



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk mengubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB III

METODOLOGI PENELITIAN DAN PERANCANGAN SISTEM

3.1 Metodologi Penelitian

Dalam penelitian ini, metodologi penelitian yang dilakukan adalah sebagai berikut.

1. Studi Literatur

Dalam proses ini mempelajari teori-teori yang berhubungan dengan *game* yang akan dibangun, seperti pembuatan *game* dengan C# berbasis Android, *Fisher-Yates Shuffle*, jenis-jenis hewan beserta ciri-cirinya, konsep dan kategori *game* edukasi.

2. Analisis Kebutuhan

Analisis kebutuhan fitur-fitur yang akan dibutuhkan untuk melakukan perancangan dan pembangunan *game* simulasi, seperti *script controller* apa saja yang dibuat. *Script controller* adalah sebuah objek yang berguna sebagai pengaturan semua alur sistem yang terdapat pada *game* yang dibangun. *Controller* yang terdapat dalam sebuah *game* sangat beragam, seperti *script controller animation*, *slider controller*, dan *music controller*.

3. Perancangan Aplikasi

Perancangan *flowchart* dibuat untuk memudahkan melihat alur keseluruhan dari sistem *game*. Setelah perancangan selesai, dilanjutkan dengan perancangan *player interface*.

4. Pembangunan Aplikasi

Pembangunan *game* menggunakan Unity versi 5.1.3f1 (64-bit) dan Mono Develop sebagai IDE. Dalam pembangunan *game* bahasa pemrograman yang

digunakan adalah C#. Perancangan aset *sprite* dan animasi yang lain dibuat menggunakan Corel Draw X7 dan Adobe Photoshop CS6.

5. *Testing* dan Survei

Pada tahapan ini pengujian aplikasi *game* dilakukan dengan menggunakan *smartphone* Samsung Galaxy Mega 6.3, Samsung Galaxy Grand Prime, dan Xiaomi Redmi 3. Proses pengujian dilakukan pada anak-anak dengan rentang usia 5 – 8 tahun.

Tahapan dalam pengujian adalah sebagai berikut.

- a. Koresponden diberikan pengenalan hewan dengan metode *text book*.
- b. Koresponden diberikan kuesioner pengukuran minat.
- c. Selama pengisian kuesioner koresponden diberikan penjelasan maksud dari setiap pertanyaan.
- d. Koresponden diberikan *game* pengenalan hewan untuk dimainkan.
- e. Koresponden diberikan kuesioner pengukuran minat kembali.
- f. Selama pengisian kuesioner koresponden diberikan penjelasan maksud dari setiap pertanyaan.
- g. Koresponden diminta untuk mengisi kuesioner ketiga mengenai tampilan dan navigasi dari *game*.

Jumlah minimum peserta yang akan dievaluasi adalah 30 orang. Menurut Gray dan Diehl (1992) sampel minimum untuk penelitian adalah 30 sampel.

6. Evaluasi

Pada tahapan ini akan dilakukan evaluasi terhadap data yang telah dikumpulkan dari pengisian survei yang dilakukan partisipan setelah

memainkan *game*. ARCS merupakan metode penyimpulan yang digunakan untuk mengetahui persentase keberhasilan *game* dalam memperkenalkan binatang terhadap minat belajar, sedangkan skala likert merupakan metode penyimpulan aplikasi yang digunakan untuk mengetahui persentase keberhasilan aplikasi *game* terhadap evaluasi penilaian rancangan *user interface* dan navigasi *game*.

3.2 Struktur Game

Judul *game* : MY ZOO

Element *game digital* yang terdapat dalam *game* sebagai berikut.

1. Format : Negara yang terdapat dalam permainan ini memiliki tingkat *level* tersendiri. Setiap negara memiliki tingkat kesulitan dalam *rescue* binatang.
2. Peraturan
 - a. *Player* akan mendapatkan *coin* yang terbatas untuk *rescue* binatang, setiap binatang yang telah di-*rescue* akan menghasilkan *coin* bagi *player*.
 - b. *Player* harus me-*rescue* semua binatang yang terdapat pada setiap negara dan akan diberikan *grade* yang berfungsi untuk membuka negara berikutnya.
 - c. Setiap binatang memiliki *hungry time* yang berbeda. Setiap kali *hungry time* mencapai 0, maka binatang tersebut tidak dapat menghasilkan *coin* bagi *player*.
 - d. Binatang harus diberikan makanan yang benar karena pemberian makanan yang salah dapat berdampak pada pengurangan *hungry time*.
 - e. *Rescue* dibagi menjadi dua tipe, tipe satu untuk binatang yang berstatus *common* dan tipe dua berstatus *rare*.

- f. *Player* hanya diberikan satu kali kesempatan untuk memilih jawaban pada *rescue* tipe satu. *Rescue* tipe 2, *player* diberikan tujuh kali kesempatan untuk membuka kotak sampai membentuk pola binatang yang sedang di-*rescue*.
 - g. Satu kandang hanya dapat berisi lima hewan, jika *player* mendapatkan hewan yang sama, hewan tersebut tidak akan dimasukkan dalam kandang dan sebagai *reward* *player* mendapatkan *coin* dalam jumlah tertentu.
3. *Policy* : *player* dapat memutuskan untuk mendapatkan *grade* yang bagus pada setiap negara baru dilanjutkan ke negara lain, atau *player* cukup memenuhi *grade* yang dibutuhkan negara tersebut sebagai syarat untuk di-*unlock*.
 4. *Skenario* : Pada awal permainan *player* akan mendapatkan *coin* yang terbatas. Dengan *coin* terbatas *player* harus dapat mengembangkan semua *zoo* menjadi besar dan binatang yang berstatus *rare* dapat berubah menjadi *common*.
 5. *Event* : Dalam *rescue* tipe 2 ada saat *player* tidak dapat melihat pola binatang yang sedang di-*rescue*.
 6. *Roles* : *Player* dituntut harus benar dalam pemberian makanan pada setiap binatang, karena benar dan salah ada efek yang dapat dilihat oleh *player*.
 7. *Decisions* : Dalam permainan ini *player* akan banyak dihadapkan pada *popup* informasi untuk memilih keputusan. Sebelum memilih keputusan, *player* harus paham pada informasi yang telah diberikan.
 8. *Level* : Setiap negara memiliki tingkat kesulitan dalam me-*rescue* hewan yang berstatus *rare*.
 9. *Score Model* : Setiap *coin* yang dihasilkan ditampilkan dalam bentuk angka, untuk *grading* ditampilkan dengan alfabet.

10. *Indicator* : Player akan mendapatkan penilaian *grading* pada setiap *zoo* dan mendapatkan *reward* jika dapat me-rescue hewan *rare* sampai menjadi tidak langka lagi.

11. *Symbols* : Simbol yang digunakan dalam permainan dapat mewakili fungsinya sendiri, seperti simbol “X” untuk *close*, dan simbol makanan untuk menunjukkan hewan sedang lapor.

3.3 Penggunaan Asset

Berikut aset-aset yang digunakan dalam *game*.

Tabel 3.1 Daftar Aset Gambar

Gambar	Penjelasan	Sumber atau Pemilik
	Desain 2D bangunan berdasarkan ciri khas dari negara Australia. Digunakan untuk mempercantik tampilan pada bagian lokasi Australia.	Aset Pribadi
	Desain 2D bangunan berdasarkan ciri khas dari negara India. Digunakan untuk mempercantik tampilan pada bagian lokasi India.	Aset Pribadi
	Desain 2D bangunan berdasarkan ciri khas dari negara China. Digunakan untuk mempercantik tampilan pada bagian lokasi China.	Aset Pribadi

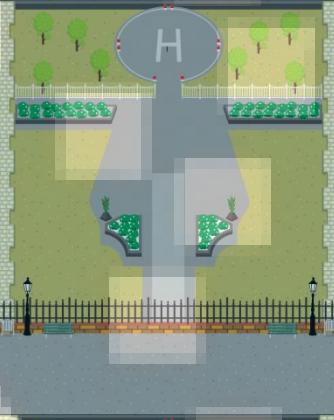
Tabel 3.1 Daftar Aset Gambar (Lanjutan)

Gambar	Penjelasan	Sumber atau Pemilik
	<p>Desain 2D bangunan berdasarkan ciri khas dari negara Thailand. Digunakan untuk mempercantik tampilan pada bagian lokasi Thailand.</p>	Aset Pribadi
	<p>Desain 2D bangunan berdasarkan ciri khas dari negara Mongolia. Digunakan untuk mempercantik tampilan pada bagian lokasi Mongolia.</p>	Aset Pribadi
	<p>Desain 2D bangunan berdasarkan ciri khas dari negara Filipina. Digunakan untuk mempercantik tampilan pada bagian lokasi Filipina.</p>	Aset Pribadi
	<p>Desain 2D bangunan berdasarkan ciri khas dari negara Srilanka. Digunakan untuk mempercantik tampilan pada bagian lokasi Srilanka.</p>	Aset Pribadi
	<p>Desain 2D bangunan berdasarkan ciri khas dari negara Bhutan. Digunakan untuk mempercantik tampilan pada bagian lokasi Bhutan.</p>	Aset Pribadi

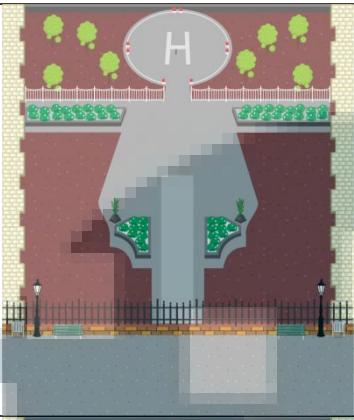
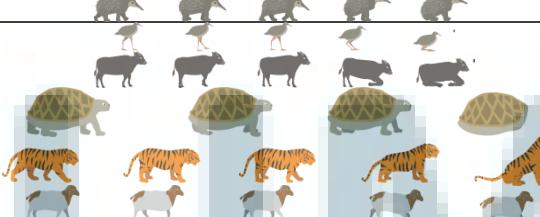
Tabel 3.1 Daftar Aset Gambar (Lanjutan)

Gambar	Penjelasan	Sumber atau Pemilik
	<p>Desain 2D bangunan berdasarkan ciri khas dari negara Indonesia. Digunakan untuk mempercantik tampilan pada bagian lokasi Indonesia.</p>	Aset Pribadi
	<p>Desain tampilan 2D, digunakan pada <i>swipe</i> pertama pada negara Australia.</p>	Aset Pribadi
	<p>Desain tampilan 2D, digunakan pada <i>swipe</i> pertama pada negara India.</p>	Aset Pribadi
	<p>Desain tampilan 2D, digunakan pada <i>swipe</i> pertama pada negara China.</p>	Aset Pribadi

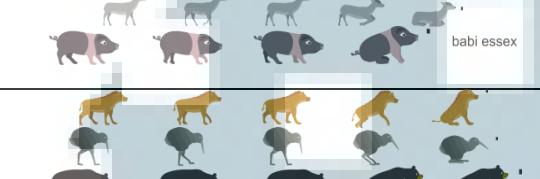
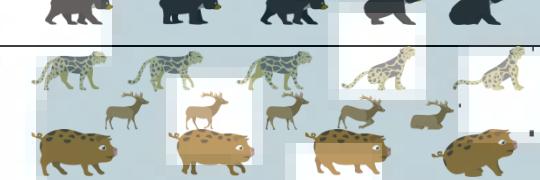
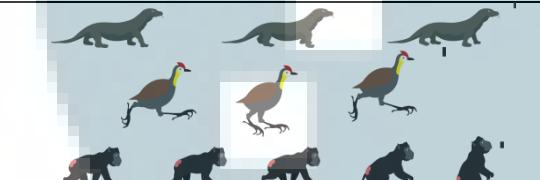
Tabel 3.1 Daftar Aset Gambar (Lanjutan)

Game	Penjelasan	Sumber atau Pemilik
	<p>Desain tampilan 2D, digunakan pada <i>swipe</i> pertama pada negara Thailand.</p>	Aset Pribadi
	<p>Desain tampilan 2D, digunakan pada <i>swipe</i> pertama pada negara Mongolia.</p>	Aset Pribadi
	<p>Desain tampilan 2D, digunakan pada <i>swipe</i> pertama pada negara Filipina.</p>	Aset Pribadi
	<p>Desain tampilan 2D, digunakan pada <i>swipe</i> pertama pada negara Srilanka</p>	Aset Pribadi

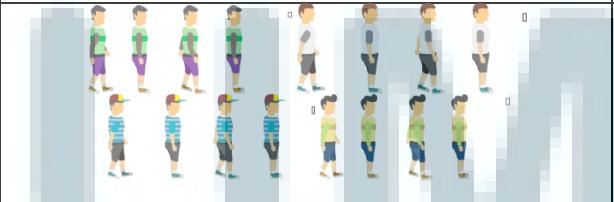
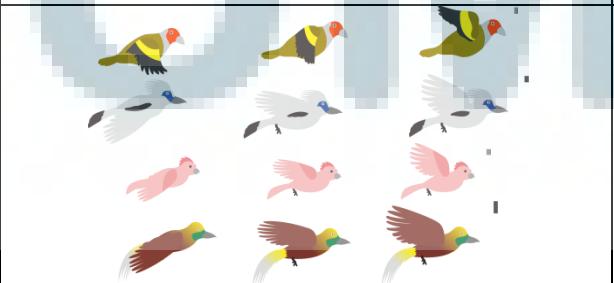
Tabel 3.1 Daftar Aset Gambar (Lanjutan)

Gambar	Penjelasan	Sumber atau Pemilik
	<p>Desain tampilan 2D, digunakan pada <i>swipe</i> pertama pada negara Bhutan.</p>	Aset Pribadi
	<p>Desain tampilan 2D, digunakan pada <i>swipe</i> pertama pada negara Indonesia.</p>	Aset Pribadi
	<p><i>Sprite</i> binatang yang digunakan pada negara Australia.</p>	Aset Pribadi
	<p><i>Sprite</i> binatang yang digunakan pada negara India.</p>	Aset Pribadi
	<p><i>Sprite</i> binatang yang digunakan pada negara China.</p>	Aset Pribadi
	<p><i>Sprite</i> binatang yang digunakan pada negara Thailand.</p>	Aset Pribadi

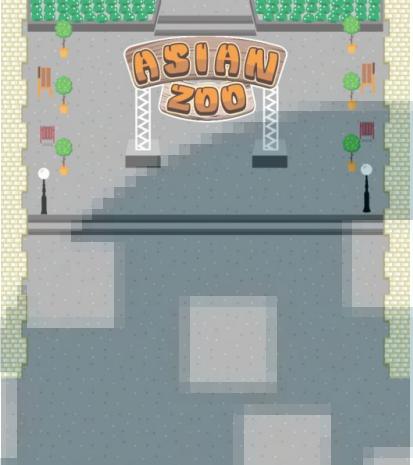
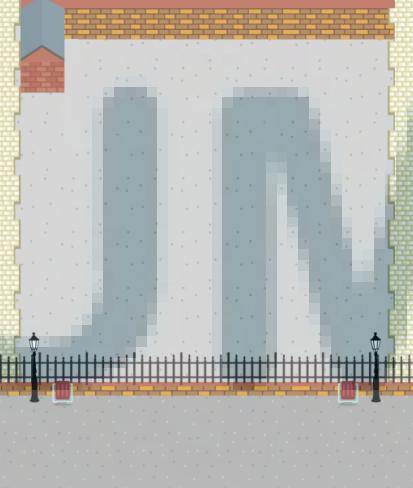
Tabel 3.1 Daftar Aset Gambar (Lanjutan)

Gambar	Penjelasan	Sumber atau Pemilik
	<i>Sprite</i> binatang yang digunakan pada negara Mongolia.	Aset Pribadi
	<i>Sprite</i> binatang yang digunakan pada negara Filipina.	Aset Pribadi
	<i>Sprite</i> binatang yang digunakan pada negara Srilanka.	Aset Pribadi
	<i>Sprite</i> binatang yang digunakan pada negara Bhutan.	Aset Pribadi
	<i>Sprite</i> binatang yang digunakan pada negara Indonesia.	Aset Pribadi
	<i>Sprite</i> Heli, digunakan saat perpindahan lokasi.	Aset Pribadi
	Desain <i>sprite</i> makanan yang digunakan pada saat pemberian makan pada hewan.	Aset Pribadi

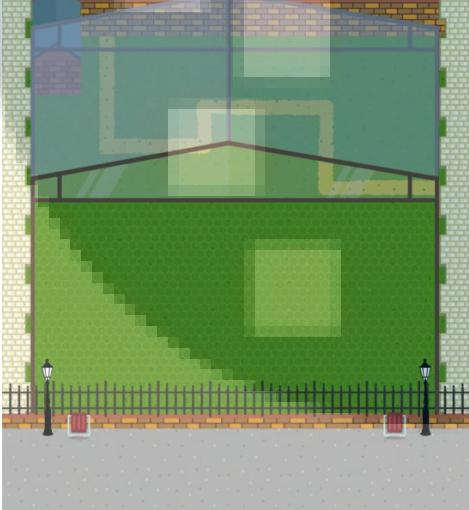
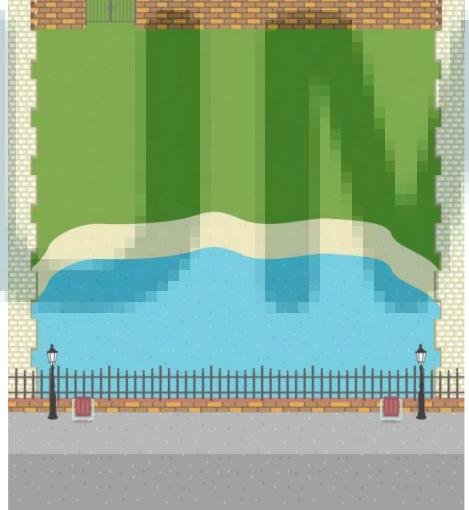
Tabel 3.1 Daftar Aset Gambar (Lanjutan)

Gambar	Penjelasan	Sumber atau Pemilik
	<p>Desain <i>sprite</i> makanan yang digunakan pada saat pemberian makan pada hewan.</p>	Aset Pribadi
	<p><i>Sprite</i> zoo keeper digunakan saat memberi makanan pada hewan.</p>	Aset Pribadi
	<p><i>Sprite</i> pengunjung digunakan untuk meramaikan suasana kebun binatang.</p>	Aset Pribadi
	<p><i>Sprite</i> burung yang akan digunakan pada <i>mini game</i>.</p>	Aset Pribadi

Tabel 3.1 Daftar Aset Gambar (Lanjutan)

Gambar	Penjelasan	Sumber atau Pemilik
	Desain tampilan 2D, saat <i>view</i> berada pada <i>lobby</i> .	Aset Pribadi
	Desain tampilan 2D untuk kandang yang banyak rumput.	Aset Pribadi
	Desain tampilan 2D untuk kandang yang bermusim dingin.	Aset Pribadi

Tabel 3.1 Daftar Aset Gambar (Lanjutan)

Gambar	Penjelasan	Sumber atau Pemilik
	Desain tampilan 2D saat kandang sedang dibangun.	Aset Pribadi
	Desain tampilan 2D untuk kandang burung.	Aset Pribadi
	Desain tampilan 2D untuk kandang yang terdapat kolam.	Aset Pribadi

Tabel 3.1 Daftar Aset Gambar (Lanjutan)

Gambar	Penjelasan	Sumber atau Pemilik
  	<p>Foto hewan yang terdapat di negara Australia. Foto digunakan sebagai detail deskripsi pada game.</p>	a-z-animals.com
  	<p>Foto hewan yang terdapat di negara India. Foto digunakan sebagai detail deskripsi pada game.</p>	a-z-animals.com

Tabel 3.1 Daftar Aset Gambar (Lanjutan)

Gambar	Penjelasan	Sumber atau Pemilik
  	<p>Foto hewan yang terdapat di negara China. Foto digunakan sebagai detail deskripsi pada game.</p>	a-z-animals.com
  	<p>Foto hewan yang terdapat di negara Thailand. Foto digunakan sebagai detail deskripsi pada game.</p>	a-z-animals.com
  	<p>Foto hewan yang terdapat di negara Mongolia. Foto digunakan sebagai detail deskripsi pada game.</p>	a-z-animals.com
  	<p>Foto hewan yang terdapat di negara Filipina. Foto digunakan sebagai detail deskripsi pada game.</p>	a-z-animals.com

Tabel 3.1 Daftar Aset Gambar (Lanjutan)

Gambar	Penjelasan	Sumber atau Pemilik
	<p>Foto hewan yang terdapat di negara Srilanka. Foto digunakan sebagai detail deskripsi pada game.</p>	a-z-animals.com
	<p>Foto hewan yang terdapat di negara Bhutan. Foto digunakan sebagai detail deskripsi pada game.</p>	a-z-animals.com
	<p>Foto hewan yang terdapat di negara Indonesia. Foto digunakan sebagai detail deskripsi pada game.</p>	a-z-animals.com

Tabel 3.2 Daftar Aset Audio

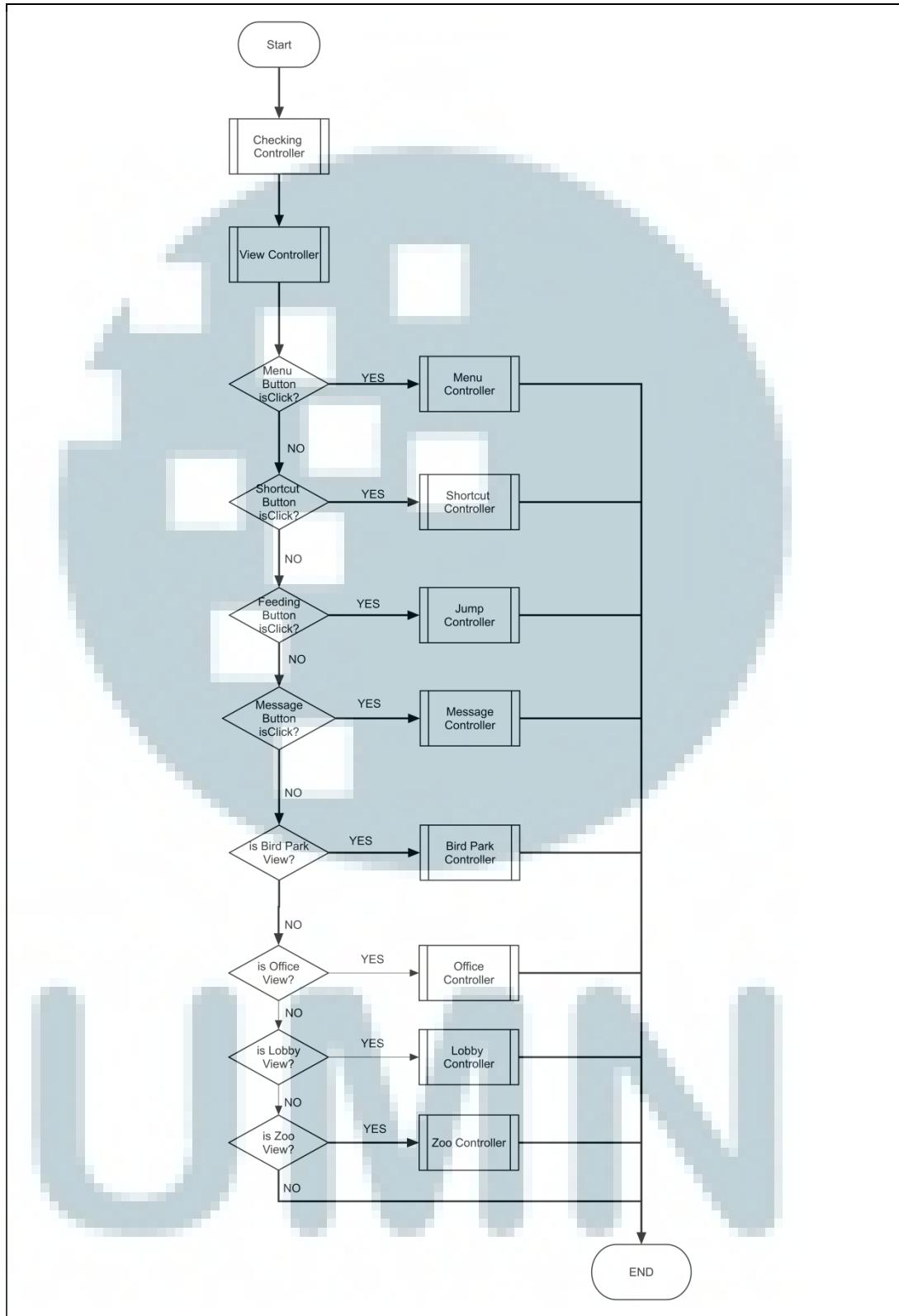
Nama	Penjelasan	Sumber atau Pemilik
Music_01.mp3	Music_01.mp3 digunakan sebagai <i>backsound</i> negara Australia.	Free Music Archive
Music_02.mp3	Music_02.mp3 digunakan sebagai <i>backsound</i> negara India.	Free Music Archive
Music_03.mp3	Music_03.mp3 digunakan sebagai <i>backsound</i> negara China.	Free Music Archive
Music_04.mp3	Music_04.mp3 digunakan sebagai <i>backsound</i> negara Thailand.	Free Music Archive
Music_05.mp3	Music_05.mp3 digunakan sebagai <i>backsound</i> negara Mongolia.	Free Music Archive
Music_06.mp3	Music_06.mp3 digunakan sebagai <i>backsound</i> negara Filipina.	Free Music Archive
Music_07.mp3	Music_07.mp3 digunakan sebagai <i>backsound</i> negara Srilanka.	Free Music Archive
Music_08.mp3	Music_08.mp3 digunakan sebagai <i>backsound</i> negara Bhutan.	Free Music Archive
Music_09.mp3	Music_09.mp3 digunakan sebagai <i>backsound</i> negara Indonesia.	Free Music Archive
Correct.mp3	Correct.mp3 digunakan untuk <i>sound</i> jika user memilih jawaban yang benar.	Freesound.org
Crash.aif	Crash.aif digunakan untuk <i>sound</i> ketika burung menyentuh <i>collider</i> .	Freesound.org

Tabel 3.2 Daftar Aset Audio (Lanjutan)

Nama	Penjelasan	Sumber atau Pemilik
006-System06.ogg	006-System06.ogg digunakan untuk <i>sound</i> ketika <i>user</i> membeli item.	RPG Maker MV
032-Switch01.ogg	032-Switch01.ogg digunakan untuk <i>sound</i> ketika <i>user</i> menekan tombol <i>close</i> .	RPG Maker MV
Miss.ogg	Miss.ogg digunakan untuk <i>sound</i> ketika <i>user</i> membeli <i>item</i> tetapi <i>coin</i> tidak mencukupi.	RPG Maker MV
Coin_01.wav	Coin_01.wav digunakan untuk <i>sound</i> ketika burung menyentuh <i>object coin</i> .	Soundsnap.com
Coin_02.wav	Coin_02.wav digunakan untuk <i>sound</i> ketika <i>user</i> mendapatkan <i>coin</i> tambahan.	Soundsnap.com
Coin_03.wav	Coin_03.wav digunakan untuk <i>sound</i> ketika pengurangan <i>coin user</i> .	Soundsnap.com
Fly.wav	Fly.wav digunakan untuk <i>sound</i> terbang pada burung.	Soundsnap.com

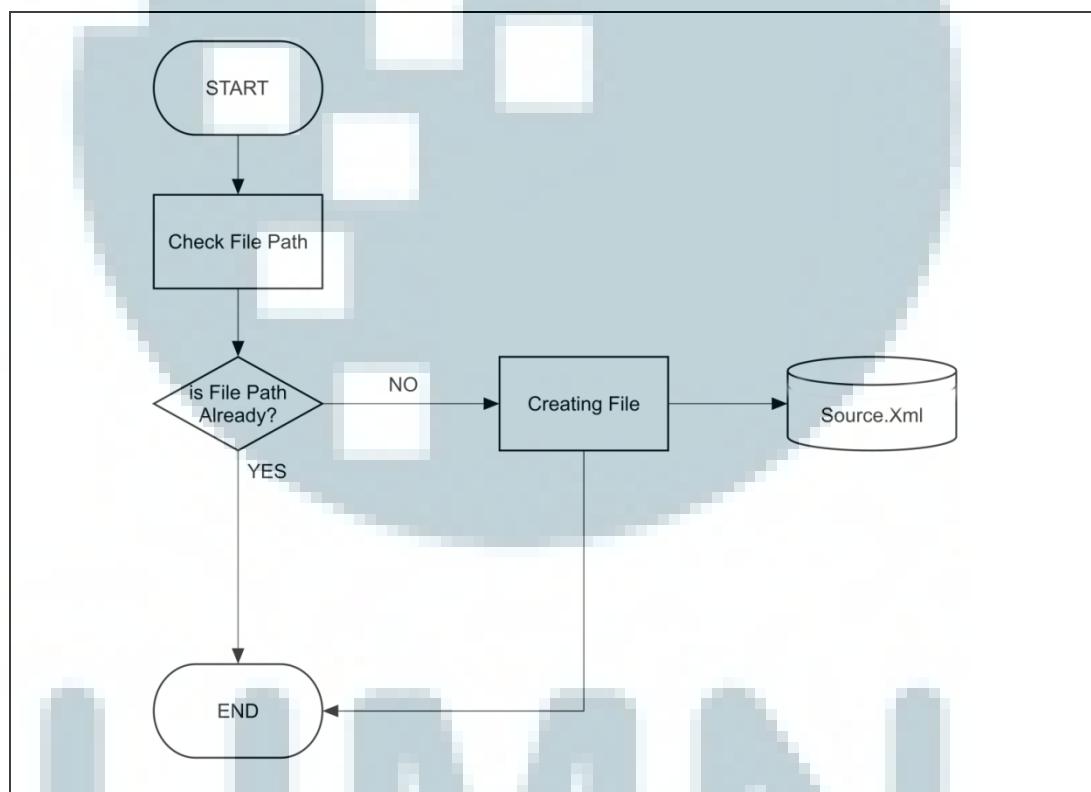


3.4 Perancangan Sistem



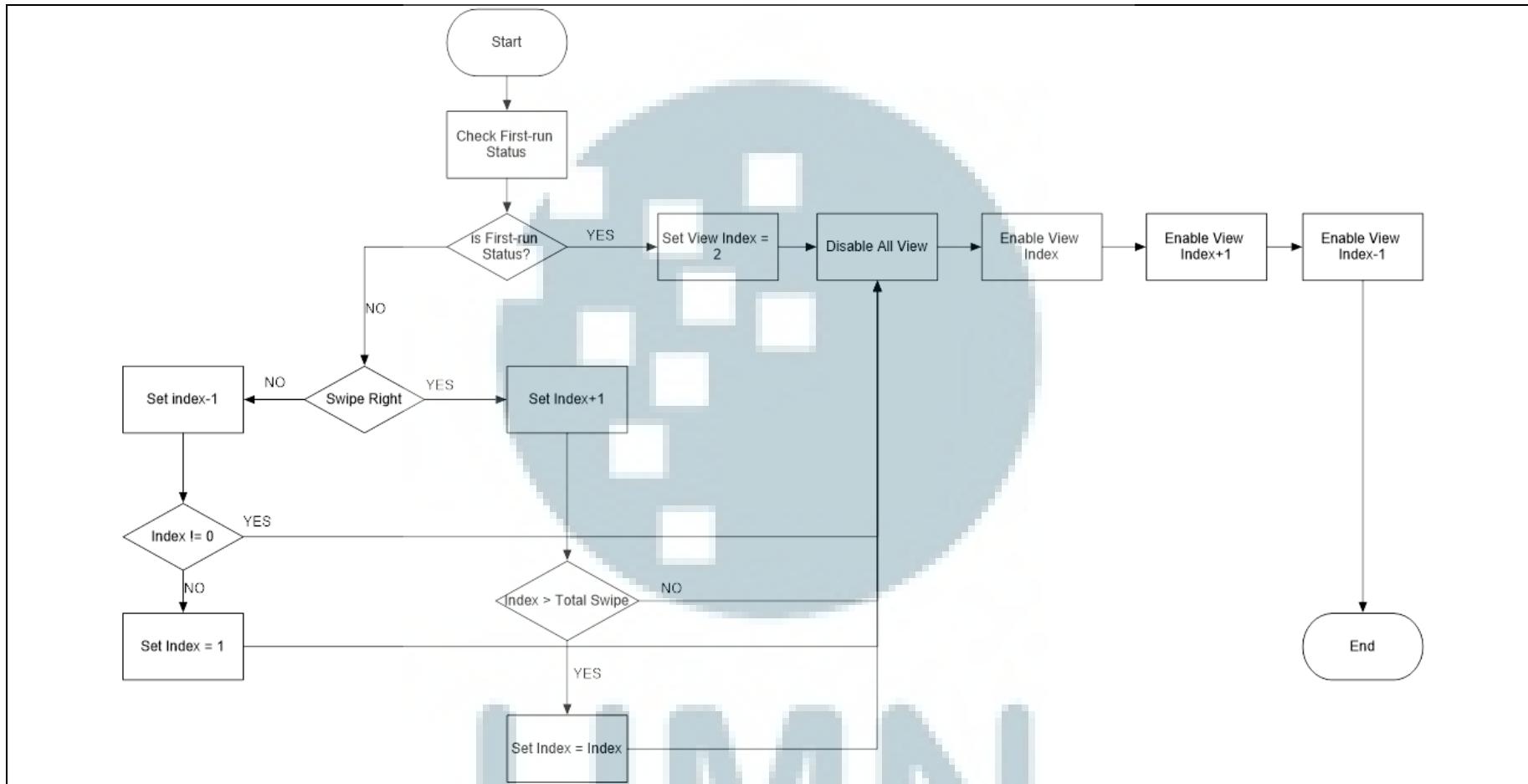
Gambar 3.1 Flowchart Main Menu

Perancangan *flowchart* dibuat untuk memperlihatkan alur proses yang terdapat dalam *game*. Gambar 3.1 merupakan *flowchart* umum *game*. Saat *game* dimulai, program akan langsung melakukan *check file*, jika *file XML* tidak ada, maka akan langsung dibuat semua *XML*. Proses selanjutnya memanggil *view controller* yang secara garis besar berfungsi mengatur panel yang aktif dan mati. Setelah itu sistem akan melakukan *checking* apakah ada tombol yang ditekan. Jika ada tombol yang ditekan, sistem akan langsung memanggil *controller* yang menangani tombol-tombol tersebut.



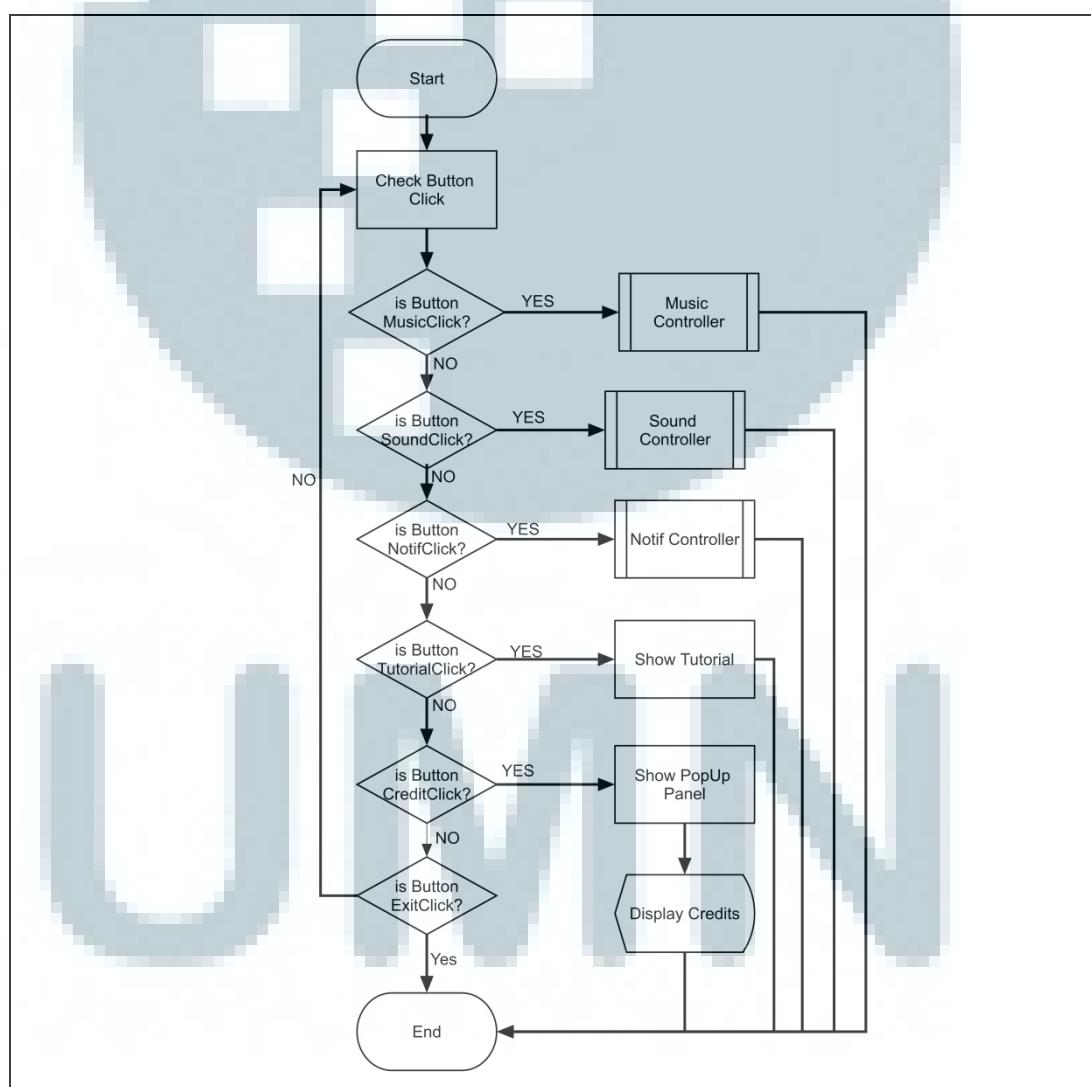
Gambar 3.2 *Flowchart Checking Controller*

Gambar 3.2 adalah *flowchart* yang menunjukkan saat *game* pertama kali dimainkan akan dilakukan pengecekan *file-file XML*. Jika *file XML* tidak ada maka, akan otomatis langsung membuat semua *file XML* dan jika *file XML* telah ada, maka proses *checking* akan langsung selesai.



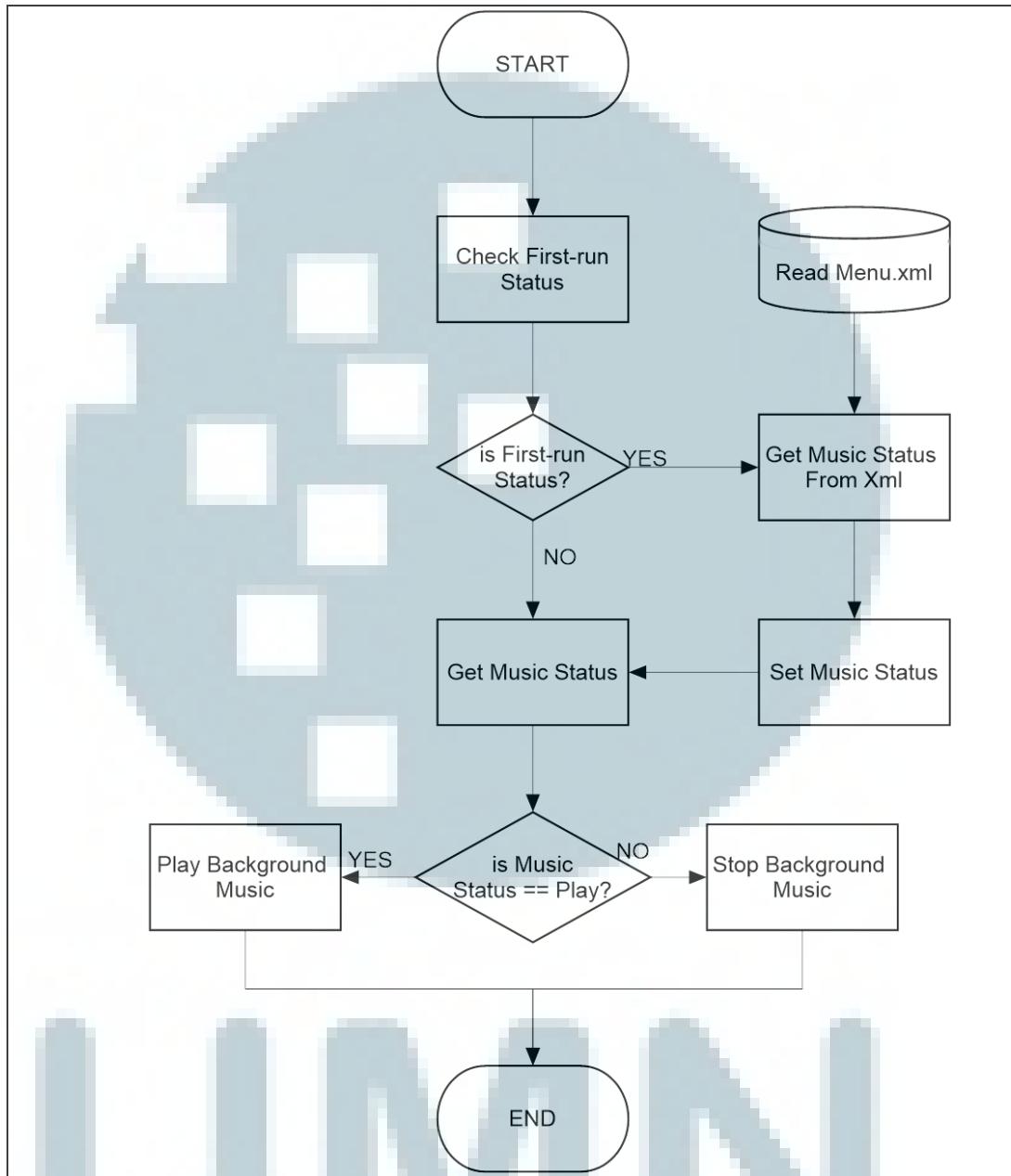
Gambar 3.3 *Flowchart View Controll*

Gambar 3.3 menunjukkan gambaran dari *flowchart view controller* yang berfungsi mengatur agar *view panel* yang aktif selalu tiga buah untuk mengurangi *lag* yang disebabkan oleh *collider* binatang. Saat pertama *game* dijalankan *view* akan selalu di-set pada panel 2, pemain dapat melakukan *swipe* ke kanan ataupun kiri. Setiap *swipe* arah kiri akan dibatasi, pemain tidak dapat melakukan *swipe* ke kandang binatang jika belum ada binatang yang diselamatkan. Jika pemain memaksa *swipe* ke *maximal swipe* kiri, maka pemain akan langsung dikembalikan ke *view* sebelumnya dan memunculkan peringatan.



Gambar 3.4 *Flowchart Menu Controller*

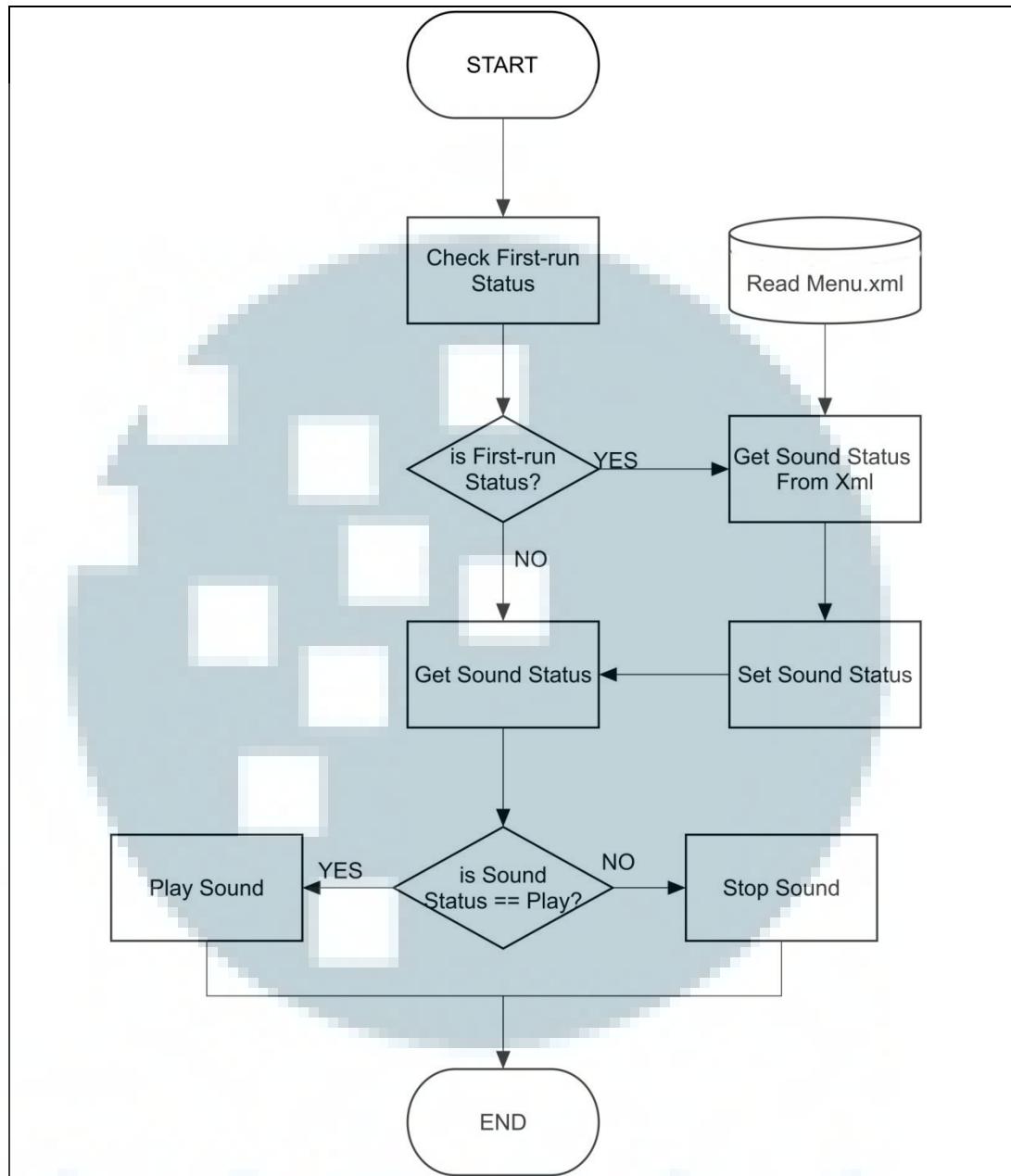
Gambar 3.4 menunjukkan *flowchart menu controller* yang berisi pengaturan pada musik, *sound*, *notification*, *tutorial*, *credits*, dan tombol keluar dari *game*.



Gambar 3.5 *Flowchart Music Controller*

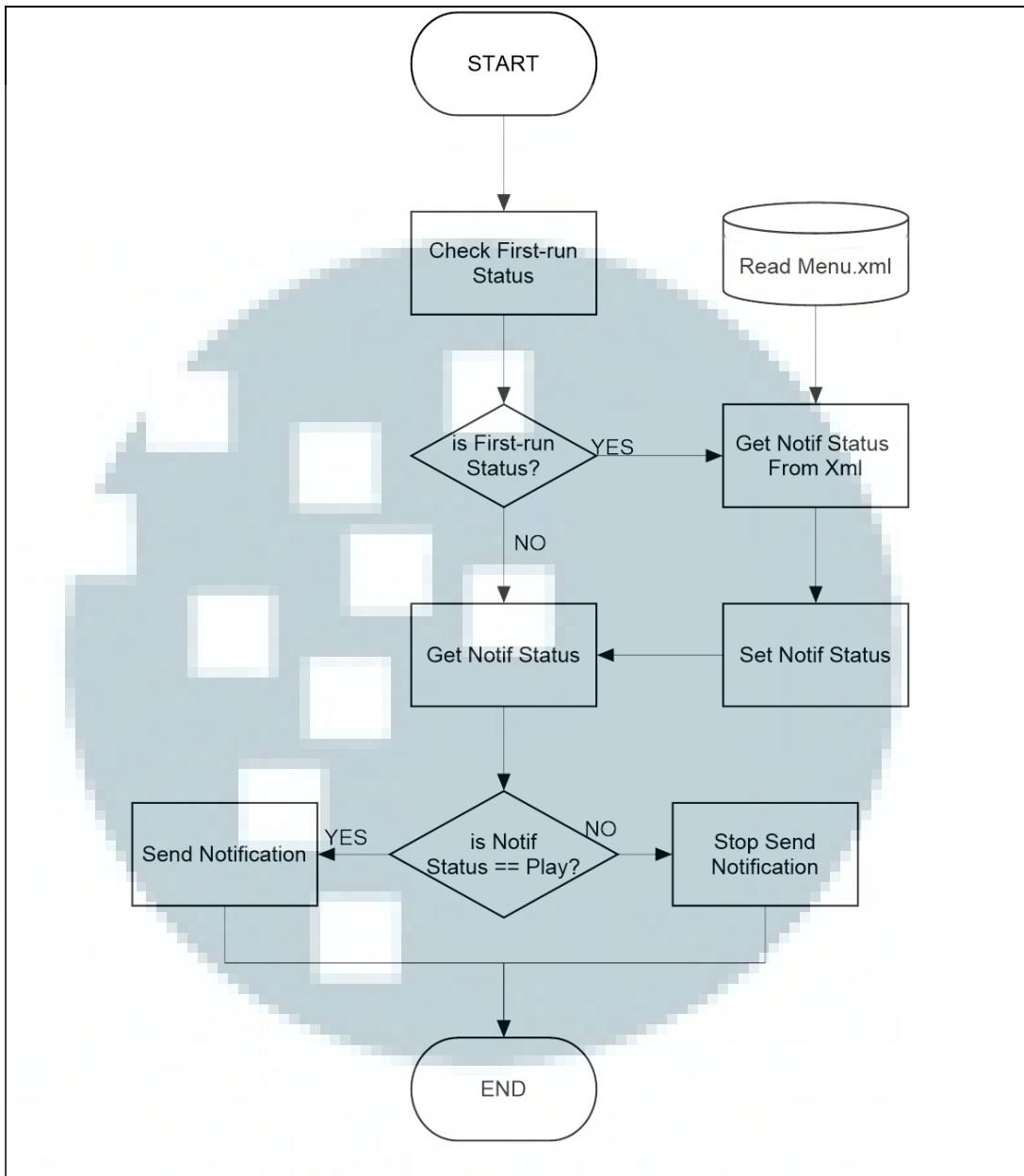
Gambar 3.5 menunjukkan proses pengaturan musik dalam permainan.

Sistem akan mengambil musik status dari XML file, dan langsung akan di-set sesuai status yang didapat dari XML file. Jika status *play* maka akan langsung memainkan musik, dan sebaliknya akan memberhentikan musik jika status *stop*.



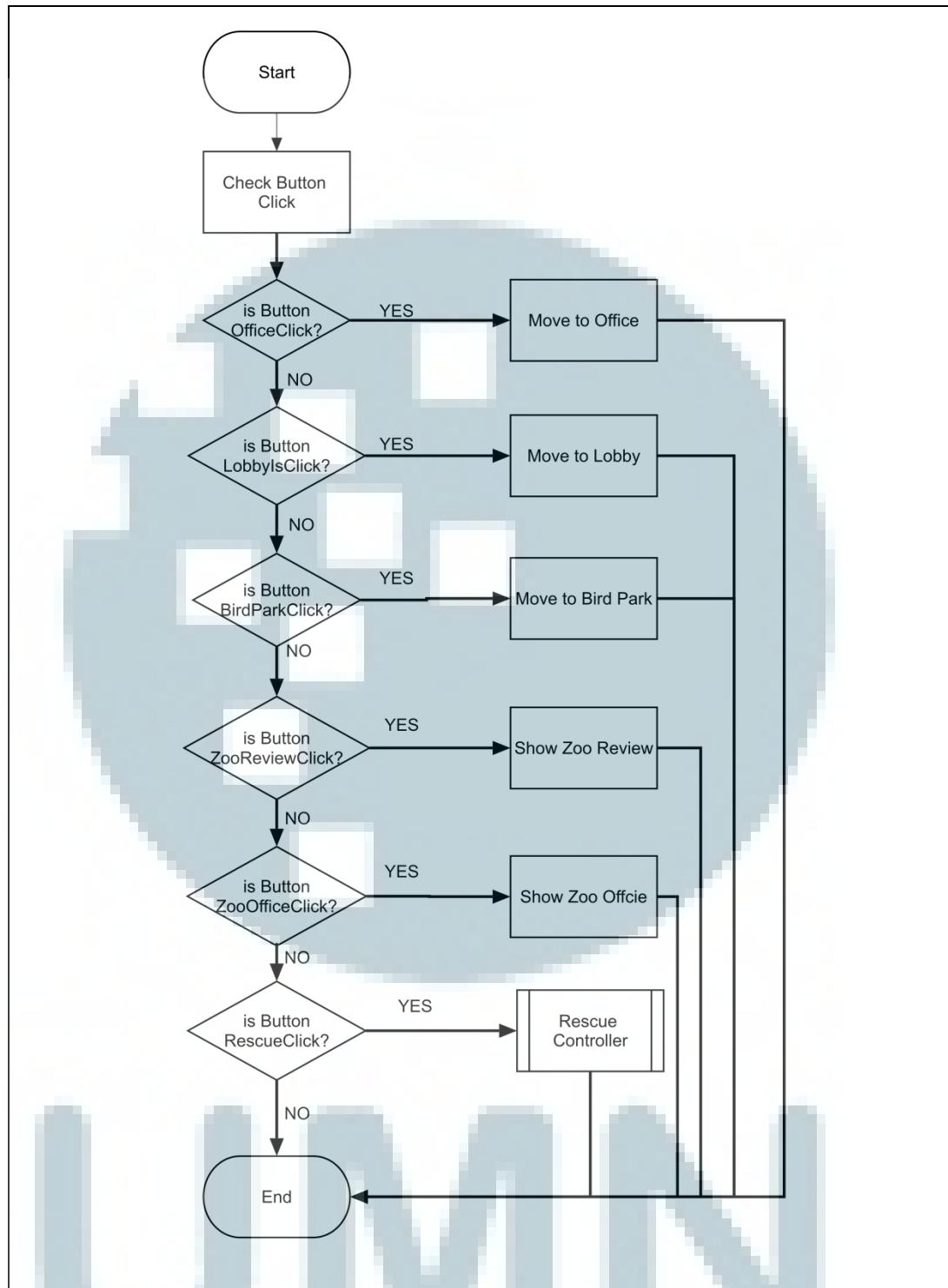
Gambar 3.6 *Flowchart Sound Controller*

Gambar 3.6 menunjukkan proses pengaturan *sound* dalam permainan. Sistem akan mengambil *sound status* dari XML file, dan langsung akan di-set sesuai status yang didapat dari XML file. Jika status *play* maka setiap tombol yang ditekan pemain akan memainkan *sound* dan sebaliknya jika *stop* maka tombol yang ditekan pemain tidak akan berbunyi.



Gambar 3.7 Flowchart Notification Controller

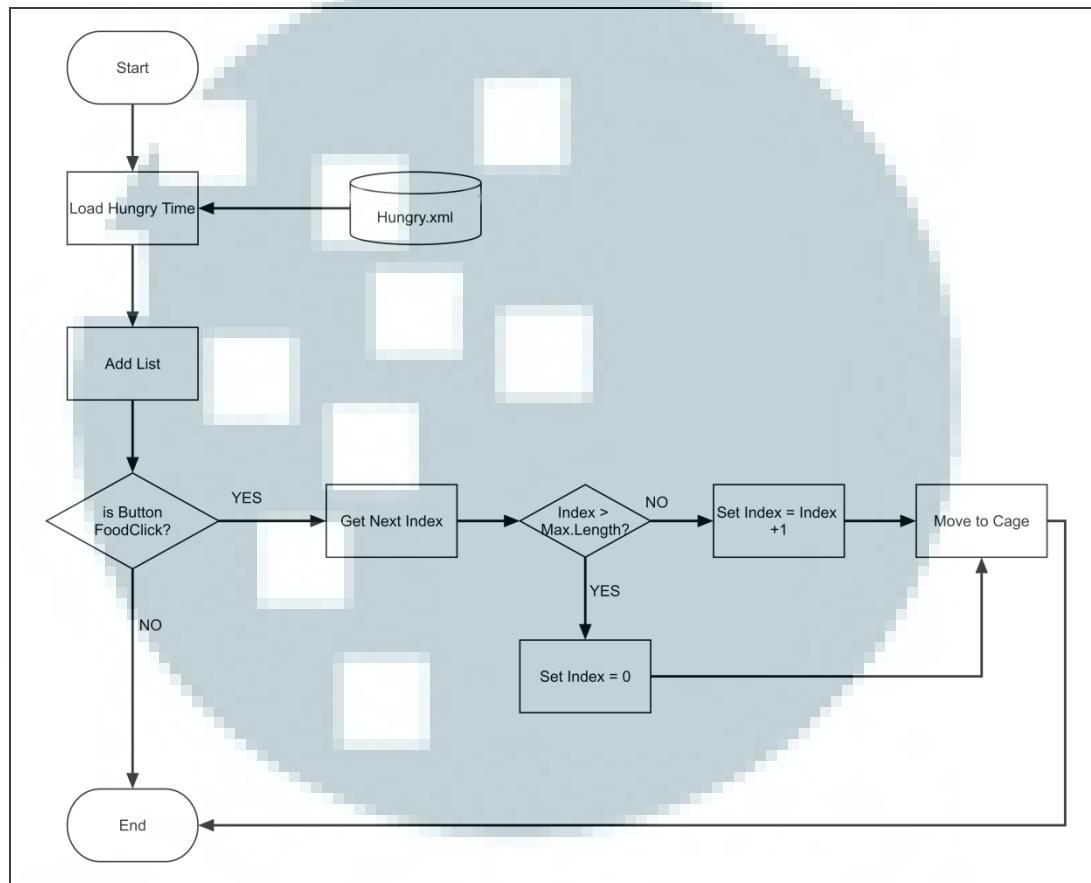
Gambar 3.7 menunjukkan flowchart *Notification Controller* yang berfungsi untuk mengatur *game* mengirimkan *notification* getar dan bunyi ke *smartphone* atau tidak. Pemain dapat mematikan *notification* dengan cara menekan tombol *off*, maka *game* akan berhenti mengirimkan *notification* jika ada pesan masuk. *Notification* akan selalu *off* sampai pemain mengubah menjadi *on*.



Gambar 3.8 Flowchart Shortcut Controller

Jika *shortcut* ditekan *player* akan melihat *popup* yang berisi tombol-tombol yang dapat mempermudah pemain. Seperti yang ditunjukkan Gambar 3.8 jika pemain menekan tombol *office* maka pemain akan dipindahkan pada *panel*

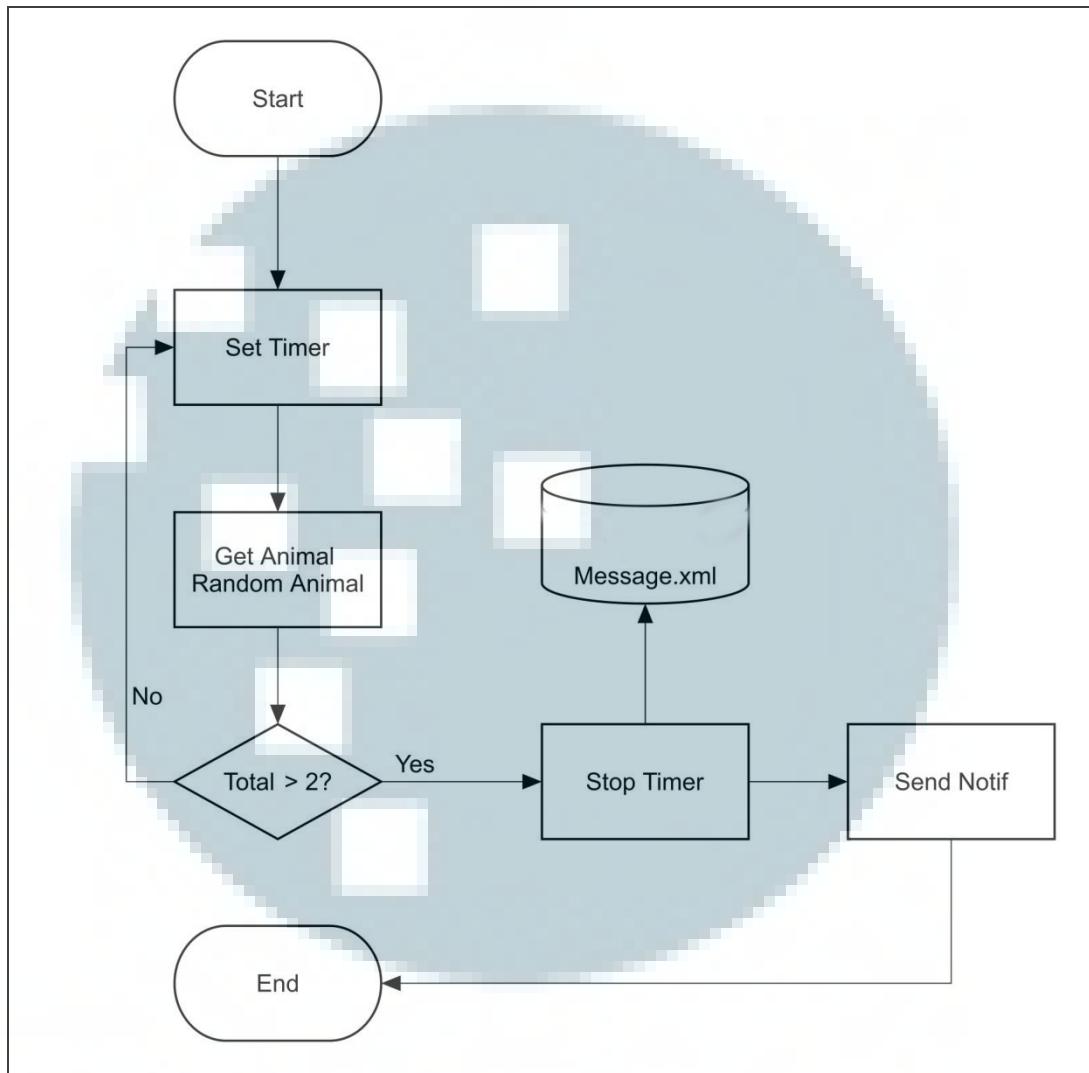
office, tombol *lobby* akan membawa pemain ke *panel lobby*, tombol *zoo review* akan meminta sistem untuk memunculkan *popup grading* pada *zoo*, *zoo office* akan meminta sistem memunculkan *popup* semua informasi binatang pada setiap negara, dan *rescue* akan meminta sistem untuk memanggil *rescue scene*.



Gambar 3.9 *Flowchart Jump Controller*

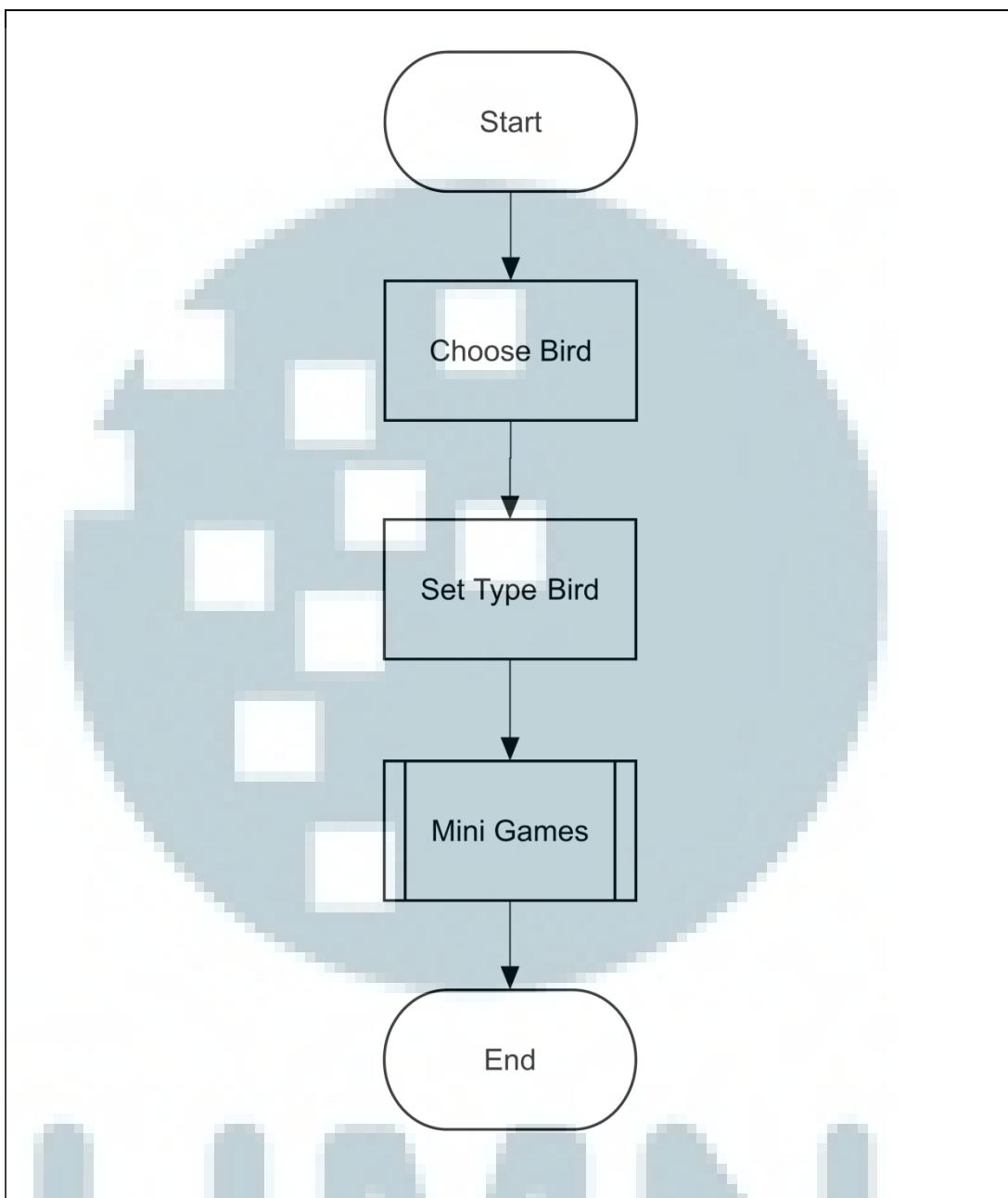
Gambar 3.9 menunjukkan *flowchart jump controller* yang berfungsi membantu pemain mempermudah menuju kandang binatang yang sedang lapar. Pertama sistem akan mengambil data dari *file XML* dan langsung dimasukkan ke *list*, *list* selalu diperbarui jika ada binatang yang lapar. Jika pemain menekan tombol tersebut, sistem akan mengambil *index* awal dan melakukan *checking* apakah *index* melebihi jumlah *index*, jika tidak pemain akan langsung pindah ke

kandang yang ditunjukkan *index*. *Index* yang melewati jumlah *index* maka *index* akan di-set ulang ke *index* ke-0.



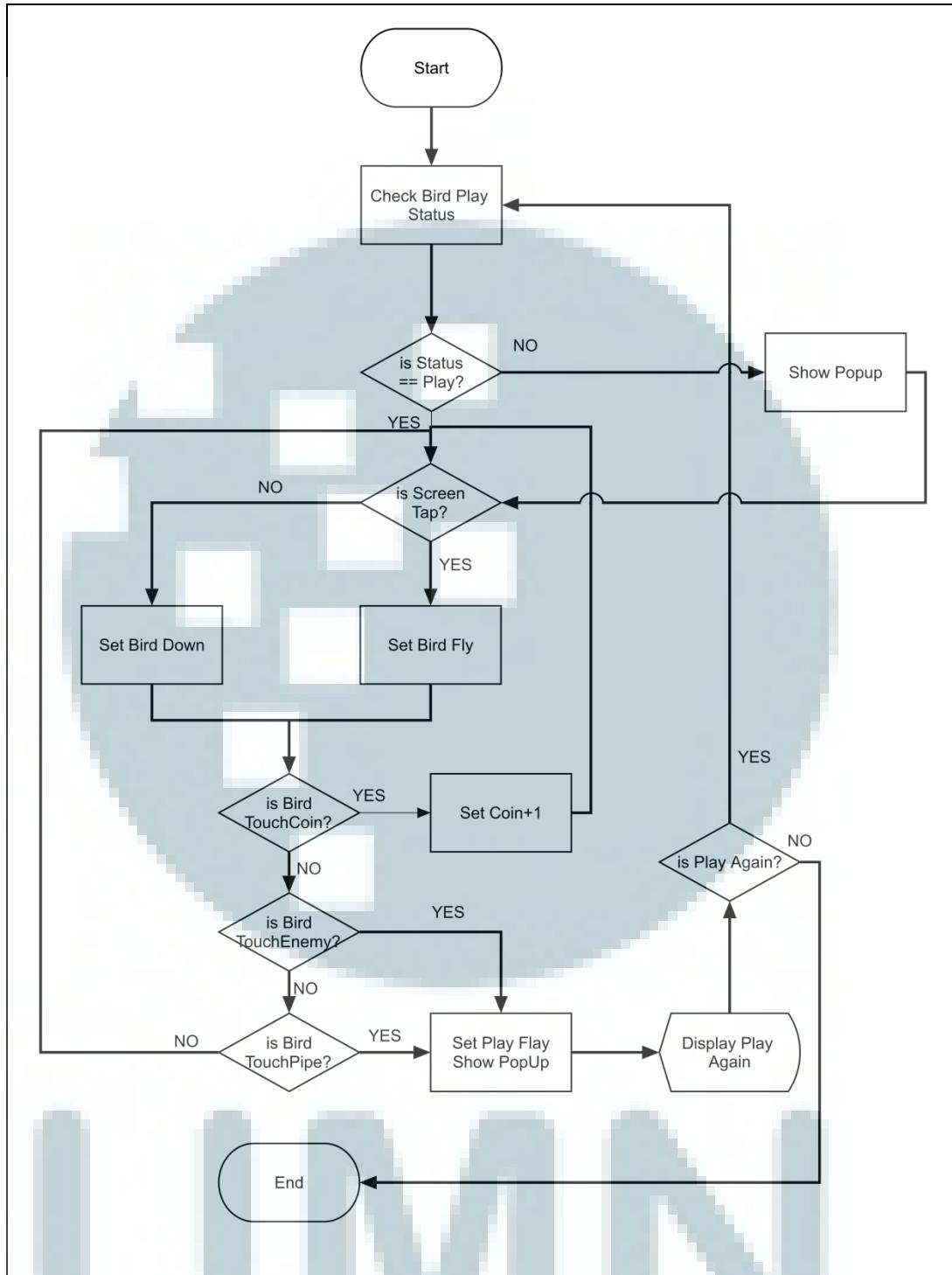
Gambar 3.10 Flowchart Message Controller

Gambar 3.10 adalah gambaran dari *flowchart Message Controller*, saat pertama dijalankan *timer* akan selalu aktif. Sistem akan melakukan pencarian pada setiap negara yang sudah dibeli dan memiliki binatang yang lebih dari satu. Saat sistem telah menemukan binatangnya, *timer* akan diberhentikan dan meng-update data ke *message.xml* yang berfungsi untuk menyimpan pesan agar tetap ada saat *game* dibuka kembali, lalu sistem akan mengirimkan *notif* ke *smartpone*.



Gambar 3.11 *Flowchart Bird Park Controller*

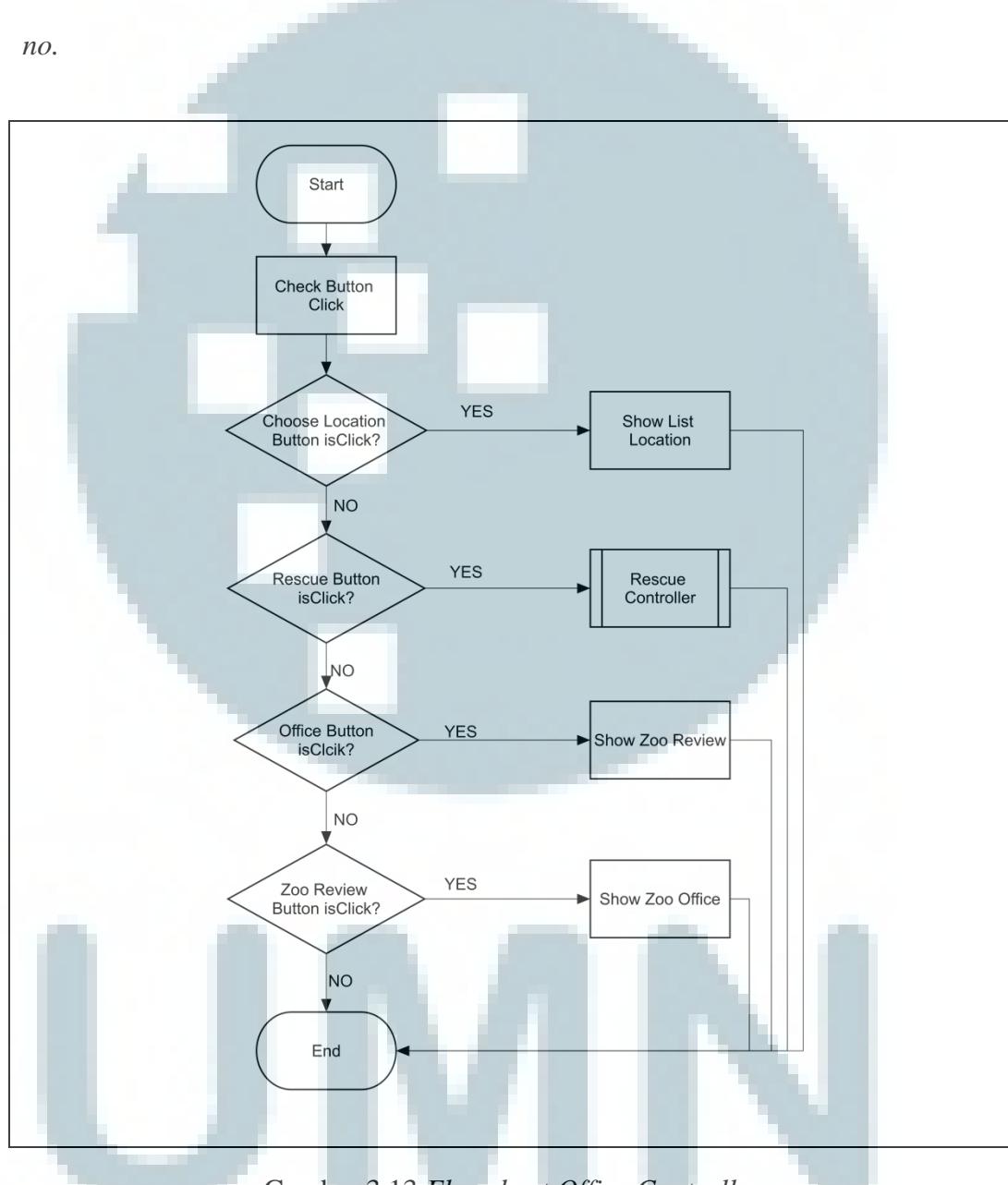
Gambar 3.11 adalah proses yang akan terjadi jika pemain sedang berada pada *panel bird park*. Pada *panel* tersebut pemain dapat memilih burung yang akan digunakan untuk menjadi *character* pada *mini games*.



Gambar 3.12 Flowchart Mini Games

Gambar 3.12 adalah proses yang terjadi di *mini games*, saat pertama mulai masuk pada *mini game scene* status *play* adalah *no* untuk membantu pemain agar burung tidak langsung terjatuh dan mati. Saat ada *input* berupa tap, *game* akan

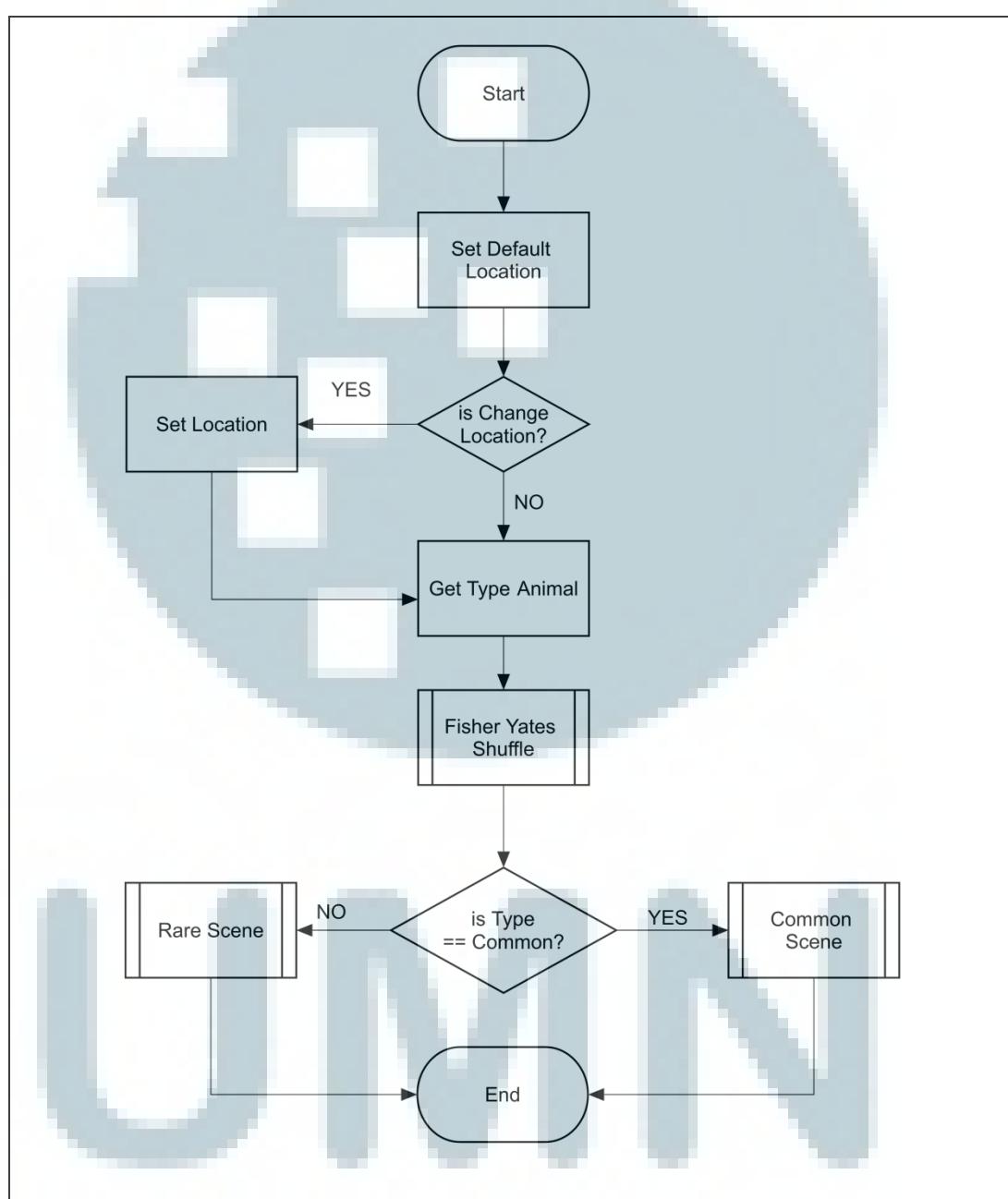
dimulai, setiap *tap* akan membuat burung terbang ke atas sebaliknya jika pemain tidak melakukan *tap* maka burung akan terjatuh. Jika burung menyentuh *coin* maka *coin* akan selalu ditambah sampai pemain menyentuh *enemy* atau *pipe* yang akan membuatnya mati. Jika terjadi *game over*, status *play* akan langsung di-set *no*.



Gambar 3.13 Flowchart Office Controller

Gambar 3.13 menunjukkan proses yang terjadi jika pemain berada pada *panel office*. Pada *panel office* terdapat *choose location* yang akan memunculkan *popup* berupa lokasi yang telah dibeli pemain dan dapat memilihnya untuk

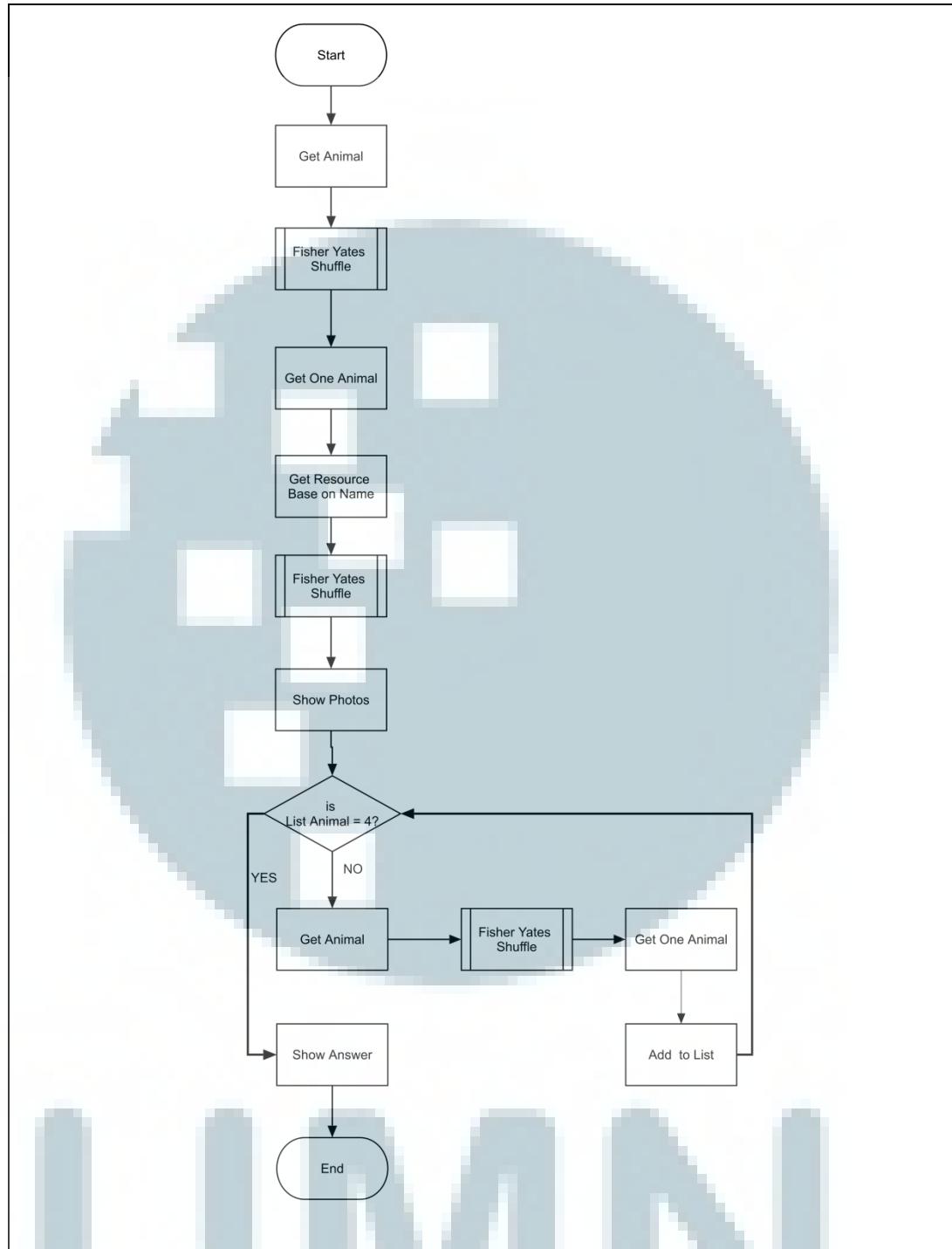
menyelamatkan binatang pada lokasi tersebut. Tombol *rescue* akan memanggil *scene rescue controller*. Tombol *office* akan memanggil *popup* yang berisi informasi *grading zoo* setiap negara, *zoo review* akan memanggil *popup* yang menampilkan informasi data semua binatang yang telah didapat.



Gambar 3.14 Flowchart Rescue Controller

Gambar 3.14 adalah gambaran dari proses penentuan binatang dengan tipe tertentu yang akan didapat. Lokasi *default* adalah lokasi yang sudah dipilih sebelumnya, jika terjadi perubahan lokasi maka lokasi *default* akan di-set ulang. Jika *rescue* dilakukan akan langsung masuk pada *checking* status, status *common* akan memanggil *common scene* dan akan memulai *rescue* pada binatang yang berstatus *common*, sebaliknya jika *rare* maka akan memanggil *rare scene*.

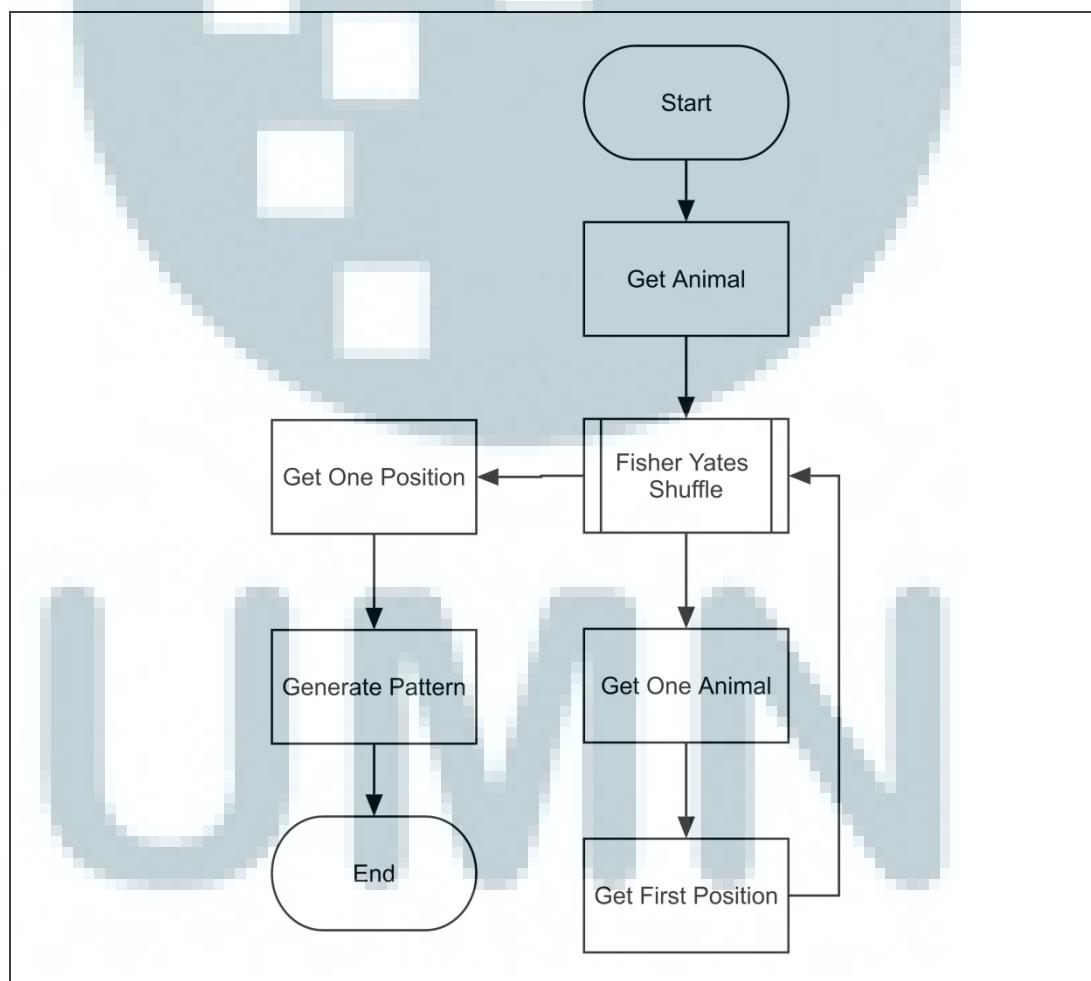




Gambar 3.15 Flowchart Common Scene

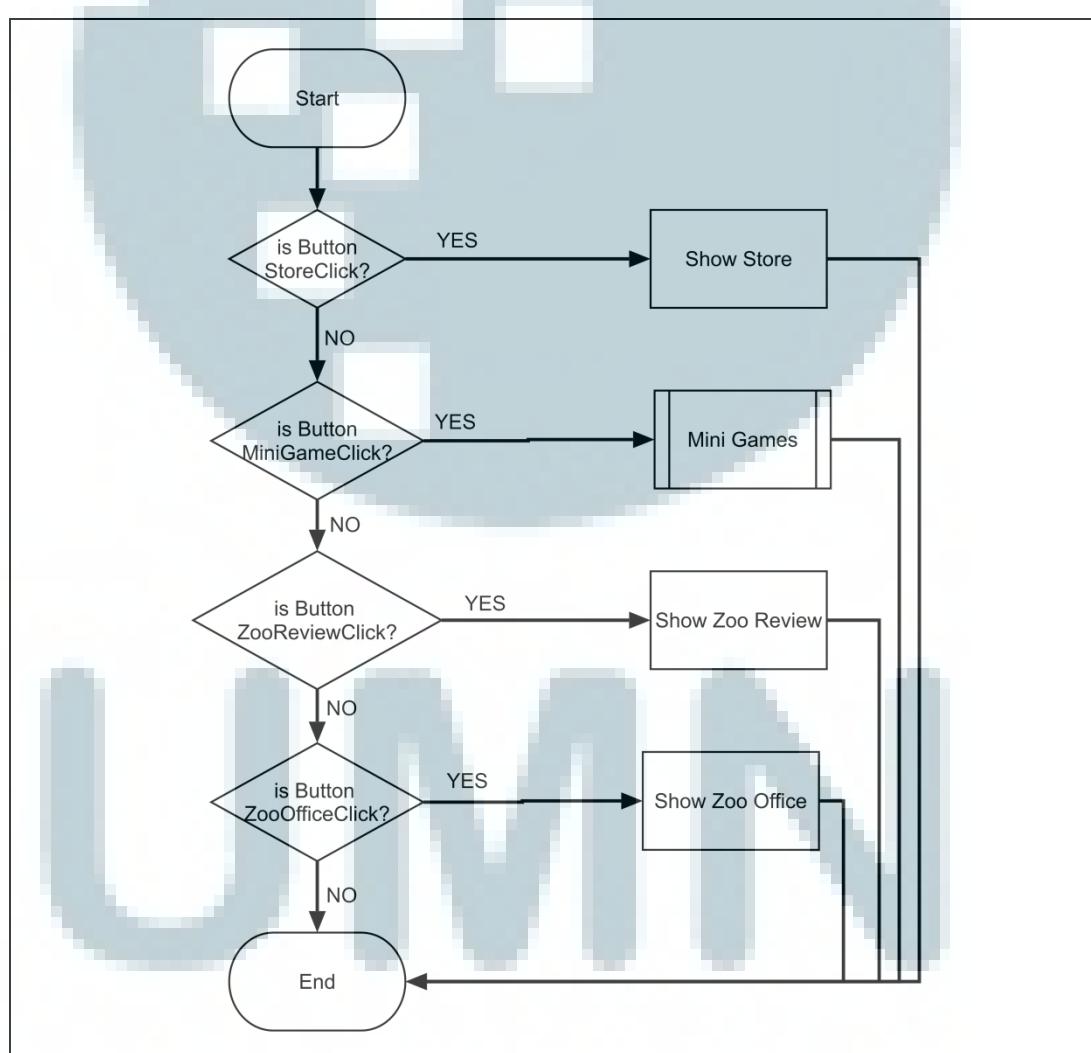
Gambar 3.15 menunjukkan proses pemilihan binatang, pengacakan foto, dan jawaban. Saat mulai dengan *rescue* tipe *common* sistem akan mengirimkan *list* binatang pada proses *shuffle fisher-yates*. Setelah dilakukan *shuffling* akan dipilih satu binatang yang akan didapatkan pemain. Nama binatang yang didapat

akan dijadikan *index* untuk mengambil *sprite resource* foto dan mengirim kumpulan *sprite* tersebut pada *shuffle fisher-yates*. Hasil dari foto yang telah di *shuffle* langsung ditampilkan. Proses berikutnya adalah pembentukan jawaban, yang dibentuk dari mengambil *list* binatang lalu dikirimkan pada *shuffle fisher-yates*. Setelah di *shuffle* dipilih satu nama hewan dan langsung dimasukkan pada *list*. *List* pertama yang ditambahkan adalah binatang yang didapat saat awal, *list* berikutnya adalah *list* nama binatang yang salah. *List* belum mencapai jumlah 4 maka akan terus dilakukan pengambilan nama binatang. Saat *list* telah mencapai jumlah 4 jawaban akan ditampilkan.



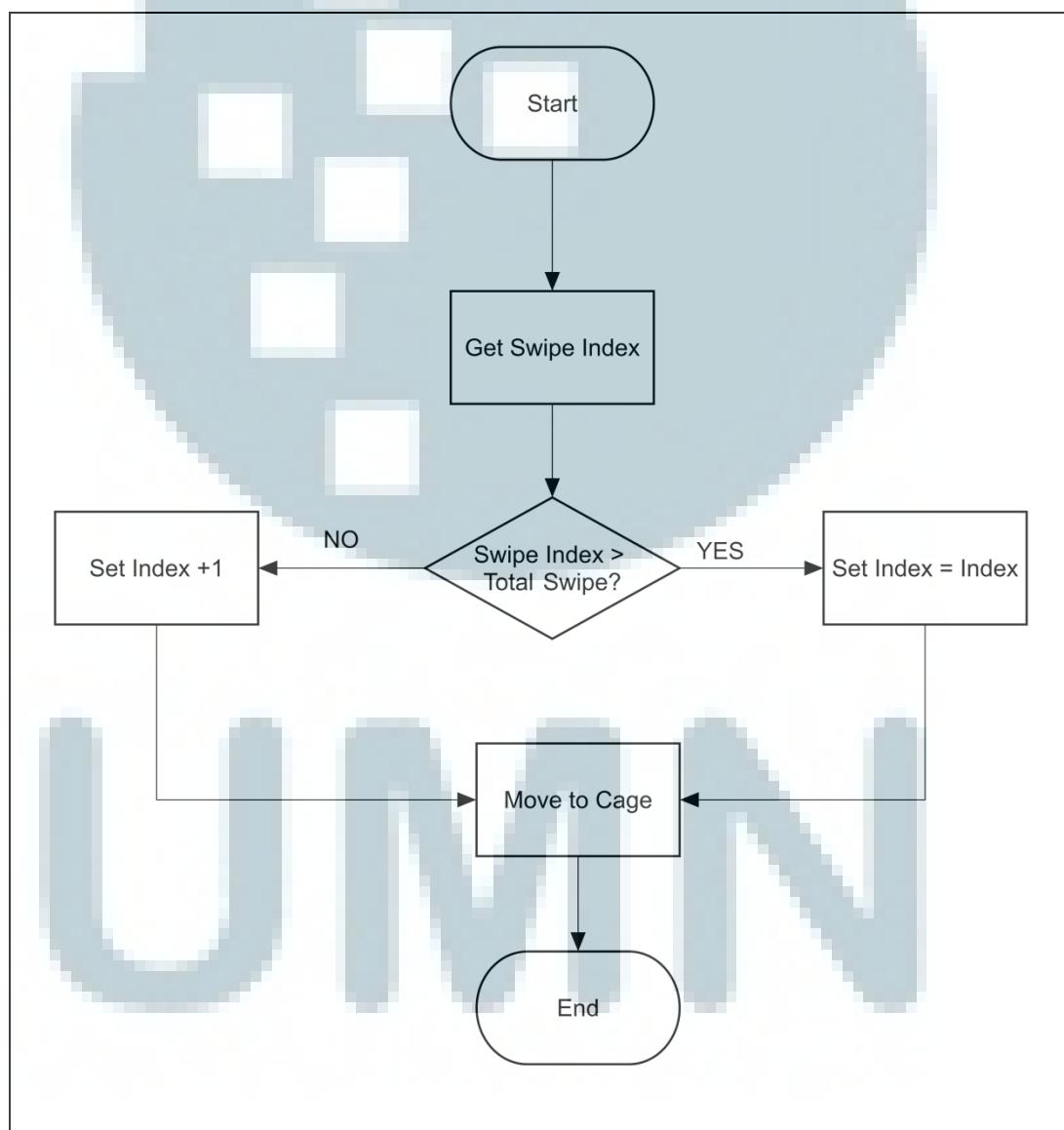
Gambar 3.16 Flowchart Rare Scene

Gambar 3.16 menunjukkan proses pemilihan binatang, dan pembentukan *pattern*. Proses pertama adalah mengambil semua *list* binatang *rare* pada negara yang sedang di-*rescue* dan mengirimkannya pada proses *shuffle fisher yates*. Hasil dari *shuffle* akan diambil satu buah nama binatang yang menjadi hewan yang akan di-*rescue*. Proses berikutnya mengambil *list* titik pembentuk *pattern* dari binatang yang akan di-*rescue*, *list* tersebut akan dikirimkan ke proses *shuffle fisher-yates* untuk mendapatkan satu dari kumpulan titik. Titik yang telah didapat akan digunakan untuk men-*generate pattern* binatang.



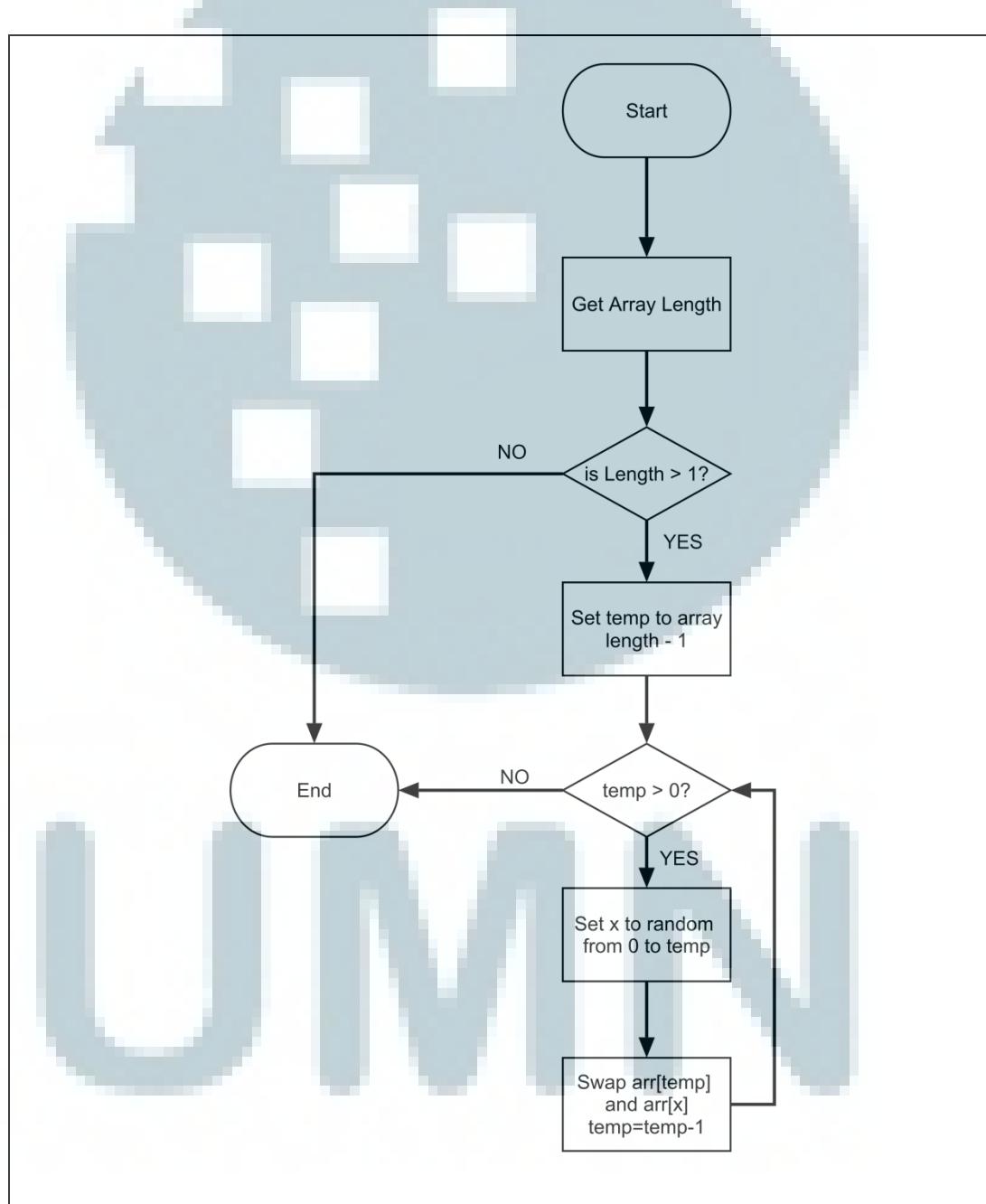
Gambar 3.17 Flowchart Lobby Controller

Gambar 3.17 menunjukkan proses yang terjadi jika pemain berada pada *panel lobby*. Pada *panel* tersebut terdapat beberapa tombol seperti *store* yang akan memanggil *popup store* yang berfungsi untuk pemain membeli lokasi yang baru, tombol *mini game* yang akan memanggil *scene mini games*, *zoo review* yang akan memanggil *popup* yang berisi informasi *grading* pada setiap negara dan tombol *zoo office* yang akan memanggil *popup* menampilkan informasi data dari semua binatang yang telah didapatkan.



Gambar 3.18 Flowchart Zoo Controller

Gambar 3.18 menunjukkan *flowchart* dari *zoo controller* yang mengatur *swipe* paling kanan bisa dilakukan pemain. Sistem akan mengambil *index swipe* arah kanan jika dalam *checking index* kurang dari total *index* maka pemain dapat melakukan *swipe*, sebaliknya jika *index* melebihi total maka saat terjadi *swipe* *index* akan di-set ke *index* dengan nilai yang sebelumnya.



Gambar 3.19 *Flowchart Fisher-Yates Shuffle*

Pada Gambar 3.19 menunjukkan alur dari algoritma *fisher-yates shuffle*.

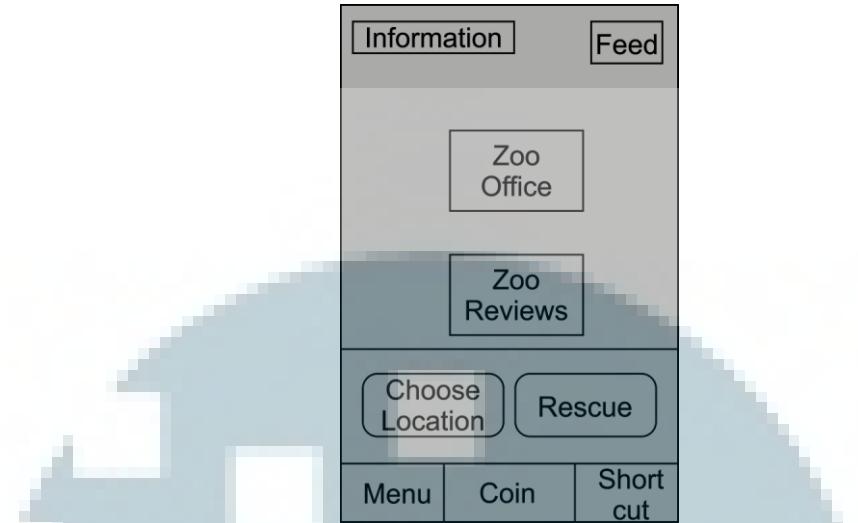
Proses diawali dengan pengambilan panjang dari *array*, jika panjang *array* tidak lebih besar dari satu proses akan langsung berhenti. Apabila panjang *array* lebih dari satu, dilakukan penetapan nilai *temp* menjadi panjang dari *array*. Setelah itu dilakukan pengurangan nilai *temp* sebesar 1. Jika nilai tidak lebih besar atau sama dengan nol, maka proses proses pengacakan dianggap selesai. Namun, jika nilai *temp* lebih besar atau sama dengan 0 maka tetapkan nilai *x* berdasarkan suatu bilangan acak dimulai dari 0 sampai *temp*. Kemudia tukar *index* ke-*temp* dengan *index* ke-*x* dan lakukan pengurangan nilai *temp* dengan 1. Setelah itu ulangi proses sampai nilai *temp* menjadi nol.

3.5 Perancangan Player Interface



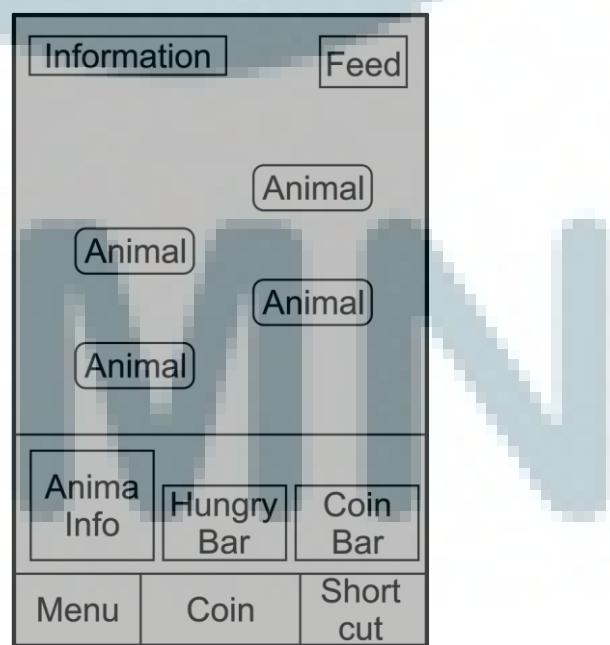
Gambar 3.20 *Mockup Panel Lobby*

Gambar 3.20 adalah *panel* yang pertama yang akan dilihat pemain saat pertama kali memulai permainan. Pada *panel* tersebut terdapat tombol *store*, *mini game*, *zoo review*, *zoo office*. Sedangkan tampilan tombol *menu*, *coin*, *shortcut*, *information*, *feed* dan *message* akan selalu mengikuti setiap terjadi *swipe*.



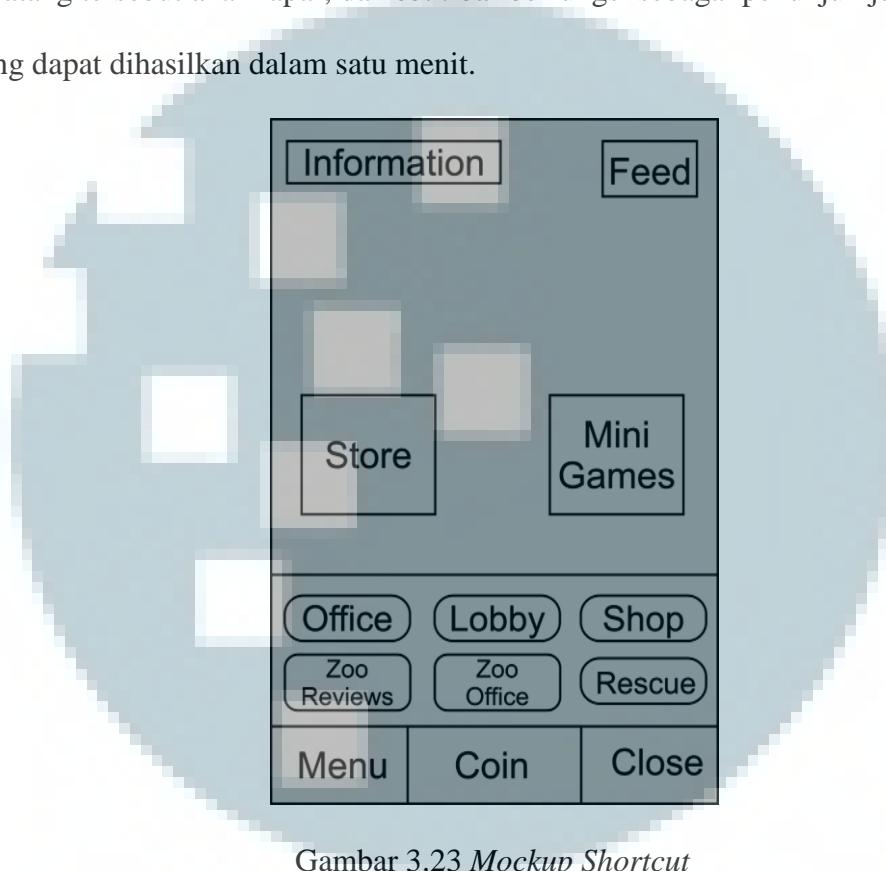
Gambar 3.21 *Mockup Panel Office*

Gambar 3.21 menunjukkan tampilan pada *panel office*. Pada *panel* ini terdapat empat tombol yang dapat ditekan oleh pemain. Tombol *choose location* berguna untuk memilih lokasi untuk menyelamatkan binatang, *rescue* berfungsi untuk masuk pada *scene puzzle* yang berguna untuk menyelamatkan binatang. Dua tombol di atas akan dibuat dalam bentuk gedung yang memiliki fungsi untuk memanggil *zoo review* dan *zoo office*.



Gambar 3.22 *Mockup Panel Kandang Binatang*

Gambar 3.22 menunjukkan tampilan pada salah satu kandang. Pada *panel* kandang terdapat *Animal info* yang jika ditekan akan menampilkan informasi dari binatang pada kandang tersebut. *Hungry bar* berfungsi sebagai penunjuk lama dari binatang tersebut akan lapar, dan *coin* bar berfungsi sebagai penunjuk jumlah *coin* yang dapat dihasilkan dalam satu menit.



Gambar 3.23 *Mockup Shortcut*

Gambar 3.23 menunjukkan tombol yang terdapat panel *shortcut*. Terdapat enam tombol yang dapat mempermudah pemain, seperti tombol *office* yang dapat membuat pemain langsung *jumping* ke panel *office*, tombol *lobby* yang dapat membuat pemain langsung *jumping* ke panel *lobby*, dan tombol *rescue* yang dapat digunakan pemain untuk melakukan *rescue*.



Gambar 3.24 Mockup Popup Menu

Gambar 3.24 menunjukkan tampilan dari *popup menu* yang akan muncul saat tombol menu ditekan. Terdapat enam tombol yang berada pada *popup menu*.



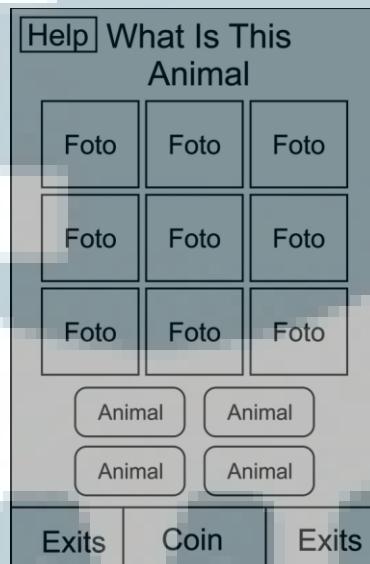
Gambar 3.25 Mockup Popup Store

Gambar 3.25 menunjukkan *panel store* yang akan menampilkan jumlah negara yang sudah terbuka dan dapat dibeli oleh *player*. Negara yang terkunci tidak dapat dibeli pemain. Negara yang terkunci akan terbuka saat kebun binatang pada negara sebelumnya telah mencapai *grade* tertentu.



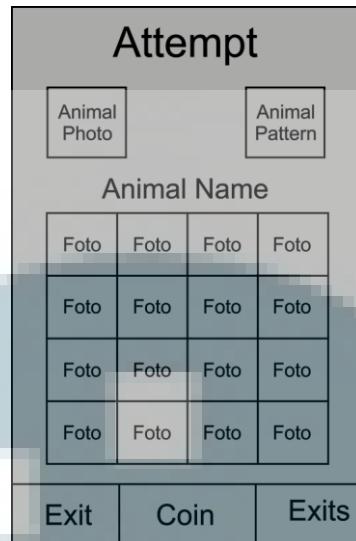
Gambar 3.26 *Mockup Popup Choose Location*

Gambar 3.26 menunjukkan tampilan *popup choose location* yang berisi negara-negara yang telah dibeli. Negara yang belum dibeli tidak akan ditampilkan pada *popup* ini.



Gambar 3.27 *Mockup Secene Rescue common*

Gambar 3.27 adalah tampilan dari *rescue common*. Pada tampilan tersebut terdapat tombol *help*, foto binatang yang telah teracak beserta empat buah jawaban yang dapat dipilih oleh pemain.



Gambar 3.28 *Mockup Rescue Rare*

Gambar 3.28 menunjukkan tampilan pada *rescue rare*, pada tampilan *rescue* tersebut terdapat *attempt* yang berisi informasi jumlah kesempatan dapat membuka kotak, *animal photo* menampilkan foto binatang yang di-*rescue* dan *animal pattern* menampilkan pola binatang yang harus dicari pada kotak 4x4. Terdapat kotak 4x4 yang didalamnya terdapat *coin*, *zonk* dan *pattern* binatang.