

Course code: **CI/JENKINS**

Course title: **Continuous Integration using a Jenkins server**

Days: 2

## Description:

### Course intended for:

1. Programmers, team leaders and architects, who produce software in Java language in their everyday work.

### Course objective:

The training provides a comprehensive discussion of the concept of Continuous Integration (CI) and Continuous Delivery (CD), using the example of a Jenkins server. The participants, on the one hand, get familiar with the key concept of software production; on the other hand, they acquire the practical skills of installation, configuration and use of the Jenkins server. Thanks to the training, the participants will be able to identify repeatable tasks during the software production process and to automate these tasks to be able to focus on the core of their work in a creative manner.

### Course strengths:

After the training, the participants will have the following skills:

1. understanding of the concept of Continuous Integration and Continuous Delivery,
2. ability to eliminate the painstaking process of integration of work of many teams at the end of the project and the so-called integration hell
3. install and configure the Jenkins server,
4. configure automatic application-building tasks,
5. configure automatic component testing,
6. configure automatic integration,
7. implement the CI/CD concept in their company.



## Requirements:

The training participants are expected to know Java language.

The ability to use such tools as git, SVN, Maven, Selenium, JMeter and Soap UI will be an additional advantage.

## Course parameters:

2\*8h in the following proportion: 80% workshops, discussions; 20% lectures.

## Course curriculum:

### 1. Project risks.

- I. Low quality of source code.
- II. Code not ready for implementation
- III. Late error detection
- IV. Merging of code changes by several teams.

### 2. Introduction to CI / CD.

- I. Continuous building.
- II. Continuous testing.
- III. Continuous inspecting.
- IV. Continuous implementing.

### 3. Jenkins.

- I. Installation and configuration
- II. Continuous building
  - i. Task configuration.
  - ii. Plugins.
  - iii. Integration with Maven.



- iv. Integration with git / SVN.
- v. Finished build information.
- vi. HTML reports.
- vii. Remote task requests via Jenkins API.
- viii. Integration with Nexus.

## 4. Continuous testing.

- I. Integration tests, e.g. using Jetta.
- II. Acceptance tests, e.g. Fitnesse.
- III. Tests using a browser, e.g. Selenium.
- IV. Performance tests, e.g. JMeter.
- V. Web Service tests, e.g. SoapUI.

## 5. Continuous inspecting

- I. Checking of code coverage with tests.
- II. PMD rules.
- III. Detection and presentation of errors using FindBugs.
- IV. Integration with Sonar.

## 6. Continuous implementations

- I. Continuous implementations.
- II. Implementation policy.
- III. Versioning.
- IV. Release plugin.

## 7. Administration.

## 8. Other CI servers.



## 9. How to implement CI/CD in your company?

