

Product requirements - LBLS

(Library Book Lending System (LBLS) - conceived 1/25/21) (this will be an open source project to demonstrate a CRUD app)

Key Stakeholders

Product - Brian K. Vagnini

Design - Brian K. Vagnini

Engineering - Brian K. Vagnini

<Link to the Trello Epic>

Problem

You are in the book store and you buy a couple of books. The problem is that you already own one of those books. While you are deciding which book to keep, you realize that you are missing a book. You remember that you lent it to someone, but can't remember who borrowed it..

Solution:

The user needs to be able to add, edit, categorize, delete, and pull a report on the book inventory that they own.

Goal

The goal is to enable people to be able to keep track of their books that they have, both for inventory purposes and for lending out to someone or borrowing a book from someone.

My mission is to help people manage their lives better.

User Stories

Epics

As a User, I want to be able to create an account and sign in, so that I can manage only my task list.

As a User, I want to be able to manage my library (adding editing, deleting, categorizing)..

Typical User:

Brian is a very busy person, who has so many books that it starts to rival small libraries. He needs a way to inventory his collection, organize them into categories, and keep track of which books he's borrowed (and from whom), as well as which books he's lent out to others.

	As a	I would like	So that I could	Phase	Notes
1	User	To scan in/type in an ISBN number	Add an item to the library	1	
2	User	To scan multiple ISBN numbers into a text file, then import that list	Add items in bulk to the library	1	
3	User	To be reminded once a (week Month) to update my library inventory	Ensure that it's accurate as I purchase new books.	2	This will be an optional feature that the user will have to opt in for.
4	User	To add a single item to the library via it's title	Add an item to the library	1	
5	User	To view my inventory via a web page	Ensure the item in my hand is NOT something I already have in my library	1	
6	Admin			3	
7	User	To be able to have a similar experience with the app	Use the app wherever I am to check if I already own this book	2	

		on my cell phone			
8	Developer	To be able to log the data flow / usage of the app	Troubleshoot issues with the app	3	Log details should include: username, data change made, action taken, time stamp, etc.
9	Developer	To secure the accessing of the users data	So that their private info remains private	2	Use of an API to deal with data may help in this matter
10	User	To be able to search inventory by title	Verify that I do or don't already own that book in question		
11	User	To be able to search inventory by Author	See what else I have by that author		
12	User	To view inventory detail via a web page	See the details on a particular book/series		See Inventory Details (below) for some of the details I want to see
13	User	To lend a book to someone	Keep track of who I lent a book to, to ensure I get my books back..		
14	User	To borrow a book from someone	Keep track of who I borrowed a book from, to		

			ensure it gets back to the right person.		
15	User	To receive a notification to return a borrowed book	Ensure that they get their book back in a timely fashion	2	
16	User	To snooze/turn off a return book notification for a definable period of time	Remind me to start/finish reading that book, or to simply return it	2	

Inventory Details:

Title, author, number of copies, ISBN, Edition (1st, 2nd, 3rd, etc.), year of publication, publisher, status (loaned out, to whom, when loaned, currently reading, borrowed, borrowed from, date borrowed), category, subcategory, rating, condition, series, number in series, height, weight, cost, value, insured

Metrics

Keep a weekly, and monthly tally of new users that sign up for the service. Keep a list of active users, by week (query on date last signed in < 7), and compare that list to the list of users to determine number of ACTIVE users

Design

There will be the ability to add new inventory one item at a time, by ISBN number. You can also do a bulk insert with a text file with ISBN numbers. Whether it's one or many, there will be a function that takes that ISBN number (10 digit or 13 digit), then reaches out to the Internet via an API to retrieve all the pertinent information about that book (author, title, subtitle, MSRP, genre). It then inserts those fields into the database. There will be the ability to pull a report of your inventory, or search for a title (useful at the book store). The entering of the ISBN for data entry purposes, will accommodate either typing it in or using a barcode scanner.

<INSERT SKETCH HERE of the UI>

Launch Plan

Eligibility: Open to waiting list of users

Test: See the Design

Control:

Ramp plan: 15% initially, then in 20% increments (this allows us to scale services appropriately).

FAQ/Press release

(2-4 pages)

Do you have a ton of books? Do you occasionally buy multiple copies of the same book? Or forget who you lent a book to? Are you about three books away from being declared a library?

Q: Why can't I just use an Excel sheet?

A: You don't always have a computer with you. By using a web based application, you can log in while at the bookstore.

Q: How do I turn off these damn notifications to use the app? Why did you enable that by default?

A: The feature is enabled by default so that as you acquire new books, you will be (gently) reminded to update your library; otherwise, it gets out of date. To turn off the notification, go to <insert exact path here>.

Wireframes and User Journeys (optional)

add a helpful visual outline of what the user journey looks like when the product or feature is complete.

These are not detailed wireframes/sketches for developers

Wireframes are the layouts of the screen that give the general outline of elements.

Release Criteria

Here we describe goals (not how we achieve them). You'll want to outline criteria in these five areas:

Functionality: Is there any baseline functionality criteria to retain at every release? Define what is absolutely mandatory and what could be discounted.

Usability: Is the product meets the minimum requirements of User-friendliness? What is the acceptable time to complete core tasks?

Reliability: What's the maximum acceptable failure rate? Are these failures predictable? Can the system recover from these failures?

Performance: What is the maximum response time, throughput, and memory consumption?

Supportability: Is it serviceable, maintainable, and configurable?

Risks

The risk is losing the customer's data due to a hardware failure, or accidental deletion of data. This can be mitigated by hosting it on a DBaaS, to include replication and DB backups.

Timelines

An overview of release targets and when each feature is expected to finish (even if it's a rough estimate). It is a relatively brief overview of development targets

A more detailed release plan can be defined in the product roadmap.

Non-functional requirement

While the functional requirement that is the bulk of your product requirements document defines how the software functions, the non-functional requirement defines requirements that may be important to your business.

Communicate any particular parameters that the developers/architect should take into account.

The Technology stack will be Python and Django, using PostgreSQL as the database.

App development process

1. Design Database Fields
2. Design API methods (Inputs & Outputs)
3. Write up API documentation
4. Design API upgrade strategy & naming scheme
5. Design Front End website app to interact with API
6. Design React Native App / IOS App to interact with API
7. Design reporting metrics for user counts, usage, last login, revenue generated etc.