

CSC301 Introduction to Software Engineering

Assignment 4

Details

Topics: agile methodology, release planning, coding standard & documentation

Given: July 19, 2008

Due: August 3rd, 11:00 p.m.

Assignment Description

To gain maximum benefits from this assignment, you need to do it in the steps detailed below.

(a) Feature Planning

For this assignment, the set of major features that you are required to support are: membership, content, production, and subscription (refer to appendix A in assignment 2 handout for the transactions list). Each feature should have a unique id, a description, pre-requisites list, and a reference to the relevant part(s) of your analysis or design documents. You need to assign priorities to these features; A, B, and C. A features are a must have, B features are a good to have and C features are nice to have but not essential for release 2.0. If you don't have time to finish implementation, start by dropping features with priority C.

From this list of features, you need to generate a set of development tasks. Each task should have a name, description and a unique id. The development tasks need to be detailed enough to eliminate any ambiguity regarding how a feature is implemented and it should also be high level enough that it does not look like an algorithm implementation.

(b) Effort Estimation

This part starts with an individual effort. You are going to try to estimate/guesstimate the number of hours each task will take if you were to develop it. You need to consider best case and worst case scenarios. For example, adding a channel to an existing list of channels takes 15 hours +/- 5 hours. Try to separate the time it the number of hours you spent actually coding vs. thinking about the solution vs. testing. We don't expect you to get it right and you will not be marked/judged in any way on the accuracy of your estimation. This is just an exercise for you to think about estimation issues. During coding, try to keep in mind your initial estimates. You are going to submit your individual effort estimation separately using the submit command.

Next, you need to meet with the team and agree on the number of hours it takes each task without telling them exactly your initial estimate (you will all know each other estimates after the assignment is submitted and are encouraged to discuss these estimates then).

(c) Resource Estimation

This part starts as an individual effort. In this step, you need to estimate the number of hours available for you to work on this assignment - given all your other obligations. You have to be realistic and also responsible. Assuming that you will be able to work 10 hours per day on this assignment is probably unrealistic. Similarly, reporting that you can only put 0.5 hours per day is also not acceptable.

Next, you need to meet with your team and discuss your estimates. You all have to agree on the total number of hours each and all of you can realistically put (regardless of the features you are going to support).

(d) Balancing effort vs. resource

You need to go back to the output of step (a) and re-evaluate the feature set for release 2.0: *can you really implement those set of features given what you know about the available resources?* If not, you have to reconsider the feature set.

(e) Work Allocation

You will need to divide your team to 2 mini-teams; and divide the tasks among those mini-teams. You are highly encouraged to do pair programming. If your OO design is highly cohesive loosely coupled, you will be able to divide the tasks so that each mini-team works on one or more subsystems without being dependent on the other mini-team to finish their work.

(f) Implementation

In this assignment, you are going to use Java annotation to document the code author(s), version, last modification date, and which task and feature the code implements (use feature and task unique numbers). In this assignment, you are also required to use the Hungarian notation in coding.

Deliverables

Deliverables:

- Due July 25th:
 1. Release plan report. This is a group effort submitted ***using cvs***. The report include: feature planning, effort estimation, resource estimation, balancing efforts vs. resources, and work allocation. You should include the initial feature set and the modified feature set after you do resource estimation.
 2. Individual effort estimation report – every group member separately submits his/her own estimates of each of the initial features reported in the release plan report. This is an individual effort submitted ***using submit command***.
- Due August 3rd:
 3. Updated Release plan report. This is a group effort submitted ***using cvs***. This version of the report should update the feature list: which ones were implemented and which ones are partially done and which

ones are not done at all. In addition, every group member reports actual hours spent on implementing assigned task to him/her.

4. Source code of version 2.0. Submitted *using cvs*.

- On the day of the course exam:
 5. Performance reports from each team member:
 - a. team member evaluation form filled for other team members.
 - b. team member evaluation summary form.

Deliverable formats:

Since we are allowing you to use any Java Editor/IDE. You are going to submit a make file with the following targets:

all (compiles all the .java files)
clean (deletes all the *.class files)
run (runs the application)

For the reports, you are going to submit a pdf file (not the file saved by the text editor you use). To do that, you will have to generate a pdf! You will need to install the following free PDF-generation software: <http://www.cutepdf.com/products/cutepdf/Writer.asp> This will install a pdf printer driver on your computer. Use the print menu to print the document using the installed printer driver.

Development Environments

Here is a non-exhaustive list of some Java IDEs tools you could use to do this assignment.

- Eclipse (<http://www.eclipse.org/>)
- NetBeans (<http://www.netbeans.org/>)
- IntelliJ IDEA (<http://www.jetbrains.com/idea/>)
- JCreator (<http://www.jcreator.com/>)

Assignment Marking

The following criteria will be used in marking your submission:

- Total number of features/transactions you have implemented
- Feedback from other team members about your performance
- Quality of Design
- Quality of Code
- Number of faults.