

Les besoins de l'application peuvent être résumés comme suit :

Gestion des frais de déplacement : L'application doit permettre aux visiteurs médicaux de saisir et de gérer les frais de déplacement engagés lors de leurs déplacements professionnels, tels que les frais de transport (billets de train, d'avion, de taxi, etc.) et les frais de carburant.

Gestion des frais de restauration : L'application doit permettre aux visiteurs médicaux de saisir et de gérer les frais de restauration engagés lors de leurs déplacements, que ce soit pour les repas pris à l'extérieur ou pour les frais de repas lors des nuitées à l'hôtel.

Gestion des frais d'hébergement : L'application doit permettre aux visiteurs médicaux de saisir et de gérer les frais d'hébergement engagés lors de leurs déplacements, que ce soit pour les nuitées à l'hôtel ou pour les frais de logement lors des séjours dans les départements et territoires d'outre-mer.

Suivi des dépenses : L'application doit permettre de suivre les dépenses des visiteurs médicaux en temps réel, avec des fonctionnalités de suivi des budgets alloués et des alertes en cas de dépassement des budgets prévus.

Gestion des justificatifs : L'application doit permettre aux visiteurs médicaux de joindre les justificatifs de leurs dépenses, tels que les tickets de transport, les factures de restauration et d'hébergement, de manière électronique, pour faciliter la gestion administrative et la comptabilisation des dépenses.

Reporting : L'application doit générer des rapports de synthèse sur les dépenses réalisées par les visiteurs médicaux, par secteur géographique, par type de dépenses, etc., pour permettre une analyse fine des coûts et une prise de décision éclairée.

En résumé, l'application de gestion des frais de déplacement, de restauration et d'hébergement pour les visiteurs médicaux de GSB devra permettre la saisie, la gestion, le suivi, la comptabilisation et le reporting des dépenses liées aux déplacements professionnels, dans le but d'optimiser les coûts et d'améliorer le suivi de l'activité de visite médicale.

Voici une description des outils de gestion de projet que j'ai mis en place en utilisant MindMeister pour réaliser le projet d'application de gestion des frais de déplacement, de restauration et d'hébergement pour le laboratoire Galaxy Swiss Bourdin (GSB) :

MindMeister : J'ai utilisé MindMeister comme outil principal pour la planification et l'organisation du projet. MindMeister est une application de cartographie mentale en ligne qui permet de créer des diagrammes visuels pour représenter les idées, les tâches, les ressources et les dépendances du projet de manière claire et organisée. J'ai créé une carte mentale pour le projet, avec des branches pour les différentes parties du projet, telles que la planification, la conception, le développement, les tests, la mise en œuvre, etc. J'ai utilisé des sous-branches pour détailler les tâches spécifiques à accomplir pour chaque partie du projet, ainsi que les ressources nécessaires et les dépendances entre les tâches.

Collaborateurs : J'ai invité les membres de l'équipe du projet à collaborer sur la carte mentale dans MindMeister. Cela nous a permis de travailler ensemble en temps réel, d'ajouter des commentaires, de partager des idées et de mettre à jour les tâches et les informations du projet. Chaque membre de l'équipe avait accès à la carte mentale et pouvait y contribuer, ce qui favorisait la communication et la collaboration entre les membres de l'équipe.

Planification : J'ai utilisé MindMeister pour créer un planning détaillé du projet, en ajoutant des tâches, des

dates d'échéance et des responsables pour chaque activité. Cela nous a permis de visualiser les dépendances entre les tâches, de suivre l'avancement du projet et de s'assurer que les délais étaient respectés.

Suivi des ressources : J'ai utilisé des icônes et des couleurs dans MindMeister pour identifier les différentes ressources nécessaires pour le projet, comme les compétences, les équipements, les budgets, etc. Cela nous a permis de garder une vue d'ensemble des ressources nécessaires et de nous assurer que nous disposions de tout ce dont nous avons besoin pour mener à bien le projet.

L'application de gestion des frais de déplacement, de restauration et d'hébergement pour les visiteurs du laboratoire Galaxy Swiss Bourdin (GSB) sera développée en utilisant plusieurs produits techniques et documents associés.

Conception de l'interface :

Croquis (wireframe) : Des dessins, qu'ils soient réalisés sur papier ou numériques, montrant les liaisons entre les différents écrans de l'application. Ces croquis

serviront de base pour la conception de l'interface utilisateur.

Mockup : Des images plus élaborées de l'interface, généralement créées avec des outils de design graphique, montrant de manière visuelle et esthétique comment l'application sera présentée à l'utilisateur. Ces images ne sont pas cliquables et permettent de mieux visualiser l'apparence finale de l'application.

Développement du programme et documents de conception divers :

Documents de conception : Il s'agit de documents décrivant en détail les fonctionnalités, les flux de données, les flux de navigation et autres éléments de conception de l'application. Ces documents peuvent inclure des spécifications fonctionnelles, des diagrammes de flux, des descriptions d'algorithmes, etc. Ils servent de référence pour les développeurs tout au long du processus de développement de l'application.

Bibliothèques du JDK et bibliothèques externes : Le JDK (Java Development Kit) est un ensemble d'outils de développement pour le langage de programmation Java. Les bibliothèques du JDK sont des collections de classes Java préconstruites qui peuvent être utilisées pour faciliter le développement d'applications. En plus

des bibliothèques du JDK, d'autres bibliothèques externes peuvent également être utilisées, en fonction des besoins de l'application. Ces bibliothèques fournissent des fonctionnalités supplémentaires et des outils pour faciliter le développement de l'application.

Conception de la base de données :

Analyse : Il s'agit de l'analyse des besoins en matière de stockage de données de l'application, des entités impliquées, des relations entre ces entités, et des fonctionnalités de recherche, de tri, de filtrage, etc. nécessaires pour manipuler ces données.

Dictionnaire de données : Il s'agit d'un ensemble de définitions et de descriptions détaillées des entités, des attributs, des types de données, des contraintes, etc. qui seront utilisées dans la base de données de l'application.

Schéma relationnel : Il s'agit de la représentation visuelle des tables, des relations et des clés étrangères dans la base de données de l'application, généralement sous forme de diagrammes.

Diagramme MCD (Modèle Conceptuel de Données) : Il s'agit d'un modèle conceptuel de données qui décrit les entités, les attributs et les relations entre les entités de manière abstraite et conceptuelle. Ce modèle est

utilisé comme base pour la conception de la base de données relationnelle de l'application.

La documentation utilisateur de l'application de gestion des frais de déplacement, de restauration et d'hébergement pourrait être sous la forme d'un document PDF, Markdown ou HTML, et serait destinée aux utilisateurs finaux du logiciel. Voici une description de son contenu :

Introduction : Une introduction concise expliquant l'objectif et les fonctionnalités de l'application de gestion des frais, ainsi que son importance pour les visiteurs médicaux de l'entreprise GSB.

Guide d'utilisation : Une section détaillée fournissant des instructions étape par étape sur la manière d'utiliser l'application, y compris les fonctionnalités principales, les actions à effectuer et les écrans disponibles. Des captures d'écran pourraient être incluses pour illustrer les étapes clés.

Gestion des frais : Une section spécifique décrivant en détail comment les utilisateurs peuvent gérer les différents types de frais de déplacement, de restauration et d'hébergement dans l'application, y compris la création, la modification et la suppression

des frais, ainsi que la recherche et le filtrage des frais existants.

Gestion des rapports : Une section expliquant comment les utilisateurs peuvent générer des rapports sur les frais enregistrés, y compris les options de filtrage, les formats de rapport disponibles et les actions possibles sur les rapports générés.

Paramètres : Une section indiquant comment les utilisateurs peuvent configurer les paramètres de l'application, tels que la gestion du compte utilisateur, la personnalisation des préférences et la gestion des notifications.

FAQ : Une liste de questions fréquemment posées avec leurs réponses correspondantes, afin d'aider les utilisateurs à résoudre les problèmes courants ou à répondre à leurs interrogations.

La documentation du code, quant à elle, sera intégrée au code source de l'application et sera documentée au format Javadoc pour les classes et les méthodes publiques. Elle permettra de comprendre le fonctionnement interne de l'application et de faciliter la maintenance future du code. Elle inclura également une section décrivant la procédure de test de l'application, notamment les étapes à suivre pour lancer l'application dans les conditions souhaitées, les

éventuels prérequis techniques, ainsi que les cas de test recommandés pour vérifier le bon fonctionnement de l'application. Cette documentation sera également générée au format HTML à l'aide de l'outil Javadoc, et sera hébergée avec le reste du rapport technique.

voici un rapport du code effectif de l'application qui sera dans le dépôt :

Structure de l'application : L'application sera développée en utilisant une architecture client-serveur, où le client sera une application web et le serveur sera une API RESTful. Le client sera développé en utilisant des technologies web modernes comme HTML5, CSS3 et JavaScript, tandis que le serveur sera développé en utilisant un framework de développement web tel que Django, Laravel, ou Express, en fonction des compétences et des préférences de l'équipe de développement.

Gestion des utilisateurs : L'application aura un système d'authentification et d'autorisation pour gérer les utilisateurs. Les visiteurs médicaux et les administrateurs de l'entreprise seront en mesure de se connecter à l'application avec leurs identifiants uniques et d'accéder aux fonctionnalités appropriées en fonction de leur rôle. Les informations des

utilisateurs seront stockées dans une base de données sécurisée et cryptée.

Gestion des frais de déplacement, de restauration et d'hébergement : Les visiteurs médicaux pourront enregistrer leurs frais de déplacement, de restauration et d'hébergement dans l'application à l'aide d'un formulaire. Les informations saisies seront stockées dans la base de données et associées au visiteur médical qui a effectué les dépenses. Les administrateurs pourront consulter et gérer les frais enregistrés, y compris les approuver, les rejeter ou les modifier si nécessaire.

Gestion des remboursements : Les administrateurs seront en mesure de gérer les remboursements des frais approuvés. L'application calculera automatiquement le montant à rembourser en fonction des frais approuvés et des politiques de remboursement de l'entreprise. Les remboursements pourront être effectués par virement bancaire ou par tout autre moyen de paiement approprié.

Génération de rapports : L'application permettra de générer des rapports sur les dépenses des visiteurs médicaux, y compris les dépenses par catégorie (déplacement, restauration, hébergement), les dépenses par période de temps, les remboursements

effectués, etc. Les rapports pourront être téléchargés au format PDF, CSV ou Excel.

Intégration avec d'autres systèmes : L'application pourra être intégrée avec d'autres systèmes internes de l'entreprise, tels que le système de gestion des ressources humaines ou le système de comptabilité, pour assurer la cohérence des données et la facilité de gestion des frais de déplacement.

Sécurité : La sécurité de l'application sera une priorité, avec la mise en place de mesures de sécurité appropriées pour protéger les données sensibles des utilisateurs, telles que l'utilisation de protocoles de cryptage pour les communications et le stockage des données, la validation des données d'entrée pour éviter les attaques par injection SQL ou XSS, et la gestion des autorisations d'accès pour s'assurer que seuls les utilisateurs autorisés peuvent accéder aux fonctionnalités appropriées

Cette expérience de développement a été assez intéressante pour moi , car dans un premier temps , j'ai pu voir à quoi pouvait ressembler une condition réelle de développement d'un projet dans une entreprise , comment pouvait se passer la coopération , la repartition des taches , les corrections que l'on pouvait

faire , etc... Cela pourra me servir dans les futurs AP , dans mon stage et potentielles formations mais surtout en entreprise.

Cependant , le point négatif que je devrais retirer de cette expérience est mon gros manque d'expérience dans le côté pratique , j'ai acquis des connaissances des langages informatiques comme Java/Php/HTML que depuis cette année donc à côté de mes collègues qui sont plus agueris à ce niveau là , je n'ai pas pu les aider en profondeur et cela peut être pénalisant dans le cadre d'un projet de développement en entreprise.

Dans les prochains AP , j'essayerai d'avoir un rôle plus important et j'essayerai d'apprendre plus en utilisant Java pour le développement de sites Internet et WorkBench pour la base de données et relier les informations avec les tables/clés.