## **TP DE MCPI**

## Les membres du groupe sont :

## KAVIRA KANGITSI Rosette

## MUGISHO CIRIMWAMI Roland

## JOSUÉ BIHANDO Albert

## FELICIEN MUKAMBA

## FAIDA BASHIMBE

## JOHN KAKUSU

## JUDITH KARUME

## POLEPOLE MIHIGO Janvier

## RAISSA MUKUNDA

## MUBASI WAMPOKO Amitier

## **Question TP 1 :**

Parlez d’un projet informatique :

Son maitre d’ouvrage, son maitre d’oeuvre, le cout du projet, le besoin auquel le projet a été mis en place, les fonctionnalité de l’application réalisée dans le cadre du projet, le modèle de développement utilisé les obstacles auxquls le projet s’était heurté, comment on les a surmonté, les causes de la réussite du projet, la période prévu pour le projet, la période réelle du projet, raisons du retard, le bénéficie généré par le projet

## **Description Détaillée du Projet Informatique**

### **Dénomination du Projet**

Le projet est intitulé : « l'Appui à l'entreprenariat des jeunes dans la Conception d'un Logiciel informatique pour la gestion optimale de 8 établissements scolaires de la ville de Bukavu » Mise en place par GEVAPOM ASBL, accompagné par Louvain Coopération et financé par l’Union Européenne.

### **Maître d’Ouvrage**

Le maître d’ouvrage de ce projet est GEVAPOM ASBL( Groupe Engagé pour les Valeurs Positives des Ménages).

### **Maître d’Œuvre**

La réalisation du projet est confiée à plusieurs acteurs :

* Une équipe de huit informaticiens entrepreneurs composée de développeurs, designers et testeurs.
* Les responsables des établissements scolaires et les administrateurs système sont également des parties prenantes internes. Le projet est supervisé par Félicien Mukamba Kazindja, en tant que Chargé de Suivi et Évaluation (S&E). Il chapeaute l'équipe des analystes, l'équipe de développement et l'équipe de sensibilisation.
* Une équipe de trois sensibilisateurs.

### **Coût du Projet**

Le budget alloué à ce projet est de 35 000 dollars américains.

### **Besoin Auquel le Projet Répond**

Le projet a été mis en place pour répondre à un besoin principal :

* **Objectif global** : Développer et déployer un logiciel web centralisé destiné à gérer les inscriptions des étudiants et les délibérations des notes au sein de huit établissements scolaires.
* **Objectifs spécifiques** :
  + Automatiser le processus d'inscription des élèves.
  + Centraliser la gestion des données relatives aux élèves et à leurs notes.
  + Faciliter l'accès aux informations pour les élèves, les enseignants, les administrateurs, concernant les cours, les bulletins d'inscription et de cours.
  + Assurer la sécurité et la confidentialité des données collectées et gérées. De plus, le projet vise à soutenir l'entreprenariat des jeunes.

### **Fonctionnalités de l’Application**

Le logiciel développé dans le cadre de ce projet offrira les fonctionnalités suivantes :

* Gestion centralisée des inscriptions des étudiants.
* Gestion des délibérations et de la publication des notes.
* Automatisation complète du processus d'inscription des élèves.
* Centralisation des données des élèves et de leurs résultats scolaires.
* Un portail d'accès sécurisé aux informations pour les différents utilisateurs (écoliers, enseignants, administrateurs), incluant les détails des cours et les bulletins.
* Un module d'authentification robuste pour garantir la sécurité des accès et la protection des données.
* Les usagers ont suggéré des fonctionnalités supplémentaires telles que la gestion des unités pédagogiques, la perception des frais scolaires, la gestion des ressources humaines et la gestion de l'horaire scolaire.

### **Modèle de Développement Utilisé**

Bien que le document de planification détaille l'utilisation de plusieurs méthodes de gestion de projet structurées (QQOQCCP, PBS, WBS, OBS, RBS, Diagramme de GANTT) et une approche phasée, le projet a été géré en utilisant la **méthode Agile**. Cette approche permet une flexibilité et une adaptation continue aux besoins.

### **Technologies et Outils Utilisés**

Le développement du logiciel a impliqué les technologies suivantes :

* **Langage de programmation :** Python
* **Framework :** Django
* **Technologies Frontend :** JavaScript, HTML, CSS, Bootstrap, jQuery

Pour la collaboration au sein de l'équipe, les outils suivants ont été utilisés :

* Git et GitHub (pour le contrôle de version)
* Slack
* Microsoft Teams
* Microsoft Forms

### **Les Obstacles aux quels le projet s'était heurté**

Un défi identifié durant la mise en œuvre est le besoin d'héberger le logiciel en ligne pour faciliter l'accès aux utilisateurs en dehors de l'école.

### **Comment on les a surmonté**

Le document de planification anticipe des défis potentiels et propose des plans de contingence, notamment l'utilisation de technologies éprouvées et compatibles, la mise en place de protocoles de sécurité rigoureux, et la prévision de marges de temps et de budget pour faire face aux imprévus.

Un défi identifié durant la mise en œuvre est le besoin d'héberger le logiciel en ligne pour faciliter l'accès aux utilisateurs en dehors de l'école.

Le projet mise sur plusieurs facteurs pour assurer son succès, notamment une planification détaillée, l'utilisation de méthodes de gestion de projet structurées, des objectifs clairs et spécifiques, l'identification des parties prenantes, et des plans de contingence pour gérer les risques.

### **Période Prévue pour le Projet**

La durée totale prévue pour le projet est de 8 mois. Le chronogramme détaillé est structuré en plusieurs étapes séquentielles : Analyse des besoins, Conception de l'architecture, Développement des modules, Tests unitaires et d'intégration, Déploiement, Formation des utilisateurs, et Maintenance post-déploiement.

### **Période Réelle du Projet**

Le projet a été mis en place dans une période de 9 mois, ce retard est dû à la fermeture des écoles.

### **Raisons du Retard**

Le manque de connaissances informatiques de certains enseignants nécessite un effort supplémentaire de la part des formateurs pour expliquer les bases de la saisie et de l'utilisation du logiciel.

Le budget limité alloué à la sensibilisation et au marketing ne permet pas d'intensifier les activités clés telles que les émissions radio et les sketchs à l'échelle de tous les établissements concernés et pendant toute la durée du projet.

La durée de huit mois du projet ne permettra pas à GEVAPOM ASBL d'accompagner toutes les écoles jusqu'à la fin de l'année scolaire, période cruciale pour la production des bulletins finaux, les délibérations et les examens de repêchage.

La contrainte pour les utilisateurs de travailler uniquement dans l'espace où se situe le serveur a limité la flexibilité et l'adoption potentielle du logiciel.

L'insécurité persistante dans la ville de Bukavu a empêché l'accès et l'accompagnement de cinq des treize établissements initialement prévus.

La fermeture des banques à Bukavu depuis le 16 février 2025 en raison du conflit avec le M23 a compliqué les opérations financières de GEVAPOM ASBL.

### **Bénéfices Générés par le Projet**

Les bénéfices attendus incluent une gestion optimisée des établissements scolaires, l'automatisation et la centralisation des processus clés, un accès facilité et sécurisé aux informations, une meilleure garantie de la sécurité et de la confidentialité des données, et un appui concret à l'entreprenariat des jeunes. L'hébergement en ligne du logiciel est perçu comme un facteur clé pour améliorer l'accessibilité, la sécurité, la visibilité et la facilité d'utilisation, augmentant ainsi l'impact dans les zones éducationnelles de Bukavu.

## **Question TP 2 :**

**Dans un projet de développement d’une application de gestion des tâches, identifiez au moins cinq parties prenantes et décrivez leur rôle.**

### **Parties Prenantes Clés dans le Développement d'une Application comme Google Tasks**

Dans un projet à l'échelle de Google Tasks, les parties prenantes sont nombreuses et leurs rôles sont très spécifiques :

1. **Le Client (Google)**
   * **Rôle :** Dans ce cas, le "client" est Google en tant qu'entreprise. Ce sont eux qui définissent la stratégie globale du produit, les objectifs commerciaux (par exemple, intégrer Tasks dans l'écosystème Google Workspace), et qui allouent les ressources (budget, équipes). Ils ont la vision long terme de l'évolution de Google Tasks et de sa place sur le marché.
   * **Exemple :** La direction de Google décide de lancer Google Tasks comme un outil simple et intégré pour aider les utilisateurs à gérer leurs listes de tâches personnelles et professionnelles, s'intégrant parfaitement avec Gmail et Google Calendar.
2. **Les Utilisateurs Finaux (Individus et Organisations utilisant Google Workspace)**
   * **Rôle :** Ce sont les millions d'utilisateurs de Google Tasks, qu'il s'agisse d'étudiants gérant leurs devoirs, de professionnels organisant leurs journées ou d'équipes coordonnant des tâches simples. Leurs besoins varient (création de tâches récurrentes, sous-tâches, intégration mobile, etc.), et leurs retours via les canaux de support ou les enquêtes utilisateurs influencent les nouvelles fonctionnalités et les améliorations.
   * **Exemple :** Un utilisateur individuel a besoin de synchroniser ses tâches entre son ordinateur et son téléphone. Une petite entreprise utilisant Google Workspace a besoin de pouvoir assigner des tâches simples entre collègues via l'intégration Gmail. Les retours sur la difficulté de partager des listes de tâches peuvent orienter le développement.
3. **L'Équipe Produit Google Tasks**
   * **Rôle :** Cette équipe est dédiée à Google Tasks et comprend les chefs de produit (Product Managers), les designers (UX/UI Designers), les ingénieurs logiciels (Développeurs) et les ingénieurs qualité (QA Engineers). Les chefs de produit définissent le "quoi" et le "pourquoi" des fonctionnalités, les designers conçoivent l'interface, les ingénieurs construisent l'application, et les QA s'assurent de sa qualité.
   * **Exemple :** Un chef de produit identifie un besoin fort pour les sous-tâches. Un designer conçoit l'interface pour ajouter et gérer ces sous-tâches. Une équipe d'ingénieurs développe la fonctionnalité et l'intègre dans les applications web et mobiles. Une équipe QA teste rigoureusement pour s'assurer qu'il n'y a pas de bugs.
4. **Les Équipes d'Infrastructure et d'Opérations (Google Cloud, SRE)**
   * **Rôle :** Dans un service à l'échelle de Google, des équipes dédiées sont responsables de l'infrastructure cloud sous-jacente (serveurs, bases de données, réseau), de la fiabilité du service (SRE - Site Reliability Engineering), de la sécurité et de la scalabilité pour gérer des millions d'utilisateurs simultanés.
   * **Exemple :** L'équipe Google Cloud fournit l'infrastructure mondiale sur laquelle Google Tasks fonctionne. Une équipe SRE surveille la performance de l'application, gère les déploiements et s'assure que le service est disponible 99.999% du temps, même en cas de pic d'utilisation.
5. **Les Équipes de Marketing et de Communication**
   * **Rôle :** Ces équipes sont responsables de la promotion de Google Tasks, de l'explication de ses fonctionnalités aux utilisateurs potentiels et existants, et de la gestion de la communication autour des mises à jour ou des changements.
   * **Exemple :** L'équipe marketing crée des articles de blog et des vidéos pour présenter les nouvelles fonctionnalités de Google Tasks. L'équipe de communication gère les annonces de lancement et répond aux questions des médias ou du public concernant le service.

## **Question TP 3 :**

**Identifiez trois risques potentiels dans un projet de développement d’un site web e-commerce et proposez une stratégie pour contourner chacune.**

### **1. Risque : Problèmes de Sécurité des Paiements et des Données Clients**

* **Description du Risque :** Les sites e-commerce traitent des informations sensibles, telles que les données de cartes de crédit, les informations personnelles des clients (adresses, numéros de téléphone, etc.) et l'historique des commandes. Un manquement à la sécurité peut entraîner des violations de données, des fraudes financières pour les clients et le commerçant, une perte de confiance massive, et des conséquences légales sévères. C'est un risque majeur dans le commerce en ligne.
* **Stratégie pour Contourner :**
  + **Mettre en œuvre des protocoles de sécurité robustes :** Utiliser systématiquement le protocole HTTPS pour crypter les données transitant entre le navigateur de l'utilisateur et le serveur.
  + **Conformité aux normes de l'industrie :** Respecter les normes de sécurité des données de l'industrie des cartes de paiement (PCI DSS) si vous traitez directement les informations de carte de crédit.
  + **Utiliser des passerelles de paiement sécurisées :** Intégrer des solutions de paiement tierces reconnues et certifiées (comme Stripe, PayPal Pro, ou les solutions bancaires spécifiques) qui gèrent la majeure partie de la complexité et de la responsabilité liée au traitement des données de paiement sensibles.
  + **Audits de sécurité réguliers et tests d'intrusion :** Faire vérifier régulièrement la sécurité du site par des experts externes pour identifier et corriger les vulnérabilités potentielles avant qu'elles ne soient exploitées.
  + **Formation du personnel :** S'assurer que toute personne ayant accès aux données sensibles est formée aux meilleures pratiques de sécurité.
* **Exemple avec l'Apple Store :** L'Apple Store utilise des technologies de cryptage avancées pour toutes les transactions. Lorsqu'un achat est effectué, les informations de paiement sont traitées via des systèmes sécurisés conformes aux normes les plus strictes. Apple Pay ajoute une couche de sécurité supplémentaire en utilisant des identifiants spécifiques à l'appareil et des codes de transaction uniques, ce qui signifie que les numéros de carte réels ne sont jamais stockés sur les serveurs d'Apple ni partagés avec les commerçants. L'entreprise investit massivement dans la sécurité pour protéger les millions de transactions effectuées quotidiennement.

### **2. Risque : Mauvaise Performance du Site (Lenteur, Indisponibilité)**

* **Description du Risque :** Un site e-commerce lent à charger ou qui devient indisponible, surtout lors de pics de trafic (soldes, lancements de produits), entraîne une frustration des utilisateurs, un taux d'abandon élevé des paniers d'achat, une perte de ventes directe et nuit à l'image de marque. La performance impacte directement l'expérience utilisateur et le référencement (SEO).
* **Stratégie pour Contourner :**
  + **Optimisation des images et des médias :** Compresser et redimensionner les images sans perte significative de qualité. Utiliser des formats d'image modernes (comme WebP).
  + **Mise en cache efficace :** Mettre en place des mécanismes de mise en cache côté serveur et côté navigateur pour réduire le temps de chargement des pages visitées fréquemment.
  + **Utilisation d'un CDN (Content Delivery Network) :** Distribuer le contenu statique (images, CSS, JavaScript) sur des serveurs répartis géographiquement pour servir les utilisateurs à partir du serveur le plus proche, réduisant ainsi la latence.
  + **Optimisation du code et des requêtes de base de données :** S'assurer que le code est efficace et que les requêtes vers la base de données sont rapides.
  + **Tests de charge et de performance :** Simuler un trafic important avant le déploiement pour identifier les goulots d'étranglement et s'assurer que le site peut gérer la charge attendue.
  + **Infrastructure scalable :** Choisir une architecture d'hébergement qui permet de faire évoluer facilement les ressources (serveurs, bande passante) en fonction du trafic.
* **Exemple avec l'Apple Store :** L'Apple Store est connu pour sa rapidité et sa capacité à gérer des millions de visiteurs, notamment lors des lancements de nouveaux iPhone ou autres produits très attendus. Ils utilisent une infrastructure cloud robuste, des réseaux de distribution de contenu mondiaux et une optimisation poussée de leurs assets web pour garantir une expérience fluide, quelle que soit la localisation ou le nombre d'utilisateurs. Des tests de charge massifs sont certainement réalisés avant chaque période de fort trafic.

### **3. Risque : Expérience Utilisateur Complexe ou Non Intuitive**

* **Description du Risque :** Si le site est difficile à naviguer, si le processus de recherche de produits est inefficace, si le panier d'achat est compliqué, ou si le processus de paiement est trop long ou confus, les utilisateurs abandonneront le site avant de finaliser leur achat. Une mauvaise UX nuit à la satisfaction client et aux taux de conversion.
* **Stratégie pour Contourner :**
  + **Conception centrée sur l'utilisateur :** Impliquer les utilisateurs potentiels dès les premières étapes de conception via des interviews, des sondages et des tests d'utilisabilité.
  + **Navigation claire et intuitive :** Organiser les produits en catégories logiques, proposer une barre de recherche performante avec des filtres pertinents, et rendre la navigation cohérente sur tout le site.
  + **Processus de paiement simplifié :** Minimiser le nombre d'étapes, proposer une option de paiement invité (sans création de compte obligatoire), et rendre le formulaire de paiement clair et facile à remplir.
  + **Design responsive :** S'assurer que le site s'affiche correctement et offre une expérience utilisateur optimale sur tous les appareils (ordinateurs de bureau, tablettes, smartphones).
  + **Tests d'utilisabilité :** Observer de vrais utilisateurs interagir avec le site (maquettes, prototypes, versions bêta) pour identifier les points de friction et les difficultés.
  + **Collecte et analyse des retours utilisateurs :** Mettre en place des mécanismes pour recueillir les avis des utilisateurs (formulaires, enquêtes) et utiliser ces informations pour apporter des améliorations continues.
* **Exemple avec l'Apple Store :** L'Apple Store est souvent cité comme un exemple de bonne pratique en matière d'UX. Le site est épuré, la navigation par produits est simple, les pages produits sont claires avec des informations bien organisées, et le processus d'achat est relativement rapide. Ils investissent dans un design minimaliste et intuitif qui guide l'utilisateur efficacement de la découverte du produit à l'achat, sur tous les types d'appareils. Leurs tests d'utilisabilité internes sont très poussés pour garantir cette fluidité.

## **Question TP 4 :**

**Elaborer un planning avec un outil de votre choix pour un projet de développent d’un logiciel en cinq étape**

* Analyse des besoins(2 semaines)
* Conception(3 semaines)
* Développement(5 semaines)
* Tests(2semaines)
* Déploiement(1 semaines)

**Outil de Représentation :** Tableau (simulant un Diagramme de Gantt textuel)

**Hypothèse :** Le projet commence le Lundi 19 Mai 2025 et les phases s'enchaînent séquentiellement. Nous considérons une semaine de travail standard (du Lundi au Vendredi).

**Planning du Projet :**

|  |  |  |  |
| --- | --- | --- | --- |
| **Phase** | **Durée (semaines)** | **Date Début Estimée** | **Date Fin Estimée** |
| Analyse des besoins | 2 | 19/05/2025 | 30/05/2025 |
| Conception | 3 | 02/06/2025 | 20/06/2025 |
| Développement | 5 | 23/06/2025 | 25/07/2025 |
| Tests | 2 | 28/07/2025 | 08/08/2025 |
| Déploiement | 1 | 11/08/2025 | 15/08/2025 |

## **Question TP 5 :**

Créer un diagramme de GANTT simplifié pour les taches suivantes d’un projet de développement d’une application :

* Recherche et Analyse(1 semaines)
* Design(2 semaines)
* Développement(4 semaines)
* Tests(1semaines)
* Lancement(1 semaines)

Projet : Développement d'Application

Date de Début Estimée : Lundi 19 Mai 2025

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tâche** | **Durée (semaines)** | **Semaine 1** | **Semaine 2** | **Semaine 3** | **Semaine 4** | **Semaine 5** | **Semaine 6** | **Semaine 7** | **Semaine 8** | **Semaine 9** |
| **Dates Approximatives** |  | 19/05-23/05 | 26/05-30/05 | 02/06-06/06 | 09/06-13/06 | 16/06-20/06 | 23/06-27/06 | 30/06-04/07 | 07/07-11/07 | 14/07-18/07 |
| Recherche et Analyse | 1 |  |  |  |  |  |  |  |  |  |
| Design | 2 |  |  |  |  |  |  |  |  |  |
| Développement | 4 |  |  |  |  |  |  |  |  |  |
| Tests | 1 |  |  |  |  |  |  |  |  |  |
| Lancement | 1 |  |  |  |  |  |  |  |  |  |

# **Question TP 6 :**

Donnez pour chaque type d’exigences suivant que vous pourriez rencontrer dans un projet de développement d’un logiciel deux exemples pour chacun.

* Exigences fonctionnelles
* Exigences non fonctionnelles
* Exigences de sécurité

**1. Exigences Fonctionnelles**

Les exigences fonctionnelles décrivent ce que le système doit faire, les fonctionnalités qu'il doit offrir pour satisfaire les besoins des utilisateurs. Elles spécifient les actions que l'application peut effectuer et les interactions avec l'utilisateur.

* **Exemple 1 :** L'utilisateur doit pouvoir créer un nouveau compte en fournissant une adresse email et un mot de passe.
* **Exemple 2 :** Le système doit permettre de rechercher des produits par catégorie, mot-clé ou prix.

**2. Exigences Non Fonctionnelles**

Les exigences non fonctionnelles décrivent comment le système doit fonctionner. Elles concernent la qualité, la performance, la fiabilité, la scalabilité, la facilité d'utilisation, etc., plutôt que des fonctionnalités spécifiques.

* **Exemple 1 (Performance) :** Le temps de chargement de chaque page ne doit pas dépasser 3 secondes, même avec 1000 utilisateurs connectés simultanément.
* **Exemple 2 (Disponibilité) :** Le système doit être disponible 99.9% du temps, hors périodes de maintenance planifiée.

**3. Exigences de Sécurité**

Les exigences de sécurité spécifient les mesures nécessaires pour protéger le système et les données contre les accès non autorisés, les modifications ou les destructions, ainsi que pour garantir la confidentialité et l'intégrité des informations.

* **Exemple 1 :** Les mots de passe des utilisateurs doivent être stockés sous forme hachée et cryptée dans la base de données.
* **Exemple 2 :** Seuls les administrateurs système doivent avoir l'autorisation de supprimer définitivement les données utilisateurs.