

# Role PL'AI

## Motivation

Dans les Jeux De Rôles il y a plusieurs actions longues et pas forcément intéressantes qui peuvent gâcher le fun de l'expérience. Automatiser ces tâches permettrait de lancer plus facilement des parties et de se concentrer sur le jeu

## Problème

La création de fiches personnages, la création de quêtes ou encore la gestion des événements sont des tâches longues et qui peuvent nuire à l'expérience du JDR.

## Solution

Un chatbot avec des agents différents qui génèrent des éléments utiles qui seront ensuite affichés dans une interface pour tout avoir sous la main. Les différentes informations générées sont ensuite réinjectées dans les chatbots pour créer de la cohérence. A long terme, l'idée est de remplacer le MJ (Maître du Jeu). des action préenregistrées peuvent également permettre de faciliter la progression

## Réflexion

Dans un premier temps, j'ai commencé par la création des fiches des personnages. Lorsqu'on est dans cette phase, un premier agent va générer un fichier json prédéfini avec toutes les informations que l'on demande. Ces informations sont rentrées dans le store et une interface vient les récupérer pour les afficher. Les fiches personnages sont également rentrées dans le contexte de discussion du chatbot pour qu'il se rappelle toujours des statistiques des personnages.

Ensuite, lorsqu'on veut commencer la partie on peut rentrer un contexte dans un textarea qui sera envoyé dans un prompt "commence l'histoire : [contexte]. L'agent changera pour celui qui raconte l'histoire.

Quand on est rentré dans l'histoire, chaque prompt est accompagné de paramètres à rentrer et est suivi d'un lancer de dé. Les paramètres entrés sont les équipements ou les pouvoirs du personnage récupéré sur sa fiche en json, il faut également rentrer la capacité dont le score sera comparé au lancer de dé, tous ses paramètres vont créer un prompt de cette forme:

```
`${
    $fichesPersos.get()[persoId].personnage.nom
} tente l'action : ${prev} ${
    utils !== 'Rien' ? ` avec ${utils}` : ''
}, c'est ${etat}`
```

prev: le prompt de base, l'action à réaliser

utils : l'objet ou le pouvoir utilisé

etat : "une réussite" ou "un échec" calculé en comparant le score de caractéristique avec le lancer de dé

```
{
  "updatepv": {
    "char": "Davidou le Hibou",
    "pv": -3
  }
}
```

Lors de sa réponse, le deuxième agent d'histoire peut, si le contexte le permet, augmenter ou réduire les PV (points de vie) en renvoyant un json formaté. Cela marche et dans l'idée il faudrait ajouter la même mécanique pour l'énergie, les points d'armure...

## Conclusion

L'application dans les grandes lignes fonctionne, je pense que pour qu'elle soit optimale, il faudrait un modèle plus performant mais également ajouter plusieurs interactions entre les réponses de l'agent et les fiches personnage, par exemple, pouvoir ajouter ou retirer des objets de l'inventaire du personnage de la même façon que pour les points de vie.

Je pourrais également améliorer l'UI pour que l'expérience soit plus immersive

