



Software Engineering Department  
ORT Braude College

Capstone Project Phase A – 61998

## **Flying Packages**

23-1-D-29

Advisor: Zeev Barzily

Student 1: Mosa Srour-205798812

E-mail 1: [mosa.srour.s9@gmail.com](mailto:mosa.srour.s9@gmail.com)

Student 2: Wessam Ghrayeb-211756820

E-mail 2: [wesamghrayeeb1@gmail.com](mailto:wesamghrayeeb1@gmail.com)

GIT Link: <https://github.com/MousaSrour/FlyingPackages.git>

# Content

<b>1.Introduction</b>	3
<b>2.Background and Related Work</b>	4
2.1 Nearest Neighbor Algorithm [1]	4
2.2 Genetic Algorithm [2]	4
2.3 Machine Learning Algorithms [3]	5
2.4 Ant Colony Optimization [4]	5
2.5 Delivery management apps	6
<b>3. Expected Achievements</b>	7
3.1 Outcomes	7
3.2 Unique Features	7
3.2.1 Finding the nearest courier	7
3.2.2 GPS Tracking	7
3.2.3 Alerts and Notifications	7
3.3 Criteria for Success	8
<b>4. The Process</b>	9
4.1 Research – Delivery Management	9
4.1.1 Constraints and Challenges – Delivery Management	10
4.2 Methodology and Development Process	11
4.2.1 Planning and preparation	11
4.2.2 Implementation	11
4.2.3 Iterative development	11
<b>5. Product</b>	12
5.1 Requirements	12
5.2 UML Diagrams	13
5.2.1 Use Case Diagram	13
5.2.2 Class Diagram	14
5.2.3 Activity Diagram	16
5.3 User Interface	18
<b>6. Verification and Evaluation</b>	30
6.1 Unit Tests	30
6.2 Functional Tests	31
<b>7. References</b>	32

## **Abstract**

Our delivery management system is an online platform that connects suppliers and organizations with couriers who work for them. Our system allows suppliers to purchase our service and connect to the website, where they can update orders as soon as they are received. The website then uses the system to send the orders to the efficient couriers, who will receive a notification and set up to carry out the order. This ensures that orders are delivered in a timely and efficient manner. Additionally, the system provides real-time tracking for the status of the orders, allowing suppliers and couriers to easily monitor the status of deliveries. The goal of the system is to improve the delivery process by automation and simplification the coordination and assignments of orders. The system is designed to be user-friendly and easy to navigate, making it accessible to businesses of all sizes.

## **1.Introduction**

Delivery management can be a complex and time-consuming task, especially for small businesses that don't have dedicated logistics departments. Many suppliers struggle to find reliable couriers, and couriers may waste time and resources traveling to pick up packages that are far from their current location. This problem is compounded by the fact that traditional delivery management solutions are often expensive and difficult to use.

By offering a user-friendly platform that connects suppliers and couriers in real-time and assists them in selecting the best delivery alternatives, our project seeks to address these issues. Our technology helps make sure that shipments are delivered as quickly and inexpensively as feasible by considering variables like courier availability, distance, and cost.

Suppliers, couriers, and the public who depend on prompt and dependable deliveries are among the project's stakeholders. Our objective is to provide a system that supports all of these groups and enhances the delivery procedure for all parties concerned.

We're eager to share our platform with you since we think it is the final piece in the delivery management puzzle. Our technology makes it simple for suppliers to locate and employ dependable couriers, and it gives couriers a selection of delivery tasks that fit their schedules and preferences. We want to make life easier for everyone involved by streamlining the delivery process and removing pointless fees and delays.

## 2. Background and Related Work

Delivery Operation has been a crucial concern for businesses for numerous times, with colorful approaches and technologies developed to streamline the process. former work in this area includes the use of GPS shadowing to cover deliveries, as well as the development of software platforms that connect suppliers with couriers. still, numerous of these results are limited in their compass or bear homemade trouble to find the efficient courier for delivery.

There are presently a range of delivery operation results available, ranging from simple apps to more advanced platforms that offer a lot of features and options. still, the utmost of these results is concentrated on a specific aspect of delivery operation, similar as tracking or payment.

We will show many algorithms that have been developed by others that is relevant to find the most suitable courier to have the order.

### *2.1 Nearest Neighbor Algorithm [1]*

In the environment of delivery operation, the Nearest Neighbor Algorithm could be used to find the nearest courier to a supplier's position, or the delivery route that's closest to the supplier's position. This could be fulfilled by storing a database of courier locales and delivery routes and comparing the new data (the supplier's position) to the stored data to find the nearest match. Our design aims to bring together all the necessary features in one place, making it easy for suppliers to find the stylish courier for their delivery and track their orders from volley to delivery.

### *2.2 Genetic Algorithm [2]*

This algorithm is a type of optimization technique that uses principles of natural evolution to find the best solution to a problem. It's inspired by the process of natural selection, in which the fittest individualities are more likely to survive and reproduce. In the environment of delivery operation, an inheritable Algorithm could be used to optimize delivery routes for a set of orders, considering factors similar as distance, time, and cost. The algorithm works by creating a population of implicit results (in this case, delivery routes) and iteratively perfecting them through a process of selection, crossover, and mutation.

The selection process involves opting the fittest individualities (delivery routes) from the population to move on to the coming generation. The crossover process involves combining the attributes of two individualities (delivery routes) to produce a new seed. And the mutation process involves making small arbitrary changes to an individual to introduce new inheritable material.

### *2.3 Machine Learning Algorithms [3]*

In the context of delivery management, machine learning algorithms could be used to predict demand for deliveries in each area, or to optimize the allocation of deliveries to couriers based on their availability and location.

For example, a supervised learning algorithm could be trained on data about past delivery requests and the corresponding allocation of couriers. The algorithm could learn to map the input data (characteristics of the delivery request such as location, size of the package, and time of day) to the output data (the best courier for the delivery). The trained algorithm could then be used to predict the best courier for a new delivery request based on the input data.

### *2.4 Ant Colony Optimization [4]*

The algorithm works by simulating the behavior of ants as they search for food. Each ant is responsible for finding a path from the starting location (the supplier's location) to the destination (the customer's location). As the ants explore the environment, they lay down pheromone trails that help to guide the other ants towards the best path.

The ACO algorithm uses this process to find the shortest or most efficient path between the starting and destination points. It does this by adjusting the pheromone trails based on the success or failure of the ants in finding food (or in this case, completing deliveries). Over time, the pheromone trails become more concentrated on the best paths, and the ants are more likely to follow these paths in the future.

In our problem, an ACO algorithm could be used to optimize the delivery routes for a set of orders, considering factors such as distance, time, and cost. The algorithm could be used to find the most efficient routes for individual deliveries or for a fleet of couriers. It could also be used to adapt to changing conditions in the environment, such as traffic delays or road closures.

In addition, we will show some active platforms [5] that have the same issue as our project,

### *2.5 Delivery management apps*

There are many delivery management apps that allow businesses to plan, optimize, and track routes for their deliveries. It is essentially targeted at small and medium businesses, such as courier services, and food delivery businesses.

Some features of this apps include:

- Allows users to enter multiple addresses for their deliveries and then optimizes the route for the most efficient order of stops.
- Allows users to track their vehicles and drivers in real-time, including their location, speed, and the estimated time of arrival.
- Users can send notifications to their customers about the status of their deliveries, including when the delivery has been dispatched, is the end route, or has been completed.
- Includes features for managing and scheduling vehicles, including the ability to set maintenance reminders and track fuel usage.

In other words, the platforms that are available today help a lot of companies, but still don't have the perfect solution to interactive between the supplier and the courier, and this what we expect to achieve.

### **3. Expected Achievements**

#### *3.1 Outcomes*

In this project, we aim to develop a comprehensive delivery management platform that connects suppliers with a network of couriers and helps them choose the best option based on location, availability, and price. Our platform will offer a range of features to make delivery management easy and efficient, including the ability for suppliers to track their orders from pickup to delivery and pick the courier from a table (K couriers) and for couriers to choose from a variety of delivery jobs and get paid for their services.

#### *3.2 Unique Features*

##### 3.2.1 Finding the nearest courier

Route optimization refers to the process of finding the most efficient or cost-effective routes for a set of deliveries. In our work, involve finding the shortest distance between the starting location (the supplier's location) and the destination (the customer's location) by using [1] with the KNN version (K neighbors).

##### 3.2.2 GPS Tracking

The GPS (Global Positioning System) tracking will be used to monitor the location, the longitude and the latitude, providing information about the status and progress of each delivery. To use GPS tracking in our project, we will need to equip each courier devices with GPS, such as smartphones. These devices will need to be able to transmit their location data to our platform, which can then be accessed in real time.

##### 3.2.3 Alerts and Notifications

This feature is a way to keep the supplier and couriers informed about the status and progress of each delivery. That could be used to communicate a variety of information.

###### 3.2.3.1 Confirmation of delivery requests

When a new order request and the delivery of it received, a notification will send to the supplier and the courier to confirm the status of the order.

#### 3.2.3.2 Updates on the status of deliveries

Notifications will send to the supplier and courier to provide updates on the status of the delivery, such as the order confirmed by the courier, order is pending and order has been delivered.

#### 3.2.3.3 Changes to delivery schedules

If there are any changes to the delivery schedule such as a delay or a change in the delivery location, notifications will be sent to the supplier and courier to inform them of the changes and provide the suitable updates.

#### 3.2.3.4 Alerts for missed or late deliveries

If a delivery is missed or running behind schedule, notifications (and an alerts) will be sent to the users to be informed about the situation and they must provide updates.

#### 3.2.3.5 Alerts for emergencies or unexpected events

In the event of an emergency or unexpected event (such as an accident or the courier cannot take the order) alerts will be sent to the supplier to inform him of the situation.

### *3.3 Criteria for Success*

The success criteria for our project will include the number of active users, the satisfaction of both suppliers and couriers with the app, and the efficiency of the delivery process. We will also measure the time and cost savings for both suppliers and couriers compared to traditional delivery management methods. In addition, we will also need to address non-trivial requirements such as ensuring the security and privacy of each user's data and providing a user-friendly interface for both suppliers and couriers.



## 4. The Process

### *4.1 Research – Delivery Management*

Regarding broadening our understanding of Delivery Management, we focused on answering the following questions:

- How should we pick the nearest courier to the supplier to take the order?
- What are the most important factors that influence a supplier's decision to use our system?
- What are the most important factors that influence a courier's decision to work with our system?
- How can our system be designed to ensure the safety and security of packages and couriers during the delivery process?

We consulted a range of materials, from academic papers and articles to videos, to address these issues and increase our expertise. After looking over the resources, we got together to talk about our results and identify the key aspects we should pay attention to when developing the application.

Some of the conclusions we reached while reading about Delivery Management with regards to developing software, there exists a need to add a questionnaire after making an evaluation in the simulation that can be:

- Overall, how satisfied are you with the delivery management app?
- How easy or difficult was it to use the app?
- What features or functionality did you find most useful in the app?
- What features or functionality would you like to see added to the app in the future?
- Was the app reliable and stable during use?
- Were you able to easily track your deliveries using the app?
- How would you rate the speed and efficiency of the app?

If we find correlation between the questionnaire and our final analysis, this will validate our delivery management application.

#### 4.1.1 Constraints and Challenges – Delivery Management

Picking a suitable courier for an order can be a challenge in our system because there may be multiple factors to consider when determining which courier is the most suitable. Some of these factors may include:

##### *4.1.1.1 Distance*

The distance between the supplier and the courier can impact the efficiency of the delivery and the time required to complete it.

##### *4.1.1.2 Availability*

The courier's availability (such as whether they are currently on delivery or have capacity for additional orders) can impact their suitability for an order.

To address this challenge, our system considers all these factors when determining the most suitable courier for an order. This obligates us to have access to detailed information about the couriers (such as their location, equipment, and availability) and the ability to weigh these factors against each other to decide.

## *4.2 Methodology and Development Process*

In order to develop a system that meets our desired goals, we have created a structured work plan that outlines all the necessary steps, starting with the learning process and proceeding through the development process. Through our research, we determined that the Agile approach would be the most effective method for our project. This iterative approach emphasizes the importance of working in stages with ongoing planning and continuous learning.

To begin, it was necessary for us to thoroughly research existing solutions and technologies in the market in order to gain a comprehensive understanding of their strengths and weaknesses. Our goal was to identify ways in which we could leverage the advantages of these solutions while also seeking opportunities to improve upon their shortcomings to create a truly unique system. This required a thorough analysis of the current market landscape to identify areas where we could make a meaningful contribution.

Our work process is divided into three main parts:

### **4.2.1 Planning and preparation**

This included meetings in which we created a detailed requirements document for the system, developed diagrams to visualize the desired outcome, and identified the tools and technologies we would need to use. Specifically, we decided to use React, JS, HTML and CSS as a front-end development, NodeJS for the back-end development, DB Microsoft SQL server for managing the data of our interactive website.

### **4.2.2 Implementation**

This includes using the Twilio service to connect to an SMS API and send confirmation messages to the suppliers, as well as defining and completing the initial tasks for the project such as creating the main screens for the users and building the basic infrastructure.

### **4.2.3 Iterative development**

As we learn and make progress, we will continue to develop and build upon the system in stages until we reach the final product. This will involve ongoing planning and adjustments based on lessons learned.

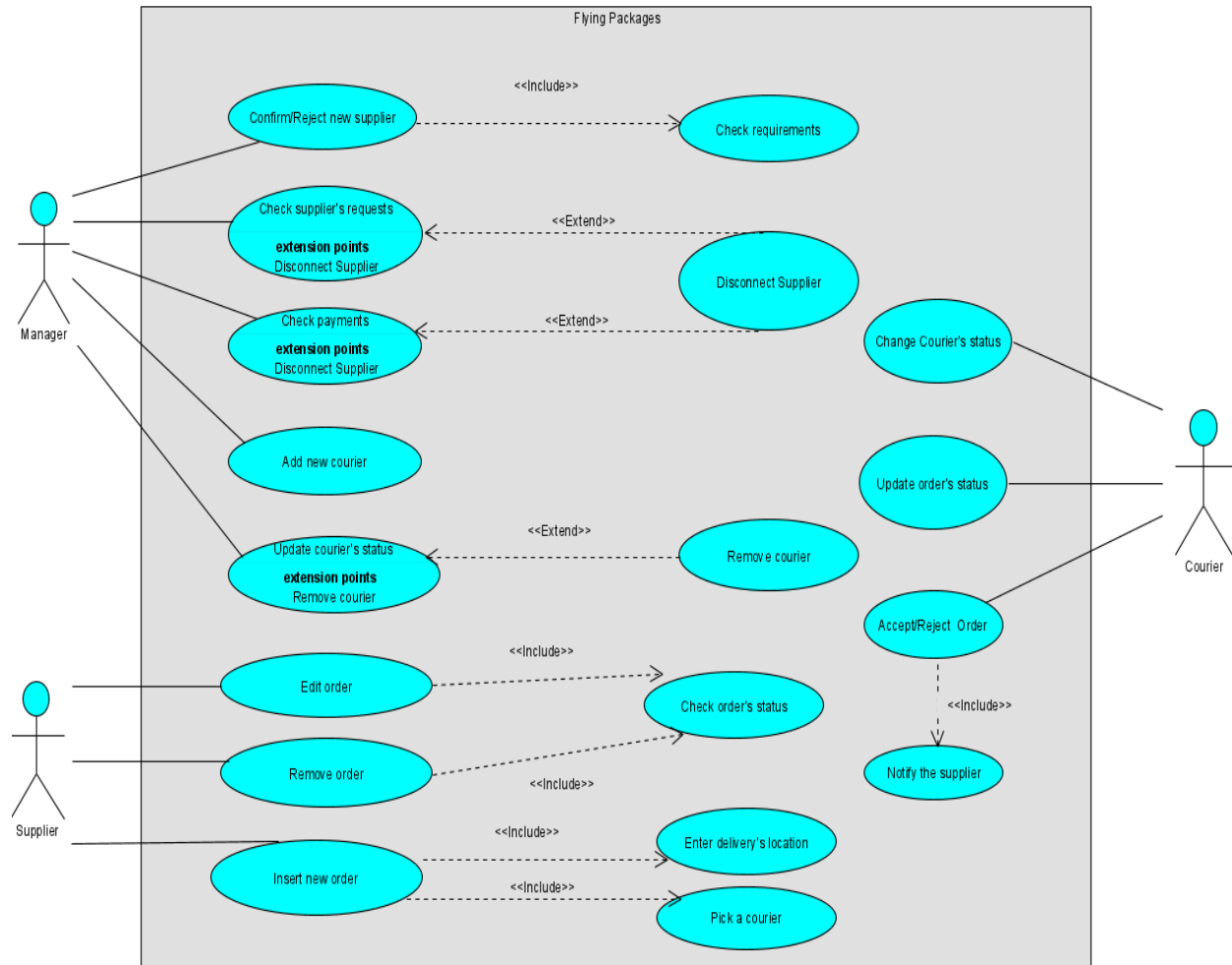
## 5. Product

### 5.1 Requirements

Req. number	Requirements description	Req.type (FR or NFR)
1	The system allows to identify users by username and password	FR
2	The username is unique for each user	NFR
3	The system allows identifying the user type immediately after logging in	FR
4	The user types are Courier, Supplier, and Manager	NFR
5	The system allows each supplier to add a new order	FR
6	Each order contains the required date and location	NFR
7	The system allows each supplier to choose the courier of his order	NFR
8	The system allows each supplier to adjust his orders	FR
9	The allowed changes to make are the date and time	NFR
10	The order can be edited/adjusted only if the order's status is "pending confirmation"	NFR
11	The system allows each supplier to remove orders	FR
12	The supplier can remove his order only if the order's status is "pending confirmation"	NFR
13	The system allows each supplier to pay	FR
14	The payments are monthly requested	NFR
15	The system allows the manager to add new couriers	FR
16	Each courier that will be added has to contain the personal information	NFR
17	The system allows the manager to change the courier's status	FR
18	The courier status can be: {Okay, warning, fired}	NFR
19	The system allows the manager to remove couriers	FR
20	The system allows the manager to follow the supplier's payments	FR
21	The system allows the manager to change the status of the supplier's payment	FR
22	The status of each payment can be: {Pending, Done}	NFR
23	The system allows couriers to accept/reject orders	FR
24	The system allows couriers to update the order's status to "completed"	FR

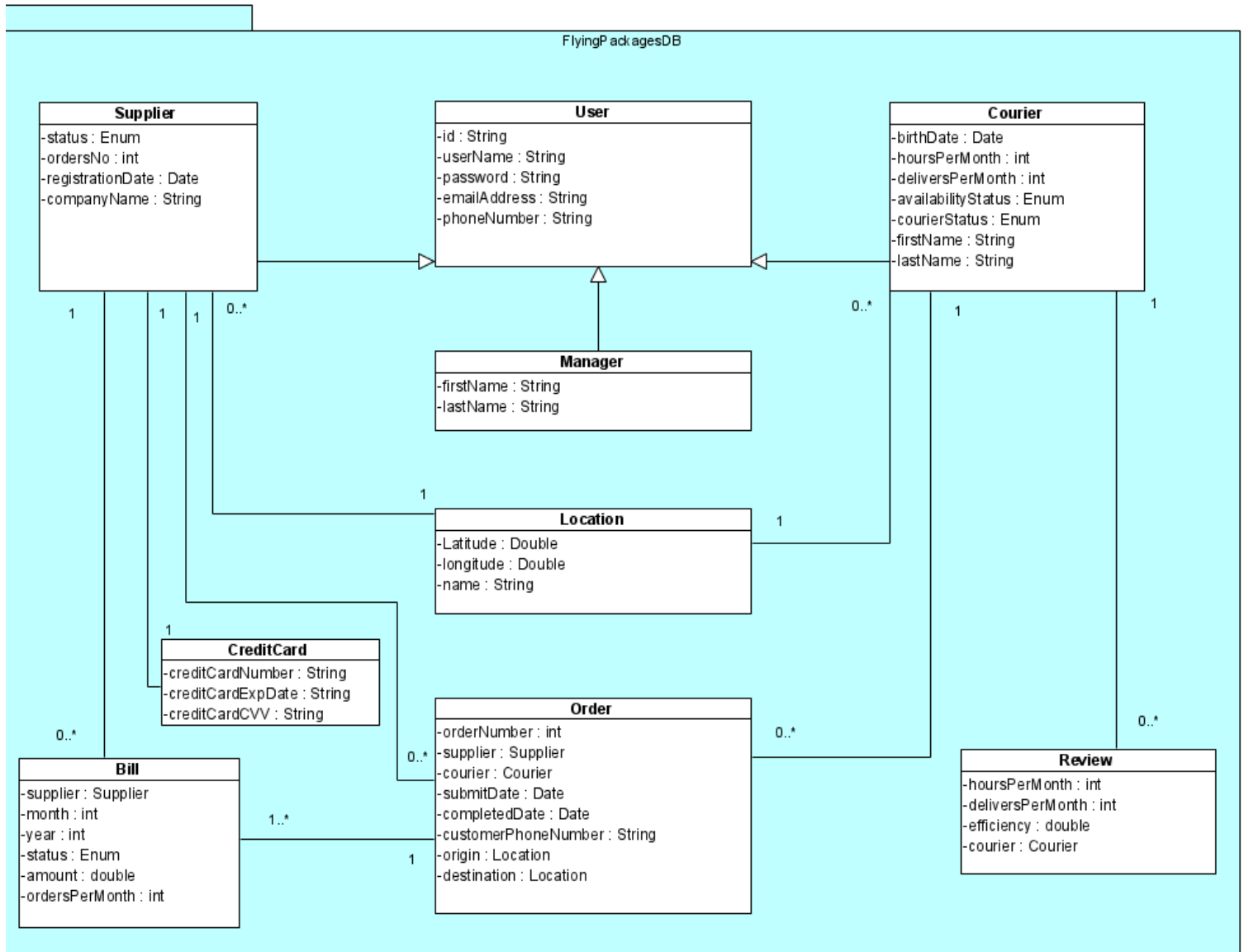
## 5.2 UML Diagrams

### 5.2.1 Use Case Diagram



**Fig1:** Use Case Diagram

## 5.2.2 Class Diagram



**Fig2:** Class Diagram - Entities

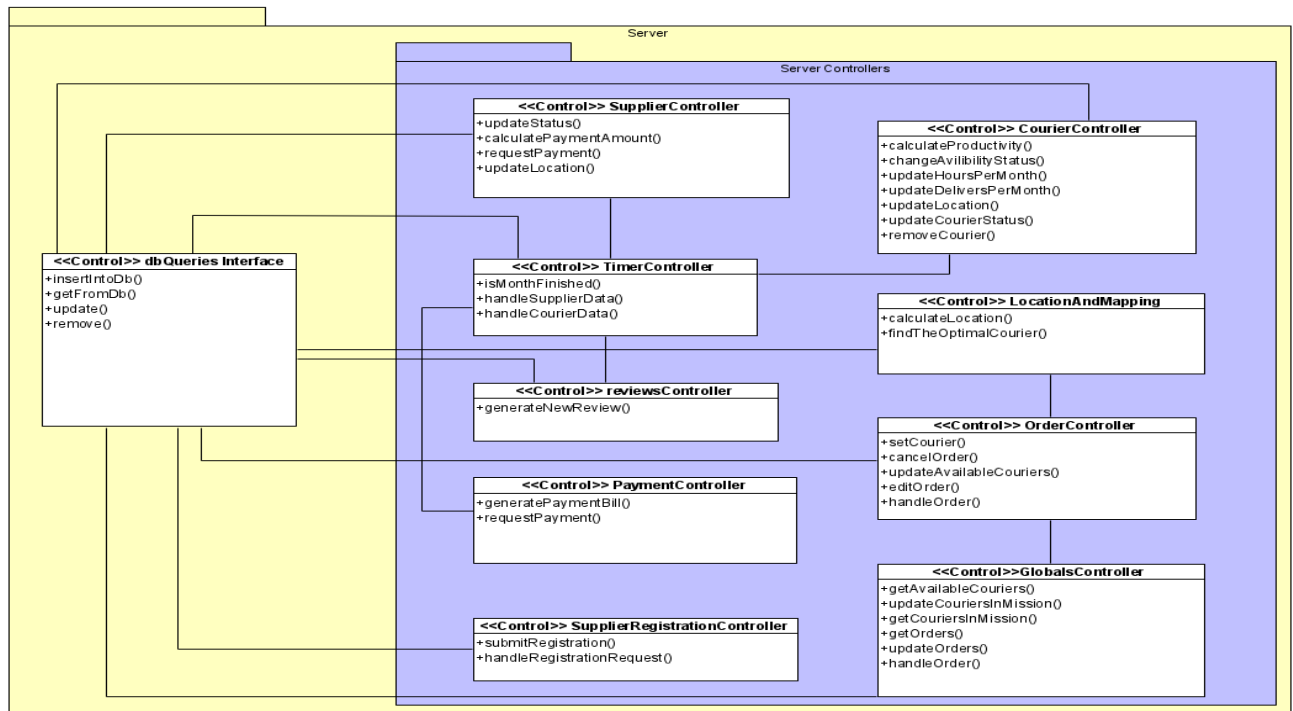


Fig3: Class Diagram - Server

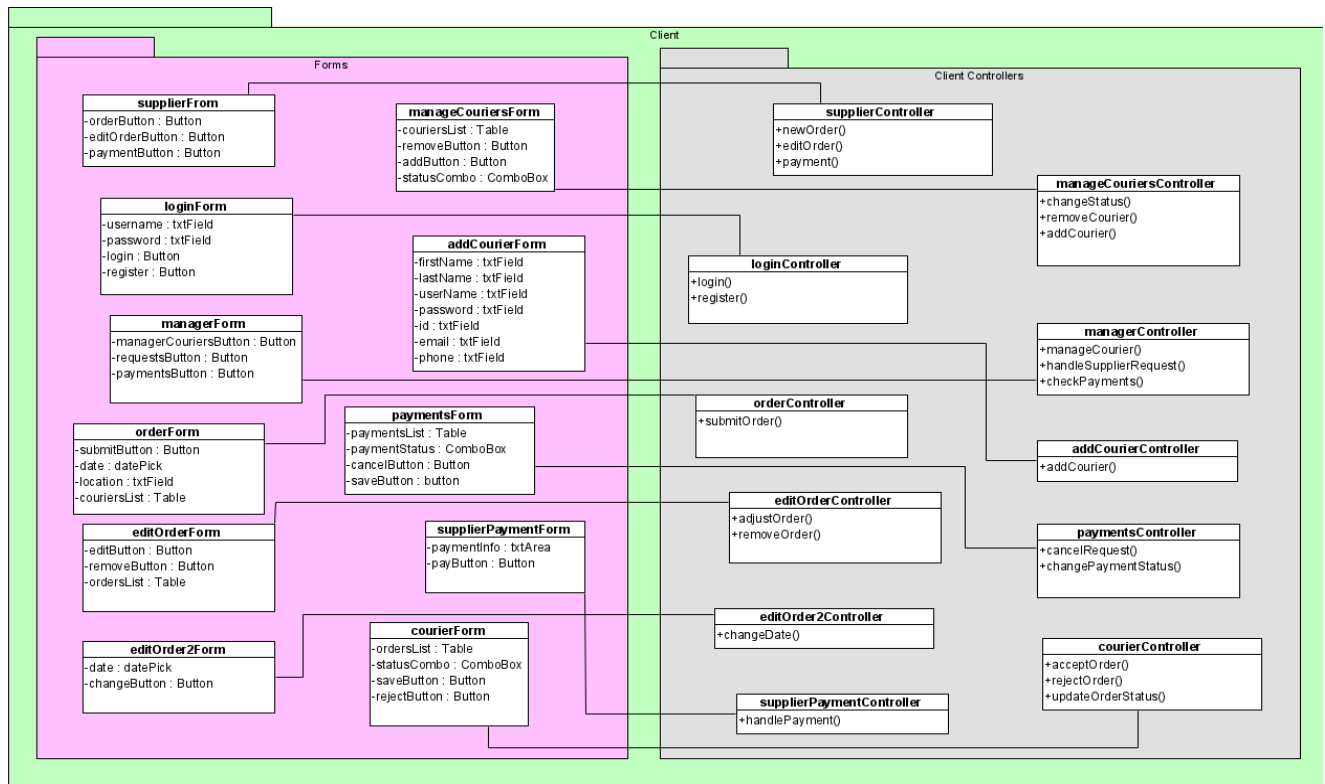
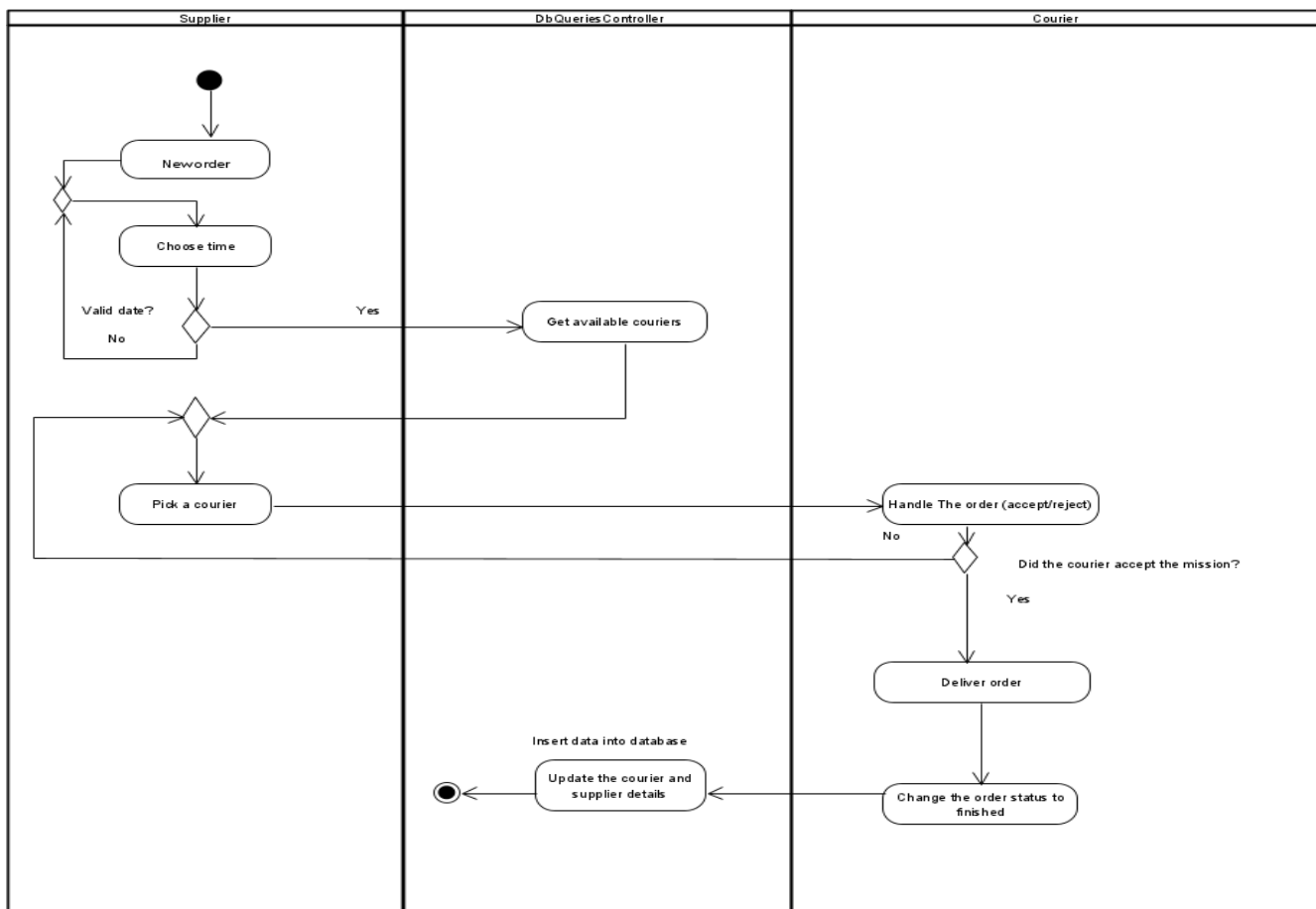


Fig4: Class Diagram - Client

### 5.2.3 Activity Diagram

#### 5.2.3.1 Order

- 1- The supplier chooses the new order option.
- 2- The supplier picks a date and time for his order.
- 3- The System checks if the date is valid.
- 3.1- if the time is not valid, we return to step 2.
- 3.2- if the time is valid, the system gets a list of available couriers.
- 4- The supplier picks the courier that he wants to carry his delivery.
- 5- The courier gets a notification.
- 6- The courier will choose to accept and carry the delivery or not.
- 7- if the courier chooses to not do it, we return to step 4.
- 8- if the courier accepts it, he delivers the order.
- 9- The courier changes the status of the order after he finishes.
- 10- The system inserts the order's details into the Database.

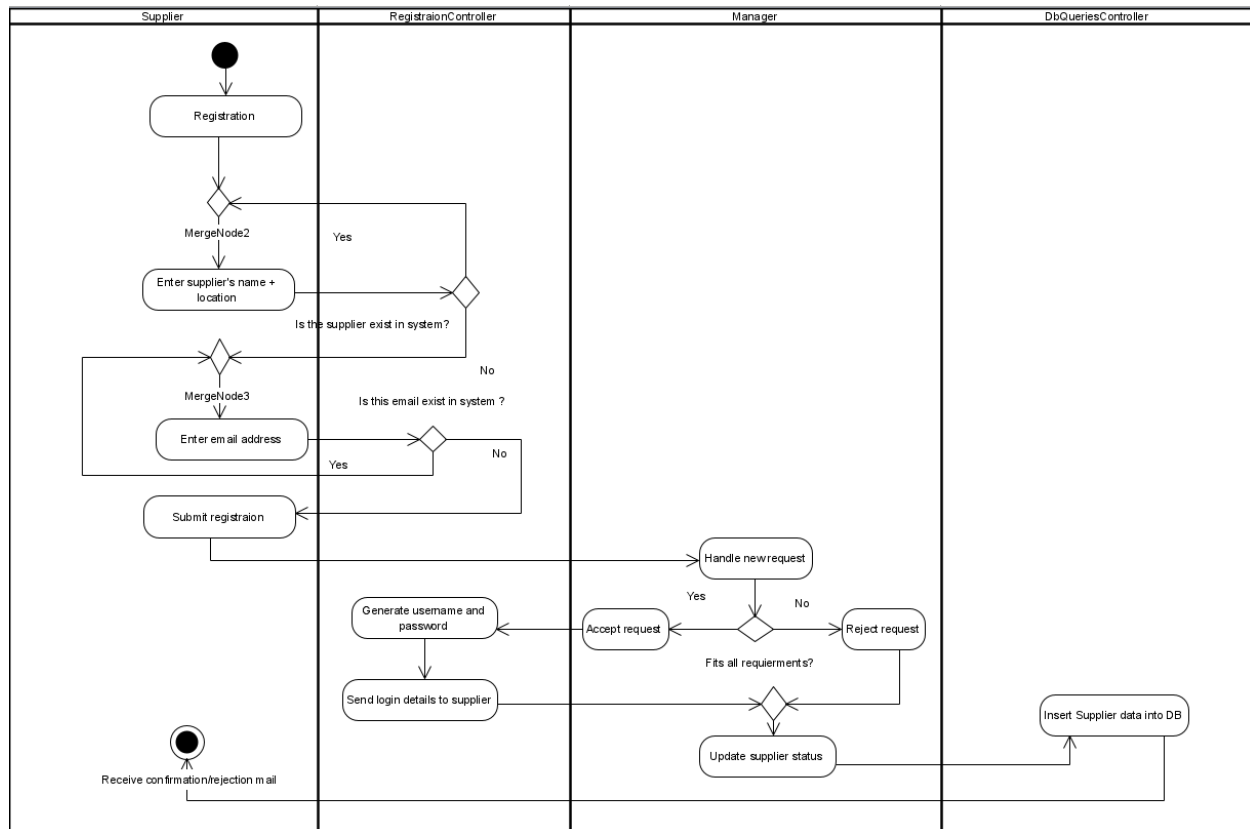


**Fig5:** Activity Diagram – Order



### 5.2.3.2 Register new supplier

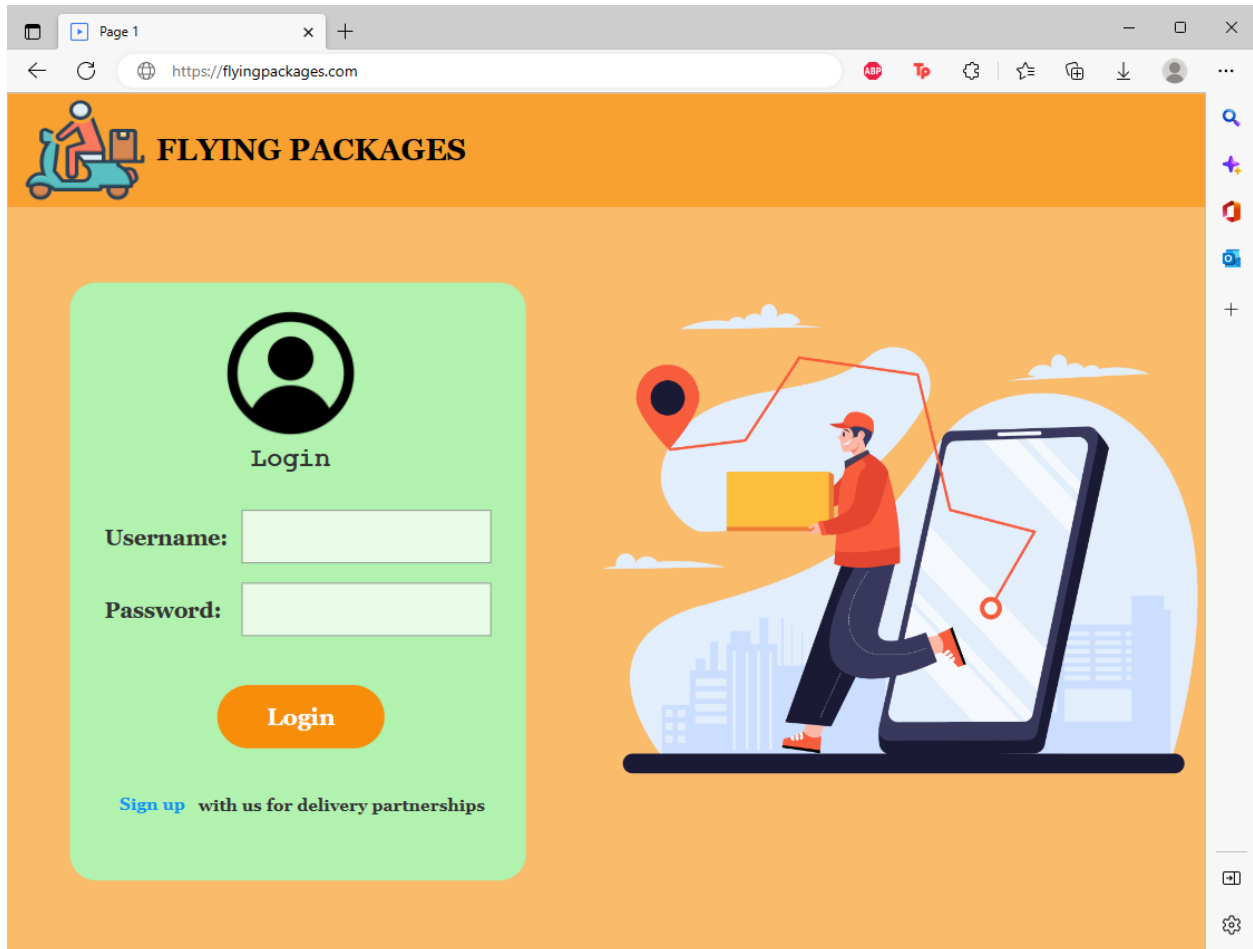
- 1- The supplier chooses to register for our system.
- 2- The supplier enters the name of the company and the location.
- 3- The system checks if the name and location exist.
- 3.1- If yes, we return to step 2.
- 3.2- If no, the supplier enters the email address.
- 4- The system checks if the email is already taken.
- 4.1- If yes, we return to step 3.2.
- 4.2- If no, the supplier submits his registration.
- 5- The manager decides whether to accept the request or not.
- 6- If he accepts, the system will generate a username and password.
- 6.1- The system sends the login details to the supplier.
- 7- The manager updates the supplier's status.
- 8- The system inserts the supplier's details into the DB.
- 9- The supplier receives a mail of confirmation/rejection.



**Fig6:** Activity Diagram – Register new supplier

### 5.3 User Interface

#### Login:



**Fig7:** Login page

This is the first page that every user sees when he enters our website, each user can join with his private username and password, and the system will recognize the type of the user (courier/manager/supplier) and open the relevant page. For the suppliers who want to join our system, they can sign up by the linked text below the Login button.

## Register:

**FLYING PACKAGES**

**Registration**

**Company's name:**

**Location**

**City:**

**Address:**

**Postal code:**

**Next**

**Fig8:** Register page

We will reach this page after clicking on signup from the previous page, The supplier will enter his company's name and fill in the location's info.

**FLYING PACKAGES**

**Complete the registration:**

**ID:**

**Email:**

**Phone number:**

**Credit card number:**

**Expiration date:**

**CVV:**

**Submit**

**Fig9:** Complete register page

This is the following page of the registration one, after clicking next this page will appear in case the name and location are not taken. The supplier fills in the details and submits his registration.

### Supplier's home page:

**FLYING PACKAGES**

Welcome, Supplier GoTo!

**Last orders**

Order num	Courier	Destination	Status
#65	846268215	Haifa, Horev 52	IN PROGRESS
#43	745692157	Haifa, Hanase 3	COMPLETED
#22	792058137	Haifa, Kerel 61	COMPLETED

[View more](#)

**New order**

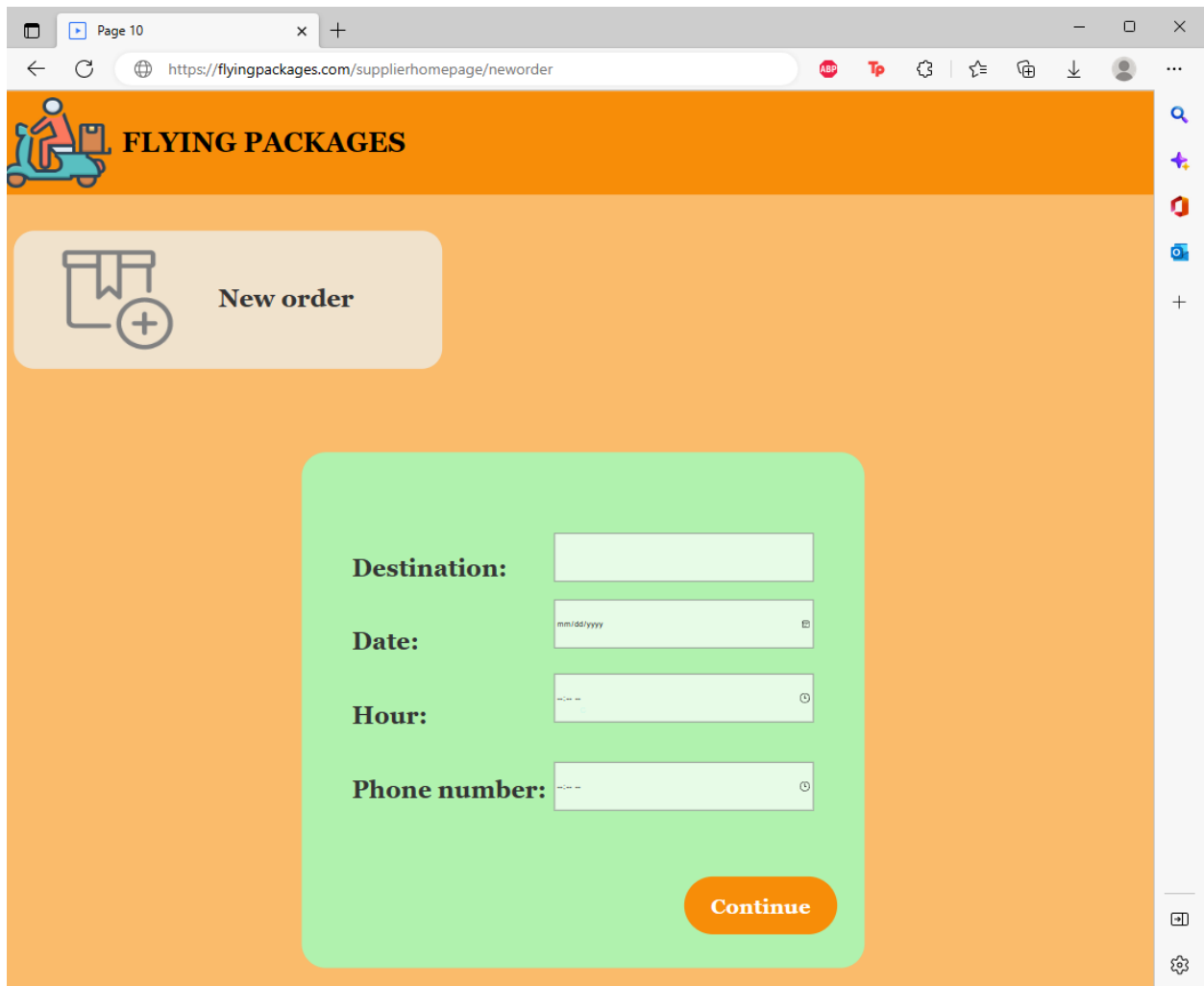
**Edit orders**

**Monthly bill payment**

**Fig10:** Supplier's homepage

This is the homepage for each supplier, after clicking login from [Figure 7](#) with the supplier's username and password. On this page the supplier can see his last orders from the table above, he can choose to insert new orders, he can choose to edit his orders (change the time if it is possible) and can see his monthly bills. The supplier can choose to click on the View more linked text and watch more orders than those who appear in the table.

## New order:



The screenshot shows a web browser window with the URL <https://flyingpackages.com/supplierhomepage/neworder>. The page features an orange header with the 'FLYING PACKAGES' logo and a 'New order' button. Below the header is a green form with the following fields:

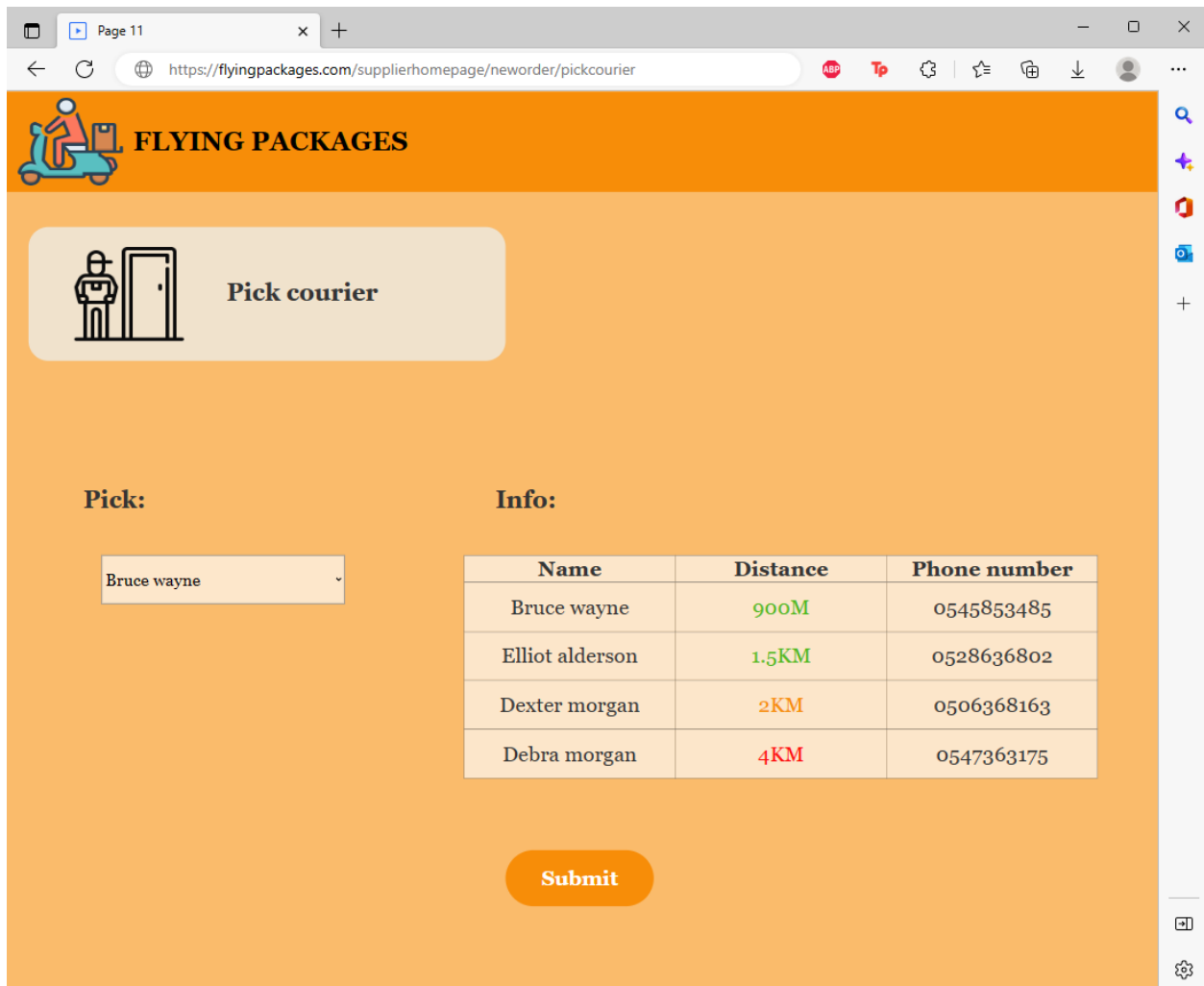
- Destination:** A text input field.
- Date:** A date input field with a placeholder 'mm/dd/yyyy' and a calendar icon.
- Hour:** A time input field with a placeholder 'hh:mm' and a clock icon.
- Phone number:** A text input field with a placeholder '000-000-0000' and a phone icon.

A green 'Continue' button is located at the bottom right of the form.

**Fig11:** New order page

We will reach this page after clicking on the new order button from the previous one, the supplier will fill in the details including the destination of the delivery and the customer's phone number, and then he will click on continue to complete the process. The system will check if the time is valid and then let the supplier move to the next page. If it's not valid, a warning message will appear on the screen.

### Pick a courier:



**FLYING PACKAGES**

**Pick courier**

**Pick:**

Bruce wayne

**Info:**

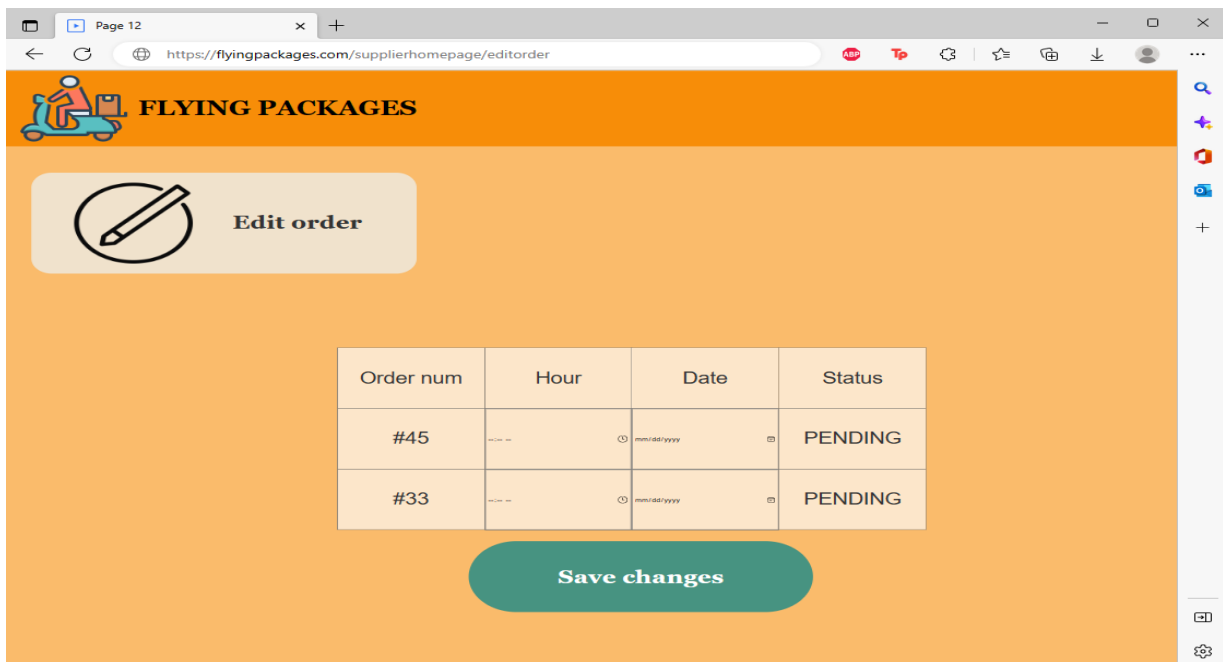
Name	Distance	Phone number
Bruce wayne	900M	0545853485
Elliot alderson	1.5KM	0528636802
Dexter morgan	2KM	0506368163
Debra morgan	4KM	0547363175

**Submit**

**Fig12:** Pick a courier page

After clicking on Continue from the previous page, this page will appear. The supplier can see the available couriers that can carry his delivery on a table, with the distance according to the courier's location, the distance is marked by colors: green, orange, and red. We don't want to choose a courier for the supplier, so we give him advice on which courier he should choose by marking the optimal couriers according to the distance on green. The supplier will see the information on the table and will pick the courier he wants by the drop list on the left side, then press on submit.

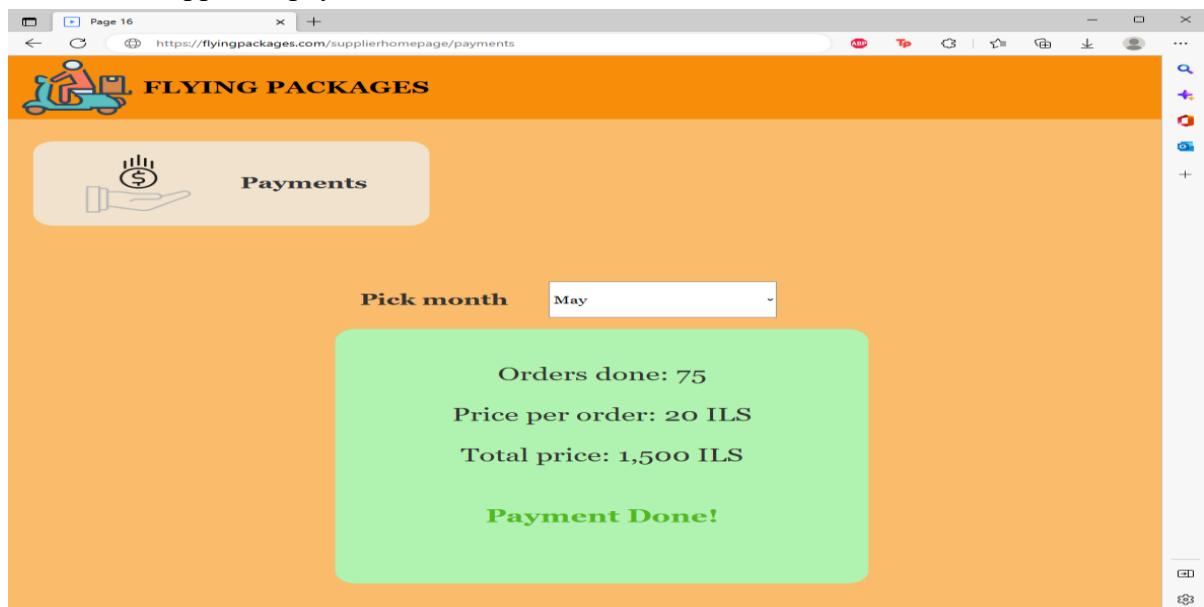
### Edit order:



**Fig13:** Edit order page

This page will appear after clicking on the edit order button from [Figure 10](#). The supplier will see his orders that are still in Pending status. He can change the time from the table.

### Supplier's payments:



**Fig14:** supplier's payments page

we will reach this page after clicking on the Monthly bill payment from [Figure 10](#). The supplier will choose the month that he wants to see its bill, and the bill appears including the details from the specific month and the payment status (if it's done or pending).

### Manager homepage:

**FLYING PACKAGES**

Welcome, manger Sam!

#### Last orders

Order num	Supplier	Courier	Status
#100	Macdonald's	846268215	IN PROGRESS
#99	Super-Pharm	745692157	IN PROGRESS
#98	Good baguette	792058137	COMPLETED

[View more](#)

**Manage couriers**

**Registration requests**

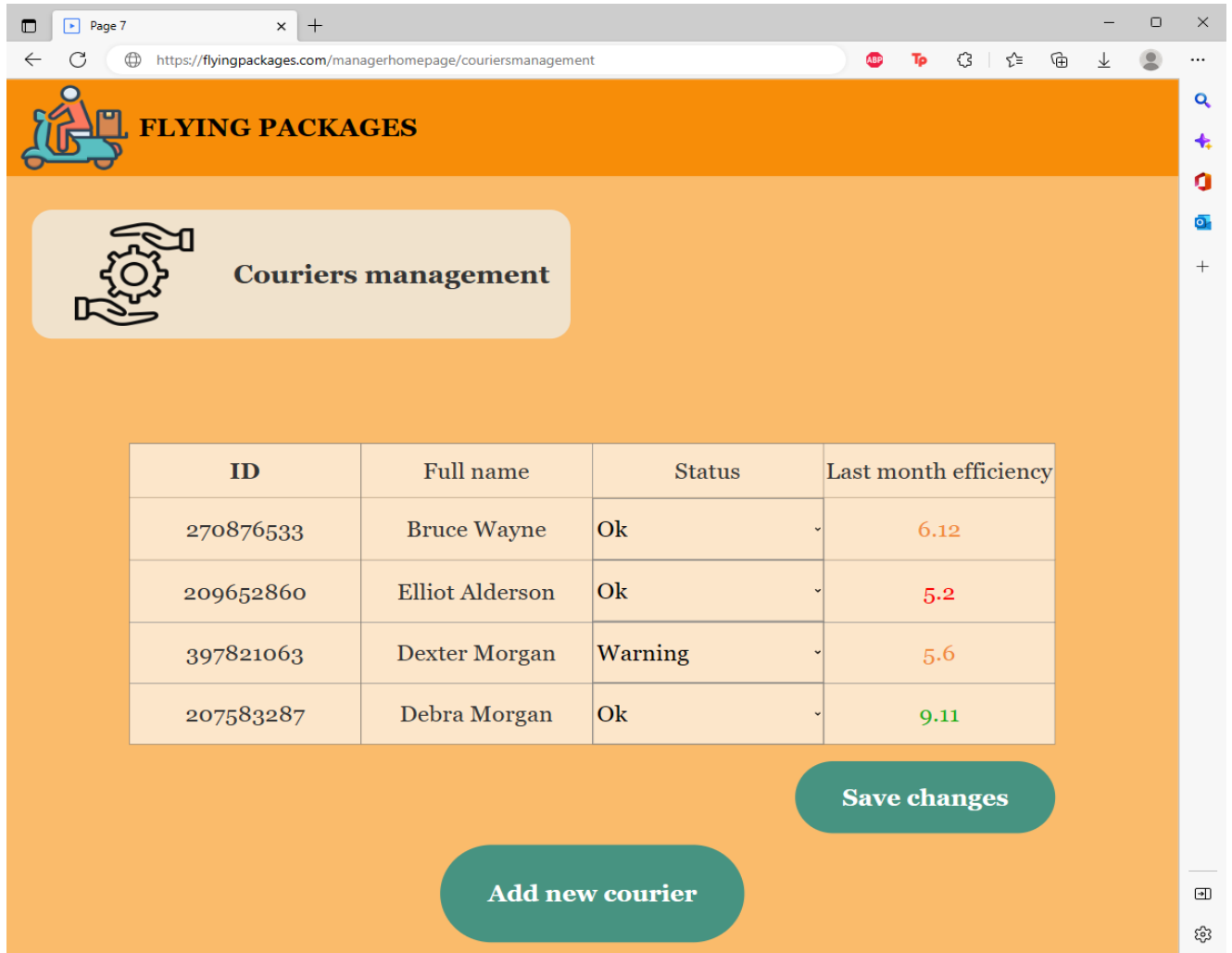
**Payments reports**

**Fig15:** manager homepage

This is the homepage for each manager, after clicking login from [Figure 7](#) with the manager's username and password. On this page, the manager can see the last orders from the table above, and he can click on view more to see more orders. The manager can click on each button from the right side: manager couriers/registration requests and payments reports.



### Manage couriers:



**FLYING PACKAGES**

**Couriers management**

ID	Full name	Status	Last month efficiency
270876533	Bruce Wayne	Ok	6.12
209652860	Elliot Alderson	Ok	5.2
397821063	Dexter Morgan	Warning	5.6
207583287	Debra Morgan	Ok	9.11

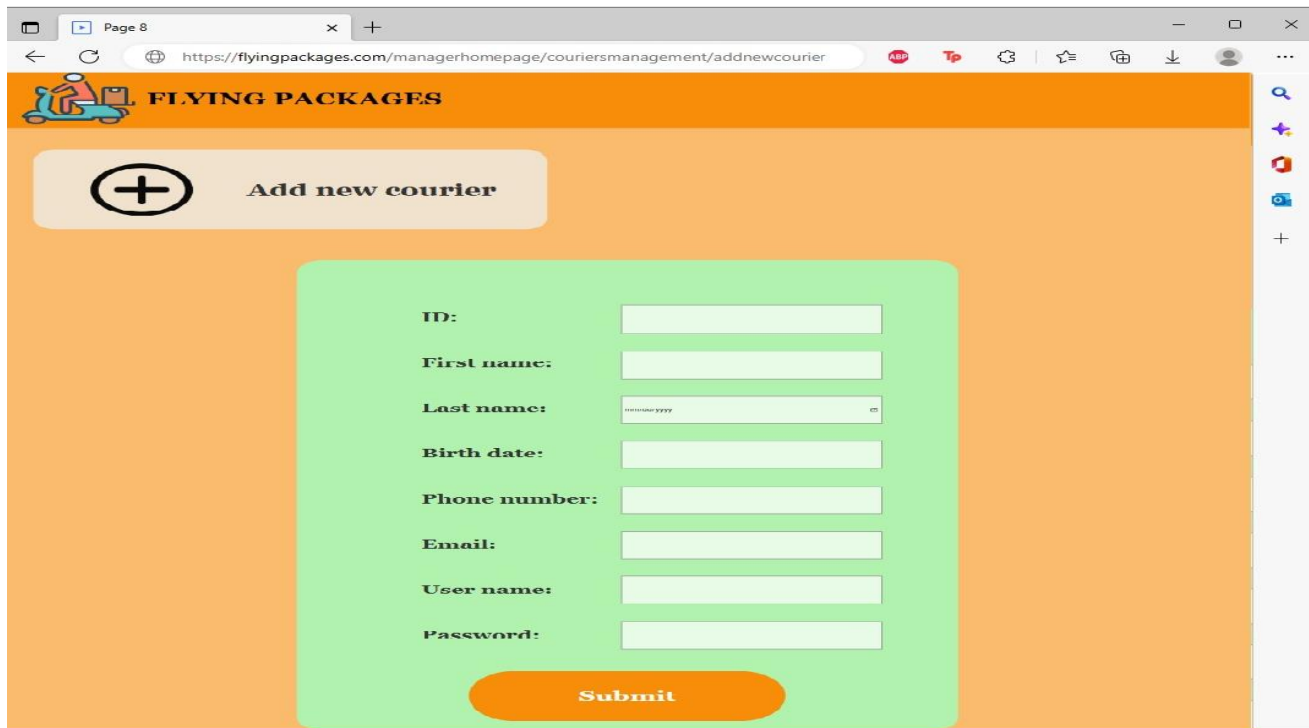
**Save changes**

**Add new courier**

**Fig16:** manage courier's page

We will reach this page after clicking on the manage couriers' button from the previous one. The manager will see the courier's details in the table, and he can change the courier's status from the status column and then click on save changes, he can see the efficiency from the last month and decide to change the status accordingly. If the manager changes a courier status to fired it means that the system will not consider the courier for the upcoming orders. The manager can add a new courier by clicking on the button below.

### Add new courier:



**FLYING PACKAGES**

**Add new courier**

ID:

First name:

Last name:

Birth date:

Phone number:

Email:

User name:

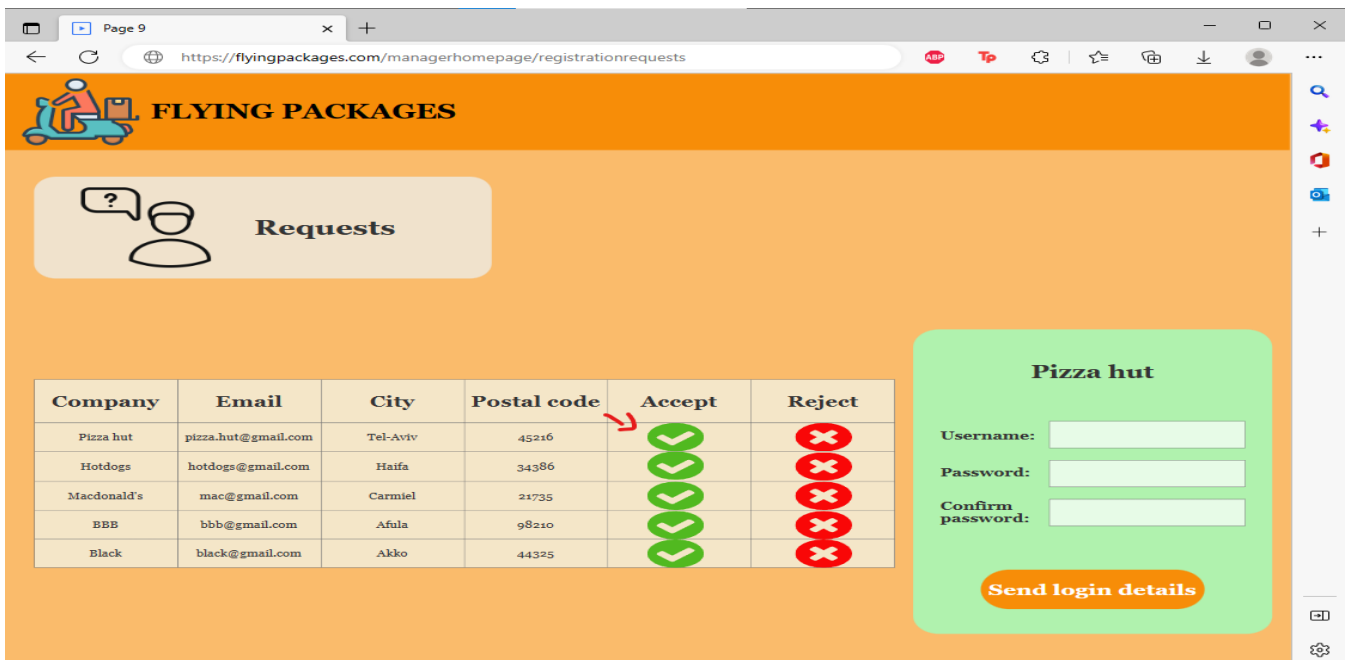
Password:

**Submit**

**Fig17:** add new courier page

This page is reachable by clicking on add new courier button from the previous page, the manager fills in the details and clicks on submit to add the courier to the system.

### Registration requests:



**FLYING PACKAGES**

**Requests**

Company	Email	City	Postal code	Accept	Reject
Pizza hut	pizza.hut@gmail.com	Tel-Aviv	45216	✓	✗
Hotdogs	hotdogs@gmail.com	Haifa	34386	✓	✗
Macdonald's	mac@gmail.com	Carmiel	21735	✓	✗
BBB	bbb@gmail.com	Afula	98210	✓	✗
Black	black@gmail.com	Akko	44325	✓	✗

**Pizza hut**

Username:

Password:

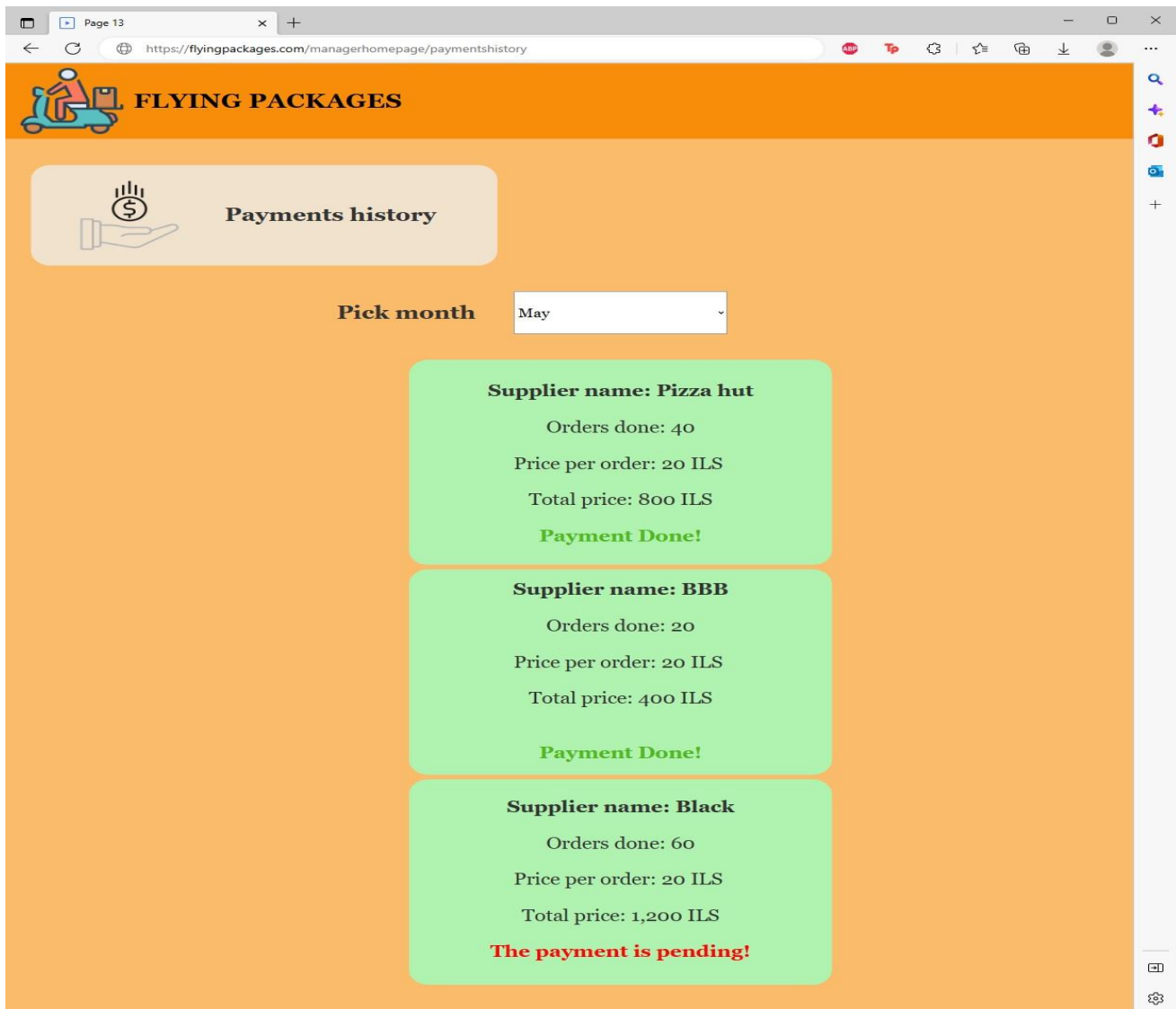
Confirm password:

**Send login details**

**Fig18:** registration requests page

The manager will reach this page after clicking on registration requests from [Figure 15](#). The manager will see the table of the suppliers who requested to join our system. He can click on the accept button to accept each one and he can reject by clicking on the red cross button. After clicking on the accept button for any supplier's request the window on the right side will appear and he will have to enter a username and password for the supplier he wants to accept (on our figure, Pizza Hut). After he fills in the details and clicks on send login details, the supplier will receive the confirmation to the e-mail he registers with, and the mail will include the supplier's login details.

#### Payments reports:



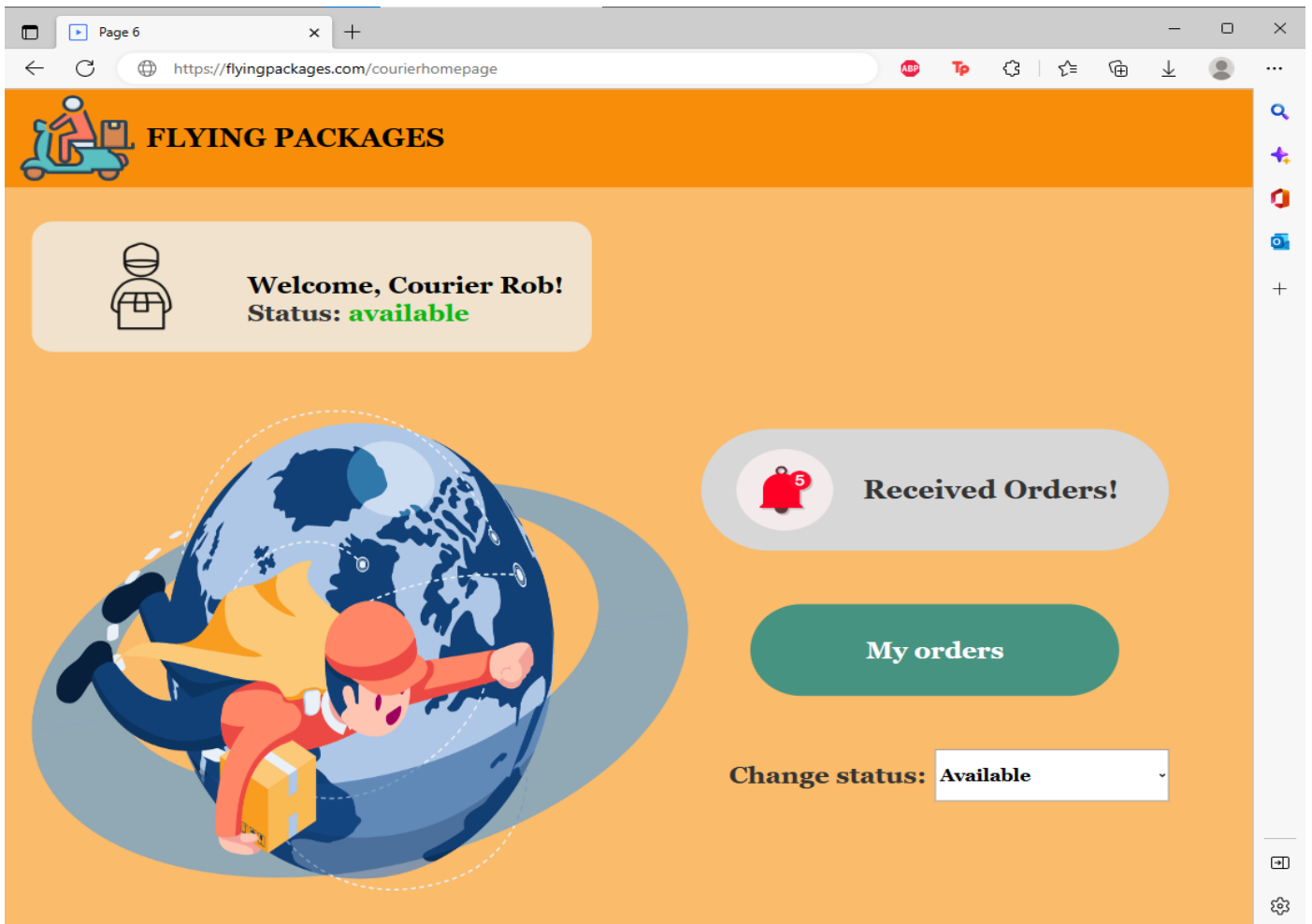
The screenshot shows a web browser window with the URL <https://flyingpackages.com/managerhomepage/paymentshistory>. The page has an orange header with the 'FLYING PACKAGES' logo. Below the header, there is a 'Payments history' section with a hand icon and a 'Pick month' dropdown menu set to 'May'. The main content area displays three supplier entries in green boxes:

Supplier name	Orders done	Price per order	Total price	Payment Status
Pizza hut	40	20 ILS	800 ILS	Payment Done!
BBB	20	20 ILS	400 ILS	Payment Done!
Black	60	20 ILS	1,200 ILS	The payment is pending!

**Fig18:** payments history

The manager will reach this page after clicking on the payments reports button from [Figure 15](#). On this page, he can pick a month and see the bills for all suppliers from the specific month.

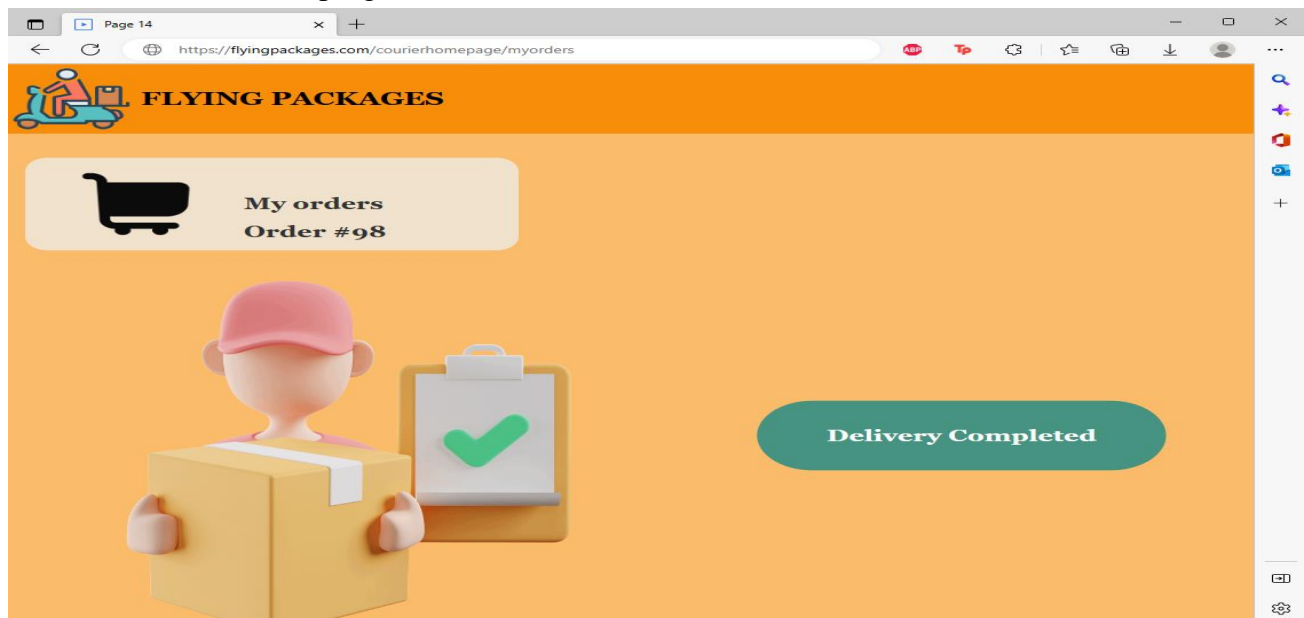
### Courier homepage:



**Fig19:** courier homepage

This is the homepage for each courier, after clicking login from [Figure 7](#) with the courier's username and password. The courier can see his status below the welcome message, and he can change it from the drop table (if he wants to take a rest for example). The courier can see a notification behind the received orders label. He will see those orders by clicking on the My orders button, if he is already carrying a delivery, he can't receive orders until he finishes. And he will click on my orders to change the status of his delivery and notify the system that he completed it.

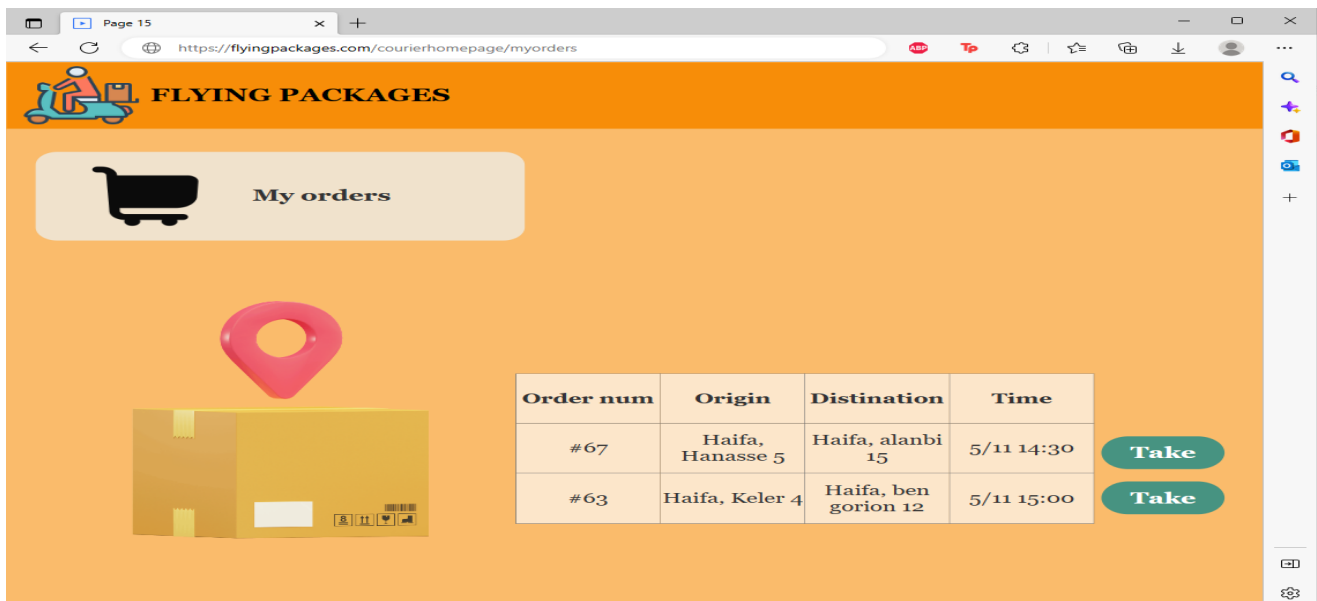
### Order in progress:



**Fig20:** my orders page

In this case, the courier clicked on my orders button from the previous page, while his last order's delivery has not been completed yet. He can see this page which includes the order number and a button that he can use to notify the system that the order is completed.

### Received orders:



**Fig21:** my orders page

Here is the other case, where the courier is not busy, and he can see the orders that he received, and he can accept only by clicking on the Take button behind the row of the order he wants to take.

## 6. Verification and Evaluation

### 6.1 Unit Tests

No#	Test Subject	Expected result
<b>Login form</b>		
1	Enter an empty username or password	Error displays a message: "Please fill in the fields".
2	Enter the wrong username or password	Error displays a message: "Invalid details, try again" We don't want to give information about which field is invalid.
3	Log in with a 'frozen' status username and password	Display message: "User is Frozen".
4	Enter a valid username and password	Display message: "Login succeed".
<b>Supplier's registration form</b>		
5	Enter an existing name and location	Error displays a message: "This supplier is already existing"
6	Enter empty details (one or more)	Display message: "Please fill in the fields".
7	Enter valid details	Display a message: "Registration succeed, please wait for approvment".
<b>Approve/Reject suppliers' form</b>		
8	Accept the supplier's request for registration.	Display a message: "Supplier confirmed"
9	Reject supplier's request for registration.	Display a message: "Supplier's request rejected".
<b>Add new courier</b>		
10	Enter empty details (one or more)	Error displays a message: "Please fill in the fields".
11	Enter an existing id	Display a message: "This courier is already in the system".
12	Enter an existing phone number	Display a message: "This phone number is already in use".
13	Enter an existing email address	Display a message: "This email address is already in use".
14	Enter an existing username	Display a message: "This username is already in use".
15	Enter valid details	Display a message: "Courier registration succeeds, he can now log in with the username and password".

## 6.2 Functional Tests

No#	Test Subject	Expected result
<b>Login form</b>		
1	Login with a valid supplier username and password	Move to the supplier's form.
2	Login with a valid courier username and password	Move to the courier's form.
3	Login with a valid manager username and password	Move to the manager's form.
<b>Supplier's registration form</b>		
4	Register with a valid detail	The request will appear on the list on the manager form.
<b>Approve/Reject suppliers' form</b>		
5	Accept supplier's registration	The supplier's status will change from 'pending_confirmation' to 'approved'
6	Reject supplier's registration	The supplier's status will change from 'pending_confirmation' to 'Frozen'
<b>Add new courier</b>		
7	Enter a valid detail	The courier can log in with the given username and password
<b>Add new order</b>		
8	Submit a new order from the supplier's form.	The order will appear in the order list in the manager's form.
9	The optimal courier for the mission has been found.	The courier gets an SMS that includes the delivery details.
10	The courier confirms that he will handle the order	The order's status will change from 'pending' to 'in progress'.
11	The courier updates that he finished his mission	The order's status will change from 'in progress' to 'completed'
<b>Monthly updates</b>		
12	A courier has less efficiency than the requirements for the first time	The courier's status will be changed from 'Ok' to 'Warning'.
13	A courier has less efficiency than the requirements for the second time	The courier's status will be changed from 'Warning' to 'Fired' and his account will be frozen.
14	A supplier has '30' orders for the last month.	The supplier will receive a bill that includes the amount of payment he has to pay "30 * X".

## 7. References

### [1] Nearest Neighbor Algorithm:

Cover, T. M., & Hart, P. E. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1), 21-27.

Dasgupta, S., & Schulman, L. J. (2008). Analysis of a greedy algorithm: The Lazy Greedy Algorithm. In *Proceedings of the 39th annual ACM symposium on Theory of computing* (pp. 671-680).

### [2] Genetic Algorithm:

Holland, J. H. (1975). *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*. University of Michigan Press.

Goldberger, J., & Ben-Arie, O. (2002). Genetic algorithms for optimization and machine learning. *IEEE Transactions on Evolutionary Computation*, 6(4), 369-380.

### [3] Machine Learning Algorithms:

Breiman, L. (2001). Random forests. *Machine learning*, 45(1), 5-32.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3), 273-297.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436-444.

### [4] Ant Colony Optimization:

Dorigo, M., & Stützle, T. (2004). *Ant colony optimization*. MIT press.

Bullnheimer, B., Hartl, R. F., & Strauss, C. (1999). An ant colony optimization approach to the generalized traveling salesman problem. In *Proceedings of the 1999 Congress on Evolutionary Computation-CEC99* (Cat. No. 99TH8406) (Vol. 3, pp. 1472-1477). IEEE.

[5] Find the best delivery management software. *Best Delivery Management Software - 2023 Reviews, Pricing and Demos*. (n.d.). Retrieved January 4, 2023, from <https://www.softwareadvice.com/fleet-management/delivery-management-comparison/>