

Reflective Software Engineering

Module 04:

Project 1

(1)



Module objectives

To begin programming in a reflective style.

To practice Leap-based time data collection and analysis.

To begin developing a really cool system :-)

(2)



Leap definitions

The 613-data.leap file contains the following:

- A project definition (psae-1)
- A phaseset definition (613)
- A document type definition (JavaSize)

Let's discuss each of these in turn.

(3)



The project definition

“Projects” are the leap mechanism for collecting together all of the relevant data on a small increment of work focussing primarily on a single work product.

Each project should be uniquely named, and this is up to you to guarantee.

We will return to project definition again and again.

(4)



The phaseset definition

The 613 phaseset contains the following phases:

- Design
- Planning
- Implementation
- Testing
- Postmortem

Let's discuss each in turn.

(5)



The design phase

Also known as "preliminary design"

All of the work involved in coming up with the set of objects to be created (in the case of new code) or modified (in the case of old code) or some combination.

For Java, "objects" means classes and methods.

You should write down the classes and methods you think you'll need for this project before beginning implementation, and record how long this design process takes using Leap.

(6)



The planning phase

In the planning phase, you use your design data to come up with estimates for various characteristics of the software project, including the time, size, and other resources required.

For this project, you don't need to do any planning! :-)

(7)



The implementation phase

In this phase, you implement the system.

This includes all coding, compiling, and unit testing activities.

(8)



The testing phase

In this phase, you perform the system-level or "acceptance" testing.

For project P1, that includes running the system on all of the test files you can come up with.

This phase also includes any and all rework activities you have to do when a test case fails.

(9)



The postmortem phase

This phase includes all the activities involved in postmortem and packaging of the system.

This may include additional commenting and so forth discovered when you look at the javadoc output.

(10)



General phaseset comments

Phases are sequential: you cannot “go back” to implementation once you are in postmortem.

It is important, in the long run, to be consistent. For example, your code commenting activities can occur either during implementation or postmortem, but try to do it the same way each time. (Unless another way is better :-)

(11)



The doctype

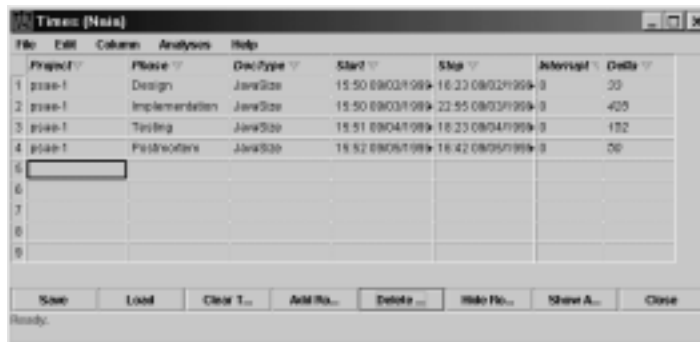
Use the “JavaSize” doctype for your time entries, since the primary artifact you are creating is java source code.

(12)



Project Condensation

To obtain aggregate statistics and graphs of your project, you should "condense" it once you've completed recording all of your time data.



Project	Phase	DocType	Start	Stop	Interval	Delta
psae-1	Design	JavaSize	15:50 08/03/1999	16:23 08/03/1999	0	33
psae-1	Implementaten	JavaSize	15:50 08/03/1999	22:55 08/03/1999	0	405
psae-1	Testing	JavaSize	15:51 08/04/1999	16:23 08/04/1999	0	152
psae-1	Postwork	JavaSize	15:52 08/05/1999	16:42 08/05/1999	0	50
6						
6						
7						
8						
9						

(13)



How to condense

1. Bring up Ilio (project viewer), then right click in the column with the project you want to condense (in this case, psae-1).

This brings up Hee on this project.

2. Fill in the Start Date and End Date fields. (Use the down arrow to get the calendar widget).

3. Press "Update" to record the start/end dates.

4. Press "Condense Raw Data".



Ilio project viewer
Secondary data for project: psae-1 08/02/1999

General GOR New Time Details Condensed

Name: psae-1

Description: First increment basic PSAC file processing

Start Date: 3:55:32 PM 01-Sep-99

End Date: 3:55:32 PM 08-Sep-99

File No:

Processor: 873 Change Processor

Condense Raw Data Cancel Update

(14)



After condensation

After condensation, the "Time" tab in Hee will show the aggregate time spent in each phase. Note "planned" time is zero, but "actual" time is 660.

Phase	Planned		Actual		Error	
	Min.	%	Min.	%	Min.	%
Design	8.0		33.8	5.8	33.8	0.8
Planning	8.0		8.0	0.8	0.0	0.0
Implementation	8.0		426.8	64.4	426.8	0.8
Testing	8.0		152.8	23.8	152.8	0.8
Postmortem	8.0		58.8	7.8	90.8	0.8
(Unallocated)	8.0		8.0		0.0	
Total	8.0		660.8		660.8	
Error Based +/-	8.0		660.8		660.8	

(15)

Final hints

Try to be careful and accurately record the time you take on this project.

Start early, and allow yourself to work on this project over multiple days.

Have fun!

(16)