**Chapitre 02 : contexte général et environnement de développement**

Table des matières

[I. Contexte général de projet 3](#_Toc118923760)

[1- Introduction 3](#_Toc118923761)

[2- La problématique 3](#_Toc118923762)

[3- Etude de l’existant 3](#_Toc118923763)

[4- Etude de besoin : 5](#_Toc118923764)

[5- Cahier des charges 10](#_Toc118923765)

[**5.1-** **Présentation de l’organisation** 10](#_Toc118923766)

[**5.2-** **Objectif de la plateforme** 10](#_Toc118923767)

[**5.3-** **La société ciblée** 10](#_Toc118923768)

[**5.4-** **Exemple de plateforme e-santé** : 10](#_Toc118923769)

[**5.5-** **Spécification des besoins** 10](#_Toc118923770)

[**5.5.1-** **Les besoins fonctionnels** 10](#_Toc118923771)

[**5.5.2-** **Les besoins non fonctionnels** 11](#_Toc118923772)

[**5.6-** **Graphisme et Ergonomie** 12](#_Toc118923773)

[**5.6.1-** **Le nom de la plateforme** 12](#_Toc118923774)

[**5.6.2-** **La palette de couleur** 12](#_Toc118923775)

[**5.6.3-** **Logo** 12](#_Toc118923776)

[**5.6.4-** **Police du texte à utiliser** 13](#_Toc118923777)

[**5.7-** **Environnement et outils** 13](#_Toc118923778)

[**5.8-** **Planning** 13](#_Toc118923779)

[6- Conclusion 13](#_Toc118923780)

[II. Généralité et environnement de développement 14](#_Toc118923781)

[1- Introduction 14](#_Toc118923782)

[2- Définitions et généralités. 14](#_Toc118923783)

[**2.1-** **Application web** 14](#_Toc118923784)

[**2.2-** **Logiciel** 14](#_Toc118923785)

[3- Les technologies et langages de développement 14](#_Toc118923786)

[**3.1-** **HTML** 14](#_Toc118923787)

[**3.2-** **CSS** 15](#_Toc118923788)

[**3.3-** **SASS** 15](#_Toc118923789)

[**3.4-** **Bootstrap** 15](#_Toc118923790)

[**3.5-** **JavaScript** 15](#_Toc118923791)

[**3.6-** **Type Script** 15](#_Toc118923792)

[**3.7-** **ANGULAR et Angular matériel** 15](#_Toc118923793)

[**3.8-** **ELECTRONJS** 17](#_Toc118923794)

[**3.9-** **Framework Ionic** 17](#_Toc118923795)

[**3.10-** **Spring boot** 17](#_Toc118923796)

[4- Les technologies et langages de modélisation 17](#_Toc118923797)

[**3.1.** **UML** 17](#_Toc118923798)

[5- Les systèmes de gestion de bases de données 17](#_Toc118923799)

[**5.1-** **PostgreSQL** 18](#_Toc118923800)

[6- Les environnements de développement 18](#_Toc118923801)

[**6.1-** **Bgadmin** 18](#_Toc118923802)

[**6.2-** **Visual Studio code** 18](#_Toc118923803)

[**6.3-** **Visual Paradigm** 18](#_Toc118923804)

[7- Démarche pattern de développement et approche de gestion de projets 18](#_Toc118923805)

[**8.1-** **Approche de gestion de projet** 18](#_Toc118923806)

[**8.2-** **L’architecteur MVC** 18](#_Toc118923807)

[8- Conclusion 18](#_Toc118923808)

Liste des abréviations

MVC

SPA

UAMB

XAMPP

UML

HTML

CSS

JS

TS

SGBD

# Contexte général de projet

1. **Introduction**

Dans cette section de notre mémoire nous avons comme objectif de présenter le contexte général de notre projet, nous allons commencer par la problématique et les raisons pour lesquelles nous avons eu l’idée de se lancer dans ce projet. Puis, nous allons faire preuve de quelque sondage pour lancer dans l’étude de l’existent et des besoins. En fin, nous allons rédiger un cahier de charge qui contient toutes les informations nécessaires et les plannings.

1. **La problématique**

Partout dans le monde, nous assistons sur le développement et à la digitalisation du secteur de la santé. La pandémie Du COVID-19 nous a montré qu’on doit avoir une plateforme numérique de santé qui a plusieurs utilités comme la prévention, la communication, la sensibilisation sur plusieurs sujets et améliorer la Gestion opérationnelle ainsi que le quotidien des praticiens de santé et des patients. Donc plusieurs pays, ont commencé la digitalisation de ce secteur alors que dans notre pays la majorité utilisant des anciennes stratégies à ce jour, ce qui Cause un certain nombre de difficultés :

* Difficultés de prise de rendez-vous pour le patient.
* Problème de localisation des médecins.
* Difficultés de sensibilisation.
* Problèmes de collecte des informations sur les Maladies dont le patient souffre auparavant.
* Le cabinet médical a une difficulté de Géré ces rendez-vous.

Est ça nous pousse à poser quelles questions, à ce qu’il est nécessaire de crée une plateformes e-santé ici en Algérie ? Et quelles sont les fonctionnalités qu’elle doit contenir cette dernière ?

1. **Etude de l’existant**

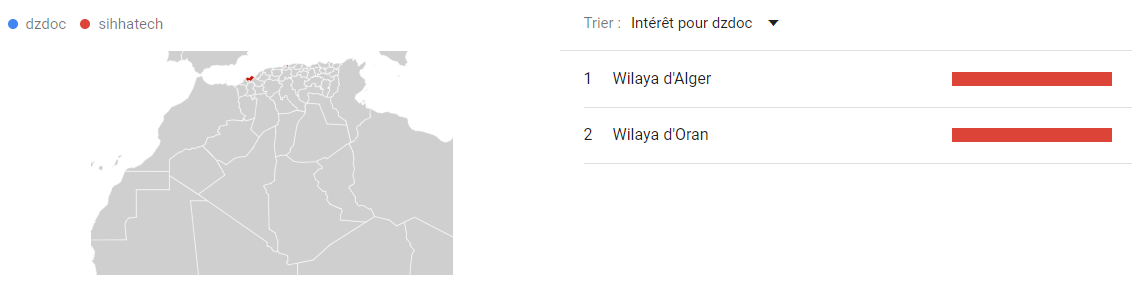
Nous avons cité auparavant que l’e-santé a vu une énorme évolution ses derniers temps dans le monde entier et sur tout dans les pays développer. Et nous allons citer quelques statistiques pour voir exactement le taux d’usage des plateformes e-santé en Algérie et en Europe.

* Le taux de recherche sur des plateformes e-santé en Algérie :

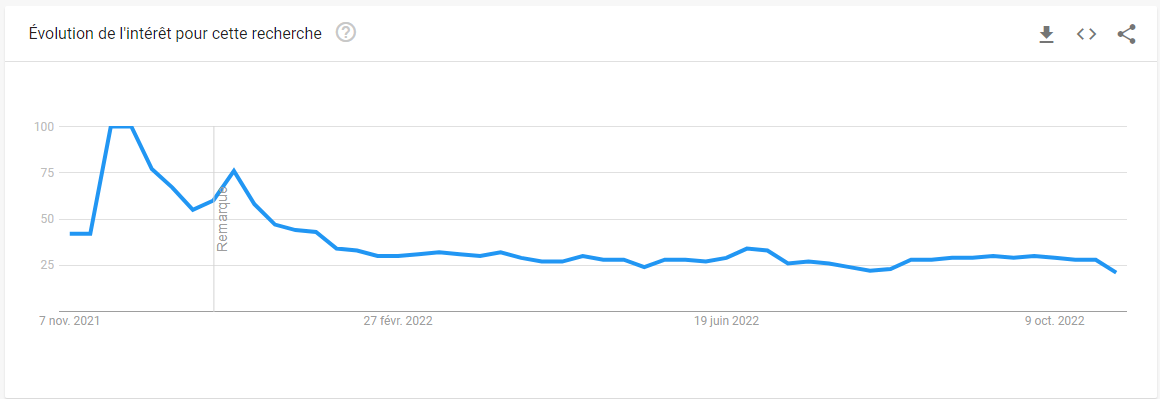


**Figure 1**: diffractogramme représente le moyen de recherche sur les plateformes e-santé en Algérie (dzdoc et sihhatech)

* Répartition des taux de recherche par région en Algérie :

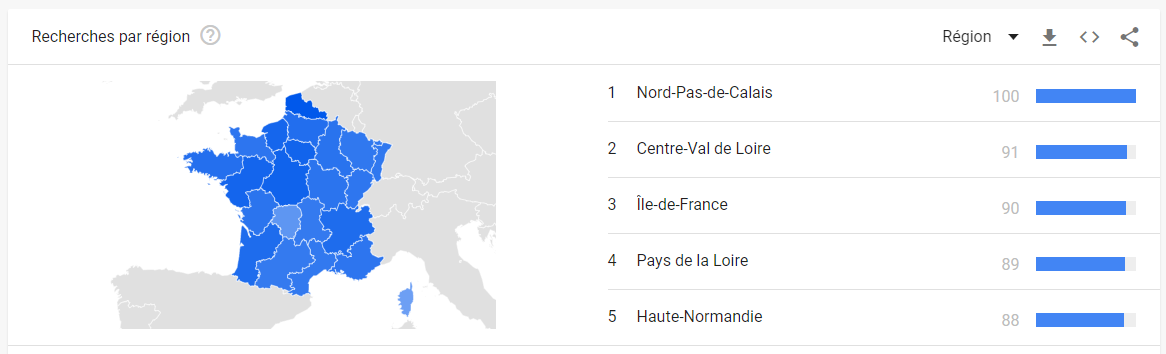


**Figure 2**: Répartition des taux de recherche sur les plateformes e-santé par région

* Le taux de recherche sur des plateformes e-santé en France (doctolib) :

**Figure 3**: diffractogramme représente le moyen de recherche sur les plateformes e-santé en France (Doctolib)

* Répartition des taux de recherche sur **Doctolib** par région en France :



**Figure 4**: Répartition des taux de recherche sur les plateformes Doctolib e-santé par région en france

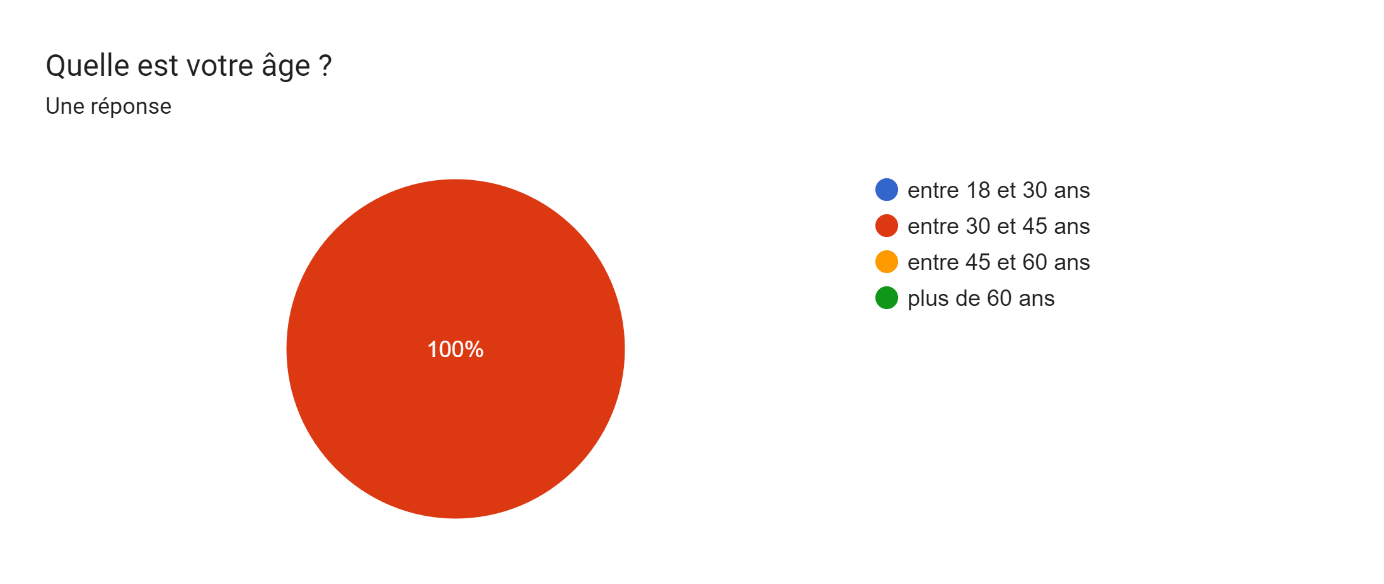
On regarde ses statistiques en peut déduire que le taux de recherche et également d’usage des plateformes d’e-santé en Algérie est très faible par apport à la France, ce que nous posse à mettre l’accent sur les critères de qualité de ses plateformes, pour pouvoir faire quelque chose de défirent et d’utilisable par toutes les catégories de la société.

1. **Etude de besoin :**

Dans le but de définir notre cible et les besoins du public. Nous avons eu l’idée de partager deux sondages réaliser avec « **Google Forms** », l’un destiner à tous les membres de la société et l’autre aux les professionnels de la sante (vous les trouvez dans l’annexe de ce mémoire).

**Utilisation de l’outil GOOGLE Forms :**

Le sondage a été réalisée sur un échantillon de 0 personne entre la période de 25/10/2022 et 25/11/2022. Les questions posées nous permettent de voir la vision de la société algérienne et de déterminer les fonctionnalités nécessaires dans notre plateforme.



Statistique représentant les déférant catégorie intéresser à une plateforme e-santé

|  |  |
| --- | --- |
| Bejaia | 5 |
| Jijel | 2 |
| … | … |
| … | … |
|  |  |

Statistique représentant les résultats des personnes intéresser à une plateforme e-santé selon les willayas



Taux des personnes préférant les rendez-vous en ligne



Taux des personnes qui ont des difficultés de trouver des praticiens



Taux des praticiens qui ne sont intéresse pour crée une plateforme e-santé



Taux des praticiens qui ont des difficultés de gère leur rendez-vous

Les besoins :

On parte des sondages, on a tiré quelques besoins très importants pour les deux côtes.

Pour les patients :

* Trouver et localiser un praticien.
* Prendre un rendez-vous.
* Accéder à son dossier médical.
* Recevoir des notifications à l’approche d’un rendez-vous chez son médecin.

Pour les praticiens :

* Gérer les rendez-vous.
* Edition automatique des ordonnances.
* Accéder au dossier d’un patient.
* Partager les dossiers avec d’autres praticiens.

1. **Cahier des charges** 
   1. **Présentation de l’organisation**

Le nom de l’organisation  : TECH-INSTINCT

La date de la création  : juillet 2018

Activité  : cabinet de conseil en informatique

Adresse : Forum de l’université 06000 Bejaia Algérie

Contacts  : +213 551 483 004

[contact@tech-instinct.com](mailto:contact@tech-instinct.com)

* 1. **Objectif de la plateforme**

L’objectif de cette plateforme est d’éliminer les problèmes liés à la prise des rendez-vous, d’améliorer l’information à destination du praticien notamment pour les sujets ayant des maladies chroniques et automatiser la gestion des rendez-vous pour les patriciens. Enfin, cette plateforme doit permettre de créer une relation de confiance entre le médecin et son patient.

* 1. **La société ciblée**

Cette plateforme vise le secteur médical et les membres de la société algérienne.

* 1. **Exemple de plateforme e-santé** :

**Doctolib et dzdoc** : sont des plateformes e-santé qui permettes d’améliorer la Gestion opérationnelle ainsi que le quotidien des praticiens de santé et des patients.

* 1. **Spécification des besoins**
     1. **Les besoins fonctionnels**

Nous détaillerons les fonctionnalités que le système doit fournir aux différents acteurs :

**Patient :**

**Profil :** contient toutes les informations personnelles du patient auxquelles il a accès pour modifier.

* **Localisation et avoir des informations :** le patient a la possibilité de localiser et avoir des informations d’une cabinet médical ou un hôpital par une simple recherche sur l’application.
* **Rendez-vous :** prendre un rendez-vous en ligne pour une visite médical.
* **Rappelle :** recevoir des rappels au fur et à mesure que le rendez-vous se rapproche.
* **Dossier médical :** le patient entre et consulte son dossier médical.

**Côte praticiens :**

* **Profil :** contient toutes les informations personnelles de médecin.
* **Dossier patient :** le médecin peut consulter et archiver les dossiers médicaux de ses patients uniquement et de les partager avec d’autre médecins si c’est nécessaire.
* **Définir les horaires de travail :** mentionné les horaires de travail.
* **Edition des ordonnances :** édition automatique des ordonnances.
* **Gérer l’agenda :** le médecin peut gérer ses rendez-vous et ses activités.
  + 1. **Les besoins non fonctionnels**

Les besoins non fonctionnels sont des indicateurs de qualité de l’exécution des besoins fonctionnels, ils permettent d’éviter plusieurs incohérences dans le système.

**La sécurité :**

* Protéger l’accès à la base de données en établissant des contraintes de contrôle qui va empêcher toutes personnes d’y accéder sauf ceux qui ont les droits.
* Chiffrer quelques données avant leurs insertions.
* Chiffrer les dossiers médicaux des patients.
* Tous les praticiens doivent avoir un compte pour gérer leur rendez-vous
* Un patient ou le praticien, n’accède qu’aux informations qui les concernent.
* La secrétaire ne peut pas accéder aux dossiers médicaux des patients
* L’administration de la plateforme est effectuée par une personne appropriée.

**La portabilité :**

* On doit créer un logiciel pour le praticien compatible sur Windows, mac-os et linux

**La fiabilité :**

* Lors d’une panne d’internet, le praticien peut utiliser quelques fonctionnalités de site/logiciel.
* Lors d’une panne dans le serveur les données ne seront pas perdu.

**La performance et efficacité :**

* Le temps de réponse de notre plateforme doit être précis et acceptable.
* La plateforme doit avoir un contrôleur des champs de saisie, pour éviter l’introduction des informations qui ne correspondent pas aux types des champs.

**Utilisabilité** :

* La plateforme doit être simple est facile à manipuler même par des non experts.

**L’ergonomie :**

* La plateforme doit s’inspirer des couleurs et adopter un logotype du secteur médical.
  1. **Graphisme et Ergonomie** 
     1. **Le nom de la plateforme**

Le nom de la plateforme est **TADAWSA**. Il est extrait du mot santé en langue amazigh qui signifie plusieurs chose « la force, la santé, la guérison »

* + 1. **La palette de couleur**

Figure 00 : représentation de palette de couleur utiliser dans notre plateforme

* + 1. **Logo**



Figure 00 : représentation des logos utiliser dans notre plateforme

* + 1. **Police du texte à utiliser**

Les polices de texte qui sont recommandées sont **HELVETICA** et **ARIAL SANS SERIF**

* 1. **Environnement et outils**
* **Le nombre de développeur :** 02 personnes
* **Le client :** le gérant de l’entreprise tech-instinct
* **L’entreprise**: TECH-INSTINCT
* **Université de Bejaia (UAMB)**
* **Les outils et langage :**

Visual paradigm, visual studio code, XAMPP, Server-web, git, GitLab

UML, HTML, CSS, JavaScript, Type Script, Bootstrap, Angular, ElectronJS, Angular Materiel, PostgreSQL, Spring Boot

* 1. **Planning**
* La durée estimée en total **: 6 mois**
* La durée estimée pour la création et validation des maquette **: 30 jours**
* La durée estimée pour la création et validation de contenu **: 30 jours**
* La durée estimée pour la création et validation du plateforme **: 120 jours**

# **Conclusion**

Durant cette section nous avons exprimé les objectifs attendus du futur système, ainsi les besoins auxquels doit répondre. Nous avons établi aussi une étude des systèmes existants qui nous permettent de déduire des fonctionnalités primordiales. Et Cela nous a permis d’avoir une idée sur la manière de la création d’une plateforme e-santé.

Dans la section suivante, nous allons étudie les outils de développement et l’environnement de travail.

# Généralité et environnement de développement

1. **Introduction**

Dans cette section, nous allons définir quelques généralités portant sur le développement web. Ensuite, nous allons présenter les langages, les environnements de développements et les SGBD ainsi que les approches de gestions de notre projet.

1. **Définitions et généralités.**
   1. **Application web**

Une application web est une application manipulable directement en ligne grâce à un navigateur web et qui ne nécessite pas d'installation sur les machines clientes, elle est généralement installée sur un serveur et se manipule en actionnant des widgets à l'aide d'un navigateur web, via un réseau informatique, en cite comme titre d’exemple La messagerie web, les systèmes de gestion de contenu, les wikis et les blogs.

* 1. **Logiciel**

En informatique, un logiciel est un ensemble de programmes interprétables par une machine pour lui permettre d’assurer une tâche ou un lot de fonctionnalités supplémentaires. Il existe plusieurs types de logiciel (logiciel client, micrologiciel, progiciel, logiciel de gestion, etc.).

1. **Les technologies et langages de développement**

Dans un esprit de défi et de vouloir mettre en œuvre les connaissances qu’on a acquises durant notre formation, nous avons choisi un ensemble de langage et des Framework qui sont très utilisés, notamment par un grand nombre de programmeurs professionnels ce qui en fait des langages de haut niveau.

* 1. **HTML**

HTML (Le HyperText Markup Language) est un langage de balisage, il nous permet de structurer et de mettre en forme le contenu des pages web, d’inclure des ressources multimédias exemple les formulaires de saisie, les images et les programmes informatiques. Il est souvent associé à des langages de programmation tels que le JavaScript, et des formats de présentation à savoir les feuilles de style en cascade.

* 1. **CSS**

Le CSS (Cascading Style Sheets) est un langage informatique coté client utilisé pour mettre en forme les fichiers HTML ou XML. Il permet de séparer la présentation d’une page html et sa structure

* 1. **SASS**

SASS est un langage de script préprocesseur compilé en CSS à l’aide d’une commande SASS. Il nous permet d’utiliser des variables, des règles imbriquées, des fonctions **etc.**

SASS il nous aide à garder les feuilles de style volumineuses bien organisées. Il est disponible en deux syntaxes :

* La syntaxe originale qui utilise extension **.sass**
* La nouvelle syntaxe, « SCSS », qui utilise l’extension **.scss**
  1. **Bootstrap**

Bootstrap est un Framework CSS populaire pour développer des platforms Web réactif. Il fournit des outils très puissants qui nous aide à développer très rapidement. On peut citer quelques composants comme les boutons, les formulaires, offcanvas. Etc.

* 1. **JavaScript**

JavaScript est un langage de script qui permet d’implémenter des programmes complexes sur les pages web, ces scripts peuvent être Écrits directement sur une page web HTML et exécutés automatiquement quand la page Se charge.

* 1. **Type Script**

Type script est un langage de programmation développé par Microsoft, il nous permet de spécifier les types de données transmises dans le code. Et de signaler des erreurs lorsque les types ne correspondent pas au moment de compilation de code, à la fin le code sera transpilé en JavaScript à l’aide d’un compilateur**.**

* 1. **ANGULAR et Angular matériel** 
     1. **Angular**

Angular est un Framework moderne open source basée sur le Type script développé par Google. Permet de crée des applications évolutives à page unique SPA (single page application). Il est basé sur l’architecture MVC qui permet de séparer les données.

* + 1. **Angular matériel**

Angular Matériel est une bibliothèque d’interface utilisateur (UI), qu’on peut utiliser dans les projets Angular pour accélérer le développement. Il nous offre des composants d’interface utilisateur magnifiques tels que des cartes, des entrées, des tableaux de données, des sélecteurs de date et bien plus encore.

* + 1. **Notre choix de Angular par rapport à REACT :**
* Angular est utilisé avec Type Script.
* Les services de langage Angular se complètent automatiquement à l’intérieur des fichiers HTML externes du composant.
* Documentation détaillée.
* Structure et architecture spécialement créées pour une meilleure scalabilité du projet.
* Étant le plus ancien Framework disponible, Angular bénéficie d’un large soutien de la communauté.

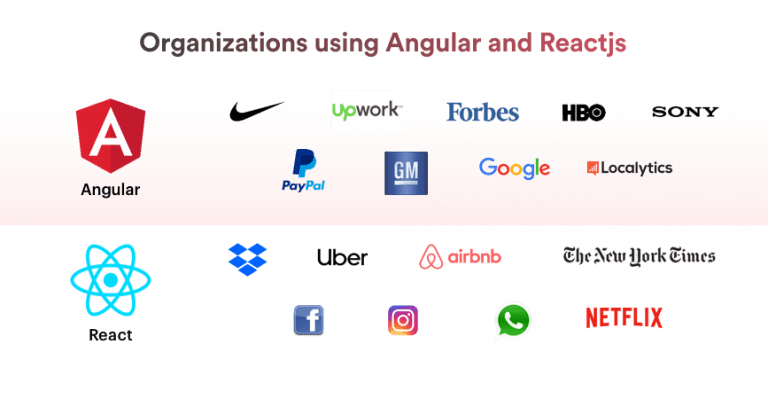


Figure 00 : figure représente les organisations qui utilisant les Framework Angular et React

* 1. **ELECTRONJS**

ElectronJs est un Framework qui permet de créer des applications desktop (logiciel) multi-platform (Win, Mac, Linux) en utilisant des technologies web telles que HTML5, CSS et JavaScript. De plus, il fournit plusieurs fonctionnalités de base comme la mise à jour automatique et des fonctionnalités spécifiques au système.

* 1. **Framework Ionic**

Ionic est un Framework open-source créé en 2013 permet de créer un code multi-support en utilisant des outils Web comme HTML, CSS, JavaScript, afin de générer des applications iOS, Android, Chrome, Windows Phone et bien d’autres.

* 1. **Spring boot**

Spring Boot est un Framework Java, utilisé pour développer des applications Web. Il nous permet de créer des API REST avec des configurations minimales. Spring Boot fournit un code passe-partout avec toutes les configurations nécessaires pour commencer immédiatement le codage.

1. **Les technologies et langages de modélisation**

Nous avons choisi UML pour bien vouloir mettre en œuvre les connaissances qu’on a acquises durant notre formation. De plus, il est très utilisé, notamment par un grand nombre de concepteur par rapport ou autre langage de modélisation.

* 1. **UML**

L’UML (Unifined Modeling Language) est un langage de modélisation graphique et textuelle, destiné à comprendre et décrire des besoins, spécifier et documenter des systèmes et concevoir des solutions.

1. **Les systèmes de gestion de bases de données**
   1. **PostgreSQL**

PostgreSQL est un système de gestion de base de données relationnelle objet (SGBDRO) open source. Il est concurrent a d'autres systèmes de gestion de base de données, qu'ils soient libres (MySQL), ou propriétaires (Oracle). PostgreSQL peut stocker et créer d’autres types de données plus que les types traditionnels, Il fonctionne sous différents systèmes d'exploitation. De plus, il est largement reconnu pour son comportement stable et pour ses possibilités de programmation étendues.

1. **Les environnements de développement**

Nous avons choisi ces logiciel grâce au service qu’ils fournissant, pour accélérer le développement de notre plateforme

* 1. **BgAdmin**

pgAdmin est une outil graphique multi-plateformes entièrement gratuit, fait partie des logiciels de gestion de bases de données Open Source pour l’administration des bases PostgreSQL. De plus, Il peut être exécuté en mode serveur web (web application) ou en mode bureau (desktop runtime).

* 1. **Visual Studio code**

Visual Studio Code est un éditeur de code open source léger et puissant, qui possède un riche écosystème d’extensions. Il est livré avec un support intégré pour JavaScript, Type Script et Node.js.

* 1. **Visual Paradigm**

Visual Paradigm est un logiciel qui permet aux programmeurs de mettre En place des diagrammes. Il possède plusieurs options permettant une large possibilité de modélisation en UML.

1. **Démarche pattern de développement et approche de gestion de projets**
   1. **Approche de gestion de projet**

La méthode agile Scrum est une approche qui met l’accent sur des méthode de travail flexibles, un developpement incrémentale et des procédures transparentes, elle est actuellement la pratique la plus adapter dans de nombreuses entreprises. Scrum établit un cadre de travail comporte des **rôles**, des **évènements** et des **artefacts**.

**Rôles Scrum :**

* **L’équipe de developpement :** est un ensemble de développeurs auto-organiser pour accomplir un objectif sans être diriger par une personne externe (pas d’hiérarchie).
* **Le Product owner :** est le responsable de qualité du produit, il rédige la liste des exigences de produit et déterminer les objectifs finaux de futur système.
* **Scrum Master** : a comme responsabilité de simplifie le processus de travail pour les membres de l’équipe (Equipede developpement, Product owner),et augmenter leur productivité et créativité par la planification des réunions est l’assurance de la modération.

**Événement Scrum :**

* **Sprint** : décrit la période d’une phase de développement. A la fin d’un cycle l’équipe de developpement doit avoir livrer un produit potentiellement livrable.
* **Sprint planning** (**planification de sprint**) : est réalisé par l’équipe de developpement en appuyant sur le backloge, il sert à déterminer les tâches à réaliser et la façon de les faire dans des délais précises.
* **Daily Scrum (mêlée quotidienne)** : doit avoir lieu chaque jour d’un sprint pour discuter des tâches à réaliser dans la journée et de ce qui a été effectue la veille, dans le but de voir l’avancée de projet.
* **Sprint review (revue de sprint)** : est effectuée à la fin de chaque sprint, il consiste à vérifier le produit livrer par des personnes ayant un intérêt considérable au produit.
* **Sprint retrospective (la rétrospective de sprint) :** tout l’équipe Scrum réounent pour discuter des problèmes rencontre et d’avancée de projet. L’objectif est de résoudre les problèmes internes et d’améliorer le travail collectif.

**Artefacts Scrum :**

* **Backlog produit :** est une liste ordonnée de tous les éléments de produit (fonctionnalités, améliorations).
* **Backlog sprint :** est générer à partir de Backlog produit, il représente toutes les étapes de réalisation d’un objectif de sprint.
* **L’incrément achevé :** le résultat de la phase de developpement.
  1. **L’architecteur MVC**

MVC est un modèle architectural qui sépare une application en trois composants logiques principaux : modèle, vue et le contrôleur vise à organisées le code source. De plus, plusieurs Framework reposent sur cette architecture exemple Angular et Ionic qu’on va utiliser dans la réalisation de notre plate-forme

1. **Conclusion**

Dans cette section, nous avons réussi à définir les différents concepts informatiques qui nous ont semblé essentiels à la réalisation de notre projet.

Dans le chapitre qui se suit nous allons détailler la conception de notre travail.