**Rapport du projet de fin d’année**

**BAC+3 (Filière : Ingénierie Logicielle)**

**Sous le thème :**

**Conception et réalisation d’une application mobile : Santé mental et bien être**

**Réalisé par :**

**TAGMOUTI Ghita**

**CHAOUI Marwa**

**Encadré par :**

**Mr. IRAQI HOUSSAINI Mehdi**

**Mr. MEKOUAR Othman**

**Jury :**

Année Universitaire 2023-2024

***Dédicaces***

***Remerciements***

Chapitre 1 :

Contexte général du projet

1. **Introduction :**

Ce chapitre d'introduction établit les fondements du projet en offrant une vision claire du contexte pour lequel on a choisie notre sujet, des objectifs qui nous tiennent a cœur et des fonctionnalités à réaliser, créant ainsi une base solide pour les étapes ultérieures du développement de l’application mobile « Mind Guard ».

1. **Contexte du projet :**

Durant ces dernières années, la santé mentale est un sujet qui intéresse de plus en plus les marocains et particulièrement les jeunes marocains. Ces derniers souhaitent prendre soin de leur santé non seulement physique mais aussi mentale. En effet, les jeunes marocains sont aujourd’hui de plus en plus à l’aise avec les nouvelles technologies, d’où l’idée de l'application mobile dédiée au bien-être mental.

Cette application se présente comme une solution pour s’intéresser à leur santé et suivre leur évolution . Il vise principalement à offrir aux utilisateurs une variété d’outils et de ressources afin de favoriser l’amélioration de leur bien-être psychologique. Priorisant la gestion du stress et des émotions, elle cherche à fournir un accompagnement personnalisé, tout en facilitant l'accès à un soutien psychologique professionnel lorsque cela s'avère nécessaire.

1. **Objectif :**

Dans le cadre de ce projet, on vise a développé une application mobile accessible à tous. Notre objectif principal pour lequel on a eu cette idée est de :

* Fournir des informations accessible et assuré sur les techniques de gestion de stress.
* Mettre à disposition une bibliothèque de ressources pour approfondir la compréhension des utilisateurs sur les questions de santé mentale.
* Faciliter l’accès aux marocains à une certaine thérapie virtuelle et a des recommandations.
* Concevoir une interface utilisateur intuitive et conviviale, adapté aux besoins des utilisateurs.

1. **Fonctionnalités principales :**

Cette application vise à fournir aux utilisateurs une expérience enrichissante en leur offrant diverses fonctionnalités :

* Suivi de l’humeur : Les utilisateurs pourront enregistrer quotidiennement leur état émotionnel et visualisé des graphes de leurs progression, ce qui leur permettra de détecter des tendances dans leurs émotions sur une période donnée.
* Suivi du Sommeil : Les utilisateurs pourront enregistrer et surveiller la qualité de leur sommeil à l'aide d'outils fournis par l'application, leur permettant ainsi de mieux comprendre l'impact de leur sommeil sur leur santé mentale. En enregistrant les heures de sommeil, évaluant la qualité du repos et identifiant les habitudes de sommeil.
* Ressources Éducatives :L'application proposera aux utilisateurs plusieurs contenus éducatifs pour favoriser la détente, réduire le stress et améliorer la concentration tels que des articles, des vidéos et des podcasts centrés sur la santé mentale.
* Méditation guidée : Elle permet aux utilisateurs de se concentrer sur leur esprit à l'aide d'un guide vocal et des exercices de relaxation, les aidant ainsi à calmer leur esprit et à se libérer des pensées négatives.
* Exercices de relaxation : De même, la relaxation musculaire progressive, permet de relâcher les tensions corporelles et mentales comme le yoga , favorisant ainsi un état de calme et de sérénité.
* Exercices de respiration : Cette fonctionnalité offre des techniques simples mais puissantes pour réguler les émotions et calmer le système nerveux. En pratiquant la respiration profonde et contrôlée.
* Listes de Tâches (To-Do List) : L’application permettra aux utilisateurs de créer et gérer des listes de tâches pour organiser leurs activités quotidiennes pour réaliser leur objectifs personnels et professionnels . Favorisant ainsi un meilleur équilibre entre leurs responsabilités et leur bien-être mental.
* Rappels Quotidiens : L'application offrira une variété de rappels pour aider les utilisateurs à recevoir des notifications pour différentes activités, telles que rester hydraté, méditer, prendre des pauses, reposer les yeux des écrans, exprimer la gratitude, consulter les tâches à faire, faire des exercices de respiration, etc.

1. **Livrables attendus :**

Tout d’abord, on devra fournir un cahier de charges. Ce document servira de références pour guider le développement de notre application et assurer la conformité aux attentes du projet .

Ensuite, nous fournirons un rapport qui comprendra cinq chapitres principaux :

* Chapitre 1 : Contexte générale du projet
* Chapitre 2 : Conception et modélisation
* Chapitre 3 : Choix technologique
* Chapitre 4 : Mise en œuvre du projet
* Chapitre 5 : Conclusion et perspectives

Également, nous livrerons une application mobile fonctionnelle conforme au cahier de charge, cette application permettra aux utilisateurs de suivre leur santé mentale.

Enfin, on produira une présentation PowerPoint de notre projet et du processus de développement qui mettra en évidence les étapes et les choix qu’on a effectué durant les deux mois de travail et ainsi présenter le résultat final .

En combinant ces éléments , on aura répondu aux besoins de ce projet qui sera compléter le 30 Mai 2024.

1. **Calendrier de développement :**

•**Semaine 1-2** *:* Conception des maquettes d’authentification et élaboration du cahier des charges.

• **Semaine 3-4** *:* Développement du frontend de l’application mobile.

• **Semaine 5-6** *:* Mise en place du backend et développement des fonctionnalités dynamiques.

**• Semaine 7-8** *:* Tests, débogage et optimisation de l’application.

• **Semaine 9** *:* Finalisation de la documentation et préparation de la présentation finale.

1. **Conclusion :**

Tout d’abord, à travers ce chapitre, on a introduit le contexte général du projet, ses objectifs. On a également défini les fonctionnalités principales de notre application en tenant compte du contexte général. Le chapitre suivant se penchera plus en détail sur la conception et la modélisation du projet.

Chapitre 2 :

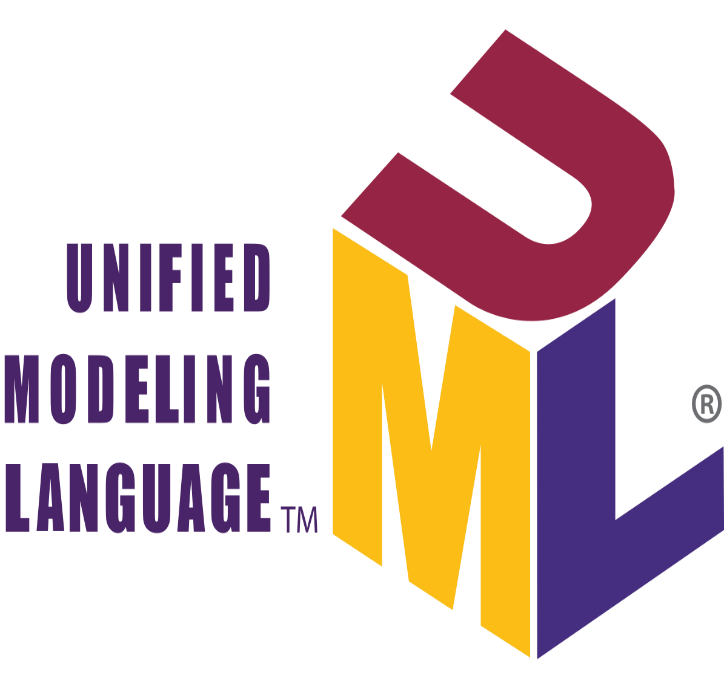
Conception et modélisation

1. **Introduction :**

Dans ce chapitre, nous allons étudier en détails les besoins fonctionnelles de l’application basé sur le cahier de charge cité dans le chapitre précédent. Pour cela nous avons choisie d’utiliser le langage de modélisation unifié UML (Unified Modeling Language).

**2 – Conception UML :**

**2.1 Présentation :**

Le langage UML (Unified Modeling Language, ou langage de modélisation unifié) a été pensé pour être un langage de modélisation visuelle commun, et riche sémantiquement et syntaxiquement. Il est destiné à l'architecture, la conception et la mise en œuvre de systèmes logiciels complexes par leur structure aussi bien que leur comportement. L'UML a des applications qui vont au-delà du développement logiciel, notamment pour les flux de processus dans l'industrie. **Figure1 : Logo UML**

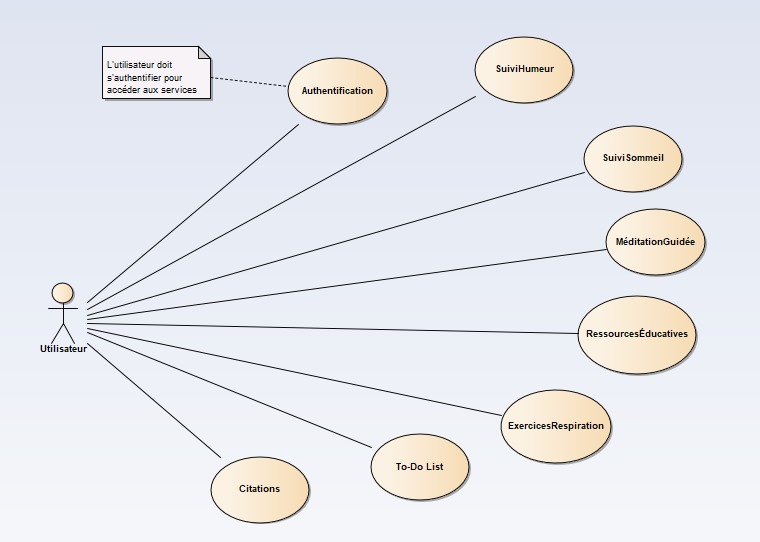
**2.2 Diagramme de cas d’utilisation :**

Le diagramme de cas d'utilisation permet de recueillir, d'analyser, d'organiser les besoins et de capturer les fonctionnalités du système avec une vision orientée utilisateur.

Son élaboration est une étape primordiale dans toute modélisation de système informatique.

Pour chaque acteur, on précisera son diagramme de cas d'utilisation :

**Cas d’utilisation général :**



**Figure2 : Diagramme de cas d’utilisation générale**

**Description détaillée :**

**Authentification :**

* L'Authentification est une étape obligatoire pour tout utilisateur de l’application avant d'accéder à l’application.

|  |  |
| --- | --- |
| **Nom de cas** | Authentification |
| **Acteur principal** | Utilisateur de l’application |
| **Objectif** | Accéder aux services càd l’application |
| **Contraintes** | Respecter la case de l’e-mail et du mot de passe et avoir un compte sinon s’inscrire |
| **Scenario normal** | 1. Saisir l’e-mail et le mot de passe. 2. L’application vérifie la saisie. 3. Vérification de la validité des informations. 4. Le système donne l’accès a l’application. |
| **Scenario d’échec** | 1. Champs vides ou incorrectes. 2. Vérification de la validité des informations :Profile inexistant ou mot de passe incorrect 3. Le système affiche un message d’erreur et donne la possibilité de récupérer le mot de passe. |
| **Post condition** | Accéder à l’application. |

**Tableau 1 : Cas d’utilisation : Authentification**

**Suivie humeur:**

|  |  |
| --- | --- |
| **Nom de cas** | Suivie humeur |
| **Acteur principal** | Utilisateur |
| **Objectif** | Enregistrer l’humeur |
| **Contraintes** | Enregistrer une humeur par jour |
| **Scenario normal** | 1. Cliquer sur le bouton AddMood. 2. Sélectionner l’emoji . 3. Cliquer sur valider. 4. Visualiser graphe humeur. |
| **Scenario d’échec** | Aucun |
| **Post condition** | Enregistrer l’humeur |

**Tableau 2 : Cas d’utilisation : Suivie humeur**

**Suivie sommeil :**

|  |  |
| --- | --- |
| **Nom de cas** | Suivie sommeil |
| **Acteur principal** | Utilisateur |
| **Objectif** | Enregistrer durée sommeil |
| **Contraintes** | Enregistrer durée de sommeil par jour |
| **Scenario normal** | 1. Sélectionner heure d’endormissement. 2. Sélectionner heure de réveil. 3. Cliquer sur valider. 4. Visualiser graphe Sommeil. |
| **Scenario d’échec** | Aucun |
| **Post condition** | Enregistrer sommeil |

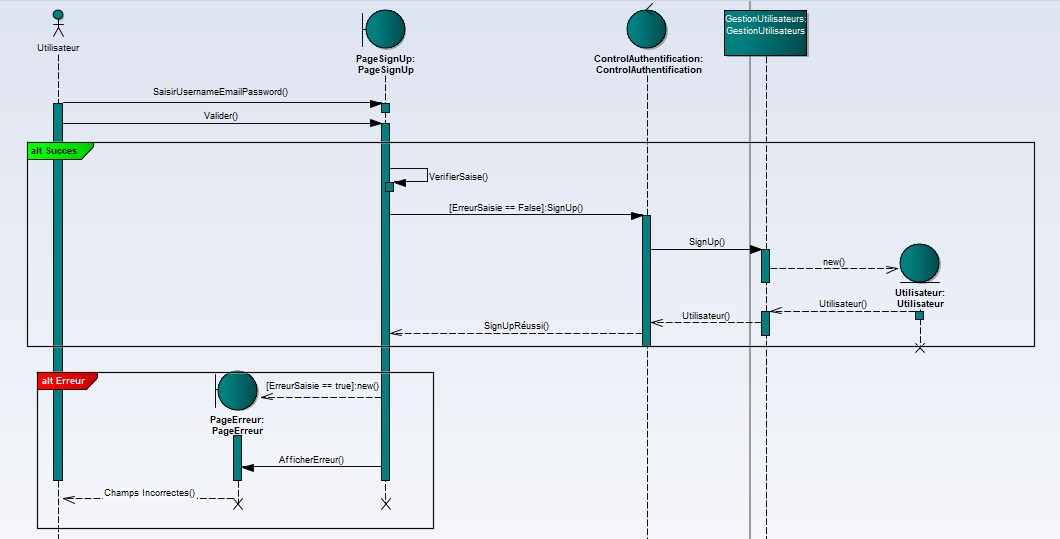
**Tableau 3 : Cas d’utilisation : Suivie sommeil**

**2.3 Diagramme de séquence:**

Les diagrammes de séquences sont la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique dans la formulation Unifie Modeling Language.

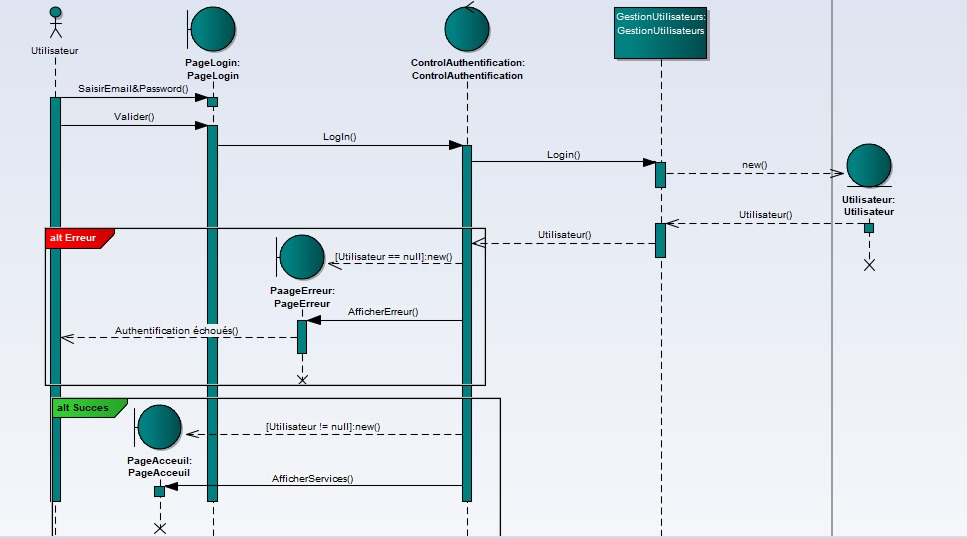
Nous présenterons quelques diagrammes de séquence de la solution.

**Authentification :**

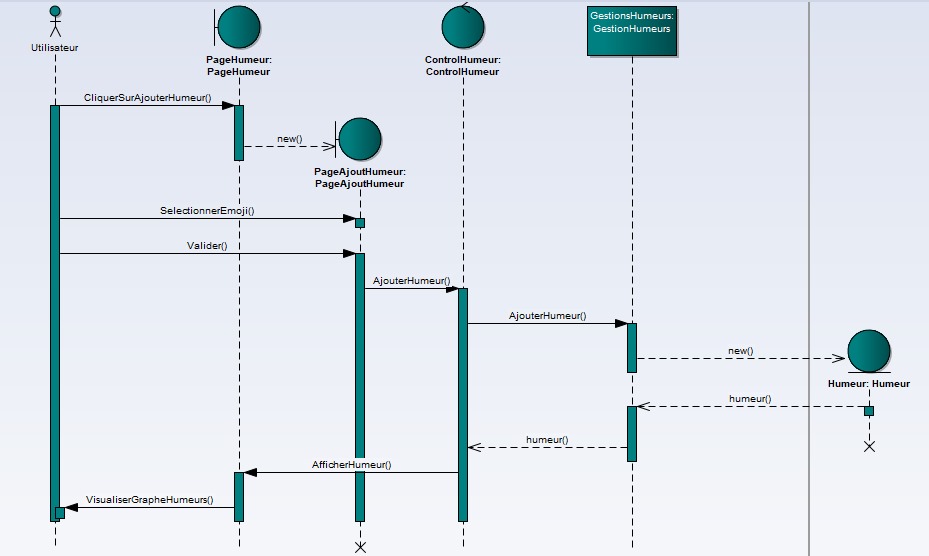
**Partie Sign Up :**

**Figure 3 : Diagramme de séquence « Sign Up »**

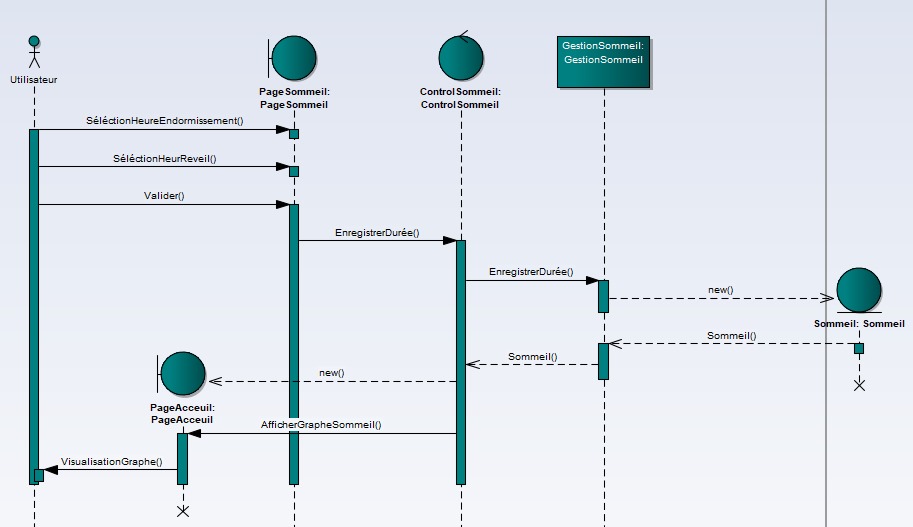
**Partie Log In :**



**Figure 4 : Diagramme de séquence « Log In »**

**Suivie humeur:**

**Figure 5 : Diagramme de séquence  « Humeur »**

**Suivie sommeil :**

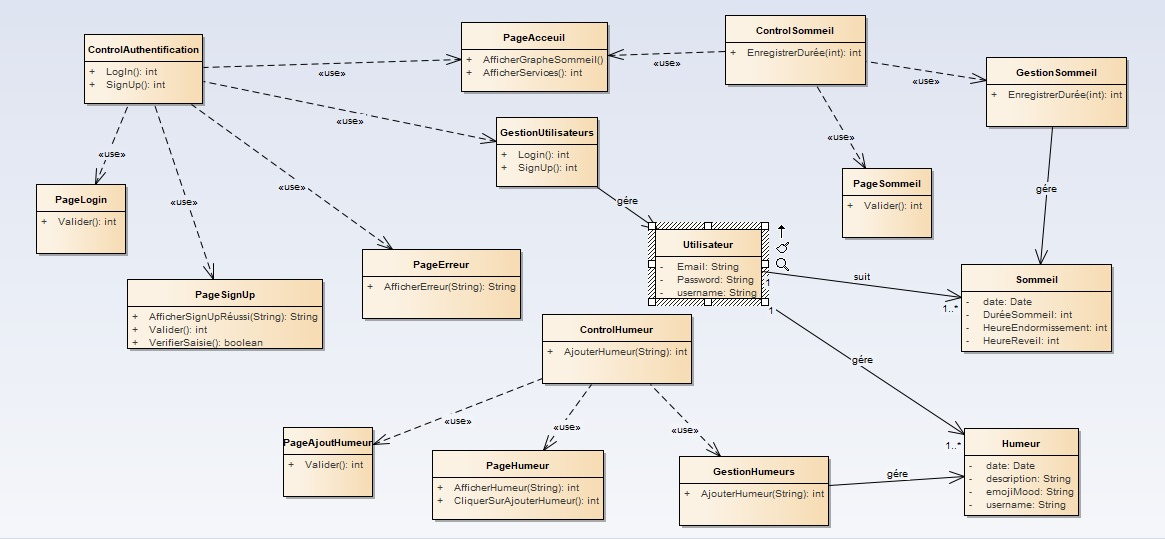
**Figure6 : Diagramme de séquence « Sign Up »**

**2.4 Diagramme de séquence:**

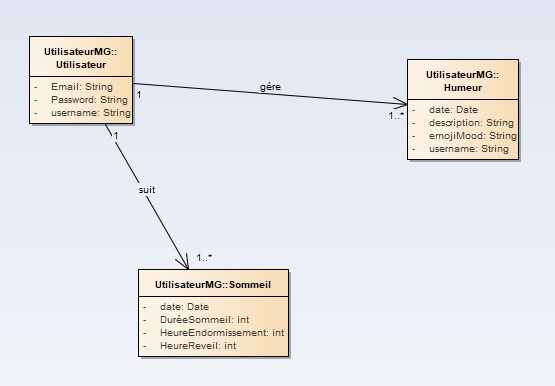
Le diagramme de classes est un schéma utilisé pour présenter les classes et les interfaces des

systèmes ainsi que leurs relations.

Notre diagramme de classes se présente sous la forme suivante :



**Figure 7 : Diagramme de classe 1**



**Figure 8 : Diagramme de classe 2**

**2.5 Conclusion :**

Ce chapitre clarifie plus la méthodologie adoptée puis la conception notamment le diagramme de cas d’utilisation, les diagrammes de séquences et finalement le diagramme de classes. Le chapitre suivant sera consacré a l’architecture et les technologies utilisé pour la réalisation de notre application.

Chapitre 3 :

Architecture et Choix Technologique

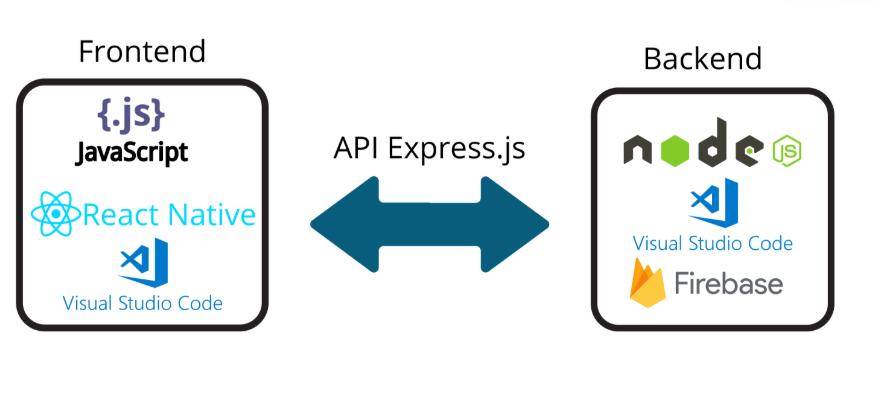
1. **Introduction :**

Apres avoir modélisé notre projet dans le chapitre précèdent, nous présenterons alors dans celui-ci les choix technologiques ainsi que l'architecture de l'application et les Framework utilisés.

1. **Architecture de l’application mobile :**

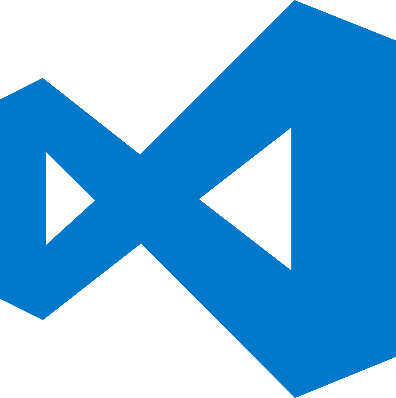
Une architecture d'application englobe les services front-end et back-end. Le développement front-end se concentre sur l'expérience utilisateur de l'application, tandis que le développement back-end assure l'accès aux données, aux services et à d'autres systèmes essentiels à l'application.

L'architecture sert de point de départ et de feuille de route pour la création de l'application.



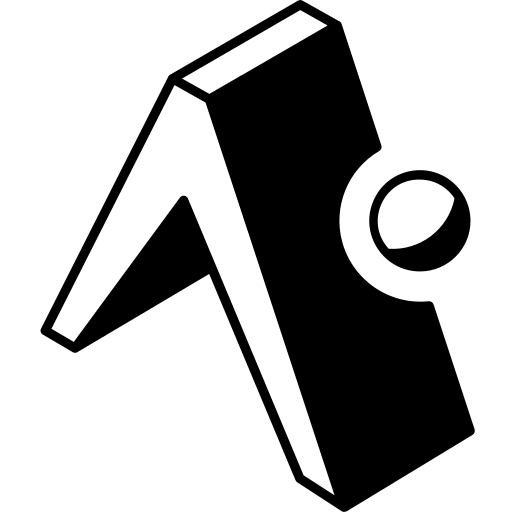
**Figure 9 : Architecture de l’application mobile**

1. **Frameworks et technologies:**

**Frameworks :**

* **Visual Studio Code :** Est un éditeur de code gratuit et open-source de Microsoft, compatible avec Windows, MacOS et Linux. Il est extensible via des extensions, dispose d'un terminal intégré, supporte le débogage et s'intègre avec Git pour le contrôle de version. VS Code est personnalisable et léger, idéal pour les développeurs.

**Figure 10 : Logo VS Code**

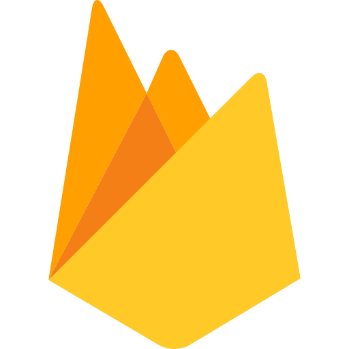


* **Expo :** C’est une plateforme qui simplifie le développement d'applications mobiles multiplateformes avec React Native. Elle offre des composants préconstruits, des API prêtes à l'emploi et des outils de déploiement pour accélérer le processus de développement et rendre la publication des applications plus facile.

**Figure 11 : Logo Expo**

* **Enterprise Architect :** Est un professionnel qui conçoit et gère l'architecture informatique d'une organisation pour aligner les stratégies technologiques avec les objectifs commerciaux. Ses responsabilités incluent l'analyse des besoins, la conception des systèmes, la planification stratégique, la gestion des projets, la mise en place de gouvernance, la communication avec les parties prenantes et l'innovation technologique. Son rôle est crucial pour assurer une infrastructure technologique efficace et alignée avec les objectifs de l'entreprise.

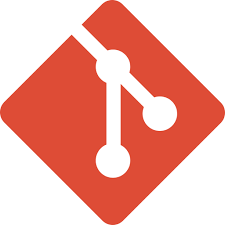
**Figure 12 : Logo Enterprise Architect**

 **Bases de données:**

* **Firebase :** C’est une plateforme de Google qui offre des outils et services pour développer, améliorer et gérer des applications mobiles et web, incluant bases de données en temps réel, authentification et notifications push.

**Figure 13 : Logo Firebase**

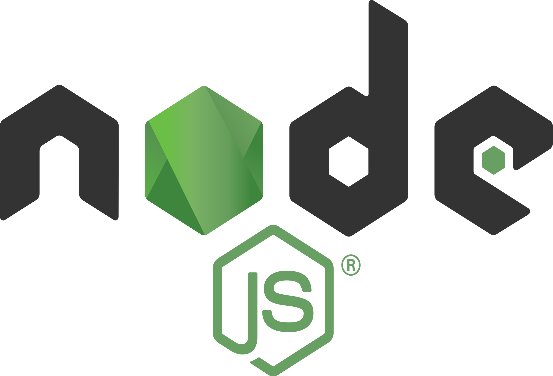
**Gestionnaire de version:**

* **Git :** C’est un système de contrôle de version distribué, conçu pour suivre les modifications apportées aux fichiers dans un projet. Créé par Linus Torvalds en 2005, Git permet aux développeurs de collaborer de manière efficace, de gérer les versions du code source, de fusionner les modifications et de maintenir un historique complet des révisions. Il est largement utilisé dans le développement logiciel pour la gestion de code source et la coordination du travail entre plusieurs développeurs.

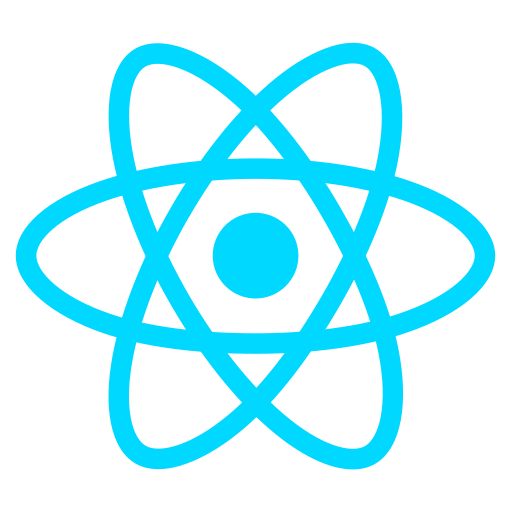
**Figure 14 : Logo de Git Technologies Back-end :**

* **Express.js :** C’est un framework web minimaliste et flexible pour Node.js, principalement utilisé pour construire des applications web et des API. Pour une application mobile, Express.js peut être utilisé pour créer le serveur backend qui gère les requêtes, les réponses et la logique métier de l'application. Il permet de gérer les routes, l'authentification, l'accès aux bases de données, et d'autres services essentiels dont une application mobile peut avoir besoin pour fonctionner correctement. En résumé, Express.js facilite le développement du côté serveur des applications mobiles, en fournissant une structure robuste et modulaire.

**Figure 15 : Logo Express.js**

* **Node.js :** est une plateforme côté serveur qui permet d'exécuter du code JavaScript. Bien qu'elle soit principalement utilisée pour le développement d'applications web, Node.js peut également être utilisée dans le cadre du développement d'applications mobiles, notamment pour la création de l'infrastructure côté serveur nécessaire au fonctionnement de l'application. Cette infrastructure peut inclure des API, des services cloud, des bases de données en temps réel et d'autres fonctionnalités essentielles pour les applications mobiles. En utilisant Node.js, les développeurs peuvent partager une partie du code entre les applications web et mobiles, ce qui permet une gestion plus efficace du code et des ressources.

**Figure 16 : Logo Node.js**

**Technologies Front-end :**

* **React Native :** C’ est un framework open-source développé par Facebook pour créer des applications mobiles multiplateformes en utilisant JavaScript et React. Il permet de construire des applications natives pour iOS et Android en partageant une grande partie du code entre les plateformes, ce qui accélère le développement et réduit les coûts.

**Figure 17 : Logo React Native**

**5 -Conclusion :**

Ce chapitre nous a permis de présenter l'environnement de réalisation de l'application mobile, en mettant l'accent sur l'environnement logiciel, les outils de gestion de version, ainsi que les outils de développement du côté backend et côté frontend.

Le dernier chapitre consiste en la phase d’implémentation qui contient les interfaces de l'application et sa fonctionnalité.

Chapitre 4 :

Mise en œuvre du projet