AC51049 - Database Systems Development

Coursework 2 - Cover Sheet

TEAM NUMBER …11…. COMPANY NAME: BUNIQ…

COMPANY SLOGAN: BE BOLD, BUNIQ

TEAM MEMBERS:

Rakshith	Ramureddy	(2514367)
Aditya	Nikumbh	(2551339)
Quoc	Tran	(2528082)
Roseline	Chibuzor Inyamah	(2585389)
Divya	Parmar	(2547318)
Benjawan	Hommak	(2524213)
Abiola	Ayinla	(2528603)
Hung	Tran	(2533373)
Panuchart	Laorungrat	(2521240)
Rohit	Advani	(2540267)

□ Report Number of pages: Content 13, Appendix 9

□ Report's first page includes instructions for accessing your database using MySQL Workbench.

□ Database with MySQL tables will be left on AWS servers

□ We confirm that the team members have read and understood the University policy on Academic Misconduct

You will also need to complete a personal peer review at

`peer review.computing.dundee.ac.uk`

(standard Computing login required)

Instructions for accessing your database using MySQL Workbench

Database login creds
DB identifier: database-team11
End point: database-team11.csbby9agbxem.us-east-1.rds.amazonaws.com
username: admin
password: password

BUNIQ user types consist of,

1. Customer Users

2. Customer Relationship Management Users

3. Logistics Management Users

4. Human Resource Management Users

Regarding the user types, we have two main website interfaces including external and internal UI. The difference in the interface type is because we mainly operate with four different types of users, which have different rights to access the company database. While with external users we develop websites to support the online shopping, profile, purchase, and return. In another UI, the internal users mainly focus on the operational function of CRM, Logistics and Human Resource departments.

The UI interface indeed has a direct relationship with the company's database. In each website interface, we manage to maintain a high level of data manipulation and data warehouse management by creating a wide range of connections and linked maps to view, insert, update and delete data between UI (website interface) and Sever (Database Warehouse SQL). Each table contains various attributes and sample data, which will briefly clarify what tables we extract from the company database for both internal and external UI, as an overview of the data volume in each utilised table (Table 1).

For the Home page, we use the table Product, Category, Customer, Payment, Invoice, Address

For the Profile page, we use the table Customer, Address,

For the Purchase page, we use the table Product, Order, Invoice, Payment

For the Return page, we use the table Return, Order, Feedback, Customer

Entity	**Tuple**	**Attribute**
Employee	152	10
Product	176	6

Customer	151	10
Category	176	2
Return	31	5
Supplier	11	5
Payment	151	3
Invoice	151	5
Address	420	3
Order	437	10
Delivery	151	5
Feedback	150	6

Table 1. BUNIQ Database

User Authorisation

Authorisation ensures that users can only access the resources or perform actions that they have been explicitly allowed to, preventing unauthorised access and ensuring data security. BUNIQ user authorisation has this process involves.

1. User Authentication

Before authorisation, the system authenticates users to verify their identity through login credentials like usernames, passwords, or multifactor authentication.

2. Access Control Policies

Once authenticated, the system applies access control policies that determine what actions or operations a user can perform and what resources they can access. These policies are based on predefined rules or permissions associated with the user's role or specific attributes.

3. Permission Assignment

Permissions are assigned to users or groups within the system, specifying the level of access granted. This include read, write, update, delete, execute, or custom permissions based on the system's requirements.

4. Role-Based Access Control (RBAC)

RBAC is rights are assigned based on the roles users have within an organisation.

5. Fine-Grained Access Control

Allowing specific permissions to be set on individual data items or resources rather than entire sections of the system.

1. Customer Users

The Customer User interface page allows user to input their email ID and password to log in. If new customers visit the page, they can create an account with BUNIQ by clicking the 'create account' button at the bottom of the webpage. Users can also log in using their existing Gmail account, Facebook or Apple account. The customer table will use Username, Password, and Email from the Database created **(Figure 1).**

The primary purpose of a login page is to verify the identity of users attempting to access protected areas of a website. User can also often manage their accounts through the login page, updating their profile information, changing passwords, or accessing account settings **(Appendix Customer UI).**

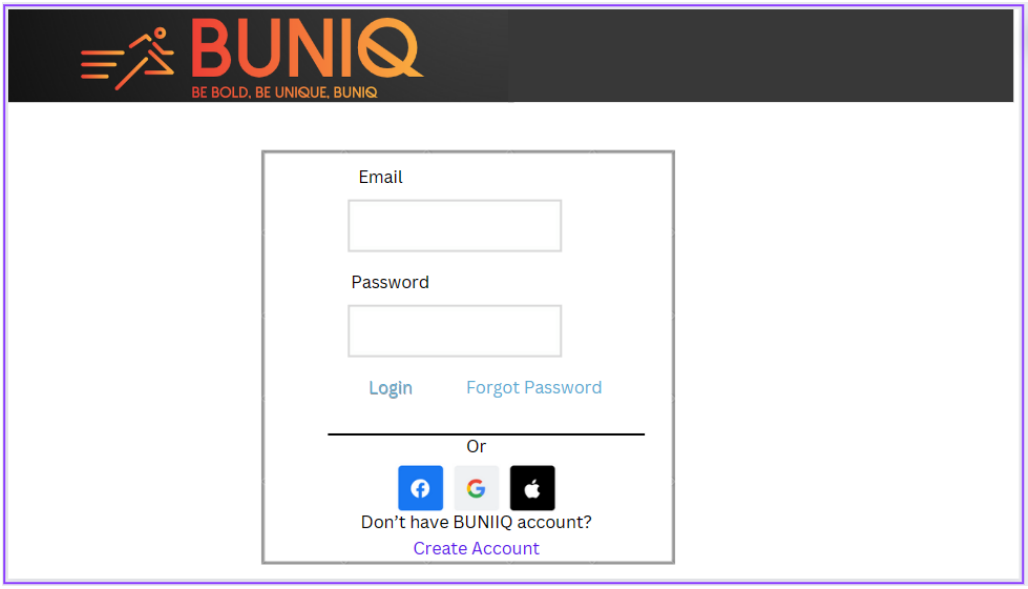

Figure 1. Login page for BUNIQ company users

1.1 Product Categories

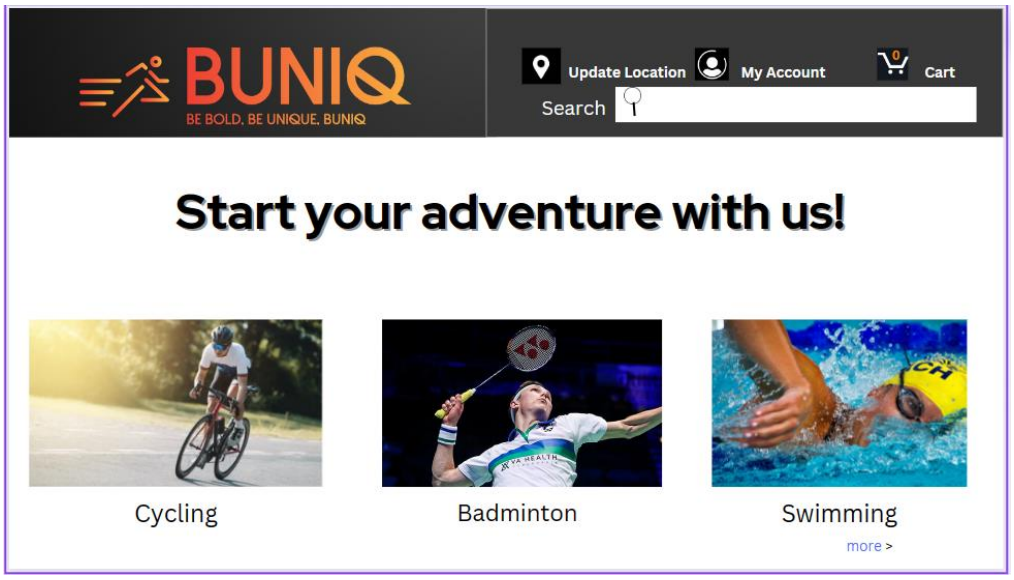

Figure 2. Product category page

After logging in, the customer will be led to the category webpage, which shows a user interface about categories like category_id and type from the Category Database table. BUNIQ offers classes in Cycling, Badminton and swimming, from which customers can choose and pick their choice of product to order (Figure 2).

The customer Category webpage is used to personalise category results. For example, a customer searching for information about cycling equipment could be shown more using cycling category results.

1.2 Customer profile

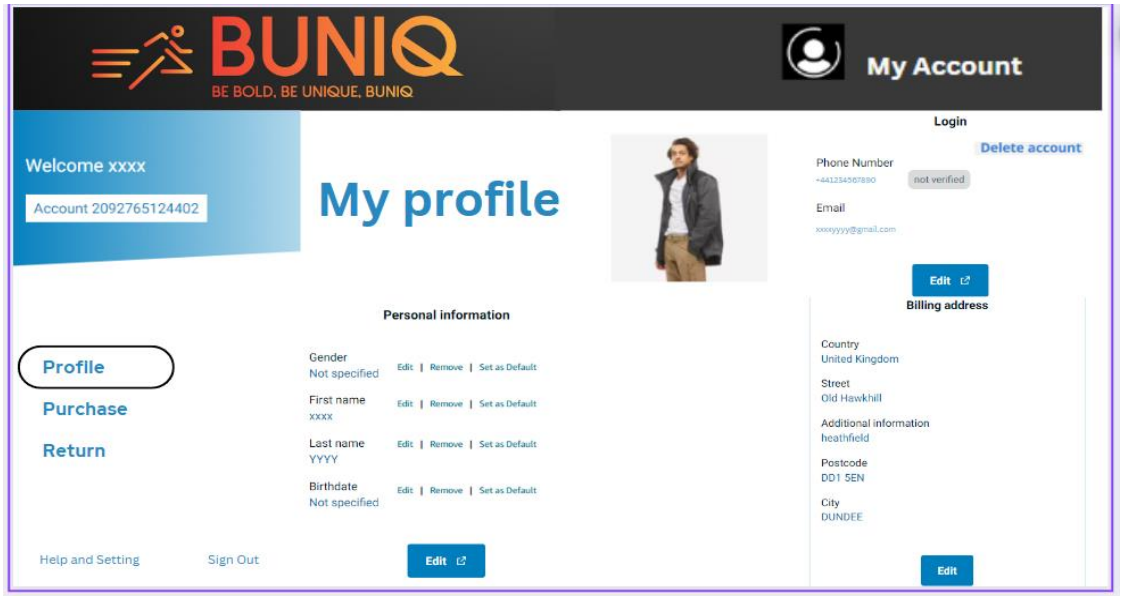

Figure 3. My profile page

Customer can view their profile once they click on My Account on the top right of the webpage **(Figure 3)**. It will give information about Customer ID, First name, Last name, Date of Birth, Phone number, Email, Username, Password, Payment info, Address, Postcode and Address and Customer_ID table from the Database tables.

Employee profiles are digital representations of employees showing their detailed information and employee status, behavioural, and attitudinal data. These profiles are crucial in website design, marketing strategies, and employee engagement.

1.3 Customer order profile

Once the customer has chosen from categories, they will be led to the My Purchase webpage; it gives an order status and list of purchases with Product ID, Category, Product Name, Size, Price (£), Address, Postcode, Order ID, Order Date, Returns ID(if applicable), Delivery ID, Payment_method and Order Status as mentioned in Database Product and Address and Order table. Customers can see their order, address and payment status **(Figure 4)**.

Customer Order webpage lets customers conveniently place orders for products or services directly from the company's website. A customer order webpage often includes descriptions, images, and product listings so customers can browse and discover items before ordering. As a result of this visual representation, customers have an enhanced shopping experience and can make informed decisions.

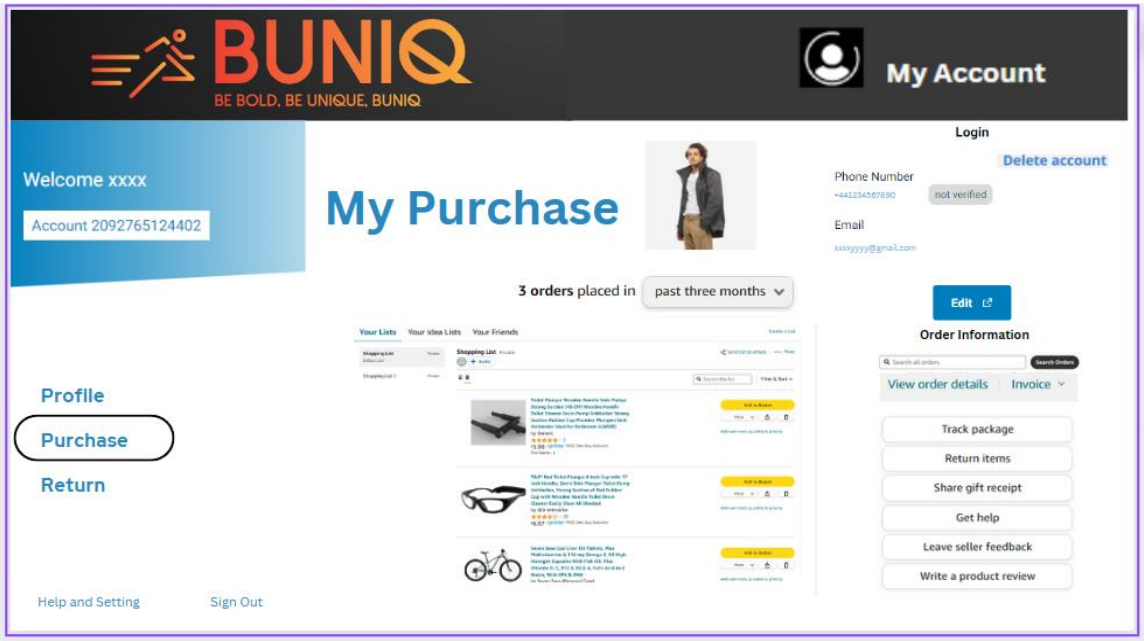

Figure 4. Customer order page

1.4 Order Return

Order return can be applicable when a customer requests a return for their product. It generates the Return ID, Order ID, Return Reason, Product ID and Refund Amount (£) from the Returns table from the database. The page also displays the billing address from the Address table, encompassing attributes like Address and Postcode (Figure 5).

The main motive of an order return page on a website is to provide customers with a convenient and streamlined process for returning or exchanging purchased items. It is a central hub for initiating, tracking, and managing return requests, ensuring a smooth and hassle-free customer experience.

Figure 5. Return page

Employee Login page

Like the customer login webpage, the employee also has a login page by providing their unique employee ID and password. The employee login webpage encompasses the Employee ID, First Name, Last Name, Address, Email, Designation and Phone Number from the Employee Database table (Figure 6).

BUNIQ webpage allows their employee to get help from the IT support team whenever there are login or password reset issues by just clicking the 'IT support contact' button on the webpage, which will raise a ticket, and the IT team will be notified of the issue.

The primary purpose of an employee login page is to authenticate the identity of employees attempting to access restricted areas of the website by using their unique username and password. This helps to ensure that only authorised individuals can access sensitive data, perform critical tasks, or manage confidential information.

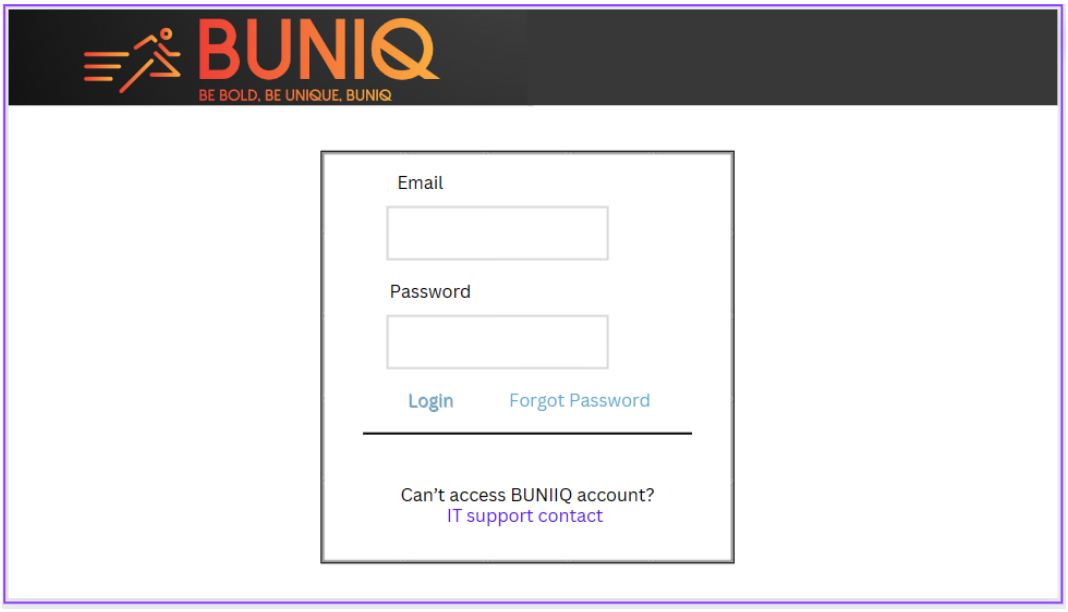

Figure 6. Login page for BUNIQ for company users

2. Customer Relationship Management Users

Figure 7. Main page for Customer Relationship Management users

Customer Relationship Management (CRM) UI oversee interactions with existing and prospective customers. Within our system, we've developed two distinct features: Feedback Management and Return Management **(Figure 7)**.

2.1 The purpose of the Feedback Management UI is to facilitate the systematic handling of feedback provided by customers regarding their orders and overall experience. It serves as a tool to collect, organise, and analyse feedback data related to product ratings, staff interactions, facility experiences, and specific review keywords. Through this interface, users can efficiently view, retrieve, and assess various feedback aspects, enabling insights into customer satisfaction, areas for improvement, and overall performance evaluation.

In feedback management, the entities encompass feedback, order, and customer, interconnected for data retrieval and analysis.

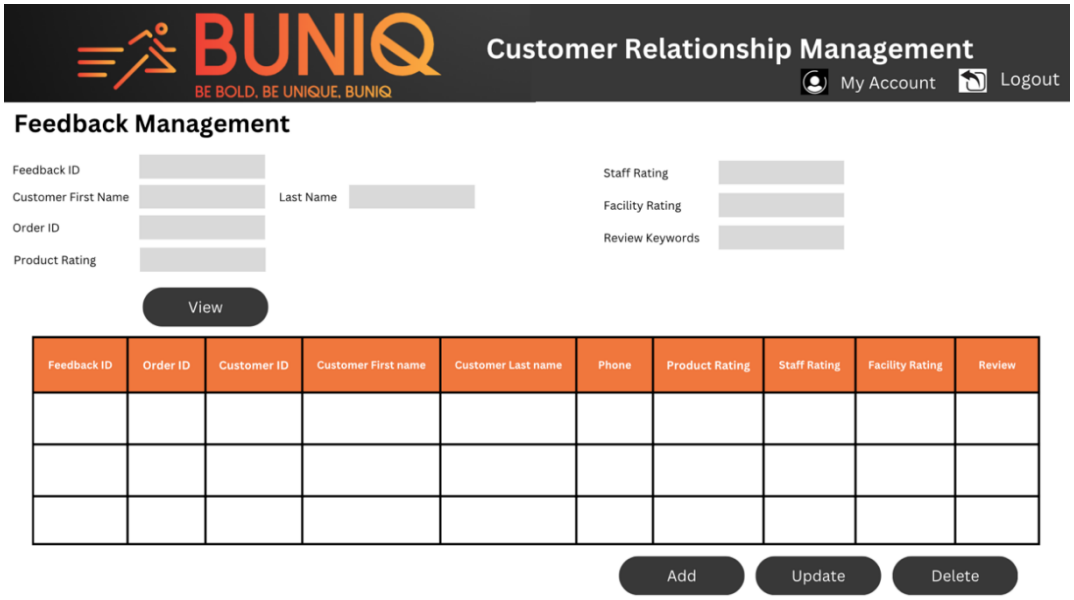

Figure 8. Feedback Management UI

The criteria for viewing data include Feedback ID, Customer First Name and Last Name, Order ID, Product Rating, Staff Rating, Facility Rating, and Review Keywords. It's not mandatory to fill out all criteria; anyone is sufficient to access data. Then click the 'View' button to Read the information **(Appendix Feedback Management).**

The retrieved table displays Feedback ID, Order ID, Customer ID, Customer First Name, Customer Last Name, Phone, Product Rating, Staff Rating, Facility Rating, and Reviews **(Figure 8)**.

2.2 The Return Management UI is designed to effectively manage and process return-related data within the system. Its purpose is to handle information regarding returns, orders, and customers in an interconnected manner, facilitating data retrieval and analysis.

The Return Management, the entities encompass Return, Order and Customer, interconnected for data retrieval and analysis.

Figure 9. Return Management UI

The criteria for viewing data include Return ID, Order ID, Customer ID, Customer First name and Last name, Customer Phone, Return Amount range (GBP) and Return Reason Keywords. It's not mandatory to fill out all criteria; anyone is sufficient to access data. Then click the 'View' button to Read the information **(Appendix Return Management).**

The retrieved table displays the Return ID, Order ID, Customer ID, Customer First name, Customer Last name, Phone, Email, Return Amount and Return Reason **(Figure 9)**.

This user type typically performs Create, Read, Update, and Delete (CRUD) operations as follows,

Create, the users can insert new records into the table. For instance, a CRM user could input a new customer feedback entry by manually entering relevant details such as Customer ID, Product Ratings, and Reviews based on direct communication from customers contacting via phone to express concerns or complaints about the products they've ordered.

Read, Data retrieval is flexible, allowing users to access information using various criteria like Feedback ID, Return ID, Customer First Name, etc. Details such as ratings, reviews, return amounts, and associated data can be viewed.

Update: the users can modify existing records. For example, CRM users can access the specific form and make necessary changes if there's an error in a customer's phone number or a need to edit a review.

Delete: the users can remove no longer needed records that contain inaccurate information. CRM users with appropriate permissions can delete that specific record from the table if a particular feedback entry is redundant or incorrect.

3. Logistics Users

The Logistics Management UI facilitates interactions with both existing and potential operations. We've developed three distinct features within our system: Inventory Management, Delivery Management, and Supplier Management **(Figure 10)**.

Figure 10. Main page for Logistics Management users

3.1 The Inventory Management UI serves as a comprehensive tool to oversee and manage inventory-related data within the system. Its primary objective is to handle information associated with Inventory, Branches, and Product entities in an interconnected manner, allowing for efficient data retrieval and analysis.

Figure 11. Inventory Management UI

The criteria for viewing data include Receiving Date range, Product ID, Product Name, Unit in Stock range, Inventory ID, Branch ID and Branch Name. It's not mandatory to fill out all criteria; anyone is sufficient to access data. Then click the 'View' button to Read the information **(Figure 11)**.

The retrieved table displays Inventory ID, Branch ID, Branch Name, Product ID, Product Name, Receiving Date, Price per Unit, Unit in Stock, and Total Inventory Value **(Appendix Inventory Management).**

3.2 The Delivery Management UI is designed to oversee and manage delivery-related information within the system. Its primary function is to handle data associated with Delivery, Order, and Product in an interconnected manner, facilitating efficient data retrieval and analysis.

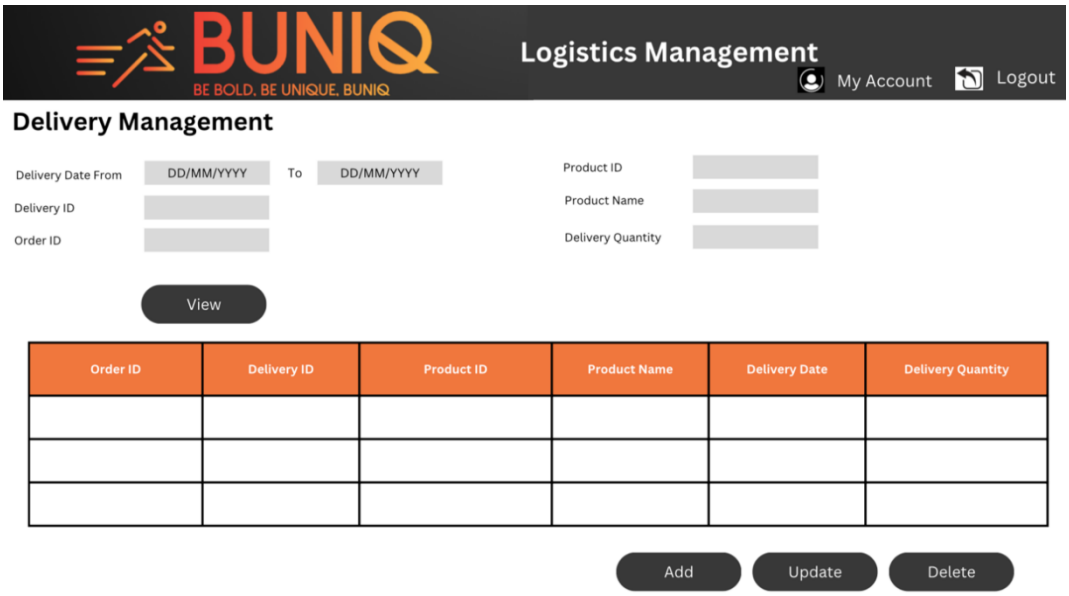

Figure 12. Delivery Management UI

The criteria for viewing data include Delivery Date range, Delivery ID, Order ID, Product ID, Product Name and Delivery Quantity. It's not mandatory to fill out all criteria; anyone is sufficient to access data. Then click the 'View' button to Read the information **(Appendix Delivery Management).**

The retrieved table displays the Order ID, Delivery ID, Product ID, Product Name, Delivery Date and Delivery Quantity **(Figure 12)**.

3.3 The Supplier Management UI is designed to facilitate efficient interactions with supplier-related data, providing users with the ability to access and analyse information related to suppliers and their associated products. By encompassing entities of Supplier and Product.

Figure 13. Supplier Management UI

The criteria for viewing data include Supplier ID, Supplier Name, Product ID, Product Name, Phone and Email. It's not mandatory to fill out all standards; anyone is permitted to access data. Then click the 'View' button to Read the information (Figure 13).

The retrieved table displays Supplier ID, Supplier Name, Product ID, Product Name, Phone and Email (Appendix Supplier Management UI).

Logistics management users typically engage in various operations within the system, including Create, Update, and Delete (CRUD) actions as follows,

This user type can create new records across Inventory, Delivery, and Supplier entities. For instance, we add inventory details, record delivery information, or input new supplier data.

Read, Data retrieval is flexible, allowing users to access information using multiple criteria. Users can view inventory details, delivery records, or supplier information based on specific filters, such as date ranges, product IDs, or supplier names.

Update: Authorised users can modify existing records. This includes updating inventory quantities, altering delivery details, or editing supplier contact information.

Delete: this user type with appropriate permissions can remove outdated or redundant records. This functionality eliminates obsolete inventory, outdated delivery entries, or irrelevant supplier data.

4. Human Resource Management Users

The Human Resource Management UI facilitates interactions with employees. Within our system, we've developed a feature called Employee Management. This UI aims to enhance organisational efficiency by enabling comprehensive oversight and informed decision-making regarding human resource management processes.

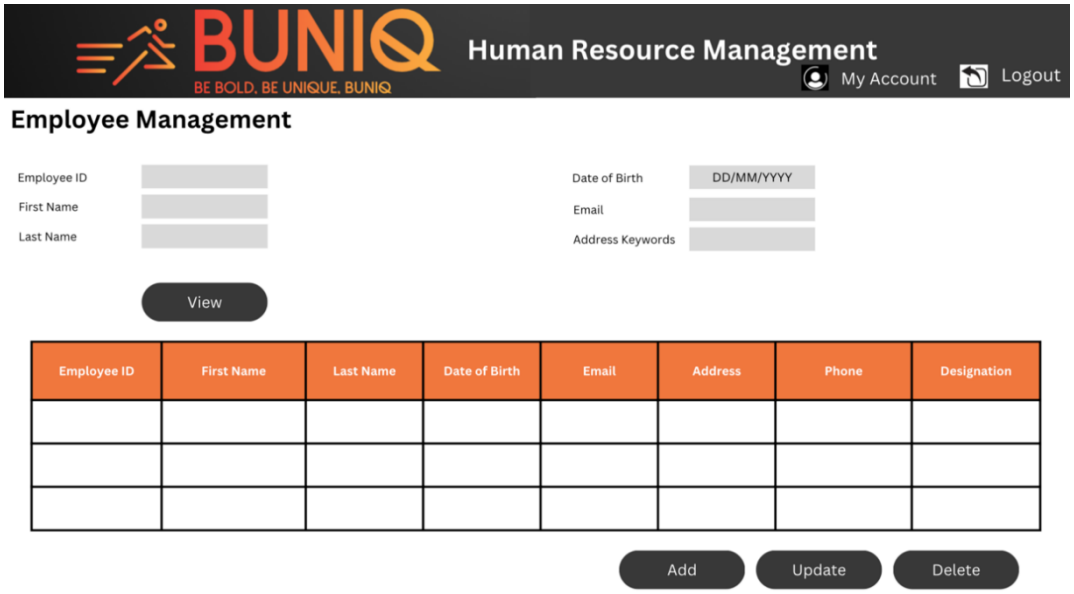

Figure 14. Employee Management UI

The Employee Management, the entities encompass Employee and Address, interconnected for data retrieval and analysis.

The criteria for viewing data include Employee ID, First Name, Last Name, Date of Birth, Email, and Address Keywords. It's not mandatory to fill out all criteria; anyone is sufficient to access data. Then click the 'View' button to Read the information (Figure 14).

The retrieved table displays Employee ID, First Name, Last Name, Date of Birth, Email, Address, Phone and Designation (Appendix Employee Management UI).

The Employee Management feature within the Human Resource Management UI allows CRUD functionalities for handling employee-related data as follows.

Creating this user type can create new employee records to the system. For example, they can input details like Employee ID, First Name, Last Name, Date of Birth, Email, Address, Phone, and Designation to create a new employee profile.

This user type can view employee details using specific criteria like Employee ID, First Name, Last Name, Date of Birth, Email, Address, Phone, or Designation. Filling in these criteria is sufficient to access and retrieve employee information. Clicking the 'View' button displays the requested data in a table format, including Employee ID, First Name, Last Name, Date of Birth, Email, Address, Phone, and Designation.

Update: this user type with appropriate permissions can modify existing employee records. They can edit contact details, addresses, or other relevant employee data.

Delete: this user can remove employee records that are no longer required. This action might occur if an employee leaves the company or the data becomes obsolete.

Appendix

Appendix Customer UI

#login

SELECT

customer_ID,

firstName,

lastName

FROM customer

WHERE username = 'AMYY71' AND password_ = 'hfu73';

#this is to access their profile using creds

SELECT

customer_ID,

firstName,

lastName,

DOB,

phoneNumber,

email,

paymentInfo,

address_ID

FROM customer

WHERE username = 'AMYY71' AND password_ = 'hfu73';

#TO create a new profile

INSERT INTO customer (customer_ID, firstName, lastName, DOB, phoneNumber, email, username, password_, paymentInfo, address_ID)

VALUES (

'C111565',

'EUGENE',

'HUANG',

'1965-05-14', -- Replace with the actual date of birth

'020 94981833', -- Replace with the actual phone number

'eugene10@gmail.com',

```
'EUGENEH65',
'Q(y1+@lYDH',
'8156 9121 4825 2234', -- Replace with the actual payment information
2 -- Replace with the actual address_ID
);

DELETE FROM customer
WHERE customer_ID = 'C111565';
```

Appendix Feedback Management

```
#feedback_:
#Viewing all feedback_ entries
SELECT *
FROM feedback_mgmt;

-- Inserting a new feedback_ record
INSERT INTO feedback_mgmt(
feedback_ID,
order_ID,
customer_ID,
firstName,
lastName,
phoneNumber,
product_rating,
staff_rating,
facility_rating,
review)
VALUES (
'FB002',
'O124',
'C457',
```

'Jane',

'Smith',

'987-654-3210',

5,

4,

5,

'Outstanding selection of products and excellent customer service.');

#Updating an existing feedback_ entry

UPDATE feedback_mgmt

SET Review = 'Updated review: customer service has improved significantly.'

WHERE feedback_ID = 'FB002';

#Deleting a feedback_ entry

DELETE FROM feedback_mgmt

WHERE feedback_ID = 'FB002';

Appendix Return Management

#return_:

#Viewing all return_ entries

SELECT *

FROM return_mgmt;

-- Inserting a new return_ record

INSERT INTO return_mgmt(

return_ID,

order_ID,

customer_ID,

firstName,

lastName,

phoneNumber,

```
email,
Return_amt,
return_reason)
VALUES (
'RT001',
'O567',
'C890',
'Alex',
'Johnson',
'321-654-0987',
'alex.johnson@email.com',
'100.00',
'Defective product');

#Updating an existing return_ entry
UPDATE return_mgmt
SET Review = 'Updated review: customer service has improved significantly.'
WHERE return_ID = 'RT002';
#Deleting a return_ entry
DELETE FROM return_mgmt
WHERE return_ID = 'RT003';
```

Appendix Inventory Management

```
#ADD DATA
INSERT INTO inventory_mgmt (
branch_ID,
receving_Date,
total_InventoryQTY)

VALUES ('BBB1000', '01/01/2024', 500);
```

```
INSERT INTO inventory_mgmt (
product_category,
price_per_unit)
VALUES ('Badminton Clothing','150');

INSERT INTO inventory_mgmt (supplier_name) VALUES ('BAS');
INSERT INTO inventory_mgmt(branch_Name) VALUES ('kAS');

#EDIT
SET SQL_SAFE_UPDATES = 0;

UPDATE inventory_mgmt
SET total_InventoryQTY = 550
WHERE inventory_ID = 'BP5776';

SET SQL_SAFE_UPDATES = 1;

#DELETE
SET SQL_SAFE_UPDATES = 0;

DELETE FROM inventory_mgmt
WHERE inventory_ID = 'BP5776';

SET SQL_SAFE_UPDATES = 1;
```

Appendix Delivery Management UI

```
INSERT INTO delivery_mgmt (
order_ID,
delivery_ID,
product_ID,
```

```
product_name,
Delivery_Date,
delivery_qty)
VALUES (
'I138884',
'SHP555666777',
'BadBad001',
'Badminton Clothing',
'2023-12-24',
1 );
UPDATE delivery_mgmt
SET
product_Name = 'Badminton Clothing',
Delivery_Date = '2023-12-24',
delivery_qty = 1
WHERE
delivery_ID = 'I138884';
DELETE FROM delivery_mgmt
WHERE delivery_ID = 'I138884';

SELECT *
FROM delivery_mgmt;

SELECT *
FROM delivery_mgmt
WHERE delivery_ID = 'I138884';

SELECT *
FROM delivery_mgmt
WHERE Delivery_Date
BETWEEN '2023-12-24' AND '2023-12-24';
```

```
SELECT *
FROM delivery_mgmt
WHERE product_name
LIKE '%BadmintonClothing%';
```

Appendix Supplier Management UI

```
#Inserting a new record into the Suppliers table:
INSERT INTO supplier_mgmt (
supplier_ID,
supplier_name,
product_ID,
product_name,
phone_number,
email)

VALUES (
'S001',
'Highland Sports',
'P100',
'Football',
'123-456-7890',
'contact@highlandsports.com');

#Updating the supplier name from 'Highland Sports' to 'HighlandSporty'
SET SQL_SAFE_UPDATES = 0;

UPDATE supplier_mgmt

SET supplier_name = 'HighlandSports'
WHERE supplier_name = 'Highland Sports';
```

#Selecting and viewing all data from the Suppliers table:

```
SELECT *
FROM supplier_mgmt
WHERE supplier_ID = 'S001';
```

#Deleting records with the supplier name 'HighlandSporty':

```
DELETE FROM supplier_mgmt
WHERE supplier_name = 'HighlandSporty';
SET SQL_SAFE_UPDATES = 1;
```

Appendix Employee Management UI

#Insert:

```
INSERT INTO employee_mgmt (
employee_ID,
first_Name,
last_Name,
DOB,
email,
address,
phoneNumber,
designation)
VALUES (
'104',
'Rob',
'Walters',
'1974-07-23',
'rob0@buniq.com',
'1825 Village Pl.',
'612-555-0100',
'sales assistant');
```

```
SELECT *

FROM employee_mgmt;


#View:

SELECT *

FROM employee_mgmt

WHERE first_Name

LIKE '%Rob%' OR last_Name LIKE '%Walters%';


# Update:

SELECT *

FROM employee_mgmt

WHERE employee_ID = '104';

SET SQL_SAFE_UPDATES = 0;

UPDATE employee_mgmt

SET first_Name = 'Rob', last_Name = 'Walters', email = 'rob0@buniq.com', phoneNumber = '612-555-0100',
designation = 'sales assistant'

WHERE employee_ID = '104';


# Delete:

DELETE FROM employee_mgmt

WHERE employee_ID = '104';
```