

# Perancangan *Game Engine* untuk *Game Visual Novel* Berbasis Android Dengan Diagram Alur Cerita

Wu Alvin Hardy<sup>1</sup>, Leo Willyanto Santoso<sup>2</sup>, Justinus Andjarwirawan<sup>3</sup>

Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Kristen Petra

Jl. Siwalankerto 121 – 131 Surabaya 60236

Telp. (031) – 2983455, Fax. (031) – 8417658

E-mail: wualvinhardy@gmail.com<sup>1</sup>, leow@petra.ac.id<sup>2</sup>, justin@petra.ac.id<sup>3</sup>

**ABSTRAK:** Untuk membuat sebuah *game*, diperlukan beberapa komponen inti seperti *graphic*, *sound* dan *animation*. Dalam pembuatan komponen tersebut, terkadang ditemukan beberapa kendala karena jumlah komponen yang dibuat cukup banyak dan harus dikerjakan secara manual. *Game engine* adalah sebuah *software* yang dapat mempermudah *developer* dalam membuat sebuah *game*. Fitur yang terdapat di dalam *game engine* termasuk *graphic*, *sound* dan *animation* akan membantu *developer*, sehingga *developer* tidak perlu memikirkan lagi proses tersebut secara manual. *Visual Novel (VN)* termasuk salah satu *video game* interaktif yang menampilkan cerita novel dalam bentuk gambar statis, audio, dan video. Sebuah *VN* yang menarik dibuat melalui proses narasi dan dialog antar karakter yang disajikan, dengan teks diiringi dengan gambar dan suara yang akan memberikan kesan hidup pada karakter yang ada.

*Game engine* untuk *game visual novel* sudah cukup banyak jenisnya, mulai dari Om3ga VNE berbasis Android, Ren'Py berbasis *Desktop* yang dibuat dengan bahasa pemrograman Python, NScripter berbasis *Desktop*, *VN Authoring Tool*. Namun, dari sekian banyak *Game Engine* untuk *visual novel* yang sudah ada, hanya Om3ga VNE dan *VN Authoring Tool* yang dapat digunakan pada platform Android. Sedangkan, untuk tampilan pada *VN Authoring Tool* yang berbasis web tidak responsif dan Om3ga VNE tidak menyediakan fitur untuk mempermudah pengguna melihat garis besar *video game* yang dikembangkan.

Pada skripsi ini dilakukan pembuatan aplikasi *game engine visual novel* berbasis Android yang menggunakan *Story Flow Diagram (SFD)* sebagai fitur utama untuk membantu pengguna dalam membuat *visual novel*. Pengguna dapat melihat garis besar alur cerita dengan fitur SFD ini, sehingga akan memudahkan pengguna dalam menentukan alur cerita dari *visual novel* yang akan dibuat.

**Kata Kunci:** *Game Engine*, *Visual Novel*, *Story Flow Diagram*, Android.

**ABSTRACT:** *In the process of creating a video game, it takes several core component such as graphic, sound, and animation. Sometimes, in the process of creating those components, some obstacles are encountered due to the number of components are quite large and need to be done manually. Game engine is a software intended to make it easier for developers to create a game. The features in game engine include graphic, sound, and animation, which will help the developer, so the developer do not need to implement those components manually. Visual Novel (VN) is an interactive video game that displays novel stories in the form of static image, audio, and video. A good*

*VN is made through the narration and dialogue shown between the presented characters, with text along with image and sound which will give an impression of the characters presented as if they live.*

*There is already a variety of game engine for visual novel games, ranging from Android-based Om3ga VNE, desktop-based Ren'Py made with the Python programming language, desktop-based NScripter, VN Authoring Tool. However, of the many existing game engine for Visual Novel, only Om3ga VNE and VN Authoring Tool can be used on the Android. Even then, the web-based VN Authoring Tool isn't responsive for mobile device and Om3ga VNE does not provide a feature to make it easier for users to see the outline of the developed video game.*

*In this thesis, an Android-based visual novel game engine is made with the use of Story Flow Diagram (SFD) as the main feature to help user in visual novel creation. Users can see the outline of the storyline with this SFD feature, so that it will make it easier for users to determine the storyline of visual novel that will be created.*

**Keywords:** *Game Engine*, *Visual Novel*, *Story Flow Diagram*, Android.

## 1. LATAR BELAKANG

Mempermudah proses pembuatan *game* adalah tujuan utama dari sebuah aplikasi *Game Engine*. Dalam istilah bidang komputer, *Game Engine* adalah sebuah *software* inti yang digunakan sebagai komponen utama dalam proses pembuatan *Video Game*. Banyak proses dalam *Video Game* yang ditangani oleh *Game Engine*, beberapa contohnya seperti *Graphic*, *Sound*, *Scripting*, *Animation*, dan masih banyak lagi. *Game Engine* akan menangani proses-proses tersebut sehingga *developer Video Game* tidak perlu menangani proses-proses tersebut secara manual.

*Visual Novel (VN)* adalah salah satu genre *Video Game* yang berbasis fiksi interaktif yang menampilkan cerita novel dalam bentuk gambar-gambar statis, audio, dan video. Penggunaan gambar statis pada *game VN* umumnya dapat dibagi menjadi 2, yaitu latar belakang / *Background*, dan gambar karakter / *Character Sprites*. Sementara penggunaan audio dapat dibagi menjadi 3 bagian, yaitu lagu pengiring / *Background Music*, efek suara / *Sound Effect*, dan dialog karakter / *Voice Cover*.

Nilai utama dari *VN* adalah pengantaran cerita melalui narasi dan dialog antar karakter yang disajikan dengan teks diiringi dengan gambar dan suara. Penyampaian cerita yang diiringi dengan elemen tersebut memberikan kesan hidup pada karakter dalam *visual novel*.

*Game engine* untuk *game visual novel* cukup banyak jenisnya, mulai dari Om3ga VNE berbasis Android [8], Ren'Py berbasis

*Desktop* yang dibuat dengan bahasa pemrograman *Python* [9], *NSriptor* berbasis *Desktop* [7], *VN Authoring Tool* [1]. Dari banyak *Game Engine* untuk *Visual Novel* yang ada, hanya *Om3ga VNE* dan *VN Authoring Tool* yang dapat digunakan pada platform *Android*. Namun, tampilan pada *VN Authoring Tool* yang berbasis *web* tidak responsif, sementara *Om3ga VNE* tidak menyediakan fitur untuk mempermudah pengguna melihat garis besar *Video Game* yang dikembangkan.

Tujuan dari skripsi ini adalah mengembangkan fitur pada *Game Engine* yang dapat menampilkan garis besar alur cerita *Video Game* yang dikembangkan menggunakan diagram.

## 2. VISUAL NOVEL

*Visual novel* adalah salah satu genre *game* fiksi interaktif di mana pemain diberikan cerita novel yang diiringi dengan media visual seperti gambar statis. Pada umumnya, *game visual novel* hanya memiliki sedikit elemen *gameplay* dan fokus pada cerita pada *game*. Selain gambar statis, beberapa *game visual novel* juga menyertai video sebagai salah satu media penyampaian cerita.

Pada umumnya penceritaan pada *game Visual novel* menggunakan teks narasi dan dialog dari karakter yang ada pada *game*, dengan diiringi dengan *character sprites* / gambar karakter yang bersangkutan. Cara ini memungkinkan penyampaian cerita yang lebih efektif dibanding dengan literatur novel biasa.

*Visual novel* memiliki berbagai elemen yang dapat dikategorikan menjadi 2 kelompok, yaitu aspek visual dan aspek audio. Berdasarkan halaman *web* [tvtropes.org](http://tvtropes.org), ada 5 aspek visual dalam *game visual novel*, antara lain: *background*, *character sprites*, *computer graphics*, *special effects*, *text presentation* [11].

Selain aspek visual di atas, *visual novel* memiliki 3 aspek audio yang memiliki peranan penting dalam penyampaian cerita. 3 aspek audio dalam *game visual novel*, antara lain: *background music*, *sound effects*, *voice acting* [11].

*Background* dalam *game visual novel* adalah gambar latar belakang yang digunakan untuk memperjelas waktu, lokasi, serta kondisi pada cerita. Penggunaan *background* merupakan hal yang paling dasar dalam *visual novel*. *Background* yang baik dapat membantu penyaluran cerita menjadi lebih menarik.

Tokoh karakter dalam *game visual novel* memiliki berbagai *character sprites* yang merepresentasikan ekspresi karakter tersebut. Beberapa *character sprites* pada satu karakter umumnya memiliki berbagai ekspresi wajah yang berbeda walaupun menggunakan pose atau pakaian yang sama. Keragaman ekspresi wajah adalah bagian penting pada *character sprites*, sementara variasi pose dan pakaian adalah opsional [10].

*Computer Graphics* (CG) dalam *visual novel* merujuk pada gambar yang ditampilkan dalam beberapa bagian cerita tanpa diikuti dengan *character sprites*. Berbeda dari *background* yang dapat digunakan berulang-ulang, satu CG biasanya hanya digunakan sekali pada momen tertentu dalam cerita. CG umumnya menggambarkan momen suatu karakter dalam cerita dari sudut pandang yang tidak bisa digambarkan dengan *character sprites*.

*Special effects* adalah efek visual yang digunakan dalam *visual novel* untuk memberikan kesan hidup dalam cerita, contohnya seperti animasi salju jatuh. Tidak semua *visual novel* menggunakan *special effects*, namun penggunaan *special effects* yang baik akan menambah daya tarik sebuah *visual novel*.

*Text presentation* adalah cara untuk menampilkan teks dalam *visual novel*. Ada berbagai cara untuk menampilkan teks, namun pada umumnya teks dalam *visual novel* ditampilkan dalam kotak dialog yang ada di bagian bawah layar.

Media audio dalam *visual novel* adalah *background music*, *sound effect*, dan *character voice*. *Background music* adalah lagu yang dijalankan dalam *visual novel* sebagai pengiring cerita dan

disesuaikan dengan situasi dalam cerita. *Sound effect* adalah efek suara yang menggambarkan atau menekankan suatu kejadian dalam cerita. *Character voice* adalah suara yang dikeluarkan oleh karakter dalam *visual novel*. Kombinasi dari aspek visual dan audio dalam *game* inilah yang membuat *visual novel* unik dan membantu pemain mendalami cerita yang disediakan.

## 3. GAMEPLAY VISUAL NOVEL

Berbeda dengan *video game* pada umumnya, interaksi pemain dalam *game visual novel* sangat terbatas. Biasanya pemain hanya perlu mengklik untuk meneruskan pergantian teks, grafik, dan suara selama *game* berlangsung [5]. Selain itu, interaksi pemain adalah ketika pemain diberikan beberapa pilihan dalam *game* yang dapat mempengaruhi jalannya alur cerita sesuai dengan percabangan cerita yang ada.

Percabangan cerita dalam *game visual novel* akan memberikan *ending* cerita yang berbeda-beda. Tidak semua *game visual novel* menyediakan percabangan cerita, *Visual novel* yang tidak memiliki percabangan dan hanya memiliki satu alur cerita biasanya disebut *kinetic novel*. *Kinetic novel* merupakan sebuah tipe *visual novel* tanpa *gameplay* [6].

### 3.1. Fitur Gameplay Visual Novel

Dalam *game visual novel*, terdapat beberapa fitur yang dapat digunakan selama permainan berlangsung.

Fitur *save game* adalah menyimpan data dalam satu titik cerita sehingga permainan dapat dilanjutkan dari titik yang sama nantinya. Fitur ini biasanya menyediakan halaman khusus untuk menampilkan *slot-slot* yang dapat dipilih untuk menyimpan data permainan yang sedang berjalan. Jumlah *slot* pada setiap *game* berbeda tergantung dari pembuat *game*.

Fitur *load game* adalah mengambil data dari satu titik cerita yang sebelumnya disimpan dengan fitur *save*. Dengan fitur ini, pemain akan diarahkan ke halaman *load* di mana pemain akan diberikan *slot-slot* yang dapat dipilih untuk melakukan *load* jika pada slot tersebut tersimpan data yang pernah di-*save*. Jumlah *slot* yang disediakan sama dengan jumlah *slot* di halaman *save*.

Sama seperti fitur *save game*, *quick save* digunakan untuk menyimpan data dalam satu titik cerita yang nantinya dapat digunakan pemain untuk melanjutkan permainan dari titik yang sama. Berbeda dari fitur *save game*, *quick save* tidak memiliki halaman khusus untuk menampilkan *slot-slot* data. *Quick save* merupakan tombol dalam permainan yang akan langsung menyimpan data jika tombol tersebut ditekan, dan *quick save* hanya memiliki satu *slot* penyimpanan data.

Fitur *quick load* digunakan untuk *load* data yang disimpan oleh fitur *quick save* tanpa menampilkan halaman *slot-slot* data.

*Autoplay* adalah fitur dalam *Visual Novel* di mana pergantian teks, grafis, dan suara akan dilanjutkan tanpa perlu menunggu interaksi dari pemain. *Autoplay* akan melanjutkan pergantian sesaat setelah animasi teks, animasi grafis, dan suara selesai dimainkan.

Sama seperti *autoplay*, *fast forward* akan meneruskan pergantian teks, grafis, dan suara tanpa menunggu interaksi dari pemain. Hal yang berbeda pada *fast forward* adalah pada mode *fast forward* animasi teks, animasi grafis, dan suara tidak dimainkan, sehingga pergantian antar *scene* menjadi lebih cepat.

Fitur *voice replay* adalah fitur untuk memutar ulang suara karakter jika ada. Umumnya fitur ini terdapat pada *visual novel* yang menyediakan *voice acting*.

Fitur *backlog* merupakan halaman yang menampilkan baris-baris teks sebelumnya. Jumlah baris yang dapat tersimpan pada *backlog* tergantung dari pembuat *visual novel*.

Fitur *hide textbox* adalah fitur untuk menyembunyikan atau menampilkan kembali kotak teks. Tujuan fitur ini adalah agar

pemain dapat melihat gambar *background* atau *computer graphic* secara keseluruhan.

#### 4. GAME ENGINE

*Game engine* adalah sebuah sistem aplikasi yang dirancang untuk mempercepat proses pembuatan dan pengembangan *game* [3]. Tujuan dari *game engine* adalah menyediakan *framework* yang dapat digunakan *developer* untuk mengembangkan *video game* yang dapat dimainkan pada komputer atau konsol *game* komersial. Fitur-fitur *game engine* bervariasi tapi umumnya mengintegrasikan solusi untuk menangani proses *rendering*, *physics and collision detection*, animasi, *networking*, audio, *artificial intelligence*, *memory management*, *scripting*, dan efek visual berbasis partikel. *Game engine* akan menangani beberapa proses *background* sehingga *developer* dapat lebih fokus pada proses pengembangan konsep *game*.

#### 5. ANDROID CANVAS

*Canvas* adalah sebuah kelas yang disediakan *Android* untuk melakukan penggambaran secara manual. Kelas *canvas* memiliki fungsi *draw* yang digunakan untuk menggambar berbagai macam bentuk dan gambar. Pada umumnya untuk menggambar sesuatu dalam *Android* diperlukan 4 komponen dasar berikut [2]: *bitmap*, *canvas*, *drawing primitive*, dan *paint*.

#### 6. FLOW CHART & STORY FLOW DIAGRAM

*Flow chart* adalah representasi grafikal atau simbolik dari suatu urutan proses [4]. Setiap langkah dalam suatu proses dilambangkan dengan berbagai symbol yang dihubungkan dengan garis panah untuk menunjukkan alur dari proses tersebut.

Dalam skripsi ini, *Story Flow Diagram* (SFD) adalah fitur utama aplikasi untuk meng-visualisasikan alur cerita *game visual novel* dengan menggunakan gambar diagram yang terinspirasi dari *flow chart*. Dalam SFD terdapat 2 unsur utama, yaitu entitas dan koneksi

#### 7. DESAIN SISTEM

Pada tahap ini akan dijelaskan tentang desain aplikasi yang dibuat. Pengerjaan desain dibagi menjadi dua bagian, yaitu desain sistem aplikasi dan desain *interface* aplikasi.

##### 7.1. Desain Sistem Aplikasi

Aplikasi ini memiliki dua fitur utama, yaitu fitur untuk membuat *game visual novel* dan fitur untuk memainkan *game visual novel* yang dibuat dengan aplikasi ini. Pada bab 3.1 akan dijelaskan struktur data yang digunakan pada aplikasi serta sistem *Story Flow Diagram* pada aplikasi.

###### 7.1.1 Story Flow Diagram

*Story Flow Diagram* (SFD) adalah sistem yang dirancang untuk meng-visualisasikan alur cerita pada *game visual novel* yang dibuat oleh aplikasi dengan menggunakan diagram. Tujuan dari SFD adalah untuk mempermudah pembuat *game* dalam melihat alur cerita dalam *game* yang dibuat. Dalam SFD, terdapat 4 objek utama, yaitu *start*, *page*, *connection*, dan *choice-selection*.

*Start* merupakan penanda awal mulainya sebuah *visual novel*. Semua *game visual novel* pada aplikasi yang dimainkan akan berawal dari *start*. Dalam SFD, *start* hanya ada satu pada lembar kerja / *worksheet* yang disediakan, dan *start* hanya bisa

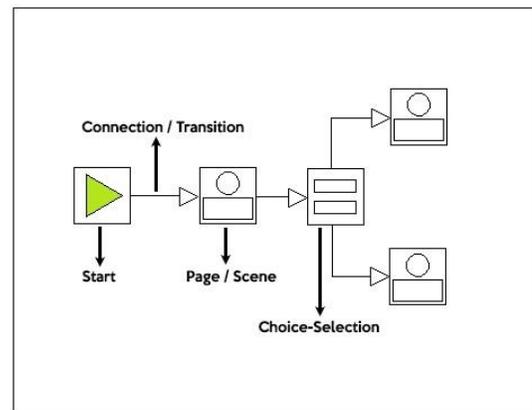
memiliki 1 *connection / transition*.

*Page / scene* adalah objek yang berisikan informasi tentang *background image*, *character sprites*, *computer graphics*, *animation*, *background music*, *sound effects*, dan *character voice*. Pada saat *visual novel* dimainkan, *page* akan dijalankan dari satu *page* ke *page* atau *choice-selection* berikutnya. Objek *page / scene* hanya bisa memiliki 1 *connection / transition*.

*Connection / transition* adalah objek penghubung yang menghubungkan objek-objek lain. *Start*, *page*, *choice-selection*, dan *end* dapat dihubungkan satu sama lain dengan *connection*.

*Choice-selection* adalah objek yang menyediakan beberapa pilihan kepada pemain. Jumlah pilihan yang disediakan dalam *choice-selection* dapat diatur dan masing-masing pilihan akan mengarah ke *page / scene* yang berbeda tergantung dari jumlah *connection / transition* dan pilihan yang ada.

Tujuan dari *choice-selection* adalah untuk membuat percabangan cerita bersama dengan *connection*. Gambar 1 berikut menampilkan contoh 4 objek *game* yang dijelaskan.



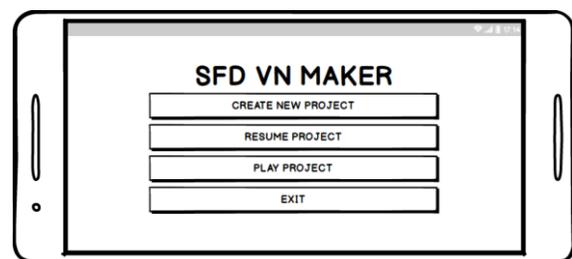
Gambar 1. Tampilan SFD

##### 7.2 Desain Interface Aplikasi

Desain *interface* aplikasi dirancang untuk setiap halaman yang dapat diakses pengguna aplikasi. Berikut beberapa kerangka tampilan antarmuka halaman yang tersedia pada aplikasi yang dibuat.

###### 7.2.1 Halaman Utama

Halaman utama adalah halaman yang pertama ditampilkan kepada pengguna saat aplikasi berjalan. Halaman ini berisi menu untuk navigasi antar fungsi utama dalam aplikasi. Terdapat 3 pilihan dalam menu, yaitu *Create new project*, *Resume project*, *Play project*, dan *Exit*. Desain halaman utama dapat dilihat pada Gambar 2.

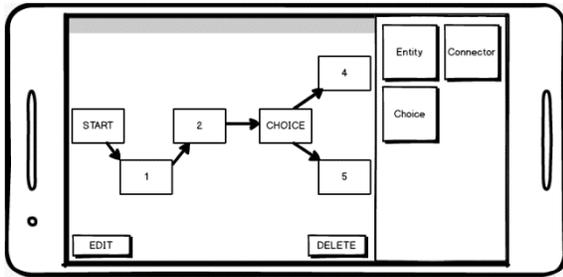


Gambar 2. Desain halaman utama

###### 7.2.2 Halaman SFD Editor

Halaman SFD Editor merupakan halaman yang digunakan *user* untuk membuat *visual novel*. *User* sebagai editor dapat

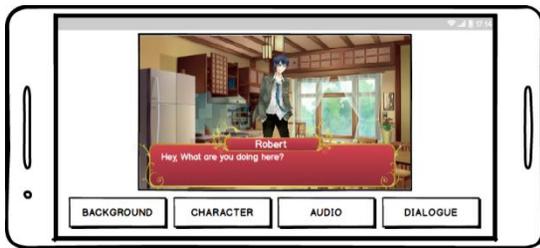
mengelola SFD dengan menambahkan dan meng-edit *game object* seperti *page* dan *connection* yang akan digunakan saat menjalankan *visual novel*. Desain halaman SFD Editor dapat dilihat pada Gambar 3.



Gambar 3. Desain halaman SFD Editor

### 7.2.3 Halaman Pengelolaan Page

Halaman pengelolaan *page* dapat diakses oleh *user* melalui halaman SFD Editor saat *user* hendak mengelola salah satu objek *page* pada SFD. Halaman ini menampilkan detail objek *page* yang tidak ditampilkan dalam SFD. Desain halaman pengelolaan *page* dapat dilihat pada Gambar 4.



Gambar 4. Desain halaman pengelolaan *page*

### 7.2.4 Halaman Play Game

Halaman *Play Game* merupakan halaman permainan yang telah dipilih oleh pengguna sebelumnya. Pada halaman ini pengguna dapat berinteraksi dengan menyentuh layar untuk berpindah dialog antara karakter. Desain halaman *Play Game* dapat dilihat pada Gambar 5.



Gambar 5. Desain halaman *play game*

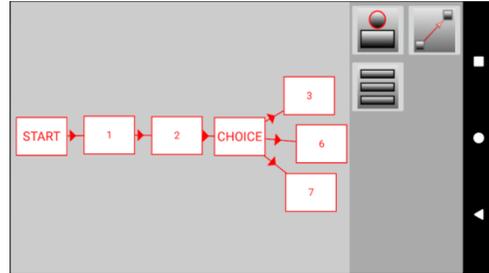
## 8. PENGUJIAN

Pada tahap ini akan dijelaskan tentang hasil dari uji coba sistem yang dilakukan untuk mengetahui *bug* dan *error* yang ada pada aplikasi yang telah dibuat. Proses uji coba dalam bab ini terbagi menjadi dua, yaitu pengujian sistem dan pengujian oleh *user*.

### 8.1 Pengujian Penambahan Objek Pada SFD

Pada aplikasi, pengguna dapat menambahkan objek berupa entitas *page scene* atau entitas pilihan. Untuk menambahkan suatu entitas, pengguna dapat melakukan *drag and drop* entitas

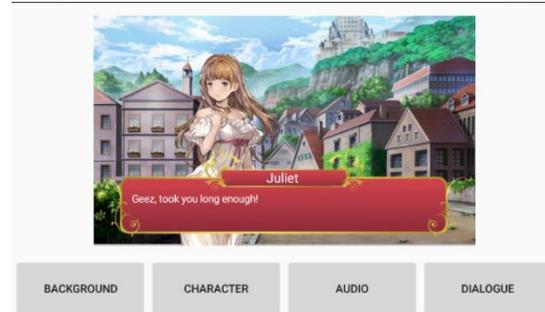
dari *sidebar* ke kanvas SFD. Setelah itu, entitas akan muncul di kanvas SFD. Untuk menambahkan *connection*, pengguna dapat menekan tombol *connection* pada *sidebar* dan memilih entitas awal dan entitas tujuan. Hasil pengujian dapat dilihat pada Gambar 6.



Gambar 6. Pengujian Pembuatan SFD pada aplikasi

### 8.2 Pengujian Penambahan Objek Pada Entitas Page Scene

Pada entitas *page scene*, pengguna dapat menambah gambar latar, gambar karakter, suara latar dan karakter dan tulisan dialog karakter atau cerita. Untuk menambah gambar latar pengguna dapat menekan tombol *background*, setelah itu pengguna memilih gambar yang akan dimasukkan ke dalam kanvas. Gambar akan muncul pada seperti yang terlihat pada Gambar 7.



Gambar 7. Gambar latar pada kanvas

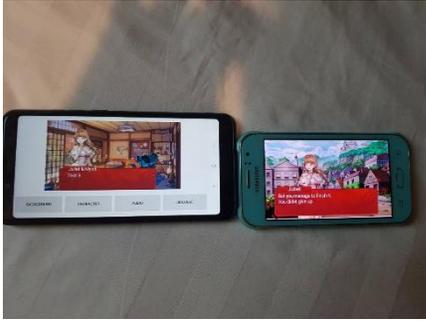
### 8.3 Pengujian Aplikasi Pada Perangkat Mobile

Pengujian aplikasi *Visual Novel Maker* dilakukan pada tiga perangkat *mobile*, setiap perangkat memiliki ukuran dimensi layar dan resolusi yang berbeda. Spesifikasi layar tiap perangkat dapat dilihat pada Tabel 1.

Tabel 1. Tabel Perangkat *Mobile*

Nama Perangkat	Size	Resolusi
Samsung Galaxy S8	5.8 inch	1440 x 2960 pixels (Full HD)
Xiaomi Redmi Note 5	5.99 inch	1080 x 2160 pixels (Full HD)
Samsung Galaxy J1	4.3 inch	480 x 800 pixels

Setelah dilakukan pengujian aplikasi pada tiga perangkat tersebut, tidak ditemukan adanya perbedaan performa dari setiap perangkat yang diuji. Dari segi tampilan untuk setiap scene yang dibuat tidak ada kendala. Ukuran gambar tidak menjadi masalah karena bisa melakukan *scaling* pada kanvas yang disediakan. Tampilan pengujian dapat dilihat pada Gambar 9.

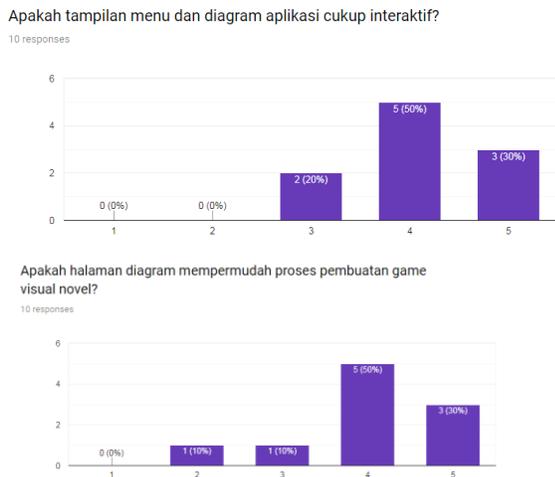


Gambar 8. Tampilan Pengujian Pada Perangkat Mobile

### 8.4 Pengujian Aplikasi pada User

Pengujian aplikasi juga dilakukan kepada responden. Responden yang menguji program adalah Mahasiswa dan pekerja. Saat pengujian responden diminta untuk menjalankan aplikasi, membuat sebuah visual novel singkat yang memanfaatkan semua fitur yang disediakan aplikasi, memainkan visual novel yang dibuatnya, mengisi pertanyaan seputar tampilan aplikasi, *user experience*, serta fitur-fitur yang ada di dalam aplikasi.

Hasil survey pengujian aplikasi mendapatkan sepuluh responden. Terdapat delapan pertanyaan pada kuisisioner dimana angka satu merupakan nilai terendah dan angka lima merupakan nilai tertinggi. Hasil dari pengisian kuisisioner oleh responden dapat dilihat pada Gambar 10.



Gambar 9. Grafik Hasil Kuisisioner Responden Terhadap Aplikasi

## 9. KESIMPULAN

Berdasarkan hasil pengujian yang dilakukan didapatkan beberapa kesimpulan, yaitu:

- Aplikasi *Visual Novel Maker* yang dibuat sudah mengimplementasikan sistem *Game Engine* yang berbasis Android dan dapat menampung data yang dibutuhkan dalam pembuatan *Visual Novel*.
- Aplikasi yang dibuat sudah dapat membantu dalam mendesain sistem yang menggambarkan alur cerita dengan menggunakan fitur *Story Flow Diagram*.
- Berdasarkan hasil kuesioner kepuasan responden terhadap aplikasi, didapatkan bahwa seluruh responden menyukai

aplikasi *Visual Novel Maker* dari segi penggunaan dan desain *User Interface*. Sehingga, dapat disimpulkan bahwa aplikasi *Visual Novel Maker* sudah memiliki desain *User Interface* yang *User Friendly*.

## 10. REFERENSI

- [1] Admaja, H. A. 2015. *Visual Novel Authoring Tool Berbasis Web*. (TA No. 03021441/INF/2015). Fakultas Teknologi Industri Program Studi Teknik Informatika, Universitas Kristen Petra, Surabaya.
- [2] Android Developers. (August 13, 2018) *Android Canvas Documentation*. Retrieved January 10, 2019, from <https://developer.android.com/reference/android/graphics/Canvas>
- [3] Buckley, C. M. (2012) *Design and Implementation of a Genre Hybrid Video Game That Integrates The Curriculum of an Introductory Programming Course*. Retrieved January 1, 2019, from <https://www.slideshare.net/CoryBuckley/educational-game-design-thesis>
- [4] Hebb, N. (n.d.). *What is a Flow Chart?* Retrieved January 12, 2019, from <https://www.breezetre.com/articles/what-is-a-flow-chart>
- [5] Kristian (2008). *Building a Gui-based Visual Novel Authoring Tool with Java Swing*. Retrieved January 10, 2019, from <http://library.binus.ac.id/eColls/eThesiscoll/Bab2HTML/Bab2BICS0850/page.html>
- [6] Logan M. (August 22, 2013). *What are the differences between visual novel, eroge, gal games, and a dating sim?*. Retrieved November 23, 2017, from <https://anime.stackexchange.com/questions/4926/what-are-the-differences-between-visual-novel-eroge-gal-game-and-a-dating-sim>
- [7] *NScripter*. Retrieved November 21, 2017, from <http://nscripter.insani.org/>.
- [8] *Om3ga VNE*. Retrieved November 22, 2017, from <http://vnengine.com/index.html>.
- [9] *Ren'Py*. Retrieved November 21, 2017, from <https://www.renpy.org/>.
- [10] The Teacup Visual Novel Creators Community. (2010). *Tutorial – Spritemaking for visual novel beginners*. Retrieved November 23, 2017, from <http://teacup.lunaen.com/index.php?topic=13.0>
- [11] TV Tropes Community. (n.d.). *Make A Visual Novel*. Retrieved November 23, 2017, from <https://tvtropes.org/pmwiki/pmwiki.php/SoYouWantTo/MakeAVisualNovel>