stage/level*. Terdapat berbagai jenis karakter musuh yang berbeda pada *game* ini yang akan menjadi rintangan pemain dalam memainkan *game* The Muslim.

## Perumusan Masalah

Berdasarkan latar belakang telah diuraikan diatas, didapatkan suatu perumusan masalah, yaitu bagaimana merancang sebuah *game* edukasi The Muslim menggunakan metode Finite State Machine sebagai pembelajaran dasar islam.

## Tujuan Penelitian

Tujuan dari penelitian ini yaitu menghasilkan sebuah *game* edukasi The Muslimmenggunakan metode Finite State Machine sebagai pembelajaran dasar islam.

## Pembatasan Masalah

Pembatasan masalah dari penelitian yang akan dilakukan adalah :

1. *Game* ini berbasis Android yang dapat dimainkan oleh satu pemain dengan tampilan dua dimensi (2D) digunakan secara *offline.*

2. *Game* yang dibangun menggabungkan dua jenis game yaitu *platform* *adventure game* dan *questions game* dengan menggunakan metode *Finite State Machine* (FSM) yang diimplementasikan pada *Non Playable Character* baik yang bersifat *Friendly* maupun *Non-Friendly.*

3. *Stage* yang ada pada *game* ini adalah Tauhid, Aqidah & Akhlak, Al-Qur’an & Hadist, serta Sejarah Islam & Fiqih.

4. Pemain diharuskan menjawab pertanyaan yang muncul di setiap *stage* pada saat melakukan petualangan.

5. *Game* dibuat menggunakan aplikasi Construct 2.

6. Sasaran pengguna yaitu anak-anak usia 8-12 tahun.

## Sistematika Penulisan

Adapun sistematika dari penulisan tugas akhir ini disusun dalam lima bab yang terdiri dari Bab I Pendahuluan, Bab II Tinjauan Pustaka, Bab III Metodologi Penelitian dan Perancangan Sistem, Bab IV Hasil Perancangan dan Analisis Sistem, serta Bab V Penutup.

**Bab I Pendahuluan** adalah bab yang berisi latar belakang, perumusan masalah, tujuan penelitian, pembatasan masalah, dan sistematika penulisan.

**Bab II Tinjauan Pustaka** adalah bab yang berisi landasan teori berkaitan dengan penelitian yang akan dilakukan, yaitu uraian tentang hasil penelitian yang dilakukan oleh peneliti lain..

**Bab III Metodologi Penelitian** adalah bab yang berisi tentang bahan penelitian, alat yang dipergunakan, metode penelitian, variabel atau data.

**Bab IV Hasil dan Analisis** adalah bab yang berisi penjelasan mengenai perancangan game edukasi The Muslim.

**Bab V Penutup** adalah bab yang berisi kesimpulan dari penelitian yang telah dilakukan dan saran/rekomendasi untuk perbaikan, pengembangan atau kesempurnaan / kelengkapan penelitian yang telah dilakukan.

**BAB II**

**TINJAUAN PUSTAKA**

## 2.1 *Game*

*Game*”diartikan sebagai permainan. Permainan merupakan bagian dari bermain dan bermain juga bagian dari permainan keduanya saling berhubungan. Permainan adalah kegiatan yang kompleks yang didalamnya terdapat peraturan, play dan budaya. Sebuah permainan adalah sebuah sistem dimana pemain terlibat dalam konflik buatan, disini pemain berinteraksi dengan sistem dan konflik dalam permainan merupakan rekayasa atau buatan, dalam permainan terdapat peraturan yang bertujuan untuk membatasi perilaku pemain dan menentukan permainan.*Game* bertujuan untuk menghibur, biasanya *game* banyak disukai oleh anak – anak hingga orang dewasa (Tim Pusat Bahasa DEPDIKNAS, 2008).

Dalam penerapannya sekarang, *game* dapat dimainkan dalam banyak media yang berbeda yaitu :

a. *Arcade games* yang biasanya memiliki *box* atau mesin yang memang khusus di desain untuk jenis *video* *games* tersebut dan didukung dengan perangkat, seperti pistol, kursi khusus, sensor gerakan, sensor injakkan dan stir mobil.

b. *PC Games*, yaitu *video game* yang dimainkan menggunakan *Personal Computers (PC).*

c. *Console games,* yaitu *video games* yang dimainkan menggunakan *console* tertentu, misalnya *Playstation 3, Playstation 4, XBOX, dan Nintendo Wii*.

d. *Handheld games*, yaitu yang dimainkan di *console* khusus *video game* yang dapat dibawa ke mana-mana, contoh *Nintendo DS* dan *Sony PSP.*

e. *Mobile games*, yaitu *game* yang dapat dimainkan atau khusus untuk *mobile phone* (Bangun et al., 2014).

Tujuan utama dari bermain game biasanya adalah untuk menghibur diri dan mengisi waktu luang. *Game* banyak disukai oleh anak – anak hingga orang dewasa. *Game* sebenarnya penting untuk perkembangan otak, untuk meningkatkan konsentrasi dan melatih untuk memecahkan masalah dengan tepat dan cepat karena dalam *game* terdapat berbagai konflik atau masalah yang menuntut kita untuk menyelesaikannya dengan cepat dan tepat. Namun demikian,

*game* juga bisa merugikan karena apabila kita sudah kecanduan *game* kita bisa lupa waktu dan akan mengganggu kegiatan atau aktifitas yang sedang kita lakukan (Chikhungunya, 2011).

*Game* sendiri memiliki berbagai macam *genre* dengan cara bermain yang berbeda-beda pula. Berikut jenis klasifikasi *genre game* tersebut.

a. *Shooter*, yaitu jenis *game* yang dimainkan menggunakan sebuah senjata, biasanya pistol, senapan, atau senjata jarak jauh lainnya.

b.*Strategy*, adalah jenis *game* yang mengharuskan pemainnya menggunakan taktik dan strategi guna menyelesaikan permainan.

c. *Racing*, merupakan *game* yang dimainkan dengan mengendalikan sebuah kendaraan.

d. *Arcade*, adalah jenis *game* yang tidak terfokus pada cerita, melainkan hanya dimainkan *“just for fun”.*

e. *Fighting*, merupakan jenis game pertarungan antar player dengan lawannya.

f. *Sports*, yaitu jenis *game* bertema olahraga.

g. *Vehicle Simulation*, yaitu mensimulasi pengoperasian beberapa kendaraan.

h. *Adventure*, adalah jenis *game* yang umumnya membuat pemain harus berjalan mengelilingi suatu tempat yang telah didesain sedemikian rupa.

i. *Musical*, merupakan jenis *game* yang berhubungan dengan music, *player* baisanya dapat memainkan berbagai alat musik.

j. *Platform*, adalah jenis *game* yang mengharuskan pemain mengarahkan suatu obyek dengan melalui berbagai tahap atau tingkatan area.

k. *Role Playing Game (RPG*), jenis *game* di mana *player* memainkan peran tertentu atau karakter tertentu pada suatu cerita dalam game.

l. *Action*, adalah jenis *game* yang identic dengan banyaknya aksi di mana pemain harus memiliki keterampilan dan reaksi yang cepat untuk menyelesaikannya.

m. *Puzzle*, merupakan game yang berintikan teknik pemecahan teka-teki.

n. *Real Life*, yaitu *game* yang dibuat seperti kehidupan manusia sehari-hari.

o. *Historic*, adalah *game* yang dibuat berdasarkan cerita-cerita sejarah pada zaman dahulu (Bangun et al., 2014).

## 2.2 *Adventure Game* / Game Petualangan

Menurut David Fox dan Roman Verhosek (2002:9), *game* dapat diklasifikasikan antara lain *Action Game, Combat Game, Adventure Game, Puzzle Game, Strategy Game,* dan *Card Game*. Pada penelitian ini, *game* dibangun dengan jenis *Adventure Game* atau *game* petualangan.

*Adventure game* sering dianggap sebagai bentuk interaksi fiksi. Interaksi fiksi merupakan sebuah istilah yang mengacu pada media dimana pemain dapat mempengaruhi hasil cerita.

Karakter kunci dari *game* petualangan sebagai berikut:

1. Narasi sebagai daya tarik dimana pergerakan pemain sebagai hasil kemajuan permainan
2. Narasi seringkali diambil dari film, komik dan lain sebagainya.
3. Pemain umumnya mengontrol karakter utama
4. Permainan sering didasarkan pertanyaan atau teka teki, yang harus diselesaikan dengan berinteraksi dengan lingkungan permainan dan objek objek yang menghasilkan pengalaman bagi pemain.
5. Penekanan dalam *game* petulangan adalah pada eksplorasi, berpikir dan kemampuan pemecahan masalah secara cepat sebagai gaya aksi permainan
6. Unsur – unsur mendasar diantaranya:
   1. *Game*s *Rules*: dijelaskan oleh penulis (pembuat *game*), yang mengatur operasi dan fungsi objek maupun karakter dalam permainan. Peraturan juga bisa didapatkan ketika pemain telah bermain.
   2. *Game World* : objek yang ada dalam permainan gua, pulau, penduduk dan lain sebagainya.
   3. *Plot* : Berisi informasi apa yang terjadi sebelum pemain bermain dan tujuan menyelesaikan permainan dan beberapa cara untuk mnyelesaikan permainan.
   4. *Theme* : Tema moral yang biasanya mendasari permainan. seperti pemaian diharuskan menjawab pertanyaan jika berhasil maka dapat memulihkan keseimbangan lingkungan.
   5. *Characters*: Player maupun NPC cenderung untuk memproses sifat tertentu atau atribut (kekuatan sihir, fitur- fitur khusus, wajah dsb)
   6. *Object / Item* : mempunyai peran penting dan biasanya dikumpulkan dan digunakan oleh pemain untuk memecahkan masalah. Seringkali pemain harus memiliki keahlian tertentu atau pengetahuan untuk menggunakannya.
   7. *Text, Graphics* dan *Sound* : *game* melibatkan kombinasi dari teks, grafis dan suara.
   8. *Animation*: animasi biasanya telah diprogram dan ditanamkan dalam permainan, yang berfungsi membantu dan memberi kesenangan pada pemain.
   9. *User Interface* : tampilan pada layar memungkinkan pengguna untuk berkomunikasi dengan mudah melalui pemilihan teks, grafis, suara dan animasi. (Dillon, 2005)

## 2.3 *Game* Edukasi

Dalam dunia edukasi, interaksi antara murid dengan gurunya merupakan suatu hal yang mutlak. Jika murid harus manusia, maka guru bisa berupa benda lain seperti buku, pengalaman, dan lain sebagainya. Salah satunya adalah *game*. Karena *game* merupakan program yang bersifat interaktif, bisa merespon apa yang dilakukan pengguna, sehingga proses belajar pun bisa terjadi.

Penggunaan *game* sebagai sarana edukasi sebenarnya bukan merupakan hal yang *negatif*, karena *game* bersifat *‘entertain’* atau menghibur yang bermain. Psikologi manusia adalah lebih suka bermain daripada belajar serius. Dalam *game* pendidikan diberikan lewat praktek atau pembelajaran dengan praktek. Dalam *game* pemain seolah masuk ke dalam dunia baru tempat mereka bisa melakukan apa saja. *Game* secara tidak langsung mendidik manusia lewat apa yang mereka kerjakan dalam *game* tersebut. Apa yang mereka kerjakan dalam *game* tersebut bisa mempengaruhi pola pikir dan perilaku mereka. Ini merupakan bagian dari edukasi, namun, karena banyak *game* pada saat ini lebih mengedepankan kekerasan dan kriminalitas dalam permainannya, maka pendidikan yang diberikan pun adalah pendidikan yang tidak baik. Dalam hal ini yang mendapat efek paling besar adalah anak-anak, karena mereka terkadang tidak dapat menyaring informasi yang di dapatkan, yang akhirnya membuat mereka merasa bebas melakukan kekerasan di dunia nyata sebagai mana yang mereka lakukan di dunia maya.

Pada awalnya *game* yang mempunyai model edukasi adalah *game* berjenis *puzzle.* Yaitu game pemecahan teka-teki. Teka-teki yang ada dalam *game* ini berupa pelajaran. Secara tidak langsung, ketika anak-anak tidak menemukan jawaban dari teka-teki ini, mereka akan langsung mencari jawaban, baik di buku, intenet, atau bertanya pada orang lain. Inilah sistem edukasi dalam *game* yang ada sampai saat ini.

## **2.4 *Finite State Machine***

*Finite State Machines (FSM)* adalah sebuah metodologi perancangan sistem kontrol yang menggambarkan tingkah laku atau prinsip kerja sistem dengan menggunakan tiga hal berikut: *State* (Keadaan), *Event* (kejadian) dan *Action* (aksi). Pada satu saat dalam periode waktu yang cukup signifikan, sistem akan berada pada salah satu *state* yang aktif. Sistem dapat beralih atau bertransisi menuju state lain jika mendapatkan masukan atau *event* tertentu, baik yang berasal dari perangkat luar atau komponen dalam sistemnya itu sendiri (misal *interupsi timer*). Transisi keadaan ini umumnya juga disertai oleh aksi yang dilakukan oleh sistem ketika menanggapi masukan yang terjadi. Aksi yang dilakukan tersebut dapat berupa aksi yang sederhana atau melibatkan rangkaian proses yang relative kompleks (Setiawan, 2006).

Berdasarkan sifatnya, metode FSM ini sangat cocok digunakan sebagai basis perancangan perangkat lunak pengendalian yang bersifat reaktif dan real time. Salah satu keutungan nyata penggunaan FSM adalah kemampuannya dalam mendekomposisi aplikasi yang relative besar dengan hanya menggunakan sejumlah kecil item state. Selain untuk bidang kontrol, Penggunaan metode ini pada kenyataannya juga umum digunakan sebagai basis untuk perancangan protokol-protokol komunikasi, perancangan perangkat lunak game, aplikasi WEB dan sebagainya (Setiawan, 2006).

Dalam FSM masing-masing karakter menempati satu state. Biasanya, tindakan atau perilaku yang terkait dengan masing-masing state. Jadi selama karakter tetap dalam keadaan itu, ia akan terus melakukan tindakan yang sama. State terhubung bersama oleh transition. Setiap transition mengarah dari satu state ke state lain yang biasanya state tujuan / *state*  target ini disebut dengan action dan masing-masing memiliki seperangkat kondisi yang terkait. Jika permainan menentukan bahwa kondisi transition terpenuhi, maka karakter berubah dari state ke state target (action) melalui transition itu.

*Finite State Machine* (FSM) adalah membagi sebuah respon objek *game* kedalam bagian-bagian (state) sehingga objek tersebut memilikki bagian untuk setiap respon objek *game*. Implementasi tersebut menghasilkan suatu urutan scenario tertentu pada *game*. Sehingga dalam *game* akan terdapat alur permainan yang harus di lewati nantinya yang dapat mendifisikan suatu set kondisi yang menentukan kapan suatu bagian harus berubah ke bagian yang lain.

Gambar Alur Metode Finite State Machine dapat dilihat pada gambar dibawah ini :



**Gambar 2.1** Alur dari metode Finite State Machine (Sumber: Brownlee, 2010)

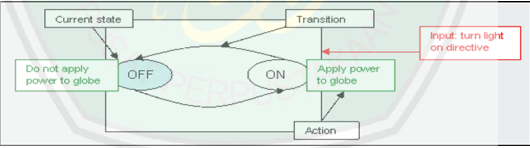
Pada Gambar 1 terdapat 4 *state* {Si, S1, S2, S3} yang mungkin terjadi, setiap *state*-nya dapat berpindah *state* jika kondisi terpenuhi. Sebagai contoh *state* S1 dapat berpindah jika kondisi t5 terpenuhi.

Dalam diagram ini, state-state yang terdapat pada sebuah sistem digambarkan sebagai lingkaran yang diberi label unik, sedangkan transisi state yang diakibatkan oleh event tertentu direpresentasikan sebagai anak panah yang berasal dari state yang ditinggalkan menuju state yang aktif. Setiap transisi yang terjadi umumnya juga diikuti oleh aksi yang dilakukan oleh sistem yang dirancang. Secara praktis setiap diagram state yang dirancang akan selalu memiliki sebuah transisi awal (insial) yang menuju salah satu state sejak sistem kontrol tersebut mulai dihidupkan.

FSM terdiri dari dua jenis, yaitu FSM ber-*output* dan FSM tidak ber- *output*. FSM tidak ber-*output* digunakan untuk pengenalan bahasa dalam komputer, dengan *input* yang dimasukkan akan diperoleh apakah *input* tersebut dikenal oleh bahasa komputer atau tidak. Salah satu penggunaan FSM tidak ber- *output* adalah program *compiler*, yaitu program untuk memeriksa apakah perintah yang digunakan pengguna benar atau salah. Sementara untuk FSM ber-*output* digunakan untuk merancang mesin atau sistem (Zen, 2008). Dan FSM yang akan digunakan dalam penelitian ini adalah FSM ber-*output*, dan untuk selanjutnya akan dituliskan dengan FSM saja

Ada dua metode utama untuk memperlakukan FSM untuk menghasilkan output. Yaitu *Moore Machine* dan *Mearly Machine* yang dinamakan berdasarkan penemunya.

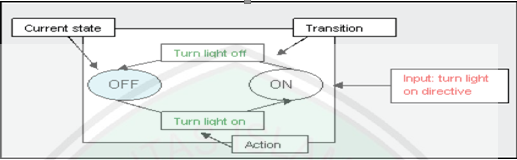
1. *Moore Machine*



**Gambar 2.2** *Moore State Machine* (Sumber: Brownlee, 2010)

*Moore Machine* adalah tipe dari FSM dimana output dihasilkan dari *state*. Pada gambar diatas mencontohkan dimana *state* mendefenisikan apa yang harus dilakukan (Brownlee, 2010). Keluaran pada *Moore Machine* diasosiasikan sebagai *state* (Hariyanto, 2004)*.*

1. *Mearly Machine*



**Gambar 2.3** *Mearly State Machine* (Sumber: Brownlee, 2010)

*Mearly Machine* berbeda dengan *Moore Machine* dimana keluarannya merupakan hasil dari transisi antar *state* (Brownlee, 2010). Keluaran pada *Mearly Machine* diasosiasikan sebagai transisi (Hariyanto, 2004)

## **Kelebihan FSM**

FSM memiliki beberapa kelebihan (Brownlee, 2010), diantaranya :

* + - 1. Sederhana, sehingga mudah diimplementasikan
      2. Bisa diprediksi responnya
      3. Komputasi ringan
      4. Relatif fleksibel
      5. Merupakan metode AI lama yang bisa digunakan pada berbagai sistem
      6. Mudah ditransfer dari abstrak menjadi kode program

## **Kelemahan FSM**

Selain memiliki banyak kelebihan, FSM juga mempunyai beberapa kelemahan (Brownlee, 2010), diantaranya :

* + 1. Karena sifatnya bisa diprediksi, maka implementasi pada *game* kurang disukai.
    2. Implementasi pada sistem yang lebih besar lebih sulit karena pengaturan dan pemeliharaannya jadi kompleks.
    3. Sebaiknya hanya digunakan pada sistem dimana sifat sistem bias didekomposisi menjadi state.
    4. Kondisi untuk transisi *state* adalah tetap

## **2.5 Kecerdasan Buatan**

Kecerdasan buatan / *Artificial Intelligence* adalah bagian dari ilmu komputer yang mempelajari tentang bagaimana sebuah komputer bisa dibuat dengan sedemikian rupa agar dapat melakukan pekerjaan seperti dan sebaik yang dilakukan oleh manusia. Menurut John McCarthy (1956), kecerdasan buatan adalah suatu sistem komputer yang terbentuk untuk mengetahui dan memodelkan proses-proses berpikir manusia dan mendesain mesin agar dapat menirukan perilaku manusia.

Manusia bisa dengan pandai menyelesaikan masalah-masalah yang muncul karena manusia memiliki pengetahuan dan pengalaman yang dapat membantu dalam memecahkan masalah. Agar komputer dapat bertindak seperti dan sebaik manusia maka komputer diberikan pengetahuan dan kemampuan untuk menalar agar dapat mendapatkan pengalaman seperti layaknya manusia.

Ada tiga tujuan kecerdasan buatan, yaitu: membuat komputer lebih cerdas, mengerti tentang kecerdasan, dan membuat mesin lebih berguna. Yang dimaksud kecerdasan adalah kemampuan untuk belajar atau mengerti dari pengalaman, memahami pesan yang kontradiktif dan ambigu, menanggapi dengan cepat dan baik atas situasi yang baru, menggunakan penalaran dalam memecahkan masalah serta menyelesaikannya dengan efektif (Winston dan Prendergast, 1994).

## **2.6 Pendidikan Islam**

Pendidikan Islam adalah usaha orang dewasa muslim yang bertakwa secara sadar mengarahkan dan membimbing pertumbuhan serta perkembangan fitrah (kemampuan dasar) anak didik melalui ajaran Islam ke arah titik maksimal pertumbuhan dan perkembangannya (Akaha, 2001, 154-155).

Pendidikan Islam merupakan sebuah usaha untuk menjadikan anak keturunan dapat mewarisi ilmu pengetahuan (berwawasan islam). Setiap usaha dan tindakan yang disengaja untuk mencapai tujuan harus mempunyai sebuah landasan atau dasar tempat berpijak yang baik dan kuat.

### **2.6.1 Tujuan Pendidikan Islam**

Tujuan pendidikan merupakan faktor yang sangat penting, karena merupakan arah yang hendak dituju oleh pendidikan itu. Demikian pula halnya dengan Pendidikan Agama Islam, yang tercakup mata pelajaran akhlak mulia dimaksudkan untuk membentuk peserta didik menjadi manusia yang beriman dan bertakwa kepada Tuhan Yang Maha Esa serta berakhlak mulia. Akhlak mulia mencakup etika, budi pekerti, atau moral sebagai perwujudan dari pendidikan agama.

Tujuan pendidikan secara formal diartikan sebagai rumusan kualifikasi, pengetahuan, kemampuan dan sikap yang harus dimiliki oleh anak didik setelah selesai suatu pelajaran di sekolah, karena tujuan berfungsi mengarahkan, mengontrol dan memudahkan evaluasi suatu aktivitas sebab tujuan pendidikan itu adalah identik dengan tujuan hidup manusia.

Dari uraian di atas tujuan Pendidikan Agama peneliti sesuaikan dengan tujuan Pendidikan Agama di lembaga-lembaga pendidikan formal dan peneliti membagi tujuan Pendidikan Agama itu menjadi dua bagian dengan uraian sebagai berikut :

1) Tujuan Umum

Tujuan umum Pendidikan Agama Islam adalah untuk mencapai kwalitas yang disebutkan oleh al-Qur'an dan hadits sedangkan fungsi pendidikan nasional adalah mengembangkan kemampuan dan membentuk watak serta peradaban bangsa yang bermartabat dalam rangka mencerdaskan kehidupan bangsa, bertujuan untuk mengembangkan potensi peserta didik agar menjadi manusia yang beriman dan bertakwa kepada Tuhan Yang Maha Esa, berakhlak mulia, sehat, berilmu, cakap, kreatif, mandiri, dan menjadi warga negara yang demokratis serta bertanggung jawab. Untuk mengemban fungsi tersebut pemerintah menyelenggarakan suatu sistem pendidikan nasional yang tercantum dalam Undang-Undang dasar No. 20 Tahun 2003

Dari tujuan umum pendidikan di atas berarti Pendidikan Agama bertugas untuk membimbing dan mengarahkan anak didik supaya menjadi muslim yang beriman teguh sebagai refleksi dari keimanan yang telah dibina oleh penanaman pengetahuan agama yang harus dicerminkan dengan akhlak yang mulia sebagai sasaran akhir dari Pendidikan Agama itu.

Menurut Abdul Fattah Jalal tujuan umum pendidikan Islam adalah terwujudnya manusia sebagai hambah Allah, ia mengatakan bahwa tujuan ini akan mewujudkan tujuan-tujuan khusus. Dengan mengutip surat at-Takwir ayat 27. Jalal menyatakan bahwa tujuan itu adalah untuk semua manusia. Jadi menurut Islam, pendidikan haruslah menjadikan seluruh manusia menjadi manusia yang menghambakan diri kepada Allah atau dengan kata lain beribadah kepada Allah.

Islam menghendaki agar manusia dididik supaya ia mampu merealisasikan tujuan hidupnya sebagaimana yang telah digariskan oleh Allah. Tujuan hidup manusia itu menurut Allah adalah beribadah kepada Allah, ini diketahui dari surat al-Dzariyat ayat 56 : “Dan aku tidak menciptakan jin dan manusia kecuali supaya mereka beribadah kepada-Ku” (Q.S al-Dzariyat, 56)

2) Tujuan Khusus

Tujuan khusus Pendidikan Agama adalah tujuan yang disesuaikan dengan pertumbuhan dan perkembangan anak sesuai dengan jenjang pendidikan yang dilaluinya, sehingga setiap tujuan Pendidikan Agama pada setiap jenjang sekolah mempunyai tujuan yang berbeda-beda, seperti tujuan Pendidikan Agama di sekolah dasar berbeda dengan tujuan Pendidikan Agama di SMP, SMA dan berbeda pula dengan tujuan Pendidikan Agama di perguruan tinggi.

Tujuan khusus pendidikan seperti di SLTP adalah untuk meningkatkan kecerdasan, pengetahuan, kepribadian, akhlak mulia, keterampilan untuk hidup mandiri dan mengikuti pendidikan lebih lanjut serta meningkatkan tata cara membaca al-Qur’an dan tajwid sampai kepada tata cara menerapkan hukum bacaan mad dan wakaf. Membiasakan perilaku terpuji seperti qanaah dan tasawuh dan menjawukan diri dari perilaku tercela seperti ananiah, hasad, ghadab dan namimah serta memahami dan meneladani tata cara mandi wajib dan shalat-shalat wajib maupun shalat sunat (Riyanto, 2006 : 160).

Sedangkan tujuan lain untuk menjadikan anak didik agar menjadi pemeluk agama yang aktif dan menjadi masyarakat atau warga negara yang baik dimana keduanya itu terpadu untuk mewujudkan apa yang dicita-citakan merupakan suatu hakekat, sehingga setiap pemeluk agama yang aktif secara otomatis akan menjadi warga negara yang baik, terciptalah warga negara yang pancasilis dengan sila Ketuhanan Yang Maha Esa

### **2.6.2 Macam Pendidikan Islam**

Berikut macam pendidikan islam yang akan disajikan di dalam game edukasi yang akan dibuat, antara lain :

1. Tauhid

Menurut Syekh Muhammad Abduh, tauhid adalah suatu ilmu yang membahas tentang wujud Allah, sifat-sifat yang wajib tetap pada-Nya, sifat-sifat yang boleh disifatkan pada-Nya, dan tentang sifat-sifat yang sama sekali wajib dilenyapkan pada-Nya. Juga membahas tentang rasul-rasul Allah, meyakinkan kerasulan mereka.

1. Aqidah & Akhlak

Menurut Ibnu Taimiyah dalam bukunya “Aqidah al-Wasithiyah” makna aqidah dengan suatu perkara yang harus dibenarkan dalam hati, dengannya jiwa yang menjadi tenang sehingga jiwa itu menjadi yakin serta mantap tidak dipengaruhi oleh keraguan dan tidak dipengaruhi oleh salah sangka. Sedangkan pengertian akhlak menurut Muhammad bin Ali Asy Syariif Al Jurjani adalah sesuatu sifat (baik atau buruk) yang tertanam kuat dalam diri yang darinya terlahir perbuatan-perbuatan dengan mudah dan ringan tanpa perlu berpikir dan merenung.

1. Alqur’an & Hadist

Menurut Dr. Subhi as-Salih Al Qur'an adalah kalam Allah swt merupakan mukjizat yang diturunkan kepada Nabi Muhammad saw ditulis dalam mushaf dan diriwayatkan dengan mutawatir serta membacanya adalah ibadah. Sedangkan Sunnah/hadits menurut istilah muhadditsin (ahli-ahli hadits) ialah segala yang dinukilkan dari nabi SAW, baik berupa perkataan, perbuatan, maupun taqrir, pengajaran, sifat, perilaku, perjalanan hidup nabi SAW, sebelum dan sesudah dianggak menjadi rasul, maupun sesudahnya.

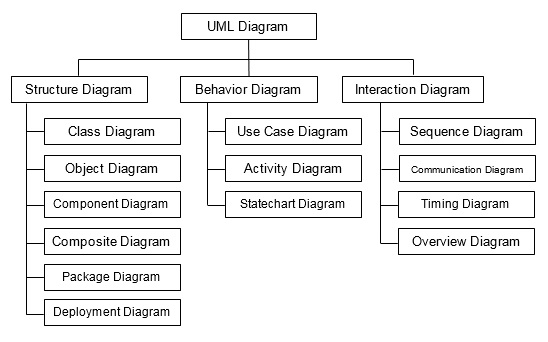
1. Sejarah Islam & Fiqih

Sejarah Islam adalah sejarah agama Islam mulai turunnya wahyu pertama pada tahun 610 yang diturunkan kepada rasul yang terakhir yaitu Muhammad bin Abdullah di Gua Hira, Arab Saudi sampai dengan sekarang. Sedangkan fiqih adalah salah satu bidang ilmu dalam syariat Islam yang secara khusus membahas persoalan hukum yang mengatur berbagai aspek kehidupan manusia, baik kehidupan pribadi, bermasyarakat maupun kehidupan manusia dengan Tuhannya.

## **2.7 *Unified Modelling Language* (UML)**

*Unified Modelling Language* (UML) adalah bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung (Sukamto dan Shalahudin. 2016). Dalam UML terdapat 13 diagram yang dikelompokkan dalam 3 kategori yaitu:

1. *Structure Diagrams*, yaitu kumpulan diagram-diagram yang menggambarkan struktur statis dari sistem yang dimodelkan.
2. *Behaviour Diagrams*, yaitu kumpulan diagram-diagram yang menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi di dalam sistem.
3. *Interaction Diagrams*, yaitu kumpulan diagram-diagram yang menggambarkan interaksi sistem dengan sistem lain ataupun interaksi antar subsistem dalam sebuah sistem.



**Gambar 2. 4** Jenis Kategori Diagram UML

### 2.7.1 *Use Case Diagram*

*Use Case* atau diagram *Use Case* merupakan pemodelan untuk kelakuan sistem informasi yang akan dibuat (Sukamto dan Shalahudin, 2016). *Use Case* mendeskripsikan interaksi antara satu atau lebih aktor dengan sistem informasi 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.

Berikut ini adalah simbol-simbol notasi yang digunakan pada tabel *Use Case* diagram(Sukamto dan Shalaludin, 2016)

**Tabel 2. 1** Deskripsi Notasi Pada *Use Case* Diagram

| **No** | **Notasi** | **Deskripsi** |
| --- | --- | --- |
| 1. | *Use Case* | Fungsi dari sistem yang berperan mengirimkan pesan antara unit-unit dan aktor. Biasanya dinyatakan dengan menggunakan kata kerja di |
| 2. | Aktor | Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat. Biasanya dinyatakan dengan menggunakan kata benda di nama aktor |

| **No** | **Notasi** | **Deskripsi** |
| --- | --- | --- |
| 3. | Asosiasi | Komunikasi antara aktor dan *Use Case*. |
| 4. | Extend | Relasi *Use Case* tambahan pada sebuah *Use Case*, mirip prinsip inheritance pada pemrograman berorientasi objek; Arah panah mengarah kepada *Use Case* yang ditambahkan. |
| 5. | Include | Relasi tambahan pada sebuah *Use Case* dimana *Use Case* yang ditambahkan membutuhkan *Use Case* tambahan ini untuk menjalankan fungsinya. Arah panah mengarah kepada *Use Case* yang ditambahkan dan *Use Case* ini akan selalu dijalankan bersamaan *Use Case* tambahan. |
| 6. | Generalisasi | Hubungan generalisasi dan spesialisasi antara 2 buah *Use Case* dimana fungsi *Use Case* turunan memiliki fungsi *Use Case* yang menjadi generalisasinya. Arah panah mengarah pada *Use Case* yang menjadi generalisasinya. |

### 2.7.2 *Class* Diagram

Sukamto dan Shalaludin (2016) menyatakan bahwa diagram kelas atau *Class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki atribut dan metode. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas. Metode atau operasi adalah fungsi-fungsi yang dimiliki oleh suatu kelas. Berikut ini adalah simbol-simbol notasi yang digunakan pada *Class* diagram dalam tabel 2.3 (Sukamto dan Shalaludin, 2016)

**Tabel 2.2** Deskripsi Notasi Pada *Class* Diagram

| **No** | **Notasi** | **Deskripsi** |
| --- | --- | --- |
| 1. | Kelas | Kelas pada struktur sistem. |

| **No** | **Notasi** | **Deskripsi** |
| --- | --- | --- |
| 2. | Antarmuka / Interface | Sama dengan konsep interface dalam pemrograman berorientasi objek. |
| 3. | Asosiasi | Relasi antar kelas dengan makna umum. |
| 4. | Asosiasi berarah | Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain. |
| 5. | Generalisasi | Relasi antar kelas dengan makna generalisasi-spesialisasi. |
| 6. | Ketergantungan | Relasi antar kelas dengan makna kebergantungan antar kelas. |

### 2.7.3 *Activity* Diagram

*Activity* diagram adalah suatu jenis diagram khusus dari *state chart* diagram yang merepresentasikan state-state dan transisi-transisi yang terjadi pada akhir operasi-operasi (Adi, 2009). *Activity* Diagram dapat menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana alir berawal, *decision* yang mungkin terjadi, dan akhir dari proses yang terjadi. Pembuatan *Activity* diagram bertujuan untuk memberikan representasi grafis dari aliran interaksi di dalam suatu skenario yang sifatnya spesifik (Pressman, 2010). Berikut deskripsi dan macam-macam simbol yang terdapat dalam *Activity* diagram akan dijelaskan dalam Tabel 2.3 (Pressman, 2010)

**Tabel 2.3** Deskripsi Notasi Pada *Activity* Diagram

| **No** | **Notasi** | **Deskripsi** |
| --- | --- | --- |
| 1. | *Activity* | Memperlihatkan bagaimana *Class* antarmuka berinteraksi satu sama lain. |
| 2. | Action | State dari sistem yang mencerminkan eksekusi dari suatu aksi. |
| 3. | Initial Node | Status awal aktivitas sistem. |
| 4. | *Activity* Final Node | Status akhir yang dilakukan sistem. Status akhir Menunjukkan akhir aktivitas dari sistem. |

| **No** | **Notasi** | **Deskripsi** |
| --- | --- | --- |
| 5. | Fork Node | Satu aliran pada tahap yang tertentu berubah menjadi beberapa aliran (aktivitas secara paralel). |
| 6. | Join Node | Menunjukkan aktivitas yang digabungkan. Beberapa aktivitas pada tahap tertentu berubah menjadi satu aliran aktivitas. |
| 7. | Signal | Signal menggambarkan pengirim dan penerima pesan dan aktivitas yang terjadi. |
| 8. | Decision | Menunjukkan aktivitas yang harus dipilih, Pilihan alur aktivitas digambarkan dengan inisial ’Y’ untuk ya atau inisial ’T’ atau "tidak". |

## 2.8 *Storyboard*

### 2.8.1 Pengertian Storyboard

*Storyboard* merupakan serangkaian sketsa (gambaran kartun) dibuat berbentuk persegi panjang yang menggambarkan suatu urutan (alur cerita) elemen-elemen yang diusulkan untuk aplikasi multimedia. *Storyboard* visualisasi ide dari aplikasi yang akan dibangun, sehingga dapat memberikan gambaran dari aplikasi yang akan dihasilkan. *Storyboard* dapat dikatakan juga visual *script* yang akan dijadikan *outline* dari sebuah proyek, ditampilkan *shot by shot* yang biasa disebut dengan istilah *scene* (Suyanto, 2003).

Sebuah *Storyboard* media interaktif dapat digunakan dalam antarmuka grafik pengguna untuk rancangan rencana desain sebuah *web*site atau proyek interaktif sebagaimana alat visual untuk perencanaan isi. Sebaliknya, sebuah *sitemap* (peta) atau *Storyboard* (diagram alur) dapat lebih bagus digunakan untuk merencanakan arsitektur informasi, navigasi, links, organisasi dan pengalaman pengguna, terutama urutan kejadian yang susah diramalkan atau pertukaran audiovisual kejadian menjadi kepentingan desain yang belum menyeluruh.

Salah satu keuntungan menggunakan *Storyboard* adalah dapat membuat pengguna untuk mengalami perubahan dalam alur cerita untuk memicu reaksi atau ketertarikan yang lebih dalam. Seorang pembuat *Storyboard* harus mampu menampilkan sebuah cerita yang bagus. Pada proyek tertentu, pembuat *Storyboard* memerlukan keterampilan menggambar yang bagus dan kemampuan beradaptasi terhadap desain yang telah dikeluarkan.

## 2.8.2 **Membuat *Storyboard***

Sebelum membuat *Storyboard*, disarankan untuk membuat cakupan *Storyboard* terlebih dahulu dalam bentuk rincian naskah yang kemudian akan dituangkan detail grafik dan visual untuk mempertegas dan memperjelas tema. Batasan produksi terakhir akan dijelaskan supaya sesuai dengan jenis produksi yang ditentukan, misalnya *Storyboard* akan digunakan untuk film, iklan, kartun ataupun video lain. Untuk mempermudah membuat proyek, maka harus dibuat sebuah rencana kasar sebagai dasar pelaksanaan.

Penggunaan *Storyboard* jelas akan mempermudah pelaksanaan dalam proses produksi nantinya. Format apapun yang dipilih untuk *Storyboard*, informasi berikut harus dicantumkan, yaitu :

1. Sketsa atau gambaran layar, halaman atau frame.
2. Warna, penempatan dan ukuran grafik, jika perlu
3. Teks asli, jika ditampilkan pada halaman atau layar
4. Warna, ukuran dan tipe font jika ada teks
5. Narasi, jika ada
6. Animasi jika ada
7. Video, jika ada
8. Audio, jika ada
9. Interaksi dengan penonton, jika ada
10. Hal-hal yang perlu diketahui oleh staf produksi

*Storyboard* tidak sulit untuk dibuat, bahkan cukup menggunakan poin-poin saja asalkan bisa memberikan penjelasan yang jelas mengenai desain tentang bagaimana materi yang diajarkan sudah lebih dari cukup. Cara membuatnya juga cukup dengan *software* pengolah kata maupun *spreadsheet*, tidak perlu menggunakan aplikasi pembuat *Storyboard* professional. Beberapa alasan mengapa menggunakan *Storyboard*:

1. *Storyboard* harus dibuat sebelum tim membuat animasi
2. *Storyboard* digunakan untuk mengingatkan animator
3. *Storyboard* dibuat untuk memudahkan membaca cerita

## 2.9 Android

Android merupakan sistem operasi berbasis Linux bagi telepon seluler seperti telepon pintar dan komputer tablet Safaat (2012:1). Android telah menyediakan platform terbuka yang ditujukan terutama kepada para pengembang yang ingin mengembangkan aplikasi dengan platform android. Yuniar (2012:3) mendefinisikan android sebagai perangkat lunak (*software*) sistem operasi yang memakai basis kode komputer yang dapat didistribusikan secara terbuka atau *open source* sehingga pengguna bisa membuat aplikasi baru di dalamnya. Pengembang aplikasi android dipermudah dengan banyaknya tools yang disediakan untuk mengembangkan aplikasi dengan platform ini. *Tools* yang disediakan mulai dari aplikasi yang digunakan untuk perancangan aplikasi android sampai tools untuk menjalankan aplikasi android tanpa menggunakan handphone android. Selain itu, android memiliki beberapa versi dalam perkembangannya hingga sekarang.

Beberapa sistem operasi android yang berkembang terus menerus dari awal kemunculannya hingga sekarang. Perkembangan versi android Menurut Jazi Eko (2013:6-12) adalah sebagai berikut:

1. Android versi 1.1 (Maret 2009)
2. Android versi 1.5 (Cupcake, Mei 2009)
3. Android versi 1.6 (Donut, September 2009)
4. Android versi 2.0/2.1 (Éclair, Desember 2009)
5. Android versi 2.2 (Froyo, Mei 2010)
6. Android versi 2.3 (Gingerbread, Desember 2010)
7. Android versi 3.0/3.1 (Honeycomb, 2011)
8. Android versi 4.0 (Ice Cream Sandwich, Oktober 2011)
9. Android versi 4.1 (Jelly Bean, Juni 2012)
10. Android versi 4.4 (KitKat, September 2013)
11. Android versi 5.0 (Lollipop, Juni 2014)
12. Android versi 6.0 (Marshmallow, 2015)
13. Android versi 7.0 (Nougat, Juli 2016)
14. Android versi 8.0 (Oreo, Agustus 2017)
15. Android versi 9.0 (Pie, Agustus 2018)

## 2.10 Apache Cordova

Telah disinggung sedikit sebelumnya bahwa salah satu *tools* untuk mengembangkan aplikasi *hybrid* yaitu dengan menggunakan *Phonegap*. *Phonegap* adalah salah satu *framework* yang *opensource* yang memungkinkan kita menjadikan aplikasi berbasis web menjadi aplikasi *native*. *Phonegap* adalah nama lama, sementara nama baru dari *phonegap* ini adalah *Apache Cordova* karena adanya akuisisi dan lain-lain, pengembangannya ditangani oleh ASP (*Apache Software Foundation*) dan berganti nama menjadi *Cordova*. Dengan *framework* ini *programmer* dapat mengembangkan suatu aplikasi *mobile* yang dapat berjalan di lebih dari satu platform dengan menggunakan *script* yang sama. Dengan *framework* ini pengembang tidak harus menguasai bahasa pemrograman Java untuk *platform Android* atau menguasai bahasa pemrograman *Objective-C* untuk *platform* iOS tetapi dengan bantuan *Apache Cordova* pengembang hanya wajib menguasai HTML5, CSS3 dan *Javascript*. Banyak tools *framework* pendukung *Cordova* ini yang memudahkan pengembang aplikasi dalam mengembangkan aplikasi *mobile*, seperti *Adobe Phonegap* dan *Ionic Framework* (Wahana, 2014)

## 2.11 *Construct 2*

*Construct 2* adalah sebuah program untuk membuat permainan komputer HTML5, tanpa mengharuskan pengguna untuk memiliki pengalaman pemrograman dan hanya menggunakan lingkungan pengembangan ”*drag and drop*” yang intuitif (Gullen, 2011). Program bisa digunakan dari antarmuka grafis tanpa harus menulis kode yang rumit. *Construct 2* digunakan untuk membuat permainan 2D, dan hadir dengan begitu banyak sumber daya yang membuatnya mudah, termasuk yang membuat benda pada *Game* dipengaruhi oleh gravitasi. *Construct 2* memiliki unsur-unsur seperti *sprite*, latar belakang, dan efek suara. Menambahkan berkas media dari luar aplikasi juga sangat mudah untuk dilakukan. Filosofi dari *Construct 2* adalah lingkungan visual yang mudah dan intuitif.

### 2.12 *Black Box Testing*

Roger S. Pressman (2010) menjelaskan bahwa, *black-box testing* berfokus pada persyaratan fungsional perangkat lunak yang memungkinkan *engineers* untuk memperoleh set kondisi *input* yang sepenuhnya akan melaksanakan persyaratan fungsional untuk sebuah program. *Black-box* *testing* berusaha untuk menemuka kesalahan dalam kategori berikut :

1. Fungsi yang tidak benar atau fungsi yang hilang
2. Kesalahan antarmuka
3. Kesalahan dalam struktur data atau akses *database* eksternal
4. Kesalahan perilaku *(behavior*) atau kesalahan kinerja
5. Inisialisasi dan pemutusan kesalahan

Meskipun dirancang untuk mengungkap kesalahan, pengujian *black-box* digunakan untuk memperlihatkan bahwa fungsi-fungsi perangkat lunak dapat beroperasi, bahwa input diterima dengan baik dan output dihasilkan dengan tepat dan integritas informasi eksternal (seperti *file* data) dipelihara. Adapun Teknik pengujian *black-box* menurut Rosa Ariani Sukamanto (2009) sebagai berikut :

1. *Boundary value Analist* (BVA) *Limit Testing* adalah banyak kesalahan terjadi pada kesalahan masukan. BVA mengijinkan untuk menyeleksi kasus uji yang menguji Batasan nilai input. BVA merupakan komplemen dari *equivalance partitioning*. Lebih pada memilih elemen-elemen di dalam kelas ekuivalen pada bagian sisi batas dari kelas.
2. *Comparison testing* adalah pengujian pada beberapa aplikasi *reliability* dari sebuah perangkat lunak. Redundansi perangkat keras dan perangkat lunak mungkin digunakan untuk meminimalisir kesalahan (*error*). Untuk redundasi perangkat lunak, gunakan tim yang terpisah untuk mengembangkan setiap versi dengan data yang sama untuk memastikan semua versi menghasilkan keluaran yang sama. Jalankan semua versi dengan *parallel* dan perbandingan keluaran secara *real-time*.
3. *Sample testing* adalah pengujian yang melibatkan beberapa nilai yang terpilih dari sebuah kelas *ekuivalen*. Mengintegrasikan nilai pada kasus uji dan nilai-nilai yang terpilih, bisa dipilih dengan urutan tertentu atau *interval* tertentu.
4. *Robustness testing* adalah pengujian dengan data input dipilih diluar spesifikasi yang telah didefiniskan. Tujuan dari pengujian ini adalah membuktikan bahwa tidak ada kesalahan jika masukan tidak valid.
5. *Behavior testing* (menguji perilaku sistem apakah sudah sesuai dengan permintaan) menguji tingkah laku sistem berdasarkan polanya “didesain sebagaimana mulanya”. Selain itu merupakan pengujian dengan hasil uji tidak dapat dievaluasi jika hanya melakukan pengujian sekali, tapi dapat dievaluasi jika pengujian dilakukan beberapa kali, misalnya pada pengujian struktur data.
6. *Requirement Testing* adalah spesifikasi kebutuhan yang terasosiasi dengan perangkat lunak *(input, output*, fungsi, performasi) diidentifikasikan pada tahap spesifikasi kebutuhan dan desain. *Requirement testing* melibatkan pembuatan kasus uji untuk setiap spesifikasi kebutuhan yang terkait dengan program,
7. *Perfomance testing* (menguji kinerja sistem) menguji software bagian dari sistem/orientasi kepada *hardware.* Selain itu mengevaluasi kemampuan program untuk beroperasi dengan benar dipandang dari sisi acuan kebutuhan, misalnya aliran data, ukuran pemakaian memori, kecepatan eksekusi dan lain-lain. Untuk mencari tahu beban kerja atau kondisi konfigurasi program dan dapat digunakan untuk menguji batasan lingkungan program.
8. *Endurance testing* (menguji daya tahan sistem) adalah melibatkan kasus uji yang diulang-ulang dengan jumlah tertentu dengan tujuan untuk mengevaluasi program apakah sesuai dengan spesifikasi kebutuhan.
9. *Equivalence Partitioning* adalah membagi input menjadi kelas-kelas data yang dapat digunakan untuk menggenerasi kasus uji dengan tujuan untuk menemukan kelas-kelas kesalahan. Selain itu, *equivalence partitioning* berdasarkan pada kesamaan kelas-kelas kondisi *input.* Sebuah kelas yang ekuivalen merepresentasikan kumpulan status/kondisi yang *valid* atau tidak *valid*. Sebuah kondisi input dapat berupa nilai *numerik* yang spesifik, rentang nilai, kumpulan nilai yang berkaitan atau kondisi *Boolean*.
10. *Cause-Effect Relationship Testing* (menguji sebab-akibat) adalah Teknik yang merupakan suplemen dari equivalence testing dengan menyediakan cara untuk memilih kombinasi data input dan melibatkan kondisi input (*cause*) dan kondisi output (*effect*) untuk mencegah pendifinisian kasus uji yang terlalu banyak.
11. *Accptance testing* adalah pengujian terakhir sebelum sistem dipakai oleh user yang melibatkan pengujian dengan data dari pengguna sistem. *User acceptance* test (UAT) adalah uji terima perangkat lunak yang dilakukan ditempat pengguna oleh user perangkat lunak tersebut. Tujuan pengujian ini adalah untuk menguji apakah sistem sudah sesuai dengan apa yang tertuang dalam spesifikasi fungsional sistem (*validation*).

## 2.13 Skala Likert

### 2.13.1 Pengertian Skala Likert

Skala Likert merupakan metode pengukuran yang digunakan untuk mengukur sikap, pendapat, dan persepsi sesorang atau kelompok orang tentang fenomena sosial (Sugiyono, 2012).

### 2.13.2 Penentuan Skor Jawaban

Skor jawaban merupakan nilai jawaban yang akan diberikan oleh responden, hal pertama yang harus kita lakukan adalah menentukan skor dari tiap jawaban yang akan diberikan. Contohnya, sikap yang akan kita pakai yaitu” setuju”. Selanjutnya kita menentukan banyaknya jawaban pada tiap pernyataan yang akan kita berikan. Misalnya 5 skala, berarti sangat tidak setuju, kurang setuju, cukup. Jika pertanyaan yang diberikan bersifat susah untuk diberikan jawaban, otomatis responden cenderung statik oleh karena itu kita dapat memberikan pilihan jawaban yang banyak, misal 7 atau 9 jawaban dari setiap pertanyaan. Hal ini bertujuan agar responden dapat memberikan penilaian sesuai dengan kriteria mereka berdasarkan pilihan yang ada. Skala nilai dapat dilihat pada Tabel 2.5 (Sugiyono, 2012).

**Tabel 2. 4** Skala Nilai Likert

| **Skala Jawaban** | **Nilai** |
| --- | --- |
| Sangat Tidak Setuju/Suka/Bagus | 1 |
| Kurang Setuju/Suka/Bagus | 2 |
| Cukup Setuju/Suka/Bagus | 3 |
| Setuju/Suka/Bagus | 4 |
| Sangat Setuju/Suka/Bagus | 5 |

### 2.13.3 Skor Ideal

Skor ideal merupakan skor yang digunakan untuk menghitung skor untuk menemukan *rating scale* dan jumlah seluruh jawaban. Untuk menghitung jumlah skor ideal (kriterium) dari seluruh *item* digunakan rumus berikut, yaitu: Skor Kriterium = Nilai skala x Jumlah responden. Seandainya skor tertinggi adalah 5 dan jumlah responden 20, maka dapat dirumuskan menjadi:

**Tabel 2. 5** Rumus Skala Likert

| **Rumus** | **Skala** |
| --- | --- |
| 5x20=100 | SB |
| 4x20=80 | B |
| 3x20=60 | CB |
| 2x20=40 | KB |
| 1x2=20 | SKB |

Selanjutnya semua jawaban responden dijumlahkan dan dimasukkan ke dalam *rating scale* dan ditentukan daerah jawabannya.

### 2.13.4 Raiting Scale Skala Likert

Rating scale berfungsi untuk mengetahui hasil data angket (kuesioner) dan wawancara secara umum dan keseluruhan yang didapat dari penilaian angket (kuesioner) dan wawancara. Dengan ketentuan seperti tabel 2.6

**Tabel 2. 6** Hasil Penelitian Skala Likert

| **Rumus** | **Skala** |
| --- | --- |
| 81-100 | SB |
| 61-80 | B |
| 41-60 | CB |
| 21-40 | KB |
| 0-20 | SKB |

### 

### 2.13.5 Persentase Persetujuan

Untuk mengetahui jumlah jawaban dari para responden melalui persentase, yaitu digunakan rumus sebagai berikut: p=f/n x100%. Keterangan dari rumus tersebut adalah p:persentase, f:frekuensi, n:jumlah skor ideal (Sugiyono, 2012).

## 2.14 Penelitian Terkait

Berikut merupakan beberapa penelitian sebelumnya yang digunakan sebagai pembanding dalam mengimplementasikan metode *Finite State Machine* dalam pembuatan *game* yang dapat dilihat pada Tabel 2.7 berikut ini.

**Tabel 2.7** Kajian Terkait

| No | Penulis | Judul | Keterangan |
| --- | --- | --- | --- |
| 1 | Chakim Annubaha | Implementasi *Finite State Machine* (FSM) Pada *Game* Edukasi Bahasa Arab | Implementasi FSM yaitu pada saat menjawab pertanyaan dan tampilan pemain saat mencapat misi tertentu. |
| 2 | Rostianingsih, Gregorius, & Wijaya | *Game* Simulasi *Finite State Machine* untuk Pertanian dan Peternakan | Implementasi FSM pada  hubungan player dan warga, kondisi tanaman, waktu tidur. |
| 3 | M Murti Hanafi | Tukarmatika Berbasis Android Menggunakan Metode *Finite State Machine* | Implementasi FSM pada saat menjawab pertanyaan, player menolong burung. |
| 4 | Nasyiya Ulfa | Penerapan *Metode Finite State Machine* Pada *Game Adventure* The Muslim | Implementasi FSM pada NPC Friendly (Karakter Pemberi Pertanyaan) & NPC Non Friendly (Musuh) |

Pada skripsi berjudul Implementasi Finite State Machine (FSM) Pada Game Edukasi Bahasa Arab oleh Chakim Annubaha (Chakim Annubaha, 2014), permainan bersifat Single Player bergenre adventure. Player harus dapat menjawab pertanyaan yang diberikan. Terdapat 3 stage pada game ini. Setiap stage memiliki tujuan berberda yang harus dicapai oleh player. Dalam game ini juga terdapat NPC yang akan memberikan petunjuk dalam melakukan suatu hal kepada player selama memainkan permainan ini. Metode Finite State Machine (FSM) yang diimplementasikan di dalamnya yaitu dalam menjawab pertanyaan - pertanyaan yang diberikan kepada player tersebut. FSM animasi *player* tergantung kepada hasil pertanyaan yang dijawab. Penerapan metode FSM juga ada pada beberapa fitur lain seperti ketika uang *player* mencapai jumlah tertentu maka baju akan berubah. Selanjutnya pada *game* buatan Rostianingsih Silvia dan kawan-kawan yang berjudul Game Simulasi Finite State Machine untuk Pertanian dan Peternakan (Rostianingsih, Gregorius, & Wijaya, 2013) adalah merupakan permainan berbasis simulasi dengan setting pedesaan. Tujuan perancangannya adalah untuk menerapkan efek simulasi pertanian dan peternakan sehingga permainan berjalan tidak membosankan dan dapat dimainkan secara terus menerus. Pemain akan mengendalikan satu karakter utama yang dapat berinteraksi dengan penduduk di desa. Pemain dapat menanam, beternak, serta membangun relasi dengan warga yang berada di desa. Metode FSM yang diimplementasikan pada *game* ini terdapat dalam berbagai fitur seperti ketika hubungan player dengan warga desa baik, maka akan terjadi sesuatu kepada warga dan player tersebut. Contoh lain yaitu pada menyiram tanaman, apabila tidak terpenuhi suatu parameter tertentu dalam waktu yang telah ditentukan maka tanaman akan mati. Begitu juga dengan waktu tidur, ketika player tidak tidur dengan cukup maka akan terjadi konsekuensi pada permainan. Berbeda dengan game Tukarmatika Berbasis Android Menggunakan Metode *Finite State Machine* oleh Hanaf M Murti, *game* ini menggabungkan metode FSM dengan algoritma A dalam pembuatannya (Murti, 2016). *Game* dimainkan oleh satu pemain saja dan bergenre puzzle,. Dalam game tukarmatika ini, *player* diharuskan untuk menjawab beberapa pertanyaan yang diberikan. Pertanyaan berupa soal matematika. *Game* ini sendiri ditujukan bukan kepada anak-anak melainkan pemain yang lebih tua. Tidak hanya itu, pada *game* tukarmatika ini juga terdapat musuh yang akan mengganggu pemain dalam menyelesaikan permainan. Soal yang diberikan dalam *game* ini sendiri memiliki beberapa tingkatan dari mudah hingga sulit*. Game* bertujuan menjadikan pemain mahir menguasai materi bentuk akar. Metode FSM diterapkan pada saat menampilkan serta menjawab pertanyaan. Metode FSM juga diimplementasikan pada karakter Andi ketika menolong burung yang sedang terjebak pada ranting ataupun akar pohon agar menampilkan pertanyaan.

# BAB III Metodologi Penelitian

## 3.1 Metodologi Penelitian

### 3.1.1 Pengumpulan Data/Informasi

Bahan penelitian berupa teori perancangan game yang diperoleh dari referensi buku dan internet. Data teori perancangan game yang diperlukan terdiri dari macam - macam pendidikan dasar Islam, teori *Game* edukasi*, Storyboard*, UML, dan metode pengembangan yaitu *Finite State Machine.*

## 3.2 Alat yang dipergunakan

### 3.2.1 Alat Penelitian

Alat penelitian yang digunakan dalam penelitian ini adalah:

1. *Unified Modelling Language* (UML), digunakan untuk menggambarkan batasan sistem dan fungsi-fungsi sistem secara umum, menggambarkan proses yang dilaksanakan, menggambarkan representasi struktur sistem, diagram yang akan digunakan dalam pengembangan sistem ini adalah *Use Case* diagram, *Activity* diagram*,* dan *Class* diagram.
2. *Storyboard,* digunakan untuk memvisualisasikan aplikasi yang akan dibangun.

### 3.2.2 Perangkat Keras

Laptop Merk Asus 14 X455LF dengan spesifikasi sebagai berikut :

1. Prosesor Inter(R) Core(TM) i3-5005U CPU @2.00GHz (4CPUs), ~2.0GHz

2. Memori RAM 4GB

3. Storage Harddisk 500GB

4. Windows 10 Pro 64-bit

### 3.2.3 Perangkat Lunak

Perangkat lunak yang digunakan dalam penelitian ini:

* + - * 1. Sistem Operasi Windows 10 *Unlimate enterprise* 64-bit, digunakan untuk mengoperasikan Komputer.
        2. Photopea, digunakan untuk mengedit gambar objek 2D pada *Game*.
        3. *Construct 2*, digunakan *Enginee* utama untuk merancang *Game* dengan bahasa HTML5.
        4. *Adobe Phonegap*, digunakan untuk mengkonversi hasil *export* construct 2 dari *cardova project* menjadi Android Apk
        5. Dan hasil *export* construct 2 dari *cardova* menjdi nw.js

## 3.3 Langkah penelitian

Langkah-langkah penelitian yang akan dilakukan adalah sebagai berikut:

1. Studi Literatur

Studi literatur dilakukan dengan mencari referensi ilmu pengetahuan dengan penelitian seperti mendalami tentang ilmu dasar islam, *Game* edukasi, UML (*Unified Modeling Language*), *Construct* 2, *Photopea, Storyboard* serta pemrograman *Android* maupun melalui penelitian-penelitian sebelumnya. Referensi yang didapat untuk mendukung penelitian ini berasal dari jurnal.

1. Analisis Kebutuhan

Analisis dilakukan dengan mengidentifikasi masalah serta kebutuhan yang berkenan dengan sistem yang akan dibuat, antara lain kebutuhan data, kebutuhan user, kebutuhan sistem dan kebutuhan fungsional.

1. Pengumpulan Data

Pengumpulan data penunjang penelitian yaitu data-data materi yang berkaitan dengan pertanyaan yang akan disajikan di dalam *game*/permainan.

1. Perancangan Aplikasi

Aplikasi ini menggunakan metode finite state machine yang diterapkan pada NPC/Non Playable Character. Perancangan aplikasi dilakukan dengan menggunakan UML (*Unified Modelling Language*) dengan tahapan antara lain, perancangan *Use Case* diagram, perancangan *Activity* diagram dan perancangan *Class* diagram. Kemudian dilanjutkan dengan perancangan antarmuka sistem, perancangan aset *Game* serta perancangan *Storyboard.*

1. Implementasi Aplikasi

Implementasi aplikasi yang dilakukan berdasarkan kepada desain dan perancangan aplikasi yang telah dilakukan pada tahapan sebelumnya. Pada tahapan ini, aplikasi akan dibuat dengan *enginee* *construct 2* menjadi program dengan bahasa HTML5. Kemudian dikonversi menjadi aplikasi *desktop* dan *android* melalui *export* *card ova* kemudian di *build* melalui *Adobe PhoneGap.*

1. Pengujian Aplikasi

Setelah proses penulisan kode pemrograman selesai, langkah berikutnya berupa pengujian atau testing. Pengujian sistem ini bertujuan untuk memastikan bahwa elemen komponen-komponen dari sistem sudah sesuai.

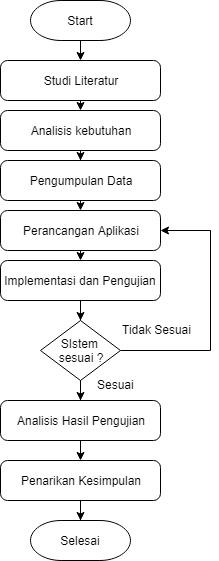
1. Analisis Hasil Pengujian Sistem

Analisis dilakukan dalam tahap pengujian dan implementasi untuk mengidentifikasi jika terdapat kesalahan dalam sistem.

1. Penarikan Kesimpulan

Kesimpulan dirumuskan berdasarkan analisis hasil pengujian sistem yang telah dilakukan.

Adapun diagram alir langkah-langkah penelitian yang akan dilakukan dapat dilihat pada **gambar 3.1** seperti berikut:



**Gambar 3. 1** Langkah-Langkah Penelitian

## 3.4 Analisis Kebutuhan Aplikasi

Tahap analisis kebutuhan merupakan tahap menganalisis kebutuhan dari perangkat lunak yang dikembangkan. Analisis kebutuhan dilakukan dengan tujuan untuk memberikan kemudahan. Analisis kebutuhan dilakukan dengan tujuan untuk memberikan kemudahan dalam pengembangan dan pengoperasian perangkat lunak oleh pengguna. Kebutuhan yang dianalisis antara lain adalah kebutuhan data, kebutuhan user, dan kebutuhan sistem. Data utama pada perangkat lunak *Game* edukasi ini yaitu materi yang diambil dari buku pelajaran kelas 3-5 SD yang berhubungan dengan pertanyaan-pertanyaan yang akan muncul pada setiap stage yaitu Tauhid, Aqidah & Akhlak, Al-Qur’an & Hadist serta Sejarah Islam dan Fiqih.

Kebutuhan selanjutnya adalah kebutuhan *user*. Kebutuhan *user* terdiri atas *user interface* dan fitur-fitur dalam *Game*. Aplikasi *Game* The Muslim ini ditujukan untuk anak usia 9-12 tahun, animasi pada *Game* sebisa mungkin disesuaikan dengan karakteristik anak yang menyukai warna-warni, animasi pada *Game* sebisa mungkin menarik dan interaktif yang disenangi untuk anak-anak.

Kebutuhan sistem dalam pengembangan *Game* ini dibuat dengan konsep bernuansa animasi. Perangkat lunak yang digunakan dalam pengembangan *Game* ini adalah Construct 2. Aplikasi ini nantinya akan digunakan *desktop* dan *platform* android *build* melalui dengan *framework* *PhoneGap*.

Spesifikasi kebutuhan sistem adalah kebutuhan yang berisi proses apa saja yang nantinya akan dilakukan oleh sistem-sistem yang ada saat ini maka dapat ditentukan spesifikasi kebutuhan yang diperlukan pada *Game* edukasi. Spesifikasi kebutuhan *Game* edukasi The Muslim dirangkum *functional Requirement* (FR) sehingga kebutuhan-kebutuhan yang harus di penuhi dapat dilihat secara jelas. FR *Game* ini dapat dilihat pada tabel 3.1

**Tabel 3. 1** Spesifikasi Kebutuhan

| **NO** | **KODE** | **DESKRIPSI** |
| --- | --- | --- |
| 1 | FR-001 | Sistem dapat masuk aplikasi |
| 2 | FR-002 | Sistem dapat menampilkan *Menu* utama aplikasi |
| 3 | FR-003 | Sistem dapat menampilkan *Menu* petunjuk |
| 4 | FR-004 | Sistem dapat menampilkan *Menu Level Select* |
| **NO** | **KODE** | **DESKRIPSI** |
| 5 | FR-005 | Sistem dapat menampilkan level 1 |
| 6 | FR-006 | Sistem dapat menampilkan level 2 |
| 7 | FR-007 | Sistem dapat menampilkan level 3 |
| 8 | FR-008 | Sistem dapat menampilkan level 4 |
| 9 | FR-009 | Sistem dapat menampilkan pertanyaan |
| 10 | FR-010 | Sistem dapat menampilkan tampilan *game over* |
| 11 | FR-011 | Sistem dapat menampilkan tampilan *level complete* |
| 12 | FR-012 | Sistem dapat keluar aplikasi |

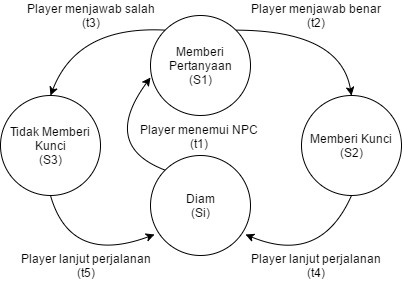
## 3.5 Perancangan *Finite State Machine* NPC *(Non Playable Charackter)*

*Finite State Machine* (FSM) adalah salah satu bidang *Artificial Intelligency* (AI) yang bisa diaplikasikan pada game AI. Prinsip kerjanya dengan menggunakan tiga hal berikut: *state* (keadaan), *event* (kejadian) dan *action* (aksi). Pada satu saat pada periode yang signifikan.

*Finite State Machine* dalam penelitian ini diimplementasikan kepada NPC yaitu terbagi menjadi 2 yaitu NPC yang bersifat *Friendly* dan NPC yang bersifat *Non-friendly*. NPC yang bersifat *Friendly* yaitu karakter pemberi pertanyaan sedangkan NPC yang bersifat *Non-friedly* adalah musuh yang terdiri dari 3 macam dengan karakteristik yang berbeda-beda.

**3.5.1 Skenario *Finite State Machine* Pada NPC *Friendly***

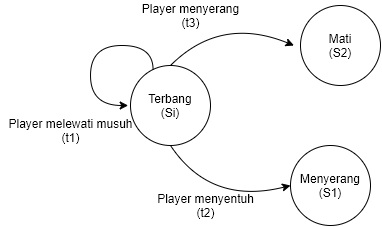
Implementasi FSM pada NPC *Friendly* atau karakter pemberi pertanyaan yaitu ketika Player berhasil menjawab pertanyaan yang diberikan dengan benar maka NPC *Friendly* akan memberikan kunci kepada player yang akan digunakan untuk membuka pintu dan melanjutkan ke stage selanjutnya. Berikut penerapan metode Finite State Machine pada NPC *Friendly* yang dapat dilihat pada Gambar 3.2 berikut :



**Gambar 3. 2** Alur FSM pada Karakter Pemberi Pertanyaan

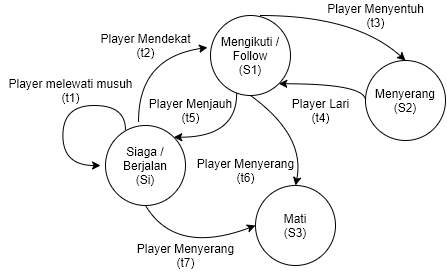
Pada gambar 3.2 terdapat 4 state {Si, S1, S2, S3} yang mungkin terjadi, dengan kondisi yaitu {t1, t2, t3, t4, t5} setiap *state*-nya dapat berpindah state jika kondisi terpenuhi. Si merupakan initial state atau disebut *state* awal dimana Si merupakan kondisi karakter pemberi pertanyaan/ NPC diam, Si dapat berpindah ke S1 apabila kondisi t1 terpenuhi yaitu *Player* menemui NPC dimana NPC nantinya akan memberikan pertanyaan kepada *player.* Apabila *player* menjawab pertanyaan dengan benar (t2) maka *state* akan berpindah ke S2 yaitu NPC memberikan kunci untuk membuka pintu ke level selanjutnya, setelah itu *player* akan melanjutkan perjalanan (t4) dan NPC kembali diam (Si). Jika *player* menjawab salah (t3) maka state akan berpindah ke S3 yaitu NPC tidak memberikan kunci, setelah itu *player* akan melanjutkan perjalanan (t5) dan NPC kembali diam (Si). Setiap *leve*l terdiri dari 5 karakter NPC pemberi pertanyaan. Pintu dapat dibuka apabila kunci yang dikumpulkan berjumlah 4 atau 5.

**3.5.2 Skenario *Finite State Machine* Pada NPC *Non-Friendly***

Implementasi FSM pada NPC *Non-Friendly* atau musuh yaitu pada saat player menyerang musuh yang akan muncul pada masing-masing stage yang terdiri dari 3 jenis musuh yaitu lebah, tikus dan siput. Musuh yang pertama digambarkan dengan karakter lebah, merupakan musuh yang terbang ke kiri dan ke kanan, musuh dapat mati apabila *player* menyerang dengan melompat ke atas badan musuh tersebut. Musuh yang kedua digambarkan dengan karakter tikus, merupakan musuh yang berjalan ke kiri dan kanan dan akan mengejar/mengikuti pemain apabila berada di dekatnya, musuh akan mati apabila player menyererang dengan melompat ke atas badan musuh tersebut. Musuh yang ketiga digambarkan dengan karakter siput yang berjalan ke kiri dan ke kanan dan dapat mati apabila player menyerang dengan melompat ke atas badan musuh sebanyak 3 kali. Berikut penerapan metode Finite State Machine pada NPC *Non-Friendly* yang dapat dilihat pada Gambar berikut :

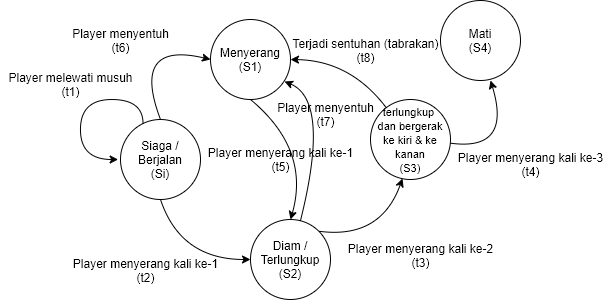
**Gambar 3.3** Alur FSM pada Musuh Lebah

Pada gambar 3.3 terdapat 3 state {Si, S1, S2} yang mungkin terjadi, dengan kondisi yaitu {t1, t2,t3} setiap state-nya dapat berpindah state jika kondisi terpenuhi. Si atau *initial state* pada alur FSM di atas yaitu Musuh Lebah terbang ke kiri dan ke kanan, apabila t1 terjadi yaitu *Player* melewati musuh maka musuh akan tetap berada pada kondisi awal yaitu musuh terbang. Apabila kondisi t2 terpenuhi yaitu player menyentuh musuh maka Si akan berpindah *state* ke S1 yaitu musuh menyerang dan apabila t3 terjadi yaitu player menyerang musuh maka Si akan berpindah *state* ke S2 yaitu musuh mati.



**Gambar 3.4** Alur FSM pada Musuh Tikus

Pada gambar 3.4 terdapat 4 *state* {Si, S1, S2, S3} yang mungkin terjadi, dengan kondisi yaitu {t1, t2,t3,t4,t5,t6,t7} setiap state-nya dapat berpindah *state* jika kondisi terpenuhi. Si atau kondisi awal yaitu musuh siaga / berjalan. Apabila *player* melewati musuh (t1) maka musuh akan tetap berapa pada kondisi awal yaitu siaga/berjalan. Apabila player mendekat (t2) maka *state* akan berpindah dari Si ke S1 yaitu Musuh mengikuti player, dan apabila *player* menyentuh musuh (t3) maka *state* akan berpindah dari S1 ke S2 yaitu Musuh menyerang *player*. Selanjutnya pada saat musuh menyerang player dan player lari (t4) maka state berpindah dari S2 ke S1 yaitu musuh kembali mengikuti player, dan apabila player menjauh (t5) maka musuh akan kembali siaga/berjalan(Si). Selanjutnya S1 dapat berpindah ke state S3 apabila *player* menyerang (t6) maka musuh akan mati, sedangakan Si dapat berpindah *state* ke S3 apabila *player* menyerang (t7) maka musuh akan mati.

****

**Gambar 3.5** Alur FSM pada Musuh Siput

Pada gambar 3.5 terdapat 5 state {Si, S1, S2, S3, S4} yang mungkin terjadi, dengan kondisi yaitu {t1, t2,t3,t4,t5,t6,t7,t8} setiap *state*-nya dapat berpindah state jika kondisi terpenuhi. Si atau kondisi awal yaitu musuh siaga / berjalan. Apabila terjadi masukan t1/ player melewati musuh maka musuh akan tetap berada pada kondisi awal (initial state) yaitu musuh siaga/ berjalan. Si dapat berpindah ke S1 yaitu musuh menyerang apabila kondisi t6 terpenuhi yaitu *player* menyentuh musuh. S1 dapat berpindah *state* ke S2 yaitu musuh diam / terlungkup apabila kondisi t5 terpenuhi yaitu player menyerang musuh kali ke-1. Si dapat berpindah state ke S2 apabila kondisi t2 terpenuhi yaitu *player* menyerang musuh kali ke-1, jadi dapat disimpulkan bahwa state S2 dapat aktif apabila terjadi masukan t2 atau t5 yaitu *player* menyerang kali ke-1 maka musuh akan diam/terlungkup. S2 dapat berpindah *state* ke S3 yaitu musuh terlungkup dan bergerak ke kiri & kanan apabila kondisi t3 terpenuhi yaitu *player* menyerang musuh kali ke-2 maka musuh akan bergerak ke kiri dan ke kanan. Dari S3 dapat berpindah *state* ke S4 yaitu musuh mati apabila kondisi t4 terpenuhi yaitu player menyerang musuh kali ke-3 maka musuh akan mati. S2 dapat berpindah state ke S1 apabila kondisi t7 terpenuhi yaitu *player* menyentuh musuh maka musuh akan menyerang *player* sama halnya dengan S3 dapat berpindah state ke S1 apabila kondisi t8 terpenuhi yaitu terjadi sentuhan/tabrakan maka musuh akan menyerang *player*, jadi dapat disimpulkan bahwa state S1 akan aktif apabila terjadi salah satu kondisi yaitu (t6,t7,t8)

**3.6 Perancangan Sistem**

Dalam melakukan perancangan sistem terdapat beberapa tahapan yang dilakukan yaitu:

1. Perancangan *Unified Modelling Language* (UML)

UML (*Unified Modelling Language*) adalah sebuah bahasa yang berdasarkan grafik atau gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasi dari sebuah sistem pengembangan *software* berbasis OO (*object-Oriented*). Diagram UML yang digunakan dalam merancang aplikasi ini adalah *Use Case* Diagram, *Activity* Diagram dan *Class* Diagram*.*

1. Perancangan *Flowchart* dan Struktur Navigasi

Tahapan ini menggambarkan sistem proses kerja pada dalam menjalankan aplikasi dan perancangan struktur navigasi pada aplikasi serta perancangan *scene* dan komponen antarmuka yang akan dibuat.

1. *Storyboard*

Pada tahapan ini dibuat sebuah *visual script* yang akan dijadikan *outline* dari sebuah proyek, ditampilkan *shot by shot* yang biasa disebut dengan istilah *scene.*

## 3.7 Peancangan *Unified Modeling Language* (UML)

Aspek yang dikembangkan meliputi perancangan *Unified Modelling Languag*e (UML) berupa *Use Case diagram*, *Activity diagram* dan *Class diagram*.

### 3.7.1 *Use Case Diagram*

*Use Case* mempresentasikan sebuah interaksi antara aktor terlibat dalam sistem atau aplikasi. Dalam aplikasi ini *user* dapat melakukan interaksi yaitu memilih *Menu* seperti, *Menu* melihat petujuk, Menu memilih play, Menu memilih level, Menu bermain , game over, dan menutup aplikasi adalah sebagai berikut :

1. *Use Case* Aktor

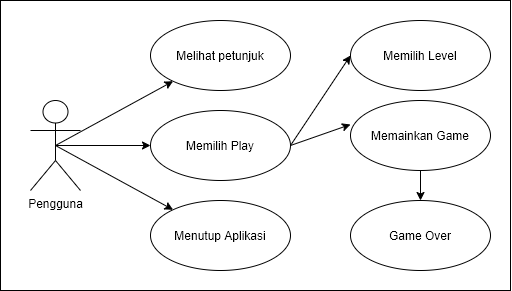
**Tabel 3. 2** *Use Case* Aktor

| **No.** | **Aktor** | **Deskripsi** |
| --- | --- | --- |
| 1. | User | *User* merupakan aktor dari perangkat lunak, *user* dapat memulai aplikasi, memilih *Menu* seperti, *Menu* melihat petujuk, *Menu* memilih play, *Menu* memilih *level*, *Menu* |
| **No.** | **Aktor** | **Deskripsi** |
|  |  | bermain, *game over*, dan menutup aplikasi. |

1. Definisi *Use Case*

**Tabel 3. 3** Definisi *Use Case*

| **No** | **Kode *Use Case*** | ***Use Case*** | **Deskripsi** |
| --- | --- | --- | --- |
| 1 | UC-01 | Melihat Petunjuk | Merupakan fitur untuk melihat panduan petunjuk bermain |
| 2. | UC-02 | Memilih *Play* | Merupakan fitur untuk memulai bermain |
| 2. | UC-03 | Memilih Level | Merupakan fitur memilih level permainan/level 1 |
| 3. | UC-04 | Memainkan game | Merupakan fitur untuk bermain level permainan |
| 4. | UC-05 | Game Over | Merupakan fitur ketika permainan berakhir |
| 5. | UC-06 | Menutup Aplikasi | Merupakan fitur keluar dari aplikasi |

Aktor dan *Use Case* yang telah didefinisikan pada Tabel 3.3 kemudian digambarkan menggunakan simbol aktor, *Use Case*, dan relasi antara aktor dengan *Use Case*. *Use Case* *Game* edukasi The Muslim dilihat pada gambar 3.6 

**Gambar 3. 6** *Use Case* *Game* Edukasi The Muslim

1. Skenario *Use Case*

**Tabel 3.4** Skenario *Use Case* Melihat Petunjuk

| **Nama Use Case** | **UC-01** | **Melihat Petunjuk** |
| --- | --- | --- |
| Pre Kondisi | Pemain berada di Menu utama | |
| **Aksi** | **Aktor** | **Sistem** |
|  | 1. Pemain memilih tombol icon petunjuk | 2. Masuk ke Menu petunjuk aplikasi |
| **Post Kondisi** | Menampilkan informasi tentang cara bermain *game* | |

**Tabel 3.5** Skenario*Use Case* Memilih *Play*

| **Nama Use Case** | **UC-02** | **Memilih *Play*** |
| --- | --- | --- |
| **Pre Kondisi** | Pemain berada di Menu utama | |
| **Aksi** | **Aktor** | **Sistem** |
|  | 1. Pemain memilih tombol *play* | 2. Sistem menampilkan pilihan level 1-4 |
| **Post Kondisi** | Pemain memilih level 1 | |

**Tabel 3.6** Skenario *Use Case* Memilih *Level*

| **Nama Use Case** | **UC-02** | **Memilih Level** |
| --- | --- | --- |
| **Pre Kondisi** | Pemain berada di Menu pilih level | |
| **Aksi** | **Aktor** | **Sistem** |
|  | 1. Pemain memilih level yang sudah terbuka | 2. Sistem menuju ke permainan dengan level yang dipilih |
| **Post Kondisi** | Pemain memainkan game | |

**Tabel 3. 7** Skenario *Use Case* Memainkan *Game*

| **Nama Use Case** | **UC-03** | **Memainkan Game** |
| --- | --- | --- |
| **Pre Kondisi** | Pemain berada di level permainan | |
| **Aksi** | **Aktor** | **Sistem** |
|  | 1. Pemain berada pada level yang dipilih | 1. Sistem menampilkan permainan dengan menampilkan musuh dan pertanyaan |
|  | 1. Pemain memainkan permainan dan menjawab pertanyaan yang muncul | 1. Sistem menampilkan pertanyaan tentang kepada pemain |
|  | 1. Pemain menjawab | 1. Jika jawaban pemain benar maka sistem akan memberikan kunci kepada pemain sebagai award/hadiah dan jika |
|  |  | 1. Jika jawaban pemain salah maka sistem tidak akan memberikan award/hadiah kepada pemain. |
|  | 1. Pemain melawan musuh | 1. Apabila pemain berhasil melawan musuh maka poin akan bertambah |
|  |  | 1. Apabila musuh menyerang pemain maka nyawa pemain akan berkurang |
|  | 1. Pemain menuju pintu ke level selanjutnya | 1. Apabila kunci cukup untuk membuka pintu maka sistem akan menampilkan level selanjutnya dan menampilkan poin dan kunci yang didapat. |
| **Aksi** | **Aktor** | **Sistem** |
|  |  | 1. Apabila kunci tidak cukup untuk membuka pintu maka sistem akan menampilkan *game over*. |
|  | 1. Pemain menyelesaikan semua level | 1. Permainan selesai, sistem akan kembali ke menu utama. |
| **Post Kondisi** | Pemain menyelesaikan permaianan | |

**Tabel 3.8** Skenario *Use Case Game Over*

| **Nama Use Case** | **UC-01** | ***Game Over*** |
| --- | --- | --- |
| Pre Kondisi | Pemain memainkan permainan | |
| **Aksi** | **Aktor** | **Sistem** |
|  | 1. Pemain diserang musuh | 2. Musuh menyerang dan nyawa pemain akan berkurang |
|  |  | 3. Apabila nyawa pemain telah habis maka sistem akan menampilkan *game over* dan pemain mengulang permainan di *level* yang sama. |
|  | 1. Pemain jatuh ke jurang | 1. Sistem akan menampilkan game over dan pemain mengulang permainan di level yang sama. |
|  | 1. Pemain sampai di depan pintu | 1. Apabila kunci yang terkumpul lebih dari atau sama dengan 4 maka pemain lanjut ke level berikutnya |
| **Aksi** | **Aktor** | **Sistem** |
|  |  | 1. Apabila kunci tidak mencukupi atau kurang dari 4 maka sistem menampilkan game over dan pemain mengulang permainan di level yang sama |
| **Post Kondisi** | Mengulang permainan | |

**Tabel 3.9** Skenario *Use Case* Menutup Aplikasi

| **Nama Use Case** | **UC-01** | **Menutup Aplikasi** |
| --- | --- | --- |
| Pre Kondisi | Pemain berada di Menu utama | |
| **Aksi** | **Aktor** | **Sistem** |
|  | 1. Pemain memilih tombol keluar | 2. Proses menutup aplikasi |
| **Post Kondisi** | Keluar dari permainan | |

### 3.7.2 Activity Diagram

#### 3.7.2.1 *Activity* Diagram Petunjuk

Gambar 3.7 adalah *Activity* diagram yang menggambarkan aktivitas saat melihat petunjuk.



**Gambar 3.7** *Activity* Diagram Petunjuk

#### 3.7.2.2 *Activity* Diagram Memilih *Play*

Gambar 3.8 adalah *Activity* diagram yang menggambarkan aktivitas memilih menu play



**Gambar 3.8** *Activity* Memilih *Play*

#### 3.7.2.3 *Activity* Diagram Memilih *Level*

Gambar 3.9 adalah *Activity* diagram yang menggambarkan aktivitas saat memilih level



**Gambar 3. 9** *Activity* Diagram Memilih *Level*

#### **3.7.2.4** *Activity* Diagram Memainkan Game

Gambar 3.10 adalah *Activity* diagram yang menggambarkan aktivitas saat bermain





**Gambar 3. 10** *Activity* Diagram Memainkan *Game*

#### 3.7.2.5 *Activity* Diagram *Game Over*

Gambar 3.11 adalah *Activity* diagram yang menggambarkan aktivitas saat permainan *game over*



**Gambar 3. 11** *Activity* Diagram *Game Over*

#### 3.7.2.8 *Activity* Diagram Keluar Permaianan

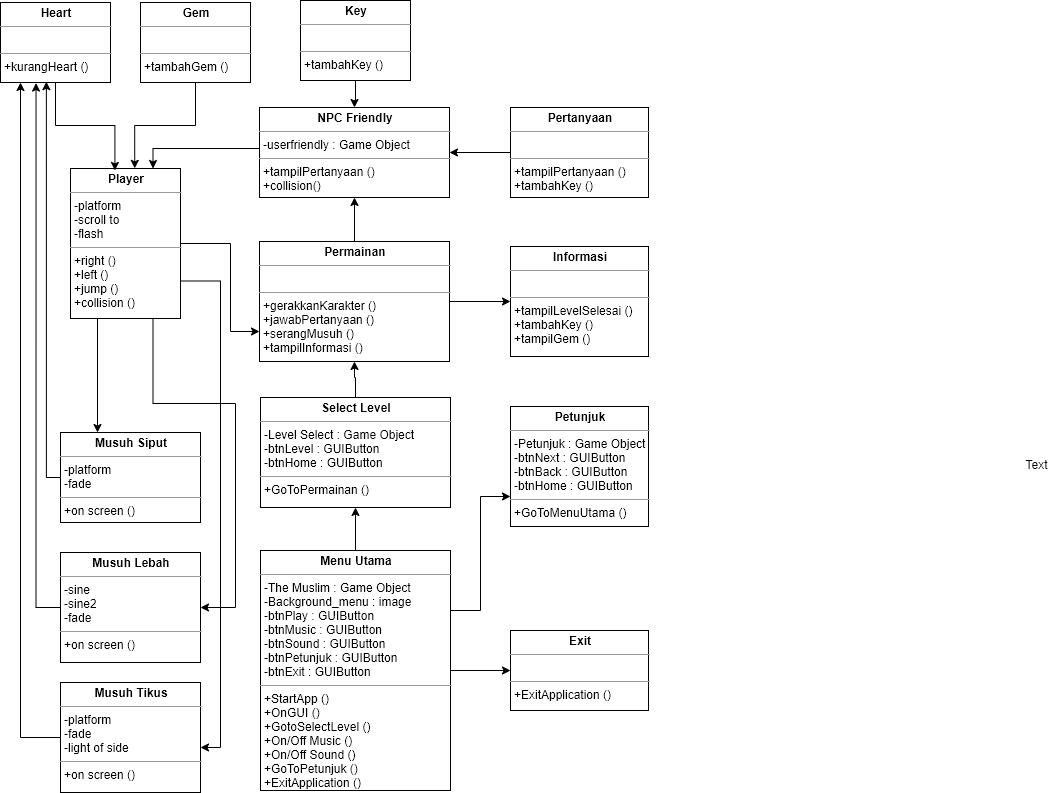
Gambar 3.12 adalah *Activity* diagram yang menggambarkan aktivitas saat bermain level 3



**Gambar 3. 12** *Activity* Diagram Keluar Permainan

### 3.7.3 *Class* Diagram

Diagram kelas atau *class* diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang dibuat untuk membangun sistem. Berikut ini adalah perancangan struktur sistem yang terdapat pada *game* edukasi The Muslim, yang digambarkan dengan class diagram pada Gambar 3.13 berikut



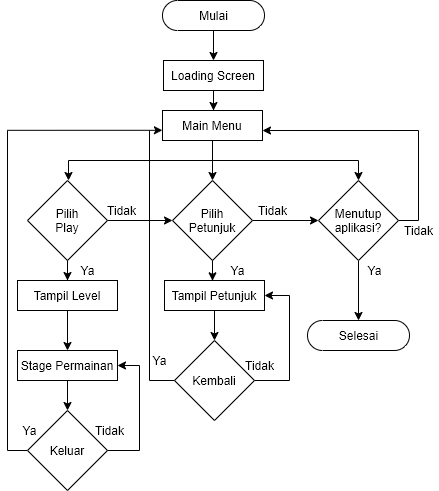
**Gambar 3. 13** *Class* Diagram *Game* Edukasi The Muslim

## 3.8 Perancangan Aplikasi

Perancangan Antarmuka meliputi perancangan *Flowchart* dan Struktur navigasi aplikasi serta perancangan *Storyboard* aplikasi.

### 3.8.1 Perancangan *Flowchart*

Perancangan *Flowchart* yang menggambarkan sistem proses kerja dalam menjalankan aplikasi dapat dilihat pada Gambar 3.14 berikut ini :

****

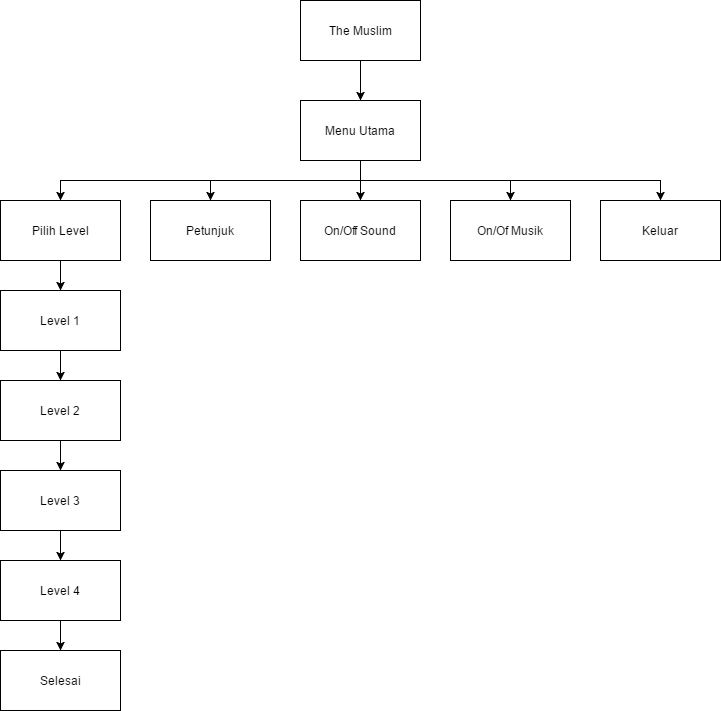
**Gambar 3. 14** *Flowchart Game* Edukasi The Muslim

Keterangan gambar flowchart di atas adalah sebagai berikut :

1. Pada saat *user* memilih menu Mulai Bermain maka sistem akan menampilkan menu pilih *level*
2. Pada saat *user* memilih menu Petunjuk maka sistem akan menampilkan menu tampilan petunjuk.
3. Pada saat user memilih level bermain maka sistem akan menampilkan menu bermain, apabila menang maka user akan lanjut ke level selanjutnya, apabila user gagal menyelesaikan level yang dimainkan maka user akan kembali bermain di level tersebut.
4. Apabila user menyelesaikan semua level maka permainan selesai, jika tidak maka user akan kembali bermain di level yang belum diselesaikan.
5. Pada saat *user* memilih menu Keluar maka *user* akan secara otomatis keluar dari aplikasi

### 3.8.2 Perancangan Struktur Navigasi

Berikut struktur navigasi yang menampilkan alur atau tahapan pada game edukasi The Muslim yang dapat dilihat pada Gambar 3.15 berikut ini :



%3CmxGraphModel%3E%3Croot%3E%3CmxCell%20id%3D%220%22%2F%3E%3CmxCell%20id%3D%221%22%20parent%3D%220%22%2F%3E%3CmxCell%20id%3D%222%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%223%22%20target%3D%225%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%223%22%20value%3D%22Game%20Edukasi%26lt%3Bbr%26gt%3BThe%20Muslim%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22380%22%20y%3D%2280%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%224%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%225%22%20target%3D%2210%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%225%22%20value%3D%22Menu%20Utama%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22380%22%20y%3D%22160%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%226%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3B%22%20edge%3D%221%22%20source%3D%2210%22%20target%3D%2212%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%3E%3CArray%20as%3D%22points%22%3E%3CmxPoint%20x%3D%22440%22%20y%3D%22300%22%2F%3E%3CmxPoint%20x%3D%22180%22%20y%3D%22300%22%2F%3E%3C%2FArray%3E%3C%2FmxGeometry%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%227%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%2210%22%20target%3D%2213%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%228%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%2210%22%20target%3D%2214%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%229%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%2210%22%20target%3D%2215%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2210%22%20value%3D%22Pilih%20Level%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22380%22%20y%3D%22240%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2211%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%2212%22%20target%3D%2217%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2212%22%20value%3D%22Level%201%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22120%22%20y%3D%22320%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2213%22%20value%3D%22On%2FOff%20Sound%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22280%22%20y%3D%22320%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2214%22%20value%3D%22On%2FOff%20Music%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22480%22%20y%3D%22320%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2215%22%20value%3D%22Keluar%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22640%22%20y%3D%22320%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2216%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%2217%22%20target%3D%2219%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2217%22%20value%3D%22Level%202%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22120%22%20y%3D%22400%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2218%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%2219%22%20target%3D%2220%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2219%22%20value%3D%22Level%203%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22120%22%20y%3D%22480%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2220%22%20value%3D%22Level%204%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22120%22%20y%3D%22560%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3C%2Froot%3E%3C%2FmxGraphModel%3E%3CmxGraphModel%3E%3Croot%3E%3CmxCell%20id%3D%220%22%2F%3E%3CmxCell%20id%3D%221%22%20parent%3D%220%22%2F%3E%3CmxCell%20id%3D%222%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%223%22%20target%3D%225%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%223%22%20value%3D%22Game%20Edukasi%26lt%3Bbr%26gt%3BThe%20Muslim%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22380%22%20y%3D%2280%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%224%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%225%22%20target%3D%2210%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%225%22%20value%3D%22Menu%20Utama%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22380%22%20y%3D%22160%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%226%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3B%22%20edge%3D%221%22%20source%3D%2210%22%20target%3D%2212%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%3E%3CArray%20as%3D%22points%22%3E%3CmxPoint%20x%3D%22440%22%20y%3D%22300%22%2F%3E%3CmxPoint%20x%3D%22180%22%20y%3D%22300%22%2F%3E%3C%2FArray%3E%3C%2FmxGeometry%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%227%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%2210%22%20target%3D%2213%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%228%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%2210%22%20target%3D%2214%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%229%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%2210%22%20target%3D%2215%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2210%22%20value%3D%22Pilih%20Level%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22380%22%20y%3D%22240%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2211%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%2212%22%20target%3D%2217%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2212%22%20value%3D%22Level%201%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22120%22%20y%3D%22320%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2213%22%20value%3D%22On%2FOff%20Sound%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22280%22%20y%3D%22320%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2214%22%20value%3D%22On%2FOff%20Music%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22480%22%20y%3D%22320%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2215%22%20value%3D%22Keluar%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22640%22%20y%3D%22320%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2216%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%2217%22%20target%3D%2219%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2217%22%20value%3D%22Level%202%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22120%22%20y%3D%22400%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2218%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%2219%22%20target%3D%2220%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2219%22%20value%3D%22Level%203%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22120%22%20y%3D%22480%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2220%22%20value%3D%22Level%204%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22120%22%20y%3D%22560%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3C%2Fro

%3CmxGraphModel%3E%3Croot%3E%3CmxCell%20id%3D%220%22%2F%3E%3CmxCell%20id%3D%221%22%20parent%3D%220%22%2F%3E%3CmxCell%20id%3D%222%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%223%22%20target%3D%225%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%223%22%20value%3D%22Game%20Edukasi%26lt%3Bbr%26gt%3BThe%20Muslim%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22380%22%20y%3D%2280%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%224%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%225%22%20target%3D%2210%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%225%22%20value%3D%22Menu%20Utama%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22380%22%20y%3D%22160%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%226%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3B%22%20edge%3D%221%22%20source%3D%2210%22%20target%3D%2212%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%3E%3CArray%20as%3D%22points%22%3E%3CmxPoint%20x%3D%22440%22%20y%3D%22300%22%2F%3E%3CmxPoint%20x%3D%22180%22%20y%3D%22300%22%2F%3E%3C%2FArray%3E%3C%2FmxGeometry%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%227%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%2210%22%20target%3D%2213%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%228%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%2210%22%20target%3D%2214%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%229%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%2210%22%20target%3D%2215%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2210%22%20value%3D%22Pilih%20Level%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22380%22%20y%3D%22240%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2211%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%2212%22%20target%3D%2217%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2212%22%20value%3D%22Level%201%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22120%22%20y%3D%22320%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2213%22%20value%3D%22On%2FOff%20Sound%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22280%22%20y%3D%22320%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2214%22%20value%3D%22On%2FOff%20Music%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22480%22%20y%3D%22320%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2215%22%20value%3D%22Keluar%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22640%22%20y%3D%22320%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2216%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%2217%22%20target%3D%2219%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2217%22%20value%3D%22Level%202%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22120%22%20y%3D%22400%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2218%22%20style%3D%22edgeStyle%3DorthogonalEdgeStyle%3Brounded%3D0%3BorthogonalLoop%3D1%3BjettySize%3Dauto%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0.5%3BentryY%3D0%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%2219%22%20target%3D%2220%22%20parent%3D%221%22%3E%3CmxGeometry%20relative%3D%221%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2219%22%20value%3D%22Level%203%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22120%22%20y%3D%22480%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2220%22%20value%3D%22Level%204%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22120%22%20y%3D%22560%22%20width%3D%22120%22%20height%3D%2240%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3C%2Froot%3E%3C%2FmxGraphMode

**Gambar 3. 15** Perancangan Struktur Navigasi *Game* Edukasi The Muslim

*Menu* Utama

Merupakan tampilan awal aplikasi, berisikan tombol-tombol *Menu* untuk mengarahkan ke *Layout* lain adalah tombol *Play, Sound dan Music*. Lalu terdapat *Menu* keluar dari aplikasi yaitu tombol keluar.

Pilih Level

Merupakan tombol untuk perpindahan dari halaman utama ke halaman pilih level yang terdiri dari level 1-4.

Level 1

*Menu* ini merupakan *menu* yang menampilkan permainan level 1 dengan tema tauhid.

Level 2

*Menu* ini merupakan *menu* yang menampilkan permainan level 2 dengan tema Aqidah & Akhlak.

Level 3

*Menu* ini merupakan *menu* yang menampilkan permainan level 3 dengan tema Al-Qur’an & Hadist.

Level 4

*Menu* ini merupakan *menu* yang menampilkan permainan level 4 dengan tema Sejarah Islam & Fiqih

Petunjuk

*Menu* ini merupakan *menu* yang menampilkan petunjuk bermain *game*

### 3.8.3 Perancangan Tampilan *Layout* dan *Storyboard* Aplikasi

Pengertian *Storyboard* adalah visualisasi dari aplikasi yang akan dibangun, sehingga dapat memberikan gambaran dari aplikasi yang akan dihasilkan. *Storyboard* dapat dikatakan juga *visual script* yang akan dijadikan *outline* dari sebuah proyek, ditampilkan *shot* by *shot* yang biasa disebut dengan istilah *scene.* Berikut dapat dilihat rancangan tampilan *Layout* dan *Storyboard* objek animasi pada *Game* ini.

**Tabel 3. 10** Perancangan Tampilan *Layout* dan *Storyboard* Aplikasi

| **NO** | **VISUAL** | **DESKRISPSI** |
| --- | --- | --- |
| 1 |  | Halaman *Splash Screen*, memunculkan animasi loading pada *Game.* |
| 2 |  | Halaman Petunjuk, menampilkan informasi cara bermain game   1. Home - Tampilkan *Layout* “Home” |
| 3 |  | Halaman *Menu* Utama, memunculkan Logo *Game*, dan beberapa tombol *Menu*, yaitu:   1. *Play* – Tampilkan *Layout* “*Play*” 2. Suara – Mengaktifkan dan menonaktifkan Suara 3. Musik - Mengaktifkan dan menonaktifkan *Music Game* 4. Keluar - Keluar *Game* 5. Petunjuk - Tampilkan Layout “Petunjuk” |
| **NO** | **VISUAL** | **DESKRISPSI** |
| 4 |  | Halaman pilih *stage* memunculkan stage yang akan dimainkan oleh *player*, terdapat 4 macam stage yaitu :   1. Stage 1 - Menampilkan stage Tauhid 2. Stage 2 - Menampilkan stage Aqidah & Akhlak 3. Stage 3 - Menampilkan stage Al-Quran & Hadits 4. Stage 4 - Menampilkan stage Sejarah Islam & Fiqih 5. Home - Tampilkan *Layout* “Home” |
| 5 |  | Halaman bermain terdiri dari :   1. Home – Tombol untuk kembali ke *menu* utama 2. Suara – Mengaktifkan dan menonaktifkan Suara 3. Musik - Mengaktifkan dan menonaktifkan *Music* pada *Game* 4. Health – Menampilkan jumlah darah yang dimiliki *player* 5. Kunci – Menampilkan jumlah kunci yang telah dikumpulkan 6. Poin Gem – Menampilkan jumlah gem yang dikumpulkan 7. Kiri – Tombol untuk menggerakkan *player* ke kiri 8. Kanan – Tombol untuk menggerakkan *player* ke kanan 9. Lompat – Tombol untuk menggerakkan *player* ke atas/lompat 10. Player – Sebagai pemain yang memiliki aksi utama yaitu lari, lompat, menyerang 11. Musuh Lebah - Musuh yang terbang ke kiri dan ke kanan, musuh dapat mati apabila player menyerang dengan melompat ke atas badan musuh tersebut 12. Musuh Tikus – Musuh yang berjalan ke kiri dan kanan dan akan mengejar/mengikuti pemain apabila berada di dekatnya, musuh akan mati apabila player menyererang dengan melompat ke atas badan musuh tersebut. |
| **NO** | **VISUAL** | **DESKRISPSI** |
|  |  | 1. Musuh Siput – Musuh berjalan ke kiri dan ke kanan dan dapat mati apabila player menyerang dengan melompat ke atas badan musuh sebanyak 3kali. 2. Pemberi Pertanyaan – NPC yang bertugas memberikan pertanyaan kepada player (Terdapat 5 NPC yang ditempatkan di berbagai lokasi) NPC akan memberikan kunci sebagai hadiah kepada player jika berhasil menjawab pertanyaan dengan benar 3. Gem – Poin yang akan dikumpulkan oleh player |
| 6 |  | Halaman tampilan level 1 dengan peranyaan seputar tauhid |
| 7 |  | Halaman tampilan level 2 dengan peranyaan seputar Aqidah & Akhlak |
| 8 |  | Halaman tampilan level 3 dengan peranyaan seputar Al-Qur’an dan Hadits |
| **NO** | **VISUAL** | **DESKRISPSI** |
| 9 |  | Halaman tampilan level 4 dengan peranyaan seputar Sejarah Islam & Fiqih |
| 10 |  | Halaman pertanyaan terdiri dari :   1. Pertanyaan – Menampilkan pertanyaan yang diberikan oleh NPC 2. Pilihan Jawaban – Menampilkan pilihan jawaban dari pertanyaan yang diberikan |
| 11 |  | Halaman game over terdiri dari:   1. Tulisan Game Over yang menandakan bahwa permainan berakhir dikarenakan player kehabisan health/darah 2. Poin Gem – Menampilkan jumlah gem yang dikumpulkan 3. Kunci - Menampilkan jumlah kunci yang didapat 4. Ulangi – Tombol untuk mengulang permainan |
| 12 |  | Halaman game over terdiri dari:   1. Tulisan *Level Complete* yang menandakan stage selesai 2. Poin Gem – Menampilkan jumlah gem yang dikumpulkan 3. Kunci - Menampilkan jumlah kunci yang didapat 4. Lanjut – Tombol untuk lanjut ke stage berikutnya |

## 3.9 Perancangan *Aset Game*

Pada perancangan *Game* ini, perancangan menggunakan *aset-aset* yang terdiri dari aset *image* dan *Aset Sounds*. Perancangan aset *Game* ini menggunakan aplikasi *Adobe Photoshop* untuk *Aset Images*. Sedangkan pada *aset* *audio* menggunakan aplikasi *audacity* untuk meng-*edit* *audio* lalu mengubah menjadi format ogg. Dalam *Game* ini, image yang dibutuhkan yaitu:

**Tabel 3. 11** *Aset Images*

| **No.** | **Aset** | **Deskripsi** |
| --- | --- | --- |
| 1. | Play1 | Button Play |
| 2. | Button_70 | Button Petunjuk |
| 3. | Button_03Button_05 | Button Level |
| 4 | Button_47 | Button next |
| 5 | Button_63 | Button back |
| 6 | Button_13 | Button Pilihan Jawaban |
| 7 | Button_50 | Button Home |
| 8 | Button_09 | Button Music |
| 9 | Button_13 | Button Sound |

| **No.** | **Aset** | **Deskripsi** |
| --- | --- | --- |
| 10 | Button_74 | Button Reply |
| 11 | Button_46 | Button Play/Resume |
| 12 | Button_23 | Button Close |
| 13 | shadedLight24 | Button Left |
| 14 | shadedLight25 | Button Right |
| 15 | shadedLight27 | Button Up |
| 16 | question | Pertanyaan |
| 17 | gameover | Level Complete |
| 18 | **complete** | Game Over |
| 19 | tileBlue_15tileBlue_15tileBlue_25tileBlue_24 | Jump tiles |
| 20 | *tileBlue_05* | Street tiles |

| **No.** | **Aset** | **Deskripsi** |
| --- | --- | --- |
| 21 | **playerstand** | Player |
| 22 | userfriendly | NPC Friendly |
| 23 | bee | Musuh Lebah |
| 24. | snail | Musuh Siput |
| 25. | mouse | Musuh Tikus |
| 26. | Gems | HUD Gem |
| 27 | hudHeart_full | HUD Heart |
| 28 | key | HUD Key |
| 29 | blueGem | Gem |

**Tabel 3. 12** *Aset Souds*

| **No.** | **Aset** | **Deskripsi** |
| --- | --- | --- |
| 1. | click.flac | Audio ketika menekan tombol/button |
| 2. | Collect Gem.wav | Audio ketika mendapatkan gem |
| 3. | Collect Key.wav | Audio ketika mendapatkan kunci |
| 3. | EnemyHit.wav | Audio ketika enemy menyerang player |
| 4. | Hit.wav | Audio ketika player menyerang enemy |
| 5. | Jingle Lose.wav | Audio ketika game over |
| 6. | Jingle Win.wav | Audio ketika level complete |
| 7. | MattOglseby - 1.m4a | Suara musik saat bermain |

## 3.10 Pengujian Aplikasi

Pengujian dalam penelitian ini dilakukan beberapa pengujian, berikut adalah pengujian yang dilakukan

### 3.10.1 Pengujian Blackbox

Pengujian Black Box ini merupakan metode pengujian dimana input berupa suatu set data digunakan untuk menguji validitas dari integrasi dan konsistensi sistem.

1. Rencana Pengujian

Rencana pengujian yang akan dilakukan pada aplikasi selengkapnya terlihat pada tabel dibawah ini.

**Tabel 3.13** Tabel rencana pengujian aplikasi

| **Nama Pengujian** | **Deskripsi** | **Jenis Uji** |
| --- | --- | --- |
|  |  |  |

Tabel rencana pengujian terdiri dari 3 kolom, yaitu :

- Objek Uji : berisi tentang objek-objek yang akan diuji.

- Detail Pengujian : berisi tentang proses yang akan diuji dari objek uji.

- Jenis Uji : berisi metode yang digunakan dalam pengujian.

1. Kasus Hasil Pengujian

Kasus hasil pengujian yang akan diambil dari aplikasi yang telah diuji terlihat pada tabel dibawah ini.

**Tabel 3.14** Tabel rencana kasus dan hasil uji aplikasi

| **Kasus dan Hasil Uji** | | | |
| --- | --- | --- | --- |
| **Data masukan** | **Target** | **Pengamatan** | **Kesimpulan** |
|  |  |  |  |

Tabel rencana pengujian terdiri dari 4 kolom, yaitu :

- Data masukan : berisi jenis masukan yang berasal dari objek yang akan diuji.

- Target : berisi tentang hasil proses yang diharapkan.

- Pengamatan : berisi tentang hasil proses dari data masukan apakah sama dengan target yang diharapkan atau tidak.

- Kesimpulan : berisi apakah data diterima atau ditolak

### 3.10.2 Pengujian Kuesioner

Pengujian yang akan dilakukan adalah *user acceptance test*, yaitu menguji tingkat penerimaan user terhadap g*ame* yang dibangun melalui kuesioner. Pengujian ini dilakukan oleh pengguna untuk menguji game dengan meminta respon pengguna terhadap game ini sebagai evaluasi. Hal yang mendasari penggunaan kuesioner sebagai alat penelitian adalah kuesioner dapat memberikan kemudahan bagi responden untuk memahami dan menjawab pertanyaan yang diajukan dengan baik. Selain itu kuesioner membuat responden lebih nyaman dan leluasa untuk menjawab pertanyaan.

Kuisioner dibagikan kepada 20 responden yang merupakan anak-anak dengan rentang usia 8-12 tahun dan beragama Islam yang nantinya akan menggunakan aplikasi secara langsung. Responden tersebut akan mencoba game secara langsung pada *smartphone* yang telah di-*install* kemudian mengisi kuesioner yang telah disediakan.

Kuesioner berisi 10 pertanyaan yang dikelompokkan menjadi 3 aspek yang digunakan dalam pengujian aplikasi game ini, yaitu aspek aesthetics, aspek story, dan aspek mechanics.

### Aspek Aesthetics

Aspek aesthetics pada kuesioner yang telah disiapkan terdapat pada pertanyaan - pertanyaan berikut ini :

* Bagaimana tampilan game edukasi ini?
* Bagaimana bentuk karakter, musuh, dan pemberi pertanyaan dalam game edukasi ini?
* Bagaimana suara musik dan suara efek dalam game edukasi ini?
* Apakah game ini menarik dan menyenangkan?

### Aspek Story

Aspek story pada kuesioner yang telah disiapkan terdapat pada pertanyaan - pertanyaan berikut ini :

* Apakah petunjuk pada permainan mudah dipahami?
* Apakah bahasa dalam game edukasi ini mudah dipahami?
* Apakah game edukasi ini dapat membantu dalam mempelajari ilmu dasar islam?
* Apakah pertanyaan yang muncul dapat dijawab dengan mudah?
* Game ini secara keseluruhan dapat digunakan sebagai sarana bermain dan belajar?

### Aspek Mechanics

Aspek mechanics pada kuesioner yang telah disiapkan terdapat pada pertanyaan berikut ini :

* Apakah game edukasi ini mudah dimainkan?

Berikut rancangan kuesioner yang dibagikan kepada user :

**Tabel 3. 15** Rancangan Kuesioner

| **No** | **Pertanyaan** | **Tanggapan** | | | | | **Total** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** |
| 1 | Bagaimana tampilan *game* edukasi ini? |  |  |  |  |  |  |
| 2 | Apakah petunjuk pada permainan mudah dipahami? |  |  |  |  |  |  |
| 3 | Bagaimana bentuk karakter player, musuh, dan pemberi pertanyaan dalam *game* edukasi ini ? |  |  |  |  |  |  |
| 4 | Bagaimana suara musik dan suara efek dalam aplikasi *game* edukasi ini ? |  |  |  |  |  |  |
| 5 | Apakah bahasa dalam *game* edukasi ini mudah dipahami? |  |  |  |  |  |  |
| 6 | Apakah game ini menarik dan menyenangkan? |  |  |  |  |  |  |
| 7 | Apakah *game* edukasi ini dapat membantu dalam mempelajari ilmu dasar islam ? |  |  |  |  |  |  |
| 8 | Apakah *game* edukasi ini mudah dimainkan ? |  |  |  |  |  |  |
| 9 | Apakah pertanyaan yang muncul dapat dijawab dengan mudah ? |  |  |  |  |  |  |
| 10 | *Game* ini secara keseluruhan dapat digunakan sebagai sarana bermain dan belajar ? |  |  |  |  |  |  |

Keterangan: 1=Sangat buruk 2=Buruk 3=Cukup baik

4=Baik 5=Sangat baik

Pengujian menggunakan 5 kategori jawaban dengan skor yang berbeda untuk setiap jawabannya seperti pada Tabel 3.16 berikut:

**Tabel 3. 16** Skor hasil kuesioner

| **Jawaban** | **Sangat Buruk** | **Buruk** | **Cukup Baik** | **Baik** | **Sangat Baik** |
| --- | --- | --- | --- | --- | --- |
| **Skor** | 1 | 2 | 3 | 4 | 5 |

Untuk mencari nilai persentase dari masing-masing pendapat kuesioner digunakan rumus skala Likert sebagai berikut:

p=f/n\*100%

p=Nilai persentase yang dicari

f=Jumlah frekuensi dikalikan dengan skor yang ditetapkan tiap jawaban

n=Skor ideal (Skor tertinggi dikalikan dengan jumlah sampel)

Dalam melakukan pengukuran usability, seperti yang telah dijelaskan pada rumus, diperlukan skor yang diobservasi dan skor yang diharapkan untuk mendapatkan persentase kelayakan. Skor yang diobservasi didapatkan dari jumlah skor roral seluruh jawaban dari 20 responden, sedangkan skor yang diharapkan siperoleh dari jumlah skor maksimal skala dikalikan dengan jumlah pertanyaan lalu dikalikan dengan jumlah responden. Kemudian data yang diperoleh dari perhitungan menggunakan rumus dikonversikan berdasarkan tabel kelayakan yang dapat dilihat pada tabel 3.17 dibawah ini.

**Tabel 3.17** Tabel Kelayakan

| Angka (%) | Klasifikasi |
| --- | --- |
| <25 | Tidak baik |
| 26-50 | Cukup baik |
| 50-75 | Baik |
| 75-100 | Sangat baik |

# Bab iv HASIL DAN ANALISIS

## Pada bab ini akan dilakukan implementasi dan pengujian terhadap aplikasi permainan hasil pengembangan yaitu Game Edukasi The Muslim. Tahap in dilakukan setelah tahap perancangan selesai dilakukan. Adapun hasil dari perancangan tersebut kemudian diimplementasikan pada bahasa pemrograman dengan menggunakan construct 2. Setelah implementasi, maka dilakukan pengujian terhadap aplikasi tersebut. Pengujian tersebut terdiri dari pengujian *alpha* dan *beta.*

## 4.1 Implementasi Sistem

Tahap implementasi merupakan tahap kelanjutan dari kegiatan perancangan sistem. Wujud dari hasil implementasi ini nantinya adalah sebuah sistem yang siap untuk diuji dan digunakan.

## 4.2 Implementasi Antar Muka Aplikasi

### Implementasi antar muka merupakan tahap dimana sebuah sistem siap dioprasikan pada tahap yang sebenarnya, sehingga bisa diketahui apakah sistem yang telah dibuat ini sudah sesuai dengan apa yang sudah dirancang sebelumnya.

### 4.2.1 Tampilan *Menu* Utama

Adapun tombol pilihan *Menu* yang ada pada tampilan *Menu* utama, yaitu:

Tombol ”play” jika tombol *Menu* ini ditekan maka *user* akan masuk ke *Menu* Pilih Level yang berisi pilihan level 1-4. Dengan level yg terbuka untuk pertama kali adalah level 1.

Tombol ”petunjuk” jika tombol ini ditekan maka *user* akan masuk ke *Menu* petunjuk cara bermain.

Tombol ”sound” jika tombol ini ditekan untuk menghidupkan dan mematikan suara sound.

Tombol ”musik” jika tombol ini ditekan untuk menghidupkan dan mematikan suara musik.

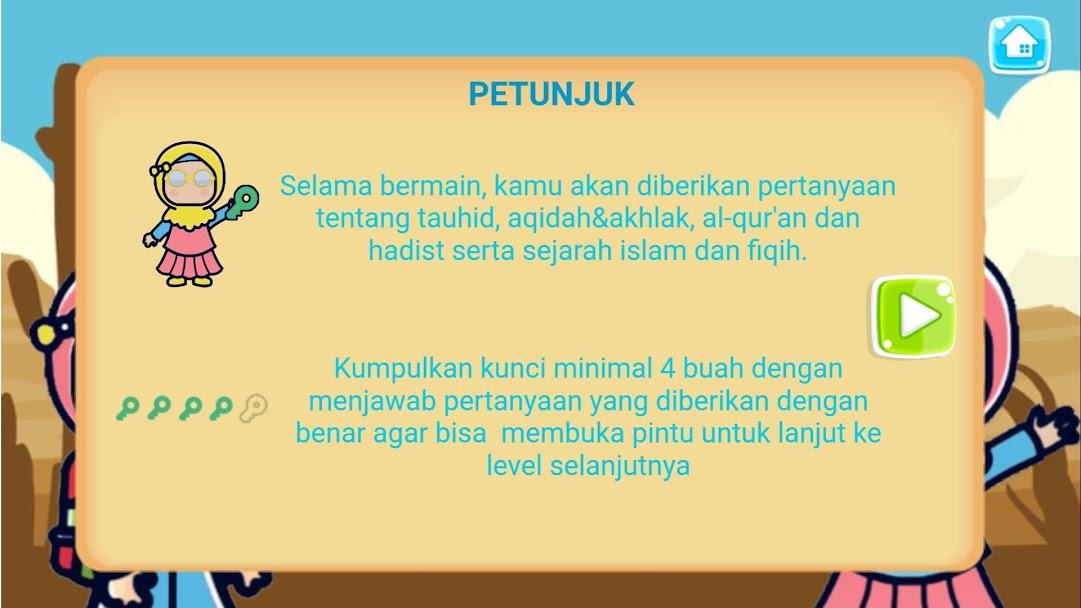
Tombol ”*close*” jika tombol ini ditekan user keluar dari aplikasi mengakhiri aplikasi.

****

**Gambar 4. 1**Tampilan Menu Utama

## 4.2.2 Tampilan *Menu* Petunjuk

Pada *menu* petunjuk ini, dijelaskan petunjuk bermain *game* yaitu dengan menjelaskan setiap karakter yang ada di *game* dan memberikan informasi apa yang harus dilakukan selama bermain. Tampilan *menu* petunjuk dapat di lihat pada gambar berikut ini



**Gambar 4. 2** Tampilan *Menu* Petunjuk Slide 1

Jika tombol next atau tombol hijau ditekan maka akan menampilkan slide ke-2 yang dapat dilihat pada gambar di bawah ini



**Gambar 4. 3** Tampilan *Menu* Petunjuk Slide 2

## 4.2.3 Tampilan *Menu* Pilih *Level*

Pada menu Pilih Level ini, menampilkan level yang akan dimainkan mulai dari level 1-4. Pada awal memulai permainan, pemain hanya bisa memainkan level 1. Level 3-4 akan terbuka apabila user berhasil menyelesaikan permainan di level sebelumnya. Adapun gambar tampilan *menu* pilih level dapat dilihat pada gambar 4.4 berikut ini

**Gambar 4. 4** Tampilan *Menu* Pilih *Level*

Adapun tombol pilihan *Menu* yang ada pada tampilan *Menu* Pilih Level, yaitu:

1. Tombol ”level 1” jika tombol *Menu* ini ditekan maka *user* akan masuk ke Permainan awal/ level 1.
2. Tombol “home:” jika tombol *Menu* ini ditekan maka *user* akan kembali ke *menu* utama.

## 4.2.4 Tampilan *Menu* Level 1

Pada awal permainan, pemain memiliki sebanyak 10 nyawa yang dilambangkan dengan hati. Pemain juga dapat mengumpulkan gem atau permata biru pada saat bermain. Selain itu terdapat 5 buah kunci yang masih kosong, kunci akan terisi apabila pemain berhasil menjawab pertanyaan dengan benar. Pemain dapat lanjut ke level selanjutnya apabila kunci yang dikumpulkan berjumlah 4 atau 5. Adapun tampilan level 1 dapat dilihat pada gambar 4.5 di bawah ini



**Gambar 4. 5** Tampilan *Menu Level* 1

Pada level 1 pemain akan diberikan 5 buah pertanyaan seputar tauhid dengan 2 pilihan jawaban. Adapun tampilan pertanyaan pada level 1 dapat dilihat pada gambar di bawah ini



**Gambar 4. 6** Tampilan Pertanyaan Level 1

## 4.2.5 Tampilan *Menu* Level 2

Setelah pemain menyelesaikan level 1 maka pemain akan lanjut ke level 2. Adapun tampilan level 2 dapat dilihat pada gambar 4.7 di bawah ini



**Gambar 4. 7** Tampilan *Menu Level* 2

Pada level 2 pemain akan diberikan 5 buah pertanyaan seputar Aqidah & Akhlak dengan 2 pilihan jawaban. Adapun tampilan pertanyaan pada level 2 dapat dilihat pada gambar 4.8 di bawah ini



**Gambar 4. 8** Tampilan Pertanyaan Level 2

## 4.2.6 Tampilan *Menu* Level 3

Setelah pemain menyelesaikan level 2 maka pemain akan lanjut ke level 3. Adapun tampilan level 3 dapat dilihat pada gambar 4.9 di bawah ini



**Gambar 4.9** Tampilan *Menu Level* 3

Pada level 2 pemain akan diberikan 5 buah pertanyaan seputar Al-Qur’an & Hadits dengan 2 pilihan jawaban. Adapun tampilan pertanyaan pada level 3 dapat dilihat pada gambar 4.10 di bawah ini



**Gambar 4. 10** Tampilan Pertanyaan Level 3

## 4.2.7 Tampilan *Menu* Level 4

Setelah pemain menyelesaikan level 3 maka pemain akan lanjut ke level 4. Adapun tampilan level 4 dapat dilihat pada gambar 4.11 di bawah ini

**Gambar 4.11** Tampilan *Menu Level* 4

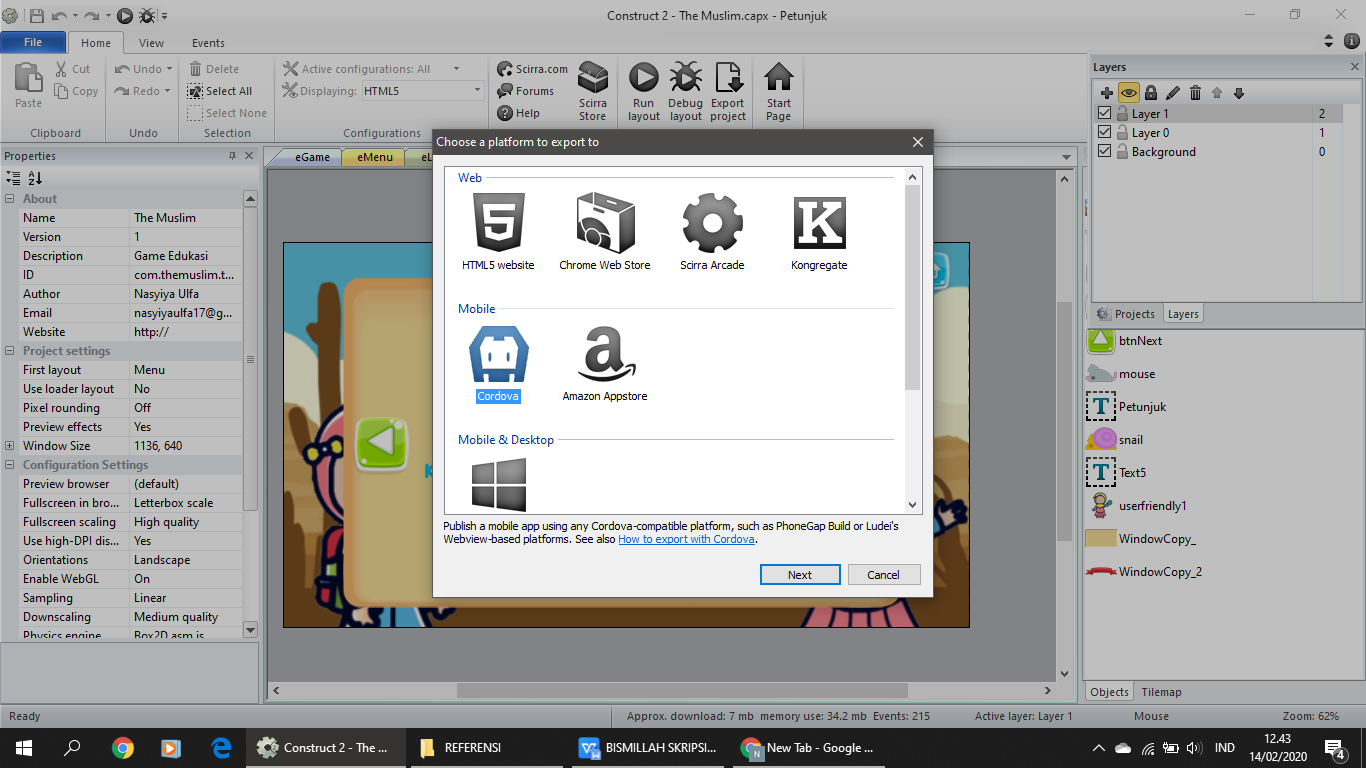
Pada level 2 pemain akan diberikan 5 buah pertanyaan seputar Sejarah Islam & Fiqih dengan 2 pilihan jawaban. Adapun tampilan pertanyaan pada level 4 dapat dilihat pada gambar 4.12 di bawah ini



**Gambar 4. 12** Tampilan Pertanyaan Level 4

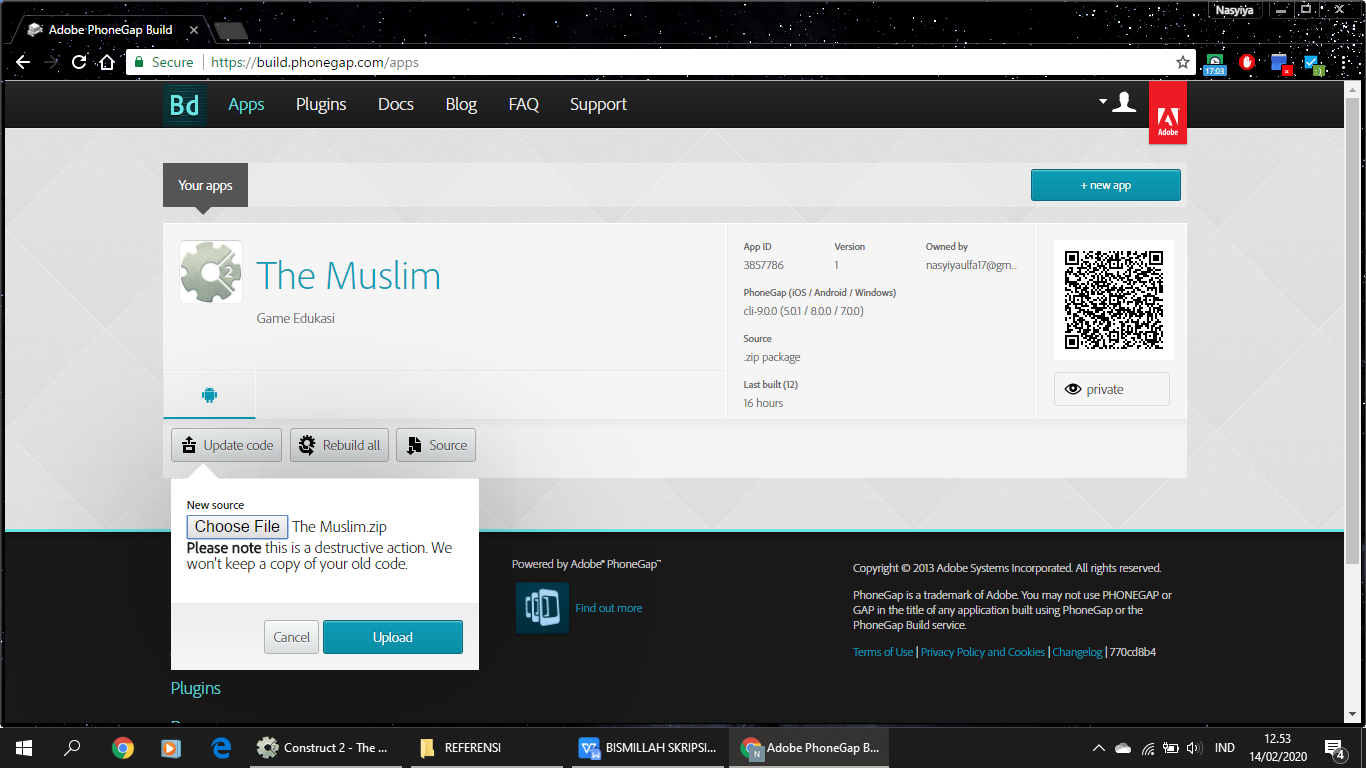
## 4.3 *Export* Aplikasi

Pada tahap ini game yang telak dibuat di aplikasi construct 2 di export ke dalam format *cordova* dengan memilih pilihan export pada aplikasi construct 2 yang telah disediakan yang dapat dilihat pada gambar 4.13 dibawah ini.



**Gambar 4. 13** Proses export ke format co*rdova*

Setelah proses export selesai, selanjutnya upload file yang telah di export tadi dalam bentuk .zip ke *Adobe PhoneGap* untuk mengubah format ke dala bentuk .apk agar bisa dimainkan di android. Setelah selesai di export dan tidak terdapat error, maka game dengan format .apk dapat langsung diunduh. *Adobe* *PhoneGap* adalah sebuah kerangka kerja/framework open source yang dipakai untuk membuat aplikasi cross-platform mobile dengan HTML, CSS, dan JavaScript. Proses upload file ke *Adobe PhoneGap* dapat dilihat pada gambar 4.14 berikut



**Gambar 4. 14** Proses export ke format *.apk*

## 4.4 Pengujian Metode *Finite State Machine (Black-Box)*

Pada hasil pengujian metode *finite state machine* yang diterapkan pada *NPC Friendly* (Karakter Pemberi Pertanyaan) dan *NPC NonFriendly* (Musuh) pada saat game dijalankan, menghasilkan data sebagai berikut :

**Tabel 4.1** Hasil Uji Coba Metode Finite State Machine pada NPC Friendly

| **State** | **Masukan** | **Keterangan** | |
| --- | --- | --- | --- |
| **Sesuai** | **Tidak** |
| Memberi pertanyaan | Player mendekat | √ | - |
| Memberi kunci | Player menjawab benar | √ | - |
| Tidak memberi kunci | Player menjawab salah | √ | - |
| Diam | Player lanjut perjalanan | √ | - |

Pada tabel 4.1 diketahui, jika player mendekat maka state yang aktif yaitu NPC akan memberi pertanyaan. Jika player menjawab dengan benar maka NPC akan memberikan kunci kepada player, dan jika player menjawab salah maka NPC tidak akan memberikan kunci. Selanjutnya jika player lanjut perjalanan maka state yang akan aktif yaitu NPC kembali diam.

**Tabel 4.2** Hasil Uji Coba Metode Finite State Machine pada Musuh Lebah

| **State** | **Masukan** | **Keterangan** | |
| --- | --- | --- | --- |
| **Sesuai** | **Tidak** |
| Terbang | Player melewati musuh | √ | - |
| Menyerang | Player menyentuh | √ | - |
| Mati | Player menyerang | √ | - |

Pada tabel 4.2 diketahui, jika player melewati musuh maka musuh akan tetap berada di kondisi awal yaitu terbang. Jika player menyentuh musuh maka state yang terjadi yaitu musuh menyerang, selanjutnya apabila player menyerang musuh maka state yang terjadi yaitu musuh mati.

**Tabel 4.3** Hasil Uji Coba Metode Finite State Machine pada Musuh Tikus

| **State** | **Masukan** | **Keterangan** | |
| --- | --- | --- | --- |
| **Sesuai** | **Tidak** |
| Siaga / berjalan | player melewati musuh / player menjauh | √ | - |
| Mengikuti player | Player mendekat/ Player lari | √ | - |
| Menyerang | Player menyentuh | √ | - |
| Mati | Player menyerang | √ | - |

Pada tabel 4.3 diketahui, jika player melewati musuh atau player menjauh dari musuh maka state yang terjadi yaitu musuh akan siaga/berjalan ke kiri dan ke kanan. Apabila player mendekat maka musuh akan mengikuti player, begitupula ketika musuh menyerang player dan terjadi masukan player lari maka musuh akan kembali mengikuti player jika masih berada di area pengejaran. Selanjutnya jika player menyentuh musuh maka state yang terjadi yaitu musuh menyerang dan jika player menyerang musuh maka musuh akan mati.

**Tabel 4.4** Hasil Uji Coba Metode Finite State Machine pada Musuh Siput

| **State** | **Masukan** | **Keterangan** | |
| --- | --- | --- | --- |
| **Sesuai** | **Tidak** |
| Siaga / berjalan | Player melewati musuh | √ | - |
| Diam / terlungkup | Player menyerang kali ke-1 | √ | - |
| Bergerak ke kiri & kanan | Player menyerang kali ke-2 | √ | - |
| Menyerang | Player menyentuh/ Terjadi sentuhan | √ | - |
| Mati | Player menyerang kali ke-3 | √ | - |

Pada tabel 4.4. diketahui, jika player melewati musuh maka state yang terjadi yaitu musuh siaga/berjalan ke kiri dan ke kanan. Jika player menyerang kali ke-1 maka musuh akan diam/terlungkup. Apabila musuh menyerang kali ke-2 maka musuh akan bergerak ke kiri & kanan dalam keadaan terlungkup. Selanjutnya jika terjadi sentuhan antara musuh dan player maka state yang terjadi yaitu musuh menyerang, dan yang terakhir jika player menyerang kali ke-3 maka musuh akan mati.

Dari tabel 4.1, 4.2, 4.3 dan 4.4 di atas, diketehui bahwa pengujian metode Finite State Machinepada *game* The Muslim telah berhasil diterapkan dan berjalan sesuai rancangan yang dibuat.

## 4.5 Pengujian Aplikasi

Pengujian dilakukan untuk mencari dan menemukan kesalahan atau kekurangan pada aplikasi *game* edukasi The Muslim. Pengujian dimaksud untuk mengetahui perangkat lunak yang dibuat sudah memenuhi kriteria yang sesuai dengan tujuan perancangan perangkat lunak tersebut. Pengujian yang digunakan dalam menguji *game* edukasi The Muslim adalah dengan menggunakan blackbox dan pengujian kuesioner.

### 4.5.1 Pengujian *Black Box*

Pengujian *blackbox* untuk mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. Pengujian *black box* didasarkan pada *use case* yang telah dibuat.

Berikut ini tabel rencana pengujian dari aplikasi game edukasi The Muslim yang dapat dilihat pada tabel 4.5 berikut.

**Tabel 4.5** Rencana pengujian aplikasi

| **Nama Pengujian** | **Deskripsi** | **Jenis Uji** |
| --- | --- | --- |
| Melihat Petunjuk | Merupakan fitur untuk melihat panduan petunjuk bermain | *Black box* |
| Memilih *Play* | Merupakan fitur untuk memulai bermain | *Black box* |
| Memilih Level | Merupakan fitur memilih level permainan/level 1 | *Black box* |
| Memainkan game | Merupakan fitur untuk bermain level permainan | *Black box* |
| Game Over | Merupakan fitur ketika permainan berakhir | *Black box* |
| Menutup Aplikasi | Merupakan fitur keluar dari aplikasi | *Black box* |

**4.5.1.2 Hasil Pengujian *Black Box***

Berdasarkan rencana pengujian aplikasi yang telah disusun, maka dapat dilakukan beberapa tahap pengujian sebagai berikut :

1. Pengujian melihat petunjuk

**Tabel 4.6** Pengujian menu melihat petunjuk

| **Kasus dan Hasil Uji** | | | |
| --- | --- | --- | --- |
| **Data Masukan** | **Target**  **yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Memilih tombol icon petunjuk | Menampilkan *menu*  petunjuk bermain | Menampilkan *menu*  petunjuk bermain | Sesuai |

1. Pengujian menu memilih *play*

**Tabel 4.7** Pengujian menu memilih *play*

| **Kasus dan Hasil Uji** | | | |
| --- | --- | --- | --- |
| **Data Masukan** | **Target**  **yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Memilih tombol *play* | Menampilkan pilihan *level* 1-4 | Menampilkan pilihan *level* 1-4 | Sesuai |

1. Pengujian menu memilih *level*

**Tabel 4.8** Pengujian menu memilih level

| **Kasus dan Hasil Uji** | | | |
| --- | --- | --- | --- |
| **Data Masukan** | **Target**  **yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Memilih level yang sudah terbuka | Menampilkan permainan dengan level yang dipilih | Menampilkan permainan dengan level yang dipilih | Sesuai |

1. Pengujian menu memainkan *game*

**Tabel 4.9** Pengujian menu memainkan *game*

| **Kasus dan Hasil Uji** | | | |
| --- | --- | --- | --- |
| **Data Masukan** | **Target**  **yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Pemain berada pada level yang dipilih | Menampilkan permainan dengan menampilkan musuh dan pertanyaan | Menampilkan permainan dengan menampilkan musuh dan pertanyaan | Sesuai |
| Menjawab pertanyaan dengan benar | NPC memberikan hadiah berupa kunci kepada pemain | NPC memberikan hadiah berupa kunci kepada pemain | Sesuai |
| Menjawab pertanyaan dengan salah | NPC tidak memberikan hadiah | NPC tidak memberikan hadiah | Sesuai |
| **Kasus dan Hasil Uji** | | | |
| **Data Masukan** | **Target**  **yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Pemain berhasil melawan musuh | Poin/Gem bertambah | Poin/Gem bertambah | Sesuai |
| Musuh menyerang pemain | Nyawa pemain berkurang | Nyawa pemain berkurang | Sesuai |
| Kunci cukup untuk membuka pintu | Menampilkan level selanjutnya dan menampilkan poin dan kunci yang didapat | Menampilkan level selanjutnya dan menampilkan poin dan kunci yang didapat | Sesuai |
| Kunci tidak cukup untuk membuka pintu | Menampilkan *game over* | Menampilkan *game over* | Sesuai |
| Menyelesaikan semua level | Permainan selesai, kembali ke menu utama. | Permainan selesai, kembali ke menu utama. | Sesuai |

1. Pengujian menu *Game Over*

**Tabel 4.10** Pengujian menu g*ame over*

| **Kasus dan Hasil Uji** | | | |
| --- | --- | --- | --- |
| **Data Masukan** | **Target**  **yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Pemain diserang musuh | Nyawa pemain berkurang | Nyawa pemain berkurang | Sesuai |
| **Kasus dan Hasil Uji** | | | |
| **Data Masukan** | **Target**  **yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Nyawa pemain telah habis | Menampilkan *game over* dan mengulang permainan di *level* yang sama | Menampilkan *game over* dan mengulang permainan di *level* yang sama | Sesuai |
| Pemain jatuh ke jurang | Menampilkan *game over* dan mengulang permainan di *level* yang sama | Menampilkan *game over* dan mengulang permainan di *level* yang sama | Sesuai |
| Kunci yang terkumpul lebih dari atau sama dengan 4 | Menampilkan level berikutnya | Menampilkan level berikutnya | Sesuai |
| Kunci tidak mencukupi atau kurang dari 4 | Menampilkan game over dan mengulang permainan di level yang sama | Menampilkan game over dan mengulang permainan di level yang sama | Sesuai |

1. Pengujian menu keluar aplikasi

**Tabel 4.11** Pengujian menu keluar aplikasi

| **Kasus dan Hasil Uji** | | | |
| --- | --- | --- | --- |
| **Data Masukan** | **Target**  **yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| Memilih tombol keluar | Keluar dari aplikasi *game* | Keluar dari aplikasi *game* | Berhasil |

### 4.5.2 Hasil Pengujian Kuesioner

Dalam pengujian kuesioner akan dilakukan secara objektif dengan cara dicoba langsung oleh responden anak-anak yang berusia 8-12 tahun untuk dapat mengetahui sejauh mana sistem yang dibangun dapat menjadi alternatif penyelesaian permasalahan yang telah dijelaskan dilatar belakang. Responden berjumlah 20 orang, nantinya responden akan dipersilahkan mencoba aplikasi dan mengisi kuesioner yang telah disediakan.

Kuesioner terdiri 10 pertanyaan, yang bertujuan untuk menguji tingkat penerimaan *user* terhadap sistem. Berikut hasil pengujian yang dibagikan kepada *user* pada tabel 4.12

**Tabel 4.12** Hasil Pengisian Kuisioner

| **No** | **Pertanyaan** | **Tanggapan** | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** |
| 1 | Bagaimana tampilan *game* edukasi ini? | 0 | 0 | 2 | 6 | 12 |
| 2 | Apakah petunjuk pada permainan mudah dipahami? | 0 | 0 | 3 | 11 | 16 |
| 3 | Bagaimana bentuk karakter player, musuh, dan pemberi pertanyaan dalam *game* edukasi ini ? | 0 | 0 | 3 | 7 | 10 |
| 4 | Bagaimana suara musik dan suara efek dalam aplikasi *game* edukasi ini ? | 0 | 0 | 5 | 8 | 7 |
| 5 | Apakah bahasa dalam *game* edukasi ini mudah dipahami? | 0 | 0 | 0 | 4 | 16 |
| 6 | Apakah game ini menarik dan menyenangkan? | 0 | 0 | 1 | 6 | 13 |
| 7 | Apakah *game* edukasi ini dapat membantu dalam mempelajari ilmu dasar islam ? | 0 | 0 | 2 | 4 | 14 |
| 8 | Apakah *game* edukasi ini mudah dimainkan ? | 0 | 0 | 4 | 7 | 9 |
| 9 | Apakah pertanyaan yang muncul dapat dijawab dengan mudah ? | 0 | 0 | 5 | 7 | 8 |
| 10 | *Game* ini secara keseluruhan dapat digunakan sebagai sarana bermain dan belajar ? | 0 | 0 | 0 | 6 | 14 |

Keterangan: 1=Sangat buruk 2=Buruk 3=Cukup baik

4=Baik 5=Sangat baik

**Tabel 4.13** Hasil Persentase Kuesioner nomor 1

| Pertanyaan | Keterangan | Skor | Frekuensi | S |
| --- | --- | --- | --- | --- |
| 1 | Sangat Buruk | 1 | 0 | 0 |
| Buruk | 2 | 0 | 0 |
| Cukup Baik | 3 | 2 | 6 |
| Baik | 4 | 6 | 24 |
| Sangat Baik | 5 | 12 | 60 |
| Jumlah | | | 20 | 90 |

P = (90/100) \* 100% = 90%

Berdasarkan persentase di atas maka dapat disimpulkan bahwa tampilan *Game* ini sangat baik dengan hasil persentase 90% dari 100% atau bisa dikategorikan sebagian besar responden menjawab sangat baik. Berikut adalah kriteria interpretasi skor

| Sangat Buruk | Buruk | Cukup Baik | Baik | Sangat Baik |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |



**Tabel 4.14** Hasil Persentase Kuesioner nomor 2

| Pertanyaan | Keterangan | Skor | Frekuensi | S |
| --- | --- | --- | --- | --- |
| 2 | Sangat Buruk | 1 | 0 | 0 |
| Buruk | 2 | 0 | 0 |
| Cukup Baik | 3 | 3 | 9 |
| Baik | 4 | 11 | 44 |
| Sangat Baik | 5 | 6 | 30 |
| Jumlah | | | 20 | 83 |

P = (83/100) \* 100% = 83%

Berdasarkan persentase di atas maka dapat disimpulkan bahwa petunjuk pada permainan mudah dipahami dengan hasil persentase 83% dari 100% atau bisa dikategorikan sebagian besar responden menjawab sangat baik. Berikut adalah kriteria interpretasi skor.

| Sangat Buruk | Buruk | Cukup Baik | Baik | Sangat Baik |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |



**Tabel 4.15** Hasil Persentase Kuesioner nomor 3

| Pertanyaan | Keterangan | Skor | Frekuensi | S |
| --- | --- | --- | --- | --- |
| 3 | Sangat Buruk | 1 | 0 | 0 |
| Buruk | 2 | 0 | 0 |
| Cukup Baik | 3 | 3 | 9 |
| Baik | 4 | 7 | 28 |
| Sangat Baik | 5 | 10 | 50 |
| Jumlah | | | 20 | 87 |

P = (87/100) \* 100% = 87%

Berdasarkan persentase di atas maka dapat disimpulkan bahwa bentuk karakter player, musuh, dan pemberi pertanyaan dalam game ini sangat baik dengan hasil persentase 87% dari 100% atau bisa dikategorikan sebagian besar responden menjawab sangat baik. Berikut adalah kriteria interpretasi skor.

| Sangat Buruk | Buruk | Cukup Baik | Baik | Sangat Baik |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |



**Tabel 4.16** Hasil Persentase Kuesioner nomor 4

| Pertanyaan | Keterangan | Skor | Frekuensi | S |
| --- | --- | --- | --- | --- |
| 4 | Sangat Buruk | 1 | 0 | 0 |
| Buruk | 2 | 0 | 0 |
| Cukup Baik | 3 | 5 | 15 |
| Baik | 4 | 8 | 32 |
| Sangat Baik | 5 | 7 | 35 |
| Jumlah | | | 20 | 82 |

P = (82/100) \* 100% = 82%

Berdasarkan persentase di atas maka dapat disimpulkan suara musik dan suara efek dalam game edukasi ini sangat baik dengan hasil persentase 82% dari 100% atau bisa dikategorikan sebagian besar responden menjawab sangat baik. Berikut adalah kriteria interpretasi skor.

kriteria interpretasi skor.

| Sangat Buruk | Buruk | Cukup Baik | Baik | Sangat Baik |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |



**Tabel 4.17** Hasil Persentase Kuesioner nomor 5

| Pertanyaan | Keterangan | Skor | Frekuensi | S |
| --- | --- | --- | --- | --- |
| 5 | Sangat Buruk | 1 | 0 | 0 |
| Buruk | 2 | 0 | 0 |
| Cukup Baik | 3 | 0 | 0 |
| Baik | 4 | 4 | 16 |
| Sangat Baik | 5 | 16 | 80 |
| Jumlah | | | 20 | 96 |

P = (96/100) \* 100% = 96%

Berdasarkan persentase di atas maka dapat disimpulkan bahwa bahasa pada game edukasi ini musah dipahami dengan hasil persentase 90% dari 100% atau bisa dikategorikan sebagian besar responden menjawab sangat baik. Berikut adalah kriteria interpretasi skor.

| Sangat Buruk | Buruk | Cukup Baik | Baik | Sangat Baik |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |



**Tabel 4.18** Hasil Persentase Kuesioner nomor 6

| Pertanyaan | Keterangan | Skor | Frekuensi | S |
| --- | --- | --- | --- | --- |
| 6 | Sangat Buruk | 1 | 0 | 0 |
| Buruk | 2 | 0 | 0 |
| Cukup Baik | 3 | 1 | 3 |
| Baik | 4 | 6 | 24 |
| Sangat Baik | 5 | 13 | 65 |
| Jumlah | | | 20 | 92 |

P = (92/100) \* 100% = 92%

Berdasarkan persentase di atas maka dapat disimpulkan bahwa game ini menarik dan menyenangkan untuk dimainkan dengan hasil persentase 92% dari 100% atau bisa dikategorikan sebagian besar responden menjawab sangat baik. Berikut adalah kriteria interpretasi skor.

| Sangat Buruk | Buruk | Cukup Baik | Baik | Sangat Baik |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |



**Tabel 4.19** Hasil Persentase Kuesioner nomor 7

| Pertanyaan | Keterangan | Skor | Frekuensi | S |
| --- | --- | --- | --- | --- |
| 7 | Sangat Buruk | 1 | 0 | 0 |
| Buruk | 2 | 0 | 0 |
| Cukup Baik | 3 | 2 | 6 |
| Baik | 4 | 4 | 14 |
| Sangat Baik | 5 | 14 | 70 |
| Jumlah | | | 20 | 90 |

P = (90/100) \* 100% = 90%

Berdasarkan persentase di atas maka dapat disimpulkan bahwa game edukasi ini dapat membantu dalam mempelajari ilmu dasar islam dengan hasil persentase 90% dari 100% atau bisa dikategorikan sebagian besar responden menjawab sangat baik. Berikut adalah kriteria interpretasi skor.

| Sangat Buruk | Buruk | Cukup Baik | Baik | Sangat Baik |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |



**Tabel 4.20** Hasil Persentase Kuesioner nomor 8

| Pertanyaan | Keterangan | Skor | Frekuensi | S |
| --- | --- | --- | --- | --- |
| 8 | Sangat Buruk | 1 | 0 | 0 |
| Buruk | 2 | 0 | 0 |
| Cukup Baik | 3 | 5 | 15 |
| Baik | 4 | 7 | 28 |
| Sangat Baik | 5 | 8 | 40 |
| Jumlah | | | 20 | 83 |

P = (83/100) \* 100% = 83%

Berdasarkan persentase di atas maka dapat disimpulkan bahwa *Game* ini mudah dimainkan, dengan hasil persentase 83% dari 100% atau bisa dikategorikan sebagian besar responden menjawab sangat baik. Berikut adalah kriteria interpretasi skor.

| Sangat Buruk | Buruk | Cukup Baik | Baik | Sangat Baik |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |



**Tabel 4.21** Hasil Persentase Kuesioner nomor 9

| Pertanyaan | Keterangan | Skor | Frekuensi | S |
| --- | --- | --- | --- | --- |
| 9 | Sangat Buruk | 1 | 0 | 0 |
| Buruk | 2 | 0 | 0 |
| Cukup Baik | 3 | 5 | 15 |
| Baik | 4 | 7 | 28 |
| Sangat Baik | 5 | 8 | 40 |
| Jumlah | | | 20 | 83 |

P = (83/100) \* 100% = 83%

Berdasarkan persentase di atas maka dapat disimpulkan bahwa pertanyaan yang muncul pada game dapat dijawab dengan mudah dengan hasil persentase 83% dari 100% atau bisa dikategorikan sebagian besar responden menjawab sangat baik. Berikut adalah kriteria interpretasi skor.

| Sangat Buruk | Buruk | Cukup Baik | Baik | Sangat Baik |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |



**Tabel 4.22** Hasil Persentase Kuesioner nomor 10

| Pertanyaan | Keterangan | Skor | Frekuensi | S |
| --- | --- | --- | --- | --- |
| 10 | Sangat Buruk | 1 | 0 | 0 |
| Buruk | 2 | 0 | 0 |
| Cukup Baik | 3 | 0 | 0 |
| Baik | 4 | 6 | 24 |
| Sangat Baik | 5 | 14 | 70 |
| Jumlah | | | 20 | 94 |

P = (94/100) \* 100% = 94%

Berdasarkan persentase di atas maka dapat disimpulkan bahwa game ini secara keseluruhan dapat digunakan sebagai sarana bermain dan belajar dengan hasil persentase 94% dari 100% atau bisa dikategorikan sebagian besar responden menjawab sangat baik. Berikut adalah kriteria interpretasi skor.

| Sangat Buruk | Buruk | Cukup Baik | Baik | Sangat Baik |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |



#### 4.5.3 User Acceptance Test

Total skor responden dirangkum dalam sebuah tabel seperti pada tabel 4.23 berikut:

**Tabel 4. 23** Hasil Total Skor Responden

| **Responden** | **Pertanyaan** | | | | | | | | | | **Total** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| Responden 1 | 5 | 4 | 5 | 3 | 4 | 5 | 4 | 5 | 3 | 5 | 43 |
| Responden 2 | 5 | 5 | 5 | 4 | 5 | 5 | 5 | 5 | 4 | 4 | 47 |
| Responden 3 | 5 | 4 | 5 | 5 | 5 | 5 | 5 | 4 | 4 | 5 | 47 |
| Responden 4 | 4 | 5 | 4 | 4 | 5 | 4 | 5 | 4 | 5 | 4 | 44 |
| Responden 5 | 5 | 4 | 5 | 4 | 5 | 5 | 5 | 5 | 5 | 5 | 48 |
| **Responden** | **Pertanyaan** | | | | | | | | | | **Total** |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| Responden 6 | 5 | 5 | 5 | 5 | 5 | 4 | 5 | 5 | 5 | 5 | 49 |
| Responden 7 | 5 | 4 | 5 | 5 | 5 | 5 | 5 | 4 | 5 | 5 | 48 |
| Responden 8 | 4 | 4 | 5 | 5 | 5 | 4 | 5 | 4 | 3 | 5 | 44 |
| Responden 9 | 5 | 5 | 5 | 4 | 5 | 5 | 5 | 5 | 5 | 4 | 48 |
| Responden 10 | 4 | 4 | 4 | 4 | 4 | 5 | 5 | 4 | 5 | 5 | 44 |
| Responden 11 | 5 | 4 | 5 | 3 | 5 | 5 | 5 | 5 | 4 | 5 | 46 |
| Responden 12 | 5 | 4 | 3 | 3 | 5 | 5 | 4 | 3 | 3 | 5 | 40 |
| Responden 13 | 4 | 4 | 3 | 4 | 4 | 5 | 5 | 3 | 4 | 5 | 41 |
| Responden 14 | 5 | 3 | 4 | 4 | 5 | 5 | 4 | 3 | 4 | 4 | 41 |
| Responden 15 | 5 | 3 | 4 | 3 | 4 | 4 | 5 | 4 | 3 | 5 | 40 |
| Responden 16 | 3 | 5 | 4 | 5 | 5 | 4 | 3 | 5 | 4 | 5 | 43 |
| Responden 17 | 5 | 4 | 5 | 5 | 5 | 5 | 5 | 3 | 5 | 5 | 47 |
| Responden 18 | 4 | 5 | 4 | 3 | 5 | 5 | 5 | 5 | 5 | 5 | 46 |
| Responden 19 | 4 | 4 | 3 | 4 | 5 | 4 | 4 | 4 | 4 | 4 | 40 |
| Responden 20 | 3 | 3 | 4 | 5 | 5 | 3 | 3 | 5 | 3 | 4 | 38 |
| **Total Skor** | | | | | | | | | | | 884 |

1. Jumlah skor untuk setiap responden:

* Skor maksimal = 50 (5x10 pertanyaan)
* Skor minimal = 10 (1x10 pertanyaan)
* Skor median = 30 (3x10 pertanyaan)
* Skor kuartil I = 20 (2x10 pertanyaan)
* Skor kuartil II = 40 (4x10 pertanyaan)

2. Jumlah skor untuk seluruh responden:

* Maksimal = 1000 (20x50)
* Minimal = 200 (20x10)
* Median = 600 (20x30)
* Kuartil I = 400 (20x20)
* Kuartil II = 800 (20x40)

3. Interpretasi jumlah skor:

* 800 < skor < 1000, artinya sangat setuju (program dinilai berhasil)
* 600 < skor < 800, artinya setuju (program dinilai cukup berhasil)
* 400 < skor < 600, artinya kurang setuju (program dinilai kurang berhasil)
* 200 < skor < 400, artinya sangat tidak setuju (program dinilai tidak berhasil)





|  |
| --- |



## 4.6 Analisis Hasil Pengujian

Berikut ini adalah hasil analisis hasil perancangan dan pengujian perangkat lunak aplikasi permainan dalam penelitian:

1. Hasil pengujian fungsional sistem yang dilakukan dengan menggunakan metode *blackbox* menunjukkan bahwa aplikasi ini dapat digunakan dan berjalan sesuai dengan perancangan.
2. Hasil pengujian metode *Finite State Machine* dengan menggunakan metode blackbox menunjukkan bahwa metode yang diterapkan pada aplikasi ini sesuai dengan perancangan yang dibuat.
3. Hasil pengujian *user acceptance test* (UAT) menggunakan metode *likerts Summarated Raiting* (LSR) menunjukan bahwa hasil penelitian berada pada skor 884 – 1000, yang artinya responden menilai aplikasi sangat positif dan dinilai berhasil.
4. Secara umum responden menyukasi aplikasi *game* edukasi “The Muslim” dan dinilai dapat membantu dalam mempelajari ilmu dasar islam.
5. Berdasarkan pengujian dapat disimpulkan bahwa aplikasi *game* edukasi The Muslim dinilai berhasil.

# Bab v Penutup

## 5.1 Kesimpulan

Setelah dilakukan pengujian analisis terhadap aplikasi *game* edukasi The Muslim, maka ditarik kesimpulan sebagai berikut :

1. Penerapan metode *finite state machine* berhasil diterapkan pada *NPC Friendly* & NPC Non-Friendly dan berjalan sesuai perancangan yang dibuat.
2. Secara umum responden menyukasi aplikasi game ini baik dari segi tampilan, storyline, gameplay, dan pertanyaan yang muncul dalam game ini dinilai dapat membantu dalam mempelajari ilmu dasar islam.
3. Hasil perancangan dan pengujian menunjukkan bahwa aplikasi ini dapat digunakan dan berjalan dengan lancar.
4. Berdasarkan pengujian *user acceptance test* (UAT), didapat bahwa pengguna menilai aplikasi sangat positif dan berhasil dengan nilai didapat adalah 884 dari nilai maksimum 1000.

## 5.2 Saran

Adapun hal atau saran yang dapat dilakukan pada penelitian berikutnya adalah sebagai berikut :

1. Penambahan penerapan *Artificial Intelligent* yang lebih kompleks sehingga game menjadi lebih menantang.
2. Aplikasi tidak hanya di *platform* android tetapi dapat berjalan di *platfrom* lainnya seperti IOS.

# Daftar Pustaka

Akaha, A. Z. 2001. *Psikologi Anak dan Remaja Muslim*. Jakarta Timur: Pustaka al-Kautsar.

Adi, N. 2009. *Analisa dan Perancangan Sistem Informasi dengan Metologi Berorientasi Objek.* Bandung: Penerbit Informatika.

Annubaha, Chakim. 2014. Implementasi Finite State Machine (Fsm) Pada Game 3d Edukasi Bahasa Arab. *Teknik Informatika Fakultas Sains Dan Teknologi.*

Brownlee, J. 2010. Finite State Machines (FSM), Artificial Intelligence Depot. 26 Juli, 2019. <http://ai-depot.com/FiniteStateMachines/FSM.html>

Chikhungunya. 2011, 26 Mei. *Definisi Game dan Jenis-Jenisnya.* 26 Juli, 2019. <https://chikhungunya.wordpress.com/2011/05/26/definisi-game-dan-jenis-jenisnya/>

Daradjat, Z. 1971. *Membina Nilai-nilai Moral di Indonesia*. Jakarta: Bulan Bintang.

Dillon, T. 2005. *Adventure Game for Learning and Story Tellin.* UK: Future Lab.

Gullen, A. 2011, 26 Juni. *Beginner’s Guide to Construct 2.* 26 Juli, 2019. https://www.construct.net/en/tutorials/beginners-guide-to-construct-2-47

Hanafi, Muhammad Murti. 2016. Game Tukkarmatika Berbasis Android Menggunakan Metode Finite State Machine Dan Algoritma A.

Hariyanto, B. 2004. *Sistem Manajemen Basis Data: Pemodelan, Perancangan, dan Terapannya.* Informatika: Bandung.

Ismail, A. 2009*. Education Games Menjadi Cerdas dan Ceria dengan Permainan Edukatif.* Yogyakarta: Pilar Media.

Istiyanto, Jazi Eko. 2014. *Pengantar Elektronika dan Instrumentasi (Pendekatan Project Arduino dan Android).* Yogyakarta: Penerbit Andi.

Pressman, Roger S. 2002. *Rekayasa Perangkat Lunak: Pendekatan Praktisi (Buku I)*. Yogyakarta: ANDI.

Reynolds, C. W. 1999*. Steering Behaviors For Autonomous Characters.* America: Sony Computer Entertainment.

Riyanto, Yatim. 2006. Pengembangan Kurikulum dan Seputar Kurikulum Tingkat Satuan Pendidikan (KTSP). *IKAPI: Universiti Press.*

Rostianingsih, S.; Satiabudhi, G.; dan Wijaya, H. K. 2013. *Game Simulasi Finite State Machine Untuk Pertanian Dan Peternakan. Konferensi Nasional*

*Sistem Informasi.*

Safaat, N. 2012. Android: Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android. Bandung: Informatika.

Setiawan, I. 2006. *Perancangan Software Embedded System Berbasis FSM*. Semarang: Universitas Diponegoro.

Sugiyono. 2012. *Metode Penelitian Pendidikan*. Bandung: Alfabeta.

Sukamto, R. A. dan Shalahuddin, M. 2013. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek.* Bandung: Informatika.

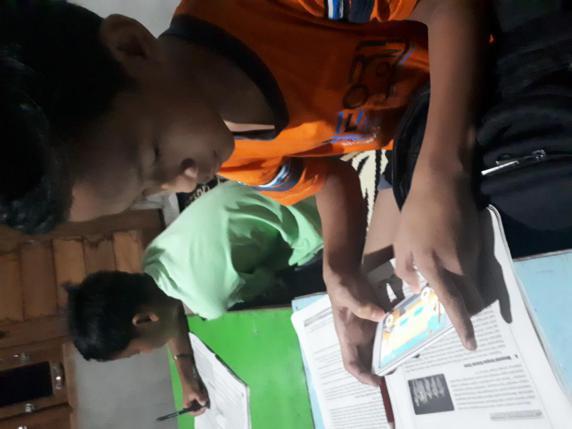
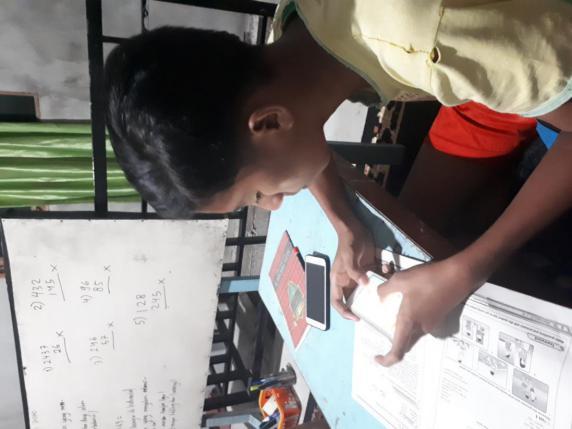
Supardi, Y. 2012. *Sistem Operasi Andal Android.* Jakarta: Elex Media Komputindo,

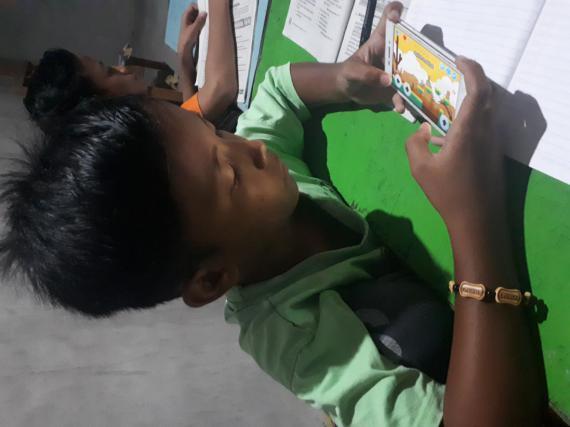
Tim Pusat Bahasa DEPDIKNAS. 2008. Kamus Bahasa Indonesia. Jakarta: Pusat Bahasa Departemen Pendidikan Nasional.

Tjandra, Cristianto. 2011. Dasar Pembuatan Game Edukasi Interaktif Untuk Mendukung Pemahaman Percentage Pada Siswa Sekolah.

Wahana, K. 2014. *Mobile App Development With PhoneGap.* Indonesia: Andi Publisher.

**DOKUMENTASI PENGUJIAN DI BIMBEL INAWA EDUCATION**





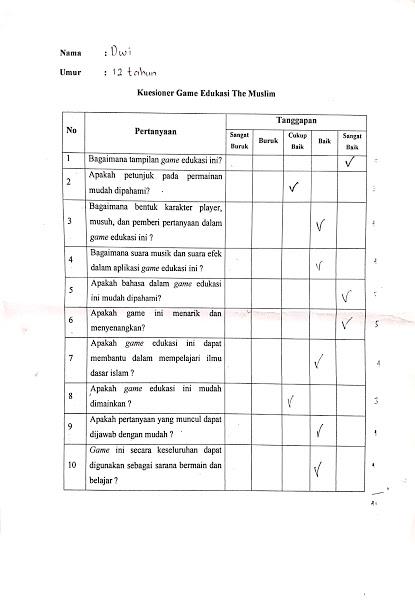


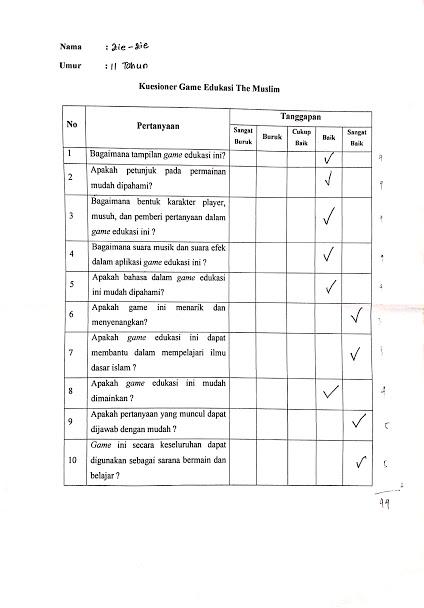


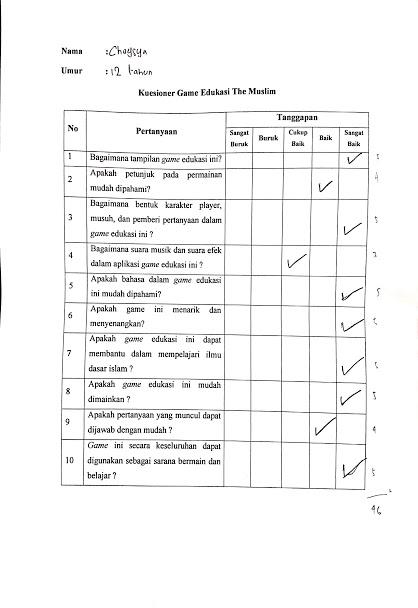


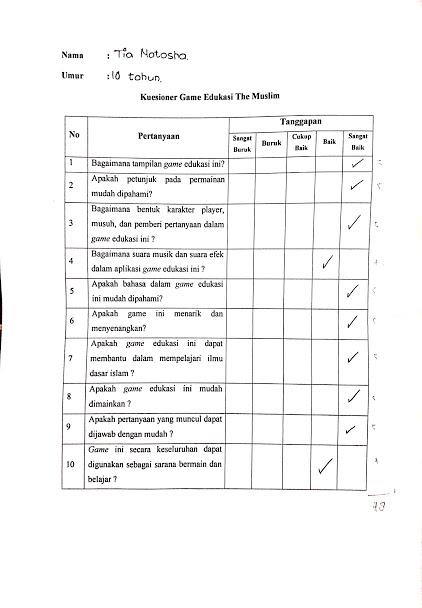
**Lampiran b**

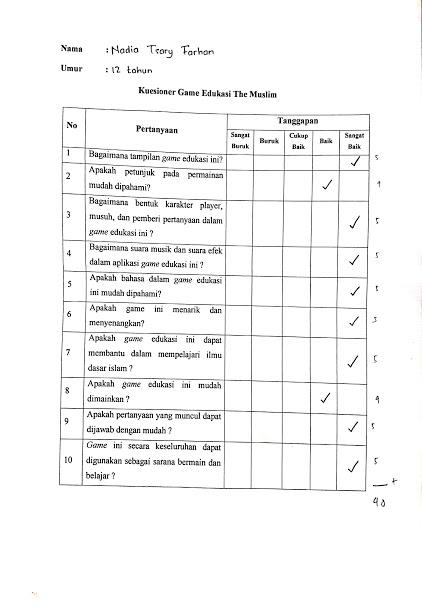
**kuesioner game edukasi the muslim**

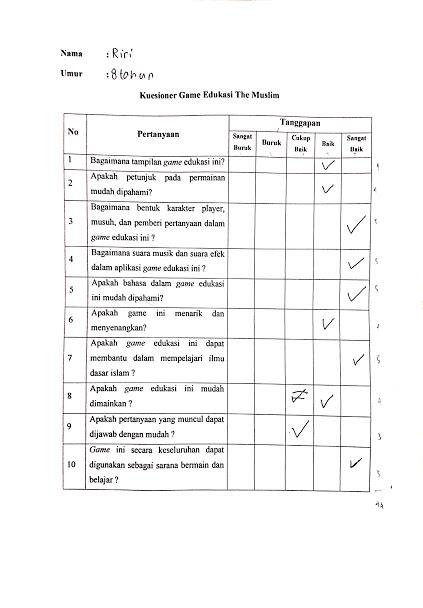
****

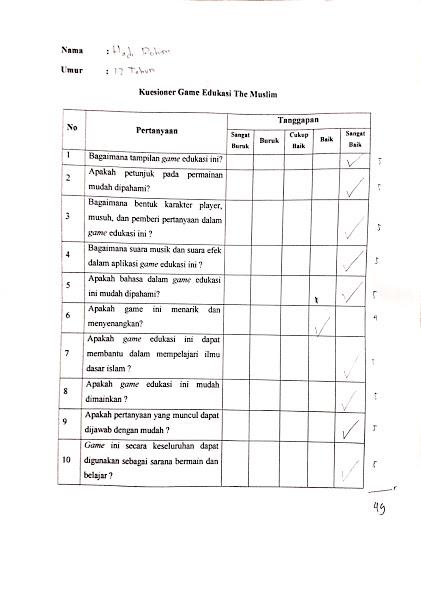
****

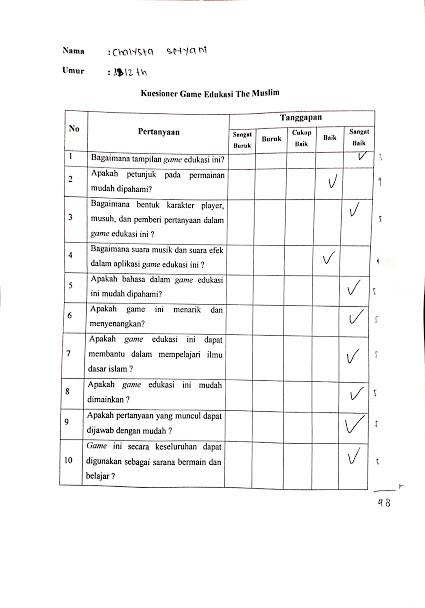


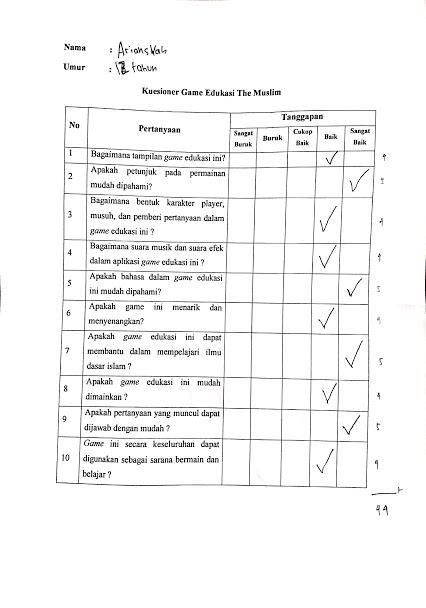


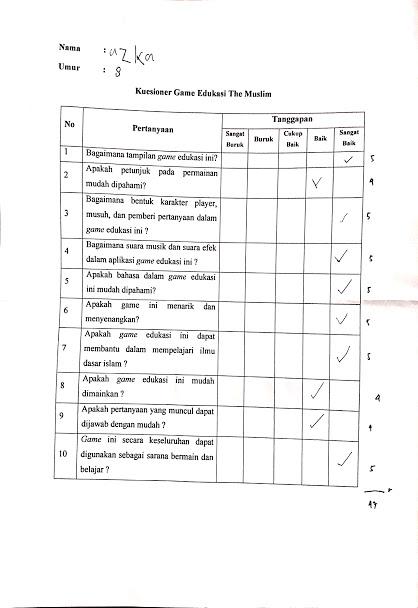


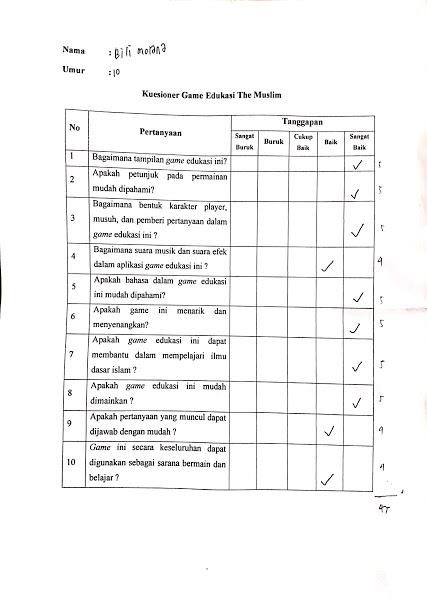


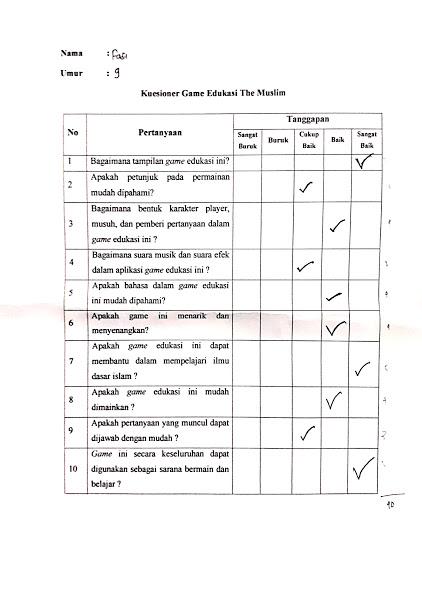


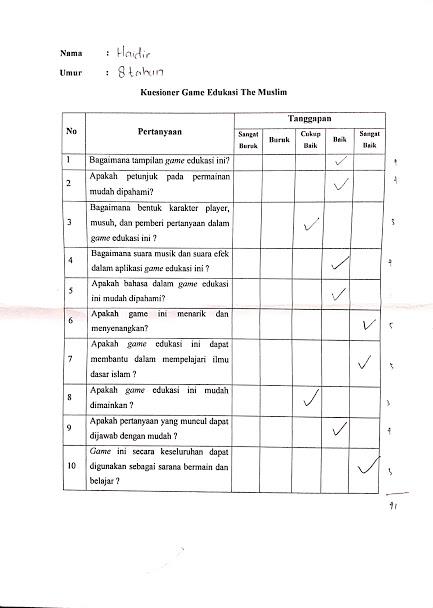




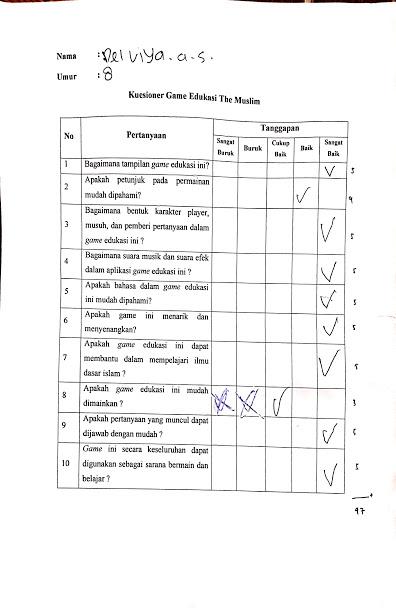


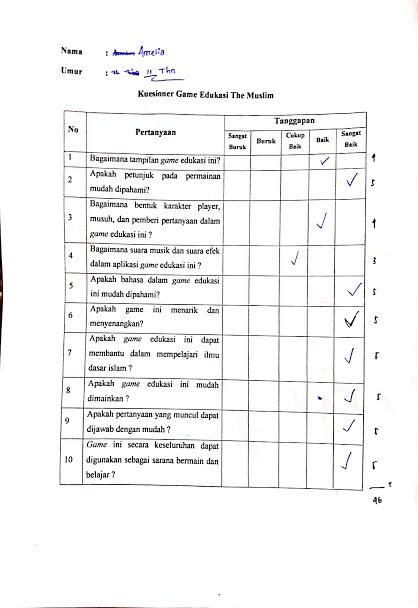


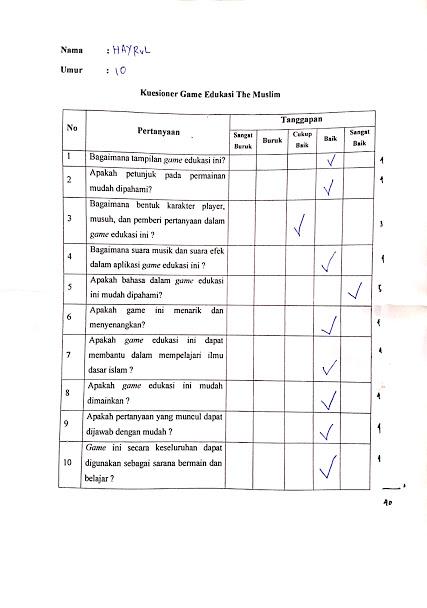


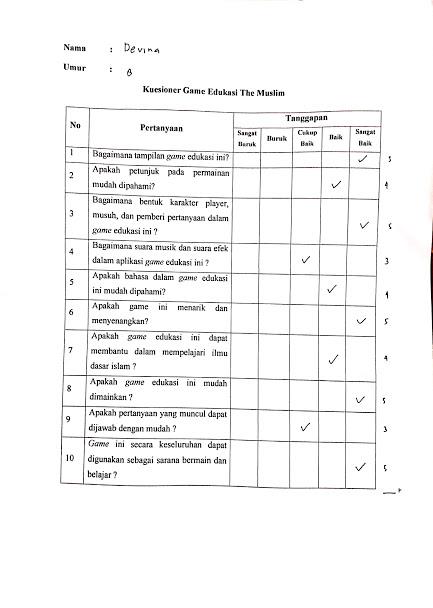


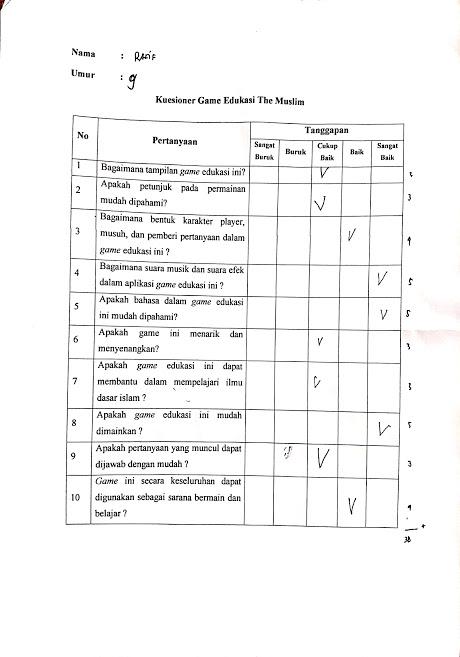


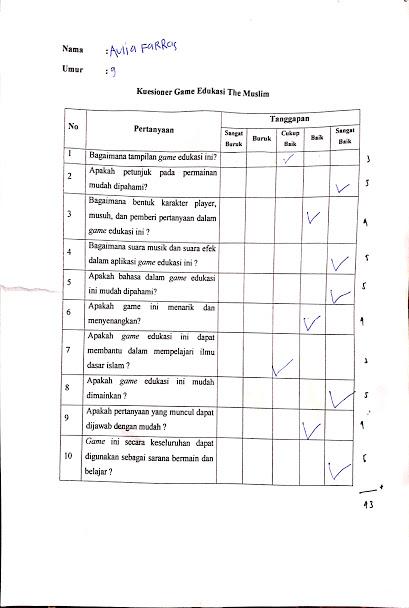






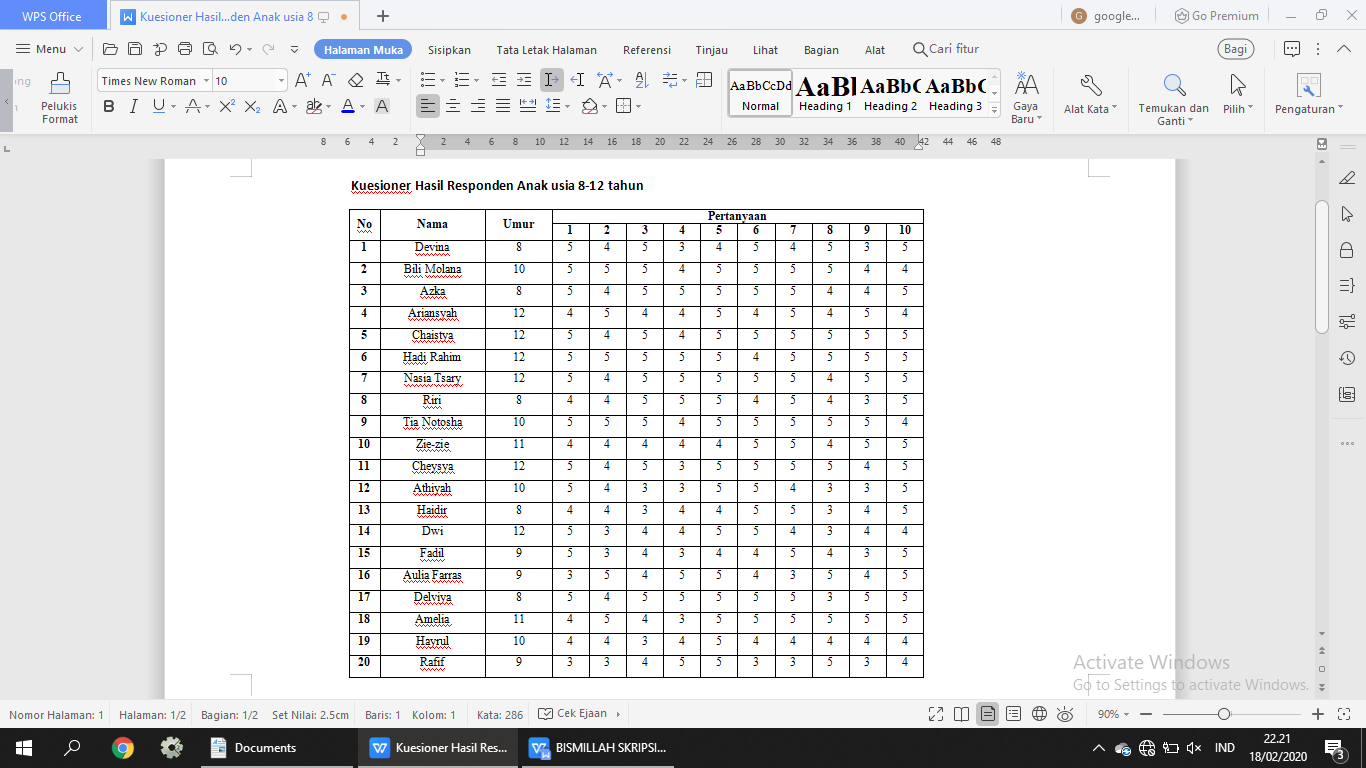






**Lampiran c**

**Hasil kuesioner game edukasi the muslim**

****