



## GS04: Tools and Environments

### Lab Session 9: Acceptance testing with FIT

The aim of this lab session is to gain experience with integration testing using FIT. We will define and execute acceptance tests in a way that they are meaningful to users for the UCAS application web site. To achieve that we will use a high-level domain specific language rather than Java (as in JUnit tests).

#### Setup

Check out the trunk configuration of Project FitLab from our subversion repository. All libraries it depends on are included in the 'lib' directory.

The project includes a spreadsheet in folder 'tests' that defines all acceptance tests. The spreadsheet uses a domain specific language to define the acceptance tests in a manner that non-programmers can understand what is being tested. This is critical to enable them to define and review the tests.

#### Implement the DSL for the acceptance tests

You now need to implement the DSL that interprets these acceptance tests. You do this by extending class ColumnFixture defined in the fitnesse library. Define fields for all the input columns and methods for all the assertion columns (the ones ending with a '?'). You will want to implement the methods using httpunit that we have used in Lab session 8. For a quick tutorial on Fit and its use of Fixtures, including column fixtures refer to <http://fitnesse.org/FitNesse.FixtureCode>

#### Executing Acceptance Tests

In order to execute the unit test you might want to use class FolderRunner, which performs a recursive descent into a file system directory, applies all tests stored in the directory and any subdirectories and generates a HTML page hierarchy with the results.

The main method in FolderRunner takes two arguments. The first one is the folder that contains all the tests and the second is the folder where the results should be generated.