REPUBLIC OF CAMEROON

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

MINISTRY OF THE HIGHER EDUCATION

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

SAINT JEAN UNIVERSITY

REPUBLIQUE DU CAMEROUN

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

MINISTERE DE L’ENSEIGNEMENT SUPPERIEUR

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

UNIVERSITE SAINT JEAN



**DEVELOPPEMENT MOBILE**

**Thème : Mboa Culture**

**Groupe 7 :**

* **MASSEHE TAGOUFFO**

***Semestre 1***

**Classe:** *ING 4 ISI*

**Superviseur:** Mr. PESSA

Année académique 2023-2024

1. **Document de Réalisation - Mboa Culture Quizz**
2. **Introduction**

Bienvenue dans le monde captivant de Mboa Culture, une application mobile conçue pour explorer et préserver les richesses culturelles du Cameroun et de l'Afrique. Mboa Culture vous emmène dans un voyage interactif à travers les questions stimulantes de notre quizz, couvrant des aspects variés tels que la population, la culture, la gastronomie, la musique, et bien plus encore.

Notre objectif est de célébrer la diversité des cultures camerounaises et africaines tout en offrant une expérience ludique et éducative. Chaque défi propose trois niveaux de difficulté, permettant aux joueurs de tester leurs connaissances, que ce soit en tant que débutant curieux, intermédiaire avide ou expert passionné.

Découvrez les subtilités de l'organisation administrative au sein des ethnies, plongez dans la richesse musicale et explorez les délices de la gastronomie. Que vous soyez un fervent défenseur de la culture africaine ou simplement curieux d'en apprendre davantage, Mboa Culture offre une expérience engageante et divertissante.

Au fil de ce document, nous allons plonger dans la conception détaillée de l'application, explorer son architecture, ses fonctionnalités principales et les défis passionnants qu'elle propose. Préparez-vous à être immergé dans une aventure culturelle unique, où le savoir et le divertissement se rencontrent.

* 1. **Objectif**

L'objectif de ce document est de fournir une feuille de route détaillée pour la conception, le développement, et le déploiement de l'application mobile "Mboa Culture Quizz".

**1.2 Portée**

Ce document couvre les aspects techniques, fonctionnels, et les échéances du projet.

**2. Caractéristiques Principales**

**2.1 Rubriques et Niveaux**

L'application comprend les rubriques suivantes :

* Population
* Culture
* Organisation administrative
* Gastronomie
* Musique

Chaque rubrique est divisée en trois niveaux : Débutant, Intermédiaire, Expert.

**2.2 Points et Durée**

* Chaque question vaut 3 points.
* La durée pour répondre à une question est de 2 minutes.

**2.3 Variété de Questions**

Les questions varient entre 11 et 23 à chaque tour de jeu et couvrent divers aspects des cultures camerounaises et africaines.

**2.4 Fonctionnalités Sociales**

* Partage du score via les réseaux sociaux (Facebook, Twitter) avec un lien d'invitation pour télécharger l'application.
* Fonction d'invitation d'amis à jouer via Mboa Store.

**2.5 Multilinguisme**

L'application sera disponible en Français et en Anglais, avec une priorité donnée à l'anglais.

**2.6 Mode Hors Ligne**

L'application sera fonctionnelle avec ou sans connexion Internet.

**2.7 Classement des Joueurs**

* Affichage du classement des joueurs basé sur le cumul de points.
* Les joueurs peuvent voir leur position par rapport aux amis.

**2.8 Proposition de Questions**

* Les utilisateurs peuvent soumettre leurs questions.
* Un comité de modération examinera et approuvera les questions proposées.

**3. Développement Technique**

**3.1 Technologies Utilisées**

* Langages : Ionic (Frontend), Node.js (Backend)
* Base de données: SQLite
* Serveur Web : Express.js
* APIs : [Liste des APIs fournie dans le document d'APIs]

**3.2 Architecture**

* Modèle client-serveur avec une base de données centralisée.
* Utilisation de RESTful APIs pour la communication entre le frontend et le backend.

**3.3 Sécurité**

* Utilisation d'un protocole HTTPS pour toutes les communications.
* Cryptage des données utilisateur.
* Gestion stricte des autorisations d'accès à la base de données.

**3.4 Tests**

* Tests unitaires et d'intégration réguliers pendant le développement.
* Tests de charge avant le déploiement pour garantir la stabilité.

**4. Calendrier de Développement**

### 4.1 Phase de Conception (Semaine 1-2)

* Conception de l'interface utilisateur.
* Conception de la base de données.

### 4.2 Phase de Développement (Semaine 3-6)

* Développement frontend et backend.
* Intégration des fonctionnalités principales.

### 4.3 Phase de Tests (Semaine 7-8)

* Tests unitaires et d'intégration.
* Tests de charge.

### 4.4 Phase de Déploiement (Semaine 9)

* Déploiement de l'application sur les plateformes (iOS, Android).

**5. Support Client**

### 5.1 Section d'Assistance Client

Une section d'assistance client sera intégrée pour gérer les commentaires et résoudre les problèmes des utilisateurs.

* Mise en place d'une section d'assistance client.
* Intégration d'un système de suivi des problèmes.

# II. Conception Détaillé - Mboa Culture Quizz

## 1. Architecture de l'Application

### 1.1 Architecture Globale

L'architecture client-serveur est choisie pour assurer la séparation des préoccupations. Le client mobile est développé en Ionic pour permettre une expérience utilisateur fluide sur iOS et Android. Le serveur, construit en Node.js, gère les requêtes du client, communique avec la base de données SQLite, et fournit des données sous forme d'API RESTful.

### 1.2 Architecture Client

* **Écran d'Accueil :**
  + Options pour commencer un quizz, inviter des amis, et accéder aux paramètres.
* **Interface de Quizz :**
  + Catégories et niveaux sélectionnables.
  + Affichage des questions, options de réponse, et une minuterie.
* **Écran de Résultats :**
  + Score du joueur, réponses correctes et incorrectes.
  + Options pour rejouer, partager sur les réseaux sociaux, et revenir à l'écran d'accueil.

### 1.3 Architecture Serveur

* Gestion des requêtes du client.
* Logique du jeu, traitement des réponses, calcul des scores.
* Interaction avec la base de données SQLite.

## 2. Interface Utilisateur

### 2.1 Wireframes

* **Écran d'Accueil :**
  + Boutons pour commencer le quizz, inviter des amis, et accéder aux paramètres.
* **Interface de Quizz :**
  + Présentation des questions, options de réponse, et minuterie.
* **Écran de Résultats :**
  + Affichage du score, des réponses correctes et incorrectes.

### 2.2 Conception Visuelle

* Couleurs vives représentant la diversité culturelle.
* Polices lisible et attrayante.
* Utilisation d'images représentatives pour chaque catégorie de quizz.

## 3. Base de Données

### 3.1 Modèle de Données

* **Questions :**
  + Texte de la question, options de réponse, réponse correcte, catégorie, niveau.
* **Utilisateurs :**
  + Identifiant, nom d'utilisateur, mot de passe crypté, score.

### 3.2 Indexation et Optimisation

* Indexation sur les catégories, niveaux et utilisateurs pour améliorer les performances.

## 4. Fonctionnalités Principales

### 4.1 Cas d'Utilisation

* **Commencer un Quizz :**
  + L'utilisateur sélectionne une catégorie et un niveau.
* **Inviter des Amis :**
  + L'utilisateur envoie des invitations à ses contacts.
* **Afficher les Résultats :**
  + L'utilisateur voit son score, les réponses correctes et incorrectes.

### 4.2 Diagrammes de Séquence

* **Système de Points :**
  + Interaction entre l'utilisateur, le frontend, et le backend pour attribuer des points.
* **Invitation d'Amis :**
  + Flux d'interaction entre l'utilisateur, le frontend, le backend, et l'ami invité.
* **Authentification :**
  + Processus d'authentification entre le frontend, le backend et la base de données.

## 5. Sécurité

### 5.1 Authentification et Autorisation

* Utilisation de JWT pour sécuriser les communications entre le client et le serveur.
* Vérification des permissions avant d'accéder à certaines fonctionnalités.

### 5.2 Cryptage des Données

* Cryptage des informations sensibles (mot de passe, jetons) pour protéger la confidentialité.

## 6. Tests

### 6.1 Tests Unitaires et d'Intégration

* Suite de tests pour chaque fonctionnalité individuelle.
* Tests d'intégration pour garantir la cohérence entre les composants.

### 6.2 Tests de Charge

* Simulation d'un grand nombre d'utilisateurs pour évaluer la performance globale.

## Introduction

## Une conception détaillée d'une application comme Mboa Culture implique plusieurs aspects, y compris l'architecture, l'interface utilisateur, la base de données, les fonctionnalités principales, la sécurité, et plus encore.

### 2.1 Objectif

L'objectif de ce document est de fournir une vision détaillée de la conception de l'application mobile "Mboa Culture Quizz".

### 2.2 Portée

Ce document couvre l'architecture, l'interface utilisateur, la base de données, les fonctionnalités principales, et est accompagné de diagrammes pour une compréhension visuelle.

## 3. Architecture

### 3.1 Diagramme d'Architecture

#### Explication du Diagramme

* Le frontend, développé en Ionic , communique avec le backend construit en Node.js via des API RESTful.
* La base de données SQLite stocke les questions, réponses, scores des joueurs, etc.

## 4. Interface Utilisateur

### 4.1 Wireframes

#### Écran d'Accueil

#### Écran de Quizz

#### Écran de Résultats

### 4.2 Navigation

#### Diagramme de Navigation

## 5. Base de Données

### 5.1 Modèle de Données

#### Modèle Entity-Relationship (ER)

## 6. Fonctionnalités Principales

### 6.1 Diagramme de Cas d'Utilisation

### 56.2 Diagramme de Séquence - Système de Points

#### Description du Diagramme :

1. L'utilisateur sélectionne une réponse.
2. Le frontend envoie la réponse au backend.
3. Le backend vérifie la réponse et attribue des points.
4. Le backend met à jour le score du joueur.
5. Le backend envoie la réponse et les points au frontend.
6. Le frontend met à jour l'interface utilisateur avec les résultats.

### 5.3 Diagramme de Séquence - Invitation d'Amis

#### Description du Diagramme :

1. L'utilisateur sélectionne l'option d'invitation d'amis.
2. Le frontend affiche la liste d'amis de l'utilisateur.
3. L'utilisateur sélectionne un ami à inviter.
4. Le frontend envoie une invitation au backend.
5. Le backend envoie une notification à l'ami.
6. L'ami reçoit la notification et accède au quizz.
7. Le backend traite l'invitation et met à jour les données.

## 7. Sécurité

### 7.1 Diagramme de Séquence – Authentification

#### Description du Diagramme :

1. L'utilisateur saisit ses informations de connexion.
2. Le frontend envoie les informations au backend.
3. Le backend vérifie les informations dans la base de données.
4. Le backend génère un jeton d'authentification.
5. Le backend envoie le jeton au frontend.
6. Le frontend stocke le jeton et autorise l'accès.

### 7.2 Diagramme de Séquence – Cryptage

#### Description du Diagramme :

1. Le frontend prépare les données à envoyer.
2. Le frontend utilise un algorithme de cryptage pour chiffrer les données.
3. Les données cryptées sont envoyées au backend.
4. Le backend utilise une clé privée pour décrypter les données.
5. Le backend traite les données décryptées.

## 8. Tests

### 8.1 Diagramme de Séquence - Tests Unitaires

#### Description du Diagramme :

1. Le système de test envoie une requête de test au backend.
2. Le backend effectue les tests unitaires.
3. Le backend génère des rapports de test.
4. Les rapports de test sont renvoyés au système de test.
5. Le système de test analyse les résultats.

### 8.2 Diagramme de Séquence - Tests d'Intégration

#### Description du Diagramme :

1. Le système de test envoie une requête de test d'intégration au backend.
2. Le backend coordonne les différents composants pour le test d'intégration.
3. Les composants sont testés simultanément.
4. Les résultats du test d'intégration sont renvoyés au système de test.

### 8.3 Diagramme de Séquence - Tests de Charge

#### Description du Diagramme :

1. Le système de test envoie une requête pour simuler une charge lourde au backend.
2. Le backend reçoit la demande et commence à traiter un grand nombre de requêtes.
3. Le système de test surveille les performances du backend.
4. Le système de test génère des rapports sur la capacité et les temps de réponse.

## 9. Calendrier de Développement

### 9.1 Diagramme de Gantt

## Défis Passionnants :

### 1. Diversité des Questions

Le défi de maintenir une base de données riche en questions diverses, reflétant la grande variété des cultures camerounaises et africaines. Cela nécessite une recherche continue et une collaboration avec des experts culturels.

### 2. Engagement Continu

Encourager l'engagement continu des utilisateurs en proposant régulièrement de nouvelles questions, catégories et mises à jour du jeu. L'objectif est de créer une communauté active et fidèle autour de Mboa Culture.

### 3. Gestion des Catégories

Équilibrer la difficulté des questions dans chaque catégorie pour s'assurer que le quizz reste accessible aux utilisateurs de tous niveaux tout en offrant un défi aux joueurs expérimentés.

### 4. Sécurité et Confidentialité

Assurer la sécurité des données des utilisateurs, en particulier en ce qui concerne les informations d'authentification et les scores. La confidentialité des utilisateurs doit être une priorité absolue.

### 5. Promotion de la Culture Africaine

Un défi constant est de présenter la richesse et la diversité des cultures africaines de manière précise et respectueuse. Cela implique une sensibilisation culturelle et un dialogue continu avec des experts et des communautés concernées.

### 6. Accessibilité

Garantir que l'application reste accessible à un large public en optimisant son utilisation, son interface utilisateur et son contenu pour différentes régions et groupes démographiques.

En relevant ces défis, Mboa Culture vise à devenir une plateforme incontournable pour l'exploration et la préservation des cultures camerounaises et africaines à travers une expérience ludique et éducative.

Lien du projet sur github : https://github.com/Tagouffo21/dev-mobile