

Assignment 1

Project description: School schedule website with comparisons with other schedules to see where people have free time at the same time.

Vision and Scope

1.2. Why does it look like we can be successful?

This is something that we think does not exist, and could be very useful, especially for project teams to find time to work on the project where everyone can show up. It will also be useful to friends in different majors to find time to meet up for lunch or other activities. All sorts of additional features could be added (personal schedule, to-do list, import from Uglu, homework diary etc) to make this an all around study organization tool.

1.5. What will the product accomplish for whom?

For students

Who want to organize their time better

The Schedule Syncer (working title)

Is a scheduling tool

That enables students to sync up schedules with others, and plan their studies

Unlike Uglu which does not have any syncing/personal planning functionality

Our product will accomplish this.

2.1. Major features

- Student creates an account using email and password.
- Student logs in using email and password.
- Student logs in, opens schedule creator and creates his school schedule.
- Student logs in, finds another student's account and syncs their schedules to see where they have common gaps in their schedule.
- Student logs in, searches for study partners and adds them to a group.

2.3 What can be rolled out later?

If we have time, additional features could be added, like

- Personal planner (doctor's appointments etc)
- Homework planner
- To-do list
- Other scheduling features

Fully-dressed use cases

First use case

1. Use case name - Create account and schedule
4. Primary actor - Student
6. Preconditions - Nothing
7. Success guarantee - Student has created an account and put together a school schedule
8. Main success scenario
 1. Student open website and chooses option to create account
 2. Student inputs info about himself (incl. username/email and passw.)
 3. Student receives a confirmation email and follows a link there to validate his account
 4. Student uses new account to log in
 5. Student opens schedule creator
 6. Student creates schedule by entering one class at a time
 7. Student selects recurring attribute for each class (so it occurs every week until semester ends)
 8. Student saves schedule
 9. Student views weekly schedule on homepage
9. Alternate scenarios
 - (2) Student inputs a too-short password while creating account and is requested to enter a valid one
 - (4) Student immediately forgets the password he chose and has to request a new one (which arrives by email)
 - (6) Student accidentally inputs wrong time for a class and deletes it/moves it to correct time
 - (8) Student accidentally navigates away from the schedule creator. When he gets back, the schedule is autosaved.
13. Open issues
 - What happens if the browser or device crashes (or device loses power)?
 - Can the student import schedules from other programs(ugla, ical, google calendar, etc)?
 - Do we want to autosave the schedule permanently or so that it expires after some time?

Second use case

1. Use case name - Find and add friend, and compare schedules
4. Primary actor - Student
6. Preconditions - Student has created an account and a schedule
7. Success guarantee - Student has found and added a friend and compared their schedules
8. Main success scenario
 1. Student logs in
 2. Student finds search bar and searches for his friend by name
 3. Student adds his friend to his friend list
 4. Student chooses to compare his schedule with the friend's
 5. Student views the compared schedule and sees where both have free time at the same time

6. Student creates a meeting item in his schedule in one of the free slots
 7. Student sends an invitation to his friend to add the meeting item to his schedule as well
9. Alternate scenarios
- (2) Student can not find his friend and has to try to search for his email instead of name
 - (4) His friend has not yet created a schedule so there is nothing to compare with
 - (5) There are no slots where both have free time.
 - (7) His friend declines the invitation as he has an unregistered dentist appointment in the free slot
13. Open issues
- Should a student only be able to compare schedules with people he has added as a friend? Should people be allowed to choose whether their schedule is open or not?
 - How do we want to show the compared schedules? Do we want people to see other people's whole schedule or just where they are occupied?
 - Should people have to accept a friend request or can anyone add anyone? Do we want the friend system to be follow-based.

Brief use cases

- Student logs in, opens schedule creator and adds in a one-time item for a doctor's appointment.
- Student logs in, opens schedule creator and chooses different colors for each type of item to make the schedule easier to read.
- Student receives a notification that another user wants to invite him to a meeting. He accepts the invitation and subsequently the item is added to his schedule as well.
- Student logs in and chooses to create a group. He selects his HBV501G teammates and gives the group a name. Next he compares the schedules of all group members and sees that they have one free time slot. He then schedules a weekly team meeting in that slot and sends invitations to all group members.
- Student is added to a group by another user, but sees that he was added as a mistake instead of another user with a similar name. He chooses to leave the group.
- Student opens schedule creator and chooses to create a label "school". He then edits all school-related items and adds the label to them. He then saves the the schedule.
- Student views his schedule and uses his pre-defined filters to view only those items that fit the label filter (e.g. school items, work items, personal items etc).