Légendes Pokémon : Giratina

*Guide de la production*

*La version PDF est sans doute plus stable que la version DOCX*

# SOMMAIRE

|  |  |
| --- | --- |
| PRESENTATION GLOBALE | 2 |
| MODELISATION | 3 - 8 |
| REALISATION | 9 |
| EXEMPLE | 10 - 13 |
| BILAN | 14 |

# Présentation globale

J’ai choisi de prendre le sujet *“Histoire dont tu es le héros”* pour créer une histoire sur un jeu Pokémon.

Cette histoire mythologique s’inspire du jeu qui sortira début 2022, “[Légendes Pokémon : Arceus](https://legends.pokemon.com/fr-fr/)”. Celle-ci est centrée sur le personnage de [Giratina](https://www.pokepedia.fr/Giratina).

Vous incarnerez le personnage d’Arceus et vous devrez sauver le monde d’une erreur, Giratina. Vous devrez menez l’enquête dans les régions de [Sinnoh](https://www.pokepedia.fr/Sinnoh) et de [Hoenn](https://www.pokepedia.fr/Hoenn) d’antan pour réussir à enfermer Giratina.

# Modélisation

Histoire

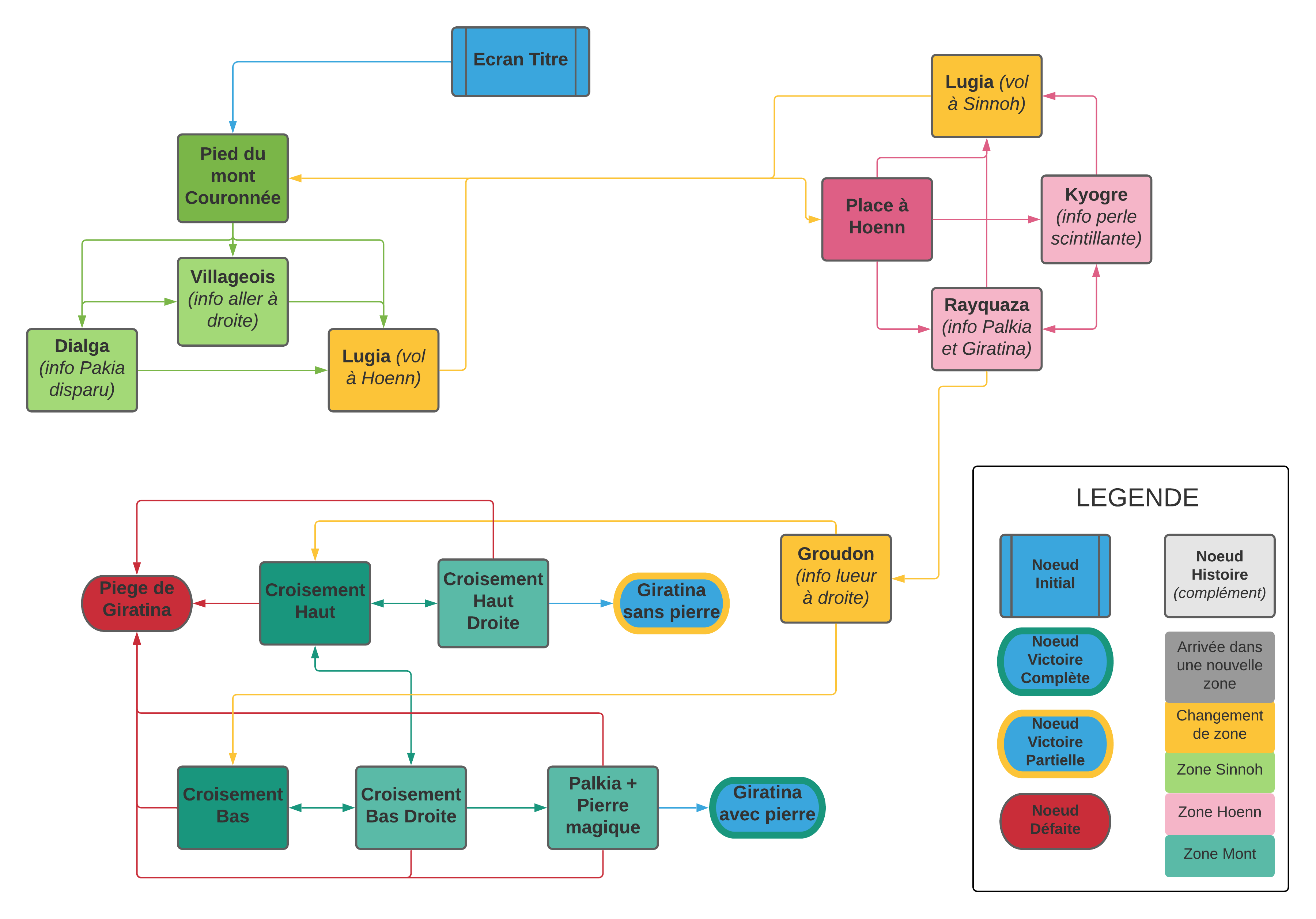
L’histoire du jeu est stockée dans un fichier JSON (res/data/data.json) pour plus de simplicité.

Un nœud du graphe correspond à une portion de l’histoire aboutissant à un choix. L’histoire du jeu contient des informations pour chaque nœud tels que :

|  |  |  |
| --- | --- | --- |
| NOM | DESCRIPTION | TYPE |
| mode | Représente le mode du nœud :  **0** correspond au nœud d’écran titre  **1** aux nœuds d'histoire classiques  **2** aux nœuds de victoire partielle  **3** aux nœuds de victoire parfaite  **4** au nœud de défaire | Nombre entier |
| histoire | Liste l’intégralité des phrases à dire lors de l’histoire pour chaque nœud. | Liste de chaîne de caractères |
| choix | Liste les deux choix pour chaque nœud.  Ils sont listés dans le même ordre que les choix associés dans *adj*. | Liste de chaine de caractères |
| image | Nom de l’image à afficher et présente dans le dossier “res/image”. | Chaine de caractères |
| bruitage | Nom du bruitage à jouer et présent dans le dossier “res/music”.  S'il n’y a pas de bruitage à jouer, la chaine de caractère est égale à vide. | Chaine de caractères |
| musique | Nom de la musique à jouer et présent dans le dossier “res/music”.  S'il n’y a pas de musique à jouer, la chaine de caractère est égale à vide et la musique de change pas. | Chaine de caractères |
| adj | Liste des nœuds connexes.  Ils sont listés dans le même ordre que les choix associés dans *choix*. | Liste d'entiers |

Les données de ce fichier sont ensuite récupérées par la méthode start() de l’objet MoteurJeu pour être utilisé dans le programme.

Voici à quoi ressemble l’histoire :



L’objet NoeudGraphe

Chaque nœud du graphe est l'objet NoeudGraphe du programme. Il représente une portion de l’histoire.

Il comporte une méthode :

|  |  |
| --- | --- |
| NOM | DESCRIPTION |
| \_\_init\_\_ (se lance à la création d’un objet NoeudGraphe) | Attribue des diverses caractéristiques du nœud tel que le mode, l’histoire, le choix, l’image, le bruitage et la musique grâce au paramètre valeurs (reprenant les données de data.json [hormis l’adjacence]) |

L’objet MoteurJeu

Le moteur du jeu est l’objet MoteurJeu. Il permet le parcours du graphe, le lancement d’une partie, le lancement des sons, l’affichage et la boucle de jeu principale.

La liste d’adjacence, représentée par l’attribut adj de l’objet MoteurJeu est construite sous cette forme :

[NoeudGraphe, *adjacence 1, adjacence 2, ...*]

Il comporte plusieurs méthodes :

|  |  |
| --- | --- |
| NOM | DESCRIPTION |
| \_\_init\_\_ (se lance à la création d’un objet MoteurJeu) | Appelle la méthode start() permettant de lancer une partie. |
| start | Importe les données du fichier “data.json” puis crée le graphe.  Appelle la méthode mainLoop() pour lancer la partie. |
| jouerBruitage | La méthode jouerBruitage() permet de jouer un bruitage si nécessaire. |
| jouerFond | La méthode jouerFond() permet de jouer un bruitage si nécessaire. |
| deplacement | Permet de se déplacer dans le graphe en modifiant le numéro de la ligne. |
| verifChoix | Vérifie si un choix donné est valide. |
| verifSouris | Vérifie si un clic est sur un bon choix. |

|  |  |
| --- | --- |
| mainLoop | Crée la fenêtre pygame avec son titre et ses dimensions.  Lance une boucle tant l’utilisateur de désire jouer :   * Commande le lancement des sons s’il y en a, * Commande l’affichage de deux phrases histoires une-à-une à partir d’un clic et joue un bruitage de de clic, * Commande l’affichage les et attend une sélection d'un des choix à partir d’un clic, * Commande le déplacement dans le graphe. |
| afficher | Gère l’affichage graphique.  Affiche l’image de fond et fond gris du bas.  Affiche l’histoire dans une bulle ou affiche les choix dans des rectangles colorés.  Affiche le curseur Pokéball. |

Multimédia

Les multimédias sont utilisés par l’objet MoteurJeu :

|  |  |
| --- | --- |
| NOM | DESCRIPTION |
| Bruitage | La méthode jouerBruitage() permet de jouer un bruitage si nécessaire. |
| Musique | La méthode jouerFond() permet de jouer un bruitage si nécessaire. |
| Image | L’image de fond est chargée à chaque actualisation de l’affichage par la méthode afficher(). |
| Police | Les polices utilisées pour afficher les textes (pkmndpb.ttf et pkmndp.ttf) sont présentes dans le dossier “res/font”.  Elles sont utilisées dans la méthode afficher(). |

# Réalisation

Une partie classique doit fonctionner comme ceci :

* Au lancement du programme, la liste d’adjacence est générée depuis les données de “data.json”.
* La fenêtre pygame est initialisée et un premier affichage avec l’image et la 1ère ligne de texte est effectué.
* S’il y a une musique ou un bruitage à jouer, ils le sont.
* A chaque clic de l’utilisateur, une nouvelle ligne d’histoire est affichée et un bruitage de parole joué.
* Lorsqu’il n’y a plus d’histoire à afficher, l’affichage passe en mode “question” et affiche les choix possibles.
* Le choix survolé est affiché en rose.
* L’utilisateur clique sur son choix et selon celui-ci, on se déplace dans le graphe.

*-> Si l’utilisateur clique autre part que sur une case-choix, rien ne se passe.*

* L’image, l’histoire, les sons et les questions du nouveau nœud sont affichés ou joués, ainsi de suite, jusqu’à ce que le joueur arrive à perde ou gagne.
* Dans ce cas-là, les choix sont soit d’arrêter la partie ou d’en recommencer une.

# Exemple

Voici un extrait d’une partie du jeu :

|  |  |
| --- | --- |
| N° | IMAGE + EXPLICATIONS |
| 1 | Voici le premier nœud, il s’agit du nœud écran titre donc l’écriture est en bleu.  L’image de l‘écran titre est inspirée du vrai jeu et une musique est jouée.  Je découvre grâce au texte qu’Arceus sauver le monde de Giratina. Je fais défiler le texte par des clics. |
| 2 | Je choisis de commencer la partie. L’option survolée est mise en rose. |
| 3 | La couleur **noire** m’indique que le nœud est de type histoire. Je lis les histoires attentivement pour ne rater aucun indice ! |
| 4 | Je peux changer de zone grâce à Lugia. Il m’emmène de Sinnoh à Hoenn et inversement. |
| 5 | Les choix sont représentés par de nombreuses couleurs. |
| 6 | La partie s’arrête si je perds ! Le texte est alors rouge, il est jaune en cas de victoire partielle et vert en cas de victoire totale ! |
| 7 | onMon score s’affiche.  Je peux alors choisir entre réessayer et quitter le jeu. |

# Bilan

J’ai beaucoup aimé ce projet car j’ai pu de nouveau choisir de faire un mimi-jeu sur un thème qui m'intéressait beaucoup !

*Bon jeu :)*

