9-Plan for ID2

Project Manager

- Creating a list of what are the things each team lead needs to do
- Improving the meetings plan and consistency
- Having meetings with Stakeholder and Dr. Osgood to make sure that everything is on the right track

QA Team Lead

- Updated Test Matrix for ID 2
    - Populate the test Matrix with the new Unit tests done to all the pages implemented and the Integration tests between those
    - Consider Starting a system test
    - Updated Report of Results from Unit Tests and Integration Tests
- Update Test list
- Results of Code reviews done with the Code Checklist
- Logs acquired during the deliverable
    - ~~If Marcus has not figured out yet how to install Rollback this would need to be done previously~~ (Will be done by Build Master Instead)
    - Create the log scripts for out system and record the results in a document.
- Create spike prototypes as needed

Dev Team Lead

Coordinate with Team to ensure:

1. **File Upload Page Functionality:**

   - Implement file upload functionality on the File Upload Page. Users should be able to select files (Fitbit or Apple Watch data) and initiate the upload.

2. **Backend Integration for File Upload:**
   - Set up the backend to receive and handle file uploads from the frontend. Ensure proper validation and handling of different file formats (XML and JSON).
3. **Connection to Data Processing Module:**
   - Integrate the frontend with the data processing module (R repository). Upon file upload, trigger the data processing logic to convert raw data into a readable format (e.g., CSV).
4. **Display Processed Files:**
   - On the Processed Files Page, fetch and display the list of processed files. Allow users to view details or download the processed data.
5. **Jest Tests for File Upload and Processing:**
   - Write Jest tests to ensure the file upload functionality works correctly, and the processed files are displayed accurately.

Build Master (Tasks ordered from most important to least)

- Install Rollbar
- Setup new CI pipeline for playwright
  - You will need to create a new yml file to run this in .github/workflows
  - Refer to this for some guidance on running playwright https://playwright.dev/docs/ci-intro
- Setup Docker containers
  - Containerize the frontend, backend and R repositories
  - Also add a container for PostgreSQL version 15

    - Please write a document/wiki on a guide on setting up the local dev environment

- Make sure the backend uses the docker container for sql
    - Try to connect the backend to use the local sql container in docker
- Create config file for local/prod environments
    - Make sure api calls point to the right environment depending if currently on local environment vs production environment
    - Do this by defining a file where all our api calls can be called from? So when we implement changes to url based on environment it will be a quick and easy change
- Setup Migration scripts
    - Scripts to create the tables in the database on first startup. Also scripts to initialize data

Risk Officer

- Send out Top 3 Risk forms for ID2
    - This will be new or repeated risks for the upcoming deliverable
    - Hoping for 80% reply rate from members of the team
- Send out Calendar for group members to fill out other impending responsibilities
    - Includes upcoming midterms and other big project deadlines group members will have in the upcoming term.
    - With this information we will be able to either lessen the load for individuals or the group as a whole so we don’t risk coming up short in our next deliverable.