Form Buster - Project Plan

Daniel Acosta, Christopher Demuro, Alex Merino,
Luka Miodrag Starčević

Goals and Motivation

- Everybody hates the current PDF system used at FIT
- Expecting an echo in outer space is more reasonable than expecting to hear back from an advisor quickly
- Docusign is an incarnation of the devil
- We want to design a web application that makes filling out and signing off forms easy and efficient
- No need for sending emails/filling out PDFs
- Get instant feedback on current state of your forms

Approach

- **Web-based Form Tracking for Students**
  - Sign in, choose a form, and your profile information will be autofilled
  - Fill it in, and send it off
  - Automatically sent to appropriate recipients
  - Track form progress
- **Creation and Management of Registrar Forms**
  - Administrators can create and manage custom forms
  - Make them available to students immediately
  - Configure who is able to start a form
- **Form Notifications**
  - Get notified when your form has been processed
  - Simple email notifications
  - Faculty will never miss your emails again

Novel Features

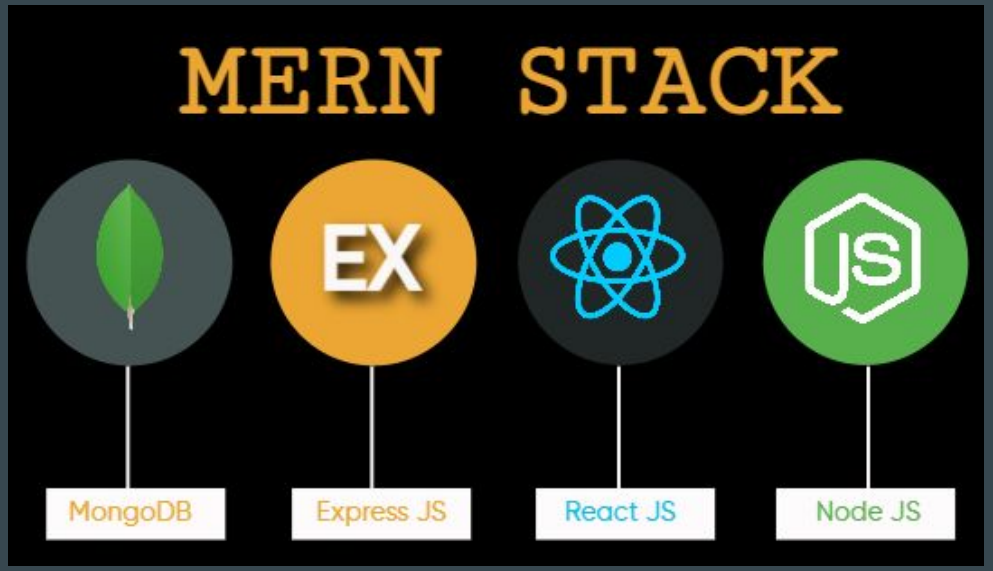
- Students can fill out and submit a course registration form online
  - Normally, Students have to email PDF files to advisors
- Students can track the progress of completed forms
  - Currently there is no way for students to track the progress of their PDF's
- Administrators can update and modify forms online
  - Currently have to modify PDFs and manually publish/email the changes to other administrators

Technical Challenges

- Working with external React packages such as Quill has proven to be difficult in how we would like to implement our Form Builder
- Attaching active forms to individual users and the required signatories is a complicated process and is the most difficult portion of assembling our database schema.
- Dynamic, per-user notifications for form updates require a complicated backend database query that touches several different parts of the database.
- Tracking the status and progress of forms and providing feedback when forms are rejected is difficult due to the arbitrary and potentially non-linear nature of the process.

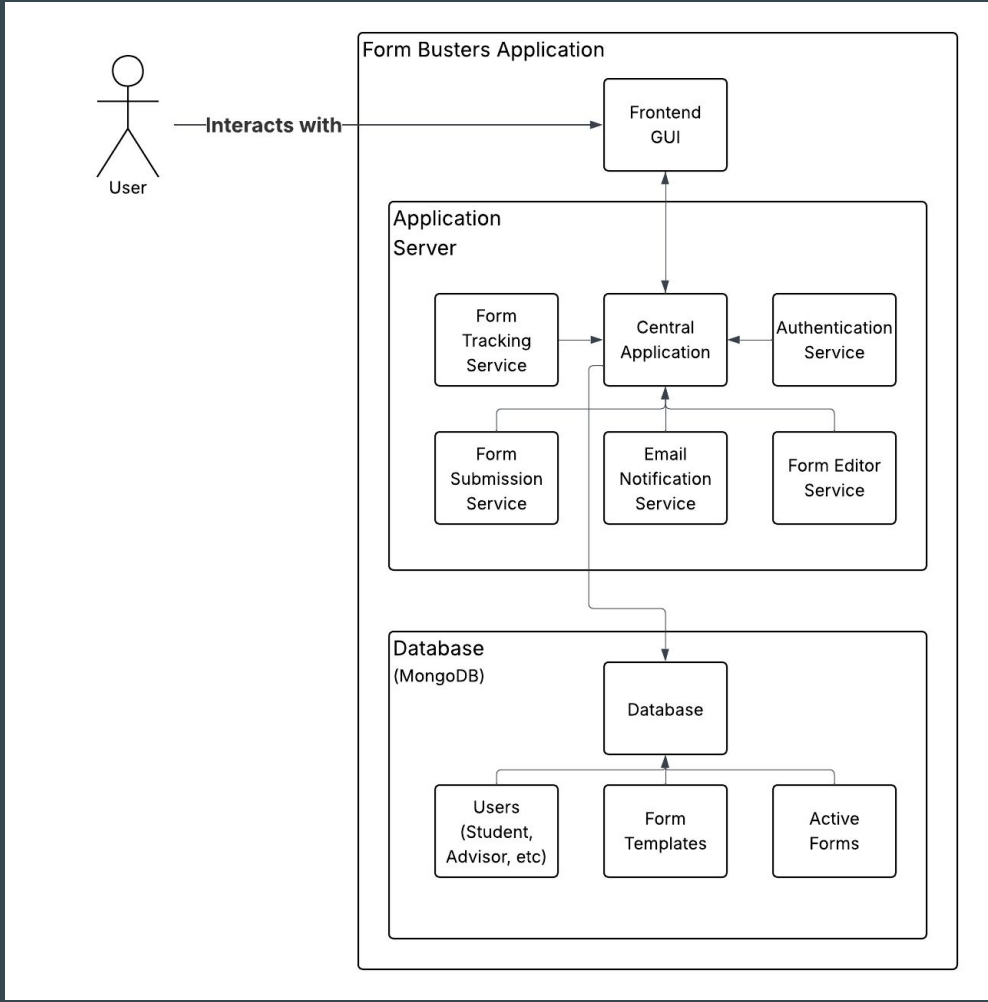
Necessary Tools

- MERN Stack
  - HTML, CSS, Javascript, React, Node, MongoDB
- Google Suite
  - Google Docs, Google Slides
- Discord
  - Communication, Meetings



https://medium.com/@amirakhaled2027/the-mern-stack-developer-roadmap-from-beginner-to-expert-bf991a9bcc1c

System Architecture Design



Progress Summary

Module/Feature	Completion %	To Do
Sign Up/Login	70%	Password auth, tying in forms to users
Dashboard/Form Tracker	40%	Warning for late signatures, page for more information about form
Menu Bar	100%	N/A
Form Builder	30%	UI Design, Adding Comments to forms
Account/Settings	70%	UI Design, change password/information
Notification Center	50%	Create notification objects, connect frontend to backend
Form Completion	50%	Add more forms, standardize UI and input fields
Home Page	20 %	Add more information about product and look more palatable

Milestone 4

- **Implement, Test Form Tracking:**
  - Update UI
  - Create a page to show full details of an active form
- **Implement, Test Active Forms:**
  - Attach correct users to the correct forms
  - Test reliability of backend database with form creation, updating, deletion, completion
- **Implement, Demo, Test Forms in the Inbox** :
  - Implement the dashboard retrieval of forms sent to the inbox after completion
  - Notification of new forms that need to be signed in faculty inboxe
  - Implement optional attached note explaining the form's outcome
  - Test user level security for viewing forms
- **Implement, Demo, Test Adding comments to Forms:**
  - Implement the ability for faculty to comment on submitted forms
  - Notify creator of forms with update

Milestone 5

- **Implement, Test, Demo Form Editor:**
  - Make creating, managing, and adding content to form templates user friendly
  - Test standardization and reliability of saving forms
  - Add form preview to editor so that changes to forms can be seen in real time as they are being made
- **Implement, Demo, Test Updating a Form:**
  - Implement the updating of an already existing template
- **Implement, Demo, Test Forms in the Inbox:**
  - Ensure Notification
- **Conduct evaluation and analyze results:**
  - User surveys, Reliability, Speed, Accuracy testing
- **Create poster for Senior Design Showcase**

Milestone 6

- Implement, Test, and Demo Form Editor
- Implement, Test, and Demo Email Notifications
- Test/Demo of the entire system
- Conduct evaluation and analyze results
  - Continue evaluations from milestone 5
- Create user/developer manual
- Create demo video

Milestone 4 Task Matrix

Task	Daniel	Chris	Alex	Luka
Implement, Test Form Tracking	0%	0%	100%	0%
Implement, Test Active Forms	100%	0%	0%	0%
Implement, Demo, Test Adding comments to Forms	0%	0%	0%	100%
Implement, Test User Authentication	0%	100%	0%	0%

Any Questions?

...