

Assignment 3 – Project Requirements

Due

24 hours prior to your lab in the week of September 29, 2014.

Late assignments will not be accepted and the whole team will receive 0 for the assignment. No exceptions will be made.

NOTE: This is the beginning of the group project work. In this course you will be working in teams based on your registered lab section.

To complete this assignment, you are expected to spend *at least* 4-6 hours (including the 2 hours in lab). Completing the user stories is team work and it will be easier as a team since you can all help to create a complete and correct set of user stories. Make sure to find time to work as a team and do so early on, since it might take you longer than you expect at first.

Objectives

The purpose of this assignment is to introduce you to the requirements phase through a hands-on activity. You will learn how to generate user stories including acceptance criteria.

Project “Epics”

An epic is a very large user story (think “whoah — this is an epic user story!”) — a user story that won’t fit into a single sprint, and instead needs to be broken up into smaller stories. In this project you will be responsible for completing the user stories related to several fundamental epics:

1. Import and clean data - access data on a remote server, parse it, and translate it into a format useful for manipulation.
2. Storage - the data should persist once uploaded. You must load your data programmatically. You should have a button that an admin can use to load the data.
3. Data display - display the imported data in a tabular and a geographic format.
4. Social network integration - integrate with at least one social network (e.g. Facebook, Twitter, etc.).
5. Access - Users need to login with a password and have some user-specific account data (profile, preferences, media, etc...)

Pre-lab work

You must look through the posted data sets (<http://data.vancouver.ca> and <http://www.data.gov.bc.ca/>) prior to your lab and **bring a written suggestion for a project topic** when you attend your lab in the week of Sept 22. Note that the Google App Engine will only allow 50000 write operations per day, so you must be sure to select a dataset of a reasonable size. If you’re unsure about the size of your dataset, talk to your TA. This will hopefully help your team quickly settle on a topic so that you can move on to the requirements elicitation during lab. The pre-lab will be graded as part of your participation/attendance grade for lab.

Procedure – Product Backlog

At the beginning of lab, your team has to come up with a name for your team (no longer than 25 characters!), be creative. As soon as you can, give your TA a list of your names, student numbers, CS ids, GitHub ids and your team name so that he/she can create your team and team's repository on GitHub. You are responsible for getting this information to your TA. Without a GitHub repository, you cannot complete the assignment and I cannot configure handin for your team.

Next, select a data set to work with. Once you've selected your data set you will perform requirements elicitation by brainstorming, researching similar products, and asking your TA questions.

After the requirements elicitation session, each team has to analyze the data they generated and specify the requirements for the product the customer wants in the form of user stories. Each user story has to include at least a summary and acceptance criteria (as presented in the lecture).

You must write each story in a GitHub issue. Use GitHub labels to label each story with the epic(s) that it covers. You may also want to create additional logical groupings and use GitHub labels to display those groupings.

Create a Milestone in GitHub called Product Backlog. Assign all of the Issues that represent your user stories to this Milestone.

Procedure – Peer Evaluation

Each person needs to submit a peer evaluation. This evaluation is individual and not shared with the group. Please submit this evaluation as soon as you finish your group work for this assignment. It is due the day of your lab in the week of Sept 29, 2014 at midnight. It must be in .rtf or .pdf format.

You can find the peer evaluation here: <http://www.ugrad.cs.ubc.ca/~cs310/peerEval.pdf>

Deliverables

24 hours prior to your next lab, your team must:

1. Create your user stories as Issues in GitHub (including the acceptance criteria) and assign them to a Milestone called Product Backlog. You must use GitHub's labels to label each user story with the epic(s) that it pertains to.
2. Write a project title and 1-2 paragraph summary describing your application on your team's GitHub wiki on a page called Summary.
3. One person on your team must submit a copy of your user stories via handin in .rtf or .pdf format. This copy of your user stories must be sorted by epic. I expect to see 5 headings (one per epic), with the user stories beneath the headings. Each user story may cover more than one epic, but please just include each user story once in this document. You can decide where it most appropriately fits. We will be able to see that it pertains to more than one epic by looking at the labels on GitHub.

By midnight on the day of your lab, each of you must:

1. Submit your peer evaluation via handin.

Grading Scheme:

1. (10%) Peer Evaluation (completing the evaluation and giving helpful comments)

NOTE: If you do not hand in your peer evaluation, you will receive 0/10 on the peer evaluation grade, so your maximum grade on the assignment will be 90/100. Your teammates will not be penalized.

2. (90%) the User Stories and Product Backlog

NOTE: this part will be adapted according to the peer evaluations, i.e. the peer evaluation will be used as a percentage-multiplier in calculating your grade

How to use handin for your list of user stories:

1. Put the file in your `~/cs310/a3-X` directory (where you must replace X with your **lab section name**, all lower case)
2. Run the command `handin cs310 a3-X` (again, replace X with **your lab section**, all lower case)

Other notes about handin

- You can confirm that your assignment was handed in correctly by using the `-c` flag (ie, `handin -c cs310 a3-X`)
- If you need to overwrite a previous version that you handed in, using the `-o` flag (ie, `handin -o cs310 a3-X`) Assignments will be accepted by handin up to the due date.
- If you are using the web handin, you must zip your assignment first because it will only accept zip files.

How to use handin for your peer evaluation:

3. Put the file in your `~/cs310/a3-Y` directory (where you must replace Y with **your team name**, all lower case)
4. Run the command `handin cs310 a3-Y` (again, replace Y with **your team name**, all lower case)

Other notes about handin

- You can confirm that your assignment was handed in correctly by using the `-c` flag (ie, `handin -c cs310 a3-Y`)
- If you need to overwrite a previous version that you handed in, using the `-o` flag (ie, `handin -o cs310 a3-Y`) Assignments will be accepted by handin up to the due date.
- If you are using the web handin, you must zip your assignment first because it will only accept zip files.