

CS242 Final Project Rubric (ykim164)

Week 3 – Upgrade User Convenience

In the third week of the final project, I will be upgrading various features of the application to improve user convenience. For example, I will be developing options for Admin Users to change settings of their account. Also, the information of the added games will be editable so that users can fix any mistakes when creating a new game. There will be more features that improves the overall quality of the application to be implemented and those features are listed below.

- **Requirement 1: Refactor / Improve Admin Functionality**
 - Make added games editable
 - Teams selectable only from actual teams in the tournament.
 - ◆ When adding a game, teams should be picked up from the actual list instead of being typed manually.
 - ◆ Prevent selecting same teams
 - Admin Main Menu
 - ◆ Create a menu screen where the admin user can:
 - Sign out
 - Change channel name
 - Delete account
- **Requirement 2: Automation of League Table Update**
 - The points will automatically update based on the final game status (game status updates when the "End Game" button is pressed) and will be displayed in the league table.
 - ◆ e.g. If the Team A beats Team B, 3 points will be added to its points and it will automatically update the league table based on this.

- ◆ Sort the league table based on its points and goal differences (if two teams have same points, then they will be sorted based on the goal difference).
- **Requirement 3: General Improvement of User Convenience**
 - Check possible duplicates in scorer names when updating the game score information.
 - ◆ Using a Levenstein Distance algorithm, check if the scorer database contains the similar name as the input.
 - ◆ If the database contains a similar name, suggest an option for using the correct name or adding a new scorer with the input name.
- **Requirement 4: Notification Settings**
 - Create an option for client user to enable notifications on goal updates
 - ◆ For all games
 - ◆ For followed games only
 - Provide various types of notifications:
 - ◆ Vibration
 - ◆ Sound
 - ◆ Header (pop-up)
 - Set a relevant notification message
 - ◆ e.g. "YJ (Team Eagles) scores a goal! Eagles 1 – 0 Tigers"
- **Requirement 5: Testing**
 - Manual test plan for notification and name similarity.
 - Unit test for checking name similarity and league table sorting.

Grading

Category	Weight	Scoring	Requirements
Basic Preparation	2	0-1	Ready to go at the start of section
Cleverness	2	0-2	The hardest points on the rubric
Rubric	2	0-1	
Code Submission	4	0-2	Submitted correct content on time and to the correct location in the repository
Decomposition	4	0-2	Project is adequately decomposed to different classes and methods
Documentation	4	0-2	Comments for each class and each function are clear and are following style guides
Effort	2	0-2	Perform considerable amount of work
Naming	2	0-2	Variable names and method names are readable and are following the naming conventions of the language you chose
Overall Design	5	0-2.5	Have nice approaches and structures in overall
Participation	5	0-2.5	Interact with the group 2 times (ask a question, make a comment, help answer a question, etc.)
Presentation	4	0-2	Present the code clearly
Requirements - Refactor / Improve Admin Functionality	5	0-2.5	2 points: Create a menu screen where the admin user can change channel name, sign out, and delete account. 2.5 points: Make game information editable and make teams selectable from actual teams from the database.

Category	Weight	Scoring	Requirements
Requirements - Automation of League Table Update	5	0-2.5	2 points: The final game score will be updated in the database when the "End Game" button is pressed. 2.5 points: Automatically update the league table based on the final game score and sort the updated league table based on points and goal differences.
Requirement – General Improvement of User Convenience	5	0-2.5	2 point: Implement a function for checking two similar names. 2.5 points: Use the Levenstein algorithm to check whether duplicates exist in the scorer database and provide options to add the correct scorer.
Requirement – Notification Settings	5	0-2.5	2 point: Create an option for the client users to enable notifications on goal updates. 2.5 points: Users should be able to make detailed settings (range of notifications and notification types)
Testing	5	0-2.5	2: points: Manual Test Plan 2.5 points: Tests for League Table sorting and name similarity.
Schedule	2	0-1	Revised final project schedule if necessary
Total	63		

Brief Rubric for Week4

- **Requirement 1: Finalize design and make it fully adaptable for all conditions**
 - Polish the design of the app to make it pretty (use better fonts, change colors, fix ugly layouts, set background images).
 - Change all layouts to LinearLayout so that it is perfectly responsive to all kinds of sizes.
 - Minor debugs

- **Requirement 2: Provide details on list item clicks**
 - Users should be able to view detailed information of items of all list views when clicked.
 - ◆ Team information (game statistics) on League Standings.
 - ◆ Scorer information (when did the player score) on Score Table.
 - ◆ Game timeline on the game list view.

- **Requirement 3: Live Game mechanism**
 - Implement a way to display exceeded time for each game (make it both pause-able and resume-able).
 - Capture the game time whenever the game is updated

- **Requirement 4: Share Application**
 - Make an option for sharing the app information (text would be fine for now) through email, messenger, Facebook, etc.

- **Requirement 5: Testing**
 - Manual test plan
 - Unit Test

