Proposal for Quidditch Scoring Applicaiton

by Matt Buland

Purpose

This app will allow users to keep score of their muggle quidditch match (International Quidditch Association version 7).

Game Mechanics

Quidditch matches end either when the snitch is caught, or when regulation time is reached. Regulation time is typically 10 minutes. Afterwhich, the game will either go into overtime if it will result in a tie, or will end, giving the victory to the team with the most points. 10 points will be awarded to a team that throws the quaffle (a ball) through one of the opposing team's hoops. When the snitch is caught, the team that catches the snitch receives 150 points, and also ends the game, as mentioned above. These are the only ways to score points in quidditch. Note about catching the snitch: If the snitch is caught, and the game results in a tie, the game is to go into overtime.

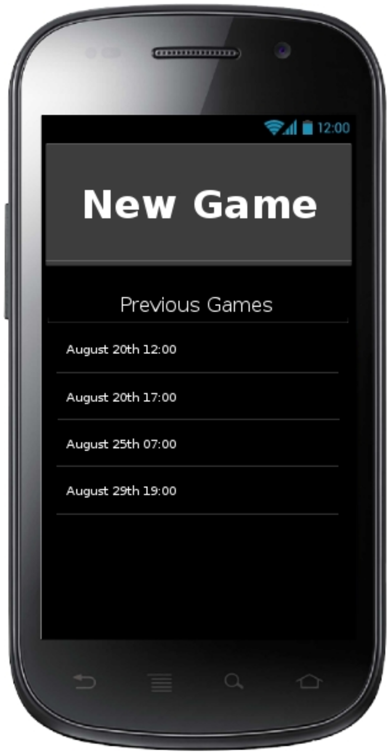
User Interface Interaction

The start screen greets the user, providing the option to view an old game, or to start scoring a new game. If a game is currently running, the “New Game” button will instead show “Resume Game”, and allow the user to view the scoring screen for the currently running game. It's important to note that only one game may be tracked at a time. Selecting an old game will take the user to the summary screen, displaying the old game. Pressing the back button on this screen will close/stop the app. If a game timer is running, it will continue running in the background. Beginning a new game will take the user to the scoring screen.

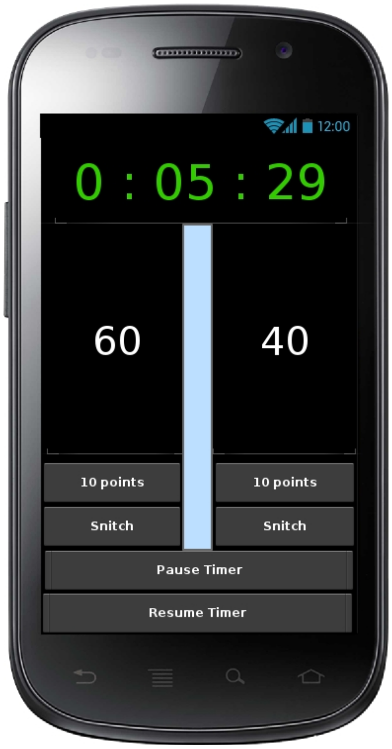
In the middle of a game, the scoring screen will be active. This is divided into two sides: one for each team. The sides are not named, and are only distiguished by the side of the screen they are on, though a naming feature could be added. When a team scores 10 points by throwing the quaffle through the opposing team's hoops, the “10 points” button should be pressed, adding 10 points to that team's score. When a team catches the snitch, the corresponding “Snitch” button should be pressed, adding 150 points to that team's score, and pausing the game timer. A toast will be made, indicating the winning team. In the case of a tie, the left-team will be chosen, to decrease the complexity of the app. After a full second, the app will move to the summary screen for the game that was just completed. If the back button is pressed (either in the middle of a game, or in the 1-second ending period), the app will show the introduction screen. If a game-timer is still running, it will continue in the background.

Viewing the results of a game is accomplished with the summary screen. The final score will be shown (in left-right correspondence), and an indication of which team caught the snitch will be shown in the form of a text-view. The final game-time will also be shown. Pressing back on this screen will always take the user to the introduction/start screen.

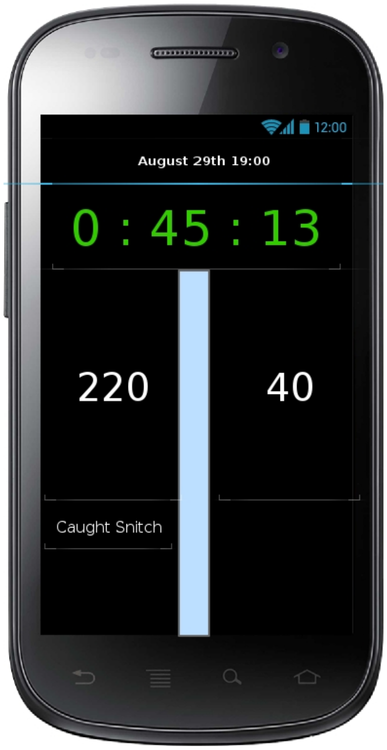
Introduction Screen



Scoring Screen



Summary Screen



Information Flow

The summary screen needs to know how many total points were scored. The raw data of these scores will be stored in an sqlite database on the SD-card (or internal memory, depending on what's available), which will be able to store many games. The scoring screen will pass an integer key (via the activity-starting intent), to the summary screen, which will use the value to lookup a specific game in the database. This method will allow for larger statistics to be ran on game storing beyond just the result of a single game, and will also cut down on the amount of data that's transferred via intents.

Intermediate data storing

When the android system transfers the running app into a paused or stopped state, the current game will be saved into the sqlite database, and will be marked as still running, so that when the app is resumed, it can easily be retrieved, and resumed. However, the timer for the running game will only be stopped when the activity is destroyed. This will allow uses to multi-task, and do other tasks without interrupting the game-timer. This does assume that the game would still be on going.

Extended/Stretch features:

- Notification-bar interface
- Notification that the time is still running
- Extended statistics (averages)
- Individual Player Scoring tracking (could add time-based statistics per person)
- Named teams
- Scoring corrections