DM2295 Game Development Project

Academic Year 2014/2015 Semester 2

Final Report

Team Number 11 (The Fluffy Plebs)

Project Title Overtale

Members Gregory Koh (Leader)  
 Almeda Glenn  
 Jacela Job  
 Rayson Poh

**GAME CONCEPTS**

* 1. **Target Audience**
* Ages 3 to 12 years old
  1. **Game Concept**

***Overtale*** is an educational game that requires the player to role-play as a boy that fell into a place called “The Guardian’s Den”. In order to find his way out, he has to play **4 different games** and accumulate **at least** **100 jellybeans** that “The Guardian” is asking for.

* 1. **Game Goals**
* Provide players the experience of resource management and what “Risk and Reward” means
* Immerse the player in a set of 2 fast-paced, timed mini-games to encourage speed thinking
* Encourage objective thinking through the use of colors and questions on the other 2 mini-games
  1. **References**
* Btoom! (Animation, Maze)
* Undertale (Music Theme)
* Stress (Memory Matching Game)
* Primary School Examination Papers (MCQ)
* Magicka! (Colors Matching and Casting)

**Controls and Game Flow**

**2.1 Controls**

***Overtale*** makes use of the following ***basic*** controls:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Control Mapping** |
| Move Up | Move character upwards | Key “W” |
| Move Down | Move character downwards | Key “S” |
| Move Left | Move character leftwards | Key “A” |
| Move Right | Move character rightwards | Key “D” |

Once the player enters into one of the four mini-games, there will be additional controls. Only ***Maze Runner (Game 1)*** does not have any additional controls

**Puzzle Party (Game2)** have the following additional controls:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Control Mapping** |
| Cast Blue | Activate blue color if present | Key “Q” |
| Cast Yellow | Activate yellow color if present | Key “E” |
| Cast Red | Activate red color if present | Key “R” |
| Cast Purple | Activate purple color if present | Key “T” |
| Combine Casted Colors | Combine 2 activated colors | Spacebar |

**MCQ Power (Game 3)** have the following additional controls:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Control Mapping** |
| Pick up/Answer question | Pick up a new answer or answer the question | Key “F” |
| Swap answer | Swap your answer with the nearest alternative if you already picked up one | Key “Q” |

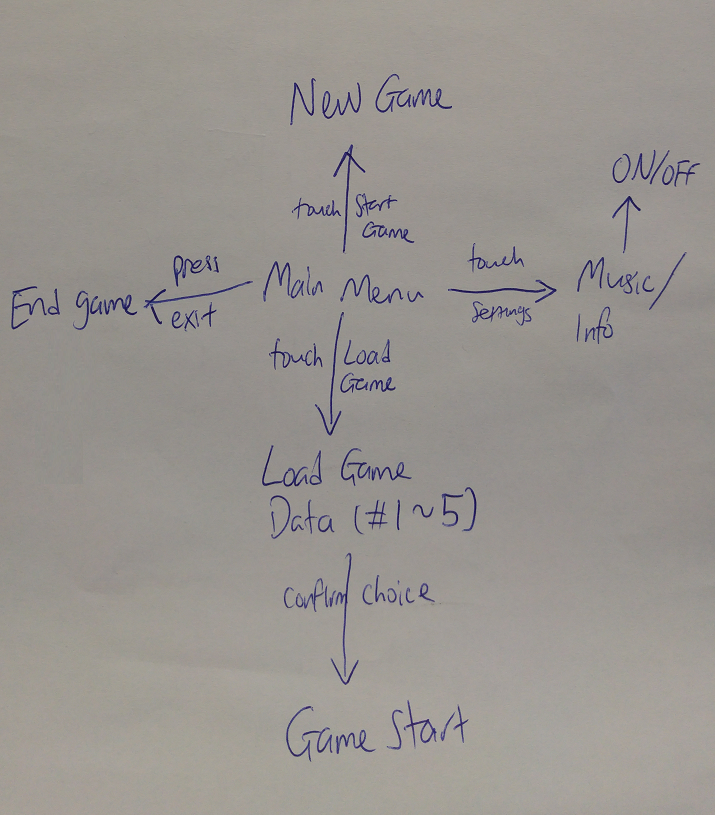
**Stress (Game 4)** have the following additional controls:

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Control Mapping** |
| Pick up card | Pick up the nearest card | Key “F” |
| Place card | Place the card in possession to nearest the tile | Key “F” |
| Push button | Press the “Stress” button to score | Key “F” |

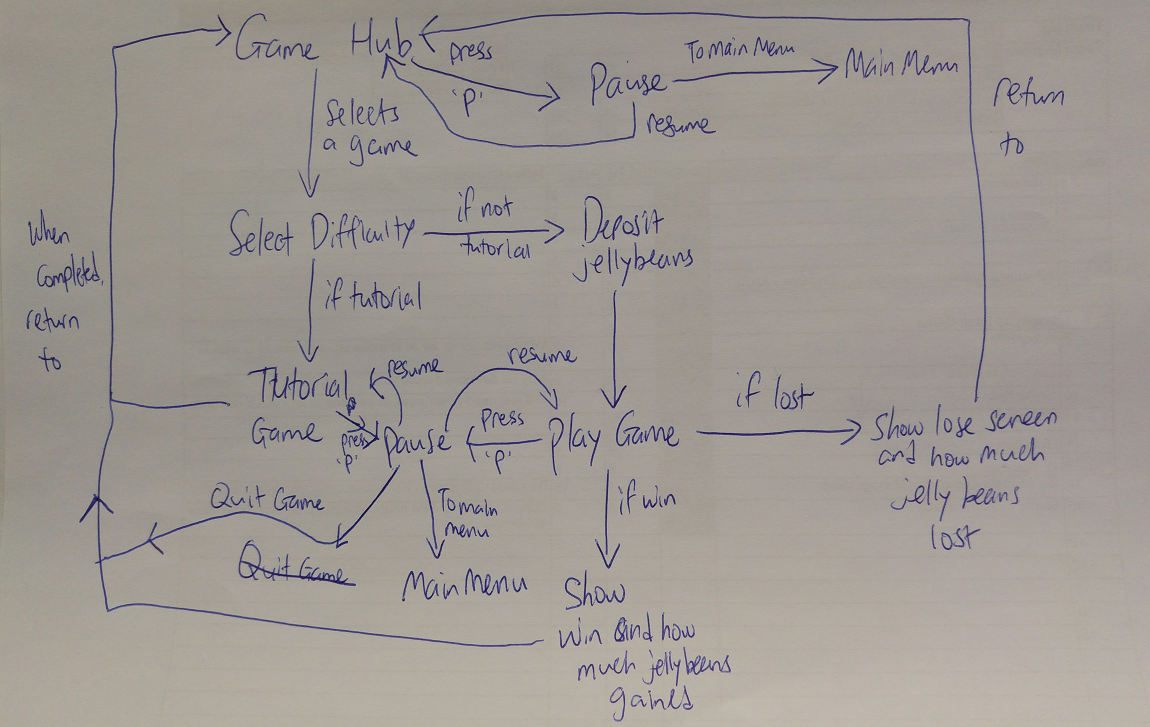
**2.2 Game Flow**

Given the size of the game, the game flow will be split into **two** components: 1) Menu Flow and 2) Game Flow

**Menu Flow**



**Game Flow**



**Game Modes**

**3.1 Overview**

***Overtale*** has 4 different types of games, each provided with a simple **tutorial, easy, medium and hard difficulties**. Every game mode will have a different set of mechanics to follow which will be mentioned upon playing the **tutorial mode**. Players will start with only the **tutorial** mode available to them and will unlock the next difficulty level as they progress through the game. Also, any difficulty after **tutorial** mode will require the player to deposit a set number of **jellybeans**. If he succeeds, he will get back double the amount. However, if he fails, he loses the amount he deposited.

**3.2 Maze Runner (Game 1)**

**Overview**

This mini game is all about finding the exit inside the maze that the player is in. The spawning point and exit point is randomized so no two replays will be the same. There is also a set of obstacles that will attempt to hinder the player’s progress.

**Win condition**

The player has to find the exit before the timer runs out.

**Lose condition**

When the time is up.

**3.3 Puzzle Party (Game 2)**

**Overview**

This mini game requires the player to find different colors scattered across the stage and use them to destroy the bricks blocking his way with the corresponding color. He will have to avoid enemies that will roam around the stage or else lose the possession of the colors.

**Win condition**

The player has to reach the exit (obstructed by the colored bricks).

**Lose condition**

When the player loses all three lives.

**3.4 MCQ Power (Game 3)**

**Overview**

This mini game requires the player to answer a set number of questions. Each question will have three different possible answers. Answering correctly will lead him to the next question, or the exit.

**Win condition**

Answering **all** the questions and finding the final exit.

**Lose condition**

When the player loses all three lives.

**3.5 Stress (Game 4)**

**Overview**

This mini game requires the player to memorize a pattern of 3 different combinations of card colors. He will have to fill up the set with the 3 cards provided and push the “Stress” button to score. If his combination matches the randomized combination, he earns points. If not, nothing happens.

**Win condition**

Getting a score higher than the minimum requirement when the time expires.

**Lose condition**

Getting a score lower than the minimum requirement when the time expires.

**Knowledge Applied, by Subject**

**4.1 Overview**

In this section will have information regarding the features that the team managed to implement over the three weeks.

**4.2 AI**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Done by** |
| Waypoints | Meant for the simple movement of the enemies | Almeda Glenn |
| Probability | To provide randomness | Gregory Koh |
| Finite State Machine | Control the number of states available to the enemy | Job Jacela |

**4.3 Advanced Game Development Techniques**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Done by** |
| Maze Runner | Mini game 1 | Gregory Koh |
| Puzzle Party | Mini game 2 | Job Jacela |
| MCQ Power | Mini game 3 | Almeda Glenn |
| Stress | Mini game 4 | Rayson Poh |
| Load and Save Game (using Lua scripts) | Load and Save progress of player | Gregory Koh |
| Difficulty Panel | Allow the player to choose his difficulty level | Gregory Koh |
| Confirmation Popout Window | Make the player confirm that he chose the previous option | Gregory Koh |
| Player avatar properties | Behavior of player in-game | Gregory Koh |
| Establish lua class | To provide an easy-to-use class for the rest of the group | Job Jacela |
| Color and Door | For use of the games (only Game 2 uses Color) | Job Jacela |
| Win/Lose Screens | Display the outcome at end of mini games | Job Jacela |
| Music Box | To play music in the game | Almeda Glenn |
| Game State and Scene Management | Transition the scenes in between to save memory usage | Almeda Glenn |
| Object class | The base level of objects for use by the rest of the group later | Almeda Glenn |
| Question and Answers | Write questions on lua scripts | Almeda Glenn |
| User Interface System | Improve the aesthetics value of the game | Rayson Poh |

**4.4 Detailed descriptions on some features**

These features are the features that have been given considerable amount of time on the different members.

**4.5 Gregory Koh**

Most notable feature: Saving/Loading from Lua script

Description:

Among the main requirements of the studio projects is to have the ability to save and load the game progress of the player. Gregory initially knew how to load the data from Lua script, but found it a little tricky when it comes to saving the progress. It took him a day to finally figure out how to save the game which was done within the first week.

**4.6 Job Jacela**

Most notable feature: Finite State Machine

Description:

One of the most critical components required for both his game and Gregory’s game mainly because it requires some AI to be effective. The FSM allows the enemies spawned in their games to have some form of intelligence in both games. This feature adds a considerable amount of challenge to Gregory’s game too.

**4.7 Almeda Glenn**

Most notable feature: Game State and Scene Management

Description:

Without some form of management on the game state and scenes, it would be very difficult to manage the sound system and the games itself. It took him nearly two days to complete the set-up and he required advice from Gregory to complete the task. Once the game state management was completed, the development of the various features accelerated considerably.

**4.8 Rayson Poh**

Most notable feature: User Interface System

Description:

Rayson took on a massive task to provide a working framework of User Interface to the rest of the team. The idea behind it was mainly to make the game look good and to do so, he centralizes all the graphics changes in one area. That way, it was easy to use the tool which helped us achieve a level of appeal that would otherwise be very difficult to achieve otherwise.

**Problems encountered/Solved/Lessons Learnt**

**5.1 Problems encountered and how we solved it**

Every team will encounter their fair share of problems. Here are some of the problems we encountered during the studio project.

* Miscommunications often occurred as a result of misinterpreting the message delivered. We usually solve the issue by clarifying what we meant to say as many times as we require so we are on the same page as often as possible.
* A few overambitious features. We initially wanted to have AI-controlled opponent in certain mini games but found that we ran short in time. These features were eventually cut.
* Varying coding standards. It is a given that not everyone in the team have the same standard of coding. Some needed more time to complete their tasks. To maintain the momentum of the project, the more proficient programmers took on the larger and harder tasks. We also exchange ideas and solutions which helps a lot in improving our coding standards.

**5.2 Lessons learnt**

There are a few insights gained from this project work. Here are some of them.

* Get a minimum-quality working game as fast as you can. It is way better to have a bare-minimum game than not to have one. For that, we made sure that our project follows a strict schedule.
* Cutting features is important. Especially those that are very difficult to do. This helps us manage our time better while not getting bogged down by an exceedingly difficult task.
* Communication must be present. Any successful team needs to be able to coordinate well with one another. Without it, it is practically impossible to succeed.
* Following a common game theme. Given that the game have four different games, it requires a careful coordination to ensure that the games have a common theme. Players will then be able to appreciate the reason behind the mini-games.

**6.1 Future improvements**

If the opportunity to improve the prototype game arises, these improvements are proposed to make the game even better.

* Improve the quality of the mini games by loading more unique situations.
* Provide a set of cinematics as the player progresses through the game.
* Make it a commercial PC game.

**Project Schedule**

**7.1 Overview**

This section shows how the team managed their time throughout the three weeks. The information is based on all the commits pushed in Github.

**7.2 Week 1**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Day/Team member** | **Gregory** | **Job** | **Glenn** | **Rayson** |
| Monday | Framework/Player animations | Lua script class | Framework/Music system/Waypoints | UI system (started) |
| Tuesday | Jellybean system/Difficulty class/Lua scripting/Bounding Box/ | Completed lua script class/Game hub scene | Object class/Memory Leak fixes | UI system (in progress) |
| Wednesday | Probability/Enhance player class/Difficulty system | Create sample maps for different games/Debugging game hub scene | Menu Flow/Tutorial Games Base/Soundtrack | UI system (still in progress) |
| Thursday | Load/Save game | Door class/Finished game hub scene (sample) | SceneGameBase (for all game stacks to inherit from)/Tutorial Game 3 | UI system (Debugging phase) |
| Friday | Tutorial Level for Game 1 | FSM AI class | Debug Object class/Tutorial Game 3 | UI Animator class |
| Weekend | Hero inherit from OBJ class/Updated from Game 1 | NIL | Updated Tutorial Game 3 | UI Image class |

**7.3 Week 2**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Day/Team member** | **Gregory** | **Job** | **Glenn** | **Rayson** |
| Monday | Tutorial Game 1 | Tutorial Game 2 | Tutorial Game 3 | Tutorial Game 4 |
| Tuesday | Easy mode Game 1 | Debug Tutorial Game 2/Color class | Update Object class/Question and Answer class | Update UI system/Game 4 |
| Wednesday | Medium and Hard mode Game 1 | AI base class | Completed object class/Easy mode Game 3 | Add game components into Game 4 |
| Thursday | Update Easy and Medium Game 1/Difficulty selection | Easy, Medium and Hard mode Game 2 | Medium mode Game 3 | Deck Class for Game 4 |
| Friday | Jellybean selection at game hub/Unlock levels/Load 5 data types/ | Implement AI into Game 2 | Pause state | Complete Game 4 |
| Weekend | Debug Difficulty selection/ Add confirmation UI/Autosave | Debug Game 2/AI | Debug Game 3 | Debug Game 4 |

**7.3 Week 3**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Day/Team member** | **Gregory** | **Job** | **Glenn** | **Rayson** |
| Monday | AI Idling class/ Game Hub state changes | Update AI class/Game 2 | Update Game 3/Pause Menu | Main Menu UI |
| Tuesday | Added Guardian and Exit/Autosave upon finishing game/NPC/Instructions on Game 1 | Win/Lose triggers | Music on/off icon, GUI improvement on Game 3 | Main Menu UI update |
| Wednesday | Lock and Delete Data/Show jellybeans gained or lost at end of game/Start Functions/Warning fixes Game 1 | GUI improvement on Game 2/Instructions Game 2 | Instructions Game 3/Warning fixes Game 3 | Main Menu settings/Instructions Game 4 |
| Thursday | Warning fixes for entire game | Documentation | Documentation (Game Report)/ | Warning fixes for entire game |

**7.3 Timeline of task breakdown**

**8.1 Screenshots**

○ The game idea/concept. Describe the game.

○ Project schedule and timeline of task breakdown

○ Screenshots

○ Description of features (> 50 words for each feature)

○ Knowledge applied

○ Problems encountered/ problems solved / Lessons learnt

○ Future enhancements/improvements