

# Game Design Document

Wordical

by Chris Livdahl

Created at the University of Washington, Bothell  
Advisor: Prof. William Erdly, Ph.D.

## Table of Contents

Overview .....	3
Game Play: Controls and User Interface .....	3
Rules and Scoring .....	6
Game System and Engine .....	6
Aesthetic Design .....	7
Title and Instruction Screens .....	7
Music and Sound Effects .....	8
Web Server .....	8
First Use .....	9
Sharing Scores.....	10
References .....	10

## Overview

Wordical is a word puzzle game where a player tries to find words in a grid of letters. Each word combination is worth points based on point values of individual letters and the length of words created.

Play takes place in rounds that have a time limit. The game also possesses a social element in that high scores are shared from all users playing the game after each round.

## Game Play: Controls and User Interface

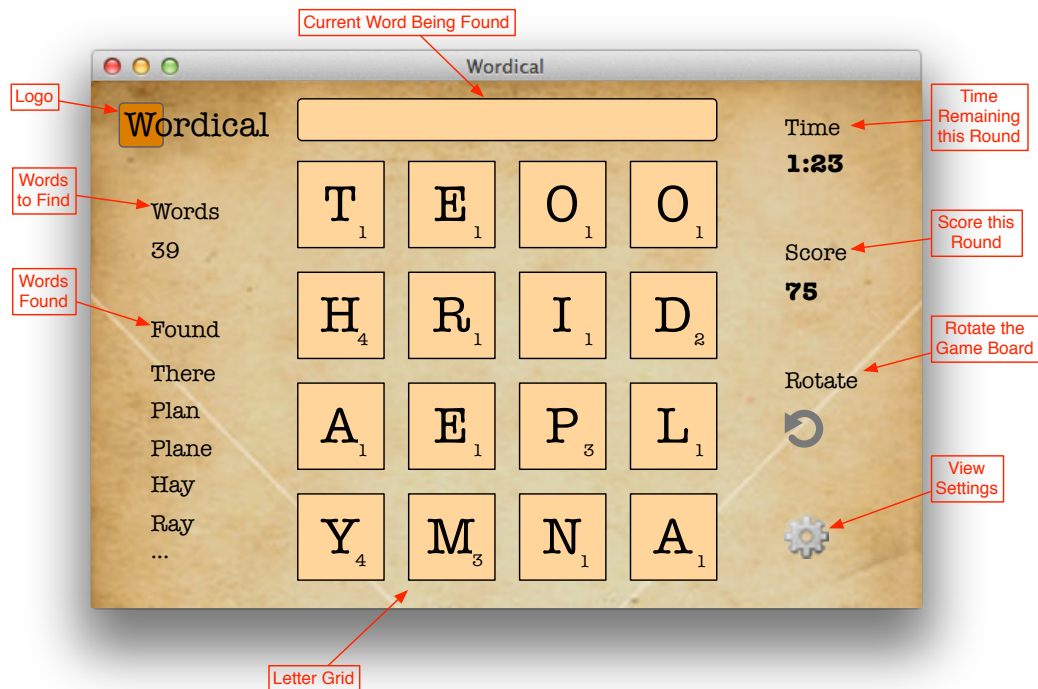


Figure 1 – Game Screen

The screen shown in Fig. 1 represents the main game screen. A countdown timer for the current round starts at two minutes, during which time the user can form words. The time remaining is displayed prominently in the upper corner. When the time expires, the round is over.

A letter grid in the center is displayed from a set of possible letter grids. The letter grid is where the user will connect letters together to form words. When the user is forming a word, the letters selected appear in the box at the top. The current score on this round is displayed prominently as well. A rotate button allows rotation of

the grid of letters 90-degrees to the right in order to help the user see the board differently if they are stuck.

On the other side of the interface, the number of possible words for the current letter grid is displayed. Below that are the most recent words that the user has found.

The settings button shows a preferences screen (Fig 2.) where the user can set their username, change music volume, and change sound effects volume. The user can enter and save their username by typing into the text box and pressing the save button. Pressing the “Instructions” button brings up an instructions screen that shows how to play the game. And the “Back to Game” returns the user to the game screen.

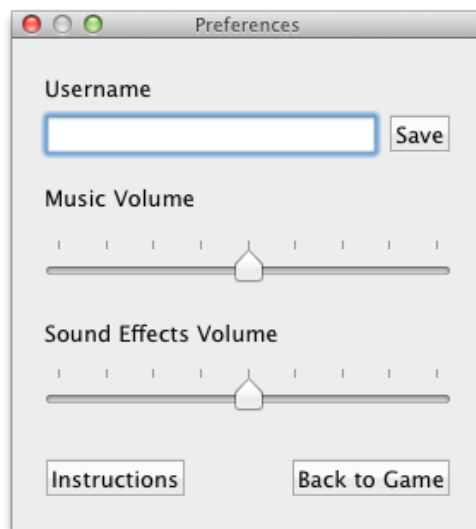


Figure 2 - Preferences Screen

Fig. 3 shows how the user interacts with the interface to find a word. When the user hovers the mouse over a letter, the color of the letter changes to a slightly darker color.

To select the first letter of a word, the user clicks on a letter. When a letter is selected it will turn a different color (shown in green in the screen prototype in Fig. 3). The user then moves the mouse over subsequent letters to start forming a word. When the user clicks again, the word is complete (this sequence of events is shown as a red dotted line in Fig. 3).

If the user has found a valid word, the word shows in green in the box at the top of the interface. The score blinks and is updated with the new total of the sum so far plus the word that was just formed. A point value of the word is flashed above the final letter (in this case, 8 points). After a brief moment, the letter colors are reset

and the word disappears from the “word found” box. If the word was not valid, then the letters are shown in red and then the letter colors are reset.

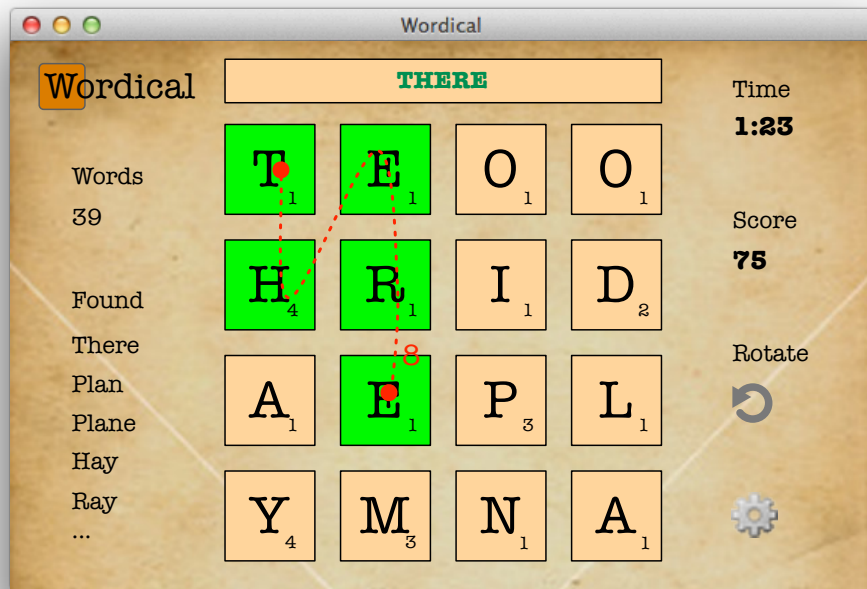


Figure 3 - Selecting a Word

When the time expires on the current round, the user is shown a summary “round over” screen, as seen in the screen prototype in Fig. 4. This screen denotes the total score as well as all the words found. Also listed are the possible words on the board. Scores from other users are shown in the “top scores” box. There is a countdown timer for when the next game will start.



Figure 4 - "Round Over" Screen

## Rules and Scoring

A user can select adjacent letters on the way to forming words. An adjacent letter is defined as any letter that is directly horizontal, vertical, or diagonal. Note that wrap-around to the other side of the board is not allowed. Also, a letter can't be selected twice in forming a word.

The point values for each letter initially will be the same as those in the game Scrabble, but could be adjusted based on play testing. The total value for a word is the sum of each letter value in the word, plus a bonus for the length of each word, per the bonus table below:

Word Length (Letters)	Bonus (Points)
5	10
6	15
7	25
8	40
9+	60

Figure 5 - Word Length Bonus Points

There is a two-minute timer for each round. The user has these two minutes to form as many words as possible on the game board. When the round is over a user will be presented with their list of words formed, a list of possible words and summary of their score, as well as a comparison to scores of other players who had the same game board.

## Game System and Engine

The game system for this game will start out as the Apple OS X desktop platform. There are many advantages with the Apple platform, such as quick prototyping in the "Interface Builder" inside of the Xcode IDE. Another advantage is that distribution is made much simpler through the Apple's App Store.

The game will be implemented in a way where other platform targets are possible, such as the iPhone and iPad. One way to make the game cross-platform is to have the game letter grids represented in a data format such as XML.

Many game engines were considered for this project, such as Unity. However, it was found that most of these engines would likely be overkill for such a game. The strongest route will be to use the native Apple frameworks. An alternative would be to use engines like Torque, cocos2d, as well as Moai for mobile development.

## Aesthetic Design

In *The Art of Game Design* by Jesse Schell, one of the “Lenses” he talks about looking at a game through is the Lens of Unification. This lens talks about finding a unifying theme of your game, then reinforcing that theme throughout the game. From this standpoint, the unifying theme of *Wordical* is how people enjoy words and language in printed form.

Since printed words appear on paper and especially in books, many aspects of the user interface will reflect paper and books. One way to accomplish this will be the texture of the background images in the graphics, which will appear like paper texture. Another way to reinforce the theme will be through the font. Thus, the letters on the game board will have a “type writer” appearance in font selection. Other fonts in the game should appear typed as well, using a serif font.

## Title and Instruction Screens

The title screen in Fig. 6 shows the game logo in large form, the author and short credits for the game. A play button takes the user to the game screen. The instructions button takes the user to an instructions screen that shows how to play the game.

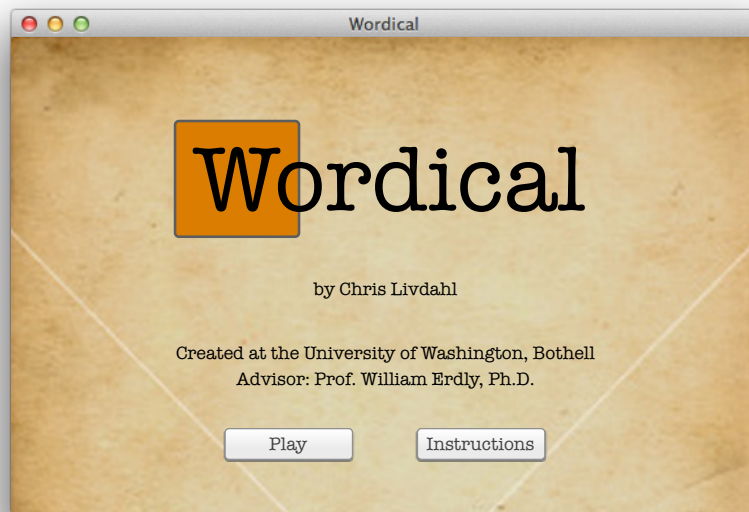


Figure 6 - Title Screen

As shown below in Fig. 7, the instructions are displayed clearly for the user. Upon pressing the close button, the user is taken back to whatever screen they were on.

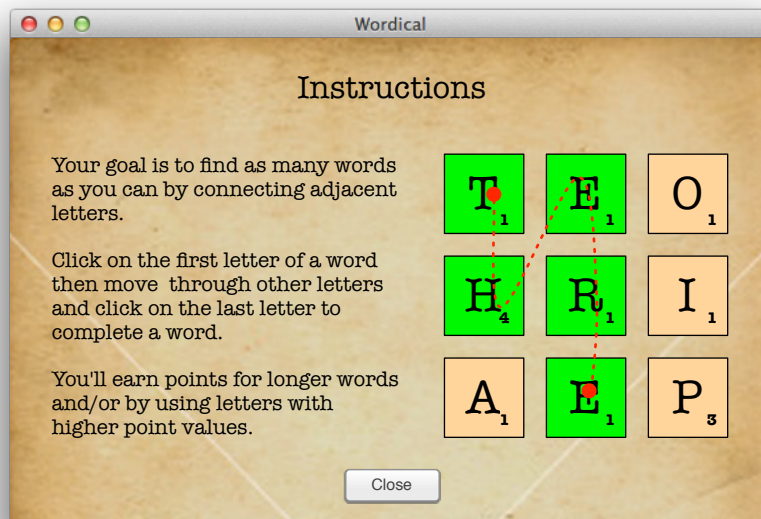


Figure 7 - Instruction Screen

## Music and Sound Effects

A basic musical background track will accompany the game play. With the theme of the application, the music will evoke a thoughtful mood. As such, ambient genre music is an appropriate choice.

When the user selects a letter, a brief tone will pulse, followed by a slightly higher pitched sound for each subsequent letter selected in forming a word. A successful completion of a word will be reinforced by a chiming sound, while an unsuccessful word will be accompanied by some kind of negative “bonk” sound.

The other, non-game play buttons in the game will have a brief click sound upon being pressed.

## Web Server

The web server provides a basic API for passing data to and from the game, and storing information in a database. For now, a simple Ruby on Rails install will be used. A web server enables the sharing of high scores. In the future, other information could be passed such as additional game content. The use of the API is discussed further in the following sections.



## First Use

When the user presses the “Play” button on the title screen for the first time, they will be prompted to enter their username (Fig. 8). Pressing the “Save” button takes them to a new game in the game screen (after syncing with the server if the user is online). Pressing the “Cancel” button takes the user back to the title screen. Note that a user can always edit their username in the preferences window. In a future version of the game, multiple users could be accommodated with a username table for the user to pick which username they want to use, or to enter a new username.

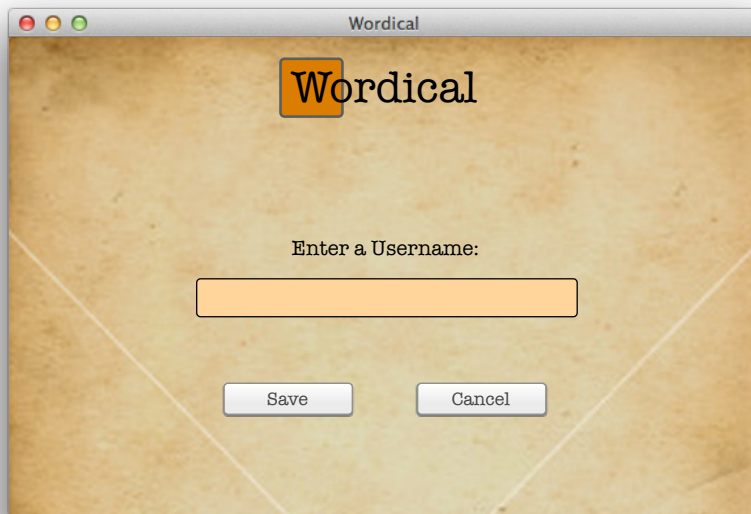


Figure 8 - Entering a Name

As shown in Fig. 9, in the first use of the game, the game registers with the server and the server issues a unique id (UID), which is stored locally in the user’s defaults. When the user enters their username, it is sent to the server to be associated to the UID in the database on the server.

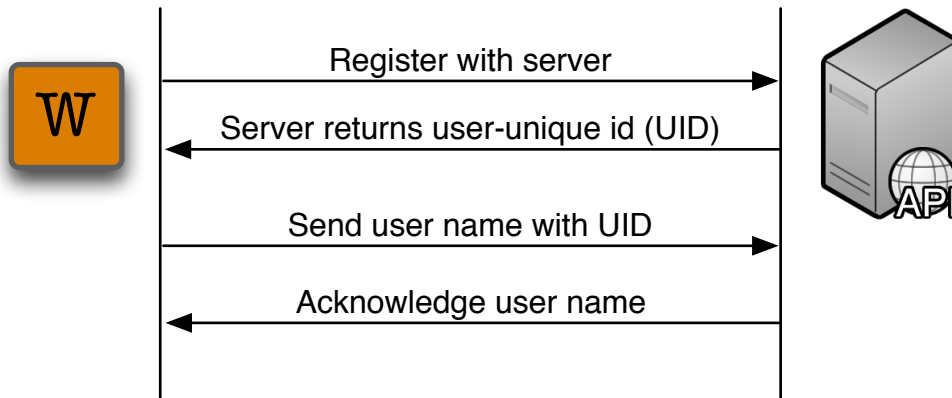


Figure 9 - Registering with the Server

## Sharing Scores

Each game board is given a game board id. When a user completes a game round, their game score is uploaded to the web server along with their UID and a game board id. Each two-minute round is given an id on the server. The score is stored and top scores for the particular game board and round are sent back.

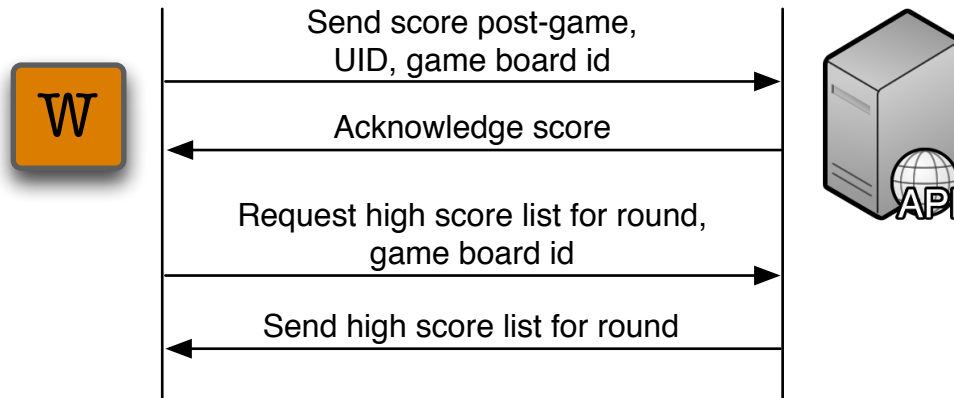


Figure 10 - Communication of Scores

## References

Schell, Jesse (2008). The Art of Game Design: A Book of Lenses Morgan Kaufmann.