EE/CprE/SE 492 Bi-WEEKLY REPORT 1

Feb 2, 2023 – Feb 16, 2023

Group number: ***40***

Project title: ***Interactive learning platform to engage large size classroom***

Client &/Advisor: ***Md Maruf Ahamed (Client and Advisor)***

Team Members/Role:

- ***Tyler Miller: Backend***
- ***Alex Swenson: Backend***
- ***Jaden Ciesielski : Frontend***
- ***Brandon Burt : Frontend***
- ***Adam Walters : Frontend***
- ***Guan Lin : Backend***

○ **Bi-Weekly Summary**

Frontend has been working on getting polls ready in the live app to demo for our client/advisor at our next in person meeting (2/22/23). Backend created the APIS required for Polls, added APIS for upload and download of media files(pdf, csv, ect…), restructured the format of our APIS, and started work on Unit and Integration Testing. Also added security checks to routes so that only authorized users can make calls to certain APIs.

○ **Past 2 week accomplishments**

- **Created UI for Poll Participation Component - Brandon**
    - Created React files for Poll Participation container
    - Created styles.css for Poll Participation container
    - Working on connecting APIs from lecture file to component
- **Created UI for Student Pill component - Brandon**
    - Created React files for Student Pill Container
    - Created styles.css for Student Pill Container
    - Working on connecting APIs from lecture file to component
- **Working on connecting API calls to variables to pull in data from backend for the poll participation view - Brandon**
    - Working on variables within components, and hooking them into the API calls to connect the view to the backend.
    - Working on connecting the pop-up component to this view to display the app in the background as this page is live.
- **Updated Frontend Live Lecture API Class - Adam**
    - Updated API class to use new APIs that include course ID

    - Added all needed poll-related functionality/endpoints to the API class
- **Created Error Page for Frontend - Adam**
    - Created a new page route for any routing or page errors (e.g. 404)
        - Allows for custom error codes and text for “fatal” errors on specific pages
- **Updated Local Storage System with React Integration - Adam**
    - Updated the “DataStore” class on the frontend to support React hooks for specific data values
    - Cleaned up the implementation to be more encapsulated
- **Integrated API classes into Login Screen - Adam**
    - Previously, the Login page manually interacted with the backend to perform the login/logout
    - Updated the Login page to use the UserAPI object to perform requests
- **Created Poll Frame for Lecture Feed - Adam**
    - Created the poll “frame” (or message) to display within the live-lecture feed
        - Displays all available options for the poll
        - Depending on the current state of the poll, will highlight certain options as Selected, Correct, or Incorrect
        - Auto-requests the backend for up-to-date information and updates lazily
- **Created Pop-up component - Jaden**
    - Created a header component
    - Created automatic formatting to separate the header from all content
    - Created button functionality to pop up and close with ability to get inputs on submission
- **Created Poll Submission Form - Jaden**
    - Created PollFormHeader component
        - Gets the current person posting and lists it at the top and title input field
    - Created Poll Icon Component for easy filtering in the future
    - Created Poll Form Body component
        - Input field for question, answers and duration
        - Contains logic for retrieving all input fields which popup reads on submission
- **Integrating Poll Form With Backend - Jaden**
    - Got all information from the text fields
    - Generating the correct object to send to the backend
- **Automating API testing with Jest - Guan**
    - Created first automated test using Jest and supertest
    - Created test for health check
    - Created test to get all users created on db
        - Grabs user roles and users
- **Created Poll Apis - Tyler**
    - Created API for Poll Creation to be consumed by Front-End
    - Created API to set a poll to a closed state
        - Updated existing endpoints that grab polls to include new data fields
    - Created API to retrieve metrics from poll
        - Only Professor and TA can see all poll-results
        - Students can only see their results
- **Security to Post/Patch Endpoints- Tyler**
    - All Post/Patch endpoints have added security checks
        - Certain endpoints require PROFESSOR or TA role within the course
        - Certain endpoints require STUDENT, or PROFESSOR, or TA role within the

course

      - Appropriate error response returned if security not met
  - **Websockets for Questions - Tyler**
    - Added functionality for users to ask questions to the course professor
      - If Professor is online and user sends a “question” message, only the course Owner receives that message
      - Student alerted if professor is not online
  - **Jest UNIT Testing - Tyler**
    - Added 100 percent unit test coverage to lectures and poll APIS
    - Created framework for backend testing to use going forward
  - **Separated Routes and Services - Tyler**
    - Moved DB calls out to services files so they can be reused
    - Routes now call service methods instead of directly calling the DB
  - **File Upload/Retrieval APIs - Alex**
    - Created request upload service function
      - Used if message contains hasMedia flag set true
    - Created file upload endpoint
      - Accepts png or jpg
      - Doesn’t allow upload if no valid guid or uploading user didn’t create upload request
    - Created media service to interact with SQL database
    - Created ability to store media on server
    - Created endpoint to download file
      - Will refuse request if requester isn’t in related course

- **<u>Pending issues</u>**
  - Unable to get Express Websocket request to contain the same session accessible to normal APIs - Tyler

- **<u>Individual contributions</u>**

<u>NAME</u>	**<u>Individual Contributions</u>** *(Quick list of contributions. This should be short.)*	**<u>Hours these Past 2 weeks</u>**	**<u>HOURS cumulative</u>**
Tyler Miller	Created Poll APIS, Added Security to Patch/Post endpoints, Added Question Functionality to websockets, Added Jest testing for lectures and Polls, Separated Routes and Services	20	32
Brandon Burt	Developed multiple UI’s for poll participation view, worked on hooking the view up to the back end via the API calls, and getting the	18	28

	variables to display the data in the component view. Worked on hooking up our pop-up component to display the app in the background.		
Adam Walters	Updated and extended frontend API integration classes, created error page, updated local storage wrapper for frontend to include React hooks and improve encapsulation, updated API usage for the Login page, and created the Poll frame/message for the live lecture (no submission requests yet; just pulls information)	20	30
Alex Swenson	Created the file upload/download handling APIs and services, as well as bug fixes for them	10	25
Jaden Ciesielski	Created the popup component that takes in both header and content props as well as adding functionality to do actions upon clicking submission, created the poll submission form	17	29
Guan Lin	Created initial tests for automating api testing using jest and supertest. Made tests to get health check, users and roles of users	10	20

Comments and extended discussion

- *Discussed how professors will add students to their course*
    - *When creating a course, a join code will be created, once created the professor will send the join code to the enrolled students, which will allow students to join the course by giving them student role in the course*
- *Testing*
    - *Backend will do unit testing and integration testing*
    - *Frontend end-to-end/integration testing*

- **Plans for the upcoming week**
    - **Brandon:**
        - **Finish implementation of the poll participation component UI's and work on**

    - **hooking them up to the API calls to establish to end to end connection to the backend**
    - **Work on connecting the pop-up component to the poll participation view in order to have it display with the app blurred in the background once the poll participation component is completed and successfully hooked up to the backend.**
- **Adam:**
    - **Make poll frame/message fully interactable, sending requests to the backend upon click and updating selection**
    - **Update Live Lecture API to have the ability to stream in new polls**
    - **Update root Lecture page to listen for the new polls and add them to the page**
- **Tyler:**
    - **Add functionality to create a "join code" when course is created**
    - **Add Ability to stop to join course from working (close course)**
    - **Finish Unit Testing our API services and controllers**
- **Alex:**
    - **PATCH endpoints for various data types (users, courses, lectures, roles, msg) to update information**
    - **Associated tests**
- **Guan:**
    - **Add tests for all of the API's that we currently use**
- **Jaden**
    - **Finish connecting the submission form to the API endpoints**
    - **Finish creating the PollSubmission component and connecting API endpoints**

○ **Summary of bi-weekly advisor meeting (we meet bi-weekly)**

- Preparing for demo on Feb. 22nd for Polls
- Discussed Advisors colleague reviewing our repository
    - We want a review ASAP
- Updated Project to now be back in Phase 1 after discussion
- Discussed what Front-end and Backend has accomplished, and what features are available
    - Security, Polls, ect…
- Discussed Load Testing
    - When ready we will have advisor use one of his large lectures to test load that can be handled by our app