**420-485**

**Barème pour le projet**

**La structure de votre projet doit comporter tous les répertoires spécifiques (App\_Data, Controllers, DAL, Images, Models, Views.**

Le projet qui sera corrigé devra comporter la BD localement dans App\_Data

1. **BD first (10%)**

* Modèle de données
* Requêtes Link
* procédures stockées
* Requête SQL
* Le modèle de données devra subir des mises à jour à partir de la base de données.

1. **Architecture** (**20%)**

* MVC
* Patrons Repository et UnitOfwork

Vous serez évalués sur les fonctionnalités suivantes

1. **Gestion des séances (25%)**

On devra être capable de

* ~~Créer une séance~~
* ~~Modifier une séance~~
* D’afficher la liste des séances futures (vous utiliserez une vue partielle)

~~Vous devez prévoir la pagination et le tri selon la date et l’horaire.~~ Une recherche des séances d’un photographe donnée par son nom ou prénom et selon le statut

* Validation des règles d’affaires pour une séance
  + La date de rendez-vous pour une séance de photo doit être comprise entre la date du jour + 1 et date du jour + 15 (incluse)
  + La date pour un rendez-vous doit être unique pour date, horaire, photographe
  + ~~Pour la même date et le même photographe il devra y avoir un minimum de 4h entre 2 rendez-vous~~

1. **Gestion des photos d’une propriété associées à une séance (25%)**

On devra être capable

* d’uploader des photos et de les sauvegarder dans un répertoire que vous devez créer. Le nom du répertoire sera composé par exemple de l’id de la propriété et du nom et prénom de l’agent. Toutes les photos d’une propriété seront dans ce répertoire. Ce répertoire sera dans le répertoire Images du projet. Les chemins d’accès aux fichiers devront être gardés dans la base de données ainsi que le type du fichier.

Un attribut personnalisé permettra de valider que le format du fichier est l’un de cela

jpg, jpeg et png

* de supprimer des photos et d’ajouter des photos. La suppression ne supprime pas le. fichier du répertoire
* d’afficher pour une propriété les informations sur l’agent, le photographe, le prix du forfait et la galerie de photos de la propriété. Pour cela vous devez créer une classe « ViewModel » qui aura des propriétés pour une séance, une propriété, un agent un photographe et les photos de la propriété. Vous devez avoir une vue typée par rapport à cette classe.

1. **Gestion des agents (10%)**

~~On affichera la liste des agents~~. Le détail d’un agent affichera les séances de l’agent de l’année en cours (à partir du 1 janvier). Le total des prix sera affiché.

Vous utiliserez une vue partielle pour l’affichage de la liste des séances.

1. **Exceptions (8%)**

* Gérer les exceptions en générale
* Gérer les accès concurrents pour au moins 2 méthodes d’actions (Edit et Delete)
* Exception de validation
* Exception sur les traitements des fichiers

1. **Vues**

* Razor
* Définir une « layout » ou utiliser celle par défaut du projet. Mais le menu doit permettre d’accéder plus facilement aux « pages ». **(2%)**
* Utiliser au moins vues partielles