



Instituto Tecnológico de Costa Rica

Sede Cartago

Ing. Computación

IC-7841 Proyecto de Ingeniería de Software

Manual Técnico

**Sistema de ventas web de maquillaje y productos
“Galería de Jeca”**

Preparado por:

Kevin Jiménez Loría
Esteven Fernández Hernández
Fabricio Ríos Montero

Verano 2023-2024

18/01/2024

Tabla de contenidos

Introducción.....	4
Objetivos.....	4
General.....	4
Específicos.....	4
Requisitos del sistema.....	5
Requerimientos de software.....	5
NodeJS.....	5
Navegador web.....	5
Editor de código.....	5
Requerimientos de hardware.....	5
Herramientas utilizadas para el desarrollo.....	5
NodeJS.....	5
Firebase.....	5
Instalación de aplicaciones.....	6
Instalación de NodeJS.....	6
Instalación de Firebase.....	6
Modelo de clases.....	7
Casos de uso.....	8
Diagrama de casos de uso.....	8
Módulo de gestión de persona.....	8
Módulo de gestión de acceso.....	8
Módulo de gestión de galería.....	9
Módulo de gestión de producto.....	9
Módulo de gestión compra.....	10
Módulo de gestión de carrito de compra.....	10
Módulo de gestión de cursos.....	11
Módulo de gestión de consultas.....	11
Descripción de casos de uso.....	12
Distribución de módulos.....	14
Módulo de gestión de acceso.....	14
Módulo de gestión de galería.....	14

Módulo de gestión de productos.....	14
Módulo de gestión de compras.....	14
Módulo de gestión de carrito de compras.....	14
Módulo de gestión de cursos.....	14
Módulo de gestión de gestión de consultas.....	14
Módulo de gestión de agenda.....	14
Módulo de gestión de inventario.....	14
Manejo de Excepciones y Errores.....	15

Introducción

El propósito de este manual técnico es proporcionar una guía detallada sobre el desarrollo de una aplicación destinada a una tienda de maquillaje. La aplicación abarcará la gestión de productos de maquillaje, la oferta de cursos de maquillaje y la posibilidad de adquirir citas para realizar el proceso de maquillarse como quiera y para cualquier situación.

Además, que el proyecto estará ambientado en NodeJS para los apartados de *frontend* y el *backend* y Firebase como medio de base de datos, por lo que se describirán los objetivos, requisitos del sistema, herramientas que se utilizarán y otros aspectos esenciales para comprender el proceso de desarrollo.

Objetivos

General

Desarrollar una aplicación web moderna y escalable que permita a los usuarios explorar y comprar productos de maquillaje, inscribirse en los cursos de maquillaje que se ofrecen y poder agendar citas para el proceso de maquillaje profesional.

Específicos

1. Implementar un sistema de compras de productos de maquillaje con carrito de compras, poder editar y gestionar la compra a gusto.
2. Ofrecer un espacio de cursos de maquillaje con inscripciones en línea con todos los datos necesarios para ofrecerle al cliente un excelente servicio.
3. Facilitar la búsqueda y reserva de citas para maquillarse según sea la necesidad del cliente y la disponibilidad de la profesional en la materia.

Requisitos del sistema

Requerimientos de software

NodeJS

Se debe contar con una versión 12 o superior.

Navegador web

Se debe contar con un navegador web moderno, ya sea Chrome, Firefox, Safari, entre otros.

Editor de código

Se debe contar con algún editor de código para modificaciones como, por ejemplo: VSCode, Sublime Text, Atom, entre otros.

Requerimientos de hardware

- Procesador dual-core o superior
- 4 GB de RAM
- Conexión a Internet para la integración con Firebase y acceso a cursos en línea.

Herramientas utilizadas para el desarrollo

NodeJS

Plataforma de ejecución de JavaScript para el servidor.

Firebase

Plataforma de desarrollo de aplicaciones web y móviles de Google.

Instalación de aplicaciones


Instalación de NodeJS


Se deberá descargar e instalar NodeJS desde su página oficial <https://nodejs.org>, según el sistema operativo empleado.


Downloads
Latest LTS Version: 20.11.0 (includes npm 10.2.4)
Download the Node.js source code or a pre-built installer for your platform, and start developing today.

LTS
Recommended For Most Users

Current
Latest Features


Windows Installer
node-v20.11.0-x64.msi


macOS Installer
node-v20.11.0.pkg


Source Code
node-v20.11.0.tar.gz

Windows Installer (.msi)	32-bit	64-bit	ARM64
Windows Binary (.zip)	32-bit	64-bit	ARM64
macOS Installer (.pkg)	64-bit / ARM64		
macOS Binary (.tar.gz)	64-bit	ARM64	
Linux Binaries (x64)	64-bit		
Linux Binaries (ARM)	ARMv7	ARMv8	
Source Code	node-v20.11.0.tar.gz		

Una vez se haya descargado se procede a generar la instalación y sigue los pasos requeridos para finalizar la instalación.

Instalación de Firebase

Luego se debe realizar la instalación del SDK de Firebase usando el siguiente comando:

```
npm install firebase
```

Se vería de la siguiente manera:

```
$ npm install firebase
```

Una vez instalado el SDK se realiza la conexión con la base de datos con Firebase con la configuración de la base de datos creada la siguiente información:

apiKey, authDomain, projectId, storageBucket, messagingSenderId, appId, measurementId.

Y, por último, inicializar Firebase.

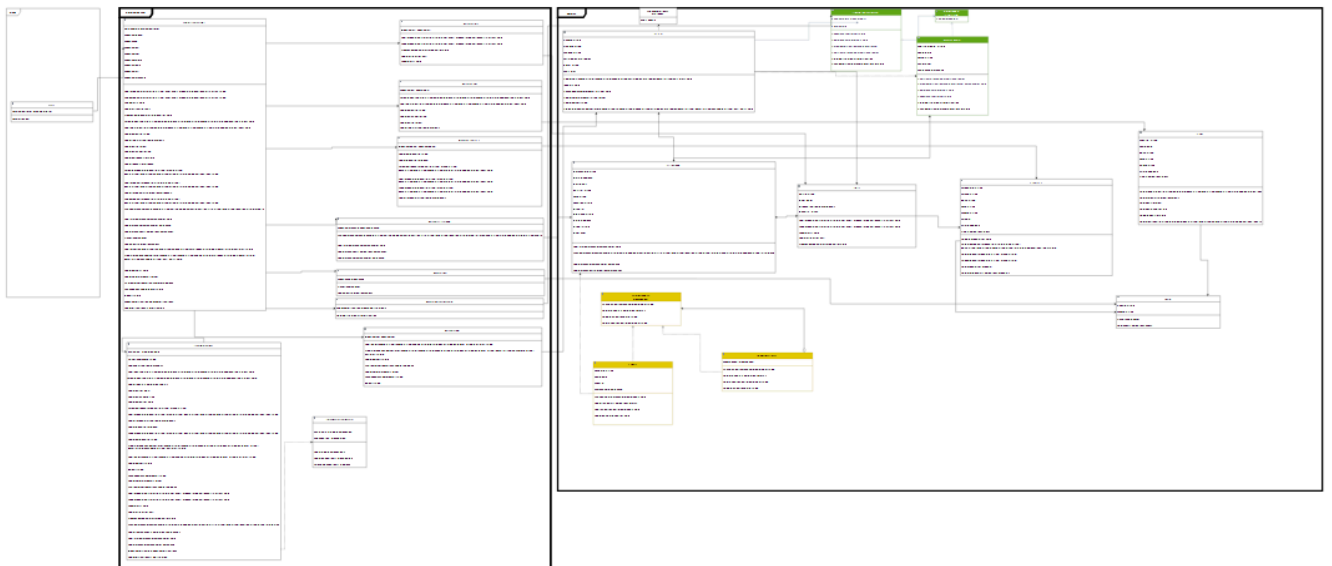
A modo de ejemplo se presentará una conexión de una base de datos en Firebase donde muestra todo lo explicado anteriormente:

```
// Import the functions you need from the SDKs you need
import { initializeApp } from "firebase/app";
import { getAnalytics } from "firebase/analytics";
// TODO: Add SDKs for Firebase products that you want to use
// https://firebase.google.com/docs/web/setup#available-libraries

// Your web app's Firebase configuration
// For Firebase JS SDK v7.20.0 and later, measurementId is optional
const firebaseConfig = {
  apiKey: "AIzaSyAv0cRQ9EzSKzsKjdBMC4u5QyD_jGcTfNs",
  authDomain: "proyecto-d3279.firebaseio.com",
  projectId: "proyecto-d3279",
  storageBucket: "proyecto-d3279.appspot.com",
  messagingSenderId: "301923966478",
  appId: "1:301923966478:web:e9938118456901722fa65f",
  measurementId: "G-T9BSM172WF"
};

// Initialize Firebase
const app = initializeApp(firebaseConfig);
const analytics = getAnalytics(app);
```

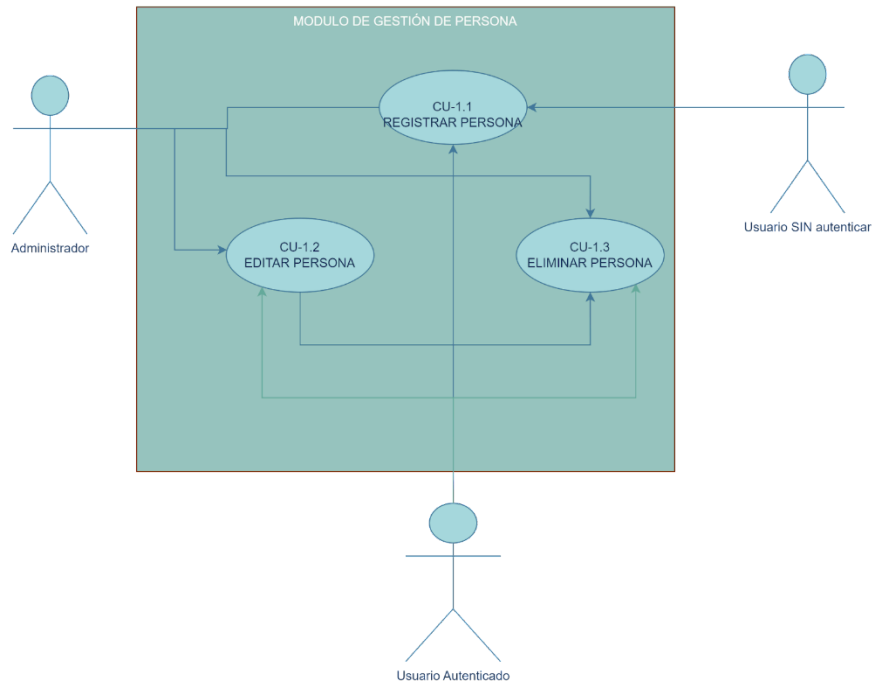
Modelo de clases



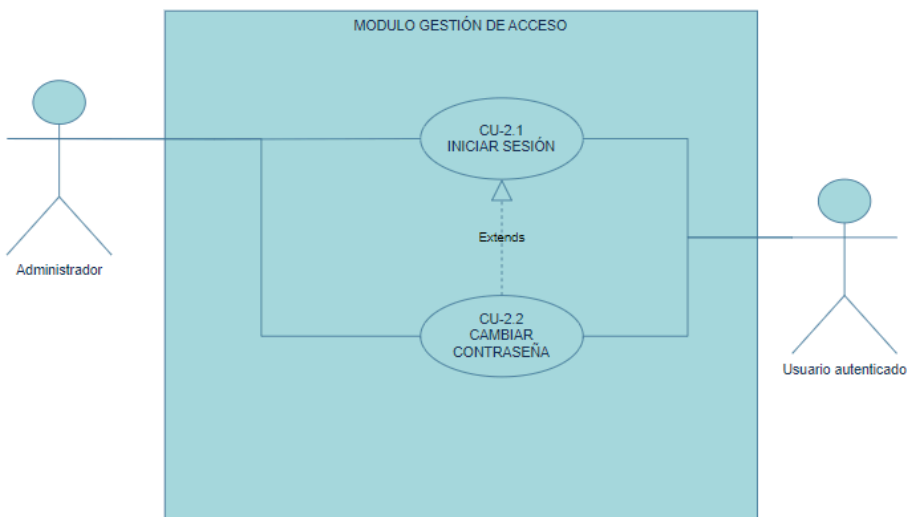
Casos de uso

Diagrama de casos de uso

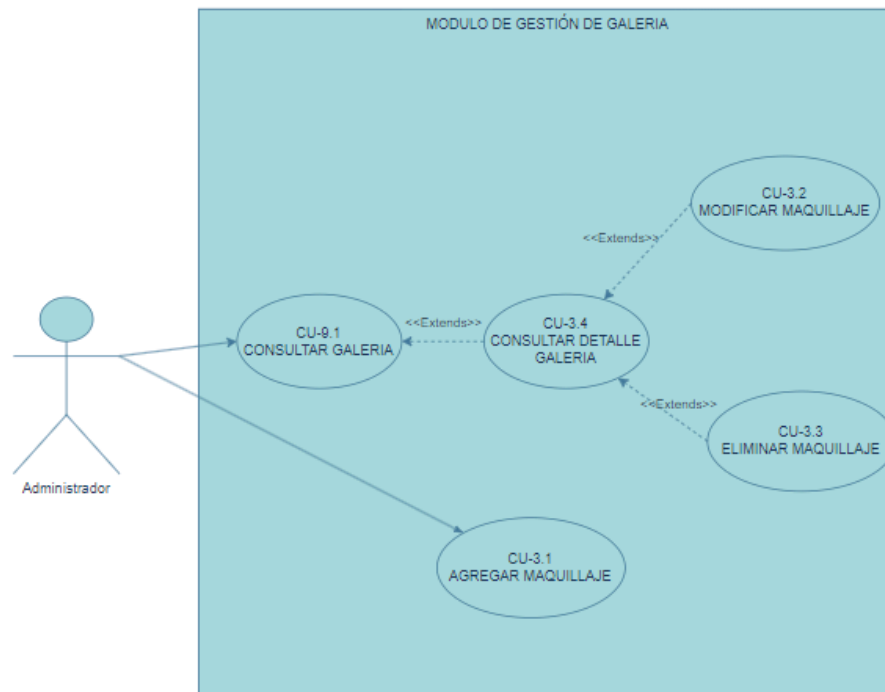
Módulo de gestión de persona



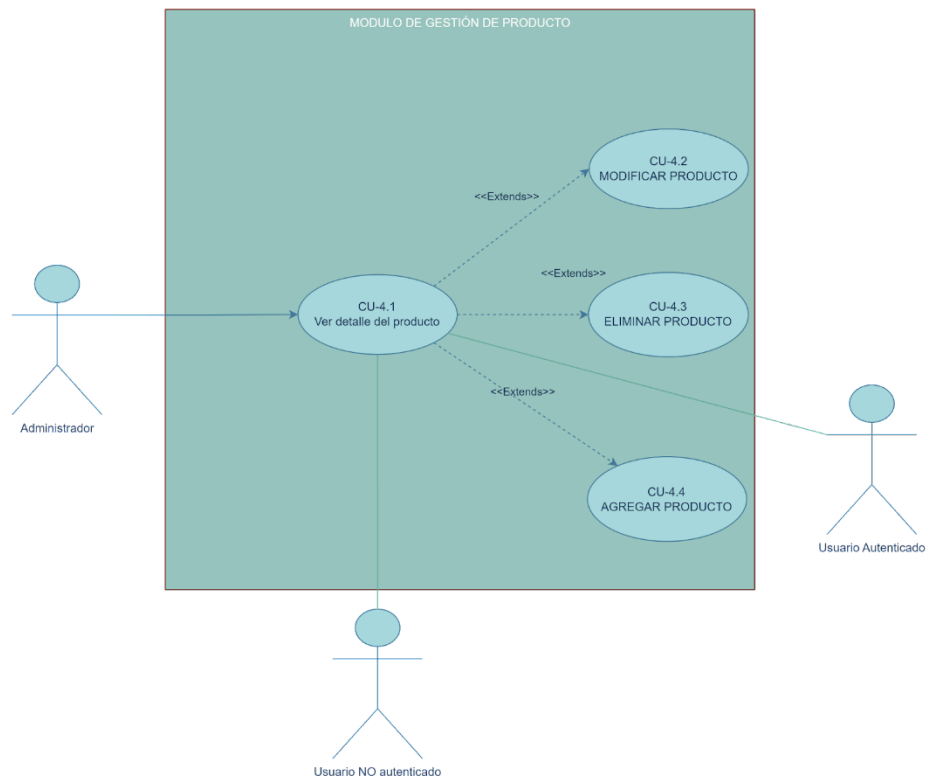
Módulo de gestión de acceso



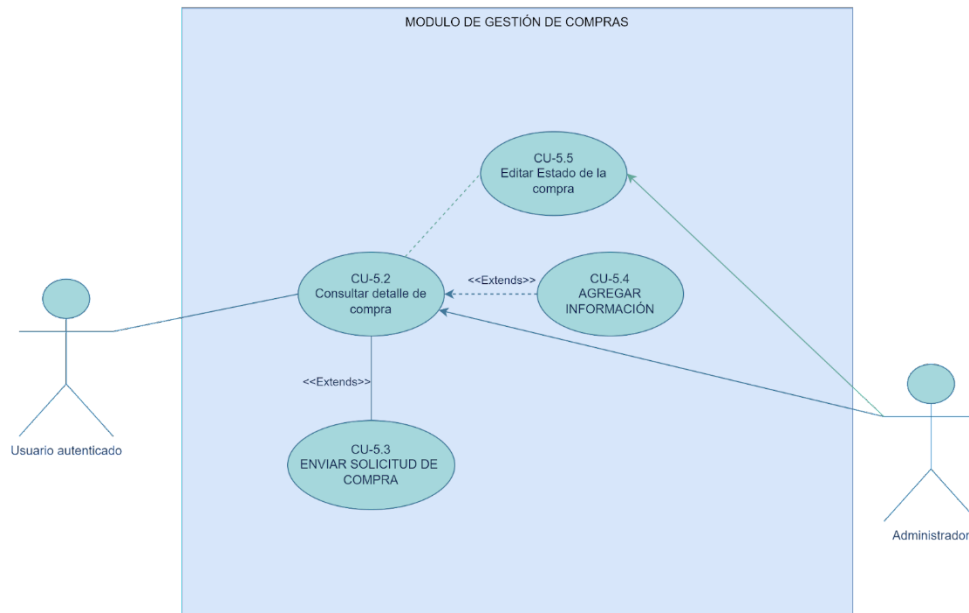
Módulo de gestión de galería



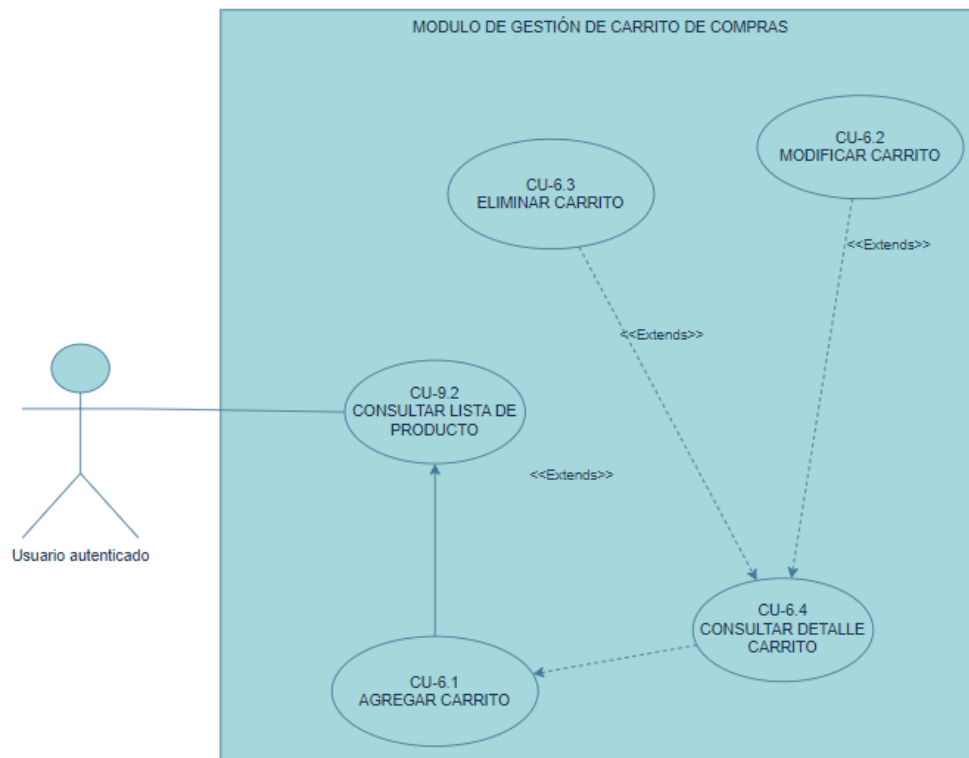
Módulo de gestión de producto



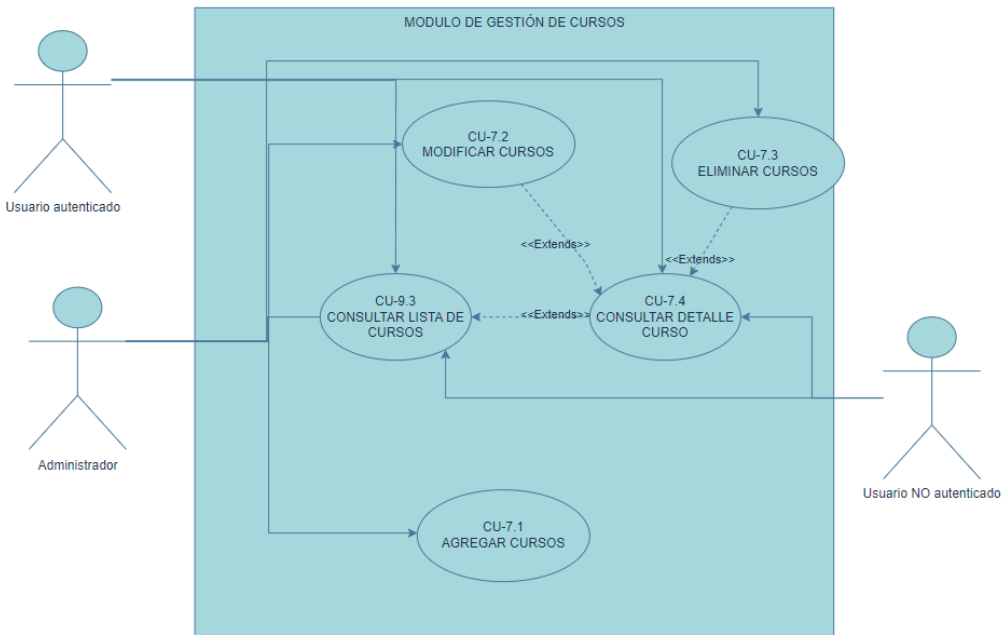
Módulo de gestión compra



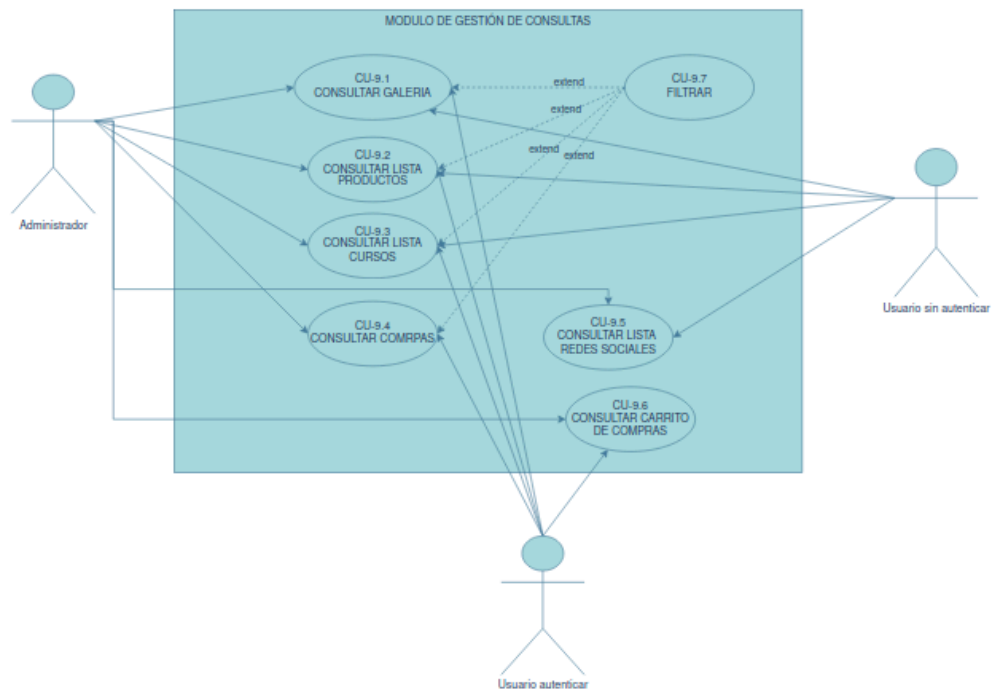
Módulo de gestión de carrito de compra



Módulo de gestión de cursos



Módulo de gestión de consultas



Descripción de casos de uso

Código	Nombre	Descripción caso de uso
CU-1.1	Registrar persona	Se desea registrar personas las cuales van a ser usuarias de la página.
CU-1.2	Editar persona	Se desea que los usuarios registrados puedan editar la información de ellos mismos.
CU-1.3	Eliminar persona	Se desea que los usuarios registrados puedan eliminar su cuenta en el momento que ya no deseen tener una cuenta.
CU-2.1	Iniciar sesión	Se desea que las personas puedan autenticarse para tener mayor control de ciertas acciones, las cuales se podrán hacer una vez autenticados.
CU-2.2	Cambiar contraseña	Se desea que las personas puedan cambiar la contraseña de su usuario.
CU-3.1	Agregar maquillaje	Se desea agregar un nuevo maquillaje, junto con la descripción, nombre, categorías, fecha y la foto.
CU-3.2	Editar maquillaje	Se desea editar la información de un maquillaje previamente creado.
CU-3.3	Eliminar maquillaje	Se desea poder eliminar maquillajes, los cuales ya no se deseen guardar.
CU-3.4	Consultar detalle de maquillaje	Se desea poder ver los detalles de un maquillaje en la galería.
CU-4.1	Ver detalle del producto	Se desea consultar el detalle de un producto en específico que se encuentra en la ventana que muestra todos los productos.
CU-4.2	Agregar producto	Permite al administrador registrar productos nuevos en el sistema.
CU-4.3	Editar producto	Se desea modificar la información de un producto.
CU-4.4	Eliminar producto	Se desea eliminar la información de un producto.
CU-5.1	Ver detalle de compra	Se desea consultar el detalle de una compra en específico que se encuentra en la ventana que muestra todas las compras.
CU-5.2	Enviar solicitud de compra	El usuario autenticado envía una solicitud de compra de uno o más productos.
CU-5.3	Agregar información de la compra	El usuario autenticado agrega información a una compra.
CU-5.4	Editar estado de la compra	El usuario administrador acepta o rechaza una compra realizada por un usuario autenticado y genera una notificación.

CU-6.1	Agregar carrito	Permite al usuario añadir al carrito todos los productos que seleccione previo a la compra.
CU-6.2	Editar carrito	Permite al usuario modificar los productos añadidos al carrito de compras previo a la compra.
CU-6.3	Eliminar carrito	Permite al usuario desechar todos los productos añadidos al carrito de compras previo a la compra.
CU-6.4	Consultar detalle del carrito	Permite al usuario consultar todos los datos pertinentes a su carrito y funciona como intermediario para realizar los cambios al mismo.
CU-7.1	Agregar cursos	Permite al administrador registrar cursos nuevos en el sistema.
CU-7.2	Editar cursos	Se desea modificar la información de un curso.
CU-7.3	Eliminar cursos	Se desea eliminar la información de un curso.
CU-7.4	Consultar detalle del curso	Se desea consultar el detalle de un curso en específico que se encuentra en la ventana que muestra todos los cursos.
CU-8.1	Consultar galería	Se desea poder ver todas las fotos de los maquillajes.
CU-8.2	Consultar lista de productos	Se desea poder ver todas las fotos de los productos disponibles.
CU-8.3	Consultar lista de cursos	Se desea poder observar los cursos que están registrados en el sistema.
CU-8.4	Consultar compra	Se desea poder ver todos los detalles de una compra.
CU-8.5	Consultar carrito de compras	Se desea poder consultar el detalle del carrito de compras.
CU-9.1	Ver detalle de la agenda	Se desea consultar el detalle de la agenda para cualquier tipo de vista, ya sea mensual, semanal o por día.
CU-9.2	Consultar la información de la agenda	Se desea poder ver toda la información de la agenda.
CU-9.3	Agregar citas a la agenda	Se desea poder agregar citas a la agenda.
CU-9.4	Editar citas de la agenda	Se desea poder editar las citas de la agenda según sea necesario.
CU-9.5	Eliminar citas de la agenda	Se desea poder editar productos del inventario.
CU-10.1	Ver detalle de inventario	Se desea consultar el detalle de un producto en específico que se encuentra en la ventana que muestra todos los productos que están en el inventario.
CU-10.2	Consultar lista de productos del inventario.	Se desea poder ver toda la información de los productos del inventario.
CU-10.3	Agregar productos al inventario	Se desea poder agregar productos al inventario.
CU-10.4	Editar productos del inventario	Se desea poder editar productos del inventario.
CU-10.5	Eliminar productos del inventario	Se desea poder eliminar productos del inventario.

Distribución de módulos

Módulo de gestión de acceso

Este módulo se encarga de administrar la autenticación y autorización de usuarios en la página web. Incluye funcionalidades como el registro, inicio de sesión, cierre de sesión y la gestión de roles y permisos.

Módulo de gestión de galería

Responsable de la administración de imágenes. Permite la carga, visualización y organización de contenido visual, como fotos de productos, cursos, maquillajes anteriores, entre otras.

Módulo de gestión de productos

Se encarga de todas las operaciones relacionadas con los productos de la tienda de maquillaje. Incluye la creación, edición, eliminación y visualización de información detallada sobre los productos.

Módulo de gestión de compras

Administra las transacciones de compra realizadas por los usuarios. Registra los detalles de cada compra, entre en un proceso de aceptación de pago y se actualiza el estado del pedido.

Módulo de gestión de carrito de compras

Permite a los usuarios agregar, modificar y eliminar productos de su carrito de compras. Facilita la gestión de las selecciones de productos antes de la compra final.

Módulo de gestión de cursos

Administra la oferta de cursos de maquillaje. Permite la creación, modificación y eliminación de cursos.

Módulo de gestión de gestión de consultas

Maneja las consultas de los usuarios. Para poder visualizar la mayoría de los elementos en la página web dependiendo del tipo de usuario que sea.

Módulo de gestión de agenda

Permite la programación y gestión de citas de maquillaje. Ayuda a organizar la agenda con la maquillista y a coordinar citas con los clientes.

Módulo de gestión de inventario

Administra el inventario de productos de maquillaje. Realiza un seguimiento en tiempo real de las existencias, notifica sobre agotamientos y se actualizaría la disponibilidad después de cada compra.

Manejo de Excepciones y Errores

Se quiere mantener un ambiente seguro para todos los clientes en general, por lo que se requiere un manejo óptimo y adecuado de excepciones y errores dentro de la página web. Con lo cual garantizamos una respuesta controlada a situaciones inesperadas y ayuda a mejorar la calidad y estabilidad del sistema, además de aportar más seguridad a los clientes.

Como estrategia de manejo de errores se requiere realizar las siguientes operaciones:

1. Registro de errores

Se implementará un sistema de registro de errores para rastrear y documentar eventos inesperados.

Además de utilizar herramientas de registro integradas del mismo NodeJS y Firebase para almacenar la información detallada sobre las excepciones generadas a lo largo de la producción.

2. Establecer una jerarquía de errores

Se requiere establecer una jerarquía de excepciones la cual se acople a nuestras necesidades para clasificar y gestionar los diferentes tipos de errores de manera coherente.

También la definición de códigos de error y mensajes descriptivos para facilitar la identificación de resolución de problemas.

Para un mejor desarrollo y conclusión de la pagina web se realizarán pruebas unitarias validando cada uno de los módulos, así como los casos de uso que se verán en la página web antes de ser lanzada a producción.

Ejemplo del manejo de excepciones en NodeJS

```
3   export class AdminProduct {
4     createProduct(name, brand, desc, img, price, state, tags) {
5       try {
6         return SingletonDAO.createProduct(name, brand, desc, img, price, state, tags);
7       } catch(error) {
8         console.error("Error al crear el Product:", error);
9       }
10    }
11  }
```

Y un ejemplo del manejo de excepciones a la hora de cargar datos a la base de datos de Firebase.

```
287 // Uploading the data of the Product to firestore
288 const productCollection = collection(SingletonDAO.instance.db, "Product");
289 const docRef = await addDoc(productCollection, {
290   name: name,
291   brand: brand,
292   img: downloadURL,
293   desc: desc,
294   price: price,
295   state: state,
296   tags: tagRefs.filter((tagRef) => tagRef !== null)
297 });
298 console.log("Product creado exitosamente.");
299
300 // Returning Id for reference
301 + return docRef.id
302 } catch (error) {
303   console.error("Error al crear el product:", error);
304   return null
305 }
306 }
307 }
```