**RAPPORT DÉTAILLÉ SUR LE PROJET**

**SUIVI DES VOLS**

**Contexte**

L’application SuiviVols v1.0 est une application web permettant les utilisateurs connectés de consulter les vols programmés et planifiés par les agences de voyages, et ils peuvent réserver ou de modifier leurs vols librement et aisément. Le détail du vol et le calcul total des services offertes sont mises à leur disposition.

Seuls les utilisateurs qui sont authentifiés peuvent entamer le processus de la réservation d’un vol concrètement.

L’application est sécurisée, bien contrôlée, les fuites communs sont fermées, les champs sont validés, et finalement on est sur qu’aucun système n’est sécurisé à 100%.

**L’environnement technique**

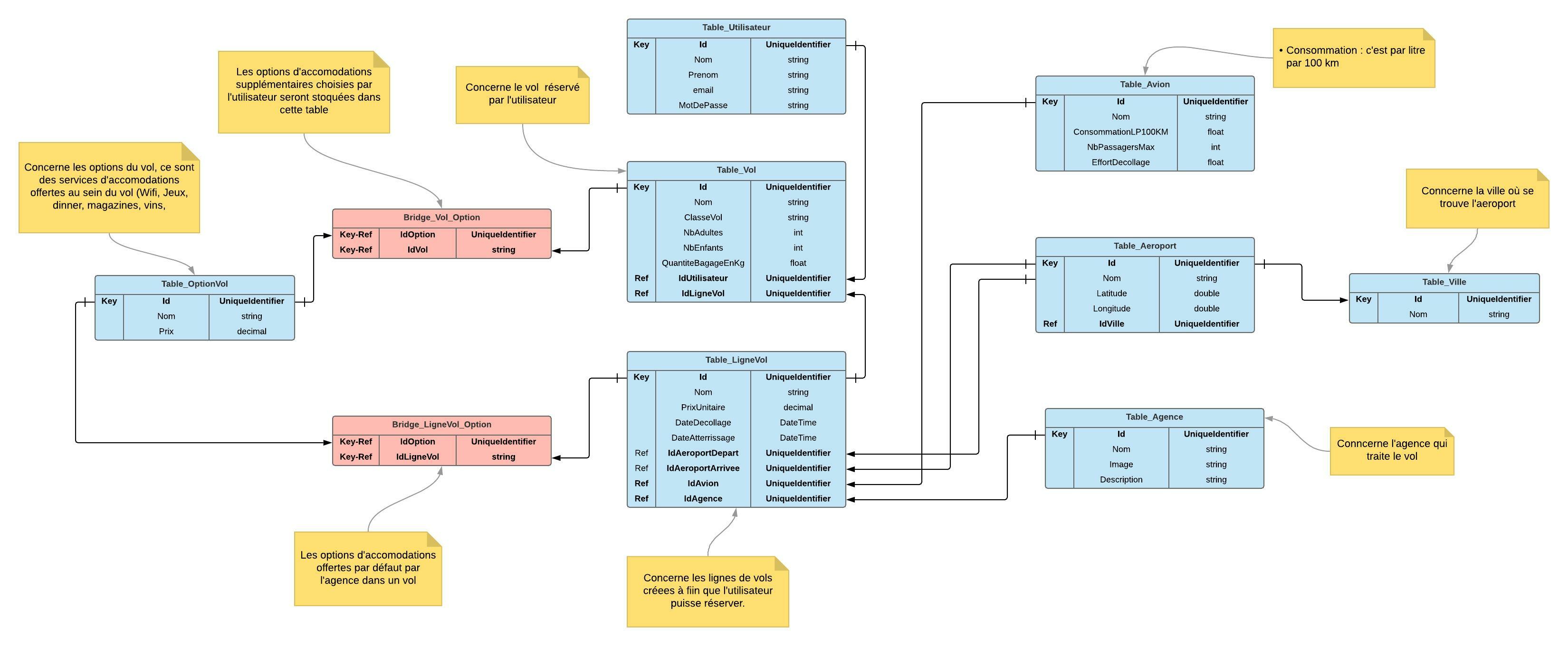
* **Langage de programmation** : c#
* **Technologie utilisée** : Asp.net Core 3.1
* **L’environnement intégré** : Visual Studio 2019
* **Librairie FrontEnd**: Bootstrap pour la responsivité des interfaces graphiques
* **Design pattern utilisé**: MVC
* **Persistance des données** : L’ORM EntityFramework InMemory est mis en place dans cette application pour couvrir le bon déroulement et le fonctionnement des opérations CRUD en toute simplicité et efficacité, Au lieu de faire des listes d’objets en mémoire puis les manipuler, Microsoft a développé l’EntityFramework Core InMemory , c’est un cadriciel pour manipuler les objets mais en mémoire sans l’utilisation d’une base de données réelle, et qui traite une vraie manipulation comme si c’était une base de donnée sur le serveur.

**Architecture Web**

Le projet SuiviVol v1.0 dans sa version Web se compose de différents dossiers/Couches qui mène au bon développement de qualité comme suivant :

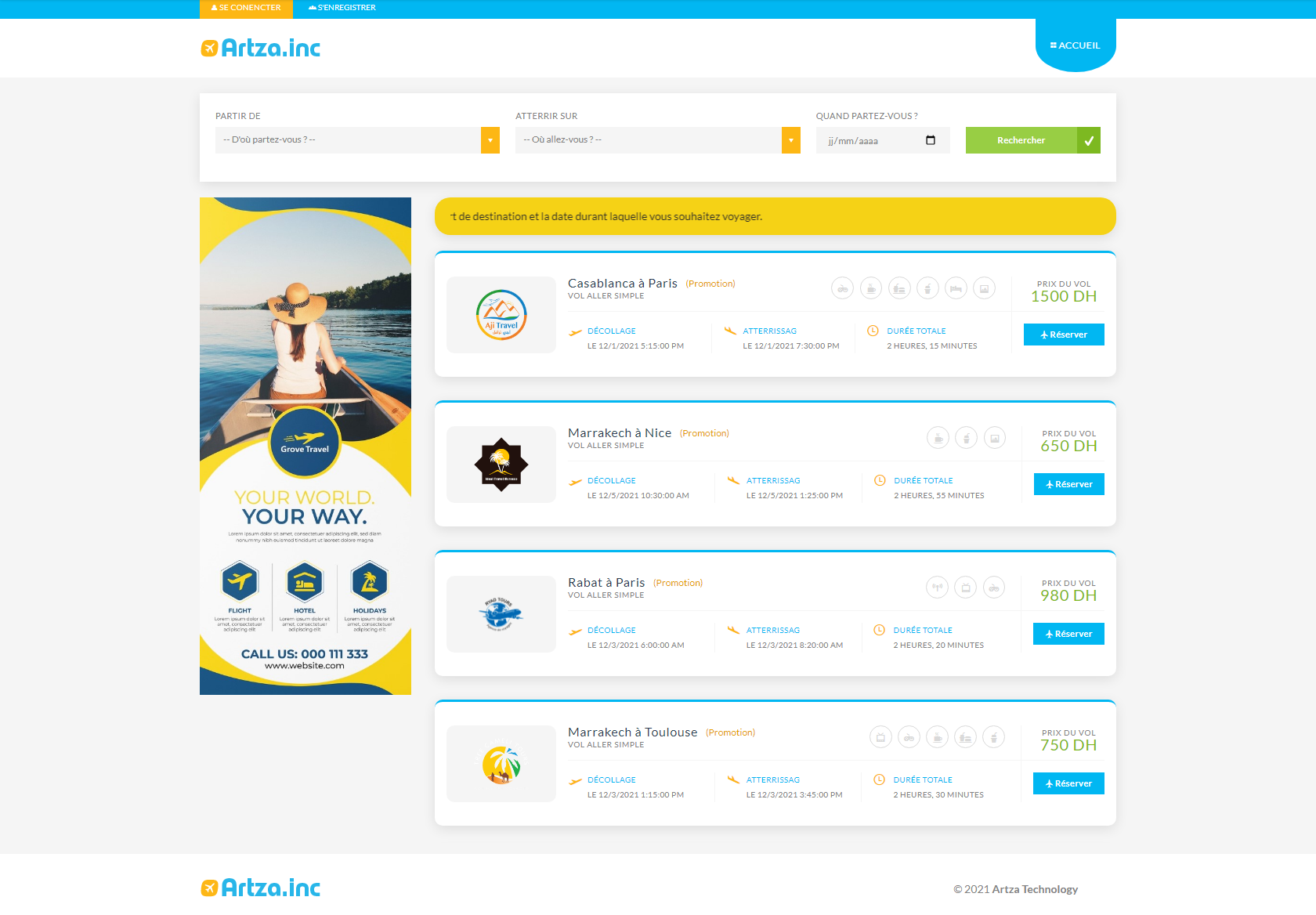
* **Wwwroot** : Contient les fichiers de styles, les bibliothèques javascript, les images, et les scripts personnalisé que j’ai développé pour des cas spécifiques dans le projet.
* **Controllers** : connait l’ensemble des contrôleurs qui prennent en charge le traitement et la synchronisation avec les autres parties prenantes.
* **HomeController** : Ce contrôleur gère tout ce qui est authentification sur l’application webb.
* **FlightsController** : Prend en charge tout ce qui est vols (Affichage de la liste des vols, et la partie de la réservation).
* **BookedFlightsController** : Ce contrôleur géré tous les vols réservés par l’utilisateur ainsi leurs détails calculés.
* **ErrorControllere** : Prend en charge tout ce qui est erreurs, exceptions, fuites de traitement, sont gérées dedans.
* **Models** : Connait l’ensemble des Views modelés pour stocker des informations spécifiques pour les afficher dans des interfaces personnalisées.
* **Views** : Contient l’ensemble des interfaces graphiques, organisées et subdivisé en partial views pour la qualité du code, la réutilisation et également l’optimisation.
* **Layers** : Ce dossier contient toutes les composantes nécessaires pour le développement du projet SuiviVols.
  + **Data** : Ce dossier contient l’ensemble des modèles qui représentent les tables de la base de données.
  + **DatabaseContexts** : Ce dossier contient l’ensemble des databaseContext pour gérer la persistance et la manipulation des données stockées en tant qu’objets modèles. Dans notre cas on a un seul databaseContext puisqu’on manipule une seule base de données virtuelle en mémoire.
  + **Repository**: Prenait en charge la mise en place du design pattern IOC pour le diminuer le couplage fort entre les classes.
    - **Abstractions** : Ce dossier contient les interfaces qui représente un contrat de méthodes à implémenter par les classes qui suivent.
    - **Implements** : Ce dossier contient l’ensemble des classes services qui implémente l’interface repository convenable et qui traite toutes les opérations CRUD de la table générique en question.
  + **Seeders** : C’est un dossier qui contient une classe Seeder qui nous génère au moment du runtime un ensemble des données déjà intégrées et insérées pour les traiter après dans l’application web.

**Conception : Modèle physique de données (MPD)**

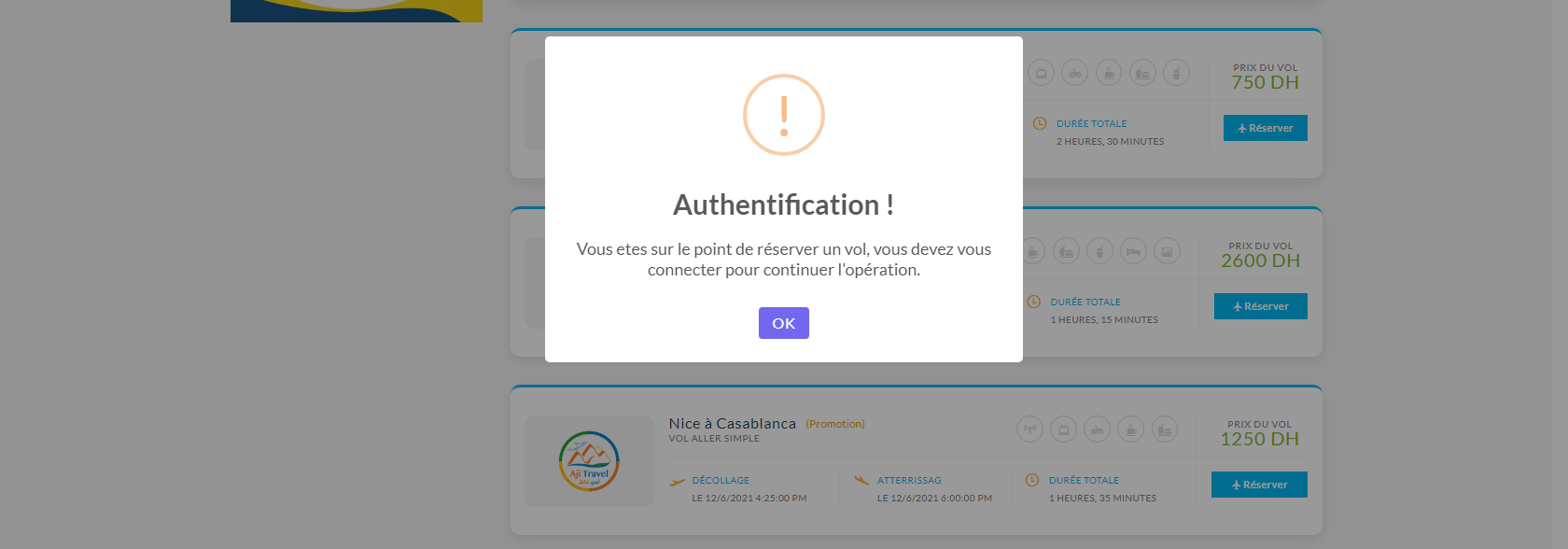


**Interfaces graphiques**

* **Accueil** : Cette page affiche la liste des vols générés dans le runtime, accompagné par un filtre multicritère qui cherche les données et les afficher sans charger la page en utilisant ajax unobtrusive. L’utilisateur peut chercher par n’importe quel critère à la fois, les dates sont définies dans la page d’accueil vous pouvez cherchez par une date qui existe pour voir le fonctionnement de la recherche comment il est.



* **Accueil : Si l’utilisateur voudra réserver un vol sans authentification**



* **Popup de réservation**

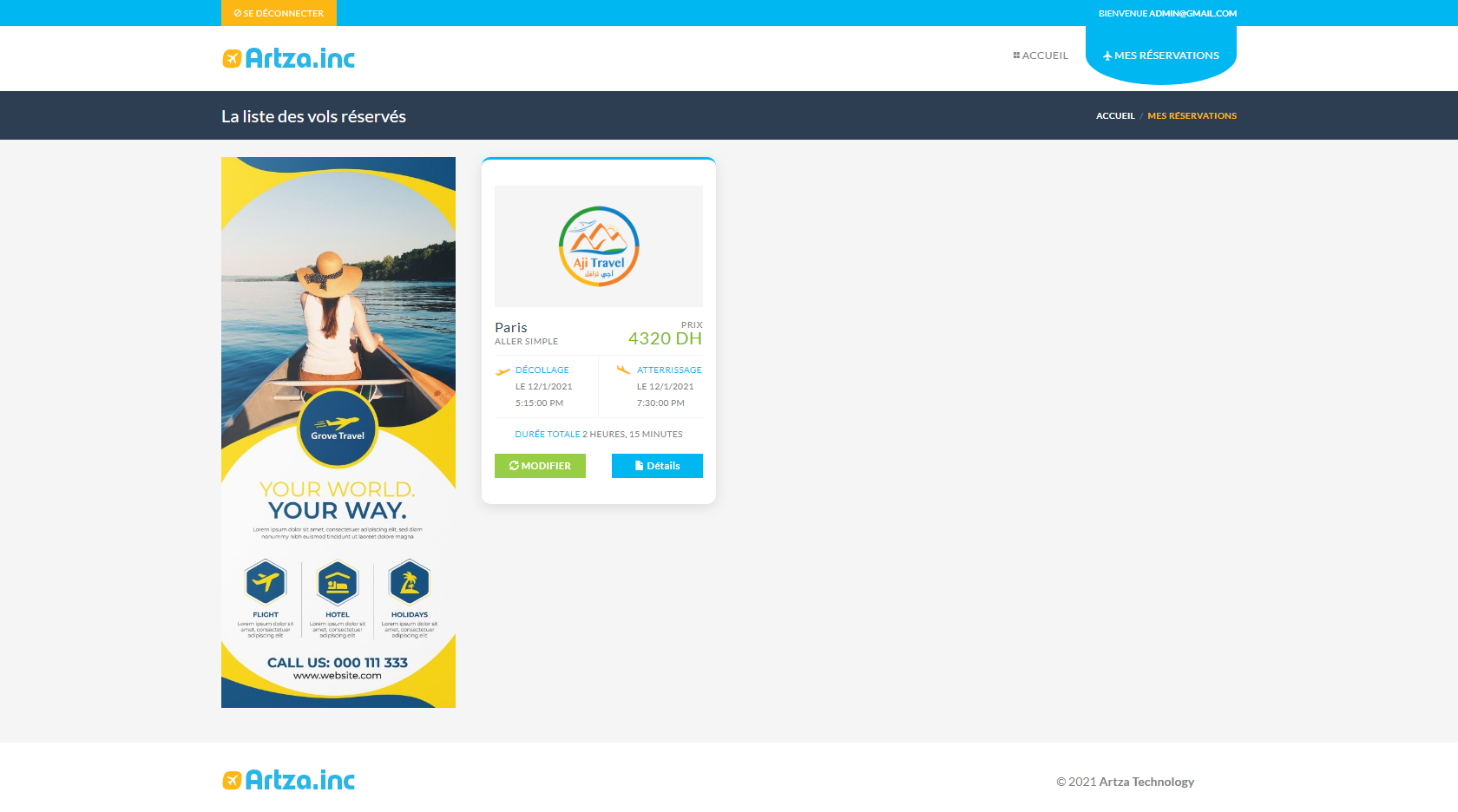
Une image contenant texte, capture d’écran, intérieur

Description générée automatiquement

L’utilisateur peut réserver un vol après l’authentification, il peut sélectionner le type du vol (Economique ou Business class), le nombre des adultes intégrés dans le vol, le nombre d’enfants qu’il possède si existent et la quantité du bagage qui ramène avec lui.  
Le tour des options arrive, l’utilisateur peut sélectionner des options comme des services d’accommodations (Les cercles que vous voyez ce sont des checkboxes à coche), les options qui sont lumineuses se sont des options qui sont cochées dans cette popup, et les autres qui sont sombres se sont des options qui ne sont pas encore cochées, et qui sont offertes par défaut.

**NB** : L’application n’affiche que les options qui ne sont pas offertes par l’agence, et ceux qui sont offertes par l’agence sont cachées car ce n’est pas la peine de les modifier, ce sont par défaut ajoutées.

* **Mes Réservations**



Cette page affiche les réservations que l’utilisateur à créer via la page d’accueil, après qu’il s’est connecté, une page qui s’appelle (Mes réservation) apparaitra pour permettre l’utilisateur de voir ses réservations à fin qu’il puisse le modifier ou de voir le détail complet de ce vol réservé.

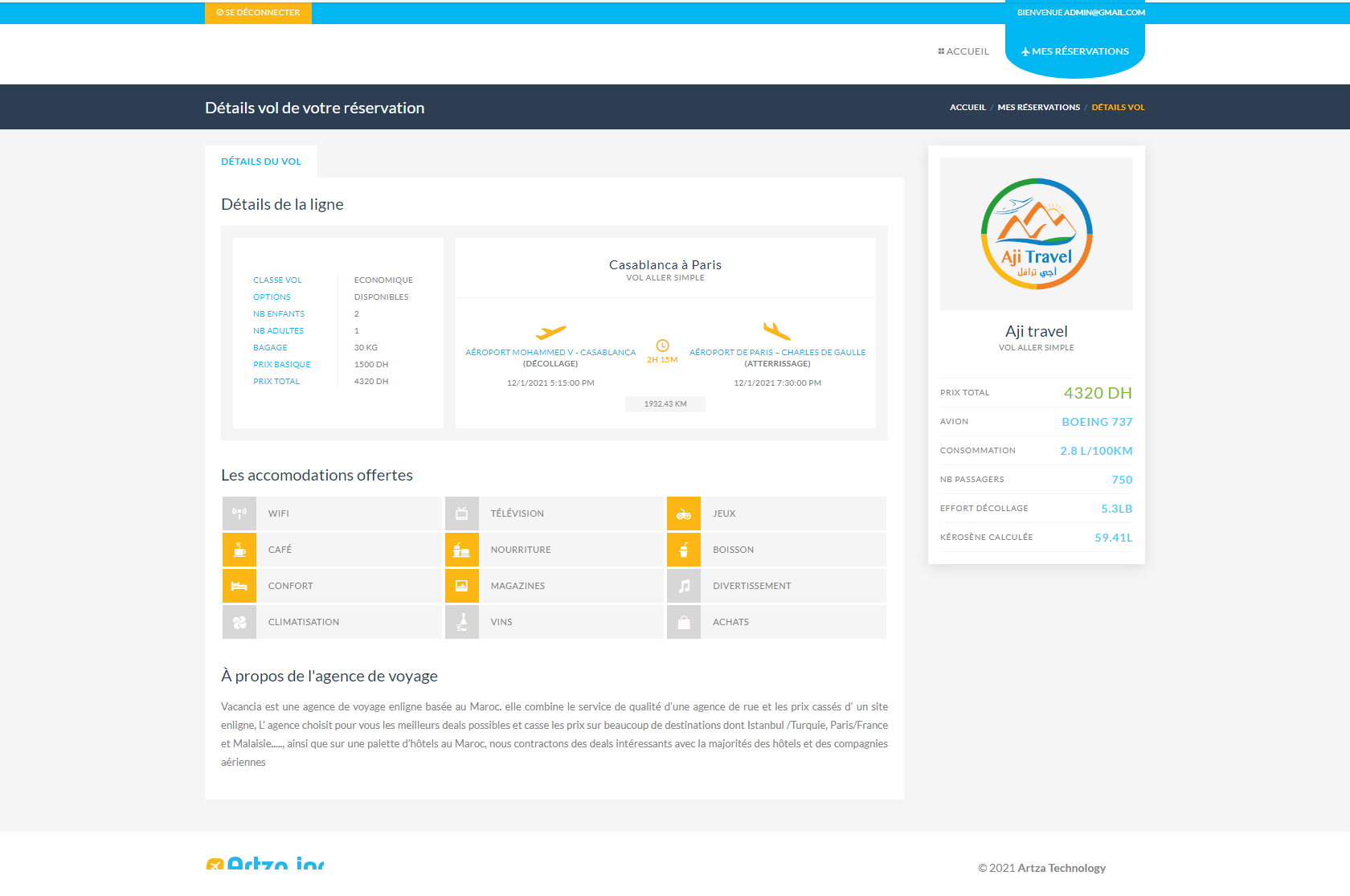
* **Modification d’une réservation d’un vol**  
    
    
    
  L’utilisateur dans ce popup a le droit de modifier son vol réservé, et il peut changer les services d’accommodations, le prix change selon les services sélectionnées ou désélectionnées.

Une image contenant texte

Description générée automatiquement

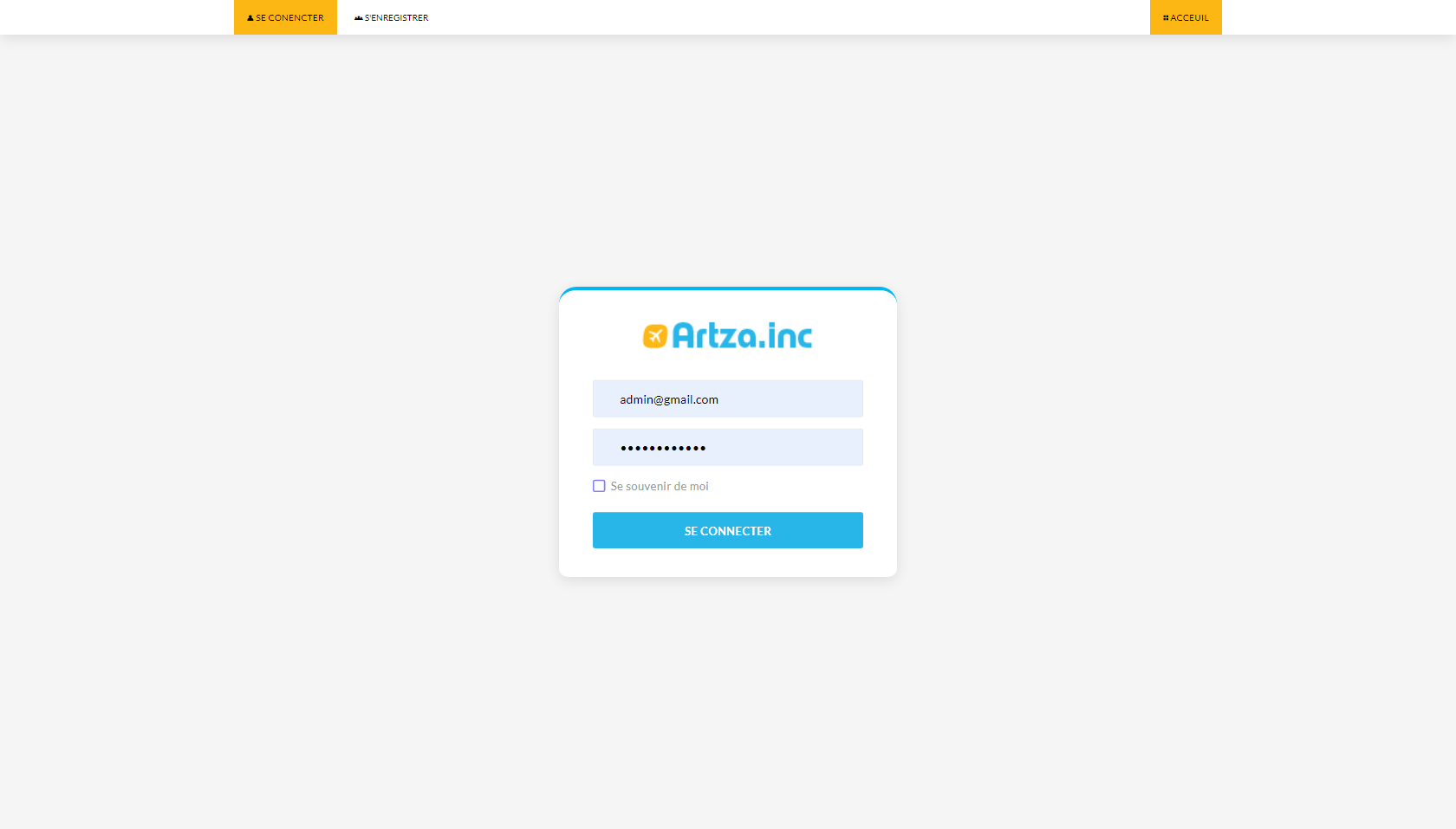
Les services qui sont grisées se sont des options que l’agence offre dans ce vol, du coup l’utilisateur n’a le droit que les voir, par contre, les autre services qu’il a choisit lui-meme il peut les gérer de son role.

* **Détails de la réservation**



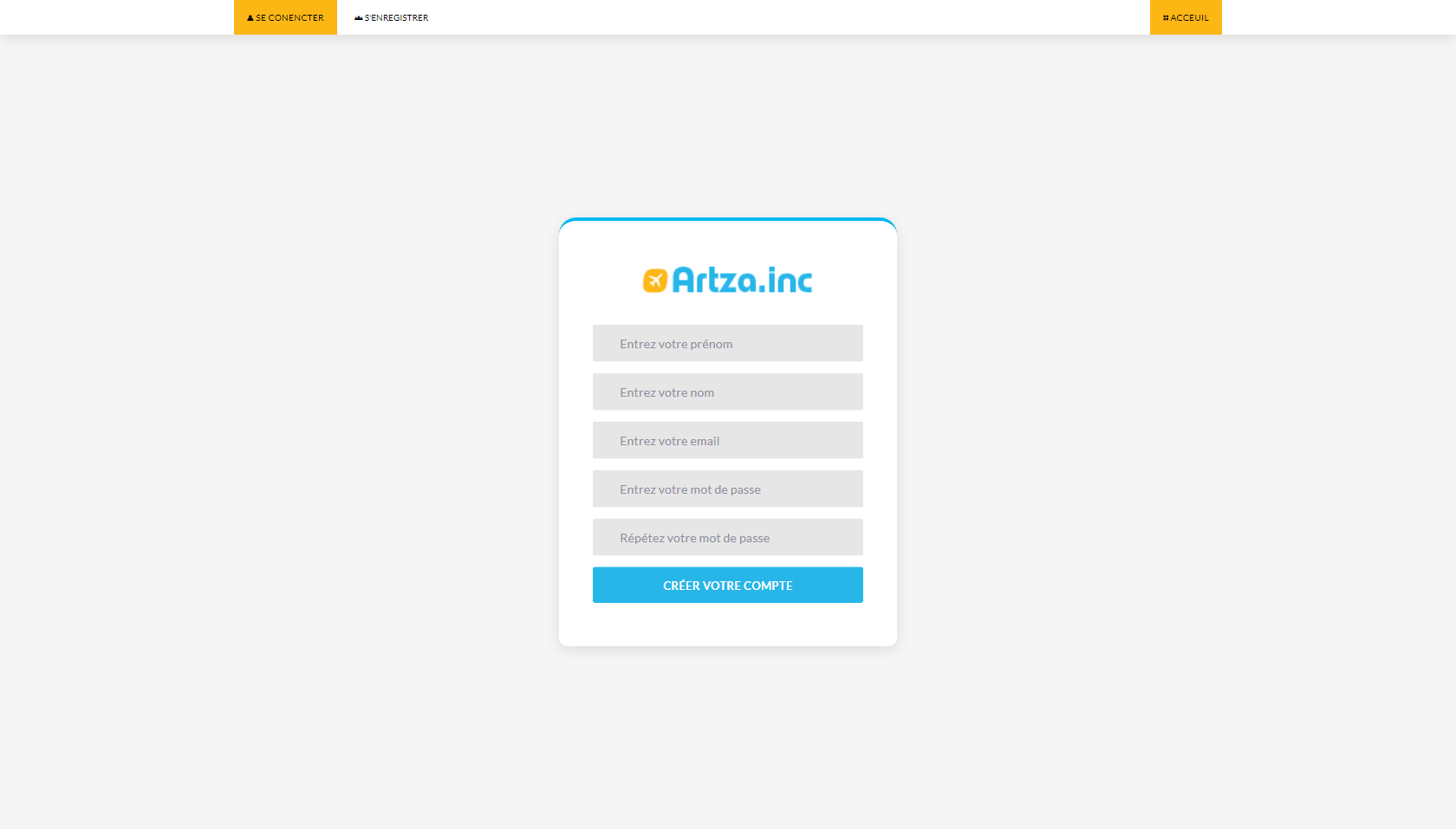
Cette page affiche les informations dynamiques du vol crée, ainsi que les données calculées de l’avion, du vol et du kilométrage comme si demandé.

* **Page d’Authentification**



La page d’authentification permet l’utilisateur de se connecter via l’application web pour réserver, modifier, et suivre son vol librement et aisément.  
  
Les champs sont validés, aucune valeur inacceptable sera rejetée par une erreur affichée sur l’écran de l’utilisateur.

* **Page d’inscription**



La page d’inscription permet l’utilisateur de créer un nouveau compte pour avoir les droits de manipuler les vols qui a réservé.  
  
Les champs sont validés, aucune valeur inacceptable sera rejetée par une erreur affichée sur l’écran de l’utilisateur.