Software Development Plan

Gruuper

Grouping Web App powered by Machine Learning

4.1 Plan Introduction

This Software Development Plan provides the details of the planned development for the Gruuper Software which is a web application to streamline homework and test grouping for Teachers, Professors, and more.

The Gruuper project aims to revolutionize the way group formation occurs within educational settings by leveraging technology to streamline the process. At its core, Gruuper is a web application designed to facilitate group creation for homework assignments, tests, and other classroom activities. Through Gruuper, teachers can easily create classrooms and generate unique codes for student enrollment. Once enrolled, students provide information about their availability, skills, and interests through a user-friendly interface. Gruuper then utilizes a sophisticated machine learning model to analyze this data and form groups of students with complementary attributes, fostering collaboration and productivity. The rationale behind the development of Gruuper stems from the recognition of the challenges inherent in traditional group formation methods, such as manual sorting and coordination, which are often time-consuming and inefficient. By automating and optimizing this process, Gruuper aims to enhance the educational experience for both teachers and students, ultimately leading to improved learning outcomes.

Master Schedule Milestones:

- ➢ Weekly written status reports
- ➢ Weekly oral reports
- ➢ Software Development Plan Document [Updated]
- ➢ Requirements Specification Document [re-submit]
- ➢ Web app testing via in-classroom use
- ➢ Completed Machine Learning Model (ML)
- ➢ ML model effectiveness testing and tuning
- ➢ Fully furnished and consistent web app user interface (UI)
- ➢ Preliminary Version of Poster
- ➢ FINAL Project Presentation
- ➢ FINAL Product Delivery [Final Report and Code]

4.1.1 Project Deliverables

a. Weekly written status reports
   - Due: Weekly
   - Explanation: Status reports are really important for tracking your own progress. As a project is developed, being able to see which milestones are approaching, which ones have passed, and which ones are late can help you understand where you need to focus more effort for success. Status reports should also include a section for identified problems and possible solutions to those problems – this is the whole idea behind risk mitigation. Status reports are also a handy mechanism for documenting the amount of time you have spent on a project, which is very important when you are employed as a consultant. The more you know about the time required to complete a task, the better you can forecast the next job, so you can bill your clients accurately. See the material on the PSP page for more ideas.

b. Weekly oral reports
    - Due: Weekly
    - Explanation: Same as written reports except they will be presented orally.
c. Software Development Plan Document [Updated]
    - Due: Week 12
    - Explanation: This plan is intended to describe the process that will be used during the semester, leading to the production of all required documents and software for your project. This SDP has sections about all the gory details of the software project, including how the development team is structured, what the configuration management plan is, how the defined coding standards go, and on and on and on.
d. Requirements Specification Document [re-submit]
    - Due: Week 13
    - Explanation: The Software Requirements Specification document (SRS) is intended as a part of the contract between the customer and the solution provider. The document specifies EXACTLY what is being built.
e. Web app testing via in-classroom use
    - Due: Week 12
    - Explanation: Web app testing via in-classroom use involves deploying the Gruuper application within actual classroom settings to assess its performance and usability in real-world scenarios. This method allows testers to observe user interactions, group formation processes, and data management within the application. By gathering feedback from teachers and students, testers can identify any issues or areas for improvement and ensure that the application meets the needs of its intended users effectively. Additionally, in-classroom testing enables the evaluation of system performance under authentic conditions, including network connectivity and user behavior variability, to ensure reliability and scalability.
f. Completed Machine Learning Model (ML)
    - Due: Week 13
    - Explanation: This ML model is an integral component of the Gruuper application, responsible for processing user-provided information and generating optimal group configurations. Once completed, the ML model should demonstrate proficiency in accurately clustering users into cohesive groups, thereby enhancing collaboration and productivity within educational settings. The completion of the ML model signifies that it has undergone rigorous training and validation processes, ensuring its reliability and effectiveness in supporting the group formation functionalities of the Gruuper application.
g. ML model effectiveness testing and tuning
    - Due: Week 15
    - Explanation: In order to ensure the effectiveness of the ML model within the Gruuper application, rigorous testing and tuning

processes are essential. ML model effectiveness testing involves subjecting the model to various datasets representing different user profiles and scenarios to evaluate its performance in accurately grouping individuals based on predetermined criteria. This testing phase aims to assess the model's ability to produce meaningful and coherent groupings while minimizing errors and misclassifications. Additionally, tuning the ML model involves refining its parameters and algorithms to optimize performance and enhance accuracy. Through iterative testing and tuning cycles, potential weaknesses or limitations of the model can be identified and addressed, ultimately improving its reliability and effectiveness in facilitating group formation within the Gruuper application.

h. Fully furnished and consistent web app user interface (UI)
    - Due: Week 16
    - Explanation: The Gruuper project necessitates the development of a fully furnished and consistent web application user interface (UI). This UI serves as the primary point of interaction for users, facilitating intuitive navigation and seamless access to all features and functionalities of the application. The UI design should prioritize user experience, ensuring clarity, consistency, and ease of use across all screens and elements. By achieving a fully furnished and consistent UI, the Gruuper application aims to enhance user engagement, satisfaction, and overall usability.
i. Preliminary Version of Poster
    - Due: Week 14
    - Explanation: The preliminary version of the poster serves as a visual representation of the Gruuper project's objectives, methodology, and anticipated outcomes. It outlines the key components and functionalities of the application, providing stakeholders with a comprehensive overview of the project's progress and direction. The poster serves as a communication tool for presenting initial findings, insights, and challenges encountered during the project's development phase. Feedback obtained from the presentation of the preliminary poster version guides further refinement and enhancement of the Gruuper application in preparation for the final project presentation.
j. FINAL Project Presentation
    - Due: Week 16
    - Explanation: The final project presentation marks the culmination of the Gruuper project, showcasing the completed application and its capabilities to stakeholders, including project sponsors, instructors, and peers. During the presentation, the project team provides a detailed overview of the application's features, functionalities, and implementation process. Additionally, the team highlights key achievements, challenges overcome, and lessons learned throughout the project lifecycle. The final presentation offers an opportunity to demonstrate the Gruuper application's value proposition, relevance, and impact within

educational settings, as well as to receive valuable feedback and insights from the audience.

k. FINAL Product Delivery [Final Report and Code]
    - Due: Week 17
    - Explanation: The final product delivery encompasses the submission of the Gruuper project's deliverables, including the final report and codebase. The final report provides a comprehensive documentation of the project's objectives, methodologies, findings, and conclusions. It includes detailed descriptions of the application's architecture, design decisions, implementation strategies, and testing methodologies. Additionally, the report presents an analysis of the project's outcomes, including user feedback, performance evaluations, and future recommendations. Alongside the final report, the project team delivers the complete codebase of the Gruuper application, ensuring transparency, reproducibility, and accessibility for future iterations or extensions of the project.

4.2 Project Resources

4.2.1 Hardware Resources

Development Machine:

- Device: Apple M1 MacBook Air
- Processor: Apple M1 chip with 8-core CPU
- RAM: 16 GB unified memory
- Storage: 512 GB SSD
- Operating System: macOS Monterey

Application Host (Server-side):

- Service: Amazon Web Services (AWS)
- Instance Type: EC2 (t3.micro)
- Processor: Virtualized
- RAM: 1 GB
- Storage: EBS (20 GB SSD)
- Operating System: Ubuntu Server 20.04 LTS

Database Server:

- Service: Google Firebase Firestore
- Plan: Spark Plan
- Storage: Cloud Firestore (scalable)

Mobile Device Testing:

- Devices: iPhone 12 Pro (iOS)
- Operating System: iOS 15

4.2.2 Software Resources

During the development of the Gruuper project, the following software tools will be required:

Development Environment:

- Text Editor: Visual Studio Code (Version 1.63.2)
- Web App Engine: Chrome (Latest)
- Operating System: macOS Monterey (for development machine)

Frontend Development:

- Framework: React.js (Version 17.0.2)
- Package Manager: npm (Version 8.4.0)
- Version Control: Git (Version 2.35.1)
- Design Tool: Figma

Backend Development:

- Server-side Framework: Node.js (Version 16.14.0)
- Database: Google Firebase Firestore
- Cloud Platform: Amazon Web Services (AWS)

Mobile Development:

- IDE: Visual Studio Code (Version 1.63.2)
- Programming Language: JavaScript (for web app)

Testing:

- Testing Framework: Jest (Version 27.5.0) (Potentially)

Collaboration and Communication:

- Project Management: Gantt Chart
- Communication: Discord

Version Control Hosting:

- Platform: GitHub

These software tools, along with the specified versions, will facilitate the development, testing, and collaboration efforts required for the successful implementation of the Gruuper application.

4.3 Project Organization

The Gruuper project is organized into three major functions, each overseen by a dedicated team member:

1. Development Team (Brady)
    - Responsibilities: Brady is responsible for overseeing the overall development of the Gruuper application, including frontend and backend development,

database integration, and ensuring the alignment of technical solutions with project requirements.
    - Development Plan: Brady will lead the development team in implementing the frontend web application using React.js, designing and deploying the backend infrastructure on AWS, integrating Google Firebase Firestore for database functionality, and collaborating with the ML team to integrate the ML model into the application.
2. Machine Learning Team (CJ)
    - Responsibilities: CJ is responsible for developing the machine learning model for Gruuper, which will be written in Python and exposed as a separate API. This model will analyze user data to form groups based on interests and work styles.
    - Development Plan: CJ will work on developing, training, and fine-tuning the machine learning model using Python libraries such as TensorFlow or PyTorch. They will also create a RESTful API to expose the model's functionality and integrate it seamlessly into the Gruuper application.
3. Quality Assurance Team (Sam)
    - Responsibilities: Sam is responsible for ensuring the quality and reliability of the Gruuper application through comprehensive testing and quality assurance processes. This includes conducting unit tests, integration tests, and user acceptance testing to identify and address any bugs or issues.
    - Development Plan: Sam will develop a testing strategy and execute test plans to validate the functionality and performance of the Gruuper application across different platforms and environments. They will collaborate closely with the development team to address any identified issues and ensure a high-quality end product.

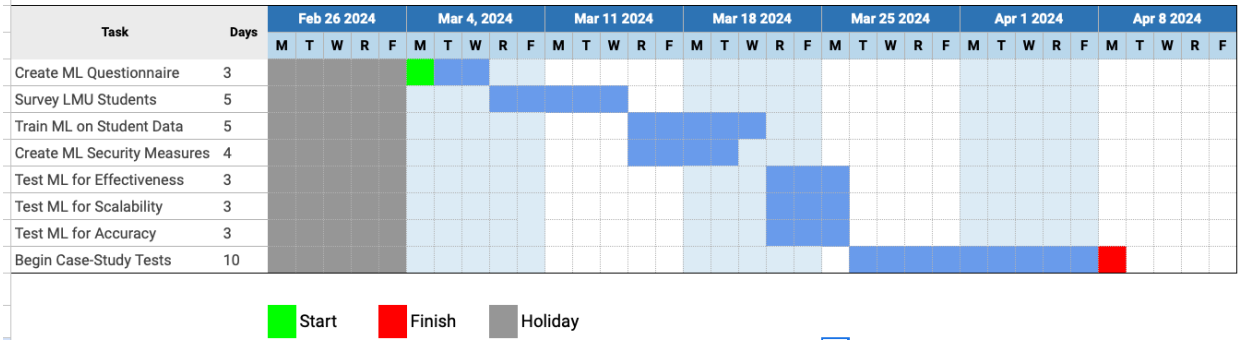
By dividing the project into these major functions and assigning responsibilities to each team member, the Gruuper project is well-organized and equipped to efficiently achieve its goals.

4.4 Project Schedule

This section provides schedule information for Gruuper. After our first semester, we have built up the main infrastructure of the Web App. Here are our remaining tasks for the semester:

1. Create ML Questionnaire
2. Survey LMU Students
3. Train ML on Student Data
4. Create ML Security Measures
5. Test ML for Effectiveness
6. Test ML for Scalability
7. Test ML for Accuracy
8. Begin Case-Study Tests

4.4.1 PERT / GANTT Chart



4.4.2 Task / Resource Table

Task	Resources
Create ML Questionnaire	Surveying, Google Forms, Academic Research
Survey LMU Students	LMU Students & Faculty
Train ML on Student Data	Method TBD
Create ML Security Measures	Manual performance and security testing
Test ML for Effectiveness	Manual testing
Test ML for Scalability	Manual Testing
Test ML for Accuracy	Manual Testing
Begin Case-Study Tests	LMU Students & Faculty