

## Milestone #02 - Progress Evaluation

FIT Schedule Planner

Pedro Moura - [pmoura2020@my.fit.edu](mailto:pmoura2020@my.fit.edu)

Jordan Synodis - [jsynodis2021@my.fit.edu](mailto:jsynodis2021@my.fit.edu)

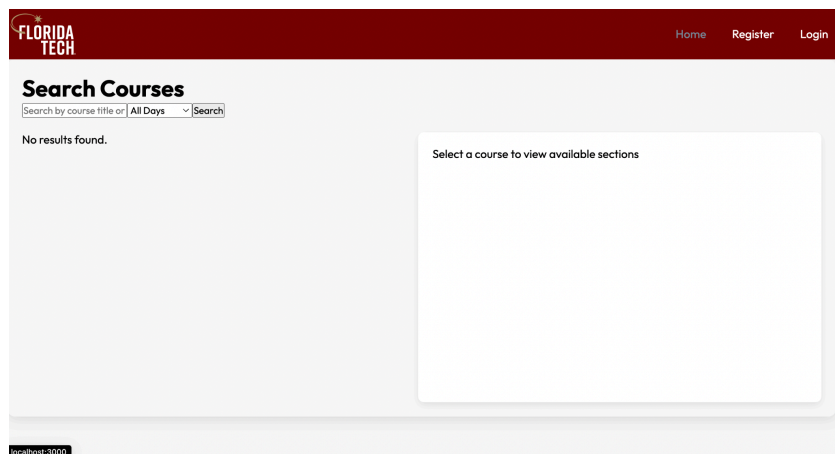
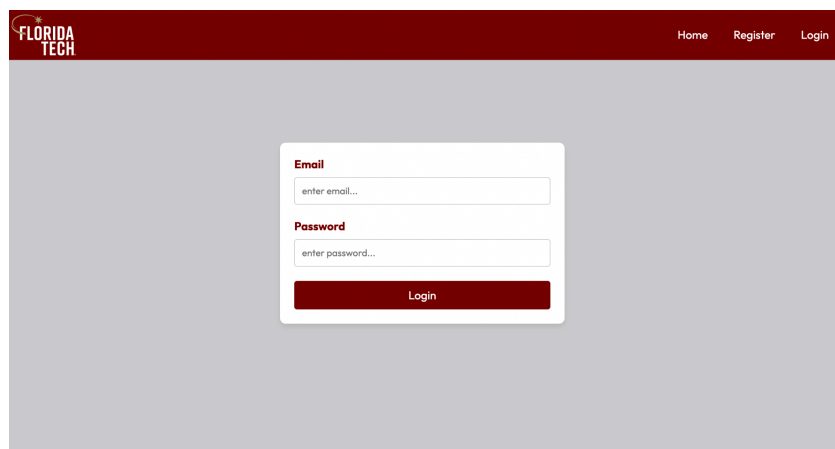
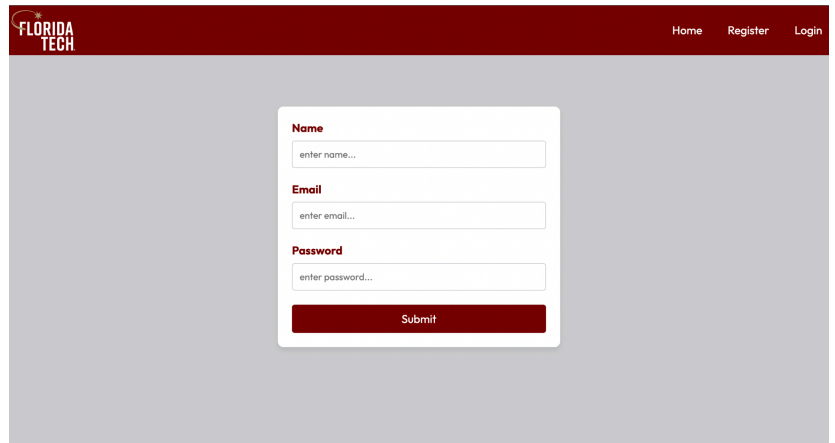
Dr. Fitz - [fneimbhard@fit.edu](mailto:fneimbhard@fit.edu)

Task	Completion	Pedro	Jordan	Todos
1. Implement, test, and demo traversing the interface	100%	50%	50%	none
2. Implement, test, and demo loading classes	100%	50%	50%	Improve filters, Debug Database
3. Implement, test, and demo loading the CAPP Degree Evaluation	50%	50%	50%	Process uploaded file
4. Implement, test, and demo accessing the program checklist	80%	50%	50%	Integrate current Python/Flask app to MERN app
5. Design Document	100%	75%	25%	none
6. Test Document	100%	50%	50%	none

- Discussion of each accomplished task (and obstacles) for the current Milestone:
  - Task 1: Implement, test, and demo traversing the interface
    - Our team developed a navigation system using React Router and created a responsive user interface. The implementation includes:
      - A navigation bar with Home, Register, and Login links
      - Protected routes requiring authentication for sensitive areas
      - Responsive design that works across different screen sizes
      - Making it consistent styling using CSS modules with FIT's color scheme ([FIT Color Palette and Typograph](#))
      - User context management for maintaining session state
    - Main obstacles: User context was very time consuming. We had to type the login data every single time we wanted to test a change. Getting data from database was difficult at first because of how React routes works.
    - Demos:

- Video: [Traversing Interface Demo](#)
  - (the actual traversing starts at 1 min 8s)
- Images:

○



- Task 2: Implement, test, and demo loading classes
  - Our team implemented the class loading system with the following features:
    - Search functionality for courses by title or course number
    - Filtering system for days of the week
    - Display of course sections with detailed information

- Interactive course selection interface
  - Visual schedule display showing selected courses
  - Conflict detection for overlapping class times
  - Prerequisites warning system
  - Class capacity checking
- Our team still needs to do some improving on the filtering system and debugging some database interactions.
  - Demos:

**Search Courses**

cse | All Days | Search

**CSE 1001: Fund of Software Dev 1**  
Credits: 4  
Prerequisites: MTH 1000, MTH 1012, MTH 1001, MTH 1002, MTH 1010, MTH 1020, MTH 2001, MTH 2010, MTH 2201, MTH 3200

**CSE 1002: Fund of Software Dev 2**  
Credits: 4  
Prerequisites: CSE 1001

**CSE 1101: Comp Discp & Careers 1**  
Credits: 1  
Prerequisites: None

**Sections for CSE 1001**

**Section 03**  
Days: MWF TR  
Time: 9:00 AM - 9:50 0930 AM  
Instructor: Silaghi  
Enrollment: 13/16  
CRN: 80415

**Section 04**  
Days: TR MWF  
Time: 9:30 AM - 10:45 1100 AM  
Instructor: Silaghi  
Enrollment: 16/16  
CRN: 81527

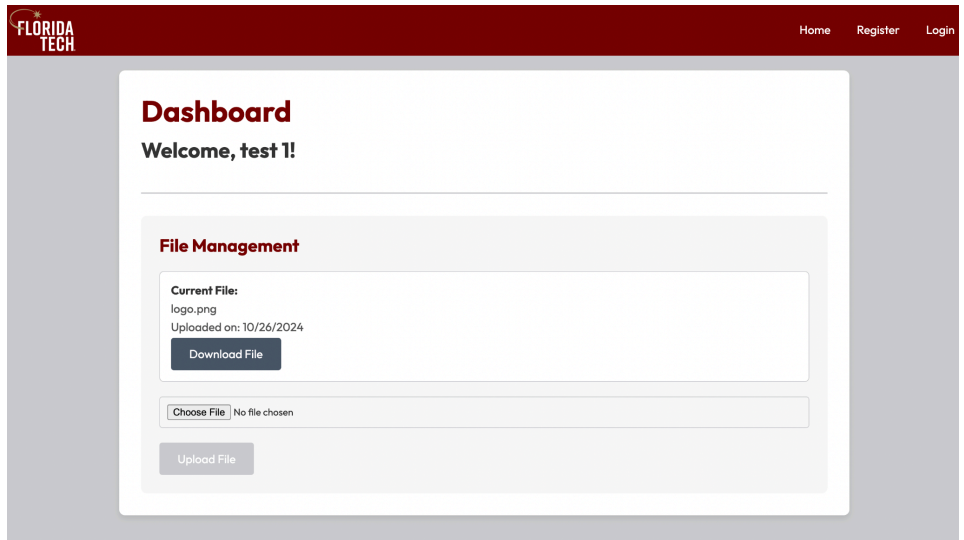
**Selected Sections**

**Weekly Schedule**

	Monday	Tuesday	Wednesday	Thursday	Friday
7:00 AM					
8:00 AM					
9:00 AM	CSE 1001 Section 03	CSE 1001 Section 03	CSE 1001 Section 03	CSE 1001 Section 03	CSE 1001 Section 03
10:00 AM					
11:00 AM					
12:00 PM					
1:00 PM		BUS 1301 Section 01 12:30 PM - 1:45 PM		BUS 1301 Section 01 12:30 PM - 1:45 PM	
2:00 PM					
3:00 PM	CSE 3411 Section 01 2:00 PM - 3:15 PM		CSE 3411 Section 01 2:00 PM - 3:15 PM		
4:00 PM					

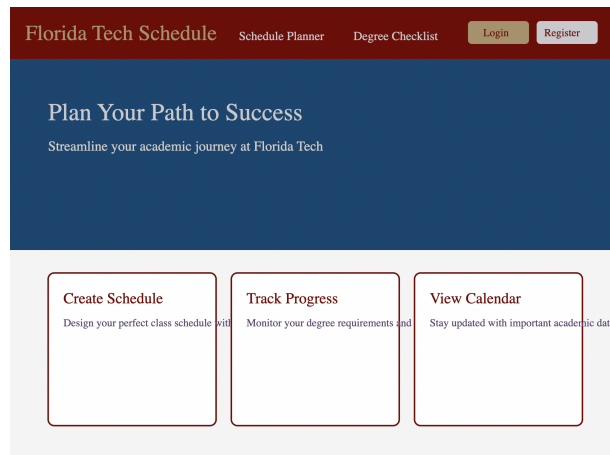
- Task 3: Implement, test, and demo loading the CAPP Degree Evaluation
  - Our team had some progress on the file upload system for CAPP evaluations:
    - Implemented basic file upload infrastructure
    - Created secure storage in MongoDB for user files
    - Added file download functionality
    - Implemented file state management

- Our team still needs to complete the processing of uploaded CAPP evaluation files to extract relevant course information.
- Demos:

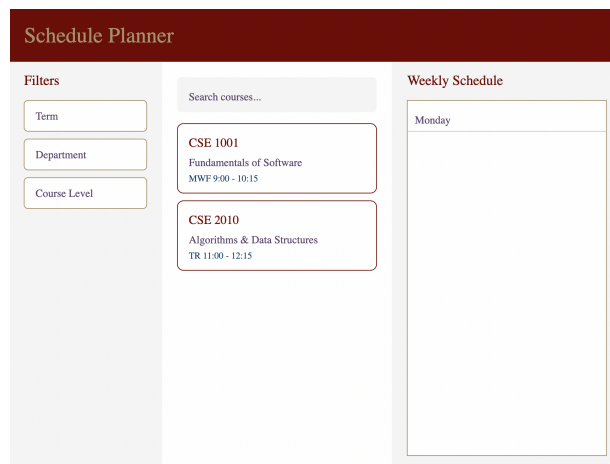


- Task 4: Implement, test, and demo accessing the program checklist
  - Our team progress on the program checklist feature:
    - Created Python script to get necessary data
    - Scraped data from FIT website
    - Formatted into JSON
    - Uploaded on Database
  - The main remaining task is integrating the existing Python/Flask application that processes program requirements into the MERN stack architecture.
  - Demos:
    - [Programs Demo \(0:17s\)](#)
- Task 5: Design Document
  - Our team finished the design document, by completing missing sections since Milestone #1:
    - Data Design
    - Component Design
    - Human Interface Design
    - Requirement Matrix

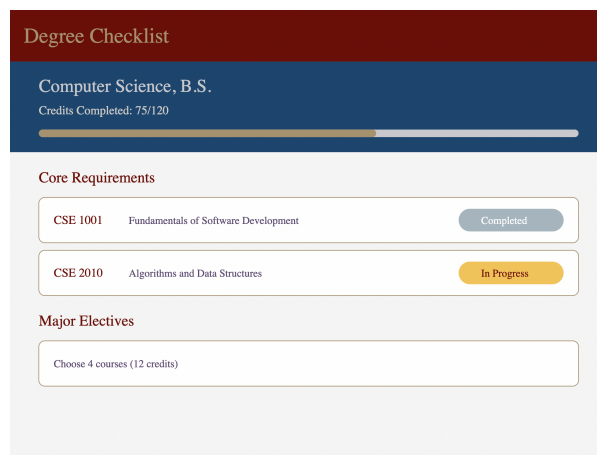
- Home Page:



- Schedule Planner Page:



- Degree Checklist Page:



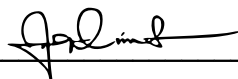
- Main Challenges: Because our team had no prior experience with making the User Interface, it was a challenge to come up with UI's for each page.

- Task 6: Test Document
  - Nothing was done because our team realized that we had finished this task before.
  - Main Challenges: Our team tried to foresee the most possible error our project could have, but we understand that due to lack of experience we might need to keep updating this document as needed.
- Discussion of contribution of each team member to the current Milestone:
  - Pedro Moura:
    - Developed the core React components for the user interface
    - Implemented authentication system with JWT tokens
    - Created the file upload/download functionality
    - Created the UI design
    - Implemented protected routing and user context management
    - Developed the database schemas and API endpoints
  - Jordan Synodis:
    - Built the course search and filtering system
    - Developed the schedule visualization component
    - Created the conflict detection algorithm
    - Implemented prerequisite checking (initial version)
- Plan for the next Milestone (task matrix):

Task	Pedro	Jordan
1. Implement, test, and demo loading the CAPP Degree Evaluation	50%	50%
2. Implement, test, and demo accessing the program checklist	50%	50%
3. Implement test, and demo advanced filtering options	50%	50%
4. Fix Database	50%	50%

Description for each task in task summary for the next milestone:

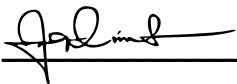
- Task 1:
  - Complete the file processing system to extract course information from uploaded CAPP evaluations
  - Implement data parsing to identify completed courses and requirements
  - Create a secure storage system for temporary CAPP data during the session
  - Develop error handling for various CAPP file formats
- Task 2:
  - Integrate the existing Python/Flask application with the main MERN stack
  - Create user interface components to display program requirements
  - Implement progress tracking visualization
  - Add functionality to compare CAPP data against program requirements
  - Develop recommendation system for remaining requirements
- Task 3:
  - Add multi-criteria filtering for course searches
  - Implement sorting by various parameters (time, professor, course level)
  - Create saved filter preferences
  - Add filter combinations for specific program requirements
  - Implement advanced search patterns for course descriptions and prerequisites
- Task 4:
  - Investigate Database to understand what might be causing error when fetching
  - Debug current database connection issues
    - (We think a possible fix is to set up access for any IP address)
  
- Date(s) of meeting(s) with Client during the current milestone:
  - See Faculty Advisor Feedback below
  
- Client feedback on the current milestone:
  - See Faculty Advisor Feedback below
- Date(s) of meeting(s) with Faculty Advisor during the current milestone: ...
  - Oct 28, 2024
- Faculty Advisor feedback on each task for the current Milestone
  - Task 1: ...
  - Task 2: ...
  - Task 3: ...
  - Task 4: ...
  - Task 5: Modify UI by using more modern sans-serif fonts and layouts to make it a bit more aesthetically pleasing
  - Task 6: ...
  - Task 7: ...

Faculty Advisor Signature:  Date: 10-28-2024

1. Evaluation by Faculty Advisor

- Faculty Advisor: detach and return this page to Dr. Chan (HC 209) or email the scores to [pkc@cs.fit.edu](mailto:pkc@cs.fit.edu)
- Score (0-10) for each member: circle a score (or circle two adjacent scores for .25 or write down a real number between 0 and 10)

Pedro Moura	0	1	2	3	4	5	5.5	6	6.5	7	7.5	8	8.5	9	9.5	10
Jordan Synodis	0	1	2	3	4	5	5.5	6	6.5	7	7.5	8	8.5	9	9.5	10

Faculty Advisor Signature:  Date: 10-28-2024