Program Review Information
System Management (PRISM)

Team Members: Leanne David, Andrew McLees, Justin Sarenas, Ben Solis
Faculty Advisor: Dr. Chengyu Sun
Office of Graduate Studies Liaisons: Dr. Karin Brown, Veronica Ramirez
Department of Computer Science
College of Engineering, Computer Science, and Technology
California State University, Los Angeles

Background
The Program Review Subcommittee (PRSS) reviews degree programs at CSULA on a regular basis. The original process involved collaboration and coordination manually over e-mail, but the goal of PRISM is to provide a system that automates the review process and streamlines collaboration workflow.

Challenges
Asynchronous Functionality
The variety of asynchronicity schemes utilized in JavaScript made certain tasks difficult. Knowing how to use callbacks, Promises, async/await, and RxJS Observables were necessary to understand how PRISM operates across the entire stack.

Database Design
Designing a NoSQL database is very different from designing a MySQL database, particularly in determining whether to use references to foreign objects or to embed the objects within a single model.

Workflow
- Consists of a side menu that is accessible on all authorized pages in the system that turns to a hamburger menu on smaller screens.
- Dashboard includes quick access to active reviews, review archives, and user actions upon login.
- Calendar contains events that store all agendas and meeting minutes.
- University Hierarchy lists the current active programs found in CSULA with their next review date.
- Resources holds all available files for users to download and/or upload at any given time.
- Group Management handles all access control for internal users.
- Settings is where users can update their information and/or password.
- External Upload is a separate and public page where external reviewers are invited by an administrator to upload their reports for a given active review.
- Email notifications are sent to users involved in charges for a review, deadline reminders, calendar events, uploads and approval of certain documents.
- Majority of adding, deleting, and manipulating data was done using models for simplicity and easy flow.

Reviews
- Acyclic graph that reflects the current status of individual documents for a review:
  - A blue fill indicates a document is ready to be worked on
  - A green fill indicates a document is finalized and can no longer be edited
  - A gray fill indicates a document needs its prerequisites completed first to be worked on
- Email notifications
- Individual reviewed documents consist of multiple revisions
- Allow creation of miscellaneous documents
- Completion estimates for the entire graph
- Comment support for a particular revision
- Automatic templating
- Access control for dean, department heads, and PRS members
- Send invites to external reviewers to upload their reports

Future Improvements
- Add a real-time in-browser document editor utilizing WebSockets
- Review graph customization
- Upgrade to Express 5 when it is out of alpha for HTTP/2
- Allow attaching multiple documents to Calendar events
- Implement more sorting and filtering functions for data tables
- Customizable email notifications for various documents and comments
- Allow replies and discussions for previous comments

Technologies
- and over 100 npm packages

Source Code
- prism-api
- prism-frontend

PrimeNG UI library
Initially, an Angular UI library, PrimeNG, was chosen for use on the frontend. It eventually became clear that it was too unstable for use in production, and the project was converted to use Twitter Bootstrap. This change came after a significant portion of the PRISM frontend had already been implemented using PrimeNG.

Performance
Latency is always present for HTTP requests and database queries. Limiting these requests and queries without limiting functionality was a consistent concern for the frontend and backend portions of PRISM.

Future Improvements
- Add a real-time in-browser document editor utilizing WebSockets
- Review graph customization
- Upgrade to Express 5 when it is out of alpha for HTTP/2
- Allow attaching multiple documents to Calendar events
- Implement more sorting and filtering functions for data tables
- Customizable email notifications for various documents and comments
- Allow replies and discussions for previous comments