

Cahier des charges



>_CypherCode

CypherCode



NOEL Andy 3B1

Présentation	2
Contexte	2
Objectifs du cahier des charges	2
Public cible	2
Cahier des Charges Fonctionnel	2
Objectifs	2
Les Besoins	3
User Stories	4
Description fonctionnelle des besoins	5
Critères de Qualité	6
Cahier des Charges Technique	7
Langage et Application	7
Base de Données	7
Sécurité	7
Dockerisation	8
Gestion et Suivi du Projet	9
Charte Graphique	11
Couleurs Principales	11
Les Logos	11
Polices d'écritures	12
Icônes	12



Ce document est un Cahier des Charges que j'ai produit seul, il sert de preuve de participation à la SAE depuis que j'ai rejoint le groupe Jack Hascoet, Damien Morice—Legouet, Jérémy Beatrix, Maxime Chargé

Présentation

Contexte

La création de cette web app découle de la nécessité de former efficacement de jeunes développeurs au sein de CypherCode. Nous cherchons à offrir un environnement d'apprentissage dynamique, stimulant la compétitivité tout en favorisant le développement de compétences. L'objectif ultime est d'améliorer les capacités de nos développeurs actuels, de créer de nouveaux talents et de susciter l'intérêt des étudiants pour une future collaboration.

Objectifs du cahier des charges

Ce cahier des charges vise à fournir une feuille de route claire pour le développement de la plateforme. Nous souhaitons créer une application web engageante, favorisant l'apprentissage continu et la compétition entre les développeurs. Les résultats escomptés incluent une amélioration mesurable des compétences techniques de nos employés en formation, ainsi que la création d'un vivier de talents potentiels parmi les étudiants participants.

Public cible

Les utilisateurs finaux de cette application sont principalement nos employés en cours de formation, mais également les étudiants en informatique que nous envisageons d'accueillir dans notre entreprise. En outre, cette plateforme est conçue pour être accessible et intéressante pour des individus novices en informatique, ouvrant ainsi la porte à un public plus large.

Cahier des Charges Fonctionnel

Objectifs

Le présent chapitre énonce les objectifs visés par la conception et le développement de la plateforme de défis pour développeurs. Ces objectifs sont formulés de manière à guider le processus de création de la plateforme tout en assurant une compréhension claire des attentes.

Objectifs généraux

Promouvoir l'apprentissage interactif : La plateforme vise à offrir un environnement interactif propice à l'apprentissage continu des étudiants en informatique en proposant des défis stimulants et diversifiés.

Faciliter la collaboration : Encourager les étudiants à travailler en équipe en intégrant des fonctionnalités favorisant la collaboration dans la résolution des défis.

Mesurer la progression individuelle : Fournir aux utilisateurs des outils de suivi de leur progression à travers des classements, des statistiques et des retours spécifiques.

Objectifs spécifiques

Gestion des Défis : Permettre aux utilisateurs de créer, partager et participer à des défis de programmation, en spécifiant les langages autorisés, les niveaux de difficulté, et en facilitant la soumission des solutions.

Comptes Utilisateurs : Mettre en place un système d'authentification sécurisé, permettant la création de profils utilisateurs, la gestion des historiques de participation, et la personnalisation des expériences individuelles.

Système de Classement : Établir un système de classement dynamique basé sur la performance des utilisateurs dans la résolution des défis, avec des critères de pondération prenant en compte la difficulté des défis relevés.

Partie Administration : Définir une interface d'administration réservée aux gestionnaires de défis, offrant des fonctionnalités de modération, de création et de suivi des défis.

Page d'accueil : Concevoir une page d'accueil attrayante et informative, présentant de manière dynamique les défis en cours, les classements, et les annonces importantes.

Les Besoins

Cette section détaille les principaux besoins de la plateforme, couvrant divers aspects tels que la gestion des défis, les comptes utilisateurs, le système de classement, la partie administration et la page d'accueil.

Gestion des Défis

La plateforme devra offrir une fonctionnalité complète de gestion des défis. Cela inclut la création, la modification et la suppression de défis. Chaque défi devra comporter des informations telles que le titre, la description, la difficulté, et éventuellement une solution de référence. De plus, les utilisateurs devraient pouvoir soumettre leurs solutions aux défis.

Comptes Utilisateurs

Un système de gestion des comptes utilisateurs est essentiel. Les fonctionnalités attendues comprennent l'inscription, la connexion, la récupération de mot de passe, la modification de profil et la suppression de compte. Les utilisateurs devraient avoir des profils personnalisables, incluant des informations telles que le nom, l'image de profil, et éventuellement un historique des défis résolus.

Système de Classement

La plateforme doit mettre en place un système de classement des utilisateurs en fonction de leurs performances dans la résolution de défis. Les critères de classement pourraient inclure le nombre de défis résolus, la difficulté des défis résolus, ou d'autres métriques pertinentes. Les classements devraient être accessibles publiquement.

Partie Administration

Une interface d'administration est nécessaire pour gérer les aspects globaux de la plateforme. Cela devrait inclure la gestion des utilisateurs, la supervision des défis, la modération des contenus, et la visualisation des statistiques globales. Les administrateurs devraient également pouvoir gérer les rapports d'abus et prendre des mesures appropriées.

Page d'accueil

La page d'accueil de la plateforme doit être soigneusement conçue pour offrir une expérience utilisateur attrayante. Elle devrait présenter des défis recommandés, les classements récents, et des informations importantes. La navigation intuitive et un appel à l'action clair pour inciter les utilisateurs à s'inscrire ou à explorer les défis sont également des éléments clés.

User Stories

Utilisateur ne possédant pas de compte

En tant qu'utilisateur ne possédant pas de compte, je souhaite pouvoir :

- Accéder à la page d'accueil de la plateforme.
- Consulter la liste des défis disponibles sans avoir à m'inscrire.
- Obtenir des détails sur un défi spécifique.
- M'inscrire sur la plateforme pour participer aux défis.
- Recevoir des informations claires sur le processus d'inscription.

Priorités :

- 1) Accéder à la page d'accueil.
- 2) Consulter la liste des défis.
- 3) Obtenir des détails sur un défi.
- 4) S'inscrire sur la plateforme.

Utilisateur possédant un compte

En tant qu'utilisateur possédant un compte, je souhaite pouvoir :

- Me connecter à la plateforme.
- Soumettre des solutions pour les défis.
- Suivre mon classement par rapport à d'autres utilisateurs.
- Recevoir des notifications sur les nouveaux défis.
- Mettre à jour mes informations de profil.
- Réinitialiser mon mot de passe si nécessaire.

Priorités :

- 1) Se connecter à la plateforme.
- 2) Soumettre des solutions pour les défis.
- 3) Suivre mon classement.
- 4) Recevoir des notifications sur les nouveaux défis.
- 5) Mettre à jour le profil.
- 6) Réinitialiser le mot de passe.

Gestionnaire de Défis (Administrateur)

En tant que gestionnaire de défis (administrateur), je souhaite pouvoir :

- Créer de nouveaux défis avec des descriptions détaillées.
- Modifier les détails d'un défi existant.
- Supprimer un défi si nécessaire.
- Visualiser les statistiques et les performances des utilisateurs.
- Suspendre temporairement l'accès à la plateforme en cas de maintenance.
- Recevoir des notifications sur les activités importantes de la plateforme.
-

Priorités :

- 1) Créer de nouveaux défis.
- 2) Modifier les détails des défis existants.
- 3) Supprimer un défi.
- 4) Visualiser les statistiques des utilisateurs.
- 5) Suspendre temporairement l'accès pour la maintenance.
- 6) Recevoir des notifications sur les activités importantes.

Description fonctionnelle des besoins

Utilisateur

Les utilisateurs de la plateforme seront confrontés à différentes fonctionnalités en fonction de leur statut. Un utilisateur non enregistré pourra explorer les défis disponibles, visualiser les classements publics, mais ne pourra pas participer activement ou créer des défis. En revanche, un utilisateur enregistré bénéficiera d'un profil personnel, pourra soumettre des solutions aux défis, et accumulera des points basés sur ses performances.

Les défis

La gestion des défis constitue le cœur de la plateforme. Chaque défi doit être défini de manière claire, comprenant un énoncé précis, des critères d'évaluation, et éventuellement des ressources associées. Les utilisateurs pourront rechercher, filtrer et sélectionner des défis en fonction de leurs préférences et compétences.

Système de Points

Chaque défi sera associé à un système de points visant à évaluer la performance des utilisateurs. Les critères de notation incluront la difficulté de la tâche, l'efficacité du code, et le nombre de tests réussis. Ces éléments contribueront à la création d'un classement global reflétant la compétence et l'efficacité des participants.

Difficulté des Défis

Les défis seront catégorisés en fonction de leur niveau de difficulté. Cette classification permettra aux utilisateurs de choisir des défis adaptés à leurs compétences, encourageant ainsi une progression graduelle. Les niveaux de difficulté pourront être définis, par exemple, comme débutant, intermédiaire et avancé, offrant une diversité d'options pour tous les niveaux d'expertise.

Catégories de Défis

Diverses catégories de défis seront définies pour couvrir un large éventail de compétences et de domaines. Ces catégories pourraient inclure, mais ne seraient pas limitées à, algorithmes, manipulation de données, conception d'interface utilisateur, etc. Chaque catégorie devra être clairement définie, permettant aux utilisateurs de trouver des défis correspondant à leurs intérêts spécifiques.

Remise du Code

La plateforme acceptera uniquement des défis principalement codés en JavaScript. Les utilisateurs soumettront leurs solutions directement sur la plateforme, où le système évaluera automatiquement la validité et l'efficacité du code. Les utilisateurs auront la possibilité de soumettre plusieurs versions de leur code.

Les classements

La plateforme doit offrir un système de classement dynamique, mettant en valeur les performances des utilisateurs en fonction de divers critères tels que la précision de leurs solutions, la fréquence de participation, etc. Les classements devront être accessibles à tous les utilisateurs enregistrés, encourageant ainsi une saine compétition.

Droits

La gestion des droits d'accès et des privilèges sera cruciale pour garantir une expérience utilisateur sécurisée et personnalisée. Différents niveaux d'autorisation seront définis, distinguant les utilisateurs standards, les administrateurs et les créateurs de défis. Les administrateurs auront un accès complet à toutes les fonctionnalités, tandis que les créateurs de défis auront des droits spécifiques pour concevoir et gérer leurs défis respectifs.

Page d'accueil

La page d'accueil servira de vitrine principale de la plateforme, offrant une vue d'ensemble des défis en cours, des classements actuels, et des dernières activités. Elle devra être conviviale, intuitive, et inciter les utilisateurs à explorer davantage la plateforme. Des sections distinctes pour les nouveaux défis, les défis populaires, et les actualités récentes seront mises en avant pour encourager une participation active.

Critères de Qualité

La qualité de la plateforme sera évaluée selon des critères clairement définis, permettant une mesure objective de la performance et de la satisfaction des utilisateurs. Ces critères de qualité visent à assurer le bon fonctionnement, la sécurité et l'expérience utilisateur optimale de la web app.

Cahier des Charges Technique

Le volet technique du cahier des charges se concentre sur les aspects liés au développement de la plateforme. Cette section détaille les choix de langage et d'architecture applicative, ainsi que les technologies spécifiques à utiliser.

Langage et Application

Single Page Application (SPA)

La plateforme sera développée en tant que Single Page Application (SPA), offrant une expérience utilisateur fluide et réactive. Cette approche permettra de minimiser les temps de chargement des pages en ne rechargeant que les éléments nécessaires lors de l'interaction de l'utilisateur.

Technologies Client / Serveur

Les technologies utilisées pour la partie client/serveur devront être soigneusement sélectionnées pour assurer une performance optimale et une interaction efficace entre l'utilisateur et la plateforme. Des frameworks modernes et bien établis seront privilégiés pour garantir la stabilité et la pérennité du développement (React, Vue, Svelte,...).

Base de Données

La gestion de la base de données constitue un élément crucial pour assurer la performance et la stabilité de la plateforme. Les sections suivantes détaillent les choix liés à la base de données.

Type de Base de Données

Le choix du type de base de données est essentiel pour répondre aux besoins spécifiques de la plateforme. Nous recommandons l'utilisation d'une base de données relationnelle pour garantir une structure organisée et une manipulation efficace des données. MySQL serait sûrement le choix à envisager en fonction des exigences spécifiques du projet.

Modèle de Données

Le modèle de données détermine la manière dont les informations seront structurées et organisées dans la base de données. Pour notre plateforme de défis pour développeurs, un modèle de données bien conçu est nécessaire. Les entités clés à considérer incluent les utilisateurs, les défis, les classements, et toute autre information pertinente.

Le modèle de données doit prendre en compte les relations entre ces entités, les contraintes d'intégrité, et les mécanismes de stockage efficaces. Un schéma clair facilitera la maintenance et l'évolution future de la base de données.

Sécurité

Sécurité de la Connexion

La sécurité de la connexion est d'une importance capitale pour garantir l'intégrité des données et la confidentialité des utilisateurs. Afin d'assurer un niveau élevé de sécurité de la connexion, les points suivants doivent être pris en compte :

Protocole HTTPS : Toutes les communications entre le client et le serveur doivent être chiffrées à l'aide du protocole HTTPS pour éviter toute interception non autorisée.

Gestion des Sessions : Mise en place d'un mécanisme robuste de gestion des sessions, avec des sessions expirantes, pour réduire les risques liés aux attaques de session via l'utilisation de cookie de connexion avec JWT par exemple.

Système de Logs

La mise en place d'un système de logs efficace est essentielle pour la détection précoce des anomalies, la résolution des problèmes et la traçabilité des actions effectuées sur la plateforme. Les éléments clés du système de logs comprennent :

Enregistrement des Événements : Capture des événements importants, tels que les connexions réussies/échouées, les opérations d'administration, et les erreurs système.

Niveaux de Logs : Utilisation de niveaux de logs (info, warning, error) pour hiérarchiser les informations enregistrées en fonction de leur importance.

Stockage Sécurisé : Les logs doivent être stockés de manière sécurisée, avec un accès restreint aux administrateurs système.

Sécurité des Requêtes

La sécurité des requêtes est cruciale pour prévenir les attaques par injection et assurer l'intégrité des données stockées. Les mesures de sécurité des requêtes comprennent :

Validation des Entrées : Toutes les entrées utilisateur doivent être validées et filtrées pour éviter les injections SQL et les attaques XSS.

Paramètres Préparés : Utilisation de requêtes préparées pour les interactions avec la base de données afin de réduire les risques liés aux attaques par injection SQL.

Mots de Passe Hachés

La sécurité des mots de passe est fondamentale pour protéger les comptes des utilisateurs. Ainsi, les mots de passe doivent être stockés de manière sécurisée à l'aide de techniques de hachage. Les pratiques de sécurité des mots de passe incluent :

Hachage Fort : Utilisation d'algorithmes de hachage robustes et réputés pour stocker les mots de passe de manière sécurisée.

Sel Unique : Application d'un sel unique à chaque mot de passe avant le hachage pour renforcer la sécurité contre les attaques par dictionnaire et par force brute.

Politique de Mot de Passe : Mise en place d'une politique de mot de passe exigeante, encourageant les utilisateurs à choisir des mots de passe complexes.

Dockerisation

La dockerisation de la web app implique la création de conteneurs distincts pour la base de données (BDD), l'API et le serveur web. Cette approche offre une gestion simplifiée des dépendances et assure la portabilité de l'application entre différents environnements.

Méthode de déploiement

Le déploiement de l'application reposera sur l'utilisation de conteneurs Docker pour assurer une mise en production efficace. La méthode de déploiement suivra les étapes ci-dessous :

Construction des Images Docker : Les images Docker distinctes seront créées pour la base de données, l'API et le serveur web, garantissant une isolation adéquate des composants.

Utilisation de Docker Compose : Docker Compose sera employé pour orchestrer le démarrage simultané des conteneurs. Cela facilitera la gestion des différents services et de leurs interactions.

Déploiement Multi-Environnement : La configuration de Docker Compose permettra une transition fluide entre les environnements de développement, de test et de production.

Gestion des Variables d'Environnement : Les variables d'environnement seront utilisées pour paramétrer les différentes configurations selon le contexte, assurant ainsi une flexibilité accrue.

Configuration Docker

La configuration des conteneurs Docker sera détaillée comme suit :

Conteneur Base de Données

- Image Utilisée : Image MySQL officielle
- Port Exposé : Configuration du port sur lequel la base de données sera accessible. (3306)
- Volume Docker : Utilisation de volumes pour assurer la persistance des données.

Conteneur API

- Image Utilisée : Une image spécifique pour l'API, intégrant les dépendances nécessaires.
- Communication avec la BDD : Configuration des paramètres de connexion à la base de données. (443,8080)
- Port Exposé : Définition du port sur lequel l'API sera exposée.

Conteneur Serveur Web

- Image Utilisée : Une image personnalisée pour le serveur web.
- Communication avec l'API : Configuration des points de terminaison pour l'interaction avec l'API.
- Port Exposé : Spécification du port d'accès au serveur web. (443,8080)

Gestion et Suivi du Projet

La gestion et le suivi du projet sont des éléments cruciaux pour assurer le développement réussi de la plateforme. Cette section du cahier des charges détaille les outils et processus spécifiques qui seront utilisés dans ce contexte.

Forge logicielle

La forge logicielle constitue l'infrastructure de gestion de code source qui facilitera le travail collaboratif des équipes de développement. Pour garantir une collaboration efficace, le choix se porte sur une forge logicielle de type git, accessible par CypherCode.

Cette plateforme devra offrir des fonctionnalités robustes de gestion de versions et d'intégration continue pour assurer la cohérence du code développé.

Jira

Jira sera l'outil central pour le suivi de projet. Il permettra de planifier, suivre et gérer les tâches liées au développement de la plateforme. Les équipes BUT2 auront accès à Jira, facilitant ainsi la communication et la collaboration entre les différentes parties prenantes du projet. Les fonctionnalités de Jira, telles que la création de tickets, la planification des sprints et le suivi des problèmes, seront pleinement exploitées pour assurer une gestion efficace du projet.

Charte Graphique

Couleurs Principales

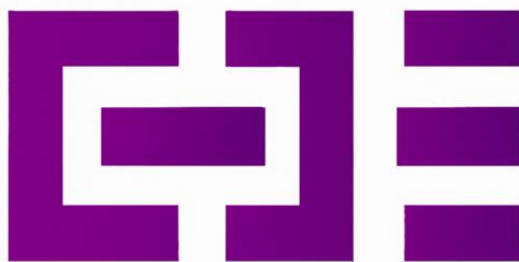
La palette de couleurs adoptée pour la plateforme sera harmonieuse et inspirée par le logo, qui est violet. Les couleurs principales seront les suivantes :

#7C0186	#0F0F0F	#FBFBFB	#4A7856	#BCABAE
---------	---------	---------	---------	---------

- Violet (Logo): #7C0186
- Fond: #FBFBFB
- Texte Principal: #0F0F0F
- Accent 1: #4A7856
- Accent 2: #BCABAE

Les Logos

Le logo de la plateforme adopte la couleur violette (#7C0186).



Logo simplifié sans texte



Logo avec texte

Polices d'écritures

La plateforme utilisera deux polices complémentaires pour assurer une variété visuelle tout en préservant la lisibilité. La première sera "FiraCode Nerd Font SemiBold" pour les éléments principaux. En complément, la seconde police sera "Montserrat Regular", une police sans empattement, pour ajouter une touche d'élégance et de clarté aux zones de texte et aux titres.

Icônes

L'utilisation d'icônes, notamment à partir de la bibliothèque Font Awesome, sera privilégiée pour améliorer l'expérience utilisateur mais d'autres bibliothèques peuvent être utilisées. Des icônes pertinentes seront intégrées aux différentes sections de la plateforme pour faciliter la navigation et la compréhension des fonctionnalités.