Tic Tac Toe Game Design Document

Rev. A draft 1

James Richey

Mar 03, 2020

CONTENTS

1 Introduction **1**
- 1.1 Purpose of this Document . . . 1
- 1.2 Scope of the Game . . . 1
- 1.3 Overview of this Document . . . 1
- 1.4 Project Objectives . . . 1

2 Gameplay **3**
- 2.1 Rules of Tic Tac Toe . . . 3
- 2.2 One Player Mode . . . 4
- 2.3 Two Player Mode . . . 4
- 2.4 Speedrun Mode . . . 4

3 User Interface **5**
- 3.1 Game Board . . . 5
- 3.2 Controls . . . 7
- 3.3 Menus . . . 7
- 3.4 Screen Flowchart . . . 13

4 Environments **15**
- 4.1 List of Environments . . . 15

5 Additional Considerations **19**
- 5.1 Target Audience . . . 19
- 5.2 Target Platforms . . . 19
- 5.3 Monetization . . . 19
- 5.4 License Compliance . . . 19
- 5.5 Similar Games . . . 19
- 5.6 Future Enhancements . . . 21

6 Technical Design **23**

7 Glossary **25**

Index **27**

CHAPTER

ONE

INTRODUCTION

1.1 Purpose of this Document

This is the game design document for Tic Tac Toe. This document describes in detail the he objectives, requirements, and design considerations of the game providing a central location for this information. This is invaluable for understanding the game's scope, planning the project milestones, and creating the games assets.

Anyone who is involved with the game's development is encouraged to read this document and keep a copy handy.

1.2 Scope of the Game

Tic Tac Toe is a game of strategy where two players, X and O, take turns placing their mark in a 3 x 3 gird. The first player to get three marks in a row, column, or diagonal wins the game.

This version of Tic Tac Toe is an unique take on the classic game. Players of all ages battle the computer or each other in a variety of stunning environments. Each environment tells part of the story of Tic Tac Toe from the past, present, and future. Each environment has a strong visual theme and complementary soundtrack.

Tic Tac Toe runs on Windows, Mac, and Linux and launches Summer 2020.

1.3 Overview of this Document

1.4 Project Objectives

Creation of the Tic Tac Toe game described in this document is part of the Pound of Rust project. This section describes the project's objectives.

1.4.1 Create Tic Tac Toe Game with Rust

The main deliverable of this project is the Tic Tac Toe game for Windows, Mac, and Linux. This project is the follow-up to the Ounce of Rust project[1] that resulted in the creation of a Rust based Tic Tac Toe logic library, `open_ttt_lib`.[3]

In addition to creating a fun game, the project provides more hand on experience using Rust and is a showcase for `open_ttt_lib`.

[1] For details on the Ounce of Rust project see the Ounce of Rust Project Manual[2]

[2] https://j-richey.github.io/project-documentation/ounce-of-rust/

[3] `open_ttt_lib` is available at https://crates.io/crates/open_ttt_lib and source code is hosted at https://github.com/j-richey/open_ttt_lib

1.4.2 Provide Free of Charge and Under an Open Source License

Tic Tac Toe and all future releases of the game are free, open source, and contain no advertisements or trackers. The game is released under a permissive open source license and its code is available from a public repository such as https://github.com/.

Many of today's games casual games are released for free, but include questionable monetization models such as microtransactions, pay-to-win schemes, advertisements, and personal data harvesters. Tic Tac Toe stands apart from these games by respecting player's who choose to spend their valuable time playing the game.

1.4.3 Release by RustConf 2020

Tic Tac Toe's initial release is scheduled to coincide with RustConf 2020 on August 21, 2020. RustConf is the annual Rust developers conference; since Tic Tac Toe is developed in Rust this makes an excellent time to launch a Rust based game.[4]

To help meet the target release date, the initial release of the game might contain a subset of the environments described in this document.

1.4.4 Easily Expandable and Modifiable

Playing Tic Tac Toe in a variety of environments is a large part of what sets this version of Tic Tac Toe apart from others. The game is designed such that developers can easily add new environments. This allows developers to focus their time and efforts creating stunning environments. Additionally, this lowers the barrier of entry for users who are interested in modifying the game. Finally, this allows quick turnaround of future releases of the game.

Automated tools, guides, checklists, and detailed documentation are some ways that can help development speed.

1.4.5 Build Risk Reduction Prototype

The development team creating Tic Tac Toe is new to the Rust programming language and the available Rust libraries for game development. To help mitigate this risk, a throwaway prototype game is created early in the project that explores various technical aspects.

Using the lessons learned from the prototype also helps the development team design a code base that is easily expandable and modifiable per the above objective.

[4] For details on RustConf see their website: https://rustconf.com/

CHAPTER

TWO

GAMEPLAY

2.1 Rules of Tic Tac Toe

Tic Tac Toe is a game of strategy where two players, X and O, take turns placing their mark in a 3 x 3 gird. The rules for Tic Tac Toe used for each game mode are as follows:

1. Play occurs on a board composed of a 3 x 3 grid of squares. The board starts empty with no marks.
2. The first player places their mark in one of the grid's squares. Traditionally, the mark is the letter *X*.
3. The second player places their mark in one of the grid's empty squares. A square that already contains a mark cannot be updated or altered. Traditionally, the second player uses the letter *O* as their mark.
4. Turns alternate between the players until the game is over.
5. The first player to get three of their marks in a line wins the game. That is: they have three marks in a row, column, or diagonally. Examples of winning games are shown in Figure 2.1.

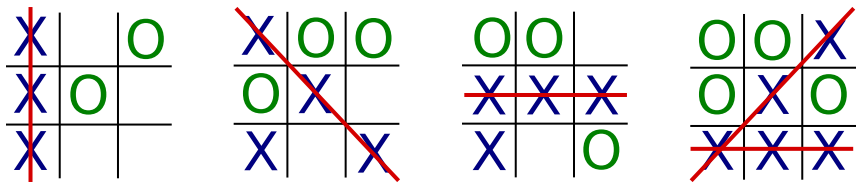

Figure 2.1: Examples of winning Tic Tac Toe games showing player X winning by getting three marks a row, diagonal, and column. The red line shows the squares that contributed to the win. Notice that it is possible to get multiple sets of three marks in a row.

6. The game ends in a draw, known as a cat's game, if no more empty squares remain and a player has failed to get three marks in a line. Examples of cat's games are shown in Figure 2.2.

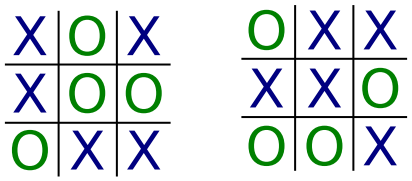

Figure 2.2: Examples of Tic Tac Toe games ending in a cat's game. No player managed to get three marks in a line.

7. The steps above are repeated for a series of games. The starting player alternates between games, that is the second game player *O* gets to make the first move.

2.2 One Player Mode

In one player mode the player battles a computer controlled opponent. There are three difficulty settings: **easy**, **medium**, and **hard**.

Easy difficulty is targeted at players who are new to Tic Tac Toe and/or computer games. The computer picks random squares allowing players to learn the game's controls and rules.

Medium difficulty is for players who have some experience with Tic Tac Toe. The computer provides a challenge to the player but games are still winnable.

At hard difficulty the computer plays almost perfect games. The player must capitalize on rare mistakes made by the computer to win. This is the recommended difficulty for experienced Tic Tac Toe players.

2.3 Two Player Mode

In two player mode two players take turns placing their marks according to the rules described above.

2.4 Speedrun Mode

A speedrun mode provides an additional challenge for experienced players. Players battle a flawless computer opponent through 10 environments completing the games as fast as possible. At the end of the run the total time is displayed along with the previous best times.[5]

The player is disqualified and the run halted if the player loses a game. Since the computer opponent never makes a mistake, each game in the speedrun ends in a cat's game. In other words, each speedrun requires the same total number of moves to complete.

Unnecessary animations are disabled in speedrun mode so they do not get in the way of the speedup gameplay. Additionally, speedrun mode has its own dramatic music that replaces the music tracks of each environment — environments are played so fast there is not sufficient time to appreciate their individual sound tracks.

[5] The time it takes the computer to pick a square is not counted towards the player's time. This ensures times are consistent between slower and faster computers.

CHAPTER

THREE

USER INTERFACE

This chapter describes the user interface of Tic Tac Toe. This includes the main game board, controls, and all game menus.

3.1 Game Board

The game board is where players spend the majority of their time. Additionally, the game loads directly to this view ensuring players get to gameplay as quickly as possible without menus getting in the way.[6] A concept drawing of the game board is shown in Figure 3.1.

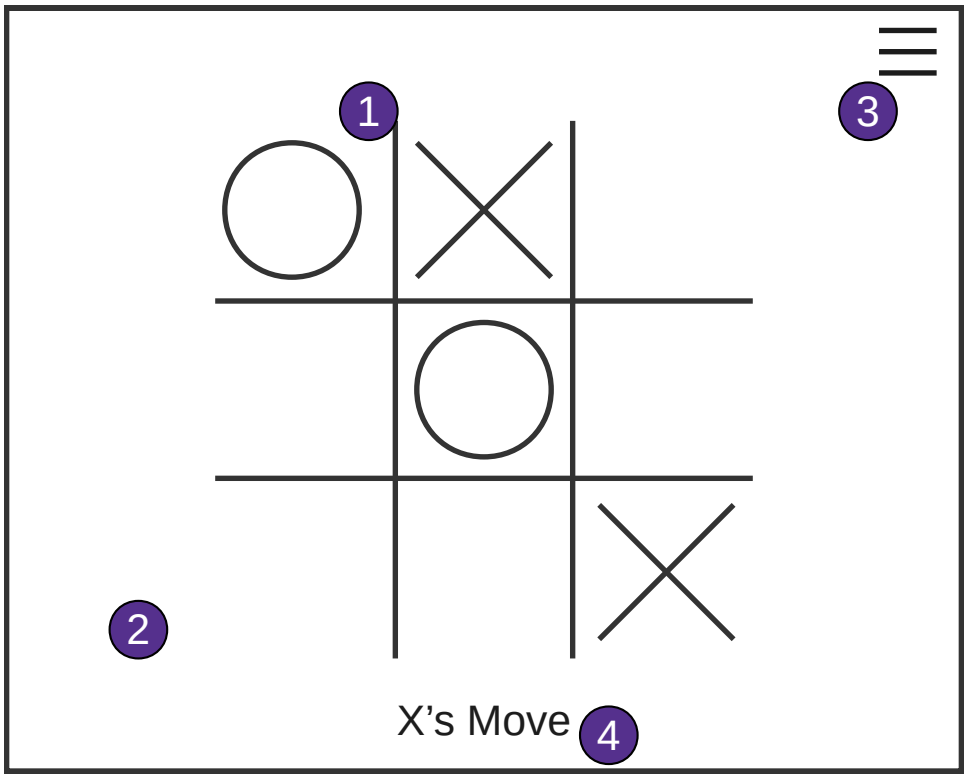

Figure 3.1: Game board concept drawing.

The game board contains the following items of interest:

[6] The loaded game is a single-player game using the last difficulty and player mark settings. The defaults for these are Medium difficulty and X marks.

1. The marks and grid. The appearance of these depend on the current environment being played. However, the marks for all environments use the same center point and have the same selectable hotspot. This ensures consistency between environments when using the mouse.
2. The background of the game board depends on the current environment.
3. The hamburger button opens the *Main Menu*.
4. Status text indicates who gets to make the next move and the outcome of the game. Once the game is over buttons to play the next game or return to the menu also appear in this area. The text is outlined or shaded such that it is visible over any possible background.

A major focus of the game is playing Tic Tac Toe in variety of stunning environments that control how the marks, grid, and background look. Thus, a minimalist approach is used for the game board view. The only UI widgets are a menu button and some status text.

3.1.1 Speedrun

Additional UI widgets are added to the game board to facilitate speedrun mode. Figure 3.2 shows the speedrun gameboard.

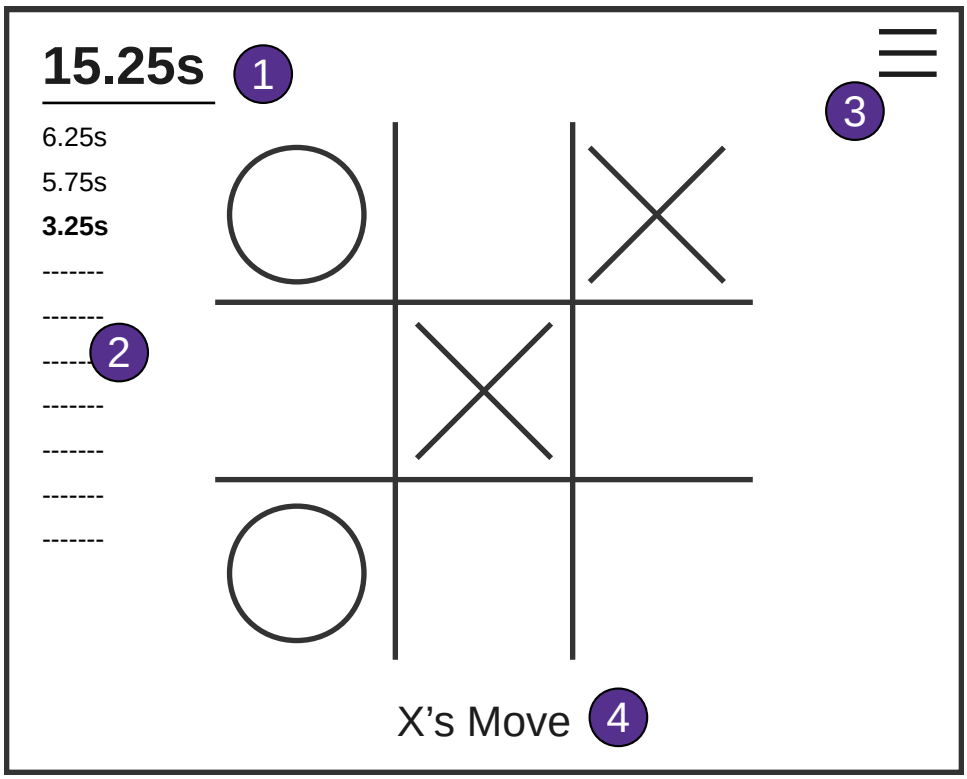

Figure 3.2: Speedrun game board concept drawing.

Items of interest are:

1. The speedrun status display prominently shows the elapsed time of the run. This helps give the player a sense of urgency and lets them see if they are on track to get a best time.
2. Splits for each game are additionally displayed. Dashes or other marks are used for games that have yet to be played. This allows players to quickly visually gauge how many games remain.
3. Opening the game's menu ends the run. The run is disqualified.

4. Status text indicates the current turn. If the player loses a game, the status text notes that the run is disqualified and the player is invited to try the run again or return to the Speedrun menu.[7]

When a game is competed successfully the next game starts immediately allowing players to go as fast as possible through the games.

3.2 Controls

The game can fully played with either a mouse or keyboard. New or casual players may prefer to use a mouse where as speedrun players may prefer to use the keyboard.

3.2.1 Mouse

Mouse left click is used to select free squares and menu press buttons. Right click and other mouse buttons are unused.

3.2.2 Key Bindings

The game supports being played using the keyboard. Figure 3.3 shows the game's keybindings for selecting squares.

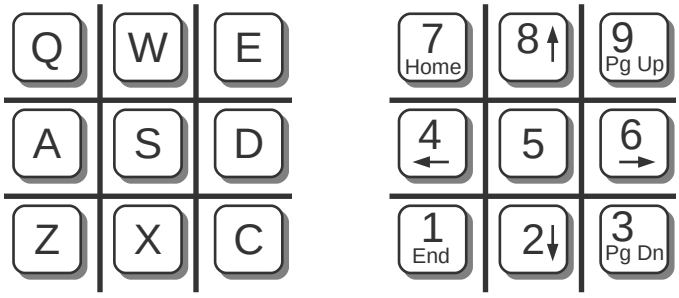

Figure 3.3: Keybindings for selecting squares.

The `numpad` keys are available for right handed players and the `QWE` set of keys are available for left handed players.

The `arrow`, `ESC`, `Enter`, and `Space` keys allow users to navigate the game's menus.

3.3 Menus

The Tic Tac Toe menus allow players to select the various game modes and to customize the game. The *Screen Flowchart* provides details on how the menus and views connect.

Each menu is described in the following sections.

[7] If the player loses a speedrun game, the board remains visible so the player can see where they made mistakes. This allows them to adjust their strategy for next time.

3.3.1 General

Unless otherwise noted, the information in this section applies to all menus.

The menus have translucent or blurred background that show the active environment. This includes any animations or environment FX. Additionally, the environment's sound track continues to play while the menus are open.

3.3.2 Main Menu

The main menu provides a central point for users to navigate to the game's various modes and settings. Figure 3.4 shows the main menu.

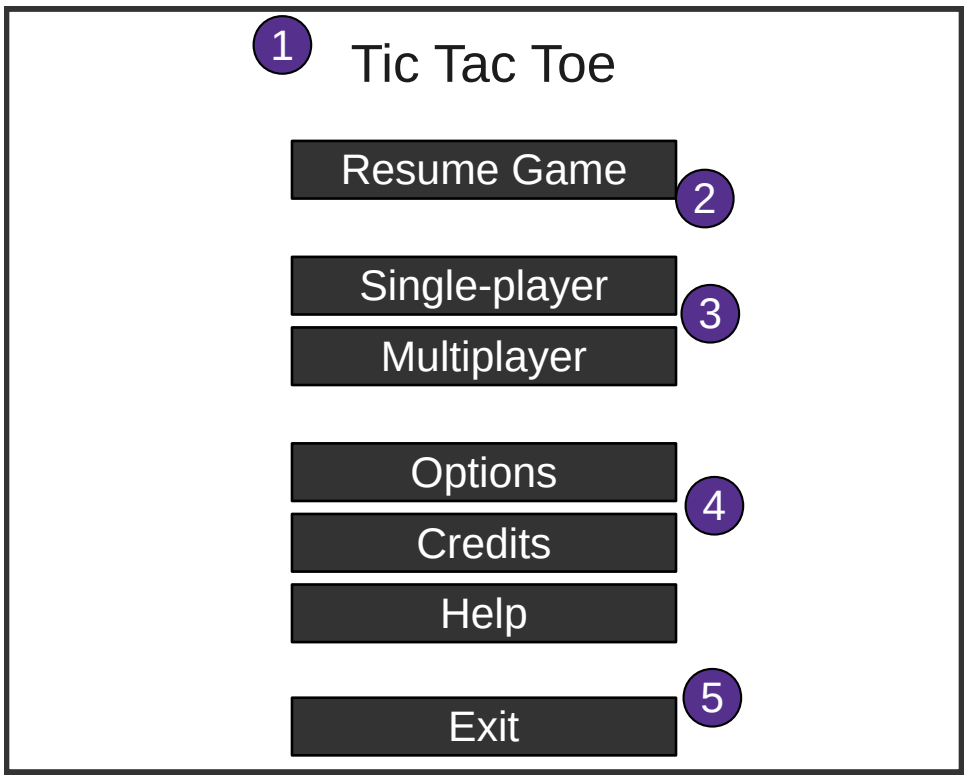

Figure 3.4: Main menu concept drawing.

1. The title of the game is prominently displayed at the top of the menu.
2. The *Resume Game* button is visible if there is a current game in progress. Selecting it closes the menu and returns the player to the game.
3. New game buttons. The *Single-player* button navigates to the while the *Single-player* screen while the *Multi-player* button immediately starts a new multiplayer game.
4. Miscellaneous buttons to open the *Options*, *Credits*, *Help* screens.
5. *Exit* closes the game and returns the user to their desktop.

3.3.3 Single-player

The single-player menu, shown in Figure 3.5, allows players start new single-player games.

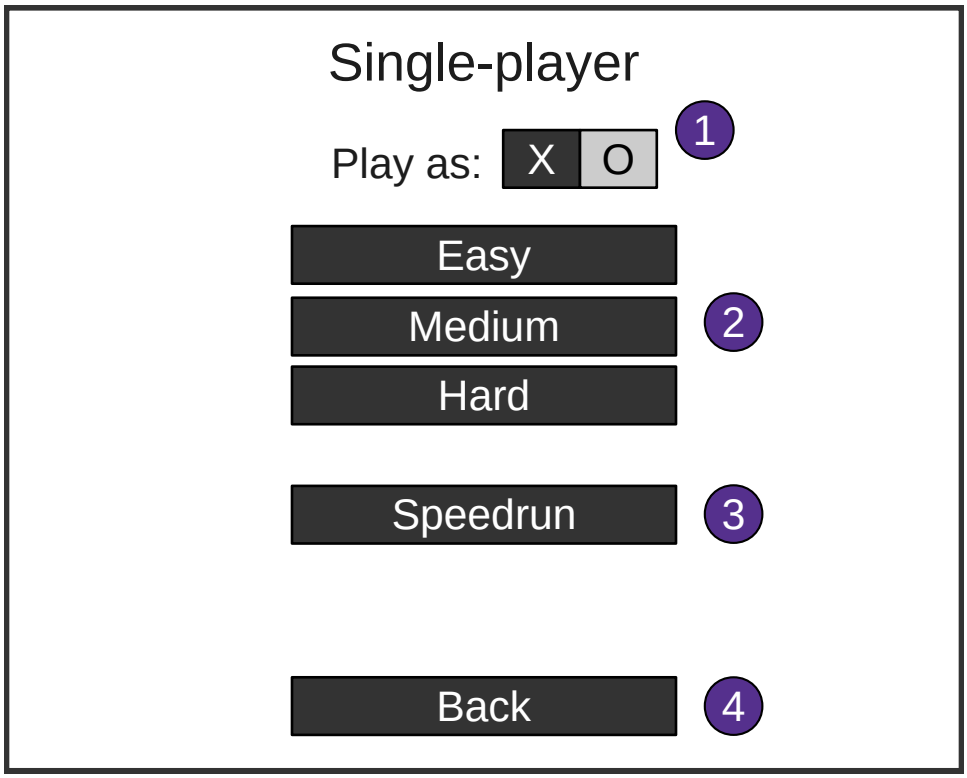

Figure 3.5: Single-player menu concept drawing.

1. The *Play as* selector allows players to select the mark they wish to use throughout the games.
2. The difficulty buttons select the difficulty then start a new single player game. Selecting one of these buttons closes the menu and launches a new single-player game with the requested settings.
3. The *Speedrun* button navigates to the *Speedrun* menu.
4. The *Back* button returns to the main menu.

3.3.4 Speedrun

The speedrun menu allows players to start a new speedrun and view best times of previous runs. Figure 3.6 shows the speedrun menu.

The speedrun menu contains the following items of interest:

1. Instructional text that provides a short overview of the speedrun rules. Once the run is completed this text is replaced with the run's result and invites the player to play again.
2. *Start* begins the run. This navigates to the *Speedrun* game board.
3. Table of previous best times sorted from fastest to slowest.
4. The *Back* button returns to the main menu.

When the speedrun begins, the game board is shown, a prominent three second countdown is show, and dramatic music starts to swell. Once the timer elapses the run begins.

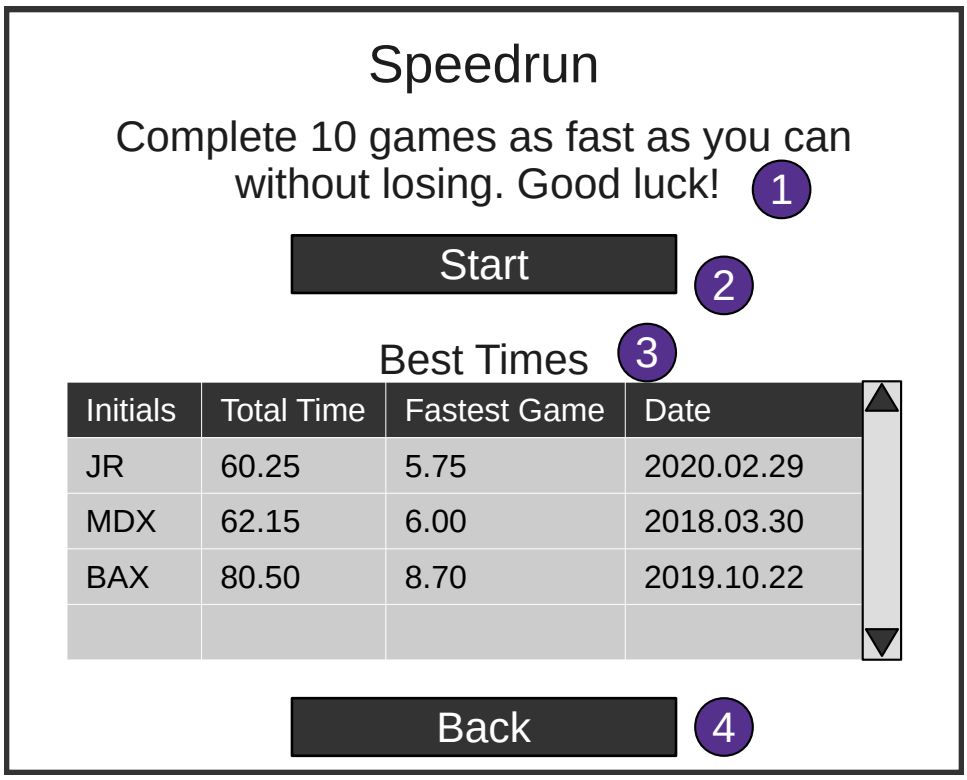

Figure 3.6: Speedrun menu concept drawing.

Once the run is completed the speedrun menu is displayed and shows the result of the run.

If the player gets a new best time the dialog shown in Figure 3.7 is presented to the user.

The best time dialog contains the following items:

1. The speedrun time time.
2. The *Initials* text box allows players to enter their initials so their best time is differentiated from other players that happen to use the same computer. The field remembers the last set of initials entered to save players from having to retype their initials.
3. The *Close* button hides the dialog allowing the speedrun menu to be fully visible.

3.3.5 Options

The options screen contains all of the game's player configurable options. Figure 3.8 shows this screen.

1. Music and Sound FX sliders to control the volume of these items. This allows players to mute some or all of the in-game sounds.
2. *Reset to Defaults* rests all options to their default values.
3. The *Back* button returns to the main menu.

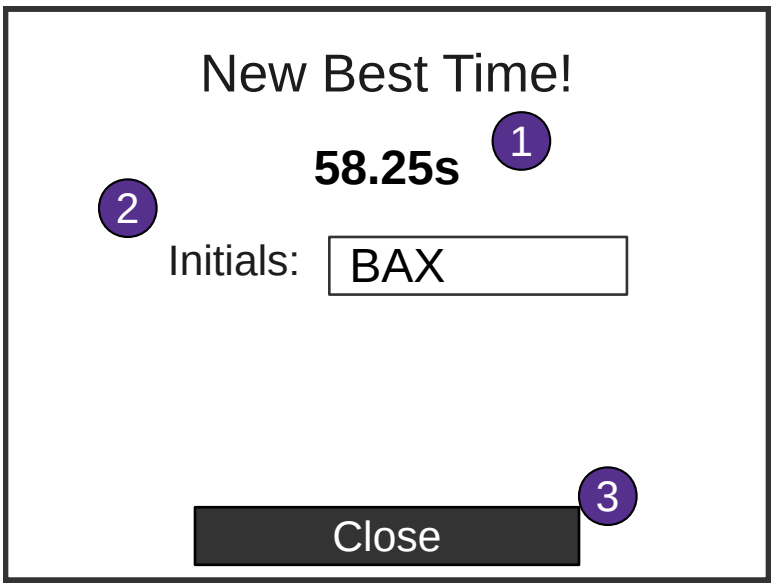

Figure 3.7: Speedrun best time dialog concept drawing.

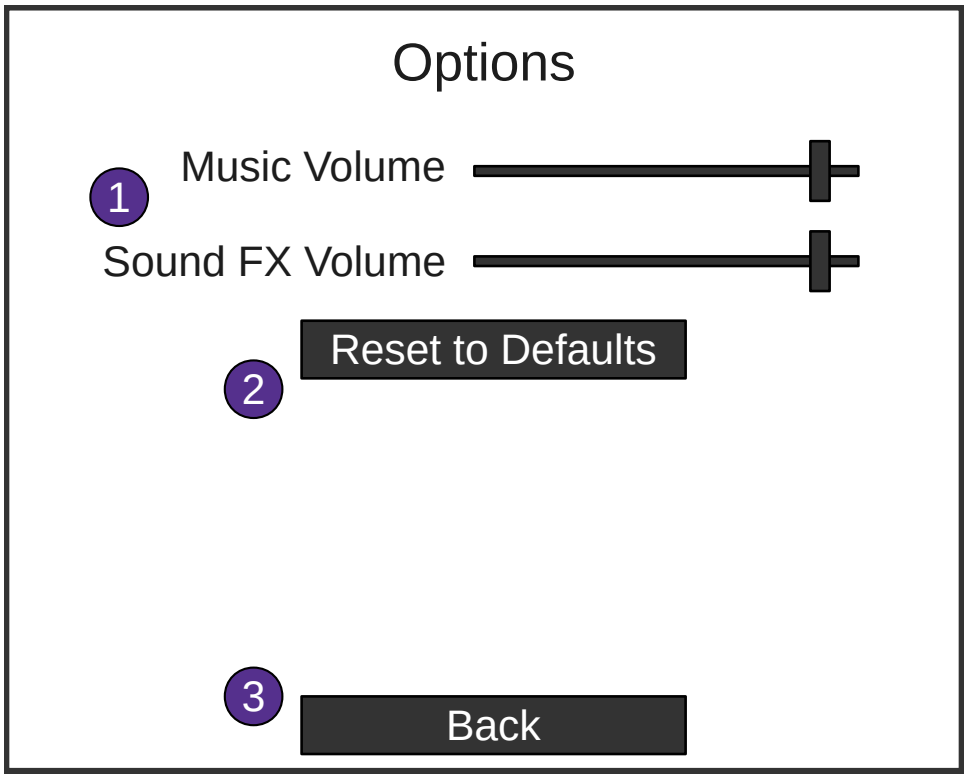

Figure 3.8: Options screen concept drawing.

3.3.6 Credits

The credits screen displays information the game's developers and helps fulfill the *License Compliance* obligations. The credits screen is shown in Figure 3.9.

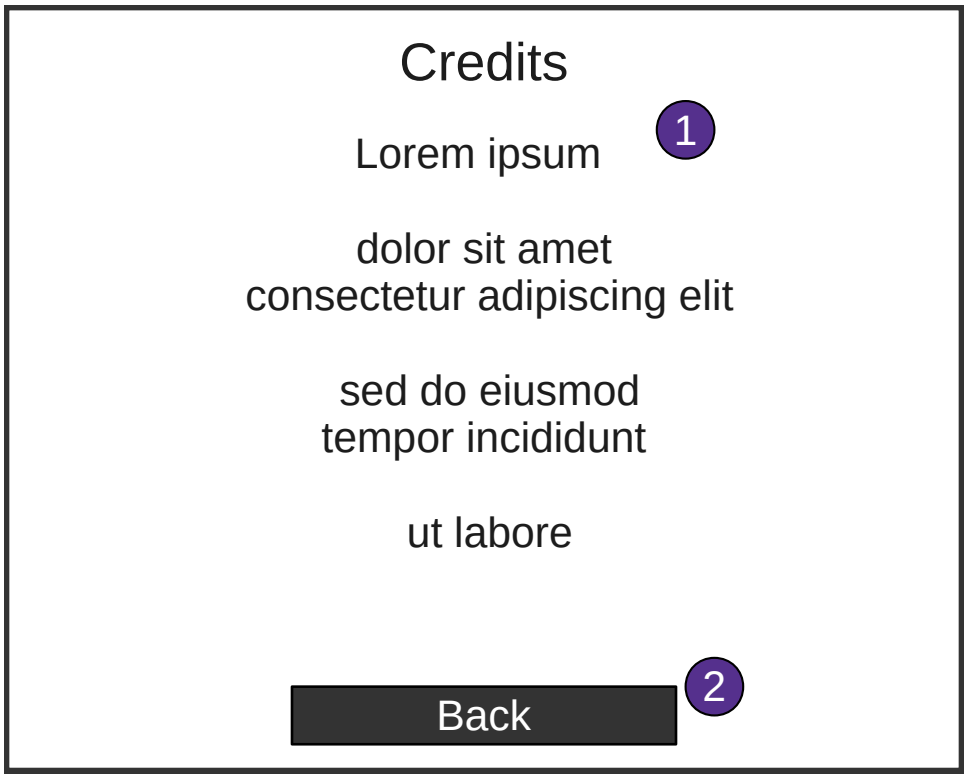

Figure 3.9: Credits screen concept drawing.

The credits screen contains the following items:

1. Scrolling list of developer names, third party assets, and other information about the game.
2. *Back* returns to the *Main Menu*.

The credits screen uses a different background and soundtrack than the other menus. The background consists of one or more Tic Tac Toe games being played in a variety of environments. Each environment is clearly visible — blurring and other effects are not used on this screen. The environments are changed several times per game. This showcases the many environments of the game.

The credits screen has its own sound track. The music and sound FX of the individual environments are not used.

Once all of the credits have played the screen remains open with Tic Tac Toe games being played in the background.

3.3.7 Loading Screen

If necessary for technical reasons, the loading screen provides feedback to the player while assets are loaded. This screen is only shown when the game first loads.

3.3.8 Help

The help screen provides information on how to play Tic Tac Toe, the application version, how to report bugs, and other information. All information is hosted locally; no internet access is required.[8]

3.4 Screen Flowchart

The flow chart in Figure 3.10 visually shows how the screens and menus are connected.

[8] The *Provide Free of Charge and Under an Open Source License* objective mentions not tracking players. Websites often contain trackers, advertisements, and other items that violate this objective.

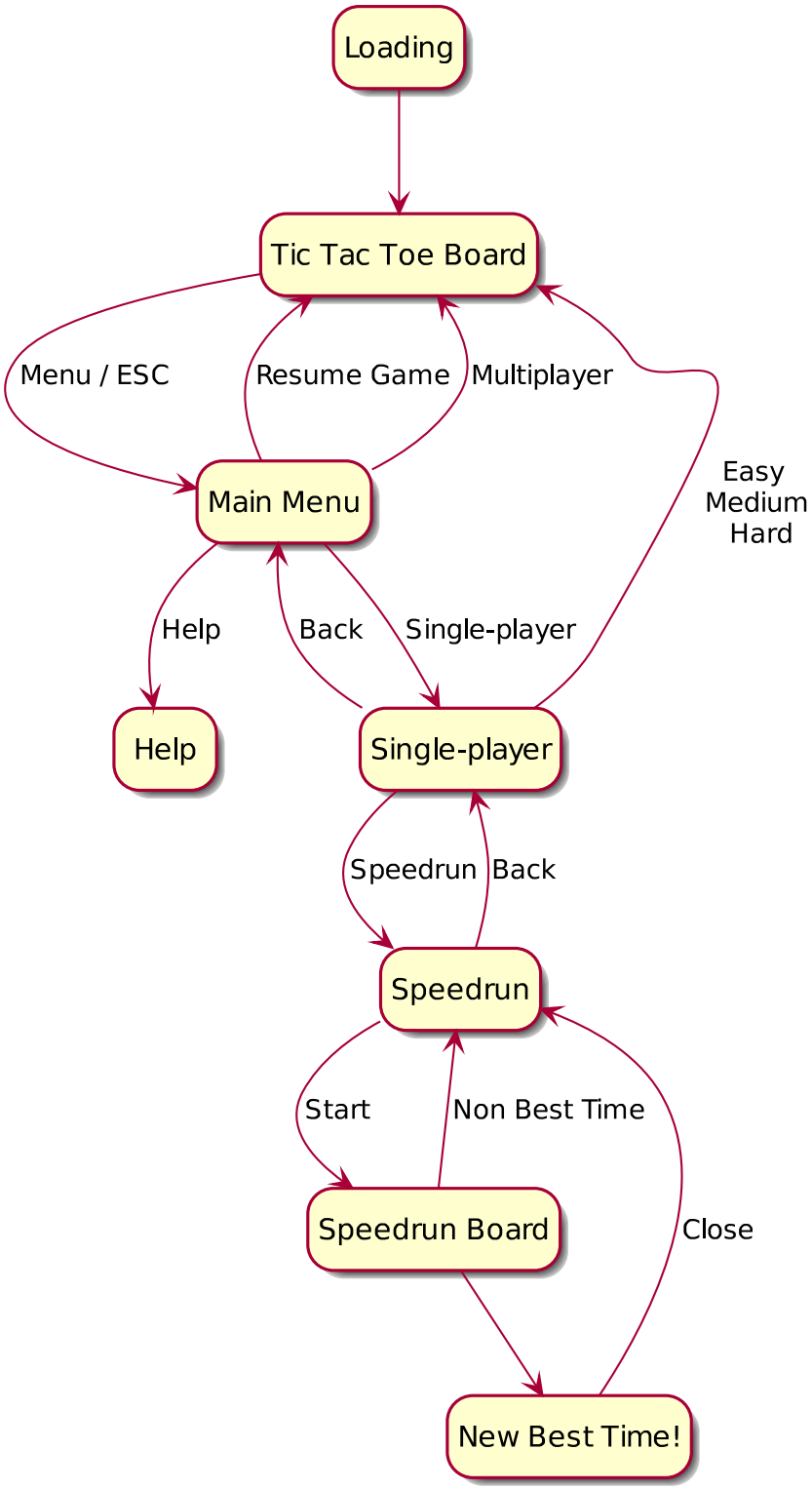

Figure 3.10: Connections between Tic Tac Toe's menus and screens.

CHAPTER

FOUR

ENVIRONMENTS

4.1 List of Environments

4.1.1 Notebook Paper and Pencil

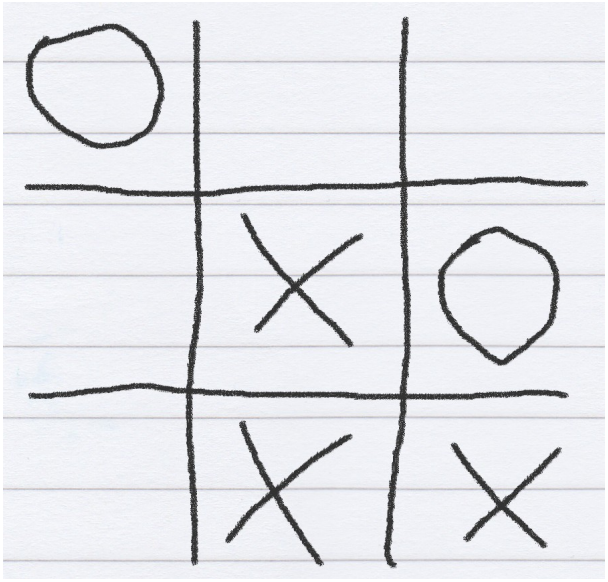

Figure 4.1: Concept art for the Notebook Paper and Pencil environment.

Color Scheme Dark graphite on white or light yellow paper

Music Theme TODO

TODO

Variants

- Notebook paper background
- Engineering paper background

4.1.2 Pen on Scrap Paper

Color Scheme Black or dark blue ink on white paper

Music Theme Gospel / churchy

TODO

Variants

- Blue, black, or red ink for the grid and marks
- Bank deposit slip background
- Store receipt background

4.1.3 Neon Lights

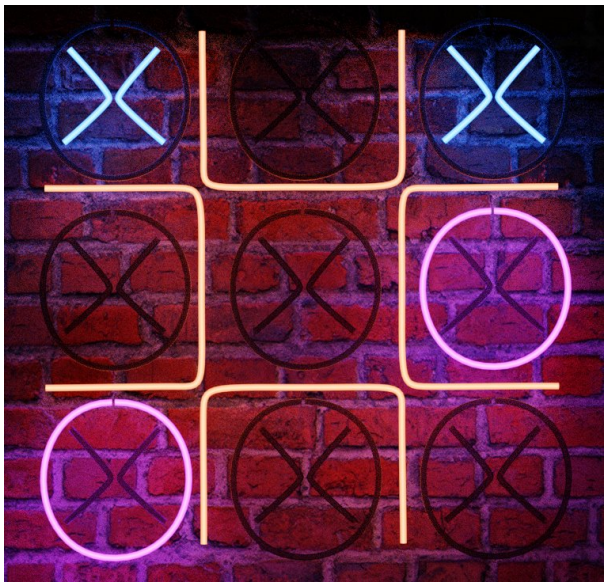

Figure 4.2: Concept art for the Neon Lights environment.

Color Scheme luminescent orange, pink, blue

Music Theme 80's electronic

TODO

4.1.4 Early Computer

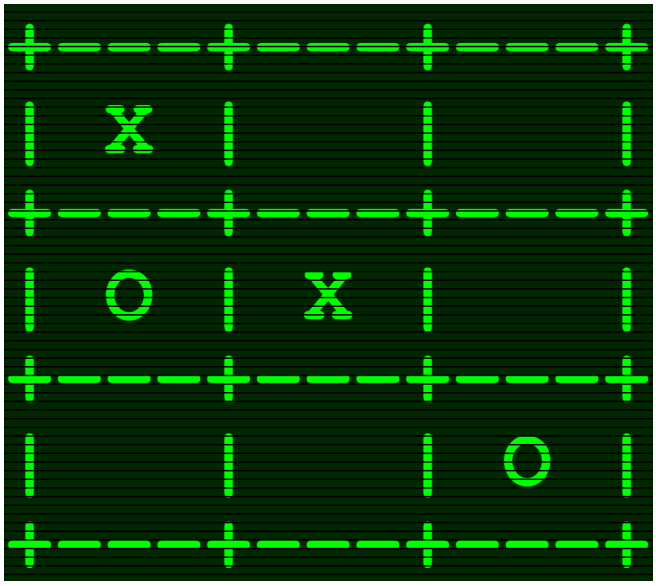

Figure 4.3: Concept art for the Early Computer environment.

Color Scheme green or amber on black

Music Theme 8-bit electronic

TODO

Variants

- Green marks and grid
- Amber marks and grid

4.1.5 Sidewalk

Color Scheme pastels including blue, yellow, and orange on gray

Music Theme Hip Hob / R&B

TODO

Variants

- Grid and mark colors randomly picked from pastel palette

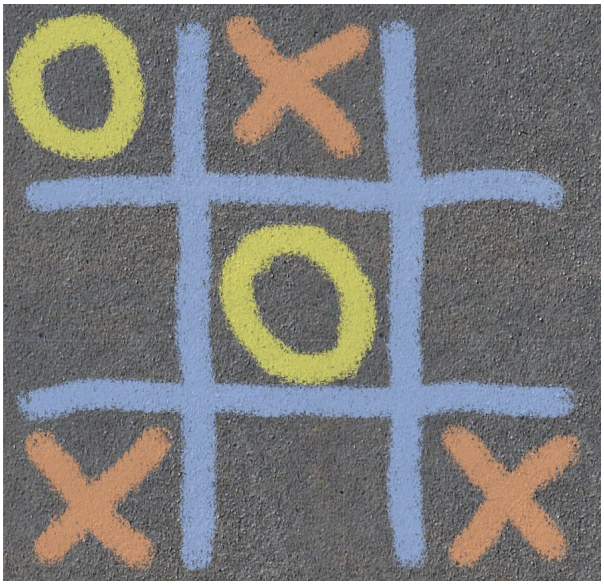

Figure 4.4: Concept art for the Sidewalk environment.

4.1.6 Papyrus Paper and Fancy Calligraphy

Color Scheme Black on faded tan

Music Theme Something classical

TODO

CHAPTER

FIVE

ADDITIONAL CONSIDERATIONS

The overview chapter provides a high level summary of the game.

5.1 Target Audience

Tic Tac Toe is targeted at casual gamers.

New PC Gamers

5.2 Target Platforms

Tic Tac Toe runs on a variety of popular desktop operating systems including Windows, macOS, and Linux. Being a casual game, it does not require a beefy video card or fast processor. Table 5.1 lists the minimum system requirements.

Table 5.1: Tic Tac Toe minimum system requirements

OS	Windows 10, Linux[9], macOS
Processor	1GHz dual core
RAM	4GB
Disk	500 MB available space
Graphics	OpenGL compatible graphics adapter

5.3 Monetization

5.4 License Compliance

5.5 Similar Games

Tic Tac Toe is a popular paper and pencil game that has been played since the times of ancient Egypt. Sandy Douglas' 1952 take on Tic Tac Toe, called OXO, was one of the first computer games.[11]

However, the major inspiration for this variant of Tic Tac Toe comes from James Richey's 2004 Tic Tac Toe[10] shown

[9] Tic Tac Toe is tested on the latest releases of Debian and Fedora. It should work on other popular distributions including Ubuntu. However, testing on other distributions is outside the scope of this project.

[11] See Wikipedia's Tic-tac-toe[12] article for additional information about the history of Tic Tac Toe.

[12] https://en.wikipedia.org/wiki/Tic-tac-toe

[10] https://www.imaginaryphase.com/ttt.html

in Figure 5.1.

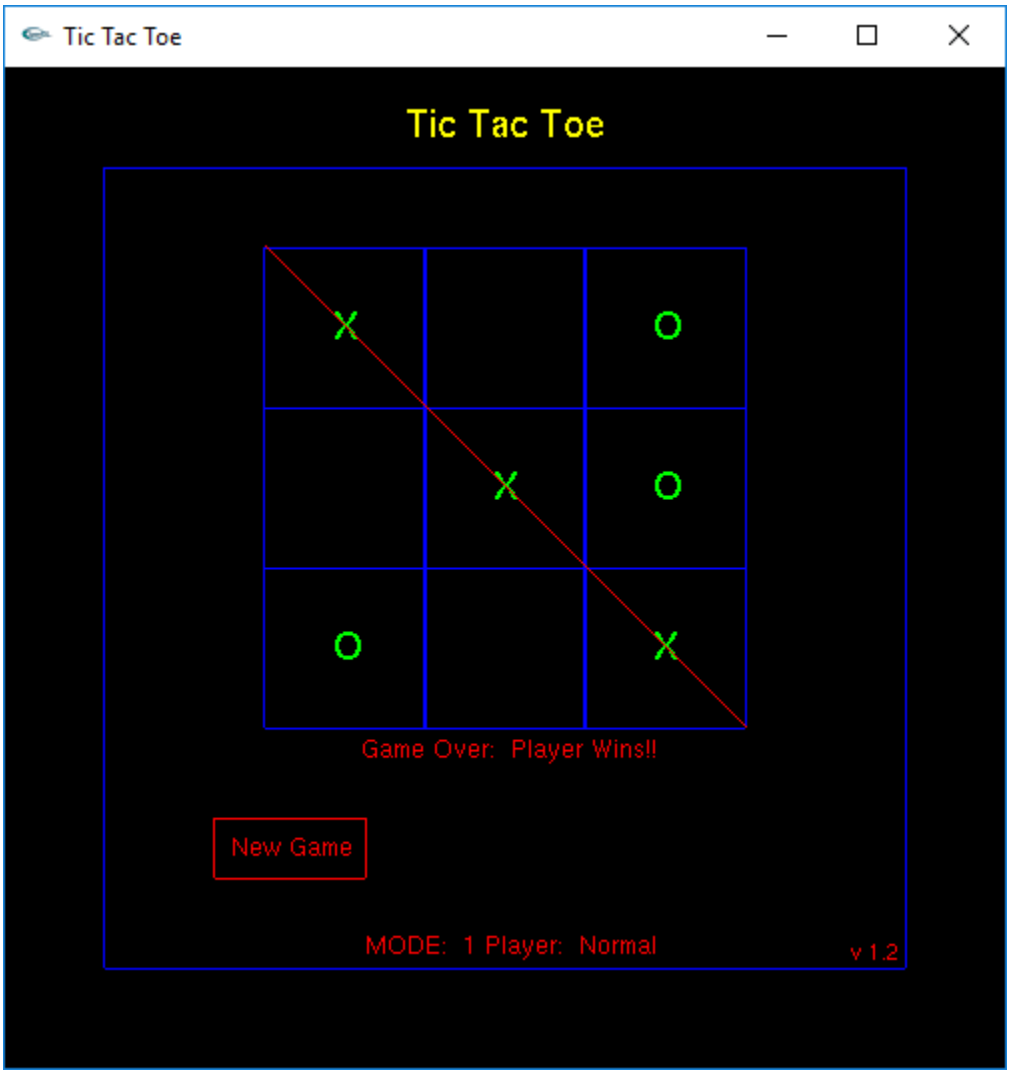

Figure 5.1: Screen shot of James Richey's 2004 Tic Tac Toe.

James developed his version of while learning the C++ programming language. Tic Tac Toe provides a fun challenge for learning a new language: it requires both algorithmic and graphical components to implement a small but nontrivial application.

When thinking of a project to use to help learn the Rust programming language, creating a Tic Tac Toe game is a fun choice.

5.6 Future Enhancements

CHAPTER

SIX

TECHNICAL DESIGN

CHAPTER

SEVEN

GLOSSARY

cat's game Term used when a game of Tic Tac Toe ends in a draw where there is no winner.

INDEX

C

cat's game, **25**
cat's game, 3