# **CHAPTER FOUR**

## **4.0 SYSTEM ANALYSIS AND REQUIREMENT MODELING**

### **4.1 Methodology**

**Design and Methodology**

The approach or methodology to implement certain projects are numerous. Thus, some of these methodologies include:

**Waterfall Methodology**

The ancestor of all life cycle models. Each successive module in the system depends on the previous module being completed.

Stages of the project flows from the top to the bottom without going back to a finished stage. It involves sequence completing one task before moving on to the next, all the way to project completion. The goals and timelines are clearly defined for project delivery.

It follows a sequence of steps:

●Planning

●Analysis

●Design

●Development or Implementation

●Testing

●Deployment & Maintenance

**Merits of Waterfall Approach**

•Requirements are identified long before programming begins

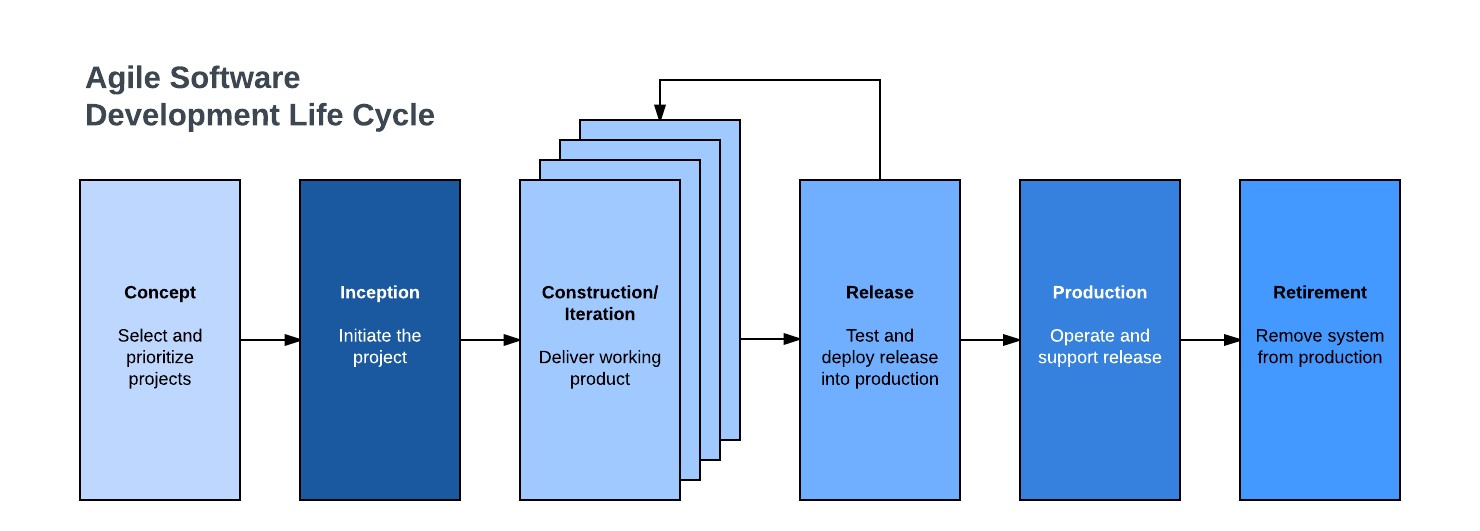
•Changes to the requirements as the project proceeds are limited.

**Demerits of Waterfall Approach**

* The design must be completely specified before programming begins.
* Time between system proposal completion (analysis phase), and system delivery is too long.
* Testing is treated almost as an afterthought in the implementation phase.

**Agile-Scrum Software Model**

Agile software methodology is an iterative approach that builds software incrementally from the start of the project rather than delivering it once. Agile models allow the use of increments or possible prototypes that can evolve into a more suited and validated requirements and eventually software application.



**SCRUM Process Model**

It is an agile process model which follows these activities: requirements, analysis, design, evaluation and delivery. Scrum emphasizes the use of a set of software process patterns that have been proven effective for projects with tight timelines, changing requirements and business criticality.

Why use agile for our project?

We will use Agile-Scrum as our methodological approach because it is:

* It is designed to curate the needs of rapidly changing environment by embracing the idea of incremental development and developing the actual final product.
* It requires constant comments from the user thus gives priority to the people than the process.
* Deployment and delivery are quicker thus gaining user confidence.

**Steps followed**

1. Product backlog creation: The team produced a product backlog, which is a prioritized list of user stories, tasks, and features that needed to be completed, after identifying the product features, needs, and functions that were required to construct the Customizable Ordering System.
2. The development team chose a number of items from the Product Backlog to work on during the next sprint. The team then developed a Sprint Backlog, which is a list of tasks and activities needed to fulfill the chosen items during the sprint.
3. Working on the Sprint: Using Agile Scrum techniques, such as daily stand-up meetings where team members reported on their progress and noted any obstacles that needed to be overcome, the development team worked on the items listed in the Sprint Backlog.
4. Testing and Product Demonstration: The development team tested the product's features and functions once the sprint was complete to make sure they were operating as planned.
5. Retrospective and Next Sprint Planning: To assess the sprint and pinpoint opportunities for development, the team convened a retrospective meeting. This involved examining what went well and poorly throughout the sprint and determining any necessary adjustments to the development process. The procedure from step two was then repeated after choosing a fresh set of items from the Product Backlog for the subsequent sprint.

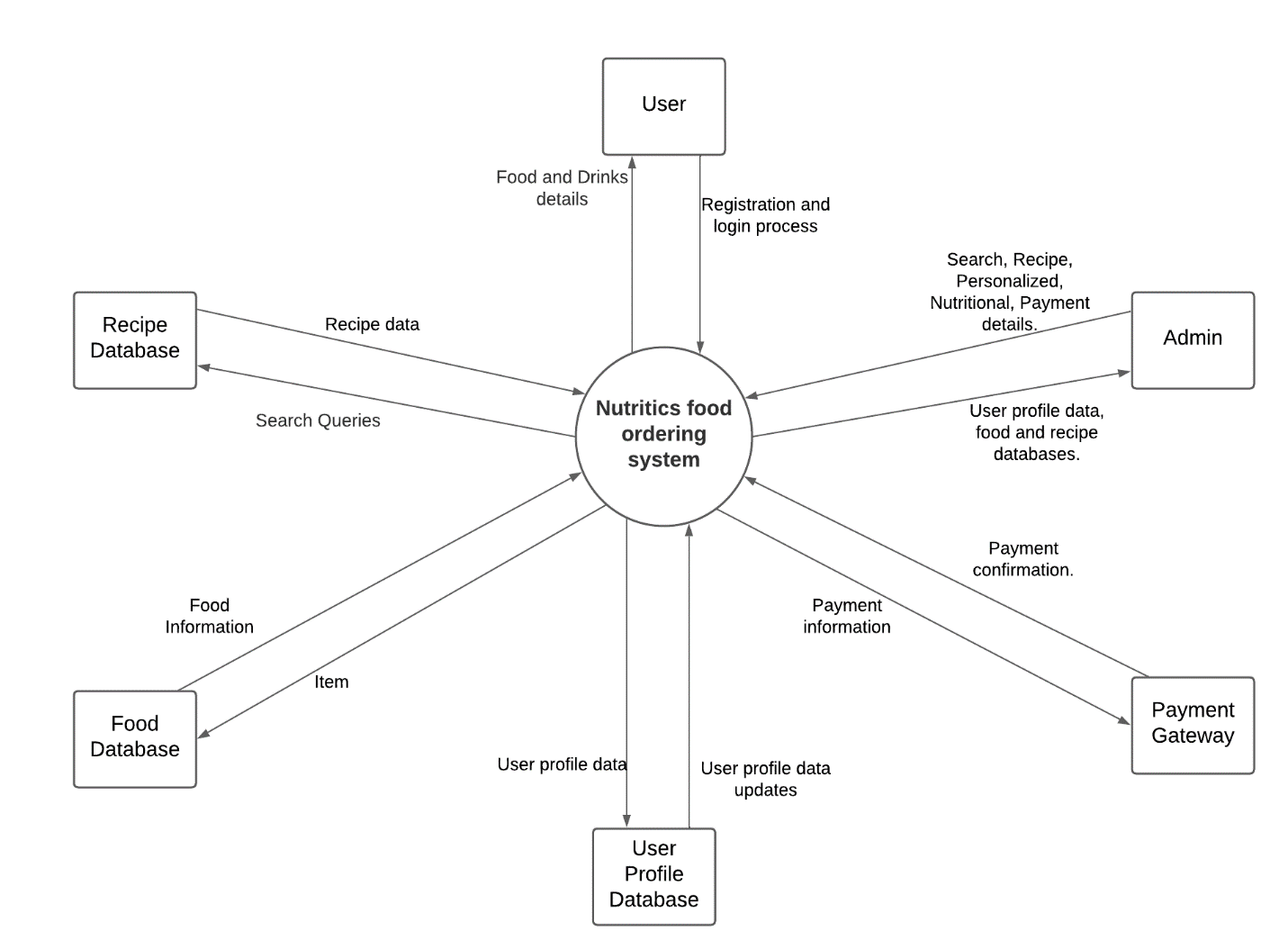
### **4.2 Existing system**

**How the existing system works.**

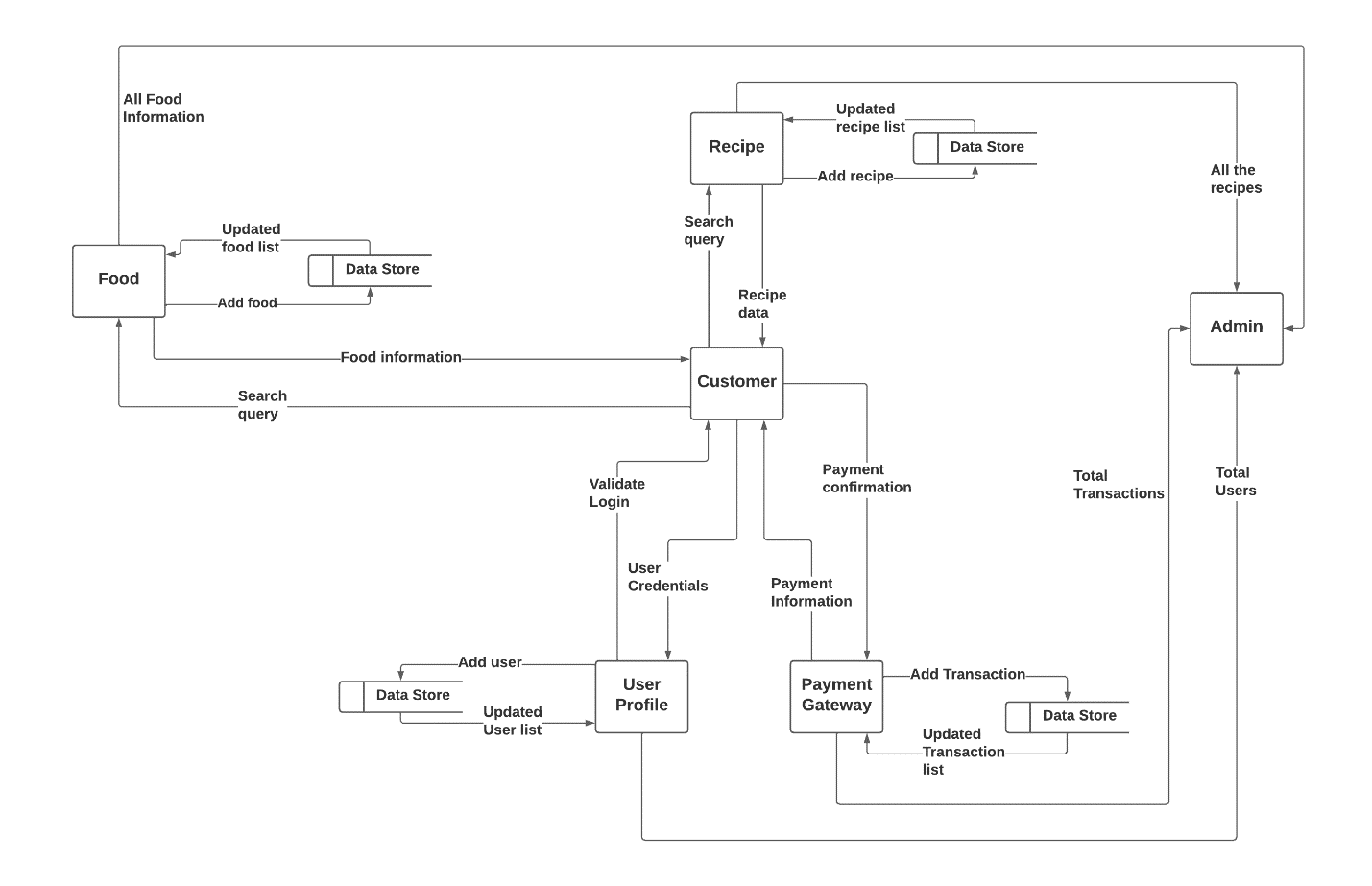
The process starts by users creating an account on the website. The user searches for food and drinks using keywords, then the system displays a list of results matching the search criteria. The search results show alongside them the nutritional information. The user selects the desired food or drink item and adds it to their meal plan or diary. The user can customize their meal plan by adjusting portion sizes, adding or removing items, and setting nutritional goals and then the order is finalized, readied for the user to come and take or be delivered. The system can generate a report showing the user's daily intake of calories, macronutrients, vitamins, and minerals, and compares it to their nutritional goals. The user can review the report and adjust their meal plan accordingly, based on their goals and preferences.

#### **4.2.1 Diagrams for Existing System**

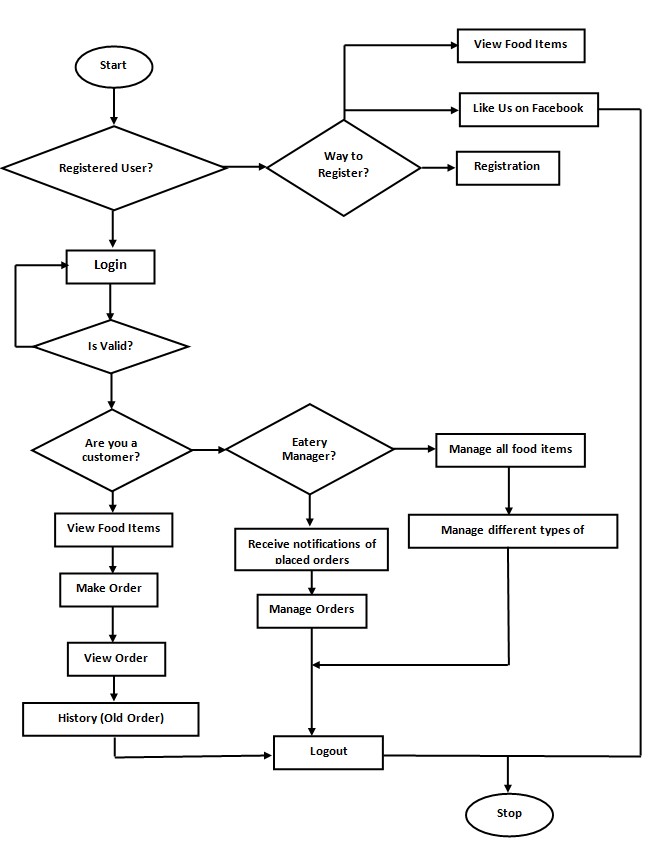
**Context Diagram**



**DFD level 1**



**Flowchart**



### **4.3 Proposed System**

#### **4.3.1 Functional Requirements**

1. Customers should be able to browse menus and add items to their order as well as customize the items to their liking.
2. Customers should be able to view the status of their orders, including when the orders were placed and when they are expected to be ready.
3. The system to should be able to support numerous restaurants and handle orders by sending them to the appropriate the restaurnts.
4. Restaurants should be able to publish their menus and update them anytime, where the menus show items available, calorie information and prices.
5. The system should be accessible and tailored for mobile use.
6. Customer should be able to examine their previous orders.
7. The system should allow the Admin to get report of customers, restaurants, orders and transactions.

#### **4.3.2 Non-functional Requirements**

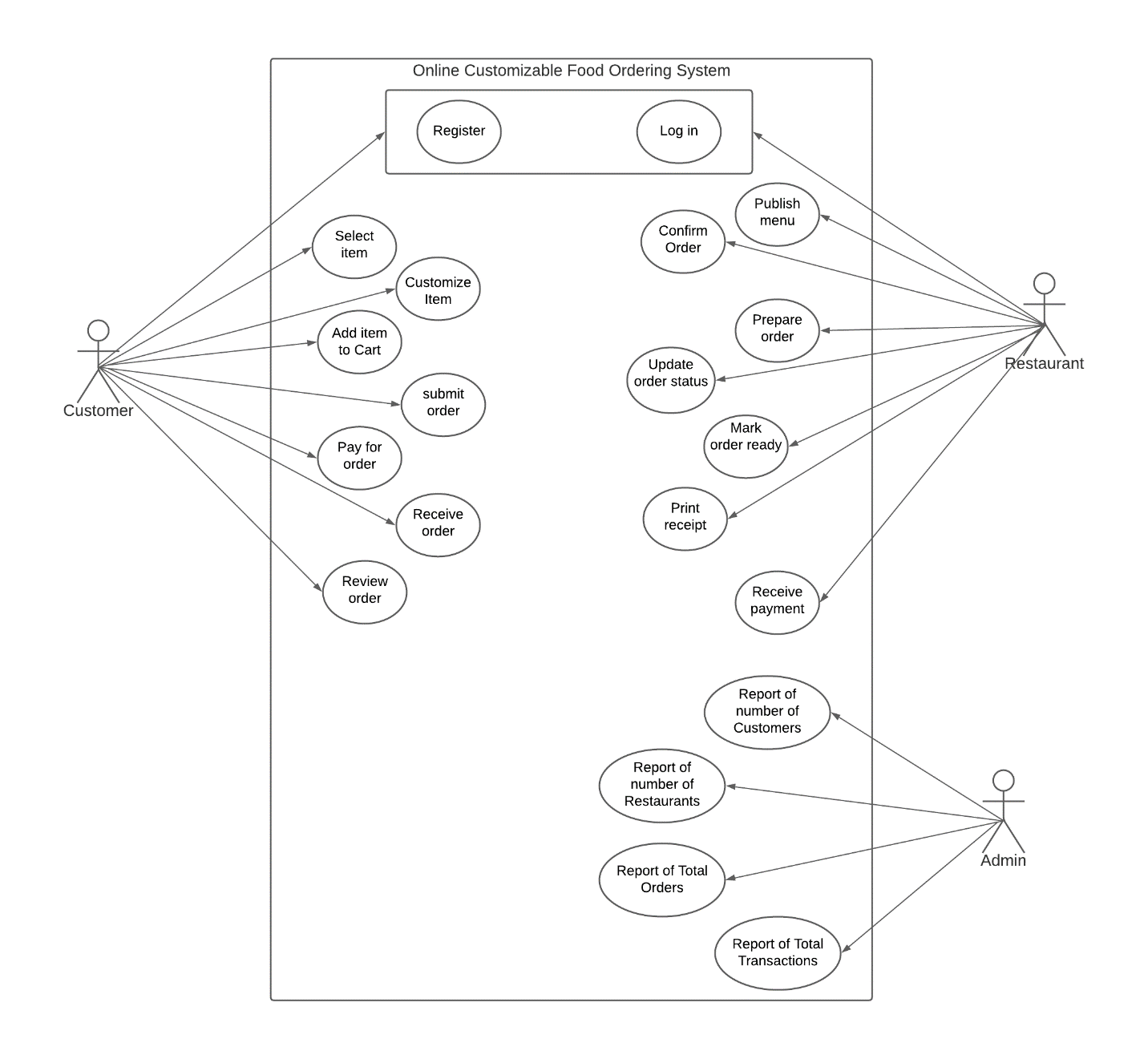
1. The user interface has to be simple and easy to use.
2. Performance: There should be no delays while the system is handling a high number of users.
3. The system should put in place the necessary security measures to safeguard the personal and private information of customers and restaurants.
4. The system should always be available and accessible to users.

#### **4.3.3 User requirements**

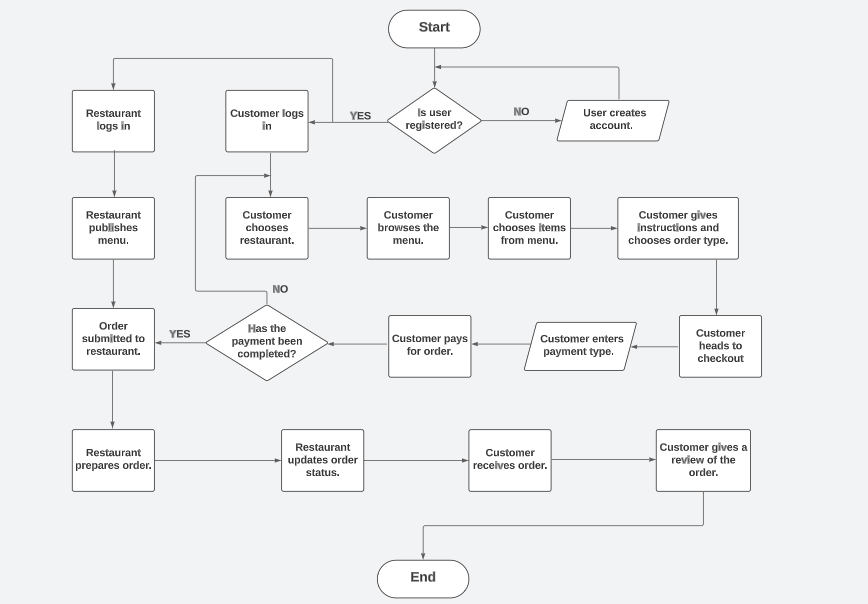
1. Easy to use interface: The system should have a simple and user-friendly interface that allows customers to place orders quickly and easily.
2. Customization options: Customers should be able to customize their orders to meet their dietary needs and preferences, such as adding or removing ingredients.
3. Menu management: The system should allow restaurants to manage their menus, including adding or removing items and updating prices.
4. Payment options: Customers should be able to pay for their orders using a variety of payment options, including credit cards, debit cards, and M-Pesa.
5. Order tracking: Customers should be able to track the status of their orders.
6. Mobile compatibility: The system should be compatible with mobile devices, allowing customers to place orders from their smartphones or tablets.

#### **4.3.4 Diagrams**

**Use Case Diagram**



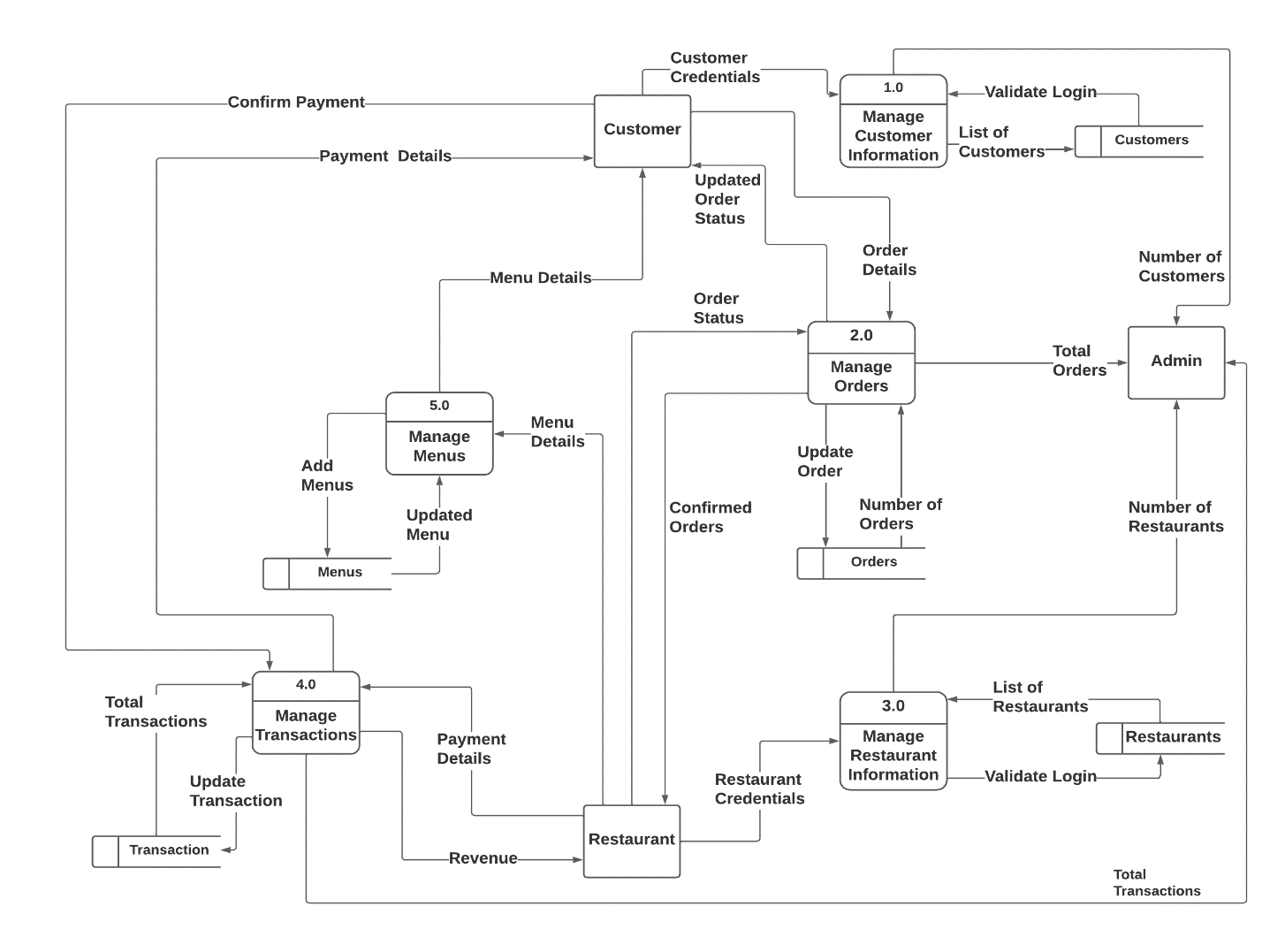
**Flowchart**



**Context Diagram**



**DFD level 1**



### **4.4 Data Collection Methods**

**One-On-One Interviews**

One-On-One Interviews can be an effective data collection method for both users and restaurants in an Online Customizable Food Ordering System. By conducting one-on-one interviews with users, the system developers can gather information about the specific needs and preferences of users, such as their preferred ordering methods, dietary restrictions, and desired customization options. One-on-one interviews can also be used to gather feedback on the ordering process, the quality of the food, and overall user experience. Similarly, one-on-one interviews with restaurants can provide valuable insights into their food preparation processes, menu customization capabilities, and any challenges they face in the current ordering system. This information can be used to develop a more effective and efficient ordering system that meets the needs of both users and restaurants. Overall, one-on-one interviews can be a valuable tool for gathering detailed information and feedback from users and restaurants in an Online Customizable Food Ordering System.

**Questions for Users:**

1. How often do you use online food ordering services?
2. What do you look for when choosing a restaurant for online ordering?
3. What types of food do you typically order online?
4. How important is nutritional information when ordering food online?
5. Have you ever encountered any issues or challenges when using online food ordering services?
6. What features or improvements would you like to see in an online food ordering service?

**Questions for Restaurants:**

1. What motivated you to offer online ordering for your restaurant?
2. How has online ordering impacted your business, both positively and negatively?
3. How do you handle custom orders and special requests through online ordering?
4. How important is nutritional information for your restaurant, and how do you provide it to customers who order online?
5. Have you ever experienced any technical difficulties with your online ordering system?
6. What features or improvements would you like to see in an online ordering system for your restaurant?

**Questionnaires**

Questionnaires can be used to gather feedback, preferences, and suggestions from users about their experience using the system, as well as to collect information from restaurants about their menu options, pricing, and delivery capabilities. The questionnaire can be designed to include specific questions that can help improve the system's features and functionalities based on the feedback received from users and restaurants.

**Customer Questionnaire:**

1. How often do you order food online?

a) Daily b) 2-3 times a week c) Once a week d) Less than once a week

1. What type of cuisine do you prefer?

a) Italian b) Arabian c) Swahili d) Indian e) Other (please specify)

1. Which payment method do you prefer when ordering food online?

a) Credit/debit card b) Online payment gateway (PayPal, Stripe, etc.) c) Cash on delivery

1. How important are the following factors when ordering food online? (**From Scale of 1-10**)

a) Fast delivery time b) Quality of food c) Menu variety d) User-friendly ordering interface e) Customer support f) Price competitiveness

1. How likely are you to recommend our online food ordering system to others?
2. Highly likely b) Somewhat likely c) Neutral d) Somewhat unlikely e) Highly unlikely

**Restaurant Questionnaire:**

1. How often do you receive online food orders?

a) Daily b) 2-3 times a week c) Once a week d) Less than once a week

1. Which online food ordering platforms do you currently use?

a) Uber Eats b) Glovo c) Jumia Foods d) Other (please specify)

1. How easy is it to manage orders on our online food ordering system?

a) Very easy b) Somewhat easy c) Neutral d) Somewhat difficult e) Very difficult

1. How important are the following features when using an online food ordering system? (**From Scale of 1-10**)

a) User-friendly interface b) Accurate order tracking c) Prompt customer support d) Integration with POS systems e) Customization options for menu and pricing f) Competitive pricing

1. How likely are you to recommend our online food ordering system to other restaurant owners?

a) Highly likely b) Somewhat likely c) Neutral d) Somewhat unlikely e) Highly unlikely

**Observation**

Observation can be an effective means of collecting data for an online customizable food ordering system. By observing users and their behavior when using the system, we can gain insights into how they interact with the platform, what features they use the most, and what issues they encounter. Here are some examples of how observation can be used to collect data for an online customizable food ordering system:

1. User testing: Conducting user testing sessions where participants are observed as they complete specific tasks on the system can provide valuable feedback on the user experience. Observing their behavior and collecting their feedback can help identify areas where the system can be improved.
2. Transaction monitoring: Observing the order fulfillment process in real-time can help identify potential bottlenecks and issues that customers may encounter. This data can be used to optimize the system and improve the overall customer experience.
3. Analytics tracking: Observing user behavior through analytics tracking tools can provide insights into how customers are using the system, which pages they visit the most, and what features they use the most. This data can be used to optimize the system and improve the overall user experience.
4. Restaurant operations monitoring: Observing restaurant operations, such as order preparation and delivery, can provide insights into how the system can be optimized to improve order accuracy and delivery times.