

# FULL STACK

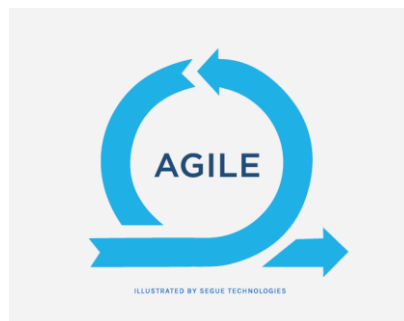


## Planning Testing with Agile

# You Already Know

## Skills:

Agile



Git



SQL



## Agile

An iterative approach to manage the development of a software project.

## Git and GitHub

A distributed version control system that helps handle software projects.

## SQL or MySQL

Relational database management system to store data in a structured way using tables.



# A Day in the Life of an Automation Test Engineer

As Automation Test Engineers, our key role is to test both client and server software with the latest test automation tools.

We shall be testing food delivery application built in Angular, Node as the front end with Spring Boot, Java, and MySQL/MongoDB as the backend.

To begin, we need to plan testing properly using Agile methodology.

Firstly, we will create user stories for epics of web app for end user and admin dashboard.

We will thereafter see how to use Jira and plan the project with our stories.





# Learning Objectives

By the end of this lesson, you will be able to:

- 👁 Understand the various aspects in developing the user stories
- 👁 Develop user stories for admin dashboard epic and web app for end user epic
- 👁 Perform hands-on Agile software development with Jira
- 👁 Develop stories and plan sprints in Jira



## Task 1: User Stories for Web Admin Dashboard Epic



# Authentication Story

## Summary

Users should be able to log in to the admin dashboard.

## Description

To verify as admin user, they will use email and password with captcha verification to access the admin backend.

## Acceptance criteria

There must be a login page featuring text fields for username and password and a login button.

## NFR (Nonfunctional Requirements)

After logging in, the user should be navigated to the Dashboard page in less than three seconds.

# Restaurant Story

## Summary

Users should be able to manage restaurants as an admin.

## Description

As admin users, they want to be able to add and change restaurant information as well as manage dishes and update information.

## Acceptance criteria

The screen should display a list of all the dishes of the restaurant. In the same screen, a button to add dish should be shown. Dish details form should appear on clicking the button. Further, to update or delete the dish, separate buttons should be available.



# Restaurant Story

## Validation

At least, the name, email, and contact must be added for a restaurant.

## NFR (Nonfunctional Requirements)

NA



# Dish Story

## Summary

Users should be able to manage dishes as an admin.

## Description

They want to add and edit dish specifics such as pricing, discounts, dish descriptions, dish photos, and other relevant information as admin users.

## Acceptance criteria

The screen should display a list of all the dishes of the restaurant. In the same screen, a button to add dish should be shown. Dish details form should appear on clicking the button. Further, to update or delete the dish, separate buttons should be available.

# Dish Story

## Validation

At least, the name and price of the dish must be added.

## NFR (Nonfunctional Requirements)

NA



# User Story

## Summary

Users should be able to access users' data.

## Description

As admin users, they want to monitor newly registered users. Users can be filtered by date of registration. Also, user's activities such as frequently visited sections in the end user app.

## Acceptance criteria

The user list page should be empty by default, with a prompt of user filtration criteria. In the prompt, the options of users retrieval on the basis of frequently active, last active, and date of registration should be added. After user list retrieval, the web page should be divided into two parts. In the first section, a list of users must be displayed. In the second section, the selected user's data should be displayed.



# User Story

## Validation

Date format should be validated before sending requests for users.

## NFR (Nonfunctional Requirements)

Retrieval of users should be seamless in a maximum of 10 seconds.



# Order Story

## Summary

Users should be able view new as well as finished orders as admin.

## Description

As admin user, they want to manage orders stats such as availability of dishes in order and order transition information.

## Acceptance criteria

There should be three sections on the website. A list of freshly placed orders should be included in the first section. A list of ongoing orders should be included in the second section. Orders that are on hold due to dish availability should be indicated in the third section. When a person selects an order, he should be taken to another page that has all of the information about the food he has chosen.

# Order Story

## Validation

When processing the order, the availability of dish in the inventory should be validated first.

## NFR (Nonfunctional Requirements)

NA



## Task 2: User Stories for End User Web App Epic





# Authentication Story

## Summary

Users should be able to log in to the end user app to access member features like browsing restaurants and dishes, placing order.

## Description

As end users, they want to authenticate the email, password, and captcha verification to access the member section. They should be able to register if they do not have an account first.

## Acceptance criteria

A web page with text fields for username and password with a login button to log in to the system. Another web page for registration asking name as additional field. Even though a user is not authenticated, they must be able to see the Home Page with Restaurants for browsing.

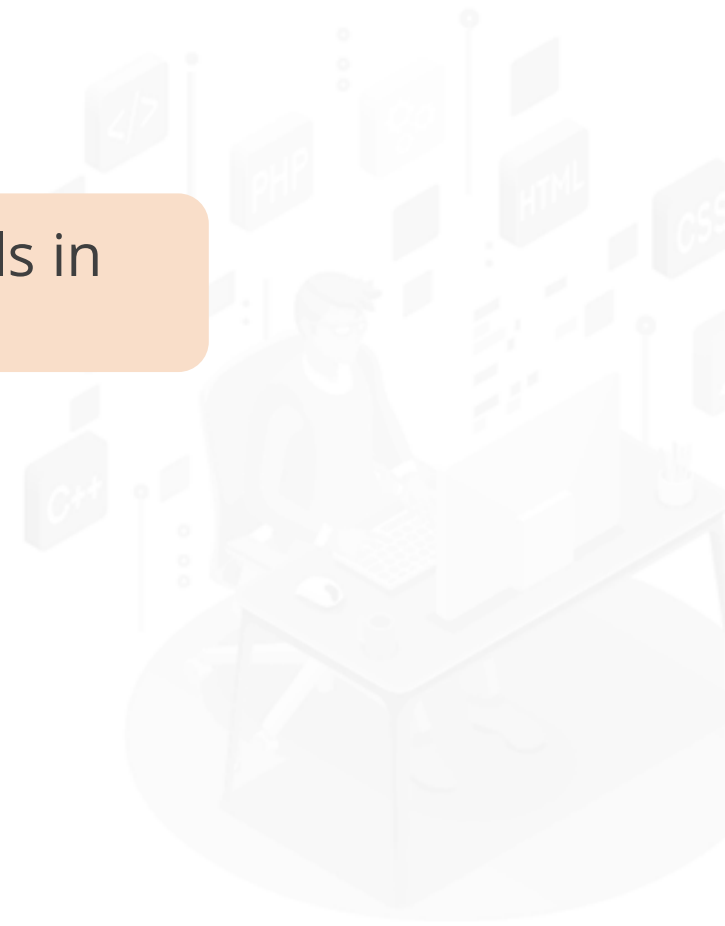
# Authentication Story

## Validation

Username and password cannot be empty. Password must be of minimum six characters.

## NFR (Nonfunctional Requirements)

After logging in, the user should be navigated to the Home Page with details in less than three seconds.



# Profile Story

## Summary

Users should be able to update his basic information.

## Description

As end users, they want to update his basic name, email, and delivery address.

## Acceptance criteria

All the fields should be mandatory to fill.

# Profile Story

## Validation

All the fields should be mandatory to fill.

## NFR (Non Functional Requirements)

NA





# Restaurant Story

## Summary

Users should be able to view restaurant details.

## Description

As end users, they want a page where restaurant's images, name, contact, user reviews and other meaningful information are displayed.

## Acceptance criteria

There must be a web page with two sections, one where the restaurant's details are displayed and second, where the description of the dishes are displayed. The user reviews should be under both the sections with the latest ones on the top.

# Restaurant Story

## Validation

Availability of dish in inventory should be checked before adding it to the order.

## NFR (Nonfunctional Requirements)

NA



# Dish Story

## Summary

Users should be able to view dish details.

## Description

As end users, they want a page where dish images, price, description, user reviews and other meaningful information is displayed.

## Acceptance criteria

There must be a web page with two sections. In the first one, dish details should be displayed vertically. In the second section, dish pricing and description should be displayed. The user reviews should be under both the sections with the latest ones on the top.

# Dish Story

## Validation

Availability of dish in inventory should be checked before adding it to the order.

## NFR (Nonfunctional Requirements)

NA





# Order Story

## Summary

Users should be able to view the checkout page for order placement.

## Description

As end users, they want a checkout page where all the dishes are displayed of the order with payment options.

## Acceptance criteria

In the checkout page, there should be two sections. In the first one, all the added dishes should be displayed vertically. In the second section, details of payment should be displayed with an option of selecting the payment mode for the order.

# Order Story

## Validation

The inventory dish and delivery of order to the selected present delivery address should be available.

## NFR (Nonfunctional Requirements)

NA



## Task 3: Working with Jira





# Different Project Options

×

Project templates

Software development

Service management

Work management

Marketing

Human resources

Finance

Design

Personal

Operations


Legal

Sales

Project templates

Software development


Plan, track and release great software. Get up and running quickly with templates that suit the way your team works. Plus, integrations for DevOps teams that want to connect work across their entire toolchain.



Kanban

Visualize and advance your project forward using issues on a powerful board.


>



Scrum **LAST CREATED**

Sprint toward your project goals with a board, backlog, and roadmap.

>



Bug tracking

Manage a list of development tasks and bugs.

>

©Simplilearn. All rights reserved.

simplilearn



# Scrum Template

×

Project templates / Software development

Project templates

Software development

Service management

Work management

Marketing

Human resources

Finance

Design

Personal

Operations

Legal

Sales

PRODUCTS

Jira Software

Scrum

Use template

×

The Scrum template helps teams work together using sprints to break down large, complex projects into bite-sized pieces of value. Encourage your team to learn through incremental delivery, self-organize while working on a problem, and regularly reflect on their wins and losses to continuously improve.

✓

✓

✓

Plan upcoming work in a backlog

Prioritize and plan your team's work on the backlog. Break down work from your project roadmap, and order work items so your team knows what to deliver first.

[Learn more about the backlog](#)

Organize cycles of work into sprints

Sprints are short, time-boxed periods when a team collaborates to complete a set amount of customer value. Use sprints to drive

PRODUCT

Jira Software

RECOMMENDED FOR

Teams that deliver work on a regular cadence

DevOps teams that want to connect work across their tools

ISSUE TYPES

+

 Epic

■

 Story

■

 Bug

✓

 Task

■

 Sub-task

Next: Select a project type

Use template

# Project Details

## Add project details

You can change these details anytime in your project settings.

Name \*

Access Anyone with access to auri bases can access and administer this project. [Upgrade your plan](#) to customize project permissions.

Key ⓘ \*

- ☐ **Connect repositories, documents, and more**  
Sync your team's work from other tools with this project for better visibility, access, and automation.

Template

Change template

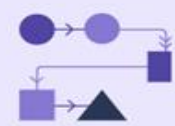


**Scrum**

Sprint toward your project goals with a board, backlog, and roadmap.

Type

Change type



**Team-managed**

Control your own working processes and practices in a self-contained space.

Cancel

Create project

# Project Home Page

The screenshot displays the Jira Software interface for a project named 'foodie'. The top navigation bar includes 'Jira Software', 'Your work', 'Projects' (selected), 'Filters', 'Dashboards', 'People', 'Apps', and a 'Create' button. A search bar is located on the right. The left sidebar shows the project 'foodie' and navigation options: 'PLANNING' (Roadmap, Backlog, Board), 'DEVELOPMENT' (Code), 'Project pages', 'Add shortcut', and 'Project settings'. The 'Board' option is selected. The main content area is titled 'FOOD board' and shows three columns: 'TO DO', 'IN PROGRESS', and 'DONE'. The 'TO DO' column contains a message: 'You haven't started a sprint. You can't do anything on your board because you haven't started a sprint yet. Go to the backlog to plan and start a sprint.' with a 'Go to Backlog' button. The 'IN PROGRESS' and 'DONE' columns are empty. A large blue circular arrow icon is positioned above the message in the 'TO DO' column.





# Roadmap

Jira Software

Your work

Projects

Filters

Dashboards

People

Apps

Create

Search

foodie

Software project

PLANNING

Roadmap

Backlog

Board

DEVELOPMENT

Code

Project pages

Add shortcut

Project settings

You're in a team-managed project

Learn more

Projects / foodie

Roadmap

IK

Status category

Epic

FOOD-1 Admin Dashboard

FOOD-2 End User Web App

+ Create Epic

APR

MAY

Today

Weeks

Months

Quarters

FOOD-2

1

End User Web App

To Do

Description

User web app is our second project to proceed with. In this project, we will maintain and manage various modules eg showing dishes to the Users, Giving them options to manage the shopping carts and place the orders, etc.

Details

Assignee

Unassigned

Labels

None

IK

Add a comment...

Pro tip: press M to comment

©Simplilearn. All rights reserved.

simplilearn



# User Story

The screenshot shows the Jira Software interface with a 'Create issue' modal open. The modal is titled 'Create issue' and has a 'Import issues' button. The 'Project' dropdown is set to 'foodie (FOOD)'. The 'Issue type' dropdown is set to 'Story'. The 'Summary' field contains the text 'Authentication'. The 'Description' field contains the following text:

Normal text ▾ | **B** *I* ... | A ▾ | [List Icon] [List Icon] | [Link Icon] [Image Icon] [Email Icon] [Emoji Icon] [Table Icon] [Code Icon] [Info Icon] + ▾

- **Summary:** As an admin user, I should be able to access users' data.
- **Description:** As an admin user, I want to monitor newly registered users. Users can be filtered by the date of registration. Also user's activities such as frequently visited sections in the end-user app.
- **Acceptance Criteria:** In the user list page, by default page should be empty but with a prompt of user filtration criteria. In the prompt, the options of users retrieval on the basis of frequently active, last

At the bottom of the modal, there is a checkbox labeled 'Create another issue' and two buttons: 'Cancel' and 'Create'.

# Link Story to Epic

Jira Software

Your work

Projects

Filters

Dashboards

People

Apps

Create

Search

foodie

Software project

PLANNING

Roadmap

Backlog

Board

DEVELOPMENT

Code

Project pages

Add shortcut

Project settings

Project

Roadmap

Epics

Issues

...

feedback

Share

Export

Create issue

Import issues

...

Story point estimate

Measurement of complexity and/or size of a requirement.

Reporter

IK

ishant kumar

Attachment

Drop files to attach or browse

Linked Issues

blocks

FOOD-2

(Exact key)

Allows to flag issues with impediments.

☐ Create another issue

Cancel

Create

# Backlog

Projects / foodie

Backlog

PLANNING

Roadmap

Backlog

Board

DEVELOPMENT

Code

Project pages

Add shortcut

Project settings

FOOD Sprint 1

Add dates

(0 issues)

0 0 0

Start sprint

...

Plan your sprint

Drag issues from the **Backlog** section, or create new issues, to plan the work for this sprint. Select **Start sprint** when you're ready.

+ Create issue

Backlog

(2 issues)

0 0 0

Create sprint

FOOD-3 Authentication

TO DO

...





# Sprint Planning

Jira Software

Your work

Projects

Filters

Dashboards

People

Apps

Create

Search

?

⚙

foodie

Software project

PLANNING

Roadmap

Backlog

Board

DEVELOPMENT

Code

Project pages

Add shortcut

Project settings

You're in a team-managed project

Learn more

Projects / foodie

Backlog

IK

Epic

Type

FOOD Sprint 1

Add dates

(1 issue)

000

Start sprint

FOOD-5

Test Authentication Component of Angular Admin Dashboard

TO DO

FOOD-6

Test Add Restaurant in Admin Dashboard

TO DO

+ Create issue

Backlog

(2 issues)

000

Create sprint

FOOD-3

Authentication

TO DO

+ Create issue

Add epic

FOOD-6

1

Test Add Restaurant in Admin Dashboard

To Do

Description

Add a description...

Details

Assignee

Unassigned

Labels

None

Sprint

None

IK

Add a comment...

Pro tip: press M to comment

# Starting a Sprint

**Jira Software** Your work ▾ Projects ▾ Filters ▾ Dashboards ▾ People ▾ Apps ▾ **Create** Search

Projects / foodie

## Backlog

FOOD Sprint 1

- FOOD-5 Test
- FOOD-6 Test

+ Create issue

Backlog (1 issue)

- FOOD-3 Auth

+ Create issue

### Start Sprint

2 issues will be included in this sprint.

Sprint name \*  
FOOD Sprint 1

Duration \*  
2 weeks ▾

Start date \*  
5/17/2022 7:39 PM ✕

End date \*  
5/31/2022 7:39 PM

Sprint goal  
Plan the Project with Agile  
Sync Projects and Test with Protractor

**Start** Cancel

**Test Add Restaurant in Ac Dashboard**

To Do ▾

Description  
Add a description...

**Details**

Assignee	Unassigned
Labels	None
Sprint	None

TK Add a comment



## Key Takeaways

- Users can do the project planning through Agile.
- They can work with user stories following Agile.
- They can perform hands-on Agile software development.
- They can also create projects and plan sprints in Jira.



## Before the Next Class

Since you have successfully completed this session, in the next discussion, you should know how to:

- Use Git commands
- Work with Angular CLI
- Build Angular projects
- Execute HTML and CSS



## What Next?

Now, we have finished our Project Planning. In our next discussion, we will:

- See how to work with Git
- Sync the repository of Angular project for admin dashboard
- Sync the repository of Angular project for end user web app
- Work with database structure

