Software Design Document

Apologies

Thomas Cuevas

Benjamin Greenfield

Bruce Johnson

Rory O'Kane

Denisa Qori

March 13, 2014

Revision 0

Revision History

Name	Date	Reason for Change	Version
Rory O'Kane, Benjamin Greenfield, Denisa Qori, Bruce Johnson, Thomas Cuevas	March 6, 2014	Initial Version	0

Table of Contents

1. Introduction

1.1. Purpose

The purpose of this document is to describe the implementation of the Apologies system, as describe in the Apologies SRS. The Apologies system is designed to be a virtual clone of the popular Hasbro board game Sorry!.

1.2. Scope

This document describes the implementation details of the Apologies system. The system contains two major functions. The first is setting up the players, and the second is playing the game. This document will not cover testing of the software system.

1.3. Glossary

SRS : Software Requirements Specification, a complete description of the behavior of the system to be developed.

Player : Any user in the system who is playing the Apologies game.

Pawn: A player's pieces. These pieces are moved around the board during a player's turn. Each player has four pawns.

Safe Zone: A series of tiles next to each player's Home where pawns cannot be targeted by an 11 or a My Apologies card.

Home: The end goal of the game. A player wins the game when he or she moves all of his or her pawns to the location labeled Home on the game board.

Start: The start location for all pawns.

Square: A tile on the game board. A square can only be occupied by one pawn at a time.

Slide: A region on the game board designated by a triangle tile. If a pawn lands on the start of a slide that is not the same color as the pawn, then the pawn may move to the end of the slide.

2. Design Overview

2.1. Problem Description

While many clones of checkers and chess exist, the board game sorry has not been implemented for the home computer since 1998. Additionally, in the 1998 edition, no network multiplayer was supported, allowing only for players to take turns side by side. Apologies will provide a more modern version of the game sorry, and in future releases will contain network multiplayer.

2.2. Technologies Used

The Apologies system will take keyboard and mouse input from the user's computer. The software will be able to run on any computer that supports the Java Virtual Machine.

2.3. System Architecture

The component diagram in Figure 1 depicts the high level system architecture. The Apologies system is built following the Model View Controller architecture, and the three distinct components are modeled below.

- **Model -** Contains classes that store the player data, and data pertaining to the state of the current game.
- **View -** Contains classes that display the graphical user interface.
- **Controller -** Contains classes that verify and operate on the data stored in the model component.

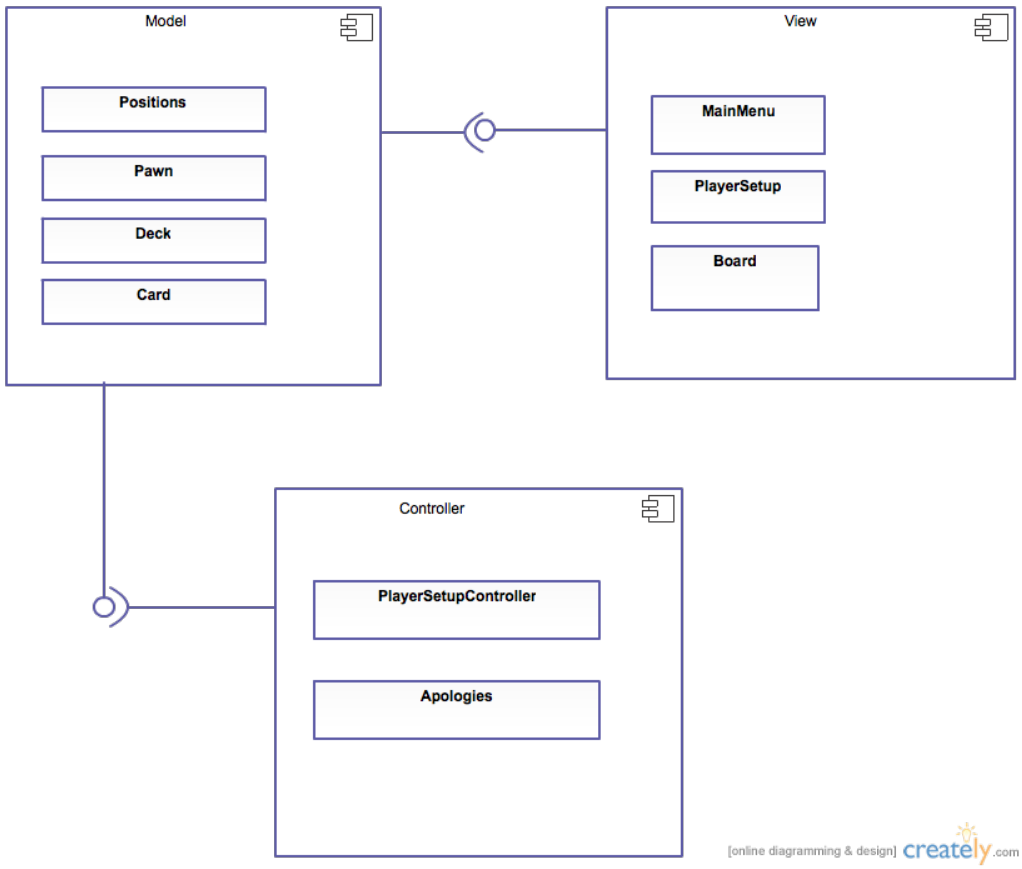

Figure 1. Apologies Component Diagram

2.4. System Operation

2.4.1. Setting up a game

The sequence diagram depicted in Figure 2 depicts the typical flow of events when a user is creating a new Apologies game.

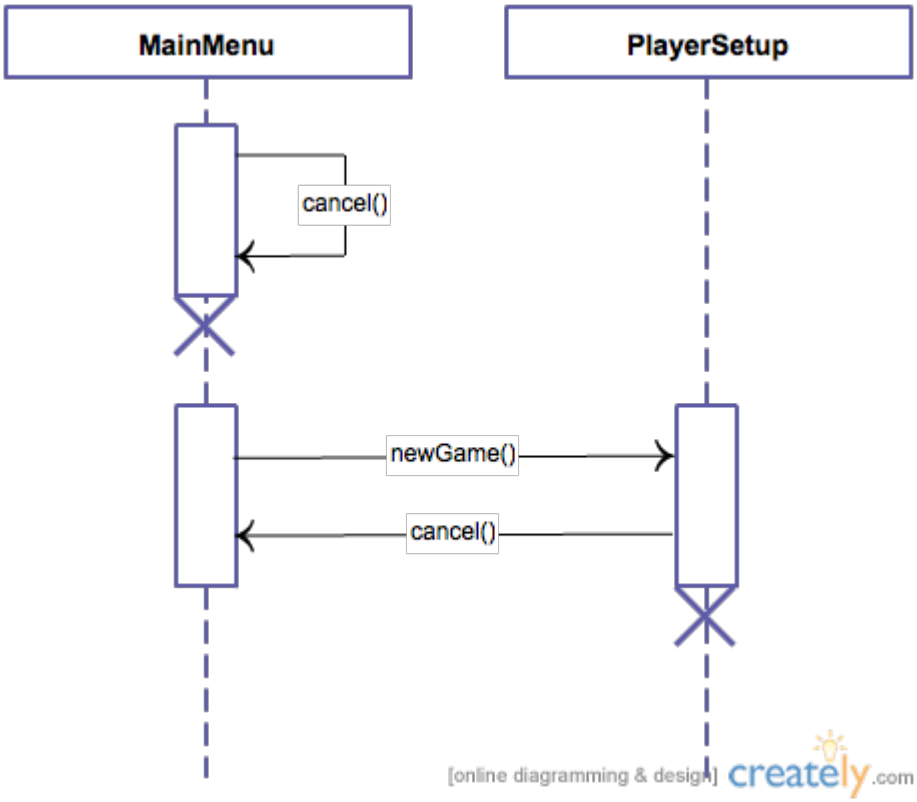

Figure 2. New Game Sequence

2.4.2 Player Setup

The sequence diagram shown in Figure 3 depicts the typical flow of events when a user is setting up players before starting a new game.

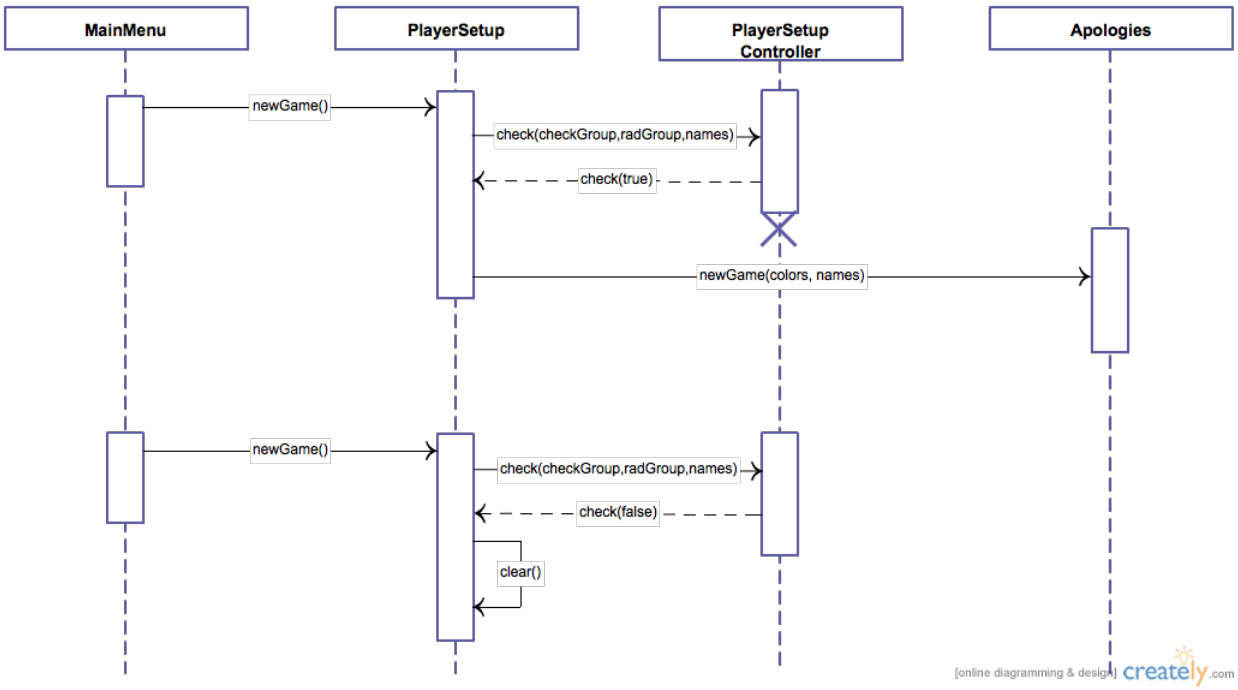

Figure 3. Player Setup Sequence

2.4.3. Taking a Turn

The sequence diagram depicted in Figure 4 depicts the typical flow of events after a game has been started and a player is taking his or her turn.

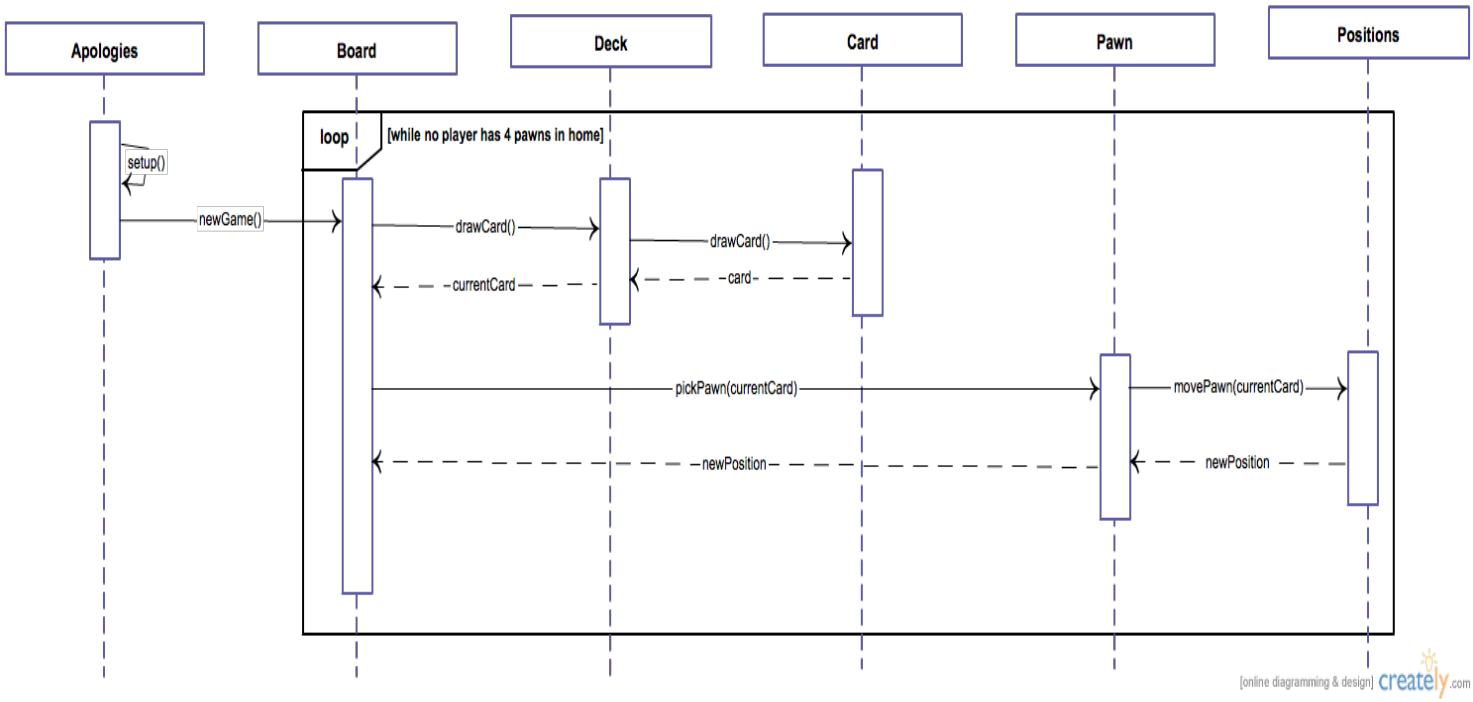

Figure 4. Take Turn Sequence

2.5. System Deployment

Figure 5 depicts the deployment requirements for the Apologies system.

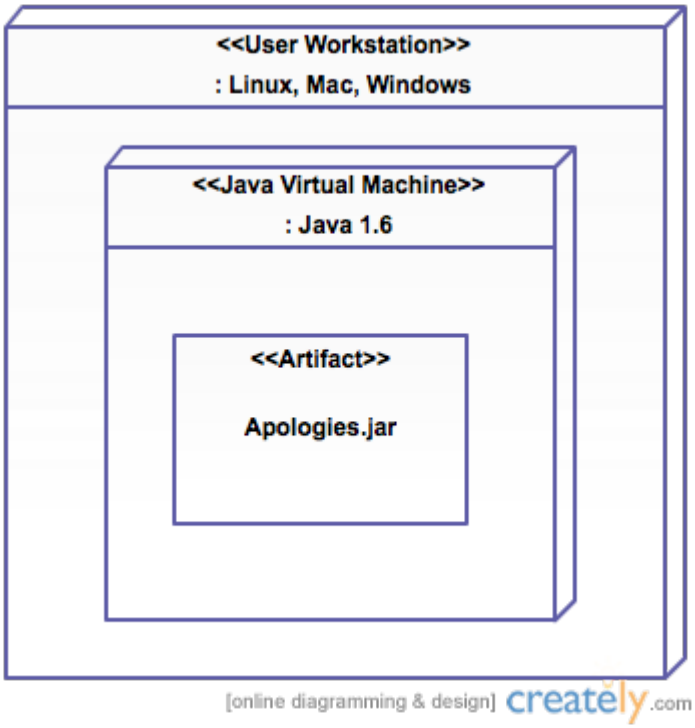

Figure 5. Apologies Deployment Diagram

3. Class Modeling

Figure 6 depicts the UML class model of the Apologies system.

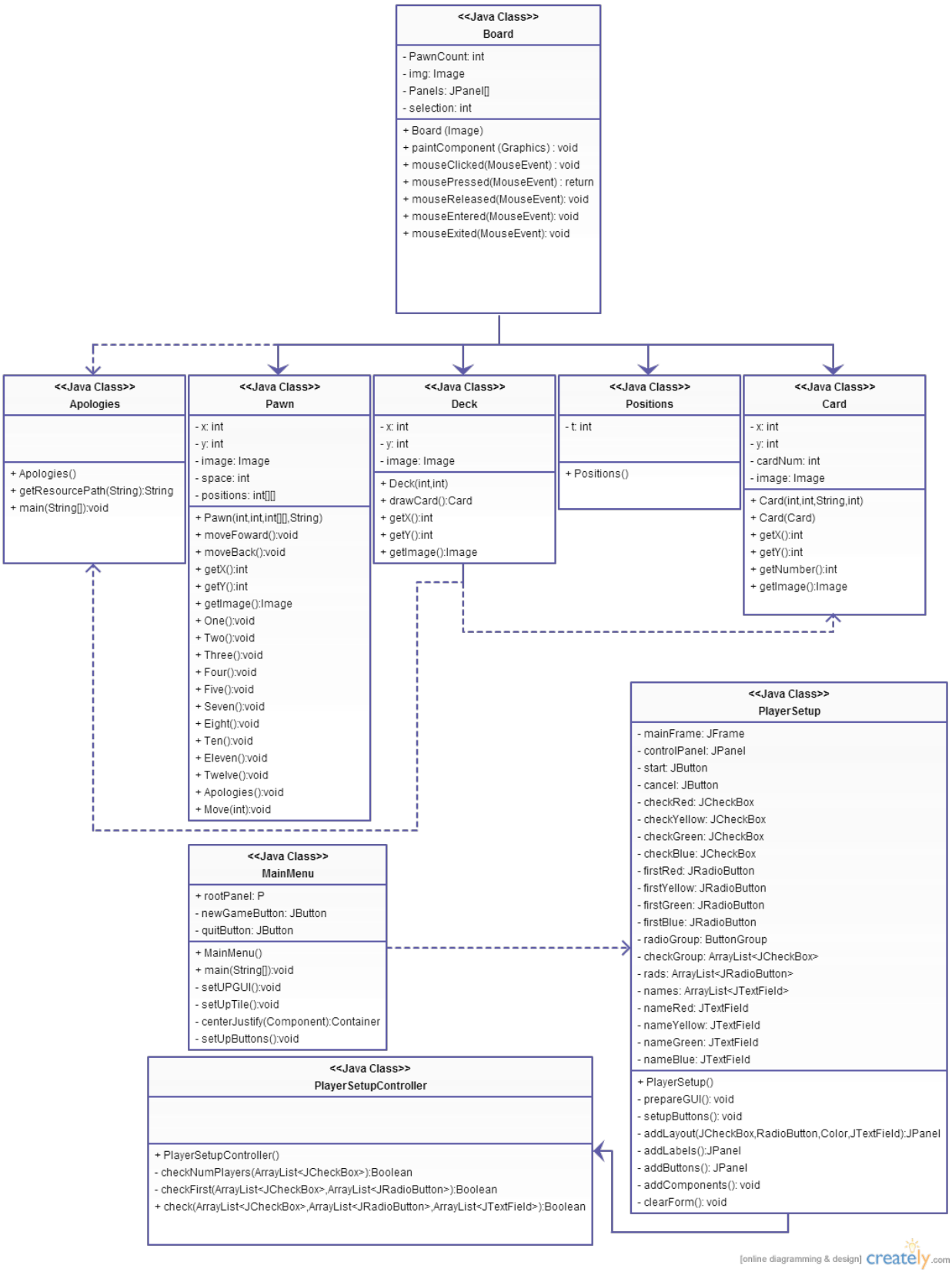

Figure 6. Apologies UML Class Diagram

4. Board State

Figure 7 depicts the state of the board as the Apologies game is being played.

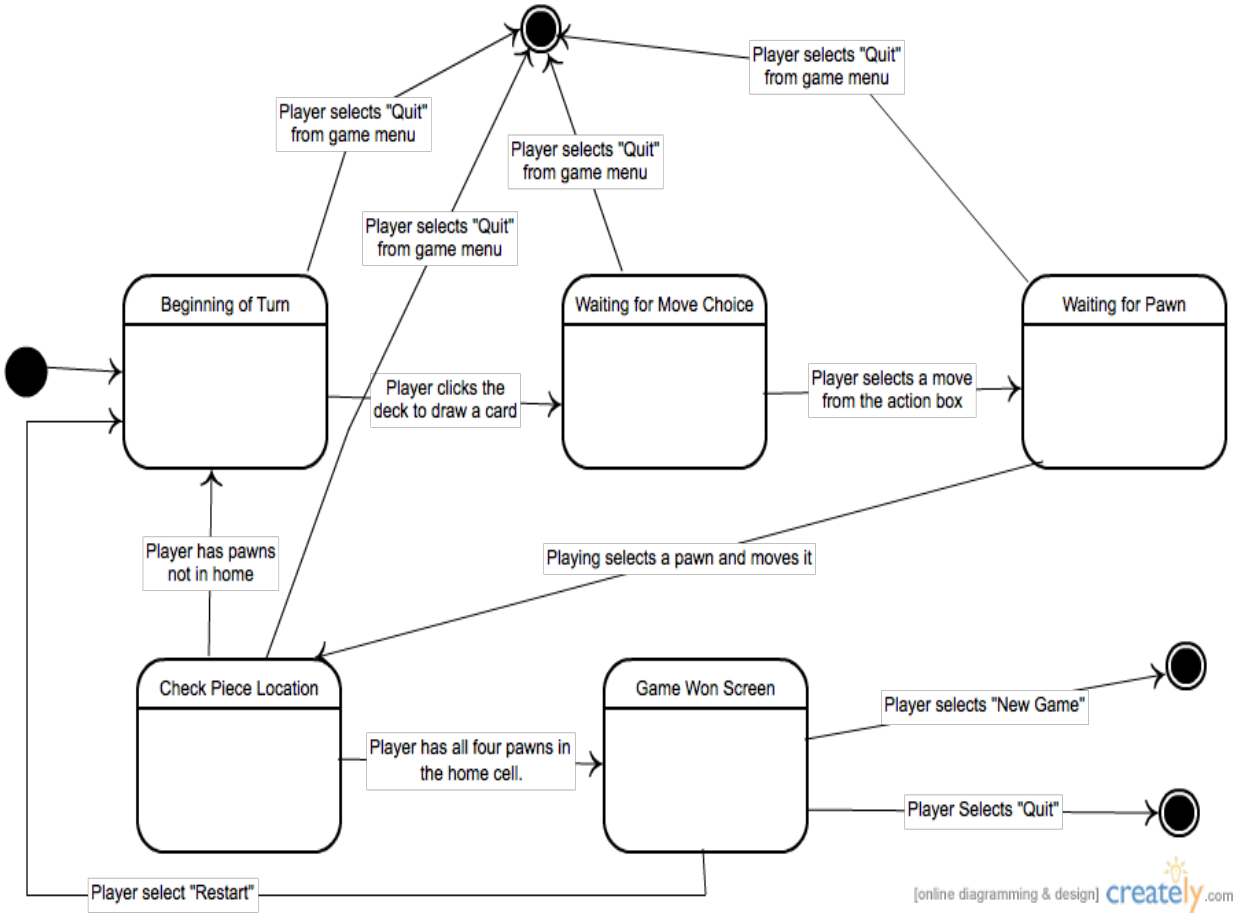

Figure 7. State diagram of board

5. Traceability

Requirement	Description	Design Reference
0100	Select Number of Players	§2, Fig. 3
0200	Select First Player	§2, Fig. 3
0300	Restart game with same settings	§4, Fig. 7
0400	Restart game with new settings	§4, Fig. 7

0500	Draw a card	§2, Fig. 4
0600	Move a pawn	§2, Fig. 4
0700	Quit game	§4, Fig. 7