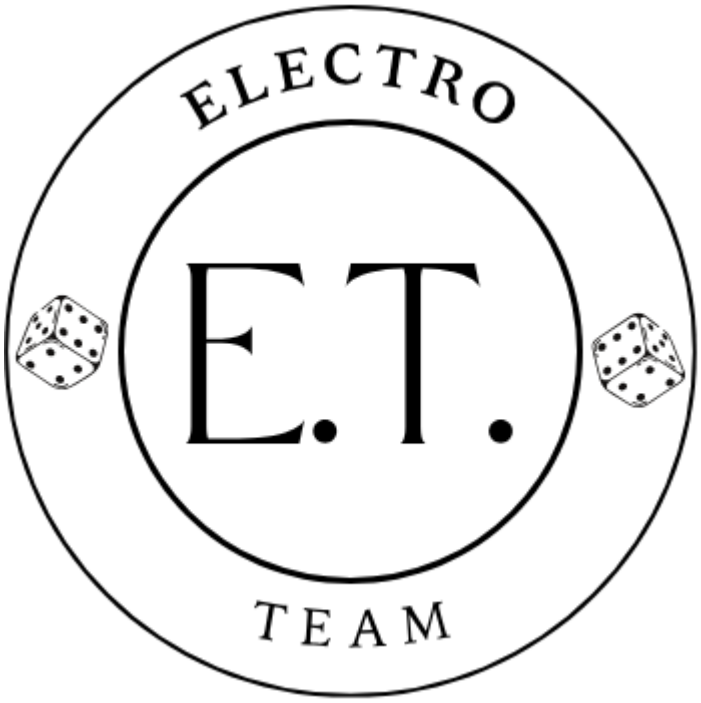
E-Team presents…

FATE/NONSTOP



Version: 0.0.1

A Game Development Project as Requirement for the course EMC131

Alfonso, Favio Maximo G.
Besana, A.L. Schatz A.
Faller, Amiel Ed Angelo A.

GAME DESIGN DOCUMENT

Game Name: Fate/Nonstop

Genre: 2D Side-Scrolling Arcady Platformer

Game Elements:

- Automatic running through levels filled with obstacles
- Being chased by a robot
- Avoiding hazards
- Collecting circles and hidden documents
- Platforming includes parkour elements
- Bar on screen that gives more points based on perfect execution
- Competing for best level time

Player:

- Single-player only

TECHNICAL SPECS

Technical Form:

- Flat 2D graphics

View:

- Side-scrolling 2D camera, following the player horizontally and vertically

Platform:

- PC

Language:

HTML and Javascript, using Phaser Engine

Device:

- PC

GAME PLAY

In *Fate/Nonstop*, formerly *Project Sprint,* players take on the role of a future resistance figure on the run through a modern urban cityscape, relentlessly pursued by a deadly robot assassin sent back in time. The action unfolds as a fast-paced, side-scrolling platformer where the player automatically sprints through dynamically designed levels filled with obstacles that demand split-second timing and perfect execution. Players must master parkour-inspired movements—vaulting over barriers, wall-jumping between narrow gaps, and sliding under hazards—to stay ahead of their pursuer. The chase is intense and constant, with a visible timer and a precision execution bar rewarding players with higher scores for flawless runs. Along the way, players collect glowing circles to boost their score and uncover hidden documents that reveal the larger story behind the AI war. Having three levels, each present a new segment of the city, from towering buildings to precarious rooftops to a military base and ending on a weapons testing laboratory, each offering increasing difficulty and tighter margins for error. Only the most skilled runners will survive the chase and live to fight for humanity's future.

Game Play Outline

Opening the game application

- Player launches *Fate/Nonstop*.
- Splash screen and logo appear.
- Main menu loads with cityscapes background, the main character, view credits, and game title.

Game options

- Toggle sound/music volume.
- Toggle windowed/fullscreen mode.
- Keybinding customization.

Story synopsis

- The year is 2025. Unknown to the world, an AI uprising is brewing. One man — a resistance leader in the near future — has become the key figure responsible for humanity's fight against the machine. A robot assassin is sent to the present day to eliminate him before he can alter the tide of war. Now, he must outrun and outwit his pursuer through the city's streets, rooftops, and hidden military facilities to survive and safeguard the future.

Modes

- **Main Story Mode:** Play through 3 story-driven levels.

Game elements

- Automatic sprinting through levels.
- Chase mechanic (robot follows the player).
- Parkour-style platforming (wall-jumping, vaulting, sliding).
- Collecting observation data (circles).
- Discovering hidden areas and collecting hidden documents.
- Precision execution (perfect flow) bar that rewards skillful movement.
- Level time and score display.

Game levels

1. **Office Buildings and Rooftops** – Starting level, introduces basic mechanics.
2. **Rooftop Escape** – Faster pace, more hazards and vertical navigation.
3. **Military Base & Weapons Lab** – Complex level design, final showdown.

Player's controls

- Jump: Spacebar, W key, or Up Arrow
- Slide: S key or Down Arrow
- Change direction to left: A key or Left Arrow
- Change direction to right: D key or Right Arrow
- Sprint: Shift key
- Wall Jump: Move toward wall and Jump (Spacebar/W/Up)
- Pause: Escape or P key

Player actions

- Auto-run (changeable directions)
- Change running directions from left to right side of screen for dynamic gameplay
- Sprint (temporary speed boost, player controlled)
- Jump
- Wall-jump
- Vault over objects
- Slide under objects or hazards
- Precision execution (perfect flow chaining)

Winning

- Reach the end of the level before the pursuing robot catches the player or time runs out.

Losing

- Player gets caught by the robot.
- Player runs into a hazard or fails a major obstacle.

End

The player's performance in collecting Observation Data and hidden documents determines the outcome of the final encounter:

- **Incomplete Observation Data:** The player narrowly escapes from the robot. The robot is then dispatched by the military but it only suffers superficial damage. The ending cutscene shows the robot severely damaged but managing to repair itself and resume the hunt, hinting that the threat is far from over.
- **Complete Observation Data only:** Using the observation data collected, the player and the military deliver significant damage to the robot, temporarily decommissioning it. However, an ending scene reveals that the machine is slowly repairing itself—leaving the future uncertain.
- **Complete Observation Data + All Hidden Documents:** Using all the data collected, the player and military deal devastating, lethal damage that completely destroys the robot. However, the final cutscene shows transmissions being sent back to the AI network—hinting that more assassins are already being prepared.

Why is all this fun?

- The **chase mechanic** delivers a constant sense of urgency and adrenaline.
- **Parkour platforming** is fluid and skill-based, rewarding player mastery.
- **Observation Data and document collection** adds depth and replayability—players will want to optimize routes to get the best ending.
- The **execution bar** provides feedback and rewards skilled movement, creating a "flow state" as players string together perfect runs.
- The game world and story offer **The Terminator/Vector-inspired dystopian atmosphere** that pulls players into a high-stakes narrative.
- Competing for **best level times** encourages replayability and skill perfection.

Key Features

- Fast-paced auto-sprint 2D platforming with parkour-inspired controls.
- Chase tension: constantly being pursued by a deadly robot.
- Observation Data and Hidden Documents impact the story's ending.
- Precision execution bar rewards skillful play.
- Timed runs encourage replayability.
- Branching endings based on collectible performance.
- Atmospheric, story-rich urban environments (city, rooftops, military base).
- Fluid player movement system that rewards learning and mastery.

DESIGN DOCUMENT

Design Guidelines

- Keep the tension **constant** — the player should always feel "chased."
- Movement must be **fluid** and **responsive**, allowing high skill expression.
- The art style should support **readability** — players must clearly see obstacles and threats.
- Robot presence should be **felt** at all times (audio cues, screen darkening, distance meter).
- The collectible system (Observation Data & Documents) must integrate naturally into level flow.
- Levels should reward exploration for advanced players without forcing it upon casual ones.
- Endings should feel impactful based on player performance.

Game Design Definitions

- **Winning:** Player reaches the end of the level before being caught or colliding against a hazard.
- **Losing:** Player is caught by the robot or dies via hazard.
- **Level transitions:** After each level, progress screen shows data collected and player proceeds to the next stage.
- **Gameplay focus:** Tense, fluid chase experience with parkour platforming and a collectible-driven ending system.

Player Definition

The player controls a **future resistance figure** being hunted through a modern urban landscape. Agile, skilled, and desperate, the player must use parkour and instinct to survive.

Player Definitions

- **Health:** No health bar; one hit = failure (caught by robot or death by hazard).
- **Weapons:** None; this is a pure escape-focused experience.
- **Actions:** Auto-run, jump, wall-jump, vault, slide, precision execution.

Player Properties

Property	**Feedback**
Movement Speed	Automatic run (base speed), Sprint (player activated — burst of higher speed, allows longer jumps and can widen distance from the robot temporarily; limited by stamina/recharge or cooldown).
Precision Execution Bar	Fills on perfect parkour and flow of movements; visual/audio feedback. Higher bar multiplies score.
Robot Distance Meter	Visual UI element shows robot proximity; screen darkens as robot nears.
Robot AI Behavior	Robot auto-chases at a base speed but dynamically accelerates if the player misses flow sequences or makes mistakes during their run and decelerates or "makes mistakes" to keep the chase going. It remains off-screen but briefly appears on-screen if the player makes a mistake, is nearly caught, or is finally caught.
Observation Data Collected	Displayed post-level; affects story ending.
Hidden Documents Collected	Displayed post-level; affects story ending if fully collected.

Player Rewards (power-ups and pick-ups)

- **Observation Data (Circles):** Main collectible — affects robot damage at the end.
- **Hidden Documents:** Unlocks more detailed lore and upgrades ending impact.
- **Precision Execution Bar Multiplier:** Higher multipliers give better score bonuses and leaderboard placement.

User Interface (UI)

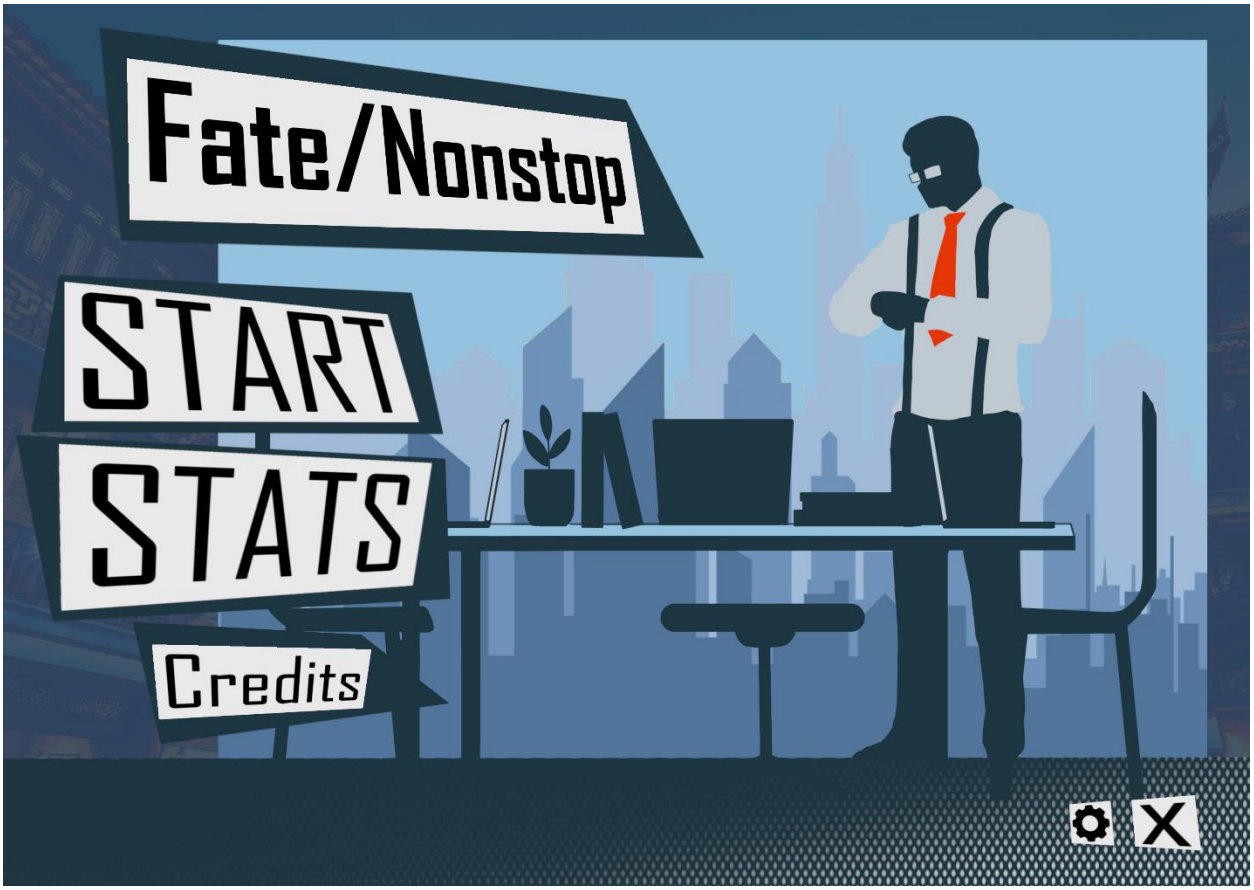


Figure 1: Conceptual Main Menu

The **Main Menu UI** shows the game's title and offers options to start the game, view player stats, or access the credits. Settings and exit options are accessible from the bottom-right corner.

.

Figure 2: Gameplay HUD (head-up display)

The **Gameplay UI** displays the player character in motion, with flow and stamina bars, collectible observation data, and quick-access restart and pause buttons.

Figure 3: In-game menu

The **Pause Menu UI** offers options to resume gameplay, access settings, restart the level, or quit to the main menu.

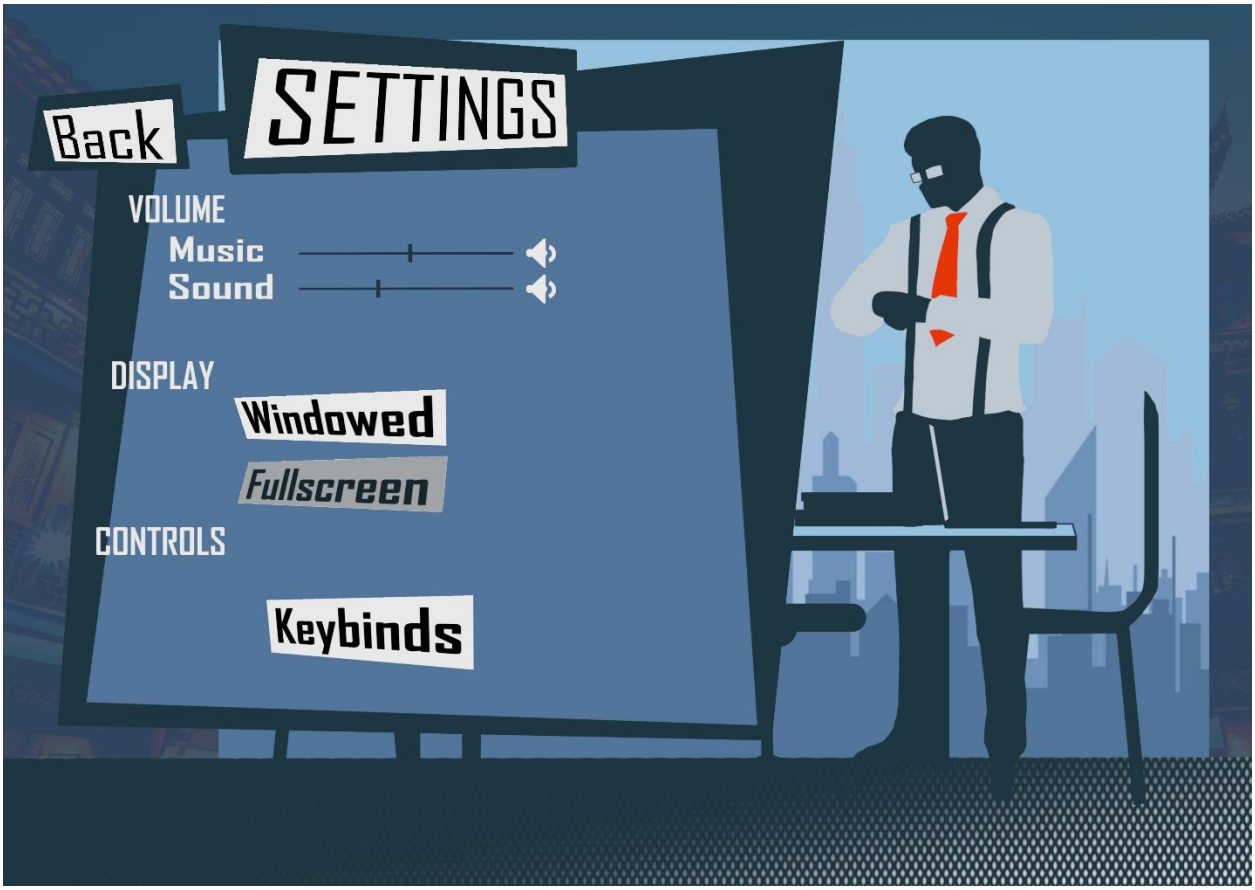


Figure 4: Settings Menu

The **Settings Menu UI** allows players to adjust volume, toggle between windowed and fullscreen display, and customize keybindings.