
Test Plan Document

for

Food Delivery Service Management System

Authors:

Anish Sofat (19CS10011)

Satvik Bansal (19CS10053)

Aryan Agarwal (19CS30009)

Instructors :

Prof. Sourangshu Bhattacharya

Prof. Abir Das

Uppada Vishnu

Indian Institute Of Technology Kharagpur

27 March 2021

Contents

Contents	1
Test Plan Outline	2
TEST PLAN IDENTIFIER FDSMS-Test-1.01	2
Introduction	2
Features to be tested	2
Sign-Up Process	2
Login Process	2
Customer	2
Delivery Agent	3
Restaurant	3
Admin	3
Test Plans	4
Test Plan for Home Page	4
Test Plan for Signup Process	4
Test Plan for Login Process	6
Test Plan for Customer Features	6
Test Plan for Restaurant Features	9
Test Plan for Delivery Agent Features	12
Test Plan for Admin Features	13
APPROACH	17
Methodology	17
Test Types	17
Meetings	18
Environmental Needs	18
Software Risk Issues	19

Test Plan Outline

TEST PLAN IDENTIFIER FDSMS-Test-1.01

Introduction

The objective of this test plan is to test the functionality of the Food Delivery Service Management System. This plan will address only those items and elements that are related to the delivery process, both directly and indirectly affected elements will be addressed. The primary focus of this plan is to ensure that the System application should run smoothly and provide appropriate information to the user without any loss of information and covering most of the data in-consistencies.

Features to be tested

1. Sign-Up Process

- a. Three options to sign up
- b. Redirection to the correct page
- c. Proper message display if the signup is successful or not.
- d. Redirection to Login Page if the signup is successful.

2. Login Process

- a. Entering of information in the fields
- b. Redirection to the correct dashboard after login
- c. Proper message if the login is unsuccessful.

3. Customer

- a. Should be able to see the personal details
- b. See the list of all the restaurants
- c. See the list of all recommended restaurants
- d. See the menu of a restaurant
- e. Create and order

- f. See the list and details of the current running orders
- g. See the list of all the past orders
- h. Give feedback to the restaurant and the delivery agent
- i. Check the details of an order

4. Delivery Agent

- a. Should be able to see their personal details
- b. Should be able to mark their current location
- c. Should be able to see the current available requests for delivery
- d. Accept and update the status for the order accepted
- e. Should be able to rate the customer

5. Restaurant

- a. Should be able to see their personal details
- b. Create the menu and delete unwanted items
- c. See the list of pending orders
- d. Update the status of the orders and accept/reject according to the availability.
- e. See the list of all the past orders
- f. See the list of all the nearby delivery agents
- g. Send a delivery request for the order.

6. Admin

- a. See the list of all the Customers, Restaurants, and Delivery Agents. The list also should display the ratings of all the users.
- b. Recommend Restaurants and Food Items
- c. Delete Users from the database.
- d. Give promotional offers to the customers
- e. Create new offers.

Test Plans

Test Plan for Home Page

1. Available Features

a. Buttons for different Functions

There should be three buttons that will redirect the user to the signup page, login page and the dashboard if the user is logged in already. The header symbol also redirects to the homepage from every page. So the redirection must be correct.

2. Activities

a. Redirection

The redirection from the given buttons must be correct without any server or debug error.

3. Spacing and Background handling

a. Background Image

The background image should not be scrollable and should resize with the window instead of getting cropped.

Test Plan for Signup Process

1. Redirection to the Right Pages

a. Buttons

There should be three buttons referring to three different type of users i.e. customers, restaurant and delivery agent. These buttons will redirect them to their respective pages.

b. Redirection

After the redirection the pages opened should confer with the user selected the pages have headings over the form for input.

2. Working input Fields

a. Responsiveness of Fields

The labels in the fields should retract when the field is active or filled. This is not true as autofill does not activate the javascript.

When the page is backed from the history the filled fields are there and the labels don't contract because of the same reason.

b. Required Input Fields

The required input fields will become red if they are clicked and then no data is filled in them. This will not allow the user to fill the form.

3. Error handling while signing up

a. Wrong or Repetitive Data

Proper message will be displayed if the sign up was unsuccessful because the email-id already existed. The data will be checked automatically.

If the image is not getting uploaded then an appropriate message is shown.

b. Redirection

If the signup is unsuccessful then the user will be redirected to the same signup page with an appropriate message stating the problem.

4. Uniqueness of Email-Id

If the Email-Id is not unique then the signup will not be successful. And as mentioned above will be redirected to the same signup page with an appropriate message.

5. Area Specific Sign-Up

If the user selects the area that is not in the list i.e. "Other", then the user will not be signed up. And the message stating that we don't deliver in this area is shown.

6. Redirection to Login Page after Signup

If the signup is successful then the user is redirected to the login page stating the message of successful signup.

Test Plan for Login Process

1. Appropriate message when wrong credentials are given

If wrong credentials are filled then a message above the input fields will state that the message telling the user to enter correct credentials.

The input fields are required so the query for login will not be created until some data is filled in those fields.

2. Redirection to correct dashboard based on the login credentials

After the login is successful the system checks the user profile and redirects the user to the correct dashboard. Each Dashboard has a Welcome message stating the name of the user and the user type. This way the user can confirm that the dashboard is correct.

3. Logout process after should remove the session details of the user and should not redirect to the dashboard again without the credentials

As the feature states if the user presses the logout button, then the user is redirected to the login button and removing any user information stored from the session and if the user tries to press the back button or tries to enter the dashboard again, then user will be redirected back to the login page.

4. The pages are protected from unverified users

If the user is logged in, then the user should not be allowed to visit pages that the user should not be visiting, for example a restaurant user should not be able to see the Customers List. This can be checked by changing the URL. If the user tries to do so, and has no permission to visit that page, the user will be redirected to the login page logging out the user from the session.

Test Plan for Customer Features

1. Personal Details

a. Button Redirection

When the button is pressed, then the user must be redirected to a page that will show the user the personal details that were added by them during signup.

b. Details Display

The details will be shown as a list, with only essential details viewable.

2. Buttons on the dashboard should redirect to the right page and provide the right functionality expected.

As mentioned, the dashboard has many buttons corresponding to different pages and functionalities. The redirections must be correct, and must serve the purpose as mentioned by the label on the button.

3. Dashboard redirection should work fine

If the user presses on the Dashboard button in the top bar, then the user should be redirected to the dashboard, showing those buttons.

4. Order Now

a. Button Action

When the button is pressed then a list of restaurants should be shown, displaying some of the essentials about the restaurant and a button to see the menu and order from that restaurant.

b. Should be able to see the complete list

When the button in the restaurant list is clicked, the complete menu of the restaurant should be visible with a selector to select the quantity of the item. The name and price will be visible in the menu.

c. Should be able to create the order from the list of the menu of a certain restaurant.

The selector in the menu is for the quantity of that item. This will create a list of all the items that are to be there

in the order. When the order button is clicked, the customer will be redirected to a new page showing the details of the order and the option to apply promotional offers. This will also show the cost of the complete order.

d. Apply Promotional Offer on the menu

This option is there when viewing the order by the customer. The customer will be able to see the offers there in the list given by the admin. The user will be allowed to use only one offer at time. The effect will be shown in the cost.

When an offer is used, after the order is placed the offer is removed from the list. This does not take into consideration if the order is accepted by the restaurant or not.

e. Correct Amount should be shown

The amount shown on the order, the sum must be correct and the offer amount must be incorporated correctly. The display of the order details must be proper.

5. Recommended Restaurants List

The list and features are similar to that of the All Restaurant List but this list will only consist of only restaurants that are recommended by the admin manually.

6. Present Orders

a. List of orders

When the button will be clicked the list of all the current running orders will show up and pressing on the details, the user will be redirected to new page that will show the details of that order

b. Detail of order

The details of the order must consist of its items, restaurant name and the timeline of the order. The timeline should tell the status if the order is accepted/rejected and approximate time of different steps.

c. Status of Order

The status of the order or the order timeline will consist of approximate times of updates and should give information about the food preparation, out for delivery and delivered. This information is updated by the restaurant and delivery agent.

d. Feedback

There will be two buttons on the order page to give rating and feedback to the restaurant and delivery agent. Then the user will be redirected to pages to take input the ratings for the restaurant and delivery agent respectively. The user should be able to give the ratings only once.

7. Past Order List

a. Details of the orders

This list will show all the orders that have been placed by the customer. The list will only show the name of the Restaurant, the Delivery Agent name, the date of placing, the complete bill and a button to see the complete details of the order.

b. Elements in the list

The list must not consist of the present orders that are currently running for the user. If an order is delivered or rejected, then the order must be added to the list.

Test Plan for Restaurant Features

1. Personal Details

a. Button Redirection

When the button is pressed, then the restaurant must be redirected to a page that will show the restaurant personal details that were added by them during signup.

b. Details Display

All the details of the restaurant should be fetched from the database and shown in as a list, with only essential details viewable on the page.

2. Buttons on the dashboard should redirect to the right page and provide the right functionality expected. As mentioned, the dashboard has many buttons corresponding to different pages and functionalities. The redirections must be correct, and must serve the purpose as mentioned by the label on the button.

3. Dashboard

a. Login

After login the restaurant should be directed to the dashboard page where buttons should be working properly as per the functionality given on their label.

b. Button in the header

If the restaurant presses on the Dashboard button in the top bar, then the user should be redirected to the dashboard, showing some buttons giving the mentioned functionality.

c. Redirection

Buttons on the dashboard should redirect to the right page and provide the right functionality expected. As mentioned, the dashboard has many buttons corresponding to different pages and functionalities. The redirections must be correct, and must serve the purpose as mentioned by the label on the button.

4. Create Menu

a. Add Food Item

Button is provided to the restaurant so that it can add new food items to the menu. Button should lead to a page that consists of input fields like name of the item, price and image to be uploaded. Input field data should be correctly inserted into the database.

b. Displays Menu

Page should consist of the menu displayed just below the add Food Item button in a tabular form. It should be vertically scalable as new food items are added. Input field data of food items should be fetched from the database using the restaurant id.

c. Deleting Food Item

When pressed, the button should be able to delete the food item from the database and keep the menu updated during the process.

5. Pending Orders List

a. Accept/Reject List

Orders given by the customers to the restaurant should be shown on this page in tabular form. When the accept button is pressed it should successfully accept the pending order and when the remove button is pressed it should successfully remove the order from the list.

b. Estimated time

Using this input field the restaurant should be able to give an estimated time for preparation which should be visible to the customer.

c. Generate Delivery request

When the generate delivery request button is pressed the accepted order should become visible for delivery agents to accept.

d. Mark Out for Delivery

When the “Out for Delivery” button is pressed by the restaurant it should mark the order to be Out for Delivery which should be visible to the customer.

6. Past Orders List

Past orders served by the restaurant to the customers should be shown on this page in tabular form.

Test Plan for Delivery Agent Features

1. Personal Details

a. Button Redirection

When the button is pressed, then the delivery agent must be redirected to a page that will show the delivery agent personal details that were added by them during signup.

b. Details Display

All the details of the delivery agent should be fetched from the database and shown in as a list, with only essential details viewable on the page.

2. Mark Location

a. Current areas List

Delivery agent will get a list of areas from which we can choose or mark his own location.

b. Updation in Database

When this button is pressed the location marked in the list should be pushed in the database to update the current location of the delivery agent.

3. Delivery Request

a. See Delivery Request

The delivery agent should be able to see the orders in his area which haven't been accepted yet. Orders should be shown in a tabular form on the webpage.

b. Accept Request

When this button is pressed, order should be marked accepted and removed from the delivery request page. It should be allotted to the delivery agent who pressed the button.

4. See the current Order and Details

a. Display

It should contain different buttons labelled as per the functionality labelled below.

b. Estimated time for pickup

Using this input field the delivery agent should be able to give an estimated time for pickup which should be visible to the customer.

c. Estimated time for delivery

Using this input field the delivery agent should be able to give an estimated time for delivery which should be visible to the customer.

d. Delivered

When the delivery agent presses this button the status of the order should be updated to delivered and the order should no longer be visible on the delivery agents screen.

e. Rate customer

Using this input field the delivery agent should be able to rate the customer and on submission the rating data of the customer should be fetched from the database and should be updated correctly in the database.

Test Plan for Admin Features

1. List of Customers

When the button on the dashboard is clicked then the user will be redirected to a page showing the list of all the customers in the database.

a. Table of Details

The list will show some of the basic details of the customer with two buttons to give an offer, and delete the user. The details provided must confer with the details provided by the customer during their signup. The list will also show the rating of the customer.

b. Give an Offer

If this button is pressed then the user will be redirected to a list showing the list of all offers present and the user can select one of the offers to give to the customer. Pressing the Give Button the following offer will be added in the list of offers of that customer.

c. Delete User

When this button is pressed the user should be deleted from the list and data must be removed from the database including the image as well. The page will refresh showing the updated list. The user deleted should be the one whose button was clicked.

2. List of Restaurants

When the button on the dashboard is clicked then the user will be redirected to a page showing the list of all the restaurants in the database.

a. Table of Details

The list will show some of the basic details of the restaurant with three buttons to recommend, open menu, and delete the user. The details provided must confer with the details provided by the restaurant during their signup. The list will also show the rating of the restaurant.

b. Open Menu

This button will open the menu of the restaurant, and show all the items that were entered by the restaurant, in a tabular form. The table is similar to what was shown to the customer. This table will also consist of a Recommend button.

c. Recommend

The list consists of a button referred to as change. This button should change the recommended state of the restaurant in the list and in the database. If the restaurant is recommended then it should be removed from the recommended and vice-versa.

d. Delete User

When this button is pressed the user should be deleted from the list and data must be removed from the database including the image as well. The page will refresh showing the updated list. The user deleted should be the one whose button was clicked.

3. List of Delivery Agents

When the button on the dashboard is clicked then the user will be redirected to a page showing the list of all the restaurants in the database.

a. Table of Details

The list will show some of the basic details of the delivery agents with one button to delete the user. The details provided must confer with the details provided by the delivery agent during their signup. This list will also show the rating of the delivery agent.

b. Delete User

When this button is pressed the user should be deleted from the list and data must be removed from the database including the image as well. The page will

refresh showing the updated list. The user deleted should be the one whose button was clicked.

4. Change of Recommendation of Restaurants and Food Items

These buttons are provided in their respective fields in the tables where their details were shown.

When this button is clicked the state of the respective thing changes. For e.g. if the restaurant is recommended and then the button is clicked it will be removed from the recommended list.

5. Create Offer

This is the button on the dashboard when clicked should redirect to a new page where the admin can see previous created offers or create new one.

a. Offer List

This list will show the basic details of the offer, the name and the value of the offer.

b. New Offer

When this button is clicked the user is redirected to a new page that will have input fields to take in new offers and add them to the list of the offers.

These offers can be later given to the customers.

6. Give Offers to Customers

The offers created are to be given to the customers through the all customers list. When the button is clicked for a given offer, a list will be shown where all the offers created will be visible.

a. Giving an offer

When the button in front of an offer is pressed the respective offer should be added in the list of that customer and user is to be redirected to the all customer list.

APPROACH

1. Methodology

We aim to achieve Agility using Scrum Development Methodology in this project. We approach the project with continual planning, learning, improvement, team collaboration, evolutionary development, and timely completion. Encouraging flexible responses to change. We emphasize on individual and team interactions over processes and tools.

2. Test Types

Software testing will be performed by developers instead of a third party due to the small nature of the project. Exhaustive testing is not possible. As we need the optimal amount of testing based on the risk assessment of the application.

a. Unit Testing

We will focus on smaller units of the software design. In this, we will test an individual unit or group of interrelated units. It is often done by the programmer by using sample input and observing its corresponding outputs.

b. System Testing

Software will be tested such that it works fine on different operating systems primarily Linux, Windows and macOS. We will follow the black box testing technique. In this, we will be focusing on the required input and output without focusing on internal working with special emphasis on page protection while doing security testing.

c. Regression Testing

Adding new files leads to changes in the program. We make sure that the whole component works properly

even after adding new components to the complete program.

3. Meetings

Test meeting will be done at the end of the day to check for errors within the code written during that day. Additional meetings can be done as required for emergency situations.

Environmental Needs

1. There are no special hardware requirements for running the app.
2. Test data will be provided to the software by performing demo orders, demo user creation etc.
3. There are no specific requirements apart from a working email id and a working internet connection, while testing the software.
4. There are no specific data ranges apart from keeping the size of images small (so image uploading and retrieval will be done faster). And also while testing use of unique data is preferred because repetition of data in user creation will give an error.
5. Each part of a multipart feature should be tested at least once by creating dummy users, dummy orders etc.
6. If you want to install a server locally, a working computer's power will be required. For the client, power will be required as per the device he/she is using. No specific power is needed from the client side. The server will be hosted on platforms like Heroku and they will manage the power requirements.
7. Supporting softwares:
 - **firebase_admin**
 - **pyrebase**
 - **requests**
 - **json**
 - **datetime**

- **name** from os
 - **credentials, auth, firestore, storage** from firebase_admin
 - **Flask, render_template, url_for, request, redirect, session** from flask
 - **wraps** from functools
 - **HTTPError** from requests.exceptions
8. There is no restricted use of the system during testing. This software is a web app, every session of the user will be separate, and all will be handled simultaneously by the server via Firebase. So other users will not be affected when testing the software.

Software Risk Issues

All pip python-pip packages should be in the versions as mentioned in the requirements.txt file. If that didn't work , you can troubleshoot by updating the pip packages to their latest version.