

PROFILE

LINKEDIN

AGENTIC AI SYSTEMS

EDYBOY BADBOY

Retro Arcade Beat-'Em-Up — Local AI-Assisted Game Engineering on STRIX HALO

AGENTIC AI SYSTEMS — TECHNICAL RESEARCH PUBLICATION

Document ID: 202611-AASPUB-0011 | Publication date: 13 November 2026 | Version: 1.0

Research platform: STRIX HALO — AMD Ryzen AI Max+ 395 / Radeon 8060S | Evidence: Project handover dated 3 November 2026

1. Executive Summary

Agentic AI Systems developed EDYBOY BADBOY, a retro pixel-art side-scrolling beat-'em-up game, using Phaser 3 and Vite on the STRIX HALO Linux workstation. The project combines browser-based gameplay, a dockside stage, keyboard combat, locally prepared sprite artwork and an iterative AI-assisted engineering workflow involving Qwen 3.8 Flash Next.

The historical engineering record confirms a successful JavaScript build, integration of 26 protagonist animation frames and HTTP 200 responses from the development server and a sprite asset. These results establish a working development baseline, not complete end-to-end browser gameplay validation.

2. Project Overview and Creative Direction

Field	Project specification
Game title	EDYBOY BADBOY (formerly EDD — THE DOCKS)
Genre	Retro 8-bit-style side-scrolling arcade beat-'em-up
World	Moonlit industrial docks and harbour, cranes and crates
Protagonist	EDD / EDYBOY: red hoodie, beard, glasses, black cargo shorts, red/white sneakers
Enemy concepts	Dock Thug, Dock Bruiser and Ninja
Controls	WASD / arrows: move; J: punch; K: kick; R: restart; M: mute

3. Research and Engineering Objectives

Develop a playable browser game using locally hosted tools; replace placeholder protagonist graphics with sprite animations; preserve original artwork during extraction; confirm JavaScript build and static asset serving; and investigate title-screen, music and menu audio integration. The handover distinguishes completed work from unverified or proposed features.



PROFILE

LINKEDIN

AGENTIC AI SYSTEMS

4. Development Platform and Toolchain

Component	Recorded configuration
Development host	strix.home.arpa / STRIX HALO
Operating system	Ubuntu 26.04.1 LTS
AI workstation	AMD Ryzen AI Max+ 395; Radeon 8060S; 128 GB unified memory
Local AI assistant	Qwen 3.8 Flash Next (development context; direct contribution metrics not recorded)
Game framework	Phaser 3.90 (historical specification)
Build and dev tools	JavaScript, Node.js, npm, Vite 7.3.7 (verified build log)
Game root	/home/edyboy/AAS-GAME-STUDIO/EDD-THE-DOCKS
Primary source	src/main.js
Vite development listener	0.0.0.0:5173; HTTP 200 recorded
Sprite Studio preview	0.0.0.0:5174; HTTP 200 recorded

The Qwen 3.8 Flash Next model is identified as part of the local AI-assisted development setup. The supplied game handover does not contain a reproducible prompt-to-code attribution log or quantified model productivity benchmark; these should not be presented as measured results.

5. Game Architecture

Browser gameplay: Phaser handles the game scene, player movement, attacks, enemy behaviour, health and audio events. Vite serves source and assets during development and produces a bundled production build.

Sprite pipeline: original sprite sheets → layout-based crops → conservative background cleanup → animation PNG assets → Phaser preload and animation registration → player visual overlay.

Development validation: source checks → npm build → local HTTP checks → browser runtime inspection (remaining requirement).

6. Source Assets and Animation Inventory

Animation	EDD frames	Recorded behaviour
Idle	4	Looping
Walk	6	Looping; 10 frames/second
Punch	4	One-shot attack
Kick	5	One-shot attack
Hit	3	One-shot damage response
Knockdown	4	One-shot collapse
TOTAL	26	Integrated as EDD V4.2 sprite set



PROFILE

LINKEDIN

AGENTIC AI SYSTEMS

7. Sprite Engineering Methodology

The original 1536 × 1024 EDD sprite sheet contained 26 poses, alongside complex dark backgrounds and tightly packed neighbouring artwork. The enemy sheet also measured 1536 × 1024 and contained three enemy archetypes with 26 candidate poses each.

Iteration	Technique	Recorded outcome
V3.2	Layout crop extraction	104 candidates: 26 EDD and 78 enemy frames; intact source pixels with fragments
V3.5–V3.8	Segmentation / GrabCut trials	Rejected for damaging dark clothing, limbs, fists and shoes
V3.9	Boundary inspection	26 EDD poses inspected; identified contamination
V4.0	Conservative flood-fill	26 EDD candidates generated; visual comparison approved as 'looking good'
V4.2	Phaser integration	26 EDD PNGs copied, source patched, syntax and build checks passed

V4.0 preserved more of the original figure than aggressive masking, but stray neighbouring fragments remained in some frames. The 78 enemy candidates were not confirmed integrated into Phaser at the time of the handover.

8. Phaser V4.2 Integration

The V4.2 update loaded 26 EDD animation frames into public/sprites/edd/, registered idle, walk, punch, kick, hit and knockdown animations, and attached an eddVisual sprite to the existing physics/combat object. Recorded visual scale was 0.65, with origin (0.5, 0.85), position tracking, depth tracking and directional flipping.

The project retained its underlying collision and combat object while replacing its visual representation. Idle and walk animations repeat; attacks and damage animations play once. The patch passed node --check and npm run build with Vite 7.3.7; the bundle-size warning was nonfatal.

9. Build and HTTP Verification

Verification	Recorded result
EDD sprite frames copied	26 / 26
JavaScript syntax check	PASS
Vite production build	PASS; 8 modules transformed; ~2.85 seconds
Development homepage	HTTP 200 on 127.0.0.1:5173
EDD idle sprite asset	HTTP 200 on /sprites/edd/idle/idle-01.png
Browser animation and combat QA	Not independently confirmed



PROFILE

LINKEDIN

AGENTIC AI SYSTEMS

10. Title Screen and Audio Design

The proposed EDYBOY BADBOY title screen includes a large game logo, moonlit dockside scene, START and CREDITS options, www.agenticaisystem.com.au and a Game Studio Entertainment 2026 copyright footer. Title artwork concepts were produced; Phaser menu integration and interactive behaviour were not verified in the historical record.

Audio asset	Intended purpose	Status
EDYBOY_BADBOY_8BIT_ROCK_TITLE.wav	Original retro rock title music	Generated; deployment unverified
BLIP / BLEEP / BEEP	Menu navigation variants	Generated; selection and playback unverified
BABING / CHA-CHING / BOING	Credits/confirmation variants	Generated; playback unverified
LET'S GO BABY!	START voice cue	Generated first attempt; quality unverified
hip-hop-theme.wav, SMACK, CRACK, UH, ARGH	Existing gameplay audio	Previously reported in game assets

11. Engineering Considerations

Image extraction required careful preservation of dark apparel, skin, fists and footwear; automated segmentation could erase legitimate character pixels. A local Python script named inspect.py shadowed the standard library and caused import failures, later repaired by renaming it. A missing Sprite Studio route caused a 404 despite the preview image existing; the route was subsequently added and verified.

An earlier connection-refused error on port 5173 reflected an absent Vite listener rather than a confirmed gameplay fault. After launching the development server, both homepage and sprite URL returned HTTP 200. Server reachability and build success do not prove that collisions, attack timing, visual alignment or browser audio work correctly.

12. Limitations and Verification Boundaries

This publication is based on the 3 November 2026 master project handover and historical terminal outputs. It is not a current audit of STRIX HALO or the later website deployment. Browser-runtime sprite animations, enemy graphics integration, the final title rename, menu navigation, title music, audio playback and end-to-end gameplay were not all confirmed in the underlying evidence.

No controlled measurements of Qwen 3.8 Flash Next's contribution, coding accuracy, token usage or time savings were included in the supplied game handover.



PROFILE

LINKEDIN

AGENTIC AI SYSTEMS

13. Key Findings

Local development: Phaser/Vite browser-game development ran on STRIX HALO. Animation integration: 26 EDD sprite frames were incorporated into Phaser. Build validation: JavaScript checks and the Vite production build passed. HTTP validation: the development game and at least one sprite asset returned HTTP 200. Engineering trade-off: conservative image cleanup better preserved protagonist geometry than aggressive segmentation.

14. Future Development

Next-stage work includes live browser testing of movement, attacks, hit and knockdown animations; visual inspection of collision alignment; confirming the game title; integrating and testing the title screen and keyboard menu; auditioning and wiring audio; and integrating enemy sprite candidates incrementally. These are engineering recommendations, not completed milestones.

15. Conclusion

EDYBOY BADBOY demonstrates a practical local AI-assisted workflow for retro browser-game engineering. The recorded development process progressed from placeholder graphics through iterative sprite extraction to an integrated Phaser character animation set, a successful build and a reachable development server. Additional browser-level verification is required before characterising the complete game and proposed menu/audio systems as fully operational.

16. Sources, Credits and Disclosure

Primary evidence: EDYBOY_BADBOY_MASTER_PROJECT_HANDOVER_2026-11-03.md; recorded V3.2–V4.2.2 engineering checkpoints, terminal outputs, asset inventories and user review. Historical evidence only; current production state not independently inspected.

Development technologies: Phaser 3, Vite, Node.js, JavaScript, Ubuntu Linux, STRIX HALO and local Qwen 3.8 Flash Next inference. Product names remain trademarks of their respective owners. No affiliation or endorsement by the technology vendors is implied.

About Agentic AI Systems

Agentic AI Systems is an Australian technology initiative focused on artificial intelligence, intelligent automation, GPU-accelerated computing and practical AI integration. Intelligence. Automation. Advantage.

www.agenticaisystem.com.au

17. Public Game Deployment

EDYBOY BADBOY is publicly listed as the first game in Agentic AI Systems' DF Game Studio portal. The live listing describes a Phaser 3 arcade-style fighting game with pixel-art characters, combat animations, music and sound effects, and includes a Play Now link. This confirms publication of the game listing, separate from the historical STRIX development checks.

Public game portal:

https://www.agenticaisystem.com.au/games/

STRIX HALO — VERIFIED TECHNICAL AUDIT

Document ID: 202611-AASPUB-0011 | Audit supplement: 13 November 2026

This supplement updates the historical engineering status in Sections 4–17 using the STRIX-hosted read-only audit supplied by the project owner. Where the historical publication describes features as unimplemented, the findings below supersede those historical statements for source-code and asset presence; runtime testing remains separate.

18. Verified Hardware and Software

Category	Audit observation
Hardware	Bosgame BeyondMax Series; AMD Ryzen AI Max+ 395, 16 cores / 32 threads; AMD Radeon 8060S graphics
Memory	122 GiB usable; 31 GiB used and 91 GiB available at capture; 8 GiB swap
Operating system	Ubuntu 26.04.1 LTS; Linux kernel 7.0.0-43-generic; x86-64
Game framework	Phaser 3.90.0; Vite 7.3.7; Node.js 22.22.1; npm 9.2.0
Game source	/home/edyboy/AAS-GAME-STUDIO/EDD-THE-DOCKS/src/main.js
Assets	51 PNG and 14 WAV files in public/
Storage	Root volume 867 GB, 792 GB used, 39 GB available; 96% utilisation

19. Current Scene and Gameplay Implementation

The current src/main.js contains DocksScene, TitleScene and CreditsScene, with title/credits scene methods and audio-loading calls. The audit also shows 26 EDD frames across six animations and 25 Enemy 01 frames across six animations, including enemy idle, walk, punch, kick, hit and knockdown.

Enemy 01 is now represented in game source: its frames are preloaded, animations registered, and a sprite visual is attached to its existing physics object. Enemy health is initialised to 3. The player has 100 health. J triggers a range-65 punch for 1 damage and K triggers a range-90 kick for 2 damage, subject to a 350 ms attack cooldown. Enemy contact within a distance threshold of 54 can reduce player health by 5, with a 1300 ms enemy attack cooldown. These are source-code observations from the line-indexed audit, not a browser playtest.

20. Audio and Menu Assets

The audit confirms presence of title artwork, background art, credits artwork, an 8-bit rock title WAV, BEEP, BABING and START voice-cue WAV files under public/aas-v45/. Gameplay audio includes hip-hop-theme, SMACK, CRACK, UH and ARGH, plus five Enemy 01 WAV files (E01–E04 and DEAD). Audio preloading and enemy sound-play calls are present in source; audible playback has not been independently tested.

21. AI and Deployment Verification Boundary

Qwen 3.8 Flash Next is recorded in the historical development context. The current audit returned no matching Qwen/llama-server/vLLM process, so it does not verify a live inference endpoint, model weights or model configuration. It also returned no matching listeners on ports 5173, 5174, 8190 or 8188 at capture time. This does not establish whether the public game hosted separately is available. The project is listed at https://www.agenticaisystem.com.au/games/.

22. Evidence and Limitations

Evidence: EDYBOY_BADBOY_FULL_TECH_AUDIT.txt (315-line STRIX read-only audit), together with the 3 November master handover. The audit includes selected grep matches rather than complete source text; therefore exact menu transitions, audio-unlock handling, scene startup order, the number of enemies instantiated and production build integrity require deeper source inspection and browser verification. The audit did not execute a fresh build or live gameplay test.