

# BDSA-Assignment03

ehel, olfw, lawu

September 2022

Link to GitHub: <https://github.com/Lignio/assignment-03>

## Exercise 1

### What level of detail should UML models have?

The detail of UML diagrams depends on their usage but it is generally somewhat low. It should not be cluttered and contain a sufficient amount of detail, depending on the circumstances. Eg. a class diagram in the problem domain should be relatively low on detail.

### What is the difference between structure diagrams and behavior diagrams in UML?

Structure diagrams display the organization of a system in terms of the components that make up that system and their relationships.

Examples of structure diagrams:

Class diagram - shows the structure of how classes in the system are associated and connected/not connected.

Object diagram - shows an instance of a class diagram and at specific time.

Behavior diagrams are used to describe individual aspects of a system, eg. states or use-cases. Behavior diagrams are models of the dynamic behavior of a system as it is executing. They show what happens or what is supposed to happen when a system responds to a stimulus from its environment.

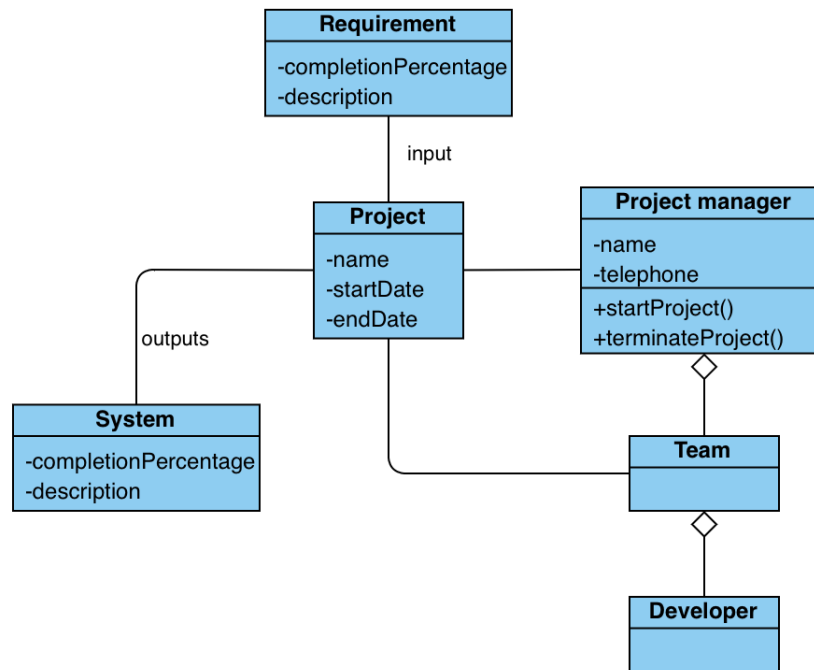
Examples of behavior diagrams:

Sequence diagram - Show the interactions between objects in the sequential order that those interactions occur.

Activity diagram - Portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed.

## Exercise 2

Draw a UML class diagram that models the following specifications:



## Exercise 3

Draw a UML state diagram that models your GitHub action configuration. Include all triggers that you have defined:

Description:

We decided to describe the Github Action configuration with three possible states, divided into two systems - DotNet and Git.

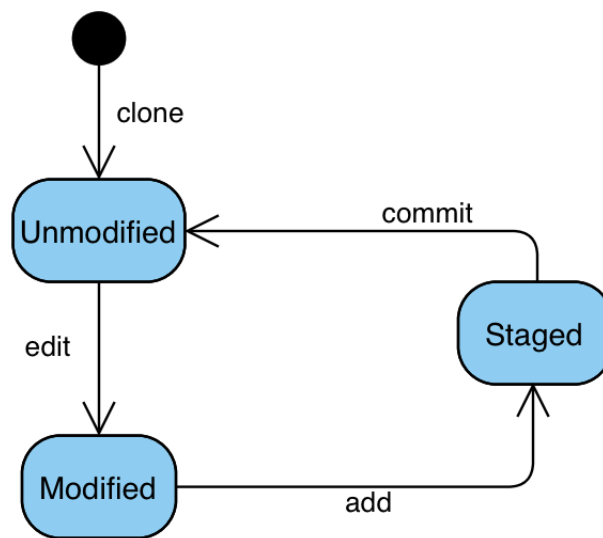
The states are: *(Dotnet)Unbuild*, *(Dotnet)Build* and *(Git)OnGit*. Besides the states there are a number of triggers that result in a state change:

- Push: Pushes the Build project to git.
- Pull: Pulls the project from Git to the local machine to the unbuild state.
- Build: Changes the state from Unbuild to Build (assuming it is successful).

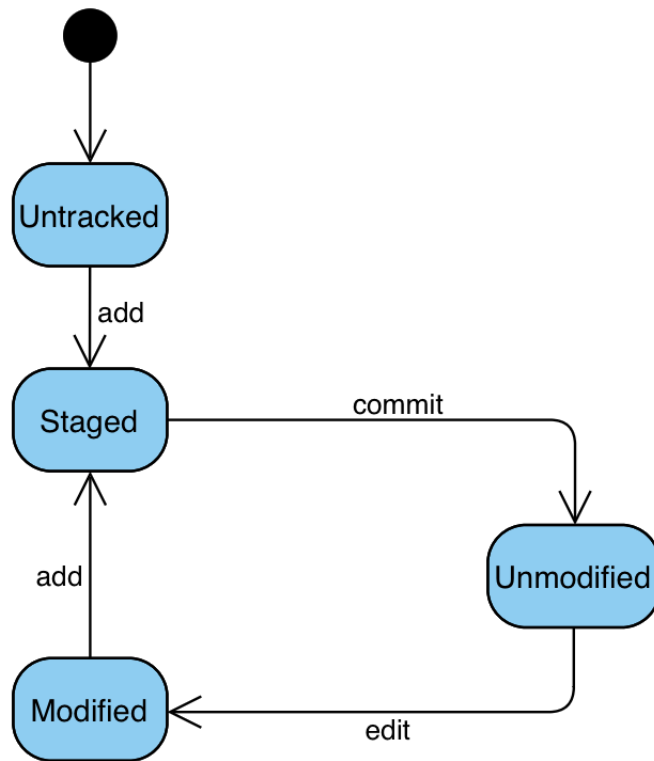
- Restore: Also changes the state from Unbuild to Build.
- Changes: To the project changes the state from Build to Unbuild.
- Testing: Tests the actual program functions, and hence does not change the current state of the system.

## Exercise 4

Let the first state diagram start with cloning a remote repository containing a file that is then edited:

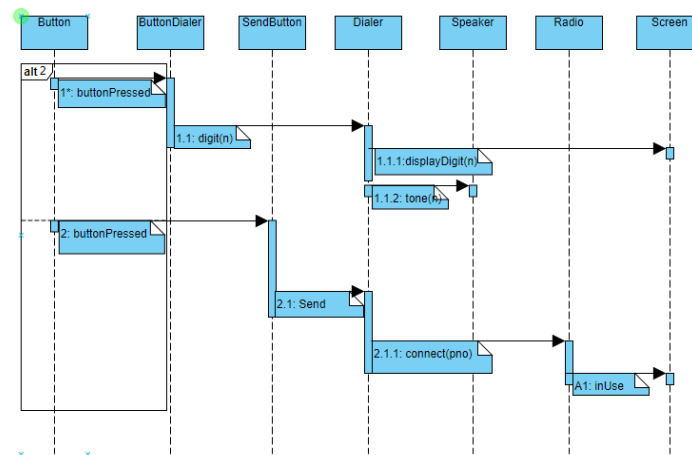


The second state diagram has to illustrate the states of a file that is newly created in a Git repository:



## Exercise 5

Translate the UML collaboration diagram (Fig. 14-14 from APPP), see below into a sequence diagram.



## Exercise 6

Draw a UML class diagram that models the structural information given in Fig. 5.7 from SE:

