Project Plan

Garrett Bunkers, Alder French, Connor Maclachlan, Juan Rios, Blake Skelly

This file is the project plan for University of Oregon CS 422 Project 1 by Group 4. This document details how our group plans on managing, scheduling, assigning workload and reporting progress. It also details how our project is going to be built, and why we decided on that approach.

Management Plan

Team Organization

- Each member in the group will have a specific "role" (GUI, Graphing, Data Scraping, etc.) that will be their primary focus. Each member will be responsible for actively working on completing their assigned "role". If another member of the group needs assistance, any of the other members may need to support them. This allows for clear communication and understanding of the current state of the project.

Work Delegation

- The work will be divided using a Kanban board through the Trello website. The Kanban board is an agile project management tool that helps to visualize a team's workflow. The board is split up into five categories: Requested/To-Do, Work in Progress, Test, Done and Archived. During meetings, clear work tasks will be defined, and members will then assign themselves to different jobs. Each task's progress will then be monitored on the Kanban board. When a task is nearing completion, it moves into the Test category, and once finished and working properly, is moved into the Archived category. This allows us to see when tasks were started and completed.

Decision Making

- Decision Making for big portions of the project will mainly be decided as a group during one of our group meetings. We will identify large tasks that need to be completed, and decide collectively what the best course of action is. Any small tasks that relate to a certain member's role in the project can be decided by that member, as long as the member informs the rest of the group.

Communication

- Our team will meet in person at the Science Library twice a week on Tuesdays and Thursdays at 3 pm. Additionally, we will meet on Fridays at 3 pm if one of the members misses previous set days. We will also hold a meeting via discord on Saturday mornings at 10 am to discuss our work plans for the weekend and allow ourselves to be able to ask questions over the weekend if we get stuck or need to understand another team member's code. The main channel of communication outside of meetings will be Discord. Go to Monitoring & Reporting for more.

Work Schedule & Project Schedule

Work Schedule Milestones:

1. Initial Deliverable - Due Sunday Jan 15th @ 8pm
    a. Project Plan
    b. SDS
    c. SRS (use one given in class - no work needed)
2. Main Development Phase - Constant throughout duration of project
    a. Prototype V1: Basic Pipeline and Core Components - complete grade data scrape to python graph data visualization pipeline with little focus on user interface
    b. Prototype V2: Fulfill SRS Requirements and Design Interface - after testing the core project components from V1, shift focus to building out project such that it meets all the requirements specified in the SRS, such as implementing a simple user-friendly graphical user interface
3. Final Deliverable - Due Sunday Feb 5th @ 8pm
    a. Final Prototype: make final design changes to GUI and graph data visualization
    b. Final Testing: Along with testing the project and its components at virtually every stage of its development, rigorous testing to develop error and edge case catching will be done at this stage. Downloading the repository from GitHub onto a foreign system to test the install process and verify the program runs on different machines.

    c. Deliver Final Version: After fixing any errors that arise in final testing, turn in our project's final version.
4. Project Presentation - In class Monday Feb 6th
    a. Individual Presentation Work - Divide work from the presentation for each team member, and individually prepare our respective parts
    b. Rehearse Presentation Together - Meet in person or on discord and practice doing all our presentation parts together to make sure they flow well and fulfill presentation requirements.
    c. Give Final Presentation In Class

Project Schedule: We will be using a Kanban board on "Trello" to divide our work. This allows anyone to create tasks to work on, as well as assign themselves tasks others have created to work on. Thus we will have a flexible and ever-changing project schedule, with no single project manager or group member assigning work. See Kanban practices in the "Management Plan" above for more detail.

Monitoring & Reporting

Link to Kanban Board that tracks project progress: https://trello.com/b/9Tfbx4BK

To monitor and report the progress made by each individual team member we will be using a Trello workspace with a Kanban board. The Kanban board allows us to create tasks, assign them to ourselves or other team members, and add them to lists with labels so we can see the various stages our project is in. With the Kanban we can:

- Move tasks across lists labeled:
    - To-Do: Tasks that are yet to be started and assigned
    - Doing: Tasks that are currently being worked on
    - Testing: Tasks that have been worked on and are ready to be tested (e.g. code that has been pushed onto an individuals branch)
    - Done: Tasks that have been tested and verified (e.g. code pushed to the main project branch)
    - Archive: Old tasks that have been finished and recorded
- Assign members to the tasks so we can monitor who is doing what part of the project

- Assign dates and times to show when the particular task is due and when it was completed.
- Use the built-in automation to track when a task has been started after being moved to the “Doing” list

For our main means of communication, Discord will also be used. Individual progress on specific tasks can be monitored in closer detail. Our discord policies will be as follows:

- Members should check the discord channel at least once a day
- If members are messaged using the “@” command, including @member_name or @everyone, they should respond as soon as possible if not immediately
- There will be specific channels for communicating certain things like:
    - A code-exclusive channel where we will share code screenshots and talk about ideas for implementation
    - A project planning channel where we will discuss the project plan and any modifications that may happen throughout the duration of the project
    - A general channel for any miscellaneous communication that needs to happen amongst the group (e.g. meeting dates, etc.)

Build Plan

1. Import the data from the database or CSV file into local hard-drive
2. Clean and process the data as necessary, for example, removing any null or duplicate values
3. Calculate the percentage of "A" and "Ds and Fs" for each teacher
4. Use matplotlib to create a bar graph of the data and display it
5. Add user interface using libraries such as Tkinter
6. Add functionality to filter the data by Class, Department or Class-Level
7. Add functionality to filter the data by "All Instructors" and "Regular Faculty" using pandas query or condition

Rationale Build Plan

These steps help to build the overall project by providing a structured approach to obtaining, cleaning, analyzing, and visualizing the data.

1. Importing necessary libraries: By importing the necessary libraries, you will have access to all the tools you need to build the project.
2. Importing data: By importing the data into a local database, you will be able to easily manipulate the data and perform calculations on it, which is necessary for the analysis of the data.
3. Cleaning and processing data: By cleaning and processing the data, you will ensure that the analysis is accurate and the results are reliable, which is crucial for the credibility of the project.
4. Filtering data: The ability to filter the data by "All Instructors" and "Regular Faculty" will allow the users to focus their analysis on specific groups of teachers, which will make the project more versatile.
5. Creating bar graphs: By creating bar graphs of the data, we will make it easy for users to understand and compare the results. This will make the project more user-friendly.
6. Adding user interface: By adding a user interface, we will make it easy for users to interact with the tool and access the data. This will increase the usability of the project.