

ICS 121 / Informatics 111: Software Tools and Methods Fall Quarter 2005

Assignment 3

Due Tuesday, November 29, 2005 at 11:59 pm

Part 1. Effort Estimation (30 points)

This problem is the similar to Part 1 from the previous two assignments. If you added categories previously, you should be consistent and continue to use the same categories.

1.1 A priori estimate (3 points) When you start working on this assignment, write down an estimate of how long it will take you to do parts 1 and 2 (each part). You are not required to use the table from Assignment 2 to generate a more accurate estimate. However, you may use that technique if you feel it will provide a more accurate estimate.

1.2 Logging (15 points) As you are working on the assignment, record what you are doing and how long you spent, in the same manner as Assignment 1. For this question, quantity of data, i.e. number of entries, is important. As a rule of thumb, you should add a log entry every time you switch tasks or at least one entry per hour. However, if you do two or three things in half an hour, you must have a log entry for each of them. You do not need to include time for logging, but should include the time spent answering the other parts of this question.

Keep a history of your logs. Either

- Place your log under CVS and submit a screen shot of the "CVS Resource History" view in the CVS Perspective of Eclipse, just as in Assignment 2; or
- Put a two digit numeric suffix at the end of your log file name (e.g. "Log-01.doc"), and every time you save the log, use Save As and increment the suffix. Submit a screen shot of Windows Explorer showing all the versions, including the Date Modified field.

For either screen print, arrange the windows so that what you include in your report is completely legible and large enough to read easily.

1.3 Re-estimation. (3 points) Take the total *a priori* estimate from 1.1 and divide the number in half. When you have worked this many hours on the assignment, update your estimates. Your new estimates should take into the account the amount of time that tasks have taken so far. Also, make a note in your log indicating when you do this re-estimate.

1.4 Analysis and Error Calculation (4 points) Tally up the time spent in each of the tasks and on the entire assignment. Calculate the error for each task for each of the estimates. Produce a bar graph of the time spent. Here's what a table of summary statistics might look like.

Task	Actual Time (minutes)	Percentage	Initial Estimate	Error	Revised Estimate	Error
1						
2.0						
2.1						
2.2						
2.3						
Total		100.0				

As always, the error is the amount off, divided by the initial estimate, and expressed as a percentage.

1.4 Discussion (5 points) You have used three different estimation techniques, simple estimation with intervals, estimation by category, and re-estimation. Compare and contrast the three techniques. What are the strengths and weaknesses of each? Which method is most effective? Can the techniques be combined? What are the advantages and disadvantages of doing so?

Deliverables

- Initial estimates
- Revised estimates
- A log of your time spent on the assignment
- Screen print showing history of log
- Summary statistics and graph
- Discussion of estimation method and resulting error

Part 2 Specification Using UML (70 points)

For Assignment 2, you were given a set of classes that form the middle component in a seminar scheduler system. For this assignment, you'll be developing the specification for the front and back ends of the scheduler using UML. The front end is an interface for users to logon, enter their information, and set up seminars. The back end stores information entered by users and sends out the notifications.

You are required to use a UML drawing tool for this assignment, either Rational Rose or some other tool. You may not use a general-purpose drawing program such as MS Paint and Visio. You will cut and paste the diagrams into your report. However, you may provide a Rational Rose .mdl file as supporting documentation.

2.0. Become familiar with Assignment and Appendix (0 points)

The Appendix contains a preliminary list of requirements, expressed as user stories. Questions about the requirements, i.e. requests for additional detail and clarifications will be entertained only during the first week after the assignment is distributed (Tuesday, Nov. 15, 2005 and Thursday, Nov. 17, 2005). After this time, the requirements will be considered "frozen," so you can make progress on the rest of the assignment.

2.1. Use Cases and Use Case Diagrams (30 points)

Starting with the system description and user stories provided in the Appendix, create the following use cases and use case diagrams. You probably will have several diagrams. A use case diagram can include use cases not on the list below (but don't actually write out more than the specified five use cases). Your use cases must follow either the template at <http://www.ics.uci.edu/~frost/ics121/Assignments/UseCaseTemplate.doc> or the ReadySET Software Engineering Templates, <http://readysset.tigris.org/nonav/templates/use-case-format.html>. Note that it may be appropriate to "include" a use case as part of another one, or as an alternative success scenario.

- A. Logon
- B. Look up other users
- C. Schedule a seminar
- D. View list of scheduled seminars
- E. Add to a personal address book

2.2 Class Diagram (25 points)

Create a class diagram of the seminar scheduler. It must include the classes from Assignment 2, Attendee, Seminar, Scheduler, and Invitation. You will need to add classes, operations, and attributes to account for the use cases that you developed. For instance, users will need personal

address books, and there should be classes for the user interface and for storing records of the seminars.

The diagram must be detailed enough to specify an implementation. For attributes, you must include visibility, name, and type. Include a default value if applicable. For operations, you must include visibility, name, parameter list, and return type. Show generalizations, realizations, associations, cardinality (multiplicities), and role names where they apply.

2.3 Sequence Charts (15 points)

Draw sequence charts for the following use cases.

- C. Schedule a seminar
- D. View list of scheduled seminars

Deliverables

- Five uses cases with use case diagrams
- One class diagram
- Two sequence charts

Handing In Your Assignment

Your assignment must be submitted electronically to checkmate.ics.uci.edu. Submit three files.

1. Part1.doc or Part1.pdf, containing the deliverables from Part 1.
2. Part2.doc or Part2.pdf, containing the deliverables from Part 2.
3. Diagrams.zip, containing the Rational Rose (or equivalent) file or files with the diagrams from Part 2

Appendix: User Stories

Overview

The purpose of the system is to schedule and initiate seminars. (Think of it as a highly simplified version of the event planning feature of www.evite.com.) Users will be able to schedule seminars with a number of attendees who are also users of the system. When a seminar is scheduled, a notification is generated and sent out to the attendees. The system will maintain a record of seminars and attendees.

User Stories

The user logs on to the scheduling system using a unique ID and a password¹.

If the user has never used the system before, he or she must register. To do so, an ID is created and a password associated with the ID. The user enters personal information, including location, time zone, and email address.

After logging in, the user is presented with a list of upcoming seminars. She or he can also review a list of previous seminars and the people who were invited to the seminars.

To schedule a seminar, the user sets a time and date in his or her local time zone. He or she can then add participants to the seminar. A user must be a participant in any seminar that he or she creates. The seminar must have a title and may have notes to provide additional details. A user can add people by specifying their ID, selecting the participant from a personal address book, or from the system directory by browsing or searching. When the user is satisfied with the seminar, he or she can ask the system to save the information and send out notifications. Up until that time, the user can make any changes to the invitation list and seminar time.

The notifications shall be sent out by email. They will contain information about the seminar including the time and date that is local to the recipient of the notification. The date format shall be mm/dd/yyyy for recipients in the USA and ISO 8601 (yyyy/mm/dd) for every other country.

A user can look up other registered users through a system-wide directory. This feature can be used to obtain more information about other users. Lookup can be done through an ID, browsing, or searching.

Each user will have a personal address book. People in the address book must also be registered users. Address book entries can be added by finding a person through the system directory.

¹ It is up to the system designer to select rules for a secure IDs and passwords, number of attempts before a timeout, etc. You do not need to deal with the case where the password is lost or forgotten.

Inviting a person to a seminar will cause that person to be added to a personal address book, if she or he is not already listed.

Intentionally Omitted Functionality

In order to make the system manageable within the scope of the assignment, a number of features or user stories have intentionally been omitted from these user stories. These features include:

- ❖ Retrieving a forgotten ID or password
- ❖ Giving a seminar a duration or end time
- ❖ Sending notifications using a method other than email
- ❖ RSVPing for a seminar
- ❖ Canceling a seminar
- ❖ Recording attendance at seminars (as opposed to just invitations)
- ❖ Initiating seminars by telephone, instant message, VOIP, or some other method
- ❖ Customized notification formats on a per user basis