ROAD TRIP

Folco DURAND
Chris LIMONGI
Thibaut TAELMAN

Table des matières

Folco DURAND
Chris LIMONGI
Thibaut TAELMAN

Existant

Road trip est un terme anglais employé pour désigner un voyage effectué sur des routes, quel que soit le nombre d'arrêts. Généralement, un road trip se fait sur de longues distances en automobile. (source Wikipédia)

Folco DURAND
Chris LIMONGI
Thibaut TAELMAN

Contexte

Le but de l'application sera d'afficher le résumer d'un voyage qui va afficher :

- sa durée.
- les destinations.
- Le temps d'escale sur chaque site.
- Distance entre chaque site en km.
- Distance Total.

Folco DURAND
Chris LIMONGI
Thibaut TAELMAN

Architecture applicative (MVVM)

Model-View-ViewModel (MVVM) est un modèle de conception d'applications destiné à découper l'interface utilisateur, le code qui n'est pas lié à l'interface utilisateur.

- Possibilité d'utiliser un style de codage itératif ;
- Simplification des tests unitaires ;
- Meilleure exploitation des outils
- Permet une meilleur collaboration puisque chaque personne va travailler sur des fichier différent.

Page pour interface : MainPage.xaml

Page code : MainPage.xaml.cs

Page Classe :RoadManageer

Folco DURAND
Chris LIMONGI
Thibaut TAELMAN

Couche Métier (BL)

Elle correspond à la partie fonctionnelle de l'application, celle qui implémente la «logique », et qui décrit les opérations que l'application opère sur les données en fonction des requêtes des utilisateurs, effectuées au travers de la couche de présentation.

Nous avons mis en place un script RoadManager.cs qui va représenter notre couche métier.

Les données :

Nous enregistrons chaque étape au moment de la création de la route dans une List global. Pour cela chaque code behind accédé a une instance de ce manager.

Le traitement :

Tous les calculs sont implémenter via des fonctions du manager.

AddLocationToRoad(Location l) : est appeler a chaque création de point par exemple.

Folco DURAND
Chris LIMONGI
Thibaut TAELMAN

Couche d’accès aux données (DAL)

Elle consiste en la partie gérant l’accès aux données du système. Ces données peuvent être propres au système, ou gérées par un autre système. La couche métier n’a pas à s’adapter à ces deux cas.

Les données ne sont pas situé sur le systèmes. Elles sont sur l’API qui contient les coordonnées, les adresses, les villes, les pays et les codes postaux

Folco DURAND
Chris LIMONGI
Thibaut TAELMAN

Couche Présentation

Elle correspond à la partie de l'application visible et interactive avec les utilisateurs. C'est la communication entre l'Homme et L'IHM.

L'application est responsive sur tous les plateforme puisque ces application Windows store.

Pour mettre un point sur le map. Ils n'ont pas besoin de l'adresse , de la ville , du code postal et du pays. Il suffit de cliquer sur la map.

Pour calculer la distance entre deux site différent il suffit de cliquer sur calculer route.

En cliquant sur Erase, nous allons supprimer tous les points et la route.

Folco DURAND
Chris LIMONGI
Thibaut TAELMAN

Intégration API Tierce

Nous allons utiliser L'API Bing Maps pour windows store apps qui combine la puissance de windows 8 et Bing Maps qui va nous permettre d'avoir la Maps du monde et une base de données sur chaque pays, ville, adresse, géolocalisation … Ce qui va nous permettre d'enregistrer la localisation de chaque point indiqué par l'utilisateur. On trouve les éléments suivant :

- **Des contrôles cartographiques**
  - AJAX API en version 6.3 et en version 7
  - Contrôle Silverlight
  - Contrôle Windows Phone – Silverlight
- **Des services**
  - REST Web Services : géocodage, géocodage inverse, itinéraire, imagerie, recherche de proximité et thématique
  - SOAP Web Services : géocodage, géocodage inverse, itinéraire, imagerie, recherche de proximité et thématique
  - Spatial Data Services : géocodage en masse, hébergement et filtre spatial sur les données.

Il faudra installer l'api pour exécuter le code source.

Description des méthode de la classes utiliser DirectionsManager qui contient des méthodes pour calculer et afficher un route sur une map. :

- Bing.Maps.Location :

  Quand un utilisateur clic sur la carte ont récupére la longitude et la latitude et on créer une Location.

- Bing.Maps.Pushpin :
  On créer un marqueur en fonction de la Location.
- Bing.Maps.Directions.WaypointCollection :

  Au moment de crée la route on ajoute chaque Location a la collection de Waypoint.

- Bing.Maps.Directions.DirectionsManager :
  On créer une instance qui va nous permettre d'interagir avec la carte courante et on set son attribue Waypoints grâce à notre collection de waypoints.
- Bing.Maps.Directions.RouteResponse response = await directionsManager.CalculateDirectionsAsync();

  Ce code nous permet d'envoyé une requete a l'api qui va nous renvoyé une route en fonction de nos points. Il suffit ensuite d'invoqué la methode ShowRoutePath avec la reponse de l'api en parametre pour afficher l'itinéraire.

Folco DURAND
Chris LIMONGI
Thibaut TAELMAN

MVVC

Folco DURAND
Chris LIMONGI
Thibaut TAELMAN

Améliorations possibles

Les améliorations possible d’afficher le temps du trajet total avec le nombre de jour sur chaque site.

Permettre de supprimer point par point mise a en place sur la Map.

Améliorer la pertinence des donnée.

Roaming

Fonctionnalité pour ajouter un point par rapport à une adresse

e

Folco DURAND
Chris LIMONGI
Thibaut TAELMAN