

CS411 Team Assignment 2 - Requirements Analysis and User Stories

Deliverable: Link to your GitHub repository submitted on Gradescope; place user stories in a docs/User Stories directory, one file per story

For a web app of this size, there will typically be three to five primary user stories, and a handful of secondary ones. For example, for a wine app, a primary user story might be “as a logged-in user, I want to search for a specific wine in order to add it to my online cellar”.

For this assignment, identify at least five user stories for your app and write a description of what each user story covers.

The description is a narrative that encompasses what the user will do, what happens when everything goes right, and what to do if something goes wrong. It should include a reference to any external user story that might be involved. For example, if a story requires that a user is logged in, you should refer the reader to the ‘user log in’ story rather than explaining in detail how the user is authenticated.

Be as complete as you can. The goal is to be able to use a story as a guide during development. It should contain enough information that a developer can understand what is needed and how the story relates to other stories. You’ll also use your collected stories when deciding what sort of architecture and infrastructure you will need. You can include diagrams if you think that they would be useful. User stories typically are about a page long. I often approach the user story as if I’m explaining to someone how to use my app, but they don’t have the actual app in front of them.

When grading this assignment, I’ll be looking for enough information about the stories so that I would be able to start designing an approach to satisfy it.

When working on these, avoid the temptation to think about implementation. The user stories should just describe actions and results, not how they will be accomplished. Try to use non-technical language; the document is meant to be understood by the business user in terms that they would understand...use the language of the user, not the developer.

To get started, think about a brainstorming session similar to what we did in class to help you identify your primary stories.

You might also want to look at tools that the team can use to organize the work and roles, and to track and assign tasks. A Google search for ‘online collaboration’ will turn up some useful ones. Many teams in the past have used Slack, but you are welcome to explore a bit and pick a tool or method that the team is comfortable with. Zoom works, too.