

Functional Specification

Web based XACML Graphical User Interface Policy Editor

Lee Wolohan C00157339

Supervisor: Dr. Joseph Kehoe

Version 1.0

Date: 17th April 2015

Table of Contents

Functional Specification.....	1
Objectives.....	2
Architecture.....	3
FURPS+.....	4
Functionality.....	4
Usability.....	4
Reliability.....	4
Performance.....	4
Supportability.....	4
Test Cases.....	5
Code Editor (create/save).....	5
Code Editor (open/amend).....	5
User Interface (radio buttons).....	5
User Interface (save text box and buttons appear).....	6
User Interface (new button functionality).....	6
User Interface (drop down box, open and populate buttons functionality).....	6
User Interface (Correct labels and list boxes are populated).....	6
Gantt Chart.....	7
Vision Document.....	8
Project Mission.....	8
Customers.....	8
Benefits.....	8
Key factors used to judge quality.....	8
Key features.....	8
Technology.....	9
Crucial product factors.....	9

Objectives

The objective of this functional specification is to provide a detailed document for Software Developers describing our product's capabilities, appearances, and interactions with its users.

This functional specification will serve as a guideline and reference point for the Software Developer while coding the application. This functional specification will also serve as a guideline for Quality Assurance Personnel. It will help them to identify many aspects of the application that need to be tested, it will also be a great resource in identifying a starting point for testing.

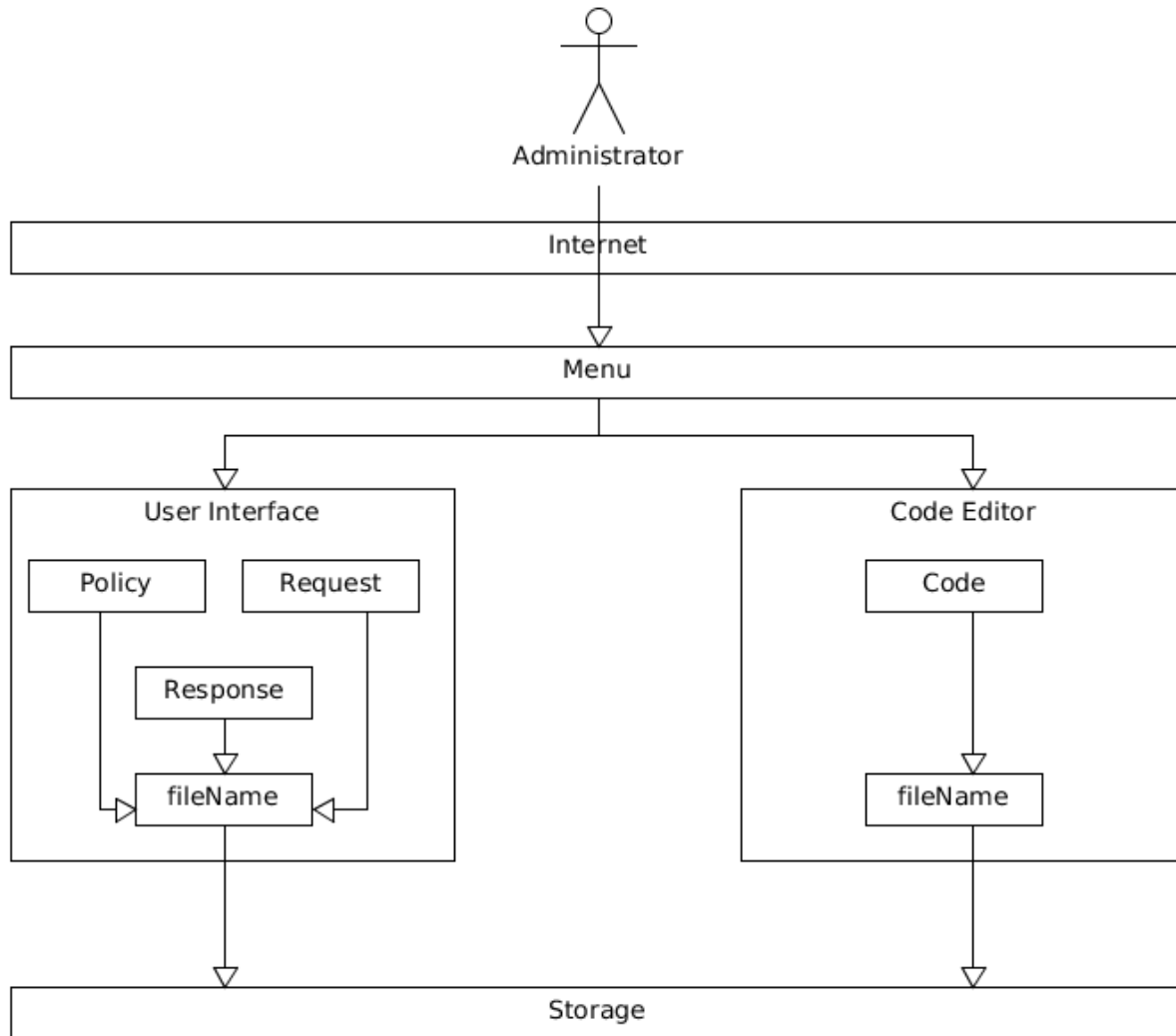
The objective of this project is to develop a web based XACML Graphical User Interface Policy Editor using Java Enterprise Edition and the ZK framework. In order to do this research must be done into Java Enterprise Edition and the ZK framework.

In this document an overview of the applications architecture, functionality, usability, reliability, performance, supportability, test plan and a time-line for the project will be provided.

Also including in the document is the Vision Document which will cover the project mission, customers, benefits, key factors used to judge quality, key features, technology used and crucial product factors.

Architecture

The following diagram shows the Architecture for the web based XACML User Interface Editor. It consists of the user accessing the application through their favourite web browser. The system allows the user to create an XACML policy through the use of a graphical user interface or a source code editor for some of the more experienced users. When the user decides to save the XACML policy it is parsed into XML format using the Document Object Model parser, then stored.



FURPS+

Functionality

- Remote access.
- An Administrator can view, edit and create the XML files via the XACML Code Editor.
- An Administrator can view, edit and create the XML files via the Graphical User Interface.
- The code editor should allow drag and drop, cut and paste.
- The code editor should have content assist.
- It should be possible to switch between the Graphical User Interface and the Code Editor.

Usability

- The user interface of the XACML Editor should be designed for ease-of-use and shall be appropriate for a computer-literate user community with no additional training on the System.
- The code editor of the XACML Editor should be easy to read and be easy to use.

Reliability

- The system must be able to perform all the functions required to create XACML policies, requests and responses. (Read,write and update)
- The application should not fail more than 3 or 4 time per year.

Performance

- The application should be efficient, creating a policy, request or response through the User Interface should take no longer than 15 minutes start to finish.
- The response time for the application should take no longer than 3 seconds.

Supportability

The application should be easily maintainable by the company developers.

It should be supported by a range of different browsers for example:

- Google Chrome
- Firefox
- Safari
- Opera
- Internet Explorer

Test Cases

Code Editor (create/save)

- Ensure that the code editor displays when the code editor option is clicked in the toolbar.
- Ensure that the version and file type radio buttons appear.
- Ensure that the text entered into the code editor is identified as xml code i.e. the code is coloured.
- Ensure that the file name text box and save button appear.
- If the save box is empty, the message “Enter a filename!” is displayed and the save is cancelled.
- Ensure that the code entered can be saved by entering a filename in the file name text box and clicking the save button.
- Ensure that the code has been saved correctly with correct indentation, in XML format.

Code Editor (open/amend)

- The user selects the version and file type radio buttons.
- The open drop down box is present and populated.
- The user chooses the name of the file they wish to open from the open drop down box.
- The user clicks the open button.
- The code editor is populated with the code contained in the chosen file.
- The save text box should now be populated with the name of the opened file.
- Modify the code in the file.
- Click the save button.
- A warning appears to confirm that the user wishes to overwrite the existing file.
- Ensure the file has been updated with the modified content.

User Interface (radio buttons)

- The user selects the User Interface option from the toolbar.
- Confirm that radio buttons for selecting the XACML version 2 and version 3 are displayed.
- Confirm that radio buttons for selecting the file type (policy, request, response) are displayed.
- Ensure the correct pages are displayed by clicking these radio buttons.

User Interface (save text box and buttons appear)

- Ensure that the save text box appears at the bottom of every page.
- Ensure that the save button appears at the bottom of every page.
- Ensure that the new button appears at the bottom of every page.

User Interface (save button functionality)

- If the save text box is empty a warning is displayed to the user and the save is cancelled.
- If the chosen file name already exists a warning with the message "File name already exists are you sure you want to overwrite?"
- Ok. File is overwritten.
- Cancel. Save is cancelled.

User Interface (new button functionality)

- When the new button is clicked display the message "Are you sure you want to clear everything?".
- Ok. Clear everything.
- Cancel. Cancel clear.

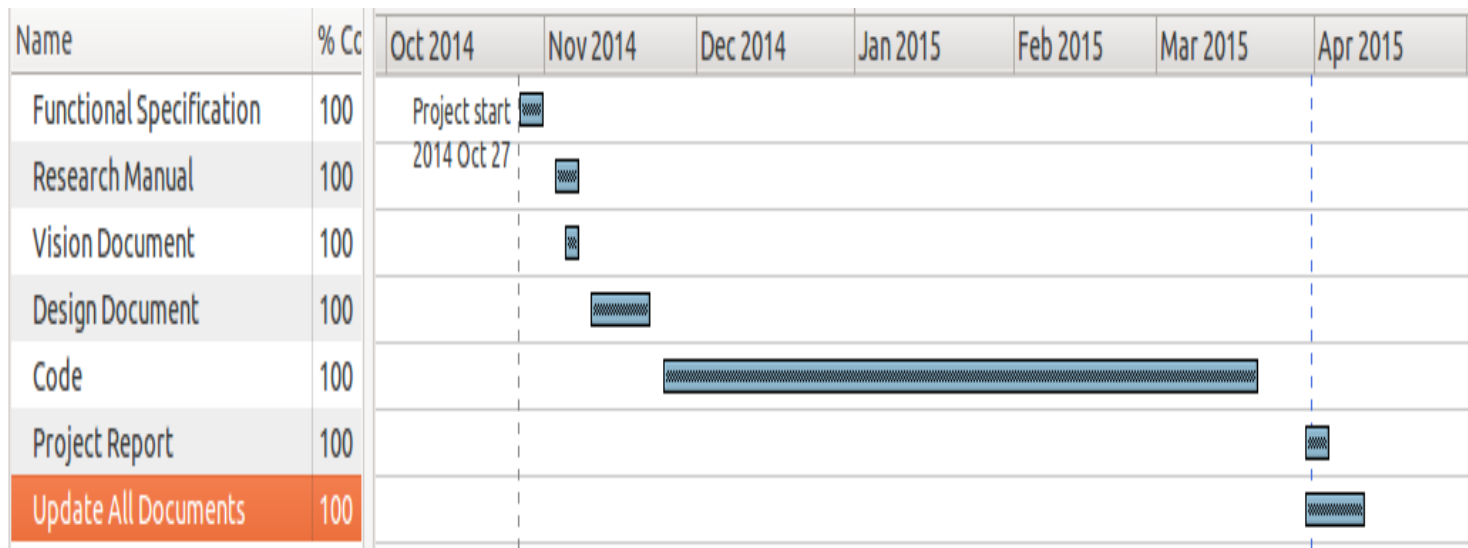
User Interface (drop down box, open and populate buttons functionality)

- Drop down box is present.
- Populate button is clicked and the drop down box is now populated.
- Select a file and click open.
- Corresponding fields are populated with data from the file.

User Interface (Correct labels and list boxes are populated)

- Ensure the text on all the labels are correct.
- Ensure that the list boxes are updated when new attributes are added.

Gantt Chart



Vision Document

Project Mission

To produce an XACML Policies UI Editor that will allow the user to Permit or Deny access and assign rules through the Web Based UI rather than having to code it in XACML. It should also provide access to the XACML code for developers or people with a technical background.

Customers

System administrators: the people who manage the operation of a computer system.

Benefits

- Primary: Users will not need a background in Software Development to assign rules and permissions.
- Secondary: The process will be faster and easier through the UI as a few simple mouse clicks will function the same as writing lines of code (LOC). It will also be easier to read the available rules as it will not be mixed in with code.

Key factors used to judge quality

- It should simplify and speed up the process of authorizing or denying employee permissions.
- A first time user should be able to use the system without assistance.
- It should be easy to create rules.
- Current rules should be easy to understand.

Key features

- Graphically edit an XACML policy
- Grant permissions.
- Deny permissions.
- Assign a rule(s).
- Cut and paste / drag and drop.
- Should support both XACML Version 2 and XACML Version 3.
- Read existing rules.
- Switch between graphical representation and XML representation.

Technology

- Java Enterprise Edition.
- ZK Framework.
- XML and XACML.
- Apache Tomcat Servlet Container.

Crucial product factors

- Web based.
- Must be extensible such that more people and rules may be added at a later time.
- Quality and reliability.
- A Graphical User Interface with code editor.