Developer’s Guide

Overview

The Tic-Tac-Toe Trivia game consists of 3 main components: the main Game class, the Board class and the Question class. The main Game class controls the main menu of the game, allowing the player to start a new game, read the instruction, and exit the game. When a new game is started, the Board class is called. This creates and controls the multiple GUIs components of the game, such as the JButtons that make up the Tic-Tac-Toe grid and the choices for the trivia. In addition, three listeners (ButtonListener, ChoiceListener, and TimeListener) help control the actions that are performed when the buttons are pressed, as well as perform tasks related to the timing of the game. For the trivia portion, the Question class is used to read the questions, the choices, and the correct answer from the appropriate file. This list of questions is then randomized and used to supply the game with random trivia questions.

Component Features

**Game Class**

The Game class creates a main menu which allows the player to either start playing the game, read the instructions, or exits the game. If a new game is chosen, the user is allowed to choose which category of trivia they want to play. Based on the button they select, a char variable will be assigned. This is used to call the Board class, which in turn uses the char to appropriate declare the Question. Once a new game has started, the main menu is set to be not visible. In the main method of this class, the class is called, and a sound file is played for background music.

**Board Class**

The Board class creates the GUI of the game. It creates two JPanels: one for the Tic-Tac-Toe grid, and one for the questions and answer choices. The appropriate JButtons are added to the JPanels, and the two panels are added to a JSplitPane, which is added to the JFrame. For the panels, images are used for the backgrounds, so the paintComponent() method is overridden. There are two separate ActionListeners that are added to the buttons in the grid and the buttons in the other panel. For the grid, the ButtonListener is added. When it is fired, it first checks whether or not the player is currently answering a question. If not, then the text in the textArea of the panel will be changed to the current question. Then, the buttons on the panel will be changed so that they have the 4 possible answers to the current question.

When the choice buttons are clicked, they trigger the code found in the ChoiceListener. It will check if the answer is correct; if it is, then the icon on the selected grid button will be changed appropriately. This is done because when the ChoiceListener is declared, the ActionEvent of the ButtonListener is a parameter, which allows identification of the button pressed. The number of the tile on the grid will be added to an array, based on whether or not the answer is correct. Next, the program will check if the game has been won or lost by using a loop to compare the correct and incorrect arrays to the 2d array with all the winning combinations stored.