

PERANCANGAN APLIKASI PERMAINAN “LET’S SAY” DENGAN INTERAKSI PENGENALAN UCAPAN

Andhika Dewanta¹⁾, R. Rizal Isnanto²⁾, Kurniawan Teguh Martono²⁾
Program Studi Sistem Komputer, Fakultas Teknik, Universitas Diponegoro
Jalan Prof. Sudharto, Tembalang, Semarang, Indonesia

Abstract – *Computer games has been widespread. Currently, almost all devices that have a screen can play the game with the experience of interactivity to entertain the public. The game does not always have a negative effect. Game application that has a learning function will be useful and easily accepted by society. This study was conducted to determine how computer games can be a fun medium of information and learning by making use of speech recognition interaction.*

The application development of this "Let's Say" game is done by using the method of multimedia development consisting stages of concept, design, material collection, implementation, testing, and distribution. The design of the application is done using UML modeling and ERD. Application development is done using Visual Basic.NET programming language, WPF system interface, and SQLite database. Application utilizes Windows Speech Recognition as a speech recognition engine. Functional testing of the application is done using black box methods. Applications testing to users is done through a questionnaire.

The test results showed that the application works well according to functions that has been specified. Application testing to users indicate that the application is informative and easy to use with a view that is quite interesting. Tests also showed that the use of speech recognition has been going well and helpful.

Index Terms : *game, speech recognition, Visual Basic.NET, WPF, SQLite*

I. PENDAHULUAN

Permainan tidak selalu memberikan dampak negatif, banyak manfaat yang dapat diperoleh, di antaranya digunakan sebagai terapi untuk melatih refleks tangannya bagi orang yang terkena stroke, belajar mengenal huruf bagi anak usia dini maupun sebagai media pembelajaran bagi anak berkebutuhan khusus. Permainan komputer juga dapat dijadikan sebagai salah satu media pembelajaran interaktif masa kini bagi masyarakat. Sebagian besar orangtua menggunakan permainan sebagai media alternatif pembelajaran bagi anak-anak.

Pengembangan aplikasi permainan dapat dilakukan dengan mempertimbangkan peluang tersebut. Aplikasi permainan yang memiliki fungsi pembelajaran akan bermanfaat dan mudah diterima masyarakat. Tema

permainan yang dapat diambil antara lain adalah pengenalan tempat pariwisata di sekitar kita. Aplikasi permainan akan lebih menarik dengan menggunakan interaktivitas tambahan, seperti pengenalan ucapan.

Pengenalan ucapan merupakan suatu pengembangan teknik dan sistem yang memungkinkan komputer untuk menerima masukan berupa kata yang diucapkan. Teknologi ini memungkinkan suatu perangkat untuk mengenali dan memahami kata-kata yang diucapkan dengan cara digitalisasi kata dan mencocokkan sinyal digital tersebut dengan suatu pola tertentu yang tersimpan dalam suatu perangkat.

Tujuan dari Tugas Akhir ini adalah mengembangkan permainan sebagai alat bantu informasi berbasis *desktop*, mengimplementasikan pengenalan ucapan pada aplikasi permainan yang dikembangkan, dan memberikan pengetahuan umum tentang wisata melalui interaksi dan grafis permainan yang menarik.

Agar tidak menyimpang dari permasalahan, Tugas Akhir ini memiliki batasan masalah antara lain aplikasi dikembangkan berbasis *desktop* untuk sistem operasi Windows, aplikasi dikembangkan menggunakan bahasa pemrograman Visual Basic.NET dengan kerangka kerja .NET versi 4 dan basis data lokal SQLite, aplikasi memanfaatkan Windows Speech Recognition sebagai mesin pengenalan ucapan yang akan digunakan, dan lingkup tema aplikasi permainan adalah beberapa tempat pariwisata yang terdapat di beberapa kota di Jawa Tengah.

II. TINJAUAN PUSTAKA

2.1 Permainan Serius

Permainan serius (*serious games*) dirancang dengan tujuan meningkatkan beberapa aspek tertentu dari pembelajaran. Permainan serius digunakan dalam pelatihan layanan darurat, pelatihan militer, pendidikan perusahaan, perawatan kesehatan, dan banyak sektor masyarakat lainnya. Mereka juga dapat ditemukan di setiap jenjang pendidikan, di semua jenis sekolah dan universitas di seluruh dunia. Genre permainan, kompleksitas, dan platform sangat beragam seperti yang ditemukan dalam permainan kasual. Bermain adalah kontributor penting bagi pengembangan manusia dan belajar adalah bahan wajib dalam permainan serius.^[3]

2.2 Pengenalan Ucapan

Pengenalan ucapan (*speech recognition*) memungkinkan pengguna memberikan masukan ke suatu aplikasi melalui ucapan. Seperti mengklik menggunakan mouse, mengetik menggunakan papan ketik, atau menekan tombol telepon seluler untuk memberikan masukan, pengenalan ucapan memungkinkan pengguna memberikan masukan dengan berbicara. Di dunia desktop, pengguna membutuhkan mikrofon untuk melakukannya.

Proses pengenalan ucapan dilakukan oleh komponen perangkat lunak yang disebut mesin pengenalan ucapan (*speech recognition engine*). Fungsi utama mesin pengenalan ucapan adalah memproses masukan yang diucapkan dan menerjemahkannya menjadi teks yang dipahami suatu aplikasi. Aplikasi kemudian dapat menginterpretasikan hasil pengenalan sebagai perintah. Aplikasi ini merupakan aplikasi perintah dan kontrol. Aplikasi menginterpretasikan hasil pengenalan sebagai teks biasa disebut aplikasi dikte.^[6]

2.3 Visual Basic.NET

Visual Basic .NET (VB.NET) adalah pengembangan Microsoft dari Visual Basic untuk platform .NET. VB.NET berangkat dengan beberapa cara signifikan yang berbeda dari versi Visual Basic sebelumnya. VB.NET telah berkembang menjadi paket pengembangan perangkat lunak berorientasi komersial.

VB.NET memiliki sejumlah fitur yang membantu menjaga kompatibilitas versi sebelumnya dengan Visual Basic 6 (VB6). Fitur lain telah ditambahkan secara spesifik untuk mengadaptasikan Visual Basic dengan pemrograman berorientasi objek dan *platform* .NET.

VB.NET menyediakan dukungan pada bahasa pemrograman untuk mencari *bug* di awal proses pengembangan. Ini dibuat agar kode lebih mudah dijaga dan program lebih handal. VB.NET tidak mendukung beberapa fitur yang tersedia di bahasa pemrograman lain yang dibuat untuk kode yang tidak aman.^[5]

2.4 Extensible Application Markup Language

Extensible Application Markup Language (XAML) adalah bahasa markah (markup language) berbasis XML untuk menspesifikasikan dan mengatur karakteristik dari kelas-kelas. Kebanyakan program WPF menggunakan C# dan XAML.

XAML digunakan utamanya untuk menspesifikasikan aspek statik dan visual suatu antarmuka dengan mengatur properti objek-objek kelas. C# digunakan utamanya untuk menspesifikasikan kebanyakan bagian aktif dari program seperti aliran kontrol dan penanganan pesan.

Sebagai contoh, pengembang mungkin menggunakan XAML untuk meletakkan tombol pada suatu jendela dan mengatur semua karakteristik visualnya. Namun, pengembang akan menggunakan C# untuk menulis kode penanganan pesan yang akan dipanggil jika tombol ditekan.

Seperti semua XML, XAML menspesifikasikan suatu elemen yang mengandung elemen lain. Struktur ini membentuk suatu pohon di mana setiap elemen dapat mengandung elemen lain.^[9]

2.5 Microsoft Speech API

Speech API memungkinkan pengembang menulis aplikasi berbasis Windows untuk menggunakan pengenalan ucapan dan TTS. API ini dispesifikasikan kumpulan objek OLE Component Object Model (COM). Penggunaan OLE membuat pengenalan ucapan siap tersedia bagi pengembang yang akan menulis aplikasi dengan Visual Basic, C/C++, atau bahasa pemrograman lain yang dapat mengakses objek OLE secara langsung atau melalui otomasi.

Speech API ditujukan sebagai standar antarmuka aplikasi bagi pengembang dan vendor mesin untuk menulis kode. Pemrogram dapat menulis aplikasi tanpa perlu khawatir tentang mesin yang digunakan, vendor mesin dapat mendapatkan kompatibilitas instan dengan semua aplikasi pengenalan ucapan, dan pengguna memperoleh kebebasan untuk memilih mesin pengenalan ucapannya sendiri sesuai keuangan dan kebutuhan kemampuan. Speech API menyediakan jalan keluar untuk mengakses mesin spesifik ketika pengembang ingin melakukan sesuatu yang khusus.^[10]

III. PERANCANGAN DAN IMPLEMENTASI APLIKASI

Pengembangan aplikasi permainan ini dilakukan dengan menggunakan metode pengembangan multimedia yang terdiri atas tahap konsep, desain, pengumpulan materi, implementasi, pengujian, dan distribusi.

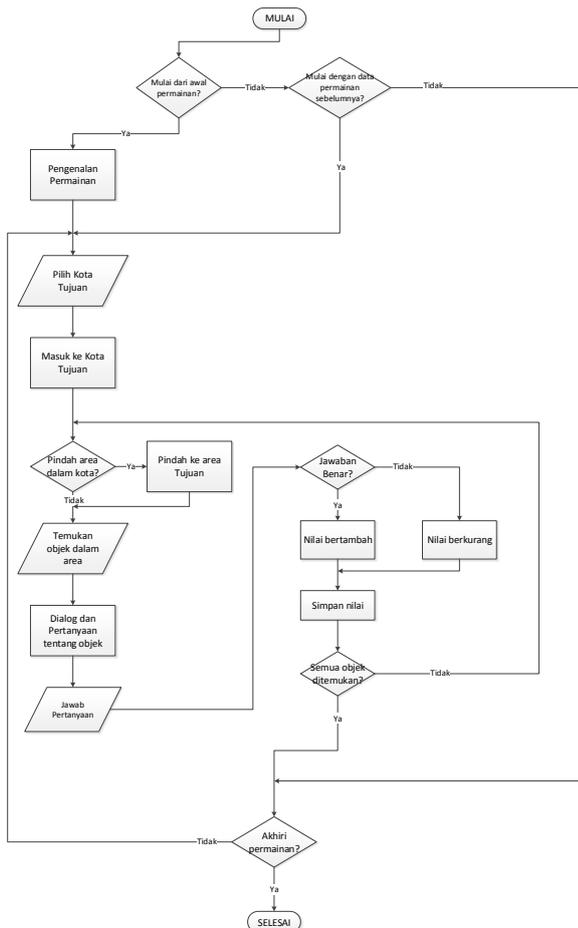
3.1. Konsep dan Logika Permainan

Permainan akan menyajikan peta utama dengan beberapa kota yang dapat pemain kunjungi. Di dalam kota yang sedang dikunjungi, pemain akan melihat bagaimana keadaan beberapa area di kota tersebut sambil berdialog dengan karakter teman.

Pemain harus mencari objek-objek yang dapat ditunjukkan kepada karakter teman. Setiap objek yang ditemukan akan menghasilkan dialog antara pemain dengan karakter teman. Dialog juga mungkin memuat pertanyaan yang harus dijawab pemain. Benar atau tidaknya jawaban yang pemain berikan akan mempengaruhi nilai kesenangan dari karakter teman. Untuk menjawab pertanyaan tersebut, pemain mengucapkan langsung jawaban yang dipilihnya melalui mikrofon.

Setiap informasi yang diperoleh selama perjalanan dalam permainan akan dicatat pada sebuah menu. Semakin banyak informasi yang ditemukan, catatan akan semakin lengkap. Pemain akan membuat ensiklopedianya sendiri untuk versi permainan ini.

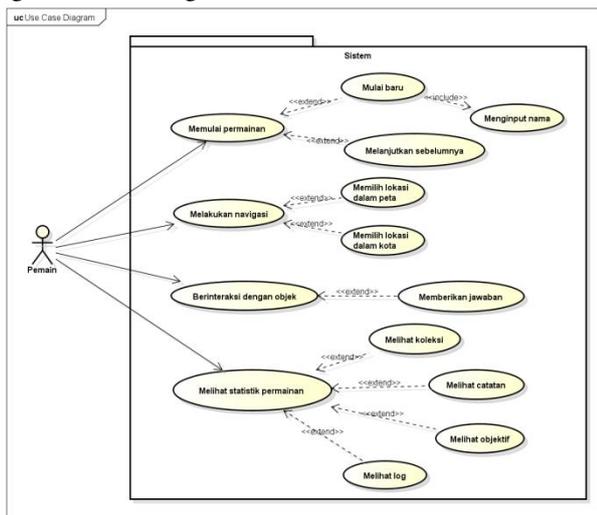
Logika permainan digambarkan pada diagram alir sesuai Gambar 1.



Gambar 1 Diagram Alir Permainan

3.2. Use Case Aktivitas

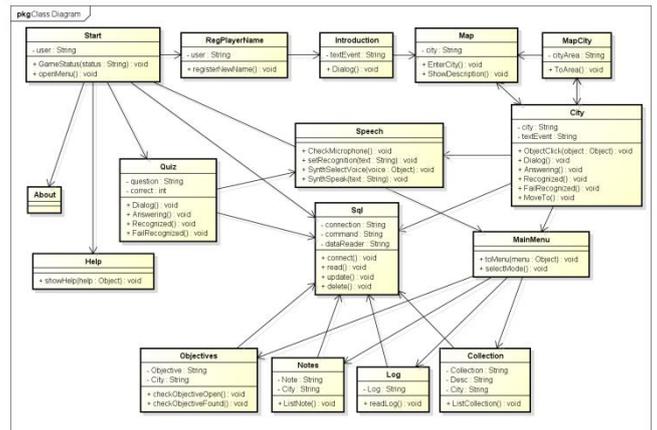
Fungsi yang harus dipenuhi dalam aplikasi permainan didefinisikan melalui *use case*. Diagram *use case* aplikasi digambarkan sebagai berikut.



Gambar 2 Diagram Use Case

3.3. Diagram Kelas

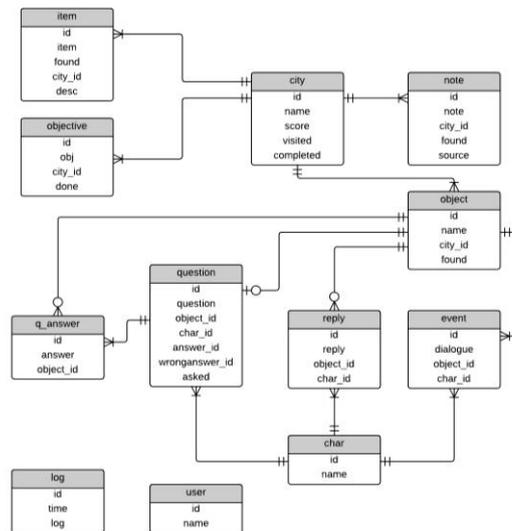
Interaksi antarkelas pada aplikasi digambarkan dengan diagram kelas. Diagram kelas perancangan aplikasi ini ditunjukkan pada Gambar 3.



Gambar 3 Diagram Kelas

3.4. Relasi Entitas Data

Data dan hubungan antardata pada aplikasi digambarkan pada diagram relasi entitas. Diagram relasi entitas perancangan aplikasi ini ditunjukkan pada Gambar 4.

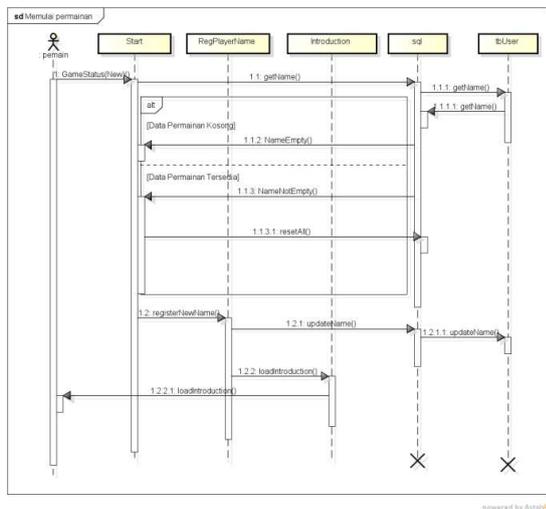


Gambar 4 Diagram Relasi Entitas

3.5. Aktivitas Program

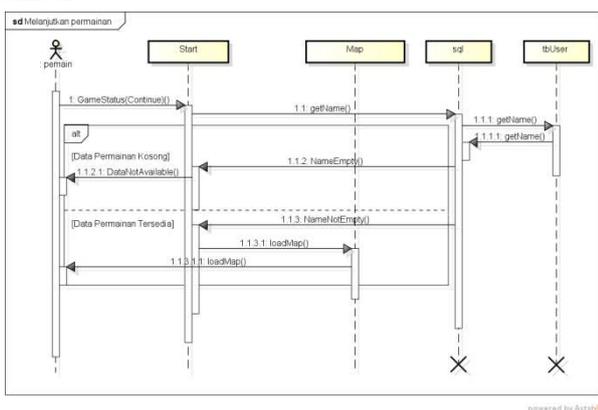
Pemain menekan tombol **START** pada kelas utama (*Main*) **Start** dan mengirim pesan **GameStatus(New)**. Kelas **Start** akan mengirim pesan **getName()** ke tabel data **tbUser** melalui kelas **sql**. Data yang diperoleh dikembalikan ke kelas **sql** dan diteruskan kelas **Start**. Jika data dari pesan **getName()** tidak kosong, kelas **Start** akan mengirim pesan **resetAll()** untuk memperbarui data menjadi kosong sebelum memulai permainan. Kelas **Start** kemudian akan mengirim pesan **registerName()** untuk membuka kelas **RegPlayerName**. Pemain mengetikkan nama pemain yang baru dan menekan tombol **Next**. Kelas **RegPlayerName** akan mengirim pesan **updateName()** ke kelas **sql** untuk memperbarui nama pemain pada tabel data **tbUser**. Kelas **RegPlayerName** mengirim pesan

loadIntroduction() untuk membuka kelas Introduction dan ditampilkan kepada pemain. Urutan proses tersebut digambarkan melalui diagram sekuensial pada Gambar 5.



Gambar 5 Diagram sekuensial memulai permainan

Pemain menekan tombol **CONTINUE** pada kelas Start dan mengirim pesan GameStatus(Continue). Kelas Start akan mengirim pesan getName() ke tabel data tbUser melalui kelas sql. Data yang diperoleh dikembalikan ke kelas sql dan diteruskan kelas Start. Jika data dari pesan getName() kosong, kelas Start akan mengirim pesan NameEmpty() kepada pemain. Jika data dari pesan getName() tidak kosong, kelas Start akan mengirim pesan loadMap() untuk membuka kelas Map dan ditampilkan kepada pemain. Urutan proses tersebut digambarkan melalui diagram sekuensial pada Gambar 6.

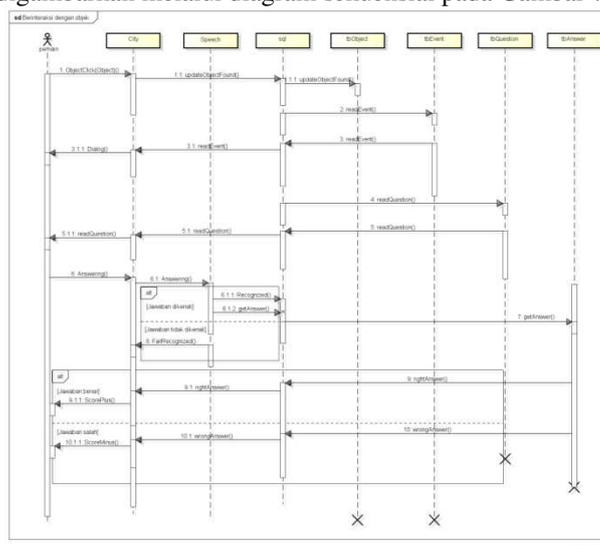


Gambar 6 Diagram sekuensial melanjutkan permainan

Pemain menekan objek yang ditemukan pada kelas City dan mengirim pesan ObjectClick(Object). Kelas City mengirim pesan updateObjectFound() ke kelas sql dan meneruskannya untuk memperbarui data pada tabel tbObject. Kelas sql mengirim pesan readEvent() ke tbEvent untuk mengambil dialog yang muncul untuk objek tersebut. Data diteruskan dan ditampilkan oleh City ke pemain melalui pesan Dialog(). Kelas sql mengirim pesan readQuestion() ke tbQuestion untuk mengambil pertanyaan atau

pernyataan untuk objek tersebut. Data diteruskan dan ditampilkan oleh City ke pemain melalui pesan readQuestion().

Pemain memilih jawaban untuk pertanyaan atau pernyataan tersebut dan mengirim pesan Answering(). Kelas City mengirim pesan getAnswer() ke kelas sql dan meneruskannya ke tbAnswer untuk mencocokkan jawaban pilihan pemain. Jika jawaban pemain benar, sql akan mengirim pesan rightAnswer() ke kelas City dan mengirim pesan ScorePlus() kepada pemain. Jika jawaban pemain salah, sql akan mengirim pesan wrongAnswer() ke kelas City dan mengirim pesan ScoreMinus() kepada pemain. Urutan proses tersebut digambarkan melalui diagram sekuensial pada Gambar 7.



Gambar 7 Diagram sekuensial berinteraksi dengan objek

3.6. Implementasi Program

Aplikasi permainan menggunakan Windows Speech Recognition yang memanfaatkan Speech API. Pada sistem operasi Windows 7 atau yang lebih tinggi, Speech API telah tersedia dan dapat langsung digunakan dalam pemrograman. Pengenalan suara membutuhkan paket pustaka System.Speech.Recognition. Paket pustaka tersebut memungkinkan kita menggunakan objek dari kelas SpeechRecognitionEngine. Objek dari kelas tersebut memungkinkan penggunaan berbagai properti, metode, dan fungsi sesuai kebutuhan aplikasi yang dikembangkan.

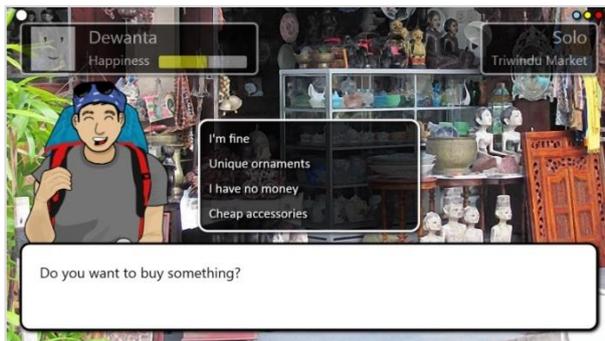
Aplikasi permainan akan memanfaatkan Text-to-Speech (TTS) yang telah tersedia di dalam sistem operasi Windows. TTS digunakan untuk mengiringi dialog yang disampaikan oleh karakter teman sehingga pemain dapat merasakan dan mendengar dialog yang disampaikan karakter teman. Implementasi TTS dalam pemrograman Visual Basic membutuhkan paket pustaka System.Speech dan System.Speech.Synthesis. Paket pustaka tersebut memungkinkan kita menggunakan objek dari kelas SpeechSynthesizer. Objek dari kelas tersebut memungkinkan penggunaan berbagai properti, metode, dan fungsi sesuai kebutuhan aplikasi yang dikembangkan.

IV. PENGUJIAN APLIKASI DAN PEMBAHASAN

4.1 Pembahasan Aplikasi

Pemain bertugas mencari objek yang terdapat di setiap kota. Setiap objek yang ditemukan akan memicu dialog yang membahas objek tersebut.

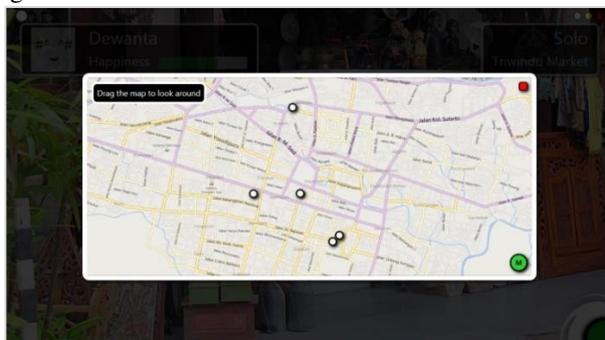
Permainan memiliki beberapa dialog yang memuat pertanyaan atau pernyataan yang dapat dijawab oleh pemain. Pertanyaan atau pernyataan tersebut ditandai dengan munculnya kotak dialog berisi 4 jawaban yang dapat dipilih oleh pemain. Gambar 8 menunjukkan contoh pernyataan dan pilihan jawaban yang dapat dipilih oleh pemain.



Gambar 8 Dialog karakter dengan pemain

Pemain harus menjawab pertanyaan atau pernyataan tersebut dengan membaca salah satu jawaban yang dipilih melalui mikrofon. Karakter permainan akan memberikan reaksi dari jawaban yang dipilih pemain. Jawaban yang dipilih juga akan mempengaruhi perubahan nilai pemain.

Pemain dapat memilih kota dan area yang dikunjungi. Setiap kota memiliki beberapa area yang dapat dikunjungi oleh pemain. Area yang dapat dikunjungi bergantung pada masing-masing kota. Pemain menekan tombol **Going to** yang terletak di bagian bawah kanan layar. Kotak dialog berisi area pilihan akan muncul pada layar. Pemain akan membuka area pilihan dengan menekan tombol area yang dipilih. Gambar 9 menunjukkan navigasi area dalam kota.



Gambar 9 Navigasi ke area lain

Pemain dapat melihat statistik permainan saat ini melalui jendela **Menu**. Menu utama diakses melalui tombol **Menu** di kiri atas layar. Jendela **Menu** memuat statistik permainan saat ini dan sub menu yang dapat dipilih. Gambar 10 menunjukkan jendela **Menu**.



Gambar 10 Menu utama dan statistik permainan

4.2 Pengujian Aplikasi

Pengujian aplikasi dilakukan menggunakan komputer dengan spesifikasi sebagai berikut.

1. Prosesor utama Intel Core i3 dengan kecepatan 1,8 GHz
2. Memori 4 GB
3. Media penyimpanan 40 GB
4. Prosesor grafis Intel HD Graphics 4000
5. Monitor dengan resolusi 1366 x 768 pixel
6. Mikrofon internal Realtek HD Audio
7. *Mouse* standar
8. *Keyboard* standar
9. Sistem operasi Windows 8
10. Perangkat lunak klien .NET Framework 4

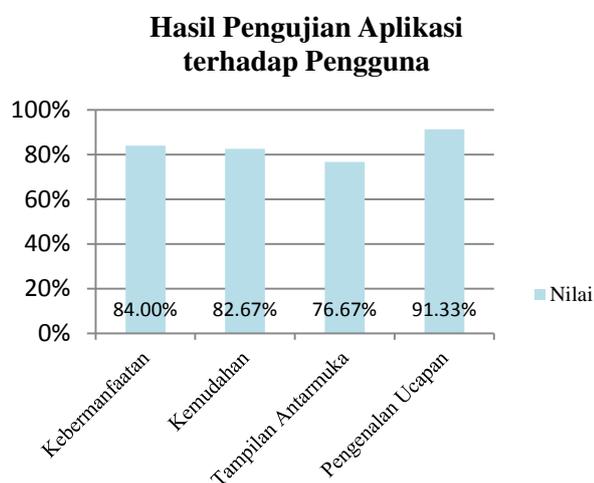
Pengujian fungsional program dilakukan untuk mengetahui apakah fungsi yang dispesifikasikan dalam perancangan permainan dapat berjalan dengan baik. Fungsi yang diuji dan hasil pengujian ditunjukkan pada Tabel 1.

Tabel 1 Pengujian fungsional

No.	Fungsi	Hasil
1.	Memulai permainan dari awal	Berhasil
2.	Memasukkan dan menyimpan nama pemain	Berhasil
3.	Melanjutkan permainan sebelumnya	Berhasil
4.	Melakukan navigasi pada peta	Berhasil
5.	Melakukan navigasi dalam kota	Berhasil
6.	Berinteraksi dengan objek dalam kota	Berhasil
7.	Memberikan jawaban melalui pengenalan ucapan	Berhasil
8.	Melihat statistik permainan	Berhasil
9.	Melihat objektif dalam masing-masing kota	Berhasil
10.	Melihat catatan perjalanan	Berhasil
11.	Melihat koleksi benda	Berhasil
12.	Melihat log permainan	Berhasil

Pengujian pengguna dilakukan untuk mengetahui ketercapaian tujuan aplikasi permainan bagi pemain. Pengguna diminta memainkan permainan ini kemudian menjawab beberapa pertanyaan pada kuesioner yang diajukan kepada mereka. Kuesioner memuat pertanyaan yang ditujukan untuk mengetahui manfaat, tampilan, dan

kemudahan dalam mengoperasikan aplikasi permainan ini bagi pemain. Pengujian pengguna dilakukan kepada 30 orang pengguna. Hasil pengujian aplikasi permainan terhadap pengguna ditunjukkan pada diagram batang sesuai Gambar 8.



Gambar 11 Pengujian pengguna

Pengujian pengenalan ucapan dilakukan untuk mengetahui apakah aplikasi permainan mampu mengenali kata atau frase yang telah ditentukan dalam permainan. Kata atau frase yang diuji ditunjukkan dalam tabel pengujian pada Tabel 2. Aplikasi dapat mengenali kata atau frase, namun beberapa kata atau frase mungkin perlu diucapkan beberapa kali agar dapat dikenali.

Tabel 2 Pengujian pengenalan ucapan

No.	Kata/Frase	Dikenali
1	<i>Japanese</i>	Ya
2	<i>Of course</i>	Ya
3	<i>It is below the sea surface</i>	Ya
4	<i>Yes, it is</i>	Ya
5	<i>That is bad</i>	Ya
6	<i>Do you want to rent?</i>	Ya
7	<i>Why?</i>	Ya
8	<i>Let's buy some</i>	Ya
9	<i>It can't be helped</i>	Ya
10	<i>I hope so</i>	Ya
11	<i>Lame</i>	Ya
12	<i>Prophets</i>	Ya
13	<i>They are huge</i>	Ya
14	<i>It is cheap</i>	Ya
15	<i>Polder</i>	Ya
16	<i>A lighthouse</i>	Ya
17	<i>Let's go</i>	Ya
18	<i>No</i>	Ya
19	<i>It is rude</i>	Ya

20	<i>What will you do?</i>	Ya
----	--------------------------	----

V. PENUTUP

5.1. Kesimpulan

Dari hasil perancangan aplikasi permainan dengan interaksi pengenalan suara ini, dapat disimpulkan hal-hal sebagai berikut.

1. Aplikasi permainan "Let's Say" dengan interaksi pengenalan suara telah berjalan sesuai fungsi yang telah dispesifikasikan.
2. Aplikasi dapat memberikan informasi bagi pengguna. Hal ini ditunjukkan dengan nilai 84% untuk aspek keinformatifan dari pengujian yang telah dilakukan.
3. Aplikasi mudah digunakan bagi pengguna. Hal ini ditunjukkan dengan nilai 82,67% untuk aspek kemudahan penggunaan dari pengujian yang telah dilakukan.
4. Aplikasi memiliki tampilan yang cukup menarik. Hal ini ditunjukkan dengan nilai 76,67% untuk aspek tampilan antarmuka dari pengujian yang telah dilakukan.
5. Implementasi pengenalan ucapan pada aplikasi ini memberikan dampak yang positif bagi pengguna. Hal ini ditunjukkan dengan nilai 91,33% untuk aspek pengenalan ucapan dari pengujian yang telah dilakukan.

5.2. Saran

Aplikasi permainan dengan interaksi pengenalan suara ini memungkinkan pengembangan lebih lanjut. Beberapa saran yang dapat diberikan adalah sebagai berikut.

1. Aplikasi permainan saat ini memanfaatkan Windows Speech Recognition dan menggunakan bahasa Inggris. Perlu dilakukan penelitian lanjutan dengan membuat aplikasi serupa yang dikembangkan menggunakan mesin pengenalan ucapan lain serta menggunakan Bahasa Indonesia.
2. Optimalisasi penggunaan mesin pengenalan ucapan dapat dilakukan antara lain dengan melatih mesin pengenalan ucapan, mencari tempat dengan kebisingan minimal, menggunakan perangkat mikrofon atau *headset* yang canggih.

DAFTAR PUSTAKA

- [1] Cai, C. J., *Adapting Existing Games for Education using Speech Recognition*, Massachusetts Institute of Technology, Massachusetts, 2013.
- [2] Crawford, C., *The Art of Computer Game Design*. McGraw-Hill, 1982.
- [3] Derryberry, A., *Serious Games: Online Games for Learning*. Adobe System Inc., California, 2007.
- [4] Frädrieh, L. dan D. Anastasiou, "Siri vs. Windows Speech Recognition", *Transl. J.*, vol. 16, no. 3, 2012.

- [5] Grundgeiger, D., Programming Visual Basic .NET, First Edit., O'Reilly & Associates, Inc., California, 2002.
- [6] Kemble, K. A., An Introduction to Speech Recognition. IBM Corp, 2001.
- [7] Kreibich, J. A., Using SQLite. C: O'Reilly, 2010.
- [8] Lunden, I., Spell Up , a Chrome Experiment, Uses Voice Recognition, Gaming to Improve Your English, <http://techcrunch.com/2014/05/13/spellup-a-chrome-experiment-uses-voice-recognition-game-theory-to-improve-your-english/>, 3 Juni 2014.
- [9] MacDonald, M., Pro WPF 4.5 in VB, Apress, New York, 2012.
- [10] Rozak, M., "Talk to Your Computer and Have It Answer Back with the Microsoft Speech API," January 1996 Issue Microsoft Syst. J., 1996.
- [11] Solis, D. M., Illustrated WPF, Apress, New York, 2009.
- [12] Thai, T. dan H. Q. Lam, NET Framework Essentials, O'Reilly & Associates, Inc., California, 2001.
- [13] ---, Say-N-Play, <http://www.saynplay.com/>, 3 Juni 2014.