C’est un projet intéressant ! Pour créer une application web qui répond à tes besoins, je te propose d’utiliser les technologies suivantes :

**Stack proposée**

* **Frontend** : HTML, CSS, JavaScript
* **Backend** : Node.js avec Express
* **Base de données** : MongoDB (pour stocker les questions et les résultats)

**Étapes du projet**

1. **Mise en place de l’environnement** :
   * Installe Node.js et MongoDB.
   * Utilise un éditeur de code comme Visual Studio Code.
2. **Création du backend** :
   * Initialise un projet Node.js.
   * Installe Express pour gérer les requêtes HTTP.
   * Crée une API pour gérer les questions et les résultats.
3. **Gestion des questions** :
   * Crée une collection de questions dans MongoDB.
   * Crée des endpoints pour récupérer les questions de manière aléatoire.
4. **Calcul des scores** :
   * Implémente la logique pour les deux barèmes :
     + Pour le barème de réponse unique : si la réponse est correcte, +1 point, sinon 0.
     + Pour le barème de pourcentage : calcule le score avec la formule fournie.
5. **Frontend** :
   * Utilise HTML/CSS pour l’interface utilisateur.
   * Utilise JavaScript pour gérer l’interaction (afficher les questions, collecter les réponses, afficher les résultats).
6. **Mise en ligne** :
   * Utilise des services comme Heroku ou Vercel pour déployer ton application.

**Exemple de code**

**Backend (Node.js + Express)** :

**const express = require('express');**

**const mongoose = require('mongoose');**

**const app = express();**

**app.use(express.json());**

**// Connecter à MongoDB**

**mongoose.connect('mongodb://localhost:27017/qcu', { useNewUrlParser: true, useUnifiedTopology: true });**

**// Schéma de la question**

**const questionSchema = new mongoose.Schema({**

**question: String,**

**options: [String],**

**correct: Number**

**});**

**const Question = mongoose.model('Question', questionSchema);**

**// Endpoint pour récupérer des questions aléatoires**

**app.get('/questions', async (req, res) => {**

**const questions = await Question.aggregate([{ $sample: { size: 10 } }]); // Prendre 10 questions au hasard**

**res.json(questions);**

**});**

**// Lancer le serveur**

**app.listen(3000, () => {**

**console.log('Server is running on http://localhost:3000');**

**});**

**Frontend (html + javascript) :**

**<!DOCTYPE html>**

**<html lang="fr">**

**<head>**

**<meta charset="UTF-8">**

**<title>QCU</title>**

**</head>**

**<body>**

**<div id="quiz"></div>**

**<button onclick="submitQuiz()">Soumettre</button>**

**<script>**

**async function loadQuestions() {**

**const response = await fetch('/questions');**

**const questions = await response.json();**

**const quizDiv = document.getElementById('quiz');**

**questions.forEach((q, index) => {**

**quizDiv.innerHTML += `<div>**

**<p>${q.question}</p>**

**${q.options.map((option, i) => `<label><input type="radio" name="q${index}" value="${i}">${option}</label><br>`).join('')}**

**</div>`;**

**});**

**}**

**function submitQuiz() {**

**// Logique pour calculer les scores**

**}**

**loadQuestions();**

**</script>**

**</body>**

**</html>**

javascript

Salut !

Comment ca va ?

Je me tate à me lancer dans un « petit » projet... mais qui necessite de coder une interface web (rien que ca !)

Le principe est assez simple, et donc le code devrait l’etre aussi... j’imagine ? J’aurais bien voulu ton avis avant de telecharger plein de trucs pour environnement et tout. Sachant que là, si je fais ca seul, comme je ne sais pas du tout coder, ca va etre du code by GPT-4.

Qu’est-ce qui pourrait mal se passer ??? (Mais qui aurait pu prédire ?)

Il s’agirait d’avoir un questionnaire QCU en ligne, avec des questions réparties aléatoirement, et chacune posée deux fois. Deux fois, parce que l’objectif est d’évaluer 2 types de barèmes qui se répondent différement.

D’une part, un bareme QCU classique. Bonne réponse => 1 point, le reste 0

D’autre part, un bareme qui se répond en attribuant un % à chaque réponse. Genre, si j’ai éliminé 2 des 4 propositions mais que je suis incapable de départages les 2 restantes, j’attribue 0%, 0%, 50% et 50%. Ce faisant je m’assure 0.5 points. (il s’agirait d’un système dit « quadratique » ou le score est égal à 1- le carré des erreurs. J’ai pas inventé le système, tu te doutes ! Par contre à ma connaissance il n’a pas été testé rigoureusement en situation réelle). Ca encourage à etre honnete et ca permet d’évaluer le degré de certitude qu’ont les répondants. Si ils ne savent pas, ils ont interet à le dire (en mettant 25% à chaque proposition)

En bonus, si on pouvait chronométrer le questionnaire et chaque question ca serait bien !

(pour voir si il y a une corrélation entre « sur de moi dans mes réponses » et « rapide à répondre à la question », et pour attester si certaines questions sont + dures que d’autres, si tout le monde met + de temps à y répondre)

Avant le début du questionnaire il faudrait un petit texte d’explication, et puis poser 2-3 questions d’essai avec le corrigé et le score attribué, pour que les gens comprennent le 2e type de bareme.

Et à la fin du questionnaire, peut-etre une zone commentaire libre, et/ou un bref sondage de satisfaction...

Dans l’idée, une fois la plateforme mise en place, je ferais remplir des questionnaires (axé sur l’anesthésie ped, mais aussi avec des questions de culture G pour faire office de contrôle) aux internes qui arrivent en debut de semestre, puis quand ils partent.

(dans le questionnaire au depart des internes, je pense qu’il faut inclure aussi les questions qu’ils ont eu au début du semestre, pour pouvoir mieux comparer... bref c’est du detail)

Comme ca on pourrait voir

* + les progres des internes en 1 semestre d’anesth ped (tant sur le nombre de bonnes réponses que dans la certitude qu’ils affichent)
  + les différences dans les scores et dans les facons de remplir entre les 2 modalités de QCU
  + potentiellement voir s’il y a une habituation au nouveau système de notation entre les 2 sessions
  + si je fais ca sur suffisament de fournées d’internes (on a environ 20 internes par semestre, donc sur 2 ans ca fait ~80 personnes, c’est pas mal ! Et si ca marche bien, on pourrait peut-etre meme envisager de le faire sur plusieurs hopitaux à la fois, augmentant le nombre de réponses et l’anonymisation), on pourrait peut-etre croiser avec des données demographiques type age, sexe, semestre d’internat... (est ce que les vieux internes sont meilleurs que les plus jeunes ? ou juste plus surs d’eux ? ou au contraire plus prudent ? Ou encore, pour taper dans les clichés, est-ce que les hommes sont plus surs d’eux que les femmes ?...)

Bref, j’aurais bien voulu avoir ton avis sur la faisabilité de la mise en place d’un site web sur lequel on répondrait à ces QCU, et qui genererait un excel avec les réponses et temps des participants.

Est-ce que c’est « trop simple je te fais ca en 2-3 heures ! », ou « ouh la la mais en fait c’est assez galère car [insert here charabia technique] »

Et question bonus, est-ce que tu penses qu’en m’alliant à GPT4, j’ai une chance d’y arriver by myself ?