**Online Food Ordering**

**Software Requirements Specifications**

# **Table of Content:**

1. Introduction
   1. Purpose of the Document
   2. Scope of the Food Ordering System
2. Model
3. Functional & Non-Functional Requirements
   1. Functional Requirements
   2. Non-Functional Requirements
4. UML Diagrams
   1. Use Case Diagrams
   2. Class Diagrams
   3. Sequence Diagrams

# **Introduction:**

## **Purpose of the Document**

The purpose of this document is to provide a comprehensive and detailed overview of the requirements and specifications for the development of the Food Ordering System. The document aims to serve as a reference guide for stakeholders involved in the development, testing, and maintenance of the system. By clearly defining the scope, objectives, and constraints of the project, this document will facilitate a shared understanding among all project participants.

## **Intended Audience**

The primary audience for the Food Ordering System is your valued customers. These individuals will interact directly with the online store to browse restaurants, place orders, and track deliveries. The system is designed to enhance their overall experience, providing a convenient and efficient platform for ordering food online.

Understanding the preferences and expectations of your customers is paramount in delivering a user-friendly and satisfying online shopping experience. The Food Ordering System is tailored to meet the needs of this audience, ensuring a seamless and enjoyable process from selection to delivery.

## **Scope of the Food Ordering System**

The Food Ordering System is designed to streamline and enhance the process of ordering food from various restaurants through an online platform. The system fulfils to both customers seeking a convenient and efficient way to browse, order, and track food deliveries and restaurant staff managing their menus and processing incoming orders. The scope includes the development of user interfaces for customers, restaurant staff, and administrators, as well as the backend processes that enable seamless order processing and delivery tracking.

# **Functional & Non-Functional Requirements:**

## **Functional Requirements:**

1. User Authentication and Authorization
   1. Users should be able to register and log in.
   2. Differentiate between customer, restaurant staff, and admin roles.
   3. Implement password recovery mechanisms.
2. Menu Management:
   1. Restaurants should be able to perform CRUD (Create, Read, Update, Delete) operations on their menu items.
   2. Categorize menu items (appetizers, main courses, desserts, etc.).
   3. Include details like item name, description, price, and availability status.
3. Ordering System:
   1. Customers should be able to add items to their cart.
   2. Enable customers to customize their orders (e.g., toppings, special instructions).
   3. Support order review and confirmation.
4. User Profiles:
   1. Users should have profiles with personal information and order history.
   2. Provide an option for customers to save multiple delivery addresses.
5. Shopping Cart and Checkout:
   1. Maintain a shopping cart for each user session.
   2. Allow users to review their orders before confirming.
   3. Integrate a secure and user-friendly checkout process.
6. Order Tracking:
   1. Implement real-time order tracking for customers.
   2. Provide status updates (order received, preparation, out for delivery).
7. **Restaurant Management:**
   1. Allow restaurants to set their availability status (open/closed).
   2. Provide a dashboard for restaurants to view and manage orders.
8. **Admin Panel:**
   1. Admins should have access to manage users, restaurants, and overall system settings.
   2. Include reporting and analytics features for order trends and user behavior.
9. **Search and Filters:**
   1. Implement a robust search functionality for users to find restaurants and menu items.
   2. Include filters based on cuisine, price range, ratings, etc.
10. Ratings and Reviews:
    1. Allow customers to rate and review restaurants.
    2. Display average ratings for restaurants and menu items.

## **Non-Functional Requirements:**

1. Performance:
   1. Ensure fast loading times, even during peak hours.
   2. Handle a large number of simultaneous users and orders.
2. Security:
   1. Implement secure user authentication and data encryption.
   2. Protect against common web application vulnerabilities (e.g., SQL injection, cross-site scripting).
3. Scalability:
   1. Design the system to scale horizontally to accommodate increased user traffic.
4. Reliability:
   1. Ensure high availability and reliability of the system.
   2. Implement regular backups to prevent data loss.
5. Usability:
   1. Design an intuitive and user-friendly interface.
   2. Provide clear navigation and instructions for users.
6. Compatibility:
   1. Ensure compatibility with different web browsers and devices.
7. Regulatory Compliance:
   1. Comply with data protection and privacy regulations (e.g., GDPR).
8. Integration:
   1. Integrate with payment gateways for secure transactions.
   2. Integrate with map services for address verification and delivery tracking.
9. System Maintenance:
   1. Include features for system updates and maintenance without disrupting service.
10. Documentation:
    1. Provide comprehensive documentation for users, administrators, and developers.

# **DFD for Online Food Ordering System: ­­**

A food ordering system is essentially a software that enables restaurant managers to oversee and take orders that are placed in-person or over the Internet. Let's use a data flow diagram, or DFD, to better understand how the food ordering system operates. Below is the DFD for the food ordering system**.**

Various DFD levels, including Level 0 DFD, Level 1 DFD, Level 2 DFD, and Level 3 DFD, are displayed here for the Food Ordering System.

## **Level 0 DFD –**

At this level, the Input and Output of the system are shown.

* + Food Ordering System has the following input:
    - Food order is input as the customer’s order for food.
  + Food Ordering System has the following output:
    - Receipt of the order.
    - For further processing the order, the food order is passed to the kitchen.
    - The restaurant manager gets the report of Bill and Management.

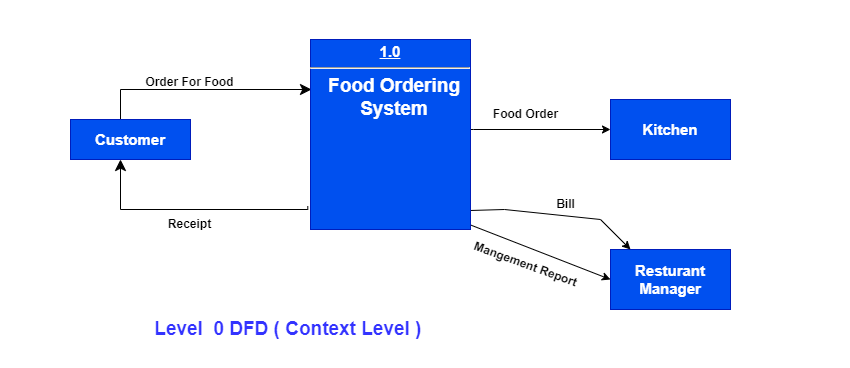
****

Figure 1 Level 0 DFD

## **Level 1 DFD –**

Process 1.0 is in charge of processing the order. The associated housekeeping tasks for food are symbolized by processes 2.0, 3.0, and 4.0. The inventory data (which describes the records of datasets such as their name, their content, their source, many useful information, etc.) should be maintained at the same time as the detailed information about daily sold items is made available to create and report management.

Therefore, the DFD level shown below uses the following two data stores:

* Database of goods sold
* Database for inventory

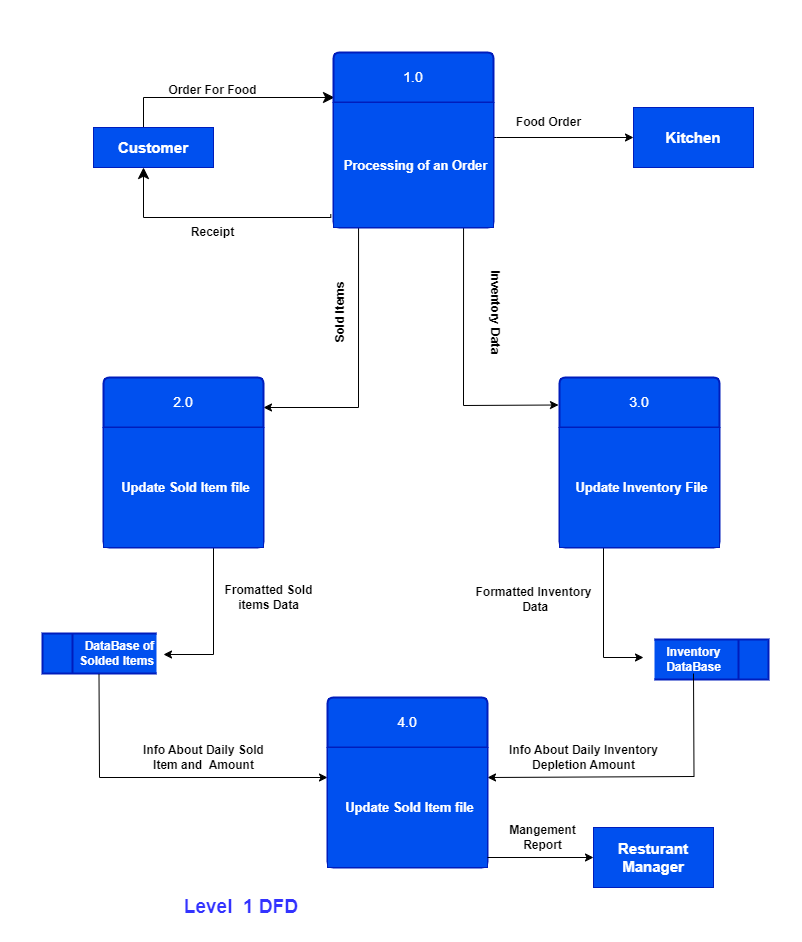


Figure 2 Level 1 DFD

## **Level 2 DFD –**

Below is detailed information regarding "Processing of an Order":

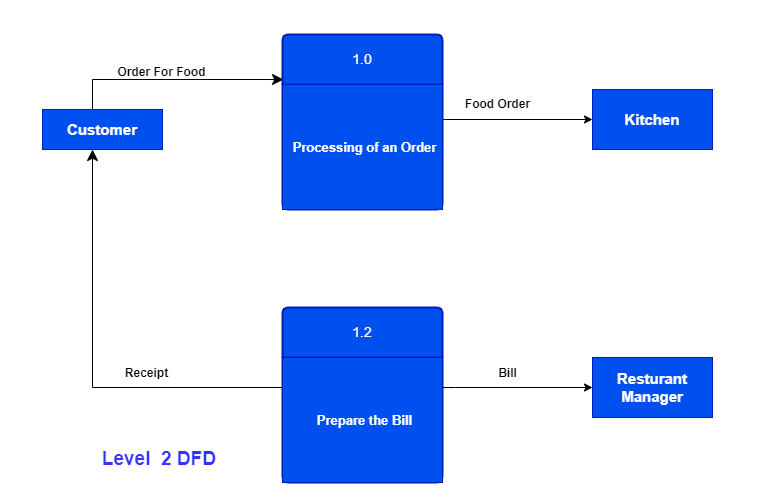


Figure 3 Level 2 DFD

## **Level 3 DFD –**

At this DFD level, more specific details regarding the "Generate Management Report" activity are provided. To generate a management report, it is necessary to have access to inventory and sold item data. The management report that is generated from the aforementioned computations should then be given to the restaurant manager once the data for both solid goods and inventory have been combined.

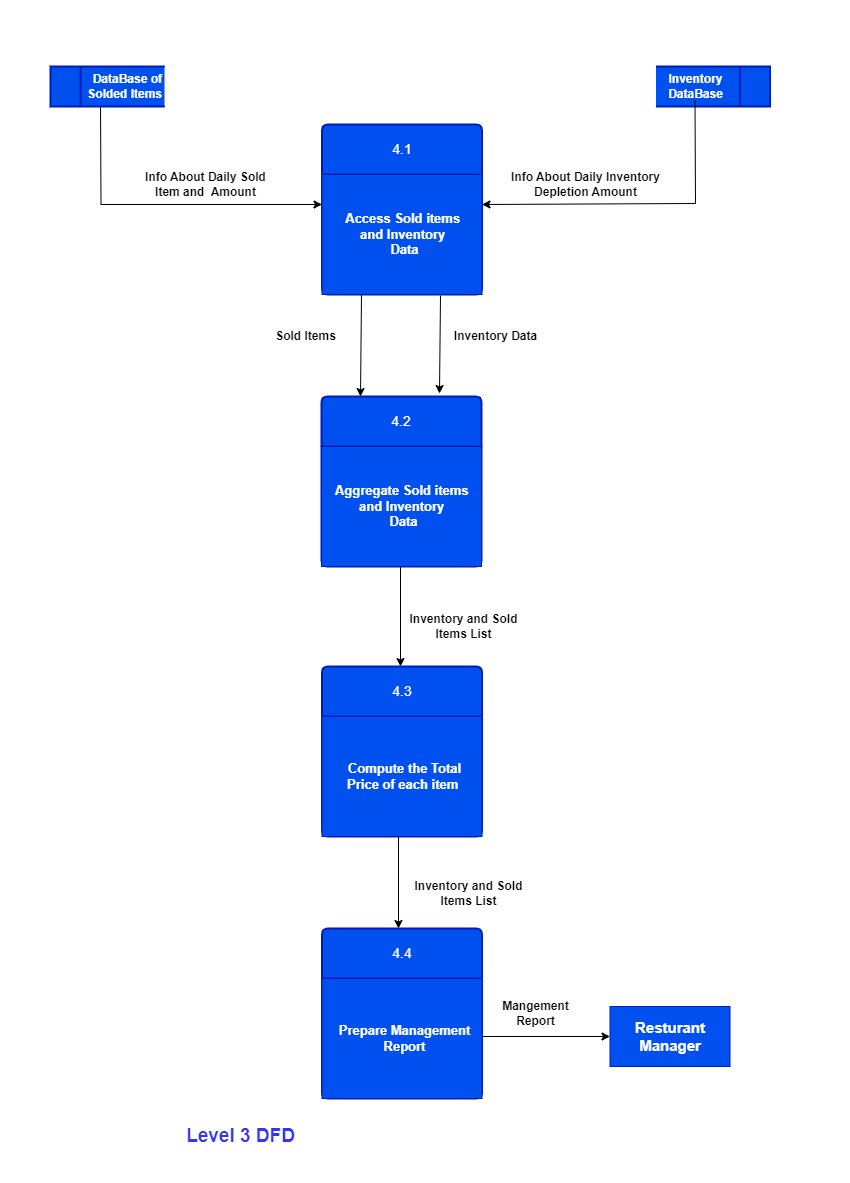
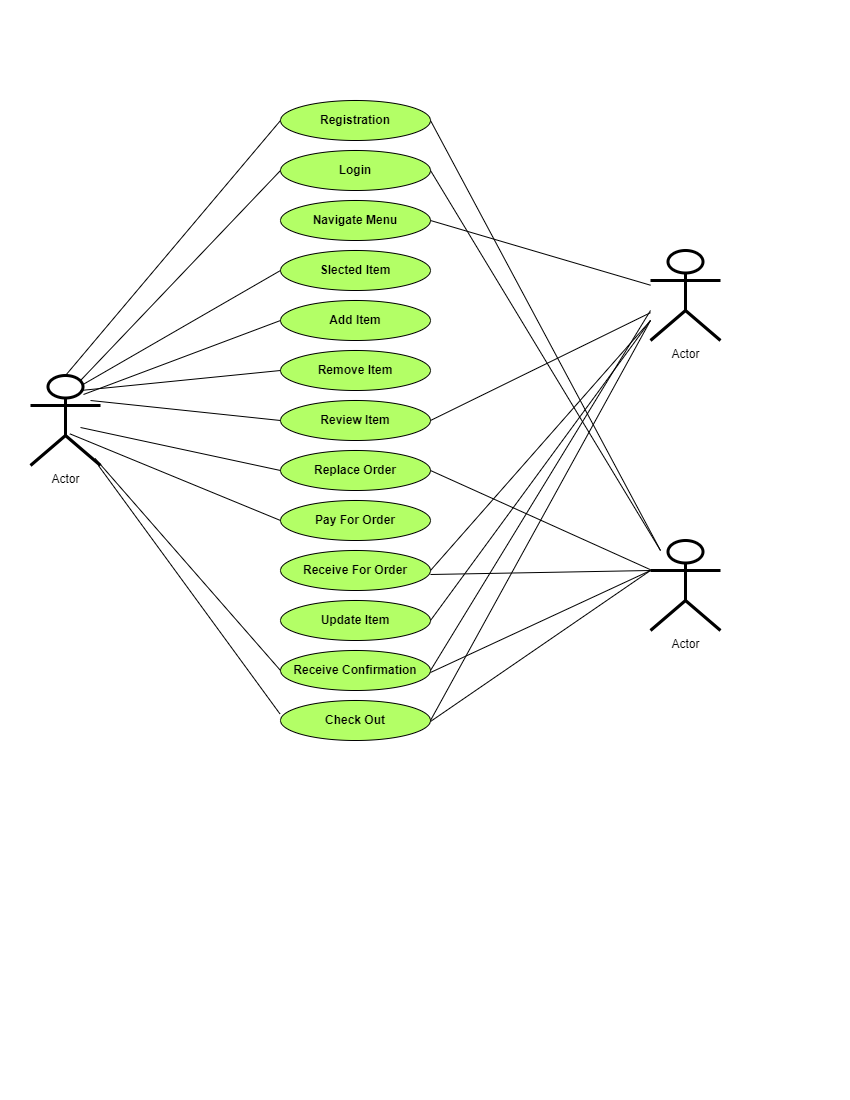


Figure 4 Level 3 DFD

# **Use Case Diagram of Online Food Ordering System**

## **First:**



**Figure 5: Use Case Diagram**

## **Second:**

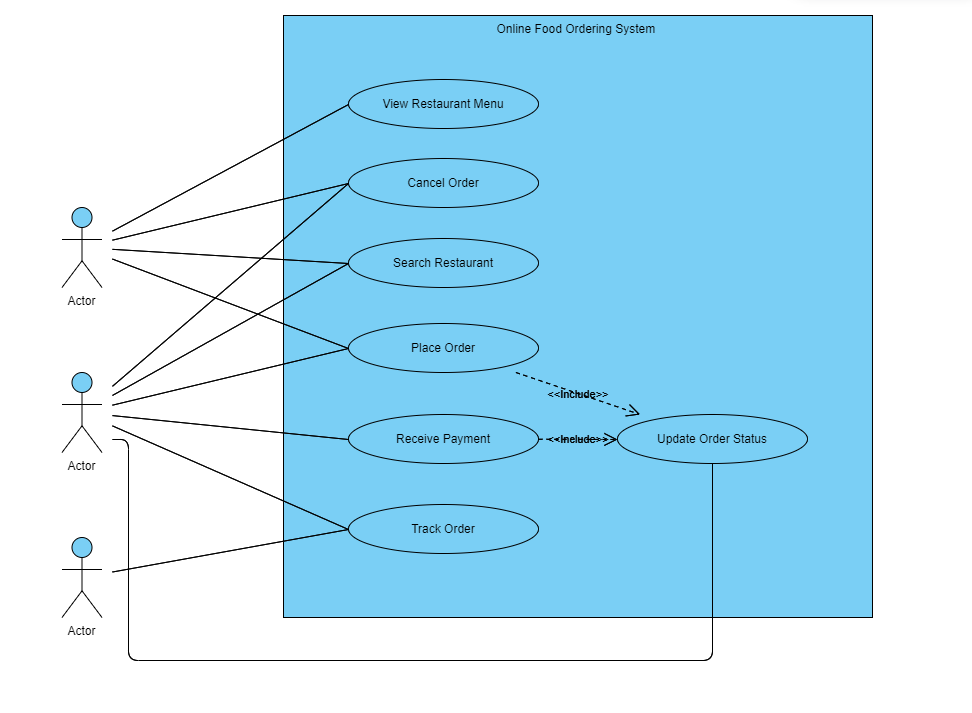


Figure 7: Use Case Diagram