# Teamwork Project Assignment for the [QA Automation Course @ SoftUni](https://softuni.bg/trainings/1584/qa-automation-march-2017)

This teamwork project assignment is designed to create overall skills for software testing as well as for easy integration in a software development process.

## Project Description

You will be given a little **web project**, which you will have to test. The project is a basic **blog** with very little functionality, but don’t worry, this is just **version 0.1**,and we have a lot to do before the application goes in production. Upon seeing the project, you should create the idea of how a real blog looks, and you should prepare **Software Requirements Specification (SRS)**. Then you will have to prepare a **test plan** – how the **test process** will go and what **test techniques** will be used. You can prepare **test scenarios** with **test cases** in them. The real job comes here – you need to **build a test framework with Selenium WebDriver**, which you will have to **merge** with **the blog** in **Continuous Integration project via TeamCity**. You need to fully **automate the UI testing process**, because at the end of the course you will be given **version 0.2** of **the blog**. Upon committing changes in **the blog**, the **test project** has to automatically test the old functionalities, **find bugs and report them**, if there are any.

## General Requirements

* Register a GitHub account (<http://github.com>)
* Write and assign tasks in ‘**GitHub** **issues**’ for each tasks, e.g.:
  + Writing the SRS – download template!
  + Writing a test plan – create test suites with priorities! TestLink
  + Write test scenarios with test cases - TestLink
  + Start automation with Selenium WebDriver test framework
  + Build Continuous Integration with TeamCity ??
  + Make full test automation for the blog
* **Software Requirements Specification** – the **description** of each functionality and use case of the project. Use a template for an SRS from the Internet.
* **Test project** - **covers everything** in the project. Use template for Test scenarios and Test cases from the Internet.
* **Test framework** – has to be **similar** to **the one** built **throughout** the **course**. You are free to improve the framework from the course, if you find it useful to do so.
* **Continuous Integration** – upon **commit changes** to the code, the **test project** should run **all tests** on the new version of the project and **report** any **bugs found**.

The **test project** should be easily extended. When the **new functionality** comes, the **new tests** for it should be **easily added** to the **project**.

## Additional (optional) Requirements

* You can update your blog project with new functionalities and build new versions for it.
* You can simulate a **Project Testing Life Cycle**, by making a **project backlog** and **daily meetings**.
* You can implement **separate roles** in the **development** of the **software project**.

## Assessment Criteria

Each team will have to deliver a **public defense** of its work in front of the other students, trainers and assistants. Each team will have **only 15 minutes** for the following:

* **Demonstrate** a test project, by submitting a new code in the blog
* Add a new test to framework
* Explain how each team member has **contributed**: display the commit logs in the **Source Control system** you have been using.
* Optionally, you might prepare a **presentation** (**3**-**4** slides).

Please be **strict in timing**! On the 15th minute you **will be interrupted**! It is good idea to leave **the last 2-3 minutes for questions** from the other students, trainers and assistants.

Be **well prepared** for presenting maximum of your work for minimum time. Bring your own laptop. Test it preliminary with the multimedia projector. Open the project assets beforehand to save time.

## Assessment Criteria

* **SRS, test plane, Test scenarios: 0 - 10 points**
* **Test Framework:**  **0 - 20 points**
* **Continuous Integration: 0 - 20 points**
* **Automate Bug Reports: 0 - 5 points**
* **Teamwork**\*: **0 - 15 points**
* **Bonus** (bonus points are given for exceptional project functionality) – **0 -10 points**

\* Each team member should have **commits** in **5 different days**. Even if not all team members have contributed to the project, the **teamwork** points will not be **affected**.

## Give Feedback about Your Teammates

You will be invited to **provide feedback** about all your teammates, their attitude to this project, their technical skills, their team working skills, their contribution to the project, etc. The feedback is important part of the project evaluation so **take it seriously** and be honest.