Business Requirements Set

<ESE Project: Platform for students and tutors>

ESE Team 2

Cockpit				
Phase	**Deadline**	**Status**	**Date**	**Person**
Creation	07.10.2015	SRS 0.5 ☒	03.10.2015	M.Wenger [MW]
Accepted		SRS 2.0 ☐		
Dev		--		
Retro		SRS 3.0 ☐		
Finish		--		

Change log

Version	Date	Person	Type of change
0.5	03.10.2015	Mischa Wenger	Create (first, not review ready version)
0.52	05.10.2015	Marc Jost	Extended intro, use cases, assumptions and more
0.53	05.10.2015	Marc Jost	Added further Use Cases
0.54	06.10.2015	Eve Mendoza Quiros	Added Use Cases
0.55	06.10.2015	Cyrill Portmann	Added Use Case Diagrams
0.6	14.10.2015	Cyrill Portmann	Changed Use Case Diagrams
0.7	10.11.2015	Eve Mendoza Quiros	Updated complete SRS

List of distributors / stakeholders

Function	Name [& shortcut]	OE/Firma	Review	Info
Project-Team				
Request Owner	Andrea Caracciolo [AC]		x	
Dev. Team	Cyrill Portmann [CP]		x	
Dev. Team	Eve Mendoza Quiros [EM]		x	
Dev. Team	Marc Jost [MJ]		x	
Dev. Team	Mischa Wenger [MW]		x	
Extended list of distributors:				
				x
				x
				x
				x

Content

1 Introduction

Every year there are new technologies and features developed all over the world. The modern bachelor student of computer science has to deal with all these changes. In this immense jungle of information, it's hard to find out what to learn, how to learn and when to learn.

With the ESE Team 2 Platform, we offer a possibility for students to easily get in touch with qualified tutors who can support the students during the semester and/or during exam preparations. These tutors display their skills by featuring grades on their profile pages. Along with a profile picture, a description of the tutor’s person and the demanded rate a student is able to make an informed decision, whether to contact the tutor or not. The to-be-developed platform offers a simple meeting-arrangement-feature allowing students and tutors to arrange their first lesson through the platform. The website furthermore offers a payment system through a third party provider, through

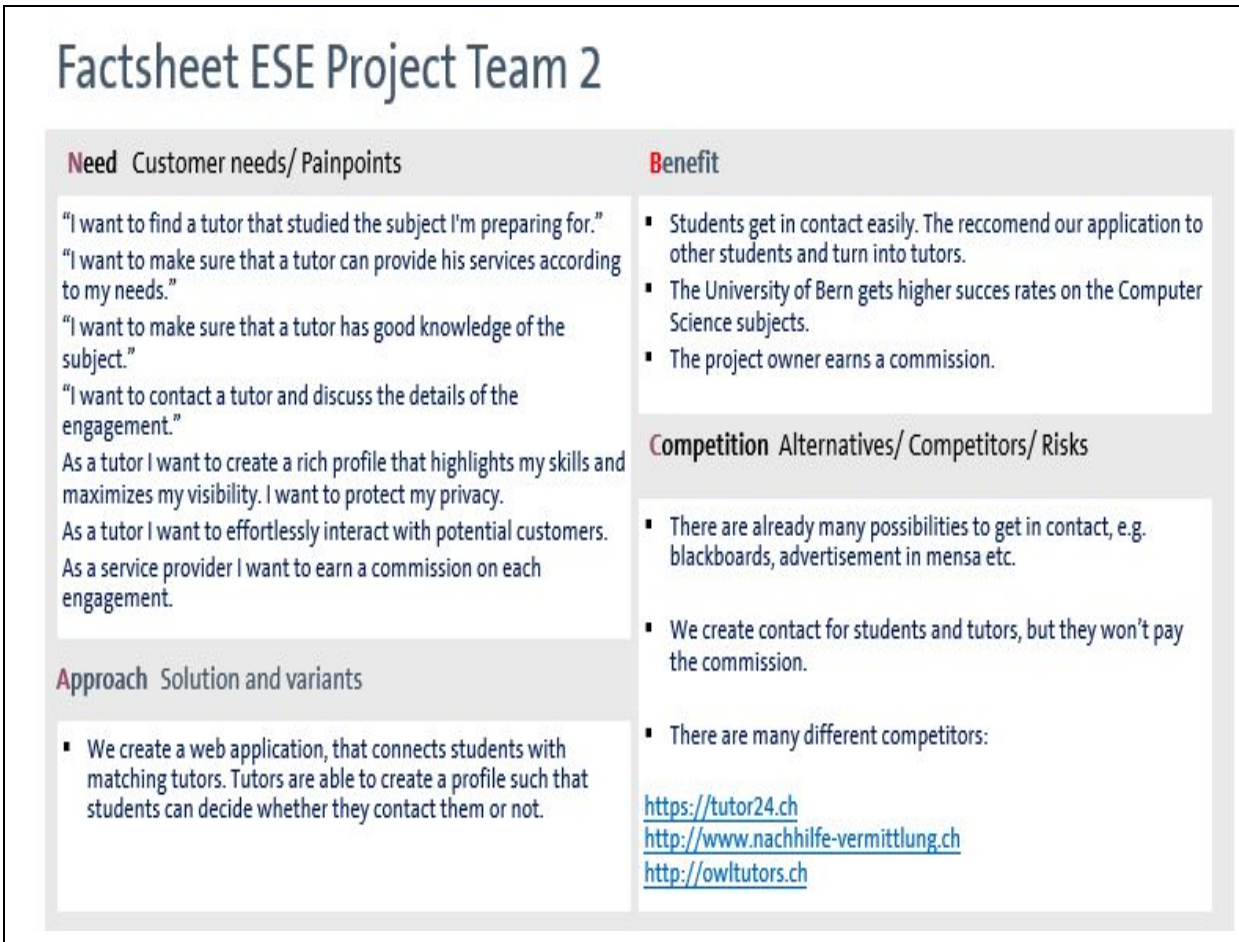
which students can pay their tutors without needing to know detailed banking information for all their tutors. The project owner will collect a commission on every payment, making the platform beneficial for all involved parties.

This document specifies the project order and the software requirements.

2 Pitch for the project (incl. vision)

Hook:

To connect students and tutors we create an interactive platform for students and tutors. Students can easily select their preferred tutor by browsing for the course they require help for. As a tutor, I don't have to create advertisements or to check the blackboards all the time – the students contact me instead. Additionally, the scope of the tutoring is clearly defined in form of a University lecture which doesn't change all too often. As a student, I don't have to browse through all the other generic tutoring websites which cover a too wide area of studies. Instead, I can visit the platform which is tailored to the Bachelor in computer science at the University of Berne. As a student, I can relate to the tutor's past experience by examining his/her grades in the same lecture I am currently taking. Since the tutor already took the exam, a student can also learn about the exam itself, further enhancing his/her chances for success.



Close:

This platform is specialized for bachelor students in computer science of the University of Bern. It is a more focused version of the existing platforms, which convinces with ease of use and professionality.

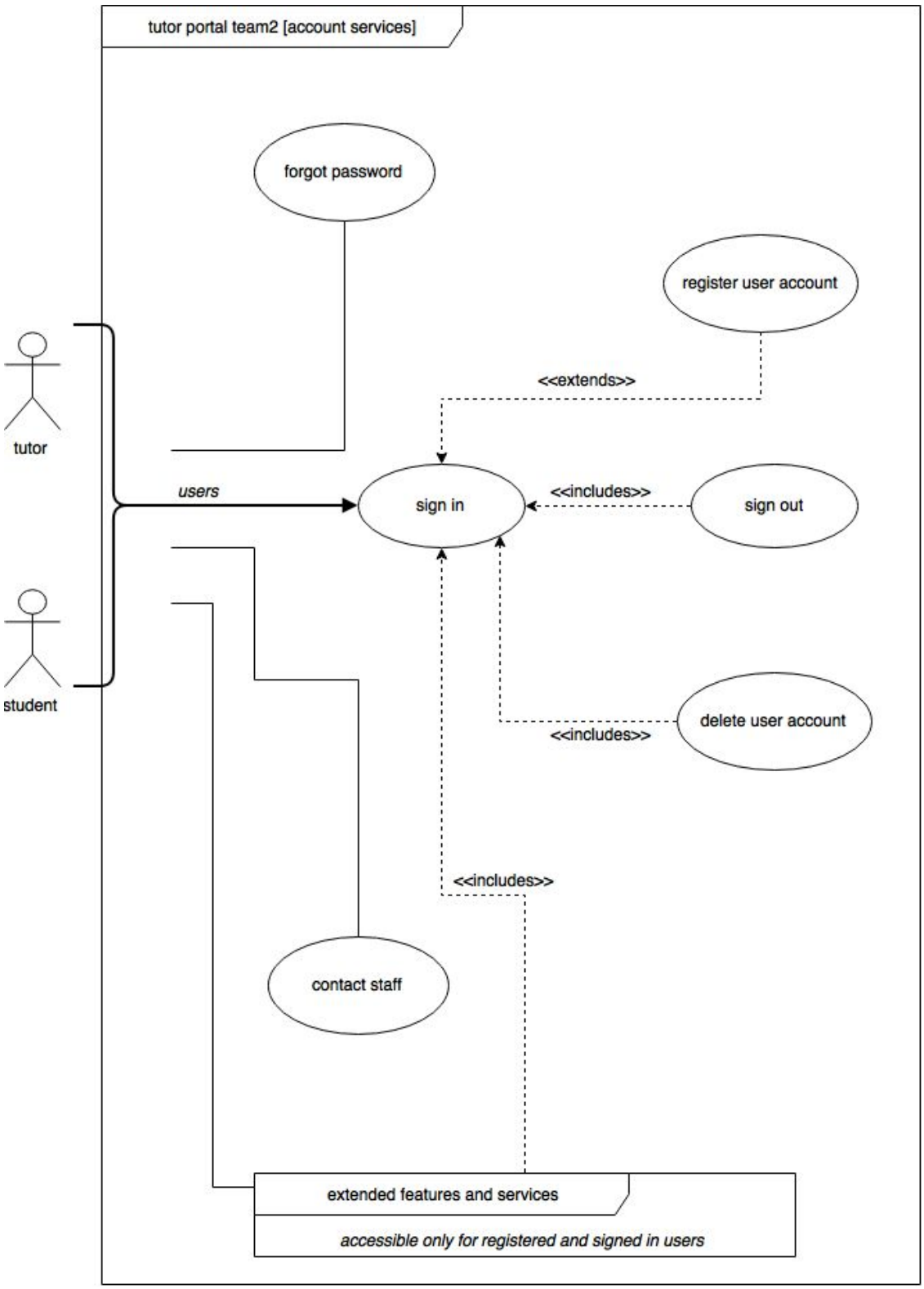
3 Customer needs

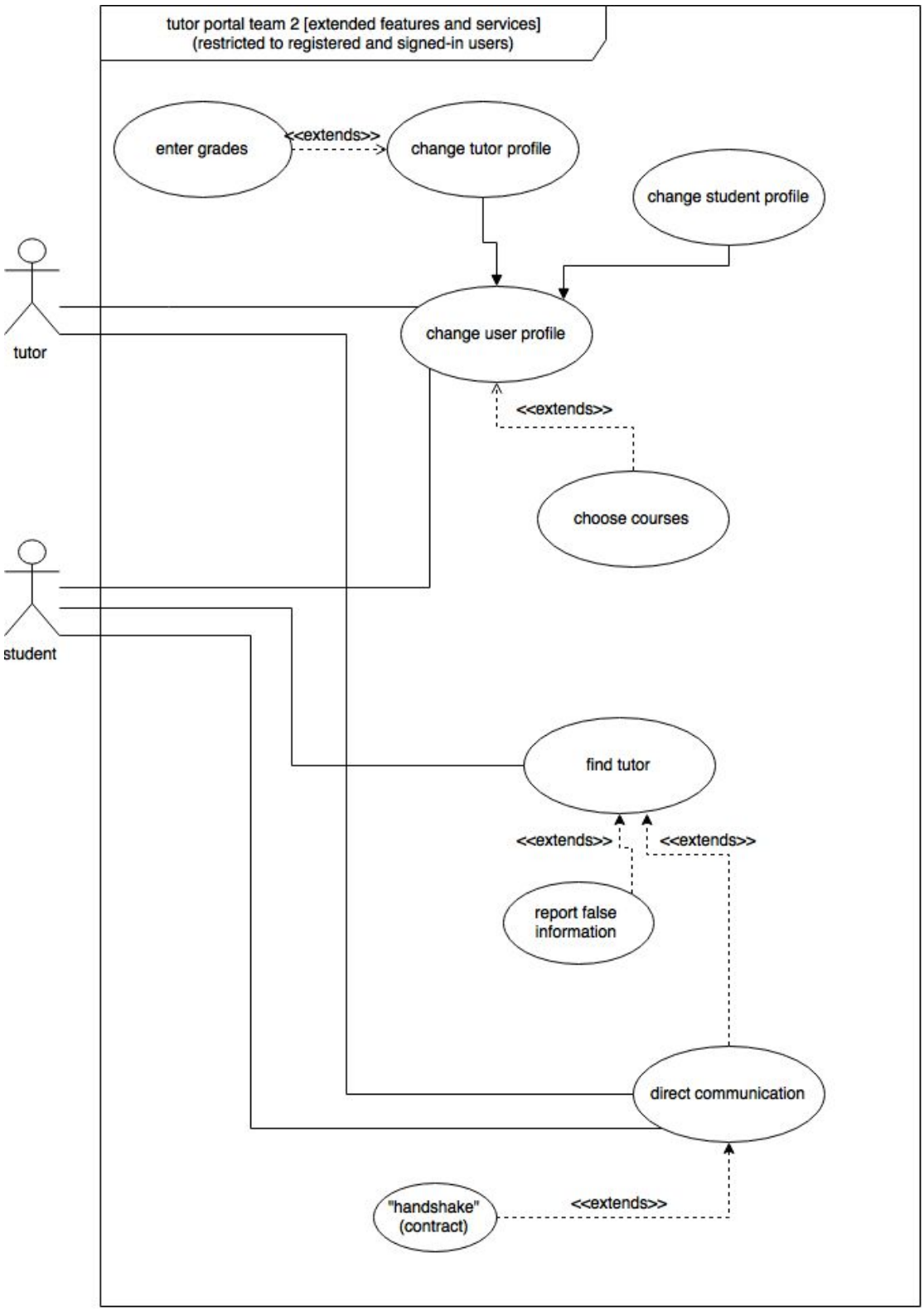
This project addresses these customer needs:

No.	Description of the need	Field	
		Customer needs *(User-Stories)*	**Project owner**
N-01	I want to find a tutor that studied the subject I'm preparing for.	X	
N-02	I want to make sure that a tutor can provide his services according to my needs.	X	
N-03	As a student I want to make sure that a tutor has good knowledge of the subject.	X	
N-04	As a student I want to contact a tutor and discuss the details of the engagement.	X	
N-06	As a tutor I want to effortlessly interact with potential customers.	X	
N-07	As a tutor I want to create a rich profile that highlights my skills and maximizes my visibility.	X	
N-08	As a tutor I want to protect my privacy.	X	
N-09	As a service provider I want to earn a commission on each engagement.		X

4 Use cases

In the following the most important use cases are listed and shown in diagrams. This set of use cases may not be complete and will be subject to change according to new information.


tutor portal team2 [account services]
forgot password
register user account
<<extends>>
tutor
users
sign in
<<includes>>
sign out
student
delete user account
<<includes>>
<<includes>>
contact staff
extended features and services
accessible only for registered and signed in users


tutor portal team 2 [extended features and services]
(restricted to registered and signed-in users)
enter grades
<<extends>>
change tutor profile
change student profile
tutor
change user profile
<<extends>>
choose courses
student
find tutor
<<extends>>
<<extends>>
report false information
direct communication
"handshake" (contract)
<<extends>>

4.1 Registration:

Actors:

Students

A person enrolled at the University of Berne in Computer Sciences is called a student. A student visiting this platform is in need of a tutor who can support him/her in a subject.

Tutors

A person who is (or was) enlisted at the University of Berne in Computer Sciences and provides tutoring services to current students is called a tutor. Tutors have typically passed the exams they are offering their tutoring services for.

Description:

Certain features, such as contacting a tutor and owning a profile are limited to registered users only. Certain profile information (e.g. grades, contact details) is also hidden to the public, requiring users to register in order to view this more sensitive information.

- As a student I want to register on the platform, so I can use all the features it offers.
- As a tutor I want to register on the platform, so I can use all the features it offers.

Trigger:

A student or a tutor visits the platform, chooses to register and visits the registration page.

Pre-conditions:

1. Both student and tutor don't have an existing account on the platform.
2. Both student and tutor must have and provide an active University of Berne email-address in order to register.
3. Both student and tutor must have a Matrikelnummer.

Post-conditions:

1. The student receives his student account and can now use all platform features.
2. The tutor receives his tutor account and can now use all platform features.

Main Scenario:

1. A person (user) visits the platform and choses to register
2. The user visits the registration page
3. The user chooses if he/she wants to register as tutor or as a student
4. The user provides all necessary information according to the chosen account type:
    a. Student: First and Last Name, E-Mail address, Password, Matriculation Number, Address, phone number
    b. Tutor: First and Last Name, E-Mail address, Password, Matriculation Number, Address, phone number

5. The system validates the provided information
6. The user clicks the register button to conclude the registration
7. The system creates a new account in the database
8. The user gets sent and email with a link to confirm the email-address
9. The user hits the link and gets redirected to our website
10. The system logs the user into his/her account
11. System asks the User to complete his profile information

Alternative Scenarios:

5a. User didn't provide a valid email-address (limited to @students.unibe.ch):

1. System will prompt the user to provide a valid email-address
2. User enters a valid email-address
3. Use case resumes at step 5

5b. User didn't provide a valid matriculation number (format: xx-xxx-xxx):

1. System will prompt the user to provide a valid matriculation number
2. User enters a valid matriculation number
3. Use case resumes at step 5

5c. User provides an email-address already in use:

1. System displays a message stating the email is already in use
2. System asks the user to either log in or provide an alternative email-address
3. Use case resumes at step 5

1-6a User cancels the registration:

1. System cancels the registration process and discards all entered data

Special Requirements:

User Registration at step 7 cannot take more than two seconds.

Notes:

Are the email-addresses limited to students.unibe.ch? ->yes the email-addresses are limited to unibe

Should there be a username? -> the username is the firstname.lastname

4.2 Sign-In

Actors:

Students and Tutors (Users)

Description:

As a user, I want to log into the platform so I can see and edit my profile, view more sensitive data and engage with other users.

Trigger:

User visits the website and completes the login form.

Pre-conditions:

1. User is not already logged in
2. User has an active account

Post-conditions:

1. User is logged into the System

Main Scenario:

1. User visits the website
2. User fills out the login form (Username and password)
3. User clicks the login button
4. System validates the provided information
5. System logs the user in
6. System removes the login form and shows a Sign-Out link instead

Alternative Scenarios:

4a. User provides false credentials:

1. System displays an error message
2. System prompts the user to provide valid credentials
3. Use case resumes at step 2

Special Requirements:

Credential Validation and Login (step 4 and 5) cannot take more than two seconds.

Notes:

1. Should the login procedure be in AJAX so the page doesn't have to be reloaded?
2. How long is an active user session (session timeout) ?

4.3 Sign-Out

Actors:

Students and Tutors (Users)

Description:

As a User I want to logout of the platform.

Trigger:

User clicks on the logout link.

Pre-conditions:

User must be logged in and have an active session.

Post-conditions:

User is logged out of the platform and can only use it the same way as an unregistered user.

Main Scenario:

1. User clicks on the logout link
2. System logs the user out
3. System removes the logout link and displays the login form instead

Alternative Scenarios:

2a. The User is already logged out because his/her session has expired.
1. Use case resumes at step 3

Special Requirements:

-

Notes:

-

4.4 Change Profile (Student)

Actors:

Students

Description:

As a student I want to be able to change my profile information.

Trigger:

Student visits his/her profile and clicks the “Edit profile ” link.

Pre-conditions:

1. Student is logged into his/her account

Post-conditions:

1. Student has changed profile information

Main Scenario:

1. Student visits his/her profile page
2. Student clicks on “Edit profile” link
3. System toggles profile display mode to editing mode, transforming labels into text fields
4. Student updates his/her information
5. Student can enter a link to his profile picture.
6. System validates entered information during editing
7. Student clicks “Save” button
8. System persists updated information into the database
9. System toggles profile editing mode back to viewing mode

Alternative Scenarios:

5a. Student provides false credentials:
1. System displays an error message next to the text field containing false information
2. System prompts the Student to provide valid information
3. Student provides accordingly
4. Use case resumes at step 5

6a. Student clicks “Cancel” button or aborts editing otherwise

1. System discards all changes
2. Use case resumes at step 8

Special Requirements:

Persisting data and returning to view mode (step 7 and 8) cannot take more than two seconds.

Notes:

4.5 Change Profile (Tutor)

Actors:

Tutors

Description:

As a Tutor I want to be able to change my profile information.

Trigger:

Tutor visits his/her profile and clicks the “Edit profile” link.

Pre-conditions:

1. Tutor is logged in with his/her account

Post-conditions:

1. Tutor has changed profile information

Main Scenario:

1. Tutor visits his/her profile page
2. Tutor clicks on “Edit profile” link
3. System toggles profile display mode to editing mode, transforming labels into text fields
4. Tutor updates his/her information
5. Tutor can insert link to his profile picture.
6. System validates entered information during editing
7. Tutor clicks “Save” button
8. System persists updated information into the database
9. System toggles profile display mode back to viewing mode

Alternative Scenarios:

5a. Tutor provides false credentials:
5. System displays an error message next to the text field containing false information
6. System prompts the Tutor to provide valid information
7. Tutor provides accordingly
8. Use case resumes at step 5

6a. Tutor clicks “Cancel” button or aborts editing otherwise
3. System discards all changes
4. Use case resumes at step 8

Special Requirements:

Persisting data and returning to view mode (step 7 and 8) cannot take more than two seconds.

Notes:

4.6 Choose course from database

Actors:

Students and Tutors (Users)

Description:

As a User, I want to be able to choose the courses I need (Students) or offer (Tutors) tutoring.

Trigger:

User is viewing his/her profile and clicks “Select courses”.
Or the user visits the page ”Courses” and selects a course to add to their profile.

Pre-conditions:

1. User is viewing his/her profile or visiting the “Courses” page.

Post-conditions:

2. User has added courses to their profile.

Main Scenario:

1. User views his/her profile
2. User clicks “Select courses”
3. System redirects the user to the “Courses” page and displays all available courses
4. User makes his/her choices
5. User clicks “Add” Button
6. The system adds the course to the users database and profile
7. The user remains on the “Courses” page and can add further courses if desired
8. When the user has added all the courses he wants to add he can return to his profile by clicking the “Profile” button.

Special Requirements:

Fetching from and persisting course selection data into the database cannot take longer than two seconds.

Notes:

-

4.7 Enter grades

Actors:

Tutors

Description:

As a Tutor, I want to enter my grades for my courses so I can display my expertise.

Trigger:

Tutor is viewing his/her profile and clicks “Edit grade” on one of his/her courses.

Pre-conditions:

1. Tutor is viewing his/her profile
2. Tutor has already selected at least one course

Post-conditions:

1. Tutor has entered a grade for his/her course

Main Scenario:

1. Tutor views his/her profile
2. Tutor clicks “Edit grade” on one of his/her courses
3. System allows Tutor to enter a grade for the course
4. Tutor confirms the grade by click or by pressing the “Enter”-key
5. System validates grade formatting
6. The user is back on the view of his profile

Alternative Scenarios:

4a. Tutor discards changes by click or by pressing the “Escape”-key
1. System discards changes
2. The user is back on the view of his profile

5a. Tutor enters invalid grade formating
1. System displays error message
2. System prompts Tutor to provide correct formatting
3. Use Case resumes at step 4

Special Requirements:

Grades can only be digits with the following formatting:
- x e.g. 4
- x.x e.g. 4.0
- x.xx e.g. 5.75

Notes:

- Grades can be entered freely and can’t be officially validated. See Assumptions.

4.8 Contact staff for help

Actors:

All website visitors (Visitors)

Description:

As a Visitor, I want to be able to contact the website’s staff in case I need help

Trigger:

Visitor browses to the contact form.

Pre-conditions:

None

Post-conditions:

An email is sent to the staff describing the visitor's problem.

Main Scenario:

1. Visitor clicks on "Contact" link
2. Visitor clicks on "send message to staff" link
3. System forwards Visitor to the contact form
4. Visitor enters his/her e-mail into the "E-Mail" field
5. Visitor enters subject
6. Visitor describes the problem
7. Visitor clicks "Send" button
8. System sends an e-mail to the staff
9. System thanks the Visitor for his/her request

Alternative Scenarios:

3a. Visitor is already logged in as Student or Tutor (User)
1. System automatically completes the "E-Mail" field with the User's data
2. Use Case resumes at step 4

6a. Visitor clicks "Cancel" or aborts the Use Case otherwise
1. System discards entered information
2. System returns Visitor to Index page

Special Requirements:

-

Notes:

-

4.9 Communication

Actors:

Students
A person enrolled at the University of Berne in Computer Sciences is called a student. A student visiting this platform is in need of a tutor who can support him/her in a subject.

Tutors
A person who is (or was) enlisted at the University of Berne in Computer Sciences and provides tutoring services to current students is called a tutor. Tutors have typically passed the exams they are offering their tutoring services for.

Description:

To set up a meeting with a tutor the student gets into contact with the tutor. He does so through a internal communication system on the website. Through a button on the tutor's profile the student can write him a message and when the tutor replies the student receives a notification in his inbox.

Trigger:

A student is on a tutor's profile and decides to send him a message by hitting the "Send message" button.

Pre-conditions:

The student must be signed into his account on the website.
The tutor must have a profile, which he has once he creates an account.
The student and the tutor are not in contact yet.

Post-conditions:

The student and the tutor are now in contact and can set up meetings.

Main Scenario:

1. A student is on a tutor's profile and hits the send message button.
2. The student is redirected to the "Create message" page where he can enter his message tothe tutor
3. The student then hits the send button.
4. The tutor receives his message in his inbox and reads it.
5. The tutor hits the reply button, which allows him to reply to the students message.
6. The tutor hits the the send button.
7. Now the student has a new message in his inbox and the steps 4 to 6 can be repeated over and over.

Alternative Scenarios:

3a. Student decides to cancel the message:
1. Student hits the cancel button on the “Create message” page
2. The messaging is discarded.
3. The student is back on the tutor’s profile again.

Special Requirements:

After pushing the send message button, the user will be redirected to the “Create message” page after a maximum of 2 seconds.

Notes:

-

4.10 Set up a contract

Actors:

Students
A person enrolled at the University of Berne in Computer Sciences is called a student. A student visiting this platform is in need of a tutor who can support him/her in a subject.

Tutors
A person who is (or was) enlisted at the University of Berne in Computer Sciences and provides tutoring services to current students is called a tutor. Tutors have typically passed the exams they are offering their tutoring services for.

Description:

Once the student and tutor have been in contact(by messaging) and they have decided on a first day to meet up, the tutor fills out a form which then serves as contract between the student and the tutor but also the project owner. With agreeing to this contract the student makes the payment, of which a percentage goes to the project owner and the rest to the tutor as a guaranty.

Trigger:

A tutor fills out a contract form.

Pre-conditions:

Both student and tutor have an existing account on the platform and the tutor is signed in.
The student and the tutor have been in contact and have agreed on a first meeting.

Post-conditions:

The student pays for the first meeting.
A contract for the first meeting has been set up.

Main Scenario:

1. A tutor and a user have decided to set up a contract.
2. The tutor goes to the set up contract form.
3. The user provides all necessary information:Name of tutor, name of student, date of first meeting, fee, agree to commission fee
4. The system validates the provided information
5. The tutor clicks the send contract button.
6. The systems sends out the contract in an email to the student
7. The student confirms the contract.
8. The system sends out a request to the third party payment system, which checks if the student's payment has been made.
9. The third party payment service notifies the system that the payment has been successful
10. The tutor and the student are now linked as friends, this information appears on their profile

Alternative Scenarios:

9a. The payment is not successful

1. The systems sends an email to the tutor and student saying that the contract has been declined
2. The tutor and student receive an email saying that the contract could not be set up

10a. Tutor or student does not want to be friends

1. The student or tutor can unfriend the other on their profile

Special Requirements:

Payment process is started automatically, once the student agrees to the contract.
The validation of the information should take no more than 2 seconds.

Notes:
Is the tutor's name filled in automatically?
Is there a minimum fee?

4.11 Finding a tutor

Actors:

Students
A person enrolled at the University of Berne in Computer Sciences is called a student. A student visiting this platform is in need of a tutor who can support him/her in a subject.

Tutors
A person who is (or was) enlisted at the University of Berne in Computer Sciences and provides tutoring services to current students is called a tutor. Tutors have typically passed the exams they are offering their tutoring services for.

Description:

As a student i want to find a good tutor,to find these tutors the website offers suggestions to me once i have chosen the courses for which i need help. This saves the student time by offering him ideal options.

Trigger:

A student chooses look at the tutor suggestions for a course on his profile.

Pre-conditions:

Both student and tutor have an existing account on the platform and the student is signed in.
The student has chosen a course out of the catalogue and put it on his profile.
The tutor offers tutoring for that course.

Post-conditions:

The student get to look at different tutors profile and gets to see which one is fitting for him.

Main Scenario:

1. A student chooses to look at suggestions for one of the courses on his profile
2. A table appears underneath the course containing the following information:Tutor name, grade, year of exam, fee
3. Student can then choose to sort the table by the different attributes
4. The student chooses a tutor and clicks on his name
5. The system takes the student to the profile of the tutor
6. The student browses through the profile
7. The student hits return and looks at the table again
8. Student repeats steps 4-7 until he finds a tutor that suits him

Alternative Scenarios:

1-3a. The student stops the searching process:
1. System removes the table and student is back on his original profile

8a.The student can not find a tutor that matches
1. Student hits refresh button
2. System updates tutor table
3. Use case resumes at step 3

Special Requirements:

Loading table can take up to 5 seconds maximum

Notes:

-

4.12 Report false information

Actors:

Students
A person enrolled at the University of Berne in Computer Sciences is called a student. A student visiting this platform is in need of a tutor who can support him/her in a subject.

Tutors
A person who is (or was) enlisted at the University of Berne in Computer Sciences and provides tutoring services to current students is called a tutor. Tutors have typically passed the exams they are offering their tutoring services for.

Project owner
The people who own and host the website are in charge of dealing with technical problems and false information. Also they are in charge of deleting profiles.

Description:

Since the information on the profiles of the students and the student is not verified, some might be tempted to post false information. For that purpose we include a report false information button on every profile, so the project owner can intervene. That way the profiles are more reliable.

Trigger:

A student or tutor chooses to report a profile, by clicking the report button.

Pre-conditions:

The student/tutor needs to know what information is false.
The student/tutor must have an account and be signed in.

Post-conditions:

The profile with the false information is reported.
The project owner acts as he sees it, by removing the false information or even deleting the profile

Main Scenario:

1. A user sees false information on a profile and decides to report it
2. The user hits the report button
3. A message window pops up
4. The user enters the exact detail of the false information
5. The user hits send
6. The system sends out a thank you and confirmation message to the user who reported

Alternative Scenarios:

4a. The user decides to abort the report:
1. User hits the cancel button
2. Message window is closed and user is again on the profile he was on before

Special Requirements:

Message window should pop up within 2 seconds

Notes:

-

4.13 Delete profile

Actors:

Students
A person enrolled at the University of Berne in Computer Sciences is called a student. A student visiting this platform is in need of a tutor who can support him/her in a subject.

Tutors
A person who is (or was) enlisted at the University of Berne in Computer Sciences and provides tutoring services to current students is called a tutor. Tutors have typically passed the exams they are offering their tutoring services for.

Project owner
The people who own and host the website are in charge of dealing with technical problems and false information. Also they are in charge of deleting profiles.

Description:

If the students or tutors no longer need the services of the website and decide to delete their account they can do so by hitting the delete button on the “Edit profile” page.

Trigger:

A student or tutor chooses to delete his profile and hits the delete button. .

Pre-conditions:

The student/tutor must have an account and be signed in.
The student/tutor is on the “Edit profile” page.
The student/ tutor does not need the services of the website anymore.

Post-conditions:

The profile of the student/tutor will be deleted.
The student/tutor can not log-in anymore.

Main Scenario:

1. A student/tutor wants to delete his profile and hits the delete profile button
2. Student/tutor is asked if he really want to delete the profile.
3. Student/tutor hits the yes button
4. The system deletes the profile
5. The project owners change their statistics

Alternative Scenarios:

3a. The student/tutor decides not to delete the profile:
1. Student/tutor hits the cancel button when asked if he really wants to delete the profile
2. The student/ tutor is taken back to the “Edit profile” page

Special Requirements:

-

Notes:

-

4.14 Retrieve forgotten password

Actors:

Students
A person enrolled at the University of Berne in Computer Sciences is called a student. A student visiting this platform is in need of a tutor who can support him/her in a subject.

Tutors
A person who is (or was) enlisted at the University of Berne in Computer Sciences and provides tutoring services to current students is called a tutor. Tutors have typically passed the exams they are offering their tutoring services for.

Project owner
The people who own and host the website are in charge of dealing with technical problems and false information. Also they are in charge of deleting profiles.

Description:

If a student/ tutor forgets his password he can not sign in to his account. That is why there is this system to retrieve the forgotten password.

Trigger:

A student/tutor clicks the forgot password button next to the login button.

Pre-conditions:

The student/tutor must have an account.
A student/tutor has forgotten his password.

Post-conditions:

The student/tutor has his password again and can sign in.

Main Scenario:

1. A student/tutor hits the forgot password button next to the login button
2. A form pops up
3. Student/tutor has to enter his @students.unibe.ch email address
4. Student/tutor hits send password to email address
5. The system gives out a message that email was sent
6. System sends email with a link containing the password
7. The student/tutor comes back onto our website
8. The student/tutor signs into his account

Alternative Scenarios:

3a. The student/tutor remembers his password in the process:
1. Student/tutor hits the cancel button in the form
2. The student/ tutor is taken back to the main website and can sign in normally

Special Requirements:

Email with password should be received within 10 seconds.

Notes:

-

5 Project order

A web-application for interacting students and tutors is developed.

The solution is available in winter 2015.

The requirements are detailed available on our ESE2015-team2 board on trello.

The project contains any implementations and changes for all correlating systems and all user interface and database applications specified on the ESE2015-team2 board on tello..

6 Trade-offs and conflict potential

The following elements could lead to trade-offs or conflicts, these points have to be discussed in an early phase with all stakeholders:

- How are changing courses managed?
- How do you ensure the truthfulness of grades?
- How do you prevent students from paying tutors directly at their physical meeting?
- Who will include new courses into the system?

- How is the tutor's expertise measured and validated? Are grades the only indicator?

7 Assumptions

- The finished platform will be incorporated into the developer's server farm
- The developer will provide the necessary infrastructure, alleviating the project owner from high initial capital investments
- The developer will maintain the platform and continue development after release
- With no initial design specifications, the developer will provide prototypes and iteratively enhance the design towards the finished product
- During development, the developer will strife to always have a representative, running version of the platform available to the customer
- With no further mechanics in place, the platform heavily relies on trust regarding tutor's expertise and payments between students and tutors.
- Tutor can not delete or change ratings on their profile
- Tutors can not take the role of students on website at the same time
- The project owner is not allowed to own payment information of tutors, the payment service provider can identify tutors at the time of payment
- Payment is provided automatically at time of payment to project owner
- The project owner does not keep records of payments
- Passwords are encrypted

8 Requirements

8.1 Priority of requirements (must / should / nice to have)

The priorities are done according must / should / nice to have model as far as possible.

Requirements with priority MUST, have to be in the release.

Requirements with priority SHOULD, have also to be in the release, however it's possible to use manual workarounds to achieve the requirement.

Requirements with priority NICE TO HAVE can be thrown out in case of missing resources.

The priorities are listed and managed directly on the trello ESE2015-team2 board.

8.2 Hints

The requirements of the SRS are structured according topic and use case on Trello:
Every use case card on trello has in its description a list of linked requirement cards. These are the resulting requirements of the use case.

8.3 List of requirements

The list of the requirements is managed directly on the trello ESE2015-team2 board.

9 Risk management

9.1 Project risks

ID	Risks	Probability[1]	Impact[2]
1	Missing project resources	frequently	high
2	User journeys are not clear / Interface isn't intuitive	possible	high (reduces the customer experience)
3	Missing of deadlines	possible	undefined

9.2 Project risks - plan

ID	Risks	Massnahmen
1	Missing project resources	Work as a team, bottlenecks have to be shown as early as possible!
2	User journeys are not clear / Interface isn't intuitive	For every use case concrete user journeys are defined and tested.
3	Missing of deadlines	Communication and organisation is the key here..

9.3 Product Risks

ID	Risks	Probability[3]	Impact[4]
1	Customers get in contact but do not pay the fee.	occasionally	medium

1 Probability: occasionally / possible / frequently / occurs

2 Impact : small / medium / high / catastrophic

3

4