

École Marocaine des Sciences de l'ingénieur

---

## **UrgenceKit : Conseils d'IA pour vos Fournitures d'Urgence**

---

**Par EL GOUFFI salma  
CHAHBAR aliaa  
KOUBICHATE jouhayna  
AGUABDRE khaoula**

23 avril 2024

Encadrants:

**Pr. Mohammed AMEKSA**   EMSI   Encadrant

Academic year 2023-24

# DÉDICACE

À tous nos amis et amies

Aucun mot ne pourrait compenser votre soutien, acceptez ce  
travail comme preuve de notre amour et amitié.

A tous les professeurs de l'Emsi

Merci pour la formation de qualité que vous nous avez donné tout au  
long de cette période d'études.

Merci pour l'expérience et les bons moments.

# REMERCIEMENTS

Au moment où l'on pense avoir fait œuvre utile, qu'il nous soit permis d'exprimer nos vifs remerciements et notre profonde gratitude à monsieur Mohammed AMEKSA , notre encadrant qui a bien voulu, en dépit de ses multiples engagements tant professionnels que personnels, accepter de diriger et de parrainer ce travail. Qu'il nous permette de rendre hommage à ses qualités exceptionnelles, à son savoir-faire, et à sa contribution bénéfique à la réalisation de ce projet.

Des remerciements spéciaux sont accordés au cadre professoral de l'EMSI, ainsi qu'au staff de la filière IRR pour la formation de qualité qu'ils nous ont prodigué tout au long de cet semestre de formation.

# TABLE DES MATIÈRES

<b>Dédicace</b>	<b>2</b>
<b>Remerciements</b>	<b>3</b>
<b>Table des matières</b>	<b>5</b>
<b>Liste des figures</b>	<b>6</b>
<b>Introduction générale</b>	<b>7</b>
<b>1 Contexte général</b>	<b>8</b>
1.1 Introduction . . . . .	9
1.2 Etude de l'existant . . . . .	9
1.3 Choix du langage de conception : . . . . .	10
1.3.1 Choix d'UML . . . . .	10
1.4 Conception . . . . .	11
1.4.1 Diagramme de Cas d'utilisation . . . . .	11
1.4.2 Diagramme de classe . . . . .	13
1.5 Conclusion . . . . .	13
<b>2 Outils de developpement et bibliothèques</b>	<b>14</b>
2.1 Introduction . . . . .	15
2.2 Langages et Frameworks . . . . .	15
2.2.1 HTML . . . . .	15
2.2.2 CSS . . . . .	15
2.2.3 Python . . . . .	15
2.2.4 Django . . . . .	16
2.2.5 JavaScript . . . . .	17
2.2.6 SQLite . . . . .	18
2.3 Bibliothèques . . . . .	18
2.4 Conclusion . . . . .	20
<b>3 Réalisation :</b>	<b>21</b>
3.1 Introduction . . . . .	22
3.2 Les interfaces utilisateur . . . . .	22
3.2.1 Interface d'inscription . . . . .	22
3.2.2 Interface d'authentification . . . . .	23
3.2.3 Interface du Tableau de bord . . . . .	24
3.2.4 Interface ai "Urgiebot" . . . . .	28
3.2.5 Interface Kit préféré . . . . .	29
3.2.6 Interface carte interactive . . . . .	29
3.2.7 Interface informations de profil . . . . .	31

---

3.3	Interface administrateur : . . . . .	32
3.3.1	Interface du Tableau de bord . . . . .	32
3.3.2	Interface Gestion du personnel et d'utilisateur . . . . .	32
3.3.3	Interface Gestion des kits . . . . .	33
3.3.4	Interface Gestion des Magazins . . . . .	34
3.3.5	Interface Gestion des produits . . . . .	35
<b>Conclusion</b>		<b>37</b>
		<b>37</b>

# LISTE DES FIGURES

1.1	Unified Modeling Language . . . . .	10
1.2	diagramme de cas d'utilisation . . . . .	12
1.3	diagramme de classe . . . . .	13
3.1	interface d'inscription . . . . .	23
3.2	Interface d'authentification . . . . .	24
3.3	Présentation d'urgencekit app . . . . .	24
3.4	Catastrophes . . . . .	25
3.5	Conseils de maintenance du kit d'urgence . . . . .	25
3.6	Suggestions de stockage du kit d'urgence . . . . .	26
3.7	Liste des Produits . . . . .	26
3.8	Liste des Numéros d'Urgences . . . . .	27
3.9	Liste des Kits d'Urgence . . . . .	28
3.10	Urgiebot . . . . .	28
3.11	My kit . . . . .	29
3.12	Map . . . . .	30
3.13	Map . . . . .	30
3.14	Map: Recherche . . . . .	31
3.15	Profile page . . . . .	31
3.16	Dashboard admin . . . . .	32
3.17	Gestion du personnel et d'utilisateur . . . . .	33
3.18	Gestion des kits . . . . .	34
3.19	Gestion des Magazins . . . . .	35
3.20	Gestion des produits . . . . .	36
3.21	liste des produits . . . . .	36

# INTRODUCTION GÉNÉRALE

Dans un monde où les catastrophes naturelles et les urgences peuvent survenir à tout moment, il est essentiel d'être préparé et bien équipé pour faire face à de telles situations. L'application "UrgenceKit" vise à répondre à ce besoin en fournissant aux utilisateurs des recommandations personnalisées sur les fournitures d'urgence en fonction du type de catastrophe et de leur emplacement géographique. Cette application, développée en utilisant le langage de programmation Python, offre une solution pratique et accessible pour aider les individus à se préparer et à gérer les situations d'urgence de manière efficace.

Grâce à "UrgenceKit", les utilisateurs auront la possibilité de créer un compte et de se connecter, leur permettant ainsi d'accéder à un large éventail de fonctionnalités. L'application permettra aux utilisateurs de fournir leur emplacement ou de permettre à l'application d'accéder à leur emplacement actuel, garantissant ainsi des recommandations précises et adaptées à leur situation géographique.

En outre, "UrgenceKit" utilisera des algorithmes d'intelligence artificielle pour analyser divers facteurs tels que le type de catastrophe, la gravité de la situation et les besoins spécifiques de l'utilisateur afin de fournir des recommandations sur les fournitures d'urgence. Ces recommandations personnalisées aideront les utilisateurs à identifier les articles essentiels dont ils ont besoin pour faire face à une situation d'urgence particulière.

De plus, l'application fournira également des fonctionnalités de localisation des magasins à proximité qui vendent des fournitures d'urgence. Les utilisateurs pourront visualiser les magasins sur une carte interactive, vérifier leur disponibilité en stock et accéder à des informations telles que les horaires d'ouverture, facilitant ainsi leur processus d'achat.

Grâce à son interface conviviale et intuitive, "UrgenceKit" offrira une expérience utilisateur optimale, permettant aux individus de se préparer et de gérer les situations d'urgence avec confiance et efficacité.

Nous allons terminer notre mémoire par une conclusion générale tirée à travers notre travail.

## CONTEXTE GÉNÉRAL

1.1	Introduction . . . . .	9
1.2	Etude de l'existant . . . . .	9
1.3	Choix du langage de conception : . . . . .	10
1.3.1	Choix d'UML . . . . .	10
1.4	Conception . . . . .	11
1.4.1	Diagramme de Cas d'utilisation . . . . .	11
1.4.2	Diagramme de classe . . . . .	13
1.5	Conclusion . . . . .	13



## 1.1 Introduction

Dans ce premier chapitre, nous explorerons la conception et l'analyse de l'application "UrgenceKit". Nous commencerons par étudier les différents cas d'utilisation de l'application à l'aide d'un diagramme de cas d'utilisation, qui nous permettra de comprendre les interactions entre les utilisateurs et le système. Ensuite, nous élaborerons un diagramme de classe pour modéliser la structure statique de l'application, en identifiant les différentes classes, leurs attributs et leurs relations. Cette analyse approfondie jettera les bases solides nécessaires pour le développement de l'application "UrgenceKit" en Python.

## 1.2 Etude de l'existant

L'application « UrgenceKit » vise à fournir aux utilisateurs des recommandations sur les fournitures d'urgence en fonction du type de catastrophe et de leur emplacement géographique. Elle aidera également les utilisateurs à localiser les magasins à proximité qui ont encore des fournitures en stock. Cette application sera développée en utilisant le langage de programmation Python.

- Gestion des comptes :
  - Les utilisateurs auront la possibilité de s'inscrire et de se connecter à l'application.
  - Ils pourront choisir de partager leur position ou de permettre à l'application d'accéder à leur position actuelle.
  - Les utilisateurs seront en mesure de mettre à jour leurs informations de profil.
- Localisation des magasins :
  - En utilisant la fonction de localisation, l'application identifiera les magasins à proximité proposant des fournitures d'urgence.
  - Ces magasins seront affichés sur une carte interactive, permettant aux utilisateurs de voir leur emplacement ainsi que des détails pertinents tels que les horaires d'ouverture et la disponibilité des stocks.
  - Les utilisateurs auront la possibilité de filtrer les magasins selon différents critères comme la distance et le type de fournitures disponibles.
- Interface utilisateur :

- L'interface utilisateur sera conviviale et intuitive, avec des fonctionnalités de recherche et de navigation facilement accessibles.
- Les utilisateurs auront la possibilité de sauvegarder leurs emplacements favoris ainsi que les recommandations de fournitures d'urgence pour une consultation ultérieure simplifiée.

### 1.3 Choix du langage de conception :

#### 1.3.1 Choix d'UML

Par rapport à toutes les méthodes orientées objets qui sont en utilisation, seule UML a la capacité de satisfaire tous les besoins de conceptions requises par les entreprises et les sociétés informatiques.

En effet, Il unifie les notations nécessaires aux différentes activités d'un processus de développement et offre en plus de ça les moyens d'établir le suivi des décisions prises depuis les spécifications jusqu'au codage.

- Notion d'UML

UML :(Unified Modeling Language) ou « Langage de Modélisation Unifié » Standardisé par l'OMG (Object Management Group) est une notion basée principalement sur les méthodes des BOOCH (de BOOCH), OMT de Rumbaugh et OOSE de Jacobson.UML a été proposé afin de standardiser les produits de développement (modèle, notation, diagramme) sans standardiser le processus de développement qui dépend des personnes, des applications, etc.



Figure 1.1: Unified Modeling Language

- Pourquoi une méthode objet ?

Les langages orientés objet constituent chacun une manière spécifique d'implémenter le paradigme objet. Ainsi, une méthode objet permet de définir le problème à haut niveau sans rentrer dans les spécificités d'un langage. Il représente ainsi un outil permettant de définir un problème de façon graphique, afin par exemple de le présenter à tous les acteurs d'un projet (n'étant pas forcément des experts en un langage de programmation).

- Les points forts d'UML :
  - UML est un langage formel et normalisé
    - \* Gain de précision
    - \* Encourage l'utilisation d'outils
    - \* Gage de stabilité
  - UML est un support de communication performant
    - \* UML a pour objectif de spécifier, construire, visualiser et documenter les systèmes à base de logiciel
    - \* UML n'est pas une méthode mais une notation qui laisse la liberté de conception.
    - \* UML est un langage graphique qui permet de modéliser tous les types de systèmes informatiques mais, qui nécessite toute fois une méthodologie de conception (UP, RUPP,...).

## 1.4 Conception

### 1.4.1 Diagramme de Cas d'utilisation

Ce diagramme est destiné à représenter les besoins des utilisateurs par rapport au système.

- Acteur : représente un rôle joué par une personne qui interagit directement avec le système étudié.
- Cas d'utilisation (use case) : représente un ensemble des séquences d'actions qui sont réalisées par le système et qui produisent un résultat observable intéressant pour un acteur particulier. L'utilisation d'un diagramme de cas d'utilisation s'avère indispensables pour décrire les besoins fonctionnels.

Le modèle de cas d'utilisation générale est représenté dans la figure suivante :

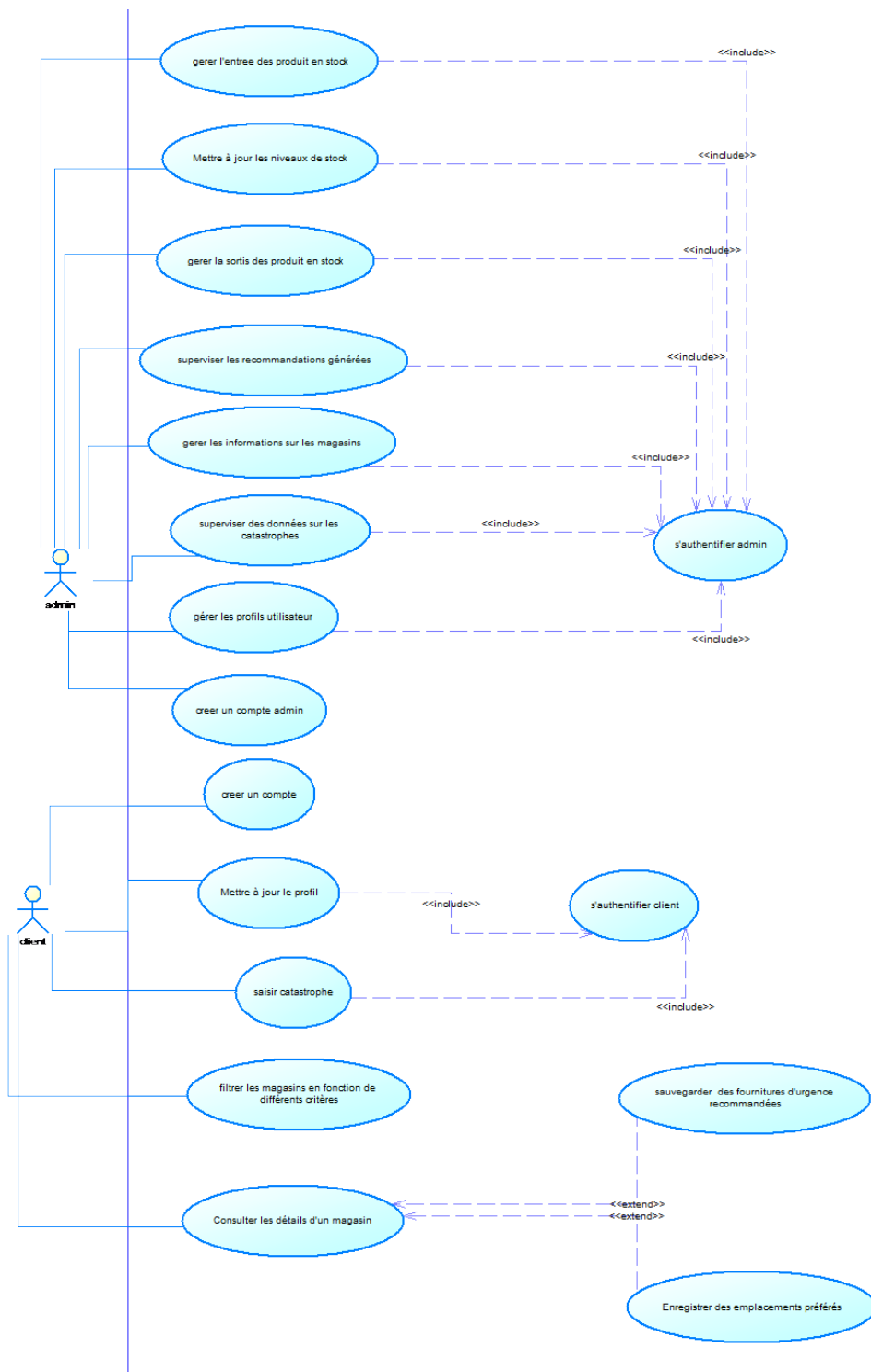


Figure 1.2: diagramme de cas d'utilisation

## 1.4.2 Diagramme de classe

Un diagramme de classe est essentiellement une représentation visuelle des composants statiques d'un système logiciel, tels que les classes, les interfaces, et leurs relations. Il offre une vue structurée des éléments du système et de leurs interactions.

Dans ce contexte, le diagramme ci-dessous présente les différentes entités et leurs associations, qui serviront de fondement à la conception de la base de données pour notre application.

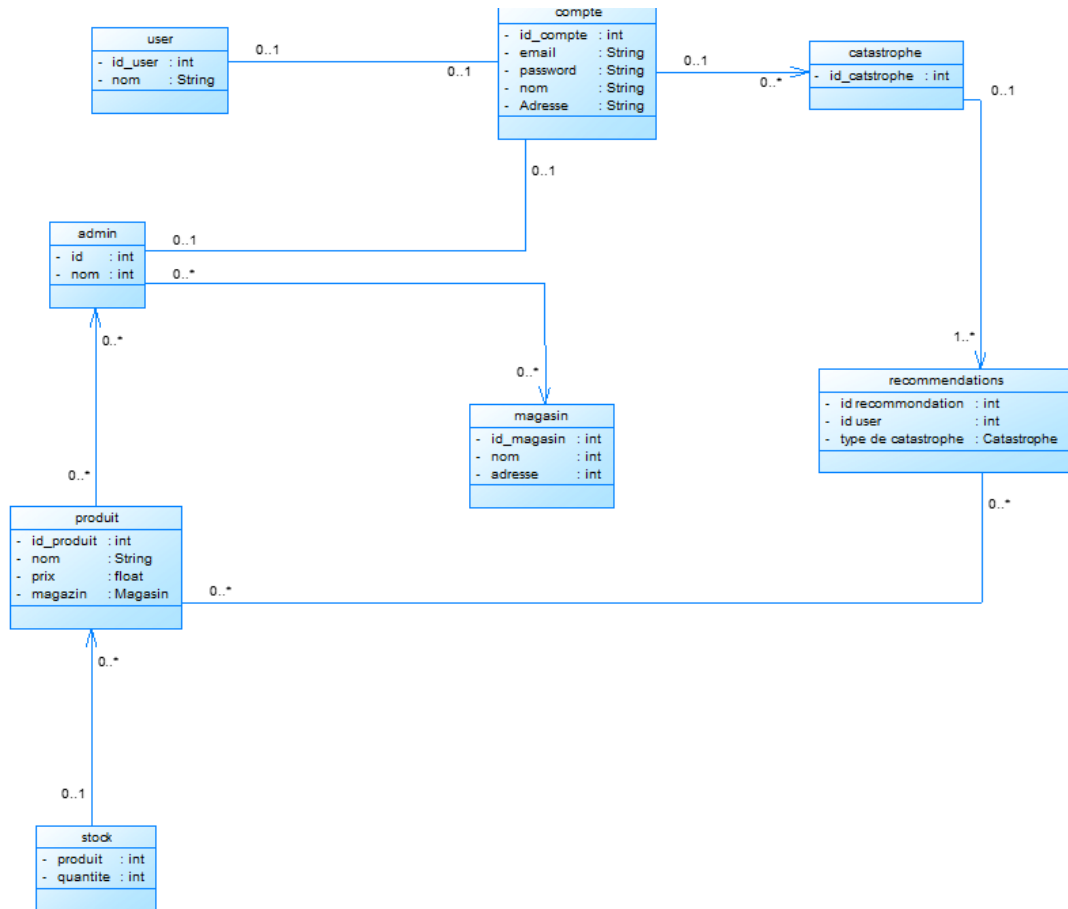


Figure 1.3: diagramme de classe

## 1.5 Conclusion

Dans ce chapitre, nous avons introduit notre application Web en utilisant différents modèles et diagrammes afin de fournir une vue d'ensemble claire et compréhensible du projet sous tous ses aspects. Toutes ces étapes seront ensuite traduites en interfaces, qui seront détaillées dans les chapitres suivants.

# OUTILS DE DEVELOPPEMENT ET BIBLIOTHÈQUES

2.1	Introduction . . . . .	15
2.2	Langages et Frameworks . . . . .	15
2.2.1	HTML . . . . .	15
2.2.2	CSS . . . . .	15
2.2.3	Python . . . . .	15
2.2.4	Django . . . . .	16
2.2.5	JavaScript . . . . .	17
2.2.6	SQLite . . . . .	18
2.3	Bibliothèques . . . . .	18
2.4	Conclusion . . . . .	20

## 2.1 Introduction

Dans ce chapitre, , nous allons explorer les outils de développement, les langages de programmation, les frameworks et les bibliothèques que nous avons utilisés pour concrétiser notre projet. Chaque élément a été soigneusement sélectionné pour répondre aux besoins spécifiques de notre application Web.

## 2.2 Langages et Frameworks

### 2.2.1 HTML

L'HTML (HyperText Markup Language) est un langage de balisage utilisé pour structurer le contenu des pages web. Il offre une variété de balises qui permettent de définir la présentation et l'organisation des éléments textuels et multimédias sur une page. Par exemple, les balises de paragraphe (<p>) sont utilisées pour séparer le texte en paragraphes distincts, tandis que les balises de lien hypertexte (<a>) permettent de créer des liens cliquables vers d'autres pages web, documents ou ressources en ligne.

Ces liens hypertexte sont un élément central de l'expérience de navigation sur le web, permettant aux utilisateurs de naviguer de manière fluide d'une page à une autre en un simple clic. Ils facilitent la découverte de nouveaux contenus et la navigation entre les différentes sections d'un site web.

### 2.2.2 CSS

CSS (Cacading Style Sheets :feuilles de style en cascade ) sert à décrire la présentation des documents HTML et XML. Les standards définissant CSS sont publiés par le World Wide Web consortium(W3c). La conception d'un document se fait dans un premier temps sans se soucier de la présentation, ce qui permet d'être plus efficace. Dans le cas d'une application web, la présentation est uniformisée : les documents (page HTML) font référence aux mêmes feuilles de style. Cette caractéristique permet de plus une remise en forme rapide de l'aspect visuel

### 2.2.3 Python

Python est un langage de programmation polyvalent, interprété et de haut niveau, apprécié pour sa syntaxe claire et concise qui favorise la lisibilité et la productivité du code. Avec une grande communauté de développeurs et une vaste bibliothèque standard, Python est large-

ment utilisé dans divers domaines tels que le développement web, la science des données, l'intelligence artificielle, l'automatisation de tâches, et bien d'autres encore.

En particulier, Python est largement adopté dans le domaine de l'intelligence artificielle (IA) et de l'apprentissage automatique (ML). Sa popularité dans ce domaine est due à plusieurs facteurs :

- Syntaxe claire et concise permettant aux développeurs de créer, d'expérimenter et de déployer rapidement des modèles d'IA sans se soucier de détails techniques complexes.
- Large communauté de développeurs et de chercheurs qui contribuent activement à l'écosystème d'IA en développant des bibliothèques, en partageant des connaissances et en fournissant un support technique.
- Facilité d'intégration avec d'autres langages et technologies, permettant l'incorporation de modèles d'IA dans des applications existantes ou l'utilisation de techniques d'IA pour résoudre des problèmes spécifiques.

### 2.2.4 Django

Django, un framework open-source en Python, offre une approche simplifiée pour développer rapidement des applications web sécurisées et fiables. Fondé sur le modèle MVC (Modèle-Vue-Contrôleur), Django encourage une architecture élégante et bien structurée du code.

Dans un projet Django, on trouve plusieurs fichiers et répertoires qui constituent la structure de base de l'application:

- `manage.py` : Un script en Python qui sert d'interface en ligne de commande pour diverses tâches liées au projet Django, telles que la création de migrations, le démarrage du serveur de développement, etc.
- `settings.py` : Un fichier de configuration principal où vous définissez les paramètres de votre application Django, tels que les bases de données, les fichiers statiques, les modèles de templates, .
- `urls.py` : Un fichier où vous configurez les routes de votre application web. Vous mappez les URLs aux vues correspondantes pour servir les pages web appropriées.
- `urls.py` : Un fichier où vous configurez les routes de votre application web. Vous mappez les URLs aux vues correspondantes pour servir les pages web appropriées.



- `wsgi.py` : Un point d'entrée pour les serveurs web compatibles avec WSGI (Web Server Gateway Interface). Il permet de déployer votre application Django sur des serveurs web comme Apache ou Nginx.
- `init.py` : Un fichier vide qui indique à Python que le répertoire est un package Python. Il peut également contenir du code d'initialisation si nécessaire.
- `apps.py` : Un fichier qui définit la configuration de l'application Django, telle que le nom de l'application et des hooks pour des événements spécifiques.
- `models.py` : Un fichier où vous définissez les modèles de données de votre application Django, en utilisant les classes Python qui représentent les tables de la base de données.
- `views.py` : Un fichier où vous définissez les vues de votre application Django, qui contiennent la logique métier pour traiter les requêtes HTTP et renvoyer les réponses appropriées.
- `forms.py` : Un fichier où vous définissez les formulaires de votre application Django, qui permettent aux utilisateurs de saisir et de soumettre des données via le navigateur web.
- `admin.py` : Un fichier où vous enregistrez les modèles de données de votre application Django pour les rendre accessibles dans l'interface d'administration générée automatiquement.
- `templates/` : Un répertoire où vous stockez les fichiers de modèles HTML utilisés pour générer les pages web de votre application.
- `static/` : Un répertoire où vous stockez les fichiers statiques tels que les feuilles de style CSS, les scripts JavaScript et les images utilisées par votre application web.

Ces fichiers et répertoires constituent une structure de base typique pour un projet Django, mais selon les besoins de l'application, on peut en trouver d'autres.

### 2.2.5 JavaScript

JavaScript est un langage de programmation de scripts largement utilisé pour développer des applications web interactives et dynamiques. Contrairement à HTML et CSS, qui sont utilisés pour structurer et styliser le contenu d'une page web, JavaScript permet d'ajouter

des fonctionnalités interactives telles que des animations, des effets visuels, des formulaires dynamiques, des jeux et des applications web en temps réel.

Les principales caractéristiques de JavaScript incluent sa capacité à être exécuté côté client dans un navigateur web, ce qui signifie que les scripts JavaScript sont interprétés et exécutés par le navigateur de l'utilisateur, et sa syntaxe flexible et expressive, inspirée de langages de programmation comme Java et C.

### 2.2.6 SQLite

SQLite est un système de gestion de base de données relationnelle (SGBDR) qui offre une solution légère, autonome et sans serveur pour la gestion de bases de données. Contrairement aux systèmes de gestion de base de données traditionnels qui fonctionnent en tant que serveurs distincts, SQLite fonctionne en tant que bibliothèque logicielle embarquée dans l'application. Cela signifie qu'au lieu d'accéder à une base de données via un processus distinct, l'application interagit directement avec la base de données via des appels de fonction.

## 2.3 Bibliothèques

Les bibliothèques et modules font partie intégrante du framework Django, offrant des fonctionnalités essentielles pour le développement d'applications web. Dans notre application, nous avons utilisé les bibliothèques suivantes pour faciliter divers aspects du développement :

- **django.shortcuts** : Ce module fournit des raccourcis et des utilitaires utiles pour simplifier le développement de vues dans Django. Les fonctions `render` et `redirect` sont souvent utilisées pour générer des réponses HTTP.
- **admin-datta.forms** : Contient des formulaires personnalisés pour l'inscription, la connexion et la gestion des mots de passe des utilisateurs dans l'application. Ces formulaires peuvent être utilisés pour créer des interfaces utilisateur conviviales pour ces fonctionnalités.
- **django.contrib.auth.views** : offre des vues prédéfinies pour les fonctionnalités d'authentification des utilisateurs, telles que la connexion, le changement de mot de passe et la réinitialisation de mot de passe. Par exemple, `LoginView`, `PasswordChangeView` et `PasswordResetView` sont des vues prédéfinies pour ces actions.
- **django.views.generic** : intègre des vues génériques prédéfinies qui peuvent être utilisées pour créer des vues basées sur des modèles, telles que `CreateView`, qui est utilisée pour créer des objets dans la base de données.

- **django.contrib.auth** : fournit des fonctionnalités d'authentification des utilisateurs, telles que la connexion, la déconnexion, la gestion des mots de passe, etc. La fonction `logout` est utilisée pour déconnecter un utilisateur.
- **django.contrib.auth.decorators** : Rassemble des décorateurs qui permettent de restreindre l'accès aux vues en fonction de l'état d'authentification de l'utilisateur. Par exemple, le décorateur `login-required` peut être utilisé pour s'assurer qu'un utilisateur est connecté avant d'accéder à une vue spécifique.
- **django.http** : Contient des classes pour gérer les requêtes HTTP et les réponses HTTP dans Django. Par exemple, `JsonResponse` est utilisé pour renvoyer des données au format JSON en réponse à une requête HTTP.
- **django.urls** : Cette bibliothèque fournit des outils pour la gestion des URL dans une application web Django. Elle contient notamment la fonction `path` que vous avez importée précédemment. Les URL sont définies dans le fichier `urls.py` de votre application Django et sont associées à des vues spécifiques.

- **google.generativeai Gemini**: En intégrant l'intelligence artificielle générative à notre application, nous avons pu exploiter les avancées réalisées dans les grands modèles de langage (LLMs), tels que ceux utilisés dans le projet Gemini de Google.

Ces LLMs, entraînés sur d'énormes corpus textuels, sont capables de prédire avec précision le prochain mot dans une phrase, assurant ainsi la cohérence et la précision du contenu généré.

Grâce à l'accès aux modèles génératifs de Gemini via son API, nous avons enrichi notre fonctionnalité en permettant à notre application de fournir des recommandations personnalisées sur les fournitures d'urgence en fonction du type de catastrophe, basées sur des descriptions de produits générées automatiquement à partir de simples titres saisis par les utilisateurs.

- **Mapbox GL JS** : Une bibliothèque JavaScript pour créer des cartes interactives basées sur Mapbox. Elle est utilisée pour afficher la carte et ajouter des fonctionnalités de zoom, de déplacement et de personnalisation des marqueurs.
- **Mapbox GL Geocoder Plugin**: plugin Mapbox GL Geocoder est une extension de Mapbox GL JS qui offre une fonctionnalité de géocodage à la carte. En permettant aux utilisateurs de rechercher des lieux spécifiques, il affiche ensuite les résultats directement sur la carte.

Cette fonctionnalité améliore l'expérience utilisateur en facilitant la découverte et la visualisation des lieux d'intérêt.

## 2.4 Conclusion

Au cours de ce chapitre, nous avons étudié les bibliothèques et modules de Django utilisés dans notre application, facilitant le développement web avec des fonctionnalités clés comme la gestion des vues et des formulaires. Nous avons également exploré l'utilisation de bibliothèques externes comme Mapbox GL JS pour intégrer une carte interactive, améliorant ainsi l'expérience utilisateur. En outre, nous avons abordé l'utilisation de Google's generative AI Gemini.

## RÉALISATION :

3.1	Introduction . . . . .	22
3.2	Les interfaces utilisateur . . . . .	22
3.2.1	Interface d'inscription . . . . .	22
3.2.2	Interface d'authentification . . . . .	23
3.2.3	Interface du Tableau de bord . . . . .	24
3.2.4	Interface ai "Urgiebot" . . . . .	28
3.2.5	Interface Kit préféré . . . . .	29
3.2.6	Interface carte interactive . . . . .	29
3.2.7	Interface informations de profil . . . . .	31
3.3	Interface administrateur : . . . . .	32
3.3.1	Interface du Tableau de bord . . . . .	32
3.3.2	Interface Gestion du personnel et d'utilisateur . . . . .	32
3.3.3	Interface Gestion des kits . . . . .	33
3.3.4	Interface Gestion des Magazins . . . . .	34
3.3.5	Interface Gestion des produits . . . . .	35

## 3.1 Introduction

"Dans ce chapitre, nous explorerons les interfaces utilisateur et administrateur de notre application, y compris l'interface fournisseur, en mettant l'accent sur la conception et la fonctionnalité pour une expérience utilisateur optimale.

## 3.2 Les interfaces utilisateur

### 3.2.1 Interface d'inscription

L'interface "Sign up" de l'application "UrgenceKit" permet aux utilisateurs de créer un compte afin d'accéder aux fonctionnalités personnalisées et aux recommandations d'urgence. Voici une description de cette interface :

- **Formulaire d'inscription** : Les utilisateurs sont invités à remplir un formulaire comprenant des champs tels que nom(username), adresse e-mail, mot de passe. Ce formulaire recueille les informations nécessaires pour créer un compte utilisateur.
- **Validation des données** : Les données saisies par l'utilisateur sont vérifiées en temps réel pour s'assurer de leur exactitude et de leur validité. Des indications sont fournies en cas d'erreur de saisie pour guider l'utilisateur dans la correction des informations.
- **Confirmation d'inscription** : Une fois que toutes les informations requises sont saisies et validées, le compte est créé avec succès.
- **Redirection vers le tableau de bord** : Après l'inscription réussie, les utilisateurs sont automatiquement redirigés vers page d'authentification.

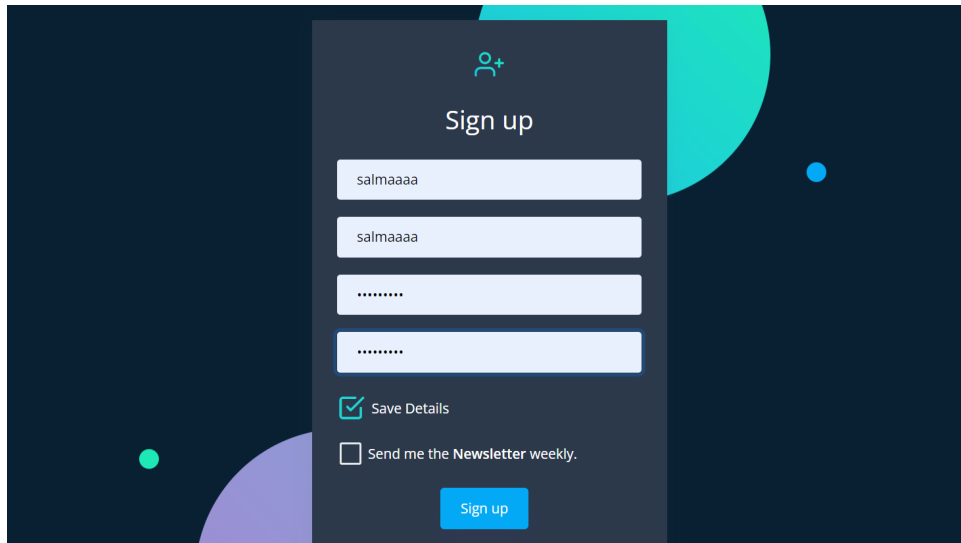


Figure 3.1: interface d'inscription

### 3.2.2 Interface d'authentification

L'interface "login" de l'application "UrgenceKit" permet aux utilisateurs enregistrés d'accéder à leur compte. Voici une description de cette interface :

- **Formulaire de connexion** : Les utilisateurs doivent saisir leur nom et leur mot de passe dans les champs prévus à cet effet.
- **Options de récupération de mot de passe** : En cas d'oubli du mot de passe, les utilisateurs ont la possibilité de récupérer leur mot de passe en cliquant sur un lien de récupération. Ils peuvent recevoir un e-mail avec un lien sécurisé pour réinitialiser leur mot de passe.
- **Validation des informations d'identification** : Les informations d'identification fournies par l'utilisateur sont vérifiées pour garantir l'exactitude et la sécurité de la connexion. Des messages d'erreur sont affichés en cas de saisie incorrecte.
- **Redirection vers le tableau de bord (Dashboard)** : Après une connexion réussie, les utilisateurs sont automatiquement redirigés vers leur tableau de bord personnel, où ils peuvent accéder aux fonctionnalités de l'application et consulter les recommandations d'urgence.

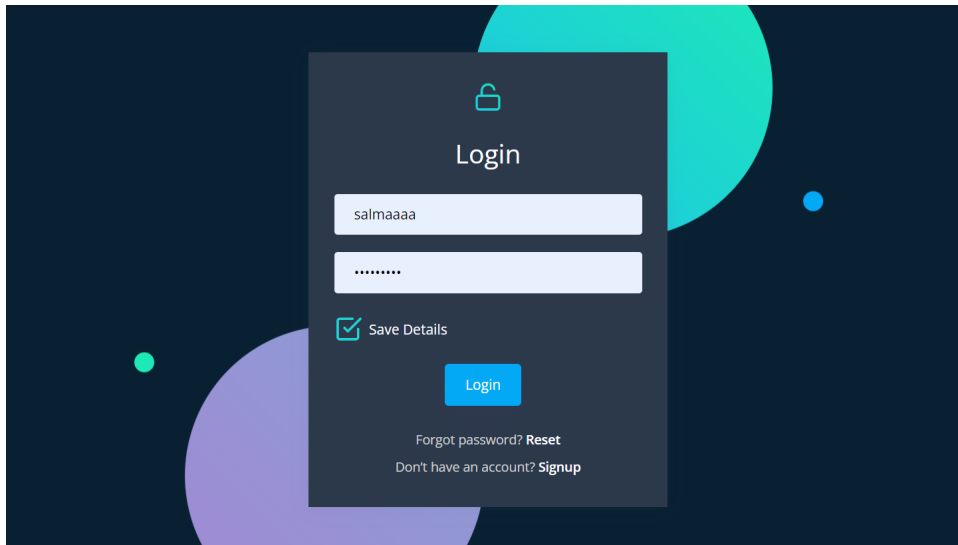


Figure 3.2: Interface d'authentification

### 3.2.3 Interface du Tableau de bord

L'interface "Tableau de bord" de l'application "UrgenceKit" offre aux utilisateurs une vue complète et conviviale des fonctionnalités disponibles. Voici une description de cette interface :

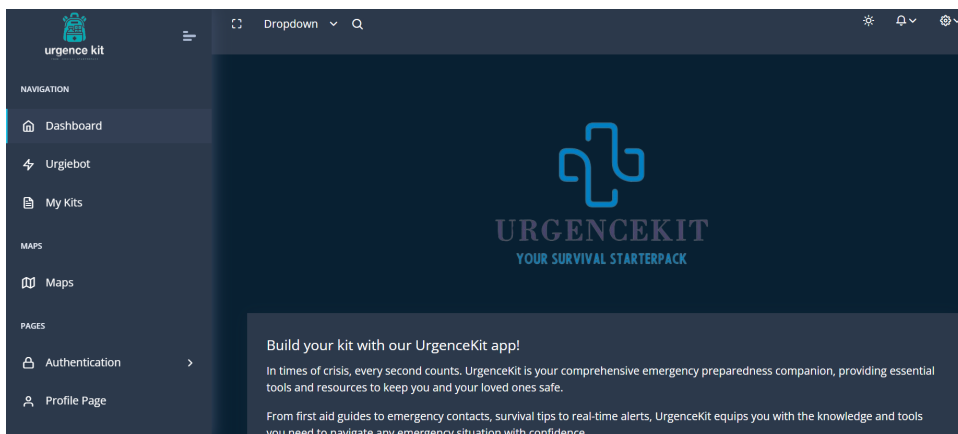


Figure 3.3: Présentation d'urgencekit app

- **Présentation des catastrophes :** Des images représentant différentes catastrophes naturelles sont affichées, telles que les tremblements de terre, les tempêtes, les incendies de maison, les vagues de chaleur et les inondations. Chaque image est associée à un lien permettant de communiquer avec Urgiebot pour obtenir des informations spécifiques sur la catastrophe sélectionnée.



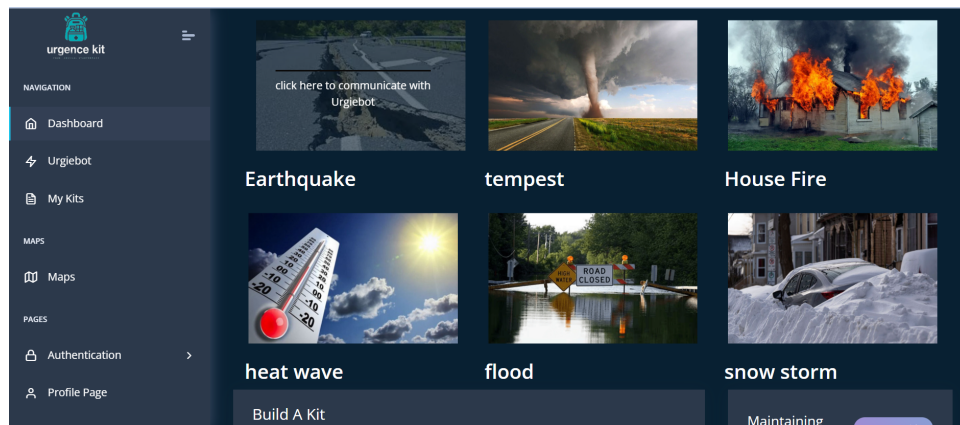


Figure 3.4: Catastrophes

- **Conseils de maintenance du kit d'urgence :** Des conseils sont fournis pour aider les utilisateurs à maintenir leur kit d'urgence, y compris des recommandations sur le stockage des aliments en conserve, le remplacement des articles périmés et la mise à jour régulière du kit en fonction des besoins changeants de la famille.

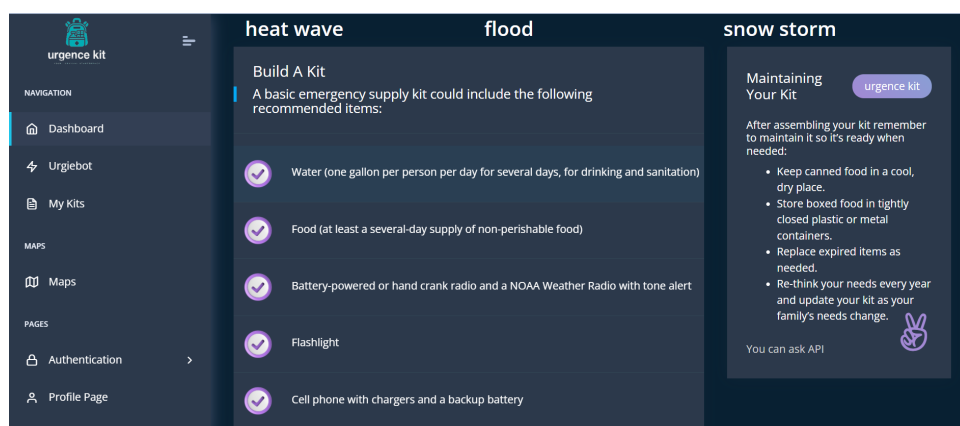


Figure 3.5: Conseils de maintenance du kit d'urgence

- **Suggestions de stockage du kit d'urgence :** Des recommandations sont fournies sur les lieux de stockage du kit d'urgence, notamment à la maison, au travail et dans la voiture, avec des conseils spécifiques pour chaque emplacement.

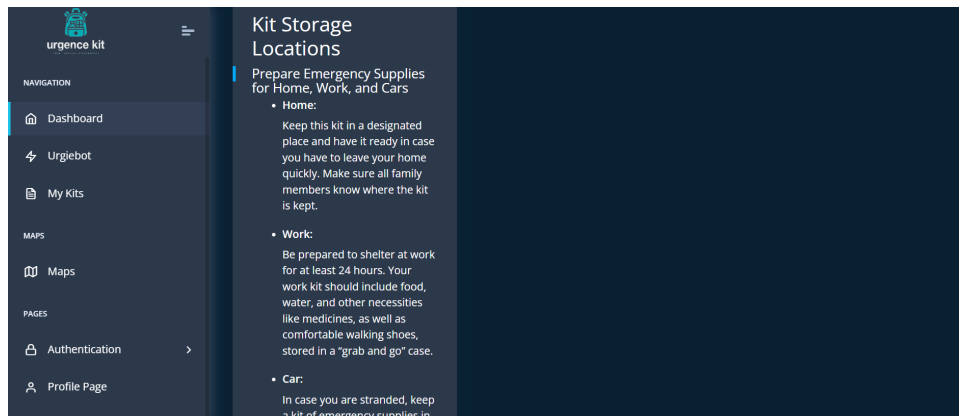


Figure 3.6: Suggestions de stockage du kit d'urgence

- **Liste des Produits :** Cette section affiche les produits disponibles, avec des détails tels que le nom du produit, une brève description et le prix. Les utilisateurs peuvent parcourir les produits et voir les détails pertinents.

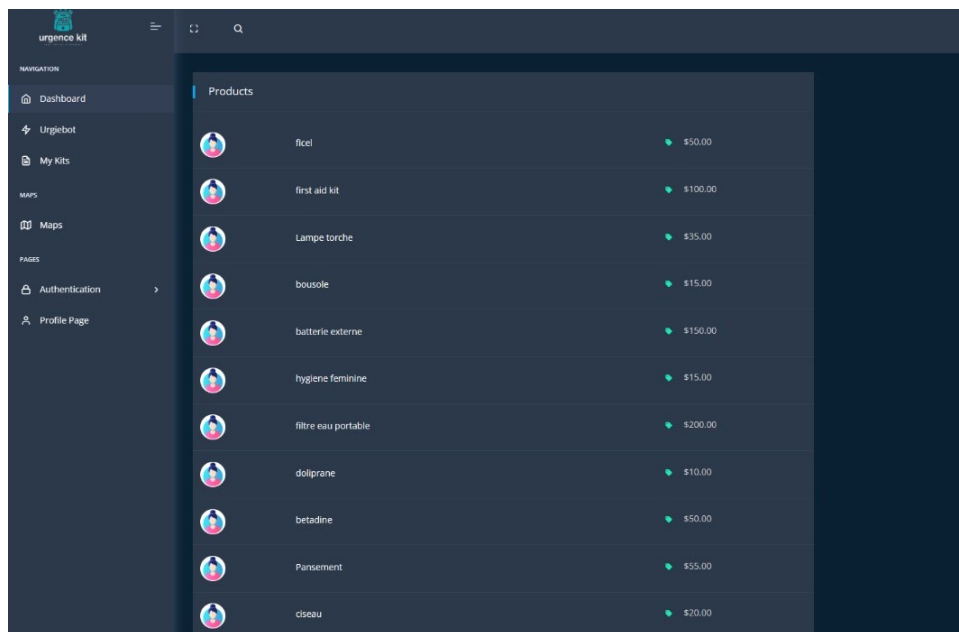


Figure 3.7: Liste des Produits

- **Liste des Numéros d'Urgences :** Cette section répertorie les numéros d'urgences importants, tels que les services d'incendie, de police et d'ambulance. Les utilisateurs peuvent accéder rapidement à ces numéros en cas de besoin.

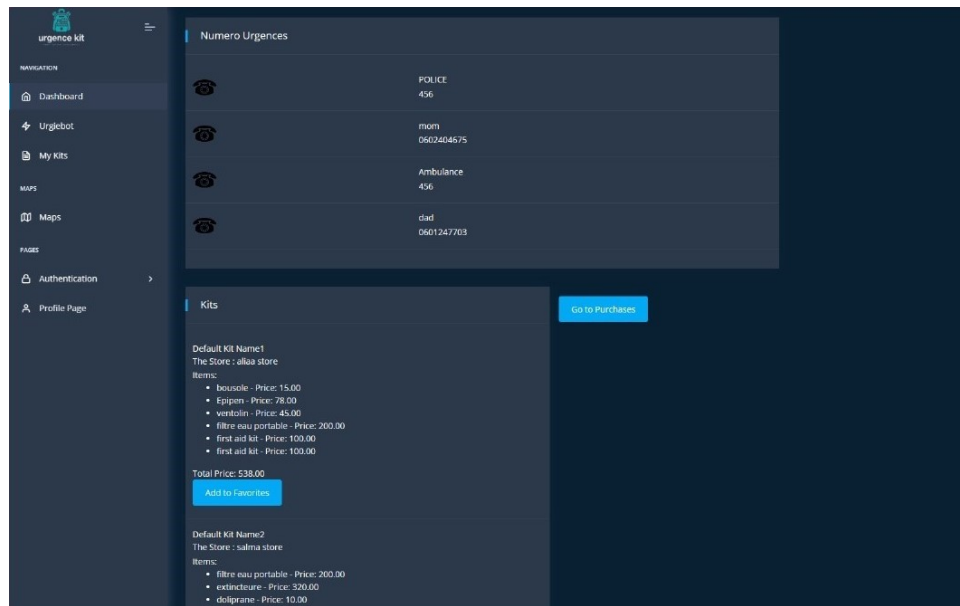


Figure 3.8: Liste des Numéros d'Urgences

- **Liste des Kits d'Urgence** : Les kits d'urgence sont répertoriés ici, avec des détails sur le contenu de chaque kit, y compris les articles inclus et le prix total. Les utilisateurs peuvent consulter les détails des kits et choisir celui qui convient le mieux à leurs besoins.
- **Ajouter aux Favoris** : Les utilisateurs ont la possibilité d'ajouter des kits à leurs favoris en cliquant sur un bouton dédié. Cela leur permet de sauvegarder les kits qu'ils trouvent intéressants et de les retrouver facilement ultérieurement.

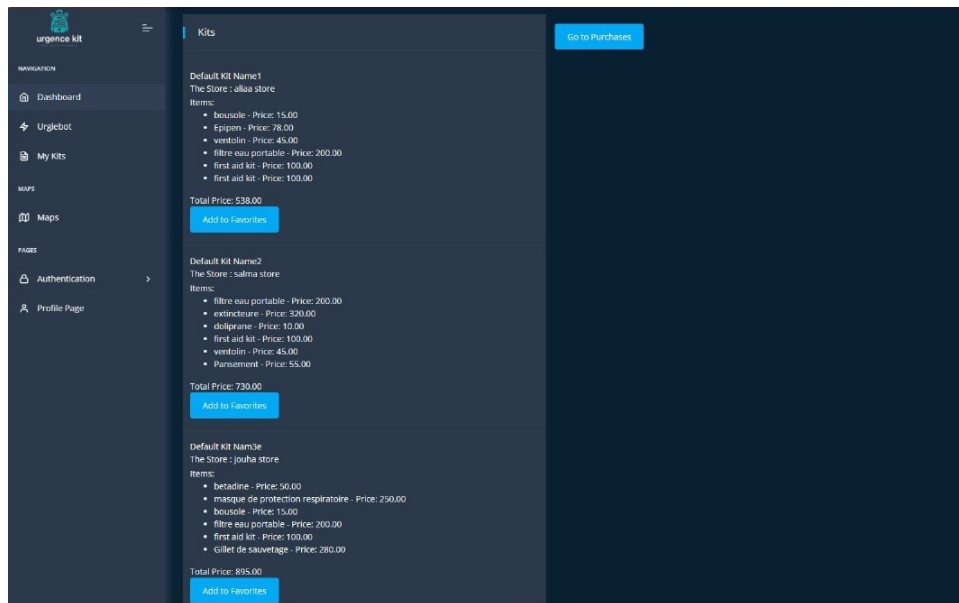


Figure 3.9: Dashboard : Liste des Kits d'Urgence

### 3.2.4 Interface ai "Urgiebot"

L'interface "Urgiebot" est conçue pour permettre aux utilisateurs d'interagir avec un chatbot. Ce dernier est basé sur le modèle génératif "gemini-1.5-pro-latest" et est configuré pour répondre aux requêtes des utilisateurs concernant les catastrophes et les urgences. Le chatbot est pré-entraîné pour fournir des informations et des conseils sur différentes catastrophes naturelles telles que les tremblements de terre, les tempêtes de neige.

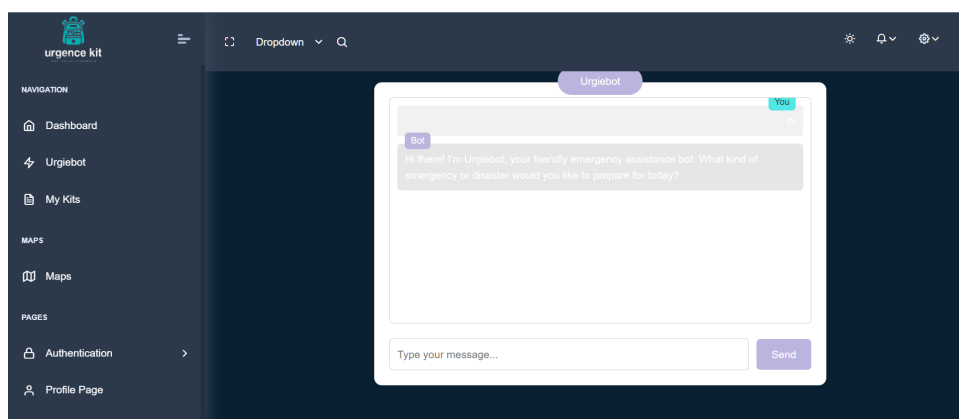


Figure 3.10: Urgiebot

### 3.2.5 Interface Kit préféré

L'interface "Kit préféré" présente une liste des kits les plus appréciés par les utilisateurs de l'application. Elle offre une vue organisée et conviviale des différents kits qui ont reçu le plus d'attention de la part des utilisateurs. Chaque kit est généralement accompagné de son nom, et éventuellement le magasin auquel il appartient.

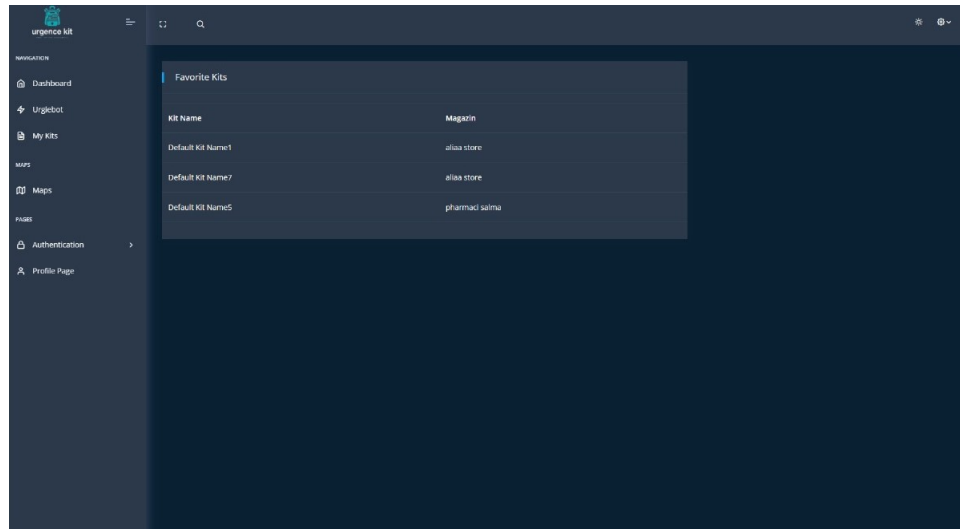


Figure 3.11: My kit

### 3.2.6 Interface carte interactive

L'interface "Maps" est une page web qui affiche une carte interactive en utilisant Mapbox GL JS. Elle permet à l'utilisateur de visualiser des emplacements spécifiques tels que les hôpitaux, les commissariats de police, les pharmacies et les supermarchés dans sa région.

Avec deux boutons "Go to emsi" et "GO other way" sont disponibles pour permettre à l'utilisateur de naviguer rapidement vers des emplacements prédéfinis ou de s'adapter automatiquement à une vue qui montre tous les emplacements.

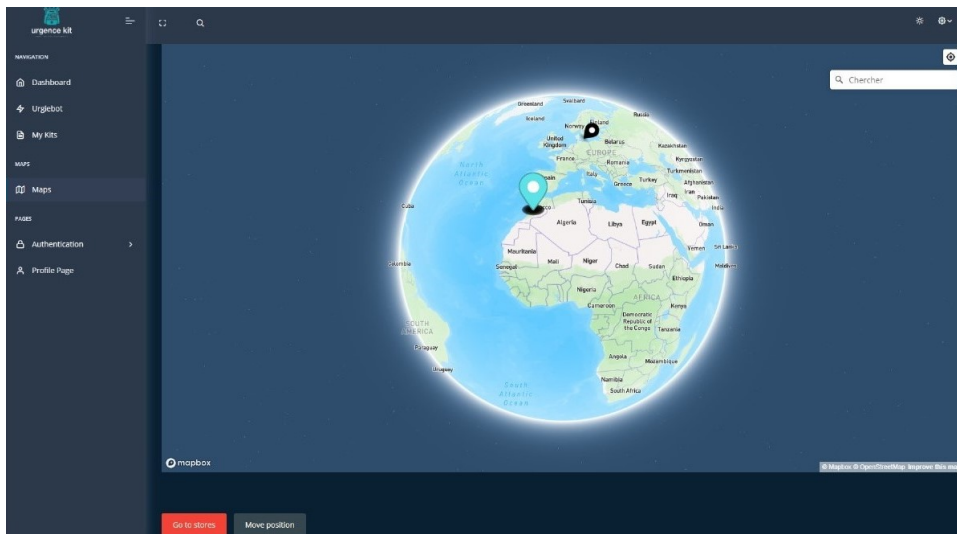


Figure 3.12: Map

Elle offre également aux utilisateurs une vue organisée et visuelle des magasins préalablement enregistrés dans l'application,

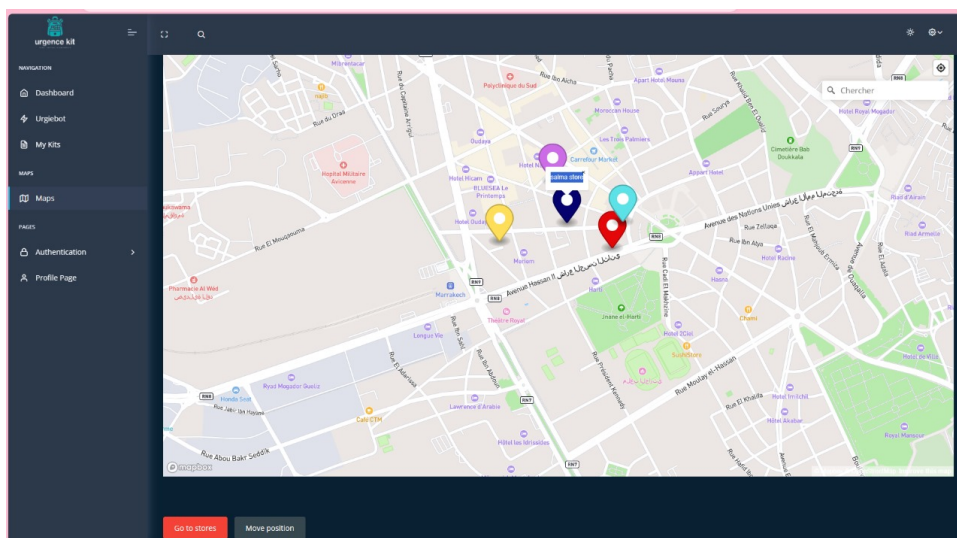


Figure 3.13: Map

La barre de recherche permet aux utilisateurs de saisir le nom ou l'adresse d'un lieu spécifique qu'ils souhaitent localiser sur la carte.

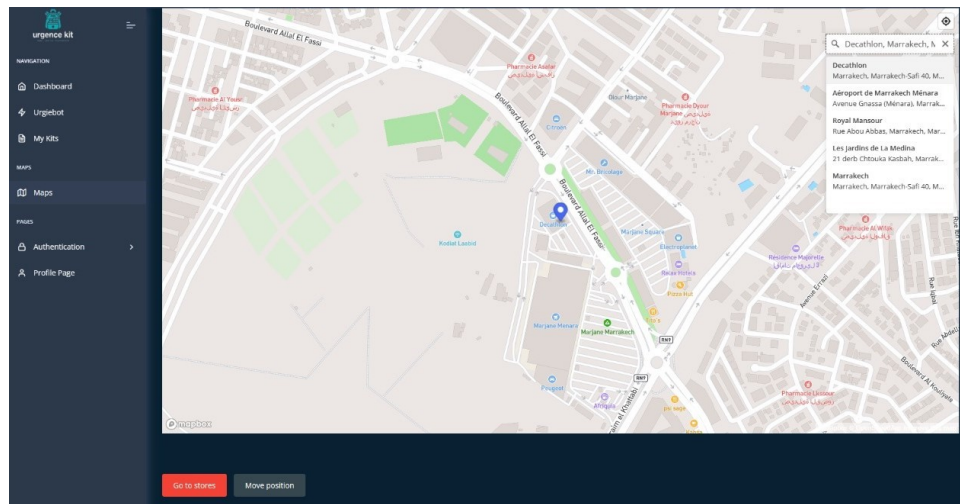


Figure 3.14: Map: Recherche

### 3.2.7 Interface informations de profil

L'interface "Profile page" est une page de modification des informations utilisateur. Elle présente un formulaire où les utilisateurs peuvent mettre à jour leur adresse e-mail, prénom, nom, adresse de livraison et informations biographiques.

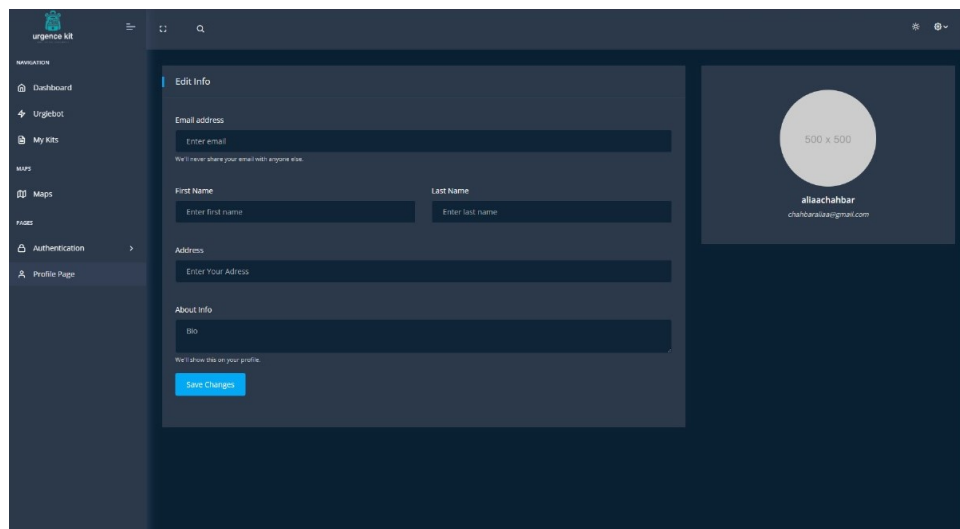


Figure 3.15: Profile page

## 3.3 Interface administrateur :

### 3.3.1 Interface du Tableau de bord

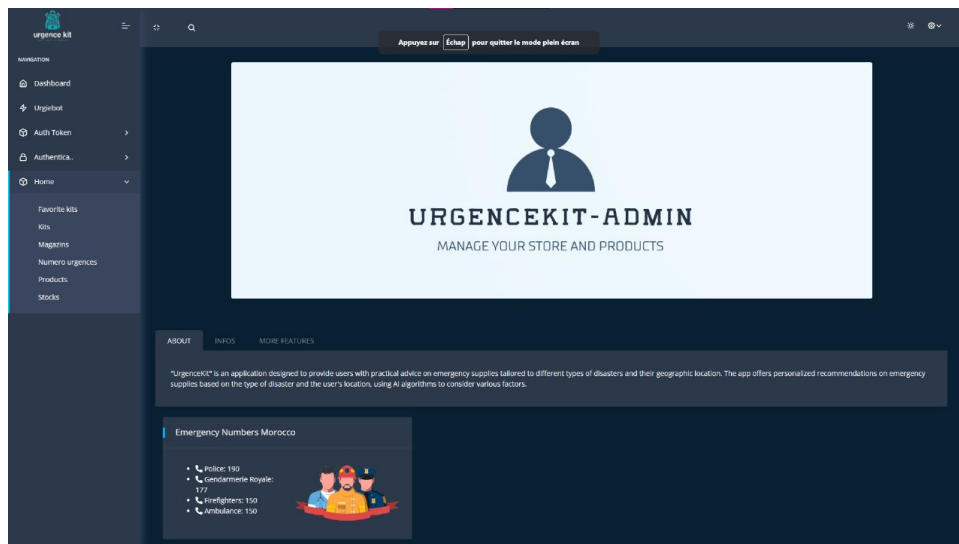


Figure 3.16: Dashboard admin

### 3.3.2 Interface Gestion du personnel et d'utilisateur

L'interface d'administration offre une vue complète des utilisateurs enregistrés dans le système ou l'application. La liste des utilisateurs présente leurs informations principales, notamment leur nom d'utilisateur, leur adresse e-mail, leur prénom et leur nom de famille.

En outre, chaque utilisateur est accompagné d'un indicateur de statut du personnel, qui spécifie s'il fait partie du personnel ou non, représenté par une valeur booléenne (True ou False). Cette disposition facilite la gestion et l'identification des utilisateurs, permettant aux administrateurs d'effectuer des actions appropriées en fonction de leurs besoins.

Cette interface offre également des fonctionnalités d'administration, notamment la possibilité de sélectionner plusieurs utilisateurs simultanément et d'effectuer diverses actions sur eux. Ces actions peuvent inclure la modification des informations utilisateur, la réinitialisation des mots de passe, la suppression de comptes .



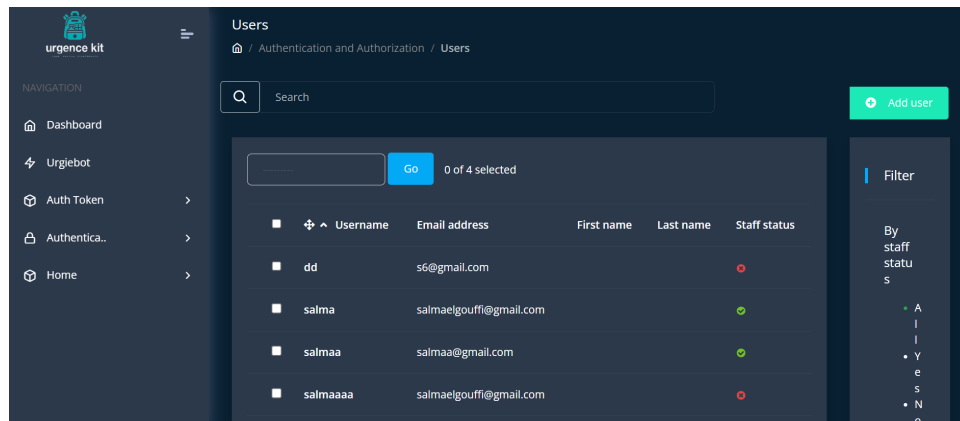


Figure 3.17: Gestion du personnel et d'utilisateur

### 3.3.3 Interface Gestion des kits

L'interface d'administration pour ajouter des kits est conçue pour simplifier la tâche de création de nouveaux kits. Voici une description des différents éléments de cette interface:

- **Add kit** : Il s'agit de l'option principale permettant d'ajouter un nouveau kit. En cliquant dessus, un formulaire d'ajout de kit s'ouvre, offrant la possibilité de saisir les détails du nouveau kit.
- **Magasin** : Ce champ ou liste déroulante permet de sélectionner le magasin auquel le kit sera associé. Cela peut être utile pour organiser les kits en fonction de leur emplacement ou de leur destination.
- **Name** : Il s'agit d'un champ texte où l'utilisateur peut saisir le nom du kit. Ce nom permettra d'identifier le kit de manière unique dans le système.
- **Items** : Cette section affiche une liste des éléments constituant le kit, avec leurs noms affichés. Cela permet à l'utilisateur de voir rapidement quels articles sont inclus dans le kit et d'apporter d'éventuelles modifications si nécessaire.

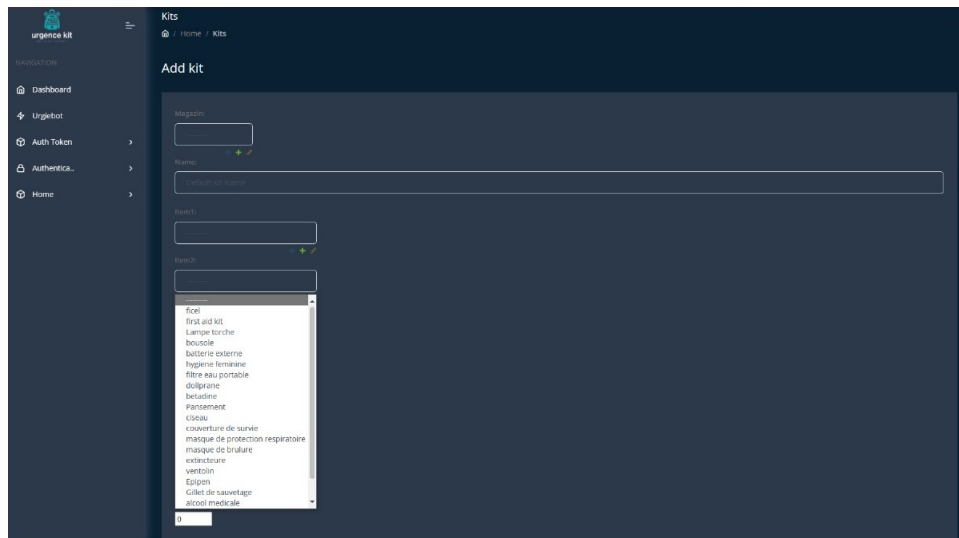


Figure 3.18: Gestion des kits

### 3.3.4 Interface Gestion des Magazins

L'interface d'administration pour l'ajout de magasins facilite grandement la gestion des magasins dans le système. En utilisant le champ "Magasin", l'administrateur peut saisir le nom du nouveau magasin à ajouter. Le bouton "Go" permet de soumettre les informations saisies et d'ajouter le magasin à la liste.

En dessous, la liste des magasins existants est affichée, permettant à l'administrateur de visualiser rapidement les magasins déjà enregistrés. Cette interface simplifiée facilite grandement la gestion des magasins dans le système, offrant une solution pratique pour les administrateurs.

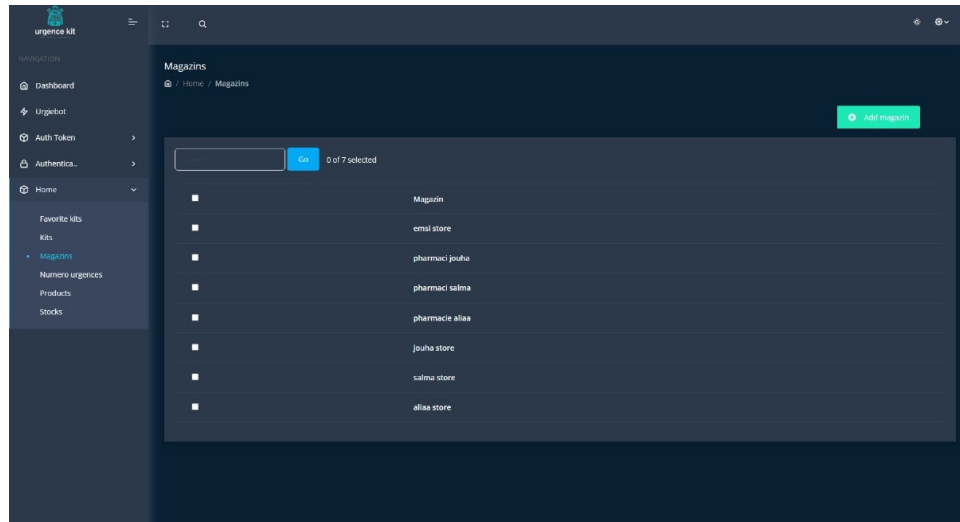


Figure 3.19: Gestion des Magazins

### 3.3.5 Interface Gestion des produits

L'interface d'administration pour l'ajout de produits propose un processus convivial pour ajouter de nouveaux articles à la base de données.

Elle offre des champs clairement étiquetés, tels que "Name" pour le nom du produit, "Image" pour télécharger une représentation visuelle, "Price" pour définir le prix, et "Number of items" pour spécifier la quantité en stock. Les options d'action, telles que "Save and continue editing" pour sauvegarder tout en restant dans l'édition, "Save and add another" pour enregistrer et ajouter un autre produit, ainsi que "Save" pour finaliser et quitter l'interface, simplifient la gestion des produits.

Une fois remplis, ces champs permettent à l'administrateur d'ajouter efficacement de nouveaux produits à la liste, améliorant ainsi la fonctionnalité globale du système d'administration.

The screenshot shows the 'Add product' form in the 'urgence kit' application. The form is titled 'Add product' and is located under the 'Products' section. It contains the following fields and controls:

- Name:** A text input field.
- Image:** A button labeled 'Choisir un fichier' and a message 'Aucun fichier n'a été sélectionné'.
- Price:** A text input field with the value '0'.
- Weight:** A text input field with the value '0'.
- Product of Weight:** A text input field with the value '0'.
- Buttons:** Three buttons at the bottom: 'Save and continue editing' (yellow), 'Save and add another' (blue), and 'Save' (blue).

Figure 3.20: Gestion des produits

De plus, L'administration a la possibilité de visualiser la liste des produits déjà enregistrés, ce qui lui permet de maintenir une vue d'ensemble de l'inventaire et de prendre des décisions informées en matière de gestion des produits.

The screenshot shows the 'Liste des produits' table in the 'urgence kit' application. The table has a header row with a checkbox and a 'Product' column. The table contains 15 rows of product data. At the top of the table, there is a search bar and a 'Go' button. The table is titled '0 of 21 selected'.

<input type="checkbox"/>	Product
<input type="checkbox"/>	Couteau suisse
<input type="checkbox"/>	bandage
<input type="checkbox"/>	alcool medicale
<input type="checkbox"/>	Gilet de sauvetage
<input type="checkbox"/>	Epipen
<input type="checkbox"/>	ventolin
<input type="checkbox"/>	extincteur
<input type="checkbox"/>	masque de brulure
<input type="checkbox"/>	masque de protection respiratoire
<input type="checkbox"/>	couverture de survie
<input type="checkbox"/>	ciseau
<input type="checkbox"/>	Pansement
<input type="checkbox"/>	betadine
<input type="checkbox"/>	doliprane
<input type="checkbox"/>	filtre eau portable

Figure 3.21: liste des produits

# CONCLUSION

L'application "UrgenceKit" vise à répondre à plusieurs préoccupations en fournissant aux utilisateurs des recommandations personnalisées sur les fournitures d'urgence en fonction du type de catastrophe et de leur emplacement géographique. Cette approche innovante renforce la préparation des individus et leur capacité à réagir efficacement face à divers scénarios d'urgence.

Dans le cadre de notre projet "UrgenceKit", nous avons développé une plateforme sécurisée qui offre une interface conviviale pour les utilisateurs. Notre approche repose sur une architecture solide, avec l'application comme pivot central pour la gestion des recommandations et des ressources d'urgence.

Pour répondre aux défis spécifiques rencontrés par les utilisateurs en situation d'urgence, notre application utilise des algorithmes avancés pour analyser les données et fournir des recommandations personnalisées. Ces recommandations prennent en compte divers facteurs tels que le type de catastrophe, la gravité de la situation et les besoins spécifiques de l'utilisateur, garantissant ainsi une réponse adaptée à chaque scénario.

En outre, "UrgenceKit" offre des fonctionnalités de localisation avancées, permettant aux utilisateurs de trouver rapidement les magasins à proximité proposant des fournitures d'urgence. Cette fonctionnalité essentielle facilite l'accès aux ressources nécessaires en cas de besoin, renforçant ainsi la préparation et la réactivité des utilisateurs en situation d'urgence.

Notre projet "UrgenceKit" a été une expérience enrichissante qui nous a permis d'explorer les défis uniques liés à la conception et au développement d'une application web dédiée à la gestion des urgences. Bien que des contraintes de temps aient limité l'étendue de nos fonctionnalités, nous sommes convaincus que notre application offre une solution précieuse pour renforcer la préparation et la résilience face aux situations d'urgence. Grâce à ce projet, nous avons acquis une maîtrise des nouveaux langages et frameworks tels que Django, démontrant ainsi notre capacité à assimiler rapidement de nouvelles technologies pour répondre aux besoins de nos utilisateurs.