

UML and UI Design Document

Become Pac-Man

version: 1

Group F8

1155127434 HO Chun Lung Terrance

Department of Philosophy, The Chinese University of Hong Kong

1155143519 WOO Pok

Department of Physics, The Chinese University of Hong Kong

1155157839 NG Yu Chun Thomas

Department of Computer Science and Engineering, The Chinese University of Hong Kong

1155157719 LEUNG Kit Lun Jay

Department of Computer Science and Engineering, The Chinese University of Hong Kong

1155143569 MOK Owen

Department of Mathematics, The Chinese University of Hong Kong

March 25, 2023

Contents

1 UML DESIGN

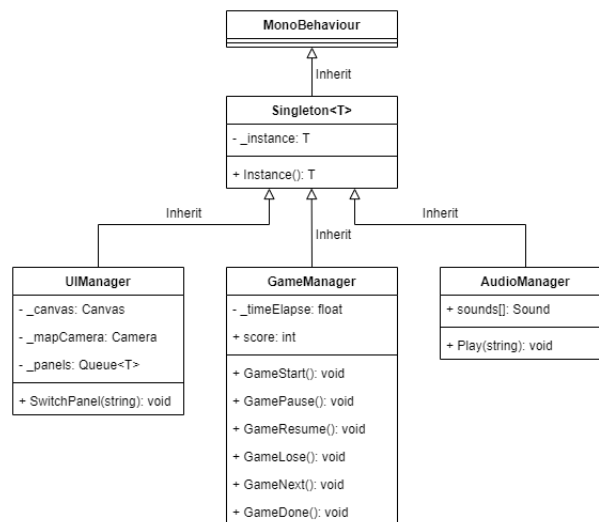
1.1 Manager Class

1.1.1 Structural Diagram

1.1.2 UMLs

use-case diagram

class diagram



sequence diagram

1.1.3 Component Functionality

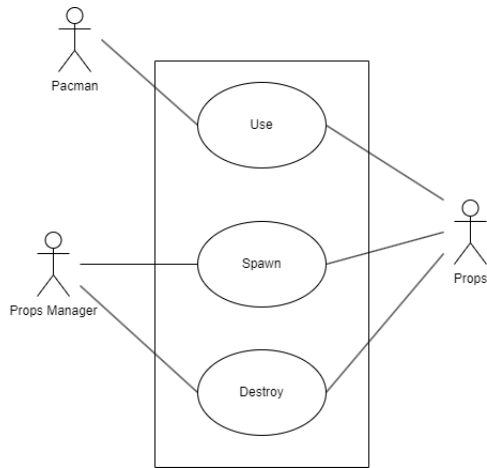
1.1.4 Major Procedure and Functions

1.2 Props Class

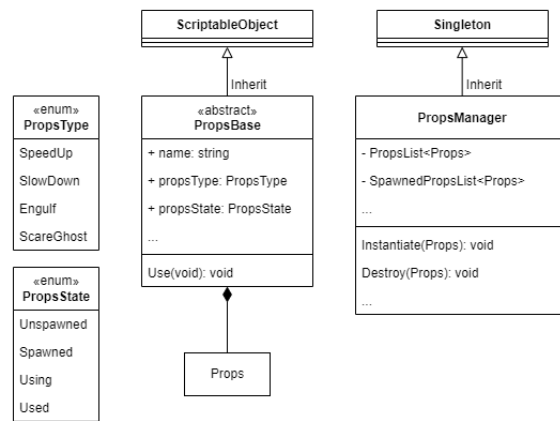
1.2.1 Structural Diagram

1.2.2 UMLs

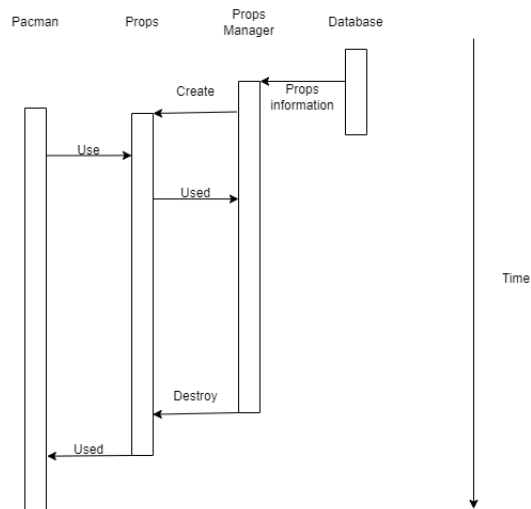
use-case diagram



class diagram



sequence diagram



1.2.3 Component Functionality

The props class is to provide the bases for game objects, such as pellets in the maze for Pacman to achieve scores, or other props that will enhance in-game experience.

1.2.4 Major Procedure and Functions

Use(void): The Use function is to obtain the action of using the prop, by destroying the prop from the maze.

Instantiate(Props): The instantiate function is to create the props for each game.

Destroy(Props): The destroy function is to destroy the prop from the maze for each game.

1.3 Database

1.4 Ghost Class

2 UI DESIGN

2.1 Login/Sign-up page


2.2 Title Screen

2.3 Shop UI

2.4 Setting UI

2.5 Main Game

2.5.1 Ghost mechanism

 Ghost_Class.png

The class diagram depicts the Ghost class and its related classes that control the ghost's movement behavior in the game. The Ghost class contains various behaviors, such as GhostHome, GhostScatter, GhostChase, and GhostFrightened, each of which extends the abstract class GhostBehavior. The Movement class is responsible for controlling the Ghost's movement, while the GhostHome class manages the ghost's behavior when it returns to its home position. The GhostScatter class manages the ghost's behavior when in scatter mode, and GhostChase class controls its behavior when chasing Pacman. GhostFrightened class manages the ghost's behavior when it's frightened. The GameManager class controls the overall game, including the Ghost class. The Ghost class uses the Transform class to move the ghost in the game.

Ghost_Sequence.png

This sequence diagram shows the various transitions that can occur between different Ghost behaviors during gameplay. The diagram starts with the GameManager resetting the state of the Ghosts, then the Ghosts disabling their current behavior before transitioning to their next behavior.

The diagram also shows two alternative paths: if Pacman eats a Power Pellet, the Ghosts transition to Frightened Mode and if Pacman collides with a Ghost, the behavior depends on whether or not the Ghost is in Frightened Mode. If the Ghost is in Frightened Mode, it is eaten, but if it is not in Frightened Mode, Pacman is eaten.

Ghost_Activity.png

The activity diagram shows the different actions taken by the ghost in each of its behaviors. In the Chase behavior, the ghost moves towards the player's location, choosing the shortest path. In the Scatter behavior, the ghost selects a random corner of the map and moves towards it, again find the shortest path. In Frightened behavior, the ghost moves randomly, choosing a random direction at each intersection. During this behavior, the ghost also flashes blue and white to indicate its state.

When the ghost is eaten, it transitions to the Home behavior where it returns to its starting position, bounces around inside the ghost home, and then exits in a random direction. Finally, in the Eyes behavior, the ghost moves to a predetermined location offscreen to respawn and start again. These actions are repeated throughout the game as the ghost transitions between behaviors based on game events and timing.

2.6 Record page