

Enlace a GitHub del proyecto:

https://github.com/Jorge-bg-05/Proyecto_DAM

Programación:

Este proyecto se enfoca en una aplicación de consola en Java llamada TaskPlan, que tiene como objetivo principal ayudar a los usuarios en la administración de sus tareas académicas. Es un recurso práctico pero sencillo para planificar el estudio, gestionar plazos de entrega y clasificar las tareas según su dificultad y prioridad.

Funciones clave:

- **Gestión de usuarios:** Todos los usuarios tienen la posibilidad de registrarse e ingresar con una contraseña. Los datos de los usuarios se almacenan en un archivo binario.
- **Creación de tareas:** Es posible incluir nuevas tareas especificando su materia, título, nivel de dificultad (alta, media o baja) y la fecha límite.
- **Listado y eliminación:** El usuario tiene la posibilidad de ver todas sus tareas almacenadas y eliminar aquellas que ya estén caducadas o finalizadas.
- **Ordenación por prioridad:** Las actividades se clasifican considerando el tiempo restante para la entrega y el nivel de dificultad.
- **Filtrado por días:** Es posible filtrar las tareas que expiran en los siguientes X días.

- **Tareas urgentes:** Muestra únicamente las tareas que expiran en los siguientes 3 días.
- **Modo estudio 100%:** Aplica diferentes métodos de trabajo dependiendo del tipo de tarea:
 - Si es un deber, utiliza Time Blocking para dividir ejercicios por días.
 - Si se trata de un examen, aplica la técnica Pomodoro para estructurar tus sesiones de estudio.

Estructura del código:

- El archivo TaskPlan.java incluye el flujo principal del programa y todas las alternativas del menú.
- En Usuarios.java se maneja la lógica de los usuarios y sus archivos de tareas.
- En Tarea.java se encuentran las clases Tarea, Deberes y Exámenes, que realizan la implementación de la interfaz MetodoDeTrabajo para llevar a cabo estrategias de estudio.

Observaciones:

Esta aplicación me ha servido para practicar distintos conceptos clave de Java como:

- Gestión de archivos (RandomAccessFile, serialización de objetos).
- Aplicación de fechas (SimpleDateFormat, Calendar).

- Estructuras de datos como ArrayList y Hashtable.
- Clases, objetos y herencia.

Bases de datos:

Elección BBDD:

Aunque el programa implementa almacenamiento mediante archivos con RandomAccessFile, he elegido el uso de Apache Derby como base de datos embebida para futuras versiones. Derby permite integrar una base de datos directamente dentro de la aplicación Java sin necesidad de instalar gestores externos, facilitando portabilidad y mantenimiento.

Entre sus **principales ventajas** destacan:

- Integración nativa con Java.
- No requiere configuración cliente-servidor.
- Soporte completo para SQL (DDL, DML, DCL).
- Ideal para aplicaciones de escritorio donde la base de datos no necesita estar en red.

1. Implementación, Explotación y Mantenimiento:

1.1. DDL:

```

^Cciclosm@pc-21:~$ ij
ij version 10.17
ij> CONNECT 'jdbc:derby:miBD;create=true';
ij> CREATE TABLE Usuarios (
    id INT PRIMARY KEY GENERATED ALWAYS AS IDENTITY,
    nombre VARCHAR(100) NOT NULL UNIQUE,
    contrasena VARCHAR(255) NOT NULL,
    archivo_tareas VARCHAR(255),
    ultimo_acceso TIMESTAMP
);

CREATE TABLE Asignaturas (
    id INT PRIMARY KEY GENERATED ALWAYS AS IDENTITY,
    nombre VARCHAR(100) NOT NULL UNIQUE
);

CREATE TABLE Tareas (
    id INT PRIMARY KEY GENERATED ALWAYS AS IDENTITY,
    nombre VARCHAR(100) NOT NULL,
    dificultad VARCHAR(10) NOT NULL CHECK (dificultad IN ('alta', 'media', 'baja')),
    fecha_entrega TIMESTAMP NOT NULL,
    tipo VARCHAR(20) NOT NULL CHECK (tipo IN ('DEBER', 'EXAMEN')),
    id_asignatura INT NOT NULL,
    id_usuarios INT NOT NULL,
    CONSTRAINT FK1_Tareas_Usuarios FOREIGN KEY(id_usuarios) REFERENCES Usuarios(id),
    CONSTRAINT FK2_Tareas_Asignaturas FOREIGN KEY(id_asignatura) REFERENCES Asignaturas(id)
);

CREATE TABLE Deberes (
    id_tarea INT PRIMARY KEY,
    cantidad_ejercicios INT NOT NULL,
    CONSTRAINT FK2_Deberes_Tarea FOREIGN KEY(id_tarea) REFERENCES Tareas(id)
);

CREATE TABLE Examenes (
    id_tarea INT PRIMARY KEY,
    temas_examen INT NOT NULL,
    CONSTRAINT FK3_Examenes_Tarea FOREIGN KEY(id_tarea) REFERENCES Tareas(id) ON DELETE CASCADE
);
> > > > 0 rows inserted/updated/deleted
ij> > > > 0 rows inserted/updated/deleted
ij> > > > > > > > 0 rows inserted/updated/deleted
ij> > > > 0 rows inserted/updated/deleted
ij> > > > 0 rows inserted/updated/deleted
ij> INSERT INTO Usuarios (nombre, contrasena) VALUES
('Juan', '123456'),
('Ana', '654321');

```

CREATE TABLE Usuarios (

id INT PRIMARY KEY GENERATED ALWAYS AS IDENTITY,

nombre VARCHAR(100) NOT NULL UNIQUE,

contrasena VARCHAR(255) NOT NULL,

archivo_tareas VARCHAR(255),

ultimo_acceso TIMESTAMP

);

CREATE TABLE Asignaturas (

id INT PRIMARY KEY GENERATED ALWAYS AS IDENTITY,

```

    nombre VARCHAR(100) NOT NULL UNIQUE
);

CREATE TABLE Tareas (
    id INT PRIMARY KEY GENERATED ALWAYS AS IDENTITY,
    nombre VARCHAR(100) NOT NULL,
    dificultad VARCHAR(10) NOT NULL CHECK (dificultad IN ('alta', 'media', 'baja')),
    fecha_entrega TIMESTAMP NOT NULL,
    tipo VARCHAR(20) NOT NULL CHECK (tipo IN ('DEBER', 'EXAMEN')),
    id_asignatura INT NOT NULL,
    id_usuarios INT NOT NULL,
    CONSTRAINT FK1_Tareas_Usuarios FOREIGN KEY(id_usuarios)
REFERENCES Usuarios(id),
    CONSTRAINT FK2_Tareas_Asignaturas FOREIGN
KEY(id_asignatura) REFERENCES Asignaturas(id)
);

```

```

CREATE TABLE Deberes (
    id_tarea INT PRIMARY KEY,
    cantidad_ejercicios INT NOT NULL,
    CONSTRAINT FK2_Deberes_Tarea FOREIGN KEY(id_tarea)
REFERENCES Tareas(id)
);

```

```

CREATE TABLE Examenes (
    id_tarea INT PRIMARY KEY,
    temas_examen INT NOT NULL,

```

CONSTRAINT FK3_Examenes_Tarea **FOREIGN KEY**(id_tarea)
REFERENCES Tareas(id) ON DELETE CASCADE

);

1.1.1. Normalización hasta 3FN

1FN:

id	nombre	dificultad	fecha_entrega	tipo	asignatura
1	Deber 1	alta	2025-06-01 12:00	DEBER	História
2	Examen 1	media	2025-06-03 08:00	EXAMEN	Matemáticas
3	Deber 2	baja	2025-06-04 13:00	DEBER	Programación

En la tabla cada campo tiene un único valor(no hay listas, repeticiones ni grupos de datos dentro de un mismo campo), por lo que cumple la 1FN.

2FN:

Para pasar a 2FN se eliminan las dependencias parciales. El nombre de la asignatura no pertenece a la tarea, si no que es información propia de la asignatura

por lo que se crea una tabla independiente llamada Asignaturas evitando así los datos redundantes. Además, se observa que una o varias tareas pertenecen a un usuario, por lo que se establece una relación con la tabla Usuarios mediante el campo Usuarios

Tabla Asignaturas:

id	nombre
1	Historia
2	Matemáticas

Tabla Tareas:

id	nombre_tarea	dificultad	fecha_entrega	tipo	id_asignatura	id_usuario
1	Deber 1	alta	2025-06-01 12:00	DEBER	1	1
2	Examen 1	media	2025-06-03 08:00	EXAMEN	2	1
3	Deber 2	baja	2025-06-04 13:00	DEBER	1	2

3FN:

Para cumplir con la 3FN, se deben eliminar las dependencias transitivas, es decir, que ningún atributo no clave dependa de otro atributo no clave.

En este caso:

- Todos los atributos de Tareas dependen únicamente de su clave primaria (id).
- id_asignatura y id_usuario son claves foráneas correctamente referenciadas.
- No hay atributos que dependan de otros atributos no clave.

Por lo tanto, el modelo cumple con la Tercera Forma Normal (3FN).

Además, como cada tarea puede ser de tipo DEBER o EXAMEN, y cada tipo tiene atributos específicos, se crean dos tablas adicionales relacionadas:

Tabla Deberes

id_tarea	cantidad_ejercicios
1	25
3	10

Tabla Exámenes

id_tarea	temas_examen
2	3

Tabla Usuarios

id	nombre	contraseña
1	Juan	123456
2	Ana	654321

1.2. DML (Inserción de datos, búsqueda de información, CHECKS, donde se gestionan...):

Insertar usuario:

```
Nombre de usuario: Juan
Usuario nuevo. Crea una contraseña: 123456
✓ Usuario registrado correctamente.
No hay tareas guardadas todavía.
```

Crear tarea:

```
Opciones:
1. Crear tarea
2. Listar tareas
3. Eliminar tarea
4. Ordenar por prioridad
5. Filtrar por proximidad de entrega(próximo x días)
6. Tareas urgentes(próximo 3 días)
7. Modo Estudio 100%
8. Salir
1

Creando tarea...
Ingrese la dificultad de la tarea (Alta/Media/Baja):
Alta
Ingrese el nombre de la asignatura:
Matemáticas
Ingrese el nombre de la tarea:
Examen
Ingrese la fecha de entrega (dd/MM/yyyy HH:mm):
21/11/2025 12:00
```

```
Creando tarea...
Ingrese la dificultad de la tarea (Alta/Media/Baja):
Media
Ingrese el nombre de la asignatura:
Programación
Ingrese el nombre de la tarea:
Ejercicios arrays
Ingrese la fecha de entrega (dd/MM/yyyy HH:mm):
14/06/2025 23:55
```

Eliminar Tareas:

```
Eliminando tarea...
Tarea: Examen
¿Ha completado esta tarea? (s/n):
n
Tarea: Ejercicios arrays
¿Ha completado esta tarea? (s/n):
s
Tarea eliminada: Completada

Opciones:
1. Crear tarea
2. Listar tareas
3. Eliminar tarea
4. Ordenar por prioridad
5. Filtrar por proximidad de entrega(próximo x días)
6. Tareas urgentes(próximo 3 días)
7. Modo Estudio 100%
8. Salir
2

Listando tareas...
Asignatura: Matemáticas
Nombre de la tarea: Examen
Dificultad: alta
Fecha de entrega: Fri Nov 21 12:00:00 CET 2025
-----
```

Ordenar tareas por orden de dificultad y fecha de entrega:

```
Ordenando tareas por prioridad...

--- TAREAS ORDENADAS POR PRIORIDAD ---
Asignatura: Base de Datos
Tarea: Crear DDL
Dificultad: BAJA
Fecha de entrega: 15/06/2025 13:30
-----
Asignatura: Proyecto Intermodular
Tarea: Crear modelo de negocio
Dificultad: ALTA
Fecha de entrega: 16/06/2025 12:30
-----
Asignatura: Sistemas Informáticos
Tarea: Configurar pc
Dificultad: BAJA
Fecha de entrega: 16/06/2025 23:55
-----
Asignatura: Matemáticas
Tarea: Examen
Dificultad: ALTA
Fecha de entrega: 21/11/2025 12:00
-----
```

Filtrar tareas por proximidad en los siguientes x días:

```
Filtrando tareas por proximidad de entrega...
¿Cuántos días hacia adelante quieres ver las tareas?
4

--- TAREAS EN LOS PRÓXIMOS 4 DÍAS ---
Asignatura: Base de Datos
Tarea: Crear DDL
Dificultad: BAJA
Fecha de entrega: 15/06/2025 13:30
-----
```

Mostrar las tareas urgentes:

```
Mostrando tareas urgentes...
🔴 TAREAS URGENTES (menos de 3 días):
Asignatura: Base de Datos
Tarea: Crear DDL
Dificultad: BAJA
Fecha de entrega: 15/06/2025 13:30
-----
Asignatura: Programación
Tarea: Ejercicios streams
Dificultad: MEDIA
Fecha de entrega: 14/06/2025 23:55
-----
```

Consultar todas las tareas de un usuario:

```
Listando tareas...
Asignatura: Matemáticas
Nombre de la tarea: Examen
Dificultad: alta
Fecha de entrega: Fri Nov 21 12:00:00 CET 2025
-----
Asignatura: Sistemas Informáticos
Nombre de la tarea: Configurar pc
Dificultad: baja
Fecha de entrega: Mon Jun 16 23:55:00 CEST 2025
-----
Asignatura: Base de Datos
Nombre de la tarea: Crear DDL
Dificultad: baja
Fecha de entrega: Sun Jun 15 13:30:00 CEST 2025
-----
Asignatura: Proyecto Intermodular
Nombre de la tarea: Crear modelo de negocio
Dificultad: alta
Fecha de entrega: Mon Jun 16 12:30:00 CEST 2025
-----
Asignatura: Programación
Nombre de la tarea: Ejercicios streams
Dificultad: media
Fecha de entrega: Sat Jun 14 23:55:00 CEST 2025
-----
```

Validaciones (CHECK):

Las tablas tienen restricciones CHECK para evitar insertar datos inválidos:

dificultad VARCHAR(10) **CHECK** (dificultad IN ('alta', 'media', 'baja')),

tipo VARCHAR(20) **CHECK** (tipo IN ('DEBER', 'EXAMEN'))

Esto impide, por ejemplo, registrar una dificultad no válida como 'extrema'.

Por ahora los datos se manejan en la aplicación mediante archivos con RandomAccessFile, pero en un futuro se van a integrar en una base de datos relacional como Apache Derby, usando SQL desde Java con JDBC para insertar, consultar, y gestionar los datos de manera más estructurada y eficiente.

1.3. “Acceso a datos”, Integración con la aplicación, “Código embebido”, otras opciones...

Acceso actual:

Actualmente, el acceso a los datos en la aplicación se realiza utilizando archivos binarios de acceso aleatorio (RandomAccessFile). Esto permite guardar, leer y modificar tareas sin necesidad de una base de datos relacional.

Cada vez que el usuario crea, consulta o elimina una tarea, la aplicación lo hace directamente sobre el archivo proximasEntregas.dat.

Estructura de acceso a datos desde el Java (Código embebido):

El acceso está embebido dentro del propio código Java. Es decir, no se utiliza aún una base de datos externa ni librerías como JDBC.

El tratamiento de los datos se realiza directamente desde métodos como crearTarea(), listarTareas() y eliminarTarea(), que escriben o leen campos binarios (Strings y fechas) en el fichero.

Integración con base de datos:

En un futuro, se sustituirá el uso de RandomAccessFile por una base de datos embebida como Apache Derby.

Para acceder a la base de datos embebida desde la aplicación Java, se usaría JDBC con la siguiente lógica básica:

```
Connection conn =  
DriverManager.getConnection("jdbc:derby:miBD;create=true");
```

```
PreparedStatement ps = conn.prepareStatement(  
    "INSERT INTO tareas (id_usuarios, id_asignatura, nombre,  
    dificultad, fecha_entrega, tipo) VALUES (?, ?, ?, ?, ?, ?)");
```

```
ps.setInt(1, usuarioid);  
ps.setInt(2, asignaturald);  
ps.setString(3, "Nombre de la tarea");  
ps.setString(4, "media");  
ps.setTimestamp(5, Timestamp.valueOf("2025-06-15 10:00:00"));  
ps.setString(6, "DEBER"); // o "EXAMEN"
```

Esta integración permite insertar, actualizar y consultar datos desde el código fuente, manteniendo el almacenamiento de forma estructurada y persistente.

1.4. DCL (usuarios, copias de seguridad, etc...)

Apache Derby en modo embebido puede gestionar permisos y seguridad mediante configuraciones adicionales de autenticación.

Para copias de seguridad, se puede utilizar el procedimiento integrado:

```
CALL SYSCS_UTIL.SYSCS_BACKUP_DATABASE('ruta/de/backup');
```

También se pueden realizar copias manuales de la carpeta del sistema de archivos que contiene la base de datos ('miBD/').

En modo cliente/servidor, Derby permite roles, autenticación de usuarios y logs de acceso para mayor seguridad y control.

Sistemas Informáticos:

Mi proyecto se basa en una aplicación en crecimiento, es decir, aún es un prototipo que no incluye muchas de las funcionalidades que incluirá al año siguiente, por lo que necesito una máquina que sea estable, potente, y que si en un futuro necesito aún más potencia sea totalmente escalable.

Caja de PC o servidor:

MSI MAG FORGE M100R Cristal Templado USB 3.2 ARGB Negra

<https://www.pccomponentes.com/msi-mag-forge-m100r-cristal-templado-usb-32-argb-negra>

Compatible con placas Micro ATX y Mini-ITX, lo que permite montar sistemas más compactos sin poner en riesgo el rendimiento. Dispone de dos puertos USB 2.0, un USB 3.2 Gen 1 y dos conectores jack de audio de 3.5 mm (entrada y salida) en el panel superior. Soporta hasta seis ventiladores: 3 frontales y 1 trasero, ya instalados, y capacidad para 2 más en la parte superior, todos de un diámetro máximo de 120 mm, aunque añadiendo uno más en la parte superior junto a los ya instalados es suficiente para una refrigeración adecuada. Soporta disipadores de CPU de hasta 160 mm y tarjetas gráficas de hasta 300 mm, lo cual permite usar componentes potentes. En cuanto a almacenamiento, la cantidad de HDDs o SSDs que soporta es de 3x2,5" + 1x3,5", o de 2x2,5" + 2x3,5".

Precio: 49,95 €

Fuente de alimentación:

Hiditec RL550 550W 80 Plus Bronze

<https://www.pccomponentes.com/hiditec-rl550-550w-80-plus-bronze>

Proporciona la suficiente energía como para alimentar una CPU potente, una GPU y almacenamiento adicional. Cuenta con la certificación 80+ Bronze por lo que garantiza hasta un 88% de eficiencia energética, reduciendo el consumo energético y el calor. Incorpora un ventilador de 120 mm silencioso. Su potencia total es de 550W, suficiente para alimentar un sistema con un procesador moderno, tarjeta gráfica dedicada de gama media (como una RTX 3060), varias unidades de almacenamiento y ventiladores adicionales.

Incluye 1 cable ATX de 24 pines (20 + 4) para la placa base, 8 conectores de energía SATA, 1 conector de alimentación EPS de 8 pines (4 + 4) para CPU, 1 conector de poder PCIe 6+2 pines para GPU y 3 conectores Molex de 4 pines para conectar 3 ventiladores adicionales. Las funciones de protección que tiene son: Sobretensión, Sobrevoltaje, Cortocircuito y Bajo Voltaje.

Precio: 43,31 €

Placa base:

MSI B550M PRO-VDH WIFI

https://www.pccomponentes.com/msi-b550m-pro-vdh-wifi?s_kwcid=AL!14405!3!599658646701!!!g!!&gad_source=1&gad_campaignid=917843211&gclid=CjwKCAjw6NrBBhB6EiwAvnT_rpGibWpjH9LaBXbzhdu3zQv_ODLE3kyj9JnDxTejLrlcPTN0dUaH7RoC4gkQAvD_BwE

La torre admite placas Micro-ATX, y esta placa es formato Micro-ATX. La placa tiene Socket AM4 y chipset B550, compatible con AMD Ryzen 3ª, 4ª y 5ª gen (si es necesario se actualiza la BIOS). Tiene 1 ranura de almacenamiento M.2 PCIe 4.0 x4, conectada a la CPU, y otra M.2 PCIe 3.0 x4, conectada al chipset, más 4 puertos SATA 3 (6 Gbps). Tiene un PCIe 4.0/3.0 x16, perfecto para una GPU moderna y es compatible con el tamaño máximo que permite la torre (355 mm). Incluye sensores y software

para controlar ventiladores, y es compatible con controladoras ARGB.

Soporta memoria DDR4 hasta 3200 MT/s (1600 MHz de reloj real) con configuración Dual Channel. En modo overclock hasta 4400 MT/s (2200 MHz de reloj real)

Puertos y expansiones:

4 slots de RAM (hasta 128 GB DDR4): Útil por si es necesario ampliar en el futuro.

2 slots PCIe 3.0 x1 adicionales: Para tarjetas de captura, sonido, etc.

Varios puertos USB:

7x USB 3.2 Gen1 (6 tipo A y 1 tipo C)

6x USB 2.0

Incluye un módulo inalámbrico Intel® Wireless-AC 3168 que proporciona Wi-Fi y Bluetooth 4.2.

También incorpora conectividad por cable LAN (Ethernet/RJ45) que soporta velocidad de hasta 1 Gbps.

Por último, incluye un sistema de sonido de alta definición de 8 canales (7.1). Además incorpora la tecnología Audio Boost de MSI, que mejora la calidad de sonido.

Precio: 111,31 €

Microprocesador:

AMD Ryzen 5 5600XT 3.7/4.7GHz

<https://www.pccomponentes.com/procesador-amd-ryzen-5-5600xt-37-47ghz?srsId=AfmBOoozYUN8sYmefxvGBL6FkniKwERQQmXGinVa1hTywva0VhZERdHw>

Serie: Ryzen 5000 Series

Arquitectura: Zen 3

Número de núcleos de CPU: 6

Número de subprocesos: 12

Frecuencia de reloj base: 3.7 GHz

Frecuencia de reloj máxima: 4.7 GHz

Caché L1: 384 KB

Caché L2: 3 MB

Caché L3: 32 MB

Potencia de Diseño Térmico (TDP): 65W

Tecnología de fabricación:

- Núcleos de CPU: 7nm
- Chip de Entrada/Salida: 12nm

Socket de CPU: AM4 (Compatible con la placa base elegida)

Chipsets soportados: X570 , X470 , X370 , B550 (soportado por la placa base), B450 , B350 , A520

Temperatura de funcionamiento máxima: 95°C

Soporte para conectar 2 unidades SATA al sistema

Compatible con PCIe 4.0

El total de carriles PCIe 4.0 usables por el procesador son 24: 20 para GPU, SSD NVMe..., y 4 para la conexión entre procesador y chipset. En este caso la placa tiene acceso solo a 20 carriles PCIe

4.0 usables por el procesador: 1 ranura PCIe x16 4.0 para GPU conectada directamente al procesador, y 1 ranura PCIe x4 4.0 para M.2 NVMe. Las otras ranuras PCIe se usan a través del chipset.

Compatible con RAID0, RAID1 Y RAID10.

Memoria máxima soportada 128 GB (DDR4)

Soporta memoria DDR4 hasta 3200 MT/s (1600 MHz de reloj real)

Precio: 145,82 €

Disipador-Ventilador para la CPU:

Forgeon Solarian Cooler 4Pipes 120mm ARGB Ventilador CPU Blanco

<https://www.pccomponentes.com/forgeon-solarian-cooler-4pipes-120mm-argb-ventilador-cpu-blanco>

Sockets compatibles:

- Intel: Socket LGA 775/1155/1156/1366/2011/1200/1700 (CPU Core i3 / i5 / i7/i9)
- AMD: AM5/ AM4 / FM2 / FM1 / AM3 + / AM3 / AM2 + / AM2 / AM5

Por lo tanto es compatible con el microprocesador elegido

Dimensiones: 158mm x 122mm x 76mm

Velocidad máxima: de 800 RPM a 1800 RPM

Vida Estimada de uso: 50.000h

Tubos de disipador de Calor: 4 Tubos

Potencia de Diseño Térmico (TDP): max 130W, suficiente para el microprocesador elegido cuya TDP es de 65W

Voltaje: 12V

Conectores: 12v alimentación 4pin (PWM), lo que significa que la

placa base puede controlar la velocidad del ventilador según la temperatura que tenga la CPU

Precio: 30,00 €

Pasta térmica:

Noctua NT-H1 3.5 g

<https://noctua.at/es/nt-h1-3-5g/specification>

Es una pasta térmica confiable, muy estable y de larga duración, que tiene una buena conductividad térmica, 8.5 W/mK, y es fácil de aplicar. Además, es compatible con el socket AM4, y no tiene conductividad eléctrica, por lo que si se derrama fuera del chip no habría riesgo de cortocircuito.

Precio: 9,95 €

Memoria RAM:

Corsair Vengeance RGB Pro Optimizado AMD DDR4 3200 16GB 2x8GB CL16

https://www.corsair.com/es/es/p/memory/cmw16gx4m2c3200c16/vengeance-rgb-pro-16gb-2-x-8gb-ddr4-dram-3200mhz-c16-memory-kit-a-black-cmw16gx4m2c3200c16?srsId=AfmBOopyaBn8rZPMP_OyZkRh9_PBlzPE-MQjMVpUNUnt8TFndPPTWagtJ#tab-overview

16 GB de memoria DDR4, cuya velocidad de reloj es de 3200 MT/s (1600 MHz). Dos módulos de memoria de 8 GB cada uno y voltaje de 1.35 V. Latencia CL16, una buena latencia para estas velocidades. Para que la memoria funcione a su 100% hay que activar XMP en la BIOS, una tecnología de Intel, compatible con AMD, permite que la placa configure automáticamente la RAM a 3200 MT/s con sus latencias y voltaje correctos.

Número de pines: 288

Totalmente compatible con AM4.

Precio: 48,99 €

Tarjeta gráfica PCI-Express x16:

Gigabyte GeForce RTX 3060 GAMING OC 12GB GDDR6 Rev 2.0
https://www.pccomponentes.com/gigabyte-geforce-rtx-3060-gaming-oc-12gb-gddr6-rev-20?srsItd=AfmBOor1dJgsILyI7QkG6d0udm2DnLKo5Q3qX_vypz2Fts_fKpbja8qp

12 GB de VRAM, una gran ventaja para aplicaciones profesionales que sean muy exigentes, y tanto el uso de GDDR6, como un ancho de banda amplio (360 GB/s max) garantizan velocidad al manejar texturas grandes, varias pantallas o resoluciones altas.

Ancho de datos: 192 bit

Velocidad de memoria del reloj: 15000 MHz

CUDA Cores: 3584, cuanto más núcleos tenga, mayor capacidad de procesamiento paralelo, ideal para edición de video, renderizado, IA y simulaciones, y también para videojuegos, por lo que es ideal para tareas como Adobe Premiere, Blender, etc.

Tipo de interfaz: PCIe 4.0 x16 lo que asegura máxima compatibilidad y velocidad con placas como la mía y garantiza que la GPU no genere cuello de botella.

Número de puertos HDMI: 2

Versión HDMI: 2.1 (permite salida 4K a 120 Hz)

Cantidad de DisplayPorts: 2

Versión de DisplayPort: 1.4a

Cuenta con una tecnología de enfriamiento que tiene 3 ventiladores que garantizan una temperatura controlada y menor ruido, incluso

aunque esté bajo una gran carga, lo que permite mantener sesiones largas de alto rendimiento estable.

Precio: 289,99 €

Disco duro:

Seagate Barracuda ST1000DM014 3.5" 1TB SATA 3

https://www.pccomponentes.com/disco-duro-seagate-barracuda-st1000dm014-35-1tb-sata-3?s_kwcid=AL!14405!3!599658692862!!!g!!&gad_source=1&gad_campaignid=917843211&gclid=Cj0KCQjwgIXCBhDBARIsAELC9ZiYdZgZXrcTj5s5B5kS_3c-yCKB67v1Vn2CjIK3YsS_-P5J6bTeg0laAt2JEALw_wcB

Tamaño: 3.5"

Capacidad: 1 TB (1000 GB)

Velocidad de rotación: 7200 RPM

Interfaz: Serial ATA III (SATA 3)

Caché o almacenamiento de búfer: 256 MB

Número de cabezales en el HDD: 2

Promedio de latencia: 6 ms

Ciclo comenzar/detener: 600000 ciclos, por lo que tiene una buena vida útil en la que soporta múltiples encendidos y apagados sin fallar.

Temperatura operativa de 0°C a 60°C

Es un disco bastante resistente:

Vibraciones soportadas: 0,25 G

Golpes en funcionamiento: 80 G

Golpes fuera de operación: 800 G

Elegido como unidad secundaria para guardar archivos grandes como backups, videos, fotos, archivos que se usen poco o proyectos que no necesiten mucha velocidad.

Precio: 55,99 €

Unidad SSD:

Samsung 980 SSD 1TB PCIe 3.0 NVMe M.2

https://www.pccomponentes.com/samsung-980-ssd-1tb-pcie-30-nvme-m2?srsId=AfmBOoqpWIsYYB61uiAh0tPMyhJEAMvBpPr7ApUoXXse777rO-lv22p_

Capacidad: 1 TB (1000 GB)

Factor de forma de disco SSD: M.2

Interfaz: PCIe 3.0 x4 NVMe, compatible con la placa base que soporta PCIe 3.0 x4

Velocidad de lectura: 3500 MB/s

Velocidad de escritura: 3000 MB/s

NVMe versión 1.4: mejoras en eficiencia energética y latencia

Lectura aleatoria (4KB): 500000 IOPS

Escritura aleatoria (4KB): 480000 IOPS

Voltaje de operación: 3,3 V

Consumo de energía (max): 5,3 W

Consumo de energía (promedio): 4,6 W

Tiempo medio entre fallos: 1500000 h

Algoritmos de seguridad soportados: 256-bit AES

He elegido este SSD como unidad principal por su alta velocidad de lectura y escritura (hasta 3500/3000 MB/s), su gran capacidad de almacenamiento y su excelente rendimiento en acceso aleatorio, lo que permite arranques muy rápidos, cargas instantáneas de programas y fluidez general del sistema.

Gracias a su interfaz PCIe 3.0 x4 y tecnología NVMe, supera por mucho a los discos SATA y es totalmente compatible con la placa base elegida. Además, cuenta con gestión térmica avanzada y soporte de cifrado por hardware, lo que lo convierte en una opción moderna, segura y eficiente para el uso diario y tareas exigentes.

Precio: 86,99 €

Grabadora DVD-BluRay:

No hace falta incluir una grabadora de DVD o Blu-ray porque la aplicación no usa discos ni ningún tipo de medio óptico. Todo se guarda directamente en el disco duro o en un SSD mediante archivos binarios de acceso aleatorio.

Sistema operativo:

Los requisitos que debe cumplir el sistema operativo para esta app son:

- Compatibilidad con Java (JDK 8 o superior)
- Buena gestión de archivos y carpetas locales
- Interfaz de consola estable y funcional
- Facilidad de uso y portabilidad para el usuario final

Por eso he elegido Windows, concretamente Windows 10 o 11, por su excelente compatibilidad con Java. Además, el método de almacenamiento utilizado en la aplicación, `RandomAccessFile`,

funciona sin restricciones sobre el sistema de archivos de Windows (NTFS). También cuenta con consolas integradas fáciles de usar como CMD, PowerShell o Windows Terminal.

Otra ventaja es que permite empaquetar la aplicación como .jar o incluso .exe y ejecutarla sin complicaciones, lo que lo hace ideal para estudiantes o usuarios con menos experiencia técnica.

Software aplicación:

La aplicación se llama TaskPlan y ha sido desarrollada en lenguaje Java. Es un programa en línea de comandos orientado a la gestión de tareas académicas que permite guardar distintos usuarios, y cada usuario puede almacenar x número de tareas, ya sean del tipo exámen o deberes. Además cada usuario puede crear, eliminar, listar tareas, filtrarlas por orden de prioridad según su dificultad y fecha de entrega, etc.

Utiliza archivos binarios (.dat) y la clase RandomAccessFile para guardar la información de forma persistente en el disco, sin necesidad de base de datos, aunque en un futuro se planea emigrar a una base de datos como Apache Derby.

El usuario interactúa mediante menús en consola, lo que hace que sea una aplicación ligera, funcional y multiplataforma, siempre que se disponga del JDK instalado.

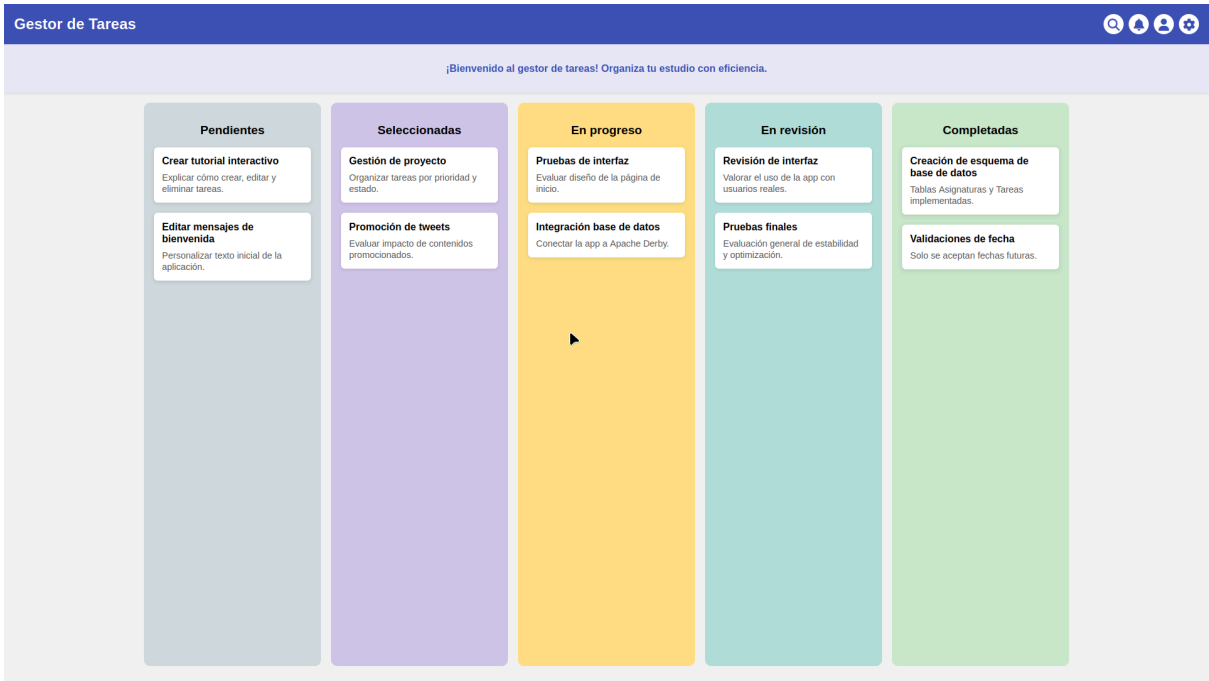
Equipamiento de red para su conectividad con Internet:

No he elegido ninguno ya que la placa base ya dispone de un módulo inalámbrico Intel® Wireless-AC 3168 que proporciona Wi-Fi y Bluetooth 4.2. Además, incorpora conectividad por cable LAN (Ethernet/RJ45) que soporta velocidad de hasta 1 Gbps.

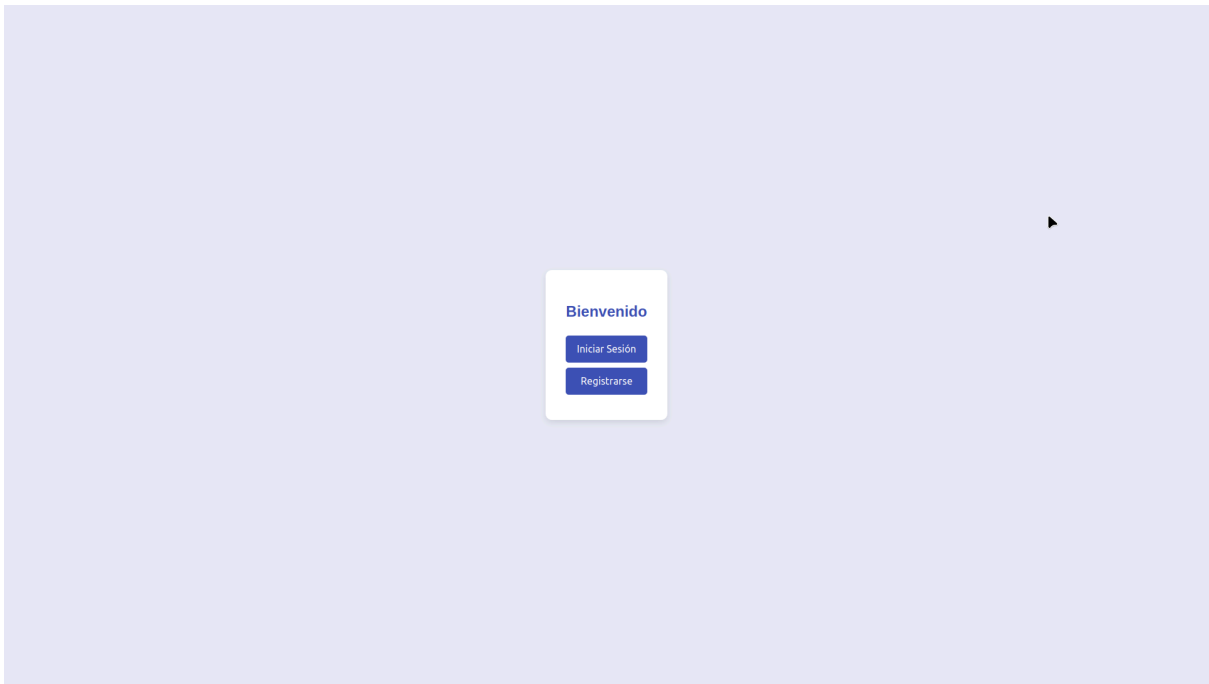
Precio Total: 872,30 €.

Lenguaje de Marcas:


Página Inicial:



Página para iniciar sesión o registrarse:




Formulario para iniciar sesión:




A login form titled "Iniciar Sesión" is centered on a light gray background. It contains two input fields: "Usuario o correo electrónico" and "Contraseña". A blue arrow points from the first field to the second. Below the fields is a blue button labeled "Entrar".

Si los campos están vacíos los marca en rojo:



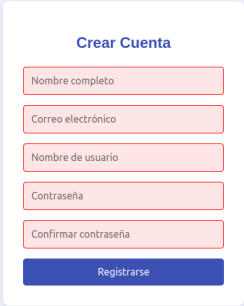
The same login form as above is shown, but the input fields "Usuario o correo electrónico" and "Contraseña" are highlighted with a red border, indicating they are empty. The blue arrow and the "Entrar" button remain the same.

Formulario para registrarse:



A registration form titled "Crear Cuenta" is centered on a light blue background. The form is a white card with a blue border. It contains five input fields, each with a label: "Nombre completo", "Correo electrónico", "Nombre de usuario", "Contraseña", and "Confirmar contraseña". All fields are filled with text. Below the fields is a blue button labeled "Registrarse". A mouse cursor is hovering over the button.

Si los campos están vacíos los marca en rojo:



The same registration form is shown, but all five input fields are highlighted with a red border, indicating they are empty. The "Registrarse" button remains blue and is still being hovered over by a mouse cursor.

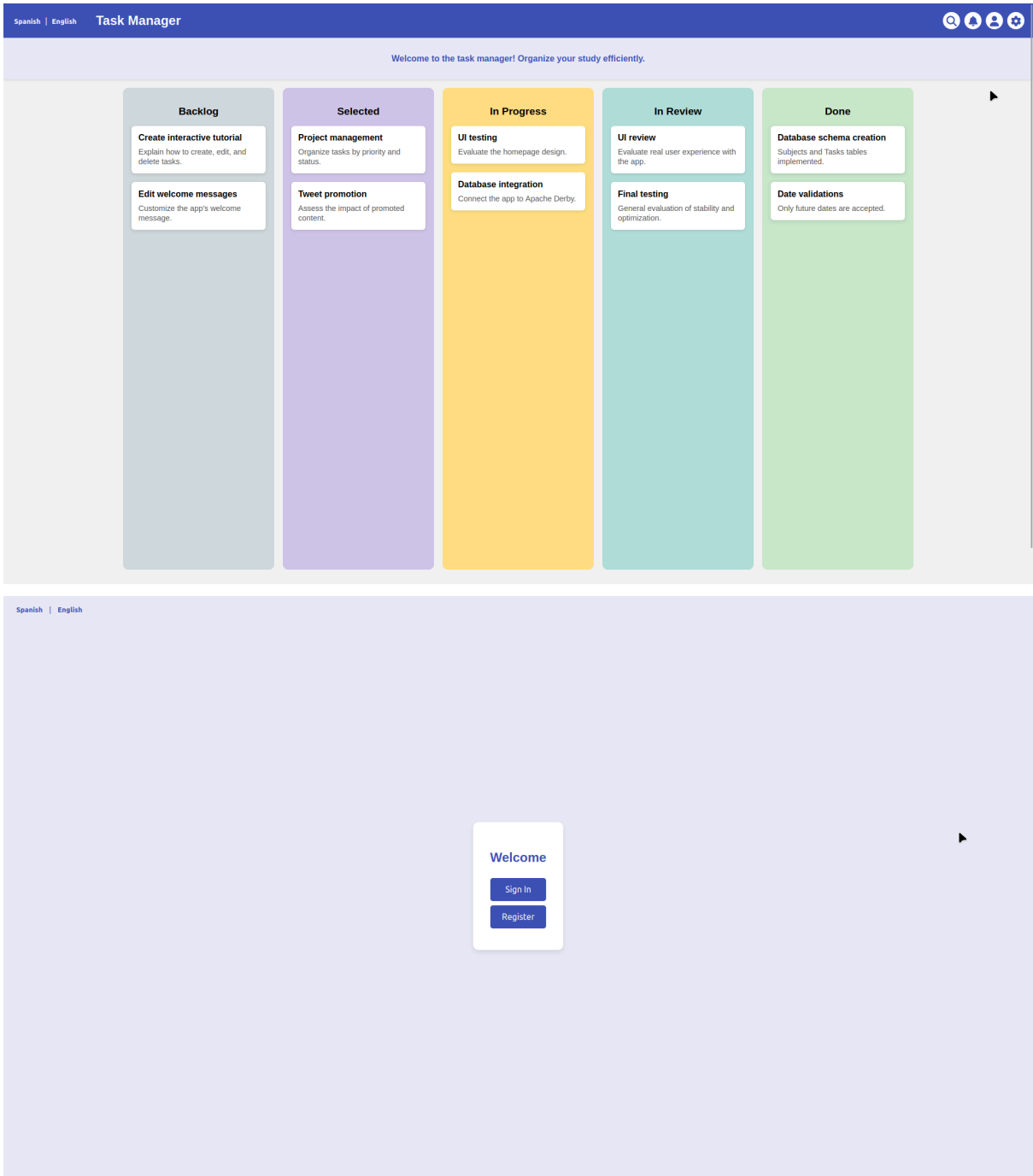
En todas las páginas he agregado dos botones (Español | Inglés o Spanish | English) para cambiar entre Español e Inglés.

[Spanish](#) | [English](#)

[Español](#) | [Inglés](#)

Inglés:

He traducido la página web al inglés:



Create Account

Register

Sign In

Login

Entornos de Desarrollo:

He creado un repositorio en Git Hub, en el que he invitado de colaborador al tutor, donde he subido el código Java de mi aplicación junto a la memoria.

