

Application Web pour la gestion du service recherche, Coopération,  
partenariat et Stages

**Réalisé par :**  
BADIAA ESSALIHI  
**Supervisé par :**  
Lotfi EL AACHAK

Juillet 2024

# Table des matières

<b>1</b>	<b>Conception</b>	<b>1</b>
1.1	Analyse des besoins . . . . .	1
1.1.1	Introduction . . . . .	1
1.1.2	Description des Espaces Utilisateurs . . . . .	1
1.2	Conception de la Base de Données . . . . .	3
1.2.1	Table . . . . .	3
1.2.2	Relations Clés . . . . .	5
1.2.3	Modèle proposé . . . . .	5
1.3	Les maquettes . . . . .	6
<b>2</b>	<b>Logiciels et Outils</b>	<b>9</b>
2.1	Figma . . . . .	9
2.2	Visual Studio Code . . . . .	9
2.3	GitHub . . . . .	9
2.4	StarUML . . . . .	10
2.5	XAMPP . . . . .	10
2.6	Postman . . . . .	10
<b>3</b>	<b>Technologies Utilisées</b>	<b>11</b>
3.1	Front-end . . . . .	11
3.2	Back-end . . . . .	14

## Table des figures

1	Modèle proposé . . . . .	5
2	Page d'accueil . . . . .	6
3	Choix de profil . . . . .	7
4	Page de connexion . . . . .	7
5	Page d'inscription . . . . .	8
6	Page de réinitialisation de mot de passe . . . . .	8
7	Le logo de Figma . . . . .	9
8	Le logo de Visual Studio Code . . . . .	9
9	Le logo de Figma . . . . .	10
10	Le logo de StarUML . . . . .	10
11	Le logo de XAMPP . . . . .	10
12	Le logo de Postman . . . . .	11
13	Logo de React.js . . . . .	11
14	Logo de Redux . . . . .	11
15	Logo de Material-UI . . . . .	11
16	Logo de Chakra UI . . . . .	12
17	Logo de Bootstrap . . . . .	12
18	Logo de Tailwind CSS . . . . .	12
19	Logo d'Axios . . . . .	12
20	Logo de React Router . . . . .	13
21	Logo de Formik . . . . .	13
22	Logo de Chart.js . . . . .	13
23	EXemple de React Big Calendar . . . . .	14
24	Logo de Node.js . . . . .	14
25	Logo d'Express.js . . . . .	14
26	Logo de Sequelize . . . . .	14
27	Logo de Nodemailer . . . . .	15
28	Le logo de phpMyAdmin . . . . .	15

# 1 Conception

## 1.1 Analyse des besoins

### 1.1.1 Introduction

#### 1.1.1.1 Contexte

L'administration souhaite développer une application web pour optimiser la gestion et la communication entre les doctorants, les enseignants et les administrateurs. L'application proposée offrira trois espaces distincts, chacun étant conçu pour répondre aux besoins spécifiques des différents utilisateurs de la plateforme.

#### 1.1.1.2 Objectifs

- **Facilitation de la communication** : Améliorer le partage d'informations entre doctorants et enseignants pour favoriser la collaboration et le suivi académique.
- **Gestion efficace des activités académiques** : Offrir des outils pour la gestion des projets de recherche, des thèses et des publications, permettant ainsi un suivi rigoureux et structuré.
- **Gestion efficace des activités académiques** : Offrir des outils pour la gestion des projets de recherche, des thèses et des publications, permettant ainsi un suivi rigoureux et structuré.

### 1.1.2 Description des Espaces Utilisateurs

#### 1.1.2.1 Espace Doctorant

##### 1.1.2.1.1 Fonctionnalités

- **Inscription et Authentification** :
  - **Inscription avec validation par email** : Les doctorants peuvent s'inscrire en utilisant leur adresse email, suivie d'une validation pour assurer la véracité des informations fournies.
  - **Authentification via email et mot de passe** : Accès sécurisé à l'application.
  - **Récupération de mot de passe** : Option pour récupérer ou réinitialiser le mot de passe en cas de perte.
- **Tableau de Bord** :
  - **Édition de documents** : Les doctorants peuvent éditer divers documents administratifs, tels que des demandes de changement de sujet de thèse, des retraits provisoires ou définitifs de diplômes, des demandes d'attestation d'inscription, des conventions de stage, et des cartes d'étudiant.
  - **Gestion des informations personnelles** : Interface permettant de gérer les informations personnelles, les données du baccalauréat, le cursus universitaire, et les inscriptions au cycle doctoral.
  - **Calendrier** : Affichage des formations et des événements académiques.
  - **Pré-soutenance et post-soutenance** : Édition du dossier administratif de pré-soutenance et des documents nécessaires après la soutenance.
  - **Demandes spécifiques** : Soumission de demandes d'inscription et de dérogations, et upload de demandes signées.

##### 1.1.2.1.2 Interface Utilisateur

- **Interface intuitive et conviviale** : Design ergonomique facilitant l'utilisation par tous les doctorants.
- **Design responsive** : Accessibilité optimisée pour une utilisation sur mobile et tablette, assurant une flexibilité d'accès en tout lieu.

#### 1.1.2.2 Espace Enseignant

#### 1.1.2.2.1 Fonctionnalités

- **Inscription et Authentification :**
  - **Inscription avec validation par email :** Inscription basée sur une liste pré-établie pour garantir que seuls les enseignants autorisés peuvent accéder à l'application.
  - **Authentification via email et mot de passe :** Sécurisation de l'accès.
- **Tableau de Bord :**
  - **Vue d'ensemble des doctorants encadrés :** Suivi des doctorants sous la supervision de l'enseignant.
  - **Calendrier des soutenances et réunions :** Planification et gestion des soutenances et des réunions académiques.
  - **Diffusion d'informations :** Partage des informations sur les événements, les formations, et les appels à projet.
- **Gestion des Projets et Thèses :**
  - **Suivi de l'avancement :** Outil de suivi de l'état d'avancement des projets de recherche et des thèses.
  - **Commentaires et feedbacks :** Possibilité de commenter et de fournir des retours sur les documents soumis par les doctorants.

#### 1.1.2.2.2 Interface Utilisateur

- **Interface intuitive et conviviale :** Simplifie l'interaction avec la plateforme.
- **Design responsive :** Adaptabilité à différents dispositifs pour une utilisation flexible.

#### 1.1.2.3 Espace Administrateur

##### 1.1.2.3.1 Fonctionnalités

- **Inscription et Authentification :**
  - **Authentification via email et mot de passe :** Assure la sécurité des accès administratifs.
- **Tableau de Bord :**
  - **Vue d'ensemble des activités :** Aperçu complet des activités des doctorants et des enseignants.
  - **Notifications et alertes :** Gestion des notifications importantes et des alertes pour un suivi proactif.
  - **Paramétrage de l'application :** Introduction des données nécessaires pour configurer et personnaliser l'application selon les besoins institutionnels.
  - **Validation des demandes :** Processus de validation pour les demandes soumises par les enseignants et les doctorants.
- **Gestion des Utilisateurs :**
  - **Création, modification et suppression de comptes :** Gestion complète des comptes utilisateurs.
  - **Gestion des droits et permissions :** Attribution et gestion des rôles et permissions pour assurer un accès contrôlé et sécurisé.
- **Supervision des Projets et Thèses :**
  - **Suivi de l'avancement :** Surveillance de l'évolution des projets et des thèses.
  - **Gestion des soutenances et des réunions :** Organisation et gestion des sessions de soutenance et des réunions académiques.
- **Communication :**
  - **Envoi de notifications et d'annonces :** Facilitation de la communication institutionnelle par des notifications et des annonces.

##### 1.1.2.3.2 Interface Utilisateur

- **Interface intuitive et conviviale :** Conçue pour une utilisation facile et efficace par les administrateurs.
- **Design responsive :** Accessibilité sur divers dispositifs pour une gestion flexible.
- **Tableau de bord personnalisé :** Personnalisation des informations affichées selon les besoins de l'utilisateur administratif.

## 1.2 Conception de la Base de Données

### 1.2.1 Table

#### 1.2.1.1 Table : Utilisateurs

La table Utilisateurs contient les informations de base sur tous les utilisateurs de l'application.

- **id\_utilisateur (PK)** : Identifiant unique pour chaque utilisateur.
- **email (Unique)** : Adresse email de l'utilisateur, utilisée pour la connexion.
- **mot\_de\_passe\_hache** : Mot de passe hashé pour des raisons de sécurité.
- **role** : Rôle de l'utilisateur dans l'application (Doctorant, Enseignant, Administrateur).
- **prenom** : Prénom de l'utilisateur.
- **nom** : nom de l'utilisateur.
- **tele** : Numéro de téléphone de l'utilisateur.

#### 1.2.1.2 Table : Doctorants

- **id\_doctorant (PK)** : Identifiant unique pour chaque doctorant.
- **id\_utilisateur (FK - Utilisateurs)** : Référence à l'utilisateur associé.

#### 1.2.1.3 Table : Infos Personnelles

La table infos\_personnelles contient des informations personnelles détaillées des doctorants.

- **id\_doctorant (FK - Doctorants)** : Référence au doctorant associé.
- **civilite** : Civilité (Monsieur, Madame, etc.).
- **sexe** : Sexe du doctorant.
- **nom** : Nom du doctorant.
- **prenom** : Prénom du doctorant.
- **date\_naissance** : Date de naissance du doctorant.
- **email** : Adresse email du doctorant.
- **telephone** : Numéro de téléphone du doctorant.
- **lieu\_naissance** : Lieu de naissance du doctorant.
- **cnie** : Numéro de la carte nationale d'identité.
- **situation\_socioprofessionnelle** : Situation socioprofessionnelle actuelle.

#### 1.2.1.4 Table : Infos Baccalauréat

La table infos\_baccalaureat contient des informations sur le baccalauréat des doctorants.

- **id\_doctorant (FK - Doctorants)** : Référence au doctorant associé.
- **annee\_bac** : Année d'obtention du baccalauréat.
- **type\_bac** : Type de baccalauréat (Scientifique, Littéraire, etc.).
- **mention** : Mention obtenue au baccalauréat.
- **cne\_massar** : Numéro national de l'étudiant (CNE) ou code Massar.
- **serie\_bac** : Série du baccalauréat.
- **academie** : Académie où le baccalauréat a été obtenu.
- **province** : Province où le baccalauréat a été obtenu.

#### 1.2.1.5 Table : Infos Universitaires

La table infos\_universitaires contient des informations sur les études universitaires des doctorants.

- **id\_doctorant (FK - Doctorants)** : Référence au doctorant associé.
- **diplome** : Diplôme obtenu.
- **etablissement** : Établissement où le diplôme a été obtenu.
- **universite** : Université où le diplôme a été obtenu.

#### 1.2.1.6 Table : Infos Cycle Doctoral

La table `infos_cycle_doctoral` contient des informations sur le cycle doctoral des doctorants.

- **id\_doctorant (FK - Doctorants)** : Référence au doctorant associé.
- **formation\_doctorale** : Formation doctorale suivie.
- **annee\_soutenance\_prevue** : Année prévue pour la soutenance de la thèse.
- **specialite** : Spécialité de recherche.
- **directeur\_these** : Directeur de thèse.
- **structure\_recherche\_directeur** : Structure de recherche du directeur de thèse.
- **co\_directeur\_these** : Co-directeur de thèse (le cas échéant).
- **structure\_recherche\_co\_directeur** : Structure de recherche du co-directeur de thèse.
- **universite\_cotutelle** : Université de cotutelle (le cas échéant).
- **sujet\_recherche** : Sujet de recherche de la thèse.

#### 1.2.1.7 Table : Enseignants

La table `Enseignants` contient des informations spécifiques aux enseignants.

- **id\_enseignant (PK)** : Identifiant unique pour chaque enseignant.
- **id\_utilisateur (FK - Utilisateurs)** : Référence à l'utilisateur associé.
- **liste\_pre\_etablie** : Liste des doctorants encadrés.

#### 1.2.1.8 Table : Administrateurs

- **id\_administrateur (PK)** : Identifiant unique pour chaque administrateur.
- **id\_utilisateur (FK - Utilisateurs)** : Référence à l'utilisateur associé.

#### 1.2.1.9 Table : Projets et Thèses

- **id\_projet (PK)** : Identifiant unique pour chaque projet.
- **titre** : Titre du projet.
- **description** : Description détaillée du projet.
- **statut** : Statut du projet (en cours, terminé, etc.).
- **date\_debut** : Date de début du projet.
- **date\_fin** : Date de fin prévue du projet.
- **id\_doctorant (FK - Doctorants)** : Référence au doctorant associé.
- **id\_enseignant (FK - Enseignants)** : Référence à l'enseignant encadrant.

#### 1.2.1.10 Table : Demandes

- **id\_demande (PK)** : Identifiant unique pour chaque demande.
- **type\_demande** : Type de demande (administrative, recherche, autre).
- **statut** : Statut de la demande (soumise, en cours, validée, rejetée).
- **date\_soumission** : Date de soumission de la demande.
- **id\_doctorant (FK - Doctorants)** : Référence au doctorant ayant soumis la demande.
- **id\_enseignant (FK - Enseignants, nullable)** : Référence à l'enseignant validant (le cas échéant).
- **details** : Détails supplémentaires sur la demande.

#### 1.2.1.11 Table : Événements

- **id\_evenement (PK)** : Identifiant unique pour chaque événement.
- **titre** : Titre de l'événement.
- **description** : Description détaillée de l'événement.
- **date\_evenement** : Date de l'événement.
- **lieu** : Lieu de l'événement.
- **type** : Type d'événement (conférence, atelier, etc.).

#### 1.2.1.12 Table : Notifications

- **id\_notification (PK)** : Identifiant unique pour chaque notification.
- **message** : Contenu du message de la notification.
- **date** : Date d'envoi de la notification.
- **id\_utilisateur (FK - Utilisateurs)** : Référence à l'utilisateur recevant la notification.

#### 1.2.2 Relations Clés

Les relations entre les différentes tables assurent la cohérence et l'intégrité des données :

- Chaque utilisateur (Utilisateurs) peut être associé à un doctorant (Doctorants), un enseignant (Enseignants) ou un administrateur (Administrateurs) via id\_utilisateur.
- Les projets et thèses (Projets) sont liés à un doctorant (Doctorants) et un enseignant (Enseignants).
- Les demandes (Demandes) sont faites par les doctorants et validées par les enseignants ou les administrateurs.
- Les événements (Événements) sont organisés par les utilisateurs et peuvent être ajoutés au calendrier personnel des utilisateurs.
- Les notifications (Notifications) sont envoyées aux utilisateurs pour les informer des mises à jour importantes.

#### 1.2.3 Modèle proposé

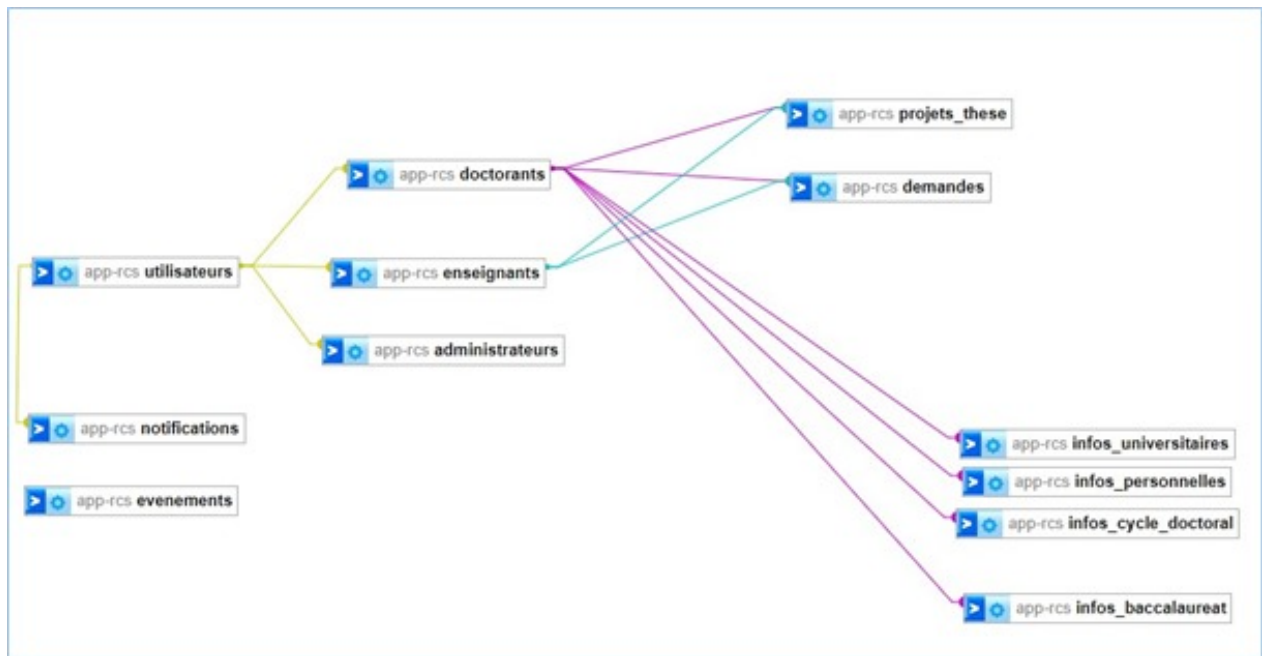


FIGURE 1 – Modèle proposé



### 1.3 Les maquettes



FIGURE 2 – Page d'accueil

## Votre profil ?



FIGURE 3 – Choix de profil

## Connexion

EMAIL

nomprenom@exemple.com

MOT DE PASSE

\*\*\*\*\*

**se connecter**

[Mot de passe oublié ?](#)

FIGURE 4 – Page de connexion

The registration form is titled "Inscription" and is set against a light blue background. It includes fields for "PRENOM" (First Name) with the value "Badiaa", "NOM" (Last Name) with the value "ESSALIHI", "EMAIL" with the value "nomprenom@exemple.com", and "MOT DE PASSE" (Password) with masked characters "\*\*\*\*\*". A green "S'inscrire" button is at the bottom.

كلية العلوم  
فakademy  
FACULTÉ DES SCIENCES

جامعة سعيدة الدكالي

## Inscription

PRENOM

Badiaa

NOM

ESSALIHI

EMAIL

nomprenom@exemple.com

MOT DE PASSE

\*\*\*\*\*

S'inscrire

FIGURE 5 – Page d'inscription

The password reset form is titled "Réinitialiser votre mot de passe" and is set against a light blue background. It includes a field for "VOTRE EMAIL ?" with the value "nomprenom@exemple.com". A green "Récupérer le !" button is at the bottom.

كلية العلوم  
فakademy  
FACULTÉ DES SCIENCES

جامعة سعيدة الدكالي

## Réinitialiser votre mot de passe

VOTRE EMAIL ?

nomprenom@exemple.com

Récupérer le !

FIGURE 6 – Page de réinitialisation de mot de passe

## 2 Logiciels et Outils

### 2.1 Figma

Figma est une plateforme de conception d'interfaces utilisateur (UI) et d'expérience utilisateur (UX) basée sur le cloud. C'est un outil largement adopté par les concepteurs, développeurs et équipes de design pour créer, collaborer et prototyper des interfaces utilisateur interactives pour des applications, sites web, et autres produits numériques. Grâce à son interface intuitive et conviviale, Figma permet aux utilisateurs de créer facilement des maquettes, wireframes, et prototypes interactifs. L'un des grands avantages de Figma est qu'il est entièrement basé sur le cloud, ce qui facilite la collaboration en temps réel entre les équipes, peu importe leur localisation. Cette fonctionnalité est idéale pour la gestion de projets à distance et la coordination entre différents membres d'une équipe. De plus, la gestion des versions est simplifiée, permettant à chaque contributeur de suivre les modifications et d'apporter des ajustements sans complexité.

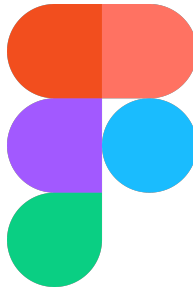


FIGURE 7 – Le logo de Figma

### 2.2 Visual Studio Code

Visual Studio Code (VSCode) est un éditeur de code source largement utilisé développé par Microsoft. Il est connu pour sa légèreté, ses options de personnalisation étendues et son support pour divers langages de programmation et technologies. VSCode propose des fonctionnalités telles que la mise en évidence du code, l'autocomplétion, les capacités de débogage, l'intégration de contrôle de version et un écosystème d'extensions qui permettent aux développeurs d'améliorer leur fonctionnalité en fonction de leurs besoins spécifiques.

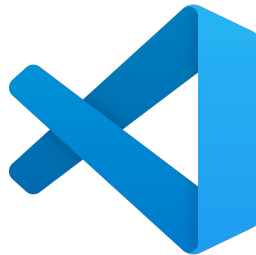


FIGURE 8 – Le logo de Visual Studio Code

### 2.3 GitHub

GitHub est une plateforme de développement collaboratif basée sur le système de contrôle de version Git. Elle permet aux développeurs de travailler ensemble sur des projets, de partager du code source, de suivre les modifications, de gérer les problèmes et les demandes de fusion, et de collaborer de manière transparente sur des projets logiciels. En utilisant GitHub, vous pouvez héberger vos dépôts de code source, collaborer avec d'autres développeurs en facilitant le travail en équipe, examiner les modifications apportées au code, signaler et résoudre des problèmes, et gérer le cycle de vie de développement d'un projet.

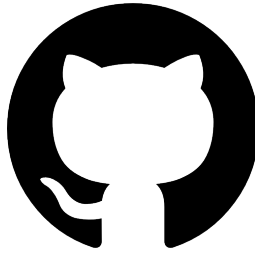


FIGURE 9 – Le logo de Figma

## 2.4 StarUML

StarUML est un logiciel de modélisation UML (Unified Modeling Language) largement utilisé pour concevoir et créer des diagrammes de modèles logiciels. Il permet aux développeurs et aux ingénieurs de créer des représentations visuelles des structures, des interactions et comportements des systèmes logiciels.



FIGURE 10 – Le logo de StarUML

## 2.5 XAMPP

XAMPP est une solution de serveur web open-source qui inclut Apache, MySQL (ou MariaDB), et PHP. Il permet de créer un environnement de développement local complet pour les projets web. XAMPP facilite la mise en place d'un serveur local pour le développement et le test des applications web avant leur déploiement sur un serveur de production.



FIGURE 11 – Le logo de XAMPP

## 2.6 Postman

Postman est un outil de développement API utilisé pour tester et déboguer les requêtes API. Il permet aux développeurs de créer et d'exécuter des requêtes HTTP, de vérifier les réponses des API, et de gérer les collections de requêtes pour le développement et le test des APIs. Postman est particulièrement utile pour tester les endpoints API et s'assurer que les services fonctionnent comme prévu avant l'intégration avec l'application frontend.



FIGURE 12 – Le logo de Postman

### 3 Technologies Utilisées

#### 3.1 Front-end

- **React.js** : Une bibliothèque JavaScript développée par Facebook pour construire des interfaces utilisateur dynamiques et interactives. Elle permet de créer des composants réutilisables et de gérer l'état de l'application efficacement. React.js est souvent utilisée pour le développement d'interfaces utilisateur modernes et performantes.



FIGURE 13 – Logo de React.js

- **Redux** : Une bibliothèque JavaScript utilisée pour la gestion d'état global dans les applications. Redux centralise l'état dans un store unique, facilitant ainsi la gestion et la synchronisation des données entre les composants.



FIGURE 14 – Logo de Redux

- **Material-UI** : Framework de composants React qui implémente les principes de Material Design. Il fournit une collection de composants stylisés pour la création d'interfaces élégantes et cohérentes.

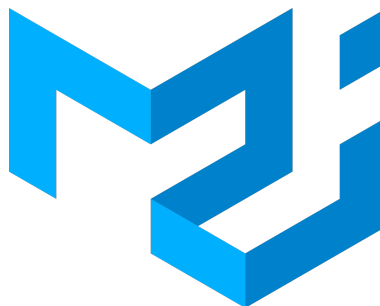


FIGURE 15 – Logo de Material-UI

- **Chakra UI** : Une bibliothèque de composants React accessible et personnalisable, axée sur la simplicité et la rapidité pour créer des interfaces utilisateur modernes.

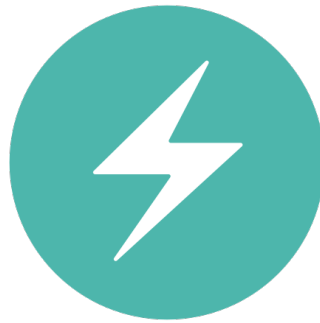


FIGURE 16 – Logo de Chakra UI

- **Bootstrap** : Framework CSS populaire pour créer des sites web réactifs et des interfaces élégantes grâce à des composants et des styles préconçus.



FIGURE 17 – Logo de Bootstrap

- **Tailwind CSS** : Un framework CSS utilitaire permettant une grande flexibilité et une personnalisation rapide des styles, facilitant une approche modulaire et maintenable pour le design web.

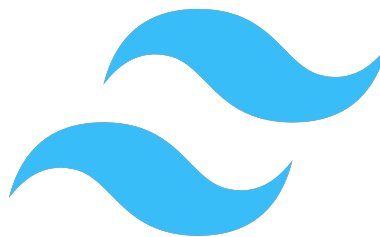


FIGURE 18 – Logo de Tailwind CSS

- **Axios** : Client HTTP basé sur les promesses, utilisé pour effectuer des requêtes API entre le frontend et le backend, permettant des communications réseau efficaces.



FIGURE 19 – Logo d'Axios

- **React Router** : Librairie de gestion de navigation pour React, permettant de gérer les différentes vues de l'application en fonction des URL de manière dynamique.

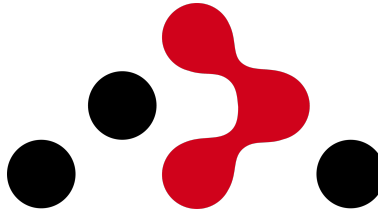


FIGURE 20 – Logo de React Router

- **Formik** : Librairie pour la gestion des formulaires dans React, simplifiant la gestion des états, des validations et des soumissions.



FIGURE 21 – Logo de Formik

- **Chart.js** : Librairie pour la création de graphiques interactifs, permettant d'afficher des données sous forme de graphiques à barres, lignes, secteurs, etc.



FIGURE 22 – Logo de Chart.js

- **React Big Calendar** : Composant de calendrier pour React, permettant de gérer des événements avec des vues mensuelles, hebdomadaires et journalières.



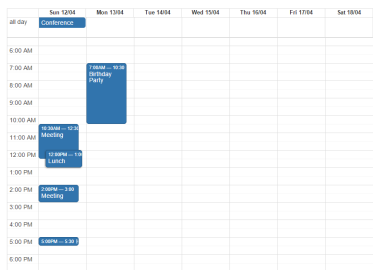


FIGURE 23 – EXemple de React Big Calendar

## 3.2 Back-end

- **Node.js** : Environnement d'exécution JavaScript côté serveur, permettant de construire des applications backend performantes, notamment pour gérer des I/O intensives et des opérations asynchrones.



FIGURE 24 – Logo de Node.js

- **Express.js** : Framework minimaliste pour Node.js, utilisé pour créer des APIs RESTful et gérer les routes dans les applications backend.



FIGURE 25 – Logo d'Express.js

- **Sequelize** : ORM pour Node.js qui permet d'interagir avec des bases de données relationnelles de manière abstraite, facilitant la gestion des modèles et des relations entre les données.



FIGURE 26 – Logo de Sequelize

- **Nodemailer** : Librairie pour l'envoi d'e-mails depuis des applications Node.js via SMTP, souvent utilisée pour l'envoi de notifications et de communications automatisées.



FIGURE 27 – Logo de Nodemailer

- **phpMyAdmin** : phpMyAdmin est un outil libre et open-source écrit en PHP pour gérer MySQL et MariaDB via une interface web. Il permet aux utilisateurs d'administrer facilement les bases de données, d'exécuter des requêtes SQL, de gérer les utilisateurs, les privilèges, les structures de tables, et d'importer ou exporter des données.



FIGURE 28 – Le logo de phpMyAdmin

## Conclusion

In the final stage, both models are integrated into a single interface. This provides users with the flexibility to choose between the two models based on their preferences or needs. Such an interface enhances the user experience by offering a tailored interaction, whether they require detailed information retrieval or more nuanced conversational engagement. The end result is a sophisticated chatbot capable of handling a variety of tasks, from answering straightforward questions to providing in-depth explanations, all while maintaining a user-friendly approach.

## Références

- [1] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, “LoRA : Low-rank adaptation of large language models,” in *International Conference on Learning Representations*, 2022. [Online]. Available : <https://openreview.net/forum?id=nZeVKeeFYf9>
- [2] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, “Automatic differentiation in pytorch,” in *NIPS-W*, 2017.
- [3] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. tau Yih, T. Rocktäschel, S. Riedel, and D. Kiela, “Retrieval-augmented generation for knowledge-intensive nlp tasks,” 2021.
- [4] B. Jacob, S. Kligys, B. Chen, M. Zhu, M. Tang, A. Howard, H. Adam, and D. Kalenichenko, “Quantization and training of neural networks for efficient integer-arithmetic-only inference,” 2017.
- [5] P. QoChuk, Benjamin, “Basics of quantization in machine learning (ml) for beginners.” [Online]. Available : <https://iq.opengenus.org/basics-of-quantization-in-ml/>
- [6] M. A. Fakhre-Eddine, “al0new0lf/unsloth-llama-3-q4\_k.m · hugging face,” 2024. [Online]. Available : [https://huggingface.co/aL0NEW0LF/unsloth-llama-3-q4\\_k.m](https://huggingface.co/aL0NEW0LF/unsloth-llama-3-q4_k.m)
- [7] —, “al0new0lf/unsloth-llama-3-lora · hugging face,” 2024. [Online]. Available : <https://huggingface.co/aL0NEW0LF/unsloth-llama-3-lora>
- [8] Ggerganov, “ggml/docs/gguf.md at master · ggerganov/ggml.” [Online]. Available : <https://github.com/ggerganov/ggml/blob/master/docs/gguf.md>