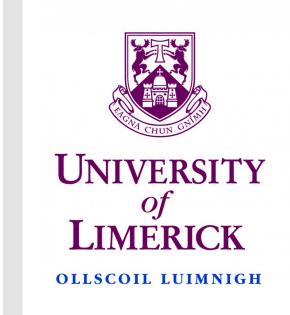
CS4004: Software Testing

JIRA & JUnit 5+ Eclipse 14th September 2021



Dr. Salim Saay Amandeep Singh

Toolsets - Introduction

JIRA

JUnit 5 + Eclipse

JIRA

www.atlassian.com/software/jira

- Project management tool designed to optimize project planning, implementation and tracking.
- Used as an issue tracking tool for software testing.
- Can handle variety of issues progress of a project, hiring of employees, product pipelines, building stories for software teams, etc.
- Free for teams of up to 10 users.

JIRA

www.atlassian.com/software/jira

- Plan Create user stories and issues, plan sprints, and distribute tasks across the software team.
- Track Prioritize and discuss each member's work in full context with complete visibility.
- Release Deploy product with up-to-date information.
- Report Analyse team performance in real-time using visual data.

- Unit testing framework for Java.
- Unit testing the process of checking the smallest independent modules to ensure they are working as required.
- Unit testing Manual/Automated
- Manual testing Executing test cases manually; nonprogrammable; less reliable.
- Automated testing Executing test cases using coding tools; programmable; more reliable.
- JUnit used for automation of unit testing.

- Open source framework
- Annotations and assertions identify test methods & compare against expected results.
- Test runners
- Runs automatically and provides immediate feedback.

Toolsets -Installation

JIRA

JUnit 5 + Eclipse

JIRA

www.atlassian.com/software/jira

 Create free online account on: <u>https://www.atlassian.com/try/cloud/signup?bundle=jira-software&edition=free</u>

- Helpful link1: <u>www.educative.io/courses/java-unit-testing-with-junit-5/B892KY261z2</u>
- Helpful link2: www.tutorialspoint.com/junit/junit_environment_setup.htm
- Helpful link3 (Windows): <u>www.softwaretestinghelp.com/download-and-install-junit/</u>
- Helpful link4 (MacOS):
 https://stackoverflow.com/questions/21369953/need-help-installing-junit-on-mac-how-to-add-junit-to-path-environmental-variabl/26977630
- System Requirements JDK and Eclipse
- Download the .jar files from this link: https://github.com/junit-team/junit4/wiki/Download-and-Install
- Add the location of these .jar files to the Environment Path variables (see link3/link4).

- Open Eclipse and create a new Java project 'JUnitTesting'.
- Create a new package inside the src folder '<name>.junittest'.
- Create a new class inside the '<name>.junittest' package 'Calculator'.

```
public class Calculator {
    public int add(int a, int b) {
        return a + b;
    }

    public int subtract(int a, int b) {
        return a - b;
    }
}
```

- Right click on the project > Build Path > New Source Folder 'test'.
- Right click on 'Calculator.java' in the src folder > New > JUnit Test
 Case > Select "New Junit Jupiter test" > Choose Source Folder as
 'test' folder > Provide class name as 'CalculatorTest' > Click 'Next'.
- Choose all methods available under 'Calculator'.
- 'Add JUnit 5 Library to build path' > 'OK'

```
class CalculatorTest {
  @Test

public void testAdd() {
        Calculator calculator = new Calculator();
        int a = 5;
        int b = 4;
        int actual = calculator.add(a, b);
        int expected = 10;
        assertEquals(expected, actual, "Sum is not correct");
  }
  @Test
  void testSubtract() {
  fail("Not yet implemented");}
}
```

- Save and run 'CalculatorTest.java'.
- Look at the JUnit report to confirm if test was successful.