**PARTIE I : PROBLEMATIQUE ET POSITIONNEMENT**

1) Introduction générale :

Des ventes de mains en mains, vers des ventes virtuelles, passent les priorités des opérations de ventes des biens et des services, ce qui nous rend obligés de donner plus d’importance à la vente électronique.

Les boutiques en ligne sont depuis des années, largement conseillés pour les sociétés qui se basent sur la vente des produits et même des services.

Ces types de sites web représentent un dispositif global fournissant aux clients un pont de passage à l’ensemble des informations, des produits, et des services à partir d’un portail unique en rapport avec son activité.

Les sites de vente en ligne permettent aux clients de profiter d’une foire virtuelle disponible est quotidiennement mise à jour sans la moindre contrainte, ce qui leur permettrai de ne jamais rater les coups de cœur, ainsi Une foire sans problèmes de distance géographique, ni d’horaire de travail ni de disponibilité de transport. D’une autre part ces sites offrent à la société de profiter de cette espace pour exposer ses produits à une plus large base de clientèle.

Dans ce contexte, Le projet est la conception et la réalisation d’une application web e-commerce dans le cadre de fin d’études de License Fondamentale en informatique.

2) Travail demandé :

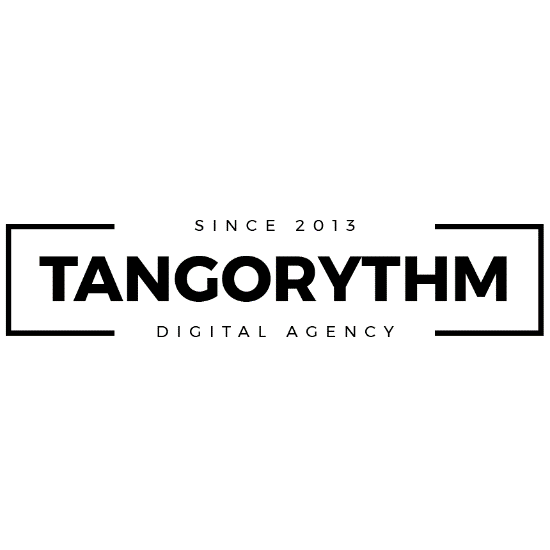
Le travail qui On a été demandé au sein de société TangoRythm est la réalisation d’une application web à l’aide du stack MERN (MongoDB, Express.js, React.js et Node.js).

Vu le manque d’expérience des technologies cités, ça nous demande un effort supplémentaire pour bien les maitrisés.

3) Cadre de stage :

3.1) Le stage :

Le stage s’est déroulé à Djerba, au sein de société TangoRythm dés 1er Mars. Mr. Marwen Hawari, le co-fondateur de la société, nous a encadré pendant cette période, Ainsi que, Mr. Hamed Hawari qui m’a aidé solidement. Malheureusement, et à cause de la pandémie de COVID-19, On n’a pas pu le choix de continuer ce stage à la société depuis le 20 Mars.

3.2) La société :

TangoRythm est une agence digitale qui fournit le développement créatif, stratégique et technique des produits et services informatiques. Ils aident leurs clients (sociétés, PMEs & les organisations non gouvernementales) à s'engager avec des marchés cibles spécifiques. Ils élaborent des stratégies de marketing et de communication. Leur domaine est beaucoup plus que de simples sites Web. Ils guident et développent des marques à travers le paysage numérique. Ils apportent une valeur ajoutée au clients grâce à des moyens digitales.

4) Plan de rapport :

Ce rapport sera divisé en quatre parties. La première partie sera consacré à l’étude de périmètre du projet. Dans cette partie, l’étude de l’existants ainsi que la présentation du projet à réaliser seront traités. Une description de la conception réalisée fera l’objectif de la seconde partie. Cette conception est effectuée à l’aide d’UML. La description du travail réalisé, l’implémentation, ainsi que les différentes technologies utilisées feront l’objet de la troisième partie. Enfin, nous clôturons par une conclusion et une série de perspectives.

**PARTIE II : ETUDE DE L’EXISITANT ET SPECIFICATION DE BESOINS**

1) Introduction :

Le travail qui nous a été confié pendant la durée de stage consiste à développer une application web e-commerce. Pour ce faire, il est indispensable de réaliser une étude de ce qui existe dans le marché pour comprendre, en premier, comment fonctionnent les applications existantes et, en second, sur quels points on va travailler. Dans cette optique, la première section de cette partie sera pour analyser et critiquer les sites existants, pour ensuite donner un aperçu de ce qui est demandé : spécification des besoins fonctionnels et non fonctionnels de l’application.

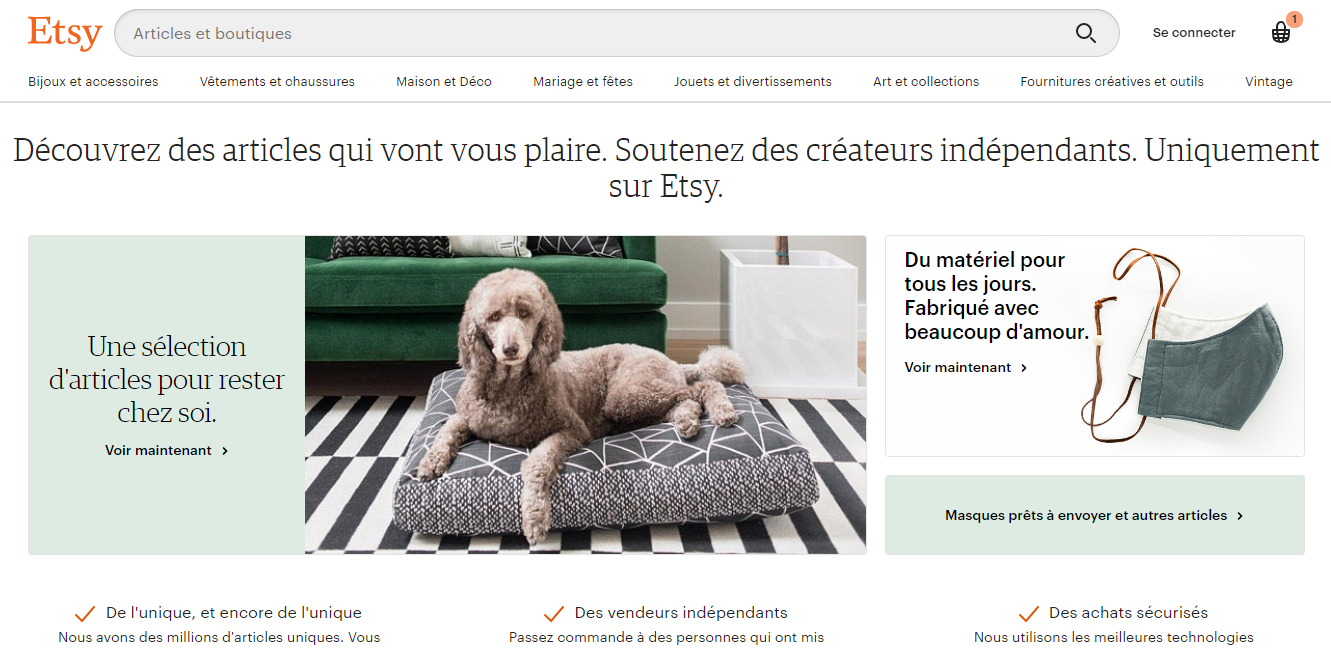
2) Etude de l’existant :

D’après une étude du marché, on a trouvé une diversité des sites web et des applications e-commerces en ligne.

On a pris à titre d’exemple les sites/applications suivantes :

2.1) Etsy:

**URL:** [**www.etsy.com**](http://www.etsy.com)

****

**Figure 1 : Etude de l’existant www.etsy.com**

**Description :**

C’est une plateforme internationale de vente de produits online qui spécialise en tous qui est habillé et décoratif.

**Inconvénients :**

- Une très grande partie de la page réservée à la publicité de telle sorte que de premier coup l’utilisateur se sont pointé sur un site publicitaire et non pas un site de vente.

- Les catégories sont présentées sous forme d’une barre des menus ce qui limite l’ajout d’autres catégories car si en ajoute plusieurs catégories nous allons obtenir soit une longue barre de menue ce qui entrainera l’apparence d’une barre de défilement horizontale dans la page.

**Avantages :**

- Une cohérence entre les tendances des couleurs choisies.

- Les liens sont clairs.

- Temps de réponse rapide lors de navigation entre les pages

2.2) Tunisianet:

**URL:** [**www.tunisianet.com.tn**](http://www.tunisianet.com.tn)

****

**Figure 2 : Etude de l’existant www.tunisianet.com.tn**

**Description :**

C’est une plateforme de vente en ligne de produits en Tunisie qui spécialise en tous au produit électroniques et informatiques.

**Inconvénients :**

- Une très grande partie de la page réservée à la publicité de telle sorte que de premier coup l’utilisateur se sont pointé sur un site publicitaire et non pas un site de vente.

- Les catégories sont présentées sous forme d’une barre des menus ce qui limite l’ajout d’autres catégories car si en ajoute plusieurs catégories nous allons obtenir soit une longue barre de menue ce qui entrainera l’apparence d’une barre de défilement horizontale dans la page.

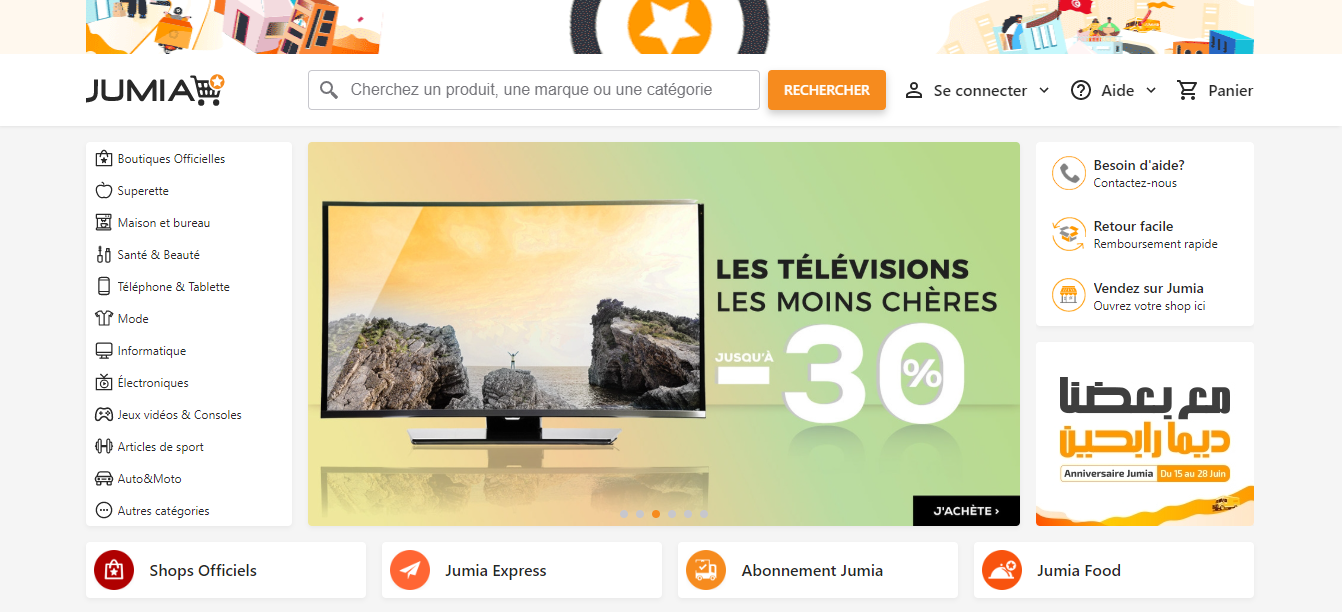
**Avantages :**

- Une cohérence entre les tendances des couleurs choisies.

- Les liens sont clairs.

2.3) jumia:

**URL:** [www.jumia.com.tn](http://www.jumia.com.tn)

****

**Figure 3 : Etude de l’existant www.jumia.com.tn**

**Description :**

C’est une plateforme tunisienne de vente en ligne de produits en Afrique.

**Inconvénients :**

- Une très grande partie de la page réservée à la publicité de telle sorte que de premier coup l’utilisateur se sont pointé sur un site publicitaire et non pas un site de vente.

**Avantages :**

- Titres claires

- Les liens sont clairs.

2.4) Conclusion :

Vu que les applications disponibles sur le marché ne sont suffisantes, On a pensé à développer une application web en incluant toutes les catégories.

On a pensé aussi d’adapter le site pour supporter les appareils mobiles et les tablettes.

3) Spécification des besoins :

Dans cette partie, on explique en détail ce que l’application est censée de faire et ceci à travers la spécification des besoins fonctionnels et non fonctionnels.

3.1) Spécification des besoins fonctionnels :

La création d’un site web passe par l’élaboration de deux parties, la première partie concerne la création des pages accessibles par tous les utilisateurs. Et la seconde partie c’est la configuration et la mise en place des espaces modérateurs et administratifs.

Exemple des pages accessibles pour tous les utilisateurs :

- Pages des produits

- Page de panier

- Page d’accueil

Exemple des pages accessibles que pour les modérateurs :

- Page de gestions de produits

- Page de gestion

Les fonctionnalités du système se compose en plusieurs parties :

3.1.1) fonctionnalités des visiteurs :

Ce sont les fonctionnalités qu’un utilisateur peut accéder sans besoin d’inscrire sur le site :

**Navigation :**

**-** Catégories

- Marques

- Prix

**Consultations des produits :**

**-** Disponibilité

- Prix

**Gestion Panier** :

- Ajout produit- Suppression produit

**Inscription** :

- Nom

- Prénom

- E-mail

- Mot de passe

- Numéro téléphone

**Authentification :**

- E-mail

- Mot de passe

3.1.2) fonctionnalités des clients :

Ce sont les fonctionnalités qu’un utilisateur doit s’inscrire peut y accéder :

**Passer commande :**

**-** Check out

**Gestion de son Compte :**

- Informations

- Historique des commandes

3.1.3) fonctionnalités des modérateurs :

Ce sont les fonctionnalités qu’un modérateur peut y accéder :

**Consultation Client**

**Gestion Produits** :

- Consultation produit

- Ajout produit

- Suppression produit

**Gestion Commandes** :

- Consultation des commandes

- Confirmer commande

- Annuler commande

3.1.4) fonctionnalités des administrateurs :

**Gestion des Clients :**

**-** Designer Client comme Modérateur

- Consultation Client

**-** Supprimer Client

3.2) Spécification des besoins non fonctionnels :

Ce sont des spécifications qui ne concernent pas spécifiquement le comportement du système mais plutôt ils identifient des contraintes internes et externes du système.

Les principaux besoins non fonctionnels de l’application se résument dans les points suivants :

**Sécurité :**

- Les comptes des utilisateurs sont sécurisés par un mot de passe.

- Chaque utilisateur accède ses informations à l’aide d’un jeton crypté.

**Performance** :

- Temps de réponse optimale à tout besoins d’utilisateurs.

**Facilité :**

**-** La navigation entre les pages doit être facile et rapide.

**Fiabilité** :

- Bon fonctionnement de l’application sans détection de défaillance.

**Convivialité** :

- Une interface claire qui désir l’internaute d’y utiliser.

- Facile à comprendre.

- Respecte l’ergonomie.

**Portabilité :**

- L’application est multiplateforme : Elle fonctionne sur tout système d’exploitation.

4) Conclusion :

Cette première partie a été dédiée à une étude détaillée de l’application : son objectif, son fonctionnement et son environnement logiciel.

On a consacré le premier mois de stage à cette étude, ce qui nous a permis de bien comprendre l’environnement de l’application, son objectif ainsi que les besoins fonctionnels et non fonctionnels.

Cette étude nous a permis de distinguer les différents points sur lesquels on va travailler et les technologies qu’on va utiliser à la réalisation de l’application.

**PARTIE III : CONCEPTION**

1) Introduction :

Dans cette partie, On reproduira les différents besoins cités précédents mais sous formes de diagrammes UML.

2) Vue fonctionnelle du système :

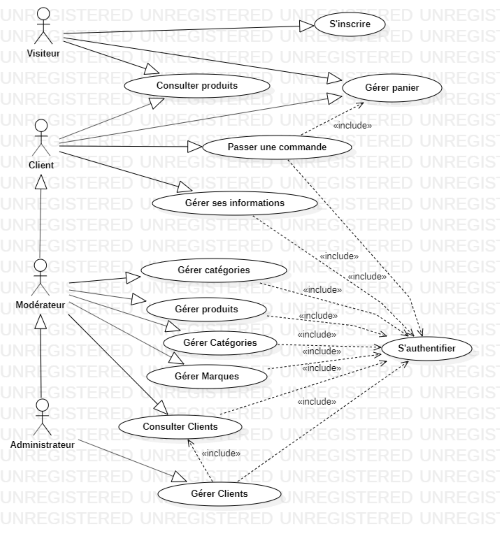
2.1) Identification d’acteurs :

Les acteurs qui interagissent avec le système sont :

* Visiteur : Tous personne qui visite le site pour consulter les produits disponibles et remplir le panier sans inscription.
* Client : Tous utilisateur inscrit pour consulter les produits et passer une commande. Il peut également gérer son profil.
* Modérateur : Tous Client privilégié de la gestion de commandes, produits et consulter les informations de Clients.
* Administrateur : Un Modérateur responsable de gérer les profils de Clients.

2.2) Diagramme de cas d’utilisation Générale :

Ce diagramme permet d’identifier les interactions possibles entre les acteurs (intervenant extérieur au système) et le système. Il représente les fonctionnalités totales que le système doit fournir.



**Figure 4 : Diagramme de cas d’utilisation Générale**

Sans authentification, le visiteur peut consulter les produits disponibles et remplir le panier. Pour devenir un Client, le visiteur doit inscrire.

Le Client doit s’authentifier pour avoir accès au fonctionnalités suivantes : gérer son profil et passer une commande.

Le Modérateur doit s’authentifier pour avoir accès au fonctionnalités Client et les fonctionnalités suivantes : gérer produits, commandes et consulter Clients.

L’Administrateur doit s’authentifier pour avoir accès au fonctionnalités Modérateur et la fonctionnalité suivante : gérer Clients.

2.3) cas d’utilisation « S’inscrire » :

**Acteur principale :** Visiteur

**Objectif :**

Ce cas d’utilisation vise à décrire toutes les étapes relatives à la l’inscription d’un Visiteur pour devenir Client.

**Préconditions :**

* Disponibilité d’accès au réseau internet.
* Serveur disponible 24/7.
* Navigateur en bonne état de fonctionnement.

**Post conditions :**

Client inscrit dans le serveur.

**Scénario nominal :**

1- L’utilisateur demande de s’inscrire.

2- Le système affiche le formulaire d’inscription.

3- L’utilisateur remplit le formulaire d’inscription et puis il valide.

4- Le système vérifie si l’utilisateur n’est pas déjà inscrit avec l’adresse émail donnée et puis enregistre le nouveau compte.

**Scénario d’exception :**

**E1 :** L’adresse e-mail déjà utilisé.

2.4) cas d’utilisation « Gérer panier » :

La figure si dessous présente le diagramme cas d’utilisation « Gérer panier » :



**Figure 5 : Diagramme de cas d’utilisation « Gérer panier »**

2.4.1) cas d’utilisation « Ajouter au panier » :

**Acteurs principaux :** Visiteur et Client.

**Objectif :**

Ce cas d’utilisation vise à décrire toutes les étapes relatives au remplissage du panier.

**Préconditions :**

* Produit disponible dans le stock.

**Post conditions :**

* Produit ajouté au panier.

**Scénario nominal :**

1- L’utilisateur cherche le produit à ajouter.

2- Le système affiche le produit ainsi que la quantité disponible.

3- L’utilisateur choisie la quantité et l’ajout dans le panier.

4- Le système vérifie l’existence d’un Client.

5- Le système sauvegarde les changements :

* Pour le Client : dans le serveur.
* Pour le Visiteur : localement.

6- Redirection vers la page de panier.

**Scénario alternatif :**

**A1 :** vérification de jeton Client au niveau de serveur.

Aller à l’opération qui suit 4

5- Le Client n’existe pas : aucun changement n’est effectué.

Aller à l’opération 6

2.4.2) cas d’utilisation « Retirer de panier » :

**Acteurs principaux :** Visiteur et Client.

**Objectif :**

Ce cas d’utilisation vise à décrire toutes les étapes relatives au vidage du panier.

**Préconditions :**

* Panier non vide.

**Post conditions :**

* Produit retiré du panier.

**Scénario nominal :**

1- L’utilisateur demande les produits ajoutés dans le panier.

2- Le système affiche les produits ajoutés dans le panier.

3- L’utilisateur choisie le produit et le supprimer.

4- Le système vérifie l’existence d’un Client.

5- Le système sauvegarde les changements :

* Pour le Client : dans le serveur.
* Pour le Visiteur : localement.

**Scénario alternatif :**

**A1 :** vérification de jeton Client au niveau de serveur.

Aller à l’opération qui suit 4

5- Le Client n’existe pas : aucun changement effectué.

2.5) cas d’utilisation « Consulter produit » :

**Acteurs principaux :** Visiteur et Client.

**Objectif :**

Ce cas d’utilisation vise à décrire toutes les étapes relatives à la consultation des produits.

**Préconditions :**

* Disponibilité d’accès au réseau internet.
* Serveur disponible 24/7.
* Navigateur en bonne état de fonctionnement.

**Post conditions :**

Produits consultés.

**Scénario nominal :**

1- L’utilisateur ouvre le site de l’application dans le navigateur.

2- Le système affiche la page d’accueil où se trouve les produits.

3- L’utilisateur clique sur un produit pour consulter sa description détaillée.

4- Le système affiche la page à consulter.

**Scénario d’exception :**

**E1 :** Erreur dans l’adresse URL :

Aller à l’opération qui suit 1

2- Le système re-directe l’utilisateur vers la page d’erreur.

**E2 :** Le produit n’existe pas

Aller à l’opération qui suit 3

4- Le système re-directe l’utilisateur vers la page d’erreur.

2.5) cas d’utilisation « Gérer ses informations » :

La figure si dessous présente le diagramme cas d’utilisation « Gérer ses informations » :



**Figure 6 : Diagramme de cas d’utilisation « Gérer ses informations »**

2.5.1) cas d’utilisation « Consulter ses informations » :

**Acteur principal :** Client

**Objectif :**

Ce cas d’utilisation vise à décrire toutes les étapes de Consultation de profil relative à un Client par lui.

**Préconditions :**

* Client authentifié.

**Post conditions :**

* Informations Consultés.

**Scénario nominal :**

1- Le Client ouvre la page de ses informations.

2- Le système affiche les informations relatives au client.

**Scénario alternatif :**

**A1 :** vérification de jeton Client au niveau de serveur.

Aller à l’opération qui suit 1

2- Client n’existe pas : le système re-directe l’utilisateur vers la page de connexion

**Scénario d’exception :**

**E1 :** Erreur dans l’adresse URL :

Aller à l’opération qui suit 1

2- Le système re-directe l’utilisateur vers la page d’erreur.

2.5.2) cas d’utilisation « Modifier ses informations » :

**Acteur principal :** Client

**Objectif :**

Ce cas d’utilisation vise à décrire toutes les étapes de Modification de profil relative au Client par lui.

**Préconditions :**

* Client authentifié.
* Informations Consulté.

**Post conditions :**

* Informations Modifiés.

**Scénario nominal :**

1- Le Client est initialement à la page de ses informations.

2- Le Client choisi l’information à changer et la change.

3- Le système valide le changement.

**Scénario alternatif :**

**A1 :** vérification de jeton Client au niveau de serveur.

Aller à l’opération qui suit 2

2- Client n’existe pas : le système re-directe l’utilisateur vers la page de connexion

**Scénario d’exception :**

**E1 :** Erreur dans l’adresse URL :

Aller à l’opération qui suit 1

1- Le système re-directe l’utilisateur vers la page d’erreur.

**E2 :** L’adresse e-mail déjà utilisé.

2.6) cas d’utilisation « Passer une commande » :

**Acteur principal :** Client

**Objectif :**

Ce cas d’utilisation vise décrire toutes les étapes relatives au passage d’une commande.

**Préconditions :**

* Panier remplit.
* Client authentifié.

**Post conditions :**

Commande passée.

**Scénario nominal :**

1- Le Client remplie le panier et valide sa commande.

2- Le système vérifie que le client est authentifié.

3- Le système affiche le formulaire de commande.

4- Le Client remplie le formulaire par son adresse, son numéro de téléphone et la méthode de paiement.

5- Le système vérifie la quantité en stock.

6- Le système enregistre la commande.

**Scénario alternatif :**

**A1 :** vérification de jeton Client au niveau de serveur.

Aller à l’opération qui suit 2

3- Le système re-directe le Client vers la page d’authentification.

**A2 :** Vérification de la quantité en stock

Aller à l’opération qui suit 5

6- Le système alerte le client et mis à jour le panier.

2.7) cas d’utilisation « S’authentifier » :

**Acteur principal :** Visiteur

**Objectif :**

Ce cas d’utilisation vise à décrire toutes les étapes relatives à l’authentification d’un utilisateur au niveau de serveur afin d’effectuer des opérations supplémentaires.

**Préconditions :**

* Disponibilité d’accès au réseau internet.
* Serveur disponible 24/7.
* Navigateur en bonne état de fonctionnement.

**Post conditions :**

* Sauvegarde de jeton d’authentification au niveau de navigateur.
* Panier mis à jour au niveau de serveur.

**Scénario nominal :**

1- L’utilisateur ouvre la page d’authentification.

2- Le système affiche le formulaire d’authentification.

3- L’utilisateur remplit le formulaire par son e-mail et mot de passe et valide

4- Le système vérifie l’adresse et le mot de passe.

5- Le système envoie le jeton d’authentification.

6- Le système mis à jour le panier au niveau de serveur.

**Scénario alternatif :**

**A1 :** Vérification d’e-mail et mot de passe au niveau de serveur

Aller à l’opération qui suit 4

5- Le système affiche un message d’erreur.

**Scénario d’exception :**

**E1 :** Erreur dans l’adresse URL :

Le système re-directe l’utilisateur vers la page d’erreur.

2.7) cas d’utilisation « Consulter Clients » :

**Acteur principal :** Modérateur

**Objectif :**

Ce cas d’utilisation vise à décrire toutes les étapes relatives à la consultation des Clients.

**Préconditions :**

Modérateur authentifié.

**Post conditions :**

Clients consultés.

**Scénario nominal :**

1- Le Modérateur ouvre la page Clients.

2- Le système vérifie le jeton de modérateur.

3- Le système affiche les informations relatives aux Clients.

**Scénario alternatif :**

**A1 :** Vérification de jeton d’utilisateur

Aller à l’opération qui suit 2

3- Le système re-directe l’utilisateur vers la page d’accueil.

**Scénario d’exception :**

**E1 :** Erreur dans l’adresse URL :

Le système re-directe l’utilisateur vers la page d’erreur.

2.8) cas d’utilisation « Gérer produits » :

La figure si dessous présente le diagramme cas d’utilisation « Gérer produits » :



**Figure 7 : Diagramme de cas d’utilisation « Gérer produits »**

2.8.1) cas d’utilisation « Supprimer produits » :

**Acteur principal :** Modérateur

**Objectif :**

Ce cas d’utilisation vise à décrire toutes les étapes relatives à la suppression d’un produit

**Préconditions :**

Modérateur authentifié.

**Post conditions :**

Produit supprimé.

**Scénario nominal :**

1- Le Modérateur ouvre la page des produits

2- Le système vérifie le jeton de modérateur.

3- Le système affiche les informations relatives aux produits.

4- Le modérateur choisie le produit à supprimer et valide.

5- Le système vérifie le jeton de modérateur.

**Scénario alternatif :**

**A1 :** L’utilisateur n’est pas un Modérateur.

-l’utilisateur se re-directe vers la page d’accueil.

**A2 :** L’utilisateur n’est pas un client.

- l’utilisateur se re-directe vers la page de connexion.

**Scénario d’exception :**

**E1 :** Erreur dans l’adresse URL :

Le système re-directe l’utilisateur vers la page d’erreur.

2.8.2) cas d’utilisation « Modifier produits » :

**Acteur principal :** Modérateur

**Objectif :**

Ce cas d’utilisation vise à décrire toutes les étapes relatives à la modification d’un produit

**Préconditions :**

* Modérateur authentifié.
* Produit existe à la base.

**Post conditions :**

Produit modifié.

**Scénario nominal :**

1- Le Modérateur ouvre la page des produits

2- Le système vérifie le jeton de modérateur.

3- Le système affiche les informations relatives aux produits.

4- Le modérateur choisie le produit à modifier.

5- Le système affiche les informations à modifier.

6- L’utilisateur modifie les informations et confirme.

7- Le système vérifie le jeton de modérateur.

8- Le système valide la modification.

**Scénario alternatif :**

**A1 :** L’utilisateur n’est pas un Modérateur.

-l’utilisateur se re-directe vers la page d’accueil.

**A2 :** L’utilisateur n’est pas un client.

- l’utilisateur se re-directe vers la page de connexion.

**Scénario d’exception :**

**E1 :** Erreur dans l’adresse URL :

Le système re-directe l’utilisateur vers la page d’erreur.

2.8.3) cas d’utilisation « Ajouter produits » :

**Acteur principal :** Modérateur

**Objectif :**

Ce cas d’utilisation vise à décrire toutes les étapes relatives à l’ajout d’un produit

**Préconditions :**

Modérateur authentifié.

**Post conditions :**

Produit Ajouté.

**Scénario nominal :**

1- Le Modérateur ouvre la page des produits et demande un ajout.

2- Le système affiche le formulaire d’ajout

3- Le Modérateur remplie le formulaire et confirme

4- Le système vérifie le jeton de modérateur.

5- Le système valide l’ajout.

**Scénario alternatif :**

**A1 :** L’utilisateur n’est pas un Modérateur.

-l’utilisateur se re-directe vers la page d’accueil.

**A2 :** L’utilisateur n’est pas un client.

- l’utilisateur se re-directe vers la page de connexion.

**Scénario d’exception :**

**E1 :** Erreur dans l’adresse URL

* Le système re-directe l’utilisateur vers la page d’erreur.

**E2 :** le formulaire est mal-remplit

2.9) cas d’utilisation « Gérer commandes » :

La figure si dessous présente le diagramme cas d’utilisation « Gérer commandes » :



**Figure 8 : Diagramme de cas d’utilisation « Gérer commandes »**

2.9.1) cas d’utilisation « Confirmer commandes » :

**Acteur principal :** Modérateur

**Objectif :**

Ce cas d’utilisation vise à décrire toutes les étapes relatives à la confirmation d’une commande.

**Préconditions :**

* Modérateur authentifié.
* Commande lancée.

**Post conditions :**

Commande confirmée.

**Scénario nominal :**

1- Le Modérateur ouvre la page des commandes.

2- Le système affiche la liste des commandes.

3- Le Modérateur confirme une commande.

4- Le système vérifie le jeton de modérateur.

5- Le système valide la confirmation.

**Scénario alternatif :**

**A1 :** L’utilisateur n’est pas un Modérateur.

* L’utilisateur se re-directe vers la page d’accueil.

**A2 :** L’utilisateur n’est pas un client.

* L’utilisateur se re-directe vers la page de connexion.

**Scénario d’exception :**

**E1 :** Erreur dans l’adresse URL

* Le système re-directe l’utilisateur vers la page d’erreur.

2.9.2) cas d’utilisation « Annuler commandes » :

**Acteur principal :** Modérateur

**Objectif :**

Ce cas d’utilisation vise à décrire toutes les étapes relatives à l’annulation d’une commande.

**Préconditions :**

* Modérateur authentifié.
* Commande lancée.

**Post conditions :**

Commande annulée.

**Scénario nominal :**

1- Le Modérateur ouvre la page des commandes.

2- Le système affiche la liste des commandes.

3- Le Modérateur annule une commande.

4- Le système vérifie le jeton de modérateur.

5- Le système valide l’annulation.

**Scénario alternatif :**

**A1 :** L’utilisateur n’est pas un Modérateur.

* L’utilisateur se re-directe vers la page d’accueil.

**A2 :** L’utilisateur n’est pas un client.

* L’utilisateur se re-directe vers la page de connexion.

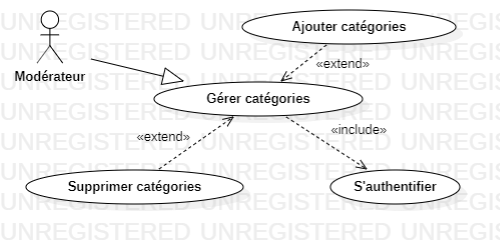
**Scénario d’exception :**

**E1 :** Erreur dans l’adresse URL

* Le système re-directe l’utilisateur vers la page d’erreur.

2.10) cas d’utilisation « Gérer Catégories » :

La figure si dessous présente le diagramme cas d’utilisation « Gérer catégories » :



**Figure 9 : Diagramme de cas d’utilisation « Gérer catégories »**

2.10.1) cas d’utilisation « Ajouter Catégories » :

**Acteur principal :** Modérateur

**Objectif :**

Ce cas d’utilisation vise à décrire toutes les étapes relatives à l’ajout d’une catégorie.

**Préconditions :**

Modérateur authentifié.

**Post conditions :**

Catégorie ajoutée.

**Scénario nominal :**

1- Le Modérateur demande l’ajout d’une catégorie.

2- Le système affiche le formulaire d’ajout.

3- Le Modérateur le remplie et confirme.

4- Le système vérifie le jeton de modérateur.

5- Le système valide l’ajout.

**Scénario alternatif :**

**A1 :** L’utilisateur n’est pas un Modérateur.

* L’utilisateur se re-directe vers la page d’accueil.

**A2 :** L’utilisateur n’est pas un client.

* L’utilisateur se re-directe vers la page de connexion.

**Scénario d’exception :**

**E1 :** Erreur dans l’adresse URL

* Le système re-directe l’utilisateur vers la page d’erreur.

2.10.2) cas d’utilisation « Supprimer Catégories » :

**Acteur principal :** Modérateur

**Objectif :**

Ce cas d’utilisation vise à décrire toutes les étapes relatives à la suppression d’une catégorie.

**Préconditions :**

Modérateur authentifié.

**Post conditions :**

Catégorie supprimée.

**Scénario nominal :**

1- Le Modérateur demande la suppression d’une catégorie.

2- Le système vérifie le jeton de modérateur.

3- Le système valide la suppression.

**Scénario alternatif :**

**A1 :** L’utilisateur n’est pas un Modérateur.

* L’utilisateur se re-directe vers la page d’accueil.

**A2 :** L’utilisateur n’est pas un client.

* L’utilisateur se re-directe vers la page de connexion.

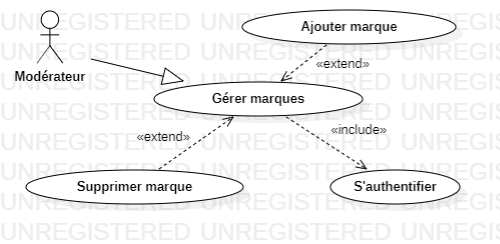
**Scénario d’exception :**

**E1 :** Erreur dans l’adresse URL

* Le système re-directe l’utilisateur vers la page d’erreur.

2.11) cas d’utilisation « Gérer Marques » :

La figure si dessous présente le diagramme cas d’utilisation « Gérer marques » :



**Figure 10 : Diagramme de cas d’utilisation « Gérer marques »**

2.11.1) cas d’utilisation « Ajouter Marque » :

**Acteur principal :** Modérateur

**Objectif :**

Ce cas d’utilisation vise à décrire toutes les étapes relatives à l’ajout d’une marque.

**Préconditions :**

Modérateur authentifié.

**Post conditions :**

Marque ajoutée.

**Scénario nominal :**

1- Le Modérateur demande l’ajout d’une marque.

2- Le système affiche le formulaire d’ajout.

3- Le Modérateur le remplie et confirme.

4- Le système vérifie le jeton de modérateur.

5- Le système valide l’ajout.

**Scénario alternatif :**

**A1 :** L’utilisateur n’est pas un Modérateur.

* L’utilisateur se re-directe vers la page d’accueil.

**A2 :** L’utilisateur n’est pas un client.

* L’utilisateur se re-directe vers la page de connexion.

**Scénario d’exception :**

**E1 :** Erreur dans l’adresse URL

* Le système re-directe l’utilisateur vers la page d’erreur.

2.10.2) cas d’utilisation « Supprimer Marque » :

**Acteur principal :** Modérateur

**Objectif :**

Ce cas d’utilisation vise à décrire toutes les étapes relatives à la suppression d’une marque.

**Préconditions :**

Modérateur authentifié.

**Post conditions :**

Marque supprimée.

**Scénario nominal :**

1- Le Modérateur demande la suppression d’une marque.

2- Le système vérifie le jeton de modérateur.

3- Le système valide la suppression.

**Scénario alternatif :**

**A1 :** L’utilisateur n’est pas un Modérateur.

* L’utilisateur se re-directe vers la page d’accueil.

**A2 :** L’utilisateur n’est pas un client.

* L’utilisateur se re-directe vers la page de connexion.

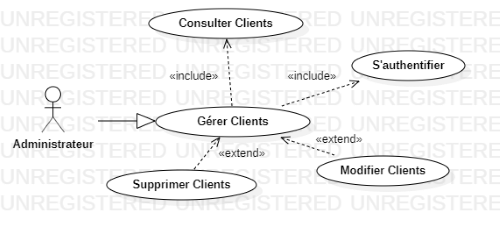
**Scénario d’exception :**

**E1 :** Erreur dans l’adresse URL

* Le système re-directe l’utilisateur vers la page d’erreur.

2.12) cas d’utilisation « Gérer Clients » :

La figure si dessous présente le diagramme cas d’utilisation « Gérer Clients » :



**Figure 11 : Diagramme de cas d’utilisation « Gérer Clients »**

2.12.1) cas d’utilisation « Supprimer Clients » :

**Acteur principal :** Administrateur

**Objectif :**

Ce cas d’utilisation vise à décrire toutes les étapes relatives à la suppression d’un client.

**Préconditions :**

* Administrateur authentifié.
* Clients consultés.

**Post conditions :**

Client supprimé.

**Scénario nominal :**

1- L’Administrateur ouvre la page des Clients.

2- Le système affiche la liste des Clients.

3- L’Administrateur demande la suppression d’un Client.

4- Le système vérifie le jeton de l’Administrateur.

5- Le système valide la suppression.

**Scénario alternatif :**

**A1 :** L’utilisateur n’est pas un client.

* L’utilisateur se re-directe vers la page de connexion.

**Scénario d’exception :**

**E1 :** Erreur dans l’adresse URL

* Le système re-directe l’utilisateur vers la page d’erreur.

**E2 :** L’utilisateur n’est pas un Administrateur.

* Une erreur s’affiche.

2.12.2) cas d’utilisation « Modification Clients » :

**Acteur principal :** Administrateur

**Objectif :**

Ce cas d’utilisation vise à décrire toutes les étapes relatives à la Modification d’un client.

**Préconditions :**

* Administrateur authentifié.
* Clients consultés.

**Post conditions :**

Client Modifié.

**Scénario nominal :**

1- L’Administrateur ouvre la page des Clients.

2- Le système affiche la liste des Clients.

3- L’Administrateur demande la Modification d’un Client.

4- Le système affiche le formulaire de modification.

5- L’Administrateur remplie le formulaire et confirme.

6- Le système vérifie le jeton de l’Administrateur.

7- Le système valide la modification.

**Scénario alternatif :**

**A1 :** L’utilisateur n’est pas un client.

* L’utilisateur se re-directe vers la page de connexion.

**Scénario d’exception :**

**E1 :** Erreur dans l’adresse URL

* Le système re-directe l’utilisateur vers la page d’erreur.

**E2 :** L’utilisateur n’est pas un Administrateur.

* Une erreur s’affiche.

3) Vue statique du système :

Le diagramme de classe exprime la structure statique du système en termes de classes et relations entre eux. L’intérêt du diagramme de classe est de modéliser les entités systèmes d’informations.

Le diagramme de classes permet de représenter l’ensemble des informations finalisées qui sont gérées par le domaine. Ces informations sont structurées, c’est-à-dire qu’elles sont regroupées dans des classes.

Ce diagramme met en évidence d’éventuelles relations entre les classes.

Le diagramme de classe de mon application est le suivant :



**Figure 12 : Diagramme de classes**

4) Vue dynamique du système :

4.1) Diagrammes de séquences :

Un diagramme de séquence représente les objets participants à une interaction particulière et les messages qu’ils échange, organise en séquences horaires.

4.1.1) Diagramme de séquence « Inscription » :

La figure si dessous présente le diagramme de séquences d’inscription :



**Figure 13 : Diagramme de séquences « Inscription »**

4.1.2) Diagramme de séquence « Authentification » :

La figure si dessous présente le diagramme de séquences d’Authentification :



**Figure 14 : Diagramme de séquences « Authentification »**

4.1.3) Diagramme de séquence « Gérer panier » :

La figure à la page suivante présente le diagramme de séquences relative à la gestion de panier :



**Figure 15 : Diagramme de séquences « Gérer panier »**

4.2) Diagrammes de séquences détaillés :

Un diagramme de séquence est un diagramme d’interaction qui expose en détail la façon dont les opérations sont effectuées.

A la page suivante, On a présenté les différents diagrammes de séquences détailles :

4.2.1) Diagramme de séquence détaillé « Inscription » :

La figure si dessous présente le diagramme de séquences détaillé d’inscription :

****

**Figure 16 : Diagramme de séquences « Inscription »**

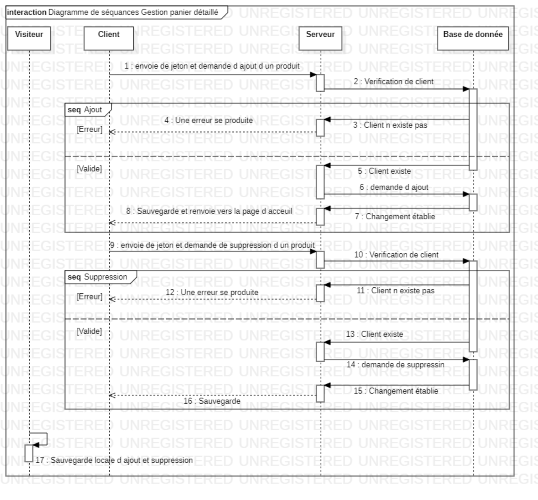
4.2.2) Diagramme de séquence détaillé « Authentification » :

La figure si dessous présente le diagramme de séquences détaillé d’Authentification :

**Figure 17 : Diagramme de séquences détaillé « Authentification »**

4.2.3) Diagramme de séquence détaillé « Gérer panier » :

La figure si dessous présente le diagramme de séquences détaillé relative à la gestion de panier :



**Figure 18 : Diagramme de séquences détaillé « Gérer panier »**

5) Conception détaillé de la base de données :

5.1) Modèle conceptuel des données :

Ce modèle a pour but d’écrire d’une façon formelle les données qui seront utilisés par le système d’information. Il s’agit donc d’une représentation des données, facilement compréhensible, permettant à décrire le système d’information à l’aide d’entités.

Si dessous, On présente le diagramme Entité-Association de système :

****

**Figure 19 : Diagramme Entités-Associations**

5.2) Modèle logique des données :

Ce modèle indique comment organiser les données. On représente ce modèle comme étant un ensemble de relations. On a des tables qui contiennent des attributs. Le plus important est la relation entre les tables. Ainsi la traduction du modèle conceptuel des données en modèle logique se forme du schéma conceptuel suivant :

* Client (**ID**, Nom, Prenom, Email, MotDePasse, DateCreation, NumeroTele, Admin, Mod, Panier)
* Categorie (**IDCa**, NomCa)
* Marque (**IDM**, NomM)
* Produit (**IDP**, NomP, PrixP, QuantiteP, DateCreationP, NomM#, NomCa#)
* Commande (**IDCom**, DescriptionCom, PrixCom, DateCreationCom, ID#)

5.3) Dictionnaire de données :

Le dictionnaire suivant illustre la liste exhaustive des diffèrent attributs :

|  |  |  |  |
| --- | --- | --- | --- |
| **Nom** | **Désignation** | **Type** | **Identifiant** |
| ID | Identifiant du client. | String | Oui |
| Nom | Nom du client. | String | Non |
| Prenom | Prénom du client. | String | Non |
| Email | Adresse électronique du client. | String | Non |
| MotDePasse | Mot de passe du client. | String | Non |
| DateCreation | Date de création de compte client. | Date | Non |
| NumeroTele | Numéro de téléphone du client. | Integer | Non |
| Admin | Identifie si le client est un administrateur ou non. | Boolean | Non |
| Mod | Identifie si le client est un modérateur au cas où il n’est pas un administrateur. | Boolean | Non |
| Panier | Identifie les produits sauvegardés dans le panier de client | Array | Non |
| IDCa | Identifiant de la catégorie. | String | Oui |
| NomCa | Nom de la catégorie. | String | Non |
| IDM | Identifiant de la marque. | String | Oui |
| NomM | Nom de la marque. | String | Non |
| IDP | Identifiant du produit. | String | Oui |
| NomP | Nom du produit. | String | Non |
| PrixP | Prix du produit. | Integer | Non |
| QuantiteP | Quantité disponible de produit dans le stock. | Integer | Non |
| DateCreationP | Date de création de produit. | Date | Non |
| IDCom | Identifiant de la commande. | String | Oui |
| DescriptionCom | Description de la commande (produits commandées, quantité de chaque produits, …). | String | Non |
| PrixCom | Prix totaux de la commande. | Integer | Non |
| DateCreationCom | Date de création de la commande. | Date | Non |

6) Conclusion :

Dans cette partie, On a présenté l’étude conceptuelle du système. La vue fonctionnelle a été illustrée par les diagrammes de cas d’utilisation. Ensuite, la vue statique, réalisé par le diagramme des classes qui nous a permis de définir la structure du système et de dégager les différentes entités qui le compose. Dernièrement, la vue dynamique nous a permis d’avoir une vue générale sur le déroulement des cas d’utilisation et leurs exécutions. Cette vue a été modélisé par des diagrammes de séquence du système puis des diagrammes de séquences détaillés afin de modéliser l’interaction des différents objets du système et décrivant ainsi son comportement.

**PARTIE IV : REALISATION**

1) Introduction :

Au niveau de cette dernière partie, nous allons énumérer les outils softwares et hardware qu’On a utilisés pour réaliser notre site ainsi que ses principales interfaces.

2) Environnement de travail :

Il s’agit dans cette partie d’identifier les différentes caractéristiques de l’environnements matériel et logiciels qui nous ont servi à l’implémentation de notre application.

2.1) Environnement matériel :

La machine utilisée pour réaliser ce projet dispose des caractéristiques suivantes :

- Hôte : Lenovo **IdeaPad 130-15IKB**

**- Système d’exploitation : Windows 10**

- Microprocesseur : Intel® Core™ i3-6006U

- RAM : 4 Go

- Disque dur : 1 To

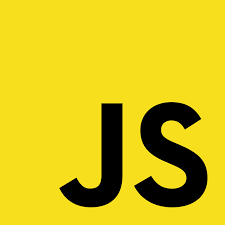
2.2) Environnement Logiciels :

**Visual Studio Code :** c’est un éditeur de texte extensible développé par Microsoft pour Windows, Linux et MacOs. C’est un éditeur multiplateforme, open source et gratuit, supportant une dizaine de langages.

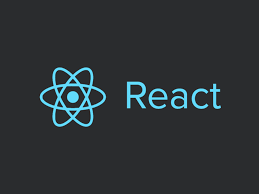
**Postman :** c’est un logiciel de test des **API** pour déterminer la fonctionnalité, fiabilité, performance et sécurité d’une application.

**StarUML :** est un logiciel de modélisation [UML](https://fr.wikipedia.org/wiki/UML_(informatique)) open source, qui permet également d’exporter les modèles sous les format JPEG, WMF, SVG et PNG.

2.3) Langage de programmation :

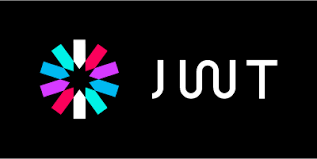
**JavaScript :** c’est un [langage de programmation](https://fr.wikipedia.org/wiki/Langage_de_programmation) de [scripts](https://fr.wikipedia.org/wiki/Langage_de_script) principalement employé dans les [pages web](https://fr.wikipedia.org/wiki/Pages_web) interactives mais aussi pour les serveurs avec l'utilisation de [**Node.js**](https://fr.wikipedia.org/wiki/Node.js).

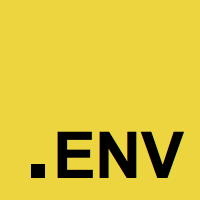
2.4) Bibliothèques :

**React.js :**  c’est une [bibliothèque](https://fr.wikipedia.org/wiki/Biblioth%C3%A8que_Logicielle) JavaScript [libre](https://fr.wikipedia.org/wiki/Logiciel_libre) développée par [Facebook](https://fr.wikipedia.org/wiki/Facebook) qui, facilite la création de l’applications web mono-pages, via la création de composants dépendant d'un état et générant une page (ou portion) [**HTML**](https://fr.wikipedia.org/wiki/HTML) à chaque changement d'état.

**Mongoose :** c’est un ORM pour la base de données **MongoDB**. Il fournit une solution simple basée sur des schémas pour modéliser les données d’une application.

**Bcrypt :** c’est une fonction de hachage basée sur l’algorithme de chiffrement BlowFish, En plus de l'utilisation d'un sel pour se protéger des attaques par [table arc-en-ciel](https://fr.wikipedia.org/wiki/Table_arc-en-ciel). C’est une fonction adoptive, c'est-à-dire que l'on peut augmenter le nombre d'itérations pour la rendre plus lente.

**Jwt :** (ou bien **JSON Web Token**) c’est un standard ouvert qui permet l’échange sécurisé de jetons entre plusieurs parties. Cette bibliothèque nous a été utile pour l’échange entre client et serveur.

**Dotenv :** bibliothèque utilisée pour charger automatiquement les variables spéciales (clés secrètes par exemple) à partir d’un fichier caché d’une extension env.

**Cors:** (ou bien **Cross-Origin Resource Sharing**) c’est un mécanisme qui permet à des [ressources](https://fr.wikipedia.org/wiki/Ressources_web) restreintes d'une [page web](https://fr.wikipedia.org/wiki/Page_web), d'être récupérées par un autre [domaine](https://fr.wikipedia.org/wiki/Nom_de_domaine) extérieur au domaine à partir duquel la première ressource a été servie.

**Multer :** c’est un middleware [**Node.js**](https://fr.wikipedia.org/wiki/Node.js)pour la gestion de multipart/form-data, qui est principalement utilisé pour télécharger des fichiers sous le serveur.

**Axios :** c’est client HTTP pour **JavaScript** qui permet l’envoie de requêtes asynchrones et effectuer des opérations CRUD.

2.5) Framework :

**Express.js :** c’ est un [Framework](https://fr.wikipedia.org/wiki/Framework) pour construire des [applications web](https://fr.wikipedia.org/wiki/Applications_web) basées sur [**Node.js**](https://fr.wikipedia.org/wiki/Node.js). C'est de fait le Framework standard pour le développement de serveur en [**Node.js**](https://fr.wikipedia.org/wiki/Node.js).

2.6) Environnement de développement :

[**Node.js**](https://fr.wikipedia.org/wiki/Node.js)**:** c’est un environnement d'exécution**,** open-source, multi-plateforme qui exécute le code **Javascript** à l'extérieur de navigateur web. Il permet aux développeurs d’utiliser **Javascript** pour écrire des lignes de commandes et développer un serveur HTTP.

**npm :** c’est un générateur de paquets installé par default sous [**Node.js**](https://fr.wikipedia.org/wiki/Node.js)**.** Il fonctionne avec un terminal qui gère les dépendances de l’application.

**Git :** est un [logiciel de gestion de versions](https://fr.wikipedia.org/wiki/Logiciel_de_gestion_de_versions) libre [décentralisé](https://fr.wikipedia.org/wiki/Gestion_de_versions#Gestion_de_versions_d.C3.A9centralis.C3.A9e) qu’ On a utilisé pour sauvegarder notre référentiel en ligne sous le site **GitHub**.

2.7) Design :



**Material-UI :** c’est unensemble de composants de **React.js** pour faciliter le développement des interfaces web.

**HTML :** (ou bien **HyperText Markup Language**) c’est le langage de balisage conçu pour représenter les pages web qui permet également de structurer [sémantiquement](https://fr.wikipedia.org/wiki/S%C3%A9mantique) la page, de mettre en forme le contenu, de créer des formulaires de saisie, d’inclure des [ressources](https://fr.wikipedia.org/wiki/Ressource_du_World_Wide_Web) [multimédias](https://fr.wikipedia.org/wiki/Multim%C3%A9dia) dont des [images](https://fr.wikipedia.org/wiki/Image_num%C3%A9rique), des [vidéos](https://fr.wikipedia.org/wiki/Vid%C3%A9o), et des programmes informatiques.



**CSS :** (ou bien **Cascading Style Sheets**) c’est un [langage informatique](https://fr.wikipedia.org/wiki/Langage_informatique) qui décrit la présentation des documents [HTML](https://fr.wikipedia.org/wiki/Hypertext_markup_language) et [XML](https://fr.wikipedia.org/wiki/Extensible_markup_language) à partir de feuilles de styles en cascade.

2.8) Système de gestion de base de données :

**MongoDB :** MongoDB est un système de gestion de base de données orienté documents, répartissable sur un nombre quelconque d'ordinateurs et ne nécessitant pas de schéma prédéfini des données.