Documentation Technique  
  
SUPWARDEN

AUBRIL Melvin

[I. Introduction 2](#_Toc176316500)

[II. Accès 2](#_Toc176316501)

[1. Backend 2](#_Toc176316502)

[2. Web 2](#_Toc176316503)

[III. Guide de déploiement 2](#_Toc176316504)

[1. API 2](#_Toc176316505)

[2. MongoDB 3](#_Toc176316506)

[3. Web 3](#_Toc176316507)

[4. Docker 3](#_Toc176316508)

[IV. Choix des langages / technologies utilisés 3](#_Toc176316509)

[1. Javascript 3](#_Toc176316510)

[2. Node.js 4](#_Toc176316511)

[3. MongoDB 4](#_Toc176316512)

[4. React 4](#_Toc176316513)

# Introduction

# Accès

## Backend

* Swagger API : <http://localhost:3001/api-docs/>
* MongoDB : <http://mongo.localhost:3001/>

## Web

* <http://localhost:3000/>

# Guide de déploiement

## API

* Se rendre dans l’API:
  + cd api
* Edition du .env pour y modifier certaines informations :

(Des mot de passes ont été générés spécifiquement pour la correction du projet).

* + Connexion à la base de données MongoDB :
    - MONGO\_USERNAME=bestadmin
    - MONGO\_PASSWORD=tUfk0ew0A5f5IYcQ
  + ID Google :
    - GOOGLE\_CLIENT\_ID=491106866633-ojomi0e3bdfb8aj90kd4gp9re918j19t.apps.googleusercontent.com
  + Chiffrement :
    - ENCRYPTION\_KEY=d30ad5676fde4bf517e672089f80e443ae3ce364ae87ae7fe73d56b9794e8427
  + Clé JWT :
    - JWT\_KEY=ONCzRs3kqu!yF?wH

## MongoDB

* Se rendre à mongo : rester à la racine
* Edition du .env pour y modifier certaines informations :

(Des mot de passes ont été générés spécifiquement pour la correction du projet).

* + Identifiants MongoDB :
    - MONGO\_USERNAME=bestadmin
    - MONGO\_PASSWORD=tUfk0ew0A5f5IYcQ

## Web

* Se rendre dans le Web :
  + cd web
* Edition du .env pour y modifier certaines informations :

(Des mot de passes ont été générés spécifiquement pour la correction du projet).

* + ID Google :
    - GOOGLE\_CLIENT\_ID=491106866633-ojomi0e3bdfb8aj90kd4gp9re918j19t.apps.googleusercontent.com

## Docker

* Exécuter une commande docker :
  + Commande : docker compose up

Maintenant l’API / MongoDB / Web tournent localement sur docker.

# Choix des langages / technologies utilisés

## Javascript

**Utilisation :** Langage principal pour le développement web côté client.

**Rôle :** Permet de créer des interfaces utilisateur interactives et dynamiques. Supporte un large écosystème de bibliothèques et frameworks.

## Node.js

**Utilisation :** Environnement d'exécution JavaScript pour le développement côté serveur.

**Rôle :** Gère les opérations asynchrones et les entrées/sorties de manière non bloquante, idéal pour les API RESTful et les applications en temps réel.

## MongoDB

**Utilisation :** Base de données NoSQL orientée document.

**Rôle :** Stocke et gère des données non structurées sous forme de documents JSON, offrant flexibilité et scalabilité horizontale pour les applications modernes.

## React

**Utilisation :** Bibliothèque JavaScript pour la construction d'interfaces utilisateur.

**Rôle :** Facilite le développement d'applications à page unique (SPA) avec des composants réutilisables et un rendu optimisé grâce au DOM virtuel.

# Schéma de la base de données