6CCS3PRJ Final Year Project
Task Allocation System

Second Progress Report

Author: Hussain Miah

Supervisor: Dr Tomasz Radzik

Student ID: 1114801

Date: 10th November 2014

Introduction

The first progress report outlined the basic requirements of the application and the functionality that it will provided to the company that will be using it. This report will detail the progress that has been made so far in relation to the database design and progress that has been to the functionality of the application and what actions the user is able to do now.

Progress

The first step in this project was to create the database that will be used to store the information relating to the employees and the tasks for the company. I have created a script will load and create a **TaskAllocation** database, after this database has been created the script will then load the schema of the **TaskAllocation** database which will create all the tables and foreign keys etc… that was specified in the database design in the first progress report. After the database and its schema have been created the script will then create a MySQL user which will be used to access the **TaskAllocation** database and this user will be given the relevant permissions to query that database and do any manipulations that are required. This user will not be able to access any other databases that is within the MySQL instance that is running, it will only be able to access the database that I have created. This username and password will then be used within the Java application to access the database as opposed to just using the default **root** username and password.

This script is separate from the Java Application, this script should be used when the user is first "installing" or running the application for the first time. **I will assume that there is already a MySQL database instance installed and running on the machine, that the application will be used on and I will not be providing any installation guides on MySQL.**

The script that I have written to load the database required is written for both a Windows and Unix environment, it is as simple as running the script and entering in the information that is required when prompted by the script.

Finally I have created the CSV import functionality which will be used to populate the database with the organisations information for them to start allocating tasks to their employees. The user is presented with a user interface and they can choose which file they would like to have imported into the database. They have the ability to navigate their file system to find the file and then import the file. Currently the application will only accept CSV files and there is also a certain format that has to be followed within the CSV file in order for it to be imported correctly. Only files with the **.csv** extension will accepted. One thing that I forgot to mention in the first Progress Report was that the application will be written using Java 7.

CSV Format

The application will accept any comma separated values file, but see below for an example of what the contents of the CSV file should look like for it to be imported into the database properly. The below example is for the **EMPLOYEES** table within the database:

ID,FIRST_NAME,LAST_NAME,SEX,COST
1,Ben,Jones,M,1000
2,Sophie,Baxter,F,500

The bold line represents the header of the table that is being imported into, this line is added for usability for the user when they are reading the file, so they can see which field matches up with which column header. This line is ignored when the CSV file is being imported, but it must be included in each CSV that the user wishes to import. At the end the user will be presented with a confirmation message detailing the number of rows that have been inserted into the database.