

FACULTY OF COMPUTERS, INFORMATICS AND MICROELECTRONICS

TECHNICAL UNIVERSITY OF MOLDOVA

OBJECT-ORIENTED MODELING AND ANALYSIS

LABORATORY WORK #3

**Sequence Diagrams. Functional and
Non-Functional Requirements. Conceptual
Object Oriented Analysis. Technical Object
Oriented Design.**

Author:

Cernei LIVIU

Supervisor:

Mihail GAVRILIȚA

Chișinău 2018

Laboratory work #3

1 Tasks

- Model your application using 3 Sequence Diagrams;
- Analyze the Functional and Non-Functional Requirements for your project (at least 5 of each).

2 Theory

2.1 Object-Oriented Analysis

- a) Elicit requirements: Define what does the software need to do, and what's the problem the software trying to solve.
- b) Specify requirements: Describe the requirements, usually, using use cases (and scenarios) or user stories.
- c) Conceptual model: Identify the important objects, refine them, and define their relationships and behavior and draw them in a simple diagram.

2.2 Conceptual Model

- a) Identifying Objects
- b) Refining Objects
- c) Drawing Objects
- d) Identifying Object Relationships
- e) Identifying Object Behaviors

3 Sequence Diagrams

In Figure 3.1 is represented the sequence diagram for the login operation.

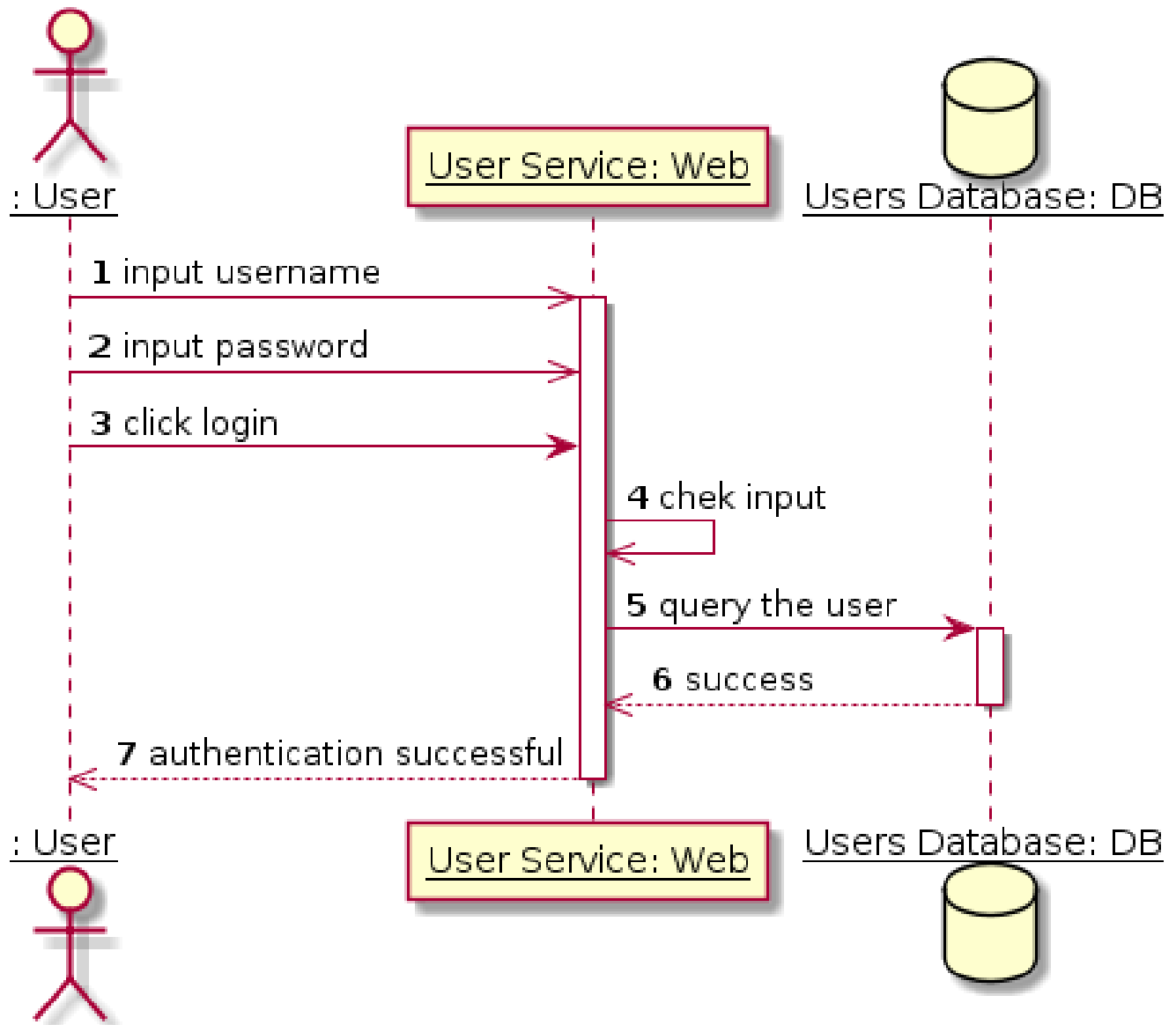


Figure 3.1– Login sequence diagram

We observe the sequence of operations needed to login any user: student, teacher, or even admin. It is a generalized operation, that's why I used an anonymous class. The user inputs username and password, the user service checks it and sends the information further to the database. The database returns an "success" message, which is propagated to the user by the service. This is the case when all information was correct, and results with the user authenticated.

In Figure 3.2 is represented the sequence diagram for creation of a new course.

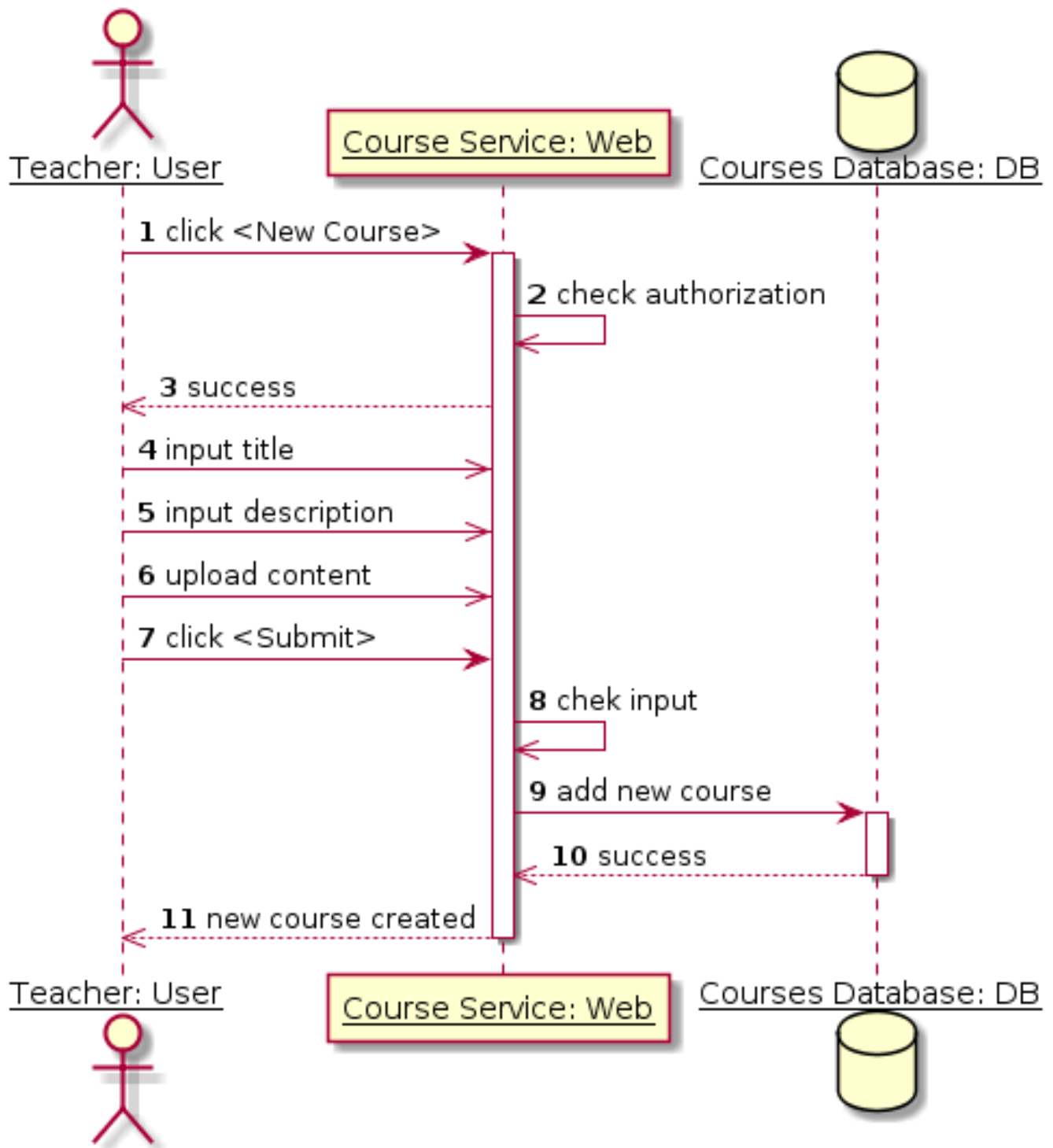


Figure 3.2– New course sequence diagram

This diagram is specifically for the teacher. In order to create a course, the teacher should have assigned a special authorization key. If it is found, he can begin to input the course information. When submitted, it is checked for invalid fields, then it is sent to the course database. In case of success the teacher is announced through the course service.

In Figure 3.3 is represented the sequence diagram for giving feedback for a course.

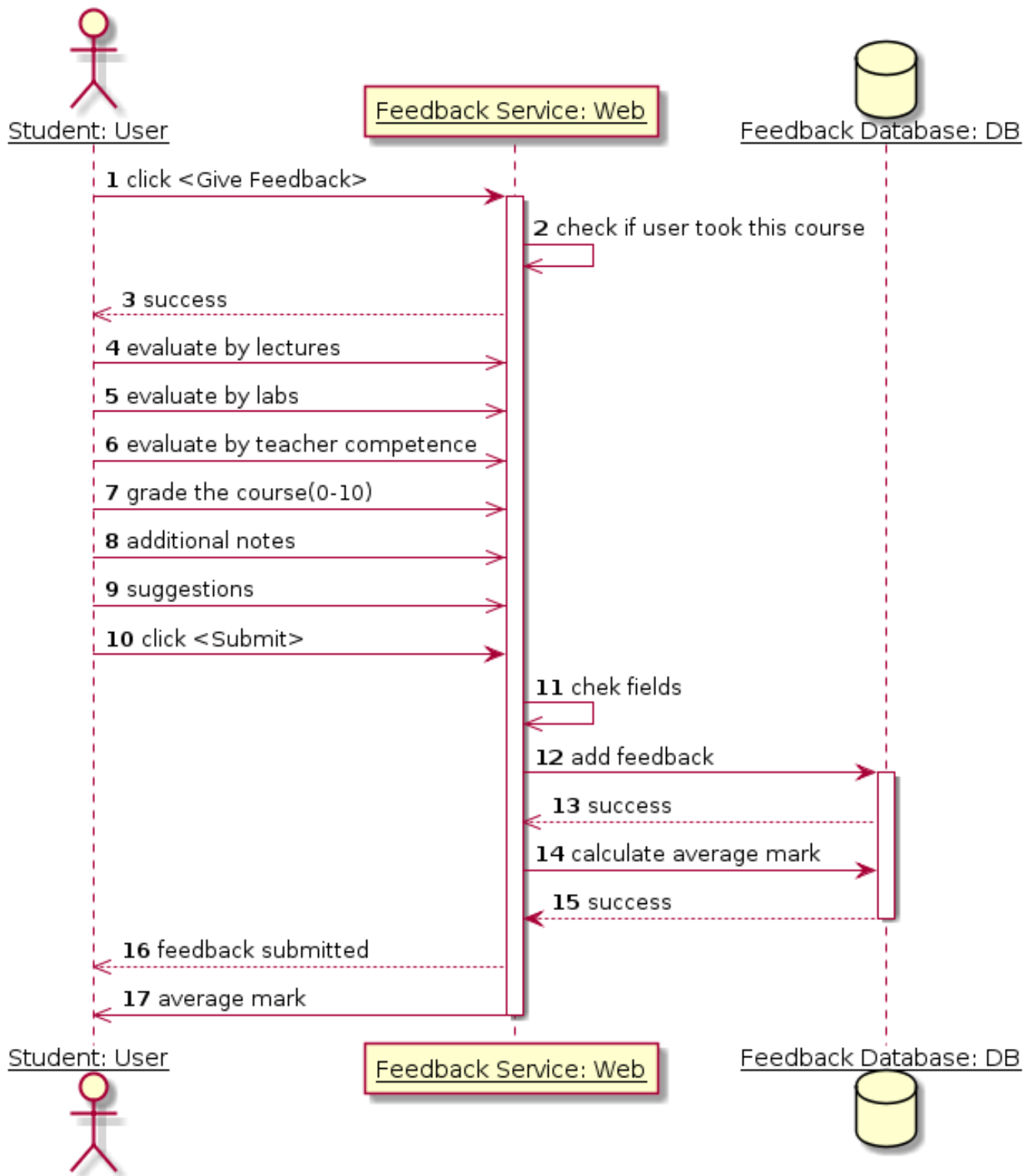


Figure 3.3– Feedback sequence diagram

To be eligible to give feedback, the student should have already completed the course. If the condition is satisfied, he can fill the form. After submission, all fields are checked and the information is sent to the database. After a "success" message, the feedback service requests the average mark for that course. Then the student receives an acceptance message and can see the overall quality of the course.

4 Functional and Non-Functional Requirements

4.1 Functional

- Provide access to courses
- Teachers can create new courses
- Students can take tests
- Students can give feedback
- Teachers can invite and enroll students

4.2 Non-Functional

- The site supports multiple users at the same time
- The information is stored and encrypted safely
- The site provides very fast access to data
- The site can provides access in multiple countries.
- The information is processed real-time.

5 Conclusion

In this laboratory work we learned to create sequence diagrams and write functional and non-functional requirements.