

MAIZEROI Gerson

BTS SIO 2^{ème} année

BREVET DE TECHNICIEN SUPERIEUR SERVICE INFORMATIQUE AUX
ORGANISATION OPTION B : SOLUTIONS LOGICIEL APPLICATION METIER

Epreuve E6

Dossier 2 :

***Développement d'une application de gestion
de ticket***

SESSION 2025

Sommaire

1. Introduction	3
1.1 Contexte général	3
1.2 Contexte local : l'entreprise fictive <i>TechnoPlus</i>	3
1.3 Présentation du projet : Application de gestion de tickets de <i>TechnoPlus</i>	3
2. Présentation de l'organisation	4
2.1 Activités de l'organisation	4
2.2 Situation géographique de l'organisation	4
2.3 Personnel de l'organisation.....	4
2.4 Organisation du travail.....	6
2.5. Problématiques rencontrées	6
3. Solutions	7
3.1 Présentation du projet.....	7
3.2 Objectifs du projet.....	7
3.3 Fonctionnement du système.....	7
3.4 Architecture technique	8
3.5 Spécifications fonctionnelles.....	9
3.6 Spécifications techniques	9
3.7 Avantages attendus	10
3.8 Limites et pistes d'évolution	10
3.9 Planning prévisionnel.....	11
3.10 Budget prévisionnel	11
3.11 Critères de réussite.....	11
3.12 Glossaire.....	11
3.13 Étude de faisabilité	12
3.14 Diagramme de cas d'utilisation	12
4. Bilan du projet.....	13
4.1 Fonctionnalités réalisées	13
4.2 Fonctionnalités non implémentées	14
4.3 Contraintes rencontrées	15
4.4 Perspectives d'évolution	15
5. Conclusion	17
6. Annexes.....	18

1. Introduction

1.1 Contexte général

Dans un contexte où les services informatiques jouent un rôle crucial dans le bon fonctionnement des entreprises, la gestion efficace des demandes d'assistance devient une priorité. Nombreuses sont les structures qui rencontrent des difficultés à suivre et à traiter les requêtes techniques, souvent en raison d'un manque d'outils adaptés ou d'une organisation manuelle peu efficace.

Mettre en place une application de gestion de tickets permet non seulement d'automatiser et de centraliser les demandes, mais aussi de garantir un meilleur suivi des incidents, d'améliorer la communication entre les utilisateurs et le service informatique, et de réduire les délais de résolution.

1.2 Contexte local : l'entreprise fictive TechnoPlus

Ce projet s'inscrit dans un scénario fictif dans le cadre de mon cursus en BTS SIO. L'entreprise TechnoPlus, spécialisée dans les services informatiques, souhaite moderniser la gestion de ses demandes d'assistance internes.

Actuellement, les collaborateurs signalent leurs problèmes techniques par e-mail ou oralement, ce qui entraîne des pertes d'information, une mauvaise répartition des tâches, et un manque de visibilité sur l'état des interventions.

Afin de répondre à ces problématiques, il a été proposé de concevoir une application de gestion de tickets pour structurer les demandes, faciliter leur traitement et améliorer le fonctionnement global du service informatique.

1.3 Présentation du projet : Application de gestion de tickets de TechnoPlus

L'objectif de ce projet est de développer une application web permettant aux utilisateurs de soumettre facilement leurs demandes d'assistance et de suivre leur avancement. De leur côté, les techniciens peuvent consulter les tickets, les classer par priorité, et assurer un suivi précis des interventions.

Les fonctionnalités principales incluent :

- L'enregistrement des demandes via un formulaire en ligne ;
- La consultation et la mise à jour du statut des tickets ;
- L'attribution des tickets aux techniciens concernés ;
- Une interface d'administration pour la gestion des utilisateurs et des catégories d'incident.

Ce projet m'a permis de mobiliser mes compétences en développement web, en base de données et en gestion de projet, tout en répondant à un besoin concret dans un environnement professionnel simulé.

2. Présentation de l'organisation

2.1 Activités de l'organisation

TechnoPlus est une entreprise fictive de type PME, spécialisée dans les services informatiques à destination des professionnels. Elle propose un accompagnement complet à ses clients en matière de maintenance, de gestion de parcs informatiques, de sécurité réseau et d'assistance technique.

Son cœur de métier repose sur :

- Le déploiement et la maintenance des infrastructures informatiques ;
- L'assistance utilisateur, aussi bien sur site qu'à distance ;
- Le conseil en sécurité informatique et la mise en place de solutions préventives (pare-feux, sauvegardes, segmentation réseau, etc.).

Grâce à une expertise reconnue dans son domaine, TechnoPlus intervient auprès d'entreprises de tailles variées, principalement situées en Île-de-France.

2.2 Situation géographique de l'organisation

TechnoPlus est localisée à Paris, dans le 12^e arrondissement, au sein d'un quartier d'affaires en plein développement.

Cet emplacement offre plusieurs avantages :

- Une bonne accessibilité pour les collaborateurs et les clients grâce aux transports en commun ;
- Une proximité avec de nombreux partenaires et fournisseurs du secteur IT ;
- Un environnement professionnel stimulant, propice aux collaborations et aux opportunités commerciales.

2.3 Personnel de l'organisation

2.3.1 Composition du personnel

L'entreprise emploie environ 50 salariés répartis sur différents pôles :

- **Service technique** : 10 techniciens assurent l'installation, la maintenance et le support des systèmes clients.

- **Développement informatique** : 5 développeurs conçoivent des applications internes et proposent des solutions logicielles sur mesure.
- **Support client** : équipe chargée de répondre aux demandes quotidiennes, de gérer les tickets et d'accompagner les utilisateurs.
- **Services administratifs** : 5 personnes gèrent la comptabilité, les ressources humaines et la logistique.
- **Direction** : une équipe de 2 personnes encadre l'ensemble des activités et définit la stratégie de l'entreprise.

2.3.2 Organisation du système d'information (SI)

Le système d'information de TechnoPlus est structuré pour répondre aux besoins internes tout en servant de modèle pour les solutions proposées aux clients :

- **Infrastructure matérielle** :
 - Un **serveur principal** centralise les données et héberge les applications métiers.
 - Environ **50 postes de travail** sous Windows 10.
 - Un réseau local performant, segmenté par VLAN, protégé par un **pare-feu** professionnel.
- **Outils logiciels** :
 - Solutions bureautiques, outils de ticketing (en cours de modernisation), CRM, outils de supervision réseau.
- **Sécurité** :
 - Sauvegardes régulières, antivirus à jour, contrôle d'accès réseau.
 - Gestion des droits via un annuaire centralisé.

2.4 Organisation du travail

Le travail est réparti par pôles afin d'assurer une répartition claire des responsabilités :

- **Équipe support** : prend en charge les incidents techniques signalés par les clients ou les collaborateurs.
- **Développement** : conçoit des outils pour optimiser les processus internes ou répondre aux besoins clients spécifiques.
- **Administration** : gère la logistique, les contrats, la facturation et les aspects RH.
- **Direction** : supervise les opérations et veille à la coordination entre les services.

Des réunions hebdomadaires sont organisées pour faire le point sur les interventions en cours, les tickets à traiter et les projets internes à venir.

2.5. Problématiques rencontrées

Avant la mise en œuvre de l'application de gestion des tickets, TechnoPlus utilisait un système basé sur les e-mails pour traiter les demandes d'assistance. Cette méthode posait plusieurs difficultés :

- **Manque de traçabilité** : il était difficile de suivre l'avancement de chaque demande, ce qui provoquait des retards ou des oublis.
- **Risque de perte d'information** : certains tickets étaient oubliés ou mal redirigés, faute de centralisation.
- **Absence de tableau de bord** : la direction ne disposait pas d'indicateurs précis sur la charge de travail du support.
- **Historique inexistant** : il n'y avait pas de base de données permettant de consulter les interventions passées, ce qui compliquait l'analyse des récurrences ou la résolution de problèmes similaires.

Face à ces constats, la création d'une application de gestion centralisée des tickets est apparue comme une solution incontournable pour améliorer l'efficacité du service support, optimiser la communication interne et renforcer la satisfaction des utilisateurs.

3. Solutions

3.1 Présentation du projet

Le projet de gestion des tickets d'incidents chez TechnoPlus vise à centraliser, suivre et résoudre les demandes d'assistance informatique des collaborateurs. Chaque utilisateur, confronté à une anomalie technique, peut créer un ticket détaillant son problème via une interface simple. Ces tickets sont ensuite pris en charge par les techniciens ou les référents informatiques, qui les traitent selon leur urgence et leur niveau de priorité.

Cette application, développée en interne, s'inscrit dans une logique d'amélioration continue du support informatique. Elle permet de rationaliser les échanges entre les utilisateurs et l'équipe IT, tout en assurant une traçabilité complète de chaque incident. Le projet s'insère dans le cadre du BTS SIO, avec un objectif pédagogique axé sur le développement d'outils métiers adaptés aux besoins spécifiques d'une organisation.

3.2 Objectifs du projet

Le projet poursuit plusieurs objectifs concrets, alignés avec les besoins opérationnels de TechnoPlus :

- Centraliser toutes les demandes d'assistance au sein d'un outil unique ;
- Réduire les délais de traitement des tickets en facilitant leur suivi ;
- Offrir aux techniciens une vision claire des tâches à réaliser ;
- Prioriser les interventions selon la gravité des incidents ;
- Assurer un meilleur suivi pour les utilisateurs en attente de résolution ;
- Produire des statistiques d'activité utiles pour piloter l'équipe informatique.

3.3 Fonctionnement du système

3.3.1 Création et enregistrement des tickets

Lorsqu'un utilisateur rencontre un problème technique (ex : imprimante non fonctionnelle, problème réseau, bug logiciel...), il se connecte à la plateforme pour créer un ticket. Il renseigne un titre, une description du problème, éventuellement une pièce jointe, et peut préciser la priorité. Chaque ticket est enregistré avec un identifiant unique et un statut initial (ex : "ouvert").

3.3.2 Suivi et modification des tickets

Le personnel IT a accès à une interface qui affiche l'ensemble des tickets avec leurs détails. Les techniciens peuvent :

- Répondre à l'utilisateur via un système de commentaires internes ;
- Modifier le statut (ex : "en cours", "en attente", "résolu") ;
- Attribuer un ticket à un technicien ou à un groupe d'intervention ;
- Ajouter un historique d'action (diagnostic, opération réalisée, etc.).

3.3.3 Gestion des statuts et priorités

Chaque ticket suit un cycle de vie structuré avec plusieurs statuts prédéfinis. La priorité du ticket (faible, normale, haute, critique) permet d'orienter l'ordre de traitement. Un algorithme simple peut filtrer les tickets urgents à traiter en premier.

3.3.4 Notifications et communication

Un système de notification automatique par email ou dans l'application permet :

- D'informer les utilisateurs de la progression de leur ticket ;
- De notifier les techniciens lorsqu'un nouveau ticket leur est affecté ;
- De relancer les tickets en attente depuis un certain temps.

3.3.5 Historique et traçabilité

L'ensemble des actions liées à un ticket (création, modification, clôture) est enregistré. Cela permet un audit complet de la prise en charge, utile en cas de réclamation ou pour mesurer la performance du service informatique.

3.4 Architecture technique

3.4.1 Infrastructure matérielle

- Serveur hébergeant l'application (Linux ou Windows Server) ;
- Postes utilisateurs : ordinateurs connectés au réseau TechnoPlus ;
- Aucun matériel supplémentaire requis, réutilisation de l'existant.

3.4.2 Environnement logiciel

- Langage : HTML / CSS / Python/ JavaScript ;
- Base de données : MySQL ;
- Frameworks possibles : Flask/ Flask-mail/;
- Hébergement local sur le serveur de l'entreprise ou en intranet.

3.4.3 Schéma de fonctionnement

[Utilisateur] → [Création ticket] → [Enregistrement base de données] → [Affectation technicien] → [Traitement et mise à jour] → [Clôture / Archivage]

3.5 Spécifications fonctionnelles

3.5.1 Interfaces utilisateur

- Tableau de bord utilisateur avec liste des tickets créés et leur statut ;
- Interface de création de ticket avec formulaire dynamique ;
- Interface technicien avec filtres par priorité, date, statut ;
- Page administrateur pour la gestion des utilisateurs, des droits, et des statistiques ;
- Module d'exportation des tickets au format CSV.

3.5.2 Fonctionnalités principales

- Création, édition et clôture de tickets ;
- Attribution des tickets à des intervenants spécifiques ;
- Messagerie interne pour chaque ticket (commentaires) ;
- Système de priorisation configurable ;
- Suivi par date de création, date limite, ou priorité.

3.6 Spécifications techniques

3.6.1 Contraintes techniques

- Compatibilité avec les principaux navigateurs web (Chrome, Firefox, Edge) ;
- Interface responsive adaptée aux écrans de bureau et tablettes ;

- Respect des normes d'ergonomie et d'accessibilité web ;
- Système multilingue possible si l'entreprise est internationale.

3.6.2 Sécurité et confidentialité

- Authentification sécurisée avec identifiants uniques ;
- Gestion des droits selon les rôles (utilisateur, technicien, administrateur) ;
- Journalisation des accès et des actions ;

Sauvegarde régulière de la base de données ;

- Protection contre les injections SQL et autres attaques courantes.

3.7 Avantages attendus

- Réduction du nombre de demandes non traitées ou oubliées ;
- Meilleure réactivité des techniciens grâce au tri automatisé ;
- Vision d'ensemble sur l'état du parc informatique via les tickets ;
- Valorisation du service informatique auprès des utilisateurs ;
- Possibilité de tirer des statistiques d'incidents pour orienter les investissements futurs.

3.8 Limites et pistes d'évolution

- Nécessité de formation des utilisateurs peu à l'aise avec les outils numériques;
- Dépendance à la qualité des données saisies dans les tickets ;
- Manque de liaison avec d'autres systèmes (ex : Active Directory, GLPI) ;
- Évolution possible vers un système de gestion de parc ou une intégration à un ERP.

3.9 Planning prévisionnel

Étape	Durée	Responsable
Analyse des besoins	1 semaine	Développeur
Maquettage et conception	1 semaine	Développeur
Développement du prototype	2 semaines	Développeur
Tests fonctionnels	1 semaine	Utilisateurs tests
Déploiement sur serveur	1 semaine	Administrateur réseau
Formation des utilisateurs	1 semaine	Chef de projet ou référent IT

3.10 Budget prévisionnel

Poste	Coût
Matériel	0 € (matériel existant)
Développement	0 € (réalisé en interne)
Hébergement	0 € (serveur local)
Maintenance	~300 € / an
Total estimé	300 € / an

3.11 Critères de réussite

- 100 % des tickets enregistrés dans l'outil (zéro demande hors système) ;
- Délai moyen de traitement < 48 heures ;
- Taux de satisfaction des utilisateurs ≥ 85 % ;
- Aucun ticket resté sans réponse plus de 72 heures ;
- Respect du calendrier de déploiement.

3.12 Glossaire

- **Ticket** : demande ou alerte créée pour signaler un incident informatique ;
- **Technicien** : personne chargée de traiter les tickets ;
- **Statut** : état d'un ticket (ex : ouvert, en cours, clôturé) ;

- **Priorité** : niveau d'urgence affecté à un ticket ;
- **Traçabilité** : capacité à suivre l'historique complet d'un ticket.

3.13 Étude de faisabilité

3.13.1 Faisabilité technique

Le projet repose sur des technologies accessibles et maîtrisées (PHP, MySQL), compatibles avec l'environnement informatique actuel. Aucun matériel supplémentaire n'est requis, et le serveur existant suffit pour héberger la solution.

3.13.2 Faisabilité économique

Le développement étant réalisé en interne, le coût global reste faible. L'hébergement et l'infrastructure sont déjà en place. Les seules dépenses concernent la maintenance et la sauvegarde.

3.13.3 Faisabilité organisationnelle

Le projet s'intègre dans les missions du service informatique. Il peut être piloté par un étudiant en BTS SIO sous la supervision d'un tuteur. Le déploiement est progressif, avec tests et formation à chaque étape.

3.13.4 Faisabilité temporelle

Le calendrier est réaliste pour une réalisation en autonomie. Les étapes sont clairement identifiées, avec des marges prévues pour les ajustements.

3.13.5 Conclusion générale

Le projet est faisable, peu coûteux, utile, et formateur. Il répond à un besoin réel de l'entreprise tout en offrant un cas concret de développement d'application métier.

3.14 Diagramme de cas d'utilisation

Acteurs principaux :

- Utilisateur : crée un ticket et suit son évolution ;
- Technicien : consulte, traite et clôture les tickets ;
- Administrateur : gère les comptes, consulte les statistiques.

Cas d'utilisation :

- Créer un ticket ;
- Modifier un ticket ;
- Répondre à un utilisateur ;
- Clôturer un ticket ;
- Attribuer un ticket ;
- Gérer les utilisateurs ;
- Exporter les rapports.

(Voir Annexe 9)

4. Bilan du projet

4.1 Fonctionnalités réalisées

À ce stade de développement, plusieurs fonctionnalités fondamentales ont été conçues, implémentées et testées avec succès dans le cadre de la gestion de tickets :

- **Interface de connexion et d'inscription :**

Un système de connexion sécurisé a été mis en place. Les mots de passe sont hachés (avec un algorithme de type SHA ou bcrypt), garantissant la sécurité des données sensibles des utilisateurs. L'inscription permet l'ajout d'un compte utilisateur avec des identifiants valides.

- **Création de tickets par les utilisateurs :**

Une interface intuitive permet aux utilisateurs de créer un ticket en spécifiant un objet, une description et une priorité.

- **Gestion des tickets par l'administrateur :**

L'administrateur dispose d'un espace dédié pour visualiser tous les tickets en cours. Il peut :

- Modifier le **statut** (ouvert, en cours, résolu, etc.) ;
- Changer la **priorité** du ticket ;

- Consulter l'**historique des tickets** (liste des tickets traités avec les actions associées).
- **Notification de résolution :**

Lorsqu'un ticket est marqué comme "résolu", l'utilisateur concerné est informé automatiquement de la mise à jour.
- **Gestion des descriptions :**

Chaque ticket peut être accompagné d'une description détaillée, permettant une meilleure compréhension du problème soumis.

4.2 Fonctionnalités non implémentées

Certaines fonctionnalités prévues ou imaginées n'ont pas encore été intégrées dans la version actuelle, mais sont clairement identifiées comme des pistes d'évolution possibles :

- **Gestion des rôles et des droits utilisateurs :**

Aucun système de différenciation des rôles (administrateur, support, utilisateur) n'est encore en place, ce qui limite la granularité des droits d'accès.
- **Gestion des utilisateurs :**

Il n'existe pas de panneau de gestion pour créer, modifier ou désactiver des comptes utilisateur via une interface dédiée.
- **Sauvegarde automatique de la base de données :**

Aucune procédure de sauvegarde régulière (automatique ou manuelle) n'a été mise en œuvre à ce jour.
- **Exportation des tickets au format CSV :**

Bien que cela faciliterait l'archivage ou les rapports, cette fonctionnalité n'est pas encore disponible.

- **Messagerie interne :**

Une communication directe entre utilisateur et administrateur (via chat ou messages intégrés) reste à implémenter.

- **Suivi des métadonnées des tickets :**

Le système ne conserve pas encore la date exacte de création, ni le nom de l'utilisateur ayant créé le ticket.

- **Système multilingue :**

L'interface ne prend en charge qu'une seule langue, sans option de traduction dynamique.

- **Interface responsive :**

L'affichage n'est pas encore optimisé pour les tablettes ou smartphones, ce qui limite l'utilisation mobile.

4.3 Contraintes rencontrées

Plusieurs contraintes techniques et humaines ont influencé le développement du projet :

- **Charge de travail individuelle :**

Le projet a été mené seul, ce qui a nécessité une gestion rigoureuse du temps et des priorités.

- **Temps limité :**

Temps limité pour développer l'ensemble des fonctionnalités prévues.

- **Découverte de certaines technologies :**

L'apprentissage de notions comme le hachage de mots de passe, la gestion des statuts dynamiques ou l'implémentation d'un système de notification a nécessité un temps d'adaptation.

- **Ergonomie à améliorer :**

Certaines interfaces sont fonctionnelles mais encore perfectibles en termes de design et d'expérience utilisateur.

4.4 Perspectives d'évolution

Le projet dispose d'un fort potentiel d'amélioration. Voici plusieurs pistes de développement à envisager pour une version future plus complète et robuste :

- Mise en place d'un **système de rôles** pour restreindre ou accorder certains accès selon le profil (utilisateur, technicien, administrateur).

- Ajout d'un **export CSV ou PDF** pour les tickets, avec filtres par statut, date ou utilisateur.
- Intégration d'un **système de messagerie interne** pour fluidifier la communication autour d'un ticket.
- Développement d'un **tableau de bord statistique** : nombre de tickets ouverts, temps moyen de résolution, taux de satisfaction, etc.
- Ajout d'une **interface responsive** compatible avec les smartphones.
- **Sauvegarde automatique** et planification des backups de la base de données.
- Ajout d'un **système de notifications en temps réel** (via WebSocket ou push).
- Intégration d'un **module multilingue** (français, anglais au minimum).
- Amélioration de l'expérience utilisateur par l'ajout d'**animations, filtres dynamiques, tri des tickets** et mode sombre.

5. Conclusion

Ce projet, réalisé dans un cadre pédagogique, a permis de concevoir une application fonctionnelle répondant à un besoin concret de gestion et de suivi de tickets dans un environnement de support ou de maintenance informatique. L'application offre aux utilisateurs la possibilité de créer des tickets détaillés, tandis que l'administrateur dispose d'une interface de gestion permettant de suivre et traiter efficacement les demandes.

Techniquement, le projet s'est appuyé sur des technologies courantes et maîtrisables, mettant l'accent sur la sécurité (notamment via le hachage des mots de passe), la clarté des interfaces, et la logique métier autour de la gestion des tickets (statuts, priorités, notifications). Un effort particulier a été porté sur la séparation des rôles (même si les droits ne sont pas encore différenciés) et la traçabilité des actions via l'historique administrateur.

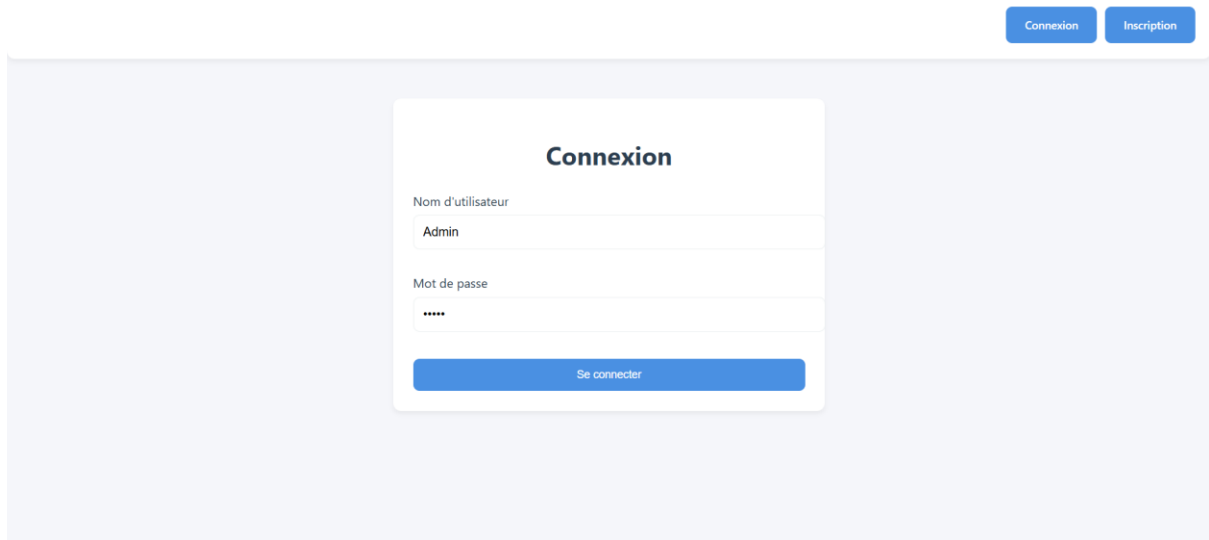
Ce projet a également été l'occasion d'approfondir des compétences fondamentales du : développement web ou applicatif, gestion de bases de données, sécurisation des accès, mise en place de formulaires dynamiques, et gestion d'un cycle de traitement des données cohérent.

Bien que certaines fonctionnalités avancées restent à implémenter (telles que la gestion des utilisateurs, l'exportation de données ou un système de messagerie interne), ce projet constitue une base solide et évolutive pour une application métier complète. Il ouvre également la voie à des perspectives intéressantes telles que l'ajout d'un tableau de bord, l'adaptation à une interface responsive, ou encore la mise en place d'un système de rôles et de permissions.

En définitive, cette réalisation s'inscrit dans une démarche professionnelle complète, mêlant analyse, conception, développement et tests, tout en illustrant les problématiques réelles auxquelles un développeur peut être confronté dans le monde de l'entreprise.

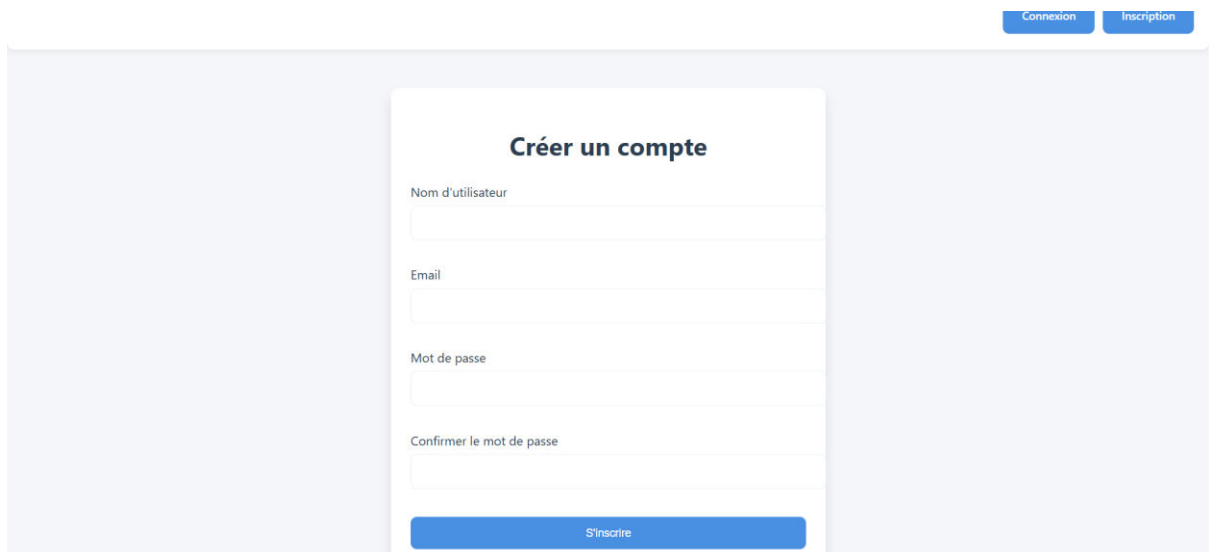
6. Annexes

1 Interface de connexion de l'application



The screenshot shows a web application interface for logging in. At the top right, there are two blue buttons: "Connexion" and "Inscription". The main content area has a light gray background. In the center, there is a white card titled "Connexion". Inside the card, there are two input fields: "Nom d'utilisateur" with the text "Admin" and "Mot de passe" with five dots. Below these fields is a blue button labeled "Se connecter".

2 Interface d'inscription de l'application



The screenshot shows a web application interface for creating a new account. At the top right, there are two blue buttons: "Connexion" and "Inscription". The main content area has a light gray background. In the center, there is a white card titled "Créer un compte". Inside the card, there are four input fields: "Nom d'utilisateur", "Email", "Mot de passe", and "Confirmer le mot de passe". Below these fields is a blue button labeled "S'inscrire".

3 Interface administrateur

The screenshot displays the administrator interface with a top navigation bar containing buttons for 'Tickets', 'Calendrier', 'Historique', 'Nouveau Ticket', and 'Déconnexion'. The main content area is titled 'Liste des Tickets' and features two ticket entries. The first entry, 'Serveur en état CRITIQUE', has a status of 'Fait vite', a priority of 'Basse', and is marked 'En attente'. It includes buttons for 'Modifier', 'Supprimer', and 'Résoudre'. The second entry, 'Serveur en panne', has a status of 'Vérifie les problèmes du serveur !!!', a priority of 'Moyenne', and is also marked 'En attente', with similar 'Modifier', 'Supprimer', and 'Résoudre' buttons.

4 Gestion du ticket par l'administrateur

The screenshot shows the 'Modifier le Ticket' form. It includes fields for 'Titre' (containing 'Serveur en panne') and 'Description' (containing 'Vérifie les problèmes du serveur !!!'). Below these are dropdown menus for 'Priorité' (set to 'Moyenne') and 'Statut'. The 'Statut' dropdown is open, showing options: 'À venir', 'À venir', 'En cours' (which is highlighted), and 'Fait'.

5 Calendrier des tickets

Tickets

Calendrier

Historique

Nouveau Ticket

Déconnexion

Ajouter un Ticket

Titre

PC non fonctionnnel

Description

Mon ordinateur a arrêté de fonctionner pendant que je travaillait.

Créer le Ticket

6 Création de tickets

Calendrier des Tickets

Retour

<

>

Aujourd'hui

mai 2025

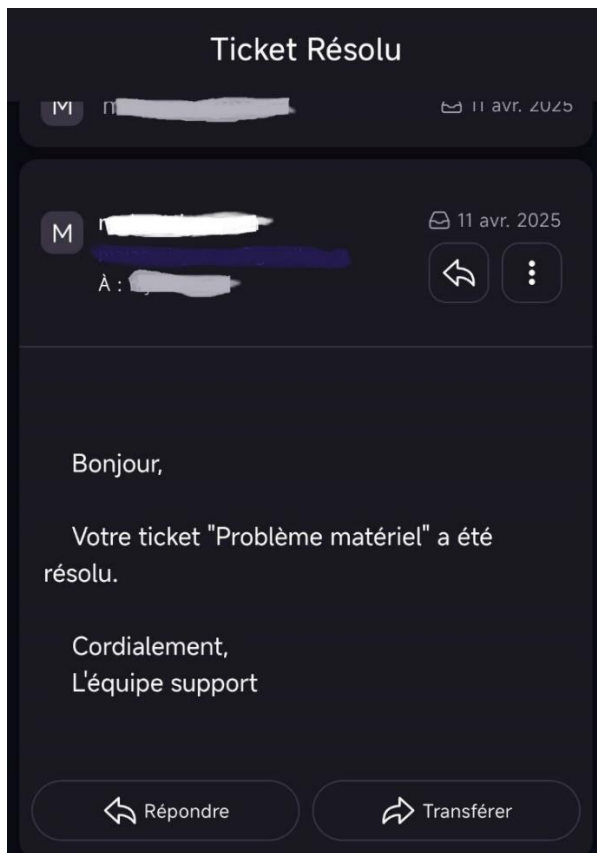
Mois

Semaine

Jour

lun.	mar.	mer.	jeu.	ven.	sam.	dim.
28	29	30	1	2	3	4 17:36 Problème matériel 17:36 Internet ne fonctionne 17:37 Vérifier le serveur 17:37 Serveur en panne
5	6	7	8	9	10	11

7.1 Capture d'écran de la notification de la résolution du ticket depuis un téléphone portable



7.2 Extrait du code source, fonction 'send_resolution_email' qui permet l'envoi de la notification

```
def send_resolution_email(user_email, ticket_titre):
    # Création d'un objet Message pour l'e-mail, avec le sujet, l'expéditeur
    et le destinataire
    msg = Message('Ticket Résolu',
                  sender='your-email@gmail.com',
                  recipients=[user_email])

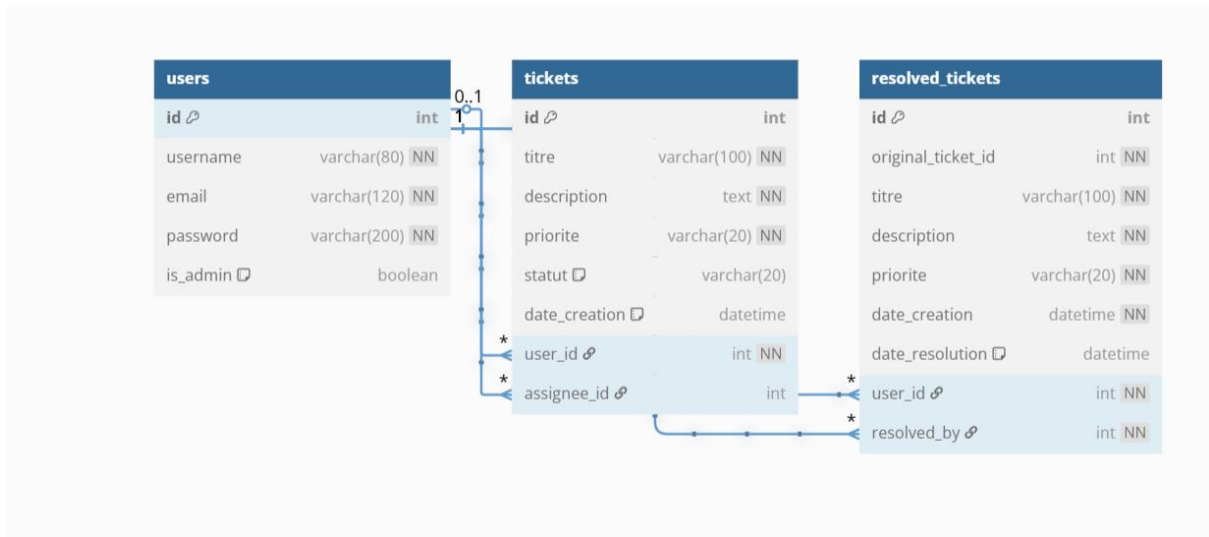
    # Corps du message envoyé à l'utilisateur
    msg.body = f"""
    Bonjour,

    Votre ticket "{ticket_titre}" a été résolu.

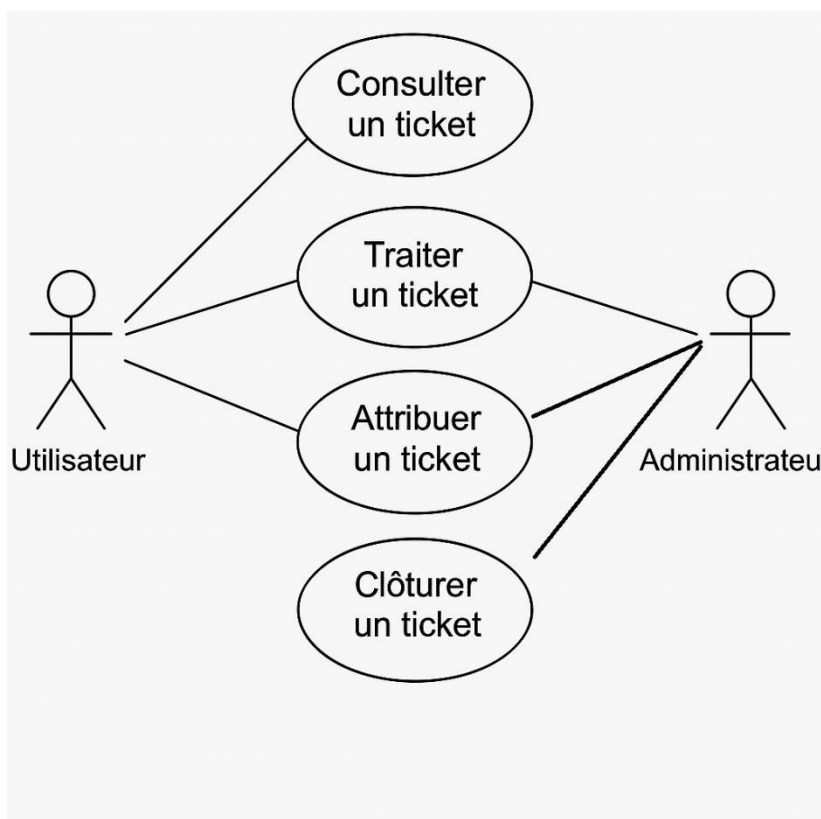
    Cordialement,
    L'équipe support
    """

    # Envoi de l'e-mail via le serveur configuré dans Flask-Mail
    mail.send(msg)
```

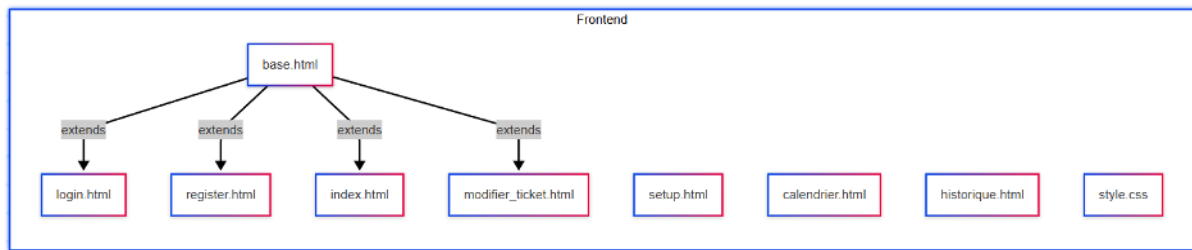
8 Structure relationnel de la base de données (depuis <https://dbdiagram.io>)



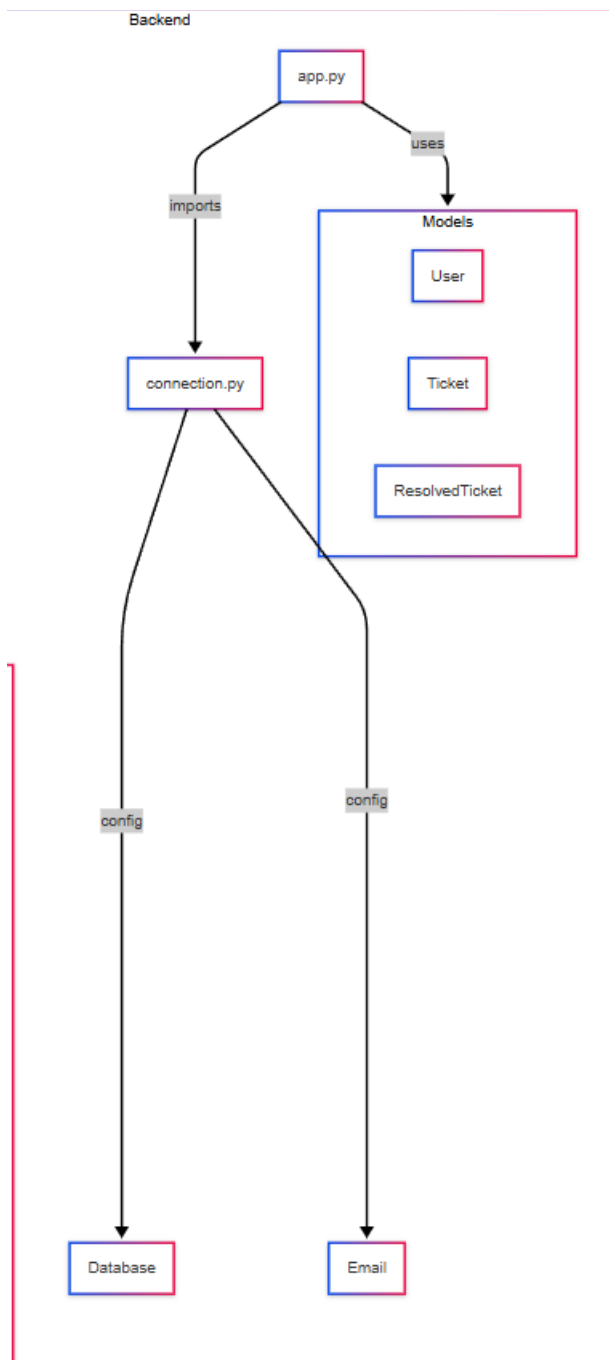
9 Diagramme de cas d'utilisations



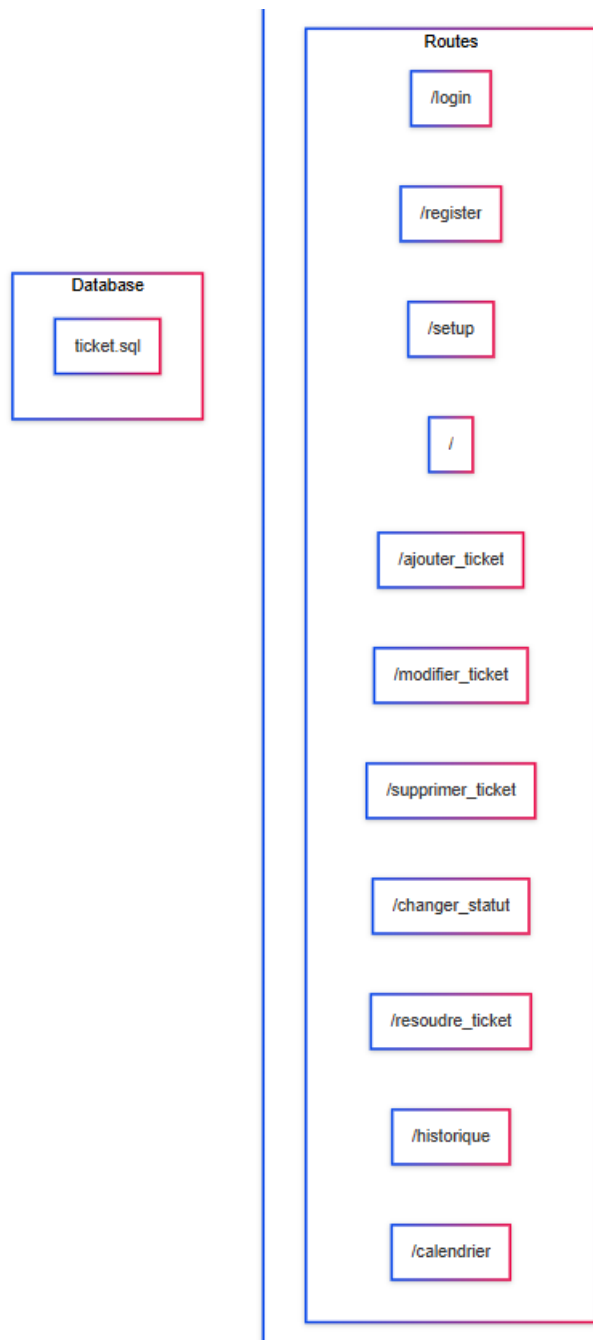
10 Structure frontend de l'application



11.1 Structure back-end de l'application



11.2 Structure des autres fichiers du back + base de données



12. Extrait de code, fonction 'déterminer_priorite' qui détermine automatiquement la priorité des tickets.

```
def determiner_priorite(description):
    # Liste des mots-clés qui indiquent une priorité haute
    # Ces mots sont généralement associés à des situations critiques ou
    urgentes
    mots_haute_priorite = ['urgent', 'critique', 'panne', 'bloqué',
                           'impossible', 'erreur grave',
                           'ne fonctionne plus', 'immédiat', 'emergency']

    # Liste des mots-clés qui indiquent une priorité moyenne
    # Ces mots sont associés à des problèmes importants mais non critiques
    mots_moyenne_priorite = ['problème', 'bug', 'erreur', 'dysfonctionnement',
                              'lent',
                              'ralentissement', 'mise à jour']

    # Conversion de la description en minuscules pour faciliter la recherche
    des mots-clés
    description = description.lower()

    # Vérification de la présence de mots-clés de haute priorité
    for mot in mots_haute_priorite:
        if mot in description:
            return 'Haute'

    # Si aucun mot de haute priorité n'est trouvé, alors on cherche les mots
    de priorité moyenne
    for mot in mots_moyenne_priorite:
        if mot in description:
            return 'Moyenne'

    # Si aucun mot-clé n'est trouvé, alors la priorité est considérée comme
    basse par défaut
    return 'Basse'
```

13. Extrait de code, fonction 'register' qui enregistre les nouveaux utilisateurs et hachent leurs mots de passe grâce à la fonction importée de Werkzeug.

```
from werkzeug.security import generate_password_hash, check_password_hash

@app.route('/register', methods=['GET', 'POST'])
def register():
    // ...Code existant ...

    # Create new user
    new_user = User(
        username=username,
        email=email,
        password=generate_password_hash(password), # Ici le mot de passe est
haché
        is_admin=False
    ) // ... Code existant ci-dessous ...
```