

Journal de bord

Wohlers Luca

25 Mai 2020	2
Objectif	2
Matinée	2
Après-midi	2
Déroulement de la journée	2
Matinée	2
Après-midi	2
Bilan	2
26 Mai 2020	3
Objectif	3
Matinée	3
Après-midi	3
Déroulement de la journée	3
Matinée	3
Après-midi	3
Bilan	4
27 Mai 2020	5
Objectif	5
Matinée	5
Après-midi	5
Déroulement de la journée	5
Matinée	5
Après-midi	5
Bilan	6
28 Mai 2020	7
Objectif	7
Matinée	7
Après-midi	7
Déroulement de la journée	7
Matinée	7
Après-midi	8
Bilan	8
29 Mai 2020	9
Objectif	9

Wohlers	TPI
Luca	
Matinée	9
Après-midi	9
Déroulement de la journée	9
Matinée	9
Après-midi	9
Bilan	9
02 Juin 2020	11
Objectif	11
Matinée	11
Après-midi	11
Matinée	11
Après-midi	11
Bilan	11
03 Juin 2020	13
Objectif	13
Matinée	13
Après-midi	13
Déroulement de la journée	13
Matinée	13
Après-midi	13
Bilan	14
04 Juin 2020	15
Objectif	15
Matinée	15
Après-midi	15
Matinée	15
Après-midi	15
Bilan	15
05 Juin 2020	17
Objectif	17
Matinée	17
Après-midi	17
Matinée	17
Après-midi	17
Bilan	17

25 Mai 2020

Objectif

Matinée

- Réception de l'énoncé du TPI
- Commencement documentation

Après-midi

- Avancement de la documentation

Déroulement de la journée

Matinée

- Lecture de l'énoncé
- Analyse de l'énoncé
- Définition des tâches importantes
- Réalisation du planning prévisionnel

Après-midi

- Finalisation du planning prévisionnel
- Réalisation du plan de test

Bilan

La journée s'est bien passée, j'ai pu entrer en contact avec mon enseignante qui me suit pour mon travail de TPI pour les quelques questions que j'avais au sujet de l'énoncé. J'ai ensuite pu bien effectuer le planning et le plan de test.

26 Mai 2020

Objectif

Matinée

- Création de la base de données
- Insertion des données dans la bases

Après-midi

- Création du projet
- Création de la page d'accueil
- Création des 4 onglets
- Ajout de la carte du monde (GMap)
- Commencement du CRUD des véhicules

Déroulement de la journée

Matinée

- Création de la base de données "Moto Care" en .sqlite
- Ajout de différentes valeurs dans la base de données.
- 11h00-11h17 Visio-Conférence avec mon enseignante de suivi du TPI Mme. Terrier.
 - Discussion principalement sur la base de données:
 - "KmInitial" est le nombre de kilomètre du véhicule lors de son achat
 - "KmReel" est potentiellement les kilomètres initial + tous les trajets
 - Les Maintenances se répètent avec une certaine fréquence de nombre de kilomètre.
- Commencement à réfléchir au design de l'interface utilisateur.

Après-midi

- Création du projet
- Ajout de la référence System.Data.SQLite
- Ajout using System.Data.SQLite;
- Ajout using System.IO;
- Création d'interfaces:
 - La page d'accueil qui contient
 - Une combobox pour le choix du véhicule.
 - Une textbox pour la description du véhicule.
 - Une picturebox pour afficher la photo du véhicule.
 - 3 bouton pour Modifier, Supprimer et Ajouter un véhicule.
 - En dessous de la page d'accueil se trouve les 4 onglets pour les trajets, le carnet d'entretiens, la gestion des entretiens et les points d'intérêts.

- L'onglet des trajets contient un bouton pour ajouter un trajet et une dataGridView ou j'afficherai tous les trajets effectués, la grille contient aussi des boutons pour modifier ou supprimer les trajets.
- L'onglet carnet d'entretien contient pour l'instant simplement une dataGridView pour afficher les entretiens
- L'onglet Gestion d'entretiens est vide mais il devrait ressembler fortement à l'onglet trajet.
- L'onglet Points d'intérêts contient pour l'instant une carte du monde.
- J'ai créé une windows form frmAjoutVehicule pour l'ajout de véhicule.
 - Elle contient simplement des champs pour: le nom, la description, le nombre de kilomètres initial, le nombre de kilomètres réel et une insertion d'une photo.
- Création d'une classe pour les véhicules avec les propriétés suivantes:
 - idVehicule (à voir si je le garde)(Rectification: je garde car utile pour la modification de véhicule)
 - nom
 - description
 - kmInitial
 - kmReel
 - photo
- Création d'une classe BD pour tout ce qui est en lien avec la base de données.
- Dans BD:
 - Connexion à la base de données.
 - Création de la méthode CreerVehicule() pour ajouter les nouveaux véhicules à la base de données.
 - Création de la méthode ImageToByteArray() pour convertir l'image en tableau de d'octets. Ce qui permet de la stocker sous cette forme dans la base de données.

Bilan

J'ai pu obtenir des réponses sur mes questions au sujet de la base de données comme mentionné plus haut. Je suis plutôt dans les temps malgré le fait que j'ai perdu beaucoup de temps à me souvenir comment insérer une image dans la base de donnée sqlite. Il m'a fallu retourner voir comment j'avais fait cela dans d'anciens projets effectués précédemment en cours. Et je n'ai pas complètement fini l'interface.

27 Mai 2020

Objectif

Matinée

- Gestion des véhicules (CRUD)

Après-midi

- Gestion des véhicules

Déroulement de la journée

Matinée

- 2h sur la récupération de photo de la base de données.
 - Au lancement de l'application, une liste d'objet Vehicule est initialisée en fonction de se qu'il se trouve dans la base de donnée. Grâce à la méthode LireVehicules()
 - Pour récupérer l'image j'ai créé une méthode ObtenirPhotoGraceldVehicule(), cette méthode va stocker le contenu de la colonne photo de la base de données en tableau d'octets, puis une fois le tableau obtenu on le convertit en image à l'aide de la méthode ByteArrayToImage().
- De 10h30- 10h40 Visioconférence avec M.Folomietow et M. Fontanini, les experts, pour s'assurer que j'ai bien tout compris le cahier des charge et que tout allait bien avec mon environnement de travail.
- Suppression d'un véhicule
 - Création d'un messageBox pour être sûr que l'utilisateur souhaite bien supprimer le véhicule.
 - Création de la méthode SupprimerVehicule(Vehicule vehicule) qui supprime le véhicule de la base de données.

Après-midi

- Création d'une méthode UpdateVehiculesAndCbxVehicules() qui va mettre à jours l'affichage en fonction des modifications que l'on apporte aux véhicules. Elle va régénérer la liste de véhicule, mettre à jour la comboBox, et s'il n'y a aucun véhicule elle va vider les textbox, désactiver certains boutons comme celui pour modifier ou supprimer et enlever les tabPage sauf celle des points d'intérêts. Cette méthode est utile pour éviter la récurrence de code, on l'appelle lors de la suppression d'un véhicule, la modification, l'ajout ou encore au lancement de l'application.
- Modification d'un véhicule
 - Création d'un Windows Form frmModifierVehicule qui est pareil que la form d'ajout de véhicule.

- Lorsqu'on veut modifier un véhicule, la form est rempli automatiquement avec les valeurs déjà enregistrées du véhicule.
- Travail sur quelques détails de design de l'application.
 - Comme l'affichage du nombre de kilomètres initial et réel sur la form principale, en plus de la description et de la photo du véhicule.
 - Activation des bouton modifier et supprimer si aucun véhicule n'est sélectionné.
 - Ajout d'une pictureBox dans la form d'ajout et de modification pour qu'on puisse voir quelle photo nous avons sélectionnée.
 - Suppression des onglets Saisie trajet, Carnet d'entretiens et Gestion d'entretiens lorsqu'aucun véhicule n'est sélectionné. On les rajoute quand un véhicule est sélectionné. Sauf celui des points d'intérêts qui est toujours visible étant donné qu'il ne dépend pas des véhicules.

Bilan

J'ai fini la journée dans les temps malgré le temps qu'il m'a fallu pour récupérer l'image de la base de données. Le programme m'indiquait une erreur qui a fini par disparaître sans rien que je fasse... De même pour l'ajout d'image. J'ai testé à nouveau aujourd'hui que cela fonctionnait toujours bien comme c'était le cas hier mais je ne pouvais pas insérer certaines images. Le problème s'est résolu en même temps que celui pour en récupérer l'image. A part ces soucis mystérieux, j'ai réussi à finir le CRUD des véhicules aujourd'hui comme j'avais indiqué dans le planning prévisionnel.

28 Mai 2020

Objectif

Matinée

- Commencer le CRUD sur la gestion des trajets

Après-midi

- Continuer le CRUD sur la gestion des trajets

Déroulement de la journée

Matinée

- Commencement de la gestion des trajets.
- Ajout des trajets
 - Création de d'un Windows Form frmAjoutTrajet composé d'un lieu de départ, d'arrivée, la distance parcourue et d'un datePicker pour sélectionner la date à laquelle le trajet a été effectué.
 - Création d'une méthode CreerTrajet(string idVehicule, string depart...) pour ajouter le trajet dans la base de données.
 - Création d'une classe Trajet pour plus simplement les afficher dans le dataGridView
 - Chaque trajet s'ajoute dans le dataGridView à l'aide de la méthode LireTrajets(Vehicule vehicule) en lui fournissant comme paramètre le véhicule actuellement sélectionné qui va ensuite retourner une liste de Trajet.
- VisioConférence avec Mme. Terrier pour s'assurer que tout allait bien. J'avais une question sur la manière de procéder pour l'ajout des trajets. Ma question était de savoir s'il fallait utiliser une Map Gmap pour indiquer les trajets puis nous avons conclu que non.
- Suppression de trajet
 - Pour supprimer un trajet j'ai commencé par créer une méthode SupprimerTrajet(string depart, string arrivee...)
 - Je me suis rendu compte plus tard lors de la modification du trajet qu'il était plus propre de faire une méthode nécessitant seulement l'id du trajet comme paramètre pour la suppression de ce dernier.
 - Création d'une méthode UpdateTrajetContent() pour éviter la redondance de code lors de l'ajout, la modification et la suppression d'un trajet. Cette méthode va actualiser le dataGridView avec les trajets se trouvant dans la base de données.

Après-midi

- Modification de trajet
 - Lors du clique sur le bouton modifier je viens chercher dans la ligne du dataGridView cliqué quels sont les valeurs des différents champs (depart, arrivee...)
 - Ensuite, avec ces informations j'ai créé une méthode `ObtenirIdTrajetAvecReste(depart, arrivee...)` pour aller chercher l'id de l'enregistrement qu'il va falloir modifier dans la base de données.
 - L'obtention de cette id m'a permis comme dit plus haut de faire une méthode de suppression de trajet légèrement simplifiée en indiquant juste grâce à l'id l'enregistrement qu'il va falloir supprimer.
 - Création d'un Windows Form `frmModifierTrajet` à qui on va fournir les informations nécessaires pour préremplir les champs avec les valeurs actuellement enregistrées dans la base de données.
 - Création de la méthode `MettreAJourTrajet(depart, arrivee,...)` en fournissant les valeurs entrées précédemment dans `frmModifierTrajet` pour aller modifier la base de données.

Bilan

Réalisation du CRUD une demie journée en avance sur le planning prévisionnel. Il m'a fallu juste revenir un peu sur l'utilisation du dataGridView mais à part ça tout s'est bien déroulé.

29 Mai 2020

Objectif

Matinée

- Finir le CRUD des trajets

Après-midi

- Modifier le kilométrage du véhicule en fonction des trajets
- Faire la validation des entretiens effectués

Déroulement de la journée

Matinée

- Finition sur les form modifier et ajouter trajet
- Trier la dataGridView des trajets par date
- Modifier le kilométrage réel du véhicule en fonction des trajets
 - Ajout d'une autre méthode MettreAJourTrajet() avec comme paramètre à fournir seulement la nouvelle valeur du kmReel et le véhicule actuellement sélectionné.
- Problème rencontré avec la base de données. "DataBasa is locked". J'ai du coup fermé puis rouvert DB Browser puis ça fonctionne à nouveau.
- Création de SupprimerTrajetVehicule(string idVehicule) pour que lors de la suppression d'un véhicule, tous les trajets correspondants au véhicule en question soient supprimés de la base de donné aussi.

Après-midi

- Beaucoup de temps passé sur la compréhension des checkbox dans le dataGridView
- Création de la méthode UpdateCarnetContent() pour mettre à jour le contenu du dataGridView contenant les information sur les maintenances à effectuer.
- Création d'une méthode LireEntretiens(string idVéhicule) qui retourne une liste d'entretien qui sert à lire tous les entretiens contenu dans la base de données en fonction du véhicule actuellement sélectionné.

Bilan

Aujourd'hui, pas de problèmes particulier à part avec la base de données mais on dirait qu'il faut simplement parfois fermer la base de données pour pouvoir lancer le projet, comme si quelquefois elle ne s'enregistrait pas bien.

02 Juin 2020

Objectif

Matinée

- Finir la validation des entretiens

Après-midi

- Commencer la gestion des entretiens

Matinée

- Avancement du manuel technique
 - Introduction
 - Rappel de l'énoncé
 - Organisation
 - Livrables
 - Matériel et logiciels à disposition
 - Description de l'application
 - Énumération des 7 points à respecter qui se trouvent dans le cahier des charges
 - Méthodologie
- De 11h00-11h15 VisioConférence avec Mme. Terrier pour discuter de mon avancement.
- Cocher la checkbox en fonction de si l'entretien en question a été effectué ou non

Après-midi

- Possibilité de cocher/décocher la checkbox grâce à deux méthodes
ObtenirIdEntretienAvecReste(string description, string freqKm,...) qui retourne l'id de la maintenance en question et aussi grâce à la méthode MettreAJourEntretien(string idMaintenance, string fait) qui met à jour dans la base de données le nouvel état de la maintenance. (Si elle a été fait ou pas)
- Ajout de couleur en fonction de l'état de la maintenance
 - Vert si elle a été effectuée
 - Rouge si elle aurait déjà dû être effectuée
 - Orange si elle doit bientôt être effectuée (Dans les 50 prochains km)

Bilan

Cette journée a bien commencée mais vers la fin je cherchais une solution pour un affichage optimal pour le carnet d'entretien. Je n'ai pas encore trouvé la meilleur solution. Car je souhaite afficher les entretiens déjà effectués mais s'il doivent être fait à nouveau je

Luca

souhaite pouvoir les afficher quand même comme des entretiens futur. Donc ils devraient apparaître plus d'une fois dans la liste mais je ne sais pas encore exactement comment faire.

03 Juin 2020

Objectif

Matinée

- Continuer gestion entretiens

Après-midi

- Continuer gestion entretiens

Déroulement de la journée

Matinée

- 9h30-9h40 Visioconférence avec M.Folomietow et M. Fontanini, pour s'assurer que j'avancerais bien et que tout soit en ordre de mon côté.
- Avancement sur les détails de couleurs
- Organisation du dataGridView des carnets dans le bon ordre
- Lorsqu'on coche une maintenance comme étant effectuée, cela crée automatiquement un nouvel entretien futur pour répéter l'entretien effectué à l'aide de sa fréquence en km.
 - Le nouvel entretiens contient comme valeur pour le champs de la date de la dernière maintenance, le jour quand quand l'ancienne maintenance a été coché (faite)
 - Il contient aussi pour le km de la dernière maintenance, le kilométrage réel de la moto au moment où l'ancienne maintenance a été coché (faite).

Après-midi

- Ajout d'un entretien à l'aide d'un bouton dans l'onglet Gestion des entretiens. Lors de la création de l'entretien on doit indiquer une description de l'entretien et aussi la fréquence à laquelle il va devoir être effectué.
 - Lors de l'ajout d'un entretien j'ai du remplir automatiquement en fonction de la date actuelle et du kilométrage du véhicule les champs kmDernierEntretien et dateDernierEntretien. Même s'il s'agit là du premier entretien..
- Dans Gestion des entretiens on peut voir la liste des entretiens avec la description, le kilométrage lors de l'entretien, la fréquence à laquelle on doit effectuer l'entretien et un bouton modifier puis supprimer.
 - Le bouton supprimer peut supprimer n'importe quel entretien du véhicule.
 - J'ai décider d'afficher le kilométrage lors de l'entretien comme ça on peut distinguer les mêmes entretiens due aux fréquences mais effectués à des moments différents sur le véhicule.
 - J'ai décidé aussi de mettre la fréquence parce que c'est ce qu'on peut potentiellement modifier pour un entretien.

- Je n'ai pas voulu mettre toutes les information car cela ferait trop d'information à mon avis pas très utile.

Bilan

Aujourd'hui j'ai rencontré plusieurs problème.

Premièrement, lors de l'ajout d'un nouvel entretien, le nouvel entretien n'est pas censé avoir une date et un kilométrage du dernier entretien. Mais c'est grâce surtout au champ `kmDernierEntretien` que je peux déterminer quand le prochain entretien aura lieu. Donc si je veux que ce soit fait parfaitement et logiquement je dois encore réfléchir à la manière de procéder.

Deuxièmement, lors de la modification d'un entretien je ne trouvais pas logique de pouvoir modifier tous les champs, j'ai donc décidé que l'utilisateur pouvait seulement modifier la description de l'entretien, étant donné que la date et le kilométrage de la dernière maintenance se modifie automatiquement lors de la sélection du `checkBox` dans le carnet. J'hésite encore à faire qu'il soit possible de modifier la fréquence d'un entretien, car pour les anciens entretiens cela perturberait tout à mon avis.

Dernièrement, il est maintenant seulement possible de cocher la `checkbox` et non de la décocher. Car lorsqu'on la coche cela crée automatiquement un nouvel enregistrement de maintenance. Donc si on la décoche puis coche à nouveau il y aura deux enregistrement pareil. Il faut donc peut-être que je supprime l'enregistrement lorsqu'on décoche la `checkBox`, je ne sais pas encore ce qui est mieux.

04 Juin 2020

Objectif

Matinée

- Finir la gestion des entretiens

Après-midi

- Commencer la consultation des points d'intérêts sur la carte

Matinée

- Décision: Je pense ajouter un champ dans la table maintenance de la base de données pour être capable de distinguer quand le premier entretien aura lieu lorsqu'on l'ajoute, de la fréquence de l'entretien. Jusqu'à maintenant, lors d'un ajout, si on mettait 2000 km pour la fréquence, le premier entretien aura lieu dans 2000 km puis le suivant dans 2000 km etc. Mais avec le champ en plus on peut dire que le premier entretien sera dans 1000 km avec une fréquence de 2000 km.
- 11h00-11h30 Visioconférence avec Mme. Terrier. Discussion sur l'affichage et la gestion des maintenances.
 - Décision de ne pas forcément ajouter un champ kmPremiereMaintenance et lors de l'ajout si c'est la première fois qu'on effectue la maintenance la fréquence vaut la même valeur que le kilométrage de la première maintenance. Si elle a déjà été effectuée mais par l'ancien propriétaire par exemple, on peut entre une estimation de la date et le kilométrage de la dernière maintenance.
- Ajout de idMaintenance (idEntretien correspond à la même chose) dans la classe Entretien pour ensuite l'ajouter dans une colonne cachée dans le dataGridView pour le retrouver plus facilement lors de la modification et suppression et de manière plus sur. J'ai ensuite fais la même chose pour le dataGridView du carnet.
- Fini le CRUD des Entretiens

Après-midi

- Affichage de points d'intérêts sur la carte
 - Création d'une classe PointInteret avec un lat, lng, visite et description
 - Création d'une méthode AffichagePointsInterets()
 - Création d'une méthode AjouterMarqueurCarte()
 - Création d'une méthode LirePointsInterets() dans la base de données qui retourne une liste de point d'intérêt
- Ajout de points d'intérêts sur la carte

Bilan

Cette journée c'est bien passé j'ai pu prendre une décision pour la gestion des entretiens et finir le CRUD. J'ai du coup décidé d'ajouter un nouveau champ dans la base de données pour pouvoir gérer le kilométrage du premier entretien, et ensuite la fréquence.

05 Juin 2020

Objectif

Matinée

- Finir la consultation des points d'intérêts
- Commencer l'ajout des points d'intérêts sur la carte

Après-midi

- Finir l'ajout des points d'intérêts sur la carte

Matinée

- Observation de quelques bugs dans l'application.
- 10h00-10h45 Visioconférence avec Mme. Terrier pour répondre à mes questions. Il s'agissait de détails sur le code et sur comment devra être effectué le rendu.
- Travail sur le Manuel Utilisateur.

Après-midi

- Manuel Technique
 - Planification
 - Généralités concernant l'implémentation
 - Base de données
- Manuel Utilisateur fini.

Bilan

La journée ne s'est pas déroulé comme je l'avais noté dans le planning prévisionnel. Aujourd'hui, j'ai énormément avancé sur le manuel Utilisateur et un peu sur le manuel technique.

08 Juin 2020

Objectif

Matinée

- Manuel technique
- Manuel utilisateur
- Tests, résolutions
- Finitions

Après-midi

- Manuel technique
- Manuel utilisateur
- Finitions

Matinée

- Documentation Résumé TPI
- Documentation Manuel Technique
 - Interfaces
 - Liste de fonctionnalités disponibles
 - Présentation de l'interface
 - la fiche frmMain
 - Les différents onglets
 - la fiche frmAjoutVehicule

Après-midi

- Documentation Manuel Technique
 - Interfaces
 - Présentation de l'interface
 - frmModifierVehicule
 - frmAjouterTrajet
 - frmModifierTrajet
 - frmAjoutEntretien
 - frmModifierEntretien
 - Les labels informatifs
 - Les messages d'informations
 - Mesures de sécurité mises en place
 - Généralités concernant l'implémentation
 - Classes

Bilan

Aujourd'hui la journée a été principalement du travail sur la documentation. J'ai profiné un petit peu le code comme les commentaires et les en têtes, mais autrement j'ai beaucoup avancé sur le manuel technique et perfectionné le manuel utilisateur.

08 Juin 2020

Objectif

Matinée

- Manuel technique
- Manuel utilisateur
- Tests, résolutions
- Finitions

Après-midi

- Manuel technique
- Manuel utilisateur
- Finitions

Matinée

- Finitions documentations

Après-midi

- Finition documentations

Bilan