

Manual técnico

Proyecto final de Grado Superior de Desarrollo de Aplicaciones Multiplataforma

Estudiante:

Sara Herrera Lagranja

ÍNDICE

1. Evaluación del proyecto	2
1.1. Contexto	2
1.2. Objetivo	2
1.3. Estudio de mercado	3
2. Solución propuesta y requisitos técnicos	4
2.1. Solución propuesta	4
2.2. Requisitos técnicos	4
3. Desarrollo de la aplicación	5
3.1. Planificación	5
3.1.1. Recursos físicos	5
3.1.2. Recursos técnicos	5
3.1.3. Recursos humanos	5
3.1.4. Previsión de carga de trabajo y programación de tareas	6
3.2. Análisis de datos y/o procesos	7
3.2.1. Requisitos funcionales	7
3.2.2. Requisitos no funcionales	7
3.2.3. Análisis de datos	7
3.2.4. Análisis de procesos	8
3.3. Diseño de datos y/o procesos	12
3.3.1. Diseño de datos	12
3.3.2. Diseño de procesos	13
3.4. Diseño de interfaz	14
3.5. Diseño de arquitectura	16
3.5.1. Modelo-Vista-Controlador (MVC)	16
3.5.2. Data Access Object (DAO)	16
4. Pruebas	16
5. Conclusiones	17
6. Presupuesto	18
7. Bibliografía	19

1. Evaluación del proyecto

1.1. Contexto

La producción de frutas en España en 2022 alcanzó un valor de aproximadamente 15.680 millones de euros, representando el sector hortofrutícola, que incluye frutas y hortalizas, un 2.6% del Producto Interno Bruto (PIB) agrícola de España y un 1.4% del PIB total del país [1]. A nivel de la Unión Europea, España es el mayor productor de frutas, con una producción de 27 millones de toneladas en 2021 [2]. Además, a nivel mundial, el valor global del comercio de frutas fue de aproximadamente 138.000 millones de euros, lo que subraya la importancia económica de este sector tanto a nivel nacional como global [1].

El cultivo de frutas implica una gran variedad de especies, como melocotonero, cerezo, manzano, peral, etc., cada una con sus propias características específicas. Dentro de cada especie, existen numerosas variedades adaptadas a diferentes condiciones climáticas y preferencias de los consumidores. Esta información es importante ya que, por ejemplo, los melocotoneros, que florecen temprano en primavera, son susceptibles a las heladas tardías por lo que en zonas propensas a heladas, es aconsejable optar por variedades que florecen más tarde para evitar pérdidas de cosecha o, por otro lado, en algunos mercados, los consumidores prefieren ciertos colores de frutos o ciertas formas de los frutos, como los albaricoques rojos sobre los naranjas tradicionales, por lo que habría que seleccionar variedades a cultivar con esas características. Estas variedades son desarrolladas tanto por centros de investigación públicos como por empresas privadas para satisfacer las demandas del mercado.

En los últimos 27 años, se han registrado un total 1121 variedades de melocotonero, 646 variedades de manzano, 333 variedades de albaricoquero, 158 variedades de ciruelo y 156 variedades de cerezo entre otros, lo que refleja el continuo desarrollo e innovación en este campo [3]. Sin embargo, la información sobre estas variedades se encuentra dispersa en diferentes fuentes, como artículos científicos, folletos comerciales y sitios web de empresas mejoradoras o viveros. Esto dificulta la consulta y comparación de las variedades disponibles, lo que puede resultar tedioso y poco práctico para agricultores, investigadores y otros interesados en el sector.

En este contexto, una aplicación de escritorio en Java con una base de datos SQL podría proporcionar una solución eficiente para recopilar, organizar y buscar información sobre variedades frutales, facilitando así la toma de decisiones informadas en el cultivo, comercialización e investigación de frutas. La aplicación podría actuar como un catálogo y buscador centralizado, permitiendo a los usuarios acceder fácilmente a información detallada sobre diferentes variedades, incluyendo características, condiciones de cultivo, rendimiento, y más. Esto ayudaría a optimizar la producción y comercialización de frutas, fomentando el crecimiento y la innovación en el sector.

1.2. Objetivo

Por ello, se propone desarrollar una aplicación de escritorio que actúe como un catálogo y buscador de variedades frutales, que permita a los usuarios acceder de manera rápida y eficiente a información detallada sobre diferentes especies y variedades de frutas, incluyendo características específicas, condiciones de cultivo, y su disponibilidad en el mercado. La aplicación buscará facilitar la búsqueda y comparación de variedades frutales, contribuyendo así a la promoción de la diversidad y la innovación en el sector frutícola, tanto a nivel nacional como internacional.

1.3. Estudio de mercado

Se ha realizado un estudio de mercado con el objetivo de investigar si existen en la actualidad aplicaciones dedicadas a la catalogación de variedades frutales o si por el contrario la idea propuesta es una oportunidad para el desarrollo de una nueva aplicación que cubra esta necesidad.

En el estudio de mercado realizado, se identificaron varias aplicaciones móviles que, aunque recopilan información sobre plantas, tienen un enfoque principal en la identificación visual a través de fotos como es el caso de PlantSnap o aplicaciones centradas en el registro y gestión de información de explotaciones agrícolas.

En cuanto a aplicaciones dedicadas exclusivamente a la información sobre variedades frutales, se han encontrado dos aplicaciones web; una de ellas, fruitipedia [4], actualmente no está en funcionamiento, y la otra, FruitID [5], es similar a la planteada en este trabajo pero especializada en la identificación de variedades de manzanas y peras.

2. Solución propuesta y requisitos técnicos

2.1. Solución propuesta

El proyecto enfoca la solución del problema de la diversificación de información como una digitalización del proceso para la optimización de las búsquedas proponiendo el desarrollo de una aplicación de escritorio que tenga acceso a la información de las variedades registradas a través de la base de datos relacional.

Las funciones que va a realizar la aplicación son:

- ❖ Login de usuarios: Implementar medidas de seguridad para proteger la integridad de los datos como la autenticación de usuarios. Permite la creación de nuevos usuarios y la restauración de la contraseña.
- ❖ Registro de Información: Permitir a los usuarios (rol administrador) registrar nueva información en la base de datos.
- ❖ Edición de Registros: Permitir a los usuarios (rol administrador) editar la información de los registros existentes para cambiar valores, corregir errores o actualizar detalles.
- ❖ Eliminación de Registros: Proporcionar la funcionalidad para que los usuarios (rol administrador) eliminen registros que ya no son necesarios o que fueron ingresados por error.
- ❖ Visualización de Registros: Mostrar los registros existentes en forma de tabla. Posteriormente, si se selecciona un registro concreto permite a los usuarios ver la información detallada.
- ❖ Búsqueda de Registros: Implementar un sistema de búsqueda que permita a los usuarios buscar registros por diferentes criterios. Esto facilita encontrar información específica dentro de la base de datos.
- ❖ Filtrado de Registros: Implementar un sistema de búsqueda que permita a los usuarios filtrar registros por diferentes criterios. Esto facilita encontrar información específica dentro de la base de datos.
- ❖ Validación de Datos: Validar los datos ingresados por los usuarios para asegurarse de que cumplan con ciertos criterios como primera letra de nombre con mayúscula o formato de correo electrónico antes de ser almacenados en la base de datos.
- ❖ Gestión de Errores: Manejar adecuadamente los errores que puedan surgir durante la interacción con la base de datos o el procesamiento de datos para proporcionar una experiencia de usuario sin problemas.

2.2. Requisitos técnicos

- Entorno de Desarrollo Integrado (IDE): Necesario para desarrollar la aplicación. Se ha elegido NetBeans.
- Java Development Kit (JDK): Paquete de herramientas de desarrollo para crear aplicaciones en el lenguaje de programación Java.
- Base de Datos SQL: Se va a desarrollar una base de datos relacional con el sistema gestor de base de datos MySQL.
- Conector para la Base de Datos: Dado que se va a desarrollar una base de datos MySQL se necesitará el conector MySQL JDBC.

3. Desarrollo de la aplicación

3.1. Planificación

3.1.1. Recursos físicos

El hardware incluye todo el equipo necesario para desarrollar el proyecto. El hardware necesario para el correcto funcionamiento sería:

Un ordenador que disponga de, como mínimo,:

- Una pantalla mínima de 10''.
- Resolución mínima de pantalla de 800x600.
- Teclado y ratón
- Procesador mayor o igual a 1 GHz.
- 2 GB de RAM para 64 bits
- Hasta 20 GB de espacio en disco duro disponible.

3.1.2. Recursos técnicos

El software incluye el sistema operativo necesario para ejecutar la aplicación, así como los programas y herramientas necesarias para su funcionamiento. El software necesario para su correcto funcionamiento sería:

- Sistema operativo Windows 10 64 bits
- Apache NetBeans 13 para Windows (Entorno de Desarrollo Integrado)
- MySQL (Gestor de Base de Datos)
- MySQL Workbench (Herramienta de administración y entorno de desarrollo para trabajar con bases de datos MySQL)
- Base de datos MySQL
- XAMPP 8.0.30 para Windows
- JDK 15 (Java Development Kit)
- my-sql-connector-j-8-2-0-jar (Biblioteca de conexión a Base de Datos)

3.1.3. Recursos humanos

El proyecto lo llevará a cabo una sola persona en calidad de **Full Stack Developer**.

Asumirá los diferentes roles que componen un proyecto de desarrollo de software así como sus funciones:

- Project Manager: Responsable de planificar y coordinar la ejecución del proyecto, asegurando que se cumplan los plazos, presupuestos y objetivos establecidos.
- Desarrollador de Software: Responsable de escribir el código de la aplicación en Java, implementando las funcionalidades requeridas y asegurando su correcto funcionamiento.
- Diseñador de Interfaz de Usuario: Encargado de diseñar la interfaz de usuario de la aplicación, asegurando que sea intuitiva, atractiva y fácil de usar para los usuarios.
- Desarrollador de Base de Datos (SQL): Encargado de diseñar la estructura de la base de datos, crear y mantener las tablas y consultas SQL necesarias para almacenar y gestionar los datos de la aplicación.
- Tester/QA Engineer: Encargado de realizar pruebas de calidad y aseguramiento de la calidad para identificar y corregir errores en la aplicación antes de su lanzamiento.
- Comercial: Encargado de promocionar la aplicación e identificar posibles clientes.

3.1.4. Previsión de carga de trabajo y programación de tareas

Fase	Tarea	Descripción	Tiempo estimado
Fase de planificación	Definición de Requisitos	Revisar el contexto y los objetivos de la aplicación para definir los requisitos funcionales y no funcionales.	1
	Análisis de Usuarios	Identificar los diferentes tipos de usuarios de la aplicación y sus necesidades específicas.	1
	Planificación del Proyecto	Establecer un cronograma detallado de actividades	1
Fase de Diseño	Diseño de la Base de Datos	Diseñar la estructura de la base de datos SQL incluyendo tablas, atributos y relaciones.	6
	Diseño de la Interfaz de Usuario	Diseñar las interfaces que compondrán la aplicación definiendo la disposición de los elementos y la navegación entre las pantallas.	6
	Planificación de la Arquitectura del Software	Diseñar la arquitectura del software definiendo componentes.	6
Fase de Desarrollo	Configuración del Entorno de Desarrollo	Configurar el entorno de desarrollo Java y establecer la conexión con la base de datos SQL.	1
	Implementación de la Lógica	Desarrollar la lógica de la aplicación.	51
	Desarrollo de la Interfaz de Usuario	Desarrollar las diferentes pantallas y componentes.	45
Fase de Pruebas y Depuración	Pruebas Unitarias	Realizar pruebas unitarias para cada componente de la aplicación.	68
	Pruebas de Integración	Probar la aplicación en su conjunto para verificar su funcionamiento adecuado.	4
	Depuración	Corregir cualquier problema identificado durante las pruebas.	3
Fase de Despliegue y Entrega	Embalaje de la Aplicación	Empaquetar la aplicación en un archivo ejecutable.	1
	Documentación	Preparar documentación detallada para los usuarios, incluyendo instrucciones de instalación, guías de uso.	100

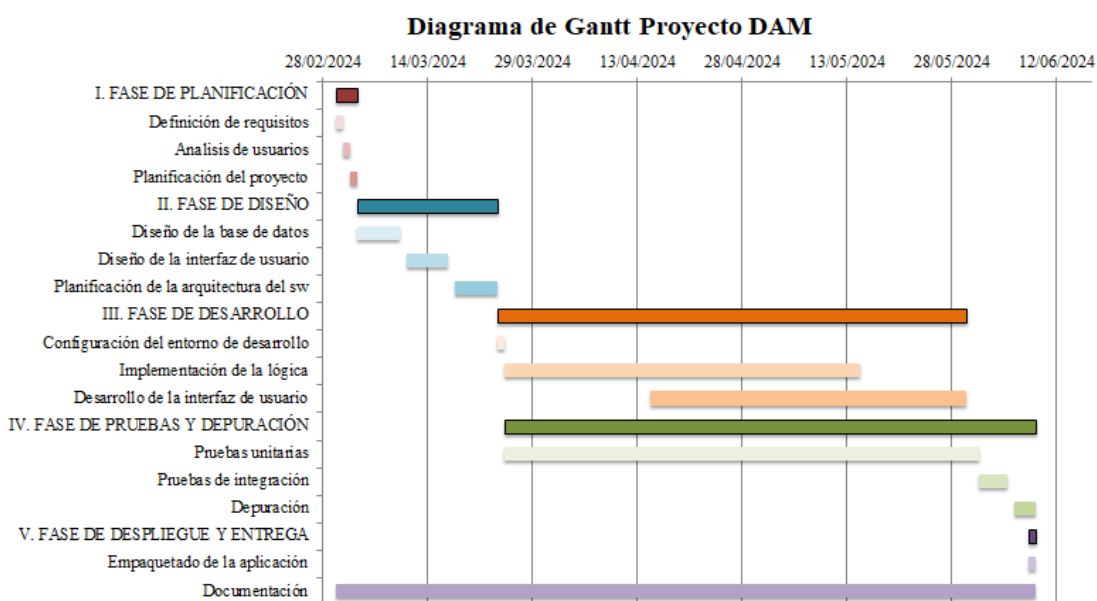


Imagen 1. Diagrama de Gantt

3.2. Análisis de datos y/o procesos

3.2.1. Requisitos Funcionales

- La aplicación debe permitir loguearse a los usuarios.
- La aplicación debe permitir poder cambiar la contraseña al usuario siempre que facilite los datos de control (DNI).
- La aplicación debe permitir crear usuarios.
- La aplicación debe indicar que usuario está conectado en todo momento en el caso de que se haya logueado.
- La aplicación debe permitir a los usuarios buscar variedades frutales por especie y/o nombre.
- Los usuarios deben poder ver información detallada sobre cada variedad frutal, incluyendo especie, nombre, floración o región geográfica.
- La aplicación debe permitir a los usuarios aplicar filtros adicionales a los resultados de búsqueda para refinarlos según sus preferencias.
- La aplicación debe permitir crear nuevas variedades siempre que el usuario sea el administrador y actualizar la base de datos.
- La aplicación debe permitir modificar los datos de variedades siempre que el usuario sea el administrador y actualizar la base de datos.
- La aplicación debe permitir eliminar variedades siempre que el usuario sea el administrador y actualizar la base de datos.

3.2.2. Requisitos No Funcionales

- La aplicación debe garantizar la integridad de los datos.
- La interfaz de usuario debe ser intuitiva y fácil de usar, garantizando una experiencia de usuario positiva y satisfactoria.
- La aplicación debe tener un rendimiento óptimo, con tiempos de respuesta rápidos y sin retrasos significativos al cargar datos o realizar búsquedas.
- Se deben implementar medidas de seguridad para proteger la integridad de los datos de la aplicación y la privacidad de la información de los usuarios.
- El código de la aplicación debe estar bien estructurado y documentado, facilitando su mantenimiento y futuras actualizaciones.
- La aplicación debe ser escalable, es decir, que permita un crecimiento futuro en el volumen de datos y usuarios sin comprometer su rendimiento o funcionalidad.

3.2.3. Análisis de datos

Se han identificado las diferentes fuentes de datos de los que se obtendrá la información para alimentar la aplicación. Se extraerán de forma manual de páginas web, folletos comerciales y publicaciones científicas.

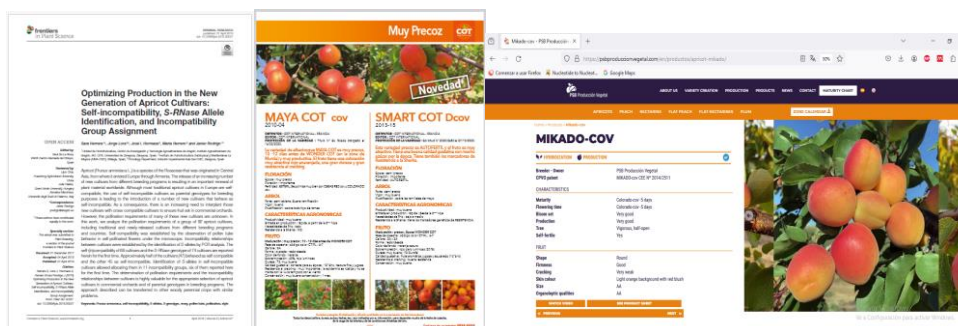


Imagen 2. Fuentes de información.

Con todo ello, una vez obtenidos los datos, se evalúa su calidad para identificar inconsistencias o deficiencias y se planifican las estrategias para abordarlos.

En base a la relevancia de los datos disponibles para los objetivos y requisitos de la aplicación, se identifican los datos más importantes y de interés.

Se observa que existen datos comunes a las diferentes especies frutales por lo que en la estructura de la base de datos se plantea una relación de herencia simulada guardando en una tabla “padre” los datos comunes y en tablas “hijas” las características específicas de cada especie frutal. Todo ello se volcará en una base de datos relacional.

En cuanto a los datos de los usuarios, se ha decidido que inicien sesión utilizando como ‘usuario’ el email introducido al crear el perfil. Además, es necesario que introduzcan su DNI en el registro ya que se utiliza como campo validador seguro a la hora de hacer un cambio de contraseña.

3.2.4. Análisis de Procesos

Casos de uso

Iniciar sesión usuario		
Precondición	El usuario tiene que estar registrado.	
Descripción	El usuario valida su nombre de usuario y contraseña para iniciar sesión.	
Actores	Usuario	
Secuencia normal	Paso	Acción
	1	El usuario ingresa a la aplicación y selecciona el botón ‘Iniciar sesión’.
	2	La aplicación abre la ventana de login.
	3	El usuario escribe los datos para el nombre de usuario y la contraseña y da click en el botón ‘Ingresar’.
	4	La aplicación comprueba que el formato del email es válido.
	5	La aplicación realiza una consulta en la base de datos utilizando los criterios de búsqueda proporcionados.
	6	La aplicación comprueba que la contraseña introducida por pantalla es igual que la contraseña devuelta en la consulta para ese determinado usuario.
	7	En las pantallas principal, resultados, nueva ficha varietal y nuevo usuario aparece una etiqueta con el nombre del usuario logueado. Además, aparecen los botones ‘Crear nueva ficha varietal’, ‘Modificar’, ‘Eliminar’ y ‘Cerrar sesión’.
Excepciones	Paso	Acción
	5	El usuario introducido no existe en la base de datos por lo que se lanza un aviso por pantalla.
	6	La contraseña introducida no coincide con la contraseña guardada para ese usuario por lo que se lanza un aviso por pantalla.

Crear usuario		
Precondición	El usuario tiene que estar logueado (caso de uso 'Iniciar sesión usuario').	
Descripción	Un usuario crea un perfil de usuario.	
Actores	Usuario	
Secuencia normal	Paso	Acción
	1	El usuario selecciona el botón 'Nuevo usuario' ubicado en la pantalla principal.
	2	La aplicación abre la ventana de nuevo usuario.
	3	El usuario rellena todos los campos y selecciona el botón 'Crear usuario'
	4	La aplicación comprueba que el formato del email es válido.
	5	La aplicación comprueba que el formato de contraseña es válido.
	6	La aplicación realiza una consulta en la base de datos creando un nuevo usuario.
Excepciones	Paso	Acción
	4	El formato del email no es válido por lo que se lanza un aviso por pantalla.
	5	El formato de la contraseña no es válido por lo que se lanza un aviso por pantalla.

Restaurar contraseña		
Precondición	El usuario tiene que estar registrado.	
Descripción	Un usuario cambia su contraseña.	
Actores	Usuario	
Secuencia normal	Paso	Acción
	1	El usuario ingresa a la aplicación y selecciona el botón 'Iniciar sesión'.
	2	La aplicación abre la ventana de login.
	3	El usuario hace click en el botón '¿Has olvidado la contraseña?'
	4	La aplicación abre la ventana de cambio de contraseña.
	5	El usuario escribe los datos para el nombre de usuario, DNI y la nueva contraseña y da click en el botón 'Restaurar'.
	6	La aplicación comprueba que el formato del email es válido.
	7	La aplicación realiza una consulta en la base de datos utilizando los criterios de búsqueda proporcionados.
	8	La aplicación comprueba que el DNI introducido por pantalla es igual que el DNI devuelto en la consulta para ese determinado usuario.
	9	La aplicación comprueba que el formato de contraseña es válido.
	10	La aplicación realiza una consulta en la base de datos y actualiza la contraseña guardada.
Excepciones	Paso	Acción
	6	El formato del email no es válido por lo que se lanza un aviso por pantalla.
	7	El usuario introducido no existe en la base de datos por lo que se lanza un aviso por pantalla.
	8	El DNI introducido no coincide con el DNI guardado para ese usuario por lo que se lanza un aviso por pantalla.
	9	El formato la contraseña no es válida por lo que se lanza un aviso por pantalla.

Buscar Variedad y listar		
Precondición		
Descripción	El usuario realiza una búsqueda de variedades, por especie o especie y nombre, y los resultados se visualizan en un listado.	
Actores	Usuario	
Secuencia normal	Paso	Acción
	1	El usuario puede seleccionar o no una especie de la lista desplegable (combobox).
	2	En el caso de que el usuario haya seleccionado una especie, puede introducir el nombre en la caja de texto para realizar una búsqueda concreta.
	3	La aplicación realiza una consulta en la base de datos y lista los resultados de la búsqueda en una tabla en la ventana resultados.
Excepciones	Paso	Acción
	2	El usuario introduce el nombre de una variedad pero no ha seleccionado a qué especie pertenece en el combobox por lo que aparecerá un mensaje de aviso por pantalla.
	3	No existen registros que cumplan con los criterios de búsqueda aplicados por lo que aparecerá un mensaje de aviso por pantalla.

Ver Detalles de Variedad Frutal		
Precondición	El listado de variedades debe contener registros.	
Descripción	El usuario visualiza información detallada sobre una variedad frutal específica.	
Actores	Usuario	
Secuencia normal	Paso	Acción
	1	El usuario selecciona una variedad frutal de la lista de resultados de búsqueda.
	2	La aplicación muestra información detallada sobre la variedad frutal seleccionada, incluyendo Nombre, Floración, Zona Geográfica, Color/Forma.

Filtrar Resultados de Búsqueda		
Precondición	El usuario ha realizado una búsqueda de variedades frutales en la pantalla principal.	
Descripción	El usuario aplica filtros adicionales a los resultados de búsqueda para refinarlos.	
Actores	Usuario	
Secuencia normal	Paso	Acción
	1	El usuario aplica filtros adicionales, como floración, color o forma, seleccionando en los combobox.
	2	La aplicación realiza una consulta en la base de datos y actualiza los resultados de la búsqueda en la tabla para reflejar los filtros aplicados.
Excepciones	Paso	Acción
	2	No existen registros que cumplan con los filtros aplicados por lo que aparecerá un mensaje de aviso por pantalla.

Crear Variedad		
Precondición	El usuario tiene que estar logueado (caso de uso ' <i>Iniciar sesión usuario</i> ').	
Descripción	El administrador de la aplicación actualiza la base de datos con una nueva variedad frutal introduciendo los datos mediante la interfaz de la aplicación.	
Actores	Administrador	
Secuencia normal	Paso	Acción
	1	El administrador accede a la interfaz 'Nueva ficha varietal' de la aplicación.
	2	El administrador introduce los datos de nuevas variedades frutales o actualiza la información existente en la base de datos.
	3	La aplicación valida y procesa los datos cargados, actualizando la base de datos.
Excepciones	Paso	Acción
	3	El formato del nombre no es válido por lo que se lanza un aviso por pantalla.

Modificar Variedad		
Precondición	El usuario tiene que estar logueado (caso de uso ' <i>Iniciar sesión usuario</i> ').	
Descripción	El administrador de la aplicación actualiza la base de datos modificando los datos de una variedad mediante la interfaz de la aplicación.	
Actores	Administrador	
Secuencia normal	Paso	Acción
	1	El administrador selecciona una variedad frutal de la lista de resultados de búsqueda.
	2	El administrador modifica la información existente en la base de datos.
	3	La aplicación valida y procesa los datos cargados, actualizando la base de datos.
Excepciones	Paso	Acción
	3	El formato del nombre no es válido por lo que se lanza un aviso por pantalla.

Eliminar Variedad		
Precondición	El usuario tiene que estar logueado (caso de uso ' <i>Iniciar sesión usuario</i> ').	
Descripción	El administrador de la aplicación actualiza la base de datos eliminando una variedad frutal mediante la interfaz de la aplicación.	
Actores	Administrador	
Secuencia normal	Paso	Acción
	1	El administrador selecciona una variedad frutal de la lista de resultados de búsqueda.
	2	El administrador selecciona el botón 'Eliminar'.
	3	La aplicación valida y procesa la consulta actualizando la base de datos.
Excepciones	Paso	Acción
	2	El usuario selecciona el botón eliminar pero no ha seleccionado ninguna variedad por lo que se lanza un aviso por pantalla.

3.3. Diseño de datos y/o procesos

3.3.1. Diseño de datos

Entidades:

- Frutal: Representa el concepto de especies frutales. Engloba en una entidad las características comunes a todas ellas con atributos como nombre y floración. La zona geográfica se ha planteado como un atributo lista de strings.
- Albaricoquero/Almendro: Representa especies específicas de frutales con atributos específicos de cada una de ellas como forma en el caso de los almendros o color en el caso de los albaricoqueros.
- Usuario: Representa un usuario de la aplicación, con atributos como nombre, email o contraseña, etc. Existe un tipo de rol dentro de los usuarios: 'admin'.

Relaciones:

- Una variedad frutal pertenece a una especie frutal específica.
- Un frutal puede estar ubicado en diferentes zonas geográficas
- Una misma zona geográfica puede contener diferentes variedades frutales.

Atributos de las Entidades:

- Frutal: Especie, Nombre, Floración, Zona geográfica.
- Albaricoquero/Almendro: Color/Forma.
- Usuario: Nombre, Apellidos, Departamento, Puesto, Correo Electrónico, Contraseña y Rol.

Estructura de la Base de Datos:

- Tabla "Frutal": Contiene las características comunes de especies de árboles frutales.
Campos: ID (Clave Primaria), nombre, especie, floración.
- Tabla "Albaricoquero": Contiene las características específicas de esta especie de frutal.
Campos: ID (Clave Primaria) y color.
- Tabla "Almendro": Contiene las características específicas de esta especie de frutal.
Campos: ID (Clave Primaria) y forma.
- Tabla "usuario": Para guardar la información de los usuarios registrados.
Campos: id_usuario (Clave Primaria), dni, nombre, apellidos, departamento, puesto, email, contrasena y rol.
- Tabla "ZonaGeografica": Contiene los nombres de las diferentes zonas.
Campos: ID (Clave Primaria) y nombre.
- Tabla "Frutal_Zona": Es una tabla que relaciona la zona geográfica en la que se encuentra cada frutal. Para ello contiene utiliza las claves primarias de ambas tablas.
Campos: FrutalID (Clave Primaria) y ZonalID (Clave Primaria).

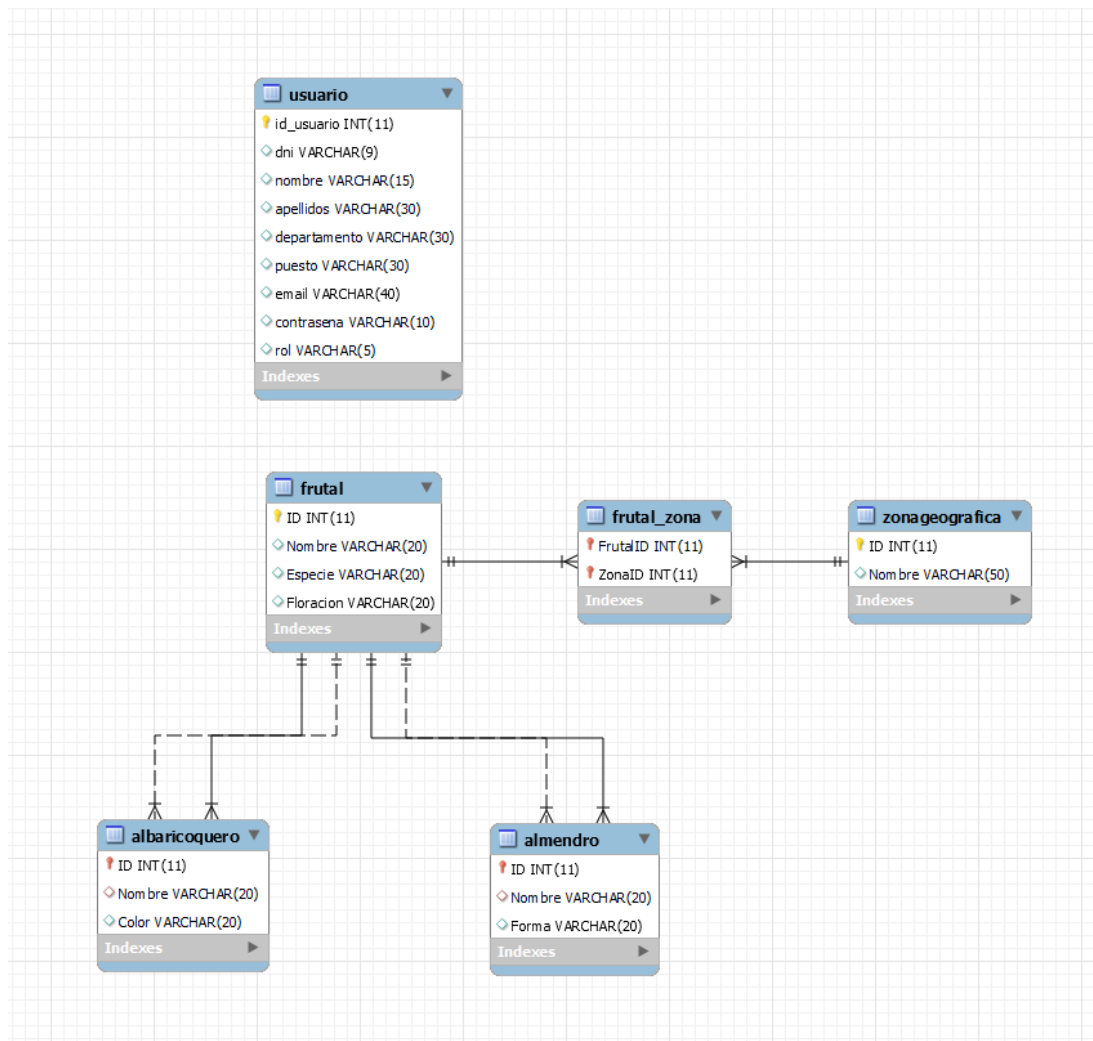


Imagen 3. Diagrama base de datos

3.3.2. Diseño de Procesos

Se han identificado los diferentes procesos dentro de la aplicación:

1. Registro y Autenticación de Usuarios. Proceso para que los usuarios se registren en la aplicación proporcionando un nombre de usuario, correo electrónico y contraseña. También incluye un proceso de autenticación para verificar las credenciales de inicio de sesión.
2. Búsqueda de Variedades Frutales. Proceso para que los usuarios busquen variedades frutales utilizando diferentes criterios como especie y/o nombre.
3. Filtrado de Resultados de Búsqueda. Proceso para que los usuarios apliquen filtros adicionales a los resultados de búsqueda para refinarlos según sus preferencias.
4. Visualización de Detalles de Variedad Frutal. Proceso para que los usuarios vean información detallada sobre una variedad frutal específica seleccionada de los resultados de búsqueda.
5. Actualización de la Base de Datos. Proceso para que el administrador de la aplicación actualice la base de datos con nuevas variedades frutales o información actualizada. Permite crear, modificar y eliminar registros.

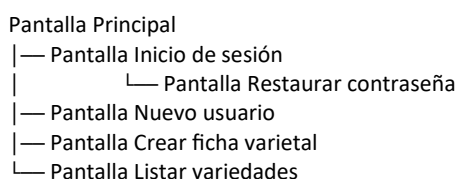
3.4. Diseño de interfaz

El objetivo del diseño de la interfaz de la aplicación es proporcionar a los usuarios una experiencia intuitiva y atractiva para consultar la información de frutales durante su rutina de trabajo. Para ello, la interfaz debe ser fácil de usar, simple y accesible.

Los usuarios son trabajadores de 18 a 65 años de centros de investigación. Son personal formado en el campo de estudio por lo que están familiarizados con los datos de la aplicación. A nivel informático, son usuarios estándar.

Estructura de navegación

Se presenta un mapa de navegación en el cual aparecen cada una de las vistas.



Para facilitar la navegación entre las diferentes ventanas, se han introducido componentes para volver a la pantalla anterior.

Usabilidad

- Simplicidad

Se ha priorizado la simplicidad en el diseño de la interfaz por lo que los elementos funcionales han sido reducidos a lo esencial para evitar la sobrecarga de información y la confusión del usuario.

- Consistencia

Se ha mantenido una consistencia visual y funcional en toda la aplicación mediante la estandarización de los patrones de diseño, la disposición de los elementos y las interacciones del usuario (desarrollado en el siguiente punto 'Guías de Estilo').

- Retroalimentación

Se han introducido mensajes de aviso en la aplicación para darle feedback al usuario sobre sus acciones.

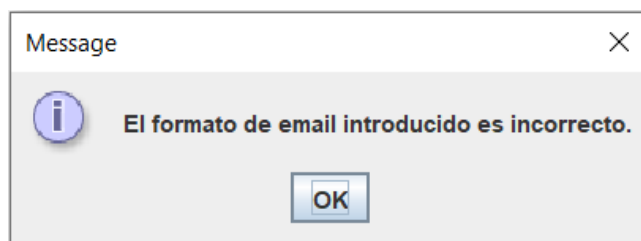


Imagen 4. Ejemplo mensaje de aviso

- Directrices de interacción

Los botones cambian a un tono más claro al ser seleccionados.

Se han introducido tooltip text para facilitar la introducción de datos del usuario. Indican el tipo de dato a introducir en las cajas de texto.



Imagen 5. Ejemplo tooltip

Guías de Estilo

- Gama de Colores

Se ha establecido una paleta de colores consistente en todas las vistas de la aplicación destacando los verdes y blancos en diferentes tonalidades.

- Cabecera

En cada vista de la aplicación, se ha incluido una cabecera en la parte superior. Esta cabecera contiene la funcionalidad de esa vista resumida en pocas palabras para facilitar al usuario una identificación rápida de la pantalla en la que se encuentra.

- Imagen de Fondo

Se ha seleccionado una misma imagen de fondo para todas las vistas de la aplicación. Esta imagen ha sido por ser representativa de la funcionalidad de la aplicación además de bonita.

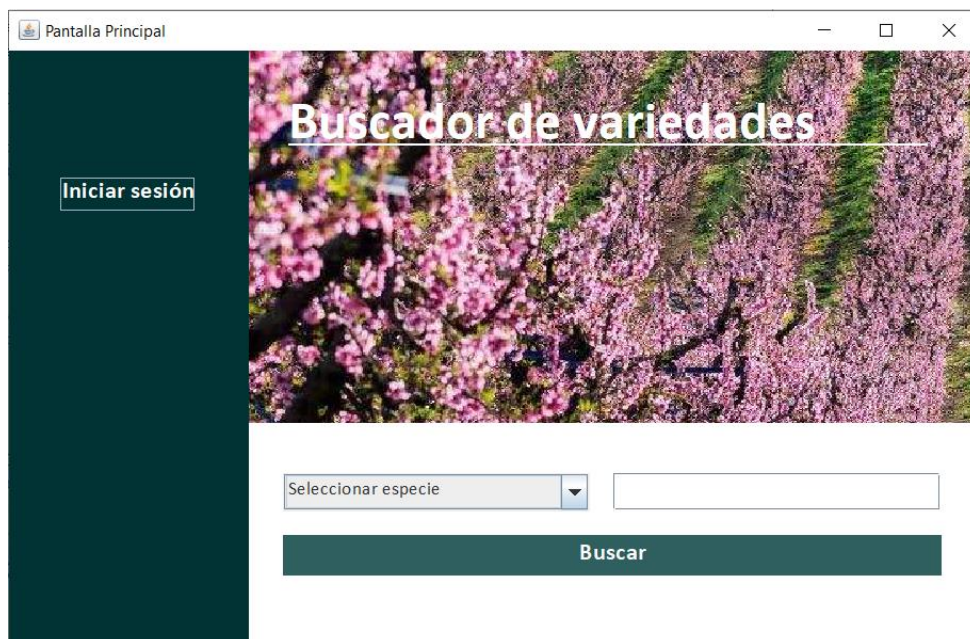


Imagen 6. Ejemplo guía de estilos de la aplicación

3.5. Diseño de arquitectura

La metodología de desarrollo que se ha elegido ha sido el modelo en V. Este modelo se caracteriza por ser una metodología de software en la que cada una de las fases del ciclo de vida del proyecto sirve para realimentar otras fases y de esta forma poder crear un producto software robusto y completo. Para ello, existe una simetría entre procesos de producción y de verificación.

Los patrones de diseño utilizados han sido:

3.5.1. Patrón Modelo-Vista-Controlador (MVC)

La aplicación está dividida en 3 componentes principales (en el proyecto se han materializado en paquetes): el Modelo que contiene la lógica de la aplicación, la Vista que contiene las interfaces con las que el usuario interactúa y el Controlador que actúa como intermediario entre Modelo y Vista. Dentro de Controlador, se han creado dos controladores. Por un lado, para la lógica del catálogo y buscador de variedades frutales y, por otro, para la gestión de los usuarios.

Dentro de la gestión de usuarios, se ha planteado un rol que actúa como administrador ('admin'). La aplicación puede ser usada por los usuarios sin loguearse. Sin embargo, en el caso de que se logueen, la aplicación ofrecerá más funcionalidades. El rol 'admin' es el único permitido para crear, modificar o eliminar los datos de las fichas varietales así como para crear nuevos usuarios que por defecto se crearán con el rol 'admin'. También permite la opción de restaurar la contraseña en el caso de que se haya olvidado o se quiera modificar por otra utilizando como variable de comprobación el DNI del usuario.

3.5.2. Patrón Data Access Object (DAO)

Este patrón se utiliza para separar la lógica de acceso a datos de la lógica de negocio de una aplicación mediante una capa de abstracción que facilita el intercambio de datos sin afectar al resto de la aplicación. En el proyecto se ha implementado mediante las clases nombreDAO y nombreDAOImpl para cada una de las entidades definidas.

La clase nombreDAO actúa como interfaz del objeto de acceso a datos (DAO) y contiene los métodos declarados mientras que la clase nombreDAOImpl es una implementación de la interfaz NombreDAO y contiene la lógica de los métodos. Esta clase contiene las consultas específicas para acceder y manipular la base de datos.

4. Pruebas

Durante el desarrollo de la lógica de la aplicación, se llevaron a cabo pruebas unitarias complementarias para asegurar que cada componente funcionara correctamente. Estas pruebas se centraron en verificar individualmente que las funciones de la aplicación operaran según lo esperado. Además se forzaron errores para comprobar que los mensajes planteados actuaban correctamente.

Una vez que la aplicación estuvo completamente desarrollada, se realizaron pruebas de integración. Estas pruebas aseguraron que la información introducida por pantalla y las consultas a la base de datos ocurrían sin problemas. En estas pruebas, se comprobó que los datos ingresados por los usuarios se almacenaban o recuperaban correctamente, y que la aplicación en su conjunto funcionaba de manera coherente y eficiente.

5. Conclusiones

Planificación

El proyecto se desarrolló en base a una planificación detallada que se cumplió en su mayoría. Aunque surgieron algunas nuevas funcionalidades que ajustaron ligeramente los tiempos, la mayoría de las tareas se realizaron según lo programado.

Aprendizaje

A través de este proyecto, he aprendido a plantear un proyecto de desarrollo de software desde el inicio manejando las diferentes fases. Todo ello me ha permitido aplicar los conocimientos aprendidos en las diferentes asignaturas del ciclo de grado superior. He desarrollado habilidades en programación en Java, en creación y diseño de bases de datos con SQL y en aplicar características de la programación orientada a objetos como son la encapsulación, la herencia, la modularidad o la sobrecarga de métodos.

Dificultades

Durante el desarrollo las dificultades encontradas fueron:

- Gestión de la Herencia: Manejar la herencia de clases en Java fue complejo, especialmente al integrar funcionalidades que dependían de los niveles de herencia.
- Relaciones Muchos a Muchos en SQL: Implementar y manejar relaciones muchos a muchos en las bases de datos fue complicado, especialmente al formular consultas eficientes.
- Manejo de Filtros: Permitir que los usuarios pudieran aplicar múltiples filtros simultáneamente requirió construir consultas dinámicas y flexibles.

Mejoras

El sistema desarrollado es escalable y permite futuras mejoras. Algunas de ellas podrían ser:

- Nuevas Funcionalidades: Se pueden añadir nuevas características para mejorar la experiencia del usuario y la funcionalidad del software.
- Nuevas especies frutales: Se pueden añadir nuevas especies con características propias.
- APIs: Desarrollar APIs para facilitar la extracción automatizada de datos, reduciendo la necesidad de entrada manual y mejorando la eficiencia del sistema.

6. Presupuesto

Se ha llevado a cabo el presupuesto del proyecto basándose en las siguientes características:

- Desarrollador de software Junior
- Promedio de 2 horas de trabajo diario.
- 35 €/hora [6].

Fase del Proyecto	Horas	Coste (€)
Fase de Planificación	3	105
Fase de Diseño	18	630
Fase de Desarrollo	97	3.395
Fase de Pruebas y Depuración	75	2.625
Fase de Despliegue y Entrega	101	3.535
Total del Proyecto	294	10.290

Se ha considerado añadir un margen de contingencia del 10% para posibles imprevistos.

Al utilizarse herramientas de desarrollo de software libre y tener el equipo de hardware amortizado, solo se aplicará una cuantía en forma de porcentaje añadido al total del coste del proyecto, como forma de sufragar el deterioro del material por uso. Este porcentaje se establece en un 3%.

Total del presupuesto:

Recursos humanos	Recursos tecnológicos	Contingencias	Total
10.290 €	308,7 €	1.029 €	11.627,7 €
IVA (21%)			2.441,8 €
Precio final			14.069,5 €

7. Bibliografía

1. Artículo de prensa.

<https://www.expansion.com/especiales/ifema/2022/10/04/633adc5de5fdea7a5b8b464f.html>

2. Sandoval Martínez-Abarca, J. (2021). Diagnóstico y análisis estratégico del sector de la fruta en España. Tesis Doctoral. Universidad Pontificia de Comillas.

<https://repositorio.comillas.edu/xmlui/bitstream/handle/11531/46867/TFG-%20SANDOVAL%20MARTINEZ-ABARCA%2C%20JOAQUIN.pdf?sequence=1#:~:text=>

3. Community Plant Variety Office (CPVO). <https://cpvo.europa.eu/en>

4. Fruitipedia. <https://fruitipedia.com/>

5. FruitID. <https://www.fruitid.com/#main>

6. Página web. <https://www.laps4.com/preguntas-y-respuestas/cuanto-cuesta-una-hora-de-programacion>