# Electronic Vendor Database Group 21

Hyunsuk Bang

George Tapia

Marcos Ferreira

# Contents

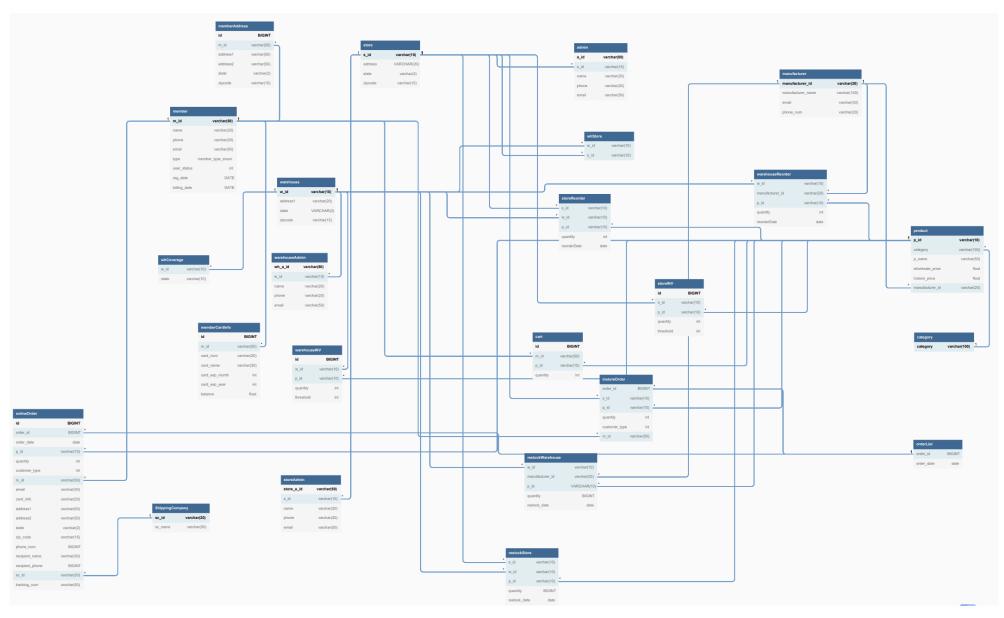
Introduction	3
ER – Diagram	4
Getting the application to Run	5
Signing Up	6
Adding a Credit Card:	7
Adding a Address:	8
Home Page	
Selected Product	
Shopping Cart	10
Checkout	11
Purchase History	12
Store and Warehouse Administrators	12
Member Contributions	15

## Introduction

The electronic vendor we built is similar to 'Best Buy'. It has the standard features such as creating an account, querying different products, modifying/viewing account information (including keeping track of credit card and address information), purchasing history, shopping cart and so on... Since our database simulates multiple stores connected to a common warehouse, we also have store and warehouse admins to manage the inventories.

All the data is stored in a mySQL database. Our application then fetches the appropriate information from the database and presents it to the customer (along with other interactions) with a easy to use user interface. To see more detail of the database, look at the ER diagram just below.

# ER – Diagram



# Getting the application to Run

- 1. Download the linked source files
- 2. Install Python (<a href="https://www.python.org/downloads/">https://www.python.org/downloads/</a>)
- 3. Install 'Django' (https://docs.djangoproject.com/en/4.0/topics/install/)
- 4. Create a mySQL database and run the provided SQL script on it
  - a. Make sure to run the create table statements with trigger statement **and then** the insert statements
- 5. Within the source file, navigate into '/CS425 FINAL PROJECT/mysite' and open 'settings.py'

a. Edit the username, password and DB name to match your mysql username, password

and DB name

- 6. Navigate back to '/CS425\_FINAL\_PROJECT' and run on the command prompt:
  - a. python manage.py makemigrations
  - b. python manage.py migrate
  - c. python manage.py createsuperuser
    - i. this is important so that you can access the DB admin page which in the URL is at '../admin'
  - d. python manage.py runserver
    - i. copy and paste the http address to access the website

**Note!** There maybe one or two libraries that are not installed on your machine, please read which library the error message says is missing and install it using 'pip' installer (<a href="https://pip.pypa.io/en/stable/installation/">https://pip.pypa.io/en/stable/installation/</a>)

# Signing Up

Here I will be showing the step-by-step process that a customer would do to shop on our website, starting with creating an account. However, creating an account is not necessary, as the user can also shop as a guest (which will be demonstrated later).

sign_up	sign_up
ID:	ID:
122	13
Password:	Password:
•••••	•••••
retype Password:	retype Password:
•••••	•••••
email:	email:
example122@gmail.com	example13@gmail.com
Name:	Name:
customer122	customer13
phone:	phone:
123456789	123654789
type Normal V	type Subscript 🗸
Sign Up	Sign Up
go back	g <u>o back</u>

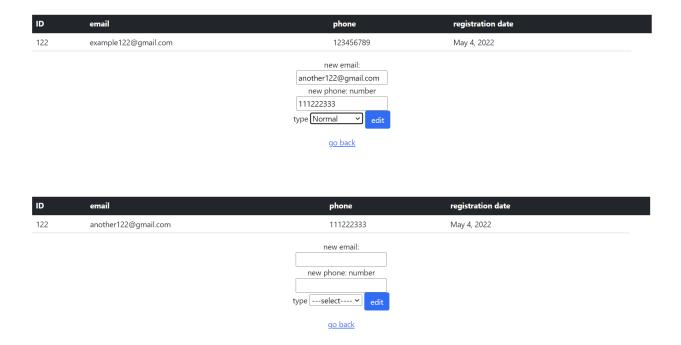
There are two types of users; normal users and subscript users. User's who subscribe to our website will have their accounts build monthly while normal users will be charged immediately for any purchases made.

# Log IN in with username or password is incorrect. or user id does not exist ID: password: password: login After creating in with greet the gr

go back

After creating an account, logging in with the wrong credentials will greet the user with this error message.

After successfully logging in, the user will be able to view and modify his/her account details as demonstrated below.



The user can also add different credit cards and addresses

## Adding a Credit Card:

# add new card



After adding two cards, the user will be able to view their balances and delete them if user want's to protect their privacy and not save their credit card information on our website.

	card number	name on card	card expiration	Balance	
1	2000312288719332	customer122	8 / 27	614576.2 USD delete	
2	9822303076652211	customercousin122	7 / 27	345163.2 USD delete	

new card go back

# Adding a Address:

# add new address

	street address:
JJohn	s Street
	Box, Apt Num:
2	
	State:
IL	
	zipcope:
60616	
	new address
	g <u>o back</u>

After adding three addresses, the user can view or delete them.

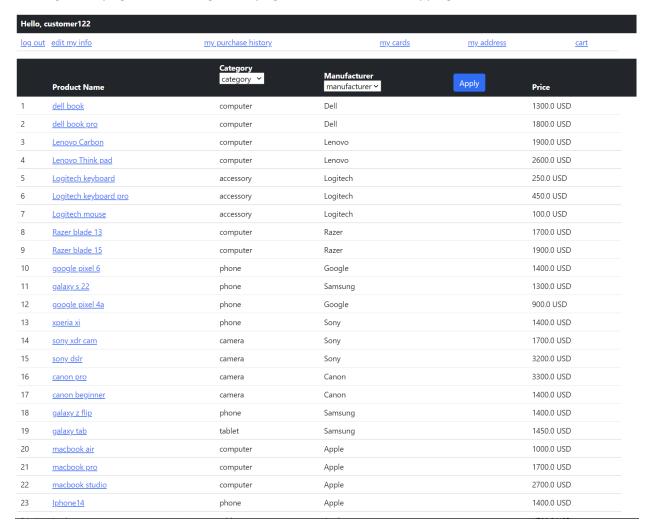
	address	state	zipcode	
1	JJohns Street 2	IL	60616	delete
2	Jupiter street 5	NY	101010	delete
3	Marsville street 2	NY	101010	delete

new address go back

After fully setting up an account, now we will explore the *home page*:

## Home Page

Here is the home page. Functionalities include links to, editing account info, seeing purchase history, viewing/modifying cards, viewing/modifying addresses, and the shopping cart.



We also see that we can query the products according to their category and manufacture. For example, if we just want to see computer from all manufactures:

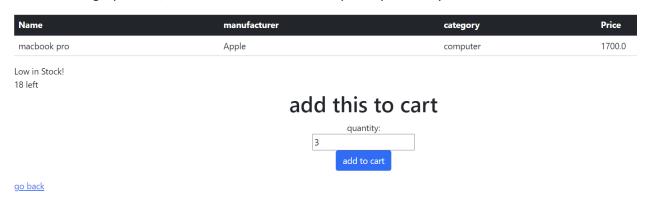
	Product Name	Category  Computer >	Manufacturer Manufacturer ✓ Apply	Price
1	dell book	computer	Dell	1300.0 USD
2	dell book pro	computer	Dell	1800.0 USD
3	Lenovo Carbon	computer	Lenovo	1900.0 USD
4	Lenovo Think pad	computer	Lenovo	2600.0 USD
5	Razer blade 13	computer	Razer	1700.0 USD
6	Razer blade 15	computer	Razer	1900.0 USD
7	macbook air	computer	Apple	1000.0 USD
8	macbook pro	computer	Apple	1700.0 USD
9	macbook studio	computer	Apple	2700.0 USD

Or instead, say we only want Apple computers:

	Product Name	Category computer ~	Manufacturer Apple  Apply	Price
1	macbook air	computer	Apple	1000.0 USD
2	macbook pro	computer	Apple	1700.0 USD
3	macbook studio	computer	Apple	2700.0 USD

# Selected Product

After selecting a product, the user can then select the quantity that they want to add to their cart.



# **Shopping Cart**

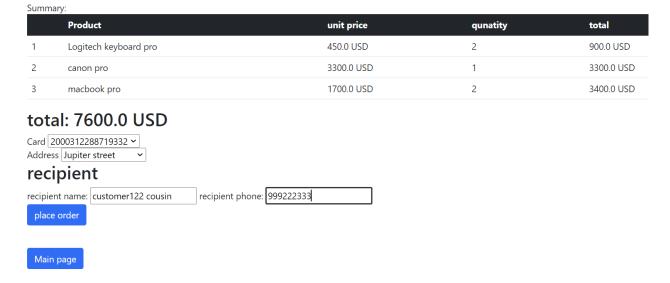
Once done shopping, the user can then view their shopping cart and either adjust the quantities or remove undesired products.

	Product	unit price	qunatity	total	update quantity	
1	Logitech keyboard pro	450.0 USD	1	450.0 USD	+ -	delete
2	canon pro	3300.0 USD	1	3300.0 USD		delete
3	macbook pro	1700.0 USD	3	5100.0 USD	+ -	delete
go ba	ack					proceed to check out

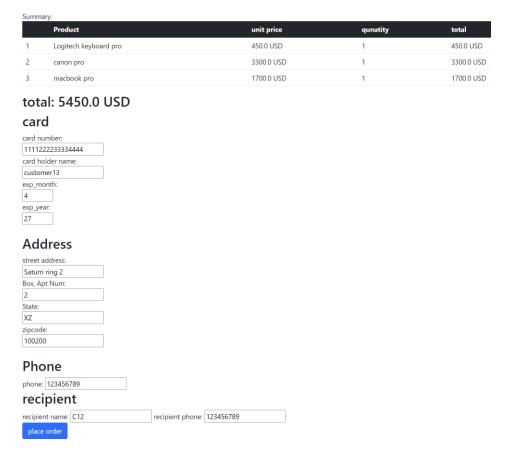
	Product	unit price	qunatity	total	update quantity	
1	Logitech keyboard pro	450.0 USD	2	900.0 USD	+ -	delete
2	canon pro	3300.0 USD	1	3300.0 USD	+ -	delete
3	macbook pro	1700.0 USD	2	3400.0 USD		delete
go b	<u>ack</u>					proceed to check out

## Checkout

For a logged in user, the user will choose the card and address out of the ones they saved on their account (of course they can always delete the cards and addresses later if they want their data privacy respected).



For a guest (user that's not logged in), the user will need to fill in the card and address details (but these details will not be permanently stored on the database).



After placing order, the user will be given their tracking number (for shipping).

# Thank you for your purchase!

your tracking number is 2022-05-04 20:30:51.458220+00:001

Main page

## **Purchase History**

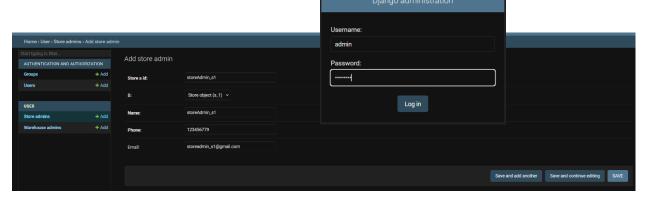
For users with accounts, they can then view their own purchase history:

	date	product	qunatity	card	address	phone
1	May 4, 2022	Logitech keyboard pro	2	2000312288719332	Jupiter street	111222333
2	May 4, 2022	canon pro	1	2000312288719332	Jupiter street	111222333
3	May 4, 2022	macbook pro	2	2000312288719332	Jupiter street	111222333

Main page

## Store and Warehouse Administrators

To manage the store and warehouse, we will need store and warehouse admins. Setting up these accounts is the responsibility of the database administrator. The DB admin will first register store and warehouse accounts with specific IDs.

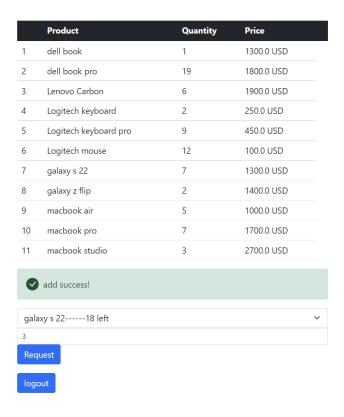


The respective store/warehouse admins will then *sign up* using the same ID (e.g in this example for a store admin, the ID is 'storeAdmin\_s1')

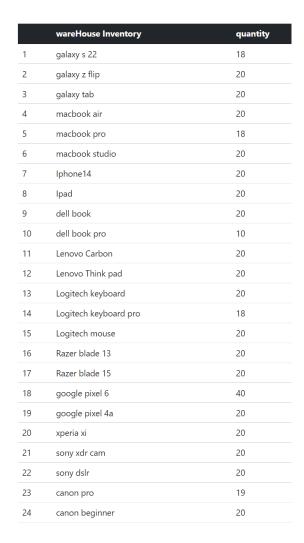
# sign\_up



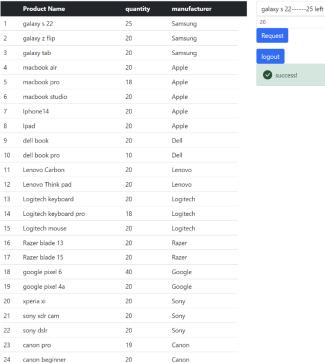
Upon logging in with a store admin account, the store admin will be show a page that will enable them to request inventory from the warehouse and a check will be done to ensure that don't request more than is available.







This is similar to the warehouse admin, except the warehouse admin order inventory from the manufacturers. Since we don't know how much inventory the manufacturers have, we assumed there to be no cap on the orders.





We can also see the record of the requested orders from both the store and warehouse admins on the database with the date stamps:



w_id		manufacturer	_id	p_id		quantity	restock_date	е
w_1	$\rightarrow$	man_01	$\rightarrow$	p_3	$\rightarrow$	1	2022-05-03	\$
w_1	$\rightarrow$	man_01	$\rightarrow$	p_3	$\rightarrow$	87	2022-05-03	0
w_1	$\rightarrow$	man_01	$\rightarrow$	p_3	$\rightarrow$	98	2022-05-03	0
w_1	$\rightarrow$	man_07	$\rightarrow$	p_19	$\rightarrow$	100000	2022-05-04	0

# Member Contributions

Although we each have areas we individually focused more on, we all contributed to some degree to all areas of the project.

Hyunsuk Bang – UI design, Application Development, Database Development

George Tapia – ER diagram, Database Development

Marcos Ferreira – Application Development, Documentation, Application Testing