# FoodAdda

# Online Food Delivery

Product Requirement Document

Online Food Delivery

*“Customers who love online Food”*



Jitendra Chauhan

SDE @ Infocache System

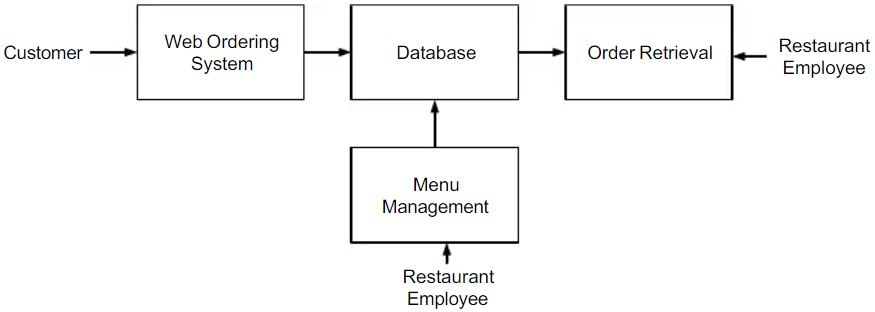
Jeetendra.chauhan@gmail.com

Noida(U.P.), India

# Intro & Goals:-

# What is Online Food Delivery

Online Food Delivery is a website designed primarily for use in the food delivery industry. This system will allow hotels and restaurants to increase scope of business by reducing the labor cost involved. The system also allows to quickly and easily managing an online menu which customers can browse and use to place orders with just few clicks. Restaurant employees then use these orders through an easy to navigate graphical interface for efficient processing.



# Goals

* Our goal is to create an Online Food Delivery application which can be like Swiggy, Zomato, uber eats or any other website or product that comes to your mind when you hear about Online Food Delivery.
* The end goal of this project is directly related to the placements calls which mentees would receive from partners of pesto.
* Enhance tech skills on the next level using his project, by using this app Can Broaden Your Brand, It’s More Convenient to Increase Your Reach. Marketing Opportunities, Scalable.
* We should aim for building a full stack Online Food Delivery application in which the user can select a Dish in any nearby restaurant, add it to his cart and then finally using available payment modes place orders through it.

# 

# User Types

1. **Customers(Registered Users)** –

The system allows to quickly and easily managing an online menu which customers can browse and use to place orders with just few clicks registered users can share review experience.

* Create an account.
* Manage their account.
* Log in to the system.
* Navigate the restaurant’s menu.
* Select an item from the menu.
* Add an item to their current order.
* Review their current order.
* Provide payment details.
* Place an order.
* Receive confirmation in the form of an order number.
* View order placed.

**2. Restaurant owner** –

They can register their restaurant and Add /update/ delete food item to/from the menu. Restaurant employees then use these orders through an easy to navigate graphical interface for efficient processing.

* + Register restaurant
  + Add /update/delete food item to/from the menu.
  + Update price for a given food item.
  + Update additional information (description, photo, etc.) for a given food item.

# Functional Requirement

* Signup/Login.
* Landing page where users can explore offers, nearby Restaurants, and food category sections with a nice hero banner.
* Listing page where users can apply their choices of search and filters and explore dishes.
* Users can click and explore the product detail page.
* Users can add dishes to their cart list to buy.
* Users can increase or decrease cart items and see live total amount calculation.
* Users can save for later (wish list) or remove any products in the cart.
* One click order from Food List page
* Users can make payments with a secure payment system from the checkout page.
* Users can select an address from the address dropdown and add a new address also.
* Users can manage account details.
* Users can manage wish list food dishes.
* After a successful order, User will get an email from our dishes about their successful order and payment confirmation.

**Non-Functional Requirement**

* Fast loading of landing page(Lazy loading implementation of images)
* Single click to place order(User will decide for this process)
* Secure transactions
* Ensure maintainable and scalable system
* Follow the standard coding practices.

**Future Work & Future Scope:**

The following section describes the work that will be implemented with future releases of the software.

* Customize orders: Allow customers to customize food orders
* Enhance User Interface by adding more user interactive features.
* Provide Deals and promotional Offer details to home page.
* Provide Recipes of the Week/Day to Home Page
* Payment Options: Add different payment options such as PayPal, Cash, Gift Cards etc. Allow to save payment details for future use.
* Allow to process an order as a Guest
* Delivery Options: Add delivery option
* Order Process Estimate: Provide customer a visual graphical order status bar
* Order Status: Show only Active orders to Restaurant Employees.
* Order Ready notification: Send an Order Ready notification to the customer

## **Component**

## Login

* Signup

## Search

## Menu

* Header
* footer

## Carousel

## Cart

## Dish Listing

## User profile

## Order History

# Additional Requirements

Cover all the bonus requirements once you are done with the basic requirements and have time left to focus on additional features.

## Payment

Integrate all payment gateway, Users should be able to set default payment method.

## Test cases

Write Test cases for your components in jest & react testing library on Front-end and mocha or chai on back-end.

## Test Suites

If you have sufficient time then also add integration tests suites on front-end in cypress or on back-end in selenium or other specific tools.

## Multiple Language Support

Add multiple language support for your site. Users should be able to change site language from header. all text content should be in the selected language. keep English as the default language.

## Google Analytics

Add Google analytics based event tracking on front-end

# High Level Overview of Application

# User Stories

1. User Authentication -
   1. Signin/Signup
2. Landing Page
   1. Hero banner
   2. Offer Banners
   3. Search dish
   4. Search restaurant
3. User Detail page
   1. User info(read and write functionality)
   2. View address list and add address
   3. Order History
4. Header
   1. Logo
   2. Search bar
   3. Login/Signup
   4. User profile
5. Footer
   1. Copyright
   2. Quick links
   3. Contact us - Social Media
   4. Newsletter
   5. Logo
6. Restaurant page

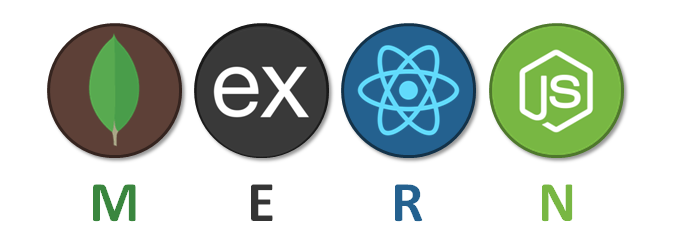
* 1. Menu
  2. Dishes images - zoom on hover
  3. Restaurant Ratings and Reviews
  4. Restaurant details

1. Cart page
   1. List of added items
   2. Edit / Deleted items from cart
   3. Payment details with all component like delivery charges and GST
   4. place order, save for letter and remove item from cart
   5. Check for offer code functionality
2. Payment details page
   1. Select payment methods
   2. Show saved card or previous payment method
   3. COD option
3. Order status page
   1. Order placed successfully or not
   2. Order details info

# Technology Stack

1. **Frontend Stacks:** ReactJS, Redux, SCSS, HTML, Material ui
2. **Backend Stacks:** React & Redux
3. **Database**: MongoDB
4. **Tools:** VS Code, Figma(Visual Design)

# Deployment

Frontend - Netlifly

Backend - Heroku

DataBase - Mongo Atlas

CI/CD - GithubAction

# Links

Frontend - <https://foodadda-pesto.netlify.app>

Backend - <https://foodadda-pesto.herokuapp.com>

# Repository

Frontend - https://github.com/pesto-students/foodada-fe-n11-iota1

Backend -https://github.com/pesto-students/foodada-be-n11-iota1

# Continuous Discovery & Release Plan

* List down all the EPICs (Backlog)
* Make sure to create MVP products that go live initially which includes landing page, product page product detail page and payment gateway.
* Incremental release planning per sprint
* Roll-out matrices based on selected service areas
* FAQ and suggestion



**Thanks**