



RÉPUBLIQUE TUNISIENNE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE
Direction Générale des Études Technologiques
Institut Supérieur des Études Technologiques de Jendouba
DÉPARTEMENT DES TECHNOLOGIES DE L'INFORMATIQUE

PROJET DE FIN D'ÉTUDES

Présenté en vue de l'obtention du
Diplôme de Licence Appliquée
En Technologies de l'Informatique
Spécialité : Développement des Systèmes d'Information

SUJET

Garde-robe virtuelle avec chatBot conseiller

Réalisé par :

Khalil DRIDI
Maram GUENOUATI

Sous la direction de :

Mme. Amina AMRAOUI Encadrante académique
Mr. Ahmed NEFFATI Encadrant professionnel



Dédicaces

C'est avec une profonde gratitude que je dédie ce travail.

À mes précieux parents, je souhaite exprimer ma reconnaissance pour l'amour inconditionnel dont ils m'ont entouré. Leur soutien indéfectible et leurs sacrifices ont façonné ma vie.

Je m'engage à être une source de fierté pour eux, à tout jamais reconnaissant pour tout ce qu'ils ont fait pour moi.

À tous les membres de ma famille, je témoigne de mon attachement et de ma profonde reconnaissance à travers ce travail.

À mes chers amis, je vous adresse mes remerciements pour votre encouragement et les moments inoubliables que nous avons partagés. Que notre amitié perdure, et que vous trouviez tout le bonheur et la réussite que vous méritez dans vos vies.

À mon binôme de travail, je tiens à exprimer ma gratitude pour notre collaboration fructueuse tout au long de ce projet.

Khalil DRIDI

Dédicaces

Avec une profonde gratitude, je dédie ce travail à mes chers parents, dont l'amour et le soutien inconditionnels ont toujours été un phare dans ma vie. Leur dévouement et leurs sacrifices sont une source d'inspiration constante, et je m'engage à être une source de fierté pour eux, en veillant à ne jamais les décevoir.

À toute ma famille, je souhaite exprimer ma sincère affection et ma gratitude pour votre soutien indéfectible. Que ce travail soit le témoignage de mon attachement et de ma reconnaissance envers vous tous.

À mon binôme de travail, je tiens à exprimer ma gratitude pour notre collaboration fructueuse tout au long de ce projet. Ta contribution a été inestimable. Ensemble, nous avons réalisé de grandes choses, et je suis reconnaissante pour ton engagement et ton soutien constant.

Que ce travail reflète ma gratitude envers ceux qui ont contribué à son accomplissement, et qu'il puisse leur apporter autant de satisfaction que j'en ai retirée lors de sa réalisation.

Maram GUENOUATI

Remerciements

Avant de présenter ce projet, nous tenons à exprimer nos sincères remerciements à tous ceux qui nous ont apporté leur aide et leur soutien tout au long de cette aventure.

Nous aimerais tout particulièrement exprimer notre profonde gratitude envers notre encadrante à l'ISET, Mme. Amina AMRAOUI, qui a accepté de nous superviser et de nous guider avec ses conseils précieux tout au long du projet.

Nos remerciements vont également à notre encadrant au sein de la société, M. Ahmed NEFFATI, pour son soutien et son assistance précieux durant la réalisation de ce projet.

Nous tenons également à exprimer nos sincères remerciements à tous les enseignants qui nous ont transmis leurs connaissances et qui nous ont soutenus dans notre parcours grâce à leur expertise.

Enfin, nous adressons nos remerciements aux membres du jury pour avoir accepté d'évaluer ce travail.

Table des matières

Introduction générale	1
1 Cadre général du projet et choix méthodologiques	3
Introduction	3
1.1 Présentation de Bee coders	3
1.2 Cadre général du travail	4
1.2.1 Problématique	4
1.2.2 Description du travail demandé	4
1.3 Étude de l'existant	4
1.3.1 Description de la situation existante	4
1.3.2 Critique de la situation existante	5
1.3.3 Solution suggérée	5
1.4 Choix méthodologique et formalisme adopté	5
1.4.1 Étude comparative entre les méthodes classiques et les méthodes agiles	5
1.4.1.1 Pourquoi Scrum ?	6
1.4.1.2 Le Framework Scrum	6
1.4.1.3 L'Équipe Scrum :	7
1.4.2 Formalisme adopté	8
Conclusion	8
2 Planification et architecture	9
Introduction	9
2.1 Identification des acteurs	9
2.2 Description des besoins	10
2.2.1 Besoins fonctionnels	10
2.2.2 Besoins non fonctionnels	10
2.3 Détails fonctionnels :	10
2.3.1 Diagramme de cas d'utilisation générale :	10
2.3.2 Diagramme de classe	11
2.4 Prototypage d'interfaces	12
2.5 Architecture du système	16
2.6 Mise en œuvre	17
2.6.1 Product Backlog	17
2.6.2 Planification des sprints	18
2.7 Environnement de développement	19
2.7.1 Environnement matériel	19
2.7.2 Environnement logiciel	19
2.8 Langage de programmation	21
Conclusion	22

3 Techniques d'intelligence artificielle	23
Introduction	23
3.1 L'intelligence artificielle	23
3.2 L'apprentissage automatique (Machine learning)	24
3.2.1 Les types d'apprentissage automatique	24
3.2.2 les principaux algorithmes de Machine Learning	25
3.3 L'apprentissage profond (Deep Learning)	26
3.3.1 Les type d'apprentissage profond	27
3.3.2 Comment fonctionne le deep learning ?	27
3.3.3 Les algorithmes de deep learning(L'apprentissage profond)	28
3.4 Traitement du langage naturel (NLP)	29
3.4.1 l'importance de le traitement du langage naturel	30
3.4.2 Comment fonctionne le traitement du langage naturel (NLP) ?	31
3.4.3 Les applications du traitement du langage naturel	31
Conclusion	32
4 Release 1 : Gestion des Comptes Utilisateurs et Administration	33
Introduction	33
4.1 Organisation des sprint de Release 1	33
4.2 Sprint 1	33
4.2.1 But du sprint	33
4.2.2 Sprint Backlog	33
4.2.3 Spécification fonctionnelle	34
4.2.4 Conception	36
4.2.5 Réalisation	38
4.2.6 Validation des Fonctionnalités du Sprint 1	42
4.3 Sprint 2	43
4.3.1 But du sprint	43
4.3.2 Sprint Backlog	43
4.3.3 Spécification fonctionnelle	43
4.3.4 Conception	44
4.3.5 Réalisation	45
4.3.6 Validation des Fonctionnalités du Sprint 2	47
Conclusion	48
5 Release 2 : Garde-Robe Virtuelle et Assistance ChatBot	49
Introduction	49
5.1 Organisation des sprint de Release 2	49
5.2 Sprint 3	49
5.2.1 But de ce sprint	49
5.2.2 Sprint Backlog	49
5.2.3 Spécification fonctionnelle	49
5.2.4 Conception	55
5.2.5 Consomation API météo	59
5.2.6 Réalisation	59
5.2.7 Validation des Fonctionnalités du Sprint 3	60
5.3 Sprint 4	61
5.3.1 But de sprint	61
5.3.2 Collecte de données	61
5.3.3 Bibliothèques utilisées	62
5.3.4 Tokenisation (Tokenization)	64
5.3.4.1 Importance de la tokenisation	64

5.3.4.2	Exemple de tokenisation	64
5.3.4.3	Gérer nos données avec la tokenisation	64
5.3.5	Lemmatisation (Lemmatization)	65
5.3.5.1	Sauvegarde des mots et des classes lemmatisés	65
5.3.5.2	Brassage des données(Shuffling)	66
5.3.6	Entraînement de modèle (Model training)	67
5.3.7	Postman : Test de l'API	68
5.3.8	Réalisation	68
Conclusion		68
Conclusion générale		70
Références		70
Annexe		73

Table des figures

1.1	Logo de la société Bee coders [1]	3
1.2	Publication en TikTok	4
1.3	Différence entre les méthodes classiques et les méthodes agiles[2]	6
1.4	Cycle de vie de la méthode Scrum[3]	7
2.1	Diagramme de cas d'utilisation générale	11
2.2	Diagramme de classe générale	12
2.3	Page d'accueil	13
2.4	Profil	14
2.5	Interface de la page Méteo	14
2.6	Interface de la page authentification Admin	15
2.7	Dashboard Admin	15
2.8	Logo application web	16
2.9	MVC [4]	16
2.10	Planification des sprints	19
3.1	Les différentes branches de l'intelligence artificielle [5]	23
3.2	L'apprentissage automatique [6]	24
3.3	L'intelligence artificielle et ses sous-domaines [7]	26
3.4	Deep Learning [8]	26
3.5	Fonctionnalité de deep Learning [9]	28
3.6	L'intersection de la NLP, du machine learning, du deep learning et de l'intelligence artificielle.[10]	30
3.7	l'importance de NLP [11]	30
3.8	Les applications NLP [12]	31
4.1	Diagramme de cas d'utilisation Sprint 1	34
4.2	cas d'utilisation Gérer profil	34
4.3	Diagramme de séquence «Gérer profil : Modifier »	36
4.4	Diagramme de séquence «Gérer profil : Supprimer»	37
4.5	Page d'accueil	38
4.6	Page d'accueil	38
4.7	Page d'accueil	39
4.8	Page d'accueil	39
4.9	Authentification	40
4.10	S'inscrire	40
4.11	Profil utilisateur	41
4.12	Tendance	41
4.13	Raffinement du cas d'utilisation du Sprint 2	44
4.14	Diagramme de séquence «Supprimer la publicité»	45
4.15	Authentification Admin	46
4.16	Dashboard Admin : Liste des utilisateurs	46
4.17	Dashboard Admin :les publications des tendances	47

5.1	Diagramme de cas d'utilisation du Sprint 3	50
5.2	Cas d'utilisation gérer Garde-Robe	50
5.3	cas d'utilisation Planification événement	53
5.4	Diagramme de séquence «Modifier vêtement »	55
5.5	Diagramme de séquence «Modifier vêtement »	56
5.6	Diagramme de séquence «Rechercher vêtement»	57
5.7	Diagramme de séquence «Supprimer événement»	58
5.8	Garde-Robe	59
5.9	calendrier	59
5.10	Météo	60
5.11	Collecte de données [13]	61
5.12	Collecte de données	62
5.13	NumPy Logo	62
5.14	Pickle Logo	63
5.15	Keras Logo	63
5.16	TensorFlow Logo	63
5.17	Exemple de tokenisation	64
5.18	Exemple de tokenisation	65
5.19	Exemple de Lemmatization	65
5.20	Exemple de Lemmatization	66
5.21	Collecter de bag des mot	66
5.22	Brassage des données	67
5.23	Code de model training	67
5.24	Test de l'API	68
5.25	Chat Bot	68
5.26	Weather By Api Dojo	73
5.27	Weather By Api Dojo	74
5.28	Weather By Api Dojo	74
5.29	Weather By Api Dojo	75
5.30	Weather By Api Dojo	75
5.31	Weather By Api Dojo	76
5.32	Weather By Api Dojo	76
5.33	Weather By Api Dojo	77
5.34	Weather By Api Dojo	77
5.35	Weather By Api Dojo	78
5.36	Transform tools	78
5.37	Weather By Api Dojo	79
5.38	Weather By Api Dojo	81

Liste des tableaux

1.1	Analyse de solution existante	5
3.1	différence entre l'apprentissage supervisé et non supervisé	25
3.2	L'apprentissage semi-supervisé et par renforcement	27
4.1	Backlog du sprint 1	33
4.2	Description textuelle du Gérer profil	35
4.3	Description textuelle du Gérer profil : Supprimer	35
4.4	Validation du sprint 1	42
4.5	Backlog du Sprint 2	43
4.6	Description textuelle de cas de utilisation Supprimer la publication	44
4.7	Validation du sprint 2	47
5.1	Backlog du Sprint 3	49
5.2	Description textuelle de cas de utilisation ajouter vêtement	51
5.3	Description textuelle de cas de utilisation modifier vêtement	51
5.4	Description textuelle de cas de utilisation supprimer vêtement	52
5.5	Description textuelle de cas de utilisation rechercher vêtement	52
5.6	Description textuelle de cas d'utilisation ajouter événement	53
5.7	Description textuelle de cas d'utilisation supprimer événement	54
5.8	Validation du sprint 3	60

Introduction générale

À l'heure actuelle, les algorithmes d'apprentissage automatique connaissent une expansion remarquable, soutenue par des avancées scientifiques qui transforment de manière significative notre quotidien. L'Intelligence Artificielle (IA) permet désormais aux ordinateurs de réaliser des tâches autrefois jugées irréalisables. La combinaison de la puissance de calcul informatique et des capacités de l'IA a engendré une forme d'intelligence surpassant largement les capacités humaines.

Les récentes avancées technologiques ont engendré de nouvelles habitudes et pratiques dans notre société. Les chatbots alimentés par le traitement du langage naturel (NLP) se démarquent parmi ces innovations. Ces chatbots utilisent l'intelligence artificielle pour analyser et comprendre les requêtes des utilisateurs, leur permettant ainsi de fournir des réponses personnalisées et pertinentes. Leur montée en importance s'explique par leur capacité à améliorer l'expérience utilisateur, à influencer les décisions et à simplifier les tâches quotidiennes, illustrant ainsi l'impact significatif du NLP dans notre vie quotidienne.

Ainsi, les avantages des progrès technologiques deviennent de plus en plus évidents pour les utilisateurs, qui bénéficient d'une assistance intelligente et personnalisée dans divers aspects de leur vie.

C'est dans cet esprit que nous nous proposerons de créer une application web visant à aider les utilisateurs à constituer leur propre garde-robe sur leur ordinateur. Cette application facilitera le choix des vêtements au quotidien tout en permettant aux utilisateurs de visualiser l'ensemble des vêtements qu'ils posséderont. De plus, notre application permettra aux utilisateurs de consulter les conditions météorologiques afin d'adapter leurs tenues en conséquence. Elle offrira également la possibilité d'enregistrer des événements dans un calendrier personnel. Enfin, nous ajouterons chatbot permettant aux utilisateurs d'obtenir des conseils personnalisés lors du choix de leurs vêtements.

Ce rapport détaillera les différentes étapes de réalisation de notre application web. Il se composera de six chapitres organisés comme suit :

Le premier chapitre, intitulé "Cadre général du projet et choix méthodologiques", débutera par une étude critique de l'existant afin de déterminer le cadre global du projet. Nous proposerons ensuite des solutions aux problèmes identifiés. Enfin, nous présenterons en détail la méthode de conception que nous aurons choisie et le formalisme que nous aurons adopté.

Le deuxième chapitre, intitulé "Planification et architecture", définira les acteurs impliqués dans ce projet. Nous préciserons ensuite les besoins fonctionnels à l'aide de diagrammes de cas d'utilisation généraux. Nous spécifierons également les besoins fonctionnels et non fonctionnels de notre application , nous présenterons quelques prototypes d'interfaces et établirons le planning des sprints Enfin, nous présenterons les différents outils matériels et logiciels utilisés lors du développement de notre application.

Le troisième chapitre, intitulé "Techniques d'intelligence artificielle", sera consacré à la présentation de l'intelligence artificielle, du machine learning et du deep learning, ainsi que de leurs points forts respectifs et de leurs algorithmes.

Le quatrième chapitre, intitulé "Release 1 : Gestion des Comptes Utilisateurs et Administration", se concentrera sur l'analyse et la conception des sprints 1 et 2.

Le cinquième chapitre, intitulé "Release 2 Garde-Robe Virtuelle et Assistance Chat-Bot", abordera l'analyse et la conception des sprints 3 et 4.

Nous conclurons ensuite par une conclusion générale résumant l'ensemble des travaux et exposant quelques perspectives.

Chapitre 1

Cadre général du projet et choix méthodologiques

Introduction

À ce début de chapitre, notre intention première est d'explorer le contexte global du projet. Nous commencerons par présenter l'organisme d'accueil, suivie de l'exposition de la problématique, de la solution proposée, et du choix de la méthode de conception. Enfin, nous clôturerons ce chapitre par une conclusion.

1.1 Présentation de Bee coders

Bee Coders, fondée en 2020 en Tunisie, est une agence spécialisée dans le développement de produits numériques, tels que les sites internet, les applications mobiles (pour Android et iOS), ainsi que le design web.

Ses activités principales incluent le développement web et mobile, le conseil en technologies de l'information, la formation en ligne sur mesure et les services d'intermédiation personnalisés. [1]



FIGURE 1.1 – Logo de la société Bee coders [1]

1.2 Cadre général du travail

1.2.1 Problématique

Le processus de sélection des vêtements dans notre vie quotidienne est souvent fastidieux et manque de conseils, ce qui peut conduire à des choix peu satisfaisants.

1.2.2 Description du travail demandé

Le projet vise à concevoir et développer une garde-robe virtuelle assortie d'un chatbot conseiller.

Notre application propose plusieurs fonctionnalités à l'utilisateur. Ce projet est réalisé dans le cadre de notre stage de fin d'études afin d'obtenir le diplôme de Licence Appliquée en Technologies de l'Informatique de l'Institut Supérieur des Études Technologiques de Jendouba.

1.3 Étude de l'existant

Cette section a pour objectif de passer en revue les applications web de garde-robe en ligne existantes qui aident les utilisateurs dans leurs choix vestimentaires sur le marché.

Cette étude présente d'abord une description de la situation actuelle, puis une critique des solutions existantes, et enfin, les solutions suggérées.

1.3.1 Description de la situation existante

Dans le paysage actuel des applications web de mode, il est frappant de constater l'absence d'outils dédiés à la gestion des garde-robés en ligne, particulièrement en Tunisie. Les utilisateurs exploitent divers outils, notamment les réseaux sociaux comme Facebook, Instagram et TikTok, pour partager leurs tenues avec leurs amis et recevoir des conseils et des avis. Bien que cette pratique aide les utilisateurs à faire de meilleurs choix vestimentaires, il n'existe pas d'application en Tunisie permettant à la fois d'organiser leur garde-robe et de recevoir des recommandations personnalisées.

Cette figure illustre un homme qui publie sa tenue sur TikTok et demande des conseils aux autres membres de la communauté.



FIGURE 1.2 – Publication en TikTok

1.3.2 Critique de la situation existante

Après analyse de la situation existante, on remarque que les réseaux sociaux ne répondent pas entièrement aux besoins des utilisateurs.

Outils	Avantages	Inconvénients
Les réseaux sociaux	Interaction sociale Inspiration de mode Avis diversifiés	Pression sociale et négativité Manque de confidentialité Manque de vie privée Absence d'organisation

TABLE 1.1 – Analyse de solution existante

1.3.3 Solution suggérée

Après une analyse approfondie de la situation actuelle, nous avons identifié divers défis. En nous focalisant sur ces enjeux, nous avons élaboré une nouvelle proposition d'application web en introduisant des solutions novatrices et modernes pour répondre aux besoins émergents.

Notre site offre aux utilisateurs la possibilité de :

- Établir une garde-robe virtuelle pour leurs vêtements.
- Un modèle IA formé pour donner des conseils sur la couleur des tenues.
- Enregistrer les vêtements dans des catégories.
- Consulter les prévisions météorologiques pour choisir leurs vêtements.
- Créer un calendrier pour leurs événements et rendez-vous.
- Consulter les publications des tendances.
- Gérer leur profil .

1.4 Choix méthodologique et formalisme adopté

Le processus de choix d'une méthodologie de projet est une étape cruciale , influencée par la nature et la taille du projet.

Elle met l'accent sur : ce qu'il faut faire, quand le faire, comment le faire et pourquoi le faire pour contribue à la réalisation du projet de manière efficace et réussie .

Et aujourd'hui , le mot Agile peut faire référence à ces valeurs .

1.4.1 Étude comparative entre les méthodes classiques et les méthodes agiles

Le modèle classique de Cycle en V et en Cascades se découpe en plusieurs phases identifiées dès le démarrage du projet : Définition des besoins, conception générale puis détaillée, réalisation et développement, tests et mise en production la validation d'une phase entraîne le début de la suivante.

Cette méthode limite les retours aux étapes précédentes. Le Métier définit un objectif à long terme.

Et De l'autre côté les méthodes Agiles (dont les plus utilisées sont aujourd'hui les méthodes Scrum, Kanban et Extrem Programming) procèdent par étape avec des objectifs à court terme. Le client, garant du produit (Product Owner), s'est placé au centre des démarches.

Le besoin métier est découpé en User Stories développées de manière itérative. Son évolution est encouragée.

Les projets sont découpés en sprints (jusqu'à un mois de durée), qui commencent par une vérification de la planification opérationnelle et se terminent par une démonstration de ce qui a été achevé, puis une rétrospective.

La figure ci-dessous représente la différence entre les méthodes classiques et les méthodes agiles :

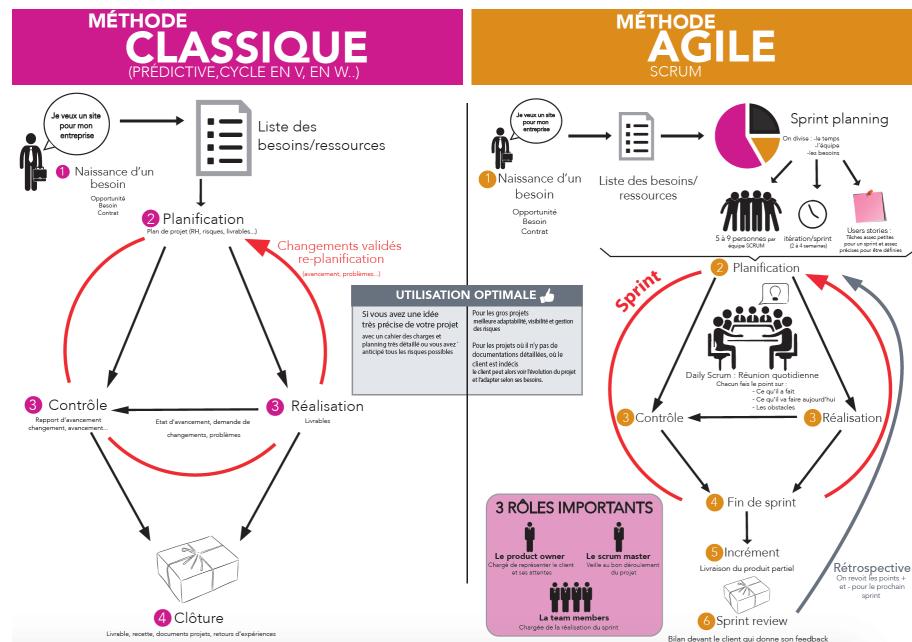


FIGURE 1.3 – Différence entre les méthodes classiques et les méthodes agiles[2]

1.4.1.1 Pourquoi Scrum ?

Nous avons choisi Scrum comme un cadre de travail de notre projet. Scrum se distingue par sa légèreté et sa simplicité conceptuelle, et répondre à des problèmes complexes et changeants, tout en livrant de manière productive et créative des produits de la plus grande valeur possible.

Il n'est pas en soi un processus ni une méthode de développement de produits ; c'est un canevas pour l'application de divers procédés et techniques de développement.

Scrum met en évidence l'efficacité relative des pratiques de gestion et de développement de produit en place, de sorte que ces dernières puissent être améliorées.

Il se caractérise par :

- Plus de souplesse et de réactivité
- La grande capacité d'adaptation au changement grâce à des itérations courtes
- Et la chose la plus importante, c'est que Scrum rassemble les deux cotés théorique et pratique et se rapproche beaucoup de la réalité.

1.4.1.2 Le Framework Scrum

Scrum représente un cadre de gestion de projet Agile, offrant aux équipes une structure et une méthode pour organiser et gérer leur travail, reposant sur un ensemble de valeurs, de principes et de pratiques.

Scrum se base sur la théorie du contrôle empirique de processus, ou l'empirisme.

Scrum utilise une approche itérative et incrémentale pour optimiser la prédictibilité et pour contrôler le risque.

Trois piliers soutiennent l'implémentation d'un contrôle empirique de processus : la transparence, l'inspection et l'adaptation.[14]

Scrum est basé sur :

- Dégager dans un premier lieu le maximum des fonctionnalités à réaliser pour former le backlog du produit.
- En second lieu, définir les priorités des fonctionnalités et choisir lesquelles seront réalisées dans chaque itération.
- Par la suite, focaliser l'équipe de façon itérative sur l'ensemble de fonctionnalités à réaliser, dans des itérations appelées Sprints.
- Un Sprint aboutit toujours à la livraison d'un produit partiel fonctionnel appelé incrément.

La figure ci-dessous représente cycle de vie de la méthode Scrum :

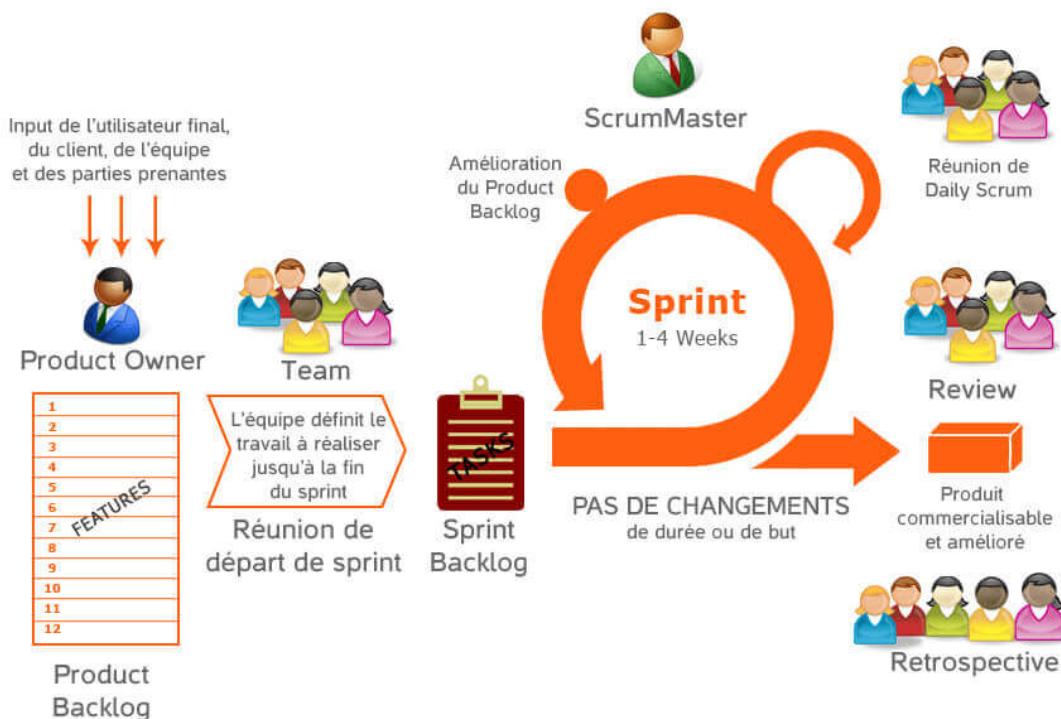


FIGURE 1.4 – Cycle de vie de la méthode Scrum[3]

1.4.1.3 L'Équipe Scrum :

L'Équipe Scrum comprend un propriétaire de produit (Product Owner), une équipe de développement (Development Team) et un Scrum Master.

Scrum définit un modèle d'équipe optimisant la flexibilité, la créativité et la productivité.

Le Product Owner :

- Le Product Owner est responsable de maximiser la valeur du produit et du travail de l'Équipe de Développement.

- Le Product Owner est la seule personne responsable de gérer le carnet de produit (Product Backlog).

L'Équipe de Développement :

- L'Équipe de Développement est constituée de professionnels qui livrent à chaque Sprint un incrément « terminé » et potentiellement livrable du produit
- Les équipes de développement sont structurées et habilitées par l'entreprise à organiser et gérer leur propre travail

Le Scrum Master :

- Le Scrum Master est responsable de s'assurer que Scrum est compris et mis en œuvre.
- Les Scrum Masters remplissent leur rôle en s'assurant que l'Équipe Scrum adhère à la théorie, aux pratiques et aux règles de Scrum(un leader au service de l'Équipe Scrum).[15]

1.4.2 Formalisme adopté

Après avoir opté pour une méthode, il est essentiel d'adopter un langage de modélisation unifié pour représenter le projet.

Dans le cadre de la conception du système, notre choix s'est porté sur UML en tant que langage de modélisation.

Cette décision repose sur les avantages inhérents à ce langage, notamment sa standardisation et la diversité des diagrammes qu'il propose.

De plus, les outils performants fournis par UML facilitent la simplification et la standardisation des systèmes complexes.

Conclusion

Ce chapitre constitue la première partie de notre projet, au cours de laquelle nous avons présenté le projet dans son contexte général en menant une étude de la situation existante, en la critiquant et en identifiant des solutions appropriées, en mettant l'accent sur la méthode de travail adoptée.

Chapitre 2

Planification et architecture

Introduction

Dans cette partie, nous décrirons l'architecture matérielle, l'environnement de développement et l'outil conceptuel utilisé pour développer notre application, ainsi que la planification des versions de travail.

2.1 Identification des acteurs

Un acteur peut être une personne, une organisation ou un système externe qui interagit avec votre application en envoyant des requêtes et en attendant un ou plusieurs services offerts par l'application. Par conséquent, nous identifions un acteur dans notre application web :

Utilisateur

L'utilisateur a accès à toutes les fonctionnalités de l'application web, y compris le chat avec le chatbot, la gestion de sa garde-robe, la fixation de rendez-vous, la consultation de la météo, etc.



Administrateur

Il est principalement chargé de gérer les comptes utilisateurs via un tableau de bord.



2.2 Description des besoins

2.2.1 Besoins fonctionnels

Utilisateur

- Inscription
- S'authentifier
- Gérer sa garde-robe
- Discuter avec chatbot
- Faire un calendrier personnel
- Consulter les publications des styles tendances
- Consulter la météo

Administrateur

- S'authentifier
- Gérer comptes utilisateurs
- Gérer les publications de style tendance

2.2.2 Besoins non fonctionnels

Pour compléter les besoins fonctionnels, notre projet devra respecter un ensemble de propriétés contribuant à une meilleure qualité de la solution obtenue. Parmi ces critères figurent :

- Performance : Il s'agit d'optimiser le temps de chargement des données ainsi qu'en utilisant les bonnes pratiques de développement,
- Portabilité : Doit être facile à utiliser et doit être accessible via des tablettes ou des téléphones.
- Sécurité : L'accès à site web et aux données doit être sécurisé
- Ergonomique : Puisque notre site sera principalement destiné aux utilisateurs, les interfaces doivent être conçues de manière à être conviviales.

2.3 Détails fonctionnels :

2.3.1 Diagramme de cas d'utilisation générale :

Le diagramme de cas d'utilisation suivant(Figure 2.1) met en évidence les interactions des principaux acteurs de notre système : utilisateur et administrateur. Il représente de manière concise les actions que chaque acteur peut entreprendre au sein de l'application, fournissant ainsi un aperçu des fonctionnalités du système.

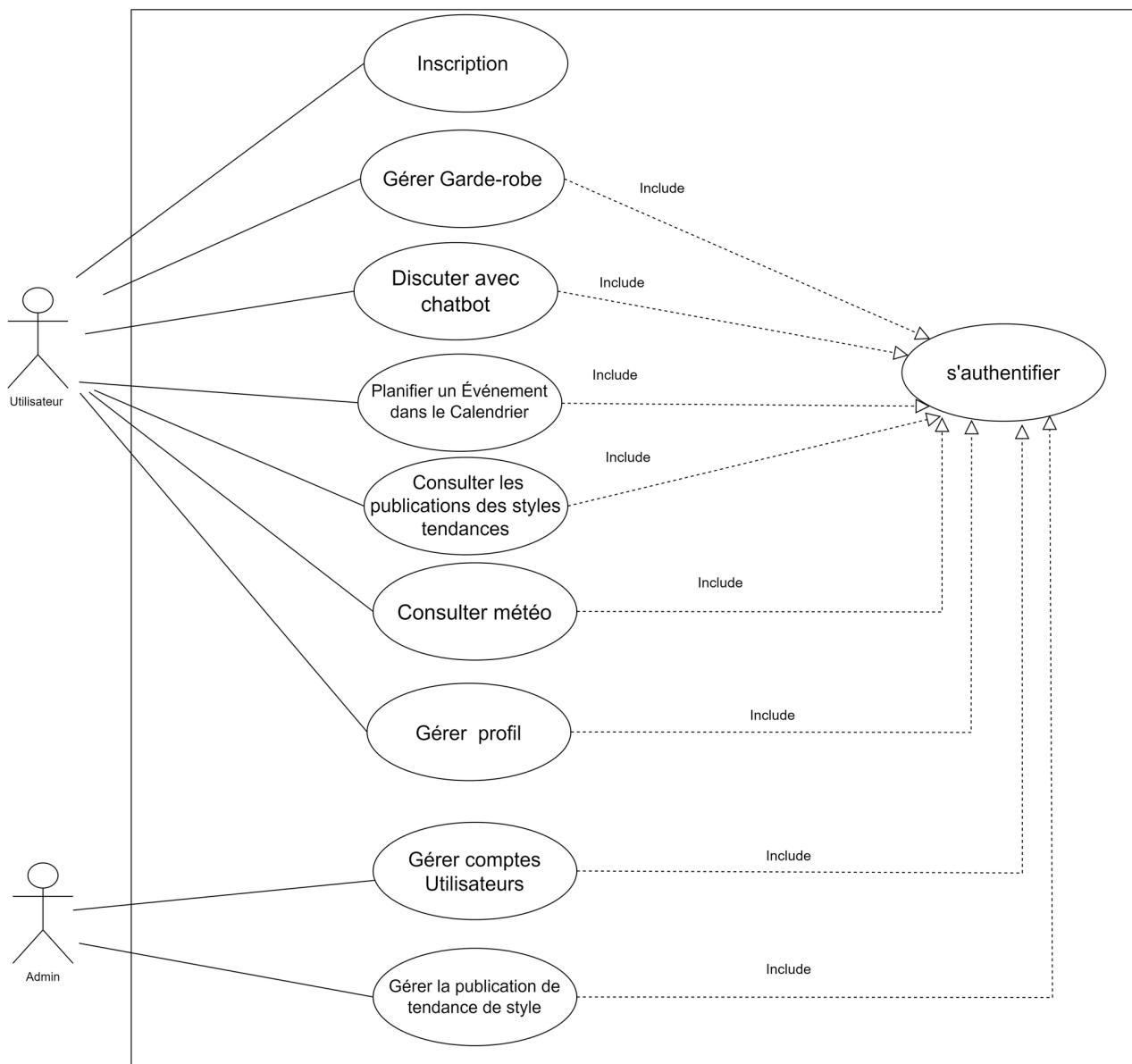


FIGURE 2.1 – Diagramme de cas d'utilisation générale

2.3.2 Diagramme de classe

Le diagramme de classes suivant (Figure 2.2) illustre la structure statique de notre système en mettant en évidence les différentes classes et leurs relations. Il offre une représentation concise des entités du système ainsi que de leurs attributs et méthodes, ce qui permet une compréhension claire de l'organisation interne de l'application.

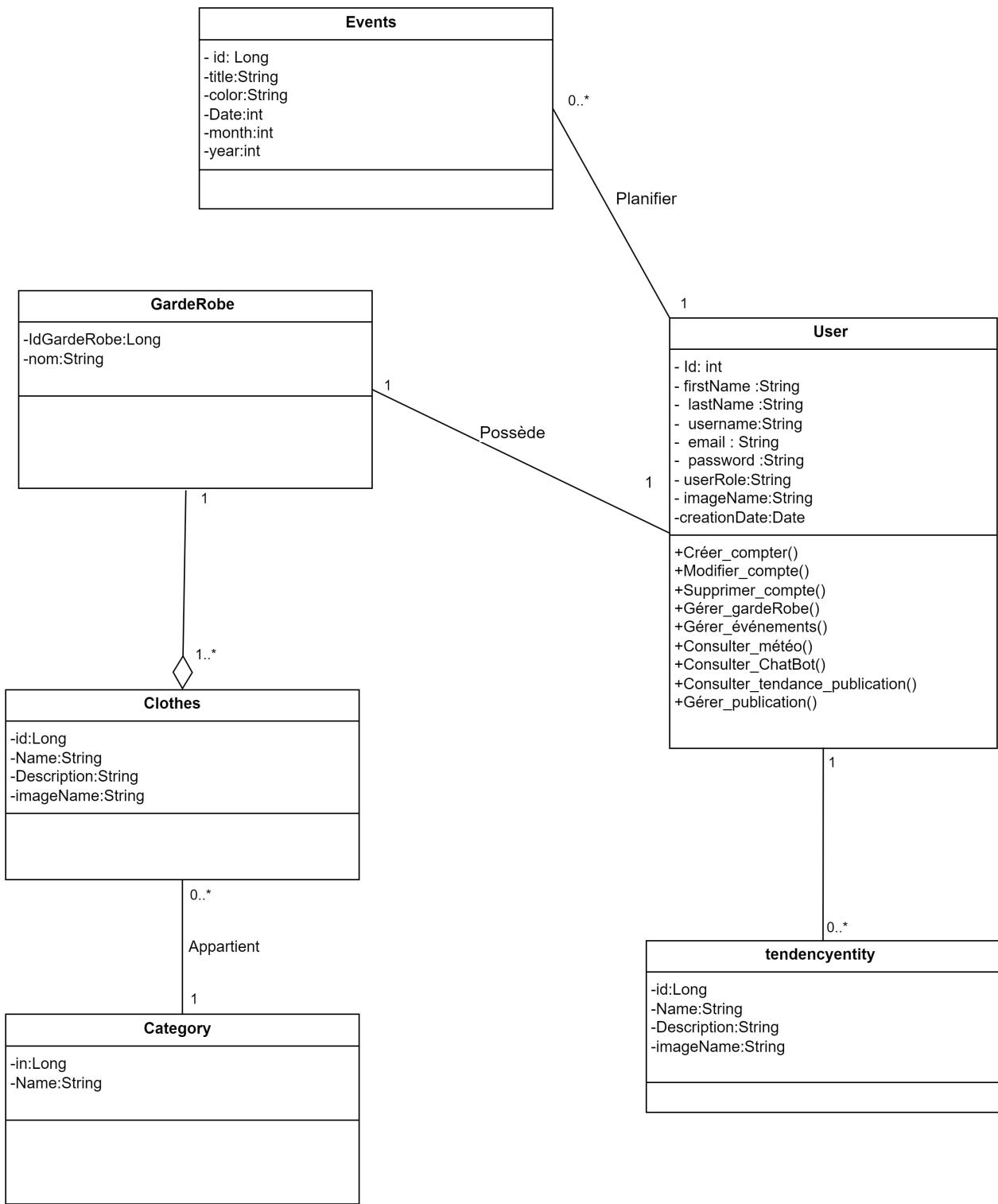


FIGURE 2.2 – Diagramme de classe générale

2.4 Prototypage d'interfaces

La modélisation est une étape essentielle, elle consiste à préparer les interfaces du futur produit en utilisant l'outil de modélisation Balsamiq. L'objectif principal de cette technique est de recueillir des informations sur l'interaction des utilisateurs avec le futur produit. Pour cela, nous présenterons quelques prototypes des interfaces de notre site web.

- **Page d'accueil**

Cette figure(Figure 2.3) présente l'interface de la page d'accueil :

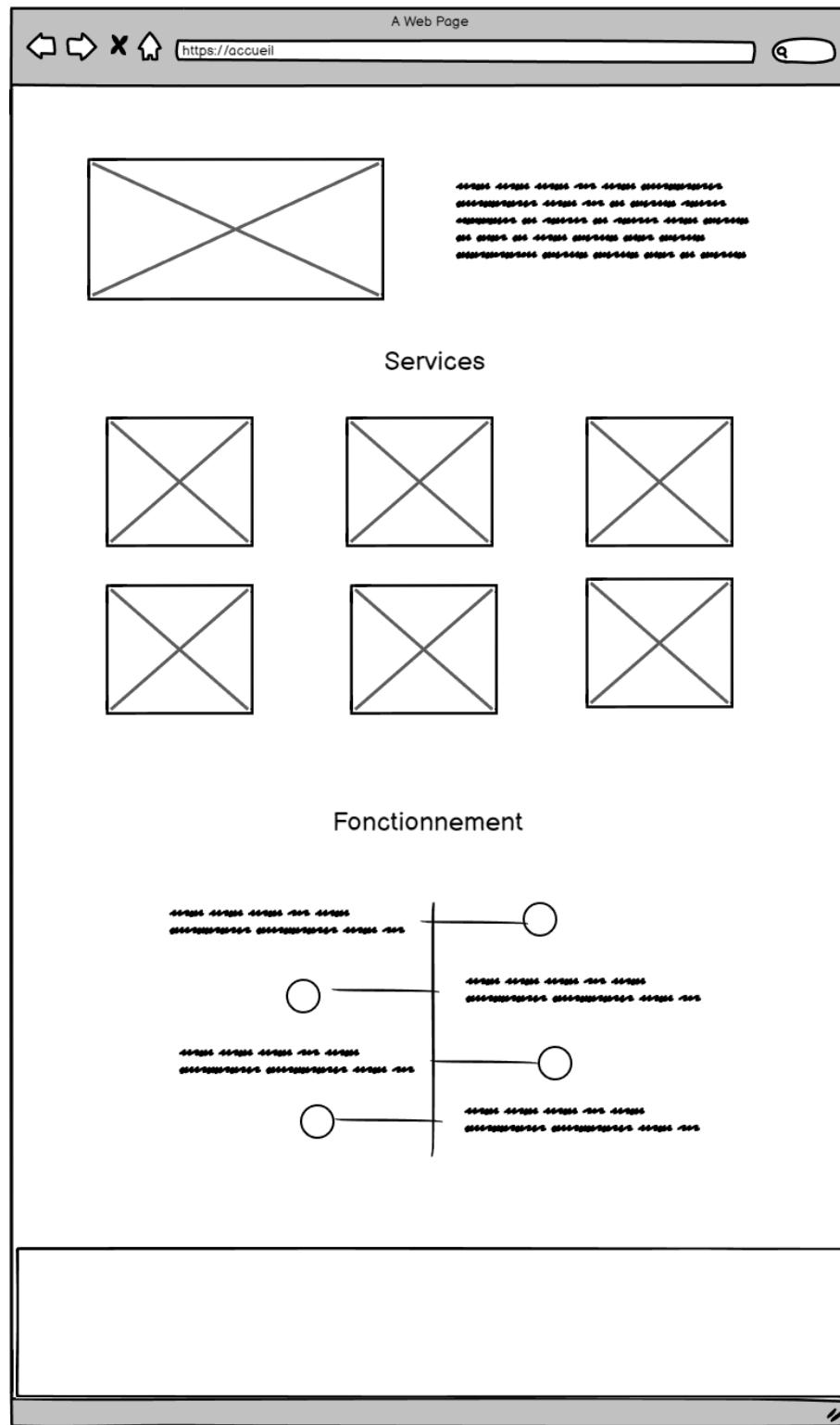


FIGURE 2.3 – Page d'accueil

- **Profile**

Cette figure(Figure 2.4) présente l'interface de la page Profil utilisateur :

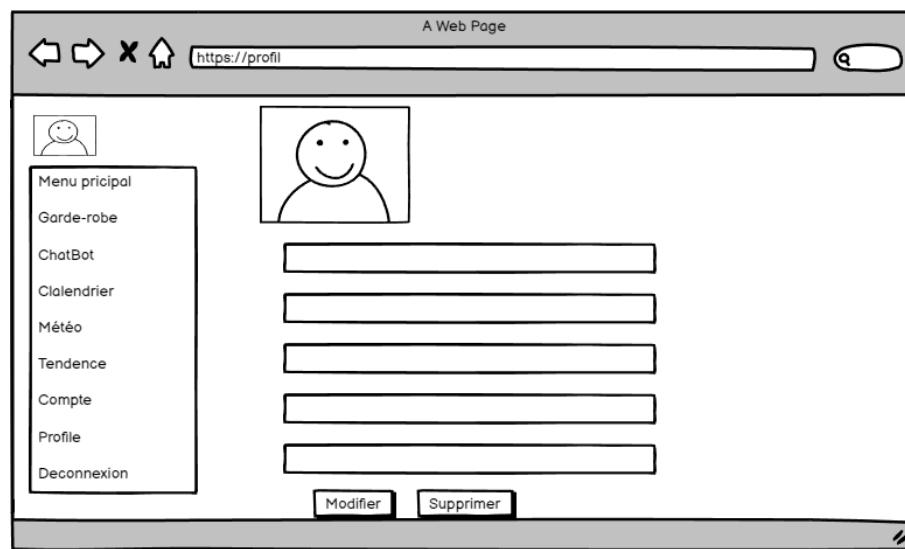


FIGURE 2.4 – Profil

- **Méteo**

Cette figure(Figure 2.6) présente l'interface de la page météo :

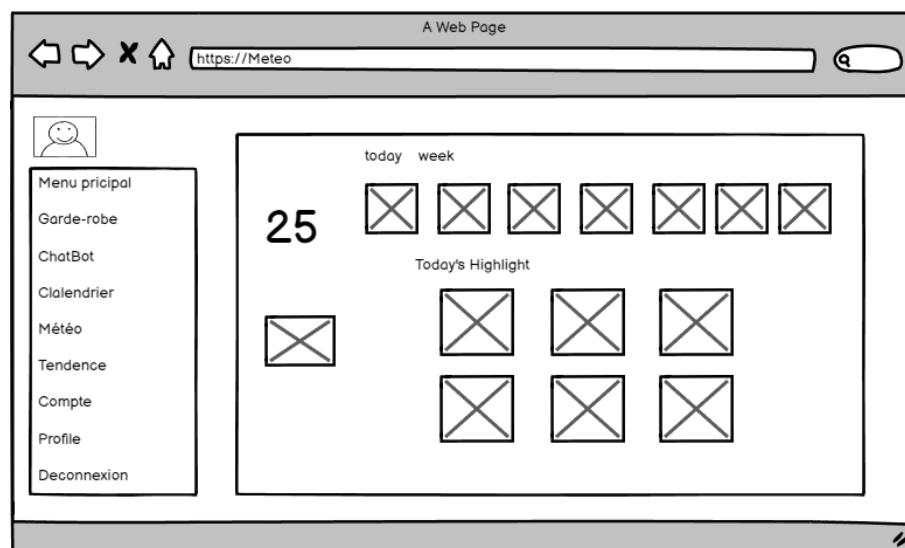


FIGURE 2.5 – Interface de la page Méteo

• Authentification Admin

Cette figure (Figure 2.7) présente l'interface de la page "Authentification Admin" où l'admin saisit son email et mot de passe.



FIGURE 2.6 – Interface de la page authentification Admin

• Dashboard Admin

Cette figure montre l'interface du tableau de bord d'administration

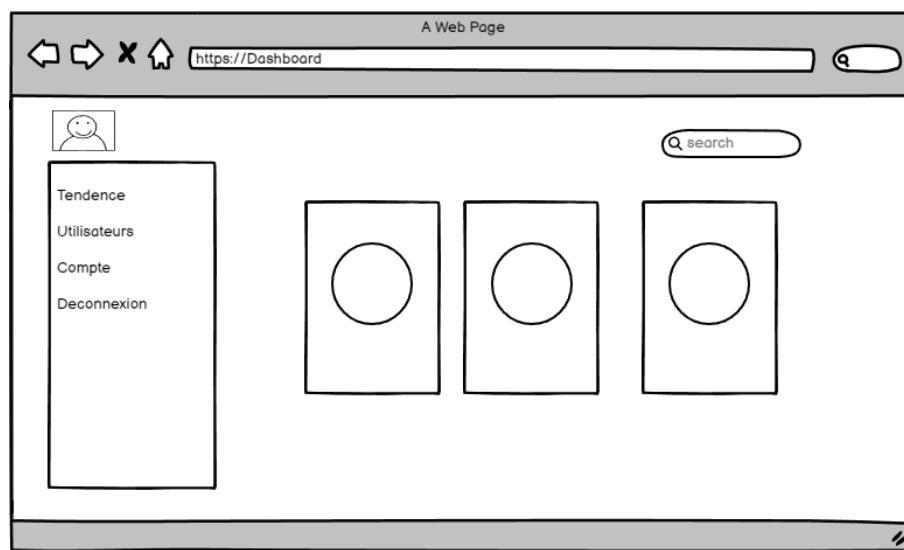


FIGURE 2.7 – Dashboard Admin

- **Logo application web**

Cette figure montre le logo de notre application web



FIGURE 2.8 – Logo application web

2.5 Architecture du système

Architecture logiciel

Modèle-Vue-Contrôleur

Le Modèle-Vue-Contrôleur (MVC) est un motif d'architecture logicielle largement utilisé pour les interfaces graphiques, en particulier dans les applications web. Il a été introduit en 1978 et est depuis devenu extrêmement populaire.

Le MVC se compose de trois types de modules distincts, chacun ayant ses propres responsabilités : les modèles, les vues et les contrôleurs.

Un modèle (Model) contient les données à afficher.

Une vue (View) représente l'aspect visuel de l'interface graphique.

Un contrôleur (Controller) gère la logique liée aux actions effectuées par l'utilisateur.[16]

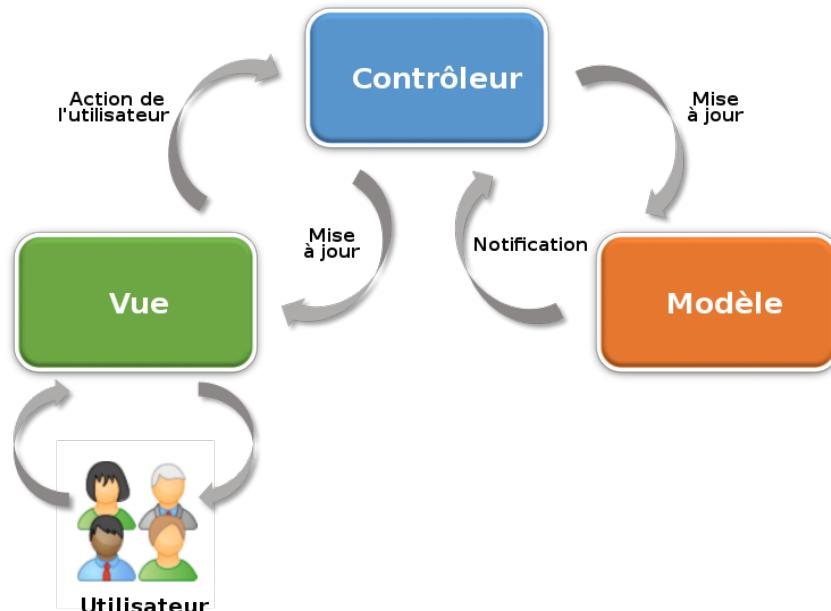


FIGURE 2.9 – MVC [4]

2.6 Mise en œuvre

2.6.1 Product Backlog

Le tableau ci-dessous résume le backlog produit de notre application. Il se caractérise par :

- Une colonne Feature : pour mieux ordonner les user stories comme par exemple S'inscrire, Authentifier...
- Une colonne User Story : qui représente la description des user stories "En tant que... je veux..."
- Une colonne Priorité : qui représente la priorité de la user story.

Feature	User Story	Priorité
S'inscrire	En tant qu'utilisateur, je veux m'inscrire afin d'accéder à l'application web.	Élevée
S'authentifier	En tant qu'utilisateur, je veux m'authentifier pour accéder à mon profil. En tant qu'admin, je veux m'authentifier pour accéder à mon dashboard.	Élevée
Gérer profil	En tant qu'utilisateur, je veux modifier mon profil. En tant qu'utilisateur, je veux supprimer mon profil. En tant qu'utilisateur, je veux consulter mon profil.	Moyenne
Consulter les publications des styles tendances	En tant qu'utilisateur, je veux consulter les publications des styles tendances.	Moyenne
Gérer comptes utilisateurs	En tant qu'admin, je veux consulter la liste des comptes utilisateurs. En tant qu'admin, je veux supprimer les comptes utilisateurs. En tant qu'admin, je veux rechercher des comptes utilisateurs.	Moyenne

Feature	User Story	Priorité
Gérer la publication de tendance de style	En tant qu'admin, je veux ajouter une publication de style tendance. En tant qu'admin, je veux supprimer une publication de style tendance. En tant qu'admin, je veux modifier une publication de style tendance.	Moyenne
Gérer garde-robe	En tant qu'utilisateur, je veux ajouter un vêtement. En tant qu'utilisateur, je veux modifier un vêtement. En tant qu'utilisateur, je veux supprimer un vêtement. En tant qu'utilisateur, je veux rechercher un vêtement.	Élevée
Consulter météo	En tant qu'utilisateur, je veux consulter la météo.	Moyenne
Planifier le calendrier	En tant qu'utilisateur, je veux ajouter un événement. En tant qu'utilisateur, je veux supprimer un événement. En tant qu'utilisateur, je veux consulter mon calendrier.	Moyenne
ChatBot	En tant qu'utilisateur, je veux être capable de discuter avec un ChatBot.	Élevée

2.6.2 Planification des sprints

Les "User stories" définies dans le Product Backlog sont triées par ordre de priorité et de valeur commerciale. L'objectif est de mettre en œuvre en premier ce qui a le plus de valeur. Les tâches seront planifiées selon les sprints que nous avons définis.

Dans le figure suivant, nous présentons la planification des sprints.

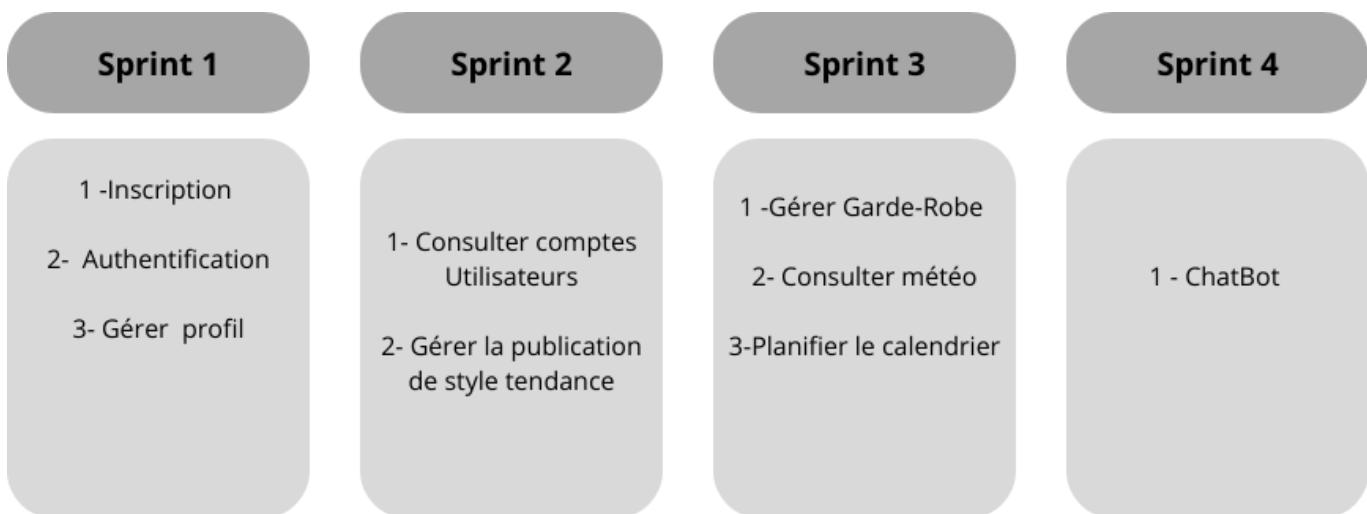


FIGURE 2.10 – Planification des sprints

2.7 Environnement de développement

2.7.1 Environnement matériel

Pour développer l'application, nous avons utilisé comme environnement matériel 2 ordinateurs portables chacun dotés du caractéristiques suivantes :

MSI thin GF63

RAM 8Go
Processeur core i 7 12 génération
Nvidia geforce rtx2050
Disque dur 1 to
Windows 11

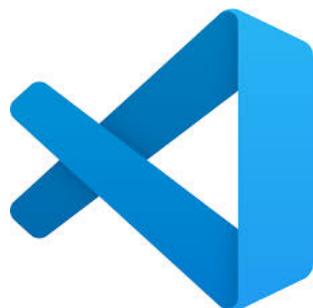
HP 255 G8

RAM 8Go
Processeur Processeur AMD 3020e with Radeon Graphics, 1200 MHz
Système d'exploitation 64 bits, processeur x64
Microsoft Windows 11 Famille
Taille de l'écran 15,6 Pouces

2.7.2 Environnement logiciel

Visual studio code

Visual Studio Code est un éditeur de code source gratuit créé par Microsoft pour Windows, Linux et macOS. Il offre de nombreuses fonctionnalités, telles que le débogage, la coloration syntaxique, la complétion de code intelligente, les extraits de code, la refactorisation de code et l'intégration de Git.[17]



IntelliJ IDEA

est un environnement de développement destiné à la création de logiciels informatiques basés sur la technologie Java. [18]



PyCharm

PyCharm est un environnement de développement intégré utilisé pour programmer en Python 2. Il offre une analyse de code, un débogueur graphique, la gestion des tests unitaires, l'intégration de logiciel de gestion de versions, et le support du développement web avec Django. [19]



Draw IO

draw.io est une application de création de diagrammes et de schémas disponible sous licence Apache. Elle peut être utilisée sur Windows, macOS, Linux ainsi que sous forme d'application web.[20]



Balsamiq

Balsamiq Wireframes est un outil rapide de création de maquettes d'interface utilisateur à faible fidélité qui reproduit l'expérience du dessin sur un bloc-notes ou un tableau blanc, mais sur un ordinateur. Cela t'encourage vraiment à te concentrer sur la structure et le contenu, en évitant les discussions prolongées sur les couleurs et les détails qui devraient intervenir ultérieurement dans le processus. [21]



Overleaf

Overleaf est une plateforme en ligne pour l'édition LaTeX en temps réel sans configuration. Elle propose des modèles de documents tels que des journaux académiques et des devoirs, ainsi que des fonctionnalités avancées comme la collaboration, les gestionnaires de références et l'intégration avec GitHub. [22]



Postman

Vous pouvez choisir la méthode de la requête, entrer l'URL du serveur que vous voulez interroger, et rajouter tous les paramètres possibles pour une requête HTTP. Le logiciel tient un historique de vos requêtes. Il est très utile pour tester une api..



GitHub

est une plateforme de développement qui permet aux développeurs de créer, stocker, gérer et partager les code.



2.8 Langage de programmation

Angular



est un framework open source basé sur TypeScript, développé par l'équipe du projet "Angular" chez Google et soutenu par une communauté de particuliers et d'entreprises. Il utilise une architecture de type MVC pour séparer les données, l'interface utilisateur et les actions, ce qui permet une meilleure gestion des responsabilités.[23]



Spring Boot

Le Spring Framework est largement utilisé dans la communauté Java pour accélérer le développement d'applications d'entreprise, y compris les applications Web et les API. On trouve également des applications basées sur le Spring Framework dans d'autres domaines.[24]



Python

Python est un langage de programmation préféré pour la programmation en raison de ses fonctionnalités, de son applicabilité et de sa simplicité. Le langage de programmation Python est particulièrement adapté à l'apprentissage automatique en raison de sa plateforme indépendante et de sa popularité au sein de la communauté des programmeurs.[25]

Conclusion

Dans ce chapitre, nous avons élaboré notre plan de travail pour construire le Product Backlog, en déterminant les priorités et en organisant les sprints. Nous avons identifié les besoins fonctionnels de notre application, défini les rôles des utilisateurs, créé les maquettes correspondantes et enfin précisé les environnements de développement matériel et logiciel.

Chapitre 3

Techniques d'intelligence artificielle

Introduction

Ce chapitre vise à présenter les technologies d'IA utilisées dans le développement de notre projet .

3.1 L'intelligence artificielle

L'intelligence artificielle (IA) est un processus qui imite l'intelligence humaine en utilisant des algorithmes dans un environnement informatique dynamique. Son but est de permettre aux ordinateurs de penser et de se comporter comme des êtres humains.

Il existe plusieurs branches de l'intelligence artificielle, qui sont bien détaillées dans la figure suivante. Parmi ces branches se trouve l'apprentissage automatique .[26]

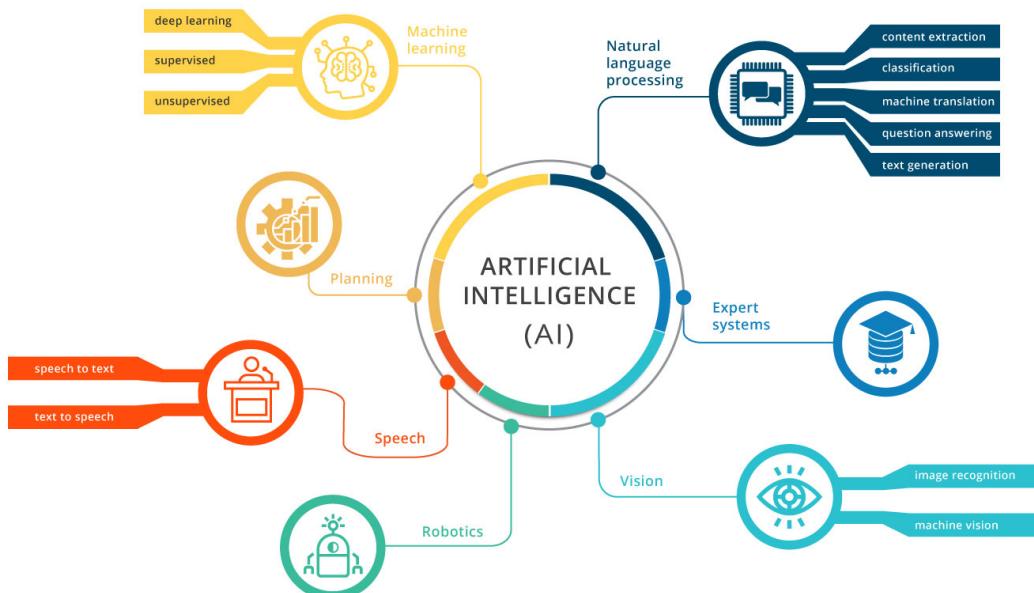


FIGURE 3.1 – Les différentes branches de l'intelligence artificielle [5]

3.2 L'apprentissage automatique (Machine learning)

L'apprentissage automatique ,aussi appelé apprentissage machine, est une branche de l'intelligence artificielle (IA) et de l'informatique qui se concentre sur l'utilisation des données et des algorithmes pour reproduire la façon dont les êtres humains apprennent. Son objectif est d'améliorer progressivement la précision des systèmes en imitant le processus d'apprentissage humain.[27]

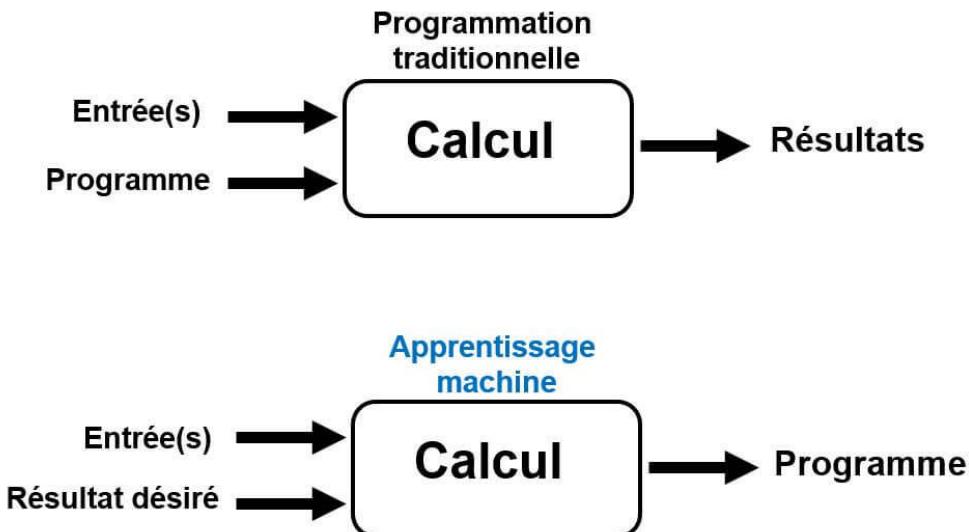


FIGURE 3.2 – L'apprentissage automatique [6]

3.2.1 Les types d'apprentissage automatique

Il y a différents types d'apprentissage automatique, en fonction des données disponibles et des méthodes d'apprentissage. Dans cet article de blog, nous nous concentrerons sur deux types en particulier : l'apprentissage supervisé et l'apprentissage non supervisé.

le tableau suivante explique les différences entre eux :

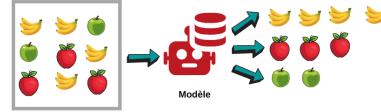
l'apprentissage supervisé	l'apprentissage non supervisé
<p>Dans ce type d'apprentissage : l'algorithme apprend à partir de données annotées ou étiquetées qui lui sont présentées. Autrement dit, les données d'entraînement comprennent les observations et leurs étiquettes, également appelées labels. (avec intervention humaine.)</p>	<p>Dans ce type d'apprentissage : caractérisée par l'analyse et le regroupement de données non étiquetées. Ces algorithmes apprennent ainsi à trouver des schémas ou des groupes dans les données, avec très peu d'intervention humaine.</p>
<p>Exemple : la classification des données : Lorsqu'on utilise une application de reconnaissance d'images pour identifier des individus, l'algorithme est entraîné à partir de plusieurs images. Chaque image est étiquetée avec le nom de la personne correspondante, ce qui permet à l'algorithme d'apprendre les caractéristiques spécifiques de chaque individu.[28]</p> <div style="display: flex; align-items: center;"> <div style="flex-grow: 1; margin-right: 20px;"> <p>Entraînement avec plusieurs images d'individus</p>  <p>Anita Eric Sophie Marc Paul</p> <p>Exemples d'entraînement : $\{(image, prénom)\}$ $\{(donnée, étiquette)\}$ $\{(X, Y)\}$</p> </div> <div style="text-align: center;"> <p>Utilisation: Qui est-ce?</p>  <p>( , prénom?)</p> </div> </div>	<p>Exemple : le partitionnement des données Le partitionnement des données, également appelé clustering, consiste à séparer un ensemble de données en différents groupes afin de maximiser la similarité entre les individus d'un même groupe et de minimiser celle entre les individus de groupes différents.[29]</p> <div style="text-align: center;">  </div>

TABLE 3.1 – différence entre l'apprentissage supervisé et non supervisé

3.2.2 les principaux algorithmes de Machine Learning

Il existe une large gamme d'algorithmes d'apprentissage automatique. Cependant, certains sont plus couramment utilisés que d'autres. Tout d'abord, différents algorithmes sont utilisés pour les données étiquetées.

1. Les algorithmes de régression : tels que la régression linéaire ou logistique, permettent de comprendre les relations entre les données. La régression linéaire est utilisée pour prédire la valeur d'une variable dépendante en fonction de la valeur d'une variable indépendante. Par exemple, prédire les ventes annuelles d'un commercial en fonction de son niveau d'études ou de son expérience.
2. La régression logistique : quant à elle, est utilisée lorsque les variables dépendantes sont binaires. Un autre type d'algorithme de régression appelé machine à vecteur de support est pertinent lorsque les variables dépendantes sont plus difficiles à classer.
3. L'arbre de décision : est un algorithme qui génère des recommandations en se basant sur un ensemble de règles de décision et des données classifiées. Par exemple, il peut être utilisé pour recommander sur quelle équipe de football parier en prenant en compte des données telles que l'âge des joueurs ou le taux de victoire de l'équipe.
4. Clustering : est souvent utilisé pour les données non étiquetées. Cette méthode implique l'utilisation d'algorithmes pour identifier les groupes présentant des enregistrements simi-

laires, puis étiqueter ces enregistrements en fonction du groupe auquel ils appartiennent.

5. Les réseaux de neurones : sont des algorithmes qui adoptent une structure en couches multiples. La première couche est responsable de l'ingestion des données, tandis que une ou plusieurs couches cachées permettent de tirer des conclusions à partir de ces données. Enfin, la dernière couche attribue une probabilité à chaque conclusion.[30]

3.3 L'apprentissage profond (Deep Learning)

Le deep learning, ou apprentissage profond, est un sous-domaine du machine learning, ou apprentissage machine, qui est lui-même un sous-domaine de l'intelligence artificielle.

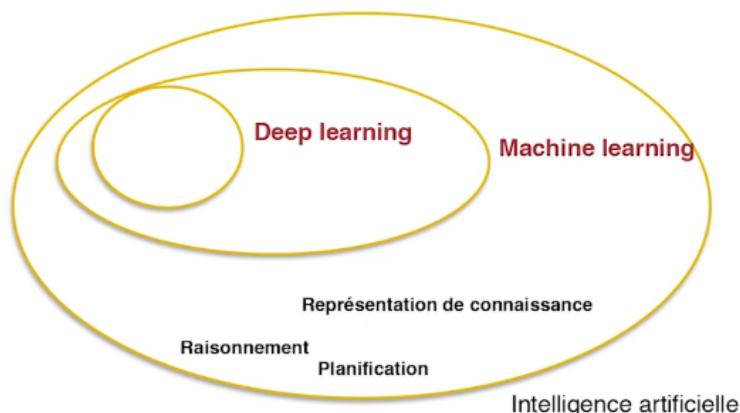


FIGURE 3.3 – L'intelligence artificielle et ses sous-domaines [7]

Le deep learning est une branche du machine learning (apprentissage automatique) dans laquelle des réseaux de neurones artificiels - des algorithmes conçus pour imiter le fonctionnement du cerveau humain - apprennent à partir de vastes quantités de données.

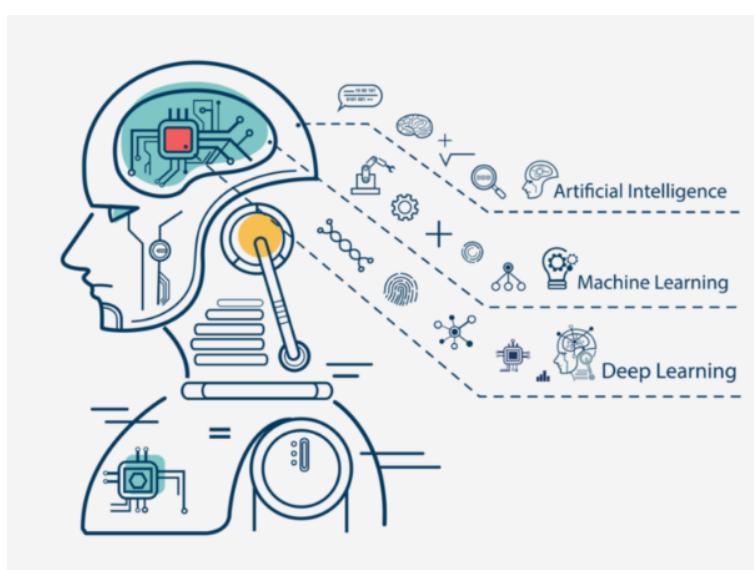


FIGURE 3.4 – Deep Learning [8]

3.3.1 Les types d'apprentissage profond

Le deep learning a 3 types d'apprentissage : supervisé, semi-supervisé, et par renforcement. Nous allons les expliquer dans ce tableau :

Apprentissage semi-supervisé	Apprentissage par renforcement
<p>L'apprentissage semi-supervisé permet de former un ordinateur à résoudre un problème en lui fournissant des exemples étiquetés contenant la réponse correcte, ainsi que des exemples non étiquetés sans la réponse correcte.[31].</p> 	<p>L'apprentissage par renforcement met l'accent sur l'apprentissage du système à travers ses interactions avec l'environnement où le système adapte ses paramètres en fonction des réactions reçues de l'environnement, qui fournit ensuite un retour d'information sur les décisions prises.[32]</p> 

TABLE 3.2 – L'apprentissage semi-supervisé et par renforcement

3.3.2 Comment fonctionne le deep learning ?

Les réseaux de deep learning[33] sont entraînés à partir de structures de données complexes avec lesquelles ils interagissent. Ils créent des modèles de calcul qui comportent plusieurs couches de traitement afin de représenter les données à différents niveaux d'abstraction.

Le Deep Learning repose sur :

- Réseaux de neurones profonds.
- Algorithme d'entraînement spécifique

Quel sont les Réseaux de neurones profonds et Algorithme d'entraînement spécifique

Les algorithmes de deep learning s'inspirent du fonctionnement du cerveau humain. Par exemple, le cerveau humain est constitué de millions de neurones interconnectés qui collaborent pour apprendre et traiter l'information.

De manière similaire, les **réseaux neuronaux de deep learning**, également connus sous le nom de réseaux de neurones artificiels, sont composés de multiples couches de neurones artificiels qui travaillent de concert à l'intérieur d'un ordinateur[34].

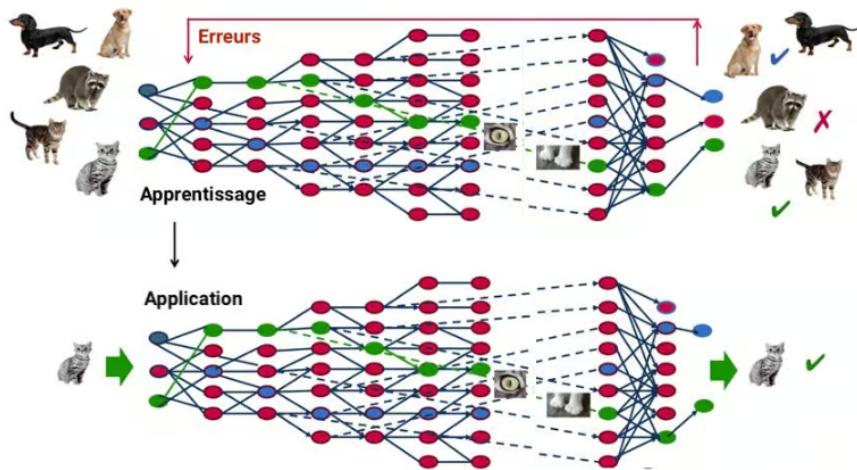


FIGURE 3.5 – Fonctionnalité de deep Learning [9]

le figure précédent (Figure 3.5) montre que la machine est capable d'identifier un chat sur une photo grâce à un processus d'auto-apprentissage appelé Deep Learning. Chaque couche du réseau neuronal correspond à un aspect spécifique de l'image.

Pour Conclure, en dit que les neurones artificiels sont des nœuds utilisés pour le traitement des données.

Et les réseaux neuronaux artificiels sont des algorithmes de deep learning qui utilisent ces neurones pour résoudre des problèmes complexes.

3.3.3 Les algorithmes de deep learning(L'apprentissage profond)

Dans cette partie, nous présentons différents types de modèles et d'algorithmes d'apprentissage profond. Dans les années 1980, la plupart des réseaux neuronaux étaient composés d'une seule couche en raison du coût élevé des calculs et de la disponibilité limitée des données. De nos jours, il est possible d'avoir davantage de couches cachées dans nos réseaux neuronaux, d'où le terme "apprentissage profond". De plus, différents types de réseaux neuronaux sont maintenant disponibles, tels que les réseaux de neurones convolutifs (CNN), Réseaux adversariaux génératifs (GAN) ,les réseaux de neurones récurrents (RNN) et les réseaux de neurones FeedForward (FNN).

- **Réseaux neuronaux convolutifs (CNN) :**

Les CNN, également connus sous le nom de ConvNets, sont utilisés pour le traitement d'images et la détection d'objets.

Ils sont largement utilisés dans divers domaines tels que l'identification d'images satellites, le traitement d'images médicales, la prédition de séries chronologiques et la détection d'anomalies. Les CNN sont composés de plusieurs couches qui traitent et extraient les fonctionnalités des données de manière efficace :

- **Couche de convolution :** le CNN possède une couche de convolution qui contient plusieurs filtres pour effectuer l'opération de convolution.
- **Unité linéaire rectifiée (ReLU) :** les CNN ont une couche ReLU pour effectuer des opérations sur les éléments. Le résultat est une carte de fonctionnalités rectifiée.
- **Couche de mise en commun :** la carte de caractéristiques rectifiée est ensuite intro-

duite dans une couche de pooling. Le pooling est une opération de sous-échantillonnage qui réduit les dimensions de la carte des caractéristiques.

- **La couche de regroupement :** convertit ensuite les tableaux 2D résultants de cartes d'entités regroupées en un seul vecteur linéaire, long et continu en les aplatisant.
- **Couche entièrement connectée :** une couche entièrement connectée est formée lorsque la matrice aplatie de la couche de regroupement est utilisée comme entrée, permettant la classification et l'identification des images.
- **Réseaux adversariaux génératifs (GAN) :**, sont des algorithmes de deep learning qui permettent de créer de nouvelles données similaires à celles utilisées pour l'apprentissage. Ils se composent d'un générateur qui apprend à générer des données fausses, et d'un discriminateur qui apprend à les distinguer. L'utilisation des GAN s'est répandue au fil du temps. Ils sont employés pour améliorer les images astronomiques et simuler des phénomènes de lentilles gravitationnelles dans la recherche sur la matière noire. Les développeurs de jeux vidéo utilisent également les GAN pour améliorer les textures 2D à faible résolution des anciens jeux en les recréant en 4K ou à des résolutions plus élevées grâce à l'apprentissage par l'image.
[35]
- **Réseaux neuronaux récurrents (RNN) :** Il s'agit d'une classe de réseaux de neurones artificiels appelés réseaux de neurones récurrents. Ces réseaux utilisent des connexions entre les nœuds pour former un réseau dirigé graphiquement le long d'une séquence temporelle.
Ils sont conçus pour résoudre des problèmes de prédiction de séquence, y compris les séquences de texte. Ils sont également utilisés comme modèles génératifs capables de produire des séquences en sortie, non seulement du texte, mais aussi dans des applications telles que la génération d'écriture manuscrite, de données textuelles et de reconnaissance vocale..[35]
- **Réseaux de neurones FeedForward (FNN) :** Un réseau neuronal Feed Forward est un type de réseau neuronal artificiel où les connexions entre les nœuds ne forment pas de boucle. C'est le contraire d'un réseau neuronal récurrent, qui suit des chemins spécifiques. Le modèle à propagation avant est le type le plus basique de réseau neuronal, car l'entrée ne fait qu'avancer dans un seul sens. Même si les données peuvent circuler à travers de nombreux noeuds cachés, elles ne font que progresser vers l'avant et jamais en sens inverse.

3.4 Traitement du langage naturel (NLP)

Le traitement du langage naturel (NLP) est une branche de l'intelligence artificielle (IA) et représente un domaine du machine learning qui habilite les ordinateurs à interpréter, traiter et saisir le langage humain.[36]

La figure ci-dessous représente la relation entre le NLP, le machine learning, le deep learning, et l'intelligence artificielle (fig 3.6).

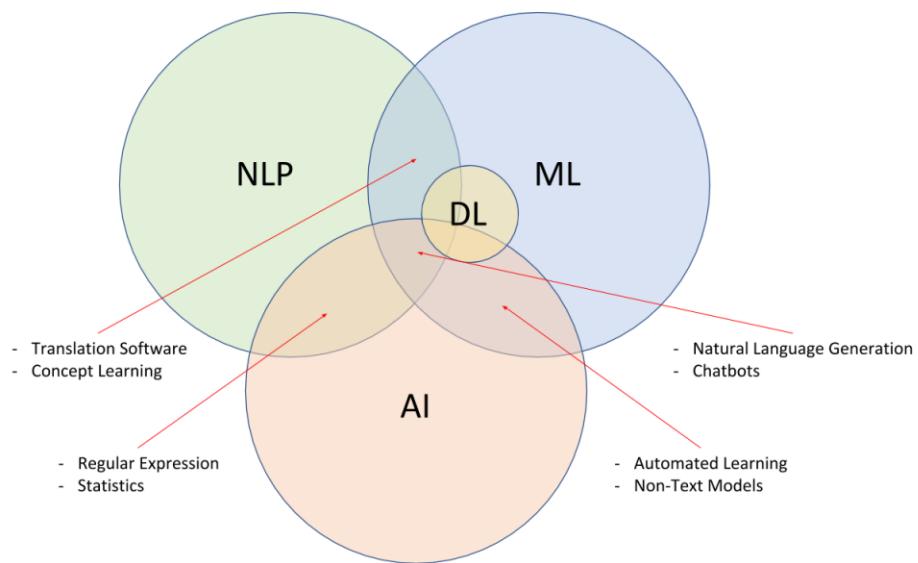


FIGURE 3.6 – L'intersection de la NLP, du machine learning, du deep learning et de l'intelligence artificielle.[10]

3.4.1 l'importance de le traitement du langage naturel

La principale raison pour laquelle le traitement du langage naturel revêt une importance extrême réside dans sa capacité à analyser et à donner un sens à d'énormes quantités de données. Il aide à traiter les données textuelles et vocales, à comprendre les émotions et les intentions, ainsi qu'à extraire des informations cruciales.

Le langage naturel est complexe et les données sont généralement non structurées. Les données textuelles peuvent contenir des erreurs orthographiques, des abréviations et des erreurs de ponctuation, tandis que les données vocales peuvent être affectées par des accents régionaux, des bégaiements, des marmonnements, etc.

Differentes langues ont leurs propres règles grammaticales et syntaxiques, et il est fréquent d'utiliser des mots et des expressions d'autres langues.

Le traitement du langage naturel (NLP) est crucial car il permet de structurer ces données non structurées et de réduire l'ambiguité.[37]

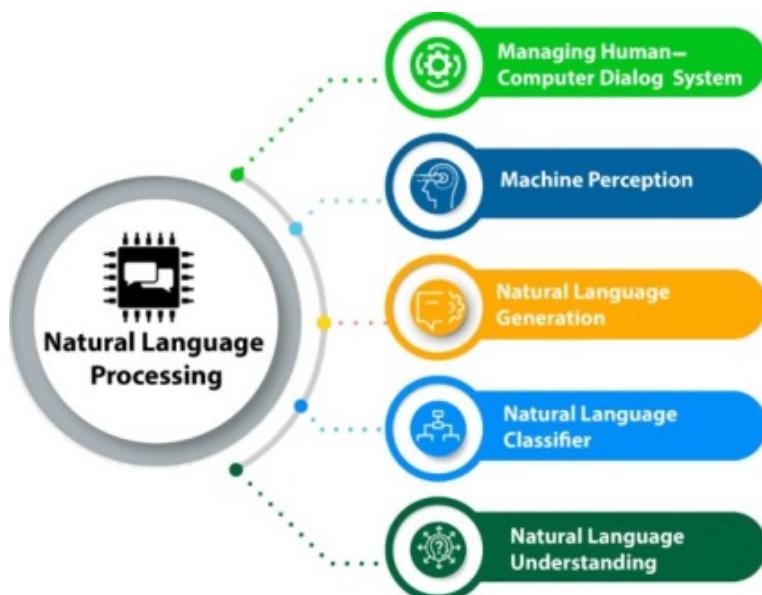


FIGURE 3.7 – l'importance de NLP [11]

3.4.2 Comment fonctionne le traitement du langage naturel (NLP) ?

Les outils de NLP permettent de convertir le texte en un format compréhensible par les machines grâce à la vectorisation du texte. Ensuite, les algorithmes d'apprentissage automatique utilisent des données d'entraînement et des sorties attendues (balises / tags) pour apprendre aux machines à établir des associations entre une entrée spécifique et sa sortie correspondante. Avant de faire des prédictions sur des données invisibles, les machines utilisent des méthodes d'analyse statistique pour créer leur propre "banque de connaissances" et déterminer quelles caractéristiques représentent le mieux les textes.[38]

3.4.3 Les applications du traitement du langage naturel

La figure ci-dessous(Figure 3.8) représente les différentes applications du traitement du langage naturel .

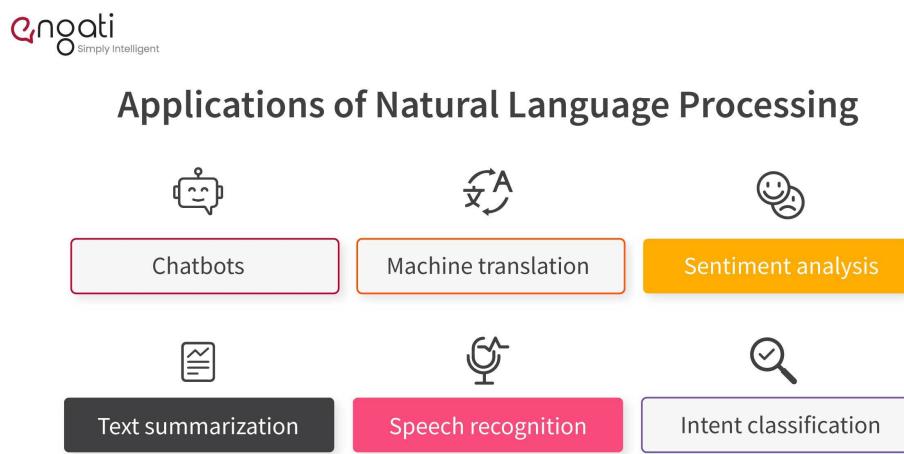


FIGURE 3.8 – Les applications NLP [12]

Ci-dessous, quelques exemples de domaines où le traitement du langage naturel (NLP) est appliqué :

1. Les chatbots :

Les chatbots sont utilisés pour automatiser le service client, la génération de leads, les ventes et d'autres fonctions. Ils utilisent le NLP (Natural Language Processing) pour comprendre les requêtes des utilisateurs et fournir des réponses appropriées.

2. Traduction automatique :

Cela consiste à automatiser la traduction de données d'une langue à une autre. Les modèles de traduction peuvent même être entraînés pour des domaines spécifiques afin d'améliorer la précision de la traduction

3. Analyse des sentiments :

Elle est utilisée pour analyser le texte et détecter les nuances des émotions et des opinions. Cela permet de comprendre le degré de positivité ou de négativité des données.

4. Résumé de texte

Cela consiste à générer des résumés de grands volumes de texte en extrayant les informations les plus critiques et pertinentes.

5. Reconnaissance vocale :

Cela fait référence à l'utilisation du NLP pour convertir les données vocales en un format lisible par une machine.

6. Classification des intentions :

Le NLP peut être utilisé pour comprendre la motivation sous-jacente et l'objectif des données textuelles. Cela est très utile dans le service client, le marketing, les ventes, etc.[39]

Conclusion

Dans ce chapitre, nous avons exposé la notion d'intelligence artificielle, défini les concepts d'apprentissage automatique et profond, ainsi que le traitement du langage naturel et son mode de fonctionnement, ainsi que ses applications.

Chapitre 4

Release 1 : Gestion des Comptes Utilisateurs et Administration

Introduction

Ce chapitre sera consacré à la réalisation de la première release. Le travail sera structuré en trois étapes : analyse, conception et réalisation.

4.1 Organisation des sprint de Release 1

Pour la Release 1, nous avons organisé notre travail en deux sprints. Chaque sprint a débuté par une réunion de planification, où nous avons défini les objectifs et sélectionné les tâches à réaliser. Dans le sprint 1, nous avons implémenté l'authentification, l'inscription des utilisateurs et la gestion des profils. Dans le sprint 2, nous avons travaillé sur les fonctionnalités d'Admin.

4.2 Sprint 1

4.2.1 But du sprint

L'objectif de ce sprint est de créer une interface principale fonctionnelle et conviviale pour le site, une interface d'inscription pour l'utilisateur afin qu'il puisse enregistrer ses données personnelles sur sa page personnelle, ainsi qu'une interface permettant d'accéder à la page de l'utilisateur.

4.2.2 Sprint Backlog

N°	User story	Estimation(jours)
1	En tant que utilisateur je veux inscrire afin que s'accède à l'application web.	4
2	En tant que utilisateur je veux s'authentifier pour accéder a mon profil.	3
3	En tant que utilisateur je veux gérer mon profil.	5
4	En tant qu'utilisateur, je veux consulter les publications des styles tendances.	2

TABLE 4.1 – Backlog du sprint 1

4.2.3 Spécification fonctionnelle

La spécification fonctionnelle dans notre cas se traduit par les diagrammes de cas d'utilisation du premier sprint et leurs descriptions textuelles.

Diagramme de cas d'utilisation Sprint 1

Cette figure (Figure 4.1) présente le diagramme de cas d'utilisation global du premier sprint.

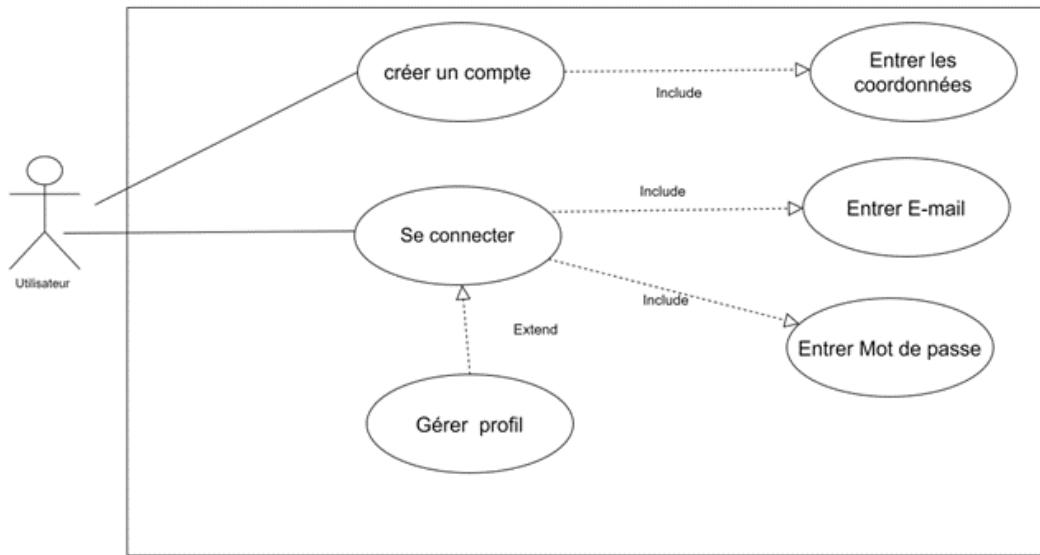


FIGURE 4.1 – Diagramme de cas d'utilisation Sprint 1

Raffinement du cas d'utilisation : Gérer profil

La figure ci-dessous (Figure 4.2) décrit le diagramme de cas d'utilisation «Gérer profil»

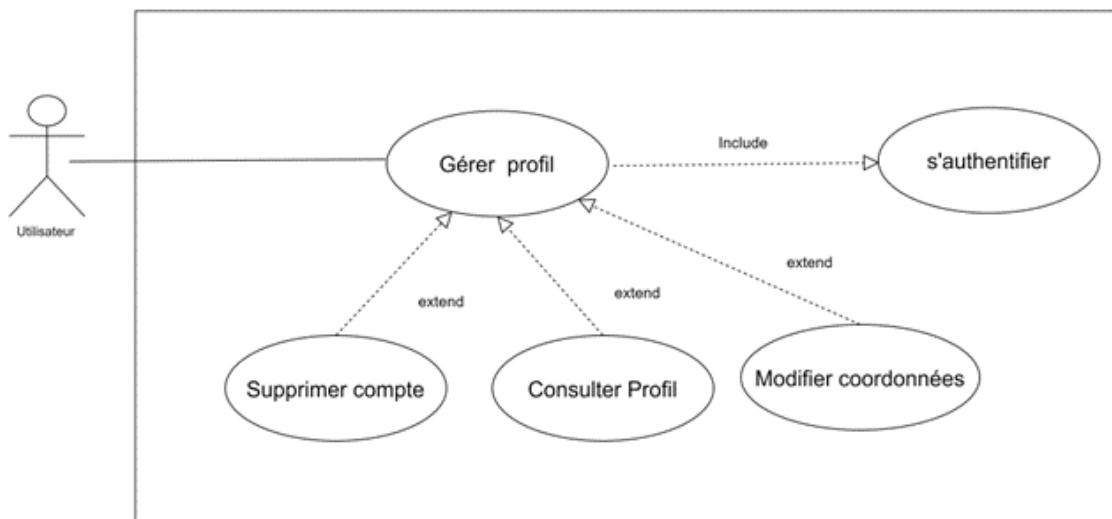


FIGURE 4.2 – cas d'utilisation Gérer profil

description de cas d'utilisation : Gérer profil

Ce tableau montre la description textuelle du cas d'utilisation « Gérer profil : Modifier »

Cas d'utilisation	Gérer profil : Modifier
Acteurs	Utilisation
Pré-condition	L'utilisateur doit être authentifié.
Scénario Normal	<ul style="list-style-type: none"> 1. L'utilisateur accède à Profile 2. L'utilisateur saisit les nouvelles informations 3. Le système afficher un popup pour confirmer . 4. L'utilisateur confirmer la modification 5. Le système vérifie les données 6. Le système enregistre les données 7. Le système affiche la nouvelle modification
Scénario alternatif	L'utilisateur ne confirme pas la modification
Post condition	Compte modifié

TABLE 4.2 – Description textuelle du Gérer profil

Ce tableau montre la description textuelle du cas d'utilisation « Gérer profil :Supprimer »

Cas d'utilisation	Gérer profil : Supprimer
Acteurs	Utilisation
Pré-condition	L'utilisateur doit être authentifié.
Scénario Normal	<ul style="list-style-type: none"> 1. L'utilisateur accède à Profile 2. L'utilisateur choisir l'option de suppression de profil 3. Le système affiche un popup de confirmation pour la suppression 4. L'utilisateur confirme la suppression 5. Le système enregistre les données
Scénario alternatif	L'utilisateur ne confirme pas la suppression
Post condition	Compte supprimé.

TABLE 4.3 – Description textuelle du Gérer profil : Supprimer

4.2.4 Conception

les diagrammes de séquence de Sprint 1

- diagrammes de séquence de cas d'utilisation :Gérer profil : Modifier

Le diagramme suivant décrit le cas d'utilisation du scénario «Gérer profil » ,Ou l'utilisateur peut effectuer toutes les modifications dans leur profil : supprimer une image de profil , changer les coordonnées personnelles...

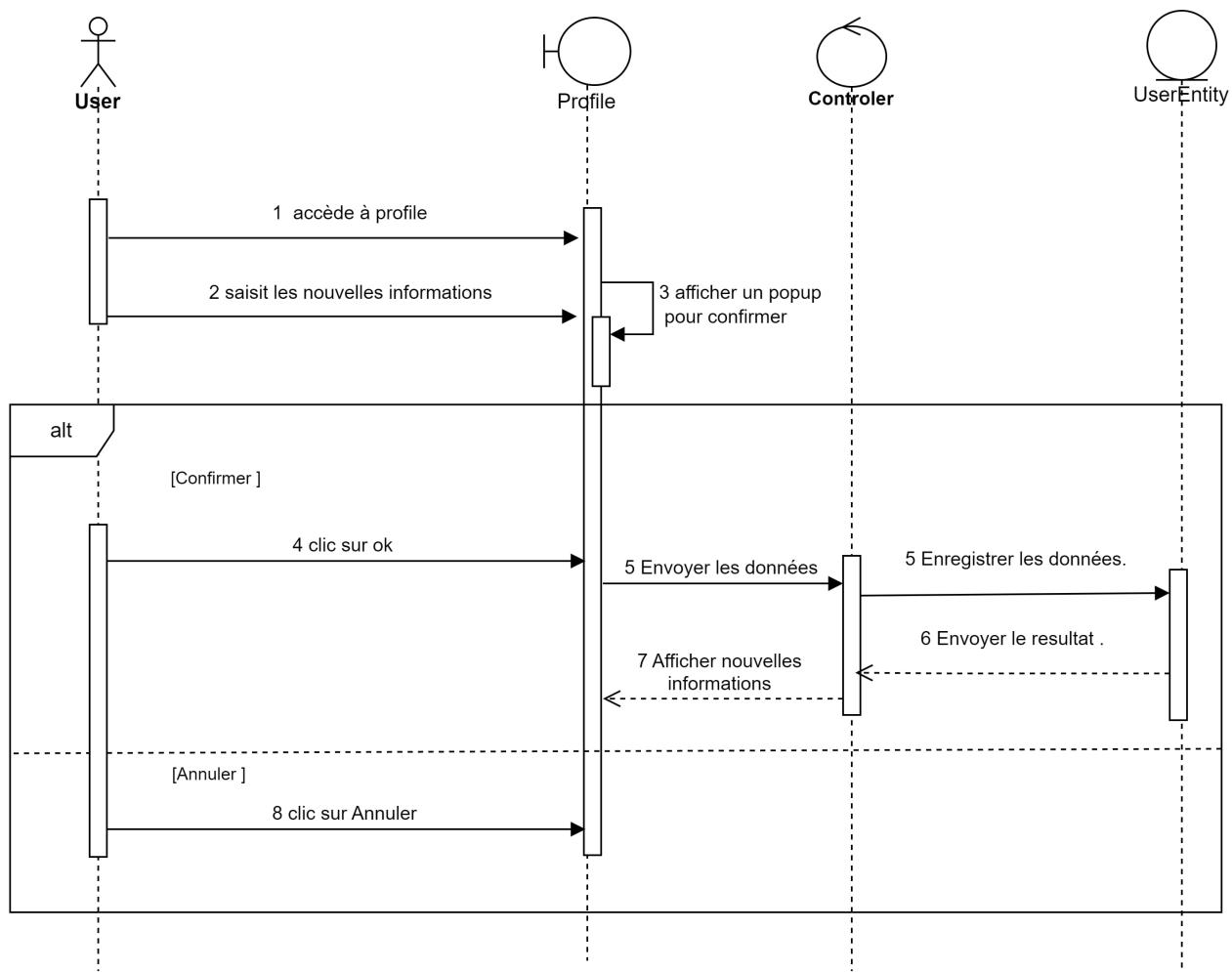


FIGURE 4.3 – Diagramme de séquence «Gérer profil : Modifier »

- diagrammes de séquence de cas d'utilisation : Supprimer

Le diagramme suivant décrit le cas d'utilisation du scénario «Supprimer » ,Ou l'utilisateur peut supprimer son profil .

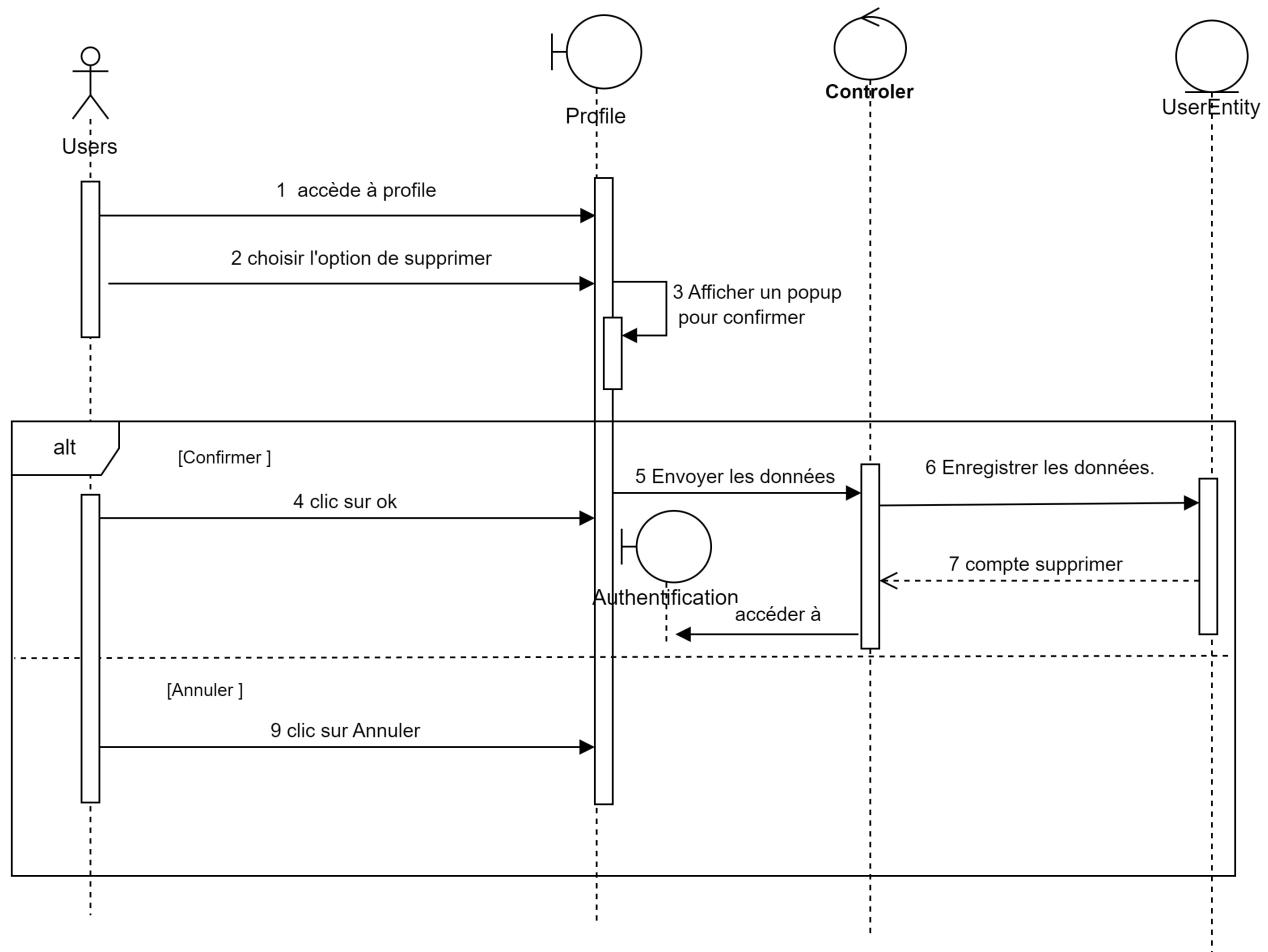


FIGURE 4.4 – Diagramme de séquence «Gérer profil : Supprimer»

4.2.5 Réalisation

Cette partie présente l'œuvre achevée à travers des captures d'écran des différentes interfaces développées lors de ce sprint.

- **Page d'accueil**

Ces figures présentent l'interface graphique de la page d'accueil. :

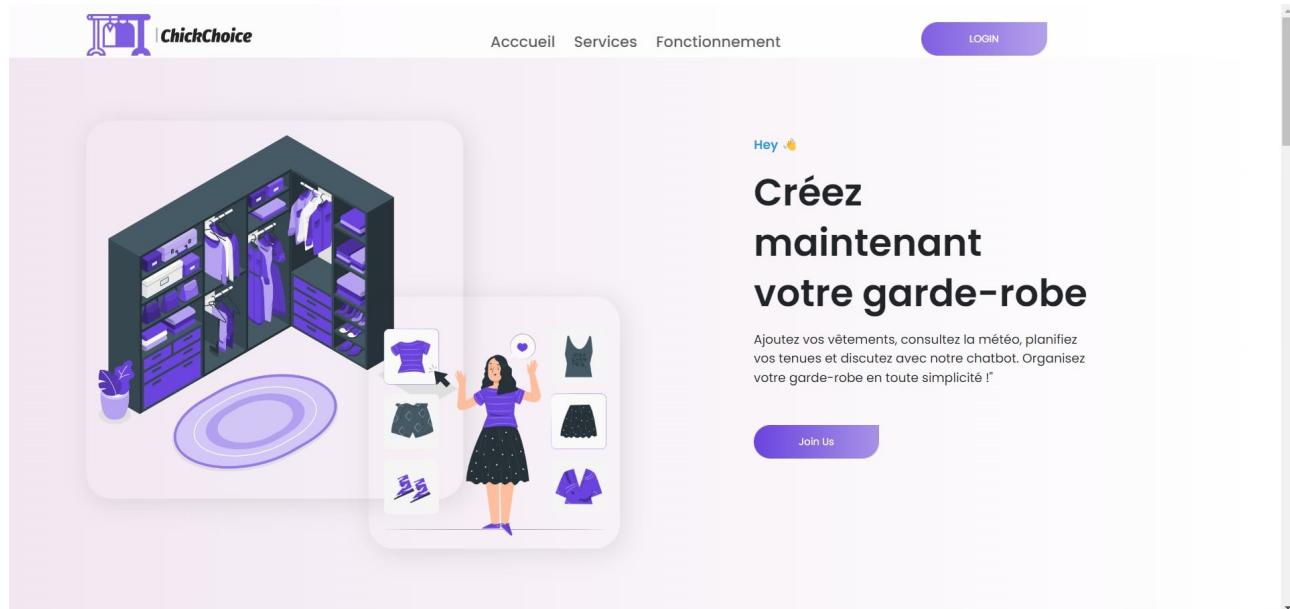


FIGURE 4.5 – Page d'accueil

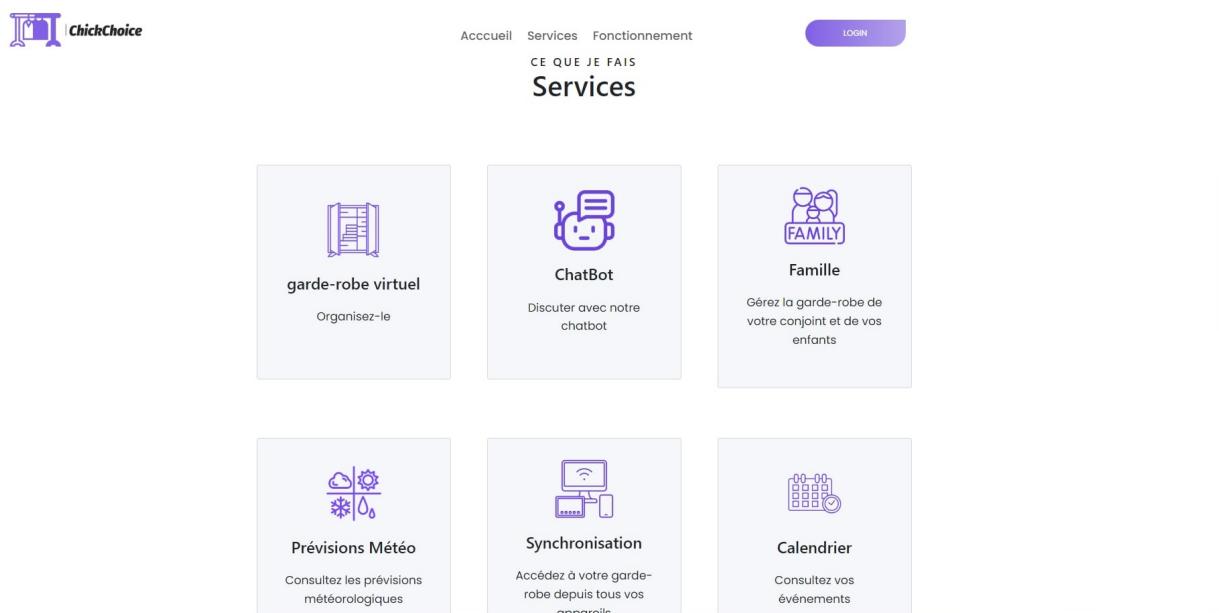


FIGURE 4.6 – Page d'accueil

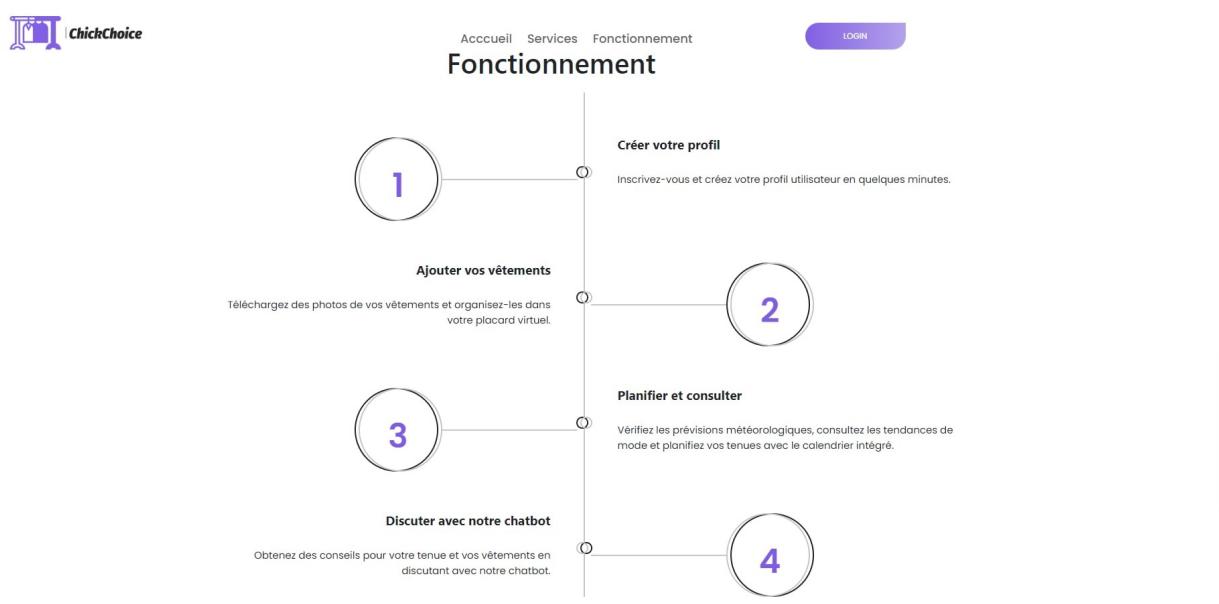


FIGURE 4.7 – Page d'accueil



This figure shows a detailed view of the ChickChoice homepage's footer area. The footer is divided into several sections:

- About**: A welcome message: "Bienvenue sur ChickChoice ! ChickChoice est une plateforme innovante conçue pour vous aider à gérer votre garde-robe de manière efficace et amusante. Notre site vous offre une gamme de fonctionnalités pour simplifier votre quotidien".
- Links**: Navigation links for 'Accueil', 'Services', and 'Fonctionnement'.
- Services**: A list of services including 'garde-robe virtuel', 'ChatBot', 'Famille', 'Prévisions Météo', 'Synchronisation', and 'Calendrier'.
- Subscribe**: A purple button labeled 'Créez votre profil'.
- Copyright**: "Copyright©2024 All rights reserved".
- Suivez-nous**: Social media links for Facebook, Twitter, Instagram, and LinkedIn.
- Page URL**: "localhost:4200/#"

FIGURE 4.8 – Page d'accueil

• Authentification

Cette figure présente l'interface graphique de l'authentification :

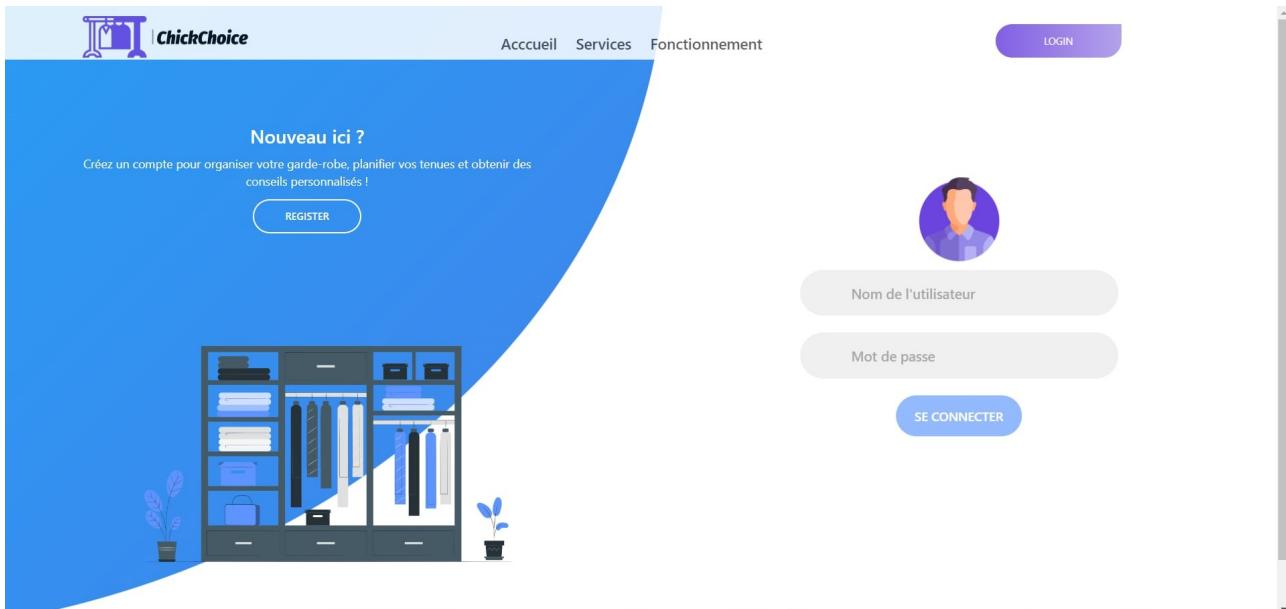


FIGURE 4.9 – Authentification

- **S'insecrire**

Cette figure présente l'interface graphique de l'inscription :

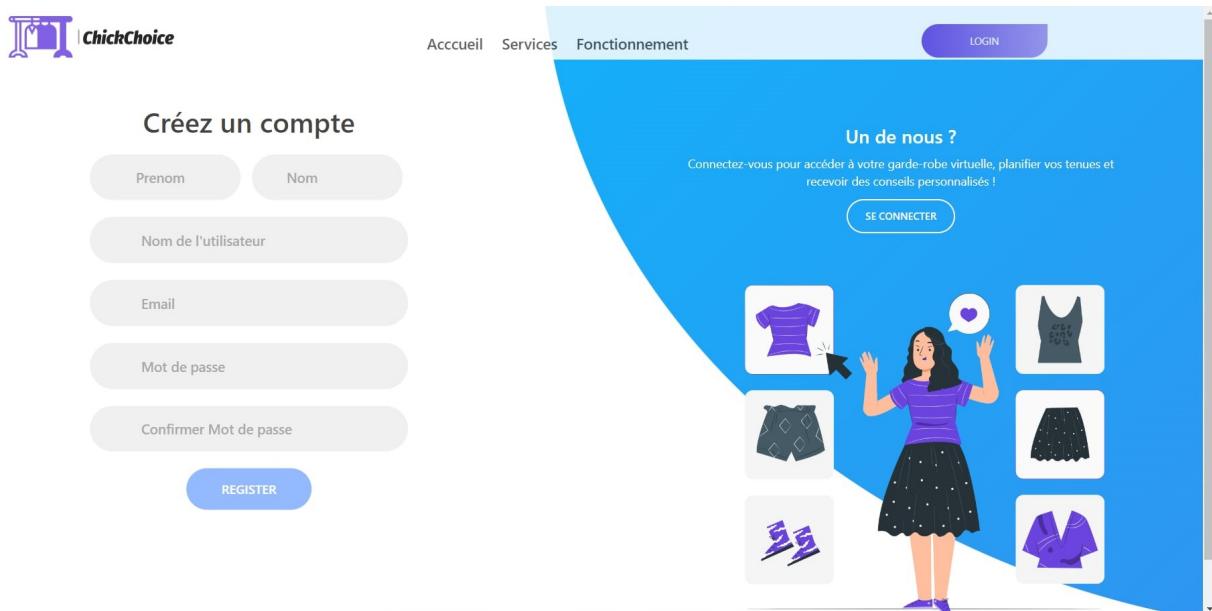


FIGURE 4.10 – S'insecrire

- Profil utilisateur

Cette figure présente l'interface graphique de la page Profil utilisateur :

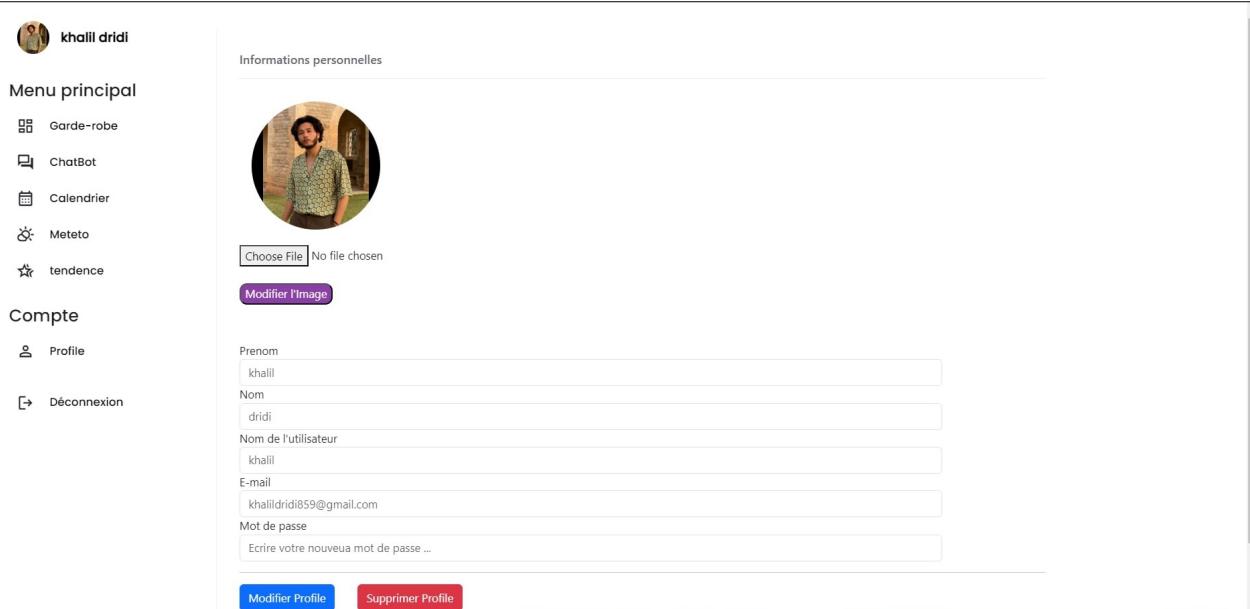


FIGURE 4.11 – Profil utilisateur

- Tendance

Cette figure présente l'interface graphique de la page tendance :

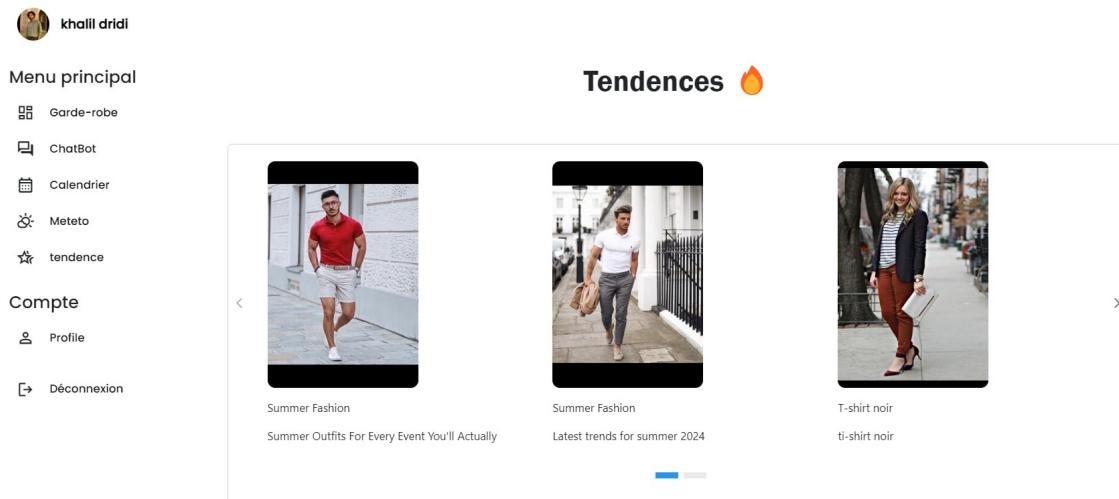


FIGURE 4.12 – Tendance

4.2.6 Validation des Fonctionnalités du Sprint 1

Les tests permettent de vérifier que le sprint assure les fonctionnalités spécifiées. Avant la fin de chaque sprint, nous effectuons des tests sur les fonctionnalités du module. Ensuite, nous validons toutes les fonctionnalités avec le Product Owner. Pour ce faire, nous avons élaboré un tableau contenant un ensemble de scénarios de tests fonctionnels relatifs à ce sprint.

Cas de test	Démarche	Comportement attendu	Résultat
S'inscrire	Introduire les informations pour créer un compte	Redirection vers la page d'authentification	Terminé
S'authentifier	Introduire les informations pour se connecter	Redirection vers profil	Terminé
Gérer profil	Introduire les informations pour gérer profil	1. Modifier Profil 2. Supprimer Profil	Terminé
Consulter les publications des styles tendances	Accéder à la section des publications tendances	Redirection vers la page de tendance	Terminé

TABLE 4.4 – Validation du sprint 1

4.3 Sprint 2

4.3.1 But du sprint

Le sprint est une période d'un mois maximum au terme de laquelle l'équipe livre un incrément du produit potentiellement livrable. Une fois la durée du sprint définie, elle reste constante pendant toute la durée du développement. Le Backlog du sprint doit inclure toutes les tâches d'administration, telles que :

- Authentification .
- Consulter des comptes utilisateurs.
- Gérer les publications de style tendance .

4.3.2 Sprint Backlog

N°	User story	Estimation (jours)
1	En tant que Admin je veux s'authentifier pour accéder a mon dashboard.	4
2	En tant que Admin je veux gérer les comptes utilisateurs .	4
3	En tant que Admin je veux gérer les publications de style tendance .	6

TABLE 4.5 – Backlog du Sprint 2

4.3.3 Spécification fonctionnelle

Diagramme de cas d'utilisation du Sprint 2

Cette figure montre le diagramme de cas d'utilisation pour le Sprint 2.

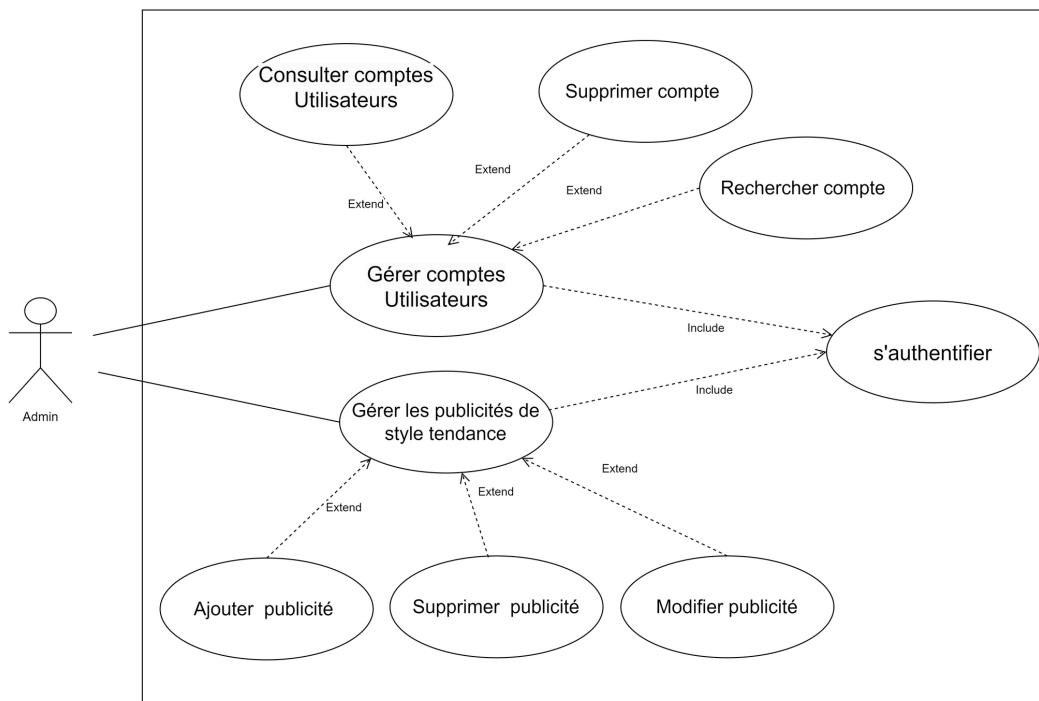


FIGURE 4.13 – Raffinement du cas d'utilisation du Sprint 2

Description du cas d'utilisation Supprimer la publication

Ce tableau montre la description textuelle du cas d'utilisation «Supprimer la publication»

Cas d'utilisation	Supprimer publication
Acteurs	Admin
Pré-condition	L'admin doit être authentifié.
Scénario Normal	<ol style="list-style-type: none"> 1. L'admin accède à son Dashboard . 2. L'admin choisir l'option de suppression la publication. 3. Le système affiche un popup de confirmation pour la suppression. 4. L'admin confirme la suppression 5. Le système enregistre les données
Scénario alternatif	L'administrateur ne confirme pas la suppression
Post condition	La publication est supprimée

TABLE 4.6 – Description textuelle de cas de utilisation Supprimer la publication

4.3.4 Conception

Diagramme de séquence du cas d'utilisation : Supprimer publication

Le diagramme suivant décrit le cas d'utilisation du scénario "Supprimer la publication", dans lequel l'administrateur supprime une publicité .

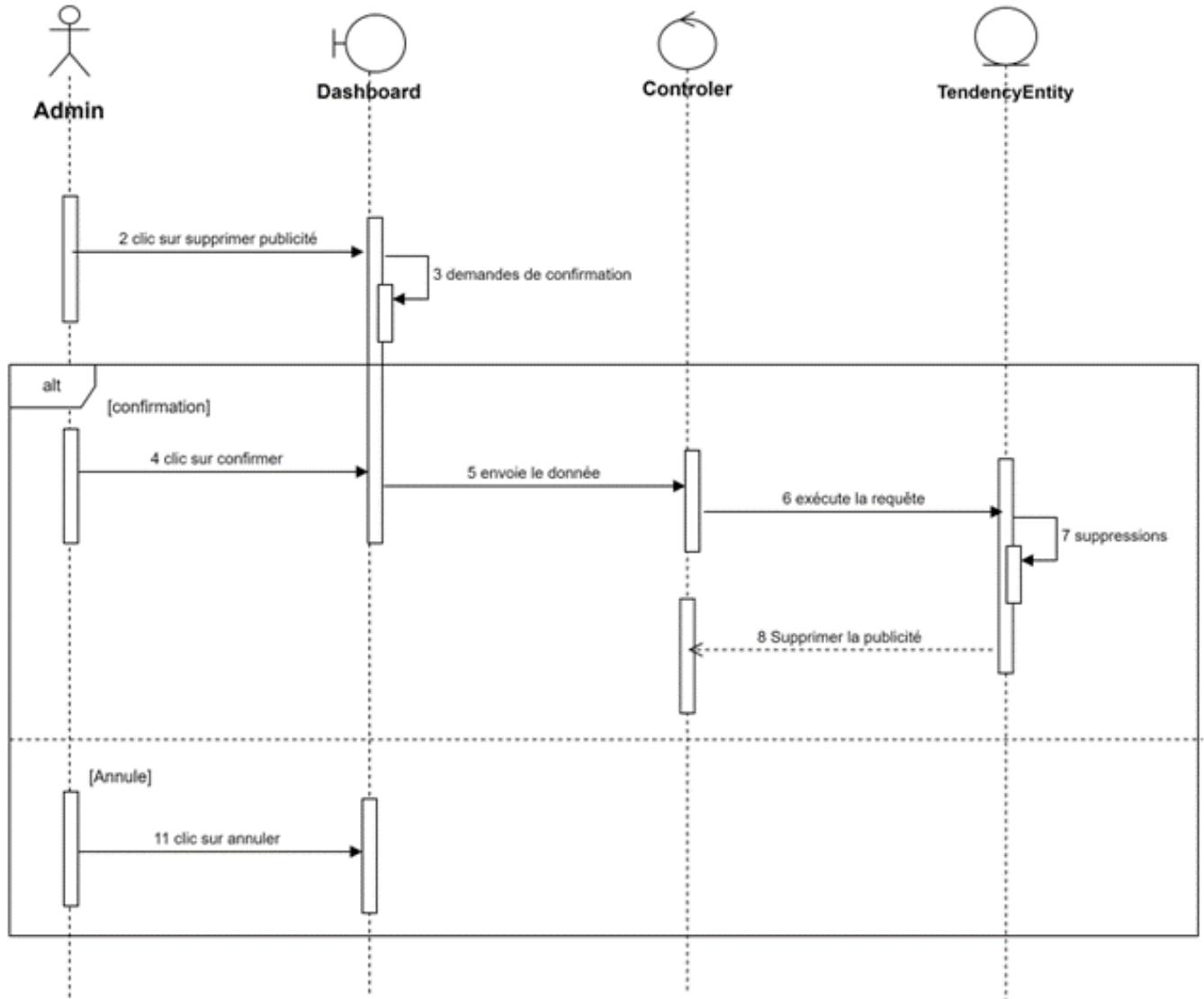


FIGURE 4.14 – Diagramme de séquence «Supprimer la publicité»

4.3.5 Réalisation

- **Authentification Admin**

Cette figure présente l'interface graphique de l'authentification admin :

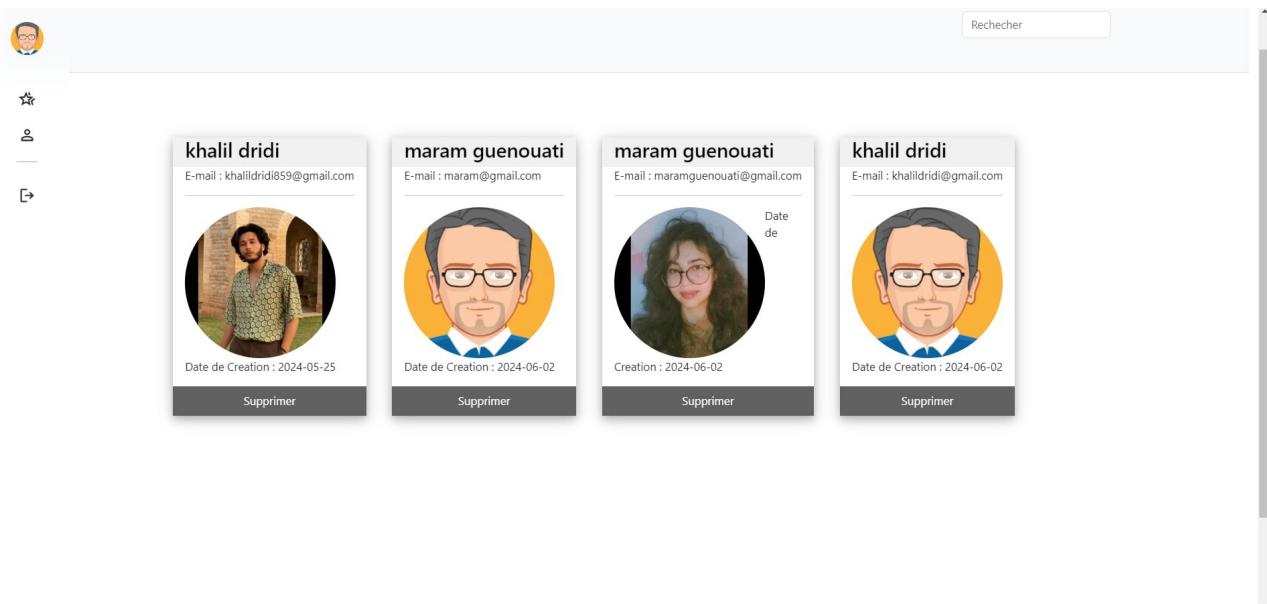
Se connecter

admin
.....
Se connecter

FIGURE 4.15 – Authentification Admin

• Dashboard Admin : Liste des utilisateurs

Cette figure présente l'interface graphique de l'Dashboar admin :



The screenshot shows the Admin Dashboard interface. On the left, there is a sidebar with icons for user management (star, person, envelope). The main area displays a list of four users in cards:

- khalil dridi**
E-mail : khalildridi859@gmail.com

Date de Creation : 2024-05-25
Supprimer
- maram guenouati**
E-mail : maram@gmail.com

Date de Creation : 2024-06-02
Supprimer
- maram guenouati**
E-mail : maramguenouati@gmail.com

Creation : 2024-06-02
Supprimer
- khalil dridi**
E-mail : khalildridi@gmail.com

Date de Creation : 2024-06-02
Supprimer

FIGURE 4.16 – Dashboard Admin : Liste des utilisateurs

- Dashboard Admin : les publications des tendances

Cette figure présente l'interface graphique de l'Dashboar admin : les publications des tendances

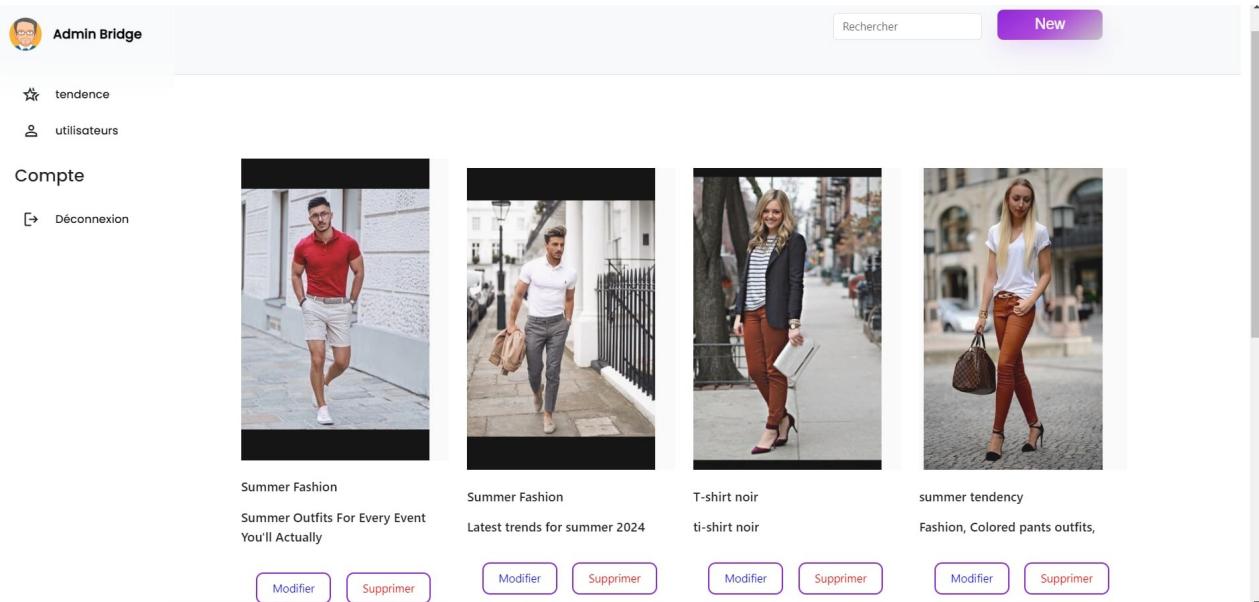


FIGURE 4.17 – Dashboard Admin :les publications des tendances

4.3.6 Validation des Fonctionnalités du Sprint 2

Ce tableau contient les ensembles de scénarios de tests fonctionnels relatifs à ce sprint.

Cas de test	Démarche	Comportement attendu	Résultat
S'authentifier	Introduire les informations pour se connecter	Redirection vers Dashboard	Terminé
Gérer comptes Utilisateurs	Introduire les informations pour gérer comptes utilisateurs	<ol style="list-style-type: none"> consulte la liste des utilisateurs Rechercher les comptes utilisateurs Supprimer les comptes utilisateurs 	Terminé
Gérer les publications de style tendance	Introduire les informations pour gérer les publications	<ol style="list-style-type: none"> Ajouter la publication Modifier la publication Supprimer la publication 	Terminé

TABLE 4.7 – Validation du sprint 2

Conclusion

Dans ce chapitre, la première partie développe la fonctionnalité permettant aux utilisateurs de gérer leur profil.

Dans la deuxième partie, nous avons mis en place les fonctionnalités dédiée à l'administrateur.

Chapitre 5

Release 2 : Garde-Robe Virtuelle et Assistance ChatBot

Introduction

5.1 Organisation des sprint de Release 2

Pour la Release 2, nous avons structuré notre travail en deux sprints. Dans le sprint 3, nous avons ajouté le module météo et calendrier, ainsi que la gestion de la garde-robe., tandis que dans le sprint 4, développé le chatbot

5.2 Sprint 3

5.2.1 But de ce sprint

L'objectif de ce sprint est de créer les interfaces de la garde-robe, du calendrier et de la météo, afin d'introduire leurs fonctionnalités. Ces fonctionnalités permettront aux utilisateurs de créer leur garde-robe virtuelle, de planifier leurs événements et de consulter les prévisions météorologiques avant de sélectionner leurs tenues vestimentaires.

5.2.2 Sprint Backlog

Le Sprint Backlog doit inclure toutes les tâches qui contribuent à la satisfaction de l'utilisateur. Les histoires du Backlog sont :

N°	User story	Estimation (jours)
1	En tant que utlistaeur je veux gérer mon Garde-Robe	6
2	En tant que utilisateur je veux Consulter météo	2
3	En tant que utilisateur je veux Planifier mon calendrier	6

TABLE 5.1 – Backlog du Sprint 3

5.2.3 Spécification fonctionnelle

- Diagramme de cas d'utilisation global du Sprint 2

Cette figure montre le diagramme de cas d'utilisation global pour le Sprint 2.

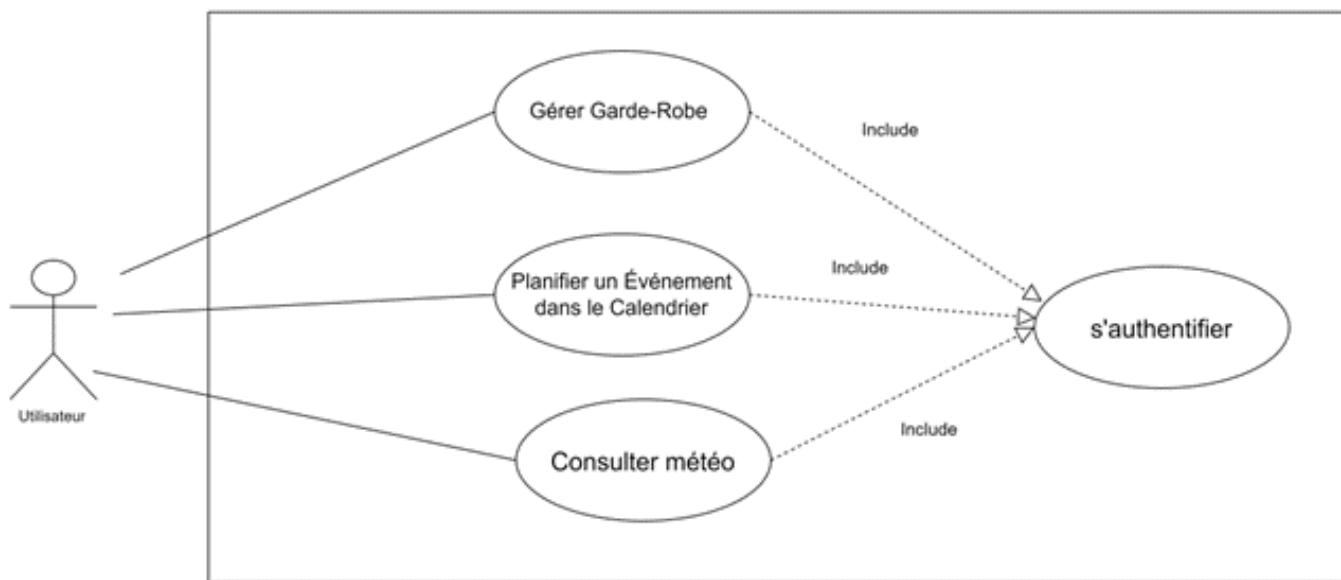


FIGURE 5.1 – Diagramme de cas d'utilisation du Sprint 3

Raffinement et description des cas d'utilisation « Gérer Garde-Robe »

Ce diagramme montre le diagramme de cas d'utilisation « Gérer Garde-Robe ».

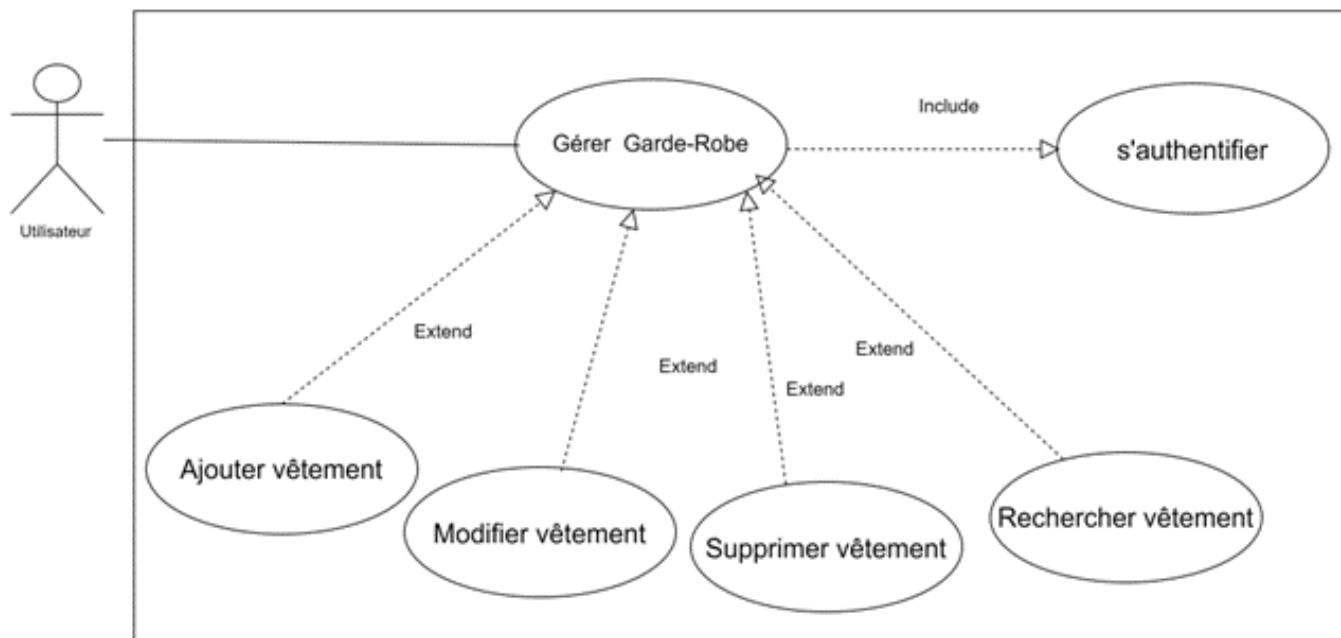


FIGURE 5.2 – Cas d'utilisation gérer Garde-Robe

- **description de cas d'utilisation :Ajouter vêtement**

Ce tableau montre la description textuelle du cas d'utilisation « Ajouter vêtement »

Cas d'utilisation	Ajouter vêtement
Acteurs	Utilisation
Pré-condition	Le utilisateur doit être déjà authentifier
Scénario Normal	<ol style="list-style-type: none"> 1. L'utilisateur accède à sa garde-robe. 2. L'utilisateur choisit le bouton d'ajouter un vêtement. 3. Le système affiche le formulaire d'ajout. 4. L'utilisateur remplit le formulaire 5. L'utilisateur confirme l'ajout . 6. Le système enregistre les données . 7. Le vêtement est enregistré dans la garde-robe .
Scénario alternatif	L'utilisateur ne confirme pas l'ajout .
Post condition	Le vêtement est ajouté à la garde-robe et à la catégorie choisie.

TABLE 5.2 – Description textuelle de cas de utilisation ajouter vêtement

- **description de cas d'utilisation : Modifier vêtement**

Ce tableau montre la description textuelle du cas d'utilisation « Modifier vêtement »

Cas d'utilisation	Modifier vêtement
Acteurs	Utilisation
Pré-condition	Le utilisateur doit être déjà authentifier
Scénario Normal	<ol style="list-style-type: none"> 1. L'utilisateur accède à sa garde-robe. 2. L'utilisateur choisit l'option de modifier le vêtement . 3. Le système affiche le formulaire de modification. 4. L'utilisateur remplit le formulaire. 5. L'utilisateur confirme la modification. 6. Le système enregister le modification .
Scénario alternatif	L'utilisateur ne confirme pas la modification.
Post condition	vêtement est modifié.

TABLE 5.3 – Description textuelle de cas de utilisation modifier vêtement

- description de cas d'utilisation supprimer vêtement**

Ce tableau montre la description textuelle du cas d'utilisation «supprimer vêtement »

Cas d'utilisation	Supprimer vêtement
Acteurs	Utilisation
Pré-condition	Le utilisateur doit être déjà authentifier
Scénario Normal	<ol style="list-style-type: none"> 1. L'utilisateur accède à sa garde-robe. 2. L'utilisateur choisit l'option de supprimer le vêtement . 3. Le système affiche un popup de suppression. 4. L'utilisateur confirmer la suppression . 5. Le systeme enregister les données .
Scénario alternatif	L'utilisateur ne confirme pas la suppression.
Post condition	vêtement est supprimé .

TABLE 5.4 – Description textuelle de cas de utilisation supprimer vêtement

- description de cas d'utilisation Rechercher vêtement**

Ce tableau montre la description textuelle du cas d'utilisation « Rechercher vêtement »

Cas d'utilisation	Rechercher vêtement
Acteurs	Utilisation
Pré-condition	Le utilisateur doit être déjà authentifier
Scénario Normal	<ol style="list-style-type: none"> 1. L'utilisateur accède à sa garde-robe. 2. L'utilisateur choisit l'option de rechercher des vêtements. 3. L'utilisateur écrit le nom du vêtement. 4. Le système affiche le vêtement.
Post condition	Vêtement affiché.

TABLE 5.5 – Description textuelle de cas de utilisation rechercher vêtement

Raffinement du cas d'utilisation : Planification événement

La figure ci-dessous décrit le diagramme de cas d'utilisation « Planification événement »

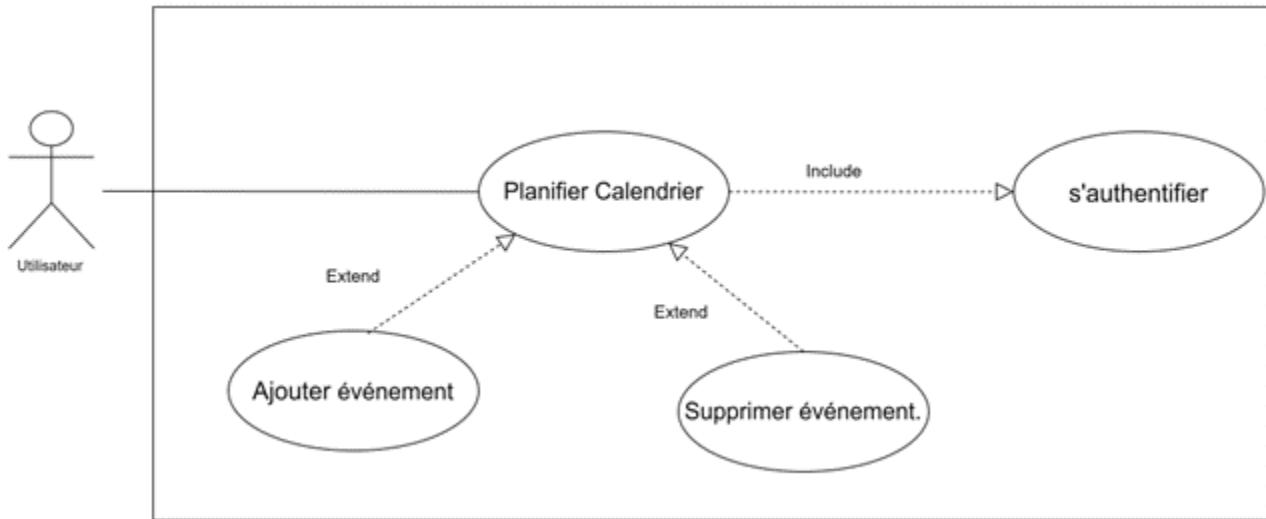


FIGURE 5.3 – cas d'utilisation Planification événement

description de cas d'utilisation : Ajouter événement

Ce tableau montre la description textuelle du cas d'utilisation « Ajouter événement »

Cas d'utilisation	Ajouter événement
Acteurs	Utilisation
Pré-condition	L' utilisateur doit être déjà authentifier
Scénario Normal	<ol style="list-style-type: none"> 1. L'utilisateur accède calendrier. 2. L'utilisateur sélectionne un jour pour l'événement. 3. L'utilisateur choisit l'option d'ajouter un événement. 4. Le système affiche le formulaire d'ajout. 5. L'utilisateur remplit le formulaire 6. L'utilisateur confirme l'ajout . 7. Le système enregistre les données .
Scénario alternatif	L'utilisateur ne confirme pas l'ajout .
Post condition	Événement est ajouté.

TABLE 5.6 – Description textuelle de cas d'utilisation ajouter événement

description de cas d'utilisation : Supprimer événement

Ce tableau montre la description textuelle du cas d'utilisation « Supprimer événement »

Cas d'utilisation	Supprimer événement.
Acteurs	Utilisation
Pré-condition	L'utilisateur doit être déjà authentifier
Scénario Normal	<ol style="list-style-type: none"> 1. L'utilisateur accède à calendrier. 2. L'utilisateur sélectionne l'événement pour supprimer. 3. L'utilisateur choisit l'option de supprimer événement . 4. Le système affiche un popup de suppression. 5. L'utilisateur confirmer la suppression . 6. Le systeme enregister les données .
Scénario alternatif	L'utilisateur ne confirme pas la suppression.
Post condition	Événement est supprimé

TABLE 5.7 – Description textuelle de cas d'utilisation supprimer événement

5.2.4 Conception

les diagrammes de séquence de Sprint 3

- Diagramme de séquence du cas d'utilisation : Ajouter vêtement

Le diagramme suivant décrit le cas d'utilisation du scénario "Ajouter vêtement", où l'utilisateur peut ajouter les images de leurs vêtements à la Garde-robe.

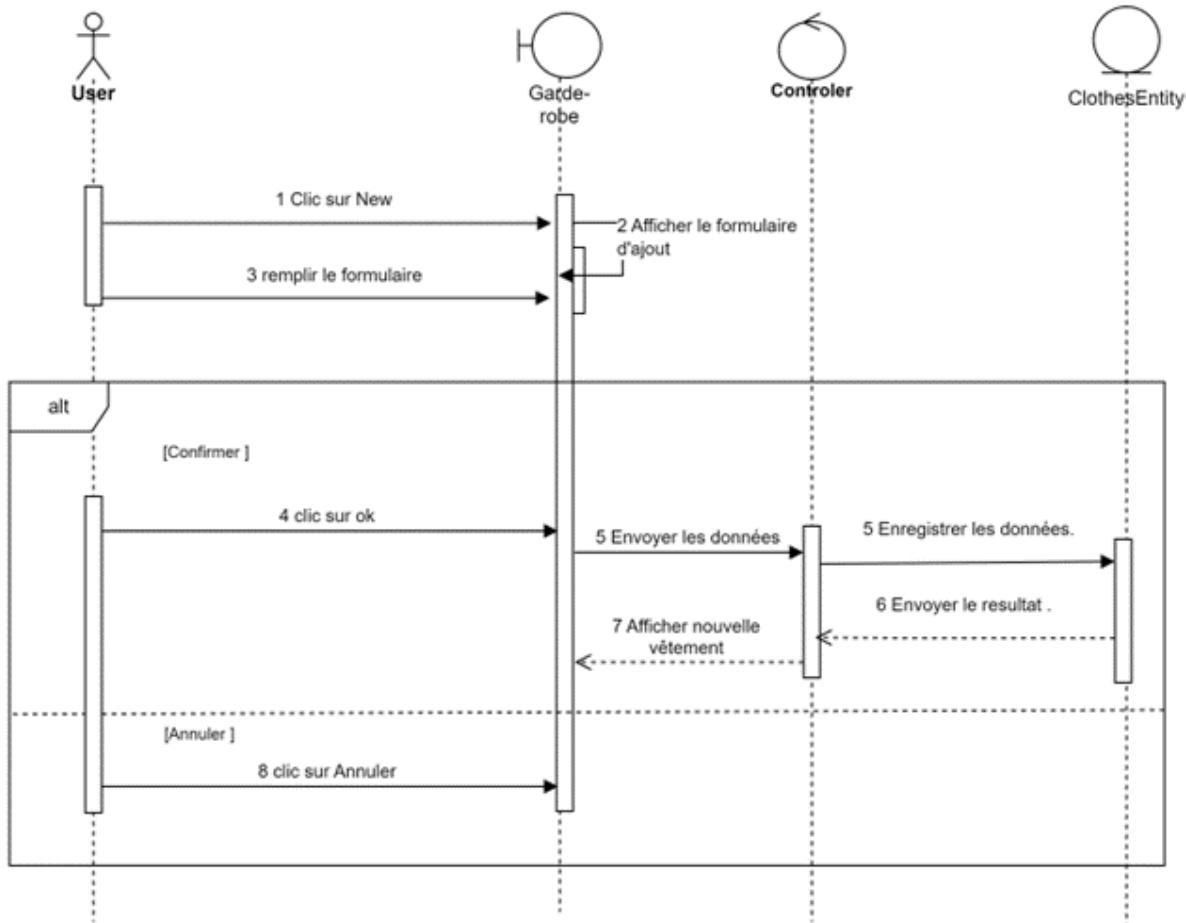


FIGURE 5.4 – Diagramme de séquence «Modifier vêtement »

•Diagramme de séquence du cas d'utilisation : Modifier vêtement

Le diagramme suivant décrit le cas d'utilisation du scénario "Modifier vêtement", où l'utilisateur peut modifier ses vêtements dans la Garde-robe.

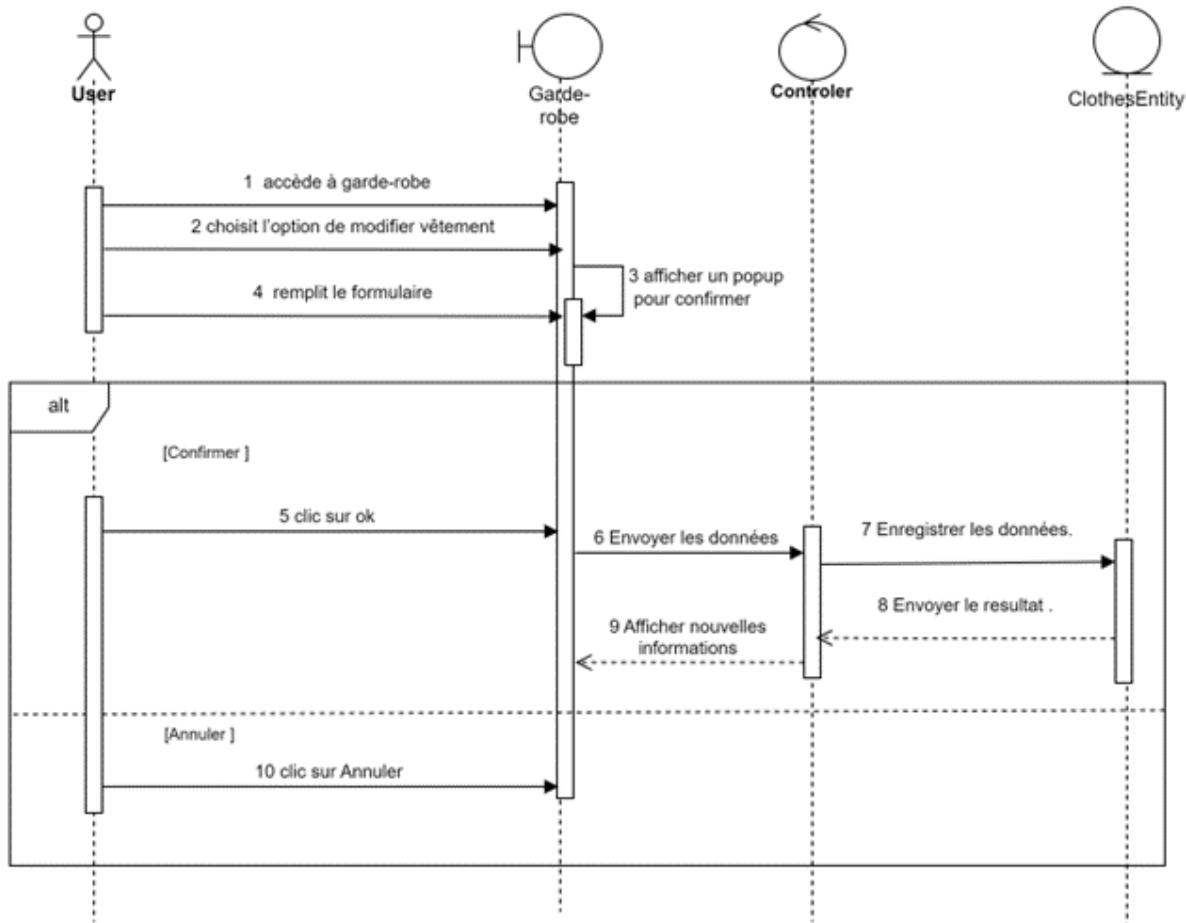


FIGURE 5.5 – Diagramme de séquence «Modifier vêtement »

- Diagramme de séquence du cas d'utilisation : Rechercher vêtement

Le diagramme suivant décrit le cas d'utilisation du scénario "Rechercher vêtement", où l'utilisateur peut rechercher les vêtements de sa Garde-robe.

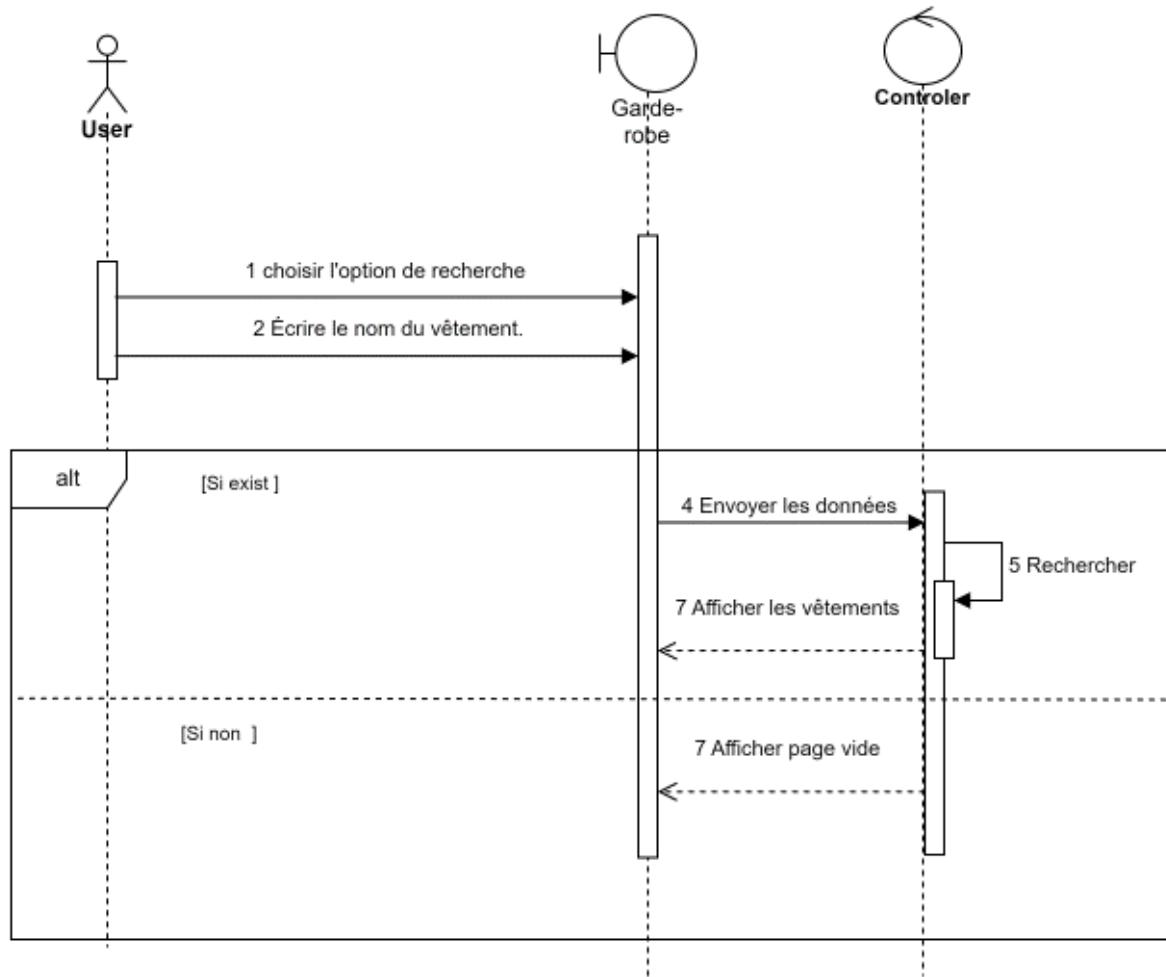


FIGURE 5.6 – Diagramme de séquence «Rechercher vêtement»

- Diagramme de séquence du cas d'utilisation :Supprimer événement

Le diagramme suivant décrit le cas d'utilisation du scénario "Supprimer événement", où l'utilisateur peut supprimer un événement dans calendrier.

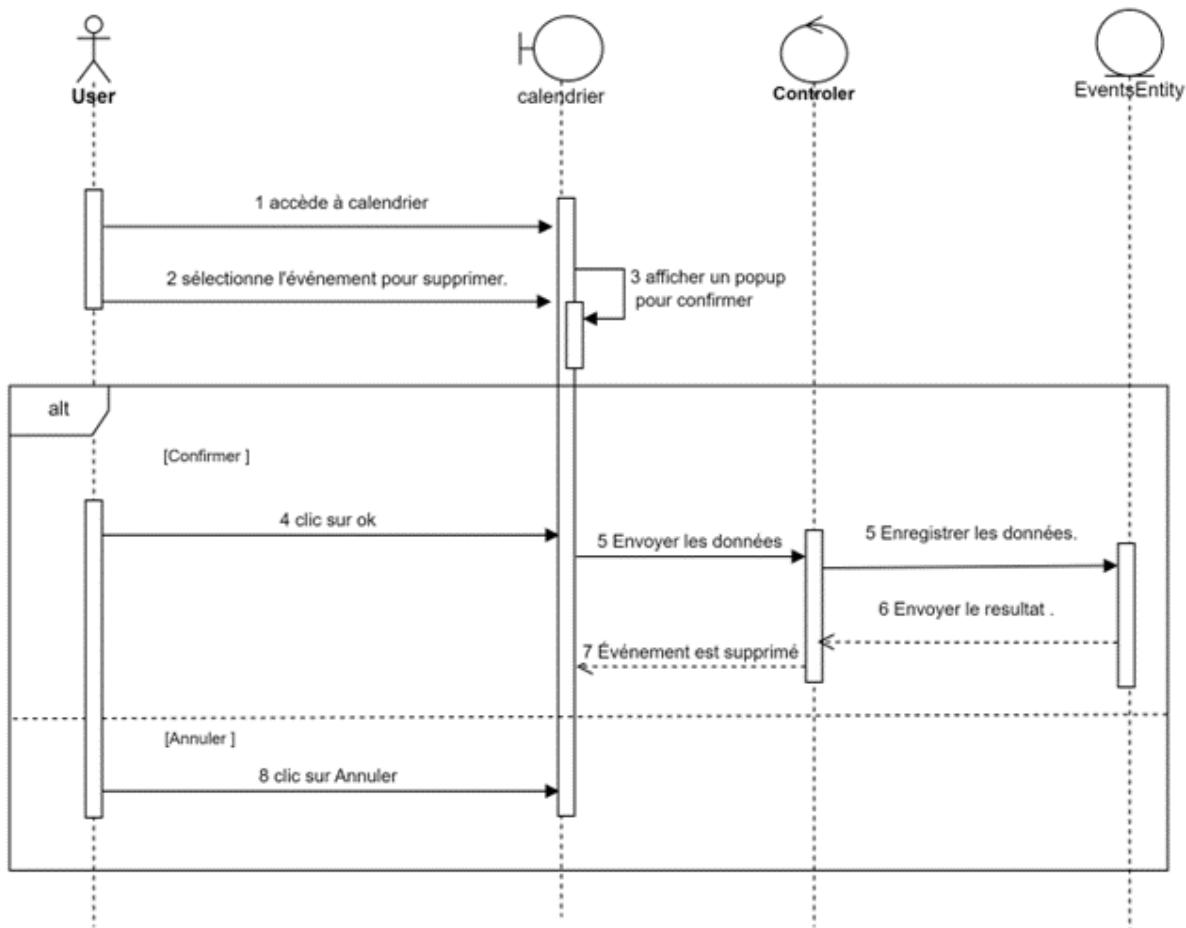


FIGURE 5.7 – Diagramme de séquence «Supprimer événement»

5.2.5 Consommation API météo

Dans cette section, nous aborderons les étapes suivies pour la consommation de l'API Weather (veuillez vous référer à l'Annexe pour plus d'informations).

5.2.6 Réalisation

- **Garde-Robe**

Cette figure présente l'interface graphique de Garde-Robe :

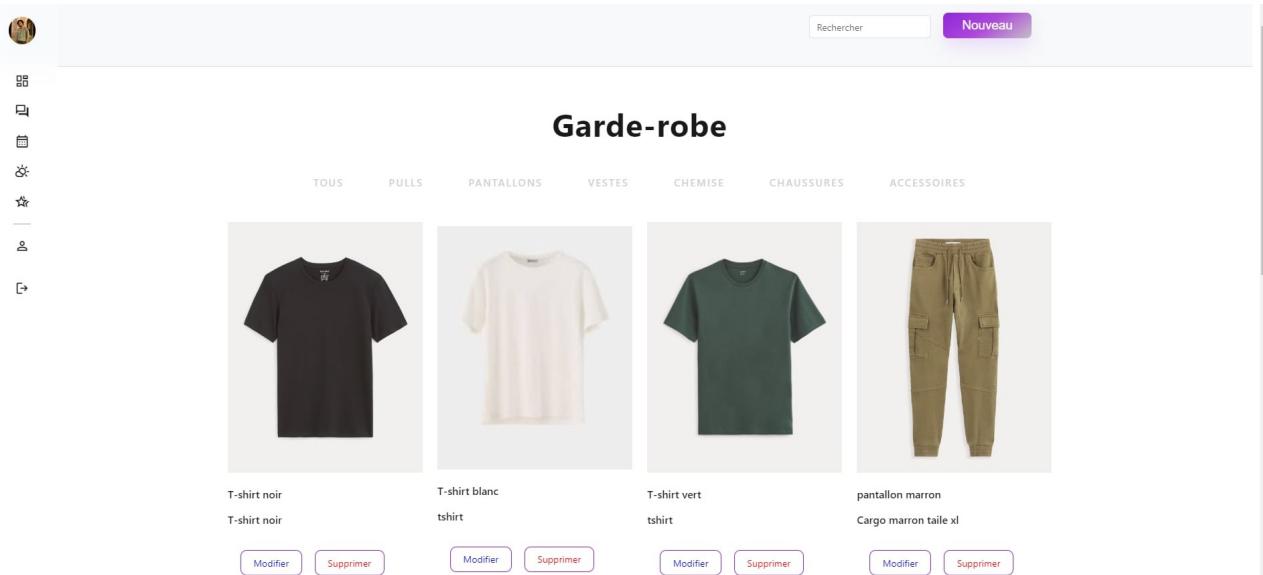


FIGURE 5.8 – Garde-Robe

- **Calendrier**

Cette figure présente l'interface graphique du calendrier :

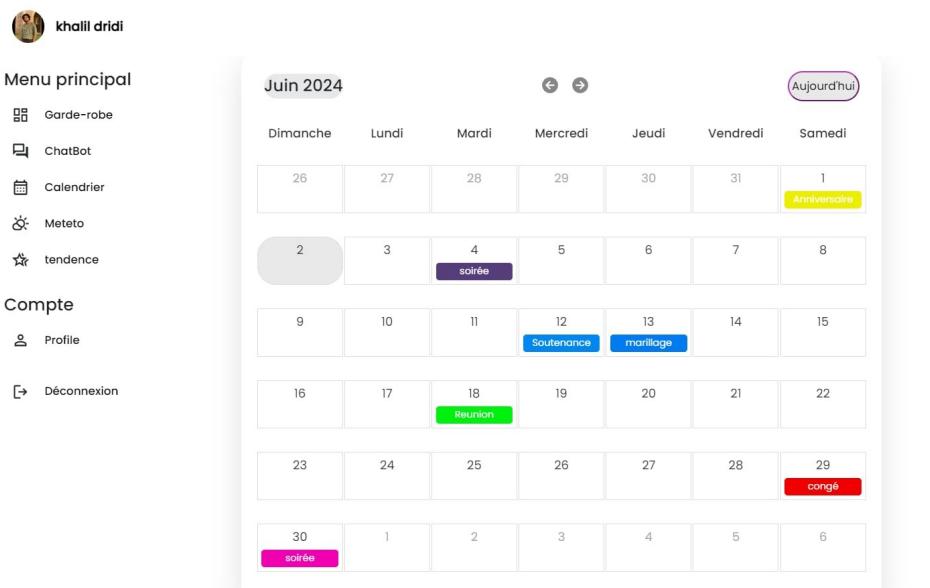


FIGURE 5.9 – calendrier

- **Météo**

Cette figure présente l'interface graphique de météo :

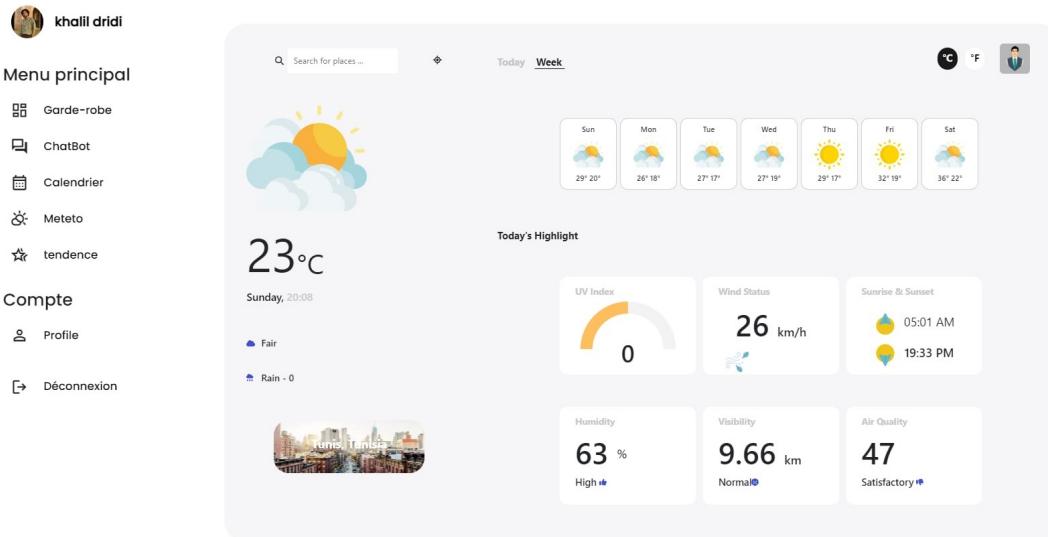


FIGURE 5.10 – Météo

5.2.7 Validation des Fonctionnalités du Sprint 3

Ce tableau contient les ensembles de scénarios de tests fonctionnels relatifs à ce sprint.

Cas de test	Démarche	Comportement attendu	Résultat
Gérer Garde-Robe	Introduire les informations pour gérer Garde-Robe	<ol style="list-style-type: none"> Ajouter vêtement Modifier vêtement Supprimer vêtement Rechercher vêtement 	Terminé
Planifier le calendrier	Introduire les informations pour gérer calendrier	<ol style="list-style-type: none"> Ajouter événement Supprimer événement 	Terminé
Consulter météo	Liez notre modèle météo à l'API Weather.	La fonctionnalité de météo est opérationnelle.	Terminé

TABLE 5.8 – Validation du sprint 3

5.3 Sprint 4

5.3.1 But de sprint

Après avoir expliqué les technologies de l'IA dans le chapitre 3, nous passons maintenant à la première version. Dans ce chapitre, nous présenterons les différentes étapes proposées dans le Arriéré.

Le Sprint Backlog doit inclure les tâches suivantes pour réaliser notre chatbot du Backlog :

- Collecter des données pour le modèle d'apprentissage automatique.
- Entraîner le modèle d'apprentissage automatique.
- Tester et évaluer la précision du modèle.
- Créer l'API pour le modèle.
- Tester et vérifier le bon fonctionnement de l'API.

5.3.2 Collecte de données

La collecte de données est une étape cruciale pour construire un chatbot efficace. Ce processus implique de rassembler des informations pertinentes que le chatbot peut utiliser pour comprendre les requêtes des utilisateurs et fournir des réponses précises. Les données collectées doivent être variées et complètes, couvrant différents sujets et scénarios auxquels le chatbot peut être confronté. De plus, il est important de veiller à ce que les données soient exactes, à jour et alignées sur les objectifs du chatbot. Grâce à une collecte de données efficace, un chatbot peut être formé pour mieux servir ses utilisateurs et offrir une expérience personnalisée et efficace.

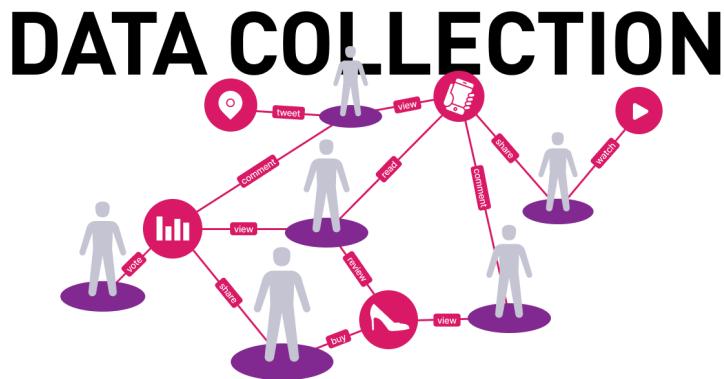


FIGURE 5.11 – Collecte de données [13]

Échantillon de données

Dans notre projet, nous utilisons presque tous les sites web de mode et de stylisme pour collecter les données que nous souhaitons utiliser afin de créer des modèles d'IA et de les publier.

Notre structure de données comprend trois éléments clés : une balise qui représente un certain sujet, des modèles qui sont les entrées de l'utilisateur, et des réponses prédéfinies que le modèle d'IA utilise pour y répondre.

En organisant les informations de cette manière, nous permettons à l'IA de répondre rapidement et avec précision à un large éventail de requêtes et d'interactions des utilisateurs.

La figure ci-dessous (Fig.) affiche les exemples des données collectées.

```
{
  "recommendations_couleurs": [
    {
      "tag": "recommendation_couleurs_vetements",
      "patterns": [
        "Est-ce que le bleu va bien avec le vert ?",
        "Puis-je porter un pantalon vert avec un pull bleu ?",
        "Le vert et le bleu vont-ils bien ensemble ?",
        "Le vert et le bleu sont-ils une bonne combinaison ?",
        "Est-ce que le vert peut être porté avec du bleu ?"
      ],
      "responses": "Oui, le bleu et le vert vont très bien ensemble. C'est une combinaison rafraîchissante et harmonieuse."
    }
  ]
}
```

FIGURE 5.12 – Collecte de données

5.3.3 Bibliothèques utilisées

Nous avons utilisé plusieurs bibliothèques pour créer notre modèle d'IA. Cela lui permet d'effectuer diverses fonctions, telles que la reconnaissance de formes et la génération de réponses. Voici quelques bibliothèques que nous utilisons :

NumPy :

NumPy est un package de traitement de tableaux à usage général. Il fournit un objet tableau multidimensionnel performant ainsi que des outils pour travailler avec ces tableaux. Il s'agit du paquet fondamental pour le calcul scientifique avec Python.

En plus de ses utilisations scientifiques évidentes, NumPy peut également servir de conteneur multidimensionnel efficace pour des données génériques.[40]



FIGURE 5.13 – NumPy Logo

Pickle :

Pickle en Python est principalement utilisé pour sérialiser et désérialiser une structure d'objet Python. En d'autres termes, il s'agit du processus de conversion d'un objet Python en flux d'octets afin de le stocker dans un fichier ou une base de données, de maintenir l'état du programme entre les sessions ou de transporter des données sur le réseau. Le flux d'octets désérialisé peut ensuite être utilisé pour recréer la hiérarchie d'objets d'origine en le désérialisant.[41]



FIGURE 5.14 – Pickle Logo

Keras :

Keras est une API d'apprentissage en profondeur développée par Google pour la mise en œuvre de réseaux de neurones. Elle est écrite en Python et facilite la mise en œuvre de ces réseaux. Elle prend également en charge différents calculs de réseaux neuronaux backend.[42]



FIGURE 5.15 – Keras Logo

TensorFlow :

TensorFlow est une bibliothèque open source développée principalement par Google pour les applications d'apprentissage en profondeur. Elle prend également en charge l'apprentissage automatique traditionnel.[43]



FIGURE 5.16 – TensorFlow Logo

5.3.4 Tokenisation (Tokenization)

La tokenisation est une étape cruciale dans tout pipeline de traitement du langage naturel (NLP). Elle consiste à décomposer le texte brut en morceaux plus petits appelés jetons, qui peuvent être des mots ou des phrases. La tokenisation des mots se fait généralement en divisant le texte en fonction des espaces, tandis que la tokenisation des phrases utilise des caractères tels que des points, des points d'exclamation et des caractères de nouvelle ligne pour séparer les phrases. La méthode appropriée de tokenisation dépend de la tâche à accomplir.

5.3.4.1 Importance de la tokenisation

Le sens d'une phrase découle des mots qu'elle contient. En analysant les mots d'un texte, nous pouvons interpréter précisément son sens. Nous pouvons également utiliser des outils statistiques et des méthodes pour obtenir des informations supplémentaires sur le texte, tels que le nombre de mots et la fréquence des mots. Ces techniques peuvent aider à identifier des modèles et des tendances dans le texte, ainsi qu'à générer des informations significatives qui éclairent la prise de décision et l'analyse.

5.3.4.2 Exemple de tokenisation

la figure suivante (Figure 5.7) montre un exemple de tokenisation

```
def clean_up_sentence(self,sentence):
    # tokenize the pattern
    sentence_words = nltk.word_tokenize(sentence)
    # stem each word
    stemmer = SnowballStemmer(self.language)
    sentence_words = [stemmer.stem(word.lower()) for word in sentence_words]
    return sentence_words
```

FIGURE 5.17 – Exemple de tokenisation

5.3.4.3 Gérer nos données avec la tokenisation

Essentiellement, dans l'exemple ci-dessous (Figure 5.8), nous effectuons la tokenisation des données de notre fichier JSON. Pour ce faire, nous ajoutons chaque phrase tokenisée à sa balise correspondante, comme le montre la sortie de la variable "documents".

```

def prepare_data_trainer(self, language):
    # import our chat-bot intents file
    with open("data/data.json") as json_data:
        intents = json.load(json_data)
    words = []
    classes = []
    documents = []
    ignore_words = ['?', '!', '.', ',', ':']
    tags = intents.keys()
    for t in tags:
        for intent in intents[t]:
            for pattern in intent['patterns']:
                # tokenize each word in the sentence
                pattern = str(pattern).replace("-", " ")
                pattern = str(pattern).replace("ç", "c")
                pattern = str(pattern).replace("é", "e")
                pattern = str(pattern).replace("â", "a")
                print(pattern)
                # filter words of stop_words
                pattern = self.filter_stopwords(language, pattern)
                print("filter: ", pattern)
                w = nltk.word_tokenize(pattern)
                # add to our words list
                words.extend(w)
                #print("words: ", words)
                # add to documents in our corpus
                if(w):
                    documents.append((w, intent['tag']))

```

FIGURE 5.18 – Exemple de tokenisation

5.3.5 Lemmatisation (Lemmatization)

La Lemmatization regroupe les différentes formes fléchies d'un mot. Elle est utilisée dans le traitement du langage naturel (NLP), la linguistique informatique et le développement de chatbots. En reliant les différentes formes d'un mot à leurs significations, la lemmatization améliore la précision et l'efficacité des outils de PNL, des chatbots et des moteurs de recherche. Ce processus rationalise l'analyse des données textuelles et améliore la qualité globale des applications de traitement automatisé du langage.

```

# stem and lower each word and remove duplicates
stemmer = SnowballStemmer(language)
words = [stemmer.stem(w.lower()) for w in words if w not in ignore_words]

```

FIGURE 5.19 – Exemple de Lemmatization

5.3.5.1 Sauvegarde des mots et des classes lemmatisés

En ce qui concerne la PNL, la lemmatization et le picking sont souvent étroitement liés. Après avoir lemmatisé un vaste ensemble de données, il est judicieux de récupérer le fichier résultant afin de pouvoir le charger rapidement pour des analyses ou des formations futures (Figure 5.10).

```
pickle.dump({'words': words, 'classes': classes, 'train_x': train_x, 'train_y': train_y},
```

FIGURE 5.20 – Exemple de Lemmatization

Extraire de bag des mots

bag de mots est une méthode simple et populaire pour représenter des données textuelles dans le domaine de l'apprentissage automatique.

Dans un modèle de bag de mots, le texte est d'abord symbolisé en mots ou en phrases individuelles, puis chaque mot est compté pour créer une représentation du texte sous forme d'histogramme. Seule la fréquence des mots est prise en compte, et non leur ordre.

Par exemple, si nous avons la phrase "Le chien court dans le jardin.", la représentation du bag de mots serait une liste des mots uniques présents dans la phrase, ainsi que leur fréquence :

```
Le : 2
chien : 1
dans : 1
jardin : 1
```

Cette représentation peut être utilisée comme entrée pour des algorithmes d'apprentissage automatique, qui peuvent apprendre à classer ou prédire en fonction de la fréquence des mots dans le texte. Cependant, le modèle du bag de mots ne capture pas le contexte ni le sens des mots, ce qui peut entraîner une perte d'informations importantes (Figure 5.11).

```
# training set, bag of words for each sentence
for doc in documents:
    # initialize our bag of words
    bag = []
    # list of tokenized words for the pattern
    pattern_words = doc[0]
    # stem each word
    pattern_words = [stemmer.stem(word.lower()) for word in pattern_words]
    # create our bag of words array
    for w in words:
        bag.append(1) if w in pattern_words else bag.append(0)

    # output is a '0' for each tag and '1' for current tag
    output_row = list(output_empty)
    output_row[classes.index(doc[1])] = 1

    training.append([bag, output_row])
```

FIGURE 5.21 – Collecter de bag des mot

5.3.5.2 Brassage des données(Shuffling)

Shuffling des données est une étape essentielle pour garantir l'impartialité des modèles d'apprentissage automatique envers tout ordre spécifique des données d'entrée lors de la formation. L'omission du mélange des données présente un risque, car le modèle pourrait apprendre à bien

fonctionner sur un sous-ensemble spécifique de données, tout en étant moins performant sur d'autres. Ce problème survient souvent lorsque les données suivent un ordre naturel ou sont regroupées. En mélangeant les données avant la formation, on peut atténuer ces problèmes et s'assurer que le modèle est formé sur une gamme diversifiée de données d'entrée, ce qui conduit à de meilleures performances et capacités de généralisation (Figure 5.12).

```
# shuffle our features and turn into np.array
random.shuffle(training)
training = np.array(training)
# create train and test lists
train_x = list(training[:, 0])
train_y = list(training[:, 1])
```

FIGURE 5.22 – Brassage des données

5.3.6 Entraînement de modèle (Model training)

Ce code construit et entraîne un modèle de réseau neuronal feedforward à l'aide de l'API TensorFlow avec l'extension TFlearn. Nous commençons par définir la structure du réseau en spécifiant les couches de neurones, puis nous entraînons le modèle en utilisant la descente de gradient stochastique (SGD). Si l'arrêt anticipé ou le nombre maximal d'époques est atteint, le modèle entraîné est sauvegardé pour une utilisation ultérieure.

```
# training set, bag of words for each sentence
for doc in documents:
    # initialize our bag of words
    bag = []
    # list of tokenized words for the pattern
    pattern_words = doc[0]
    # stem each word
    pattern_words = [stemmer.stem(word.lower()) for word in pattern_words]
    # create our bag of words array
    for w in words:
        bag.append(1) if w in pattern_words else bag.append(0)

    # output is a '0' for each tag and '1' for current tag
    output_row = list(output_empty)
    output_row[classes.index(doc[1])] = 1

    training.append([bag, output_row])
```

FIGURE 5.23 – Code de model training

5.3.7 Postman : Test de l'API

Dans cette figure , nous allons tester notre API créée sur Postman

The screenshot shows the Postman interface with a POST request to `http://localhost:5000/reply`. The request body is set to `JSON` and contains the following JSON:

```

1: {
2:   "question": "Le vert et le bleu vont-ils bien ensemble ?"
3:
4:
5:
}
  
```

The response tab shows a `200 OK` status with a response message in JSON:

```

1: {
2:   "message": "Oui, le bleu et le vert vont très bien ensemble. C'est une combinaison rafraîchissante et harmonieuse."
3:
}
  
```

FIGURE 5.24 – Test de l'API

5.3.8 Réalisation

- Chat Bot

Cette figure présente l'interface graphique de Chat Bot :

The screenshot shows a mobile application interface. On the left is a sidebar menu with the following items:

- Menu principal
- Garde-robe
- ChatBot
- Calendrier
- Meteo
- Tendance
- Compte
- Profile
- Déconnexion

On the right is a main screen titled "Chat". It features a message input field with placeholder text "Type your message here..." and a purple "Envoyer" button.

FIGURE 5.25 – Chat Bot

Conclusion

Dans ce chapitre, la première partie traite du développement de la garde-robe en incluant les aspects liés au calendrier et à la météo.

Dans la deuxième partie, nous présentons une méthodologie détaillée pour créer un chatbot

NLP (Traitement du Langage Naturel). Nous commençons par collecter les données, puis nous les préparons en les tokenisant et en les lemmatisant afin d'améliorer la précision de notre modèle. Après avoir entraîné le modèle, nous le soumettons à des tests.

Conclusion générale

Dans le cadre de notre projet de fin d'études, nous avons conçu et développé une application web de garde-robe virtuelle avec un chatbot conseiller et diverses fonctionnalités pour aider l'utilisateur à choisir ses vêtements et à discuter avec le chatbot pour recevoir des conseils.

Ce rapport détaille toutes les étapes que nous avons franchies pour atteindre les résultats attendus. Nous avons commencé par comprendre le contexte général du projet et identifier les différents besoins du futur système.

Ensuite, nous avons préparé un Product Backlog en respectant les priorités des besoins suite à une discussion avec l'équipe Scrum.

Malgré les contraintes de temps et les difficultés techniques, principalement liées à l'exécution des processus, nous avons réussi à compléter toutes les tâches.

De plus, ce projet nous a permis d'approfondir nos connaissances en gestion de projet et en bonnes pratiques, car nous avons pu organiser son déroulement dès le début.

Enfin, maintenant que le projet est terminé, nous espérons vivement qu'il sera couronné de progrès et de succès. Nous souhaitons également la satisfaction des membres du jury.

Références

- [1] <https://www.beecoders.tn/> , Consulté le 17-02-2024.
- [2] <https://www.marine-guyot.ovh/methode-agile-ou-classique/>, Consulté le 17-02-2024.
- [3] <https://www.heflo.com/fr/blog/agile/methode-agile-scrum/>, Consulté le 19-02-2024.
- [4] <https://fr.wikipedia.org/wiki/Mod>
- [5] <https://www3.technologyevaluation.com/research/article/how-ai-is-transforming-erp.html>, Consulté le 22-02-2024.
- [6] <https://www.bba.ca/ca-fr/publications/lapprentissage-automatique-un-atout-puissant-pour-une-meilleure-exploitation-de-vos-donnes> , Consulté le 23-02-2024.
- [7] <https://culturesciencesphysique.ens-lyon.fr/ressource/IA-apprentissage-Rousseau.xml>, Consulté le 26-02-2024.
- [8] <https://www.redbubble.com/fr/i/poster/Intelligence-Artificielle-Machine-Learning-Deep-Learning-par-Phantasmal-2-do/86985077.LVTDI>, Consulté le 25-02-2024.
- [9] <https://www.futura-sciences.com/tech/definitions/intelligence-artificielle-deep-learning-17262/>, Consulté le 30-02-2024.
- [10] <https://www.altoros.com/blog/natural-language-processing-saves-businesses-millions-of-dollars/> , Consulté le 17-03-2024.
- [11] <https://www.engati.com/glossary/natural-language-processing> , Consulté le 19-03-2024.
- [12] <https://www.engati.com/glossary/natural-language-processing> , Consulté le 19-03-2024.
- [13] <https://brocoders.com/blog/data-collection-methods-definition-types-and-tools/>, Consulté le 19-04-2024.
- [14] <https://www.atlassian.com/fr/agile/scrum> : :text=Scrum
- [15] <https://www.appvizer.fr/magazine/operations/gestion-de-projet/equipe-scrum>, Consulté le 19-02-2024.
- [16] <https://fr.wikipedia.org/wiki/Mod>
- [17] <https://code.visualstudio.com/brand> , Consulté le 26-03-2024.
- [18] <https://www.jetbrains.com/idea/?var=1> , Consulté le 26-03-2024.
- [19] <https://fr.wikipedia.org/wiki/PyCharm>, Consulté le 19-02-2024.
- [20] <https://app.diagrams.net/> , Consulté le 26-03-2024.
- [21] <https://balsamiq.com/> , Consulté le 26-03-2024.
- [22] <https://fr.overleaf.com/> , Consulté le 26-03-2024.

- [23] <https://angular.io/> , Consulté le 26-03-2024.
- [24] <https://spring.io/projects/spring-boot> , Consulté le 26-03-2024.
- [25] [https://fr.wikipedia.org/wiki/Python_\(langage\)](https://fr.wikipedia.org/wiki/Python_(langage)), Consulté 29 – 03 – 2024.
- [26] <https://www.netapp.com/fr/artificial-intelligence/what-is-artificial-intelligence/>, Consulté le 22-02-2024.
- [27] <https://www.ibm.com/fr-fr/topics/machine-learning> , Consulté le 22-02-2024.
- [28] <https://www.bba.ca/ca-fr/publications/lapprentissage-automatique-un-atout-puissant-pour-une-meilleure-exploitation-de-vos-donnes> , Consulté le 24-02-2024.
- [29] <https://www.bba.ca/ca-fr/publications/lapprentissage-automatique-un-atout-puissant-pour-une-meilleure-exploitation-de-vos-donnes> , Consulté le 24-02-2024.
- [30] <https://datascientest.com/machine-learning-tout-savoir> , Consulté le 24-02-2024.
- [31] <https://mailchimp.com/fr/resources/semi-supervised-learning/> : :text=L'apprentissage
- [32] https://iset.uvt.tn/pluginfile.php/2795822/mod_resource/content/5/Chapitre
- [33] <https://netapp.com/fr/artificial-intelligence/what-is-deep-learning/> : :text=les
- [34] <https://aws.amazon.com/fr/what-is/deep-learning/> : :text=Les
- [35] <https://mobiskill.fr/blog/conseils-emploi-tech/quels-sont-les-algorithmes-de-deep-learning/> , Consulté le 26-03-2024.
- [36] <https://aws.amazon.com/fr/what-is/nlp/> , Consulté le 17-03-2024.
- [37] <https://www.engati.com/glossary/natural-language-processing> , Consulté le 17-03-2024.
- [38] <https://www.engati.com/glossary/natural-language-processing> , Consulté le 19-03-2024.
- [39] <https://www.engati.com/glossary/natural-language-processing> , Consulté le 19-03-2024.
- [40] <https://www.geeksforgeeks.org/python-numpy/> , Consulté le 29-03-2024.
- [41] <https://www.geeksforgeeks.org/understanding-python-pickling-example/> , Consulté le 29-03-2024.
- [42] <https://www.simplilearn.com/tutorials/deep-learning-tutorial/what-is-keras> , Consulté le 29-03-2024.
- [43] <https://www.simplilearn.com/tutorials/deep-learning-tutorial/what-is-tensorflow> , Consulté le 29-03-2024.

Annexe

Consommation d'une API météo.

Dans notre application, nous avons une fonctionnalité météo qui permet à l'utilisateur de découvrir les prévisions de la semaine afin de l'aider à choisir ses vêtements.

Pour activer la fonction météo, utilisez l'API Weather. Suivez les étapes suivantes :

- En premier lieu, nous choisissons le site web de l'API rapide (C'est un site qui permet le développement d'une API météo gratuite pendant un mois) pour consommer notre API weather.

Cette figure(Figure 6.1) montre le site de l'api de notre application :

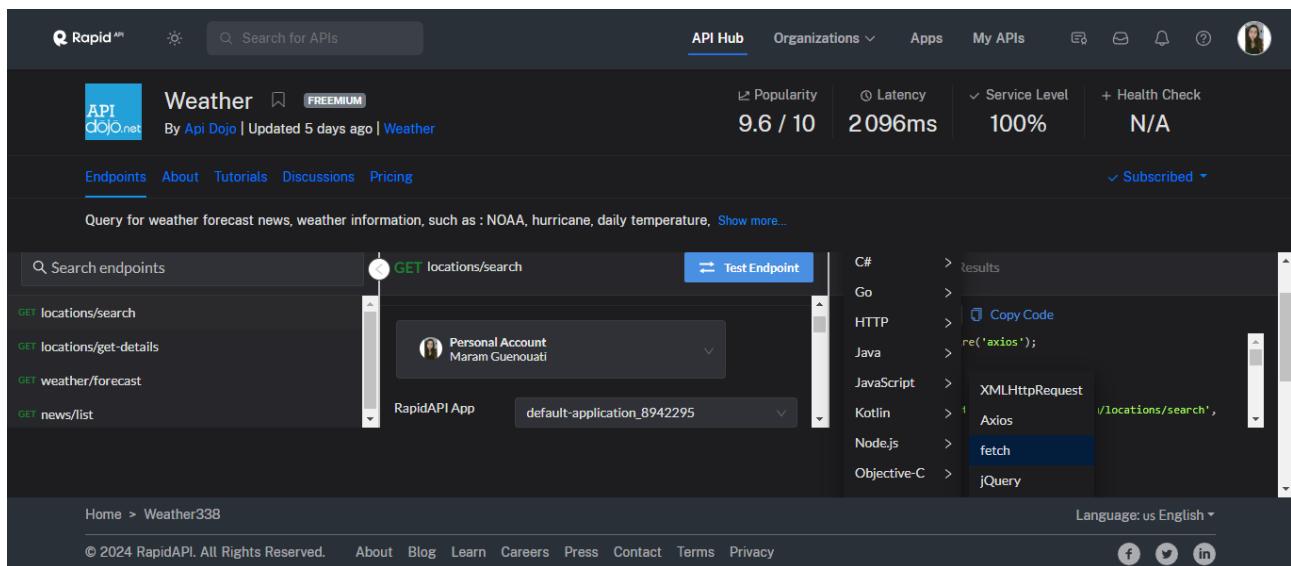


FIGURE 5.26 – Weather By Api Dojo

- On change la forme du code existant en JavaScript en choisissant "Fetch".

The screenshot shows the RapidAPI API Hub interface. At the top, there's a search bar and navigation links for 'API Hub', 'Organizations', 'Apps', 'My APIs', and user profile. Below that, the 'Weather' API by 'Api Dojo' is displayed, showing a popularity score of 9.6 / 10, latency of 2096ms, and service level of 100%. The 'Endpoints' tab is selected, listing four endpoints: 'GET locations/search', 'GET locations/get-details', 'GET weather/forecast', and 'GET news/list'. The 'GET locations/search' endpoint is highlighted. To the right, a sidebar shows code snippets for C#, Go, HTTP, Java, JavaScript, Kotlin, Node.js, and Objective-C. Under 'JavaScript', the 'fetch' method is highlighted. At the bottom, there's a footer with social media links and a copyright notice.

FIGURE 5.27 – Weather By Api Dojo

- Tout d'abord, cliquez sur "locations/search" pour indiquer notre zone, puis recherchez notre pays

This screenshot is similar to Figure 5.27 but includes handwritten annotations. The 'GET locations/search' endpoint is selected. A dropdown menu for the 'query' parameter is open, showing the value 'Tunis'. Two handwritten numbers are overlaid: '1' points to the arrow of the dropdown menu, and '2' points to the input field where 'Tunis' is typed. The rest of the interface is identical to Figure 5.27.

FIGURE 5.28 – Weather By Api Dojo

- Voilà tout le résultat de "Tunis".

The screenshot shows the Weather By Api Dojo interface. On the left, there's a sidebar with endpoints: GET locations/search, GET locations/get-details, GET weather/forecast, and GET news/list. The main area shows a GET request for 'locations/search'. The 'X-RapidAPI-Host' field is set to 'weather338.p.rapidapi.com'. The 'query' parameter is set to 'Tunis', and the 'language' parameter is set to 'en-US'. A 'Test Endpoint' button is visible. On the right, the results tab is selected, showing a 200 Success status. The response body is displayed as JSON, with an arrow pointing to the first item in the 'address' array: '0: "Tunis, Tunisia"'.

```

{
  "location": {
    "name": "Tunis, Tunisia",
    "lat": 36.775,
    "lon": 10.195
  },
  "address": [
    "0: \"Tunis, Tunisia\"",
    "1: \"Tunis, Texas, United States\"",
    "2: \"Tunis, New Mexico, United States\"",
    "3: \"Tuniszki, Gołdap County, Warmian-Masurian Voivodeship, Poland\"",
    "4: \"Tunis, Cochrane, Ontario, Canada\"",
    "5: \"Tunis Mills, Maryland, United States\"",
    "6: \"Tunis, Maryland, United States\"",
    "7: \"Tunis, North Carolina, United States\"",
    "8: \"Tunis, Virginia, United States\"",
    "9: \"Tunis, Michigan, United States\""
  ],
  "adminDistrict": [],
  "adminDistrictCode": [],
  "city": []
}

```

FIGURE 5.29 – Weather By Api Dojo

- Ensuite, vérifiez le TimeZone

This screenshot is identical to Figure 5.29, showing the Weather By Api Dojo interface with the same endpoint, parameters, and JSON response. An arrow points to the first item in the 'address' array: '0: "Tunis, Tunisia"'.

```

{
  "location": {
    "name": "Tunis, Tunisia",
    "lat": 36.775,
    "lon": 10.195
  },
  "address": [
    "0: \"Tunis, Tunisia\"",
    "1: \"Tunis, Texas, United States\"",
    "2: \"Tunis, New Mexico, United States\"",
    "3: \"Tuniszki, Gołdap County, Warmian-Masurian Voivodeship, Poland\"",
    "4: \"Tunis, Cochrane, Ontario, Canada\"",
    "5: \"Tunis Mills, Maryland, United States\"",
    "6: \"Tunis, Maryland, United States\"",
    "7: \"Tunis, North Carolina, United States\"",
    "8: \"Tunis, Virginia, United States\"",
    "9: \"Tunis, Michigan, United States\""
  ],
  "adminDistrict": [],
  "adminDistrictCode": [],
  "city": []
}

```

FIGURE 5.30 – Weather By Api Dojo

- L'étape suivante consiste à choisir la latitude et la longitude correspondant à notre pays. Et cliquez sur weather/forecast et écrire la nouvelle latitude et longitude.

The screenshot shows the API Dojo interface with the 'V1 (Current)' dropdown selected. On the left, a sidebar lists endpoints: GET locations/search, GET locations/get-details, GET weather/forecast, and GET news/list. The main area displays the 'GET locations/search' endpoint. It has a summary table with columns 'METHOD' (GET), 'NAME' (locations/search), and 'DESCRIPTION'. A 'Test Endpoint' button is present. Below this, the 'X-RapidAPI-Host' field is set to 'weather338.p.rapidapi.com'. The 'Required Parameters' section contains 'query' (Tunis) and 'language' (en-US). The 'Optional Parameters' section is collapsed. On the right, the 'Results' tab is active, showing a 200 Success status. The 'Body' tab is selected, displaying a list of items under 'latitude': 3: 36.793, 4: 36.82, 5: 36.84, 6: 36.86, 7: 36.88, 8: 36.9, 9: 36.92, and 10: 36.94. An arrow points to the value '3: 36.793'. The 'Headers' tab is also visible.

FIGURE 5.31 – Weather By Api Dojo

This screenshot is identical to Figure 5.31, showing the 'locations/search' endpoint for Tunis. The 'Results' tab is active, displaying the same list of 'latitude' items. An arrow points to the value '3: 36.793'. Another arrow points to the value '0: 10.196' under the 'longitude' section, which is located further down the results page.

FIGURE 5.32 – Weather By Api Dojo

- Voilà

The screenshot shows the 'weather/forecast' endpoint configuration in the Weather By Api Dojo interface. The endpoint is a GET request with the following parameters:

- date**: 20200622 (String, REQUIRED)
- latitude**: 37.765 (Number, REQUIRED)
- longitude**: -122.463 (Number, REQUIRED)
- language**: en-US (Optional Parameter)

The 'weather/forecast' endpoint is highlighted in the search bar on the left.

```

const url = 'https://weather338.p.rapidapi.com/weather/Forecast?date=20200622&latitude=37.765&longitude=-122.463&language=en-US&units=m'
const options = {
  method: 'GET',
  headers: {
    'X-RapidAPI-Key': 'cfd293aeefefaf16a6d943bfaf72p3h5d5junc7a3le4che7e',
    'X-RapidAPI-Host': 'weather338.p.rapidapi.com'
  }
}

try {
  const response = await fetch(url, options)
  const result = await response.text()
  console.log(result)
} catch (error) {
  console.error(error)
}

```

FIGURE 5.33 – Weather By Api Dojo

The screenshot shows the 'weather/forecast' endpoint configuration in the Weather By Api Dojo interface. The endpoint is a GET request with the following parameters:

- date**: 20200622 (String, REQUIRED)
- latitude**: 36.8 (Number, REQUIRED)
- longitude**: 10.186 (Number, REQUIRED)
- language**: en-US (Optional Parameter)

Arrows point to the latitude and longitude input fields.

```

const url = 'https://weather338.p.rapidapi.com/weather/Forecast?date=20200622&latitude=36.8&longitude=10.186&language=en-US&units=m'
const options = {
  method: 'GET',
  headers: {
    'X-RapidAPI-Key': 'cfd293aeefefaf16a6d943bfaf72p3h5d5junc7a3le4che7e',
    'X-RapidAPI-Host': 'weather338.p.rapidapi.com'
  }
}

try {
  const response = await fetch(url, options)
  const result = await response.text()
  console.log(result)
} catch (error) {
  console.error(error)
}

```

FIGURE 5.34 – Weather By Api Dojo

- On clique sur "locations search" et on copie le code existant.

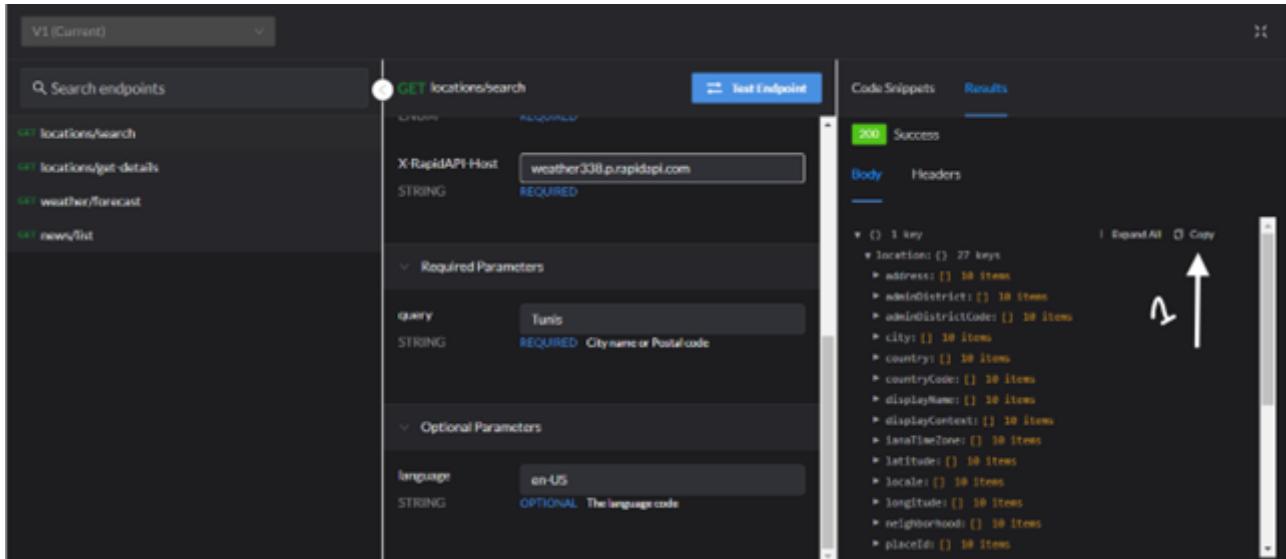


FIGURE 5.35 – Weather By Api Dojo

- Et sur le site Transform tools, on transforme le code de weather de json en TypeScript et on le met dans notre projet Angular.

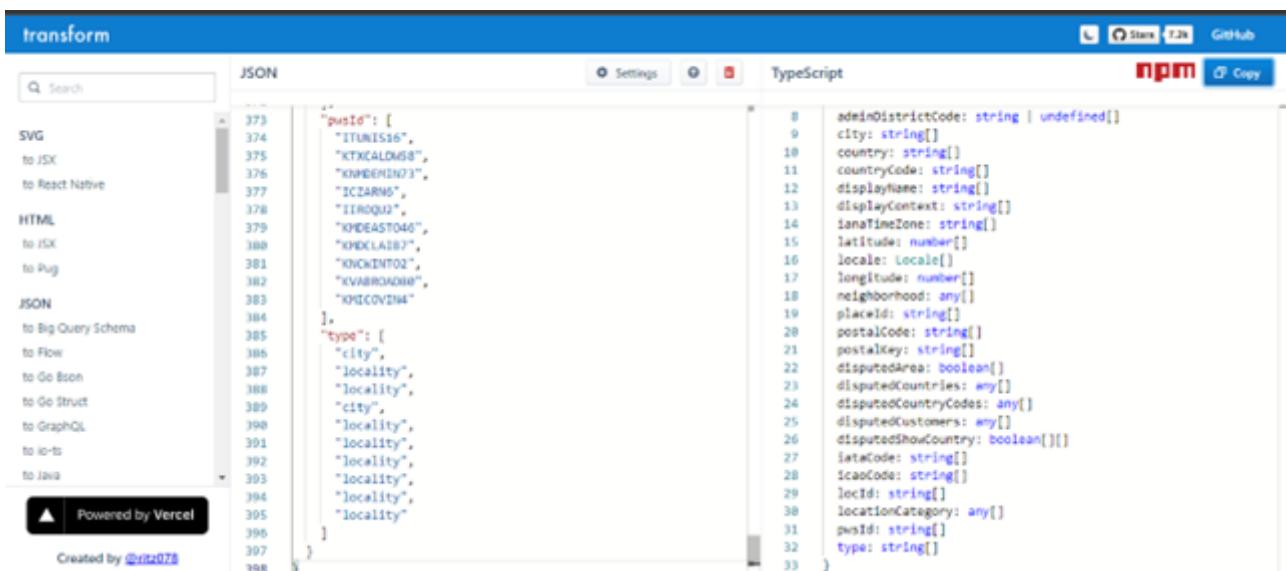
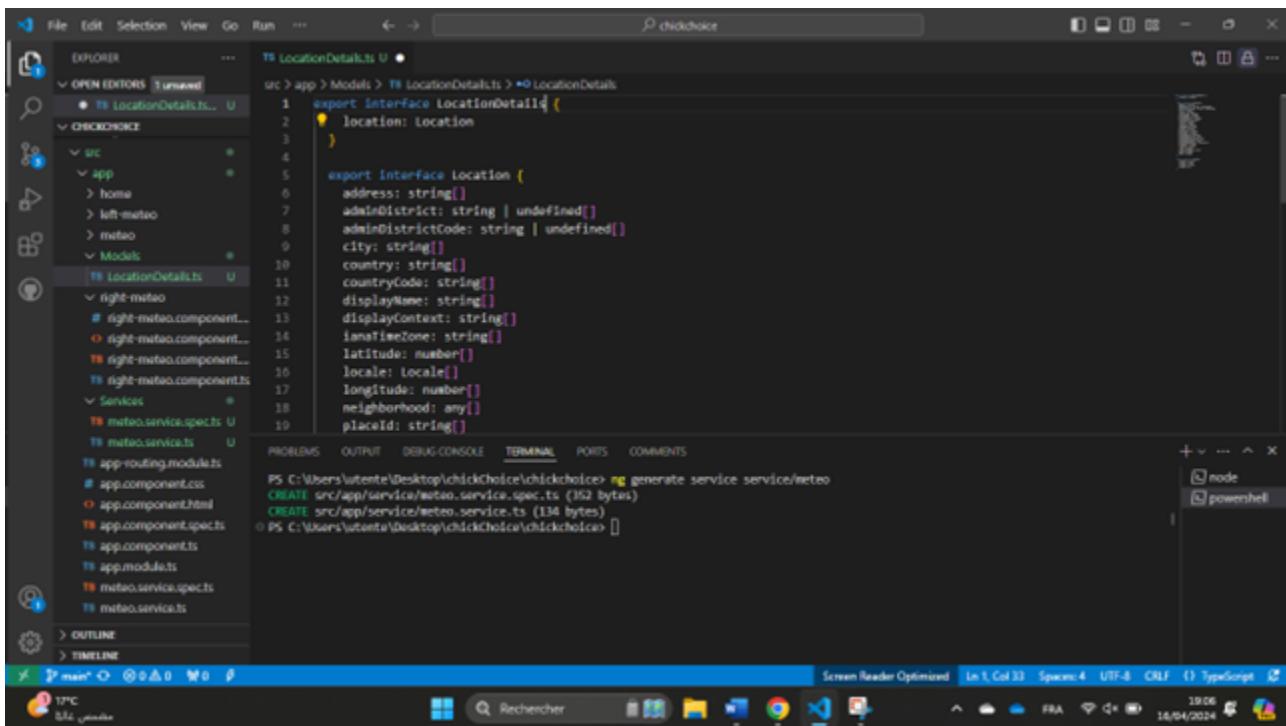


FIGURE 5.36 – Transform tools



- Et on revient au weather/forecast, et pour faire la même chose, on copie le code et on le transfère ensuite pour le mettre dans notre projet.

The screenshot shows the Microsoft Azure API Management interface. On the left, a sidebar lists several endpoints: `GET locations/search`, `GET locations/get-details`, `GET weather/forecast`, and `GET news/list`. The main area displays the `GET weather/forecast` endpoint details. It includes a 'Required Parameters' section with three fields: `date` (set to `20200622`), `latitude` (set to `36.8`), and `longitude` (set to `101.86`). Below this is an 'Optional Parameters' section. To the right, there's a 'Test Endpoint' button, a 'Code Snippets' tab, and a 'Results' tab. Under the 'Results' tab, a successful response is shown with a status of `200 Success`. The response body is a JSON object containing various weather forecast details like daily summaries, hourly forecasts, and precipitation levels.

FIGURE 5.37 – Weather By Api Dojo

```

1 export interface meteoDetails {
2   id: string
3   "v3-wx-Forecast-daily-15day-cognitiveHealth": any
4   "v3-wx-conditions-historical-dailysummary-30day": V3wxConditionsHistoricalDailysummary30day
5   "v3-wx-Forecast-hourly-10day": V3wxForecastHourly10day
6   "v3-wx-lightning-15minute-mobile": any
7   "v3-wx-skiconditions": V3wxSkiconditions
8   "v3-wx-globalAirQuality": V3wxGlobalAirQuality
9   "v3-wx-observations-current": V3wxObservationsCurrent
10  "v3-wx-forecast-daily-15day": V3wxForecastDaily15day
11  vtilddbreathingDaypart: VtilddBreathingDaypart
12  vt1madir: Vt1madir
13  vt1nowcast: Vt1nowcast
14  v2idxRunDaypart5: V2idxRunDaypart5
15  vtipastpollin: any
16  vtcontentMode: VtcontentMode
17  "v3-location-point": V3locationPoint
18  vtipollenobs: any
19  vticurrentTides: any

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

PS C:\Users\Intente\Desktop\chickchoice> ng generate service service/meteo
CREATE src/app/service/meteo.service.spec.ts (162 bytes)
CREATE src/app/service/meteo.service.ts (134 bytes)
PS C:\Users\Intente\Desktop\chickchoice> []

- Finallement copier l'api et le mettre dans notre projet dans fichier Environment .

The screenshot shows a code editor interface with two tabs: "Code Snippets" and "Results". The "Code Snippets" tab is active. A dropdown menu labeled "(Node.js) Axios" is open, and a "Copy Code" button is visible. The code itself is a Node.js script using the Axios library to make a GET request to a weather API. It includes options for method, URL, parameters (query and language), headers (X-RapidAPI-Key and X-RapidAPI-Host), and a try-catch block for handling errors.

```

const axios = require('axios');

const options = {
  method: 'GET',
  url: 'https://weather338.p.rapidapi.com/locations/search',
  params: {
    query: 'san fran',
    language: 'en-US'
  },
  headers: {
    'X-RapidAPI-Key': '22143f76c6msh71d6f8ed5e3cdadp1f2bb7jsn71e18c067282',
    'X-RapidAPI-Host': 'weather338.p.rapidapi.com'
  }
};

try {
  const response = await axios.request(options);
  console.log(response.data);
} catch (error) {
  console.error(error);
}

```

FIGURE 5.38 – Weather By Api Dojo

The screenshot shows a code editor interface displaying the `EnvironmentVariables.ts` file. The file is located under `src > app > Environment`. It defines an `EnvironmentalVariables` object with properties for `production`, `weatherApiLocationBaseURL`, `weatherApiForecastBaseURL`, `xRapidApiKeyName`, `xRapidApiKeyValue`, `xRapidApiHostName`, and `xRapidApiHostValue`. Lines 14 and 15 are highlighted with a yellow background.

```

src > app > Environment > TS EnvironmentVariables.ts > [e] EnvironmentalVariables
1  export const EnvironmentalVariables = [
2    production:false,
3
4    //BaseUrls to access the end points using the http client get method
5    weatherApiLocationBaseURL:'https://weather338.p.rapidapi.com/locations/search?',
6    weatherApiForecastBaseURL:'https://weather338.p.rapidapi.com/weather/forecast?',
7
8    //Variables for API key name and value
9    xRapidApiKeyName:'X-RapidAPI-Key',
10   xRapidApiKeyValue:'22143f76c6msh71d6f8ed5e3cdadp1f2bb7jsn71e18c067282',
11
12   //Variables for host name and value
13   xRapidApiHostName:'X-RapidAPI-Host',
14   xRapidApiHostValue:'weather338.p.rapidapi.com'
15 ]
16

```