Software Requirements Specification

Apologies
Version 1.0
2-07-2014

Prepared By:
Thomas Cuevas, Benjamin Greenfield,
Bruce Johnson, Rory O'Kane,
Denisa Qori

Drexel University

Revision History

Name	Date	Reason for Change	Version
Rory O'Kane, Benjamin Greenfield, Denisa Qori, Bruce Johnson, Thomas Cuevas	February 6, 2014	Initial Version	1.0

Table of Contents

1. Introduction

1.1. Purpose

This document gives the requirements specification for the Apologies program. It provides a high-level description of the behavior of the system, functional and non-functional specifications, projected future modifications, and definitions for several technical terms used throughout the document.

The Apologies program is a virtual board game that mimics the popular board game Sorry. Apologies is played with two to four players. The game is played continuously until one player wins or the game is ended.

1.2. Technical Term Definitions

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. Overall Description

2.1. Product Perspective

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.

2.1.1. System Interfaces

The Apologies software interacts with clients and a text file for saving games.

Client The apologies software has an interface that accepts input from the user, and translates it to the appropriate commands.

Text File In future releases the software will have an interface that allows it to write the current state of the board to a text file, and to read from that text file to enable the saving and loading of games in progress.

2.1.2. User Interface

The Apologies software contains a convenient user interface for the user, described in more detail in section 3.2. The interface is included with the main game client and requires no installation. Players interact with the interface by clicking on a graphical representation of a deck to draw cards, selecting radio buttons to select move options where applicable, and by clicking on tokens representing pawns to confirm their move.

2.1.3. Hardware Interfaces

The Apologies software requires the following to be run:

- A computer capable of running the Java Virtual Machine.
- A keyboard and mouse for user input.
- A hard drive for installing and saving/loading games.
- For network play, an Internet connection with an IP address.

2.1.4. Software Interfaces

The apologies software only requires an external software interface for network play.

Client The Apologies software connects the clients via a TCP/IP direct connection.

2.2. Product Functions

The Apologies software will provide the following functions

1. A clone of the Hasbro board game Sorry!, with all its game play and rules.
2. Support for 2-4 players for either side by side or network play.
3. Saving and loading of games in progress.

2.3. User Characteristics

Users of Apologies need no prior experience with the sorry board game, as the rules will be contained in a manual along with the game. A user must only be able to obtain his or her IP address in the case of wanting to partake in network play, as well as the IP addresses of the other players.

3. Specific Requirements

Priority Level	Description
Priority 1	This is the most important priority level. These items should be considered vital. These tasks are necessary for the success of the application.
Priority 2	Priority two features are secondary features that we would like to see added but are not required

	for this release.
Priority 3	Priority three tasks are expected to be completed in future releases and not necessarily essential for the current product.

3.1.Functional Requirements

0100 The user shall be able to select the number of players. **Priority 1**

0200 The user shall be able to choose which player goes first. **Priority 1**

0300 When a game is over, the user shall be able to start a new game with the same settings. **Priority 1**

0400 When a game is over, the user shall be able to start a new game with different settings. **Priority 1**

0500 The user shall be able to draw a card. **Priority 1**

0510 When a card has multiple move options, the user shall be able to pick which one he or she will take. **Priority 1**

0600 The user shall be able to move a piece. **Priority 1**

0700 The user shall be able to quit a game. **Priority 1**

0800 The user shall be able to save a game. **Priority 3**

0900 The user shall be able to load a saved game. **Priority 3**

1000 The user shall be able to play a game with AI opponents. **Priority 2**

1100 The user shall be able to play a game with other users through a network connection. **Priority 2**

Non-functional Requirements

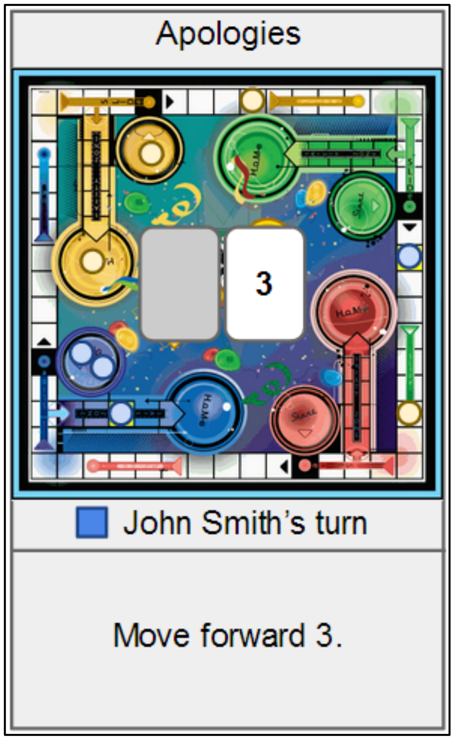
3.1.1. Compatibility Requirement

The system shall execute on any system that supports Java VM.

3.2. User Interface

3.2.1. Interface Requirements

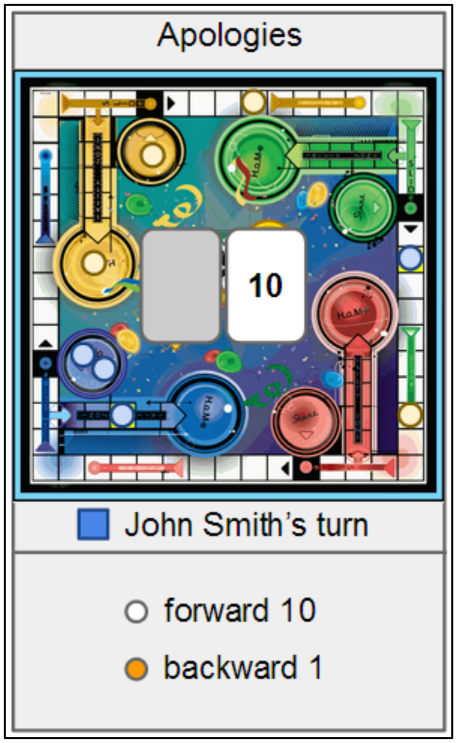
The system will provide a graphical user interface, which can be seen below. The image on the left shows that it is John Smith's turn and card three has been drawn. At the bottom of the GUI in Fig.1, John Smith is prompted to select a pawn to move forward three spaces. Fig.2 shows it is John Smith's turn and card ten has been drawn. Here, John Smith has the option of either moving forward ten spaces or moving backwards one space. When John Smith clicks a pawn, the move corresponding to the radio button that is pressed will be the move that is applied to that pawn.


Apologies

🟦 John Smith's turn

Move forward 3.

Figure 1


Apologies

🟦 John Smith's turn

○ forward 10
● backward 1

Figure 2

3.2.2. Main Menu

As soon as the user starts the game, the main menu will appear. This interface will allow the player to select either to start a new game, or exit. An example of such a scenario is represented below, in Fig.3.

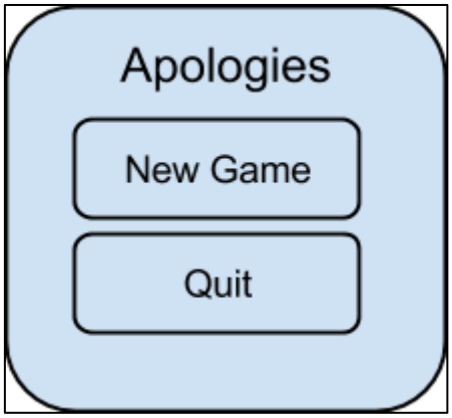
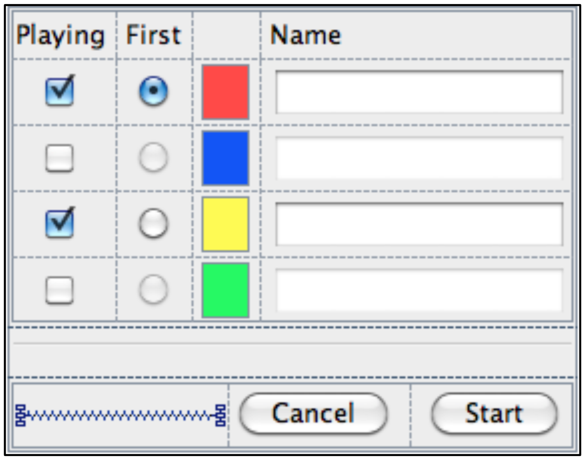

Apologies

Figure 3

3.2.3. Player Setup Screen

After the player has selected the option of starting a new game from the Main Menu, he will be prompted to enter his name, the names of the other players, as well as select one of the four predetermined colors for himself. Whichever color the user selects for himself will go first on the board, with the other colors continuing in a clockwise direction. The user also chooses how many players will be playing each round. This is a number between two and four inclusive. An example of the Player Screen is illustrated in Fig.4. According to this scenario, the user selected the Red Color to go first. The player will be playing against another player, whose color on the board will be yellow.



Playing	First		Name
☑	●	🟥	
□	○	🟦	
☑	○	🟨	
□	○	🟩	

Figure 4

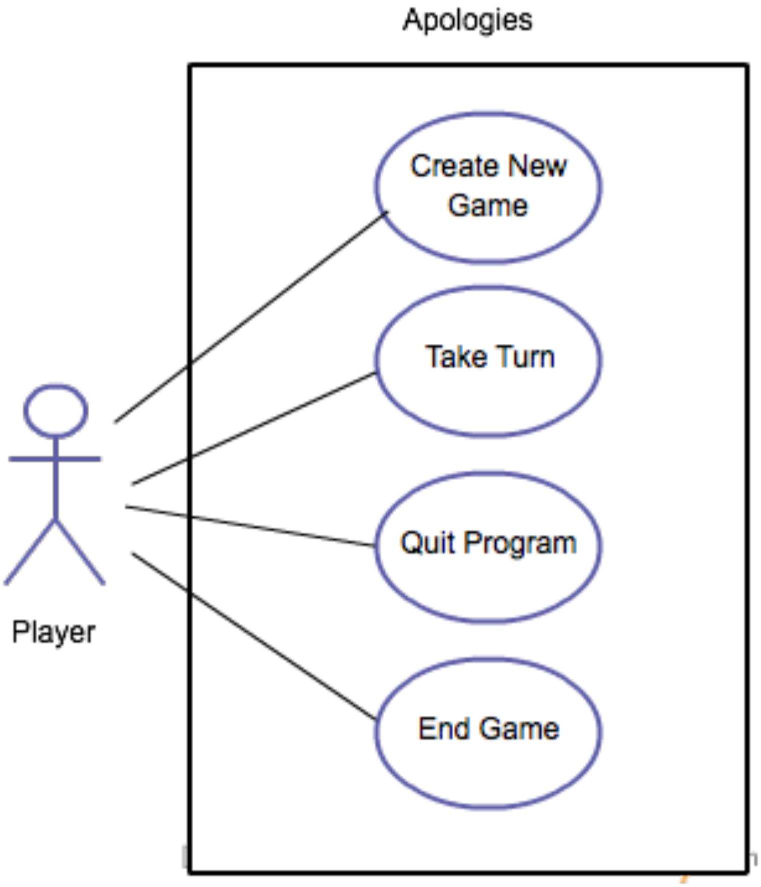
4. System Evolution

In future releases of Apologies it is expected that the game play and player options will be expanded.

The player options will be expanded to allow for AI players, as well as allowing for networked multiplayer.

Game play will be expanded to allow for keyboard shortcuts, sound effects, additional animations, and additional rule sets.

5. Use Case Diagram



6. Use Case Specification

6.1. Use Case: Create New Game

6.1.1. Brief Description

From the main menu a player will be able to create a new game to be played with between two and four players.

6.1.2. Actor Brief Description

A player is any user on the system that is playing the Apologies game.

6.1.3. Preconditions

A player must have the Apologies program running.

6.1.4. Basic Flow of Events

1) The use case begins when a player clicks the New Game button from the main menu. [S-1]
2) The player is taken to the player setup screen. [S-2]
3) The player is brought to the game board screen. [S-6]
4) The use case ends.

6.1.5. Alternative Flows

6.1.5.1. Quit

If in step 1 the player clicks Quit instead of New Game, then

1) The program is exited.
2) The player must restart the program if they wish to play again.
3) The use case resumes at step 1.

6.1.5.2. Not Enough Players

If in step 6 the player clicks Start but only checked one color, then

1) A dialog box appears notifying the player not enough colors were checked.
2) The use case resumes at step 1.

6.1.5.3. No Names

If in step 6 the player clicks Start but names were not entered next to every checked color, then

1) A dialog box appears notifying the player that names were not entered for every color.
2) The use case resumes at step 1.

6.1.6. Sub Flows

6.1.6.1. [S-1]: New Game

The player clicks the new game button from the main menu and is brought to the player setup screen.

6.1.6.2. [S-2]: Pick Player Colors

The player clicks a check box next to a color to indicate which color piece they want.

6.1.6.3. [S-3]: Select First Player

The user selects one player to go first.

6.1.6.4. [S-4]: Enter Names

The player enters a name in the text field next to each checked color.

6.1.6.5. [S-5]: Check Number of Players

The system checks to make sure at least two colors are chosen.

6.1.6.6. Check Names

The system checks to make sure there is a name in the text field next to each active player.

6.1.7. Post Conditions

Players are brought to the game board screen. The game board displays the name of the player who was selected to go first.

6.2. Use Case: Quit Program

6.2.1. Brief Description

When any window is open, closing the window will immediately exit the program, discarding any game in progress or current data entered.

6.2.2. Preconditions

A player has successfully set up a new game and is at the game board screen.

6.2.3. Basic Flow of Events

1) The use case begins when a player clicks the close button on an open window.
2) The program exits immediately, closing all windows.
3) The use case ends.

6.2.4. Post Conditions

The application is closed.

6.3. Use Case: End Game

6.3.1. Brief Description

From the game board screen the player will be able to open a drop down menu and exit the current game.

6.3.2. Actor Brief Description

A player is any user on the system that is playing the Apologies game.

6.3.3. Preconditions

A player has successfully setup a new game and is at the game board screen.

6.3.4. Basic Flow of Events

1) The use case begins when a player selects the Game menu while playing a game.
2) The player clicks the Exit Game button from the drop down menu.
3) The player is brought back to the main menu.
4) The use case ends.

6.3.5. Post Conditions

The user is brought back to the main menu.

6.4. Use Case: Take Turn

6.4.1. Brief Description

From the game board screen a player will be able to draw a card, select a move, and move a piece around the game board.

6.4.2. Actor Brief Description

A player is any user on the system that is playing the Apologies game

6.4.3. Preconditions

A player has successfully setup a new game and is at the game board screen. The player's name is displayed above the deck, indicating it is his or her turn.

6.4.4. Basic Flow of Events

1) The use case begins when a player draws a card from the deck [S-1].
2) The player sees the value of the card and a list of possible moves [S-2].
3) The player selects a move from the list.
4) Game pieces that can be moved are highlighted [S-15].
5) The player selects the piece they wish to move, finalizing their turn [S-16].
6) The use case ends when the next player's name is displayed above the deck, indicating their turn [S-17].

6.4.5. Sub flows

6.4.5.1. [S-1]: Draw Card

The player draws a card from the face down deck, and the drawn card is displayed face-up, next to the deck.

6.4.5.2. [S-2]: List Moves

Moves associated with the card drawn are displayed to the player. [S-3] – [S-14]

6.4.5.3. [S-3]: 1 Card

In the action bar, the player sees two options; move a piece from start, or move one pawn forward one space.

6.4.5.4. [S-4]: 2 Card

In the action bar, the player sees two options; move a piece from start, or move one pawn forward two spaces. Player also gets another turn.

6.4.5.5. [S-5]: 3 Card

In the action bar, the player sees one option; move one pawn forward three spaces.

6.4.5.6. [S-6]: 4 Card

In the action bar, the player sees one option; move one pawn backwards four spaces.

6.4.5.7. [S-7]: 5 Card

In the action bar, the player sees one option; move one pawn forward five spaces.

6.4.5.8. [S-8]: 7 Card

In the action bar, the player sees two options; move one pawn forward seven spaces, or split seven spaces between two pawns.

6.4.5.9. [S-9]: 8 Card

In the action bar, the player sees one option; move one pawn forward eight spaces.

6.4.5.10. [S-10]: 10 Card

In the action bar, the player sees two options; move one pawn forward ten spaces, or move one pawn backwards one space.

6.4.5.11. [S-11]: 11 Card

In the action bar, the player sees two options; move one pawn forward eleven spaces, or switch one of their pawns with an opponent's pawn.

6.4.5.12. [S-12]: 12 Card

In the action bar, the player sees one option; move one pawn forward twelve spaces.

6.4.5.13. [S-13]: My Apologies! Card

In the action bar, the player sees one option; move one piece from start and replace it with an opponent's piece, sending the opponent back to start.

6.4.5.14. [S-14]: Check Moves

If a move cannot be completed by any of the player's pieces, then the move is disabled in the list and cannot be selected. If no moves can be completed, or the only moves would cause a player to bump their own piece, then the player loses his or her turn [S-17].

6.4.5.15. [S-15]: Highlight Pieces

After a move is selected from the list, any pieces that can perform the specified move are highlighted. When a highlighted piece is hovered over, its destination space is highlighted in a different color.

6.4.5.16. [S-16]: Finalize Turn

A user selects a highlighted piece, and the piece moves to the destination location.

6.4.5.17. [S-17]: Next Player

After the turn is finalized, play proceeds clockwise and the next player's name is displayed above the deck.