

RAPPORT DE STAGE

Réalisée par :

Houneyda Cherni

Classe :

Ing4 B GL

Encadrante :

Mme.Amira Dridi

Année Universitaire

2022

– 2023

**Système de gestion de quiz**

**Remerciements**

Nos remerciements les plus sincères et notre profonde gratitude s’adressent à tous ceux qui ont voulu apporter l’assistance nécessaire

au bon déroulement de ce projet. Nous adressons nos remerciements à notre encadrant, Madame Amira Dridi pour

ses conseils intéressants, son encouragement continu, ainsi que le

temps qu’il nous a réservé malgré sa grande occupation. Finalement, Nous tenons aussi à remercier vivement les membres du jury qui nous ont fait l’honneur d’évaluer ce travail.

[Introduction Générale 1](#_Toc149494995)

[Chapitre 1 : Présentation du cadre du projet 2](#_Toc149494996)

[Introduction 3](#_Toc149494997)

[I. Présentation de sujet 3](#_Toc149494998)

[1.Contexte générale 3](#_Toc149494999)

[2.Objectifs du projet 3](#_Toc149495000)

[Conclusion 3](#_Toc149495001)

[Chapitre 2 : Analyse et spécification de besoins 4](#_Toc149495002)

[Introduction 5](#_Toc149495003)

[I. Spécification des besoins 5](#_Toc149495004)

[1. Identification des Acteurs 5](#_Toc149495005)

[2.Analyse des besoins fonctionnels 5](#_Toc149495006)

[2.1Les besoins fonctionnels liés à l’étudiant 5](#_Toc149495007)

[2.2 Les besoins fonctionnels liés à l’administrateur 6](#_Toc149495008)

[3. Analyse des besoins non fonctionnels : 6](#_Toc149495009)

[Conclusion 6](#_Toc149495010)

[Introduction 8](#_Toc149495011)

[I.Le choix de l’atelier génie logiciel 8](#_Toc149495012)

[II.Diagramme des cas d’utilisation 8](#_Toc149495013)

[III. Diagramme de classe 9](#_Toc149495014)

[IV. Diagrammes de séquences 10](#_Toc149495015)

[1. Le diagramme de séquences de l’authentification 10](#_Toc149495016)

[2. Le diagramme de séquences d’ajout d’un quiz 11](#_Toc149495017)

[3. Le diagramme de séquence de passer un quiz 13](#_Toc149495018)

[Conclusion 13](#_Toc149495019)

[Chapitre 4 : Réalisation 14](#_Toc149495020)

[Introduction 15](#_Toc149495021)

[I.Environnement logiciel : 15](#_Toc149495022)

[II .Technologies utilisées : 16](#_Toc149495023)

[III.Démonstration des interfaces 18](#_Toc149495024)

[3.Partie administrateur 19](#_Toc149495025)

[4. Partie Etudiant 21](#_Toc149495026)

[Conclusion générale 24](#_Toc149495027)

Figure 1 : Diagramme de cas d’utilisation ........................................................................ 10 Figure 2 : Diagramme de classe ...................................................................................... 11

Figure 3 : Diagramme de séquence d’authentification ......................................................12

Figure 4 : Diagramme de séquence d’ajout un quiz ..........................................................13

Figure 5 : Diagramme de séquence de passer un quiz …………………………................ 14

Figure 7 : Page Login ………………………........................................................................17

Figure 8 : Page Inscription...................................................................................................18

Figure 9 : Profile de l’administrateur....................................................................................19

Figure 10 : L’ajout d’un category..........................................................................................19

Figure 11 : Liste des catégories............................................................................................20

Figure 12 : L’ajout d’un quiz................................................................................................. 20

Figure 13 : Les quizes...........................................................................................................20

Figure 14 : L’ajout d’une question........................................................................................ 21

Figure 15 : L’authentification d’étudiant............................................................................... 22

Figure 16 : Exemple d’un quiz…………………………………………………………………… 22

Figure 17 : Exemple d’un quiz Java ……………………………………………………………. 23

Figure 18 : Résultat d’un quiz……………………………………………………………………. 24

# Introduction Générale

Grâce aux technologies révolutionnaires employées dans les ordinateurs ainsi qu’aux avancées inouïes en matière de réseaux informatiques, le web est partout. L'évolution rapide de ce phénomène et sa popularité ne sont pas dues seulement à la stabilité du réseau Internet et la rapidité des transactions mais aussi grâce à la commodité qu'offrent les applications qui se chargent de servir le client.

Ces applications, qui sont dans la majorité des cas sous forme de site web dynamique, ont subi un développement incomparable qui les rend de plus en plus capables de satisfaire les besoins des clients en fournissant plusieurs services.

A travers notre site, il est possible de faire un quiz pour tester ses compétences en technologies informatiques.

Dans ce rapport, nous allons présenter dans la première partie une présentation générale, dans la seconde partie la spécification et l'analyse des besoins qui nous permettent d'aboutir aux différents services et fonctionnalités offerts par le site. La troisième partie est consacrée à la conception de l'application et la conception. La dernière partie décrit la réalisation de site ainsi que ces différentes interfaces.

Notre travail s’achèvera par une conclusion générale et un ensemble de perspectives.

# Chapitre 1 : Présentation du cadre du projet

## Introduction

Au niveau de ce chapitre, nous allons présenter le projet réalisé tout en expliquant ses objectifs ainsi que les fonctionnalités nécessaires pour le réaliser.

## I. Présentation de sujet

### 1.Contexte générale

Cette première partie du rapport donne une vision globale du projet Conception et développement d’un système de gestion de quiz.

### 2.Objectifs du projet

L’objectif du projet est de réaliser une application intitulée «WeLoveQuiz ». Le projet consiste dans la conception et le développement d’un quiz pour tester ses compétences en langages informatiques.

## Conclusion

Dans ce chapitre, nous avons présenté le contexte général et l’objectif de projet. Dans le chapitre suivant, nous aborderons en détails l’architecture existante ainsi que ces lacunes.

# Chapitre 2 : Analyse et spécification de besoins

## Introduction

Dans le cadre de ce chapitre, nous allons présenter la spécification de besoins qui consiste à la qualification des besoins fonctionnels et non fonctionnels attendus du système afin de mieux comprendre le projet.

## I. Spécification des besoins

La spécification de besoins, constitue la phase de départ de toute application à développer dans laquelle nous allons identifier les besoins de notre application.

Nous avons distingué des besoins fonctionnels qui présentent les fonctionnalités attendues de notre application et les besoins non-fonctionnels pour éviter le développement d’une application non satisfaisante ainsi de trouver un accord commun entre les spécialistes et les utilisateurs pour réussir le projet.

### 1. Identification des Acteurs

Nous allons maintenant énumérer les acteurs susceptibles d'interagir avec le système.

Les acteurs de notre application web sont :

* **Etudiant** : toutes les personnes peuvent consulter les services de l’application web.

* **Administrateur** : les administrateurs qui gérer les fonctionnalités de l’application web.

### 2.Analyse des besoins fonctionnels

##### Les besoins fonctionnels liés à l’étudiant

* S’inscrire,
* S’authentifier,
* Choisir la catégorie de quiz à compléter,
* Répondre aux questions,
* Envoyer les questions,
* Voir le résultat,
* Se déconnecter

##### 2.2 Les besoins fonctionnels liés à l’administrateur

* S’authentifier,
* Ajouter les quiz,
* Ajouter une nouvelle catégorie,
* Voir tous les quiz,
* Publier ou dé-publier un quiz,
* Modifier un quiz,
* Supprimer un quiz,
* Se déconnecter

### 3. Analyse des besoins non fonctionnels :

Les spécifications non fonctionnelles décrivent les contraintes auxquelles est soumis le système pour sa réalisation et son bon fonctionnement :

**La sécurité** : Le système doit être sécurisé, en assurant l’intégrité et le non-répudiation des données stockées dans la base, compte tenu qu’elles reflètent des informations privées sur les différents acteurs.

**L’ergonomie** : Les interfaces utilisées par le logiciel doivent être claires, concises et faciles à manipuler.

**La performance** : le fait que le site est un multi utilisateur et contient un nombre de données, le temps de traitement doit être réduit impérativement.

## Conclusion

Dans ce chapitre, nous avons distingué les rôles de chaque utilisateur et spécifié les besoins fonctionnels et non fonctionnels. Dans le chapitre suivant, nous allons présenter la conception qui a été mise en œuvre tout au long de la réalisation de ce projet.

**Chapitre 3 : Conception**

## Introduction

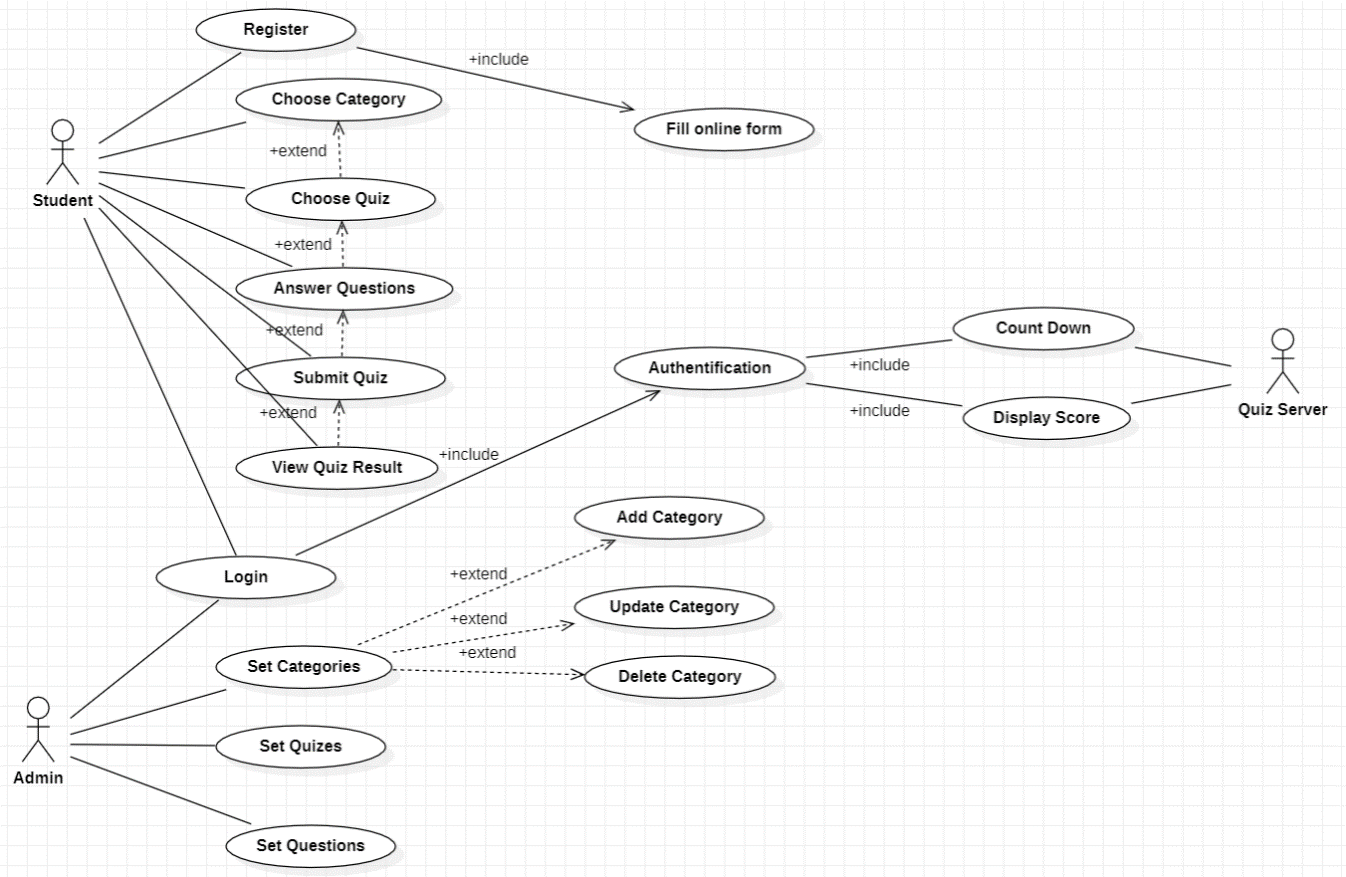
Après avoir cité les différentes fonctionnalités de notre projet dans le deuxième chapitre, on va commencer maintenant une phase importante et indispensable dans le cycle de vie d’une application. Cette phase est la conception qui a pour but d’expliquer le déroulement de notre application ainsi qu’assurer une bonne compréhension des besoins des utilisateurs.

## Le choix de l’atelier génie logiciel

Notre outil de conception UML est le logiciel StarUML, qui est un outil de modélisation open source, sans limite et gratuit.

## Diagramme des cas d’utilisation

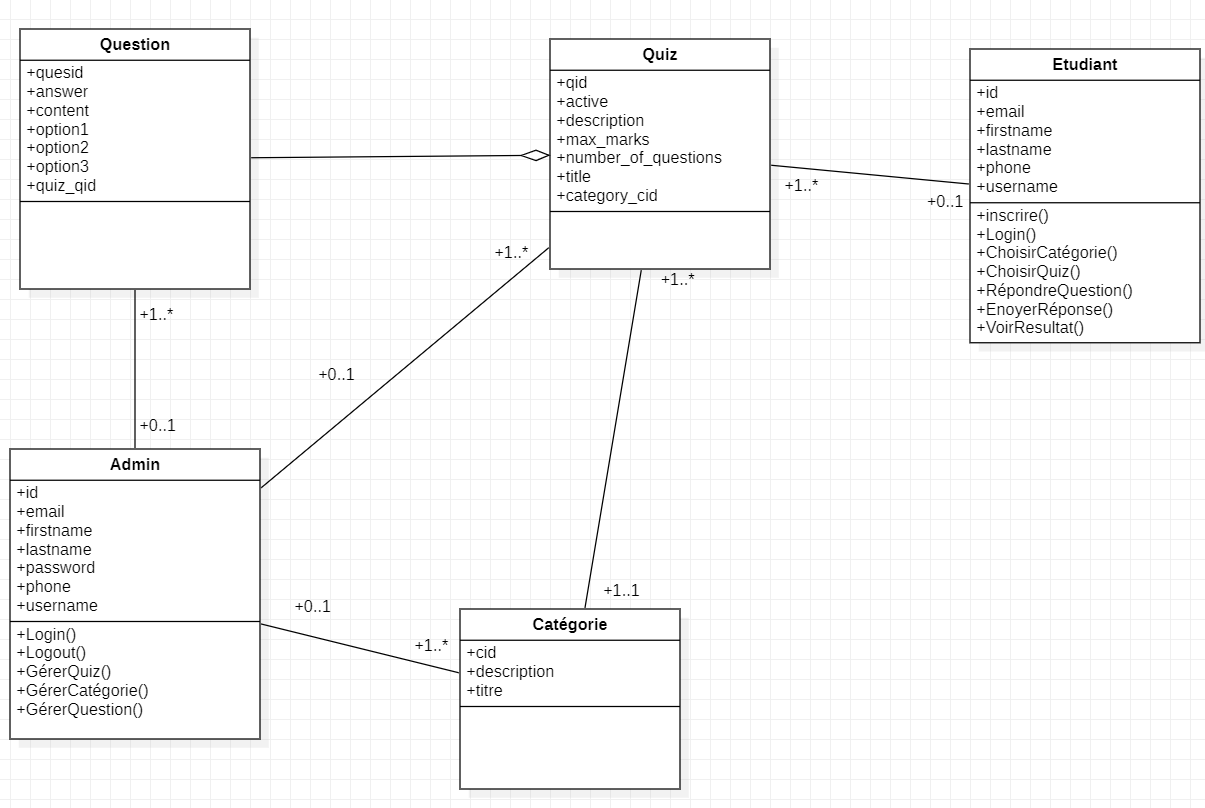
Les rôles des diagrammes de cas d’utilisation sont de recueillir, d’analyser et d’organiser les besoins, ainsi que de recenser les grandes fonctionnalités d’un système. Il s’agit donc de la première étape UML pour la conception d’un système.



***Figure 1: Diagramme de cas d’utilisation***

## III. Diagramme de classe

Un diagramme de classes UML décrit les structures d’objets et d’informations utilisées sur notre site web, à la fois en interne et en communication avec ses utilisateurs. Il décrit les informations sans faire référence à une implémentation particulière. Ses classes et relations peuvent être implémentées de nombreuses manières, comme les tables de bases de données, les nœuds XML ou encore les compositions d’objets logiciels.

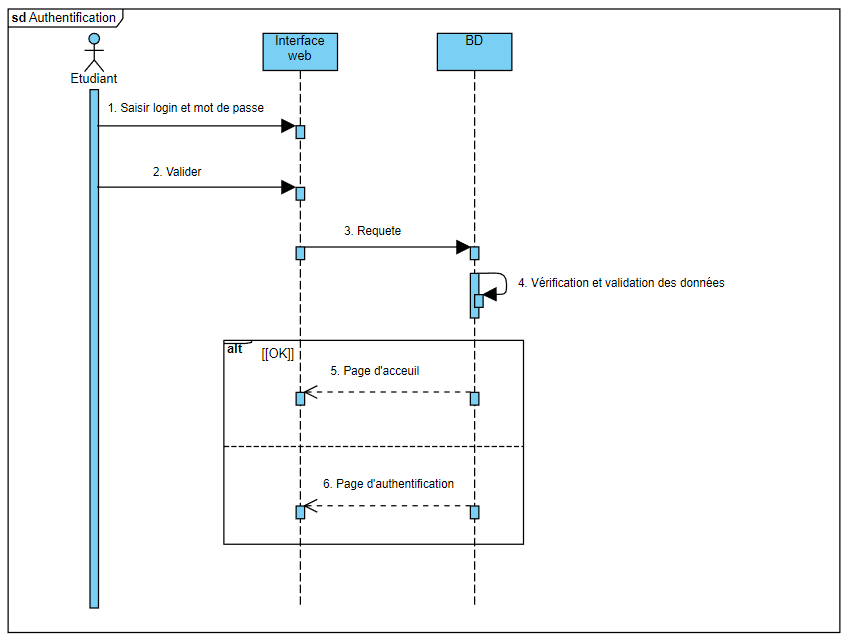


*Figure 2 : Diagramme de classe*

## IV. Diagrammes de séquences

Un diagramme de séquences est un diagramme d’interaction qui expose en détail la façon dont les opérations sont effectuées : quels messages sont envoyés et quand ils le sont. Les diagrammes de séquences sont organisés en fonction du temps qui s’écoule au fur et à mesure que nous parcourons la page. Les objets impliqués dans l’opération sont répertoriés de gauche à droite en fonction du moment où ils prennent part dans la séquence.

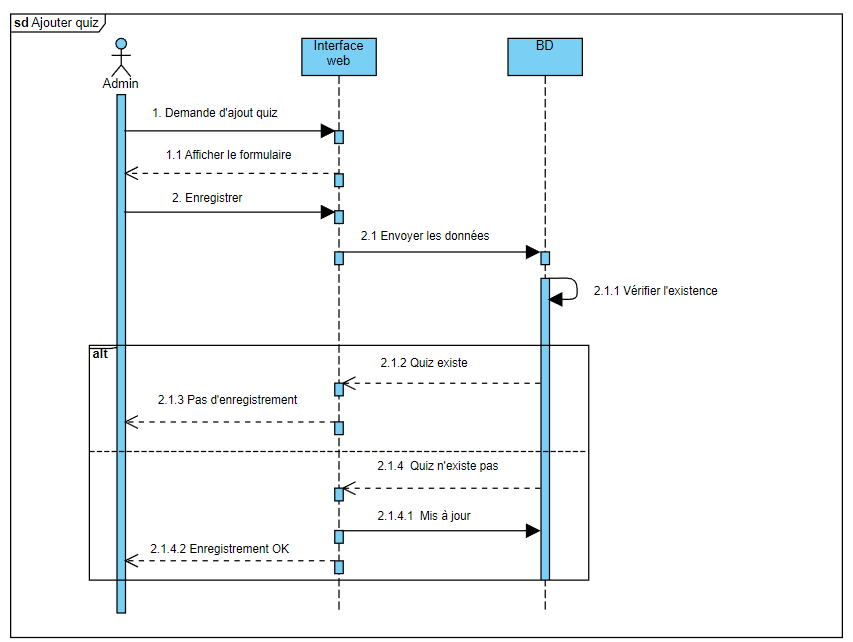
### 1. Le diagramme de séquences de l’authentification



*Figure 3 : Diagramme de séquence d’authentification*

* L’étudiant / l’administrateur accède à l’interface web,
* L’étudiant / l’administrateur saisie le login et le mot de passe,
* Le système envoie les données à la base,
* La base de données vérifie les données,
* Si les données sont correctes, l’interface affiche la page d’accueil de l’étudiant ou la page de l’admin pour l’administrateur,
* Si les données ne sont pas correctes, l’interface retourne au l’étudiant ou à l'administrateur que les données sont erronées.

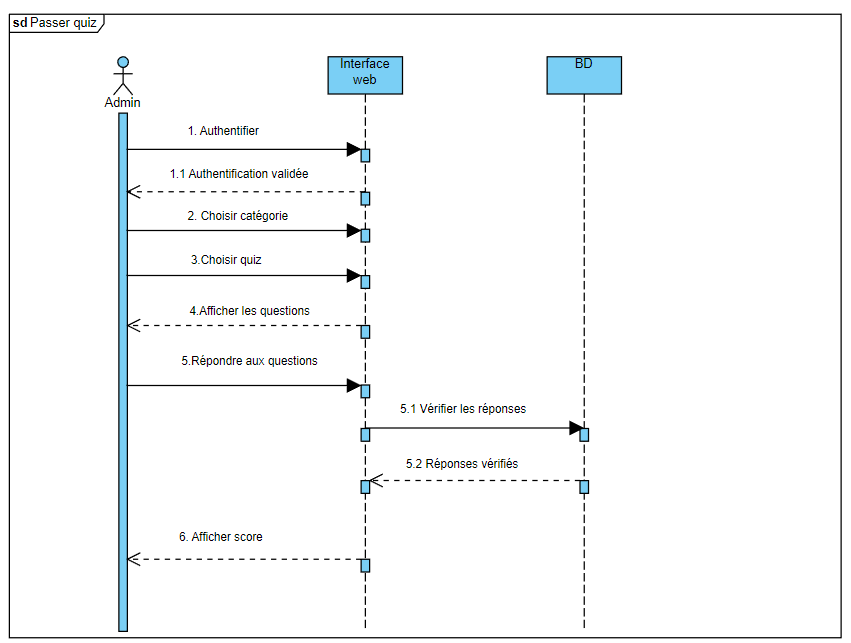
### 2. Le diagramme de séquences d’ajout d’un quiz



*Figure 4 : Diagramme de séquence d’ajout d’un quiz*

* L’administrateur accède à l’interface web,
* L’administrateur s’authentifie,
* L’administrateur demande d’ajouter un quiz,
* Le système envoie le formulaire d’ajout,
* L’administrateur enregistre les données ajoutes,
* Le système envoie ces derniers au BD pour vérifier l’existence de quiz,
* Si le quiz existe déjà il n’aura pas d’enregistrement sinon il va être ajouter.

### 3. Le diagramme de séquence de passer un quiz



*Figure 5 Diagramme de séquence de consultation de passer un quiz*

* L’administrateur accède à l’interface web,
* L’administrateur s’authentifie,
* L’administrateur demande au système de consulter l’historique de réservation,

## Conclusion

Nous venons de terminer la partie de conception. Dans le chapitre suivant nous allons aborder la dernière partie qui représente la partie réalisation de notre projet en se basant sur les mécanismes et les solutions déterminés dans la phase de conception.

# Chapitre 4 : Réalisation

## Introduction

Après avoir affecter l'étude et la conception de notre application, nous passons à la phase d'implémentation. Ce chapitre présente le résultat du travail effectué durant ce projet. Nous allons présenter, aussi, l'environnement matériel et les outils de développement utilisés. Nous clôturons ce chapitre par quelques captures d'écran montrant les fonctionnalités de notre application.

## Environnement logiciel :

Les outils utilisés pour le développement de ce site web sont :

**Eclipse IDE :** est un environnement de développement intégré libre (le terme Eclipse désigne également le projet correspondant, lancé par IBM) extensible, universel et polyvalent, permettant potentiellement de créer des projets de développement mettant en œuvre n'importe quel langage de programmation. Eclipse IDE est principalement écrit en Java (à l'aide de la bibliothèque graphique SWT, d'IBM), et ce langage, grâce à des bibliothèques spécifiques, est également utilisé pour écrire des extensions.



Figure : Eclipse

**WampServer** : WampServer est une plate-forme de développement Web sous Windows pour des applications Web dynamiques à l’aide du serveur Apache2, du langage de scripts PHP et d’une base de données MySQL. Il possède également PHPMyAdmin pour gérer plus facilement vos bases de données.

****

Figure : WampServer

**Postman :** permet de construire et d’exécuter des requêtes HTTP, de les stocker dans un historique afin de pouvoir les rejouer, mais surtout de les organiser en Collections. Cette classification permet notamment de regrouper des requêtes de façon « fonctionnelle ».



Figure : Postman

**MySql** :est un logiciel de gestion et d'administration de bases de données MySQL créé en 2004. Via une interface graphique intuitive, il permet, entre autres, de créer, modifier ou supprimer des tables, des comptes utilisateurs, et d'effectuer toutes les opérations inhérentes à la gestion d'une base de données. Pour ce faire, il doit être connecté à un serveur MySQL.



Figure : workbench

## II .Technologies utilisées :

Cette rubrique montre la définition des choix techniques utilisés dans le développement du travail :

**Spring Boot** : est un framework qui facilite le développement d'applications fondées sur Spring en offrant des outils permettant d'obtenir une application packagée en JAR, totalement autonome.



Figure : Logo SpringBoot

**Angular :** est un framework Javascript côté client qui permet de réaliser des applications Cross-platform Web, Mobilie et Desktop.

Angular est Open Source et développé par Google. Il utilise l’architecture **MVM** (Modèle Vue Modèle), proche du modèle MVC. Le code source d’Angular est écrit en TypeScript. Le TypeScript est une couche supérieure au javascript développé par Microsoft qui se compile en JavaScript simple.



Figure : Logo Angular

[**MySQL**](https://www.mysql.com/) est un système de gestion de bases de données relationnelles SQL open source développé et supporté par Oracle.

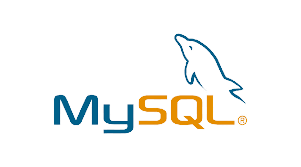
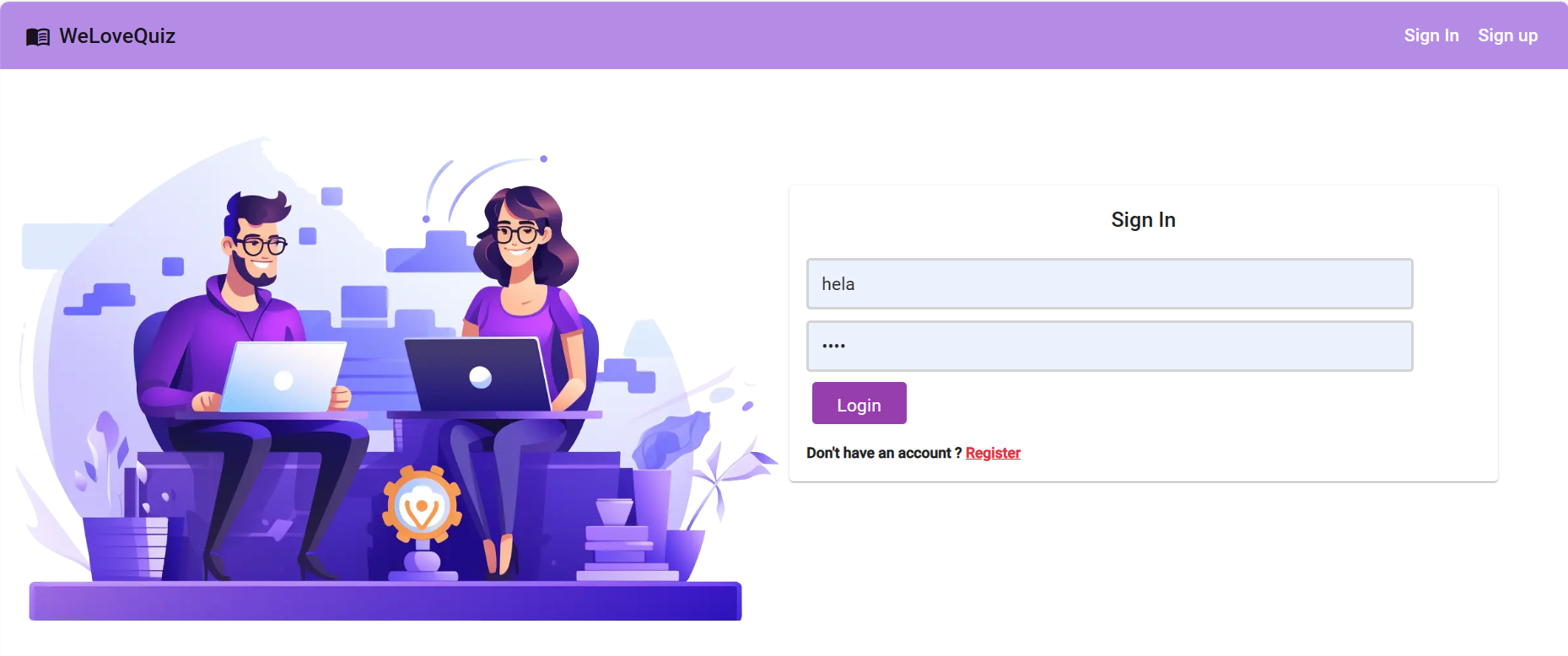


Figure : MySQL

## Démonstration des interfaces

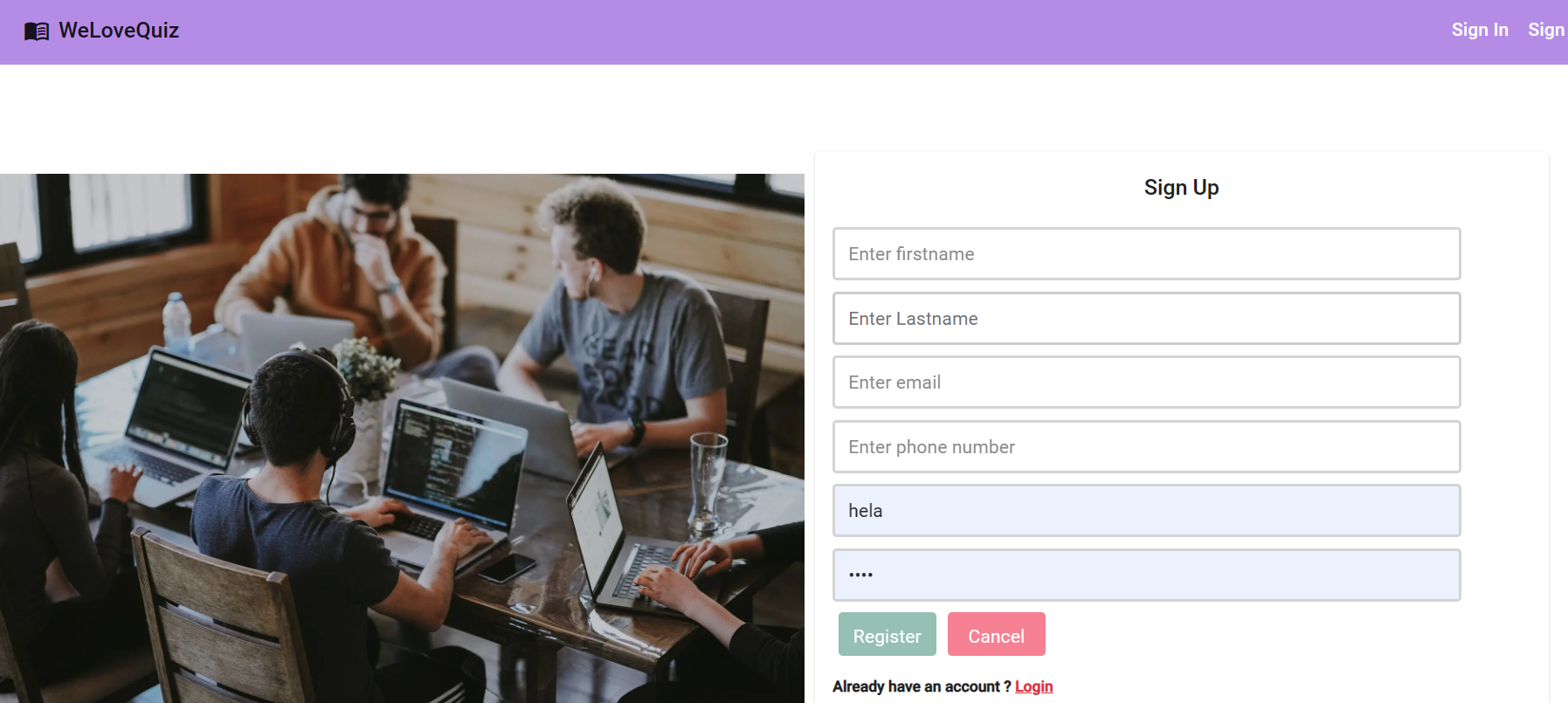
1. **Authentification**

****

*Figure 7 : Page Login*

La page d’authentification de l’administrateur et l’utilisateur. Il faut avoir un compte pour connecter sinon l’utilisateur peut s’inscrire.

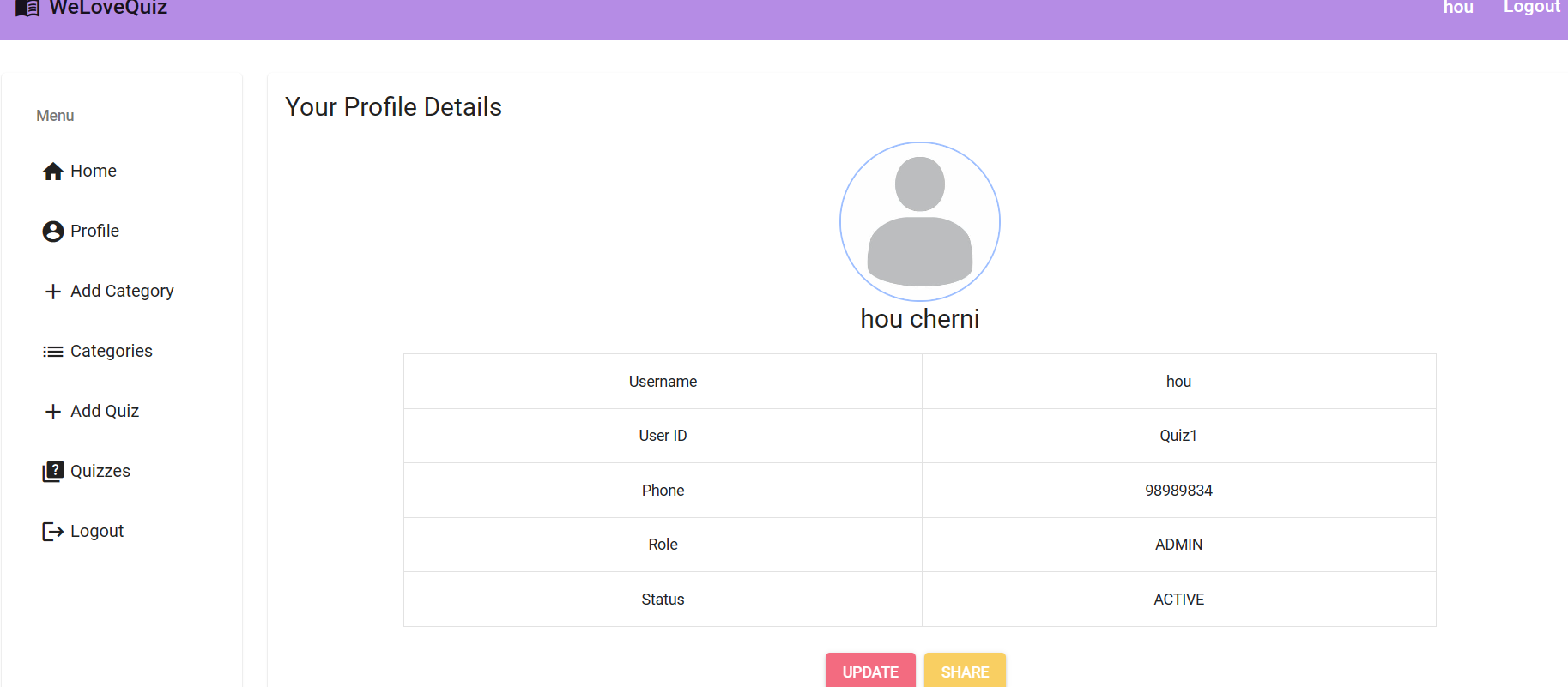
1. **Inscription**

****

*Figure 8 : Page Inscription*

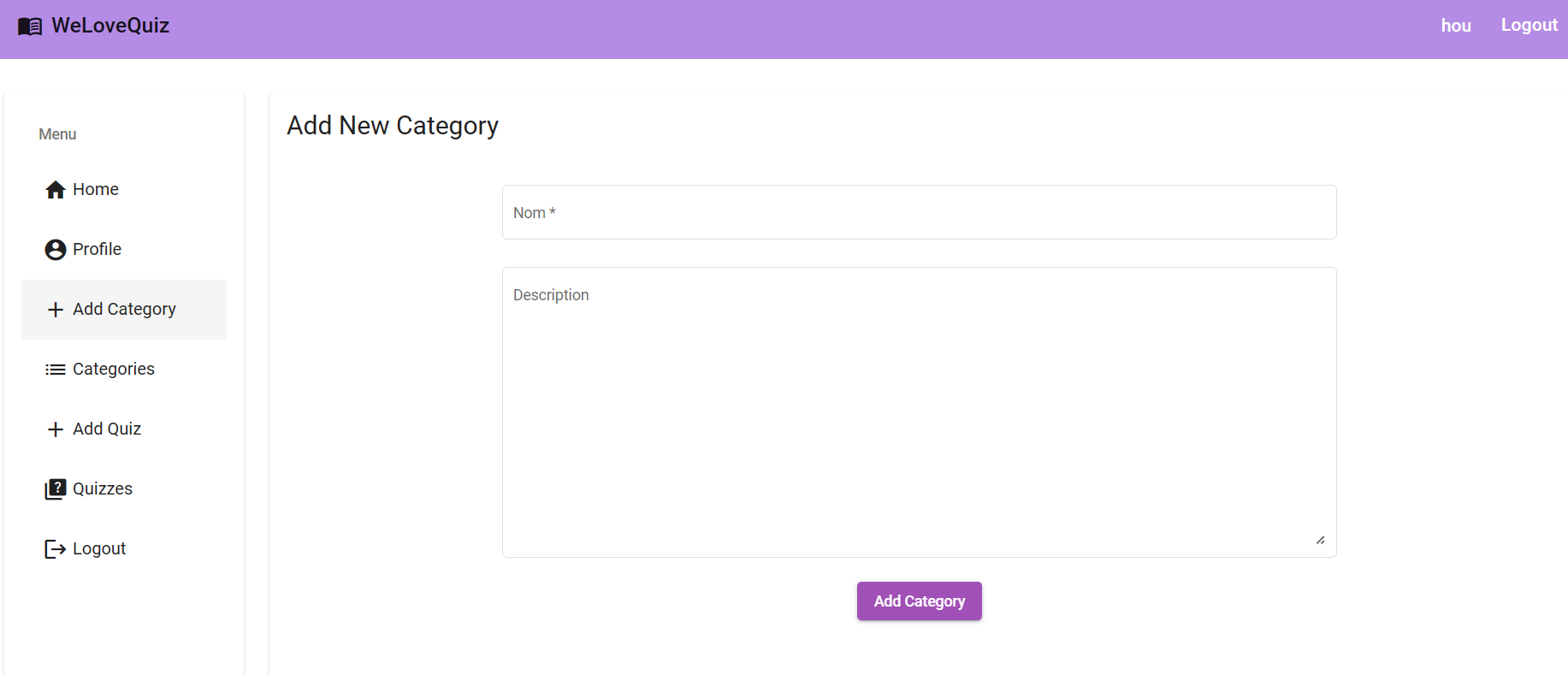
La page de l’inscription de l’utilisateur.

### 3.Partie administrateur



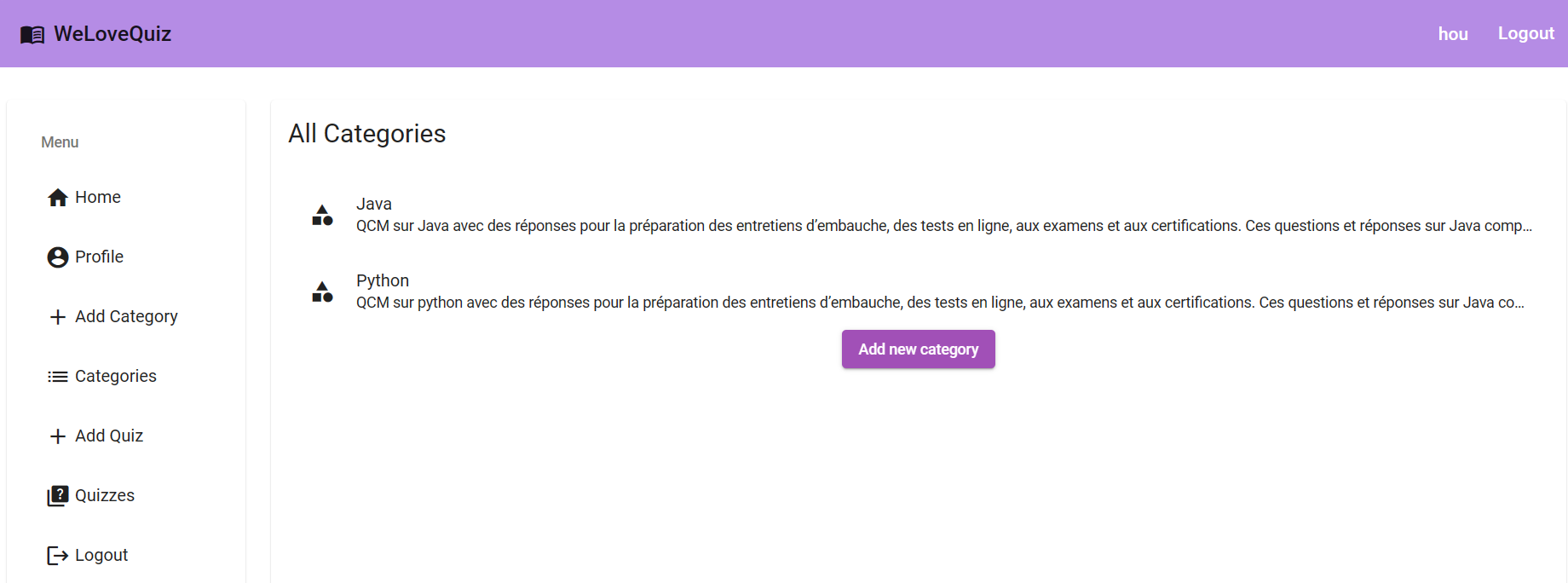
*Figure 9 : Profile de l’administrateur*

Cette interface représente le profil de l’administrateur.



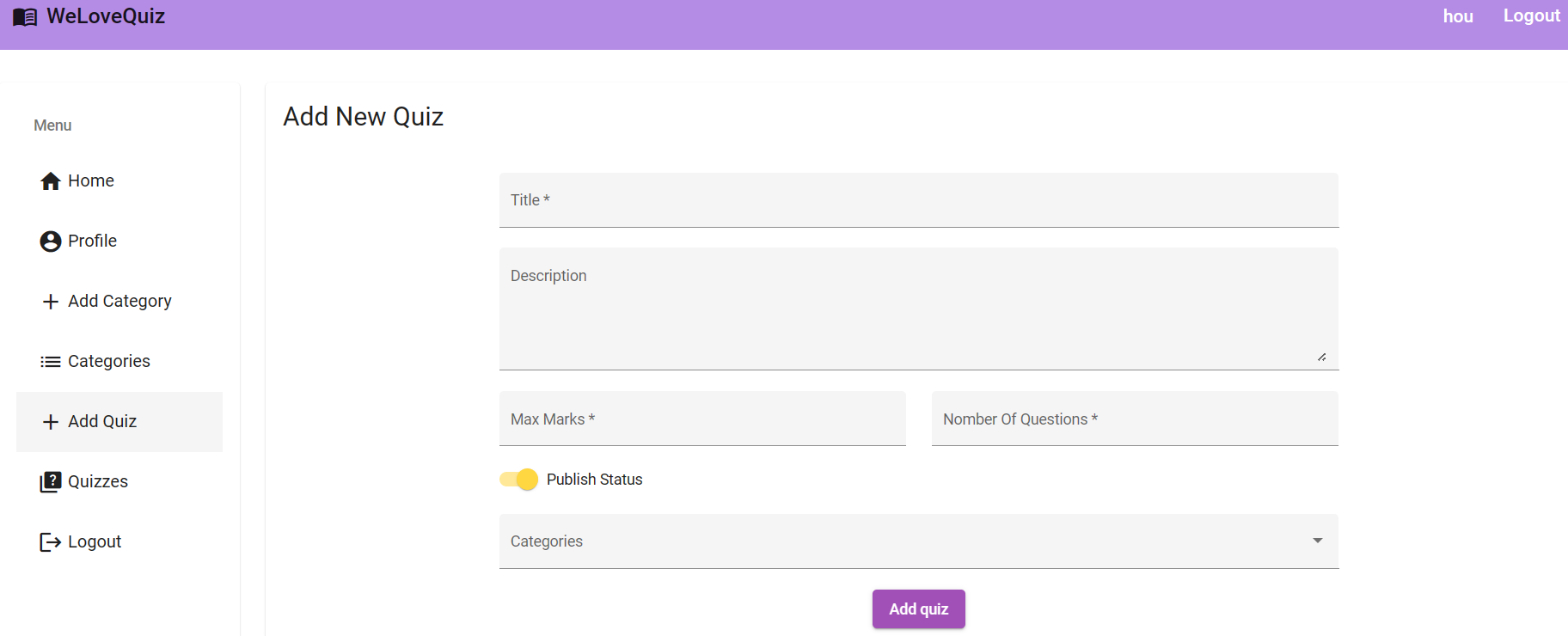
*Figure 10 : L’ajout d’un category*

Cette interface représente la page de l’ajout d’une nouvelle catégorie.



*Figure 11 Liste des catégories*

Cette interface représente la page de toutes les catégories disponibles.



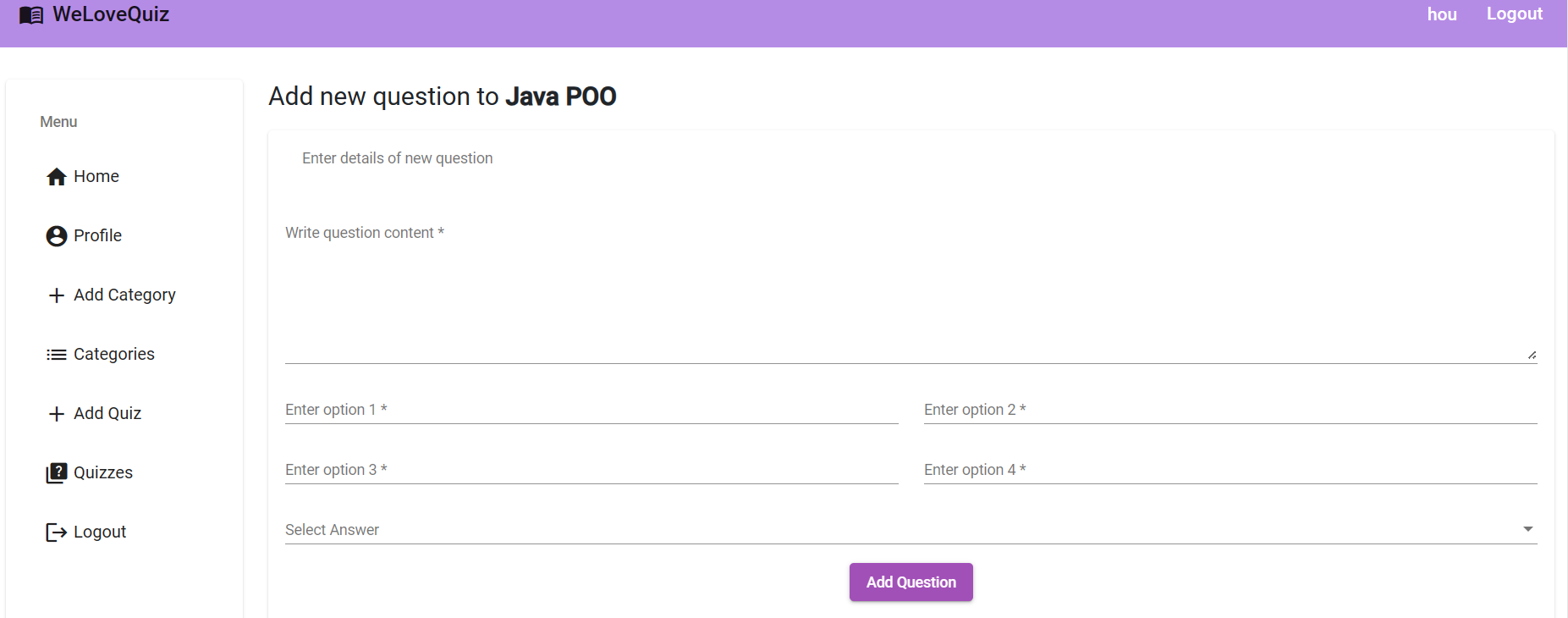
*Figure 12 L’ajout d’un quiz*

Cette interface représente la page de l’ajout d’un nouveau quiz.

****

*Figure 13 Les quizes*

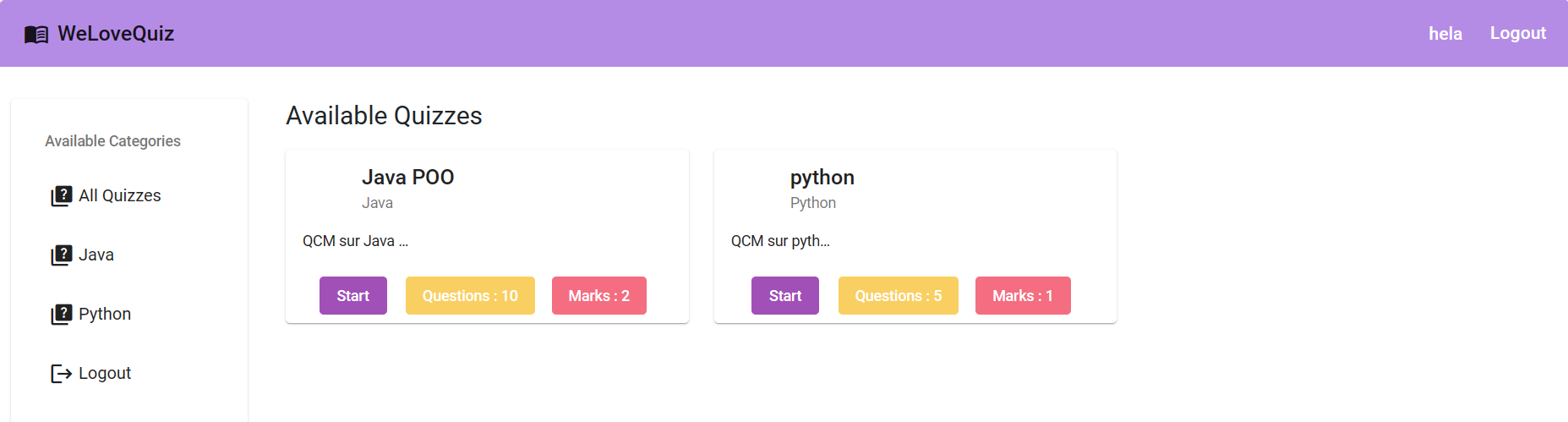
Cette interface représente la page qui affiche tous les quiz.



*Figure 14 : L’ajout d’une question*

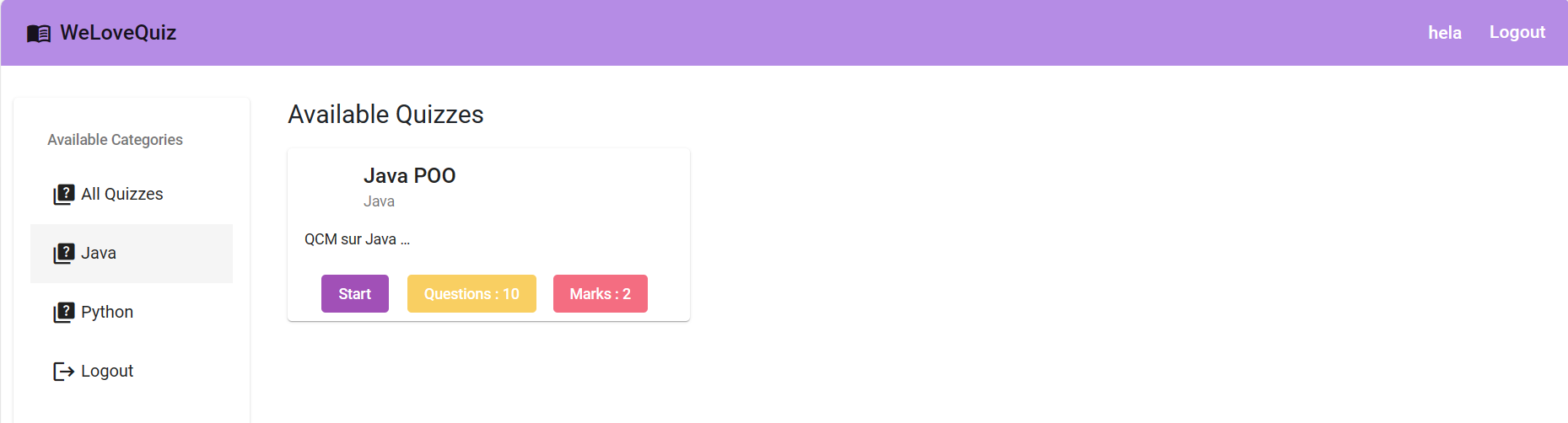
Cette interface représente la page de l’ajout d’une nouvelle question.

### Partie Etudiant



*Figure 15 L’authentification d’étudiant*

Cette interface représente la page de l’affichage des quizz disponibles.

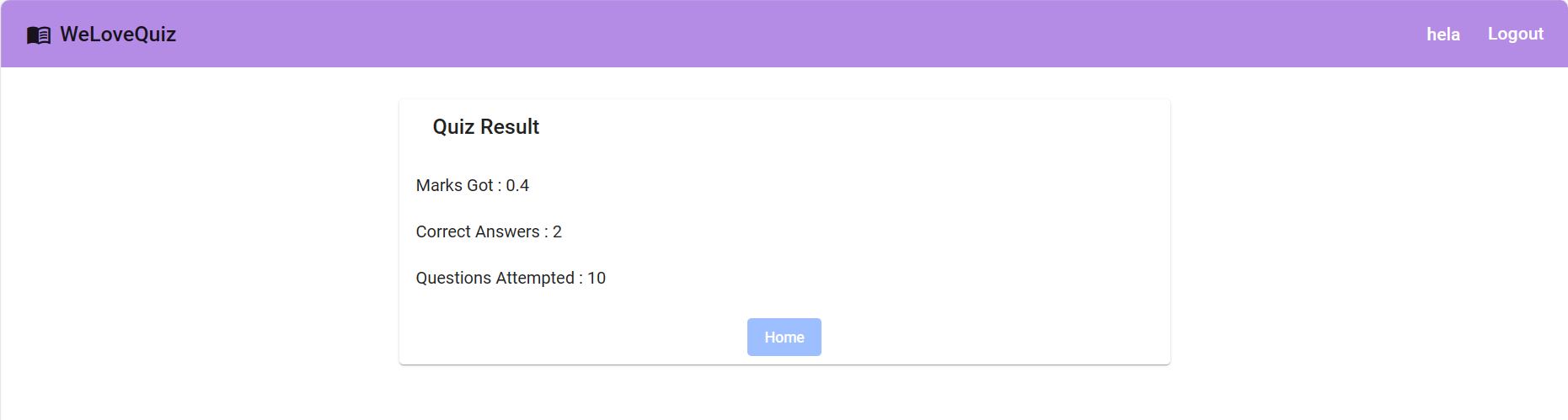


*Figure 16 : Exemple d’un quiz*

Cette interface représente la page du quiz choisie.



*Figure 17 : Exemple d’un quiz Java*

****

*Figure 17 : Résultat du quiz*

# Conclusion générale

L’objectif de notre travail était de développer une application tout en suivant une démarche spécifique. Notre problématique de départ était de pouvoir réaliser cette

application en dépit de toutes les contraintes. La réalisation de ce travail nous a donné

l’occasion d’acquérir de nouvelles connaissances, et surtout de mettre en pratique les connaissances théoriques et techniques acquises durant notre cursus universitaire en Conduite de Projets.

**webographie**

1. [**https://www.jablotron.cloud/2021/02/18/docker-compose-in-gitlab-ci/**](https://www.jablotron.cloud/2021/02/18/docker-compose-in-gitlab-ci/)

1. [**https://fr.slideshare.net/safwenbenfredj/rapport-de-projetdefindtudespfesafwen-8**](https://fr.slideshare.net/safwenbenfredj/rapport-de-projetdefindtudespfesafwen-8)

1. [**https://openclassrooms.com/fr/courses/2035766-optimisez-votre-deploiement-en-creant-des-conteneurs-avec-** **docker/6211517-creez-votre-premier-dockerfile**](https://openclassrooms.com/fr/courses/2035766-optimisez-votre-deploiement-en-creant-des-conteneurs-avec-docker/6211517-creez-votre-premier-dockerfile)

1. [**https://www.syloe.com/glossaire/gitlab-ci/#:~:text=gitlab%2Dci.-,yml,que%20doit%20ex%C3%A9cuter%20GitLab%**](https://www.syloe.com/glossaire/gitlab-ci/#:~:text=gitlab%2Dci.-,yml,que%20doit%20ex%C3%A9cuter%20GitLab%20CI)

[**20CI.**](https://www.syloe.com/glossaire/gitlab-ci/#:~:text=gitlab%2Dci.-,yml,que%20doit%20ex%C3%A9cuter%20GitLab%20CI)

**https://fr.theastrologypage.com/intellij-idea**

**https://blog.webnet.fr/presentation-de-postman-outil-multifonction-pour-api-web/**