Project Requirements

Project Name: Tutor Negotiating

Team: ese2015-team7

Customer: Niklaus

Revision History

Version	Date	Revision Description
.01		
.02		
..		
1.0		

Date: October 6, 2015

Table of Contents

06/10/2015

Introduction

1 Purpose

The goal of this project is to build a web portal targeted to university students who are preparing for a specific exam. The portal/platform connects students with tutors so they get well prepared for exams and improve their knowledge as well as understanding for specific lectures.

2 Stakeholders

- Students
- Tutors
- Customer Niklaus
- ese2015-team7

3 Definitions

4 System overview

The system provides services to sign up and login. When signing up, there's no difference between signing up as a student or tutor. Both user types have to fill in basic information like name, password, email and user name. It should also be possible to extend basic- with optional information like profile picture, short biography, grades, current semester and available time slots.

Once logged in, a selection based search can be done the find matching tutors. There are two types of searching: basic search and specific search. For basic search, the user first selects a university, then a subject and finally a lecture. This will list all tutors offering their help regarding the selections. The specific search mode extends basic search with grades and available time slots selections.

If a student finds a tutor he wants to get in touch with, he contacts the tutor by sending a request. The tutor then can accept and pays a fee for the contact information (like an e-mail-address or a phone number). Payment can be a monthly fee (that starts from the first engagement) or just a single fee for every contact engagement. There is no other communication possible (like a built-in communication system).

Students who got in touch with tutors should have the possibility to rate them. Due to this rating system, students can choose more reliable between several tutors regarding their rating.
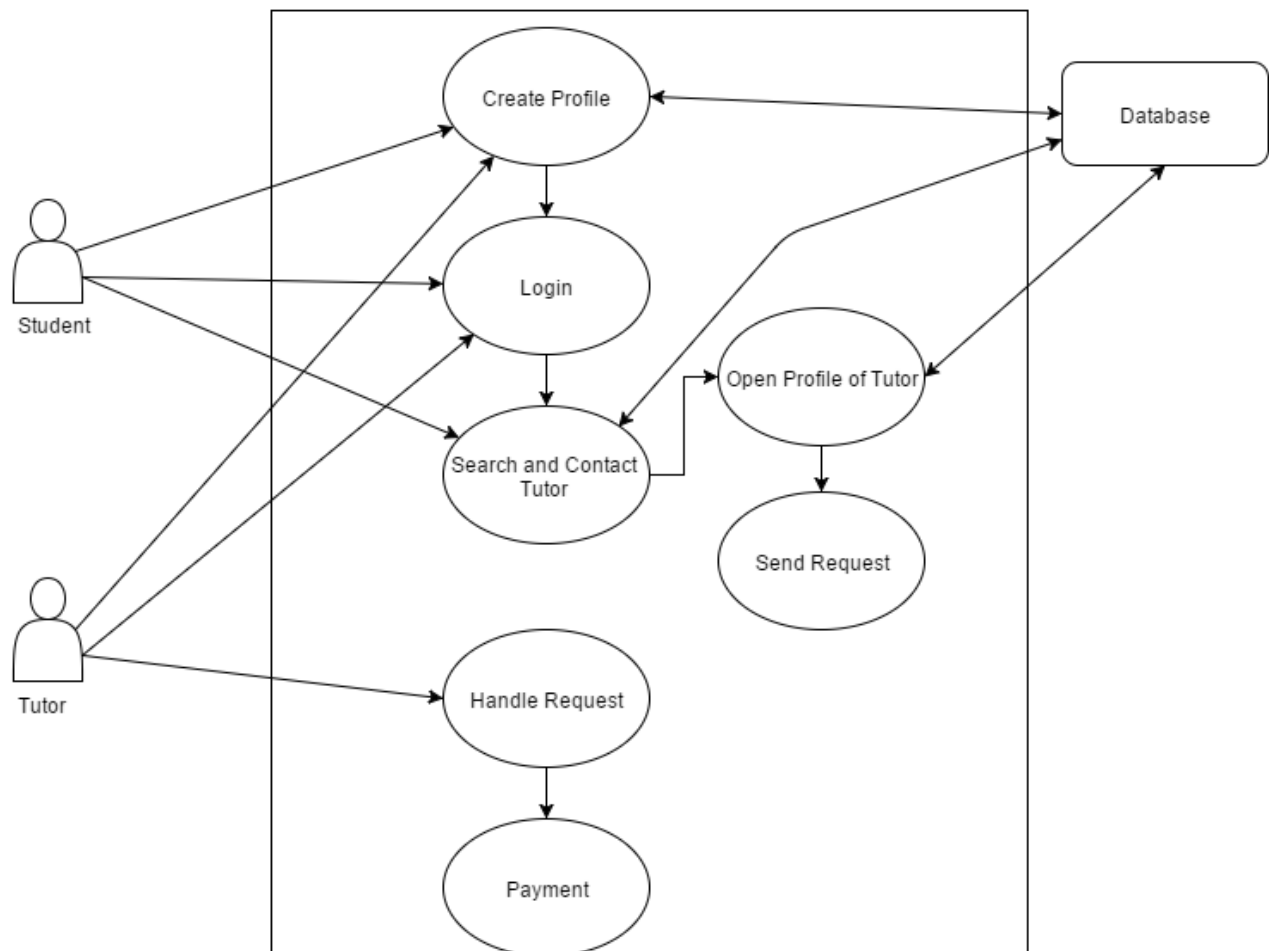
5 References

Here are some links to existing portals (too generic ones) our customer mentioned.

- https://tutor24.ch
- http://www.nachhilfe-vermittlung.ch
- http://owltutors.ch

Overall description

A. Diagram

B. Use cases

List of Use cases:

<mark>On TRELLO!!! Added when finished!</mark>

1. **Create Profile (Student/Tutor)**

   1.1. Actors

   Customer (Student/Tutor)

   1.2. Description

   As a Customer I want to create a new profile/account to use your services

   1.3. Trigger

   Click on the sign-up button

   1.4. Pre-conditions
   1. Customer has a valid email-address
   2. Customer gives a not yet used Username

   1.5. Post-conditions
   1. A new profile/account is created in the database with the given information
   2. Validation email is sent to the given email-address

   1.6. Main scenario
   1. Customer clicks on the sign-up button
   2. Customer enters his information
   3. System checks validity of email-address
   4. System checks validity of Username
   5. System sends a validation email to the given email-address
   6. A new profile/account is created in the database with the customer's data

   1.7. Alternative scenarios

   3.1.   Given email-address is not valid

       3.1.2. System will prompt customer to enter a valid email-address

       3.1.3. User gives a valid email-address

       3.1.4. Use case resumes on step 4

   4.2.   Given username is already in use

       4.2.1. System will prompt customer to enter a different username, stating that the given username is already in use

       4.2.2. User gives a valid username

       4.2.3. Use case resumes on step 5

   1.8. Special requirements

   1.9. Notes
   1. What is the maximum amount of characters used for username?
   2. Are there any special conditions on the username, e.g. at least one number, or at least one capital letter?

2. Login

2.1. Actors
Customer (Student/Tutor)

2.2. Description
As a Customer I want to log in to the page.

2.3. Trigger
Customer clicks on login.

2.4. Pre-conditions
1. Customer has an account.

2.5. Post-conditions
1. Customer is at the Home-Page.

2.6. Main scenario
1. System requests User – Information
2. Customer enters user-name/e-mail
3. Customer enters user-password (******)
4. Customer presses login button
5. System verifies valid user-account
6. System loads Personal-Home-Page

2.7. Alternative scenarios
2.1. Wrong or/and incomplete login information
    2.1.1   Customer enters wrong or/and e-Mail/Username or/and password
    2.1.2   Customer presses login button
    2.1.3   System requests valid account-information
    2.1.4   Scenario continues at step 1

1.1. Extra Steps for validation of account
    1.1.1.   System requests User – Information
    1.1.2.   Customer enters user-name/e-mail
    1.1.3.   Customer enters user-password (******)
    1.1.4.   Customer presses login button
    1.1.5.   System checks account and requests validation
    1.1.6.   Customer enters validation-code
    1.1.7.   System validates account
    1.1.8.   System loads Welcome-Page

2.8. Special requirements

2.9. Notes

3. Search and Contact Tutor

3.1. Actors
Customer (Student)

3.2. Description
As a Customer I want to find a tutor in a specific course at a specific university in a specific subject.

3.3. Trigger
Customer clicks on the search box.

3.4. Pre-conditions
1. Customer has a validated account
2. Customer must be signed up
3. Customer must be logged in

3.5. Post-conditions
1. Customer receives Mail with confirmation of his request.
2. Tutor gets a request from the Student

3.6. Main scenario
1. The Customer chooses his university
2. The customer chooses his subject
3. The customer chooses his course
4. The customer clicks the search button
5. System prompts customer to choose one tutor out of a list of all possible tutors
6. User chooses a tutor and presses "new request"
7. the customer presses "send request".

3.7. Alternative scenarios
  4.1. Incomplete search information
    4.1.1   Customer presses Search button, before having chosen a course
    4.1.2   System prompts request of missing information.
    4.1.3   Scenario continues at step 1

3.8. Special requirements
Customer should not be able to search classes he teaches as well. (why not?)

3.9. Notes

4. Tutor handles Student request

4.1. Actors
Customer (Student) - Student
Customer (Tutor) - Tutor

4.2. Description
A Tutor receives a request from a student and accepts or rejects the request.

4.3. Trigger
Student sends request to tutor.

4.4. Pre-conditions
1. Both Customers have a validated account.
2. Tutor must be signed up
3. Tutor must be logged in

4.5. Post-conditions
1. Tutor must pay fee to see contact information
2. Student gets informed whether the tutor accepted him (with contact info) or not
3. In case Tutor accepts the request, he/she gets the student's contact information after valid payment

4.6. Main scenario
1. Tutor gets request on Home-Page.
2. Tutor presses on the new anonymous request
3. System loads the new anonymous request
4. Tutor presses "accept" and reveals the username of the student
5. System initiates payment
6. System deletes requests
7. System notifies Student

4.7. Alternative scenarios
4.1. Tutor rejects the request
    4.1.1. Tutor presses "reject"
    4.1.2. System deletes request
    4.1.3. System notifies Student by email

4.8. Special requirements

4.9. Notes

06/10/2015

5. Open Profiles of Tutors

5.1. Actors

Customer (Student) - Student

Customer (Tutor) - Tutor

5.2. Description

As a Student I want to see the profile of the Tutor to decide if I want to contact him

5.3. Trigger

Student clicks on the name of the tutor in search

5.4. Pre-conditions

1. Student must have used the search engine to find the name of the tutor
2. Student must be signed up
3. Student must be logged in

5.5. Post-conditions

1. Student sees the profile of the Tutor

5.6. Main scenario

1. Student clicks on the name of a Tutor
2. System loads the profile of the Tutor

5.7. Alternative scenarios

2.1. System produces error while loading the profile of the Tutor
   2.1.1. System produces error while loading the profile of the Tutor
   2.1.2. System creates a message describing the error and prompting the Customer to refresh

5.8. Special requirements

5.9. Notes

6. Payment Procedure

6.1. Actors
Customer (Tutor) - Tutor

6.2. Description
As a Tutor I want to pay the fee to unlock the contact information of the students

6.3. Trigger
Tutor clicks to pay the fee

6.4. Pre-conditions
1. Tutor has received a request
2. Tutor must be signed up
3. Tutor must be logged in
4. Tutor must have a valid payment method
5. Tutor's payment method must have enough credit for the transaction

6.5. Post-conditions
1. The fee is paid
2. The contact information of the students is unlocked

6.6. Main scenario
1. Tutor receives a request
2. Tutor clicks to pay the fee
3. System prompts for payment information
4. Tutor enters his payment information
5. System validates the payment information
6. System makes the transaction
7. The fee is paid
8. System sends a confirmation to the Tutor

6.7. Alternative scenarios
4.1. Tutor enters invalid payment information
   4.1.1. Tutor enters invalid payment information
   4.1.2. System finds the payment information invalid
   4.1.3. System prompts for valid information
   4.1.4. Scenario continues at step 3

5.1. Payment method of Tutor does not have enough credit
   5.1.1. System validated the insufficiency of Tutor's payment method
   5.1.2. System displays a message with the information
   5.1.3. System prompts for another payment method
   5.1.4. Scenario continues at step 3

6.1. System fails to make the transaction
   6.1.1. System tries to make the transaction and fails
   6.1.2. System backrolls
   6.1.3. System displays a message with the error information
   6.1.4. System prompts to try again
   6.1.5. Scenario continues at step 3

6.8. Special requirements

6.9. Notes

Specific requirements

(define all the [functionalities](#) that your application needs to fulfil the scenarios described in section 2)

1   Functional requirements

1) The web portal allows to all the users to create a profile (sign up) by just indicating the **name**, an **e-mail-address**, a **password** and a **username**. The only public viewable information of this (basic) profile is the username.

2) If a user is once signed up, the web portal allows him to login with his **username** or **e-mail-address** and his **password**. If the login-data are correct and he clicks on the login button, he will be automatically gets the view of his profile.

3) The users can **extend his profile** with optional information like profile **picture**, short **biography**, **grades**, current **semester** and available **time slots**.

4) A student can start a **basic search** for a tutor where he can chose with the first dropdown-field the **university**, then with the second one the **subject** and on the third one the **lecture**. Once chosen the student can click on the "search" button and the web portal will check on the databases for all the **matching tutors**.

5) With a **specific search function** the student can although search for minimum **grades** and open **time slots**.

6) To **protect** the **privacy** of the tutors there are only this two search options (basic and specific search). A user can't search for real names or for an e-mail-address.

7) As a **search result** the student gets a list of all the **matching tutors**, with:

   - the **usernames** (directly linked to the tutors profiles),
   - the grads of the tutors (if indicated) and
   - for every tutor a "**new request**"-button

    Or he gets an **error message**, "that no tutor is available for the lecture he searched".

8) If at least one tutor is found, the student can click on the "**new request"**-button behind every tutor on the list and our portal sends automatically a **request-message** to the chosen tutor.

9) The tutor gets this anonymous request-message on his own profile and can then click the "**accept**"- or the "**reject**"-button.

10) If the tutor has already paid the monthly fee he can click the "accept"-button without any other costs. Otherwise the system prompts him to pay choosing between the two available options: Monthly or individual fee. As a result the tutor gets the contact information of the students or he gets the confirmation that he didn't accept the request of this student.

2   Non-functional requirements

(external, performance, etc.)

Product requirements:

1) The users got to have **access** on the web portal **from different platforms**. (Windows, Mac, Smartphones, Tablets,…)

2) The **data** are always **on the server**, the users can only access them with a server request for the web portal.

Company requirements:

3) The development of the web portal will be organized and executed on the platforms **Trello** and **GitHub**. The developers and the clients have to have access to these sources at any time.

4) The progress of the development process and the state of the requirements will be checked every week in **a scrum standup meeting** or in a more specific meeting. This protects us from doing unnecessary work and guarantees that the developers and the customers are on the same page.

Safety requirements

5) The developers and the clients have to **protect the data** of the users and both are doing this to the best of their knowledge and belief.