



WhatEat

Prise en main

Cours PDG

HEIG-VD

2022

Table des matières

Introduction	3
Prérequis	3
Configuration du projet	3
Backend	3
Frontend	4
Documentation de l'API	4
Structure du projet	5
Backend	5
Frontend	5
Prise en main par un nouveau développeur de l'équipe WhatEat	6
Prérequis	6
Méthodes de développement	6
Mise en place des outils de développement	6
Environnement de déploiement et pipeline de livraison	6
Pipeline de livraison	7

Introduction

Ce document a pour objectif de vous expliquer comment installer et utiliser WhatEat sur votre machine.

WhatEat est composé d'un frontend et d'un backend. Le frontend est un serveur web qui gère l'affichage des recettes et l'interface utilisateur. Le backend est une API qui propose divers endpoint, appelle la base de données et effectue des appels à l'API de Spoonacular.

Spoonacular est une API qui permet de rechercher des recettes et de les afficher. Si vous souhaitez utiliser WhatEat, il est nécessaire d'obtenir un token pour cette API au préalable.

Prérequis

Les prérequis suivants sont nécessaires pour installer WhatEat :

- [Node.js](#)
- [npm](#)
- [Git](#)

Configuration du projet

Pour pouvoir faire tourner WhatEat sur votre machine, voici les étapes à suivre :

Backend

Le backend utilise NodeJS et Express ainsi qu'une base de données MongoDB.

1. Cloner le projet

```
git clone https://github.com/WhatEatInc/WhatEat-back
```

2. Créer le fichier `.env`

Une fois le projet cloné, il est nécessaire de créer un fichier `.env` qui contiendra les différentes variables d'environnement. Vous pouvez renommer le fichier `.env.example` en `.env` et remplir les différentes variables. Pour ce fichier, vous devez posséder une clé d'API pour Spoonacular ainsi qu'une base de données MongoDB.

Pour le secret du token JWT, vous pouvez saisir un mot de passe aléatoire.

3. Installer les dépendances

```
npm install
```

4. Lancer le serveur

Pour lancer le serveur en mode dev, il suffit de lancer la commande

```
npm run dev
```

Pour lancer le serveur en mode `prod`, il suffit de lancer la commande

```
npm run start
```

Il est ensuite possible d'effectuer des requêtes HTTP sur l'API du backend.

```
GET http://localhost:<PORT>/
```

Frontend

Le frontend utilise ReactJS

1. Cloner le projet

```
git clone https://github.com/WhatEatInc/WhatEat-front
```

2. Créer le fichier `.env`

Une fois le projet cloné, il est nécessaire de créer un fichier `.env` qui contiendra les différentes variables d'environnement. Vous pouvez renommer le fichier `.env.example` en `.env` et remplir les différentes variables.

3. Installer les dépendances

```
npm install
```

4. Lancer le serveur

```
npm run start
```

Vous pouvez ensuite accéder à la page d'accueil du site via l'adresse suivante : <http://localhost:3000/>

Documentation de l'API

La documentation de l'API du backend est disponible à l'adresse suivante :

<https://whateat-backend.herokuapp.com/v0/docs/>

Structure du projet

Pour avoir une meilleure vue d'ensemble du projet, voici la structure des dossiers et fichiers du backend et du frontend.

Backend

Structure des dossiers du backend

apidoc/	Contient la documentation de l'API
src/	
config/	Fichiers de configs nécessaire à l'app
controllers/	Gère la logique
auth.controllers.js	Gestion de l'authentification
recipe.controllers.js	Gestion des recettes
user.controllers.js	Gestion des utilisateurs
models/	Gère les données et leur structure
user.model.js	Modèle de l'utilisateur
preference.model.js	Modèle des préférences
recipe.model.js	Modèle des recettes
routes/	Contient les routes de l'API
apiDoc.routes.js	Routes de la documentation
recipe.routes.js	Routes des recettes
user.routes.js	Routes des utilisateurs
v0.routes.js	Routes de la v0 de l'API
tests/	Contient les tests de l'application
app.js	Point d'entrée de l'application
server.js	Contient la connexion à la DB
.env	Contient les variables d'environnement

Frontend

Structure des dossiers du frontend

public/	Point d'entrée de l'application
index.html	
src/	
components/	Contient des fonctionnalité spécifique de l'app
app/	
App.js	Composant principal de l'application
WhatEat.js	Composant principal du site web
button/	Défini les boutons
counter/	Défini le compteur
error/	Défini les messages d'erreurs
footer/	Défini la bannière en bas de l'application
header/	Défini la bannière en haut de l'application
ingredients/	Défini l'affichage de la liste des ingrédients
recipe-card/	Défini l'affichage de la recette
settings/	Défini les champs des paramètres
steps/	Défini l'affichage des étapes de la recette
config/	Contient les configs pour l'API
css/	Contient les fichiers de style de l'app
base/	Fichiers de styles globaux
components/	Fichiers de styles des composants
images/	Contient les images de l'application
pages/	Contient les pages de l'application
index.js	Point d'entrée de l'application ReactJS
.env	Contient les variables d'environnement

Prise en main par un nouveau développeur de l'équipe WhatEat

Si vous êtes nouveau développeur dans l'équipe de WhatEat et que vous souhaitez prendre en main le projet, voici les différentes étapes à suivre pour pouvoir travailler sur le projet.

Prérequis

Pour pouvoir travailler sur le projet, veuillez avoir lu au préalable la première partie de ce document afin d'avoir les connaissances nécessaires pour installer et configurer le projet.

Méthodes de développement

Lorsque vous collaborez sur le projet WhatEat, merci de bien vouloir respecter les méthodes de développement décrites dans le document [Rapport de projet](#) plus précisément au chapitre [Méthodes de développement](#)

Dans ce chapitre vous trouverez les différentes manières d'organiser les branches, gérer les issues, nommer ses commit, etc...

Mise en place des outils de développement

Afin d'avoir un environnement de développement similaire à celui des autres développeurs, veuillez suivre les étapes décrites dans le document [Rapport de projet](#) plus précisément au chapitre [Mise en place des outils de développement](#)

Environnement de déploiement et pipeline de livraison

Pour terminer, il vous est fortement conseillé d'aller lire la chapitre [Mise en place d'un environnement de déploiement](#) et [Mise en place d'un pipeline de livraison et de déploiement \(CI/CD\)](#) afin de comprendre comment le projet est déployé et comment le pipeline de livraison fonctionne. (Voir la figure ci-dessous)

Pipeline de livraison

Pour rappel, le pipeline de livraison est le suivant :

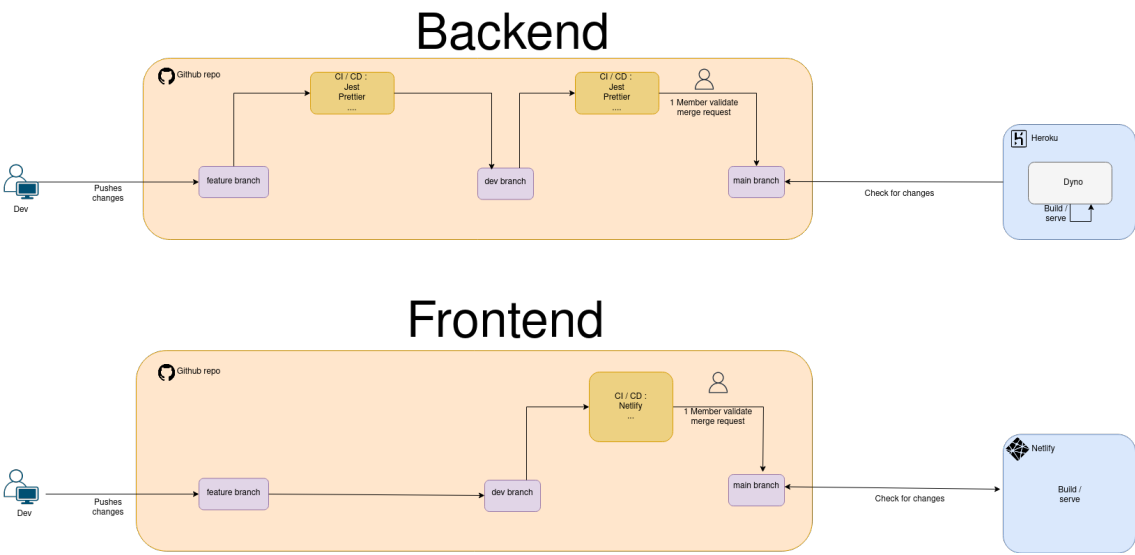


Figure 1: Pipeline de livraison