**🧙‍♂️** Défi 16 — Tu es un Sorcier Harry ! – Dockerwarts Edition

**🎯 Objectif du défi**

Concevoir une **application mobile de QCM** 100 % générée par IA, inspirée du monde de **Poudlard** et de **l’informatique moderne**, afin d’offrir une expérience ludique et éducative mêlant **magie et code**.

L’application devait :

* Poser **20 questions aléatoires** sur un thème original ;
* Calculer le score final automatiquement ;
* Afficher un **message personnalisé** selon le niveau du joueur ;
* Fonctionner **sans serveur**, de façon entièrement locale.

Notre équipe a choisi de créer une version inédite :

🪄 *Dockerwarts – l’école des Développeurs Sorciers !*  
Un mélange d’humour, de magie et de technologie, où les sorts se codent en Python et les potions se compilent dans Docker.

**🧩 Présentation générale de l’application**

**Nom du projet**

**Tu es un Sorcier Harry ! – Dockerwarts Edition**

**Type d’application**

Application mobile multiplateforme (Android / iOS)  
Réalisée avec **React Native (Expo)** et un fichier **JSON local** pour stocker les questions.

**Description**

L’utilisateur entre dans le rôle d’un **développeur-sorcier de Poudlard Tech**, et répond à 20 questions aléatoires sur le thème du code magique :

* Langages de programmation ensorcelés,
* Commandes UNIX revisitées,
* Sécurité des sorts numériques,
* Bugs démoniaques,
* Protocoles mystiques du Web magique.

Chaque partie se conclut par un **score** et un **titre de sorcier** :

* 🧙‍♂️ Maître Sorcier (18 à 20 points)
* ✨ Sorcier confirmé (13 à 17 points)
* 🔮 Apprenti prometteur (8 à 12 points)
* 🐍 Moldu curieux (0 à 7 points)

**💻 Architecture technique**

**📁 Structure du projet**

tu\_es\_un\_sorcier\_harry/

│

├── mobile/

│ ├── App.js → logique du QCM + interface

│ ├── questions.json → base de 20 questions locales

│ ├── package.json → dépendances React Native (Expo)

│ └── README.md → guide d’installation

│

└── Rapport\_QCM.docx

**🧠 Fonctionnement**

* L’application charge **les questions depuis un fichier JSON local** ;
* Les questions sont affichées **une par une**, dans un ordre fixe ou aléatoire ;
* À chaque réponse, le score est mis à jour instantanément ;
* À la fin du quiz, une **page de résultat** affiche le niveau du joueur avec une touche d’humour.

**⚙️ Technologies utilisées**

* **React Native (Expo)** pour le développement mobile multiplateforme
* **JavaScript ES6** pour la logique du quiz
* **JSON local** pour les données des questions
* **StyleSheet API** pour le design en mode parchemin doré

**🎨 Design et expérience utilisateur**

Le design s’inspire directement de l’univers magique de Poudlard :

* **Fond beige clair** pour un effet parchemin 📜
* **Boutons dorés et ombrés** rappelant les armoiries de l’école 🦁
* **Typographie sombre et lisible** pour une ambiance de grimoire numérique
* **Icônes emoji** pour accentuer le côté ludique

Les utilisateurs n’ont besoin d’aucune connexion Internet : tout fonctionne localement.

**🚀 Lancement du projet**

**Installation**

npm install -g expo-cli

cd mobile

npm install

npx expo start

**Exécution**

* Scanner le **QR code** généré par Expo avec l’application **Expo Go** sur smartphone
* Le QCM démarre automatiquement

**Sans serveur**

Toutes les données sont locales : aucune connexion ou API externe n’est utilisée.

**📊 Résultats et tests**

* Testé sur Android (Expo Go) et navigateur web (via Expo Web)
* Temps moyen d’une partie : **2 à 3 minutes**
* 100 % des fonctionnalités requises validées :
  + Interface fluide
  + Score correct
  + Résultat final affiché
  + Aucune dépendance réseau

**🪄 Conclusion**

Ce défi illustre parfaitement l’esprit du Workshop IA :

*Allier créativité, technique et humour à travers des outils automatisés.*

Le projet **Dockerwarts Edition** démontre qu’il est possible, en moins d’une journée, de générer :

* une application mobile complète,
* un contenu ludique et thématique,
* une interface fonctionnelle,
* un rapport structuré et cohérent — le tout avec l’assistance de l’IA.

**🔮 Prochaines améliorations possibles :**

* Génération automatique de nouvelles questions via GPT
* Classement des joueurs par maison (Gryffondor, Serpentard, etc.)
* Mode multijoueur “Duel de sorciers”
* Intégration d’une API IA pour commenter le score du joueur en direct

**💬 Citation finale**

“Les vrais développeurs ne meurent jamais…  
ils respawnent dans un nouveau build magique.”  
— *Albus FireWallDore, cours de DevOps Sorcier, Dockerwarts 2025*