

# KYOTO NINJA

## *Game Design Document*

#36 Games#

## STAGE ONE : CONCEPT

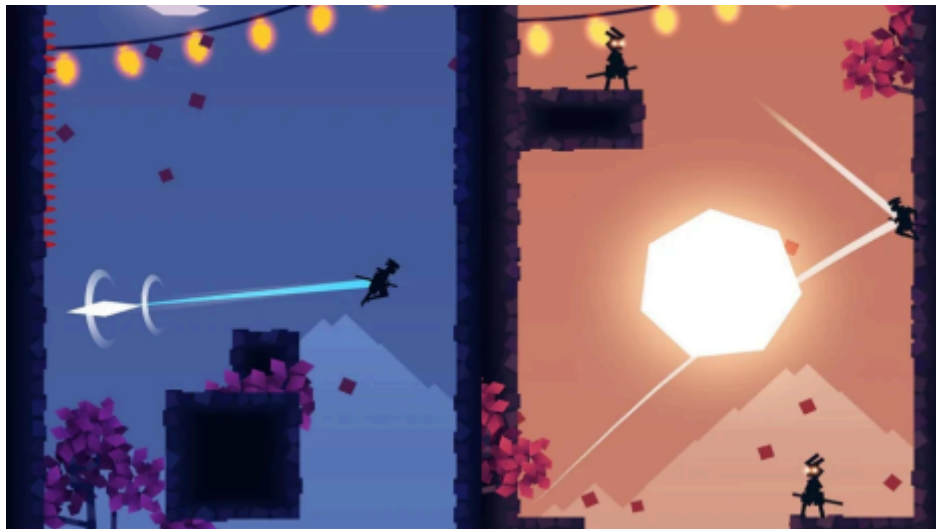
### Game Summary

This is a game where a ninja goes wall to wall to reach the top. But in his path there will be some traps and enemies that he needs to evade to arrive at the goal. It will have some power ups or extra things that will make it easier. The player has 3 lives.

### Game Lore

A 16-year-old boy in high school in Japan wakes up one day with no memory of anything. He finds himself in a room completely different from his home, one he doesn't recognize, and has no idea how he got there. Some men enter the room and tell him about his new mission, handing him a portrait of a girl his age. They tell him he has to find her and kill her.

The game follows the protagonist as he progresses through different levels, fighting monsters and enemies. When he finally reaches the girl, she reveals that he is part of an assassin organization. The monsters he has been fighting were actually humans, distorted by his own perception. She gives him an antidote so he can see reality as it truly is, and together they escape.

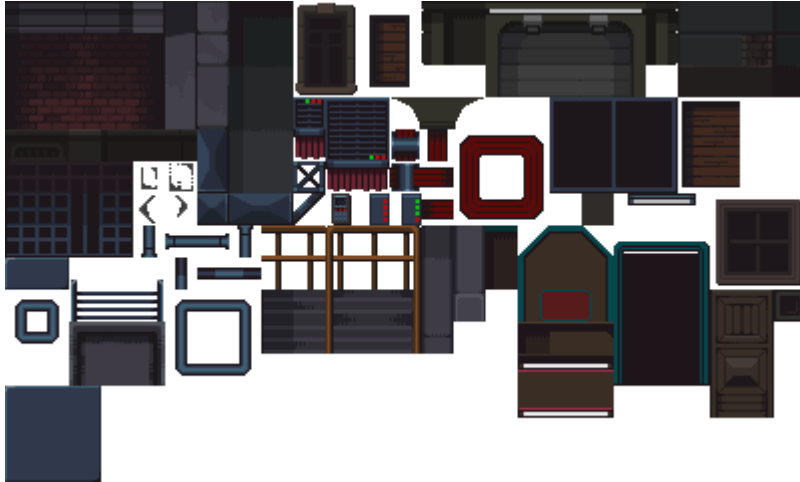


## STAGE TWO : SPECIFICATIONS

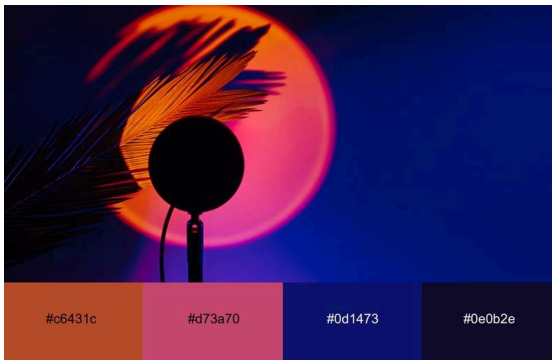
### Art Style

For the art style we used a more futuristic type of assets following this type of aesthetic





## Color Palette



## Game Mechanics

**Jump/Dash:** The main mechanic is the Jump/Dash. Its main characteristic is that the character has a number of dashes that can be upgraded by playing starting at 2, and when the player clicks or taps the phone, if it's in the air, the game starts a timer of the remaining time to perform the dash, meanwhile the game time slows. With some in-game or meta-game power ups it can change the way of playing to make it more interesting.

**Obstacles:** The main goal for the player would be to reach the highest point possible while avoiding obstacles. These obstacles can vary from enemies to static or moving obstacles. - Enemies:

- Static Melee: It stands with a frustum of what he can see and attacks the player if it enters in it.
- Static Ranged: It stands still while following the player with its ranged weapon and finally shoots.
- Flying enemy: It follows the player, if it collides with the player while standing it deals damage, but if the player is dashing the enemy dies.
- Summoner: It spawns flying enemies with a certain cooldown.
- Following ninja: It follows the player imitating its dash movement and tries to catch him. If enemy and player collide while the player is standing, enemy deals damage, but if it's in the air, enemy dies.

- Obstacles:

- Spikes
- Moving Blade
- Poison cloud
- Ranged trap

- Obstacle Contraptions:

- Moving Flying block covered in spikes
- Rotating rectangle covered in spikes
- Moving block covered in spikes with a jointed moving blade
- Flying enemy carrying a block covered in spikes
- Ranged trap that leaves a trail of poison cloud
- Poisonous enemy: killing this enemy leaves a poison cloud after a certain

time.

This minigame is a singleplayer game.

The player has 3 lives and enemies or traps can only deal 1 life. In the run you can collect coins to exchange them in the shop. Coins just sit on the map or drop when the enemy dies.

In endless mode, combinations and types of obstacles and enemies will vary depending on the height the player reaches.

- Up to 200 units: Static Melee, Static Ranged and Spikes
- Up to 500 units: Addition of flying enemies and first two Obstacle contraptions
- - Up to 1000 units: Addition of Moving blades and ranged traps
- Up to 2000 units: Addition of Summoner
- Up to 3000 units: Addition of 3rd and 4th Obstacle contraptions
- Up to 5000 units: Addition of Following ninja
- Last addition: The rest of obstacle contraptions and combinations of them

**Power Ups:** There is two types of power ups:

- Meta Power Ups: You unlock and buy them in the meta shop of the app. You can exchange coins to have these power ups and buff your player. Each power up can be upgraded several times and multiplying its cost depends on the power up each with a base price of 2 coins.

- Dash: Number of dashes the player can do in the air before landing somewhere, starting at 1.

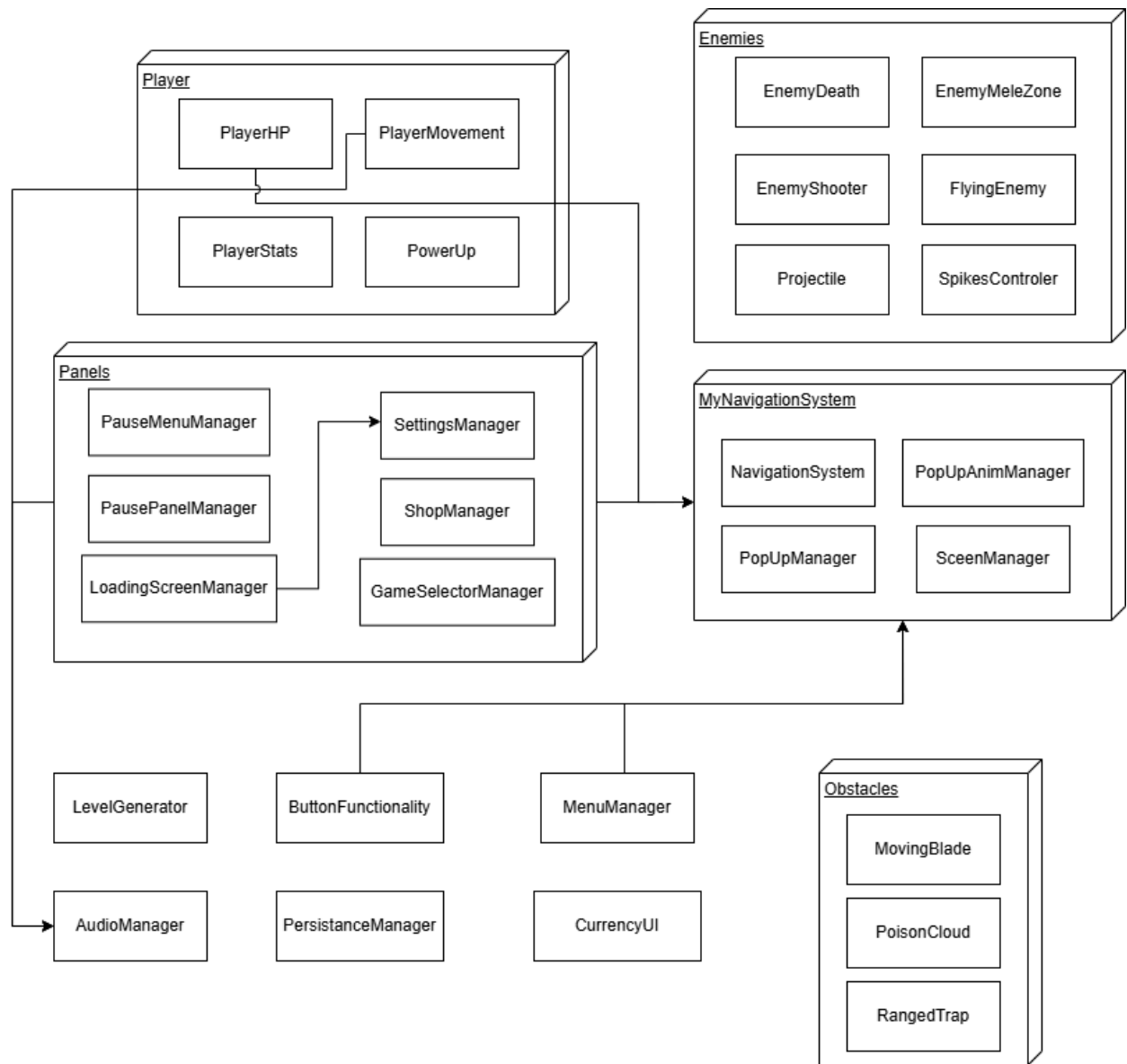
- Dash time: The amount of time the player has to perform and aim the dash
- Time Stop: The intensity of the slowdown of time when aiming the dash.
- Coin collection: The radius the player catches coins.

- **Temporal PowerUp duration: duration of the temporary power ups the player can collect during gameplay NEW!!**

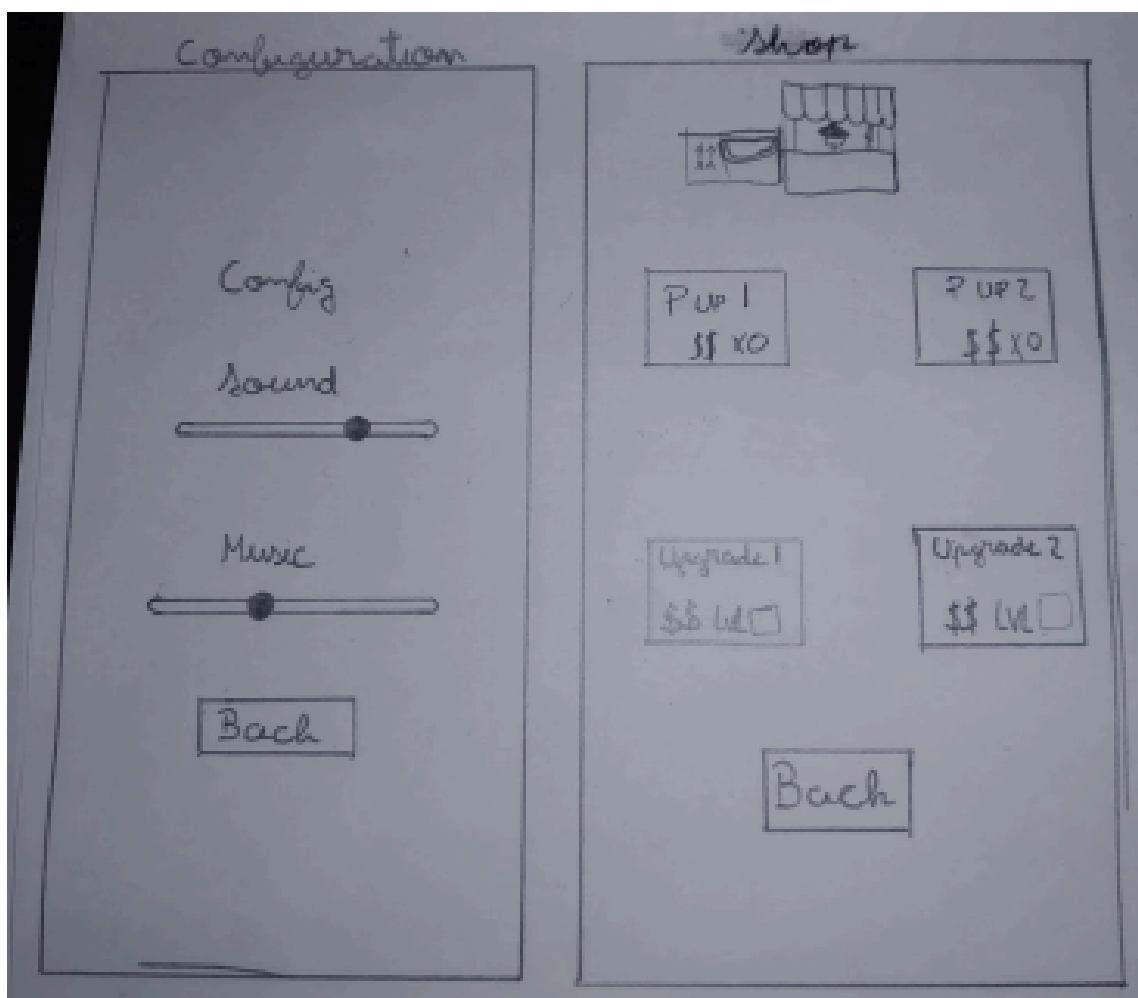
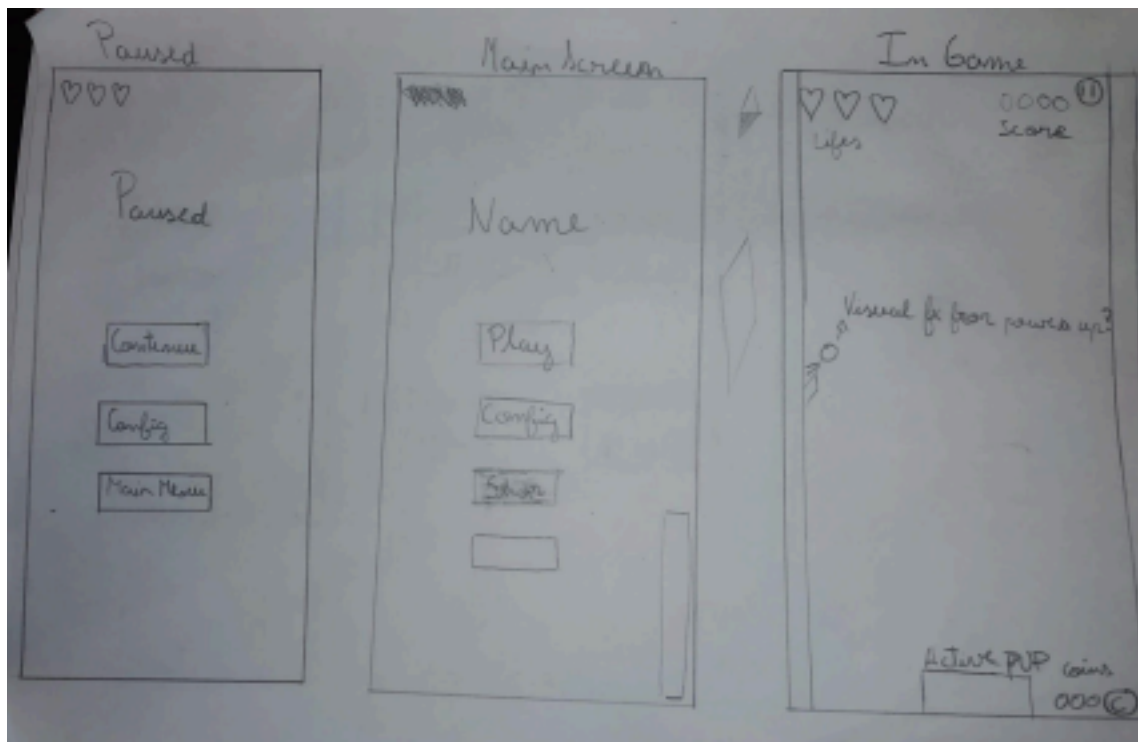
- Luck: A multiplier chance of obtaining more than 1 coin when collecting them.

- Temporary Power Ups: You can have them mid run when you defeat an enemy or at some point of the run and you have to reach it to have it. Temporary power ups are just a +1 to a random Meta Power Up for a period of time.

## UML



## UI Design



## Audio Design

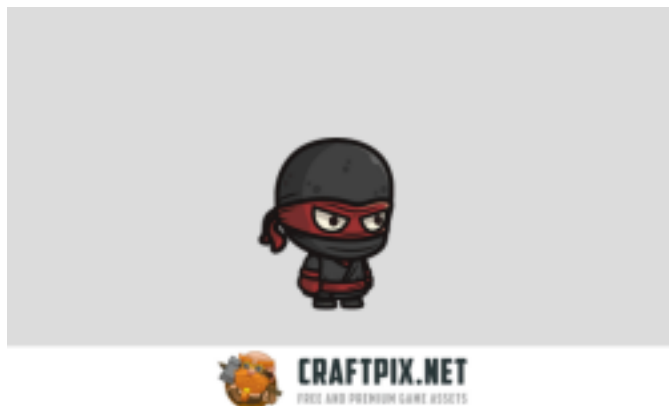
The sound design of our game aims to create an immersive experience for players. We'll use a combination of ambient sounds, realistic environmental effects, and dynamic audio systems to enhance gameplay and evoke emotions. Integration: Sound design elements will be seamlessly integrated with visual cues and narrative events to provide players with a cohesive and engaging experience.

## Asset Requirements

Audio Assets:

- Jumping SFX
- Attacking SFX
- Pause Music
- In-Game Music
- Screams of death SFX
- UI Clicks SFX

The main character controlled by the player has to be a personalized ninja asset (hence the name of the game) that simultaneously works with three different palettes of colors and reflects the isekai nature of the protagonist. Then we'll very likely need to create 3 different tileset of walls, one for each environment, a number of enemies and very large backgrounds.



For reference we have:

The Pixel Art The Chibi The Realistic.

Lastly, for the UI font, UI buttons, life hearts and point score we'll use already created asset sources from the internet.



### Specific Assets

For the Game UI we may use the [Basic Pixel Health bar and Scroll bar](#) asset pack, from this we can remark the dynamic animations of its minimalistic health bars.

For the Menu UI we may use the [Silent](#) asset pack that is especially characteristic for being extremely clean, smooth and simple making it very appealing to the users view.