CS 4985 – Fall 2013
Assignment 3, 40 pts.
Due: Friday, Oct. 18th at 9a.

Specifications

This assignment will implement the Connect64 game as discussed in class. The assignment will be broken into three parts. Part 1 is worth 30 points. Parts 2 and 3 are worth 5 points each. You are not required to complete Parts 2 and 3. Part 2 will not be graded if you do not earn 25 of the 30 points on Part 1. Part 3 will not be graded if you did not successfully implement the timer or 10 top scoreboard for Part 2.

The details of each part are given below. Do not move onto the next part until you have successfully completed all of the previous part.

Any project that does not compile will receive a zero.

Getting Started

1. Create an new Android project as follows:

    a. Project name: YourNameConnect64

    b. Build target: Android 4.3

    c. Application name: Connect64 by *Firstname Lastname*

    d. Base package name: edu.uwg.*firstnamelastname*.connect64

Part 1

Part 1 of the project will consist of providing the following functionality:

1. Create 10 different starting puzzles that increase in difficulty. The player will automatically advance from one puzzle to the next after successfully completing the prior puzzle.

    a. The UI should display the current puzzle number when playing the puzzle.

2. Display the puzzle in an 8x8 “grid” with its starting numbers in place and allow the player to begin playing the game. The gameplay should allow and enforce the following:

    a. The player should not be able to edit any of the provided starting numbers.

    b. The user should be able to enter a number within a cell.

        i. The number entry approach is up to you. When entering the number, if the number has already been entered or is not within the range of 1-64, the program should indicate so to the player. Note: the player should be allowed to enter a number in an open space that is not currently touching a numbered cell.

3. When the board is completed, it should be evaluated to see if it correctly completes the puzzle. If not, the player should be informed.

4. The game should allow the player the ability to delete numbers entered in cells and reset the entire board back to its starting point.

5. The game should remember the current puzzle and board entry for the player so that if the player quits they can return to their current board and current state.

6. The player should be able to select a particular puzzle to play, even if they have already played that puzzle.

7. The game must be playable on both phones and tablets. You may lock the orientation in landscape. The design of the UI and overall look and feel is completely up to you. A portion of your grade will be based on the overall look and feel and usability of the UI you develop.

Part 2

Part 2 of the project will consist of providing the following functionality:

1. Adding an additional five puzzles for a total of 15 puzzles.

2. Providing a displayed timer that displays as the player plays each puzzle.

    a. If the player exits the game, the timer should start from where it left off.

    b. If the player resets the game, the timer should be reset to 0.

3. A top 10 scoreboard that will display the player's name, the length of time, and the puzzle number that it was achieved on. The scoreboard should be sorted according to the length of time it took to complete a puzzle in ascending order. There only needs to be a single top 10 scoreboard for the entire game. I.e., you will not create a separate scoreboard for each puzzle. The player should be able to view the scoreboard and also to reset the scoreboard.

    a. The top 10 scoreboard must be stored in a database.

Part 3

1. Adding an additional five puzzles for a total of 20 puzzles.

2. Allow the ability to pause the game without exiting.

    a. If the game is paused the puzzle board should not be visible during this time.

3. Add ability to provide the player aural and/or haptic feedback.

    a. The aural and/or haptic feedback should be employed when the player enters an invalid number or does not complete the puzzle correctly.

    b. If the player successfully completes the puzzle, then an appropriate sound should be played.

4. Add user settings for the following:

    a. Turn on/off aural and haptic feedback. Aural and haptic feedback should be mutually exclusive of each other.

    b. Change the default color of the cells and numbers.

    c. The settings should be persistent through various invocations of the game.

5. WOW factor. Add some additional functionality to your project that was not specified. Please note your WOW factor feature(s) in your `status.txt` file.

Submission

Add a status.txt file to your Eclipse project and in this file detail what parts of the project you completed successfully, if there are known issues, please detail them. Additionally, if you implemented a WOW factor for part 3, please note what it is.

Export your project in Eclipse, naming the archive file *FirstnameLastName*Connect64.zip, e.g., JohnDoeConnect64.zip and submit the file in Moodle by the due date.

Grading breakdown

Part 1 – 30 pts.

- 24 pts. – Required functionality.
- 3 pts. – Design and usability of GUI.
- 3 pts. – Class design and code implementation.

Part 2 – 5 pts.

- 1 pt. – Additional puzzles.
- 2 pts. – Timer
- 2 pts. – Top 10 scoreboard with database.

Part 3 – 5 pts.

- 4 pts. – Requirements 1 – 4.
- 1 pt. – WOW factor.