**Programming Languages & Translators**

**Project WhitePage**

**The "Game Wizzard"**

——Online Role Playing Card Games design language

Team Member: Liyuan Zheng(lz2375)

Chaozhong Lian(cl3190)

Yue Huang(yh2640)

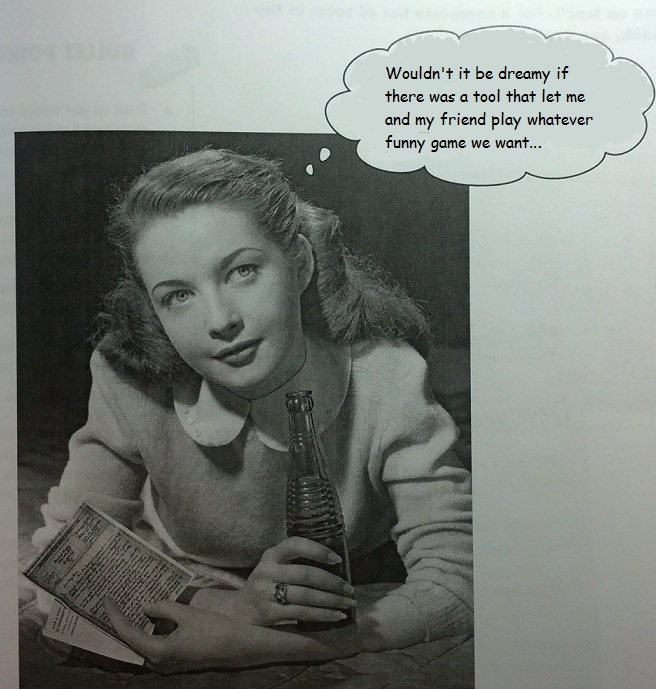
Ke Liao(kl2735)

Songqiao Su(ss4555)



1. Yes! Playing any game you want with your friends on-line!

One may recall the painful moment when not being able to playing card grams with your friend. All the people on earth are complaining that getting a group of friends sitting together in the same place at the same time is increasingly difficult.



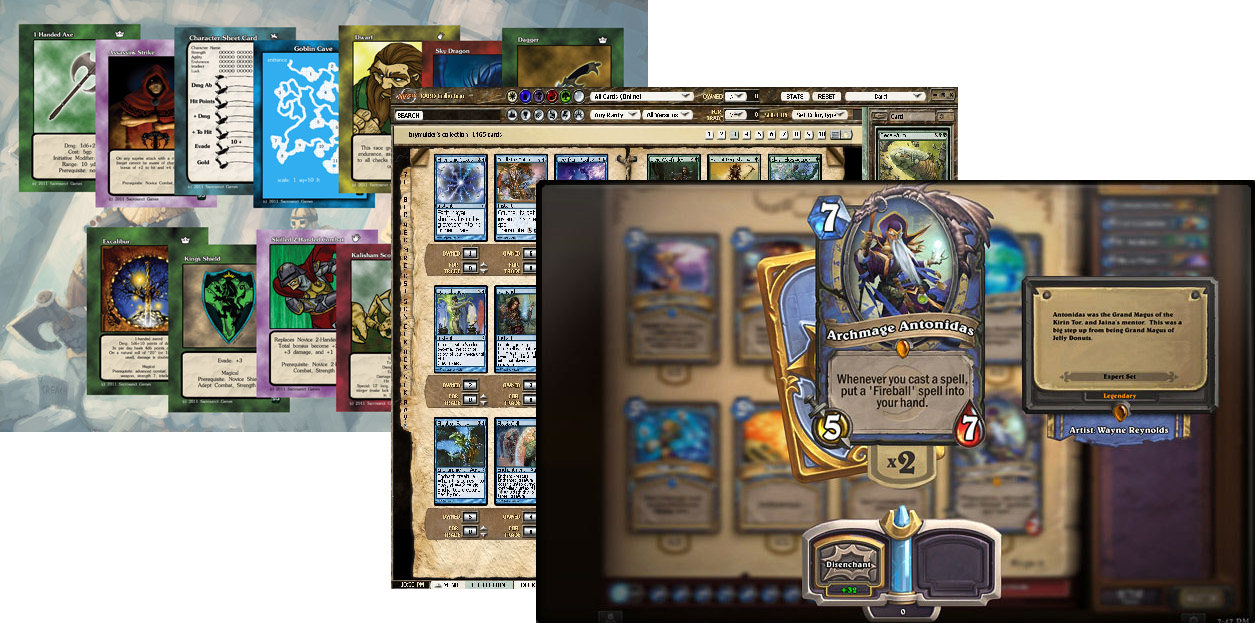
And here is the savior of the day! The idea of GameWizard comes from the fact that we are all experiencing such a disappointment when we could not start some kind of really enjoyable games with friends, simply because the lacking of certain playing pieces. With GameWizard, people can now easily start a tabletop game, whether the game is uncommon or not, without being limited by playing pieces, or being limited by cannot getting everyone at the same place. It is internally designed with an on-line platform to support remote interactions by different players from different terminals.

With the help of wizard, all that we need is only some game rules and some leisure time. You can define whatever role playing card games you and your friend are eager to play together, or, if not a big fans of role playing card games, our language is compatible for writing normal card games(poker games, Uno, etc...)!

GameWizard abstract out the common patterns of role-playing card games(such as Magic The Gathering, HeartStone, etc...), and so we simply our language targeted at this type of games, making the language shorter and most efficient than simply writing java or python. Using GameWizard to define new games would be a simple and interesting process, so why not be creative?

1. Why our language is so different?

* Define any RPG CARD GAMES With less lines of code

 Our language, GameWizard, aims to provide an easy way to design role playing card games(such as Magic The Gathering, HeartStone, etc...). By abstracting out the common operation, attributes, and state, we simplify our language so that programmer can write much lesser code than writing in Java, not to mention C/C++. Our language also simplifies the initialization of List, HashTable like container, further simplifying the code. For example, our language is internally design with keyword like shuffle or dealing card, shortening the lines of code needed to write a rpg card games.

Every rpg card games has three parses, the starting parse(for example, initializing the count of "blood" and "mana" for each users), the middle parse(this is the parse when every user performs some actions in turn, using some card, invoking some effects, etc.), the ending parse(deciding which player is the winner). Also, every player has some abilities(be it releasing monsters or creating fire) or may have some magical items(something like magical ruby or magic wand ). Our programming language provides an easy interfaces for users to easily define their rpg card games.

* Compiles, and yes, now play on-line!

Once the game is written, all the user need to do is simply compiling the source code, after that, a server program and a client program will be generated immediately. The user now only need to distribute the client side program to his/her friends, and then once the user himself/herself runs the server program, all his/her friends can now connect to him/her. And whoa la, the game works like a charm!

1. For what kind of users are we designing this language?

The kind of users that we are designing this language for is the kind of user who has some basic computer programming experience, knowing how to use for loop, if statement to represents logic. And as we have mentioned previously, because our language is fully targeted at writing rpg card games, we optimize many keywords and syntax intended to specific this type of games, which makes the line of code needed to write becomes much lesser than simply writing Java code. Besides, our program internally builds the server program and the client program, so no needs for Computer Networking knowledge needed for the user to write an on-line multi-player game, all the programmers care about is to defining the players and cards, as well as their associated logic.