# AVERTISSEMENT

LA HAUTE ÉCOLE DE GESTION ET DE JOURNALISME N’ENTEND DONNER NI APPROBATION, NI DÉSAPPROBATION AUX OPINIONS ET IDÉES ÉMISES DANS CE MÉMOIRE. ELLES DOIVENT ÊTRE CONSIDERÉES COMME PROPRES À LEUR AUTEUR.

**DÉDICACE ET REMERCIEMENTS**

# DÉDICACE

Je dédie ce travail à :

* Mon père **FADONOUGBO Aimé**
* Ma mère **HOUSSOU Justine**

# REMERCIEMENTS

Je tiens à exprimer ma profonde gratitude et remerciements envers toutes les personnes qui m’ont accompagné et soutenu durant la réalisation de ce mémoire et en particulier :

* Au Directeur General de la **Haute Ecole de Gestion et de Journalisme (HEGJ), M Césaire AGOSSA.**
* À notre Directeur des Etudes, **M Nicolas GBOSSOU.**
* À l’assistant du Directeur des Etudes, **M Abraham COUEGNONHOU.**
* À tous les enseignants de la **Haute Ecole de Gestion et de Journalisme. (HEGJ) pour** les trois années d’éducation qu’ils nous ont offertes durant notre formation.
* À mon maître de mémoire, **M AHISSOU Y. Gildas Aimé**, pour avoir accepté de superviser ce travail avec rigueur. Ses suggestions, remarques et encouragements m’ont été d’une grande aide.
* À mon maitre de stage, **M Roland TOHOUMON** pour sa disponibilité, son soutien et ses encouragements.
* À tout le personnel de la **Radio Trait D’Union** pour leur disponibilité
* À mes frères et sœurs pour leur soutien et leur patience à mon égard.

# LISTE DES SIGLES ET ABRÉVIATIONS

|  |  |
| --- | --- |
| **Sigles et Abréviations** | **Désignation** |
| UML | Unified Modeling Language |
| JS | JavaScript |
| TS | TypeScript |
| CSS | Cascading Style Sheet |
| HTML | Hyper Text Makup Language |
| RTU | Radio Trait d’Union |
| SIL | Système Informatique et Logiciel |
| PHP | Hypertext Preprocessor |

Tableau 1 Liste des sigles et abréviations

# LISTE DES FIGURES

[Figure 1 Organigramme de la RTU 8](file:///C:\Users\ADMIN\Documents\gautier_utils_memoire\gautier_new_memoire%20-%20Copie.docx#_Toc168503765)

[Figure 2 Liste des diagrammes UML 15](#_Toc168503766)

[Figure 3 Diagramme de cas d'utilisation 17](#_Toc168503767)

[Figure 4 Diagramme d'activité du cas <<s'inscrire>> 21](#_Toc168503768)

[Figure 5 Diagramme d’activité du cas <<Souscrire à un service>> 22](#_Toc168503769)

[Figure 6 Diagramme de séquence du cas <<S'authentifier>> 23](#_Toc168503770)

[Figure 7 Diagramme de séquence du cas <<modifier une demande>> 24](#_Toc168503771)

[Figure 8 Diagramme de classe 25](#_Toc168503772)

[Figure 9 Page d'accueil 31](#_Toc168503773)

[Figure 10 Interface de connexion 32](#_Toc168503774)

[Figure 11 Section podcasts 32](#_Toc168503775)

[Figure 12 Interface utilisée pour faire une demande de spot publicitaire 33](file:///C:\Users\ADMIN\Documents\gautier_utils_memoire\gautier_new_memoire%20-%20Copie.docx#_Toc168503776)

[Figure 13 Interface de paiement 34](#_Toc168503777)

[Figure 14 Dashboard utilisateur 34](#_Toc168503778)

[Figure 15 Dashboard administrateur 35](#_Toc168503779)

[Figure 16 Vue de la plateforme sur mobile 35](file:///C:\Users\ADMIN\Documents\gautier_utils_memoire\gautier_new_memoire%20-%20Copie.docx#_Toc168503780)

[Figure 17 Code utilisé pour la migration de la table users 36](#_Toc168503781)

[Figure 18 Code utilisé pour enregistrer une demande publicitaire 37](file:///C:\Users\ADMIN\Documents\gautier_utils_memoire\gautier_new_memoire%20-%20Copie.docx#_Toc168503782)

[Figure 19 Route dashboard 38](#_Toc168503783)

# LISTE DES TABLEAUX

[Tableau 1 Liste des sigles et abréviations v](#_Toc168503784)

[Tableau 2 liste des acteurs et des cas d'utilisation 16](#_Toc168503785)

SOMMAIRE

[INTRODUCTION 1](#_Toc168477897)

[CHAPITRE 1 : CADRE CONDITIONNEL DU STAGE 2](file:///C:\Users\ADMIN\Documents\gautier_utils_memoire\gautier_new_memoire%20-%20Copie.docx#_Toc168477898)

[I. Historique de la Radio Trait d’Union 3](#_Toc168477899)

[II. Organisation fonctionnelle 3](#_Toc168477900)

[CHAPITRE 2 : PRESENTATION DE LA PLATEFORME Á CONCEVOIR 9](file:///C:\Users\ADMIN\Documents\gautier_utils_memoire\gautier_new_memoire%20-%20Copie.docx#_Toc168477906)

[I. Problématique 10](#_Toc168477907)

[II. Objectifs généraux 10](#_Toc168477908)

[III. Objectifs spécifiques 11](#_Toc168477909)

[IV. Raison du choix du thème 11](#_Toc168477910)

[CHAPITRE 3: METHODOLOGIE 12](file:///C:\Users\ADMIN\Documents\gautier_utils_memoire\gautier_new_memoire%20-%20Copie.docx#_Toc168477911)

[I. Présentation de la méthode utilisée 14](#_Toc168477912)

[II. Présentation du diagramme de comportement 15](#_Toc168477913)

[III. Présentation du diagramme d’activité 19](#_Toc168477917)

[IV. Présentation du diagramme de séquence 21](#_Toc168477918)

[V. Présentation du diagramme de classe 22](#_Toc168477919)

[Chapitre 4 : Conception du système 24](file:///C:\Users\ADMIN\Documents\gautier_utils_memoire\gautier_new_memoire%20-%20Copie.docx#_Toc168477922)

[I. Présentation des outils de développement 25](#_Toc168477923)

[II. Présentation de l’application 29](#_Toc168477930)

[CONCLUSION 36](file:///C:\Users\ADMIN\Documents\gautier_utils_memoire\gautier_new_memoire%20-%20Copie.docx#_Toc168477933)

**RÉSUMÉ**

Notre projet de fin de formation en SIL porte sur le thème << CONCEPTION ET RÉALISATION D’UNE PLATEFORME POUR LA GESTION DES SERVICES DE LA **RADIO TRAIT D’UNION** >>. Son objectif est de simplifier la gestion des divers services offerts par cette radio, notamment les demandes de spots publicitaires, les avis de recherche, les annonces, etc., afin d'atteindre un public plus large. L’analyse de ce projet a été faite avec UML et sa réalisation a été possible grâce à des langages de programmation et frameworks. Nous avons notamment utilisé PHP et TYPESCRIPT comme langages de programmation, LARAVEL, REACT JS et TAILWIND CSS comme frameworks, Visual Studio Code en tant qu'éditeur de code extensible, et PostgreSQL en tant que SGBD.

**ABSTRACT**

Our final project in SIL focuses on the theme <<DESIGN AND IMPLEMENTATION OF A PLATFORM FOR MANAGING THE SERVICES OF **TRAIT D’UNION RADIO**>>. Its objective is to simplify the management of the various services offered by this radio station, including advertising requests, search notices, announcements, etc., in order to reach a wider audience. The analysis of this project was done using UML, and its implementation was made possible through programming languages and frameworks. We used PHP and TYPESCRIPT as programming languages, LARAVEL, REACT JS, and TAILWIND CSS as frameworks, Visual Studio Code as an extensible code editor, and PostgreSQL as the DBMS.

# INTRODUCTION

À l'ère numérique actuelle, les médias traditionnels doivent impérativement adapter leur présence en ligne pour rester pertinents et compétitifs. Le monde de la radiodiffusion n’échappe pas à cette nécessité. En effet, ce secteur doit adopter les technologies de l’information et de la communication (TIC) pour assurer la visibilité et la pertinence de ses divers services. Dans le but de faciliter et d’améliorer la gestion de ces services, notre stage à la **Radio Trait d’Union** nous a permis de développer une solution informatique consistant en la création d’une plateforme de gestion pour les différents services, offerts par cette radio.

Ce mémoire se concentre sur l'analyse approfondie du processus de conception et de réalisation d'une telle plateforme pour la **Radio Trait d'Union**. Il vise à :

- Identifier les défis techniques et les contraintes spécifiques liées à la conception d'une plateforme pour une station de radio.

- Explorer les différentes étapes du développement de la plateforme, de la conception initiale à la mise en ligne.

- Évaluer l'impact de la plateforme sur l'interaction avec le public et la promotion de la radio sur le web.

# CHAPITRE 1 : CADRE CONDITIONNEL DU STAGE

## Historique de la Radio Trait d’Union

**Radio Trait d’Union (RTU)** est la deuxième chaîne radiophonique installée dans la commune de Bohicon, ville située à 120 km de Cotonou. Son promoteur a participé avec succès à un appel à candidature lancé par la Haute Autorité de l’Audiovisuel et de la Communication (HAAC) pour l’installation et l’exploitation des fréquences de radiodiffusion sonore privée .Selon le promoteur Césaire AGOSSA ,la **Radio Trait d’Union** vit le jour et émit pour la première fois le 24 mars 2003 sur la 95.3 en bande FM .Elle est une chaîne locale commerciale créée avec un capital social de dix millions de franc CFA .Ceci conformément à la loi N° 97-010 du 20 aout 1997 dont l’article 39 stipule : «  pour être autorisée, une radiodiffusion sonore privée commerciale doit avoir un capital d’au moins dix millions (10 000 000 ) de francs CFA » . Radio Trait d’Union couvre les départements du Zou et des Collines, du Mono et du Couffo et une partie de l’Atlantique et de l’Ouémé. Elle est actuellement située au quartier Agonvèzoun et a pour mission principale d’informer, de distraire et d’éduquer.

## Organisation fonctionnelle

### Fonctionnement de la radio trait d’union

**Radio Trait d’Union** dispose d’une direction générale assurée par le promoteur. Celle-ci s’occupe de la gestion courante et du bon fonctionnement de l’entité médiatique. Elle est assistée dans sa mission par une secrétaire de direction et un comptable.

La direction générale dispose aussi d’une équipe de journalistes et de techniciens rompus à la tâche, ainsi que d’un service commercial.

**Les instances de décision**

Elles sont au nombre de deux : le Conseil d’Administration (CA) et le Comité de Direction (CODIR). Le conseil d’Administration (CA) « donne les grandes orientations, assure le contrôle de la gestion de direction et veille au respect du manuel de procédure ». Le Comité de Direction (CODIR) est présidé par la directrice de la Radio, Madame Natalie ALOTCHEKPA. Il se réunit en séance hebdomadaire et en séance extraordinaire en cas de besoin. « Les décisions du CODIR prennent effet dès leur validation par la Directrice de la Radio ».

### Mission de la radio trait d’union

À l’instar des autres chaines de radiodiffusion sonore qui diffusent des programmes généralistes, **RTU** a pour mission d’informer, d’éduquer et distraire. A travers les émissions qu’elle diffuse, elle accompagne le développement local, contribue au changement de comportement des habitudes citoyennes ; participe à la lutte contre le chômage des jeunes ; fait du droit à l’information une réalité, et contribue à l’accès à l’information des citoyens.

### Organisation de la radio trait d’union

La Radio Trait d’Union est structurée comme suit : Le service commercial, Le service des programmes et de production, Les services de l’information encore appelés la rédaction, Le Service Technique et de l’Animation

**● Le service commercial**

La **Radio Trait d’Union** se préoccupe, en tant que commercial, des clients et de ses partenaires qu’ils soient des personnes physiques ou morales, publiques ou privées. Iloffre aux clients des prestations de marketing, des couvertures médiatiques à caractère commercial et toute action de communication visant à stimuler leurs diverses activités.

**●Le service des programmes et de production**

Le service des programmes et de production coordonne les différentes étapes de la réalisation des émissions au sein de la radio. Pour les émissions enregistrées les fonctions sont celles de production (présentation, animation, reportage, montage, etc). Ce service est assuré par Monsieur Roland AZATASSOU. Cette mission s’effectue quotidiennement par le biais d’une grille des programmes.

**● Les services de l’information encore appelés la rédaction**

C’est dans ce service que se préparent les informations en français et en langue nationale (fon-gbe) diffusées sur la chaîne. Chaque matin à 08h30 a lieu une grande réunion des journalistes au cours de laquelle on décide du menu des différentes éditions de journal de la journée. Cette réunion est appelée conférence de rédaction et est dirigée par Monsieur Arnaud Xavier AVODAGBE. C’est lors de la conférence de rédaction que les différentes tâches professionnelles tels les reportages à effectuer, les interviews à réaliser, la rédaction des articles etc. sont assignées. C’est sur Internet qu’ils recherchent les nouvelles internationales et utilisent les sites des agences de presse et des médias internationaux tels que Radio France Internationale (RFI), France 24, TV5 Monde ou la BBC. Des magazines alimentant les éditions du journal sont souvent proposés par les stagiaires sur instructions des journalistes plus expérimentés. Les sujets traitent généralement des faits de société et parfois de la santé. Tous les articles du journal sont soumis à la directrice ou à un autre journaliste supérieur avant la diffusion. C’est ce qui fait la force et la particularité de ce service. Le processus menant à la présentation du journal est toujours un moment de stress mais aussi d'apprentissage.

**●Le Service Technique et de l’Animation**

Le service technique s'occupe du bon fonctionnement et de la qualité technique des émissions transmises à l'antenne ainsi que de la visibilité de la radio en ligne. Il veille à ce que les auditeurs bénéficient d'une bonne qualité d'écoute des spots et des annonces publicitaires, des émissions, des éditions du journal parlé et des différentes animations. Ce service est assuré par Monsieur Oscar SAGBADJA, chef technique du matériel, et Monsieur Roland TOHOUMON, chef technique des TIC.

### Observation relative au cadre de recherche

À ce niveau nous présenterons les forces et les faiblesses de radio Trait D’union

A-Forces

La Radio Trait D’union est dotée de plusieurs qualités qui constituent ses forces. Ces qualités les suivantes :

* Des ordinateurs de qualité pour la rapidité du travail
* Des techniciens professionnels pour la diffusion des émissions
* Des appareils et logiciels de diffusion de qualité
* Diffusion des émissions en direct sur les réseaux sociaux
* Diffusion de la radio en continue sur Zeno radio et africa web radio
* Possibilité de réécouter les émissions et éditions du journal parlé sur les réseaux sociaux
* Disponibilité de wifi
* Tous les ordinateurs en réseaux sans fil

Mais comme toute structure Radio Trait D’union a aussi quelques faiblesses.

B-Faiblesses

Les faiblesses de la Radio Trait D’union s’énumèrent comme suit :

- Insuffisance de personnel (journalistes)

### Organigramme

Figure 1 Organigramme de la RTU

# CHAPITRE 2 : PRESENTATION DE LA PLATEFORME Á CONCEVOIR

## Problématique

Dans le contexte dynamique du paysage médiatique contemporain, où la concurrence est féroce et les attentes du public en perpétuelle évolution, la **Radio Trait d’Union** se voit contrainte d'embrasser les technologies et les outils informatiques pour mieux répondre aux besoins de sa clientèle et garantir sa fidélité. Cette démarche permettrait également à un plus large public de découvrir les services proposés par la Radio et de s'y abonner en cas d'intérêt. Ainsi, pour répondre à cette nécessité, la **Radio Trait d’Union** doit mettre en place un outil de vulgarisation de ses services ainsi qu'un guichet électronique pour faciliter l'adhésion des clients potentiels. Dans cette perspective, la création d'une plateforme dédiée à la gestion de ses services devient indispensable pour assurer sa pérennité dans un environnement médiatique constamment en mutation.

## Objectifs généraux

Le développement et la mise en place d’une plateforme vise non seulement à faire connaitre les services de la radio et á écouter la radio en ligne mais aussi, á faciliter l’accès á ces services.

Pour parvenir á ce niveau, nous allons décliner l’objectif général en des objectifs spécifiques.

## Objectifs spécifiques

* Mettre en ligne des revues de presse
* Écouter en replay les émissions de journal et des débats
* Écouter la radio en ligne
* Souscrire á des services directement en ligne
* Modifier les informations concernant le service auquel on n’a souscrit
* Créer des comptes utilisateur
* Avoir la liste des souscriptions sous forme de PDF
* Visualiser la grille tarifaire
* Visualiser la grille des programmes

## Raison du choix du thème

Nous trouvons une motivation profonde dans ce sujet, car nous sommes fermement convaincus que la transition vers une plateforme web pour la gestion des services de la Radio Trait d'Union peut offrir d'énormes avantages en termes d'accessibilité, de flexibilité et d'efficacité. En concevant une plateforme web personnalisée, nous pourrions fournir aux équipes de la radio un outil centralisé et convivial pour gérer divers aspects de leurs opérations, qu'il s'agisse de la gestion des publicités ou de la coordination des événements. De surcroît, une plateforme web permettrait une accessibilité accrue, donnant ainsi aux membres de l'équipe la possibilité d'accéder aux informations et aux fonctionnalités essentielles depuis n'importe où et à tout moment. En tant que développeurs, nous sommes enthousiasmés par le défi technique de concevoir une interface intuitive et réactive, tout en garantissant la sécurité et la fiabilitéde la plateforme pour répondre aux exigences évolutives de l'industrie de la radio.

# CHAPITRE 3 : METHODOLOGIE

Dans l'élaboration d'un système d'information, la modélisation des données englobe l'analyse et la conception de l'information intégrée dans le système. Son objectif est de représenter de manière précise la structure de ces données et de mettre en place une organisation efficace pour le stockage ainsi que les traitements informatiques. Diverses méthodes de modélisation sont à notre disposition, parmi lesquelles :



**UML (UNIFIED MODELING LANGUAGE)**

C’est clairement la norme de référence, la technique la plus connue et enseignée à ce jour. Il est vrai qu’elle permet de visualiser et de conceptualiser à peu près toutes les facettes d’une demande de changement, et qu’elle est compréhensible autant par les interlocuteurs métier que techniques. En tant qu'analyste d'affaires, cette boîte à outils permet de modéliser les cas d’utilisation, les fonctionnalités, les processus, de représenter les interactions entre composants et acteurs, de préparer la stratégie de déploiement de la solution cible… Bien entendu, il faut apprendre sa syntaxe, mais même sans l’appliquer stricto sensu, ses diagrammes et vues permettent à l’analyste d’affaires d’analyser et de rédiger la plupart de ses livrables.

****

**MERISE**

Merise est une méthode séquentielle, se distinguant des approches itératives et incrémentales. Elle se révèle particulièrement efficace pour l'analyse, la conception et la mise en œuvre de systèmes d’information. Bien que requérant l'apprentissage de sa syntaxe, tout comme l’UML, sa maîtrise en fait un instrument puissant et rigoureux, surtout lorsqu'elle est partagée entre développeurs et Business Analysts. Cependant, elle tend à être de moins en moins enseignée dans les écoles d’ingénieurs et les programmes de master informatique, au profit de l’UML, et sa réputation de « lourdeur » la rend moins adaptée aux projets agiles. C'est pourquoi nous privilégions l’UML, un langage formel et normalisé, gage de stabilité et facilitant l’utilisation des outils, en tant que support de communication performant."

## Présentation de la méthode utilisée

UML se définit comme un langage de Modélisation Unifié, en anglais Unified Modeling Language. Il se présente comme le langage de modélisation graphique à base de pictogrammes conçu comme une méthode normalisée de visualisation dans les domaines du développement logiciel et de la conception orientée objet. Ce langage est constitué de diagrammes intégrés par les développeurs informatiques pour la représentation visuelle des objets, des états et des processus dans un logiciel ou dans un système. Cependant, UML intéresse fortement les spécialistes de l’ingénierie système. UML unifie également les notations nécessaires aux différentes activités d’un processus de développement et offre, par ce biais, le moyen d’établir le suivi des décisions prises, depuis la spécification jusqu’au codage. Dans ce cadre, un concept appartenant aux besoins des utilisateurs projette sa réalité dans le modèle de conception et dans le codage

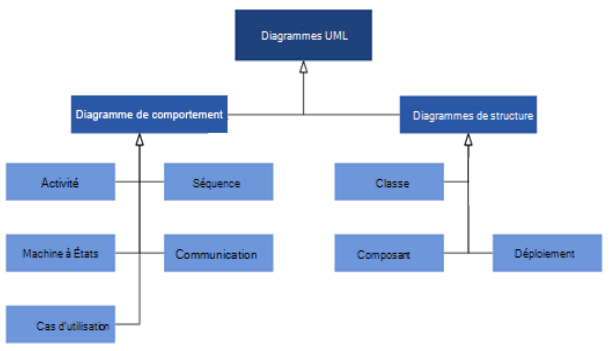


Figure 2 Liste des diagrammes UML

## Présentation du diagramme de comportement

### **Identification des acteurs**

Un acteur d’un système représente une entité externe à celui-ci qui interagit directement (saisie de données, réception d’informations…) avec lui. Les acteurs sont identifiés en observant les utilisateurs directs du système, les responsables de la maintenance, ainsi que les autres systèmes qui interagissent avec lui. Pour notre système, nous avons comme acteurs :

* Le client
* Le visiteur
* L’administrateur

|  |  |
| --- | --- |
| **Acteurs** | **Cas d’utilisation** |
| Visiteur | * Écouter la radio en ligne * Lire les articles du blog |
| Client | * Tous les cas d’utilisation de l’acteur `visiteur` * S’authentifier * Souscrire á un service * Payer en ligne * Modifier son profile * Suivre sa demande |
| Administrateur | * Tous les cas d’utilisation de l’acteur ‘client’ * Voir la liste des demandes * Valider une demande * Supprimer des comptes utilisateurs * Ecrire un article de blog |

Tableau 2 liste des acteurs et des cas d'utilisation

### **Elaboration du diagramme de cas d’utilisation**

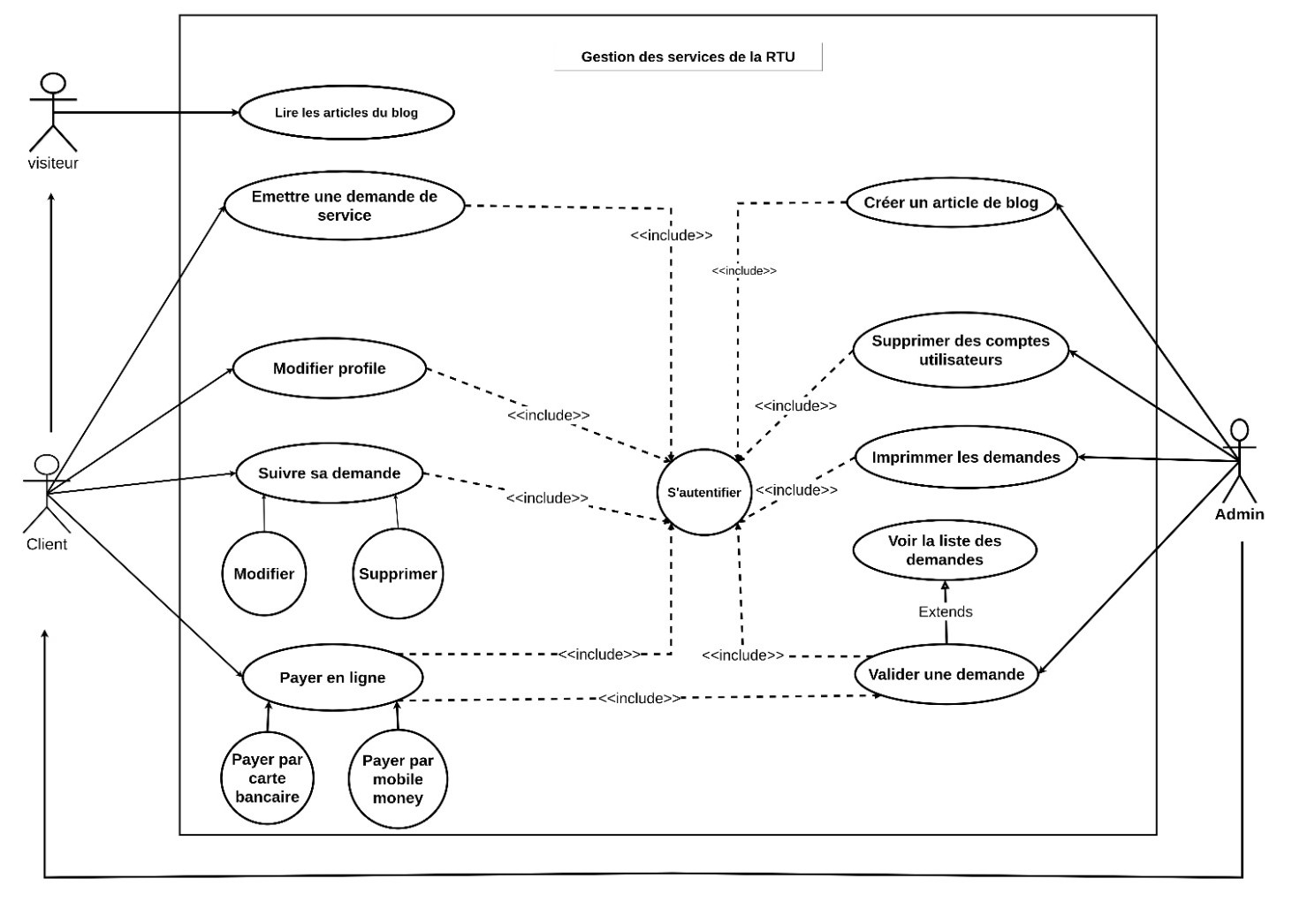


Figure 3 Diagramme de cas d'utilisation

### **Description textuelle du diagramme des cas d’utilisation**

* Cas d’utilisation s’authentifier

**Titre** : S’authentifier

**Résumé** : L'authentification est la première étape essentielle que tout utilisateur du système doit franchir avant de pouvoir effectuer une demande.

**Acteurs** : Utilisateur (Client), Administrateur (Admin)

**Description du scénario :** Ce cas d'utilisation démarre lorsque l'utilisateur souhaite accéder à son tableau de bord. Le système affiche alors l'interface d'authentification, vérifie l'existence de l'utilisateur et lui accorde l'accès à la plateforme.

**Préconditions** : Être un utilisateur disposant d'une adresse e-mail et d'un mot de passe.

**Enchainement nominal** :

1- L'utilisateur saisit l'URL de la plateforme dans la barre de recherche de son navigateur.

2- La page de connexion s'affiche en premier plan.

3- L'utilisateur entre son adresse e-mail et son mot de passe.

4- Après avoir appuyé sur le bouton "Se connecter", l'utilisateur valide ses informations.

5- Le système vérifie la concordance entre l'adresse e-mail et le mot de passe fournis.

6- En cas de succès, l'utilisateur est redirigé vers sa page d'accueil et ses informations de connexion sont enregistrées.

**Alternative** : Si l'appareil utilisé par l'utilisateur n'est pas enregistré dans la base de données, le système ne propose pas la page de connexion et affiche une erreur 404. En cas d'informations invalides ou si l'utilisateur n'est pas enregistré, le système affiche un message d'erreur et réaffiche la page de connexion.

* Cas d’utilisation souscrire á un service de la radio

**Titre :** Souscrire á un service de la radio

**Résumé :** Démarche á suivre pour souscrire á un service

**Acteur :** Client, Admin

**Description du scenario :** Ce cas commencelorsque l’utilisateur veut souscrire á un service, le système affiche un formulaire á remplir

**Précondition :** Être un utilisateur connecté

**Enchainement nominal :**

1. L'utilisateur accède à la section "Services" de la plateforme de la radio.

2. L'utilisateur parcourt les différents services proposés par la radio.

3. L'utilisateur sélectionne le service auquel il souhaite souscrire.

4. Le système affiche les détails du service sélectionné, y compris les conditions de souscription et les tarifs.

5. L'utilisateur examine les informations fournies et décide de souscrire au service.

6. L'utilisateur clique sur le bouton "Souscrire" ou une option similaire.

7. Le système vérifie la disponibilité du service et la validité des informations fournies par l'utilisateur.

8. Si toutes les conditions sont remplies, le système enregistre la souscription de l'utilisateur au service.

9. Le système envoie une confirmation de la souscription à l'utilisateur par e-mail ou par tout autre moyen de communication.

10. L'utilisateur peut désormais accéder au service souscrit via la plateforme de la radio.

**Scenario alternatif** :

- Si le service sélectionné n'est pas disponible ou si les informations fournies par l'utilisateur sont incorrectes, le système affiche un message d'erreur et invite l'utilisateur à corriger les informations ou à sélectionner un autre service.

- Si l'utilisateur n'est pas connecté à la plateforme, le système le redirige vers la page de connexion et lui demande de se connecter avant de pouvoir souscrire à un service.

**Postconditions** :

- L'utilisateur a souscrit avec succès au service de la radio et peut en bénéficier selon les conditions spécifiées.

## Présentation du diagramme d’activité

Le diagramme d’activités représente les activités réalisées par un ou plusieurs objets. Il peut correspondre à la description détaillée d’une activité du diagramme d’états-transitions ou à la description d’une méthode. Il peut également décrire l’activité d’un système ou d’un sous-système en attribuant des responsabilités à chaque acteur. Le diagramme d’activités constitue également un bon choix pour décrire un cas d’utilisation.

* Diagramme d’activité du cas << s’inscrire >>

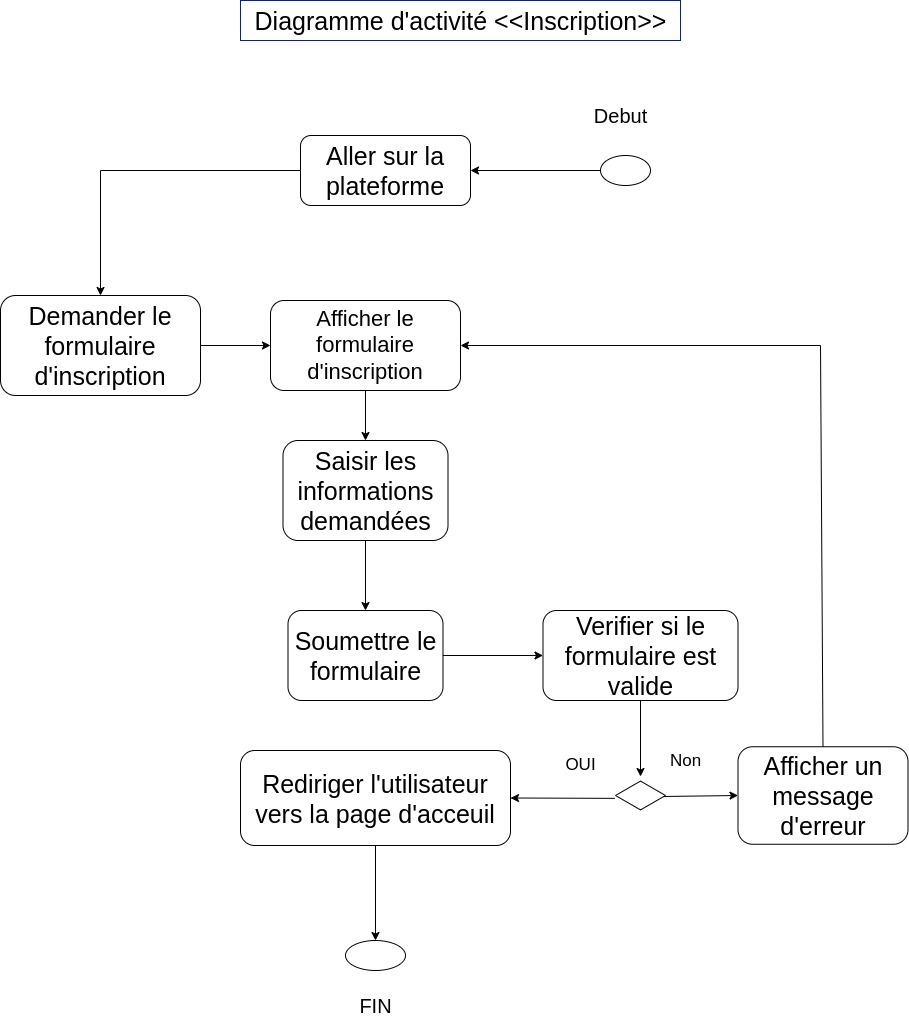


Figure 4 Diagramme d'activité du cas <<s'inscrire>>

* Diagramme d’activité du cas <<Souscrire à une demande>>

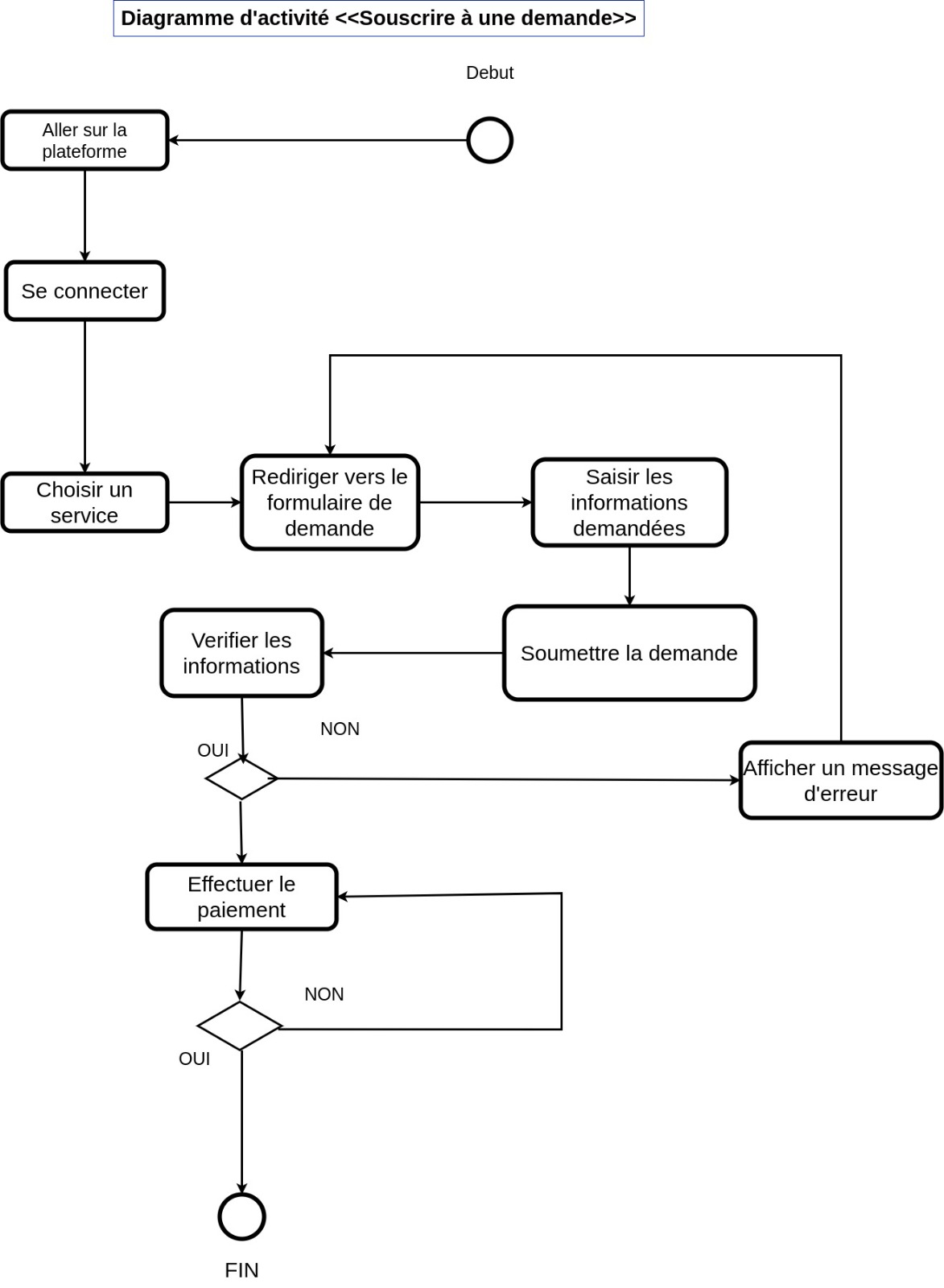


Figure 5 Diagramme d’activité du cas <<Souscrire à un service>>

## Présentation du diagramme de séquence

Le diagramme de séquence décrit la dynamique du système. À moins de modéliser un très petit système, il est difficile de représenter toute la dynamique d’un système sur un seul diagramme. Aussi la dynamique globale sera représentée par un ensemble de diagrammes de séquence, chacun étant généralement lié à une sous fonction du système. Le diagramme de séquence décrit les interactions entre un groupe d’objets en montrant, de façon séquentielle, les envois de message qui interviennent entre les objets. Le diagramme peut également montrer les flux de données échangées lors des envois de message.

* Diagramme de séquence du cas <<S’authentifier >>

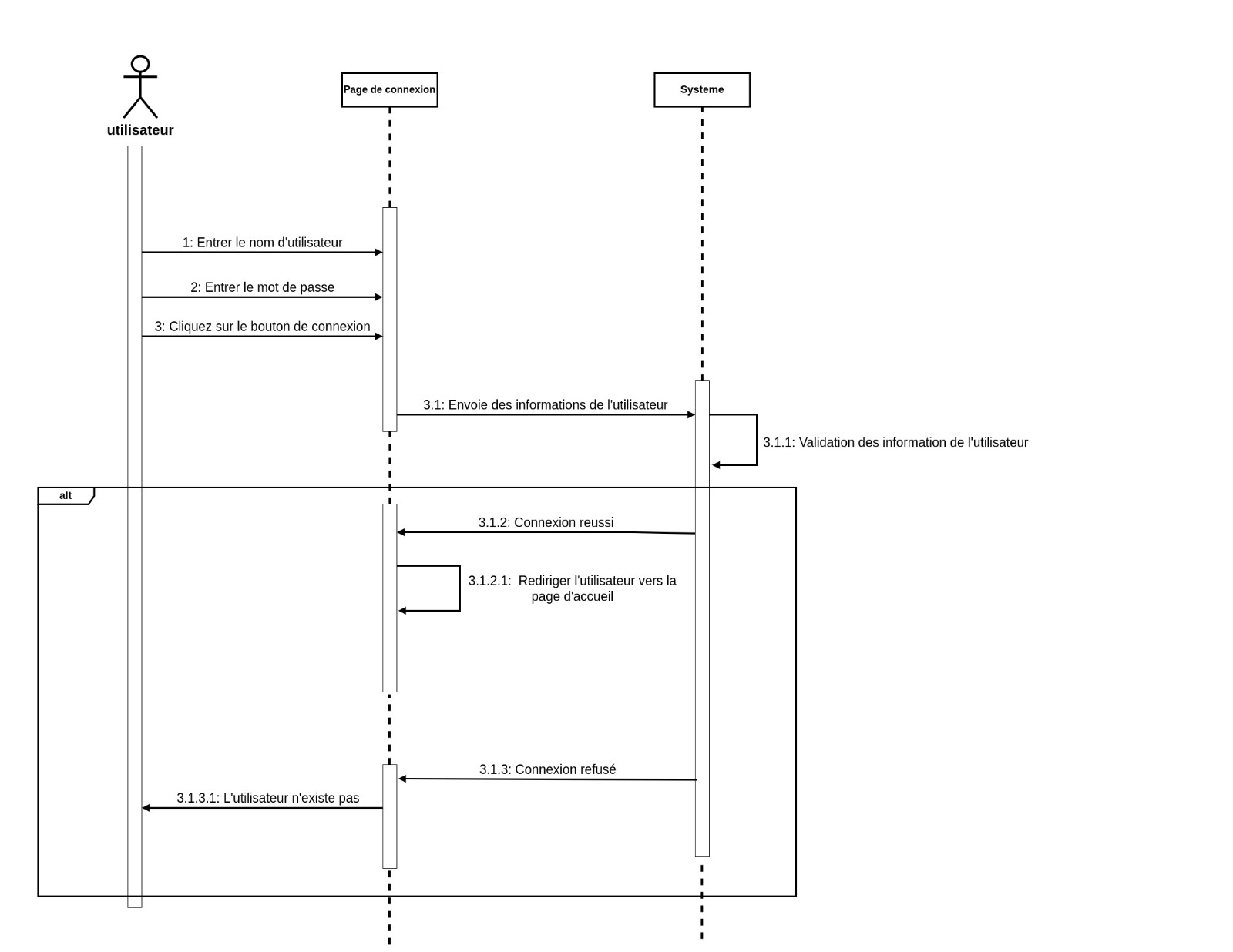


Figure 6 Diagramme de séquence du cas <<S'authentifier>>

* Diagramme de séquence du cas <<Modifier une demande >>

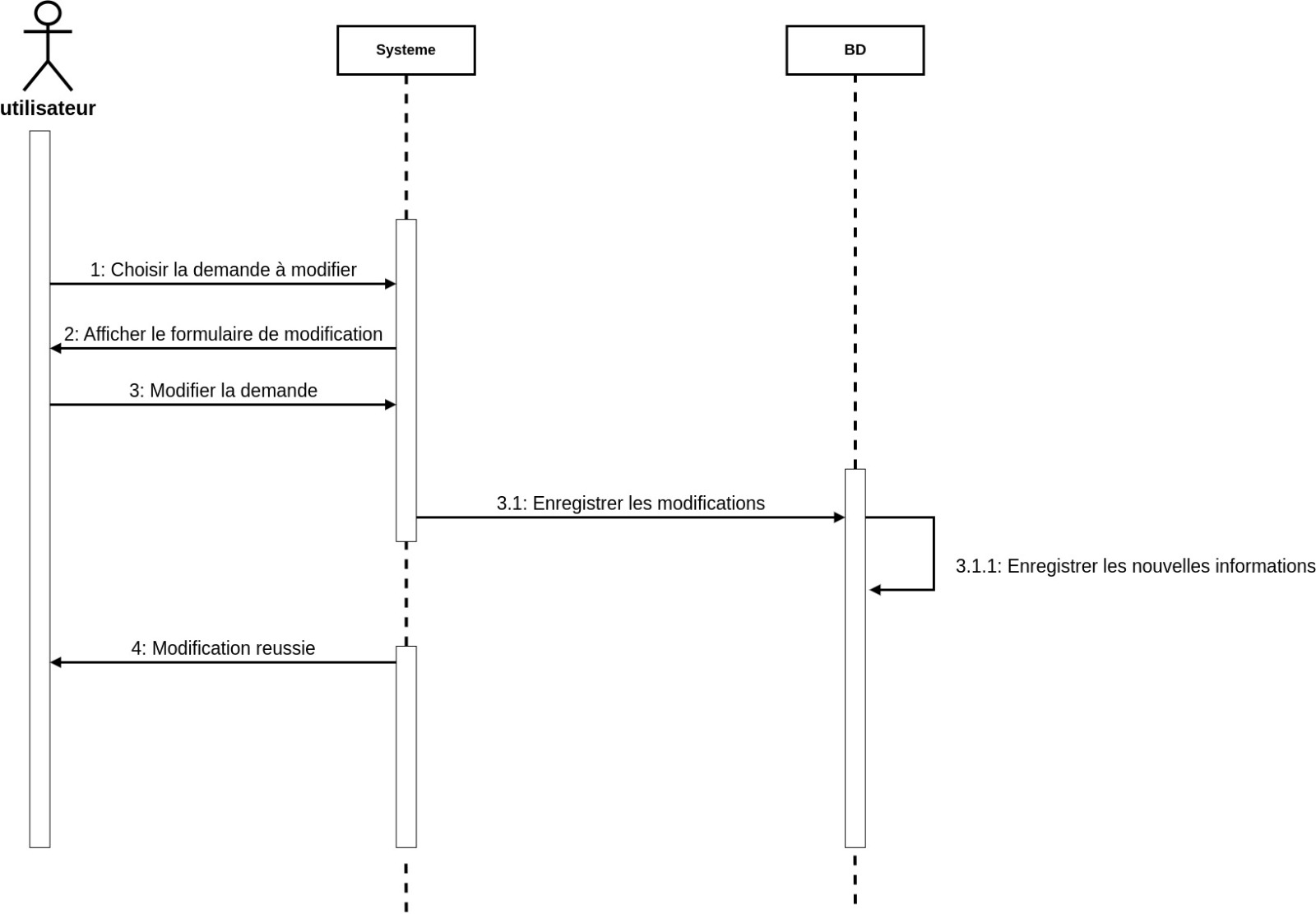


Figure 7 Diagramme de séquence du cas <<modifier une demande>>

## Présentation du diagramme de classe

Le diagramme de classe représente bien la structure statique du code, par le biais des attributs et des relations entre classes. C'est un diagramme principal qui est la vue du plus haut niveau avec l'ensemble des classes de l'application. Il montre les briques de base statiques : classes, associations, interfaces, attributs, opérations, généralisations, etc.

### Diagramme de classe

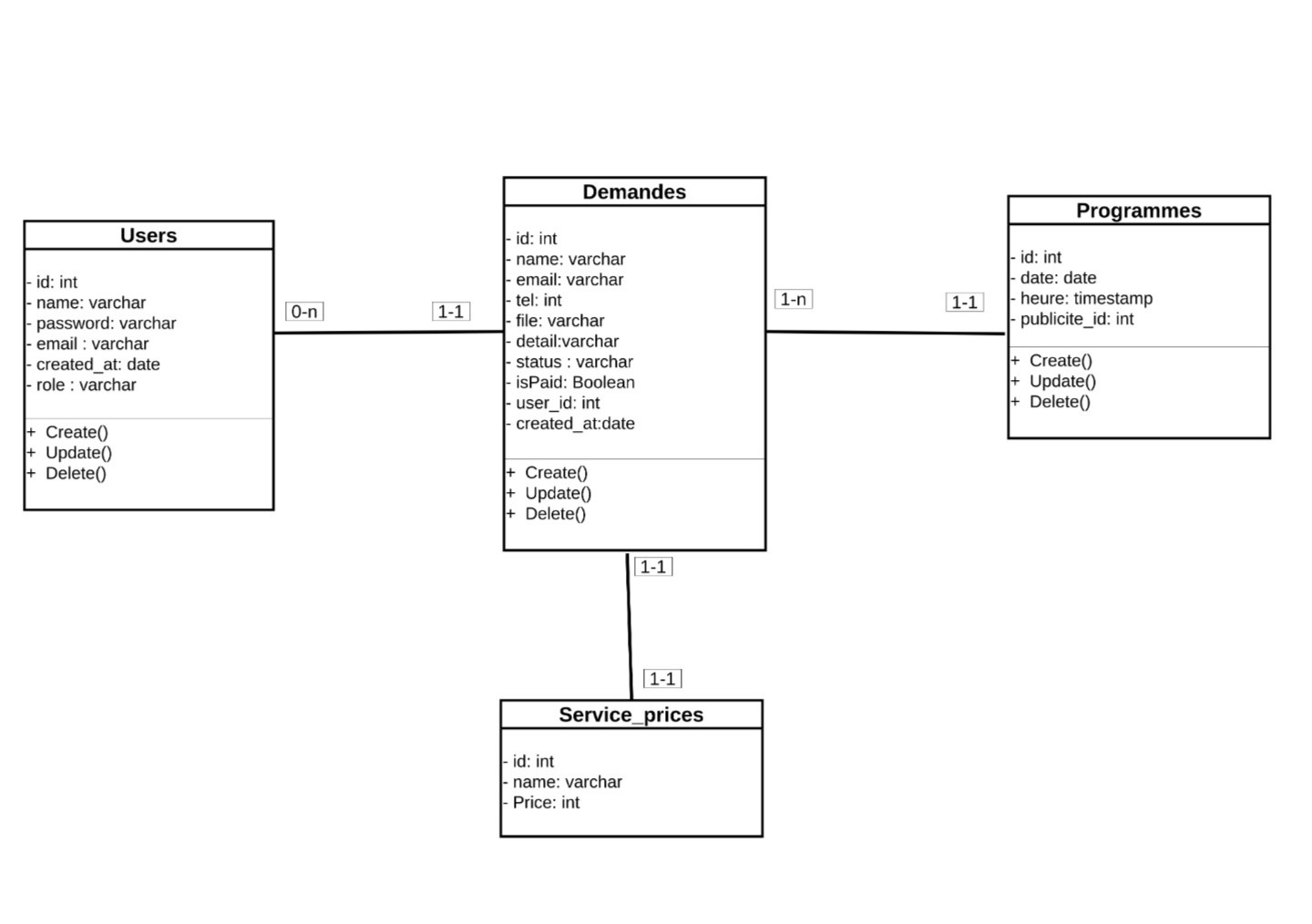


Figure 8 Diagramme de classe

### Modèle logique relationnel

**Users** (id,nom,password,email,role,created\_at )

**Demandes** (service\_id, tel,name,file,user\_id,status,isPaid,detail,created\_at)

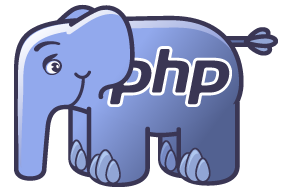
**Service\_prices(**id,price,name**)**

**Programmes** (programme\_id,date,heure,publicite\_id)

# Chapitre 4 : Conception du système

## Présentation des outils de développement

### **Les langages de programmation utilisés**

* **PHP**

Hypertext Preprocessor, plus connu sous son sigle PHP (sigle auto-référentiel), est un langage de programmation libre, principalement utilisé pour produire des pages Web dynamiques via un serveur web, mais pouvant également fonctionner comme n'importe quel langage interprété.



* **TYPESCRIPT**

TypeScript est un langage de programmation libre et open source développée par Microsoft qui a pour but d'améliorer et de sécuriser la production de code JavaScript. Il s'agit d'un sur-ensemble syntaxique strict de JavaScript (c'est-à-dire que tout code JavaScript correct peut être utilisé avec TypeScript)

### **Autres langages**

* + - **HTML 5**

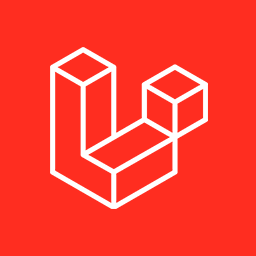
HTML signifie « HyperText Markup Langage » qu’on peut traduire par langage de balises pour l’hypertexte. C’est un langage de balisage qui sert à indiquer au navigateur comment structurer les pages web visitées.



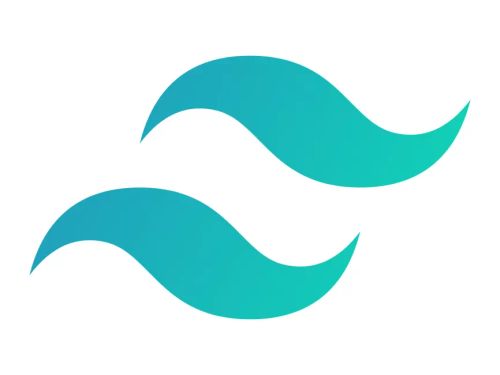
* **CSS 3**

Les feuilles de style en cascade, généralement appelées CSS de l’anglais « Cascading Style Sheets », forment un langage informatique qui décrit la présentation des documents HTML et XML.

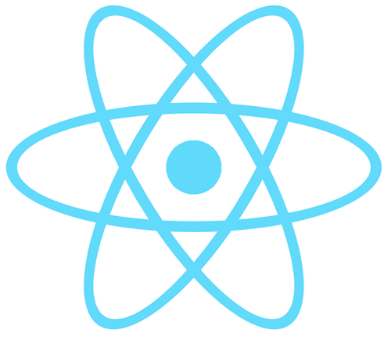
### **Frameworks**

* **LARAVEL 10**

Laravel est un Framework web open-source écrit en PHP respectant le principe modèle-vue-contrôleur et entièrement développé en programmation orientée objet. Laravel est distribué sous licence MIT, avec ses sources hébergées sur GitHub. Laravel a été créé par Taylor Otwell en juin 2011.

* + - * **TAILWIND CSS**

Tailwind CSS est un Framework CSS populaire intégralement utility-first. Il fournit aux utilisateurs des classes CSS de bas niveau en PostCSS qui peuvent être utilisées pour **définir des composants et des designs de manière indépendante**. Tailwind CSS fut développé par Adam Wathan et est sorti pour la première fois en 2017. Depuis lors, il a été adopté par des millions d’utilisateurs.

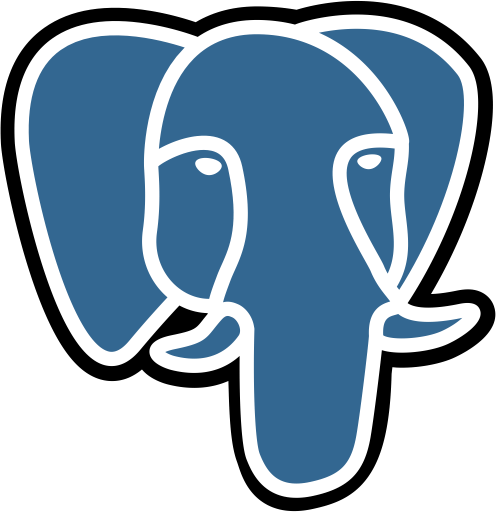


* + - * **REACT JS**

React.js, communément appelé simplement React, est une [bibliothèque JavaScript](https://kinsta.com/fr/blog/bibliotheques-javascript/) utilisée pour construire des interfaces utilisateur. Chaque application web React est composée de composants réutilisables qui constituent des parties de l’interface utilisateur – nous pouvons avoir un composant distinct pour notre barre de navigation, un pour le pied de page, un autre pour le contenu principal, et ainsi de suite. Vous comprendrez mieux cela lorsque nous arriverons à la section où nous devons travailler avec des composants.

ssss

### **SGBD**



* **POSTGRESQL**

PostgreSQL est un [système de gestion de base de données](https://fr.wikipedia.org/wiki/Syst%C3%A8me_de_gestion_de_base_de_donn%C3%A9es) [relationnelle](https://fr.wikipedia.org/wiki/Base_de_donn%C3%A9es_relationnelle) et [objet](https://fr.wikipedia.org/wiki/Base_de_donn%C3%A9es_orient%C3%A9e_objet) ([SGBDRO](https://fr.wikipedia.org/wiki/Syst%C3%A8me_de_gestion_de_base_de_donn%C3%A9es_relationnel-objet)). C'est un outil [libre](https://fr.wikipedia.org/wiki/Logiciel_libre) disponible selon les termes d'une licence de type [BSD](https://fr.wikipedia.org/wiki/Licence_BSD). Ce système est comparable à d'autres systèmes de gestion de base de données, qu'ils soient libres (comme [MariaDB](https://fr.wikipedia.org/wiki/MariaDB" \o "MariaDB) et [Firebird](https://fr.wikipedia.org/wiki/Firebird_(base_de_donn%C3%A9es)" \o "Firebird (base de données))), ou propriétaires (comme [Oracle](https://fr.wikipedia.org/wiki/Oracle_Database), [MySQL](https://fr.wikipedia.org/wiki/MySQL), [Sybase](https://fr.wikipedia.org/wiki/Sybase), [DB2](https://fr.wikipedia.org/wiki/DB2), [Informix](https://fr.wikipedia.org/wiki/Informix) et [Microsoft SQL Server](https://fr.wikipedia.org/wiki/Microsoft_SQL_Server)). Comme les projets libres [Apache](https://fr.wikipedia.org/wiki/Apache_HTTP_Server) et [Linux](https://fr.wikipedia.org/wiki/Linux), PostgreSQL n'est pas contrôlé par une seule entreprise, mais est fondé sur une communauté mondiale de développeurs et d'entreprises.

### **Editeur de code**

* **VISUAL STUDIO CODE**

Visual Studio Code est présenté lors de la conférence des développeurs Build d'avril 2015 comme un éditeur de code [multiplateforme](https://fr.wikipedia.org/wiki/Logiciel_multiplateforme), [open source](https://fr.wikipedia.org/wiki/Open_source) et gratuit, supportant une dizaine de [langages](https://fr.wikipedia.org/wiki/Langage_informatique). Il est basé sur [Electron](https://fr.wikipedia.org/wiki/Electron_(framework)), une structure utilisée pour déployer des applications [Node.js](https://fr.wikipedia.org/wiki/Node.js) pour le bureau exécuté sur le moteur Blink. Bien qu'il utilise le Framework [Electron](https://fr.wikipedia.org/wiki/Electron_(framework)), le logiciel n'utilise pas [Atom](https://fr.wikipedia.org/wiki/Atom_(%C3%A9diteur_de_texte)) mais utilise le même composant éditeur (nommé Monaco) utilisé dans Azure DevOps (anciennement appelé Visual Studio Online et Visual Studio Team Services).

### **Les autres outils**

* **Git**

Git est un logiciel de gestion de versions décentralisé. C'est un logiciel libre et gratuit, créé en 2005 par Linus Torvalds, auteur du noyau Linux, et distribué selon les termes de la licence publique générale GNU version 2

## Présentation de l’application

### **Quelques interfaces de notre projet**

1. **Page d’accueil**

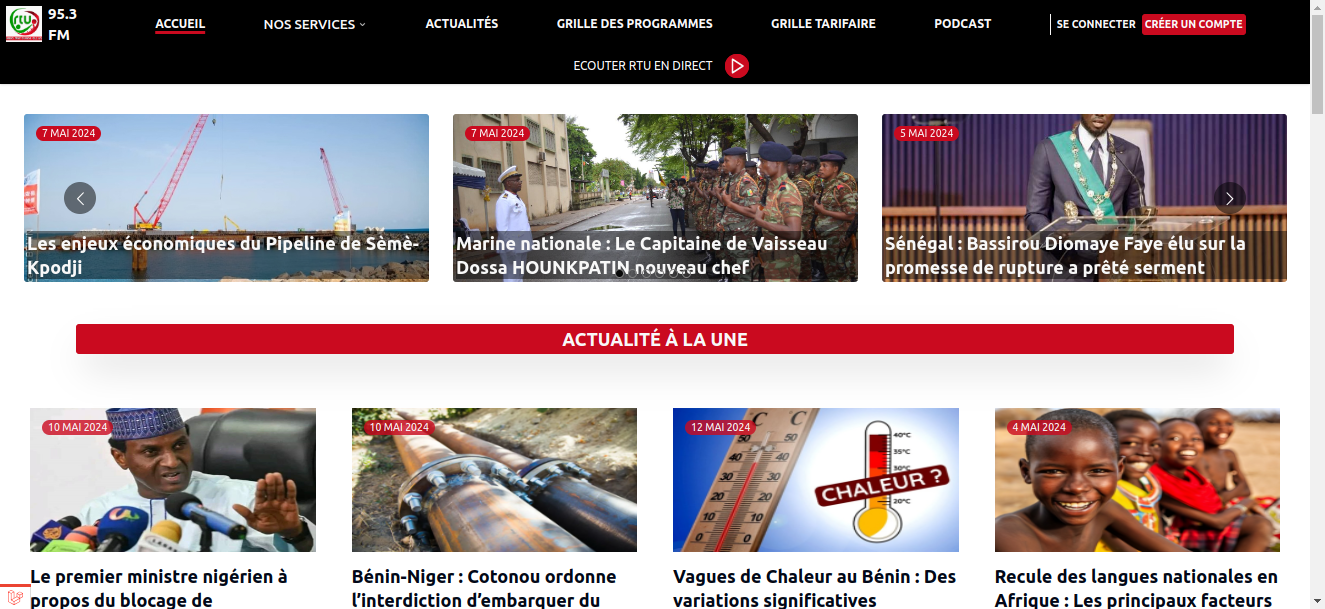
La page d’accueil de la plateforme que nous avons conçue se veut accueillante et intuitive 



Figure 9 Page d'accueil

1. **Interface de connexion**

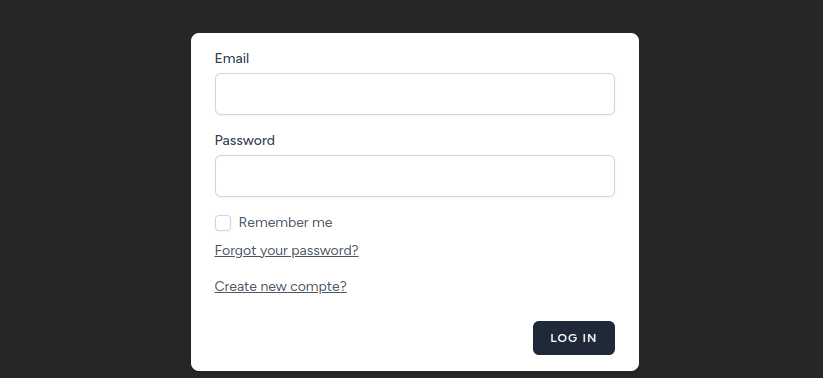
L'interface de connexion est conçue pour permettre aux utilisateurs d'accéder facilement aux services offerts par la radio

Figure 10 Interface de connexion

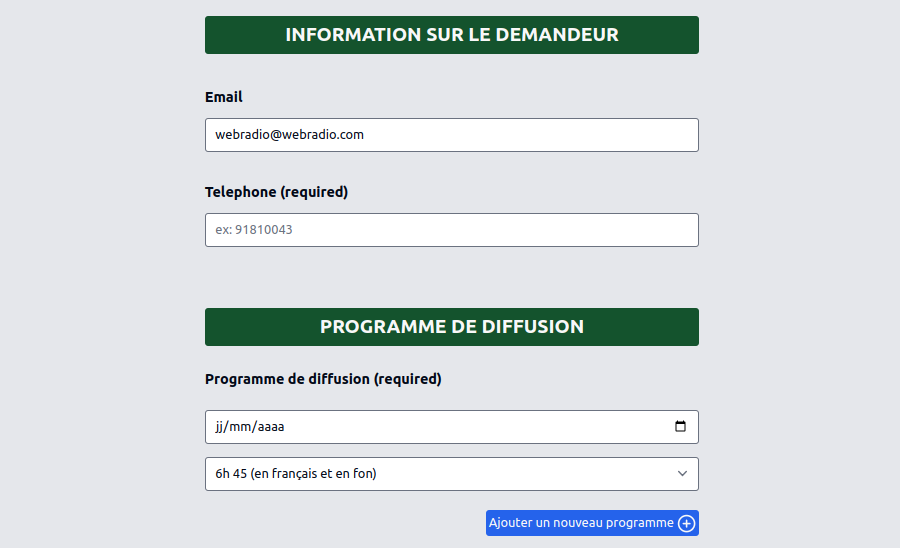
1. **Section podcasts**

Cette section affiche des émissions á écouter sous forme de podcasts



Figure 11 Section podcasts

1. **Interface utilisée pour faire une demande de spot publicitaire**



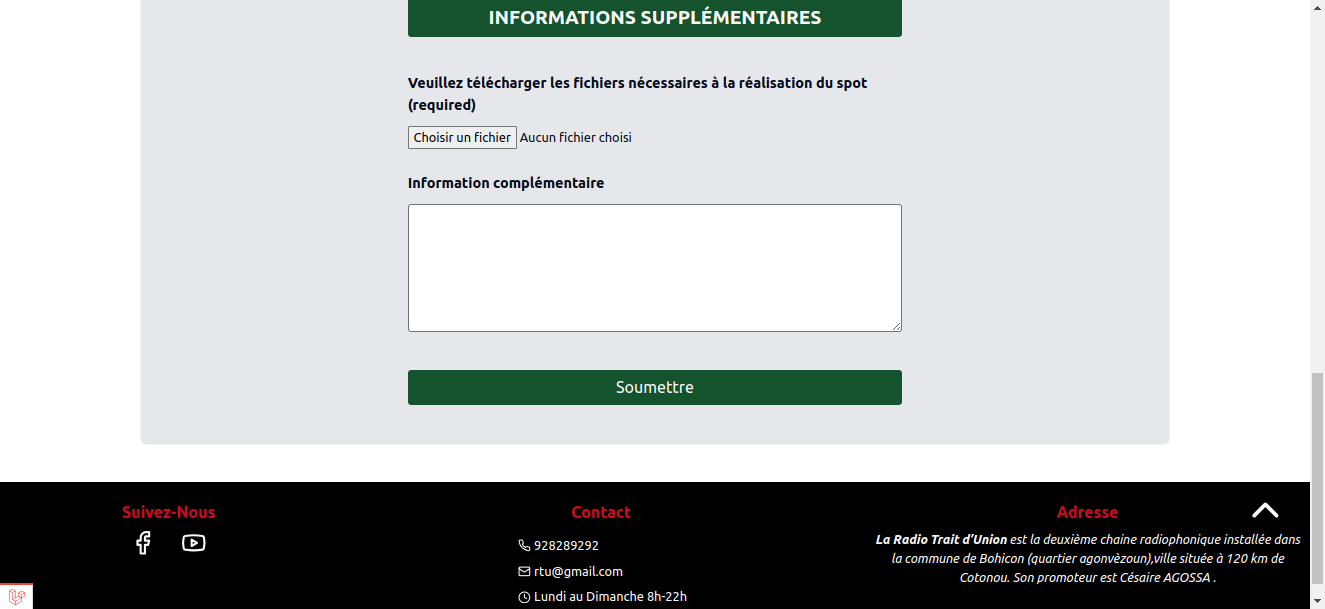


Figure 12 Interface utilisée pour faire une demande de spot publicitaire

1. **Interface de paiement**

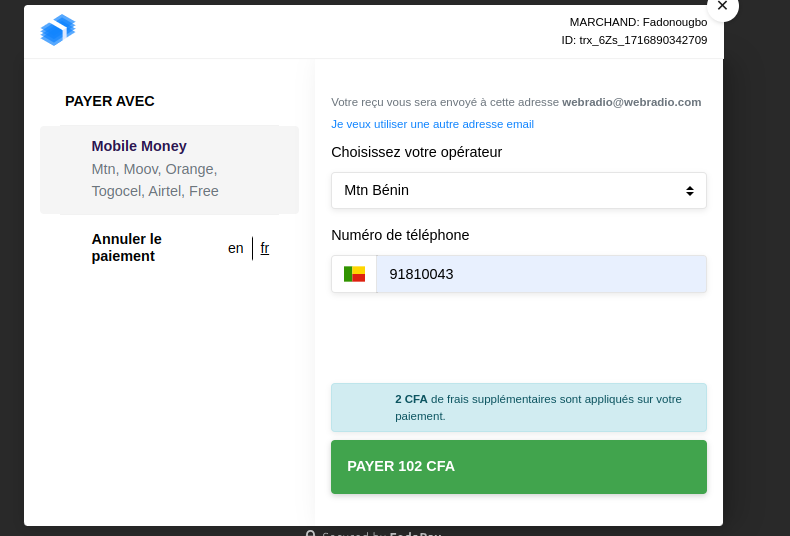


Figure 13 Interface de paiement

1. **Dashboard utilisateur**



Figure 14 Dashboard utilisateur

1. **Dashboard administrateur**

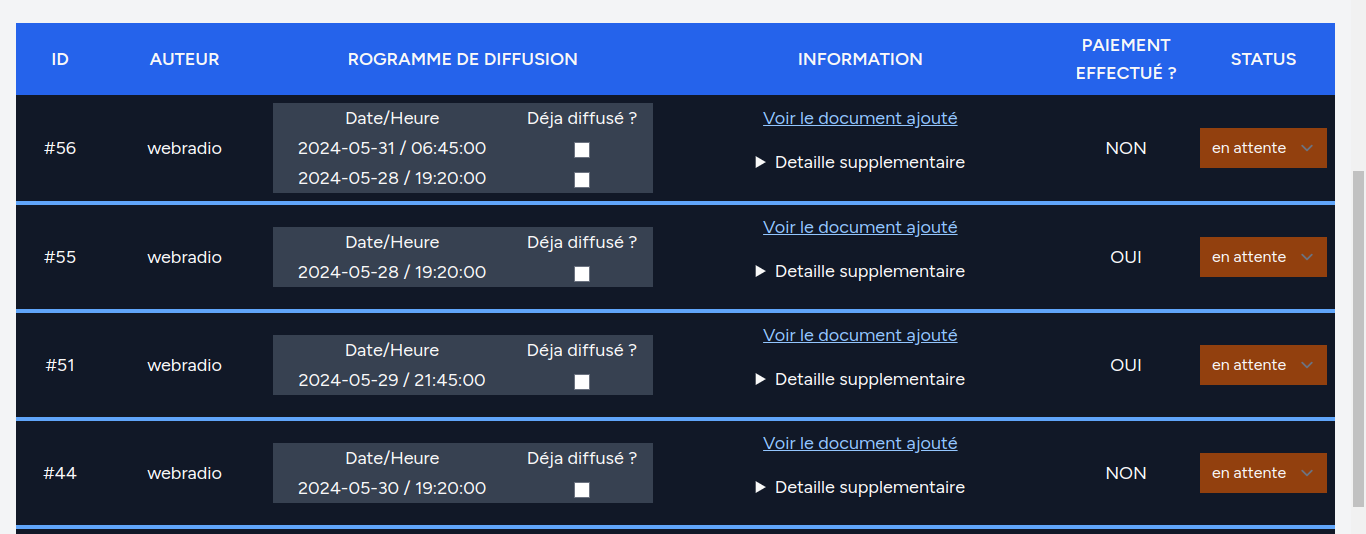


Figure 15 Dashboard administrateur

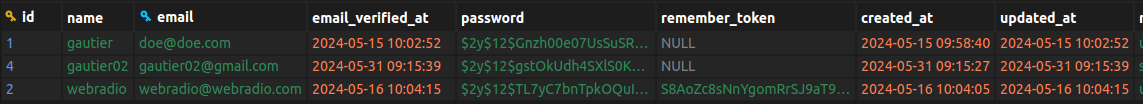
1. **Vue de la plateforme sur mobile**



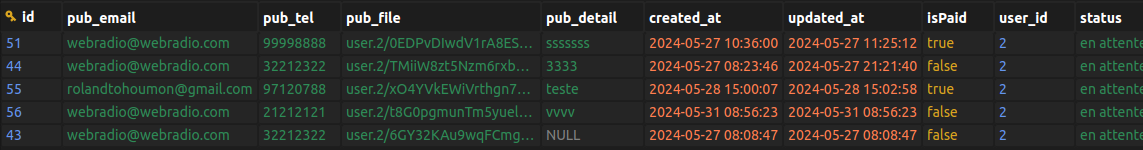
Figure 16 Vue de la plateforme sur mobile

### **Quelques tables de la base de données**

1. **Table users**



1. **Table demandes**



### **Quelques codes sources du projet**

1. **Code utilisé pour la migration de la table users**

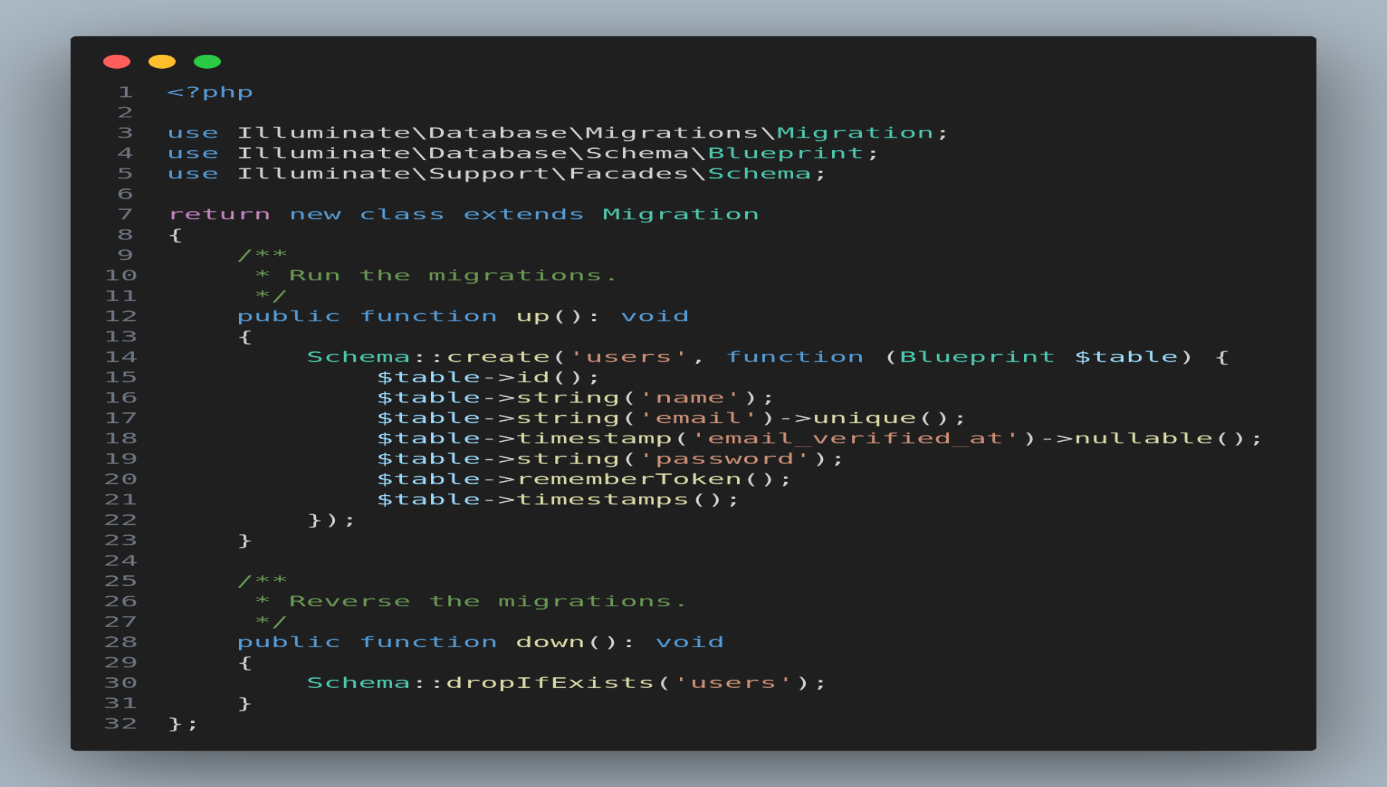


Figure 17 Code utilisé pour la migration de la table users

1. **Code utilisé pour enregistrer une demande publicitaire**

Figure 18 Code utilisé pour enregistrer une demande publicitaire

1. **Code utilisé pour afficher la vue de la route dashboard**



Figure 19 Route dashboard

# CONCLUSION

En conclusion, notre stage à la Radio Trait d’Union a revêtu une importance cruciale, car il nous a permis de concrétiser les connaissances acquises au fil de notre formation, tant sur le plan personnel que professionnel. Notre contribution, bien que modeste, a permis de relever certains problèmes au sein de la radio, en particulier l'absence d’une plateforme web et d'un logiciel pour la digitalisation des services offerts par la radio. Après une analyse approfondie des obstacles rencontrés, nous avons choisi de concentrer nos efforts sur la conception et la réalisation de cette plateforme web dédiée à la Radio Trait d’Union. Bien que notre parcours ait été jalonné de défis, ceux-ci n'ont pas entravé la finalisation de notre projet. Toutefois, il est essentiel de reconnaître que la solution que nous proposons comporte des limites et reste perfectible. Nous espérons que notre travail pourra réellement catalyser le développement de la radio, en contribuant positivement à son évolution.

# REFERENCES

* Webographie
* php.net
* laravel.com
* www.google.com
* www.devdocs.io
* [www.wikipedia.com](http://www.wikipedia.com)
* Bibliographie
* Mémoire OLONI E. Joseph & AKOWANOU Sylvestre, CONCEPTION ET REALISATION D’UNE APPLICATION DE GESTION DES AFFAIRES PEDAGOGIQUES : CAS DE L’INSTITUT DE FORMATION POLYTECHNIQUE VALEUR UNIVERSELLE
* Mémoire BODJRENOU Y.G Ravène et BLENON Mardoché Charbel, DEVELEPPEMENT D’UNE PLATEFORME DE SUPERVISION D’UN RESEAU LOCAL : Cas de 2CL-Services

TABLE DES MATIERES

[AVERTISSEMENT i](#_Toc168503618)

[DÉDICACE iii](#_Toc168503619)

[REMERCIEMENTS iv](#_Toc168503620)

[LISTE DES SIGLES ET ABRÉVIATIONS v](#_Toc168503621)

[LISTE DES FIGURES vi](#_Toc168503622)

[LISTE DES TABLEAUX vii](#_Toc168503623)

[SOMMAIRE viii](#_Toc168503624)

[INTRODUCTION 1](#_Toc168503625)

[CHAPITRE 1 : CADRE CONDITIONNEL DU STAGE 2](file:///C:\Users\ADMIN\Documents\gautier_utils_memoire\gautier_new_memoire%20-%20Copie.docx#_Toc168503626)

[I. Historique de la Radio Trait d’Union 3](#_Toc168503627)

[II. Organisation fonctionnelle 3](#_Toc168503628)

[1. Fonctionnement de la radio trait d’union 3](#_Toc168503629)

[2. Mission de la radio trait d’union 4](#_Toc168503630)

[3. Organisation de la radio trait d’union 4](#_Toc168503631)

[4. Observation relative au cadre de recherche 7](#_Toc168503632)

[5. Organigramme 8](#_Toc168503633)

[CHAPITRE 2 : PRESENTATION DE LA PLATEFORME Á CONCEVOIR 9](file:///C:\Users\ADMIN\Documents\gautier_utils_memoire\gautier_new_memoire%20-%20Copie.docx#_Toc168503634)

[I. Problématique 10](#_Toc168503635)

[II. Objectifs généraux 10](#_Toc168503636)

[III. Objectifs spécifiques 11](#_Toc168503637)

[IV. Raison du choix du thème 11](#_Toc168503638)

[CHAPITRE 3: METHODOLOGIE 12](file:///C:\Users\ADMIN\Documents\gautier_utils_memoire\gautier_new_memoire%20-%20Copie.docx#_Toc168503639)

[I. Présentation de la méthode utilisée 14](#_Toc168503640)

[II. Présentation du diagramme de comportement 15](#_Toc168503641)

[ **Identification des acteurs** 15](#_Toc168503642)

[ **Elaboration du diagramme de cas d’utilisation** 17](#_Toc168503643)

[ **Description textuelle du diagramme des cas d’utilisation** 17](#_Toc168503644)

[III. Présentation du diagramme d’activité 20](#_Toc168503645)

[IV. Présentation du diagramme de séquence 22](#_Toc168503646)

[V. Présentation du diagramme de classe 24](#_Toc168503647)

[1. Diagramme de classe 25](#_Toc168503648)

[2. Modèle logique relationnel 25](#_Toc168503649)

[Chapitre 4 : Conception du système 26](file:///C:\Users\ADMIN\Documents\gautier_utils_memoire\gautier_new_memoire%20-%20Copie.docx#_Toc168503650)

[I. Présentation des outils de développement 27](#_Toc168503651)

[ **Les langages de programmation utilisés** 27](#_Toc168503652)

[ **Autres langages** 27](#_Toc168503653)

[ **Frameworks** 28](#_Toc168503654)

[ **SGBD** 29](#_Toc168503655)

[ **Editeur de code** 30](#_Toc168503656)

[ **Les autres outils** 30](#_Toc168503657)

[II. Présentation de l’application 31](#_Toc168503658)

[**1.** **Quelques interfaces de notre projet** 31](#_Toc168503659)

[**2.** **Quelques tables de la base de données** 36](#_Toc168503660)

[**3.** **Quelques codes sources du projet** 36](#_Toc168503661)

[CONCLUSION 39](file:///C:\Users\ADMIN\Documents\gautier_utils_memoire\gautier_new_memoire%20-%20Copie.docx#_Toc168503662)

[REFERENCES 41](#_Toc168503663)