Projet réalisé dans le cadre de la soutenance

devant le Jury Final Pour l’obtention du titre

**Concepteur Développeur d’Applications**

**SOMMAIRE**

Table des matières

[Résumé / Abstract 6](#__RefHeading___Toc1813_1098725337)

[1. Introduction 6](#__RefHeading___Toc1815_1098725337)

[2. Contexte du projet 6](#__RefHeading___Toc1817_1098725337)

[3. Cahier des charges 6](#__RefHeading___Toc1819_1098725337)

[3.1. Benchmark 6](#__RefHeading___Toc1821_1098725337)

[3.2. Persona 7](#__RefHeading___Toc1823_1098725337)

[3.3. Expression & analyse des besoins 7](#__RefHeading___Toc1825_1098725337)

[3.4. Exigences technico-fonctionnelles 8](#__RefHeading___Toc1827_1098725337)

[3.5. MVP (V. 1.0) 8](#__RefHeading___Toc1829_1098725337)

[3.6. Version 2.0 8](#__RefHeading___Toc1831_1098725337)

[3.7. Version 3.0 8](#__RefHeading___Toc1833_1098725337)

[4. Méthodologie de travail 8](#__RefHeading___Toc1835_1098725337)

[4.1. Cycle en Spirale 8](#__RefHeading___Toc1837_1098725337)

[4.2. Cycle en V 9](#__RefHeading___Toc1839_1098725337)

[4.3. Méthodologie Agile/Scrum 9](#__RefHeading___Toc1841_1098725337)

[4.4. Méthodologie Agile/Kanban 9](#__RefHeading___Toc1843_1098725337)

[4.5. Méthodologie Agile/Scrumban 9](#__RefHeading___Toc1845_1098725337)

[4.6. Méthodologie Agile/TDD 9](#__RefHeading___Toc1847_1098725337)

[4.7. Méthodologie Agile/FDD 9](#__RefHeading___Toc1849_1098725337)

[5. Spécifications fonctionnelles 9](#__RefHeading___Toc1851_1098725337)

[5.1. Fonctionnalité "S'authentifier" 10](#__RefHeading___Toc1853_1098725337)

[5.2. Fonctionnalité "S'inscrire" 10](#__RefHeading___Toc1855_1098725337)

[5.3. Fonctionnalité "Payer par CB" 10](#__RefHeading___Toc1857_1098725337)

[5.4. Fonctionnalité "Ajouter article au panier" 10](#__RefHeading___Toc1859_1098725337)

[6. Spécifications techniques 10](#__RefHeading___Toc1861_1098725337)

[6.1. L'architecture logicielle multicouches MVC 10](#__RefHeading___Toc1863_1098725337)

[6.2. L'architecture n-Tiers  (3-Tiers) 10](#__RefHeading___Toc1865_1098725337)

[6.3. Technologies choisies 10](#__RefHeading___Toc1867_1098725337)

[6.3.1. Technologies côté Front-end 10](#__RefHeading___Toc1869_1098725337)

[6.3.2. Technologies côté Back-end 10](#__RefHeading___Toc1871_1098725337)

[6.3.3. Technologies côté Persistance des données 10](#__RefHeading___Toc1873_1098725337)

[6.3.4. Technologies transverses 11](#__RefHeading___Toc1875_1098725337)

[7. Conception de l'interface graphique 11](#__RefHeading___Toc1877_1098725337)

[7.1. Benchmark 11](#__RefHeading___Toc1879_1098725337)

[7.2. Persona 11](#__RefHeading___Toc1881_1098725337)

[7.3. Expérience UX/UI 11](#__RefHeading___Toc1883_1098725337)

[7.4. Charte graphique 11](#__RefHeading___Toc1885_1098725337)

[7.5. Responsive 11](#__RefHeading___Toc1887_1098725337)

[7.6. Zoning 11](#__RefHeading___Toc1889_1098725337)

[7.7. Wireframes 11](#__RefHeading___Toc1891_1098725337)

[7.8. Maquettes 11](#__RefHeading___Toc1893_1098725337)

[7.9. Arborescence du site 11](#__RefHeading___Toc1895_1098725337)

[8. Conception de l'application 12](#__RefHeading___Toc1897_1098725337)

[8.1. Le langage de modélisation UML 12](#__RefHeading___Toc1899_1098725337)

[8.2. Diagramme de cas d'utilisation 12](#__RefHeading___Toc1901_1098725337)

[8.3. Diagramme de séquences 12](#__RefHeading___Toc1903_1098725337)

[8.4. Diagramme d’activité 12](#__RefHeading___Toc1905_1098725337)

[8.5. Diagramme de classes 12](#__RefHeading___Toc1907_1098725337)

[8.6. Diagramme des entités 12](#__RefHeading___Toc1909_1098725337)

[9. Conception de la Base de données 12](#__RefHeading___Toc1911_1098725337)

[9.1. La méthode MERISE 12](#__RefHeading___Toc1913_1098725337)

[9.2. Le Modèle Conceptuel de Données (MCD) 12](#__RefHeading___Toc1915_1098725337)

[9.3. Le Modèle Logique de Données (MLD) 12](#__RefHeading___Toc1917_1098725337)

[9.4. Le Modèle Physique de Données (MPD) 12](#__RefHeading___Toc1919_1098725337)

[9.5. Le Dictionnaire de Données (DD) 12](#__RefHeading___Toc1921_1098725337)

[10. Développement et Intégration 12](#__RefHeading___Toc1923_1098725337)

[10.1. Structure du projet 12](#__RefHeading___Toc1925_1098725337)

[10.2. Fonctionnalité « s'authentifier » 13](#__RefHeading___Toc1927_1098725337)

[10.3. Fonctionnalité « s'inscrire » 13](#__RefHeading___Toc1929_1098725337)

[10.4. Fonctionnalité « payer par CB » 13](#__RefHeading___Toc1931_1098725337)

[10.5. Fonctionnalité « ajouter un article » 13](#__RefHeading___Toc1933_1098725337)

[10.6. …etc 13](#__RefHeading___Toc1935_1098725337)

[11. Sécurité de l'application et des données 13](#__RefHeading___Toc1937_1098725337)

[11.1. Les différentes attaques de site 13](#__RefHeading___Toc1939_1098725337)

[11.1.1. Les attaques XSS 13](#__RefHeading___Toc1941_1098725337)

[11.1.2. Les attaques CSRF 13](#__RefHeading___Toc1943_1098725337)

[11.1.3. Les attaques CROS 13](#__RefHeading___Toc1945_1098725337)

[11.1.4. Les attaques Force Brute 13](#__RefHeading___Toc1947_1098725337)

[11.1.5. Les attaques de type injection SQL 13](#__RefHeading___Toc1949_1098725337)

[11.1.6. Authentification/Autorisation avec JWT 13](#__RefHeading___Toc1951_1098725337)

[11.1.7. Authentification Duo/Trio/Quad 13](#__RefHeading___Toc1953_1098725337)

[11.2. HTTPS 13](#__RefHeading___Toc1955_1098725337)

[11.3. La loi RGPD 13](#__RefHeading___Toc1957_1098725337)

[11.3.1. Confidentialité des données 13](#__RefHeading___Toc1959_1098725337)

[11.4. Mentions Légales 14](#__RefHeading___Toc1961_1098725337)

[11.4.1. Mentions Légales 14](#__RefHeading___Toc1963_1098725337)

[11.4.2. Conditions Générales d'utilisation du site web 14](#__RefHeading___Toc1965_1098725337)

[11.4.3. Conditions Générales de vente (sites de commerce ou de service) 14](#__RefHeading___Toc1967_1098725337)

[11.4.4. Politique de confidentialité 14](#__RefHeading___Toc1969_1098725337)

[11.4.5. Cookies 14](#__RefHeading___Toc1971_1098725337)

[11.4.6. Droit de rétractation 14](#__RefHeading___Toc1973_1098725337)

[12. Sécurité de la base de données 14](#__RefHeading___Toc1975_1098725337)

[12.1. Choix des noms de user et des password 14](#__RefHeading___Toc1977_1098725337)

[12.2. Limiter les droits au user 14](#__RefHeading___Toc1979_1098725337)

[12.3. Limiter l'accès à la base de données 14](#__RefHeading___Toc1981_1098725337)

[13. Sécurité et qualité du code 14](#__RefHeading___Toc1983_1098725337)

[13.1. Qualité du code 14](#__RefHeading___Toc1985_1098725337)

[13.1.1. Convention de nommage 14](#__RefHeading___Toc1987_1098725337)

[13.1.2. Conventions de codage 14](#__RefHeading___Toc1989_1098725337)

[13.1.3. Journalisation (LOG) 14](#__RefHeading___Toc1991_1098725337)

[13.1.4. Commentaires sur les lignes de code 14](#__RefHeading___Toc1993_1098725337)

[13.2. Sécurité du code 15](#__RefHeading___Toc1995_1098725337)

[13.2.1. Tests unitaires et couverture de code 15](#__RefHeading___Toc1997_1098725337)

[13.2.2. Gestion des erreurs et des exceptions 15](#__RefHeading___Toc1999_1098725337)

[13.2.3. Fermeture correcte des flux de données 15](#__RefHeading___Toc2001_1098725337)

[14. Déploiement et hébergement 15](#__RefHeading___Toc2003_1098725337)

[15. Référencement SEO 15](#__RefHeading___Toc2005_1098725337)

[16. Veilles technologiques 15](#__RefHeading___Toc2007_1098725337)

[16.1. Veille globale 15](#__RefHeading___Toc2009_1098725337)

[16.2. Veille concernant le développement 15](#__RefHeading___Toc2011_1098725337)

[16.3. Veille sur la sécurité des applications 15](#__RefHeading___Toc2013_1098725337)

[16.3.1. Les différentes failles de sécurité les plus dangereuses 15](#__RefHeading___Toc2015_1098725337)

[16.3.2. Les recommandations de l'ANSSI (https://cyber.gouv.fr/) 16](#__RefHeading___Toc2017_1098725337)

[16.3.3. La CNIL et ses recommandations 16](#__RefHeading___Toc2019_1098725337)

[16.4. Veille sur la sécurité des bases de données 16](#__RefHeading___Toc2021_1098725337)

[17. Perspectives 16](#__RefHeading___Toc2023_1098725337)

[18. Conclusion 16](#__RefHeading___Toc2025_1098725337)

[Annexe 17](#__RefHeading___Toc2027_1098725337)

# **Résumé / Abstract**

# **Introduction**

Je m’appel Jester CESAR, j’ai 24 ans.

# **Contexte du projet**

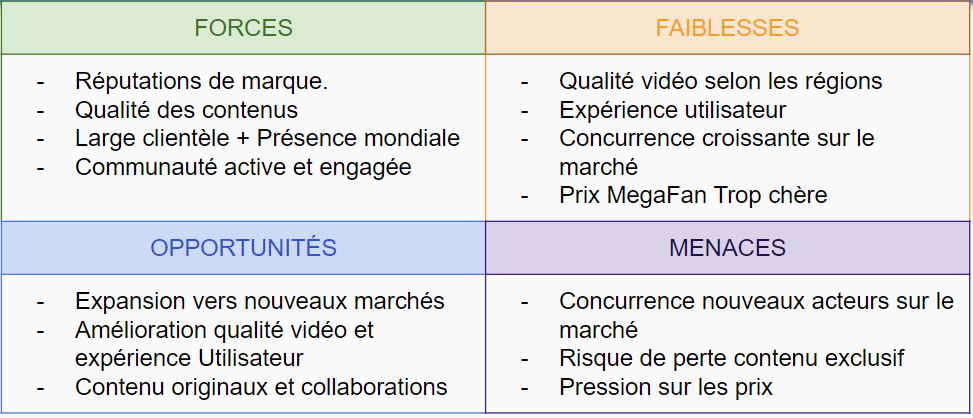
(problématique, constat ...)

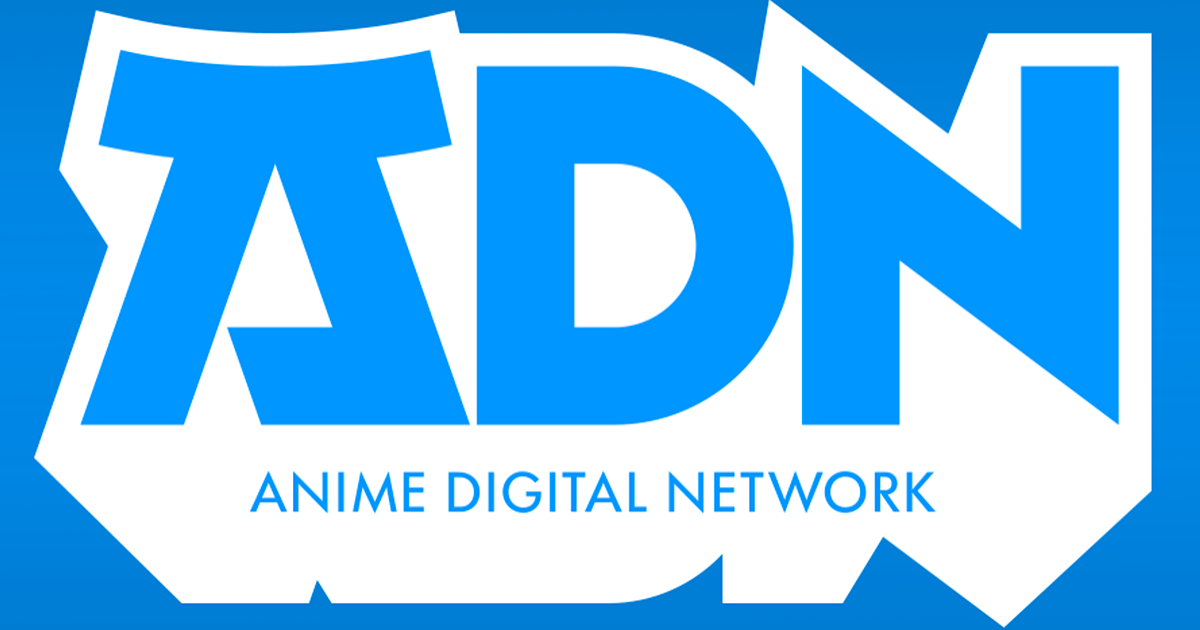
# **Cahier des charges**

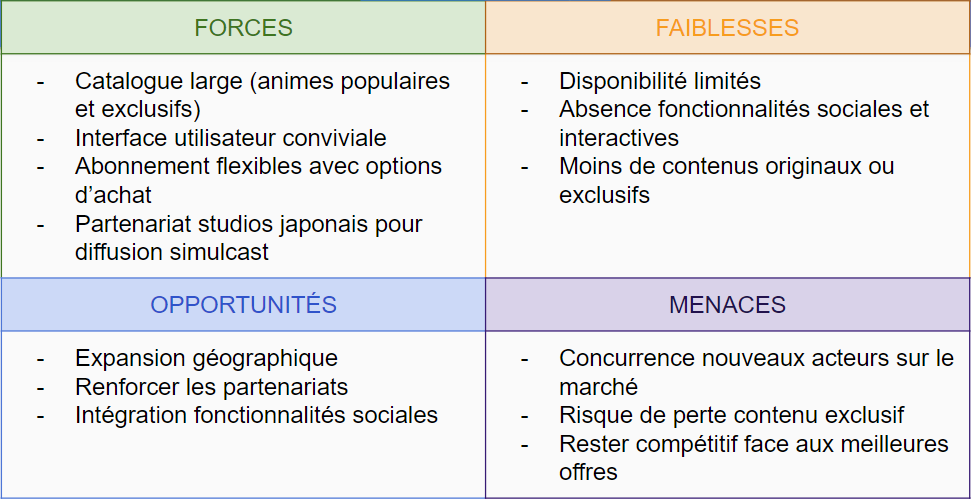
## Benchmark

Définition : Benchmark aussi appelé Benchmarking, est un terme anglais qui peut être traduit par les mots « référence » ou « repère ». Le benchmark est une technique marketing basé sur l’analyse comparative. Elle vise à étudier d’autres entreprises pour optimiser ses propres techniques de gestion et modes d’organisation.

Voici 3 exemples de Benchmark :







## Persona

## Expression & analyse des besoins

Liste ou tableau des fonctionnalités par profil

## Exigences technico-fonctionnelles

Navigateurs, Sécurité des données (Respect de la loi RGPD, Respect des recommandations de la CNIL, Mentions légales, CGU, Cookies, Droit de rétractation), se prémunir et se protéger des différentes attaques de site.

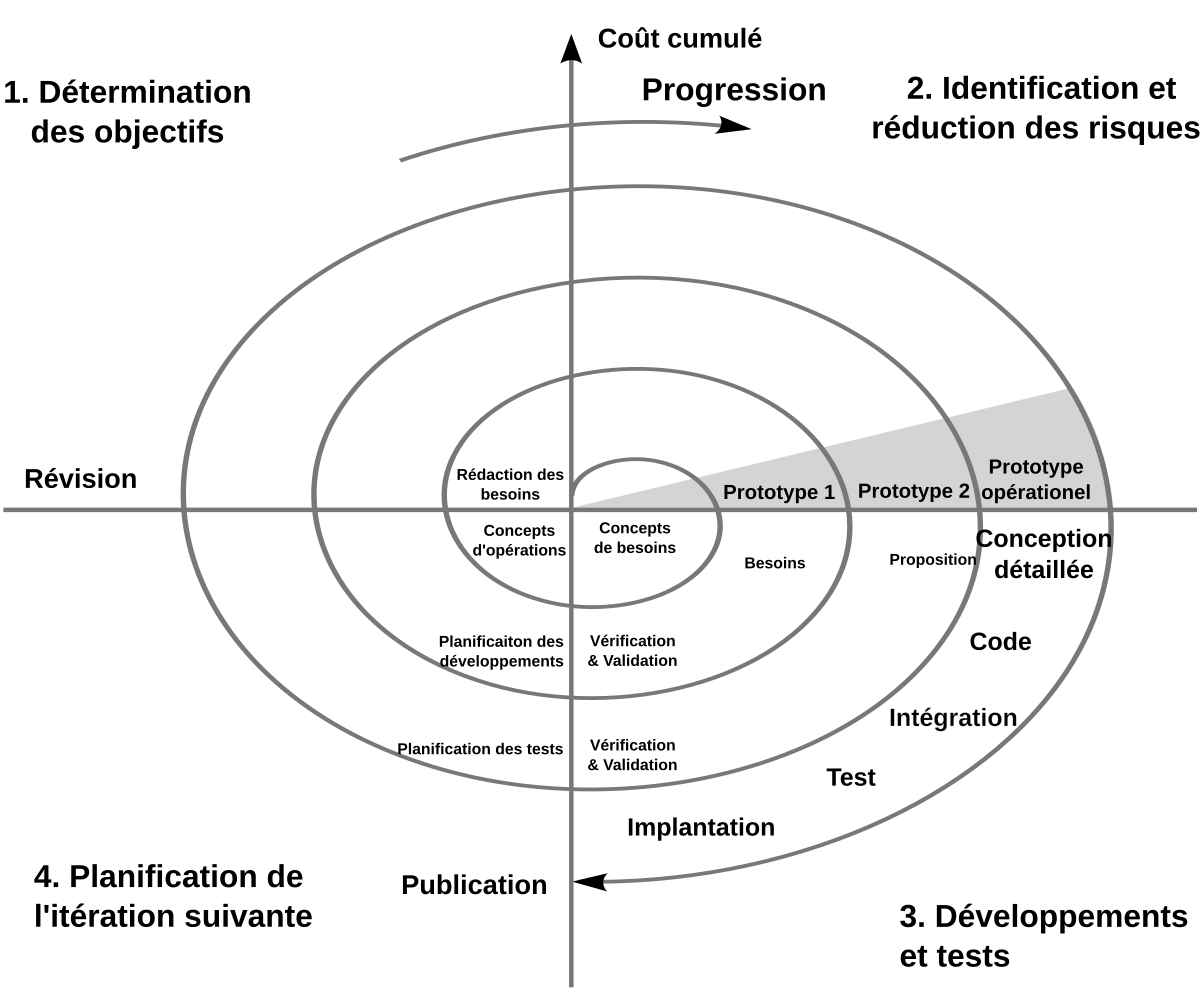
## MVP (V. 1.0)

## Version 2.0

## Version 3.0

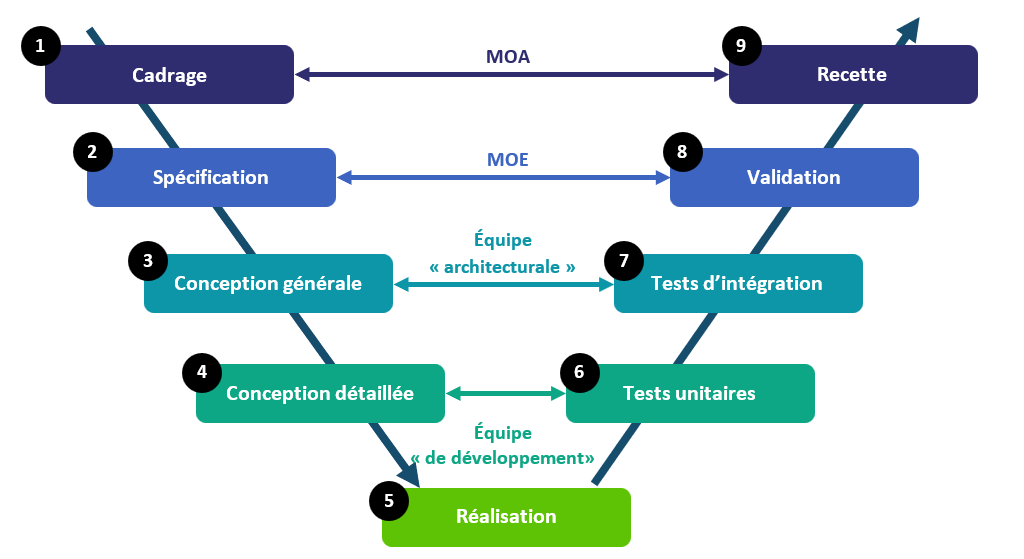
# **Méthodologie de travail**

## Cycle en Spirale



Définition :

## Cycle en V



(parler du diagramme de Gantt, et des différentes phases d'un projet en cycle en V)

## Méthodologie Agile/Scrum

(Définir les sprints, ...)

## Méthodologie Agile/Kanban

(Pour la gestion et planification des tâches => pour cela, utilisez l'outil TRELLO)

## Méthodologie Agile/Scrumban

## Méthodologie Agile/TDD

(Pour la mise en place de tests unitaires avant le développement d’une méthode)

## Méthodologie Agile/FDD

(Pour la réalisation d’une fonctionnalité de bout en bout càd de l’interface graphique jusqu’à la base de données)

# **Spécifications fonctionnelles**

(expliquer les fonctionnalités une à une et par profil, en montrant pour trois ou quatre la **description textuelle**)

## Fonctionnalité "S'authentifier"

## Fonctionnalité "S'inscrire"

## Fonctionnalité "Payer par CB"

## Fonctionnalité "Ajouter article au panier"

# **Spécifications techniques**

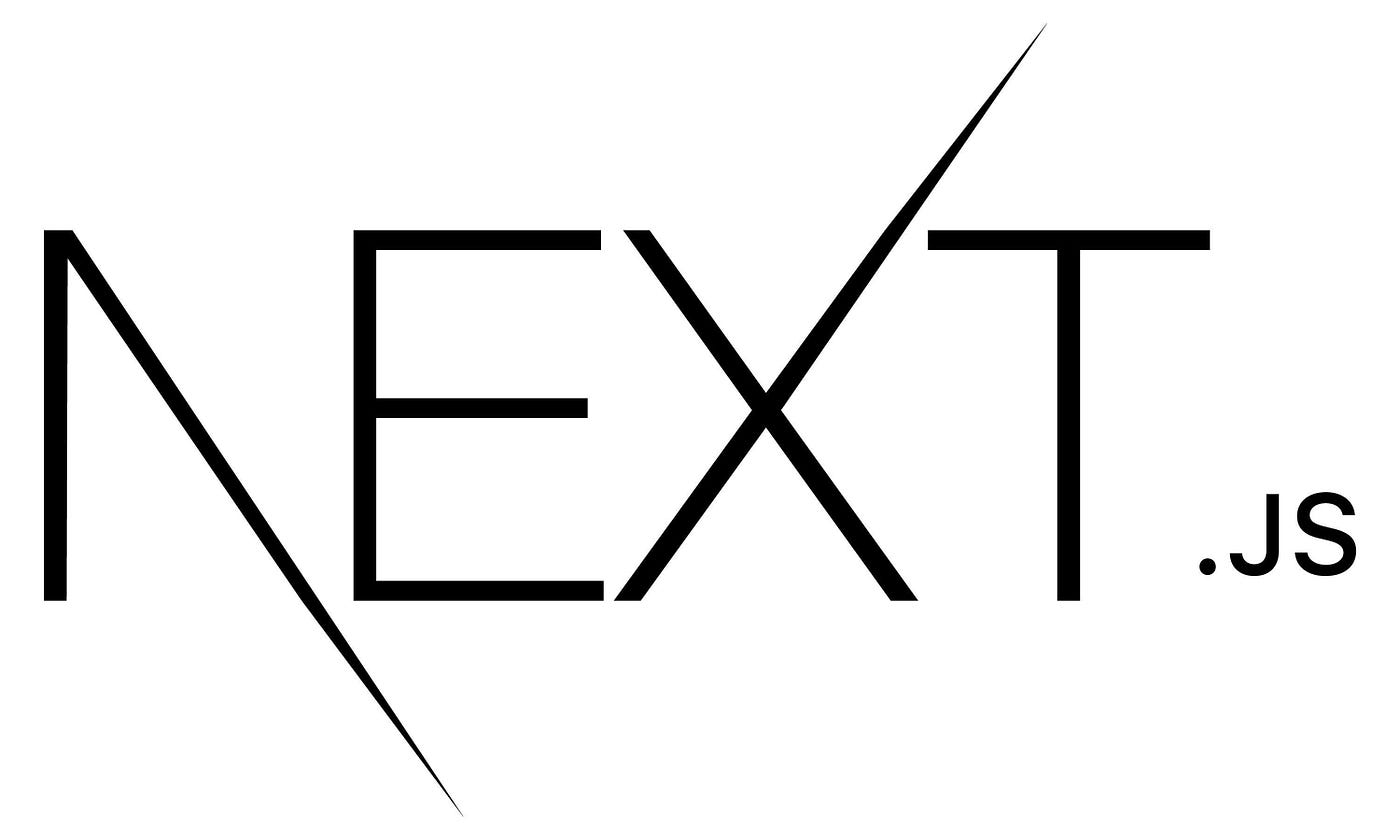
## L'architecture logicielle multicouches MVC

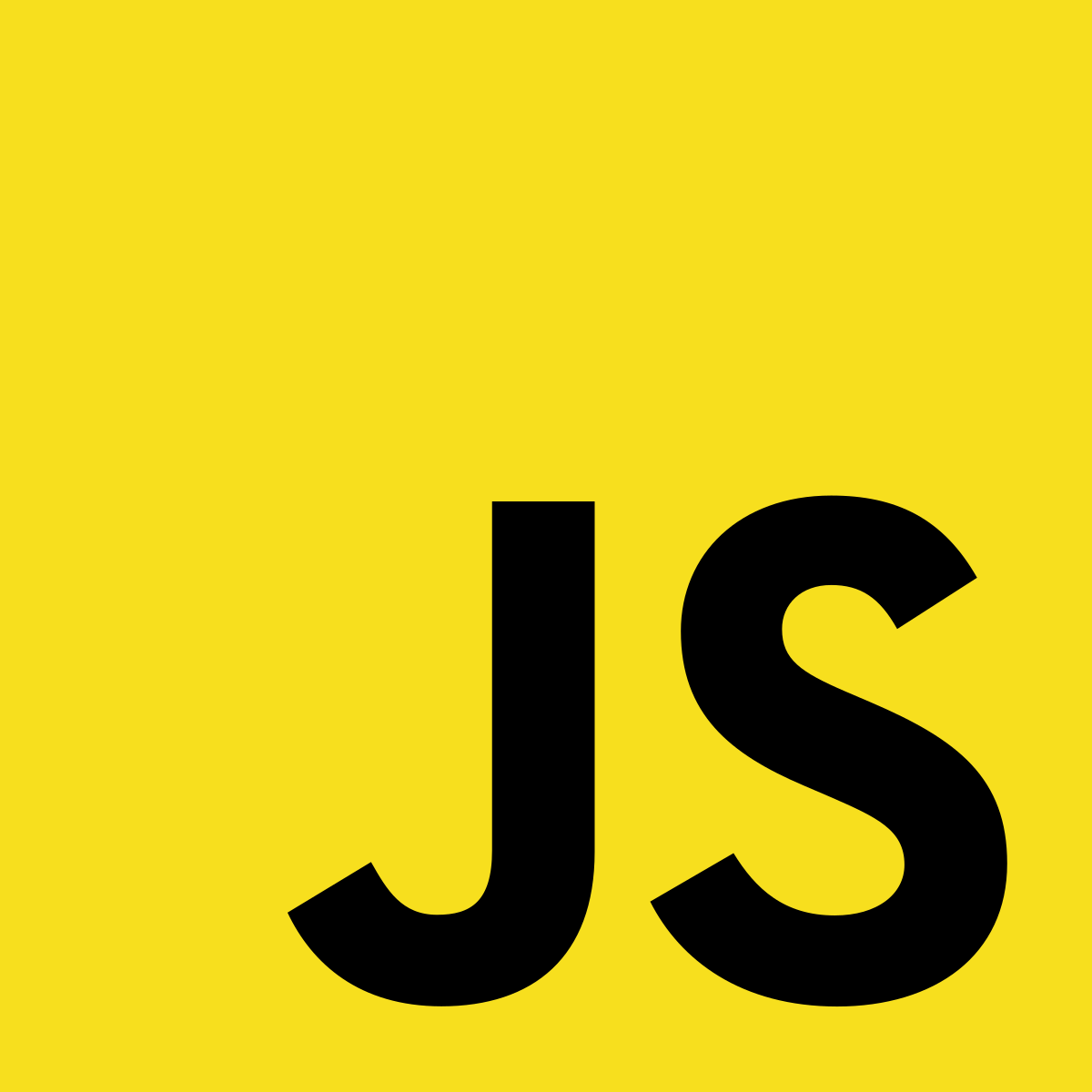
## L'architecture n-Tiers (3-Tiers)

## Technologies choisies

### Technologies côté Front-end

* **Langages et Framework :**

*Next Js :* c’est un framework gratuit et open source s’appuyant sur la bibliothèque JavaScript React et sur la technologie Node Js. J’ai choisit ce framework car il prend en charge le côté serveur (Server Side Rendering), le côté statique d’une page web (Static Site Generation) et l’application web est dynamique.



*JavaScript :* c’est un langage de programmation de scripts employé dans les pages web. Avec HTML et CSS, JavaScript fait partie des langages les plus utilisés par les développeurs. J’ai choisit ce langage pour ajouter de l’interactivité et des fonctionnalités de l’expérience utilisateur

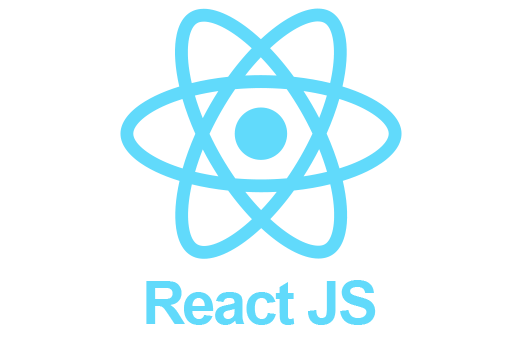


*HTML :* c’est un langage de balisage conçu pour représenter les pages web. Ce langage permet d’écrire de l’hypertexte, de créer des formulaires, et des programmes informatiques. Il est souvent utilisé conjointement avec JavaScript et CSS. J’ai choisit ce langage car c’est le langage le plus connu pour créer une page web.



*CSS :* c’est un langage informatique de type feuilles de style en cascade qui permet de mettre en forme des documents de types HTML et XML. J’ai choisit ce langage car il est toujours utilisé avec HTML.

* **Outils :**

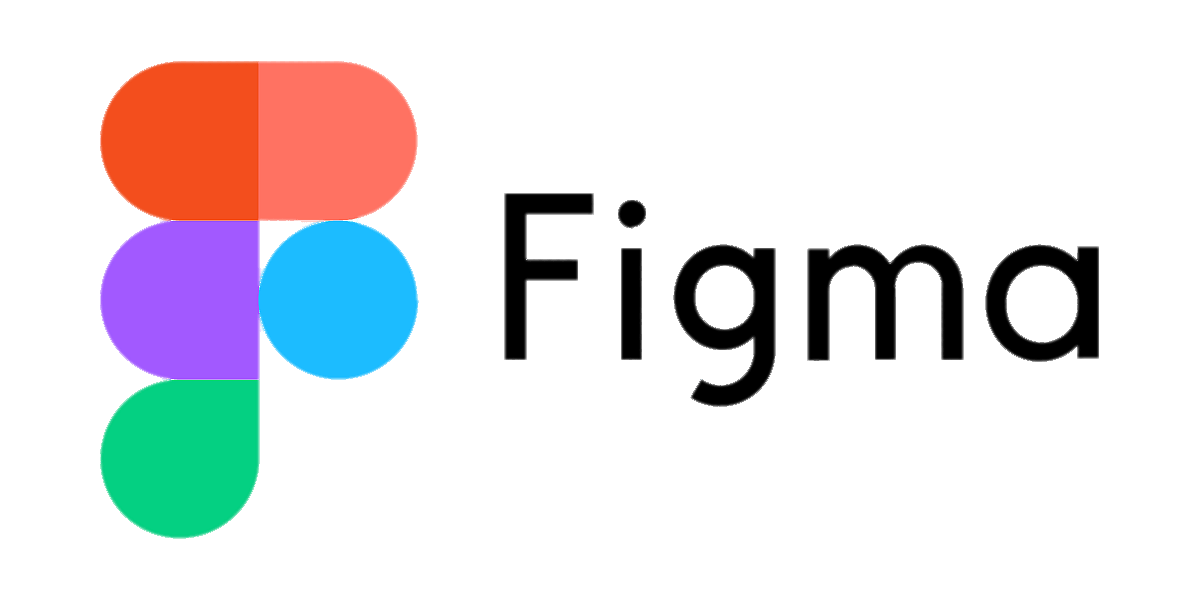
****

*React Js :* c’est une librairie JavaScript pour créer des interfaces utilisateurs qui peut être utilisée avec un framework MVC comme Next Js. J’ai choisit cette bibliothèque car c’est plus simple à écrire et permet d’utiliser des composants qu’on peut réutiliser.

*Tailwind CSS :* c’est une bibliothèque CSS open source qui facilite le développeur. J’ai choisit cette librairie car contrairement a Bootstrap, il n’y a pas de classes prédéfinis comme pour les boutons ou les tables. C’est au développeur de choisir ces propres classes et les modifies a sa manière comme sur les feuilles de styles de CSS.

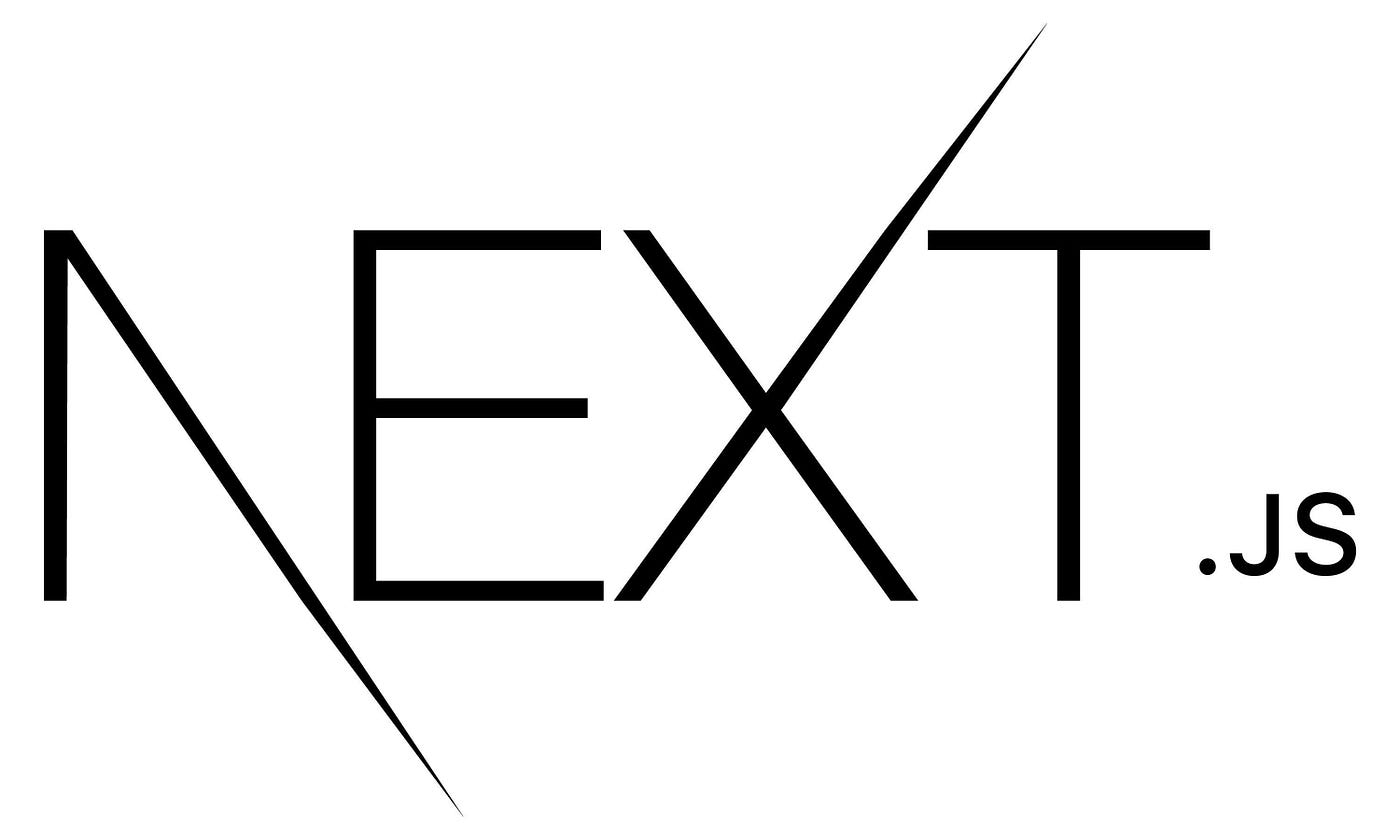


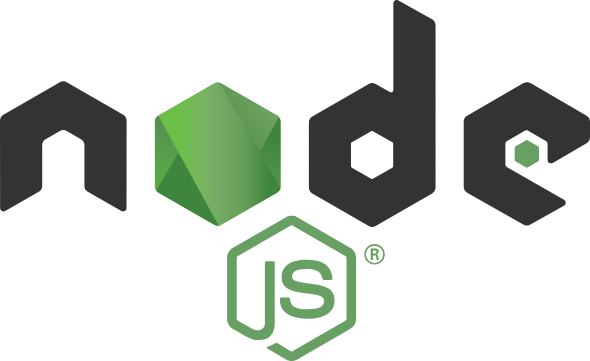
*Canva :* c’est un outil de conception graphique en ligne. J’ai choisit cette outil pour créer un logo et pour redimensionner mes images afin de les mettre a la même taille. C’est un outil très facile a utiliser fait pour tout type de personne.

*Figma :* c’est un éditeur de graphiques vectoriels et un outil de prototypage. Ses fonctionnalités sont axés sur l’utilisation dans la conception de l’interface utilisateur et de l'expérience utilisateur, en mettant l'accent sur la collaboration en temps réel. J’ai choisit cet outil pour créer mes différentes maquettes et l’utilisation est plutôt simple.

### Technologies côté Back-end

* **Langages et Framework :**

*Next Js : c’est un framework gratuit et open source s’appuyant sur la bibliothèque JavaScript React et sur la technologie Node Js. J’ai choisit ce framework car il prend en charge le côté serveur (Server Side Rendering), le côté statique d’une page web (Static Site Generation) et l’application web est dynamique.*

*Node Js :* c’est une plateforme logicielle libre en JavaScript qui permet le développement de serveur HTTP. Donc qui est axée sur le côté serveur. J’ai choisit cette plateforme car comme j’utilise la librairie de ReactJS, Node permet de créer des applications web évolutives qui peuvent gérer de grandes quantités de trafic et de connexions simultanées.

 *Spring Boot :* c’est un framework Java open source utilisé pour programmer des applications web et de microservices avec Spring Framework grâce a la configuration automatique. J’ai choisit ce framework car il est orienté objet et il est très facile d’utilisation

* **Outils :**



*MySQL Workbench :* c’est un logiciel de gestion et d’administration de base de données MySQL. Via une interface graphique intuitive, il permet, entre autres, de créer, modifier ou supprimer des tables, des comptes utilisateurs, et d'effectuer toutes les opérations inhérentes à la gestion d'une base de données. Pour ce faire, il doit être connecté à un serveur MySQL.

 MongoDB : c’est un système de gestion de base de données orienté documents en NoSQL. Avec ces applications Atlas et Compass on peut gérer les données depuis n’importe ou avec le CRUD (Create Read Update Delete).

TiDB Cloud : c’est une base de données de service qui est open source qui permet de créer ou exporter une base de données MySQL. Comme MongoDB, TiDB utilise un service de Cloud. J’utilise TiDB pour le déploiement de l’application en exportant la base de donnée qui est sur Workbench.

### Technologies côté Persistance des données

* **Langages et Framework :**
* **Outils :**

### Technologies transverses

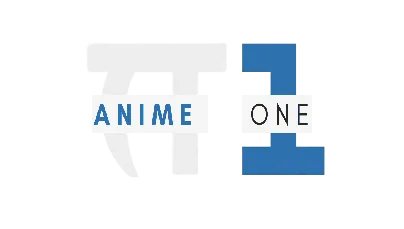
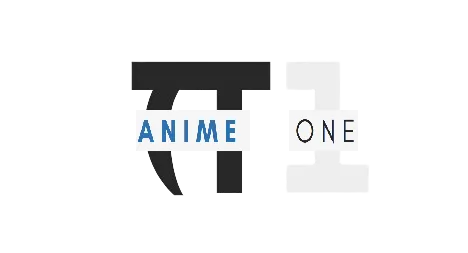
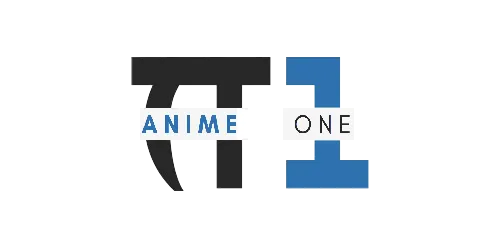
* **Langages et Framework :**
* **Outils :**

# **Conception de l'interface graphique**

## Expérience UX/UI

## Charte graphique

Mes Différents Logos :





Bannière :

Les couleurs changent dynamiquement en fonction de la page (ex : accueil, films, anime sélectionner)→

Cyan foncé (#164E63) :

Teal foncé (#134E4A) :

Rouge (#DC2626) :



Jaune foncé (#FBBF24) :



Styles de texte :

J’ai choisit

(logo, bannière, choix des couleurs et styles de texte)

## Responsive

## Zoning

(versions Desktop, mobile et si possible tablette)

## Wireframes

(versions Desktop, mobile et si possible Tablette)

## Maquettes

(versions Desktop, mobile et si possible tablette)

## Arborescence du site

Il faut mettre une arborescence indiquant l’enchainement des page web

(versions Desktop, mobile)

# **Conception de l'application**

## Le langage de modélisation UML

## Diagramme de cas d'utilisation

## Diagramme de séquences

## Diagramme d’activité

## Diagramme de classes

## Diagramme des entités

# **Conception de la Base de données**

## La méthode MERISE

## Le Modèle Conceptuel de Données (MCD)

## Le Modèle Logique de Données (MLD)

## Le Modèle Physique de Données (MPD)

## Le Dictionnaire de Données (DD)

# **Développement et Intégration**

## Structure du projet

(copie d'écran de l'arborescence de votre projet sur VsCode et/ou Eclipse/Inteliij)

Pour toutes les fonctionnalités ci-après, aites des copies d'écran de vos codes sources partie front (HTML/CSS/JS), back-end (Java/PHP/NextJS /...) et persistance (Java/PHP/NextJS/...)

## Fonctionnalité « s'authentifier »

## Fonctionnalité « s'inscrire »

## Fonctionnalité « payer par CB »

## Fonctionnalité « ajouter un article »

## …etc

# **Sécurité de l'application et des données**

## Les différentes attaques de site

Pour ce qui suit, il faut montre une copie d'écran du code front-end et back-end et persistance qui permet de contrer toutes ses attaques

### Les attaques XSS

### Les attaques CSRF

### Les attaques CROS

### Les attaques Force Brute

### Les attaques de type injection SQL

### Authentification/Autorisation avec JWT

### Authentification Duo/Trio/Quad

Duo : password + case à cocher (code Captcha)

Trio : Duo + question supplémentaire

Quad : Trio + confirmation par email + saisie du code reçu par email

## HTTPS

## La loi RGPD

### Confidentialité des données

(Cryptage des données (du mot de passe) (du numéro de carte CB))

Suivi des recommandations de sécurité de la CNIL : Parler surtout du mot de passe (12 caractères minimum depuis Janvier 2023 et doit être composé au minimum de : un caractère minuscule, un caractère majuscule, un chiffre, un caractère spécial)

## Mentions Légales

### Mentions Légales

### Conditions Générales d'utilisation du site web

### Conditions Générales de vente (sites de commerce ou de service)

### Politique de confidentialité

### Cookies

### Droit de rétractation

# **Sécurité de la base de données**

## Choix des noms de user et des password

Création d'un user avec un nom d'utilisateur et mot de passe complexes (C'est ce user qu'il faut utiliser pour que votre application accède à votre base de données)

## Limiter les droits au user

On créé un user spécifique pour la base de données concernée

## Limiter l'accès à la base de données

uniquement à la machine qui contient votre application serveur.

# **Sécurité et qualité du code**

## Qualité du code

### Convention de nommage

(des méthodes, des classes, des variables, des constantes, des interfaces, ...)

### Conventions de codage

### Journalisation (LOG)

### Commentaires sur les lignes de code

Documentation du code

JavaDoc ou PhpDoc ou autre

## Sécurité du code

### Tests unitaires et couverture de code

(mettre des copies d'écran de vos codes tests unitaires et couverture de code)

### Gestion des erreurs et des exceptions

### Fermeture correcte des flux de données

(Connexions, fichiers, flux, ...)

# **Déploiement et hébergement**

* + Montrer le déploiement du site

# **Référencement SEO**

* + C’est quoi ? pourquoi ? comment le faire ?
  + <https://blog.hubspot.fr/marketing/guide-seo-referencement>
  + <https://www.seo.com/fr/basics/>
  + Si votre application est un intranet (destinée uniquement aux salariés de l’entreprise) alors le référencement n’est pas nécessaire

# **Veilles technologiques**

## Veille globale

Sur les différentes versions (surtout dernières versions), c’est-à-dire les nouveautés, sur les Framework, les librairies, les outils, les environnement de développement, …

## Veille concernant le développement

-- Montrer les copies d'écran des erreurs -- Montrer les démarches faites pour résoudre le bug (recherche internet, solutions etc...)

## Veille sur la sécurité des applications

### Les différentes failles de sécurité les plus dangereuses

<https://owasp.org/>

(TOP 10 des failles de sécurité)

Le numéro 1 des failles de sécurités les plus dangereuses est : l’injection SQL

### Les recommandations de l'ANSSI (<https://cyber.gouv.fr/>)

Agence nationale de la sécurité des systèmes d'information

(suivre les nouvelles recommandations)

<https://cyber.gouv.fr/>

L'ANSSI est l'autorité nationale en matière de sécurité et de défense des systèmes d'information.

### La CNIL et ses recommandations

CNIL (suivre les nouvelles recommandations) (<https://www.cnil.fr/fr>)

## Veille sur la sécurité des bases de données

# **Perspectives**

Dire qu'on va continuer à entretenir et améliorer le projet

Futurs améliorations/fonctionnalités (réaliser la Version 2)

...

# **Conclusion**

# **Annexe**

Annexe A : ….

Annexe B : …

Annexe C : …