Oregon State University
Ecampus

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?

Asana

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?

Develop a command-line program that provides a variety of Japanese vocabulary quizzes to users. At the end of the quizzes, the program provides a score to the user.

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:** *Implement the main menu UI and functionality*.
- **User story 1 "As a…" format:** *As a user, I want to see the main menu page so that I can start the quiz or exit the quiz by entering the corresponding options*.

- **User story 2 name:** *Implement About page functionality*.
- **User story 2 "As a…" format:** *As a user, I want to access to the About page option so that I can obtain the information about the program.*

- **User story 3 name:** *Display the quiz selection menu and take user input*.
- **User story 3 "As a…" format:** *As a user, I want to see the quiz menu page so that I can select the level of quizzes (beginner, intermediate, and advanced).*
-
- **User story 4 name:** *Implement the quiz question section UI*.
- **User story 4 "As a…" format:** *As a user, I want to see the quiz questions which are in the multiple-choice format so that I can select the correct option among 4 options.*

- **User story 5 name:** *Implement the user answer section UI*
- **User story 5 "As a…" format:** *As a user, I want to see only one correct option so that I can select the option by entering the corresponding number.*
-

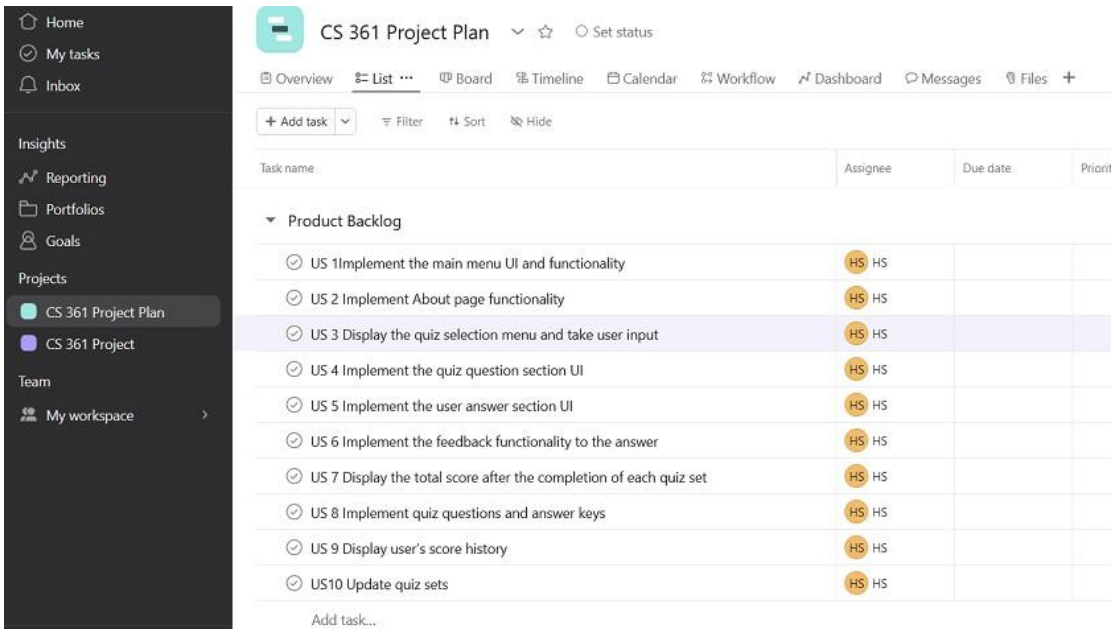
- **User story 6 name:** *Implement the feedback functionality to the answer*.
- **User story 6 "As a…" format:** . *As a user, I want to know the correct answer so that I can learn if my answer is correct or not.*
-
- **User story 7 name:** *Display the total score after the completion of each quiz set*.
- **User story 7 "As a…" format:**. *As a user, after each quiz set execution, I want to see the total score so that I can learn how well I did.*

- **User story 8 name:** *Implement quiz questions and answer keys*.
- **User story 8 "As a…" format:** *As a user, I want to try more quiz questions so that I can learn more words.*

- **User story 9 name:** *Display user's score history*
- **User story 9 "As a…" format:** *As a user, I want to access the score history of the past my quiz results so that I can see my progress.*

- **User Story 10 name:** *Update quiz sets*
- **User Story 10 "As a…**" format: *As a user, I want to try a new quiz set by selecting the new quiz set option so that I can learn new vocabulary*.

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.



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:** *A user can easily use the product*.

- **Quality attribute 2:** *Localizability*
- **Quality attribute 2 definition:** *The product supports the target language*.

- **Quality attribute 3:** *Accuracy*
- **Quality attribute 3 definition:** *The content and results which the product provides are correct*.

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:** *This is a command-line quiz program which a user interacts with. The program UI should be user-friendly, and a user can easily navigate*.
- **Why quality attribute 2 is relevant to your project:** *This product is designed so that a user can learn Japanese Vocabulary, and this product should display Japanese language correctly*.
- **Why quality attribute 3 is relevant to your project:** *This product is a quiz program, all the quiz questions and answers are correct and score correctly*.

4) **Sprint 1 Plan (for Milestone #1)**

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

Implement the UI for the main screen and the quiz screen

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) ***Implement the main menu UI and functionality.*** *As a user, I want to see the main menu page so that I can start the quiz or exit the*

quiz by entering the corresponding options.

(Back of index card)

Acceptance criteria

Functional requirements

- Given a user *enters "S" and the program Start, then a user enters "Q", the program exits.*
- *Given a user enters an incorrect option, and the message to ask a user to enter the option again, then the user can re-enter the option.*

Quality attributes & Non-functional requirements

- *Usability. The screen should display the options clearly and provides errors and navigates the user properly if a user enters incorrect input.*

Second user story

(Front of index card)

Display the quiz selection menu and take user input.

- *As a user, I can see the quiz menu page so that I can select the level of quizzes (beginner, intermediate, and advanced).*

(Back of index card)

Acceptance criteria

Functional requirements

- *Given a user enters B, and the beginner level question set is displayed, then the user can start the beginner level question set.*
- *Given a user enters I, and the intermediate level question set is displayed, then a user start the intermediate level question set.*
- *Given a user enters A, and the advanced level question set is displayed, then a user start the advanced level question set.*
- *Given a user enters an incorrect option, and the message to ask a user to enter the option again, then the user can re-enter the option.*

Quality attributes & Non-functional requirements

- *Usability: A user can easily select the options and the program takes user input accurately and navigates the user to the level which the user selected. Also, if the user enters incorrect input, a user is prompted to enter correct options.*

Third user story

(Front of index card)

Implement the quiz question section UI

- *As a user, I can see the quiz questions which is in the multiple-choice format.*

(Back of index card)

Acceptance criteria

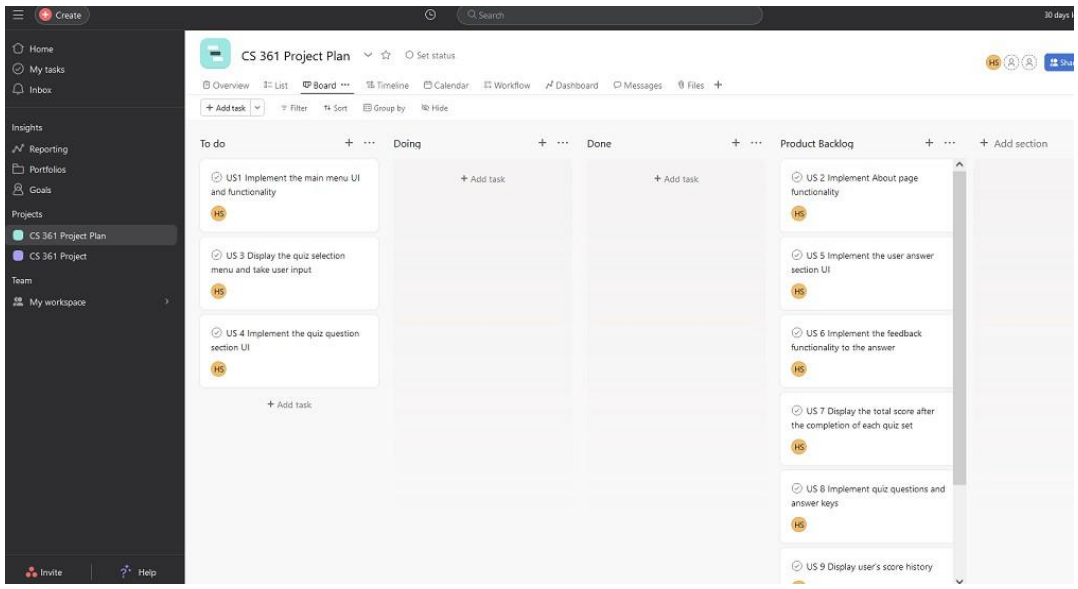
Functional requirements

- *Given a user selects the level of quiz questions, and the first question is displayed to the user, and the format of the question is a multiple-choice format, then the user can see one Japanese word and there are four options in English.*

Quality attributes & Non-functional requirements

- *Localization: The Japanese word which is displayed uses Japanese writing system and displayed correctly.*
- *Accuracy: The Japanese word is correctly displayed and there is only one correct answer in the options.*

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



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.