

Software requirements specification

Sport@Unibe - Team 2

Table of Content

[Table of Content](#)

[Introduction](#)

[Purpose](#)

[Stakeholders](#)

[Definitions](#)

[System overview](#)

[References](#)

[Overall description](#)

[Actor characteristics](#)

[Specific requirements](#)

[Functional requirements](#)

[Non-functional requirements \(external, performance, etc.\)](#)

[Product requirements:](#)

[Organisational requirements:](#)

[External requirements:](#)

[Project: Sport@Unibe](#)

[Customer: Andrea](#)

[Group: 2](#)

[Diagram](#)

[See all courses](#)

[Actors](#)

[Description](#)

[Trigger](#)

[Pre-conditions](#)

[Post-conditions](#)

[Main scenario](#)

[Alternative scenarios](#)

[Special requirements](#)

[Notes](#)

[Show details of a certain course](#)

[Actors](#)

[Description](#)

[Trigger](#)

[Pre-conditions](#)

[Post-conditions](#)

[Main scenario](#)

[Alternative scenarios](#)

[Special requirements](#)

[Notes](#)

[Actors](#)

[Description](#)

[Trigger](#)

[Pre-conditions](#)

[Post-conditions](#)

[Main scenario](#)

[Alternative scenarios](#)

[Special requirements](#)

[Notes](#)

[See all subscribed courses](#)

[Actors](#)

[Description](#)

[Trigger](#)

[Pre-conditions](#)

[Post-conditions](#)

[Main scenario](#)

[Alternative scenarios](#)

[Special requirements](#)

[Notes](#)

[Access settings to subscribed course](#)

[Actors](#)

[Description](#)

[Trigger](#)

[Pre-conditions](#)

[Post-conditions](#)

[Main scenario](#)

[Alternative scenarios](#)

[Special requirements](#)

[Notes](#)

[Show place where the course is held](#)

[Actors](#)

[Description](#)

[Trigger](#)

[Pre-conditions](#)

[Post-conditions](#)

[Main scenario](#)

[Alternative scenarios](#)

[Special requirements](#)

[Notes](#)

[Search courses](#)

[Actors](#)

[Description](#)

[Trigger](#)

[Pre-conditions](#)

[Post-conditions](#)

[Main scenario](#)

[Alternative scenarios](#)

[Special requirements](#)

[Notes](#)

Rate courses

Actors

Description

Trigger

Pre-conditions

Post-conditions

Main scenario

Alternative scenarios

Special requirements

Notes

See rating of courses

Actors

Description

Trigger

Pre-conditions

Post-conditions

Main scenario

Alternative scenarios

Special requirements

Notes

Broadcast subscribed courses

Actors

Description

Trigger

Pre-conditions

Post-conditions

Main scenario

Alternative scenarios

Special requirements

Notes

Attendance monitoring

Actors

Description

Trigger

Pre-conditions

Post-conditions

Main scenario

Alternative scenarios

Special requirements

Notes

Introduction

Purpose

The purpose of this application is that the users can get the information about the sport activities at the University of Berne easily, through an android gadget. This application should be very easy to use, so that the users have no problem to get to what they are looking for. There will be many features for them in this application, that the users can make their personal settings or even share information with other people.

This indicates, that the users can see the list of different sports with schedules in one go. If necessary they can make a separate list with their own favourite activities and schedules. This application should also be able to offer them the location and direction of the selected sport. They can have an option to search certain activities with a specific criteria. Users should be able to make a reminder or even subscribe to a chosen sport at a specific time and automate the sign in process. Every sport can be rated by them and they should also be able to see the ratings of other people. The users should also know and be notified, if their friends are also participating in the same sport.

Stakeholders

In this project our internal stakeholder is the customer, Andrea Caracciolo, who has given us an assignment to program an android application of the University Sports Berne website.

Also we, the developers are internal stakeholders, as we want to create a successful product, which gets used by as many users as possible.

We also have external stakeholder like the students, who are the users of this application. They do not have anything to do with the project directly, but can affect its actions or can be affected by it.

Definitions

User: Students

Customer: Andrea Caracciolo

System overview

We will update this chapter with the results of the next exercise.

References

- Ian Sommerville, Software Engineering 2011
- Slides of the lecture "Introduction Software Engineering" Universität Bern 2013

Overall description

Actor characteristics

Users of the Sport@Unibe App are typically students at the University of Berne. Since there are eight different faculties and many different programs of study the characteristics of someone using the App turn out to be quite varied. As an example, let us point out three possible users:

Andreas, 20, studies Psychology in the first semester. He's not a very sporty person, but he used to enjoy the sports classes when he was still in the Gymnasium. Hearing about the wide sport offered by the University he decides to check out some courses. He downloads the App and searches for team sports and their timetables. There are 23 results. Andreas sets a reminder for the football course on Wednesday noon.

Tania, 24, is a Sports student in her masters course. Sport is not only her field of studies but also her passion and she exercises regularly. She is looking for a possibility to coordinate her timetable. Thus she downloads the App, marks her favourite sports and checks on their timetables. Now she can compare her course schedule with the sports schedules.

Cathrine, 22, studies Physics. She likes doing sports but usually she is too lazy to go to the sports courses offered by the University. That is why she makes an agreement with her fellow student Patrick to do at least one sports course a week. They both download the App and add each other as friends. They do not decide on a specific course but mark the courses they are planning to visit. Through the App they see the information who is planning to go to which course and are able to decide on a course they visit together.

Specific requirements

A functional requirement describes what a software system should do, while non-functional requirements place constraints on how the system will do so.

Functional requirements

- A list of all sports in the campus should be shown.
- The app should enable the user to see course schedules for a selected sport.
- The user should be able to define his favorite courses and see their schedules.
- It must be possible to see the location of a selected course and get directions to get to this place.
- The user should be able to see a list of courses that match with a certain criteria e.g start at a given day/time i.e. it should be possible to search the list of all courses with a certain filter.
- The user should get notifications for upcoming courses e.g. 2h before the course starts.
- There should be a feature in which the user can rate attended courses.
- The user should see how other people have rated a certain course
- The user should be able to let his friends know to which course he has planned to go. If they want to join him, the user should be notified.

Non-functional requirements (external, performance, etc.)

Product requirements:

The app should support all versions of Android between Android 4.0 (Ice Cream Sandwich) and Android 4.2 (Jelly Bean).

Usability: The app should be designed in a way that it is easy understandable and fast to learn for the user on how to operate the app. An attractive design should be sought.

Emotional factors: It should be fun to use the app and the app should encourage the user to do more sports at the University of Berne.

Accessibility: The app should be accessible for all members of the University of Berne.

Backup: There won't be any backup provided. The user will download the app and use it

locally on his Android device. If he deletes the app there will not be a possibility to restore the collected data.

Documentation: The SRS (Software requirements specification) will be kept up-to-date throughout the whole design process.

Extensibility: This app is made during the “Introduction to Software Engineering” class in the fall semester 2013 at the University of Berne. After that, future growth is not intended.

Organisational requirements:

Development environment: How proposed in the lecture, we'll work with Eclipse and SDK (Android)

Used programming language therefore is java.

Operational system: Android devices

External requirements:

Safety: In case the user has to provide personal data, i.e. a log in or something similar is implemented, the user should be able to inspect his personal data and correct errors in the data / change the data.

Project Requirements

Project: Sport@Unibe

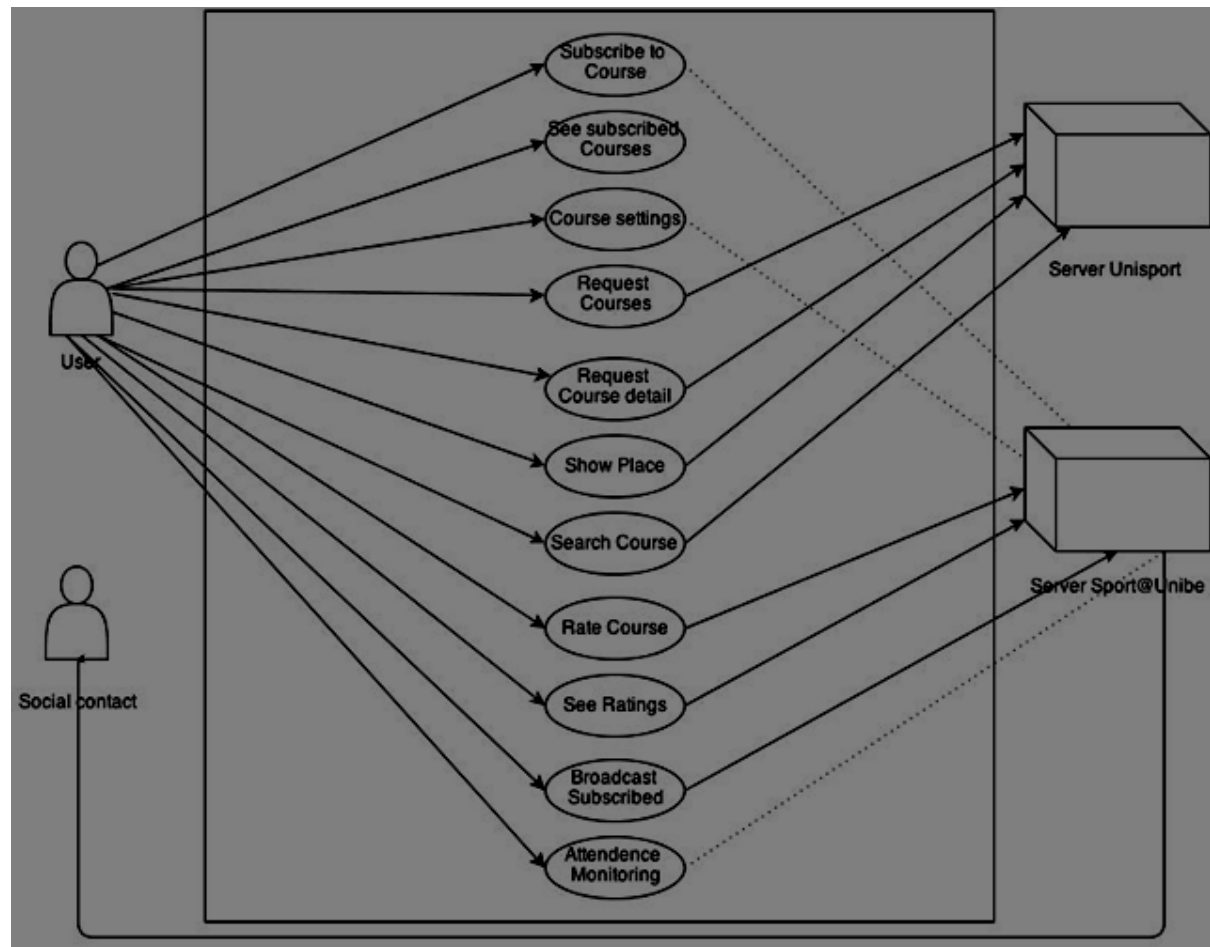
Customer: Andrea

Group: 2

Revision History:

Version	Date	Revision Description
.01	29.09.13	First approach
.02		
...		
1.0		

Diagram



See all courses

Actors

User

Description

- As a user i want to display a list of all available courses

Trigger

- User clicks „All Courses“ tab

Pre-conditions

- User is connected to the internet

Post-conditions

- A list of all courses appears

Main scenario

- User opens app
- User clicks „All Courses“ tab

Alternative scenarios

- User is not connected to the internet
 - Error message appears: no internet connection

Special requirements

- none

Notes

-
-

Show details of a certain course

Actors

User

Description

- As a user i want to display the details of a course by clicking on it. These include Time, Place, Rating and some other information depending on the type of course.

Trigger

- User clicks on a course (in the „All Courses“ tab, inside a search, or in the „Subscribed“ tab)

Pre-conditions

- User is connected to the internet
- At least one course is displayed in the current view

Post-conditions

- A standardized view appears, showing all relevant information on the selected course

Main scenario

- User navigates to either „All Courses“, „Subscribed“ or searches for a course.
- User chooses a course by clicking on it
- Details appear

Alternative scenarios

- User is not connected to the internet
 - Error message appears: no internet connection

Special requirements

- none

Notes

-
-

Subscribe to a course

Actors

- User

Description

- As a user i want to be able to subscribe to a course i want to attend.

Trigger

- User presses the „Subscribe“ button

Pre-conditions

- User is connected to the internet
- User is displaying the details of a course at the moment

Post-conditions

- The selected course is added to the users subscribed courses, meaning it appears in the list seen when clicking the „Subscribed“ tab
- the „Subscribed“ tab is opened

Main scenario

- User navigates to either „All Courses“ or searches for a course.
- User chooses a course by clicking on it
- Details appear
- User clicks on the „Subscribe“ button

Alternative scenarios

- User is not connected to the internet
 - Error message appears: no internet connection

Special requirements

- none

Notes

-
-

See all subscribed courses

Actors

User

Description

- As a user i want to display a list of all my subscribed courses

Trigger

- User clicks „Subscribed“ tab

Pre-conditions

- User is connected to the internet
- User must have at least one subscribed course

Post-conditions

- A list of all subscribed courses appears

Main scenario

- User opens app
- „Subscribed“ is the starting tab

Alternative scenarios

- User is not connected to the internet
 - Error message appears: no internet connection
- User has no subscribed courses
 - A box is displayed, informing the user that there are no subscribed courses available, followed with an instruction on subscribing on courses

Special requirements

- none

Notes

-
-

Access settings to subscribed course

Actors

User

Description

- As a user i want to have additional options for subscribed courses. These include Notifications for upcoming courses, broadcasting my subscribed courses to friends, assistance with courses that i have to sign up for at a exact time etc.

Trigger

- User clicks „Options“ inside the detail view of a course

Pre-conditions

- User is connected to the internet
- User has subscribed to the course

Post-conditions

- A options menu for the selected course appears

Main scenario

- User opens app
- User clicks „Subscribed“ tab
- User clicks on a subscribed course
- user clicks on the „Options“ button
- A options menu for the selected course appears

Alternative scenarios

- User is not connected to the internet
 - Error message appears: no internet connection
- User has not subscribed to the courses
 - The „Settings“ button is not shown in the details view

Special requirements

- none

Notes

-

Show place where the course is held

Actors

User

Description

- As a user i want to display the place where a course is held

Trigger

- User clicks the map in the details view

Pre-conditions

- User is connected to the internet

Post-conditions

- Google map is opened and shows the address where the course takes place

Main scenario

- User opens app
- User navigates to the details view of course he wants to know the place of
- User clicks on the map in the details view
- The Address is shown in google maps

Alternative scenarios

- User is not connected to the internet
 - Error message appears: no internet connection

Special requirements

- none

Notes

-

Search courses

Actors

User

Description

- As a user i want to search for a course by either name, time or date

Trigger

- User clicks the a „Search“ button in the taskbar

Pre-conditions

- User is connected to the internet

Post-conditions

- A search view appears where the user can choose from the different search criteria and fill in an adapted search request form

Main scenario

- User opens app
- User clicks the „Search“ button
- Serch view opens

Alternative scenarios

- User is not connected to the internet
 - Error message appears: no internet connection

Special requirements

- none

Notes

-
-

Rate courses

Actors

User

Description

- As a user i want to rate a course i have attended

Trigger

- User clicks on a rating scale in the details view of a course

Pre-conditions

- User is connected to the internet

Post-conditions

- The users feedback is sent to the server so it can be seen by other users

Main scenario

- User opens app
- User navigates to a desired courses details view
- User clicks on a rating bar, representing his overall impression on the course
- A rating view opens, where the user can fill in additional information and comments on the course
- User fills in the form and clicks the „Confirm“ button to send the feedback
- The detail view of the course returns

Alternative scenarios

- User is not connected to the internet
 - Error message appears: no internet connection

Special requirements

- none

Notes

- can a user that has not subscribed to a course rate said course?

// irrelevant due to the rating beeing shown in the details

See rating of courses

Actors

User

Description

- As a user i want to display a list of all available courses

Trigger

- User clicks „All Courses“ tab

Pre-conditions

- User is connected to the internet

Post-conditions

- A list of all courses appears

Main scenario

- User opens app
- User clicks „All Courses“ tab

Alternative scenarios

- User is not connected to the internet
 - Error message appears: no internet connection

Special requirements

- none

Notes

-
-

// open for debate

Broadcast subscribed courses

Actors

User

Description

- As a user i want to display a list of all available courses

Trigger

- User clicks „All Courses“ tab

Pre-conditions

- User is connected to the internet

Post-conditions

- A list of all courses appears

Main scenario

- User opens app
- User clicks „All Courses“ tab

Alternative scenarios

- User is not connected to the internet
 - Error message appears: no internet connection

Special requirements

- none

Notes

-
-

Attendance monitoring

Actors

User

Description

- As a user i want to keep note of how often i have attended or missed a course. This box could look like this:



Trigger

- The attendance information box is shown in the details view and can be edited by clicking on the box and then choosing whether the user has attended the the next appointment in the list or not.
- The app could also request the user to fill in if he has attended the course or not via a push notification when the course is over

Pre-conditions

- User is connected to the internet
- User has subscribed to the course

Post-conditions

- The attendance information is up to date

Main scenario

- User opens app
- User navigates to his desired course
- user scrolls down to the attendance box where he can see how frequently he has been attending the course
- the user clicks on the attendance box and selects the proper option when asked if he attended the course or not.

Alternative scenarios

- User is not connected to the internet
 - Error message appears: no internet connection
- The Attendance box is not shown in the details view

Special requirements

- none

Notes

-
-
-