

**Rapport Travaux Pratiques**

**:**

**Plateforme de gestion de Projet et attribution des tâches**

**Du personnel de l’entreprise INGDEUX**

**INTRODUCTION**

En vue d’accroitre la productivité et le rendement de ses activités en facilitant le travail collaboratif, l’entreprise INGDEUX a entrepris mettre sur pied une plateforme de gestion et attribution des tâches de son personnel, pour cela une équipe compétente et qualifiée dans le domaine a été sollicité pour la réalisation de cette dite plateforme.

De par ses compétences diverses, l’équipe est constituée de trois Ingénieurs de conception en Systèmes d’information, présentés comme suit :

|  |  |
| --- | --- |
| **GROUPE\_8** | |
| Noms et prénoms | Qualifications et tâches |
| TAKOUMBO MBE Yvan (chef du groupe) | Elève ingénieur de conception en systèmes d’informations à IAI (2ème année). |
| Controlleurs(userServlet, LoginServlet) |
| Config web.xml, UserDAO |
| Analyse et conception du script DB |
| Models |
| KOUEMOU MODI Emmanuelle Samanta | Elève ingénieur de conception en systèmes d’informations à IAI (2ème année) |
| Vue admin.jsp, superAdmin.jsp |
| Analyse et conception du script DB |
| Rédaction du rapport |
|  |
| MOUTEKE Destin | Elève ingénieur de conception en systèmes d’informations à IAI (2ème année) |
| TacheDAO, ProjetDAO |
| Vue user.jsp |
| Analyse et conception du script DB |
|  |

# I. TECHNOLOGIES UTILISEES

Afin de mener une bonne conception et réalisation du projet de l’entreprise INGDEUX, plusieurs technologies ont été utilisés notamment :

## 1. Langages

* Le langage utilisé pour le développement de cette plateforme est le Java EE (*Java Enterprise Edition)*.

L'objectif majeur de Java EE est de faciliter le développement d'applications web robustes et distribuées, déployées et exécutées sur un serveur d'applications. Ce qui est le cas pour la plateforme de gestion et attribution des tâches du personnel de INGDEUX, qui se veut être une application web qui permettra au personnel de consulter leur tableau de bord ou qu’il soit et aussi aux administrateurs de gérer la plateforme.

Le choix de langage dépendant du cahier de charges mis à la disposition de l’équipe.

* Les langage html, css qui seront utilisés respectivement pour le formalisme et le design des pages de l’application.

## 2. Plateforme de développement

**Eclipse for JEE Developpers** qui est un environnement intégré de développement (IDE) pour le langage Java

## 3. Serveur d’applications

**Tomcat V 9.0** est un serveur d'applications Java.

4. Serveur de base de données

MySQL est un système de gestion de bases de données relationnelles SQL open source développé et supporté par Oracle. Il nous permet la création de notre base de données.

5. Technologies pour la vue

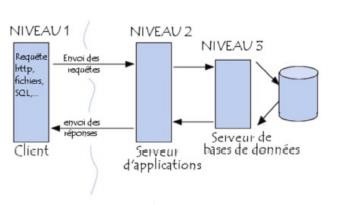


***Bootstrap*** est une collection d'outils utiles à la création du design (graphisme, animation et interactions avec la page dans le navigateur, etc.) de sites et d'applications web.

6. JSTL ***JSTL*** est l'acronyme de **Java server page Standard Tag Library**. C'est un ensemble de tags personnalisés développé sous la JSR 052 qui propose des fonctionnalités souvent rencontrées dans les JSP :

# II. Architecture de la solution

L’architecture utilisée pour l’application de INGDEUX est l’architecture **3-tiers.**

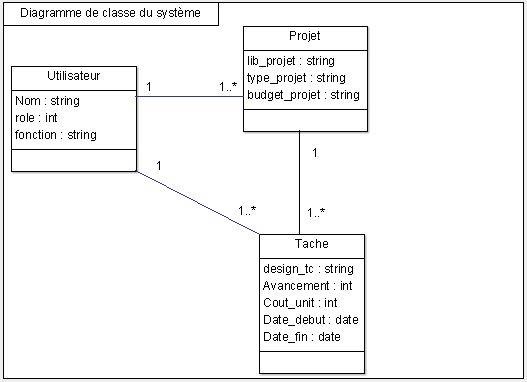


Le niveau 1 : qui correspond aux différents postes clients ou seront installés les clients web c’està-dire un navigateur web à l’exemple de Mozilla Firefox.

Le niveau 2 : correspond au serveur d’application qui est par occurrence **Tomcat 9** ce dernier nous permettant d’exécuter les différentes pages jsp de notre **vue**.

Le niveau 3 : correspondant ici au serveur de base de données qui par occurrence ici MySql nous permettant de créer notre base de données et les différentes relations entre les tables de celle-ci.

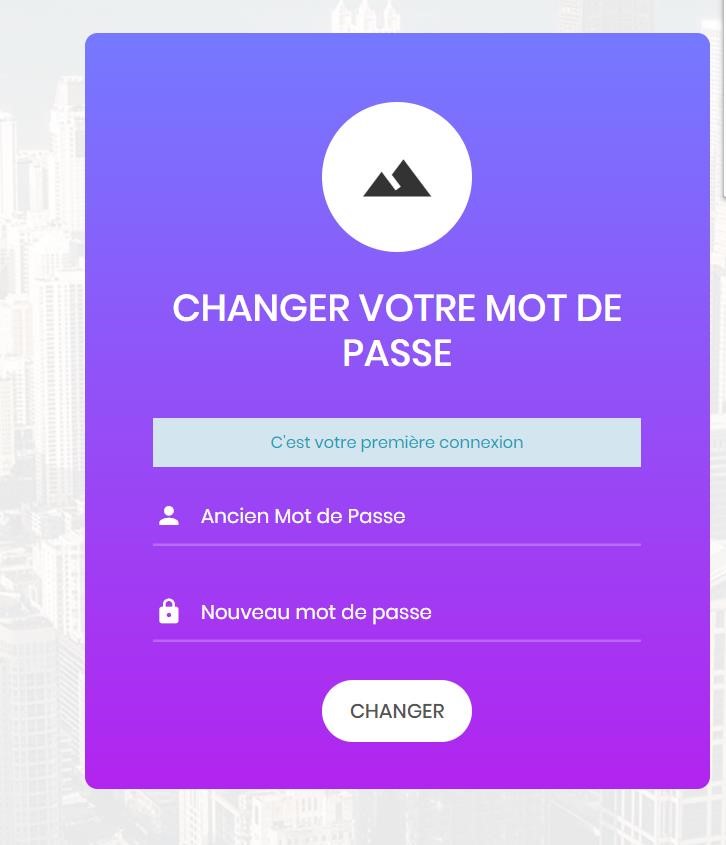
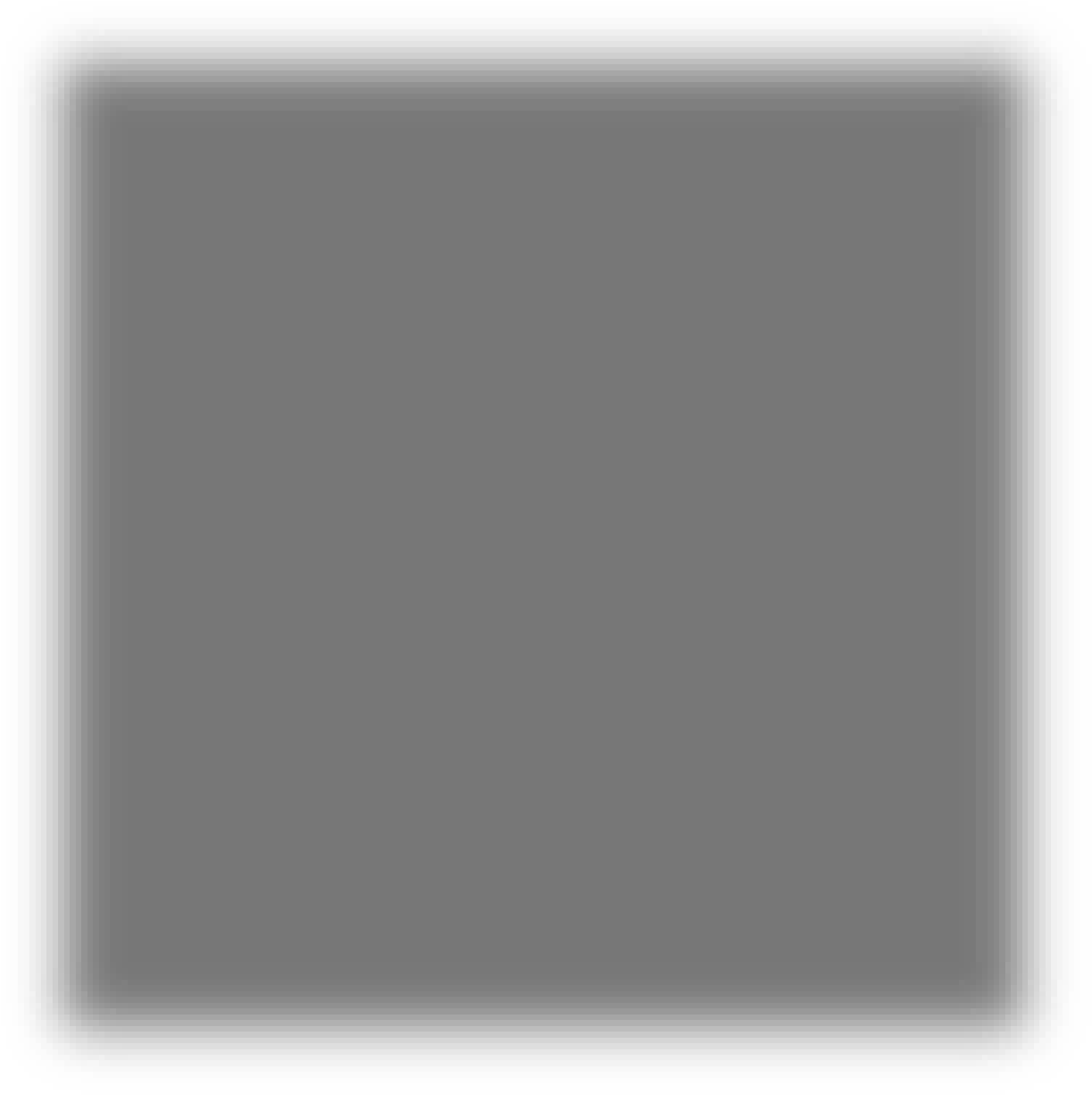
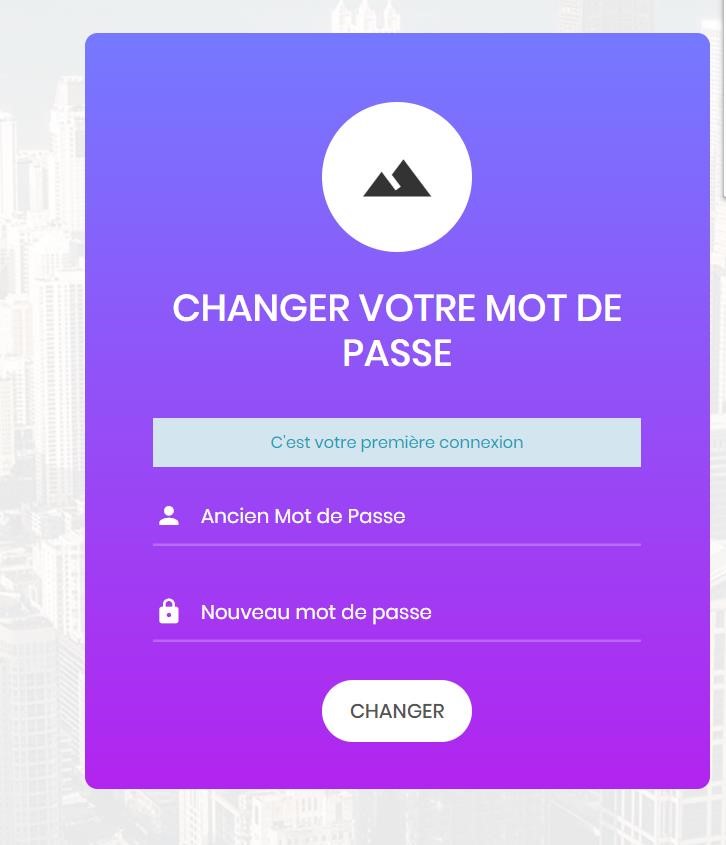
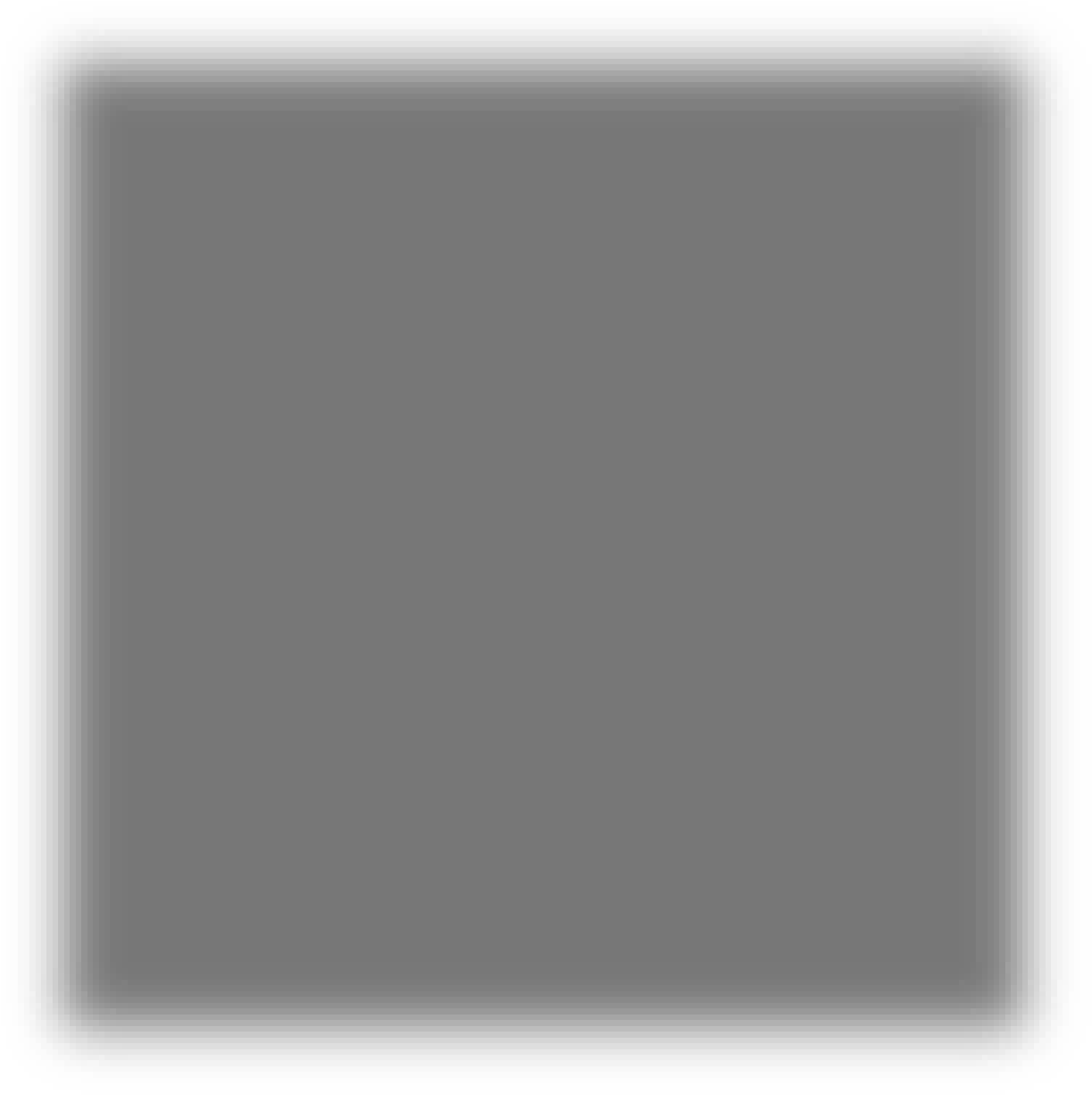
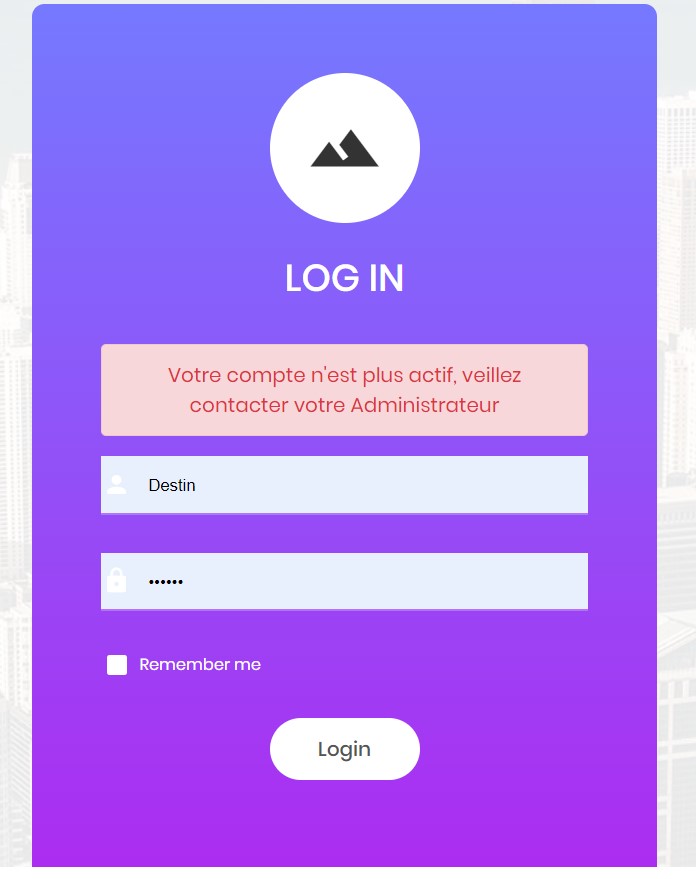
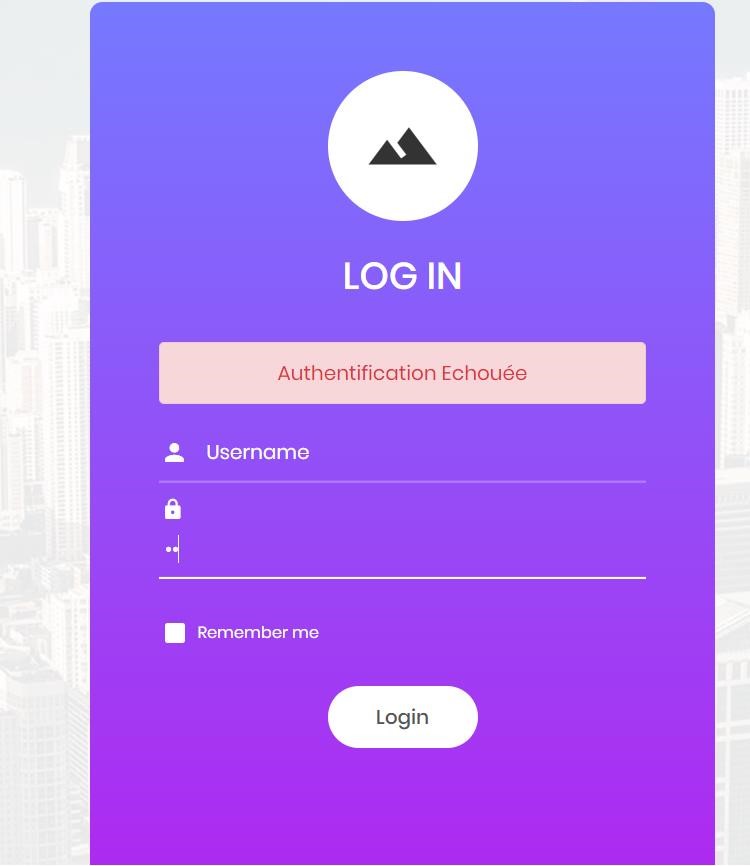
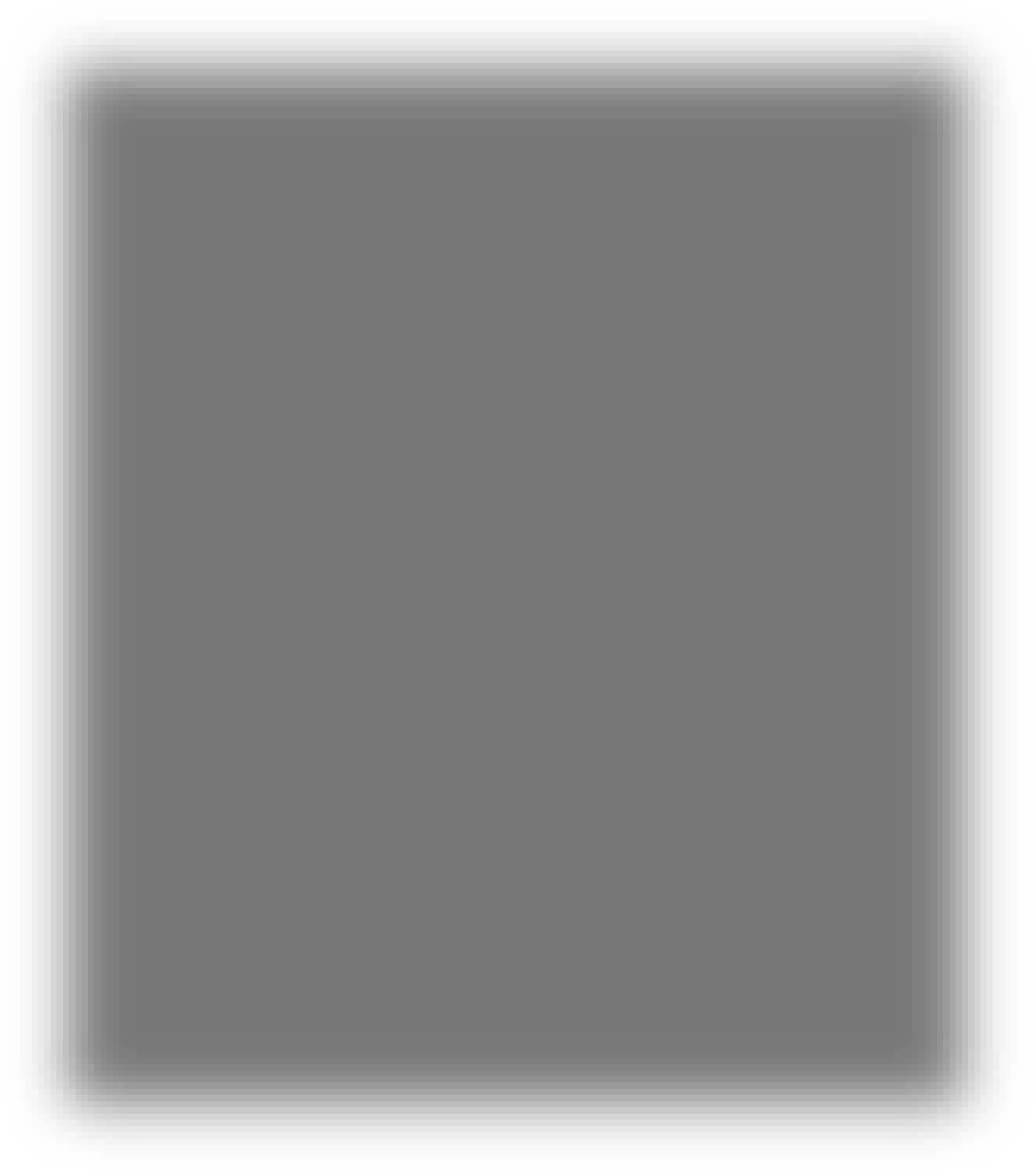
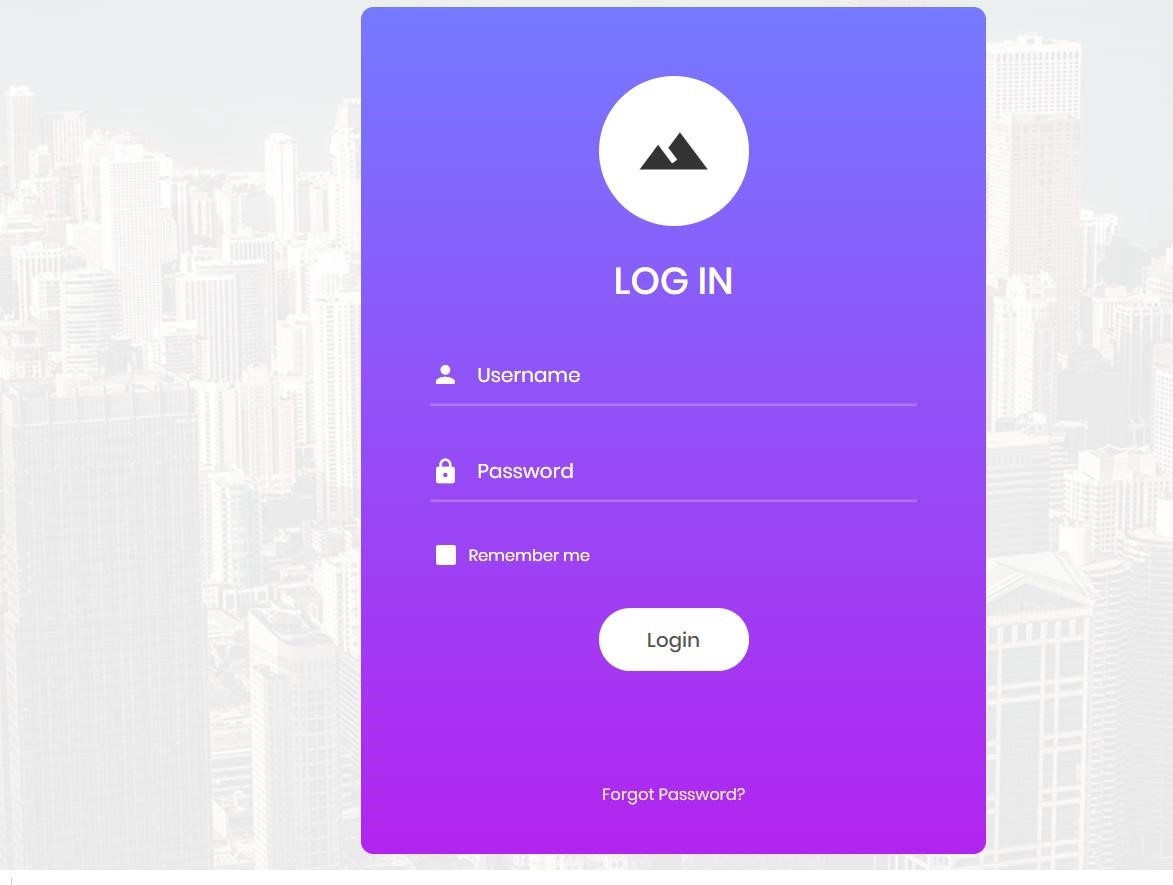
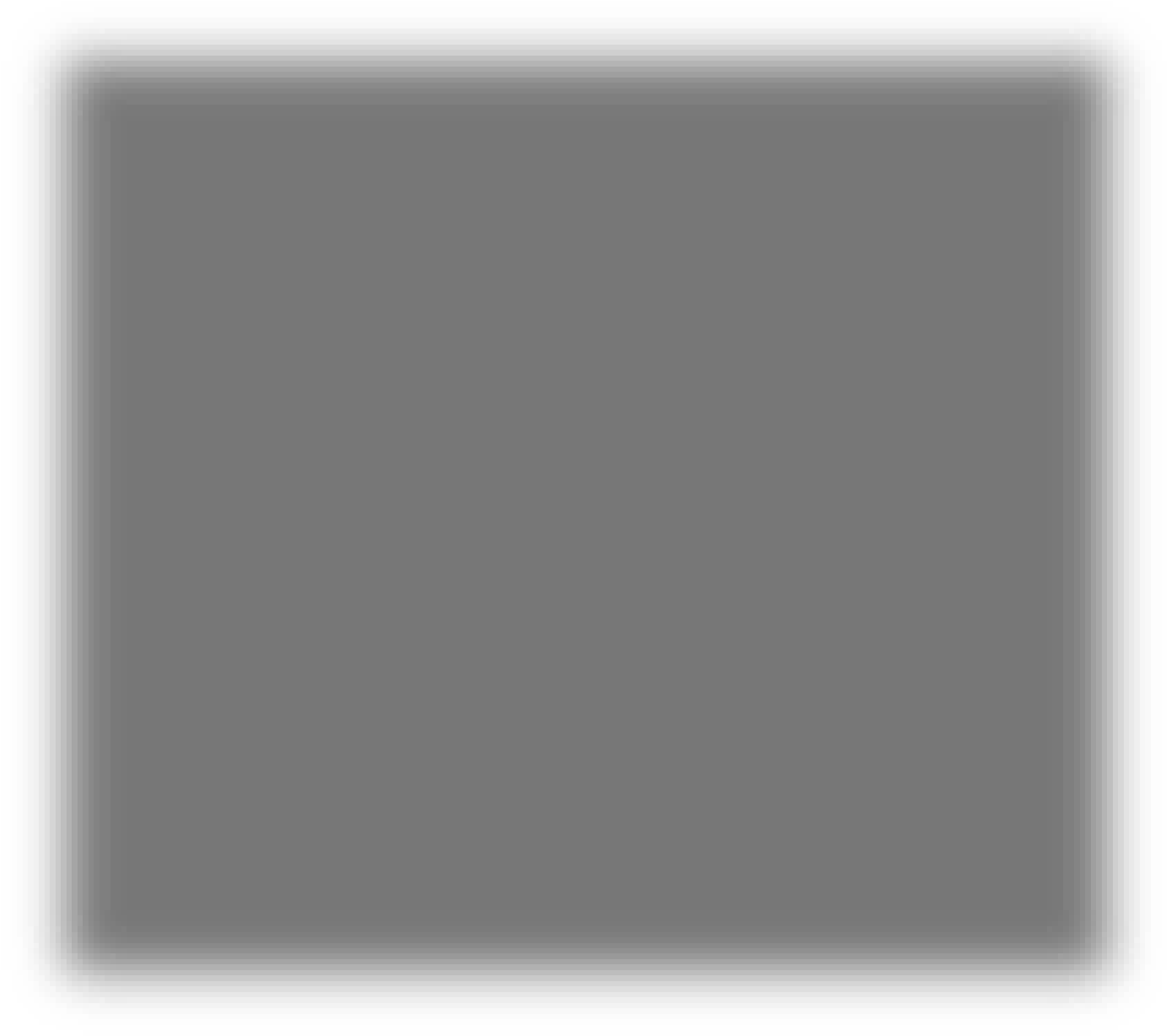
Notre base de données est constituée ainsi qu’il suit :



# III. METHODES UTILISEES

L’application étant codées suivant le modèle MVC, nous avons séparées les méthodes en fonction des différentes couches.

|  |  |  |
| --- | --- | --- |
| **Nom** | **Méthodes** | **Explications** |
| **Modèle** |  |  |
| **ProjetDAO** | getConnection() |  |
| insertProjet() | public boolean AnnulerProjet(String code\_proj) |
| public boolean  SuspendreProjet(String code\_proj) | public boolean TerminerProjet(String code\_proj) |
| public boolean  DemarerProjet(String code\_proj) | public boolean AffecterProjet(String code\_proj, int idAdmin) |
| public boolean RetirerProjet(String code\_proj, int ancienAdmin, int nouvelAdmin) | public Projet selectProjet(String code\_projet) |
| **TacheDAO** | public void insertTache(Tache tache, int id\_utilisateur, String code\_proj) | public boolean AnnulerTache(String code\_tc) |
| public boolean  SuspendreTache(String code\_tc) | *public boolean TerminerTache(String code\_tc)* |
| public boolean  DemarerTache(String code\_tc) | public boolean AffecterTache(String code\_tc, int idAdmin) |
| **Contrôleur** |  |  |
| **LoginServlet.java** | private void  showAdminDashBoard(int id, HttpServletRequest request,  HttpServletResponse response) | private List<Projet> loadTachesProjet(List<Projet> listProjet) |
| **UserServlet** | private void  showAdminDashBoard(int id, HttpServletRequest request,  HttpServletResponse response) | private List<Projet> loadTachesProjet(List<Projet> listProjet) |



Page de Login

A chaque première connexion,

on oblige l’utilisateur à changer

A chaque connexion, on vérifie si le compte

de

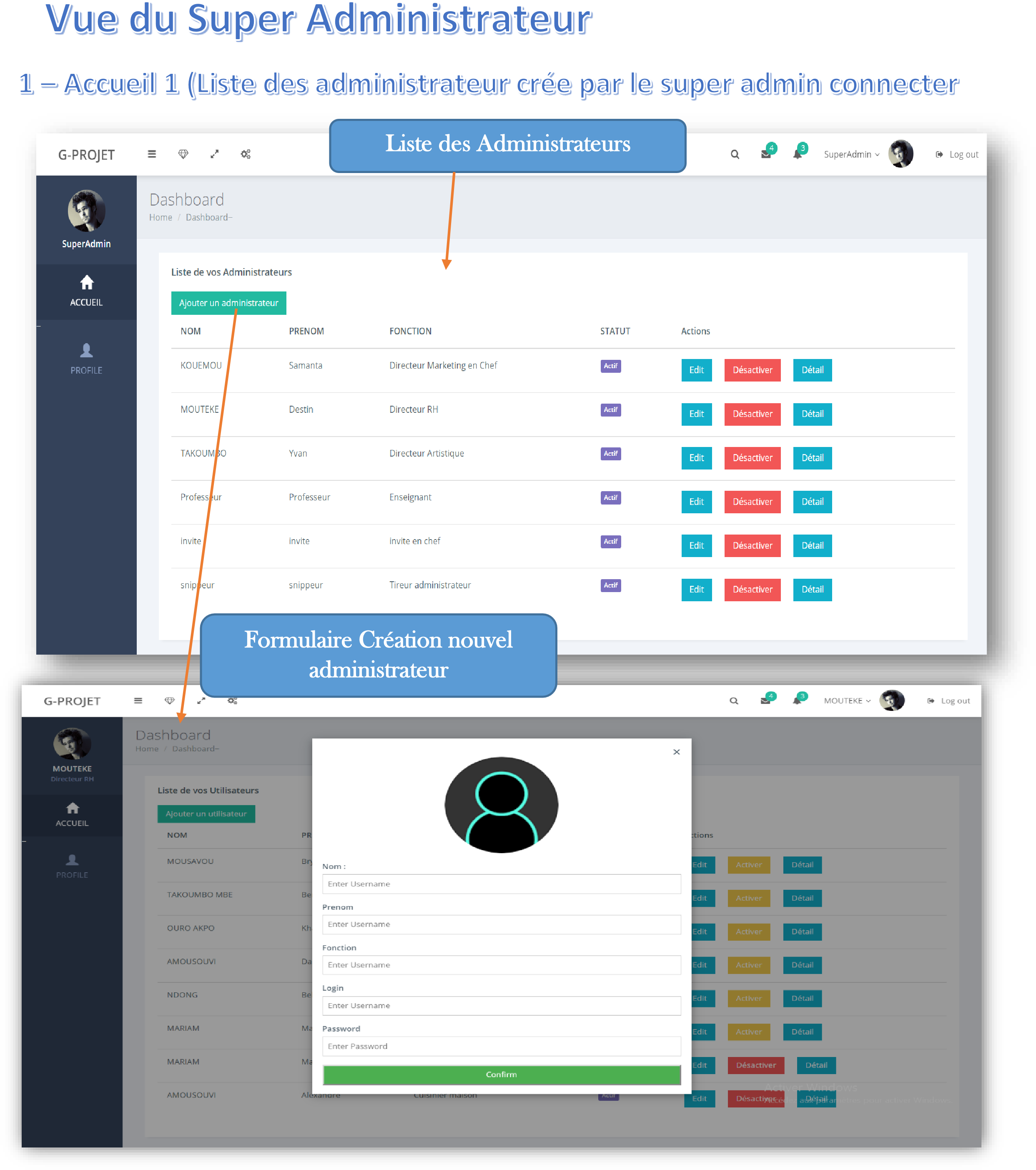
l’utilis

ateur est actif, sinon il ne se

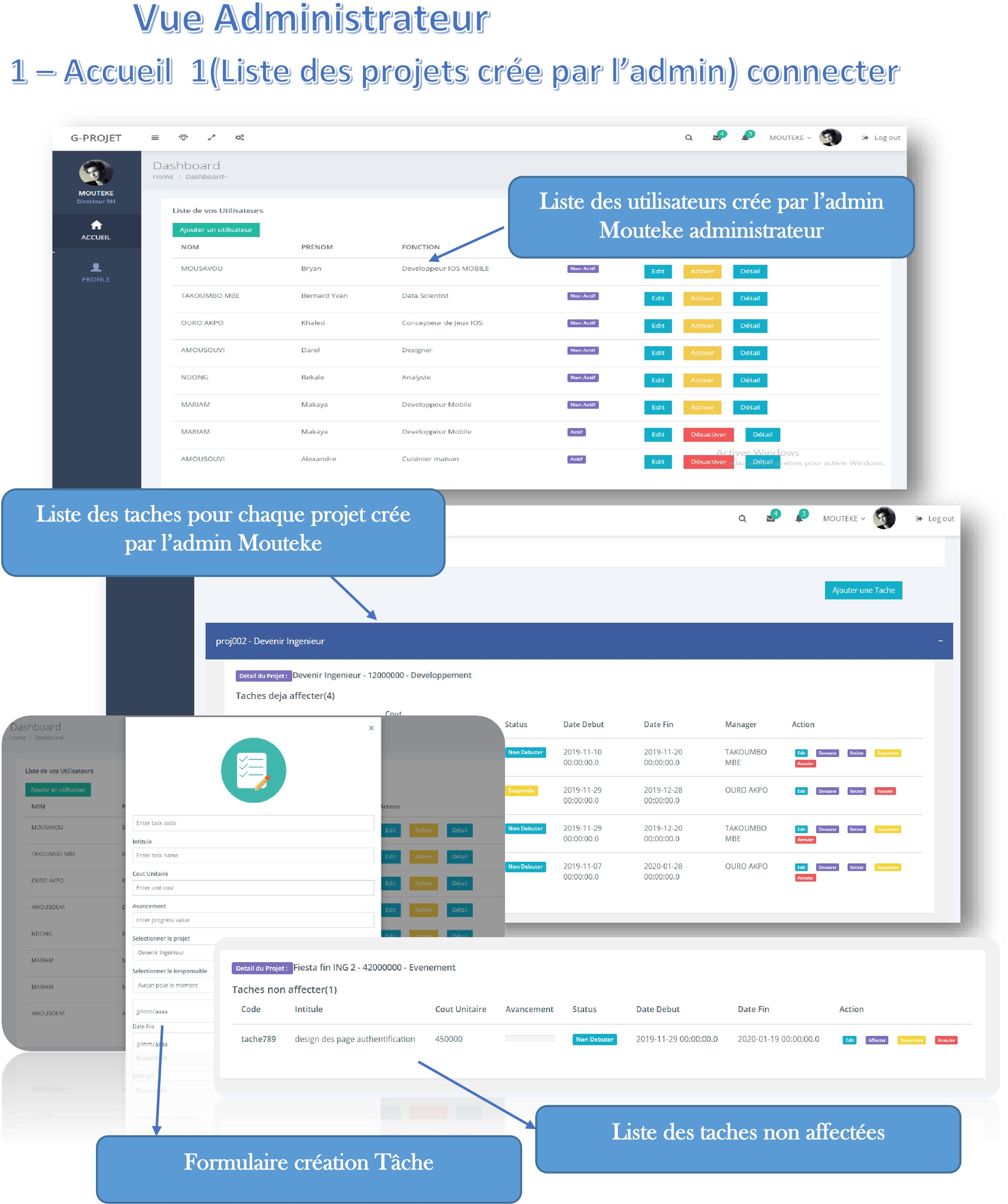
connecte pas

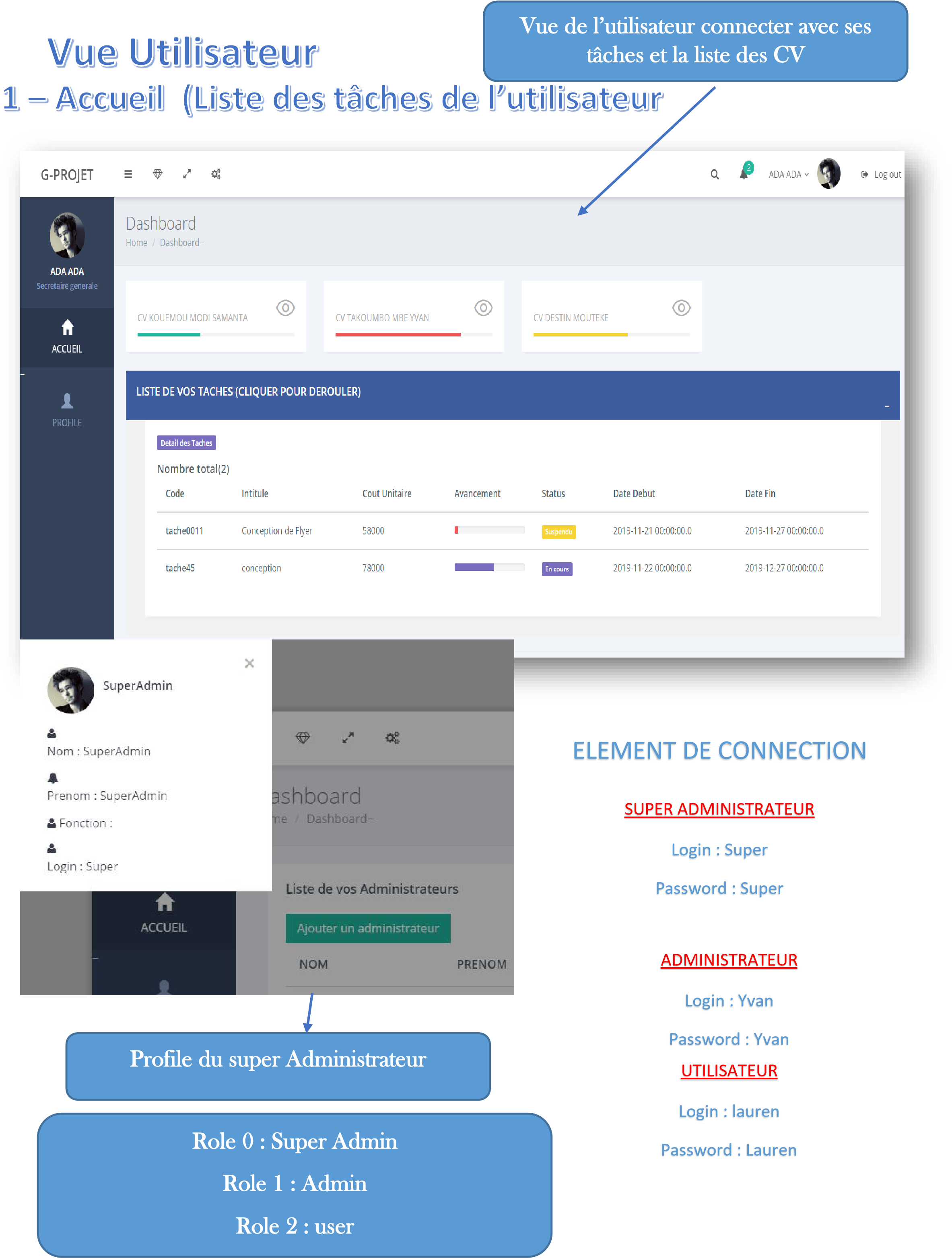
Login et/ou mot de passe

incorrect









# CONCLUSION

Parvenu au terme de notre travail qui consistait à mettre sur pied une plateforme de gestion et attribution des tâches de son personnel, nous avons dans un premier temps fait une analyse consistant à ressortir les différentes entités du système et les méthodes qui leur sont propres, par la suite nous avons créé la base de données, enfin implémenter les différentes classes en utilisant le modèle MVC (Modèle-Vue-Contrôleur). Nous avons eu des difficultés au début à charger nos styles Css, Js, images et autres éléments de la vue. Suite à notre compréhension du fonctionnement de l’environnement JAVA EE, nous avons pu constater que c’était un problème de route. Donc nous avons dans notre servelet mis **@WebServlet("/UserServlet")** et non plus **@WebServlet("/")** Ford de cette expérience nous en sommes ressortis avec bien de connaissance acquise et aptitudes à faire face à un travail en équipe.