

Faeterna



Elaboración de plan de pruebas.



MIT License

Copyright (c) 2026 Faeterna

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Índice

Índice.....	2
1. Introducción.....	3
2. Alcance.....	3
Funcionalidades a probar:.....	3
Funcionalidades a NO probar:.....	3
3. Estrategia.....	3
Tipos de pruebas:.....	3
Entorno:.....	4
4. Entorno.....	4
4. Criterios de entrada y salida.....	4
5. Recursos.....	4
6. Cronograma.....	5
7. Entregables (Informes de resultados).....	5

1. Introducción

Definir el enfoque, los recursos y el cronograma para asegurar que el juego sea divertido, estable y libre de errores críticos que rompan la progresión (game-breaking bugs). Se busca validar tanto la fluidez del movimiento (feel) como la lógica de la gestión de recursos.

2. Alcance

Funcionalidades a probar:

- **Mecánicas de Movimiento:** Salto, dash, doble salto, colisiones con paredes y techos.
- **Combate:** Hitboxes de ataque, frames de invulnerabilidad, comportamiento de enemigos (IA).
- **Gestión de Recursos:** Recolección de materiales, menús de fabricación (crafting), consumo de energía/munición.
- **Progresión (Metroidvania):** Desbloqueo de habilidades que permiten acceso a nuevas áreas (*Ability Gateways*).
- **Sistema de Guardado:** Persistencia de inventario y posición en el mapa.

Funcionalidades a NO probar:

- Multijugador (si es solo single-player).
- Rendimiento en consolas (si el alcance actual es solo PC/Windows).

3. Estrategia

Tipos de pruebas:

1. **Pruebas Funcionales:** ¿Funciona el botón de inventario? ¿El personaje muere al tocar pinchos?
2. **Pruebas de Balanceo:** ¿Es el jefe demasiado difícil? ¿Los recursos son muy escasos o abundantes?
3. **Pruebas de Nivel (Level Design):** Asegurar que el jugador no se quede atrapado (*soft-lock*) en un área sin la habilidad necesaria.
4. **Pruebas de Usabilidad (UI/UX):** Claridad de los menús de gestión y legibilidad del pixel art.

Entorno:

- **Hardware:** Pruebas en PCs de gama baja, media y alta para asegurar estabilidad de FPS (crucial en pixel art para evitar el *screen tearing*).
- **Software:** Windows 10/11, drivers de video actualizados.
- **Herramientas:** Uso de **icaria TDM** (como mencionas) para gestionar sets de datos de prueba, como perfiles de jugador con diferentes niveles de recursos o habilidades desbloqueadas sin tener que jugar todo el juego desde el inicio.

4. Entorno

Planificación de pruebas de software se define qué características deberán presentar los entornos de hardware y software para ejecutar las pruebas.

Por ejemplo, la plataforma de Test Data Management que se empleará durante el proceso de testeo. Se trata de un paso crucial, pues la elección de herramientas adecuadas como icaria TDM permite la mejora en la calidad del software gracias a una mejor cobertura de pruebas que, además, anticipa la detección de fallos.

4. Criterios de entrada y salida

- **Criterios de Entrada:** * Build del juego estable (Alpha/Beta).
 - Documento de diseño de juego (GDD) actualizado para comparar el comportamiento esperado.
 - Herramientas de reporte de bugs listas (ej. Jira, Trello o Excel).
- **Criterios de Salida:** * 0 bugs críticos (bloqueos de juego).
 - 100% de las salas del mapa recorridas y validadas.
 - Pruebas de "ruta crítica" completadas con éxito.

5. Recursos

- **Equipo Humano:**
 - **QA Lead:** Coordina y diseña los casos de prueba.
 - **Testers de Funcionalidad:** Buscan errores técnicos y colisiones.
 - **Playtesters:** Jugadores externos para evaluar la dificultad y el flujo de la gestión de recursos.

6. Cronograma

- **Semana 1:** Pruebas de mecánicas básicas y colisiones (Movimiento).
- **Semana 2:** Pruebas de sistemas de gestión y menús.
- **Semana 3:** Pruebas de integración (Mundo abierto y progresión).
- **Semana 4:** Corrección de errores y test de regresión (verificar que lo arreglado no rompió otra cosa).

7. Entregables (Informes de resultados)

Informe de Errores (Bug Report): Lista detallada con capturas o videos de los fallos.

Matriz de Trazabilidad: Documento que vincula cada mecánica con su prueba superada.

Resumen Ejecutivo de Calidad: Valoración final sobre si el juego está listo para ser publicado o necesita más pulido.

info de como hacer un plan de prueba:

<https://www.gamakers.com/p/ejemplo-de-plan-de-pruebas.html>