

# LetterGrab - Software spec

## LetterGrab

Inventor: Moshe Fiddle

## Overview

This document describes the requirements for the software implementation for a newly invented game called “LetterGrab”. A detailed description of the game and software implementation are provided below.

## The Game

### Players:

LetterGrab is a two-player game.

### Game Description and Rules:

- The game begins with an empty 10x10 grid.
- Each player will use their turn to try and fill as many empty spots by filling them with one word per turn.
- Words can be inputted either horizontal, vertical or diagonal. Horizontal must flow from left to right, vertical words must flow from top to bottom and diagonal words must flow left to right and top to bottom.
- Players can use the existing letters in the grid for their new inputted word. The additional filled spots will be acquired to the new player, and the original spots will remain with the original player
- At least two additional inputs are required for forming a word.
- Players have the choice to forgo their turn if they don't see any word to add. When both players forgo their turn consecutively, it's game over and the winner is calculated.

### Points Calculation:

- Points are simply calculated – one point is earned per filled spot.

### The Winner:

- The player who fills the most spots is the winner of this game.

### Score Calculation:

- There are two score calculations for LetterGrab –
  - Game Score – calculates the score of the current games.

# LetterGrab - Software spec

- Session Score – calculates the total score of all games.

## Game Specifics:

- The two players are defined as Red and Blue.
- Red is the first player.
- As the players take their turn, their current total score and all-time score is displayed on the score board.
- At the end of each game the players can choose whether they wish to begin a new game or a new session. The max number of games per session is 10.

## Software Implementation

### UI Elements:

- Header Label with name of game and tagline. Maybe we'll add a logo as a pic...
- Message Label to display current turn, winner or error.
- 10x10 grid of textboxes.
- All buttons should display descriptions when hovering over button.
- Row of buttons beneath the grid as follows:
  - Forgo Turn – when giving up and switching turn.
  - Refresh Input – to clear attempted word and input new one.
  - Submit- to submit inputted word.
- Row of buttons above the grid –
  - New Game – Starts new game.
  - New Session – starts new session.
- Score Board
  - All below scores are to use an identical format.
  - Score for current game displayed at top in larger format.
  - As the games are completed, the total score per games, are added below, beginning with game one at top. Max games - 10
  - Total Session score is displayed on bottom in semi-large.

### Critical Code Structure:

#### Form Level Lists:

- Iinstallwords – built in dictionary.
  - For procedures and constraints for each text box.
- Iinstalltxtboxes- master list of all textboxes.
- Iinstallsets - *List <List <List<TextBoxes>>>*
  - Iinstallrows- *List <List<TextBoxes>>*

# LetterGrab - Software spec

- Each row list- *List<TextBox>*
- *Istallcolumns*
  - Each column list of textboxes
- *Istalldiagdown*
  - Each diagdown list
- *Istredacquired* – list all acquired by red.
- *Istblueacquired* – list all acquired by blue.

## *When Player hits submit:*

- Procedure captures current inputs as a list by order of *Istalltxtboxes*.
- Confirms new list has more than one input in only one *Istallsets*. Otherwise returns error.
- If no *Istallsets* is returned display error.
- The missing letters already inputted are acquired in the correct position from *Istallsets*, as a new list.
- Procedure confirms there is such a word. If not an error returns and player gets another chance.
- If yes, the inputted textboxes are added to the players List. Any textboxes already in the other players list are omitted.