

REPUBLICUE DU CAMEROUN
Paix-Travail-Patrie
***** MINISTERE DE
L'ENSEIGNEMENT SUPERIEUR
***** INSTITUT AFRICAIN
D'INFORMATIQUE (IAI)

REPUBLIC OF CAMEROON
Peace-Work-Fatherland
***** MINISTRY OF HIGHER
EDUCATION
***** AFRICAN INSTITUT OF
COMPUTER SCIENCES (AICs)



BP 13 719 Yaoundé (Cameroon) Tel. (237) 22 72 99 57/ (237) 22 72 99 58
Web site: www.iaicameroun.com E-mail: contact@iaicameroun.com

A MOBILE FOOD SALES APPLICATION

Developed by

Group 16 Members (SE3A):

- ❖ JONATHAN ZALI NDA VALY 65%
- ❖ Ntse Lennvstephen Mancho 35%

Supervised By

Mr. FOMEKONG Nze Stephane

**Academic Year:
2024/2025**

MOBILE FOOD SALES APP

GENERAL INTRODUCTION	4
PRESENTATION OF THE PROJECT	5
CONTEXT.....	5
JUSTIFICATION.....	5
PROJECT TARGET	6
ANALYSIS METHODOLOGY	7
Introduction.....	7
Comparison of UML and Merise.....	7
DIAGRAMS	8
USE CASE DIAGRAM	8
TEXTUAL DESCRIPTION.....	11
COMMUNICATION DIAGRAM.....	16
SEQUENCE DIAGRAM	18
ACTIVITY DIAGRAM	20
STATE MACHINE DIAGRAM	22
CLASS DIAGRAM	23
PACKAGE DIAGRAM.....	25
COMPONENT DIAGRAM	26
DEPLOYMENT DIAGRAM	27
PROJECT ILLUSTRATIONS.....	28
CONCLUSION	30

LIST OF FIGURES

Figure 1: formalism of the use case	8
Figure 2: general use case diagram	10
Figure 3: Purchase product.....	11
Figure 4: formalism of communication diagram	16
Figure 5: Purchase product communication diagram	17
Figure 6: authentication sequence diagram.....	18
Figure 7: Purchase product sequence diagram	19
Figure 8: activity diagram formalism.....	20
Figure 9: authentication activity diagram	20
Figure 10: Purchase product activity diagram.....	21
Figure 11: formalism of state diagram	22
Figure 12: Order state diagram	22
Figure 13: class diagram formalism.....	23
Figure 14 System class diagram:.....	24
Figure 15: A package diagram formalism	25
Figure 16: system package diagram	25
Figure 17 component diagram formalism.....	26
Figure 18: system component diagram.....	26
Figure 19: deployment diagram format	27
Figure 20: System deployment diagram.....	27
Figure 21: Onboarding.....	28
Figure 22: Splash screen	28
Figure 23: home screen	28
Figure 24: login screen	28
Figure 25: Landing screen.....	28
Figure 26: add product screen.....	29
Figure 27: checkout screen.....	29
Figure 28: products screen	29
Figure 29: product categories page.....	29
Figure 30: track order screen	29
Figure 31: user profile screen.....	29

LIST OF TABLES

Table 1: components of communication diagram	16
Table 2: components of a class diagram	23

GENERAL INTRODUCTION

Food sales are a cornerstone of community nourishment and local economic vitality, offering a platform that celebrates culinary diversity while driving growth in small businesses. The Food Sales Mobile Application is designed to transform the way consumers discover, order, and enjoy food. By connecting local restaurants, food vendors, and farmers with customers, the app delivers a streamlined experience featuring real-time menus, online ordering, and efficient delivery tracking. With its user-friendly interface and integrated smart technologies, the application not only simplifies the food buying process but also helps preserve and promote regional culinary traditions.

PRESENTATION OF THE PROJECT

CONTEXT

Local food markets and culinary establishments are vibrant hubs that showcase a region's unique gastronomic heritage. However, many consumers face challenges when trying to access authentic local flavors due to scattered information, outdated ordering methods, and limited digital presence of small vendors. The Food Sales Mobile Application addresses these issues by consolidating diverse food offerings into a single, accessible digital platform. It leverages modern technologies such as GPS-based delivery tracking, multimedia menus, and real-time inventory updates to enhance both the customer and vendor experience. This approach supports the growth of local food ecosystems while ensuring that culinary treasures are just a tap away.

JUSTIFICATION

The development of the Food Sales Mobile Application is driven by several key factors:

1. **Lack of Centralized Ordering Platforms** – Consumers often rely on fragmented sources to discover local food options. This app creates a one-stop solution for exploring and ordering a wide range of culinary delights.
2. **Enhanced Customer Experience** – With an intuitive interface, real-time updates, and visually engaging menus, the application simplifies the ordering process and elevates user satisfaction.
3. **Empowerment of Local Businesses** – By offering an accessible digital space, the app enables local restaurants, street vendors, and farmers to reach a broader audience, driving economic benefits and encouraging culinary innovation.
4. **Integration of Smart Technologies** – Features such as online payments, order tracking, and personalized recommendations provide a modern, efficient, and secure shopping experience.
5. **Sustainability and Cost Efficiency** – The digital format reduces reliance on paper menus and physical receipts, promoting environmentally friendly practices while lowering operational costs.
6. **Safety and Convenience** – With secure transactions, contactless delivery options, and real-time order monitoring, the app ensures a safe and convenient food purchasing experience for all users.

PROJECT TARGET

The Food Sales Mobile Application is designed to serve a diverse group of stakeholders who share an interest in quality food and local culinary experiences. The primary targets include:

⊕ Consumers (Local and International)

- Local residents seeking convenient access to a variety of quality food options.
- International visitors eager to explore authentic local cuisine.

⊕ Local Food Vendors and Restaurants

- Establishments aiming to expand their digital presence and showcase their unique offerings.
- Street food vendors and boutique eateries looking for innovative ways to reach new customers.

⊕ Grocery Stores and Farmers

- Small-scale producers and local farmers who wish to sell fresh produce and specialty items directly to consumers.

⊕ Government and Regulatory Authorities

- Agencies focused on promoting food safety, supporting small business growth, and enhancing regional culinary tourism.

⊕ Students and Culinary Enthusiasts

- Individuals interested in discovering local recipes, food traditions, and emerging culinary trends.

⊕ Health-Conscious and Eco-Friendly Consumers

- Customers looking for organic, sustainably sourced, and locally produced food options, with detailed information on ingredients and sourcing practices.

By addressing these diverse needs, the Food Sales Mobile Application aims to revolutionize the food sales landscape, fostering a closer connection between consumers and the rich culinary heritage of the region while promoting local economic development.

ANALYSIS METHODOLOGY

Introduction

The project analysis for the Tour Guide Application involves employing the Unified Modeling Language (UML) as the primary modeling tool. UML is a standardized visual language that enables us to effectively capture and analyze the various aspects of the project, from requirements to design and implementation. By utilizing UML, we can create clear and concise visual representations of the system, facilitating communication and understanding among stakeholders and development teams. This introduction sets the stage for how UML will be used throughout the project analysis.

Comparison of UML and Merise

When it comes to modeling methodologies, UML offers distinct advantages over the Merise methodology. While Merise focuses primarily on data modeling, UML provides a more comprehensive approach, encompassing the modeling of system behavior, structure, and interactions. UML allows us to create a broader range of diagrams, such as use case diagrams, class diagrams, and sequence diagrams, enabling a more holistic understanding of the system.

Unlike Merise, UML is not limited to database design but offers a versatile modeling approach that can be adapted and extended as the project evolves. UML's flexibility allows for easy modification and scaling, accommodating changing requirements and ensuring the system's future growth. This adaptability is crucial for the digitalized mortuary application, as it needs to cater to the evolving needs of mortuaries and the families they serve.

Overall, UML's comprehensive nature and flexibility make it a superior choice for our project analysis. It provides us with the means to capture a complete system representation, communicate effectively, and adapt to future changes. By utilizing UML, we can ensure a robust and well-designed digitalized mortuary application that meets the needs of both mortuary staff and grieving families.

DIAGRAMS

USE CASE DIAGRAM

a. Definition

Use case diagram shows the functionalities of a system, their interdependencies and how they relate with actors of the system. A use case is a specification of behaviour. The main objectives of the use case diagram are:

- ❖ Provide a high-level view of the system.
- ❖ Identify the functions of the system.

Use case diagrams are completed with a textual description of each use case that is intended to define the use case in greater details

b. Formalism

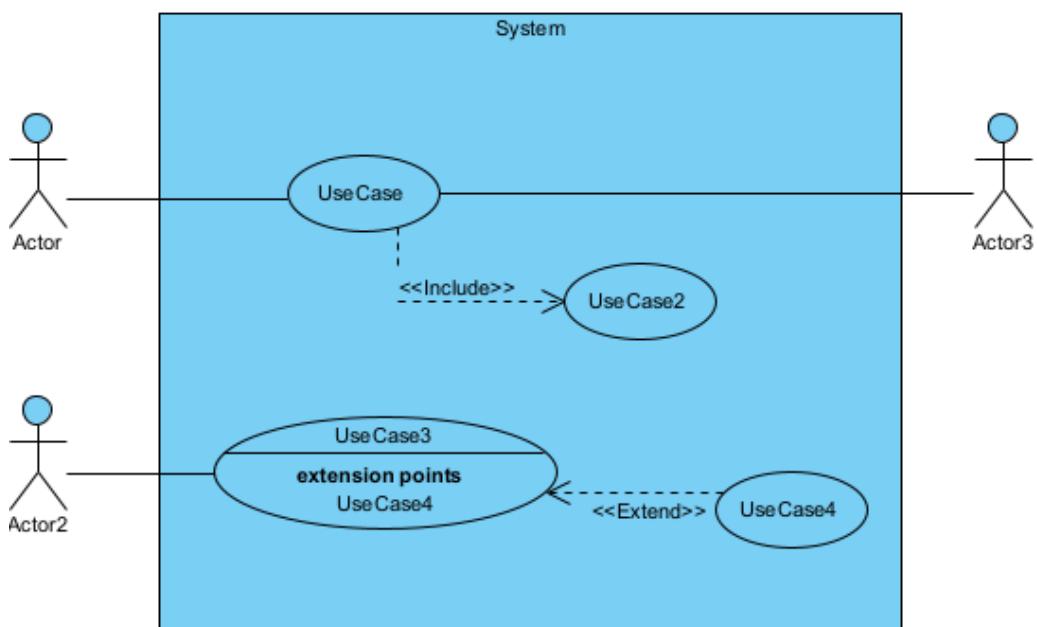


Figure 1: formalism of the use case

c. Components of a use case diagram

Table 1: components of a use case

Elements	Diagrammatic Representation	Description
Actor		Represents an entity that directly interacts with the system. The actor is what performs the different possible actions of the system
Use case		A use case represents a functionality of the system. It is an action that can be performed by an actor.
Association		It indicates that an actor takes part in a use case.
Include		An inclusion denotes that an included action must be performed before the including action can be performed.
Extend		An extension denotes that an extending action may be performed while an extended action is being performed.
Generalization		This shows that an actor or a use case is a kind of another abstract or concrete actors can be defined and later specialized using generalization relationship.
System		It is a container of use cases which interact with external actors

d. Actors of the System

Table 2: actors of the system

System Actors	
Admin	The user that is in charge of controlling the entire system
Vendor	A particular food seller or a registered restaurant
Customer	A registered user that can order for food
Delivery Agent	An actor in charge of transferring the ordered food(s) from the vendor to the customer

e. General Use case

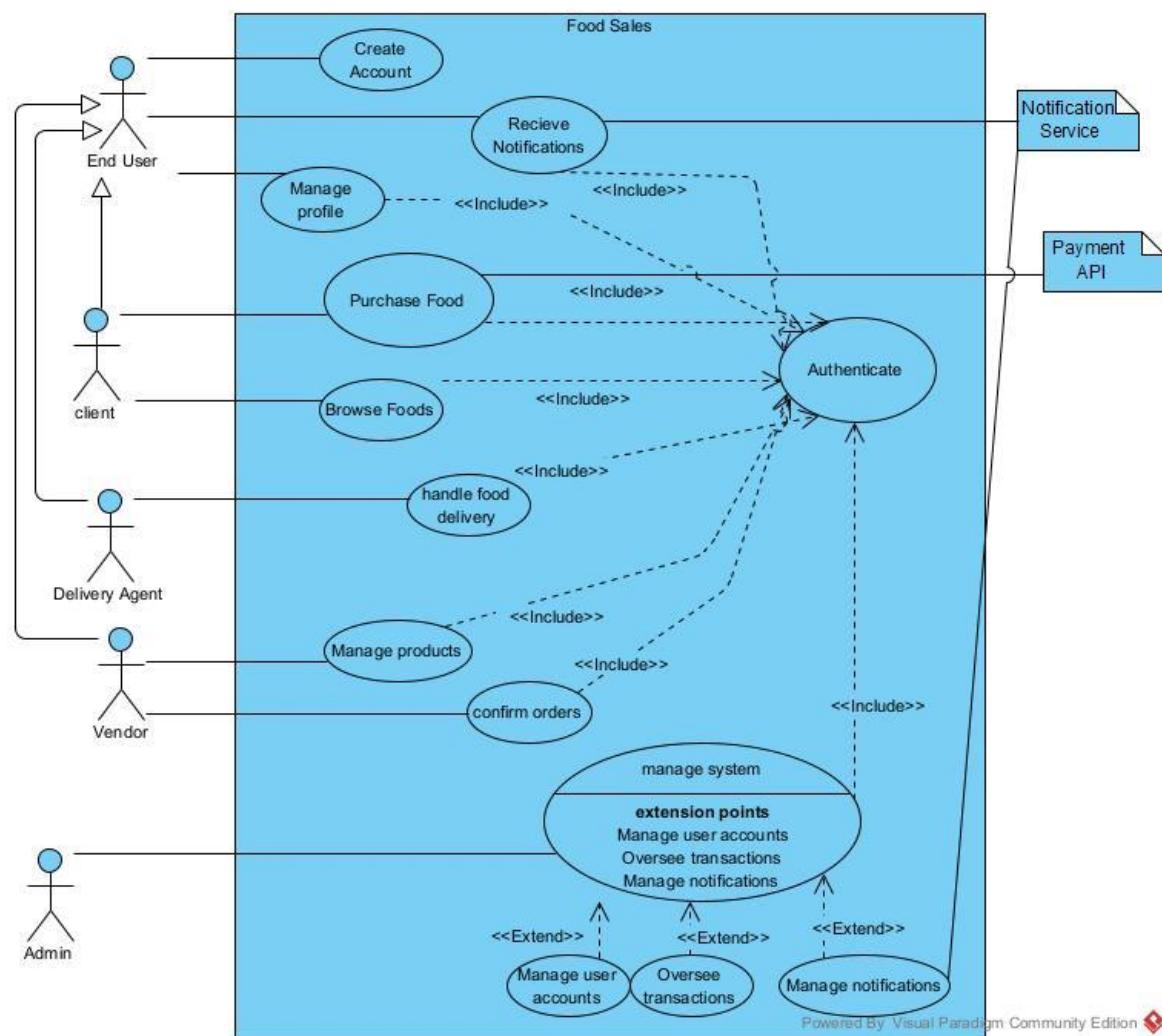


Figure 2: general use case diagram

f. Purchase Product

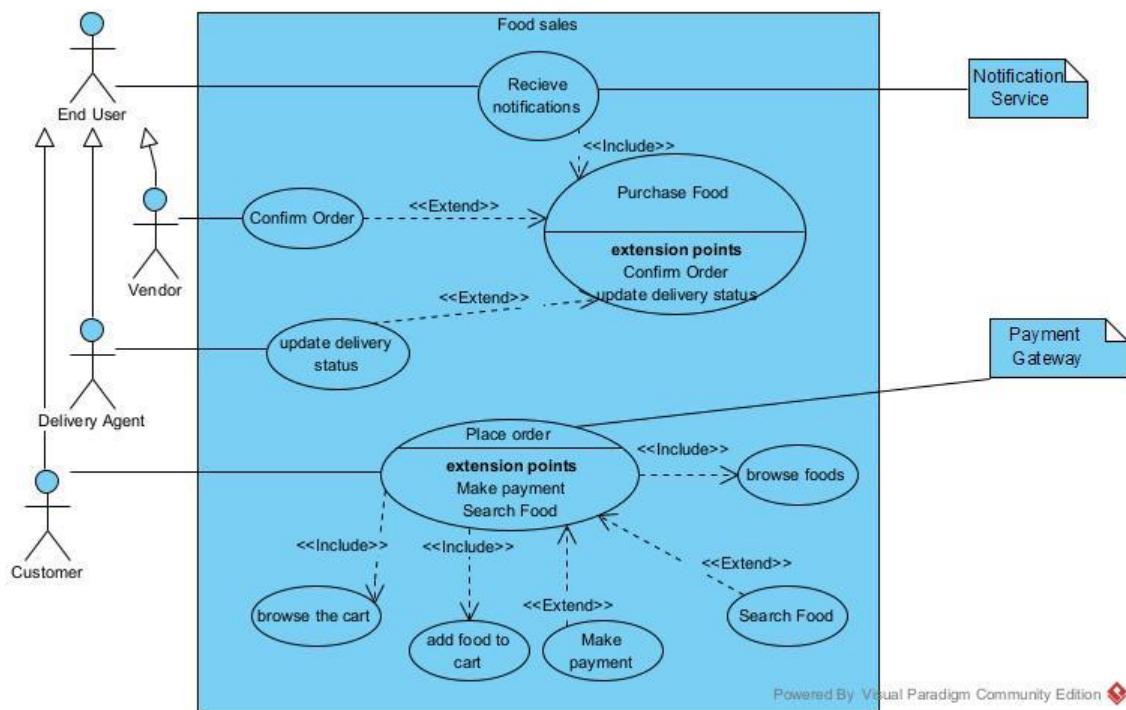


Figure 3: Purchase product

TEXTUAL DESCRIPTION

a. Authentication

Table 3: Authentication textual description

Title	Authenticate
Summary	The user needs to authenticate
Actors	Customer, Delivery agent, Vendor, admin
Date	February 11, 2025
Stakeholder	Food Sales Directors
Version	1.0
Precondition (s)	<ol style="list-style-type: none"> 1. The app is launched. 2. The actor has an account on the platform.
Triggers	The user clicks on login button

MOBILE FOOD SALES APP

Nominal Scenario	<ol style="list-style-type: none">1. The system displays the login form.2. The actor fills and submits the form.3. The system verifies conformity of the form fields4. The system sends the data to the dbms.5. The dbms returns result of the query.6. The system displays a success message to the actor.
Alternative Scenarios	<ol style="list-style-type: none">7. At step 4 of the nominal scenario, the user enters mismatched or missing information.8. The system displays an error message then returns to step 2 of the nominal scenario.
Postcondition of success	The user has access to the homepage of the application
Postcondition of failure	The user does not have access to the platform
Non-functional requirement	Entering the password must not be visible on the screen

b. Purchase product

Table 4: Purchase product textual description

Title	Purchase Product
Summary	The customer wants to purchase a product from the website
Actors	End Users (Customer, Vendor, Delivery Agent)
Date	January 19, 2025
Stakeholder	Food Sales directors
Version	1.0
Precondition (s)	<ul style="list-style-type: none"> • Actor must be authenticated
Trigger (s)	Actor clicks on the reserve ticket button
Nominal scenario	<ol style="list-style-type: none"> 1. The customer enters the application URL in a web browser 2. System displays the home page with products 3. Customer selects products and adds to their shopping cart by clicking the add button 4. The system confirms the addition 5. Customer clicks on the cart icon 6. System displays the cart page with product details and price 7. Customer reviews the cart to confirm the items and total price and proceeds to checkout by clicking on the checkout button 8. The system displays the checkout form 9. Customer fills and submits the form 10. System verifies the conformity of the form and queries the information to the DBMS 11. The DBMS execute the query by creating the Order and sends the result to the system

	<p>12. The system notifies the Farmer of the order via the notification service</p> <p>13. The Farmer reviews and confirms the order</p> <p>14. The system displays the payment form</p> <p>15. The customer fills and submits the form</p> <p>16. The system checks for form conformity and forwards the details to the payment gateway</p> <p>17. The gateway performs the transaction and sends a response back to the system.</p> <p>18. The system forwards the order details to the delivery agent via the notification service and notifies the customer of the delivery progress</p> <p>19. The delivery agent gets the pickup and drop-off addresses of the order and performs the delivery accordingly</p> <p>20. The delivery agent then updates the delivery status of the Order</p>
Alternative scenario	3.a the customer may equally search for a product of her choice before selecting
Exceptional scenario	<p>10.a. At step 10 of the nominal scenario, the actor enters wrong information</p> <p>10.b. The system displays error message and returns to step 8 of the nominal scenario.</p> <p>11.a At step 11 of the nominal scenario, the DBMS returns a failed response</p> <p>11.b The system displays an error message and returns to step 8 of the nominal scenario</p> <p>13.a At step 13 of the nominal scenario, the farmer cancels the order request</p>

	<p>13.b The system displays the error message and returns to step 2 of the nominal scenario</p> <p>16.a At step 16 of the nominal scenario, the customer fills in invalid information</p> <p>16.b. The system displays the error message and returns to step 14 of the nominal scenario</p> <p>17.a At step 17 of the nominal scenario, The payment gateway returns a failed message</p> <p>17.b the system displays the error message and returns to step 14 of the nominal scenario</p>
Postcondition of success	The customer successfully receives the product she purchased
Postcondition of failure	The customer doesn't get the product
Non-functional requirement	<ul style="list-style-type: none"> • There must be good internet connection • The system should be user friendly for customers to easily navigate • The payment information of the customer should be encrypted • The system must comply with data protection and regulations of user information

COMMUNICATION DIAGRAM

a. Definition

Communication Diagrams model the interactions between objects in a sequence. They describe both the static structure and the dynamic behavior of a system. It is a simplified version of a Collaboration Diagram introduced in UML 2.0. A communication diagram is more focused on showing the collaboration of objects rather than the time sequence.

b. Formalism

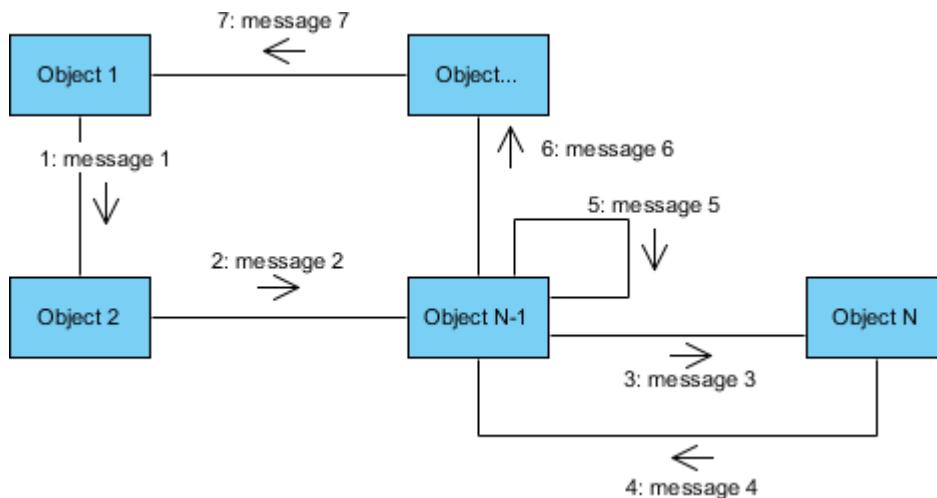


Figure 4: formalism of communication diagram

a) Components of a communication diagram

Table 1: components of communication diagram

Elements	Diagrammatic Representation	Description
Object		An object represents an individual participant in the interaction conversation
Actor		An actor is a special object in charge of initiating conversations
message	1: message 1 	Defines a particular communication between objects
link		Represents the relationship that exists between lifelines

c. Purchase Product Communication Diagram

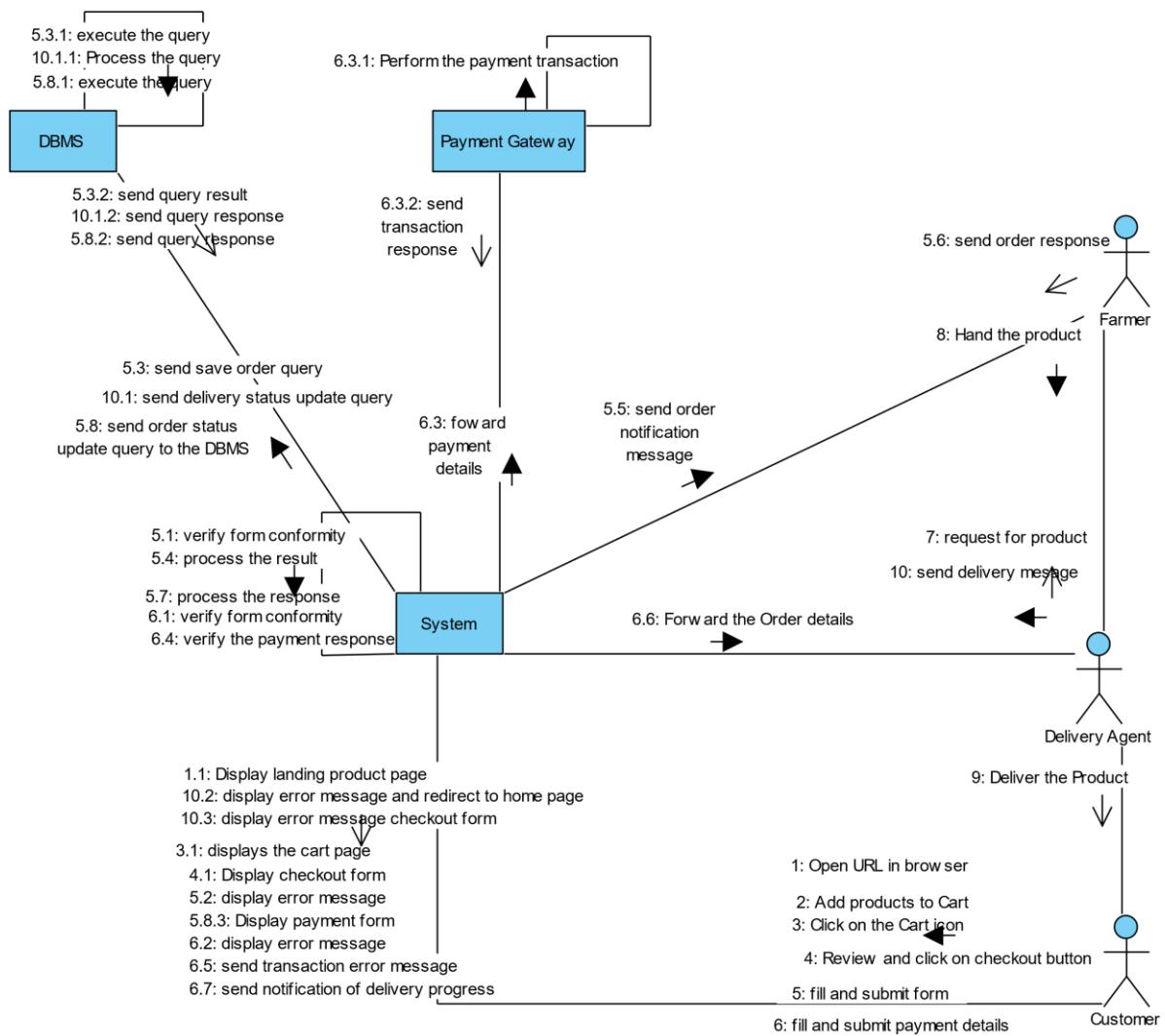


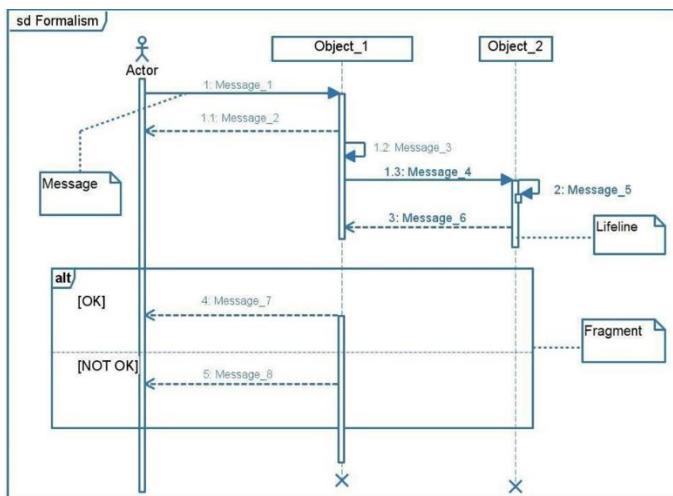
Figure 5: Purchase product communication diagram

SEQUENCE DIAGRAM

a. Definition

A Sequence diagram describes interactions among classes in terms of an exchange of messages over time. They are also called event diagrams. A Sequence diagram is a good way to visualize and validate various runtime scenarios. These can help to predict how a system will behave and to discover responsibilities a class may need to have in the process of modelling a new system.

b. Formalism



c. Authentication Appointment diagram

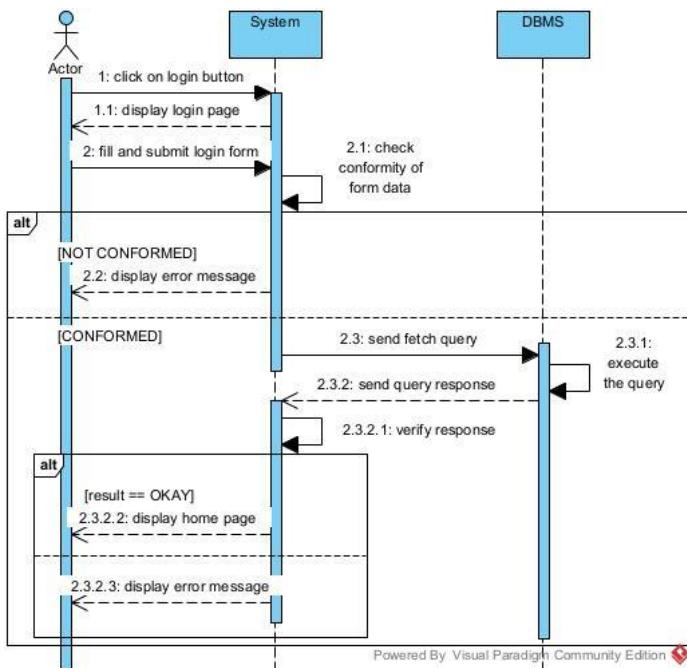


Figure 6: authentication sequence diagram

d. Purchase product sequence diagram

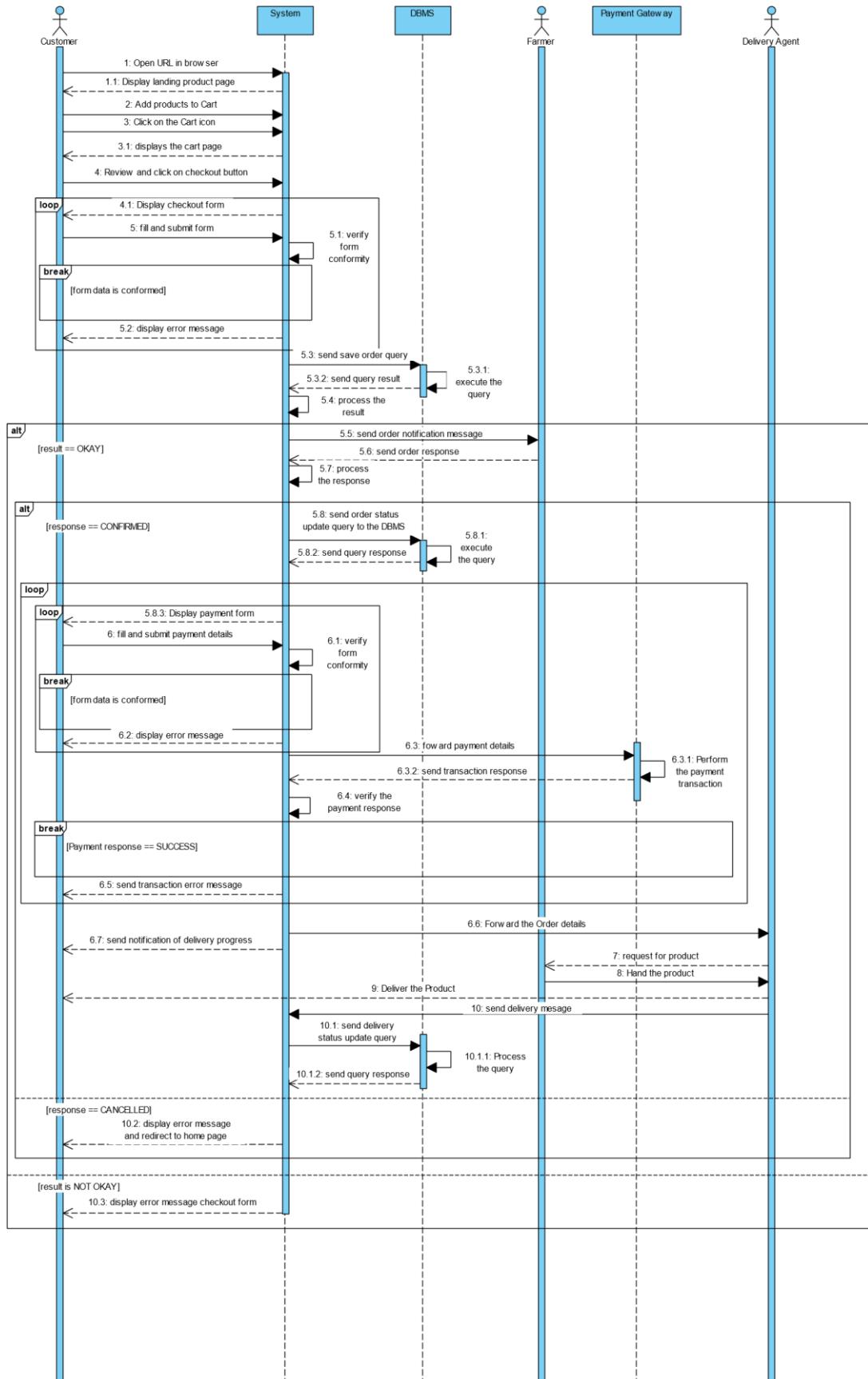


Figure 7: Purchase product sequence diagram

ACTIVITY DIAGRAM

a. Definition

An activity diagram is a graphical representation of workflows that show the steps needed in the realization of a process; showing the details from a start point to an end point through all decisions and actions that can possibly be performed. Activity diagrams are intended to model both the computational and organizational process. They flow can be sequential, branched or concurrent. Below is an activity diagram formalism.

b. Formalism

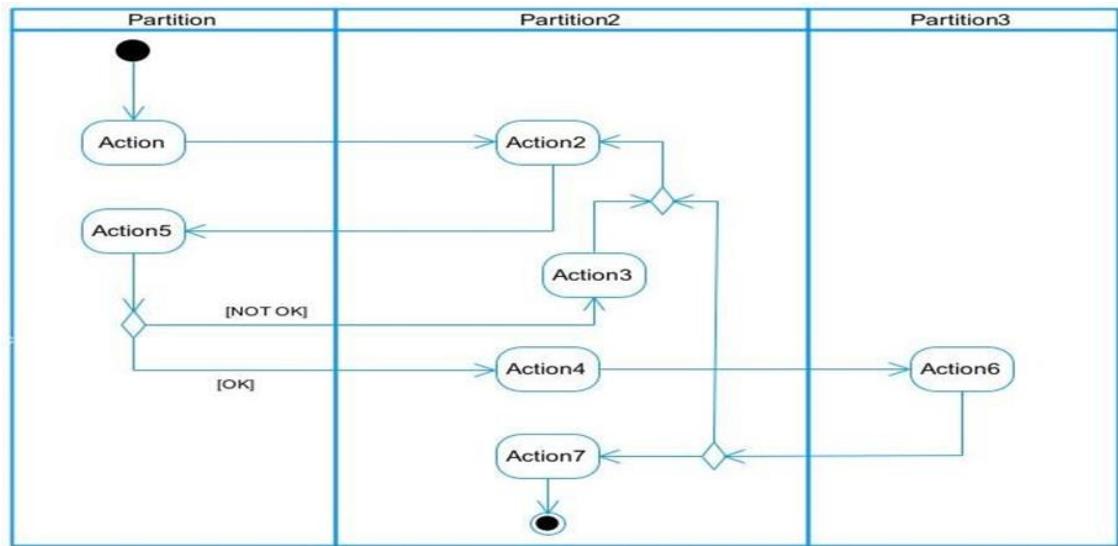


Figure 8: activity diagram formalism

c. Authentication Activity

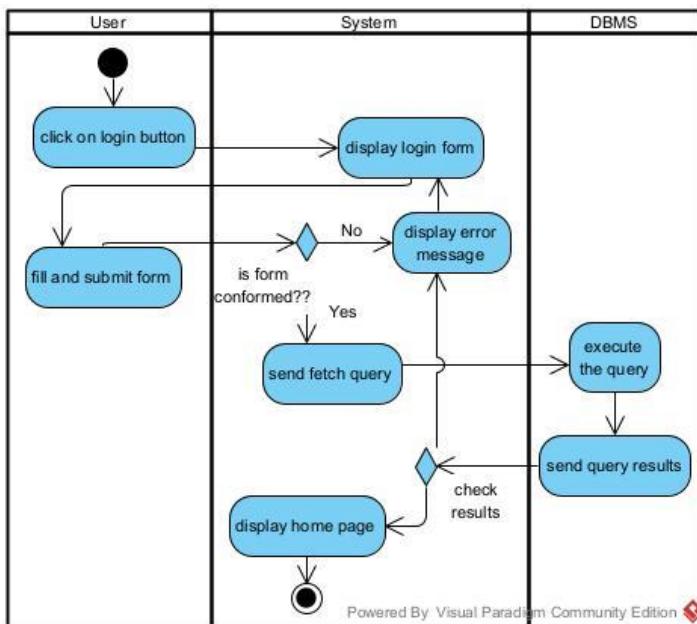


Figure 9: authentication activity diagram

d. Explore activity diagram

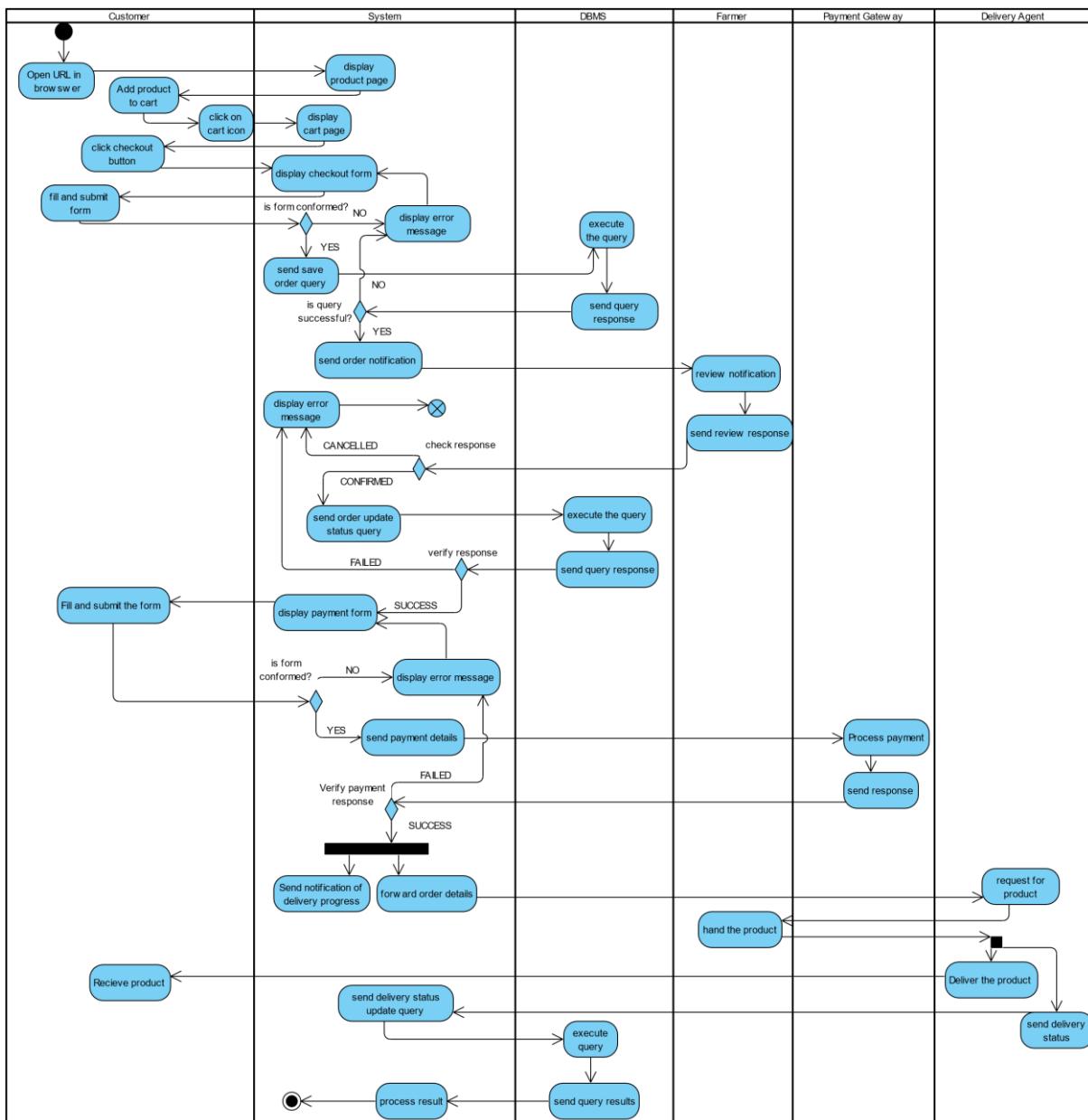


Figure 10: Purchase product activity diagram

STATE MACHINE DIAGRAM

a. Definition

A state machine diagram describes the behavior of a single object in response to a series of events in a system. Also known as the state machine diagram, it models the dynamic flow of control from the state of a particular object within a system.

b. Formalism

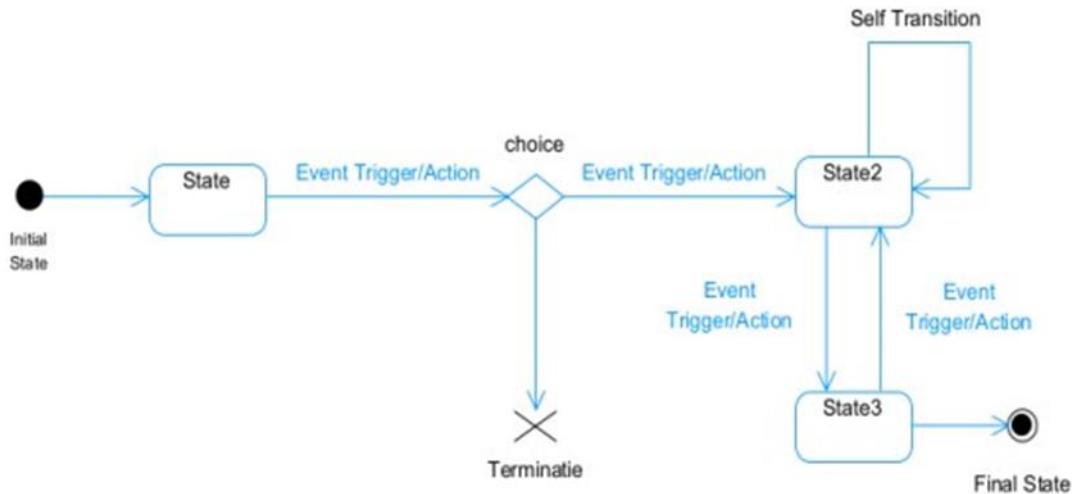


Figure 11: formalism of state diagram

c. Order State Diagram

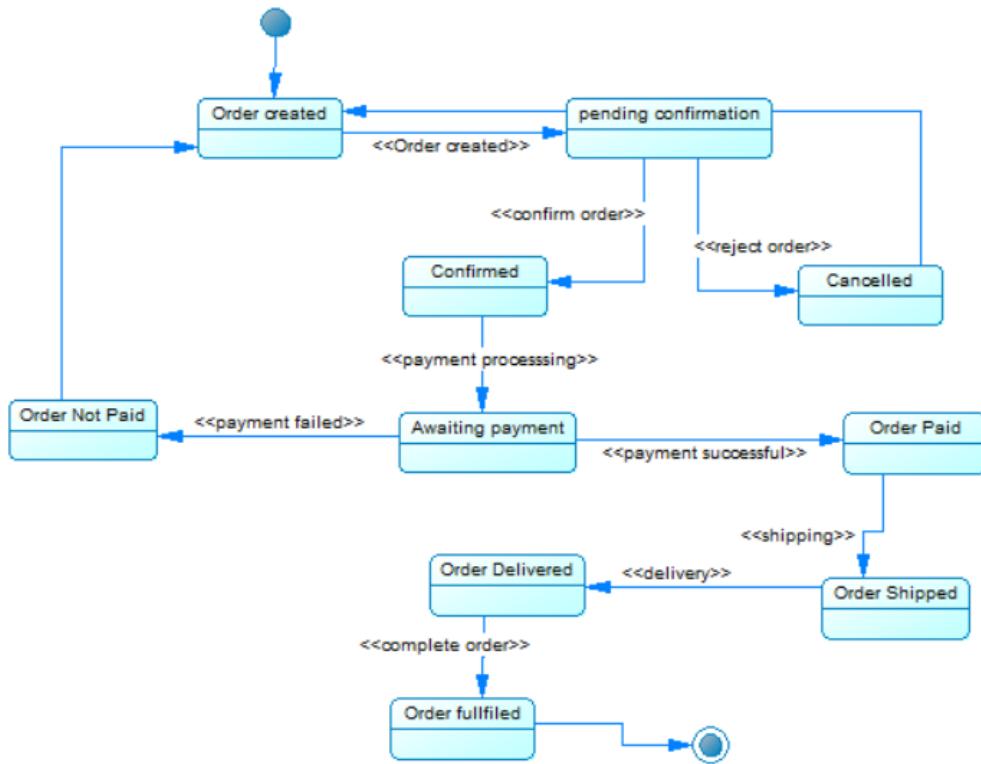


Figure 12: Order state diagram

CLASS DIAGRAM

a. Definition

A class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing and documenting different aspect of the system but also for constructing executable code of the software application. Class diagram describes the attribute and operation of a class and constraints imposed on the system. Its purpose is to model the static view of an application.

b. Formalism

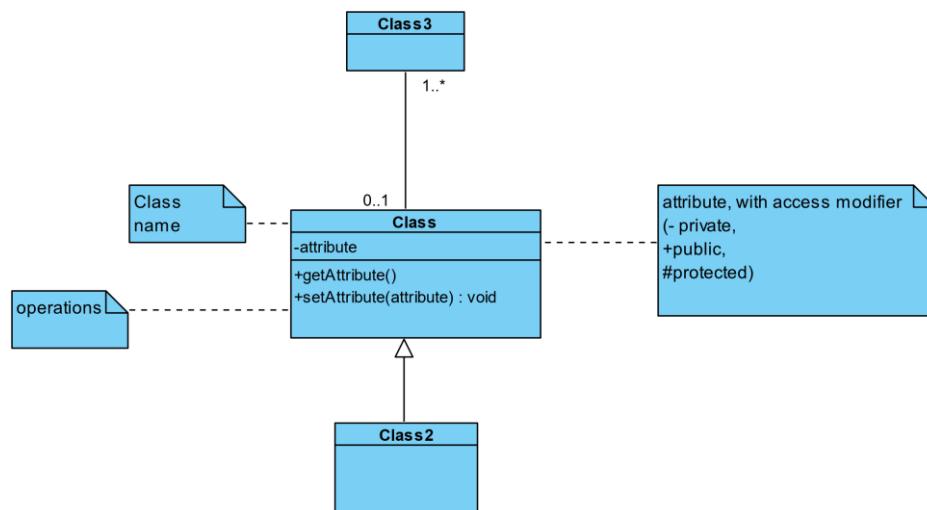


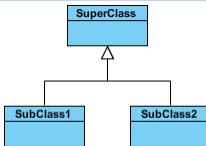
Figure 13: class diagram formalism

c. Components of A class Diagram

Table 2: components of a class diagram

Element	Diagrammatic Representation	Description
Class	<pre> Class -attribute +getAttribute() +setAttribute(attribute) : void </pre>	A class is an element that defines the structure and behaviours that an object can possess.
Aggregation	<pre> Class <--> Class2 </pre>	If the parent of the aggregate is deleted, usually the children are not deleted.
Composition	<pre> Class <--> Class2 </pre>	If a parent of a composite is deleted, usually, all its parts are deleted with it.
Dependency	<pre> Class --> Class2 </pre>	It exists between two classes, where a change in one influence the other.

MOBILE FOOD SALES APP

Generalization		It's a relationship between a whole class (called superclass) and a more specific class (called subclass)
Association		It is a general type of relationship between elements, it may include cardinality, roles etc.

d. System Class Diagram

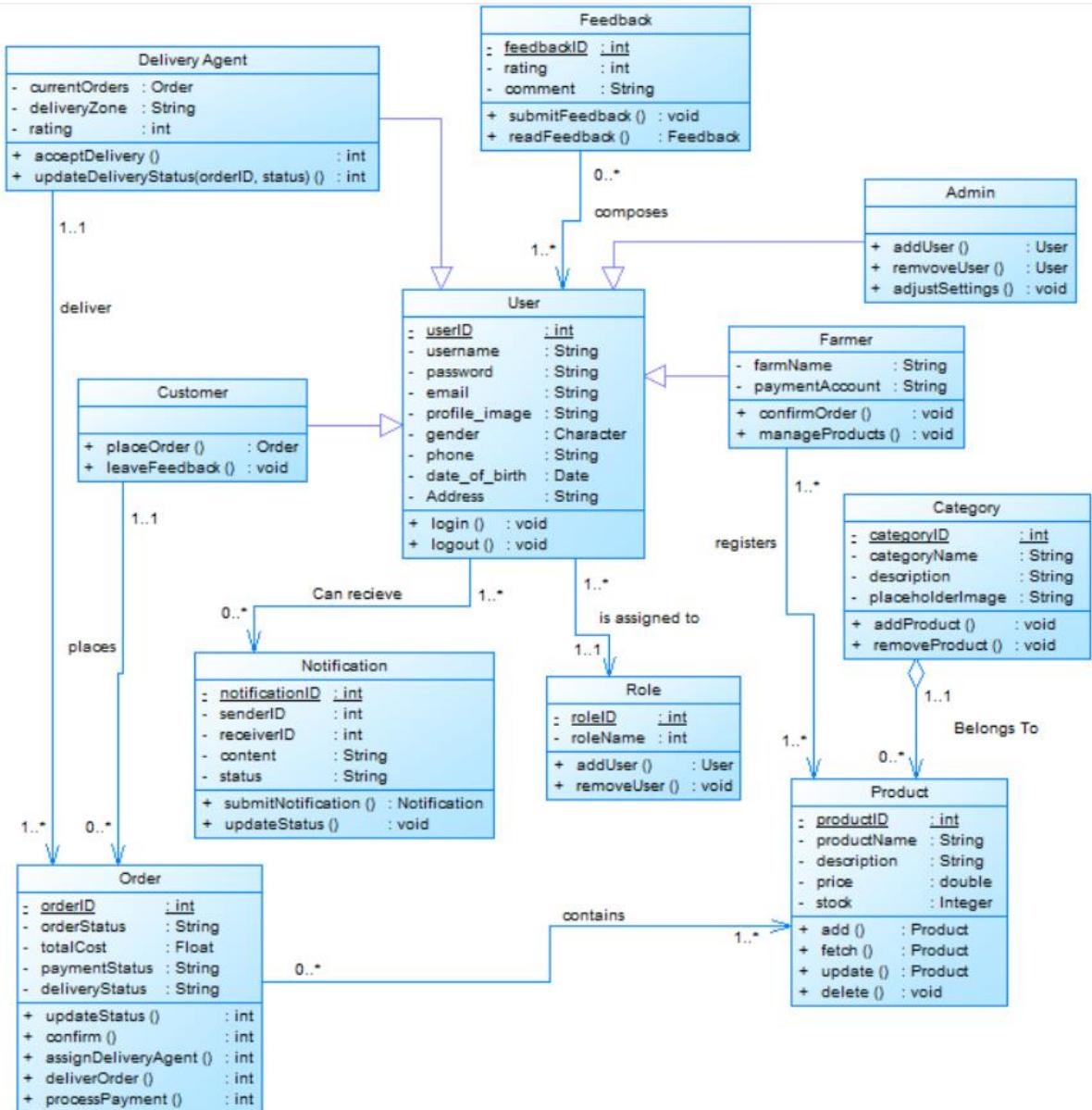


Figure 14 System class diagram:

PACKAGE DIAGRAM

a. Definition

This is a structural diagram used to show the organization and arrangement of various model elements in the form of packages. A package diagram is the grouping of related UML elements such as classes, diagrams or even other packages.

b. Formalism

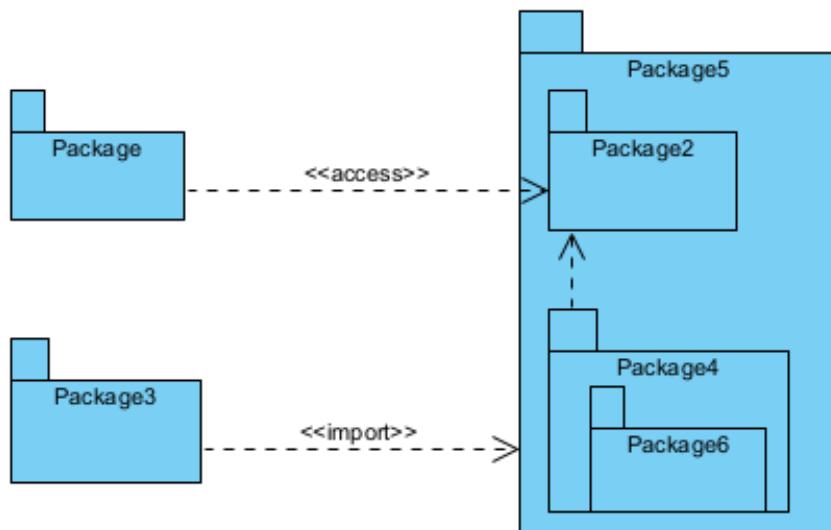


Figure 15: A package diagram formalism

c. System package diagram

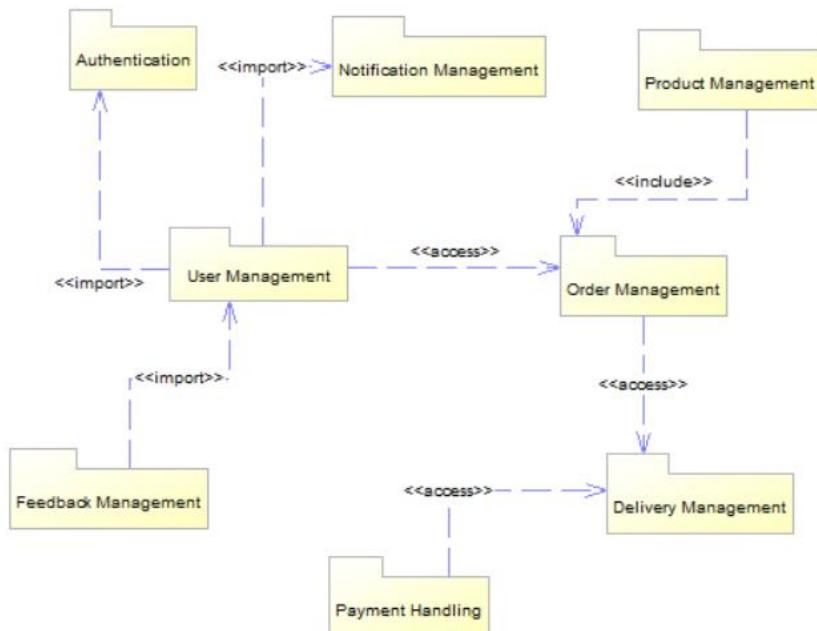


Figure 16: system package diagram

COMPONENT DIAGRAM

a. Definition

Component diagrams are used to model the physical aspect of a system. Now the question is what are this physical aspect? They are elements such as Executables, libraries, files, document etc. which resides in a node. The component diagram does not describe the functionality of the system, but it describes the components used to make those functionalities.

b. Formalism

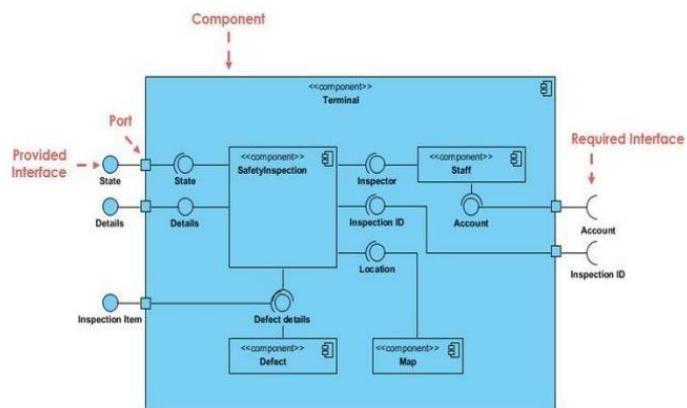
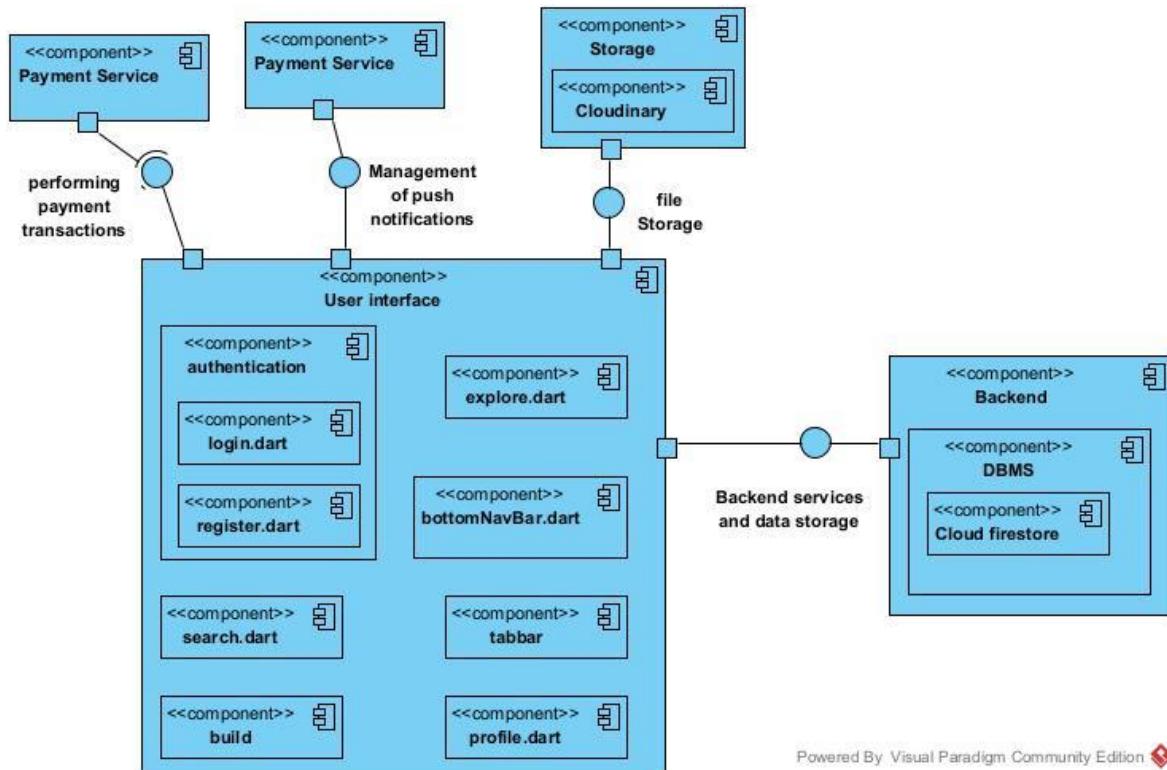


Figure 17 component diagram formalism

c. System Component Diagram



Powered By Visual Paradigm Community Edition

Figure 18: system component diagram

DEPLOYMENT DIAGRAM

a. Definition

Deployment diagram is a structural diagram used to visualize the topology of the physical components of a system, where the software is deployed. They consist of nodes and their relationship. It is related to the component diagram because the components are deployed using the deployment diagram. A deployment diagram consists of nodes.

Nodes are nothing but physical hardware used to deploy the application.

b. Formalism

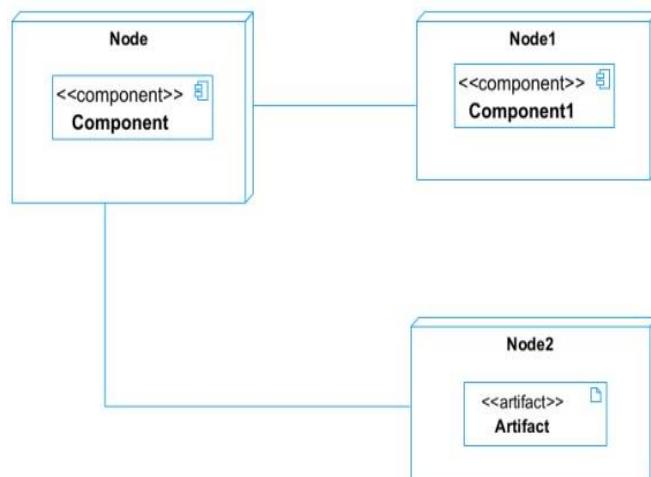


Figure 19: deployment diagram format

c. System Deployment Diagram

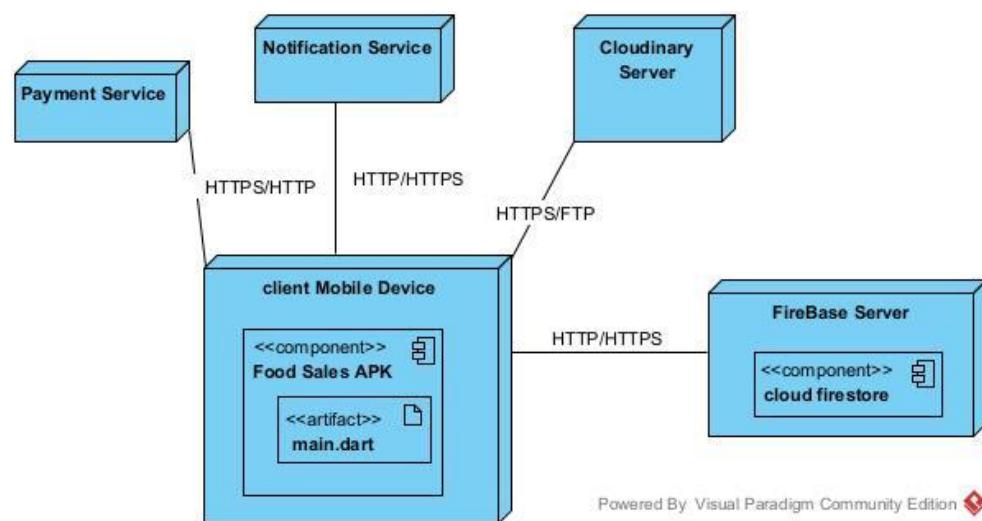


Figure 20: System deployment diagram

MOBILE FOOD SALES APP

PROJECT ILLUSTRATIONS



Figure 22: Splash screen

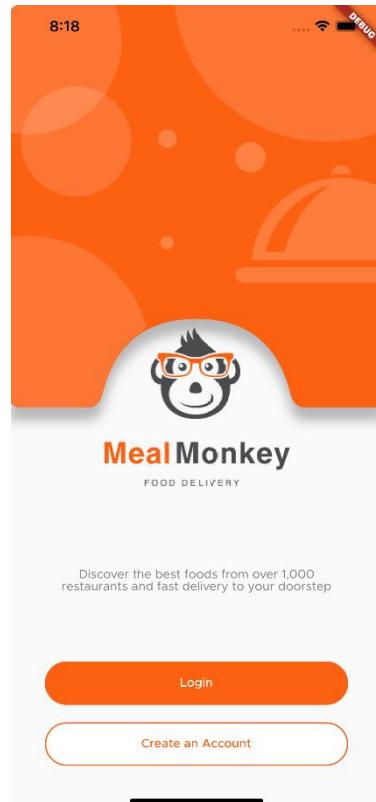


Figure 25: Landing screen

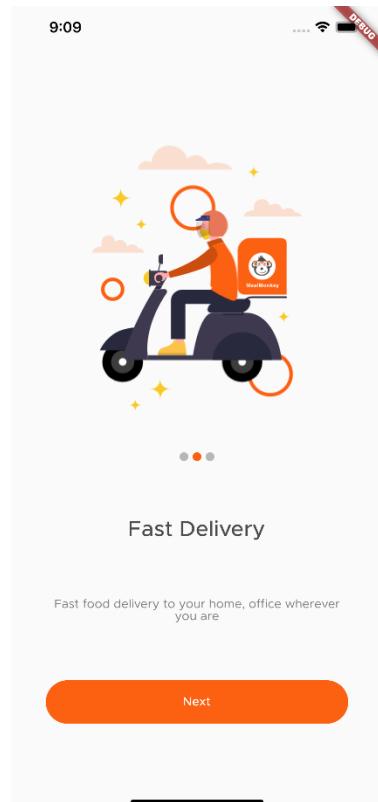


Figure 21: Onboarding

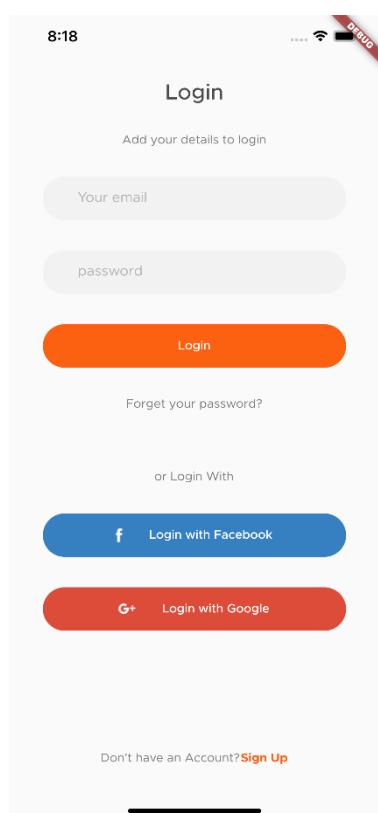


Figure 24: login screen

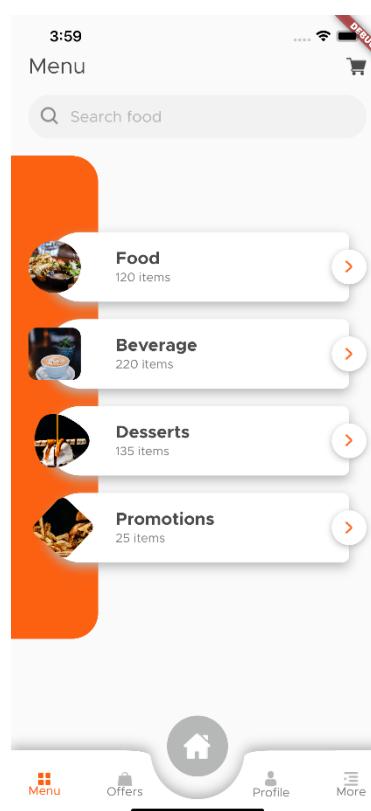


Figure 23: home screen

MOBILE FOOD SALES APP

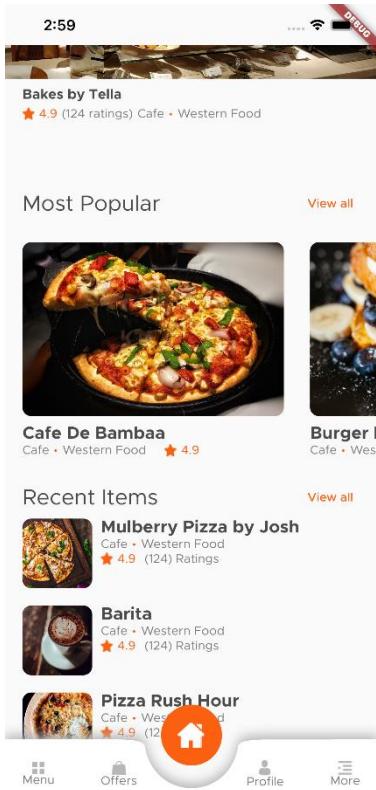


Figure 28: products screen

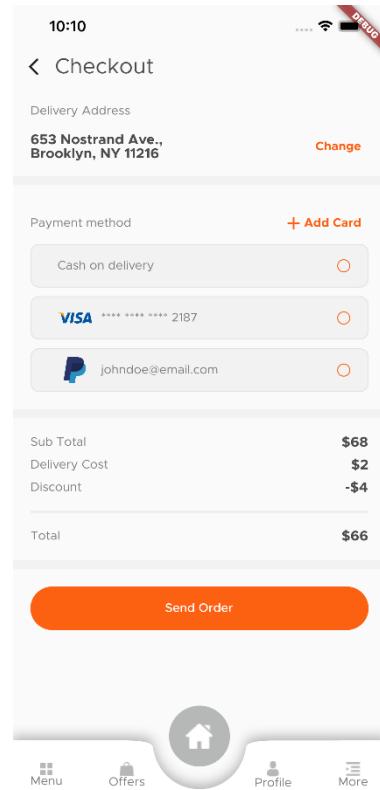


Figure 27: checkout screen

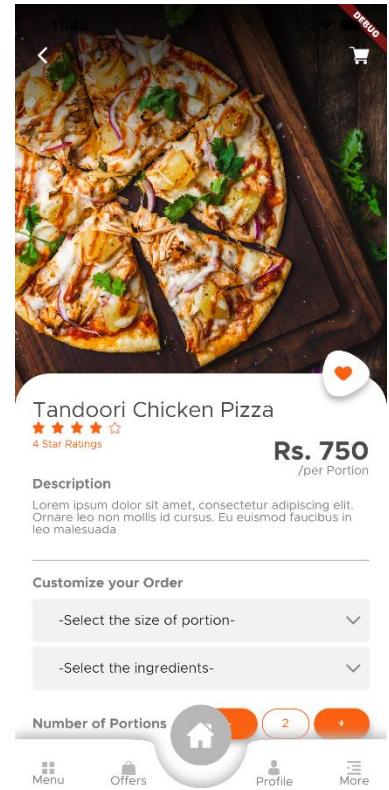


Figure 26: add product screen

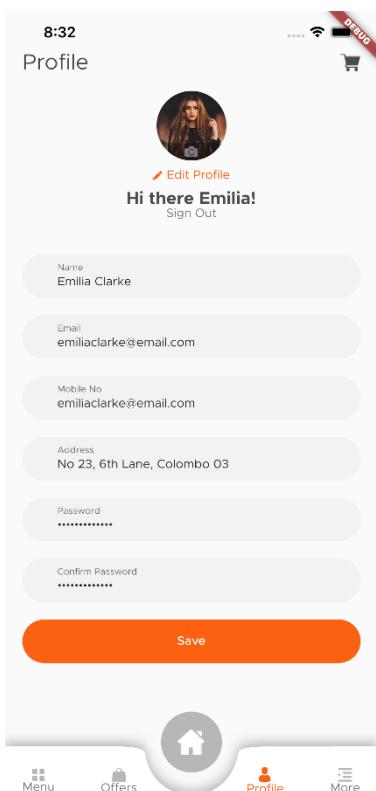


Figure 31: user profile screen

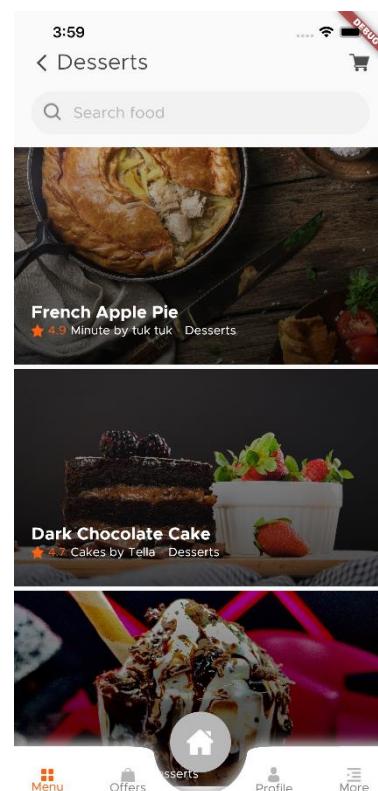


Figure 29: product categories page

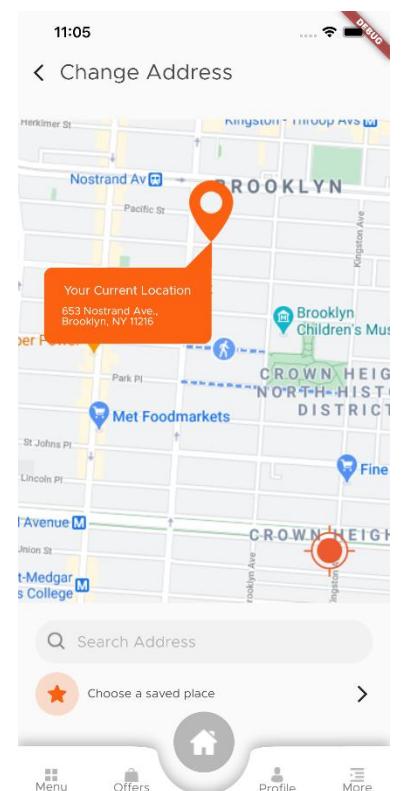


Figure 30: track order screen

CONCLUSION

The development of this food sales app provides an efficient and user-friendly platform for customers to browse, order, and receive food products seamlessly. By integrating real-time order tracking, multiple payment options, and a structured role-based system (admin, vendor, delivery agent, and customer), the app ensures a smooth and reliable experience.

With features like push notifications, multilingual support, and offline order management, the application enhances accessibility and convenience for users. Additionally, the use of robust technologies such as **Flutter for the frontend, Firebase for Backend** ensures scalability and performance.

Moving forward, further improvements could include AI-based recommendations, loyalty programs, and enhanced analytics to optimize vendor performance and user engagement. By continuously refining the app based on user feedback, the platform can grow into a sustainable and competitive solution in the food sales industry.