2 Project Plan

2.1 Project Management/Tracking Procedures

Waterfall+agile will be the style we use. This way, we can use the traits of both styles that will fit our group for the short amount of time we have available for our project. Along with that, a combination of waterfall and agile will allow us to change the design of the project in a faster manner if needed.

Our group will be using GitLab for version control and to track progress. We can create and assign tickets for the week using the software while we manage a master branch of the project.

2.2 Task Decomposition

- Create Frontend (ReactJS)
    - Create UI mockup
    - Decide on necessary components
    - Fetch data from back end
    - Display data correctly
    - Send data to back end
    - Add features
        - Users can signup and login with different roles
        - Students can ask questions that are answered by TAs and/or Professors
        - TAs and/or Professors can create polls for students to answer
        - TAs and/or Professors can see archived questions and polls
        - Professors can look up student and TA participation data
    - Write tests and third party review process
        - These will be tests for the code to check its functionality
        - The review process will be through third party people who will make comments on the UI, so we can get unbiased feedback.

Task Decomposition (Continued)

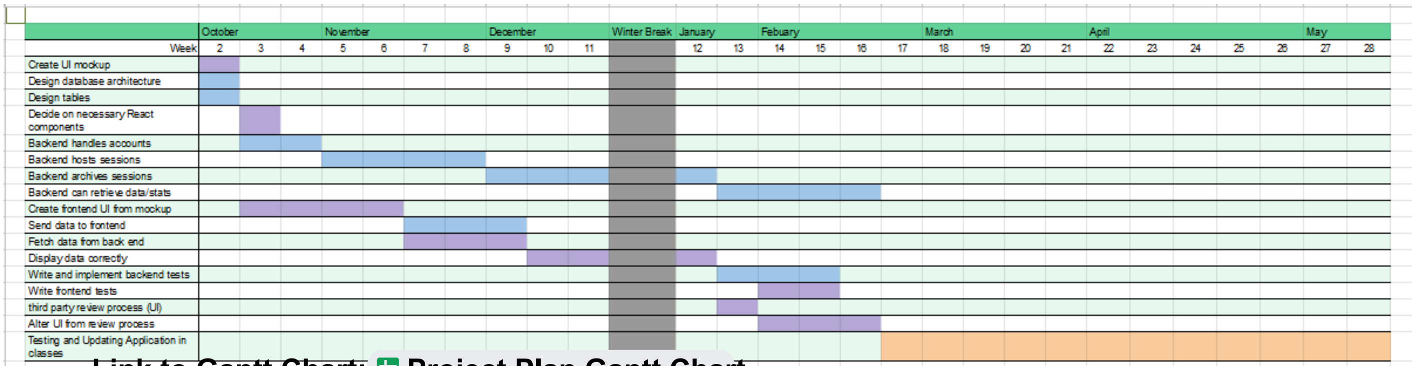
- Create Backend (Spring boot w/ MySQL)
  - Spring Boot
    - Setup database architecture
      - Make decisions on how we need to set up the backend to meet our requirements
    - Setup server architecture
      - Connect with the MySQL tables on the server
    - Encrypted Passwords
      - Passwords of every user will be encrypted when stored in the server table.
    - Write tests (JUnit and Mockito)
      - Decide on what we will be mocking to test our tables.
  - MySQL
    - Design Tables
      - Decide what data we will need to store in MySQL and the relations between the tables
    - Implements tables on given server
      - Create the tables in using MySQL on the school server

2.3 Project Proposed Milestones, Metrics, and Evaluation Criteria

Some Key milestones:

- Design phase is complete
  - First implementation of UI Mockups
  - Peer reviewed implementation of UI
- Round trip (front end can communicate with back end)
  - Web based
  - Android based
- SQL Database tables have been set up and will be in normal form with appropriate granularity.
- Connect Backend to MySQL and get the correct response on a local server.
- 2000 concurrent users connected with no error messages
- Base Features
  - Successful question and reply posted with text, image, audio, and video for both
  - Successful archive and opening of conversations
  - Successful poll and response of users
  - Correct response for class statistics for professor, TA, and user data

2.4 Project Timeline/Schedule



	October			November				December			Winter Break	January		Febuary				March				April					May	
Week	2	3	4	5	6	7	8	9	10	11		12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28
Create UI mockup	🟪																											
Design database architecture	🟦																											
Design tables	🟦																											
Decide on necessary React components		🟪																										
Backend handles accounts		🟦	🟦																									
Backend hosts sessions				🟦	🟦	🟦	🟦																					
Backend archives sessions								🟦	🟦	🟦		🟦																
Backend can retrieve data/stats													🟦	🟦	🟦	🟦												
Create frontend UI from mockup		🟪	🟪	🟪	🟪																							
Send data to frontend						🟦	🟦	🟦																				
Fetch data from back end						🟪	🟪	🟪																				
Display data correctly									🟪	🟪		🟪																
Write and implement backend tests													🟦	🟦	🟦													
Write frontend tests														🟪	🟪													
third party review process (UI)													🟪															
Alter UI from review process														🟪	🟪	🟪												
Testing and Updating Application in classes																	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧

Link to Gantt Chart: **Project Plan Gantt Chart**

2.5 Risks And Risk Management/Mitigation

Risks	Risk Mitigation
ISU servers may not be able to handle 2000 concurrent user requests	Optimize our application. Petition for more ISU servers.
ISU servers may not be able to archive all the question and participation data	Add features to limit the amount of data stored and/or the amount of time the data is stored. Petition for more storage on our server(s).
We might not have enough person hours to complete all of the features (either desired or required) of the application	Plan out which features we want to create first based on importance. Once the round trip is done, keep creating (and testing) new features on a weekly basis.
Students and/or professors might not want to use our application	Advertise our application to professors. Advertise to students that our application is free for students.
Sensitive data leak	Encrypt data traffic. Hash emails and passwords. Limit data access between roles.
Users may not know how to use our application	Spend time designing and implementing an intuitive UI. Change based on user feedback.

2.6 Personnel Effort Requirements

Task	Category	Projected hours (Person Hours)
Create UI mockup	Frontend	8 hours
Decide on necessary components	Frontend	8 hours
Fetch data from back end	Frontend	40 hours
Display data correctly	Frontend	40 hours
Send data to back end	Frontend	40 hours
Write tests and third party review process	Frontend	60 hours
Setup database architecture	Backend (Spring Boot)	15 hours
Setup Server architecture	Backend (Spring Boot)	20 hours
Encrypted passwords	Backend (Spring Boot)	20 hours
Backend handles accounts	Backend (Spring Boot)	20 hours
Backend hosts sessions	Backend (Spring Boot)	80 hours
Backend archives sessions	Backend (Spring Boot)	60 hours
Backend can retrieve data/stats	Backend (Spring Boot)	50 hours
Write tests	Backend (Spring Boot)	60 hours
Send data to frontend	Backend (Spring Boot)	10 hours
Design tables	MySQL	8 hours
Implement tables on given server	MySQL	15 hours
Implement Server logic to handle requests	Backend (SpringBoot)	15 hours

2.7 Other Resource Requirements

One resource we will need is ISU servers. We are provided with one server to work with, but if the project expands into something more permanent or is used campus wide, additional servers will need to be provided.

During the product testing phase, we may need additional resources (devices/mock users/etc) in order to ensure the app can meet client demand.