

Remerciements

Je tiens à remercier vivement nos encadrants de la filière GL, pour les recommandations et directives qu'ils nous ont prodiguées afin de mener à bien ce projet. Au terme de ce travail, Nous avons le plaisir d'exprimer nos sincères remerciements à M. El Hamlaoui qui nous a permis de mieux suivre le processus de la construction de notre application. Nous le remercions aussi pour son amabilité, collaboration, disponibilité et ses inestimables conseils. Nous remercions également le corps enseignant à l'ENSIAS et notamment nos professeurs qui nous ont offert une formation de qualité. De même, nous remercions tous les membres du cadre administratif de l'ENSIAS pour leurs efforts accrus fournis afin de procurer aux élèves ingénieurs un environnement pédagogique de haut niveau et assurer le bon déroulement des études en général.

Enfin nous remercions tous les gens qui ont contribué de près ou de loin dans la réalisation et la réussite de notre projet d'ingénierie du web.

Table des figures

Figure 1 : facebook_blog	5
Figure 2 : diagramme de cas d'utilisation	9
Figure 3 : Diagramme de séquence 'Téléchargement de fichiers'	10
Figure 4 : Diagramme de séquence 'Gestion fichiers'	11
Figure 5 : Diagramme de séquence 'Gestion adhérent'	12
Figure 6 : Diagramme de séquence 'Modifier taille espace adhérent'	13
Figure 7 : Diagramme de séquence 'Ajout admin'	14
Figure 8 :Diagramme de classe.	14
Figure 8 :Page d'inscription d'un nouveau adhérent.	18
Figure 9 :Page d'inscription erronée.	19
Figure 10 :Page d'inscription avec succès.	19
Figure 11 :Page d'authentification.	20
Figure 12 :Pages mon espace	21
Figure 13 :Page d'upload de fichier.	21
Figure 14 :Page de succès d'upload du fichier.	22
Figure 15 :Page de recherche d'un fichier.....	22
Figure 16 :Page des fichiers recherchés.	23
Figure 17 :Page de recherche d'un fichier pour un utilisateur non connecté.	24
Figure 18 :Page de succès de recherche d'un fichier pour utilisateur.	25
Figure 19 :Page d'ajout d'un administrateur.....	25
Figure 20 :Page de gestion des adhérents.....	26
Figure 21 :Page de changement de taille max d'un espace adhérent par un administrateur ..	26
Figure 22 :Page de changement du mot de passe	27
Figure 23 :Page de profil.....	27

Table des matières

Introduction générale.....	4
1 Présentation du sujet :	5
1.1 Contexte et enjeu :.....	5
1.2 Etude de l'existant.....	5
2 Spécification des besoins :	6
2.1 Introduction :.....	6
2.2 Spécification des besoins :	6
3 Etude conceptuelle	8
3.1 Diagrammes de cas d'utilisation :	8
3.2 Diagrammes de séquences :	9
3.3 Diagramme de classe :.....	14
3.4 Conclusion :.....	15
4 Réalisation.....	15
4.1 Environnement de travail :	15
4.2 Mise en œuvre :	18

Introduction générale

Les sites d'hébergement de fichiers sont des services qui permettent aux utilisateurs d'héberger des fichiers sur leurs serveurs. Certains hébergent tous les types de fichiers, d'autres se classent selon les types de fichiers qu'ils prennent en charge : vidéos, photographies, ou autres.

C'est dans cette vision que s'inscrit notre projet d'Ingénierie du web qui vise à créer un hébergeur de fichiers.

Ces applications constituent alors une plateforme qui propose des modules facilitant le travail collaboratif, le partage, les interactions, les échanges via des dossiers ou fichiers se trouvant dans un espace adhérent. De plus, la plupart du temps, l'enregistrement des fichiers est possible, chacun peut y revenir si besoin, revoir ou partager ce qu'il a déjà téléverser.

1 Présentation du sujet :

1.1 Contexte et enjeu :

Uploader les fichiers est un moyen indispensable pour le partage de l'information et aussi pour sauvegarder ses données sur un ordinateur ou un serveur distant.

Une conséquence de tout ça c'est que l'échange d'information et le partage des données(fichiers) est une activité primordiale pour les internautes pour la diffusion de l'information et pour l'enregistrement de ses données personnels , c'est dans cette politique qu'on a suggéré d'implémenter une application web qui a pour but l'échange toutes informations (fichiers) et le partage des ressources qui pourrait être utiles pour l'échange des données toujours dans l'intérêt de l'internaute , constituant ainsi une plateforme de partage importante qui peut être consultée par tous les utilisateurs ou adhérents qui donne alors un aspect d'élaboration des pratiques nouvelles de partage et d'enregistrement des informations.

1.2 Etude de l'existant

Dans le contexte de partage et de circulation des documents on trouve que les sites comme *Facebook*, *YouTube* ou *Twitter*, les *blogues* qui se sont développés à grande vitesse au cours des dernières années, s'avèrent hautement populaires auprès des internautes notamment parce qu'ils leur offrent l'opportunité de collaboration à distance et le partage des ressources et la diffusion de l'information.



Figure 1 : facebook_blog .

Cependant les hébergeurs de fichiers ont des limites c'est qu'ils spécifient toujours des espaces très limités pour les adhérents et le temps de téléversement d'un fichier est très grand.

2 Spécification des besoins :

2.1 Introduction :

Une étape essentielle de tout cycle de développement logiciel ou conceptuel consiste à effectuer une étude préalable. Le but de cette phase est de comprendre le contexte du Système. Il s'agit d'éclaircir au mieux les besoins fonctionnels et non fonctionnels, faire apparaître les acteurs et identifier les cas d'utilisation.

Dans ce chapitre, nous allons essayer d'exprimer les besoins sous forme de diagrammes des cas d'utilisation.

2.2 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 distinguons 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.

2.2.1 Spécification des besoins fonctionnels

Après une étude détaillée de système, cette partie est réservée à la description des exigences fonctionnelles des différents acteurs de l'application. Ces besoins se regroupent dans

les diagrammes des cas d'utilisation.

- Les acteurs de l'application sont :
 - Utilisateur
 - Adhérent
 - Administrateur
- Les exigences fonctionnelles sont :
 - Gérer les fichiers :

Notre application permet aux administrateurs de faire tous type de gestion sur les fichiers et permet aux adhérents de téléverser,télécharger ou modifier les fichiers et de les consulter par la suite, de plus elle donne la possibilité aux utilisateurs de télécharger ou de rechercher un fichier tout en possédant son url.

-Gérer les adhérents :

Notre application offre la possibilité aux administrateurs de modifier ou supprimer les adhérents du site ainsi que d'ajouter des nouveaux administrateurs.

-Gérer les données personnels :

Notre application permet à chaque adhérent accédant à son espace personnel de consulter ses informations ainsi que de les modifier .

2.2.2 Spécification des besoins non fonctionnels

Les besoins non fonctionnels décrivent toutes les contraintes techniques, ergonomiques et esthétiques auxquelles est soumis le système pour sa réalisation et pour son bon fonctionnement. Et ce qui concerne notre application, nous avons dégagé le besoins suivants :

- **La disponibilité** : l'application doit être disponible pour être utilisé par n'importe quel utilisateur.
- **La fiabilité** : les données fournies par l'application doivent être fiables.
- **Ergonomie et bonne Interface** : L'application doit être adaptée à l'utilisateur sans qu'il ne fournisse aucun effort (utilisation claire et facile) de point de vue navigation entre les différentes pages, couleurs et mise en textes utilisés.
- **Une solution ouverte et évoluée** : l'application peut être améliorée par l'ajout d'autres modules pour garantir la souplesse, l'évolutivité et l'ouverture de la solution.

3 Etude conceptuelle

Introduction

Dans la phase de la conception détaillée, et pour mieux éclaircir les diagrammes d'analyse, nous avons un ensemble de diagrammes UML. Ces diagrammes servent à bien structurer la couche métier de l'application, en rassemblant les classes d'analyse par fonctionnalité.

3.1 Diagrammes de cas d'utilisation :

Les intérêts des diagrammes des cas d'utilisation sont d'abord et avant tout de :

- modéliser le système;
- de déterminer les fonctionnalités du système à travers une vue d'un acteur et de les représenter.

Les cas d'utilisation permettent de forcer l'utilisateur ou l'expert à définir ce qu'il attend du système:

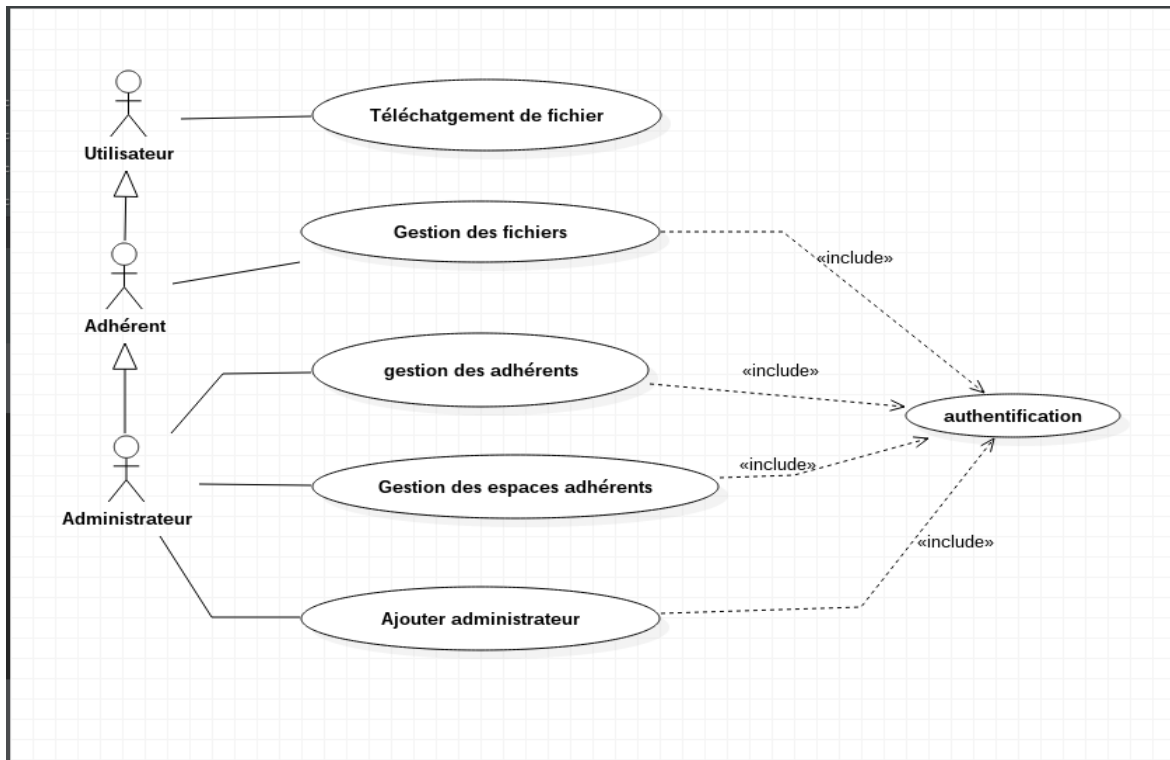


Figure 2 : diagramme de cas d'utilisation .

L'utilisateur possédant un url ou un nom de fichier peut rechercher et télécharger ce fichier.

L'adhérent peut télécharger, modifier ou téléverser les fichiers ainsi que gérer ses données personnels.

L'administrateur a le droit de gérer les fichiers, les adhérents, les espaces adhérents ainsi que l'ajout d'un nouveau utilisateur.

3.2 Diagrammes de séquences :

Le diagramme de séquences est un modèle dynamique d'UML (Unified Modelling Language). Il permet de représenter l'enchaînement des opérations au sein d'un processus, en faisant apparaître les actions des différents objets qui sont concernés et les flux d'informations échangés entre eux.

Les diagrammes de séquences les plus importants notre application sont les suivants :

- Diagramme de Séquence « Téléchargement de fichiers »

- Diagramme de Séquence « Gestion fichiers »
- Diagramme de Séquence « Gestion adhérents »
- Diagramme de Séquence « Modifier taille espace adhérent »
- Diagramme de Séquence « Ajout admin »

❖ Scenario (Téléchargement de fichiers) :

- L'utilisateur choisit l'action télécharger
- Le système retourne un formulaire
- L'utilisateur saisie l'url ou le nom de fichier
- L'utilisateur choisit l'action valider
- L'utilisateur va sélectionner le fichier
- Le système va retourner le fichier à télécharger

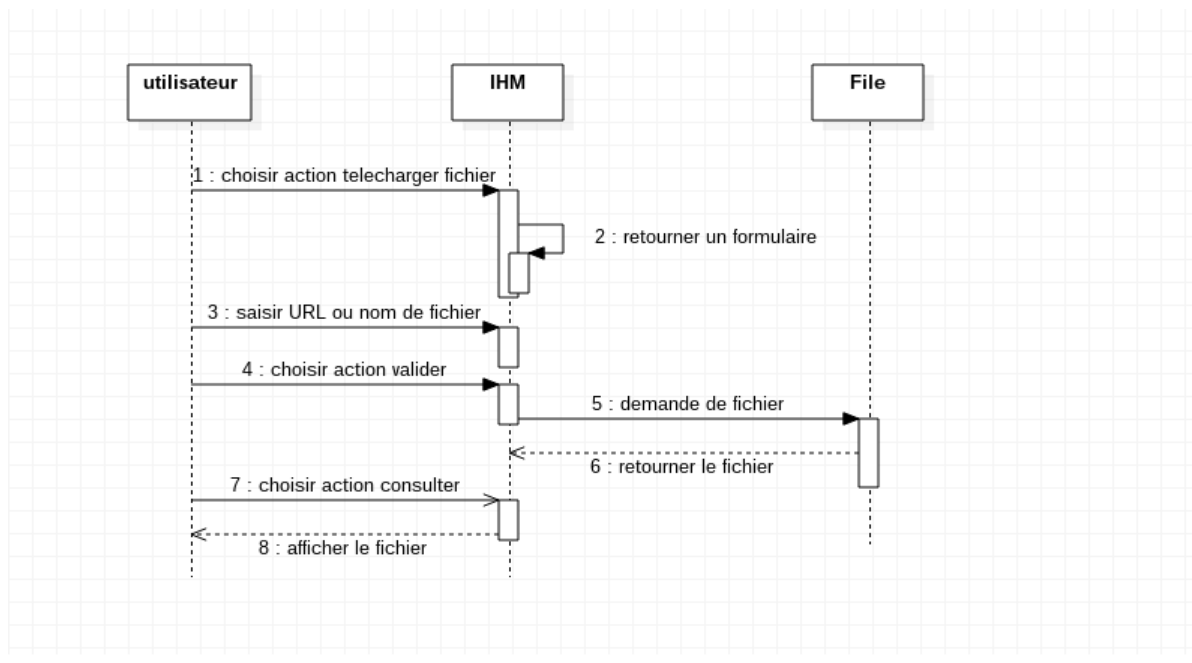


Figure 3 : Diagramme de séquence ‘Téléchargement de fichiers’.

❖ Scenario (Gestion fichiers) :

- L'adhérent ou l'administrateur s'authentifie
- Le système valide l'authentification de l'utilisateur
- Le système va afficher la liste des fichiers
- L'utilisateur choisit l'une des actions téléverser, supprimer ou partager
- Le système va réagir selon le choix de l'action

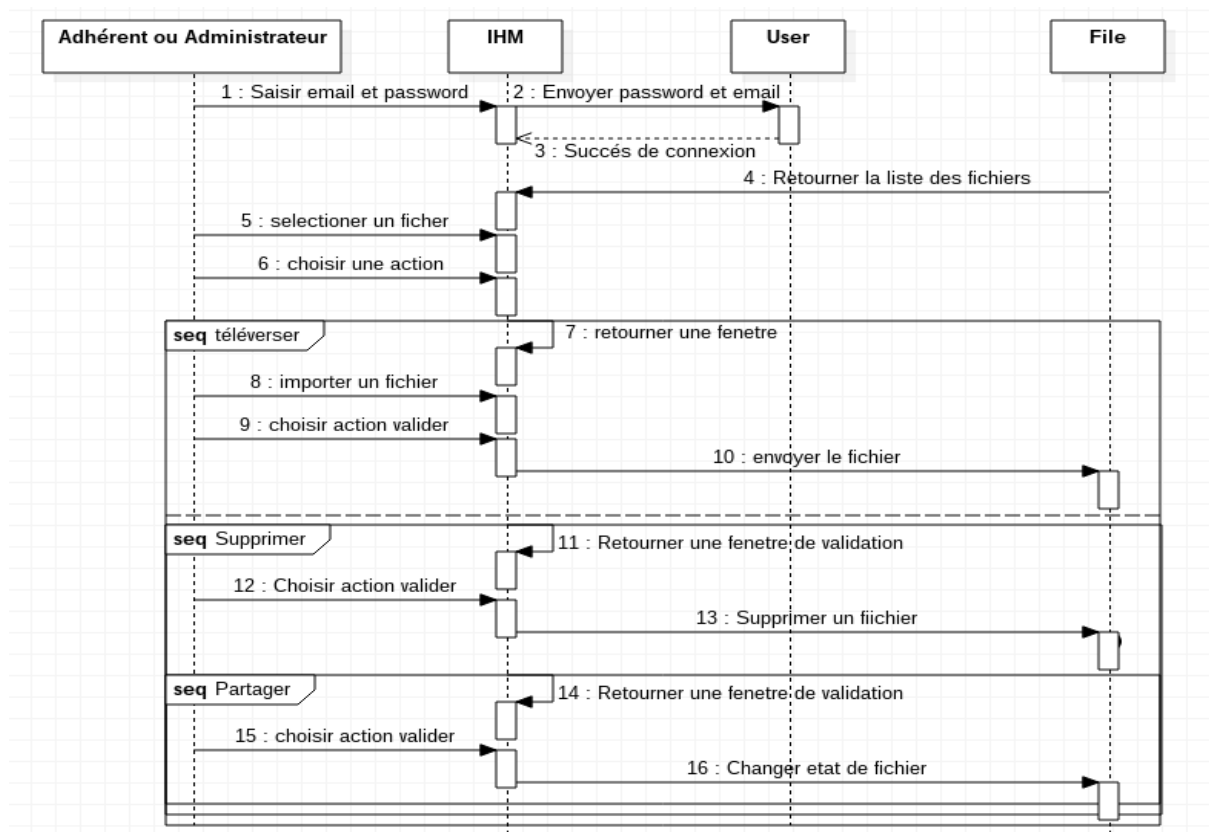


Figure 4 : Diagramme de séquence 'Gestion fichiers'.

❖ Scenario (Gestion adhérents) :

- L'administrateur s'authentifie
- Le système valide l'authentification de l'utilisateur
- L'administrateur choisit l'action afficher liste des adhérents
- Le système va afficher la liste des adhérents
- L'administrateur sélectionne un adhérent
- L'administrateur choisit une action soit changer état soit supprimer
- Enfin le système réagit selon l'action choisit.

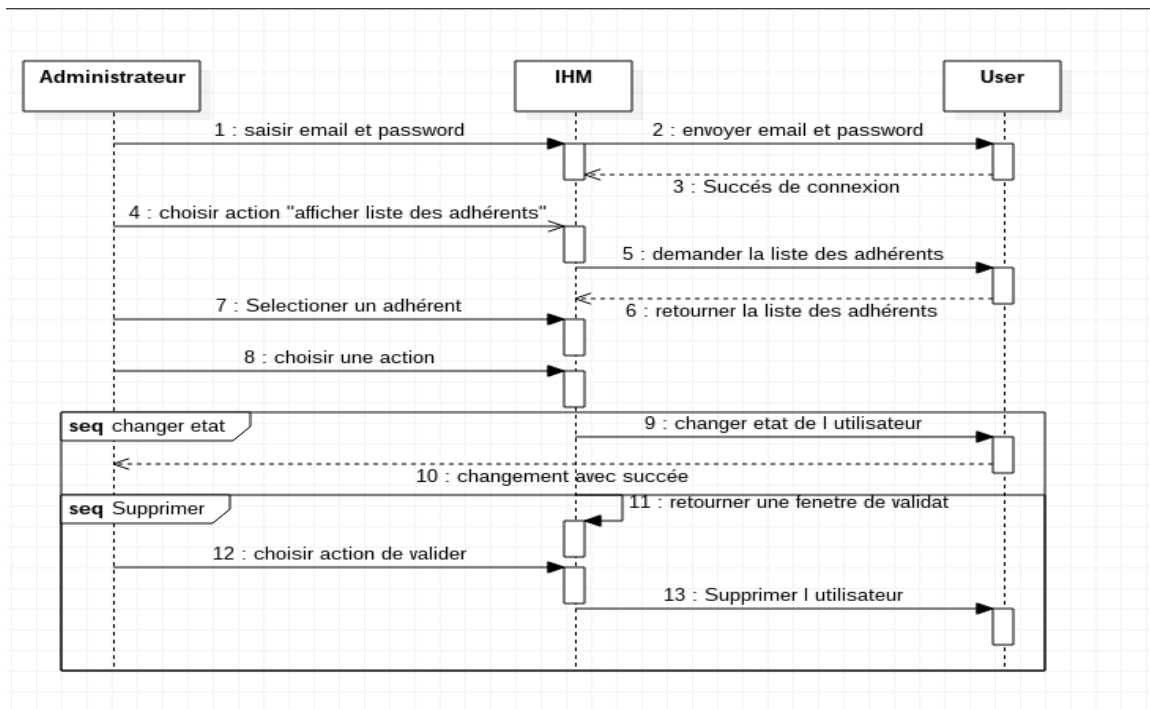


Figure 5 : Diagramme de séquence ‘Gestion adhérent’.

❖ Scenario (Modifier taille espace adhérent) :

- L’administrateur s’authentifie
- Le système valide l’authentification de l’utilisateur
- L’administrateur choisit l’action afficher liste des adhérents
- Le système va afficher la liste des adhérents
- L’administrateur sélectionne un adhérent
- L’administrateur choisit l’action changer espace
- Enfin le système modifie la taille de l’espace adhérent

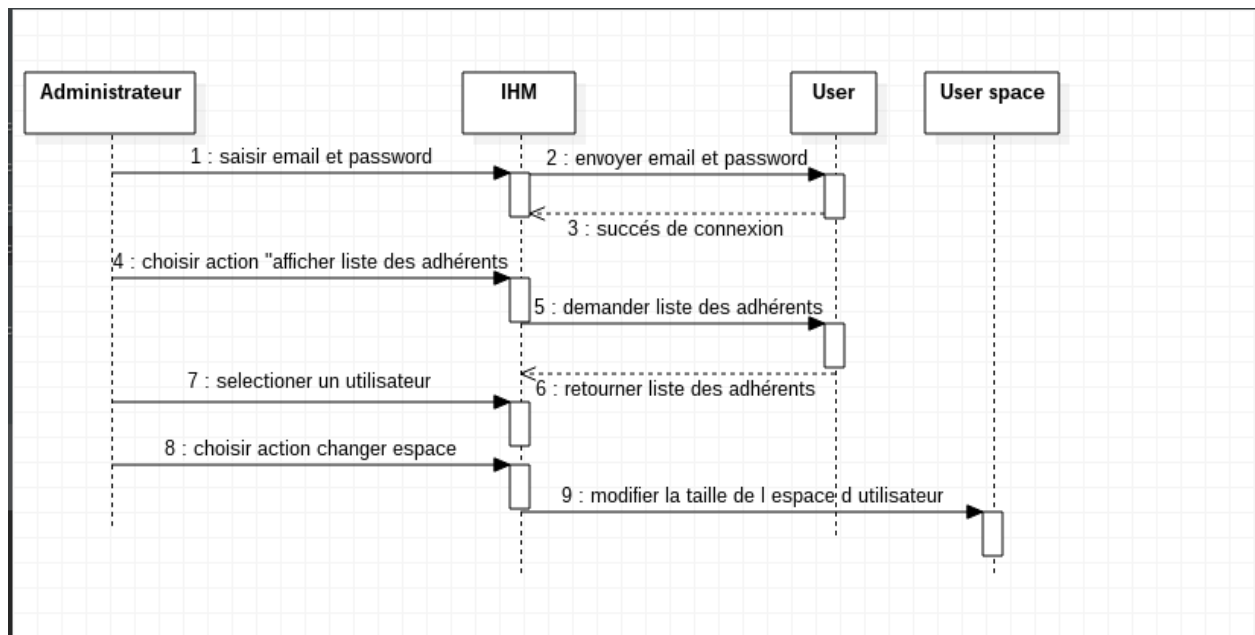


Figure 6 : Diagramme de séquence ‘Modifier taille espace adhérent’.

❖ Scenario (Ajout admin) :

- L’administrateur s’authentifie
- Le système valide l’authentification de l’utilisateur
- L’administrateur choisit l’action afficher liste des adhérents
- Le système va afficher la liste des adhérents
- L’administrateur sélectionne un adhérent
- L’administrateur choisit l’action changer espace
- Enfin le système modifie la taille de l’espace adhérent

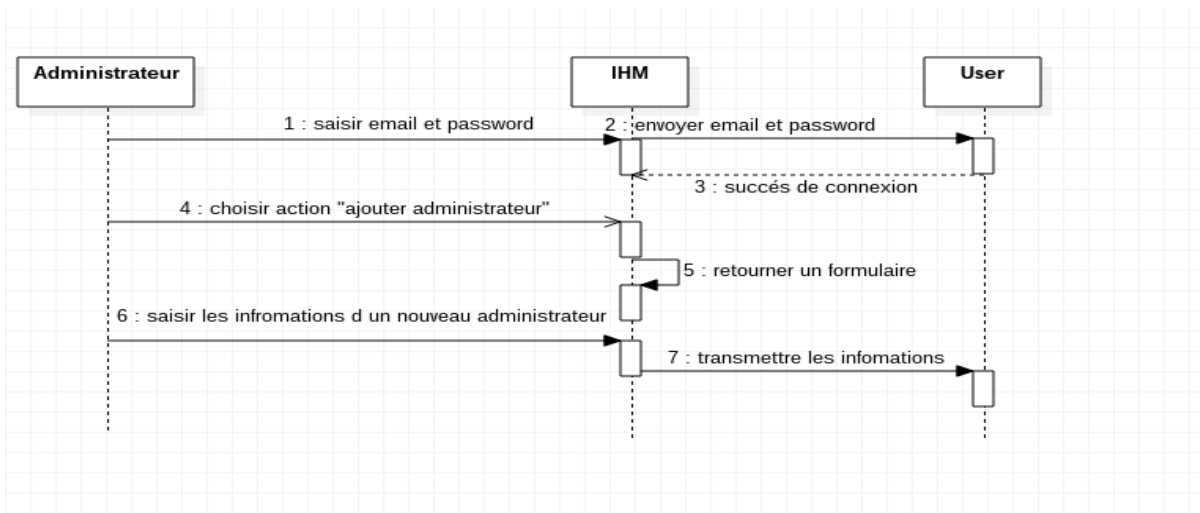


Figure 7 : Diagramme de séquence ‘Ajout admin’.

3.3 Diagramme de classe :

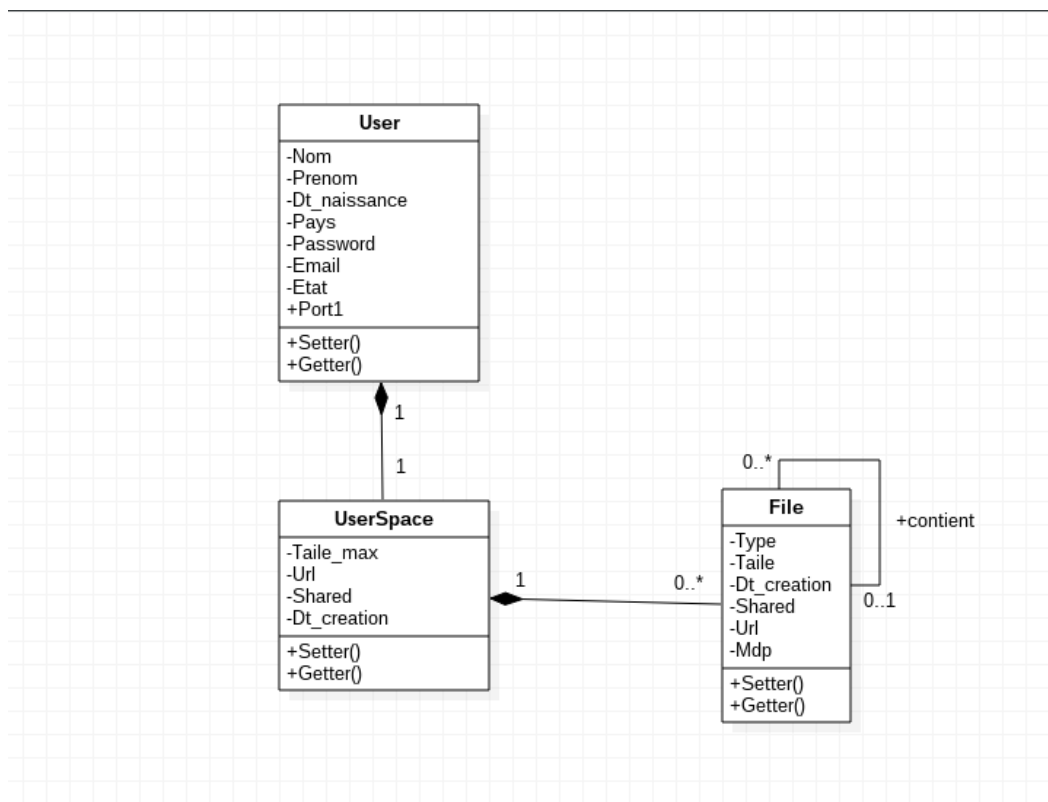


Figure 8 :Diagramme de classe.

Le diagramme de classes permet d'appréhender, d'un point de vue logique, la structure statique du système en indiquant la structure de ses objets et les liens entre ces derniers.

3.4 Conclusion :

Après avoir présenté les différentes étapes de conception de l'application tout en spécifiant les détails, on aura maintenant un aspect clair sur le déroulement de différentes tâches disponibles par notre application, nous avons arrivé à la fin de ce chapitre pour aborder un autre qui sera à propos de la réalisation de notre solution.

4 Réalisation

4.1 Environnement de travail :

4.1.1 Eclipse



Eclipse est un projet, décliné et organisé en un ensemble de sous-projets de développements logiciels, de la fondation Eclipse visant à développer un environnement de production de logiciels libre qui soit extensible, universel et polyvalent, en s'appuyant principalement sur Java.

4.1.2 Wampserver :



WAMP installe un environnement de serveur local sur Windows, il se réfère à un ensemble de logiciels libres couramment utilisés ensemble, pour lancer des sites web dynamiques sur les serveurs exécutant le système d'exploitation Windows.

4.1.2 Technologies JEE utilisés

4.1.2.1 JSP



Le Java Server Pages ou JSP est une technique basée sur Java qui permet aux développeurs de créer dynamiquement du code HTML, XML ou tout autre type de page web. Cette technique permet au code Java et à certaines actions prédéfinies d'être ajoutés dans un contenu statique. Depuis la version 2.0 des spécifications, la syntaxe JSP est complètement conforme au standard XML.

La syntaxe du JSP ajoute des balises XML, appelées actions JSP, qui peuvent être utilisées pour appeler des fonctions. De plus, cette technique permet la création de bibliothèques de balises JSP (tag lib) qui agissent comme des extensions au HTML ou au XML. Les bibliothèques de balises offrent une méthode indépendante de la plate-forme pour étendre les fonctionnalités

d'un serveur HTTP. Il existe aussi un langage de script particulier, appelé Expression Language (EL) destiné à réduire l'injection de code java au sein des pages JSP ainsi qu'à étendre les possibilités des taglibs, tel que la JSTL.

4.1.3 Langages utilisés

4.1.3.1 Java



Java est un langage de programmation orienté objet, développé par Sun Microsystems et destiné à fonctionner dans une machine virtuelle, il permet de créer des logiciels compatibles avec des nombreux systèmes d'exploitation.

4.1.3.2 Javascript

JavaScript est un langage de programmation de scripts principalement employé dans les pages web interactives mais aussi pour les serveurs avec l'utilisation (par exemple) de Node.JS.



4.1.3.3 jQuery :

jQuery est une bibliothèque JavaScript libre et multi-plateforme créée pour faciliter l'écriture de scripts côté client dans le code HTML des pages web



4.1.4 Outils graphiques

4.1.4.1 Bootstrap



Bootstrap est une collection d'outils utile à la création du design (graphisme, animation et interactions avec la page dans le navigateur ... etc. ...) de sites et d'applications web. C'est un ensemble qui contient des codes HTML et CSS, des formulaires, boutons, outils de navigation et autres éléments interactifs, ainsi que des extensions JavaScript en option. C'est l'un des projets les plus populaires sur la plate-forme de gestion de développement Git Hub.

4.2 Mise en œuvre :

4.2.1 Espace adhérent :

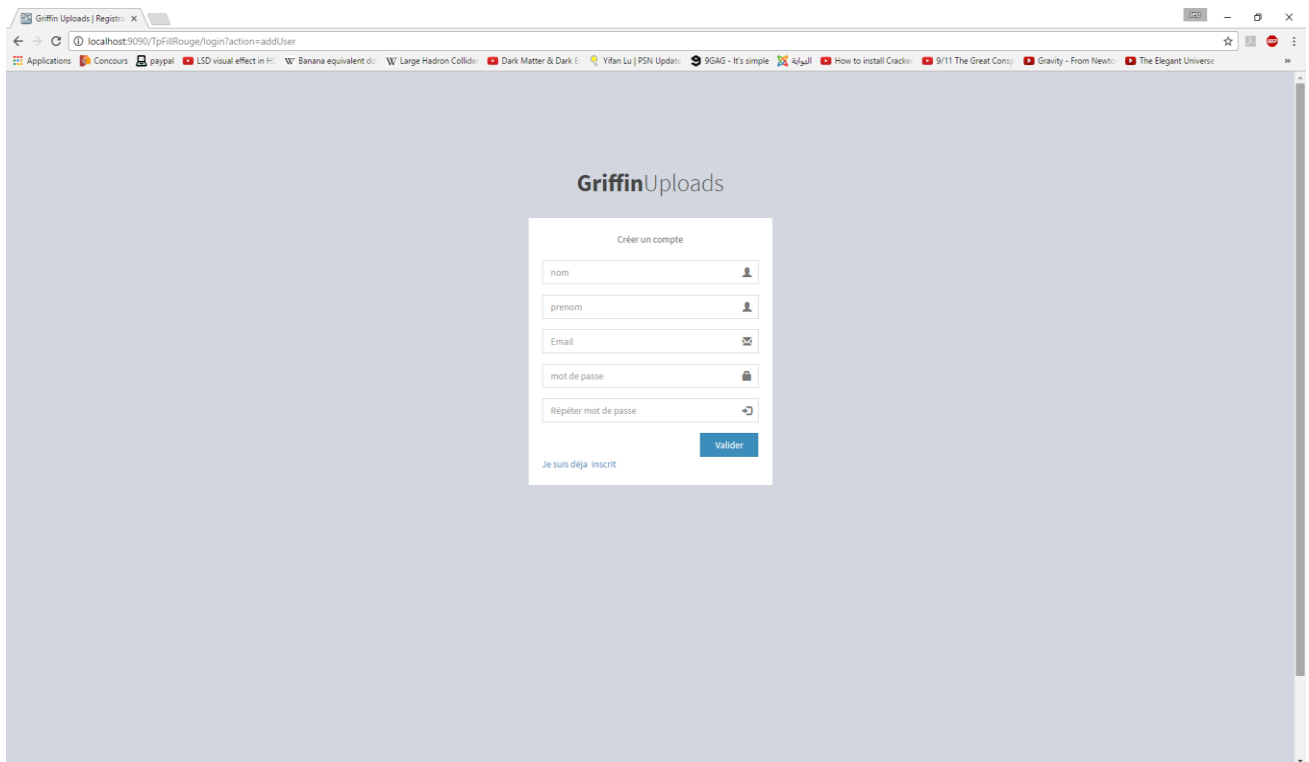


Figure 9 :Page d’inscription d’un nouveau adhérent.

Le nouveau adhérent doit saisir ces informations personnels pour s’inscrire.

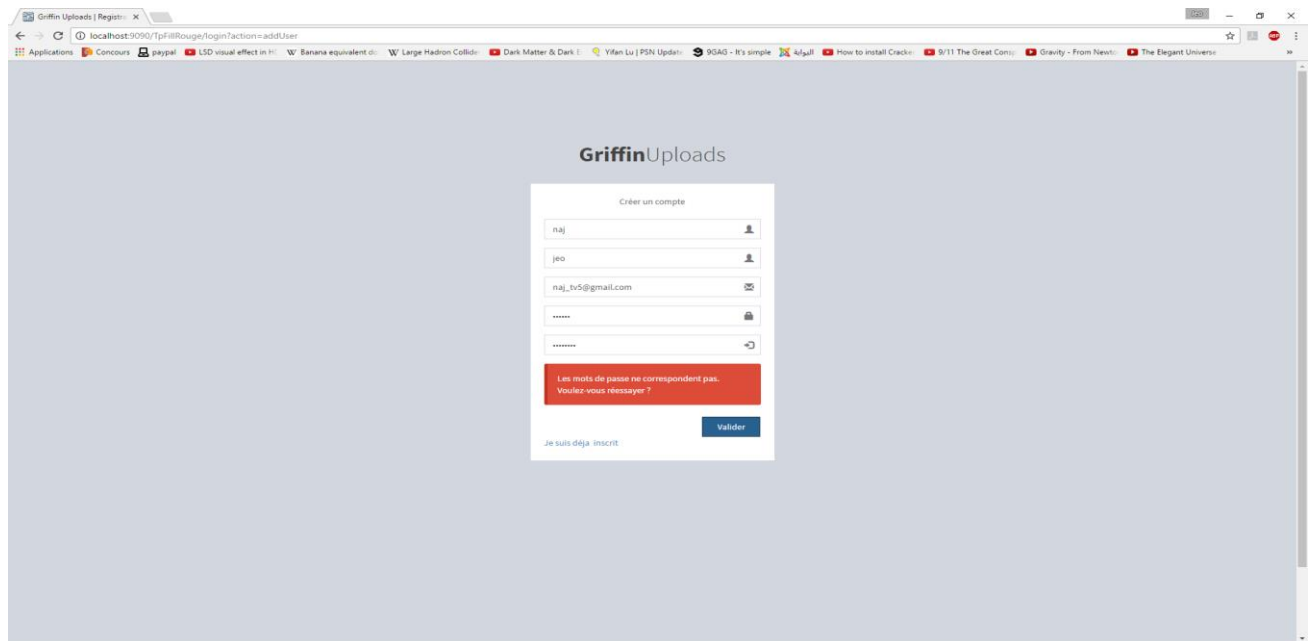


Figure 10 :Page d'inscription erronée.

Si la nouvelle personne saisit un email déjà utilisée un message d'erreur s'affiche.

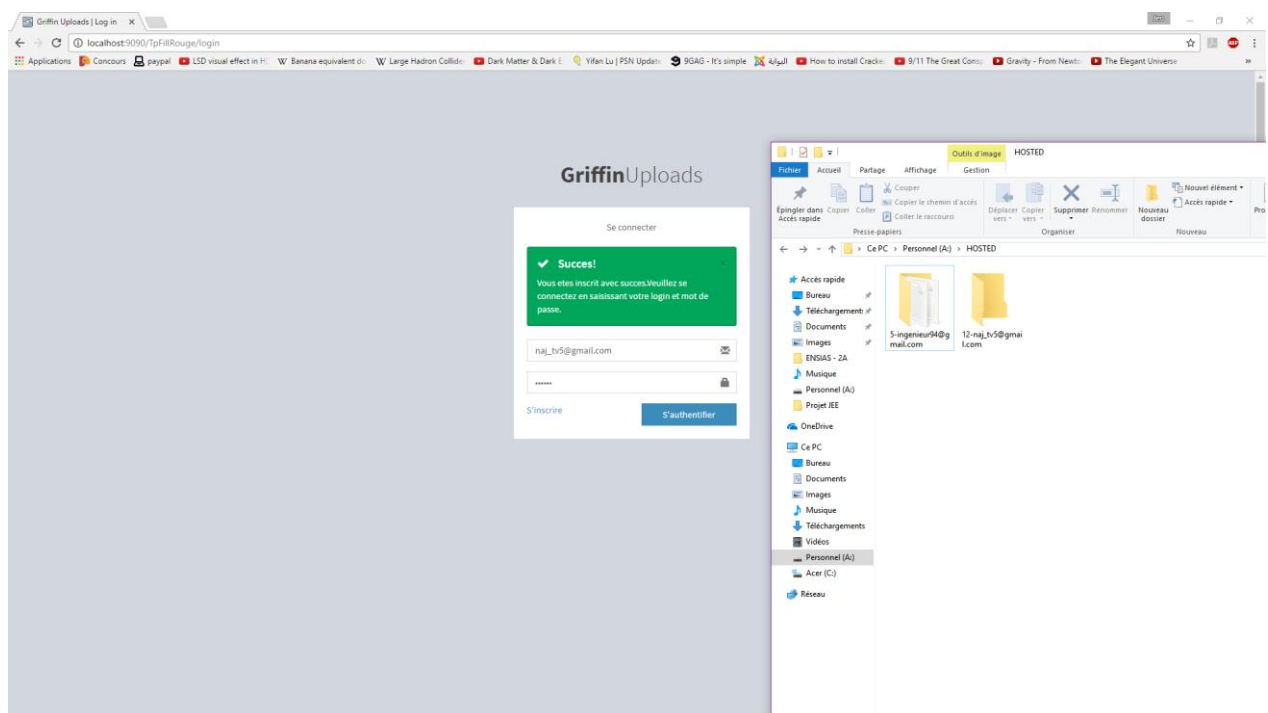


Figure 11 :Page d'inscription avec succès.

Après une nouvelle inscription d'un adhérent, un fichier personnel se génère automatiquement.

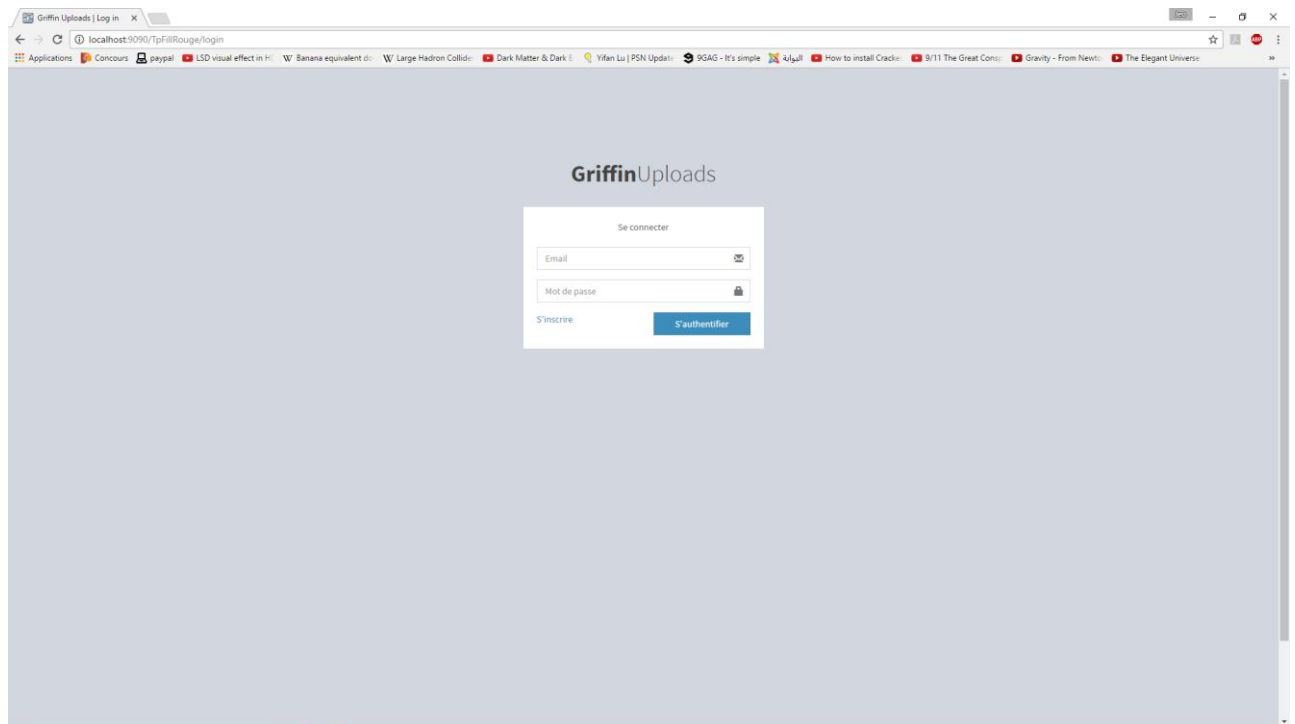


Figure 12 :Page d'authentification.

L'adhérent doit saisir son email et son mot de passe pour accéder à son espace personnel.

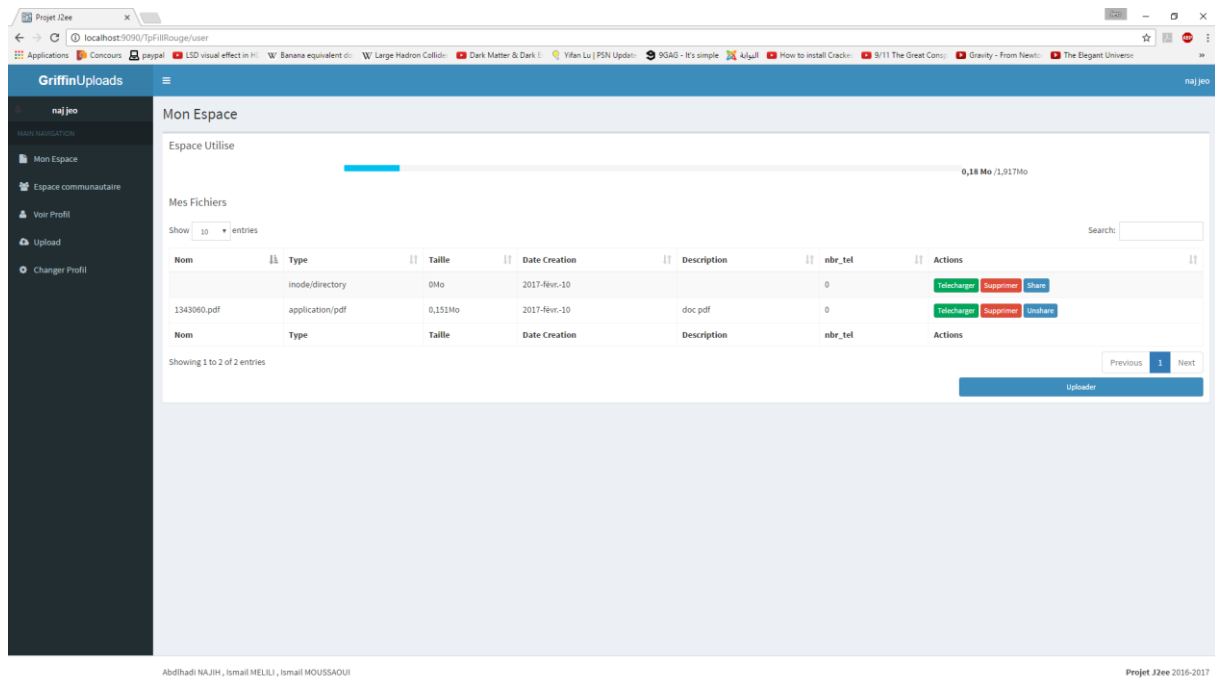


Figure 13 :Pages mon espace

La page contient tous les fichiers uploader par l'utilisateur.

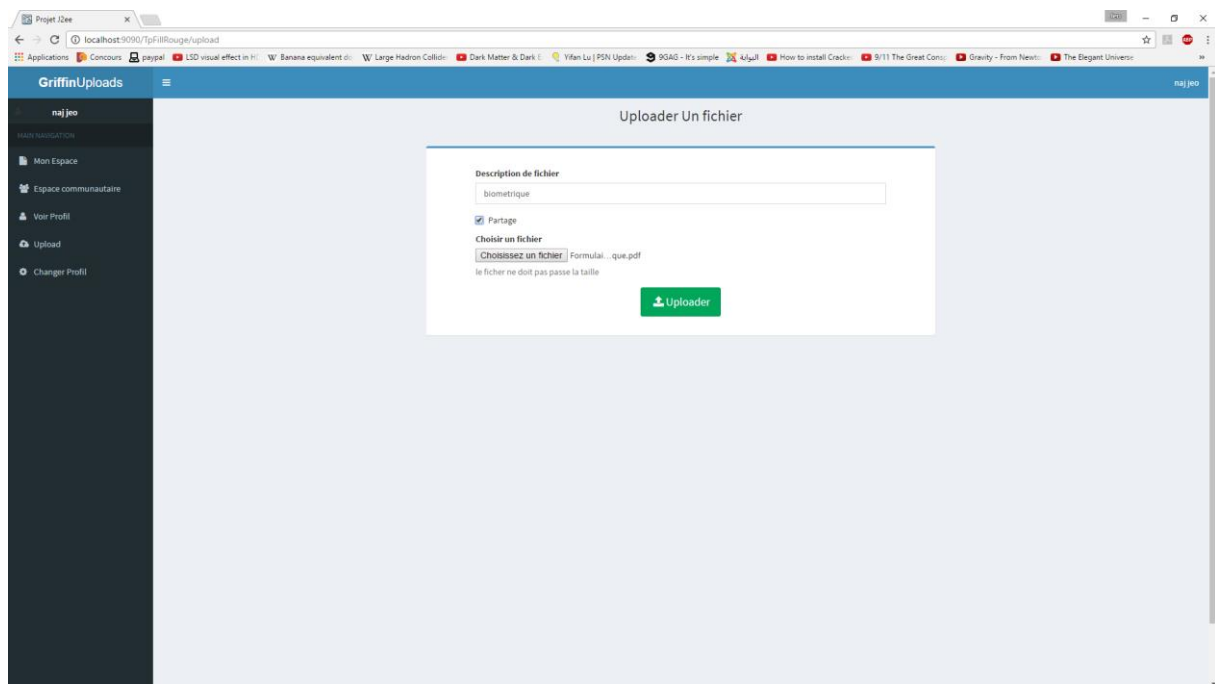


Figure 14 :Page d'upload de fichier.

L'adhérent choisit le fichier à uploader.

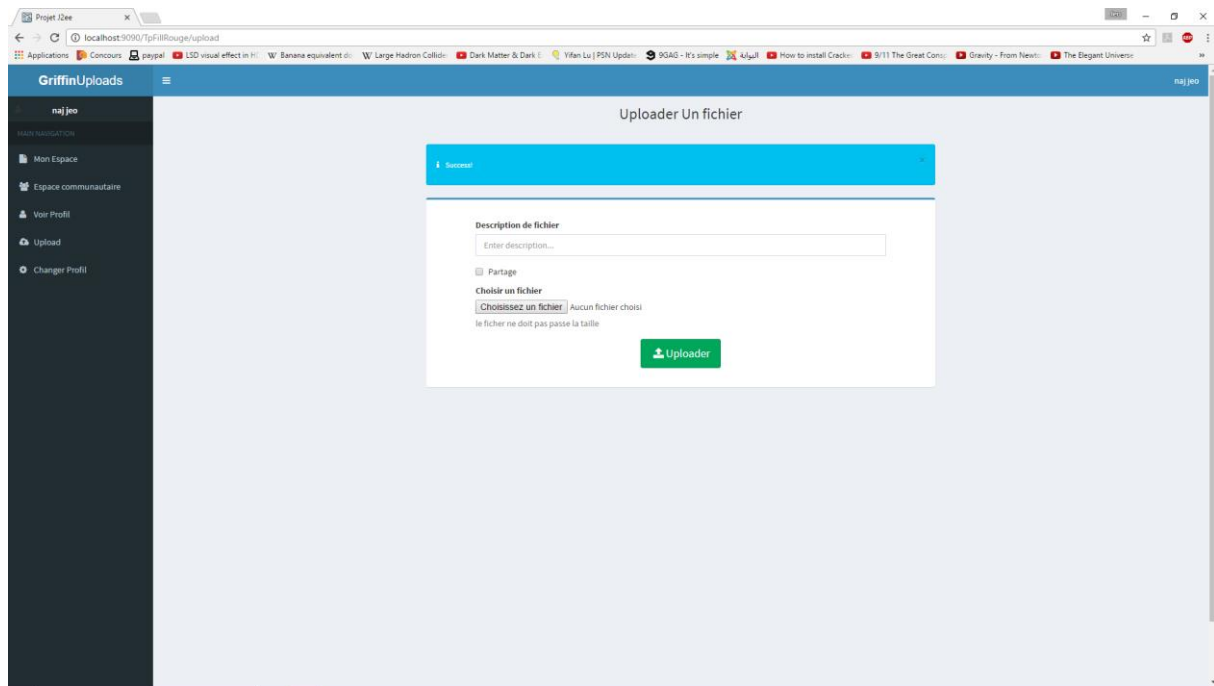


Figure 15 :Page de succès d'uploade du fichier.
L'adhérent a uploader son fichier avec succès.

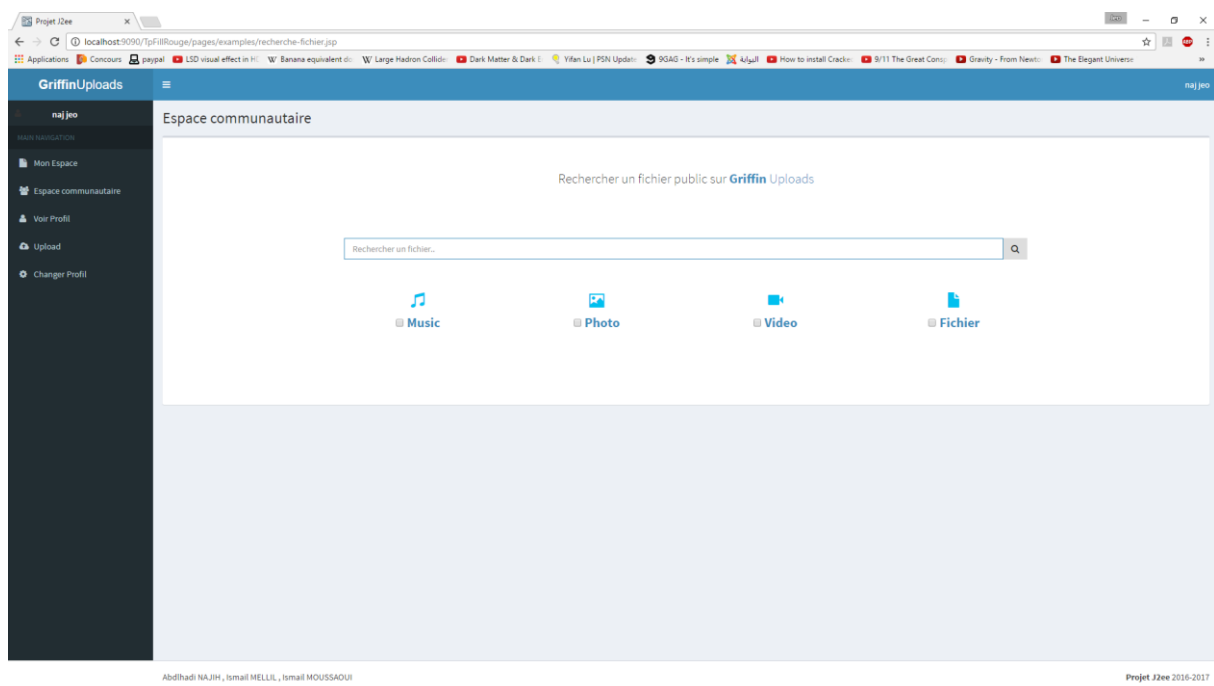


Figure 16 :Page de recherche d'un fichier.

L'adhérent peut rechercher un fichier par son type sinon, il s'affiche tous les fichiers propriétaire à cet adhérent.

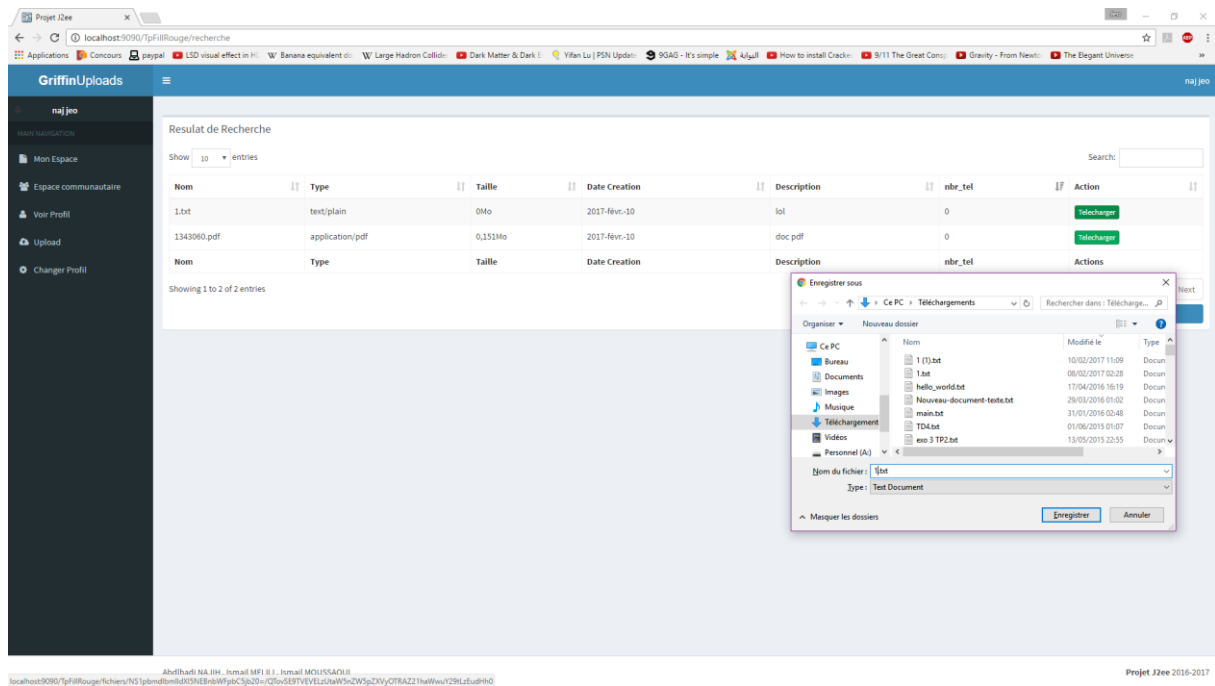


Figure 17 :Page des fichiers recherchés.

L'affichage des fichiers par ordre de téléchargement.

4.2.2 Espace Utilisateur :

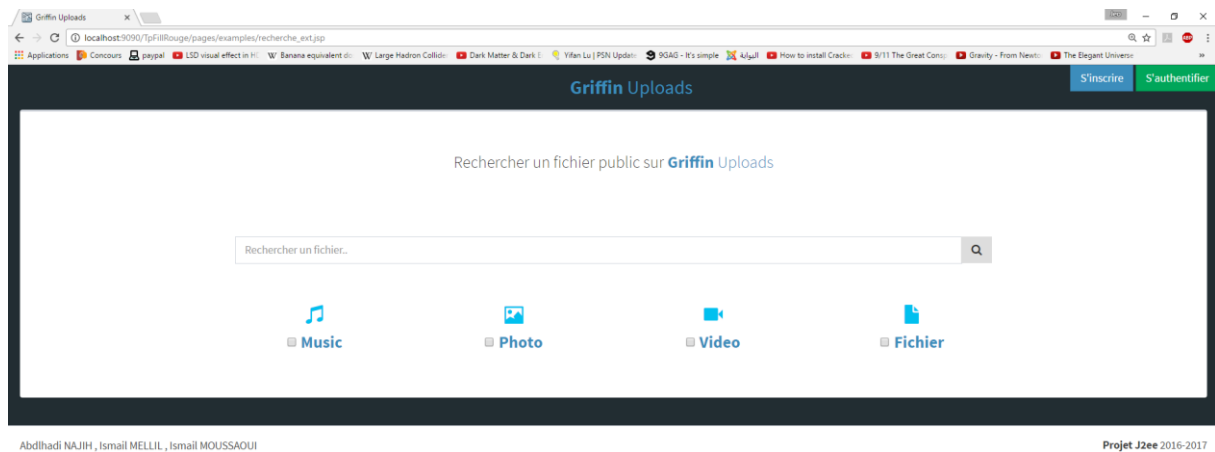


Figure 18 :Page de recherche d'un fichier pour un utilisateur non connecté.

Un utilisateur non connecté peut soit rechercher un fichier par son nom et le télécharger ,soit taper son url pour le télécharger directement .

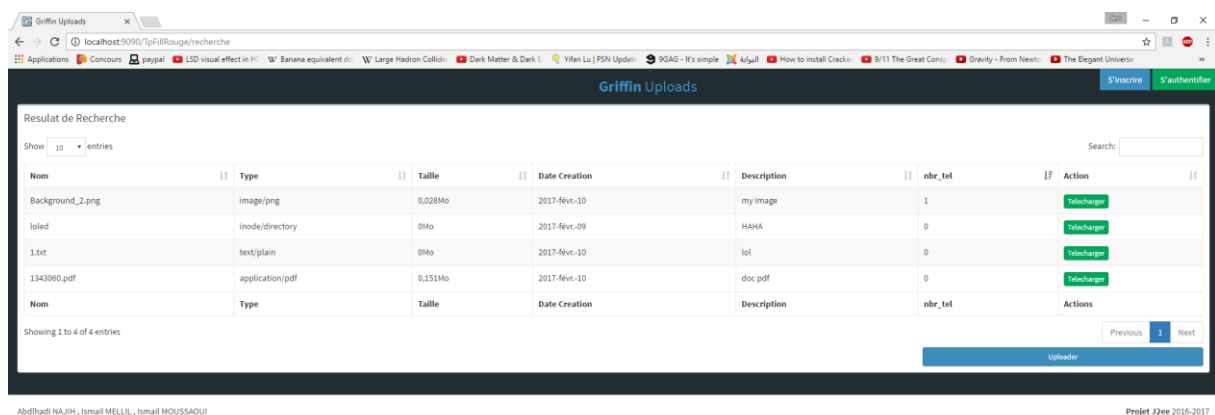


Figure 19 :Page de succès de recherche d'un fichier pour utilisateur.

L'utilisateur peut télécharger les fichiers qui ont le droit d'être partagés.

4.2 .3 Espace Administrateur :

The screenshot displays the GriffinUploads web application in a browser. The address bar shows the URL: localhost:9090/TpFillRouge/admin?action=addAdmin. The page has a dark blue header with the 'GriffinUploads' logo and the user 'ismail moussaoui'. A dark sidebar on the left contains navigation links: 'Gestion des utilisateurs', 'Ajouter un admin', 'Voir Profil', and 'Changer Profil'. The main content area is titled 'Ajouter un admin' and features a 'Créer un compte' form. The form includes input fields for 'nom', 'prenom', 'Email', 'mot de passe', and 'Répéter mot de passe', each with a corresponding icon (person, person, envelope, lock, and refresh). A blue 'Valider' button is at the bottom right of the form. The footer contains the text 'Abdihadi NAJH, ismail MELLIL, ismail MOUSSAOUI' and 'Projet I2ee 2016-2017'.

Figure 20 :Page d'ajout d'un administrateur.

L'utilisateur a le droit d'ajouter un nouveau utilisateur.

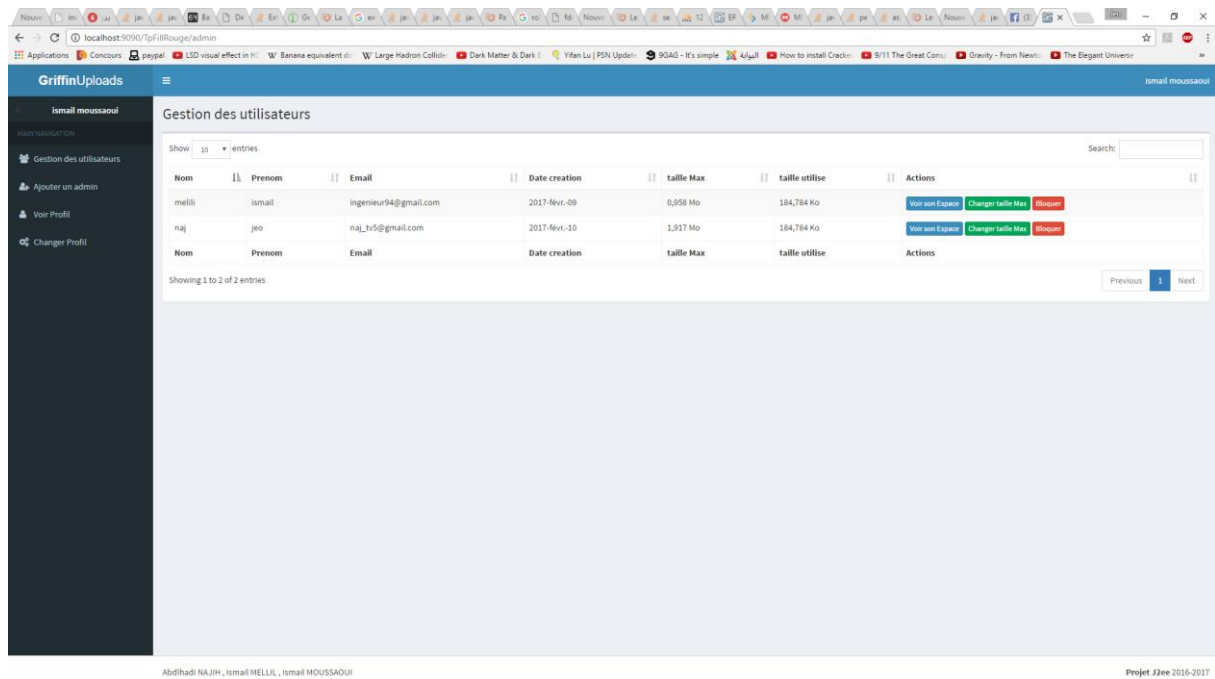


Figure 21 :Page de gestion des adhérents.

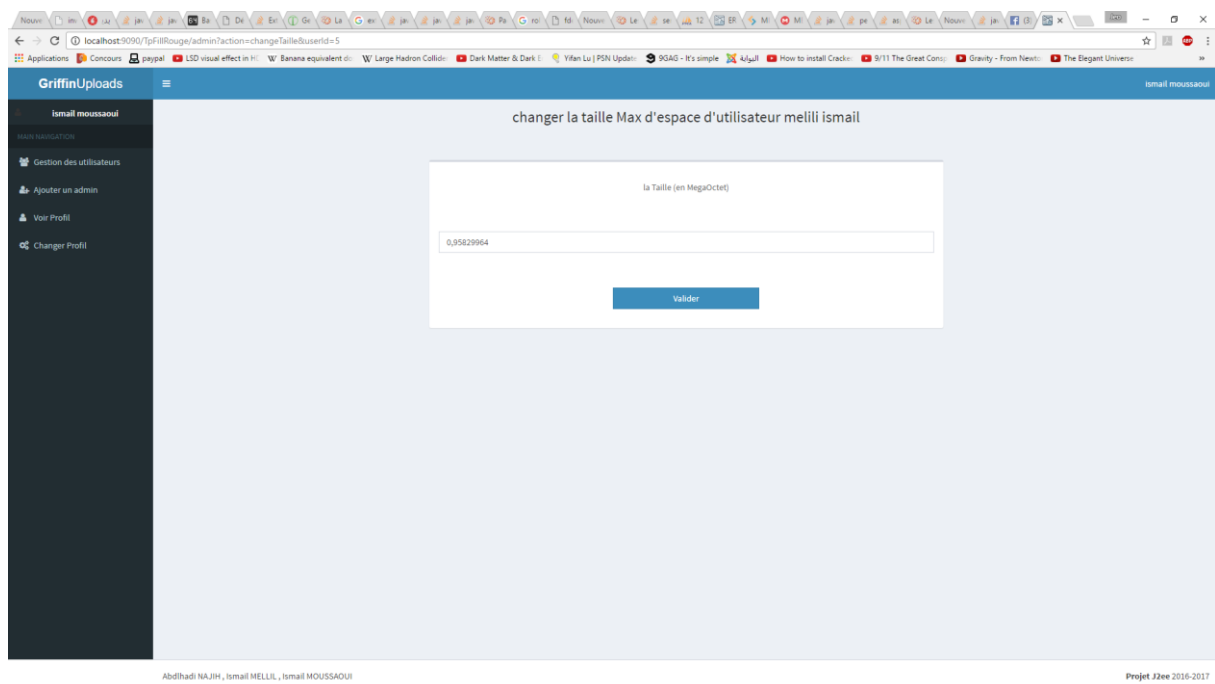


Figure 22 :Page de changement de taille max d'un espace adhérent par un administrateur.

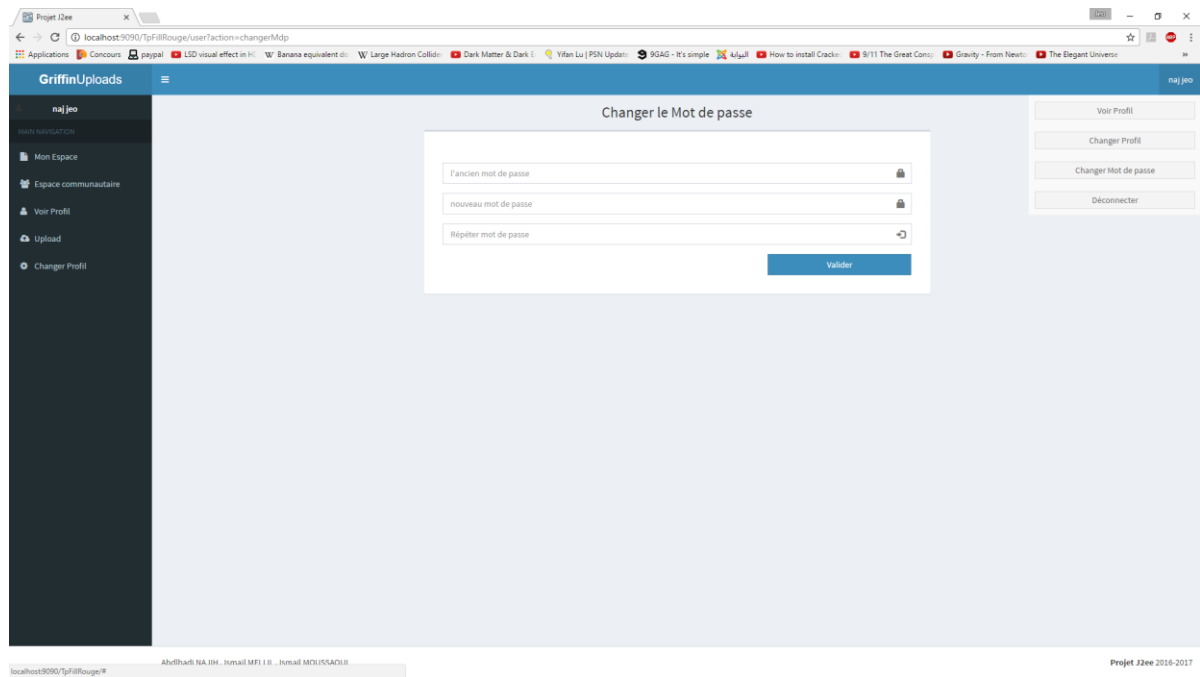


Figure 23 :Page de changement du mot de passe .

L'administrateur ou l'adhérent ont le droit de changer leur mot de passe .

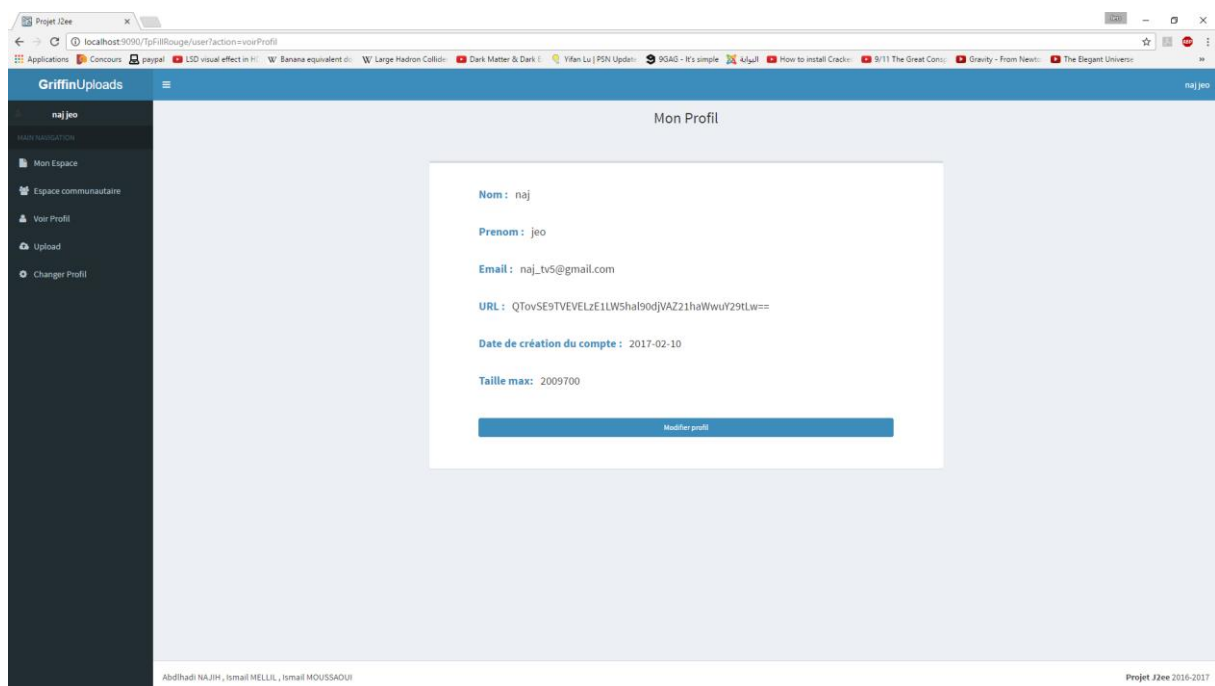


Figure 24 :Page de profil.

L'administrateur ou l'adhérent ont le droit de consulter leur profil personnel .

Conclusion

Notre projet consistait à développer un hébergeur de fichier pour les internautes pour faciliter le stockage et le partage des données entre adhérent et utilisateur portant un url ou un nom de fichier.

La réalisation du projet a commencé par une étude détaillée des besoins requis. Ensuite, nous avons construit une modélisation du futur système en utilisant le langage UML, au terme de l'étude fonctionnelle. Puis, nous avons proposé une architecture logicielle tout en choisissant les différents outils de développement nécessaires à sa réalisation.

Toutefois, des améliorations restent envisageables. Par exemple on peut développer la partie upload pour uploader un dossier tout entier sous format .zip par exemple .

Webographie

<https://openclassrooms.com/courses/creez-votre-application-web-avec-java-ee>

<https://www.mistra.fr/tutoriels-java/tutoriel-jee.html>

Bibliographie

Cours JEE de M.EL HAMPLAOUI

Cours UML de Mme.Berrada