

BookCub - Design Specification

Project Manager: Grace Miles (gmiles@princeton.edu)

Mohamed El-Dirany (mohamede@princeton.edu)

Alex Vogelsang (aav5@princeton.edu)

Patrick Boroughs (boroughs@princeton.edu)

Section 1: Overview

Our project is to make a Textbook Exchange web-app for Princeton students. This web app would be a platform for students to buy and sell textbooks for their classes at Princeton. We hope to offer all students a better way to buy and sell their used textbooks that is straightforward and efficient.

Section 2: Requirements & Target Audience

Currently most students use the Facebook page “Textbook Exchange” or the Facebook page “Free & For Sale” (where students currently also list textbooks). It is difficult to search on these pages, know the condition of the textbook that you are purchasing or even if it is still available. We plan to resolve these problems by presenting a user-friendly and efficient platform where users can list the textbooks that they are selling and search for textbooks by class, quality and price. We believe that there will be widespread adoption and usage as the current “Textbook Exchange” Facebook page has almost 4,000 members.

Requirements

- Users must be able to list the books that they want to sell specifying course, condition and price upon listing.
- Users must be able to search all listed textbooks and filter on class, book condition and price
- Limited to individuals with princeton netid
- Users must be able to search for a specific listed textbook
- Users must be able to sort out required texts by class
- Users must be able to contact each other to make sales transaction
- Once sold, books must be removed from listing.

Section 3: Functionality - Use Cases

Scenario: User looking to sell a book

User loads front page on desktop browser. Clicks on “Sell a Book” and enters the department and course number for the textbook’s corresponding class and textbook name. User then enters condition (fair, good, great, new), quantity and price to list book. They also will be

allowed to enter a brief note about the general condition. Submit listing and it goes live on the site. It is up to the seller to remove a book once it has been sold.

Scenario: User looking to buy a book

User loads front page on desktop browser. Clicks on “Buy a Book” and enters search for course number or department name or specific book name. The page then loads all book listed that correspond to the keyword the user types in. The user can then sort on price, condition or the time the listing was posted. Click on a specific book read the listing description and further information. If interested the user can then click “message seller” which will generate an email to the seller that the user is interested.

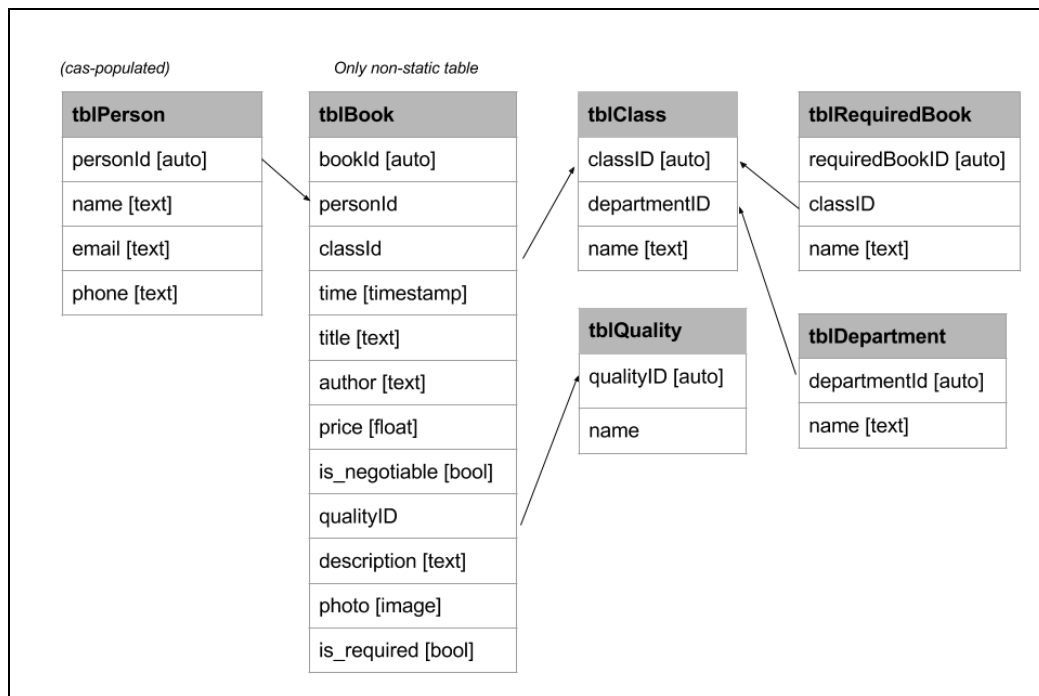
Scenario: Seller editing/deleting a listing

At any time sorting looking through the website, the user can select a ‘Manage My Listings’ button located on the top left corner, which will take them to a page containing all of his/her book listings. From here, the seller can opt to edit or delete any previously posted listing from the site.

Section 4: Design

The website will be written using Django as our web framework. Then use AWS for hosting, SQLite for database and GitHub for version control.

Database Design:



Section 5: Timeline

- March 19th - **Design Document Due**
- March 20th - Create empty database and link to Django
- March 21st - Import user profiles using CAS Princeton identification
- March 23rd - View any data on hosted website
- March 25th - Get all Textbook-Course information from blackboard and loaded into database
- March 29th - Functionality to list a textbook
- March 31st - Pre-computed drop-down lists
- April 2nd - Functionality to buy a textbook
- April 5th - Filter functionality
- April 9th - Search functionality
- April 14th - **Project prototype due**
- April 16th - Get users on app to critique and test load
- April 28th - **Alpha Test**
- May 4-5th - **Beta Tests in Class**
- May 6th - Bugfix from Beta Test
- May 14th - **FINAL PROJECT DUE**

Section 6: Risks & Outcomes

Ensuring Payment/Exchange of Information

It is going to be difficult to ensure a positive customer experience on the actual exchange of the textbook and payment. Our current idea is to put the students in contact over email, but we have no way of making sure the exchange happens. For ensuring payment: a possible solution is integrating with Venmo, but we still have to trust the pair to meet up. For ensuring exchange: we may implement a rating system to avoid this as much as possible.

Validation of Up-to-Date Information

The users may not keep their information up-to-date, especially in regard to textbooks already sold. This is a common problem and is the pain point on the Facebook page and we're unsure how we can force users to update the information that a textbook is sold. Possible solutions include expiring old book listings and sending out reminders to sellers to update their listings when a new semester begins.