# Term Project: Task Management

### Prof: Fei Xie and TA: Hoang Le

#### Spring 2015

# 1. Overview

In addition to the final implementation, each student is required to provide documentation at key milestones in the software engineering process. Specifically, each student must prepare a requirements document, a design document and a test plan. The project's motivation, requirements, and deliverables are listed below and may be subject to change. Please see the Files tab for additional documentation.

The project will be written using Java 7 and NetBean 8.0.2.

* http://www.oracle.com/us/technologies/java/jdk-7-netbeans-download-432126.html

# 2. Goals

The primary goal of this project is to provide the students experiences in the following areas:

* Working with a medium-scale software project
* Coping with imperfect requirements
* Developing complete, coherent documentation
* Practicing good object oriented design and implementation
* Practicing developing a graphical user interface
* Practice writing unit tests

# 3. Schedule

The project will be announced in the second week of class and concluded in the last week of class. Important Dates:

* April 9: The term project is announced
* April 21: The requirements document is due
* May 5: The design document is due
* May 28: The test plan is due
* June 4: The project report and final deliverables are due

# 4. Grading

Grading for the project will be based primarily on work done at different phases. Please direct your further questions to Prof. Fei Xie and TA: Hoang Le.

Point Distribution:

* 25 pts: Requirements document
* 25 pts: Design document
* 25 pts: Test plan
* 25 pts: Project report and final deliverables

Notes:

* + All the documents must be submitted through the Redmine in to your repository before the 11:59pm of the due day. Late submission will be ignored except you have permission from the instructor.
  + The final project is also submitted through Redmine repository. You need to deliver the whole NetBean project, including the database. I will be tested using the same system

# 5. Project Description

**Task Management**

* Task Management is a program that helps a user to manage his/her daily tasks.
* This program has a simple graphic user interface to let a user interact with it
* This program can be used for multiple users. It has an account management module. Each user has a private account. Each user has to login with a username and password before using the program. Account creation module is not required. Accounts can be pre-stored.
* The program maintains three task lists: ***To Do***, ***In Progress***, and ***Done***. The program displays all three lists so a user can see and interact with it
  + ***To Do***: this list contains a list of tasks that a user plans to do
    - A user can create and add a new task to this list
      * A task is a short string describing a task of a user.
    - A user can select a task and add it to the *In Progress* list
  + ***In Progress***: this list contains a list of tasks that a user is doing
    - When a task is done, a user can select the task and add it to the *Done* list
  + ***Done***: this list contains a list of tasks that have been done
    - A user can select a task and delete it permanently
* All data of the program are stored on the hard disk. The program will load data into memory when it starts. The status of a user is saved on the hard disk. For each interaction, the program has to save all necessary data.
  + For the database management, you are recommended to use a DBMS system. If you are not familiar with it, you are welcome to manage all data using a regular file system (text files, directories, etc.) Security is not a requirement of this project.

**(1) Requirements**

In order to complete a well-structured requirements document, please refer to the template of requirements document. The purpose of this document is for you to get a better understanding of what is expected. The project specification is vague.

**(2) Analysis and Design**

In order to complete a well-structured design document, please refer to the template of design document. And the important sections of this document are section 4 & 5.

For Section 4 (System Architecture), in addition to the descriptions for subsystems, component or object, you are also required to give a dataflow diagram (please refer to the diagrams in chapter 12 of textbook)

For Section 5 (Detailed System Design), in addition to the highly detailed description of your subsystem, component or object, you are also required to give detailed class diagrams (please refer to the Figure 14.11 in textbook).

**(3) Test Plan**

In order to complete a well-structured test document, please refer to the template of test plan document.

To ensure that your project can meet the requirements, you should give test information in detail for all units (Section 3) and the whole system (Section 4). Please note that you are not going to give just part of them as you did for the design document. The specific requirements for each part of these sections have been included in the test plan document template. For unit testing, test cases for a certain testing unit can be considered as different scenarios. You can give one passing case and one exceptional case for each unit.

**(4) Final Project Deliverables**

A short 5 minutes demonstration is required with the TA before the end of finals week. You should demonstrate that the application meets the requirements as specified in the project description. In addition, you should run the unit tests for the TA. A one page project summary is required. Talk about the structure of the code / tests and any challenges you experienced while working on the project. Please talk about any special features you developed in your application above and beyond the requirements. These extra features may be awarded extra credit at the discretion of the TA, however all functional requirements must be met first! Please check in your code and report into the repository.