Zapis gry

Zapis gry

Strona referencyjna. Pliki: `core/save/save_manager.gd` (autoload `SaveManager`), `core/ui/save_slots_screen.tscn`. Opowieść i klocki: odcinek „Ognisko, które leczy, uczy i pamięta”.

Gdzie i co

- 3 sloty: `user://saves/slot_1.json` … `slot_3.json` (`SaveManager.absolute_dir()` podaje ścieżkę w systemie; na Windows zwykle `%APPDATA%\Godot\app_userdata\Horrorek\saves`).
- Plik (JSON):

Pole	Zawartość
`format`	Wersja koperty zapisu (`FORMAT_VERSION = 1`).
`meta`	Czas zapisu, pokój, poziom, godzina i dzień gry, rzeczywistość (do listy slotów).
`state`	`GameState.serialize()` (własna wersja `SAVE_VERSION`, obecnie 8: z pozostałością po śmierci).
`player`	Pokój, pozycja (x, y, z) i obrót gracza.

Po wczytaniu gracz stoi tam, gdzie stał w chwili zapisu: `LocationManager` ładuje pokój i ustawia gracza przez `place_player()`.

API

Metoda	Działanie
`save_slot(slot)`	Zapis (1..3).
`load_slot(slot)`	Odczyt i zastosowanie (stan, pokój, pozycja).
`read_slot(slot)`	Odczyt i sprawdzenie bez zmiany stanu gry.
`slot_info(slot)`, `list_slots()`	Metadane do listy w menu (`exists`, `ok`, `error`, `meta`).

Metoda	Działanie
`save_dir`	Katalog zapisów (testy podmieniają go na katalog tymczasowy).
Sygnały	`saved(slot)`, `loaded(slot)`, `failed(slot, error)`.

Każda metoda zwraca `{ ok, error, detail }`: `error` to klucz tłumaczenia, a `detail` opis techniczny (trafia też do logu jako ostrzeżenie).

Bezpieczeństwo zapisu

- **Zapis atomowy**: najpierw `slot_N.json.tmp`, potem zmiana nazwy na `slot_N.json`. Przerwany zapis zostawia najwyżej plik `.tmp`; poprzedni zapis jest cały, a następny zapis sprząta.
- **Odczyt bez ryzyka**: cały plik jest sprawdzany (poprawny JSON, `format`, sekcja `state`, jej wersja) **przed** zmianą stanu gry. Błędy:

Klucz	Kiedy
`SAVE_ERROR_CORRUPT`	Plik nie jest poprawnym JSON albo brakuje wymaganych pól.
`SAVE_ERROR_VERSION`	Plik z nowszej wersji gry (`format` albo `state.version` większe niż obsługiwane).
`SAVE_ERROR_MISSING`	Pusty slot.
`SAVE_ERROR_INVALID_SLOT`	Numer slotu spoza 1..3.
`SAVE_ERROR_WRITE`	Nie udało się zapisać pliku.

- Starsze wersje stanu są wczytywane: brakujące pola dostają wartości domyślne (np. zapis bez statystyk startuje z pełnym zdrowiem).

Interfejs

- **Zapisz**: pozycja w menu medytacji, tylko przy ognisku (także we śnie). Ekran pokazuje 3 sloty z metadanymi („Pokój A, poziom 1, dzień 3, 20:00 (data zapisu)”), „pusty” albo czytelny błąd.
- **Wczytaj**: pozycja w menu pauzy, ta sama lista. Pusty albo uszkodzony slot jest wyszarzony. Po wczytaniu menu pauzy się zamyka i gra rusza.
- Teksty przez klucze tłumaczeń (`SAVE_*`, `LOAD_*`).

Menu deweloperskie

Sekcja „Zapis”: „Zapisz slot 1–3”, „Wczytaj slot 1–3”, ścieżka katalogu zapisów i wynik ostatniej operacji.

Testy

`tests/test_save.gd`, `tests/test_dev_menu.gd`.

Opublikowano z repozytorium Horrorek (commit 87717ad) skryptem publikuj-bookstack.ps1.

Revision #1
Created 2026-10-28 22:54:02 UTC by Tomasz Konieczny
Updated 2026-10-28 22:54:02 UTC by Tomasz Konieczny