Uh-Oh! Project

Emma Bradford and Nolan Daly
Advisor: Prof. Fred Annexstein

To-Be-Read Booklist Generator

- Generate list of books for a user to read
- Handle user queries from a front-end
- Communicate to a back-end, obtain books from database
- Return and display booklist to user

Group Information

- Emma Bradford
  - emma.abe.bradford@gmail.com
- Nolan Daly
  - dalyng@mail.uc.edu
- Project Advisor: Fred Annexstein

Project Goals

Our project is focused on creating an interactive, web-based application that is capable of generating a list of potential books to read based on user criteria. The application will be split into two main parts: a front-end for user interaction and output, and a back-end for processing queries and interacting with our database. The front-end will allow the user to specify some criteria for filtering results, which will then be used by the back-end to generate a list from the database. This list will then be sent to the front-end of the application and displayed to the user.

Intellectual Merits

- Novel to Emma: a get request that filters and gets the desired books from database
- Novel to Nolan: Javascript using Node.js and React

Broader Impacts

- Making reading more accessible
- Broaden your book horizons
- Directing consumerism
- Connecting book nerd to books

Design Specifications

Box types:
Circle: Input/output
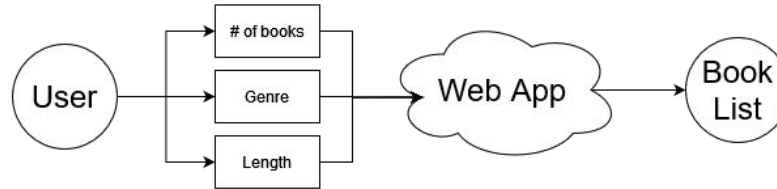Rectangle: processing step
Cloud: movement from front- to back-end (or vice-versa)

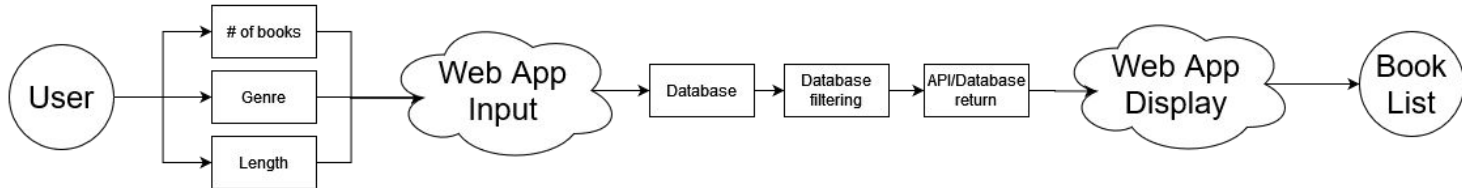Lines represent a movement of information between steps

D0

User → Web App → Book List

D1

User → [# of books | Genre | Length] → Web App → Book List

D2

User → [# of books | Genre | Length] → Web App Input → Database → Database filtering → API/Database return → Web App Display → Book List

Technology

- mongoDB
- Postman
- React
- Node.js
- Languages
  - Html, Javascript, Typescript, css
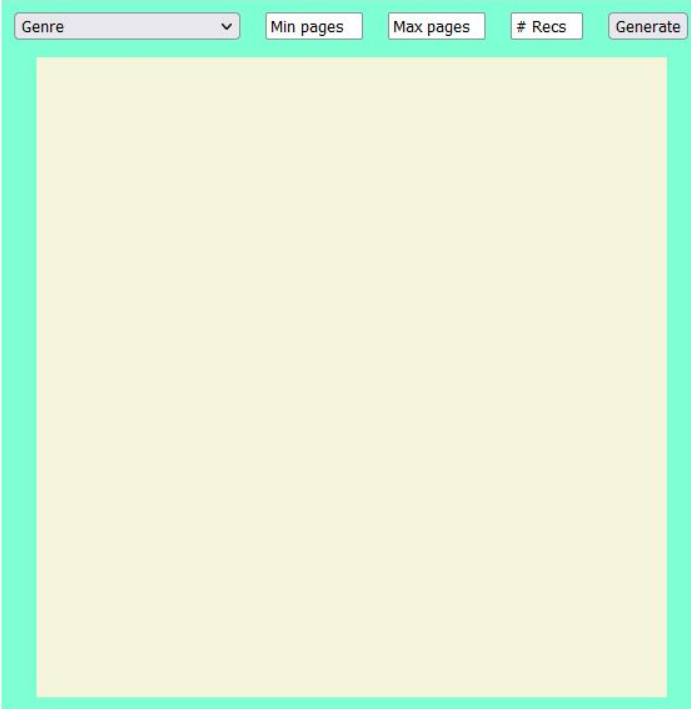
Milestones

Rudimentary frontend and backend

Frontend shows what the website looks like

Backend postman endpoints working

Frontend

To-Be-Read Book List Generator

| Genre ⌄ | Min pages | Max pages | # Recs | Generate |

Backend

_id: ObjectId("622e8cdef84edf123cb11b36")
> genre: Array
numPages: 416
author: "Adolfo Gilly"
title: "The Mexican Revolution"
__v: 0

_id: ObjectId("622e8cfcf84edf123cb11b38")
> genre: Array
numPages: 203
author: "Alan Macfarlane & Gerry Martin"
title: "Glass"
__v: 0

_id: ObjectId("622e8d13f84edf123cb11b3a")
> genre: Array
numPages: 301
author: "Alan Walker & Pat Shipman"
title: "The Wisdom Of The Bones"
__v: 0

localhost:3000

GET   localhost:3000

Params   Authorization   Headers (8)   Body ●   Pre

● none   ● form-data   ● x-www-form-urlencoded   ● r

```
1  {
2      "num rec": 2,
3      "max pages": 600,
4      "min pages": 300,
5      "genre": ["Romance", "LGBT"]
6
7  }
8
```
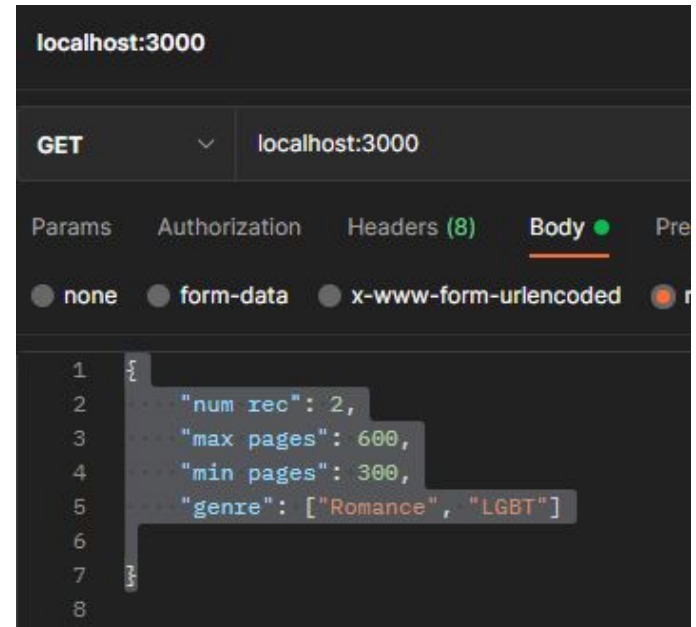
Challenges

Emma: Connecting to MongoDB with typescript, integrating book module into other parts of backend

Nolan: Learning frontend development and the associated languages (HTML, CSS, Javascript), and web development in general