



## CS361: Assignment 3: Project Plan & Sprint 1 Plan (for MVP)

### 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. The timespan for the user stories is this term but you don't have to finish implementing all the user stories this term. These user stories will need to go in a **Product Backlog** column/section/category of your task management system.

Next, move some of those user stories into a **Sprint Backlog**—these will be the user stories you WILL complete during this Sprint (for your minimum viable product) and comprise your **Sprint Plan**. **Note these minimum requirements for your MVP:**

- Has a way for users to interact (e.g., provide input, push buttons, etc.)
- Reflects CSH
- Reflects three quality attributes you choose
  - *Hint: If you choose “usability” or “inclusivity” as a quality attribute, your corresponding non-functional requirement can involve the CSH*

### 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 based on the spike you did previously.

**Which task management system** did you choose?

Jira

#### 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 setting. You will, however, be experiencing Scrum Events and Artifacts.

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

*Develop an app that will randomly choose an eat-out destination for the user: When people say “I don’t know what I want to eat”, they may want options presented to them in order to decide whether or not they want that particular food option. Other*

*times, when people say “I don’t care what I eat”, they truly don’t care and the options are overwhelming so they just want someone to pick for them. This app will shuffle through food options randomly for both of these scenarios.*

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 software I can use to communicate with my dog when I’m on vacation.”

- 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.

#### Requirements for user stories:

- Each has a **name**
- Each uses the “**As a... I want to... so that...**” **format** (see textbook)
- Total of at least **20** user stories (you will not have to implement all of these)
- As a set, must have **no obvious violations of INVEST**

	Name	As a... I want to... so that... format
1	Randomizer	<i>As a Diner, I want the app to randomly pick a dining option for me so that I don't have to pick one myself</i>
2	Shuffle	<i>As a Diner, I want to be able to shuffle through different options so that I can see another option if the first option isn't to my liking</i>
3	Price Filter	<i>As a Diner, I'd want the app to only show my dining options within a certain price point so that it doesn't give me an option outside of my budget</i>
4	Location Filter	<i>As a Diner, I'd want the app to only show my dining options within a certain area so that it doesn't give me an option outside of my traveling range</i>
5	Cuisine Filter	<i>As a Diner, I'd want the app to only show my dining options for certain cuisines so that it doesn't give me an option for cuisines that I know I do not like.</i>
6	Display Menu	<i>As a Diner, I want to be able to pull up the menu once a dining option has been presented to me so that I can make sure they have something I want</i>
7	Share Results	<i>As a Diner, I want to be able to share my dining results with my friends</i>
8	Rating	<i>As a Diner, I want to be able to rate the dining option after I have tried it so that other diners can benefit from my experience</i>

<b>9</b>	<b>Personal Favorites</b>	<i>As a Diner, I want to be able to add the presented dining options to my list so that if I enjoyed the experience, I can look it up the next time.</i>
<b>10</b>	<b>Review</b>	<i>As a Diner, I want to be able to leave a review for the restaurant so that other diners can benefit from my experience.</i>
<b>11</b>	<b>Wait Time</b>	<i>As a Diner, I want to be able to see an approximate wait time for the option presented to me so that I know whether or not I have time to eat there.</i>
<b>12</b>	<b>Discounts</b>	<i>As a Business, I want to be able to give customers discounts if my restaurant shows up as an option so that customers will be incentivized to try out my restaurant</i>
<b>13</b>	<b>Option to be added or removed from the list</b>	<i>As a Business, I want to be able to put my restaurant on or off the list of possible options for customers so that if my restaurant undergoes change, it is not advertised to new customers during the change.</i>
<b>14</b>	<b>Non-Dine in Options</b>	<i>As a Diner, I want to be able to see whether or not the option presented to me can be delivered or picked up so that I know whether or not I have to go into the restaurant for the particular option</i>
<b>15</b>	<b>Diner Sign-up</b>	<i>As a Diner, I want to be able to create an account and sign-up so that I do not have to enter my information each time I use the app</i>
<b>16</b>	<b>Don't display if closed</b>	<i>As a Diner, I don't want to see a restaurant if it is closed so I don't spend time looking through the menu and end up not being able to dine there</i>
<b>17</b>	<b>Email Alerts</b>	<i>As a Business, I want to be able to send my customer's promotional events so that I can promote new initiatives</i>
<b>18</b>	<b>History</b>	<i>As a Diner, I want to be able to see the history of the restaurants that I ended up going to so that I have a curated list of restaurants that I know I've tried</i>
<b>19</b>	<b>Business Sign-up</b>	<i>As a Business, I want to be able to create an account so that I can upload my restaurant information for customers to see</i>
<b>20</b>	<b>Blacklist</b>	<i>As a Diner, I want to be able to blacklist a restaurant so that if I don't like that restaurant, it will never show up as an option for me.</i>

Enter these user stories into your task management system. Take a **screenshot** so that the grader can confirm you added the stories.

## ScreenshotOfUserStoriesList

The screenshot shows the Jira Software interface for the 'CS361 Dining Option' project. The left sidebar includes sections for Planning (Roadmap, Backlog, Board), Development (Code, Project pages, Add shortcut, Project settings), and a note about being a team-managed project. The main area displays the 'Backlog' with 20 items:

- DINE-1 Randomizer
- DINE-2 Shuffle
- DINE-3 Price Filter
- DINE-4 Location Filter
- DINE-5 Cuisine Filter
- DINE-6 Display Menu
- DINE-7 Share Results
- DINE-8 Rating
- DINE-9 Personal Favorites
- DINE-10 Review
- DINE-11 Wait Time
- DINE-12 Discounts
- DINE-13 Option to be added or removed from the list
- DINE-14 Non-Dine in Options
- DINE-15 Diner Sign-up
- DINE-16 Don't display if closed
- DINE-17 Email Alerts

A 'Quickstart' sidebar on the right provides links to create a project, deliver more often with scrum, and create an issue.

This screenshot shows the same Jira interface as the first one, but the backlog now contains 17 items:

- DINE-7 Share Results
- DINE-8 Rating
- DINE-9 Personal Favorites
- DINE-10 Review
- DINE-11 Wait Time
- DINE-12 Discounts
- DINE-13 Option to be added or removed from the list
- DINE-14 Non-Dine in Options
- DINE-15 Diner Sign-up
- DINE-16 Don't display if closed
- DINE-17 Email Alerts
- DINE-11 Wait Time
- DINE-12 Discounts
- DINE-13 Option to be added or removed from the list
- DINE-14 Non-Dine in Options
- DINE-15 Diner Sign-up
- DINE-16 Don't display if closed
- DINE-17 Email Alerts

This screenshot shows the backlog again with 20 items, identical to the first one but with a blue '+' icon at the bottom left of the backlog table.

--

### 3) Non-Functional Requirements

Select the top three quality attributes you care about for your individual project. See [https://en.wikipedia.org/wiki/List\\_of\\_system\\_quality\\_attributes](https://en.wikipedia.org/wiki/List_of_system_quality_attributes) for ideas. Convert them into **non-functional requirements**.

- a) **Which three quality attributes** did you choose?

*Simplicity*

*Orthogonality*

*Fidelity*

- b) **Why did you choose these quality attributes?** Ex: How are they relevant to your end user? How are they relevant to your Product Goal? (1+ sentence per quality attribute)

Quality attribute	Why it's relevant to YOUR project
<i>Simplicity</i>	<i>The app should be simple such that it takes no time for the user to learn the app. This app's goal is to provide convenience to the user which makes this the most important attribute.</i>
<i>Orthogonality</i>	<i>This app will connect to many different API's such as location services and delivery services which means it can start to create a lot of intricate dependencies. Constructing the app with an orthogonal attribute will reduce potential failure when updating or changing parts of the codebase.</i>
<i>Fidelity</i>	<i>This app directs users to a dining location. Fidelity is incredibly important to avoid users going to the wrong location and not getting the dining experience they desired.</i>

- c) Convert your quality attributes to non-functional requirements. Must be **testable**. **What are your three non-functional requirements? How would you test them?** (1+ sentence each)

Quality Attribute 1	<i>Simplicity</i>
Non-functional requirement 1	<i>User should see the app and intuitively know how to use it</i>

<b>How you would test that the non-functional requirement is satisfied</b>	<i>The “how-to” link should be clicked less than 90% of the time the app is visited</i>
--	---

<b>Quality Attribute 2</b>	<i>Orthogonality</i>
<b>Non-functional requirement 2</b>	<i>When a new feature is added, or when an existing feature is modified or deleted, no other feature’s function should change.</i>
<b>How you would test that the non-functional requirement is satisfied</b>	<i>Test and make sure feature functions output the same results each time another feature is added, changed, or deleted</i>

<b>Quality Attribute 3</b>	<i>Fidelity</i>
<b>Non-functional requirement 3</b>	<i>The app should represent the true states of the restaurants and be as described</i>
<b>How you would test that the non-functional requirement is satisfied</b>	<i>When restaurants are up on the app, users should not report that the restaurant is closed 95% of the time</i>

#### 4) Sprint 1 Plan (for MVP)

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

*The Sprint Goal is to implement a functioning product that allows users and businesses to sign up, and to randomize the businesses for users to choose from*

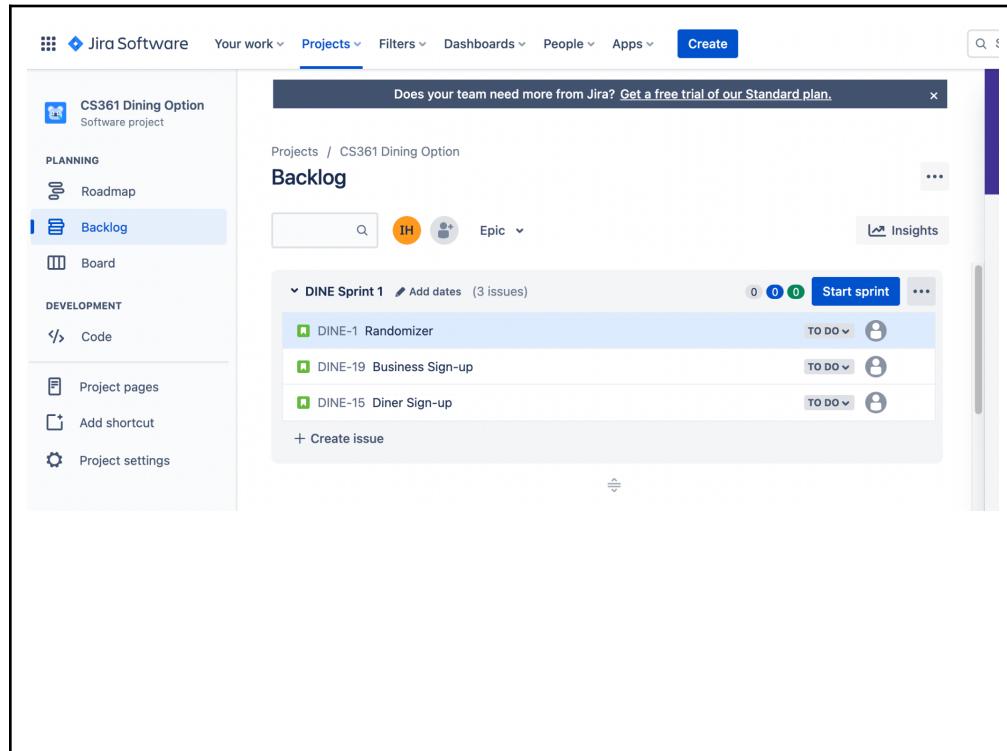
- b) Next, select user stories from your Product Backlog and move them to your **Sprint Backlog**. Choose what you think you can complete, but recall the minimum requirements (see Overview). Which user stories did you select?

*As a Diner, I want to be able to create an account and sign-up so that I do not have to enter my information each time I use the app*

*As a Business, I want to be able to create an account so that I can upload my restaurant information for customers to see*

*As a Diner, I want the app to randomly pick a dining option for me so that I don't have to pick one myself*

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



- d) Now that you've selected your Sprint user stories, you will define several tasks for each user story—the tasks you'll need to complete to complete the user story. Add tasks to each user story and take **screenshots** showing each user story's tasks in your task management system. **Note:** It might work best to make your user story a task and your tasks subtasks of the user story.

The screenshot shows the Jira interface with a yellow box highlighting the 'Create HTML Main App Page' dialog. The dialog contains fields for creating a new issue, including 'Description' and 'Details' sections. The background shows the 'Backlog' view for the 'DINE Sprint 1' sprint, with the first three issues (DINE-1 Randomizer, DINE-19 Business Sign-up, DINE-15 Diner Sign-up) highlighted.

Does your team need more from Jira? [Get a free trial of our Standard plan.](#) ×

Projects / CS361 Dining Option

## Backlog

IH + Epic ...

DINE Sprint 1 Add dates

- DINE-1 Randomizer
- DINE-19 Business Sign-up
- DINE-15 Diner Sign-up

+ Create issue

DINE Sprint 1 Add dates

DINE-1 Randomizer

DINE-19 Business Sign-up

DINE-15 Diner Sign-up

+ Create issue

Backlog (17 issues)

- DINE-2 Shuffle
- DINE-3 Price Filter
- DINE-4 Location Filter

DINE-1 Randomizer / DINE-28 1 Like ... X

**Create Mechanism to randomize all the businesses stored in database**

Attach Link issue ...

To Do ▼

**Description**  
Add a description...

**Details**

IH Add a comment...

Pro tip: press **M** to comment

Does your team need more from Jira? [Get a free trial of our Standard plan.](#) ×

Projects / CS361 Dining Option

## Backlog

IH + Epic ...

DINE Sprint 1 Add dates

- DINE-1 Randomizer
- DINE-19 Business Sign-up
- DINE-15 Diner Sign-up

+ Create issue

DINE Sprint 1 Add dates

DINE-1 Randomizer

DINE-19 Business Sign-up

DINE-15 Diner Sign-up

+ Create issue

Backlog (17 issues)

- DINE-2 Shuffle
- DINE-3 Price Filter
- DINE-4 Location Filter

DINE-19 Business Sign-up / DINE-21 1 Like ... X

**Create HTML page for Business sign-up**

Attach Link issue ...

To Do ▼

**Description**  
Add a description...

**Details**

Assignee Unassigned

IH Add a comment...

Pro tip: press **M** to comment

Does your team need more from Jira? [Get a free trial of our Standard plan.](#) x

Projects / CS361 Dining Option

## Backlog

Search | IH | +1 | Epic ▾ | Insights

**DINE Sprint 1** Add dates

- DINE-1 Randomizer
- DINE-19 Business Sign-up
- DINE-15 Diner Sign-up

+ Create issue

**Backlog (17 issues)**

- DINE-2 Shuffle
- DINE-3 Price Filter
- DINE-4 Location Filter

DINE-19 / DINE-22 IH 1 Like ... X

### Create Database to store Business information

Attach Link issue ▾ ...

To Do ▾

Description

Add a description...

Details

IH Add a comment... Pro tip: press M to comment

Does your team need more from Jira? [Get a free trial of our Standard plan.](#) x

Projects / CS361 Dining Option

## Backlog

Search | IH | +1 | Epic ▾ | Insights

**DINE Sprint 1** Add dates

- DINE-1 Randomizer
- DINE-19 Business Sign-up
- DINE-15 Diner Sign-up

+ Create issue

**Backlog (17 issues)**

- DINE-2 Shuffle
- DINE-3 Price Filter
- DINE-4 Location Filter

DINE-19 / DINE-23 IH 1 Like ... X

### Create mechanism to submit business information into database

Attach Link issue ▾ ...

To Do ▾

Description

Add a description...

Details

IH Add a comment... Pro tip: press M to comment

Does your team need more from Jira? [Get a free trial of our Standard plan.](#) x

Projects / CS361 Dining Option

## Backlog

IH + Epic ...

▼ DINE Sprint 1 Add dates

- DINE-1 Randomizer
- DINE-19 Business Sign-up
- DINE-15 Diner Sign-up

+ Create issue ^

▼ Backlog (17 issues)

- DINE-2 Shuffle
- DINE-3 Price Filter
- DINE-4 Location Filter

DINE-15 / DINE-24 o 1 ... X

### Create HTML Diner sign-up Page

Attach Link issue ...

To Do ▼

Description  
Add a description...

Details ^

Assignee Unassigned

IH Add a comment... Pro tip: press M to comment

Does your team need more from Jira? [Get a free trial of our Standard plan.](#) x

Projects / CS361 Dining Option

## Backlog

IH + Epic ...

▼ DINE Sprint 1 Add dates

- DINE-1 Randomizer
- DINE-19 Business Sign-up
- DINE-15 Diner Sign-up

+ Create issue ^

▼ Backlog (17 issues)

- DINE-2 Shuffle
- DINE-3 Price Filter
- DINE-4 Location Filter

DINE-15 / DINE-25 o 1 ... X

### Create Database to store Diner information

Attach Link issue ...

To Do ▼

Description  
Add a description...

Details ^

Assignee Unassigned

IH Add a comment... Pro tip: press M to comment

The screenshot shows a Jira backlog for the project 'CS361 Dining Option'. On the left, there's a sidebar with a search bar, a 'IH' button, a user icon, and an 'Epic' dropdown. Below that is a section for 'DINE Sprint 1' containing issues like 'DINE-1 Randomizer', 'DINE-19 Business Sign-up', and 'DINE-15 Diner Sign-up'. Another section for 'Backlog (17 issues)' lists 'DINE-2 Shuffle', 'DINE-3 Price Filter', and 'DINE-4 Location Filter'. On the right, a detailed view of a user story card for 'DINE-15 Diner Sign-up' is shown. The card has a title 'Create Mechanism to submit Diner information to database', a 'To Do' section, a 'Description' field with placeholder 'Add a description...', a 'Details' section, and a comment input field with placeholder 'Add a comment...'. A note at the bottom says 'Pro tip: press M to comment'.

- e) Lastly, you'll need a **Definition of Done** for each user story. Figure out where to put this on your user story task in your task management system (e.g., in the task's description area).

#### Requirements for each Definition of Done:

- Placed with its user story
- Has at least four acceptance criteria
- Acceptance criteria should include checking that your **non-functional requirements** are met
- Acceptance criteria use the “Given... when...” format

Enter your Definitions of Done below and in your task management system.

### User Story Definition of Done 1

Which **user story** is this Definition of Done for?

*As a Diner, I want to be able to create an account and sign-up so that I do not have to enter my information each time I use the app*

What is its **Definition of Done**?

Acceptance criterion	For which quality attribute (if any)?
<i>Given the diner user is first using the app when they input their information then the database should be populated with the new Diner</i>	<i>None, Functional</i>

<i>Given I'm an Abi-like user, when I see the app, I should immediately know how to use it</i>	<i>Simplicity</i>
<i>Given I'm a Tim-Like user, when I use the app, all of the features offered should be functional</i>	<i>Orthogonality</i>
<i>Given I'm a Pat-like user, when I use the app, all functions must be working as expected</i>	<i>Fidelity</i>

## User Story Definition of Done 2

Which **user story** is this Definition of Done for?

*As a Business, I want to be able to create an account so that I can upload my restaurant information for customers to see*

What is its **Definition of Done**?

Acceptance criterion	For which quality attribute (if any)?
<i>Given the business user is first using the app when they input their information then the database should be populated with the new business</i>	<i>None, Functional</i>
<i>Given I'm an Abi-like user, when I see the app, I should immediately know how to use it</i>	<i>Simplicity</i>
<i>Given I'm a Tim-Like user, when I use the app, all of the features offered should be functional</i>	<i>Orthogonality</i>
<i>Given I'm a Pat-like user, when I use the app, all functions must be working as expected</i>	<i>Fidelity</i>

## User Story Definition of Done 3

Which **user story** is this Definition of Done for?

*As a Diner, I want the app to randomly pick a dining option for me so that I don't have to pick one myself*

What is its **Definition of Done**?

Acceptance criterion	For which quality attribute (if any)?
<i>Given the user is trying to find a dining option, when they press the randomize button, then they should see a restaurant option pop up.</i>	<i>None, Functional</i>

<i>Given I'm an Abi-like user, when I see the app, I should immediately know how to use it</i>	Simplicity
<i>Given I'm a Tim-Like user, when I use the app, all of the features offered should be functional</i>	Orthogonality
<i>Given I'm a Pat-like user, when I use the app, all functions must be working as expected</i>	Fidelity

Add more user story Definitions of Done as needed

Take a **screenshot** of at least one user story's Definition of Done in your task management system for your grader to confirm.

**Business Sign-up**

**Description**

As a Business, I want to be able to create an account so that I can upload my restaurant information for customers to see

**Child issues**

Issue	Description	Status
DINE-21	Create HTML page fo...	TO DO
DINE-22	Create Database to s...	TO DO
DINE-23	Create mechanism to...	TO DO

**Activity**

Show: All Comments History Newest first ↴

Add a comment... (by IH)

Pro tip: press M to comment

**Details**

Definition of Done  
User is able to input business information, submit it, and have it stored in the database

Assignee  
Unassigned

Labels  
None

Sprint  
DINE Sprint 1

Story point estimate  
None

Reporter  
I Chun Hung

Created 4 hours ago  
Updated 4 hours ago

Now that you have a plan, begin implementation!

## Submission

PDF or Word format via Canvas.

## **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.