

Candy Land - User Manual

Developed by Alexa Horvath, Claire Engel, Viveka Kurup, and Julia Calderone

Introduction

This program encapsulates a simple, easy to play implementation of the classic board game Candy Land. The digital version captures the basic mechanics of the pawns and cards with respect to the game board. User interaction is kept limited as the players' responsibilities are solely to pick a card; this is relatively consistent with the physical game. The movement of the pawns is done automatically. The screen design is full of color and thematic elements to pay tribute to the original 1949 game. All rules and game components are executed including bridge, licorice, and character spaces. Because of the basic user interaction required and fun user interface, this game is playable by and for all ages.

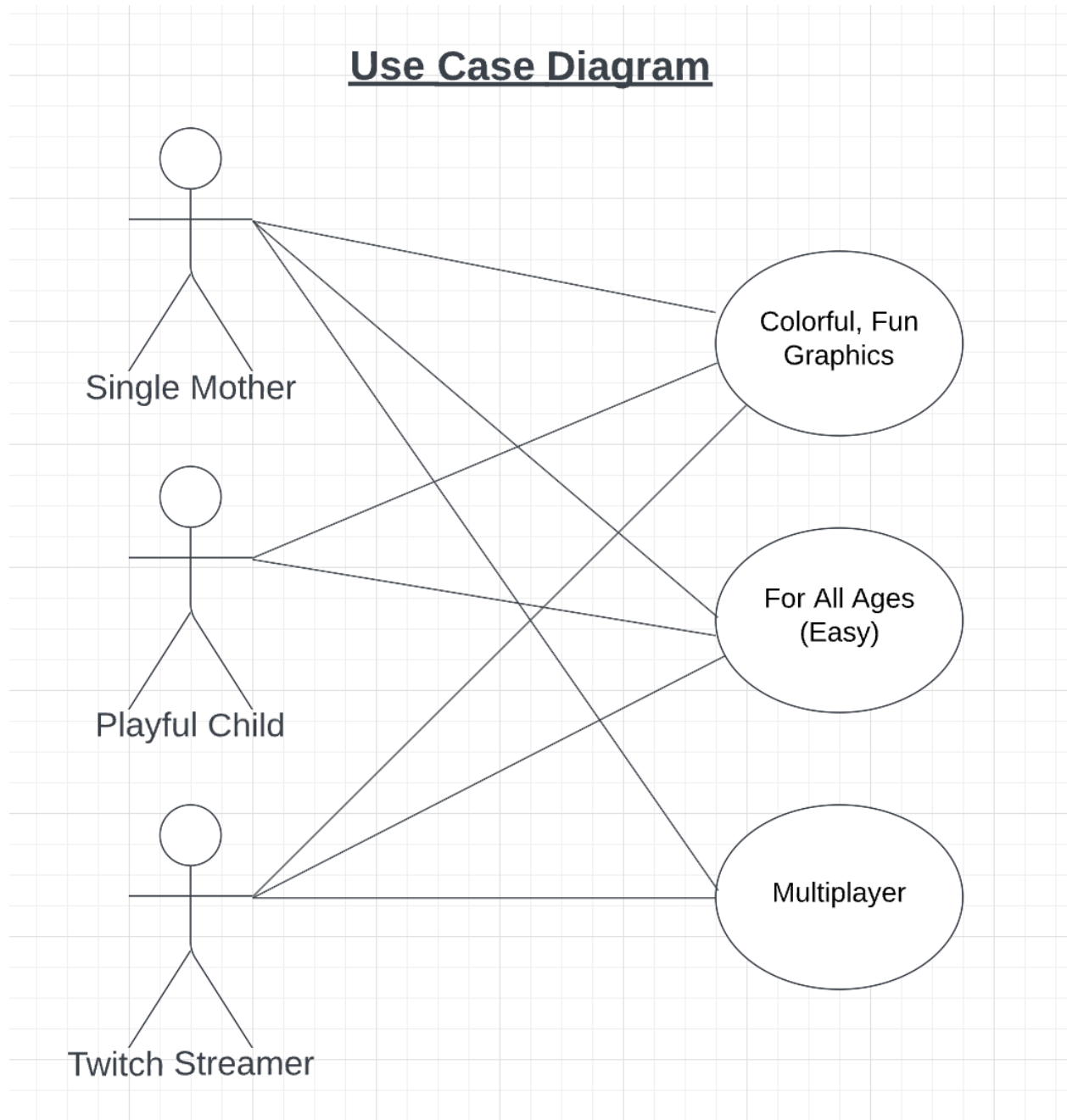
Background & Motivation

The first edition of Candy Land came out in 1949, however, there are few digital implementations of it. Those that do exist are now outdated or have extra features that divert from the original game play in order to attract consumers and new generations. The game's popularity and timelessness comes from its simple instructions, so this was an important component to preserve. Candy Land is a childhood memory for many, so its familiarity was appealing when deciding what game to program. This project set the goal of recreating Candy Land, a simple children's board game, using Java, JavaFX, and Scene Builder.

Candy Land follows four players as they traverse a game board toward King Candy's Castle. Players advance by a process of moving to colored squares based on selected cards. There are obstacles such as licorice spaces which cause players to skip a turn and character spaces that could jump a player far ahead or several spaces back. The objective of the game is to reach the final rainbow space first. The analysis of the original instructions drove the class design; 7 backend classes were created to compliment the graphical user interface. The classes include a Card deck and a card class, a player and a pawn class, a game board and a space class. Within the Candy Land class, four player objects are created and assigned a pawn. For a player's turn, they must pick a card object from the generated card deck array list which will specify the amount of spaces and the color they must move to. With the chosen card, the pawn moves over an array list of space objects that make up the game board, until they are at the correct location. The players' states are saved as an enumeration class, so whose turn it is and whether someone is a winner can be tracked.

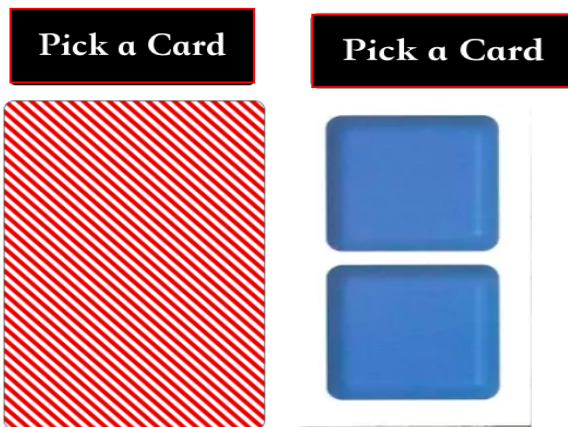
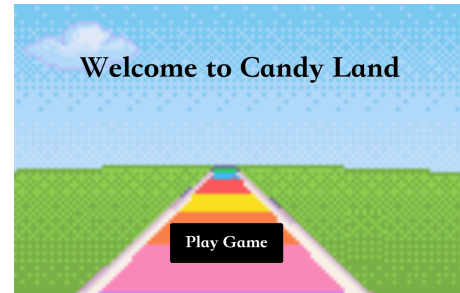
The user stories emphasize the main concepts of developing a simple, fun game for all ages. The user stories covered a wide range of demographics including a young child, a single

mother, and a professional gamer. Although the three primary design priorities are important to everyone, the Use Case Diagram more specifically shows which user persona aided each.



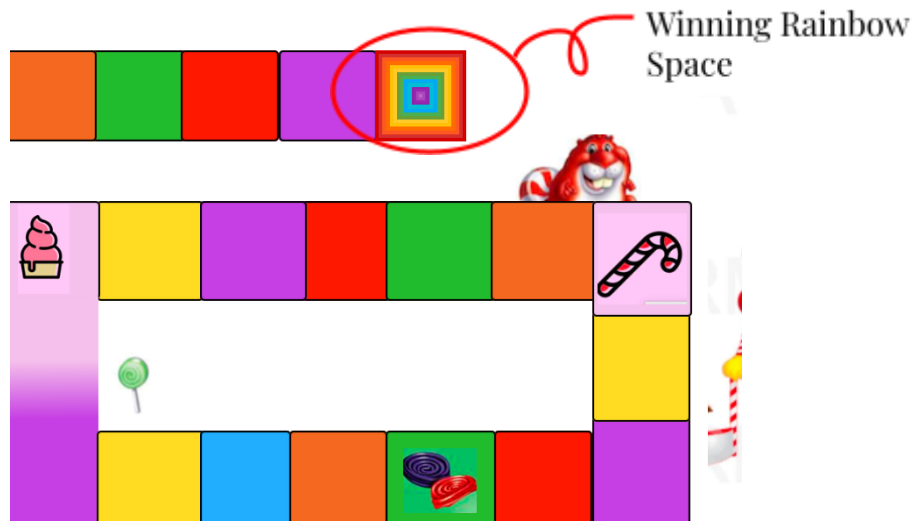
Instructions

When a new game is started, four players are automatically created and assigned to a gingerbread game pawn. It is up to the player's to decide who is who. The game begins when the play button is selected and guides the users to the game board screen.



When it is a player's turn, they must click the "Pick a Card" button which turns over the next card in the card deck. Based on the given card, the player must click the correct next position to move the pawn.

In a consistent order, the four players continue this cycle. The end of the game comes when a player reaches the final rainbow space. The winner is the first to do so.



There are many obstacles and advantages along the way. Three licorice spaces are located throughout the board. When a player lands on a licorice space, their next turn is skipped. Two bridge spaces are also present. When a player lands on one, the pawn travels forward over the

bridge to the connected space. An element that may help or harm a player are character cards which send players to their corresponding space. Character spaces are all over the board, therefore, they have the power to send a player back to the beginning or very close to the end.

