

Software Hut 2007-8 Managers Record Sheet

Date:

Team:

Members absent:

a. What has the team changed/done this week - is this evident on the wiki pages?

In particular are they recording comments that show the problems/errors/bugs they found and had correct and a brief description of what each task was meant to achieve?

b. What new/changed requirements did the client specify this week?

c. What problems have the encountered this week as team? (e.g. illness, lazy member etc)

Things to check –

d. Results of testing on wiki (per team member):

- | | |
|----|----|
| 1. | 3. |
| 2. | 4. |
| 5. | 6. |

e. Production of at least one set of minutes: NOT PRESENT

f. Wiki Story cards and process documents: (per team member)

- | | |
|----|----|
| 1. | 3. |
| 2. | 4. |
| 5. | 6. |

g. Stories to be implemented in the next week: (per team member)

(IF A STORY TAKES MORE THAN 1 WEEK TO IMPLEMENT IT IS TOO LONG)

- | | |
|----|----|
| 1. | 3. |
| 2. | 4. |
| 5. | 6. |

Please record any problems with the management tool or test server here:

Comments on this form and how to fill it in.

a. This question is primarily to support research, record anything that the team did not plan to do in the previous week (in terms of story cards). This should include any additional tasks that they assigned and completed during the week.

For the benefit of the markers make sure the wiki is kept up to date, either check with the team in the meeting (if you are in the lab) or before the meeting (ask Chris for the admin login)

b, c. The students should be able to report this directly from their minutes – encourage them to do so, and if they report anything which is not in their minutes remind them that this should be in the minutes!

Again check this in the management tool prior to the meeting.

The changes to the requirements should be reflected in the story cards, along with any bugs which may also be documented as changed tests/code.

Problems should at least include: any technical problems that they had difficulty solving; communication issues within the team or with client; and anything that has made it impossible for them to work on the project this week (e.g. ill health, other course work, etc.)

d. Each team member should update their on the wiki after testing. All teams should do so at least twice – once before they write any code, when the tests should fail. Ask the students to comment on the results of the tests – and any problems they had.

e. Ensure the team produced minutes in the last week – ask to see the print out, or view in the management tool before the meeting.

f. Time sheets can be checked from the management tool prior to the meeting. In the meeting ask the students to justify the time spent (with evidence: documents created etc.) – try to weed out anybody who is recording random hours.

g. The teams should have assigned a set of tasks in their last team meeting, check these and progress on the implementation of the stories. From week 2 onwards we expect at least one story to be implemented every week per person, fully tested and integrated into the working system. If the stories are too big get the team to split them. Ask for the system to be demoed to you every so often to confirm this.