# Blue System Requirements

CSC 480-800 / HCI 525-800 - Spring 2018

SUNY Oswego

Version 1.2.1

# Contents

Cierro, M., Anilonis, M., Sumano, C., Santos, J., Spagnola, J., Downey, A., & LeRoy, N. (2018). System Requirements CSC 480-800 / HCI 521-800 Spring 2018 (Vol. 2.1, Oswebble). Oswego, NY: SUNY Oswego.

## Section 1 - Introduction:

### Pretext

The product at question is the resultant of the State University of New York [SUNY] Oswego's HCI 521-800 / CSC 480-800 Software Design Spring 2018 course; directed and managed by Dr. Bastian Tenbergen. The objective of this course is to apply the concepts of software engineering and software development in a structured group project. The entire class will work together on a project assigned by the instructor(s) and will collaborate throughout the semester.

Our purpose is to sufficiently outline and shape a system that can effectively deliver a unique variation of a word building game to a workstation in the Richard S. Shineman Center of SUNY Oswego. The system will only interact with those close enough to connect a mobile device, laptop, or any device with web capabilities to the hardware.

This document exists as a model of the solution to the Board game kiosk requested by our stakeholder as interpreted by Blue Team. It serves as a supplement and companion to the Solution Neutral System Requirements Document and as such, references to it are made below.

### 1.1 System Definition Glossary:

[Notice]: See primary requirements document. Place further definitions below.

| | |
|---|---|
| DIRTY_WORD_TABLE | - Database table keeping track of invalid words and statistics for each. |
| GAME_TABLE | - Database table containing statistics about the current game session. |
| PLAYER_TABLE | - Database table keeping track of a user's statistics. |
| PLAYER_TEAM | - Database table specifying the team a player belongs to. |
| Oswego Themed | - refers to the use of the color scheme as follows: Oz Green #2a6342, Dark Oz Green #1a3f28, Oz Gold #ffcc33, Dark Oz Gold #c19a24, Grey Black #4d4d4d, White #f3f3f3. |
| TEAM_TABLE | - Database table keeping track of team statistics, spanning over all players who |

belong to that team.

Tile rack/rack - The area that contains an individual player's current tiles. This area will be represented by an image of a tile rack from a traditional Scrabble game.

USER_TABLE - Database table containing all devices to connect to the system by MAC address.

VALID_WORD_TABLE - Database table containing all valid words played in a game session.

Word Validation -The process for which a word is confirmed whether or not it is a legal playable word. It will also be determined whether or not the word is vulgar, and if the word has an oswego bonus.

### 1.2 Misc Glossary:

---

[Notice]: See primary requirements document. Place further definitions below.

QR Code - a machine-readable code consisting of an array of black and white squares, typically used for storing URLs or other information for reading by the camera on a smartphone.

See primary requirements document :

Cierro, M., Anilonis, M., Sumano, C., Santos, J., Spagnola, J., Downey, A., & LeRoy, N. (2018). System Requirements CSC 480-800 / HCI 521-800 Spring 2018 (Vol. 2.1, Oswebble). Oswego, NY: SUNY Oswego.

---

## Section 2 - Overall Description

### 2.1 Product Perspective:

---

Based from the instructor selected topic, 'arcade', the class agreed upon developing a SUNY Oswego Themed word building board game. Players will place tiles, in which the tiles must form words in: crossword fashion, read left to right in rows or downwards in columns, and be defined in a standard dictionary or lexicon.

This product will be built in the perspective of Team Blue's vision.

## 2.2 Product Functions:

The game itself will consist of a game session with four players--be them all AI players, all human players, or a combination of the two. Each player will take their turn in placing randomly selected tiles from their hand--any amount of tiles so long as it does not exceed the amount of tiles the player is currently holding--onto the game board; in an attempt to stem valid words off of one another until the game session runs into a end-game scenario. The end-game scenario can be one of two: (1) the game board is filled to a point where no possible moves are available, or (2) the players in the present game session 'pass' a given amount of times, such that they activate end-game scenario (1).

The players will be able to shuffle the tiles within their hand, place tiles onto the game board, exchange a given amount of tiles for new ones, access a 'help' button that shall explain the rules of the game and how to operate the game.

This product provides a features that allows players to pass their turn without exchanging tiles in order to signify that the players do not see any available words. This additionally leads to another end-game condition in which all human players pass their turn, signifying the game is over.

## 2.3 User Characteristics:

### C1.0 SUNY Oswego Student(s)

- Visits Shineman Center frequently: has some course in Shineman
- Knows that the game exists and how to access it

#### C1.1 Education Level

- C1.1a : English as primary language : has a fluent level of English vocabulary
- C1.1b : English as secondary language : has a basic level of English vocabulary

#### C1.2 Game Experience

- C1.2a : Has played some variation of Scrabble™ prior
- C1.2b : Has never played some variation of Scrabble™ prior

### C2.0 SUNY Oswego Faculty

- Knows that the game exists and how to access it

C2.1 Visits Shineman Center frequently
- C2.1a : Has some course in Shineman
- C2.1b : Has an office in Shineman

C2.2 Education Level
- C2.2a : English as primary language : has a fluent level of English vocabulary
- C2.2b : English as secondary language : has a basic level of English vocabulary

C2.3 Game Experience
- C2.3a : Has played some variation of Scrabble™ prior
- C2.3b : Has never played some variation of Scrabble™ prior

## 2.4 Constraints:

There is a kiosk stationed at the main entrance of Richard S. Shineman Science Center of SUNY Oswego, where two monitors will be garrisoned. The primary monitor will be projecting a live imitation of the game board, as well as the player's hand, all placed words on the game board, and the current game scores between the two teams. The secondary monitor will project game statistics--i.e. Each team's cumulative scores, team's best game scores, highest yielding word score, most frequently used word, etc.

The players are able to interact with the game via any device that has the ability to access the internet--such as smart devices, laptops, tablets, etc.

## 2.5 Assumptions and Dependencies:

(1) User Assumptions:
   (a) The User has a device that can connect to the target hardware.
   (b) The User is able to use their device correctly to play the game.
   (c) The User is literate in the English language.
   (d) The User understands how to connect and disconnect the game.
(2) Software Assumptions and Dependencies:
   (a) The Ubuntu operating system, on the target hardware, will be stable.
   (b) The server will be constantly running.
(3) Hardware Assumptions and Dependencies:

(a) The monitor hardware will be free of error.
(b) The CPU will successfully to execute instructions, on the target hardware.

(4) Other Assumption and Dependencies:
(a) All libraries used during development are reliable and consistent.
(b) The interface will be understandable to the players.

### 2.6 Design and Implementation Constraints:

(1) The Chosen Dictionary database
(a) English language
(b) Must not contain any proper nouns
(i) The Oswego Themed dictionary database is the only exception to this constraint
(c) Must not contain any word that is only a suffix, abbreviation, or prefix
(d) Must not include, nor contain, any word that requires a hyphen or apostrophe
(e) Shall not contain words of profanity

(2) Word Validation
(a) Words placed on the front end will not be considered valid until backend validation has occurred. This is to prevent, not only cheating, but rather maintain a consistent board state between the backend and the visual representation on the front end.

(3) Connection
(a) When connecting to the game server users will be prompted for an alias in which they can identify themselves with.
(b) If there is a spot open, a connected user will replace an AI and wait until it is their respective turn before being allowed to play a word.
(c) If there is no spot open the user will be prompted a message indicating the game is full. ~[TBD]
(d) When leaving the game, user stats will be recorded and sent to the database and then an AI will replace the respective player.

## Section 3 - System Requirements:

This section contains the formal natural language requirements in order to complete the goals listed in the KAOS diagram in section 6.1. Use these requirements in conjunction with section 6.2 for a comprehensive Solution Neutral Design Space.

## Section 3.1 - User interfaces

These requirements pertain to the environment facet of requirements design. They cover the hardware the system will be implemented on, in addition to how information is shown to a user.

| ID | Type | Requirement |
|---|---|---|
| BUI 1.0 | Constraint | All GUI items must be colored according to the oswego color theme. |
| BUI 2.0 | Functional | After a blank tile is played, it shall have the letter it represents on the tile as if it were a normal tile. |
| BUI 2.1 | Functional | After a blank tile is played, it shall not have a listed point value. |
| BUI 3.0 | Constraint | The gameboard shall be instantiated with predetermined multipliers |
| BUI 3.1 | Functional | The gameboard multipliers shall be the same for every game session |
| BUI 4.0 | Functional | Whenever the gameboard state changes, the gameboard on all game interfaces shall update to match these changes. |

### 3.1.1 Game Interface

The following requirements describes the functionality of the interface a human player will use to interact with the game.

| ID | Type | Requirement |
|---|---|---|
| BGI 1.0 | Constraint | The entire user interface must take place on a single HTML page. |
| BGI 2.0 | Functional | When the user first navigates to the HTML page, the system shall display either the login panel or welcome panel. |
| BGI 2.1 | Functional | The login panel shall allow the user to input their username. |
| BGI 2.1.1 | Functional | The login panel shall have a textbox to enter their username. |
| BGI 2.2 | Functional | The login panel shall have a button. |
| BGI 2.2.1 | Functional | The button shall be labeled “Log in”. |

| | | |
|---|---|---|
| BGI 2.2.2 | Functional | The button shall allow the user to begin the login process as described in S2.2 when pressed. |
| BGI 2.3.1 | Functional | The welcome panel shall have a prompt welcoming the user back to the game. |
| BGI 2.3.2 | Functional | The welcome panel shall have a button. |
| BGI 2.3.2.1 | Functional | The button shall be horizontally centered and below the prompt. |
| BGI 2.3.2.2 | Functional | When pressed the button shall redirect the user to the team panel. |
| BGI 2.4 | Functional | If the game session currently has four human players, the system shall not allow the user to proceed to the login panel or welcome panel. |
| BGI 2.4.1 | Functional | When this occurs, the system shall follow the scenario as described in A10.0 in the Solution Neutral Document. |
| BGI 3.0 | Functional | After a successful login the user shall be redirected to the team panel. |
| BGI 3.1 | Functional | The team panel shall have a switch. |
| BGI 3.1.1 | Functional | The default state of the switch shall be determined by the player's assigned team. |
| BGI 3.1.2 | Functional | The switch shall be labeled "Team". |
| BGI 3.1.3 | Functional | The switch shall be the color of the team that is currently selected. |
| BGI 3.1.4 | Functional | When the switch is to the right, the Gold Team shall be selected. |
| BGI 3.1.5 | Functional | When the switch is to the left, the Green Team shall be selected. |
| BGI 3.2 | Functional | The team panel shall have a button. |
| BGI 3.2.1 | Functional | The button shall be above the switch. |
| BGI 3.2.2 | Functional | The button shall be labeled "Join Game". |
| BGI 3.2.3 | Functional | When the button is pressed, the system shall move the player to the game panel. |
| BGI 4.0 | Functional | The game panel shall allow a human player to participate in the current game session. |
| BGI 4.1 | Functional | The game panel shall display the gameboard in the center. |

| | | |
|---|---|---|
| BGI 4.2 | Functional | The game panel shall display the score of the player on the horizontal center. |
| BGI 4.3 | Functional | The game panel shall display the time remaining in a turn to the left of the horizontal center. |
| BGI 4.4 | Functional | The game panel shall display the player's tiles below the gameboard. |
| BGI 4.5 | Functional | The game panel shall allow the users to place tiles on the gameboard. |
| BGI 4.5.1 | Constraint | The user must only be allowed to place a tile on the gameboard where no other tile exists. |
| BGI 4.5.1.1 | Functional | If the user attempts to place a tile on a space containing a tile, the tile being placed shall return to its place on the rack. |
| BGI 4.5.2 | Functional | The user shall be able to place tiles by pressing a tile on the rack and then pressing an empty space on the gameboard. |
| BGI 4.5.3 | Functional | The user shall be able to return tiles to the rack played that turn by pressing them and then selecting an empty space on the tile rack after they have been placed on the gameboard. |
| BGI 4.5.4 | Functional | If the user plays a blank tile, they shall be able to select the letter they wish to play. |
| BGI 4.6.0 | Functional | The game page shall have the following buttons. |
| BGI 4.6.0.1 | Functional | All buttons shall have images representative of their respective functions. |
| BGI 4.6.1 | Functional | The first button shall be located in the top left corner. |
| BGI 4.6.1.1 | Functional | When the button is pressed, the system shall prompt the user, confirming if the user wants to quit the game session. |
| BGI 4.6.2 | Functional | The second button shall be located at the bottom, second to the left. |
| BGI 4.6.2.1 | Functional | When the button is pressed, the system shall start the exchange process as described in BGI 3.7. |
| BGI 4.6.3 | Functional | The third button shall be located bottom second from the right. |
| BGI 4.6.3.1 | Functional | When the button is pressed, the system shall change the order of the tiles on the tile rack randomly. |

| | | |
|---|---|---|
| BGI 4.6.4 | Functional | The fourth button shall be located in the bottom right corner. |
| BGI 4.6.4.1 | Functional | If the human player has played a valid word and the button has been pressed, a prompt shall appear with the points the player earned. |
| BGI 4.6.4.2 | Functional | If the human player has played an invalid word and the button has been pressed, a prompt shall appear that notifies the player that their word is invalid. |
| BGI 4.6.4.3 | - | Removed as of v1.2.0. |
| BGI 4.6.4.4 | Functional | If the human played a vulgar word, the system shall warn the user on playing vulgar words. |
| BGI 4.6.4.5 | Functional | If the human player has violated any game rules on tile placement, the system shall notify the player of their mistake. |
| BGI 4.6.5 | Functional | The fifth button shall be located in the bottom left corner. |
| BGI 4.6.5.1 | Functional | When the button is pressed, all tiles the player has placed this turn shall be returned to the tile rack. |
| BGI 4.6.6 | Functional | The sixth button shall be located in the top right corner. |
| BGI 4.6.6.1 | Functional | When the button is pressed, the system shall display a help panel. |
| BGI 4.6.7 | Functional | The only buttons that can be pressed when it is not the player's turn are the buttons that allow the user to shuffle, show the help panel, and to leave the game. |
| BGI 4.6.7.1 | Functional | Buttons that cannot be pressed shall be grayed out. |
| BGI 4.7 | Functional | The game page shall have an exchange process. |
| BGI 4.7.1 | - | Removed as of v1.2. |
| BGI 4.7.2 | Functional | The exchange process shall add two new buttons on the bottom. |
| BGI 4.7.2.1 | Functional | The first button shall be labeled "Cancel". |
| BGI 4.7.2.1.1 | Functional | The button shall be located at the bottom left of the panel. |
| BGI 4.7.2.1.2 | - | Removed as of v1.2. |
| BGI 4.7.2.1.3 | Functional | When the button is pressed, the system will stop the exchange process and return them to the game panel. |

| | | |
|---|---|---|
| BGI 4.7.2.2 | Functional | The second button shall be labeled “Confirm”. |
| BGI 4.7.2.2.1 | Functional | The button shall be located in the bottom right of the panel. |
| BGI 4.7.2.2.2 | Functional | When pressed, the second button shall remove the selected tiles and replace them with ones randomly selected. |
| BGI 4.7.2.2.3 | Functional | When pressed, the second button shall end the players turn. |
| BGI 4.7.2 | Functional | There shall be a prompt in the center of the screen explaining how to exchange tiles. |
| BGI 4.7.3 | Functional | The human player shall be able to select multiple tiles on their tile rack. |
| BGI 4.7.3.1 | Functional | The tiles that are selected shall have a colored highlight. |
| BGI 4.8 | Functional | The exchange button will be highlighted during the entirety of the exchange process. |
| BGI 5.0 | - | Removed as of v1.2. |
| BGI 6.0 | - | Moved to BUI 4.0 |
| BGI 7.0 | - | Moved to BUI 3.0. |
| BGI 8.0 | - | Removed as of v1.2. |
| BGI 9.0 | Functional | The help screen shall display relevant information to assist the user in using the system. |
| BGI 10.0 | Functional | The top and bottom of the help screen shall have a close button. |
| BGI 10.1 | Functional | The close button shall have the word “close” displayed on it in the color white. |
| BGI 10.1.1 | Functional | The close button shall have a green background. |
| BGI 11.0 | Functional | The screen shall be titles “Help Screen” in bold and black letters. |
| BGI 11.1 | Functional | There shall be a section on the help screen titled “Rules”. |
| BGI 11.1.1 | Functional | The Rules section shall contain a list of seven rules that assist the user in understanding the basic rules of the game. |
| BGI 12.0 | Functional | Underneath the Rules Section shall be a section titled “Buttons”. |
| BGI 12.1 | Functional | The Buttons Section shall have a list describing the five interactable |

| | | |
|---|---|---|
| | | buttons during the game session. |
| BGI 12.2 | Functional | To the left of each button description shall be a picture of the associated button along with the name of the button. |

### 3.1.2 Primary Monitor

The following requirements describes the functionality of the primary monitor, which is responsible for displaying all information about a single game session.

| ID | Type | Requirement |
|---|---|---|
| BPM 1.0 | Functional | There shall be a section of the screen dedicated to the game session. |
| BPM 1.1 | Functional | That section shall be square. |
| BPM 1.2.1 | Functional | The square shall have a height equal to the height of the monitor. |
| BPM 1.3.0 | Functional | The gameboard shall be in the center of the square. |
| BPM 1.4.0 | Functional | At each of the edges of the square there shall be a panel as described in BPM2.0. |
| BPM 1.5.0 | Functional | Each panel shall be oriented such that each panel shall face towards the gameboard. |
| BPM 1.5.1 | Quality | The panel text shall be oriented such that the user can read it easily. |
| BPM 1.6.0 | - | Removed as of v1.2.0. |
| BPM 1.7 | Functional | Whenever new tiles are requested those tiles shall appear in an empty space on the player's tile rack. |
| BPM 1.8.0 | Functional | When a Team wins, the system shall prompt a victory message. |
| BPM 1.8.1 | Functional | The message shall be centered over the gameboard. |
| BPM 1.9.0 | Functional | When a player plays a word, the system shall display the point value. |
| BPM 1.9.1 | Functional | If the word played is a bonus word, the system shall have an additional prompt indicating that a bonus occured with the bonus amount. |
| BPM 2.1 | Functional | The panel shall contain information relevant to the player. |
| BPM 2.1a | Functional | The information shall be team, and current tiles. |

| | | |
|---|---|---|
| BPM 2.2 | Functional | A player's panel shall be highlighted when it is that player's turn. |
| BPM 3.0 | Functional | The remaining section not used as specified in BPM1.0, shall be used to display current game session statistics. |
| BPM 3.0.1 | Functional | The game session statistics shall be each teams scores, player scores, and current turn for the current game session. |
| BPM 3.0.2 | Functional | The section shall be to the right of the remaining section not used by the requirements of BPM1.0. |
| BPM 3.1 | Functional | The section shall have a panel for displaying current game session information. |
| BPM 3.1.1 | Functional | The panel shall have a 2x3 table. |
| BPM 3.1.1.1 | Functional | The top row shall contain labels for each team. |
| BPM 3.1.1.2 | Functional | The bottom two rows will display a player and that player's score for each column. |
| BPM 3.2 | Functional | In the bottom of the section there shall be a game log panel. |
| BPM 3.2.1 | Functional | The game log panel will display the actions of the last 5 panels. |

### 3.1.3 Secondary Monitor

The following requirements describes the functionality of the Secondary Monitor, which is responsible for displaying statistical information about all game sessions and the Green and Gold teams.

| ID | Type | Requirement |
|---|---|---|
| BSM 1.0 | Functional | The monitor shall have a leaderboard section. |
| BSM 1.1 | Functional | The leaderboard section shall have a horizontally centered label. |
| BSM 1.1.1 | Functional | The label shall be "Leaderboard". |
| BSM 1.2 | Functional | There shall be two tables. |
| BSM 1.2.1 | Functional | The tables shall be just below the label as described in BSM 1.1.1. |
| BSM 1.2.1.1 | Functional | The table for the Green Team shall be on the left side. |

| | | |
|---|---|---|
| BSM 1.2.1.1.1 | - | Moved to Appendix as of v. 1.2. |
| BSM 1.2.1.2 | Functional | The table for the Gold Team shall be on the right side. |
| BSM 1.2.1.2.1 | - | Moved to Appendix as of v. 1.2. |
| BSM 1.2.2 | Functional | Each of the team tables shall have 5 columns. |
| BSM 1.2.2.1 | Functional | The columns shall be labeled: “Rank”, “Player”, ”Score”, ”Games Played” and “Average Score”. |
| BSM 1.2.2.1.1 | Functional | The columns shall be labeled in the order listed in BSM 1.2.2.1. |
| BSM 1.2.3 | Functional | Each table shall have a row for each relevant statistic with an additional row for labels. |
| BSM 1.2.3.1 | Functional | The relevant statistics are “Total score”, “Avg Score per Game”, “Highest Word Score”, and “T-Test”. |
| BSM 1.2.3.2 | Functional | The players shall be displayed in descending order, top to bottom. |
| BSM 2.0 | Functional | The monitor shall have a statistics section. |
| BSM 2.1 | Functional | The statistics section shall have a centered label. |
| BSM 2.1.1 | Functional | The label shall be labeled “Statistics”. |
| BSM 2.2 | Functional | The statistics section shall have a table. |
| BSM 2.2.1 | Functional | The table shall be located just below the label as described in BSM 2.1. |
| BSM 2.2.2 | Functional | The table shall have a row for each team, with an additional row for labels. |
| BSM 2.2.2.1 | Functional | The first row shall be used for labels. |
| BSM 2.2.2.1.1 | Functional | The column shall contain the labels described in BSM 2.2.3.*. |
| BSM 2.2.2.2 | Functional | The second column shall be labeled Green. |
| BSM 2.2.2.3 | Functional | The third column shall be labeled Gold. |
| BSM 2.2.3 | Functional | The table shall have columns ordered left to right; as specified below. |
| BSM 2.2.3.1 | Functional | The first row shall be labeled “Team” |
| BSM 2.2.3.2 | Functional | The second row shall be labeled “Cumulative”. |

| | | |
|---|---|---|
| BSM 2.2.3.3 | Functional | The third row shall be labeled “Longest Word”. |
| BSM 2.2.3.4 | Functional | The fourth row shall be labeled “Wins”. |
| BSM 2.2.3.5 | Functional | The fifth row shall be labeled “Tiles”. |
| BSM 2.2.3.6 | Functional | The fifth row shall be labeled “Losses”. |
| BSM 2.2.3.7 | Functional | The fifth row shall be labeled “Highest Word Score”. |
| BSM 2.2.3.8 | Functional | The fifth row shall be labeled “B-Word Count”. |
| BSM 2.3 | Functional | The cells in the table shall contain overall statistics as defined by the intersection of each label described in BSM 2.2.2 and BSM 2.2.3. |
| BSM 3.0 | Functional | In the bottom left, there shall be a section for a T-Test. |
| BSM 3.1 | Functional | The section shall be labeled “T-Test” |
| BSM 3.2 | Functional | The section shall have a table. |
| BSM 3.2.1 | Functional | The table shall have columns; labeled left to right; as specified below. |
| BSM 3.2.1.1 | Functional | The first column shall be labeled “Team”. |
| BSM 3.2.1.2 | Functional | The second column shall be labeled “Average Score”. |
| BSM 3.2.2 | Functional | The table shall have a row for each team, with an additional row for labels. |
| BSM 4.0 | Functional | To the right of the section in BSM 3.0 shall contain a layman’s explanation of the T-test. |
| BSM 4.1 | Functional | The section shall be labeled “Conclusion:” |
| BSM 4.2 | Functional | The section shall indicate which team is performing better, and how severely. |
| BSM 5.0 | Functional | There shall be a QR Code in the lower right corner of the screen. |
| BSM 5.1 | Functional | The QR Code shall give a link when read for the user to connect to the system. |

## Section 3.2 - Functional Requirements:

These are requirements which pertain to the basic and necessary requirements for Team Blue's implementation of the "Oswebble" system.

### 3.2.1 Game Rules

See primary requirements document for overarching requirements. This section contains rules regarding the unique interpretation of blue team.

| ID | Type | Requirement |
|---|---|---|
| BGR 1.0 | Functional | If the game session currently only has AI players, the game is over when all four players exchange tiles three times in a row. |
| BGR 2.0 | Functional | If the game session has at least one human player, the game is over when all four players exchange tiles four times in a row. |

### 3.2.2 Design Constraints

See primary requirements document for overarching requirements.

| ID | Type | Requirement |
|---|---|---|
| BDC 1.0 | Constraint | The primary monitor display must use Java Swing. |
| BDC 2.0 | Constraint | The secondary monitor display must use Java Swing. |
| BDC 3.0 | Constraint | The User interface must use a combination of HTML, CSS, jquery and JavaScript. |
| BDC 3.1 | Constraint | The system must use HTML5. |
| BDC 3.2 | Constraint | The system must use CSS 3. |
| BDC 3.3 | Constraint | The system must use jquery 3.3.1. |
| BDC 3.4 | Constraint | The system must use javascript 1.7. |
| BDC 4.0 | Constraint | The system must use a Tomcat server. |

| | | |
|---|---|---|
| BDC 5.1 | Constraint | The Tomcat Version must be version 8 |
| BDC 5.0 | Constraint | The system must use the Ubuntu 16.04 english dictionary as a base list of valid words. |
| BDC 5.1 | Constraint | The Ubuntu dictionary must be trimmed to remove invalid words referenced in DC 3.0 in the Solution Neutral document. |
| BDC 6.0 | Constraint | The system must use MySQL for the database. |
| BDC 6.1 | Constraint | The system must use MySQL 5.7. |
| BDC 7.0 | Constraint | The system must use Java 8. |

### 3.2.3 Software System Attributes

The following section contains requirements pertaining to the functioning of the AI, along with information regarding database operations. Diagrams regarding the database can be found in Section 5.3. See primary requirements document for overarching requirements.

| ID | Type | Requirement |
|---|---|---|
| BSA 1.0 | Functional | Any playing position not held by a human player shall be held by an AI. |
| BSA 2.0 | Functional | Word validation shall occur before the state of the gameboard is changed. |
| BSA 2.1 | Functional | During word validation, a word will be determined if it is valid or invalid. |
| BSA 2.2 | Functional | If a word is valid, the word shall checked to determine if it is an Oswego bonus word. |
| BSA 2.3 | Functional | If a word is invalid, the word shall be checked to determine if it is an vulgar word. |
| BSA 3.0 | - | Moved to Appendix as of v. 1.2. |
| BSA 4.0 | Functional | A timer shall be used to replace a human player with an AI after the timer has reached a predetermined amount of seconds. |
| BSA 4.1 | Functional | The timer shall start once the player’s turn has commenced. |

| | | |
|---|---|---|
| BSA 5.0 | Functional | User information about a human player shall be pushed to the database once the player has left the game in any manner. |
| BSA 6.0 | Functional | The overall statistic scores, as defined in BSM 2.3, shall be pushed to the database when the game session ends. |
| BSA 7.0 | Functional | The database shall consist of seven separate data tables. |
| BSA 7.1 | Functional | The database shall contain a VALID_WORD_TABLE table. |
| BSA 7.1.1 | Functional | The VALID_WORD_TABLE table shall hold all played words in each game session. |
| BSA 7.1.1.1 | Functional | Each word in the VALID_WORD_TABLE table shall have an ID. |
| BSA 7.1.1.2 | Functional | Each word in the VALID_WORD_TABLE table shall have a value. |
| BSA 7.1.1.3 | Functional | Each word in the VALID_WORD_TABLE table shall have a length. |
| BSA 7.1.1.4 | Functional | The VALID_WORD_TABLE table shall specify whether a word is an extension of an already existing word. |
| BSA 7.1.1.5 | Functional | The VALID_WORD_TABLE table shall specify whether a word has a bonus applied to it. |
| BSA 7.1.1.5.1 | Functional | The VALID_WORD_TABLE table shall specify the bonus awarded, if the specification in BSA 7.1.1.5 is true. |
| BSA 7.2 | Functional | The database shall contain a PLAYER_TABLE table. |
| BSA 7.2.1 | Functional | The PLAYER_TABLE table shall contain a user ID that references a unique user ID in the USER_TABLE table. |
| BSA 7.2.2 | Functional | The PLAYER_TABLE table shall track the user's cumulative score. |
| BSA 7.2.3 | Functional | The PLAYER_TABLE table shall hold the longest word played by the player in all game sessions they have participated in. |
| BSA 7.2.4 | Functional | The PLAYER_TABLE table shall hold the single highest scoring word played by the player in all game sessions they have participated in. |
| BSA 7.3 | Functional | The database shall contain a GAME_TABLE table. |
| BSA 7.3.1 | Functional | The GAME_TABLE table shall contain a game session ID. |
| BSA 7.3.2 | Functional | The GAME_TABLE table shall contain both of the team's scores per |

| | | |
|---|---|---|
| | | game session. |
| BSA 7.3.3 | - | Moved to the Appendix as of v. 1.2. |
| BSA 7.4 | Functional | The database shall contain a TEAM_TABLE table. |
| BSA 7.4.1 | Functional | The TEAM_TABLE table shall contain a team name. |
| BSA 7.4.2 | Functional | The TEAM_TABLE table shall contain the cumulative score for the team over all game sessions. |
| BSA 7.4.3 | Functional | The TEAM_TABLE table shall store the single highest scoring word played by any player on each team over all game sessions. |
| BSA 7.4.4 | Functional | The TEAM_TABLE table shall store the single highest game session score achieved by each team over all game sessions. |
| BSA 7.4.5 | Functional | The TEAM_TABLE table shall contain a win counter. |
| BSA 7.4.6 | Functional | The TEAM_TABLE table shall contain a tie counter. |
| BSA 7.4.7 | Functional | The TEAM_TABLE table shall contain a loss counter. |
| BSA 7.4.8 | Functional | The TEAM_TABLE table shall store the longest word played by any player on each team over all game sessions. |
| BSA 7.4.9 | Functional | The TEAM_TABLE table shall contain a count of every bonus word played by each team. |
| BSA 7.4.10 | Functional | The TEAM_TABLE table shall contain all dirty word submission attempts by any player on each team over all game sessions. |
| BSA 7.5 | Functional | The database shall contain a USER_TABLE table. |
| BSA 7.5.1 | Functional | The USER_TABLE table shall contain an ID for every registered user. |
| BSA 7.5.2 | Functional | The USER_TABLE table shall contain the mac address of every registered user. |
| BSA 7.6 | Functional | The database shall contain a DIRTY_WORD_TABLE table. |
| BSA 7.6.1 | Functional | The DIRTY_WORD_TABLE table shall hold the words a player cannot submit. |
| BSA 7.6.2 | Functional | Each word in the DIRTY_WORD_TABLE table shall have its own ID. |
| BSA 7.6.3 | Functional | Each word in the DIRTY_WORD_TABLE table shall have a counter. |

| | | |
|---|---|---|
| BSA 7.6.3.1 | Functional | The counter for each dirty word shall keep track of the attempts to play said word. |
| BSA 7.7 | Functional | The database shall contain a PLAYER_TEAM table. |
| BSA 7.7.1 | Functional | The PLAYER_TEAM table shall contain a user ID for each user. |
| BSA 7.7.2 | Functional | The PLAYER_TEAM table shall contain a team ID. |
| BSA 8.0 | Functional | When the game session is over, all human players will be removed and replaced with AI. |

### 3.2.4 Other Requirements

See primary requirements document for overarching requirements.

| ID | Requirement Type | Requirement |
|---|---|---|
| BOR 1.0 | Functional | The AI shall use a subset of the dictionary mentioned in BDC 5.1. |
| BOR 1.1 | Quality | The subset dictionary shall contain words that allows the AI to be beatable. |
| BOR 1.2 | Functional | The AI shall calculate all possible moves available. |
| BOR 1.2.1 | Functional | The AI shall play a random word from all possible moves. |
| BOR 1.3 | Quality | The AI shall wait no less than 3 seconds to play a word. |
| BOR 1.3.1 | Functional | If the AI has calculated all possible moves before the minimum play time, it shall wait until the minimum play time has elapsed. |
| BOR 1.4 | Quality | The AI shall play a word before 60 seconds have passed. |
| BOR 1.4.1 | Functional | If the AI has not finished processing moves by 60 seconds, the AI shall return what moves it has calculated, followed by selecting a move at random. |
| BOR 1.5 | Funcional | If there are no possible moves, the AI shall exchange a random number of tiles. |

## Section 4 - System Scenarios:

This section contains stimuli that the system will most likely encounter while operating and the responses it should have to these said stimuli. Diagrams for each of these scenarios can be found in section 5.1.2.

### 4.1 Primary Scenarios

These scenarios pertain to ideal stimuli and responses from the system. See primary requirements document for overarching requirements.

| ID | Actor Action | System Response |
|---|---|---|
| BPS 1.0 | Every player passes their turn the required amount of times. | BAS 1.0 response is enacted as well as overall team score being pushed to the database. Then the system proceeds as per as per S7. |
| BPS 2.0 | - | Removed as of v1.2. |
| BPS 3.0 | - | Removed as of v1.2. |
| BPS 4.0 | A word is submitted for validation. | The word is examined by the database, and determined if it is valid or not. If it is valid, it determines if it is an oswego word. If it is not valid, it determines if it is vulgar or not. Regardless of the outcome, the user is informed of the result. |

### 4.2 Alternative Scenarios

These scenarios pertain to non-ideal stimuli and the system's responses to them. All responses will attempt to return to ideal operating conditions. See primary requirements document for overarching requirements.

| ID | Actor Action | System Response |
|---|---|---|
| BAS 1.0 | A human player leaves the game | The human player is then replaced with an AI player. |

| | | |
|---|---|---|
| | session. | Afterwords, the overall score, words played, and other statistics about the player are pushed to the database. |

---

## Section 5 - Diagrams:

The following section contains diagrams that add clarity to the requirements and scenarios described in section 5.3.

### Section 5.1 - Engine Diagrams

All Diagrams relating to the Engine Team's implementation.

---

#### 5.1.1 - Class Diagrams

UML Class Diagrams representing the Engine Team's Java Source Code.

#### 5.1.1.1 : Views Package

#### 5.1.1.2 : Session Package

#### 5.1.1.3 Models Package

#### 5.1.1.4 Components Package

### 5.1.2 - Sequence Diagrams

These diagrams show the flow of scenarios found in Section 4.

**BPS 1.0 : A Game Session Ends**

**BPS 4.0 : Word Validation**

**BAS 1.0 : A Human Leaves the Game Session**

## Section 5.2: GUI Diagrams

These following diagrams are visual representations of what the requirements in section 3.1 are describing. The subsections contain wireframes relating to how a user interacts with the user interface.

---

**5.2.1 : Login panel**

**5.2.2 : Game Full Panel**

**5.2.3 : Welcome Back Panel**

### 5.2.4 : User Boardgame Panel

**5.2.5 : Blank Tile Panel**

### 5.2.6 Help Menu Panel

**5.2.7 : Primary Monitor Display**

**5.2.8 : Secondary Monitor Display**

## Section 5.3 - Database Diagrams

All Diagrams relating to the Database.

### 5.3.1: Database Schema

## Appendix A - Depreciated Items

This section contains requirements and scenarios that were removed from section 3 and section 4. The original location of these requirements will be marked as such in section 3. Their IDs here are kept for reference purposes.

[As of version 1.1]

| ID | Pre-Text | Post-Text |
|---|---|---|
| BGI 4.7.1 | Functional | The exchange process shall remove the four buttons located along the bottom. |
| BGI 4.7.2.1.2 | Functional | When the button is pressed, the system shall replace the current buttons with the four removed in BGI 3.7.1. |

[As of version 1.2.0]

| ID | Pre-Text | Post-Text |
|---|---|---|
| BGI 4.6.4.3 | Functional | If the human player has played no tiles and the button has been pressed, the system shall warn the player they are about to pass their turn. |
| BGI 4.6.4.3.1 | Functional | The system shall allow the player to either pass their turn or resume their current turn. |
| BGI 5.0 | Functional | If the user attempts to join the game session and there are four human players already in a game session, then the user shall be redirected to a page that indicates that the current game session is currently full. |
| BGI 8.0 | Functional | An option shall be displayed for a player to quit. |
| BPM 1.6.0 | Functional | When a player plays a word, the relevant tiles on their tile rack shall move to their respective positions to form the word. |
| BSM 1.2.1.1.1 | Functional | This table shall show information regarding to Green Team human players. |
| BSM | Functional | This table shall show information regarding to Gold Team human |

| | | |
|---|---|---|
| 1.2.1.2.1 | | players. |
| BSA 3.0 | Constraint | The User class shall initiate the tile count to 7. |
| BSA 7.3.3 | Functional | The GAME_TABLE table shall hold the Green Team's score per game session. |

| ID | Pre-Text | Post-Text |
|---|---|---|
| BPS 3.0 | A user submits a word. | Word validity will be checked against the played word. |
| BPS 2.0 | Every player passes their turn the required amount of times. | The game session ends and BPS 1.0 is enacted. |