Menu deweloperskie

Menu deweloperskie

Strona referencyjna. Pliki: `core/dev/dev_menu.tscn`, `core/dev/dev_menu.gd`, `core/dev/dev_actions.gd`, węzeł `DevMenu` w `content/main.tscn`. Opowieść i klocki: odcinek „Menu, które widzi tylko twórca”.

Do czego służy

Do oglądania i testowania stanów gry bez grania: wymuszenie snu albo koszmaru, zmiana aktu i zakresu questu, przejście do dowolnego pokoju, ustawienie czasu i flag, nowy stan gry. Menu **nie trafia do wydania**.

Otwieranie i zamykanie

Element	Opis
Akcja `dev_menu`	F9. Otwiera i zamyka menu.
Zamykanie	F9, Esc (`ui_cancel`) albo akcja `pause_menu`. Menu przechwytuje je, zanim zobaczy je menu pauzy.
Otwarte menu	Wstrzymuje grę i pokazuje mysz (`MouseMode`); zamknięcie przywraca poprzedni tryb myszy.
Wykluczanie	Nie otwiera się, gdy gra jest wstrzymana (np. pauza) albo otwarte jest menu medytacji. Menu pauzy nie otwiera się przy otwartym menu deweloperskim (grupa `dev_menu`). Ogniska nie da się użyć, bo gracz stoi.
Wydanie (release)	W `_ready` węzeł usuwa się sam, gdy `DevMenu.debug_build_check` (domyślnie `OS.is_debug_build`) zwraca `false`. `force_enabled = true` włącza menu mimo to (tylko testy).

Pola (Inspektor, scena główna)

Pole	Opis
`game_root`	Scena główna (`GameRoot`), potrzebna do „Nowy stan gry”.
`known_flags`	Flagi do przełączania, np. `door_cellar_open`. To treść, więc ustawia się ją w `content/main.tscn`, nie w kodzie.
`give_items`	Przedmioty do dania jednym kliknięciem, np. `item_key_cellar.tres`.
`enemy_scenes`	Przeciwnicy do przywołania w sekcji „Walka”, np. `content/combat/enemy_test.tscn`.
`force_enabled`	Tylko testy: menu działa także w wydaniu.

Sekcje

Sekcja (`id`)	Zawartość
Rzeczywistość (`realm`)	Jawa (`return_to_waking`), Sen, Koszmar. Z jawy: `enter_dream` w bieżącym pokoju, bez losowania i warunków. Sen ↔ koszmar: `switch_dream_kind` bez budzenia. Zawsze z zaciemnieniem; menu zamyka się przed przejściem.
Koszmar (`nightmare`)	„Następne zaśnięcie = koszmar” (`NightmareDirector.force_next`, jednorazowo, bez zużycia limitu) i „Resetuj zużycie koszmaru” (`GameState.clear_nightmare_used()`).
Akt i zakres questu (`story`)	Akt 1–3, pole tekstowe `quest_scope` (Enter albo „Ustaw zakres”).
Pokój (`room`)	Przyciski dla wszystkich pokoi z rejestru (`LocationManager.get_room_ids()`), przejście `go_to` na pierwszy spawn.
Stan (`state`)	Godzina i dzień (`GameState.set_time`), przełączniki `known_flags`, „Daj: …” dla `give_items`, „Nowy stan gry” (drugie kliknięcie potwierdza; `GameRoot.start_new_game()`).
Sen proceduralny (`procedural_dream`)	Bieżące ziarno snu, „Nowe ziarno (przeładuj sen)”, „Ustaw ziarno” (pole tekstowe), liczniki wygenerowanych elementów według typu, „Pokaż strefy slotów” (zob. strona „Sen proceduralny”).
Postać (`character`)	Wartości statystyk (pole i „Ustaw”), „+100 XP”, „+1000 XP”, „Ustaw poziom” (bez premii z awansu). Zob. „Statystyki, leczenie i awans”.

Sekcja (`id`)	Zawartość
Walka (`combat`)	„Przywołaj: …” (przeciwnik 3 m przed graczem, w bieżącym pokoju), „Zabij wszystkich”, „Zabij gracza” (śmierć i powrót do ogniska), licznik żywych przeciwników, „Nieśmiertelność gracza”, „Podgląd kolizji walki” (hitboxy i hurtboxy), „Napisy nad celem” i „Liczby obrażeń nad celem” (`CombatText`). Zob. „Walka”.
Zapis (`save`)	„Zapisz slot 1–3”, „Wczytaj slot 1–3”, ścieżka katalogu zapisów, wynik ostatniej operacji. Zob. „Zapis gry”.

Na górze menu jest pasek stanu: rzeczywistość, pokój, akt, zakres questu, godzina i dzień.

Nowa sekcja

```
dev_menu.add_section(&"my_tool", "Moje narzędzie", func(box: VBoxContainer) -> void:
	var button: Button = Button.new()
	button.text = "Zrób coś"
	button.pressed.connect(func() -> void: print("moje narzędzie"))
	box.add_child(button))
```

- Kolejność wyznacza `SECTION_ORDER`; sekcje spoza listy trafiają na koniec.
- Builder dostaje pusty `VBoxContainer` sekcji i jest wołany przy każdym otwarciu menu, więc pokazuje aktualny stan.
- Teksty menu są po polsku, bez kluczy tłumaczeń (menu nie trafia do wydania), ale w jednym słowniku `TEXTS` na górze `dev_menu.gd`.

DevActions

Logika przycisków jest w `core/dev/dev_actions.gd` (funkcje statyczne): `set_realm(kind)`, `force_next_nightmare()`, `reset_nightmare_usage()`, `set_act(act)`, `set_quest_scope(scope)`, `go_to_room(room_id)`, `set_time(hour, day)`, `set_flag(flag, value)`, `give_item(item)`, `new_game(game_root)`, a dla walki `spawn_enemy(tree, scene)`, `kill_all_enemies(tree)`, `enemy_count(tree)`, `kill_player()`, `set_immortal(value)`, `set_combat_debug(tree, value)`, `set_combat_labels(tree, value)`, `set_damage_numbers(tree, value)`. Można je wołać także z własnych narzędzi.

Testy

`tests/test_dev_menu.gd`, `tests/test_pause_menu.gd` (grupa `dev_menu`).

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

Revision #2
Created 2026-10-28 00:48:15 UTC by Tomasz Konieczny
Updated 2026-10-28 22:53:59 UTC by Tomasz Konieczny