

# Projet IFT3150: Chatbot Dans L'outil Planificateur

---

**Thèmes:** NLP, Science de données, Génie logiciel

**Superviseur:** Louis-Edouard LAFONTANT **Project URL:** <https://tibbysshr.github.io/Planificateur/>

## Équipe

- Haoran Sun 20260543

## Description du projet

### Contexte

Dans le contexte universitaire, la planification du parcours académique reste une tâche complexe pour de nombreux étudiants. Ces derniers doivent non seulement consulter les structures de programmes, vérifier les prérequis, suivre leur accumulation de crédits, mais aussi choisir parmi une multitude de cours répartis sur plusieurs trimestres. Lorsque s'ajoutent des intérêts personnels (comme l'intelligence artificielle ou le développement web), la recherche manuelle d'un cheminement optimal devient fastidieuse.

Face à ces enjeux, l'intégration d'un assistant conversationnel intelligent devient pertinente. Un tel chatbot permettrait aux étudiants d'exprimer librement leurs besoins en langage naturel — par exemple : « Je veux me spécialiser en IA » — et de recevoir instantanément des suggestions de cours pertinentes et cohérentes avec leur avancement. L'outil vise ainsi à rendre la planification plus fluide, plus intuitive, et mieux alignée avec les objectifs individuels de chaque étudiant.

### Problématique ou motivations

Dans un contexte où l'offre de cours est vaste, technique et souvent difficile à interpréter, de nombreux étudiants peinent à choisir les cours les plus pertinents pour eux. Il est fréquent qu'ils ne sachent pas quels cours correspondent réellement à leurs intérêts, à leurs forces personnelles ou à leur orientation académique (ex. : intelligence artificielle, systèmes, développement logiciel).

De plus, même lorsqu'ils identifient un domaine cible, il leur est difficile de savoir quels cours sont requis, quels préalables doivent être validés ou dans quel ordre les suivre. Ce manque d'accompagnement intelligent oblige les étudiants à effectuer des vérifications manuelles dispersées, sans assurance de faire les bons choix.

Face à cette problématique, le projet vise à développer un agent conversationnel intelligent capable de comprendre les préférences, les objectifs et les contraintes de l'étudiant, afin de lui proposer des suggestions de cours adaptées, alignées à la fois avec les exigences du programme et ses aspirations personnelles. L'ambition est de rendre la planification académique plus fluide, plus personnalisée et plus stratégique.

### Proposition et objectifs

Ce projet propose le développement d'un agent conversationnel intelligent destiné à accompagner les étudiants dans leurs décisions de parcours académique. Basé sur les avancées récentes en traitement du langage naturel et les modèles de langage à grande échelle (LLM), ce chatbot vise à comprendre les

intentions, les préférences et les contraintes exprimées en langage courant, afin de proposer des recommandations de cours ciblées et pertinentes.

Contrairement aux systèmes de recherche traditionnels, ce chatbot permet un dialogue fluide et contextuel.

Les objectifs principaux de cette proposition sont :

- Comprendre les préférences académiques (domaine d'intérêt, les matières maîtrisées, rythme de travail, etc.) exprimées en langage naturel.
- Analyser les cours déjà suivis pour déterminer les préalables déjà validés et les crédits restants.
- Proposer des combinaisons de cours compatibles avec les contraintes exprimées : conflits d'horaire, équilibre de la charge de travail, programmes doubles, etc.
- Fournir des suggestions interactives (avec possibilité de relancer, affiner ou modifier la demande).
- Accompagner la planification sur plusieurs trimestres, en construisant un cheminement cohérent selon les exigences du programme.

## Méthodologie

- Architecture technique Le module chatbot est intégré dans une architecture modulaire en deux parties :

Front-end : un composant ChatBot.js est ajouté dans l'application React, chargé de gérer l'affichage du dialogue, la saisie utilisateur, et l'envoi des messages.

Back-end : un nouveau point d'entrée `/api/chat.py` est créé dans FastAPI pour traiter chaque message, avec transmission d'un identifiant de session.

Le dialogue suit un modèle requête-réponse asynchrone, permettant une communication en temps réel avec l'interface, sans rechargement.

- Traitement du langage naturel (NLP)

Les messages envoyés par l'étudiant sont analysés avec spaCy, une bibliothèque NLP qui permet :

- a. d'extraire les intentions principales (domaine d'intérêt, contraintes horaires, objectifs)
- b. d'identifier des entités spécifiques, comme des codes de cours, des trimestres ou des préférences lexicales
- c. Une couche de règles personnalisées est ajoutée pour mapper ces intentions vers des éléments du programme académique.

- Gestion de session et de contexte

Pour permettre des interactions fluides et multi-tours :

- a. chaque session est identifiée par un `session_id` côté client
- b. le back-end conserve en mémoire (ou optionnellement en base MongoDB) : les intentions déjà détectées; les réponses antérieures; l'état courant du dialogue

Cela permet de gérer l'historique et d'adapter dynamiquement la réponse à l'étudiant, tout au long de l'échange.

- Logique d'interaction adaptative

Le chatbot est conçu pour être proactif et guidant.

a. Si la demande est claire, il propose directement des recommandations.

b. Si elle est floue ou ambiguë, le chatbot pose des questions ciblées pour clarifier : "Souhaites-tu un cours plus théorique ou pratique ?", "Dans quel domaine souhaites-tu te spécialiser ?"

Ce mécanisme repose sur un état conversationnel, implémenté via une machine à états ou une logique conditionnelle souple.

- Génération des recommandations

Une fois les intentions suffisamment claires :

a. le système appelle un moteur de règles dans /utils/recommendation.py b. ce moteur sélectionne les cours compatibles avec : les intérêts exprimés; les préalables validés; les contraintes temporelles; les crédits restants

Plusieurs combinaisons possibles sont générées, classées par pertinence.

- Intégration avec le calendrier

Les cours recommandés sont ensuite formatés pour pouvoir :

a. être affichés immédiatement dans l'interface conversationnelle

b. être ajoutés automatiquement dans le calendrier de planification si l'étudiant accepte la combinaison proposée

c. ou être modifiés via une interface glisser-déposer