

OOAD

Lab#3: Use Case and Scenario

Product Owner: Tharita Tipdecho 59090035

System Analysis: Napat Traikityanukul 59090011

All actors:

1. Student
2. Tutor
3. Payment Authorization Service.

Brief Use Case:

- **Search tutor:** A student chooses to search for a tutor. The system displays a list of all the suggested tutors due to the user's previous search preferences. If the student has no previous search yet, the system displays nearby tutors based on the student's location. A student inputs all the preferences information about the tutor and about the session (gender, age, subject, price, period, place, etc.) The system checks the database and displays the matched tutor(s). If the student has marked any tutor as his/her favorite, the system will collect the specifications and characteristics of that tutor into the database.
- **Choose tutor:** When a student chooses a tutor, the system sends the request to the tutor. Waiting for the tutor's response, if the tutor accepts, the student performs *Pay Payment* use case within a certain period of time. The tutor has been successfully chosen.
- **Cancel tutor:** If the student has necessary reason(s) to cancel the course, he/she has to give the reason why, and if he/she has been cancelling too many times, the student will be banned from

the system. The days of banning will depend on the number of times the student has been putting non-sense reasons, or no reason at all, for each tutor cancellation.

- **Rate the tutor:** After ending each session, the student has to rate the tutor from one to five. With five stars implying the most satisfied. If the tutor has high ratings, when others student search for tutors, the tutor will be shown in the top of the list.
- **Sign up:** A user has to fill in all the required information such as name, address, phone number, bank account, and e-mail address, in order to register to the system. Then, the user has to verify himself/herself to make sure that he/she is a real person.
- **Log In:** A user has to log in before being able to get in to the system. The user fills in the required details in order to log in to the app. If they do not remember their password, they perform *Forgot Password* use case.
- **Log out:** When a user wants to sign out/ log out, the user can choose the log out option. The user would then be logged out from the system.
- **Forgot Password:** if a user forgets his/her password, the system will ask the user to type in his/her email again and the system will send a reset password link to his/her email address.
- **Pay payment:** After confirming from both side (student and tutor), student has to pay for the session by online banking with his/her bank account that student used when registered.
- **Respond to Request:** When student wants to choose the tutor, the system will send a request to the tutor. The tutor has to

accept or reject the request. If the tutor rejects the request, the tutor has to provide a reason for it.

- If the tutor wants to accept but prefers other period of time, or other location, the tutor can reject the request first, and specify his/her preference in the reason box.

Casual Use Case: Choose Tutor

Main Success Scenario:

When a student chooses a tutor, the system sends the request to the tutor. Waiting for the tutor's response, if the tutor accepts, the student performs *Pay Payment* use case within a certain period of time. The tutor has been successfully chosen.

Alternative Scenario:

- a. If student/tutor wants to cancel the course, they have to give the reason why and the system will record the activity. If they have cancelled the course too many times in the same time the system will ban the user, with the number of banning days depending on the number of times they have cancelled the courses with unacceptable reasons (non-sense reasons, or no reason at all).
- b. If after the course student has not rated the tutor, the system will remind the student in several times.
- c. If student chooses the tutor but did not authenticate themselves, they have to authenticate themselves first, then authenticate their bank's account. After that the system will continue performing *Choose Tutor* use case.

Fully-dressed: Choose Tutor

Primary Actor: Student

Stakeholders and Interests:

- Student: Wants tutor that matches the student's preferences
- Tutor: Wants money
- Payment Authorization Service: Wants to receive digital authorization requests in the correct format and protocol. Wants to accurately account for their payables.

Preconditions: Student and Tutor are identified and authenticated.

Success Guarantee (Postcondition): Student gets the tutor matching the student's preferences. Tutor gets student who surely attend each session. Payment authorization approvals are recorded.

Main Success Scenario (or Basic flow):

1. Student performs *Search Tutor* use case.
2. Student chooses their desired tutor from the list.
3. System sends the request to the tutor.
4. Tutor accepts the request.
5. Student performs *Pay Payment* use case.
6. The tutor has been successfully chosen.

Extensions (or Alternative flow):

1. The student cancels the session
 - a. Give the reason why
 - b. System records the activity
2. The tutor cancels the session
 - a. Give the reason why
 - b. System records the activity
3. Invalid identifier
 - a. Let student/tutor authenticate him/her self.
 - b. Authenticate bank's account.
4. Student has not rated the tutor
 - a. Remind them in several time.

Personas

1. Elementary student

- a. Age 6 to 13 years old.
- b. In their first school years, where they get primary education before they enter secondary education.
They do not need professional at subject, high-school or university student can be their tutor.

2. High-school student

- a. Age 14 to 18 years old.
- b. They need some tutor that might be undergraduate or person who is profession about the subject that they going to learn to make sure that the tutor can be teaches them correctly in the right way.

3. Undergraduate or higher student

- a. Age 18 above.
- b. They need tutor that who is profession about the subject like *high-school student*, but need more specify the tutor's requirement (lots of experience in teaching, very nice portfolio, etc.)

4. Tutor

- a. Age no requires.
- b. Depends on tutor's education that can teaches which subject and level(Elementary, high-school, undergraduate ,or higher).