

COMSW4156_001_2015_3: ADVANCED SOFTWARE ENGINEERING (Fall 2015)

View Site As: - Select Role - ▼

Home
Announcements
Piazza
Calendar
Textbooks
Syllabus
Assignments
Gradebook
Mailtool
Roster
Files & Resources
Site Settings
Evaluation
Help

Assignments

Viewing assignment...

▼ Settings for "Project Use Cases and Tasks (2015 Team Assignment 3)"

Created by	Gail E. Kaiser
Date created	Oct 11, 2015 11:21 pm
Open	Oct 13, 2015 12:00 am
Due	Oct 20, 2015 11:55 pm
Accept Until	Oct 21, 2015 11:55 pm
Modified by instructor	Oct 12, 2015 1:25 am
Student Submissions	Single Uploaded File only
Number of resubmissions allowed	Unlimited
Accept Resubmission Until	Oct 27, 2015 11:55 pm
Grade	Points (max 20.0)
Alert:	Yes
Honor pledge:	No

Assignment Instructions

Get together with the other members of your entire team.

"ASE, LLC" has accepted your project proposal, so now you need to elaborate your user stories into use cases, and then define technical tasks to implement each of your use cases (in the real world, some developers go directly from user stories to technical tasks, some expand to use cases as we will do, and some start with use cases in the first place without the simpler user stories). Your team may also need to do some technical tasks that do not correspond to any particular user story/use case, which we will call "Infrastructure". Recall this is for Version 0.1, which should be sufficient to demonstrate the idea for your product, but does not need to be "complete".

Submit on courseworks (per team) a document with the following contents:

page 1: Team name, and full names and unis of all members of the team. This should be on a separate page, do not start the material for page 2 on the same page.

page 2: Write a header "Synopsis" at the top of page 2. Then briefly describe your idea for the application, including who are the target users and what value your application will provide to those users. You can copy/paste this from your (revised) Project Proposal if nothing has changed, but it might be wise to reduce the scope of what you expect to be able to demonstrate for Version 0.1.

page 3: Write a header "User Stories" at the top of page 3. Then write only the *titles* of your user stories for Version 0.1 together with the page number where you begin to elaborate each user story. This is intended to be a table of contents.

page 4: Write a header "<title of first user story>" (fill in the actual title) at the top of page 4. Then write your user story. Remember this should only be (approximately) three sentences in customer language, so should definitely fit on one page. You can copy/paste from your Project Proposal if this user story hasn't changed.

pages 5 to N: Write a header "Use Case for <title of first user story>" (fill in the actual title again) at the top of page 5. Then elaborate your user story into a use case, with the detailed steps of at least a basic flow and possibly one or more alternative flows. If there are any alternative flows, make sure it is clear exactly *where* and *why* each alternative diverges from the basic flow and, if applicable, later converges (each alternative flow necessarily must diverge at some step but might end separately without ever

converging).

pages N+1 to M: Write a header "Technical Tasks for <title of first user story>" (duh, fill it in again) at the top of page N+1. Then write the technical tasks you expect your team will need to do to implement this user story. These might include design, coding, testing, possibly other activities; we expect this to be vague, that's ok.

Repeat the above for each user story, that is, page M+1 should have the header "<title of second user story>", page M+2 should have the header "Use Case for <title of second user story>", and page M+Q should have the header "Technical Tasks for <title of second user story>", and so on.

pages Y to Z: Finally, write a header "Infrastructure Tasks", and write any technical tasks you think your team will need to do that do not correspond to any particular user story. If none, then just write here "None".

Your document can be either MS Word, or equivalent (.doc), or Adobe pdf.

In addition, for each user story, use case and technical task in your document, make sure there is an issue in JIRA that is verbatim the same. This might involve editing existing user stories and deprecating other user stories (please do not actually delete them). JIRA has a concept of links for linking issues to each other, so you should link your use cases and technical tasks to their corresponding user stories (if any). Ask Georgios for help if you have trouble working with JIRA; if you're doing something we didn't envision, he may need to modify the relevant issue definitions. Hint: You can write all this stuff in JIRA first, and then export to a document.

CourseWorks runs on Sakai[2.9-COLUMBIA (2016_3-1830) - kabocha-cj], set to EST.

[CourseWorks Help/Support](#)