



MANUAL TÉCNICO

GESTIÓN EMPRESARIAL DE INVENTARIO

IDEA PUBLICIDAD - EL SALVADOR

Desarrollo de Software

Primer año de bachillerato

Proyecto Técnico Científico

Hecho por:

- Luis Edgardo Soriano Rivas
- Omar Ernesto Rivera Hernández
- Rosenberg Astul Ramos
- Fabiola Nicole Henríquez Portillo
- Chelsea Mariana Mejía Martínez

Índice

| | |
|--|-----------|
| 1.Introducción..... | 3 |
| 2. Tecnologías y herramientas..... | 5 |
| 3. Estructura de la base de datos..... | 8 |
| 4. Diccionario de datos..... | 17 |
| 5. Arquitectura de Software..... | 18 |
| 6. Estructura del proyecto..... | 19 |
| 7. Diseño de la aplicación..... | 25 |
| 8. Buenas prácticas de desarrollo..... | 29 |
| 9. Requerimientos de hardware y software..... | 32 |
| 10. Instalación y configuración..... | 33 |

1.Introducción

El presente manual técnico está destinado a desarrolladores, administradores de sistemas y equipos de soporte involucrados en la instalación, configuración y mantenimiento del proyecto Gestión Empresarial de Inventario IPES. Su propósito es ofrecer una guía eficiente sobre la arquitectura, tecnologías, herramientas y buenas prácticas aplicadas en el desarrollo del sistema, garantizando una administración segura.

El manual está organizado en las siguientes secciones:

- El presente manual técnico está destinado a desarrolladores, administradores de sistemas y equipos de soporte involucrados en la instalación, configuración y mantenimiento del proyecto Gestión Empresarial de Inventario IPES. Su propósito es ofrecer una guía eficiente sobre la arquitectura, tecnologías, herramientas y buenas prácticas aplicadas en el desarrollo del sistema, garantizando una administración segura.
- El manual está organizado en las siguientes secciones:
- Tecnologías y herramientas empleadas: Descripción de todas las tecnologías, librerías y herramientas utilizadas en el frontend y el backend, indicando los enlaces a su documentación oficial.
- Presentación del modelo relacional, incluyendo un diagrama visual y el script SQL completo de la estructura.
- Detalle de cada tabla y sus columnas, especificando campos, tipos de datos, anulabilidad, valores predeterminados y una breve explicación funcional.
- Arquitectura del software: Descripción de la arquitectura (pendiente)
- Estructura del proyecto: Organización sobre archivos del proyecto, con una guía para navegar y comprender las distintas partes del código fuente.
- Diseño de la aplicación: Presentación de los lineamientos visuales de la interfaz del usuario, incluyendo paleta de colores, tipografía, uso de imágenes en el diseño y consideraciones de adaptabilidad y responsividad..
- Buenas prácticas de desarrollo: Definición de los estándares de programación empleados en C# y SQL Server, así como las convenciones para comentarios y documentación.
- Requerimientos de hardware y software: Especificaciones mínimas y recomendadas de hardware y software para ejecutar el sistema, en entornos locales.

- **Instalación y configuración:** Instrucciones paso a paso para la marcha del sistema en un servidor local.

Este manual técnico está pensado como una guía completa y precisa para comprender todos los aspectos técnicos del proyecto Gestión Empresarial de Inventario IPES, comprometiendo su desarrollo, mantenimiento y futuras actualizaciones más eficientes.

Estándares de programación en SQL Server

Para el desarrollo en SQL Server, se definieron reglas que aseguran coherencia, rendimiento y seguridad en la base de datos.

Convenciones de nomenclatura

-Tablas: singular con prefijo funcional (ej. Talla_Ropa, Categoria_Ropa).

Definición de tablas

- Usar tipos de datos adecuados y evitar nvarchar(MAX) innecesarios.
- Definir claves primarias explícitas (PRIMARY KEY).
- Incluir valores predeterminados cuando aplique (DEFAULT GETDATE() para fechas).
- Usar NOT NULL para campos obligatorios cuando sea posible.

Procedimientos y consultas

- Evitar SELECT *Innecesarios, Siempre usarlo cuando es requerido:
- Uso de Identificadores para la llaves foráneas

2. Tecnologías y herramientas

Frontend

HTML5 y CSS3

Uso: HTML5 para la maquetación y estructuración de la página web, junto con CSS3 para definir estilos y diseño.

Sitio web oficial: <https://developer.mozilla.org/es/docs/Web>

Backend

C# (.NET Framework)

Sitio web oficial: <https://dotnet.microsoft.com/>

SQL Server

Sitio web oficial: <https://learn.microsoft.com/sql>

C# (.NET FRAMEWORK) y SQL Server

Librerías

- Formulario Transparente:

[Formulario Transparente](#)

- Botones Redondeados

[Botones Redondeados](#)

- Redimensión con Formularios sin Bordes:

[Enlace al repositorio de la Librería](#)

- Botones Redondeados Personalizados:
[Enlace a los Botones Personalizados](#)
- Degradado de Botones:
[Degradado de Botones](#)
- Hora actualizable con un label en Windows Forms:
[Hora Actualizable](#)
[Hora Actualizable Microsoft](#)
- Efecto Hover del Botón
[Hover de los Botones](#)
- Creacion de Control de Usuario UC
[Control de Usuario](#)
- Optimización de Windows FORMS
[Renderización de Windows Forms](#)
[Doble Buffer](#)
[Optimización](#)
- Formulario Padre e Hijo: (Video de Youtube)
[Formulario Padre e Hijo](#)
- Pantalla de Carga (Video de Youtube)
[Pantalla de Carga](#)
- Formato de Hora y Fecha:
[Formato de Hora y Fecha](#)
- Efecto Hover del Botón
[Color al hacer hover](#)
- Cuadro de Mensaje Personalizado
[Mensajes Personalizados](#)
[Cuadro de Dialogo Personalizado](#)
- Botón Redondeado
[Botones Personalizados](#)
- Datagridview Personalizado
[Datagridview Personalizado](#)

- Animaciones Fade In Fade Out
[Algoritmo para animar](#)
- WX_Composited
[wx_Composited](#)

[WX_Composited](#)
- Fade in para el User Control
[Fade In Para User Control](#)
- Dibujar los bordes en cualquier control
[Dibujar Bordes en Controles](#)

[Dibujar Bordes en Paneles](#)
- Double Buffer Avanzado
[Aplicar doble buffer en controles](#)

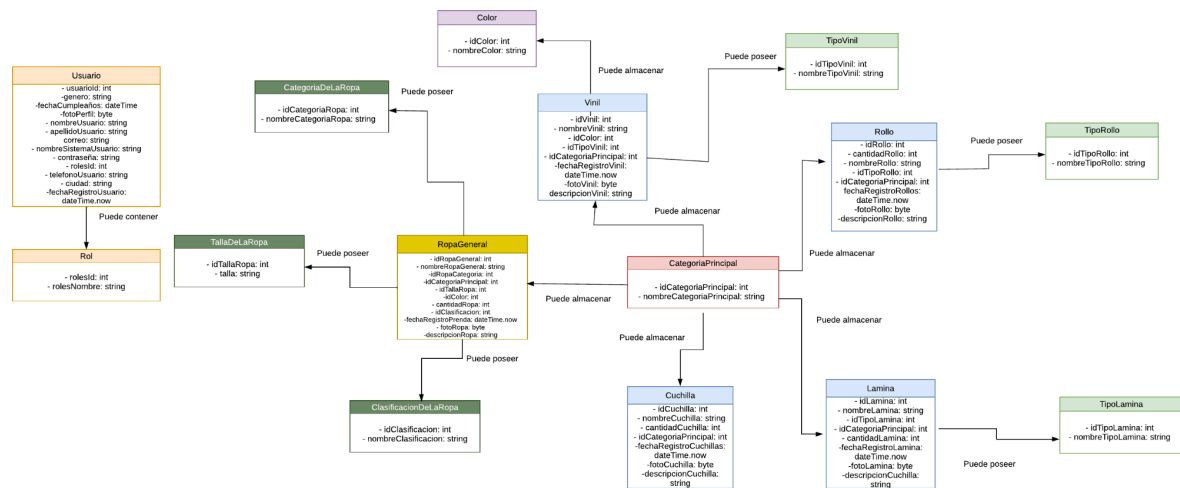
[Reducción de Parpadeo con doble buffer](#)
- Collate para arreglar Inicio de Sesión:

[Uso de COLLATE para Inicio de Sesión](#)

3. Estructura de la base de datos

Modelo relacional

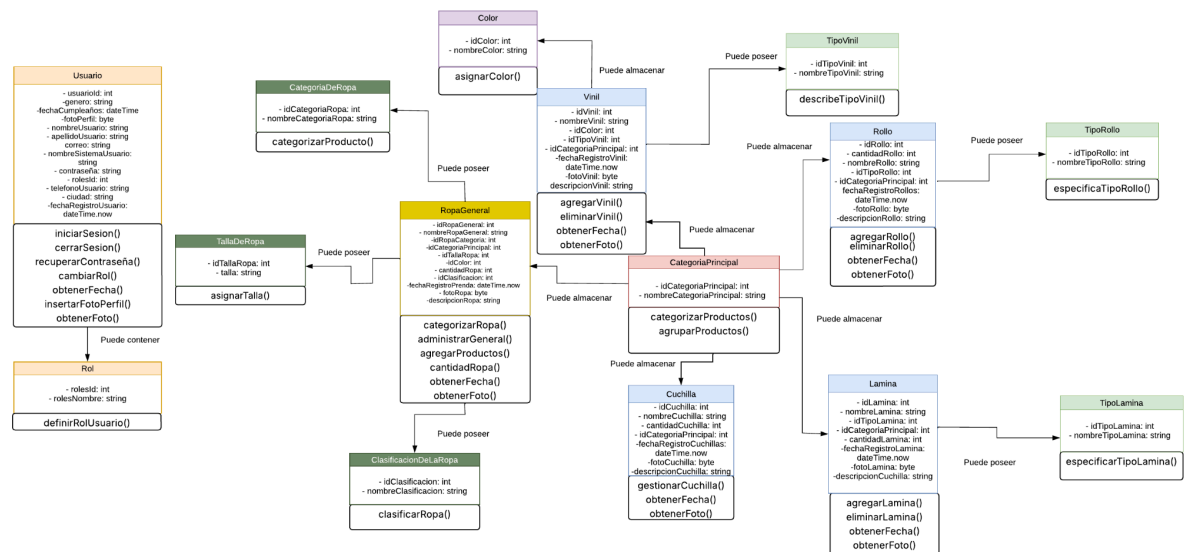
Diagrama de dominio de la base de datos del proyecto PTC.



Nota: Diagrama, mejor calidad en el siguiente enlace

<https://drive.google.com/file/d/15mlFic6K-wz315PANCuSkdsjVO-y65fe/view?usp=sharing>

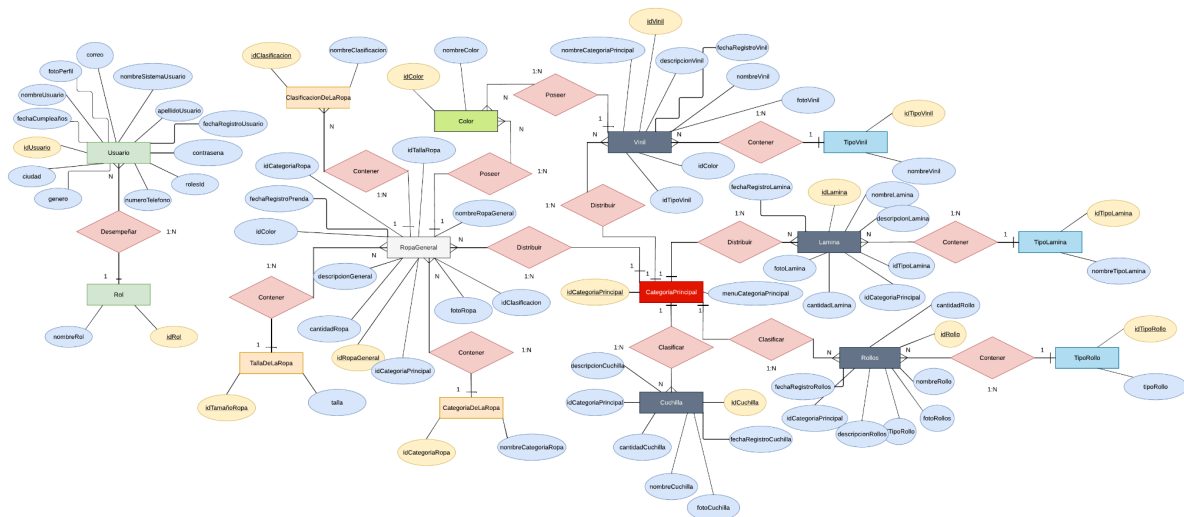
Diagrama de Clases de la base de datos del proyecto PTC.



-Nota: Diagrama, mejor calidad en el siguiente enlace

<https://drive.google.com/file/d/19oqGdR3dDRk449LwVDYpt6TfOqWzRBYx/view?usp=sharing>

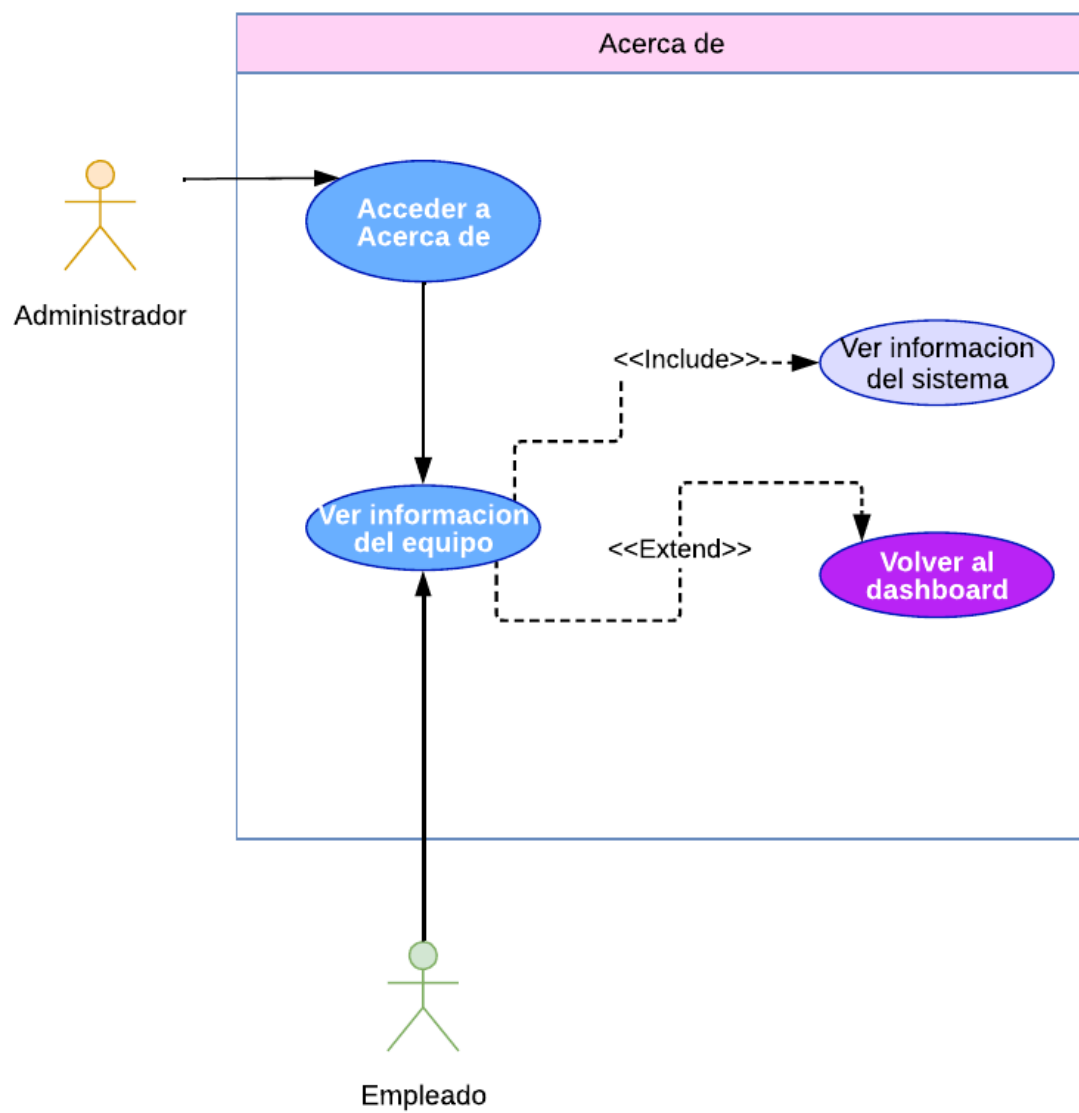
Diagrama de Entidad Relación de la base de datos del proyecto PTC.

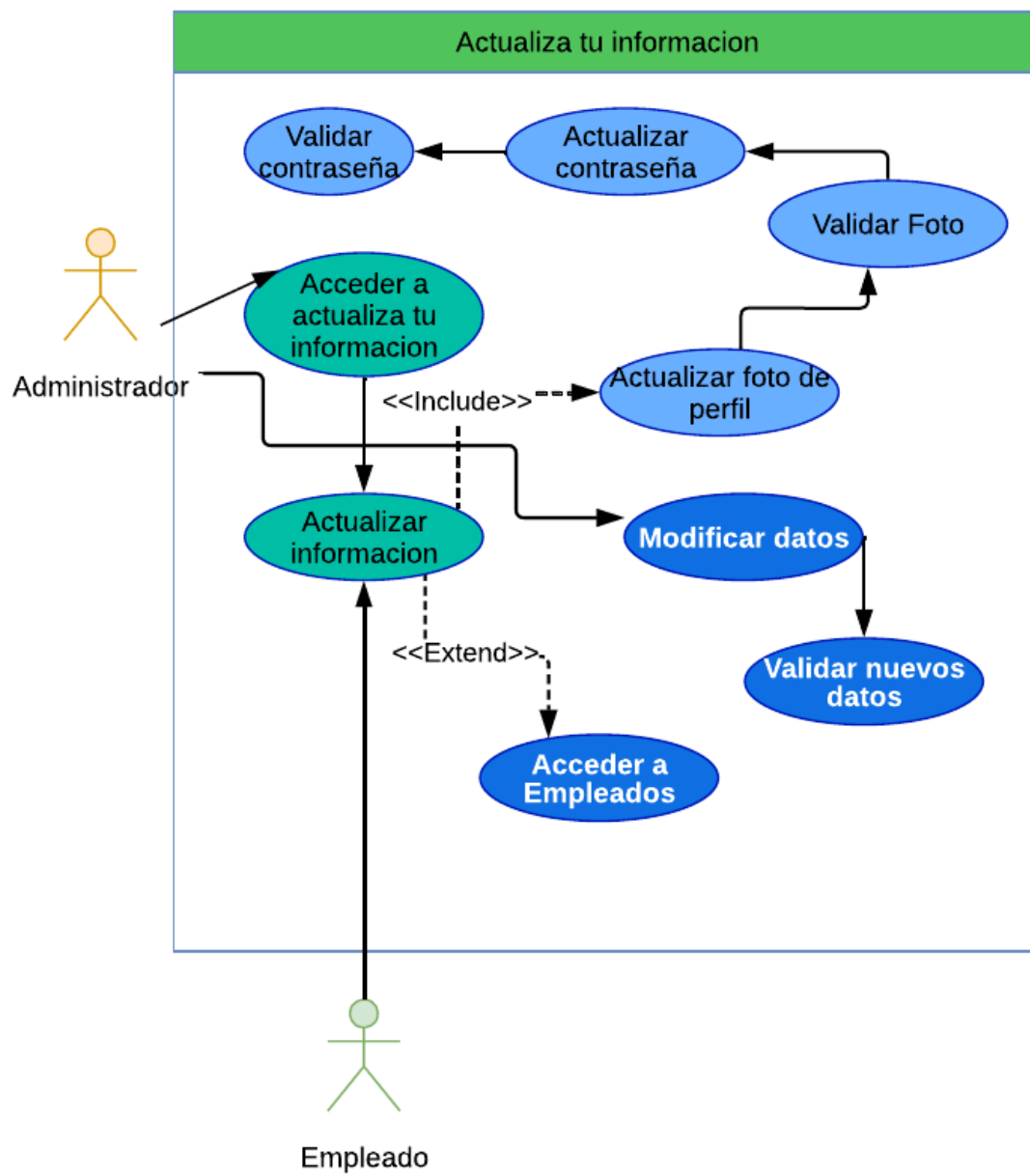


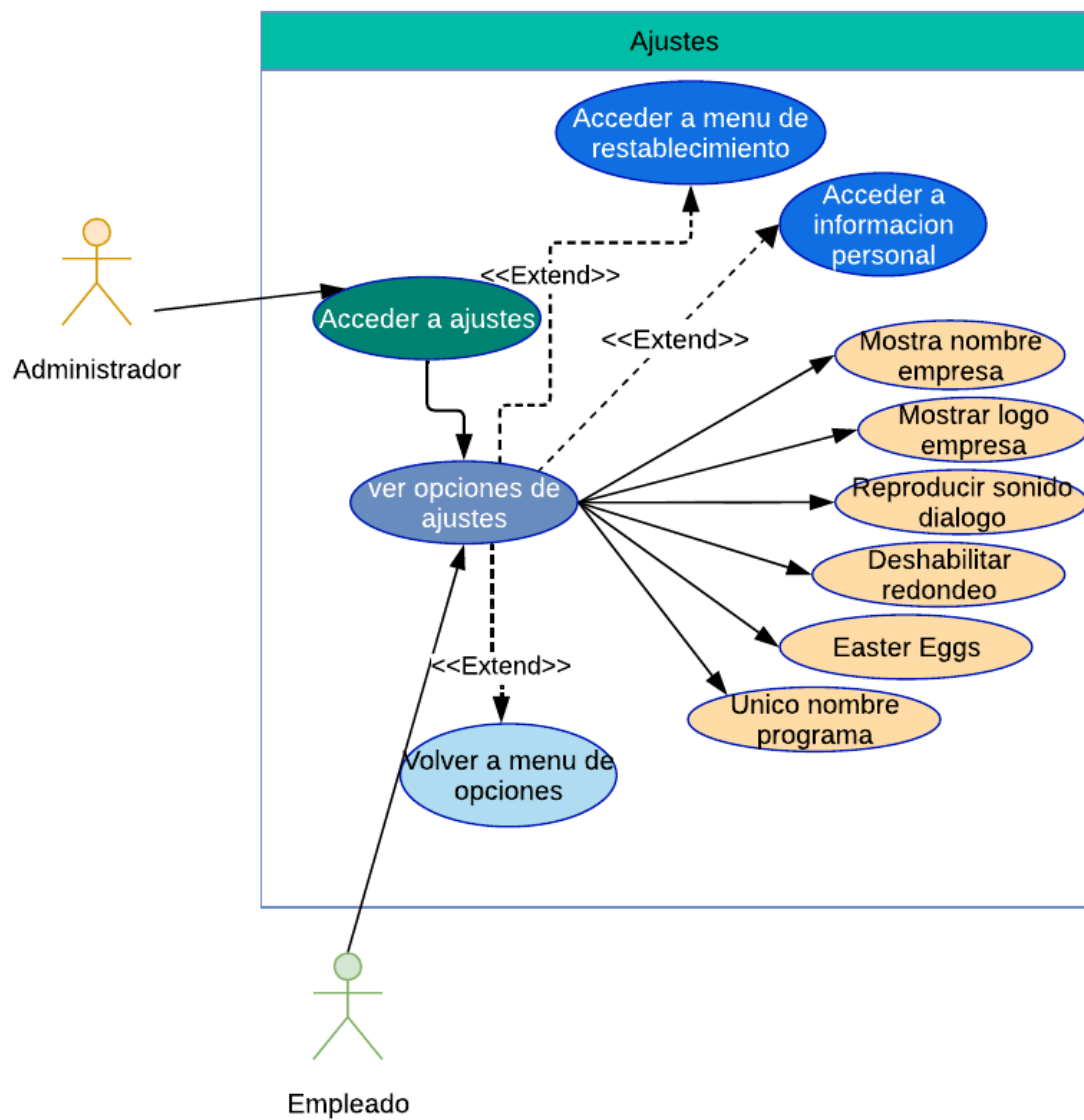
Nota: Diagrama, mejor calidad en el siguiente enlace

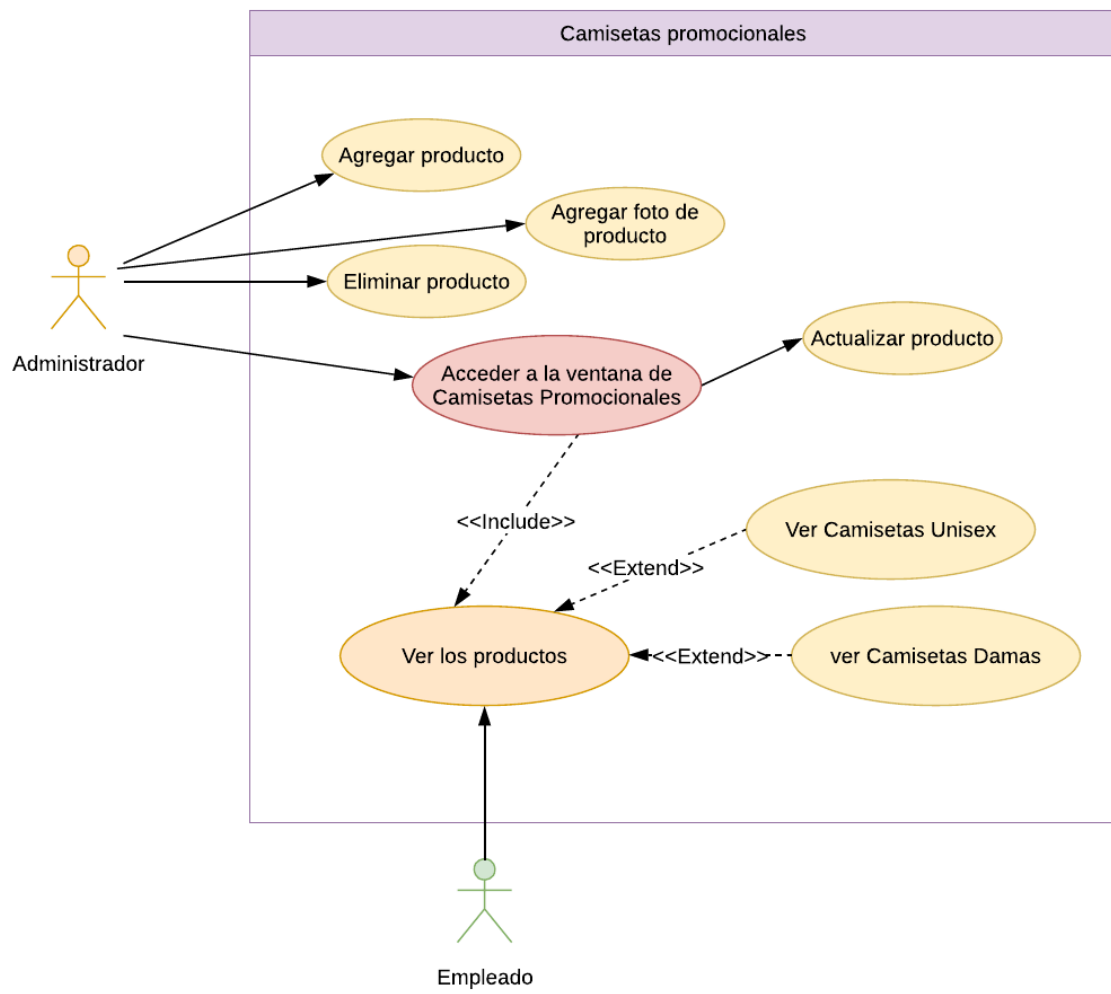
<https://drive.google.com/file/d/1XY4Nd6xBR5ZhObwN94lFFWSpqBO6u8XP/view?usp=sharing>

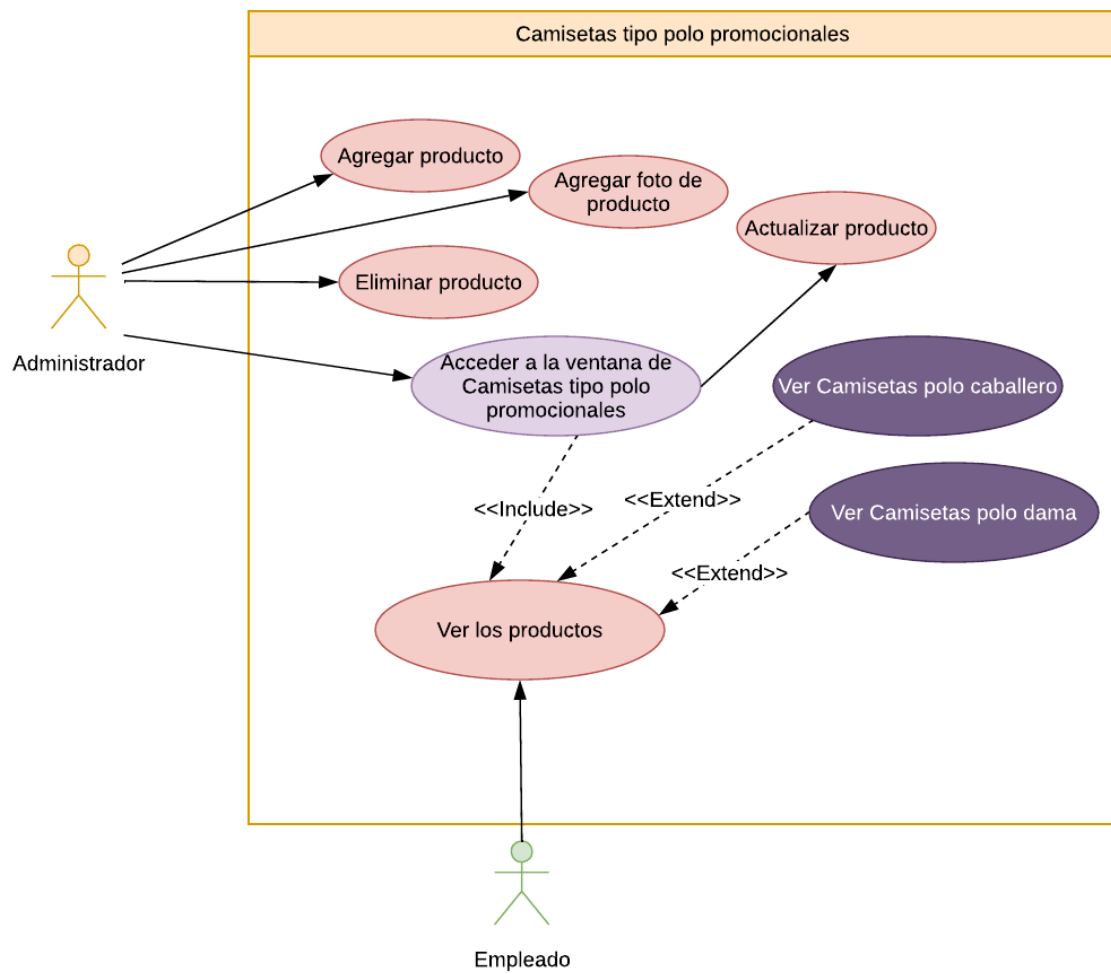
Diagrama de Caso de Uso de la base de datos del proyecto PTC.

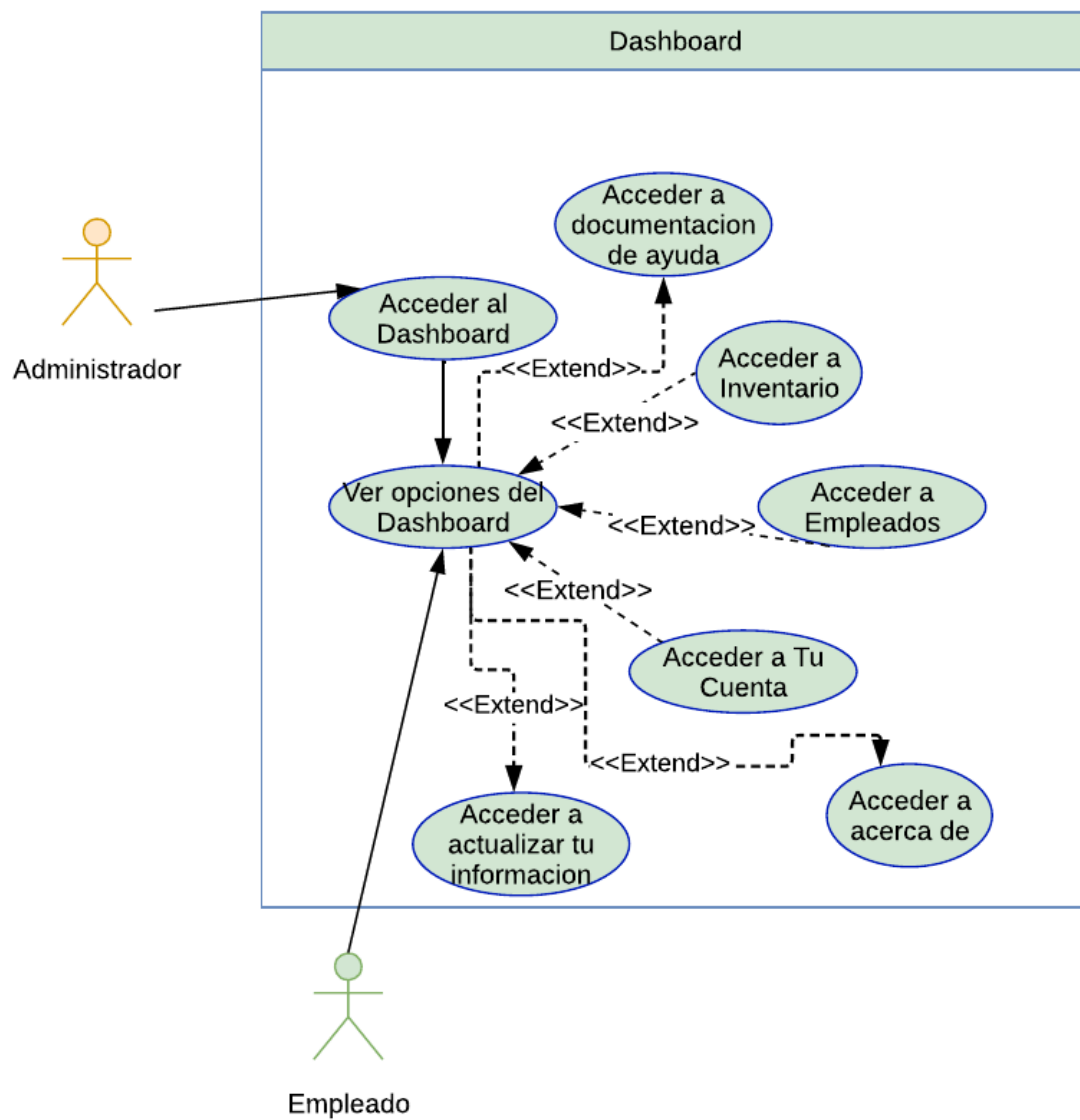


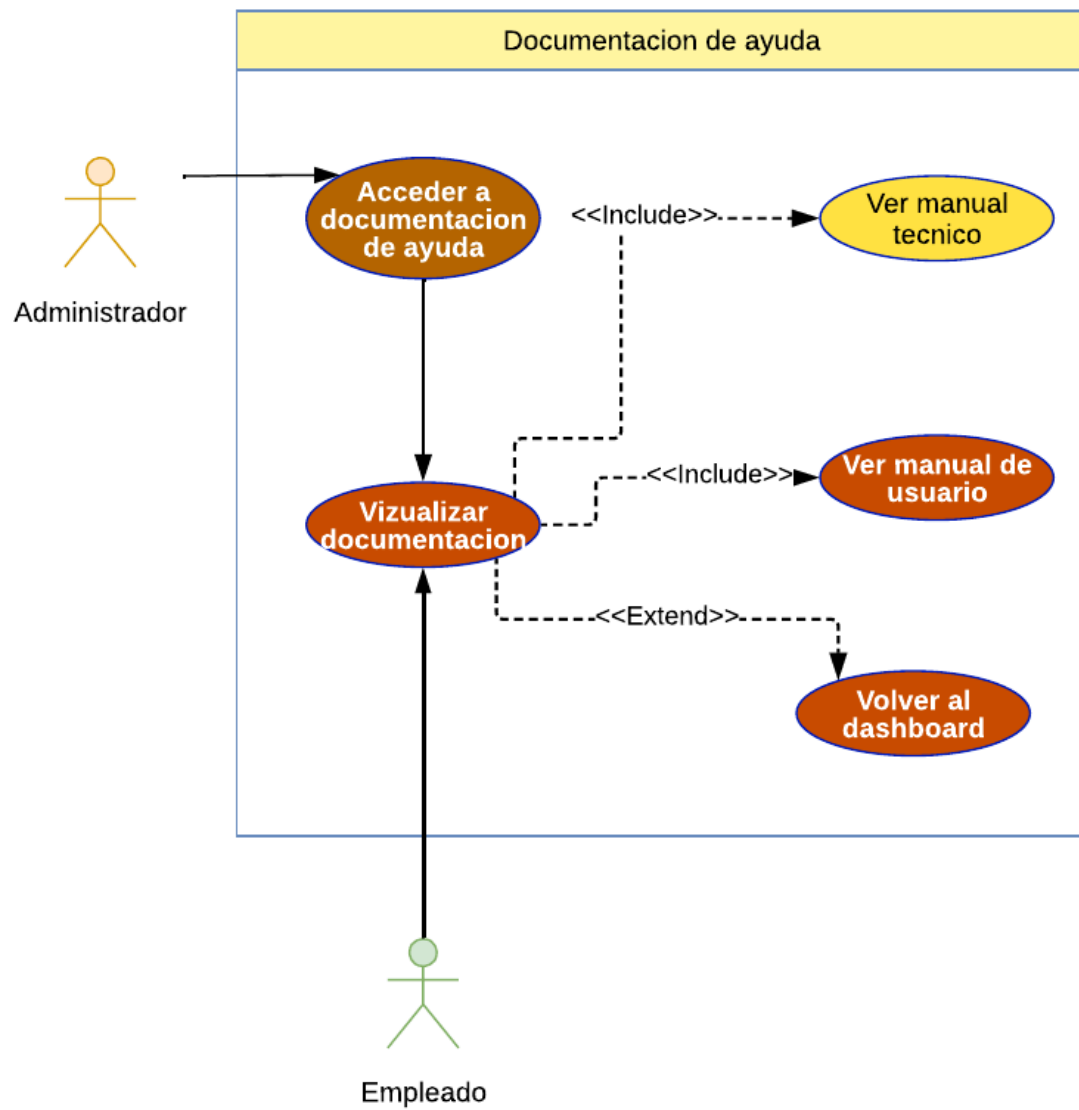


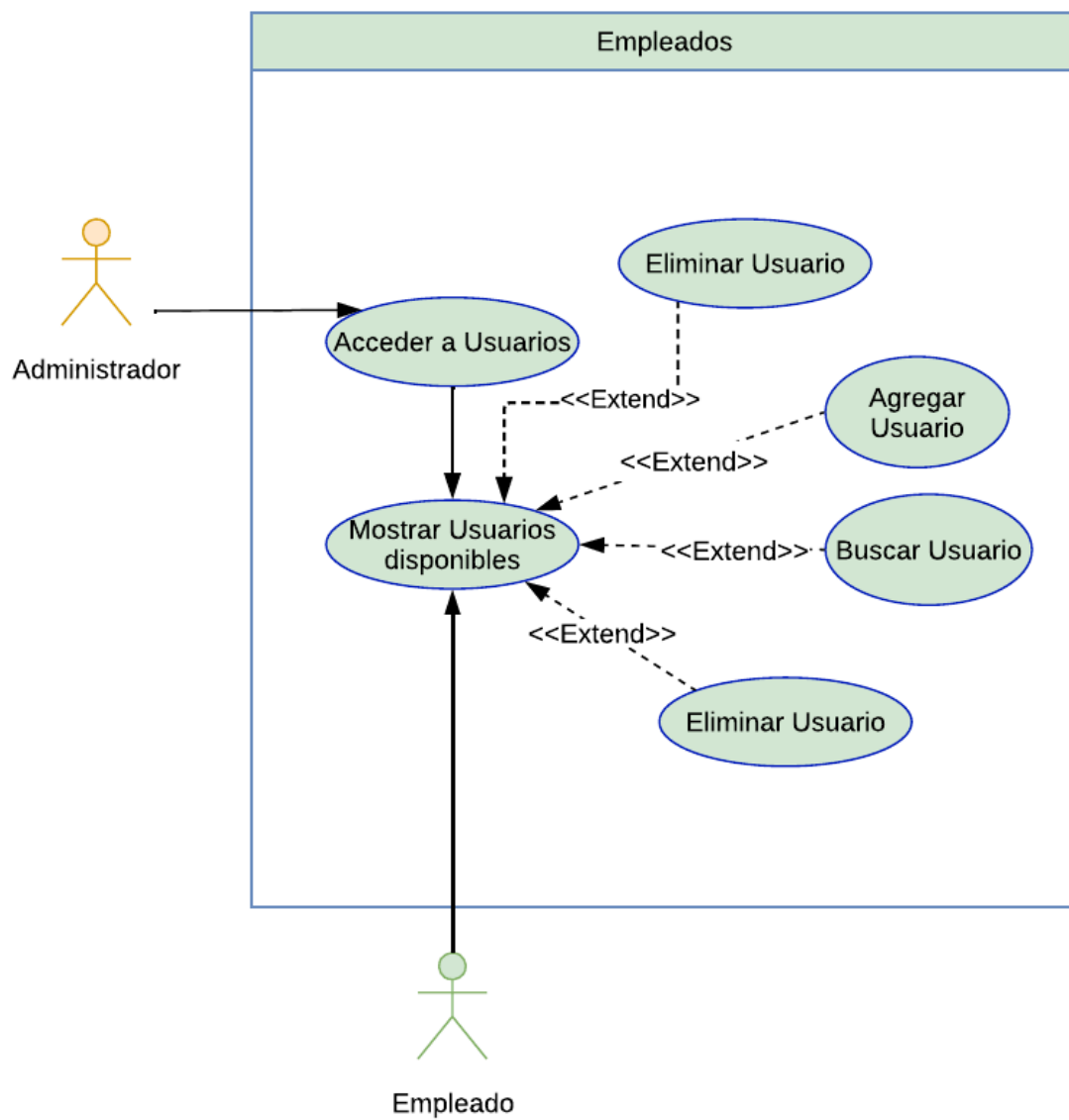


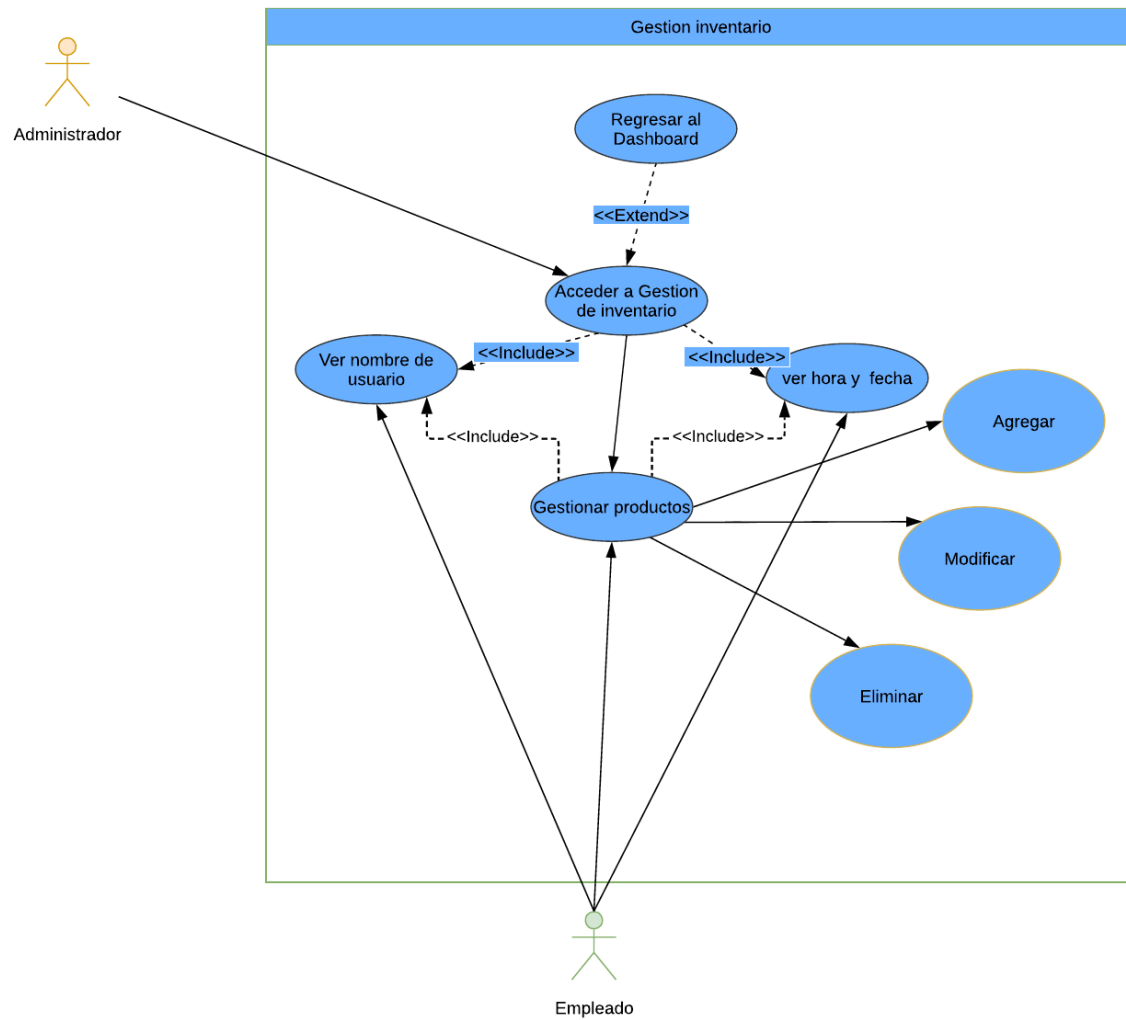


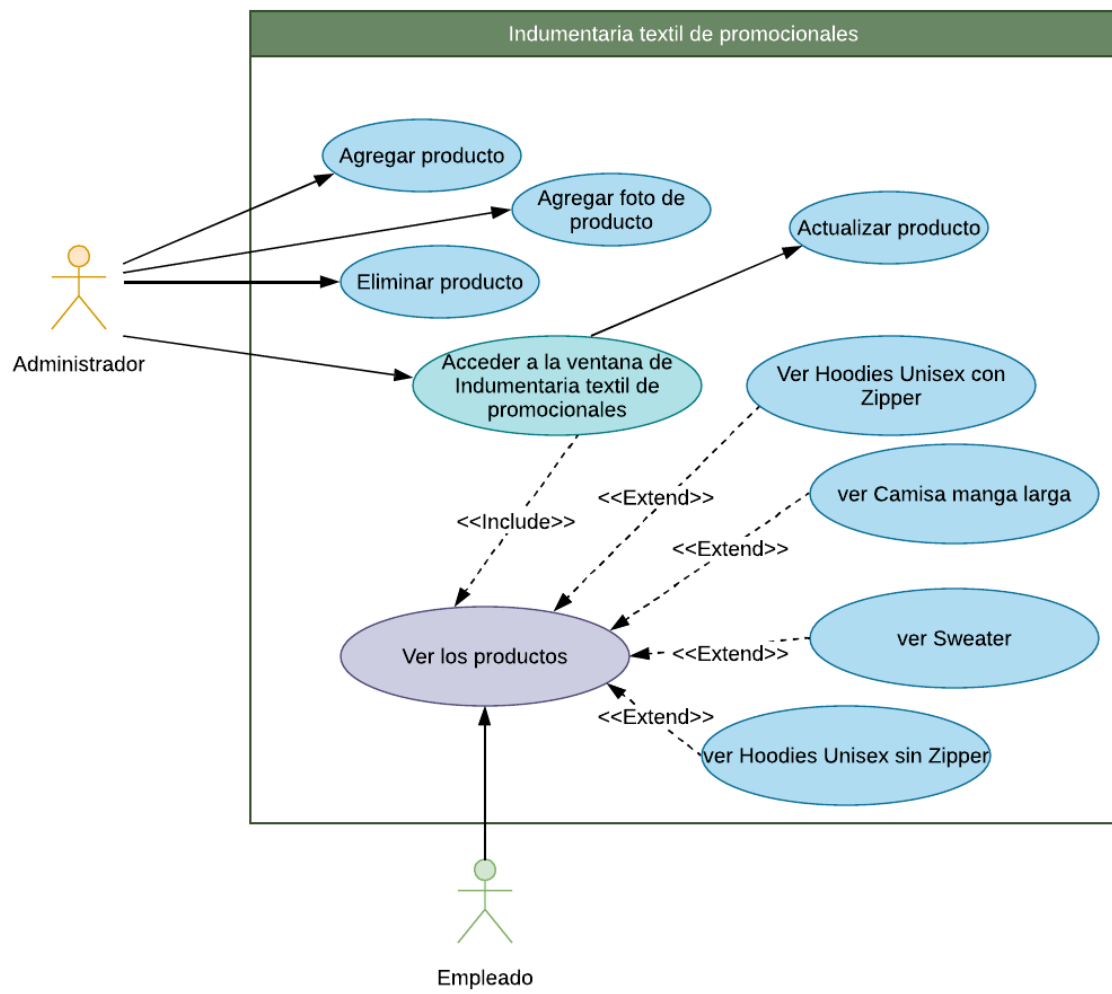


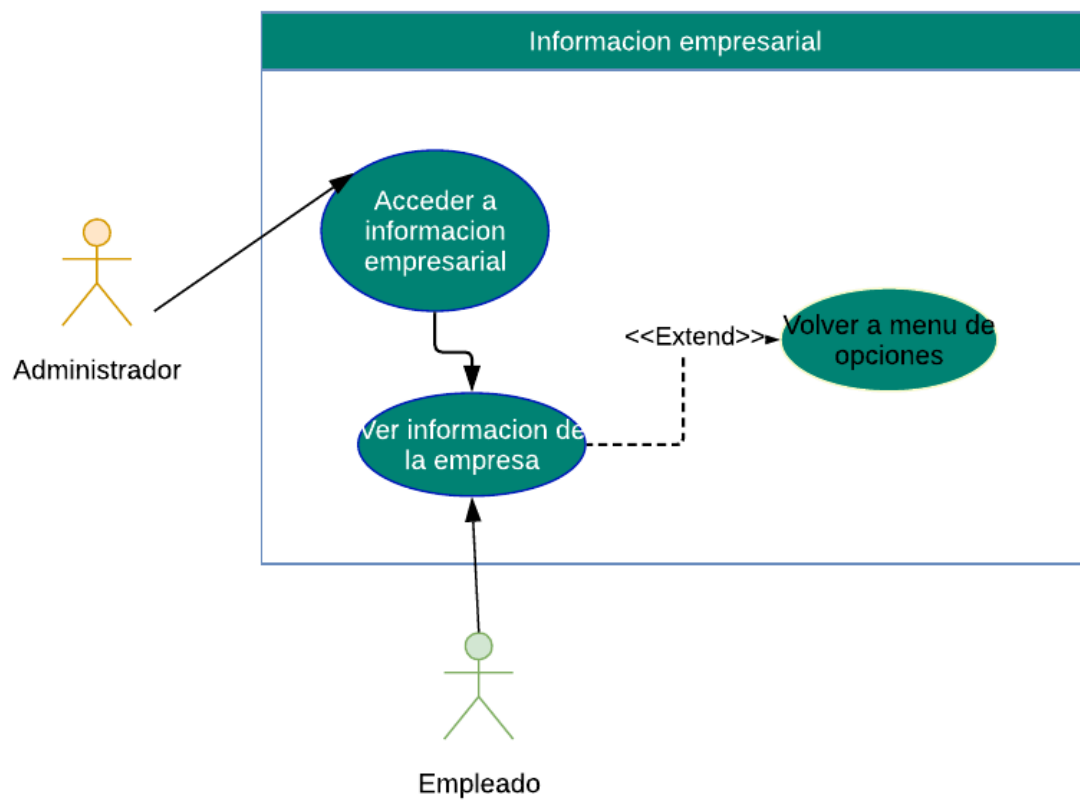


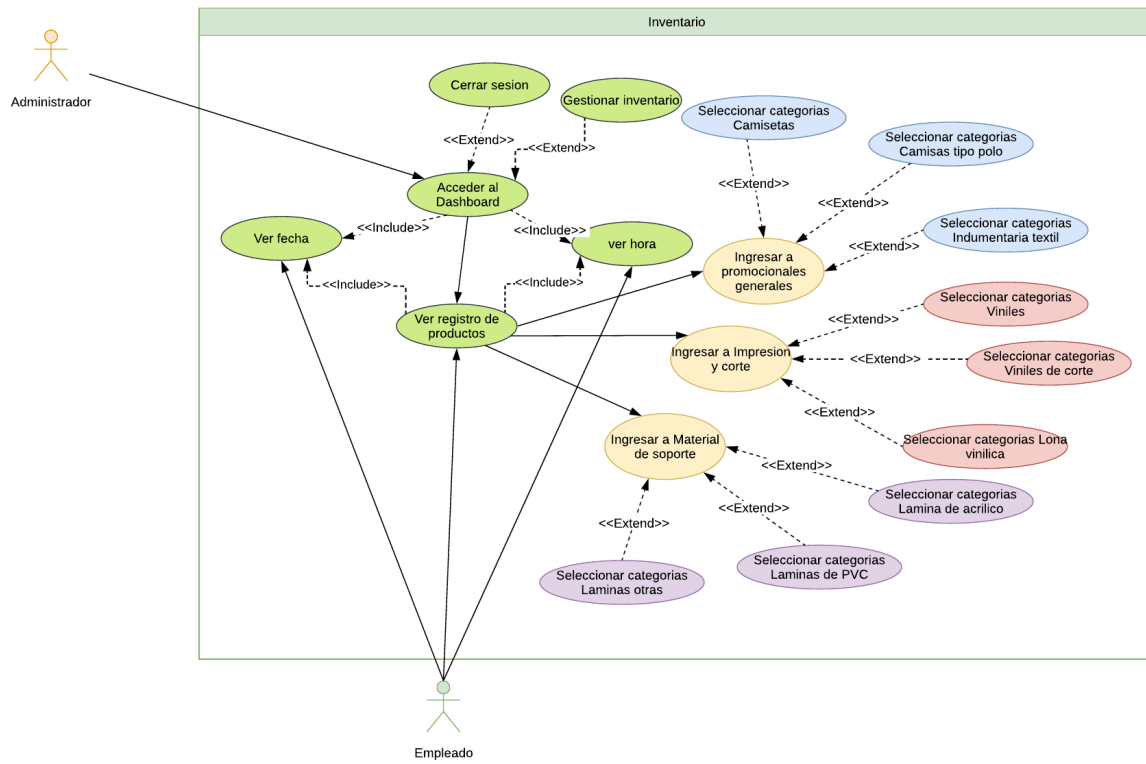


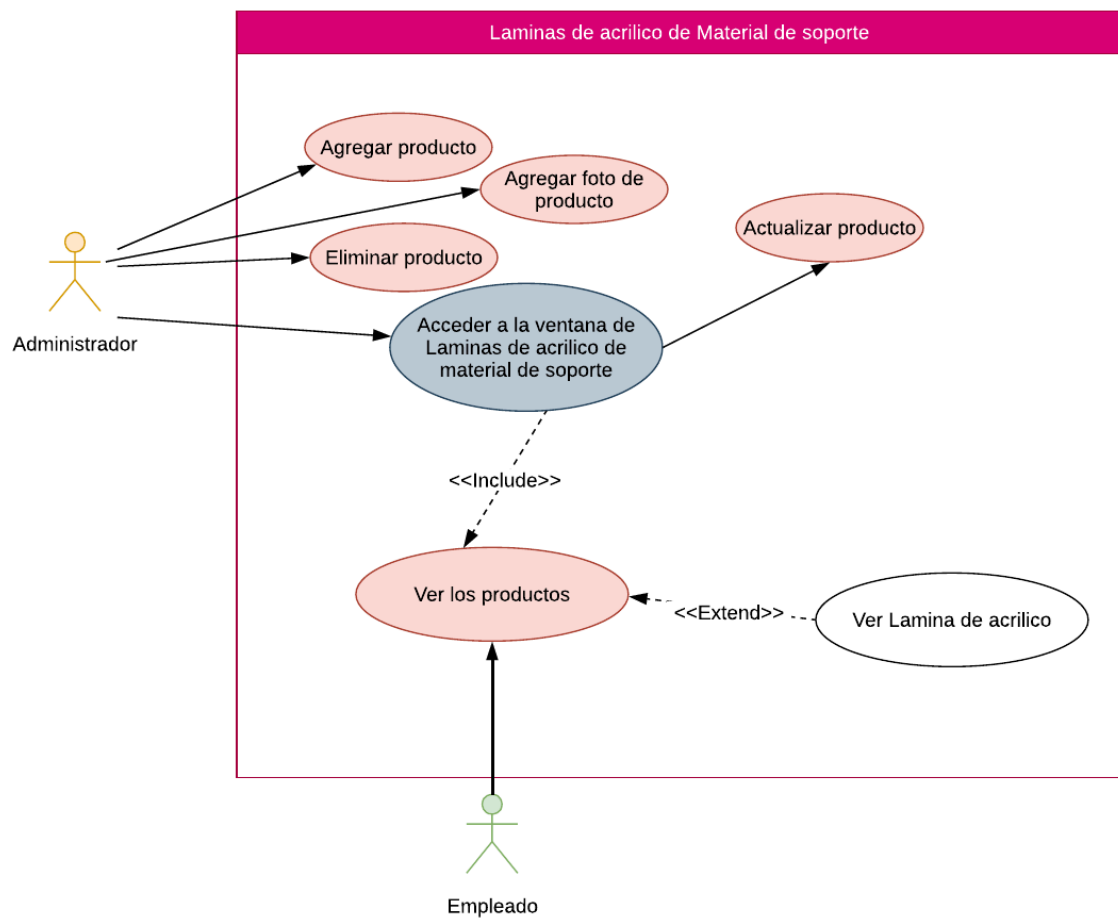


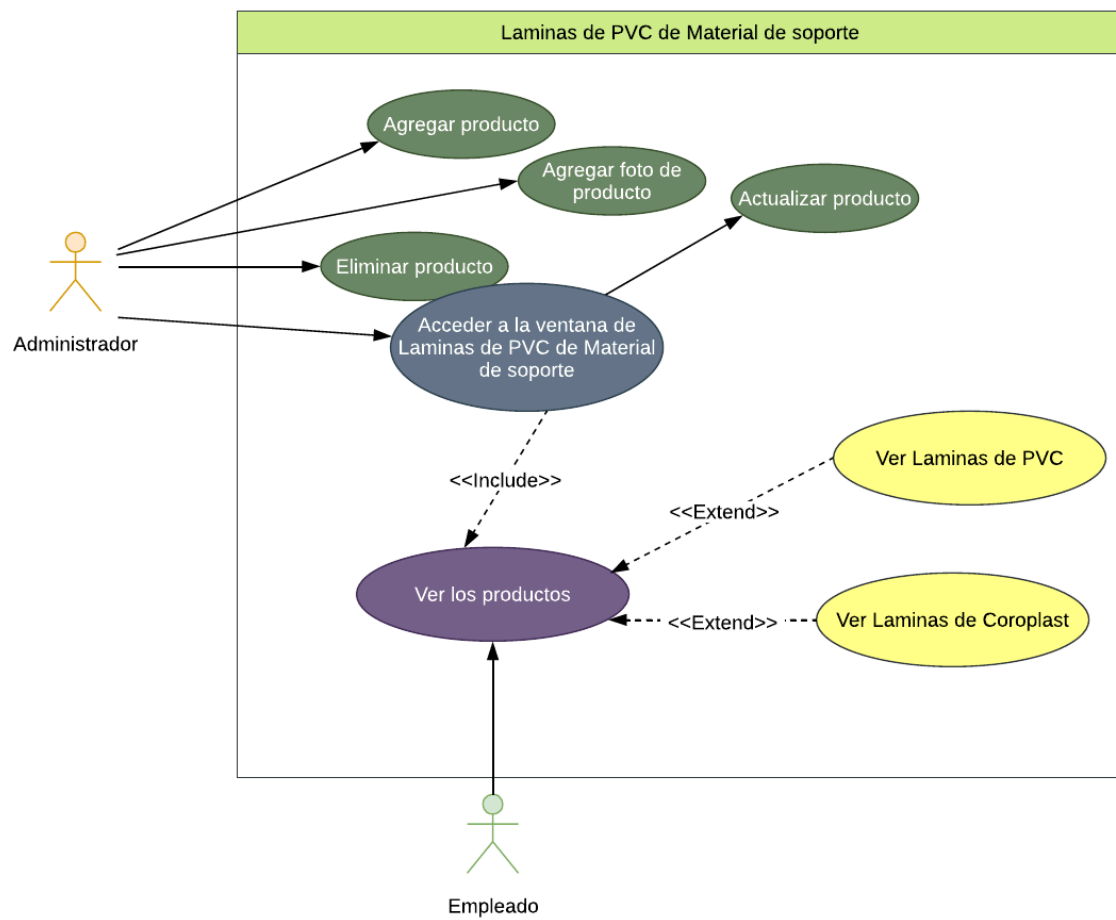


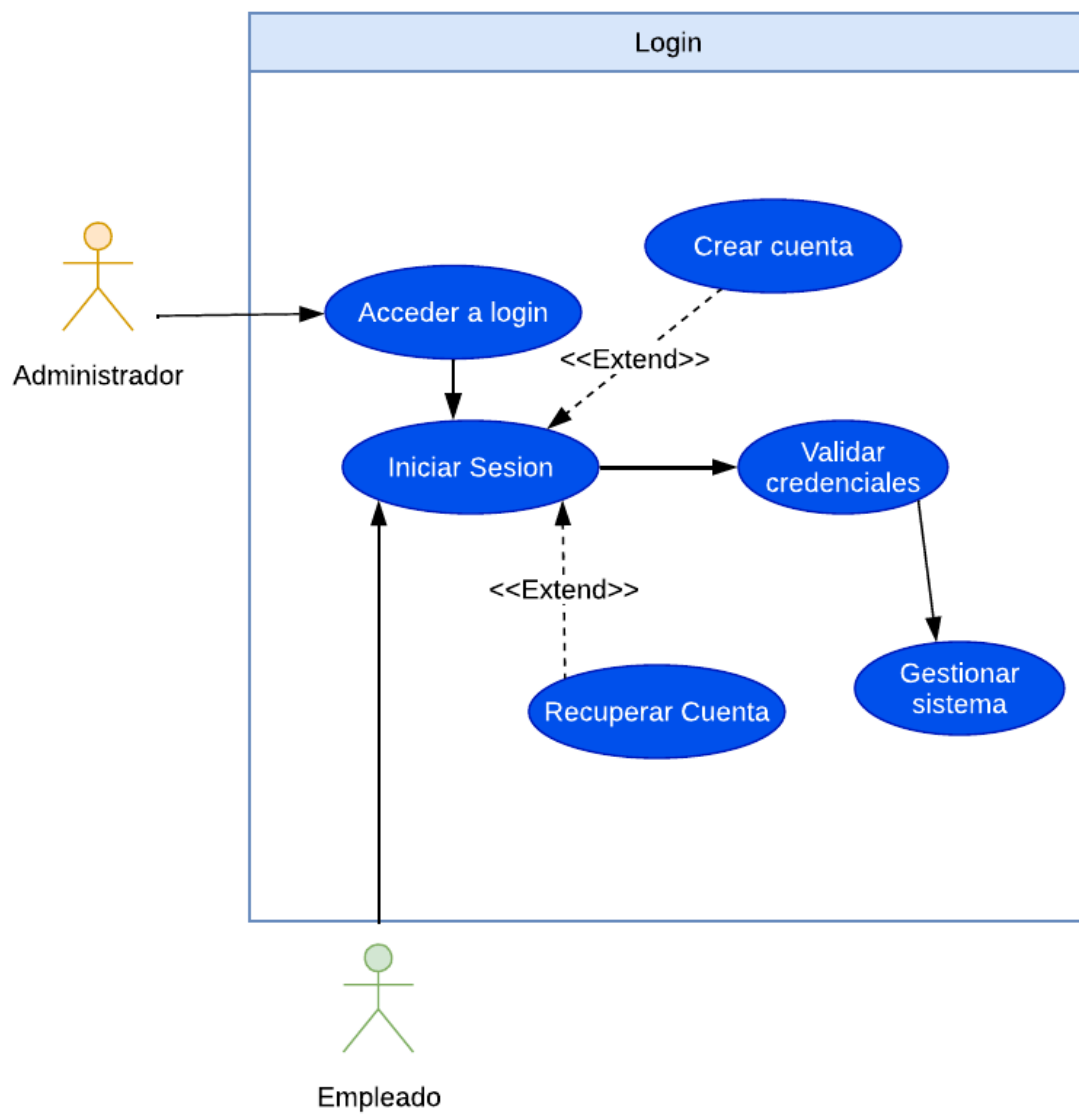


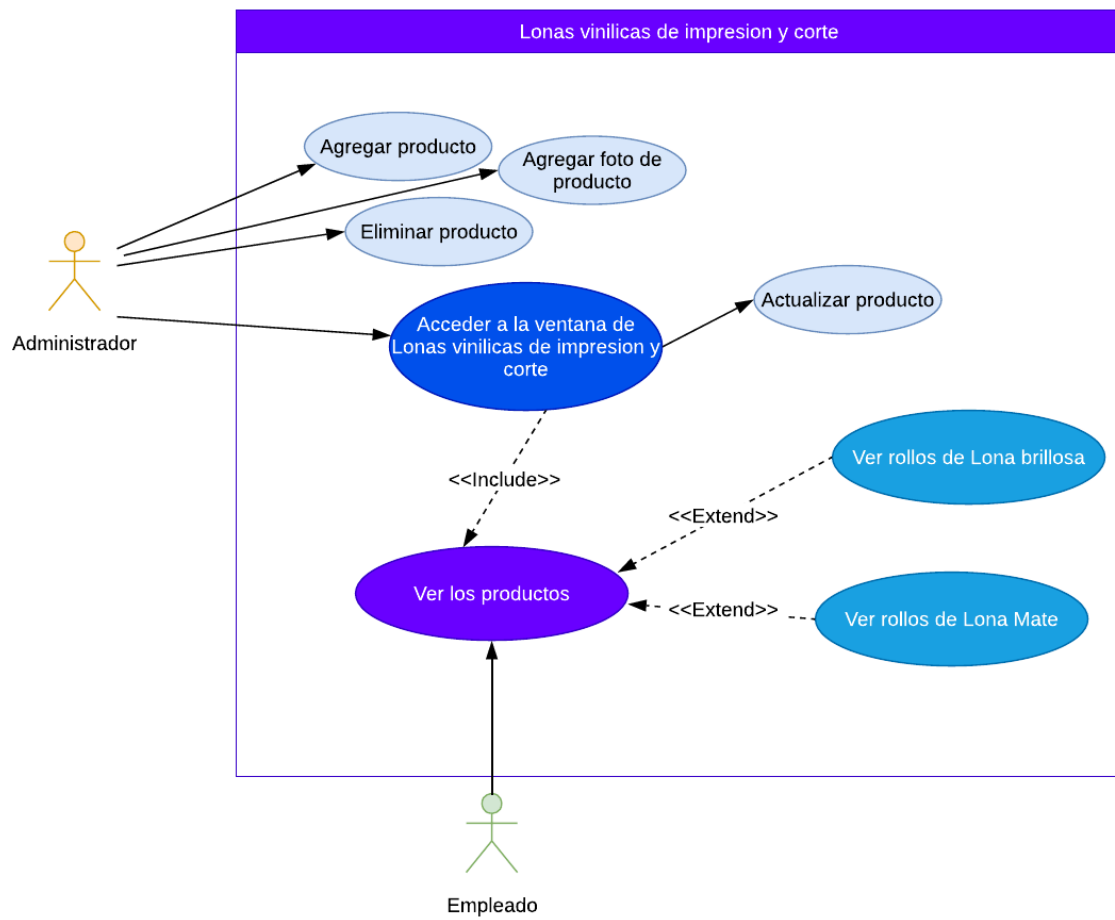


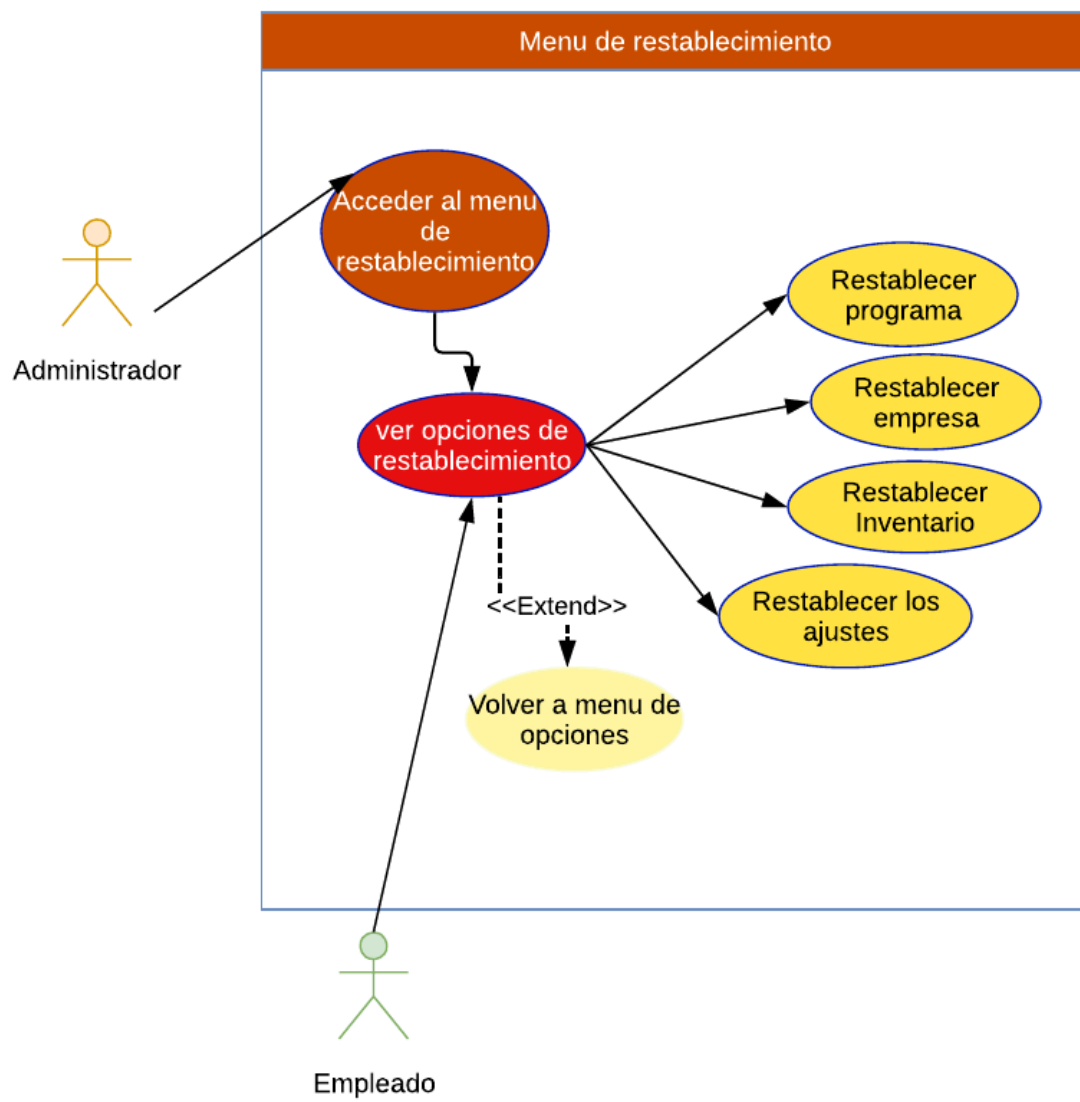


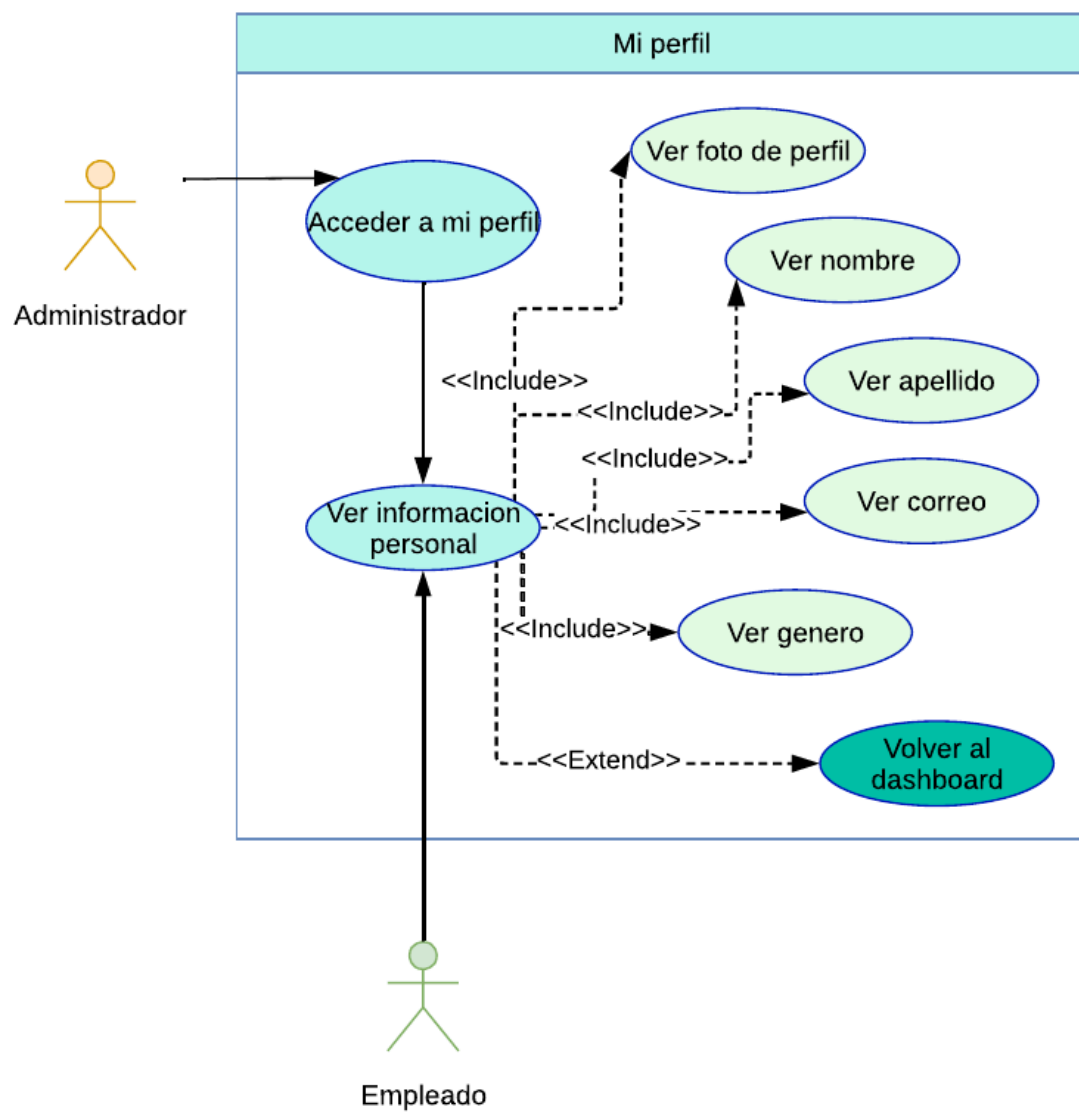


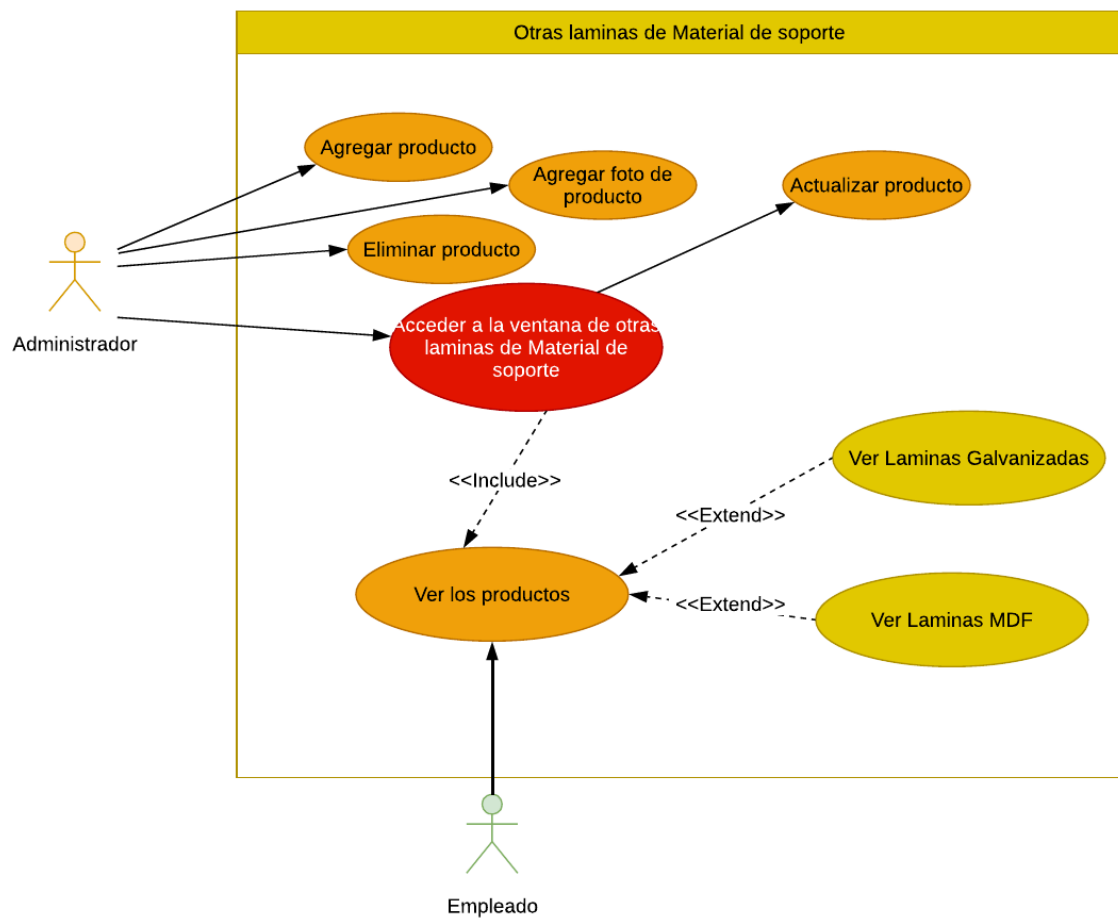


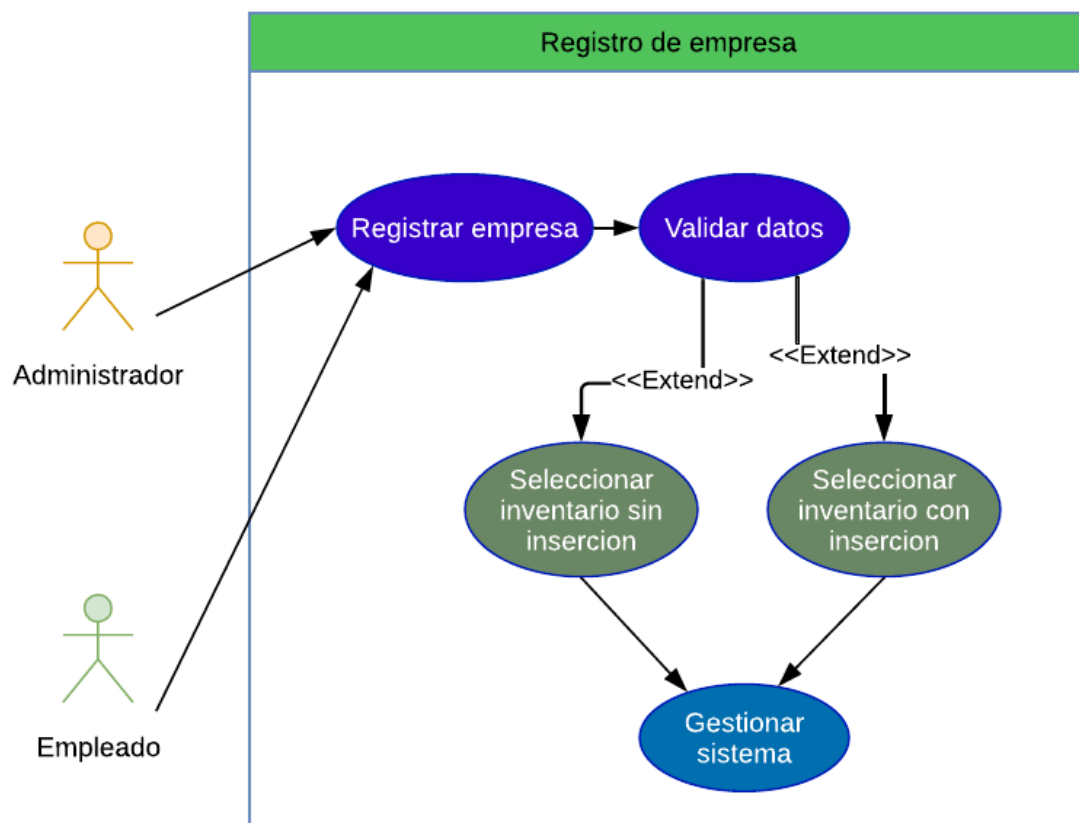


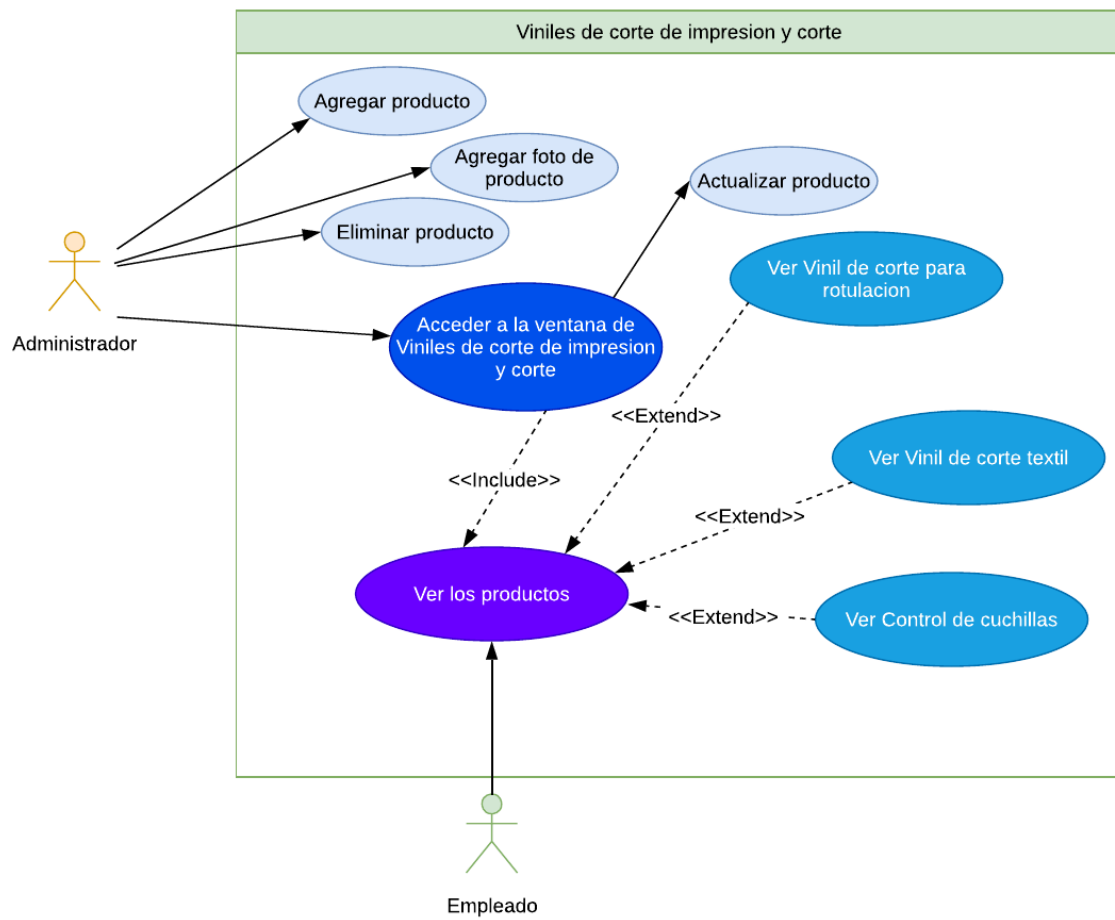


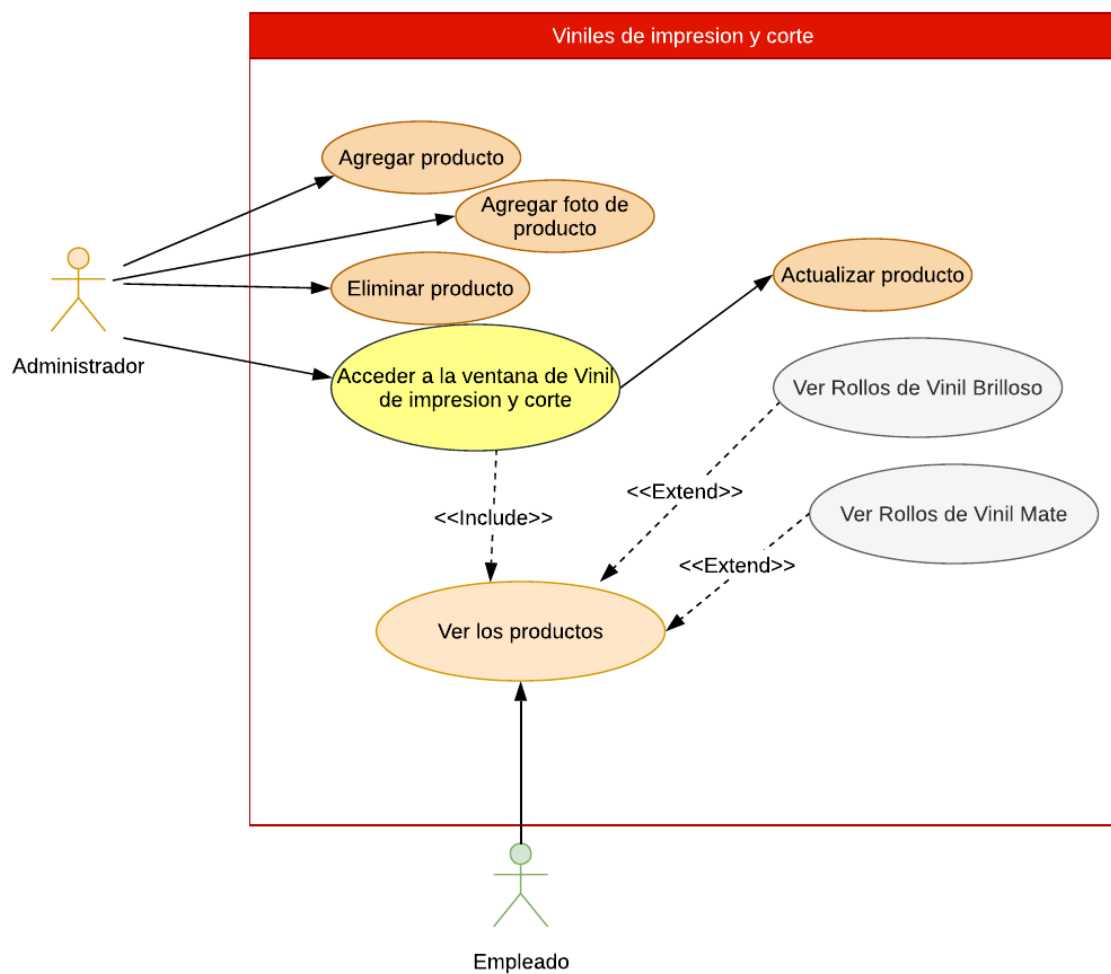












Nota: Diagrama, mejor calidad en el siguiente enlace

https://drive.google.com/file/d/1gmYsTONfvu2cFGPIJXQdOkNI-Z_KikrY/view?usp=sharing

Script de la base de datos

En el siguiente enlace se encuentra el script de la base de datos con los procedimientos almacenados, funcionales y estructuras de tablas.

[Script de Creación de Base de Datos](#)

4. Diccionario de datos

En el siguiente enlace se encontrará el diccionario de datos de la base de datos.

[Diccionario de la Base de Datos](#)

5. Arquitectura de Software

La solución se ha realizado con una arquitectura de Modelos y Vistas en Modelos, va la conexión de la base de datos, las entidades de la base de datos y los métodos del CRUD.

Mientras tanto en Vistas van todos los formularios, componentes personalizados para la visualización correcta de la arquitectura del software de nuestra aplicación.

Capa de Modelos

- **Propósito:** Gestionar todo el acceso y la lógica de datos
- **Componentes:**

ConexionBD: Clase dedicada para gestionar la configuración y apertura de la conexión del C# con SQL

Entidades: Clase que representa las acciones que realizará cada tabla en la base de datos

Capa de Vista

- **Propósito:** Gestionar toda la interfaz de usuario y la interacción con el usuario
- **Componentes:**

Formularios: Ventanas principales de la aplicación

Componentes Improvisados: Componentes creados por nosotros que son reutilizables

Flujo de Comunicación

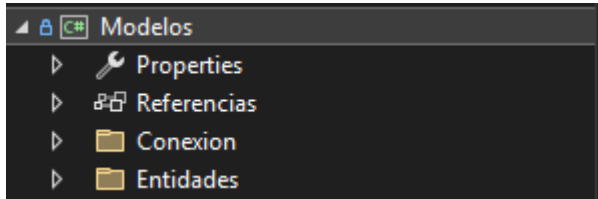
- La **Vista** captura la interacción del usuario (ej: hacer clic en un botón personalizado)
- La **Vista** recoge los datos del botón personalizado y los manda a un objeto **Entidad**
- La capa **Modelos** ejecuta la operación en la **base de datos SQL** y devuelve el resultado a la **Vista**
- La **Vista** muestra la interfaz al usuario y actualiza si es necesario

6. Estructura del proyecto

El proyecto se divide en backend y frontend, teniendo un grupo de carpetas apartado para todo los apartado para el mantenimiento local de los niveles del programa.

Estructura del backend:

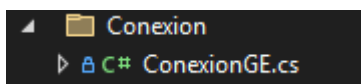
MODELOS



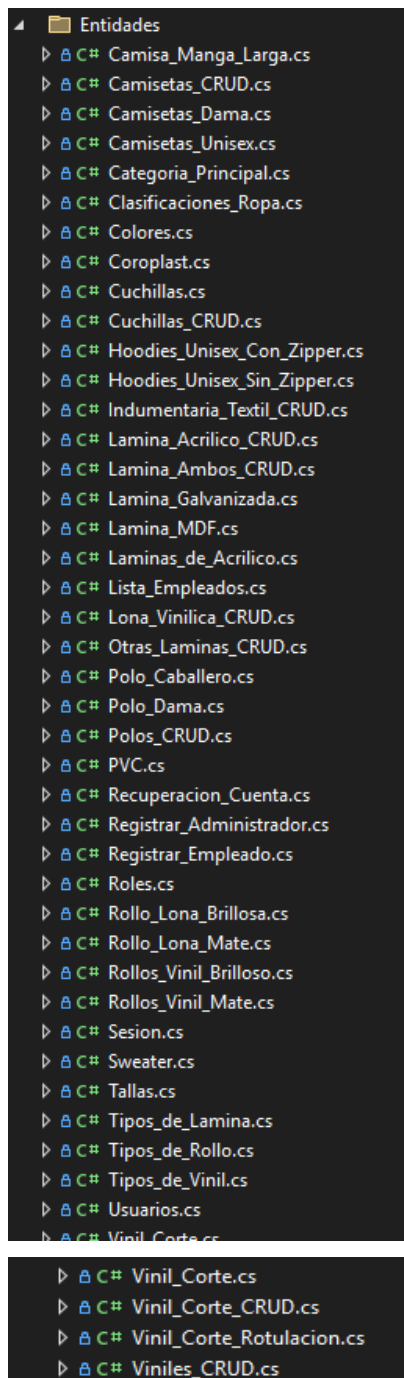
para la estructura del backend la aplicación tiene una estructura modular, para ello se optó por esa organización, reflejando todo la información que traerá la app desde el servidor, en resumen cada carpeta hace lo siguiente:

- **Conexión:** Es el archivo encargado de conectar el script de la base de datos con la línea de conexión de la computadora.
- **Entidades:** Es una carpeta donde contiene muchas clases donde se muestra la información recopilada, métodos de insertar, actualizar, eliminar, buscar, también contiene métodos para mostrar listas, datatables y la visualización de las vistas creadas en SQL Server Management Studio 20/21.

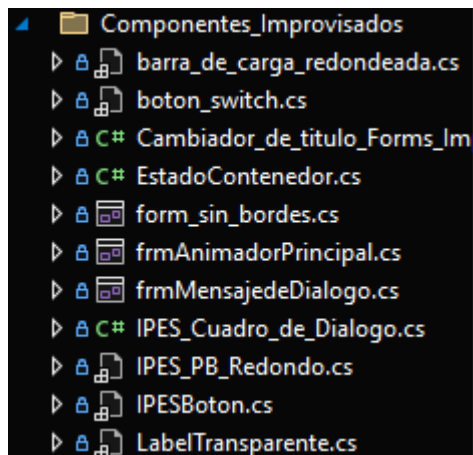
Estructura detallada de la carpeta conexión:



Esta carpeta contiene el método de conexión que se puede utilizar tanto en las entidades como en los mismos formularios, es el encargado de establecer la conexión entre la computadora y el sql, para poder identificar y acceder a todo el contenido del servidor con un solo método.

Estructura detallada de la carpeta entidades:

Aquí se encuentra toda la información recopilada de SQL en tablas distintas, contando con sus variantes y métodos del CRUD especificados para cada tipo de Inventario Distinto, contando con sus propios métodos para llamar a cada uno de estos datos seleccionados acorde a su categoría.

VISTAS:

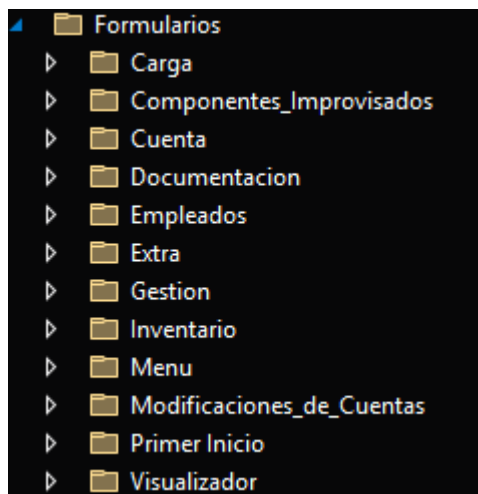
En la carpeta de **Vistas/Formularios/Componentes_Improvisados** se encuentran varias clases nuevas entre ellas contienen métodos, identificadores y cambios dinámicos, con los siguientes elementos:

- **barra_de_carga_redondeada.cs:** En esta clase se busca heredar al control “ProgressBar”, para poder redibujar una versión personalizada con capacidades nuevas de redondeo, colores personalizados y optimización para poder ser utilizado en cualquier formulario con varios estilos de llenado.
- **Cambiador_de_titulo_Forms_Improvisado.cs:** En esta clase lo que se busca en realidad es la instancia actual la que se está ejecutando para capturar la información del título llamando a su método, luego con ese dato capturado se lo envía al formulario contenedor “frmContenedor” ubicado en Vistas/Formularios/Visualizador/frmContenedor.
- **EstadoContenedor:** Es una clase donde se busca detectar si el formulario Contenedor esta Maximizado o en estado normal para poder llamar a su método en cualquier formulario, de esta manera se puede asignar cualquier función deseada con su método llamado, es un puente de detección entre un formulario hijo y el formulario padre.
- **form_sin_bordes:** Como todos sabrán, al eliminar los bordes de los formularios se restringen muchas funciones, como la de redimensionar, mover ventanas o maximizar, esta clase se encarga de traer de vuelta estas funciones con solo llamarla
- **frmAnimadorPrincipal:** Junto con la herencia de form_sin_bordes este formulario es el encargado de animar todos los formularios con llamarlo, gracias a esta clase, permite la herencia en cadena, de esta manera form_sin_bordes => frmAnimadorPrincipal => (formulario que procedera a usarla).
- **frmMensajedeDialogo:** Formulario tipo modal, es el encargado de remplazar los clasicos mensajes de advertencia, pregunta, error, información de Windows, con un diseño nuevo, personalizado y con 2 probabilidades de sonido en cada tipo mencionado, para dar un nuevo toque renovado y combinar aún más el estilo del minimalismo que acompaña al proyecto.
- **IPES_Cuadro_de_Dialogo:** Clase que realiza todas las llamadas al sistema Windows para convertir los mensajes clásicos en un renovado mensaje de diálogo modal.

- **IPES_PB_Redondo:** Clase que se encarga de convertir los pictureBox normales en imágenes redondeadas, con contorno redondo elástico, con bordes incluidos ya sea de un solo color o degradados.
- **IPES_Boton:** Clase que convierte los botones convencionales en botones con redondeado personalizado.
- **Boton_Switch:** Clase que da la posibilidad de convertir un boton en switch
- **Label Transparente:** Clase que permite que los Labels del Formulario sean transparentes

Estructura del frontend:

VISTAS



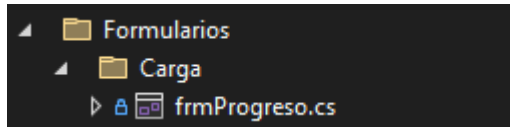
El frontend se divide de manera modular, contando con una organización acorde a la categoría que se va mostrando, para poder ser identificado de manera más sencilla, dividido en las siguientes carpetas:

- **Carga:** Contando con una pantalla de carga, esta se muestra al iniciar el programa indicando al usuario cual es el progreso que lleva de inicialización del programa.
- **Componentes_Improvisados:** Carpeta donde se encuentran todos los controles personalizados
- **Cuenta:** Carpeta donde va toda la información de las cuentas del usuario, ya sea administrador o empleado.
- **Empleados:** Carpeta donde va toda la gestión de empleados, administrado por el administrador (valga la redundancia).
- **Extra:** Carpeta donde va varios Easter Egg.
- **Gestión:** Carpeta donde va todos los apartados de Inserción // Modificación o Eliminación del Inventario con su respectiva búsqueda de C/U
- **Inventario:** Carpeta donde se almacena todos los formularios donde muestran las tablas del servidor en SQL
- **Menú:** Carpeta donde se almacena la pantalla principal (Dashboard) // Selector de Opciones
- **Documentación:** Apartado donde se muestran los documentos oficiales del programa.
- **Primer Inicio:** Apartado donde salen las primeras configuraciones del sistema.

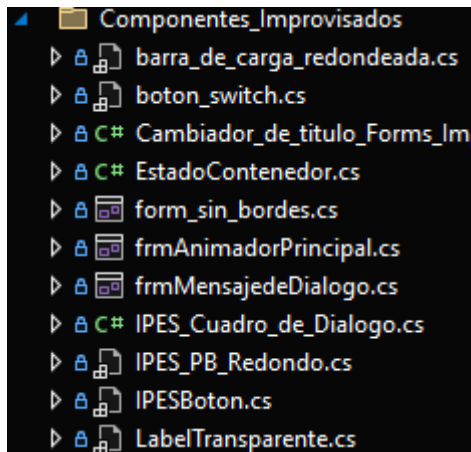
- **Visualizador:** Carpeta donde se almacena el Formulario Contenedor, Formulario donde se pintan todos los demás subformularios en el panel contenedor principal.

Estructura detallada de cada carpeta:

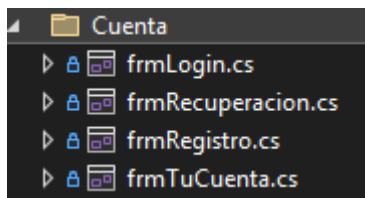
- **Carga:** Dentro de esta carpeta se encuentra **frmProgreso**, Formulario donde se indica la carga, con un diseño minimalista y moderno.



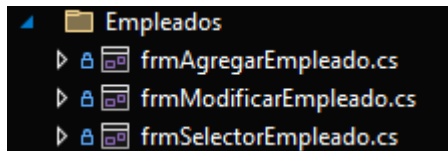
- **Componentes Improvisados:** En este apartado solo se mencionara al **frmAnimadorPrincipal**, también al **frmMensajeDialogo**, ya que estos son los únicos formularios que forman parte del frontend, **frmAnimadorPrincipal** es un formulario que se sobrepone encima de los otros, para mostrar las animaciones, mientras que el **frmMensajeDialogo**, es el formulario donde se muestran todos los mensajes de Windows, personalizados, con manejo del try catch entre otros.



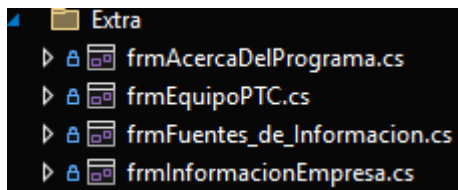
- **Cuenta:** En esta carpeta se encuentra lo relacionado a la cuenta administrador // empleado, aquí se encuentra frmLogin (Inicio de Sesión Universal), frmRegistro (Registro de la Cuenta Administrador), frmRecuperacion (Recuperacion de Contraseña Universal), frmTuCuenta (Formulario de Información de tu cuenta, Universal).



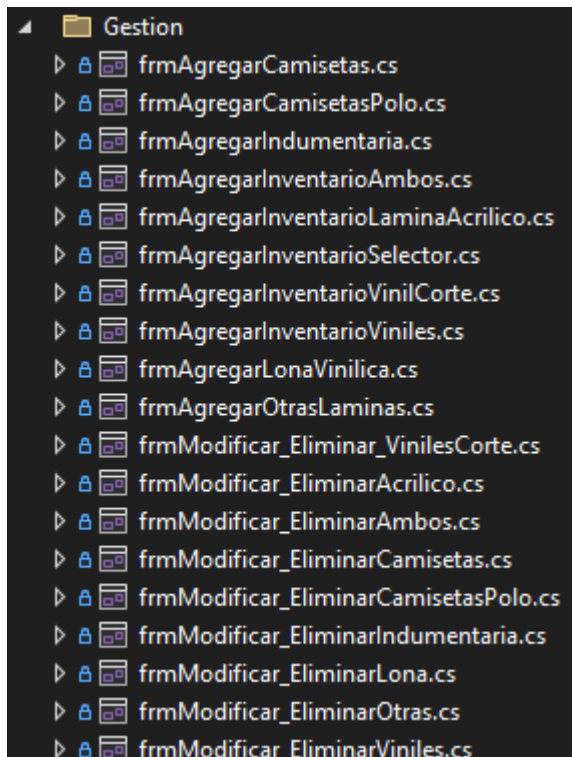
- **Empleados:** En esta sección se encuentra todo la gestión de empleados que puede acceder únicamente el administrador, contando con la capacidad de agregar/eliminar/Modificar empleados.



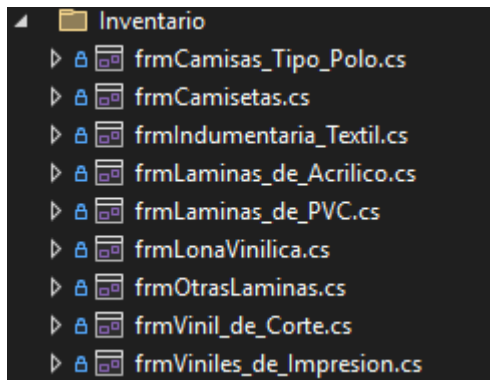
- **Extra:** En este apartado se encuentra todo lo relacionado a los easter egg (frmFuentesdeInformación), acá en esta sección se muestra todas las fuentes de donde se sacaron las librerías, los componentes del diseñador entre otras novedades más utilizadas en este proyecto, incluyendo los easter eggs y el acerca del programa.



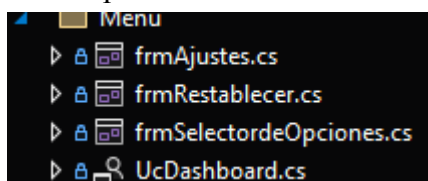
- **Gestion:** Carpeta Esencial de la solución informática, aquí generalizamos un poco la definición, el prefijo “Agregar” simboliza la inserción de datos de la categoría seleccionada, con un proceso sencillo y semi-automatizado, gracias a esto podemos categorizarlas en distintos formularios, los del prefijo “Modificar_Eliminar” son los formularios donde realizan lo opuesto de agregar, es decir que estos modifican // eliminan el inventario de la categoría seleccionada, contando con sus propias funciones de búsqueda, para localizar los productos de manera más rápida al igual que los de inserción.



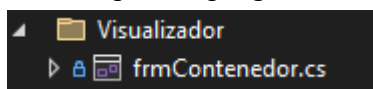
- **Inventario:** Aquí se almacenan los datos insertados, de cada categoría, marcados con su nombre identificativo, mismo que tendrá el “Dashboard”



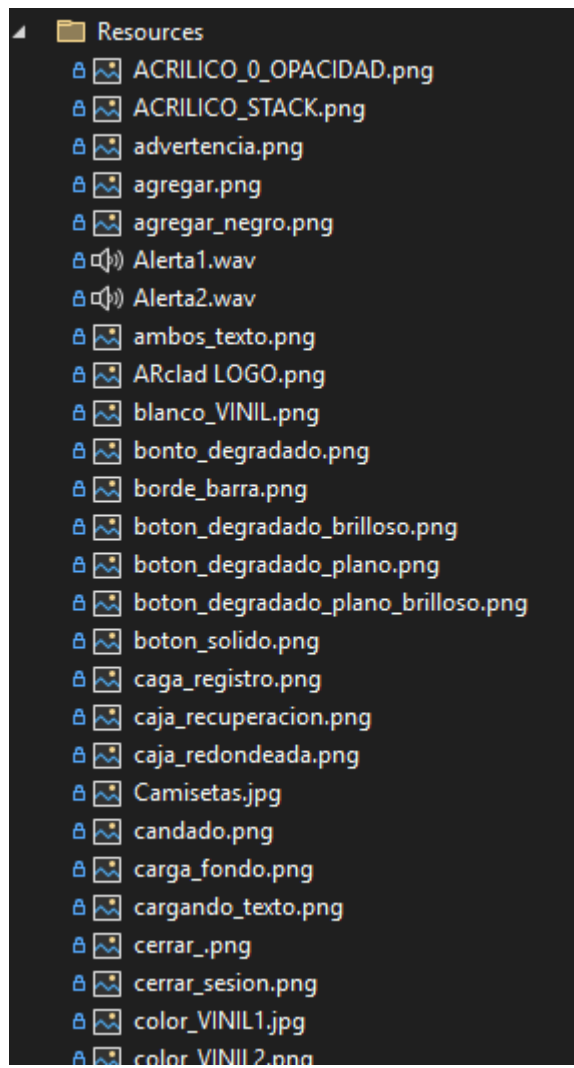
- **Menú:** Pantallas Centrales del Programa, aquí cambia un poco la manera en que organizamos el formulario, el UserControl hace presencia en el “UcDashboard”, acá es donde se muestra todos los botones para ir al menú de opciones, acceso a todo el contenido de la carpeta “Inventario” o a la carpeta “Gestión” luego nos encontramos con frmSelectordeOpciones, en este formulario se encarga de darnos las opciones, ocultando las que no son apropiadas acorde al rol, luego tenemos los ajustes en “frmAjustes” del programa, un apartado que nos permite configurar todas las preferencias del mismo, adicionalmente está el apartado para restablecer “frmRestablecer”, para eliminar configuraciones o restablecer el programa a los valores predeterminados.































- **Visualizador:** Carpeta donde aloja el formulario importante de todos, el formulario padre denominado “frmContenedor”, formulario donde se pintan todos los formularios hijos en su panel, es el visualizador principal, contando con funciones importantes y personalizadas, como títulos dinámicos, movimiento de ventana, botones personalizados, con bordes creados por nosotros mismos, un componente esencial para el programa.



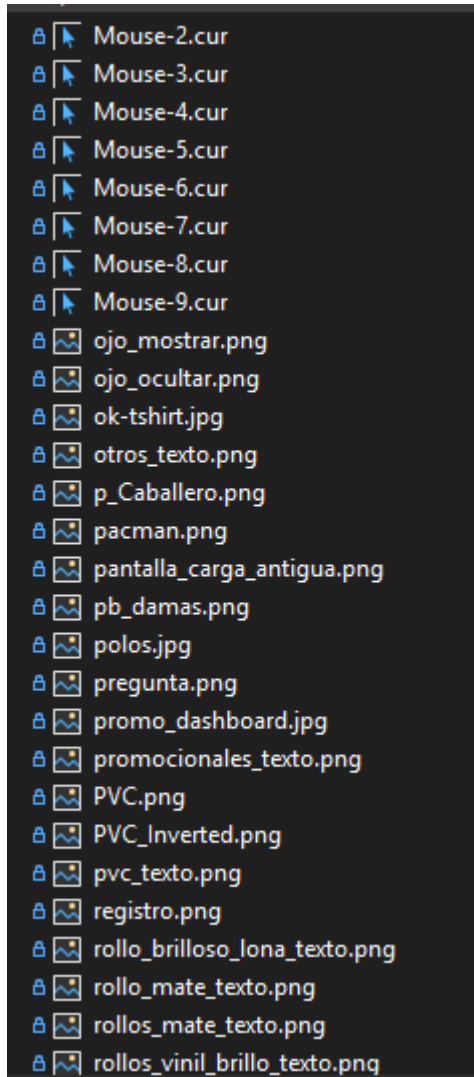
- **Resources:** Carpeta donde se encuentran todos los recursos // sprites // assets // audio // imágenes // PDF // entre otros, sin más que detallar acá se almacena todo.

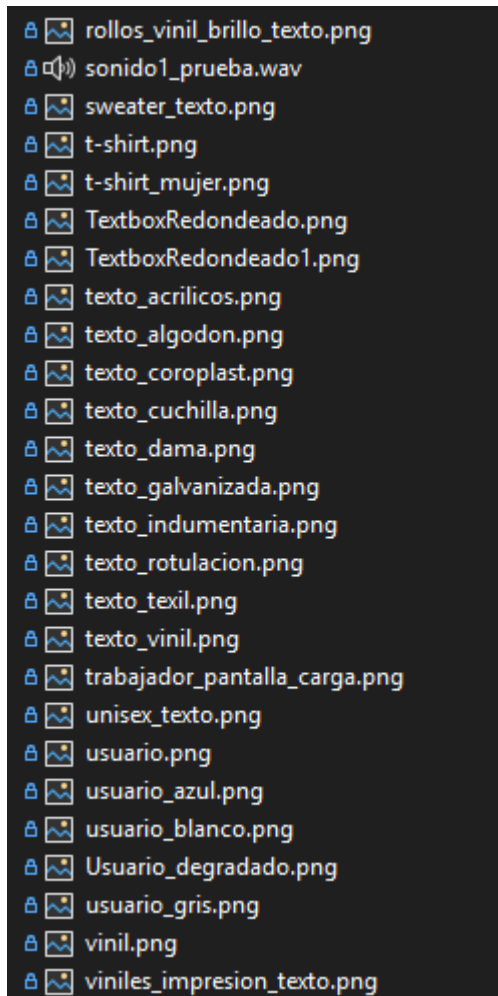


- 🖼️ color_VINIL3.png
- 🖼️ completado_texto.png
- 🖼️ contraseña.png
- 🖼️ COROPLAST.jpg
- 🖼️ correo.png
- 🖼️ corte_vinil.png
- 🖼️ cuadro_blanco.png
- 🖼️ cuarto_caja.png
- 🖼️ cuarto_caja_cara.png
- 🖼️ cuarto_caja_verde.png
- 🖼️ cuchilla.jpg
- 🖼️ cuenta_negrita.png
- 🖼️ cuenta_normal.png
- 🖼️ dashboard_impresion.jpg
- 🖼️ dashboard_solucion.png
- 🖼️ derecha.png
- 🖼️ dgv2.png
- 🖼️ Empleado_Selector1.png
- 🖼️ Empleado_Selector2.png
- 🖼️ error.png
- 🔊 Error1.wav
- 🔊 Error2.wav
- 🖼️ fondo_azul.png
- 🖼️ fondo_dgv.png
- 🖼️ fondo_gris.png
- 🖼️ Fondo1.png
- 🖼️ Fondo2.png
- 🖼️ Fondo3.png

 Fondo4.png
 Fondo5.png
 Fondo7.png
 fondo8.png
 fondo9.png
 galvanizado.jpg
 GE-MUSICATUTORIAL.wav
 GE MUSICATUTORIAL.mp3
 graduacion_negrita.png
 graduacion_normal.png
 hoodie_hombre.jpg
 hoodies.png
 hoodies_sin_zipper_texto.png
 hoodies_zipper_texto.png
 hora_redondeado.png
 icono_Blanco.png
 icono_Mini_Blanco.png
 icono_Mini_Negro.png
 icons8-audio-96.png
 icons8-invitado-masculino-96.png
 icons8-light-on-96.png
 icons8-no-audio-96.png
 icons8-user-settings-96.png
 icons8-usuario-pulsar-gradient-96.png
 informacion.png
 Informacion1.wav
 Informacion2.wav
 inicio_de_sesion.png

- 🔊 Interrogacion1.wav
- 🔊 Interrogacion2.wav
- 🖼️ Inventario_selector1.png
- 🖼️ Inventario_selector2.png
- 🖼️ inventario_textoo.png
- 🖼️ inventario_verde.png
- 🖼️ izquierda.png
- 🖼️ lamina_acrilico_texto.png
- 🖼️ linea_azul.png
- 🖼️ linea_verde.png
- 🖼️ logo.png
- 🖼️ LOGO_IPES.png
- 🖼️ logo1.png
- 🖼️ lona_cafe.jpg
- 🖼️ manga_larga_texto.png
- 🖼️ marca_de_agua.png
- 🖼️ material_soporte.jpg
- 🖼️ maximizar.png
- 🖼️ MDF.jpg
- 🖼️ mdf_texto.png
- 🖼️ medio.png
- 🔊 Mensaje-1.wav
- 🖼️ mensaje.png
- 🖼️ Minimizar.png
- 🖼️ Modificar_normal.png
- 🖱️ Mouse-1.cur
- 🖱️ Mouse-10.cur
- 🖱️ Mouse-2.cur





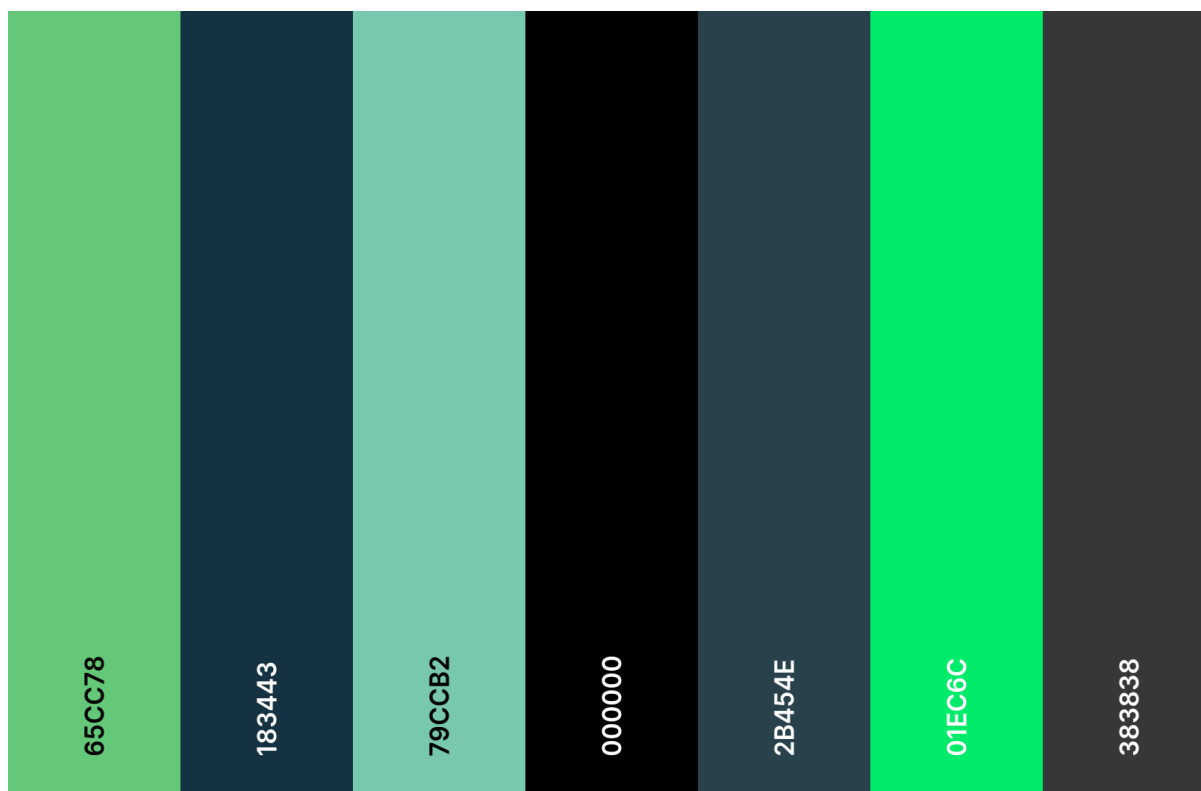
Contando con una cantidad inmensa de archivos, entre ellos archivos que se utilizan con mucha frecuencia, que dan el diseño, la forma del programa, es una cantidad considerable, tomando en cuenta la cantidad amplia de categorías y apartados que tiene nuestro programa.

7. Diseño de la aplicación

Paleta de colores

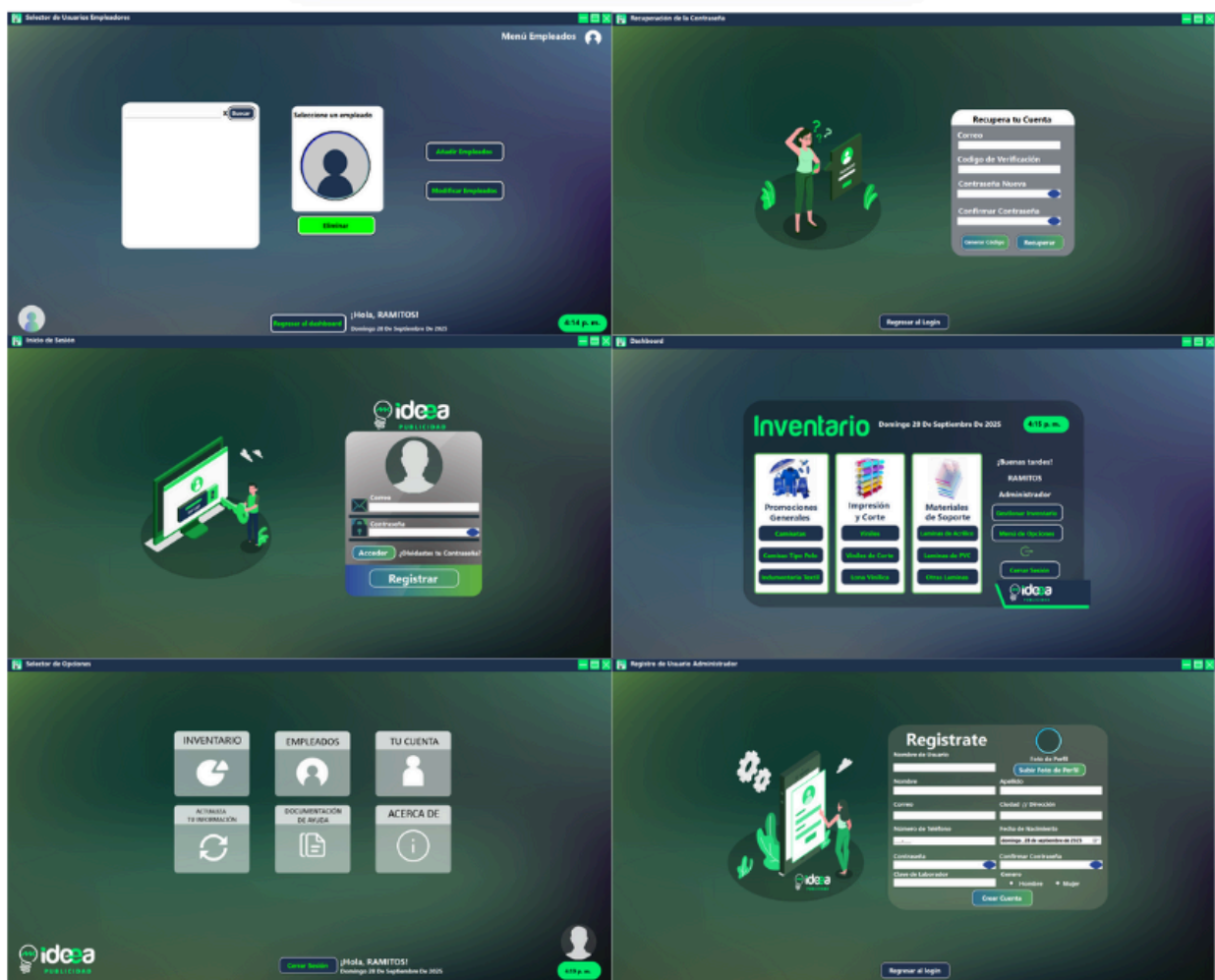
En el proyecto Gestión Empresarial de Inventario – Idea Publicidad El Salvador, se han empleado una paleta de colores que se centra en tonalidades de verde y negro, para mantener una estética innovadora y profesional, con el objetivo de focalizar identidad visual del proyecto. La paleta de colores se ha seleccionado cuidadosamente para asegurar una experiencia visual agradable y moderna.

- Paleta de Colores:



Aplicación en los diferentes elementos

En el proyecto Gestión Empresarial de Inventario IPES, nos enfocamos en tonos verdes que se utilizan de manera innovadora en elementos de la aplicación para ofrecer una experiencia de usuario eficiente y sencilla. A continuación, se muestran algunas imágenes para ver cómo se emplean estas paletas en los diferentes componentes:



Tipo de fuente utilizado en el proyecto Gestión Empresarial de Inventario IPES es:

Harabara
Calibri
 Segoe UI

Tipos y dimensión de imágenes utilizadas

Los tipos de imágenes utilizados en la aplicación son PNG y JPG. Los fondos tienen de dimensión 1920x1080px, en las imágenes si es variable el tamaño de la dimensión.

Tamaños de pantalla soportados

En el proyecto Gestión Empresarial de Inventario IPES, el sistema de escritorio está diseñado para ejecutarse en resoluciones mínimas de 1366 x 768 px (HD) y funciona eficientemente en resoluciones mayores como 1920 x 1080 px (Full HD).

El formulario principal Inicio de Sesión y sus ventanas principales han sido probadas en configuraciones estándar de monitores de 14" pulgadas a 24" pulgadas.

Al ser un desarrollo en Windows Forms, la aplicación se ajusta mediante los controles anclados (Dock/Anchor).

Resolución mínima recomendada: 1366 x 768 px (HD, portátiles y monitores pequeños).

Resolución estándar: 1920 x 1080 px (Full HD, monitores de escritorio).

En cuanto a la interfaz web complementaria desarrollada con HTML5 y CSS3, ofrece compatibilidad normal con navegadores básicos (Google Chrome, Microsoft Edge, Mozilla Firefox) y soporta los siguientes tamaños de pantalla:

Móviles: 375 px de ancho.

Tablet: 768 px de ancho.

Escritorio: 1366 px de ancho o superior.

8. Buenas prácticas de desarrollo

Estándares de programación en C#

Para el proyecto Gestión Empresarial de Inventario IPES, se adoptaron estándares de programación en C# que buscan asegurar que el código sea ordenado, claro y eficiente de mantener a largo plazo.

- Renombrar los elementos con su prefijo
- Las minúsculas las principio de los campos de SQL y las mayúsculas para cambiar de palabra
- Ordenar los formularios en subcarpetas para categorizarlos mejor
- Nombrar los botones acorde a su funcionamiento
- Uso de clases para facilitar su invocación en cada formulario
- Usar la versión más reciente y estable de las librerías
- Uso de los indicadores de ayuda (ToolTips) para ayudar al usuario con la interacción del programa.
- Uso del doble buffer para optimizar el formulario
- Añadir comentarios al código para saber qué función realiza
- Uso de encriptado para mejorar la seguridad de las cuentas
- Uso de vistas para simplificar los inner joins en SQL
- Uso del Try Catch para evitar errores con SQL

Conclusión:

Dentro del proyecto Gestión Empresarial de Inventario IPES permite mantener un código ordenado, legible y eficiente. Gracias a la correcta organización de formularios, el adecuado nombre de botones y controles, el uso de clases reutilizables, la implementación de algunas ayudas visuales, el desarrollo de la aplicación se vuelve más claro y sostenible. Estas prácticas contribuyen a que la interfaz gráfica sea accesible y fácil de mantener en el tiempo.

Estándares de programación en SQL Server

Los estándares de programación en SQL Server utilizados en el sistema local Gestión Empresarial de Inventario IPES conforman algunas reglas y buenas prácticas que aseguran la coherencia y eficiencia en la definición de tablas, consultas y procedimientos almacenados.

- Uso de nombres en minúscula para tablas y campos, empleando guión bajo para separar palabras.
- Declaración de llaves primarias y foráneas en todas las tablas para garantizar la integridad.
- Creación de índices en columnas clave para optimizar consultas frecuentes.
- Definición de procedimientos almacenados en lugar de consultas directas para mejorar la seguridad y la reutilización del código.
- Uso de vistas para simplificar consultas complejas y facilitar la gestión de reportes.
- Inclusión de comentarios en cada procedimiento realizado, vista o función para documentar su propósito y uso.
- Aplicación de convenciones en nombres de variables dentro de procedimientos almacenados, iniciando con el prefijo p_ para parámetros de entrada y v_ para variables locales.
- Aplicación de COLLATE adecuado para evitar problemas en el inicio de sesión por diferencias en mayúsculas o acentos.
- Normalización de las tablas para evitar redundancia de datos y asegurar la consistencia.
- Validación de datos con CHECK, DEFAULT, NOT NULL (constraints) para reforzar la lógica empresarial en la base de datos.

Conclusión:

El establecimiento de estándares en SQL Server en el proyecto Gestión Empresarial de Inventario IPES asegura que la base de datos mantenga su integridad y rendimiento. La adopción de convenciones en nombres de tablas y campos, el uso de claves primarias y foráneas, la normalización, la definición de índices claros, la creación de procedimientos almacenados y vistas también, así como la integración de transacciones y constraints, fortalecen la seguridad y eficiencia del sistema. Estos lineamientos ayudan a la administración de los datos, garantizando confiabilidad en el futuro.

9. Requerimientos de hardware y software

Local

Cuando se trata de requerimientos del hardware y software en entorno local se refiere a los requerimientos análogos y digitales para una ejecución efectiva del sistema.

Hardware mínimo:

- 8GB de RAM.
- Procesador Intel core i5 o AMD Ryzen 5 3500U
- 500GB de espacio.

Software mínimo:

- Sistema operativo :
- Windows 7, Windows 8/8.1, Windows 10/11.

Entorno del desarrollo en el sistema:

- Visual Studio 2022, Visual Studio 2017
- SQL Management Studio 20/21, SQL Management Studio 2014

Lenguajes:

- C#
- SQL Server.

10. Instalación y configuración

Instalación en entorno local.

Para instalarlo necesitarás descargar el repositorio, ejecutar el script (GE_IPES_DB),

Abierto el Proyecto en SQL Server Management Studio establece la conexión en **(local)**, y listo, ya estaría conectado.

Requisitos previos.

Antes de comenzar la instalación, asegúrate de tener los siguientes componentes instalados:

1. Servidor local: SQL Management Studio 20/21, o SQL 2014 (Sql Server)
2. Lenguaje de programación: C# (Visual Studio 2022) (Compatible con Visual Studio 2017, pero no recomendado)
3. Para clonar el repositorio del proyecto: Git.
4. Pasos para la instalación
5. Clonar el proyecto o descargar el código fuente.
6. Clonar el repositorio del sistema o descarga el archivo zip desde el repositorio:
7. <https://github.com/Omar-Rivera-98/Idea-Publicidad-El-Salvador> (Acceso Exclusivo a desarrolladores pertenecientes al proyecto // Evaluador/a).

Configurar la base de datos.

Crea una base de datos en tu servidor local de (Nombre de la base) e importa el archivo .sql que se encuentra en el siguiente enlace proporcionado:

1. En SQL Establece la Conexión en **(local)**
2. [Script de Creación de Base de Datos](#)
3. Esto creará las tablas que se necesitan para que el sistema funcione bien.
4. Instalar dependencias.
5. Probar la aplicación