CASSANDRA CHAPUT | CCHAPUT

MICHAEL BUZIL | MBUZIL

CONNECT-4 FINAL PROJECT

CSCI-C323

4/28/2023

COURSEWORK SUBMISSION INSTRUCTIONS:

INSTRUCTIONS

- start by connecting a iphone or ipad to the device running xcode
- if propted choose to trust the computer
- configure to run on your personal device (iphone / ipad)
- run the project / open the Connect-4 app
- enter in a name for player 1
- click the arrow button
- enter in a name for player 2
- click the arrow button
- player 1 tap the connect-4 board to drop a chip in the column
- player 2 tap the connect-4 board to drop a chip in the column ** NOTE: the goal is to get 4 chips of your color in a row horizontally, vertically, or diagonally **
- continue swwapping turns tapping the board to place chips
- when a player connects 4 of their colored chips the device will release a continuous haptic vibration for 1.5 seconds
- press the end game button
- press the tab bar item at the bottom of the screen to view other information
- go to the history tab to see the results of previous games
- go to the game tab to decide on next game
- press new game for the same players as last time play on a new empty board
- press the new player's button to create new players and a new empty board

XCODE ENVIORNMENT INFO

** Final project / connect-4 requires iOS hardware for testing!! ** iOS Hardware used for testing: iPhone 12 Pro Max iOS Version for hardware testing: iOS 16.3.1

REQUIREMENTS & WHERE

1. Achieves MVC architectural pattern MODEL: Connect4Model.swift This file does the following:

- saves the game number of games and the winner and loser from that game
- saves the player1's name
- saves the player2's name
- clears the player's names for new player's button CONTROLLER: GameTableViewController.swift, playerInfoViewController, etc.
- all of these controllers manage views in the storyboard
- allows user interaction (like textfields) to store into the model (ex: player1 & player2's names) VIEW: GameBoardView.swift, main storyboard
- the GameBoardView was a UIView that we used to draw the connect-4 board
- the main storyboard is the view the user will see and it is controlled by the viewControllers

2. USER INTERFACE INPUT:

- GameBoardView.swift contains the code to interpret touch
- playerInfoViewController.swift contains code for UITextfields and UIButtons OUTPUT:
- GameBoardView.swift contains the code to draw CoreGraphics to make the connect-4 board & allows the device to recieve haptics VIEWS (3):

1. tableView with a table view controller to display game history

2. UIView for connect-4 drawing

3. tabView containing tableView and GameOverView

4. Connect 4 was never done in class!

5. core data used to make game history ppersistent & appear in the table view

6. FRAMEWORKS IN CLASS (2)

- CoreData to save game results
- CoreGraphics to draw the game board

6. FRAMEWORK NOT COVERED IN CLASS (1)

- CoreHaptics used to notify that someone connected 4