Carleton University
COMP4905 – Honours Project

MER-A
Designing New Mechanics of an Online Game

Student: Xueli Zhang
Supervisor: Oliver van Kaick
Date: Tuesday, October 24, 2017
Abstract:

The purpose of this project is to develop a new game mechanic for an online multiplayer game and investigate how general players react to the new game mechanic. This report mainly covers: (1) the motivation and goals that the project needs to achieve, which is investigating the fun provided by a new game mechanic for an attacking system with a user case study; (2) methodologies of deployment in the networking system used by the project; (3) the feedback of players after playing the game in early development phase, middle development phase and final development phase; (4) the conclusion of the investigation, which summarizes how fun the new game mechanic is, what improvements need to be done in future work, and how the new game mechanic can be adopted by other games.

Acknowledgments:

During the development process of the project, professor Oliver van Kaick helped me with the concepts of building an online multiplayer game, and this helped me to have a significant clue of where the project should start with. And RuoHao Xu helped me on building a basic game terrain with Unity 3D.
# Table of Contents

0.1 Abstract ......................................................................................................................... 2

0.2 Acknowledgments ......................................................................................................... 2

1. Introduction ....................................................................................................................... 6
   1.1 Purpose ......................................................................................................................... 6
   1.2 Background .................................................................................................................. 6
   1.3 Development Tools ..................................................................................................... 6

2. Game Design ..................................................................................................................... 7
   2.1 Game Description ......................................................................................................... 7
   2.2 Game Component Systems with Innovating Game Mechanics ................................. 8
   2.3 Game Component Systems ........................................................................................ 10

3. Project Progress and Players Feedbacks ........................................................................ 12
   3.1 Phase One .................................................................................................................... 17
   3.2 Phase Two ................................................................................................................... 24
   3.3 Phase Three ................................................................................................................ 28

4. Conclusion ...................................................................................................................... 33

5. Reference ......................................................................................................................... 34
List of Figures

Figure 2-1 Attacking System Flowchart ................................................................. 9
Figure 2-2 the Structure Diagram of Client-Server with Dedicate Server Pattern ................................................................. 11
Figure 2-3 Design Diagram of Tile-Matching Game ............................................. 13
Figure 2-4 Number Guessing Game Design Prototype Diagram .......................... 14
Figure 2-5-1 Game View of In Game UI 1 .............................................................. 15
Figure 2-5-2 Game View of In Game UI 2 .............................................................. 16
Figure 2-6 Game View of Menu UI ..................................................................... 17
Figure 3-1 Game View of been Aimed ................................................................. 20
Figure 3-2 Game View of been Attacked ............................................................... 21
Figure 3-3 Game View of Hacking ....................................................................... 21
Figure 3-4 Game View of Successfully Hack System ......................................... 21
Figure 3-5 Game View of Failed to Defend System ........................................... 22
Figure 3-6 Online Survey Form ........................................................................... 24
Figure 3-7 Enemy Entity in Patrolling State ......................................................... 26
Figure 3-8 Enemy Entity in Chasing State ............................................................ 26
Figure 3-9 Game View of Maze Mini-Game ......................................................... 27
Figure 3-10 Game View of Guessing Number Mini-Game .................................... 30
Figure 3-11 Upgrading Menu ............................................................................. 31
Figure 3-12 Top View of Terrain ....................................................................... 32
Introduction:

1.1 Purpose

The main purposes of this report are to explain the reason of developing a new game mechanic for the project, MER-A, describe the game system structure and innovations of the new game mechanic, and discuss the feedback from players to conclude how likable and acceptable the new game mechanic is.

1.2 Background

With the evolution of Internet technologies, multiplayer games have gradually become more and more common as one of the core elements in the gaming industry. There are many excellent game mechanics for action games developed with multiplayer game elements, involving actions such as attacking other players. For example, Counter-Strike, combines shooting with the multiplayer mechanic, and World of Warcraft, combines RPG with the multiplayer mechanic. These games are successful and popular among most of players, and they have a good reputation as multiplayer games. For these games, their attacking systems are similar in many aspects, where enemies take direct hits by players. And players enjoy the attacking systems by exhilarating feelings of shooting or hitting enemies.

Recently, a few games were released with unique and innovative attacking systems. The most representative game of this type is Outlast, where the attacking actions do not involve directly hitting enemies, but activating traps or escaping from enemies. From players’ feedback, Outlast is a successful game due to its unique attacking system.

Based on the success of multiplayer elements, and the success of these recent unique attacking and task systems, we were inspired to simulate a real rover driving experience as a MAR-A driver. This project focuses on investigating the likeability and acceptability of general players to new game mechanics that combines developing a unique attacking system, an innovative task system, and the multiplayer element.

1.3 Developing tools
This project is mainly developed with Unity and Socket.IO. The game server is hosted on a personal computer (PC).

1.3.1 Unity

Unity is a multiplatform game engine which supports both 2D and 3D graphics, drag and drop functionality and scripting through its three custom languages \(^1\). Unity also has high-level scripting API and built-in networking features to fully support development of multiplayer games. Most importantly, Unity supports Socket.IO asset, and Socket.IO can be downloaded from the Unity app store.

1.3.2 Socket.IO Asset

Socket.io is an event-driven networking library that is built for real-time bi-directional communication between each connected client-side \(^2\). It has two parts: client-side library and server-side library which have a nearly identical API. Most importantly, it supports all platforms, browsers and devices that supports Unity.

Game Design:

2.1 Game Description

The name of this game is MER-A, which is a third person adventure multiplayer game. The main gameplay element consists with cooperating tasks such as solving puzzles, trading, exploring, gathering new materials, and competing tasks such as robbing other players, take territory, brake other players’ rovers on a large Mars map. And players will try to capture exploration points as high as possible to upgrade the rovers’ abilities such as speed, battery life etc. to get stronger and be dominant in the game world.

MER-A expected to play fluently under 20 people online at the same time due to the game server running on a portable PC with limited calculating ability and limited network speed.
The storyline of MER-A is based on exploring Mars as a MER-A driver. The game is set sometime in the future, when every country has their own technology to send their exploring rovers to Mars. The player will play as an exploring rover driver to complete tasks to make the country you belong to win the exploration points as much as possible.

On the Mars map, there are some bases set up. A player can upgrade, repair in the base, which belongs to the same country as the player, and player can hack enemies’ bases to get exploration points.

2.2 Game Component Systems with Innovating Game Mechanics

There are mainly two game systems that contain innovating game mechanics, which are the attacking system and the task system.

2.2.1 Attacking System

Instead of the mechanic of attacking system that directly attacking (shooting, punching or kicking etc.) other players or enemies that other multiplayer games usually used, this game will introduce an new way of attacking other players, which scans to aim a enemy with leaser beam and initiates a cipher attack, which requires the attacker to win a mini puzzle game to successfully hacking, to the aimed enemy. As Figure 0-1 attacking system flowchart, when player one use laser targeting the player two first player two’s radar will start to alarm to show someone is hacking, and if player one presses a hacking button to start hacking, mini game system will generate a mini puzzle game for player one such as Tile-matching, Pac-Man, Sokoban etc. to play. And if player two finds player one, player two will starts to hack as well, so it becomes to compete how fast two players can solve puzzles. If one of the player wins the game firstly, the other player’s rover will be broken, and lose some of exploring scores to be given to the wining player, and the losing player will start at respawn point.

Based on this mechanics:

a. Upgrade system could upgrade the anti-virus level, which increasing the difficulty of the puzzle game that attacker
played, and hacking skill, which decreasing the difficulty of the puzzle game that the player played.

b. Item system could have electric bombs be assembled by resources, which can temporarily shut down the other players’ system to avoid attacking from other players.

Figure 2-1 Attacking System Flowchart
2.2.2 Task System

Instead of the mechanics that directly telling player what to do for the main task line the other games usually used, this game will not have a specific main task line to follow. The only task that player can explicitly been told is everyday experiment. This highly simulates the real-world situation, which when firstly come to the unknown planet, the main thing is exploring the world freely. However, tasks including solving puzzles are triggered by some location of the Terrain. And Event as well.

2.3 Game Component Systems

MER-A consists with nine core game components, which are: networking, task system, camera movement manager, attacking system, mini game generator, event handler, interactable terrain, UI system, and upgrade system.

2.3.1 Networking

The network pattern of MER-A is client-server with dedicate server (Figure 2-2 the Structure Diagram of Client-Server with Dedicate Server Pattern), which the server maintains all information of the game state such as the position, rotation of the game objects, and each client only communicate to the server and let the server change the game state and inform other players. And Socket.IO is applied to this pattern to create the server. Because it is based on Javascript, which can inform other players
synchronously with callback functions to optimize the performance.

![Figure 2-2 the Structure Diagram of Client-Server with Dedicate Server Pattern](image)

2.3.2 Task System

Task System is responsible to create corporation tasks and single tasks. From the programing architecture of view, this system has three parts: task library, which stores different tasks, task generator, which trigger out the tasks stored in the task library, task controller, which keep tracking of task state for each task.

2.3.3 Camera Movement Manager

Camera Movement Manager is allowing players to follow and watch the game objects from third person view in any angle during the tasks smoothly without getting through the wall or other game objects, also allowing event handler use for showing event, for
example, switch the camera to the place where event happened to give players a hint.

2.3.4 Attacking System

Attacking system allows players to aim other players with a red laser in the front of the rover and press hacking key to starting hacking. Hacking could be successful if the player wins the mini game, after successfully hacking, the player will be rewarded with some exploration points from other players, or unsuccessful if the player loses the mini game, then, nothing happens. The player who is hacked successfully by other players loses some exploration points and return to respawn point around base.

2.3.5 Mini-Game Generator

Mini-game generator which associate with attacking system mainly generates two different types of mini puzzle games, which are tile-matching game and number guessing game:

2.3.5.1 Tile-Matching Game

Tile-matching game asks players to find two matching pictures in limited time. To meet the theme of hacking, the pictures used for tile-matching mini-game are screen shots of some codes as code pieces. The layout of the tile-matching mini-game panel is arranged as Figure 2-3 Design Diagram of Tile-Matching Game. players are asked to find the codes from the left side to matching the codes on the right side.

For example, in Figure 2-3, code pieces panel, which is on the left side, has some pictures of code pieces for players to select. In this case, there are five selectable code pieces. Target code panel consists with pictures of code piece that players is looking for the same on the code pieces panel. In this case, it is Target Code 1. When players click on one of these pictures, the picture gets brighten, which means that picture is set as selected. In
this case, if we select code piece 4, which suppose it looks the same as Target Code 1. When player select two of these pictures, it checks if these two pictures are matching. If they match, they are cancel out. In this case, Code piece 4 and Target Code 1 disappeared, because the player finds out all matching pictures of code pieces, the player wins. Otherwise, the chance of trying decreases one until the player finds out all matching pictures of code pieces or the chance of try becomes zero.

![Design Diagram of Tile-Matching Game](image)

**Figure 2-3 Design Diagram of Tile-Matching Game**

### 2.3.5.2 Number Guessing Mini-Game

Number guessing mini-game asks players to find some unknown digits of a number with some information provided by the game. The layout of the number guessing game is as Figure 2-4 Number Guessing Game Design Prototype Diagram shows. The information of the number shows on the left panel, players are asked to guess the missing digits of the guessing number with “_” marked on the middle panel,
which is the panel to show the guessing number, by pressing the digits buttons on the right panel that is an input panel.

For example, in Figure 2-4, the guessing number is “1785720”, which is randomly generated, the second, fourth and sixth digits are missing. On the left panel, which is the hints of the numbers, shows some information, for example, the sum of all digits is 25, the product of all digits is 0, etcetera. And the first input digit is the second digit in this number. The player can guess a combination of the missing digits based on the hints, and press seven, then press 5, then press 2 and press OK button to check if the guess is right, if it is right, the player wins. Otherwise, the chance of trying decreases one, and the hints panel shows that password was wrong to ask the player guess again.

![Figure 2-4 Number Guessing Game Design Prototype Diagram](image)

**2.3.6 Event Handler**
Event handle triggers events, for example, if the puzzle is done, a
door will open, this is different from task generator, because task
generator is to start a task, but event handler is to trigger some
events.

2.3.7 Interactable Terrain

Interactable Terrain is created with Unity terrain system, and the
textures and shapes are using Mars assets. Tasks are set on
different locations of the terrain. When players drive close to the
task area or interact with some activate game objects, there will be
some hints to ask player to press a button to start the task, and the
task will be activated.

2.3.8 UI System

UI system contains two parts: in game UI and menu UI.

2.3.8.1 In Game UI

As Figure 2-5-1 Game View of In Game UI 1 and Figure
2-5-2 Game View of In Game UI 2, in game UI shows
different states of the players such as current speed, health, battery, exploring scores etc. when players aim on a game object, in game UI also shows the information of the aimed game object. And players can open an upgrade menu to upgrade abilities for the rover when players are near to the base.

**Figure 2-5-1 Game View of In Game UI 1**

![Image of In Game UI 1](image1)

**Figure 2-5-2 Game View of In Game UI 2**

![Image of In Game UI 2](image2)

### 2.3.8.2 Menu UI

As Figure 2-6 Game View of Menu UI, menu UI is responsible for signing up a new game, signing in to the saved game, adjusting volume and change control etc.
2.3.9 Upgrade System

Upgrade system allows players to upgrade the rover speed to move faster, increase the health, battery and increase the hacking ability to lower other players hacking defending system and defending system to make more difficult for other player to hacking by spending exploration points.

Project Progress and Players Feedbacks

The project progress is divided into three phases with three user tests and studies: a playable prototype is finished on phase one and six players are invited to test the game and give some feedbacks

3.1 Phase One
End of phase one, the core elements of the game server, basic terrain, the attacking system, mini-game generator, and camera movement manager are built. And a survey about the game is taken by six players.

3.1.1 Game Server

The game server allows players to sign up a new game to play, log in to a saved game, and it stores each players information into a database which built with JSON files. The information of each player includes player name, player password to login into the game, and player’s exploring scores, player position, player rotation, if the player’s leaser is on, if the player is attacking or under attacking. And when one of the players’ information changes, Socket.IO sends the changing variable to the server, and server updates the changes and broadcast to other players. The game server also holds the number of current online players and send it to each client to inform players current online players.

3.1.2 Basic Terrain

The basic terrain is built based on Mars topography and textures with one piece. There are two interactable bases built on the terrain, which player can get upgrade, repaired, or hacking to the enemies’ base. The sandstorm effect is also added to the terrain.

3.1.3 Attacking System

Player to player attacking is finished, which allows players to aim a player by leaser and activate a hacking attack to damage the aimed player. When the player aimed to other players, a warning message “you are lock on” is shown on the aimed players screen, as Figure 3-1 Game View of been Aimed, and when the player activates a hacking attack, the aimed player’s screen shows “you are under a cipher attack” as a waring message, as Figure 3-2 Game View of been Attacked, at the same time, the attacking player starts to play a mini-puzzle game, as Figure 3-3 Game View of Hacking. And if the player hacked successfully, the aimed player would get the waring message “you failed to defend your system”, as Figure
3-4 Game View of Successfully Hack System and Figure 3-5 Game View of Failed to Defend System

Figure 3-1 Game View of been Aimed

Figure 3-2 Game View of been Attacked
Figure 3-3 Game View of Hacking

Figure 3-4 Game View of Successfully Hack System
3.1.4 Mini-Game Generator

The main framework of mini-game generator is achieved, and this framework allows programmer to add or remove mini-games easily without effect other game components. And tile-matching game has been developed and added to the mini-game generator, so when the generator generates a mini-game, it just passes the parameters such as difficulty, count-down time, and remaining times of trying to tile-matching game creator to create a tile-matching game. A new mini-game creator can plug in to the mini-game generator easily in the future work.

3.1.5 Camera Movement Manager

Camera movement manager allows players to view the rover and the environment around players as a third person view. The camera can also detect walls or other game objects and avoid getting through the wall or objects. Also, when there is an object between camera and player objects, the camera zooms in automatically to
avoid blocking players view. Another feature of the camera movement manager is allowing players to zoom in or zoom out the view from the player objects.

3.1.6 Phase One Player Feedbacks and User Case Study

When the phase one of development is done, a game playing session is held, and six players are invited to the session to play MER-A. After the game playing session, a survey is taken by six players, and some bugs and advises are collected as improving data for phase two.

3.1.6.1 Survey

The survey contains five questions, which are: (1) How enjoyable was the attacking system? (2) How enjoyable was the game play? (3) How well does the networking and multiplayer component of the game work? (Is it not lagging?) (4) How enjoyable are the mini-games that need to be played when hacking other players? And (5) Do you have any suggestions for improving MER-A?

As Figure 3-6 Online Survey Form shows, for the first four questions, there are five grading scales to choose from, which are perfect (5 points), great (4 points), good (3 points), bad (2 points) and worst (1 point). After the game play session, the grading points of the first question is 25/30, the second question is getting 27/30, and for the networking system, because of some bugs, it gets 23/30. And the grading points for the last question is 28/30.
From the grading points, the phase one of development is considered to have been finished successfully. Players are enjoying the most of the new game mechanics and think the new game mechanics are fun and novel.

### 3.1.6.2 Bugs

During the game play testing, there are some bugs happened: (1) due to using update function instead of fixed update function in Unity, the speed of each players is slightly different; (2) after the rover flips, instead of stops on the flips position, there are some slightly shifts added on flips position to make rover slightly moving. (3)
because the mass of the rover is set to very small value, when two rovers runs into each other, they fly away by hitting. And these bugs are fixed in phase two.

3.1.6.3 Advices

There are some inspiring and valuable ideas that players provided to improve the game after the game playing session: (1) adding the break to the rover to halt the rover faster; (2) fixing some problems of physical algorithm of collision, because some collisions of rover and other game objects looks unnatural; (3) after the rover flips, respawning the rover around current position, not in the base; (4) adding some animation when start the laser; (5) polishing some sound effects to make it sounds more natural; (6) when getting exploring points, showing up a message to inform player that the player gains what rewards; (7) polishing the text fields for sign up and log in menu.

Some of these valuable advises are considered as a reference when developing phase two.

3.2 Phase Two

In phase two, enemy AI system is developed, and two sample AI enemies are placed on the terrain. A maze mini-game is built and added into the mini-game generator library. The physical algorithm of collision is fixed to ensure that the Collison of rover and other game objects looks more real. Adding some puzzle objects on the map to create some task for player to explore.

3.2.1 Enemy AI System

Enemy AI system mainly contains two parts: enemy base and enemy entities. For one enemy base, there could be random numbers of enemy entities patrol. As Figure 3-7 Enemy Entity in Patrolling State and Figure 3-8 Enemy Entity in Chasing State, each enemy entity has two states: patrolling state and chasing state.
When in patrolling state, enemy entity is walking around the enemy base and looking for players. If the entity’s laser scanned the player, the entity switches to chasing state, which enemy entity is chasing after the player, and firing. When enemy entity is shooting to player, it causes damage to the rover. And the player can avoid being attacked only by running away from the enemy entity in a certain distance.
3.2.2 Maze Mini-Game

A new mini-game, Maze mini-game, is added into the mini-game library for mini-game generator to create. It is triggered randomly when the player hack to other players. As Figure 3-9 Game View of Maze Mini-Game, it is in first person view, and the goal to win this mini-game is to find the exit in a limited time. The maze is loaded from the maze library so that player is in a different maze every time.

![Figure 3-9 Game View of Maze Mini-Game](image)

3.2.3 Fix Bugs

The player controller update function is updated to fixed update function, and game speed of all players with different PC is synchronized.

3.2.4 Phase Two Player Feedbacks and User Case Study

After the end of phase two, eight players are invited to play MER-A, including two players joined previous user test, and six players who have never try this game before. For two players joined
previous user test, a survey of satisfaction of improvement is given. And a survey from previous user test is given to eight players.

3.2.4.1 Survey of Satisfaction of Improvement

Survey of satisfaction of improvement has three questions to evaluate how well the improvement meets players’ expectation. These three questions are: (1) how smooth the player control is? (2) how real the movement of rover performance? (3) is there any improvement still not enough? For question (1) and (2), they are graded as perfect (5 points), great (4 points), good (3 points), bad (2 points) and worst (1 point), and for question (3), player can write a comment.

According to the result of this survey, the improvement and bug fixing is successful with question (1) 9/10 and question (2) 8/10. And the improvement which not enough is the sound effect, which is planning to be polished in next phase.

3.2.4.2 Survey of Game Play Experience

Survey of game play is the same as the previous user test. According to the result, the game play experience has average 36/40 points except the game play experience of the new added mini-game, which is the maze min-game.

From the comment, the reasons of the maze mini-game not suitable for hacking can be summarized as four points: (1) spending too much time on walking in the maze to find the exit; (2) almost impossible to find the right exit in a limited time; (3) walking through a maze feels like playing another game, which is not matching the game theme; (4) the maze game distracts players from playing the main game.

3.2.4.3 Bugs
There is one bug found from this game session by players, which is when the player is playing the mini-game, the game view of mini-game is overlapping the game view of driving rover, which makes mini-game hard to play by some distractions.

3.2.4.4 Improvement for Phase Three

Based on the result of the survey, the priority for next phase is to replace maze mini-game with other type of mini-game. According to some online researches, advices of players and experience of playing some online mini-games, guessing number mini-game which is inspired by Hangman(game) is decided to be implement in next phase. Then overlapping UI when switching game state is also planned to improve in next phase.

3.3 Phase Three

The main tasks that phase three achieved are: (1) developing a guessing number mini-game, (2) finishing upgrading system, (3) polishing UI System, (4) polishing sound effect and game background soundtrack, (5) final user test, user case study and game improvement, (6) game and game server maintenance.

3.3.1 Guessing Number Mini-Game

After removing the maze mini-game from the mini-game library, the guessing number mini-game which is like game Hangman. Instead of guessing missing letters of a word, the guessing number mini-game is asking players to guess missing digits of the number with some given information of the digits.

As Figure 3-10 Game View of Guessing Number Mini-Game, information of the missing integer shows on the left hint panel. All missing digits are represented by underscore in the middle, and red underscore is the current pointing missing digit. The player can
click the digit button on the digit panel on the right to input number, after fill all missing digits, player can click OK button to confirm answer. And player can also click BACK button to go back to previous digit to correct the number of the digit.

3.3.2 Upgrading System

After players earning some exploration points, they can spend the exploration points on upgrading the rover. The upgrading menu can be opened by interacting with the base that the player belongs.

As Figure 3-11 Upgrading Menu shows, there are currently three upgrade items: speed, battery, and health points. Players can upgrade each item by clicking the “+” button. When the item upgraded, it costs corresponding exploration points.
3.3.3 Polishing

All UI items in phase three are polished, and the bug of switching menus causing overlapping is fixed. And an option menu is added to control the sound effects and game volume. All terrain objects and interactable game objects are also polished as Figure 3-12 Top View of Terrain. The terrain is reshaped to make it more fun to explore. Players can explore more interesting game object to unlock some hidden tasks to get awards. Finally, rover engine sound effects, AI entity fire sound effects, and sand storm sound effects are added to the game, some background game soundtracks are also added for both normal mode and under attack mode, which is when AI entities are shooting to players, the soundtrack of under attack mode raises, and after players run away from the AI entities, soundtrack is switched back to normal mode.
3.3.4 Final User Test and User Case Study

During the final user test session, there are fifteen players invited, including seven players who played previous version of the game and eight players who is first time playing the game. The game play session is divided into two groups, and players in group one played previous version of the game, players in group two are first time to play the game.

The survey for the group one has three questions including (1) how better is the guessing number mini-game than the maze mini-game? (2) How satisfied are you with the of the game polishing? (3) What is the most attractive element to you? For the first question, the ranks of this question are much better (2 points), better (1 point), the same (0 point), worse (-1 point) and much worse (-2 points), after playing, the result of this question shows that the improvement of mini-game is successful, which the points of this question is 10/14 with the range of total points from -14 to 14. For the second question, the ranks are very satisfied (3), satisfied (2),
normal (1), and not satisfied (0). From the result of the second question, which total marks is 19/21, players are satisfied with the polishing of the game. For the last question, attacking system is the most mentioned game element of all, the main reasons are (1) mini-games are easy and fun; (2) when hacking to other players, mini-games are suitable for the theme which is hacking; (3) when meet with enemy entities, the rover can only run away to avoid them not attacking them directly, which makes the game more challenging.

The questions of the survey of the second group are based on previous questions from previous survey, which are (1) How enjoyable was the game play? (2) How well does the networking and multiplayer component of the game work? (Is it not lagging?) (3) How enjoyable are the mini-games that need to be played when hacking other players? And (4) What components of the game do you think is the most attractive one and (5) what component of the game do you think still need to improve? According to the result of the survey, the experience of the game play is enjoyable and very interesting for players, specially for the way of attacking, but since there is not many mini-games, after playing for many hours, players is getting sick of the mini-games. Therefore, developing more interesting mini-games can be future work to make the game more enjoyable. And for the performance of the networking system, players are very satisfied, because there is no lagging and every player has a synchronizing game state. Finally, the most interesting and attractive game components are the attacking system and enemy system, and the reason is because players can play some strategies with the AIs and other players, which trick AI or other players to fight against each other. However, the most wanted component to be improved is the task system, because there is not much tasks to play, only there are only some hidden tasks amiable. And the main task is short.
3.3.5 Game and Game Sever Future Update and Maintenance

Based on all surveys before, there are some future updates are planned to achieve: (1) enlarging the game terrain and create more tasks, (2) creating more interactable game objects to let players explore, (3) crafting more rover models and added into game for players to purchase and collecting, (4) adding communication channels so players can communicate with each other in game, (5) creating more mini-games in attacking library.

According to the evaluation of game performance, the game and game server should be maintained once a month to ensure there is no error saving data on the database server and adding new features into the game.

Evaluation and Conclusion of Shippable State of MER-A

From the feedback of players and the four keys to fun[^4], MER-A has all keys to make the game fun to play, which are:

Hard fun: when players explore the terrain while avoiding AI enemies’ chasing and fighting with other players, which the player who plays well can get a feeling of great satisfaction and pleasure caused by getting lots of exploration points and updating its abilities.

Easy fun: players can explore the terrain that contains a variety of mysterious landscapes. The exploration of terrain brings players with curiosity that makes players wants to discover more hidden tasks of the game.

Serious fun: In hacking state, MER-A offers some relaxing mini-game for players to win in limited time, this element gives players both relaxation and excitement.

People Fun: the game element of multiplayer strongly supports the key: people fun, which players compete each other with driving and attacking skills and cooperate with some players defeat other players.
In conclusion, the project, MER-A, achieves the goal of developing and investigating a new game mechanics, which mainly are an attacking system and the free-task system. And these two new game mechanics can be adapted in many other action games in the way of attacking and the way of exploring, which can offer players who get sick of gun-shooting and weapon-cutting with a relaxing attacking experience.

Additionally, more mini-games are required to be added into the mini-game library. The requirements of the mini-games are (1) all mini-games should be puzzle-like game; (2) all mini-games should be simple enough to be possible for players winning within 3 minus, which is the time limit of mini-games; (3) all mini-games should be challenging, which players have chance to win the mini-game in 3 minus, but it has certain difficulty to challenge players; (4) all mini-games should meet the theme, which cannot make players feel like playing another game; (5) and all mini-games should not distract player from playing the main game, which is exploring mars and competing with other players.

References


