Dumb Dino Inc.

Cuadrado
Use-Case-Realization Specification

Version <2.0>

Revision History

Date	Version	Description	Author
22/OCT/2022	<1.0>	First Draft	Devin Setiawan John Zheng
29/OCT/2022	<2.0>	Formatting and figure labels	Devin Setiawan John Zheng

Table of Contents

Figures

Use-Case-Realization Specification

1. Introduction

1.1 Purpose

The purpose of this document is to provide diagrams and details to document and give a better image of how the software will work

1.2 Scope

The scope of this document is to provide an overview of the realizations for the use cases of Cuadrado. Cuadrado allows the player to interact with the board game through the UI and keyboard input to achieve a game-winning condition. This use case realization document provides the overview of the use case developed.

1.3 Definitions, Acronyms, and Abbreviations

Python - Interpreted Programing Language
SM - Settings Module
WM - Window Module

1.4 References

- Use Case Specification

1.5 Overview

The Use-Case Realization document describes use cases in terms of their flow of events, participating objects, and corresponding diagrams.

2. USE CASE <Settings>

2.1 Flow of Events - Design

Upon the start of the game, the user goes to the settings screen and selects from the various options that change the game behavior or exit back to the main menu. The Setting Module waits for a change from the Window Module and toggles the desired change.

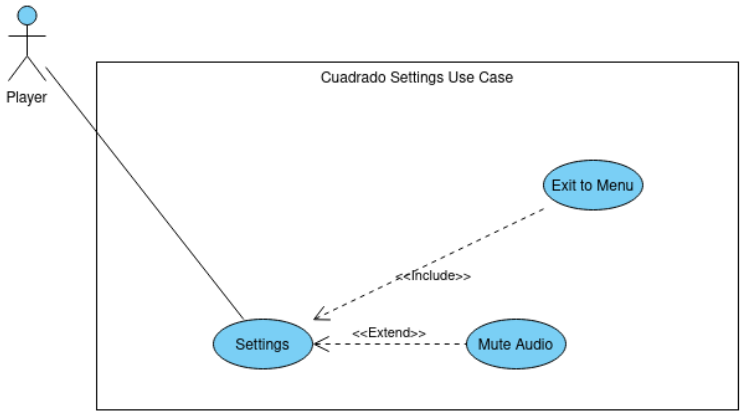

Figure 1: Settings

2.2 Interaction Diagrams

- WM check for input
- SM change the settings
- WM update UI

2.2.1 Sequence Diagrams

The Sequence Diagram shows how the various Actors and Objects exchange messages in the use-case <Settings>.

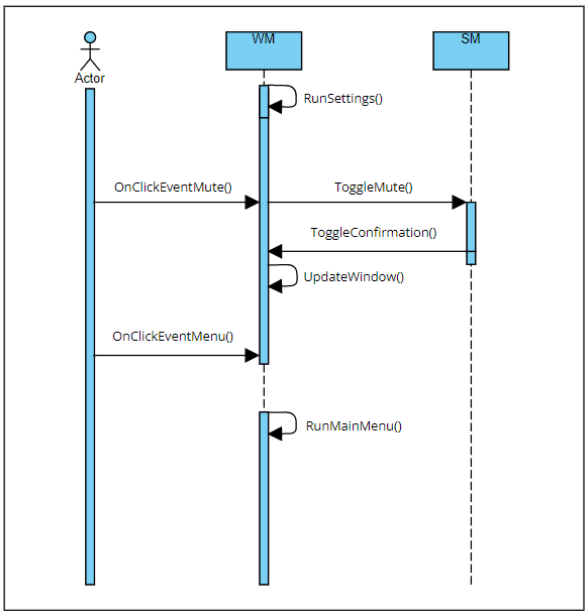

Figure 2: Sequence Diagram: Setting

2.2.2 Participating objects

The following objects define the use-case <settings>:

- SM: This object interacts with the properties file in order to change the behavior of the game.
- WM: This object is responsible for calling instances of the UI to be displayed.

3. USE CASE <Play>

3.1 Flow of Events - Design

Upon the start of the game, the user goes to the game screen and plays the game that tracks the number of points a player has. A player makes moves in the game. Checking for win condition is run until a winner emerges. Scores are updated, and eventually, a win condition is met and a player is victorious, sending them back to the main menu.

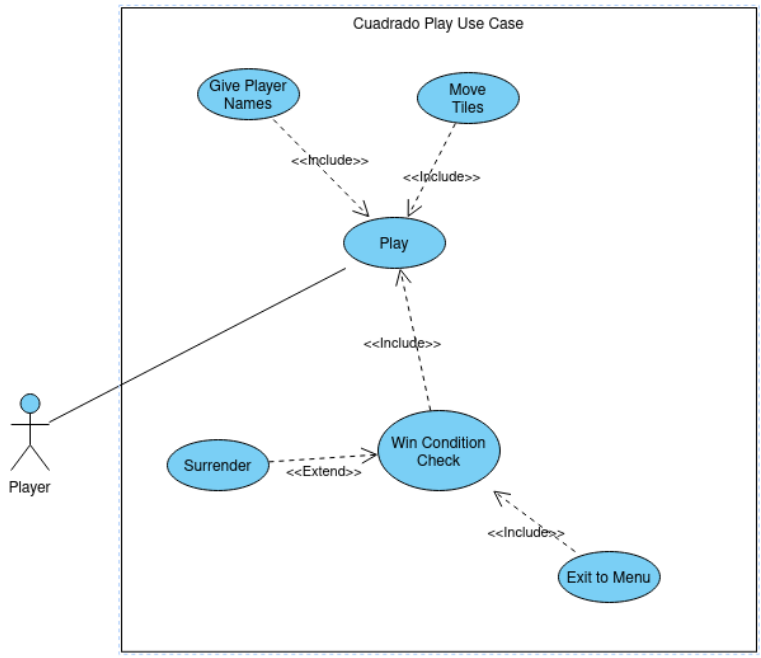

Figure 3: Play

3.2 Interaction Diagrams

- WM run game
- WM load setting
- WM check input
- Win Condition check win

3.2.1 Sequence Diagrams

The sequence diagram shows the interactions from user to use case <Play>

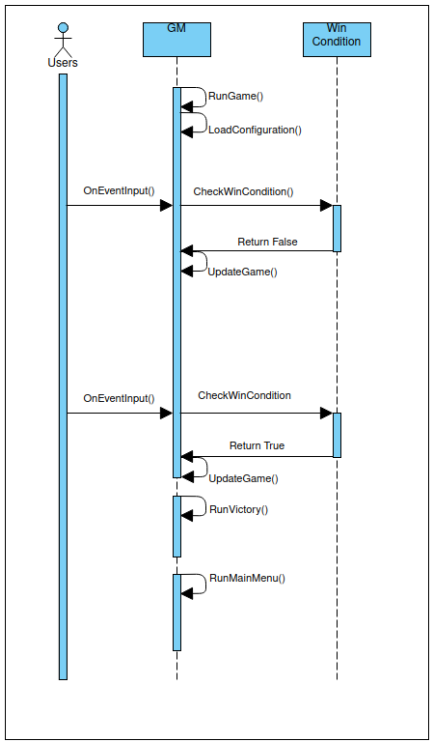

Figure 4: Sequence Diagram: Play

3.2.2 Participating objects

The following objects define the use-case <settings>:

- GM: This object is responsible for handling and updating the board game based on user input.
- Win Condition: This object is responsible for checking for the victory state.