**Technical Specification Sheet**

**Product Documentation**

(Remade preference system)

1. System Description

This system is a brand new system based on the previously used Google Sheets. Students can choose seven preference items and assign values to them (between 1-7). The collected data is stored in Google Sheets, which can be viewed, edited, and downloaded by the teaching team.

The preference model includes:

* Positive preferences:teams can choose up to 7 positive preferences

The system manages team numbers and student IDs to ensure that submissions are only accepted from valid, registered teams. Submitted data is saved into a Google Sheet.

Requirements

1. **Functional Requirements (FR)**

* **FR1**: User Interface

FR1.1: The system should provide a web-based user interface that allows users to input and submit their preference items.

FR1.2: Users can select up to 7 preference items.

FR1.3: Users should be able to save their choices through the web interface and be able to revisit and modify these choices as needed.

* **FR2:** Data Storage and Management

FR2.1: All available preference items and their details should be stored in a separate worksheet in Google Sheets.

FR2.2: User selections should automatically be recorded in another worksheet in Google Sheets.

FR2.3: When users modify their selections, the system should automatically update the corresponding data in Google Sheets.

* **FR3:** Google Apps Script Features

FR3.1: Use Google Apps Script to handle and validate form inputs to ensure users do not select more than 7 items.

FR3.2: The script should implement data read and write operations, providing seamless data synchronization between the user interface and Google Sheets.

1. **Non-Functional Requirements (NFR)**

* **NFR1**: Performance

NFR1.1: The system should respond quickly when users submit data, ensuring processing and storage operations are completed within 5 seconds.

* **NFR2**: Usability

NFR2.1: The interface should be simple and intuitive, enabling users to complete tasks without additional guidance.

NFR2.2: Provide basic error feedback mechanisms to help users correctly fill out the form.

* **NFR3**: Security

NFR3.1: Ensure user data is protected through encryption and appropriate access controls.

NFR3.2: The application should prevent unauthorized users from accessing or modifying data.

* **NFR4**: Reliability

NFR4.1: The system should ensure the accuracy and integrity of all user data, especially during data synchronization and updates.

User stories

As a team, I want to submit project preferences so that my project assignments reflect my team’s needs.

As a user, I want to view instructions clearly before making a submission to ensure my preferences are submitted correctly.

As a developer, I want to manage submissions through Google Apps Script to simplify the backend logic.

You can see the user Story here: [User Story](https://docs.google.com/spreadsheets/d/1XHC7Md2xk-OWQvvGMNbcslDmw10_Eo51/edit?gid=1732174599#gid=1732174599)

1. System Diagram

* Flow diagram

**Start**: Marked as "start," it represents the beginning of the process.

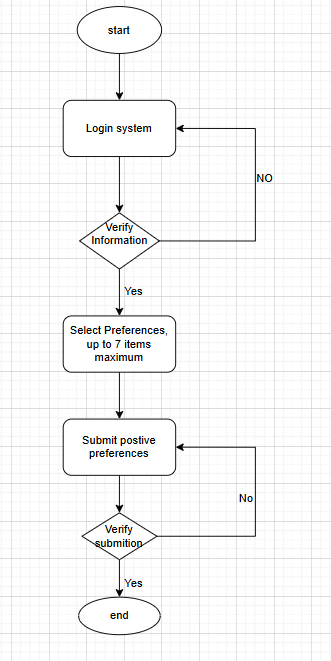
**Login System**: Users need to log into the system by entering the necessary information.

**Verify Information**: The system verifies the information entered by the user. If the information is incorrect, the process returns to the "Login System" step; if correct, it proceeds to the next step.

**Select Preferences**: After verification, users can select their preference items, with a maximum of 7 items allowed.

**Submit Preferences**: Users submit their chosen preference items.

**Verify Submission**: The system verifies the submission. If the submission does not meet the requirements , the process goes back to the "Submit Preferences" step for resubmission; if it meets the requirements, the process ends, marked as "end".



1. System Details

**Framework/Software:**

* + 1. Google Apps Script for backend scripting
    2. HTML, JavaScript, jQuery for frontend
    3. Google Sheets as the database

**Versions Used:**

* + 1. Google Apps Script platform
    2. jQuery 3.0.0, Bootstrap 3.3.7

**File Structure:**

* + 1. index.txt: Contains the main form page with inputs for positive and negative project preferences.
    2. formPage.txt: Manages the form submissions for team details, including student IDs and team passwords.
    3. Code.txt: Includes the backend logic for processing and storing the preferences and team data in Google Sheets.

1. System setup
   1. Accessing the Google Apps Script editor.
   2. Deploying the web app to manage form submissions.
   3. Connecting the script to the Google Sheets for data storage.
2. System Testing
   1. **Functionality Testing:**
      1. **Positive Preference Submission:** Testing ensures that the system accepts up to 7 positive preferences, verifies that no preferences are missing, and stores them correctly in the sheet.
   2. **Error Testing:**
      1. **Form Validation:** Verifies that forms will not submit without required fields (email, team number, password, and student IDs).
      2. **Duplicate Submission:** Ensures that new submissions overwrite previous ones for the same team.
3. System troubleshooting

**Known Issues:**

If multiple teams submit preferences for the same project at the same preference level, data is concatenated.

**Limitations:**

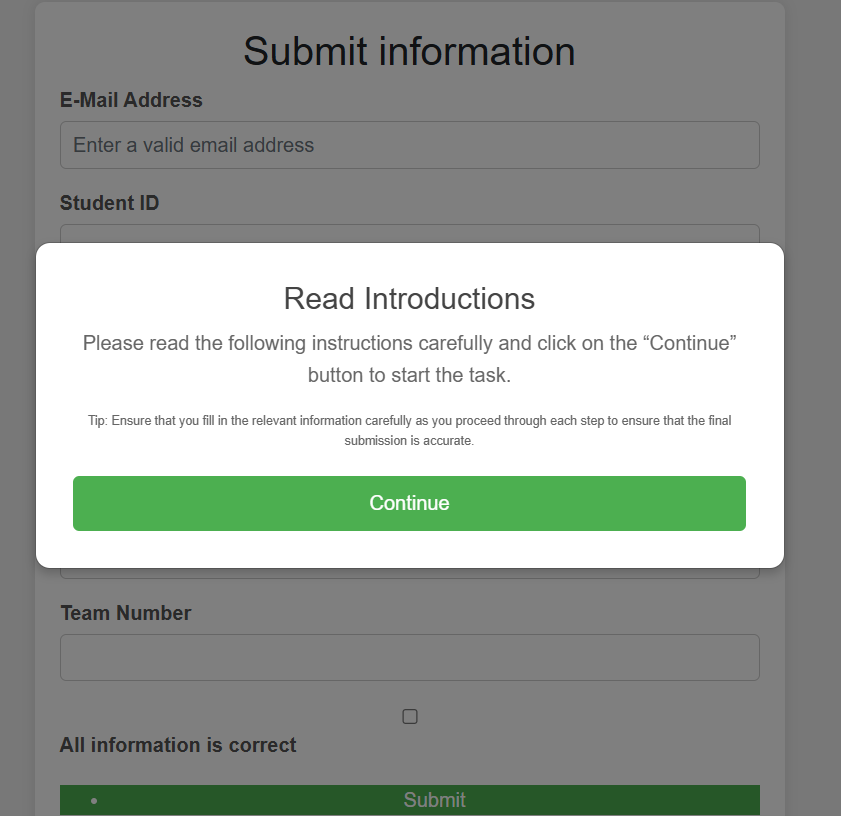
* Currently only supports up to 7 positive preferences per submission.
* Only the most recent submission is stored per team

**User Manual**

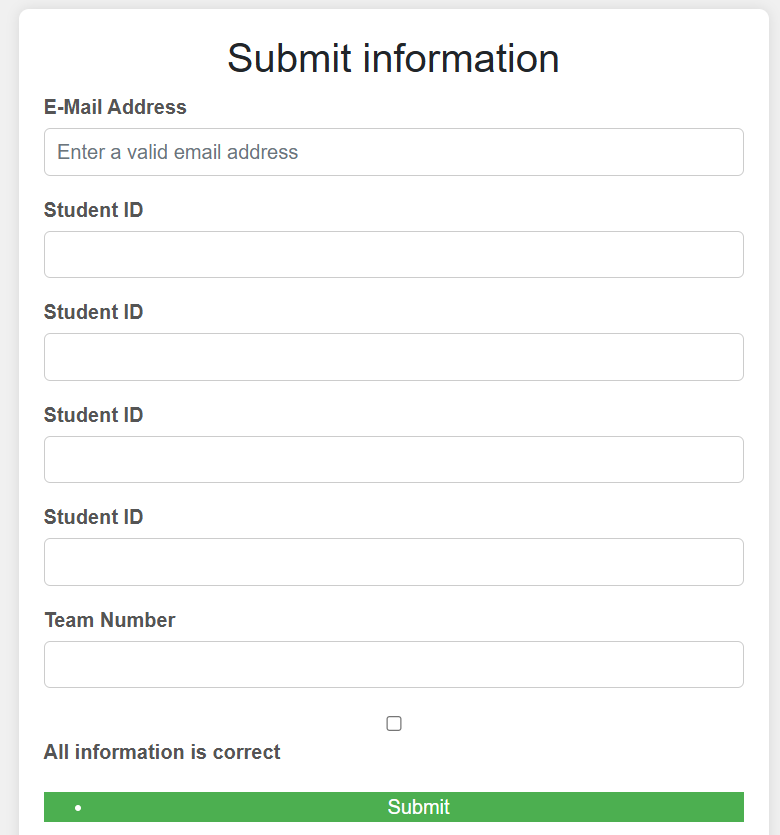
1. Software information

This system is a brand new system based on the previously used Google Sheets. Students can choose seven preference items and assign values to them (between 1-7).

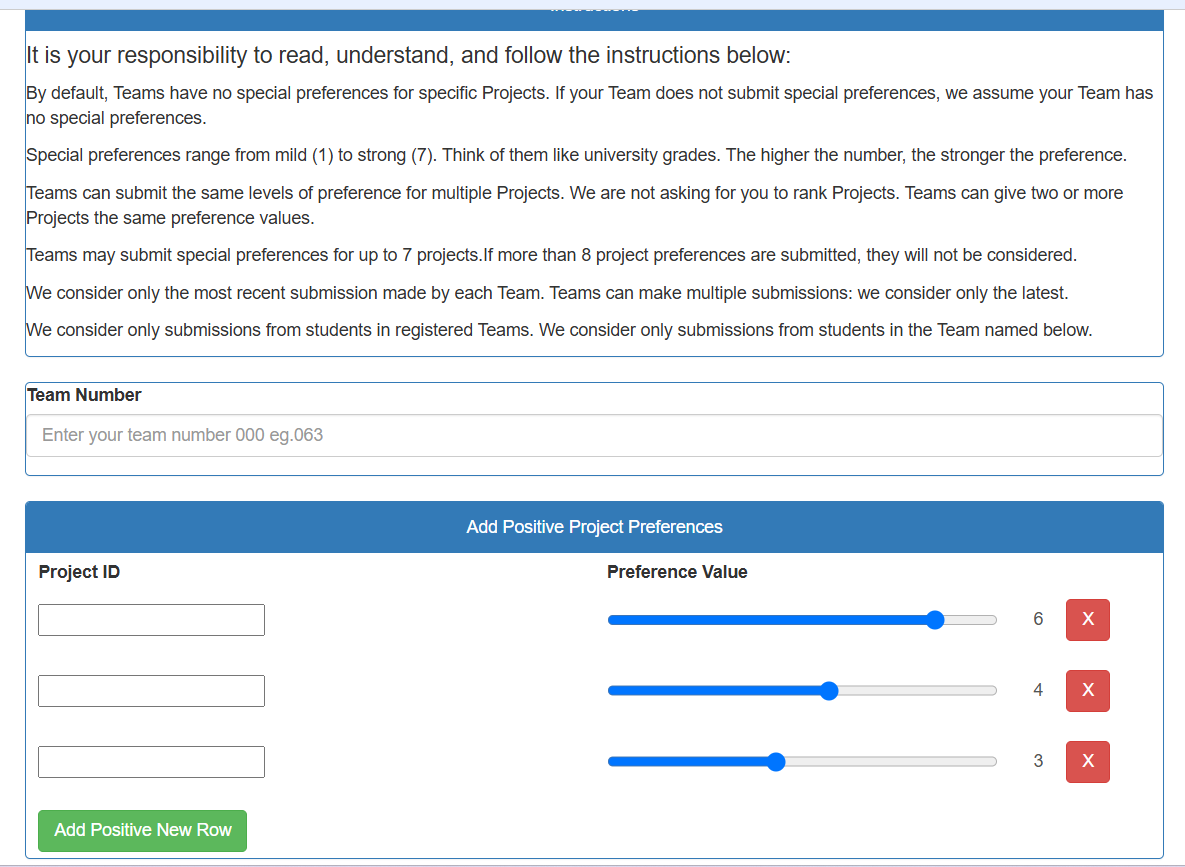
1. Quick start/guide for users
   1. Read introduction and click Continue



b.Fill in all information and check the checkbox,then click submit button



c.enter team number and enter all positive preference numbers while selecting preference values



1. System Installation

For the web app:

* + Ensure access to Google Apps and the necessary permissions for interacting with Google Sheets.

1. Using the Software
   1. Positive Preferences: Enter up to 7 project preferences.
   2. Each submission is saved into a Google Sheet, with the most recent submission overwriting older data for the same team.
2. User Troubleshooting
   1. Form Errors: Ensure all fields are completed before submitting. Missing information will trigger an error message.
   2. Submission Issues: If submissions fail, ensure your internet connection is active and you have the correct permissions for accessing Google Sheets.