1.   Problem Mission Statement

Team of 5 will develop a mobile application that enables Singaporeans to rate their experiences towards Singapore institution's Courses ranging from Pre-Schools all the way to university education levels. This project will be considered complete when the mobile applications have been considered sufficiently tested and approved for release by the Customer. This project aims to support Singaporean families who are looking to enroll their children into Singapore Educational Institutions but do not have the necessary knowledge or experience about them.

2.   Functional Requirements

Source: https://www.nuclino.com/articles/functional-requirements

User Stories: As a non-existing user, I want to be able to create an account.
Functional Requirements: ● The system must allow users to create a new account by entering their email address and desired password. ● The system must verify their password by asking them to key it in again.

User Stories: As an existing user, I want to be able to log into my account.
Functional Requirements: ● The system must allow users to enter their account by entering their email address and password. ● The system must allow users to reset their password by clicking on the "forgot my password" button and receiving a link (via email address tied to the account) to reset their passwords.

User Stories: As an existing user, I want to be able to filter what I am viewing.
Functional Requirements: ● The system must allow users to filter their desired results based on education levels, specific institutions, and course type. ● By default, the system is set to display all available schools and courses onto the screen [via alphabetical order].

User Stories: As an existing user, I want to be able to view available courses in the application based on my filtered selection.

Functional Requirements:
- The system must display that filtered list onto the screen for the user to see.
- By default, the system must display an unfiltered list onto the screen for the user to see [via alphabetical order].

User Stories: As an existing user, I want to be able to prioritize what I am viewing based on the proximity to my home address.

Functional Requirements:
- The system must allow users to key in their home address.
- The system will sort the institutions based on how close your home address is to them.
- Google maps will be used to handle the distance calculation.
- Starting with the closest at the top and the furthest at the bottom.
- By default, the system is set not to display via proximity.

Course Specific

User Stories: As an existing user, I want to be able to leave a review on a polytechnic course.
Functional Requirements: • The system must allow users to type in their comments in a text box. • The text box accepts any characters and any length of them. (Ignoring any security concerns) • The comment will then be posted onto the course page and any other existing users will be able to see it.

User Stories: As an existing user, I want to be able to leave a rating on a polytechnic course.
Functional Requirements: • The system must allow users to leave a rating on a polytechnic course. (1-5 rating) • The system will then update the course with the given rating. • If multiple ratings are given by multiple existing users, the total rating will be summed up and averaged. • The system will then display the ratings on the polytechnic course page.

School Specific

<School> = {Preschool, Primary School, Secondary School, Junior College}

User Stories: As an existing user, I want to be able to leave a review on a <School>.
Functional Requirements: • The system must allow users to type in their comments in a text box. • The text box accepts any characters and any length of them. (Ignoring any security concerns) • The comment will then be posted onto the <School> page and any other existing users will be able to see it.

User Stories: As an existing user, I want to be able to leave a rating on a <School>.
Functional Requirements: • The system must allow users to leave a rating on a <School>. (1-5 rating) • The system will then update the <School>with the given rating. • If multiple ratings are given by multiple existing users, the total rating will be summed up and averaged. • The system will then display the ratings on the <School> page.

3. Non-Functional Requirements

Usability	● Language of the system will change depending on what the user selects. Multi-Language support. ● UI layouts will be done with accordance to UX rules (Golden Ratio, 1/3s Rule, etc.) to be simple and easy to use.
Security	● Existing users must log into their account (email and password) to use the application. ● Several Incorrect attempts will result in the application preventing the current user from trying again for several minutes. ● Password Requirements – Several Characters, Upper-case, Lower-case, Special Characters are required.
Performance	● Response time will be kept to a minimum. (Loading times, Application open, refresh times, etc.) ● Processing time will be kept to a minimum. (Calculation, Functions, etc.)
Recoverability	● Backups will be kept onto the cloud. ● Backups will be done frequently.
Scalability	● The application will be able to handle many different institutions, not just from Singapore. ● It can be expanded into other institutions in other countries.