

CESI ZEN
Cahier des Charges
Référence : INFCDAAL1
Auteur: Bassignot Alexis
Destinataire(s) : ministère de la Santé et de la Prévention



Participants à ce document :

Nom de la personne	Trigramme	Fonction	Rôle	Email
Bassignot	ABA	Dev back	Chef de projet / Développeur	alexis.bassignot@viacesi.fr

Sommaire

1.	RAPPEL DU CONTEXTE	4
1.1.	Contexte	4
1.2.	Terminologie	4
1.3.	Présentation du document.....	5
2.	CHOIX TECHNOLOGIQUES ET ARCHITECTURE	5
2.1.	Méthodologie de développement	5
2.2.	Langages et Frameworks retenus	5
2.2.1.	Solution Backend :	5
2.2.2.	Solution Frontend :	5
2.2.3.	Architecture logicielle	5
3.	DEVELOPPEMENT DE L'APPLICATION	6
3.1.	Modules et fonctionnalités clés	6
3.1.1.	Authentification et gestion des sessions.....	6
3.1.2.	Fonctionnalités principales	6
4.	PLAN DE TEST ET VALIDATION	6
4.1.	Stratégie de tests.....	6
5.	GUIDE D'INSTALLATION	6
5.1.	Prérequis.....	6
5.2.	Installation	7
6.	CAHIER DE TESTS	7

1. RAPPEL DU CONTEXTE

1.1. Contexte

Le ministère de la Santé et de la Prévention est chargé de la mise en œuvre des politiques publiques en matière de santé mentale, de prévention, et de promotion du bien-être général. En collaboration avec d'autres institutions, il s'engage à sensibiliser les citoyens aux enjeux liés à leur santé mentale, tout en offrant des solutions concrètes pour prévenir les effets négatifs du stress et améliorer leur qualité de vie.

Dans ce contexte, le projet CESI Zen propose une plateforme numérique à destination d'un large public tels que des enfants, les adultes ou les personnes âgées a pour vocation d'informer sur les risques liés au stress chronique et de promouvoir une meilleure gestion des émotions au quotidien. L'objectif principal de CESI Zen est de fournir des exercices interactifs de relaxation et de respiration.

En s'alignant sur les missions réelles du ministère et en intégrant une dimension technologique innovante, CESI Zen ambitionne de moderniser l'accès à la prévention en santé mentale, tout en favorisant une prise de conscience collective autour de ces enjeux cruciaux. Cette initiative s'inscrit dans une démarche globale visant à rendre le soutien psychologique accessible à tous, à tout moment, et en tout lieu.

1.2. Terminologie

Acronyme	Définition
Usecase	Scénarios d'utilisation de l'application
Javascript	Langage de programmation
CSS	Fichier texte qui permet de définir l'affichage visuel d'un site internet
Bootstrap	Fichier CSS avec un style prédéfini
BDD	Base de données
Frontend	Serveur utilisé par l'utilisateur final qui permet d'afficher les pages sur le site internet
Backend	Serveur non accessible par l'utilisateur final, sert à stocker les données et à les traiter
Framework	Environnement de travail simplifié pour le développement
API	Interface de communication avec le backend
MySQL	Base de données

(L'idée est de pouvoir utiliser librement ces acronymes tout en rendant le document lisible par les acteurs non gestionnaires, et inversement par les acteurs non informaticiens)

1.3. Présentation du document

Ce document constitue le cahier des charges pour le projet Cesi Zen. Il vise à formaliser les besoins, les contraintes, et les attentes afin d'établir une compréhension commune entre toutes les parties prenantes : le ministère de la Santé et de la Prévention (commanditaire), et les utilisateurs finaux. Le document est organisé en plusieurs sections complémentaires :

1. **Rappel du contexte** : Cette section présente les besoins du projet, ses objectifs, et ses enjeux principaux.
2. **Description générale des besoins** : Elle propose une vue d'ensemble des fonctionnalités attendues, des avantages escomptés, et des risques associés à une non-réalisation. Une première ébauche des modules de l'application y est également incluse.
3. **Exigences détaillées** : Cette partie décrit les attentes fonctionnelles et techniques à l'aide de tableaux, schémas et descriptions, en tenant compte des contraintes de sécurité, de compatibilité, et de budget.
4. **Architecture technique** : Elle détaille l'organisation logicielle prévue, la modélisation des données, et les schémas techniques nécessaires pour répondre aux exigences.
5. **Annexes** : Enfin, les annexes regroupent des documents complémentaires et des références utiles pour la réalisation du projet.

2. CHOIX TECHNOLOGIQUES ET ARCHITECTURE

2.1. Méthodologie de développement

- Le code est versionné sur github, lien : <https://github.com/SNEAXIII/Cesi-Zen>
- Le code a une large couverture de tests

2.2. Langages et Frameworks retenus

2.2.1. Solution Backend :

- Language : Python 3.12
- Base de données : Mysql
- Hashage de mot de passe : Bcrypt
- Serveur : FastApi avec Uvicorn (serveur ASGI)
- ORM : SqlModel
- Validation de données : Pydantic
- Outil de migration de base de données : Alembic
- Sanitization de code HTML : Bleach
- Framework de tests unitaires : Pytests
- Stratégie d'autorisation/authentification : JWT

2.2.2. Solution Frontend :

- Language : TypeScript
- Serveur : Next.js 15 avec React 19 et NextAuth.js
- Composants : Shadcnui avec TailwindCss
- Editeur de texte WYSWYG : Tiptap

2.2.3. Architecture logicielle

- Architecture MVC avec API / client
- Api RESTful.
- Base de données MySQL.
- Authentification par JWT.
- Documentation OpenAPI (Swagger).

- Appel API avec fetch/axios.

3.DEVELOPPEMENT DE L'APPLICATION

3.1.Modules et fonctionnalités clés

3.1.1. Authentification et gestion des sessions

- Inscription/Connexion utilisateur.
- Gestion des tokens JWT.
- Protection des routes.
- Récupération de mot de passe.

3.1.2. Fonctionnalités principales

- Exercices de respiration :
 - Minuteur personnalisable.
 - Guide visuel.
 - Historique des sessions.
- Articles informatifs :
 - Triables par catégorie
 - Editeur de texte enrichi
 - Ajout et suppression par les administrateurs
- Tableau de bord utilisateur pour les administrateurs
 - Suivi des activités.
 - Statistiques.
 - Préférences.

4.PLAN DE TEST ET VALIDATION

4.1.Stratégie de tests

- **Tests unitaires :**
 - Couverture minimale de 90% (actuellement 91%)
 - Tests des services et des validator critiques.
 - Tests des validateurs et utilitaires.
- **Tests End-to-End :**
 - Scénarios utilisateur complets pour les cas critiques.

Voir annexe 1

5.GUIDE D'INSTALLATION

Lien du projet : <https://github.com/SNEAXIII/Cesi-Zen>

5.1. Prérequis

- Docker / docker compose
- Node.js / npm
- Python / pip
- GNU make (optionnel, si vous ne souhaitez pas l'utiliser, veuillez utiliser les commandes dans le Makefile)
- Git cli

5.2. Installation

Dans un terminal PowerShell :

```
# Télécharger le projet  
git clone https://github.com/SNEAXIII/Cesi-Zen
```

```
# Installer et lancer le front  
cd front  
npm ci  
npm build  
npm start
```

```
# Installer le backend  
make create-venv  
make dl_dev_reqs
```

```
# Migrer la base de données  
make migrate
```

```
# Charger les fixtures (une seule fois)  
make fixtures
```

```
# Lancer les tests u avec coverage  
make test-cov
```

```
# Lancer le backend  
make run-dev
```

6.CAHIER DE TESTS

- **Scénario 1:** Connexion réussie avec des identifiants valides
 - Login : admin_active_0, mot de passe : test
- **Scénario 2:** Inscription réussie avec des informations valides
 - Login : User, mot de passe et mot de passe confirmation: Securepass1!, email : user@gmail.com
- **Scénario 3:** Création d'un article avec succès
 - **En temps qu'administrateur ;** Titre : Article, contenu : Contenu, Catégorie
- **Scénario 4:** Affichage de la liste des articles
 - L'article précédemment créé apparait dans la liste
- **Scénario 5:** Consultation du détail de l'article créé avant
 - Les infos sont les mêmes que saisies précédemment