

# SpiritPlanner – Project Status

## Estado general

El proyecto se encuentra en una fase de **estabilización y consolidación**, con la funcionalidad principal ya implementada y validada.

El foco actual no está en añadir nuevas features, sino en:

- mejorar claridad del código,
  - simplificar la UI,
  - reducir complejidad innecesaria,
  - y reforzar la documentación técnica para facilitar mantenimiento y evolución.
- 

## Estado funcional

- Gestión de **Eras, Periodos, Incursiones y Sesiones** completamente operativa.
  - Firestore es la **única fuente de verdad** para todos los datos.
  - Las sesiones:
    - se crean, pausan y reanudan correctamente,
    - pueden abarcar largos periodos de tiempo (días o meses),
    - la duración se calcula siempre a partir de timestamps persistidos.
  - Finalización de incursión funcional:
    - el formulario de resultado y puntuación se muestra al iniciar la acción de finalizar,
    - los datos se validan antes de persistir,
    - la puntuación se calcula según las reglas del README.
- 

## UI / UX

- UI funcional en **Android (objetivo final)** y **Windows (entorno de debug)**.
- Control de sesiones mediante:
  - FAB en Android,
  - botón fijo en barra inferior en Windows.

- La UI está en fase de **refinamiento visual y estructural**:
    - priorizando alineación, agrupación y uso correcto de componentes Flet,
    - evitando sobreestilado manual (colores, estilos duplicados),
    - buscando una base clara y mantenible antes de reutilización avanzada.
- 

## Arquitectura y código

- Arquitectura basada en separación clara de responsabilidades:
    - pantallas (UI),
    - handlers y lógica de interacción,
    - servicios (Firestore, puntuación),
    - utilidades compartidas.
  - Código mayoritariamente funcional/procedimental, de forma consciente:
    - sin sobreingeniería orientada a objetos,
    - priorizando legibilidad y trazabilidad.
  - Se está revisando la organización interna de pantallas para:
    - reducir funciones demasiado largas,
    - mejorar comprensión de la UI (especialmente Flet),
    - facilitar correcciones manuales sin depender excesivamente de Codex.
- 

## Documentación

- README.md es la **fuente canónica** de reglas, modelo y alcance.
  - DOCUMENTATION.md ha sido ampliado y validado:
    - describe estructura de carpetas,
    - flujo general de la app,
    - responsabilidades de cada módulo y función,
    - principios de diseño aplicados.
  - Documentación orientada a **uso práctico y comprensión**, no a marketing ni academicismo.
- 

## Pendiente / fuera de alcance actual

- Exportación a TSV (descrita en README):
  - **pendiente**, no prioritaria en esta fase.
- Revisión global de estilo visual compartido:

- posible mejora futura,
  - no bloqueante para funcionalidad ni arquitectura.
- 

## Próximo foco previsto

- Refinar pantallas complejas (especialmente detalle de Incursión).
- Reducir fricción al editar o corregir UI directamente.
- Consolidar patrones internos antes de añadir nuevas funcionalidades.