



Generic Authentication Framework – GAUTH

Minor Project

Disclaimer

This Software Requirements Specification document is a guideline. The document details all the high level requirements. The document also describes the broad scope of the project. While developing the solution if the developer has a valid point to add more details being within the scope specified then it can be accommodated after consultation with IBM designated Mentor.

INTRODUCTION

The purpose of this document is to define scope and requirements of a Generic Authentication Framework (GAUTH) for the developers of a software services company, engaged in developing Java Servlet/JSP based web applications. The proposed system will eliminate the need to develop a user authentication system from scratch everytime. GAUTH framework will provide a loosely coupled and easy to use authentication system based on user-id/password.

This document is the primary input to the development team to architect a solution for this project.

System Users

The entire team of developers of the software services company is expected to benefit from the generic authentication framework, GAUTH.

Assumptions

1. The user id and password pairs of authorized users will be already available in a database table. You may populate the user table manually in the database for testing purpose. In real life situation, passwords are never stored in clear in the database; instead a secured hash of password is stored. Please see section on optional requirements.
2. It is assumed that the standard Java EE Filter Mechanism will be used for the development.

REQUIREMENTS

GAUTH will provide a very simple and loosely coupled mechanism to add authentication to any new web application requiring user-is/password based secured access.

GAUTH will leverage the Java EE standard “filters” to build this framework. The overall flow is outlined below:

1. The end-user attempts to access a “secured” page (URI) in the web application;
2. GAUTH “filter” intercepts the end-user’s request;
3. GAUTH filter now retrieves the end-user’s “user object” from the session;
4. If valid “user object” exists, filter allows end-user to access the secured page; otherwise it redirects the end-user to the “login page”. A GAUTH “login servlet” authenticates the end-user using entered credentials against the user table in the database. If the authentication is successful, GAUTH stores the “user object” in the session and redirects user to the “welcome page”. In the event of authentication failure, GAUTH redirects the end-user to the “login page” again.

Therefore, the GAUTH framework will consists of 2 key artifacts viz. (a) a filter to intercept requests to secure pages of the web application, and (b) a login servlet.

It will also allow developers to create their own “login page” and configure the same in GAUTH framework during integration. Similarly, it will also allow configuration of the “welcome page”.

It will also have a mechanism to specify the web application pages that require secured access.

All configurations will be done either using Java EE standard configuration files or in database along with proper corresponding form(s) to update configuration.

Optional Requirements

Consider developing a user registration module along with secure hash password storage feature.

DEVELOPMENT ENVIRONMENT

GAUTH will be developed as a web application using Java/JSP/Servlets/Filters and DB2 database. Eclipse will be used as the IDE for the same. You may consider using a JavaScript framework like jQuery/Prototype/ Scriptaculous.

Details about Java EE 6 Filters can be found at <http://docs.oracle.com/javaee/6/tutorial/doc/bnagb.html> URL.