

SUNY Korea Community: Software Requirements Specification

Team Members:

Sungwoong Park

Gyuri Kim

Table of Contents

1. Introduction
 - 1.1 Product description
 - 1.2 Scope
 - 1.3 Users
 - 1.4 User feedback
 - 1.5 Existing alternatives
 - 1.6 Definitions
 - 1.7 References
2. Requirements
 - 2.1 Functional Requirements
 - 2.2 Use cases
 - 2.2.1. Use Case: Edit timetable
 - 2.2.2. Use Case: Edit password
 - 2.2.3. Use Case: Add courses
 - 2.2.4. Use Case: Write a review
 - 2.3 User Interfaces
 - 2.4 Non-functional Requirements
- 3. Contributions**

1. Introduction

1.1. Product description

Many new students have a problem finding information about their colleges including SUNY Korea. After being a SUNY Korea student for two years, I felt the absence of the school community among SUNY Korea students, because of this, I was not able to know which courses by which professors are better. This inconvenience disturbed me to choose classes to take.

SUNY Korea Community(Not finalized name) is a web application designed to help communication between SUNY Korea students, and manage their timetable.

- Ability to create timetable and add courses by their course number
- Ability to view and create the review of professors
- Ability to share the timetable with their friends
-

1.2. Scope

This product is to support students to make decisions for which courses they should take. Product will not suggest any information or opinions to users directly. Our product is to give only the public opinion. Our data collection will not be connected to the SBU system to get a grade they got, but all information should be given when they are volunteering.

The product will be running on Chrome and a range of devices will be smartphone(Android and iOS), and desktop(Chrome).

1.3. Users

The product's primary users are SUNY Korea students including SBU and FIT, who want to get information about professors and courses, and use and share their timetable. Since our users are college students, we are expecting them to be familiar with the common user interface.

Currently, we are limiting our target to SUNY Korea because of our workload, but various users will be using it.

1.4. User feedback

We got feedback from professors. These are:

1. Indication where the users are very weak.
2. The input of reviews are not explicitly explained.
3. There are possible factors that we can monetize.

For these feedbacks, we noticed that it was the fault of our mockup process. We tried to think and make it look perfect, but we never thought in the way of the new users. Therefore, the change of the mockup was inevitable. For the first feedback, we changed our mockup to indicate what page they are in. Secondly, we changed the text explanation to be more descriptive on “Write review” page. For the third one, we are trying to think whether the monetization could be a moral hazard, but it is definite that we need a business model.

1.5 Existing Alternative

There are existing alternative apps which are Everytime, RatemyProfessor, and Campus-pick. The Everytime application handles the timetable and community service, and the RateMyProfessor website shows reviews of the professors. However, when we use timetable service in Everytime, we have to type every single class name, classroom, and instructor manually. Also, there are no reviews of professors from SUNY Korea in the RateMyProfessor. Therefore, we try to make a website that don't need to type all the information about the class to add timetable. Also, we are going to provide professor review for SUNY Korea students.

1.6 Definition

No special word.

1.7 References

https://www.figma.com/file/gYfUOC8VJASMSwOnGk7pZf/CSE416_Prototype?node-id=0%3A1&t=AfgzczaCEki2VnD3-1

<https://phdkim.net/>

<https://everytime.kr/>

<https://www.kakaocorp.com/page/service/service/KakaoTalk?lang=en>

<https://www.ratemyprofessors.com/>

2. Requirements

2.1 Functional Requirements

Common Interface

- Must Level
 - Function to “Register”
 - Function to “Log in”
 - Function to “Log out”
 - Function to “View the review of professors”
- Should Level
 - Function to “Edit its profile”
 - Editing user Id or email should require more authentication process.
 - This includes almost every information of its information such as email, id, pwd, and email

Common Users

- Must Level
 - Function to “add courses to their table”
 - This automatically adds the designated chatroom
 - Function to “edit or remove courses on their timetable”
 - This automatically edit or remove the designated chatroom
 - Function to “leave and remove(edit) a review for professors”
 - Function to “message and view of the chatroom”
- Should Level
 - Function to “add friends by userid”
 - Function to “view the friends’ timetables”
 - Function to “find the course by its information”

Faculty Moderator

- Must Level
 - Function to “request to remove the reviews”
 - This is limited to only when students leave an insulting message which is not beneficial to the community.
 - Too many frequent and unreasonable requests will cause a ban from this system, but they should be able to contact us with inconvenient way.
 - Function to “modify its information”
 - This includes the office hour and office room number of professors

Administrator

- Must Level
 - Function to “remove the reviews”
 - Function to “ban users for short period or permanently”
 - Function to “add courses to the database”
 - Function to “moderate the chat rooms” such as muting
 - This function will be given to Student Moderator in further development.
- Should Level
 - Function to “assign common user as Student Moderator”

2.2 Use Cases

2.2.1.

Use Case	Edit timetable
Primary Actor	Common User
Priority	Essential
Scenario	<ol style="list-style-type: none">1. User try edit(add/delete) timetable.2. User add course(s) using course ID.3. System checks course ID is valid and provides alert.
Extensions	If the course ID is invalid or does not exist, provide an alert.

Use Case	Edit Profile
Primary Actor	Common User
Priority:	Essential
Scenario:	<ol style="list-style-type: none">1. User edit password.2. Users save the password.3. System checks if the password is valid or not and if it's valid, provides an alert.
Extensions:	If the password is invalid form, provide an alert to fix the password in the right form.

Use Case	Add courses
Primary Actor	Common Users
Priority	Essential
Scenario	<ol style="list-style-type: none"> 1. User logs in the website 2. User is redirected to the timetable website. 3. User press + button. 4. User types the course number and press add course. 5. The system shows the changed time table.
Extensions	4-1. If the course number doesn't exist, it will not accept and give feedback to the user that course number doesn't exist.

Use Case	Write a review
Primary Actor	Common User
Priority	Essential
Scenario	<ol style="list-style-type: none"> 1. User logs in the website. 2. User presses "Reviews" to go to the review page. 3. User find professor by major and name 4. User presses "write a review" 5. User fills the information requested. 6. User presses the submit button, and the system will redirect to the review page of the professor.
Extensions	5-1. On the recommended level, it will automatically block bad words with some language filters.

2.3 User Interfaces

<First draft>

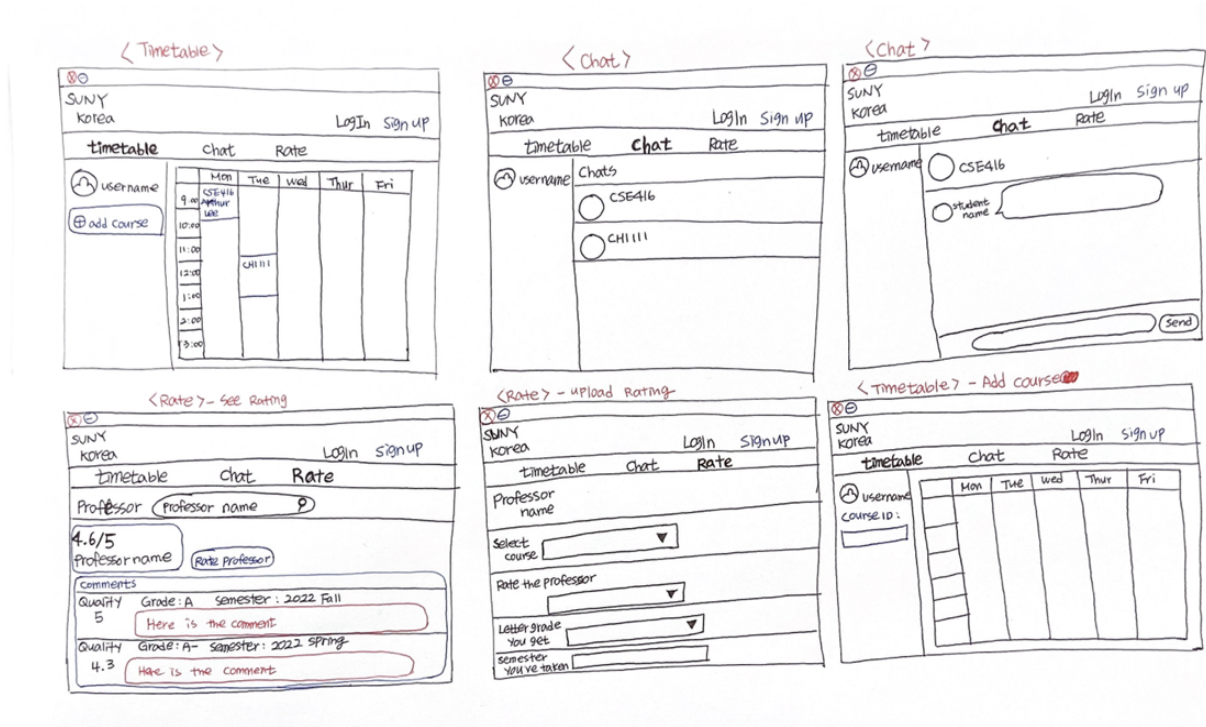




Figure1: Main Page

The image shows a digital mockup of the main page of the SUNY Korea web application. The page has a yellow header with the "SUNY Korea" logo. Below the header is a large white area containing a "Log-In" form. The form has fields for "Email:" and "Password:", and buttons for "Log-In" and "Sign-up". A link for "Forgot your password?" is also present.



SUNY
Korea

Sign-Up

Name:

Student ID:

Email:

Password:

Register

[illegible]

Figure 4: Add Course Page

The screenshot shows a web application interface for SUNY Korea. At the top, there is a navigation bar with the SUNY Korea logo on the left, and 'Username' with a profile icon on the right. Below the navigation bar, there are three main sections: 'Chat' on the left, 'Timetable' in the center, and 'Prof Review' on the right. The 'Chat' section contains a list of chat groups, including 'CSE353 Group' and 'CSE416 Group'. The 'Timetable' section displays a grid with days of the week (Mon, Tue, wed, Thur, Fri, Sat, Sun) as columns. A modal window titled 'Add Course' is overlaid on the timetable, featuring a 'Course ID:' label, a text input field, and an 'Enter' button.

SUNY Korea Username

Chat **Timetable** **Prof Review**

Add Course

Course ID:

Enter

Figure 5: Chatting Page

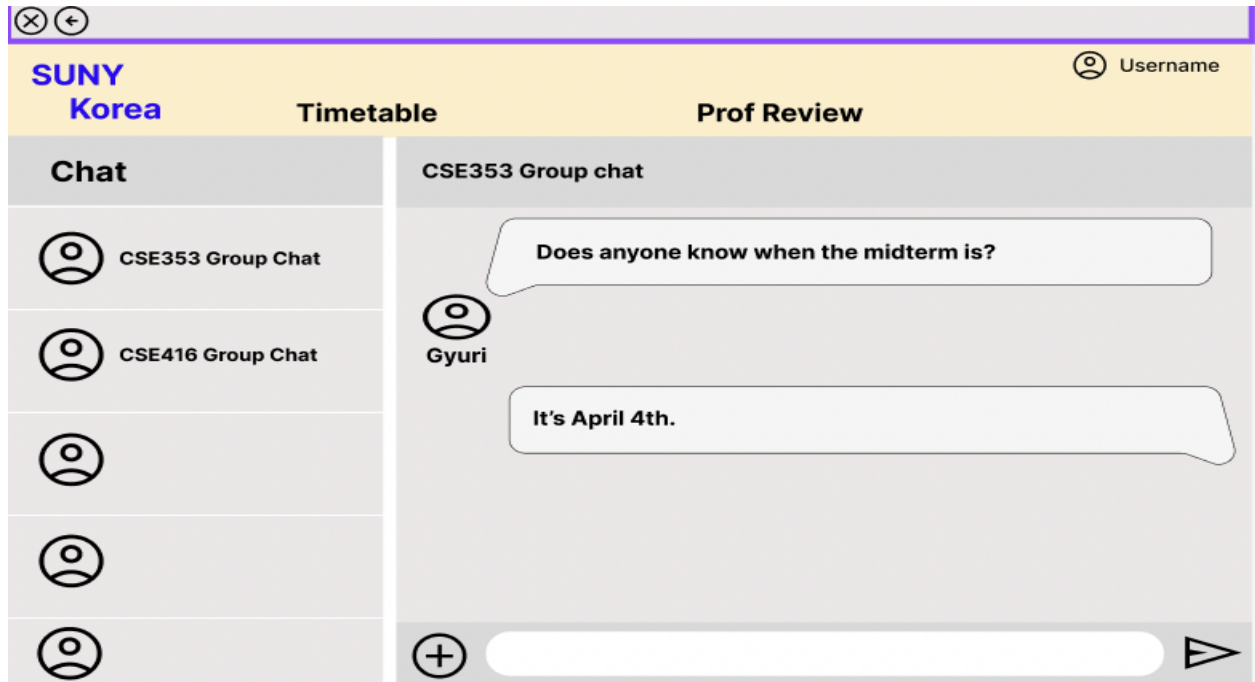


Figure 6: Edit Profile Page

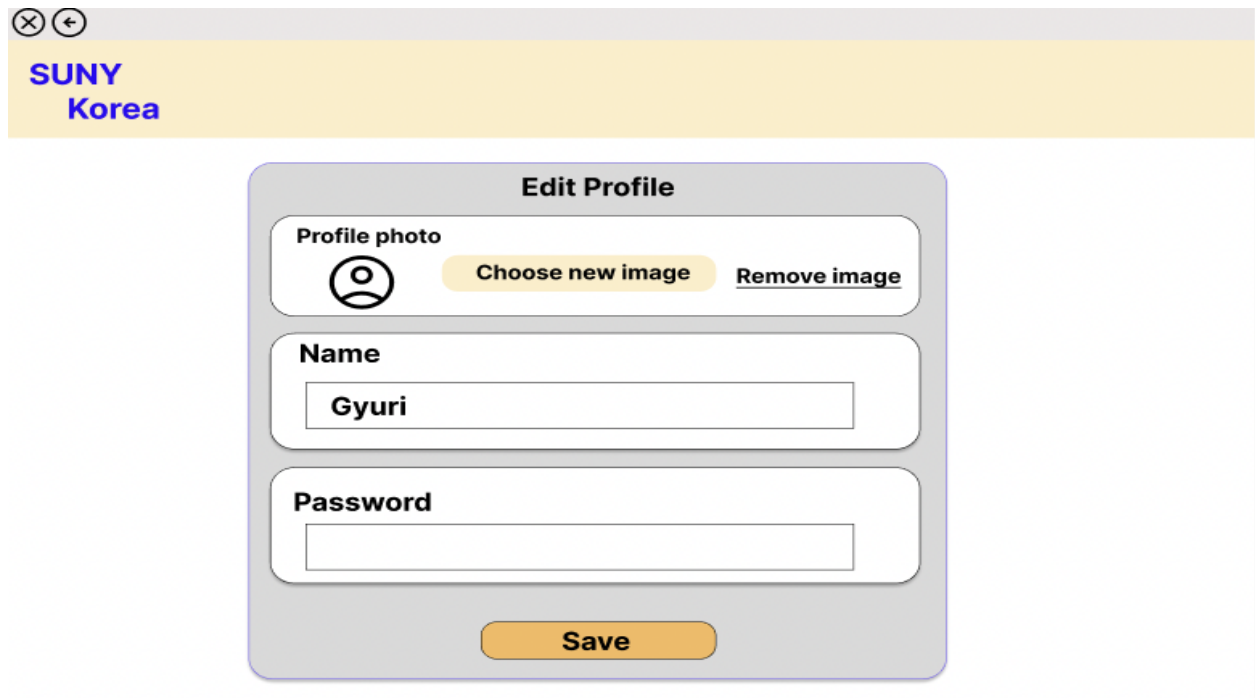


Figure 7: Prof Review Page

ⓧ

⬅

SUNY Korea

Timetable

Prof Review

Username

Professor

Course

Professor name

Recent Review

CSE

AMS

MEC

Course Name: 101 - Computer Science principles

Instructor: Francois Rameau

5

He is a great professor I've ever met

Course Name: AMS310 - Survey of Probability and Stat

Instructor: George

1

I withdraw this class because of horrible lecture

Course Name: CSE214 - Data structure

Instructor: Byungkon Kang

4

He is so good at teaching but the material in this class is so hard.

Course Name: MEC101 - Frashman Design Innovation

Instructor: Cornelius Bradter

3

No comment

Figure 7.1:

ⓧ

⬅

SUNY Korea

Timetable

Prof Review

Username

Professor

CSE

Arthur Lee

4.6/5

Arthur Lee

Rate Professor

Comments

Quality

Grade: A

Semester: 2022 Fall

5

Here is the comment

Quality

Grade: A-

Semester: 2022 Spring

4

Quality

Grade: C-

Semester: 2021 Fall

1

Figure 8: Rate Professor Page

ⓧ

⬅

SUNY
Korea

Timetable

Prof Review

Ⓔ

Username

Arthur Lee

Select
Course

▼

Rate
the professor

▼

Letter grade
you get

▼

Semester you've
taken

▼

Comment

2.4. Non-functional Requirements

1. Users should be able to complete use cases and perform all function requirements in the web application via Chrome.
2. All users' private data should be obtainable and accessible only when necessary.
3. On this web application, all communication should be secured with SSL.
4. This web application should handle at least 200 connections and secure the average load time in 1 second.
5. The web application will be given in English.

3. Contributions

Gyuri Kim: Design brain-storming, Figma Disign, Discuss software requirements specification, write the document.

Sungwoong Park: Design brain-storming, Discuss software requirements specification, write the document, Presentation.