Antoine Saliba

Laker Books

Maintenance Plan

**API**

The application relies heavily on the DirectTextbook.com API to retrieve information on textbooks based on the ISBN and title. This API has numerous problems with their ability to retrieve information based on the title of the textbook. This was discussed with them throughout the semester and for the most part fixed. However, there are a few instances where their API does not behave properly and returns weird errors. To prevent this from breaking the application, I would create a database to keep information on all textbooks. Every time a user decides to sell a textbook on Laker Books, the database will be checked first. If there is no information on that specific textbook, then the DirectTextbook.com API will be used to retrieve the information. This information will then be stored in the database and every time another user wants to sell the same textbook, the information on that textbook will be retrieved from the database. This improvement allows for the website to rely less on the API and not need to make as many requests to the API on a daily basis. This will also reduce the chance that the API request returns an error because of an error on their end.

**Authentication**

Currently, to authenticate the emails that are entered in by the user to either sell or buy a book, the email is first checked to see if it ends with ‘@oswego.edu’. If it does, it then checks an excel sheet that contains all currently enrolled students to see if it finds a match. This is not practical because, once the number of enrolled students changes after the semester, the list will be outdated and there will be missing email addresses from the excel file. This was done this way for now because getting approval from the university would have taken too long and there was no way to parse the directory full of email addresses that we all have in our Gmail contacts. To improve upon this, I would go through the process of getting an online list of all currently enrolled students from Oswego. This list would come from the database that has a record for every SUNY Oswego student and would always be up-to-date. As a result, it would never be outdated and all students would always be accounted for. This would prevent individuals who have graduated from SUNY Oswego from using the website and also for new students to be able to use it without any problems.

**Textbooks**

While doing this project, the textbooks were never taken offline after being on the website for a certain amount of time. Also, when bought, the books would remain in the database, with a record of who bought it and who sold it. This would not be practical because the size of the database would quickly become very big. To prevent the database from getting too big and for books to appear on the website for so long that the seller has already figured out something else to do with the book, a time stamp would be used of when the book was posted on the website. Regardless of whether the book was sold or not, it would be removed from the database and also the website within 4 months of it being posted. This gives enough time for a user to report a bad transaction and for there to still be a record of it in the database and also provides a way to regulate the size of the database so that it does not get too big with old information.

**Code**

All of the code for Laker Books is located on GitHub and is available for everyone to see. This will also allow anyone a place to proposes fixes or bugs in the application. The more people that will look at the code behind the application, the better the application will be. This also acts as a way of reassuring people of what the application does. By looking at the source code, one can see that no information is taken from the user other than what is needed. If any changes are made to this application, people will be able to see them. Also, this gives people a way of proposing improvements that could be made for the application in future releases.

**Stress Testing**

This website will have its most traffic during the first few days of each semester and the last few days of each semester. This is because most students are trying to get rid of their books and buy new ones during those time periods. To make sure that the application can keep up and that it can handle all of that traffic without slowing down, stress testing would need to be done to simulate the number of students that would be using the website. This would prevent the site from crashing or reacting weirdly during the times when traffic is at its peak. This would also allow us to make sure the application can scale up as much as needed based on the traffic that there will be.