Table of Contents

Introduction

ReqTracker is an innovative software development tool designed for software engineering companies worldwide, which need to track requirements through the entire software life cycle. ReqTracker provides a simple and intuitive way of analyzing and reporting the up-to-date impact of requirements or changes to your software. ReqTracker can be used both as a stand-alone solution and enabled in your IDE so that you get live feedback while you are working in your familiar work environment. ReqTracker is designed for being used by software developers all over the world on a daily basis to improve their workflows.

Stand-alone application

Configuration

The video manual for Configuration & Login can be viewed at http://uv48uson.github.io/amos-ss15-proj3/artifacts/WebDocs/VideoTutorials/Login.htm

To use the application, the following are required

- Path of the repository, which contains the source code files of the project.
- The credentials of the database, which stores the information about additional user defined linkages between requirement and source code is required

To start the application, run the ReqTracker.jar file. In the configuration screen (Figure 1), enter the repository path, click the Choose Repository button and chose the folder where source code files are stored. (Example: “D:\ReqTrackerPoject\.git”)

Enter the Requirement Pattern, which is the pattern of requirement numbering, used for linking the requirements and commits. If multiple patterns are required, provide them by separating the patterns with the “|” symbol.

To optimize the performance of the application, select the Enable Indexing checkbox, which enables loading and caching of all data about linked requirements right after the application start. Indexing takes a few minutes after which the application’s performance is improved.

Enter the database credentials to connect to the database, which stores the requirements-source code linkage information.

Click the Configure button to login into the application. A dialog box pops up to get user approval to store the configuration details which will be used to automatically fill in the configuration fields during the next login. When the user approves this request, the configuration details are fetched from the stored values and are filled up during the next application startup. If there is already a configuration stored, this will be over written by the new configuration. If the user denies the request, the configuration details are not stored.

- amos-ss15-proj3

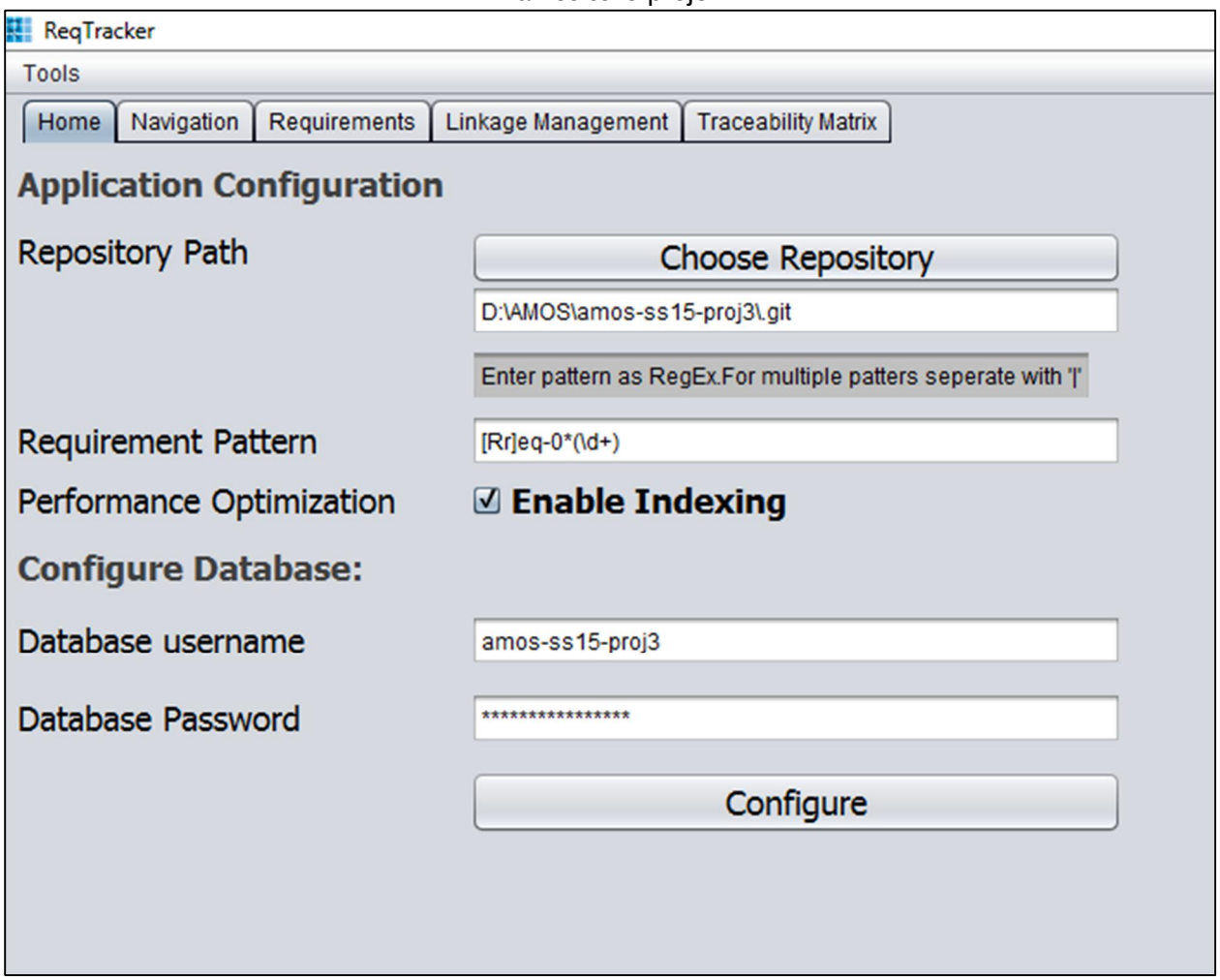

Figure 1. Configuration screen

After the login and once inside the application, the user has the possibility to change the requirement pattern or the repository path. For this, one has to go to the Tools menu and click Configure, and then have to choose the required option from the dialog box (Figure 2).

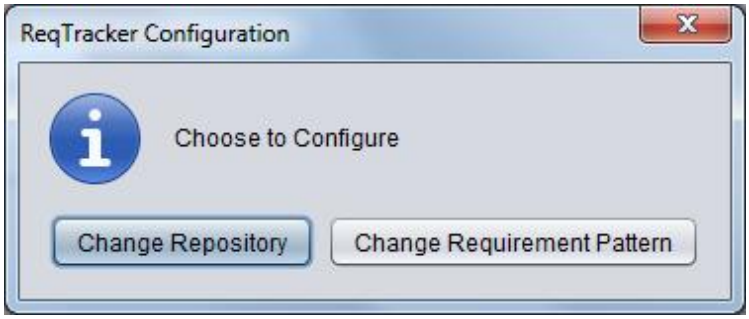

Figure 2. Reconfigure dialog

- amos-ss15-proj3

The Functionalities

The 'Navigation' tab

The video manual for Navigation tab can be viewed at http://uv48uson.github.io/amos-ss15-proj3/artifacts/WebDocs/VideoTutorials/ReqTrackerApplication-Navigation.htm

The Navigation tab (Figure 3) displays information about all requirements, related to them commits and source code files.

To view the information about the source code lines that are influenced by a requirement, in the RequirementID pane click the requirement number or enter the requirement number in the RequirementID textbox and press 'Enter'. The Commit pane displays the list of commits, which were commited because of the chosen requirement. It also lists all the files (in the Files pane) that were influenced by the requirement over multiple commits

In the Commit window, click the commit title to view the files which were influenced by the chosen commit. In the Files window, the list of files, related to the selected requirement and the commit are displayed. On the right of the file title, the percent of code lines (impact percentage), influenced by the chosen requirement, is displayed.

Upon clicking the file title in the Files window, the Code window displays the file content with all related requirement numbers related to each line of the file shown in the ## window. Code lines, which are related to the chosen requirement are displayed in red.

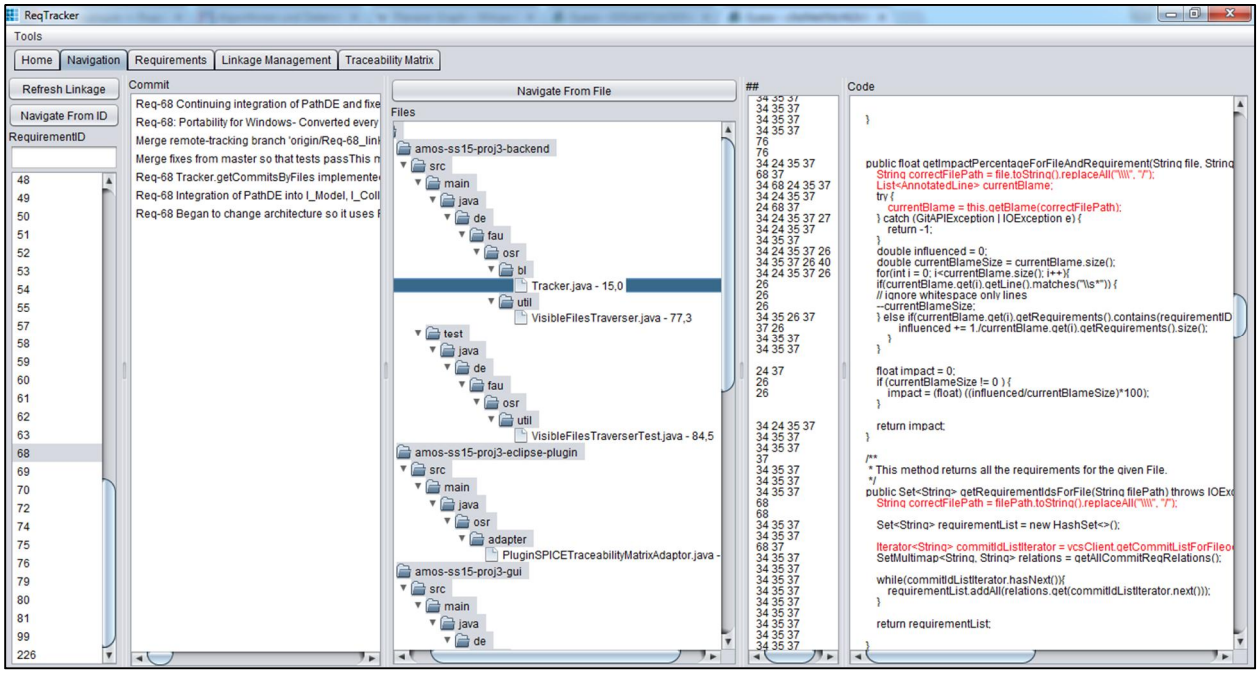

Figure 3. Navigation tab – Navigate from ID

To review the list of source code files and requirement numbers, which affected the file, click the Navigate From File (Figure 3) button, then click the required file title.

- amos-ss15-proj3

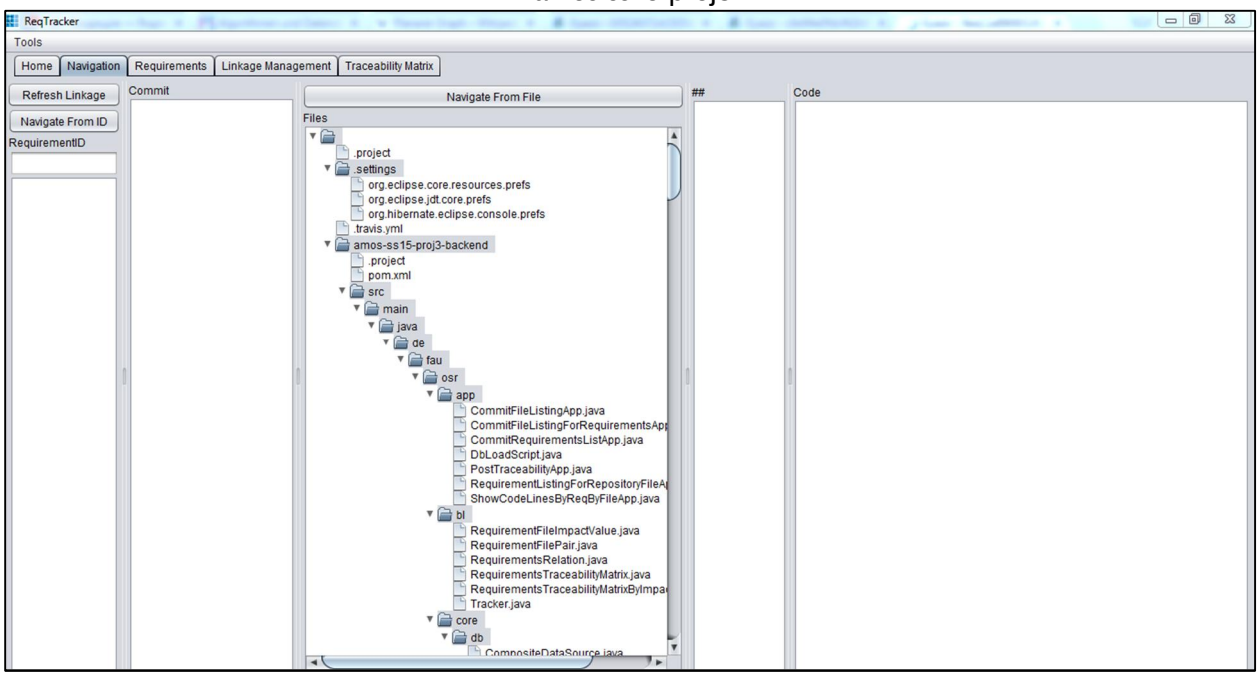

Figure 3. Navigation tab – Navigate from Files

Click the Refresh Linkage button to refresh the Navigation tab.

The 'Requirements' tab

The Requirements tab (Figure 4) provides the information about the requirement title and the description of the requirement.

To get the information about the requirement, in the RequirementID window click the requirement or enter the requirement number in the RequirementID textbox and click the Navigate From ID button.

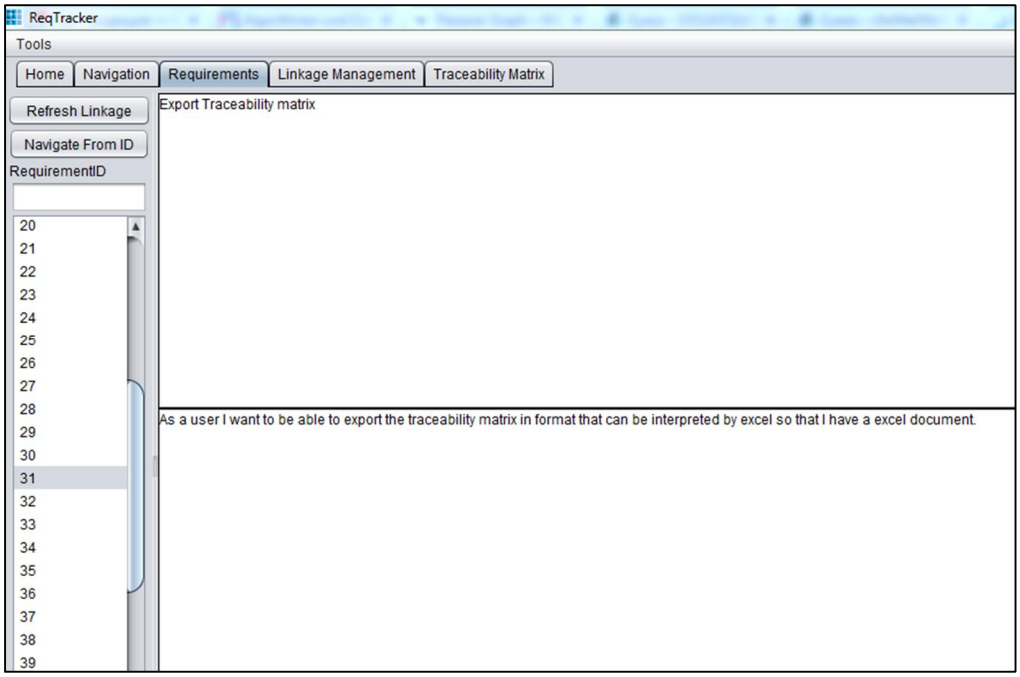

Figure 4. Requirements Tab

- amos-ss15-proj3

The 'Linkage Management' Tab

The video manual for Linkage tab can be viewed at http://uv48uson.github.io/amos-ss15-proj3/artifacts/WebDocs/VideoTutorials/Linkage.htm

The Linkage Management tab (Figure 5) provides the ability to link requirements to commits manually.

To link a particular requirement with a commit, in the RequirementID window click the requirement or enter the requirement number in the RequirementID textbox and then click the appropriate Commit that has to be linked with the requiremend (or) enter the commit id the Commit ID text box. Thereafter, click the Add Linkage button to save the new mappings.

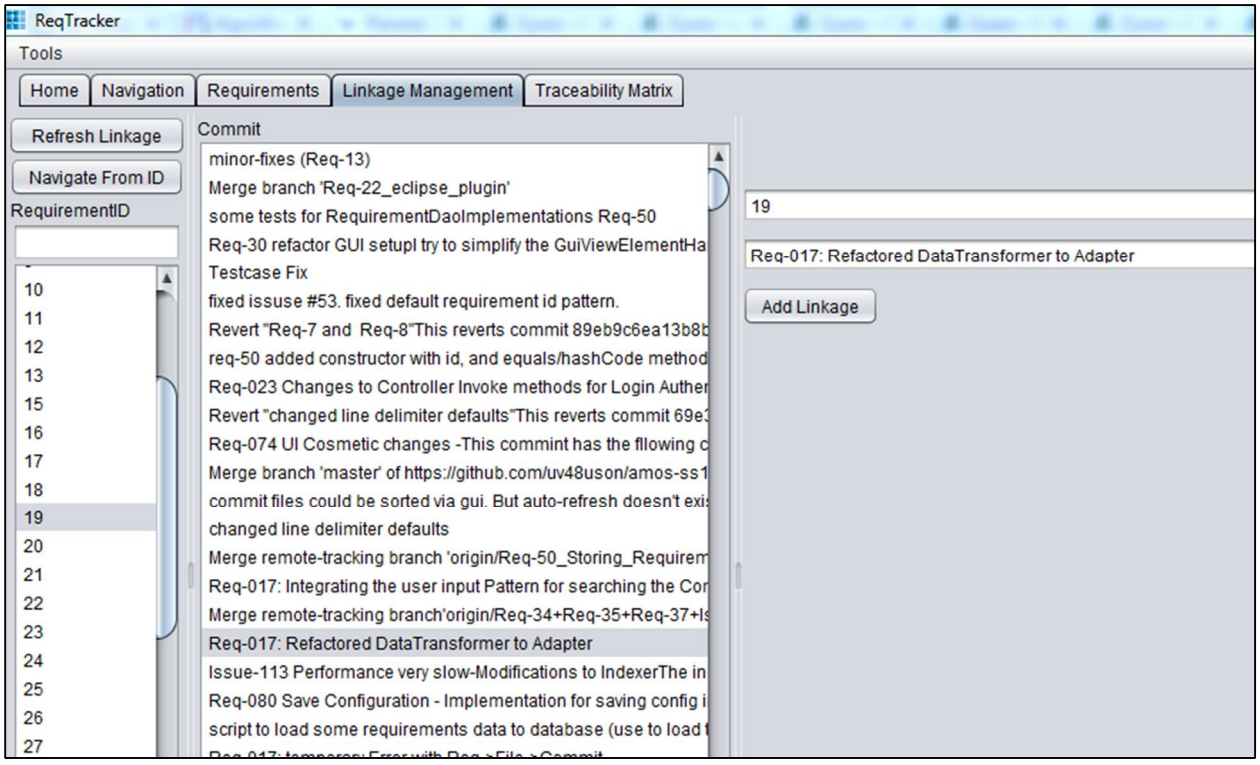

Figure 5. Linkage Management Tab

The 'Traceability Matrix' tab

The video manual for Traceability Matrix tab can be viewed at http://uv48uson.github.io/amos-ss15-proj3/artifacts/WebDocs/VideoTutorials/TraceabilityMatrix.htm

Traceability matrix is a table that shows M:N relations between requirements and source code files to visualize their relationship. Every cell in the matrix represents the impact percentage of a requirement (shown in columns) on a particular file (shown in rows).

To view the traceability matrix, on the Traceability Matrix tab (Figure 6), click the Generate Traceability Matrix button to create the latest state of the matrix.

To save the traceability matrix as a CVS file or as a PDF file, click the To CVS or To PDF button respectively.

- amos-ss15-proj3

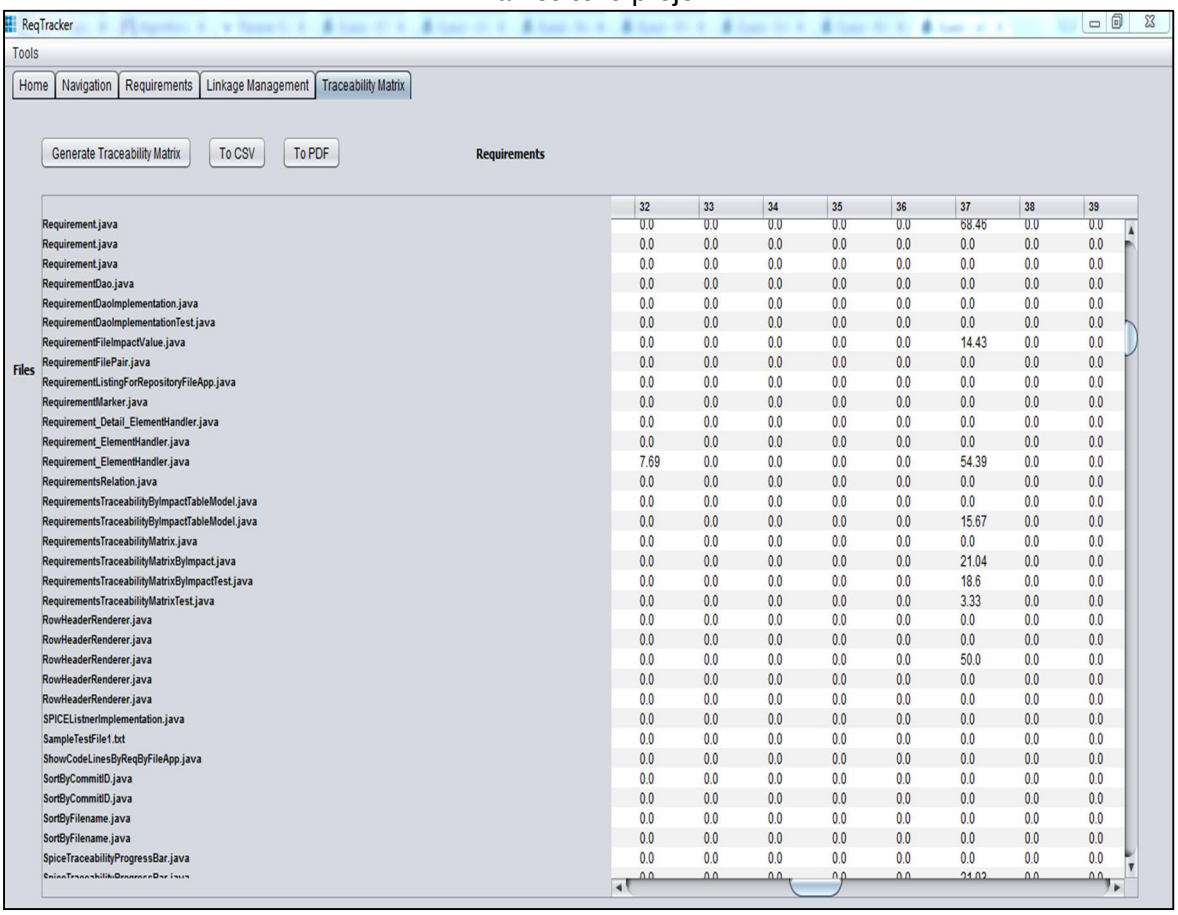

Figure 6. Traceability matrix tab

Eclipse IDE plug-in

Plugin Installation

The video manual for Eclipse plugin installation can be viewed at http://uv48uson.github.io/amos-ss15-proj3/artifacts/WebDocs/VideoTutorials/EclipseInstallation.htm

The ReqTracker plugin for Eclipse can be installed by following the steps listed below

1. Go to 'Help' menu in Eclipse tool bar
2. Click the menu item 'Install New Software
3. In the 'Add' window that pops up, click the 'Add' button.
4. In the 'Add Repository' dialog box that shows up, click the 'Archive' button and point to the file 'amos-ss15-proj3-eclipse-updatesite-1.0.0.zip' distributed with the ReqTracker product package and click 'Ok'.
5. Check the item 'ReqTracker Eclipse IDE Integration' that gets added to the list and click 'Next'
6. Click 'Next' from 'Install Details' window
7. In the 'Review Licences' window click the 'I accept the terms of the licence agreement' option and click 'Next'
8. If a 'Security Warning' dialog pops up, click 'OK'
9. Now the plugin will get installed
10. Eclipse will prompt to restart the application for successful installation.
11. Accept the Restart and when Eclipse restarts the ReqTracker can be seen as a menu item. This shows that the ReqTracker plugin had successfully installed

- amos-ss15-proj3

Configuration

The video manual for Eclipse plugin configuration can be viewed at http://uv48uson.github.io/amos-ss15-proj3/artifacts/WebDocs/VideoTutorials/EclipseUsage.htm

To use the plugin, the following are required

- Path of the repository, which contains the source code files of the project.
- The credentials of the database, which stores the information about additional user defined linkages between requirement and source code is required

In the ReqTracker menu, click Enable ReqTracker. In the ReqTracker Configuration (Figure 7) window that pops up, enter the Git Repository URL where source code files are stored. Enter the Requirement Pattern, which is the pattern of requirement numbering, used for linking the requirements and commits. If multiple patterns are required, provide them by separating the patterns with the "|" symbol. Enter the database credentials to connect to the database, which stores the requirements-source code linkage information. Now click 'OK'. Now the login should happen. From now on until the application is closed (or) the ReqTracker is disabled, when a user opens a file inside the eclipse IDE that belongs to the folder/subfolders of the GIT repository path the lines will be marked with a marker which when hovered over by mouse pointer reveals the Requirement Ids that influenced the line in the file.

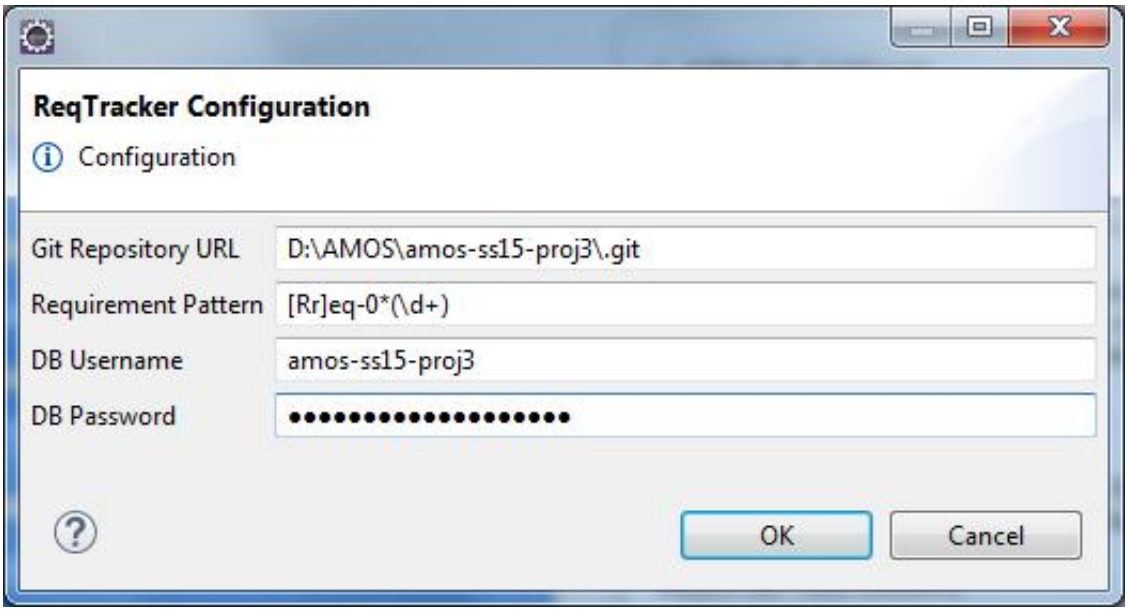

Figure 7. ReqTracker Configuration

Requirement ID – Code Line Marker

The video manual for Eclipse plugin usage can be viewed at http://uv48uson.github.io/amos-ss15-proj3/artifacts/WebDocs/VideoTutorials/EclipseUsage.htm

The plugin allows seeing the requirement number, which affected the code line, directly in IDE. To see the requirement(s) number, in the Navigator window of the IDE select the required source code file. When the source code file opens in the IDE, in the left marker area the icon can be seen. When mouse pointer is hovered over this icon, it shows the requirements that influenced these lines (Figure 8).

- amos-ss15-proj3

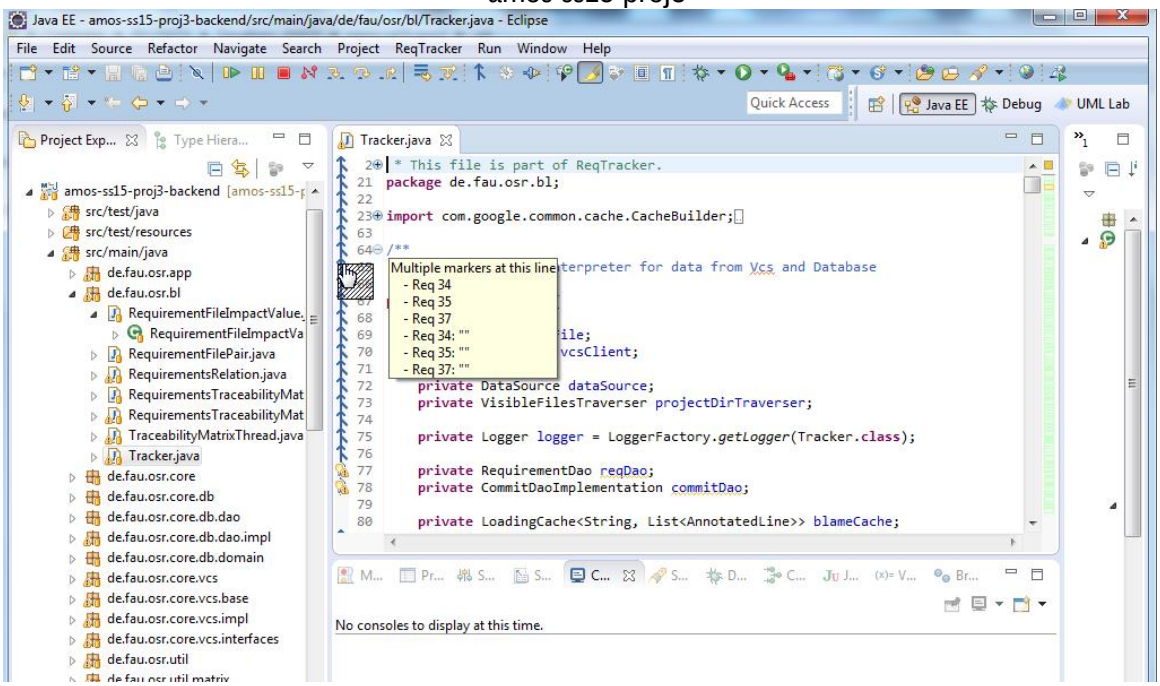

Figure 8. Requirement ID – Code line mapping

Traceability Matrix

The video manual for generating Traceability Matrix using Eclipse plugin of ReqTracker can be viewed at http://uv48uson.github.io/amos-ss15-proj3/artifacts/WebDocs/VideoTutorials/EclipseUsage.htm

Traceability matrix is a table that shows M:N relations between requirements and source code files to visualize their relationship. Every cell in the matrix represents the impact percentage of a requirement (shown in columns) on a particular file (shown in rows).

In the ReqTracker menu, click Traceability Matrix by Impact to create the latest state of the traceability matrix (Figure 9).

To save the traceability matrix as a CVS file or as a PDF file, click the To CVS or To PDF button respectively.

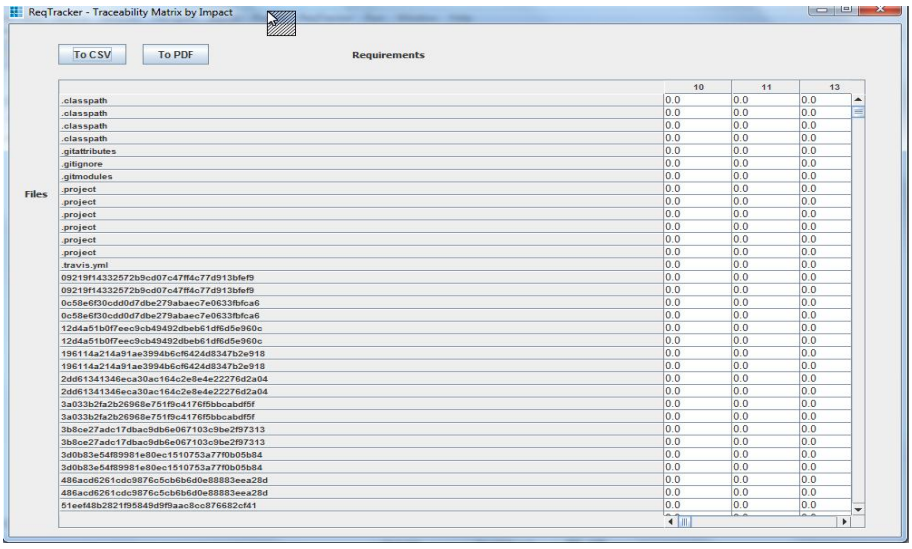

Figure 9. Traceability Matrix from ReqTracker Eclipse Plugin