

**Rancang Bangun *Mobile Game* Pengenalan Tanaman Herbal
pada Platform Windows Phone**

SKRIPSI

Untuk memenuhi sebagian persyaratan untuk mencapai gelar Sarjana Komputer



Disusun Oleh:

ARIANTY ANGGRAINI

NIM. 0910683020

KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN

UNIVERSITAS BRAWIJAYA

PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER

PROGRAM STUDI INFORMATIKA / ILMU KOMPUTER

MALANG

2014

LEMBAR PERSETUJUAN

**RANCANG BANGUN *MOBILE GAME* PENGENALAN TANAMAN
HERBAL PADA PLATFORM WINDOWS PHONE**

SKRIPSI

Untuk memenuhi sebagian persyaratan untuk mencapai gelar Sarjana Komputer



Disusun Oleh :

ARIANTY ANGGRAINI

NIM. 0910683020

Telah diperiksa dan disetujui oleh :

Dosen Pembimbing I

Dosen Pembimbing II

Eriq M. Adams J. , S. T. , M. Kom

Wibisono Sukmo Wardhono, S. T. ,M. T

NIK. 850410 06 1 1 0027

NIK. 820404 06 1 1 0091

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Tuhan Yang Maha Esa, karena berkat rahmat dan hidayah-Nya, penulis dapat menyelesaikan penyusunan proposal skripsi dengan judul “Rancang Bangun *MobileGame* Pengenalan Tanaman Herbal Pada Platform Windows Phone.”

Penulisan proposal ini disusun untuk sebagian persyaratan memperoleh gelar Sarjana Komputer di Program Studi Teknik Informatika Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya Malang. Dalam penyusunan proposal ini telah banyak mendapat bantuan dari berbagai pihak.

Ucapan terima kasih penulis sampaikan kepada:

1. ALLAH SWT beserta junjungannya nabi Muhammad SAW.
2. Orangtua penulias atas dukungannya.
3. Bapak Eriq Muhammad Adams J, ST M. Kom dan bapak Wibisono Sukmo Wardhono, ST MT. Selaku pembimbing penulisan skripsi ini.
4. Keluarga besar alm. Bapak Satrolan.
5. Bapak Ir. Sutrisno, MT, selaku ketua Program Teknik Informasi dan Ilmu Komputer. Bapak Arief Andy Soebroto, ST, M. Kom, selaku dosen pembimbing akademik. Bapak Drs. Marji, MT.
6. Dan semua Dosen PTIIK Universitas Brawijaya. Terima kasih atas ilmu yang telah diberikan. Beserta seluruh staff dan karyawannya. Laboran-laborannya terutama mas didit.
7. Teman-teman penulis atas semua dukungan semangatnya.

Dan semua pihak yang tidak bisa disebutkan satu persatu. Harapan penulis semoga proposal ini dapat memberikan manfaat pada semua pihak, terutama kepada penulis dan kalangan yang mengembangkan aplikasi ini.

Malang, Februari 2014

Penulis

ABSTRAK

Arianty Anggraini. 2014. : Rancang Bangun *MobileGame* Pengenalan Tanaman Herbal pada Platform Windows Phone. Skripsi Program Studi Teknik Informatika, Program Teknologi Informasi dan Ilmu Komputer, Universitas Brawijaya, Malang. Dosen Pembimbing : Eriq M. Adams J. , S. T. , M. Kom. dan Wibisono Sukmo Wardhono, S. T. , M. T.

Indonesia merupakan negara agraris dimana berbagai tanaman, mulai dari sayur dan buah-buahan yang kaya akan manfaat, tumbuh dengan subur. Namun sayangnya banyak masyarakat yang kurang mengetahui manfaat dari tanaman tersebut. Maka dari itu dibuatlah *game adventure* untuk mengenalkan sekaligus mengetahui manfaat yang terkandung dalam tanaman tersebut yang dikemas dengan mudah dan menyenangkan. Permainan ini dibuat di tengah minimnya pengetahuan masyarakat tentang tanaman yang semakin hari meninggalkan cara pengobatan tradisional. Mereka cenderung memilih pengobatan modern yang menggunakan obat-obatan kimia karena dipengaruhi oleh majunya dunia kesehatan terutama dibidang farmakologi.

Permainan edukasi ini dibuat melalui tahap *game concept design* dan *technical design*. Implementasi permainan menggunakan bahasa pemrograman C# dan framework XNA. Untuk pembuatan objek-menggunakan Adobe Flash CS6 dan Adobe Fireworks CS6. Permainan ini dapat diaplikasikan pada platform *windows phone*. Pengujian aplikasi ini menggunakan metode *black-box testing* dengan strategi pengujian validasi. Pengujian aplikasi ini menggunakan metode *white-box testing* meliputi pengujian unit dan integrasi. Pengujian performa permainan dilakukan melalui pengujian *Frame Per Second* (FPS) dan dihasilkan rata-rata FPS game yaitu 30 FPS. Dari hasil pengujian didapatkan kesimpulan bahwa permainan ini telah memenuhi kebutuhan fungsional dan non fungsional.

Kata Kunci : Permainan Edukasi, tanaman herbal, XNA

ABSTRACT

Arianty Anggraini. 2014. :MobileGame Development of Introducing Herbs on Windows Phone. Undergraduate Thesis of Informatic Study Program, Information Technology and Computer Science Program, Brawijaya University, Malang. Advisor : Eriq M. Adams J. , S. T. , M. Kom and Wibisono Sukmo Wardhono, S. T. , M. T.

Indonesia is an agricultural country where a variety of plants, start from vegetables and fruits which provide a lot of usefulness growth well. But unfortunately, many people still lack of information about the usefulness of these plants. So that's why this adventure game is made to introduce and also to know the benefits that contained in the plant that is packed with easy and fun way. This game is made in the middle of public knowledge deficiency about the plants, which is increasingly leaving traditional treatments. They tend to choose modern medicine that uses chemical drugs because they are influenced by the advance of the hygiene framework, especially in the field of pharmacology.

This educational game created through the stages of game concept design and technical design. This game's Implementation use C# programming language and the XNA framework. For the making of the object, we use Adobe Flash CS6 and Adobe Fireworks CS6. This game can be applied to windows phone platform. This application use black - box testing with strategy of validation testing. This application testing use white - box testing includes unit testing and integration testing. Testing is done by testing the performance of the game Frames Per Second (FPS) and the result is it produced an average of 30 FPS in this game. From the test results it was concluded that this game has met the functional and non- functional requirements.

Keywords : education game, herbs, XNA

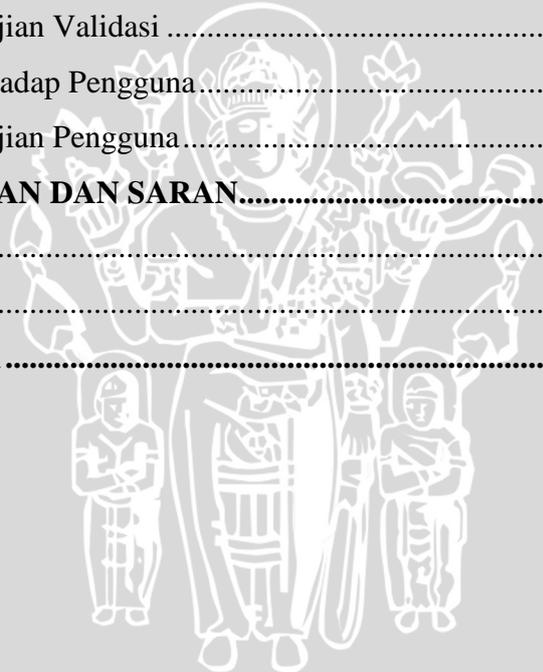
DAFTAR ISI

DAFTAR ISI.....	iv
DAFTAR GAMBAR.....	vii
DAFTAR TABEL	ix
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	2
1.4 Batasan Masalah.....	2
1.5 Manfaat	2
1.6 Sistematika Penulisan	3
BAB II DASAR TEORI.....	5
2.1 Permainan.....	5
2.2 Asset Maker	5
2.2.1 Adobe Flash	5
2.2.2 Adobe Firework	6
2.3 Bahasa pemrogramana C#.....	8
2.4 Windows Phone : Visual Studio XNA.....	8
2.5 Tanaman Herbal	10
2.6 Unified Modelling Language	11
2.6.1 Diagram Use case.....	11
2.4.2 Diagram Class	12
1.4.3 Diagram <i>Activity</i>	13
2.7 Pengujian Perangkat Lunak.....	14
2.7. 1 Teknik Pengujian	14
2.7. 2 Strategi Pengujian	16
BAB III METODE PENELITIAN DAN PERANCANGAN	19
3.1 Metode Penelitian.....	19
3.1.1 Studi Literatur	19



3.1.2	Perancangan	20
3.1.3	Implementasi	20
3.1.4	Evaluasi dan analisa (pengujian).....	21
3.2	Perancangan	21
3.2.1	<i>Game Concept Design</i>	21
3.2.2	Technical Design.....	31
BAB IV IMPLEMENTASI		39
4.1	Spesifikasi Sistem	39
4.1.1	Spesifikasi Lingkungan Perangkat Keras.....	39
4.1.2	Spesifikasi Lingkungan Perangkat Lunak.....	39
4.1.3	Batasan-Batasan Dalam Implementasi.....	40
4.2	Implementasi Antarmuka serta BGM	40
4.2.1	Implementasi Karakter Poka.....	40
4.2.2	Implementasi Karakter Ibu.....	41
4.2.3	Implementasi Karakter Pak Wuluh	41
4.2.4	Implementasi Karakter Coco.....	41
4.2.5	Implementasi Karakter Oryza	42
4.2.6	Implementasi Karakter Rica.....	42
4.2.7	Implementasi MainMenu	42
4.2.8	Implementasi <i>help screen</i>	43
4.2.9	Implementasi <i>credit screen</i>	43
4.2.10	Implementasi <i>option screen</i>	43
4.2.11	Implementasi <i>Environment</i>	45
4.3	Implementasi Prosedur Program	47
4.3.1	Implementasi Untuk Menampilkan Dialog Percakapan	47
4.3.2	Implementasi untuk <i>Hit item</i>	48
4.3.3	Implementasi untuk menampilkan <i>Random puzzle</i>	49
4.3.4	Implementasi untuk Menampilkan Level.....	50
4.4	Implementasi Musik / <i>Soundtrack</i>	51
BAB V PENGUJIAN DAN ANALISA.....		53
5.1	Pengujian Unit.....	53
5.1.1	Pengujian Unit Untuk Dialog Percakapan	53

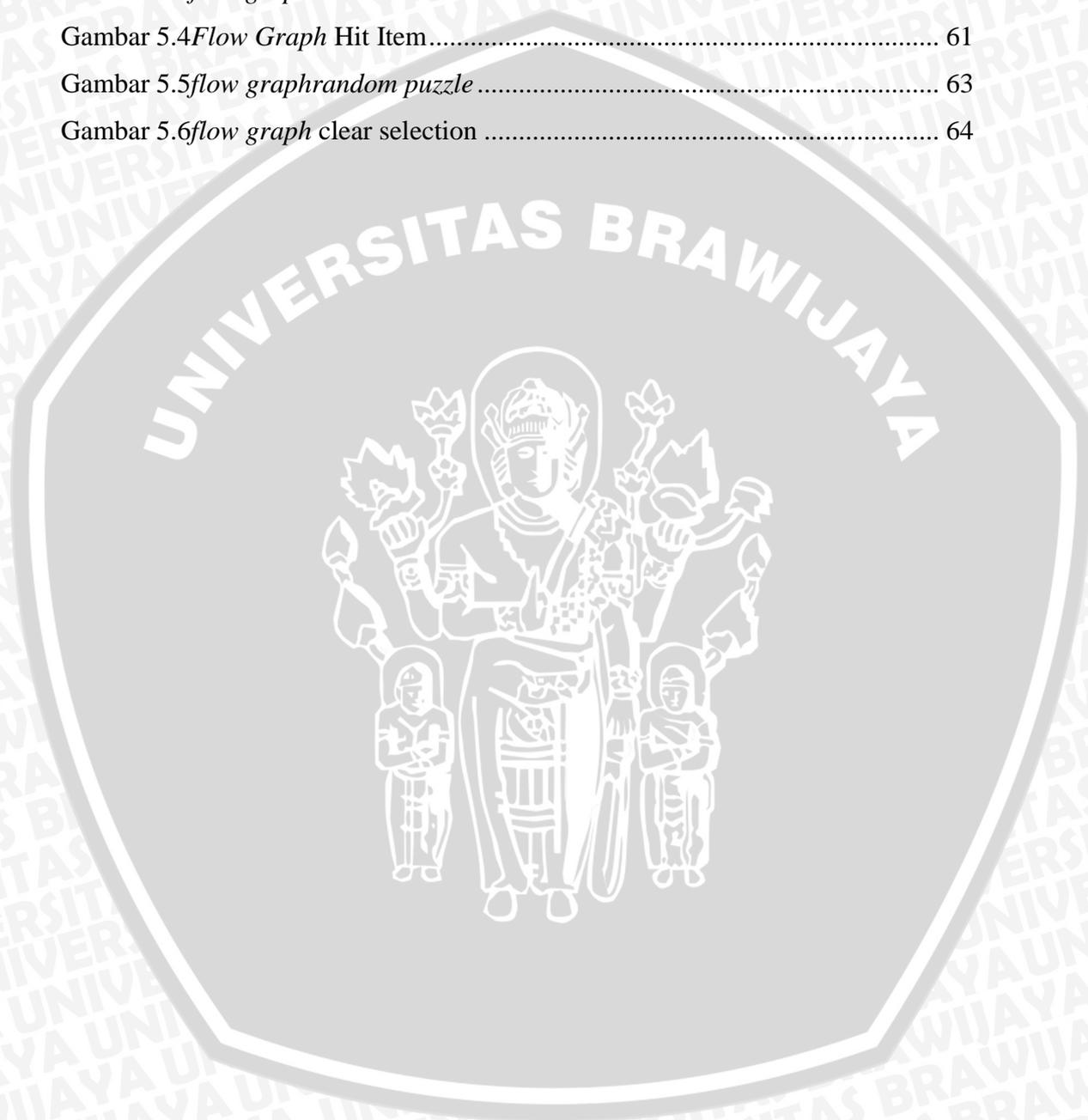
5.1.2	Pengujian Unit Fungsi Menampilkan background.....	54
5.1.3	Pengujian Unit Inisialisasi koordinat Level	56
5.1.4	Pengujian Unit Memilih Level.....	57
5.2	Analisis Pengujian Unit	59
5.3	Pengujian Integrasi.....	59
5.3.1	Pengujian <i>Hit item</i>	60
5.3.2	Pengujian Unit <i>Random puzzle</i>	62
5.3.3	Pengujian Unit <i>Clear Selection</i>	63
5.4	Analisis Pengujian Integrasi.....	65
5.5	Pengujian Validasi	65
5.5.1	Kasus uji validasi dan hasil pengujian	65
5.6	Analisis Pengujian Validasi	67
5.7	Pengujian Terhadap Pengguna.....	68
5.8	Analisis Pengujian Pengguna.....	68
BAB VI KESIMPULAN DAN SARAN.....		70
6.1	Kesimpulan	70
6.2	Saran.....	71
DAFTAR PUSTAKA		DP-1



DAFTAR GAMBAR

Gambar 2.1 Tampilan Awal Adobe Flash	6
Gambar 2.2 Tampilan Awal Adobe Fireworks	7
Gambar 2.3 Tampilan Screen yang menggunakan Adobe Fireworks.....	7
Gambar 2.4 Tampilan Awal Visual Studio XNA	9
Gambar 2.5 Alur kerja Method XNA	10
Gambar 2.6 Contoh Diagram Use Case	12
Gambar 2.7 Contoh Diagram Kelas	12
Gambar 2.8 Contoh Diagram Aktivitas	13
Gambar 3.1 Flowchart Pengerjaan Penelitian	19
Gambar 3.2 storyboard	24
Gambar 3.3 Logo Universitas Brawijaya	25
Gambar 3.4 Logo PTIIK	25
Gambar 3.5 Logo <i>Game Lab</i>	26
Gambar 3.6 <i>Game Flow</i>	26
Gambar 3.7 <i>Use Case Diagram Game</i> poka si tabib cilik	33
Gambar 3.8 <i>Class Diagram</i>	34
Gambar 3.9 <i>Activity Diagram splashscreen</i>	36
Gambar 3.10 <i>Activity Diagram</i> melihat <i>story</i>	37
Gambar 4.1 Poka	41
Gambar 4.2 Ibu	41
Gambar 4.3 Pak Wuluh	41
Gambar 4.4 Coco	42
Gambar 4.5 Oryza	42
Gambar 4.6 Rica	42
Gambar 4.7 Implementasi <i>main menu</i>	44
Gambar 4.8 Implementasi <i>Help screen</i>	44
Gambar 4.9 Implementasi credit screen	44
Gambar 4.10 Implementasi <i>option screen</i>	45
Gambar 4.11 Menemukan buku untuk mengakhiri segmen pertama	45
Gambar 4.12 Halaman Belakang Tempat Mencari Tanaman	46

Gambar 4.13 <i>Puzzle</i> yang harus Disusun.....	46
Gambar 5.1 <i>Flow graph</i> Pengujian Dialog Percakapan.....	54
Gambar 5.2 <i>flow graph</i> fungsi menampilkan background.....	55
Gambar 5.3 <i>flow graph</i> memilih level	58
Gambar 5.4 <i>Flow Graph</i> Hit Item.....	61
Gambar 5.5 <i>flow graph</i> random puzzle	63
Gambar 5.6 <i>flow graph</i> clear selection	64

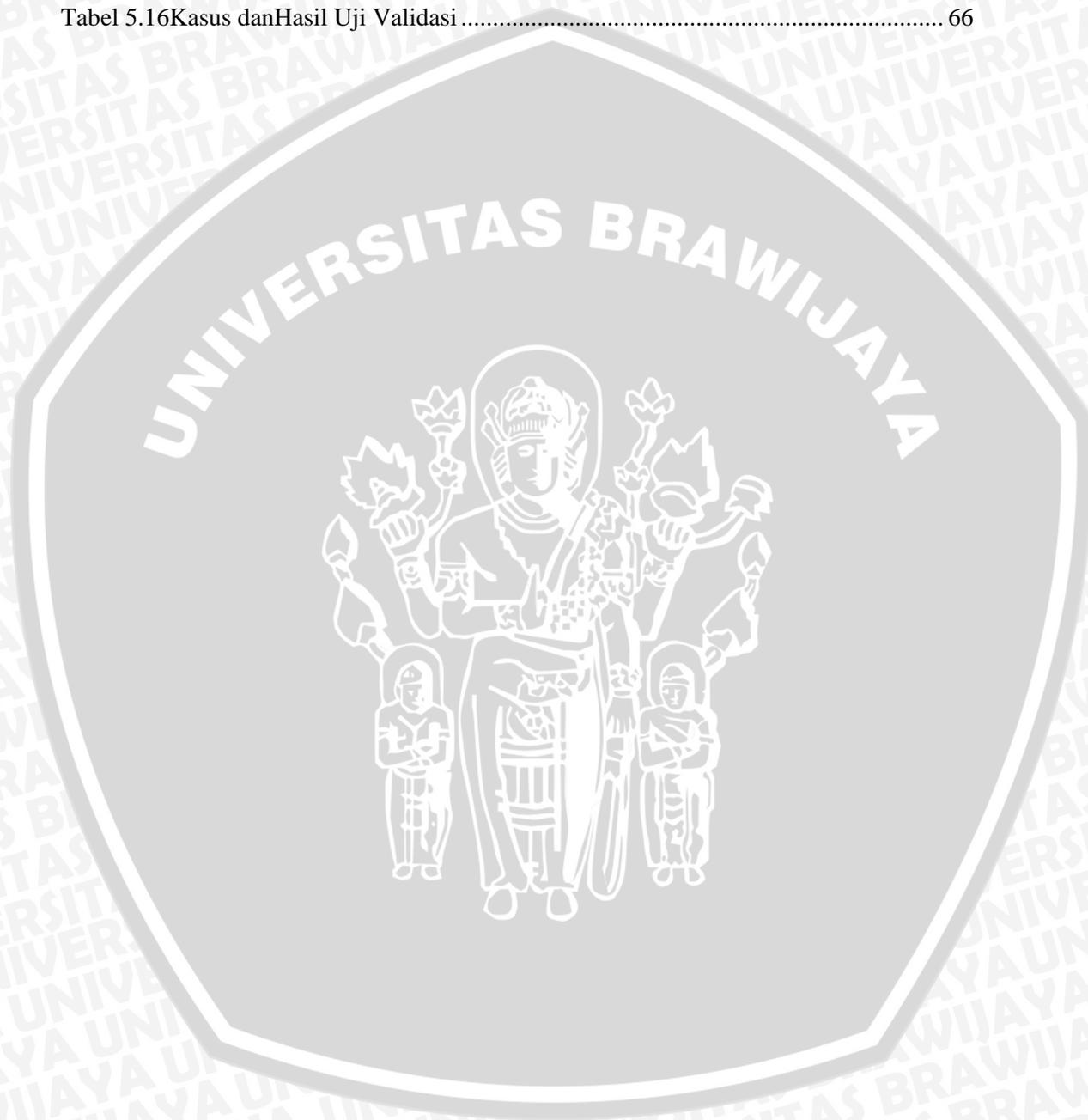


DAFTAR TABEL

Tabel 3.1 Implementasi Pembuatan Permainan	21
Tabel 3.2 Konsep Utama Permainan	22
Tabel 3.3 Daftar Kebutuhan Teknologi	22
Tabel 3.4 Rancangan Antarmuka Menu Utama	27
Tabel 3.5 Karakter Poka	27
Tabel 3.6 Karakter ibu	28
Tabel 3.7 Karakter Pak wuluh	29
Tabel 3.8 Karakter Coco	29
Tabel 3.9 Karakter Oryza	29
Tabel 3.10 Karakter Rica	30
Tabel 3.11 Deskripsi Level	30
Tabel 3.12 Identifikasi Aktor	31
Tabel 3.13 Deskripsi Daftar Kebutuhan	31
Tabel 3.14 Deskripsi class Diagram	34
Tabel 4.1 Spesifikasi Lingkungan perangkat keras komputer	39
Tabel 4.2 Spesifikasi Lingkungan Perangkat Lunak Komputer	40
Tabel 4.3 Prosedur Menampilkan Dialog Percakapan	47
Tabel 4.4 Prosedur Hit Item	48
Tabel 4.5 Prosedur <i>Random puzzle</i>	49
Tabel 4.6 Prosedur Menampilkan Level	50
Tabel 5.1 Permodelan <i>Flow graph</i> Algoritma Dialog Percakapan	53
Tabel 5.2 Hasil pengujian Dialog Percakapan	54
Tabel 5.3 Permodelan <i>flow graph</i> menampilkan fungsi <i>background</i>	55
Tabel 5.4 Hasil pengujian Menampilkan background	56
Tabel 5.5 Clear <i>Puzzle</i>	56
Tabel 5.6 Koordinat Level	56
Tabel 5.7 Hasil Pengujian Mendapatkan Koordinat Level	57
Tabel 5.8 Memilih Level	57
Tabel 5.9 Hasil Pengujian Memilih Level	59
Tabel 5.10 Pengujian <i>Hit item</i>	60
Tabel 5.11 Hasil Pengujian <i>Hit item</i>	61



Tabel 5.12	Prosedur <i>Random puzzle</i>	62
Tabel 5.13	Pengujian <i>random puzzle</i>	63
Tabel 5.14	Prosedur pengujian Clear selection.....	64
Tabel 5.15	Hasil pengujian Clear Selection.....	65
Tabel 5.16	Kasus dan Hasil Uji Validasi	66



BAB I PENDAHULUAN

1.1 Latar Belakang

Indonesia mendapat julukan sebagai negara agraris karena memiliki tanah yang subur. Sehingga bukan hal yang mencengangkan apabila berbagai tanaman mulai dari sayur dan buah-buahan yang kaya akan manfaat tumbuh dengan mudahnya di Indonesia. Namun sayangnya banyak masyarakat yang kurang mengetahui manfaat dari tanaman tersebut [WIJ-08:3]. Padahal kita bisa memanfaatkan tanaman tersebut untuk pengobatan secara tradisional. Biasanya cara tradisional lebih baik dibandingkan dengan obat-obatan kimia. Hanya saja nampaknya cara tradisional kini mulai ditinggalkan dan dilupakan. Mereka cenderung memilih pengobatan modern yang menggunakan obat-obatan kimia karena dipengaruhi oleh majunya dunia kesehatan terutama dibidang farmakologi.

Indonesia sendiri memiliki cara pengobatan yang paling baik perkembangannya dan paling tinggi tingkatnya di dunia ini. Cara pengobatan ini meliputi apotek hidup. Pengobatan tradisional umumnya lebih aman daripada obat-obatan modern karena sifatnya tidak terlalu keras [WER-10:01]. Sebagai contoh, biasanya orang yang menderita masuk angin akan membuat air hangat yang ditambahkan jahe. Ada juga yang menggunakan perasan jeruk nipis yang diberi sedikit kecap untuk yang sakit batuk.

Tergesernya cara tradisional juga tidak lepas dari pengaruh canggihnya teknologi saat ini. Perangkat telepon genggam yang telah dimiliki oleh hampir seluruh masyarakat mulai dari anak sekolah, tukang penjual makanan keliling hingga tentu saja pegawai kantoran, ini menandakan bahwa telepon genggam bukan hal yang asing lagi. Dan umumnya telepon genggam pasti memiliki fitur permainan.

Maka dari itu dibuatlah *game adventure* untuk mengenalkan sekaligus mengetahui manfaat yang terkandung dalam tanaman tersebut yang dikemas dengan mudah dan menyenangkan. Membuat permainan yang membantu penggunaanya dalam mengenali tanaman herbal serta dapat mengolahnya menjadi obat yang bermanfaat. Harapannya dengan adanya aplikasi ini dapat membantu

penggunanya mencoba pengobatan secara herbal terlebih dahulu sebelum menggunakan obat-obatan dari bahan kimia yang bisa menimbulkan efek samping.

1.2 Rumusan Masalah

Dari uraian yang telah disampaikan pada latar belakang diatas, maka dapat diambil beberapa rumusan masalah diantaranya :

1. Bagaimana merancang permainan yang sesuai dengan keminatan terhadap pengetahuan tentang tanaman herbal?
2. Mengimplementasikan *game play* permainan pengenalan tanaman herbal?
3. Bagaimana menguji permainan pengenalan tanamanyang telah dibuat?

1.3 Tujuan

Tujuan dari pembuatan skripsi ini adalah membangun *mobile game* pengenalan tanaman herbal Indonesia. Dengan judul “Poka si tabib Cilik”.

1.4 Batasan Masalah

Agar tidak menyimpang dari maksud dan tujuan, maka penyusunan skripsi ini dibatasi menjadi pokok – pokok permasalahan sebagai berikut:

1. Permainan yang dibangun merupakan permainan dua dimensi (2D).
2. Implementasi permainan menggunakan *windows phone* versi 7.0 keatas.
3. Tanaman dan resep obat yang digunakan dalam game berasal dari resep tradisional Indonesia. Serta tanaman yang digunakan dalam game hanya untuk penyakit tertentu menyesuaikan cerita dan berasal juga dari buku resep tradisional Indonesia.

1.5 Manfaat

1. Bagi penulis:
 - a. Mengaplikasikan ilmu yang didapat selama mengikuti perkuliahan di Teknik Informatika Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya.
 - b. Mendapatkan pengetahuan tentang pembuatan *mobile game* pada Windows Phone berplatform XNA versi 4.1.

2. Bagi pengguna:
 - a. Mendapatkan wawasan akan pengimplementasian dari aplikasi pembuatan permainan pembelajaran ini.
 - b. Mendapatkan wawasan akan bagaimana cara kerja metode ini untuk menentukan membantu menyampaikan materi pelajaran kepada siswa dengan cara yang ringan dan menyenangkan ini.
 - c. Mendapatkan hiburan yang bermanfaat.

1.6 Sistematika Penulisan

Sistematika penulisan dalam skripsi ini sebagai berikut:

BAB I Pendahuluan

Memuat latar belakang, rumusan masalah, tujuan, batasan masalah, manfaat tujuan, dan sistematika penulisan.

BAB II Dasar Teori

Membahas teori dasar dan teori penunjang yang berhubungan dengan pembuatan permainan pada *Windows Phone*.

BAB III Metodologi dan Perancangan

Berisi tentang langkah-langkah dalam merancang pembuatan permainan. Serta membahas analisis kebutuhan dan perancangan yang sesuai dengan teori yang ada dalam membangun permainan agar sesuai dengan keinginan penulis serta bermanfaat bagi pengguna.

BAB IV Implementasi

Membahas tentang implementasi permainan yang dibangun. Meliputi pembuatan *environment*, tekstur, alur permainan, menu permainan menggunakan software-software yang dibutuhkan. Tidak terkecuali pemrograman permainan menggunakan bahasa pemrograman C#.

BAB V Pengujian

Pada bab ini memuat hasil pengujian serta analisis terhadap permainan yang telah dibuat.

BAB VI Penutup

Pada bab ini berisi kesimpulan yang diambil berdasarkan analisa hal-hal penting serta saran-saran untuk penyempurnaan dan pengembangan dari aplikasi yang dibuat.



BAB II DASAR TEORI

2.1 Permainan

Game, atau dalam bahasa Indonesia disebut dengan permainan adalah aktivitas yang dimainkan untuk olahraga atau hiburan menurut aturan. Seperti yang kita ketahui, pada permainan terdapat juga kondisi menang atau kalah [MAH-10].

Permainan dikategorikan menjadi beberapa tipe atau yang disebut dengan *genre* yaitu *action*, *adventure*, *casual*, *educational*, *role-playing game* (RPG), *simulation*, *sports* dan *strategy*. *Game mobile* pengenalan tanaman herbal poka si tabib cilik adalah permainan yang memiliki *genre adventure*.

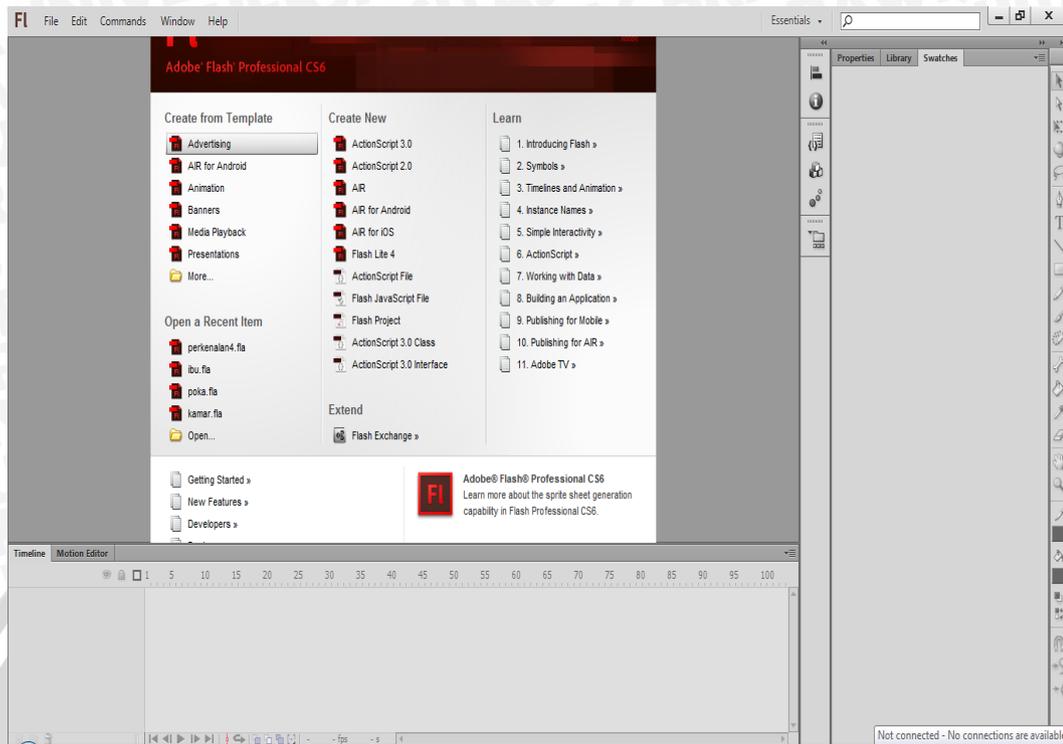
Secara umum, *adventure* adalah tipe permainan yang lebih mengarah ke pemecahan teka teki, mengumpulkan barang-barang dan manajemen inventory. Dan identiknya permainan petualangan itu berbasis teks atau cerita.

2.2 Asset Maker

Pembuatan permainan poka si tabib cilik tidak lepas dari beberapa software yang digunakan untuk membuat objek 2D. Pembuatan objek menggunakan lebih dari satu *software* dengan beberapa pertimbangan. Salah satunya adalah karena setiap *software* memiliki kelebihan dan kekurangan. Sehingga dibutuhkan lebih dari satu *software* yang dapat menutupi *software* lainnya.

2.2.1 Adobe Flash

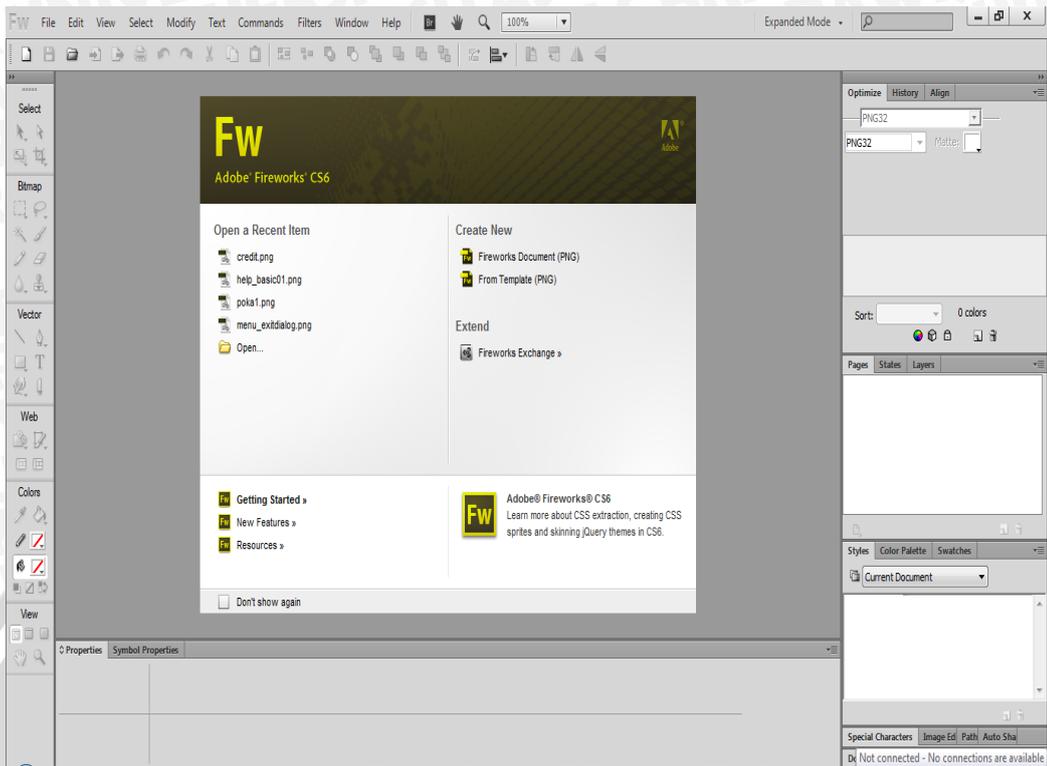
Adobe Flash merupakan sebuah program yang didesain khusus oleh Adobe dan program aplikasi standar authoring tool professional yang digunakan untuk membuat animasi dan bitmap yang sangat menarik untuk keperluan pembangunan situs web yang interaktif dan dinamis. Flash didesain dengan kemampuan untuk membuat animasi 2 dimensi yang handal dan ringan sehingga flash banyak digunakan untuk membangun dan memberikan efek animasi pada website, CD Interaktif dan yang lainnya. Gambar 2.1 menunjukkan tampilan dari Adobe Flash CS6.



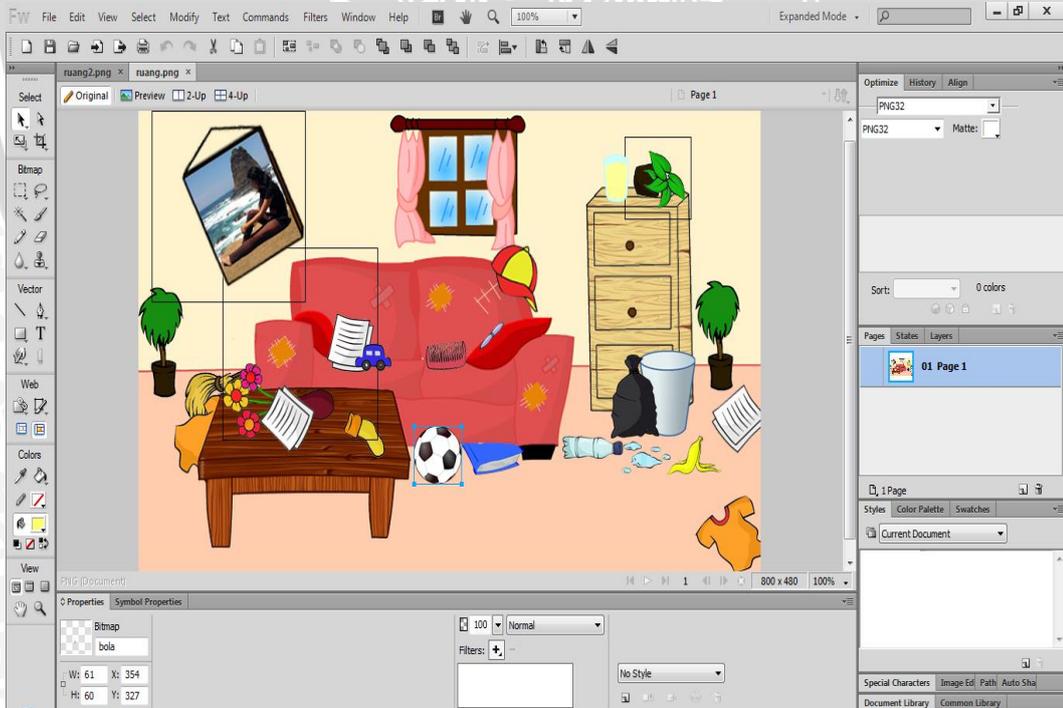
Gambar 2.1 Tampilan Awal Adobe Flash

2.2.2 Adobe Firework

Hampir sama seperti Adobe Photoshop, Adobe ini juga bekerja berdasarkan *pixel* serta *vector*. Alasan penulis menggunakan Adobe Fireworks, karena di beberapa *stage* penulis membutuhkan informasi mengenai letak koordinat. Karena permainan yang dibuat berbasis mencari barang. Maka penulis membutuhkan informasi tentang lokasi koordinat letak barang tersebut. Serta terdapat beberapa gambar yang diatur agar letaknya berpindah. Bukan menghilang. Jadi Adobe Fireworks sangat membantu bagi anda yang membutuhkan data mengenai letak koordinat. Tampilan awal dan tampilan *gameplay* menggunakan Adobe Fireworks CS6 akan ditunjukkan oleh gambar 2.2 dan gambar 2.3.



Gambar 2.2 Tampilan Awal Adobe Fireworks



Gambar 2.3 Tampilan Screen yang menggunakan Adobe Fireworks

2.3 Bahasa pemrograman C#

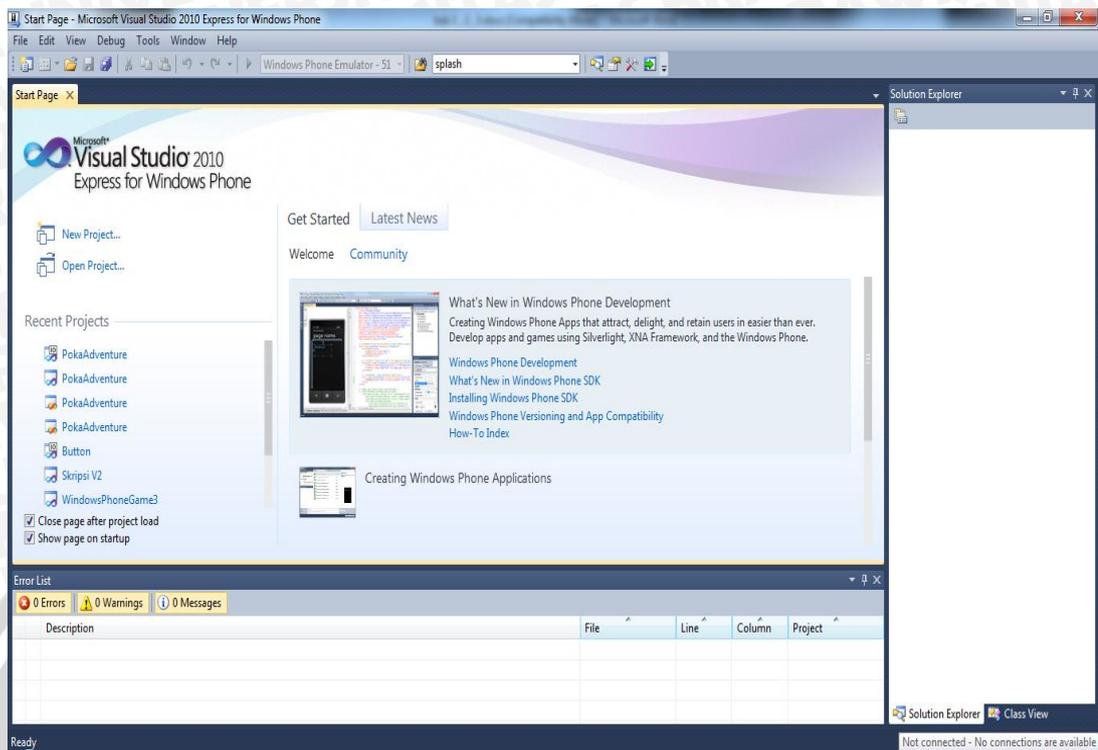
C# adalah bahasa pemrograman yang berorientasi objek yang dikembangkan oleh Microsoft Corporation sebagai bagian dari kerangka .Net Framework. Bahasa ini yang sangat fleksibel dan kuat. C# memiliki kemiripan kuat dengan C++ dan bahasa pemrograman Java, setelah memperbaiki fitur yang disediakan oleh bahasa tersebut.

Seperti halnya bahasa Java, bahasa C# telah membuang beberapa fitur berbahaya dari bahasa C. C# adalah bahasa pemrograman yang telah di-compile. Komputer tidak mengerti bahasa tersebut secara langsung, oleh karena itu sebuah program yang bernama *compiler* mengkonversi teks C# menjadi instruksi level bawah yang lebih sederhana [ROM-11].

2.4 Windows Phone : Visual Studio XNA

Windows phone merupakan sistem operasi yang tengah dikelola serta dikembangkan oleh microsoft. Sistem operasi ini bahkan didesain semirip mungkin dengan windows versi dekstop. Sedangkan XNA merupakan *library* dari microsoft yang didesain untuk memudahkan user dalam membangun sebuah permainan. Kelebihan XNA antara lain :

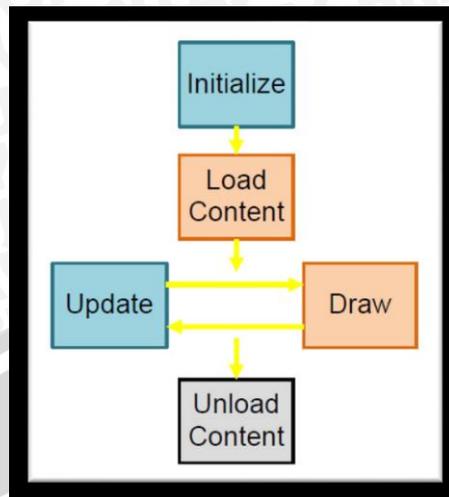
1. Gratis
2. memori yang relatif lebih besar, memiliki banyak *library* untuk membantu dalam perancangan
3. Fleksible dalam pengintegrasian antar komponen. Artinya Permainan yang dibuat dengan XNA dapat dijalankan di semua *Platform* yang mendukung XNA Framework. Selain ditujukan untuk pengembangan permainan-permainan yang berbasis Windows *desktop*, XNA *game Studio* juga dapat digunakan untuk mengembangkan permainan yang berbasis *console* Xbox 360 dan Zune. Sebuah *project* yang dibuat untuk *platform* Windows *desktop* pun dapat dikonversikan ke dalam *project* Xbox atau Zune. Sehingga hal ini sangat memudahkan pengembang yang ingin membuat permainan *multiplatform*. Tampilan awal XNA akan ditunjukkan oleh gambar 2.4.



Gambar 2.4Tampilan Awal Visual Studio XNA

XNA sendiri memiliki lima method utama yang terdiri atas [MUS-12]:

1. Initialize, method ini digunakan untuk menginisialisasi logika permainan, hal seperti data level, kondisi permainan saat ini, dan lainnya.
2. Load Content, method ini digunakan untuk digunakan untuk memload semua resource yang digunakan pada permainan. *Resource* dapat berupa *image*, *model*, *vertexbuffer*, dsb.
3. Update, method ini dijalankan setiap beberapa milisecond sekali sebelum method draw, method ini bertujuan untuk mengupdate logika permainan termasuk untuk mengambil data input dari pemain.
4. Draw, sesuai dengan namanya, method ini berfungsi untuk untuk menggambar *model/image* yang dipakai pada *permainan* (melakukan *rendering graphics* ke layar).
5. Unload Content, method ini dilakukan untuk membersihkan *resource* dari memori, biasa digunakan pada saat penggantian level dilakukan.



Gambar 2.5 Alur kerja Method XNA

Sumber : [MUS-12]

2.5 Tanaman Herbal

Bangsa Indonesia telah lamamengenal dan menggunakan tanamanberkhasiat obat sebagai salahsatu upaya dalam menanggulangi masalah kesehatan. Pengetahuan tentang tanaman berkhasiat obat berdasar pada pengalaman dan ketrampilan yang secara turun-temurun telah diwariskan dari satu generasi ke generasi berikutnya. Penggunaan bahan alam sebagai obat tradisional di Indonesia telah dilakukan oleh nenek moyang kita sejak berabad-abad yang lalu terbukti dari adanya naskah lama pada daun lontar Husodo (Jawa), Usada (Bali), Lontarak pabbura (Sulawesi Selatan), dokumen Serat Primbon Jampi, Serat Racikan Boreh Wulang nDalem danrelief candi Borobudur yang menggambarkanorang sedang meracik obat (jamu) dengan tanaman sebagai bahan bakunya [SUK-06].

Obat herbal telah diterima secara luas di hampir seluruh Negara di dunia. Menurut WHO, negara-negara di Afrika, Asia dan Amerika Latin menggunakan obat herbal sebagai pelengkap pengobatan primer yang mereka terima. Bahkan di Afrika, sebanyak 80% dari populasi menggunakan obat herbal untuk pengobatan primer (WHO, 2003). Faktor pendorong terjadinya peningkatan penggunaan obat herbal di negara maju adalah usia harapan hidup yang lebih panjang pada saat prevalensi penyakit kronik meningkat, adanya kegagalan penggunaan obat modern untuk penyakit tertentu di antaranya kanker serta semakin luas akses

informasi mengenai obat herbal di seluruh dunia [SUK-06]. WHO merekomendasi penggunaan obat tradisional termasuk herbal dalam pemeliharaan kesehatan masyarakat, pencegahan dan pengobatan penyakit, terutama untuk penyakit kronis, penyakit degeneratif dan kanker. WHO juga mendukung upaya-upaya dalam peningkatan keamanan dan khasiat dari obat tradisional (WHO, 2003). Penggunaan obat tradisional secara umum dinilai lebih aman dari pada penggunaan obat modern. Hal ini disebabkan karena obat tradisional memiliki efek samping yang relatif lebih sedikit dari pada obat modern.

Efek samping obat tradisional relatif kecil jika digunakan secara tepat, yang meliputi kebenaran bahan, ketepatan dosis, ketepatan waktu penggunaan, ketepatan cara penggunaan, ketepatan informasi, dan tanpa penyalahgunaan obat tradisional itu sendiri. Penelitian yang telah dilakukan terhadap tanaman obat sangat membantu dalam pemilihan bahan baku obat tradisional. Pengalaman empiris ditunjang dengan penelitian semakin memberikan keyakinan akan khasiat dan keamanan obat tradisional.

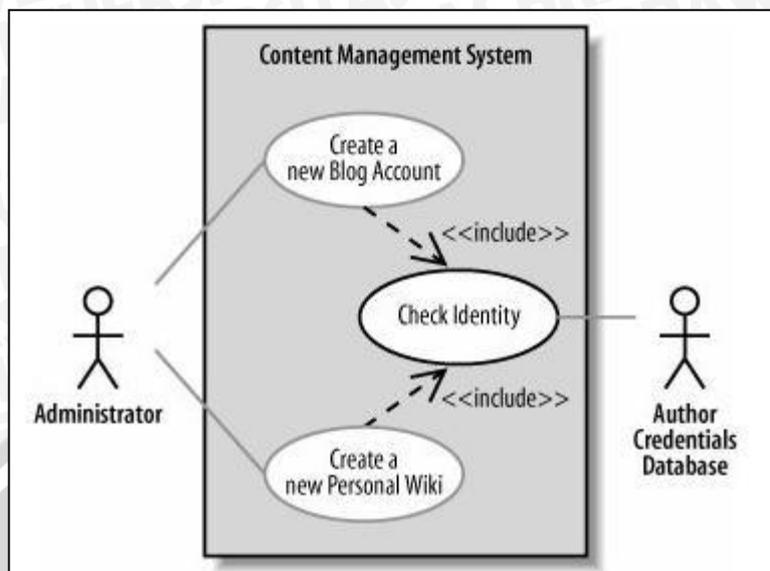
2.6 Unified Modelling Language

Modelling karakter dibutuhkan untuk membuat model serta *enviromental* yang mendukung dari *view* atau *world* (dunia) dari permainan itu sendiri.

Setelah gambar sketsa dibuat, maka model dari bentuk tersebut akan dibuat model 2d-nya dengan *tools* pendukung yaitu Adobe Flash dan Adobe Fireworks. Pemakaian *tools* digunakan sesuai kebutuhan. Penjelasan dari masing-masing *tools* tersebut telah dijelaskan diatas.

2.6.1 Diagram Use case

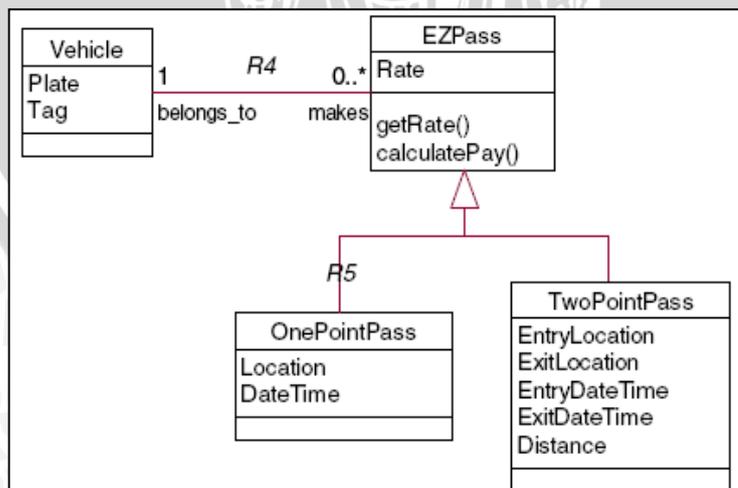
Use case adalah abstraksi dari interaksi antara sistem dan aktor. Use case bekerja dengan cara mendeskripsikan tipe interaksi antara user sebuah system dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. Use case merupakan konstruksi untuk mendeskripsikan bagaimana sistem akan terlihat di mata user. Sedangkan use case diagram memfasilitasi komunikasi diantara analis dan pengguna serta antara analis dan client. Gambar 2.6 menunjukkan contoh dari diagram *usecase*.



Gambar 2.6 Contoh Diagram Use Case

2.6.2 Diagram Class

Diagram kelas merupakan diagram paling umum dipakai di semua pemodelan berorientasi objek. Pemodelan kelas menunjukkan kelas-kelas yang ada di sistem dan hubungan antar kelas – kelas itu, atribut – atribut dan operasi – operasi di kelas – kelas. Diagram kelas menunjukkan aspek statik sistem terutama untuk mendukung kebutuhan fungsionalitas sistem. Kelas di diagram kelas dapat secara langsung diimplementasikan ke dalam bahasa pemrograman berorientasi objek yang secara langsung mendukung bentuk kelas [HAR-04:277]. Contoh diagram kelas ditunjukkan pada gambar 2.7.



Gambar 2.7 Contoh Diagram Kelas

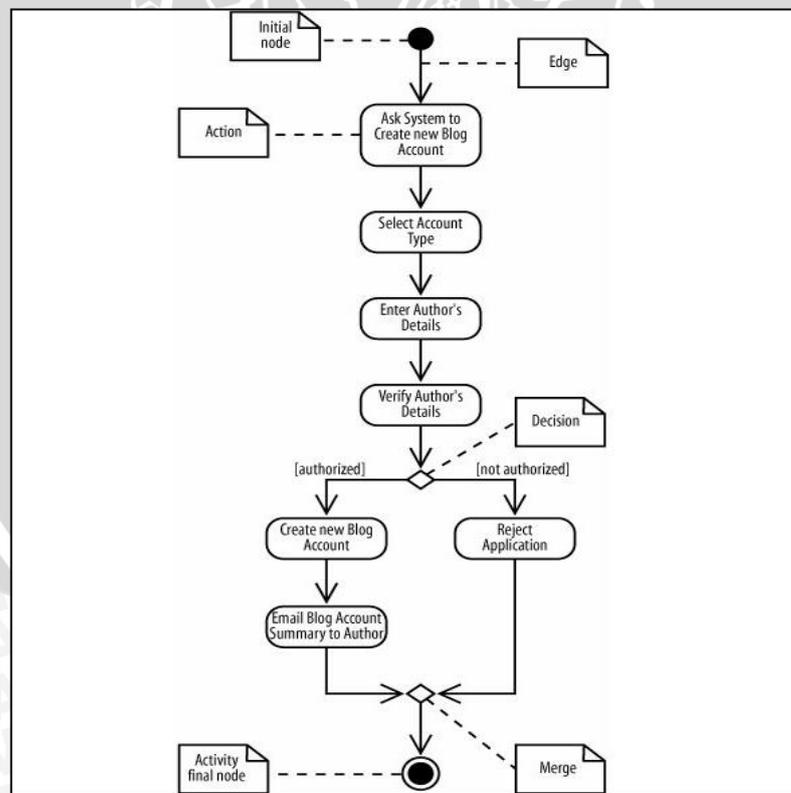
Sumber : [LIU-06:19]

Elemen – elemen esensi di diagram kelas adalah sebagai berikut :

1. Kelas
2. Antarmuka (*Interfaces*)
3. Kolaborasi
4. Hubungan (*Relationship*) seperti kebergantungan, generalisasi dan asosiasi.

2.6.3 Diagram Activity

Tahap perancangan selanjutnya adalah pembuatan diagram aktivitas. Dengan diagram aktivitas memungkinkan untuk menentukan bagaimana sistem akan mencapai tujuannya [HAM-06]. Sebagai contoh, diagram aktivitas digunakan untuk memodelkan langkah – langkah dalam pembuatan akun sebuah blog. Dalam perancangan permainan, diagram aktivitas digunakan untuk menggambarkan aliran aktivitas atau proses dari setiap use case. Gambar 2.8 menunjukkan contoh dari diagram aktivitas proses pembuatan akun blog.



Gambar 2.8 Contoh Diagram Aktivitas

Sumber : [HAM-06]

2.7 Pengujian Perangkat Lunak

Pengujian adalah kegiatan yang dilakukan untuk mengevaluasi kualitas produk, untuk meningkatkan dan mengidentifikasi cacat dan masalah. Arsitektur dari perangkat lunak berorientasi objek menghasilkan sekumpulan *layered subsystems* yang mengenkapsulasi kelas-kelas yang berkolaborasi. Setiap elemen sistem (subsistem dan *class*) melakukan fungsi yang membantu untuk mencapai kebutuhan sistem. Hal ini sangat penting untuk menguji sebuah OO *system* pada berbagai macam level yang berbeda dalam sebuah usaha untuk menemukan kesalahan-kesalahan yang mungkin terjadi dari kolaborasi kelas-kelas dan komunikasi subsistem melewati *architetural layer* [PRE-10].

2.7.1 Teknik Pengujian

Pengujian perangkat lunak memerlukan perancangan kasus uji (*test case*) agar dapat menemukan kesalahan dalam waktu singkat dan usaha minimum. Berbagai macam metode perancangan kasus uji telah berevolusi. Metode-metode ini menyediakan pendekatan sistematis untuk pengujian oleh *developer*. Terlebih lagi metode-metode ini menyediakan mekanisme yang dapat membantu memastikan kelengkapan dari pengujian dan menyediakan kemungkinan tertinggi untuk menemukan kesalahan-kesalahan dalam perangkat lunak [PRE-10].

Strategi untuk pengujian perangkat lunak merupakan aktifitas mengintegrasikan metode perancangan kasus uji ke dalam sebuah rangkaian langkah-langkah pengujian yang terencana sehingga menghasilkan perangkat lunak yang baik. Strategi menyediakan urutan langkah yang harus dilakukan sebagai bagian dari pengujian. Setiap langkah direncanakan kemudian dikerjakan dan ditentukan berapa banyak usaha, waktu dan sumber daya yang dibutuhkan. Strategi untuk melakukan pengujian perangkat lunak, dimulai dari dengan “pengujian kecil” bergerak menuju ke “pengujian besar”. Pengujian berorientasi objek akan memulai pengujian dari *unit testing*, bergerak menuju *integration testing* dan berakhir pada *validation testing* [PRE-10]. Teknik atau metode perancangan kasus uji yang digunakan adalah *black-box testing* dan *white-box testing*.

2.7.1.1 White-Box Testing

White-box testing atau *glass-box testing* merupakan sebuah metode perancangan kasus uji yang menggunakan struktur kontrol dari perancangan prosedural untuk memperoleh kasus uji [PRE-10]. Ada dua jenis pengujian yang termasuk *white-box testing* yaitu *basis path testing* dan *control structure testing*.

Pada skripsi ini menggunakan *basis path testing* yang diusulkan pertama kali oleh Tom McCabe [PRE-10]. *Basis path testing* ini memungkinkan perancang kasus uji memperoleh ukuran kompleksitas logis dari sebuah perancangan prosedural dan menggunakan pengukuran ini sebagai pedoman untuk mendefinisikan *basis set* dari jalur eksekusi (*execution path*). *Test case* yang dilakukan untuk menggunakan *basis set* tersebut dijamin untuk menggunakan setiap *statement* di dalam program paling tidak sekali selama pengujian. Sebelum metode *basis path* dapat diperkenalkan, notasi sederhana untuk representasi aliran kontrol yang disebut diagram alir (*flow graph*) harus diperkenalkan. Setiap representasi desain prosedural yang berupa *flow chart* dapat diterjemahkan ke dalam *flow graph*. *Cyclomatic complexity* adalah metrik perangkat lunak yang memberikan pengukuran kuantitatif terhadap kompleksitas logis suatu program. Bila metrik ini digunakan dalam konteks metode pengujian *basis path*, maka nilai yang terhitung untuk *cyclomatic complexity* menentukan jumlah jalur independen (*independent path*) dalam *basis set* suatu program dan memberi batas atas bagi jumlah pengujian yang harus dilakukan untuk memastikan bahwa semua *statement* telah dieksekusi sedikitnya satu kali. Jalur independen adalah jalur yang melalui program yang mengenalkan sedikitnya satu rangkaian *statement* proses baru atau suatu kondisi baru. Untuk menentukan *cyclomatic complexity* bisa dilakukan dengan beberapa cara, diantaranya [PRE-10]:

1. Jumlah *region* pada *flow graphs* sesuai dengan *cyclomatic complexity*.
2. *Cyclomatic complexity* $V(G)$, untuk grafik G adalah $V(G) = E - N + 2$, dimana E adalah jumlah *edge*, dan N adalah jumlah *node*.
3. $V(G) = P + 1$, dimana P adalah jumlah *predicate node* yaitu *node* yang merupakan kondisi (ada 2 atau lebih *edge* akan keluar *node* ini).

2.7.1.2 Black-Box Testing

Black-box testing atau *behavioral testing* berfokus pada persyaratan fungsional perangkat lunak [PRE-10]. Dengan demikian, pengujian *black-box* memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi *input* yang sepenuhnya menggunakan semua persyaratan fungsional untuk semua program. Pengujian *black-box* bukan merupakan alternatif dari teknik *white-box*, tetapi merupakan pendekatan komplementer yang kemungkinan besar mampu mengungkap kelas kesalahan daripada metode *white-box*.

Pengujian *black-box* berusaha menemukan kesalahan dalam kategori berikut :

1. Fungsi-fungsi yang tidak benar atau hilang
2. Kesalahan *interface*
3. Kesalahan dalam struktur data atau akses *database* eksternal
4. Kesalahan kinerja
5. Inisialisasi dan kesalahan terminasi.

Tidak seperti pengujian *white-box*, yang dilakukan pada saat awal proses pengujian, pengujian *black-box* cenderung diaplikasikan selama tahap akhir pengujian. Karena pengujian *black-box* memperhatikan struktur kontrol, maka perhatian berfokus pada domain informasi. Pengujian didesain untuk menjawab pertanyaan-pertanyaan berikut :

- Bagaimana validitas fungsional diuji?
- Kelas input apa yang akan membuat *test case* menjadi baik?
- Apakah sistem sangat sensitif terhadap harga input tertentu?
- Bagaimana batasan dari suatu data diisolasi?
- Kecepatan dan volume data apa yang dapat ditolerir oleh sistem?
- Apa pengaruh kombinasi tertentu dari data terhadap operasi sistem?

2.7.2 Strategi Pengujian

Strategi untuk pengujian perangkat lunak mengintegrasikan metode desain *test case* perangkat lunak ke dalam sederetan langkah yang direncanakan dengan

baik, dan hasilnya adalah konstruksi perangkat lunak yang berhasil. Sejumlah strategi pengujian perangkat lunak telah diusulkan di dalam literatur. Strategi pengujian harus mengakomodasi pengujian tingkat rendah yang diperlukan untuk membuktikan bahwa segmen kode sumber yang kecil telah diimplementasikan dengan tepat, demikian juga pengujian tingkat tinggi yang memvalidasi fungsi-fungsi sistem mayor yang berlawanan dengan kebutuhan pelanggan. Proses pengujian dimulai dengan pengujian yang berfokus pada setiap modul secara individual (*unit testing*), dilanjutkan dengan pengujian integrasi (*integration testing*) dan berakhir pada pengujian validasi (*validation testing*) [PRE-10].

2.7.2.1 Pengujian Unit

Pengujian unit berfokus pada usaha verifikasi pada inti terkecil dari desain perangkat lunak, yakni modul. Dengan menggunakan gambaran desain prosedural sebagai panduan, jalur kontrol yang penting diuji untuk mengungkap kesalahan di dalam batas modul tersebut. Kompleksitas relatif dari pengujian dan kesalahan yang diungkap dibatasi oleh ruang lingkup batasan yang dibangun untuk pengujian unit. Pengujian unit biasanya berorientasi pada *white-box*, dan langkahnya dapat dilakukan secara paralel untuk model bertingkat [PRE-10].

Interface modul diuji untuk memastikan bahwa informasi secara tepat mengalir masuk dan keluar dari inti program yang diuji. Struktur data lokal diuji untuk memastikan bahwa data yang tersimpan secara temporal dapat tetap menjaga integritasnya selama semua langkah di dalam suatu algoritma dieksekusi. Kondisi batas diuji untuk memastikan bahwa modul beroperasi dengan tepat pada batas yang ditentukan untuk membatasi pemrosesan. Semua jalur independen (jalur dasar) yang melalui struktur kontrol dipakai sedikitnya satu kali. Dan akhirnya, penanganan kesalahan uji [PRE-10].

2.7.2.2 Pengujian Integrasi

Pengujian integrasi adalah teknik sistematis untuk mengkonstruksi struktur program sambil melakukan pengujian untuk mengungkap kesalahan sehubungan dengan *interfacing*. Sasarannya adalah untuk mengambil modul yang dikenai pengujian unit dan membangun struktur program yang telah ditentukan oleh

desain. *Integration testing* berorientasi *black box* dan mempunyai dua pola pengujian yaitu integrasi *top-down* (*top-down integration*) dan integrasi *bottom-up* (*bottom-up integration*).

Integrasi *top-down* adalah pendekatan inkremental terhadap struktur program. Modul diintegrasikan dengan menggerakkan ke bawah melalui hirarki kontrol, dimulai dengan modul kontrol utama (program utama). Subordinat program terhadap modul kontrol utama digabungkan ke dalam struktur dengan cara *depth-first* atau *breadth-first*.

Proses integrasi *top-down* dilakukan dalam lima langkah :

1. Modul kontrol utama digunakan sebagai *test driver* dan *stub* digunakan untuk menggantikan semua komponen dibawahnya.
2. Pemilihan pendekatan integrasi yang diinginkan (*depth* atau *breadth first*).
3. Pengujian dikerjakan untuk setiap komponen yang diintegrasikan.
4. *Stub* digantikan dengan komponen yang sebenarnya setelah menyelesaikan serangkaian pengujian.
5. Proses akan terus dilakukan sampai membentuk sebuah perangkat lunak yang utuh.

Pengujian integrasi *bottom-up* memulai konstruksi dan pengujian dengan modul atomik (modul pada tingkat paling rendah pada struktur program). Karena modul diintegrasikan dari bawah ke atas, maka pemrosesan yang diperlukan untuk modul subordinat ke suatu tingkat yang diberikan akan selalu tersedia dan kebutuhan akan *stub* dapat dieliminasi].

Integrasi *bottom-up* dapat diimplementasi dengan langkah-langkah berikut:

1. Komponen pada level terendah digabung ke dalam sebuah sub fungsi (*cluster*).
2. Driver akan dibuat untuk menguji setiap *cluster*.
3. Driver akan diganti dengan modul sesungguhnya setelah sub fungsi teruji.
4. Proses akan terus dilakukan sampai membentuk sebuah perangkat lunak yang utuh[PRE-10].

2.7.2.3 Pengujian Validasi

Pada kulminasi pengujian terintegrasi, perangkat lunak secara lengkap dirakit sebagai suatu paket; kesalahan *interfacing* telah diungkap dan dikoreksi, dan seri akhir dari pengujian perangkat lunak, yaitu pengujian validasi dapat dimulai. Validasi dapat ditentukan dengan berbagai cara, tetapi definisi yang sederhana adalah bahwa validasi berhasil bila perangkat lunak berfungsi dengan cara yang dapat diharapkan secara bertanggung jawab oleh pelanggan. Validasi perangkat lunak dicapai melalui sederetan pengujian *black-box* yang memperlihatkan konformitas dengan persyaratan. Rencana pengujian menguraikan kelas-kelas pengujian yang akan dilakukan, dan prosedur pengujian menentukan *test case* spesifik yang akan digunakan untuk mengungkap kesalahan dalam konformitas dengan persyaratan. Baik rencana dan prosedur didesain untuk memastikan apakah semua persyaratan fungsional dipenuhi; semua persyaratan kinerja dicapai; dokumentasi benar dan direkayasa oleh manusia; dan persyaratan lainnya dipenuhi (transportabilitas, kompatibilitas, pembetulan kesalahan, maintainabilitas) [PRE-10].

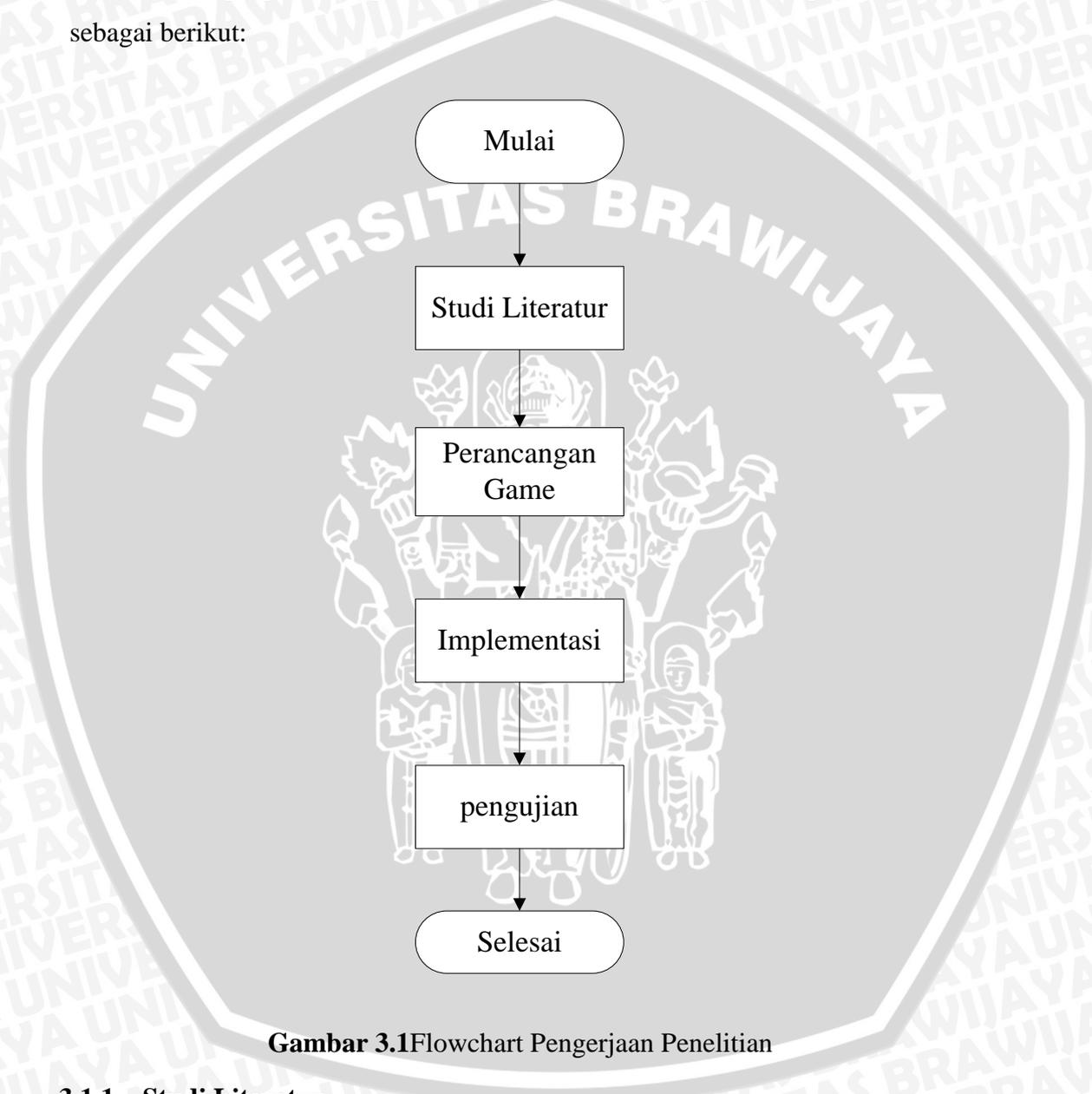
2.7.2.4 Pengujian Performa

Setelah semua langkah pengujian perangkat lunak secara terstruktur dilakukan, maka perlu dilakukan pengujian sistem di lingkungan dimana dia bekerja untuk mengetahui performa dari perangkat lunak tersebut. Pengujian sistem dirancang untuk menguji kinerja *run-time* dari perangkat lunak dalam konteks sistem terintegrasi. Pengujian performa melibatkan *monitoring* pemanfaatan sumber daya dari perangkat lunak yang diuji seperti perangkat lunak pendukung dan perangkat keras. Pengujian performa dilakukan secara spesifik sesuai dengan tipe perangkat lunak yang diuji. Pengujian performa bertujuan untuk mengungkap situasi yang menyebabkan degradasi dan kemungkinan kegagalan sistem [PRE-10].

BAB III METODE PENELITIAN DAN PERANCANGAN

3.1 Metode Penelitian

Dalam proposal skripsi ini metode penelitian yang akan digunakan adalah sebagai berikut:



Gambar 3.1Flowchart Pengerjaan Penelitian

3.1.1 Studi Literatur

Studi literatur menjelaskan dasar teori yang digunakan untuk menunjang penulisan skripsi. Teori-teori pendukung tersebut meliputi:

1. Pengertian permainan (*game*) sebagai aktivitas yang dimainkan untuk olahraga atau hiburan menurut aturan. Dalam teori permainan juga mencakup tentang klasifikasi permainan.
2. Tanaman herbal
3. Asset Maker yang digunakan. Seperti Adobe Flash dan Adobe Fireworks.
4. Library yang digunakan. Microsoft visual studio XNA

Penggunaan teknik *white-box testing* dan *black-box testing* untuk pengujian perangkat lunak. Strategi yang digunakan dalam pengujian adalah *unittesting*, *integration testing*, *validation testing* dan *performance testing*.

3.1.2 Perancangan

Perancangan dalam sebuah permainan merupakan proses merancang atau mendesain sebuah permainan, yang isinya adalah kebutuhan serta prosedur teknis permainan yang akan dibuat. Tujuan dari desain permainan adalah untuk memberikan gambaran secara umum kepada pengguna tentang permainan yang akan dibuat. Perancangan permainan dibagi menjadi 2 jenis yaitu *game concept design* serta *technical design*.

Untuk merancang sebuah permainan, langkah-langkah yang dilakukan terlebih dahulu adalah mencari ide permainan dengan *brainstorming*. Setelah menemukan ide *game play* yang sesuai, langkah berikutnya adalah menentukan *rules*, *goal*, serta genre seperti apa yang diinginkan. Setelah *rules*, *goal* serta genre langkah berikutnya yang harus dilakukan adalah membuat dokumentasi. Langkah pertama dalam membuat dokumentasi permainan adalah dengan membuat kerangka dalam bentuk *one sheet document*. Langkah berikutnya adalah membuat *ten pager* agar pembaca lebih mudah memahami hasil akhir dari permainan. Setelah membuat *one sheet document* dan *ten pager* langkah terakhir dalam perancangan *game concept design* adalah membuat *game Design Document*

3.1.3 Implementasi

Implementasi pembuatan permainan dilakukan dengan mengacu kepada perancangan permainan serta pembuatan aset dan desain. Implementasi yang dilakukan di sini adalah implementasi musik, implementasi grafis, implementasi

art, implementasi prosedur program. Adapun Tabel 3.1 menunjukkan implementasi dalam pembuatan permainan ini adalah sebagai berikut:

Tabel 3.1 Implementasi Pembuatan Permainan

Jenis Implementasi	Nama Program
Desain Karakter	Adobe Flash CS6
Bahasa Pemrograman	C#
Pembuatan Materi 2D	Adobe Fireworks CS6 Adobe Photoshop CS6

3.1.4 Evaluasi dan analisa (pengujian)

Pengujian sistem merupakan hal terpenting yang bertujuan untuk menemukan kesalahan – kesalahan atau kekurangan – kekurangan pada perangkat lunak yang diuji. Metode pengujian yang akan digunakan adalah *white-box testing*, *black-box testing* dan pengujian *frame per second* (FPS). Pengujian bermaksud untuk mengetahui perangkat lunak yang dibuat sudah memenuhi kriteria yang sesuai dengan tujuan perancangan perangkat lunak tersebut.

3.2 Perancangan

Tahap perancangan permainan terdiri dari dua bagian yaitu *gameconcept design* dan *technical design*.

3.2.1 GameConcept Design

Tahap *gameconceptdesign* dilakukan dengan pembuatan *gamedesign document*. Pada tahap pembuatan *game design document*, langkah yang pertama kali dilakukan adalah mencari ide. Pencarian ide dapat dilakukan dengan menuliskan ide yang ada atau berdiskusi dengan kelompok orang yang memiliki disiplin ilmu yang berbeda. Kemudian dari ide-ide yang didapatkan akan ditulis pada sebuah catatan yang disebut dengan *brainstormingnote* [ROG-10:32].

Pada bab ini dijelaskan pembuatan *game concept design document* yang merupakan detail-detail dalam permainan. Tujuan dibuatnya dokumen ini adalah untuk mengkomunikasikan kepada tim untuk meminimalisir kesalahan pada saat implementasi karena kurang mengerti tentang *permainan* yang dibuat [ROG-10:75]. *Game concept design document* mencantumkan segala rincian dalam

desain pembuatan permainan. Rincian tersebut meliputi Konsep utama permainan, Kebutuhan Teknologi, Tujuan permainan, Cerita, Deskripsi *game play*, Kontrol Permainan, Dunia Dalam Permainan, Sudut Kamera, Layar Awal, *gameflow*, Antarmuka, Karakter Pemain, Musuh, Koleksi, Nyawa, Level, Musik dan Suara.

3.2.1.1 Konsep Utama Permainan

Konsep utama Permainan ditunjukkan oleh tabel 3.2 berikut :

Tabel 3.2 Konsep Utama Permainan

No.	Elemen	Keterangan
1.	Judul	Poka si tabib cilik
2.	Platform	<i>Windows Phone</i>
3.	Target Device	XNA
4.	Target Usia	7+
5.	Genre Permainan	Education, Adventure
6.	Product Sejenis	Criminal Case
7.	USP (Unique Selling Point)	<ul style="list-style-type: none"> • Media edukasi memperkenalkan tanaman herbal • Gabungan dua genre <i>permainan</i> dalam satu aplikasi • Art 2D yang user friendly

Sumber: Metodologi dan perancangan

3.2.1.2 Kebutuhan Teknologi

Daftar kebutuhan teknologi untuk membangun permainan ditunjukkan pada tabel 3.3.

Tabel 3.3 Daftar Kebutuhan Teknologi

No.	Kebutuhan	Kegunaan
1.	Aplikasi Pengolah Grafis	Sebagai media untuk menggambar desain 2D

2.	Aplikasi Visual Studio C#	Untuk membangun Aplikasi
3.	Windows phone	Sebagai alat uji perangkat lunak

Sumber: Metodologi dan perancangan

3.2.1.3 Tujuan Permainan

Tujuan utama dari permainan ini adalah menyelesaikan misi permainan untuk membuat sebuah ramuan sesuai dengan penyakit yang terdapat dalam jalan cerita. Bagaimana pemain mencari tanaman – tanaman herbal yang kelak akan diberikan informasi bagaimana cara meramunya sehingga menjadi obat tradisional.

3.2.1.4 Cerita

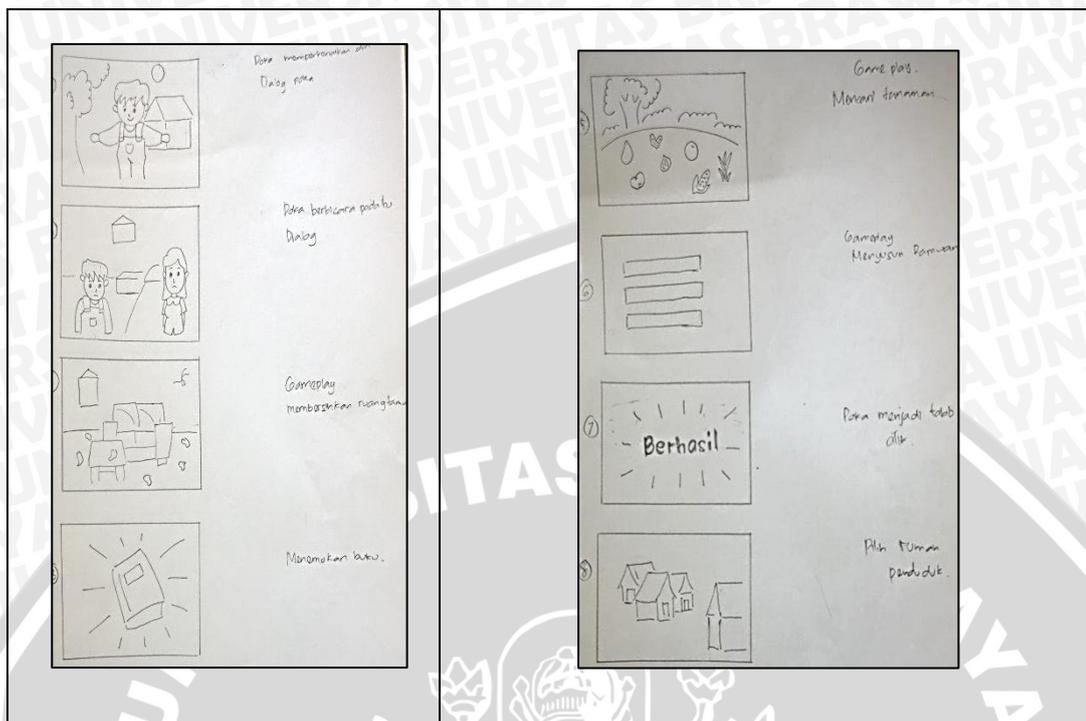
pemain (poka) bersifat baik hati dan penyayang keluarga. Diceritakan bahwa dia merupakan seorang anak yang hidup hanya berdua saja dengan ibunya. Sebelumnya ayahnya merupakan tabib yang terkenal. Namun karena ayahnya meninggal ia dan ibunya harus hidup sederhana. Namun ia tetap menjadi anak yang periang sehingga masyarakat menyayanginya. Hingga suatu ketika ibunya jatuh sakit. Poka yang sedih karena tidak memiliki uang untuk membawa ibunya berobat pun menjadi murung.

Namun karena Poka anak yang berjiwa besar ia tidak mau lama-lama murung. Maka dia pun melakukan apa saja yang ia bisa. Ia menggantikan pekerjaan ibunya. Salah satunya membersihkan rumah.

Pada saat membersihkan rumah maka permainan ini baru dimulai. Dan ketika poka berhasil membersihkan rumah ia akan menemukan buku peninggalan ayahnya. Yaitu buku yang berisi ramuan-ramuan obat yang bisa ia gunakan untuk mengobati ibunya.

Stage kedua pun dimulai, *setting*-nya berada di halaman. Poka sedang mencari tanaman apa saja yang ia perlukan untuk membuat ramuan itu. Setelah menemukan semua ramuan maka poka akan menyusun *puzzle* untuk mendapatkan cara mengolah tanaman dengan benar.

Di akhir cerita ibu poka menjadi sehat kembali. Setelah itu poka sering diminta warga untuk membantu mengobati bila ada warga yang sakit. Rancangan cerita akan ditunjukkan oleh Gambar 3.2.



Gambar 3.2 Storyboard

3.2.1.5 Kontrol Permainan

Kontrol Permainan menunjukkan bagaimana pemain mengendalikan permainan. Namun tentu saja control permainan ini sudah pasti menggunakan *Touch screen*. *Touch screen* merupakan kendali utama dalam permainan ini. Fungsi *touch* dapat digunakan untuk memilih menu pada menu utama, melanjutkan dialog percakapan serta memilih objek.

3.2.1.6 Permainan Camera

Permainan ini menggunakan tipe *static camera* atau *locked camera*. *Static camera* tidak mengubah posisi dan tetap diam dalam satu screen lokasi. Keuntungan dari *static camera* ini adalah pemain bisa lebih terfokus dengan semua elemen atau objek yang ada dalam satu screen, sehingga bisa memaksimalkan unsur art dari sebuah permainan tanpa harus bersusah payah menciptakan suatu suasana baru karena scene hanya dilihat dari sudut pandang yang sama.

3.2.1.7 Start Screen

Layar awal merupakan layar pembuka permainan yang berisi logo institusi yang menaungi pembuatan permainan meliputi Logo Universitas Brawijaya yang ditunjukkan pada gambar 3.3. Logo PTIIK yang ditunjukkan pada gambar 3.4 dan Logo *Game Lab* yang ditunjukkan pada gambar 3.5.



Gambar 3.3 Logo Universitas Brawijaya
Sumber: Metodologi dan Perancangan



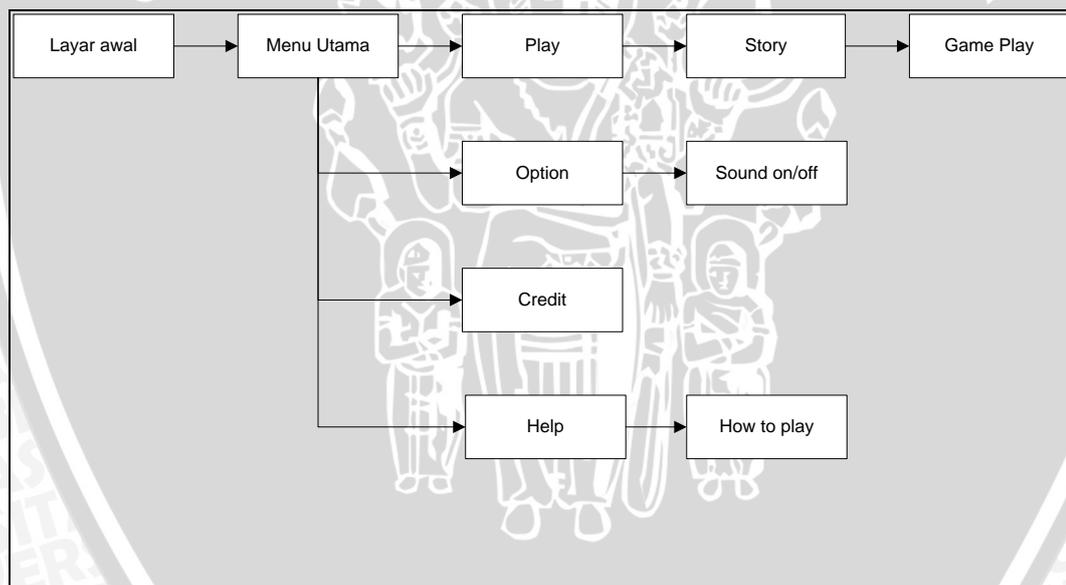
Gambar 3.4 Logo PTIIK
Sumber: Metodologi dan Perancangan



Gambar 3.5 Logo Game Lab
Sumber: Metodologi dan Perancangan

3.2.1.8 Game Flow

Game flow merupakan gambaran aliran jalan permainan yang dapat dilihat oleh pemain. Terdapat beberapa layar dalam permainan ini meliputi Layar awal, cerita, menu utama, dan lain-lain.



Gambar 3.6 Game Flow

3.2.1.9 Interface

Pada permainan POKA si tabib cilik dirancang empat *screen* yang terdiri dari menu utama, *help screen*, *credit screen*, *option permainan screen*. Rancangan ditunjukkan pada tabel 3.4.

Tabel 3.4 Rancangan Antarmuka Menu Utama

Nama Layar	Main Menu
Sketsa	
Keterangan	<p>Main menu berisi tombol <i>play</i>, <i>option</i>, <i>credit</i>, <i>help</i> dan <i>exit</i>.</p> <ol style="list-style-type: none"> 1. <i>play</i> : Mulai permainan. 2. <i>Option</i>: Pengaturan suara <i>on/off</i>. 3. <i>credit</i> : berisi informasi <i>permainan</i> serta keterangan tentang pembuat <i>permainan</i>. 4. <i>Help</i>: Informasi cara bermain.

3.2.1.10 Karakter Pemain

Disini terdapat dua karakter utama yaitu karakter poka yang ditunjukkan tabel 3.5 dan karakter ibu yang ditunjukkan tabel 3.6. serta terdapat empat karakter pembantu. Dikatakan karakter pembantu karena mereka sebagai parameter yang menunjukkan level dalam cerita. Empat karakter itu merupakan Pak Wuluh, coco, Oryza dan Rio. Masing masing karakter akan ditunjukkan gambar 3.7, 3.8, 3.9 dan 3.10.

Tabel 3.5 Karakter Poka

Sketsa	Profil	
	Nama	Poka
	Kelamin	Laki-laki
	Usia	8 Tahun
	Kepribadian	Ceria

	Pakaian	<i>Casual</i>
	Kebangsaan	Indonesia
	Suku	Jawa
	Posisi awal	Menghadap kekanan
	Mimik wajah	Senang, Sedih, Bingung, berpikir
	Nyawa	Pemain akan diberi 3 kali kesempatan untuk mencoba. Apabila selama tiga kali pemain gagal menyelesaikan misi maka keluar dari permainan.

Tabel 3.6 Karakter ibu

Sketsa	Profil	
	Nama	Ibu
	Kelamin	Perempuan
	Usia	40 Tahun
	Kepribadian	Penyayang
	Pakaian	<i>Casual</i>
	Kebangsaan	Indonesia
	Suku	Jawa
	Mimik wajah	Senang, Sedih

Tabel 3.7 Karakter Pak wuluh

Sketsa	Profil	
	Nama	Pak wuluh
	Kelamin	Laki-laki
	Usia	50 Tahun
	Kepribadian	Bijaksana
	Pakaian	<i>Casual</i>
	Kebangsaan	Indonesia
	Suku	Jawa

Tabel 3.8 Karakter Coco

Sketsa	Profil	
	Nama	Coco
	Kelamin	Laki-laki
	Usia	10 Tahun
	Kepribadian	Setia kawan
	Pakaian	<i>Casual</i>
	Kebangsaan	Indonesia
	Suku	Jawa

Tabel 3.9 Karakter Oryza

Sketsa	Profil	
	Nama	Oryza
	Kelamin	Perempuan
	Usia	20 Tahun
	Kepribadian	Baik hati
	Pakaian	<i>Casual</i>
	Kebangsaan	Indonesia
	Suku	Jawa

Tabel 3.10Karakter Rica

Sketsa		Profil	
	Nama	Rica	
	Kelamin	Perempuan	
	Usia	13 Tahun	
	Kepribadian	Senang bermain	
	Pakaian	<i>Casual</i>	
	Kebangsaan	Indonesia	
	Suku	Jawa	

3.2.1.11 Musuh

Dalam permainan ini karakter utama tidak memiliki musuh.

3.2.1.12 Level

Level disini ditunjukkan dengan gambar rumah-rumah yang ada. Setelah poka menyelesaikan level pertama yaitu ketika ibu sudah sembuh maka poka akan memilih gambar rumah rumah itu untuk permainan selanjutnya.

Tabel 3.11Deskripsi Level

Level name	Deskripsi
Rumah 1	Disini poka bertemu dengan temannya. Coco. Lalu poka menolong coco yang sedang demam dan sakit perut.
Rumah 2	Disini poka bertemu kak oryza. Kak oryza sering membantunya. Sehingga poka juga ingin membantu kak oryza yang sedang sakit kencing manis.
Rumah 3	Disini poka bertemu dengan pak wuluh. Pak wuluh menderita penyakit malaria.
Rumah 4	Disini poka juga bertemu kak rica. Kak Rica juga merupakan teman poka. Disini poka membantu kak rica yang sedang sakit kolesterol.

3.2.2 Technical Design

Technical design menjelaskan tentang kebutuhan teknis dari *game*. *Technical design* dari perancangan *game* ini meliputi pembuatan *use case diagram*, *class diagram* dan *activity diagram*.

3.2.2.1 Use Case Diagram

Use case diagram dalam perancangan *game* digunakan untuk memodelkan fungsionalitas dari permainan. Sebelum membuat *usecase diagram* kita harus mengidentifikasi aktor serta kebutuhan terlebih dahulu.

1. Identifikasi aktor

Tahap ini adalah tahap untuk melakukan identifikasi terhadap aktor yang akan berinteraksi dengan *permainan* Poka si tabib cilik. Tabel 3.12 memperlihatkan aktor yang terlibat beserta penjelasannya yang merupakan hasil dari proses identifikasi aktor.

Tabel 3.12 Identifikasi Aktor

Aktor	Deskripsi
Pemain	Pemain adalah pengguna yang dapat memainkan permainan, mengontrol <i>permainan</i> , mendapatkan akses untuk melihat info dan bantuan.

2. Daftar kebutuhan

Daftar kebutuhan adalah daftar yang menjabarkan kebutuhan fungsional yang ada di dalam permainan Pengenalan Tanaman Herbal, ditunjukkan pada Tabel 3.13.

Tabel 3.13 Deskripsi Daftar Kebutuhan

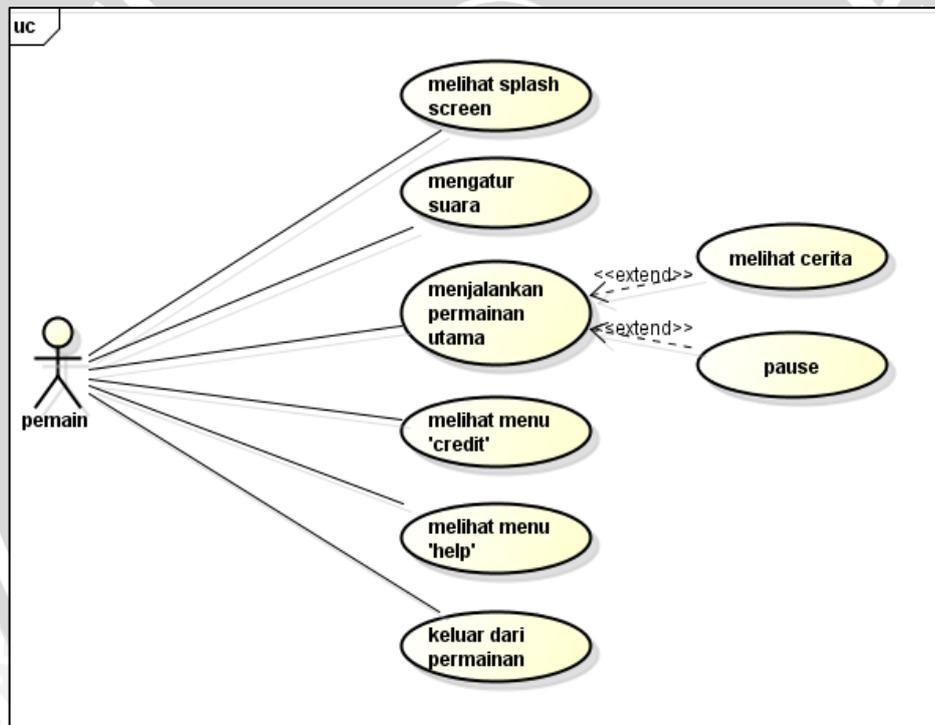
ID	Kebutuhan	Aktor	Nama Use Case
F01	Permainan harus dapat menampilkan <i>splash screen</i> berupa logo UB, PTIIK dan PermainanLab sebelum masuk ke dalam	Pemain	Melihat <i>Splash Screen</i>

	permainan.		
F02	Perangkat lunak harus menyediakan antarmuka untuk menampilkan berbagai menu utama permainan	Pemain	Melihat menu utama.
F03	Permainan harus dapat menampilkan cerita. Dimulai dari pengenalan karakter utama kemudian masuk ke cerita	Pemain	Melihat cerita
F04	Permainan harus menyediakan fasilitas untuk melihat bantuan, yaitu halaman yang menunjukkan cara kontrol permainan.	Pemain	Melihat Bantuan
F05	Permainan harus menyediakan fasilitas untuk melihat <i>credit</i> . Sehingga pemain dapat melihat siapa saja yang terlibat dalam pembuatan permainan, <i>referensi</i> pembuatan permainan dan kontak pembuat permainan.	Pemain	Melihat credit menu
F06	Permainan harus mampu memulai permainan ketika pemain menyentuh tombol 'play'	Pemain	Menjalankan permainan utama
F07	Perangkat lunak harus menyediakan proses untuk menampilkan waktu yang tersisa serta score yang diperoleh serta menampilkan item-item yang dicari	Pemain	Melihat GUI
F08	Permainan harus mampu menyusun permainan <i>puzzle</i> ketika pemain berhasil menyelesaikan item yang dicari.	Pemain	Menjalankan serta Menyusun <i>puzzle</i>
F09	Perangkat lunak harus menyediakan komponen untuk menampilkan informasi skor pemain.	Pemain	Melihat skor pemain.
F10	Permainan harus mampu menyediakan	Pemain	Melihat option

	fasilitas pengaturan suara, apakah pemain ingin bermain menggunakan music dan sound effect atau tidak.		menu
F11	Permainan harus dapat melihat percakapan antar karakter.	Pemain	Melihat dialog

3. Use case Diagram

Use case diagram dalam perancangan game digunakan untuk memodelkan fungsionalitas dari permainan. Diagram use case ini melibatkan user sebagai aktor dan beberapa use case.

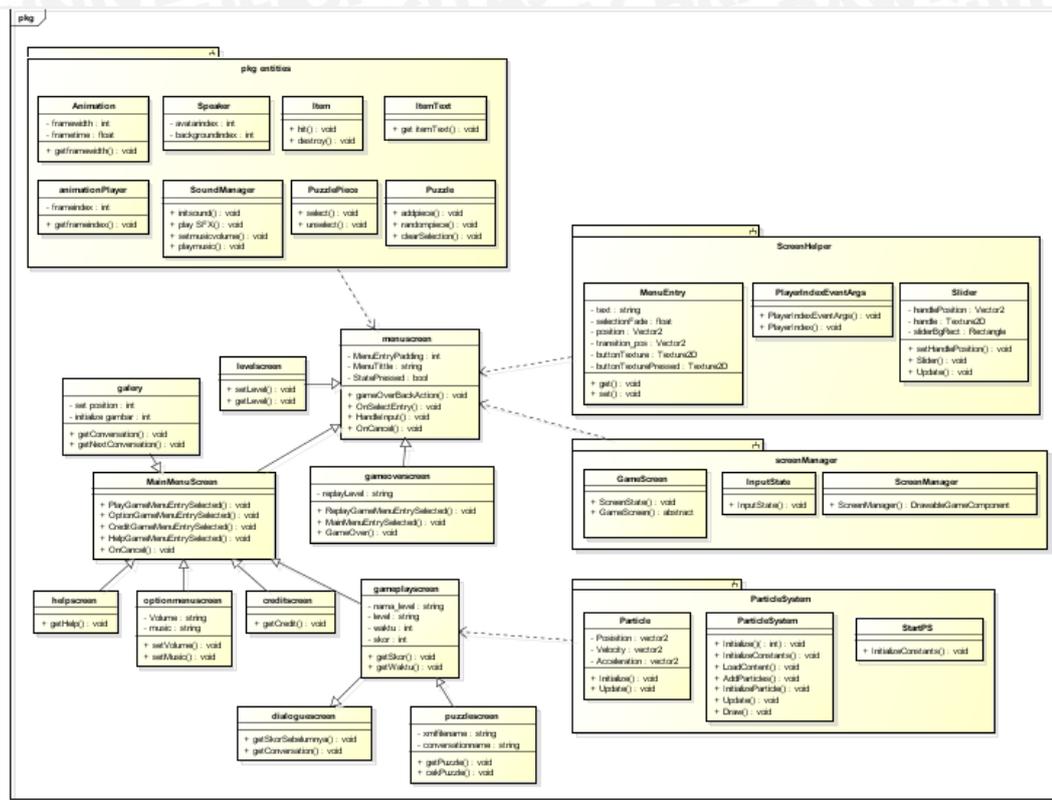


Gambar 3.7 Use Case Diagram Game poka si tabib cilik

3.2.2.2 Class Diagram

Class Diagram memberikan gambaran pemodelan elemen-elemen class yang membentuk sebuah perangkat lunak. Class didapatkan dengan menganalisis secara detail terhadap use case yang dimodelkan. Deskripsi dari class diagram permainan ini ditunjukkan pada Gambar 3.8 dan Tabel 3.14.





Gambar 3.8 Class Diagram

Tabel 3.14 Deskripsi class Diagram

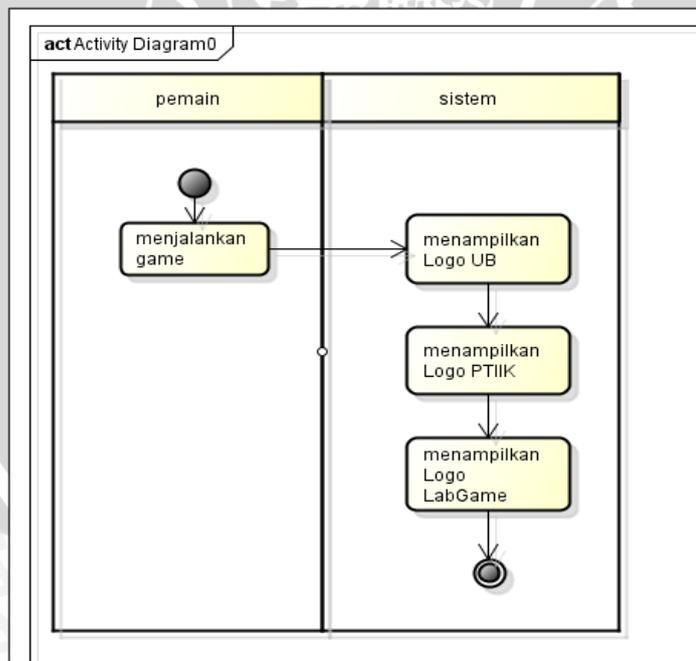
No	Class	Keterangan
	Animation	Class untuk menyusun animasi
	AnimationPlayer	Class untuk menjalankan satu class animasi
	Puzzle	class untuk menyusun puzzle piece dan merandomnya.
	PuzzlePiece	Class untuk meninisialisasi tiap 'piece'.
	Item	Class untuk meninisialisasi item yang ada.
	itemmTeks	Class untuk menginisialisasi teks yang digunakan untuk item.
	Speaker	Class yang digunakan untuk menginisialisasi siapa yang melakukan dialog.
	soundManager	Class untuk menginisialisasi musik yang digunakan.
	Menuscreen	Class yang berfungsi menghandle screen

	menu.
Levelscreen	Class yang berfungsi untuk menginisialisasi level yang ada.
Gameoverscreen	Class yang mengatur screen ketika kondisi permainan telah berakhir.
Puzzlescreen	Class yang berfungsi untuk mengatur screen puzzle.
CreditsScreen	Class yang berfungsi untuk mengatur screen credits.
DialogScreen	Class yang berfungsi untuk mengatur dialog permainan, seperti pause, reload dan reset. Yang mana nantinya dapat dipanggil oleh class – class screen.
GamePlayScreen	Class yang berfungsi untuk memanggil screen di tiap levelnya(memanggil . xml).
MainMenuScreen	Class yang berfungsi untuk mengatur screen main menu.
Gallery	Class yang mengatur tampilan dari galeri game.
HelpScreen	Class yang berfungsi untuk mengatur screen help.
OptionsMenuScreen	Class yang berfungsi untuk mengatur screen option. Mengatur kerasnya musik dan suara.
screenManager	Class yang berfungsi untuk membantu mengatur dari screen yang ada. Serta menginisialisasi inputan
Gamescreen	
inputState	
Particle	Class untuk menginisialisasi animasi bintang-bintang pada saat item ditekan dengan benar serta menuju kearah itemteks.
Particlesystem	
StarPs	
PlayerIndexArgs	Class untuk menginisialisasi index player.
Slider	Class untuk mengatur slider yang ada di

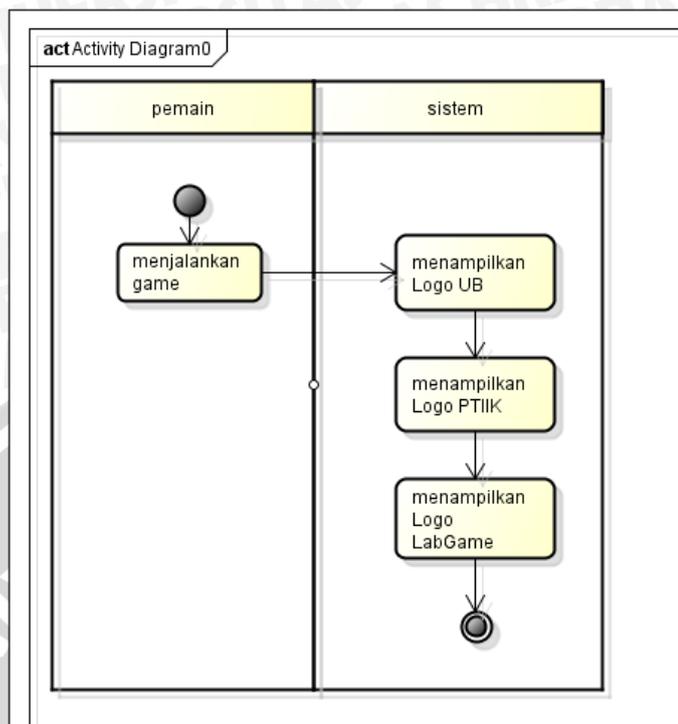
	option screen. Agar dapat mengatur volume music dan sound.
MenuEntry	Class untuk menginisialisasi menu. Menambahkan menu serta inialisasi apa yang akan dilakukan ketika menu ditekan.

3.2.2.3 Activity diagram

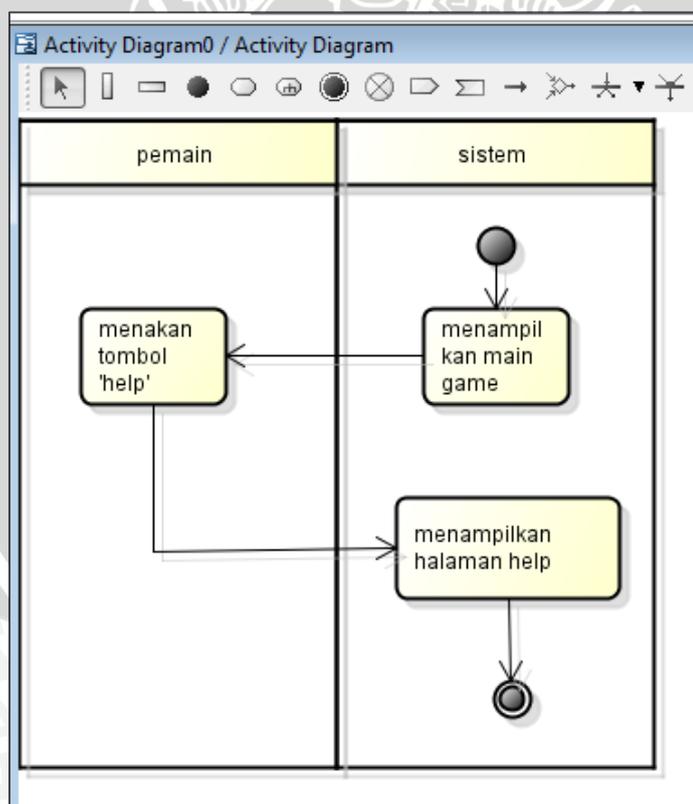
Pembuatan *activity diagram* ini bertujuan untuk menggambarkan urutan aktivitas dari proses pada setiap *use case* yang ada. Berikut merupakan gambar dari masing-masing *activity diagram*. Gambar 3.9 menunjukkan *activity diagram* ketika menjalankan *splashscreen*, gambar 3.10 menunjukkan *activity diagram* melihat *story*, gambar 3.11 menunjukkan *activity diagram* ketika melihat *help screen*, gambar 3.12 menunjukkan *activity diagram* melihat *option screen* (dimana *option screen* bertugas mengatur *volume sound*), gambar 3.13 menunjukkan *activity diagram* ketika memulai permainan dan gambar 3.14 menunjukkan *activity diagram* melihat menu *credit*.



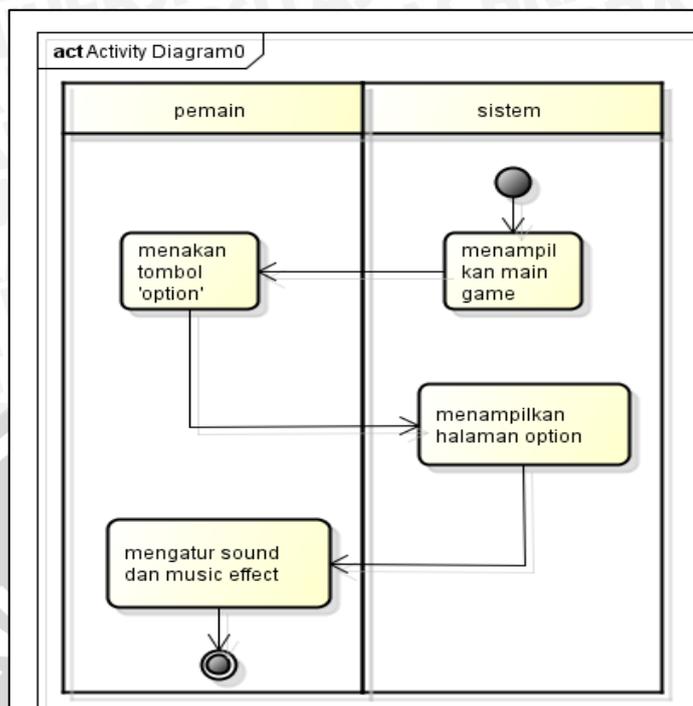
Gambar 3.9 Activity Diagram splashscreen



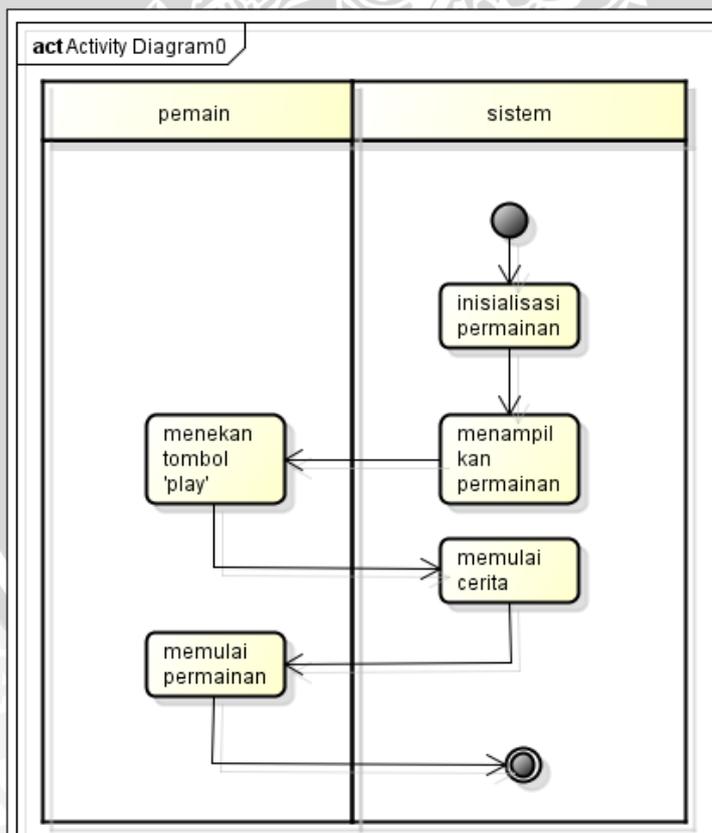
Gambar 3.10 Activity Diagram melihat story



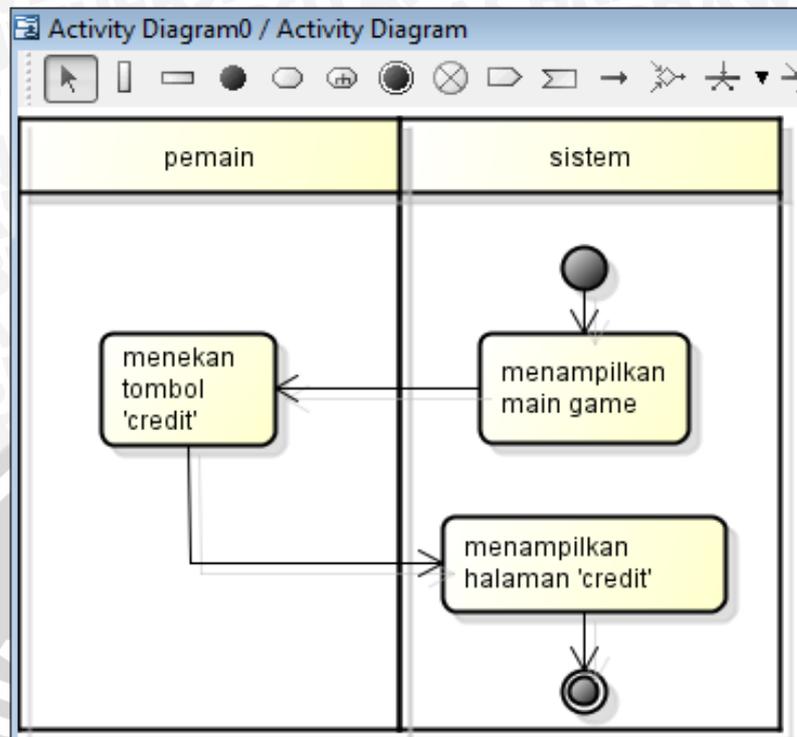
Gambar 3.11 Activity diagram melihat help screen



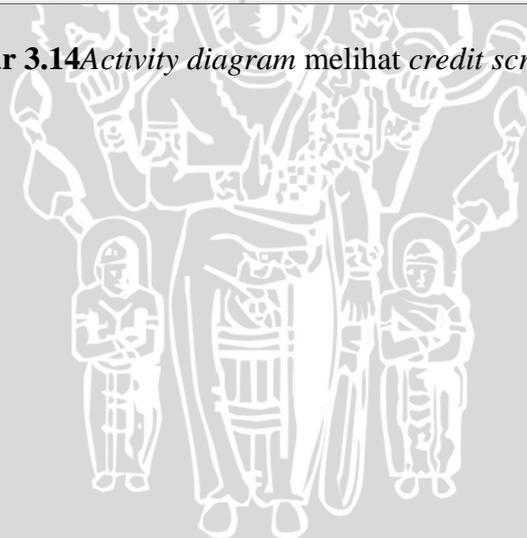
Gambar 3.12 Activity diagram melihat option screen



Gambar 3.13 Activity diagram memulai permainan



Gambar 3.14 Activity diagram melihat credit screen



BAB IV IMPLEMENTASI

Bab ini akan membahas implementasi dari pembuatan multimedia pengenalan tanaman herbal sesuai dengan dokumen desain permainan.

4.1 Spesifikasi Sistem

Perangkat lunak ini dikembangkan dalam lingkungan implementasi yang terdiri dari perangkat keras dan perangkat lunak.

4.1.1 Spesifikasi Lingkungan Perangkat Keras

Spesifikasi perangkat keras yang dipakai dalam proses pengembangan dijelaskan pada Tabel 4.1.

Tabel 4.1 Spesifikasi Lingkungan perangkat keras komputer

PERSONAL COMPUTER (PC)	
<i>Processor</i>	Intel(R) Core(TM) i3 CPU M330 @ 2.13Ghz (4CPUs)
<i>Memory (RAM)</i>	2GB
<i>Harddisk</i>	320 GB HDD
<i>Motherboard</i>	HP Pavilion dv3 series
<i>Graphic Card</i>	ATI radeon

4.1.2 Spesifikasi Lingkungan Perangkat Lunak

Spesifikasi perangkat lunak yang dipakai dalam proses pengembangan perangkat lunak dijelaskan pada Tabel 4.2.

Tabel 4.2Spesifikasi Lingkungan Perangkat Lunak Komputer

SOFTWARE	
<i>Operating System</i>	Microsoft Windows 7 Ultimate 64-bit (6.1 build 7600)
<i>DirectX Version</i>	DirextX 11
<i>Programming Language</i>	C#
<i>Graphics Editor</i>	Adobe Fireworks CS6 Adobe Flash CS6
<i>Integrated Development Environment</i>	Microsoft visual studio

4.1.3 Batasan-Batasan Dalam Implementasi

Beberapa batasan dalam mengimplementasikan perangkat lunak ini adalah sebagai berikut:

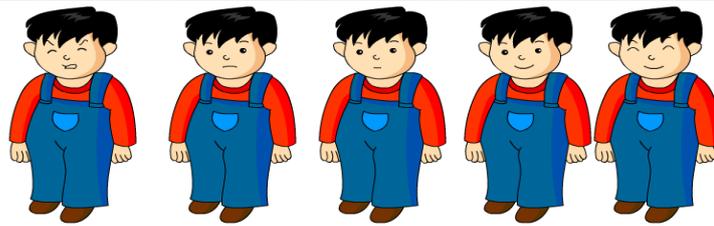
1. Aplikasi multimedia pembelajaran bangun ruang ini dirancang untuk platform windows phone versi 7. 0. menggunakan platform XNA.
2. Permainan merupakan single player.
3. Fitur yang dibahas pada implementasi aplikasi ini adalah:
 - a. Multimedia Pengenalan tanaman.
 - b. Konsep jalan cerita yang disesuaikan dengan permainan.
4. Permainan menggunakan bahasa Indonesia
5. Permainan yang dibangun merupakan permainan 2D.

4.2 Implementasi Antarmuka serta BGM

Implementasi *art* terdiri dari implementasi karakter dan implementasi lingkungan permainan.

4.2.1 Implementasi Karakter Poka

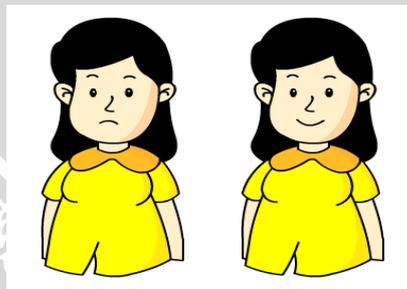
Karakter utama bernama poka, didesain bertubuh sedang dengan mengenakan baju merah dan biru agar merepresentasikan anak yang periang. Desain mimik wajah dikategorikan menjadi 5 bagian besar yaitu ekspresi bahagia, standar serta ketika sedang bersedih serta bingung dan perpikir.



Gambar 4.1 Poka

4.2.2 Implementasi Karakter Ibu

Ibu merupakan ibu dari poka. Digambarkan dengan dua ekspresi yaitu sedih dan senang. Namun ibu juga digambarkan dengan aura yang menggambarkan sosok penyayang.



Gambar 4.2 Ibu

4.2.3 Implementasi Karakter Pak Wuluh

Pak Wuluh merupakan penghuni rumah nomor satu. Pak wuluh digambarkan mempunyai sifat kebabakan serta bijaksana. Mempunyai wibawa. Dalam cerita pak wuluh digambarkan menderita penyakit malaria.



Gambar 4.3 Pak Wuluh

4.2.4 Implementasi Karakter Coco

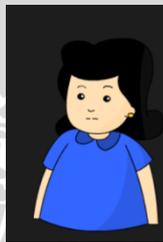
Coco merupakan teman poka. Coco anak yang baik. Diceritakan bahwa dia keracunan makanan. Namun orang tuanya sedang pergi keluar kota. Sehingga poka memberikan pertolongan sementara sebelum dibawa ke rumah sakit.



Gambar 4.4Coco

4.2.5 Implementasi Karakter Oryza

Oryza merupakan tetangga poka. Dia sangat baik dan ramah. Dia rajin menolong poka. Oryza diceritakan sedang sakit demam berdarah. Sehingga poka bertugas membantu memberikan pertolongan sebelum kerumah sakit.



Gambar 4.5Oryza

4.2.6 Implementasi Karakter Rica

Rica merupakan teman poka. Namun dia lebih tua dibandingkan poka. Dia anak yang baik. Dan diceritakan dia menderita penyakit kolesterol.



Gambar 4.6Rica

4.2.7 Implemetasi MainMenu

Main menu merupakan halaman menu yang akan tampil pertama kali ketika *permainan* dijalankan. Di halaman *main menu*, *button* yang bisa dipilih untuk menuju menu selanjutnya yaitu :

- *Play*, *button* ini digunakan untuk masuk ke *main permainan*.

- **Help**, button ini digunakan untuk masuk ke *help screen*
- **Credit**, button ini digunakan untuk masuk ke *credit screen*
- **Option**, button ini digunakan untuk pengaturan suara dan musik yang ada.

Implementasi *main menu* ditunjukkan oleh gambar 4.7.

4.2.8 Implementasi *help screen*

Tampilan halaman *Help* yang memberikan informasi cara bermain ditunjukkan pada gambar 4.8. Pada umumnya untuk berpindah ke halaman selanjutnya, user hanya perlu memilih tombol arah panah ke kanan. Dan memilih halaman sebelumnya dengan memilih tombol arah panah ke kiri. Namun pada permainan ini halaman *help* akan berjalan ke atas. Sehingga pemain tidak memerlukan tombol '*next*'.

4.2.9 Implementasi *credit screen*

Tampilan halaman *credit* yang memberikan informasi tentang orang-orang yang terlibat dalam pembuatan permainan ini. Implementasi *credit* akan ditunjukkan oleh gambar 4.9.

4.2.10 Implementasi *option screen*

tampilan halaman *option* adalah untuk pengaturan volume. Baik volume *sound* ataupun musik yang ada di dalam permainan. Halaman ini akan ditunjukkan oleh gambar 4.10.



Gambar 4.7 Implementasi main menu



Gambar 4.8 Implementasi Help screen



Gambar 4.9 Implementasi credit screen



Gambar 4.10 Implementasi *option screen*

4.2.11 Implementasi *Environment*

Ketika pemain mengklik tombol 'play' maka, kisah poka dalam mempelajari tanaman, diawali dengan cerita perkenalan diri. Dengan memakai latar belakang gambar poka sendiri. Disertai dengan dialog box dimana juga terdapat ekspresi wajah poka yang berbeda-beda. Tergantung dari dialog yang muncul. Setelah itu digambarkan sebuah ruangan dimana ruangan tersebut harus dibersihkan. beberapa gambar pada segmen ini tidak selalu harus menghilang. Ada beberapa gambar yang berpindah posisi agar terlihat lebih rapi. Setelah itu poka akan menemukan buku peninggalan ayahnya yang berisi buku ramuan herbal. Buku merupakan item terakhir yang akan ditemukan pada segmen ruang tamu. Akan ditunjukkan dengan gambar 4.11.



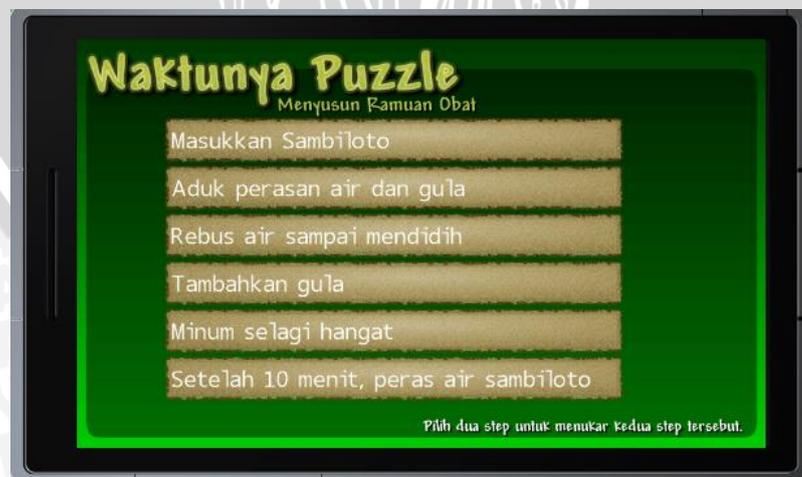
Gambar 4.11 Menemukan buku untuk mengakhiri segmen pertama

Kemudian setelah ruangan bersih dan poka menemukan buku ramuan herbal tersebut kembali menampilkan story (cerita). Dimana menceritakan poka yang sedang bercakap-cakap dengan ibunya untuk mengetahui penyakit apa yang dimiliki ibunya. Setelah selesai menemukan nama penyakit itu maka poka akan langsung menuju ke halaman belakang untuk mencari tanaman yang dibutuhkan. Ini akan ditunjukkan dengan gambar 4.12.

Setelah berhasil mengumpulkan semua tanaman yang dibutuhkan. Akan ada susunan kalimat yang acak. Baru kemudian pemain mengurutkannya menjadi satu kesatuan yang benar. Implementasinya akan ditunjukkan dengan gambar 4.13. setelah selesai mengurutkan menjadi kesatuan yang benar maka akan muncul halaman yang menyatakan bahwa pemain telah berhasil.



Gambar 4.12 Halaman Belakang Tempat Mencari Tanaman



Gambar 4.13 Puzzle yang harus Disusun

4.3 Implementasi Prosedur Program

Aplikasi multimedia pembelajaran ini memiliki beberapa proses atau *method*. Beberapa *method* yang akan dicantumkan dalam penulisan makalah skripsi ini hanya untuk algoritma dari beberapa proses (operasi) utama saja, sehingga tidak semua *method* akan dicantumkan. Adapun beberapa algoritma tersebut adalah:

4.3.1 Implementasi Untuk Menampilkan Dialog Percakapan

Dialog sisi cerita dalam aplikasi ini dibuat dengan menggunakan editor. Yang tersimpan dalam format .xml.pada editor dibuat Array 2 dimensi, di mana dimensi pertama berisi nama serta avatar karakter yang sedang berbicara sedangkan dimensi kedua berisi tentang isi percakapan tersebut. Akan ditunjukkan tabel 4.3.

Tabel 4.3Prosedur Menampilkan Dialog Percakapan

Pseudocode Dialog Percakapan
Deklarasi awal ContentManager content; SpriteFont permainanFont; Conversation conversation;
Deklarasi atribut dialog List<Texture2D> avatars = new List<Texture2D>(); List<Texture2D> backgrounds = new List<Texture2D>(); SoundEffect continueSound; Texture2D bg; Texture2D icon;
Deklarasi atribut screen Texture2D background; Texture2D rotator; Texture2D pokaFull; Vector2 rotatorOrigin; Vector2 rotatorPosition; Vector2 pokaPosition; Vector2 pokaOrigin; float angle = 0.0f;
Deklarasi xml path string xmlFileName; string levelname;

```

Deskripsi
1 //Fungsi Loadconversation
2 Mulai
3     Baca DialogArray
4     set IndeksDialog = 0
5     set Icon = DialogArray[IndeksDialog][Speaker]
6     set Dialog = DialogArray[IndeksDialog][Text]
7 Selesai
8
9 //Fungsi menampilkan background
10 Mulai
11     IF (getconversation)
12         Cetak conversation
13         jalankan rotator
14     ELSE
15         Load background
16 Selesai
    
```

4.3.2 Implementasi untuk *Hit item*

Prosedur *hit item* merupakan proses yang akan menjelaskan langkah-langkah item yang di sentuh. Sehingga dapat diketahui item yang belum ditemukan serta menambahkan skor. Implementasi Prosedur *hit item* ditunjukkan pada tabel 4.4.

Tabel 4.4Prosedur Hit Item

Pseudocode <i>hit item</i>
Deklarasi awal
<pre> ContentManager content; SpriteFont permainanFont; Conversation conversation; </pre>
Deklarasi atribut dialog
<pre> List<Texture2D> avatars = new List<Texture2D>(); List<Texture2D> backgrounds = new List<Texture2D>(); </pre>



```

SoundEffect continueSound;
Texture2D bg;
Texture2D icon;

Deklarasi atribut screen
Texture2D background;
Texture2D rotator;
Texture2D pokaFull;
Vector2 rotatorOrigin;
Vector2 rotatorPosition;
Vector2 pokaPosition;
Vector2 pokaOrigin;
float angle = 0.0f;

Deklarasi xml path
string xmlFileName;
string levelname;

Deskripsi
1 //Fungsi Hit item
2 Mulai
3     For (int i= 0 ; i<jumlah item ; i++)
4     If items[i]. ishit
5         Check = true
6         Skor + 10;
7         Sisa bahan --;
8     If sisa bahan == 0 ;
9         Last = jumlah item ;
10 Selesai
    
```

4.3.3 Implementasi untuk menampilkan *Random puzzle*

Prosedur *random puzzle* merupakan proses yang akan menjelaskan langkah-langkah merandom kepingan *puzzle*. Implementasi Prosedur *random puzzle* ditunjukkan pada tabel 4.5.

Tabel 4.5 Prosedur *Random puzzle*

Pseudocode <i>random puzzle</i>	
Deklarasi awal	
private List<PuzzlePiece>	pieces = new List<PuzzlePiece>();
private ContentManager	content;



```
private SpriteFont font;
private int xpos = 100;
private int index = 0;
private int idx1 = -1;
private int idx2 = -1;
private bool menang = false;
```

Deskripsi

```
1 //Fungsi Randompieces
2 Mulai
3     Inisialisasi class random
4     set a=jumlah pieces
5     for (int i=1; i<=a; i++)
6 pieces. Shuffle();
7 Selesai
8 //Fungsi clearselection
9 Mulai
10     foreach (PuzzlePiece piece in pieces)
11     {piece. unSelect();}
12 Selesai
```

4.3.4 Implementasi untuk Menampilkan Level

Prosedur menampilkan level merupakan proses yang akan menjelaskan langkah-langkah memilih level. Implementasi Prosedur menampilkan level ditunjukkan pada tabel 4.6.

Tabel 4.6 Prosedur Menampilkan Level

Pseudocode menampilkan level	
Deklarasi awal	
<pre>ContentManager content; SpriteFont permainanFont; SoundEffect continueSound; Texture2D background; Texture2D GaleriButton; Texture2D GaleriButtonPressed;</pre>	
Deskripsi	
1	//Fungsi inisialisasi koordinat
2	Mulai

```
3      Inisialisasi koordinat level
4  Selesai
5
6  //Fungsi memilih level
7  Mulai
8      Set posisi touch
9      If pintu 1 = cetak level 1
10     Else pintu 2 = cetak level2
11     Else pintu 3 = cetak level 3
12     Else pintu 4 = cetak level 4
13 Selesai
```

4.4 Implementasi Musik / Soundtrack

Terdapat lima musik dalam permainan Poka si tabib cilik. Adapun lima musik tersebut adalah:

1. *Pixel Peer Polka*

Musik berjudul *Pixel Peer Polka* merupakan musik utama sekaligus menjadi *opening* serta *ending* dalam permainan ini. Musik berdurasi 3 menit 22 detik ini menggambarkan ketenangan semangat yang menggebu-gebu. Karena musiknya yang mencirikan berpetualang. Diharapkan dengan musik yang ceria ini maka akan menimbulkan jiwa semangat bermain.

2. *Move Forward*

Move Forward ini merupakan musik yang melatar belakangi ketika ada story. Dengan nuansa yang membawa ketenangan dimaksudkan agar pemain memahami ketika ada cerita yang muncul di sela-sela permainan. Lagu ini berdurasi 1 menit 9 detik. Jadi musik ini akan melatar-belakangi setiap cerita.

3. *Pamgea*

Musik dengan judul *pamgea* merupakan lagu dengan penuh semangat. Musik dengan durasi 2 menit 48 detik ini muncul ketika pemain memasuki setiap segmen. Baik itu pada saat membantu ibu membersihkan rumah atau pada saat di halaman mencari tanaman.

BAB V PENGUJIAN DAN ANALISA

Bab ini membahas mengenai tahapan pengujian dan analisis permainan Poka si tabib cilik yang telah dibangun. Proses pengujian dilakukan melalui tiga tahapan yaitu pengujian unit, pengujian integrasi dan pengujian validasi. Pengujian unit dan integrasi menggunakan teknik pengujian *White Box (White Box Testing)*. Pengujian validasi menggunakan teknik pengujian *Black Box (Black Box Testing)*.

5.1 Pengujian Unit

Pada pengujian unit digunakan metode *White Box Testing* dengan teknik *Basis Path Testing*. Pada teknik *Basis Path Testing* proses pengujian dilakukan dengan memodelkan algoritma pada sebuah *flow graph*, menentukan *cyclometric complexity* dan melakukan uji kasus untuk setiap path yang ada.

5.1.1 Pengujian Unit Untuk Dialog Percakapan

Dialog merupakan komponen penting dalam permainan ini untuk memberikan cerita. Fungsi utama dialog percakapan ini disimpan dalam file .xml. Sedangkan keseluruhan dialog percakapan disimpan dalam folder conversation. Pemodelan dialog percakapan dalam bentuk *flow graph* ditunjukkan pada Tabel 5.1. Flow graph pengujian ditunjukkan pada Gambar 5.1. Tabel hasil pengujian ditunjukkan pada Tabel 5.2.

Tabel 5.1 Permodelan *Flow graph* Algoritma Dialog Percakapan

Pseudocode Dialog Percakapan
<p>DEKLARASI AWAL</p> <p>DialogArray[][]: Array</p>
<p>DESKRIPSI</p> <p>//Fungsi Loadconversation</p> <p>Mulai</p>

```

Baca DialogArray
set IndeksDialog = 0
set Icon = DialogArray[IndeksDialog][Speaker]
set Dialog = DialogArray[IndeksDialog][Text]
Selesai
    
```

1

Gambar 5.1 Flow graph Pengujian Dialog Percakapan

Pemodelan ke dalam *flow graph* yang telah dilakukan untuk dialog percakapan menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis, E merupakan sisi atau *edge* (garis penghubung antar *node*) dan N merupakan jumlah simpul (*node*).

$$\begin{aligned}
 V(G) &= E - N + 2 \\
 &= 0 - 1 + 2 \\
 &= 1
 \end{aligned}$$

Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan maka ditentukan tiga buah basis set dari jalur *independent*, yaitu :

Jalur 1 : 1

Tabel 5.2 Hasil pengujian Dialog Percakapan

Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Menampilkan percakapan	Dialog percakapan muncul	Dialog percakapan muncul

5.1.2 Pengujian Unit Fungsi Menampilkan background

Pengujian untuk menampilkan gambar background dari *gameplay* yang ada. Ditunjukkan pada Tabel 5.3. Flow graph pengujian ditunjukkan pada Gambar

5.2. Tabel hasil pengujian ditunjukkan pada Tabel 5.4. sedangkan didalam pseudocode terdapat beberapa tambahan source code dari penghapusan fungsi puzzle. Akan ditunjukkan oleh Tabel 5.5.

Tabel 5.3 Permodelan *flow graph* menampilkan fungsi *background*



Gambar 5.2 *flow graph* fungsi menampilkan background

Pemodelan ke dalam *flow graph* yang telah dilakukan untuk fungsi menampilkan background menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis, E merupakan sisi atau *edge* (garis penghubung antar *node*) dan N merupakan jumlah simpul (*node*).

$$\begin{aligned}
 V(G) &= E - N + 2 \\
 &= 4 - 4 + 2 \\
 &= 2
 \end{aligned}$$

Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan maka ditentukan satu buah basis set dari jalur *independent*, yaitu :

Jalur 1 : 1 2 4

Jalur 2 : 1 3 4

Tabel 5.4 Hasil pengujian Menampilkan background

Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Get conversation true	Menampilkan conversation	Menampilkan conversation
2	Get conversation false	Menampilkan background	Menampilkan background

Tabel 5.5 Clear Puzzle

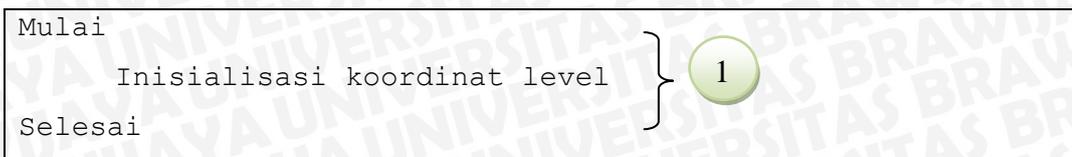
Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Iterasi telah sampai batas iterasi	Clear Pieces puzzle	Clear Pieces puzzle
2	Iterasi tidak memenuhi syarat	Puzzle tidak bisa ter 'clear'	Puzzle tidak bisa ter 'clear'

5.1.3 Pengujian Unit Inisialisasi koordinat Level

Pada tahap ini dilakukan proses mendapatkan koordinat level. Akan ditunjukkan oleh tabel 5.6 dan table 5.7 menunjukkan hasil dari pengujian.

Tabel 5.6 Koordinat Level

Pseudocode inisialisasi koordinat level
DEKLARASI AWAL hitungJawaban: Number
DESKRIPSI



Pemodelan ke dalam *flow graph* yang telah dilakukan untuk mendapatkan koordinat level menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis, E merupakan sisi atau *edge* (garis penghubung antar *node*) dan N merupakan jumlah simpul (*node*).

$$\begin{aligned}
 V(G) &= E - N + 2 \\
 &= 0 - 1 + 2 \\
 &= 1
 \end{aligned}$$

Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan maka ditentukan tiga buah basis set dari jalur *independent*, yaitu :

Jalur 1 : 1

Tabel 5.7 Hasil Pengujian Mendapatkan Koordinat Level

Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Mendapatkan koordinat level	Mendapatkan titik koordinat yang diinginkan	Mendapatkan titik koordinat yang diinginkan

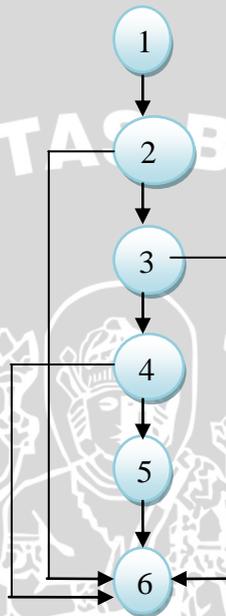
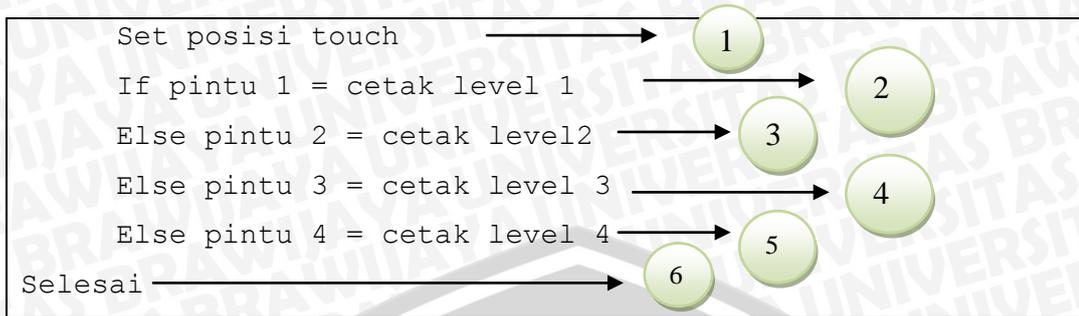
5.1.4 Pengujian Unit Memilih Level

Pada tahap ini dilakukan proses memilih level. Dimana terdapat empat level. Akan ditunjukkan oleh tabel 5.8. Flow graph pengujian ditunjukkan pada Gambar 5.3. Tabel hasil pengujian ditunjukkan pada Tabel 5.9.

Tabel 5.8 Memilih Level

Pseudocode memilih level
DEKLARASI AWAL
DESKRIPSI
//Fungsi memilih level
Mulai





Gambar 5.3flow graph memilih level

Pemodelan ke dalam *flow graphy* yang telah dilakukan untuk memilih level menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis, E merupakan sisi atau *edge* (garis penghubung antar *node*) dan N merupakan jumlah simpul (*node*).

$$\begin{aligned}
 V(G) &= E - N + 2 \\
 &= 8 - 6 + 2 \\
 &= 4
 \end{aligned}$$

Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan maka ditentukan tiga buah basis set dari jalur *independent*, yaitu :

- Jalur 1 : 1 – 2 – 6
- Jalur 2 : 1 – 3 – 6
- Jalur 3 : 1 – 4 – 6



Jalur 4 : 1 – 5 – 6

Tabel 5.9 Hasil Pengujian Memilih Level

Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Pintu 1 = true	Masuk ke level 1	Masuk level 1
2	Pintu 2 = true	Masuk ke level 2	Masuk level 2
3	Pintu 3 = true	Masuk level 3	Masuk level 3
4	Pintu 4 = true	Masuk level 4	Masuk level 4

5.2 Analisis Pengujian Unit

Hasil analisis yang didapatkan dari pengujian unit yaitu:

1. Berdasarkan kesesuaian antara hasil pengujian tiap unit dengan output unit yang diharapkan pada permainan Poka si tabib cilik, maka dapat diambil kesimpulan bahwa unit modul dari program sudah sesuai dengan output yang diharapkan.
2. Berdasarkan hasil perhitungan *cyclomatic complexity* dari tiap *flow graph* kode unit, kode unit yang menghasilkan jalur kasus uji terbanyak adalah kode operasi untuk menghasilkan nilai acak sebanyak 4 kasus uji. Jumlah kasus uji kode ini sebagian besar dipengaruhi dari seleksi kondisi.

5.3 Pengujian Integrasi

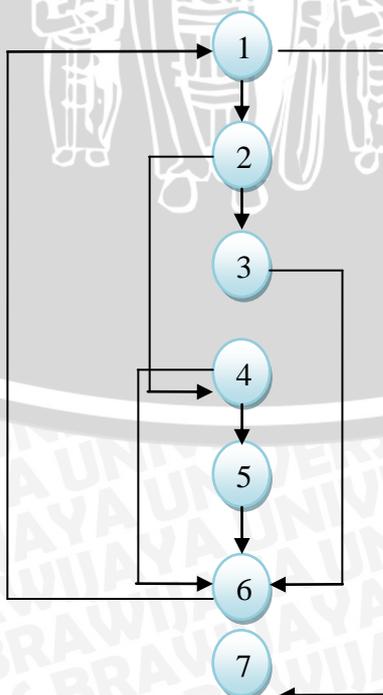
Pengujian integrasi diterapkan pada method yang mengintegrasikan fungsionalitas dari *class-class* lain untuk melakukan sebuah operasi tertentu. Pada pengujian integrasi perangkat lunak inidigunakan teknik *White-Box Testing* dengan teknik *Basis Path Testing*. Pada teknik *Basis Path Testing*, proses pengujian dilakukan dengan memodelkan algoritma pada suatu *flow graph*, menentukan jumlah kompleksitas siklomatis (*cyclomatic complexity*), menentukan sebuah basis set dari jalur independen dan memberikan kasus uji (*test case*) pada setiap basis set yang telah ditentukan.

5.3.1 Pengujian *Hit item*

Pada tahap ini dilakukan proses pengujian hit item. Ditunjukkan oleh tabel 5.10. Flow graph pengujian ditunjukkan pada Gambar 5.4. Tabel hasil pengujian ditunjukkan pada Tabel 5.11.

Tabel 5.10 Pengujian *Hit item*

Pseudocode <i>hit item</i>	
1	//Fungsi <i>Hit item</i>
2	Mulai
3	For (int i= 0 ; i<jumlah item ; i++)
4	If items[i]. ishit
5	Check = true
6	Skor + 10;
7	Sisa bahan --;
8	If sisa bahan == 0 ;
9	Last = jumlah item ;
10	End
11	End
12	Selesai



Gambar 5.4 *Flow Graph* Hit Item

Pemodelan ke dalam *flow graph* yang telah dilakukan untuk *hit item* menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis, E merupakan sisi atau *edge* (garis penghubung antar *node*) dan N merupakan jumlah simpul (*node*).

$$\begin{aligned} V(G) &= E - N + 2 \\ &= 9 - 7 + 2 \\ &= 4 \end{aligned}$$

Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan maka ditentukan tiga buah basis set dari jalur *independent*, yaitu :

- Jalur 1 : 1 - 2 - 3 - 6 - 1 -
- Jalur 2 : 1 - 2 - 4 - 5 - 6 - 1 -
- Jalur 3 : 1 - 2 - 4 - 6 - 1 -
- Jalur 4 : 1 - 7

Fungsi utama untuk memuat musik tersimpan dalam *main.as*. sedangkan perintah untuk melakukan pemanggilan fungsi ini dan meload musik terdapat di frame-frame setiap scene. Pemodelan dalam pembuatan fungsi latihan soal dengan bentuk *flow graph* ditunjukkan dalam Tabel 5.11.

Tabel 5.11 Hasil Pengujian *Hit item*

Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Pengecekan apakah item memenuhi syarat. Jika item = true	Item yang dicari berkurang dan skor bertambah	Item yang dicari berkurang dan skor bertambah
2	Pengecekan sisa bahan. Sisa bahan = 0	Jumlah item yang dicari terpenuhi	Jumlah item yang dicari terpenuhi
3	Pengecekan apa-	Terus dilakukan looping	Terus dilakukan looping

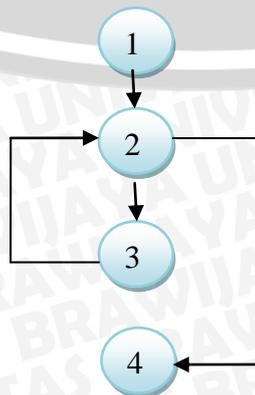
	kah item memenuhi syarat. Jika item = false		
4	Pengecekan apakah syarat iterasi terpenuhi.	Proses <i>hit item</i> selesai dilakukan	Proses <i>hit item</i> selesai dilakukan

5.3.2 Pengujian Unit *Random puzzle*

Prosedur *random puzzle* adalah prosedur yang digunakan untuk mengacak pieces dari *puzzle*. Pemodelan prosedur *random puzzle* dengan bentuk *flow graph* ditunjukkan dalam Tabel 5.12, gambar 5.5. dan hasil pengujian akan ditunjukkan table 5.13.

Tabel 5.12 Prosedur *Random puzzle*

Pseudocode Prosedur random puzzle	
DEKLARASI AWAL	
Ran: Number	
DESKRIPSI	
Mulai	
Inisialisasi class random	
set a=jumlah pieces	
for (int i=1; i<=a; i++)	
pieces. Shuffle();	
Selesai	



Gambar 5.5 *flow graph* random puzzle

Pemodelan ke dalam *flow graph* yang telah dilakukan untuk algoritma *merandom puzzle* menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis, E merupakan sisi atau *edge* (garis penghubung antar *node*) dan N merupakan jumlah simpul (*node*).

$$\begin{aligned} V(G) &= E - N + 2 \\ &= 4 - 4 + 2 \\ &= 2 \end{aligned}$$

Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan maka ditentukan delapan buah basis set dari jalur *independent*, yaitu :

- Jalur 1 : 1-2-3-2 ...
- Jalur 2 : 1-2-4

Tabel 5.13 Pengujian *random puzzle*

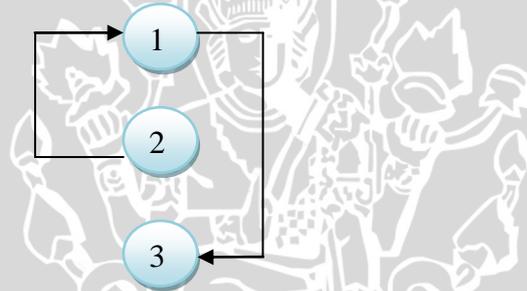
Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Iterasi telah sampai batas iterasi	Pieces <i>puzzle</i> tersusun dengan benar	Pieces <i>puzzle</i> tersusun dengan benar
2	Iterasi tidak memenuhi syarat	<i>Puzzle</i> tidak tersusun dengan benar	<i>Puzzle</i> tidak tersusun dengan benar

5.3.3 Pengujian Unit *Clear Selection*

Prosedur *clear selection* adalah prosedur yang digunakan untuk menghapus *puzzle* yang telah selesai dipilih. Pemodelan prosedur *clear selection* dengan bentuk pseudocode dan *flow graph* ditunjukkan dalam Tabel 5.14, gambar 5.6.

Tabel 5.14 Prosedur pengujian Clear selection

Pseudocode <i>clear selection</i>	
DEKLARASI AWAL	
DialogArray[][]: Array	
DESKRIPSI	
//Fungsi clearselection	
Mulai	
foreach (<i>PuzzlePiece</i> piece in pieces)	→ 1
{piece.unSelect();}	→ 2
Selesai	→ 3



Gambar 5.6flow graph clear selection

Pemodelan ke dalam *flow graph* yang telah dilakukan untuk algoritma *clear selection* menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis, E merupakan sisi atau *edge* (garis penghubung antar *node*) dan N merupakan jumlah simpul (*node*).

$$\begin{aligned}
 V(G) &= E - N + 2 \\
 &= 3 - 3 + 2 \\
 &= 2
 \end{aligned}$$

Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan maka ditentukan delapan buah basis set dari jalur *independent*, yaitu :

Jalur 1 : 1-2-1

Jalur 2 : 1-3



Tabel 5.15 Hasil pengujian Clear Selection

Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Iterasi telah sampai batas iterasi	Pieces <i>puzzle</i> dihapus	Pieces <i>puzzle</i> dihapus
2	Iterasi tidak memenuhi syarat	<i>Puzzle</i> terus ditukar	<i>Puzzle</i> terus ditukar

5.4 Analisis Pengujian Integrasi

Proses analisis terhadap hasil pengujian integrasi dilakukan dengan melihat kesesuaian beberapa unit modul yang menyusun satu blok fungsi dalam permainan. Dari seluruh pengujian integrasi yang dilakukan, jumlah jalur pada logika setiap prosedur telah sesuai dengan perhitungan cyclomatic complexity. Setiap kasus uji yang dibuat berdasarkan jumlah jalur tersebut telah diuji dan memberikan hasil sesuai dengan hasil yang diharapkan. Berdasarkan hal tersebut, maka dapat diambil kesimpulan bahwa unit modul dari program sudah memenuhi kebutuhan fungsional yang telah dirancang pada tahap perancangan.

5.5 Pengujian Validasi

Pengujian validasi digunakan untuk mengetahui apakah sistem yang dibangun sudah benar sesuai dengan yang dibutuhkan. Daftar kebutuhan akan menjadi acuan untuk melakukan pengujian validasi. Pengujian validasi menggunakan metode pengujian *Black Box*, karena tidak diperlukan konsentrasi terhadap alur jalannya algoritma program

5.5.1 Kasus uji validasi dan hasil pengujian

Keseluruhan kasus uji validasi dan hasil pengujian dijelaskan pada Tabel 5.16.

Tabel 5.16 Kasus dan Hasil Uji Validasi

No.	Kasus Uji	Objek Uji	Prosedur dan Input	Kondisi yang diharapkan	Hasil
1	Melihat splash screen.	F01	Memulai <i>run-ning</i> atau eksekusi program.	Aplikasi dapat menampilkan splash screen.	Valid
2	Melihat menu utama.	F02	Setelah menampilkan splash screen.	Aplikasi dapat menampilkan menu utama.	Valid
3	Melihat cerita	F03	Pemain memilih tombol "Play" pada menu utama.	Aplikasi dapat menampilkan karakter poka.	Valid
4	Melihat bantuan	F04	Pemain memilih tombol "help" pada menu utama.	Aplikasi menampilkan <i>help screen</i> .	Valid
5	Melihat credit menu	F05	Pemain memilih tombol "credit" pada menu utama.	Aplikasi menampilkan credit screen.	Valid
6	Menjalankan permainan utama	F06	Setelah selesai menampilkan cerita muncul gambar ruang tamu.	Mengklik barang-barang yang ditunjukkan pada box.	Valid
7	Melihat GUI	F07	Pemain bisa melihat skor yang diperoleh, waktu yang tersisa	Aplikasi menampilkan GUI screen.	Valid

			serta barang yang dicari		
8	Menyusun <i>Puzzle</i>	F08	Pemain bisa memilih dua pieces sehingga bisa ditukar.	Aplikasi bisa menukar dua pieces yang dipilih.	Valid
9	Melihat skor pemain	F09	Pemain memilih kran untuk mengisi bak mandi hingga penuh dengan mouse	Sistem menampilkan animasi air yang mengalir sesuai dengan kran yang dipilih hingga air penuh.	Valid
10	Melakukan setting suara musik.	F10	Pemain memilih tombol "option" pada menu utama.	Aplikasi akan muncul slider yang digunakan untuk pengaturan volume musik.	Valid
11	Membaca Dialog Percakapan	F11	Pemain bisa membaca dialog antar karakter dan melanjutkan percakapan.	Aplikasi menampilkan dialog percakapan antar karakter per huruf hingga dialog tersebut selesai.	Valid

5.6 Analisis Pengujian Validasi

Berdasarkan kesesuaian antara hasil uji terhadap implementasi dan fungsionalitas permainan Poka si tabib cilik dengan hasil yang diharapkan dalam

daftar kebutuhan permainan, melalui pengujian menggunakan *Emulator Windows Phones* serta *device* didapatkan keseluruhan fungsionalitas permainan telah memenuhi kebutuhan yang telah dijabarkan dalam daftar kebutuhan. *Device* yang paling kompatibel dengan permainan poka si tabib cilik adalah windows phone dengan spesifikasi os. 7.0 keatas.

5.7 Pengujian Terhadap Pengguna

Pengujian terhadap pengguna dilakukan dengan cara memberikan kuesioner kepada responden. Pengujian dilaksanakan pada saat dilakukan pameran skripsi, jumlah responden yang didapatkan adalah sepuluh orang. Target pengguna dalam penelitian ini adalah siapa saja yang memainkan game ini dengan syarat minimal sudah dapat membaca dengan baik. Tujuan utama dari pemberian kuesioner ini adalah untuk mendapatkan respon pengguna sehubungan dengan penggunaan aplikasi ini. Adapun langkah-langkah yang dilakukan dalam mengambil data untuk keperluan pengujian pengguna di sini yaitu responden diberikan kuesioner pada Lampiran L-1 untuk diisi. Setelah itu responden dicobakan untuk bermain game. Setelah responden selesai bermain, dilanjutkan penyelesaian pengisian kuesioner oleh responden. Hasil pengujian kuesioner terdapat pada Lampiran L-2.

5.8 Analisis Pengujian Pengguna

Kuesioner diberikan kepada keseluruhan total responden sebanyak 10 orang. Umur koresponden rata-rata 22 tahun. Adapun hasil yang didapat sesuai dengan kuesioner terdapat pada lampiran L-2 sedangkan analisis dari kuesioner tersebut adalah sebagai berikut:

1. Dari pertanyaan kuesioner yang pertama tentang bermain *game* didapat bahwa 60% responden sangat sering bermain *game*, 10% responden kadang-kadang, 20% responden mengatakan dia sering bermain *game* dan 10% mengatakan tidak pernah bermain *game*. Analisis terhadap pengguna membuktikan bahwa semua responden pernah bermain

gamemenggunakanmobile (misalnya tablet, telepon genggam dan semacamnya).

2. Dari pertanyaan kuesioner yang kedua apakah responden pernah menggunakan pengobatan tradisional ketika sakit didapat bahwa 60% pernah melakukan pengobatan secara tradisional. Dimana pada kolom penjelasan mereka menggunakan beberapa tanaman herbal juga, dan ada yang secara pijat. 40% mengatakan tidak pernah.
3. Dari pertanyaan kuesioner yang ketiga tentang apakah responden merasa bahwa mendapatkan informasi setelah memainkan game ini. Hasilnya adalah 100% responden mengatakan bahwa setelah memainkan game ini mereka merasa mendapatkan informasi.
4. Dari pertanyaan kuesioner yang keempat tentang apakah game ini bermanfaat memberitahukan fungsi dari tanaman herbal itu sendiri dan 90% mengatakan bahwa ya, mereka merasa terbantu. Dan 10% mengatakan lumayan.
5. Dari pertanyaan kuesioner yang keenam tentang konsep cara belajar sambil bermain *game* seperti contoh *game* yang dibuat didapat bahwa 100% dari responden menyatakan menyukai konsep tersebut.

Dari keseluruhan pengujian dan analisa terhadap pengguna didapat kesimpulan bahwa aplikasi ini telah memenuhi kebutuhan dan sesuai dengan tujuan yaitu mengenalkan tanaman herbal.

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

“Poka si tabib cilik” adalah sebuah permainan edukasi yang membantu pemain dalam memahami tanaman herbal dengan memberikan kisah petualangan yang disajikan dalam bentuk jalan cerita. Berdasarkan hasil pengamatan selama perancangan, implementasi, dan proses pengujian permainan yang dilakukan, maka diambil kesimpulan sebagai berikut:

1. “Poka si tabib cilik” telah berhasil di implementasikan sebagai permainan pengenalan tanaman herbal. Dimana unsur pengenalan terdapat pada art segmen dua. Permainan edukasi ini menyuguhkan konten multimedia dahulu baru kemudian menyajikan pengenalan tanaman herbalnya sesuai dengan cerita. Terdapat beberapa segmen pada permainan. Yaitu pemain harus mencari bahan-bahan yang dibutuhkan dahulu baru kemudian mengolahnya menjadi minuman herbal atau yang biasa dikenal dengan jamu.
2. Perancangan permainan edukasi “Poka si tabib cilik” terdiri dari pembuatan *game concept* dan *technical design*. Implementasi permainan edukasi “Poka si tabib cilik” menggunakan microsoft XNA dengan bahasa pemrograman C#. Untuk membuat asset 2D digunakan *adobe flash* dan *adobe fireworks*.
3. Pengujian unit permainan edukasi “Poka si tabib cilik” dengan metode *White box* menghasilkan kesimpulan bahwa unit modul dari program sudah sesuai dengan output yang diharapkan dan kode unit dalam pengujian yang menghasilkan kasus uji terbanyak adalah pengujian unit untuk menghasilkan nilai acak yaitu sebanyak empat kasus uji.

6.2 Saran

Saran untuk pengembangan permainan “Poka si Tabib Cilik” lebih lanjut antara lain:

1. Dapat dilakukan pengembangan dengan membuat versi PC, tablet dsb. Aplikasi ini dikembangkan hanya untuk platform *mobile* dengan sistem operasi *windows phone*. Dengan adanya pengembangan dengan membuat pengguna bisa memainkan dimana saja karena bersifat *portable*.
2. Penambahan fungsi jumlah tanaman herbal itu sendiri. Serta mengembangkan lagi beberapa khasiat tiap masing-masing.
3. Penambahan fungsi “skip” pada cerita.
4. Penambahan jumlah level yang lebih sulit lagi. Penambahan level bisa disesuaikan dengan tingkat kesulitan tanaman, atau baik dari waktu sendiri. Serta penambahan *reward* untuk beberapa kondisi.



DAFTAR PUSTAKA

- [HAM-06] Hamilton, Kim & Miles, Russell. 2006. *Learning UML 2.0*. O'Reilly Media, Inc.
- [HAR-04] Hariyanto, Bambang. 2004. *Rekayasa Sistem Berorientasi Objek*. Informatika Bandung.
- [LIU-06] Liu, Liping & Roussev, Boris. 2006. *Management of The Object-Oriented Development Process*. Idea Group Publishing.
- [MUS-12] mustofa kamal. 2012 akses dari <http://students.netindonesia.net/blogs/mustofakamal/> tanggal akses 19 April 2013.
- [PRE-10] Pressman, Roger. 2009. *Software Engineering: A Practioner's Approach, 7th Edition*. Mc Graw-Hill.
- [REP-08] Pedersen, Roger E. 2008. *Game Design Foundations, 2nd Edition*. Wordware Publishing, Inc.
- [ROG-10] Roger, Scott. 2010, *Level UP! The Guide to Great Video Game Design*, Wiley: West Sussex.
- [SCH-08] Schell, Jesse. 2008. *The Art of Game Design*. Morgan Kaufmann Publishers.
- [SOM-11] Sommerville, Ian. 2011. *Software Engineering 9th Edition*. Pearson.
- [SUK-06] Sukandar E Y. 2006. *Tren dan Paradigma Dunia Farmasi, Industri-Klinik-Teknologi Kesehatan*. Departemen Farmasi, FMIPA, Institut Teknologi Bandung.
- [UUR-03] Undang-Undang Republik Indonesia Nomor 20 Tahun 2003 tentang Sistem Pendidikan Nasional. 2004. Jakarta.
- [WER-10] Werner, David and Friends. 2010. *where there is no doctor*. Andi Publisher.
- [WIJ-08] Wijayakusima, Hembing. 2008. *Ramuan Lengkap Herbal Takhlukan Penyakit*. Niaga Swadaya. INA

LAMPIRAN

Nama :
 Umur :
 Pekerjaan :

TandaTangan

Soal kuisoner

1. Seberapa sering kamu bermain game di telepon genggammu ?
 - a. Sangat sering
 - b. Kadang-kadang
 - c. sering
 - d. tidak pernah
2. Apakah kamu pernah mencoba pengobatan tradisional ketika sakit ?
 - a. Pernah (jelaskan)
 - b. tidak

.....

.....
3. Apakah menurut kamu setelah memainkan game ini memberikan informasi tentang tanamanherbal ?
 - a. Ya
 - b. tidak
4. Apakah menurut kamu game ini bermanfaat memberitahukan fungsi dari tanaman herbal itu sendiri?
 - a. Ya
 - b. Lumayan
 - c. Sedikit
 - d. Tidak
5. Apakah kamu suka dengan konsep belajar sambil bermain seperti pada game ini ?
 - a. Suka
 - b. Kurangsuka
 - c. Biasa saja
 - d. Tidak tahu

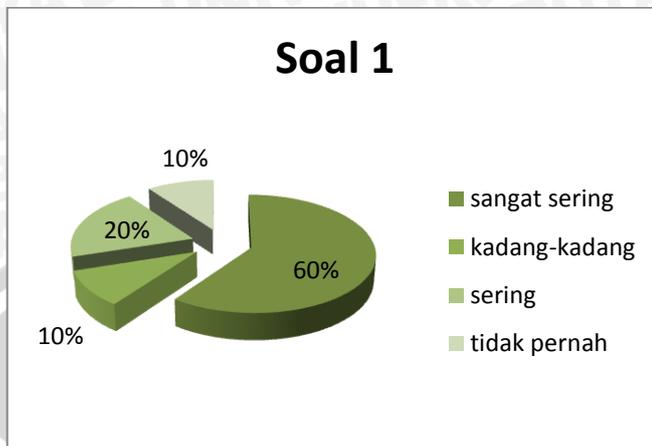
Saran :



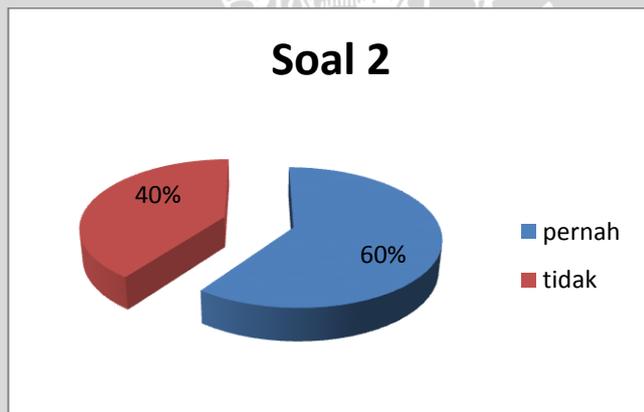
Kuesioner No	Jawaban	Jumlah	Prosentase
1	Sangat Sering	6	60%
	Kadang-kadang	1	10%
	Sering	2	20%
	Tidak pernah	1	10%
	TOTAL	10	100%
2	Pernah	6	60%
	Tidak	4	40%
	TOTAL	10	100%
3	Ya	10	100%
	Tidak	0	0%
	TOTAL	10	100%
4	Ya	9	90%
	Lumayan	1	10%
	Sedikit	0	0%
	Tidak	0	0%
	TOTAL	10	100%
5	Suka	10	100%
	Kurang suka	0	0%
	Biasa saja	0	0%
	Tidak tahu	0	0%
	TOTAL	10	100%

Lampiran 3 hasil kuisioner

1. Seberapa sering kamu bermain game di telepon genggammu ?



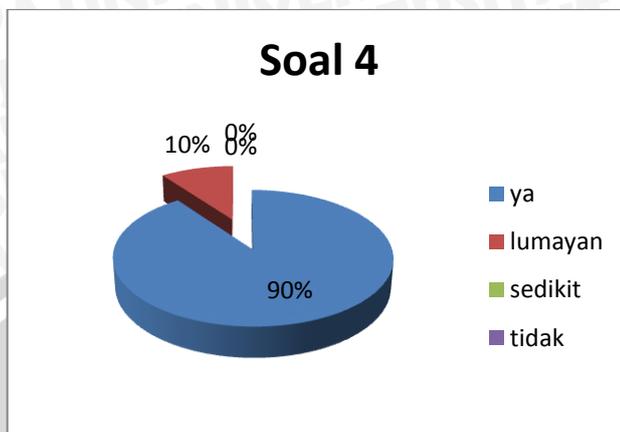
2. Apakah kamu pernah mencoba pengobatan tradisional ketika sakit ?



3. Apakah menurut kamu setelah memainkan game ini memberikan informasi tentang tanamanherbal ?



4. Apakah menurut kamu game ini bermanfaat memberitahukan fungsi dari tanaman herbal itu sendiri?



5. Apakah kamu suka dengan konsep belajar sambil bermain seperti pada game ini ?

