

# Game Design Document

*by Rémi Adriano*

# Runes

### **Context:**

This Game Design Document was written for a Game Design test. This work is entirely mine and was done within a week following these rules:

#### **Game characteristics:**

- Designed for 8/9 years old children.
- Mobile game.
- Based in a natural environment (stone, rocks, plants, trees, rivers, etc).
- Wholesome, no violence.

Inside this game, you have to design a minigame with the following characteristics:

- 2 minutes long at most.
- Fixed camera during the minigame.
- Actions mostly done on game environment elements.
- Minigame should instantiate dynamically within the world. Specific conditions can be mandatory.
- **The game has the objective to develop the player's attention and the player's memory.**

*You are free to use any asset as long as you describe them inside the GDD.  
The document should be clear enough for a team to develop without asking any question.*

## **General description of the game:**

The player lives in a world mingling real life elements and magical elements that are shown step by step as the player progress inside the game.

The progression system contains multiple aspects:

- Experience

Experience is gathered by playing the game itself (completing quest, discovering new areas, etc). This experience allows the player to acquire new sets of skills.

- Collectible

Magical gems are hidden in the world and can be collected by playing minigames.

- Story

A story is unveiled step by step, showing a powerful magic that was unknown at the start of the game.

## **General description of the minigame:**

Gems were dispersed around the world and are protected with a powerful magic. To collect them, you need to deactivate its protection by completing a minigame. On this particular minigame, the gem is enclosed inside a tree's bark and protected by a magical shield.

## **Triggering the minigame:**

The minigame is dynamically triggered as soon as the player walks closely to a tree with an initial proc rate of 3%. This value can be change during balancing test.

## **The minigame:**

### Goal:

Break the 4 magical links protecting the gem to collect it.

### How to break a magical link:

To break a magical link, the player needs to memorise a sequence of runes which he has to recompose using neutral runes.

### What is a sequence of runes?

A sequence of runes is a combination of 12 runes. Each sequence is split in sub-sequence of 2, 3, 4 or 6 runes.

### What is a neutral rune?

A neutral rune is an inactive rune from the world. At specified moments, the rune will be active and the player can compose the sequence he memorised by tapping those runes.

### Flow:

- a. The sequence to memorise is shown to the player for 7 seconds and all neutrals runes are deactivated.
- b. The sequence to memorise disappear.
- c. A set of neutral runes activates for 5 seconds. The player taps the runes in the correct order corresponding to the sequence he memorised.
- d. All neutral runes deactivate and a new set of neutral runes is available to the player.
- e. When the sequence is correctly entered by the player, a magical link breaks and a new sequence has to be memorised.

Every 15 seconds, the sequence is shown to the player to allow him to continue playing.

Every time the player completes a sub-sequence, the entire sequence is shown.

### Win condition:

The player breaks all 4 magical links.

### Lose condition:

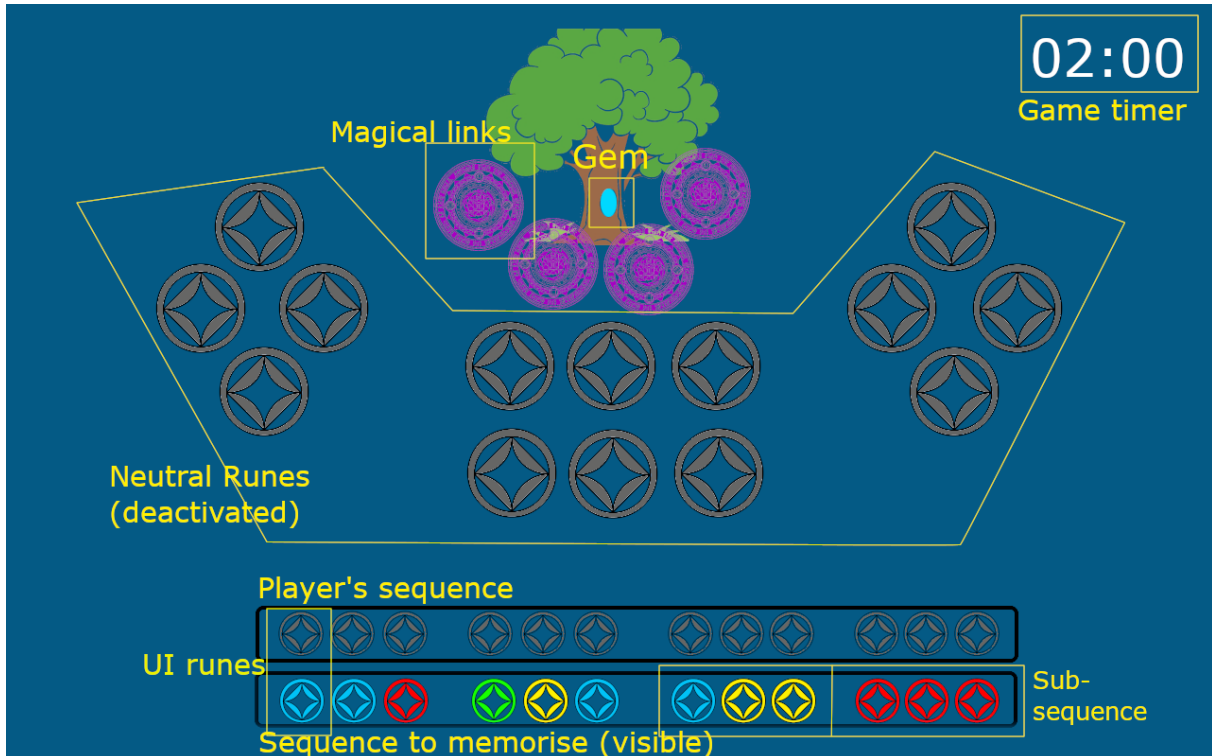
The player is not able to break all 4 links before the time runs out.

### Reward:

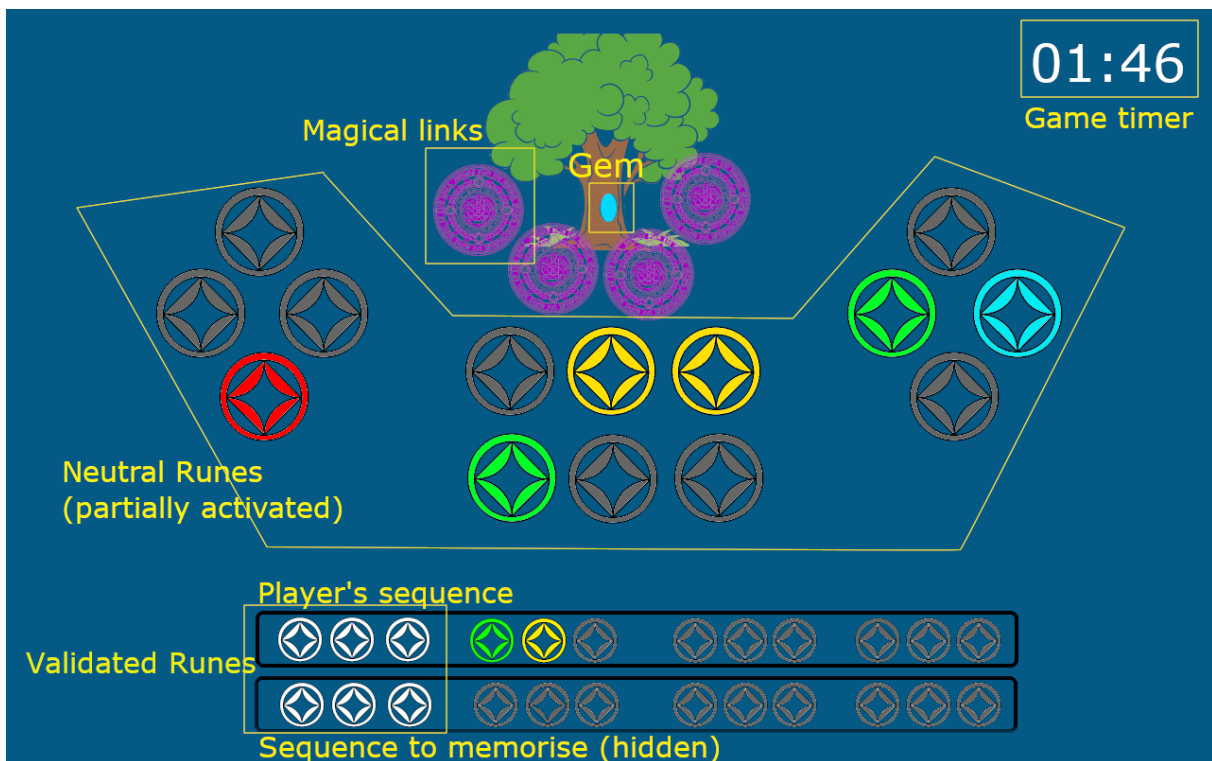
The player gains experience based on the number of magical links he was able to break. If all 4 links are broken, the player wins the gem.

## Game Design:

Sample screen:



Screen 1: initial state when minigame starts



Screen 2: playable state

## Game Design Elements:

### Neutral Rune (Environmental element):

Neutral Runes are the elements the player is going to use to make the sequence he memorised by tapping them.

A neutral rune has 2 states:

- **Deactivated rune:** The rune is passive (appears grey). It cannot be selected by the player.
- **Activated rune:** The rune is active (appears with a colour). It can be selected by the player.

### UI Rune (UI element):

The UI Rune is a user interface element representing a neutral rune. Those runes are used in the player's sequence and the sequence to memorise. It has 3 states:

- **Selected rune:** it represents a rune selected by the player to recompose the sequence. The colour is bright.
- **Unselected rune:** It represents a rune that is not yet selected in the sequence. The colour is dark and transparent.
- **Validated rune:** It represents a rune that is confirmed correct by the game. It appears white and cannot be modified anymore.

### Sequence to memorise (UI element):

The sequence to memorise is the solution to the minigame. The player needs to remember the entire sequence (or sub-sequence) to solve the minigame.

The sequence to memorise is a sequence of 12 UI runes randomly generated and is split in sub-sequences of different length according to the player's progression as follow:

Link	Split pattern	Sequence format
1 <sup>st</sup> link	2 x 6	OO – OO – OO – OO – OO – OO
2 <sup>nd</sup> link	3 x 4	OOO – OOO – OOO – OOO
3 <sup>rd</sup> link	4 x 3	OOOO – OOOO – OOOO – OOOO
4 <sup>th</sup> link	6 x 2	OOOOOO - OOOOOO

*O: Represents a UI rune*

The sequence to memorise has 2 states:

- **Visible:** The sequence is visible by the player.

This state is used on the following situations:

Situation	Duration
Minigame starts.	7 seconds
A magical link breaks and a new sequence to memorise is generated.	7 seconds
A sub-sequence is valid.	3 seconds
The sequence to memorise has not been shown for the last 15 seconds (the player is stuck).	5 seconds

- **Hidden:** the player cannot see the sequence to memorise.

This state is used during all undescribed situations.

### The player's sequence (UI element):

The player's sequence is a sequence of 12 UI runes and is the result of the player's actions. This sequence can be modified with the following actions:

- **Add a rune to the sequence:** When the player taps an active neutral rune from the environment, the rune is added to the player's sequence. If the player taps a green neutral rune, a green UI rune is added to the player's sequence **on the first available slot found starting from the left.**
- **Remove a run from the sequence:** When the player taps a selected UI rune from the player's sequence, the rune is removed from his selection, allowing him to correct the sequence.

A validation check runs every time a sub-sequence is filled from the player's actions.

- **If the sub-sequence is correct,** all the UI runes from the sub-sequence are validated and appear white. These runes will no longer be shown to the player.
- **If the sub-sequence is incorrect,** nothing happens and the player understand he made a mistake. He is able to change his selection.

When the entire player's sequence matches the sequence to remember, a link breaks and a new sequence is generated with a new format as previously mentioned.

### Magical Links (Environmental element):

Magical links represent the progression of the player during the minigame. The player cannot interact with them.

When the player's sequence matches the sequence to remember, a link breaks and disappears from the screen.

When all links are gone, the minigame ends.

### Game timer (UI element):

The game timer represents the remaining time given to the player to complete the minigame.

The timer stops during the very first presentation of a new sequence to memorise.

The timer starts after this very first presentation and never stops.

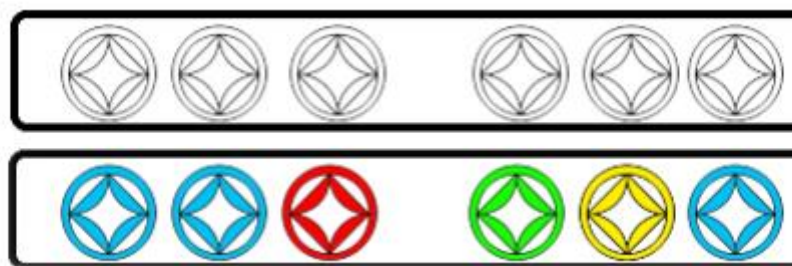
The player can access the abandon screen by tapping the Game timer.

### Neutral Runes grid (System):

The Neutral Runes grid system is composed of 14 Neutral Runes that activates and deactivates on a 5 second cycle. Each cycle runs on the following rules:

- A random selection of 6 runes activates on each cycle.
- The random selection needs to allow the player to progress on his selection for at least 3 steps.

#### Example:



*The grid has to propose 2 blue runes and 1 red rune on his next cycle. If the player selects the first two correctly, the next cycle will offer at least 1 red rune, 1 green rune and 1 yellow rune to allow the player to complete his selection correctly and avoid any softlock situation.*



## **Game flow:**

### **Instantiation:**

The player is walking inside a forest. When he gets close to a tree, the game decides if it has to instantiate the minigame. If true, the player loses the control of the character, the gem appears on the tree's bark. The character walks to the gem, tries to pick it and the magical protection appears as a protective shield. The minigame instantiates, the magical links appears on the ground, the rune grid is shown and the player's UI changes for the minigame's UI.

### **Initial state:**

- Game timer is set to 2 minutes.
- All neutral runes are deactivated.
- The sequence to remember is randomly generated and shown to the player for 7 seconds.
- The player's sequence is empty.

The player memorise the sequence and the game timer start.

### **Game loop:**

- The sequence to remember is hidden.
- The neutral runes grid starts its first cycle, selecting 6 runes and renewing the selection every 5 seconds.
- The player interacts with the different elements to complete the challenge.

The game loop goes on as previously described until the player meet a winning or a losing condition.

### **End state:**

Two different scenarios are possible for the end:

- The player did not have the time to break all 4 magical links before the game timer expires. He gains experience based on the number of links he broke as a reward.
- The player had the time to break all 4 magical links. He gains the maximum experience and the gem.

Regardless of the player's result, the minigame's UI disappear and a recap of the reward is shown on the screen for a few seconds. The player can control his character again, the camera goes back to normal, end all elements belonging to the minigame disappear (gem, runes, etc).

## Reward:

Completing the minigame gives the player a reward on every scenario at the only exception of abandoning. A quantity of experience is generated by the game and a multiplier is applied depending on the player's progression inside the minigame. For example:

- The game selects a base XP value of 1000.
- A multiplier applies based on player's results.

Base XP	Abandon	0 link	1 link	2 links	3 links	4 links
	x 0	x 0.3	x 1	x 1.2	x 1.5	x 3
1000	0	300	1000	1200	1500	3000

If the player breaks all 4 links, he gains the gem.

## Inputs:

There is a single input in the minigame: tapping the screen

- Tapping an active neutral rune from the grid will add the rune to the player's sequence on the first empty slot starting from the left.
- Tapping a rune from the player's sequence remove the rune from the selection.
- Tapping the game timer shows the abandon button.
- Tapping the abandon button ends the minigame.

## Game variables and gameplay evolutions:

In order to improve the player's experience through the game, the gameplay will change according to the player's results. Those results need to be saved and the game adjust the difficulty by itself.

### Analysis data:

To adjust the difficulty of the game, the following data are saved:

- Minigame iteration count.
- Game completion and Cumulative game completion.

*Game completion* is defined by the number of broken links and is calculated as followed:

	Abandon	0 link	1 link	2 links	3 links	4 links
Game Completion		0	0.25	0.5	0.75	1

*Note 1: Abandon does not count.*

*Note 2: Game suddenly ending (game crash, hardware crash, etc) are considered abandoned.*

**Cumulative game completion** is the sum of all game completion values.

	Game 1	Game 2	Game 3	Game 4	Game 5	Cumulative Game Completion
Player 1	1	1	0.75	0.75	1	4.5
Player 2	0.75	0.25	0.5	0.5	0.75	2.75
Player 3	0.25	0.5	0.5	0.25	0.25	1.75

Owing to the cumulative game completion and the count of minigame iterations, you can easily have an average completion rate that can be used to adjust the game difficulty.

### **Game variables:**

To adjust the difficulty of the minigame, the following balancing variables need to be changed:

- Game timer.
- Number of colours available.
- Time to memorise the sequence.

Those values are registered inside a preset structure and define the game difficulty.

Preset 1	Preset 2	Preset 3
Time: 180 seconds Colours: 2 Memorisation: 10 seconds	Time: 180 seconds Colours: 3 Memorisation: 10 seconds	Time: 150 seconds Colours: 3 Memorisation: 8 seconds
Preset 4 (base)	Preset 5	Preset 6
Time: 120 seconds Colours: 4 Memorisation: 7 seconds	Time: 120 seconds Colours: 5 Memorisation: 7 seconds	Time: 120 seconds Colours: 5 Memorisation: 5 seconds
Preset 7	Preset 8	Preset 9
Time: 90 seconds Colours: 6 Memorisation: 7 seconds	Time: 90 seconds Colours: 6 Memorisation: 6 seconds	Time: 90 seconds Colours: 7 Memorisation: 6 seconds

### **Preset selection:**

All players start with the preset 4. Every 5 games, adjustments are made accordingly to the average completion rate, following these rules:

- Average completion rate is above 80%
  - ➔ The minigame is too easy, the next preset will apply for the 5 next games.
- Average completion rate is between 40% and 80%
  - ➔ The difficulty is correct, nothing changes for the 5 next games.
- Average completion rate is below 40%
  - ➔ The difficulty is too high, the previous preset will apply for the 5 next games.

	Game count	Cumulative Game Completion	Average Completion Rate	Current Preset	Next Preset
Player 1	5	4.5	90%	Preset 5	Preset 6
Player 2	5	2.75	55%	Preset 4	Preset 4
Player 3	5	1.75	35%	Preset 6	Preset 5

Note: Data resets every 5 games.

### **Art:**

This minigame needs artistic assets.

### **Character:**

- Walking animation from the character's position to the gem
- Character raising the gem in the air when obtaining it (i.e: *link* raising his new sword).

### **Gem:**

A 3D gem needs to be attached to every tree's bark in the game. It should look like a precious stone.

Two animations are required for the gem:

- Shield activation when the minigame starts.
- Shield deactivation if the player completes the minigame.

### **Magical links:**

The magical links belongs to the world environment. It acts as a power source for the gem's protection. It is a circle on the ground coloured with a violet / red tint, reminding black magic.

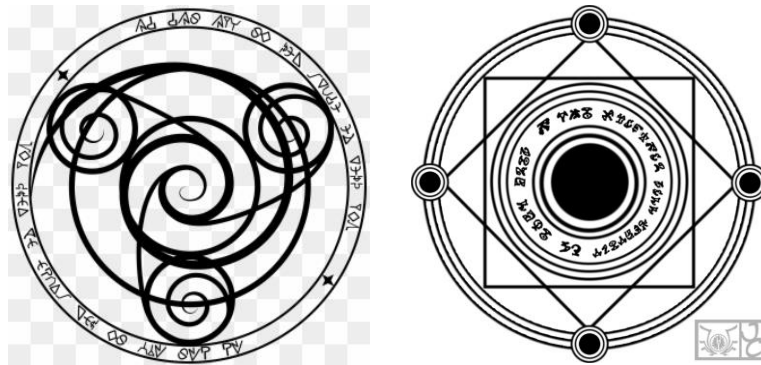


The magical links need 2 animations:

- Activating the link between the ring and the gem when the minigames initiate.
- Deactivating the link when a sequence is correctly completed.

## Neutral runes:

Neutral runes are an element from the game environment and needs a 3D mesh. The initial state has no colour and it should be possible for to game to assign one dynamically.



The neutral runes need 2 animations:

- **Activation:** the rune switches from deactivated (no colour, no movement) to activated (colour, movement). An inner part of the rune starts rotating to emphasize the activated state.
- **Deactivation:** the rune switches from activated (colour, movement) to deactivated (no colour, no movement). The inner part stops rotating.

## UI Runes:

UI runes belongs to the minigame's UI. They are a 2D representation of neutral runes and needs to have 3 different representations:

- **Validated:** the rune turns white with full opacity.
- **Selected:** the rune is assigned with a colour and full opacity depending on the player's selection.
- **Unselected:** the rune has a dark grey colour with half opacity.

## Sound:

The general ambient sound remains the same during the minigame. The music keeps playing.

Multiple sounds are required to give the player audio feedback alongside the visual feedback:

- Player touches a neutral rune from the grid.
- Player removes a rune from his selection.
- Player completes a sub-sequence.
- Player completes a full sequence.