Software Quality and Testing Computer Lab 3

Programming Assignment

In this assignment you will implement a verifier for Sudoku game. The task description and the link to Github resository is available on SQAT Lab web page.
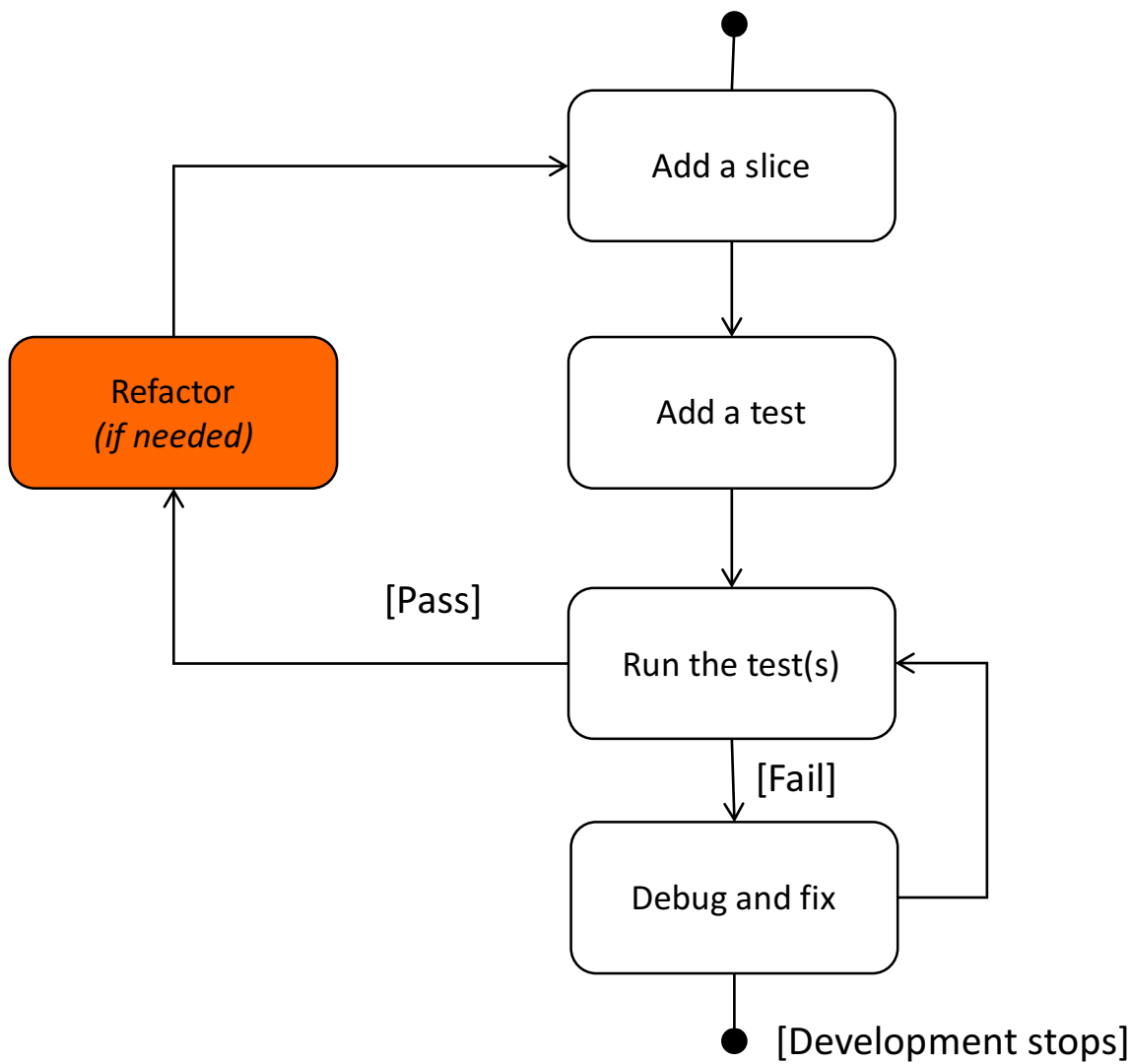
Instructions:

Follow these instructions carefully step by step.

Get the assignment files:

- Locate the repository: Go to https://github.com/M3SOulu/SQAT_SudokuVerifier
- Create a fork of this repository
- Import into Eclipse
    - In Eclipse: Select File > Import
    - In Import Window: Select Git > Projects from Git
    - In Import Projects from Git Window: Select Clone URI
    - In the next window:
        - URI: paste the URL for your repository The URL of your repo is available at your repository's web page on Github under HTTPS clone URL on the right-hand side of the page. Link should be https://github.com/[YourGithubaccountName]/SQAT_SudokuVerifier.git
        - In the Branch Selection window select master branch
        - Proceed with the default options
    - Open SudokuVerifier.java file

Open Besouro Plugin:

- In Eclipse: Select Window > Show View > Other
- In Show View Dialog: Select Other > episodeView
- In episodeView Window: Click green play button to start recording

```mermaid
stateDiagram-v2
    [*] --> Add_a_slice: Add a slice
    Add_a_slice --> Add_a_test: Add a test
    Add_a_test --> Run_the_tests: Run the test(s)
    Run_the_tests --> Refactor: [Pass]
    Refactor --> Add_a_slice
    Run_the_tests --> Debug_and_fix: [Fail]
    Debug_and_fix --> Run_the_tests
    Debug_and_fix --> [*]: [Development stops]
```

Add a slice

Add a test

Run the test(s)

[Pass]

Refactor *(if needed)*

[Fail]

Debug and fix

[Development stops]

<u>**Implementation**</u>

- Implement the task according to the requirements described in task description following **Iterative Test-last Development** methodology.
   - o Focus on slicing the requirements into small user-stories and implement a user-story in each cycle.
   - o Write unit tests immediately after you complete a user-story

<u>**Submit your work**</u>

- In episodeView window> Click red stop button to stop recording
- Commit and Push:
   - o In Eclipse: Right-click project in the package explorer. Then Select **Team > Commit**
   - o Write "Project Finished" in commit message textbox
   - o In Commit Changes window click **Commit and Push**

- On your Github repository page check the status to verify that it is up to date with your local copy
- On your Github repository page check last commit message is "assignment finished"
 Create Pull Request
  - On your Github repository page create a pull request
    - Click New Pull Request button located above the listing of repository files

NOTE:

If you encounter an error "There are no staged files" during commit, select all the files in *Unstaged changes* area and move them to *Staged area* in commit window.

You can disable staging view:

 Open Window > Preferences
 In preferences window open Team>Git>Commiting
 Uncheck "Use staging view …"