Group 4 EasyA Project Plan

Table of Contents

1. Revision History **1**

2. Project Plan **1**

2.1. Management Plan 1

2.2. Work Breakdown Schedule 2

2.3. Monitoring and Reporting 3

2.4. Build Plan 3

2.5. Build Plan Rationale 4

1. Revision History

Authors: Meagan Beckstrand, Eliza Black, Etienne Casal-Jouaux, Angel Soto, Connie Williamson

Date	Author	Description
1-13-2024	cjw	Created the initial document.
1-13-2024	cjw	Finished “2.1. Management Plan”, “2.2. Work Breakdown Schedule”, and “2.3. Monitoring and Reporting” sections with information from our first group meeting.
1-16-2024	meb	Finished “2.4 Build Plan” and “2.5 Build Plan Rationale”
1-19-2024	cjw	Completed References, Acknowledgements, and final copyediting.
2-04-2024	meb	Update Project Plan to reflect design changes made.
2-05-2024	cjw	Copy editing and final updating.

2. Project Plan

2.1. Management Plan

Our team is organized so that everyone has a voice. We do not have a defined leader, and to make decisions we bring up ideas and then deliberate on them. Because there are an odd number of people in the group, we are able to come to a majority decision on what we plan to do. Work is divided as evenly as possible, with each member choosing the work they find most interesting when possible. We have established communication channels via texting and online chat, and have agreed to meet twice per week for 30 minutes following CS 422 class time. We met another two times outside of this schedule, once in person and once in an online call.

2.2. Work Breakdown Schedule

We plan to develop as much in parallel as possible because of the short time frame for this project. The base of our schedule is that Eliza will create the base application, with options for admin and user mode, then we will continue to work on those modes in parallel. Items that will be worked on in parallel are indicated as sub-items, but are milestones in their own right. I have chosen to list them this way to be better understood as progress happening side by side.

1. **Tuesday, Jan 16th:** Discuss project logistics and write Project Plan and SDS. (Everyone)
2. **Thursday, Jan 18th:** Create basic display view using tkinter python module, including a way to choose between student and admin mode. (Eliza)
3. **Tuesday, Jan 23rd:**
    a. Create a script to convert the *gradedata.js* file into a *.csv* file, and make that data easily accessible in the rest of the program. (Etienne)
    b. Make "Create Graph" page UI in user mode using tkinter. The UI elements will be superficial only at this point. (Eliza)
4. **Thursday, Jan 25th:**
    a. Create a web scraper using Beautiful Soup python module for use in the admin mode update data function. (Connie)
    b. Learn about the python module matplotlib, and add a function to display a basic graph without customization. (Meagan)
5. **Tuesday, Jan 30th:**
    a. Complete the admin mode update data function by adding the ability to resolve data discrepancies between *gradedata.js* and the scraped data. (Angel)
    b. Add functionality to graph master function for customization (Meagan)
    c. Connect the updated graph master function to the UI so that changes in the UI affect the resulting graph. Should also connect the "Create Graph" button so that graphs with the options selected are created when pressed. (Etienne)
6. **Monday, Feb 5th:**
    a. Connect functionality from graph master function to the UI for different graph options (single class, single department, and all classes of particular level in a department.) (Connie)
    b. Add functionality to display multiple custom graphs side-by-side. (Meagan)
    c. Make presentation slides. (Angel)
    d. Write user manual documentation (this does not include technical documentation) using the finished or near-finished project. (Connie)

2.3. Monitoring and Reporting

We will keep track of who completes what items by referencing the Work Breakdown Schedule and GitHub repository alongside each other. In the case that an item takes longer than expected we plan to compensate by changing items between people so that each item receives an adequate

amount of work time. Discussion about how close to the schedule we are and what we should do to remain on schedule will occur in our biweekly meetings.

2.4. Build Plan

I. UI and Data Conversion: Done by Tuesday, Jan 23rd:

1. The first component of the system we will build is the UI, led by Eliza, who will use Tkinter to create an environment we can interact with, adding choices between admin and student mode.
    a. Research must first be done on the functionality of Tkinter, as none of the members on our team are very familiar with the application.
    b. Eliza will implement the system, and create two buttons, one for selection of admin mode, and one for selection of student mode, that will later be functional.
2. The initial data that we will be working with must be converted from gradedata.js to .csv such that the data is in a consistent format and is usable in the rest of the system. This will be done by Etienne

II. Data Scraping and MatPlotLib Basic Implementation: Done by Thursday Jan 25th:

1. Implement a way to scrape data from the Emerald Media Group grade-data that is provided. Connie will use Beautiful Soup to scrape the data, which will then be converted into a .csv file and will be used to update existing data at the discretion of an admin.
2. Using matplotlib, Meagan will develop a basic bar graph in the format that is similar to what is shown in the requirements, with the variables that will allow for customization later on once actual data is being inputted.

III. Complete Admin Mode and Connect UI to Graphs: Done by Tuesday, Jan 30th

1. Within admin mode, once an “Update Data” selection is made, implement the web scraper created by Connie and then Angel will work to resolve data discrepancies between this data and the data from gradedata.js.
2. Add buttons in student mode to the UI that will allow for multiple different options to be selected depending on the data that the student wants to see, including single class data, single department data, or all classes of a particular level in a department.
3. Begin inputting real data into the graph function. Then connect the UI and the data selected such that the student will be able to click “Create Graph” and see different graphs for the specific data they want.
4. Work to display multiple graphs side-by-side.

2.5. Build Plan Rationale

The intent behind the build plan was to divide the work into digestible chunks that could feasibly be worked on in parallel. The first key parts are creating the UI and converting the data into a usable form, that the rest of the code can be built around. For admin mode, the largest task is data scraping and resolving the discrepancies, which will be worked on by 2 members. Student mode is more interactive, with the data shown being selected by the user. Three members will work on creating the UI via Tkinter, which will then be connected to matplotlib to allow for visualization of the data. Once these tasks are completed, we will all come together to work on the documentation, and presentation of our completed project. Getting stuck at certain steps is a risk, however we budgeted enough time that hiccups can be overcome. Additionally, each task assigned is small enough to allow for members to float to different tasks as extra assistance is needed.