Kamera: priorytety, strefy i sterowanie

Kamera: priorytety, strefy i sterowanie

Strona referencyjna. Pliki: `core/camera/player_camera.tscn`, `core/camera/camera_zone.gd`, `core/camera/camera_control.gd`. Wtyczka: Phantom Camera 0.11.0.3 (`addons/phantom_camera`).

Elementy w scenie

Węzeł	Typ	Rola
`MainCamera3D`	`Camera3D`	Jedyne „oko”, czyli kamera, która faktycznie renderuje obraz.
`MainCamera3D/PhantomCameraHost`	`PhantomCameraHost`	Wybiera kamerę wirtualną z najwyższym priorytetem i przenosi do niej oko.
`PlayerCamera`	`PhantomCamera3D`, tryb Third Person	Kamera za plecami gracza, priorytet 10, `follow_target` = gracz, `follow_offset` na wysokość głowy.
`CornerCamera`	`PhantomCamera3D`	Kamera stałego ujęcia strefy, priorytet 0 (poza strefą).
`CornerZone`	`Area3D` + `camera_zone.gd`	Strefa, która podnosi priorytet swojej kamery, gdy wejdzie w nią gracz.
`CameraControl`	`Node` + `camera_control.gd`	Obrót kamery za plecami myszą i prawą gałką.

Priorytety

- Host zawsze pokazuje obraz z kamery wirtualnej o najwyższym `priority`.
- Kamera za plecami ma 10. Strefa ustawia swojej kamerze `active_priority` (domyślnie 20) przy wejściu gracza i `inactive_priority` (domyślnie 0) przy wyjściu.
- Gracza rozpoznajemy po grupie `player` (`CameraZone.player_group`).
- Metody `CameraZone.activate()` i `deactivate()` są publiczne, więc da się je wołać ze skryptu lub testu.

Sterowanie (CameraControl)

Pole	Domyślnie	Znaczenie
`player_camera`	(brak)	Kamera za plecami (PhantomCamera3D, Third Person).
`host`	(brak)	PhantomCameraHost; puste = wejście zawsze aktywne.
`mouse_sensitivity`	0.12	Stopnie obrotu na piksel ruchu myszy.
`stick_sensitivity`	140	Stopnie na sekundę przy pełnym wychyleniu gałki.
`invert_y`	false	Odwrócona oś pionowa.
`pitch_min` / `pitch_max`	-60 / 70	Granice pochylenia ramienia kamery (stopnie).
`stick_deadzone`	0.15	Martwa strefa prawej gałki.
`capture_mouse_on_start`	true	Przechwycenie myszy przy starcie sceny.

- Akcje pada: `look_left`, `look_right` (oś 2), `look_up`, `look_down` (oś 3).
- Mysz: na starcie tryb przechwycony. Akcja `ui_cancel` (Esc) zwalnia mysz, a lewy przycisk w oknie przechwytuje ją ponownie.
- `apply_look(delta_degrees: Vector2)`: `x` to zmiana obrotu w poziomie (yaw, bez limitu), `y` to zmiana pochylenia (pitch, z limitami). Woła ją obsługa myszy i gałki, a także testy.
- Gdy aktywna jest inna kamera niż `player_camera` (np. ujęcie strefy), wejście jest ignorowane, a `apply_look` zwraca `false`.
- Ruch gracza liczy się względem aktywnej `Camera3D`, więc obrót kamery od razu zmienia kierunek „przodu”.

Pułapki

- W trybie Third Person Phantom Camera po starcie przenosi kamerę wirtualną pod nowy węzeł `SpringArm3D`. Ścieżka `PlayerCamera` przestaje wtedy działać, a obrót ustawia się

metodami `set_third_person_rotation_degrees()` itp. dopiero, gdy ramię istnieje.
- Ramię kamery omija kolizje tylko z obiektem w `follow_target`, więc kamera za plecami śledzi samo ciało gracza, a nie marker.
- Host wybiera aktywną kamerę dopiero po pierwszej klatce procesu. Testy muszą odczekać klatkę procesu, nie tylko klatkę fizyki.
- W trybie bezgłowym (testy) Godot nie przechwytuje myszy, więc przełączanie trybu myszy trzeba sprawdzić ręcznie.

Testy

`tests/test_room.gd` (priorytety i strefa), `tests/test_camera_control.gd` (obrót, limity, `invert_y`, martwa strefa, ignorowanie wejścia przy kamerze strefy).

Opublikowano z repozytorium Horrorek (commit 3ec71cf) skryptem publikuj-bookstack.ps1.

Revision #1
Created 2026-10-27 21:45:47 UTC by Tomasz Konieczny
Updated 2026-10-27 21:45:47 UTC by Tomasz Konieczny