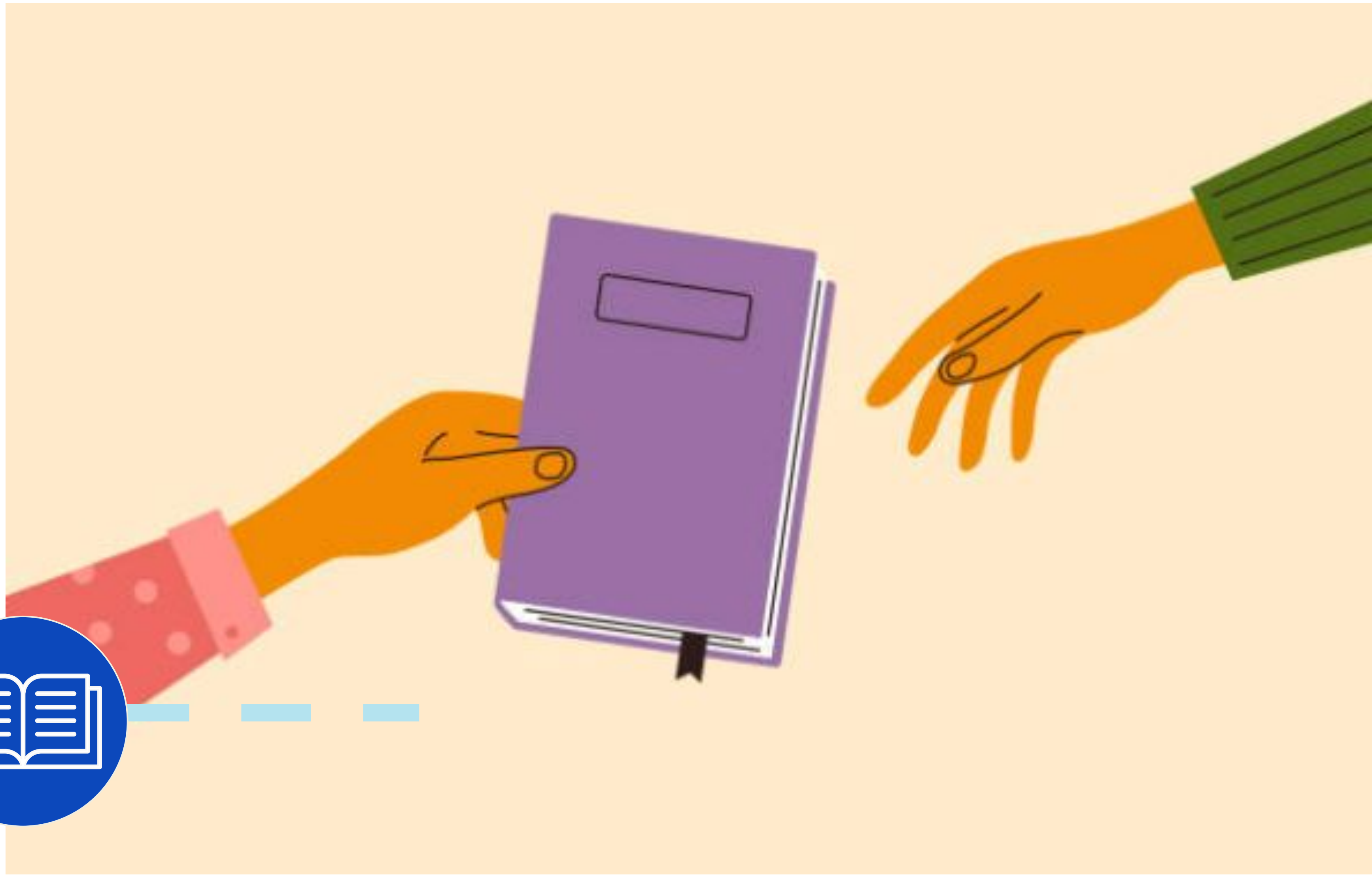
Book Cycle

Team CSJ

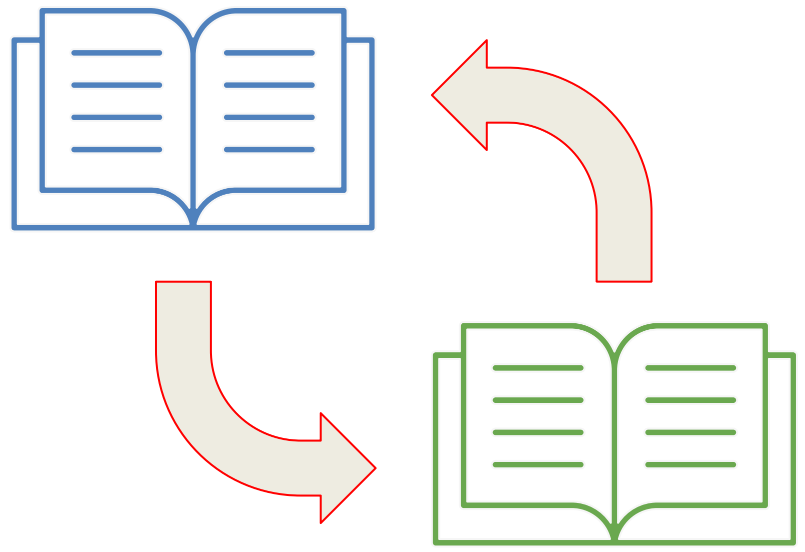


What are we?

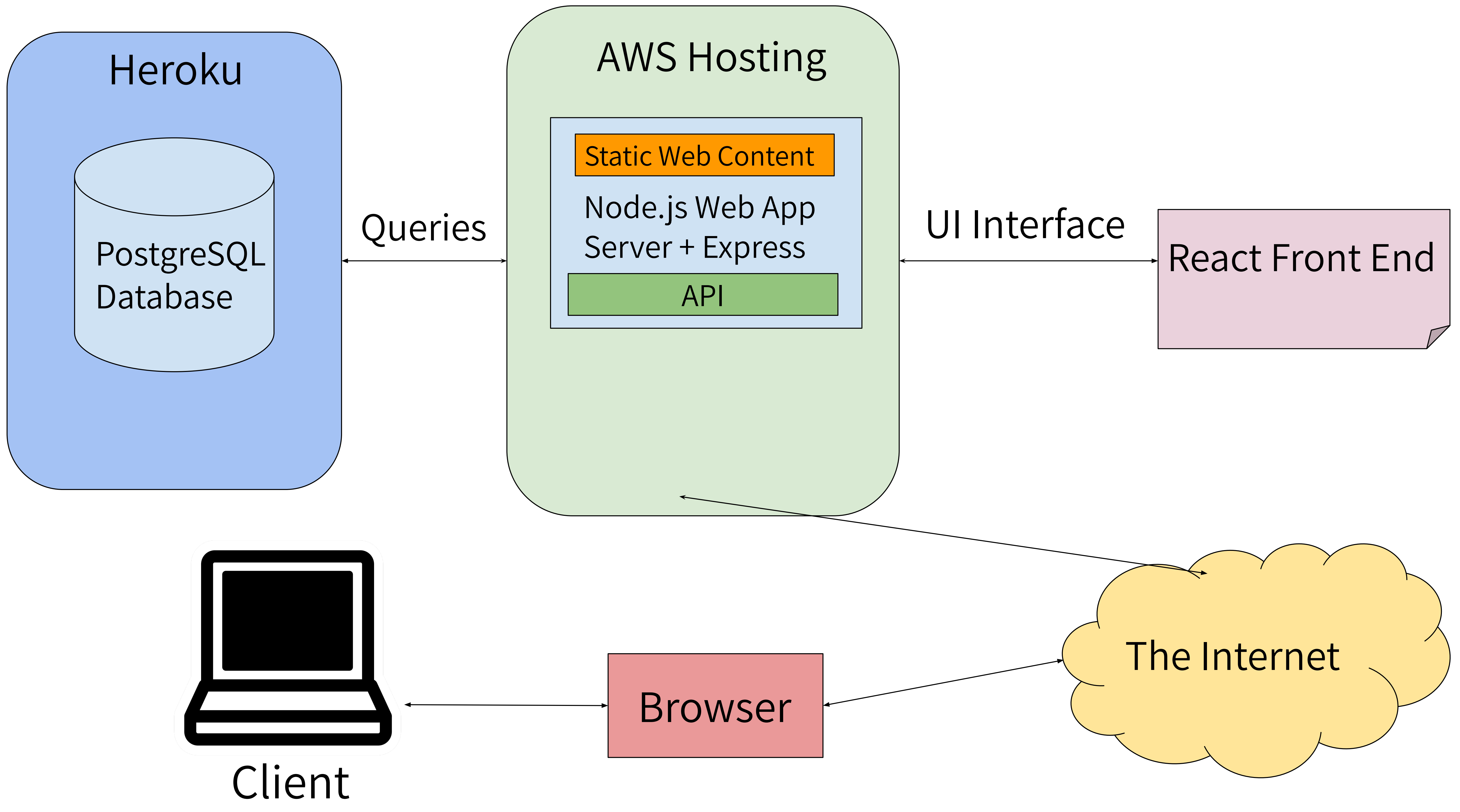
- Web app that allows students to donate and swap textbooks
  - Within a trusted zone and community on campus
- Reducing the financial burden of buying textbooks
  - Promote reuse

System Goals

1. User sign-in with verified GW email
2. User donates a textbook
3. User receives a list of matched textbooks upon request
4. User fills out a form detailing time/place of exchange
5. User successfully picks up a textbook

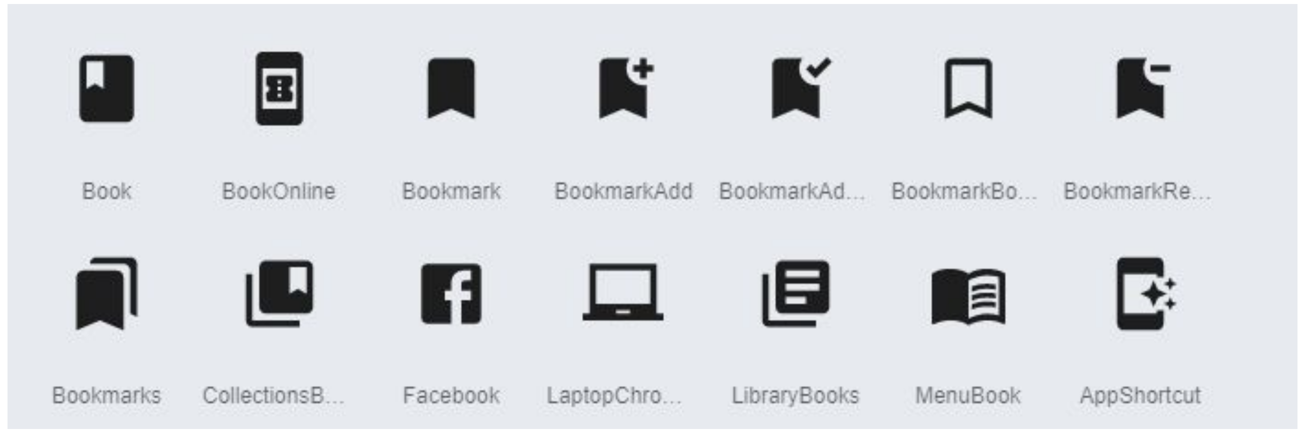


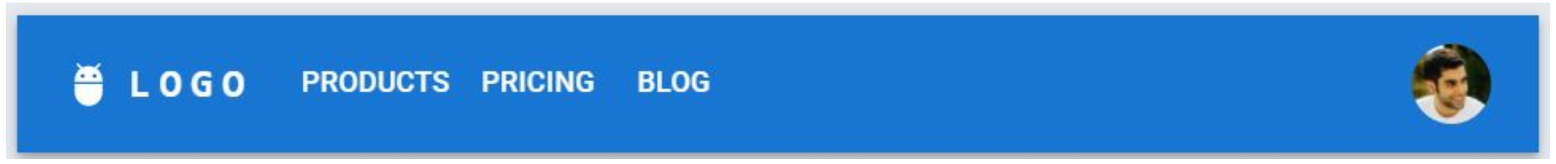
System Architecture Diagram

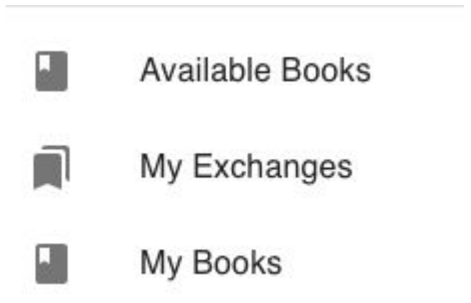


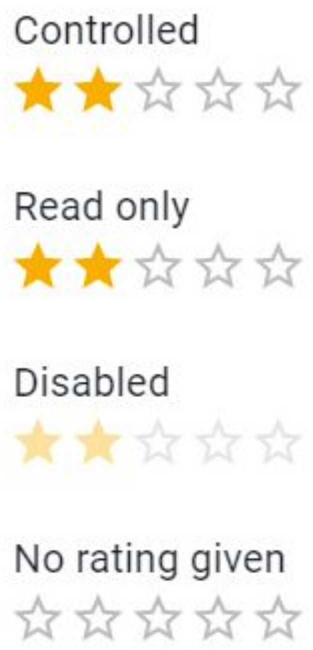
UX/UI Design and Front-end

- React - Frontend JavaScript framework
- Axios - JavaScript library to make HTTP requests to the API
- Material UI - framework for React







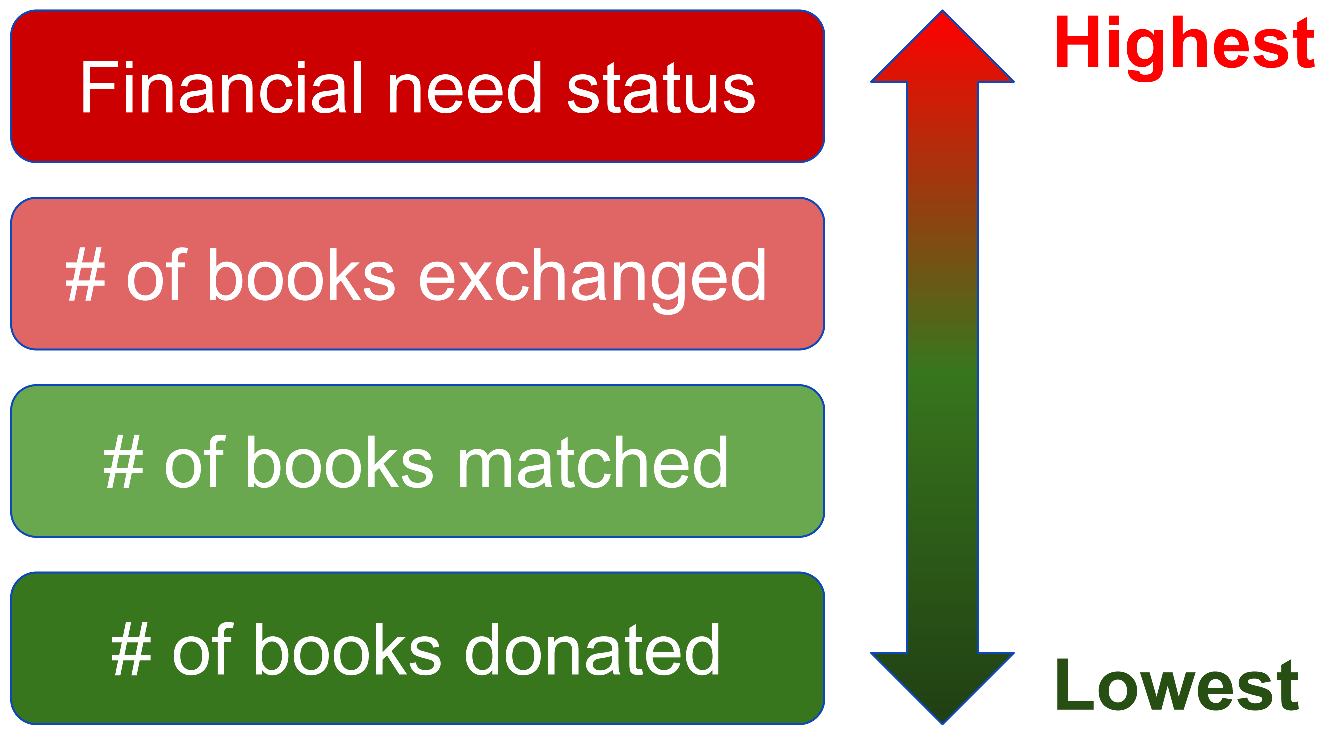


Matching Algorithm

- Goal:
  - Optimize so that students who are in financial need can get textbooks
- Considerations:
  - Prices of books
  - Financial need
  - Book:
    - Matches
    - Donations
    - Exchanges

Priority Ranking

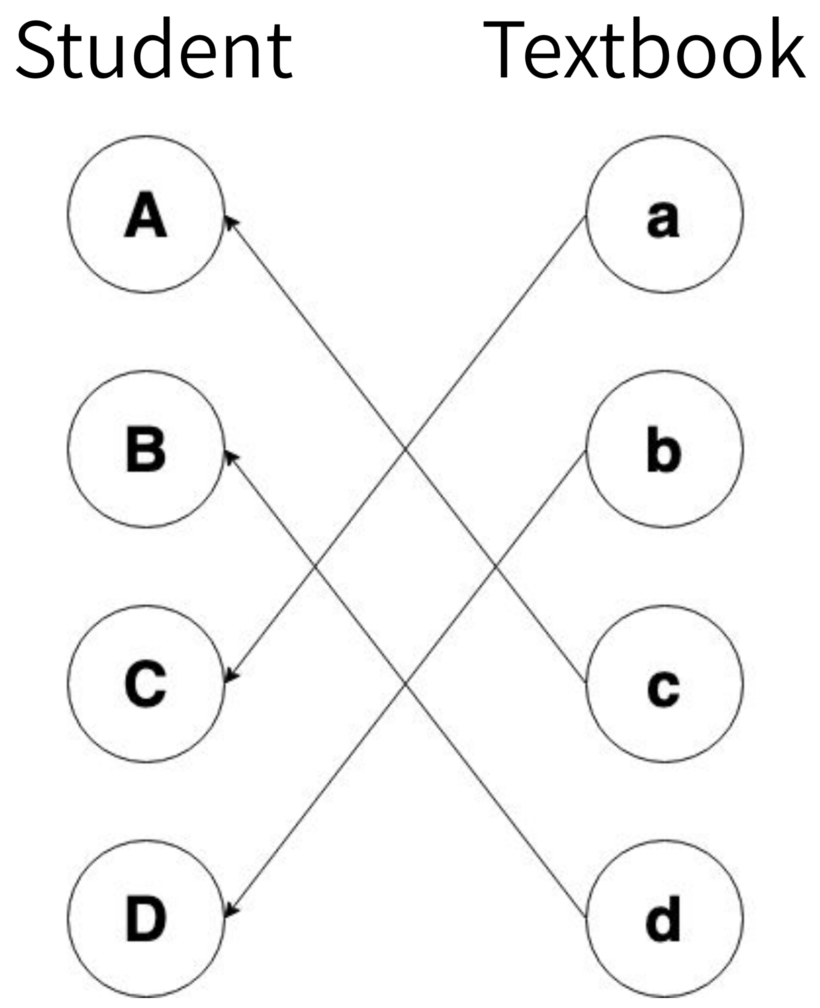
- Create preference lists of:
  - Students per book



  - Books per students
    - Prices of books
- Priority ranking can be updated

Gale-Shapley Algorithm

- Used for stable-matching problems
  - Help find stable pairs between two sets
- For each unmatched book, get the most preferred student
  - Pair with that student
  - Rematch existing pair if needed



Backend/API

- Node.js - JavaScript runtime environment
- Express - Framework to build REST API
- PostgreSQL - Relational Database
- TypeORM - Connect Node and Postgres

How we're different

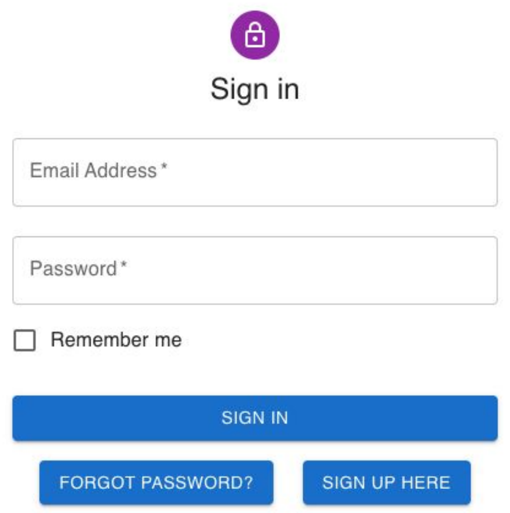
1. Matching Algorithm

2. Meetup feature

Authentication Feasibility

- Sign in verification

GW Support	Client/Server Verification
• OAuth • Opens up other features • Integration with registration	• Check email substring on both client and server • Bouncer email validation • Handle user authentication


Sign in

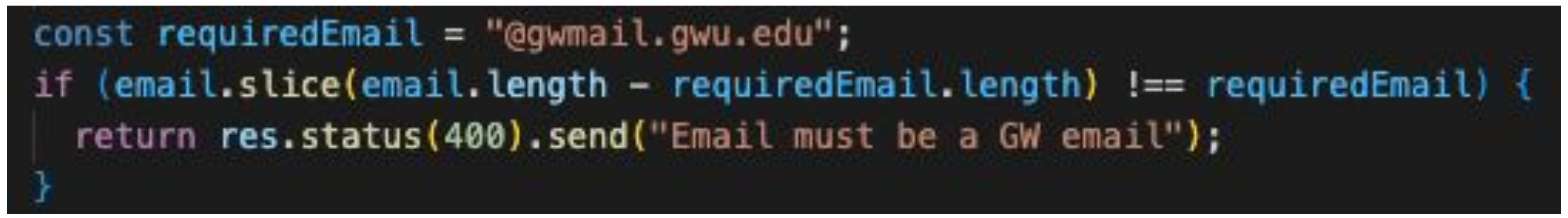
Email Address *

Password *

☐ Remember me

SIGN IN

FORGOT PASSWORD? SIGN UP HERE


```
const requiredEmail = "@gwmail.gwu.edu";
if (email.slice(email.length - requiredEmail.length) !== requiredEmail) {
  return res.status(400).send("Email must be a GW email");
}
```

Technical Challenges

?

Unknowns:

- Don’t have GW systems to verify users or financial aid

!

Risks:

- Security
- Book donations

Alpha Prototype

1. Sign in
2. Matching Algorithm/ Find Textbooks
3. Request what you need
4. Agree on time/ place
5. Complete exchange

Any Questions?