

PEMBUATAN GAME TOWER DEFENSE SUMMONER'S QUEST MENGGUNAKAN UNITY DENGAN TOUCH GESTURE RECOGNIZER

Bill Kevin¹, Jeanny Pragantha², Darius Andana Haris³

Teknik Informatika, Fakultas Teknologi Informasi, Universitas Tarumanagara
Jl. Letjen S. Parman No.1, Jakarta

Email : eldranis@yahoo.com¹, jeannyp@fti.untar.ac.id², dariush@fti.untar.ac.id³

ABSTRAK

Game Summoner's Quest adalah *game* yang ber-*genre Tower Defense*. *Game* ini dibuat dengan tujuan memberikan pengalaman baru pada pemain dengan mengimplementasikan fitur *touch gesture recognizer*. *Game* ini mempunyai tampilan berbentuk dua dimensi dan dimainkan dengan menggunakan *Smartphone* berbasis *Android*. *Game* ini dibuat menggunakan *Unity* dengan *C#* sebagai bahasa pemrograman. Pemain harus menghancurkan *base* musuh dengan menggambar simbol pada layar untuk mengeluarkan prajurit yang akan mengalahkan monster-monster musuh yang melindungi *base* musuh. Pengujian dilakukan dengan metode *blackbox*, *alpha testing*, dan *beta testing* melalui survei pada 40 responden. Hasil pengujian menunjukkan bahwa fitur menggambar simbol pada *game* *Summoner's Quest* membuat *gameplaygame* ini menjadi unik dan menarik sehingga pemain tertarik untuk bermain lagi.

Kata Kunci: *Summoner's Quest, Tower Defense, Unity, Game 2D, Touch Gesture Recognizer.*

PENDAHULUAN

Video game merupakan sebuah permainan elektronik yang melibatkan manusia untuk berinteraksi dengan tampilan antarmuka untuk menghasilkan umpan balik visual pada perangkat video seperti layar televisi atau monitor computer (Bates, 2004) Sekarang ini *video game* merupakan salah satu sarana hiburan yang sudah tidak asing lagi untuk semua orang, hal itu disebabkan *video game* dapat diakses tidak hanya melalui PC ataupun konsol, tetapi juga melalui ponsel pintar. *Video game* juga memiliki banyak *genre* (tipe), salah satunya adalah *strategy*. *Strategy game* adalah sebuah *video game* yang hasil akhirnya ditentukan dari cara apa yang dipilih oleh pemain. *Genre strategy* ini memiliki beberapa sub-*genre*, salah satunya adalah *Tower Defense*. Sub-*genre* ini memiliki tujuan utama melindungi markas atau *base* pemain dari musuh-musuh yang datang melalui suatu jalur yang telah ditentukan. Sub-*genre* ini biasanya berfokus pada pemilihan dan penempatan *tower* atau prajurit dengan efek yang berbeda-beda untuk melawan musuh-musuh dengan kemampuan yang berbeda.

Berdasarkan penjelasan tersebut, maka dirancanglah permainan *Summoner's Quest*. Permainan ini merupakan permainan ber-*genre Tower Defense*. Pemain harus mengeluarkan prajurit untuk melawan musuh-musuh yang berjalan menuju *base* pemain. Berbeda dengan permainan *Tower Defense* pada umumnya, permainan ini akan menggunakan *touch gesture recognizer*, yaitu pemain akan menggambar simbol atau bentuk tertentu untuk mengeluarkan prajurit.

DASAR TEORI

Dalam suatu perancangan *game*, diperlukan penjelasan terlebih dahulu mengenai beberapa teori yang mendukung dan berkaitan dengan perancangan tersebut. Hal tersebut antara lain adalah teori mengenai *genre game*, *gameplay*, dan *Game Controller*.

GenreGame

Genre menjelaskan tentang tema dan alur sebuah permainan secara garis besar. *Genre* yang berhubungan dengan *game* yang akan dirancang

adalah *strategy game*. *Strategy game* secara umum fokus pada kemampuan berpikir pemain dalam membuat strategi atau taktik yang tepat untuk memenangkan permainan.

Strategy Game terbagi menjadi beberapa sub-genre :

1. *Real-Time Strategy*

Real-Time Strategy adalah *strategy game* yang berfokus pada kemampuan pemain dalam menentukan pilihan dan tindakan dalam permainan yang keadaannya terus berubah setiap saat.

2. *Real-Time Tactics*

Real-Time Tactic adalah sub-genre *strategy game* tentang *tactical wargames* yang dimainkan secara *real-time*.

3. *Turn-Based Strategy*

Pada *Turn-Based Strategy*, pemain memiliki jangka waktu tertentu dalam menentukan tindakan yang akan dilakukan.

4. *Turn-Based Tactics*

Turn-Based Tactics adalah sub-genre *strategy game* dimana pemain mempertimbangkan taktik dan bertindak dalam medan perang melalui sistem *stop-action*. Medan perang dalam permainan ini umumnya berupa konfrontasi skala kecil.

5. *Tower Defense*

Tower Defense adalah bagian dari *real-time strategy game*. Pemain harus melindungi markas atau *base* pemain dari musuh-musuh yang datang melalui suatu jalur yang telah ditentukan. Contoh *game* dari sub-genre ini adalah *Epic War 5* dan *The Battle Cats*.

6. *Multiplayer Online Battle Arena (MOBA)*

MOBA merupakan sub-genre *real-time strategy game* yang pemainnya masing-masing mengontrol satu karakter di salah satu dari dua tim dengan tujuan untuk menghancurkan struktur utama tim lawan.

7. *Artillery*

Artillery adalah *strategy game* dimana pemain biasanya mengendalikan tank dan berkelahi dengan pemain lain dalam pertempuran.

8. *4X*

4X game adalah sub-genre *strategy game* yang pemainnya mengontrol sebuah kerajaan dan mengeksplorasi, mengembangkan, memanfaatkan, dan memusnahkan (*explore, expand, exploit, and exterminate*).

9. *Wargames*

Wargames adalah sub-genre *strategy game* yang berfokus pada strategi atau taktik perang pada peta.

Genre game yang dibuat adalah *Tower Defense*.

Gameplay

Gameplay adalah istilah yang biasanya dipakai untuk menunjukkan cara pemain berinteraksi dengan *video game*. (Techopedia, 2016) *Gameplay* dikarakterisasi lebih lanjut sebagai cara permainan ini dimainkan, termasuk aturan, plot, tujuan dan bagaimana cara menaklukkan *game* tersebut, serta keseluruhan pengalaman pemain. Setiap permainan mempunyai *gameplay* yang berbeda dengan permainan lainnya.

Dalam *game* ini pemain ditugaskan untuk menghancurkan *base* musuh. Pemain menggambar simbol pada layar untuk memanggil prajurit, prajurit tersebut lalu akan berjalan ke arah *base* musuh dan akan melawan monster musuh yang menghalangi jalan. Pemain juga dapat menggunakan *Spell* untuk membantu membunuh musuh.

Game Controller

Setiap permainan memerlukan sebuah perantara yang disebut sebagai *game controller* untuk menciptakan interaksi didalamnya. *Game controller* merupakan sebuah perangkat yang digunakan dalam memberikan perintah ke dalam suatu permainan. Keberadaan sebuah *game controller* sangat penting dalam sebuah permainan, karena *game controller* memberikan perintah untuk menggerakkan suatu objek, mengeluarkan senjata, dan hal-hal lainnya yang ada didalam sebuah permainan. Biasanya *game controller* dirancang sedemikian rupa untuk memudahkan pemain dalam melakukan interaksi di dalam permainan. (Sethi, 2005) *Game controller* dapat berupa: (Dark Watcher, 2016)

1. *Joystick Controller*

Joystick controller merupakan salah satu *controller* pertama yang diciptakan untuk memainkan *video game*. Berbeda dengan *D-Pad* yang hanya memiliki empat *directional movement*, *Joystick Controller* memiliki delapan *directional movement*.

2. *D-Pad Controller*

D-Pad controller merupakan salah satu *game controller* yang menjadi dasar dari *controller* yang ada sekarang ini.

3. Wireless Controller

Wireless Controller merupakan *game controller* yang tidak memerlukan kabel untuk dihubungkan ke *console game*. Pemain dapat dengan bebas bermain dimana saja selama masih didalam jangkauan frekuensi dari *controller* ini.

4. Touch Screen

Touch screen umumnya digunakan pada perangkat *mobile*, seperti ponsel dan *Tablet PC*. Penggunaan *touch screen* sangat mudah karena pengguna hanya perlu menyentuh layar dari perangkat yang mereka miliki untuk melakukan interaksi apapun. *Controller* lain tidak dapat melakukan hal seperti *swipe*, *shake*, dan sebagainya karena hal tersebut hanya dapat dilakukan dengan *touch screen*.

Game yang dibuat akan menggunakan *touchscreen* sebagai *controller*-nya.

Unity

Unity adalah sebuah *game engine* dengan bermacam-macam *tools* yang dapat dimanfaatkan untuk memenuhi kebutuhan spesifik pengembang. (Unity, 2016) Unity merupakan *game engine* yang fleksibel untuk mengembangkan permainan 2D atau 3D *multiplatform*. (Unity, 2016) Unity merupakan *game engine* yang dipakai untuk membuat *game Summoner's Quest*.

C#

C# (*C Sharp*) merupakan bahasa pemrograman berorientasi objek (berbasis *class*) yang memungkinkan pengembang untuk membangun berbagai aplikasi yang aman dan kuat yang berjalan di .NET Framework. (Microsoft, 2016) C# dibuat sebagai bahasa pemrograman berorientasi objek yang lebih sederhana, dan modern. C# merupakan bahasa pemrograman yang digunakan dalam pembuatan *game Summoner's Quest*.

Touch Screen Gesture

Gestures dibagi menjadi Mekanika Sentuh dan Aktivitas Sentuh. (Google, 2016) Mekanika sentuh memiliki beberapa hasil tergantung konteks. Aktivitas sentuh dapat dicapai melalui beberapa mekanik sentuh. *Touch*, *Pinch*, *Drag*, *Swipe*, dan *Fling* adalah aktivitas sentuh yang paling sering dipakai dalam kegiatan yang berhubungan dengan *touchscreen*.

Touch Gesture Recognition

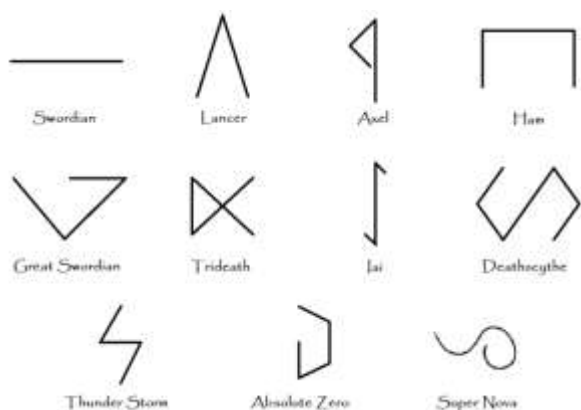
Gesture Recognition merupakan teknik untuk mengidentifikasi bentuk yang spesial tidak sebagai *handwriting input*, melainkan sebagai indikator dari suatu perintah khusus. (Liquisearch, 2016) Metode *Gesture recognition* yang akan digunakan berupa *point-cloud recognizer*. *Point-cloud recognizer* ini akan mengubah bentuk *input gesture* menjadi kumpulan titik atau *points*. *Point-cloud recognizer* ini hanya akan menerima *input* dari jari pertama yang menyentuh layar. Saat pemain menyentuh jari pada layar, *point-cloud recognizer* ini akan membaca *input* pemain setiap *frame*-nya sampai pemain berhenti menyentuh layar. *Input* ini akan dibaca sebagai titik-titik atau *points* dengan koordinat X dan Y. Koordinat *points* ini akan dibandingkan dengan koordinat *points* yang terdapat pada *files* yang telah disimpan sebelumnya dengan menggunakan metode *greedy*. Jika Gambar *input* pemain berbeda ukuran dengan gambar yang terdapat pada *files*, ukuran gambar tersebut akan diubah skalanya dengan proses normalisasi. *Pointinput* dan *point* yang terdapat pada *files* akan mengalami proses *resampling* agar jumlah *pointinput* dan yang terdapat pada *file* memiliki jumlah yang sama. *Gesture recognition* ini akan digunakan untuk menggambar suatu simbol atau bentuk tertentu pada layar sebagai *input* untuk mengeluarkan prajurit dan *ability* yang ada pada permainan yang dibuat.

ALUR APLIKASI

Permainan *Summoner's Quest* adalah permainan *bergenre* Tower Defense. Dalam permainan ini pemain ditugaskan untuk menghancurkan *base* musuh. Pemain menggunakan *touch screen* untuk memanggil prajurit, prajurit tersebut lalu akan berjalan ke arah *base* musuh dan akan melawan monster musuh yang menghalangi jalan. Pemain juga dapat menggunakan *Spell* untuk membantu membunuh musuh.

Pemain membutuhkan *magic power* untuk memanggil prajurit dan menggunakan *spell*. *Magic power* akan terus bertambah dengan sendirinya seiring waktu berjalan dalam permainan, selain itu pemain juga akan mendapat *magic power* ketika pemain berhasil membunuh monster musuh. Pemain juga akan mendapatkan *soul point* yang dapat

digunakan untuk meng-upgrade prajurit dan base dengan menyelesaikan *stage-stage* yang ada. Pemain harus menggambar simbol untuk mengeluarkan prajurit, simbol-simbol tersebut dapat dilihat pada **Gambar 1**. Pemain dapat memanggil 8 macam prajurit yang berbeda, status setiap prajurit dapat dilihat pada **Tabel 1**. Terdapat 6 jenis musuh yang dapat dilawan oleh pemain. Status setiap musuh dapat dilihat pada **Tabel 2**.



Gambar 1 Simbol-simbol sihir (rune)

Tabel 1 Status Prajurit

Name	Health Point (HP)	Damage	Movement Speed	Attack Type	Attack Speed	Summon Cost	Growth Rate (per level)
Swordian	250	70	3	Single	3	200	+50 HP, +40 Damage
Lancer	300	50	3	Area	3	300	+70 HP, +30 Damage
Axel	360	160	2	Single	4	600	+90 HP, +80 Damage
Ham	580	120	2	Area	4	700	+120 HP, +70 Damage
Great Swordian	1000	400	1.5	Area	6	1300	+400 HP,

Trideath	800	280	3	Area	3	1800	+120 Damage, +300 HP, +90 Damage
Lai	600	400	2.5	Area	2	1800	+200 HP, +100 Damage
Deathscythe	800	1000	2.5	Single	4	2500	+350 HP, +400 Damage

Tabel 2 Status Musuh

Name	Health Point (HP)	Damage	Movement Speed	Attack Type	Attack Speed	Magic Power when killed
Goblin	350	80	3	Single	3	100
Orc	750	175	3	Single	3	250
Golem	2000	250	2	Area	5	400
Griffin	2500	500	2	Single	6	1500
Vampire	1750	350	3	Area	3	800
The Great One	50000	1000	0.5	Area	8	30000

HASIL PENGUJIAN

Setelah melewati tahap pembuatan, *game* yang telah selesai dibuat akan memasuki tahap pengujian. Pengujian *game* dilakukan untuk memastikan bahwa *game* telah dapat berjalan dengan baik. Pengujian permainan Summoner's Quest dilakukan dengan 3 metode, yaitu *blackbox testing*, *alpha testing* dan *beta testing*.

Blackbox Testing

Pengujian dengan metode *blackbox testing* yang dilakukan pada permainan Summoner's Quest berupa pengujian terhadap modul dan juga

menjalankan permainan pada perangkat *smartphone* yang berbeda. Pengujian modul dilakukan untuk memastikan apakah setiap modul pada permainan sudah berjalan dengan baik. Modul-modul yang diuji pada *blackbox testing* adalah sebagai berikut:

1. Modul Main Menu

Pada modul *mainmenu* terdapat 3 buah tombol yaitu *Story Mode*, *Options*, *Exit*. Tombol *StoryMode* berfungsi untuk memindahkan pemain ke *menuStageSelect*. Tombol *Options* berfungsi untuk memindahkan pemain ke *menuOptions*.

2. Modul Main Menu

Pada modul *mainmenu* terdapat 3 buah tombol yaitu *Story Mode*, *Options*, *Exit*. Tombol *StoryMode* berfungsi untuk memindahkan pemain ke *menu Stage Select*. Tombol *Options* berfungsi untuk memindahkan pemain ke *menu Options*. Tombol *Exit* berfungsi untuk menghentikan permainan. Berdasarkan hasil pengujian, seluruh tombol sudah berjalan dengan baik dan mengembalikan hasil yang sesuai. Tampilan modul *main menu* dapat dilihat pada **Gambar 2**.



Gambar 2 Modul Main Menu

3. Modul Options

Pada modul ini pemain dapat mengatur volume permainan. Pemain juga dapat membaca informasi *ABOUT* dan *HELP* yang berisi tentang pembuat permainan dan informasi singkat tentang cara bermain. Seluruh fitur pada modul ini sudah berfungsi dengan benar dan sesuai. Tampilan modul *Options* dapat dilihat pada **Gambar 3**.



Gambar 3 Modul Options

4. Modul Stage Select

Pada modul ini pemain dapat memilih *stage* yang ingin dimainkan dengan menyentuh ikon bendera pada peta, kemudian pemain menyentuh tombol *Start* untuk memulai permainan. Pemain juga dapat berpindah ke menu *Upgrade* dengan menekan tombol *Upgrade*. Pengujian menunjukkan modul ini sudah berjalan dengan baik dan sesuai dengan yang diinginkan. Tampilan modul *Stage Select* dapat dilihat pada **Gambar 4**.



Gambar 4 Modul Stage Select

5. Modul Upgrade

Pada modul ini pemain dapat melihat informasi *status* setiap karakter. Pemain juga dapat melakukan upgrade pada karakter yang diinginkan dengan menekan tombol *upgrade* jika *Soul Point* yang dimiliki cukup untuk membayar *upgrade cost*. Pengujian pada modul ini menunjukkan modul ini sudah berjalan dengan baik dan sesuai. Tampilan modul *Upgrade* dapat dilihat pada **Gambar 5**.



Gambar 5 Modul Upgrade

6. Modul Gameplay

Pada modul ini *Magic Power* pemain akan terus bertambah seiring waktu, kemudian pemain dapat memanggil prajurit dengan menggambar simbol (*Rune*) jika *Magic Power* telah mencukupi jumlah yang telah ditentukan. Musuh-musuh akan terus bermunculan sampai *base* musuh berhasil dihancurkan oleh pemain. Jika pemain berhasil menghancurkan *basemusuh* maka pemain berhasil memenangkan *stage*. Jika *base* pemain berhasil dihancurkan oleh musuh maka pemain kalah. Berdasarkan hasil pengujian, modul sudah berjalan dengan baik dan sesuai. Tampilan modul *Gameplay* dapat dilihat pada Gambar 6.



Gambar 6 Modul Gameplay

Berdasarkan hasil pengujian, didapatkan hasil bahwa semua modul telah berjalan dengan baik dan terdapat kekurangan pada tampilan permainan saat

dimainkan pada perangkat yang tidak memiliki *display aspect ratio* 16:10. Tampilan modul gameplay saat bermain pada perangkat ASUS Zenfone 5 yang memiliki aspect ratio 16:9 dapat dilihat pada Gambar 7 dan tampilan modul gameplay saat bermain pada perangkat ASUS Nexus 7 (2013) yang memiliki aspect ratio 16:10 dapat dilihat pada Gambar 8.



Gambar 7 Tampilan bermain pada perangkat ASUS Zenfone 5



Gambar 8 Tampilan bermain pada perangkat ASUS Nexus 7 (2013)

Alpha testing

Alpha testing dilakukan oleh pihak yang mengerti bidang pemrograman. *Alpha testing* yang dilakukan pada permainan *Summoner's Quest* ini dilakukan oleh dua dosen pembimbing, sebagai pihak yang telah mengerti konsep dari permainan ini. Hasil *alpha testing* dapat dilihat pada Tabel 3. Dengan dilakukannya *alpha testing*, dapat diketahui bahwa masih terdapat kekurangan yang harus diperbaiki dan ditambah pada permainan *Summoner's Quest* ini. Kekurangan yang ditemukan pada saat *alpha*

testing telah diperbaiki sesuai dengan petunjuk pengujian agar game dapat berjalan dengan lebih baik.

Tabel 3 Hasil Alpha Testing

No	Penguji	Komentar	Solusi
1	Ir. Jeanny Pragantha, M.Eng	Tulisan diperbesar dan warna tulisan dibuat lebih kontras	Tulisan sudah diperbesar dan warna tulisan sudah dibuat lebih kontras
2	Darius Andana Haris, M.TI	Tambahkan animasi saat base hancur Musuhnya dipermudah agar lebih balance	Loading scene sudah diberi tanda Animasi base hancur ditambahkan Musuh sudah dipermudah

Beta testing

Beta testing adalah pengujian yang dilakukan kepada masyarakat awam setelah alpha testing diujikan. Pengujian ini dilakukan dengan membiarkan pengujian memainkan permainan Summoner's Quest. Pengujian ini dilakukan oleh 40 responden pada tanggal 14 Juni 2016 sampai 22 Juni 2016 dan tanggal 17 Juli 2016 sampai 19 Juli 2016. Pengujian dilakukan dengan menyebarkan kuesioner untuk diisi secara manual kepada 11 orang dan diisi secara langsung pada laptop yang telah disediakan kepada 20 orang. Selain itu juga dikirimkan file berupa data permainan dan kuesioner melalui facebook kepada 9 orang. Kuesioner berisi 7 pertanyaan yang terdiri dari pengetahuan mengenai permainan serupa, dan pendapat mengenai permainan Summoner's Quest. Hasil beta testing dapat dilihat pada **Tabel 4**.

Tabel 4 Hasil Beta Testing

No.	Kategori/Pertanyaan	Jawaban	Jumlah (orang)	Persentase
1	Jenis Kelamin	L	35	87,50%
		P	5	12,50%

2	Pernakah memainkan game ber-genre Tower Defense?	Pernah	27	67,50%
		Tidak Pernah	13	32,50%
3	Pernakah memainkan game dengan fitur untuk menggambar simbol tertentu sebagai input?	Pernah	21	52,50%
		Tidak Pernah	19	47,50%
4	Apakah tampilan status karakter di dalam game Summoner's Quest cukup jelas?	Ya	37	92,50%
		Tidak	3	7,50%
5	Pendapat tentang gameplay Summoner's Quest	Menarik	33	82,50%
		Biasa saja	7	17,50%
		Tidak Menarik	0	0%
6	Apakah tertarik untuk bermain lagi?	Ya	31	77,50%
		Tidak	9	22,50%
7	Apakah fitur untuk menggambar simbol sebagai input memberikan pengalaman baru?	Ya	22	55,00%
		Tidak	18	45,00%

Pembahasan Hasil Pengujian

Setelah selesai dilakukan pengujian beta testing, berdasarkan jawaban yang telah diberikan oleh 40 responden, terkumpul hasil sebagai berikut:

1. Sebanyak 67,50% responden menyatakan pernah bermain permainan yang serupa.
2. Sebanyak 47,50% responden menyatakan tidak pernah memainkan permainan dengan fitur menggambar simbol tertentu sebagai input.
3. 92,50% responden menyatakan tampilan status karakter di dalam permainan Summoner's Quest cukup jelas.
4. Sebanyak 82,50% menyatakan gameplay permainan Summoner's Quest menarik.

5. 77,50% responden menyatakan tertarik untuk bermain lagi.
6. Sebanyak 55,00% responden menyatakan menggambar simbol sebagai input memberikan pengalaman yang baru.

Berdasarkan hasil pengujian, responden dengan pekerjaan sebagai mahasiswa atau karyawan swasta memberikan komentar dan saran bahwa fitur menggambar simbol pada permainan Summoner's Quest merupakan fitur yang unik dan menarik, tetapi perlu adanya perbaikan dalam kualitas tampilan, animasi, dan suara pada permainan ini. Sedangkan responden dengan pekerjaan sebagai pelajar memberikan komentar dan saran bahwa fitur menggambar simbol menarik, tetapi garisnya perlu dipertebal dan permainan terasa pelan karena gerakan karakter yang terlihat lambat. Selain itu juga dilakukan pengujian pada perangkat yang berbeda. Pengujian ini dilakukan pada perangkat ASUS Zenfone 5 dan Sony Xperia TX yang memiliki *display aspect ratio* 16:9, dan ASUS Nexus 7 (2013) yang memiliki *display aspect ratio* 16:10. Pada perangkat dengan *display aspect ratio* 16:9, tampilan *status* karakter dan gambar tanah pada modul *gameplay* terpotong sebagian sehingga nama setiap karakter dan *spell* menjadi tidak terlihat, selain itu animasi *Spell* terlihat tidak menyentuh tanah (lihat pada **Gambar 7**). Tetapi jika permainan dimainkan pada perangkat ASUS Nexus 7 (2013) yang memiliki aspect ratio 16:10 seluruh layar tampil sempurna (lihat pada **Gambar 8**).

7. Pengujian Touch Gesture Recognizer dilakukan dengan menggambar simbol-simbol yang ada dan mengamati apakah output yang ditampilkan sesuai dengan input simbol. Berdasarkan hasil pengamatan saat pengujian, Touch Gesture Recognizer pada permainan ini sudah berfungsi dengan cukup baik. Touch Gesture Recognizer ini hanya menerima input dari jari pertama yang menyentuh layar, sehingga tidak mungkin untuk menggambar 2 simbol secara bersamaan.

KESIMPULAN DAN SARAN

Setelah selesai melakukan pengujian pada permainan Summoner's Quest, berdasarkan data dan komentar yang diterima dapat ditarik kesimpulan sebagai berikut:

1. Fitur menggambar simbol sebagai input pada permainan Summoner's Quest merupakan fitur yang unik dan menarik.
2. Permainan Summoner's Quest memiliki *gameplay* yang menarik sehingga pemain tertarik untuk bermain lagi.
3. Pada karakter The Great One, Griffin, Golem dan Orc, animasi jalannya terlihat tidak lancar.
4. Permainan Summoner's Quest tampil sempurna jika dimainkan pada perangkat dengan layar yang memiliki *aspectratio* 16:10.
5. Touch Gesture Recognizer pada permainan Summoner's Quest sudah berjalan dengan cukup baik.

Selain kesimpulan, terdapat juga saran-saran yang dapat digunakan untuk mengembangkan *game* Summoner's Quest. Saran-saran yang didapat adalah sebagai berikut:

1. Tampilan prajurit-prajuritnya dapat dibuat lebih bervariasi dan unik.
2. Simbol-simbol yang terdapat di dalam permainan dipermudah bentuknya dan diperbanyak jenisnya.

Tampilan permainan diperbaiki penskalaannya agar dapat dimainkan pada perangkat yang memiliki layar dengan *aspectratio* 16:9.

DAFTAR PUSTAKA

- Bates, B. (2004). *Game Design 2nd Edition*. Boston: Thomson Course Technology.
- Dark Watcher. (2016, Februari 24). *Game Controller*. Retrieved from Video Games Console Library: <http://www.videogameconsolelibrary.com/art-controller.htm#page=motion>
- Google. (2016, February 25). *Gestures*. Retrieved from Google: <https://www.google.com/design/spec/patterns/gestures.html>

- Liquisearch. (2016, February 25). *Pen Computing - General Techniques of Pen Computing - Gesture Recognition*. Retrieved from Liquisearch: http://www.liquisearch.com/pen_computing/general_techniques_of_pen_computing/gesture_recognition
- Microsoft. (2016, February 25). *Introduction to the C# Language and the .NET Framework*. Retrieved from Microsoft MSDN: <https://msdn.microsoft.com/en-us/library/z1zx9t92.aspx>
- Sethi, M. (2005). *Game Programming for Teen Second Edition*. Boston: Course Technology PTR.
- Techopedia. (2016, Februari 24). *Gameplay*. Retrieved from Techopedia: <https://www.techopedia.com/definition/1911/gameplay>
- Unity. (2016, Februari 25). *The Best Development Platform for Creating Games*. Retrieved from Unity 3D: <http://unity3d.com/unity>
- Unity. (2016, Februari 25). *Unity Overview*. Retrieved from Unity 3D: <http://docs.unity3d.com/Manual/UnityOverview.html>