1. Technologies et Compatibilité

Technologies utilisées pour le Site Web

Front-end

|  |  |
| --- | --- |
| **Angular** |  |
| Framework robuste pour les applications complexes, avec une bonne structure et une meilleure maintenabilité à long terme. | |

|  |  |
| --- | --- |
| **TypeScript** |  |
| TypeScript est un superset de JavaScript développé par Microsoft, qui ajoute des types statiques et améliore la sécurité et la maintenabilité du code. Il est particulièrement utile pour les applications complexes, offrant une meilleure gestion des erreurs, un outillage performant et une compatibilité totale avec JavaScript. | |

|  |  |
| --- | --- |
| **HTML5** |  |
| HTML5 est la dernière version du langage HTML, utilisé pour structurer et organiser le contenu d’une page web. Il apporte de nouvelles balises sémantiques, des API multimédia et une meilleure gestion des performances et de l’accessibilité. | |

|  |  |
| --- | --- |
| **Tailwind CSS** |  |
| Tailwind CSS est un framework CSS utilitaire qui permet de styliser rapidement une interface sans écrire de fichiers CSS personnalisés. Contrairement à Bootstrap, il ne propose pas de composants prédéfinis, mais une approche modulaire et ultra-flexible pour un design unique et personnalisé. | |

Back-end

|  |  |
| --- | --- |
| **Node.js** |  |
| Node.js est un environnement bas niveau qui nous permet d'utiliser le langage JavaScript sur le serveur. Il nous permet donc de faire du JavaScript en dehors du navigateur !  Node.js bénéficie de la puissance de JavaScript et par extension du typescript pour proposer une toute nouvelle façon de développer des sites web dynamiques. | |

|  |  |
| --- | --- |
| **Express.js** |  |
| Express.js est un framework web minimaliste et rapide pour Node.js, permettant de créer des API et applications web avec une gestion simplifiée des routes, des requêtes et des middlewares, tout en offrant flexibilité et performances pour les applications modernes | |

Base de données

|  |  |
| --- | --- |
| **MongoDB** |  |
| Mongo DB est base NoSQL flexible, parfaite pour stocker des données utilisateurs et historiques des jeux-concours. | |

API & Communication

|  |  |
| --- | --- |
| **GraphQL** |  |
| Un langage de requête pour les API permettant aux clients de récupérer uniquement les données nécessaires via une seule requête, offrant ainsi plus de flexibilité et d’optimisation par rapport aux APIs traditionnelles. | |

|  |  |
| --- | --- |
| **REST API** |  |
| Une architecture d’API standardisée basée sur des endpoints définis utilisant les méthodes HTTP (GET, POST, PUT, DELETE) pour échanger des données entre un client et un serveur de manière simple et scalable. | |

CMS (Gestion de contenu)

|  |  |
| --- | --- |
| **Strapi** |  |
| CMS headless facile à intégrer avec Angular, idéal pour gérer le contenu. | |

Stockage des médias

|  |  |
| --- | --- |
| **AWS S3 (Amazon Simple Storage Service)** |  |
| Un service de stockage cloud scalable permettant de stocker, sécuriser et récupérer des fichiers (images, vidéos, documents, etc.) avec une haute disponibilité et une gestion avancée des permissions et de la réplication des données. | |

Authentification

|  |  |
| --- | --- |
| **OAuth 2.0 (Google, Facebook)** |  |
| Un service de stockage cloud scalable permettant de stocker, sécuriser et récupérer des fichiers (images, vidéos, documents, etc.) avec une haute disponibilité et une gestion avancée des permissions et de la réplication des données. | |

|  |  |
| --- | --- |
| **JWT (JSON Web Token)** |  |
| JWT est un standard de sécurisation des échanges de données entre un client et un serveur sous forme de jeton signé, permettant une authentification sans état, sécurisée et performante pour les applications web et mobiles. | |

Sécurité

|  |  |
| --- | --- |
| **HTTPS** |  |
| HTTPS (HyperText Transfer Protocol Secure) est un protocole sécurisé de communication sur le web, qui chiffre les échanges de données entre un client et un serveur à l'aide de TLS/SSL, garantissant ainsi confidentialité, intégrité et authentification des informations. | |

|  |  |
| --- | --- |
| **Content Security Policy** |  |
| Content Security Policy (CSP) est un mécanisme de sécurité web qui permet de prévenir les attaques XSS et l'injection de contenu malveillant en définissant des règles strictes sur les sources autorisées de scripts, styles, images et autres ressources dans une application web. | |

|  |  |
| --- | --- |
| **OWASP Best Practices** |  |
| OWASP Best Practices désigne un ensemble de recommandations de sécurité définies par l'OWASP (Open Web Application Security Project) visant à protéger les applications web contre les vulnérabilités courantes telles que les injections SQL, XSS, CSRF, et les failles d'authentification, en appliquant des principes de codage sécurisé, de chiffrement des données et de gestion des accès. | |

**Compatibilité Navigateur (Site Web)**



Technologies utilisées pour la WebApp (Version Mobile)

Afin d’offrir une expérience utilisateur fluide et performante sur mobile, la WebApp sera développée avec les technologies qui suivent :

Framework Mobile

|  |  |
| --- | --- |
| **Flutter (Dart)** |  |
| Flutter (Dart) est un framework open-source développé par Google permettant de créer des applications mobiles, web et desktop avec une seule base de code. Il utilise le langage Dart, conçu pour être rapide et optimisé, et offre une interface fluide grâce à son moteur graphique performant (Skia). Flutter permet de développer des applications natives sur Android et iOS, avec une expérience utilisateur homogène et réactive. | |

UI/UX

|  |  |
| --- | --- |
| **Material Design (Flutter)** |  |
| Material Design (Flutter) est un système de design développé par Google qui fournit une bibliothèque complète de composants UI pour créer des interfaces modernes, cohérentes et responsives sur les applications Flutter. Il intègre des éléments tels que des boutons, cartes, barres de navigation et animations fluides, tout en respectant les principes d’ergonomie et d’accessibilité pour offrir une expérience utilisateur intuitive et esthétique sur mobile et web. | |

Gestion des Données

|  |  |
| --- | --- |
| **REST API** |  |
| Une architecture d’API standardisée basée sur des endpoints définis utilisant les méthodes HTTP (GET, POST, PUT, DELETE) pour échanger des données entre un client et un serveur de manière simple et scalable. | |

Gestion des Données

|  |  |
| --- | --- |
| **Secure Storage (Flutter)** |  |
| **Secure Storage (Flutter)** est un **plugin de stockage sécurisé** qui permet de **stocker des données sensibles** (tokens d'authentification, informations utilisateur, clés API…) sur un appareil mobile en utilisant **les solutions de chiffrement natives** de chaque système d’exploitation. | |

Notifications Push

|  |  |
| --- | --- |
| **Firebase Cloud Messaging (FCM)** |  |
| Firebase Cloud Messaging (FCM) est un service de notification push multiplateforme permettant d’envoyer des messages en temps réel aux utilisateurs sur le web, Android et iOS, favorisant ainsi l’engagement et l’interaction avec l’application. | |

Authentification

|  |  |
| --- | --- |
| **OAuth 2.0 (Google, Facebook)** |  |
| Un service de stockage cloud scalable permettant de stocker, sécuriser et récupérer des fichiers (images, vidéos, documents, etc.) avec une haute disponibilité et une gestion avancée des permissions et de la réplication des données. | |

|  |  |
| --- | --- |
| **JWT (JSON Web Token)** |  |
| JWT est un standard de sécurisation des échanges de données entre un client et un serveur sous forme de jeton signé, permettant une authentification sans état, sécurisée et performante pour les applications web et mobiles. | |

Performance & Monitoring

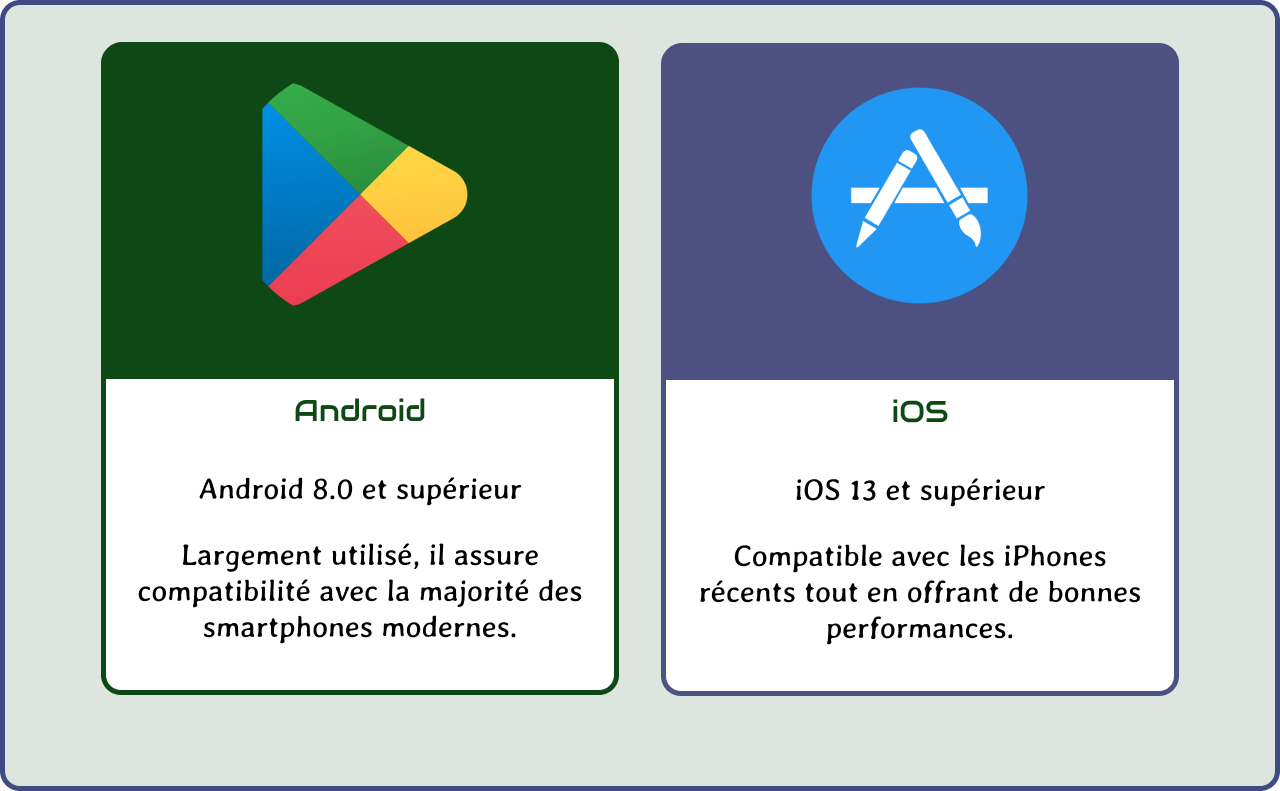
|  |  |
| --- | --- |
| **Firebase Performance Monitoring** |  |
| Il permet de suivre les performances et gestion des crashs pour assurer une expérience fluide. | |

Publication App Stores

|  |  |
| --- | --- |
| **Google Play Store** |  |
| Google Play Store est la plateforme officielle de distribution d'applications Android, permettant aux utilisateurs de télécharger, installer et mettre à jour des applications, jeux et contenus numériques en toute sécurité. | |

|  |  |
| --- | --- |
| **Apple App Store** |  |
| Apple App Store est la plateforme officielle de distribution d'applications pour les appareils iOS (iPhone, iPad, Mac), permettant aux développeurs de publier, monétiser et mettre à jour leurs applications tout en garantissant sécurité et qualité aux utilisateurs. | |

Compatibilité Mobile (WebApp)



1. Software – Logiciels et Outils Utilisés

Gestion de Projet

Des outils de gestion de projet sont utilisés pour **suivre les tâches, organiser les sprints et gérer les plannings**.

Outils Graphiques

Des logiciels de conception graphique et de prototypage sont utilisés pour le design du site, la charte graphique et l’expérience utilisateur.

Outils Techniques