



# HATLEY

Information System Department



Dr. Ali Hussein

# HATLEY

## (Delivery Website &app)

Graduation Project Report

Prepared by

**Abdullah Salah**

**Abd El-Rahman Omar**

**Ahmed Ali Farahan**

**Santy Osama Mina**

**Rahma Bahaa**

Academic Supervisor

**Dr. Ali Hussein Ahmed**

Faculty of computers and information (information system),

Assiut University

2023-2024

# Table of Contents:

<b>Table of Contents .....</b>	<b>2</b>
<b>Abstract.....</b>	<b>4</b>
<b>1. Software Proposal .....</b>	<b>5</b>
1.1 Introduction.....	5
1.2 Problem statement.....	6
1.2.1 User Perspective.....	6
1.2.2 Delivery Perspective .....	7
1.3 Objectives .....	7
<b>2. Software Analysis.....</b>	<b>9</b>
2.1 Introduction.....	9
2.1.1 Problem Analysis and Motivation.....	9
2.1.2 Scope of the Project .....	10
2.1.3 Target User Group .....	11
2.2 System Requirement .....	12
2.2.1 Project Sponser .....	12
2.2.2 Business Need .....	12
2.2.3 Business Value .....	13
2.3 Fesability Study.....	13
2.3.1 Technical Feasibility .....	14
2.3.2 Economica Feasibility .....	15
2.3.3 Organizational Feasibility .....	16
2.4 The Survey .....	16
2.5 Definition of Requirements.....	22
2.5.1 Functional Requirements .....	22
2.5.2 Non-Functional Requirements .....	25
2.6 Use Cases .....	27
2.6.1 Register Use Case .....	27
2.6.2 Login Use Case .....	28
2.6.3 Display requests and Interaction Use Case .....	29
2.6.4 Create Order Use Case .....	30
2.6.5 Accept offer Use Case .....	31
2.6.6 Introduce offer Use Case.....	31
2.6.7 Notification and Alert Use Case .....	32
2.6.8 Customer Support Use Case.....	33
2.6.9 Driver Ratings for Customers Use Case .....	34
2.6.10 Request Tracking and Delivery Use Case .....	35
2.6.11 User Ratings and Comments Use Case .....	37
2.6.12 Location and Address Selection Use Case .....	38
2.6.13 Order Confirmation and Payment Use Case .....	39
2.6.14 Request Processing and Assignment Use Case .....	40
2.6.15 Admin control Use Case .....	42
2.7 Management Plan.....	43
2.7.1 Current State of the Project .....	43
2.7.2 Phases.....	44
2.7.3 Gantt Chart & Timeline .....	46
2.7.4 Task Plan with Milestones .....	47
2.7.5 Division of Responsibilities Among Team Members .....	48
2.7.6 Risk Management .....	50
<b>3. Software Design.....</b>	<b>51</b>
3.1 Introduction.....	51
3.1.1 Project Background/Purpose .....	51
3.1.2 Hosting Platform .....	51
3.2 System Architecture .....	52
3.2.1 Architectural Design .....	53
3.2.2 Decomposition Description .....	53
3.2.3 Design Rationale .....	54

3.3 Data Design.....	55
3.3.1 Data Description .....	55
3.3.2 Data Dictionary .....	56
3.4 Components Design .....	57
3.4.1 User Authentication Module .....	57
3.4.2 Data Storage Module .....	59
3.4.3 Data Processing Module .....	61
3.4.4 Data Flow Diagram .....	63
3.5 User Interface Design.....	69
3.6 Testing and Quality Assurance .....	86
<b>4. Software Requirements Specification .....</b>	<b>88</b>
4.1 Introduction.....	88
4.1.1 Intended Audience and Reading Suggestions .....	88
4.1.2 Product Scope .....	88
4.2 Overall Description.....	89
4.2.1 Product Perspective.....	89
4.2.2 Product Functions .....	89
4.2.3 User Classes and Characteristics.....	90
4.2.4 Operating Environment.....	90
4.2.5 Design and Implementation Constraints .....	90
4.2.6 Assumptions and Dependencies.....	91
4.3 External Interface Requirements.....	91
4.3.1 Hardware Interfaces .....	91
4.3.2 Software Interfaces .....	91
4.3.3 Communications Interfaces.....	91
4.4 System Features .....	92
4.5 Other Non-functional Requirements .....	93
4.5.1 Performance Requirements .....	93
4.5.2 Safety Requirements .....	93
4.5.3 Security Requirements .....	93
4.5.4 Software Quality Attributes .....	94
4.5.5 Business Rules .....	94
<b>5. Conclusion.....</b>	<b>95</b>

# Abstract:

*Welcome to Hatley! Hatley is a platform that connects users with delivery personnel, providing a faster and more convenient way to purchase daily essentials in Egypt.*

*Hatley is a comprehensive delivery platform designed to provide a seamless and convenient delivery experience for our users in Egypt. Our goal is to create a competitive marketplace for delivery drivers, encouraging them to provide faster and more cost-effective delivery options for users.*

*Hatley is a website that connects users with delivery drivers who compete to fulfill user orders in the shortest amount of time for a competitive price. By allowing users to submit their requests for products and specify a preliminary price and time, the platform creates a competitive marketplace for delivery drivers, encouraging them to provide faster and more cost-effective delivery options for users.*

*Delivery men can then compete for these orders by either accepting the offer or presenting a competitive alternative in terms of time and cost, it can lead to faster and more efficient deliveries.*

*At Hatley, we strive to provide a fast, efficient, and transparent delivery service that meets the needs of modern users and delivery personnel. Our platform offers a user-friendly app, real-time notifications and updates, and a rating and review system for both users and delivery personnel.*

*Hatley can help users save time and effort when it comes to ordering their daily needs. The idea of competition between delivery men based on time and price is an interesting concept that can lead to faster and more efficient deliveries.*

# 1. Software Proposal

## 1.1 Introduction:

Hatley is a web-based delivery platform that is designed to provide a more competitive and efficient delivery ecosystem in Egypt. The platform connects users with delivery drivers who compete to fulfill user orders in the shortest amount of time for a competitive price. By allowing delivery drivers to take the full price of delivery, Hatley provides them with more earning opportunities and promotes a more transparent and accountable delivery service for users.

The Website includes several key features, such as user registration and account management, product request submission and tracking, competitive delivery pricing and time offers, notification system for order updates, user-friendly interface, payment system, and review and rating system for delivery personnel and user experience. These features are designed to provide a seamless and convenient delivery experience for both users and delivery drivers.

Hatley allows users to order a variety of products by submitting their request through the app, specifying a preliminary service price and time. Delivery personnel compete for these orders by either accepting the offer or presenting a competitive alternative in terms of time and cost. Notifications are specifically sent to the customer, giving them the freedom to choose the best delivery option based on price and time. The delivery personnel then proceed to any nearby or distant store where the product is available, purchase it, and deliver it to the location specified by the customer through the app. This creates job opportunities for young individuals and transforms the Egyptian market into a competitive space, enhancing product quality and increasing revenue for some stores without the need for advertisements.

In this software proposal, we will outline the features and functionality of the Hatley platform, as well as the development process and timeline. We will also provide an overview of the project scope, including project objectives, user requirements, and technical specifications.

## 1.2 Problem statement:

### 1.2.1 User Perspective:

people often face significant challenges when it comes to purchasing daily essentials. The country's notorious traffic congestion can make it difficult and time-consuming to go to physical stores, especially during peak hours. Additionally, long queues and limited parking spaces can further add to the frustration and inconvenience of in-store shopping.

Moreover, People with busy schedules may face challenges in accessing delivery services due to their limited time and the need to go to multiple stores for their daily needs.

These challenges can lead to a lack of access to a wide range of products and services, as people may not have the time or patience to visit multiple stores to find what they need. This can be particularly challenging for those who live in remote or underserved areas, where access to physical stores may be limited.

Moreover, small and medium-sized stores often struggle to compete with larger stores, who have the resources and scale to offer lower prices and a wider range of products. This can lead to lower revenue and limited product availability for smaller stores, which can further limit access to essential goods and services for consumers.

In addition to the delay of the delivery companies in responding and waiting a long time to confirm your order, then waiting for another period of time until the delivery company arrives at the place of order, purchases it, and then brings it to you. This takes a very long time and is considered a waste of time.

Hatley aims to address these challenges by providing a fast, convenient, and transparent delivery service that connects users with a wide range of products and services.

By allowing delivery personnel to compete for orders based on price and time, the platform can help to incentivize faster and more efficient delivery services, while also providing users with greater choice and flexibility. Additionally, by providing job opportunities for young individuals and helping to transform the Egyptian market into a more competitive space, the platform can help to increase revenue and product availability for smaller stores, while also providing consumers with greater access to essential goods and services.

### 1.2.2 Delivery Perspective:

The traditional delivery market is often characterized by high delivery fees, lack of competition, and undervalued delivery drivers. Delivery companies often take a large percentage of the delivery fee, leaving delivery drivers with a smaller portion of the earnings. This can lead to delivery drivers feeling undervalued and exploited, as well as a lack of competition in the delivery market. Additionally, high delivery fees can be a burden for users and small businesses, limiting their ability to access delivery services.

Hatley aims to address these issues by allowing delivery drivers to take the full price of delivery, providing them with more earning opportunities and promoting a more transparent and accountable delivery service for users. The platform's competitive pricing and time offers can also benefit users and small businesses by providing more cost-effective delivery options.

By promoting a more competitive and efficient delivery ecosystem, Hatley aims to benefit both users and delivery drivers, promoting a more sustainable and profitable delivery market. The platform's unique business model and features can help address the issues of undervalued delivery drivers and high delivery fees, providing a more equitable and affordable delivery solution.

## 1.3 Objectives:

Hatley has many objectives, this including the following:

- To develop a user-friendly and accessible website and mobile app that allows users to easily submit their requests for daily essentials and track the delivery status.
- To ensure transparency and fair competition, Hatley can implement various measures, such as:
  1. Clearly displaying delivery fees and prices for each delivery option on the platform.
  2. Providing users with the option to sort and filter delivery options based on price, delivery time, and other relevant factors.
  3. Encouraging delivery personnel to provide accurate and up-to-date information about their delivery times, prices, and service quality.
  4. Implementing a rating and review system that allows users to rate and review delivery personnel based on their service quality and delivery time.
- To provide a fast and efficient delivery service that connects users with delivery personnel. Hatley aims to provide a fast and efficient delivery service that connects users with delivery personnel. By allowing delivery personnel to compete for orders based on price and time,

Hatley can help to incentivize faster and more efficient delivery services, ultimately benefiting both delivery personnel and users.

- To provide job opportunities for young individuals in the delivery industry. Hatley aims to provide job opportunities for young individuals in the delivery industry by connecting them with users who need delivery services. By allowing delivery personnel to compete for orders based on price and time, Hatley can help to create a more level playing field for young and inexperienced delivery personnel, ultimately helping to promote competition and innovation in the delivery industry.
- To help transform the Egyptian market into a more competitive space. Hatley aims to help transform the Egyptian market into a more competitive space by providing a platform for delivery personnel to compete for orders based on price and time. By promoting competition and innovation in the delivery industry, Hatley can help to drive down delivery fees and improve service quality, ultimately benefiting both delivery personnel and users.
- To increase revenue for smaller stores. Hatley aims to increase revenue for smaller stores by connecting them with users who need delivery services. It allows to compete with larger stores, Hatley can help to level the playing field and promote competition.

# 2. Software Analysis

## 2.1 Introduction:

### 2.1.1 Problem Analysis and Motivation:

The Problem Analysis and Motivation phases involve understanding the challenges faced by consumers in purchasing daily essentials, the limitations of small and medium-sized stores, and the potential benefits of a delivery service.

#### Problem Analysis (PA):

The Problem Analysis phase involves understanding the problem domain, identifying the requirements, and determining the constraints that the solution must satisfy. This phase involves gathering and analyzing information about the problem, such as user requirements, constraints, and context. It also involves identifying potential solutions and evaluating their feasibility and effectiveness.

Traffic congestion: The country's notorious traffic congestion can make it difficult and time-consuming for people to purchase daily essentials. This can lead to a lack of access to a wide range of products and services.

Limited parking spaces: Limited parking spaces can make it difficult for delivery personnel to access and park their vehicles. This can impact the efficiency of delivery services.

Remote or underserved areas: Some areas may be underserved or remote, making it difficult for delivery personnel to access these areas and deliver orders.

Small and medium-sized stores: Small and medium-sized stores often struggle to compete with larger stores, who have the resources and scale to offer lower prices and a wider range of products.

#### Motivation:

The motivation behind Hatley is to provide a fast, convenient, and transparent delivery service that connects users with a wide range of products and services. By allowing delivery personnel to compete for orders based on price and time, the platform can help to incentivize faster and more efficient delivery services, while also providing users with greater choice and flexibility. Additionally, by providing

job opportunities for young individuals and helping to transform the Egyptian market into a more competitive space, the platform can help to increase revenue and product availability for smaller stores, while also providing consumers with greater access to essential goods and services.

### 2.1.2 Scope of the Project:

**Delivery of daily essentials:** Hatley provides a comprehensive delivery solution for daily essentials, covering a wide range of products and services. This includes groceries, household items, electronics, and more. Hatley can help to simplify the delivery process and save users time and effort.

**Competition-based delivery service:** Hatley allows delivery personnel to compete for orders by either accepting the offer or presenting a competitive alternative in terms of time and cost. This encourages competition in the delivery, ultimately benefiting both users and delivery personnel.

**Real-time tracking:** Hatley enables users to track the arrival of their request, ensuring transparency and accountability. By providing real-time updates on the status of the delivery, users can have greater peace of mind and confidence in the delivery process.

**Job opportunities:** Hatley creates job opportunities for young individuals and transforms the Egyptian market into a competitive space, enhancing product quality and increasing revenue for some stores without the need for advertisements. By providing a platform for delivery personnel to compete for orders, Hatley can help to promote competition in the delivery.

**User-friendly platform:** Hatley provides a user-friendly and accessible platform for users and delivery personnel. By designing a simple and intuitive user interface, Hatley can help to ensure that users and delivery personnel can easily navigate the platform and access the information they need.

**Expansion:** Hatley expands the service to new markets and regions, both domestically and internationally. By expanding into new markets, Hatley can help to increase revenue, and provide job opportunities for delivery personnel in new regions.

Overall, the scope of Hatley includes providing a comprehensive delivery solution, promoting competition, ensuring transparency and accountability, creating job opportunities, promoting sustainable and environmentally-friendly practices,

ensuring safety and security, providing a user-friendly platform, and expanding to new markets and regions. By focusing on these areas, Hatley can help to meet the needs of its users and transform the Egyptian market into a more competitive space.

### 2.1.3 Target User Group:

Hatley targets the following user Group:

- Busy professionals: Hatley is designed for busy professionals who do not have the time to go to multiple stores to purchase their daily essentials. By providing a comprehensive delivery solution, Hatley can help these individuals save time and effort, allowing them to focus on other important tasks.
- Individuals with mobility issues: Hatley is also suitable for individuals with mobility issues who may have difficulty traveling to physical stores. By providing delivery services, Hatley can help these individuals access a wide range of products and services from the comfort of their own homes.
- Small and medium-sized businesses: Hatley can also be useful for small and medium-sized businesses that require regular deliveries of supplies and equipment. By allowing delivery personnel to compete for these orders, Hatley can help these businesses save time and money, ultimately increasing their revenue and competitiveness.
- Individuals in remote or underserved areas: Hatley can also be a valuable resource for individuals in remote or underserved areas who may have limited access to physical stores. By connecting users with delivery personnel who can purchase and deliver items from any nearby or distant store, Hatley can help to ensure that these individuals have access to a wide range of products and services.

Overall, the target user group for Hatley is broad and diverse, encompassing busy professionals, individuals with mobility issues, small and medium-sized businesses, environmentally-conscious consumers, and individuals in remote or underserved areas. By providing a comprehensive delivery solution that is fast, efficient, and user-friendly, Hatley can meet the needs of these users and help to transform the Egyptian market into a more competitive space.

## 2.2 System Requirement:

### 2.2.1 Project Sponsor:

The Egyptian Ministry of Higher Education in general and the Faculty of Computers and Information (Information System), Assiut University in particular finance the project and financial support and work to meet any needs of the project from the beginning of the project until its completion.

### 2.2.2 Business Need:

In studying the needs of the Egyptian market and the delivery services provided, we identified customer demand for a delivery service that assists them in obtaining their orders without the need to leave their premises, saving time, effort, transportation costs, and avoiding congestion. The problems with existing services include:

1. **Delayed Responses:** Delivery companies in the Egyptian market, specifically in Assiut Governorate, often take up to 45 minutes to confirm an order, followed by additional time for delivery.
2. **Price Increase:** Some companies exploit delivery personnel by taking a large percentage, up to 40%, of their earnings.
3. **Limited App Coverage:** Certain areas, especially restaurants, are not adequately covered by existing delivery apps.
4. **Busy Families:** Heads of households, occupied with work, require delivery for simple household items such as vegetables, fruits, school supplies, and other family needs.
5. **Uncertainty in Delivery:** Customers lack real-time information on the delivery time and location.
6. **Address Ambiguity:** Delivery personnel sometimes struggle to reach the exact address due to a lack of automatic location detection.

In response, we propose the development of a new application allowing users to order a variety of products by writing their request and setting a preliminary service price and time. Delivery personnel can then compete for the order by accepting the offer or providing a competitive alternative in terms of time and cost. Notifications are sent to the customer, allowing them to choose the best delivery option based on their preferences. This approach not only opens up job opportunities for the youth but also enhances market competition, improving product quality and increasing the revenue of local businesses without the need for extensive advertising.

#### Features:

For users, the application offers:

1. Ordering from anywhere with delivery to any location.
2. Order tracking through real-time location updates.
3. Rating and commenting on delivery services after order completion.

4. The freedom to choose the preferred delivery option based on cost and time.

### **Features for Delivery Personnel:**

#### **1. Smart Order Matching:**

- Delivery personnel can choose suitable orders using GPS features that display all nearby orders.
- Receive notifications when customers post new requests, allowing them to promptly respond.

#### **2. Competitive Bidding:**

- Delivery personnel can submit alternative offers to customers who have posted orders, competing based on time and cost.
- Enhances the chances of securing orders and encourages efficient service.

#### **3. Efficient Address Navigation:**

- Simplified access to precise delivery locations through automatic or manual customer location detection.
- Facilitates accurate and timely deliveries, improving overall customer satisfaction.

#### **4. Increased Earnings Percentage:**

- Application allows a higher percentage of the delivery cost to go to the delivery personnel, motivating them to use the app rather than traditional services with high commission rates.
- Encourages fair compensation for the delivery efforts.

#### **5. Customer Rating and Feedback:**

- Delivery personnel can be rated by customers, fostering a trustworthy and transparent system.
- Customers can leave comments, building a reputation for reliable and efficient delivery personnel.

#### **6. Flexible Work Zone:**

- Delivery personnel can set their preferred work area and adjust it when transitioning between different locations or governorates.
- Provides flexibility and convenience for delivery personnel based on their current location.

### **2.2.3 Business Value:**

In our current era, digital services have become indispensable. As the individual's responsibilities increase and the speed of life increases, we thought of providing a home delivery service for all services that an individual may need, in order to save time and effort while ensuring quality service and good treatment. We conducted a survey to see the extent of the project idea's resonance and quality among the audience, and it was clear that the community needed such a service, as many survey participants explained that long distance and crowding are among the main factors that increase their desire to

provide such a service. In addition, the quality of the service and its cost are among the main factors that encourage their use of it and this is what our project aims to achieve and its greatest challenge, as it works to provide the user's order in the fastest time at the lowest cost and best quality also it provides knowledge of the status of your order and tracking it. In addition, the lack of similar competitors in such a service also makes the idea more desirable. Therefore, it is very expected that there will be a strong demand and response from the community for the service provided

## 2.3 Fesability Study:

### 2.3.1 Technical Feasibility:

#### 1. familiarity with web application is medium risk:

- The team has some good experience with the web application design and functions.
- The team has low experience with the frameworks and how real application work in the real world.
- there is some new features is still under studying by the team.

#### 2. familiarity with the delivery System is medium risk:

- some of the team has good experience of how the mechanism of the delivery application work and how user and the delivery will contact with each other.
- some of team has good user experience to make a good design and UX to make the app easy to use and fast and have good UI design.

#### 3. size of project is considered low risk:

- in first version of the application; the number of team is good to do this app
- the project time is long so there is low risk in the deadline of the app
- Up till now there is no app in competitive with us.

#### 4. compatibility with the delivery app is good:

- because it's a new system so we have to get new server and it must be fast and strong to handle the number of users of the app fast.
- in this version the marketing will be good and will scale by time to deliver the application every were

### 2.3.2 Economic Feasibility:

	2023	2024	2025	2026	Total
<b>Benefits</b>					
<b>Number of views to the site</b>	5,000	15,000	30,000	30,000	50,000
<b>Percentage from each order</b>	0	120	1,825	1,945	
<b>Investments</b>	0	50,000	150,000	200,000	
<b>Total Benefits</b>	5,000	65,120	181,825	251,945	
<b>PV Total Benefits</b>	4,319	56,253	157,067	217,640	
<b>Development costs</b>					
<b>Domain</b>	500	0	0	0	500
<b>Virtual server</b>	20,000	0	0	0	20,000
<b>Total development costs</b>	<b>20,500</b>	0	0	0	<b>20,500</b>
<b>Operational costs</b>					
<b>Private Server</b>		20,000	20,000	20,000	60,000
<b>Office Space Workplace</b>			24,000	24,000	48,000
<b>Marketing</b>		20,000	30,000	50,000	100,000
<b>Total operational costs</b>	<b>40,000</b>	<b>74,000</b>	<b>94,000</b>	<b>208,000</b>	
<b>Total costs</b>	<b>20,500</b>	<b>40,000</b>	<b>74,000</b>	<b>94,000</b>	<b>228,500</b>
<b>Present value total costs</b>	17,709	34,554	63,924	81,201	197,387
<b>Total benefits – Total costs</b>	-17,709	-30,234	-7,671	75,867	20,253
<b>Cumulative Net cash Flow</b>	-17,709	-47,943	-55,614	20,253	
<b>Return on investment</b>	10.26%				
<b>Break-even point</b>	2.733 Years				

### 2.3.3 Organizational Feasibility:

- **Champion:** The goal of the Hatley Project is to provide a distinctive delivery service that provides job opportunities for young people away from delivery companies exploiting them, facilitating the service for users so that it saves them the time and cost of ordering, the time required to complete the project is one year.
- **organizational Management:** we will support this project with budget **228,500EGP** (Editing in progress).
- **system users:** their role is to make important decisions and they are asked for feedback to improve system.

## 2.4 The Survey:

**Survey:** We have conducted a survey to gauge the market's response to the idea of a delivery application. This survey has been instrumental in understanding people's needs, existing problems, customer requirements, satisfaction levels, and the anticipated success of the project. The survey results can be summarized as follows:

A total of 58 individuals responded, with 55.2% being female and 44.8% male.

Reasons for using delivery services included distance, traffic, work commitments, delivery of various items, and quantities from different locations.

The satisfaction rate regarding pricing received an average score of 5 out of 10, with 27.6% giving this rating.

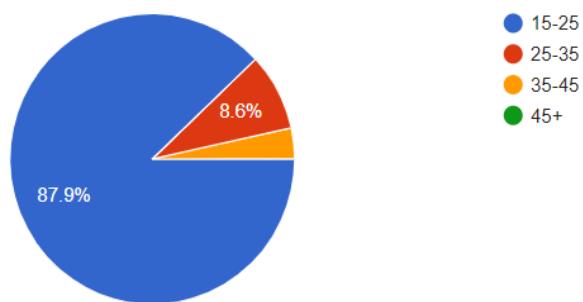
Customers expressed an 86% satisfaction rate with the concept of a delivery application. Additional feedback included suggestions for reducing delivery fees, implementing a rating system for delivery personnel, managing order confusion, offering a reliable GPS tracking feature, addressing delivery time concerns, and ensuring the application is not tied to specific products, allowing flexibility in ordering.

This valuable feedback will guide our project development, addressing specific concerns and enhancing the overall customer experience.

 نسخ

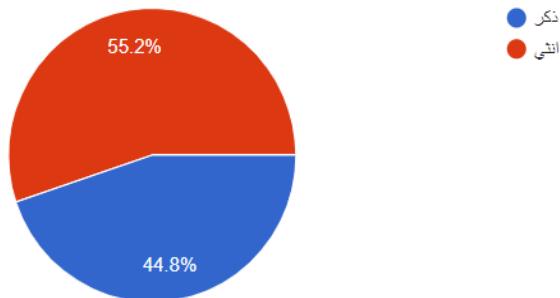
السن

رداً 58


 نسخ

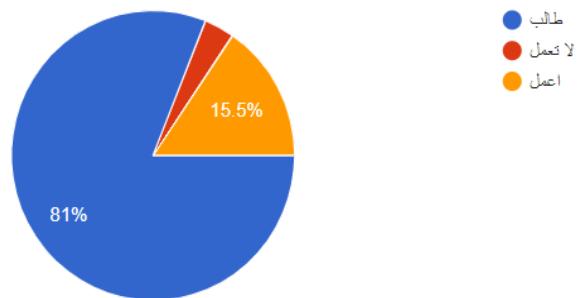
النوع

رداً 58


 نسخ

العمل

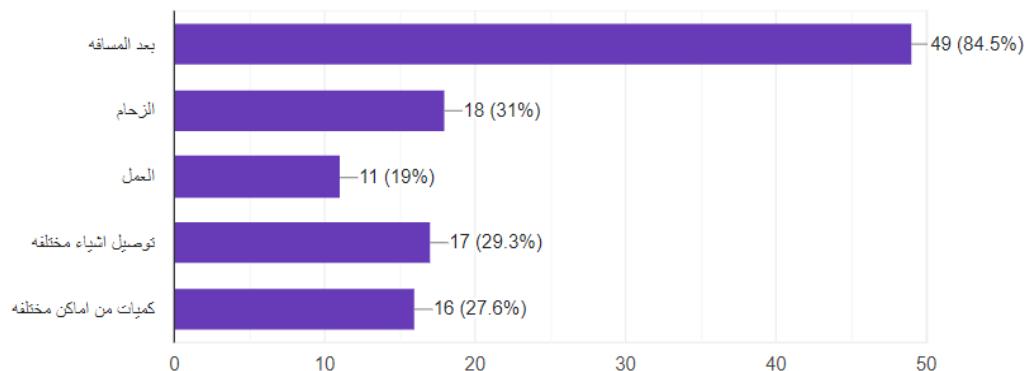
رداً 58



[نسخ](#)

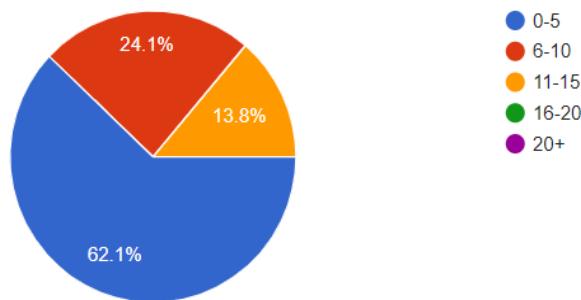
ما الذي يدفعك لاستخدام خدمة الدليفري

رداً 58

[نسخ](#)

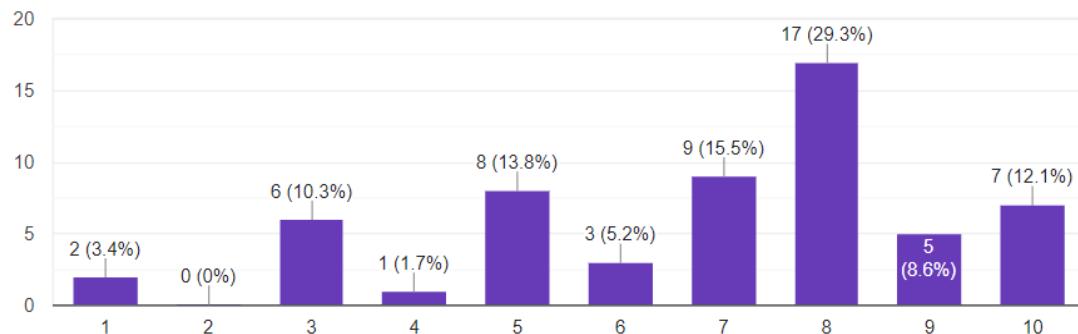
كم مرّه تستخدم خدمة التوصيل شهرياً

رداً 58

[نسخ](#)

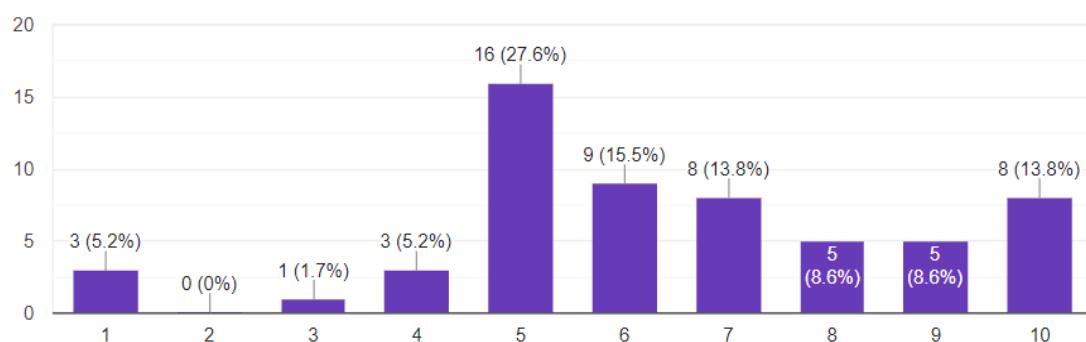
نسبة رضاك عن وقت التوصيل

رداً 58



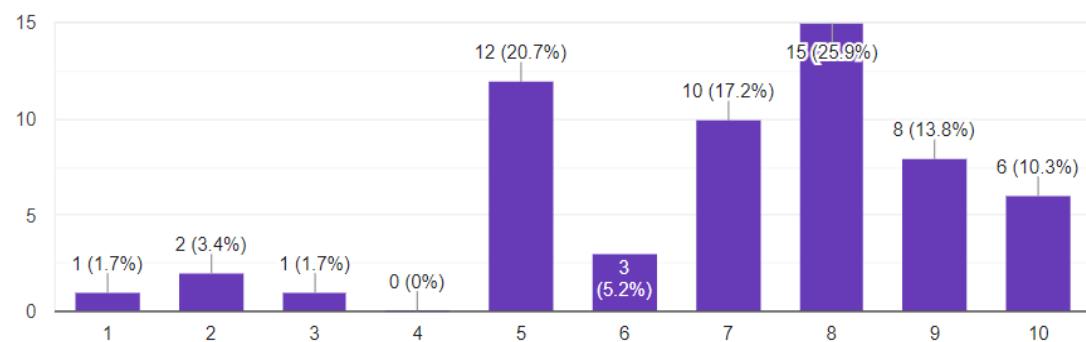
نسبة رضاك عن سعر التوصيل

رداً 58



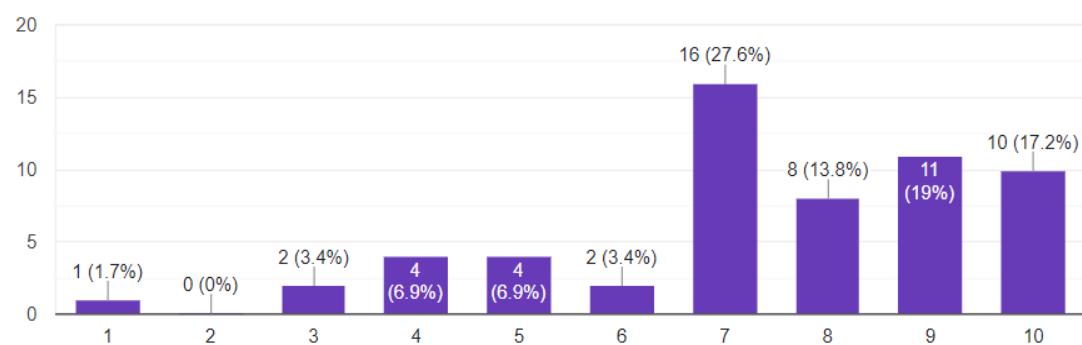
نسبة رضاك عن جودة الخدمة

رداً 58



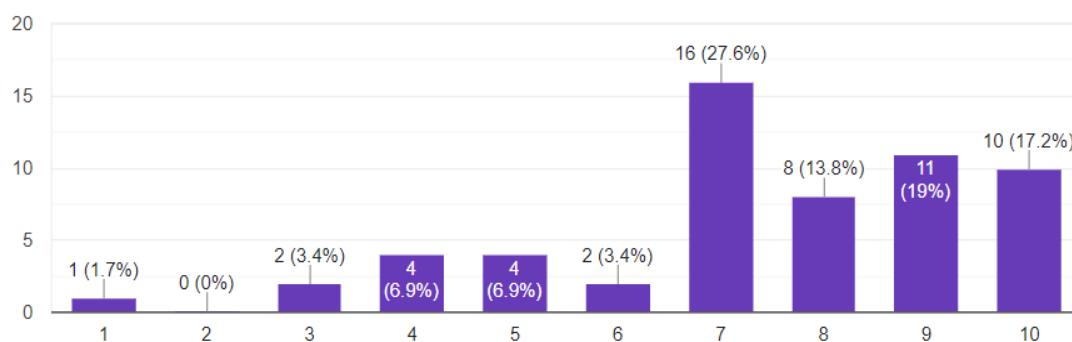
نسبة رضاك عن جودة الشحن

رداً 58



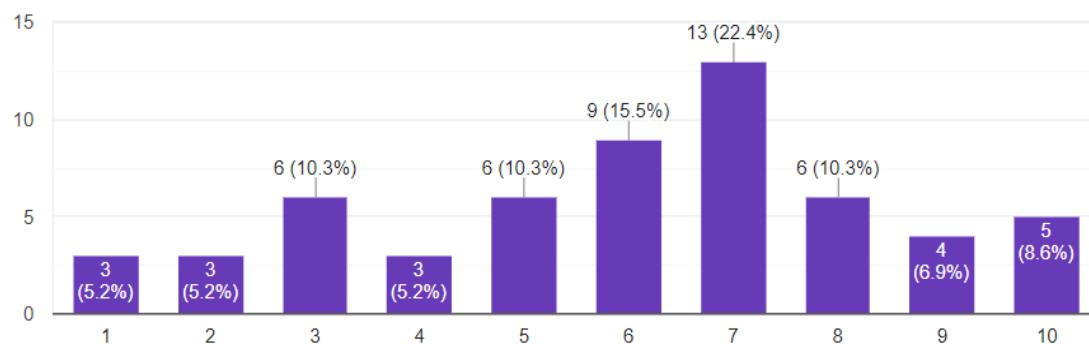
نسبة رضاك عن جودة الشحن

رداً 58



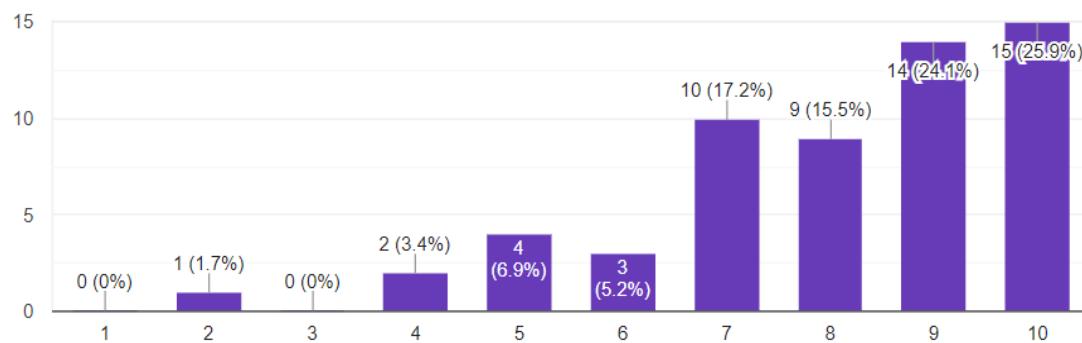
نسبة رضاك عن شركات الدلivery التي توفرها منطقتك

رداً 58



نسبة رضاك عن امانة الدلivery

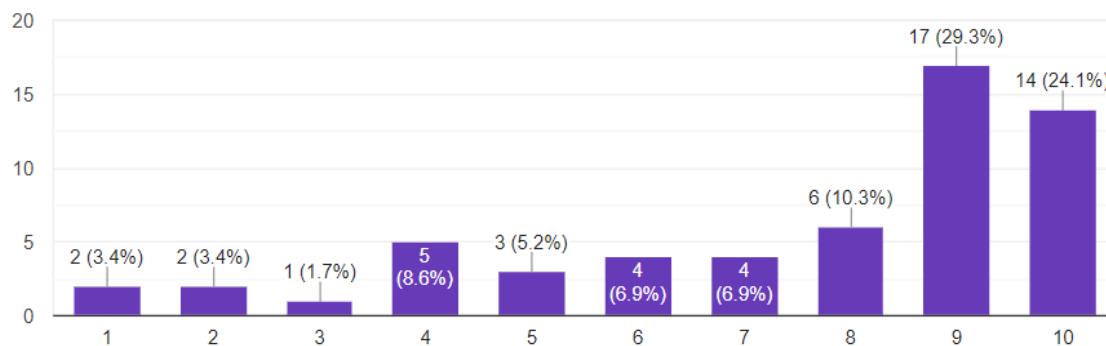
رداً 58



نسخ

نسبة رضاك عن توصيل

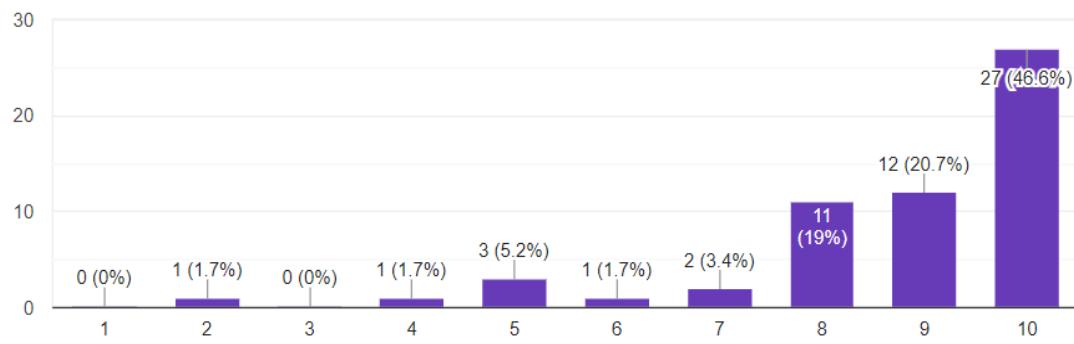
رداً 58



نسخ

فكرة وجود التطبيق لتقديم خدمة الدليلي بشكل عام

رداً 58



مميزات وعيوب الخدمة او مقتراحات لتطويرها (من تعاملاتك السابقة)

رداً 22

المميزات : توفير الخدمات في اقرب وقت وتغطية المسافات  
العيوب : عدم توافرها في كافة المناطق

تقليل سعر الدليلي

التعامل بلطف

الوقت طويلاً ياريت يقل شويه (وقت التوصيل)

اقتراح وجود دليلي بذات

شایطة الفكرة او يريدي موجوده وفيه شركات كبار كذا ف اي اللي يخليني اتعامل مع شركة جديدة ومশ مضسونه؟ لازم تكون بتقدم خدمة زياده + شركات التوصيل بتكون اسعارها اعلى من الدليلي اللي بيشتغل لوحده لحسابه ودا حيب بيخلني معظم الناس تتجه لتعامل مع ناس عاديين + شركات التوصيل  
متش بتنهتم ولا بالمواقع ولا بالتنفيذ ع عكس الدليلي العادي

الميزة بتوفير عليا المسافة  
العيوب اهها احياناً بيحصل لغبطة ف الاوردرات وبيجيلى الاورد الخطأ

ميزة انه بيوصل منتجات مثلاً مثل موتوقة في المنطقة او صعب الاقبها وكمان بيوصل اكل او طلب محتاجاه في وقت صعب انزل فيه عيوب انه بيكون في شركات مثل موتوقة بدفعها عربون ومبنيوصلش المنتج او مثلاً منتج مبيكونش زى اللي عايزاه او مبيكونش في مرتجعات المقترن اكمل تندلو من العيوب وتعاملو مع شركات موتوقة و معروفة في السوق

ان يبقى فيه طريقة اعرف الدليلي وصل فين و فاضله قد ايه و يوصل

ربط الموقع لمعرفة متى تحرك الطلب وain ومعرفة موعد الوصول بالضبط

الاهتمام بسرعة التوصيل

عرض الخضار واسعاره والماركت واسعاره وانته تتعامل مع الناس بتاعتاك وانا اختار من مكان واحد بس وهو اللي يخلص

لو فيه قسائم بتبقا فيه اربروز كبير في تشغيل القصائم دي و التحفصات بتبقا حاجه مستفزة لكن لو اتعدلتو بتبقا احسن طبعا

إن يفهم العنوان بسهولة مثل كل شوية يتصل بالعميل يعني ممكن هو يتتابع gbs

الابليكتشن فكره حلوه جداً بدل الرن الكبير ولكن الأفضل سرعة التوصيل تتحسن ♥♥

المميزات

١. توفير الوقت

٢. حماية من زاحم المواصلات

العيوب

بتعلمك الكسل

المميزات: توفير الوقت و سهولة الطلب العيوب: عدم كفاءة للموظفين أحياناً و التأخير

الدليلي يكون عنده امانه والبائع يكون صادق شركات الشحن تشغل ناس عندها ضمير توصل الحاجه زي ما هيا والبائع بصور المنتج على الحقيقه من يكبره

الالتزام بالمواعيد وجوده فكه غير ذلك جيد جداً

عدم التأخير الدائم في الطلب

## 2.5 Definition of Requirements:

### 2.5.1 Functional Requirements:

✓ Functional requirements	
User Registration and Authentication	<ul style="list-style-type: none"> <li>✓ Users should be able to register, create accounts, and log in securely.</li> <li>✓ Implement password recovery and account verification processes.</li> </ul>

<b>Order Creation</b>	<ul style="list-style-type: none"> <li>✓ Allow users to create new orders.</li> <li>✓ Create a simple request form with fields for entering basic information such as the name of the desired item, the quantity needed, available budget, specific specifications for the requested item, and any other relevant information.</li> <li>✓ We consider additional options, such as attaching images or files to provide more information about the requested item.</li> </ul>
<b>Location and Address Selection</b>	<ul style="list-style-type: none"> <li>✓ Provide an interface for users to input or select their delivery location and address.</li> <li>✓ Include options for saving multiple delivery addresses.</li> </ul>
<b>Order Confirmation and Payment</b>	<ul style="list-style-type: none"> <li>✓ Confirm the order details, including the total cost, and prompt users for payment.</li> <li>✓ Support various payment methods, such as credit cards, digital wallets, or cash on delivery.</li> <li>✓ The user needs a “Submit Request” button to send the information they have entered.</li> </ul>
<b>Display Requests and Interaction</b>	<ul style="list-style-type: none"> <li>✓ Display user requests to service providers who can respond to them.</li> <li>✓ Providing an interface for service providers to interact with requests by presenting their offers.</li> <li>✓ Enable service providers to submit their offers and specify prices and terms.</li> </ul>

<b>User Selection of Preferred Offer</b>	<ul style="list-style-type: none"> <li>✓ Grant users the ability to choose from among the service providers based on the offers and terms presented.</li> <li>✓ Add a "Select" or "Accept" button for users to choose from among the offers.</li> <li>✓ Update the status of the request and notify the user of their selection.</li> </ul>
<b>Request Processing and Assignment</b>	<ul style="list-style-type: none"> <li>✓ Implement a system to process requests and assign them to selected service providers.</li> <li>✓ Provide options for service providers to interact with posts by using "Accept" or "Offer" buttons.</li> <li>✓ Develop a dashboard for managing offers submitted by service providers, including specifying prices and terms.</li> </ul>
<b>Notifications and Alerts</b>	<ul style="list-style-type: none"> <li>✓ Implement a real-time notification system to inform users of offer details and request status.</li> <li>✓ Send notifications to both users and service providers regarding request status.</li> </ul>
<b>Request Tracking and Delivery</b>	<ul style="list-style-type: none"> <li>✓ Allow users to track the status of their requests and receive updates on the delivery process.</li> <li>✓ Send automatic updates on the provider's location and estimated arrival time.</li> </ul>
<b>User Ratings and Comments</b>	<ul style="list-style-type: none"> <li>✓ Enable users to provide ratings and comments after the service or delivery is completed.</li> <li>✓ Display the average ratings for users.</li> </ul>
<b>Customer Support</b>	<ul style="list-style-type: none"> <li>✓ Develop a customer support system that allows users and service providers to communicate and resolve issues.</li> </ul>

<b>Admin Dashboard</b>	<ul style="list-style-type: none"> <li>✓ Create an administrative control panel that allows administrators to manage posts, offers, and monitor operations.</li> </ul>
------------------------	--

## 2.5.2 Non-Functional Requirements:

<b>✓ Non-functional requirements</b>	
<b>Security</b>	<ul style="list-style-type: none"> <li>✓ The website must implement high levels of security to protect sensitive information and prevent security breaches and protect user data from unauthorized access.</li> <li>✓ User authentication and identity verification mechanisms must be secure.</li> </ul>
<b>performance</b>	<ul style="list-style-type: none"> <li>✓ The website should load quickly and provide good performance to ensure a smooth user experience, even with a large number of users.</li> <li>✓ Performance can be improved by reducing loading times, optimizing images, and utilizing caching.</li> <li>✓ The system supports a large number of concurrent users at all other times.</li> </ul>
<b>Availability</b>	<ul style="list-style-type: none"> <li>✓ The website must be available 24/7/365 and maintain a high level of uptime.</li> <li>✓ Contingency plans and disaster recovery strategies should be in place in case of website failure.</li> </ul>
<b>Cross-Browser Compatibility</b>	<ul style="list-style-type: none"> <li>✓ The website should be designed to work consistently across different common web browsers (Chrome, Firefox, Safari, etc.).</li> </ul>
<b>Responsive Design</b>	<ul style="list-style-type: none"> <li>✓ The website should be responsive, adapting to various devices and screen sizes.</li> </ul>

<b>Ease of Maintenance</b>	<ul style="list-style-type: none"> <li>✓ Code should be well-organized and maintainable to facilitate future maintenance and development.</li> </ul>
<b>Scalability</b>	<ul style="list-style-type: none"> <li>✓ The website must be continuously scalable to accommodate the increase in the number of users and the volume of data.</li> </ul>
<b>Database Performance</b>	<ul style="list-style-type: none"> <li>✓ The database should be well-structured and optimized for fast response times.</li> </ul>
<b>Documentation</b>	<ul style="list-style-type: none"> <li>✓ The project and codebase should be well-documented to ease future handling by development teams.</li> </ul>
<b>Data Security</b>	<ul style="list-style-type: none"> <li>✓ Personal data and sensitive information must be protected with strict information security measures.</li> </ul>
<b>Continuous Improvement</b>	<ul style="list-style-type: none"> <li>✓ There should be a plan for ongoing management and implementation of improvements to the website.</li> </ul>
<b>Compliance with Standards</b>	<ul style="list-style-type: none"> <li>✓ The website must comply with international web standards such as W3C and WCAG for accessibility.</li> </ul>
<b>Usability</b>	<ul style="list-style-type: none"> <li>✓ The system must be easy to use and understand.</li> </ul>

## 2.6 Use Cases:

### 2.6.1 Register Use Case:

Use Case Name: <b>Register</b>		ID: UC_1	Level: Primary
Actor	New User or New Delivery		
Description	New User: Wants to create an account to use the delivery app for ordering goods. New Delivery Person: Wants to create an account to become a delivery service provider		
Preconditions	<ul style="list-style-type: none"> <li>• The user has Navigated to the register page.</li> <li>• The user or delivery person has not registered an account previously.</li> <li>• The website is accessible.</li> </ul>		
Main Success Scenario	<ol style="list-style-type: none"> <li>1. The new user opens the website and selects the "Register" option.</li> <li>2. The system presents a registration form with fields for personal information, such as name, email, phone number, a password, Photo of ID card and Photo of the user or delivery with ID card.</li> <li>3. The new user or delivery person enters their information into the form.</li> <li>4. The new user or delivery person selects the type of account they want to create (either "User" or "Delivery Person").</li> <li>5. The system validates the entered information:               <ul style="list-style-type: none"> <li>• If the information is valid, the system generates a unique ID for the user or delivery person.</li> <li>• If any information is missing or invalid, the system displays an error message and allows the user to correct it.</li> </ul> </li> <li>6. The new user or delivery person confirms their registration by submitting the form.</li> <li>7. The system stores the user's or delivery person's information, including their unique ID, in the database.</li> </ol>		
Postconditions	<ul style="list-style-type: none"> <li>• If successful, a new user or delivery person account is created with a unique identifier (ID).</li> <li>• The new user or delivery person can log in with their newly created account.</li> <li>• The user's or delivery person's information is stored securely in the system's database</li> </ul>		
Exceptions	<ul style="list-style-type: none"> <li>• <b>Existing User/Delivery Person (Step 1, Alternative Flow):</b> If a user or delivery person with the same email or phone number already exists, the system should prompt the user to log in or initiate a password reset if they forgot their password.</li> <li>• <b>Validation Errors (Step 5, Alternative Flow):</b> If the system encounters errors during the validation of the registration information (e.g., invalid email format, password requirements not met), it should display error messages and allow the user to correct the information.</li> </ul>		

## 2.6.2 Login Use Case:

Use Case name: Login		ID: UC_2	Priority:
<b>Actor</b>	<ul style="list-style-type: none"> <li>• Customer</li> <li>• Delivery Driver</li> <li>• System</li> </ul>		
<b>Description</b>	This use case describes the login mechanism for the user or the delivery or the admin		
<b>Preconditions</b>	The customer must be registered.		
<b>Main Success Scenario (Main Flow)</b>	<p>1-The user navigates to the login page</p> <p>2-the system make the user choose between delivery or user or admin</p> <p>3-The system presents the user with a login form, including fields for username/email and password</p> <p>4-the user enter their valid username and password</p> <p>5- the user submit the form by clicking on Login button</p> <ul style="list-style-type: none"> <li>• If the credentials are valid, the system grants access to the user and proceeds to the main application.</li> <li>• If the credentials are invalid, the system displays an error message (e.g., "Invalid username or password") and returns to the login form</li> </ul>		
<b>Postconditions</b>	The system will navigate the user automatic to the home page or the last page the user was in it and the page ask him to login before complete		
<b>Alternative Flow</b>	the user make login with account blocked by system. the system will display a message that this account is blocked for bad behavior		
<b>Exceptions</b>	<p>1-the user login with user account in the delivery account or the opposites</p> <p>2- the user forget the password</p> <p>3-invalid data for many times must be locked for five minutes or more cooldown</p>		

### 2.6.3 Display requests and Interaction Use Case:

<b>Use Case Name: Display requests and Interaction</b>		<b>ID:UC_3</b>	<b>Priority:</b>
<b>Description</b>	This scenario enables service providers (delivery men) to accept or interact with orders, they also can submit their offers, price and time to customer.		
<b>Actor</b>	<ul style="list-style-type: none"> <li>○ Service provider (Delivery driver)</li> </ul>		
<b>Precondition</b>	<ul style="list-style-type: none"> <li>✓ Delivery should have active account and logged in to the site</li> <li>✓ There are active orders posted by users (customer) not be done yet.</li> </ul>		
<b>Normal Course</b>	<ol style="list-style-type: none"> <li>1- Delivery man open home page.</li> <li>2- From his dashboard, delivery man can view a list of available requests and its details.</li> <li>3- Delivery man can select a specific delivery request.</li> <li>4- Delivery can accept the request by pressing on accept button.</li> </ol>		
<b>Alternative Course</b>	<ol style="list-style-type: none"> <li>1- Delivery man open home page or notification.</li> <li>2- Choose a specific order.</li> <li>3- View its details.</li> <li>4- Press on “another offer”.</li> <li>5- New page appear that contain time and price fields.</li> <li>6- Delivery fills these fields.</li> <li>7- Press “send” or “confirm” button.</li> </ol>		
<b>Postcondition</b>	<ul style="list-style-type: none"> <li>• Notification contains all offers will be sent to the customer that contain delivery name, rate, picture, price and time.</li> <li>• Customer reviews the offers and compare them each offer has two button accept and reject.</li> <li>• The customer has the authority to accept any offer then when he accept: <ol style="list-style-type: none"> <li>1- A notification must be sent to the service provider that the customer accept his offer.</li> <li>2- The order is being converted or marked as unavailable now.</li> </ol> </li> <li>• The customer and the chosen delivery must be linked together.</li> <li>• If none of the service providers offer for an order for long time, the user can choose to resubmit the order or explore other options</li> </ul>		
<b>Exception</b>	<ul style="list-style-type: none"> <li>➤ When he offers another by filling time and price and clicks Confirm or Send, it has be agreed by someone else or converted to not available now.</li> </ul>		

## 2.6.4 Create Order Use Case:

<b>Use Case Name: Create Order</b>		<b>ID: UC_4</b>	<b>Priority: High</b>
<b>Description</b>	This use case describes how the user will make an order.		
<b>Actor</b>	User		
<b>Precondition</b>	<ul style="list-style-type: none"> <li>✓ The user opens the system.</li> <li>✓ The user must log in.</li> </ul>		
<b>Normal Course</b>	<ol style="list-style-type: none"> <li>1- The user clicks "make an order" button.</li> <li>2- The system opens the form to the user to fill it out.</li> <li>3- The user writes the name of items he\she needs.</li> <li>4- The user writes the quantity he\she needs.</li> <li>5- The user specifies the available budget he\she can pay.</li> <li>6- The user can attach a photo or file if he\she wants(optional).</li> <li>7- The user clicks "Done" button.</li> </ol>		
<b>Postcondition</b>	The system will redirect the user to location selection interface. The system will redirect the user to location selection interface.		
<b>Exception</b>	<ul style="list-style-type: none"> <li>➤ If the user wrote a strange name for the item he\she needs, an error message will appear to tell him\her to explain what he\ she needs.</li> <li>➤ If the user wrote a MINS quantity (or any not reasonable amount) for the item he\she needs, an error message will appear.</li> <li>➤ If the budget were zero, an error message will appear.</li> </ul>		

## 2.6.5 Accept offer Use Case:

<b>Use Case Name: Accept offer</b>		<b>ID: UC_5</b>	<b>Priority: High</b>
<b>Description</b>	This use case describes how the user will choose from the offers he\she has received.		
<b>Actor</b>	<ul style="list-style-type: none"> <li>○ User</li> </ul>		
<b>Precondition</b>	<ul style="list-style-type: none"> <li>✓ The user opens the system.</li> <li>✓ The user must log in.</li> <li>✓ The user makes an order.</li> <li>✓ The user determines his\her location.</li> <li>✓ The user determines payment way.</li> </ul> <p>The user receives offers.</p>		
<b>Normal Course</b>	<ol style="list-style-type: none"> <li>1. The user reviews the offers he\she has received.</li> <li>2. The user clicks "Accept" button of the offer he\she preferred.</li> </ol>		
<b>Postcondition</b>	<ol style="list-style-type: none"> <li>1. A success message will appear for the user.</li> <li>2. The chat will be opened.</li> </ol>		
<b>Exception</b>	If the user did not find a suitable offer, he\she can cancel the order.		

## 2.6.6 Introduce offer Use Case:

<b>Use Case Name': Introduce offer</b>		<b>ID: UC_6</b>	<b>Priority: High</b>
<b>Description</b>	This use case describes how the service provider will provide an offer.		
<b>Actor</b>	<ul style="list-style-type: none"> <li>○ Service provider (Delivery driver)</li> </ul>		
<b>Precondition</b>	<ul style="list-style-type: none"> <li>✓ The service provider opens the system.</li> <li>✓ The service provider must log in.</li> </ul>		
<b>Normal Course</b>	<ol style="list-style-type: none"> <li>1. The service provider receives a notification of the order</li> <li>2. The service provider reviews the orders.</li> <li>3. The service provider provides an offer for the order he\she wants.</li> <li>4. The service provider determines the price and duration for the order.</li> </ol>		
<b>Postcondition</b>	A notification will be received for the service provider if the user accepted his\ her offer.		
<b>Exception</b>	The order has just started.		

## 2.6.7 Notification and Alert Use Case:

USE CASE NAME: Notification and Alert		ID: UC_7 Priority: medium
<b>Actor</b>	System	
<b>Description</b>	This use case describes the notification mechanism of how to notify the user or the delivery man of the request status for the user and new order for delivery.	
<b>Preconditions</b>		<ul style="list-style-type: none"> <li>✓ The customer or delivery must be registered and logged into their account.</li> </ul>
<b>Main Success Scenario (Main Flow)</b>		<ol style="list-style-type: none"> <li>1. For delivery:           <ul style="list-style-type: none"> <li>• If any user post an order that match with the delivery location, the system send notification to the delivery that there is new post and brief of the post .</li> <li>• If the delivery sends an offer to a user and the user accept or reject the offer, the system should send notification with the answer of the use.</li> <li>• The messages between the user and delivery have also notification with every message</li> </ul> </li> <li>2. For User:           <ul style="list-style-type: none"> <li>• If the user post order and start to receive offers from deliveries each one have single notification</li> <li>• The status of the order has notification the describe to the user where its order now</li> <li>• The messages between the user and the delivery also have notification</li> </ul> </li> </ol>
<b>Postconditions</b>		Any of them if clicked must open the website on the page that pushed the notification
<b>Exceptions</b>		No Exception

## 2.6.8 Customer Support Use Case:

Use Case Name: Customer Support		ID:UC_8	Priority:
<b>Description</b>	This scenario enables users and service providers to communicate and resolve problems to the delivery service.		
<b>Actor</b>	<ul style="list-style-type: none"> <li>○ User (Customer)</li> <li>○ Service provider (Delivery driver)</li> </ul> <p>Customer support representative.</p>		
<b>Precondition</b>	<ul style="list-style-type: none"> <li>✓ The user or service provider is logged into their own accounts.</li> <li>✓ The user or service provider has encountered a problem or has a question related to system services.</li> </ul>		
<b>Normal Course</b>	<ul style="list-style-type: none"> <li>• Press on “Support System” button.</li> <li>• Choose type of problem with delivery men or with application.             <ul style="list-style-type: none"> <li>1- If he choose with delivery men (customer) :                     <ul style="list-style-type: none"> <li>✓ All the delivery men (customers) that the customer (delivery man) deal with must be appear as a list on a new page.</li> <li>✓ Choose one of them and press on Reporting button</li> <li>✓ New page should be opened must have several choices and empty box to write details of problem.</li> <li>✓ Choose type of problem, write details and send some proofs.</li> <li>✓ Press on “confirm” button.</li> </ul> </li> <li>2- If he chooses with application:                     <ul style="list-style-type: none"> <li>✓ New page should be opened must have list of some expected problems and its solutions take in consideration he might choose another problem in an empty box.</li> </ul> </li> </ul> </li> </ul>		
<b>Alternative Course</b>	<ol style="list-style-type: none"> <li>1- User clicked on “last deals” button.</li> <li>2- Choose the order that has a problem.</li> <li>3- Open the account of delivery man (customer) who he wants to complain him.</li> <li>4- Press on “Reporting” button.</li> <li>5- New page should be opened must have several choices and empty box to write details of problem.</li> <li>6- Choose type of problem, write details and send proof if exist.</li> <li>7- Press on “confirm” button.</li> </ol>		
<b>Postcondition</b>	<ol style="list-style-type: none"> <li>1- If he chooses specific question, new page appears with answer this question and solve the problem.</li> <li>2- If he writes a report, it should be successfully stored in database related with the data of user.</li> <li>3- The system send notification to Admin with the report.</li> <li>4- An admin checks the report.</li> <li>5- An admin replay to the user and when he solved this problem he marked as “resolve or done”</li> </ol>		
<b>Exception</b>	<ul style="list-style-type: none"> <li>○ No place in server to store report.</li> <li>○ The delivery has just deleted his account.</li> </ul>		

## 2.6.9 Driver Ratings for Customers Use Case:

USE CASE NAME: Driver Ratings for Customers		ID: UC_9	Priority: Moderate
<b>Actor</b>	Delivery Driver		
<b>Description</b>	This Use Case outlines the process of allowing delivery drivers to rate customers based on their interactions and experiences.		
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>✓ The customer has successfully completed a delivery request.</li> <li>✓ The delivery driver has successfully completed a delivery.</li> </ul>		
<b>Main Success Scenario (Main Flow)</b>	<ol style="list-style-type: none"> <li>1. Driver completes the delivery: <ul style="list-style-type: none"> <li>✓ The delivery driver successfully completes the delivery to the customer's location.</li> </ul> </li> <li>2. Driver rates the customer: <ul style="list-style-type: none"> <li>✓ After completing the delivery, the delivery driver has the option to rate the customer based on their interactions and behavior during the delivery.</li> <li>✓ The driver provides a rating for the customer (e.g., on a scale of 1 to 5 stars) reflecting their experience.</li> </ul> </li> <li>3. Driver leaves optional comments: <ul style="list-style-type: none"> <li>✓ The delivery driver can also provide optional comments or feedback regarding their experience with the customer, including any specific positive or negative aspects.</li> </ul> </li> <li>4. System records the driver's rating and comments: <ul style="list-style-type: none"> <li>✓ The system records the rating and comments provided by the delivery driver, associating them with the specific delivery and customer.</li> </ul> </li> </ol>		
<b>Postconditions</b>	<ul style="list-style-type: none"> <li>✓ The delivery driver's rating and comments about the customer are stored in the system for reference.</li> <li>✓ The customer's profile may include the average rating given by drivers.</li> </ul>		
<b>Alternative Flow</b>	<p>Driver chooses not to rate or comment:</p> <ul style="list-style-type: none"> <li>• If the delivery driver decides not to provide a rating or comments about the customer.</li> <li>• The system records the delivery as completed without specific feedback.</li> </ul>		
<b>Exceptions</b>	<ol style="list-style-type: none"> <li>1. Inappropriate Comments: <ul style="list-style-type: none"> <li>• If the delivery driver leaves inappropriate or offensive comments about the customer.</li> <li>• The system should have a mechanism in place to flag and review such comments. Inappropriate comments may be removed, and appropriate actions may be taken.</li> </ul> </li> <li>2. Pattern of Inappropriate Ratings:</li> </ol>		

- If a delivery driver consistently provides low ratings and negative comments about customers without justifiable reasons.
- The system may trigger a review of the driver's behavior and may take corrective actions if necessary.

## 2.6.10 Request Tracking and Delivery Use Case:

USE CASE NAME: Request Tracking and Delivery		ID: UC_10	Priority: High
<b>Actor</b>	<ul style="list-style-type: none"> <li>• Customer</li> <li>• Delivery Driver</li> <li>• System</li> </ul>		
<b>Description</b>	This use case describes the process of tracking and delivering customer orders, it focuses on the interactions between customers, delivery drivers and the system to ensure efficient and transparent order tracking and successful delivery.		
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>✓ The customer must be registered and logged into their account.</li> <li>✓ Orders must be recorded in the system with the required information.</li> </ul>		
<b>Main Success Scenario (Main Flow)</b>	<ol style="list-style-type: none"> <li>Customer places an order: <ul style="list-style-type: none"> <li>✓ The customer logs into their account.</li> <li>✓ The customer specifies the item he/she want to order.</li> <li>✓ The customer creates a delivery request post, specifying the item for delivery and delivery details.</li> <li>✓ The system records the delivery request and sets the initial status to "Request Submitted."</li> </ul> </li> <li>Delivery drivers provide offers: <ul style="list-style-type: none"> <li>✓ Delivery drivers who are available to take delivery requests view the posted requests.</li> <li>✓ Each interested driver submits an offer, including the delivery fee and estimated delivery time.</li> <li>✓ The customer receives notifications of new delivery offers.</li> </ul> </li> <li>Customer selects a delivery offer: <ul style="list-style-type: none"> <li>✓ The customer reviews the delivery offers provided by drivers.</li> <li>✓ The customer selects a preferred delivery offer based on the driver's fee, estimated delivery time and the service provider's rating.</li> <li>✓ The system assigns the chosen delivery driver to the request, and the status is updated to "Driver Assigned."</li> </ul> </li> <li>Delivery in progress: <ul style="list-style-type: none"> <li>✓ The assigned delivery driver follows the navigation provided by the system to reach the pickup location if it exists.</li> <li>✓ The driver collects the specified item for delivery.</li> <li>✓ The system continuously tracks the driver's location in real-time.</li> </ul> </li> </ol>		

	<ul style="list-style-type: none"> <li>✓ The customer can track the delivery progress in real-time through their account.</li> </ul> <p>9. Delivery driver completes the delivery:</p> <ul style="list-style-type: none"> <li>✓ The driver arrives at the customer's destination and delivers the specified item.</li> <li>✓ The system updates the request status to "Delivered."</li> <li>✓ The customer receives an instant notification confirming the successful delivery of the requested item.</li> <li>✓ If the customer had chosen to pay in cash, payment is collected from them after the delivery.</li> </ul>
<b>Postconditions</b>	<ul style="list-style-type: none"> <li>✓ The request status is updated at each stage of the delivery process.</li> <li>✓ Customers receive real-time updates on their request's status and delivery progress.</li> <li>✓ The system records completed deliveries for future reference.</li> <li>✓ The final state of the request is "Delivered" upon successful delivery.</li> </ul>
<b>Alternative Flow</b>	<p>1. Customer cancels the request:</p> <ul style="list-style-type: none"> <li>➤ If the customer decides to cancel the request before a driver is assigned, the request is canceled, and no delivery is made.</li> </ul>
<b>Exceptions</b>	<p>2. Driver Unavailability Exception:</p> <ul style="list-style-type: none"> <li>➤ If there are no available drivers to provide delivery offers when the customer submits a delivery request, the system must notify the customer of the unavailability of drivers at the moment.</li> <li>• The customer can try resubmitting the request later.</li> </ul> <p>3. Driver Offer Rejection Exception:</p> <ul style="list-style-type: none"> <li>➤ If the customer decides to reject all delivery offers provided by drivers, the system must reassign the request and notify the customer that no offers were accepted.</li> <li>• The customer can try again.</li> </ul> <p>4. Driver Assignment Delay:</p> <ul style="list-style-type: none"> <li>➤ If unexpected delays occur in assigning a driver to the request (e.g., due to connectivity issues or driver availability problems)</li> <li>• The system should periodically update the customer about the delay and provide an estimated time for driver assignment.</li> </ul> <p>5. Delivery Delay:</p> <ul style="list-style-type: none"> <li>➤ If the delivery is significantly delayed beyond the expected time due to unforeseen circumstances (e.g., heavy traffic or adverse weather conditions).</li> <li>➤ The customer or system should update the customer with the delay information and provide a new estimated arrival time.</li> </ul>

## 2.6.11 User Ratings and Comments Use Case:

USE CASE NAME: User Ratings and Comments		ID: UC_11	Priority: High
<b>Actor</b>	<ul style="list-style-type: none"> <li>• Customer</li> <li>• System</li> </ul>		
<b>Description</b>	This Use Case outlines the process of allowing customers to rate and leave comments on their experiences with the delivery service.		
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>✓ The customer has successfully completed a delivery request.</li> <li>✓ The delivery driver has successfully completed a delivery.</li> </ul>		
<b>Main Success Scenario (Main Flow)</b>	<p>10. Customer rates the delivery service:</p> <ul style="list-style-type: none"> <li>✓ After the successful delivery of an order, the customer has the option to rate the delivery service provider on three criteria: speed of delivery, product quality, and cost-effectiveness.</li> <li>✓ The customer provides ratings for each of these criteria (e.g., on a scale of 1 to 5 stars).</li> </ul> <p>11. Customer leaves optional comments:</p> <ul style="list-style-type: none"> <li>✓ The customer can also provide optional comments or feedback regarding their delivery experience, including specific comments related to speed, product quality, or cost-effectiveness.</li> </ul> <p>12. System records the ratings and comments:</p> <ul style="list-style-type: none"> <li>✓ The system records the customer's ratings and comments, associating them with the specific delivery request and the involved delivery driver.</li> </ul> <p>13. Driver views the ratings and comments:</p> <ul style="list-style-type: none"> <li>✓ The delivery driver can view the ratings and comments left by the customer for the delivery service, including feedback on speed, product quality, and cost-effectiveness.</li> </ul>		
<b>Postconditions</b>	<ul style="list-style-type: none"> <li>✓ The customer's ratings and comments, including feedback on speed, product quality, and cost-effectiveness, are stored in the system for reference.</li> <li>✓ The delivery driver can access and review the detailed feedback provided by the customer.</li> </ul>		
<b>Alternative Flow</b>	<p>2. Customer does not provide ratings or comments:</p> <ul style="list-style-type: none"> <li>➤ If the customer chooses not to provide ratings or comments on speed, product quality, or cost-effectiveness, the system still records the delivery as completed but without specific feedback.</li> </ul>		
<b>Exceptions</b>	<p>6. Inappropriate Comments:</p> <ul style="list-style-type: none"> <li>➤ If the customer leaves inappropriate or offensive comments, the system should have a mechanism in place to flag and review such comments.</li> <li>➤ Inappropriate comments may be removed, and appropriate actions may be taken.</li> </ul>		

## 2.6.12 Location and Address Selection Use Case:

Use Case Name: Location and Address Selection		ID:UC_12	Priority: Primary
<b>Description</b>	This use case allows users to choose between two options for providing their delivery location and address: manual input or automatic detection. Users can also save multiple delivery addresses for future orders.		
<b>Actor</b>	User (Customer)		
<b>Precondition</b>	<ul style="list-style-type: none"> <li>✓ User is logged in to their account on the Hatley App.</li> <li>✓ User has one or more saved delivery addresses (optional).</li> </ul>		
<b>Normal Course</b>	<p>1-User selects the option to place an order.</p> <p>2-The system presents the user with two options for providing the delivery address:</p> <ul style="list-style-type: none"> <li>• <b>Option 1: Input Address Manually</b> <ul style="list-style-type: none"> <li>• User selects this option to manually input their delivery address.</li> <li>• They provide the following details:           <ul style="list-style-type: none"> <li>• Street address</li> <li>• City</li> <li>• State/Province</li> <li>• Country</li> </ul> </li> </ul> </li> <li>• <b>Option 2: Define Address Automatically</b> <ul style="list-style-type: none"> <li>• User selects this option to have the system automatically detect their delivery location based on their device's geolocation data or IP address.</li> <li>• The system displays the automatically detected address.</li> </ul> </li> </ul> <p>3-User has the option to either:</p> <ul style="list-style-type: none"> <li>• <b>Select the Automatically Detected Address:</b> <ul style="list-style-type: none"> <li>• The user chooses to use the automatically detected address as the delivery address.</li> <li>• They proceed to the next steps of the order placement process.</li> </ul> </li> <li>• <b>Change or Input a Different Address:</b> <ul style="list-style-type: none"> <li>• The user can choose to change the automatically detected address or input a new delivery address manually.</li> <li>• If they select this option, they provide the following details:</li> </ul> </li> </ul> <p>4-User confirms the selected or newly inputted delivery address.</p>		
<b>Postcondition</b>	<ul style="list-style-type: none"> <li>• The selected or newly inputted delivery address is associated with the user's current order.</li> <li>• The user can proceed with the order placement process.</li> </ul>		
<b>Exception</b>	<p>If the user cancels the order or navigates away from the address selection screen:</p> <ul style="list-style-type: none"> <li>• The address selection process is abandoned.</li> <li>• No address is associated with the order.</li> <li>• The user may return to this step later if they decide to continue with the order.</li> </ul>		

## 2.6.13 Order Confirmation and Payment Use Case:

<b>Use Case Name: Order Confirmation and Payment    ID:UC_13    Priority: Primary</b>	
<b>Description</b>	This use case covers the process of confirming the order details, including the total cost, and facilitating the payment process. It supports various payment methods such as credit cards, digital wallets, or cash on delivery.
<b>Actor</b>	User (Customer)
<b>Precondition</b>	<ul style="list-style-type: none"> <li>✓ User has provided a delivery address.</li> <li>✓ User has chosen a preferred payment method.</li> <li>✓ User has reviewed the order details.</li> </ul>
<b>Normal Course</b>	<p>1-User reviews the delivery order details, including the delivery address, and the total cost of the order.</p> <p>2-User selects the preferred payment method from the available options:</p> <ul style="list-style-type: none"> <li>• <b>Credit Card:</b> User enters credit card information (card number, expiration date, CVV).</li> <li>• <b>Digital Wallet:</b> User selects a digital wallet option and logs in to their wallet account.</li> <li>• <b>Cash on Delivery:</b> No additional action is required at this step.</li> </ul> <p>3-User clicks the "Submit Request" button to initiate the payment process.</p> <p>4-If the user has chosen:</p> <ol style="list-style-type: none"> <li>1. <b>Credit Card or Digital Wallet:</b> The system securely processes the payment. If the payment is successful, the delivery order is confirmed.</li> <li>2. <b>Cash on Delivery:</b> The delivery order is confirmed immediately without a payment process.</li> </ol>
<b>Postcondition</b>	<ul style="list-style-type: none"> <li>• The delivery order is confirmed.</li> <li>• Payment (if applicable) is successfully processed, and a payment confirmation is displayed.</li> </ul>
<b>Exception</b>	<p>1-If the payment fails (for credit card or digital wallet), the system displays an error message, and the delivery order is not confirmed.</p> <p>2-If the user cancels the delivery order or navigates away from the payment screen, the delivery order is not confirmed.</p> <p>3-If there are issues with the selected payment method (e.g., insufficient funds for credit card payment), the system displays an error message and does not confirm the delivery order.</p>

## 2.6.14 Request Processing and Assignment Use Case:

Use Case Name: Request Processing and Assignment		ID:UC_14 Priority: Primary
<b>Description</b>	This use case describes the process of handling user requests, assigning them to selected service providers, and providing options for service providers to interact with requests using "Accept" or "Offer" buttons. It also includes the development of a dashboard for managing offers submitted by service providers, allowing them to specify prices and terms.	
<b>Actor</b>	<ul style="list-style-type: none"> <li>○ User (Customer)</li> <li>○ Service Provider</li> </ul>	
<b>Precondition</b>	<ul style="list-style-type: none"> <li>✓ User (Customer) is logged in.</li> <li>✓ Service providers are registered in the system.</li> </ul>	
<b>Normal Course</b>	<ol style="list-style-type: none"> <li>1. User (Customer) submits a service request, providing details about the request and any specific requirements.</li> <li>2. The system receives the service request and stores it as a pending request.</li> <li>3. Service providers registered in the system are notified of the new request.</li> <li>4. Service providers review the request and decide whether to:           <ul style="list-style-type: none"> <li>• <b>Accept the Request:</b> A service provider clicks the "Accept" button if they are willing to take on the request. This action marks them as the assigned provider for the request.</li> <li>• <b>Offer a Proposal:</b> Alternatively, a service provider can click the "Offer" button to submit a proposal with details such as price and terms.</li> </ul> </li> <li>5. If a service provider accepts the request, the system assigns the request to that provider.</li> <li>6. If a service provider submits an offer, the system stores the proposal and notifies the user (Customer) of the offer.</li> <li>7. The user (Customer) reviews the offers and either:           <ul style="list-style-type: none"> <li>• <b>Accepts an Offer:</b> The user (Customer) selects a service provider's offer and marks them as the assigned provider.</li> <li>• <b>Rejects Offers:</b> The user (Customer) can reject one or more offers.</li> </ul> </li> </ol>	

<b>Postcondition</b>	<ul style="list-style-type: none"> <li>• The service request is either assigned to a service provider who accepted it or is still pending if no provider was selected.</li> <li>• If an offer is accepted, the service request is associated with the selected service provider.</li> <li>• If offers are rejected, the request remains pending, and other service providers can still make offers.</li> </ul>
<b>Exception</b>	<ol style="list-style-type: none"> <li>1- If there are no service providers available to handle the request, the request remains pending.</li> <li>2- If no service provider accepts the request or submits an offer, the request remains pending.</li> <li>3- If the user (Customer) cancels the request or navigates away from the request screen, the request remains pending.</li> <li>4- If the service provider's offer is rejected, the system updates the request status but does not assign it to the rejecting provider.</li> </ol>

### 2.6.15 Admin control Use Case:

Use Case Name: Admin control		ID:UC_15	Priority:
Description	This scenario explains how the administrator controls the system and this is through interacting with system, managing users, monitoring data and statistics, and making reports		
Actor	Admin (Customer support representative.)		
Precondition	Admin logged in using his own User name and password		
Normal Course	<ul style="list-style-type: none"> <li>✓ After correct login or authentication, the admin will be redirected to control panel (Admin Dashboard).</li> <li>✓ The admin dashboard displays the following:           <ol style="list-style-type: none"> <li>1. Home page that contains charts of data as how many orders was made last 7 days, information about regions, percentage of increasing on new users (customer, delivery).</li> <li>2. Emails and reports: Admin can review all reports or answer all questions from users, first he open email if it is report about someone, the admin read it carefully then he opens the account of bad behavior user and check if this is true or no, if true he open all users search about him then delete or ban his account and send to him a notification with warning.</li> <li>3. Members:               <ol style="list-style-type: none"> <li>3.1. The admin can view the list of registered users (customer or delivery) and all users' data like pic, phone, email, rate and how many orders he made...etc.</li> <li>3.2. The admin can view and modify user account details, including their roles and permission by searching or clicking on any account he also can ban an account for any time with sending a warning notification.</li> </ol> </li> <li>4. Report and analyses:               <ol style="list-style-type: none"> <li>4.1. The admin can create reports and analyses about the performance, and more statistics.</li> </ol> </li> </ol> </li> </ul>		

	<p>4.2.These reports can include user activities, reports on the service provider's performance, and more statistic.</p> <p>5. Forms And Table:</p> <p>He can control and change the shape of any form, he can add, delete, edit and update any field of each table by clicking on forms and table →edit → add field or delete.</p>
<b>Postcondition</b>	<ul style="list-style-type: none"> <li>✓ Saving data and changes that the administrator has just made to user, service providers account, or system settings.</li> <li>✓ If he made a report we can use these data to make decision to improve our system.</li> </ul>
<b>Exception</b>	No exception.

## 2.7 Management Plan:

Hatley is a delivery platform that aims to provide a faster and more convenient way for users to purchase daily essentials in Egypt. With a focus on creating a competitive marketplace for delivery drivers, Hatley encourages them to provide faster and more cost-effective delivery options for users.

To ensure the long-term success of the platform, it is essential to have a well-thought-out management plan in place. This management plan outlines the key strategies and initiatives that will be implemented to drive growth, improve efficiency, and enhance the user experience for both users and delivery drivers.

In this section, we will outline the management plan for Hatley, including Current State of the Project, Phases, Gantt Chart & Timeline, Task Plan with Milestone, Division of Responsibilities Among Team Members and finally Risk Management.

### 2.7.1 Current State of the Project:

The current state of the project highlights the progress made so far, including the completion of documentation, UI/UX for website, UI/UX for mobile application. Additionally, preparations are underway for the website presentation.

## 2.7.2 Phases:

The management plan consists of a series of phases, each with specific tasks and objectives. The goal of each phase is to bring the project closer to completion and ensure that it meets the desired standards as shown in Figure 2.1. The phases include:

**Phase 1:** Project initiation and research. This including Defining the problem and project idea.

**Phase 2:** Planning and documentation. This including Studying the community need for the idea, creating a roadmap, and starting chapter 1 of software proposal.

**Phase 3:** Preparing and design creation. This including Imagining how the website will work, starting Chapter 2 of the software analysis documentation, and creating the UI/UX for mobile and website.

**Phase 4:** Prototype creation. This including Creating databases, finishing Chapters 3 and 4 of the software design documentation, and finishing the documentation.

**Phase 5:** Hosting prototype. This including Finishing the database and backend for the website, hosting the prototype, and preparing for the integration of the system.

**Phase 6:** Testing and integration. This including Testing and launching on real time, preparing the final web and app version, and preparing for the second presentation and discussion phase.

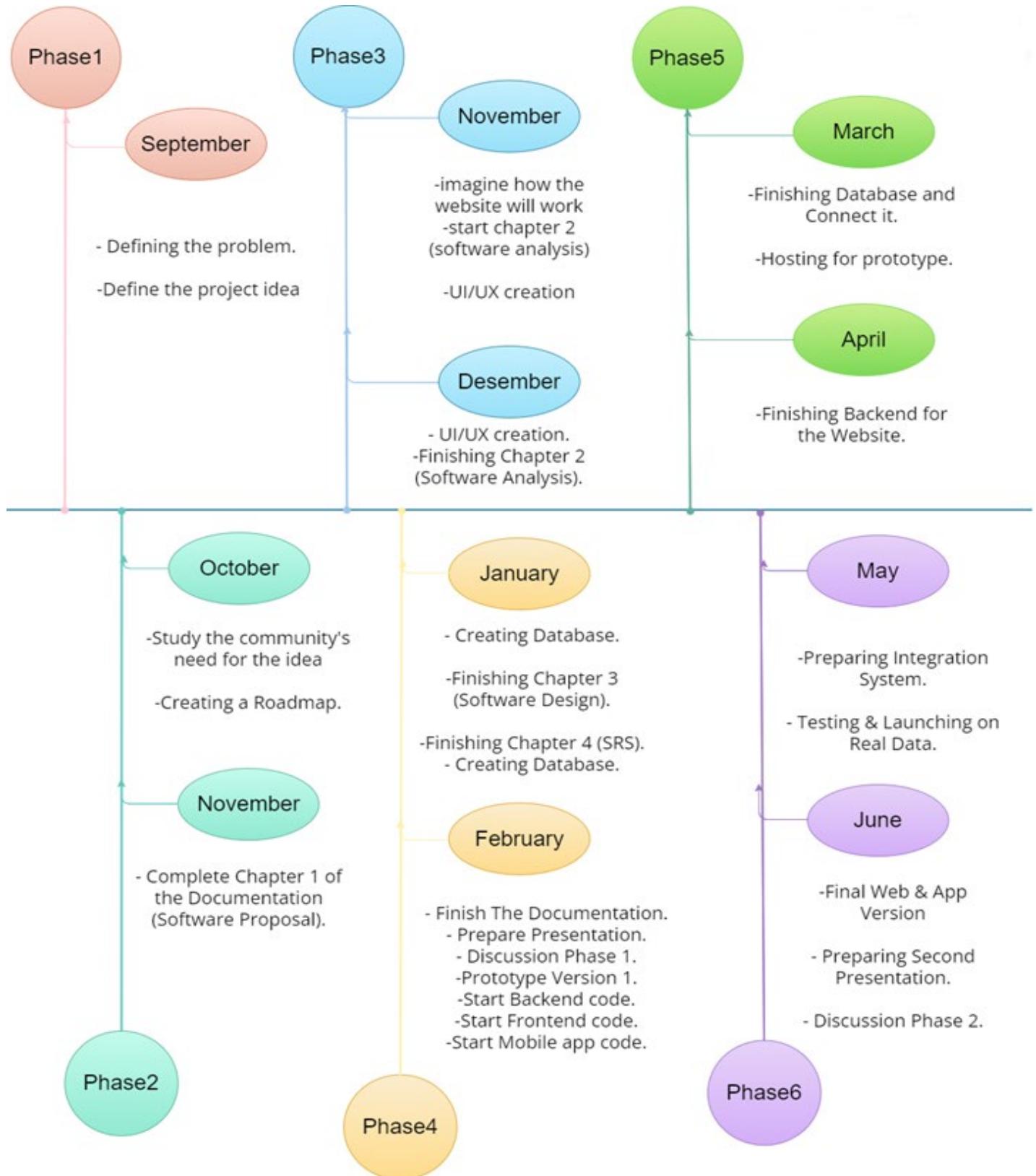


Figure 2.1: Phases and Timeline for Hatley Project.

### 2.7.3 Gantt Chart & Timeline:

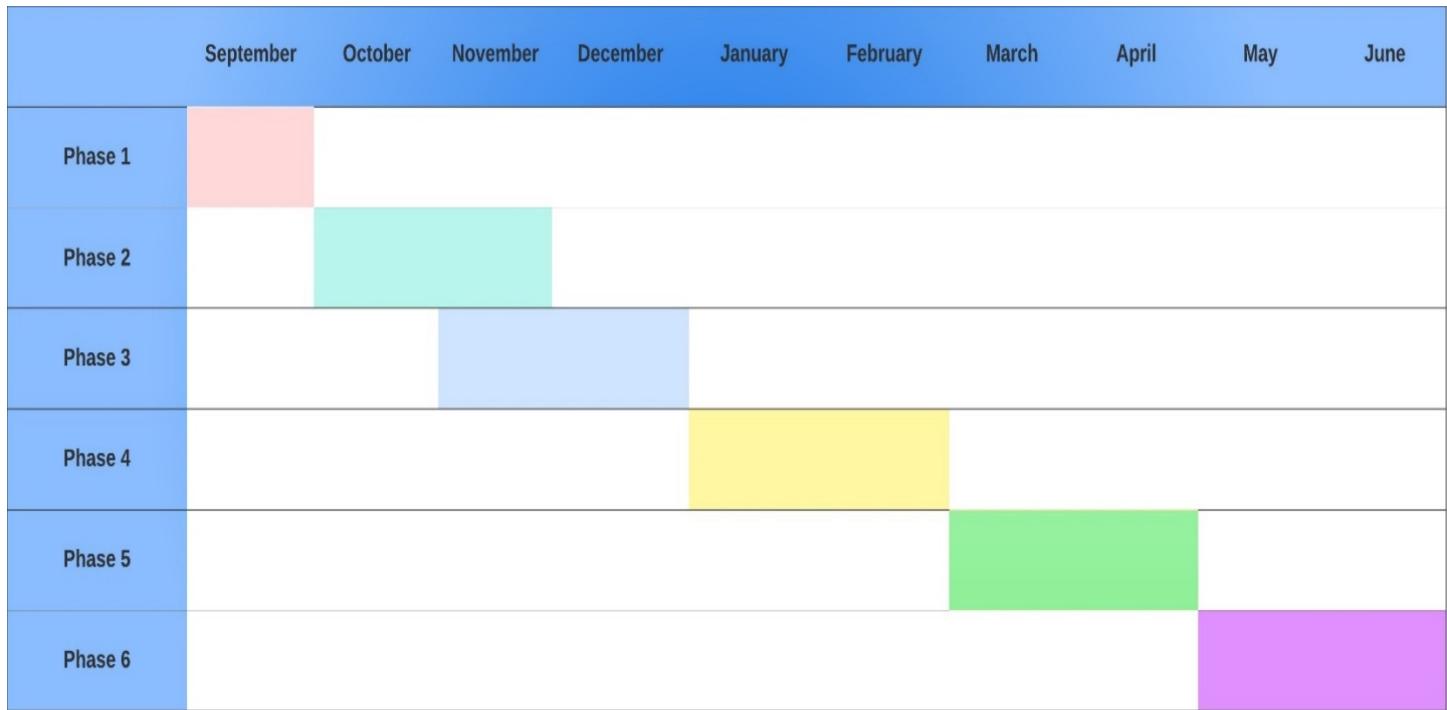


Figure 2.2: The Gantt Chart of General Management and Time Plan of Hatley Project.

## 2.7.4 Task Plan with Milestone:

	Task Description	2023				2024					
		SEP	OCT	NOV	DEC	JAN	FEB	MAR	APR	MAY	JUN
1	Define the problem.	✓									
2	Define the project idea.	✓									
3	Study the community need for the idea.			✓							
4	Creating a Roadmap.		✓								
5	complete Chapter 1 of the Documentation (Software Proposal).				✓						
6	Imagine How the website will work.			✓							
7	Start Chapter 2 (Software Analysis).			○		✓					
8	UI/UX creation.			○		✓					
9	Finishing chapter 2 (software Analysis).				✓						
10	Creating Databases.				○		✓				
11	Finishing Chapter 3. (Software Design).					✓					
12	Finishing Chapter 4 (SRS).					✓					
13	finish the documentation.							✓			
14	Prepare presentation.							✓			
15	Discussion Phase1.							✓			
16	Prototype version 1.							✓			
17	Start coding.					○				✓	
18	Finishing Database and connect it.							✓			
19	Hosting for prototype							✓			
20	Finishing backend for website.								✓		
21	Preparing integration for system.									✓	
22	Testing& launching on real time.									✓	
23	Final web &app version.										✓
24	Preparing second presentation.										✓
25	Discussion phase 2.										✓

Figure 2.3: Task Plan with Milestone.

## 2.7.5 Division of Responsibilities Among Team Members:

Division of Responsibilities Among Team Members is an important aspect of the Management Plan for the project. This helps to ensure that every team member understands their role and responsibilities, and it also helps to distribute the workload evenly as shown in Figure 2.4.

	Ahmed Ali	Santy Osama	Abdullah Salah	Rahma Bahaa	AbdelRahman Omar			
Phase 1	Define the problem							
	Define the project idea							
Phase 2	Study the community need for the idea							
	Creating a Roadmap							
	complete Chapter 1 of the Documentation (Software Proposal)							
Phase 3	Imagine How the website will work							
	Start Chapter 2 (Software Analysis)							
	UI/UX creation				UI/UX creation			
	Finishing chapter 2 (software Analysis)							
Phase 4			Creating Databases					
			Finishing Chapter 3 (Software Design)		Finishing Chapter 3 (Software Design)			
	Finishing Chapter 4 (SRS)			Finishing Chapter 4 (SRS)				
	finish the documentation							
	Prepare presentation							
	Discussion Phase1							
	Prototype version 1							
	Start frontend code		Start backend code		Start mobile code			
Phase 5			Finishing Backend for the Website					
	Hosting for prototype							
			Finishing backend for website					
phase 6	Preparing integration for system							
	Testing& launching on real time							
	Final web &app version							
	Preparing second presentation							
	Discussion phase 2							

Figure 2.4: Table of Division of Responsibilities for Hatley Project Team.

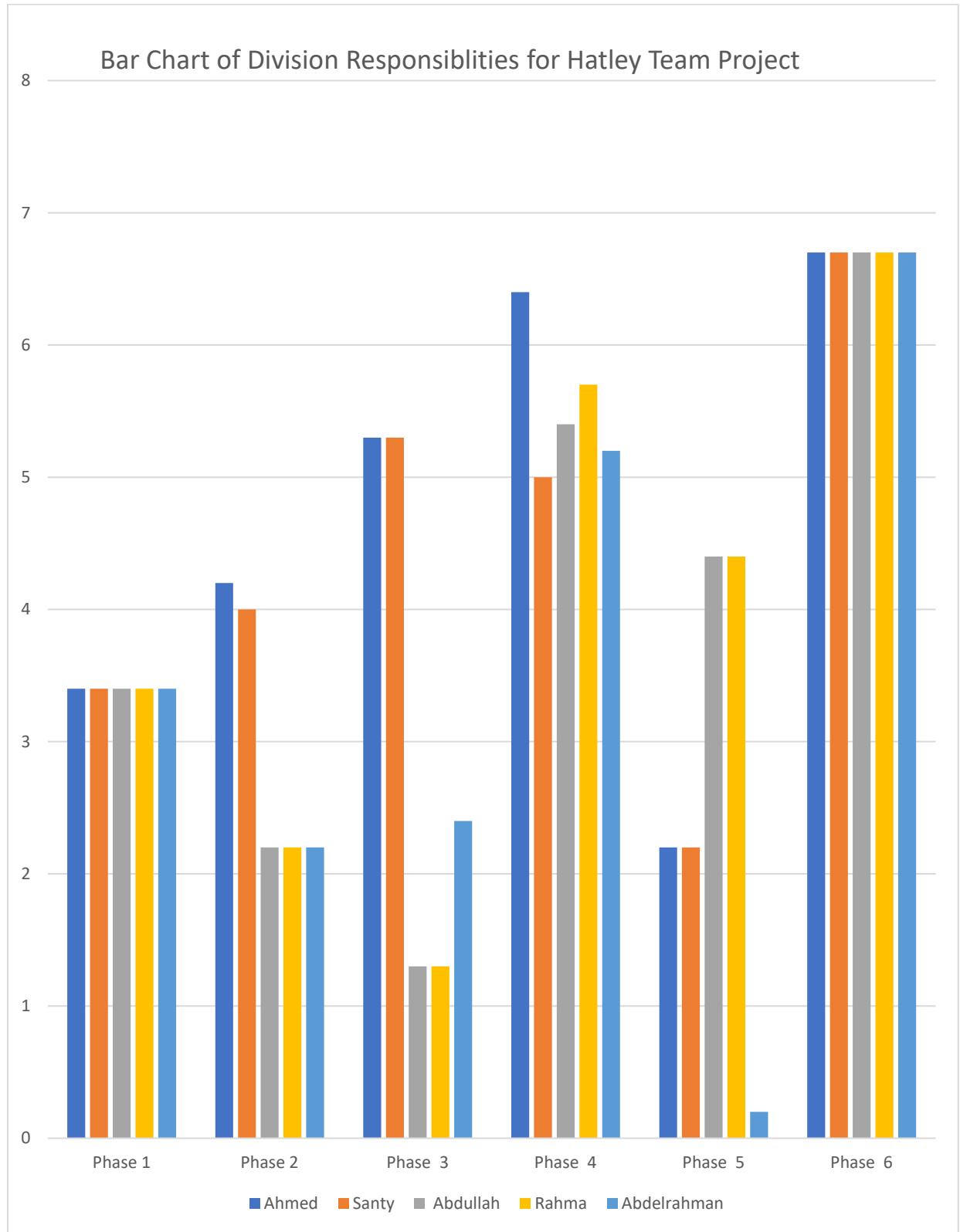


Figure 2.5: Bar Char of Division of Responsibilities for Hatley Project Team.

## 2.7.6 Risk Management:

Moving forward with the Hatley management plan brings potential obstacles. By smart planning, evaluating risks, and using right strategies, we can navigate through these. Hatley might encounter risks like:

1. **Challenging Market:** There's a chance our service doesn't hit the mark, or perhaps there's an oversupply of similar offerings, making distinction quite difficult.
2. **Financial Hurdles:** Funding isn't always plentiful and unforeseen expenses might surface throughout the project's course.
3. **Operational Obstacles:** Occasionally, plans derail and unpredicted hitches materialize.
4. **Security Concerns:** Upholding the safety and privacy of user information is critical. But, it's no small task.

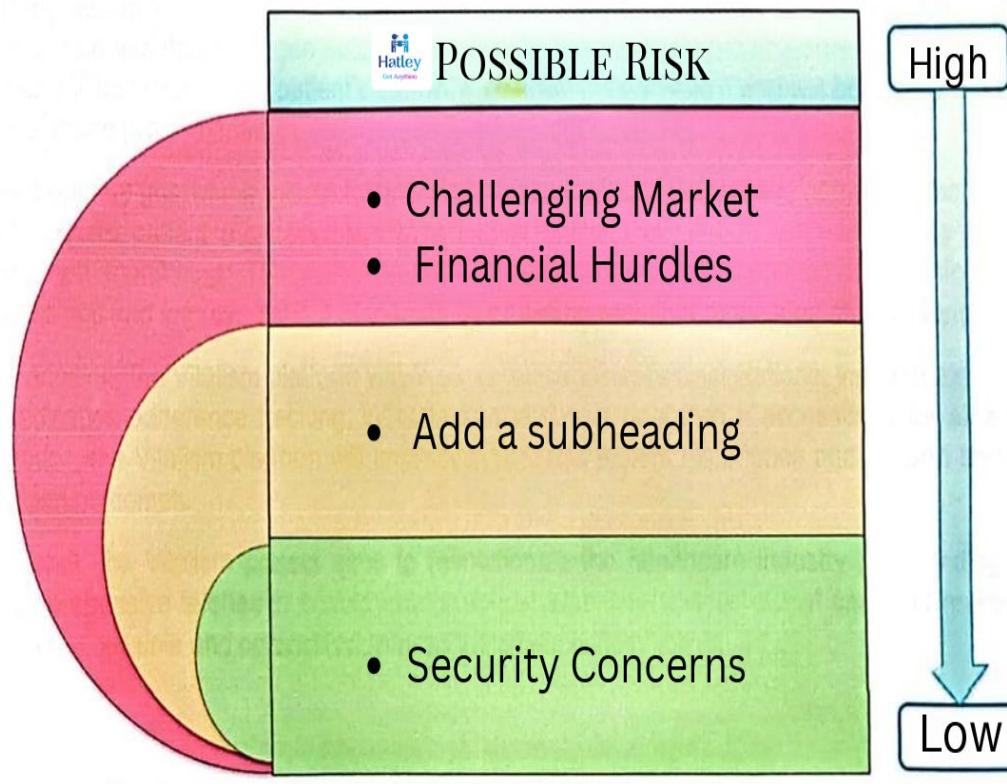


Figure 2.6: The Possible Risks of Hatley.

# 3. Software Design

## 3.1 Introduction:

The software design phase for Hatley is crucial in creating a robust delivery service application. It involves defining the application's architecture, modules, and interfaces, considering functional and non-functional requirements. The design must prioritize user-friendliness, security, reliability, and performance while ensuring comprehensive functionality. It must align with stakeholder expectations and provide a clear depiction of the application. After the design, a thorough analysis of requirements traceability documents should show a correlation between design components and requirements, confirming all requirements have been addressed.

### 3.1.1 Project Background/Purpose

The purpose of the Hatley project is to develop an innovative delivery service application that offers convenience and efficiency to users in Egypt. Hatley aims to streamline the delivery process by connecting users with delivery personnel through a user-friendly platform. The primary objective of Hatley is to provide a cost-effective, time of delivery and accessible solution for ordering and delivering various products.

The Hatley application will facilitate seamless order placement and tracking, allowing users to specify their requirements and preferences. Delivery personnel will utilize the platform to receive and fulfill orders promptly, optimizing delivery routes and ensuring timely service. Through its intuitive interface and broad accessibility, Hatley aims to enhance the overall delivery experience and support efficient logistics management.

In addition to standard delivery services, Hatley will also offer features such as competitive bidding and real-time order matching to improve service efficiency and customer satisfaction. By leveraging technology and user-centric design, Hatley seeks to revolutionize the delivery industry in Egypt, making it more convenient, reliable, and accessible to all.

Overall, the Hatley project aims to transform the delivery landscape by providing a comprehensive and user-friendly platform that enhances the delivery experience for both users and delivery personnel, ultimately fostering efficiency and convenience in the delivery process.

### 3.1.2 Hosting Platform

The Hatley app is compatible with multiple cloud hosting platforms, providing scalability and flexibility to meet its requirements. One such platform that offers a

range of cloud computing services for hosting web applications is Microsoft Azure. Google Cloud Platform (GCP), which provides a variety of modular cloud services like computation, data storage, analytics, and machine learning, is yet another feasible choice for hosting Hatley. Usually, financial information like a credit card number or bank account is needed to register for GCP. Although Google Cloud Platform and Microsoft Azure offer reliable hosting options, it's important to take into account specific aspects like cost, integration needs, and scalability when selecting Hatley's hosting platform.

## 3.2 System Architecture:

The Hatley system architecture, depicted in Figure, encompasses a comprehensive framework designed to facilitate efficient and seamless delivery services. It consists of various modules and components, including order management, delivery tracking, user authentication, bidding system, and rating feedback mechanism. The architecture prioritizes scalability, reliability, security, and performance to ensure optimal service delivery for users and delivery personnel.

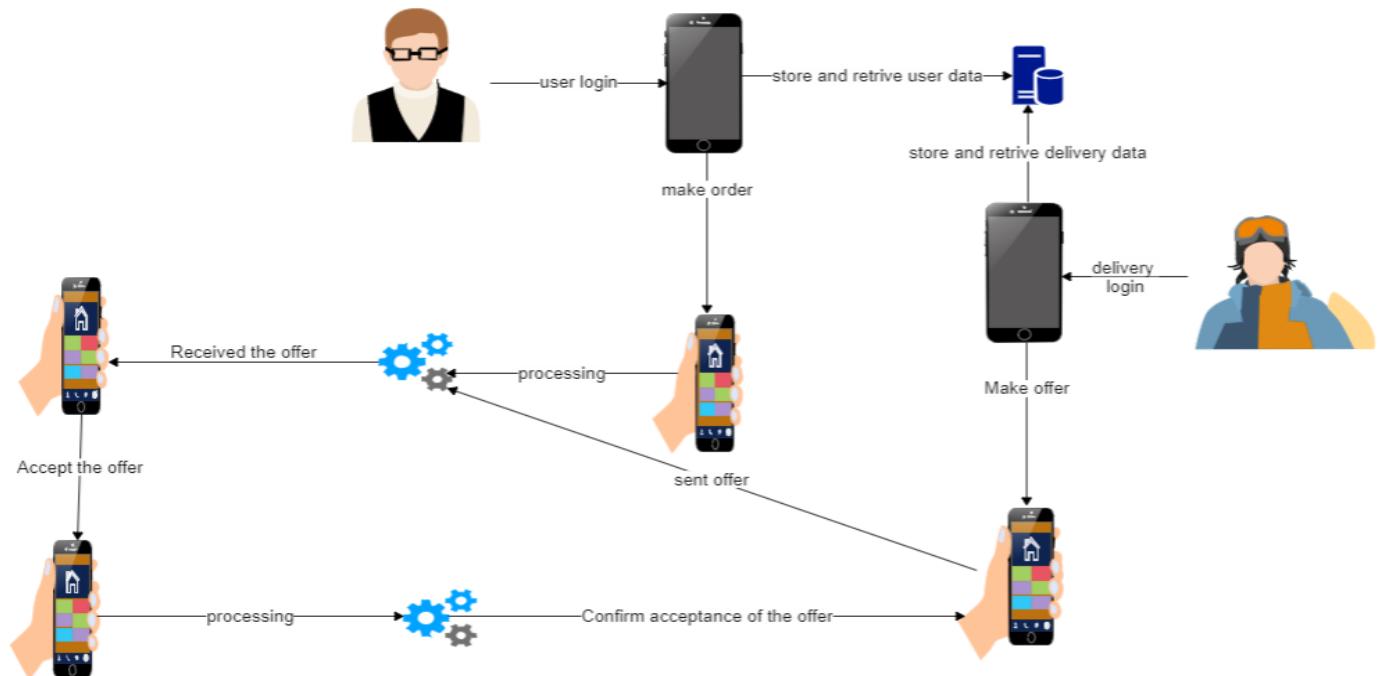


Figure 2.7: System Life Cycle.

The Hatley system architecture leverages modern technologies such as geolocation services for dynamic order management and efficient delivery routing and cloud computing and real-time communication to deliver a highly responsive and intuitive delivery service platform. By embracing innovative solutions and user-centric design principles. The system aims to optimize the delivery process, minimize delivery times, and maximize customer satisfaction, revolutionizing the Egyptian delivery industry.

### 3.2.1 Architectural Design

The architecture of the Hatley app follows a multi-tier architecture, as depicted in Figure 42, which comprises the following layers:

- 1. Front-end layer:** This layer is responsible for presenting the user interface and interacting with users. In the case of Hatley, the front-end layer utilizes HTML, CSS, and JavaScript for web-based applications, and react for creating dynamic and responsive user interfaces. For the mobile application, React Native is employed to develop cross-platform mobile apps that offer consistent user experiences across different devices.
- 2. Application layer:** Situated between the front-end and back-end layers, the application layer hosts the business logic of the Hatley app. It processes user inputs, performs necessary computations, and communicates with the back-end layer to fetch or update data. In Hatley, the application layer is implemented using ASP.NET Core API, which provides a robust framework for building scalable and secure web APIs.
- 3. Backend layer:** This layer manages the storage and retrieval of data, serving as the backbone of the application. The backend layer of Hatley utilizes SQL Server as the database management system, ensuring efficient data storage and retrieval. SQL Server offers features such as reliability, scalability, and transaction support, making it suitable for handling the data requirements of a delivery service application.

By employing this architectural design, Hatley ensures separation of concerns, scalability, and maintainability of the application. The use of modern technologies such as React and React Native for the front-end, ASP.NET Core API for the application layer, and SQL Server for the backend layer enables Hatley to deliver a seamless and efficient delivery service experience to its users.

### 3.2.2 Decomposition Description

Decomposition involves breaking down a complex system like the Hatley delivery application into smaller, more manageable components. Here's how we can decompose its architecture:

- 1. User Interface (UI):** This component focuses on providing a user-friendly interface for customers to interact with the Hatley app. It includes features such as order placement, tracking, and feedback submission. Separating the UI ensures clarity and ease of maintenance.

- 2. Order Management:** Responsible for handling the processing of orders, including creation, assignment to delivery personnel, and status updates. This component ensures efficient order handling and scalability as the application grows.
- 3. Bidding System:** Enables delivery personnel to bid on orders based on factors like delivery time and cost. Separating this functionality ensures flexibility and optimization without affecting other parts of the app.
- 4. User Authentication and Authorization:** Ensures secure access to the app, managing user accounts, authentication, and authorization levels. Separating this component enhances security and simplifies user management.
- 5. Notification System:** Facilitates communication between users and delivery personnel, sending notifications for order updates and important events. Isolating this functionality ensures timely and reliable notifications.
- 6. Location Services:** Handles location tracking and geocoding functionalities for accurate delivery location specification and real-time order tracking. Separating this component ensures efficient handling of location-related tasks.
- 7. Payment Gateway Integration:** Integrates with payment gateways for secure online payments. Segregating this component ensures reliable and secure payment processing.
- 8. Rating and Feedback System:** Allows users to rate their delivery experience and provide feedback. This component contributes to service improvement and customer satisfaction without complicating other functionalities.

Each component contributes to the Hatley application's overall functionality and efficiency. Decomposing the system architecture in this way ensures a clear separation of concerns, scalability, and maintainability, facilitating efficient development and deployment processes.

### 3.2.3 Design Rationale

We chose to organize the Hatley app using a multi-tier architecture because it helps us keep things organized and manageable as the app grows. Here's why we made this decision:

- 1. Clear Separation of Concerns:** Each part of the app has its own job to do. The front-end layer handles what you see and interact with, like placing orders. The middle layer takes care of the logic behind the scenes, like processing orders. And the back-end layer deals with storing and retrieving data, like your order history. Keeping these parts separate makes it easier to work on and update the app without causing problems in other areas.

2. **Scalability:** Our architecture allows us to grow the app smoothly. We can scale each part independently to handle more users and data as the app becomes more popular. This means we can keep providing a smooth experience even when lots of people are using the app at the same time.
3. **Maintainability:** With everything neatly organized into layers, it's easier for us to keep the app running smoothly over time. If we need to make changes or fix something, we can focus on one layer without worrying about breaking the rest of the app.
4. **Flexibility with APIs:** We use APIs to connect the different layers of the app. This gives us flexibility to add new features or connect with other services in the future. It's like having different parts of the app talk to each other in a way that's easy to understand and change.

Overall, this architecture helps us build and grow Hatley in a way that's reliable, easy to work with, and adaptable to future needs.

## 3.3 Data Design:

The Data Design section of the Hatley app documentation outlines the architecture of the software and how the various components interact with each other. It defines the different modules of the system, their functionalities, and the data structures that they use. Additionally, it describes the relationships between the different modules and their data flow. This section provides an overview of the data flow within the system and how the various modules interact with each other to perform the necessary operations.

### 3.3.1 Data Description

The data description of the Hatley app can be divided into three main entities, each representing different aspects of the application:

- 1. User Entity:** This entity represents the individuals who request orders through the app. Users can place multiple orders. The user entity includes attributes such as user ID, username, and contact information.
- 2. Delivery Entity:** This entity manages the delivery process within the app. It includes information about each order, such as order description, delivery price, and estimated delivery time. The delivery entity establishes a relationship between users and their respective orders, indicating who placed each order and its details.
- 3. Order Entity:** This entity contains details about the orders placed by users. Each order is associated with a specific user and includes information such as the items requested, delivery

address, and any special instructions. The order entity facilitates the interaction between users and delivery personnel, ensuring smooth order processing and delivery.

### 3.3.2 Data Dictionary

A data dictionary serves as a comprehensive guide to the data objects within the Hatley app, defining their structure, characteristics, and relationships. It ensures consistency and accuracy of data throughout the system. Here are some key data elements in the Hatley app's data dictionary:

#### 1-User

Column Name	Data Type	Allow Nulls	Description
User ID	int	NO	Unique identifier for the user
Name	nvarchar(MAX)	NO	Name of the user
Phone	nvarchar(MAX)	NO	Phone number of the user
Email	nvarchar(MAX)	NO	Phone number of the user
Password	nvarchar(MAX)	NO	Password of the user
rate	float	Yes	Rate of user

#### 2-Delivery

Column Name	Data Type	Allow Nulls	Description
Delivery ID	int	NO	Unique identifier for the delivery
Name	nvarchar(MAX)	NO	Name of the delivery person
Phone	nvarchar(MAX)	NO	Phone number of the delivery person
Email	nvarchar(MAX)	NO	Email address of the delivery person
Password	nvarchar(MAX)	NO	Password of the delivery person
rate	float	Yes	Rating of the delivery person
National ID	int	NO	National ID of the delivery person

Front National ID Img	nvarchar(MAX)	NO	Image of the front of the delivery person's National ID
Back National ID Img	nvarchar(MAX)	NO	Image of the back of the delivery person's National ID
Face with National ID Img	nvarchar(MAX)	NO	Image of the delivery person's face with their National ID
Governorate ID	int	NO	Foreign key to the Governorates table

### 3-Order

Column Name	Data Type	Allow Nulls	Description
Order ID	int	NO	Unique identifier for the order
User ID	int	NO	Foreign key to the Users table
Description	nvarchar(2000)	NO	Description of the order
Location	nvarchar(MAX)	NO	Location of the order pickup
Price	int	NO	Price of the order
Status	int	NO	Status of the order
Delivery ID	int	YES	Foreign key to the Delivers table

## 3.4 Component Design:

The component design describe communication, interfaces, algorithms and the functionality of each component regarding the whole design.

### 3.4.1 User Authentication module

#### 1. Introduction:

Provide a concise overview of the User Authentication module and its significance in securing user and delivery personnel accounts within the delivery service application.

#### 2. Technology Stack

Outline the technologies used in the project, including:

- Frontend: React for the website, React Native for mobile apps.
- Backend: ASP.NET Web API.
- Database: SQL Server on a server like Smart ASP.

### **3. Authentication Mechanism**

- Describe the use of JSON Web Tokens (JWT) for user and delivery personnel authentication.
- Explain the token generation process during login.
- Highlight token validation and expiration.

### **4. Data Storage**

Detail the data storage approach:

- Utilization of SQL Server for storing user and delivery personnel data.
- Use of Smart ASP server for secure data storage.

### **5. User Authentication Workflow**

Walk through the steps users and delivery personnel take to register and log in:

- User registration and login process.
- Data required during registration (name, email, phone).
- Optional data (photo).
- ID requirement for delivery personnel.

### **6. Security Measures**

Discuss security measures implemented:

- Encryption of sensitive data during transmission.
- Secure storage of passwords using appropriate hashing algorithms.
- Token-based authentication for secure user sessions.

### **7. Integration with Delivery Personnel**

Explain how user authentication integrates with delivery personnel authentication:

- Separate authentication for delivery personnel.

- Verification process for delivery personnel ID.

## 8. User Authentication API

Detail the API endpoints for user authentication:

- Registration endpoint.
- Login endpoint.
- Token generation and validation endpoints.

### 3.4.2 Data Storage Module

#### 1. Introduction

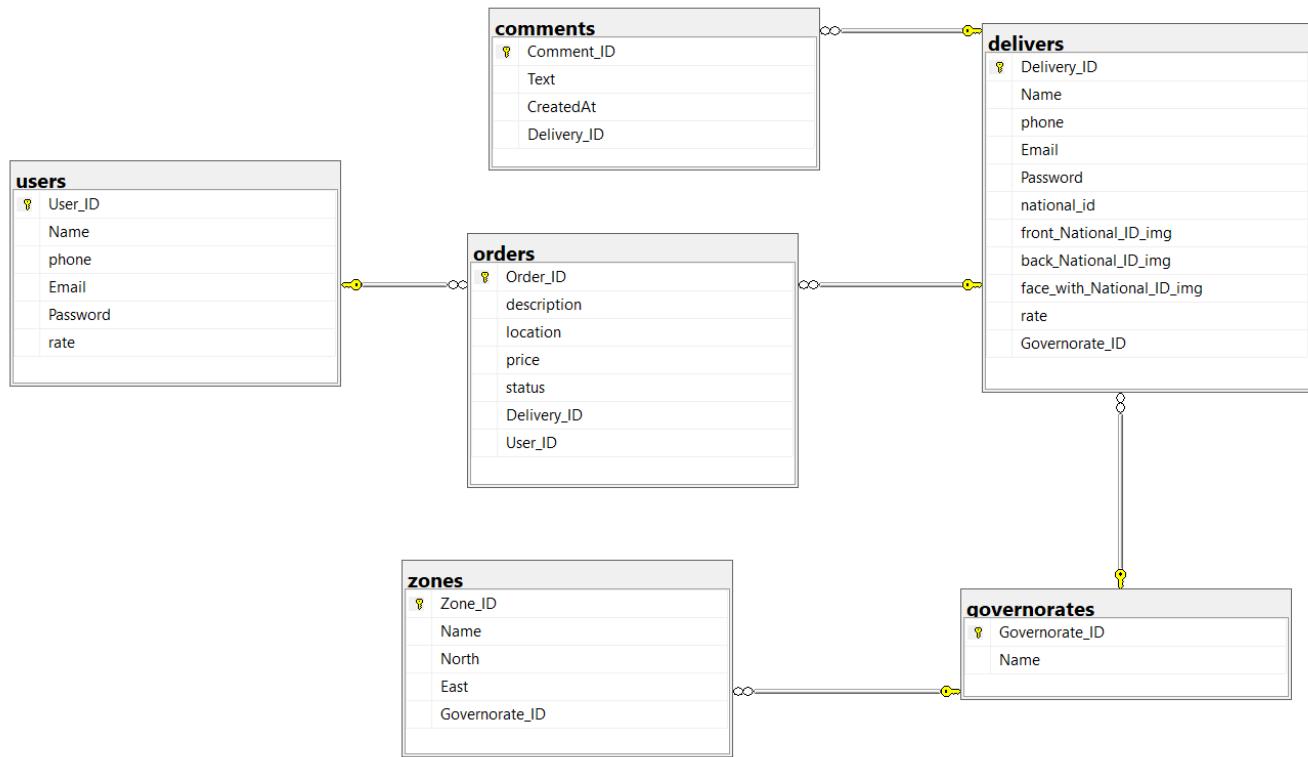
The Data Storage Module in the "Hatley" project is responsible for managing and storing data across five key tables: User, Delivery, Order, Location, and Chat.

#### 2. Database Management

Utilizing SQL Server as the chosen Database Management System for the "Hatley" project.

#### 3. Schema and ERD

- User Table:
  - Attributes: name, email, password, comments, rate.
- Delivery Table:
  - Attributes: name, email, password, comments, rate, personal id, photo.
- Order Table:
  - Attributes: description, location, price, state.
  - Relationships: Connects with User and Delivery tables.
- Location Table:
  - Attributes: location name.
  - Relationships: Connected with Delivery table.



## 4. Data Security

Implementing measures for data security within each table:

- Encryption of sensitive data such as passwords.
- Strict access control and permissions.

## 5. Data Access Layer

Establishing an effective Data Access Layer to facilitate interactions between the "Hatley" application and the database:

- Utilizing SQL queries to retrieve, insert, and update data.
- Implementing an ORM (Object-Relational Mapping) tool if applicable.

## 6. Scalability

Considering database scalability for each table:

- Designing database architecture to accommodate potential data growth.
- Evaluating strategies for scaling the database as needed.

## 7. Performance Optimization

Employing methods for optimizing database performance:

- Implementing indexing strategies.
- Utilizing query optimization techniques.

## 8. Data Integrity

Ensuring data integrity through constraints:

- Primary and foreign keys to establish relationships.
- Validation rules for input data.

## 9. Backup and Recovery

Establishing a robust backup and recovery strategy:

- Regularly backing up data to prevent loss.
- Implementing procedures for data restoration in case of failures.

## 10. Monitoring and Maintenance

Incorporating tools and practices for database monitoring and maintenance:

- Regular monitoring for performance issues.
- Scheduled maintenance tasks to optimize database performance.

### 3.4.3 Data Processing Module

#### A Data Flow Diagram (DFD) Level 0 for "Hatley" Project

##### 1. Introduction

The DFD Level 0 illustrates the high-level overview of data flows and processes within the "Hatley" delivery service application.

##### 2. Processes:

##### **Order Processing:**

- Coordinates the processing of user orders.
- Assigns orders to delivery personnel.

##### 3. Data Stores:

##### a) **User Data Store:**

- Stores user-related information (name, email, etc.).

**b) Delivery Personnel Data Store:**

- Stores delivery personnel-related information (name, email, etc.).

**c) Order Data Store:**

- Stores order-related details (description, location, etc.).

**d) Location Data Store:**

- Stores location-related information.

**e) Chat Data Store:**

- Stores chat-related details.

**4. External Entities:**

**a) User Interface (UI):**

- Represents the user interface of the "Hatley" mobile and web applications.

**b) External Services:**

- Represents any external services or systems interacting with the "Hatley" application.

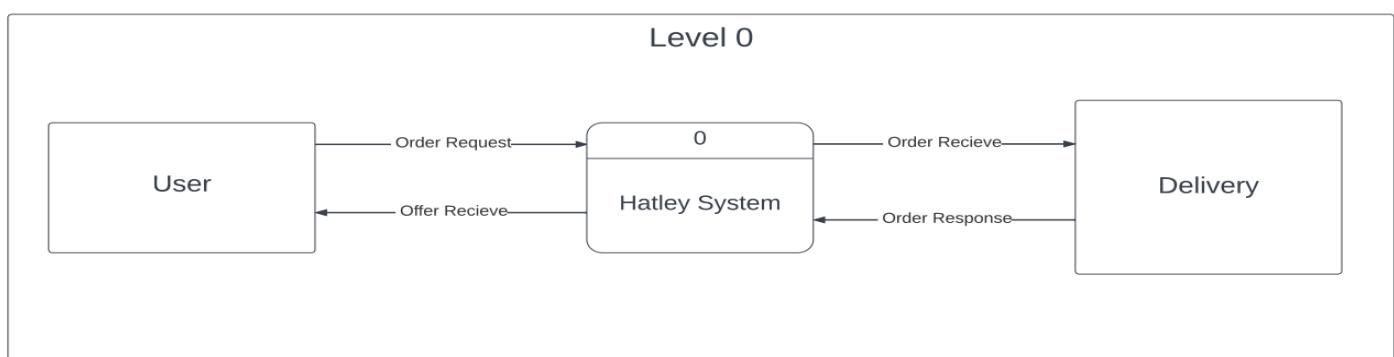
**5. Data Flows:**

**a) Order Data Flow:**

- Manages the flow of data associated with order processing.

**b) Data Processing Data Flow:**

- Enables the flow of data for processing tasks.



### 3.4.4 Data flow Diagram:

The DFD Level 1 illustrates specific processes and data flows within the "Hatley" delivery service application.

#### Processes:

##### 1. User Registration Process:

- Captures and verifies user details for registration.
- Outputs: User Data Flow.

##### 2. User Login Process:

- Manages user authentication and authorization.
- Outputs: User Authentication Data Flow.

##### 3. Delivery Registration Process:

- Gathers and validates delivery personnel details for registration.
- Outputs: Delivery Personnel Data Flow.

##### 4. Delivery Login Process:

- Handles authentication for delivery personnel.
- Outputs: Delivery Personnel Authentication Data Flow.

##### 5. Make Order Process:

- Enables users to create new delivery requests.
- Inputs: User Authentication Data Flow.
- Outputs: Order Data Flow.

##### 6. Notification Alert Process:

- Generates and sends notifications to users and delivery personnel.
- Inputs: Order Confirmation Data Flow, Delivery Selection Data Flow.
- Outputs: Notification Data Flow.

##### 7. Display Request and Information Process:

- Presents order details and information to users and delivery personnel.

- Inputs: Order Data Flow, Delivery Selection Data Flow.

- Outputs: Order Display Data Flow.

#### 8. Delivery Selection Process:

- Facilitates the selection of available delivery personnel by users.

- Inputs: User Authentication Data Flow, Order Data Flow.

- Outputs: Delivery Selection Data Flow.

#### 9. Order Selection Process:

- Allows delivery personnel to choose and accept orders.

- Inputs: Delivery Personnel Authentication Data Flow, Order Data Flow.

- Outputs: Order Selection Data Flow.

#### 10. Order Confirmation Process:

- Confirms the acceptance of an order by the selected delivery personnel.

- Inputs: Delivery Selection Data Flow.

- Outputs: Order Confirmation Data Flow.

#### 11. Order Tracking Process:

- Enables users to track the status and location of their orders.

- Inputs: Order Confirmation Data Flow, Delivery Rate Data Flow.

- Outputs: Order Tracking Data Flow.

#### 12. User Rate Process:

- Allows users to provide feedback and ratings for delivery service.

- Inputs: Order Tracking Data Flow.

- Outputs: User Rate Data Flow.

#### 13. Delivery Rate Process:

- Allows delivery personnel to receive feedback and ratings from users.

- Inputs: Order Tracking Data Flow.

- Outputs: Delivery Rate Data Flow.

#### Data Stores:

1. User Data Store:
  - Stores user-related information.
2. Delivery Personnel Data Store:
  - Stores delivery personnel-related information.
3. Order Data Store:
  - Stores order-related details.

**Data Flows:**

1. User Data Flow:
  - Carries data related to user registration and login.
2. User Authentication Data Flow:
  - Carries data for user authentication and authorization.
3. Delivery Personnel Data Flow:
  - Carries data related to delivery personnel registration and login.
4. Delivery Personnel Authentication Data Flow:
  - Carries data for delivery personnel authentication and authorization.
5. Order Data Flow:
  - Carries data related to order creation.
6. Notification Data Flow:
  - Carries data for generating and sending notifications.
7. Order Display Data Flow:
  - Carries data for displaying order details.
8. Delivery Selection Data Flow:
  - Facilitates the selection of available delivery personnel.
9. Order Selection Data Flow:
  - Allows delivery personnel to choose and accept orders.
10. Order Confirmation Data Flow:
  - Confirms the acceptance of an order.

**11. Order Tracking Data Flow:**

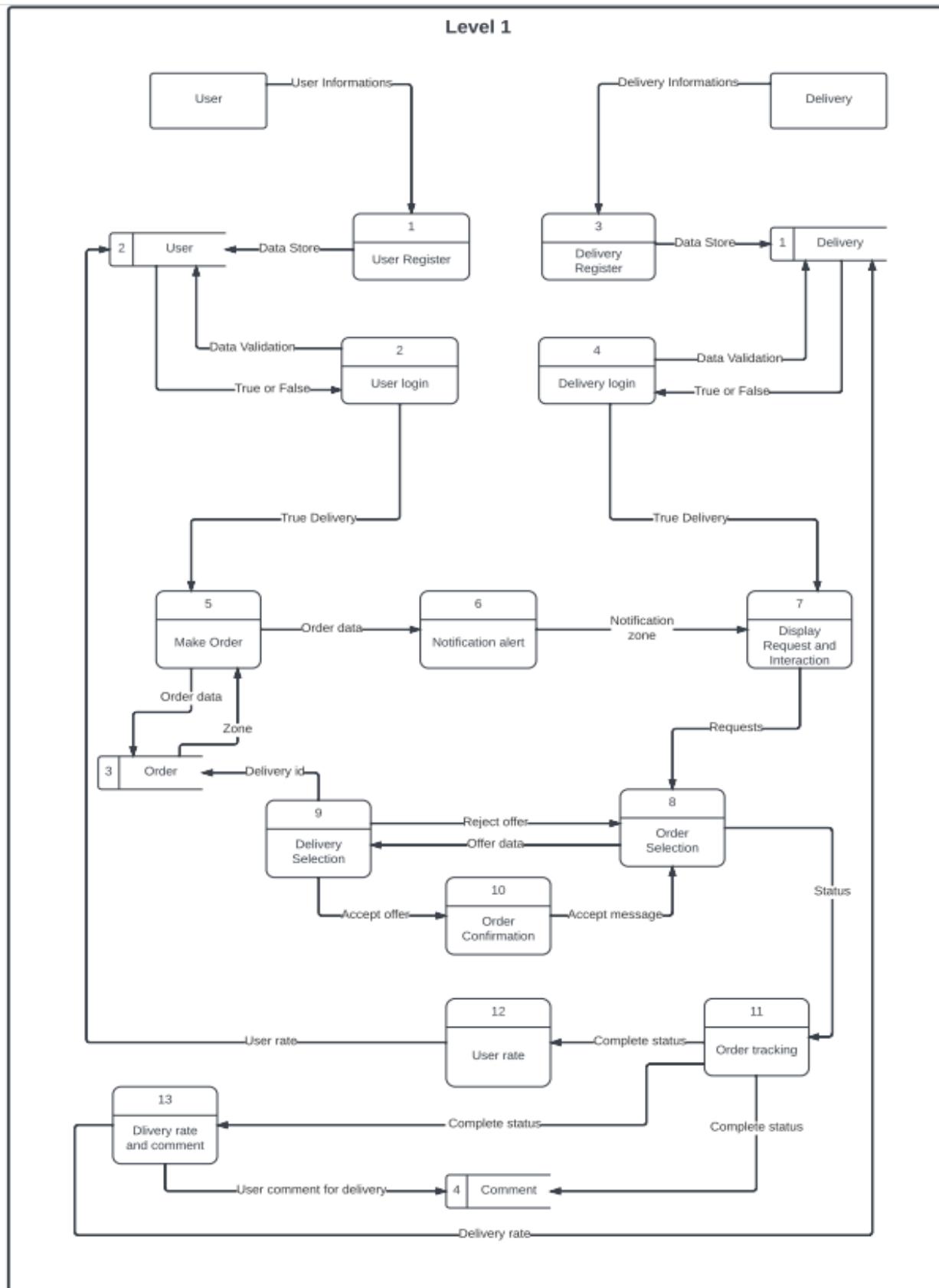
- Facilitates order tracking.

**12. User Rate Data Flow:**

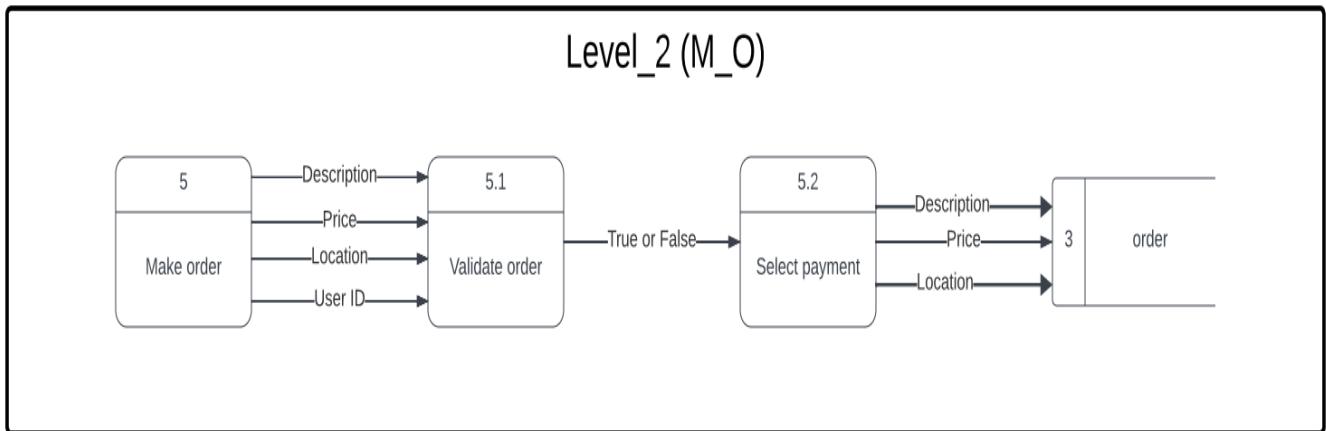
- Carries data for user feedback and ratings.

**13. Delivery Rate Data Flow:**

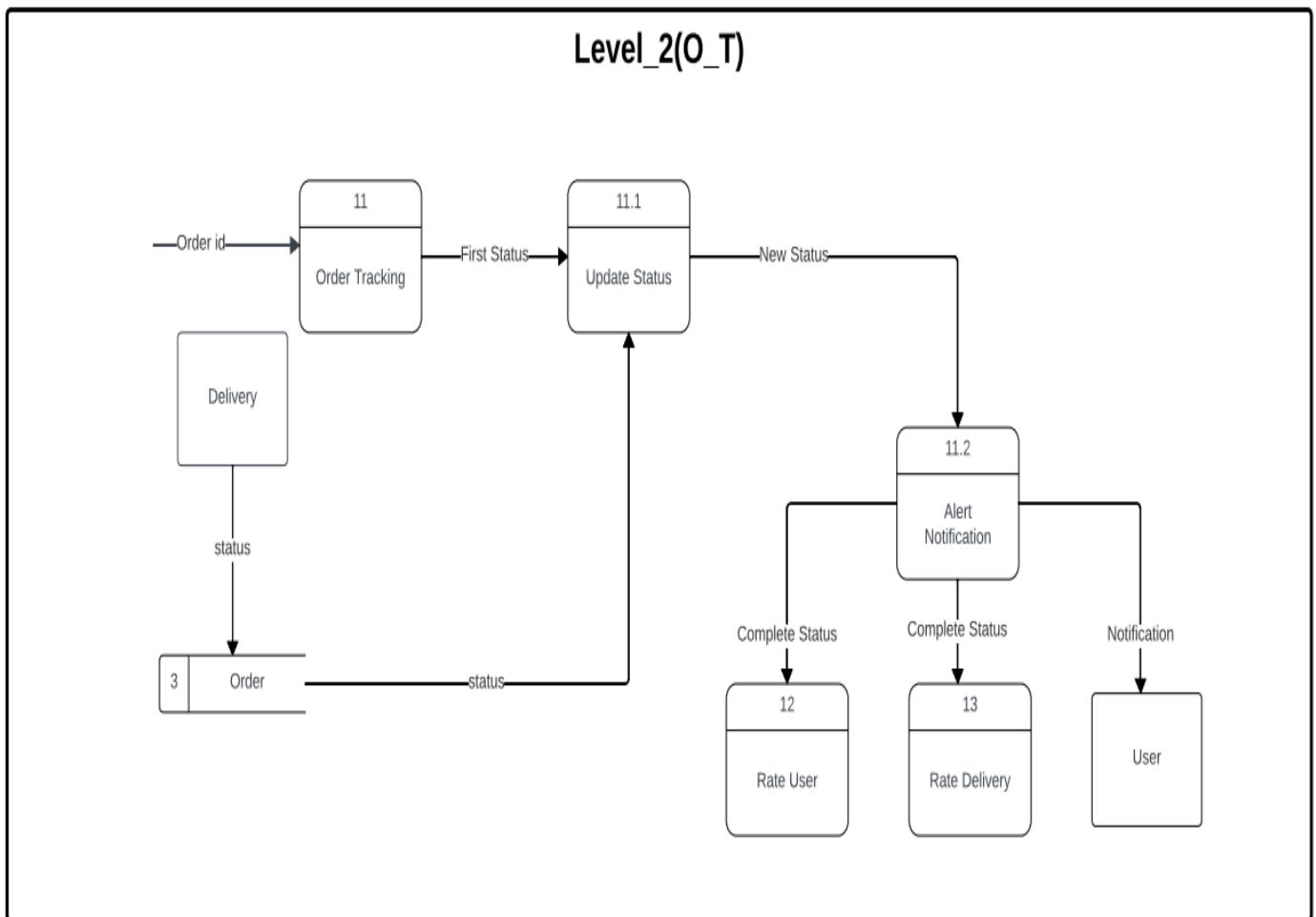
- Carries data for delivery personnel feedback and ratings.



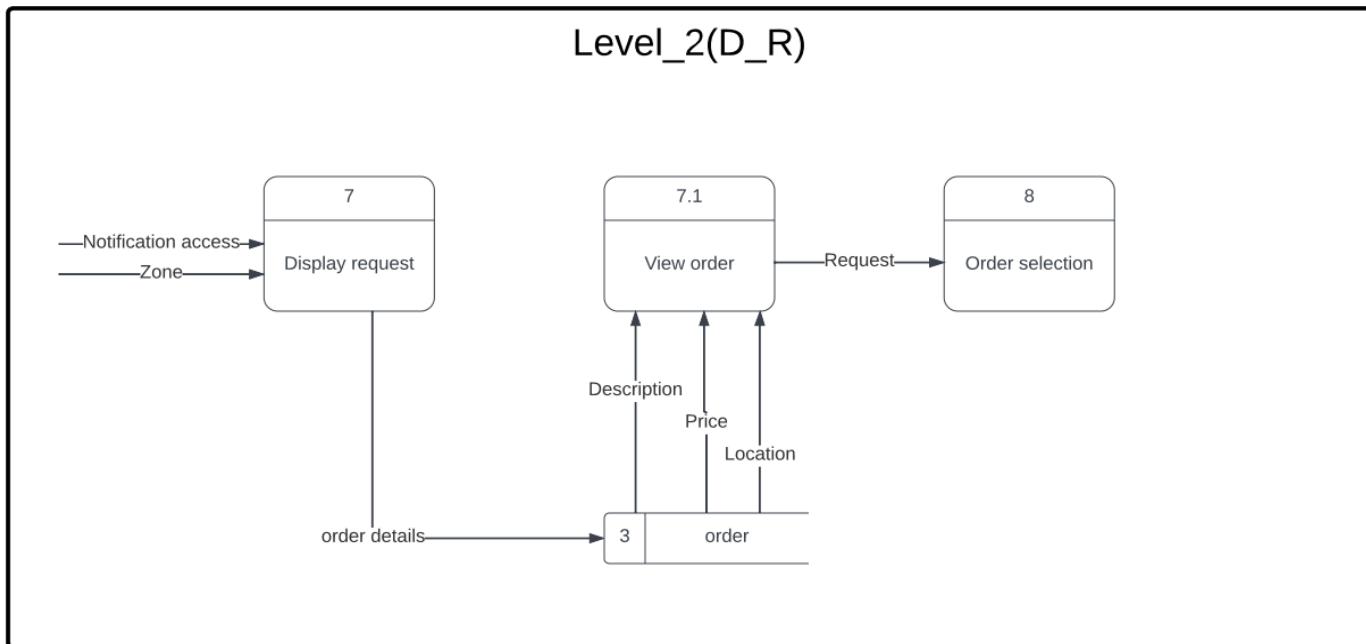
## Make Order Level 2



## Order Tracking Level 2



## Display Request Level 2



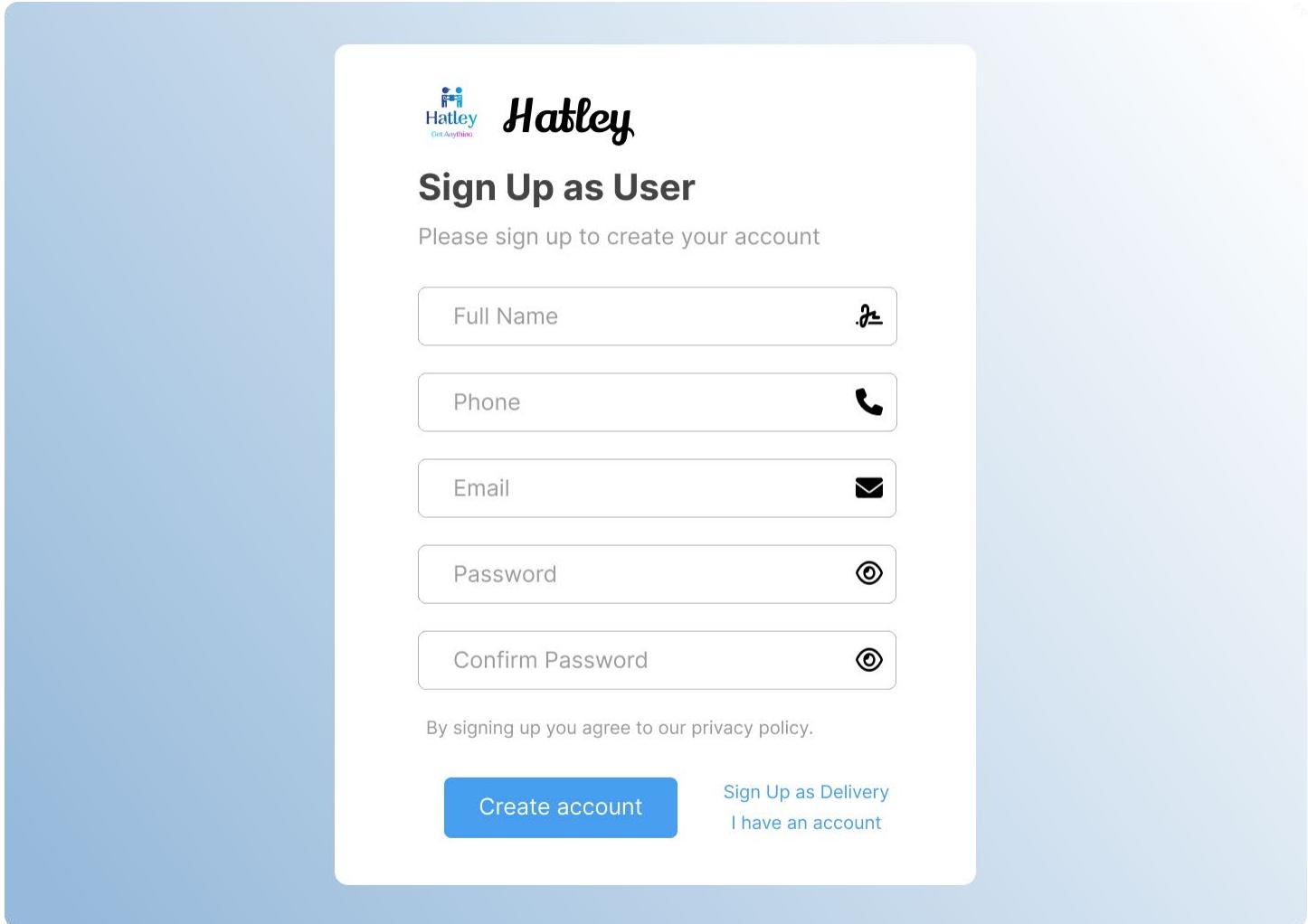
### 3.5 User Interface:

Embark on a journey of simplicity and efficiency with the "Hatley" User Interface. Designed for seamless interactions, our interface bridges the gap between users and delivery personnel, offering an intuitive and hassle-free experience. Join us as we explore the key features and functionalities that make "Hatley" a user-centric platform, redefining the world of delivery services.

Here are concise descriptions for each of the 21 screens in our "Hatley" app:

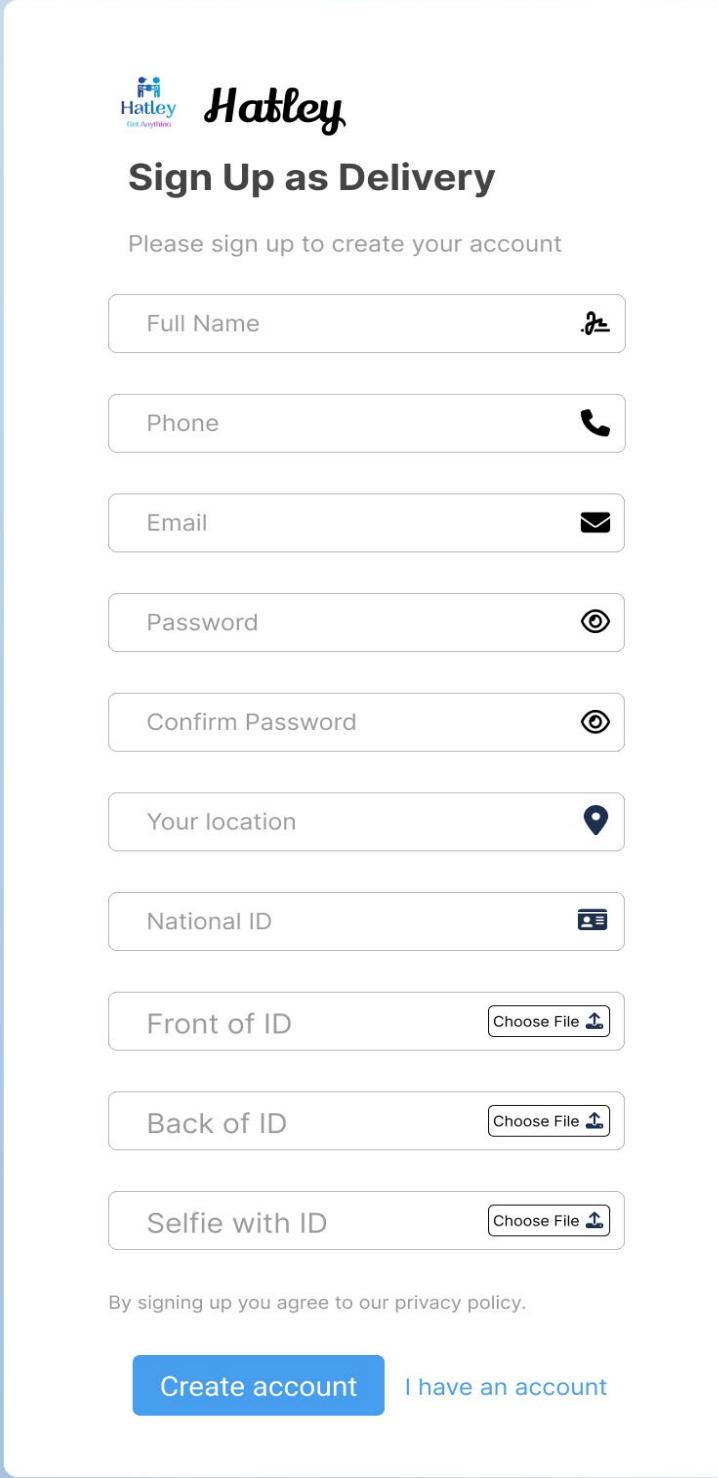
## 1. User Register Screen:

Capture essential user details, including name, email, phone, password, and confirmation for seamless registration.



## 2. Delivery Register Screen:

Gather user and delivery personnel information, including location, ID photos, and national ID number for secure and comprehensive registration.



The image shows a mobile application interface for 'Hatley' titled 'Sign Up as Delivery'. The screen has a light blue background with a white central card. At the top left is the Hatley logo ('Get Anything'). The title 'Sign Up as Delivery' is centered above a subtitle 'Please sign up to create your account'. Below are ten input fields: 'Full Name' (with a person icon), 'Phone' (with a phone icon), 'Email' (with an envelope icon), 'Password' (with an eye icon), 'Confirm Password' (with an eye icon), 'Your location' (with a location pin icon), 'National ID' (with a card icon), 'Front of ID' (with a 'Choose File' button and a file icon), 'Back of ID' (with a 'Choose File' button and a file icon), and 'Selfie with ID' (with a 'Choose File' button and a file icon). At the bottom, a note says 'By signing up you agree to our privacy policy.' followed by two buttons: a blue 'Create account' button and a smaller 'I have an account' link.

Hatley  
Get Anything

### Sign Up as Delivery

Please sign up to create your account

Full Name

Phone

Email

Password

Confirm Password

Your location

National ID

Front of ID

Back of ID

Selfie with ID

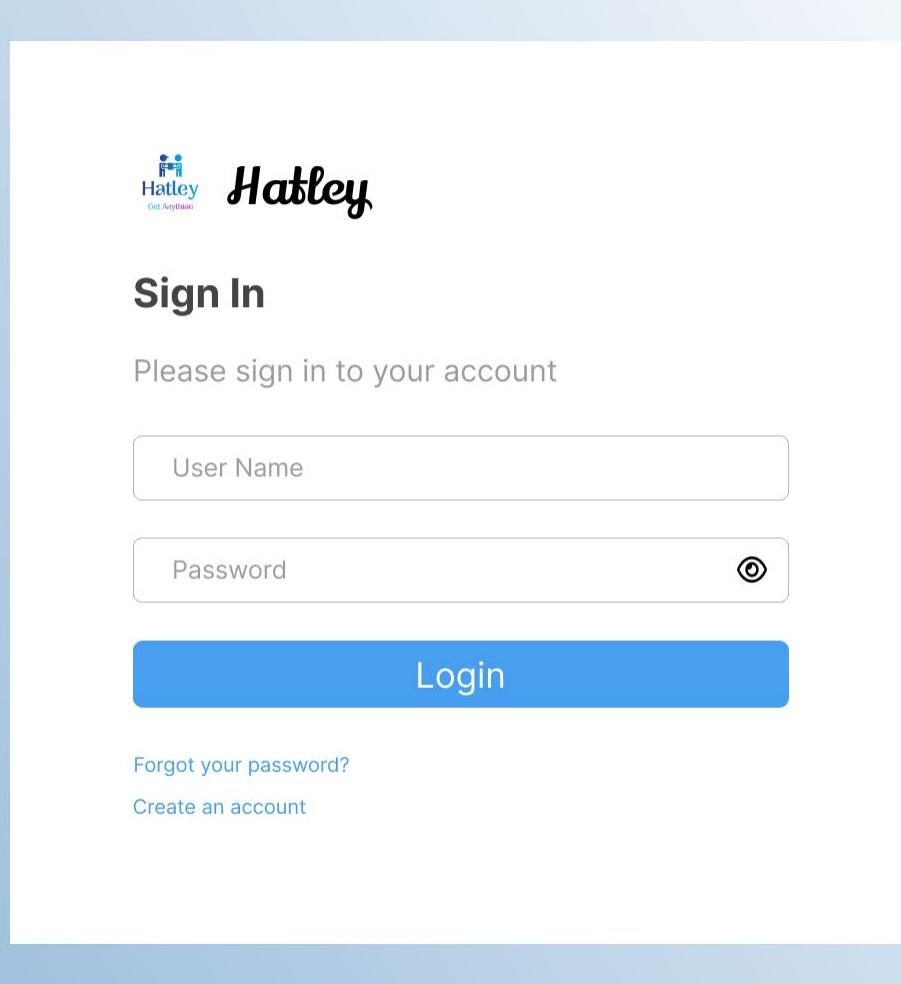
By signing up you agree to our privacy policy.

Create account

I have an account

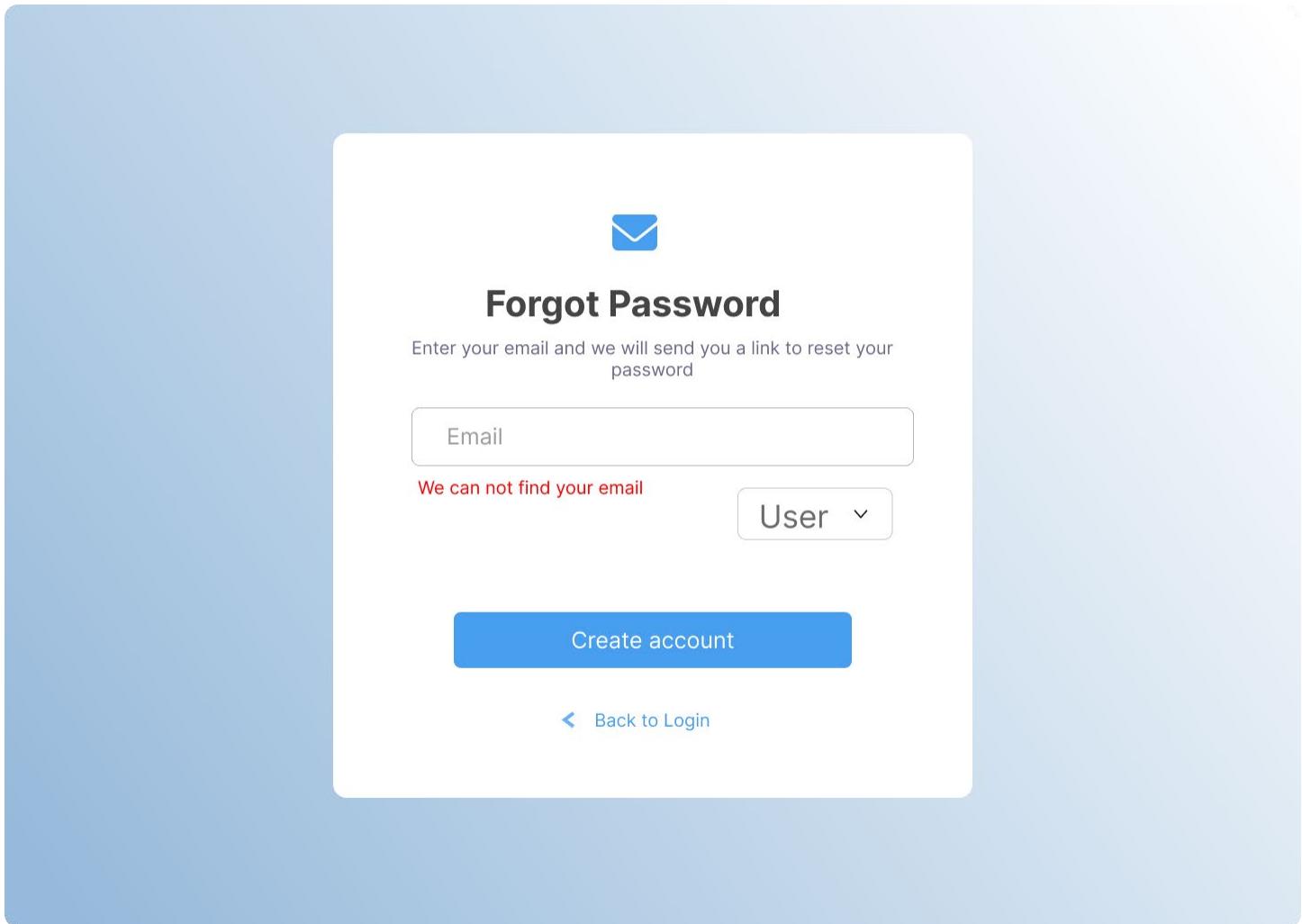
### 3. Login Page Screen

Provide a secure login interface on the Login Page Screen for users to access their accounts seamlessly.



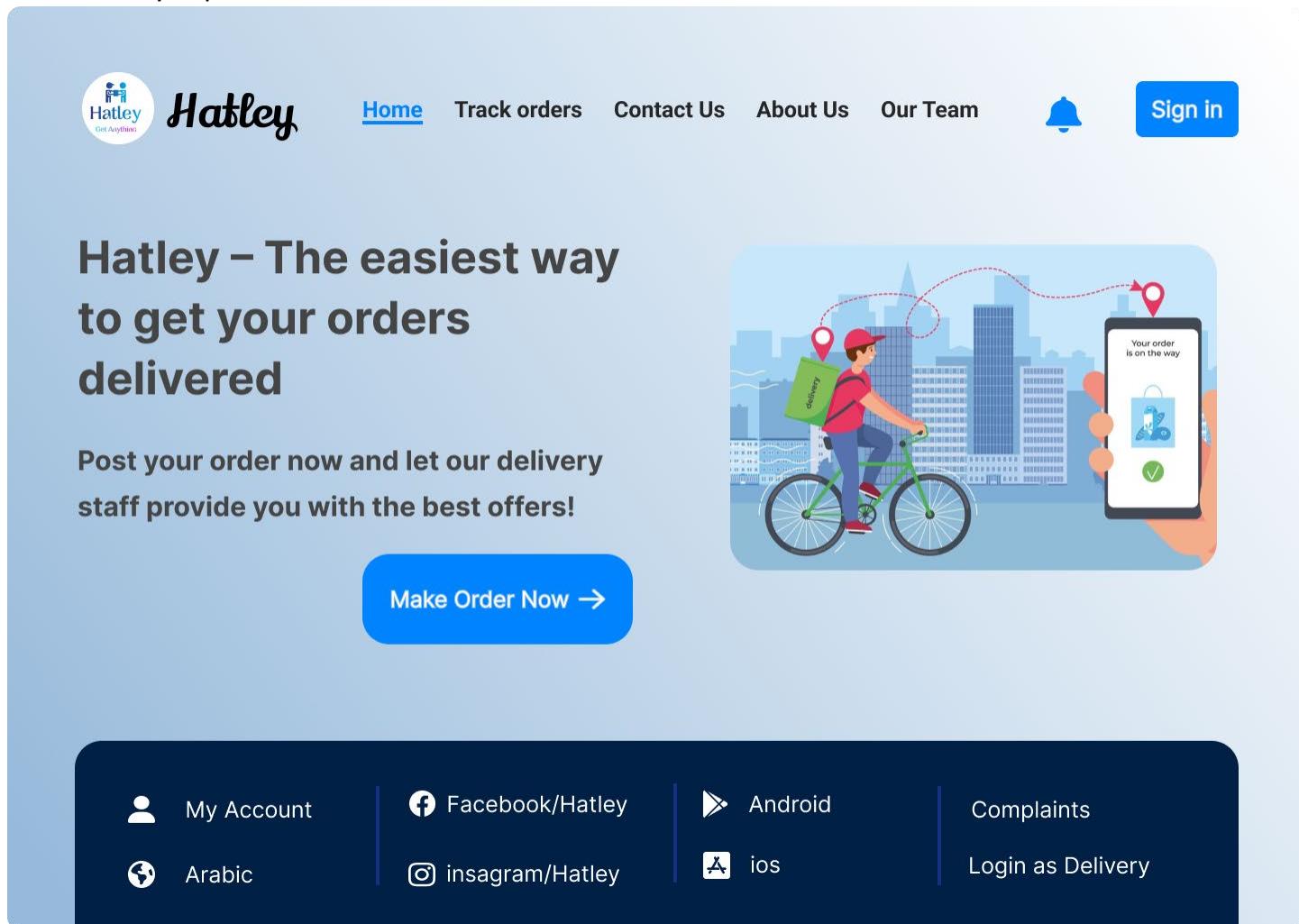
## 4. Forget Password Page Screen

Assist users in recovering forgotten passwords on the Forget Password Page Screen with a user-friendly interface.



## 5. Home Page Screen

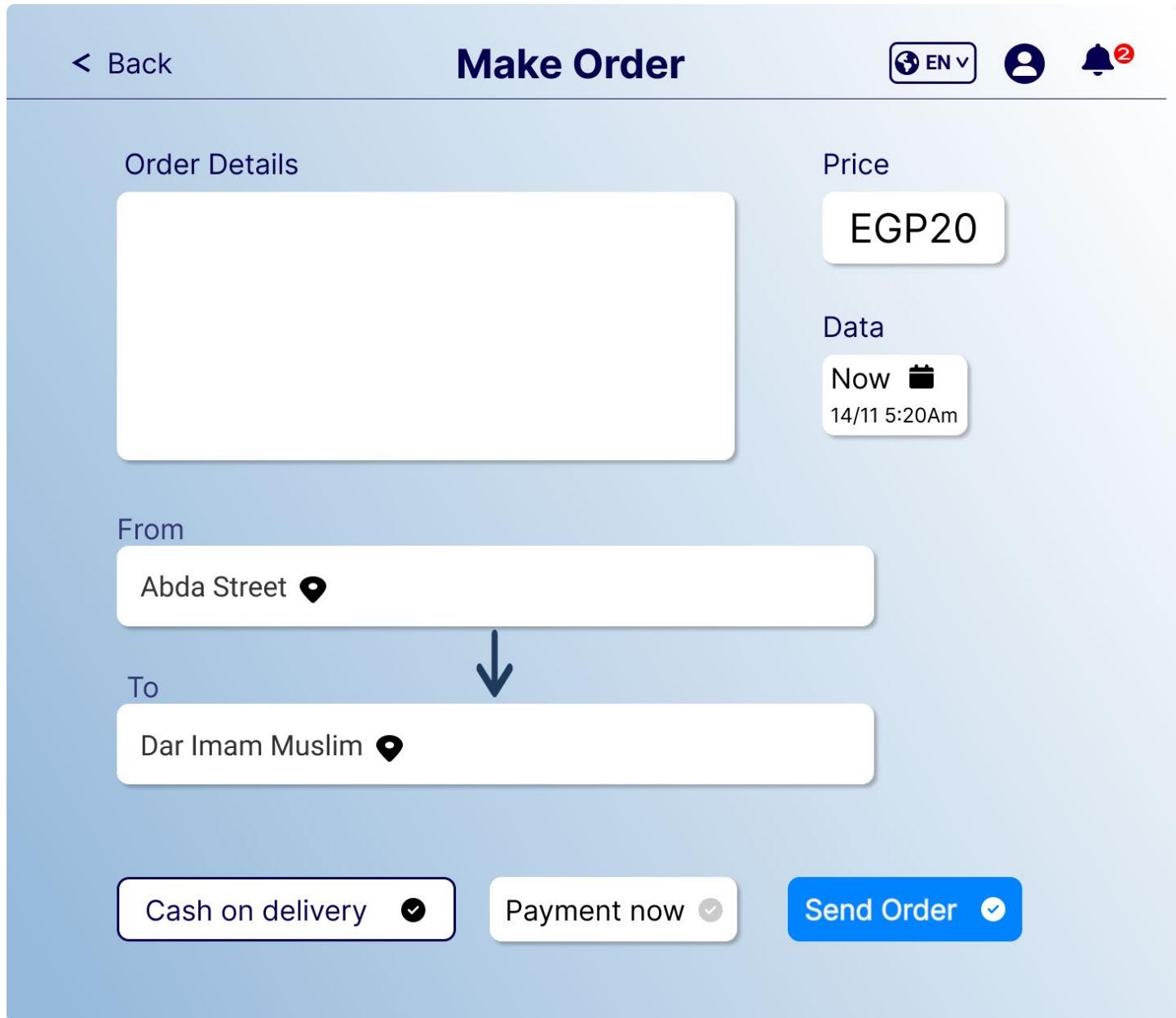
Feature a Home Page Screen with a navigation bar for seamless access to all app sections, ensuring user-friendly exploration.



The image shows the home page of the Hatley delivery app. At the top, there is a navigation bar with the Hatley logo, a bell icon, and a "Sign in" button. Below the navigation bar, the main content area features a large title: "Hatley – The easiest way to get your orders delivered". Below the title, a subtext encourages users to "Post your order now and let our delivery staff provide you with the best offers!". A prominent blue button labeled "Make Order Now →" is centered below the subtext. To the right of the text, there is a graphic illustration of a delivery person on a green bicycle carrying a yellow delivery bag, with a dashed line showing the route through a city skyline. Next to the illustration is a smartphone displaying a delivery tracking interface with the message "Your order is on the way". At the bottom of the page, a dark footer bar contains links for "My Account", "Facebook/Hatley", "Android", "Complaints", "Arabic", "Instagram/Hatley", "ios", and "Login as Delivery".

## 6. Make Order Screen

Facilitate the order creation process on the Make Order Screen with a textbox for order description, initial price, date, and location details.



## 7. My Current Orders Screen

Display ongoing order details on the My Current Orders Screen, including offers from delivery personnel with name, rate, price, time, and location.

The screenshot shows the 'My Current orders' screen for the Hatley app. At the top, there is a navigation bar with the Hatley logo, a user profile picture, and links for Home, Track orders, Contact Us, About Us, Our Team, language selection (EN), and account settings. The main title 'My Current orders' is centered above a list of four delivery offers.

**Offer 1:** EGP20 Now (14/11 5:20Am) Cash on delivery. Description: I need the book of lights from the office of Dar Imam Muslim to Abda Street and bring some papers as well from the Al-Awan library. From Dar Imam Muslim to Abda Street. Status buttons: Decline (grey) and Accept (green).

**Offer 2:** EGP25 3 min. 1,0 km. Delivery by Motorcycle by Salah Ali (4.6 rating). Status buttons: Decline (grey) and Accept (green).

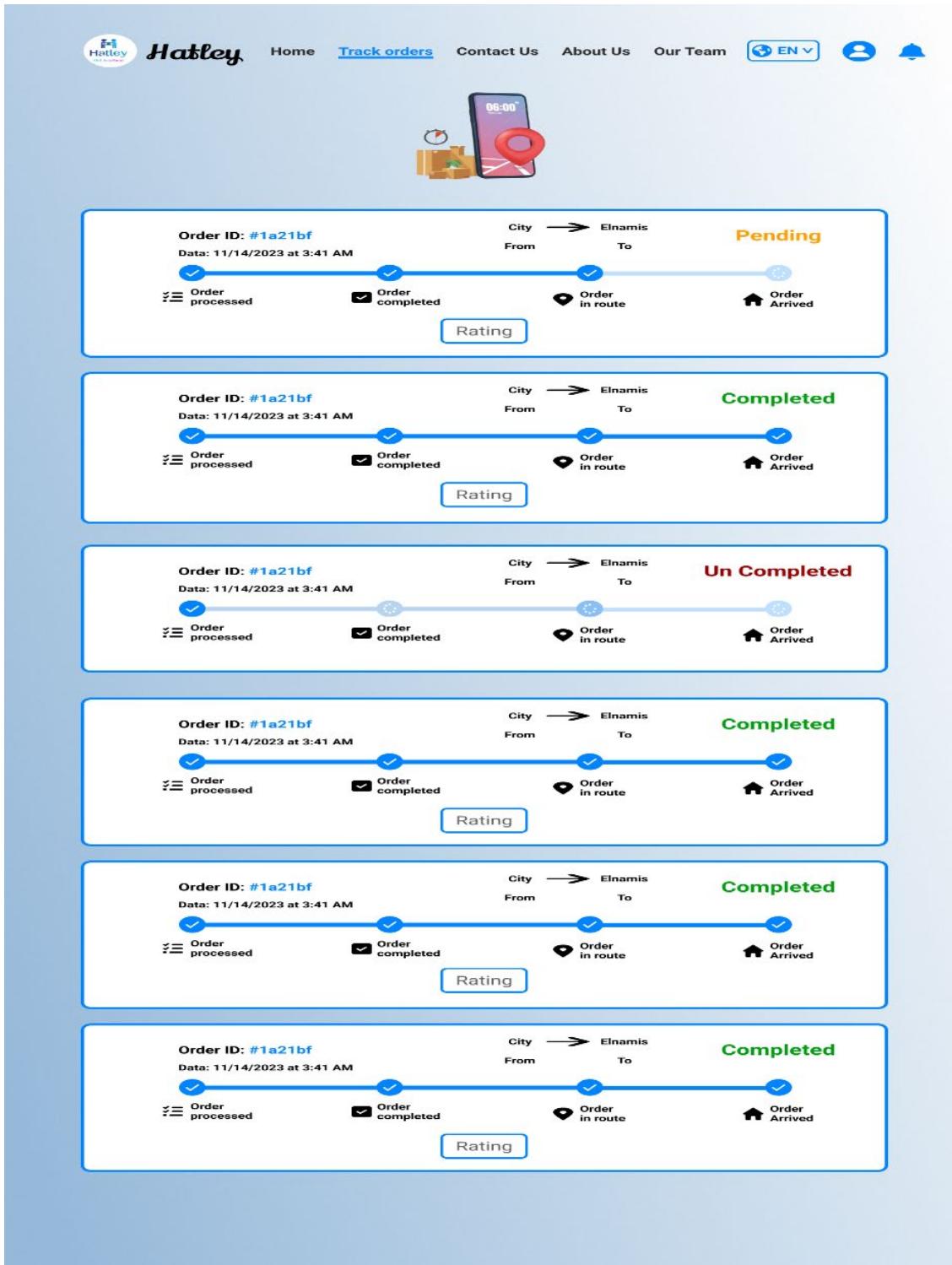
**Offer 3:** EGP20 6 min. 1,5 km. Delivery by Motorcycle by Ahmed Mohamed (4.0 rating). Status buttons: Decline (grey) and Accept (green).

**Offer 4:** EGP30 10 min. 1,9 km. Delivery by Motorcycle by Salem Mahmoud (4.9 rating). Status buttons: Decline (grey) and Accept (green).

**Offer 5:** EGP20 6 min. 1,5 km. Delivery by Motorcycle by Ahmed Mohamed (4.0 rating). Status buttons: Decline (grey) and Accept (green).

## 8. Tracking Page Screen

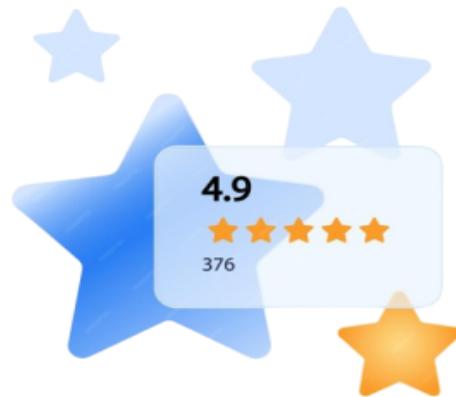
Allow users to track order status on the Tracking Page Screen with four steps: Order Processed, Order Completed, In Route, and Order Arrived. Includes a button for order review.



## 9. Rating Order Screen

Allow users to rate their orders using stars and provide additional feedback through a textbox on the Rating Order Screen.

# Rate & Review



How do you rate your experience?

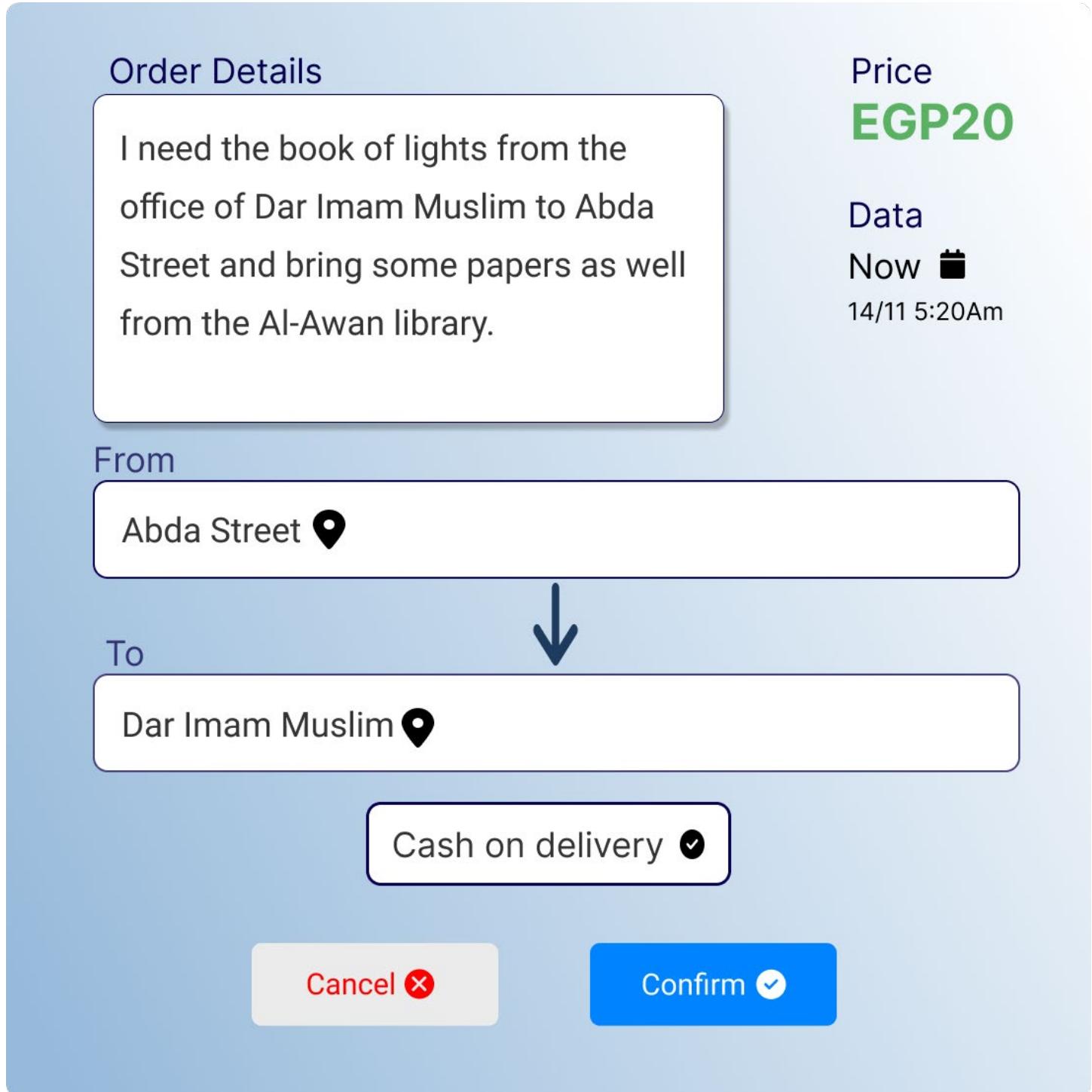


Enter your review

Add review

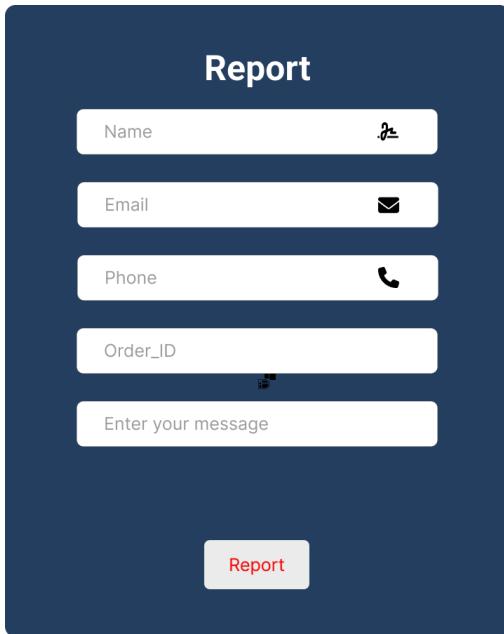
## 10. Order After Confirmation Screen

Notify users that their order has been successfully confirmed on the Order After Confirmation Screen with a loading page.



## 11. Report Page Screen

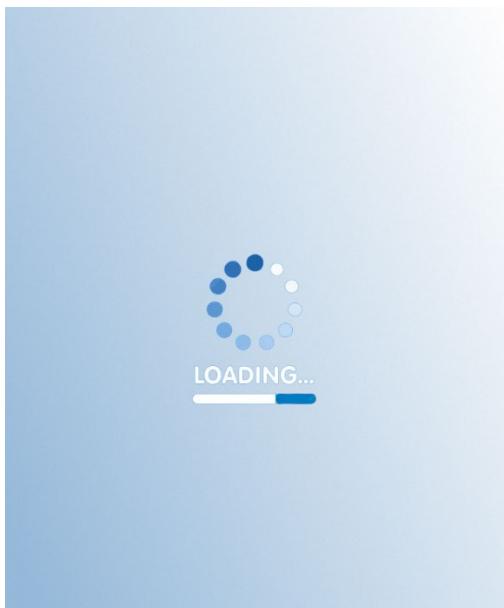
Display user and order details on the Report Page for efficient reporting, featuring name, email, phone, order ID, and a message.



A dark blue rectangular form titled "Report". It contains five input fields with icons: "Name" (person icon), "Email" (envelope icon), "Phone" (phone receiver icon), "Order\_ID" (barcode icon), and a larger "Enter your message" field. Below these fields is a red "Report" button.

## 12. Loading Page Screen

Show a loading screen during transitions or data retrieval on the Loading Page Screen to enhance user experience.



## 13. Previous Orders Screen

Showcase order history on the Previous Orders Screen, including status, user ratings for deliveries, and detailed order information.

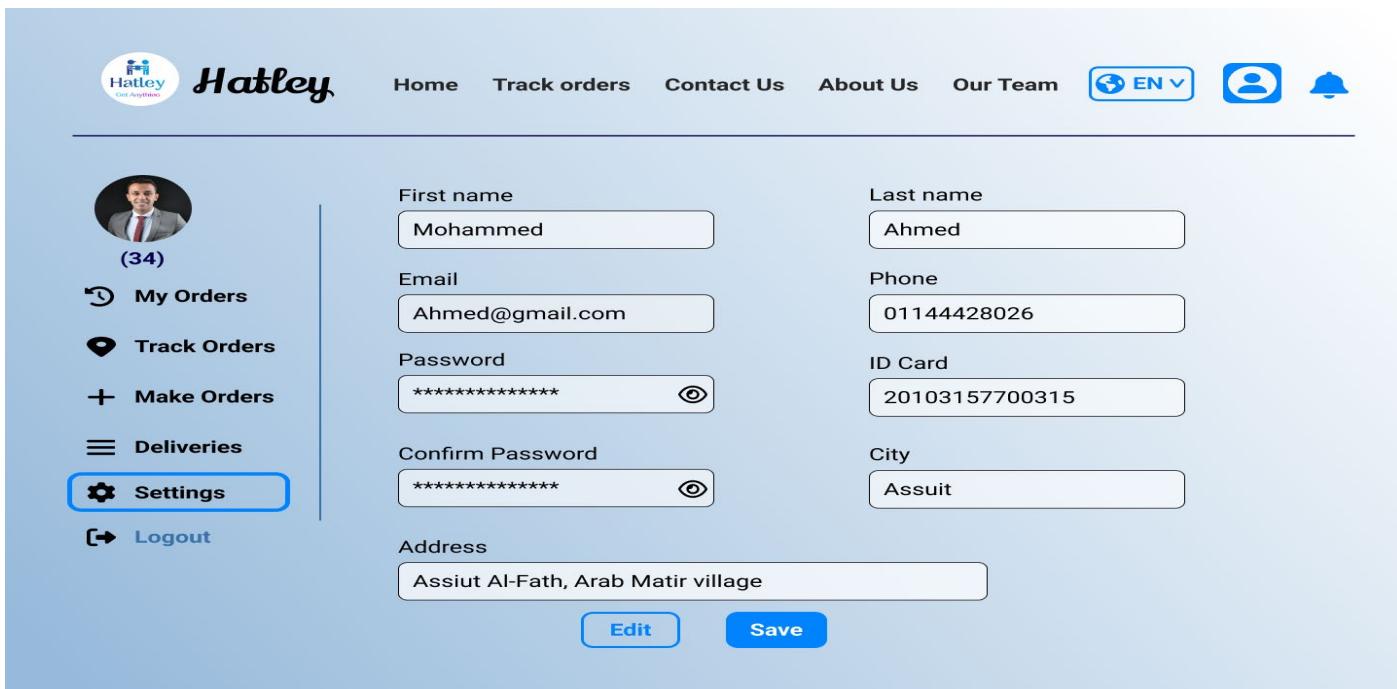
The screenshot displays the 'Previous orders' section of the Hatley application. At the top, there is a navigation bar with the Hatley logo, a user profile icon with '(34)', and links for Home, Track orders, Contact Us, About Us, Our Team, a language switcher (EN), and user account options. On the left, a sidebar menu includes My Orders, Track Orders, Make Orders, Deliveries (which is selected and highlighted in blue), Settings, and Logout.

The main area is titled 'Previous orders' and shows a grid of nine completed delivery cards. Each card contains the following information:

- User Profile:** Mohamed Salam (5 stars)
- Order ID:** #1a21bf
- Date:** 11/14/2023 at 3:41 AM
- Price:** 25\$
- Status:** Completed
- Delivery Details:** City → Elnamis  
From To
- Details:** Lorem Ipsum has been the industry's standard dummy text ever since
- Actions:** Report (red button) and Re-order (blue button)

## 14. Profile Page Screen

Present user details on the Profile Page, including name, email, phone, and address for easy management and personalization.



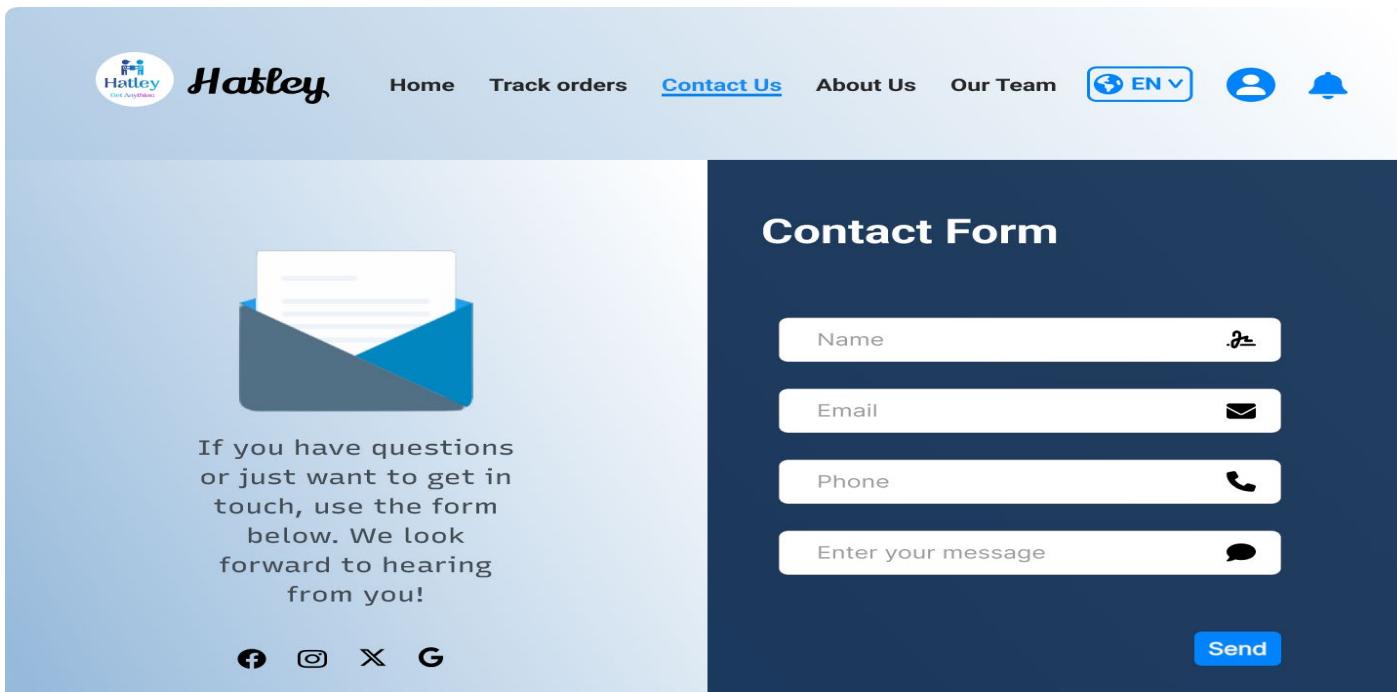
The screenshot shows the profile page of a mobile application. At the top, there is a header with the app logo 'Hatley Get Anything', navigation links 'Home', 'Track orders', 'Contact Us', 'About Us', 'Our Team', and language settings 'EN'. Below the header is a sidebar with user profile information: a circular profile picture of a man, '(34)' indicating friend count, and several menu items: 'My Orders', 'Track Orders', 'Make Orders', 'Deliveries', 'Settings' (which is selected and highlighted in blue), and 'Logout'. The main content area displays user details in a grid format:

First name	Mohammed	Last name	Ahmed
Email	Ahmed@gmail.com	Phone	01144428026
Password	*****	ID Card	20103157700315
Confirm Password	*****	City	Assuit
Address	Assuit Al-Fath, Arab Matir village		

At the bottom of the form are two buttons: 'Edit' and 'Save'.

## 15. Contact Us Screen

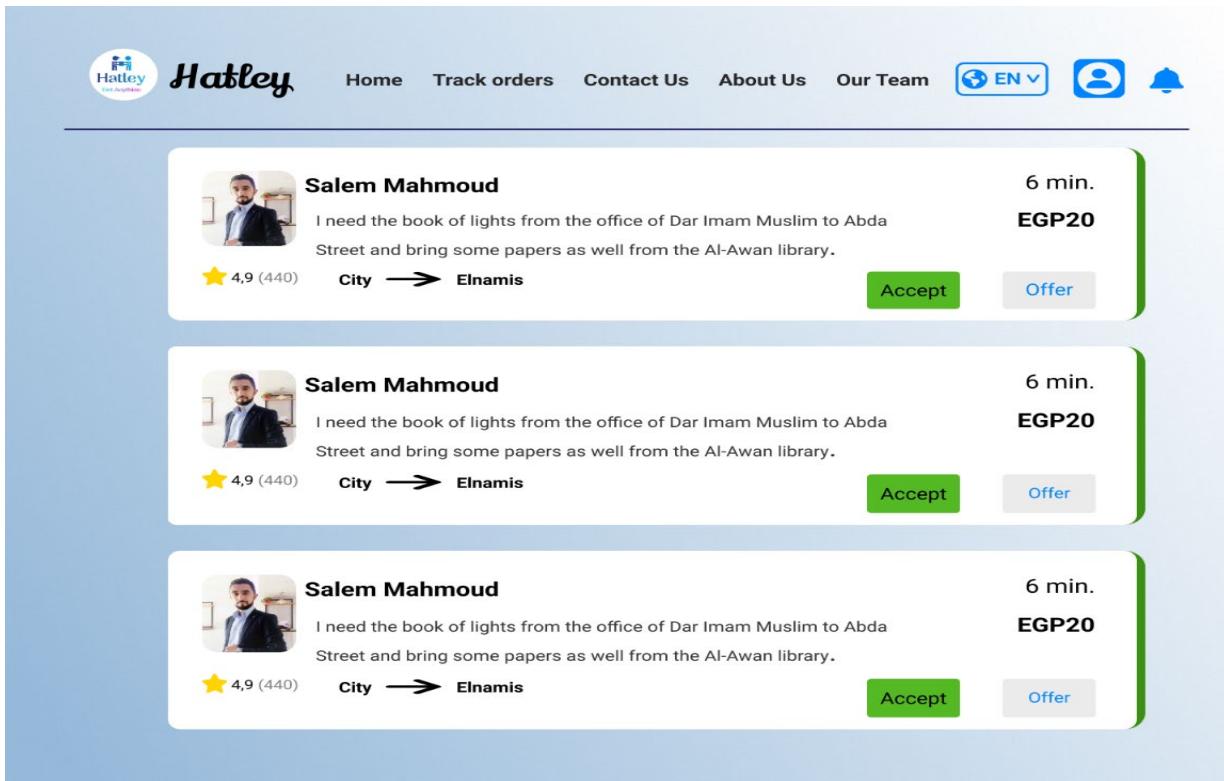
Offer a Contact Us Screen with a contact form for users to reach the app management, including fields for name, email, phone, and a message.



The screenshot shows the contact us screen of the mobile application. At the top, it features the app logo 'Hatley Get Anything', navigation links 'Home', 'Track orders', 'Contact Us' (which is selected and highlighted in blue), 'About Us', 'Our Team', and language settings 'EN'. The main content area has two sections: a light blue section on the left containing a large envelope icon and a message encouraging users to use the contact form, and a dark blue section on the right titled 'Contact Form' containing a form with four fields: 'Name', 'Email', 'Phone', and 'Enter your message', each with a corresponding icon (person, envelope, phone, speech bubble). A 'Send' button is located at the bottom right of the form.

## 16. Notification Page Screen

Notify users of delivery offers with a user-friendly display similar to the Delivery Offer Page Screen.



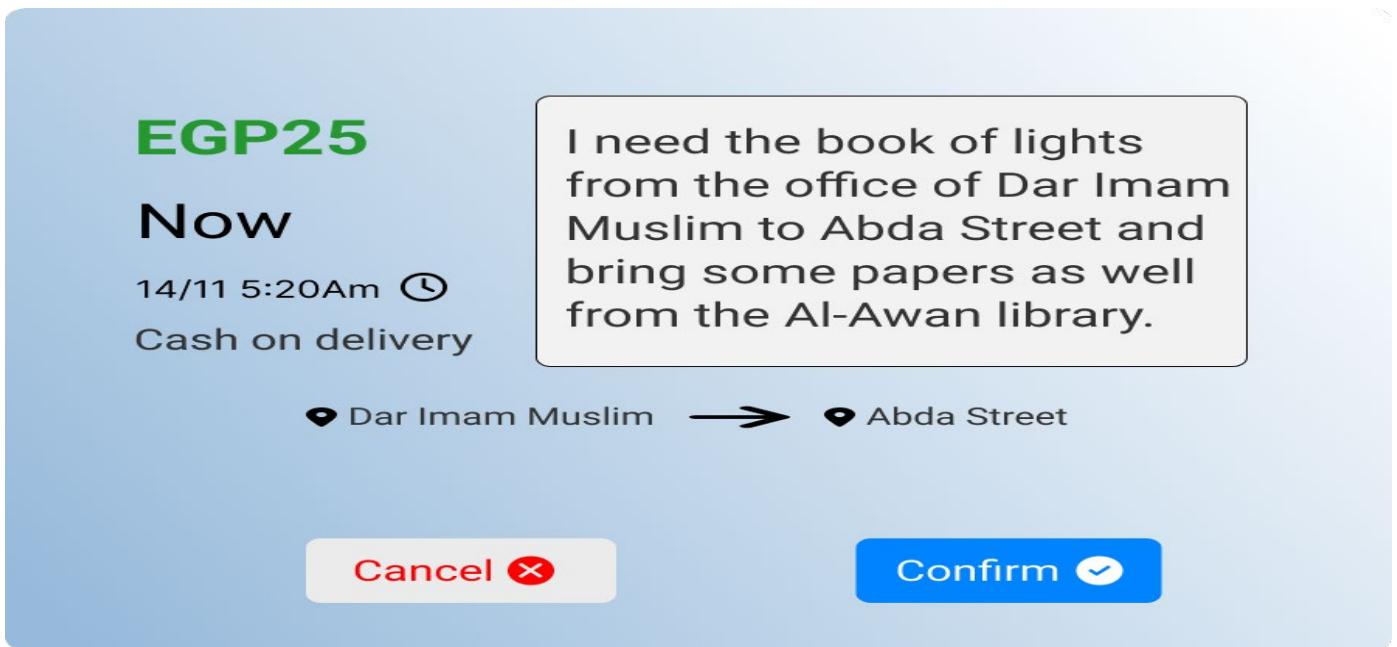
## 17. Delivery Offer Page Screen

Enable delivery personnel to send offers with details such as price, time, and a message to the user on the Delivery Offer Page Screen.



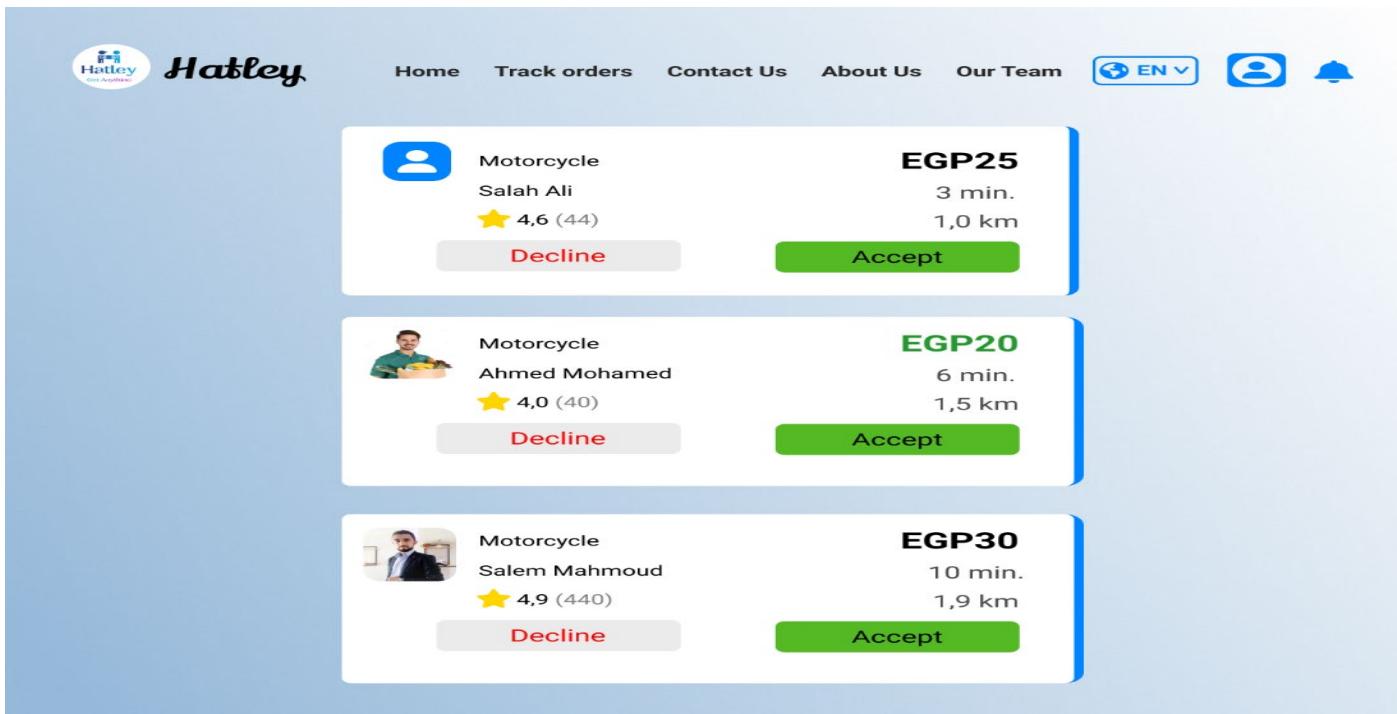
## 18. Order Confirmation Screen

Confirm order details such as description, price, time, payment method, date, and locations on the Order Confirmation Screen. Includes Cancel and Confirm buttons for user convenience.



## 19. Orders Page for Delivery Men Screen

Display ongoing orders for delivery personnel on the Orders Page Screen, including details such as time, initial price, and location.



## 20. Our Team Screen

Introduce the development team on the Our Team Screen, showcasing names, roles, and contact information for transparency and user connection.

Name	Name	Name	Name	Name
Position	Position	Position	Position	Position
LinkedIn	LinkedIn	LinkedIn	LinkedIn	LinkedIn

## 21. About Us Page Screen

Provide an About Us Page Screen with a brief description of the development team and an overview of the app.

**About Us**

At Hatley, we're all about delivering convenience to your doorstep. We understand that time is precious, and that's why we've created a platform that puts the power in your hands. You can place your order, choose from a variety of delivery offers, and enjoy a hassle-free delivery experience. Your satisfaction is our priority, and we're here to make your life easier, one delivery at a time.

## 3.6 Testing and Quality Assurance

### 1. Testing Strategy:

- a) Develop a comprehensive testing strategy encompassing both manual and automated testing approaches.
- b) Lay the groundwork for automated testing frameworks to be implemented as development progresses.

### 2. Testing Types:

#### a) Functional Testing:

Design detailed test cases for critical functionalities like user registration and order creation.

#### b) Usability Testing:

Prepare usability testing scenarios to evaluate the user interface for ease of use and navigation.

#### c) Performance Testing:

Outline plans to simulate application performance under various conditions, anticipating future development.

### 3. Testing Documentation:

- a) Create detailed test cases with inputs, expected outputs, and step-by-step instructions for each screen's expected behavior.
- b) Develop comprehensive test scenarios to validate the robustness of the application.

### 4. Quality Assurance:

- a) Plan for code reviews once development begins, ensuring alignment with project requirements and coding standards.
- b) Establish coding standards in advance to maintain consistency in the codebase.

### 5. Collaboration with Development:

Maintain regular communication with the development team to understand progress and adjust testing plans accordingly.

### 6. Tool Selection:

Identify and select testing tools suitable for manual testing, test case management, and potential future automated testing needs.

### 7. Feedback Loop:

Anticipate an iterative process, adapting testing and quality assurance strategies based on the evolving project.

**8. User Acceptance Testing (UAT):**

- a) Outline plans for involving end-users in testing once the application is ready for user acceptance testing.
- b) Define user acceptance criteria for effective testing.

**9. Security Considerations:**

- a) Consider potential security concerns, even at this early stage.
- b) Plan for security testing to be incorporated as development advances.

**10. Documentation Review:**

Ensure that testing documentation aligns seamlessly with project requirements and UI/UX designs.

**11. Training:**

Provide training to the testing team on the testing strategy, tools, and methodologies they will employ throughout the project.

# 4. Software Requirements Specification

## 4.1 Introduction:

Think of this document as your project's handbook, guiding you through all the important details about what we're building and why it matters.

### 4.1.1 Intended Audience and Reading Suggestions:

This document is for everyone involved in the project, project manager, development team, and quality assurance testers. It gives insights into project requirements and scope, catering to different needs within the project lifecycle.

#### Reading Suggestions:

- Clients are encouraged to review the entire document to understand project requirements and scope thoroughly.
- Developers should focus on sections detailing functional requirements, system architecture, and data design.
- Testers should pay attention to sections outlining system features, testing procedures, and requirements for validation and verification.

### 4.1.2 Product Scope:

The product scope defines the boundaries of the software project and outlines the features and functionalities to be included in the final product.

#### The project scope encompasses:

- Essential features and functionalities necessary to achieve project objectives.
- Exclusions and limitations that clarify what is not within the project scope.
- Interfaces with external systems or components essential for software operation.
- Constraints, assumptions, or dependencies that may impact software development or implementation.

## 4. 2 Overall Description:

Hatley, a cutting-edge online platform and mobile application, has been meticulously crafted to revolutionize the process of procuring essential supplies. This innovative solution is dedicated to ensuring that individuals can acquire their needed items with utmost ease, expediency, and uncompromised quality. By leveraging the intuitive interface of the website or mobile app, users are empowered to effortlessly submit requests for their desired supplies, fostering a seamless and user-centric experience.

However, Hatley is not just a facilitator of supply requests; it functions as a multifaceted ecosystem that transcends conventional transactional platforms. In addition to serving as a conduit for streamlined supply acquisition, Hatley doubles as a dynamic marketplace that opens avenues for meaningful employment opportunities. Users can actively participate in this symbiotic marketplace by submitting compelling offers in response to posted requests, creating a marketplace where demand seamlessly intersects with a diverse pool of services.

### 4.2.1 Product Perspective:

The core ethos of Hatley lies in minimizing the intricacies associated with obtaining essential supplies, ensuring that users can navigate the platform with unparalleled efficiency. Beyond this, the platform stands as a beacon for economic empowerment, providing a space where individuals can not only meet their immediate supply needs but also capitalize on their skills and resources to seize lucrative job opportunities.

In essence, Hatley stands at the forefront of digital innovation, weaving together the threads of convenience, speed, and quality to redefine the landscape of supply acquisition. By fostering a collaborative marketplace, it not only addresses immediate needs but also acts as a catalyst for economic growth and empowerment in the ever-evolving digital realm.

### 4.2.2 Product Functions:

Hatley provides a variety of customized functions to accelerate user needs and facilitate the submission of offers by service providers. **At the forefront of its capabilities, several key functions stand out of utmost importance:**

1. Create and publish order posts complete with relevant details.

2. Choose the preferred offer from the available options.
3. Monitor and track order status in real time.
4. Provide feedback on the quality of the delivery service.
5. Effectively submit competitive offers in response to posted requests.
6. Provide feedback on users.
7. Enjoy flexibility in payment options, including cash or Visa card transactions.
8. Allow automatic or manual determination of the user's location.

#### 4.2.3 User Classes and Characteristics:

Hatley is designed for two categories of users: the consumer category and the service provider category. The consumer utilizes the platform to obtain his needs by posting the details of the order. The service provider category can obtain a job as a service connector. He accomplishes this by submitting offers for published orders and subsequently accepting the most suitable one.

#### 4.2.4 Operating Environment:

The Hatley system is designed to work on software platforms. It works consistently across a variety of web browsers (Chrome, Firefox, etc.). The software components and tools used to develop the project include visual studio 2022 with asp.net core 7 web Api, react Js, react native version 0.73.4. The minimum required Android version for the application is 18.2.0. The application works on iOS devices as well. The database used for the project is Hatley database based on Microsoft SQL Server.

#### 4.2.5 Design and Implementation Constraints:

The Hatley system will be subject to some design and implementation constraints, including:

- Compatibility constraints as Hatley project compatible with at least Android version 18.2.0 for the application.
- User security and privacy, the user may enter the Visa card number and password.
- The service provider must be at least 18 years old.

- The needs that are ordered must be obtained in a manner consistent with applicable legal regulations and standards.
- Regular and continual updates are essential to maintain the quality and efficiency of the system.

#### 4.2.6 Assumptions and Dependencies:

The system functions based on a few assumptions and dependencies like:

- Reliable internet connectivity to support real-time notifications for new orders posted and communication between the service provider and user.
- In the event that a user submits orders for needs that are legally prohibited on more than three times, the system will initiate the deletion of their account as a consequence.

It is imperative that the server remains consistently operational without any instance of shutdown, ensuring uninterrupted availability and continuous accessibility for users and associated services.

### 4.3 External Interface Requirements:

#### 4.3.1 Hardware Interfaces:

The sole essential hardware needed for the functionality of the system includes either a mobile phone or a personal computer, facilitating access to the system through either the designated website or application.

#### 4.3.2 Software Interfaces:

The mobile application demonstrates compatibility with both Android and iOS devices, ensuring versatility and accessibility across a wide range of platforms. Furthermore, it is noteworthy that the system operates independently, exhibiting a lack of dependency on any other external systems. The system incorporates the Signa IR framework to facilitate the seamless delivery of real-time notifications.

#### 4.3.3 Communications Interfaces:

The Hatley system incorporates specific communication requirements, which include:

- The system is designed to support the HTTP protocol, ensuring compatibility and effective communication over the web.
- The system employs API (Application Programming Interface) technology as a fundamental mechanism to initiate requests and facilitate the seamless exchange of information between different software components, enabling efficient and standardized communication within its architecture.

- It consistently operates effectively on a diverse array of web browsers, including but not limited to Chrome, Firefox, Safari, and others.
- The system accommodates email addresses following the format "[xxxxxxxx@your-website.com](mailto:xxxxxx@your-website.com)", allowing users to use diverse email domains without mandating a specific type of website.

## 4.4 System Features:

Here's a summary of the application's features:

### For Users:

1. Fair Competition: The app enables delivery service providers to compete with each other in a fair and clean manner, contributing to the improvement of service quality.
2. Competitive Offers: Delivery providers can offer competitive deals and discounts to users through private and public messages.
3. Provider Details: The app offers comprehensive details about each delivery service provider, including their ratings, history, and customer reviews.
4. Automatic Location: The app automatically determines the provider's location for deliveries, with the option to change it if necessary.
5. Multiple Payment Options: The app offers various payment methods for users to meet their preferences, including cash, E-wallet and Visa payment.
6. Rating and Reviews: Users can provide ratings and reviews for the delivery service providers after the delivery is complete.
7. Order Tracking: Users can directly track their orders to know the delivery progress.
8. Encouraging Improvement: The app encourages service providers to enhance their skills and improve the quality of their services to attract more orders.
9. Complaint Center: The app provides a platform for users to submit complaints and share feedback with the service providers.
10. Pre-booking: The application provides the possibility of making an early order and setting a time for its arrival.

### For Delivery Providers:

1. Order Notifications: Delivery providers receive notifications about new delivery orders.

2. Order Status: Delivery providers can update the order status, keeping users informed about the progress.
3. Complaint Submission: Providers have the option to submit complaints if needed.
4. Payment Options: Providers can receive funds either through bank accounts or electronic wallets.

## 4.5 Other Nonfunctional Requirements:

### 4.5.1 Performance Requirements:

The website should load quickly and provide good performance to ensure a smooth user experience, even with a large number of users.

Performance can be improved by reducing loading times, optimizing images, and utilizing caching.

The system supports a large number of concurrent users at all other times.

The database should be well-structured and optimized for fast response times.

Scalability: The website must be continuously scalable to accommodate the increase in the number of users and the volume of data.

Availability: The website must be available 24/7/365 and maintain a high level of uptime.

Contingency plans and disaster recovery strategies should be in place in case of website failure.

### 4.5.2 Safety Requirements:

The Website is mandated to adhere to globally recognized web standards, including those set forth by the World Wide Web Consortium (W3C) and the Web Content Accessibility Guidelines (WCAG). Compliance with these international standards ensures that the site is designed and developed according to best practices, enhancing accessibility to a diverse audience and promoting inclusivity by meeting standards set by the W3C and WCAG for an enhanced user experience.

### 4.5.3 Security Requirements:

1. The website must include strong security measures aimed at ensuring a high level of protection for sensitive information. This includes preventing security breaches and unauthorized access, with a focus on protecting user data through comprehensive security protocols.

2. User authentication and identity verification mechanisms must be implemented with an emphasis on security, ensuring the integrity and confidentiality of user identities. The focus is on creating secure ways to confirm and validate user credentials.
3. Strict information security measures must be implemented to protect personal data and sensitive information. The focus is on implementing strict protocols to protect against potential risks and maintain the confidentiality and integrity of user data.

#### 4.5.4 Software Quality Attributes:

1. The system is required to exhibit a user-friendly interface, ensuring not only ease of use but also fostering a clear and understandable user experience. This includes intuitive navigation and comprehensibility for users interacting with the system.
2. The system is independent, allowing it to work seamlessly across different operating systems and environments. Furthermore, it works consistently across a variety of web browsers, ensuring a uniform and reliable user experience regardless of the chosen platform.
3. One notable aspect of system design is its potential for future maintenance and development. Facilitating continuous adaptability and ease of modification to accommodate evolving requirements, technologies and user needs.
4. The software must undergo comprehensive testing and verification processes to ensure high levels of reliability and stability in its performance.
5. Access to the system is available to administrators and other users; however, the level of access granted to each user is tailored based on their specific responsibilities and tasks within the system.

#### 4.5.5 Business Rules:

The Website is mandated to adhere to globally recognized web standards, including those set forth by the World Wide Web Consortium (W3C) and the Web Content Accessibility Guidelines (WCAG). Compliance with these international standards ensures that the site is designed and developed according to best practices, enhancing accessibility to a diverse audience and promoting inclusivity by meeting standards set by the W3C and WCAG for an enhanced user experience.

## 5. Conclusion:

The conclusion of Hatley's documentation is a summary of the key findings and insights obtained from the four chapters of the documentation. These chapters have provided a comprehensive overview of the proposed software, from its initial proposal to the software requirements specification.

In the first chapter, the software proposal, the overall concept of the Hatley software was introduced, and the key objectives were outlined. Hatley is not just a facilitator of supply requests; it functions as a multifaceted ecosystem that transcends conventional transactional platforms. In addition to serving as a conduit for streamlined supply acquisition, Hatley doubles as a dynamic marketplace that opens avenues for meaningful employment opportunities. The core ethos of Hatley lies in minimizing the intricacies associated with obtaining essential supplies, ensuring that users can navigate the platform with unparalleled efficiency.

The second chapter, the software analysis, delved deeper into the requirements and constraints of the software. A thorough analysis of the market and competitors was conducted, and user requirements and constraints were identified. The results of the analysis provide important insights into the key features and functionalities that need to be included in the software to make it appealing to users and competitive in the market. This chapter also included a potential challenges and limitations of the software, and how these could be addressed in the design and development process.

The third chapter, the software design, detailed the design of the software, including the user interface, system architecture, and the data storage model. The design was created in such a way as to meet the requirements outlined in the analysis. While also being aesthetically pleasing and easy to use. The design also took into account the technical constraints of the software, to ensure that it will be feasible to develop and implement.

The final chapter, the software requirements specification, provided a comprehensive description of the functional and non-functional requirements of the software. This chapter served as a contract between the development team and stakeholders, outlining the product's intended behavior, constraints, and features. It provided a clear understanding of the requirements, constraints, and objectives for the Hatley software, and will serve as the basis for design, development, testing, and delivery.

In conclusion, the four characters of Hatley documentation have provided a comprehensive overview of the proposed software and have set the foundation for its development. The software proposal, analysis, design, and requirements specification have all been carefully crafted to ensure that the end product meets the expectations of stakeholders and end-users. This documentation will serve as a guide for its development and success. The development team is confident that the software will be a success, and that it will be widely adopted by users around the world.



The ProtoType design tailored for the Hatly project has been developed and creatively integrated into the Figma platform