Ryan Bentz

ECE 558

Project 2 Report

10/29/18

App Architecture

The app starts with a splash screen that contains a spinner to let the user choose a quiz. It then moves to the quiz activity (the central activity of the app). Quiz activity manages the quiz process by invoking an instance of the quiz manager class to manage the quiz information. This leaves quiz activity free to actually manage the information on the screen. Once the quiz is done, the app moves to the score activity where the user score is shown and they can enter their name. After the user submits their name, the app determines if they were a new high score for the quiz. If they were, the user adds them as the new high score. The app moves to the high score activity where the user with the highest score is shown. On return, the app moves back to the main activity splash screen. See figure 1 for more information.

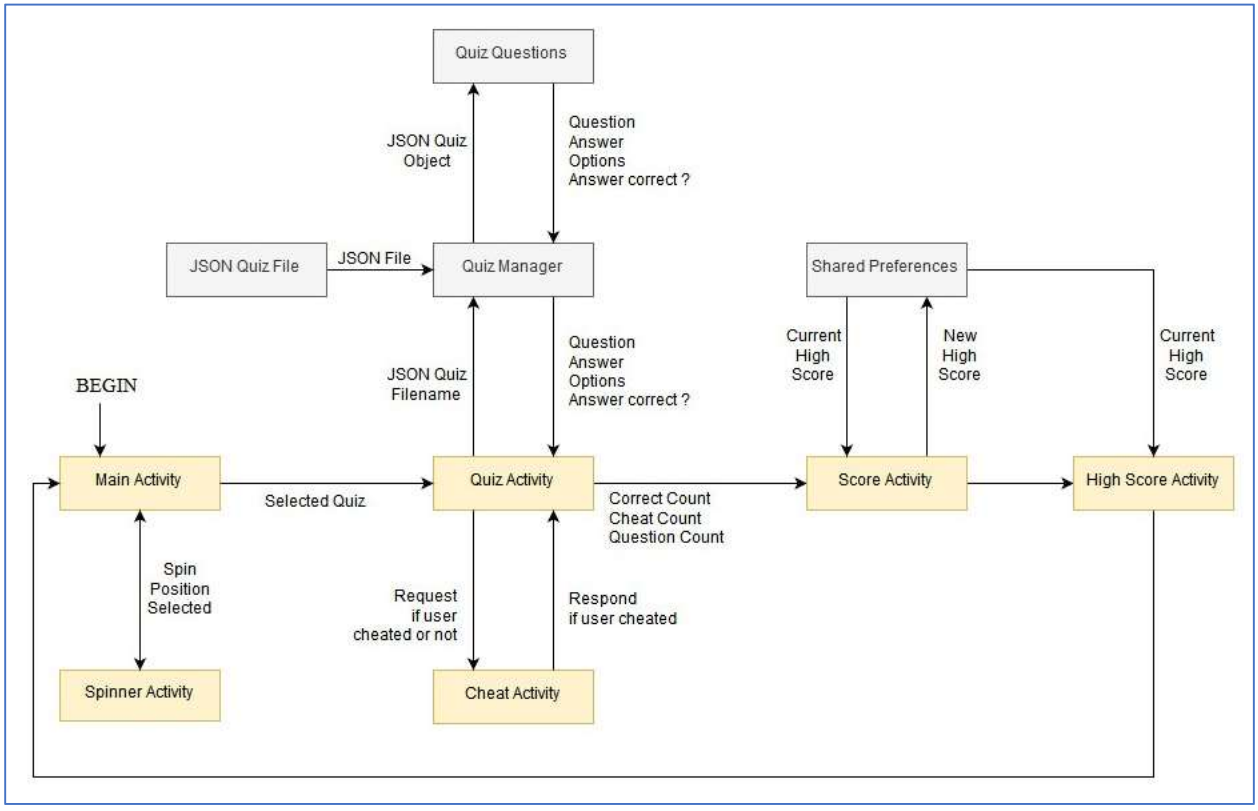

Figure 1 – Quiz App Architecture
By: Ryan Bentz, 10/29/18

JSON Quiz File

Quizzes are stored in JSON format. The quiz JSON file contains a single quiz JSON object made of an array of JSON object quiz items. Each quiz item is a JSON object with question and answer fields and a JSON array of answer options. The design of the JSON file is as follows:

JSON Object → JSON Object → JSON Array [JSON Object [JSON Array]]

QuizManager and QuizQuestions

QuizManager takes in a filename for a JSON quiz and accesses the JSON file in the assets folder. It parses the JSON file into a string. That string is then parsed into JSON objects for each question. Each JSON quiz question object is passed to an instance of QuizQuestion object that parses the object data into the relevant strings for the question, answer, and answer options. QuizManager keeps an array of QuizQuestion objects as the entire quiz and iterates through them as the user takes the quiz. QuizManager acts as the intermediary between the quiz data and QuizActivity. It retrieves the text for the answers, questions, and options on behalf of QuizActivity and forwards requests to the specific QuizQuestion object to evaluate if an answer is correct or not.

To evaluate if an answer is correct or not, QuizActivity gets the integer position of the radio button that was selected and passes it to QuizManager. QuizManager calls the specific method for the current QuizQuestion which takes the integer and gets the text at that answer option array index. It evaluates the text of the answer and the text at that array index and returns true/false on if the text or not.

Cheating

The user has the option to cheat via a separate activity. QuizActivity calls CheatActivity if the user is thinking about cheating. CheatActivity returns a result code if the user wants to cheat or not. If the user wants to cheat, QuizActivity gets the text for the answer and displays it in a toast. If the user still selects the wrong answer after cheating, they are not deducted points from it. Only correct answers after cheating are deducted from the final score.

Score Results

Score results are displayed in ScoreActivity. The relevant information is passed to ScoreActivity from QuizActivity via Intent Extras. The ScoreActivity gets the user name from the EditBox and the current high score from shared preferences. The app stores separate high scores for each quiz. It compares the current score with the high score and if the user has the new high score for the quiz, it will put them in as the new high score. It saves the high score in shared preferences. The HighScoreActivity gets the high score and name for the specific quiz from shared preferences and displays them on the screen.

Handling Device Rotation

On device rotation, the activity is destroyed and created again. In the MainActivity (splash screen), there is no data that needs to be saved. During testing, the selection on the spinner was persistent across

rotations. In QuizActivity, the activity saves the cheat count, correct count, quiz name, and current question number. On create activity for the first time, everything but the quiz name is initialized to zero. If the On Create is called with a valid instance of savedInstance, then we get the values from saved instance. The QuizManager constructor uses these values to build the quiz and either start from the beginning or start from where we left off in the case of OnCreate due to a rotation.

In the ScoreActivity and HighScoreActivity, both get their persistent information from the intent of the parent activity and so no information needs to be saved in savedInstance.

App Algorithm

Main Activity

```
Choose a quiz from the spinner
If invalid choice
    Notify user
Else
    Save selected quiz
    Go to Quiz Activity
```

Quiz Activity

```
Check if restarting after destroyed for rotation
If starting after destroyed rotation
    Get values from saved instance and use to populate members in QuizManager
Else
    Start QuizManager with initialized values
Get selected quiz
Start Quiz Manager
    Build Quiz
        Read file from assets
        Parse JSON file
        Create QuizQuestion array
        Parse JSON Quiz Objects to QuizQuestion Objects
Fill in the activity widgets with the quiz information
Set up radio button widgets
Set up button widgets
    Cheat Button
        Go to cheat activity with request if the user cheats or not
    Answer Button
        Get selected radio button
        Check if user got the right answer
        If correct answer
            If user cheated
                Notify user: cheater
                Increment cheat count
            Else
```

Notify user: success
Increment correct count
Else
Notify user: incorrect
Clear radio button selection
Check if there is another question
If next question available
Set next question text
Else
Save quiz results
Go to Score Activity

Cheat Activity

Set up button widgets
Yes Button
Set YES result and return to Quiz Activity
No Button
Set No result and return to Quiz Activity

Score Activity

Get quiz results
Show quiz results on screen
Set up button widget
Submit Button
Get high score from shared preferences for specific quiz
If user score > high score
Set user score and name as new high score in shared preferences
Save quiz name for High Score Activity
Go to High Score Activity

High Score Activity

Get selected quiz from Score Activity
Get high score from shared preferences for specific quiz
Show high score on screen
Set up button widget
Finish Button
Return to splash screen