

MANUAL TÉCNICO

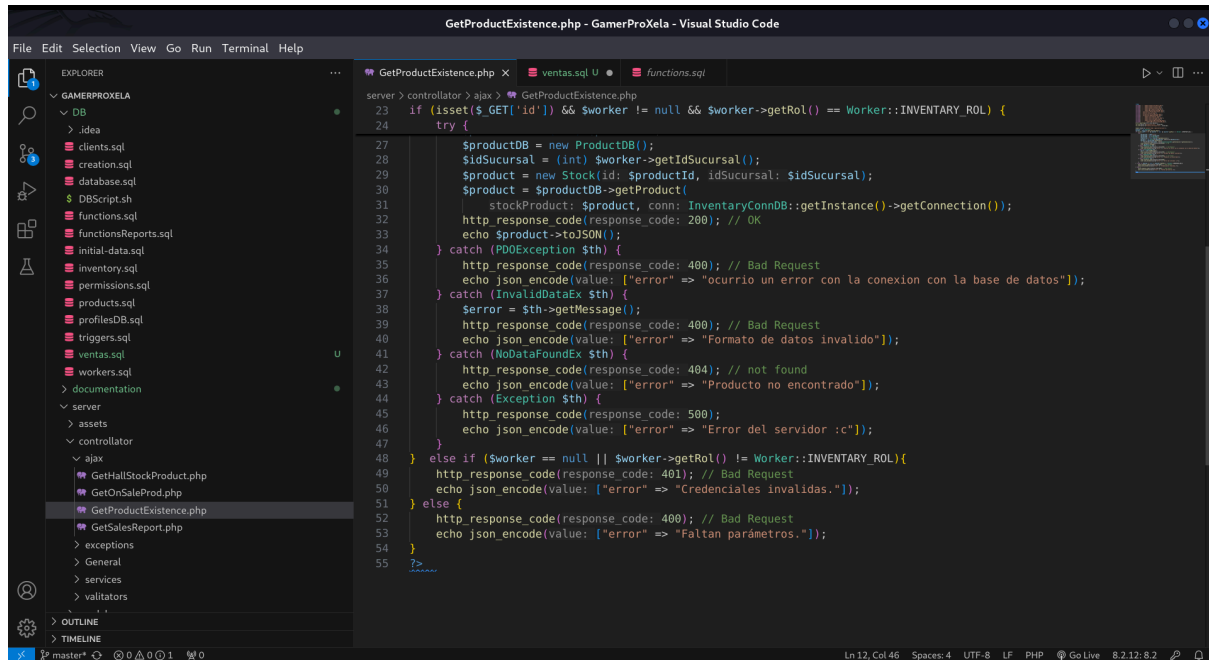


ÍNDICE

ÍNDICE.....	1
TECNOLOGÍA USADA PARA EL DESARROLLO DEL PROYECTO.....	2
REQUERIMIENTOS PARA EL FUNCIONAMIENTO DEL PROGRAMA.....	3
DIAGRAMA DE CLASES.....	4
SERVIDOR.....	5
ANÁLISIS DE LA BASE DE DATOS.....	5
DIAGRAMA DE ENTIDAD/RELACIÓN.....	5
DIAGRAMA DE TABLAS.....	5
CONFIGURACIÓN DE LA BASE DE DATOS.....	5

TECNOLOGÍA USADA PARA EL DESARROLLO DEL PROYECTO

- Visual Studio Code V.1.93.1



```
server > controller > ajax > GetProductExistence.php
23 if (isset($_GET['id']) && $worker != null && $worker->getRol() == Worker::INVENTORY_ROL) {
24     try {
27         $productDB = new ProductDB();
28         $idSucursal = (int) $worker->getIdSucursal();
29         $product = new Stock(id: $productId, idSucursal: $idSucursal);
30         $product = $productDB->getProduct(
31             stockProduct: $product, conn: InventoryConnDB::getInstance()->getConnection());
32         http_response_code(response_code: 200); // OK
33         echo $product->toJson();
34     } catch (PDOException $th) {
35         http_response_code(response_code: 400); // Bad Request
36         echo json_encode(value: ["error" => "ocurrio un error con la conexion con la base de datos"]);
37     } catch (InvalidDataEx $th) {
38         $error = $th->getMessage();
39         http_response_code(response_code: 400); // Bad Request
40         echo json_encode(value: ["error" => "Formato de datos invalido"]);
41     } catch (NotFoundException $th) {
42         http_response_code(response_code: 404); // not found
43         echo json_encode(value: ["error" => "Producto no encontrado"]);
44     } catch (Exception $th) {
45         http_response_code(response_code: 500);
46         echo json_encode(value: ["error" => "Error del servidor :c"]);
47     }
48 } else if ($worker == null || $worker->getRol() != Worker::INVENTORY_ROL){
49     http_response_code(response_code: 401); // Bad Request
50     echo json_encode(value: ["error" => "Credenciales invalidas."]);
51 } else {
52     http_response_code(response_code: 400); // Bad Request
53     echo json_encode(value: ["error" => "Faltan parametros."]);
54 }
55 ?>
```

- Sistema Operativo/Servidor:

KALI LINUX

Distributor ID:Kali

Description: Kali GNU/Linux Rolling

Release: 2024.3

Codename: kali-rolling



REQUERIMIENTOS PARA EL FUNCIONAMIENTO DEL PROGRAMA

★ Php

PHP 8.2.21 (cli) (built: Jul 25 2024 10:12:09) (NTS)

Copyright (c) The PHP Group

Zend Engine v4.2.21, Copyright (c) Zend Technologies

with Zend OPcache v8.2.21, Copyright (c), by Zend Technologies



★ PostgreSQL

psql (PostgreSQL) 16.4 (Debian 16.4-1)



★ Apache2

Server version: Apache/2.4.62 (Debian)



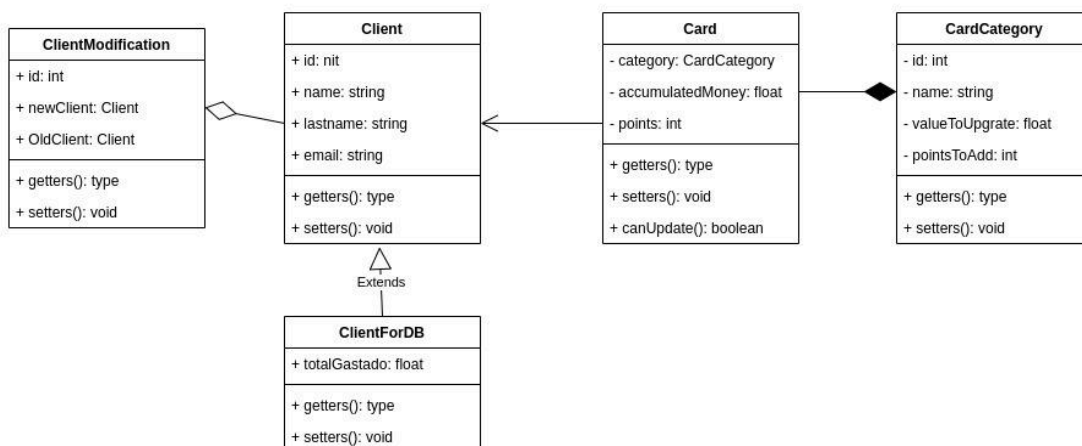
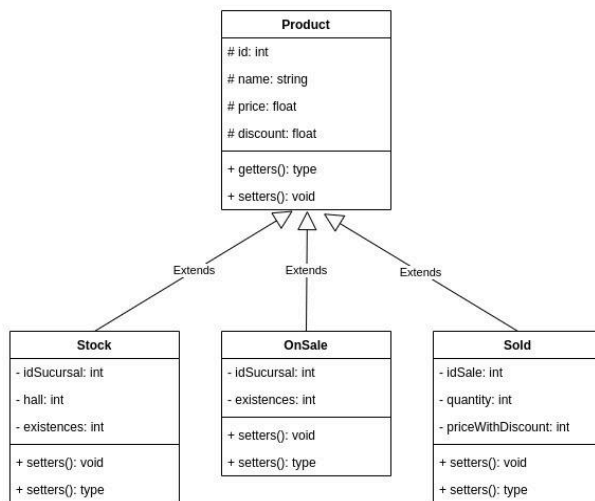
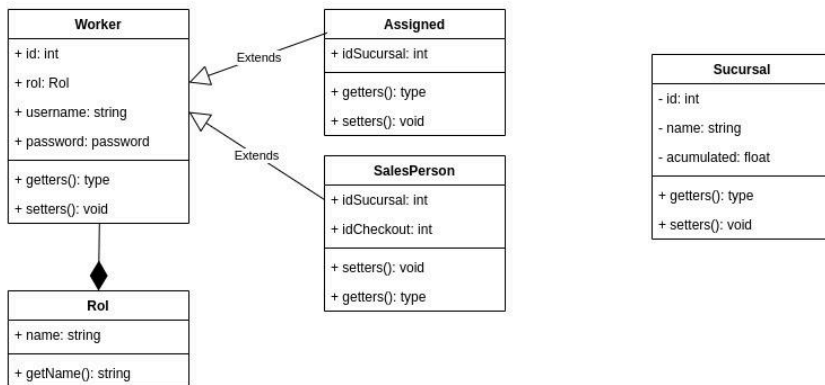
★ XAMPP

(Opcional, debería incluir los requerimientos anteriores).

Versión: 8.2.12



DIAGRAMA DE CLASES



SERVIDOR

Para iniciar el servidor usando XAMPP se debe hacer lo siguiente:

```
//clonar el repositorio en la carpeta por defecto de xampp para mostrar las paginas web
cd /opt/lampp/htdocs
git clone https://github.com/Yennifer017/GamerProXela_php

//iniciar el servidor
cd /opt/lampp
sudo ./xampp startapache
```

Ya se podra ver la pagina en la siguiente dirección en el navegador de preferencia

```
http://localhost/GamerProXela/server/view/homepage.html
```

ANÁLISIS DE LA BASE DE DATOS

DIAGRAMA DE ENTIDAD/RELACIÓN

Se encuentra en la misma carpeta de este documento.

```
./GamerProXela-ER.jpg
```

DIAGRAMA DE TABLAS

Se encuentra en la misma carpeta de este documento.

```
./GamerProXela-TD.jpg
```

CONFIGURACIÓN DE LA BASE DE DATOS

Antes de poder ejecutar la aplicación se deben tener los requerimientos especificados anteriormente. Luego se debe de ejecutar los siguientes comandos:

```
//iniciar el servicio de postgresql
sudo systemctl start postgresql

//ir a la carpeta de DB,
cd <path>/DB

//crear usuarios, esto se hara una sola vez
sudo -u postgres psql -f profilesDB.sql

//darle permisos al script para cargar toda la base de datos
sudo chmod +x DBScript.sh

//cargar todo lo necesario,
//ADVERTENCIA: esto ejecuta un DROP DATABASE gamer_pro_xela;
sudo ./DBScript.sh
```