Requirements Specification Document

Gruuper

Grouping Web App powered by Machine Learning

5.1 Introduction

Gruuper maintains three components that work together:

- A **web app** for simple user interaction and group formation
- A **database** for storing user information and classroom members
- An **ML model** to group people with similar interests and workstyles

The remainder of this document is structured as follows:

- Section 5.2 contains the functional requirements for the application
- Section 5.3 describes the application's performance requirements
- Section 5.4 includes the environment requirements

5.2 Functional Requirements

This section describes the requirements each component of the application must meet to be a fully functional online game. At a high level, this means that the game server must be engineered to fully support both human and AI clients, and that the datastore must be able to store both game state and user information. The following sections describe the specific requirements that must be fulfilled to meet these goals.

5.2.1 Web App (Frontend) Functional Requirements:

5.2.1.1 Users shall be greeted with a sign in page with email and password authentication.

5.2.1.2 The front-end shall utilize the Material UI library for a clean and consistent design across all components.

5.2.1.3 There will be a screen with the option to create or join a classroom.

5.2.1.4 The website shall feature a profile page for the user to view their information.

5.2.1.5 The profile page shall include a survey to give information to the ML model

5.2.1.6 The classroom page shall have a list of the members of the class, as well as their groupings.

5.2.1.7 When in the classroom view, a professor shall have the ability to randomize groups, select group size, and manually edit groups

5.2.1.8 Students shall not have this ability

5.2.1.9 There shall be a section where users can view a list of their enrolled classes, and be able to view them with a click

5.2.2 Firebase Datastore Functional Requirements:

5.2.3.1 The database will be hosted on Google Firebase using the Realtime database.

5.2.3.2 The database will store user authentication information and process log-in attempts.

5.2.3.3 The Realtime Database will store user data such as name, user ID, and enrolled classrooms

5.2.3.4 The database shall also store classroom information such as the creator, group pairings, and members

5.2.3.5 Users and data can also be deleted from firebase by an administrator.

5.2.3.6 Database administrators will be able to track user activity in Firebase.

5.2.3.7 The authentication shall allow users to update and change their email and password.

5.2.3.8 The authentication shall validate users' emails when signing up for the first time.

5.2.3 Machine Learning Functional Requirements:

5.2.4.1 The model shall group people in the class by their interests and workstyles

5.2.4.2 The model shall get its information from each user's personality survey answers

5.2.4.3 The model shall be trained on labeled data.

5.2.4.4 The model shall adhere to security best practices, such as encryption of sensitive data and access control mechanisms

5.2.4.5 The model shall allow instructors to override automatic groupings to meet specific pedagogical needs.

5.2.4.6 The training process shall be scalable to large datasets such as large numbers of students in a given classroom.

5.3 Performance Requirements

The performance requirements of the Gruuper application are crucial to ensure a seamless user experience and efficient operation. These requirements dictate the responsiveness, reliability, and scalability of the system under various conditions. In this section, we outline the performance criteria that the application must meet to meet user expectations and support its intended usage scenarios.

5.3.1 Web App (Frontend) Performance Requirements

5.3.1.1 The webpage must load within one second.

5.3.3 Firebase Datastore Performance Requirements

5.3.3.1 The datastore shall respond to queries from the web app within 5 seconds. If the datastore fails to do this, the request should be logged for debugging.

5.3.4 Machine Learning Performance Requirements

5.3.4.1 The model shall be scalable to handle increasing volumes of data and computational resources.

5.3.4.2 The model shall respond to queries from the web app within 10 seconds.

5.4 Environment Requirements

The following sections describe what is required for a developer to run and deploy the application.

5.4.1 Development Environment Requirements

A developer for Gruuper will need a system meeting the following requirements to build the web app:

- Windows, MacOS, or Linux operating system
- A modern web browser, such as Chrome or Firefox
- Node.js v18.17.1 runtime and Python v3.11 or newer

In development, the only external server required is the Firebase datastore, which is automatically kept online and accessible. All other components can be run locally on the developer's system, and source files can be modified with any code editor. The Smart Grouping Machine Learning Model, while in development, will be hosted using Flask. The web app uses node.

5.4.2 Production Environment Requirements

Currently, the web app is deployed on Vercel out of ease. Eventually, both the web app and the Smart Grouping Machine Learning API will be hosted on AWS.

5.4.3 Execution Environment Requirements

- Operating System: Compatible with modern web browsers on Windows, macOS, and Linux.
- Internet Connection: Stable internet connection required for accessing the web application.