Oregon State University
Ecampus

Brian Dy

CS361: Assignment 3: Project Plan and Sprint 1 Plan (for Milestone #1)

Overview

Now that you've been introduced to the microservices concept, start planning your individual project. It's OK to change your plan later!

First, define the entire set of user stories for your individual project. These user stories will need to go in a **Product Backlog** column/section/category of your task management system. **You won't have to finish implementing the entire Product Backlog this term**.

Next, move some of those user stories into a **Sprint Backlog**—these will be the user stories you WILL complete during this Sprint (for Milestone #1) and comprise your **Sprint Plan**.

Note these minimum requirements for Milestone #1:

- At least three user stories completed
- All features that are part of the milestone must be working. The milestone should not have partially completed features.
- Has a way for users to interact (e.g., provide input, push buttons, etc.)
- Reflects each of the Inclusivity Heuristics
- Reflects three quality attributes of your choice (i.e., satisfies the non-functional requirements you write for each quality attribute)
    - *Hint: If you choose "usability" or "inclusivity" as a quality attribute, your corresponding non-functional requirement can involve the Inclusivity Heuristics.*
    - *Hint 2: "Maintainability" is another good quality attribute to select because, later in the course, you will be asked to improve the maintainability of your code by eliminating the "code smells" listed in Chapter 8 of the textbook.*

Instructions

Complete each item below by replacing the highlighted text (**Usability note**: double-click the text to select it).

1) **Task Management System**
Choose a task management system. It would make sense to choose a system you spiked for an earlier assignment but you're not required to (we won't check).

Which task management system did you choose?

ClickUp

2) Product Goal and Backlog

You'll be using *some* Scrum methods in this course. Unfortunately, the Scrum Master and Product Owner roles don't work well in this course setting. You will, however, experience Scrum Events and Artifacts.

a) What is your **Product Goal** for your individual project?

The goal for this app will be to create a simple python app that runs on a command line interface that can provide simple recipes for the user. It will have simple functionality that a user can click on to select what type of recipe's they can choose from(i.e., Asian, Italian, Soups, Mexican).

The Scrum Guide (https://scrumguides.org/scrum-guide.html) doesn't give a detailed description of the Product Goal: "**describes a future state**", "**long-term objective**". Example Product Goal: "Develop a desktop app that listens to what people are saying and automatically shows content that might be relevant to their conversation."

b) In a **Product Backlog** column, section, or category of your task management system create **user stories** for your entire **individual project**. Use INVEST to guide you.

Assignment requirements for Product Backlog user stories:

- Each has a **name** that briefly describes the functionality (e.g., "Login")
- Each uses the **"As a… I want to… so that…" format** (explained in textbook)
- Each is about **functionality** and not about the quality of the functionality or a constraint (user stories are functional requirements, not non-functional requirements)
- Total of at least **10** user stories (you will not have to implement all of these)
- As a set, must have **no obvious violations of INVEST**

- **User story 1 name:** *Navigation*
- **User story 1 "As a…" format:** *As a user who prefers using the command line, I want to be able to navigate through recipe categories quickly and efficiently, so that I can find and select recipes of interest with ease*

- **User story 2 name:** *Profiles/ Connecting with Others*
- **User story 2 "As a…" format:** *As a site member, I can connect with other users so that I can share and view different types of recipes*

- **User story 3 name:** *Courses and Events*
- **User story 3 "As a…" format:** *As a site user, I can see the different courses and events that are available so that I can attend events in my area that interest me*

- **User story 4 name:** *Search ratings*

- **User story 4 “As a…” format:** *As a site visitor, I can view a list of popular rated recipes so that I can only pick the best items*

- **User story 5 name:** *Personal Ratings*
- **User story 5 “As a…” format:** *As a user, I want people to be able to rate my recipes so that I can build a personal brand that people can refer to for future recipes*

- **User story 6 name:** *Search by Ingredients*
- **User story 6 “As a…” format:** *As a user, I want to search for recipes by ingredients so that I can find recipes based on what I have available in my fridge.*

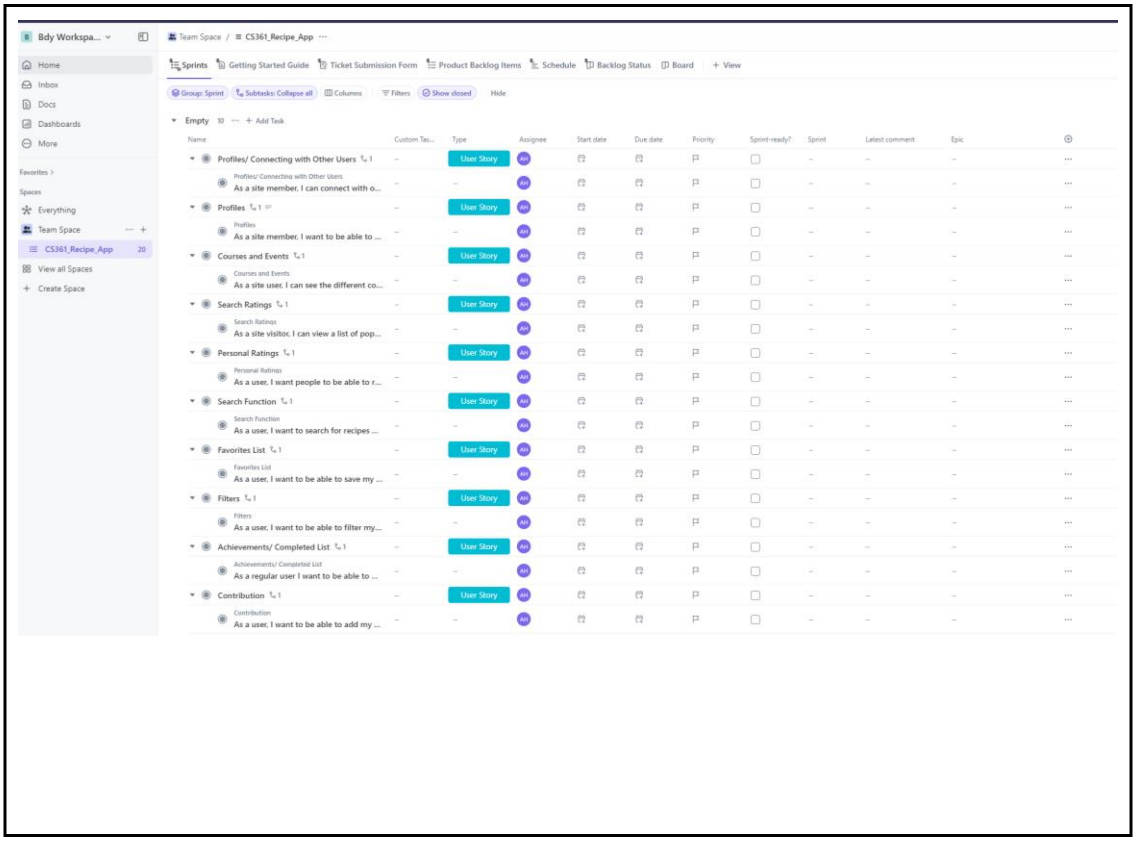
- **User story 7 name:** *Favorites List*
- **User story 7 “As a…” format:** *As a user, I want to be able to save my favorite recipes to my favorites list so that I can quickly access them without having to search again*

- **User story 8 name:** *Filtering by Prep Time*
- **User story 8 “As a…” format:** *As a user, I want to be able to filter my search results based off of preparation and cooking time so that I can plan my cooking schedule.*

- **User story 9 name:** *Achievements/ Completed*
- **User story 9 “As a…” format:** *As a regular user I want to be able to mark recipes I have completed / tried out, so that I can track which recipes I have already tried*

- **User story 10 name:** *Contribution*
- **User story 10 “As a…” format:** *As a user, I want to be able to add my recipes and ingredients so that I can contribute to the recipe database*

Enter the user stories into your task management system in a **Product Backlog column/section/category**. Take a **screenshot** so that the grader can confirm you added the stories.



Name	Custom Tas...	Type	Assignee	Start date	Due date	Priority	Sprint-ready?	Sprint	Latest comment	Epic
Profiles/ Connecting with Other Users 1	–	User Story	AH				□	–	–	–
Profiles/ Connecting with Other Users As a site member, I can connect with o...	–	–	AH				□	–	–	–
Profiles 1	–	User Story	AH				□	–	–	–
Profiles As a site member, I want to be able to ...	–	–	AH				□	–	–	–
Courses and Events 1	–	User Story	AH				□	–	–	–
Courses and Events As a site user, I can see the different co...	–	–	AH				□	–	–	–
Search Ratings 1	–	User Story	AH				□	–	–	–
Search Ratings As a site visitor, I can view a list of pop...	–	–	AH				□	–	–	–
Personal Ratings 1	–	User Story	AH				□	–	–	–
Personal Ratings As a user, I want people to be able to r...	–	–	AH				□	–	–	–
Search Function 1	–	User Story	AH				□	–	–	–
Search Function As a user, I want to search for recipes ...	–	–	AH				□	–	–	–
Favorites List 1	–	User Story	AH				□	–	–	–
Favorites List As a user, I want to be able to save my ...	–	–	AH				□	–	–	–
Filters 1	–	User Story	AH				□	–	–	–
Filters As a user, I want to be able to filter my...	–	–	AH				□	–	–	–
Achievements/ Completed List 1	–	User Story	AH				□	–	–	–
Achievements/ Completed List As a regular user I want to be able to ...	–	–	AH				□	–	–	–
Contribution 1	–	User Story	AH				□	–	–	–
Contribution As a user, I want to be able to add my ...	–	–	AH				□	–	–	–

3) Quality Attributes

Quality attributes can help guide the entire development of your project. They can remind you (and other developers) what aspects of your project matter the most and can help you decide which features to implement and in what way.

Select the top three quality attributes you care about for your individual project. See https://en.wikipedia.org/wiki/List_of_system_quality_attributes for ideas.

a) **Which three quality attributes did you choose? Name** and **define** each**.**

- **Quality attribute 1:** *Usability*
- **Quality attribute 1 definition:** *The app will be easy to navigate and allow users to interact with the program with clear instructions and labels*

- **Quality attribute 2:** *Reliability*
- **Quality attribute 2 definition:** *The app will be reliable and be relatively free of any errors to provide a good user experience*

- **Quality attribute 3:** *Availability*
- **Quality attribute 3 definition:** *The app will be available for 99.99% of the time that its running*

b) **Why did you choose these quality attributes?** Explain how each quality attribute is particularly relevant to your particular project (1+ sentence per quality attribute)

- **Why quality attribute 1 is relevant to your project:** *Usability is important* *because as a user, I would want the app im using to be easy to navigate and able to provide the results or output that I requested without too much difficulty.*

- **Why quality attribute 2 is relevant to your project:** *Having a software/ app be reliable because it should provide the service that its supposed to provide without issues with crashing or errors.*
- **Why quality attribute 3 is relevant to your project:** *Having an app be available when the user wants it should be a key attribute. If an app isn't available for the user, they are likely to go somewhere else.*

4) Sprint 1 Plan (for Milestone #1)

a) What is your **Sprint Goal**?

First Sprint Goal will be to implement a simple command line interface program using python to create a basic UI for the program.

b) Next, you will need to select **at least three** user stories from your Product Backlog and move them to your Sprint Backlog. Because you will be implementing these user stories during the Sprint, you need to write more specific requirements in the form of **acceptance criteria**.

Acceptance criteria can cover both functional and non-functional requirements. The non-functional requirements can serve to carry through your intention to reflect quality attributes.

Some developers write their user stories on 3" by 5" index cards: The user story name and "As a" format go on the front of the card and the acceptance criteria can go on the back. **Example**:

(Front of index card)

Automatic IMDB

As a user speaking during a conversation, I want to automatically see the IMDB.com webpage for the movie I'm talking about, so that I can continue with my conversation and examine the webpage as needed.

(Back of index card)

Acceptance criteria

Functional requirements

- Given a person is speaking in English at 60 dB or louder, when the software is at least 80% sure it knows what movie the person is talking about, then it will open and focus the default web browser and navigate to the movie's IMDB.com webpage.

Quality attributes & Non-functional requirements

- Responsiveness: Once the software is 80% sure about what movie is being spoken about, it will display the movie's IMDB.com webpage within 3 seconds.

Use this format to fill out each of your Sprint Backlog user stories.

Assignment requirements for Sprint Backlog user stories:

- For each of the three (or more) user stories…
  - The front of the card must contain the user story's name and "As a" format
  - The back of the card must contain at least one functional requirement and each functional requirement must use the "Given… when… then…" format.
- Each of your three quality attributes must appear at least once on a user story's "back of index card" and must be converted to a non-functional requirement.
- All of the functional and non-functional requirement must be testable.

Later, you will be asked to show that your functional and non-functional requirements are met.

First user story

(Front of index card)

Navigating the Interface

As a user who prefers using the command line, I want to be able to navigate through recipe categories quickly and efficiently, So that I can find and select recipes of interest with ease.

(Back of index card)

Acceptance criteria

Functional requirements

- *Given a user who runs the recipe application, when the user is at the main page, then the user should see a clear navigation system that allows the user to select the items that they want.*

Quality attributes & Non-functional requirements

- *Usability: When the user opens up the application a simple interface with clear instructions and labels should provide the user with exactly what they are looking for without having to spend too much time with navigation*

Second user story

(Front of index card)

Search by Ingredients

As a user, I want to search for recipes based on the ingredients so that I can find recipes based on what I have available in my fridge

(Back of index card)

Acceptance criteria

Functional requirements

- *Given a user who wants to select a recipe based off ingredients, when the user selects the "search by ingredients" filter, then the application will provide a list of recipes based off the ingredients checked*

Quality attributes & Non-functional requirements

- *Accuracy: Once user has selected the appropriate category to search for recipes by ingredients, the application should provide a reliable method for returning / outputting the recipe with at least 95% confidence*

Third user story

(Front of index card)

Filter by Preparation Time

As a user, I want to be able to filter my search results based off of preparation and cooking time so that I can plan my cooking schedule accordingly.

(Back of index card)

Acceptance criteria

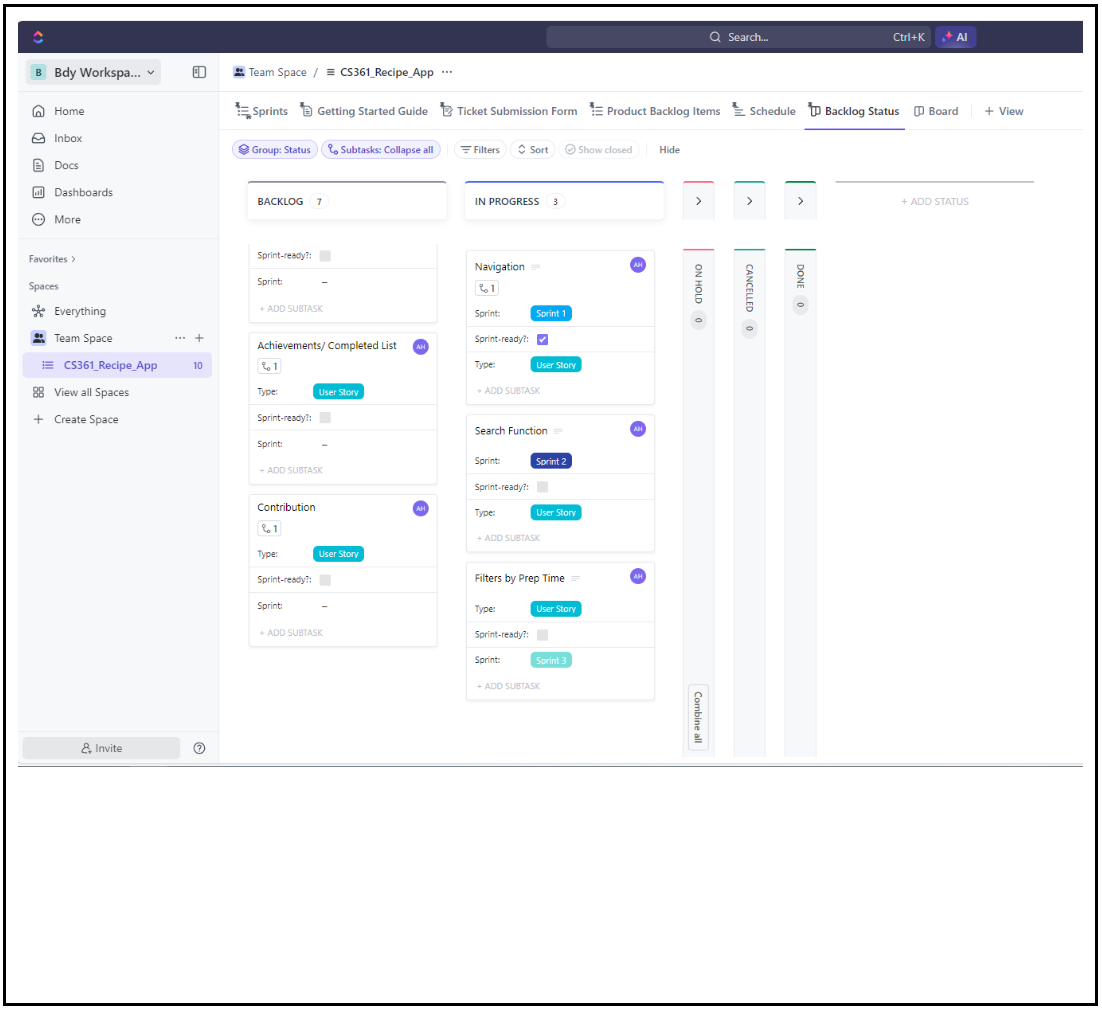
Functional requirements

- *Given a user who wants to select recipes based off how long it will take to prepare the recipes, when the user selects a dropdown list that provides a range of different preparation time, then the application will provide a list of all the recipes that match those criteria.*

Quality attributes & Non-functional requirements

- *Reliability: The app should reliably maintain consistent performance in filtering recipes based on preparation time under varying conditions and loads*

c) Take a **screenshot** that shows you've moved these user stories into a Sprint Backlog in your task management system.



BACKLOG 7

Sprint-ready?: □
Sprint: –
+ ADD SUBTASK

Achievements/ Completed List
Type: User Story
Sprint-ready?: □
Sprint: –
+ ADD SUBTASK

Contribution
Type: User Story
Sprint-ready?: □
Sprint: –
+ ADD SUBTASK

IN PROGRESS 3

Navigation
Sprint: Sprint 1
Sprint-ready?: ☑
Type: User Story
+ ADD SUBTASK

Search Function
Sprint: Sprint 2
Sprint-ready?: □
Type: User Story
+ ADD SUBTASK

Filters by Prep Time
Type: User Story
Sprint-ready?: □
Sprint: Sprint 3
+ ADD SUBTASK

Your **Definition of Done** for the Sprint would typically include, “The acceptance criteria are satisfied for all Sprint Backlog user stories.” You aren’t required to write your DoD or put it in your task management system.

This would also be **a good time to break each of your user stories into a list of specific tasks** you need to complete. Task management systems are, as you might imagine, a great place to do that!

Submission

PDF or Word format via Canvas.

You must follow instructions at Modules > 'HOW TO: Attach a Document to "Text Entry" Field'.

Grading

You are responsible for satisfying all criteria listed in the Canvas rubric for this assignment. You will be able to revise this assignment if you miss points.

Questions?

Please ask via Ed so that others can benefit from the answer.