Functional requirements

Find players

Description:
Our system must allow users to have the option of finding other users to play against. Once the user finds his opponents our system needs to create a lobby of players ready for the match. Both will be managed by Firebase realtime database.

Criticality:
This a critical feature as the end goal of our app is a multiplayer game.

Technical issues:
We need to manage and store data between users in real time. We will be using the Firebase realtime database to solve this issue.

Dependencies:
Player choosing to play against other users, Firebase

Add bots

Description:
Our system gives users an alternative to playing against other people by allowing them to play against bots. The user will have the option to either add players or bots into their lobby.

Criticality:
This is a critical feature as it allows our game to be played regardless of availability of opponents or user's connection to the internet.

Technical issues:
There is an issue of making bots that can solve the challenges presented in the game without being too efficient at it. Bots that cannot finish the rounds are useless while bots that do it beyond human capability are not fun to play against.

Dependencies:
Player choosing to play against bots.

Letter round

Description:
The first round of our game is the letter round in which the player selects a random vowel or consonant 9 times and then attempts to create the longest word possible from the available letters.

Criticality:
Critical feature due to the fact that this round is a core part of the game rules

Technical issues:

There is an issue of checking the validity of words given by the players which is solved by using a database containing a file with a list of all english words.
To improve performance we plan to be able to scan the zipped file without having to extract it.

Dependencies:
Database (Firebase), English dictionary file

Letter round solver
Description:
Our game implements a feature of finding the best possible solutions for the letters round using our algorithm that compares longest possible combinations of letters given to the player with the list of all english words.

Criticality:
Low priority feature

Technical issues:
Finding the best answers in an efficient manner.

Dependencies:
Participation in the letter round, database, English dictionary file

Number round
Description:
The second round of our game which is the numbers round in which the players pick 6 numbers from 24 possible numbers. A random number is generated and the contestants try to manipulate their numbers and arithmetic operators in order to get a number that is as close as possible to the number generated.

Criticality:
Critical feature due to the fact that this round is a core part of the game rules

Technical issues:
None

Dependencies:
Player finished the letter round,

Number round solver
Description:
Our game implements a feature of finding the best possible solutions for the number round using our algorithm that uses greedy algorithm to search a tree made of permutations of possible operations of the numbers given to the player.

Criticality:
Low priority feature

Technical issues:
Finding the best answers in an efficient manner

Dependencies:
Participation in the number round

Conundrum round
Description:
The third round in our game is the conundrum round in which the player gets 9 random letters and has to use all of them to create an english word.

Criticality:
Critical feature due to the fact that this round is a core part of the game rules

Technical issues:
Same as letters round

Dependencies:
Player finished the number round, database, english dictionary file

Conundrum round solver
Description:
Modified letters round solver

Criticality:
Low priority feature

Technical issues:
Finding the best answers in an efficient manner

Dependencies:
Participation in the conundrum round, database, english dictionary file

Scoreboard
Description:
Displays the winner as well as the scores of all the players in the match

Criticality:
Low priority feature

Technical issues:
None

Dependencies:
All rounds completed, database