# DoorWords

Created by: Rivky Schnall

## Overview

This document describes requirements for the software implantation of DoorWords. This software will allow one person to play against himself. A description of the game plus the requirements of this implementation is provided below.

## The Game

When the game starts, the mystery word is generated. The user will be presented with a set of unique doors. Behind each door is one of the letters of the word. Users can view the scrambled letters one at a time by opening each door. Doors are opened and closed by clicking on them. After all the letters are viewed, the user will be shown the unknown word. The user will be prompted to spell out the word by clicking the doors in the correct letter order of the word.

If the player gets a word right, he gets a correct point, otherwise known as a key Ñ (Found on the character map – font: Webdings, character code: 0xD1).

If the player gets the word wrong, he gets an incorrect point, otherwise known as a lock Ï (Found on the character map – font: Webdings, character code: 0xCF).

Player wins at 5 keys or loses at 5 locks, whichever comes first.

The player can also pick how many doors will be active in each game. This determines how long the word will be. The options are four to eight.

## Software Implementation

### UI elements

The software will present the DoorWords game with the UI elements listed below.

* 3 rows on top like a toolbar:
  1. Display row: a label which will display game status and instructions. In this spec, the label will be referred to as the screen.
  2. Controls row: will have 2 buttons side by side, One will be a new game button, the other will be a next button. Will also have radio buttons for the number of doors 4-8.
  3. Score row: Will display 5 inactive keys and locks. Will also say 5 keys = win, 5 locks = lose, one on top of the other.
* 4 x 2 grid of doors (picture box).
* Label displaying word to be spelled out by clicking on doors. There should be a question mark for each letter.

### Game Process and Rules:

#### Process

* When initialized, all doors should be inactive. The screen should display “Welcome to DoorWords! A word is scrambled and its letters are hiding behind closed doors. Peek behind each door and memorize its letter. When the unscrambled word is displayed, click the doors according to the letters of the unscrambled word. Set the game options and click “Start Game” to begin.”
* When “Start Game” is clicked, game is active. Screen should display “Click on a door to view its hidden letter. Only one door may be opened at a time. Doors must be opened at least once and at most three times.”
* When all the doors are clicked at least once, the screen should display “Click Next to continue”.
* When “Next” is pressed, the mystery word is shown. Screen should display, “Click the doors according to the order of the letters in the word below.”
* Player can now click each door once. As each door is clicked, behind the scenes the software should catch each letter, building a word.
* If the player won the round, the screen should display “Great Job! You have earned a key. Press Next to continue the game.” 1 key should also become active.
* If the player lost the round, the screen should display “Try again! You have earned a lock. Press Next to continue the game.” 1 lock should also become active.
* When the game is won, the screen should display “Wow! What a win, you are a memory master!” The board should change to an exciting color.
* When the game is lost, the screen should display “Aww! Keep on playing to improve your score.” The board should change to a dull color.

#### Rules

* Doors can only be clicked once to build a word. (Take for example the word “puzzle”. Two separate doors will contain the letter “z” behind them. The user will need to click each one to build the word, but the order doesn’t matter.)
* If a player wants to abort the game and start again, he can click “start game” at any time.
* Only the number of doors in the game should be active.
* The “Next” button should only be active when applicable.
* When a door is clicked:
* If it’s closed, it should “open” and the picture should change to the open door version with the letter in the doorway.
* If the door is open, it should “close” and the picture should change to the closed door version.