

ISEN

Générateur de Quiz

SIMPLON

.co



Malgorzata RYCZER-DUMAS

Mathieu LARONCE

Khawla MILI



Sommaire

1- Contexte et Objectifs

2- Organisation et Choix Technique

3- Architectures

4- Démonstration

5- Conclusion



-1-
Introduction :
Contexte et Objectifs





Contexte

Le projet Miskatonic Quiz a pour objectif était de concevoir une application web pour l'université *Miskatonic*.

Objectifs:

Permettre aux enseignants:

- Créer et gérer des quiz en ligne
- Importer ou ajouter des questions
- Faire passer les quiz aux étudiants

Contraintes:

- Authentification sécurisé
- Connexion via une API
- Hashage des mots de passe
- Deux BDD
- ETL pour le chargement automatique



-2-
**Organisation
et
Choix Technique**




Exemple US

Id	Description (User Story)	Fonctionnalité principale	Critères d'acceptation (ICS)
MQ-1	En tant qu'architecte de données, je veux créer un MCD MongoDB clair pour représenter Question, Réponse et Quiz afin de préparer correctement la base de données.	BDD	- Le MCD doit inclure toutes les entités listées.- Chaque entité doit avoir un identifiant unique. - Les relations (ex : un utilisateur → plusieurs quiz, un quiz → plusieurs questions) doivent être clairement représentées.

Backlog produit

T	Clé	Résumé	Personne assignée	Rapporteur	Pr	État	Résolution	Création	Mise à jour
❖	MQ-19	DOC	Non assigné	Maigorzata Ryczer-Dumas	=	À FAIRE	Non résolu	10/sept/25	10/sept/25
📌	MQ-18	Présentation orale	Non assigné	khaoula MILI	=	À FAIRE	Non résolu	10/sept/25	10/sept/25
📌	MQ-17	Rédiger un README complet	Non assigné	khaoula MILI	=	À FAIRE	Non résolu	10/sept/25	10/sept/25
📌	MQ-16	Générer un quiz aléatoire via l'IHM	Non assigné	khaoula MILI	⚠	À FAIRE	Non résolu	10/sept/25	10/sept/25
📌	MQ-15	Créer une question via un formulaire dans l'IHM	Non assigné	khaoula MILI	⚠	À FAIRE	Non résolu	10/sept/25	10/sept/25
📌	MQ-14	Visualiser toutes les questions dans l'IHM	Non assigné	khaoula MILI	⚠	À FAIRE	Non résolu	10/sept/25	10/sept/25
❖	MQ-13	WEB	Non assigné	khaoula MILI	=	À FAIRE	Non résolu	10/sept/25	10/sept/25
📌	MQ-12	Organiser le code API en modules	Non assigné	khaoula MILI	⚠	À FAIRE	Non résolu	10/sept/25	10/sept/25
❖	MQ-11	API	Non assigné	Maigorzata Ryczer-Dumas	=	À FAIRE	Non résolu	10/sept/25	10/sept/25

Backlog Sprint

Rechercher dans le...  Epic ▾

☐ **Tableau Sprint 1** 11 sept. – 16 sept. (1 ticket) 0 0 2 Démarrer un sprint ...

☐ MQ-1 ETL_Questions BDD TERMINÉ(E) 2 ⚠ 👤 ...

+ Créer

⇅

1 ticket | Estimation : 2

☐ **Tableau Sprint 2** 17 sept. – 22 sept. (5 tickets) 0 0 0 Démarrer un sprint ...

☐ MQ-8 Documentation OpenAPI API TERMINÉ(E) - ⚠ 👤

☐ MQ-9 Requête API questions API TERMINÉ(E) - ⚠ 👤

☐ MQ-10 Générer un quiz aléatoire via l'API API TERMINÉ(E) - ⚠ 👤

☐ MQ-12 Organiser le code API en modules API TERMINÉ(E) - ⚠ 👤

☐ MQ-21 Authentification sécurisée avec permissions pour protéger les données s... API TERMINÉ(E) - ⚠ 👤



Github : <https://github.com/go2375/Miskatonic-Quiz>



Choix Techniques

API



Backend, add user,
gestion question et quiz,
doc OpenApi

Frontend



Interface web, affichage
des quiz

BDD Questions



Stocke questions quiz en
JSON

BDD Utilisateurs



Stock user, roles

Sécurité



Sécurisé mdp hashage

Docker



Conteneurise l'api et la
base mongo • • • • •

-3- Architectures



Architecture hybride de nos collections principales :

Collection *Questions*

Questions sont stockées pour une seule fois et référencées ultérieurement.

Origine : proviennent du CSV ou sont ajoutées par un enseignant.

Collection *Questionnaires*

Les quiz contiennent une **copie embeddée** des questions sélectionnées depuis la collection *Questions*.

```
quiz = QUESTIONNAIRES_COLLECTION.find_one({"_id": ObjectId(quiz_id)})
```

Performance

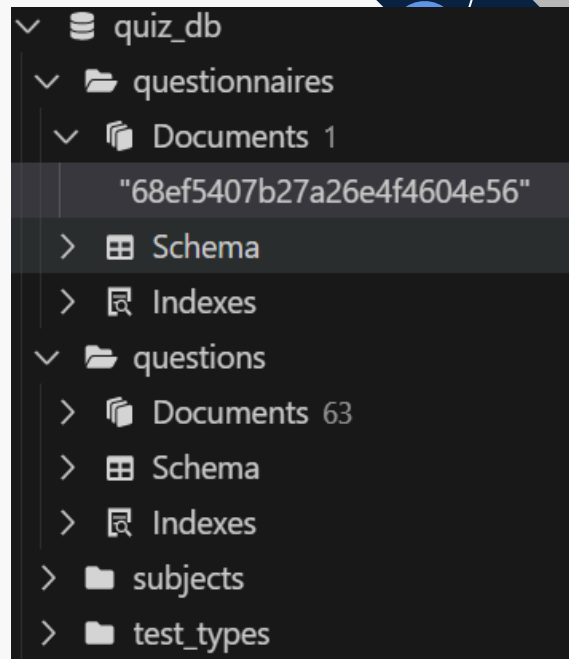
Optimisation de la lecture avec une seule requête MongoDB pour récupérer un quiz complet

Traçabilité

Les quiz créés peuvent être partagés avec les élèves.

Maintenance

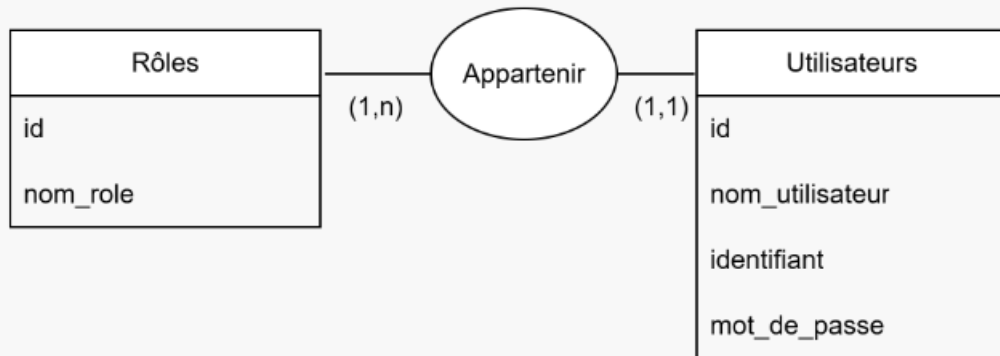
Les questions restent centralisées et réutilisables pour d'autres quiz.



Une génération de quiz filtrée par sujet et type.

Architecture de SQLite

MCD



MLD

Roles : role_id, role

Utilisateurs : utilisateur_id, nom_utilisateur, identifiant, mot_de_passe, role_id#

Sécurité et authentification optimisées

Mots de passe hachés, rôles clairement définis pour contrôler l'accès.

Cohérence avec l'API

Endpoints centralisent toutes les opérations liées aux utilisateurs.

Performance et réactivité de l'API lors de la connexion

Accès léger et efficace aux comptes pour l'authentification.

Séparation claire des responsabilités Utilisateurs gérés dans SQLite, questions/quizz dans MongoDB.

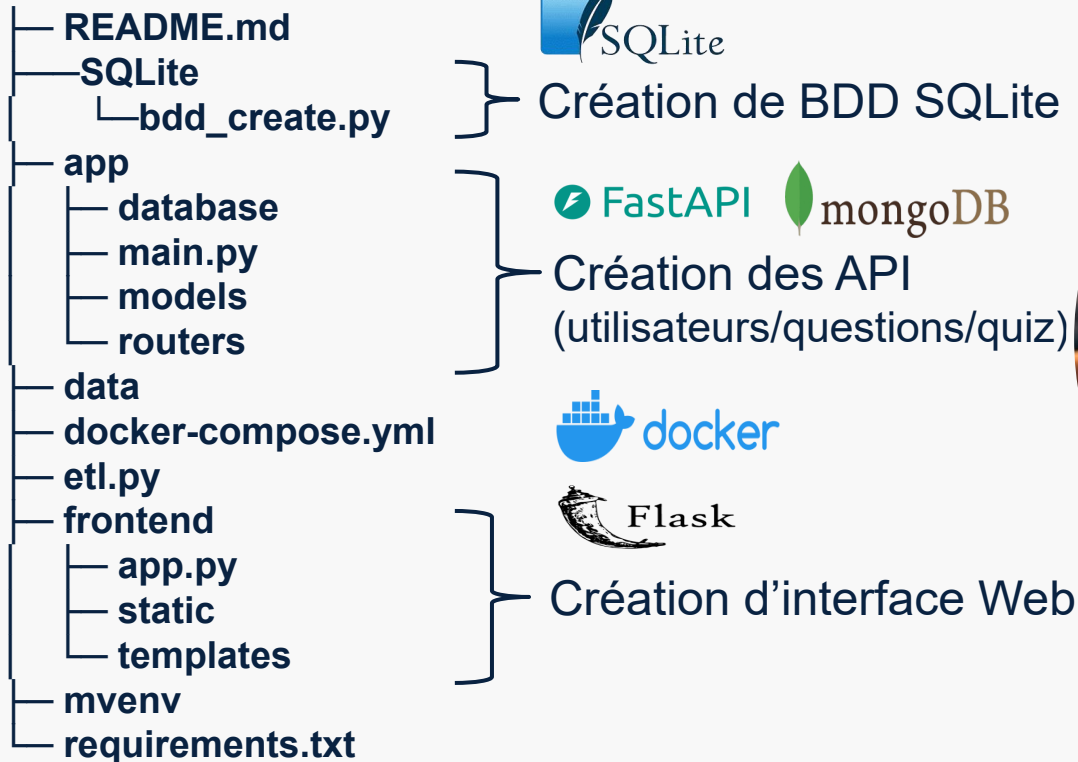
Maintenance simplifiée

Modifications locales dans SQLite sans impacter MongoDB et sans risque pour les données quizz.

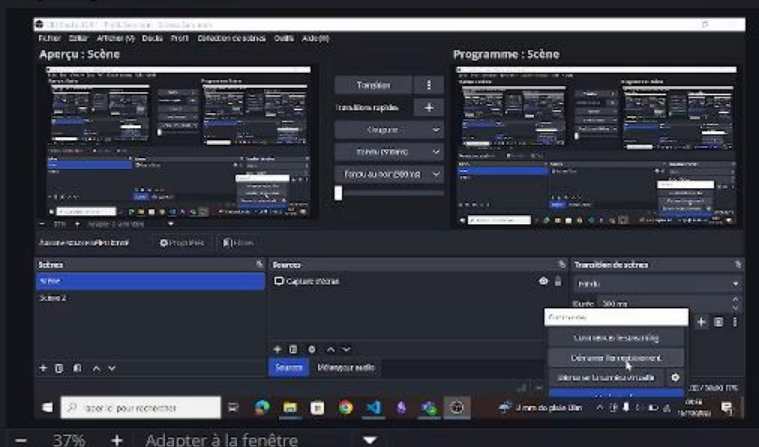
-4- Démonstration



Structure de code



Aperçu : Scène



- 37% + Adapter à la fenêtre

Aucune source sélectionnée

Propriétés

Filtres

Scènes

Scène

Scène 2

+ - [icon] [icon]

Sources

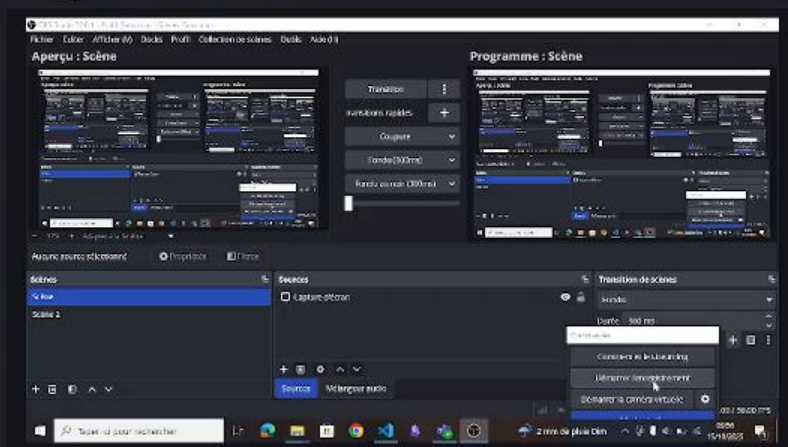
Capture d'écran

+ - [icon] [icon]

Sources

Mélangeur audio

Programme : Scène



Transition

Transitions rapides

Coupure

Fondu (300ms)

Fondu au noir (300ms)

Transition de scènes

Fondu

Durée 300 ms

Commandes

Commencer le streaming

Arrêter l'enregistrement

Démarrer la caméra virtuelle

.00 / 30.00 FPS

-5-

Conclusions

- Solution innovante favorisant l'autonomie et l'efficacité de l'apprentissage.
- Répond aux attentes des enseignants et des étudiants.
- Prépare le terrain pour les futures évolutions d'éducation numérique.



Merci !

Avez-vous des questions ?

CREDITS: This presentation template was created by [Slidesgo](#), and includes icons by [Flaticon](#), and infographics & images by [Freepik](#)

