Service Name: You & Meet

Members:

Taebum Kim <StudentId1> phya.ktaebum@gmail.com
Dongsu Zhang <StudentId2> 96lives@gmail.com
Philsik Chang <StudentId3> lasagnaphil@snu.ac.kr
Minwoo Jeong <StudentId4> jumen02@snu.ac.kr

Target Customers: For everyone who meet often with their friends or acquaintances

Overall Description: Schedule management web service

Description:

Nowadays, people use messengers to make an appointment(KakaoTalk, Line, etc). However, there are some inconveniences with this method. Generally, a majority rule is used to determine the appointment time or place. If the time available for each person do not converge to a single point, they need additional conversations, or even re-vote. This is very inconvenient and time-consuming. Without using votes, they usually make an appointment by comparing each person's calendar or timetable. However, this method falls short if there are many people. Hence, the purpose of our service is to manage their schedules efficiently and give the best appointment time for them. In addition, we will add other features, such as dutch payment, suggestion for the best appointment placement, and timetable uploading.

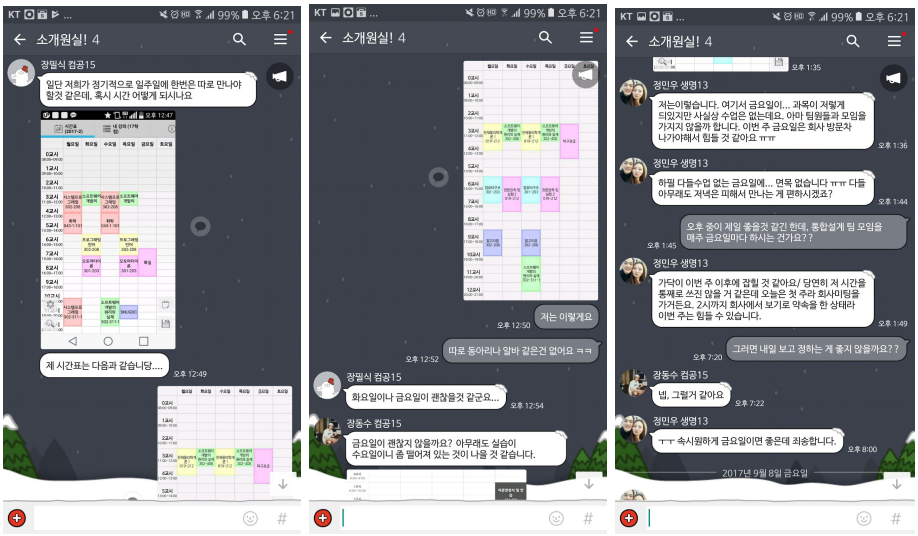

Figure 1. Example of making an appointment via messenger

Essential Functions:

1. Checking available time interval.
2. Get calendar data from other calendar api(Google Calendar, iCalendar, etc…)
3. Set priority when checking available time.
4. If the service cannot find the most optimized solution (time that everyone could attend), suggests the second best plan and so on.

Optional Functions:

1. *Dutch payment*: Get data of who paid partially, and from total number of people in group, calculate how much each person should pay.
2. *Uploading time tables*: Students who have pictures of their time table can upload their timetables. Our service will interpret the pictures and automatically mark unavailable time
3. *Location*: Determining the best place to meet

Instructions for Using the Service:

1. The person who wants to make an appointment goes to www.younmeet.com.
2. If one clicks 'New Schedule', one will obtain a shareable link.
3. Set password for the appointment (Not mandatory. If one doesn't, the site will alert the user with the warning message: "*Other people who are not related to your group could open your appointment plan*".)
4. Send the link to friends or coworkers via messenger.
5. If other people open the link, the browser will show a calendar UI which they can mark the time they are available.
    a. If they want to use the schedule stored in their You & Meet account, which is linked with other schedule services(such as Google, iCalendar, etc), they can login.
    b. Also, without creating an account, they can mark their available time manually.
6. Based on everyone's schedules, the website will suggest the best time to meet.

Demo

Assume that four people have to schedule an appointment. Show that the old way of checking each other's schedule via messengers (such as KakaoTalk) is inefficient. Then show that our web service can handle this in a much easier way.

1. Show that our web service's user interface is intuitive and pleasing to look at, so that people could manage their appointment schedule conveniently.
2. Show that our web service can be integrated easily with other accounts such as Google / Apple schedule services. Also show that people can join in an appointment without making a You & Meet account.

Test

The web service will have two components: an "Account" service which manages all the You & Meet accounts, and a "Appointment" service which manages the calculation of the best schedule for an appointment.

1. *Account service*: Make an automated tool which creates multiple mock accounts. Then we can stress test the server and calculate how many accounts it will withstand.
2. *Appointment service***:** Make an automated tool which creates multiple mock schedules for each account. Then test correctness of our services.
3. *Integration test***:** Make sure the two services are integrated well. Pick random accounts from the Account service and see if the schedules from the accounts are loaded properly into the Appointment service.