

USULAN TUGAS AKHIR

1 IDENTITAS PENGUSUL

NAMA : Muhammad Agil Mahbuby
NRP : 5110100191
DOSEN WALI : Prof. Ir. Joko Lianto Buliali, M.Sc., Ph.D.
DOSEN PEMBIMBING : 1. Imam Kuswardayan, S.Kom., M.T.
2. Abdul Munif, S.Kom., M.Sc.

2 JUDUL TUGAS AKHIR

“Rancang Bangun *Web Service* untuk Implementasi Aturan Main dan Manajemen Transaksi pada Permainan Sosial Perangkat Bergerak *Card Warlock Saga*”

3 LATAR BELAKANG

Permainan video telah berkembang cepat dan modern, di mana dahulu permainan video hanya dimainkan oleh beberapa kalangan saja, sekarang permainan video hampir dimainkan oleh seluruh kalangan. Permainan video pada saat ini juga menyertakan teknologi *broadband* yang berguna untuk mengakses internet.

Teknologi informasi yang berkembang dengan sangat cepat, memungkinkan permainan video untuk dimainkan di perangkat bergerak seperti telepon genggam pintar. Perkembangan sistem operasi Android sebagai salah satu sistem operasi telepon genggam yang populer menyebabkan banyaknya pengguna telepon genggam berbasis sistem operasi tersebut.

Saat ini mulai bermunculan banyak situs-situs penyedia layanan jejaring sosial seperti Facebook dan Twitter. Dengan jumlah pengguna yang cukup banyak, dimungkinkan

untuk menciptakan permainan video yang terintegrasi dengan jejaring sosial yang dinamakan dengan permainan sosial.

Role Playing Game (RPG) adalah permainan yang pemainnya memainkan peran tokoh-tokoh khayalan dan berkolaborasi untuk merajut sebuah cerita bersama. Para pemain memilih aksi tokoh–tokoh mereka berdasarkan karakteristik tokoh tersebut, dan keberhasilan aksi mereka tergantung dari sistem peraturan permainan yang telah ditentukan .

4 RUMUSAN MASALAH

Rumusan masalah yang akan diangkat dalam Tugas Akhir ini dapat dipaparkan sebagai berikut.

1. Bagaimana membuat aplikasi *web service* yang dapat berjalan pada permainan sosial *Card Warlock Saga*?
2. Bagaimana merancang dan mengimplementasikan aturan main dan manajemen transaksi pada permainan sosial *Card Warlock Saga* dalam bentuk *stored procedure*?
3. Bagaimana membuat arsitektur *web service* dan fungsi-fungsinya agar dapat diakses oleh modul–modul lain pada permainan sosial *Card Warlock Saga*?
4. Bagaimana membuat manajemen transaksi basis data pada permainan sosial *Card Warlock Saga* agar dapat diakses secara *massive*?
5. Bagaimana mengimplementasikan aturan main dan manajemen transaksi agar dapat diuji pada *layer database* sebelum diintegrasikan pada modul-modul lain?

5 BATASAN MASALAH

Permasalahan yang dibahas dalam Tugas Akhir ini memiliki beberapa batasan, di antaranya adalah sebagai berikut.

1. Bahasa pemrograman yang digunakan untuk membangun *web service* ini adalah PHP.
2. *Web service* yang dibangun menggunakan MySQL sebagai sistem manajemen basis data.
3. *Web service* yang dibangun hanya digunakan untuk permainan *Card Warlock Saga*.

6 TUJUAN PEMBUATAN TUGAS AKHIR

Tujuan dari pembuatan Tugas Akhir ini adalah untuk membuat *web service* yang dapat diintegrasikan dengan modul–modul lain pada permainan sosial *Card Warlock Saga*.

7 MANFAAT TUGAS AKHIR

Manfaat dari tugas akhir ini adalah untuk menyempurnakan modul–modul yang ada pada permainan sosial *Card Warlock Saga*, sehingga permainan ini dapat digunakan sebagai sarana hiburan, dan sebagai media untuk melakukan interaksi sosial secara virtual.

8 TINJAUAN PUSTAKA

8.1 Unity

Unity merupakan sebuah ekosistem pengembangan permainan yang terintegrasi, dan kaya akan alat atau perlengkapan yang sangat berguna untuk membangun permainan interaktif seperti pencahayaan, efek khusus, animasi dan mesin fisika. Unity dapat digunakan untuk membangun permainan dengan dukungan grafis tiga dimensi atau dua dimensi. Unity juga dapat digunakan untuk melakukan perubahan maupun menguji secara bersamaan pada permainan yang dibuat, dan ketika sudah siap permainan dapat dipublikasikan ke berbagai macam perangkat yang mendukung seperti Mac, PC, Linux, Windows Store, Windows Phone 8, Android, Blackberry 10, Wii U, PS3 dan Xbox 360 [1].

8.2 Game Sosial

Game sosial adalah permainan yang dimainkan dengan teman-teman yang berada pada suatu jaringan sosial sehingga pemain bisa mengundang atau mengajak teman untuk melakukan aktivitas tertentu dalam *game* [2]. *Game* sosial pada umumnya bersifat *asynchronous* di mana untuk memainkannya pemain tidak perlu *online* secara bersama–sama serta adanya fitur *multiplayer*.

Fitur-fitur dalam permainan sosial yaitu antara lain adalah sebagai berikut [3].

- *Asynchronous gameplay*, memungkinkan permainan dapat diselesaikan tanpa perlu bantuan pemain lain yang berada dalam jaringan untuk bermain pada waktu yang sama.
- Komunitas, salah satu fitur pembeda antara permainan sosial daring dan permainan daring lainnya. *Quest* atau objektif dari permainan hanya dapat dilakukan jika pemain *men-share* permainannya dengan teman-teman mereka (terhubung melalui jaringan sosial seperti Facebook) atau membuat pemain lain bermain sebagai *allies* mereka.
- Tidak ada kondisi menang. Kebanyakan pengembang mengandalkan pengguna permainan mereka untuk sering bermain, biasanya tidak ada kondisi menang. Artinya, permainan tidak pernah berakhir dan tidak ada yang pernah dinyatakan sebagai *winner*. Sebaliknya, permainan memiliki banyak *quest* atau yang dapat diselesaikan.
- Mata uang virtual. Permainan sosial daring menggunakan mata uang virtual (misalnya *gold*), di mana pemain harus membelinya dengan uang

asli. Penggunaan mata uang virtual biasanya digunakan untuk membeli *upgrade* yang akan memakan waktu lebih lama jika didapatkan melalui *game achievement*. Dalam banyak kasus, beberapa *upgrade* hanya tersedia melalui mata uang virtual.

Permainan sosial dibuat agar pemain mendapatkan hiburan dan berinteraksi dengan pemain pada saat memainkannya. Permainan yang terlalu mudah maupun terlalu sulit akan membuat pemain cepat merasa bosan. Oleh karena itu, perlu dilakukan perencanaan yang matang dalam mendesain permainan. Perencanaan permainan dimulai dengan perancangan alur cerita, tingkat kesulitan pada tiap level permainan, *gameplay*, dan sebagainya.

8.3 Prosedur Penyimpanan Data

Prosedur penyimpanan data atau biasa disebut dengan *stored procedure* adalah suatu kumpulan fungsi objek yang tersimpan dalam basis data dan dapat digunakan untuk menggantikan berbagai kumpulan perintah yang sering digunakan [4]. *Stored procedure* sangat berguna ketika pengembang aplikasi tidak ingin pengguna mengakses tabel pada basis data secara langsung (membatasi hak akses pengguna) dan mencatat operasi yang dilakukan sehingga resiko kerusakan data dapat diminimalisir.

8.4 MySQL

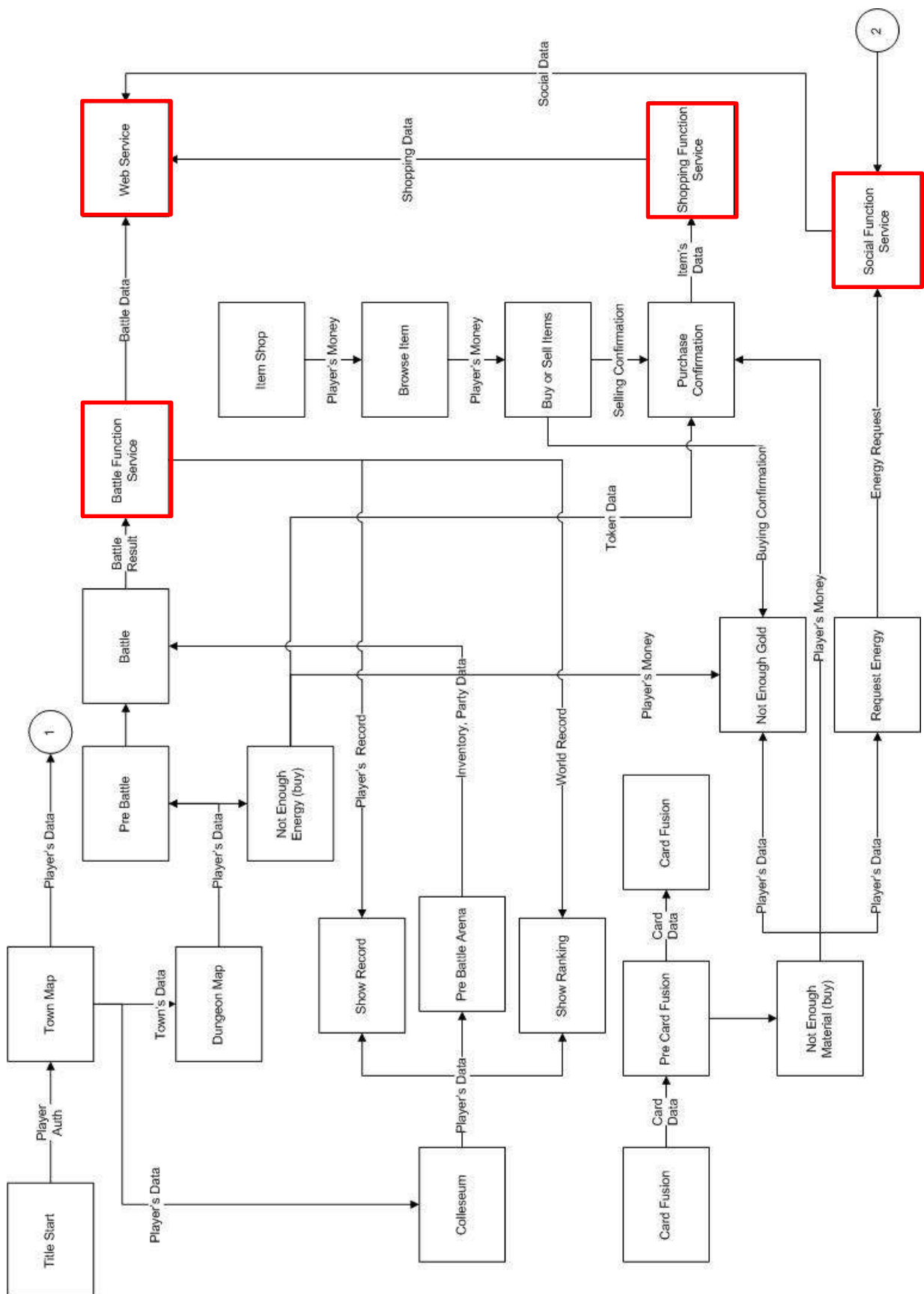
MySQL adalah sebuah perangkat lunak sistem manajemen basis data relasional [5]. MySQL dibuat oleh TcX dan telah dipercaya mengelola sistem dengan 40 buah basis data yang berisi 10.000 tabel dan 500 di antaranya memiliki 7 juta baris.

MySQL AB merupakan perusahaan komersial Swedia yang memiliki MySQL. Pendiri MySQL AB adalah dua orang Swedia yang bernama David Axmark dan Allan Larsson serta satu orang Finlandia yang bernama Michael Monty. Setiap pengguna MySQL dapat menggunakannya secara bebas yang didistribusikan secara gratis di bawah lisensi GPL (*General Public License*), namun tidak boleh dijadikan produk turunan yang bersifat komersial.

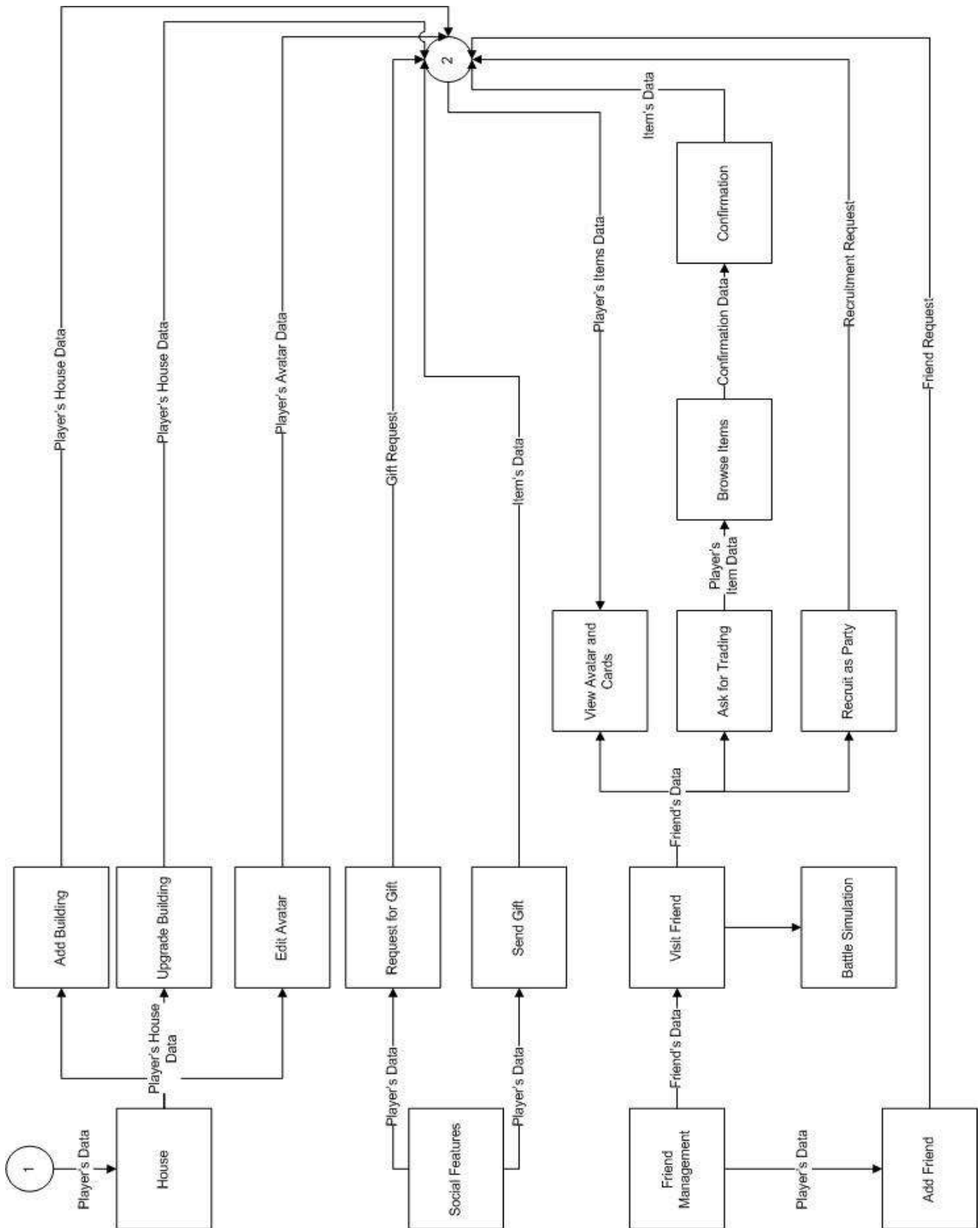
Bahasa *query* dasar yang digunakan untuk mengakses basis data MySQL yaitu SQL (*Structured Query Language*). SQL pertama kali diterapkan pada sebuah proyek riset pada laboratorium riset San Jose, IBM yang bernama sistem R. Saat ini versi MySQL yang terbaru adalah versi 5.6.11.

9. RINGKASAN ISI TUGAS AKHIR

Card Warlock Saga adalah aplikasi permainan *Turn-Based Collectible Card*, yaitu permainan bergenre RPG yang menggunakan kartu sebagai mekanisme pertarungan permainan. Pemain memiliki satu *hero* yang disebut *magician* dan set kartu sebagai mekanisme pertarungan. Keseluruhan sistem pada permainan ini digambarkan dengan *block diagram* pada Gambar 1 dan Gambar 2.



Gambar 1. Block diagram modul battle dan modul shopping pada permainan Card Warlock Saga



Gambar 2. Block diagram modul social pada permainan Card Warlock Saga

Gambar 1 dan Gambar 2 menunjukkan gambaran keseluruhan sistem permainan *Card Warlock Saga*. Pada *block diagram* ini terdapat beberapa modul yang dikategorikan menjadi 3 modul utama, yaitu modul *battle*, modul *shopping*, dan modul *social*. Ketiga modul tersebut mempunyai fungsi masing-masing, yang nantinya akan menuju ke *web service*. Modul *web service* yang akan diimplementasikan ditandai dengan kotak berwarna merah.

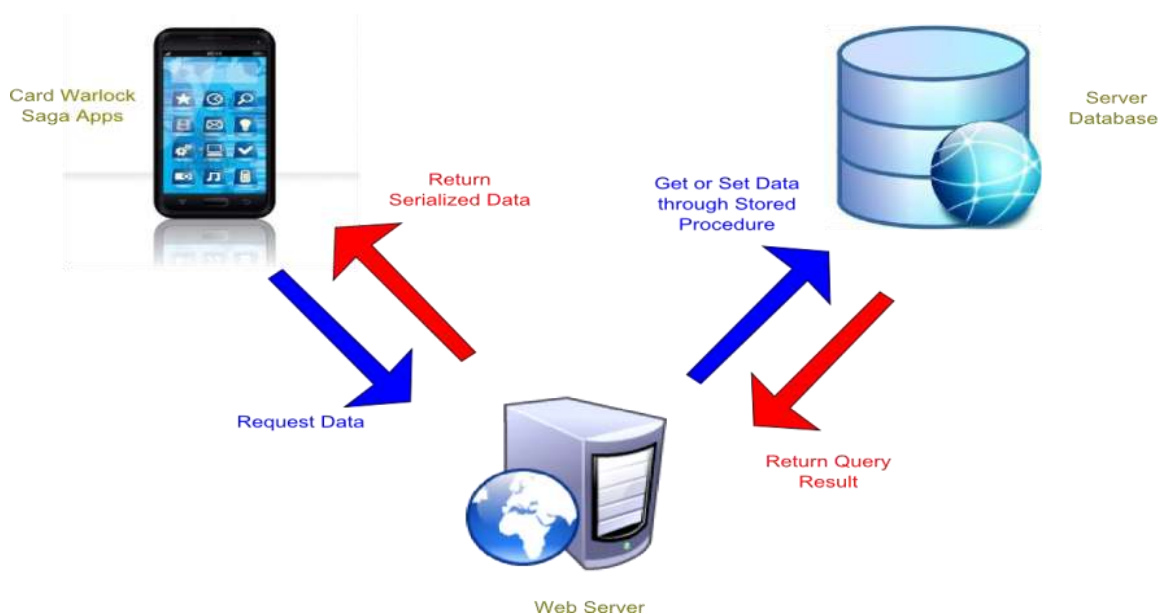
Pemain dapat mengunjungi teman pada menu *visit friend*, di mana pemain dapat mengakses beberapa fitur sosial seperti *trading card*, *VS battle*, *trading items*, *invitation request*, dan *send gift*.

Permainan ini merupakan gabungan dari strategi, RPG, dan *card battle*. Sehingga, dengan penggabungan tersebut memberikan suatu sensasi baru dalam permainan, di mana selain dapat mengeksplorasi dunia *game*, pemain juga harus mengatur strategi kartu apa saja yang dimasukkan ke dalam *deck* yang dipakai untuk bertarung.

Seperti unsur game RPG lainnya, permainan ini tidak memiliki kondisi tamat. Dan agar tidak bosan, permainan ini dibekali dengan *avatar*, kartu, *dungeon*, dan *monster-monster* yang beragam. Selain itu ditambahkan unsur sosial agar pemain dapat berinteraksi dengan pemain lain seperti *trading card*, *battle*, *invitation request*, serta *send gift*.

Permainan sosial *Card Warlock Saga* dibangun dengan menggunakan teknologi Unity 2D dan menggunakan sistem manajemen basis data pada bagian *server* guna menyimpan data-data yang ada, seperti data kartu dan pemain, transaksi *trading card* serta transaksi pembelian kartu di *shop*. Setiap proses transaksi data dari klien ke server dan sebaliknya, akan diarahkan ke *web service* terlebih dahulu. *Web service* ini dibangun dengan menggunakan bahasa pemrograman PHP.

9.1 Arsitektur Sistem



Gambar 3. Arsitektur *web service* pada sistem permainan sosial *Card Warlock Saga*

Gambar 3 menggambarkan tentang arsitektur *web service* pada sistem permainan sosial *Card Warlock Saga*. Pengguna mengakses permainan *Card Warlock Saga* pada perangkat bergerak pengguna, dan untuk transaksi data yang akan dilakukan oleh pengguna menuju *server database*, akan diarahkan ke *web service* yang kemudian akan diproses dan diteruskan ke *database*.

Pertama, pengguna akan *me-request* data melalui *HTTP Request* kepada *web service* disertai dengan mengirimkan parameter yang dibutuhkan oleh *web service*. Kemudian *web service* akan menggunakan parameter yang telah diterima untuk mengakses fungsi yang berupa *stored procedure*. Fungsi inilah yang menghubungkan antara *web service* dengan *database*. Setelah fungsi tersebut dieksekusi, *database* akan mengembalikan data yang diminta kepada *web service*.

Supaya data yang ada pada *web service* dapat di-*transport* kepada pengguna, perlu adanya serialisasi struktur data. Pada *web service* ini, struktur data yang ada akan diserialisasikan menjadi bentuk XML. Data yang telah diubah menjadi *serialized data* tersebut akan dikirim kembali kepada pengguna. Perangkat bergerak pengguna akan merubah kembali data yang telah diterima ke dalam struktur data yang sebenarnya dan kemudian data tersebut dapat dikonsumsi.

9.2 Fungsi–Fungsi pada Card Warlock Saga

Pada sub bab 9.1, telah dijelaskan tentang arsitektur *web service* pada sistem permainan sosial *Card Warlock Saga* ini. Untuk dapat mengakses fungsi–fungsi yang ada pada *web service*, pengguna harus mengirimkan parameter yang dibutuhkan untuk setiap proses transaksi data yang dilakukan. Untuk setiap proses transaksi yang ada pada permainan ini, akan dibuatkan *stored procedure* dari masing–masing fungsi. Maka dari itu, berikut diuraikan tentang fungsi–fungsi utama yang ada pada permainan *Card Warlock Saga* ini.

9.2.1 Fungsi Sosial

Fungsi ini berhubungan dengan jalannya transaksi sosial antar pemain, seperti pertukaran kartu atau kosmetik *item*, *request gift*, *send gift*, dan *add friend*. Data utama yang diperlukan untuk jalannya transaksi ini adalah data kode masing–masing *player* dan kode *card* atau *item* yang dipertukarkan. Fungsi ini kemudian akan mengembalikan data berupa konfirmasi bahwa transaksi berhasil atau tidak kepada pengguna.

9.2.2 Fungsi Item Shopping

Fungsi ini berhubungan dengan jalannya transaksi pembelian *item* dari *shop*. Di mana pengguna dapat membeli *item* yang dapat berupa kosmetik untuk *avatar* pemain atau pembelian kartu baru. Data utama yang diperlukan untuk jalannya transaksi ini adalah data kode *item* yang dibeli serta kode *player* yang membelinya. Fungsi ini kemudian

akan mengembalikan data berupa konfirmasi bahwa transaksi berhasil atau tidak kepada pengguna.

9.2.3 Fungsi Battle

Fungsi ini berhubungan dengan jalannya *battle*. Baik *battle* antara *player* dengan kecerdasan buatan, maupun *battle* antara *player* satu dengan *player* yang lain. Fungsi *battle* memiliki beberapa sub fungsi seperti *show player ranking* dan *show player battle record*. Data utama yang dijadikan parameter pada fungsi ini adalah data kode *player*. Fungsi ini akan mengembalikan data hasil *query* yang telah diserialisasi kepada pengguna.

9.2.4 Fungsi Aturan Main

Fungsi ini merupakan fungsi tambahan yang berisi tentang aturan-aturan main pada permainan dan batasan-batasan bagi pemain, seperti batasan aksi yang dapat dilakukan, batasan energi yang dapat dimiliki oleh pemain, pembatasan pembelian *item* apabila tidak mempunyai cukup *gold*, dan peraturan pengaturan *deck* kartu.

Setiap fungsi dapat mempunyai satu atau lebih jenis dari empat jenis *query* yang ada. Berikut adalah jenis *query* yang ada.

- *Create*, yaitu penambahan data.
- *Read*, yaitu dan pembacaan data.
- *Update*, yaitu pengubahan data.
- *Delete*, yaitu penghapusan data.

Pada Tabel 1 diuraikan secara lebih rinci tentang fungsi-fungsi yang ada pada permainan *Card Warlock Saga*.

Tabel 1. Daftar fungsi utama pada permainan *Card Warlock Saga*

Fungsi	Deskripsi Fungsi	<i>Return Value</i>
<i>CreateHouse()</i>	Pemain dapat membuat <i>house</i> baru.	Keterangan berhasil tidaknya proses pembuatan <i>house</i> .
<i>ConfirmBuilding()</i>	Konfirmasi pembuatan dan peng- <i>upgrade</i> an bangunan pada <i>house</i> .	Konfirmasi Ya/Tidak kepada pemain.

<i>UpgradeHouse()</i>	Meng- <i>upgrade house</i> yang sudah ada.	Keterangan berhasil tidaknya proses <i>upgrade</i> .
<i>ViewHouse()</i>	Menampilkan data <i>house</i> pemain.	Data <i>house</i> yang dimiliki pemain.
<i>SendBattleResult()</i>	Mengirimkan hasil <i>battle</i> pemain, dan meng- <i>update</i> status pemain.	Keterangan berhasil tidaknya proses <i>update</i> data pemain.
<i>ItemReceived()</i>	Mengirimkan data <i>item</i> yang didapatkan oleh pemain setelah <i>battle</i> .	Data <i>item</i> yang didapatkan oleh pemain.
<i>ViewBattleRecord()</i>	Menampilkan data <i>battle history</i> pemain.	Data <i>battle history</i> pemain.
<i>ViewPlayerRank()</i>	Menampilkan <i>ranking</i> seluruh pemain yang ada.	Data <i>ranking</i> pemain.
<i>GetItemList()</i>	Menampilkan data <i>item</i> yang ada pada <i>shop</i> .	Data <i>item</i> .
<i>BuyItem()</i>	Membeli <i>item</i> dari <i>shop</i> .	Keterangan berhasil tidaknya transaksi pembelian.
<i>SellItem()</i>	Menjual <i>item</i> .	Keterangan berhasil tidaknya transaksi penjualan.
<i>ConfirmShopping()</i>	Konfirmasi pembelian atau penjualan <i>item</i> .	Konfirmasi Ya/Tidak kepada pemain.
<i>FindItem()</i>	Mencari <i>item</i> yang diinginkan pada <i>shop</i> .	Data <i>item</i> yang diinginkan.

<i>ViewPlayerItems()</i>	Menampilkan data <i>item</i> yang dimiliki oleh pemain.	Data <i>item</i> pemain.
<i>ViewPlayerProfile()</i>	Menampilkan data seorang pemain.	Data <i>player</i> .
<i>SearchPlayerItem()</i>	Mencari <i>item</i> yang dimiliki oleh pemain.	Data <i>item</i> yang diinginkan.
<i>TradeItems()</i>	Menukar <i>item</i> seorang pemain dengan <i>item</i> pemain lain.	Keterangan berhasil tidaknya proses pertukaran.
<i>ConfirmTrade()</i>	Konfirmasi penukaran <i>item</i> .	Konfirmasi Ya/Tidak kepada pemain.
<i>FindFriend()</i>	Mencari data seorang pemain.	Data <i>player</i> .
<i>AddFriend()</i>	Menambahkan teman ke daftar teman pemain.	Keterangan berhasil tidaknya proses penambahan teman.
<i>RemoveFriend()</i>	Menghapus teman dari daftar teman pemain.	Keterangan berhasil tidaknya proses penghapusan teman.
<i>RecruitFriend()</i>	Mengundang pemain lain untuk masuk sebagai anggota dalam mode <i>quest</i> .	Keterangan berhasil tidaknya penambahan teman ke daftar anggota <i>party</i> pemain.
<i>ConfirmFriend()</i>	Konfirmasi penambahan, penghapusan teman, dan perekrutan pemain lain ke dalam <i>party</i> pemain.	Konfirmasi Ya/Tidak kepada pemain.

<i>RequestGift()</i>	Mengiriman permintaan <i>gift</i> kepada pemain lain.	Keterangan berhasil tidaknya pengiriman <i>request</i> .
<i>RequestEnergy()</i>	Mengiriman permintaan <i>energy</i> kepada pemain lain.	Keterangan berhasil tidaknya pengiriman <i>request</i> .
<i>AcceptRequest()</i>	Menerima <i>request</i> dari pemain lain.	Keterangan berhasil tidaknya penerimaan <i>request</i> .
<i>SendGift()</i>	Mengirimkan <i>item</i> kepada pemain lain sebagai hadiah.	Data <i>item</i> .
<i>ConfirmRequest()</i>	Konfirmasi pengiriman dan penerimaan <i>request</i> .	Konfirmasi Ya/Tidak kepada pemain.

10. METODOLOGI

a. Penyusunan proposal tugas akhir

Tahap awal untuk memulai pengerjaan tugas akhir ini adalah penyusunan proposal tugas akhir. Proposal tugas akhir ini berisi tentang deskripsi pendahuluan dari tugas akhir yang akan dibuat. Pendahuluan ini terdiri atas hal yang menjadi latar belakang diajukannya usulan tugas akhir, rumusan masalah yang diangkat, batasan masalah untuk tugas akhir, tujuan dari pembuatan tugas akhir, manfaat dari hasil pembuatan tugas akhir dan ringkasan. Selain itu dijabarkan pula tinjauan pustaka yang digunakan sebagai referensi pendukung tugas akhir mengenai pengembangan *web service* pada permainan sosial *Card Warlock Saga*.

b. Studi Literatur

Pada studi literatur ini, akan dilakukan pengumpulan informasi dan referensi yang diperlukan dalam proses perancangan dan implementasi aplikasi yang akan dibangun, yaitu mengenai proses serialisasi data dalam bentuk XML, serta pengintegrasian antara Unity dengan *web service*.

c. Analisis dan desain perangkat lunak

Pada tahap ini dilakukan analisa awal dan pendefinisian kebutuhan sistem untuk mengetahui permasalahan yang sedang dihadapi. Selanjutnya, dirumuskan rancangan sistem yang dapat memberikan solusi terhadap permasalahan tersebut. Langkah yang akan dilakukan pada tahap ini adalah sebagai berikut.

1. Perancangan dan pemodelan basis data dalam bentuk model konseptual dan fisik pada permainan sosial *Card Warlock Saga*.
2. Perancangan *stored procedure* basis data pada permainan sosial *Card Warlock Saga*.
3. Perancangan integrasi antara Unity dengan *web service*.
4. Analisis kebutuhan non fungsional.

d. Implementasi perangkat lunak

Pada tahap ini desain perangkat lunak diwujudkan dalam bentuk kode program. Tahap ini merealisasikan apa yang terdapat pada tahapan sebelumnya menjadi sebuah aplikasi yang sesuai dengan apa yang telah direncanakan, di antaranya sebagai berikut.

1. Implementasi pembuatan basis data dengan menggunakan sistem manajemen basis data relasional MySQL.
2. Implementasi pembuatan *stored procedure* dengan menggunakan bantuan kakas SQLyog.
3. Implementasi pengintegrasian antara Unity dengan *web service* yang dibangun dengan menggunakan bahasa pemrograman PHP. Kakas bantu yang digunakan adalah NetBeans IDE 7.2.

e. Pengujian dan evaluasi

Pada tahap ini dilakukan pengujian perangkat lunak yang bertujuan untuk menemukan kesalahan-kesalahan dalam pengembangan aplikasi secara langsung pada perangkat bergerak Android agar dapat diperbaiki sesegera mungkin.

Pengujian akan dilakukan pada masing-masing fungsi yang ada pada *web service*. Pengujian akan dilakukan secara *white box* dan *black box testing*. Adapun poin-poin pengujian yang akan dilakukan adalah sebagai berikut.

- *Correctness*

Pengujian ini merupakan uji coba beberapa skenario pada fungsi-fungsi pada *web service* untuk menguji apakah *output* yang dikeluarkan oleh sistem sudah sesuai dengan *input* yang diberikan, dan sesuai dengan kebutuhan.

- *Integrity*
Pengujian ini bertujuan untuk menguji beberapa fungsi yang memiliki hak akses yang berbeda-beda seperti *read*, *insert*, *update*, dan *delete*. Uji coba dilakukan untuk memastikan bahwa setiap fungsi telah dibatasi pada hak akses yang telah ditentukan.
- *Reusability*
Pengujian terhadap beberapa fungsi sehingga dapat digunakan dan diakses kembali oleh fungsi-fungsi dari modul lain.
- *Efficiency*
Pengujian *performance* sistem untuk mengetahui apakah fungsi yang telah dibangun dapat bekerja sesuai dengan kebutuhan serta dapat bekerja dengan *delay* kurang dari 2 *second* apabila diakses secara *massive*.

f. Penyusunan Buku Tugas Akhir

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam tugas akhir ini serta hasil dari implementasi aplikasi perangkat lunak yang telah dibuat. Sistematika penulisan buku tugas akhir secara garis besar antara lain sebagai berikut.

1. Pendahuluan
 - a. Latar Belakang
 - b. Rumusan Masalah
 - c. Batasan Tugas Akhir
 - d. Tujuan
 - e. Metodologi
 - f. Sistematika Penulisan
2. Tinjauan Pustaka
3. Desain dan Implementasi
4. Pengujian dan Evaluasi
5. Kesimpulan dan Saran
6. Daftar Pustaka

11. JADWAL KEGIATAN

Pengerjaan tugas akhir akan dijadwalkan pada Tabel 2.

Tabel 2. Jadwal pengerjaan tugas akhir

Tahapan	Tahun 2014																							
	Februari				Maret				April				Mei				Juni							
Penyusunan Proposal																								
Studi Literatur																								
Perancangan sistem																								
Implementasi																								
Pengujian dan evaluasi																								
Penyusunan buku																								

12. DAFTAR PUSTAKA

- [1] Unity, "Unity3D," 2013. [Online]. Available: <http://unity3d.com/pages/create-games>. [Diakses 18 Februari 2014].
- [2] N. Lovell, "Games Brief," Januari 2011. [Online]. Available: <http://www.gamesbrief.com/2011/01/what-is-a-social-game>. [Diakses 3 Maret 2014].
- [3] J.Radoff, Game On : Energize Your Business With Social Media, Wiley, 2011.
- [4] MySQL Developer Team, [Online]. Available: <http://dev.mysql.com/doc/refman/5.1/en/create-procedure.html>. [Diakses 3 Maret 2014].
- [5] S.Rifa'i, "Belajar Website," 6 Februari 2013. [Online]. Available: <http://upyes.wordpress.com/2013/02/06/pengertian-dan-sejarah-mysql>. [Diakses 3 Maret 2014].