

FACULTY OF COMPUTERS, INFORMATICS AND MICROELECTRONICS
TECHNICAL UNIVERSITY OF MOLDOVA

OBJECT-ORIENTED MODELING AND ANALYSIS

LABORATORY WORK #4

**Modeling your project with Collaboration
Diagrams. Technologies. Design Principles.**

Author:

Cernei LIVIU

Supervisor:

Mihail GAVRILIȚA

Chișinău 2018

Laboratory work #4

1 Tasks

- Model your application using 3 Collaboration Diagrams;
- Describe the Technologies you plan on using in your project. I want to know what, why and how you are planning to use them. Sell them to me.

2 Theory

2.1 Collaboration Diagrams

The UML Collaboration diagram is used to model how objects involved in a scenario interact, with each object instantiating a particular class in the system. Objects are connected by links, each link representing an instance of an association between the respective classes involved. The link shows messages sent between the objects, and the type of message passed (synchronous, asynchronous, simple, balking, and timeout).

Collaboration diagrams offer a better view of a scenario than a Sequence diagram when the modeler is trying to understand all of the effects on a given object and are therefore good for procedural design.

3 Collaboration Diagrams

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

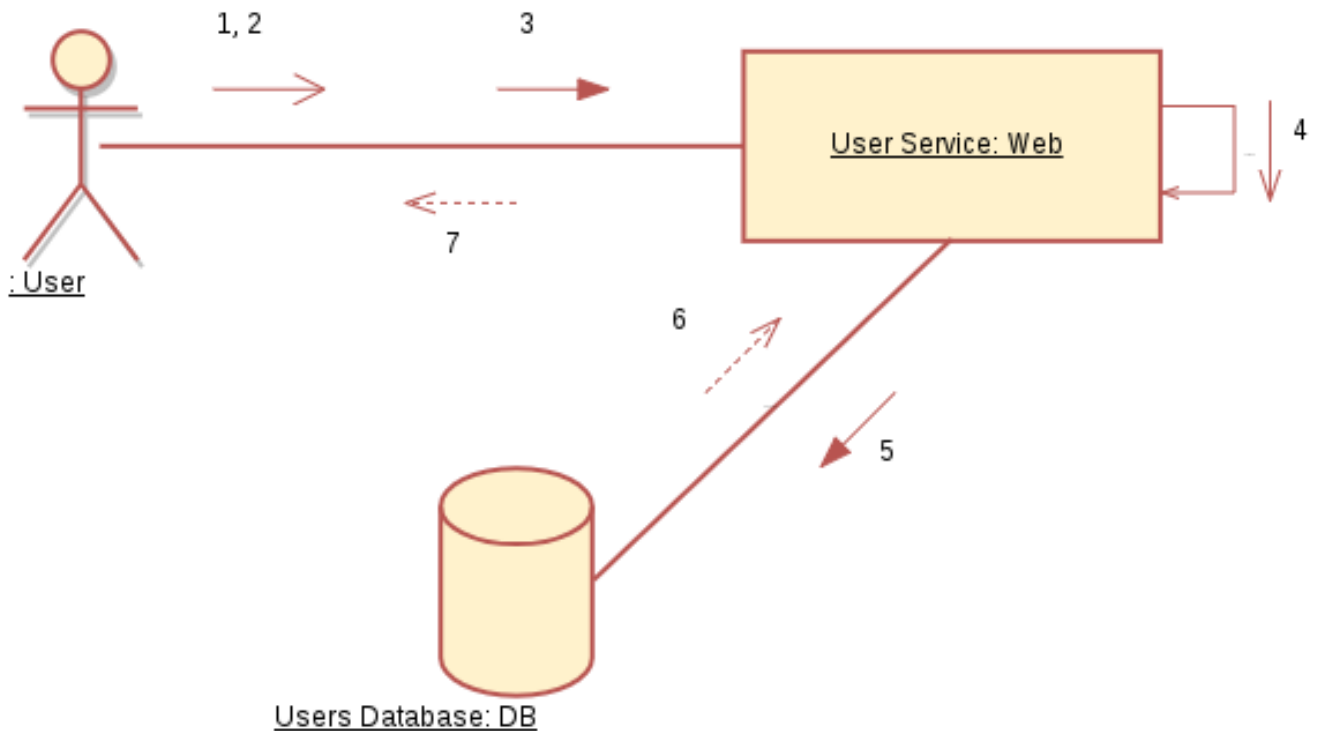


Figure 3.1– Login collaboration diagram

Here we can see the interaction between objects in our project. These diagrams are the equivalent of the previous sequence diagrams.

1 - input username; 2 - input password; 3 - click login; 4 - check input; 5 - query the user; 6 - success; 7 - authentication successful;

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

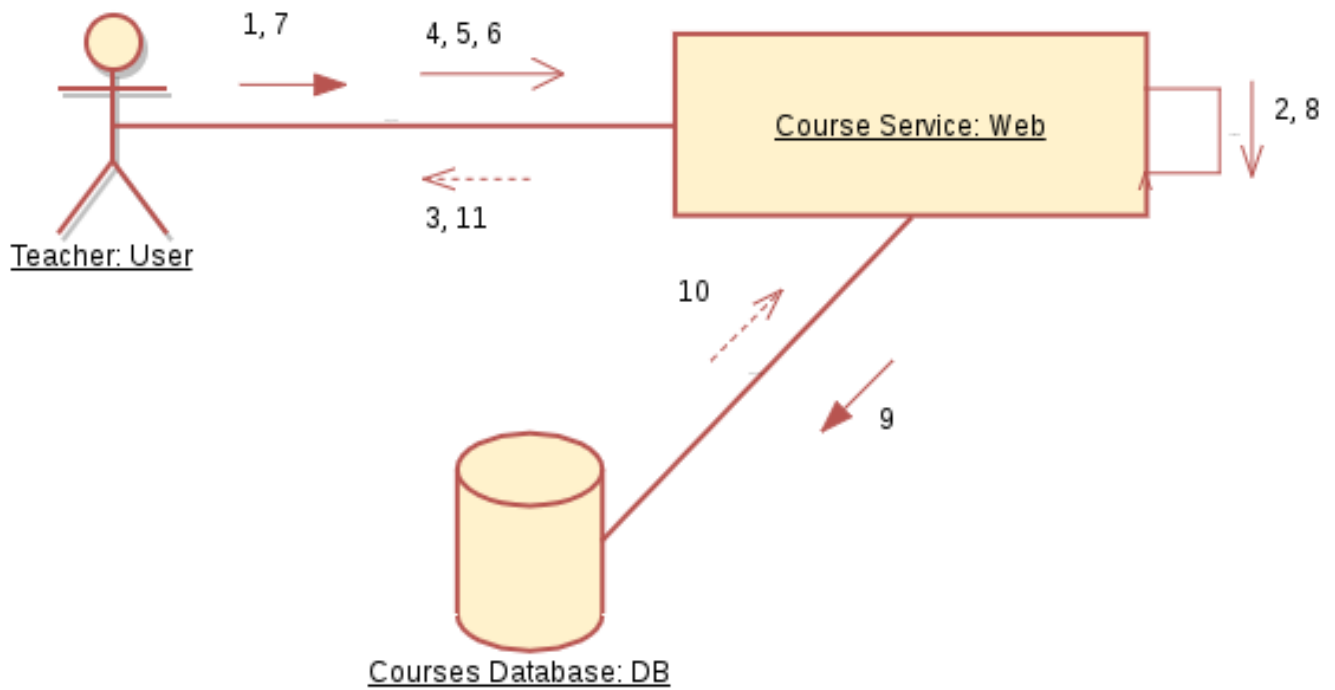


Figure 3.2– New course collaboration diagram

1 - click [New Course]; 2 - check authorization; 3 - success; 4 - input title; 5 - input description; 6 - upload content; 7 - click [Submit]; 8 - chek input; 9 - add new course; 10 - success; 11 - new course created;

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

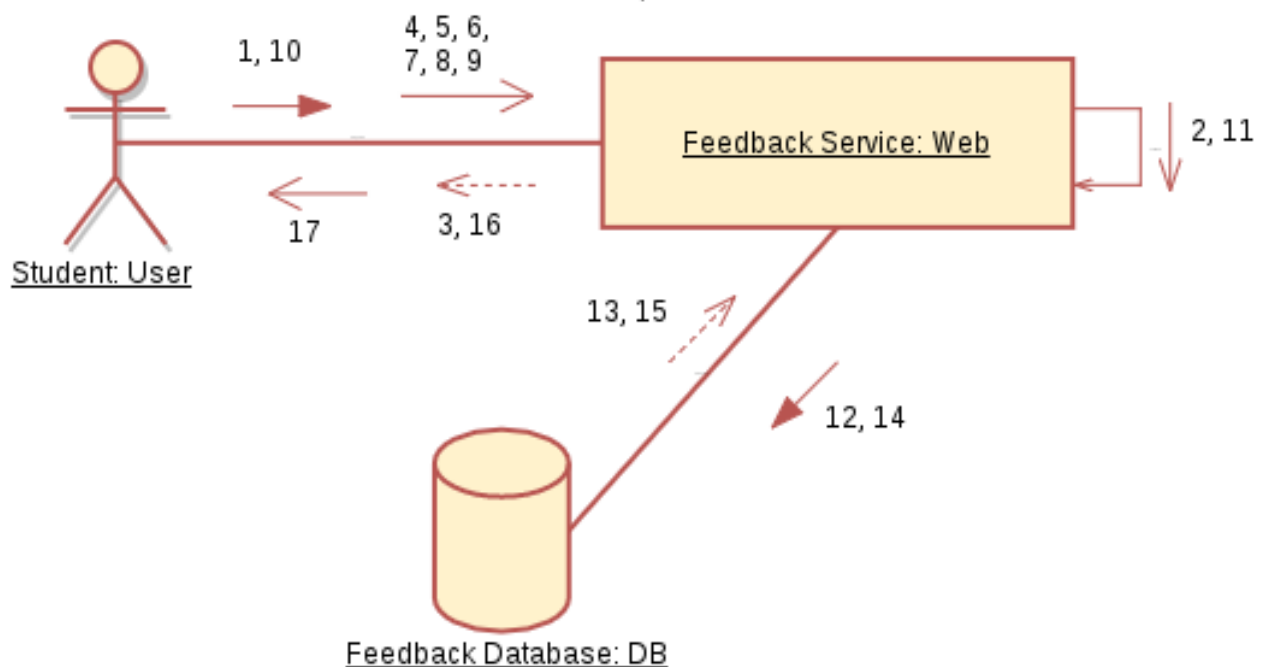


Figure 3.3– Feedback collaboration diagram

1 - click [Give Feedback]; 2 - check if user took this course; 3 - success; 4 - evaluate by lectures; 5 - evaluate by labs; 6 - evaluate by teacher competence; 7 - grade the course(0-10); 8 - additional notes; 9 - suggestions; 10 - click [Submit]; 11 - chek fields; 12 - add feedback; 13 - success; 14 - calculate

average mark; 15 - success; 16 - feedback submitted; 17 - average mark;

4 MVC Pattern

Model–view–controller is commonly used for developing software that divides an application into three interconnected parts. This is done to separate internal representations of information from the ways information is presented to and accepted from the user. The MVC design pattern decouples these major components allowing for efficient code reuse and parallel development.

Traditionally used for desktop graphical user interfaces (GUIs), this architecture has become popular for designing web applications and even mobile, desktop and other clients. Popular programming languages like Java, C#, Ruby, PHP and others have popular MVC frameworks that are currently being used in web application development straight out of the box.

5 Conclusion

In this laboratory work we learned to create Collaboration Diagrams.