 **DriveGuard : Application Mobile Alimentée par l'IA pour Conduire en Toute Sécurité**

## 1. Introduction

### Contexte du projet choisi

La conduite reste une activité sujette à de nombreux risques, notamment liés à la distraction, à la fatigue ou à l’imprévisibilité des conditions routières. L’augmentation des accidents de la route, souvent causés par ces facteurs, met en lumière le besoin d’outils innovants pour améliorer la sécurité au volant.  
  
DriveGuard est une application mobile alimentée par l’intelligence artificielle (IA) qui vise à assister les conducteurs en temps réel pour éviter les situations dangereuses.

### Objectifs du projet

- Détecter les signes de fatigue et de distraction chez les conducteurs.  
- Fournir des alertes et des recommandations en temps réel pour améliorer la sécurité.  
- Analyser les données de conduite pour proposer des rapports détaillés et personnalisés.

## 2. Planification du projet

### Déclinaison des objectifs en livrables

- Prototype fonctionnel de l’application avec détection de la fatigue et des distractions.  
- Module d’analyse des données pour produire des rapports.  
- Interface utilisateur intuitive avec des alertes en temps réel.

### Fractionnement des livrables en tâches

1. Pour le livrable de détection des signes de fatigue et de distraction :  
 - Implémentation des algorithmes d’IA pour la reconnaissance faciale et comportementale.  
 - Entraînement des modèles IA avec des données pertinentes.  
 - Développement d’un module d’analyse comportementale en temps réel.  
  
2. Pour le livrable des alertes et recommandations en temps réel :  
 - Conception d’un système de notification et d’alerte sonore.  
 - Optimisation des performances pour une réponse rapide (moins de 2 secondes).  
  
3. Pour le livrable des rapports détaillés et personnalisés :  
 - Création d’un tableau de bord dans l’application.  
 - Développement d’une fonctionnalité de stockage et d’analyse des données de conduite.  
 - Génération de rapports graphiques et textuels.

### Structuration du projet (SDP)

Le projet est divisé en 4 étapes principales :  
1. Étude préliminaire : Analyse des besoins des utilisateurs et des exigences techniques.  
2. Développement : Conception et programmation de l’application.  
3. Tests et validation : Ajustements basés sur les retours des tests.  
4. Lancement : Mise en production et communication.

### Estimation des ressources et des coûts

- Ressources humaines :  
 - 1 expert en IA et machine learning.  
 - 2 développeurs.  
 - 1 designer UX/UI.  
- Coûts estimés :  
 - Développement : 15 000 Dhs.  
 - Hébergement et serveurs : 9 000 Dhs.  
 - Marketing : 7 000 Dhs.

### Techniques de planification

- Diagramme PERT : Pour identifier les dépendances critiques entre tâches.

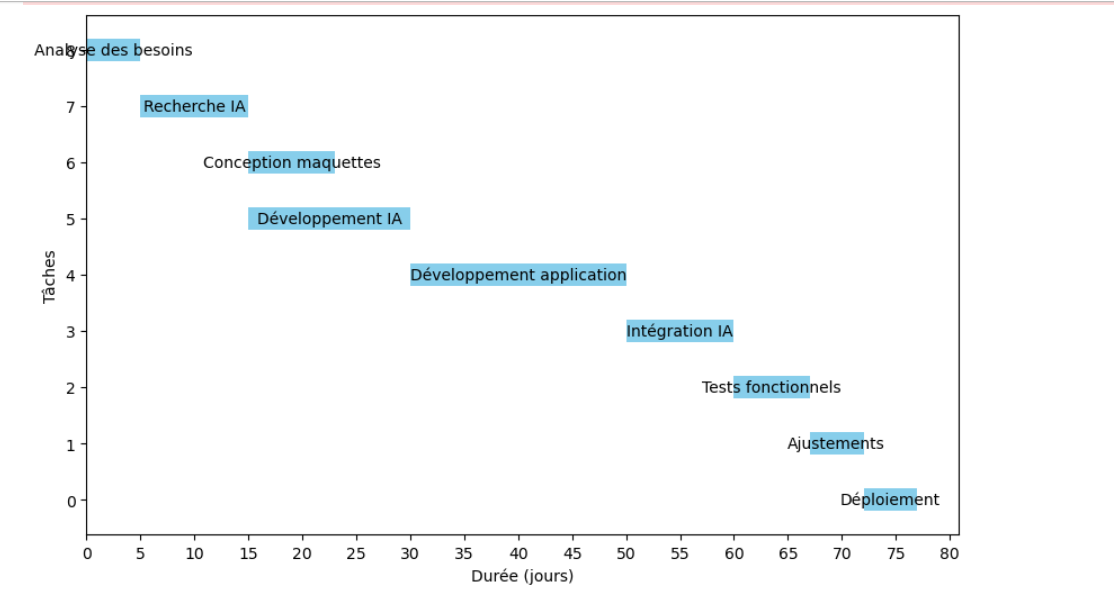
### Insertion de l’image...

### **Calcul de la durée totale du chemin critique :**

5 + 10 + 15 + 20 + 10 + 7 + 5 + 5 = **77 jours**

Le chemin critique est donc : **Analyse des besoins → Recherche IA → Développement IA → Développement application → Intégration IA → Tests fonctionnels → Ajustements → Déploiement**.

- Diagramme de Gantt : Pour suivre l’avancement du projet.



## 3. Exécution, surveillance et contrôle

### Communication dans le projet

Les membres de l’équipe utilisent des outils comme Slack et Jira pour la coordination et le suivi des tâches.

### Mise en place des KPIs

- Taux de détection des signes de fatigue ou distraction (objectif : 95 %).  
- Temps de réponse des alertes (objectif : moins de 2 secondes).  
- Satisfaction des utilisateurs mesurée via des enquêtes (objectif : 4.5/5).

### Gestion des dérives et ajustements

**Tests et ajustements :**

* **Précision des détections :** L'IA doit être capable d'identifier des comportements spécifiques (excès de vitesse, franchissement de ligne, distance de sécurité insuffisante, etc.). Si les tests révèlent des lacunes dans la précision (par exemple, si l’IA manque des incidents ou identifie des comportements normaux comme des dangers), des ajustements doivent être faits pour affiner ces détections.

**Exemples d'ajustements :**

* + Ajuster les modèles de reconnaissance d'image pour détecter des comportements anormaux avec plus de précision.
  + Raffiner les seuils de détection, comme la vitesse maximale ou la distance de sécurité, pour réduire les fausses alertes.
* **Réduction des fausses alertes :** Un système qui envoie trop de fausses alertes pourrait être ignoré par les utilisateurs, réduisant ainsi son efficacité. Les ajustements à ce niveau peuvent impliquer :
  + L'amélioration des algorithmes pour filtrer les événements pertinents (par exemple, en différenciant un freinage brusque nécessaire d'un freinage dangereux et évitable).
  + L'application de techniques de machine learning qui apprennent à distinguer des situations courantes des réels dangers potentiels.

### Mesure des écarts et réajustement des processus

**Analyse des écarts :**

* **Écarts dans les détections d'incidents :** Si le système détecte un taux élevé de faux positifs ou manque des incidents clés, il est nécessaire d'analyser les écarts pour comprendre où les algorithmes ou les règles décisionnelles doivent être révisés.
* **Écarts dans les performances prédictives :** Si l’IA est censée prédire des événements à risque (comme une probabilité élevée d'accident basée sur des comportements antérieurs), ces prédictions doivent être comparées à la réalité pour ajuster le modèle prédictif.

**Affinage des algorithmes :**

* **Prédiction des comportements à risque :** Les algorithmes de prédiction peuvent être affinés pour mieux prévoir les comportements risqués en analysant des données comme la vitesse, les conditions météorologiques, le trafic environnant, etc. Par exemple, le modèle pourrait tenir compte du fait qu'une vitesse élevée sous la pluie est plus dangereuse qu'une vitesse similaire par temps clair.
* **Réentraînement avec des données locales :** Si l’application est déployée dans une nouvelle région, les écarts pourraient provenir d'une différence dans les comportements de conduite ou les infrastructures routières locales. Dans ce cas, il pourrait être nécessaire de réentraîner le modèle avec des données spécifiques à cette région.

**Amélioration de l'interface utilisateur (UI) :** L’interface utilisateur doit permettre aux utilisateurs (conducteurs, gestionnaires de flotte, ou autorités) de comprendre clairement les alertes émises et de prendre des décisions en conséquence.

* **Clarté des alertes :** Les alertes doivent être claires et différenciées entre celles à haute priorité (par exemple, risque de collision imminent) et celles à basse priorité (conseil de réduire la vitesse).
* **Simplicité d'accès aux données :** L'utilisateur doit pouvoir accéder facilement aux informations détaillées, comme la visualisation de la situation qui a conduit à l’alerte (franchissement de ligne, vitesse excessive).
* **Réajustement des processus :**
* **Priorisation des incidents critiques :** En fonction de l'analyse des écarts, des règles pourraient être ajoutées ou ajustées pour prioriser les alertes selon la gravité de l'incident. Par exemple, un freinage brusque dans un traffic dense pourrait être signalé avec une priorité plus haute qu’un excès de vitesse léger sur une route dégagée.
* **Automatisation des rapports d'incidents :** Automatiser la création de rapports après un incident grave peut permettre une meilleure gestion par les autorités ou les gestionnaires de flotte.

## 4. Conclusion

### Résumé des résultats attendus

DriveGuard devrait permettre une conduite plus sûre en fournissant des alertes basées sur les comportements des conducteurs. Elle vise à réduire les accidents causés par la fatigue et la distraction grâce à l’utilisation de l’IA et des technologies avancées.

### Discussion sur les difficultés potentielles et les solutions envisagées

- Difficulté : Résistance des utilisateurs à adopter l’application.  
 Solution : Campagnes de sensibilisation sur les bénéfices de DriveGuard.  
- Difficulté : Précision des algorithmes dans des conditions variées (nuit, port de lunettes).  
 Solution : Recueillir davantage de données pour entraîner les modèles IA.