

Reorganización completa del proyecto

Sin eliminar ningún archivo actual interno (aunque se propongan carpetas nuevas)

Reubicación, no eliminación: conservar todo el contenido actual y reubicarlo dentro de una estructura ordenada y centralizada

## ESQUEMA COMPLETO Y PROPUESTA DE ORGANIZACIÓN

Proyecto: PassQuirk RPG (Discord Bot)

### 1. ESTADO ACTUAL DEL PROYECTO

Se identifican tres zonas principales:

#### 1.1 Código fuente del bot:

Disperso entre:

Info-Bot/

bot/

PassQuirk-RPG-Official/

Raíz del proyecto (archivos sueltos .js, .ts, etc.)

#### 1.2 Interfaz gráfica modular (panels y embeds):

Cada módulo está en carpetas independientes:

battle-panel/

character-creation-panel/

discord-embeds/

dungeon-panel/

embeds/

inventory-panel/

tournament-panel/

#### 1.3 Documentación y análisis:

Archivos Markdown como:

PLAN\_MAESTRO\_PASSQUIRKRPG.md

README.md

ANALISIS\_PANELES\_V0DEV

Mezclados con código

### 2. OBJETIVO

Organizar todo el proyecto para que:

Todo el código fuente esté centralizado

Toda la documentación esté separada

Los assets y referencias de diseño estén bien localizados

NO se elimine código existente dentro de las carpetas

Slash commands estén ordenados y funcionales

### 3. PROPUESTA DE ORGANIZACIÓN

Esta estructura propone mover y organizar sin borrar contenido interno existente.

plaintext

Copiar

Editar

PassQuirk-RPG/

```
    └── docs/          # Documentación del proyecto
        ├── sistema-de-lucha.md
        ├── PLAN_MAESTRO_PASSQUIRKRPG.md
        ├── ANALISIS_PANELES_V0DEV.md
        └── estructura-futura.md
            └── futuro/      # (Reemplazo planeado de la web "web-preview")

    └── src/           # Código fuente central del bot
        ├── commands/      # Todos los slash commands ordenados
            ├── combate/
            ├── administración/
            ├── en_revisión/
            └── futuros/

        ├── core/          # Lógica y controladores del bot (mover desde "Info-Bot" y "bot")
        ├── config/         # Archivos de configuración
        ├── db/             # json.sqlite u otro almacenamiento
        ├── embeds/         # Embeds generales
        ├── utils/          # Funciones utilitarias
        ├── events/         # Manejadores de eventos (por ejemplo: onInteraction)
        └── index.ts        # Entrypoint principal del bot

    └── panels/         # Todos los paneles de UI (sin borrar código existente)
        ├── battle-panel/   # Mover o vincular como subcarpeta
        ├── character-creation-panel/
        ├── dungeon-panel/
        ├── embeds/          # Visuales para embeds
        ├── inventory-panel/
        ├── tournament-panel/
        └── discord-embeds/

    └── assets/          # Música, sprites, imágenes
        ├── sprites/
        ├── sonidos/
        └── diseño/

    └── public/          # Archivos estáticos (frontend o CDN)
        ├── styles/         # CSS / Tailwind
        └── .env             # Variables de entorno
```

```
└── package.json  
└── tsconfig.json  
└── tailwind.config.ts  
└── README.md  
└── .gitignore
```

#### 4. ACCIONES A REALIZAR

Reorganización interna

Fusionar Info-Bot/ + PassQuirk-RPG-Official/ → mover contenido a src/

Mantener todos los archivos y subcarpetas. NO eliminar ningún archivo dentro de carpetas como:

battle-panel/, inventory-panel/, etc.

Renombrar web-preview/ como docs/futuro/ (no eliminarlo, solo reubicarlo)

Slash Commands

Eliminar los slash commands actuales solo desde la terminal (los que están desplegados en Discord, no el código)

Crear carpeta única: src/commands/ con subcategorías:

combate/, admin/, en\_revision/, futuros/

Mover todos los comandos funcionales y limpiar los rotos (marcarlos en revisión, no borrarlos aún)

Limpieza lógica (sin eliminar aún)

Carpeta / Archivo Acción sugerida

web-preview/ Mover a docs/futuro/

config/ y lib/ Fusionar en src/config/

hooks/, .cursor/ Marcar para revisión futura

#### 5. NOTA CLAVE PARA EL EQUIPO

No eliminar contenido existente aún.

Todos los archivos, incluso si están duplicados o antiguos, deben mantenerse y solo moverse a la estructura adecuada.

Marcar archivos desactualizados como en\_revision, legacy, o deprecated, pero no borrarlos sin revisión.

¿Quieres que esto lo empaquete ahora en un documento PDF limpio para compartir con tu agente de IA y equipo? ¿O prefieres que genere un script base mover\_archivos.sh que automatice parte de esta reorganización?

Tú dijiste:

Si

ChatGPT Plus

Memoria guardada actualizada

Perfecto. A continuación te presento el esquema completo del estado actual del proyecto PassQuirk

RPG, junto con una propuesta detallada de reestructuración del proyecto que:

Fusiona correctamente la lógica, diseño y documentación.

No elimina código existente.

Centraliza y organiza los módulos funcionales.

Deja todos los archivos anteriores intactos.

Prepara el entorno para que el bot esté bien estructurado, limpio y mantenible.

#### **ESQUEMA ACTUAL DEL PROYECTO**

Archivos sueltos (raíz del proyecto)

Contiene scripts, herramientas, backups y entradas principales del bot:

index.js, start.js, server.js, main-bot-manager.js, etc.

Backups: package.json.backup

Otros: LICENSE, README.md, .env, pnpm-lock.yaml, tsconfig.json

Problema: muchos scripts sueltos sin clasificar. Slash commands desorganizados. Archivos duplicados y poco claros.

Carpetas Clave actuales

Carpeta Uso actual Problemas o tareas

Info-Bot Contiene doc y parte del bot Mezcla de doc + código

PassQuirk-RPG-Official Contiene diseño funcional de paneles Muy útil, pero no está fusionado con el resto

bot Contiene comandos