

Late-term Assignment

Purpose

Learn to set up technical infrastructure to support agile development. The main challenge is to glue together the bits and pieces through the latter half of this class, into a coherent infrastructure.

Description

Each team should implement the game TicTacToe. The main focus is on the infrastructure and best coding practices described below but not on implemented features.

The project and infrastructure should tackle the following:

1. The code base is stored in source control system on GitHub from early start of the project (the teacher should be able to track the history of the project from the creation of the very first file).
2. The system is setup with automatic build that must run all unit tests.
3. The output is a deployable or executable artifact, such as a jar/war or equivalent.
4. The build script has deploy/install target that deploys the executable outside of the working directory or on external server.
5. The business logic should be coded using Test Driven Development.
6. The code should be loosely coupled and follow good object oriented design practices.
7. It is possible to get extra points for the following
 1. More build targets (0,5 point per item, never more than 1 for this part)
 - Code coverages
 - Code inspection
 - Integrate database
 2. Run focused integration or End-to-end tests (e.g. Selenium). It is

necessary to run this on external staging server (e.g. Heroku). To run integration test the application needs to connect to external system e.g. database. To run end-to-end test the system needs to have GUI (e.g. webUI). (1 point)

3. Use Automated Continuous Integration Server (e.g. Travis). (1 point)
4. Continuous Delivery. For each code change the CI server runs these steps:
 - Run build script (all targets)
 - Run unit and integration tests (if you have integration tests)
 - Deploy on staging and run End-to-end tests.
 - Deploy to production server.
 - Only continue next step if previous step is success
 - Generate feedback to developer
5. Something else ... Surprise me !!!

Grade

Solving parts 1-6 **perfectly** will together give a grade of 6, it's ok to have all targets in build script manual. In order to get a 10, you'll need to solve at least 4 extra points.

Return form

1. URL of root of the project in the source control system
2. Development manual, that is, what is needed and how to get the project to build on a fresh machine
 - Source control client and access to source control
 - Build environment
 - Other necessary dependencies
3. Administration manual, how to set it up and get it to run, also on a fresh machine.
 - How to install and run the program on clients machine (in case of simple desktop program)
 - How to deploy, run and maintain in case of client/server or web application.

4. Design report
 - Document describing initial design

Demonstration

The team will have a 15 minutes demonstration of the technical infrastructure and the simple functionality developed. The demonstrations will be held in the last week, the detailed schedule will be published later. The distributed groups will demonstrate using shared desktop. The demonstration is part of the grade for this assignment. **If you are not able to show some part of the build process (if it is broken) in the demonstration it will not count in the grade.**