

# Book Swap App

---

Noa Lahan

App Development

Kenneth Lim

University of the Arts London -  
Creative Computing Institute

11 June 2024

# Contents

---

<b>Introduction</b>	3
- Application Abstract	4
<b>Research</b>	5
- Community	6
- Need for the App	7
- Similar Case Studies	8
<b>App Prototypes</b>	10
- App Hierarchy	11
- Design Features	12
- Design Evolution	13
<b>Code Implementation</b>	14
<b>Areas of Improvement</b>	18
- Concept Improvements	19
- App Improvements	20
<b>Project Results</b>	21
- End Product	22
- Links	22
<b>Bibliography</b>	23



# Introduction



## Application Abstract



# Introduction

I have aptly named my app “Book Swap.” The idea of it is to create a more efficient and automated way to swap books with people in one’s local reading community. The app will be useful in fostering a sense of belonging, saving both money to users and paper overall with users sharing their books instead of only using them a few times, and in encouraging more people to read more as there is a lower entry point to start out as a reader when using the app.

The main app functionality will work by having each user publish their wish list of books and a list of books they own (and do not mind giving away), which will be visible to other users. Other people in the area can look at those lists and message a person to inquire about a book swap they would be interested in.



# Research



Community  
Need for the App  
Similar Case Studies



# Community

---

According to Cambridge Dictionary, a community is a group of people “who are considered as a unit because of their common interests, social group, or nationality” (Cambridge Dictionary, 2024). Meaning, every person is a part of many communities both generic and specific. For example, I am a part of the UC Santa Cruz’s community, CCI’s community, artist community, etc.

One community I belong to that I am fond of is the reading community, especially because of its rapid decline. The National Literacy Trust recorded the lowest reading enjoyment level in 2023 since they started collecting data back in 2005 (National Literacy Trust, 2023). Reading has many health, emotional, and social benefits (BBC Teach, 2019), and yet less and less people read for a few reasons: many say it costs money or they do not know what to read, but the main reason mentioned across various sources is the fact that people are using their phones more and read less (Cronk, 2019; Common Sense Ethics, 2024).

# Why the App is Necessary

---

Book Swap combats the previously mentioned reasons that people are reading less, and as such can be a way to help strengthen and expand the reading community.

Reading costs: The app would remove any monetary issue by fostering a free cycle of reusing pre-loved books. While that does insinuate a need for a book to swap with, that is not necessarily the case as the app just connects you with other book owners who are willing to share some of their collection. Some may be open to lending books with no book in return.

I do not know what to read: The app grants its users access to others' personal wishlists. Get inspiration from those, or message another user for direct recommendations on books they have enjoyed and think you may as well.

Books are competing with phones for attention: Rather than working against it, Book Swap works with the public's likeness to phones, by encouraging creating and maintaining connections with the reading community via phone, meaning not having to change one's habits much or, as previously mentioned, giving reading a more accessible entry point.

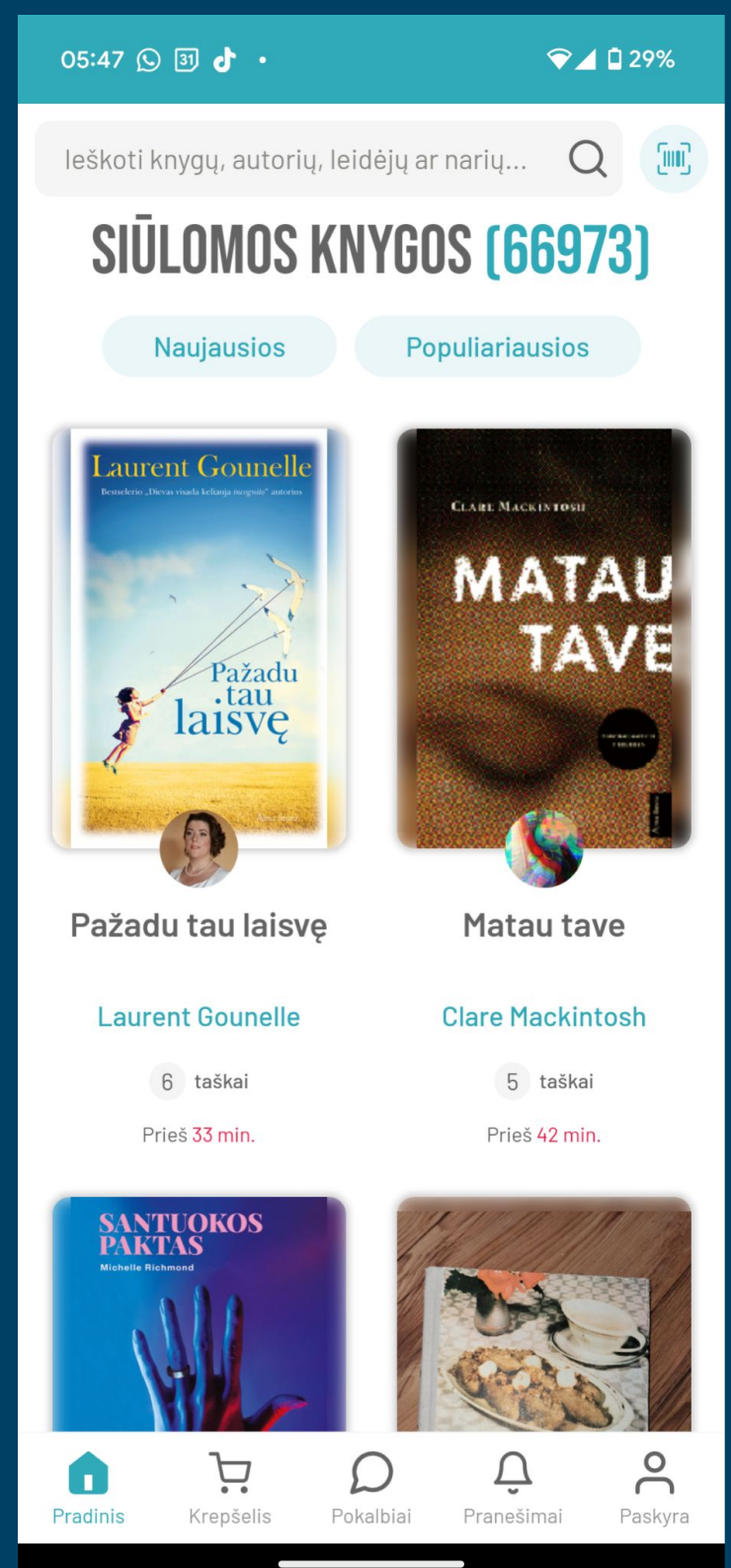
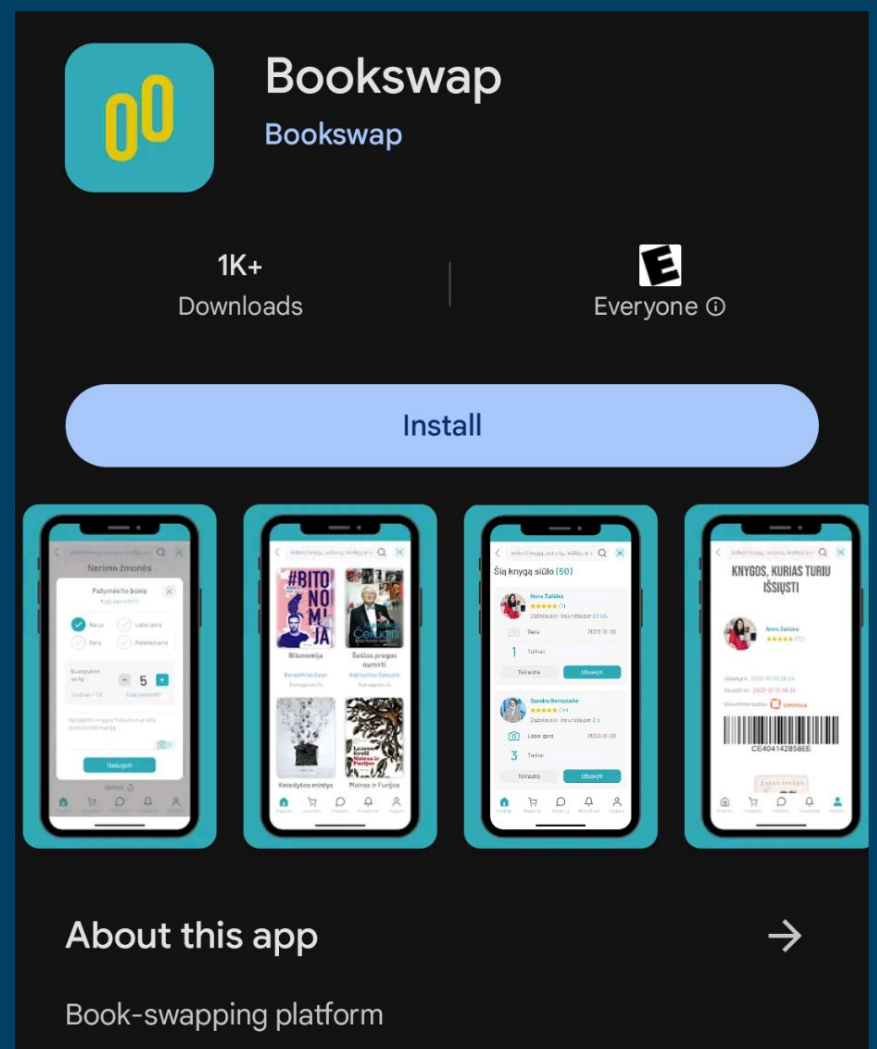


# Bookswap

## Case Study 1

Bookswap is an app designed for book swapping. It depends on its users signing into their Facebook accounts as soon as they open the app and many functions are unavailable until you do so. The app is in Lithuanian and so I could only gather minimal data, but it has multiple tabs who's purposes seem unnecessary.

Takeaways: Focus on connection to readers rather than the books. Only have necessary tabs. Do not bring users straight to profile.



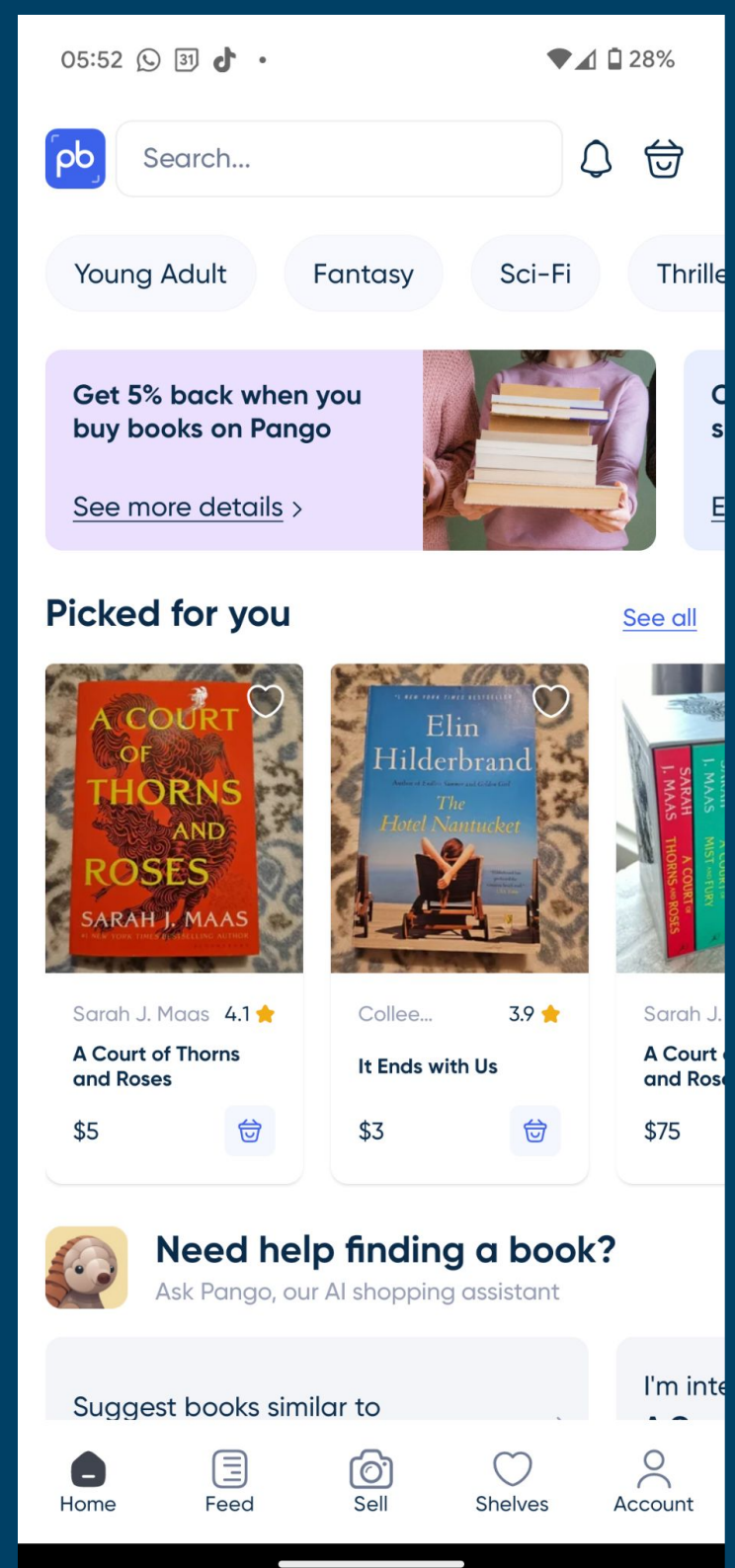
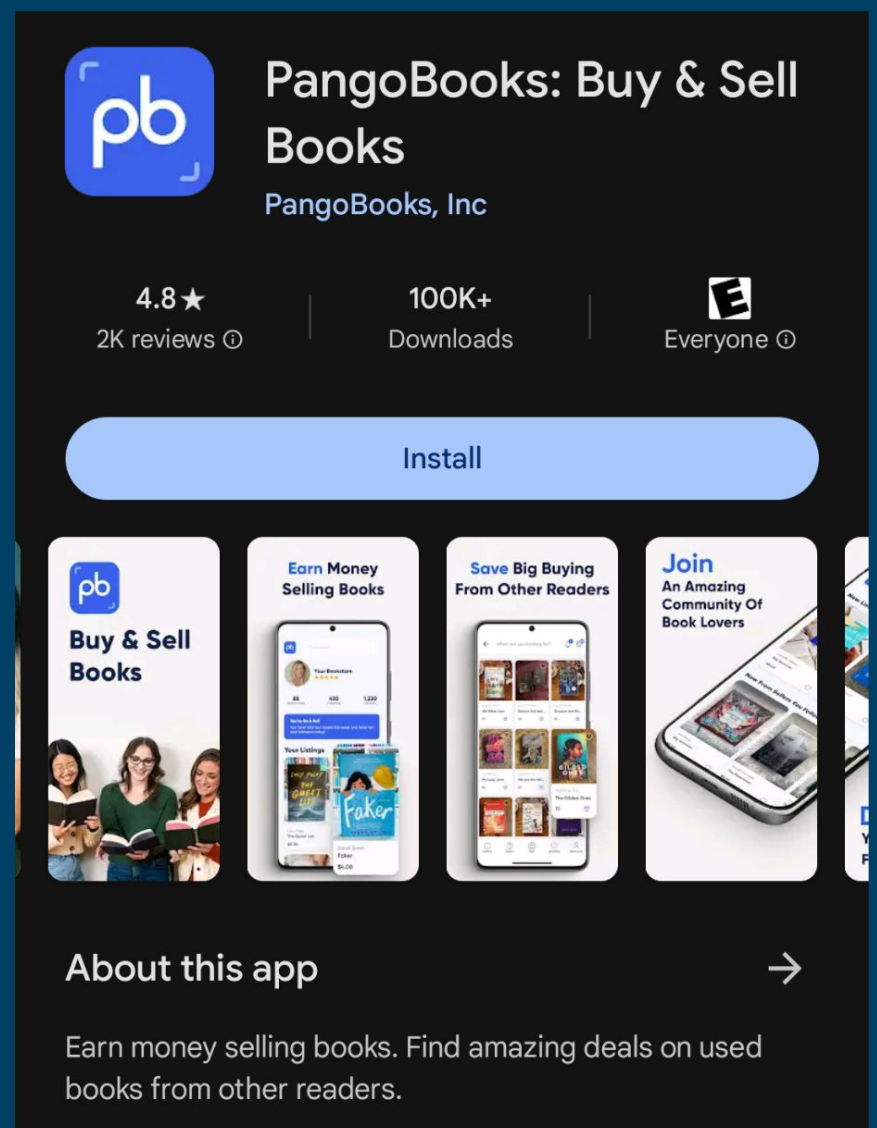


# PangoBooks

## Case Study 2

Pango is a book market app. The idea is to post your book with your rating of it and the asking price, and others will be able to buy it. The Home page had a lot of information at once but was still understandable with the consistent formatting. Shelves tab saved any purchases, books that you are selling, and books that are on the wishlist (added by “heart-ing”)

Takeaways: Have consistent formatting for a given element. Add search filters. Save user’s books in its own tab.





# App Prototypes and Design Development



App Hierarchy  
Design Features  
Design Evolution

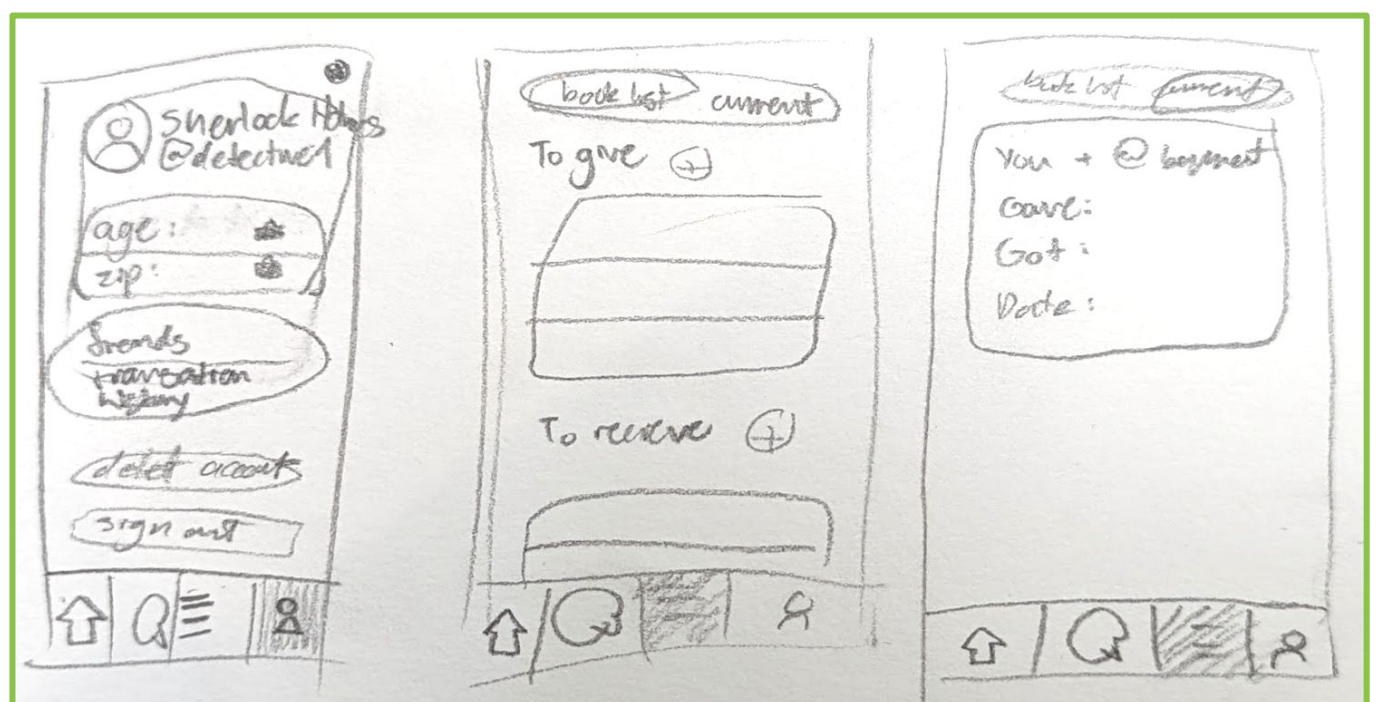


# Hierarchy

I started the app design process by writing out the app's page hierarchy. I knew I wanted 4 tabs.

1. Main page: First thing user sees. From both case studies I know this needs to have the books.
2. Chat: Since fostering community is the focus of this app, having communication is essential.
3. Library: Inspired by PangoBooks' shelves tab, this will keep record of the users book lists and past swaps.
4. Profile: always seems to be the farthest page from the landing one (see case studies, Instagram, Netflix, etc.)

From there I could start sketching layout ideas



Low fidelity sketch of profile and library (then list) pages

- Profile
  - Main
    - Name
    - Username
    - Age (priv)
    - Zip (priv)
  - Relations
    - Friends
    - Transaction history
  - Info
    - Edit profile
    - Notification settings
    - Privacy policy (i)
  - Delete account
  - Sign out
- List
  - Library
    - Add button
    - To give
    - To receive
  - Transactions
    - You + @....
    - Inner
      - Gave: book
      - Got: book
      - Date:
      - Message @...
      - Complete
- Messages
  - Name
  - Picture
  - Date
  - Last message
  - Start new chat (from friends)
- Main
  - Name
  - distance
  - Picture
  - Want
  - Has
  - Inner
    - Name
    - Username

Hierarchy list



# Logo, Name, Color scheme

To start building the idea of a product I began contemplating the app's name and designing its corresponding logo. For the logo

SWAP

Stories worth another p??

Sharing words, acquiring pages

Supporting wonderful a

I knew I wanted it to have an arrow elements (like a swap sign or a recycling sign) and a book element. I initially wanted the app to be called SWAP and be an acronym, but I was unable to come up with any good ones and the idea was scrapped for a more straightforward choice.



Figure 1.



Figure 2.

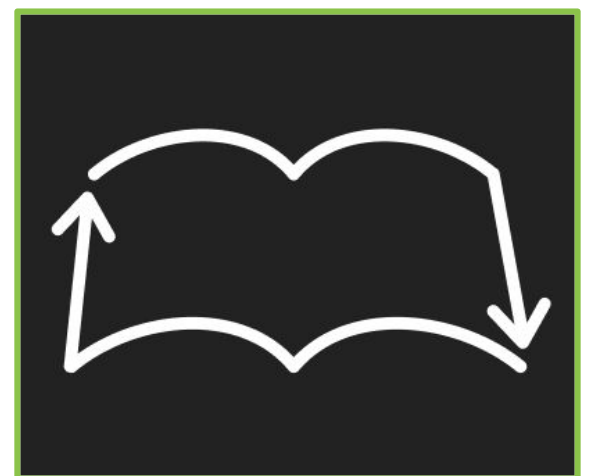


Figure 3.

Figure 1 was scrapped both because of the name but mainly because of the complexity of the logo, it looked too messy and was hard to design in a clean manner. Figure 2 was a personal favorite but after asking a few people for an external opinion it was deemed a little confusing and the arrow elements were not arranged in the “recycle manner” which brings us to the final result of Figure 3. Clean, clear, arrows, book.

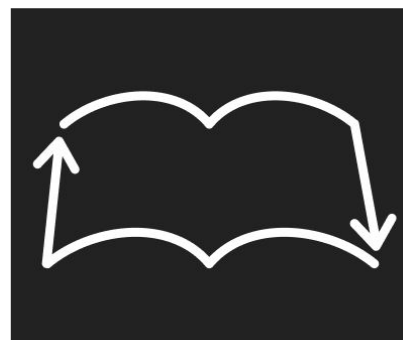
Staying on theme, the color scheme was based on old book colors, alternating leather-bound-book red and stained-pages beige.



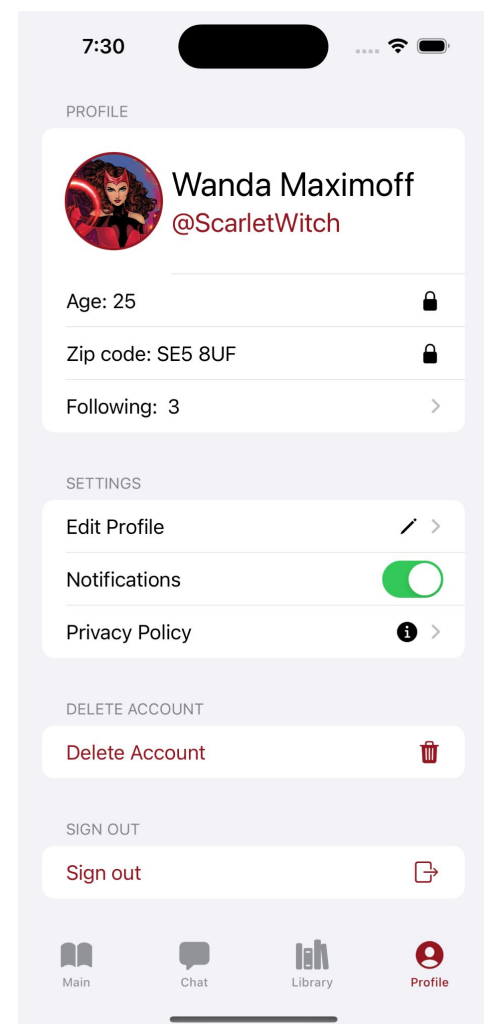
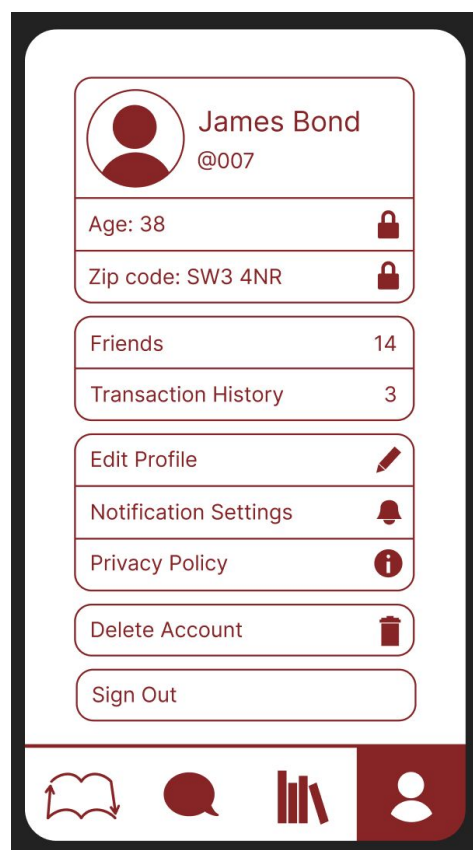
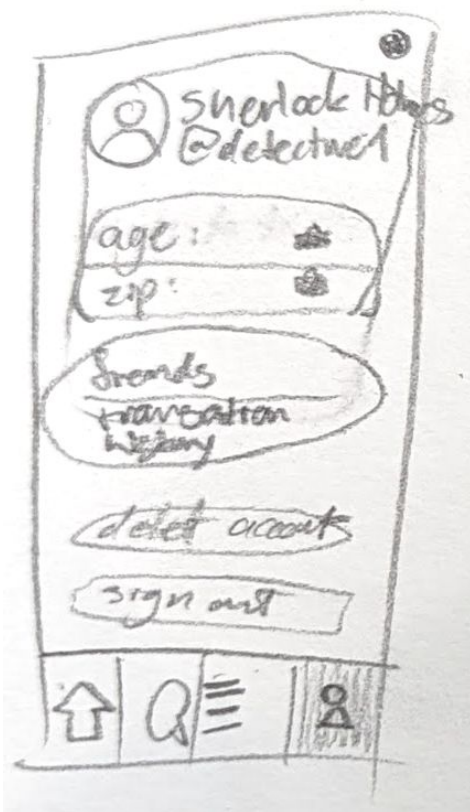
In the app itself only the red lasted as the accent color while the beige was replaced for plain old white.

# Design Evolution

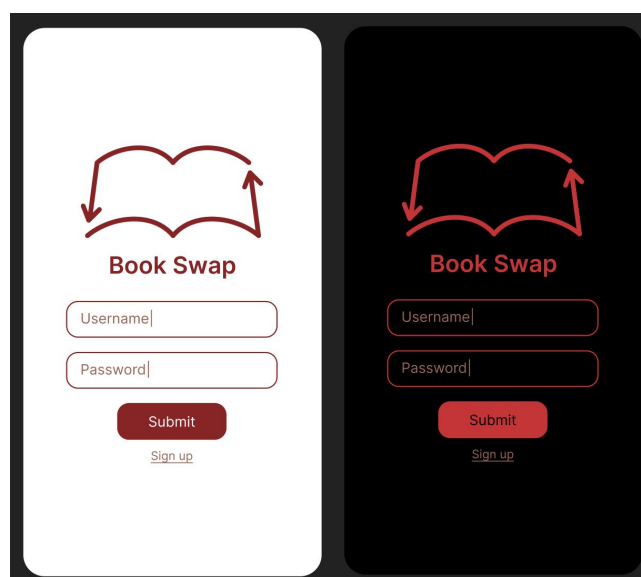
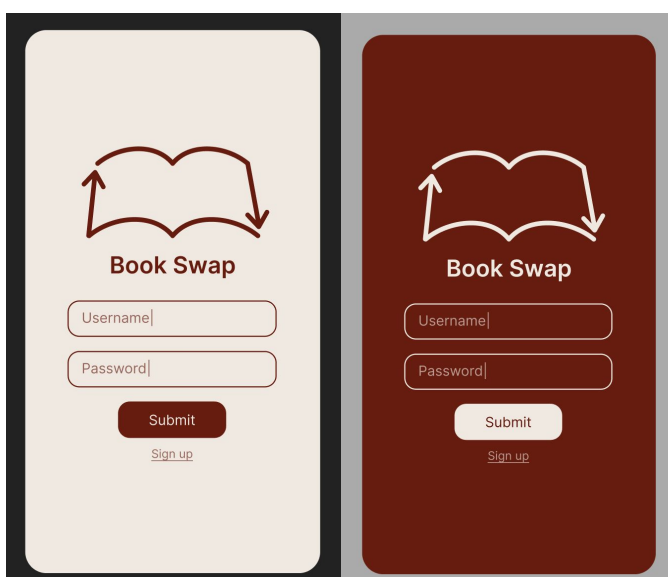
## Logo Evolution



## Profile Evolution



## Color Scheme Evolution




This decision was made for comfort. It was easier to implement figure 2 than 1



# Code Implementation



All the different files, what  
they do, and how they  
relate



# Code Files

Page files: Main, Chat, Library, and Profile are the main page files, each corresponding to a different tab (see figure 1) which are created and handled by ContentView.

Main: shows all the users sorted by distance. The app user can look up a particular book to filter the user list to only those interested in that book. The selector filters at the top allow for a more precise search.

User: Contains a user class with all the needed fields and the users database with some pre existing users (seen in figure 1).

UserDetail: takes a user instance as an input. Used to make the user profile.

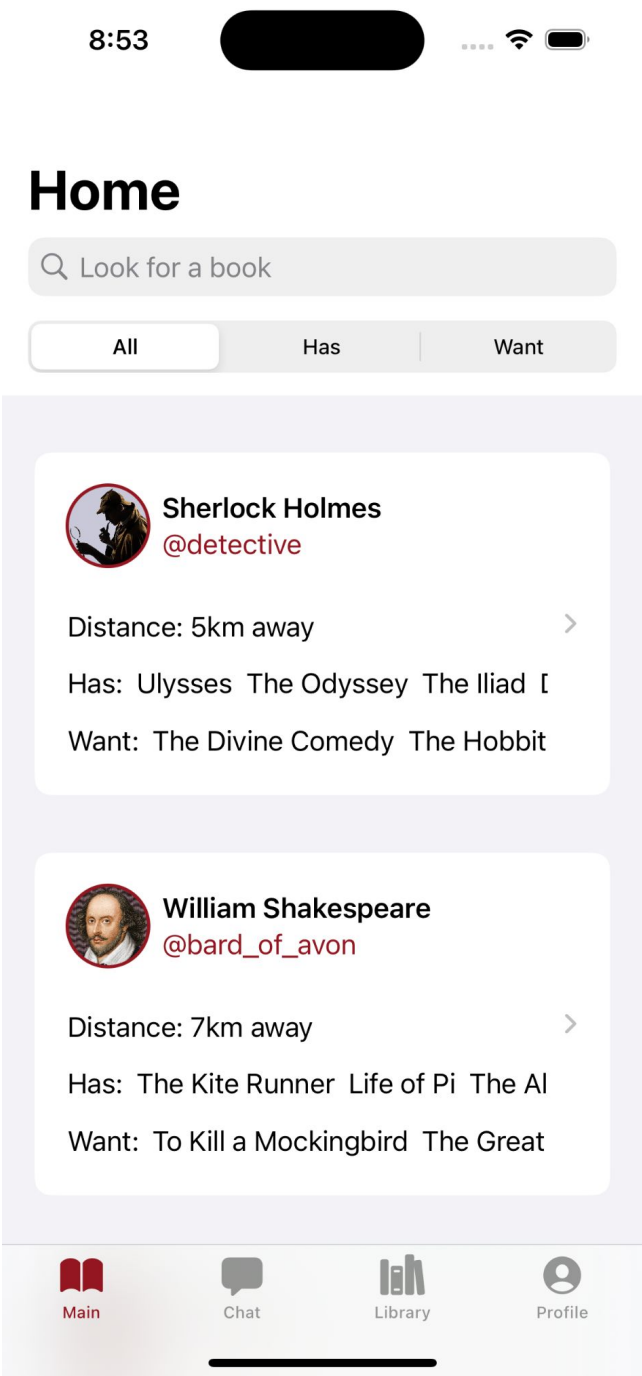
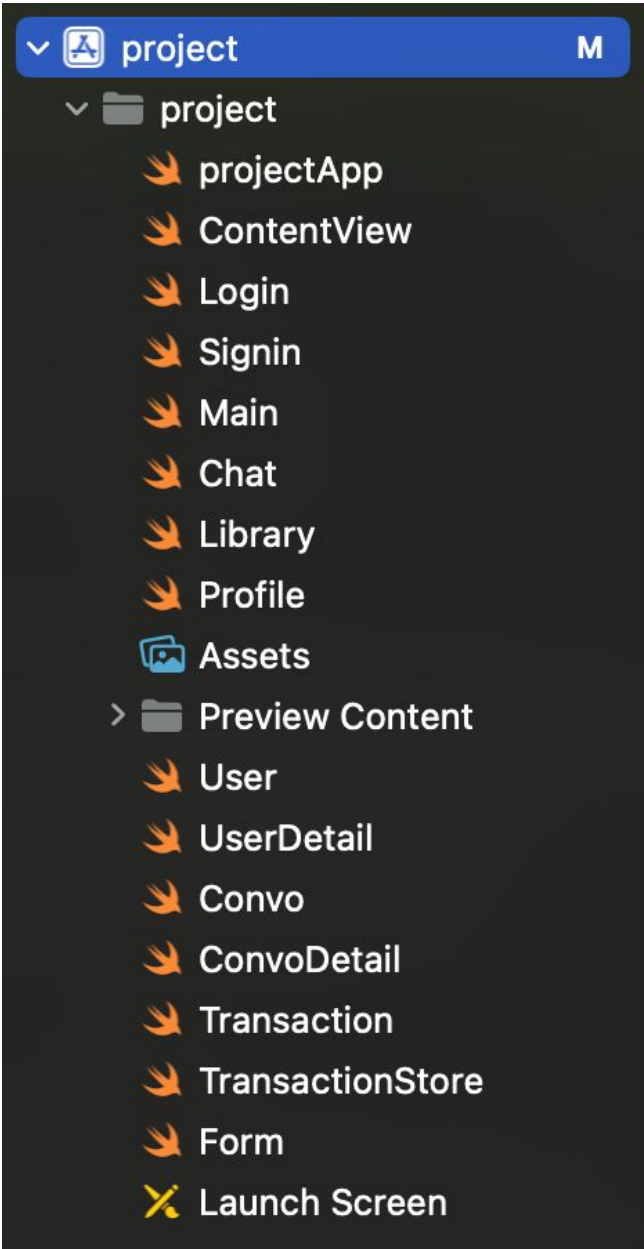


Figure 1

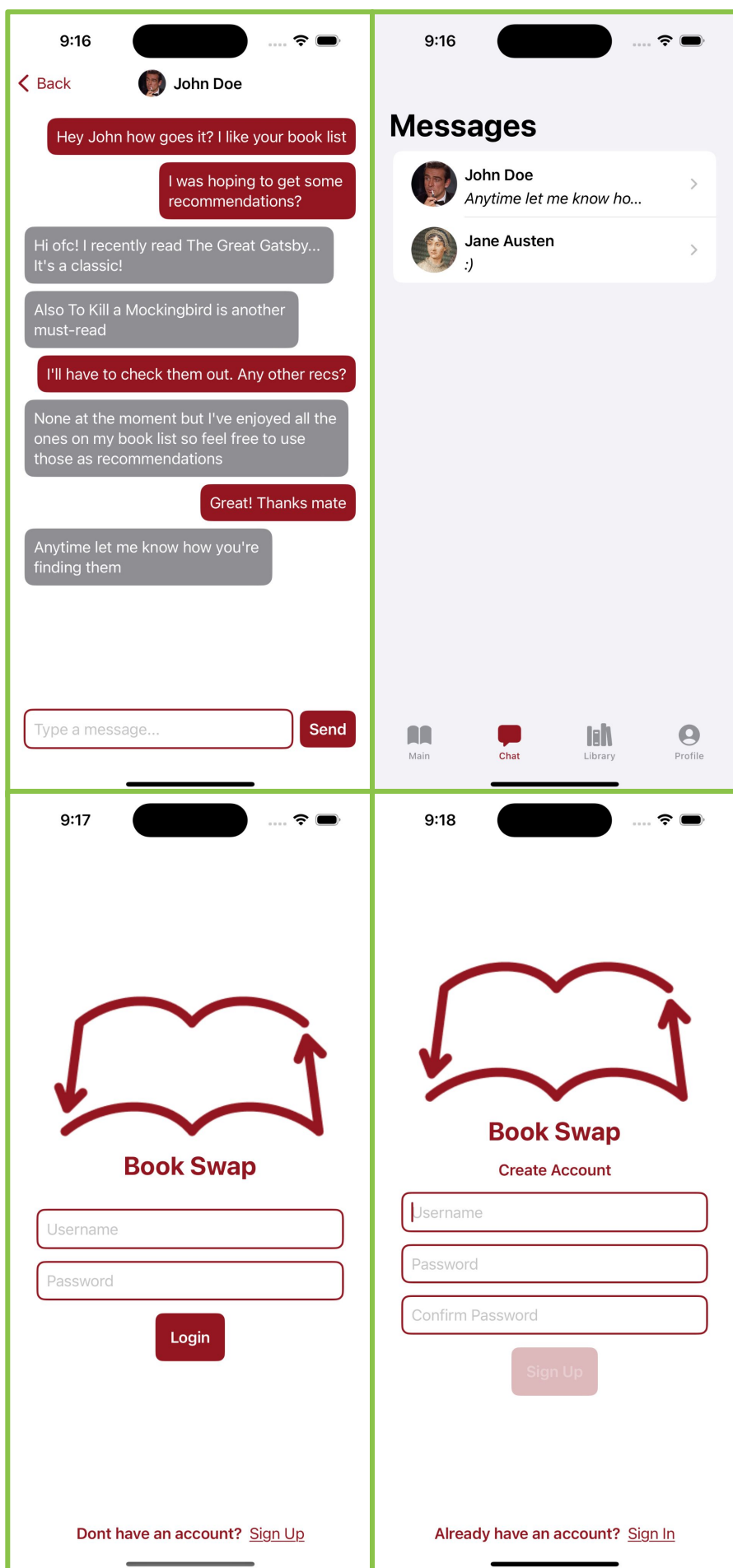
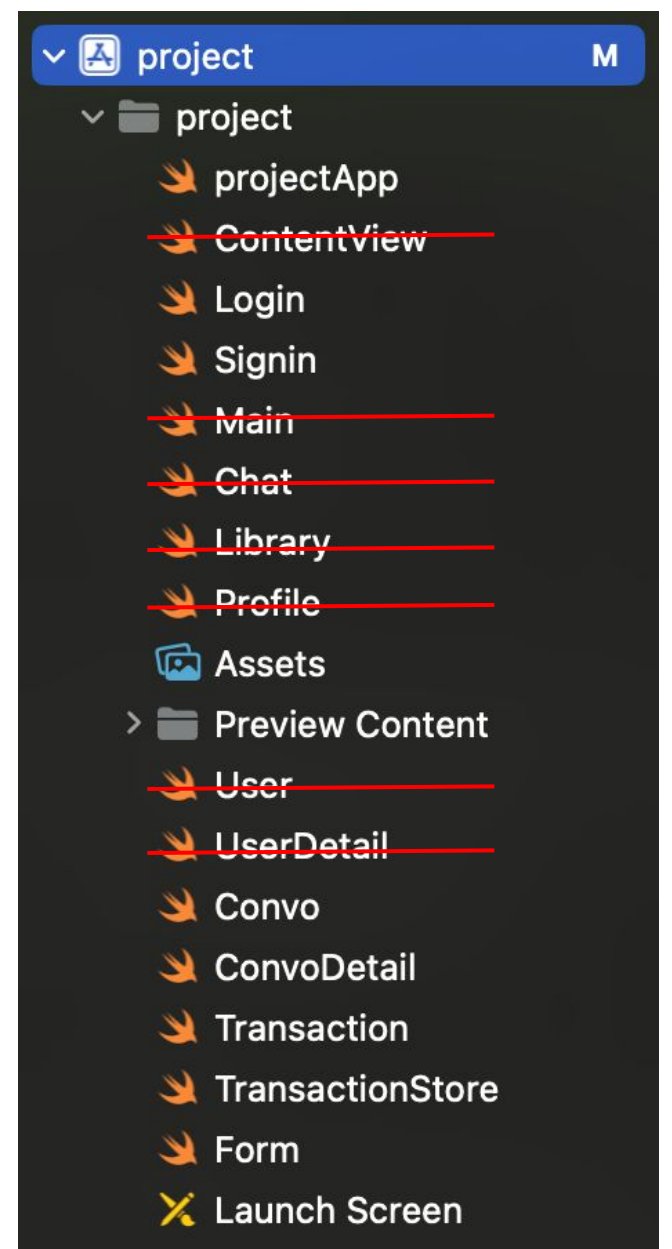


User profile before and after styling  
(inspired by instagram profiles)



# Code Files cont.

Chat, Convo, and ConvoDetail files work together as well. Chat creates the list of navigable conversations seen straight from the tab. Convo creates a Convo class element, a Message object (used in convo to know which message belongs to which side of the conversation), and a convos list database, similar to that of users.



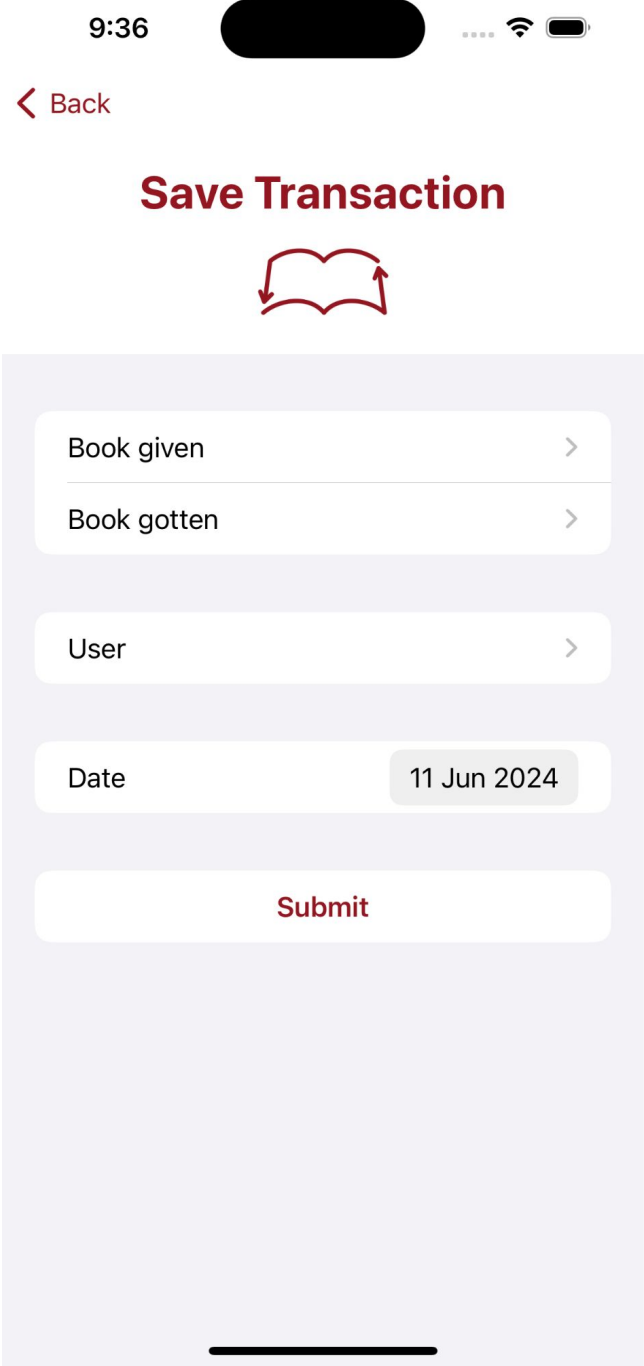
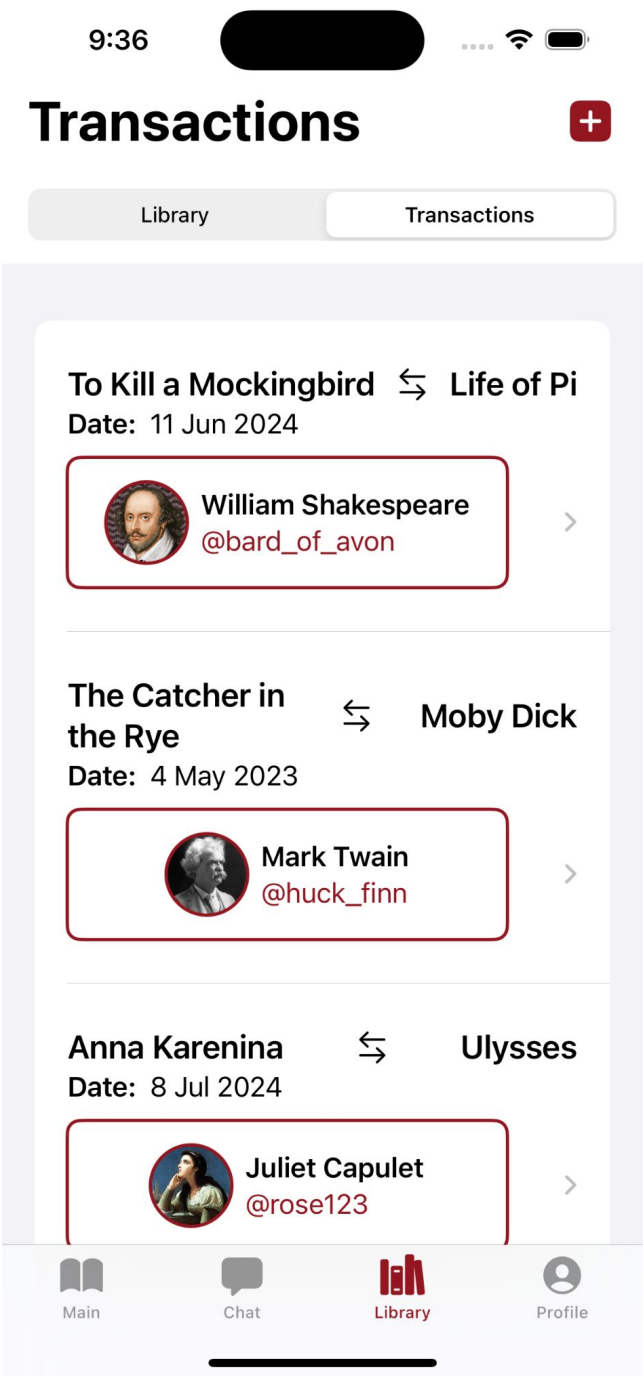
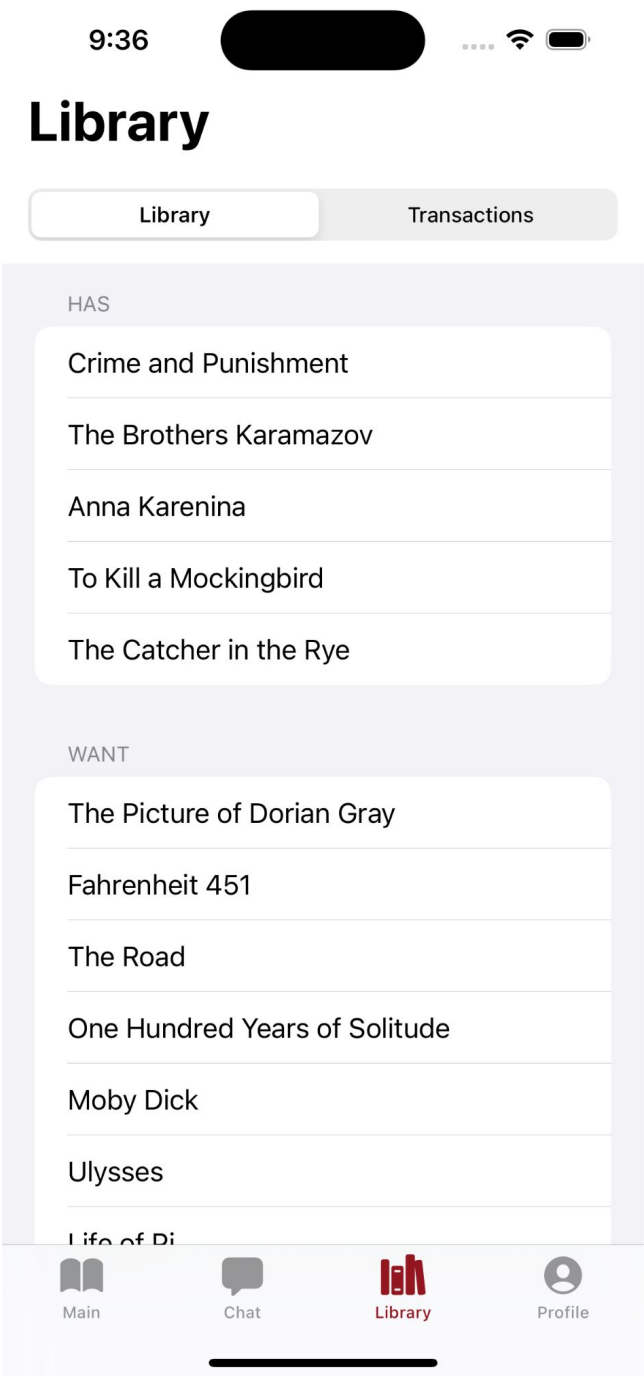
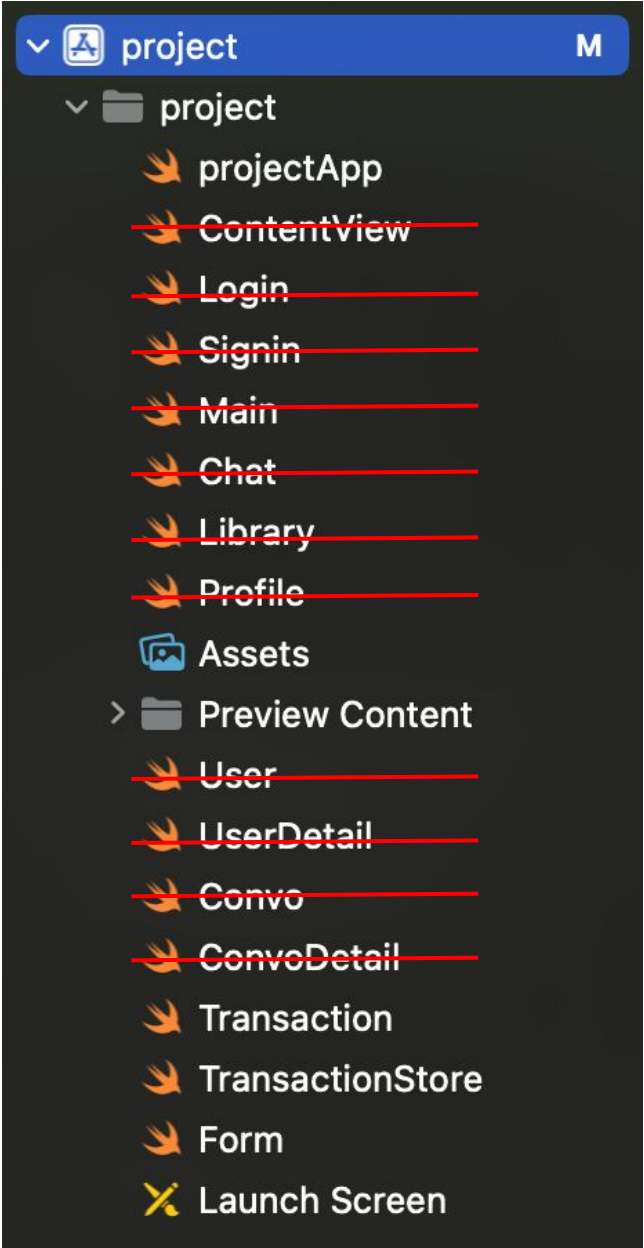
The Login file is the first the project opens (bottom left corner) and asks for a username and password to match the correct set. If it does, the login button will navigate to the tabs (ContentView) else it generates an alert. The same thing happens with Signin (bottom right) but instead of requiring the correct set it just asks for a valid one (matching passwords and a username). You can navigate between the two through the bottom link

# Code Files cont.

The most detailed hierarchy belongs to Library. It employs Transaction, TransactionStore, and Form in making it work properly. The page is split into 2 (library and transactions) and the transactions page needs Transaction for the Transaction class, TransactionStore for handling any changes to transaction database, and Form to create new transactions.

Lastly, Profile doesn't navigate to any external files and handles most of its necessities in itself.

Otherwise, projectApp runs project (linked to Login file; Launch Screen launches as a loading page until the app is ready.





# Areas of Improvement

---

Concept Improvements  
App Improvements



# Concept Improvements

---

The main issue with the app concept itself is that it heavily relies on the users in multiple ways.

Firstly, and the reason that there are not more examples of this concept out in the market, is that it requires a fair amount of active users to be useful. Without other users to exchange books, there is no reason for the app, which makes it difficult to take off.

Secondly, it relies on user honesty and safety. The “age” feature attempts to ensure a bit more safety by only allowing users to interact with users in their age groups (not actually implemented just shown that the information is saved in profile). Which makes the app unable to take responsibility for any issues in terms of bad people, unreturned books, etc. A way to possibly improve this would be make this an app for an external “library box” product, which allows users to leave and take books from a given box, like Amazon Lockers (Amazon.co.uk, 2024)



# App Improvements

---

There are a few ideas I have to improve the app that I was unable to implement in the name of not overcomplicating the app.

In the library, I originally tried to implement transaction requests, similar to follow requests in other social apps, where you can go to a user's profile, click on a book you either want to get from them or one you want to supply them with, and it will create a transaction request that you can fill out and send to the user. On their end they will be able to accept or deny it. The implementation was unsuccessful so I have made simpler transaction objects that are more personal notes than a usable feature.

Another feature I started implementing but ran out of time on was more complex book objects. It would be nice to know what a user thought about a book they have available before you borrow it from them, have pictures and not just text, add the author, and maybe even a star rating. This would have been doable but since I wanted to focus on a more complete app interface it was not prioritized but I do think it could improve the overall user experience.

Lastly, a nice feature that is beyond my current knowledge would be some sort of a recommendation system. Some code that will organize the user list order based on most fitting person for you, whether that is based on similar books you own, you both have books the other person wants, both in the same area, both enjoy the same genre, etc. For the purposes of this project, I sorted the user list by distance but a proper recommendation system would elevate the app significantly



# Project Results



End product  
Links



# Results

The final project is a functioning app intended to swap books and interact with other members of the reading community in order to encourage reading even in this increasingly digital world.

Other users were hardcoded into the project to get the full feel of how the app would operate if it was fully functional.

## Links

Github repository:

<https://git.arts.ac.uk/23043904/appdev/tree/main/project>

- Repo also includes this file and a video walkthrough of the app



# Bibliography

Amazon.co.uk. (2024). Amazon Locker - Amazon Customer Service. [online] Available at: <https://www.amazon.co.uk/gp/help/customer/display.html?nodeId=GXCWH4CXLKJD8Z52> [Accessed 11 Jun. 2024].

BBC Teach. (2019). Why is reading good for me? [online] Available at: <https://www.bbc.co.uk/teach/articles/zmbtwty> [Accessed 11 Jun. 2024].

Cambridge Dictionary (2024). community. [online] @CambridgeWords. Available at: <https://dictionary.cambridge.org/dictionary/english/community> [Accessed 11 Jun. 2024].

Common Sense Ethics. (2024). 5 Reasons People Don't Read Anymore, And 5 Good Solutions. [online] Available at: <https://www.commonseethics.com/blog/5-reasons-people-dont-read-anymore-and-5-good-solutions> [Accessed 11 Jun. 2024].

Cronk, M. (2019). 5 Reasons Why Adults Don't Read Books - Sartorial Geek. [online] Sartorial Geek. Available at: <https://sartorialgeek.com/5-reasons-why-adults-dont-read-books/> [Accessed 11 Jun. 2024].

National Literacy Trust. (2023). Children and young people's reading in 2023. [online] Available at: <https://literacytrust.org.uk/research-services/research-reports/children-and-young-peoples-reading-in-2023/> [Accessed 11 Jun. 2024].