

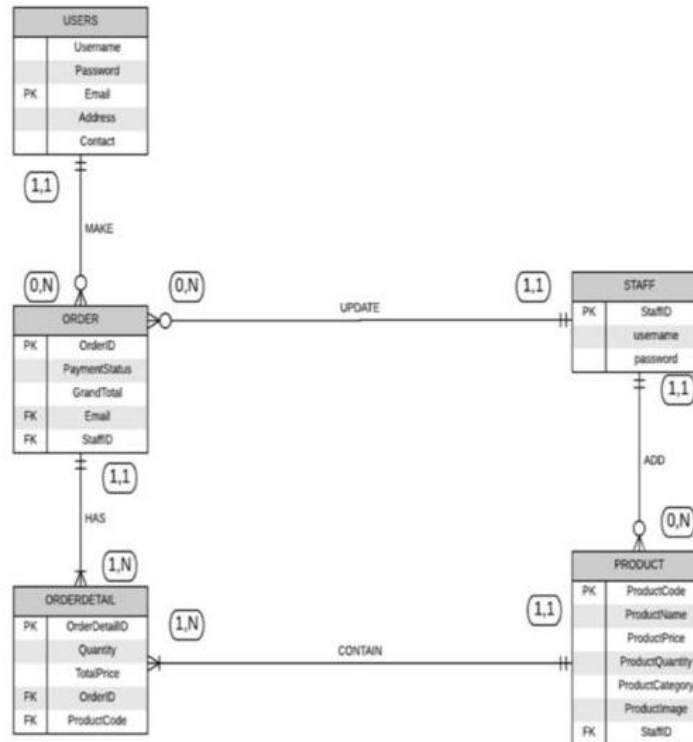
SHOPP

PHP E-Commerce site

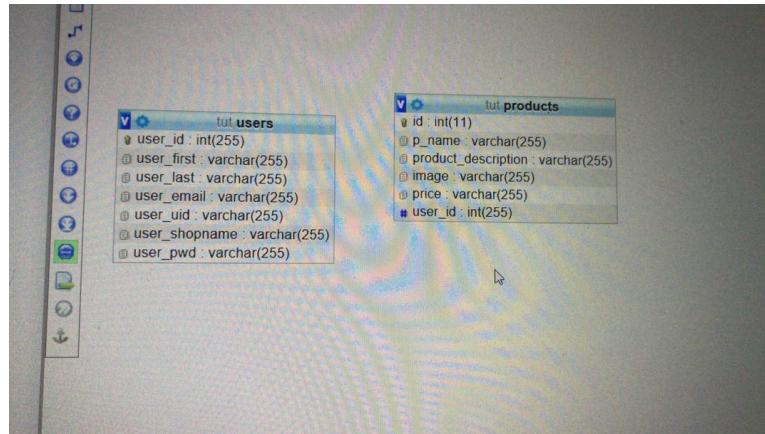
Group 1 - Database design and backend setup

DATABASE DESIGN

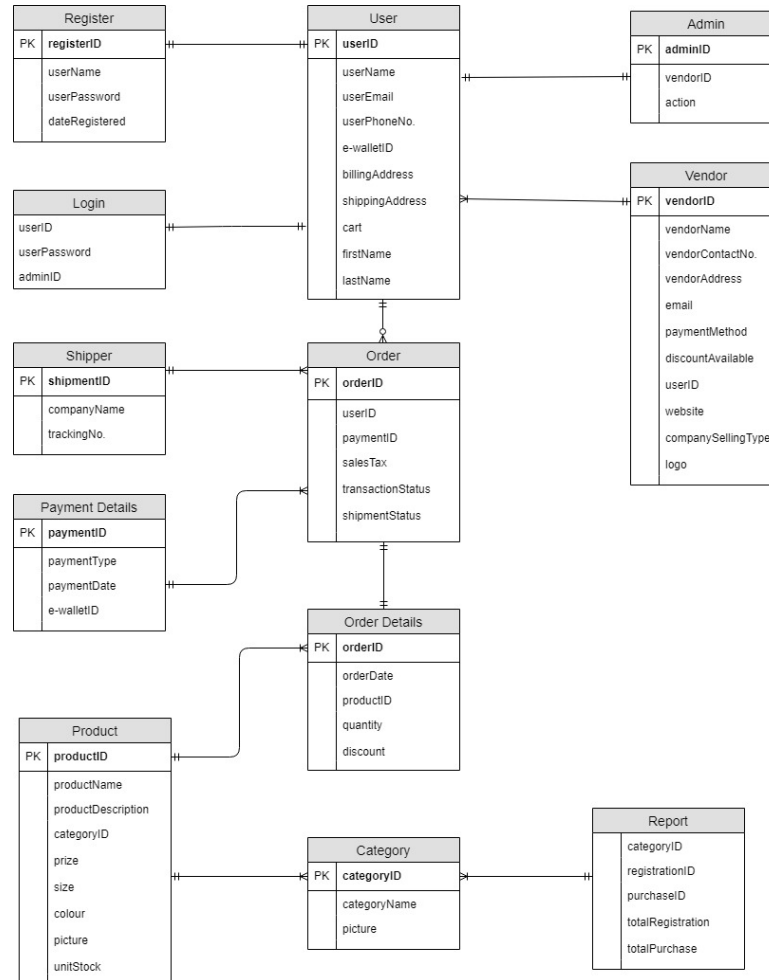
ERD



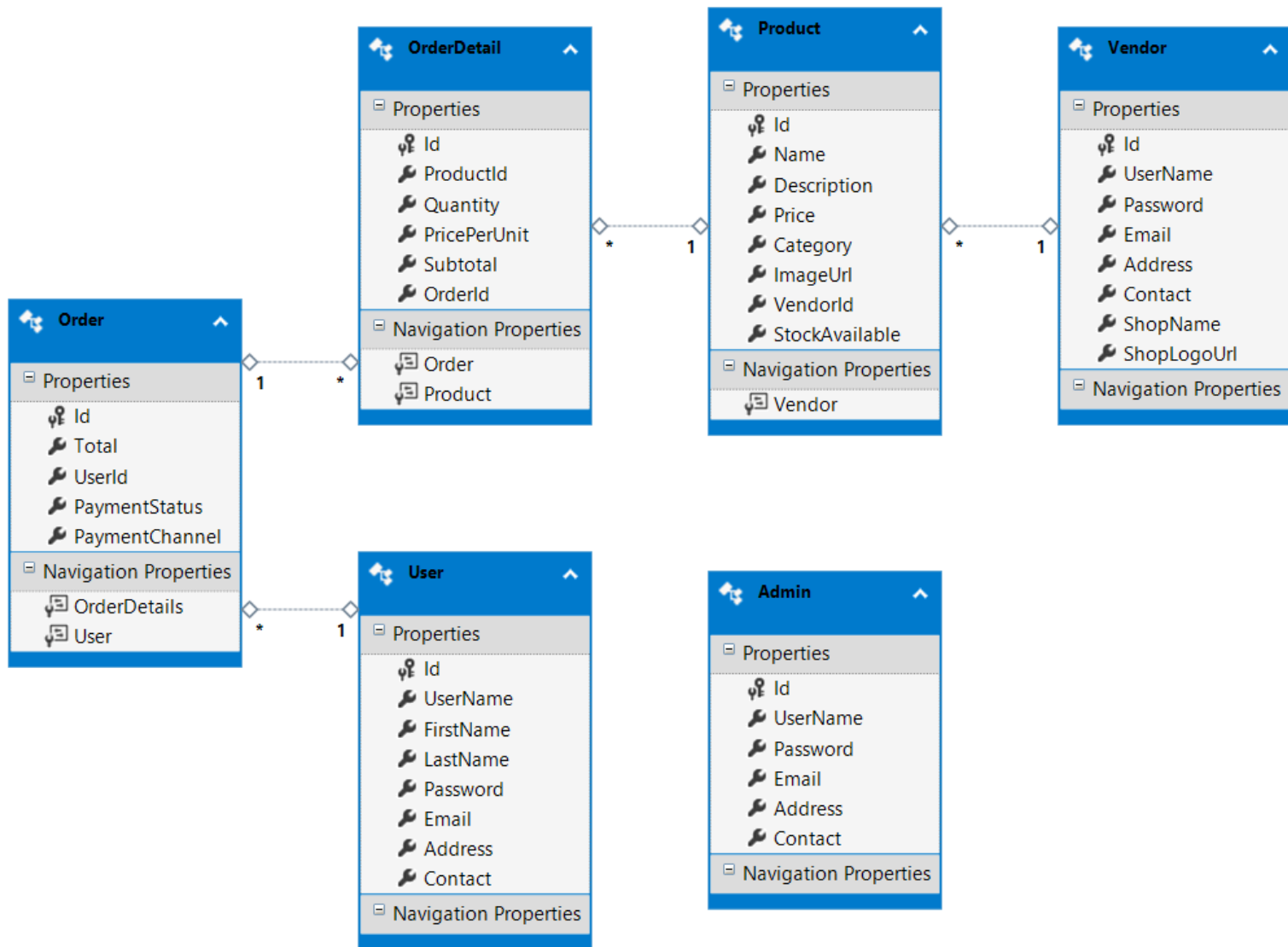
Group Sarmela



Group Nazmi



Group Eugene



RELATIONSHIPS

1. each User can have multiple Order
2. each Order can have multiple OrderDetail
3. each OrderDetail can have one Product
4. each Product belongs to one Vendor
5. Admin

POSSIBLE EXTENSION POINTS

1. persist items in Cart
2. choosing Shipment method and tracking status
3. displaying Promotion offered by Vendor

any questions/add ons?

CONNECTING PHP AND MYSQL

config.php

stores variables related to the database

```
<?php
$DB_HOST = 'mysql';
// $DB_HOST = 'localhost'

$DB_NAME = 'shopp';
$DB_USER = 'shopp';
$DB_PASSWORD = 'shopp';
?>
```

if you are using XAMPP,
replace `$DB_HOST = 'mysql';`
with `$DB_HOST = 'localhost';`

main.php

a barebone example of connecting to MySQL and
retrieve some Products

```
require_once 'config.php';
```

include the file `config.php` so that we can use the variables defined inside

```
class Product
{
    public $Id;
    public $Name;
    public $Price;
    public $Category;
    public $ImageUrl;
}
```

define a class that will be used to hold the result coming back from the database


```
try {  
    $connection = new PDO("mysql:host=$DB_HOST;dbname=$DB_NAME  
} catch (PDOException $e) {  
    die("Unable to connect to database $DB_NAME at $DB_HOST\n"  
}
```

establish a connection to the MySQL database
using PDO (PHP Data Objects) using the variables
defined in config.php

if it fails, print the error message and exit

```
$sql = 'SELECT *  
      FROM Products';  
$products = $connection->query($sql);
```

constructs and executes the SQL query to get all
Products from the database

you should use

1. prepare
2. execute

to prevent SQL injection

<https://phpdelusions.net/pdo#prepared>

```
// ...
<?php while ($product = $products->fetchObject('Product')) {
    echo '<tr>';
    echo '<td> ' . $product->Name . '</td>';
    echo '<td> ' . $product->Price . '</td>';
    echo '<td> ' . $product->Category . '</td>';
    echo '<td> 



# laradock

a PHP development environment for docker

<https://laradock.io/>

# BUT...

you can still use XAMPP if you want to

# DEMO

# GETTING THE CODE

by either `git` cloning, or just copy from the cd

<https://github.com/garyng/Shopp>

# CLONING FROM GITHUB

```
git clone git@github.com:garyng/Shopp.git
```

# STARTING THE DEVELOPMENT ENVIRONMENT

1. mysql
2. apache2

*the same thing if you are using XAMPP, but you will need to copy this folder into your XAMPP folder*

```
pushd src/web/laradock
docker-compose up apache2 mysql
```

# main.php

my server is running at port 8080, change it  
according to your XAMPP settings

<http://localhost:8080/main.php>



# CREATING AND SEEDING THE DATABASE

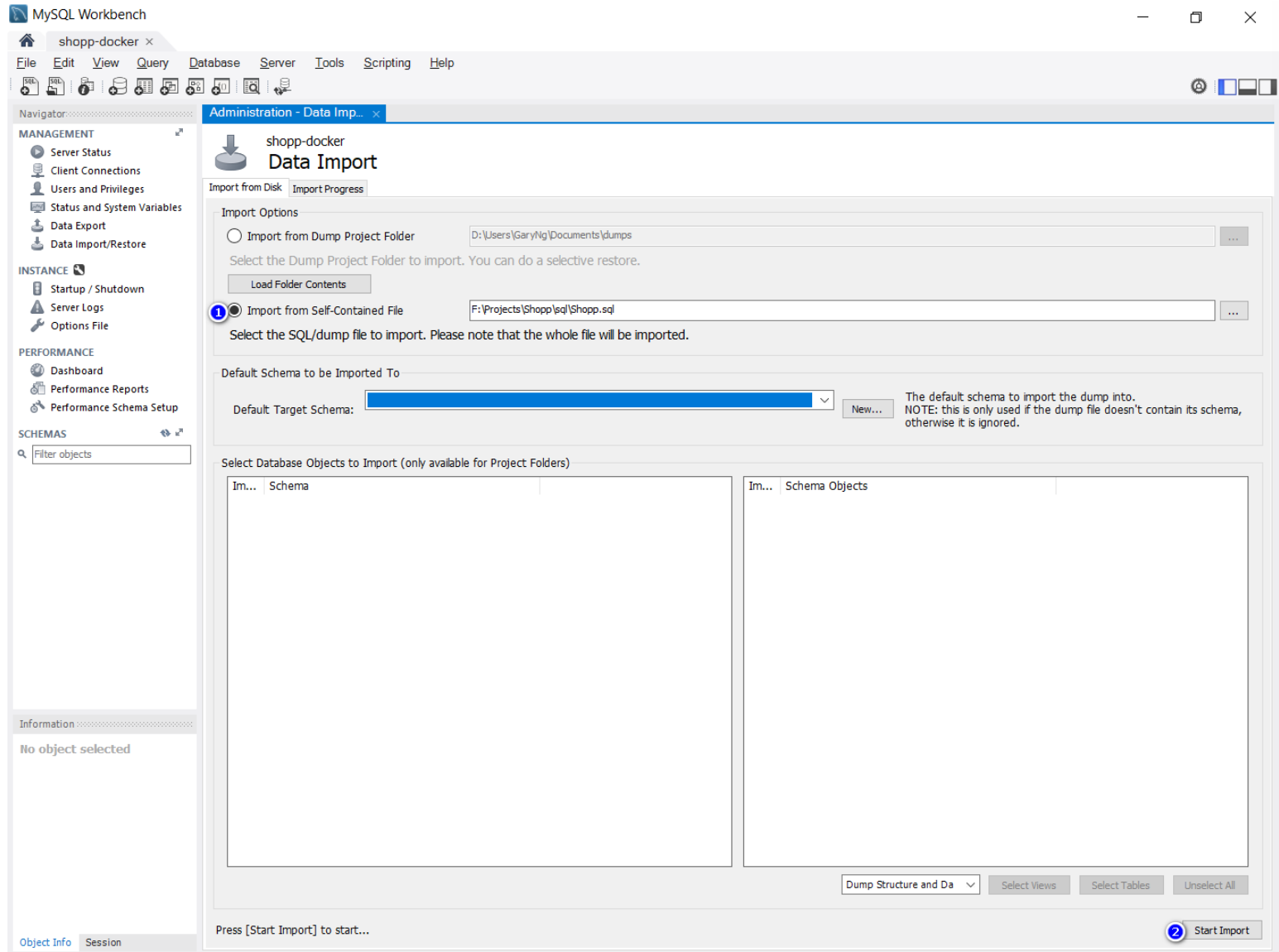
you can import the `Shopp.sql` file, or use the application under `database-init`.

# database-init

```
pushd sql\database-init
Shopp.exe
```

# **Shopp.sql**

import it with PHPMyAdmin or MySQL Workbench



# CHECKING THE DATABASE

with MySQL Workbench or PHP My Admin

# DONE!

<http://localhost:8080/main.php>

**THANKS!**