



UNIVERSITÉ MOHAMMED V
ECOLE NATIONALE SUPERIEURE
D'INFORMATIQUE ET ANALYSE DES SYSTEMES
ENIAS - Rabat



Projet de JEE

Genie Logiciel - GL -

Développement d'une plateforme de bricolage en ligne - Titsuite

Réalisé par :

EZZAOUI RAHALI Hamza
HAJAZI Soufiane
JAMAI Mohammed Amine
KAHLAOUI Ismail

Encadré par :

Mr EL HAMLAOUI Mahmoud

Soutenu le Dimanche 07 Mars 2021

Année universitaire : 2020-2021

Remerciements

Avant d'entamer tout développement de cette expérience fructueuse, nous aimerions tout d'abord exprimer notre gratitude à un nombre de personnes qui ont contribué à la réalisation et au bon déroulement de notre projet. Nous commençons ainsi par présenter nos remerciements à l'Ecole Nationale supérieure d'informatique et d'analyse des systèmes qui ne cesse de prôner de telles expériences afin d'assurer une meilleure intégration de ses lauréats dans le milieu professionnel.

Par la même occasion, nous tenons à remercier profondément notre encadrant **Monsieur. MAHMOUD EL HAMLAOUI** qui a su se rendre disponible pour nous aider quand le besoin s'en faisait ressentir.

Table de matières

Remerciements	1
Table des figures	4
1 Présentation de l'application	5
1.1 Problématique :	5
1.2 Description de l'application	5
1.2.1 Besoins fonctionnels	6
Les clients	6
Les auto-entrepreneurs	6
Les Administrateurs	7
2 Etude et conception	8
2.1 Diagramme de cas d'utilisation	8
2.2 Diagrammes de séquence	9
2.2.1 Authentification	10
2.2.2 Postuler sa candidature	11
2.2.3 Gestion des offres	11
2.2.4 Création des annonces	12
2.3 Diagramme de classe	13
3 Réalisation et Développement	14
3.1 Méthodologie de travail	14
3.1.1 Github	14
3.1.2 Scrum (weekly sprints)	14
3.2 Backend	14
3.2.1 API TitSuite V 1.0	14
JAX-RS	14
Base de données	15
Authentification et Autorisation	15
Les routes	16
3.3 Frontend	17
3.3.1 Réalisation de la maquette -WIRE FRAMING-	17
3.3.2 Réalisation du siteweb	22

4 Tests et Déploiement	23
4.0.1 Tests	23
4.0.2 Déploiement	23
Conclusion générale	25

Table des figures

2.1	Diagramme de cas d'utilisation	9
2.2	Diagramme de séquence : Authentification	10
2.3	Diagramme de séquence : Postuler sa candidature	11
2.4	Diagramme de séquence : Gestion des offres	11
2.5	Diagramme de séquence : Création des annonces	12
2.6	Diagramme de classes	13
3.1	Page d'accueil	18
3.2	Page d'authentification	18
3.3	Page d'enregistrement	19
3.4	Page des annonces	19
3.5	Page d'accueil du Freelancer	20
3.6	Page d'accueil Client et ajout d'offre	20
3.7	Profile du freelancer	21
4.1	Dockerfile	24

Chapitre 1

Présentation de l'application

Le projet en cours d'étude consiste à développer une plateforme qui va permettre aux professions libérales et auto-entrepreneurs d'exposer leurs services aux particuliers.

1.1 Problématique :

Notre solution est née du constat suivant : pour créer de la valeur dans un pays il faut connecter les gens. En d'autres mots connecter l'offre avec la demande. Car nous croyant que chacun de nous porte en lui une valeur ajoutée qu'il est capable de délivrer si on lui offre l'opportunité.

Grâce à cette plateforme nos besoins ne vont plus prendre des jours, voire des mois pour être réalisées car nous réussirons en quelques cliques à travers la plateforme à trouver un prestataire compétent et proche de nous.

1.2 Description de l'application

Notre mission porte sur la création d'une application web qui va permettre d'implémenter cette solution avec la manière la plus optimale possible tout en exploitant les technologies offertes par l'environnement JEE.

Identification des acteurs :

- CLIENT.
- AUTO-ENTREPRENEUR .
- ADMINISTRATEUR.
- RESPONSABLE SERVICE CLIENT .

1.2.1 Besoins fonctionnels

Les clients

L'application Titsuite va permettre au client de :

- DE GÉRER SON COMPTE.
- CONSULTER LES OFFRES DISPONIBLES SUR LE MARCHÉ.
- EFFECTUER SES PAIEMENTS EN LIGNE.
- LANCER UNE OFFRE D'EMPLOI.
- DONNER UNE NOTE/AVIS CONCERNANT LE SERVICE D'UN AUTO-ENTREPRENEUR.
- AJOUTER UNE RÉCLAMATION.
- CHATTER EN DIRECT AVEC LE SERVICE CLIENT.

Les auto-entrepreneurs

L'application Titsuite va permettre aux auto-entrepreneurs de :

- GÉRER SON PROPRE PROFILE.
- CANDIDATER À UN JOB.
- ACCEPTER OU REFUSER UNE OFFRE.
- CHATTER EN DIRECT AVEC LE SERVICE CLIENT.
- SIGNALER UN CLIENT.
- GÉRER SON PORTEFEUILLE.

Les Administrateurs

- GÉRER LES UTILISATEURS.
- GÉRER LES PAIEMENTS.
- VALIDER/REJETER LES ANNONCES/OFFRES D'EMPLOI SUR LA PLATEFORME.

Chapitre 2

Etude et conception

Ce chapitre porte sur les différents diagrammes et méthodes utilisés pour concevoir notre système, plus précisément les diagrammes offerts par le langage UML.

2.1 Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation nous offre la possibilité de spécifier les besoins de nos acteurs (Client, Freelancer, etc).

Chaque utilisateur de l'application doit s'authentifier avant d'accéder aux différents services présentés dans le diagramme suivant.

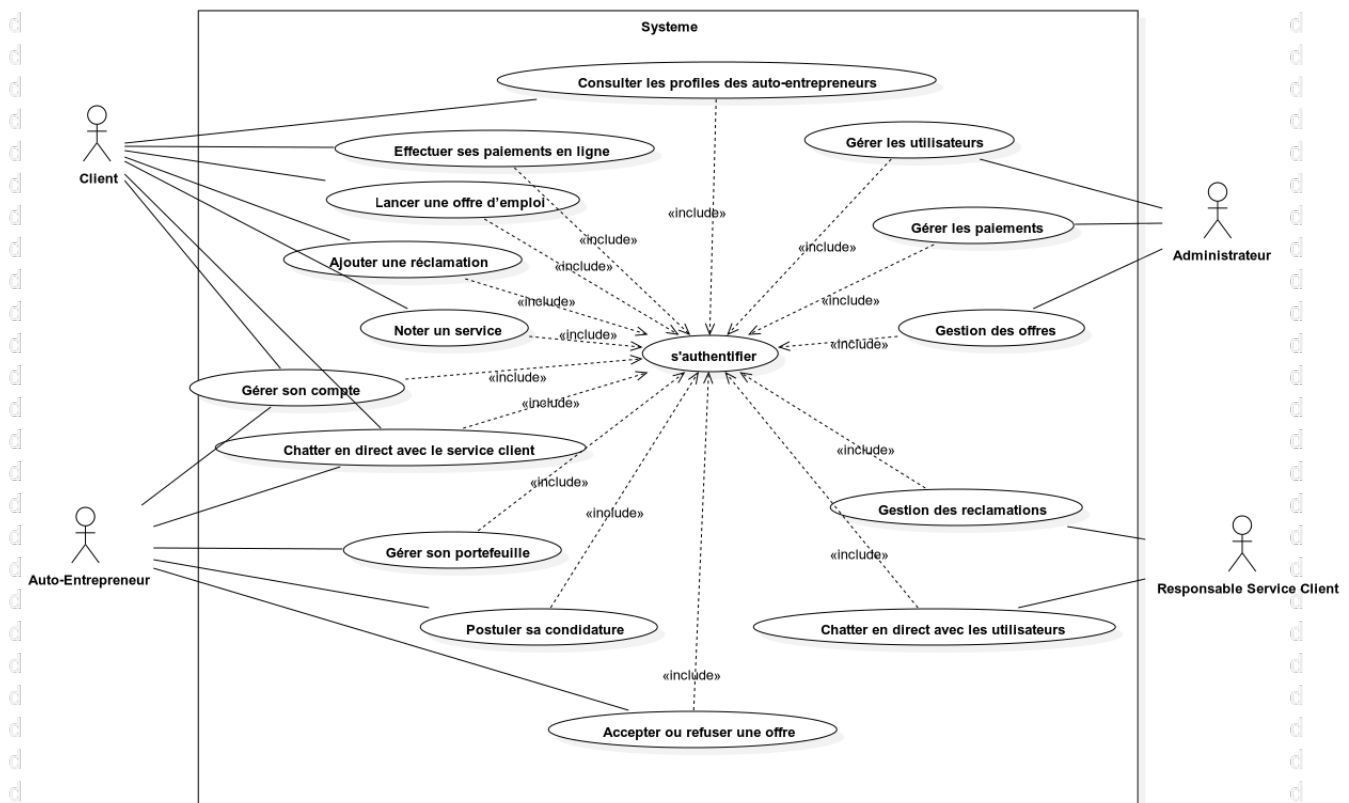


FIGURE 2.1 – Diagramme de cas d'utilisation

2.2 Diagrammes de séquence

Les diagrammes de séquence ont pour but de détailler les différentes interactions entre les entités externes et le système.

Plusieurs diagrammes ont été établis et peuvent être retrouvés dans le répertoire du projet. Nous mentionons les plus intéressants par la suite.

2.2.1 Authentification

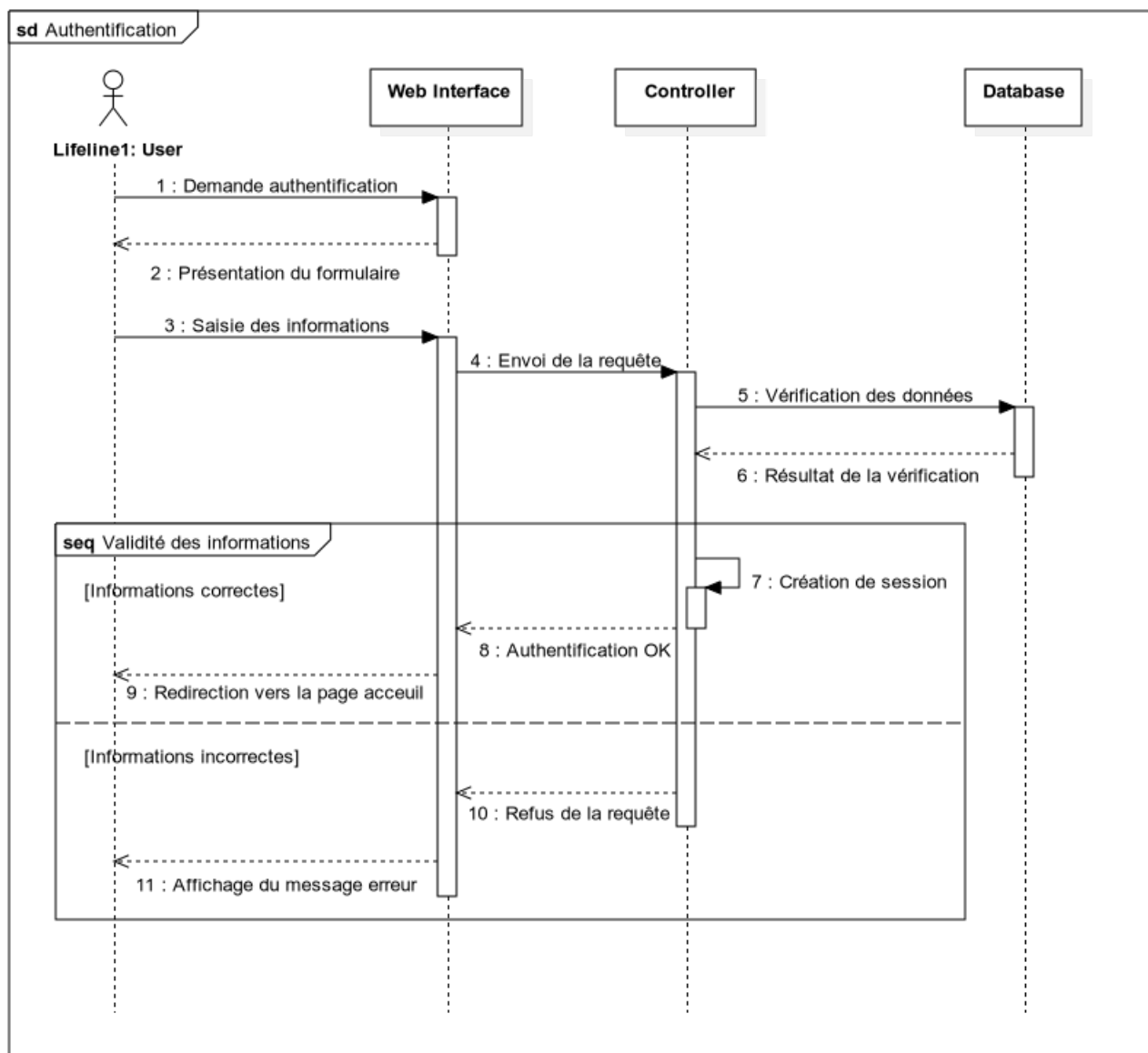


FIGURE 2.2 – Diagramme de séquence : Authentification

2.2.2 Postuler sa candidature

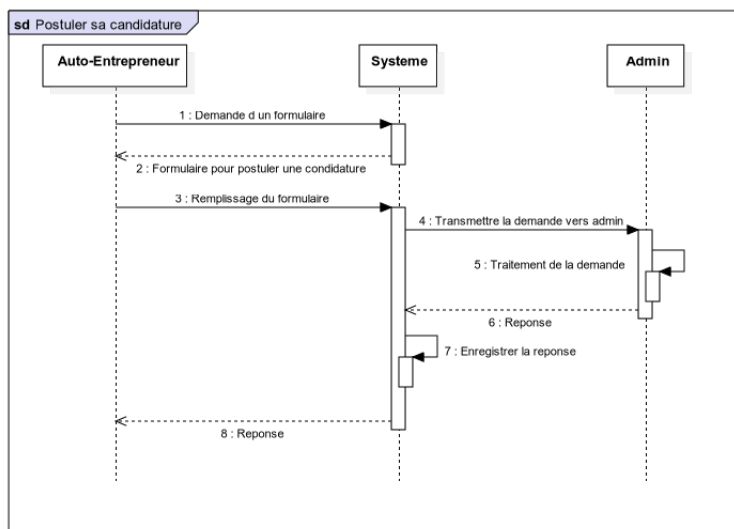


FIGURE 2.3 – Diagramme de séquence : Postuler sa candidature

2.2.3 Gestion des offres

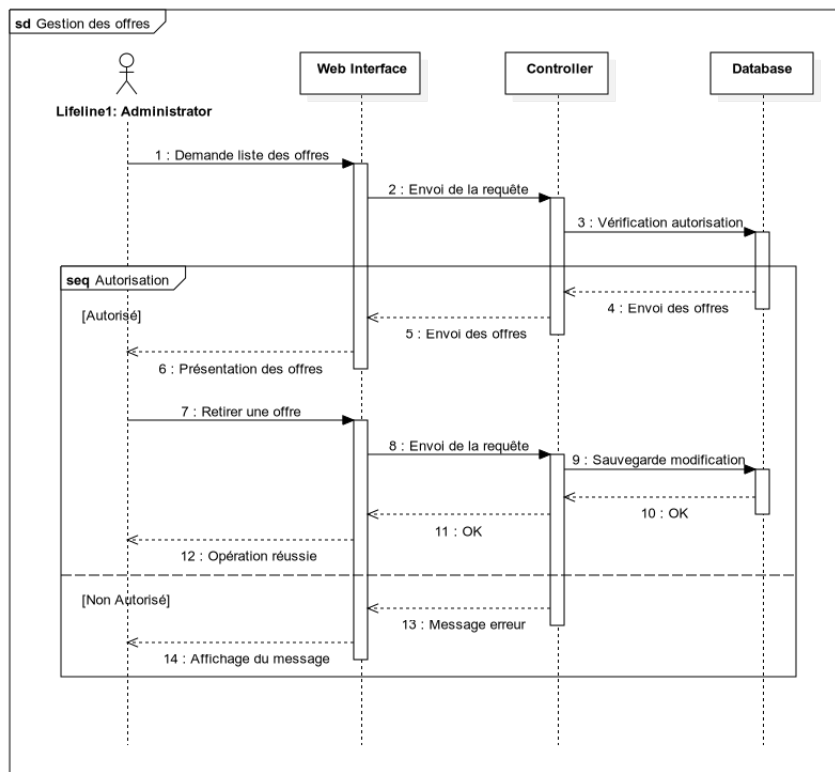


FIGURE 2.4 – Diagramme de séquence : Gestion des offres

2.2.4 Création des annonces

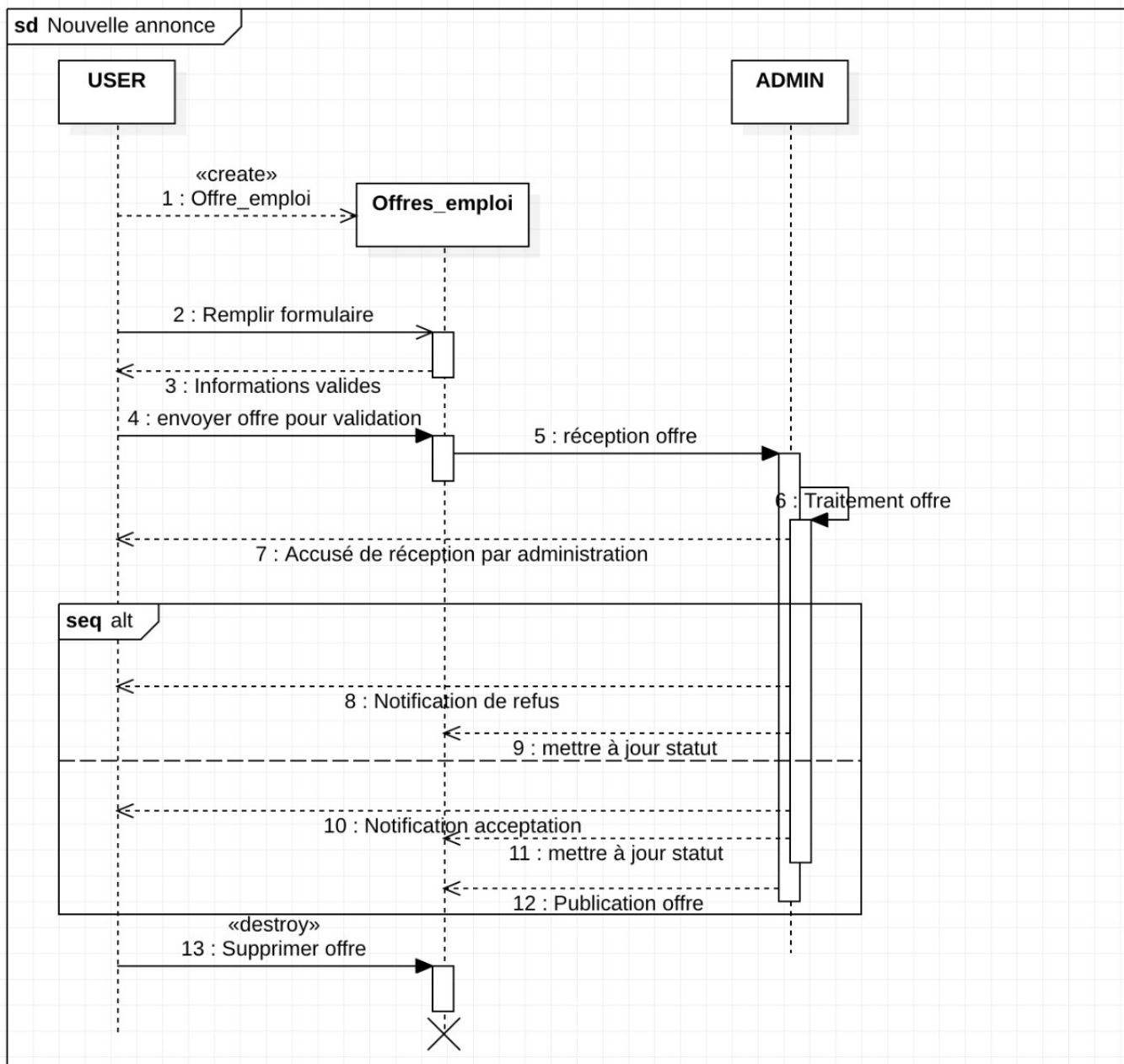


FIGURE 2.5 – Diagramme de séquence : Création des annonces

2.3 Diagramme de classe

Le diagramme de classe permet de visualiser les différentes relations entre les classes du système ainsi que leurs cardinalités.

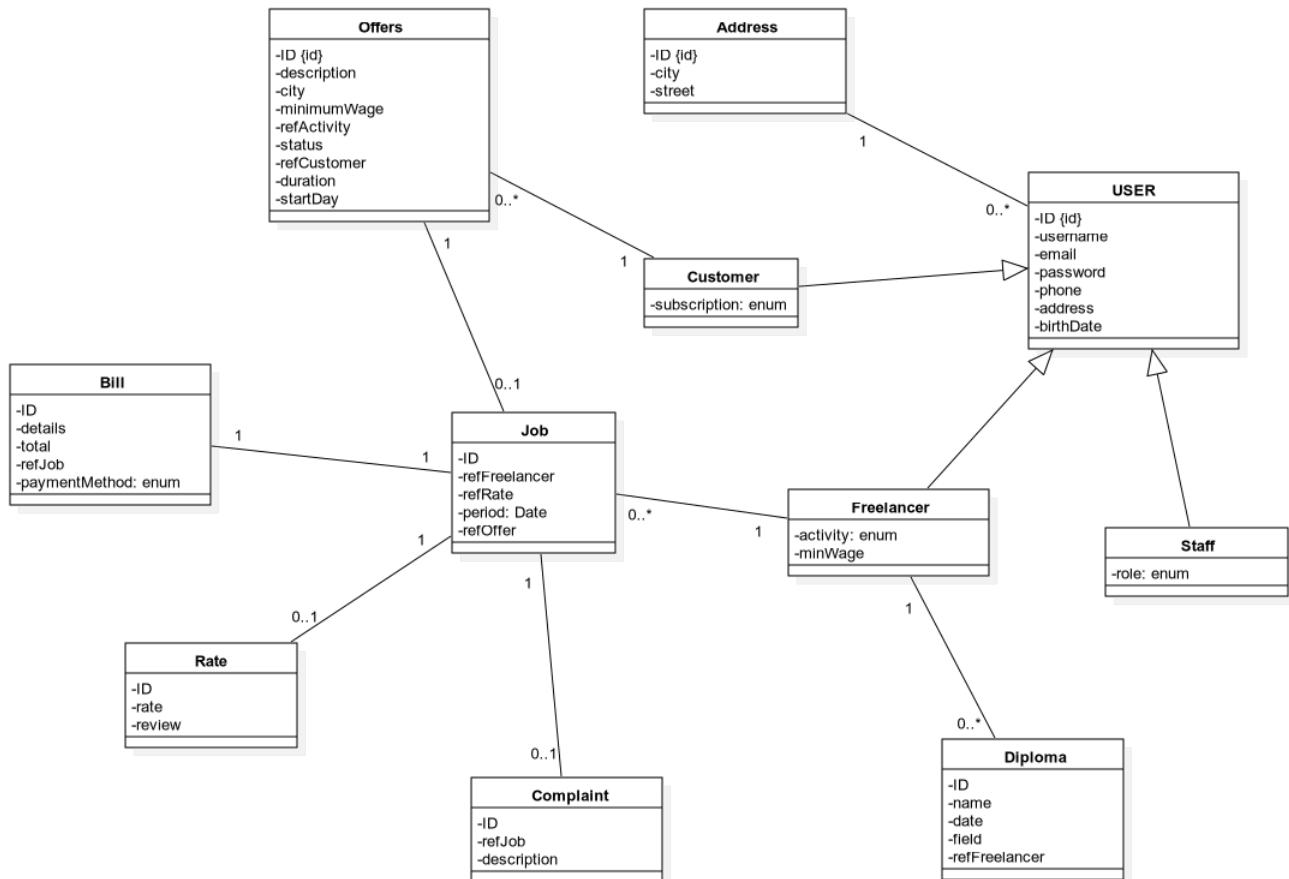


FIGURE 2.6 – Diagramme de classes

Chapitre 3

Réalisation et Développement

3.1 Méthodologie de travail

3.1.1 Github

Pour mieux s'organiser au niveau de la phase de conception et développement, on a commencé par tracer notre tableau de bord Kanban au niveau du repertoire public Titsuite vous trouverez le lien ci-dessous.

3.1.2 Scrum (weekly sprints)

Afin de mieux maitriser le processus de développement nous avons opté pour la méthodologie scrum, qui a été au coeur de notre travail.

Les réunions hebdomadaires avaient pour objectif de discuter l'état d'avancement du projet.

3.2 Backend

3.2.1 API TitSuite V 1.0

JAX-RS

Afin de construire notre propre API qui va nous servir comme interface pour communiquer facilement avec notre base de données oracle, nous avons fait le choix de l'API JAX-RS qui facilite le développement de services Web REST à partir de la plateforme de développement Java.

JAX-RS permet d'implémenter des ressources sous la forme de composants Java EE. Une classe représentant une ressource est identifiée grâce à l'annotation `@Path`.

Base de données

Comme base de données nous avons choisi le SGBD Oracle Database, qui a été d'une importance centrale à la de gestion des données relatives à notre application.

La connexion à notre base de données se fait à partir du driver JDBC oracle, tout en implémentant le pattern DAO ainsi qu'une Connection Factory.

Authentification et Autorisation

Vu l'état RESTful du serveur que nous avons créé, nous avons opté pour les JSON Web Token comme technique d'authentification.

- Utilisation et Mesures :

Les « JSON Web Token » ou JWT sont des jetons générés par un serveur lors de l'authentification d'un utilisateur sur une application Web, qui sont ensuite transmis au client. Evidemment, les informations contenues sont signées à l'aide d'une clé privée détenue par le serveur.

Aussi, pour éviter les types d'attaques XSS, un jeton expirable (15 minutes) est envoyé autant que HTTPOnly cookie, ce qui l'empêche d'être récupéré par un script JS.

Finalement pour contrer les attaques du types "Man in the middle", il faut s'assurer que la communication se fasse sur un canal HTTPS à fin d'éviter le sniffing du JWT.

- Rafraichissement du token :

Le JWT (et le cookie) est configuré de telle façon à s'expirer automatiquement après 15 minutes de sa creation. Pour éviter donc de demander à chaque fois à l'utilisateur de se reconnecter, nous avons implémenté un "Refresh Token" qui sert à rafraichir automatiquement le JWT lors de l'utilisation de l'application Web.

Le "Refresh Token" dure plus longtemps qu'un jeton normal (expiration après une semaine) mais il est aussi plus sécurisé et stocké dans la base de donnée.

- Rôles et Autorisation :

Le JWT que nous créons contient des informations relatives à l’ID et le rôle de chaque utilisateur. Ces informations sont décryptées et vérifiées avant le traitement de toute requête qui en a besoin. (Le client seul peut créer une offre, le freelancer seul peut accepter un job, etc.)

Les routes

- Afficher toutes les annonces sur le site :

`titsuite/api/offers/all`

- Afficher certaines annonces :

GET : `titsuite/api/offers/city={ville}`

GET : `titsuite/api/offers/id={ID}`

GET : `titsuite/api/offers/activity={activity}`

GET : `titsuite/api/offers/city={ville}`

GET : `titsuite/api/myoffers` : affiche les annonces publiés par le client connecté.

- Créer une annonce :

POST : `titsuite/api/offers/new`

- Mettre à jour une annonce :

POST : `titsuite/api/offers/update/ref={id}`

- Afficher l’historique des missions accomplies par le freelancer

GET : `titsuite/api/myjobs/all`

- Afficher l’historique de toutes les missions accomplies par les freelancer

GET : `titsuite/api/jobs/all`

GET : `titsuite/api/jobs/refFreelancer={id}`

- Création d’un compte :

POST : `titsuite/api/users/register` : crée un nouveau utilisateur et envoi un mail de confirmation à l’adresse mentionnée.

- Activation d'un compte :

GET : `titsuite/api/users/activateAccount?role={role}&email={email}&code={code}` : reçoit une requête GET et vérifie le compte ciblé si les informations sont correctes.

- Authentification :

POST : `titsuite/api/users/login` : renvoie un HTTPOnly cookie contenant le jeton si les informations sont correctes et le compte est vérifié.

- Rafraichissement du token :

POST : `titsuite/api/users/refreshToken` : envoie un nouveau jeton valide et expirable si les informations sont correctes.

- Déconnexion :

POST : `titsuite/api/users/logout` : envoie un cookie vide pour supprimer le token du navigateur et rend nulle le Refresh Token de façon asynchrone dans la base de données.

- Afficher les données du profile :

GET `titsuite/api/users/profile` : envoie les informations relatives au profile de l'utilisateur valide

GET `titsuite/api/diplomas/get?id={id}` : renvoie les informations du diplôme ciblé.

- Modifier les données du profile :

POST `titsuite/api/users/profile/update` : met à jour les informations du profile d'un utilisateur valide.

POST `titsuite/api/diplomas/update` : met à jour les informations du diplôme.

3.3 Frontend

3.3.1 Réalisation de la maquette -WIRE FRAMING-

Afin d'avoir une vue globale sur le systeme conçu, on a entamé la réalisation du Frontend par la création des maquettes.

Outils utilisés : Adobe XD.



FIGURE 3.1 – Page d'accueil

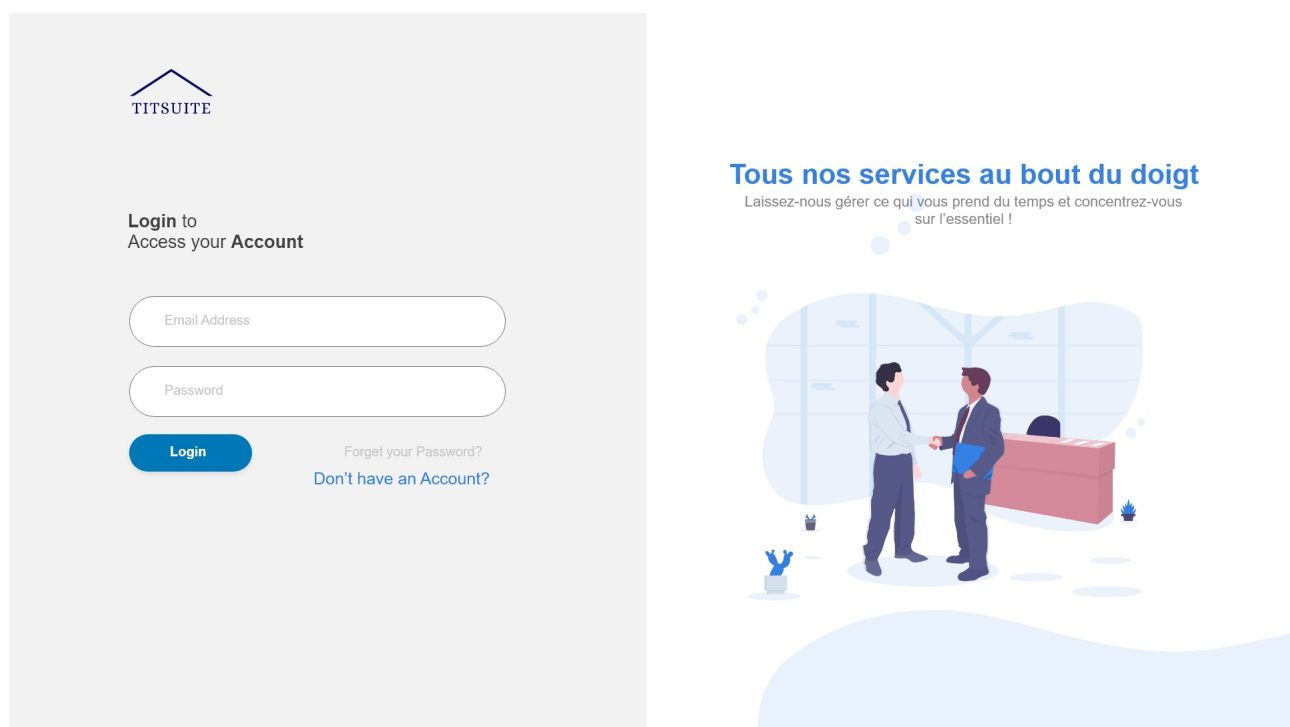
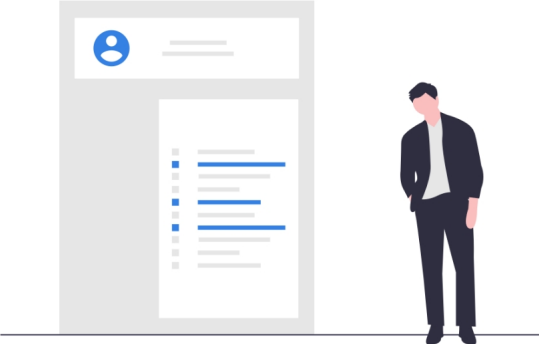


FIGURE 3.2 – Page d'authentification

Tous nos services au bout du doigt



Complete the form below

Email

Password

Repeat your password

Address

City

Postal code

☐ I'm a client, and I have a job that needs to be done.
☐ I'm a freelancer, and I want to showcase my services.

Sign up

FIGURE 3.3 – Page d'enregistrement

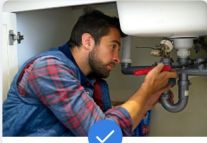
TITSUITE

- Dashboard
- Mes Annonces
- Service client
- Portfeuille
- Historique
- Paramètres

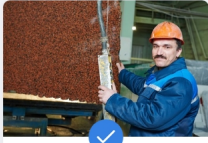
Hamid aarif

© 2021 Titsuite Inc. Tous droits réservés


382 services available




Plomberie Allal
Visible sur Rabat, Salé.
+6 Expériences 300 MAD



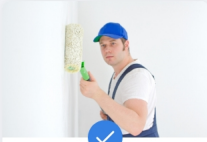
Marbrerie professionnel
Visible sur Casablanca.
2 Expériences 250 MAD



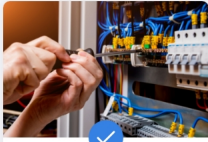
Réparation électroménager
Temporairement indisponible.
0 Expérience 350 MAD



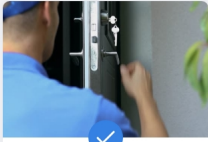
Femme de ménage
Visible sur Salé.
3 Expériences 400 MAD



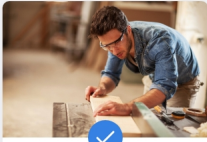
Peintre en bâtiment
Visible partout.
Une Expérience 300 MAD



GNS - ELEC
Visible partout.
+17 Expériences 450 MAD



Serrurier chez Abdellah
Visible sur Rabat, Témara.
+11 Expériences 250 MAD



Menuiserie tous types
Visible partout.
2 Expériences 200 MAD

< 1 2 3 4 ... 11 12 >

FIGURE 3.4 – Page des annonces

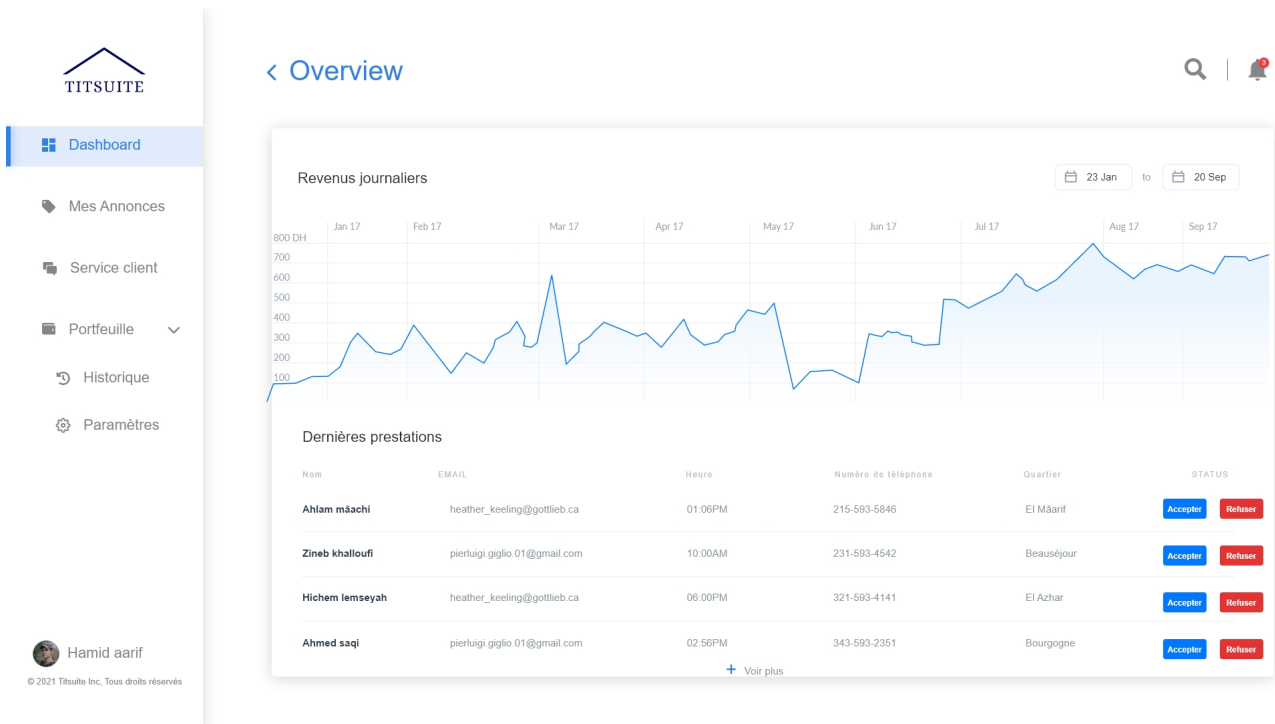


FIGURE 3.5 – Page d'accueil du Freelancer

Add a new offer

[Add](#)

Description

Type your offer description here....

City

Casablanca

Start Date

28/10/2021

Activity

Marberie

Minimum wage

\$ 3000DH

My offers

ID	Description	Activity	City	Minimum wage	Start Date	Payer
4			Rabat			
5			Casablanca			
6			Fés			

+ Voir plus

FIGURE 3.6 – Page d'accueil Client et ajout d'offre

TITSUITE

Dashboard

Mes Annonces

Service client

Portfeuille

Historique

Paramètres

Hamid aarif

© 2021 Titsuite Inc. Tous droits réservés

Logout

Personal details

First name

Prénom....

Last name

Nom....

Birth Date

28/08/1995

Adresse

16, Hay el fath

City

Ville

Phone number

+212-67-07-07-48

Activity

Marberie

Minimum wage

\$ 3000 DH

Save

Diplomas

Delete

Save

Name

Technicien spécialisé promo 2014

Date d'acquisition

12/07/2014

Field

Marberie

Add a new diploma

Add

Name

Technicien spécialisé promo 2014

Date d'acquisition

12/07/2014

Field

Marberie

FIGURE 3.7 – Profile du freelancer

Les pages attachées ci-dessus présentent la majorité des maquettes de notre application.

3.3.2 Réalisation du siteweb

Pour mettre en oeuvre notre application, on a décidé entre groupe d'utiliser la framework ReactJS.

C'était une première occasion pour apprendre les notions de base de ce puissant outil de développement qui se base sur la création et la réutilisation des composants. React nous a permis de mettre notre application sous forme d'une SPA et d'exploiter les différents avantages de cette dernière.

Chapitre 4

Tests et Déploiement

4.0.1 Tests

Afin de s'assurer du bon fonctionnement de notre application Web, plusieurs tests unitaires ont été menés ainsi que quelques tests d'intégration. Les outils Arquillian et Shrinkwrap nous ont permis de bundler notre API en fichier WAR et de tester le serveur. Ainsi la cohésion entre les différentes routes et services proposés par l'API a été confirmée.

4.0.2 Déploiement

Afin de déployer notre application JEE nous avons opté pour le docker compose, qui permet de rassembler les différentes images dont dépend notre application dans un seul fichier. Un fichier docker-compose.yml est placé dans le dossier main où on déclare toutes nos images Docker dont dépend l'application, en plus d'un fichier docker file où on a déclaré les étapes du build à suivre afin de démarrer notre serveur.


```
version: '3'
services:
  web:
    build: .
    ports:
      - 8080:8080
    links:
      - db:store/oracle/database-enterprise:12.2.0.1
  db:
    image: store/oracle/database-enterprise:12.2.0.1
    environment:
      ORACLE_ROOT_PASSWORD: Prince
      ORACLE_DATABASE: Prince

  ports:
    - 1521:1521
```

FIGURE 4.1 – Dockerfile

Conclusion générale

D'une manière générale, on pense que le pari a déjà été gagné car à la fin de ce projet nous étions capables de mieux définir et manipuler des concepts avancés de l'environnement JEE, l'application Titsuite certes a besoin de plus de fonctionnalités pour l'enrichir davantage, ainsi qu'une automatisation des tests pour garantir sa fiabilité que nous avons envisagé de faire à l'aide de Postman. Mais d'une manière générale l'application répond au besoins fonctionnels définis au préalable et prête à révolutionner la relation qui lie le Freelancer avec ses clients.