



Université Ibn Zohr
ÉCOLE NATIONALE SUPÉRIEURE DE
L'INTELLIGENCE ARTIFICIELLE ET SCIENCE DE
DONNÉES - TAROUDANT
Filière : Ingénierie Logicielle

Rapport de Mini Projet JAVA

Conception et développement d'un jeu de quiz en temps réel

Réalisé par :
Mohamed El aassal
Mohamed EL Guarir
Mohamed El Haddati
Mahmoud Chaoubi

Encadré par :
Mme. Loubna Karbil

ANNÉE UNIVERSITAIRE : 2024/2025

Table des matières

Résumé	3
1 Introduction générale	4
2 Analyse des Besoins	4
2.1 Les Besoins fonctionnels	4
2.1.1 Authentification sécurisée :	4
2.1.2 Mode solo :	4
2.1.3 Mode multijoueur :	4
2.1.4 Leaderboard :	4
2.1.5 Historique des résultats :	4
2.2 les Besoins non fonctionnels	5
2.2.1 Sécurité :	5
2.2.2 Une bonne expérience utilisateur :	5
2.2.3 Protection des données personnelles :	5
3 Conception	5
3.1 Conception-UML	5
3.1.1 Introduction :	5
3.1.2 Diagramme de classes	6
3.1.3 Conclusion	7
3.2 Conception Front-end	8
3.2.1 Introduction :	8
3.2.2 Maquettes et Prototypes des Interfaces	8
3.2.3 Conclusion :	10
4 Réalisation	11
4.1 Démarche	11
4.1.1 MVC : Modèle-Vue-Contrôleur	11
4.2 Outils et environnement Techniques	12
4.3 Capture d'écrans des interface du Jeu :	15
4.3.1 Authentification :	15
4.3.2 page d'accueil :	17
4.3.3 Jouer seul :	18
4.3.4 Multi-joueurs (temps réel) :	21
4.4 Perspectives :	23
4.5 Conclusion :	23
5 Conclusion générale	24
Références bibliographiques	25
Les liens utiles	25

Table des figures

1	Diagramme de classes partie1	6
2	Diagramme de classes partie2	7
3	interface 1	8
4	interface 2	9
5	interface 3	9
6	interface 4	10
7	interface 5	10
8	MVC	11
9	JAVA	12
10	IntelliJ IDEA	12
11	Scene Builder	12
12	Excalidraw	13
13	MySQL Logo	13
14	Nginx Logo	13
15	git Logo	14
16	github logo	14
17	Docker Logo	14
18	page du login	15
19	La connexion a échoué	15
20	page du registre	16
21	ajout d'un utilisateur	16
22	ajout d'un utilisateur avec succes	17
23	Login avec succes	17
24	page d'accueil	18
25	crée un quiz	18
26	jouer seul 1	19
27	jouer seul 2	19
28	le bouton finish	20
29	le bouton show results	20
30	résultats du quiz	20
31	choix du type du quiz	21
32	configurer les parametres du quiz	21
33	attente 1	22
34	attente 2	22
35	quiz multi-joueurs	22
36	progress dans le quiz	23

Résumé

Ce projet consiste en la conception et le développement d'un jeu de quiz en temps réel en Java, intégrant plusieurs technologies et concepts avancés. Le système repose sur l'utilisation des sockets pour la communication réseau entre les clients et le serveur, et JDBC pour la gestion des interactions avec la base de données. La conception suit une approche modulaire, mettant en œuvre des design patterns tels que MVC pour séparer les préoccupations, DAO pour l'accès aux données, et Singleton pour garantir une instance unique des ressources critiques. L'application gère également le multitâche grâce à l'utilisation des threads, et exploite les entrées/sorties (I/O) pour assurer une gestion fluide des données entre les différentes couches du système. L'objectif principal est de fournir une expérience de jeu interactive et performante, tout en mettant en pratique des concepts clés de développement logiciel.

1 Introduction générale

Dans le cadre de notre projet d'équipe, nous développons un jeu de quiz en temps réel en utilisant Java. Ce projet met en œuvre des échanges rapides et synchronisés grâce à un système de communication réseau basé sur les sockets. La gestion des données est assurée via JDBC, permettant une interaction efficace avec la base de données. En tant qu'équipe, nous avons conçu une architecture robuste reposant sur les design patterns MVC, DAO et Singleton pour garantir une organisation modulaire et maintenable. Les threads sont utilisés pour gérer le multitâche et assurer la fluidité des interactions entre les joueurs, tandis que les mécanismes d'I/O permettent d'optimiser les échanges de données. Ce projet est une véritable collaboration, combinant les efforts de chacun pour atteindre les objectifs d'un jeu en temps réel interactif et performant.

2 Analyse des Besoins

2.1 Les Besoins fonctionnels

2.1.1 Authentification sécurisée :

L'application permet aux utilisateurs de s'inscrire et de se connecter de manière sécurisée grâce à un système de gestion des identifiants et mots de passe. Ce mécanisme garantit la confidentialité et l'intégrité des données des utilisateurs.

2.1.2 Mode solo :

Le mode solo permet aux joueurs de participer à des quiz personnalisés adaptés à leur niveau et leurs préférences, favorisant un apprentissage et un divertissement individuel.

2.1.3 Mode multijoueur :

L'application offre la possibilité de jouer en mode multijoueur, avec des parties comprenant 2, 3 ou 4 joueurs. Ce mode encourage l'interactivité et la compétition entre amis ou autres utilisateurs.

2.1.4 Leaderboard :

Un tableau des scores est intégré pour classer les meilleurs joueurs en fonction de leurs performances. Cette fonctionnalité motive les participants à améliorer leurs scores et à viser le sommet du classement.

2.1.5 Historique des résultats :

L'application conserve un historique des résultats des matchs multijoueurs, permettant aux utilisateurs de suivre leurs performances et d'évaluer leur progression au fil du temps.

2.2 les Besoins non fonctionnels

2.2.1 Sécurité :

L'application garantit un haut niveau de sécurité en intégrant des mécanismes de chiffrement pour la gestion des identifiants et des mots de passe, ainsi que des mesures de protection contre les accès non autorisés et les attaques potentielles.

2.2.2 Une bonne expérience utilisateur :

Une attention particulière a été portée à la conception de l'interface utilisateur, en la rendant intuitive, fluide et ergonomique, afin d'assurer une navigation facile et agréable pour tous les utilisateurs.

2.2.3 Protection des données personnelles :

L'application respecte les normes de protection des données en veillant à ce que les informations personnelles des utilisateurs soient stockées de manière sécurisée et utilisées uniquement dans le cadre des fonctionnalités prévues, conformément aux réglementations en vigueur.

3 Conception

3.1 Conception-UML

3.1.1 Introduction :

La conception d'un système d'information représente une étape cruciale dans tout projet informatique. Elle s'agit non seulement de définir la structure, les fonctionnalités, mais également de préciser les interactions complexes entre les différents composants du système. Pour assurer le succès de cette étape, il est impératif d'adopter une méthode de conception rigoureuse et bien établie. C'est dans ce contexte que l'UML (Unified Modeling Language) entre en jeu. L'UML, en tant que langage de modélisation standardisé, offre une palette d'outils puissants pour représenter visuellement les aspects statiques et dynamiques d'un système. Son utilisation répandue dans l'industrie du logiciel témoigne de son efficacité à favoriser la communication et la collaboration au sein des équipes de développement. En permettant une vision d'ensemble claire et précise du système, l'UML facilite la prise de décision et garantit une cohérence globale tout au long du processus de développement logiciel.

3.1.2 Diagramme de classes

Un **diagramme de classes** est un outil de modélisation en génie logiciel qui décrit la structure statique d'un système. Il représente les classes, leurs attributs, leurs méthodes ainsi que les relations entre elles, comme les associations, héritages ou dépendances. Ce diagramme est essentiel pour concevoir et comprendre l'architecture du système avant son implémentation.

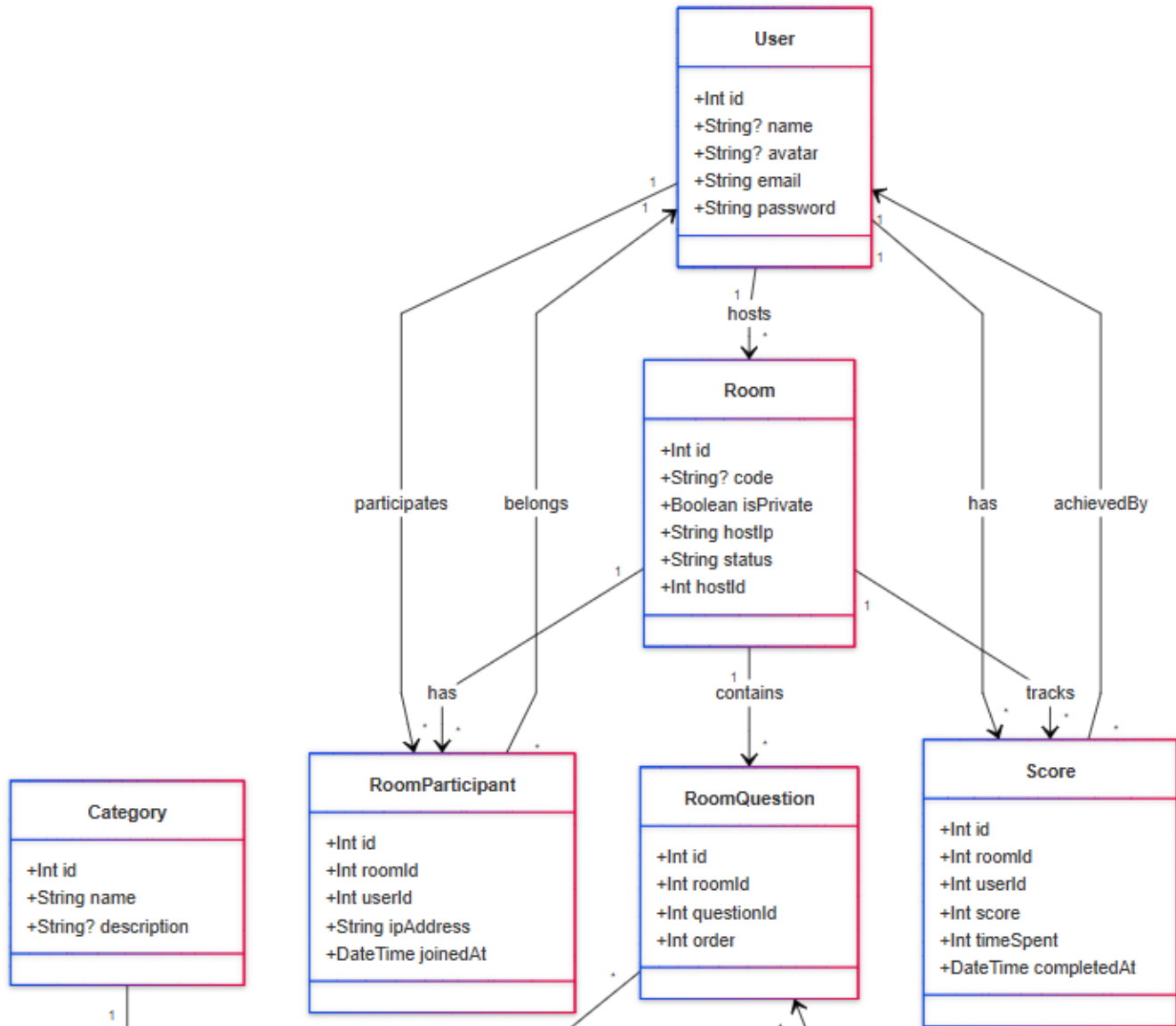


FIGURE 1 – Diagramme de classes partiel1

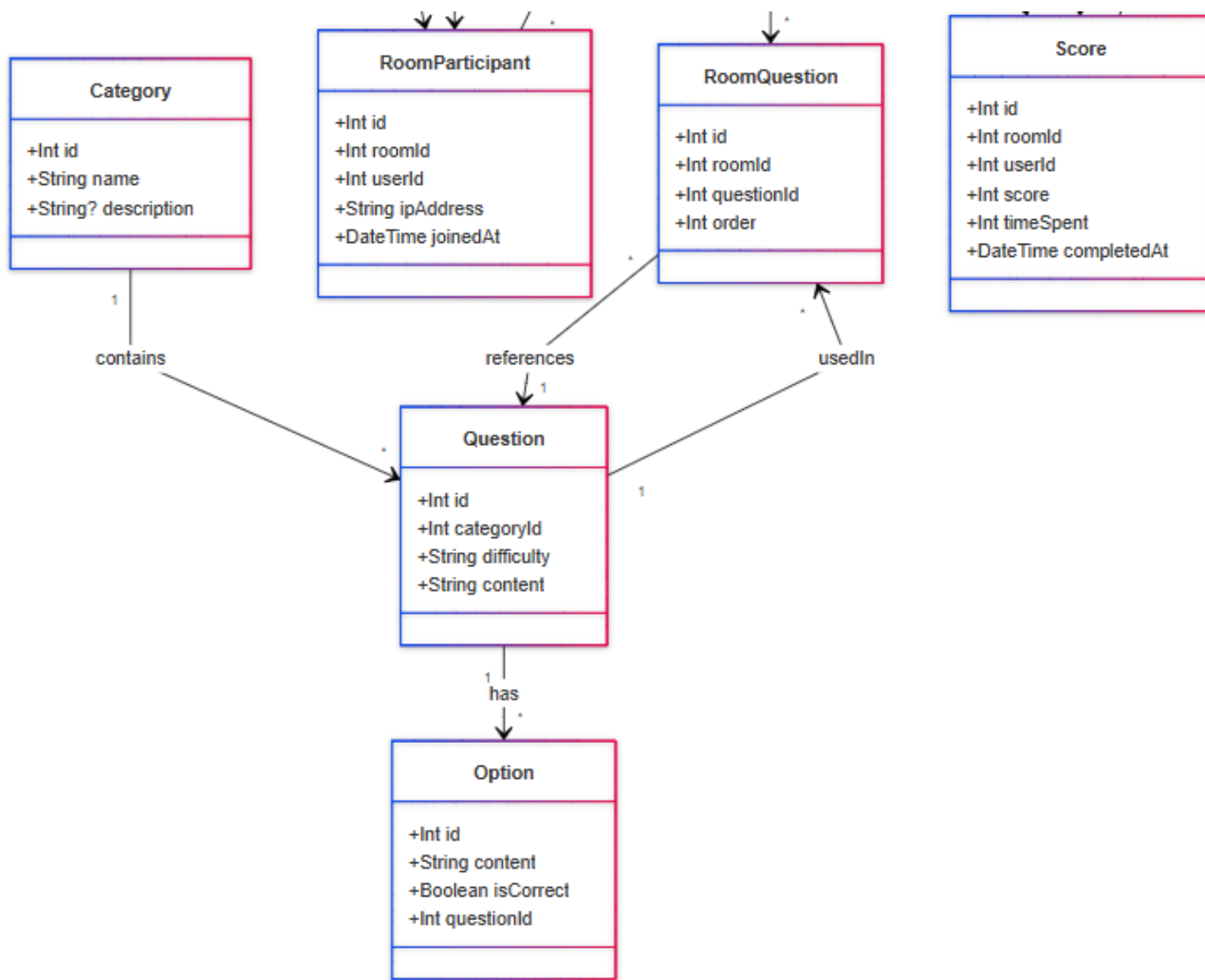


FIGURE 2 – Diagramme de classes partie2

3.1.3 Conclusion

La conception du système de jeu de quiz en temps réel, réalisée avec UML, a permis une représentation visuelle claire des interactions entre les joueurs, le serveur et la base de données. Cette méthodologie a facilité la compréhension des besoins fonctionnels, amélioré la collaboration au sein de l'équipe et posé des bases solides pour une implémentation efficace et structurée du projet.

3.2 Conception Front-end

3.2.1 Introduction :

La conception de l'interface utilisateur constitue une étape essentielle dans le développement d'une application logicielle, car elle détermine l'expérience directe de l'utilisateur avec le système. Elle vise à créer une interface intuitive, ergonomique et visuellement attrayante, tout en répondant aux besoins fonctionnels définis. Pour atteindre cet objectif, il est nécessaire d'adopter des outils et des méthodes adaptés au prototypage rapide et collaboratif. C'est dans ce cadre que l'outil Excalidraw s'avère particulièrement utile. En tant qu'outil de dessin numérique interactif, Excalidraw permet de créer des prototypes front-end visuels et clairs, facilitant ainsi la communication des idées de conception au sein de l'équipe. Grâce à sa simplicité d'utilisation et à sa flexibilité, il favorise une approche itérative dans le développement, assurant ainsi une cohérence et une efficacité accrues tout au long du processus de conception de l'interface.

3.2.2 Maquettes et Prototypes des Interfaces

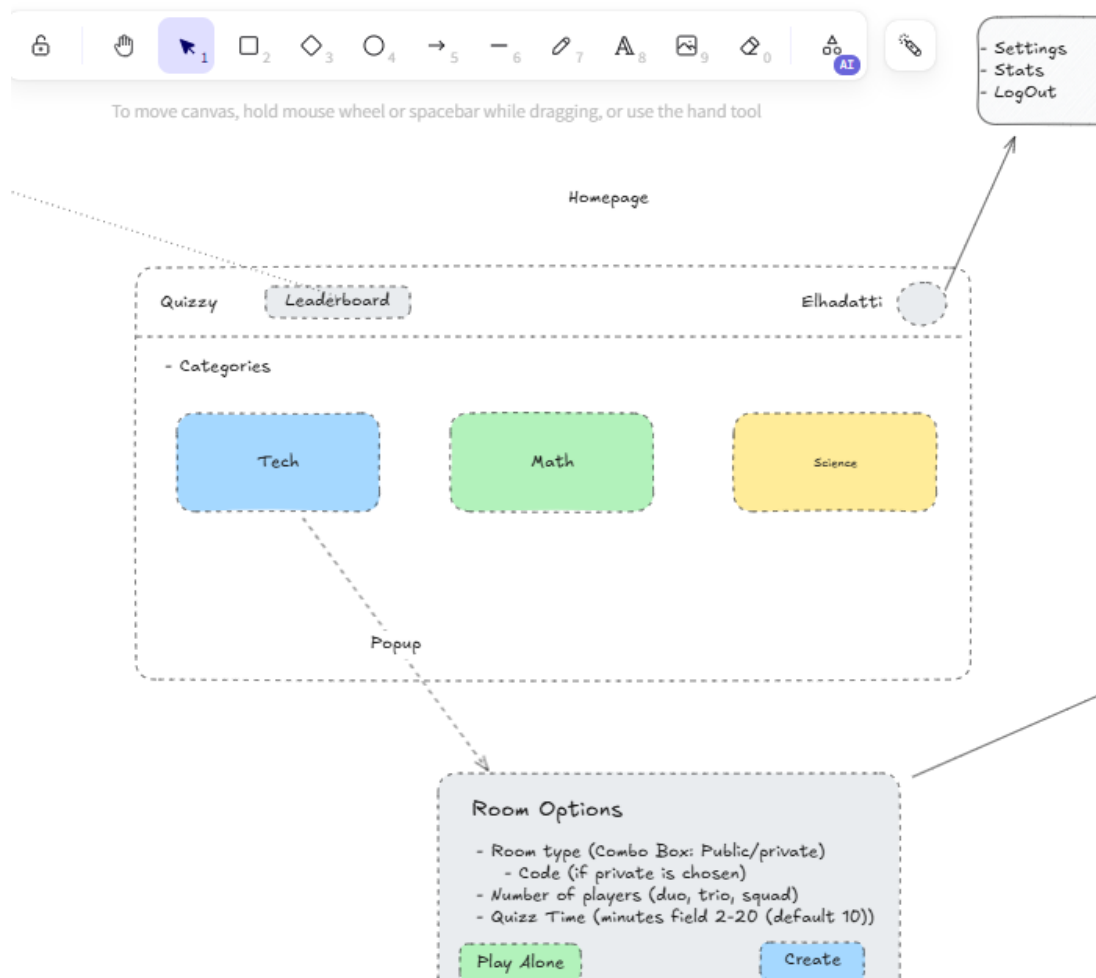


FIGURE 3 – interface 1

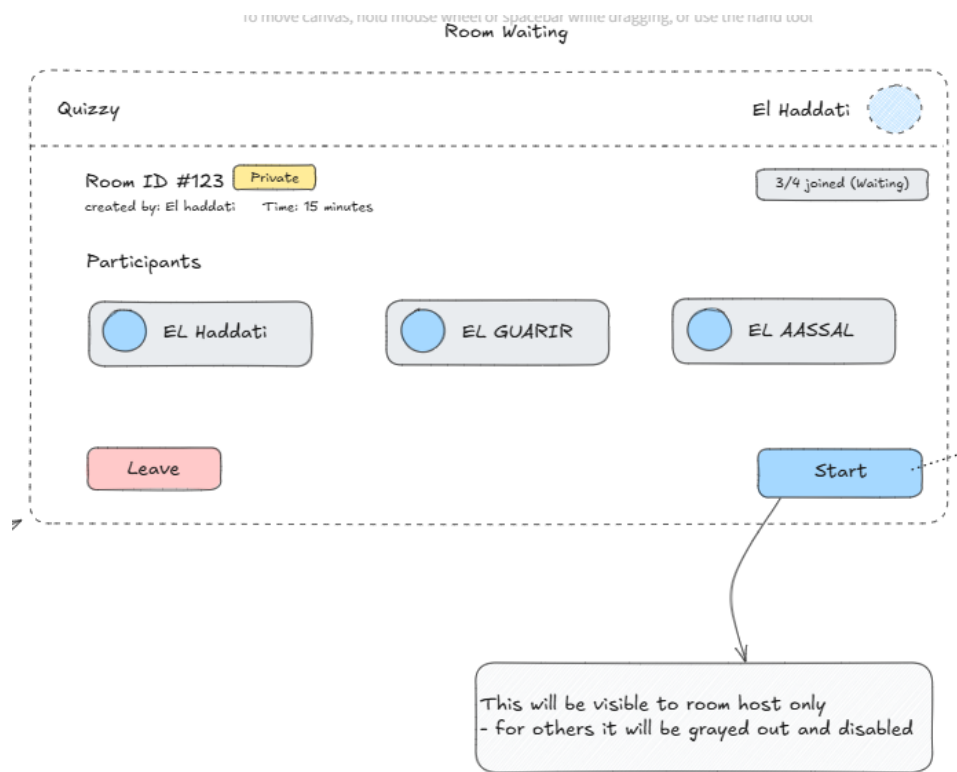


FIGURE 4 – interface 2

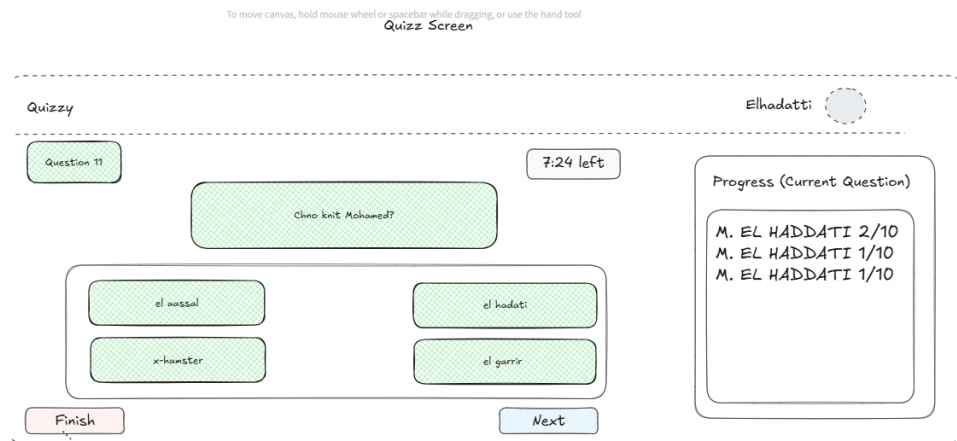


FIGURE 5 – interface 3

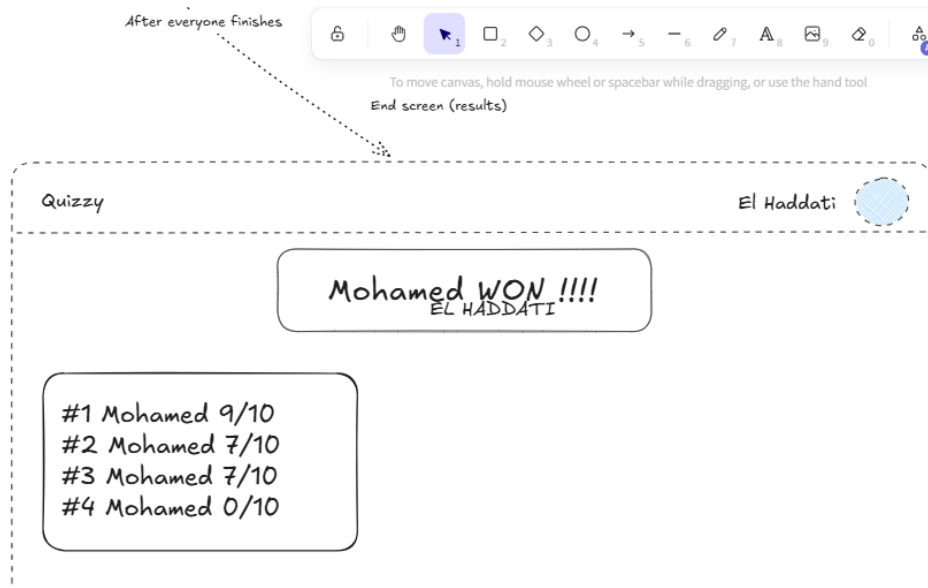


FIGURE 6 – interface 4

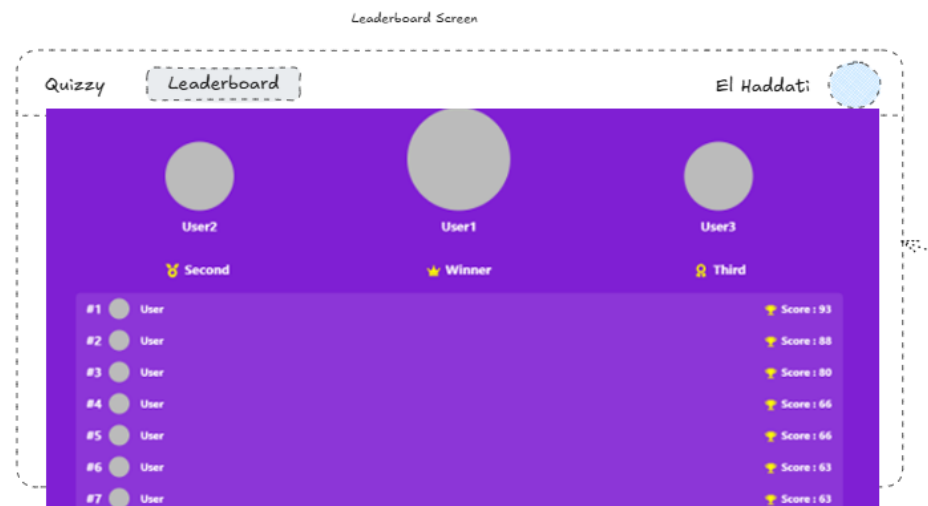


FIGURE 7 – interface 5

3.2.3 Conclusion :

La réalisation des maquettes et prototypes des interfaces avec Excalidraw a permis de visualiser clairement l'architecture du front-end et d'anticiper les interactions utilisateur. Cette étape a facilité la communication au sein de l'équipe, assuré une meilleure compréhension des besoins et posé une base solide pour le développement de l'application. Grâce à cet outil, nous avons pu adopter une approche itérative et collaborative, garantissant une conception ergonomique et cohérente répondant aux attentes des utilisateurs finaux.

4 Réalisation

4.1 Démarche

Pour notre projet de jeu de quiz en temps réel, nous avons choisi d'utiliser Java et ses technologies associées. Java, avec sa robustesse et sa gestion efficace du multitâche via les threads, nous permet d'assurer des échanges fluides entre les joueurs. L'architecture du projet repose sur les design patterns MVC, DAO et Singleton, garantissant une organisation claire et modulaire. Grâce aux sockets, la communication en temps réel est optimisée, et JDBC assure une interaction efficace avec la base de données. Cette combinaison de technologies nous permet de concevoir un jeu interactif et performant, offrant une expérience utilisateur dynamique et synchronisée.

4.1.1 MVC : Modèle-Vue-Contrôleur

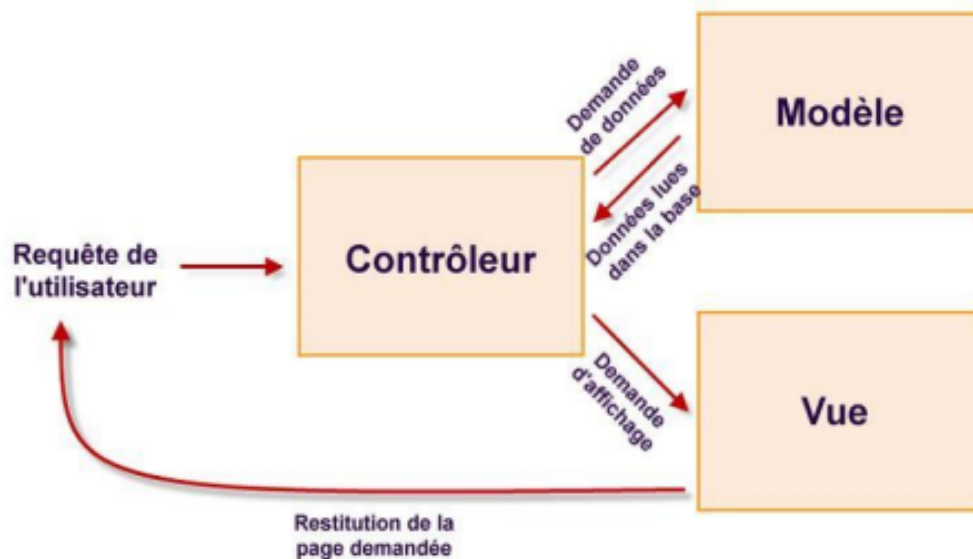


FIGURE 8 – MVC

MVC : Modèle-Vue-Contrôleur (MVC) est un modèle d'architecture logicielle utilisé dans des frameworks tels que Laravel pour organiser le code en trois parties : le modèle pour les données, la vue pour l'interface utilisateur, et le contrôleur pour gérer les interactions entre les deux.

4.2 Outils et environnement Techniques



FIGURE 9 – JAVA

JAVA : est un langage de programmation orienté objet, robuste et multiplateforme, conçu pour développer des applications variées, allant des logiciels de bureau aux applications web et mobiles.



FIGURE 10 – IntelliJ IDEA

IntelliJ IDEA : est un environnement de développement intégré (IDE) performant, conçu pour le développement Java et compatible avec plusieurs autres langages, offrant des outils intelligents et une interface intuitive pour coder efficacement



FIGURE 11 – Scene Builder

Scene Builder : est un outil visuel permettant de concevoir des interfaces graphiques en JavaFX, en générant automatiquement les fichiers FXML nécessaires à leur intégration dans des applications Java.



FIGURE 12 – Excalidraw

Excalidraw : est un outil de dessin collaboratif en ligne qui permet de créer des croquis, des schémas et des prototypes interactifs, idéal pour la conception rapide et intuitive d'idées.



FIGURE 13 – MySQL Logo

MySQL : est un système de gestion de base de données open source qui utilise le langage SQL. Il permet de stocker, organiser et récupérer des données de manière efficace. MySQL est largement utilisé dans le développement web pour sa fiabilité, sa rapidité et sa compatibilité avec de nombreuses plates-formes. Il offre des fonctionnalités avancées telles que la réplication, la gestion des utilisateurs et des privilèges, ainsi que des outils de sauvegarde et de récupération..



FIGURE 14 – Nginx Logo

Nginx : est un serveur web open source performant qui peut également être utilisé comme proxy inverse, proxy de messagerie ou équilibrage de charge. Il est conçu pour gérer simultanément un grand nombre de connexions, ce qui le rend idéal pour des applications nécessitant une haute disponibilité et une faible latence. Nginx est souvent utilisé pour servir des sites web statiques, distribuer la charge entre plusieurs serveurs ou sécuriser les applications web grâce à son rôle de proxy.

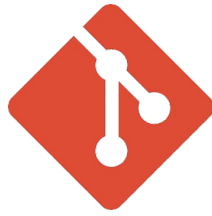


FIGURE 15 – git Logo

Git : Git est un système de contrôle de version distribué gratuit et open source conçu pour tout gérer, des petits aux très grands projets, avec rapidité et efficacité.

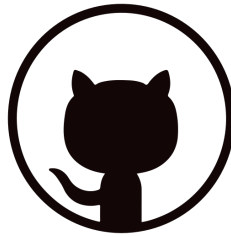


FIGURE 16 – github logo

GitHub : GitHub est une plateforme de développement logiciel qui permet aux développeurs de collaborer, de stocker et de gérer des projets de code source en utilisant le système de contrôle de version Git.

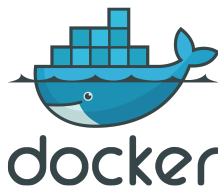


FIGURE 17 – Docker Logo

Docker : est une plateforme logicielle qui permet de créer, déployer et exécuter des applications dans des conteneurs. Un conteneur est une unité légère et isolée qui regroupe tout ce dont une application a besoin pour fonctionner : son code, ses dépendances, ses bibliothèques et ses fichiers de configuration. Cela garantit que l'application fonctionne de manière cohérente, quelle que soit l'environnement (développement, test ou production).

4.3 Capture d'écrans des interface du Jeu :

4.3.1 Authentification :

Cette interface représente la page d'authentification, elle permet aux utilisateurs d'accéder à l'application si les informations saisies sont correctes .

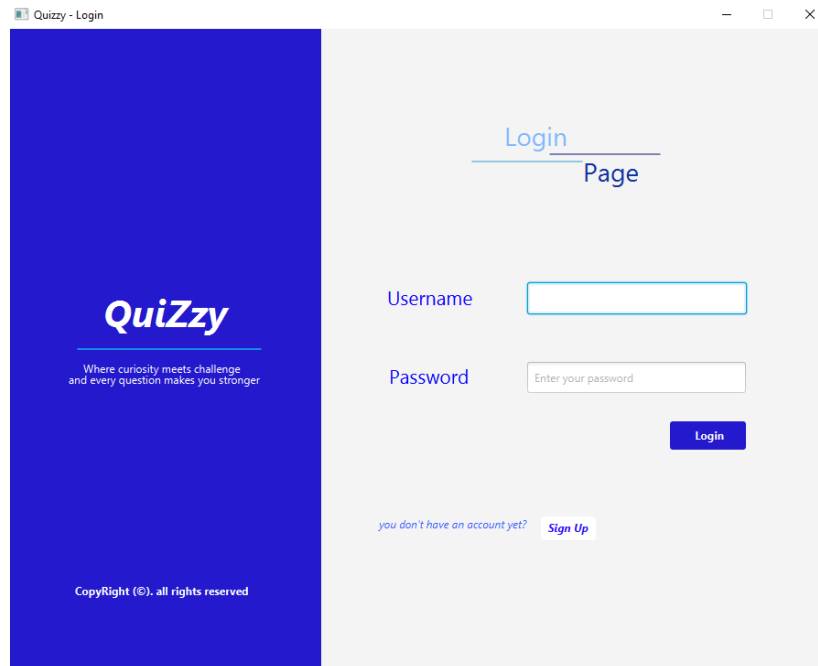


FIGURE 18 – page du login

si les informations sont incorrèts il retourne une alerte avec un message en rouge de "**invalid credentials!**" sinon si les deux sont corrects il m'a redireger vers la page d'accueil

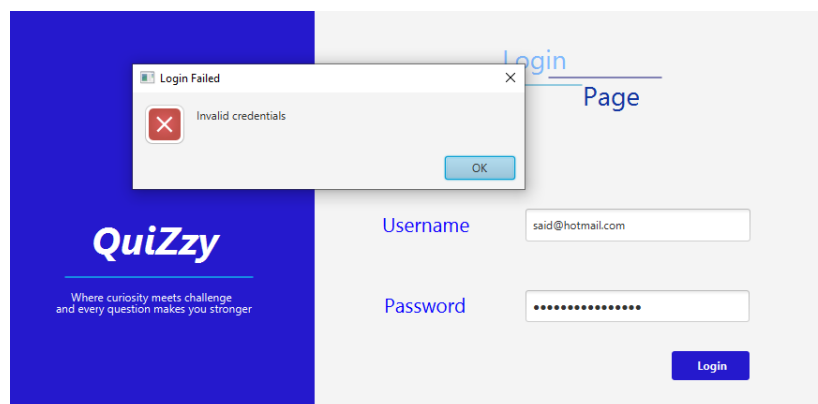


FIGURE 19 – La connexion a échoué

Cette interface représente la page du registre, elle permet l'ajout aux utilisateurs et puis d'accéder à l'application si les informations saisies sont correctes .

Quizzy - Sign Up

Quizzy
Where curiosity meets challenge
and every question makes you stronger

CopyRight (©). all rights reserved

Registration
Page

Name

Email

Password

Password confirmation

[I have already an account](#)

Registre

FIGURE 20 – page du registre

Exemple d'ajout de l'utilisateur Mohamed :

Registration
Page

Name

Email

Password

Password confirmation

Registre

FIGURE 21 – ajout d'un utilisateur

on click sur le button registre ,il m'afficher une alerte du success :
(Voir la page suivante)

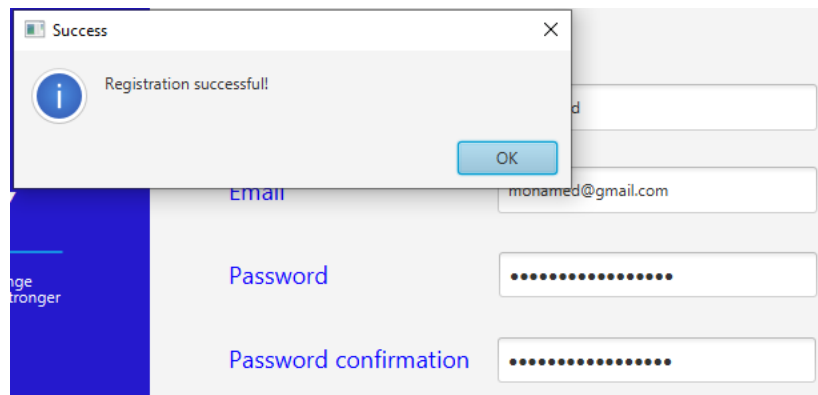


FIGURE 22 – ajout d'un utilisateur avec succes

maintenant essayons d'accéder à l'application avec cet utilisateur :

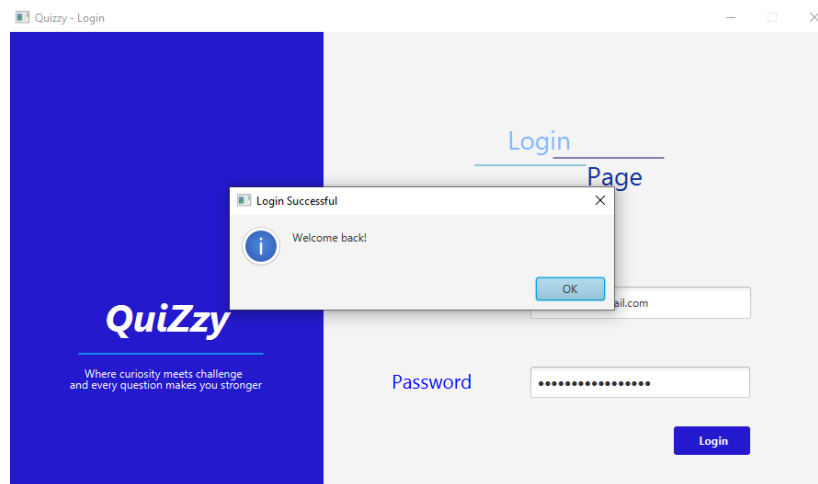


FIGURE 23 – Login avec succes

les données sont correctes ,alors il m'a redireger à la page d'accueil :

4.3.2 page d'accueil :

Comme vous pouvez le voir sur la page d'accueil, il y a une section pour les chambres publiques qui ont déjà été lancées. Sinon, si l'utilisateur souhaite intégrer une chambre privée, il doit entrer le code de cette dernière dans le champ de texte prévu à cet effet, situé dans la barre en haut

(Voir la page suivante)

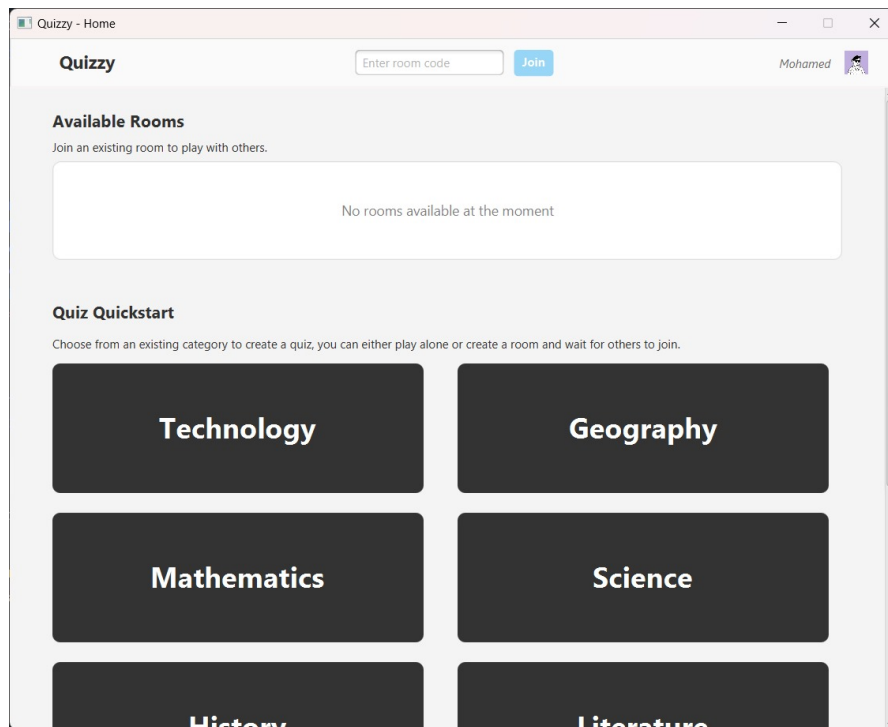


FIGURE 24 – page d'accueil

4.3.3 Jouer seul :

Si l'utilisateur veut jouer tout seul il doit cliquer sur l'un des catégories pour choisir leur type préféré, on click sur le bouton **single player** il peut jouer

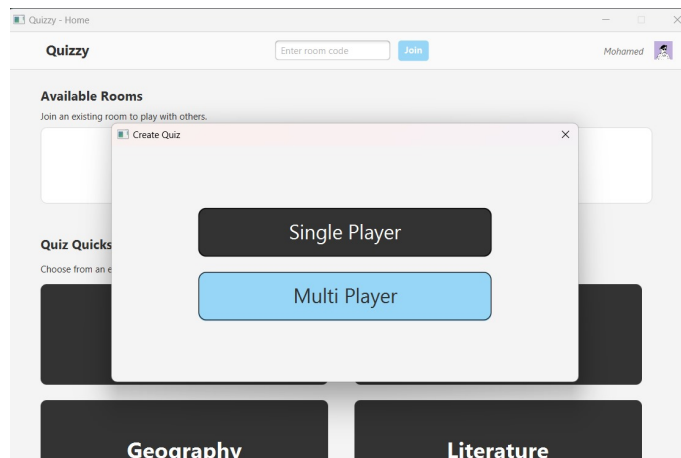


FIGURE 25 – crée un quiz

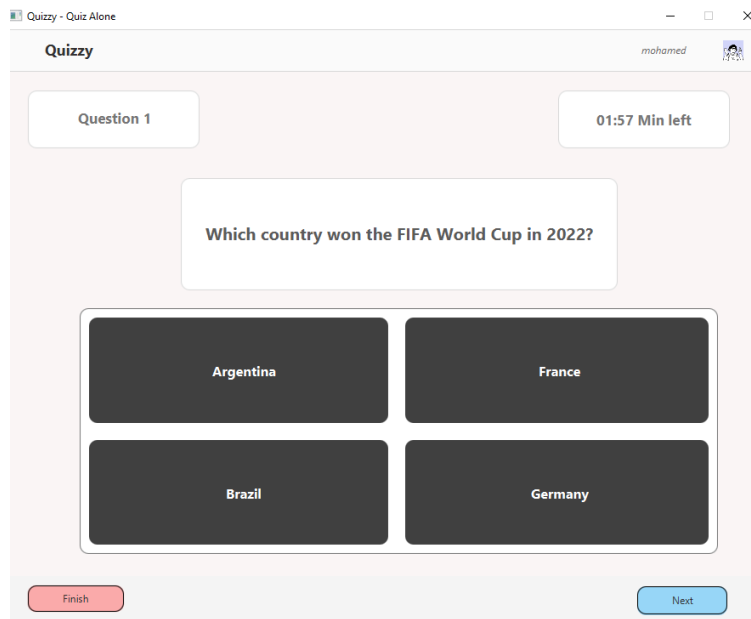


FIGURE 26 – jouer seul 1

l'utilisateur doit choisir une seule réponse qui est valide et cliquer sur le bouton **Next**

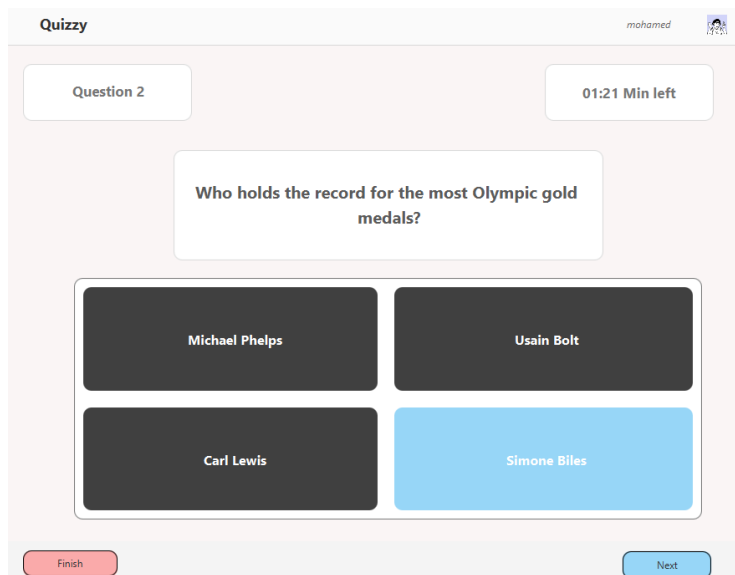


FIGURE 27 – jouer seul 2

lorsque l'utilisateur clique sur le bouton **finish** avant terminer le quiz une alerte de confirmation va afficher indique au utilisateur que les résultats du quiz ne vont pas sauvegarder

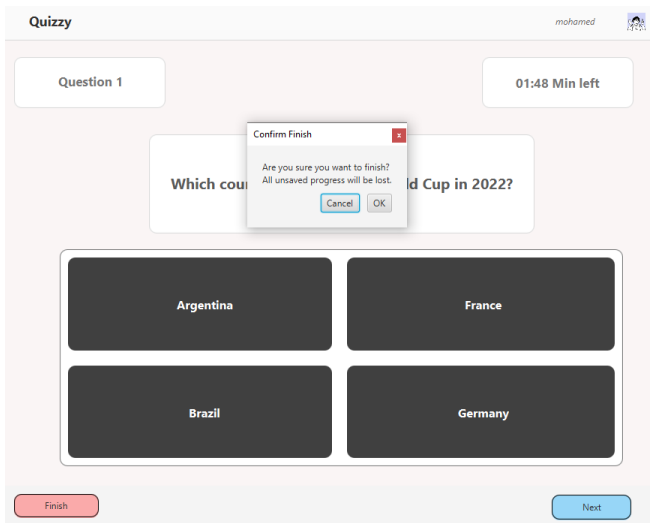


FIGURE 28 – le bouton finish



FIGURE 29 – le bouton show results

lorsque l'utilisateur a terminer et choisir la dernière question il peut cliquer sur le bouton **show results** pour voir la note du quiz

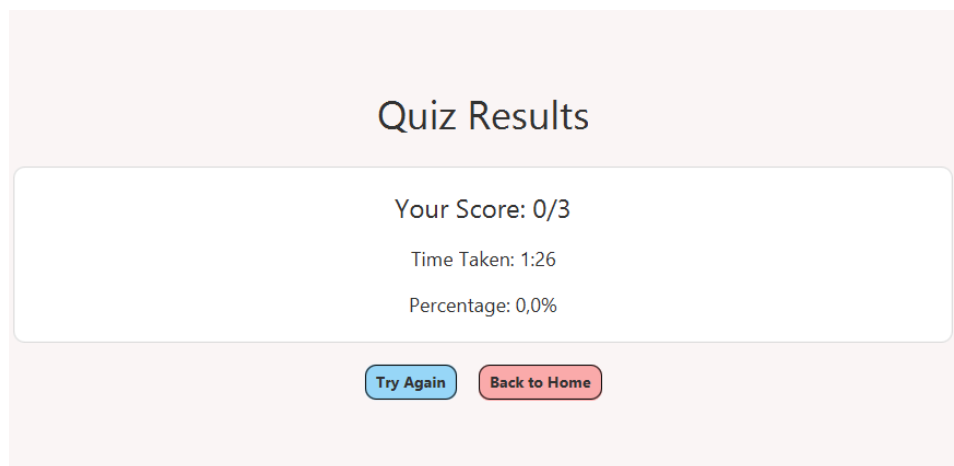


FIGURE 30 – résultats du quiz

si l'utilisateur a cliquer sur le bouton **try again** alors il va redireger vers le meme quiz avec les meme questions sinon il va etre redireger vers la page d'accueil

4.3.4 Multi-joueurs (temps réel) :

si l'utilisateur a cliquer sur le bouton **Multi Player** alors il doit choisir les parametres du quiz et ce dernier(utilisateur) va etre le **host** ou bien **serveurSocket**

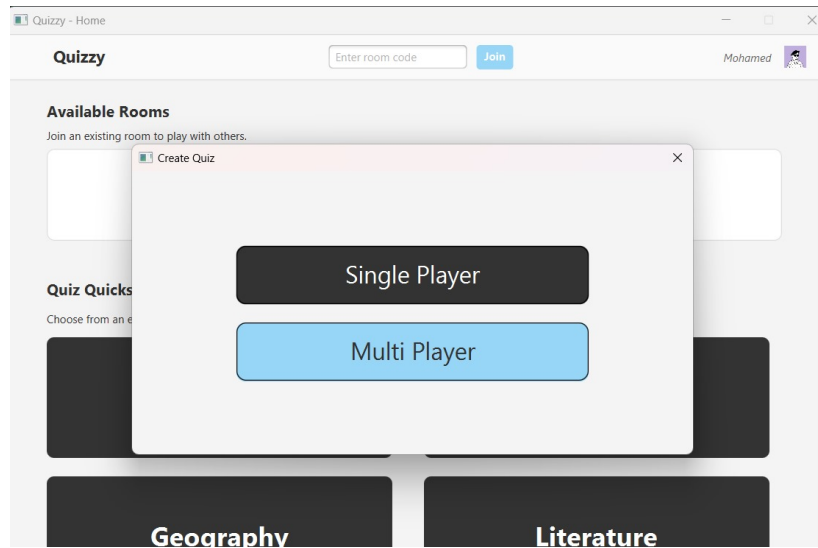


FIGURE 31 – choix du type du quiz

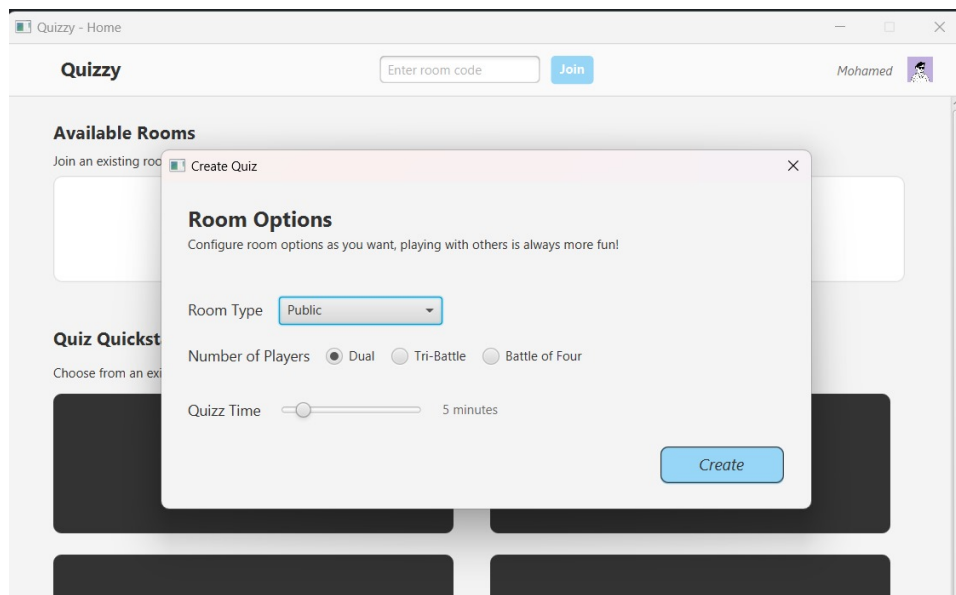


FIGURE 32 – configurer les parametres du quiz

alors après la configuration d'une chambre l'utilisateur va redireger vers cette page , il doit attend les autres utilisateurs puis il peut lancer le quiz :

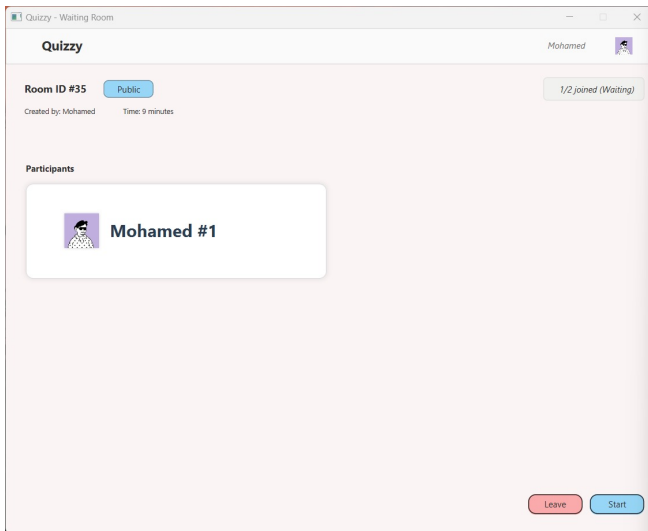


FIGURE 33 – attente 1

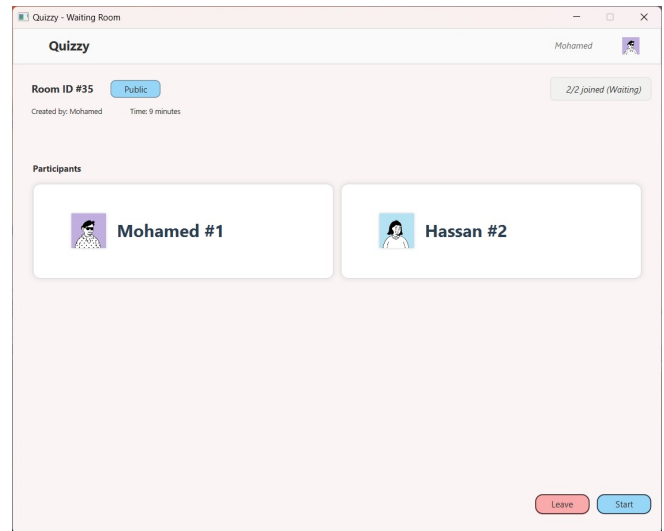


FIGURE 34 – attente 2

lorsque tous les utilisateurs ont entré le createur de la quiz peut la lancer on click sur le bouton start

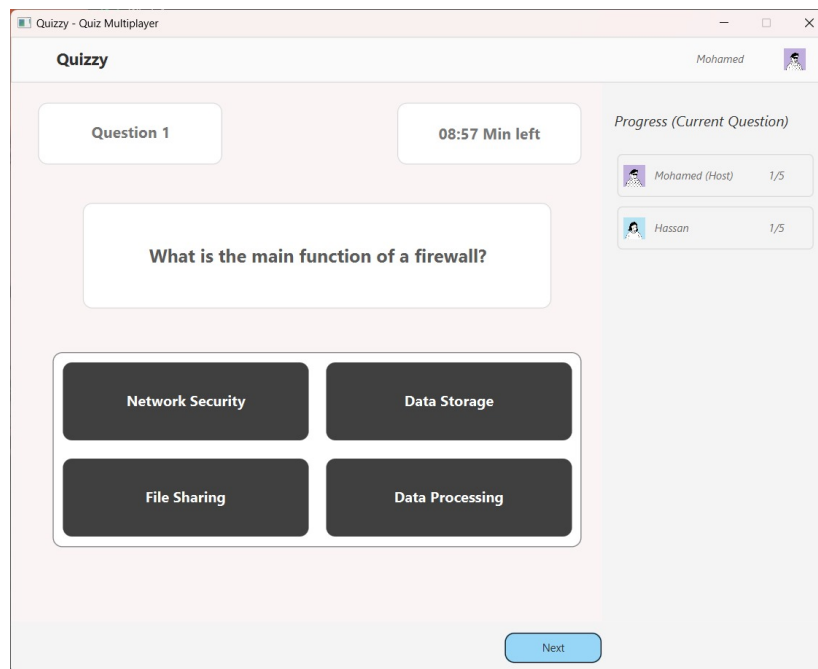


FIGURE 35 – quiz multi-joueurs

d'après la session de l'utilisateur HASSAN on peut voir le progress du mohamed dans la bare qui s'existe à droite alors chacun des utilisateurs peut voir le progress des autres, car le temps est un axe pour la note

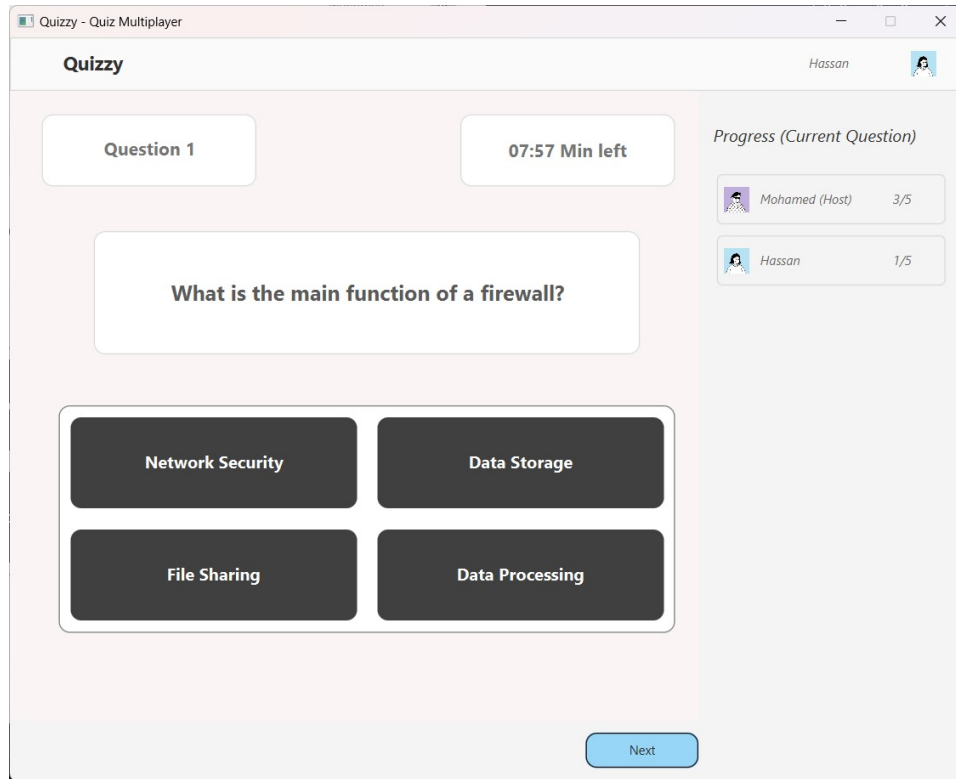


FIGURE 36 – progress dans le quiz

4.4 Perspectives :

Ce projet offre plusieurs possibilités d'amélioration pour enrichir l'expérience utilisateur et augmenter la valeur fonctionnelle de l'application :

1. ****Intégration d'un leaderboard**** : Ajouter une fonctionnalité de classement permettant aux joueurs de voir leurs performances par rapport aux autres participants. Cela renforcerait l'aspect compétitif du jeu et encouragerait une plus grande participation.
2. ****Interface de création de quiz**** : Développer une interface conviviale permettant aux utilisateurs ou aux administrateurs de créer facilement de nouveaux quiz. Cela rendrait l'application plus flexible et autonome, en facilitant l'ajout de contenu personnalisé.

Ces ajouts permettraient de rendre l'application plus interactive et de répondre aux besoins des joueurs tout en augmentant son attrait global.

4.5 Conclusion :

En conclusion, ce projet de jeu de quiz en temps réel représente une étape importante dans le développement d'applications interactives et performantes. Grâce à Java et aux principes de conception logicielle utilisés, nous avons construit une base solide sur laquelle il est possible de s'appuyer pour de futures évolutions.

5 Conclusion générale

Pour conclure, ce projet de jeu de quiz en temps réel a permis à notre équipe de mettre en pratique une large gamme de compétences techniques et de travailler en synergie pour atteindre un objectif commun. En utilisant Java et des technologies telles que les sockets et JDBC, nous avons construit une application performante, capable de gérer des échanges en temps réel entre les joueurs tout en garantissant une expérience fluide. L'architecture, construite autour des design patterns MVC, DAO et Singleton, assure une maintenance aisée et une évolutivité de l'application. L'utilisation des threads pour le multitâche et des mécanismes d'I/O a optimisé la réactivité du jeu, permettant ainsi une gestion efficace des données et des interactions. Ce projet a non seulement été un challenge technique stimulant, mais il a également renforcé notre capacité à collaborer efficacement en équipe pour relever des défis complexes et fournir une solution complète et aboutie.

Références bibliographiques

Les liens utiles

Java Documentation <https://www.javatpoint.com/java-tutorial> (consulté le 20-11-2024)

JavaFX Documentation <https://openjfx.io/> (consulté le 15-11-2024)

Docker Documentation <https://docs.docker.com/> (consulté le 10-12-2024)

Stack Overflow <https://stackoverflow.com/> (consulté le 20-11-2024)

GitHub <https://github.com/> (consulté le 15-11-2024)