Adrian Romano

December 6th, 2010

Final Project Report

Introduction

For the project, I chose to combine the reusable library presentation with the final project. I created an API for ESPN fantasy football. It allows the user to login to his ESPN account and perform operations that are normally only available via the website. I combined the API I created for the presentation and created a GUI for the final project.

Implementation

The fact that ESPN doesn't actually have an official API, or way to login and make changes without going through the web interface, presented a problem at first. I had to actually create the functionality from what was available from the m.espn.com website. This was a bit tedious, because it required me to parse out the relevant information from an HTML page designed to be shown in a browser, but there was really no other option. I decided to implement a few basic functions including view team players, view scoreboard, and view standings. I also did the advanced function of moving players. I made extensive use of the indexOf method of the String class in order to parse the pages.

When I created the GUI for the presentation, I just wanted to show the different parts of the API. For the project, I wanted to make something more functional. The main thing I changed was on the 'My Team' section. The API only allows a player to be moved from one position to another, rather than swap one player for another. I added the swap functionality by combining API calls in the new GUI. When a player wants to move a player in the project version, he selects the player by pressing the fire button. All available positions that the player can be moved will be highlighted in yellow. The user can select one of those positions by moving the selector to that player and pressing fire. This will swap the players. The other thing I added was a score monitor. There is a thread that gets the user's score and plays a sound effect if the score has changed since the last time the score was acquired. The thread starts as soon as the user selects the league, so as long as the program is running, the thread will be running.

Problems and Issues

The first issue I ran into was with HTTPS. ESPN requires an HTTPS connection to log in, but the Java ME emulator does not have any root certificates, so I was receiving an HTTPS exception. I solved this by importing the root certificate that the ESPN cert was signed with. To import the certificate, I had to export the root certificate from my web browser, and then use the java keytool and mekeytool to import the certificate into the keystore of Java and the emulator instance.

The other big problem I had while developing was testing. I was limited to my own ESPN account and fantasy football team. I was not able to create a new league/team this late in the season to test with. When I started the project I wanted to be able to add and drop players, but I did not have a viable way of testing it, so I scrapped it. It would have required me to drop my players, which I was not willing to do. The other problem I had with testing was that the structure of the team page changes when players are locked. This only happens from a player's game time until the following Tuesday morning. This is usually some time on Sunday, although late in the season there are games on Thursday's as well. Because of this, I did not notice it until I was testing on a Sunday and my code broke. I subsequently went back and fixed it, but it just goes to show how brittle this method is.

The final problem is the overall brittleness of the API. Whenever ESPN changes the structure of one of their pages, the API code needs to change in order to parse it correctly. It would be nice if ESPN would expose some kind of XML page or something similar so developers could easily make an ESPN program, but I suspect that is not going to happen.

Conclusion

I accomplished all the things I wanted to when I started this project except the add/drop player function. I was still able to create a functional program that allows the user to manage his ESPN fantasy football team. I believe it is easier to use than the m.espn.com page even though it doesn't have all the functionality.