1

# DRONE DELIVERY SYSTEM

A PROJECT REPORT

## *In partial fulfilment for the course*

*Of*

DRONE DELIVERY SYSTEM

GROUP: CLUSTERS

TEAM MEMBERS:

1. *Avinita Shetty*
2. *Neha Reddy*
3. *Pragalbha Yadav*
4. *Priyanka Kadam*
5. *Vishwa Shah*

2



NORTHEASTERN UNIVERSITY

BOSTON, MASSACHUSSETTS

## APRIL 2015

3

### Acknowledgement

We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. We would like to extend our sincere thanks to all of them.

We are highly indebted to Prof. Chaiyaporn Mutsalklisana for his guidance and constant supervision as well as for providing necessary information regarding the project & also for his support in completing the project.

We would like to express our gratitude towards the members of Northeastern University for their kind co-operation and encouragement which help us in completion of this project.

We would like to express our special gratitude and thanks to our Teaching Assistant Tanmay Ingle and Vaibhav Pattani for giving us his attention, time and valuable suggestions.

Our thanks and appreciations also go to our colleague in developing the project and people who have willingly helped us out with their abilities.

4

**INDEX**

|  |  |  |
| --- | --- | --- |
| **Sr. No** | **Title** | **Page no.** |
| **1** | **Problem Statement** |  |
| **2** | **Limitations** |  |
| **3** | **Use Case Diagram** |  |
| **4** | **EER Diagram** |  |
| **5** | **Normalization** |  |
| **6** | **Views** |  |
| **7** | **Procedure** |  |
| **8** | **Users and Privileges** |  |
| **9** | **Triggers** |  |
| **10** | **Transactions** |  |
| **11** | **Indexes** |  |
| **12** | **Conclusion and Future Scope** |  |
| **13** | **References** |  |
| **14** |  |  |
| **15** |  |  |

5

#### Introduction

Technology is evolving at a rapid phase, to keep up with the demands and needs of people across the globe. One such technical innovation is the rapid expansion in the usage of Drones. Drones are unmanned aerial vehicles that are used mostly to aid the military and combat services. Drones are faster and less prone to human error, and hence have potential to take over industries where speed and accurate delivery is a key criteria. There are several industries such as healthcare and ecommerce industries that are looking to invest in this technology to improve the services that they provide.

The objective of our project is to build a database for a typical ecommerce website which includes the use drones to deliver their products rather than the traditional methods of delivery.

It includes details about customers using the services of this website, along with the drone details and their related information. It also includes data about products and payment information, which shows the jest of the functionality of this website with this kind of delivery service.

The drones are going to use zip codes to find the exact path that the drone takes from the warehouse address to the customer shipping address. We have added the weather capabilities to the drone as an extra functionality, which might benefit the company in better understanding how the system and overcome certain difficulties.

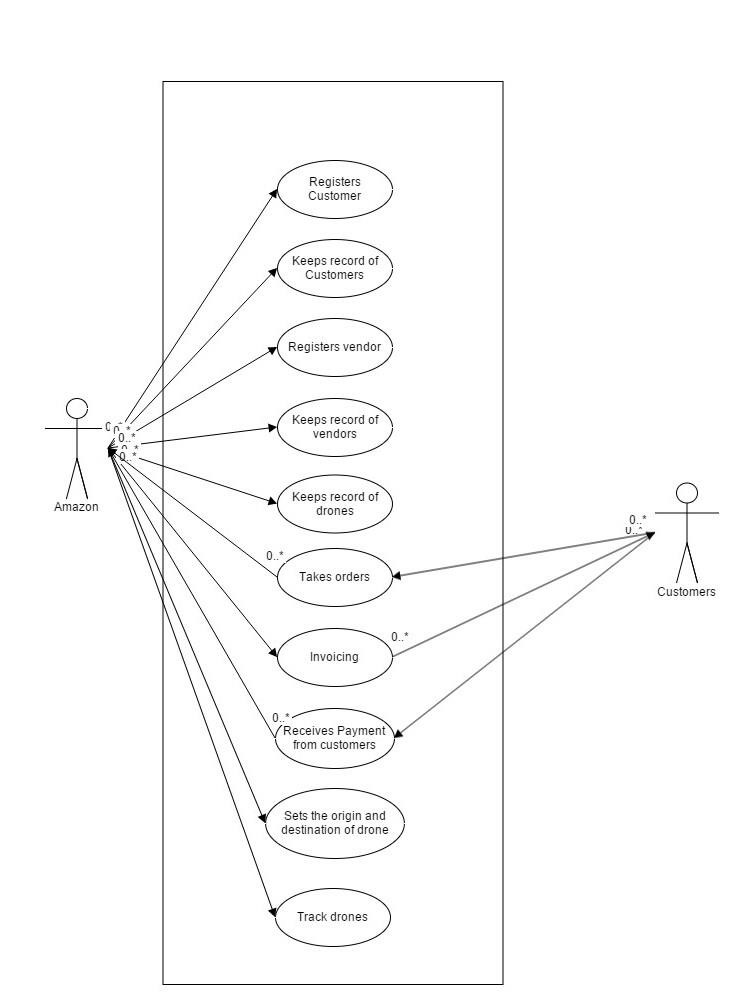
6

#### Limitations

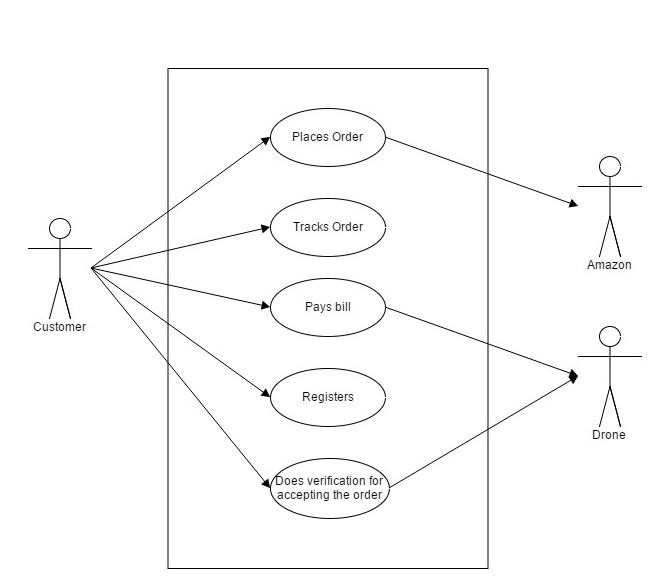
* The e-commerce website is assumed to be amazon.com.
* It has vendors and customers from USA only.
* Warehouses includes vendor warehouses and amazon warehouses.
* The product is delivered by a drone present in that product vendor warehouse.
* Customers can order only one product at a time.
* One drone will carry only one order item at one time.
* No cash on delivery option.
* The customer is assumed to get package directly, i.e drop off at the zip code location.
* No partial transactions are taken into consideration.
* The order item is created only after the payment is completed.
* One category is associated to only one department.

7

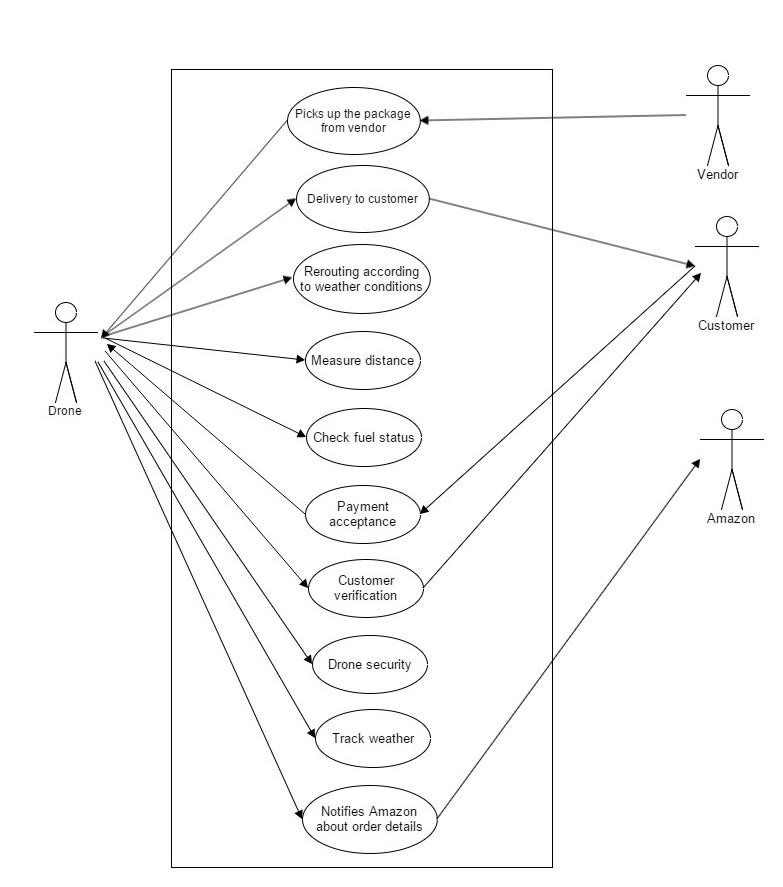
**Use Case Diagrams**



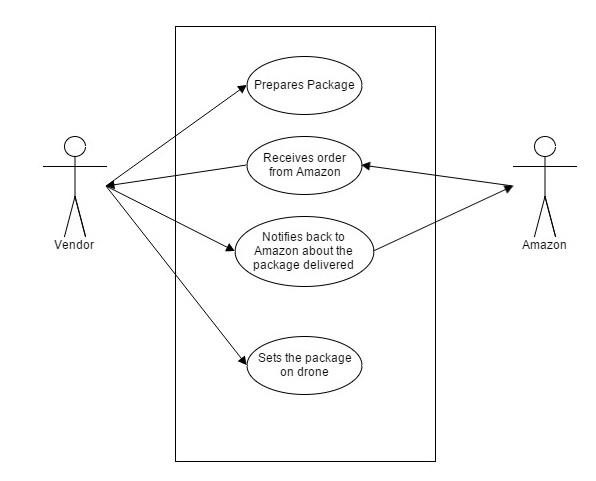
8



9

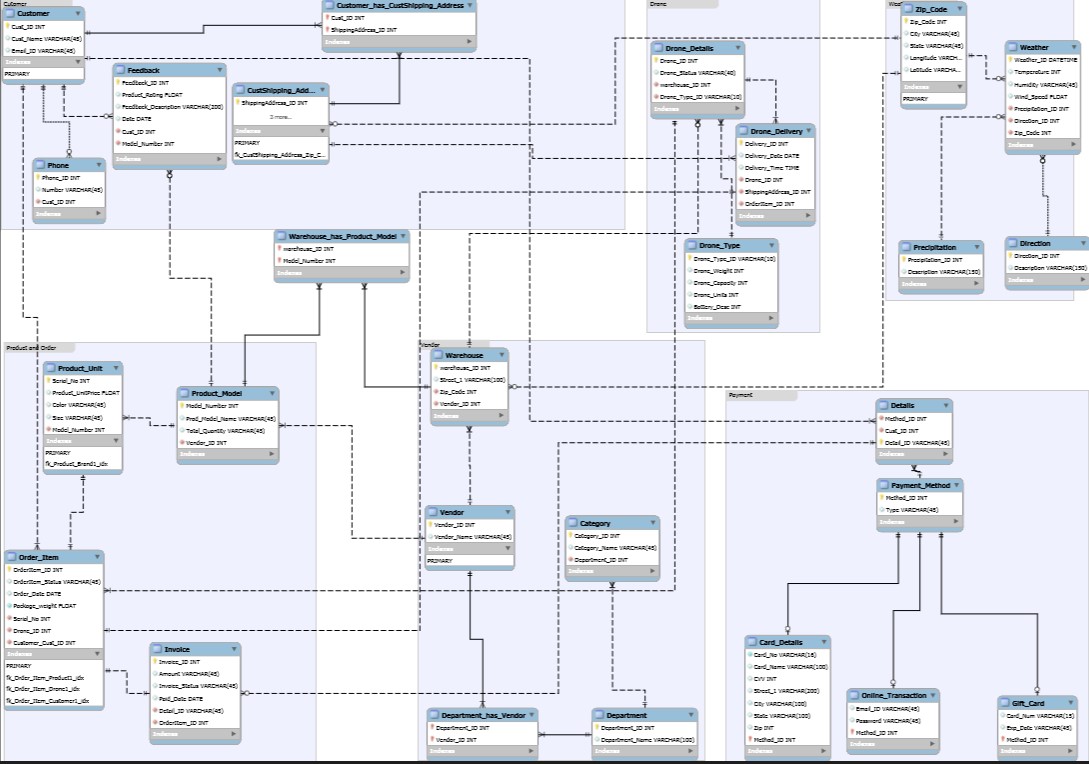


0



1

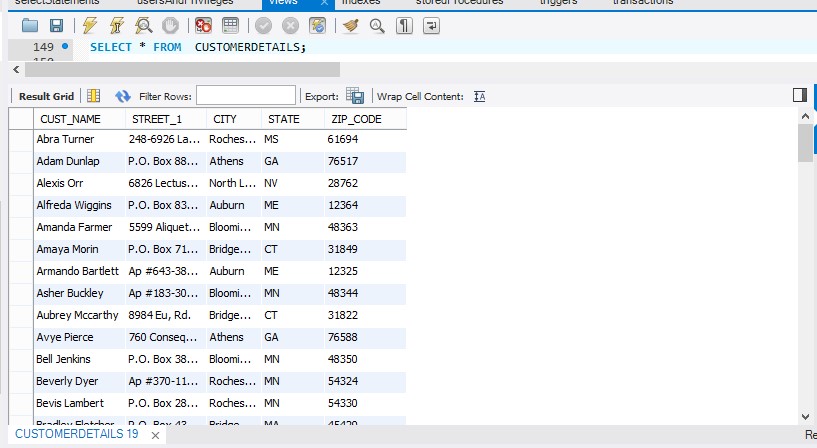
**EER DIAGRAM**



2

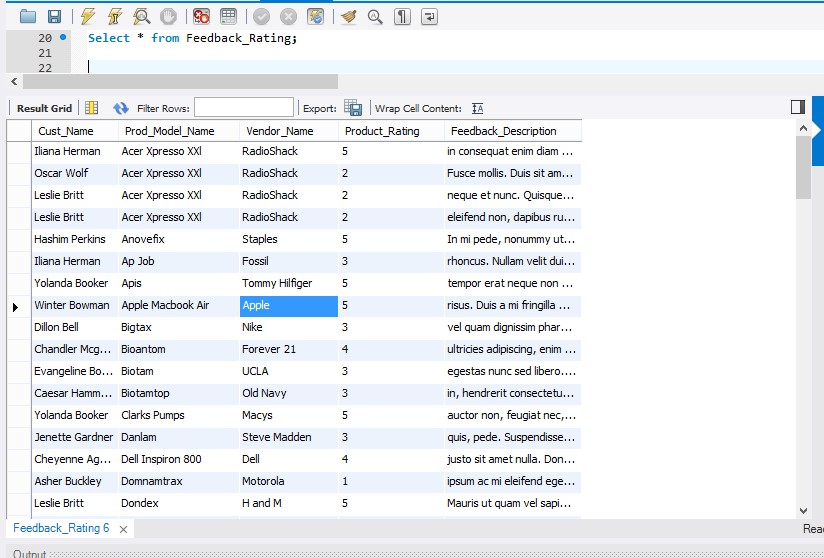
**Views**

* 1. **Customer Details View**: This view will display all the customer details like name and address.



3

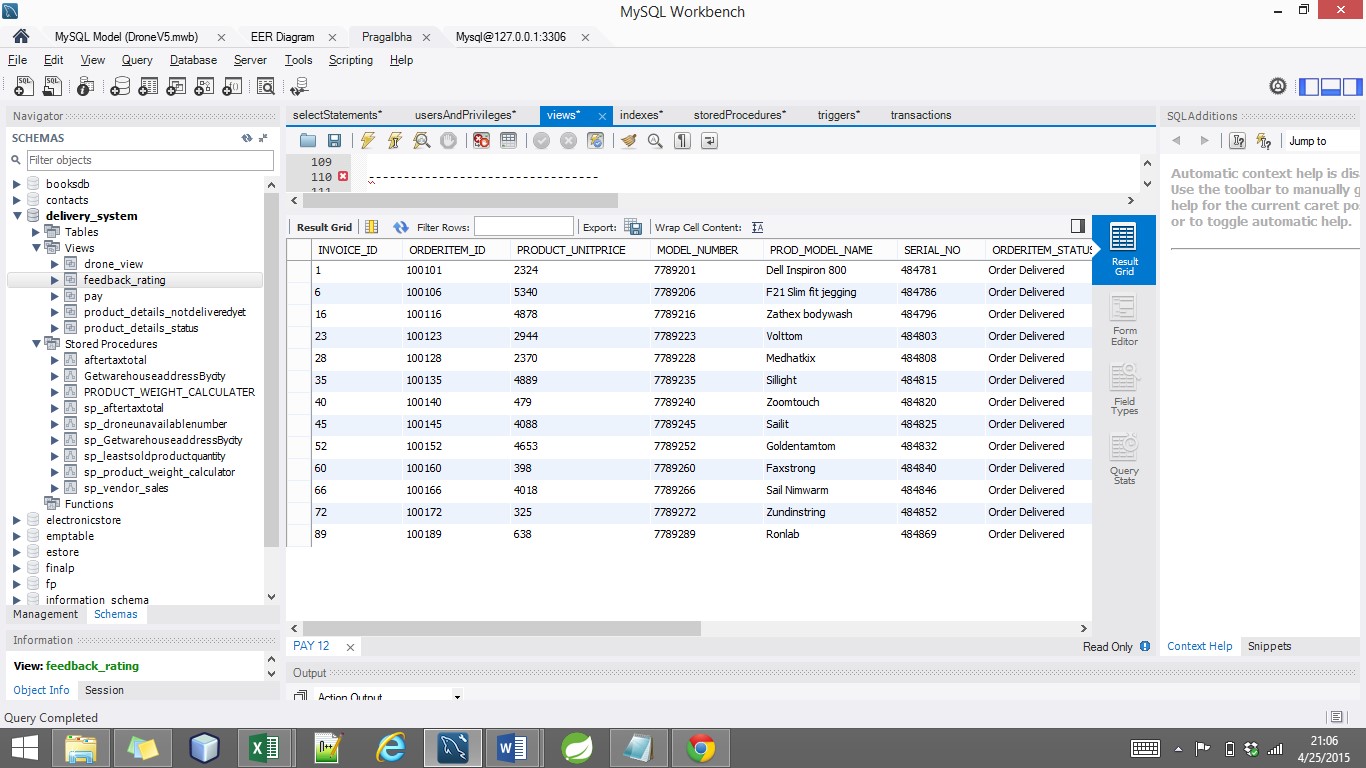
* 1. Feedback Rating View: It is a view which will allow the customers and vendors to see the feedback of the various products.



4

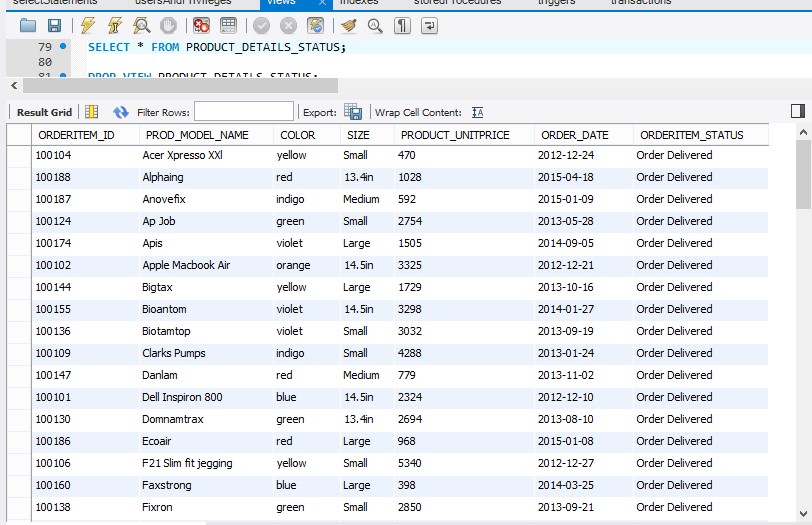
* 1. Payment View: This will include the customer details, card details , invoice details and

product details



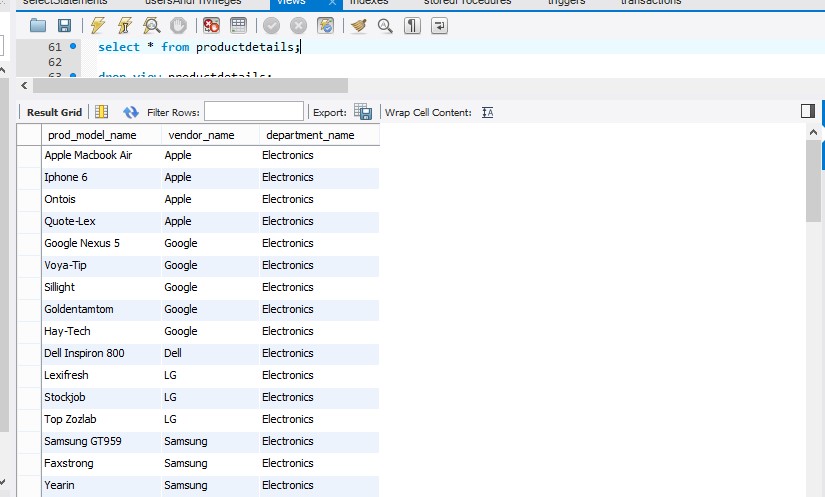
5

* 1. **Product Details Status View**: To check the status of the product.



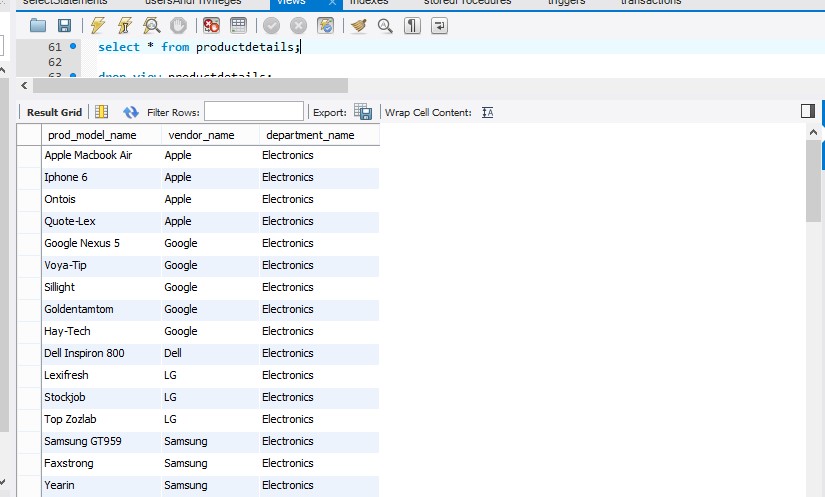
6

* 1. **Product Details Status View**: To check the status of the product.



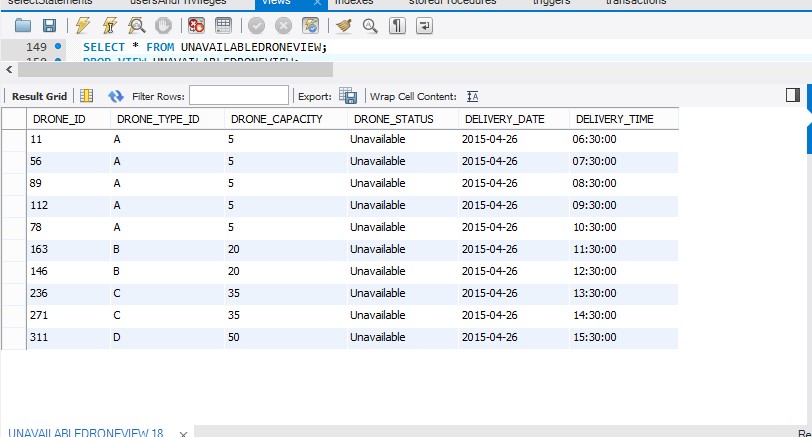
7

* 1. **Product Details View**: To check the status of the product.



8

* 1. **Unavailable Drone View**: This view will display all the drones which are unavailable.



1

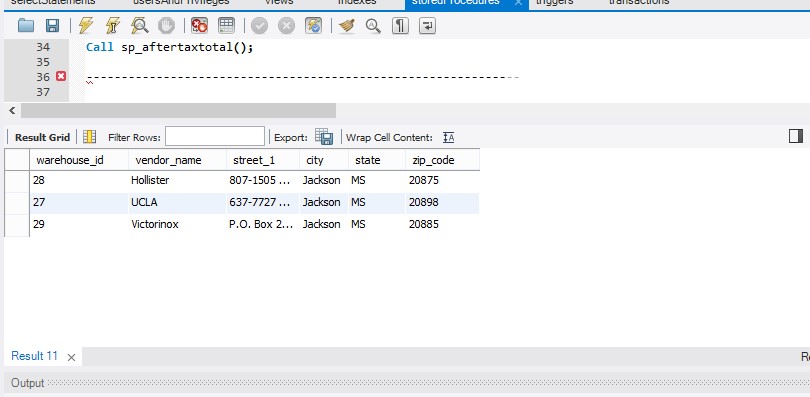
9

#### Stored Procedures

1. Main Procedure:A procedure to assign a Drone\_Id whenever an order is placed by a customer. This takes into consideration the package weight and compares it with the drone capacity to find an appropriate match with the drones that are available in the warehouse.

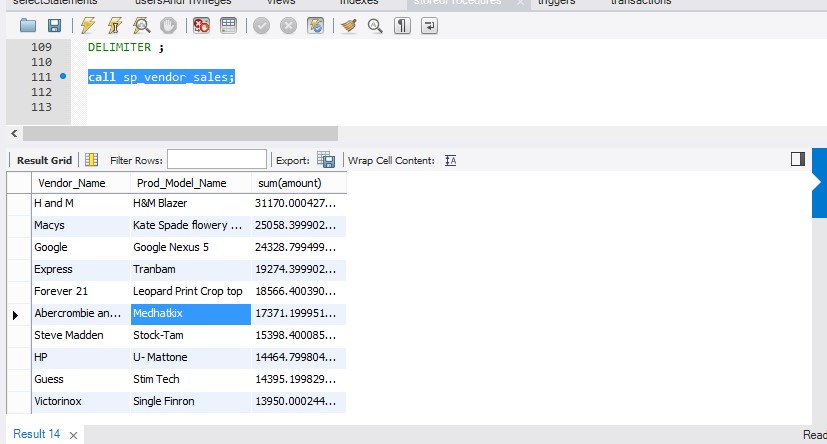
0

1. AfterTaxTotal: A procedure to get total amount invoice by adding tax to the product unit price.



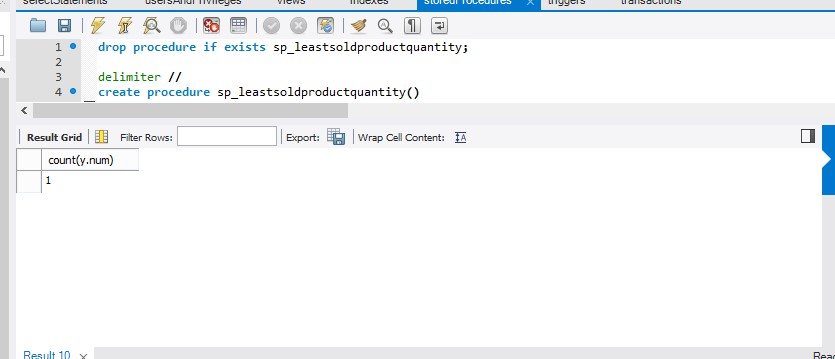
1

1. Vendor Sales: A procedure to find the top ten vendors with the product model that has the maximum sales.



2

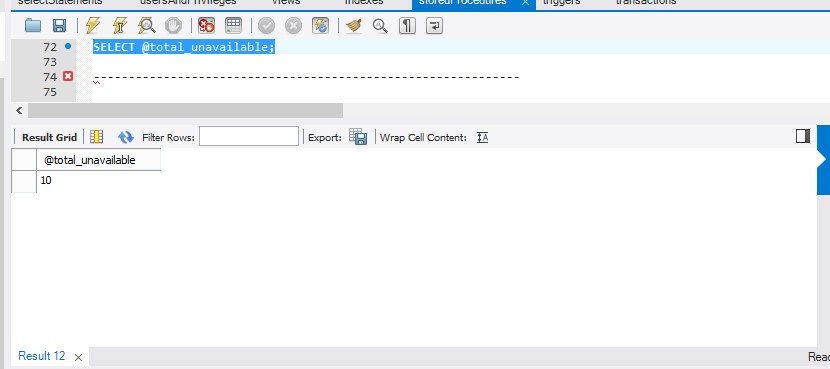
1. Least Sold Product Quantity: A procedure to find the least sold product



3

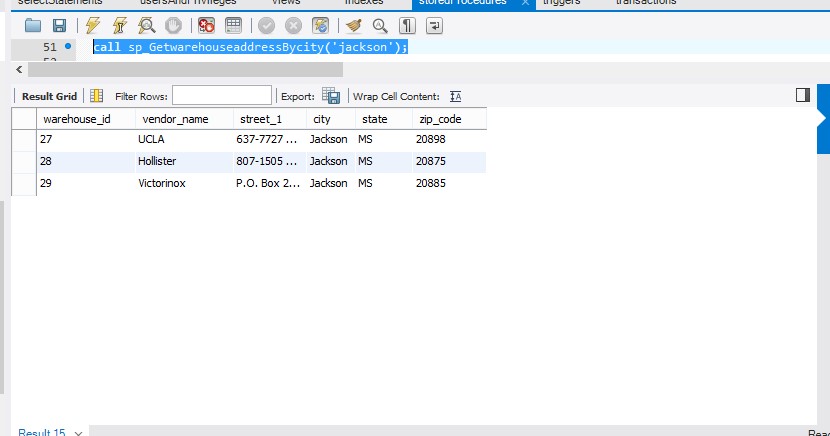
##### Parametric Procedures

1. Drone Unavailable Number: Parametric OUT procedure to count the number of drones with a particular status.



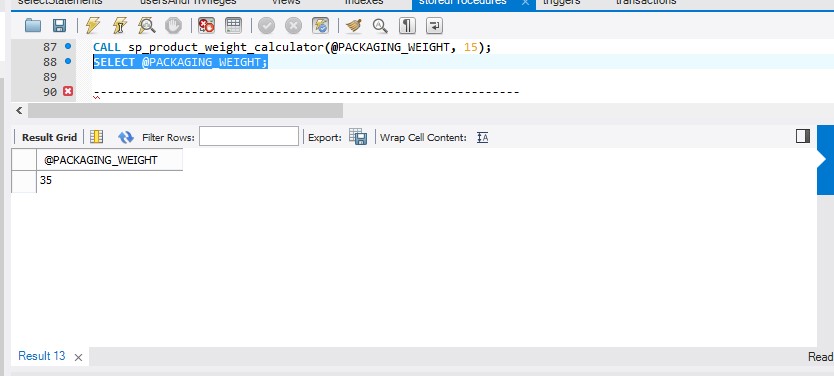
4

1. Get Warehouse Address by City: Parametric IN procedure to get warehouse address by inputting a particular city.



5

1. Product Weight Calculator: Parametric INOUT procedure to calculate approximate package weight.



6

#### TRIGGERS

1. **A trigger to reduce the quantity of the Product\_model when an item is inserted into the order\_item table. This reduces the total quantity available by 1.**

Delimiter //

Create trigger reducequantity after insert on order\_item for each row begin

update product\_model pm inner join product\_unit pu on pm.Model\_number=pu.Model\_Number

inner join order\_item oi on oi.serial\_no= pu.serial\_no set pm.total\_quantity= pm.total\_quantity-1 ; end //

drop trigger reducequantity;

1. **A trigger to find the average feedback rating in product\_model, when a feed feedback is entered in the feedback table.**

Delimiter //

Create trigger averagefeedback after insert on feedback for each row begin update product\_model pm inner join feedback fb

on

pm.Model\_number=fb.Model\_Number set pm.average\_rating= num where

##### num=(sum(fb.product\_rating)/count(fb.product\_rating));

end //

7

#### TRANSACTION

* In almost all applications that access SQL databases, multiple users concurrently attempt to view and modify data. The simultaneous operations may result in data that is inconsistent and inaccurate –Or worse, database corruption. Using “Transactions” avoids these problems by isolating each operation.
* Transaction is a set of one or more SQL statements that perform a set of related actions.
* The statements are grouped together and treated as a single unit whose success or failure depends on the successful execution of each statement in the transaction.



start transaction; savepoint savepoint1;

select @OrderItem\_ID:=max(OrderItem\_ID) from order\_item; set @OrderItem\_ID=@OrderItem\_ID+1;

insert into

order\_item(OrderItem\_ID,OrderItem\_Status,Order\_Date,Package\_weight,Serial\_No,Drone\_ID) values (@OrderItem\_ID,"Order InProcess",now(),3.79,484784,12);

rollback to savepoint savepoint1;

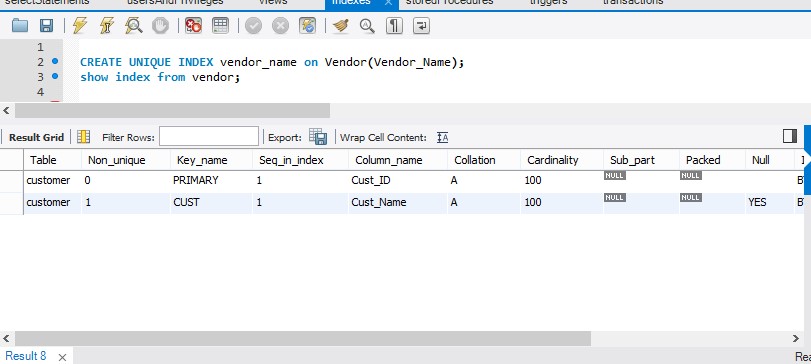
insert into

order\_item(OrderItem\_ID,OrderItem\_Status,Order\_Date,Package\_weight,Serial\_No,Drone\_ID) values(@OrderItem\_ID,"Order InProcess",now(),6.78,484786,15); commit;

8

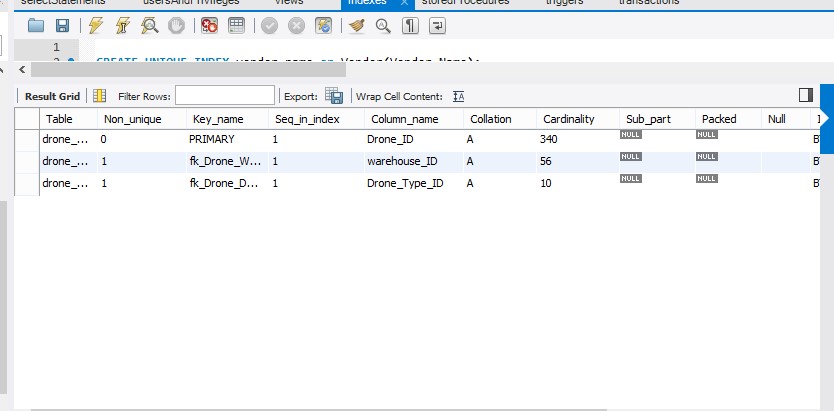
#### INDEXES

 Customer Index: This is an Index on the Customer table using Customer name.



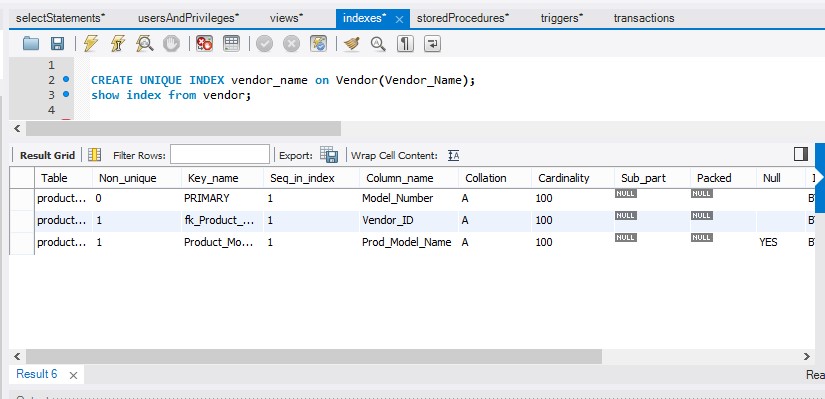
9

Drone Details: This is an Index on the Drone\_Details table using Drone\_ID .



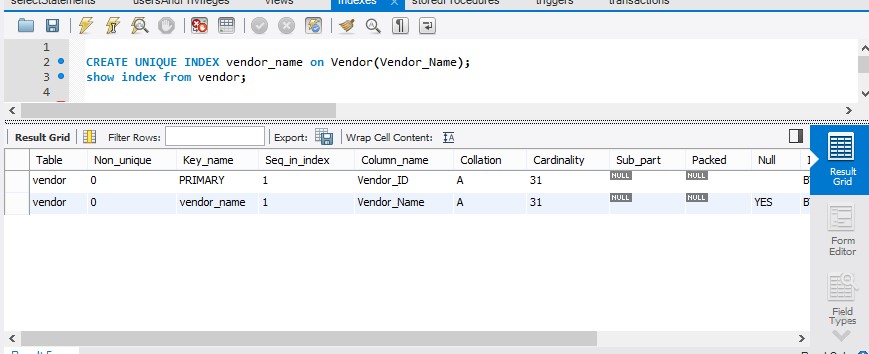
0

Product Model: This is an Index on the Product table using Product\_Model\_name.



1

Vendor: This is an Index on the Vendor table using Vendor\_Name.



2

#### USERS AND PRIVILEGES

MySQL is an open source database management software that helps users store, organize and later retrieve data. It has a variety of options to grant specific users nuanced privileges within the tables and databases.

**Users:**

* Ecommerce Admin:
* Drone Manager:
* Vendor:
* Customer:

**Privileges:**

* Create: allows them to create new tables or databases

#####  Insert: allows them to insert rows into tables

* Select: allows them to use the select command to read through databases  Update: allow them to update table rows.
* Drop: allows them to delete rows from tables.

3

##### Backups

Our Database implements two kinds of backup

* Full Backup
* Incremental Backup

* Since it is an e-commerce website, we have assumed that there is least traffic during the night.
* We have also assumed that there is lesser traffic during the weeknights rather than weekend nights.
* We assume that full backup takes 6 hours to complete.
* We assume that an incremental backup takes about 3 hours to complete.
* If full backup is scheduled on public holidays (Christmas, Thanksgiving) then it will be completed on the previous day as well as the day after the holiday.

|  |  |  |
| --- | --- | --- |
| Day of the week | Type of Backup | Timing |
| Sunday | Incremental Backup | 1:30 am-4:30 am |
| Monday | Full Backup | 12 am-6 am |
| Tuesday | Incremental Backup | 1:30 am-4:30 am |
| Wednesday | Incremental Backup | 1:30 am-4:30 am |
| Thursday | Incremental Backup | 1:30 am-4:30 am |
| Friday | Full Backup | 12 am- 6 am |
| Saturday | Incremental Backup | 1:30 am-4:30 am |

##### Normalization

Database normalization is the process of organizing the fields and tables of a relational database

1. to minimize redundancy. Normalization usually involves dividing large tables into smaller tables and defining relationships between them.

All the tables in the database system are normalized. It involves 1NF, 2NF and 3NF.

Enforcing 1 NF:

A table is in 1NF if it is free from multi-valued rows. In this database system all the tables contain only single values.

Enforcing 2NF & 3NF:

A table is in 2NF if it is in 1NF and the whole of the composite primary key is used to uniquely identify all the other non-key attributes. Attributes which have partial dependent on one part of the composite key alone are removed and formed as a separate table. A table is in 3NF if it is in 2NF and no non-key field depends on a key which is not a primary key. It overcomes the update and deletion anomalies

Customer

Phone

**References**

[**https://dev.mysql.com/**](https://dev.mysql.com/)

1. [**http://www.mysqltutorial.org/**](http://www.mysqltutorial.org/)  [**http://stackoverflow.com/**](http://stackoverflow.com/)  [**http://www.w3schools.com/**](http://www.w3schools.com/)  [**http://generatedata.com/**](http://generatedata.com/)  [**http://www.wikipedia.com/**](http://www.wikipedia.com/)

***Books:***

**MySQL Documentation**

**Modern Database Management**

**Beginning SQL**

**Presentation slides by Prof. Mutsalklisana**