



Projet Tutoré

---

## Daily Print

---

*Auteurs :*

Saliou BA  
Julien DAURAT  
Ariella LEVY  
Bastien MARAIS

*Encadrant :*

M. Fabrice HOGUIN

Version 1.0 du  
22 février 2018

# Table des matières

<b>Introduction</b>	<b>2</b>
<b>1 Cahier des charges</b>	<b>3</b>
1.1 Énoncé . . . . .	3
1.2 Pré-requis . . . . .	5
1.3 Priorité . . . . .	5
<b>2 Analyse des besoins</b>	<b>6</b>
2.1 Lecture du cahier des charges . . . . .	6
2.1.1 Identification des actions (fonctions), acteurs et objets du problème.	6
2.1.2 Identification des exigences . . . . .	7
2.2 Analyse des interactions logiciel/environnement . . . . .	8
2.2.1 Diagrammes des cas d'utilisations . . . . .	9
2.2.2 Spécifications des cas d'utilisations . . . . .	11
2.3 Exigence non fonctionnelle . . . . .	16
2.4 Exigence Organisationnelle et Métiers . . . . .	16
2.5 Glossaire . . . . .	16
<b>3 Dossier concepteur</b>	<b>17</b>
3.1 Spécifications détaillées de la structure du système . . . . .	17
3.1.1 Comportement . . . . .	21
3.1.2 Base de données . . . . .	33
3.2 Spécifications détaillées des interfaces utilisateurs . . . . .	35
3.2.1 Prototype de l'interface . . . . .	35

<b>Table des matières</b>	<b>1</b>
3.2.2 Définition de la charte graphique . . . . .	35
<b>4 Dossier architecture</b>	<b>37</b>
4.1 Vue modulaire . . . . .	37
4.2 Vue composant/Connecteur . . . . .	37
4.3 Vue d'attribution . . . . .	38
<b>Conclusion</b>	<b>40</b>

# Introduction

Le projet tutoré représente l'aboutissement de nos savoirs acquis lors de notre cursus par une application : la réalisation d'une solution pour répondre au besoin d'un client spécifique. Nous devons en effet réaliser une application servant à la gestion de la reprographie de l'IUT de Vélizy, projet proposé par notre client et professeur de l'IUT de Vélizy : **Monsieur Fabrice Huguin**.

Ce projet consiste donc en la création d'une application WEB qui vise à faciliter la gestion du système de reprographie de l'IUT de Vélizy. Celui-ci devra permettre une meilleure communication entre les clients, ici les professeurs/chercheurs et le service de reprographie. Pour le design du site Web nous utiliserons les langage de programme suivant :

- HTML langage de balisage.
- CSS pour l'ajout de style
- JavaScript pour dynamiser le site

Pour les fonctionnalités derrière le site

- PHP pour gérer les flux de données
- SQL pour gérer les données présente sur la base de données MySQL hébergée sur Hostinger.

Notre équipe composé de 4 membres. La répartition des tâches est la suivante :

- **BA Saliou** : Développeur, chef de projet adjoint.
- **DAURAT Julien** : Développeur et designer.
- **MARAIS Bastien** : Lead-développeur, chef de projet.
- **LEVY Ariella** : Concepteur, développeur.

Ce rapport est composé de 4 chapitre ; le cahier des charges, l'analyse des besoins, le dossier concepteur et le dossier architecture. A travers ces chapitre nous allons vous communiquer les problèmes et solutions apportées ainsi que les démarches utilisées pour répondre aux attentes du client.

# Chapitre 1

## Cahier des charges

Ce projet est une application web qui cherche à faciliter la gestion du système de reprographie de l'IUT de Vélizy. Celle-ci doit permettre une meilleure communication entre les clients, ici les professeurs et le service de reprographie. Dans ce document nous commencerons par présenter une description avancée du problème qui se pose avec le système de reprographie actuel, chose qui nous permettra de mettre en avant la solution envisagée en donnant les cas d'utilisations courantes pour les clients. Nous procéderons ensuite à une présentation des détails techniques en spécifiant les outils que nous utiliserons au cours de ce projet. Pour finir, il sera important de mettre en évidence les différentes contraintes présentes ainsi que les exigences du client, chose essentielle à la bonne réalisation du projet.

### 1.1 Énoncé

Le système de reprographie actuel n'est pas le plus efficace. Les professeurs ont deux possibilités :

- Faire une demande sous la forme d'un email via un formulaire pdf.
- Faire une demande directement à la reprographie avec un formulaire papier à remplir.

Avec le système actuel, les professeurs ne sont pas informés du statut de leur demandes d'impressions ; en effet des déplacements sont nécessaires pour obtenir des informations. Ce n'est donc pas un système extrêmement pratique et efficace. L'objectif sera donc de proposer une alternative à ce projet afin d'améliorer le confort de l'utilisateur ainsi que de faciliter le travail de la reprographie. Le projet sera créé dans le cadre du remplacement du système actuel par une application Web qui permettra d'optimiser le temps de traitement des demandes et faciliter la création de celles-ci. Le projet devrait permettre de créer un système où les professeurs peuvent demander des photocopies, en spécifiant des informations comme le nombre de copies, la façon dont elles sont photocopiées (recto,

recto/verso...), les finitions (agrafe, livret...). Ils auront la possibilité de suivre l'état des demandes de photocopies (reçues par la reprographie, traitées ou prêtes à être récupérées). Les personnes qui travaillent à la reprographie recevront les demandes dans l'ordre de leur arrivée, et au fur et à mesure que ces demandes seront traitées, elles auront la possibilité d'en modifier le statut, pour informer les professeurs de l'état d'avancement de ces dernières par l'envoi d'un email ou d'une notification. Les différents statuts possibles pour accéder au site sont les suivants : "Professeur" et "Reprographie". Un professeur pourra donc en se connectant faire une demande d'impression qui sera gérée par les personnes qui travaillent à la reprographie. Pour que les demandes soient traitées plus facilement grâce à un système de tri et pour faire des études statistiques, les professeurs seront classés en fonctions de leur principal département d'enseignement (INFO, MMI, GEII, RT, ISTY...). Ces statistiques pourront, à terme, permettre une meilleure gestion et répartition des tâches au sein de la reprographie. De plus, notre projet permettra de centraliser les demandes. Un exemple de cas d'utilisation du point de vue d'un professeur serait :

- *Le professeur se connecte sur le site grâce à son email et son mot de passe.*
- *Il ne voit que ses demandes et a la possibilité de créer autant de requêtes d'impression qu'il le souhaite.*
- *Pour créer une nouvelle demande, il doit accéder à l'onglet "Créer une nouvelle demande", sur cette page il doit entrer différentes informations nécessaires, tel que le nombre de copies, la façon de photocopier, la finition, le fichier à imprimer et également signaler les éventuelles publications protégées. Une fois les informations entrées, il n'a plus qu'à valider en appuyant sur le bouton en bas de la page.*
- *Une fois cette demande envoyée, elle sera transférée au service de reprographie.*
- *Le professeur peut afficher à tout moment ses demandes depuis l'onglet "Mes demandes".*

Un exemple de cas d'utilisation du point de vue d'un membre de la reprographie serait :

- *Le membre de la reprographie se connecte sur le site grâce à son email et son mot de passe.*
- *Il a accès à toutes les demandes et aura la possibilité de filtrer celles-ci en fonction du département auquel appartient le professeur demandeur.*
- *Une fois qu'une demande est sélectionnée, il pourra accéder à ces informations et donc traiter cette dernière tout en mettant à jour progressivement son statut d'avancement.*
- *Lorsque cette demande est traitée, le membre de la reprographie met à jour le statut de celle-ci sur "Terminée", ceci entraînera l'envoi d'un email automatique à l'adresse du professeur pour le prévenir.*

## 1.2 Pré-requis

Pour la réalisation de ce projet nous utiliserons différents outils :

- Langages **WEB : PHP, HTML, CSS et JavaScript**
- Base de données : **MySQL**
- Langages de base de données : **PL/SQL et SQL**
- Support de fonctionnement : **Ordinateurs et mobiles**

L'interface devra être pratique et fonctionnelle.

## 1.3 Priorité

Nous devons rendre la partie conception, qui comprend le cahier des charges, le diagramme de classe, le diagramme de cas d'utilisation et le diagramme de séquence pour le 23 février 2018.

# Chapitre 2

## Analyse des besoins

Ce document présente les différentes facettes des besoins exprimés par l'utilisateur. On cherche dans ce projet à créer un système de ticketing pour un système de reprographie. Le but de cette annexe est de définir clairement les besoins du client afin d'y répondre au mieux. Dans une première partie, nous étudierons le cahier des charges que nous avons réalisés, en collaboration avec le client et en feront une lecture complète de celui-ci. Puis, nous étudierons les différents cas d'utilisations possibles que nous avons pu identifier au sein du système.

### 2.1 Lecture du cahier des charges

#### 2.1.1 Identification des actions (fonctions), acteurs et objets du problème.

Acteurs	Actions
Service reprographie	<ul style="list-style-type: none"><li>— Reçoit les demandes</li><li>— Traite les demandes</li><li>— Gestion du système de statistiques</li></ul>
Professeur	<ul style="list-style-type: none"><li>— Faire une demande</li><li>— Annuler une demande</li></ul>
Recherche	<ul style="list-style-type: none"><li>— Faire une demande</li><li>— Annuler une demande</li><li>— Choisir les spécifications de la demande</li></ul>



### 2.1.2 Identification des exigences

L'identification des exigences est un point permettant de définir les futurs fonctionnalités de notre site web. Celles-ci seront classées par ordre d'importance :

- **mineur** : dans le cas où l'implémentation de cette fonctionnalité n'est pas quint-essentielle au site.
- **majeur** : dans le cas où l'implémentation de cette fonctionnalité est importante au bon développement et au respect des attentes.
- **critique** : dans le cas où il est nécessaire d'ajouter cette fonction sans quoi le site web ne répond pas aux attentes du client.

Ces éléments permettront de définir les étapes à suivre en amont des autres, afin de pouvoir rendre en temps et en heure une solution dans laquelle le client saurait reconnaître toutes ses attentes.

Dans le tableau ci-dessous nous utilisons des Acteurs différents :

- **Reprographie** : Correspond aux membres du service de reprographie.
- **Administrateur** : Correspond à l'administrateur du site web.
- **Utilisateur** : Correspond aux utilisateurs c'est-à-dire les professeurs et chercheurs.

Numéro	Exigences	Acteur	Importance
1	Pouvoir faire une demande d'impression avec les options déjà existante.	Utilisateur	Critique
2	Gestion des demandes entrantes	Reprographie	Critique
3	Récupération du mot de passe	Utilisateurs	Mineur
4	Confirmer les inscriptions	Administrateur	Critique
5	Visualisation des demandes	Utilisateurs	Critique
6	Afficher des statistiques pour aider le service reprographie à gérer son service	Service reprographie	Mineur
7	Système ergonomique	Utilisateurs	Majeur
8	Pouvoir supprimer des utilisateurs	Administrateur	Majeur
9	Pouvoir modifier ses informations	Utilisateurs	Majeur
10	Pouvoir visualiser ses demandes et leur statut d'avancement	Utilisateurs	Critique
11	Avoir accès à un guide d'utilisation du site web	Utilisateurs	Majeur

## 2.2 Analyse des interactions logiciel/environnement

Cette annexe vise à décrire les interactions entre notre site web et son environnement. Le système d'exploitation utilisé n'est pas important, ce site web sera hébergé sur hostinger et utilisera un serveur apache : il sera donc disponible sur tous systèmes d'exploitation. En terme de compatibilité, ce site Web fonctionnera avec de nombreux navigateurs, nous listons dans le tableau ci-dessous l'efficacité de celui-ci sur les plus utilisés.

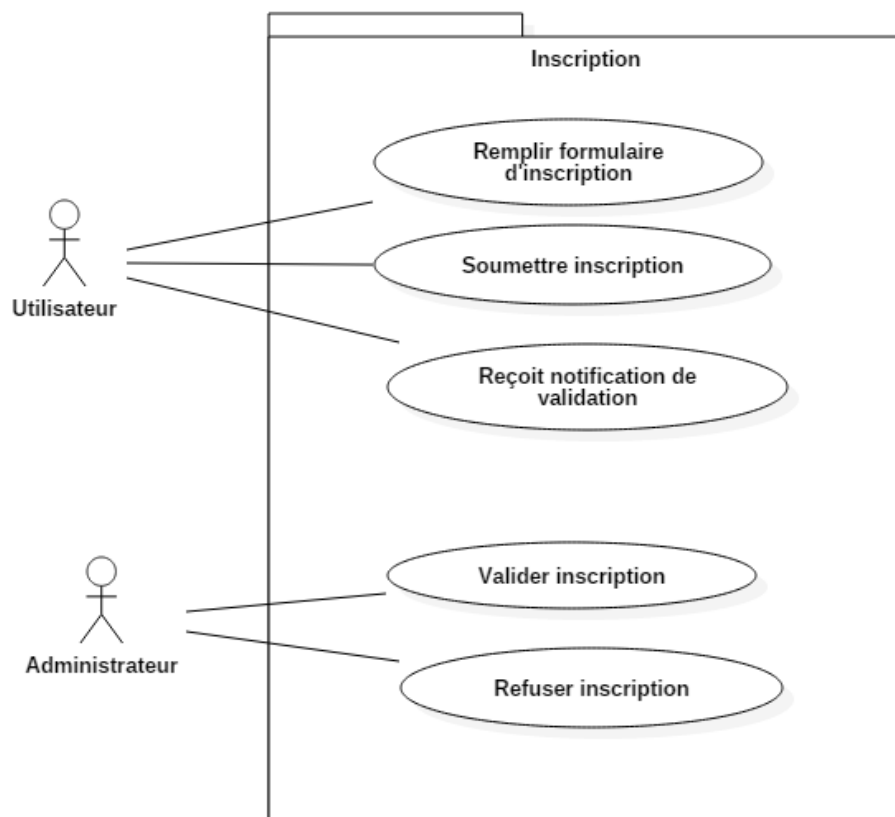
Notons que le terme **accessibilité** correspond à la facilité d'accès pour les personnes présentant un handicap en respectant les multiples règles énoncées par les normes d'accessibilité. Elle est vérifiée à l'aide de différents outils notamment Axe Le terme **responsive** correspond quand-à-lui à l'adaptation du site web sur différents supports (tablettes, mobiles ..) possédant des tailles d'écran différentes.

Navigateur	Compatibilité	Accessibilité	Responsive
Chrome	Excellente	Validée	Oui
Firefox	Optimale	Validée	Oui
Safari	Optimale	Validée	Oui
I. Explorer	Limitée	Validée	Oui

### 2.2.1 Diagrammes des cas d'utilisations

Cette sous-section correspond aux diagrammes des cas d'utilisation, c'est la représentation des besoins des acteurs dans une situation particulière. Selon les utilités du site web, les cas d'utilisations nécessaires sont ceux liés à l'inscription et la création de demande car ce sont ces cas qui vont impliquer différents acteurs.

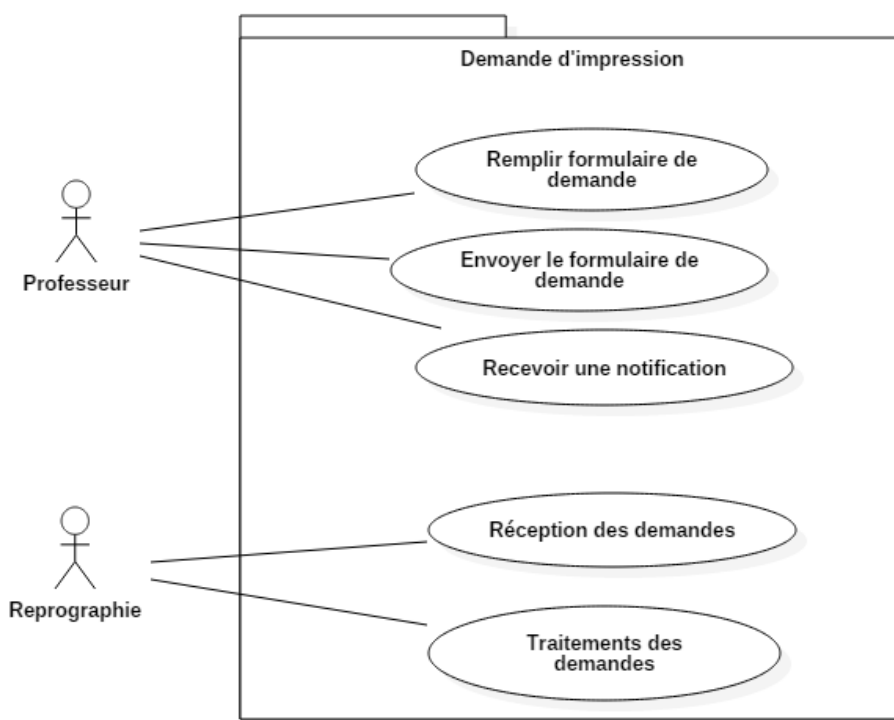
FIGURE 2.1 – Diagramme de cas d'utilisation de l'inscription



Ce diagramme ci-dessus nous permet de montrer les différentes actions des acteurs impliqués dans la situation suivante : **inscription**. Nous pouvons voir que les deux acteurs concernés sont :

- **l'utilisateur** (qui représente les professeurs et chercheurs mais aussi les membres du service reprographie) : Ceux-ci auront comme actions le fait de remplir/soumettre inscription et de recevoir une notification en cas de validation
- **l'administrateur** : qui aura comme action le fait de valider ou refuser l'inscription.

FIGURE 2.2 – Diagramme de cas d'utilisation de la création d'une demande



Ce diagramme ci-dessus nous permet de montrer les différentes actions des acteurs impliqués dans la situation suivante : **faire une demande**. Nous pouvons voir que les deux acteurs concernés sont :

- **le professeur** (ou chercheur) : Il aura comme action de remplir le formulaire de demande, l'envoyer ainsi et recevoir une notification du site lorsque la demande aura changée de statut.
- **le service reprographie** : qui aura comme action la réception des demandes ainsi que leur traitement (en soit ce dernier point correspond (à l'échelle du site) à au changement de statut de la demande).

## 2.2.2 Spécifications des cas d'utilisations

### Cas d'utilisation Administrateur

#### Cas d'utilisation 1 : Gestion des demandes d'inscriptions

**Contexte d'utilisation :** Cas d'utilisation du système de gestion des demandes d'inscription utilisé par l'administrateur.

**Portée :** Interface de validation.

**Niveau :** utilisateur / stratégique (dépend du temps de réaction de l'administrateur).

**Acteur principal :** Administrateur.

**Intervenants et intérêts :** Administrateur, utilisateur.

**Scénario nominal :**

1. L'administrateur insère son adresse mail.
2. L'administrateur insère son mot de passe.
3. Les données sont bonnes, il est rédigé vers son espace de gestion.
4. L'administrateur reçoit les demandes d'inscriptions déroulées entre sa dernière connexion et celle actuelle.
5. L'administrateur valide les demandes d'inscriptions à l'aide du bouton de validation.
6. Notification automatique envoyée à l'utilisateur.
7. L'administrateur se déconnecte.

**Scénario annexe :**

1. L'administrateur insère son adresse mail.
2. L'administrateur insère son mot de passe.
3. Les données ne sont pas bonnes, l'action recommence.

#### Cas d'utilisation 2 : non-validation des inscriptions

**Contexte d'utilisation :** Cas d'utilisation du système de gestion des demandes d'inscription utilisé par l'administrateur dans le cas où l'inscription n'est pas validée.

**Portée :** Interface de validation.

**Niveau :** utilisateur / stratégique (dépend du temps de réaction de l'administrateur).

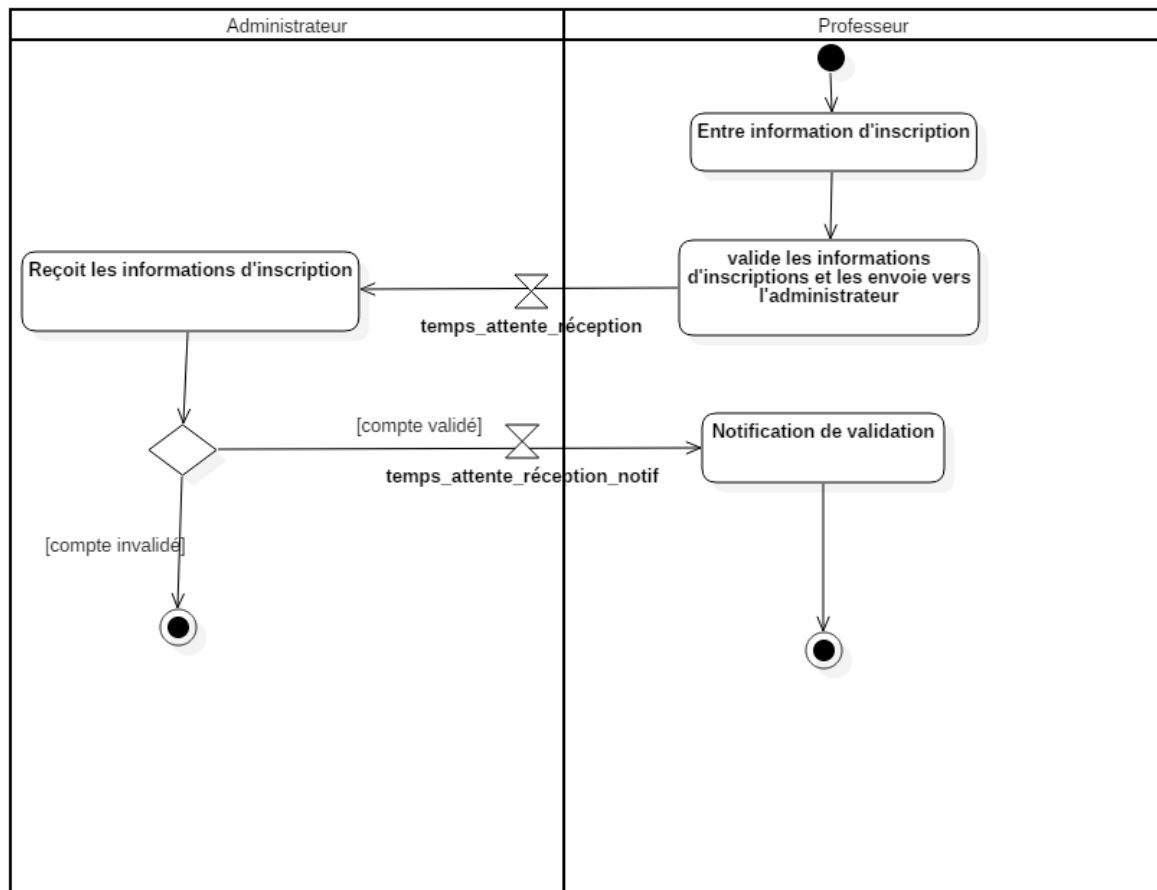
**Acteur principal :** Administrateur.

**Intervenants et intérêts :** Administrateur, utilisateur.

**Scénario nominal :**

1. L'administrateur insère son adresse mail.
2. L'administrateur insère son mot de passe.
3. Si les données sont bonnes, il est rédigé vers son espace de gestion.
4. L'administrateur invalide les demandes d'inscriptions.
5. L'administrateur se déconnecte.

FIGURE 2.3 – Diagramme d'activité représentation du système de gestion de l'administrateur.



### Cas d'utilisation Utilisateurs

#### **Cas d'utilisation 1 : Envoi d'une demande**

**Contexte d'utilisation :** Cas d'utilisation du système de création des demandes utilisé par les professeurs.

**Portée :** Interface de création des demandes

**Niveau :** Utilisateur.

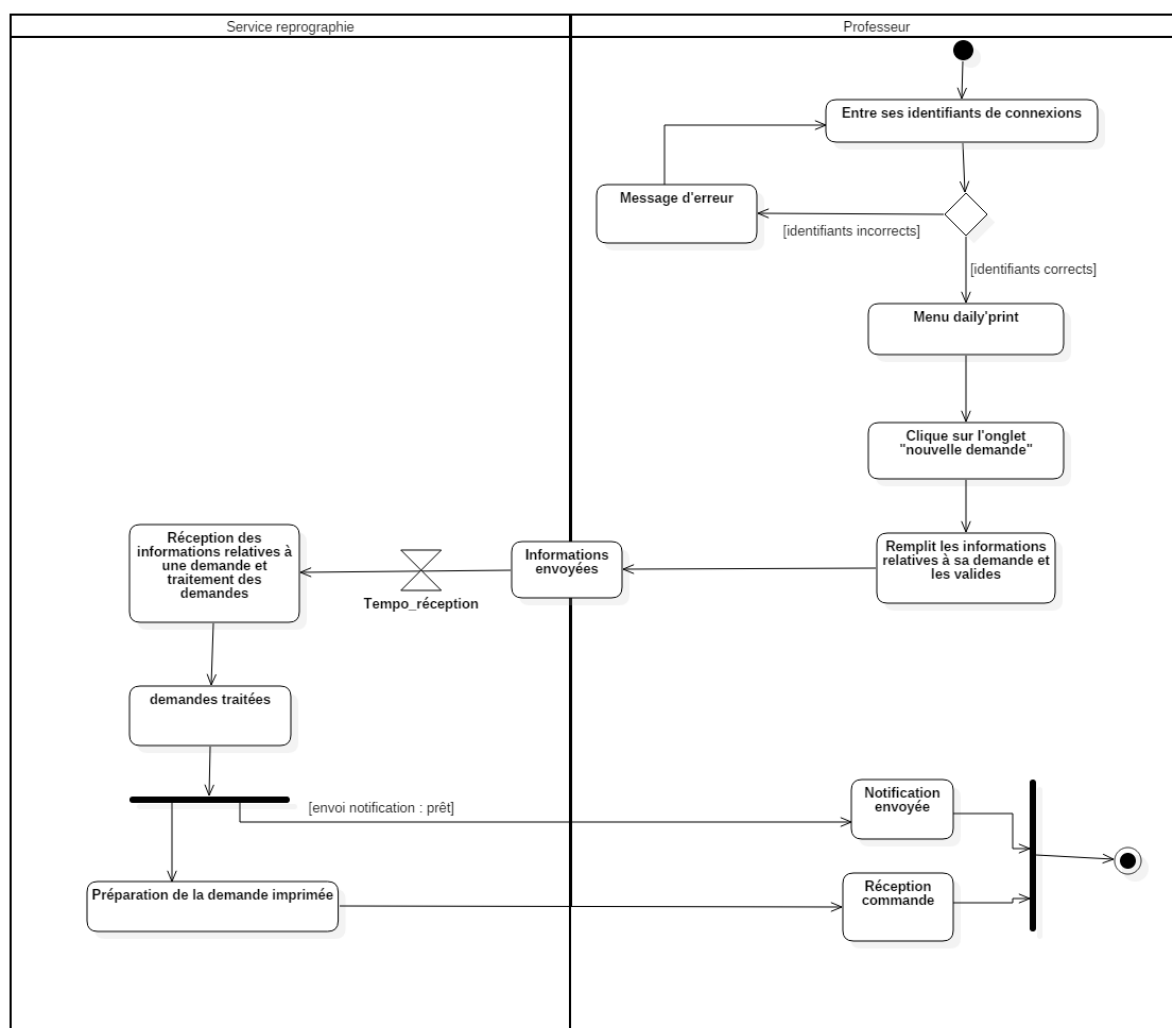
**Acteur principal :** Utilisateur..

**Intervenants et intérêts :** Utilisateur, service reprographie.

**Scénario nominal :**

1. L'utilisateur insère son adresse mail
2. L'utilisateur insère son mot de passe
3. Si les données sont bonnes, il est redirigé vers son espace personnel
4. L'utilisateur se dirige sur le panel de création des demandes
5. L'utilisateur sélectionne les informations pour sa demande
6. L'utilisateur valide sa demande qui est envoyée au service de reprographie
7. temporisation
8. Le service de reprographie reçoit la demande
9. Le service de reprographie traite la demande
10. temporisation
11. L'utilisateur reçoit une notification sur son mail
12. temporisation
13. L'utilisateur récupère le résultat de sa demande

FIGURE 2.4 – Diagramme d'activité représentation la création et traitement des demandes.





**Cas d'utilisation 2 : Inscription validée.**

**Contexte d'utilisation :** Cas d'utilisation où l'inscription de l'utilisateur est validée.

**Portée :** Site web

**Niveau :** Utilisateur.

**Acteur principal :** Utilisateur..

**Intervenants et intérêts :** Utilisateur.

**Scénario nominal :**

1. L'utilisateur se dirige sur l'espace d'inscription
2. L'utilisateur insère un nom
3. L'utilisateur insère un prénom
4. L'utilisateur insère une adresse mail valide
5. L'utilisateur insère un mot de passe et le confirme
6. L'utilisateur choisit un statut
7. L'utilisateur spécifie son département d'appartenance
8. L'utilisateur valide son inscription
9. Le système affiche un message lui expliquant que sa demande d'inscription est en cours
10. Temporisation inconnue
11. Si l'administrateur valide la demande d'inscription l'utilisateur reçoit une notification par mail

Nous ne faisons pas de diagramme d'activité car celui décrivant ce cas d'utilisation est représenté plus haut dans la *figure 2.3* donnant une indication claire du système global de l'inscription. Nous décrivons ce mécanisme du point de vu utilisateur plus tard lors de l'étude du comportement du système dans l'annexe de *conception*

## 2.3 Exigence non fonctionnelle

Une exigence non-fonctionnelle est une exigence qui caractérise une propriété (qualité) désirée du système telle que sa performance, sa robustesse, sa convivialité, sa maintenabilité, etc..

Dans le cadre de notre site web DailyPrint, ces exigences non-fonctionnelles correspondront par exemples aux droits d'accès.

Il est important pour chacun de respecter son univers propre :

- Le professeur/chercheur ne pourra accéder qu'à SES demandes et n'aura aucune visibilité sur celles des autres
- L'administrateur n'aura aucune visibilité sur les demandes, son compte n'aura pour fonction que les flux entrants (inscription) et sortant (suppression) d'utilisateurs.
- Le service reprographie aura une visibilité sur toutes les demandes parvenues ainsi que le statut d'avancée, mais n'aura en aucun cas le droit de suppression, seulement le droit de modifier le statut.

## 2.4 Exigence Organisationnelle et Métiers

Les exigences sont définies par une organisation : l'IUT de Vélizy et son personnel.

Il est nécessaire de respecter ces exigences, créer un système viable permettant de gérer la reprographie et disponible sur les navigateurs de l'IUT. Un site web pouvant être utilisé sur un serveur de l'IUT (d'où le choix d'Apache)

## 2.5 Glossaire

**Reprographie** : *Ensemble des procédés permettant la reproduction directe d'un document*

**Ticketing** : *Utilisateur envoi une demande spécifique à une plateforme et les membres gérant la plateforme gèrent ceux-ci une fois reçu.*

**Accessibilité** : *facilité d'accès pour les personnes présentant un handicap en respectant les multiples règles énoncées par les normes d'accessibilité. Elle est vérifiée à l'aide de différents outils notamment Axe*

**Responsive** : *l'adaptation du site web sur différents supports (tablettes, mobiles ..) possédant des tailles d'écran différentes.*

**Implémentation** : *Passer de la version théorique d'une fonctionnalité à sa version développée en codée*

# Chapitre 3

## Dossier concepteur

Le dossier concepteur est destiné aux programmeurs qui veulent comprendre la méthode de conception appliquée, le raisonnement suivi, les choix faits et les raisons de ces choix. Ce dossier montre le modèle de conception, dans lequel on y ajoute des concepts informatiques présents dans les outils, les langages de programmation ou les plate-formes de développement. Ainsi, dans ce rapport, nous étudions la Spécification détaillée de la structure du système, le Comportement, la Spécification détaillée des interfaces utilisateurs, le Modèle de réalisation, la Politique et stratégie et enfin nous aborderons les Choix des outils à utiliser.

### 3.1 Spécifications détaillées de la structure du système

Ce schéma représente l'organisation des pages de notre application et les différentes interactions qu'elles possèdent les unes avec les autres :

#### Les pages communes à tous

- **la page `index.php`** : C'est la page d'accueil de l'application dans laquelle l'utilisateur (quel qu'il soit pourra se connecter). Elle permet un accès à la page *accueil.php* si la connexion est effective. Autrement elle permet un accès *inscription.php* et *oubli.php*.
- **la page `inscription.php`** : Elle comporte un formulaire d'inscription qui permet à tout individu de faire une demande de création de compte. Étant donné qu'il faut une validation de l'inscription par l'administrateur, cette page renvoi (à l'aide d'un bouton retour) vers *index.php* si l'on se résigne ou alors affiche un message d'information.
- **la page `oubli.php`** : Cette page permet, en cas d'oubli de mot de passe, de faire une demande de changement de celui-ci. Elle ne renvoie vers aucune autre page, hormis *index.php* à l'aide du bouton retour.

### Les pages accessibles par les professeurs/chercheurs

- **la page accueil.php** : C'est la page d'accueil de l'application qui nous affiche une petite indication sur ce que permet de faire ce site (documentation). Elle renvoie grâce à une barre de navigation aux pages *Visual.php*, *new\_demande.php* et *aide.php* et aussi vers *param.php* et *index.php* en cas de déconnexion.
- **la page visual.php** : C'est une des pages centrales de l'application, elle permet à l'utilisateur de visualiser ses demandes ainsi que leur statut d'avancement. Elle renvoie grâce à une barre de navigation aux pages *accueil.php*, *new\_demande.php* et *aide.php* et aussi vers *param.php* et *index.php* en cas de déconnexion.
- **la page new\_demande.php** : c'est une des pages centrales de l'application puisqu'elle permet à l'utilisateur de faire ses demandes d'impressions et de spécifier leur caractère. Elle renvoie grâce à une barre de navigation aux pages *accueil.php*, *Visual.php* et *aide.php* et aussi vers *param.php* et *index.php* en cas de déconnexion.
- **la page aide.php** : Elle contient les informations relatives à l'application, une sorte de page de documentation permettant à l'utilisateur de comprendre le fonctionnement de l'application. Elle renvoie grâce à une barre de navigation aux pages *accueil.php*, *new\_demande.php*, *Visual.php* et aussi vers *index.php* en cas de déconnexion.
- **la page param.php** : Elle contient un formulaire permettant à l'utilisateur de modifier certains de ses paramètres comme son mot de passe et d'afficher les informations de l'utilisateur. Le retour nous permet d'aller à la page *accueil.php*.

### Les pages accessibles par l'administrateur

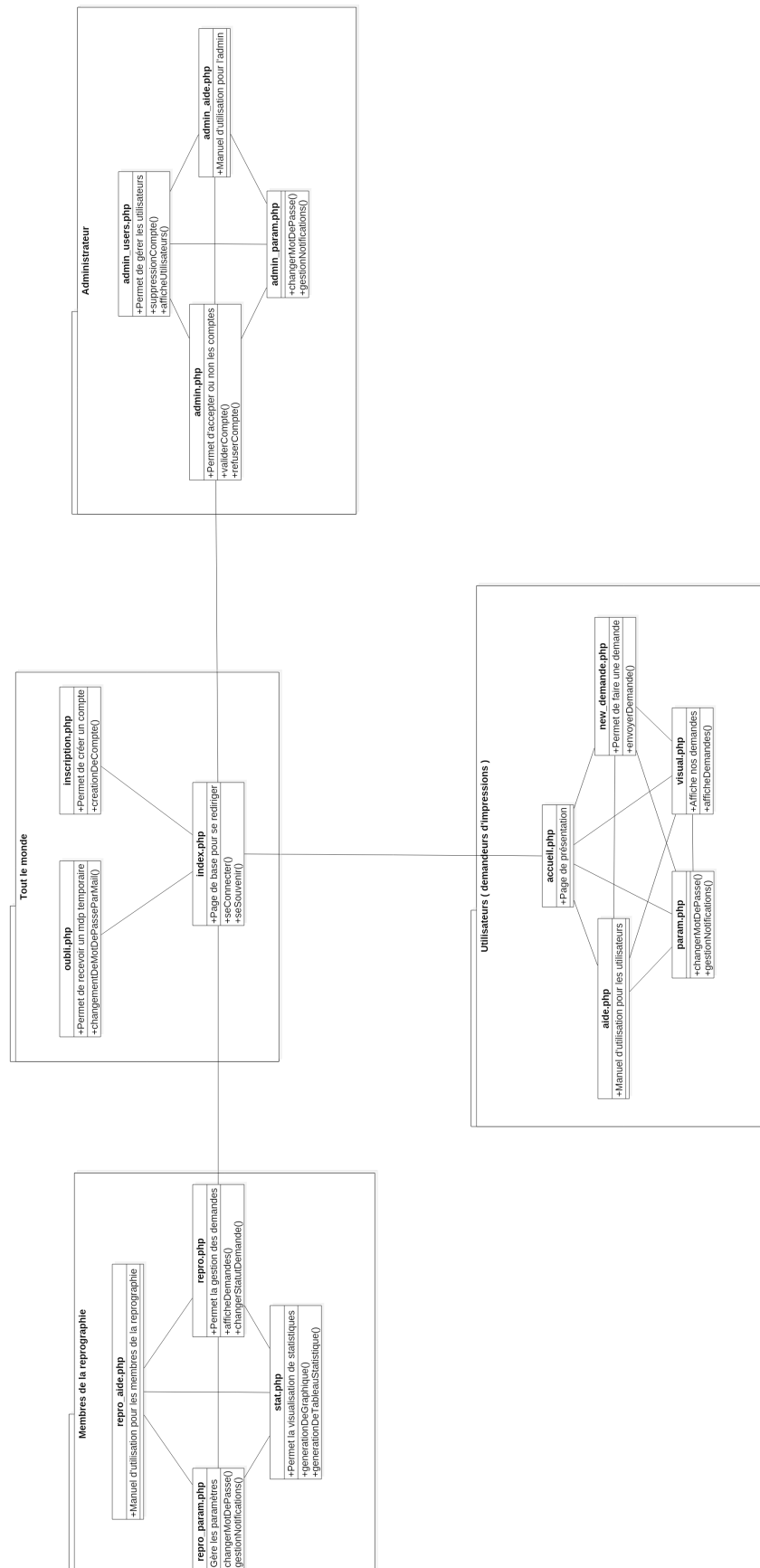
- **la page admin.php** : Elle contient un tableau qui représentera les demandes d'inscription en attente de validation par l'admin. Elle renvoie grâce à une barre de navigation aux pages *admin\_users.php*, *admin\_aide.php* et aussi vers *index.php* en cas de déconnexion et *admin\_param.php* afin qu'il puisse modifier ses paramètres
- **la page admin\_users.php** : Elle contient un tableau de la liste des utilisateurs de l'application. Elle renvoie grâce à une barre de navigation aux pages *admin.php*, *admin\_aide.php* et aussi vers *index.php* en cas de déconnexion et *admin\_param.php*
- **la page admin\_aide.php** : Elle contient les informations relatives à l'application, une sorte de page de documentation permettant à l'utilisateur de comprendre le fonctionnement de l'application. Elle renvoie grâce à une barre de navigation aux pages *admin\_users.php*, *admin.php* (page d'accueil), et aussi vers *index.php* en cas de déconnexion et *admin\_param.php*
- **la page admin\_param.php** : Elle contient un formulaire permettant à l'administrateur de modifier certains de ses paramètres comme son mot de passe et d'afficher les informations de l'administrateur.

**les pages accessibles par les membres du services de reprographie**

- **la page repro.php** : Elle contient un tableau récapitulatif des demandes entrantes au service reprographie ainsi que leur statut. Elle renvoie grâce à une barre de navigation aux pages *stat.php*, *repro\_aide.php* et *repro\_param.php* pour voir les paramètres du compte et les modifier. La déconnexion mène vers *index.php*
- **la page stat.php** : Elle contient des statistiques sur le taux d'impression faite entre des dates spécifiées par l'utilisateur. Elle renvoie grâce à une barre de navigation aux pages *repro\_aide.php* et aussi vers *repro.php* (page d'accueil) et *repro\_param.php* pour voir les paramètres du compte et les modifier.
- **la page repro\_aide.php** : Elle contient les informations relatives à l'application, une sorte de page de documentation permettant à l'utilisateur de comprendre le fonctionnement de l'application. Elle renvoie grâce à une barre de navigation aux pages *repro\_aide.php* (en terme de pages fonctionnelles), et aussi vers *repro.php* (page d'accueil) *repro\_param.php* pour voir les paramètres du compte et les modifier. La déconnexion mène vers *index.php*
- **la page repro\_param.php** : Elle contient un formulaire permettant à l'administrateur de modifier certains de ses paramètres comme son mot de passe et d'afficher les informations de l'administrateur. Le retour nous mène vers *repro.php* et la déconnexion vers *index.php*

Le diagramme ci-dessous représente toutes ces pages (définies ci-dessus) ainsi que leur interactions dans l'application, elles sont classées par paquet représentant les acteurs ayant accès à ces pages.

FIGURE 3.1 – Représentation organisationnelle de l'application Web.



### 3.1.1 Comportement

Le système étant un site web, les méthodes seront plutôt interprétée comme des fonctionnalités propres à chacune des pages composantes de celui-ci. Nous établirons une liste détaillée de chacun de ces fonctionnalités en les classant sous forme de tableau pour chacun des acteurs utilisant une des fonctionnalité décrite :

Reprographie	Fonctionnalité	Description	Page
1	changerMotDePasse()	Permet à l'acteur de changer son mot de passe dans la base de donnée	repro_param.php
2	gestionNotifications()	permet au membre du service de la reprographie de gérer les notifications reçues	repro_param.php
3	generationDeGraphique()	permet de récupérer les données de la base de donnée pour en faire des statistiques	stat.php
4	generationDeTableauStatistique()	permet de récupérer les données de la base de données pour générer un tableau statistique	stat.php
5	afficheDemandes()	permet de récupérer les données de la table ... qui concernent les demandes faites par un utilisateur	repro.php
6	changerStatutDemande()	permet de modifier le statut de la demande reçue dans la base de donnée	repro.php

**Tableau 1** : représentant les fonctionnalités des pages du service reprographie

Professeur	Fonctionnalité	Description	Page
1	envoyerDemande()	permet de créer une nouvelle demande dans la base de données	new_demande.php
2	afficheDemande()	permet de gérer l'affichage des demandes faites par l'utilisateur, celles qui sont présente dans la base de donnée et propres à lui-même	visual.php
3	changerMotDePasse()	permet de changer le mot de passe en base de donnée	param.php
4	gestionNotifications()	permet de gérer les notifications reçues et envoyées	param.php

**Tableau 2** : représentant les fonctionnalités des pages des clients (professeurs/chercheurs)



Administrateur	Fonctionnalité	Description	Page
1	validerCompte()	Permet à l'utilisateur d'ajouter un compte dans la table REAL_USERS de la base de donnée	admin.php
2	refuserCompte()	Permet à l'administrateur de refuser un compte et donc le supprimer de la table TMP_USERS sans l'ajouter dans la table REAL_USERS	admin.php
3	suppressionCompte()	permet à l'administrateur de supprimer un utilisateur de la table REAL_USER	admin_users.php
4	afficheUtilisateurs()	permet à l'administrateur d'afficher les différents utilisateurs présents dans la base de données	admin_users.php

**Tableau 3** : représentant les fonctionnalités des pages de l'administrateur

Tous	Fonctionnalité	Description	Page
1	changementDeMotDePasseParMail()	permet à tout utilisateur du site de changer de mot de passe avant connexion en cas d'oubli en recevant un mot de passe généré automatiquement en base de données envoyé au mail de l'utilisateur.	oubli.php
2	creationDeCompte()	permet à tous de créer un compte dans la table TMP_USERS en envoyant les informations du formulaire rempli	inscription.php
3	seConnecter()	permet à tous de se connecter en entrant les identifiants et mots de passes présents en base de données	index.php
4	seSouvenir()	permet à tous de faire en sorte que le site web mémorise les identifiants de connexion pour l'adresse IP liée à la connexion durant laquelle cette fonction a été utilisée	index.php

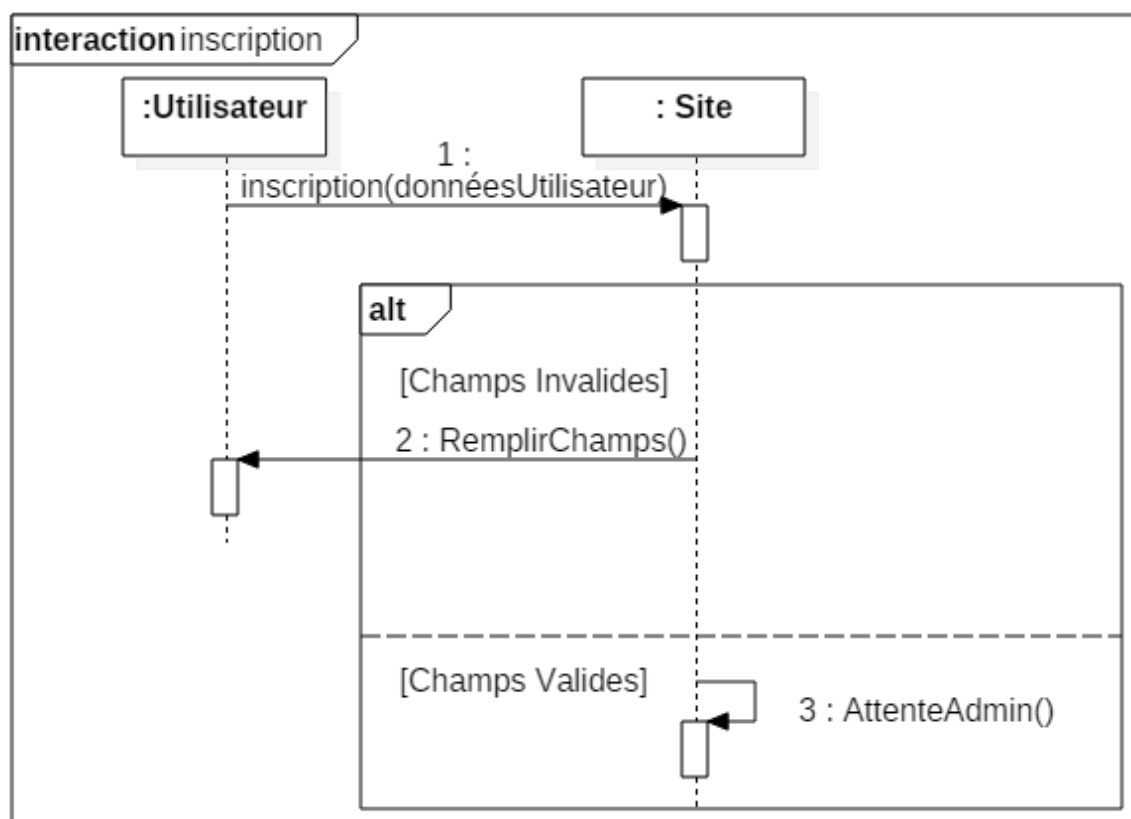
**Tableau 4** : représentant les fonctionnalités des pages communes à chaque acteurs

### Fonctionnalité d'inscription

La page inscription.php est une page composée de plusieurs champs permettant de créer un utilisateur dans la base de donnée. Lorsque l'utilisateur entre ses informations dans le formulaire d'inscription, celles-ci sont renvoyées vers le site :

- Si les champs entrés ne sont pas valides, alors le site web renverra dans l'affichage une erreur.
- Sinon s'ils sont valides, alors ils seront insérés dans une table temporaire de la base de données en attendant la validation de l'administrateur

FIGURE 3.2 – Diagramme de séquence représentant la méthode Inscription.



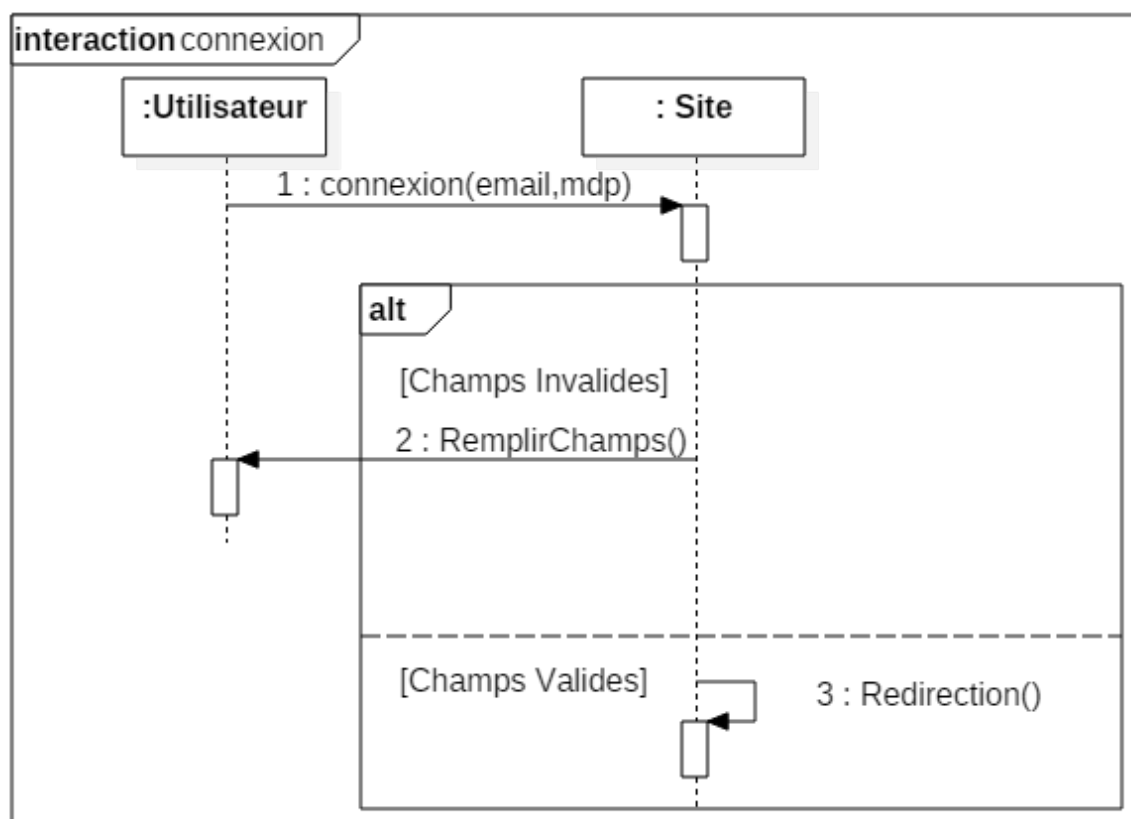
Dans ce diagramme on peut voir qu'une instance anonyme d'utilisateur envoie un message à une instance s1 qui est le système. Le système va renvoyer un message d'erreur si le champs est incorrect et si le champs est correct il va s'envoyer un message à lui même pour ajouter l'utilisateur a la base de données.

### Fonctionnalité connexion

La page connexion est constituée de 2 champs, un champs identifiant et un champs mot de passe ainsi que d'une méthode connexion qui va chercher dans la base de données si l'identifiant existe et si le mot de passe entré correspond a celui qui est enregistré.

- si les deux champs correspondent à ceux enregistrés en base de données : l'utilisateur est redirigé vers la page index.php
- sinon il le site web affichera son erreur et lui proposera de ressaisir ses informations.

FIGURE 3.3 – Diagramme de séquence représentant la méthode Connexion.

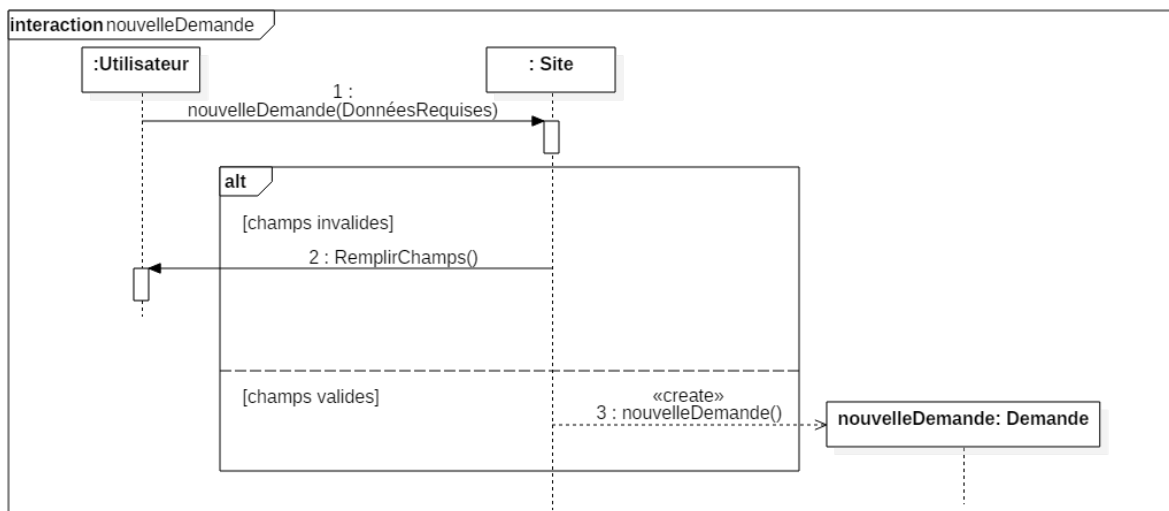


### Fonctionnalité NouvelleDemande

La fonctionnalité permettant de créer une nouvelle demande consiste en un remplissage du formulaire de demandes avec les données requise qui seront transférée au serveur.

- Si les champs ne sont pas correctement remplis, alors l'utilisateur devra les remplir, autrement on crée une nouvelle instance de demande (en soit une nouvelle demande)

FIGURE 3.4 – Diagramme de séquence représentant la méthode NouvelleDemande.

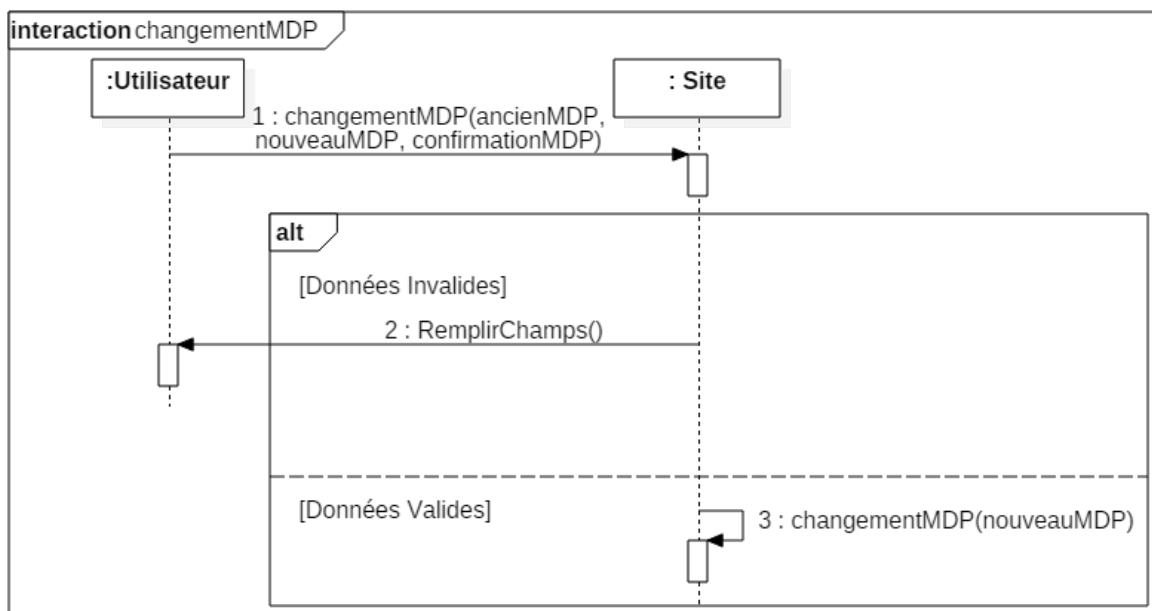


### Fonctionnalité changerMotDePasse

Cette fonctionnalité permet de modifier le mot de passe lorsque que l'utilisateur après connexion via ses paramètres, elle est utilisable par chacun des acteurs clés composants du site web.

- L'utilisateur entre 3 paramètres : son ancien mot de passe, le nouveau puis une confirmation.
- si l'utilisateur remplit mal les champs et insère des données invalides (mot de passe ne respectant pas les règles de caractères) alors le site lui propose de réitérer sa demande de changement
- sinon, les données seront insérées dans la base de données et le mot de passe sera donc changé.

FIGURE 3.5 – Diagramme de séquence représentant la méthode ChangementMDP.



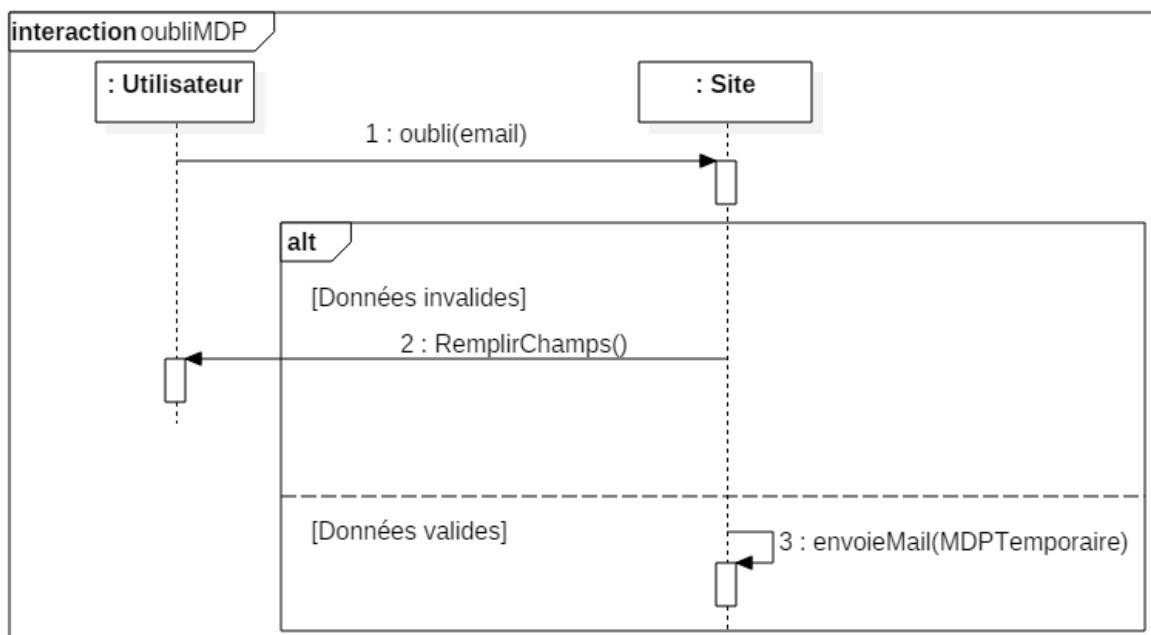
### Fonctionnalité OubliMotDePasse

Cette méthode se déclenche quand un utilisateur interagit avec la page oubli de mot de passe sur l'index de connexion.

Lorsque que l'utilisateur arrive sur cette page et entre son mail :

- soit il n'est pas reconnu par la base de données et doit donc remplir de nouveau les champs avec un email juste
- sinon la fonctionnalité permettra au site web d'envoyer en mail un mot de passe temporaire généré en base de données lui permettant de se connecter

FIGURE 3.6 – Diagramme de séquence représentant la méthode OublieMDP.

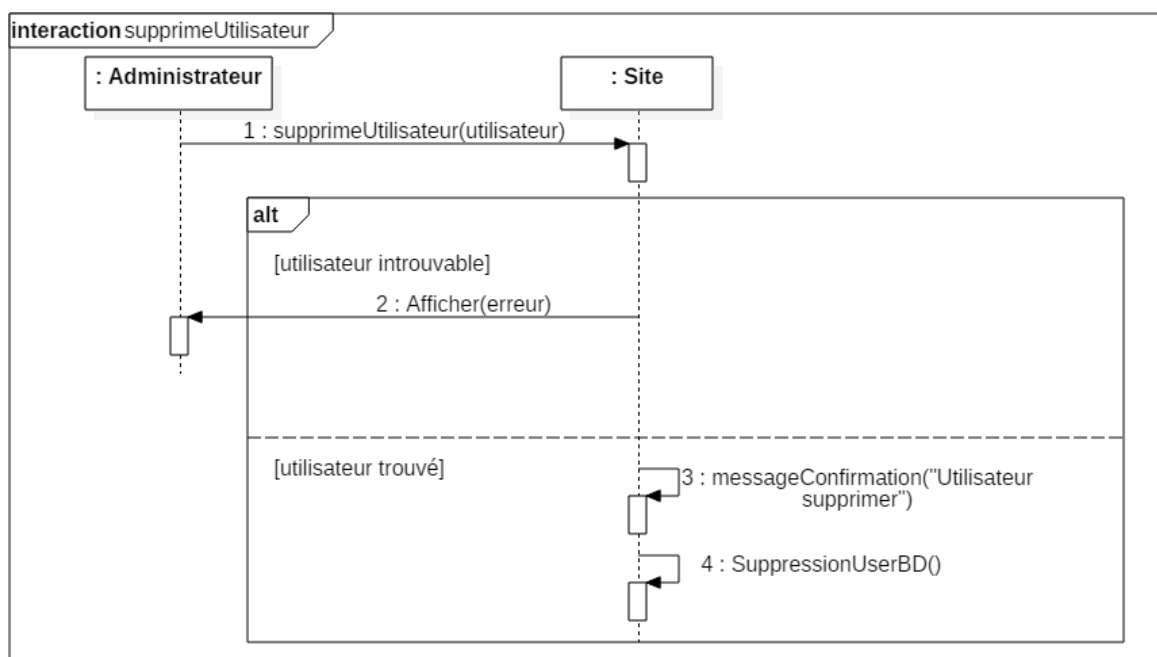


### Fonctionnalité de suppression d'utilisateur

La fonctionnalité SupprimeUtilisateur consiste en la suppression d'un utilisateur par l'administrateur :

- si l'utilisateur n'existe pas dans la base de données, alors le site web renvoi une erreur.
- sinon s'il existe bien, alors il envoie une demande de confirmation de suppression que doit valider l'administrateur, auquel cas l'utilisateur sera supprimé de la base de données.

FIGURE 3.7 – Diagramme de séquence représentant la méthode SupprimeUtilisateur.



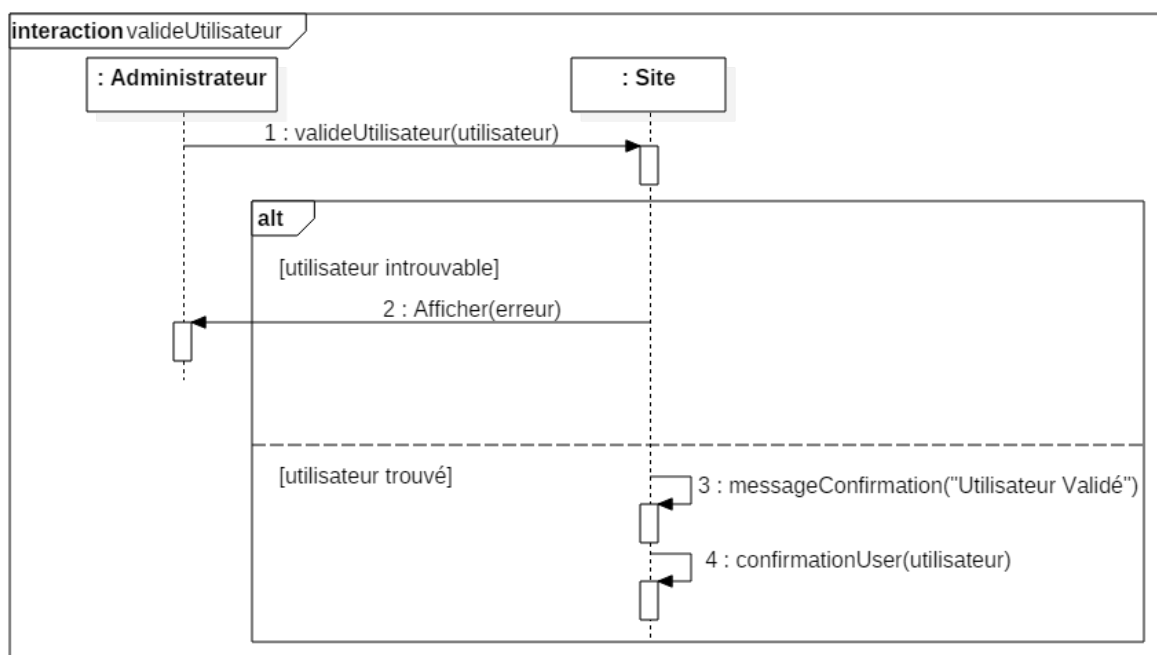


### Fonctionnalité permettant de ValideUtilisateur

Cette fonctionnalité permet à l'administrateur de valider un utilisateur désirant s'inscrire.

- si l'utilisateur n'existe pas dans la base de données, alors le site web renvoi une erreur.
- sinon s'il existe bien, alors il envoie une demande de confirmation de suppression que doit valider l'administrateur, auquel cas l'utilisateur sera supprimé de la base de données.

FIGURE 3.8 – Diagramme de séquence représentant la méthode ValideUtilisateur.



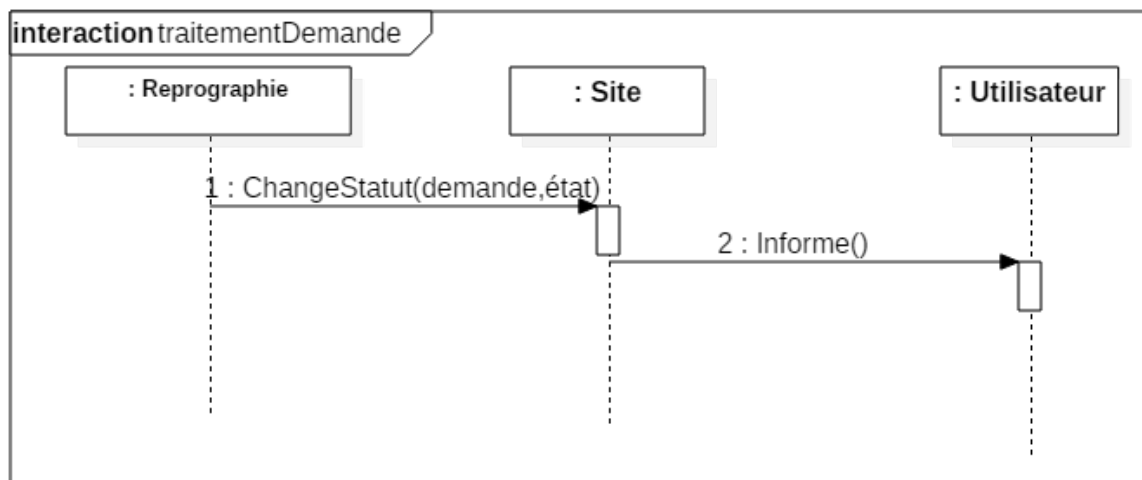
### Fonctionnalité de traitement de commande

#### — Traitedemande

La fonctionnalité de traitement de demande permet à un membre de la reprographie de changer le statut de celle-ci.

Une fois que le membre de la reprographie change le statut de la demande, l'utilisateur reçoit une notification.

FIGURE 3.9 – Diagramme de séquence représentant la méthode TraiteDemande.



### 3.1.2 Base de données

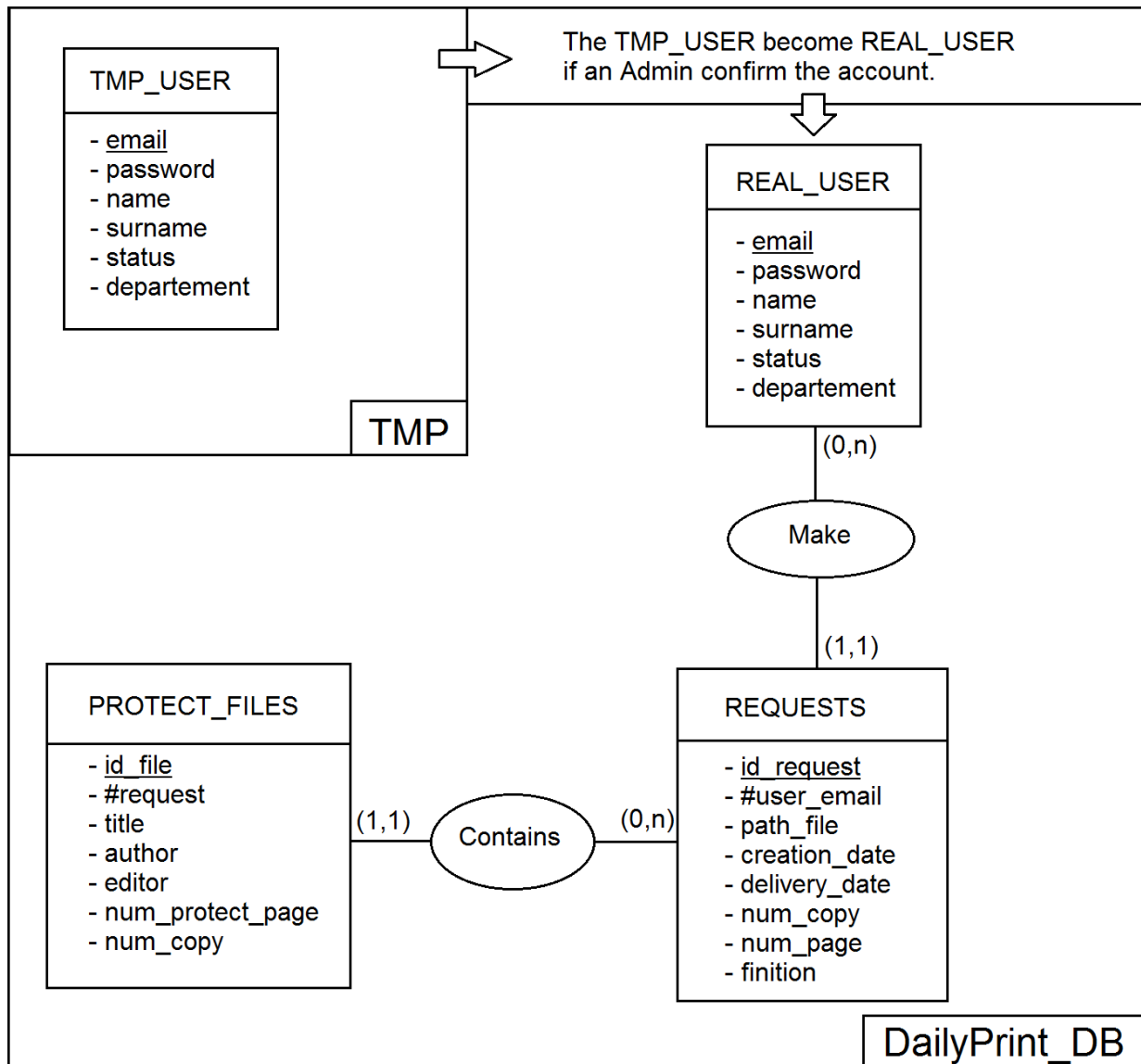
Voici le modèle de la base de données, nous avons ici 4 tables qui sont :

- TMP\_USER(email, name, surname, password, departement, status) : ce sont tous nouveaux utilisateurs.
- REAL\_USER(email, name, surname, password, departement, status) : ce sont les utilisateurs qui ont été validés et peuvent effectuer des demandes.
- RESQUESTS(id\_resquest, creation\_date, delivery\_date, user\_email, num\_copy, num\_page, finition, path\_file) : les commandes effectuées par les différents utilisateurs.
- PROTECT\_FILES(id\_file, title, author, editor, num\_protect\_page, num\_copy, requests) : ce sont les informations sur les fichiers qui vont passer en reprographie.

Les utilisateurs temporaires n'ont pas accès à l'application mais ils deviennent réels quand ils sont validés par un administrateur. Un utilisateur validé peut faire des demandes, ses fichiers sont protégés.

Voici le schéma associé :

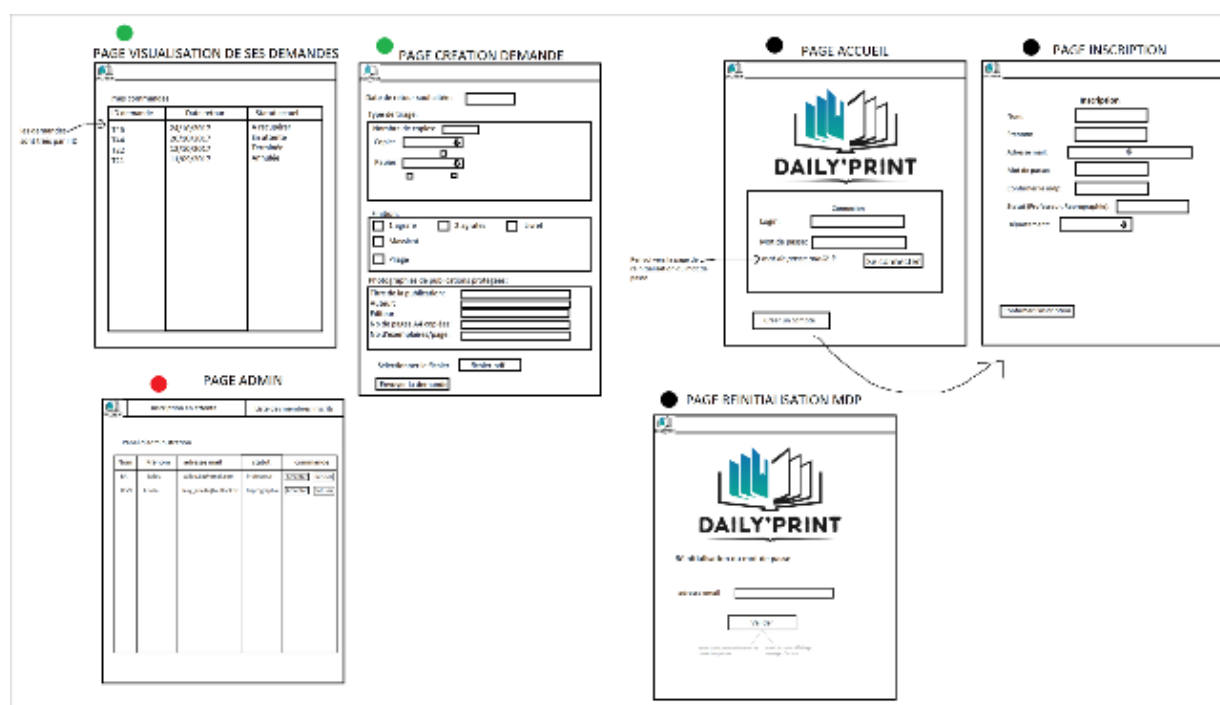
FIGURE 3.10 – Schéma de la base de donnée.



## 3.2 Spécifications détaillées des interfaces utilisateurs

### 3.2.1 Prototype de l'interface

FIGURE 3.11 – Prototype d'interface.



### 3.2.2 Définition de la charte graphique

Logo : 

Typographie :

- Normal
- Calibri
- Italique
- gras
- A-B-C-D-E-F-G-H-I-J-K-L-M-N-O-P-Q-R-S-T-U-V-W-X-Y-Z
- a-b-c-d-e-f-g-h-i-j-k-l-m-n-o-p-q-r-s-t-u-v-w-x-y-z
- 0-1-2-3-4-5-6-7-8-9 et les caractère spéciaux

Couleurs :

- Blanc(code couleur : RGB :255,255,255)
- Bleu(code couleur : RGB :51,122,183)

- Bleu vers Gris (Dégradée)
- Bleu vers Vert (Dégradée)
- Jaune vers Rouge (Dégradée)
- Vert vers Vert foncé (Dégradée)
- Rouge vers Rouge foncé (Dégradée)
- Jaune(code couleur : RGB : 255,255,0)
- Vert(code couleur : RGB : 0,238,0)
- Gris/Blanc(code couleur : RGB : 207,207,207)
- Noir(code couleur : RGB : 0,0,0)

**Fournisseur d'accès :** Hostinger

**Nom du site et lien :** <http://www.dailyprint.xyz/>

**Image issue d'internet :** pas d'image issue d'internet

# Chapitre 4

## Dossier architecture

Ce dossier a pour objectif de modéliser la structure globale du système en utilisant plusieurs vues ainsi qu'en décomposant le système en sous-système. Il se décompose en plusieurs modèles d'architecture. Ces modèles représentent les vues du site web donc les différents composants de celui-ci ainsi que leur relation.

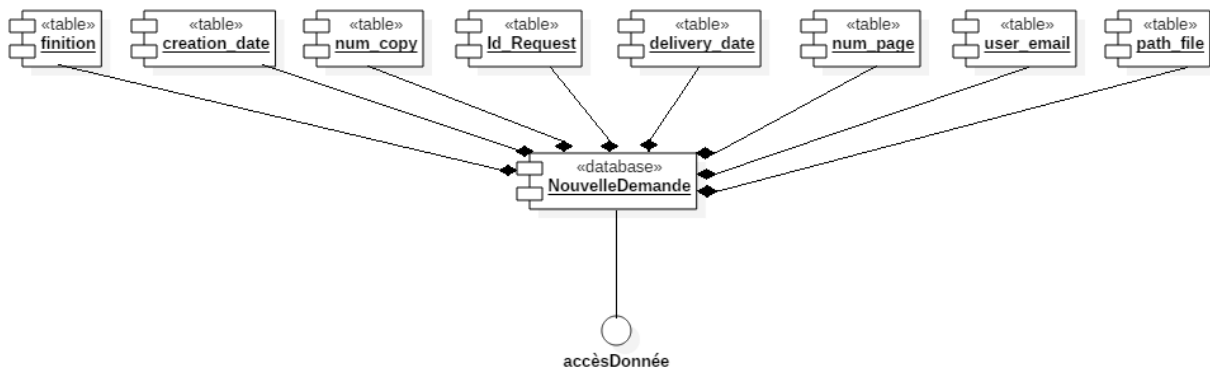
### 4.1 Vue modulaire

Ce chapitre décrit l'ensemble d'éléments qui composent le système d'un point de vue statique. Il montre également la description des interconnexions entre les composants ainsi que les interfaces offertes par eux. La structure modulaire peut être spécifiée en UML par le biais de diagrammes de paquetages, diagrammes de sous-systèmes et diagrammes de classes. Ce chapitre doit aussi préciser le choix effectué par rapport au style d'architecture (en couches..)

### 4.2 Vue composant/Connecteur

D'un point de vue dynamique, le système utilise une base de données pour recenser toutes les données essentielles à l'application WEB. L'application WEB accède à cette base pour mettre en place le processus de gestion de la reprographie. Elle est constituée des pages et méthodes indiquées dans le schéma de représentation organisationnel, et forme notre exécutable.

FIGURE 4.1 – Diagramme de package représentant la vue de composant/connecteur



### 4.3 Vue d'attribution

Cette partie contient la description des associations entre les composants logiques et les composants physiques de notre site web. Les structures d'attributions vont permettre de spécifier les composants dans l'infrastructure matérielle. Les associations entre les composants et les nœuds d'exécution sont spécifiées. Chaque nœud et leurs interconnexions sont décrits en UML et leur modélisation peut-être faite via un diagramme de déploiement : Les associations entre les composants et les éléments logiciel physiques (fichiers, tables, bibliothèques, etc..) La spécification en UML peut-être représentée à l'aide d'un diagramme de composant :

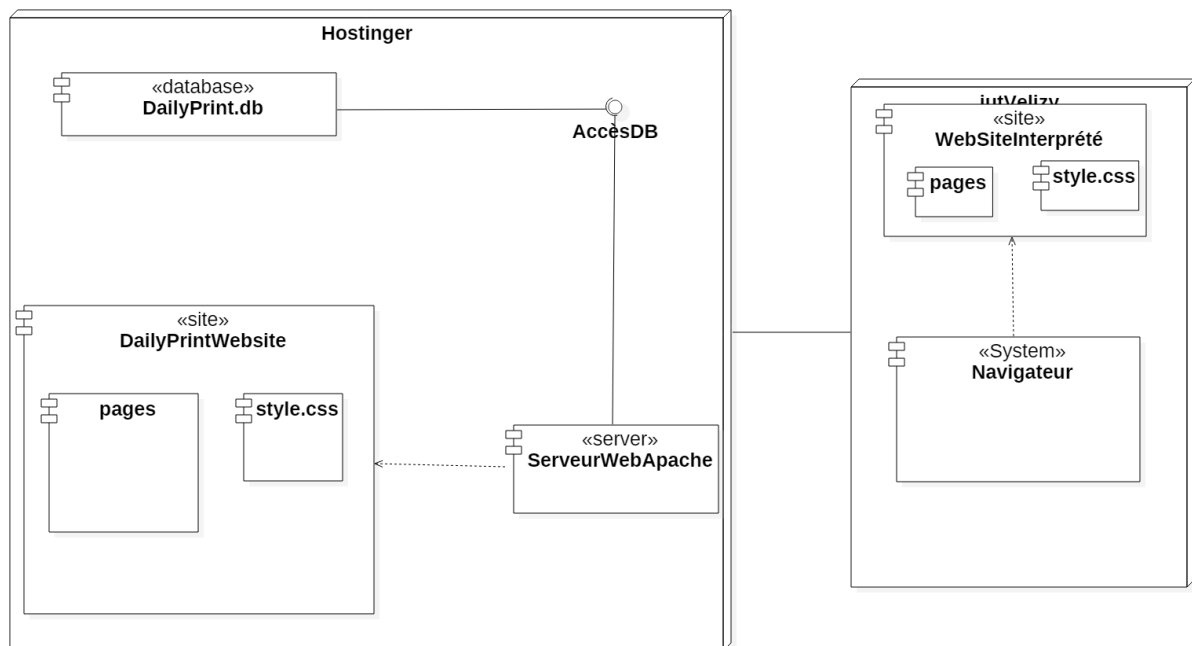
Dans notre cas, les éléments seront réarrangés en fonctions de leur nœuds d'appartenance :

- La base de donnée sera hébergée chez Hostinger qui nous fourni un serveur apache
- Le site web non interprété quand-à-lui appartient au nœud des service DailyPrint.
- A contrario le site web interprété appartient au nœud des services de l'IUT de vélizy car c'est en son sein que sera exécuté le site web.

D'où le diagramme de déploiement suivant :



FIGURE 4.2 – Diagramme de package représentant la vue d'attribution



# Conclusion

Notre projet comporte une marge de progression et d'évolution très grande à l'heure de la rédaction de ce rapport notamment en terme de fonctionnalité et d'ajout de caractère dynamique au site web. Au cours de ce projet nous avons pu mettre en pratique nos connaissances mais nous avons aussi pu en acquérir de nouvelles notamment des compétences en terme de management et de travail en coordination.