09 OCTOBRE 2024

COMPTE RENDU FINAL

EQUIPE 4

AP5 : Service Web et Site

« Galaxy Swiss Bourdin (GSB) »

Projet réalisé par

Dylan AUBRAS
Arnaud BIDEL
Henri DAMOUR
Dylan GILBOIRE
Rayan RABOZIVELO

Projet encadré par

Gilles Emmanuel ALPHONSINE
Nicolas DEFAY

Lycée Marguerite JAUZELON

Table des matières

I. INTRODUCTION 4
II. BESOINS ET OBJECTIF DU PROET 5
1. CONTEXTE 5
2. OBJECTIFS ET CONTRAINTES 6
a. Objectifs Techniques 6
b. Contraintes Cybersécurité 6
c. Contraintes Juridiques 6
d. Contraintes de Planification et de Collaboration 6
III. GESTION DE PROJET 7
1. L'EQUIPE 7
a. SLAM 7
b. SISR 7
2. PLANIFICATION ET OUTILS UTILISES 8
a. Le cahier des charges 8
b. Diagramme de Gantt via Monday.com 8
c. Espace Collaboratif 9
3. RESSOURCES ET ESTIMATION DES COÛTS 10
a. Ressources Humaines et Planification des Heures 10
b. Estimation des Coûts 10
a. Concertations Inter-Spécialités 11
b. Répartition des Tâches et Intégration 11
c. Gestion des Conflits et Adaptation 11
IV. PARTIE JURIDIQUE 12
a. Qualification juridique des informations composant la base de données 12
b. Base juridique pour le recueil de ces informations 12
c. Principales règles applicables au traitement de ces informations 12
d. Protection de la base de données par le droit d'auteur. 13
V. PARTIE DEVELOPPEMENT (SLAM) 15
1. ARCHITECTURE APPLICATIVE 15
a. Schéma de Principe 15
b. Principe Général de l'Architecture MVC 17
2. BASE DE DONNÉE 18
a. Conception de la Base de Données Visiteurs et Comptables 18
b. Justification des Choix de Conception des Modèles de Données 18

3. CYBERSÉCURITÉ 19
4. TESTS ET VÉRIFICATION DE L'INTÉGRITÉ 20
VI. PARTIE RESEAUX (SISR) 21
1. MISE EN PLACE DE L'INFRASTRUCTURE 21
a. Schéma de l'architecture réseau 21
2. SÉCURISATION ET SAUVEGARDE 22
a. Objectif 22
b. Pré-requis 22
c. Tests 22
d. Rapport de Test 22
3. PREUVE DE CONCEPT (POC) 23
a. Objectif 23
b. Architecture du POC 23
c. Solution Proposée : Utilisation de SSL/TLS 23
d. Étapes de Mise en Œuvre du POC 23
e. Résultats Attendus 24
f. Conclusion 24
VII. BILAN DU PROJET 25
1. BILAN GÉNÉRALE (SLAM/SISR) 25
2. JUSTIFICATION ÉCARTS PREVISIONNEL/RÉALISÉ 26

Table des figures

Figure 1 - Diagramme de Gantt 8
Figure 2 - Aperçu Kanboard 9
Figure 3 - Schéma de l'architecture applicative 15
Figure 4 - Interaction MVC 17
Figure 5 - BD avant modification 18
Figure 6 - BD après modification 18
Figure 7 - Schéma de l'architecture réseau 21

I. INTRODUCTION

Dans le cadre de notre deuxième année de BTS Services Informatiques aux Organisations (SIO), spécialité Solutions Logicielles et Applications Métiers (SLAM) et Solutions d'Infrastructure, Systèmes et Réseaux (SISR), nous avons été amenés à travailler sur un projet de huit semaines. Ce projet, proposé dans le cadre du module AP5, nous permet de mettre en pratique nos compétences acquises tout au long de notre formation, en répondant à un cahier des charges défini par une entreprise, tout en travaillant dans des conditions proches du milieu professionnel.

Notre équipe, l'équipe 4, composée de trois étudiants en SLAM et deux en SISR, a saisi cette opportunité pour contribuer au développement d'une application concrète et innovante, tout en explorant la collaboration entre les deux spécialités.

Ce travail de huit semaines, réalisé en étroite collaboration entre les membres de l'équipe, a représenté pour nous une occasion d'approfondir nos compétences techniques, tant en développement qu'en administration de systèmes, et de nous préparer aux défis du milieu professionnel.

II. BESOINS ET OBJECTIF DU PROET

1. CONTEXTE

Le laboratoire Galaxy Swiss Bourdin (GSB) est né de la fusion de deux géants pharmaceutiques : Galaxy, spécialisé dans le traitement des maladies virales comme le SIDA et les hépatites, et Swiss Bourdin, un conglomérat européen actif dans la production de médicaments plus conventionnels. Cette fusion, en 2009, a permis de créer un acteur majeur de l'industrie pharmaceutique, avec son siège social à Philadelphie (États-Unis) et son siège administratif à Paris. Cette réorganisation a permis de renforcer l'activité de visite médicale, un secteur clé pour GSB afin de maintenir une relation de proximité avec les prescripteurs comme les médecins et les pharmaciens.

GSB emploie actuellement 480 visiteurs médicaux en France, ainsi que 60 dans les départements et territoires d'outre-mer. Ces visiteurs sont chargés de promouvoir les nouveaux produits du laboratoire directement auprès des prescripteurs, engendrant ainsi des frais de déplacement importants. Pour optimiser la gestion de ces frais, GSB a décidé de développer une application Web dédiée, qui permettra une saisie efficace des frais par les visiteurs médicaux ainsi qu'une validation rigoureuse par le service comptable.

C'est dans ce contexte que notre projet s'inscrit, avec pour objectif de concevoir et développer une solution de gestion des frais de déplacement centralisée et sécurisée. Ce développement est confié à une équipe composée de spécialistes du développement applicatif (SLAM) et de l'administration réseau (SISR), travaillant conjointement pour mettre en place une solution complète, de l'interface utilisateur à l'infrastructure d'hébergement sécurisée.

2. OBJECTIFS ET CONTRAINTES

a. Objectifs Techniques

L'objectif principal du projet est de développer une application Web permettant aux visiteurs médicaux de saisir leurs frais de déplacement et de soumettre ces informations pour validation au service comptable. L'application est basée sur une architecture MVC (Modèle-Vue-Contrôleur) pour assurer une séparation claire entre les différentes couches de l'application et faciliter la maintenance future du système. De plus, la solution doit être conforme aux exigences de sécurité et permettre un accès via des noms de domaines sécurisés, tels que www.swiss-galaxyn-france.fr, pour améliorer l'accessibilité tout en garantissant la sécurité des échanges.

b. Contraintes Cybersécurité

La cybersécurité est un aspect fondamental du projet, étant donné la nature sensible des données traitées. L'application doit être hébergée sur un serveur Debian 12 configuré pour utiliser des protocoles sécurisés, tels qu'HTTPS pour Apache2 et FTPS pour la gestion des transferts de fichiers. Des mesures additionnelles, comme l'installation d'un pare-feu et la configuration de permissions strictes pour les comptes FTP, sont mises en place pour minimiser les risques de compromission. Le serveur de base de données, hébergé sur une autre machine, doit être sécurisé pour garantir la confidentialité des informations échangées entre le serveur Web et la base de données.

c. Contraintes Juridiques

Le projet comporte également des contraintes juridiques importantes liées à la gestion des données personnelles des visiteurs médicaux et des comptables. Le traitement de ces données est encadré par le Règlement Général sur la Protection des Données (RGPD). Ainsi, il est nécessaire de qualifier juridiquement les informations traitées, de s'assurer que la collecte de ces données est basée sur un consentement préalable, et de garantir que les utilisateurs sont informés de la finalité de la collecte et de leur droit d'accès et de rectification des données.

De plus, nous devons déterminer si la base de données peut être protégée par le droit d'auteur ou si elle bénéficie d'un autre régime de protection, comme celui des bases de données sui generis. Ces aspects juridiques influent sur la conception du système, en imposant des mesures spécifiques pour garantir la conformité de la gestion des données.

d. Contraintes de Planification et de Collaboration

Le projet doit être mené en respectant des délais précis, avec une remise des travaux fixée au 10 octobre. L'organisation du travail est essentielle pour respecter ces échéances, impliquant l'utilisation d'outils de gestion pour le suivi des tâches et le diagramme de Gantt pour la planification globale. Le projet est également rythmé par des moments de concertation entre les spécialités SLAM et SISR pour assurer une bonne coordination et une intégration harmonieuse des différentes composantes du système.

III. GESTION DE PROJET

1. L'EQUIPE

Notre équipe de projet est composée de membres aux compétences complémentaires, regroupant des étudiants en développement applicatif (SLAM) et en administration des systèmes et réseaux (SISR). Tous les membres de l'équipe suivent le cursus BTS SIO depuis l'année dernière, ce qui nous a permis de construire une bonne synergie et une connaissance approfondie de nos compétences respectives.

Voici les membres de notre équipe :

a. SLAM

- **BIDEL** Arnaud : Arnaud est particulièrement compétent en **gestion de projet** et en **SQL**. Son expertise dans la gestion des bases de données a permis de structurer efficacement les fiches de frais, garantissant leur bonne organisation et un accès rapide. En tant que référent sur les aspects de planification, il a également joué un rôle essentiel dans la coordination et le suivi de l'avancement des différentes tâches, assurant une bonne communication entre les membres de l'équipe.
- **DAMOUR** Henri : Henri est le "couteau suisse" de l'équipe. Sa polyvalence lui permet d'intervenir sur plusieurs aspects du projet, allant du développement front-end au back-end, ainsi qu'à la documentation technique. Grâce à cette polyvalence, Henri a su combler les manques dans différents domaines et a été un véritable atout lors de la résolution rapide des problèmes imprévus. Sa capacité à s'adapter rapidement à des tâches variées a considérablement amélioré la productivité du groupe.
- **GILBOIRE** Dylan : Dylan est spécialisé en **PHP**, et se distingue par sa capacité à développer des fonctionnalités avancées pour l'application de gestion des frais. Son aisance avec PHP a permis d'implémenter efficacement les interactions entre la base de données et l'interface utilisateur, garantissant ainsi une bonne performance du système. Dylan a également travaillé sur des aspects de sécurité, en s'assurant que les formulaires utilisateurs respectaient les normes de sécurisation des données.

b. SISR

- **AUBRAS** Dylan : Dylan est particulièrement doué pour la **gestion des serveurs** et l'**administration réseau**. Il a joué un rôle clé dans la mise en place du serveur Debian 12, en assurant la configuration des services tels qu'Apache2, FTPS, et en garantissant une bonne disponibilité du site web. Sa connaissance approfondie des systèmes serveurs a également été déterminante pour la gestion des opérations de sauvegarde et de restauration.
- **RABOZIVELO** Rayan : Rayan a apporté son expertise en sécurité des systèmes d'information. Il s'est chargé de l'implémentation des mesures de sécurité nécessaires pour protéger les données et en sécurisant les transferts via des protocoles tels que HTTPS. Son travail a assuré que l'application était conforme aux exigences de sécurité, notamment pour les échanges entre le serveur web et la base de données. Sa rigueur et sa vigilance ont été cruciales pour prévenir les vulnérabilités et sécuriser l'ensemble de l'infrastructure.

2. PLANIFICATION ET OUTILS UTILISES

La gestion efficace du projet a été essentielle pour assurer l'accomplissement des tâches dans les délais impartis et garantir la qualité du produit final. La planification et le suivi des travaux se sont appuyés sur plusieurs outils dédiés à la gestion et à la collaboration.

a. Le cahier des charges

Le **cahier des charges** a défini les exigences du projet, incluant le contexte de GSB, les objectifs techniques, les contraintes et les livrables attendus. Il a servi de guide structurant pour l'équipe, garantissant la cohérence des développements et la répartition efficace des tâches entre les spécialités SLAM et SISR, tout en assurant la faisabilité du projet dans les délais impartis.

b. Diagramme de Gantt via Monday.com

Pour la planification détaillée du projet, nous avons utilisé un diagramme de Gantt, créé à l'aide de la plateforme **Monday.com**. Monday.com est une plateforme de gestion de projet qui nous a permis de structurer toutes les tâches nécessaires à la réussite du projet, de leur attribuer des échéances et de visualiser l'ensemble du planning de manière claire et conviviale. Les fonctionnalités de Monday.com, comme les mises à jour automatiques et les notifications, ont contribué à garder toute l'équipe informée des évolutions en temps réel.

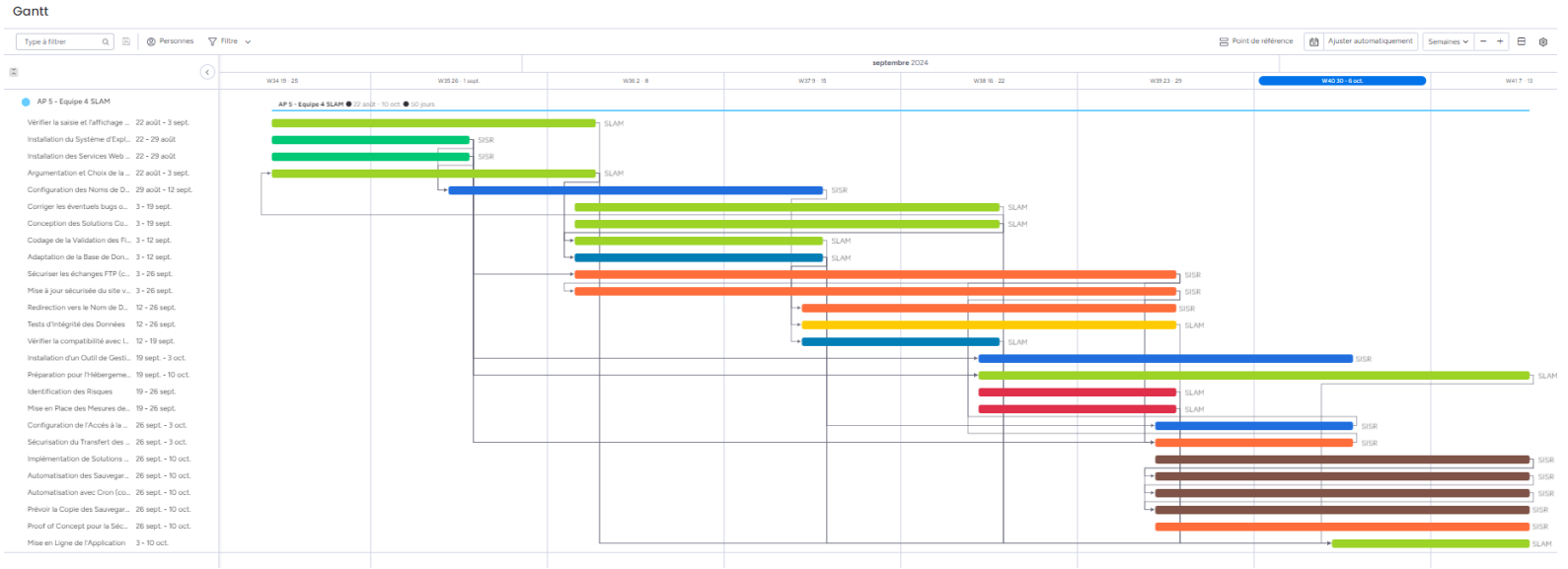

Figure 1 - Diagramme de Gantt

c. Espace Collaboratif

Pour la communication interne et le partage de documents, nous avons principalement utilisé :

- **Discord** : Pour les discussions en temps réel, les appels vocaux, et la planification des réunions. Discord nous a permis de maintenir une communication fluide entre tous les membres de l'équipe, en particulier lors de la résolution de problèmes complexes ou de concertations inter-spécialités.
- **Moodle** : Utilisé comme espace de partage des documents liés au projet, tels que le cahier des charges, les versions du projet, les livrables, et les guides de développement. Moodle a également permis de centraliser toutes les informations de référence utiles.
- **Kanboard** : Un outil de gestion des tâches qui nous a permis de visualiser et de suivre l'avancement des différentes étapes du projet. Grâce à son approche "Kanban", nous avons pu classer les tâches en fonction de leur statut ("En attente", "Prêt", "En cours", "Terminé"), ce qui a facilité la gestion des priorités et la répartition des responsabilités au sein de l'équipe.

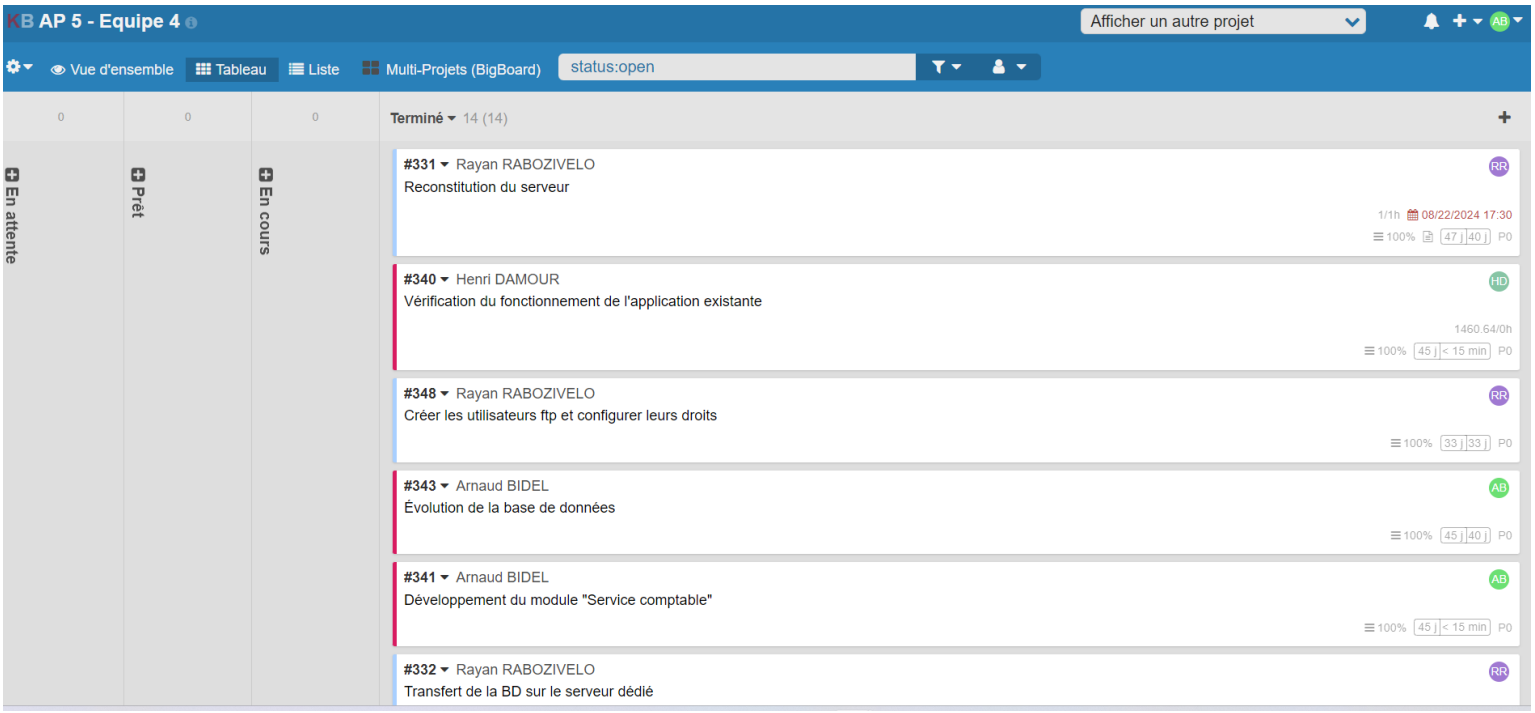

Figure 2 - Aperçu Kanboard

3. RESSOURCES ET ESTIMATION DES COÛTS

a. Ressources Humaines et Planification des Heures

Le projet s'est déroulé sur une période de **8 semaines**, du **22 août 2024** au **10 octobre 2024**. Pour la gestion des ressources humaines, nous avons réparti les tâches en fonction des compétences de chaque membre de l'équipe, assurant ainsi une complémentarité entre les spécialités SLAM (développement applicatif) et SISR (administration des systèmes et réseaux).

b. Estimation des Coûts

L'estimation des coûts a été réalisée afin de simuler un environnement professionnel réel, en prenant en compte à la fois les heures de cours et les heures supplémentaires réalisées en dehors des cours. Chaque membre de l'équipe est estimé à un salaire de **1600€ net par mois**, soit un taux journalier de **80€** pour une base de **20 jours ouvrés** par mois, ce qui équivaut à **20€/heure** (80€ pour 4 heures).

Le projet s'est déroulé sur une période de **8 semaines**, avec **4 heures de cours par semaine** pendant **7 semaines**. En supplément, des heures de travail ont été réalisées en dehors des cours : **15 heures supplémentaires** pour l'équipe SLAM et **8 heures supplémentaires** pour l'équipe SISR. Voici le détail de l'estimation des coûts pour chaque membre de l'équipe.

1. Heures de Cours : 4 Heures par Semaine Pendant 7 Semaines

- **Nombre d'heures de cours par membre :** 4 heures/semaine * 7 semaines = 28 heures
- **Coût par heure :** 20€/heure
- **Coût pour 28 heures de cours par membre :** 28 heures * 20€/heure = 560€

2. Heures Supplémentaires

- Équipe SLAM (3 membres) : 15 heures supplémentaires par membre :
  - **Coût pour 15 heures supplémentaires par membre :** 15 heures * 20€/heure = 300€
  - **Total pour chaque membre SLAM :** 560€ (cours) + 300€ (supplémentaires) = 860€
- Équipe SISR (2 membres) : 8 heures supplémentaires par membre :
  - **Coût pour 8 heures supplémentaires par membre :** 8 heures * 20€/heure = 160€
  - **Total pour chaque membre SISR :** 560€ (cours) + 160€ (supplémentaires) = 720€

SLAM	SISR
BIDEL Arnaud : **860€** **DAMOUR** Henri : **860€** **GILBOIRE** Dylan : **860€**	**AUBRAS** Dylan : **720€** **RABOZIVELO** Rayan : **720€**

3. Coût Total des Ressources Humaines

- Total équipe SLAM (3 membres) : 860€ * 3 = 2 580€
- Total équipe SISR (2 membres) : 720€ * 2 = 1 440€
- Coût total des ressources humaines : 2 580€ + 1 440€ = 4 020€

4. MOMENTS DE COLLABORATION SISR/SLAM

a. Concertations Inter-Spécialités

La collaboration entre les spécialités SLAM et SISR a été essentielle au succès du projet. Nous avons mis en place des points de concertation réguliers afin de garantir une bonne synchronisation entre les différentes tâches et de nous assurer que l'ensemble des éléments techniques s'imbriquaient correctement.

- **Réunions Hebdomadaires** : Des réunions hebdomadaires étaient organisées sur Discord chaque semaine pour évaluer l'avancement des travaux, ajuster la planification si nécessaire, et discuter des points critiques. Ces réunions ont été essentielles pour aligner les équipes sur les objectifs et résoudre les problèmes techniques de manière collective. Les comptes rendus hebdomadaires ont été rédigés afin d'assurer une traçabilité des décisions et de suivre les progrès du projet.
- **Collaborations Pratiques** : Lors des phases de développement nécessitant des ajustements de l'infrastructure, les spécialistes SLAM et SISR ont travaillé en étroite collaboration. Par exemple, la mise en place des fonctionnalités de sécurité pour l'application, telles que l'intégration d'une connexion HTTPS, a nécessité une forte coordination entre les équipes.

b. Répartition des Tâches et Intégration

Développement Applicatif (SLAM) : L'équipe SLAM a pris en charge la conception et le développement des fonctionnalités de l'application. Cela inclut la saisie et la consultation des fiches de frais par les visiteurs médicaux ainsi que le module de validation des frais par le service comptable. Les développeurs ont également travaillé sur l'amélioration de l'interface utilisateur pour garantir une expérience utilisateur fluide.

Infrastructure Réseau (SISR) : L'équipe SISR s'est concentrée sur la mise en place de l'infrastructure technique nécessaire à l'hébergement de l'application. Cela a impliqué la configuration d'un serveur Debian 12, la mise en place des services requis (Apache2 pour le serveur Web, FTPS pour les transferts sécurisés de fichiers), ainsi que des mesures de sécurité telles que les pare-feu et les règles d'accès.

c. Gestion des Conflits et Adaptation

Au cours du projet, nous avons dû surmonter certains défis, notamment des conflits liés aux exigences de sécurité et à l'intégration des modules développés. Par exemple, les mesures de sécurisation des transferts de données implémentées par l'équipe SISR ont nécessité des adaptations dans le code côté SLAM pour assurer la compatibilité. Ces défis ont été relevés grâce à une communication efficace sur Discord et à la flexibilité de chacun des membres pour s'adapter aux besoins des autres.

IV. PARTIE JURIDIQUE

a. Qualification juridique des informations composant la base de données

Les informations composant la base de données peuvent être qualifiées juridiquement de **données à caractère personnel** au sens du **Règlement Général sur la Protection des Données (RGPD)** si elles concernent des personnes identifiables (visiteurs médicaux, comptables, etc.). Les données telles que le nom, prénom, adresse, identifiants de connexion, ainsi que les détails des frais (qui peuvent révéler des habitudes ou des informations financières personnelles) sont des données personnelles.

b. Base juridique pour le recueil de ces informations

Le recueil et le traitement des données à caractère personnel doivent s'appuyer sur l'une des bases légales définies par le RGPD. Les bases légales les plus pertinentes pour le recueil des informations de la base de données GSB seraient :

- **L'exécution d'un contrat :** Si le recueil de ces informations est nécessaire pour l'exécution d'un contrat auquel la personne concernée est partie (par exemple, un contrat de travail ou de prestation).
- **L'obligation légale :** Si le traitement est requis pour se conformer à une obligation légale (par exemple, des obligations fiscales ou comptables).
- **L'intérêt légitime :** Dans certains cas, l'intérêt légitime de l'organisation (GSB) pourrait justifier le traitement des données, à condition que cet intérêt ne soit pas supplanté par les droits et libertés fondamentaux des personnes concernées.

Le consentement préalable n'est pas nécessaire si le traitement repose sur l'une des bases légales mentionnées ci-dessus (exécution d'un contrat ou obligation légale). Cependant, pour le traitement de certaines données sensibles ou si aucun autre fondement n'est approprié, le consentement explicite des utilisateurs pourrait être requis.

c. Principales règles applicables au traitement de ces informations

Les règles applicables au traitement des données personnelles sont principalement régies par le RGPD. Les principales obligations sont :

- **Transparence et information :** Les personnes concernées doivent être informées de la finalité du traitement, de la durée de conservation des données, de leurs droits (accès, rectification, suppression, opposition, portabilité) et de l'identité du responsable de traitement.
- **Limitation de la collecte et de la finalité :** Les données doivent être collectées de manière pertinente et limitée aux finalités définies.
- **Sécurité des données :** Les mesures techniques et organisationnelles appropriées doivent être mises en place pour garantir la confidentialité, l'intégrité et la disponibilité des données.
- **Minimisation des données :** Seules les données nécessaires au regard des finalités pour lesquelles elles sont traitées doivent être collectées.
- **Durée de conservation limitée :** Les données ne doivent pas être conservées plus longtemps que nécessaire pour les finalités pour lesquelles elles sont traitées.

d. Protection de la base de données par le droit d'auteur.

En général, le droit d'auteur ne protège pas les bases de données elles-mêmes, mais plutôt la structure, la sélection et la présentation du contenu, à condition que celles-ci démontrent une originalité suffisante. Notre base de données, GSB, pourrait être protégée par le droit d'auteur si nous parvenons à démontrer que sa structure et son organisation sont suffisamment originales (par exemple, par une manière innovante de lier ou de représenter les données).

Cependant, les données elles-mêmes (comme les noms des visiteurs médicaux, les frais, etc.) ne sont pas couvertes par le droit d'auteur, car ce sont des faits bruts et non des créations intellectuelles.

Originalité de la Base de Donnée

La protection par le droit d'auteur s'applique à la structure et à l'organisation des éléments d'une base de données, pour autant que cette structure soit jugée suffisamment originale. En analysant notre base de données, nous estimons qu'elle présente certains aspects qui contribuent à cette originalité :

Complexité des Relations :

Notre base de données est organisée avec plusieurs entités (tables) qui représentent différents acteurs dans la gestion des frais (par exemple, visiteur, comptable, fraisforfait, lignefraisforfait, fichefrais, etc.). Ces entités sont reliées par des relations bien définies.

La création de tables telles que ValidationFicheFrais pour tracer les validations effectuées par les comptables est un exemple d'une approche originale permettant de suivre l'état de validation et de gérer les autorisations. Ce niveau de structuration montre une certaine complexité dans les relations mises en place.

Différenciation des Utilisateurs :

Nous avons opté pour la Solution 2, qui consiste à avoir des tables séparées pour les Visiteurs et les Comptables. Cela facilite la gestion des rôles et des permissions, offrant ainsi une plus grande flexibilité et une gestion granulaire des utilisateurs. Cette approche démontre une volonté d'organiser la base de données de manière méthodique, tout en répondant aux exigences de sécurité et de différenciation des rôles.

Conception orientée rôles et actions :

L'organisation des fiches de frais, la gestion de la validation par les comptables, ainsi que la manière de lier ces entités pour suivre les validations (par exemple, la table ValidationFicheFrais), témoignent d'une organisation logique et raisonnée. Cela reflète une volonté d'optimiser la gestion des actions spécifiques (comme la validation des frais) par des rôles distincts, ce qui est un aspect d'originalité en termes de structuration.

Estimation de l'Originalité

Pour qu'une base de données soit protégée par le droit d'auteur, il faut que la structure de cette base soit le fruit d'un effort intellectuel personnel démontrant un choix créatif. En ce qui concerne notre base de données :

- La manière dont les tables sont créées et les relations établies (ValidationFicheFrais, Comptable, FicheFrais, etc.) montre une réflexion approfondie sur les processus métiers liés à la gestion des frais et des autorisations.
- La séparation des rôles (Visiteurs et Comptables) et l'utilisation de tables distinctes pour faciliter la gestion des droits démontrent notre volonté de construire une base de données flexible et évolutive.

Ces éléments tendent à montrer une certaine originalité dans la conception, notamment en raison de la complexité et de la logique des relations établies entre les différentes tables. Cela pourrait être suffisant pour revendiquer une protection par le droit d'auteur de la structure de notre base de données.

Cependant, il est important de noter que les données elles-mêmes (comme les noms des visiteurs ou les informations de frais) ne sont pas protégées par le droit d'auteur, car il s'agit de faits bruts. La protection porterait sur la manière d'organiser ces données (la structure, la sélection et la présentation des éléments), tant que cette organisation est suffisamment originale.

Conclusion

En conclusion, nous pensons que notre base de données possède une structure suffisamment originale pour être protégée par le droit d'auteur, notamment en raison de la manière réfléchie dont les tables et les relations ont été conçues pour répondre aux besoins spécifiques des utilisateurs (visiteurs et comptables). Cette organisation témoigne d'un effort intellectuel de structuration qui dépasse une simple compilation de données, mais constitue une conception réfléchie visant à optimiser la gestion des frais et des autorisations.

Pour maximiser la protection, il est recommandé de documenter nos choix de conception (comme l'utilisation des tables séparées pour les rôles distincts) et de souligner l'effort créatif dans l'organisation des données. Cela permettra de montrer clairement en quoi notre base de données présente une originalité suffisante pour être protégée par le droit d'auteur.

Autre régime de protection applicable

La base de données GSB peut également bénéficier de la protection « sui generis » des bases de données en vertu de la directive européenne sur la protection des bases de données (directive 96/9/CE). Ce droit est conféré au producteur de la base de données s'il peut démontrer un investissement substantiel (financier, humain ou technique) dans la constitution, la vérification ou la présentation du contenu de la base de données.

Cette protection permet au producteur de protéger contre l'extraction ou la réutilisation non autorisée de tout ou d'une partie substantielle du contenu de la base de données.

V. PARTIE DEVELOPPEMENT (SLAM)

1. ARCHITECTURE APPLICATIVE

a. Schéma de Principe

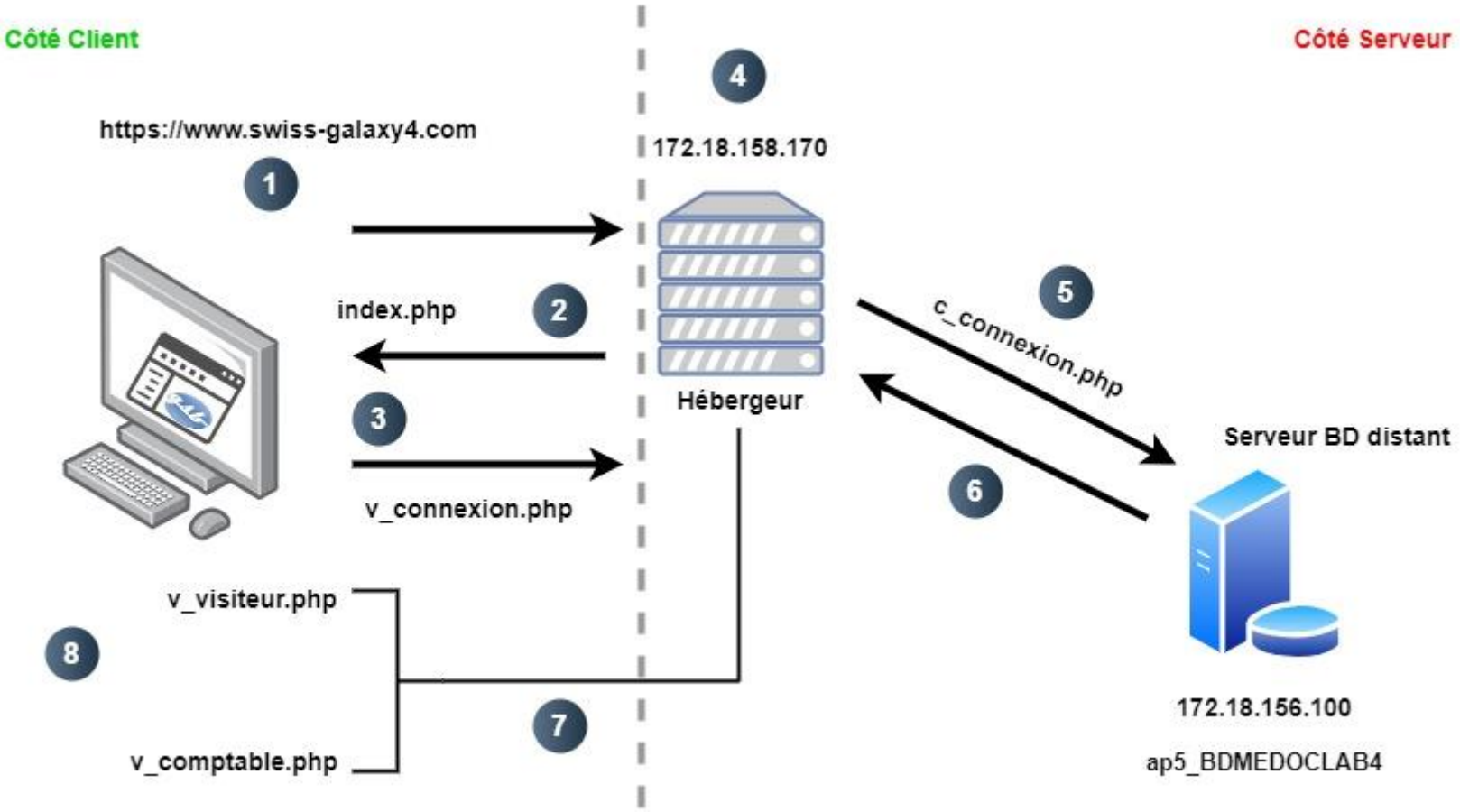

Figure 3 - Schéma de l'architecture applicative

Côté Client

1. **Requête initiale du client** : Le client, à travers son navigateur, envoie une requête vers l'URL https://www.swiss-galaxy4.com.
2. **Chargement de la page index.php** : Le serveur reçoit la requête et renvoie la page **index.php** au client. Cette page contient les éléments de base pour initialiser l'interaction.
3. **Page de connexion (v_connexion.php)** : Sur la page **index.php**, une authentification est requise pour accéder aux différents modules. Le client remplit le formulaire de connexion, qui est ensuite envoyé au serveur via la page **v_connexion.php**.

Côté Serveur

4. **Hébergement** : La demande de connexion arrive à un serveur hébergeur, identifié par l'adresse IP **172.18.158.170**.
5. **Communication avec le serveur de base de données** : Le serveur hébergeur interagit avec un serveur de base de données distant via le script c_connexion.php pour vérifier les informations fournies par l'utilisateur. Cette communication permet de valider les informations d'authentification.
6. **Serveur de base de données distant** : Le serveur de base de données, situé à l'adresse IP **172.18.156.100** et contenant la base de données ap5_BDMEDOCLAB4, vérifie si les informations fournies sont correctes. Ensuite, il renvoie la réponse au serveur hébergeur.

Retour vers le Client

7. **Accès aux vues spécifiques** : En fonction du résultat de l'authentification :
    - Si l'utilisateur est un visiteur, la page **v_visiteur.php** est renvoyée.
    - Si l'utilisateur est un comptable, la page **v_comptable.php** est renvoyée.
8. **Affichage sur le client** : Le client reçoit soit la page visiteur soit la page comptable en fonction des privilèges accordés, permettant à l'utilisateur d'interagir avec l'interface appropriée.

b. Principe Général de l'Architecture MVC

Model (Modèle) :

- **Rôle** : Le modèle est la partie qui représente la logique métier et les données de l'application. Il interagit avec la base de données et gère les règles métiers.
- **Fonctionnement** : Il est responsable de récupérer, insérer, modifier et supprimer les données. Le modèle est totalement indépendant de l'interface utilisateur, ce qui permet de faire évoluer la base de données sans affecter directement l'interface.

View (Vue) :

- **Rôle** : La vue est la partie de l'application qui est responsable de l'affichage des données à l'utilisateur. Elle est chargée de présenter les informations de façon compréhensible.
- **Fonctionnement** : La vue récupère les données du modèle par l'intermédiaire du contrôleur et les affiche. Elle ne contient aucune logique métier mais est focalisée sur le rendu visuel.

Controller (Contrôleur) :

- **Rôle** : Le contrôleur est le lien entre la vue et le modèle. Il reçoit les entrées de l'utilisateur (comme les actions via des formulaires ou des clics), traite les demandes, fait appel aux modèles nécessaires, et renvoie les résultats aux vues appropriées.
- **Fonctionnement** : Il gère les requêtes HTTP (GET, POST) de l'utilisateur, récupère les données depuis le modèle, et détermine quelle vue doit être utilisée pour l'affichage des informations.

Le principe de séparation des préoccupations est au cœur de l'architecture MVC. Le modèle gère la logique métier et les données, la vue s'occupe de l'affichage, et le contrôleur fait le lien entre les deux. Ainsi, chaque composant est indépendant des autres, ce qui permet de modifier l'un sans impacter directement les autres.

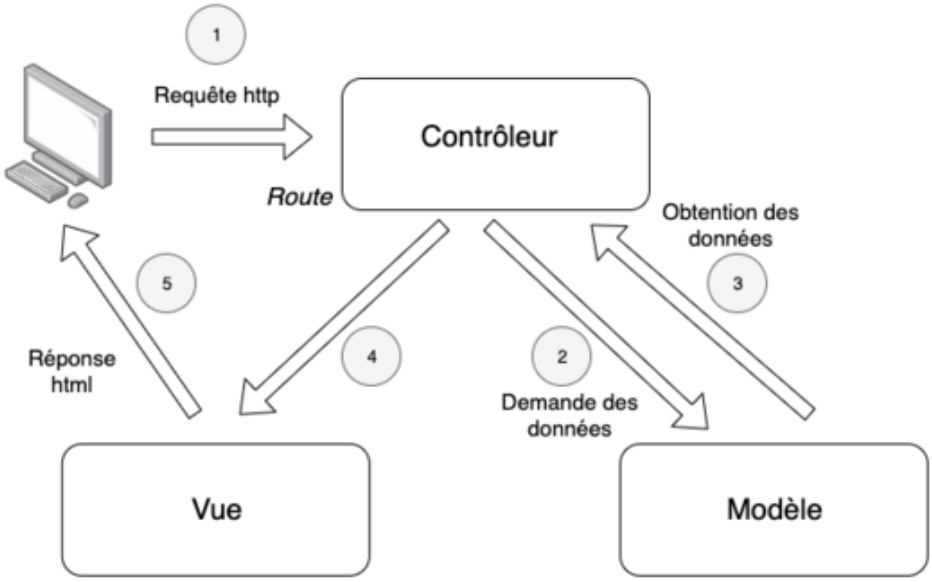

Figure 4 - Interaction MVC

2. BASE DE DONNÉE

a. Conception de la Base de Données Visiteurs et Comptables

La conception de la base de données repose sur la séparation des utilisateurs en deux entités distinctes : **Visiteurs** et **Comptables**. Une table Visiteur a été définie pour gérer les informations personnelles des visiteurs, tandis qu'une table Comptable est dédiée aux comptables, incluant des attributs tels que *nom*, *prenom*, *login*, *adresse*, et *dateEmbauche*. Cette séparation permet de gérer facilement les rôles et les responsabilités spécifiques à chaque type d'utilisateur, notamment les validations de frais effectuées par les comptables.

b. Justification des Choix de Conception des Modèles de Données

Le choix de créer des tables distinctes pour les **Visiteurs** et les **Comptables** répond aux besoins suivants :

1. **Séparation des Rôles** : Les comptables possèdent des responsabilités distinctes, comme la validation des fiches de frais et le suivi des paiements. Cette séparation facilite la gestion des accès et des autorisations spécifiques.

2. **Sécurité et Gestion des Droits** : En isolant les comptables dans une table spécifique, il est plus facile d'appliquer des règles de sécurité appropriées, comme la gestion de l'accès aux fonctions critiques (validation des frais).

3. **Évolutivité** : Cette conception permet de rajouter facilement de nouveaux types d'utilisateurs, comme des administrateurs, sans affecter la structure des données des visiteurs ou des comptables, offrant ainsi une meilleure flexibilité pour les évolutions futures du système.

Ces choix assurent une gestion optimisée des utilisateurs, en tenant compte de leurs rôles spécifiques, tout en facilitant l'ajout de nouvelles fonctionnalités à mesure que le projet évolue.

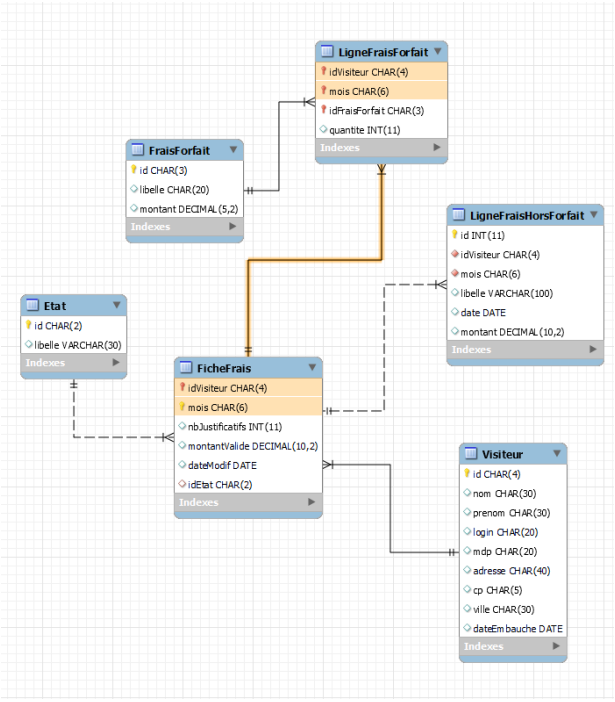

Figure 5 - BD avant modification

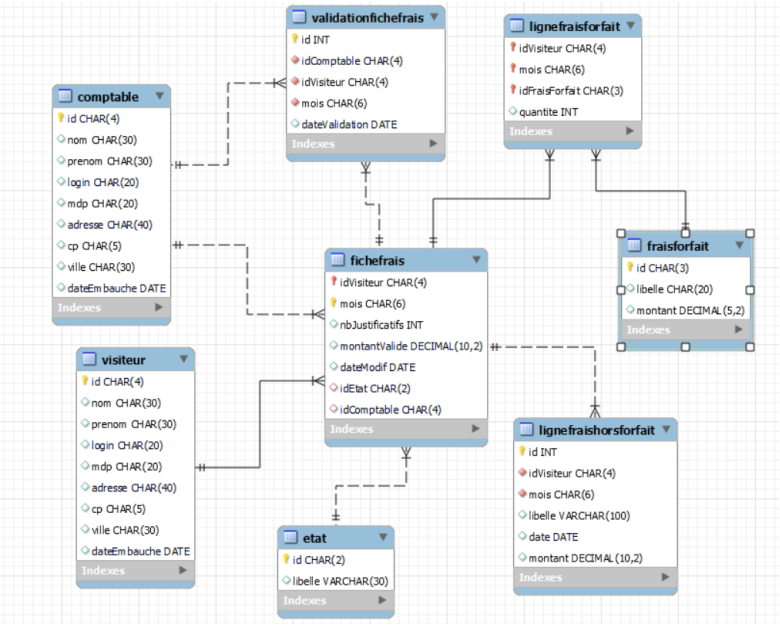

Figure 6 - BD après modification

3. CYBERSÉCURITÉ

- **Injection SQL**
    - **Risque** : L'injection SQL est l'un des risques les plus communs dans une application utilisant une base de données. Cela se produit lorsque des entrées utilisateur ne sont pas correctement filtrées, permettant ainsi à un attaquant d'insérer des requêtes SQL malveillantes.
    - **Impact** : L'attaquant peut manipuler les requêtes pour accéder, modifier, ou même supprimer des données sensibles de la base de données.
    - **Mitigation** : Utiliser des requêtes préparées avec des paramètres (requêtes paramétrées) ou des ORM (Object-Relational Mapping) qui filtrent les entrées.
- **Absence de Gestion d'Accès Fine**
    - **Risque** : Dans l'architecture actuelle, il y a une différenciation entre les utilisateurs visiteurs et comptables, mais si cette gestion des rôles n'est pas correctement appliquée, un utilisateur non autorisé pourrait accéder à des fonctionnalités critiques.
    - **Impact** : Un utilisateur visiteur pourrait potentiellement accéder à des actions réservées aux comptables, telles que la validation des fiches de frais.
    - **Mitigation** : Utiliser des mécanismes de contrôle d'accès robustes, tels que la vérification des rôles et des permissions avant chaque action critique.
- **Scripts Cross-Site Request Forgery (CSRF)**
    - **Risque** : Si des protections contre CSRF ne sont pas en place, un attaquant peut inciter un utilisateur authentifié à exécuter une action non désirée sur l'application.
    - **Impact** : Cela peut permettre à un attaquant de réaliser des actions à la place de l'utilisateur, par exemple, valider une fiche de frais sans que l'utilisateur légitime ne le veuille.
    - **Mitigation** : Utiliser des jetons CSRF pour valider la légitimité de chaque requête provenant de l'utilisateur.

4. TESTS ET VÉRIFICATION DE L'INTÉGRITÉ

1er Test des Contraintes de Not Null

- Lors de nos tests, nous avons tenté d'insérer une ligne dans la table visiteurs sans remplir le champ nom. Comme attendu, cette opération a échoué. Ce test nous a permis de vérifier que les champs obligatoires sont bien respectés dans notre base de données.
- **But** : Assurer que tous les champs obligatoires sont correctement renseignés pour éviter des incohérences dans les données.

2e Test des Contraintes de Clé Primaire

- Nous avons essayé de créer deux enregistrements avec la même clé primaire, par exemple deux fiches de frais avec le même identifiant (idFicheFrais). Comme prévu, cette tentative a été refusée par le système. Cela nous a permis de confirmer l'unicité des identifiants.
- **But** : Vérifier l'unicité des identifiants afin d'éviter les doublons dans la base de données.

3e Test de Cohérence des États de Frais

- Nous avons tenté de valider une fiche de frais déjà marquée comme "payée". Le système a empêché cette action, garantissant ainsi une progression logique des états de la fiche (créé → validé → payé). Cela confirme que notre application respecte les règles de gestion des états.
- **But** : Assurer la cohérence des processus métier et garantir une progression ordonnée des différents états d'une fiche de frais.

Ces tests nous ont permis de confirmer que notre base de données respecte les contraintes essentielles et que les processus métiers sont correctement gérés.

VI. PARTIE RESEAUX (SISR)

1. MISE EN PLACE DE L'INFRASTRUCTURE

a. Schéma de l'architecture réseau

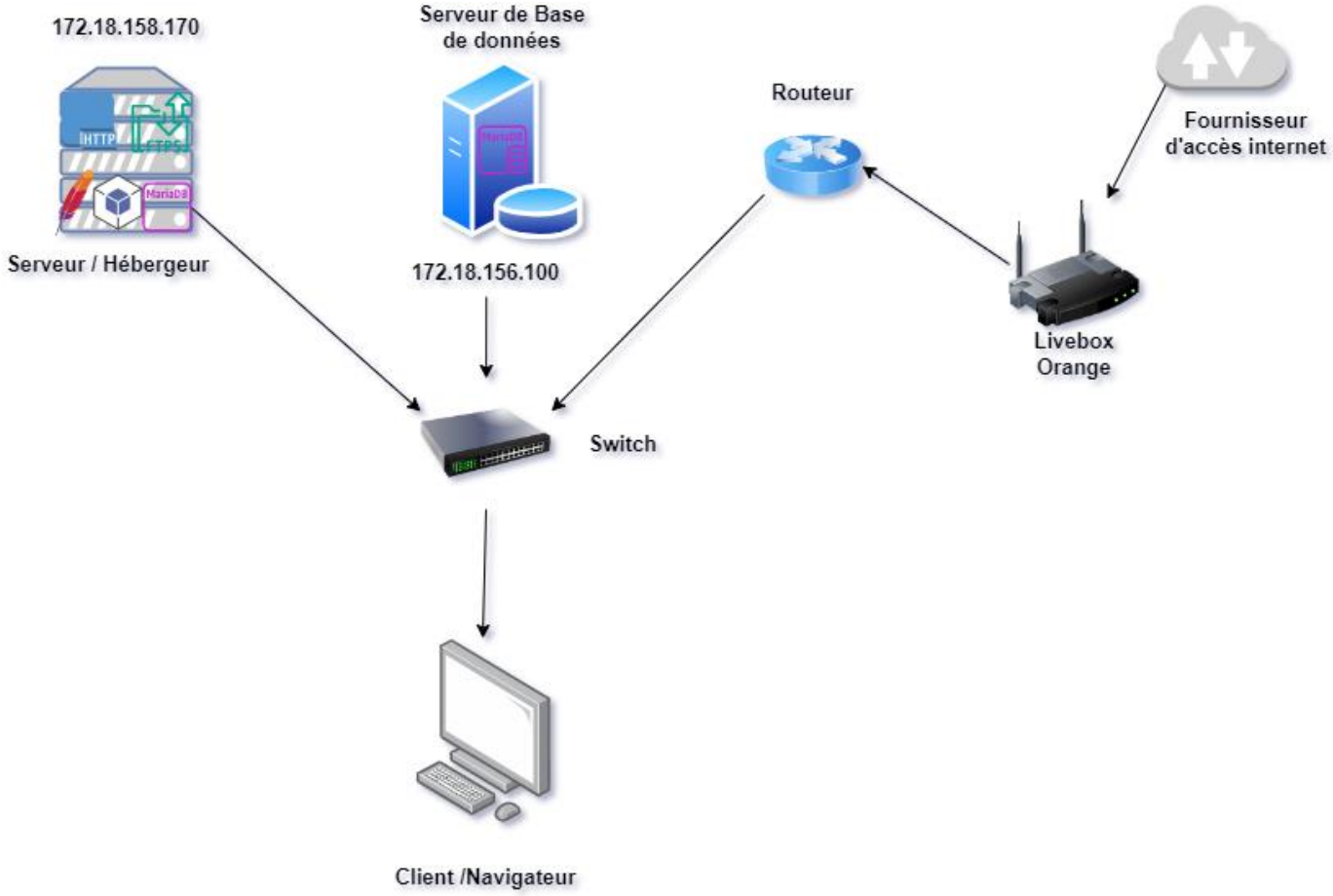

Figure 7 - Schéma de l'architecture réseau

2. SÉCURISATION ET SAUVEGARDE

a. Objectif

Le document explique comment tester une solution de sauvegarde et de restauration afin de garantir que le site web puisse être restauré rapidement en cas de sinistre (comme une perte de données ou une défaillance).

b. Pré-requis

Le document liste les pré-requis pour exécuter ces tests, notamment :

- Un système de sauvegarde automatisé fonctionnel.
- Un accès root ou sudo pour exécuter les commandes.
- Les scripts de sauvegarde et de restauration (sauvegarde_script.sh, restauration_script.sh).
- Un menu (***menu.sh***) pour faciliter les opérations.

c. Tests

Test de la Sauvegarde :

- Vérification des sauvegardes disponibles sur le serveur via une commande (***ls -lh /backups/***).
- Vérification de l'intégrité des sauvegardes avec la commande (***tar -tzvf***).
- Création d'une nouvelle sauvegarde après modification du fichier ***index.php*** via un menu interactif.

Test de la Restauration :

- Simulation d'un sinistre en supprimant des fichiers du site pour tester la restauration.
- Utilisation du menu pour restaurer la version souhaitée du site.
- Vérification que la restauration est complète et que le site fonctionne correctement après avoir relancé le service web.

d. Rapport de Test

Le rapport détaille les résultats des tests, comme :

- Vérification des sauvegardes existantes et de leur intégrité.
- Confirmation que la restauration a été effectuée avec succès.
- Test de la fonctionnalité du site après la restauration.

3. PREUVE DE CONCEPT (POC)

Sécurisation des Échanges entre la Plateforme Web et le Serveur de BDD avec SSL/TLS

a. Objectif

Le but de ce POC est de prouver la faisabilité de l'utilisation de **SSL/TLS** pour sécuriser les communications entre la plateforme web et le serveur de base de données distant hébergeant la base de données BDMEDOCLAB4. Cela garantit que toutes les données échangées sont **chiffrées**, **authentifiées**, et **protégées** contre toute interception ou altération.

b. Architecture du POC

L'architecture du POC comprend deux composants principaux :

- **Plateforme Web (Serveur d'application)** : Le serveur web héberge l'application qui interagit avec la base de données distante.
- **Serveur de Base de Données (BDD)** : Il héberge la base de données BDMEDOCLAB4 et est configuré pour n'accepter que des connexions sécurisées via SSL/TLS.

Les **certificats SSL/TLS** permettent de chiffrer les communications et de s'assurer que seules des connexions sécurisées sont autorisées.

c. Solution Proposée : Utilisation de SSL/TLS

Le protocole **SSL/TLS** (Secure Sockets Layer / Transport Layer Security) va chiffrer les données échangées entre la plateforme web et le serveur de base de données distant. Le **certificat SSL** permet d'authentifier chaque serveur, et le chiffrement empêche quiconque d'intercepter ou de lire les données échangées.

d. Étapes de Mise en Œuvre du POC

Étape 1 : Vérification des Certificats SSL/TLS

Les certificats SSL/TLS doivent être déjà installés sur le serveur de base de données distant. Vous devrez vérifier que le certificat est en place et que le serveur est configuré pour accepter des connexions chiffrées.

- **Vérifier le certificat SSL/TLS** avec openssl
    - Cela permettra de s'assurer que le certificat est valide et que la connexion via TLS est possible.

Étape 2 : Configuration du Serveur de Base de Données pour SSL/TLS

Le serveur de base de données doit être configuré pour **n'accepter que des connexions chiffrées**.

Exemple pour MySQL :

- **Modifier le fichier de configuration MySQL** pour forcer l'utilisation de TLS
- **Redémarrer le serveur MySQL** pour appliquer les changements

Étape 3 : Configuration de la Plateforme Web pour Utiliser SSL/TLS

La plateforme web doit être configurée pour se connecter à la base de données en utilisant SSL/TLS.

- Cela permet à l'application web de se connecter uniquement via une connexion sécurisée et chiffrée.

Étape 4 : Tests de Validation

Test 1 : Connexion Sécurisée avec TLS

- Utilisez **Wireshark** ou **openssl** pour vérifier que les communications sont bien chiffrées.
    - Résultat attendu : la sortie doit indiquer que la connexion est sécurisée avec TLS.

Test 2 : Refus des Connexions Non Sécurisées

- Essayez de vous connecter à la base de données **sans SSL** pour vérifier que la connexion est refusée
    - Résultat attendu : La connexion non sécurisée est rejetée.

e. Résultats Attendus

1. **Sécurisation des communications** : Toutes les données échangées entre la plateforme web et le serveur de BDD sont chiffrées avec TLS.
2. **Authentification mutuelle** : Le serveur web et le serveur de BDD s'authentifient mutuellement via les certificats SSL.
3. **Rejet des connexions non sécurisées** : Toute tentative de connexion non chiffrée est refusée par le serveur de BDD.

f. Conclusion

Ce POC montre que la **sécurisation des échanges** entre la plateforme Web et le serveur de base de données distant via **SSL/TLS** est non seulement faisable, mais qu'elle offre également un niveau de sécurité élevé. Le POC prouve que :

- Les communications sont chiffrées, protégeant les données sensibles contre les interceptions.
- Les connexions non sécurisées sont rejetées, garantissant que seules les connexions chiffrées sont acceptées.

La mise en place de cette solution de sécurité est **fortement recommandée** pour garantir l'intégrité, la confidentialité et l'authenticité des données échangées entre vos serveurs.

VII. BILAN DU PROJET

1. BILAN GÉNÉRALE (SLAM/SISR)

Le projet de développement d'une application Web de gestion des frais pour le laboratoire Galaxy Swiss Bourdin (GSB) a été une expérience extrêmement formatrice pour notre équipe, composée de membres spécialisés en développement applicatif (SLAM) et en administration des systèmes et réseaux (SISR). Ce projet nous a permis de travailler dans des conditions proches de celles du monde professionnel, nous confrontant à des contraintes techniques et organisationnelles significatives.

Au démarrage, la mise en route du projet a été relativement compliquée. La répartition des tâches n'était pas encore claire et les membres de l'équipe devaient s'adapter à des technologies et des environnements qui leur étaient parfois peu familiers. En outre, le fait de devoir composer avec un code déjà existant pour les membres SLAM a ajouté une couche supplémentaire de complexité. Cependant, après quelques semaines, nous avons réussi à trouver un **rythme de travail fluide** et à établir une collaboration efficace entre les deux équipes. La communication a joué un rôle clé dans cette évolution : des échanges réguliers et ouverts via des outils comme Discord ont permis de résoudre rapidement les problèmes qui surgissaient et de maintenir tout le monde aligné sur les objectifs du projet.

En dépit de ces défis initiaux, **la synergie entre les équipes SLAM et SISR s'est considérablement améliorée au fil du projet**. Chaque membre a pu mettre à profit ses compétences spécifiques tout en apprenant des autres. Cette collaboration interdisciplinaire a renforcé la dynamique du projet, chaque équipe apportant son expertise pour surmonter les défis techniques et organisationnels. Grâce à une répartition plus claire des responsabilités au fur et à mesure du projet, nous avons pu atteindre une efficacité opérationnelle et assurer un suivi précis des différentes étapes.

La **gestion des priorités** a également joué un rôle crucial dans la bonne conduite du projet. Confrontés à des délais serrés et à une charge de travail importante, nous avons dû faire des choix stratégiques quant aux fonctionnalités à développer en priorité, afin de garantir un produit final opérationnel et sécurisé. Les outils de planification utilisés, comme Monday.com et Kanboard, ont été très utiles pour visualiser l'avancement et ajuster les tâches en fonction des besoins du moment.

En résumé, ce projet a été une expérience enrichissante qui nous a permis d'acquérir de nouvelles compétences techniques, d'améliorer notre capacité à travailler en équipe, et de mieux comprendre les exigences d'un environnement professionnel réel. Nous avons également appris l'importance de la **flexibilité** et de l'**adaptation** face aux imprévus, des qualités essentielles pour mener à bien tout projet complexe.

2. JUSTIFICATION ÉCARTS PREVISIONNEL/RÉALISÉ

Malgré une planification initiale bien établie, certains **écarts** ont été constatés dans l'exécution du projet en raison de la complexité des tâches techniques et des découvertes tardives de certaines contraintes.

Côté SLAM, l'un des principaux facteurs de retard a été la **compréhension tardive des cas d'utilisation** pour le développement du site. L'équipe a sous-estimé la complexité des cas d'utilisation détaillés, ce qui a retardé le codage de certaines fonctionnalités clés. De plus, l'adaptation au code préexistant a nécessité plus de temps que prévu, car il a fallu analyser et comprendre en profondeur la structure mise en place avant de pouvoir y apporter des modifications. Enfin, la charge de travail a été sous-évaluée en début de projet, ce qui a conduit à un rythme de travail plus lent que prévu dans les premières semaines. Ces défis ont, toutefois, permis à l'équipe SLAM de se familiariser davantage avec les technologies et de mieux anticiper les difficultés pour des projets futurs.

Côté SISR, la plupart des tâches ont été accomplies dans les délais fixés, à l'exception de deux aspects importants qui ont pris plus de temps que prévu. La configuration du **FTPS** s'est avérée particulièrement complexe, nécessitant une reprise complète du processus, ce qui a retardé son implémentation jusqu'à la dernière semaine du projet. De plus, la **découverte tardive du driver PDO PHP manquant** a entraîné des ajustements techniques imprévus pour garantir la communication entre l'application Web et la base de données. Ces retards n'ont pas compromis le déroulement général du projet, mais ils ont obligé l'équipe SISR à ajuster ses priorités pour finaliser les éléments essentiels dans les délais.

Ces écarts, bien qu'ils aient ralenti certaines phases du projet, ont permis aux deux équipes de **monter en compétence** sur des technologies et des méthodologies qu'elles ne maîtrisaient pas initialement. En fin de compte, cela a contribué à l'enrichissement de notre savoir-faire et à une meilleure gestion des imprévus.