**PROYECTO TAD – GRUPO 3**

**PRODUCTOS DE BELLEZA PROFESIONAL: UPOCOSMETIC**



**Autores:**

**Luna De Torres Espejo**

**Kacper Zaleski**

**Jesús María Sena Franconetti**

**INDICE:**

1. **HISTORIAL DE VERSIONES ………………………………………………………… Página 3**
2. **OBJETIVOS Y REQUISITOS ………………………………………………………… Página 3**
3. **ARQUITECTURA DE BASE DE DATOS ………………………………………… Página 4**
4. **CASOS DE USOS ………………….…………………………………………………. Página 5**
5. **SISTEMA DE CONTROL DE VERSIONES CON GITHUB ……………..… Página 10**
6. **DISEÑOS DEL PROYECTO ………………………………………………………… Página 10**
7. **CORRECCIONES ……………………………………………………………………… Página 10**
8. **HISTORIAL DE VERSIONES:**

|  |  |  |
| --- | --- | --- |
| VERSION | CONTENIDO | FECHA |
| V.1 | Desarrollo del panel de administración | 14/04/2023 |
| V.2 | - Correcciones V.1 | 05/05/2023 |

1. **OBJETIVOS Y REQUISITOS**

Se desarrollará un proyecto basado de una tienda de cosméticos online.

Se necesitará un usuario que podrá comprar productos de estética, ver el estado de su perdido y recibir un correo electrónico con los detalles de su pedido. El usuario deberá autentificarse previamente.

También se necesitará un administrador que ejecute operaciones sobre los productos de estética y pueda comprobar los pedidos.

El proyecto deberá desarrollarse en Laravel. Puntos a destacar para su desarrollo:

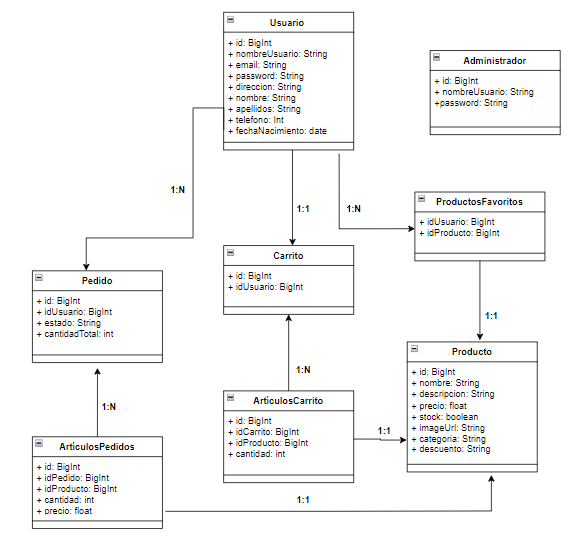
* Operaciones GRUD para compra y productos.
* Análisis de la base de datos.
* Uso de login-Signin.
* Diseño de la web.
* Recuperación de contraseña, SMTP con Laravel.

Para todo ello se deberá:

* Tener un sistema de versiones en Github, para los miembros del equipo.
* Diseñar una arquitectura de base de datos y sus casos de uso.
* Mockups del diseño de la web.
* Implementación en lenguaje php, haciendo uso de bootstrap5

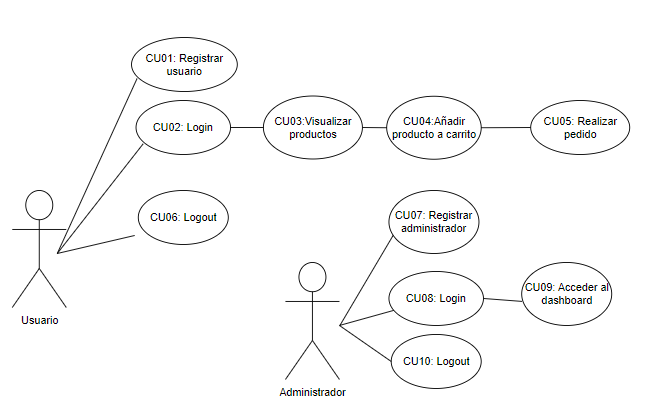
1. **ARQUITECTURA DE BASE DE DATOS**

Se ha desarrollado una arquitectura de la base de datos de la tienda junto con sus relaciones.



1. **CASOS DE USOS**

Diagrama de los casos de uso:



**Descripción de los casos de uso:**

**CU01. REGISTRAR USUARIO**

**DESCRIPCIÓN**: Se registrará un nuevo usuario en el sistema.

**ACTORES**: Usuario.

**PRECONDICIÓN:** Ninguna.

**POSTCONDICIÓN:** Se ha registrado un nuevo usuario en el sistema.

**FLUJO NORMAL:**

1. El usuario accede a la opción “registrar usuario”.
2. El sistema solicita introducir nombre, email, password, confirmación password,

del usuario.

1. El usuario introduce los datos en el sistema.
2. El sistema añade al usuario a la lista de usuarios e informa que se ha

añadido el nuevo usuario correctamente.

**FLUJO ALTERNATIVO: Línea2. Datos usuario incorrectos**

1. El sistema informa del error y solicita de nuevo los datos.

**FLUJO ALTERNATIVO: Línea4. Usuario ya existe**

1. El sistema informa del error y se finaliza el caso de uso.

**CU02. LOGIN**

**DESCRIPCIÓN**: Se logará el usuario en el sistema.

**ACTORES**: Usuario.

**PRECONDICIÓN:** El usuario debe estar registrado en el sistema.

**POSTCONDICIÓN:** El usuario accede al sistema.

**FLUJO NORMAL:**

**1**. El usuario accede a la opción “Login”.

1. El sistema solicita introducir email y password

del usuario.

1. El usuario introduce los datos en el sistema.
2. El sistema accede a la compra de productos.

**FLUJO ALTERNATIVO: Línea2. Datos login del usuario incorrectos**

**1**. El sistema informa del error y solicita de nuevo los datos.

**CU03. LOGOUT**

**DESCRIPCIÓN**: Se deslogará el usuario en el sistema.

**ACTORES**: Usuario.

**PRECONDICIÓN:** El usuario debe logado en el sistema.

**POSTCONDICIÓN:** El usuario sale del sistema.

**FLUJO NORMAL:**

1. El usuario accede a la opción “Logout”.
2. El sistema accede a login.

**CU04. VISUALIZAR PRODUCTOS**

**DESCRIPCIÓN**: el usuario visualizará los productos del sistema.

**ACTORES**: Usuario.

**PRECONDICIÓN:** El usuario debe logado en el sistema.

**POSTCONDICIÓN:** ninguna.

**FLUJO NORMAL:**

1. El usuario accede a visualizar los productos.

**CU05. AÑADIR PRODUCTO A CARRITO**

**DESCRIPCIÓN**: el usuario añadirá productos a su carrito de compra.

**ACTORES**: Usuario.

**PRECONDICIÓN:** El usuario debe logado en el sistema.

**POSTCONDICIÓN:** El producto quedará añadido al carro de la compra.

**FLUJO NORMAL:**

1. El usuario accede a visualizar los productos y selecciona un producto.
2. El sistema añade el producto al carrito de la compra del usuario.

**CU05. REALIZAR PEDIDO**

**DESCRIPCIÓN**: el usuario realizará el pedido de compra.

**ACTORES**: Usuario.

**PRECONDICIÓN:** El usuario debe logado en el sistema y debe de tener productos en su carrito de compras.

**POSTCONDICIÓN:** El usuario realizó un pedido.

**FLUJO NORMAL:**

1. El usuario accede a visualizar los productos en su carrito de compra y envía el pedido.
2. El sistema realizar el pedido del usuario.

**FLUJO ALTERNATIVO: Línea1. Usuario cancela pedido**

1. El sistema informa al usuario que canceló el pedido.

**CU06. REGISTRAR ADMINISTRADOR**

**DESCRIPCIÓN**: Se registrará un nuevo administrador en el sistema.

**ACTORES**: administrador.

**PRECONDICIÓN:** Ninguna.

**POSTCONDICIÓN:** Se ha registrado un nuevo administrador en el sistema.

**FLUJO NORMAL:**

**1.** El usuario accede a la opción “registrar administrador”.

**2**. El sistema solicita introducir nombre, email, password, confirmación password,

del administrador.

**3**. El administrador introduce los datos en el sistema.

**4**. El sistema añade al usuario a la lista de administrador e informa que se ha

añadido el nuevo usuario correctamente.

**FLUJO ALTERNATIVO: Línea2. Datos administrador** **incorrectos**

**1**. El sistema informa del error y solicita de nuevo los datos.

**FLUJO ALTERNATIVO: Línea4. administrador** **ya existe**

**1**. El sistema informa del error y se finaliza el caso de uso.

**CU07. LOGIN**

**DESCRIPCIÓN**: Se logará el administrador en el sistema.

**ACTORES**: administrador.

**PRECONDICIÓN:** El administrador debe estar registrado en el sistema.

**POSTCONDICIÓN:** El administrador accede al sistema.

**FLUJO NORMAL:**

**1**. El administrador accede a la opción “Login”.

1. El sistema solicita introducir email y password

del administrador.

1. El administrador introduce los datos en el sistema.
2. El administrador accede al sistema.

**FLUJO ALTERNATIVO: Línea2. Datos login del administrador** **incorrectos**

**1**. El sistema informa del error y solicita de nuevo los datos.

**CU08. LOGOUT**

**DESCRIPCIÓN**: Se deslogará el administrador en el sistema.

**ACTORES**: administrador.

**PRECONDICIÓN:** El administrador debe logado en el sistema.

**POSTCONDICIÓN:** El administrador sale del sistema.

**FLUJO NORMAL:**

1. El administrador accede a la opción “Logout”.
2. El sistema accede a login.
3. **SISTEMA DE CONTROL DE VERSIONES CON GITHUB**

Para el desarrollo del proyecto creamos un repositorio en GitHub para que todo el equipo pudiese trabajar coordinado, pero al tener problemas técnicos entre los equipos de los miembros al actualizar el proyecto, decidimos no actualizarlos todos.

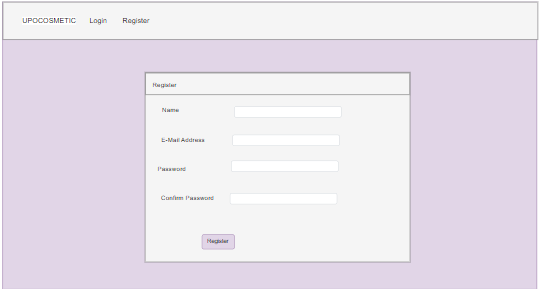
Enlace:

<https://github.com/Lunadetorres/ProyectoTAD>

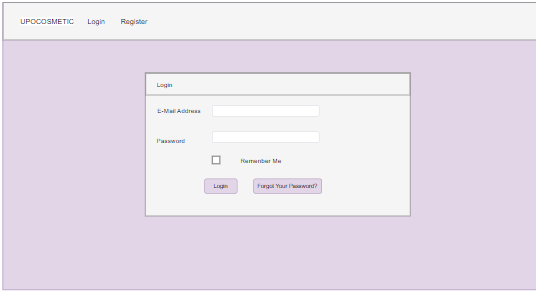
1. **DISEÑO DEL PROYECTO**

En este apartado destacamos los diseños más importante y aportamos Mockups

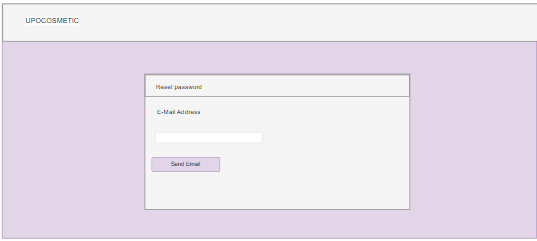
Registrar Usuario:



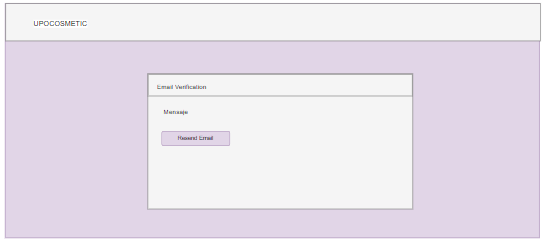
Login:



Olvidar contraseña:



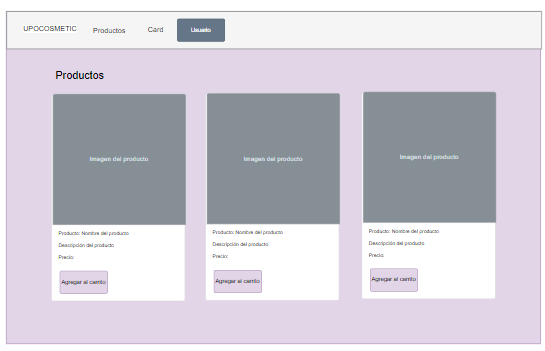
Verificar Email:



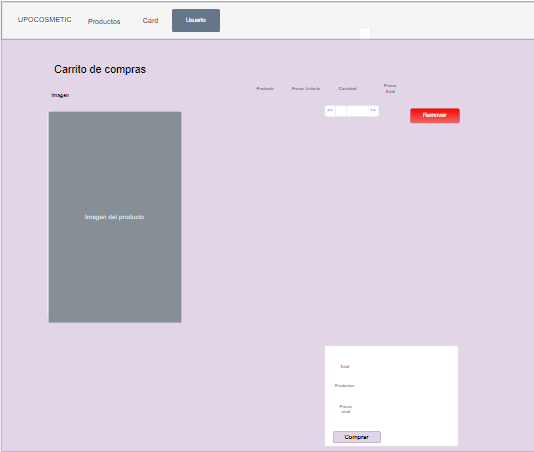
Dashboard:



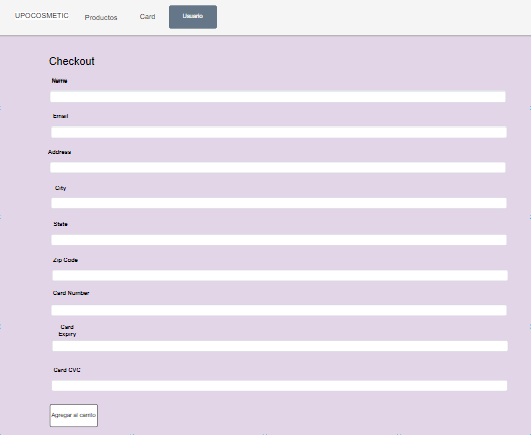
Visualizar Productos:



Carrito de la compra:



Checkout:



1. **CORRECIONES**