

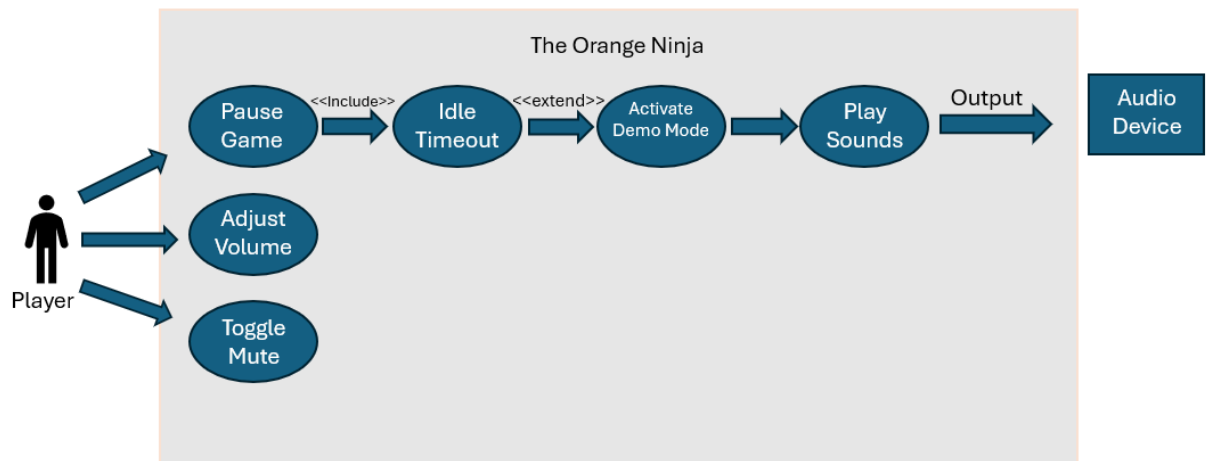
[Instructions: Remove everything that is not a heading below and fill in with your own diagrams, etc.]

1. Brief introduction __/3

I will implement two features for our team's game: (1) Demo Mode that auto-plays a curated gameplay sequence after the game has been idle for a configurable period; and (2) an adaptive Sound System that manages background music, sound effects, and user controls (volume, mute). Demo Mode showcases core mechanics, increases player engagement at kiosks/menus, and serves as an attractor loop. The Sound System reacts to in-game events and demo playback, ensuring timely audio feedback and consistent loudness.

2. Use case diagram with scenario __14

Use Case Diagrams



Scenarios

Name: Activate Demo Mode

Summary: System starts demo after inactivity.

Actors: Player, Game System.

Preconditions: Game running on Menu/Pause screen; assets preloaded.

Basic sequence:

- 1) The player pauses or idles on the menu.
- 2) The system starts an idle timer.

3) When idle \geq threshold, the system launches a demo sequence.

Exceptions (extend from Idle Timeout):

E1) Player input before threshold \Rightarrow cancel demo launch and reset timer.

Postconditions: Demo plays until interrupted; timer suspended.

Priority: 1 (Must have) ID: UC01

Name: Play Sounds

Summary: Engine plays context-appropriate SFX/BGM.

Actors: Game System, Audio Device.

Preconditions: Audio device available; volume set > 0 unless muted.

Basic sequence:

- 1) Game events occur (e.g., jump, hit, win).
- 2) Sound router selects clip and submits to mixer.
- 3) Mixer applies volume/mute and plays.

Exceptions:

E1) Audio device missing \Rightarrow queue log, show non-blocking warning, continue without sound.

Postconditions: Audible feedback consistent with the event.

Priority: 1 ID: UC02

Name: Adjust Volume

Summary: Player changes master volume (0–100%).

Actors: Players.

Preconditions: Sound system initialized.

Basic sequence:

- 1) Player moves slider; 2) Mixer updates gain; 3) Subsequent clips reflect change.

Exceptions: None.

Postconditions: Volume persists to settings.

Priority: 2 ID: UC03

Name: Toggle Mute

Summary: Player toggles global mute on/off.

Actors: Player.

Preconditions: Sound system initialized.

Basic sequence: 1) Player presses mute; 2) Mixer toggles gate; 3) UI icon updates.

Exceptions: None.

Postconditions: All audio suppressed or restored.

Priority: 2 ID: UC04

Name: Idle Timeout

Summary: Track inactivity and signal demo start when threshold is reached.

Actors: Game System.

Preconditions: Idle timer running on pause/menu.

Basic sequence: 1) Start timer; 2) Accumulate idle time; 3) When \geq threshold, raise DemoStart event.

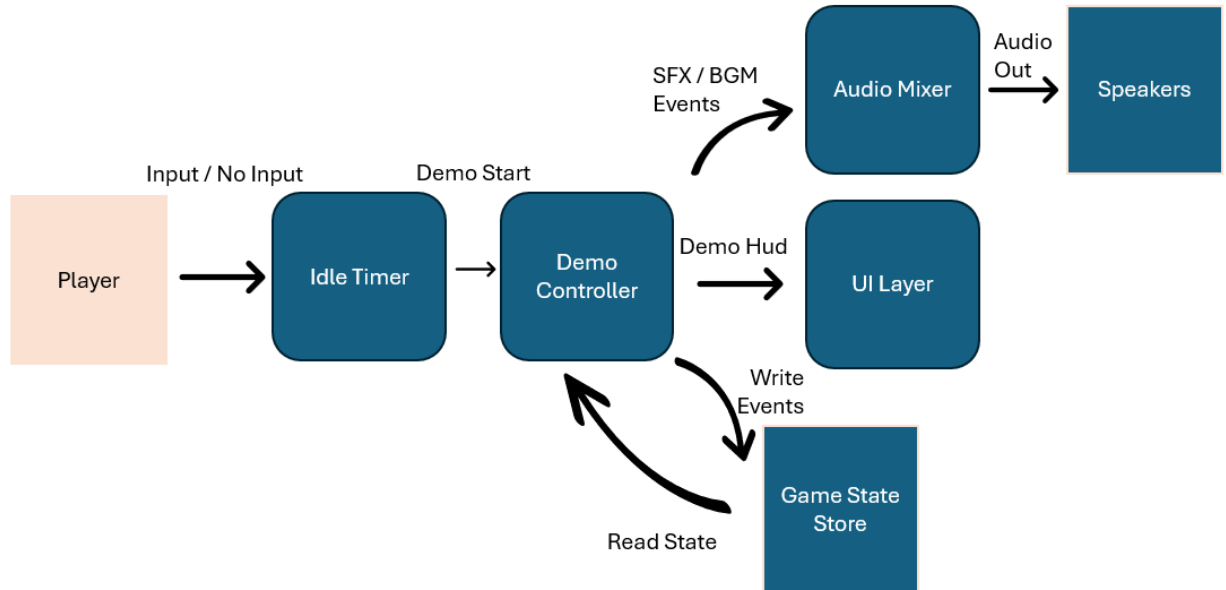
Exceptions: E1) Any input before threshold \Rightarrow reset timer (matches exception in UC01).

Postconditions: Either timer reset or demo launched.

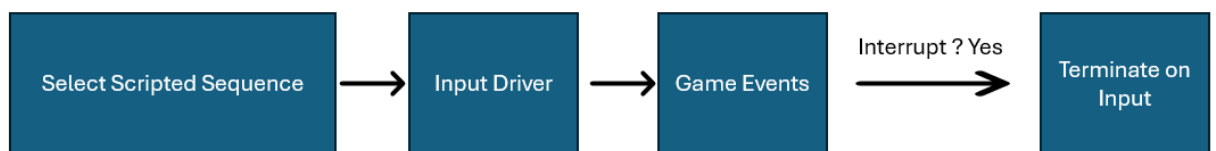
Priority: 1 ID: UC05

3. Data Flow diagram(s) from Level 0 to process description for your feature ____14

Data Flow Diagrams



Process: Demo Controller (Level 1)



Process Descriptions (Demo Controller)

```
WHILE game_state ∈ {Menu, Paused} DO
  IF idle_time ≥ threshold THEN
    Load pre-scripted sequence (seeded, deterministic).
    Initialize AI Input Driver with actions (move, jump, attack) and delays.
    FOR each action in sequence DO
      Emit input event to game loop.
      Trigger sound events via Audio Router.
    END FOR
    IF any player input THEN terminate demo and resume UI.
    ELSE continue accumulating idle_time.
  END IF
END WHILE
```

4. Acceptance Tests _____9

AT-01: Demo starts after exact threshold

Setup: Idle threshold = 60 seconds. Start from the Pause/Menu screen with no input devices generating events.

Steps: (1) Do not touch controls; (2) Observe system timer; (3) Measure time of DemoStart signal.

Expected: Demo Mode activates at $t = 60s \pm 0.5s$; demo HUD becomes visible; input driver begins scripted actions.

Pass/Fail: Pass if activation occurs within tolerance and demo renders first action; otherwise fail.

AT-02: Reset on pre-threshold input (exception path)

Setup: Idle threshold = 60s; begin idling from $t = 0s$.

Steps: (1) At $t \approx 40s$ press any input; (2) Continue idling again; (3) Observe timer reset.

Expected: No demo launch after first 40s; a full new 60s of inactivity is required before DemoStart can trigger.

Pass/Fail: Pass if DemoStart does not fire before the new 60s window; otherwise fail.

AT-03: Immediate termination on user input during demo

Setup: Allow demo to start per AT-01.

Steps: (1) As soon as demo begins, press any control; (2) Measure termination latency from input to DemoStop.

Expected: Demo stops within ≤ 100 ms; UI returns to previous state; gameplay is responsive to the new

input.

Pass/Fail: Pass if latency budget met and state restored; otherwise fail.

AT-04: Volume control at extremes

Setup: Ensure audio devices are present. Master volume initially 50%.

Steps: (1) Set volume to 0%; trigger SFX; (2) Set volume to 100%; trigger same SFX; (3) Verify perceived loudness difference via meter.

Expected: At 0%: no audible output, meters still show signal; at 100%: audible and louder vs 50%.

Pass/Fail: Pass if mixer gain mapping is monotonic and endpoints behave as specified.

AT-05: Mute gating

Setup: Volume 50%.

Steps: (1) Toggle mute ON; trigger SFX; (2) Toggle mute OFF; trigger SFX.

Expected: With mute ON, no audible output; with mute OFF, output resumes at prior volume setting; UI icon reflects state.

Pass/Fail: Pass if gate fully suppresses audio and restores correctly.

AT-06: Audio device missing

Setup: Unplug/disable audio device or simulate missing backend.

Steps: Trigger SFX and BGM events.

Expected: Non-blocking warning logged or displayed; game remains playable; no crash; demo and gameplay continue.

Pass/Fail: Pass if system degrades gracefully and preserves UX responsiveness.

AT-07: Stability under rapid pause/resume churn

Setup: Threshold 60s; automation available.

Steps: Programmatically perform pause/resume cycles 20 times within 2 minutes; monitor memory and timer state.

Expected: No duplicate timers, no leaked demo processes, stable memory footprint ($\Delta < 5\%$).

Pass/Fail: Pass if no leaks/races and all states consistent.

AT-08: Long-run demo loop integrity

Setup: Start demo and let it run for 5 minutes.

Steps: Observe camera paths, scripted actions, and audio loop points.

Expected: No visual drift or desync; audio loops at boundaries cleanly; sequence repeats predictably.

Pass/Fail: Pass if no cumulative timing errors or stutters are observed.

AT-09: Boundary thresholds

Setup: Test thresholds at 5s and 300s.

Steps: For each value, idle and measure activation time; repeat 3 trials each.

Expected: Activation at target within $\pm 0.5s$ at 5s and $\pm 1.0s$ at 300s; jitter within tolerance.

Pass/Fail: Pass if all trials meet tolerance windows.

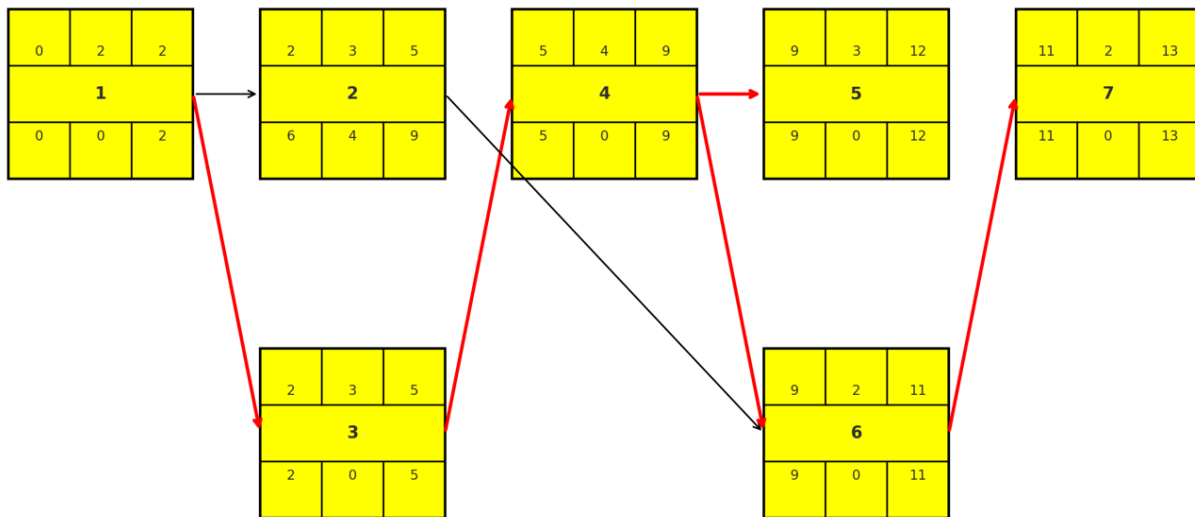
Tests	Input / Setup	Expected Output/Behavior	Notes
AT-1	Idle threshold = 60s; no input for 60s	Demo Mode starts at $t \approx 60s$	Tolerance $\pm 0.5s$
AT-2	Any player input at $t < 60s$	Timer resets; no demo	Covers exception path
AT-3	Input during demo	Demo terminates $\leq 100ms$; UI resumes	Latency budget
AT-4	Volume=0%	No audible output; meter shows activity	Mute vs volume edge case
AT-5	Mute=ON; SFX event	No output; unmute restores	State persists
AT-6	Device missing	Non-blocking warning; gameplay continues	Play Sounds exception
AT-7	Rapid pause/resume x20	No double-starts, no leaks	Stability under churn
AT-8	Demo for 5 min	No drift; sequence loops cleanly	Long-run behavior
AT-9	Idle threshold set to 5s & 300s	No drift; sequence loops cleanly	Boundary values

5. Timeline ____/10

Work items

Task ID	Task	Duration (PWks)	Predecessor Task(s)
1	Requirements Collection	2	-
2	Sound System Design	3	1
3	Demo Logic Design	3	1
4	Implementation	4	2, 3
5	Testing	3	4
6	Integration	2	4, 5
7	Documentation	2	6

Pert diagram



Gantt timeline

