

# Développement d'un chatbot pour un Escape Room numérique sur la santé mentale

*Stage de fin d'études - Document de spécifications*

Jalal IZEKKI

jalal.izekki@enseirb-matmeca.fr

Tuteur de stage : M. Laurent ANDRIAMIFIDY

**16 juillet 2022**

*Enseirb-Matmeca  
Département Informatique  
Option intelligence artificielle*

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Présentation de l'entreprise . . . . .	3
1.2	Service occupé durant le stage . . . . .	3
<b>2</b>	<b>Environnement de travail</b>	<b>4</b>
2.1	Méthode de gestion du projet . . . . .	4
2.2	Technologies utilisées . . . . .	4
<b>3</b>	<b>Présentation du sujet</b>	<b>5</b>
<b>4</b>	<b>Étapes de la réalisation</b>	<b>5</b>
<b>5</b>	<b>Avancement de la réalisation</b>	<b>6</b>
<b>6</b>	<b>Conclusion</b>	<b>7</b>
	<b>Annexes</b>	<b>7</b>
<b>A</b>	<b>Outils de gestion de projet</b>	<b>7</b>

# 1 Introduction

## 1.1 Présentation de l'entreprise

**Tricky** est une entreprise universitaire qui développe la prévention expérientielle dans le domaine de santé. Créée en 2017, **Tricky** propose des *escape games* physiques et digitaux afin de favoriser les comportements vertueux en santé, diminuer les arrêts maladie et réduire les accidents de travail. Un *escape game* est composé d'un *Escape Room* dont les participants vivent une immersion individuellement ou en groupe afin de résoudre un enigma qui met à l'épreuve leurs émotions et raisonnements. Celui-ci est suivi d'un *débriefing* pendant lequel les participants discutent avec un médiateur en santé, l'objectif étant de libérer la parole des participants pour lever les doutes sur les enjeux de la santé au travail. Les participants sont suivis dans le temps afin d'évaluer leur changement de comportement.

L'équipe de **Tricky** est composée de 16 personnes (dont 4 stagiaires), réparties sur 4 services principaux : produit, ventes, exploitation et ressources humaines (figure 1).

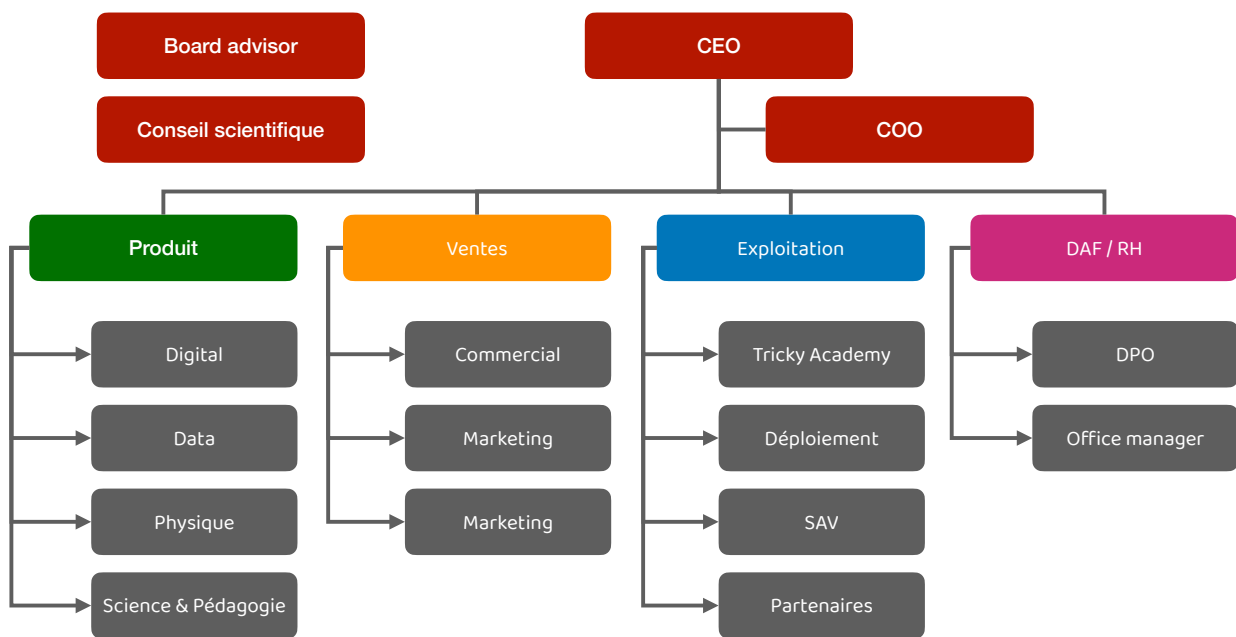


FIGURE 1 – Organigramme de l'entreprise **Tricky**

## 1.2 Service occupé durant le stage

Le service Produit de **Tricky** est composé de plusieurs équipes, à savoir :

- une équipe digitale qui travaille sur le développement de la partie informatique des jeux virtuels et des sites d'administration.
- une équipe data dont la mission est de collecter, traiter et analyser les données.
- une équipe physique qui développe la partie mécatronique des *escape rooms* physiques.

- une équipe science & pédagogie dont le rôle est l'analyse des sujets de prévention, la scénarisation et la *gamification* des scénarios ainsi que la rédaction des recommandations personnalisées selon les résultats des participants.

En tant que stagiaire en *data science*, j'interviens dans les deux équipes ; digitale et data afin de concevoir, développer et déployer un chatbot qui servira pour les sessions virtuelles d'un jeu collaboratif afin d'accompagner les utilisateurs dans les différentes étapes, répondre à leurs questions techniques, les aider à s'orienter dans les étapes du jeu en leur donnant des indices en cas de besoin et leur présenter des informations qui leur permettront de s'apercevoir des enjeux de santé.

## 2 Environnement de travail

### 2.1 Méthode de gestion du projet

La gestion du projet est effectuée à l'aide de **Github Projects**, qui est un système de suivi de projet sur **Github**. Il permet de séparer les tâches et de les attribuer à des *sprints* et suivre la progression du développement. Cet outil a été utilisé en complément de *Github Issues* et *Github Pull requests* (voir les figures 2, 3 et 4 de l'annexe) qui permettent d'indiquer au reste de l'équipe les modifications apportées à une branche dans un dépôt sur *GitHub* et de discuter et examiner les modifications potentielles à apporter au code.

### 2.2 Technologies utilisées

La réalisation de ce projet a fait appel à plusieurs technologies, que ce soit pour l'apprentissage automatique ou au niveau de l'infrastructure :

- **TensorFlow** : un outil open source utilisé pour l'apprentissage automatique, il a été utilisé dans le cadre du projet pour construire les réseaux de neurones profonds utilisés pour prédire les réponses du chatbot.
- **Django** : un *framework* open source de haut niveau utilisé pour le développement web en Python. Il a été conçu pour rendre la création des applications web plus rapide et avec moins de code. Un projet est composé de plusieurs applications qui implémentent chacune un groupe de fonctionnalités autonomes. Il est utilisé par certains sites web les plus fréquentés comme **YouTube**, **Spotify** et **Instagram**.
- **Docker** : un outil qui permet d'empaqueter une application et ses dépendances dans un conteneur virtuel isolé. Celui-ci peut s'exécuter sur n'importe quel ordinateur. Le but à terme est de déployer une image **Docker** avec l'application du chatbot sur un serveur de développement sous **Kubernetes**<sup>1</sup> et ensuite sur un serveur de production. La configuration de l'infrastructure **Kubernetes** sera commune à tous les projets en cours de développement et n'est pas incluse dans les missions du stage.

---

1. **Kubernetes** est une solution d'orchestration de conteneurs applicatifs, il permet de déployer des applications conteneurisées sur tout type d'infrastructure IT et de gérer de façon centralisée les différentes ressources dont elles ont besoin. Certaines de ces informations sont extraites de [cette page web](#).

- **Poetry** : un outil de gestion de dépendances et leur version. Il permet de déclarer les dépendances d'un projet et de les installer/mettre à jour dans un environnement virtuel isolé. **Poetry** se caractérise par ses commandes intuitives et faciles à utiliser ainsi que par son comportement configurable.

### 3 Présentation du sujet

Une session d'un *escape game* se fait en 3 étapes :

- Une première partie pendant laquelle les participants sont accueillis par un médiateur et invités à remplir un questionnaire.
- Une deuxième étape où les joueurs entrent dans le jeu et rassemblent des indices qui leur permettent de passer d'une salle à l'autre.
- Une dernière étape où le médiateur demande aux participants de répondre à des questions et donner leur avis sur les différentes thématiques du jeu. Le médiateur essaye de corriger les fausses idées que peuvent avoir les participants et leur proposer des astuces qui leur permettront de favoriser des comportements vertueux de santé.

Le but du projet est de construire un chatbot pouvant dialoguer avec les utilisateurs pendant les sessions virtuelles d'un *escape game* par le biais d'un service de conversations automatisées, il permettra de réduire l'intervention des médiateurs dans les sessions de jeu. Le serveur du chatbot devra tourner dans une application **Django**. Celle-ci devra fournir plusieurs services :

- Un service pour la collection des questions et réponses qui serviront à l'entraînement des modèles de l'apprentissage automatique utilisés par le chatbot. Les questions seront catégorisées et enregistrées dans une base de données. Le service fournira un **CRUD**<sup>2</sup> qui permettra d'ajouter, consulter, supprimer et mettre à jour les données. Ce service permettra également d'ajouter des données en téléchargeant un fichier de données (en format `.csv` par exemple).
- Un service pour l'entraînement du chatbot, celui-ci contiendra tout le flux de l'apprentissage automatique. Il devra permettre de récupérer les données pertinentes de la base de données, prétraiter les questions et les tags et construire et entraîner des modèles. Ce service fournira aussi un système de versionnage des modèles entraînés.
- Un service **API** qui permettra de communiquer avec le chatbot depuis les autres applications via une interface **API Rest**. Ce service devra pouvoir recevoir des demandes de prédictions, vérifier les permissions et renvoyer la réponse du chatbot.

### 4 Étapes de la réalisation

Le projet se compose de plusieurs tâches capitales. Celles-ci sont aménagées pour permettre une mise en production rapide sur un serveur interne afin de tester et contrôler constamment la qualité du chatbot. Les tâches principales du projet sont donc les suivantes :

---

2. **CRUD** : acronyme en anglais (pour create, read, update, delete) qui désigne les quatre opérations de base permettant la gestion d'une collection d'éléments dans une base de données.

- Dans un premier temps, un service est à mettre en place pour faciliter la collecte des données. Celui-ci permettra aux médiateurs de rassembler les questions les plus posées afin de les utiliser pour entraîner le chatbot. Ce service sera contenu dans une application autonome d'un projet **Django**.
- Ensuite, afin de faciliter la tâche d'entraînement, une deuxième application sera introduite (dans le même projet **Django**), celle-ci devra contenir tout le flux du traitement du langage naturel et de l'apprentissage profond. Elle devra permettre d'entraîner plusieurs modèles, de les versionner et de sélectionner le modèle avec le meilleur score pour les prédictions.
- Le projet s'étendra pour permettre de garder en mémoire les conversations anonymes dans le but d'enrichir sa base de connaissances.
- À terme, le but est d'avoir un modèle entraîné pour chaque catégorie de questions, une catégorie correspond à une phase du jeu. Le serveur devra pouvoir détecter la phase du jeu et utiliser le modèle correspondant pour prédire une réponse.
- Le chatbot sera ensuite enrichi avec des fonctionnalités supplémentaires, pour qu'il puisse suivre l'avancement du jeu et prendre l'initiative de donner des conseils ou indices aux utilisateurs.

## 5 Avancement de la réalisation

Durant les premières semaines, une version prototype du chatbot a été réalisée. Cette version comporte un projet **Django** avec ces différentes fonctionnalités :

- Une application qui implémente un **CRUD** permettant de collecter et catégoriser les données d'entraînement dans une base de données **PostgreSQL**. Les données contiennent des questions et réponses catégorisées. Des méthodes ont été implémentées pour charger les données via un formulaire où directement depuis un fichier **.csv**. Il est également possible de modifier ou supprimer des éléments de la base de données via une interface graphique.
- Une application qui recueille les données depuis la base de données, traite les questions et les tags et entraîne un modèle prototype construit avec **TensorFlow**. Cette application fournit une interface permettant à l'utilisateur de configurer certains paramètres d'entraînement et de tester les modèles entraînés via un système de conversation. Les modèles entraînés sont organisés avec un système de versionnage et peuvent être utilisés de manière interchangeable.
- Une version initiale d'une application qui fournit une interface d'**API** pour communiquer avec le chatbot depuis d'autres applications.
- Des tests unitaires pour assurer le bon fonctionnement des différentes applications.
- Des fichiers **Dockerfile** et **docker-compose** qui permettent de définir et exécuter l'application dans des conteneurs **Docker**.

## 6 Conclusion

Mon stage à **Tricky** est une expérience pleine d'apprentissage. Il m'a permis de mettre en pratique les connaissances apprises le long des trois années à l'école afin de mener à bien les différentes tâches qui me sont confiées. Pendant les premières semaines, j'ai pu découvrir les différentes technologies et méthodes utilisées en entreprise afin de faciliter la tâche de développement et la rendre à la fois efficace et rigoureuse. Les sessions de revue de code font partie de ces méthodes, il s'agit de séances hebdomadaires de quelques heures dont le but est de présenter aux autres des méthodes astucieuses utilisées pour résoudre une problématique, ou faire de la programmation en groupe (mob programming). J'ai également eu l'occasion de découvrir les activités de l'entreprise notamment en expérimentant certains services fournis par celle-ci.

# Annexes

## A Outils de gestion de projet

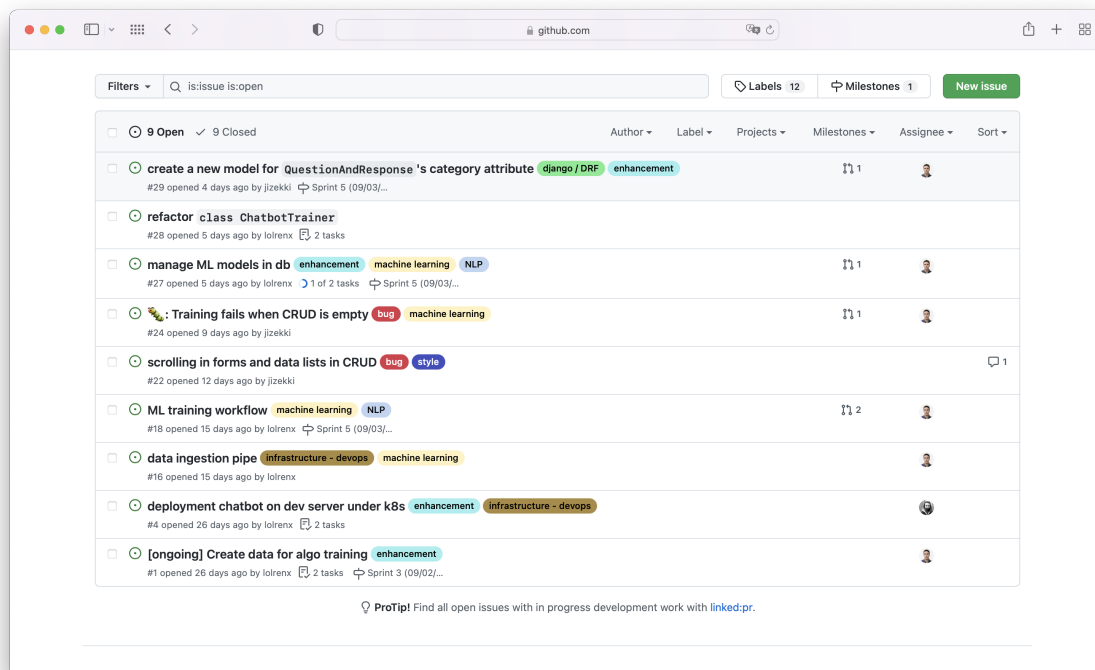


FIGURE 2 – Github Issues (04/03/2022)

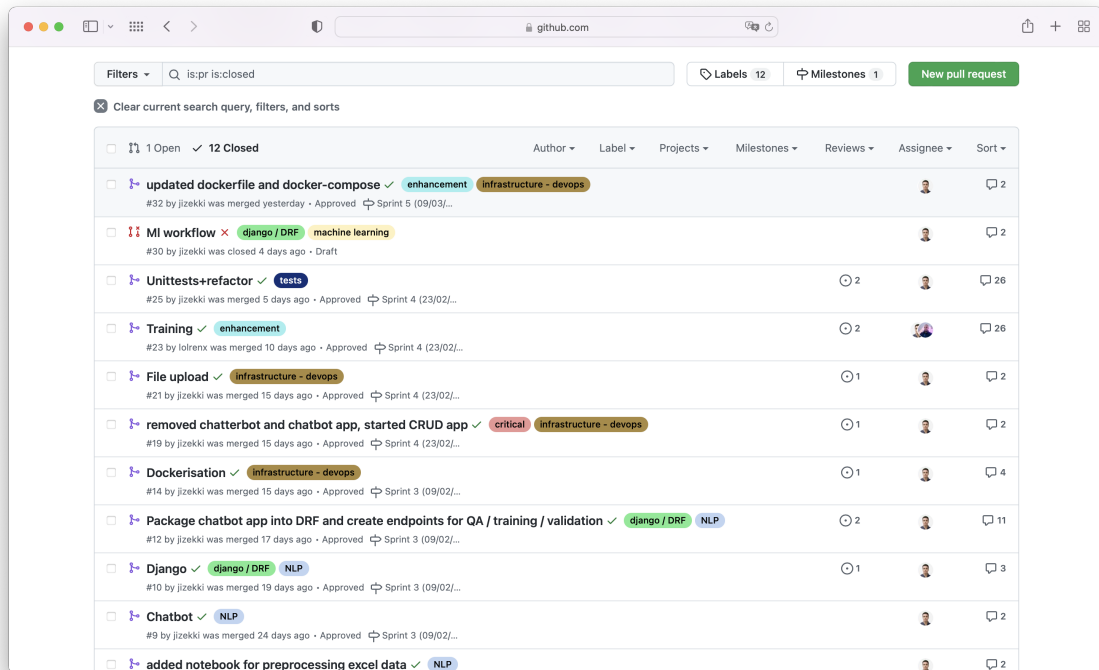


FIGURE 3 – Github Pull requests (04/03/2022)

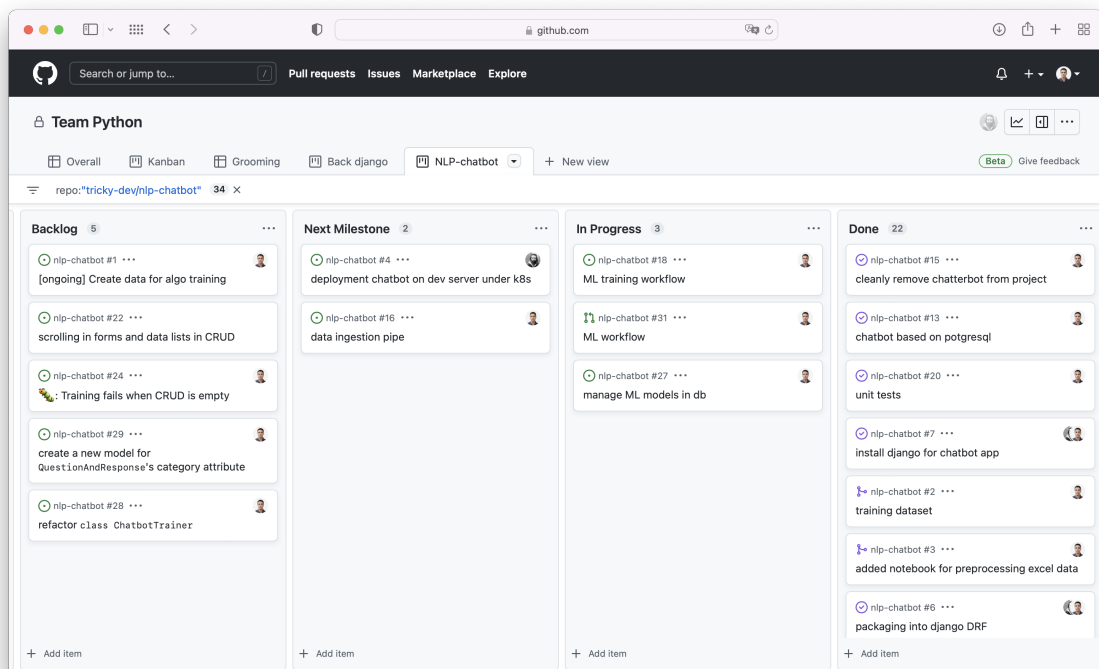


FIGURE 4 – Github Projects (05/03/2022)