University of Hong Kong

CSIS0801 Final Year Project
Customizable Problem Solving Game
(FYP14002)

Project Plan

Supervisor: Dr. Vincent Lau
Student: Kwok Chun Hin
UID: 2012555370
http://i.cs.hku.hk/fyp/2014/fyp14002/
# Table of Contents

Summary .................................................................................................................................................. 3
Background ............................................................................................................................................... 3
Objectives ................................................................................................................................................. 4
  Cross Platform ....................................................................................................................................... 4
  Infinity Levels ....................................................................................................................................... 4
Game Design ............................................................................................................................................ 5
  Layout .................................................................................................................................................. 5
  Game Rule ........................................................................................................................................... 5
  Level Editor ......................................................................................................................................... 7
  Sharing ................................................................................................................................................ 9
    Through our Database ...................................................................................................................... 9
    Through QR Code and Social Network .............................................................................................. 9
Scope ....................................................................................................................................................... 10
Prerequisites .......................................................................................................................................... 11
  Hardware ............................................................................................................................................ 11
    Server ............................................................................................................................................. 11
    Computer ........................................................................................................................................ 11
    Mobile Devices ............................................................................................................................... 11
  Software ............................................................................................................................................. 11
    Unity3D ........................................................................................................................................... 11
    Xcode ............................................................................................................................................... 11
    Adobe Photoshop ............................................................................................................................ 11
    Blender ............................................................................................................................................ 11
  Techniques ......................................................................................................................................... 11
Deliverables ........................................................................................................................................... 11
Approach ................................................................................................................................................ 12
Methodology .......................................................................................................................................... 13
  Client Side ......................................................................................................................................... 13
  Server Side ......................................................................................................................................... 13
Risk Assessment ..................................................................................................................................... 13
Schedule ............................................................................................................................................... 14
  Major Milestones .............................................................................................................................. 14
Budget ..................................................................................................................................................... 16
Conclusion .............................................................................................................................................. 16
Summary

There are many problem-solving games available in the market, but most of them have a limited number of levels in each game (Table 1). As a result, you will have nothing to play when you completed all the levels and you want to continue.

What are the reasons? Developer is the only one who can develop the levels, which is a small group of people. As a result, the number of levels is limited to their creativity and the effort they spent.

We decided to develop a Problem Solving Game – Swipe Solving, which is a cross platform game with infinity levels.

We want our players to contribute to the game. Level Editor is included for players to design levels on their own, which allow them to decide where to place a ground, ball and destination. When completed, they can share their levels to public. Other players can download and play.

Everyone can contribute to the game and our game can keep on providing new levels to players.

<table>
<thead>
<tr>
<th>Game</th>
<th>Number of Levels</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cut The Rope</td>
<td>400 Levels</td>
</tr>
<tr>
<td>Where’s My Water</td>
<td>567 Levels</td>
</tr>
<tr>
<td>Drop the Chicken</td>
<td>80 Levels</td>
</tr>
</tbody>
</table>

Table 1 Number of levels in some popular problem-solving games

Background

According to research, humans’ processing speed and memory deteriorate starting from 20s\(^1\). While it cannot be stopped, we can slow it down. As it is not a disease and it is incurable, prevention is much more important.

Cognitive Training is the key to prevention\(^2\), so we want to develop a problem solving game that train players’ problem solving skills.

Everyone will get old, with a decrease of processing speed, and it could not be stopped. We should start brain training as early as possible.


\(^2\)CNNHealth[Internet]:Strength training key in preventing Alzheimer’s. Available from: http://thechart.blogs.cnn.com/2012/07/16/strength-training-key-in-preventing-alzheimers/
Many problem-solving games are available in the market. How to make a game success among this competition is a big challenge. We try to find out what makes the game like "Cut The Rope", "Where's My Water" and "Drop the Chicken" success.

After some researches and observation, it is found that they have some common characteristics. First, they use an adorable character with smooth animations, which can attract the children (Figure 1). Second, they have a simple game rules. This helps players to learn the game fast. Third, they start with an interesting story or opening, which enable players to understand the goal of the game.

However, most of them have limited number of levels, which affect the "life" of the game. The developers have to design new levels regularly to offer new challenges to players. This made the maintenance be a tough job. The more levels they created, the more difficult it is to design new levels since they cannot repeat the old levels.

![Figure 1 Most of the successful game have a cute character](image)

### Objectives

**Cross Platform**
The game support IOS, Android and Web, which are the most popular platforms nowadays. These platforms have covered mobile and computer, so our target users is much more.

**Infinity Levels**
Level Editor is included to allow players to design their own levels and share with others. Players are able to upload their own levels to our server for sharing. And we will keep these levels in our database and allow other players to download. Player can never complete all the levels as it increases every day.
Game Design

Layout
In each level, there are some balls (in range 1 to 4) and the same number of destinations with the matching color on the screen. Horizontal and vertical walls are placed to block the way.

Game Rule
Players can move the balls in the direction of up, down, left and right by swipe gesture.
By one swipe, all the balls on the screen will move by one unit in the same direction (Figure 4).

If all the balls reach the destination with the matching color, players win.
If any of the balls falls out of the world, players lose.
Figure 4: Show how the balls move.
Level Editor

Players can design their own levels with our editor.

Before sharing, our algorithm would try to find solutions of the levels. If no solution is found, we would not allow it to be released. If the player is sure that the level can be solved, he has an option to try it. And we would keep a record of his solution in our database for debugging.
Customizable Problem Solving Game

Figure 7 Begin with Empty Level

Figure 8 Place items on it

Figure 9 Export as JSON
Sharing

The levels created in Level Editor can be exported to JSON format. There are two ways of sharing a level.

Through our Database

Designers: Players can upload their level to our server by clicking a button in Level Editor.
Players: Players can browse the list of levels in game. All the levels are shown like a gallery. They can download the levels they want to play by onedick.

Through QR Code and Social Network

Designers: Level Editor is able to generate a QR code from the level settings.
Players: Players can share this QR code to anywhere, such as Facebook, Whatsapp, Forum, etc.
Players: Players can scan the QR code to get the level setting and start the game.

Figure 10 Screen to download Levels in game
Scope

The project will include the following:

- Problem Solving Game
- Level Editor
- Function that upload levels to Server
- Function that download levels from Server
- Algorithm that solve the levels
- Setup Server
- Database design
- Layout Design

The project does not include the following:

- QR code generation
Prerequisites

Hardware

Server
We need a server with good network connection to provide communication between players and our server. The server needs to receive “levels” from players and store them in database. Players can also download “levels” from our server and store in their devices.

Computer
We need a computer with Unity3D installed for game development.

Mobile Devices
Our game will support IOS, Android and Web. So we need an iPhone and an Android mobile for testing purpose.

Software

Unity3D
The software we used to develop the game. It can build the project for different platforms, such as IOS, Android, and Web.

Xcode
For deploying the project on IOS, we must use the Xcode to build and run the project.

Adobe Photoshop
Many 2D graphics is needed. We will use Photoshop to create new graphics and modify existing graphics.

Blender
Tools to design 3D Model. We will try to design the 3D characters.

Techniques

- C# Programming
- PHP Programming
- Database design
- Knowledge of server connection

Deliverables

At the end, we will complete a problem solving game, which can be played on IOS, Android and Web. It will be available on the App Store, Play Store and our website. Level Editor is included to allow players to design their own levels and share to public through our Server.
Approach

We will design the game play first. Then we will design an algorithm to find solutions of the “level”, so that we can verify if the level can be solved and provide hints to player dynamically.

After that, we will develop our first prototype for testing purpose. In the next step, we will try to setup a private server for testing and include the upload and download functions to the game.

After implement all the basic functions, we will add new features, such as special items, collectables, and special effect.

Phase 1: Design Gameplay, rules and algorithm that solve the “level”
Phase 2: Develop Prototype (Offline Version)
Phase 3: Config Server and Design Database
Phase 4: Develop Prototype 2 that connect with Server
Phase 5: Add-On Features
Phase 6: Design Graphic
Methodology

Client Side
We will use C# as the main programming language. We plan to design an algorithm base on Breadth-First Search to solve the levels, hence to provide hints to players. Players can design their own levels. When they upload their levels to our server, the “level” is converted to JSON format and sent to our server.

Server Side
We will use PHP as the main programming language and MYSQL as the database. Server is responsible for handling requests from clients, such as download levels and upload levels.

Risk Assessment
There is a chance that server cannot handle the requests if we have a heavy load. We may have to reduce the connection to database. Algorithm may be complicated if special items are added in future. It may be take a long time for our algorithm to find the solutions.
## Schedule

### Major Milestones

1. **Design and Preparation (End of September)**  
   Design the game play, rules and algorithm related to game. All these will be used in next phased.

2. **Development of Client Side (Mid of October)**  
   Develop the first prototype for the client side. When completed, a problem-solving game will be available, with Level Editor which allow players to design levels and play locally.

3. **Development of Server Side (Early November)**  
   Setup the server and design the database for storage. Write program to handle request from client side.

4. **Networking (Mid of December)**  
   Enable connection between client and server side. Players should be able to upload their levels to our server, and download others’ levels from our server.

5. **Add-On Features (End of February)**  
   Design special items that make the game more challenging. Implement special items.

6. **Graphics Design (End of March)**  
   Design the User Interface and graphics for the final build.
### Phase 1 – Design and Preparation

<table>
<thead>
<tr>
<th>Begin</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>2014/09/01</td>
<td>Design Game Play</td>
</tr>
<tr>
<td>2014/09/05</td>
<td>Project Plan</td>
</tr>
<tr>
<td>2014/09/12</td>
<td>Project Website</td>
</tr>
<tr>
<td>2014/09/15</td>
<td>Design algorithm for solving the game</td>
</tr>
</tbody>
</table>

### Phase 2 – Development of Client Side

<table>
<thead>
<tr>
<th>Begin</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>2014/09/25</td>
<td>Develop Game Prototype</td>
</tr>
</tbody>
</table>

### Phase 3 – Development of Server Side

<table>
<thead>
<tr>
<th>Begin</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>2014/10/20</td>
<td>Config Server</td>
</tr>
<tr>
<td>2014/10/22</td>
<td>Design Database</td>
</tr>
<tr>
<td>2014/10/30</td>
<td>Develop Server Side Programming</td>
</tr>
</tbody>
</table>

### Phase 4 – Networking

<table>
<thead>
<tr>
<th>Begin</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>2014/11/7</td>
<td>Develop Game Prototype 2</td>
</tr>
</tbody>
</table>

### Phase 5 – Add-On Features

<table>
<thead>
<tr>
<th>Begin</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>2014/12/17</td>
<td>Design special items</td>
</tr>
<tr>
<td>2015/01/10</td>
<td>Implement Special Items</td>
</tr>
</tbody>
</table>

### Phase 6 – Graphics Design

<table>
<thead>
<tr>
<th>Begin</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>2015/03/01</td>
<td>Design the User Interface and graphics</td>
</tr>
</tbody>
</table>
**Budget**

<table>
<thead>
<tr>
<th>Items</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>3D Models</td>
<td>$500</td>
</tr>
<tr>
<td>Script for QR Code</td>
<td>$200</td>
</tr>
<tr>
<td>Sum</td>
<td>$700</td>
</tr>
</tbody>
</table>

**Conclusion**

This is not just a game. It is a game with benefit, which can keep your brain health and smart.

We have designed the game play and the basic algorithm to find the solutions of levels. We are developing the first prototype. Basic game rules are implemented, and Level Editor is in progress.