



DEPARTEMENT DES SCIENCES DE L'INFORMATIQUE

Rapport de Stage Obligatoire d'Été

Cycle d'ingénieur en génie logiciel

Réalisé par :

Haddad Narmine

Développement d'une application de gestion des randonnées

Réalisé au sein de :

Medianet

MEDIANET

Encadrant professionnel : Mr. Hassen Benzarti

Année Universitaire : 2024-2025

Table des matières

Introduction générale	6
1 CONTEXTE GENERAL DU PROJET	8
1.1 Présentation de l'organisme d'accueil «MEDIANET»	9
1.1.1 Présentation du secteur d'activité.	9
1.1.2 Chiffre Clé.	9
1.1.3 Valeurs et Culture	9
1.2 Présentation du projet	10
1.2.1 Contexte du projet	10
1.2.2 Présentation du sujet	10
1.2.3 Solution proposée	10
1.2.4 Travail demandé	10
1.3 Méthodologie adoptée	11
1.3.1 Présentation de la méthodologie	11
1.3.2 Backlog	13
1.4 Conclusion	15
2 ANALYSE DES BESOINS	16
2.1 Étude des besoins	17
2.1.1 Besoin fonctionnels	17

2.1.2	Besoin non fonctionnels	18
2.2	Modélisation des besoins	19
2.2.1	Langage de Conception	19
2.2.2	Identifications des acteurs	19
2.2.3	Diagramme des cas d'utilisation	20
2.3	Conclusion	21
3	ETUDE CONCEPTUELLE	22
3.1	Concept et architecture	23
3.2	Vue comportementale	24
3.3	Répartition des sprints	25
3.5	Conclusion	25
4	REALISATION	26
4.1	Environnement de travail	27
4.1.1	Environnement matériel	27
4.1.2	Environnement logiciel	27
4.2	Présentation de l'application réalisée	31
4.3	Conclusion	36
	Conclusion générale et perspective	37

Table des figures

1.1 Le logo «MEDIANET»[N01]	10
1.2 Processus de «SCRUM»[N02]	12
2.1 Langage de conception «UML»[N03]	19
2.2 Identification des acteurs	20
2.3 Diagramme de cas d'utilisation	21
3.1 «Architecture 3 tiers»[N04]	23
3.2 «Diagramme de classe	24
4.1 VS code [N05]	28
4.2 Angular [N06]	28
4.3 Node JS [N07]	29
4.4 MySQL Workbench [N08]	29
4.5 POSTMAN [N09]	30
4.6 Cacao [N10]	30
4.7 Interface Home [N11]	31
4.8 Interface Login[N12]	31
4.9 Interface Register[N13].	32
4.10 Interface des Randonnées[N14]	33
4.11 Interface d'une Randonnée[N15].	33

4.12 Interface de Participation [N16]	34
4.13 Interface de Remerciement[N17]	34
4.14 Interface de Feedback[N18]	35
4.15 Interface de gestion des randonnées[N19]	35
4.16 Interface d'ajoute d'une randonnées[N20]	35
4.17 Interface de modification d'une randonnées[N21]	36
4.18 Interface de suppression de randonnée[N22]	36

Liste des tableaux

1.1	Niveau de complexité	13
1.2	tableau Backlog	13
3.1	«La répartition des sprints»	25
4.1	Environnement matériel	27

INTRODUCTION GENERALE

Introduction générale

À l'ère numérique, l'informatique continue de transformer tous les domaines d'activité humaine. Cette transformation s'explique par sa capacité à automatiser les traitements, centraliser les données et exécuter rapidement les tâches. Tous les secteurs peuvent tirer parti de ces avantages pour devenir plus compétitifs. La plupart des entreprises cherchent à améliorer leurs processus de gestion des données pour maximiser leur efficacité et leur performance.

La gestion des randonnées, une activité de plein air de plus en plus populaire, n'échappe pas à cette tendance. Organiser des randonnées peut être complexe et exigeant, nécessitant une coordination minutieuse entre les organisateurs et les participants. Pour relever ces défis, ce projet propose de créer une plateforme numérique dédiée à la gestion des randonnées. Cette plateforme permettra de centraliser les informations, d'automatiser les processus logistiques et de faciliter la communication, offrant ainsi une expérience plus fluide et sécurisée pour tous les utilisateurs. En intégrant des fonctionnalités avancées et une interface conviviale, cette solution vise à transformer la manière dont les randonnées sont organisées et vécues, en exploitant pleinement les avantages de l'ère numérique.

De ce fait, nous avons élaboré ce rapport contenant toutes les informations relatives à ce projet qui sera réparti en trois chapitres définis comme suit :

Le premier chapitre est consacré à la présentation de l'organisme d'accueil, à la description du contexte général du projet et à la démarche adoptée pour la conduite du projet

Dans le deuxième chapitre, nous allons faire un étude sur l'existant et spécifier les besoins fonctionnels et non fonctionnels.

Chapitre 1

CONTEXTE GENERAL DU PROJET

Introduction

Dans ce chapitre nous allons mettre le projet dans son cadre général , nous allons présenter en premier lieu l'organisme d'accueil, à posteriori le cadre de notre projet, ainsi que les objectifs à atteindre et finalement nous allons exposer la solution proposée accompagnée de la méthodologie adoptée.

I. Présentation de l'organisme d'accueil «MEDIANET»

1.1 Présentation du secteur d'activité

Medianet est une agence digitalement transformée avec plus de 20 ans d'expérience dans le web, le mobile et le marketing digital. En tant que leader dans la transformation digitale 360°, Medianet accompagne les entreprises dans leur transition vers le numérique. La vision de Medianet est de développer un écosystème basé sur le co-leadership, rassemblant des experts créatifs et socialement responsables pour promouvoir l'innovation et la transformation technologique. Sa mission est de créer, développer et optimiser les activités commerciales des entreprises à travers le monde en utilisant la technologie et la créativité.

1.2 Chiffres Clés

- **21 ans** d'expertise dans le digital
- **Présence dans 24 pays** sur 4 continents
- **100 consultants**
- **48% femmes** et **52% hommes**
- **2900 projets** réussis dans divers secteurs

1.3 Valeurs et Culture

Medianet valorise la diversité culturelle et sociale pour renforcer le sentiment d'appartenance. La collaboration est au cœur de sa culture, visant à développer une société co-créatrice et solidaire. Medianet encourage ses employés, appelés les "medianauts", à explorer leur potentiel d'innovation et à promouvoir le co-leadership pour progresser continuellement.

Aujourd'hui, Medianet est présente sur quatre continents et continue d'étendre son activité à l'échelle internationale, avec des opérations au Canada, aux USA, en France, en Angleterre, en Allemagne, en Italie, en Tunisie, en Algérie, au Maroc, en Mauritanie, au Sénégal, en Côte d'Ivoire, au Burkina Faso, au Bénin, au Mali, en

Guinée, au Gabon, au Congo, au Burundi, au Tchad, en Libye, en Égypte, en Arabie Saoudite, au Yémen et aux Émirats Arabes Unis.



Figure 1.1 – Le logo «MEDIANET»[N01]

II. Présentation du projet

2.1. Contexte du projet

Afin de valider notre première année cycle d'ingénieur, nous sommes appelées à effectuer un stage d'été au sein d'une entreprise dans le but de mettre en valeur nos compétences acquises tout au long de notre cursus universitaire à la faculté des sciences de Tunis.

2.2. Présentation du sujet

Le projet consiste à développer une plateforme numérique pour gérer efficacement les randonnées.

2.3. Solution proposée

La réponse est la création d'une application web dédiée à la gestion des randonnées. Cette application permettra aux organisateurs de créer des itinéraires, de gérer les inscriptions, de suivre les participants en temps réel et de collecter des feedbacks post-randonnée. Les participants pourront s'inscrire facilement, recevoir

des informations pratiques et interagir avec les organisateurs et les autres randonneurs.

2.4. Travail demandé

Notre travail consiste à réaliser une application dédiée aux organisateurs et aux participants de randonnées. Notre projet se présente comme suit :

1. Identification du besoin métier et formalisation du cahier des charges fonctionnel pour nous guider dans la réalisation du projet.
2. Conception de la plateforme.
3. Maîtrise de l'architecture et du mode de fonctionnement des outils et des frameworks de développement : Angular, springboot, postgresql, VS Code.
4. Réalisation du projet.

Cette application permettra de :

- Alerter en temps réel par mail pour toute mise à jour ou changement important.
- Gérer les inscriptions et les données des participants.
- Visualiser les itinéraires et les détails des randonnées.
- Gérer les utilisateurs et les rôles au sein de la plateforme.

III. Méthodologie adoptée

3.1. Présentation de la méthodologie

Le choix de la méthodologie de développement est crucial pour assurer le succès du projet, en définissant les étapes clés et en précisant les caractéristiques essentielles de son fonctionnement. Ainsi, une approche de développement adaptée aux spécificités et aux exigences du projet est sélectionnée, garantissant la création d'un produit de haute qualité qui répond pleinement aux besoins et aux attentes des utilisateurs. Dans ce contexte, nous avons opté pour une approche itérative et collaborative, favorisant la méthode Agile.

SCRUM:

Pour garantir le succès du projet, nous adoptons une approche agile qui favorise l'adaptabilité et la collaboration continue. Cette méthodologie permet une gestion efficace en suivant un processus itératif et incrémental. Voici les principales étapes :

- Collecte des besoins: Les fonctionnalités prioritaires sont recueillies dans un backlog, représentant les objectifs du produit définis en collaboration avec les parties prenantes.

- Planification et itérations: Le développement du produit est divisé en itérations appelées sprints, généralement d'une durée fixe. Chaque sprint est défini par l'équipe projet en fonction des priorités et de sa capacité de travail.
- Exécution et ajustements: Pendant chaque sprint, des réunions quotidiennes permettent à l'équipe de synchroniser ses efforts et d'ajuster les priorités en temps réel pour maximiser l'efficacité.
- Revues et adaptations : À la fin de chaque sprint, une revue est menée pour évaluer l'incrément du produit livrable. Cette évaluation guide les ajustements du backlog pour les sprints suivants, assurant ainsi une progression continue du projet.

Cette approche agile favorise la transparence, l'adaptabilité et la livraison continue de valeur, tout en intégrant les retours d'utilisateur et en maximisant la satisfaction des parties prenantes.

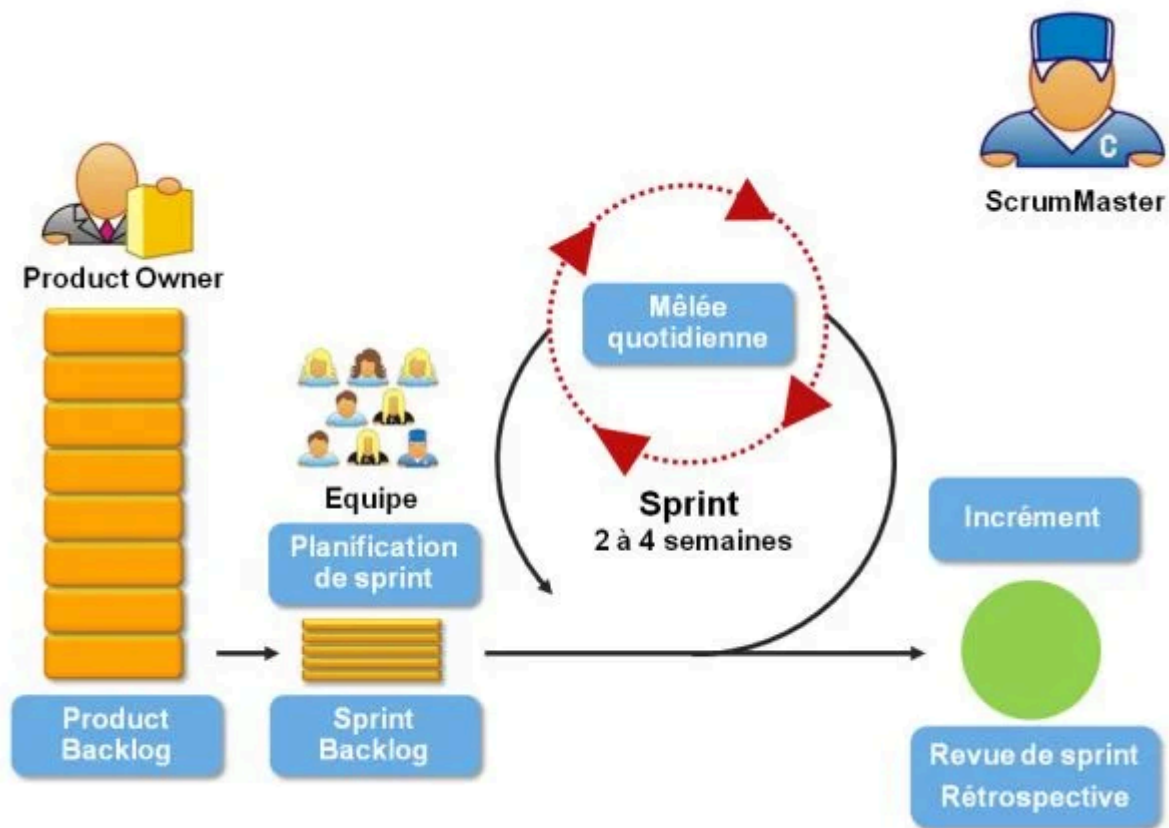


Figure 1.2 – Processus de «SCRUM»[N02]

3.2. Backlog

Initialement, il est essentiel d'établir une liste structurée des fonctionnalités prioritaires pour les utilisateurs (le backlog). Ce backlog évoluera tout au long du développement du produit, reflétant continuellement les besoins et les priorités des utilisateurs.

Le tableau suivant indique les niveaux de complexités :

Complexité	Niveau de complexité
De 0 à 2	Facile
De 2 à 4	Simple
De 5 à 8	Normal
De 8 à 13	Difficile
De 13 à 20	Très difficile

Table 1.1 – Niveau de complexité

Notre «Backlog» est composé de 10 lignes et de 5 colonnes avec :

- ID : représente l'identifiant du user story
- Feature : pour mieux ordonner les user stories
- User story : comporte la description de la fonctionnalité
- La Complexité : c'est le degré de difficulté technique de «User Story»
- La priorité : de «user story» selon la valeur métier et l'ordre de réalisation

ID	Feature	User Story	Priorité	Complexité
1	Gestion des randonnées	En tant qu'organisateur de la randonnée , je veux définir les itinéraires(les points de départs,d'arrivée, d'intérêt...)	1	8
2		En tant qu'organisateur de la randonnée , je veux fixer les dates et heures.	2	2
3		En tant qu'organisateur de la randonnée , je veux ajouter des descriptions détaillées(niveaux de difficulté, distance ...)	3	6

4		En tant qu'organisateur de la randonnée , je veux intégrer des cartes et traces GPS.	5	11
5		En tant qu'organisateur de la randonnée , je veux définir le nombre maximum de participants pour chaque randonnée.	6	5
6		En tant qu'organisateur de la randonnée , je veux identifier le budget pour chaque randonnée.	7	9
7	Gestion des participants	En tant qu'organisateur de la randonnée , je veux gérer les inscriptions (approuver ou refuser les demandes de participants)	8	8
8		En tant qu'organisateur de la randonnée , je veux envoyer les informations aux participants avant la randonnée.	11	7
9		En tant qu'organisateur de la randonnée , je veux envoyer des confirmations d'inscription et des rappels aux participants avant la randonnée.	10	7
10		En tant qu'organisateur de la randonnée , je veux maintenir une liste à jour des participants inscrits.	9	4
11	Logistique et transport	En tant qu'organisateur de la randonnée , je veux organiser l'équipement nécessaire.	11	8
12		En tant qu'organisateur de la randonnée , je veux réserver le transport et l'hébergement.	13	13
13		En tant qu'organisateur de la randonnée , je veux communiquer avec les autorités locales afin d'obtenir les autorisations nécessaires.	12	14
14	La sécurité et la gestion des risques	En tant qu'organisateur de la randonnée , je veux identifier les risques potentiels.	14	15
15		En tant qu'organisateur de la randonnée , je veux préparer les plans d'urgence .	16	20
16		En tant qu'organisateur de la randonnée , je veux élaborer les guides sur les mesures de sécurité.	15	13
17	Evaluation et feedback	En tant qu'organisateur de la randonnée , je veux collecter les avis des participants après la randonnée.	23	8
18		En tant qu'organisateur de la randonnée , je veux analyser les retours afin d'améliorer les futures randonnées.	24	17

19		En tant que participant, je veux évaluer l'expérience en laissant des commentaires.	22	7
20	Exploration et Inscription aux Randonnées	En tant que participant, je veux parcourir les randonnées disponibles sur la plateforme.	17	2
21		En tant que participant, je veux consulter les détails des randonnées.	18	5
22		En tant que participant, je veux m'inscrire.	19	7
23		En tant que participant, je veux consulter les informations pratiques fournies par l'organisation(horaires, point de rendez-vous).	20	3
24		En tant que participant, je veux participer au groupe de discussion afin d'interagir avec d'autres participants.	21	9
				Complexité Totale : 208

Table 1.2 : tableau Backlog

3.3. Conclusion

Dans ce chapitre , nous avons présenté l'organisme d'accueil , la problématique et la solution proposée , nous avons terminé ce chapitre avec la choix de la méthodologie Scrum avec le backlog Dans le chapitre suivant , nous allons passer vers l'étude de l'existant , analyse de besoins et la réalisation des maquettes qui vont nous aider dans la réalisation du projet.

Chapitre 2

ANALYSE DES BESOINS

Introduction

Il est essentiel de spécifier et d'analyser les besoins. Dans cette section, nous allons finaliser l'étude théorique du projet. Nous détaillerons les besoins fonctionnels et non fonctionnels afin de définir les qualités requises pour le projet et d'identifier les acteurs impliqués.

I. Étude des besoins

Dans cette section, nous nous intéressons aux besoins des utilisateurs concernés par le projet, et nous nous concentrerons sur la caractérisation de l'ensemble des propriétés à travers des spécifications fonctionnelles et non fonctionnelles.

1.1. Besoin fonctionnels

Les Besoins fonctionnels se présentes en 6 grandes parties :

- **La gestion des randonnées**

Elle consiste à définir les itinéraires, fixer les dates et heures, ajouter des descriptions détaillées, intégrer des cartes et traces GPS, définir le nombre maximum de participants, et identifier le budget pour chaque randonnée.

- **La gestion des participants**

Elle permet à l'organisateur de gérer les inscriptions, envoyer des informations et rappels, et maintenir une liste à jour des participants inscrits.

- **Logistique et transport**

Elle permet à l'organisateur de préparer l'équipement, réserver le transport et l'hébergement, et obtenir les autorisations nécessaires des autorités locales.

- **Sécurité et la gestion des risques**

Elle consiste à identifier les risques potentiels, préparer des plans d'urgence, et élaborer des guides de sécurité.

- **Exploration et Inscription aux Randonnées**

Elle permet au participant de parcourir et consulter les randonnées disponibles, de s'inscrire, d'accéder aux détails des randonnées ainsi qu'aux informations pratiques fournies par l'organisation, et de participer à des discussions avec d'autres participants.

- **Evaluation et feedback**

Elle permet au participant d'évaluer leur expérience en laissant des commentaires. Ainsi, la collecte et l'analyse des avis des participants par les organisateurs après la randonnée.

1.2. Besoin non fonctionnels

Maintenant nous présentons ci-dessous l'ensemble des contraintes à respecter pour garantir la performance du système en considérant les exigences de l'utilisateur.

1.2.1. Sécurité

- Protection des données personnelles des utilisateurs.
- Sécurisation des transactions financières.

1.2.2. Fiabilité

- Disponibilité élevée de la plateforme.
- Sauvegarde régulière des données.

1.2.3. Performance

- Temps de réponse rapide de l'application.
- Capacité à gérer un grand nombre d'utilisateurs simultanément.

1.2.4. Compatibilité

- Compatibilité avec divers navigateurs web et appareils (ordinateurs, tablettes, smartphones).
- Intégration avec des outils de cartographie et de navigation GPS.

1.2.5. Scalabilité

- Capacité à évoluer pour gérer un nombre croissant d'utilisateurs et de randonnées.
- Flexibilité pour ajouter de nouvelles fonctionnalités à l'avenir.

1.2.6. Maintenance

- Facilité de mise à jour et de maintenance de la plateforme.
- Documentation claire pour les utilisateurs et les administrateurs.

En prenant en compte ces aspects fonctionnels et non fonctionnels, la plateforme de gestion de randonnée sera à la fois robuste, performante et user-friendly, offrant une expérience optimale pour les organisateurs et les participants.

II. Modélisation des besoins

Dans ce chapitre, nous traduisons les exigences en diagrammes UML pour une meilleure visualisation. Nous introduisons ces diagrammes et concluons avec une représentation globale du diagramme de cas d'utilisation.

2.1. Langage de Conception

UML: Unified Modeling Language en anglais, est un langage standardisé utilisé pour spécifier, visualiser, construire et documenter les artefacts d'un système logiciel, Il est couramment utilisé en développement logiciel et en conception orientée objet. UML fournit une série de diagrammes pour représenter les différentes perspectives et aspects d'un système.



Figure 2.1 – Langage de conception «UML»[N03]

2.2. Identifications des acteurs

Les acteurs sont des personnes ou des systèmes externes qui interagissent avec le système à développer pour obtenir une fonctionnalité ou une valeur ajoutée.

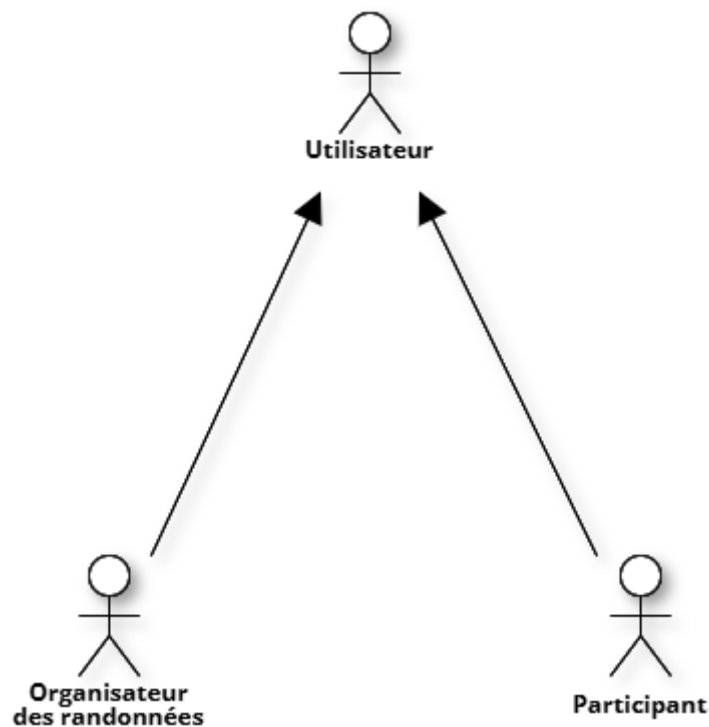


Figure 2.2 – Identification des acteurs

Les acteurs sont des entités externes qui interagissent avec le système pour obtenir une valeur ajoutée, citant :

- **Organisateur des randonnées:** il s'occupe de la gestion des randonnées, des participants, sécurité, gestion des risques et analyse les avis des participants dans le but d'améliorer les futures randonnées.
- **Participant:** une personne qui s'occupe seulement de parcourir les randonnées, participer en une, consulter les informations pratiques fournies par l'organisation, enfin, évaluer l'expérience en laissant commentaire.

2.3. Diagramme des cas d'utilisation

Nous allons présenter les besoins de manière formelle, c'est à dire en utilisant le diagramme des cas d'utilisation du UML.

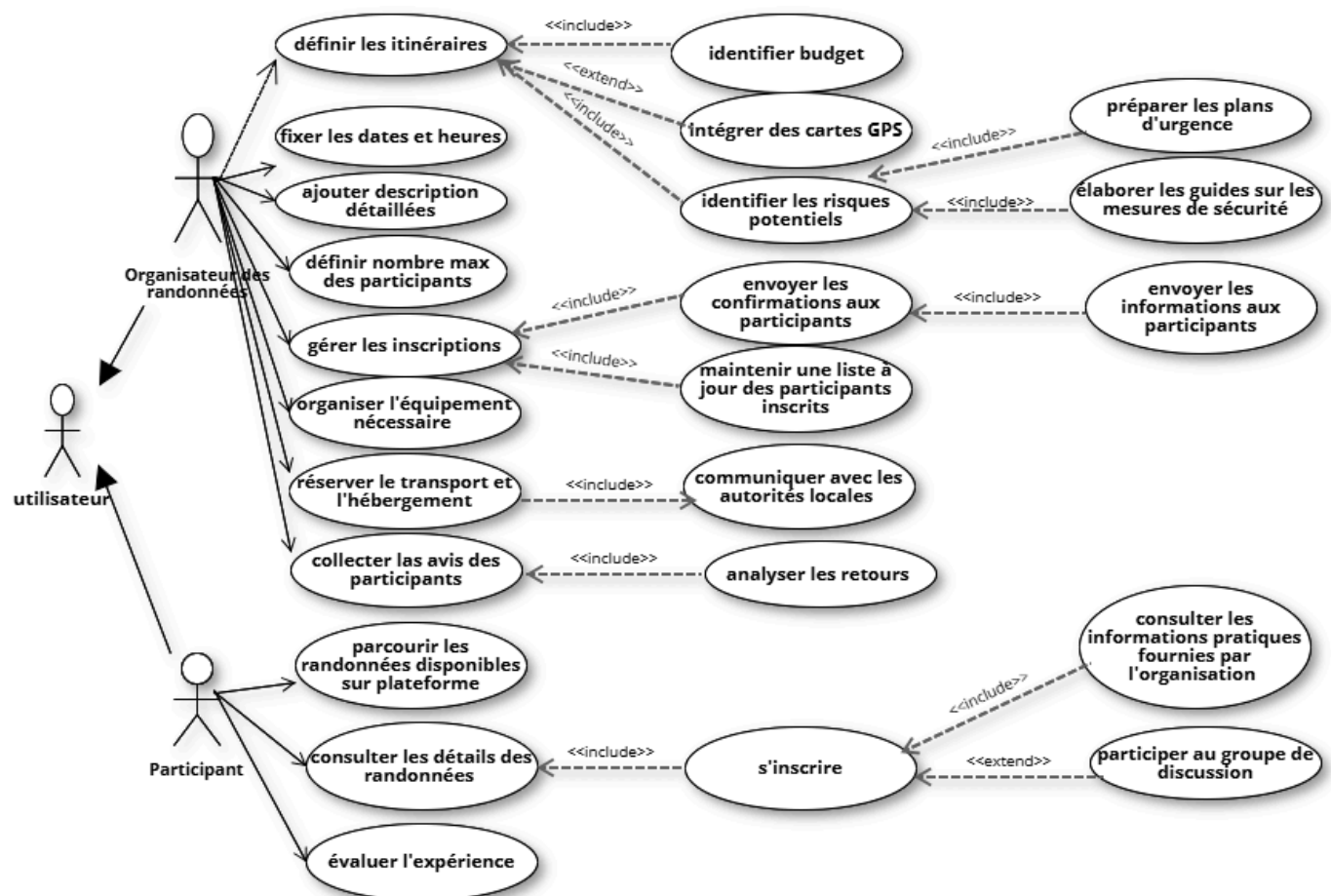


Figure 2.3 – Diagramme de cas d'utilisation

III. Conclusion

Nous avons fourni dans ce chapitre une analyse détaillée de ces besoins indispensables pour mieux faciliter le travail à réaliser. Nous allons voir dans le chapitre suivant l'architecture de notre système tout en mentionnant toutes les actions possibles, les acteurs et les diagramme

Chapitre 3

ETUDE CONCEPTUELLE

Introduction

I. Concept et architecture

Notre projet consiste à réaliser une application web en se basant sur l'architecture 3 tiers, constituée de trois couches distinctes :

1. Couche de Présentation (Client) :

- **Description** : Cette couche comprend les interfaces homme-machine (IHM) que les utilisateurs interagissent avec. Cela inclut les pages web qui affichent les itinéraires de randonnée, les détails, les formulaires d'inscription, et d'autres informations pertinentes.
- **Technologies** : frameworks front-end comme Angular.

2. Couche de Logique d'Application

- **Description** : elle gère les processus de validation des inscriptions, la gestion des itinéraires, l'envoi des notifications, et l'analyse des retours des participants. Elle orchestre également les communications entre la couche de présentation et la couche d'accès aux données.
- **Technologies** : Node.js, Express.js, ou d'autres frameworks de serveur d'applications.

3. Couche d'Accès aux Données

- **Description** : Cette couche gère l'accès aux données et leur stockage. Elle comprend les bases de données où sont stockées les informations sur les itinéraires de randonnée, les inscriptions des participants, les avis, et autres données pertinentes.
- **Technologies** : SGBD comme PostgreSQL.

Cette architecture assure une organisation claire, une maintenance simplifiée et une scalabilité accrue pour l'application de gestion des randonnées. Elle possède plusieurs avantages :

- elle diminue la complexité lors de la conception
- elle nous facilite la maintenance de l'application
- une meilleure répartition de la charge entre les différents entités clients et serveurs

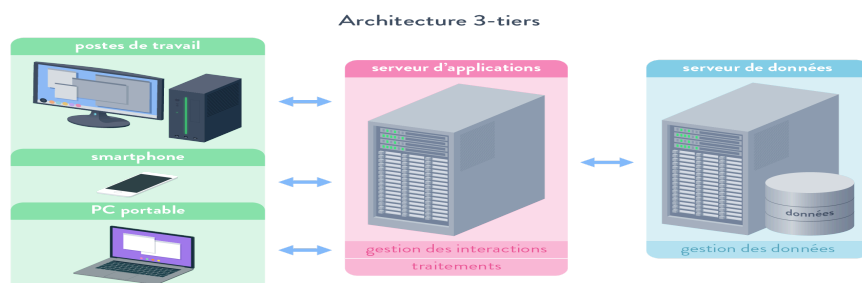


Figure 3.1 – «Architecture 3 tiers »[N04]

II. Vue comportementale

- Définition de diagramme de classes

Les diagrammes de classes sont considérés comme les plus importants dans la modélisation orientée objet, et c'est le seul qui doit être utilisé dans ce type de processus de modélisation. Le diagramme de classes montre la structure interne du système. Il fournit une représentation abstraite des objets système qui interagissent pour réaliser des cas d'utilisations. Il s'agit d'une vue statique car le facteur temps dans le comportement du système n'est pas pris en compte.

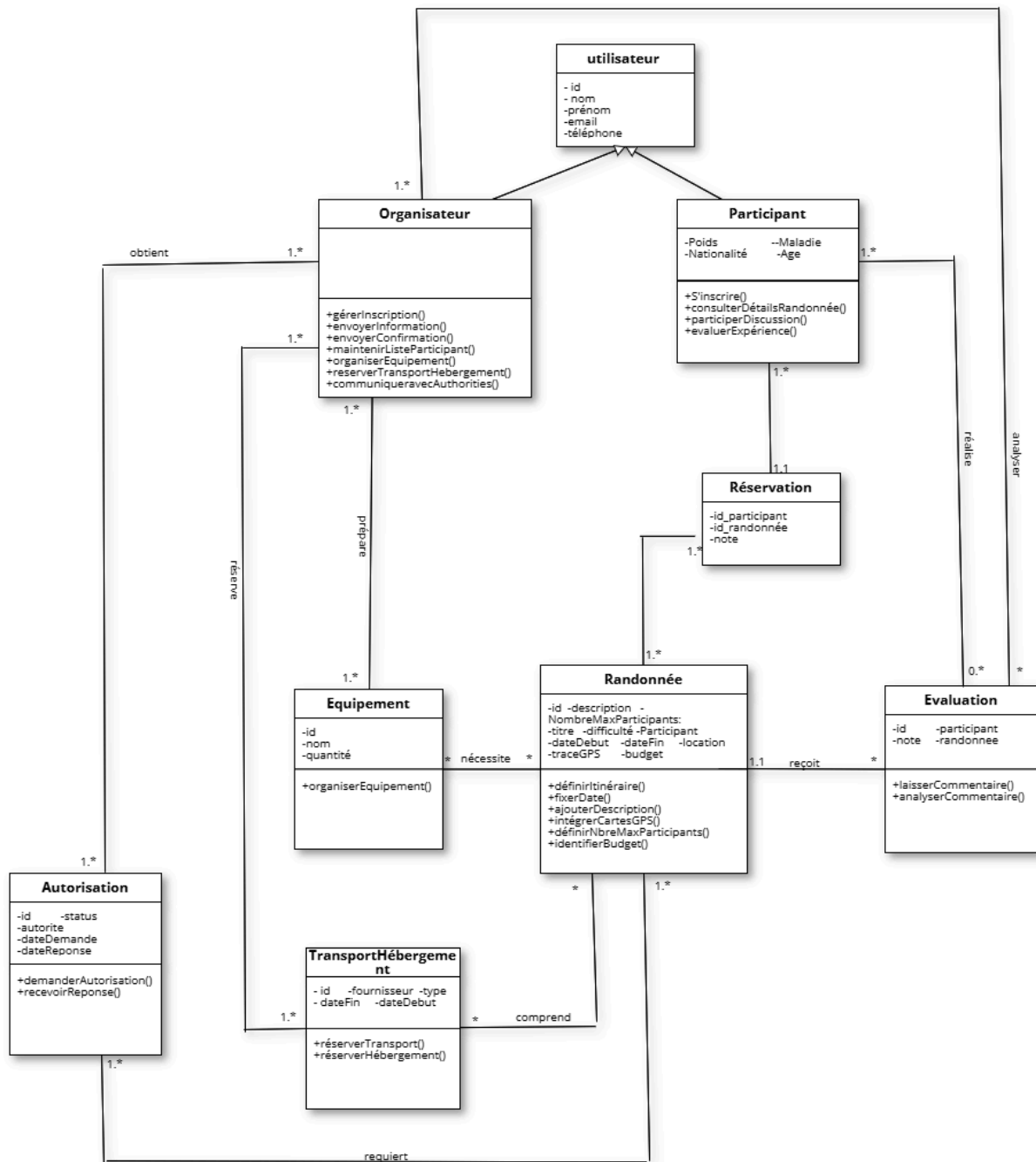


Figure 3.2 – Diagramme de classe

III. Répartition des sprints

Une fois qu'on a déterminé le «backlog» du produit, nous le découpons en parties qui s'appellent des «sprints». Le tableau suivant décrit la répartition de ces itérations :

<<Sprint 1>>	de priorité 1 jusqu'à 5
<<Sprint 2>>	de priorité 6 jusqu'à 10
<<Sprint 3>>	de priorité 11 jusqu'à 15
<<Sprint 4>>	de priorité 16 jusqu'à 20

Table 3.1 : «La répartition des sprints»

IV. Conclusion

Pour conclure ce chapitre, nous avons exploré les concepts fondamentaux et l'architecture en trois tiers qui structurent notre projet. Nous avons également détaillé le diagramme de classes et la réalisation des sprints, offrant ainsi une vue d'ensemble claire et cohérente du développement et de la mise en œuvre de notre solution.

Chapitre 4

RÉALISATION

Introduction

Le choix des outils de développement et des technologies adéquates est crucial pour garantir l'efficacité et la pérennité du projet. Une sélection judicieuse facilite non seulement le processus de développement et de maintenance, mais assure également une performance optimale. Dans ce chapitre, nous aborderons d'abord la sélection des technologies appropriées, suivie d'une présentation des frameworks utilisés, de l'architecture logicielle adaptée, et enfin des interfaces homme-machine mises en place.

I. Environnement de travail

Nous allons présenter dans cette partie les choix relatifs à l'environnement matériel ainsi que l'environnement logiciel.

1. Environnement matériel

La réalisation de cette application a été effectuée en utilisant un ordinateur portable avec les caractéristiques suivantes :

Marque	Dell PC Portable VOSTRO 3500
Processeur	11th Gen Intel(R) Core(TM) i3-1115G4 @ 3.00GHz 3.00 GHz
Mémoire	16,0 Go
Carte Graphique	Intel(R) UHD Graphics
Système d'exploitation	Windows 10 Professionnel

Table 4.1 – Environnement matériel

2. Environnement logiciel

* Technologies utilisées

Nous avons travaillé sur plusieurs frameworks et logiciels pour pouvoir réaliser ces applications qui sont :

- **VS Code :**

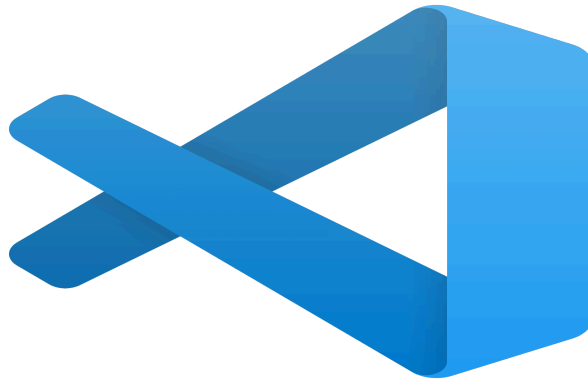


Figure 4.1 – VS code [N05]

Visual Studio Code est un éditeur de code flexible développé par Microsoft pour Windows, Linux et macOS. Il offre des fonctionnalités telles que le débogage, la coloration syntaxique, l'autocomplétion du code, et l'intégration de Git. Les utilisateurs peuvent personnaliser leur environnement de travail avec des thèmes, des raccourcis clavier et des extensions.

- **Angular :**



Figure 4.2 – Angular [N06]

Angular est un framework open source développé par Google pour la création d'applications web modernes, notamment des Single Page Applications. Basé sur TypeScript, il offre une architecture MVC pour séparer les données, la présentation et la logique, facilitant ainsi la maintenance et le développement collaboratif.

- **Node JS :**

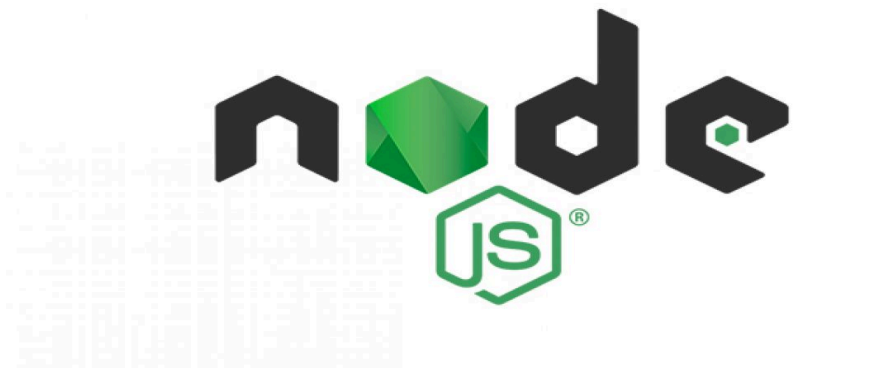


Figure 4.3 – Node JS [N07]

Node.js est une plateforme open source en JavaScript conçue pour les applications réseau hautement concurrentes. Utilisant la machine virtuelle V8 et la librairie libuv, elle permet l'exécution de JavaScript côté serveur et inclut des modules natifs pour le développement de serveurs HTTP. Node.js simplifie le déploiement d'applications web en évitant la nécessité de serveurs web externes comme Nginx ou Apache.

- **MySQL Workbench :**

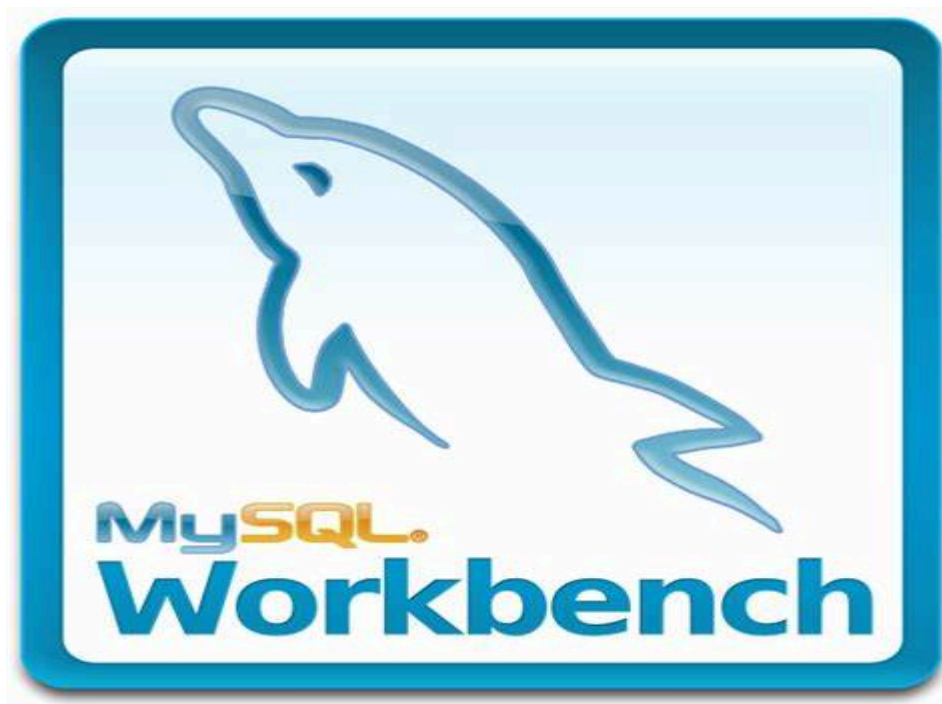


Figure 4.4 – MySQL Workbench [N08]

- **Postman :**

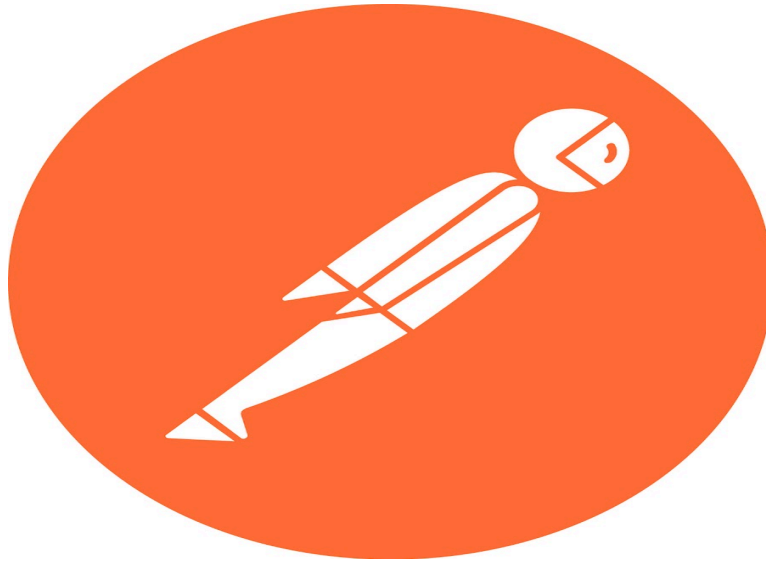


Figure 4.5 – Postman [N09]

Postman est un outil de développement API qui permet de concevoir, tester et documenter les requêtes HTTP. Il offre une interface utilisateur intuitive pour envoyer des requêtes, inspecter les réponses et automatiser les tests d'API. Postman facilite le développement et le débogage des services web.

- **Lucidchart :**



Figure 4.6 – Cacoo [N10]

Cacoo est un outil en ligne de création de diagrammes collaboratifs, permettant de concevoir des schémas, des organigrammes et des maquettes en temps réel. Il offre des fonctionnalités de collaboration en simultané, facilitant le travail d'équipe sur des projets visuels. Cacoo supporte divers types de diagrammes et facilite le partage et l'intégration avec d'autres outils.

II. Présentation de l'application réalisée

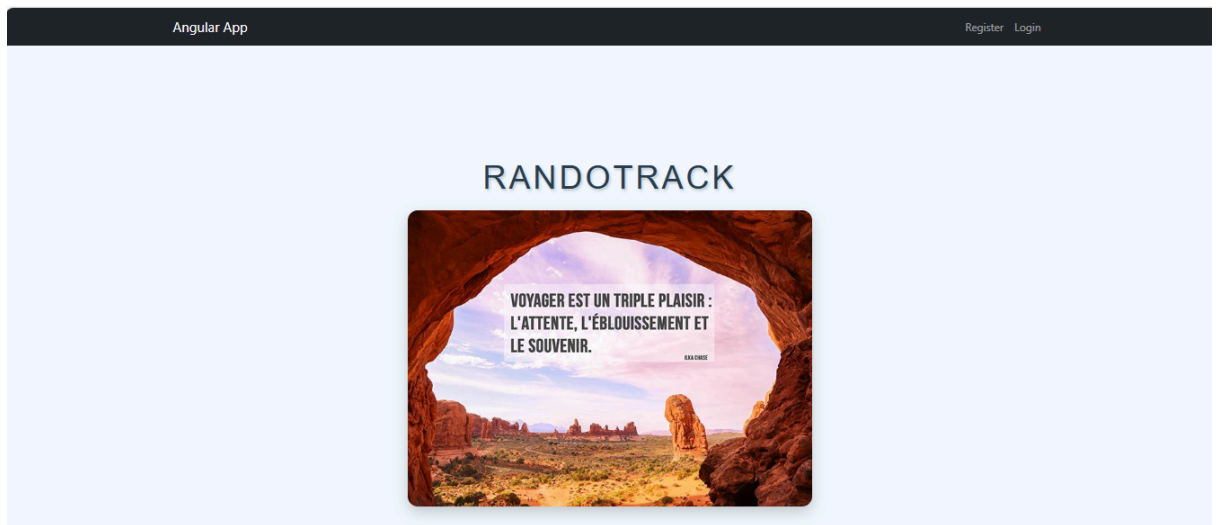


Figure 4.7 – Interface Home [N11]

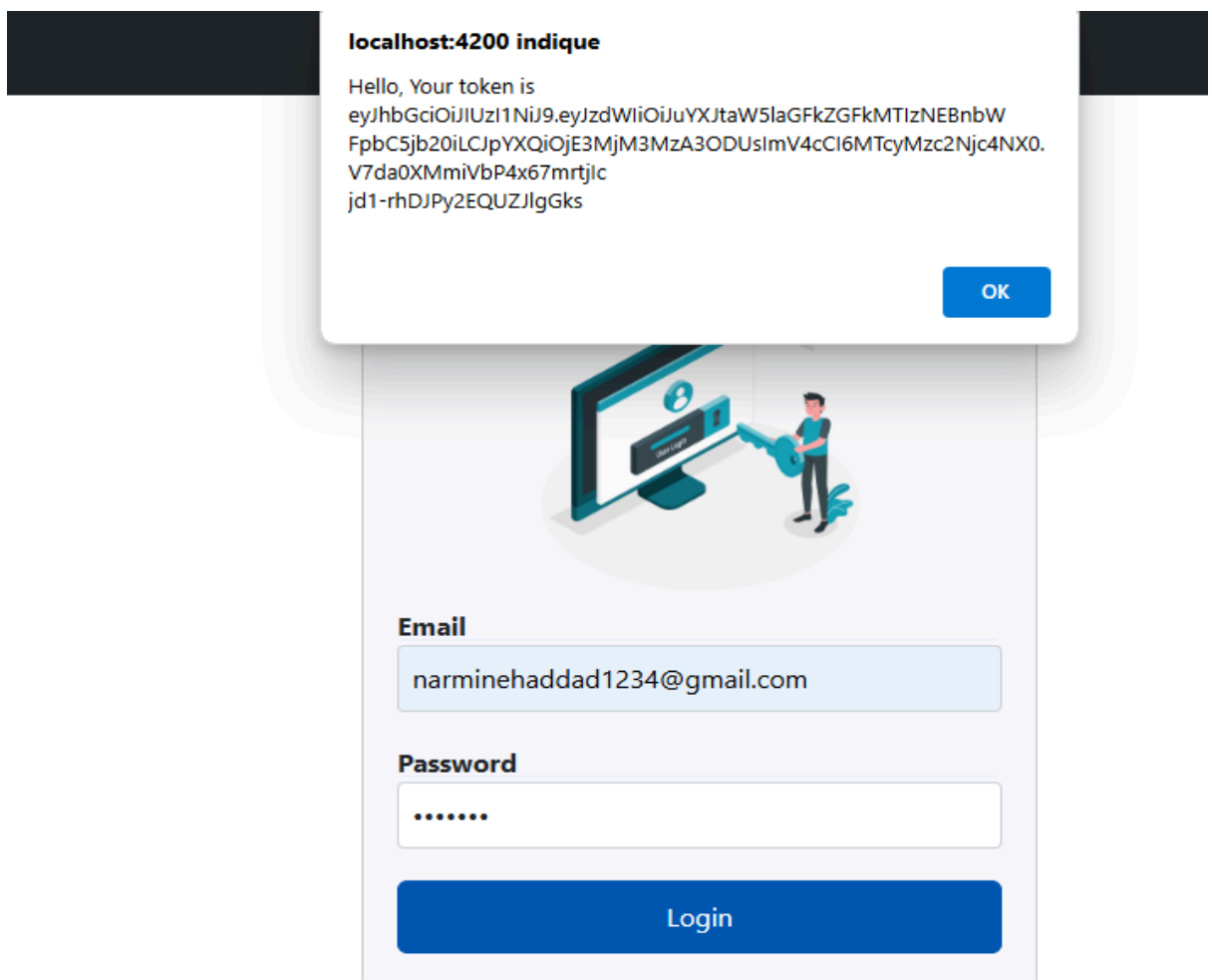



Figure 4.8 – Interface Login [N12]

localhost:4200 indique

Hello Narmine

OK



Name

Email

Password

Confirm Password

Register

Figure 4.9 – Interface Register [N13]



Figure 4.10 – Interface des Randonnées [N14]

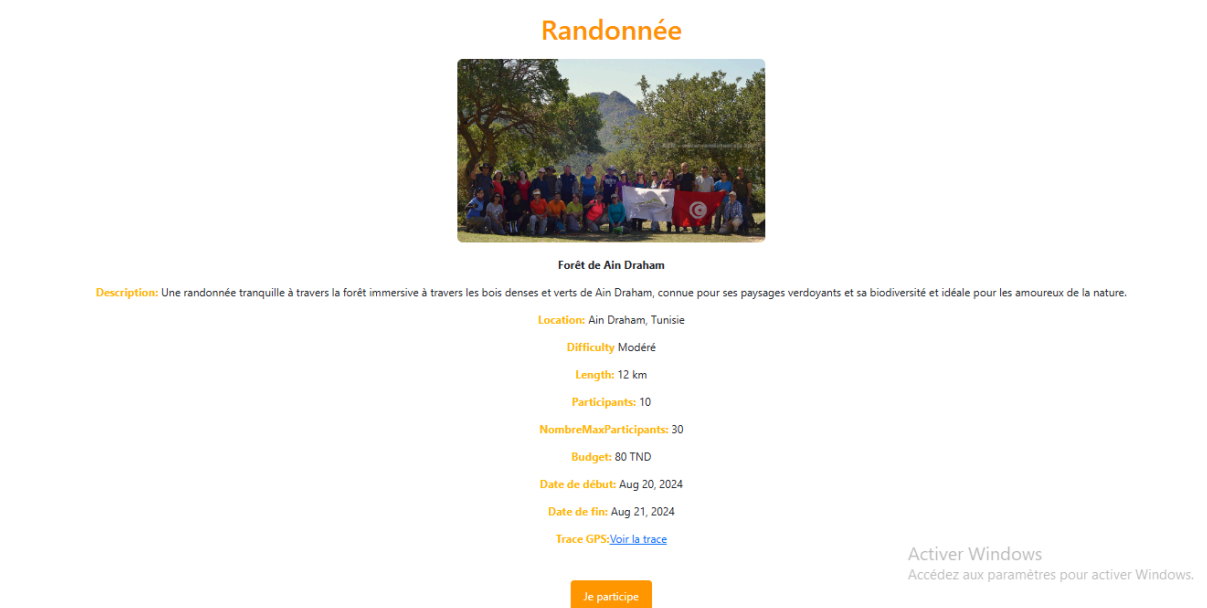
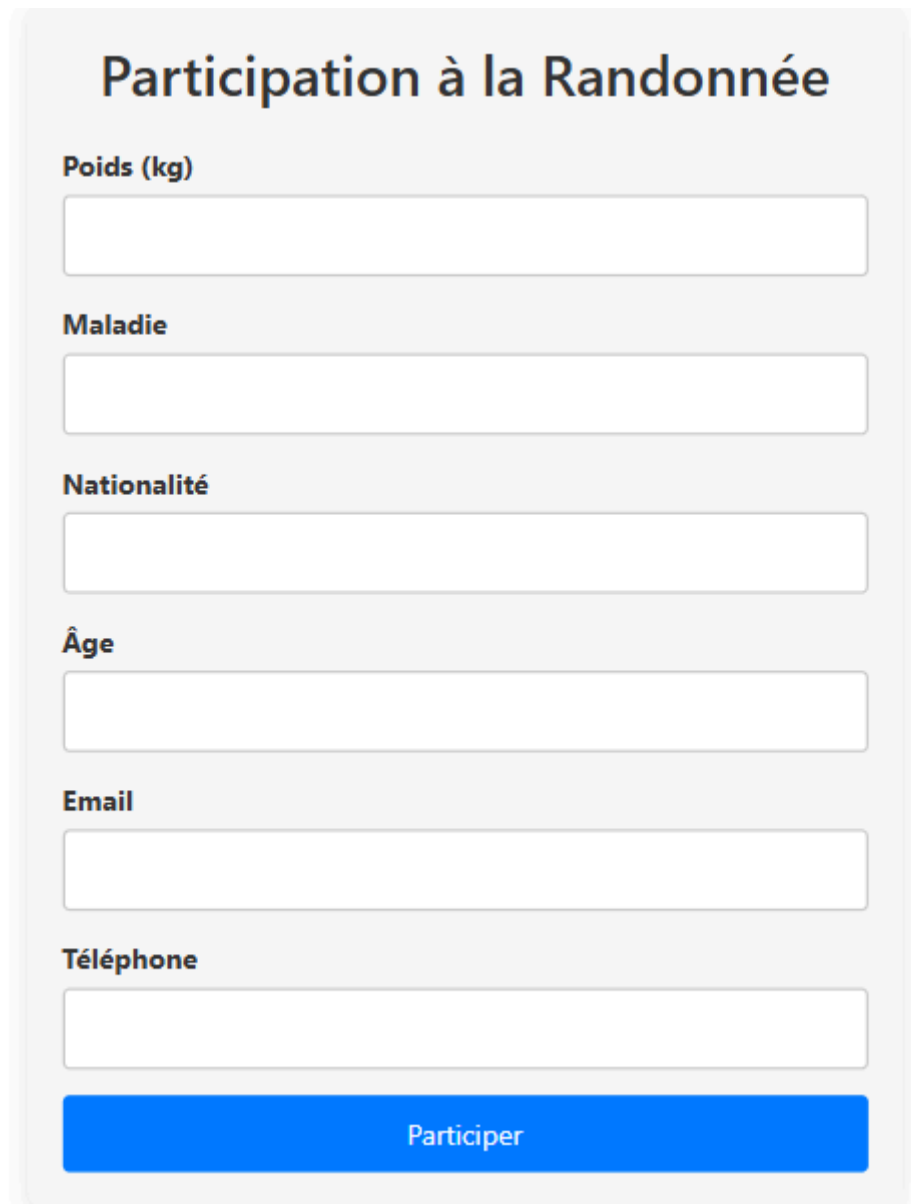


Figure 4.11 – Interface d'une Randonnée [N15]



Participation à la Randonnée

Poids (kg)

Maladie

Nationalité

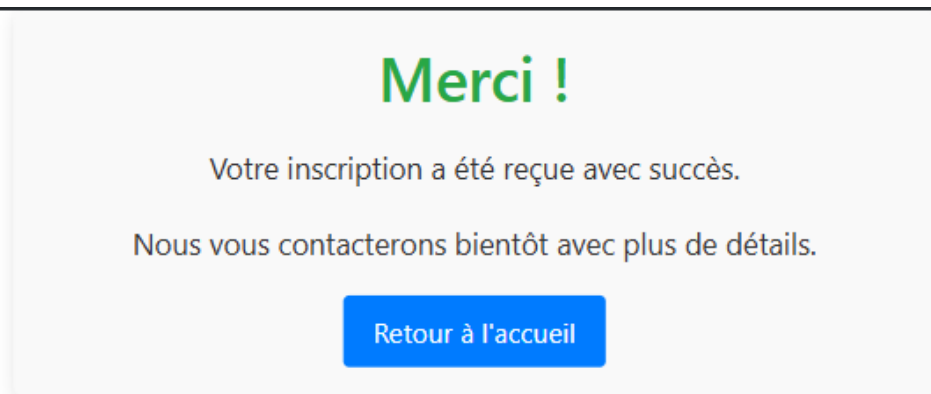
Âge

Email

Téléphone

Participer

Figure 4.12 – Interface de Participation [N16]



Merci !

Votre inscription a été reçue avec succès.

Nous vous contacterons bientôt avec plus de détails.

Retour à l'accueil

Figure 4.13 – Interface de Remerciement [N17]

Laissez votre feedback pour la randonnée

Votre Feedback :

Entrez votre feedback ici...

Envoyer

Figure 4.14 – Interface de Feedback [N18]

Admin Dashboard

Ajouter Randonnée

Modifier Randonnée

Supprimer Randonnée

Figure 4.15 – Interface de gestion des randonnées [N19]

Ajouter une Randonnée

Nom:

Description:

Lieu:

Difficulté:

Longueur (km):

Date de début:

jj/mm/aaaa



Date de fin:

jj/mm/aaaa



Nombre maximum de participants:

GPS Track:

Budget:

Ajouter Randonnée

Activer Windows

Figure 4.16 – Interface d'ajoute d'une randonnée [N20]

Modifier une Randonnée

Nom:

Description:

Location:

Difficulty:

Length:

Figure 4.17 – Interface de modification de randonnée [N21]

Supprimer Randonnée

ID de la Randonnée à Supprimer :

Figure 4.18– Interface de suppression de randonnée [N22]

III. Conclusion

Dans ce chapitre final , nous avons décrit l'environnement matériel et logiciel utilisé dans la réalisation de ce projet , et nous avons terminé par des captures sur les différents interfaces et fonctionnalité du l'application Réalisé.

Conclusion générale et perspective

RandoTrack est une application web conçue pour simplifier et améliorer la gestion des activités de randonnée. Grâce à une interface bien organisée et conviviale, les utilisateurs peuvent explorer divers parcours de randonnée, consulter des informations détaillées, participer à des randonnées, et soumettre leurs retours d'expérience via un formulaire de feedback. Le frontend de l'application a été développé avec Angular, offrant une navigation fluide entre les différentes pages et fonctionnalités. Ce projet a principalement mis l'accent sur la partie frontend, avec des fonctionnalités permettant à l'administrateur d'ajouter, modifier, et supprimer des randonnées. RandoTrack vise à offrir une expérience utilisateur enrichissante grâce à un design attrayant intégrant des images et des descriptions qui capturent l'esprit de l'aventure en plein air.

Les prochaines étapes pour RandoTrack incluent l'intégration des fonctionnalités backend pour la gestion des randonnées. Cela comprend la création d'API permettant d'ajouter, de modifier et de supprimer des randonnées, tout en assurant la sécurité des actions de l'administrateur et le stockage fiable des données dans une base de données.

Une autre perspective importante est de fournir à l'administrateur des outils pour analyser les feedbacks des participants. Par exemple, l'implémentation d'un tableau de bord interactif pourrait offrir une vue d'ensemble des retours, permettant ainsi d'identifier les tendances et les domaines à améliorer pour enrichir l'expérience des utilisateurs.

Enfin, bien que des fonctionnalités telles que la gestion avancée des inscriptions, le calendrier interactif, et la géolocalisation des randonnées soient déjà en place, il reste essentiel de continuer à affiner ces outils et à explorer de nouvelles façons d'optimiser l'expérience utilisateur et l'efficacité administrative.