PROJECT DUDEPANDA

Project Members;
-Hasan Naseem, Faraz Ozair and Hozefa Haider

Introduction

Database Management Systems is one of the Course that Computer Science Students at Habib University have to take in order to complete their course requirements. The course aims to target the concept of databases, their management and targets some concepts of Data Science. Part of the Course Requirements is create a project that aims to satisfy a real business scenario, incorporating all the principles taught in the Course.
My friends and I created a website and named it Dudepanda, a website where people can order bakery items and get it delivered to their place. Following is a detailed description of our project and the steps carried out to complete it.

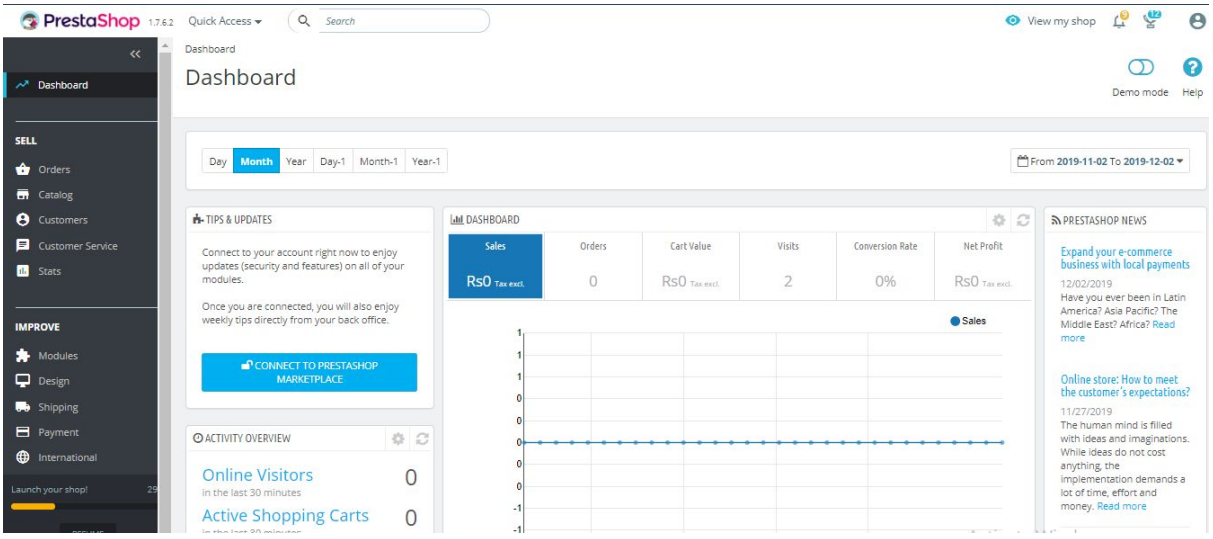
Business Scenario

Dudepanda is a simple ecommerce marketplace of bakeries located in karachi. Customers can save themselves from the hassle of going to the bakery and buying items rather place an order through this website and get it delivered directly to their place like ordering a birthday cake for their loved ones.
The website required a database which would essentially contain the data of All the Vendors, Customers, Food items, Orders etc. So the first job was to decide and create an entity relationship diagram (ERD) that would form the basis of our Database.
Since creating a website and then launching it meant for us to buy a domain, we just made an offline website using Prestashop.

MODULES & FUNCTIONALITY

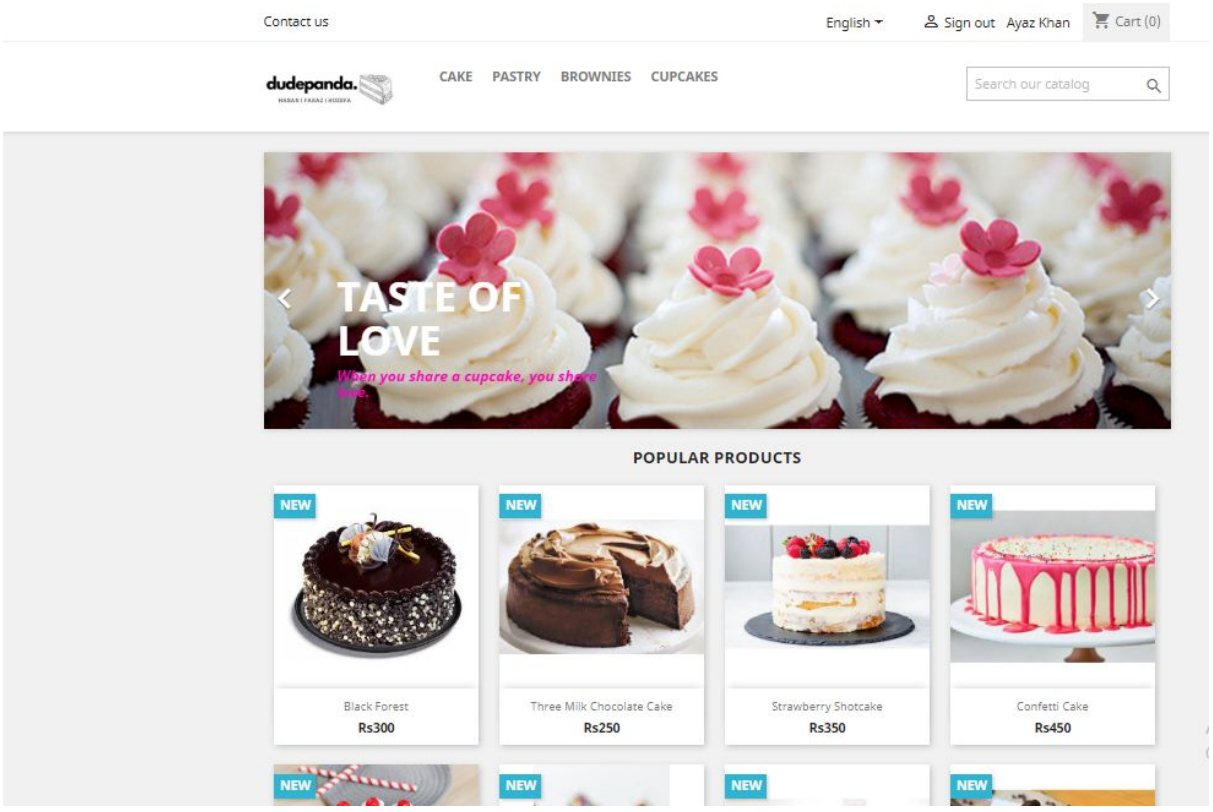
The website has three main interfaces or modules which is the user interface, admin interface and MySQL administration tool.

1) admin interface

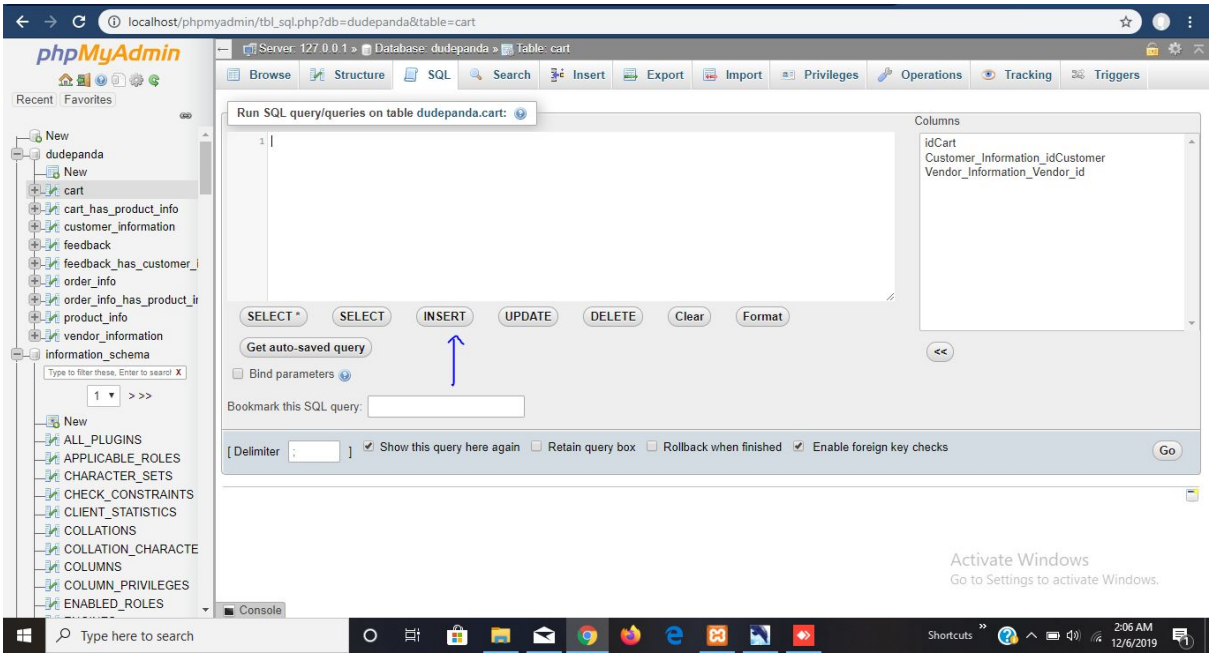


PROJECT DUDEPANDA

2) and user interface/Front end



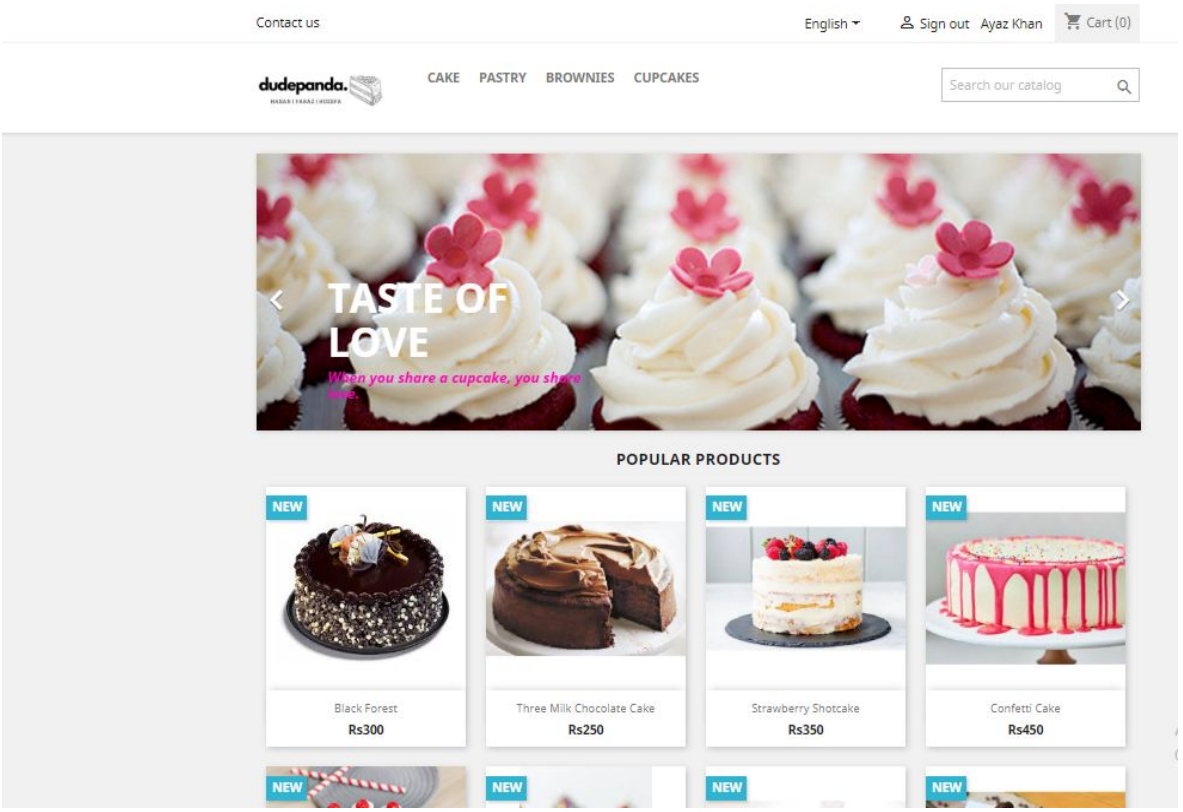
3) MySQL administration tool;



PROJECT DUDEPANDA

Now I'll walk you through some of the functionalities of the website.

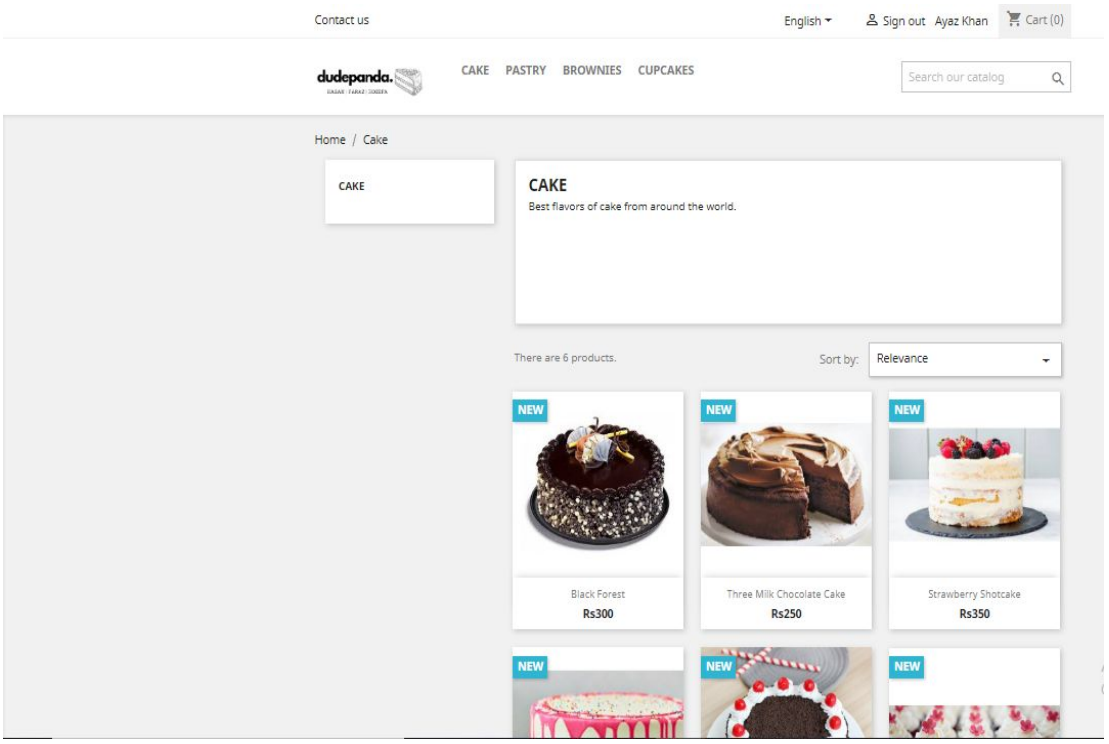
1) **HOW TO ORDER AN ITEM**
    a) Following is the front end;



So let's order a cake

    b) Click on *cake* on the top

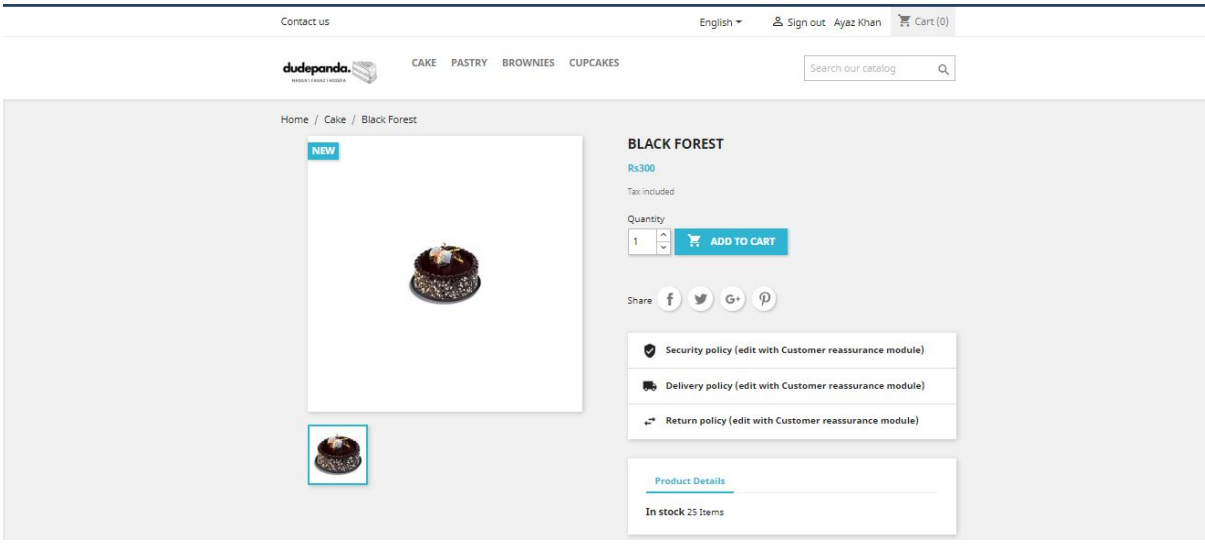
PROJECT DUDEPANDA



The following screen will appear

c) Next choose a cake that you want. I'll be choosing *Black Forest* cuz why not

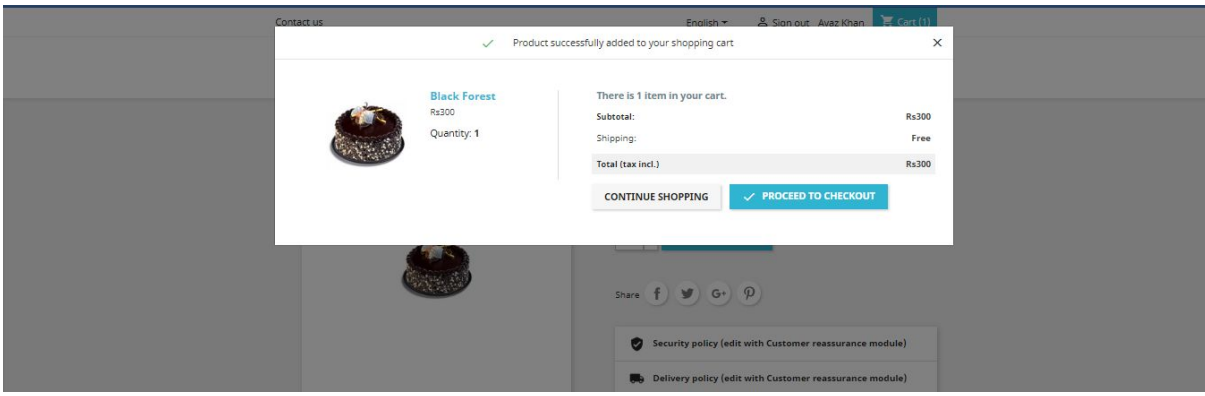
d) The following screen will then appear



Select the quantity and then press add to cart

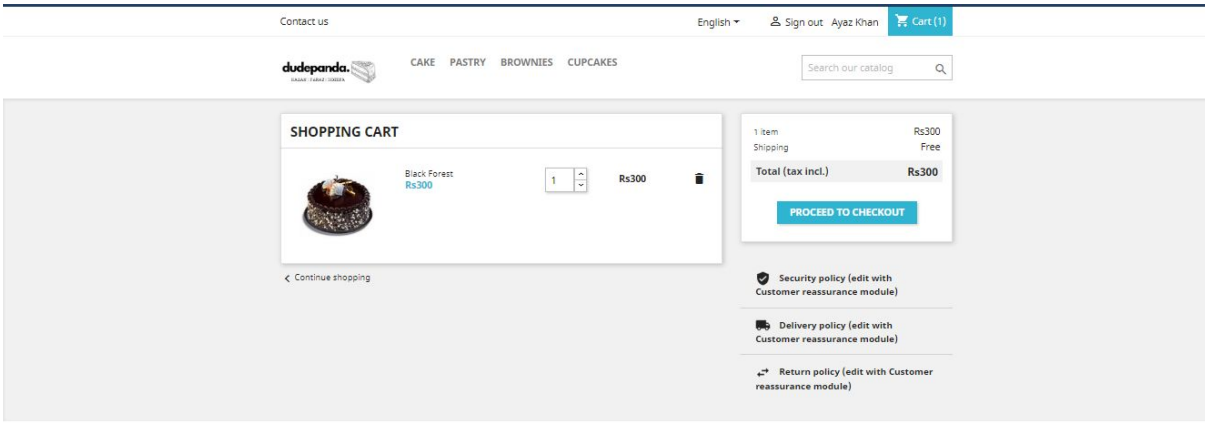
e) On doing so, this window will pop up

PROJECT DUDEPANDA



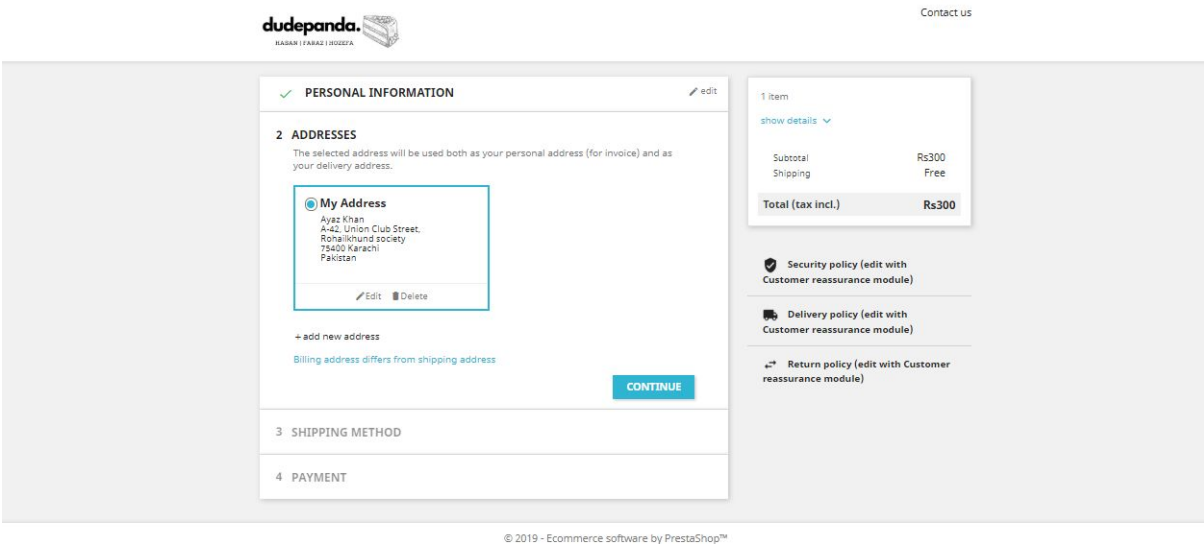
f) If that's all what i want to order, click *Proceed to Checkout*

g)



This window shows the order amount details (shipping cost if any)
click *Proceed to Checkout*

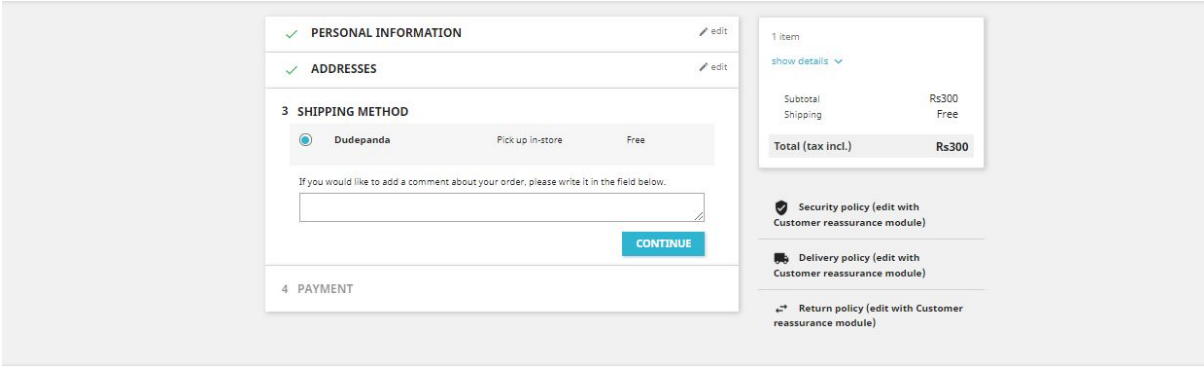
h)



Edit address and then press continue

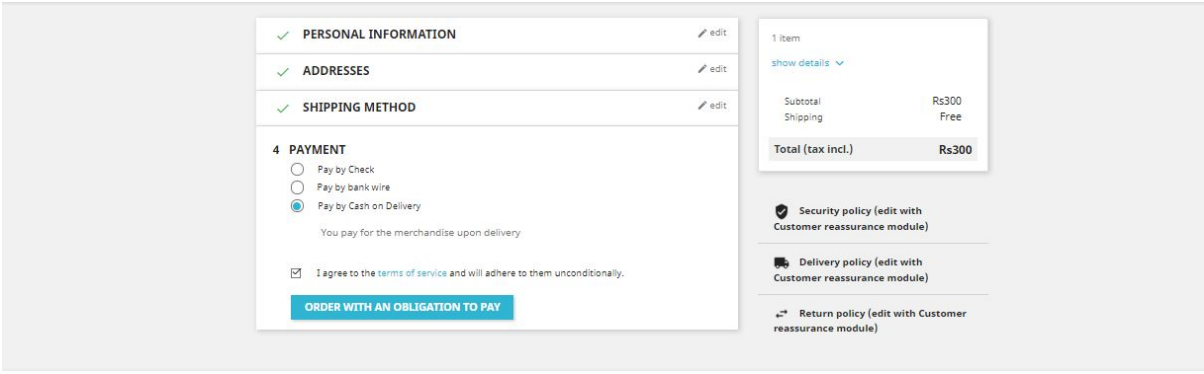
PROJECT DUDEPANDA

i)

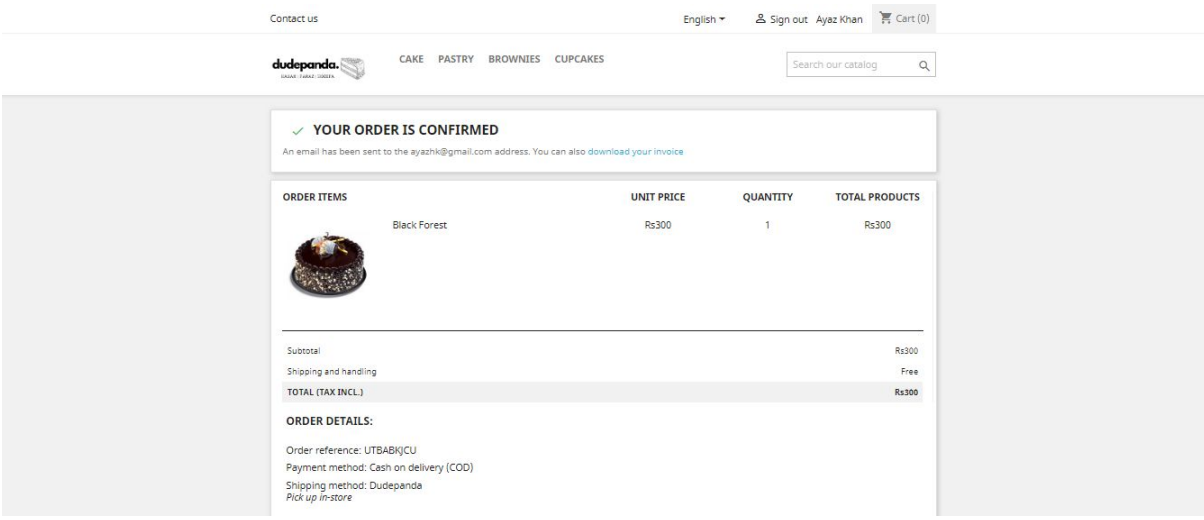


j) Press continue again

k)



l)

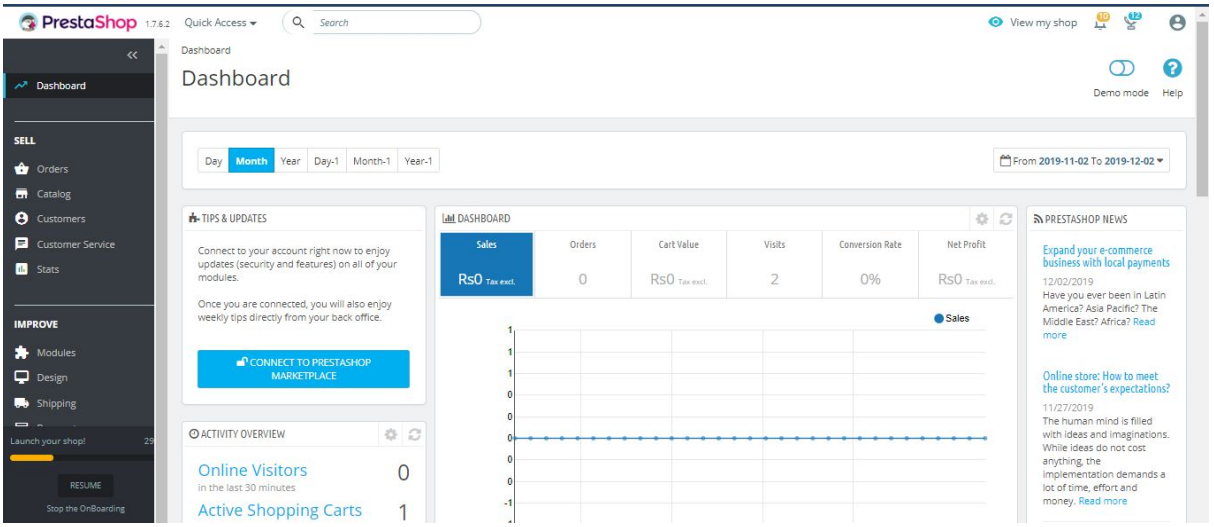


Order successfully placed

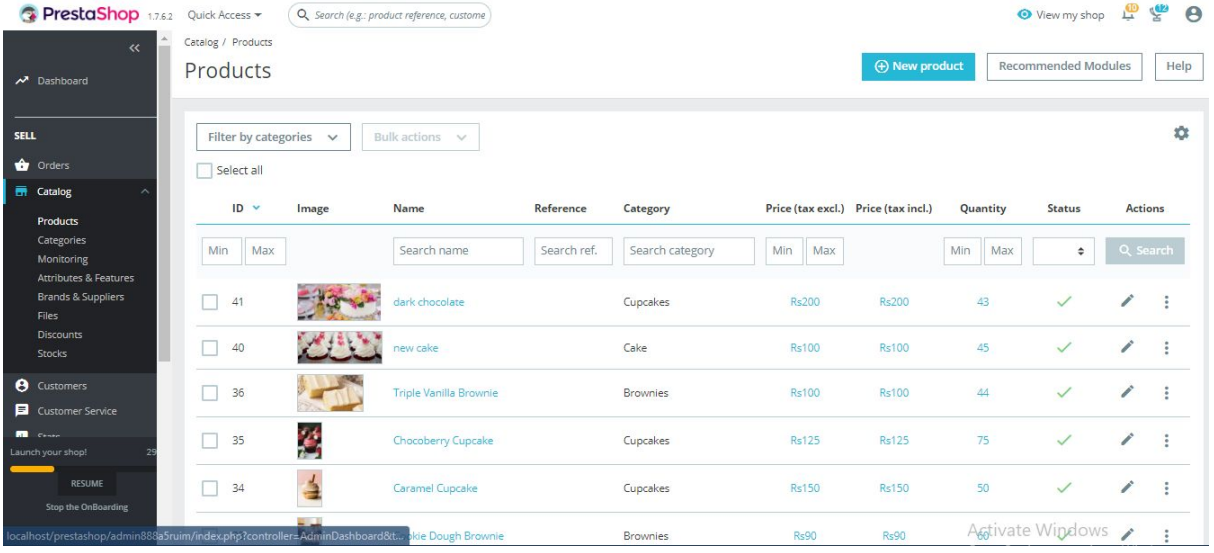
2) **HOW TO CHANGE PRODUCT DETAILS FROM ADMIN**

a) Open the admin interface

PROJECT DUDEPANDA

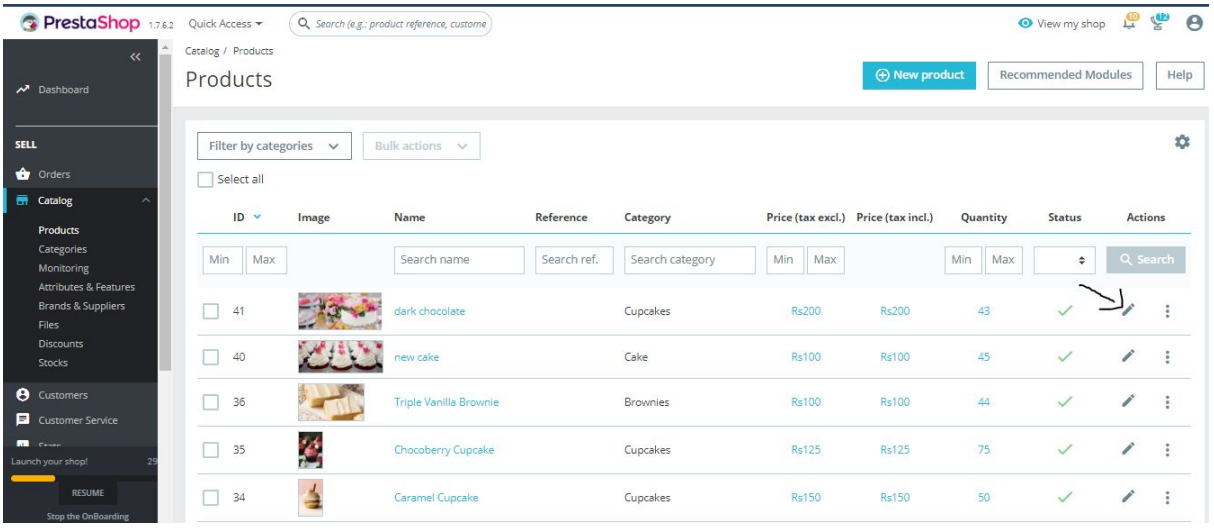


b) Click on catalog and then click on products



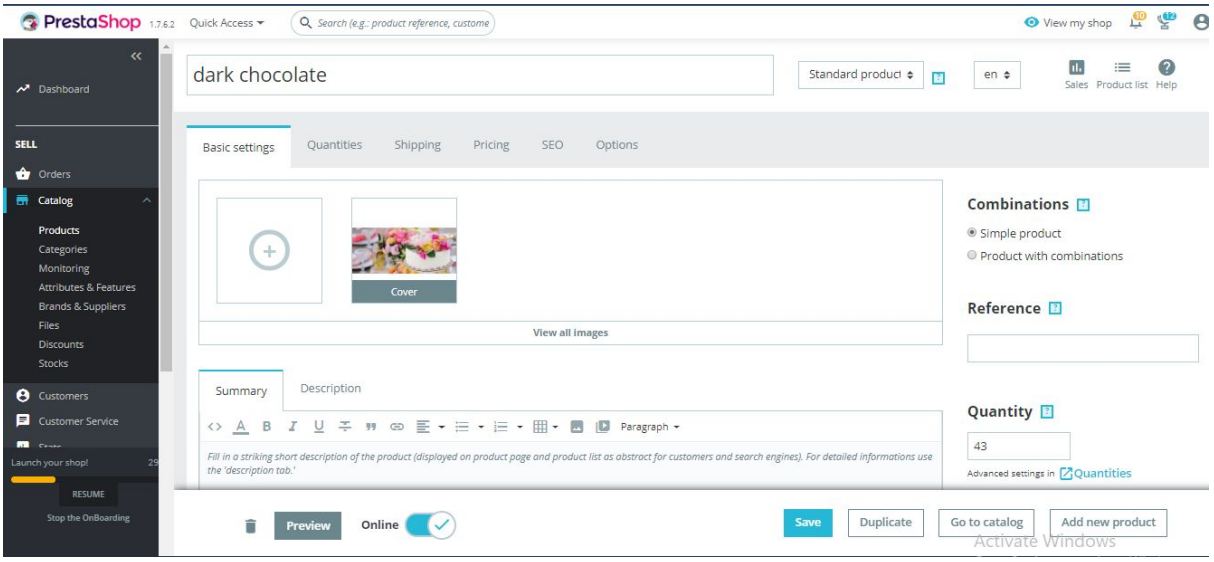
c)

I am going to change the price and stock of dark chocolate

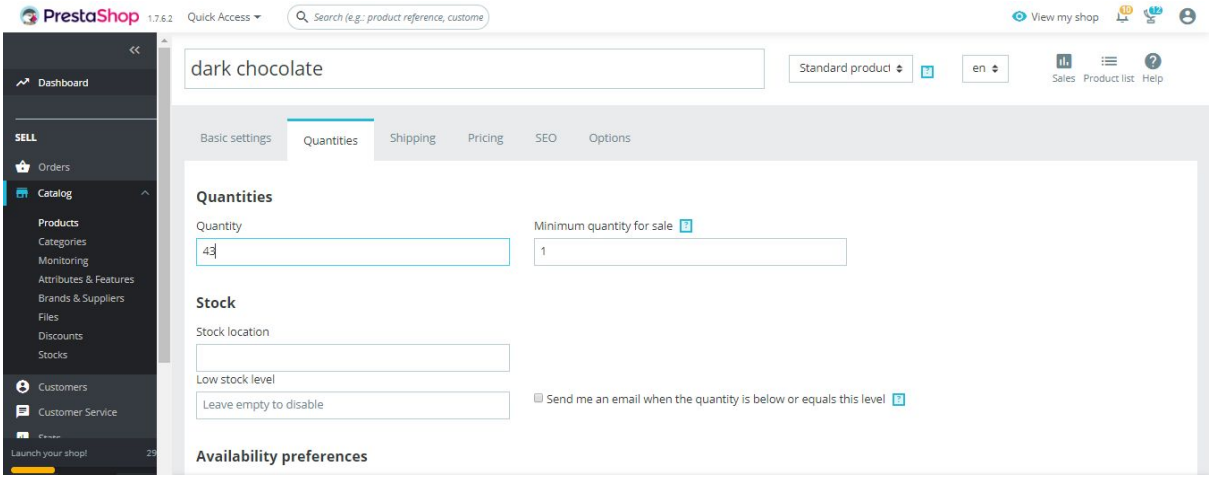


PROJECT DUDEPANDA

By clicking on the edit mark

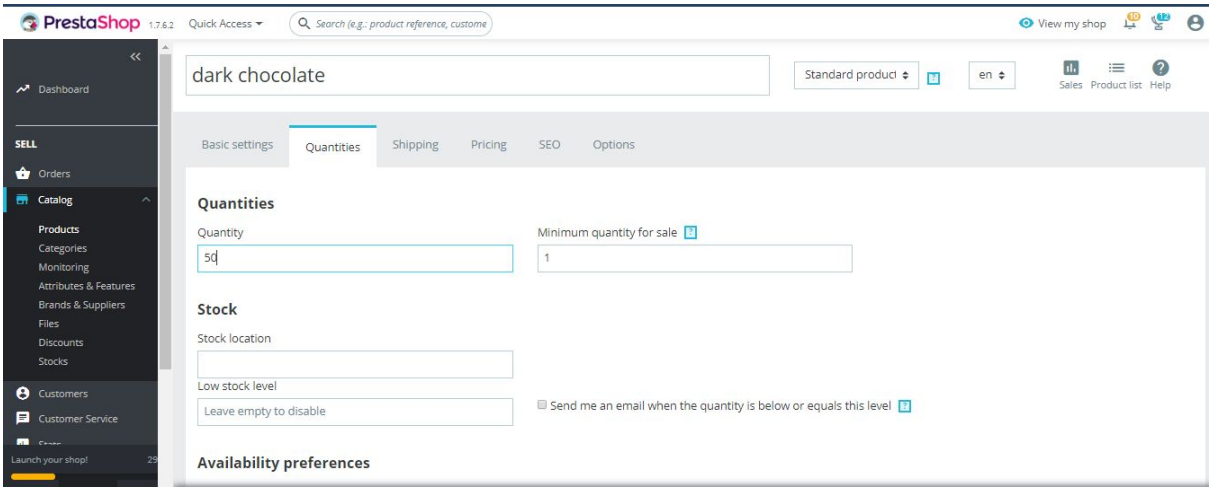


After the following window appears, move to *Quantities*

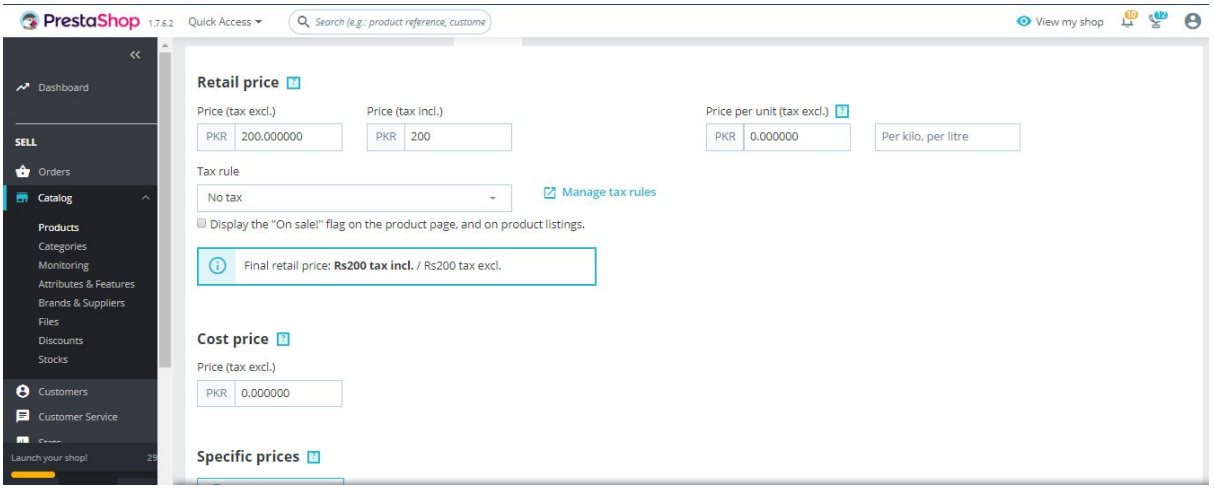


d)

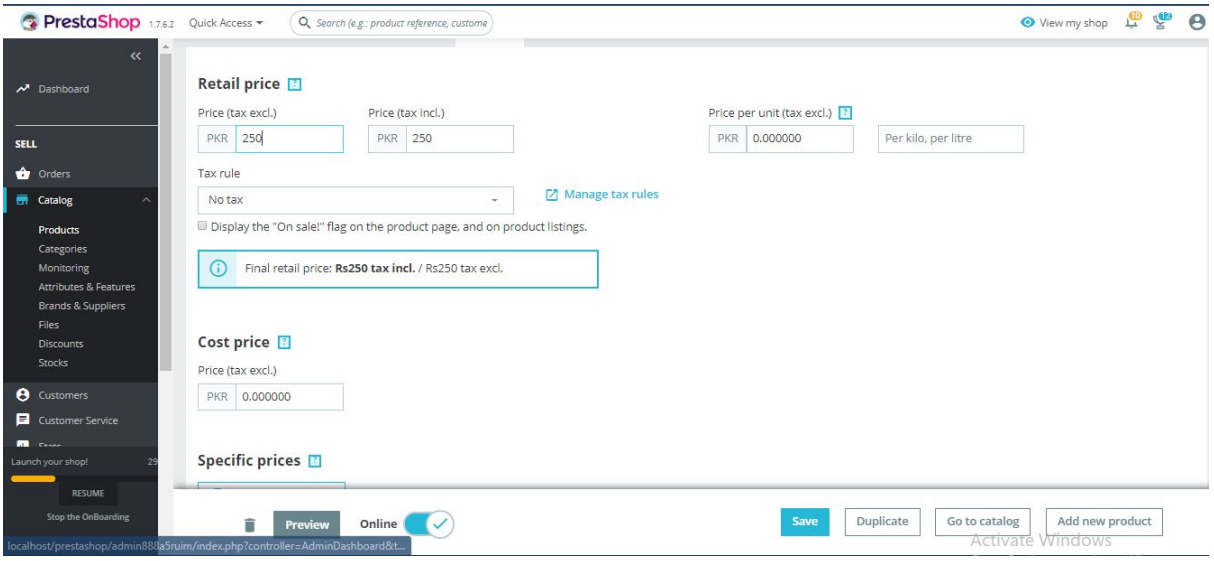
Change the quantity by entering a number. I am going to enter and 50 and then move onto prices



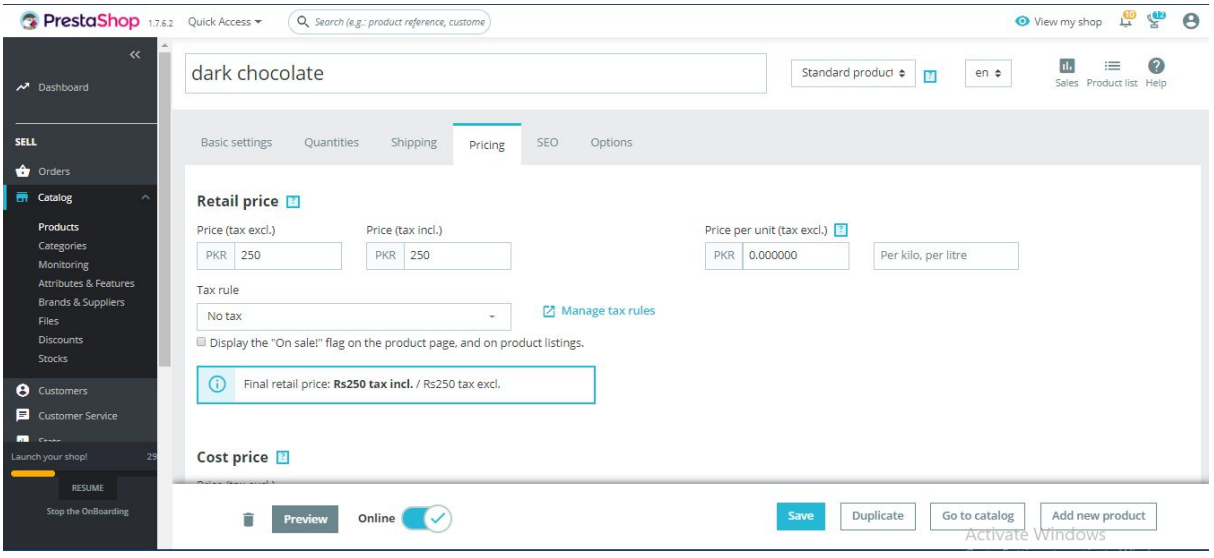
PROJECT DUDEPANDA



And change the price to 250



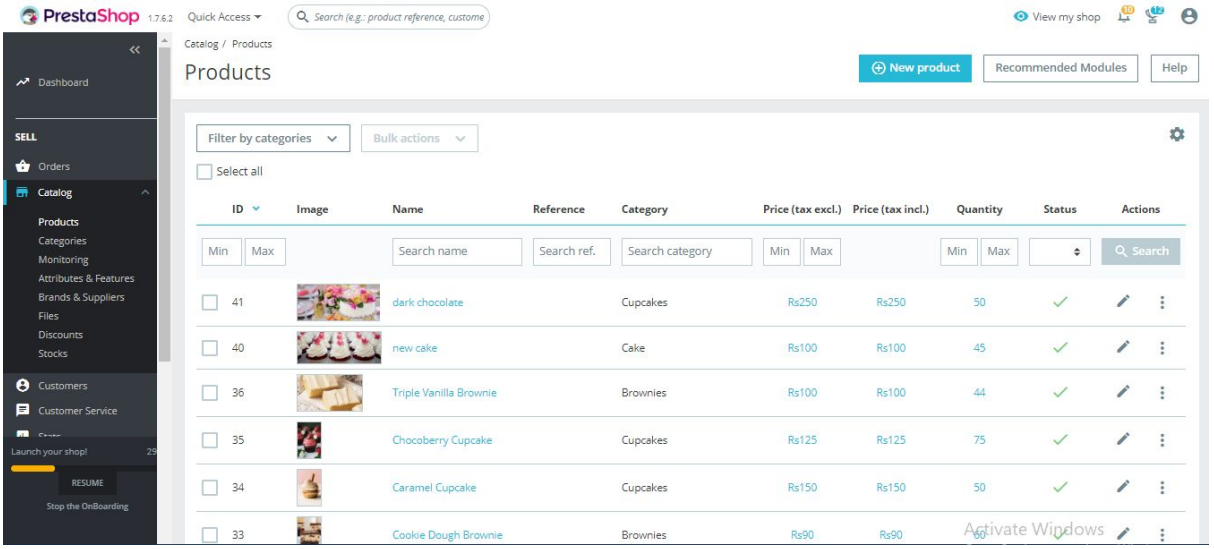
e)



Click save and make sure the page is online

PROJECT DUDEPANDA

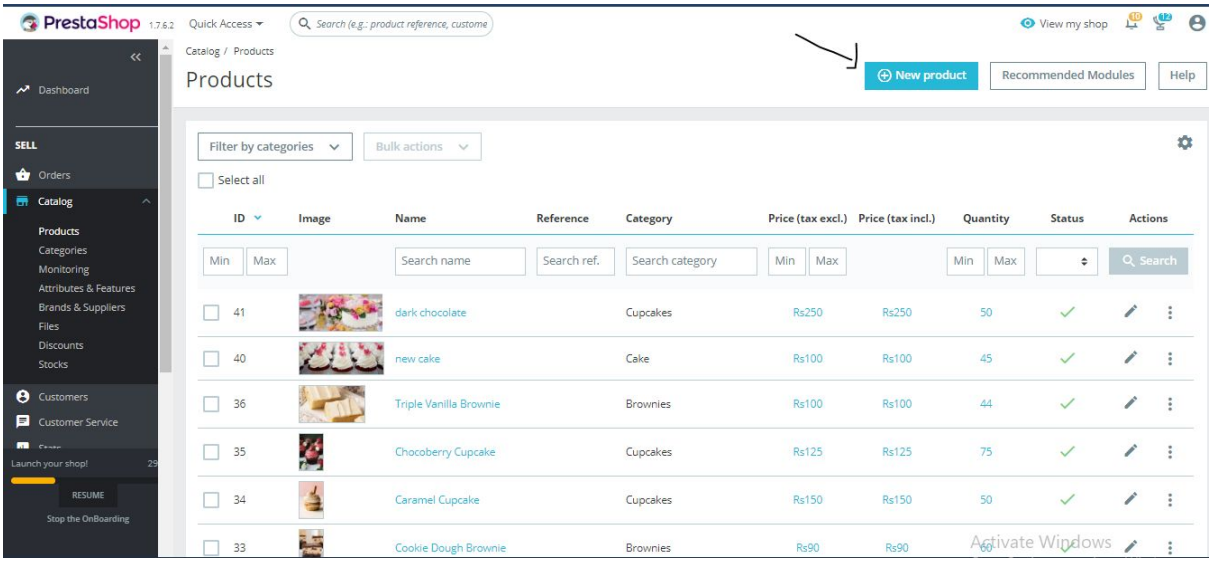
f)



Changes updated!!.

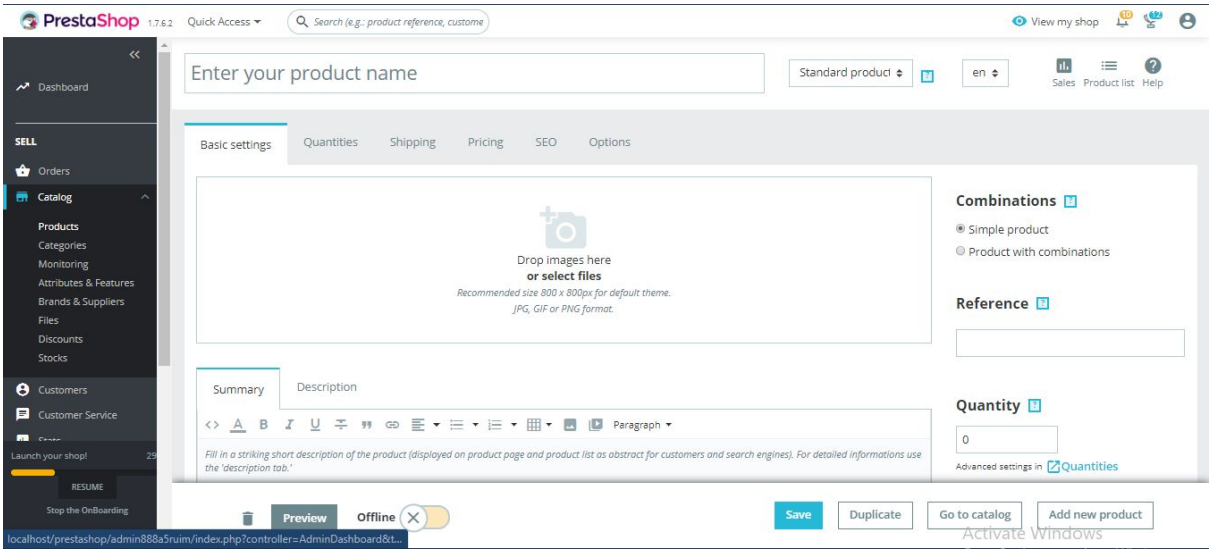
3) **HOW TO ADD A NEW ITEM INTO THE MENU**

1)

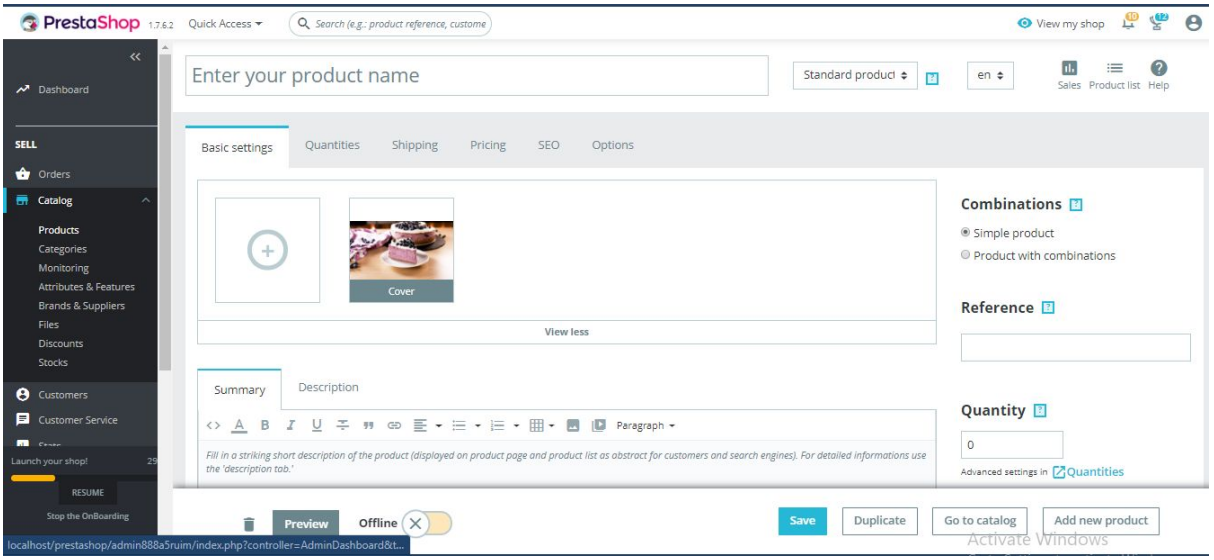


Click on new product

PROJECT DUDEPANDA

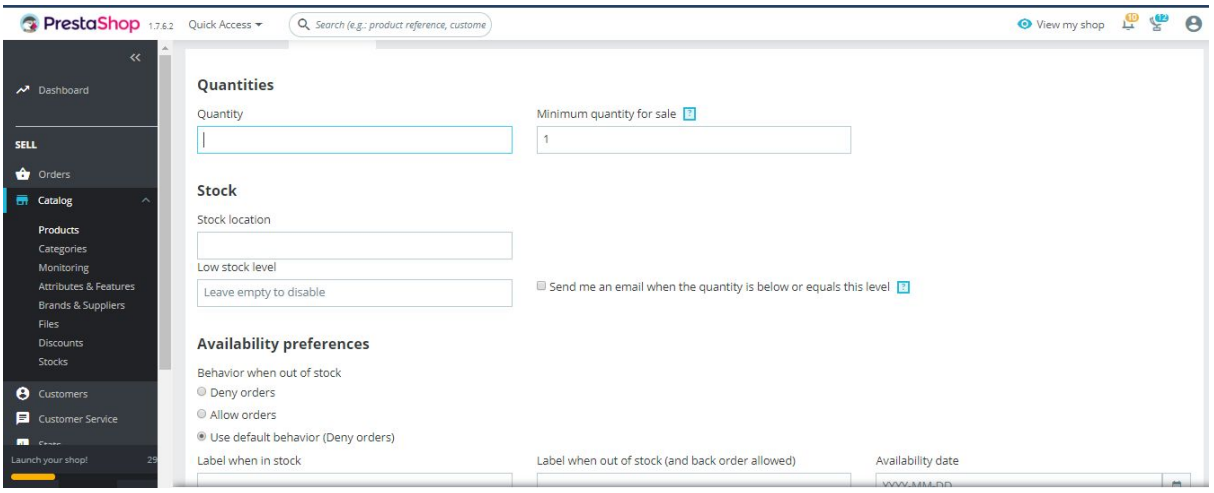
2) 

Upload an image of the product
Since i don't have one, i am just going to upload any random picture from google
I am going to upload a picture of blueberry cheese cake

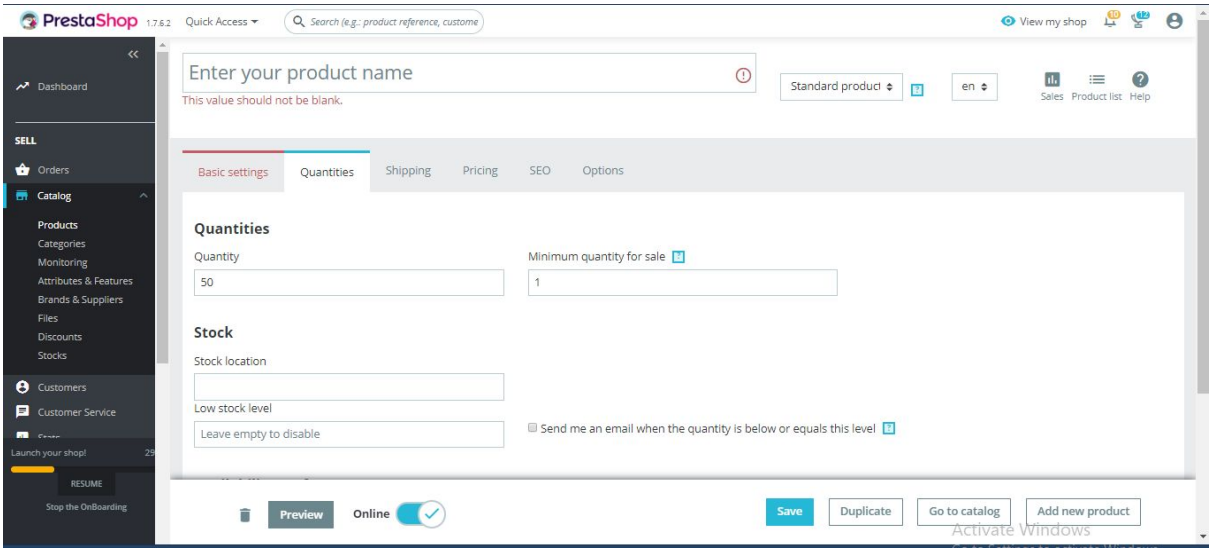


3) Then move to quantities

PROJECT DUDEPANDA

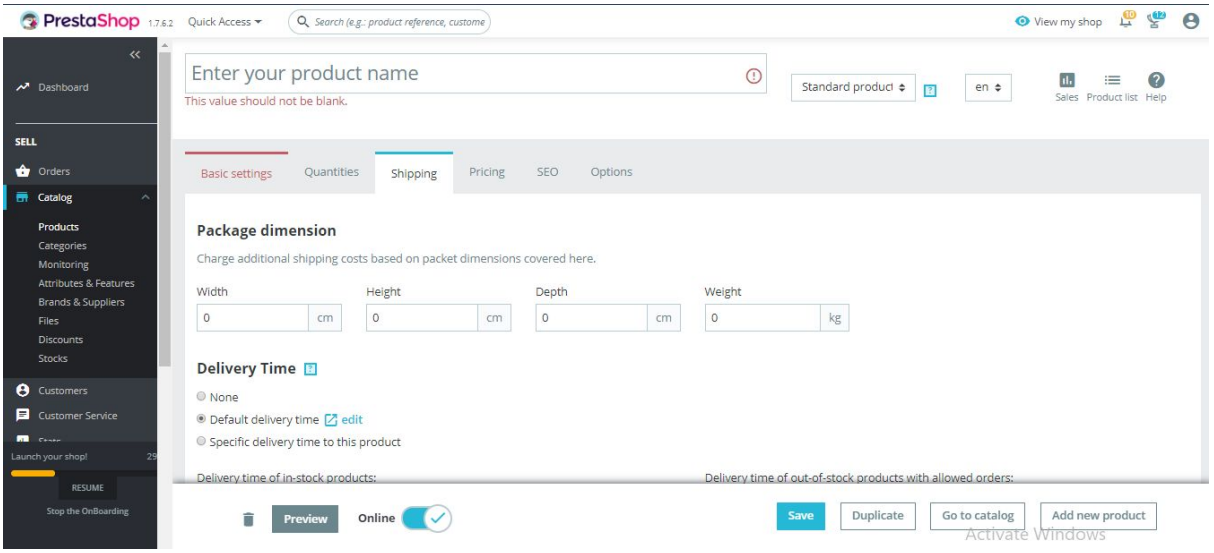


Enter the quantity number (e.g 50)



Make sure the online bar is on when you are doing your work

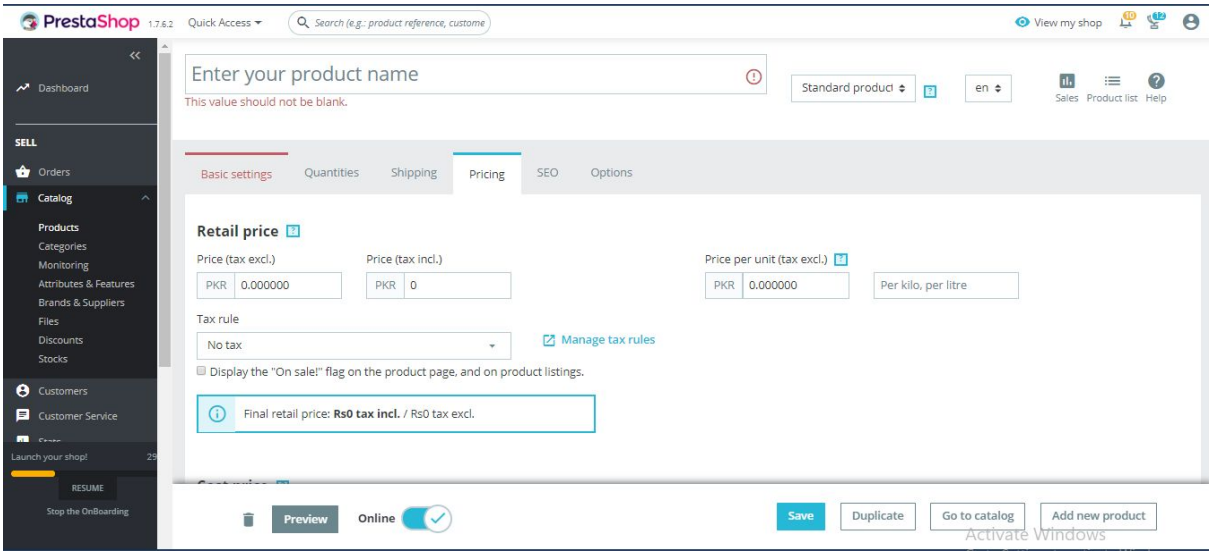
4) Now move to shipping



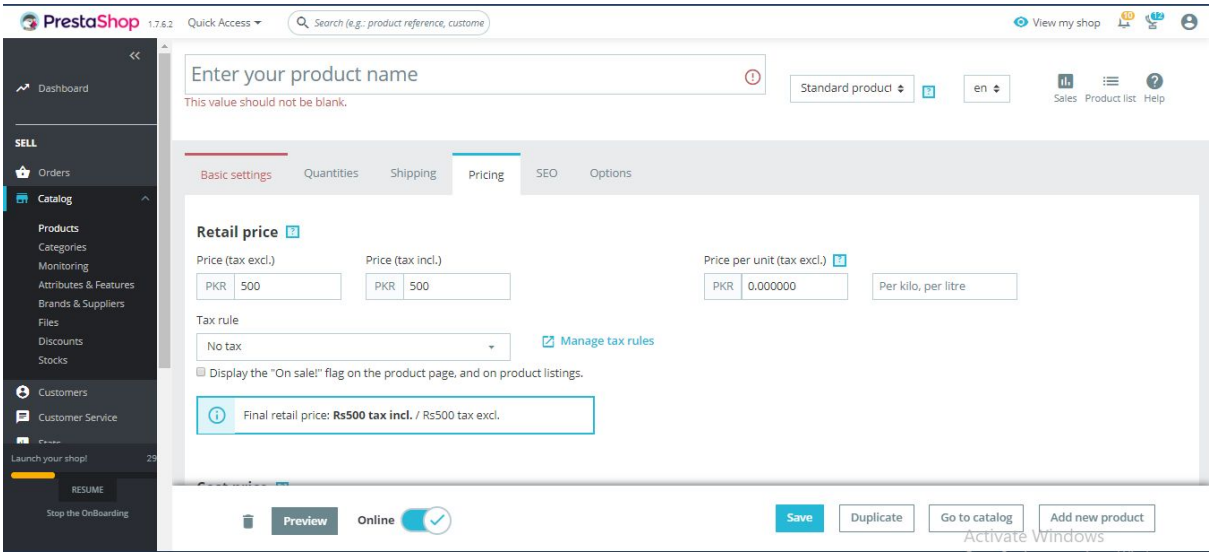
Enter any details if necessary

PROJECT DUDEPANDA

5) Now move to pricing

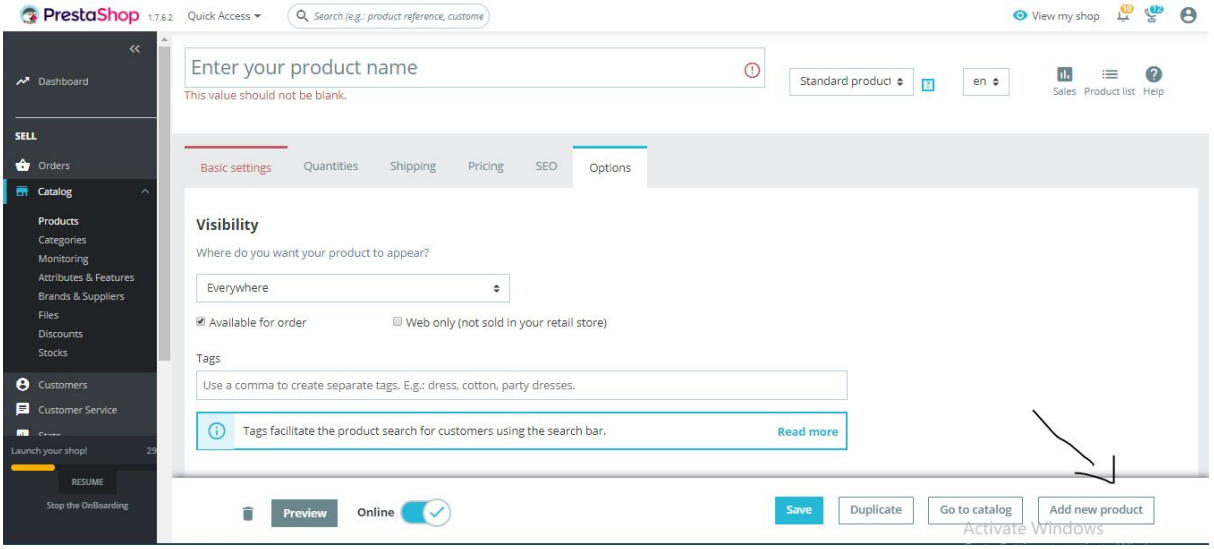


Enter price details
For e.g



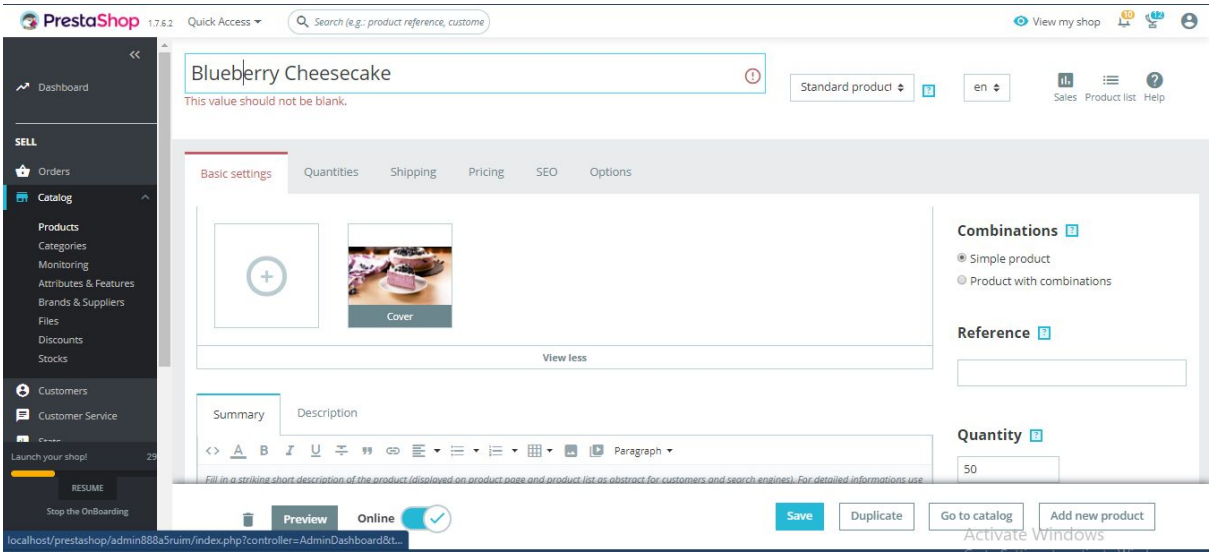
6) Now move to options

PROJECT DUDEPANDA



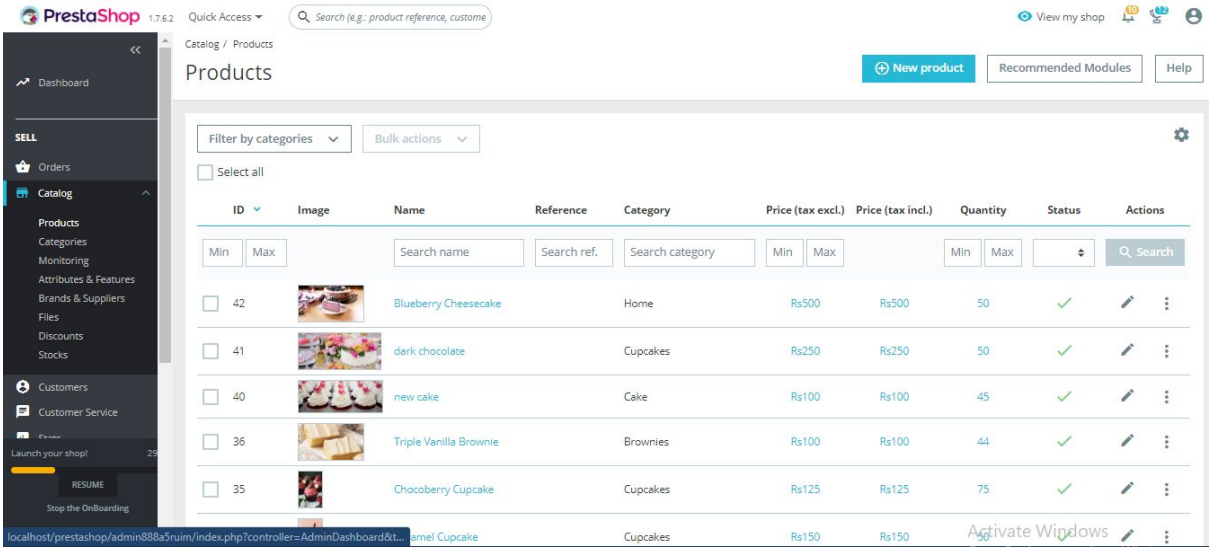
And click on *Add new Product*

Oops I forgot to enter the product name in the beginning



Click on add new product and then wait for changes to update
Then click on products under catalog and

PROJECT DUDEPANDA



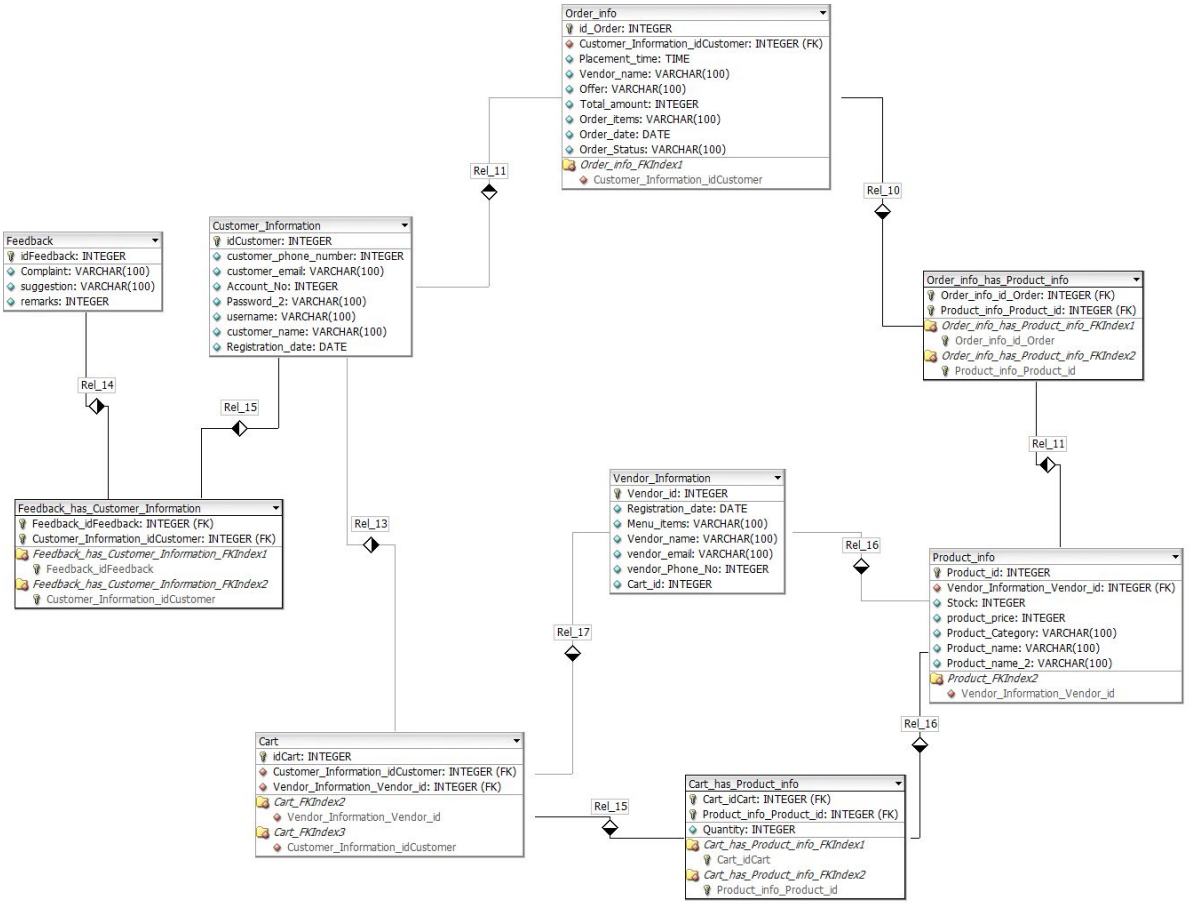
Blueberry cheesecake successfully added.

Functionalities of backend/ MYSQL administration tool are listed after the description of the ERD.

PROJECT DUDEPANDA

ERD;

Following is a picture of our ERD;



PROJECT DUDEPANDA

The entities in this ERD are **Vendor Information, Customer Information, Product_Info, Cart,Order_Info, Feedback**
Following is the description of the entities and the relationships between them:

-**Vendor Information**
This table will contain the list of Vendors registered with the Company. Each Vendor has certain attributes such as Vendor name, phone no#, Registration date etc. Each Vendor is assigned a Vendor ID by the Company
- **Customer Information**
Customer info will contain the details of the users that have signed up on the website whether or not they use the application or not. This table contains the users account details (account# , password, username) and other details such as phone number, email address and of course the name of the Customer. Each Customer is assigned a Customer ID by the company at the time when the Customer creates an account
-**Product Information**
Product Information will display the Menu of the Different Vendors. Each Product is assigned a Product ID by the Company
-Cart
Each Customer will have its own Cart (and hence a seperate ID assigned by the Company) which will hold information about the order.
-**Order Information**
This table will contain the information about the details of a particular order placed by a customer, including its placement time. Each order will have a distinct ID assigned by the Company.
-**Feedback**
Customers will be reviewed about their experiences regarding their usage and satisfaction. This table will contain all that data

PROJECT DUDEPANDA

Relationships:

These Relationships are with respect to what is the source of the relationship

-**Vendor Information**

- 1:n relationship with Product_info table since the vendor would have the information of the product he is giving to the company. Therefore, the vendor will have information for many products at any given time and many products could have the same vendor.
- 1:n relationship with Cart since 1 vendor information could be in many different carts.

-**Customer Information**

- 1:n relationship with Order_info table since a Customer could have many orders.
- M:n relationship with Feedback table since the customer may have many feedbacks and same feedback may also be given by many customers.
- 1:n relationship with Cart since 1 customer could have many carts in a day if he orders multiple times and many carts could be of the same customer as well.

-**Product Information**

- M:n relationship with Order_info table, 1 product could have many orders and similarly 1 orders could be of many different products.
- M:n relationship with Cart since many products could be in a single cart and many carts could have one product as well.

- **Order Information**

- M:n relationship with Product_info since one order could have many products and many products could be in one order.

-**Feedback**

- M:n relationship with Customer information since 1 customer could have many feedbacks and 1 feedback could be given by many customers as well.

-**Cart**.

- M:n relationship with Product_info since 1 cart could have many products in it and many carts could contain the same product.

PROJECT DUDEPANDA

Environment For Database:

The environment used for Database is Prestashop, which is an open source E-commerce solution, which can be used to run stores in the cloud or via self-hosting. It is currently used by 250,000 shops and is available in 65 languages. Prestashop is fairly easy to use and provides a powerfully responsive store interface for shoppers, offers a comprehensive set of features for free.

Setup for Prestashop:

- **What you need to get started:**

  Here is a list of things which are required before you proceed to download Prestashop.

  - System requirements:
    - PHP 5.4 or later.
      - Useful settings (in the `php.ini` file):
        - `allow_url_fopen` set to On,
        - `register_globals` set to Off,
        - `upload_max_filesize` set to "16M" (or more).
      - Must-have PHP extensions (in the `php.ini` file): PDO_MySQL, cURL, SimpleXML, mcrypt, GD, OpenSSL, DOM, SOAP, Zip, fileinfo.
      - Useful server tools: cron/crontab, Memcached.
    - MySQL 5.0 or later.
    - Better if:
      - Unix/Linux hosting.
      - Apache Web Server 2.0 or later or nginx Web Server.
        - Apache module settings:
          - `mod_rewrite` enabled,
          - `mod_security` disabled,
          - `mod_auth_basic` disabled.
      - At least 128 Mb of RAM dedicated to PHP. The more the better.
  - Access codes to your FTP server, your MySQL database
    - These should be provided by your web host if you are not doing a local installation.
  - Any text editor.
  - Any FTP client.
  - Any modern Web browser (if using Internet Explorer: at least IE9).

  (taken from http://doc.prestashop.com/display/PS17/What+you+need+to+get+started)

- **Prestashop Setup:**
  - Download zip file for Prestashop version 1.7.6.2, which is available on its website, i.e. https://www.prestashop.com/en/download.

PROJECT DUDEPANDA

- Use Xampp, which is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in PHP and Perl programming languages. After installing Xampp, we will start Apache and MySQL module. Now, our local host is available for use.
- Now, when you type http://localhost/phpmyadmin on your web browser, it will open your local host database. Create a new database with name, “prestashop” from the Database tab.
- Now, when you type http://localhost/prestashop on your web browser, it will start the installation setup for Prestashop.
- Follow the steps and input details as required.
- During the steps, when it asks about the database_host, database_name, database_user and database_password, type the details of the database you created on phpmyadmin.
- As soon as you complete the steps, it will give you the generated links for both, prestashop customer hand and prestashop admin panel.
- You can edit the shop according to your product through the admin panel, such as logo, products, themes, carousel, pictures etc.
- You can also manage the orders, shipping, sales and overall performance of the store through the admin panel.

SQL SCRIPT of ERD:

```
CREATE TABLE Customer_Information (
  idCustomer INTEGER  NOT NULL   AUTO_INCREMENT,
  customer_phone_number INTEGER UNSIGNED  NULL  ,
  customer_email VARCHAR(100)  NULL  ,
  Account_No INTEGER UNSIGNED  NULL  ,
  Password_2 VARCHAR(100)  NULL  ,
  username VARCHAR(100)  NULL  ,
  customer_name VARCHAR(100)  NULL  ,
  Registration_date DATE  NULL    ,
PRIMARY KEY(idCustomer));



CREATE TABLE Feedback (
  idFeedback INTEGER UNSIGNED  NOT NULL   AUTO_INCREMENT,
  Complaint VARCHAR(100)  NULL  ,
  suggestion VARCHAR(100)  NULL  ,
  remarks INTEGER UNSIGNED  NULL    ,
PRIMARY KEY(idFeedback));
```

PROJECT DUDEPANDA

```
CREATE TABLE Vendor_Information (
  Vendor_id INTEGER UNSIGNED  NOT NULL   AUTO_INCREMENT,
  Registration_date DATE  NULL  ,
  Menu_items VARCHAR(100)  NULL  ,
  Vendor_name VARCHAR(100)  NULL  ,
  vendor_email VARCHAR(100)  NULL  ,
  vendor_Phone_No INTEGER UNSIGNED  NULL  ,
  Cart_id INTEGER UNSIGNED  NULL    ,
PRIMARY KEY(Vendor_id));
```

```
CREATE TABLE Order_info (
  id_Order INTEGER UNSIGNED  NOT NULL   AUTO_INCREMENT,
  Customer_Information_idCustomer INTEGER  NOT NULL  ,
  Placement_time TIME  NULL  ,
  Vendor_name VARCHAR(100)  NULL  ,
  Offer VARCHAR(100)  NULL  ,
  Total_amount INTEGER UNSIGNED  NULL  ,
  Order_items VARCHAR(100)  NULL  ,
  Order_date DATE  NULL  ,
  Order_Status VARCHAR(100)  NULL    ,
PRIMARY KEY(id_Order)  ,
INDEX Order_info_FKIndex1(Customer_Information_idCustomer),
  FOREIGN KEY(Customer_Information_idCustomer)
    REFERENCES Customer_Information(idCustomer)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION);
```

```
CREATE TABLE Product_info (
  Product_id INTEGER UNSIGNED  NOT NULL  ,
  Vendor_Information_Vendor_id INTEGER UNSIGNED  NOT NULL  ,
  Stock INTEGER UNSIGNED  NULL  ,
  product_price INTEGER UNSIGNED  NULL  ,
  Product_Category VARCHAR(100)  NULL  ,
  Product_name VARCHAR(100)  NULL  ,
  Product_name_2 VARCHAR(100)  NULL    ,
PRIMARY KEY(Product_id)  ,
INDEX Product_FKIndex2(Vendor_Information_Vendor_id),
  FOREIGN KEY(Vendor_Information_Vendor_id)
    REFERENCES Vendor_Information(Vendor_id)
```

```
    ON DELETE NO ACTION
    ON UPDATE NO ACTION);



CREATE TABLE Order_info_has_Product_info (
  Order_info_id_Order INTEGER UNSIGNED  NOT NULL  ,
  Product_info_Product_id INTEGER UNSIGNED  NOT NULL    ,
PRIMARY KEY(Order_info_id_Order, Product_info_Product_id)  ,
INDEX Order_info_has_Product_info_FKIndex1(Order_info_id_Order)  ,
INDEX Order_info_has_Product_info_FKIndex2(Product_info_Product_id),
  FOREIGN KEY(Order_info_id_Order)
    REFERENCES Order_info(id_Order)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION,
  FOREIGN KEY(Product_info_Product_id)
    REFERENCES Product_info(Product_id)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION);



CREATE TABLE Cart (
  idCart INTEGER UNSIGNED  NOT NULL   AUTO_INCREMENT,
  Customer_Information_idCustomer INTEGER  NOT NULL  ,
  Vendor_Information_Vendor_id INTEGER UNSIGNED  NOT NULL    ,
PRIMARY KEY(idCart)  ,
INDEX Cart_FKIndex2(Vendor_Information_Vendor_id)  ,
INDEX Cart_FKIndex3(Customer_Information_idCustomer),
  FOREIGN KEY(Vendor_Information_Vendor_id)
    REFERENCES Vendor_Information(Vendor_id)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION,
  FOREIGN KEY(Customer_Information_idCustomer)
    REFERENCES Customer_Information(idCustomer)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION);



CREATE TABLE Cart_has_Product_info (
  Cart_idCart INTEGER UNSIGNED  NOT NULL  ,
  Product_info_Product_id INTEGER UNSIGNED  NOT NULL  ,
  Quantity INTEGER UNSIGNED  NULL    ,
PRIMARY KEY(Cart_idCart, Product_info_Product_id)  ,
```

PROJECT DUDEPANDA

```
INDEX Cart_has_Product_info_FKIndex1(Cart_idCart)  ,
INDEX Cart_has_Product_info_FKIndex2(Product_info_Product_id),
  FOREIGN KEY(Cart_idCart)
    REFERENCES Cart(idCart)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION,
  FOREIGN KEY(Product_info_Product_id)
    REFERENCES Product_info(Product_id)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION);



CREATE TABLE Feedback_has_Customer_Information (
  Feedback_idFeedback INTEGER UNSIGNED  NOT NULL  ,
  Customer_Information_idCustomer INTEGER  NOT NULL    ,
PRIMARY KEY(Feedback_idFeedback, Customer_Information_idCustomer)  ,
INDEX Feedback_has_Customer_Information_FKIndex1(Feedback_idFeedback)  ,
INDEX
Feedback_has_Customer_Information_FKIndex2(Customer_Information_idCustomer),
  FOREIGN KEY(Feedback_idFeedback)
    REFERENCES Feedback(idFeedback)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION,
  FOREIGN KEY(Customer_Information_idCustomer)
    REFERENCES Customer_Information(idCustomer)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION);
```

PROJECT DUDEPANDA