



SESAME
TECHNOLOGY

المدرسة العليا للعلوم التطبيقية والتصرف
ÉCOLE SUPÉRIEURE PRIVÉE DES SCIENCES
APPLIQUÉES ET DE MANAGEMENT

CAHIER DE CHARGE DE PROJET DE FIN D'ETUDES INGENIEURS EN INFORMATIQUE

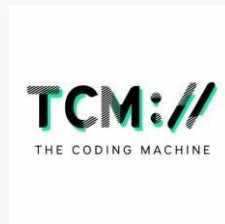
Option : Génie Logiciel/DevOps

Intitulé du stage

Développement d'une application web envie autonomie

Réalisé par
Amal Grami

Entreprise d'accueil



THE CODING MACHINE

Encadrant Entreprise
Louis CAILLE

Encadrant SESAME
Amal BOUAZIZ

**Année Universitaire
2022-2023**

Sommaire

Table des matières

1 - Contexte et définition du projet	3
2 - Objectif du projet	3
3- Acteurs / Les actions associées à chaque acteur	3
4 – Choix de technologie	4
<i>Environnement Logiciel</i>	<i>4</i>
<i>Langages de programmation.....</i>	<i>4</i>
5– Méthodologie de conception adoptée.....	5
Scrum	5
Les Rôles SCRUM.....	5
6– Architecture	6

1 - Contexte et définition du projet

Dans le cadre de notre projet de fin d'études chez The Coding Machine, nous sommes chargés de concevoir et développer une plateforme spécialisée dans le domaine de la santé appelée "LEA"

2 - Objectif du projet

LEA vise à faciliter la collecte, le reconditionnement et la remise en bon état des matériels médicaux. En offrant aux établissements de santé un processus fluide et sécurisé, notre plateforme permettra de récupérer les équipements médicaux usagés, de les restaurer à un état fonctionnel optimal, et de les réintégrer dans leur inventaire.

3- Acteurs / Les actions associées à chaque acteur

Dans le cadre de notre projet, il est essentiel de définir les acteurs impliqués ainsi que les actions qui leur sont associées afin de garantir la réussite de notre entreprise.

Acteur	Action
Administrateur	Paramétrer et gérer les rôles. <ul style="list-style-type: none">• Gérer les agences.• consulter les statistiques.
Utilisateur	<ul style="list-style-type: none">• S'authentifier.• Gérer les articles.• Gérer les collectes.• Gérer les tournées.
Ingénieur	<ul style="list-style-type: none">•Simplifier l'installation du projet pour les développeurs en réduisant le nombre d'étapes nécessaires et en automatisant autant que possible le processus d'installation.• Remédier les problèmes de configuration d'environnement.• Isoler les dépendances afin de permettre aux développeurs de travailler et d'effectuer des évolutions sur chaque partie séparément, sans interférer avec les autres composants du projet.

4 – Choix de technologie

Environnement Logiciel

PhpStorm

PhpStorm est un environnement de développement intégré (IDE) spécifiquement conçu pour le développement d'applications en PHP. Il est développé par JetBrains L'IDE offre une intégration étroite avec les outils de gestion de versions tels que Git, Mercurial et SVN, ce qui facilite la collaboration sur les projets. Il prend en charge les frameworks populaires tels que Symfony, Laravel, WordPress, Drupal, etc., et offre des fonctionnalités spécifiques à ces frameworks.

GitLab

GitLab est une plateforme de gestion de code source très complète qui offre une multitude de fonctionnalités pour le développement collaboratif. Son principal avantage réside dans la combinaison du contrôle de version écentralisé avec Git, permettant aux développeurs de travailler ensemble sur un même projet, de suivre l'historique des modifications et de fusionner facilement les branches de code. GitLab intègre également des fonctionnalités avancées d'intégration continue et de déploiement continu (CI/CD), automatisant ainsi les tests, les builds et les déploiements d'applications.

Langages de programmation

Notre choix de langage de programmation est basé sur une évaluation approfondie des avantages et des inconvénients de chaque langage. Il est essentiel de sélectionner un langage en tenant compte des besoins spécifiques de l'entreprise afin d'éviter les pertes de temps liées à un changement de langage en cours de projet. Pour la réalisation de notre solution, nous avons opté pour Symfony, un framework PHP, pour la partie backend, et Vue.js, un framework JavaScript, pour le frontend. Cette combinaison de technologies nous permet de bénéficier des fonctionnalités avancées offertes par Symfony pour la gestion de la logique métier et du traitement des requêtes, ainsi que de la flexibilité et de la réactivité de Vue.js pour la création d'interfaces utilisateur dynamiques et interactives.

5- Méthodologie de conception adoptée

Scrum

SCRUM tient son origine du terme sportif de rugby signifiant : mêlée. Tout comme cet aspect technique de la partie du jeu, la méthodologie demande à ses acteurs d'être soudés dans l'accomplissement d'un projet, dans l'atteinte d'un but. SCRUM est un processus agile qui permet de produire la plus grande valeur métier dans la durée la plus courte.

Caractéristique du Scrum

- Itératif, lié à des processus incrémentaux.
- Approche basée sur l'équipe.
- Assure le développement des produits/applications nécessitant une grande adaptabilité.
- Contrôler le chaos résultat de conflits d'intérêt et des différents besoins.
- Augmenter la communication et maximiser la coopération.
- Protéger l'équipe des éléments externes perturbateurs.
- Un moyen d'augmenter la productivité.

Les Rôles SCRUM

SCRUM définit trois rôles principaux :

- Le « Product Owner » qui porte la vision du produit à réaliser (représentant généralement le client).
- Le « Scrum Master » garant de l'application de la méthodologie Scrum.
- Le « Scrum Team » L'équipe de développement qui réalise le produit.

La figure ci-dessous présente le principe de fonctionnement du SCRUM.



6– Architecture

Architecture client-serveur

Pour notre application nous avons orientée vers une architecture à deux niveaux, également connue sous le nom d'architecture client-serveur, est un modèle d'architecture logicielle dans lequel une application est divisée en deux parties distinctes : le backend et le frontend.

Le backend, également appelé serveur, est responsable de la gestion des données, de la logique métier et de l'exécution des opérations. Il fournit des services et des fonctionnalités à travers des API ou des services web, répondant ainsi aux demandes des clients.

Le frontend, également appelé client, est la partie de l'application avec laquelle l'utilisateur interagit directement. Il est responsable de la présentation des données, de l'interface, utilisateur et de la gestion des interactions utilisateur. Le frontend communique avec le backend en utilisant les API ou les services fournis par celui-ci.

Voici quelques avantages cette architecture :

- Séparation claire des responsabilités entre le backend et le frontend.
 - Facilité de maintenance et de gestion de l'application grâce à la division des tâches.
 - Possibilité de développer et de déployer indépendamment le backend et le frontend, ce qui permet des cycles de développement plus rapides.
 - Scalabilité facilitée en permettant d'ajuster et de mettre à l'échelle individuellement les ressources du backend et du frontend.
- Réutilisabilité du code grâce à la modularité de l'architecture, ce qui facilite le développement de nouvelles fonctionnalités et

