



CONSEJERÍA DE EDUCACIÓN
Comunidad de Madrid

IES ENRIQUE TIERNO GALVAN
Parla

**CFGS DESARROLLO DE APLICACIONES
MULTIPLATAFORMA**
Curso 2024/2025

Proyecto DAM

***TITULO: Desarrollo de un sistema de gestión
multiplataforma con Flutter y Odoo***

Alumno: Laura Salas Ávila

Tutor: Julián Parra Perales

Junio de 2025

Contenido

Descripción y Análisis general del proyecto.	3
Contexto y mundo real del problema.....	3
Análisis del mercado.....	3
Justificación de este proyecto y diferencia del mercado.....	4
Casos de uso.....	4
Requisitos (funcionales y no funcionales)	9
Análisis y Diseño de la base de datos.....	9
Aspectos funcionales y de diseño de la aplicación.....	11
Implementación de la aplicación.	11
Entorno de desarrollo.....	11
Herramientas, lenguajes y librerías usadas en el proyecto	11
Conexión entre el backend y la base de datos.....	11
Interfaz de usuario.	11
Descripción de la interfaz.	11
Diagrama de navegación.....	12
Manual de usuario.....	13
Realización de pruebas	13
Aspectos destacables del desarrollo	13
Problemas a la hora del desarrollo	13
Error 500 en odoo.....	13
Innovaciones del proyecto	13
Líneas de trabajo futuro.....	14
Requisitos pendientes e ideas pensadas para las siguientes versiones.....	14
Conclusiones.....	14
Bibliografía/Enlaces de interés.....	14
Anexos.	15
Código.....	15
GitHub	15

Descripción y Análisis general del proyecto.

Contexto y mundo real del problema

Actualmente las empresas prefieren estar más cerca de lo digital, la comodidad y también la movilidad para poder gestionarse de una manera más eficiente y rápida, es por eso que las empresas buscan herramientas o sistemas que las permitan realizar fácilmente sus tareas desde diferentes plataformas y dispositivos. Por este motivo es que las aplicaciones ERP pueden llegar a ser muy utilizadas para poder llegar a tener o mejorar esa gestión de información, pero muchas de estas soluciones ERP llegan a ser algo difíciles de usar o no tienen forma de usarse en múltiples plataformas ya sea porque no es compatible o porque te hacen comprar una licencia para la personalización. Es por esto que se ha decidido solucionar este problema que iremos explicando a lo largo de este documento.

Análisis del mercado

Se ha realizado un pequeño análisis del mercado para ver que aplicaciones intentan realizar esta función de sistema de gestión y poder compararlas con nuestra idea. Se han escogido para realizar una pequeña demostración los siguientes cuatro:

- Odoo: Es bastante usado, de código abierto, personalizable... Pero es más usado solo para escritorio ya que su versión móvil está menos actualizada y a los usuarios no les gusta.
- SAP: Es un sistema que tiene muchísimas opciones, pero se debe de pagar mucho por las licencias.
- Holded: Es para pequeñas empresas y de pago, es intuitiva y tiene algunas limitaciones para la personalización.
- Sage: Tiene varios productos dependiendo del tamaño de tu empresa, pero todos ellos de pago, tiene limitaciones y debes de pagar más por módulos adicionales que quieras añadir.

Como se puede ver en estos ejemplos, la mayoría de estos sistemas pueden llegar a no ser compatibles del todo, algunos son caros, en otros tienes que pagar una licencia para poder usarlos de manera completa o realizar personalizaciones, pagar de más por tener más opciones, o simplemente ni siquiera tener la libertad de personalizar.... Estas cosas

pueden a llegar a ser no muy buenas para empresas pequeñas o medianas que no puedan permitirse pagar o tener más dispositivos para poder usar estos sistemas, o incluso empresas más grandes que no quieren desperdiciar nada de dinero y prefieren versiones gratis y más útiles.

Justificación de este proyecto y diferencia del mercado

Es por esto que en este proyecto vamos a dar solución a los problemas explicados anteriormente y vamos a diferenciarnos del resto de sistemas de gestión, permitiendo a las empresas tener una forma fácil, cómoda y multiplataforma de acceder o interactuar con sus datos desde cualquier sitio y con cualquier dispositivo de manera gratis y libre, de esta manera nuestro proyecto se diferenciaría mucho del resto en el mercado.

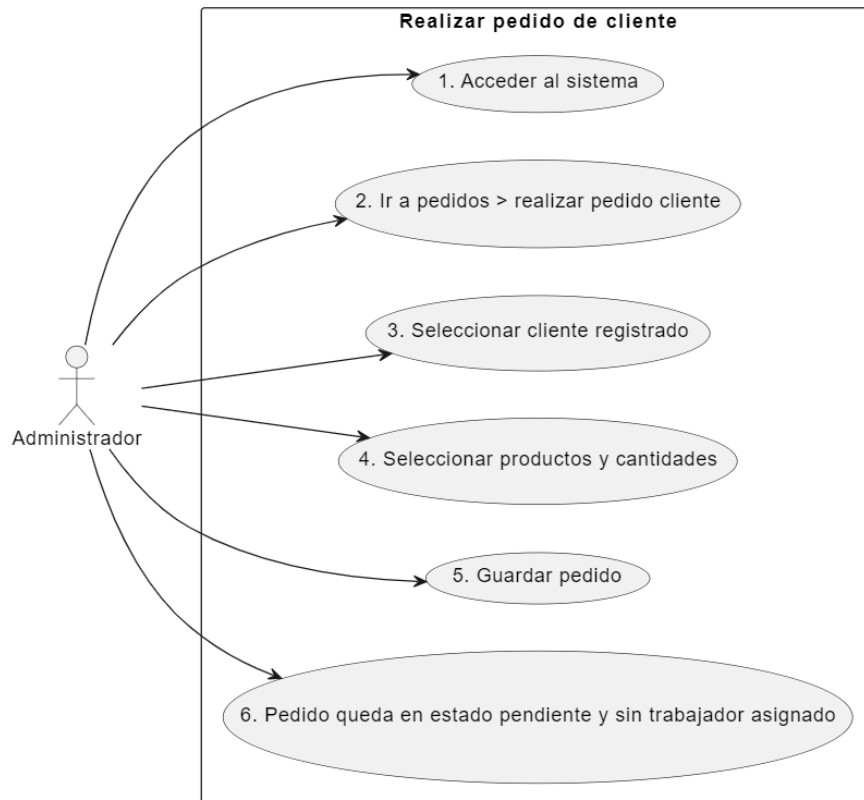
En resumen, en este proyecto se va a desarrollar una aplicación de gestión multiplataforma con el objetivo de hacer más fácil y cómoda la forma en la que se accede o se realizan cambios en la información de una empresa desde los diferentes dispositivos de los empleados. En este sistema se hará uso de Flutter para el desarrollo de la interfaz de usuario (esta parte será el frontend y con ella tendremos la parte visual), de Odoo para la gestión de los datos de la empresa (actuará como backend y podremos gestionar productos, clientes, trabajadores...) que usa PostgreSQL como su base de datos donde se guarda toda la información, para el intercambio de datos entre el backend y el frontend se usará JSON de manera HTTPS (se intentará) para tener una conexión mucho más segura y por último (por el momento) Docker, que es usado como herramienta de virtualización para nuestra base de datos, el proxy que se encargará de la seguridad y Odoo.

Casos de uso

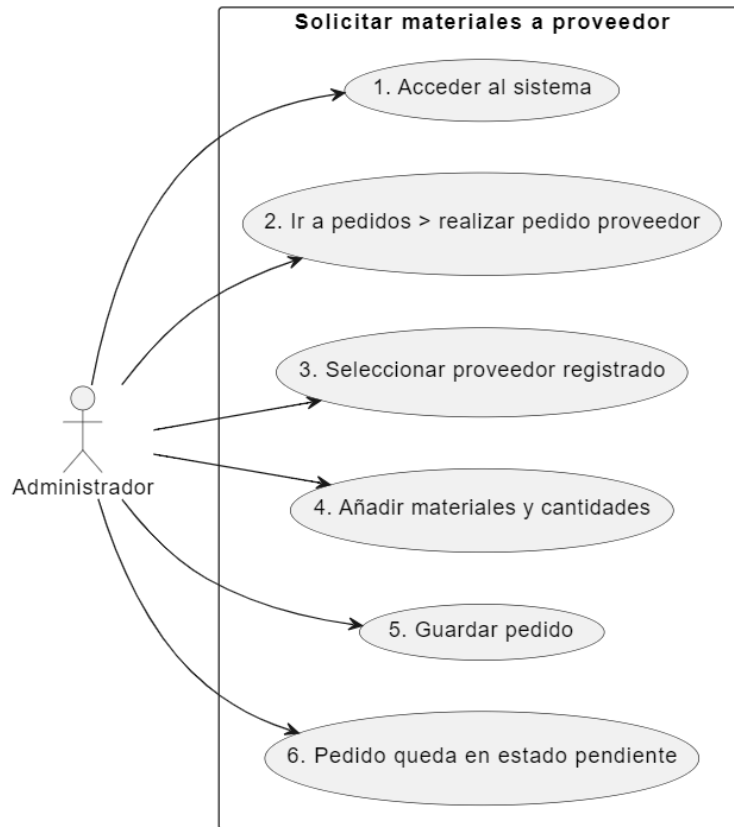
Se han realizado varios casos de usos para cada una de las acciones que se van a poder realizar en nuestra aplicación (por el momento llevo solo estas que están relacionadas con el tema de pedidos que será la parte funcional de la aplicación, veré que puedo ir añadiendo):

- Pedido de un cliente:

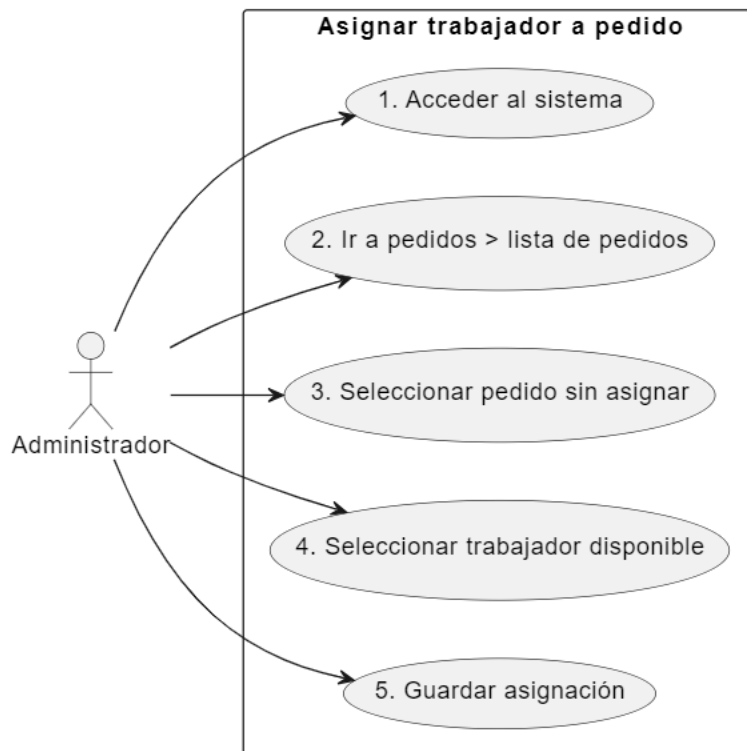
- Para realizar el pedido de un cliente necesitamos ser administrador y acceder a la aplicación, debemos de ir al apartado de pedidos > realizar pedido cliente, allí seleccionaremos al cliente, el producto que quiere más su cantidad y guardaremos el pedido, el pedido será registrado y estará en un estado de pendiente y también en estado de necesitar ser asignado a un trabajador.



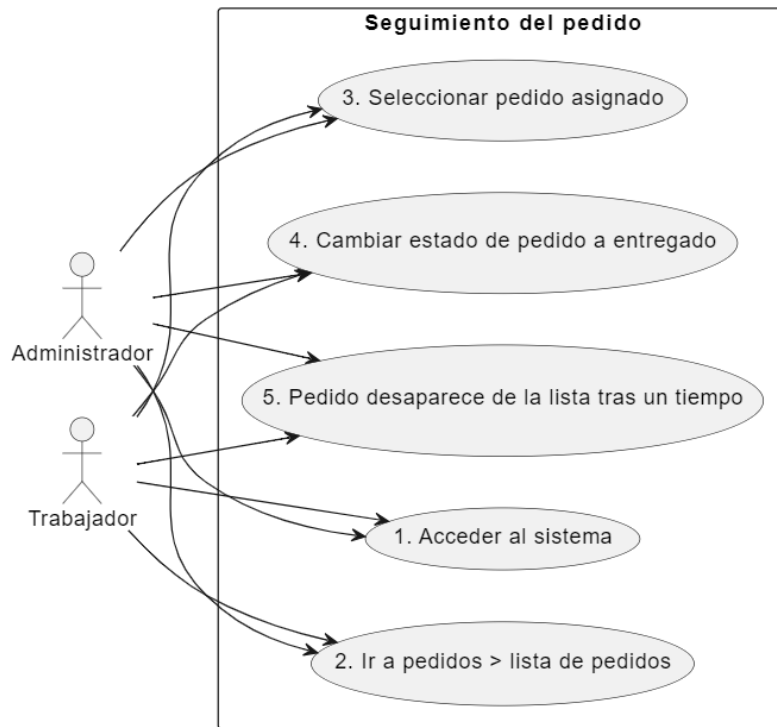
- Solicitud de materiales a los proveedores:
 - Para solicitar materiales a los proveedores necesitamos ser administrador, acceder al apartado de pedidos > realizar pedido a proveedor, allí seleccionamos al proveedor que queramos junto con la cantidad del material que venda, el pedido se guardará y será registrado como pedido pendiente.



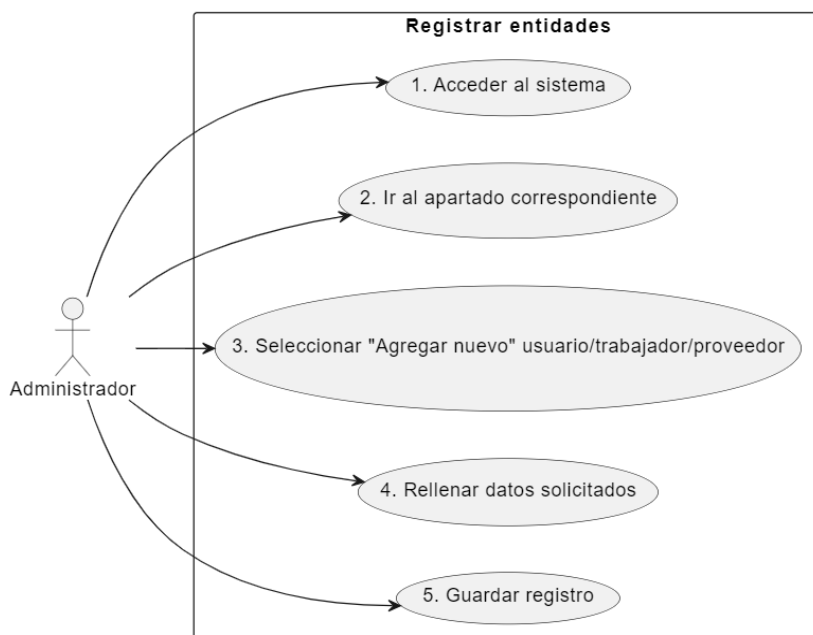
- Asignar el pedido/trabajo a un empleado:
 - Para asignar un pedido/trabajo a un empleado debemos de ser administrador, acceder al apartado de pedidos > lista de pedidos, donde podremos ver todos los pedidos y asignar a cada uno de ellos un trabajador disponible, el pedido será guardado de nuevo y se verá que ahora tiene asignado un trabajador.



- Seguimiento del pedido:
 - Para el seguimiento de un pedido podemos ser tanto administrador como trabajador, se debe de acceder al apartado de pedidos > lista de pedidos, allí podremos cambiar el estado SOLO si somos administrador o el responsable de ese pedido, al cambiar el estado, el pedido se mostrará como entregado y después de un rato desaparecerá de la lista. (por algún motivo se descoloca todo al escribir, esto ha sido lo mejor que se ha quedado)



- Registrar un usuario/trabajador/proveedor
 - Para registrar a cualquiera de ellos simplemente debemos de ser administradores, entrar al apartado correspondiente y elegir la opción de agregar un nuevo (usuario/trabajador/proveedor), rellenaremos los datos correspondientes y simplemente guardaremos los datos.



Requisitos (funcionales y no funcionales)

Repasando todo lo anterior los requisitos tanto funcionales como no funcionales que son necesarios para este proyecto son los siguientes: (no sé si me faltarán, iré viendo)

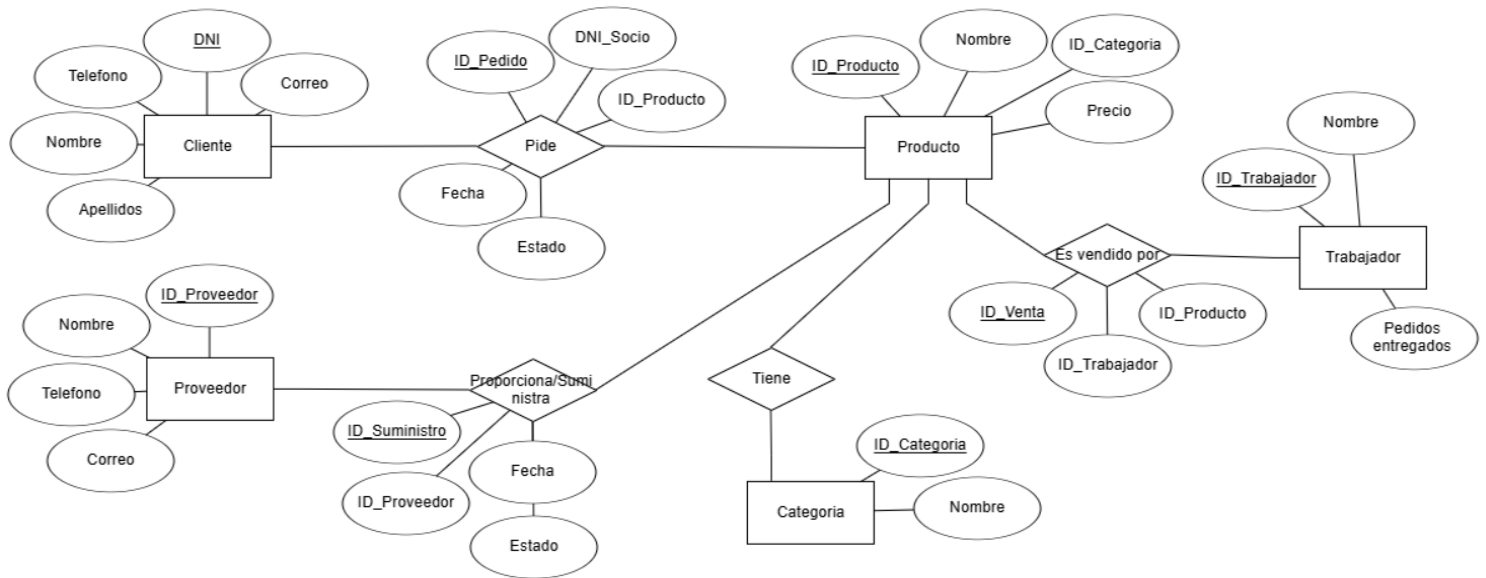
- Funcionales:
 - Se deben de poder registrar y guardar clientes, proveedores, trabajadores, productos y categorías.
 - Se deben de poder crear pedidos de clientes o pedidos a proveedores.
 - Se debe de poder asignar trabajadores a pedidos.
 - Se debe de hacer seguimiento del pedido y poder marcarlos como entregados.
 - Se debe de tener un login que dependiendo del usuario tendrá más permisos o menos.
- No funcionales:
 - La aplicación será multiplataforma.
 - La comunicación entre front y back será mediante HTTPS y JSON.
 - La interfaz debe de ser sencilla y fácil de usar para los usuarios.

Análisis y Diseño de la base de datos.

Nuestro proyecto hace uso de Odoo como ERP, y este utiliza una base de datos PostgreSQL que se encarga de guardar toda la información de los módulos que tenga Odoo. Esta base de datos no fue creada desde cero ya que se creó automáticamente junto con Odoo al realizar la dockerización, pero se han revisado todas las tablas y se han analizado para ver cuáles de ellas faltan y cuáles de ellas nos podrían ser de utilidad entre Odoo y la aplicación que estamos desarrollando.

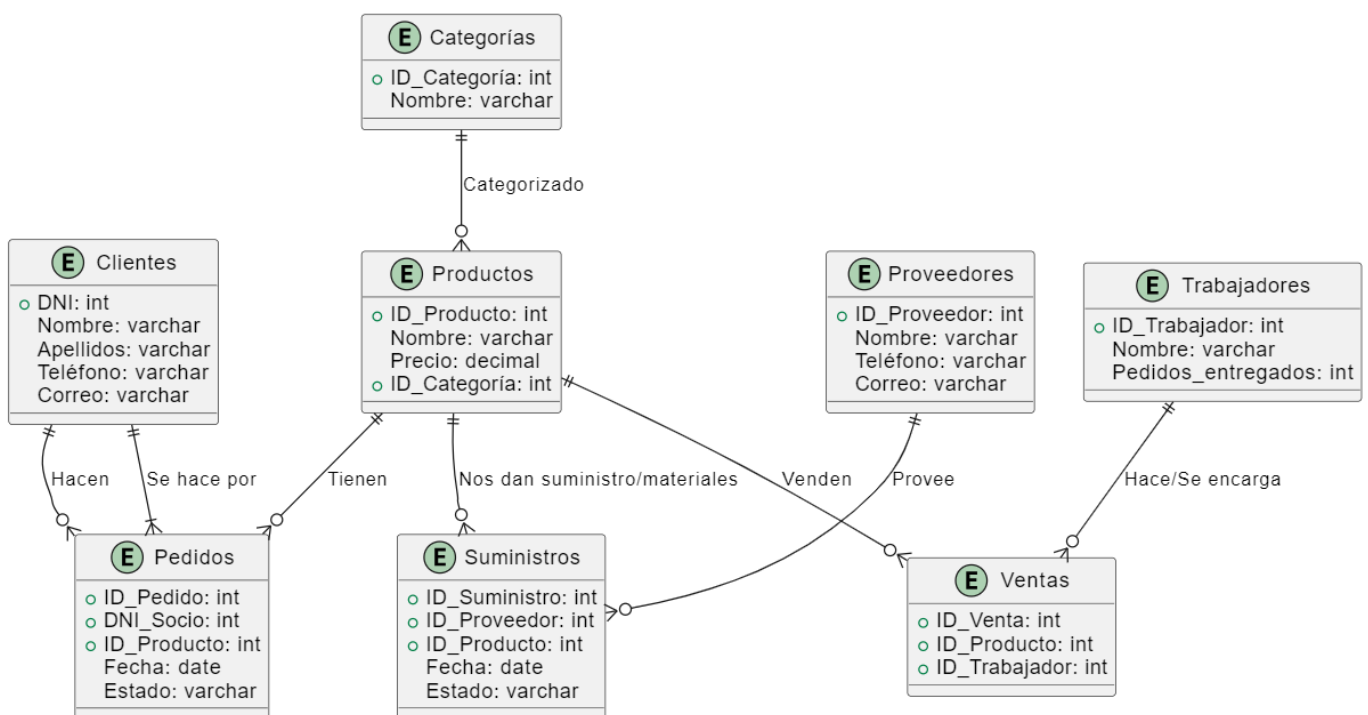
- Modelo conceptual MER:

- Este es el modelo entidad relación del proyecto, donde se pueden ver las entidades principales y sus relaciones. (hacer explicación próximamente)



- **Modelo lógico:**

- El modelo lógico de nuestro proyecto es el siguiente, basándonos en el modelo MER hemos sacado el lógico. (hacer explicación próximamente)



- Relación con los módulos de Odoo (hacer explicación próximamente)

Aspectos funcionales y de diseño de la aplicación

En este apartado vamos a explicar la información de los paquetes/clases/interfaces que componen todo nuestro código, este se divide en varias partes que son las siguientes: (se ha puesto lo que se lleva por el momento, se deben ir añadiendo todo más completo)

- Parte backend: En esta parte tenemos el archivo que se ha configurado para poder crear nuestra base de datos y Odoo, se trata de un docker-compose que.....
- Parte frontend: En este apartado tenemos todo lo necesario para que la aplicación funcione de manera multiplataforma, se ha intentado mantener una organización limpia usando BLoC

Implementación de la aplicación.

Los aspectos más relevantes y consideraciones que debemos tener en cuenta a la hora de implementar la aplicación son varios,

Entorno de desarrollo

Herramientas, lenguajes y librerías usadas en el proyecto

Conexión entre el backend y la base de datos

Interfaz de usuario.

Descripción de la interfaz.

He realizado unos bocetos sobre cuál sería la idea para la interfaz del usuario, como se va a poder ver, es un diseño bastante agradable y simple, en cada apartado haremos una breve explicación de los aspectos de usabilidad que se han tenido en cuenta:

(ignora todo esto, es para mi organización)

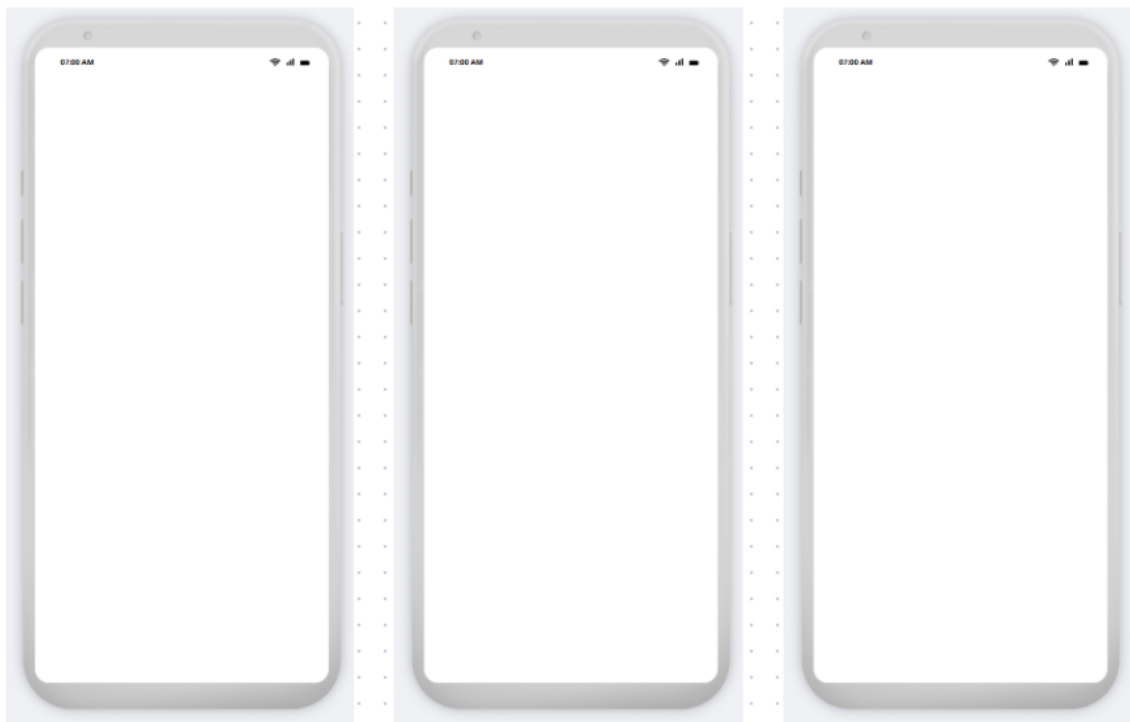
Diseño de las pantallas:

- UI (vistas)
- IxD (interacción)
- UX (experiencia)

- Inicio: La pantalla de inicio/login es simple y sencilla de usar, contiene varias cajas de texto para poder identificarte (nombre y contraseña), una imagen, un botón, texto... En esta interfaz se ha asegurado que la letra sea adecuada para el usuario, que los colores sean suaves, no dañinos y que sea muy intuitiva de usar.
- Menú: El menú consta de unos botones, el perfil del usuario y un botón de ayuda, al igual que en el diseño anterior se ha tenido en cuenta la letra, los colores, la organización, la facilidad de uso de esta interfaz, ayuda al usuario....
- Dentro de las opciones: Este es un ejemplo de cómo se verían algunas de las pantallas de opciones que vienen en el menú, cada una de las opciones será diferente y adecuada. (poner diseño del apartado de pedidos).

Diagrama de navegación

Este es el diagrama de navegación de las pantallas de nuestra aplicación, en él explicaremos cómo nos desplazaríamos por cada una de ellas. (imagen por defecto, tengo que ponerme a diseñar la idea que tenía pensada para cada pantalla)



Manual de usuario.

El manual de usuario viene a parte en otro documento a menos que se deba de poner aquí, en ese caso se cambiará todo, en este manual se explicará de manera sencilla y detallada junto con imágenes cómo funciona la aplicación y cómo usarla para que un usuario no tenga ningún problema, también se van a añadir las formas de acceso para las pruebas (los nombres de usuarios y las contraseñas).

Realización de pruebas

Aspectos destacables del desarrollo

A medida que se ha ido realizando el proyecto, fueron surgiendo innovaciones y varios problemas que se fueron solucionando, vamos a explicar cada uno de ellos en diferentes apartados y en el caso de los problemas, serán explicados en el orden en el que aparecieron mientras se iba avanzando.

Problemas a la hora del desarrollo

Error 500 en odoo.

Al docketizar Odoo y la base de datos con docker-compose saltó el error 500 al intentar acceder a Odoo, revisando en la terminal los logs realizados se vio que el error era por los puertos que estaban ocupados y por la inicialización de la base de datos, este problema fue solucionado al cambiar los puertos y al añadir unas líneas de código que no había añadido en mi docker-compose.

Innovaciones del proyecto

En este proyecto hemos usado varias innovaciones que iremos puntualizando y explicando sobre por qué se usaron y dónde se usaron.

- Ir poniendo poco a poco

Líneas de trabajo futuro.

Requisitos pendientes e ideas pensadas para las siguientes versiones

En este momento se debe de realizar la parte de la conexión y la interfaz, se pondrá en funcionamiento alguno de los apartados de la aplicación, pero en un desarrollo futuro todas las opciones estarían disponibles, más optimizadas e incluso se podrían añadir más, mejorando así la aplicación. Algunos de los requisitos o ideas pensadas (por el momento) para las futuras versiones de la app son las siguientes:

- Poder tener mejor seguridad.
- Que la aplicación sea muy escalable y no de problemas.
- Tener una mejor accesibilidad para personas con discapacidades.
- Poder realizar actualizaciones y añadir nuevas opciones/módulos.
- Poder añadir automatización en ciertos apartados de la aplicación, incluso hacer uso de inteligencia artificial para crear diagramas o analizar datos que den reportes.
- Tener notificaciones en tiempo real e incluso chats entre trabajadores para el paso de información.
- Mejorar el seguimiento del pedido y sus estados.
- Mejorar los roles de los usuarios y todos sus permisos.

Conclusiones.

Creo que podría quedar algo chulo, he estado aprendiendo y leyendo cosas nuevas que han estado muy interesantes a medida que he ido avanzando e investigando para la realización de este proyecto. Se irán corrigiendo las cosas que estén mal y se irán añadiendo más cosas. (no sé bien qué añadir aquí)

Bibliografía/Enlaces de interés.

Ignorar esto por el momento, solo voy añadiendo, ya lo pondré de manera correcta.

Información sobre mercado sacado de:

- [Comparación de las 10 principales soluciones ERP en 2025](#)

Información sobre Docker y creación de mi docker.compose:

-

Anexos.

Código

Futuros apartados de código importante que comentar.

GitHub

Repositorio de GitHub con el documento subido: [LauraSA29 \(Laura Salas\)](#)