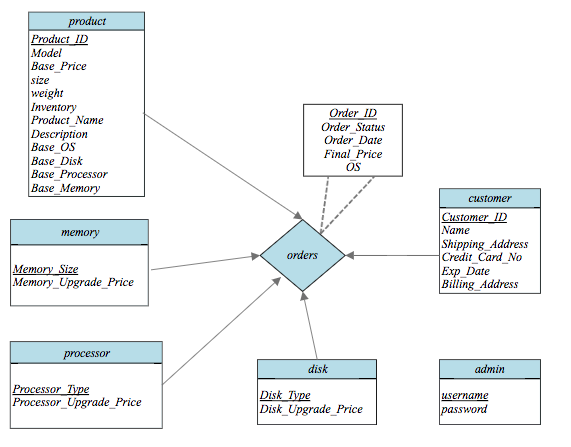
**Final Project Report**

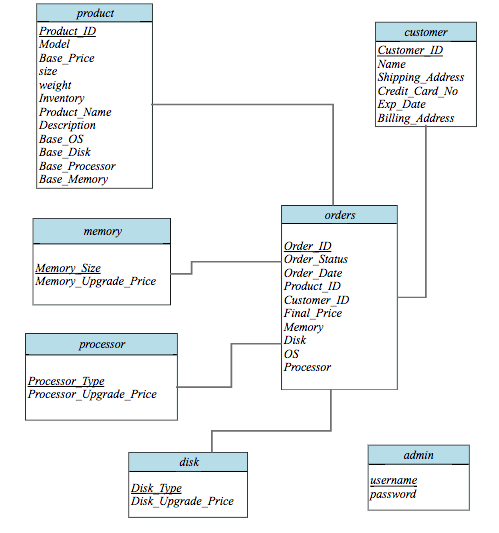
**Distribution of Work**

|  |  |
| --- | --- |
| **Student** | **Core Tasks Completed** |
| Zhong Huang | 1. Relational Model and ER Model 2. homepage.php 3. details.php 4. search.php 5. addorder.php 6. order.php 7. Set up AWS server and FTP server |
| Logan Rohde | 1. Relational Model and ER Model 2. checkout.php 3. Database set up 4. Initial Database data 5. Final Report draft |
| Na Wen | 1. Relational Model and ER Model 2. admin.php 3. index.php 4. signup.php 5. homepage.php 6. Website appeal |

**Entity Relationship Model**

****

**Relational Model**

****

**Relational Definitions and Functional Dependencies**

Product = {Product\_ID, Model, Base\_Price, size, weight, Inventory, Product\_Name, Discription, Base\_OS, Base\_Disk, Base\_Processor, Base\_Memory}

{Product\_ID} → {Model, Base\_Price, size, weight, Inventory, Product\_Name, Discription, Base\_OS, Base\_Disk, Base\_Processor, Base\_Memory}

memory = {Memory\_Size, Memory\_Upgrade\_Price}

{Memory\_Size} → {Memory\_Upgrade\_Price}

processor = {Processor\_Type, Processor\_Upgrade\_Price}

{Processor\_Type} → {Processor\_Upgrade\_Price}

disk = {Disk­\_Type, Disk\_Upgrade\_Price}

{Disk­\_Type} → {Disk\_Upgrade\_Price}

customer = {Customer­\_ID, Name, Shipping\_Address, Credit\_Card\_No, Exp\_Date, Billing\_Address}

{Customer­\_ID, Name} → {Shipping\_Address, Credit\_Card\_No, Exp\_Date, Billing\_Address}

orders = {Order\_ID, Order\_Status, Order\_Date, Product\_ID, Customer\_ID, Final\_Price, Memory, Disk, OS, Processor}

{Order\_ID**}** → {Order\_Status, Order\_Date, Product\_ID, Customer\_ID, Final\_Price, Memory, Disk, OS, Processor}

admin = {username, password}

{username} → {password}

**Part of Database Structure and Data**

mysql> show tables;

+------------------------+

| Tables\_in\_StoreProject |

+------------------------+

| admin                  |

| customer               |

| disk                   |

| memory                 |

| orders                 |

| processor              |

| product                |

| test                   |

+------------------------+

8 rows in set (0.00 sec)

mysql> DESCRIBE product;

+----------------+--------------+------+-----+---------+----------------+

| Field          | Type         | Null | Key | Default | Extra          |

+----------------+--------------+------+-----+---------+----------------+

| Product\_ID     | int(11)      | NO   | PRI | NULL    | auto\_increment |

| Model          | varchar(30)  | YES  |     | NULL    |                |

| Base\_Price     | decimal(6,2) | YES  |     | NULL    |                |

| size           | int(30)      | YES  |     | NULL    |                |

| weight         | int(30)      | YES  |     | NULL    |                |

| Inventory      | int(11)      | NO   |     | NULL    |                |

| Product\_Name   | varchar(30)  | NO   |     | NULL    |                |

| Description    | varchar(100) | YES  |     | NULL    |                |

| Base\_OS        | varchar(30)  | NO   |     | NULL    |                |

| Base\_Disk      | varchar(30)  | NO   |     | NULL    |                |

| Base\_Processor | varchar(30)  | NO   |     | NULL    |                |

| Base\_Memory    | varchar(30)  | NO   |     | NULL    |                |

+----------------+--------------+------+-----+---------+----------------+

12 rows in set (0.00 sec)

mysql> DESCRIBE orders;

+--------------+---------------+------+-----+---------+----------------+

| Field        | Type          | Null | Key | Default | Extra          |

+--------------+---------------+------+-----+---------+----------------+

| Order\_ID     | int(11)       | NO   | PRI | NULL    | auto\_increment |

| Order\_Status | varchar(30)   | YES  |     | NULL    |                |

| Order\_Date   | date          | YES  |     | NULL    |                |

| Product\_ID   | int(11)       | NO   | MUL | NULL    |                |

| Customer\_ID  | int(11)       | NO   | MUL | NULL    |                |

| Final\_Price  | decimal(10,2) | YES  |     | NULL    |                |

| Memory       | varchar(30)   | YES  |     | NULL    |                |

| Disk         | varchar(30)   | YES  |     | NULL    |                |

| OS           | varchar(30)   | YES  |     | NULL    |                |

| Processor    | varchar(30)   | YES  |     | NULL    |                |

+--------------+---------------+------+-----+---------+----------------+

10 rows in set (0.00 sec)

mysql> DESCRIBE customer;

+------------------+-------------+------+-----+---------+----------------+

| Field            | Type        | Null | Key | Default | Extra          |

+------------------+-------------+------+-----+---------+----------------+

| Customer\_ID      | int(11)     | NO   | PRI | NULL    | auto\_increment |

| Name             | varchar(30) | YES  |     | NULL    |                |

| Shipping\_Address | varchar(50) | YES  |     | NULL    |                |

| Credit\_Card\_No   | varchar(16) | NO   |     | NULL    |                |

| Exp\_Date         | varchar(7)  | YES  |     | NULL    |                |

| Billing\_Address  | varchar(50) | YES  |     | NULL    |                |

+------------------+-------------+------+-----+---------+----------------+

6 rows in set (0.00 sec)

mysql> select Model, Base\_Price, size, weight, Inventory, Product\_Name, Base\_OS, Base\_Disk, Base\_Processor, Base\_Memory from product;

+--------+------------+------+--------+-----------+----------------+------------------+------------+----------------+-------------+

| Model  | Base\_Price | size | weight | Inventory | Product\_Name   | Base\_OS          | Base\_Disk  | Base\_Processor | Base\_Memory |

+------------+--------+------------+------+--------+-----------+----------------+------------------+------------+-----------------+

| Laptop |     698.00 |   14 |      8 |         7 | HP Elite Book  | Windows 10 Basic | 250 GB HHD | Intel Core i7  | 8 GB        |

| Laptop |     599.99 |   10 |      7 |        16 | HP Pro Book    | Windows 10 Basic | 250 GB HHD | Intel Core i7  | 8 GB        |

| Hybrid |    1199.99 |   20 |     15 |         5 | HP Phantom 360 | Windows 10 Basic | 250 GB HHD | Intel Core i7  | 8 GB        |

| Hybrid |     799.99 |   12 |      8 |         5 | HP Wraith 360  | Windows 10 Basic | 250 GB HHD | Intel Core i7  | 8 GB        |

| Hybrid |     899.99 |   18 |     10 |         4 | HP Spectre 360 | Windows 10 Basic | 250 GB HHD | Intel Core i7  | 8 GB        |

| Tablet |     149.99 |    7 |      3 |         0 | HP Illusion    | Ubuntu           | 250 GB HHD | Intel Core i7  | 8 GB        |

| Tablet |     199.99 |    8 |      4 |         5 | HP Revenant    | Ubuntu           | 250 GB HHD | Intel Core i7  | 8 GB        |

| Tablet |     199.99 |    8 |      4 |         5 | HP Revenant    | Ubuntu           | 250 GB HHD | Intel Core i7  | 8 GB        |

| Tablet |     299.99 |   11 |      5 |         5 | HP Shade       | Ubuntu           | 250 GB HHD | Intel Core i7  | 8 GB        |

| Laptop |       4.00 |   16 |     10 |         5 | Lenovo Yoga    | Windows 10 Basic | 250 GB HHD | Intel Core i7  | 8 GB        |

+--------+------------+------+--------+-----------+----------------+------------------+------------+----------------+-------------+

11 rows in set (0.00 sec)

mysql> select \* from orders;

+----------+--------------+------------+------------+-------------+-------------+--------+------------+------------------+---------------+

| Order\_ID | Order\_Status | Order\_Date | Product\_ID | Customer\_ID | Final\_Price | Memory | Disk       | OS               | Processor     |

+----------+--------------+------------+------------+-------------+-------------+--------+------------+------------------+---------------+

|      557 | Process      | 2017-02-10 |        115 |           1 |      799.99 | 8 GB   | 250 GB HHD | Windows          | Intel Core i7 |

|      558 | Process      | 2017-03-14 |        119 |           1 |      199.99 | 8 GB   | 250 GB HHD | Windows          | Intel Core i7 |

|      559 | Shipped      | 2017-01-01 |        112 |          38 |      500.00 | 8 GB   | 250 GB HHD | Windows          | Intel Core i7 |

|      567 | Process      | 2017-04-22 |        112 |          38 |      599.99 | 8 GB   | 250 GB HHD | Windows 10       | Intel Core i7 |

|      568 | Process      | 2017-04-22 |        112 |          38 |      599.99 | 8 GB   | 250 GB HHD | Windows 10       | Intel Core i7 |

|      569 | Process      | 2017-04-22 |        112 |          38 |      599.99 | 8 GB   | 250 GB HHD | Windows 10       | Intel Core i7 |

|      570 | Process      | 2017-04-22 |        112 |          38 |      599.99 | 8 GB   | 250 GB HHD | Windows 10       | Intel Core i7 |

|      571 | Process      | 2017-04-22 |        115 |           1 |      899.99 | 8 GB   | 250 GB HHD | Windows 10       | Intel Core i7 |

|      572 | Process      | 2017-04-22 |        111 |           1 |      699.99 | 8 GB   | 250 GB HHD | Windows 10       | Intel Core i7 |

|      573 | Shipped      | 2017-04-22 |        111 |          41 |      699.99 | 8 GB   | 250 GB HHD | Windows 10       | Intel Core i7 |

|      574 | Process      | 2017-04-22 |        115 |           1 |      899.99 | 8 GB   | 250 GB HHD | Windows 10 Basic | Intel Core i5 |

|      575 | Process      | 2017-04-23 |        111 |           1 |      699.99 | 8 GB   | 250 GB HHD | Windows 10 Basic | Intel Core i7 |

|      577 | Check Out    | 2017-04-23 |        113 |          42 |     1199.99 | 16 GB  | 500 GB HHD | Windows 10 Basic | Intel Core i7 |

|      578 | Shipped      | 2017-04-24 |        111 |          38 |      699.99 | 8 GB   | 250 GB HHD | Windows 10 Basic | Intel Core i7 |

|      579 | Shipped      | 2017-04-24 |        111 |          38 |      699.99 | 8 GB   | 250 GB HHD | Windows 10 Basic | Intel Core i7 |

|      580 | Process      | 2017-04-24 |        112 |          38 |      599.99 | 8 GB   | 250 GB HHD | Windows 10 Basic | Intel Core i7 |

|      581 | Wishlist     | 2017-04-24 |        116 |          38 |      149.99 | 8 GB   | 1 TB HHD   | Ubuntu           | Intel Core i7 |

|      582 | Shipped      | 2017-04-24 |        111 |          38 |      698.00 | 16 GB  | 1 TB HHD   | Windows10 Pro    | Intel i7      |

|      583 | Process      | 2017-04-24 |        112 |           1 |      599.99 | 8 GB   | 1 TB HHD   | Windows 10 Basic | Intel Core i5 |

|      584 | Process      | 2017-04-24 |        112 |          38 |      599.99 | 16 GB  | 1 TB HHD   | Windows 10 Basic | Intel Core i7 |

|      585 | Process      | 2017-04-24 |        112 |          38 |      599.99 | 16 GB  | 1 TB HHD   | Windows 10 Basic | Intel Core i7 |

|      586 | Process      | 2017-04-24 |        112 |          38 |      599.99 | 16 GB  | 1 TB HHD   | Windows 10 Basic | Intel Core i7 |

|      587 | Process      | 2017-04-24 |        112 |          38 |      999.99 | 8 GB   | 250 GB HHD | Windows10 Pro    | Intel Core i7 |

|      588 | Process      | 2017-04-24 |        112 |          38 |      599.99 | 8 GB   | 250 GB HHD | Windows10 Pro    | Intel Core i7 |

|      589 | Process      | 2017-04-24 |        111 |          41 |      698.00 | 8 GB   | 250 GB HHD | Windows 10 Basic | Intel Core i7 |

|      593 | Cart         | 2017-04-25 |        111 |          38 |     2056.00 | 16 GB  | 1 TB HHD   | Windows10 Pro    | Intel i7      |

|      594 | Cart         | 2017-04-25 |        111 |           1 |      698.00 | 8 GB   | 250 GB HHD | Windows 10 Basic | Intel Core i7 |

+----------+--------------+------------+------------+-------------+-------------+--------+------------+------------------+---------------+

**Customers’ Queries and Transactions**

*Q1. Search. The results of a search should be a listing of systems with all relevant information such as configuration details and price.*

1. *Search by price (less than or equal to a price specified by a user).*
2. *Search by weight (less than or equal to a weight specified by by a user).*
3. *Size*
4. *Type*

On the left-hand side of the home page, there are input boxes. One for “Search by Price”, one for “Search by Weight”, and one for “Search by Size”. These items are then sent to search.php and input into the following query:

$sql = "

SELECT \*

FROM product

WHERE " . $attr. "< " . $value;

This query is adaptable to whatever search the customer does, whether it is by weight, price, or size.

In order to search by type, you can click on the “Product” drop down list that contains the types of devices provided, laptop, tablet, and hybrid. They type is also sent to search.php and input to the following query:

$sql = "

SELECT \*

FROM product

WHERE " . $attr. "= " . $value;

*Q2. System configuration: The result of search is a list of base systems, including processor, operating system, base memory, base disk capacity, weight, and size. Users must be able to easily specify larger memory and larger disk capacity, as well as a solid state drive. Your system should provide updated price for the desired configuration.*

Once the customer clicks on the details button of an item on the homepage, it takes them to the details.php page for that product. The base configuration, set up by the admin, is displayed initially. An initial query is used to extract all this information from the database and then the information extracted is displayed on the page in the dropdown menus using HTML.

$sql = "

SELECT \*

FROM product

WHERE Product\_ID = '" . $productId . "'";

Here on this page, the user can select different configurations for their device. They can change the processor, operating system, base memory, base disk capacity.

As the user makes changes to the configuration, the price is updated on the screen. The user can do three things with this product: add to cart, add to wishlist, or buy today.

*Q3. Purchase. Customer selects a configuration to purchase. The system asks for information (customer name address, credit-card number, etc…) and records the purchase.*

A customer has multiple pages to check out from. They can checkout from their wishlist, their cart, or their buy now. From these pages, the customer clicks the “Check Out”. This takes them to the checkout.php page which lists the customer’s previously entered information and the total price for their order total. This is displayed by the following:

$sql1 = "

SELECT \*

FROM customer

WHERE Name = '$customerName'";

$sql2 = "

SELECT SUM(o.Final\_Price) AS Total\_Price

FROM orders o, product p, customer c

WHERE c.Name = '$customerName'

AND Order\_Status = 'Check Out'

AND o.Product\_ID = p.Product\_ID

AND c.Customer\_ID = o.Customer\_ID";

The customer’s information is output as a table listing the customer’s Name, Shipping Address, Credit Card Number, Expiration Date, and Billing Address.

The user can then hit the “Process Order” button on this page. Their checkout is successful if at least one of the products are in stock. If a product is out of stock, it is still in their check out bin in the future. The admin can the process the order for the product that was in stock. If it is successful, then the inventory for these items is decreased by one and the order status is changed to “Processed”. This is shown by the following queries:

$sql2 = "

UPDATE product p, orders o, customer c

SET o.Order\_Status = 'Process'

WHERE c.Name = '$customerName'

AND o.Customer\_ID = c.Customer\_ID

AND o.Order\_Status = 'Check Out'

AND o.Product\_ID = p.Product\_ID

AND p.inventory > 0";

If the user has more than one item ready to check out and one of the items are out of stock, then the order is still processed for the items that are in stock, but the item that is out of stock is added to the customers wishlist for future reference. This is done by the following query:

$sql1 = "

UPDATE product p, orders o, customer c

SET o.Order\_Status = 'Wishlist'

WHERE c.Name = '$customerName'

AND o.Customer\_ID = c.Customer\_ID

AND o.Order\_Status = 'Check Out'

AND o.Product\_ID = p.Product\_ID

AND p.inventory = 0";

Since, when the customer signs up, they enter all their information, there is no need for this on the checkout.php page. Only registered users are allowed to enter our homepage.php and buy products.

*Q4. Wish list. Customer can add a configuration to her/his wish list.*

On the details.php page for a product the customer has the option to add the current configuration to their wish list by clicking the “add to wishlist” button

$sql="

INSERT INTO orders (Order\_ID,Order\_Status,Order\_Date,Product\_ID,Customer\_ID,Final\_Price,Memory,Disk,OS,Processor)

VALUES

(NULL,'" . $Type. "','". date("Y-m-d") ."'," . $PID. "," . $CID. "," . $Price. ",'" . $Memory. "','" . $Disk. "','" . $OS. "','" . $Processor. "')";

This query is also used in order to add a product to the customers, cart or buy now. It just depends on what the “$Type” is.

*Q5. Status. Customer can inquire about the status of a purchase.*

The customer can click on their name in the top right corner of the page. A drop down menu opens with “Cart”, “Wishlist”, and “Order History”. Clicking on “Order History” will take them to a page that shows them the status of their purchases. It will only show items whose Order Status is either “Process”, meaning the customer has checked out and paid for that item and the item has yet to be shipped out, or “Shipped”, meaning that admin has shipped out the order to the customer.

$sql = “

SELECT Order\_Status, Order\_Date, Final\_Price, Memory, Disk, OS, Processor, Product\_Name

FROM orders NATURAL JOIN product natural JOIN customer

WHERE Name='$customerName'

AND (Order\_Status='Shipped' OR Order\_Status='Process')";

The results are output as a table listing the Order Date, Product Name, OS, Processor, Memory, Disk, Final Price, and Order Status

**Administrators’ Transactions and Queries**

*Q1. Enter information for a new base system*

In the admin.php page, the admins have the options to:

* Search by date in the “Overview” tab
* Add a new product in the “Product” tab
* Add new configuration options in the “Config” tab
* Update a current product in the “Update” tab
* Ship out processed orders in the “Process” tab

In the “Product” tab, a table is shown listing all current products in the store and details of each. At the end of the table, the admin can add a new base system by adding information in each input box and clicking the “Add” button. The following query is then ran:

$sql="

INSERT INTO product ( Model,Product\_Name,Base\_Price,size,weight,Inventory)

VALUES

('$Model','$Product\_Name','$Base\_Price','$size','$weight','$Inventory')";

*Q2. Specify options (such as available memory upgrades for each base system – including the added price for each option.*

Under the “Config” tab, the admin can add new configuration upgrades for each base system. Admin can add new configuration for the processor, disk, and memory.

When the admin clicks the “Insert” button to insert a new processor, the following query is executed:

$sql="

INSERT INTO processor (Processor\_Type,Processor\_Upgrade\_Price)

VALUES

('$Processor\_Type','$Processor\_Upgrade\_Price')";

When the admin clicks the “Insert” button to insert a new disk, the following query is executed:

$sql="

INSERT INTO disk (Disk\_Type,Disk\_Upgrade\_Price)

VALUES

('$Disk\_Type','$Disk\_Upgrade\_Price')";

When the admin clicks the “Insert” button to add a new memory upgrade, the following query is executed:

$sql="

INSERT INTO memory (Memory\_Size,Memory\_Upgrade\_Price)

VALUES

('$Memory\_Size','$Memory\_Upgrade\_Price')";

*Q3. Update information regarding an existing system.*

In the “Update” tab, a table of the current products is displayed. At the bottom of the table, the admin can update either the product name, the model, the base price, or the inventory. This is done by the following queries.

$sql = "

UPDATE product

SET Product\_Name= '$Product\_Name'

WHERE Product\_ID = ".$Product\_ID."";

$sql2 = "

UPDATE product

SET Model= '$Model'

WHERE Product\_ID = ".$Product\_ID."";

$sql2 = "

UPDATE product

SET Base\_Price= '$Base\_Price'

WHERE Product\_ID = ".$Product\_ID."";

$sql2 = "

UPDATE product

SET Inventory= '$Inventory’

WHERE Product\_ID = ".$Product\_ID."";

*Q4. Process a purchase (the number of available systems in the inventory should be decremented, the status of the purchased changed to “processed”.)*

Under the “Process” tab, the admin actually ships out the orders. The current orders that are able to be shipped are displayed in a table. Our order status’s are just a bit different. It is the same concept and the inventory is still decreased. This is done by the following query:

$sql2 = "

UPDATE orders, product

SET product.Inventory = Inventory-1,orders.Order\_Status = 'Shipped'

WHERE Order\_ID = ".$Order\_ID."

AND orders.Product\_ID = product.Product\_ID ";

*Q5. Information listing.*

On the “Overview” tab, admin can see all shipped purchases and then search by a date range to see the purchases that have been shipped. This is displayed in a table and performed by the following query:

$sql2 = "

SELECT \*

FROM orders

WHERE Order\_Date BETWEEN '$Starting\_Date' AND '$End\_Date'

AND Order\_Status = 'Shipped'

ORDER BY Order\_Date DESC";

**Development Environment**

This website in running on a server, which is a virtual machine hosted on an EC2 t2.micro instance on Amazon Web Services. For more information about the Free Tier of the service, please see their website. https://aws.amazon.com/free/

The server is running a fairly standard LAMP stack with Ubuntu Server 14.04, Apache/2.4.7 (Ubuntu) , MySQL 14.14, and PHP 5.5.9.

To view this website demo, please follow the link below, input username: ‘csc671’ and password ‘671project’ at the prompt.  http://zhonghuang.sytes.net//csc671/

**Application Programs**

Index.php

Homepage.php

Search.php

Signup.php

Details.php

Order.php

Addorder.php

Addorder.php

Checkout.php