

# **Final Project**

## **Arcade - PacMan**

**By:**  
**Ryan Li**  
**Jie Zhou**  
**George Agustin**

### **Abstract:**

In this report we will discuss the overall process of our project and go over the different objectives we wanted to accomplish during the development of the classic game called Pacman. These objectives cover the design process of making our game, utilizing a different maze maps and pacman dots for our game, design challenges and accomplishment that we encountered and the overall tasks we each did. We will also go over the style and gameplay of our game such as instructions that will help the game tester during play.

## **1 - Introduction:**

We would like to introduce the classic arcade game Pac Man that we will implement into our design and development for our final project this semester. Similar to how the classical pacman game has multiple levels, pac dots, ghosts and different mazes for Pacman to journey around, we will be using these same concepts into our game. These concepts will have a fresh and local appeal to it as we discuss the development process later on. It also will implement different parts of homework 2 and 3 in order to have a functional arcade game platform ready for demonstration.

## **2 - Objective:**

Our first task is to figure out what we would like to see in our game and what kinds of things we would like to implement as a version of Pac Man. The next task is the breakdown of building our arcade game that would use the same principle idea of Pacman with different maze maps. We wanted the player to feel at home in way so we thought of using the areas of Oahu to facilitate the map for the maze. Our next objective was to decide our the pacman dots our pacman was to “eat” while traveling through our maze. We wanted to be unique and different so we didn’t want to use the regular pacman dots that was readily available as sprites. Fourth objective of our project, we needed to figure out how to implement the controls for our pacman to move and how it will gain points by eating cars on the road. Next, we needed to figure out the ghost A.I. and how to implement them as the villians of the game that will try to kill Pacman. We also wanted to implement a second player to play as the ghost to eat the Pac man. Lastly, is developing a game title screen for gameplay and utilizing the different levels for our game.

## **3 - Design:**

We started our arcade design with figuring out our maze for our pacman to traveling on. We decided to use our own island roads and maps for the maze. Areas include Kaimuki, McCully, and ChinaTown to serve as multiple levels. We wanted to use the the same Pacman sprite and animation. However the PacMan in our game is black because the original yellow will be different to see with the white background. The premise of the game is to simulate the awful traffic we have here in hawaii. Cars fill the streets of Oahu in bumper to bumper traffic. PacMan will come to clean up the road by consuming the cars (red cars, the equivalent of pac dots in the original game) There are also PowerCars(yellow cars, the equivalent of power pellets in the original game) that will grant PacMan in power up momentarily. The power up changes PacMan’s color to

green like the hulk, it will have twice the speed and be able to eat ghosts in the power up mode.

For our mazes, we needed to add 2D physic box colliders so that our PacMan will only access the roads of the maps. We also implemented circle rigid body for our Pacman in order to register a “collision” between it and the cars on our maze. We also considered the implementation of a running ghost A.I. for it to be the typical antagonist reminiscence of an age old arcade. There are 4 total ghosts and 3 of which just have random movements, while one of the ghost will be control by a second player.

## **4 - Development:**

### **4.1 - Problems:**

Throughout our development process we’ve encountered of problems while building our arcade game. We’ve noticed that we would need to research and import new sprites for use like “car” images that the Pac Man will “eat”. We didn’t want to use a typical Pacman dot but bring a little realism to our map is based upon the streets of Oahu. Another obstacle occurred when researching for the “right” maze or area on Oahu that would be as symmetrical as possible since there’s alot of roads on Oahu that are not symmetrical and have weird diagonal patterns. Next problem that came up in the process of our development is bringing life of our Pacman like moving of its mouth and to mirror the direction of its mouth to the direction of where it’s going. Another problem we’ve encountered is the movement and player control of our Pacman. We also needed to figure out our collisions with our pacman and the walls on the map for a smooth movement through our maze.

Another obstacle we’ve notice is trying to adjust the scaling of our arcade game so that that resolution would be more clear and well-defined throughout gameplay. We also have a continuous problem with collaboration feature with Unity software. It lacks the ability to see the changes made from other group members, and if other members try to push their changes to the unity cloud, it causes a push/pull conflicts on collaboration feature and depending on what you want, you could either lose or keep the C# script changes made. It greatly decreased our production time to develop our game. Another huge problem we encountered during the developmental process is writing too many functions and game actions within one script. For example, our Pacman script consists alot of our game functionality like interactions with the ghosts or gameplay respawning functions. So when the pacman dies, the entire game crashes. Finally adding ghosts and a game screen would be the last obstacle we’ve encountered in this project. Also Unity would periodically crash and slow our troubleshooting of our game. Going through the troubleshooting and debugging process was a difficult process because of errors in code and again collaboration function within unity. These challenges we’ve faced so far have solutions that we’ll be going over.

## **4.2 - Solutions:**

For the first problem it was easily solvable by looking online through google for car images we could use because of how homework 1 uses png. Files for our 2D Pac Man game. There's a number of resources online (refer to cited references page) that helped us to design, develop and build our game. So once we've found the right sprites for cars and for pacman we move on to the background we wanted to use.

Since we decided we wanted to use the street of Oahu as our maze it took a lot of time to look for 3 map areas we could use as mazes. We finally located 3 areas on Oahu we could use as our maze which is in Mccully, ChinaTown and Kaimuki. Each of these areas has close symmetrical properties while minimizing diagonal and weird angles that would hinder movement of our pacman.

Although Chinatown was the map with a few weird diagonal angles, in order to outline our rigid colliders we had to use unity's "polygon" box collider attribute to lower the effects it will have on our Pacman moving through it. Another solution we found in dealing with our pacman is adding animation for our sliced Pacman sprites and running them through an animation sequence in reference to the direction of where the Pacman is facing. For example if the Pacman was to move to the right then we made it so that when all the Pacman sprites facing right would cycle through each other and animate in the right direction. We repeated the same procedure for the left, up and down direction of pacman. After we established this, we moved on to syncing the direction animation of Pacman to the player controls.

We used a C# script to utilize the movement and control of Pacman in unity that we learned from homework 1. From the template UFO player control movement we learned how to implement a working, functional Pacman with player control movements. Through some tweaking and working on our C# script for our pacman collisions with the walls we were able to somewhat have a smooth transition of hitting walls and also along diagonal street routes which were difficult to achieve. We were also able to simulate a working ghost to a certain degree for our game. It seems a bit buggy but it still works as part of our game. To minimize our collaboration errors, we try not to push changes to unity at the same time in order to receive our revisions accurately. Other things we tried as a solution to overlapping scripts is by sending script texts to each other via email so that when we push our data to unity then we can compare and see what kind of changes had and what we missed. We were also able to implement a multi-player capability for a player being the pac man and the second player being a ghost.

Each map/level is a different scene, so we used the scene manager in Unity to link the title page and different levels with button UI.

## **5 - Responsibility:**

We each share a different responsibility to build for our game. Jie research and found each level maps about 3 at different areas of Oahu. Next, we each took a map and made sprite mazes and imported our car sprites used for “eat” by our Pacman. We each added component parts like box colliders for each map mazes to simulate the city blocks of Mccully, Chinatown and Kaimuki map areas. Next, as Jie was our project leader he oversaw our project, combining all codes and scenes, and scheduling meeting times. Ryan oversaw the overall background in C# script coding for pacman controls, movement, and collisions. George provided support and report development of our project.

## **6 - Evaluation:**

Overall, our project had a lot of ups and downs during the process of development. The kinds of things that worked was everyone in the group willing to commit and come together for each meeting to work on our game. For example, we always tried to establish a time around our schedules. But some things didn't work like for example, trying to troubleshoot and debugging through the codes that have minor errors which affected the entire compiler like the ghost-pac man interaction.

## **7 - Player Instruction:**

Gameplay: Open game and then player 1 will go to open screen title, it will give the option to whether or not to enter our pacman game. Once player 1 chooses to enter/play game then he/she will go to our first maze map “Kaimuki”. The object of our game is pretty simple, Pacman has to transverse through the map and eat all the cars stuck in traffic. Once Pacman has eaten the last car, then he/she will move on to the next level. When the ghosts dies, it doesn't respawn. Also when the Pac-man hits a ghost then it dies unless Pac-man runs into a powerup. PacMan has three lives and respawns in the same spot it died. When he respawns, he is given a limited time of powerup for eat the ghosts in nearby area. If a player wants to restart a level then he/she can click on the restart level icon and start over. Another aspect of our game mechanics is the addition another player being able to play as the ghost, but the idea is to kill the PacMan. Since the ghost player is controlled by a person; the fact that PacMan is in power mode shortly after respawn prevents the ghost player from camping in the same spot and killing PacMan repeatedly.

## **8- Conclusion:**

Overall, We are able to accomplish our objectives and achieve our goals in building a classic game of Pacman. We were also able to implement a local feel to the game by using actual google map areas of certain locations on oahu for our mazes. We were also able to implement a player control design of our game for our Pacman to go

around our maze and collect the “cars” for points. We were also able to include music as part our gameplay and value. We also had power-up feature for the pac-man so that it can kill the ghost nearby for a limited time. We also implemented a two-player capability for the pacman and ghost for the second player. Despite our success, we had to fight through the challenges that presented itself during the process. From trying to figure out the animation for our pacman to implementing a ghost A.I. or using and making mazes from our streets on Oahu, and writing our scripts, all the problems was an obstacle that we came together and overcame in the long run. This project was a great learning experience and stepping stone into learning more about programming the software side of engineering.

## **Cited References**

- [1] noobtuts, “Unity 2D Pacman Tutorial”, *noobtuts.com*  
[Online]. Available: <https://noobtuts.com/unity/2d-pacman-game>
  
- [2] 2D UFO Game tutorial, “Beginner 2D UFO Game 1 of 9: introduction...”, *youtube.com*  
[NOTE: continued reference, tutorial 1 through 9 for UFO game tutorial]  
[Online]. Available: <https://www.youtube.com/watch?v=jTtCsOjNwJQ>
  
- [3] arcade game sprite resources, “The spriters resource”, *spriters-resource.com/*  
[Online]. Available: <https://www.spriters-resource.com/arcade/pacman/sheet/52631/>
  
- [4] Textbook, “Introduction to Game Design, Prototyping, and Development: From Concept to Playable Game with Unity and C#, 2nd Edition”, Author: Jeremy Gibson Bond. Addison-Wesley Professional, 2018