VR Shooting Game

# Introduction

Github Repo: <https://github.com/NekoMimiUnagi/VR-Shooting-Game>

GitHub push policy:

* Develop under your git branch. Each branch is named after your initial.
* Before pushing the development codes/models, please submit a pull request. It will be merged after being reviewed by another member.

Member: (write down your GitHub account email so that I can add you as a collaborator on Github)

* Jun Li (jaydenli99@hotmail.com)
* Yu Chen (cy771761929@163.com)
* Ruosen Li (rxl190028@utdallas.edu)
* Xiaoxing Chen (xiaoxingchen630@gmail.com)
* Yifan Lin (yxl200063@utdallas.edu)

# Tasks (Single-Player Related)

**All of the following can be adjusted when developing. But please update them in this document before implementing them.**

## Weapon Related (Yifan Lin)

**\*\* Do not need to build models for weapons at this step. \*\***

Weapon: Bow/Crossbow, Shotgun, Rifle, Ray Gun, Grenade, Grapple, frog generator.

Ammunition: Arrow, Bullets, Ray, frog.

Weapon features:

* Bow/Crossbow: Shoot one arrow per round. Need a long time to reload. Arrow's initial speed is medium and has gravity. ✔
* Shotgun: 6 bullets in a Magazine. No more than 7 in a shotgun (one can be in the chamber). Players can reload before the magazine is empty. Reload time depends on how many bullets to load. Bullet's initial speed is fast and will be determined by a detailed model later. Bullets have gravity. ✔
* Rifle: 30 bullets in a Magazine. At most31 (One can be in the chamber). Can reload before the magazine is empty. A non-empty magazine is kept in the pocket for the next reload. They will be put at the last of the magazine queue. ✔ Reload time is short. Bullet's initial speed is fast and will be determined by a detailed model later. Bullets have gravity. ✔
* Ray Gun: Shoot one ray per round. It has a time limit for each magazine. The remaining time automatically grows up when not shooting. The ray is just a line with no gravity.
* Grapple: It is used to attract supply boxes.
* Frog generator: Shoot once. Then, the player can move the frog directly.
* Grenade: Throw one per round. It needs little time to reload. Its initial speed is slow. Grenade has gravity. Its initial movement vector is higher than the point direction and moves with a positive angle (to be determined).

Assumptions for development:

* The main focus of the development is on the physics of bullets/arrows/rays/grenades.
* The number of magazines and grenades are variables. They will be determined later.
* All ammunition has a collider and gravity.
* All weapons can be represented by a tube for now. All bullets and rays come out from one end of the tube. The tube will be replaced by detailed models later.
* All items can be represented by a box or sphere. The assumption is that all items will disappear after being shot.
* One play can take at most one weapon + grenades + a frog generator + a grapple.

## Movement in Lobby and Shooting Range and UI (Ruosen Li)

**\*\* Do not need to build models for the lobby and scene at this step. \*\***

The whole moveable space includes the lobby and three shooting ranges. Do not need to set the lobby as square or circle. The main development point is the moving logic.

Main UI:

* Has three items: Start, Join, and Quit.
* Join is an item for the multiplayer part. Just leave it there.

Inventory UI:

* As shown in the project pitch, it has multiple weapons to choose from. Every play has the same weapon inventory. Do not need to draw pictures of weapons. Other members will provide them later.
* It is triggered by shooting the table.

Lobby Features:

* There are some non-moveable items in the lobby, so the Player cannot go through them. It can be examined by some unmoveable boxes.
* The lobby has space limitations (to be determined by a detailed scene model).
* Players can call the menu to quit the game.

Shooting Range Features:

* The shooting range has space limitations (to be determined by a detailed scene model). This can be tested by setting a box range, and players cannot move out.
* There is a table in the shooting range for players to choose weapons. When the player aims at the table and presses the shot, it will pop the inventory UI. The table is not moveable.

Assumptions for development:

* The player and all items in all movement places have the corresponding collider.
* Players can move freely in the lobby.
* In a lobby, players can go to a scene by standing in a shooting range. It can be tested by setting an area as a virtual shooting range. This is a kind of teleportation. At this time, no scene is loaded. The shooting range is not the same as a real shooting range.
* In a scene (real shooting range), players can move to an edge or an area representing a virtual lobby to quit the scene. This is a kind of teleportation.
* The lobby is loaded with the main UI menu. The main menu can be semi-transparent to the lobby or specifically designed.
* Jump, squat, and lay down are automatically implemented by cardboard movement. In other words, you do not need to implement by code.
* Each player can choose a color.

## Items in Scenes and Scene UI(Yu Chen)

**\*\* Do not need to build models for items at this step. \*\***

Item: pan (Gloucester Clay), animals (to be determined by detailed models), ufo, supply box, vision block (frog).

Item features:

* Pan: Starts from the ground. The initial move vector is to the sky. The difficulty level can adjust its initial speed (to be determined later). It has gravity and a collider. It will disappear when touching the terrain.
* Animals: Move on the terrain. Both move direction and speed are random. The moving speed is low. May stay at a place for a while. It appears and disappears randomly. It has gravity and a collider.
* UFO: Move on the sky. Both moving direction and speed are random. It moves fast. May stay at a place for a while. It appears and disappears randomly. It has no gravity but a collider.
* Supply Box: It appears on the ground or falls from the sky with a parachute. It has gravity and a collider. It doesn’t move (on the ground) and moves slowly (in the sky). It moves to a player when a specific weapon attracts it. Only one player can draw it. If someone attracts it, none of the others can attract it anymore.
* Vision block: a frog in the sky. No gravity and collider. It is selected in the inventory. Put it in the sky by aiming and moving the joystick. At this moment, the player cannot move. In other words, it will override movement operation.

Scene UI:

* As the project pitch describes, the score is in the upper right. The remaining ammunition is at the lower left.

Assumptions for development:

* Use ray to simulate the shooting method.
* Every item has a collider. Discuss with another member who implements task 1.
* Every item disappears when got shot.
* When an item is shot, the total score increases.

## Build Small Models (Preference: Weapson, Hands, Items in Scenes) (Xiaoxing Chen)

**\*\* This task is mainly to design detailed models for small items. If the task is finished and the member for task 5 does not finish, please help that member. \*\***

Build Models for weapons and items for tasks 1 and 3.

Model List:

* Items
  + Pan (clay)
  + Animals (>= 3 species). Select by you. Please provide animal names and moving speeds.
  + UFO
  + Supply box
  + Frog
* Weapons: Please provide specific names, basic information, and initial ammunition speed.
  + Bow/Crossbow
  + Shotgun
  + Rifle (such as M16)
  + Ray Gun (you can fabricate data by yourself)
  + Grapple
  + Grenade
  + Fog generator (you can use grenade launcher or others)
* Ammunition:
  + Arrows
  + Bullets
  + Rays
* Avatars for players

## Build Large Models (Preference: Scenes and Lobby)(Jun Li)

**\*\* This task is mainly to design detailed models for large items. If the task is finished and the member for task 4 does not finish, please help that member. \*\***

Build Models for scenes for task 2.

Model List:

* Lobby
  + It has chairs, tables, trees, and tables (for inventory checking).
* Plain
  + It can be plain or has some small hills. The terrain has a collider so that items can land on it.
* Forest
  + It has plenty of trees. Trees will block some lights.
* The city (optional: along with the sea)
  + Buildings are in the scene. Some UFOs will fly behind buildings. Players can be on the sea (discuss with the member working on task 2).
  + May try the official asset published by Japan government

Audio:

* Background music
* Shotting audio
* Hit audio

# Tasks (Single-Player Tasks Integration)

## Integrate Models in tasks 4 and 5 to tasks 1, 2, and 3. (Ruosen Li)

## Integrate tasks 1 to 3 to form a complete single-player game. (Ruosen Li)

**(In Plan)**

## Build and join room for multi-player

## Voice chat based on the previous task.

## Fully test