

# **RAPPORT DE STAGE**

Développement application Web

Octobre à Décembre 2020

**Anbar Hamri**

Etudiante en 2eme année cycle ingénieur en Mécatronique d'Automobile

**Encadrant du stage :** Monsieur Hamza Bennani

**Etablissement :** ENSA Kenitra

**Ministère d'accueil :** Ministère de l'Economie des Finances et de la Réforme de l'Administration

# Remerciements

Je tiens à remercier vivement mon encadrant de stage Mr Hamza BENNANI Cadre au sein du service études et développement dans la Direction des Affaires Administratives et Générales DAAG, pour son accueil, son orientation, son suivi et surtout pour le temps accordé à l'aboutissement de ce projet.

Par ailleurs je remercie Mr Ait SALAH Said Chef du service études et développement au sein de la DAAG pour m'avoir permis d'effectuer mon stage au sein de ce groupe.

# Tables des matières

<b>I.Introduction.....</b>	<b>5</b>
<b>II.Analyse de l'existant.....</b>	<b>6</b>
A. Historique et représentation du Ministère.....	6
B. Organigramme du Ministère.....	8
C. Numerisation de l'Administration publique.....	9
D. Cadre du stage.....	9
E. Le dot Net.....	9
F. Outils et compétences informatiques.....	11
<b>III.Analyse des besoins et cahiers des charges.....</b>	<b>12</b>
A. Cahier des charges.....	12
B. Objectifs personnels.....	13
<b>IV.Réalisations.....</b>	<b>14</b>
A. Volet conception.....	14
a) MCD.....	14
b) MLD.....	15

B. Choix de l'architecture.....	24
a) ASP.NET MVC5.....	24
b) Entity Framework6.....	25
c) ADO.NET.....	25
d) Modèle.....	26
e) Vue.....	27
f) Controller.....	27
g) Explorateur de solution.....	28
 C. Programmes développés.....	28
a) Le moteur de vue RAZOR.....	28
b) Bootstrap.....	29
c) WebGrid.....	30
d) Authentification.....	30
e) LINQ .....	30
 <b>V.Présentation du site Web.....</b>	<b>31</b>
<b>VI.Bilan.....</b>	<b>34</b>
A. Bilan des actions réalisées.....	34
B. Problèmes rencontrés.....	34
C. Conclusion.....	34
 <b>VII.Bibliographie.....</b>	<b>35</b>
<b>VIII.Tables des annexes.....</b>	<b>35</b>

# **I .Introduction**

J'ai effectué un stage dans le service études et développement au sein de la division des systèmes d'information de la **DAAG** du Ministère de l'Economie, des Finances et de la Réforme de l'Administration. Sous la responsabilité de Monsieur Bennani Hamza, j'ai pu notamment travailler sur la réalisation d'un site de recrutement en ligne.

Le projet **e-Recrutement** consiste à créer un site web qui permettrait à tout citoyen intéressé de postuler aux postes libres dans l'une des directions du MEFRA. Le site permet, en outre l'inscription aux concours disponibles, le suivi de sa propre candidature jusqu'à l'annonce des résultats.

Bien plus qu'enrichir mes connaissances, ce stage a été pour moi l'occasion d'aiguiser mes techniques de planifications, de modélisation et d'analyse de besoins. Aussi, je me suis intéressée aux enjeux du temps et de sa gestion.

## II. Analyse de l'existant

### A. Historique et représentation du Ministère

Le **Ministère de l'Economie, des Finances et de la Réforme de l'Administration** est le ministère marocain chargé de la politique économique et financière du Maroc. Le siège du ministère est situé à Rabat. Le ministre actuel du ministère est Mohamed Benchaâboun.

Dans le cadre de ses attributions, le Ministère des Finances est en charge des questions financières et monétaires, y compris les politiques des crédits et de finances extérieures.

Par ailleurs il assure la tutelle financière des entreprises et établissements publics ou à participation étatique et, le cas échéant, contribue dans une large mesure à leur transfert au secteur privé, dans les cadres, législatif et réglementaire du processus de privatisation. Plus précisément, les principales missions du Département sont :

#### En matières budgétaire et fiscale :

- Elaboration de la loi de finances et suivi de sa mise en œuvre et de son exécution effective.
- Définition, discussion et élaboration des politiques fiscales et douanière et suivi de leur mise en pratique.
- Recouvrement des recettes publiques.
- Paiement des dépenses publiques.

#### En matière financière :

- Définition des conditions des équilibres financiers interne et externe et élaboration des dispositions et mesures nécessaires à leur réalisation.

- Tutelles réglementaire et technique des établissements et intermédiaires financiers.
- Représentation du Maroc au sein des instances financières régionales et internationales.

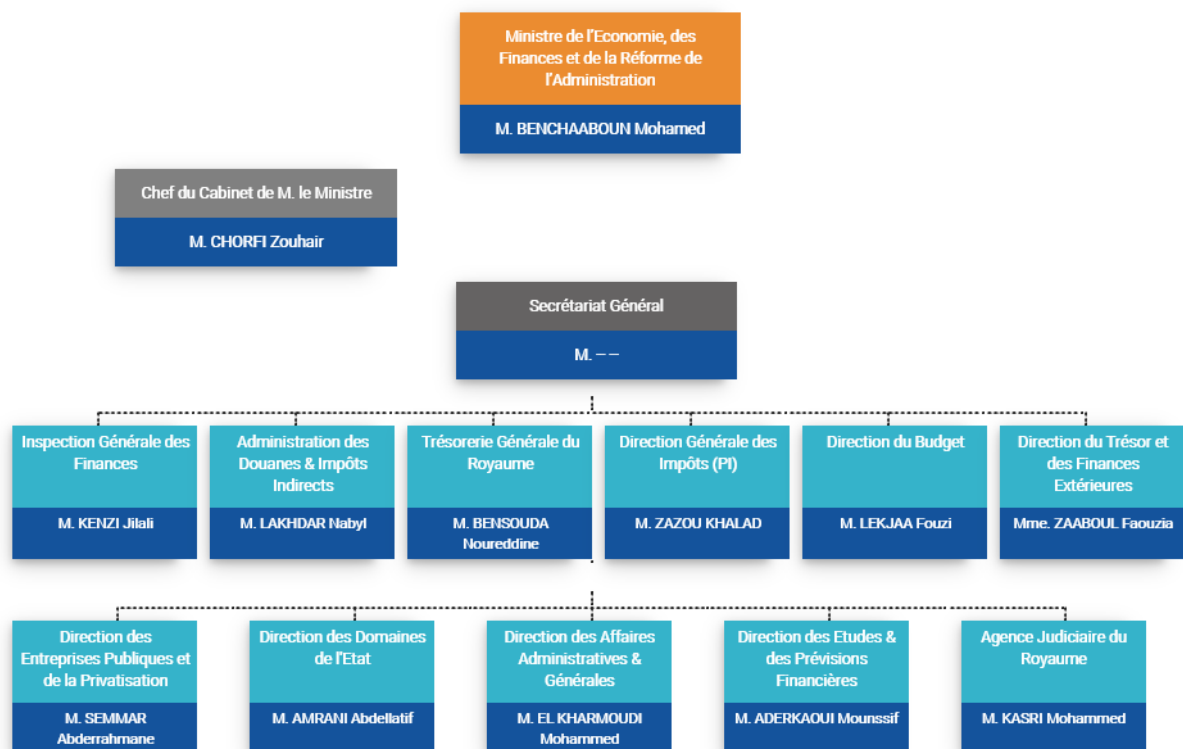
### **Concernant le contrôle :**

- Contrôle des recettes et dépenses publiques.
- Contrôle financier des entreprises et établissements publics et, plus généralement, toute entité bénéficiant d'un concours financier de l'état ou de ses démembrements.
- Contrôle des finances des collectivités territoriales.
- Réglementation et contrôle de l'activité des compagnies d'assurance et de réassurance.

### **En matière de restructuration et de cession d'actifs :**

- Réalisation d'audits des entreprises et établissements publics, en vue notamment de leur restructuration et nationalisation éventuelles.
- Etablissement de la liste des entités à céder ou à privatiser dans le cadre de la loi de finances.
- Elaboration des dossiers de cession et de privatisation accompagnés des avis autorisés dans le cadre d'une large concertation.
- Contribution de façon décisive à la préparation des contrats programmes à conclure avec les établissements publics et à leur mise en œuvre effective.

## B. Organigramme du Ministère





## **C. Numerisation de l'Administration publique**

Actuellement les événements liés au Covid19 et les mesures de confinement ont poussé les états et les entreprises à prendre des mesures sanitaires et changer leur manière de travail. Face à une telle situation, les administrations marocaines ont opté pour des outils de numérisation des services publics particulièrement lorsque les activités ne nécessitent pas une présence physique. Il s'agit en effet d'un moyen d'assurer le respect du principe de la continuité des services publics, suite à la prorogation croissante de COVID 19.

Ainsi, l'Agence de développement du digital ADD a implémenté, en coordination avec le ministère de l'Économie, des Finances et de la Réforme de l'Administration, un ensemble d'outils numériques dans une plateforme de ce dernier afin d'assurer l'accomplissement de ses missions et services. Suivant ce sens, le lancement de l'opération d'inscription en ligne aux concours du MEFRA a été programmé.

Mon sujet de stage a en effet consisté à créer un site qui permet l'accès et la recherche des annonces de postulations aux postes libres du ministère à travers l'inscription au concours associé. Sans oublier la mise à niveau d'information sur l'éventuelle évolution du statut dudit concours ainsi que de sa propre candidature.

## **D. Cadre du stage**

Il s'agit d'un stage effectué dans le cadre du service des études et développement au sein de la division des systèmes d'information de la Direction des Affaires Administratives et Générales. Ce dernier a été sous forme de formation et de supervision distanciée.

## **E. Le dot Net**

.NET est un standard proposé par la société Microsoft pour le développement d'applications d'entreprises multi-niveaux, basées sur des composants.

On parle généralement de «Framework» (traduisez «socle») pour désigner l'ensemble constitué des services (API) offerts et de l'infrastructure d'exécution.

## Applications web ASP.NET

ASP.NET est une infrastructure web open source qui permet de créer des applications web pour des sites web à l'aide des langages HTML, CSS et JavaScript grâce à laquelle il est possible de créer des API web et des sites mobiles.

### C#

Le langage C# a été développé spécifiquement pour la plate-forme .NET. Il s'agit d'un langage orienté objet, dérivé de C++. Il en reprend les majeures caractéristiques, tout en simplifiant son utilisation.

### ASP.NET

ASP.NET c'est le socle de la plateforme de développement de Microsoft pour réaliser des applications web. WebForms et MVC sont deux logiques différentes de développement s'inscrivant au-dessus du socle d'ASP.NET.

ASP.NET se décompose en deux grandes parties à savoir :

**ASP.NET WebForms** : créé en 2002, ce modèle implique une certaine logique de développement. Les applications ASP.NET WebForms se rapprochent d'un modèle événementiel, c'est-à-dire que chaque composant d'une page est capable de réagir à une action de l'utilisateur. Le modèle ASP.NET WebForms permet également de conserver l'état d'une page, chose que le protocole HTTP ne permet pas. Il s'agit d'une logique de développement très puissante pour réaliser des applications web.

**ASP.NET MVC** : Ici le principe n'est pas d'offrir une couche d'abstraction permettant de simplifier le travail, mais plutôt un cadre de réalisation des applications web, grâce à ce patron de conception réputé ; une espèce de ligne directrice qui va nous guider dans la réalisation d'une application web. Mais ASP.NET MVC donne également plus de liberté dans le rendu du HTML en permettant notamment d'utiliser des bibliothèques JavaScript externes afin d'améliorer l'expérience utilisateur et de créer des applications web 2.0 puissantes, grâce à l'AJAX.

## F. Outils et compétences informatiques

La réalisation du projet a nécessité l'installation de 3 logiciels à savoir SQL Server, Visual Studio Code et Power AMC.

### 1. Power AMC



**PowerAMC** est un logiciel de conception créé par la société *SAP*, qui permet de modéliser les traitements informatiques et leurs bases de données associées.

Il permet de travailler avec la méthode Merise afin d'améliorer la modélisation, les processus, le coût et la production d'applications.

Ce logiciel a été utilisé dans la première étape du projet à savoir la création du schéma conceptuel. Grâce à ce dernier, il a été possible de fournir une représentation formelle des données nécessaires à la gestion du site de recrutement. Par la suite, le MCD a été dérivé en modèle physique MPD directement exploitable par un SGBDR. Lors de la génération du MPD, le logiciel spécifie les modalités de mise en œuvre physique d'une base de données. Le MPD est directement dépendant du SGBD cible et permet ainsi de traiter les contraintes relatives à l'accès et au stockage. Power AMC permet aussi d'estimer la taille de la base de données avant son implémentation.

### 2. SQL Server



**Microsoft SQL Server** est un système de gestion de base de données (SGBD) en langage SQL incorporant entre autres un SGBDR « SGBD relationnel » développé et commercialisé par la société Microsoft.

Concrètement, un **SQL server** est un outil qui possède toutes les caractéristiques pour pouvoir accompagner l'utilisateur dans la manipulation, le contrôle, le tri, la

mise à jour, et bien d'autres actions encore, de bases de données grâce au langage SQL.

Après génération du MPD, la génération d'une base de données se fait sur SQL Server. L'implémentation de cette dernière dépend du contenu du fichier MPD édité.

### 3. Visual Studio



Visual Studio est un ensemble complet d'outils de développement permettant de générer des applications web ASP.NET, des services web XML, des applications bureautiques et des applications mobiles. Visual Basic, Visual C++, Visual C# utilisent tous le même environnement de développement intégré (IDE), qui leur permet de

partager des outils et facilite la création de solutions faisant appel à plusieurs langages. Par ailleurs, ces langages permettent de mieux tirer parti des fonctionnalités du framework .NET, qui fournit un accès à des technologies clés simplifiant le développement d'applications web ASP et de services web XML grâce à Visual Web Developer.

**Visual Studio Community** est la version utilisée lors de la réalisation de ce projet. Cette version open source, plus complète que Visual Studio Express, équivaut en matière de fonctionnalités, à Visual Studio Professionnel (version payante).

## III. Analyse des besoins et cahiers des charges

### A. Cahier des charges

En résumé de mon entretien avec la DAAG à propos du site web e-Recrutement, des objectifs pour la création du site ont été fixés :

- L'ensemble des logiciels informatiques requis pour cette mission sont des logiciels libres (Visual Studio, navigateur web). Certains sont de durée limitée (6 semaines pour Power AMC et 6 mois pour SQL Server).
- L'ajout du logo du MEFRA sur la bande de navigation existante comme entête du site web.
- La création d'une partie publique réservée aux intéressés pour rechercher et s'inscrire aux concours et une partie gestionnaire destinée à récupérer les inscriptions, les départager pour évaluer la candidature, ajouter les concours, modifier les statuts de ces derniers ainsi que ceux des inscriptions .

**Les gestionnaires** sont en général sont chargés de recevoir plusieurs inscriptions concernant un concours spécifique, vérifier les informations insérées dans la fiche d'inscription. Pour bien répartir les tâches mais aussi pour assurer l'équité dans la sélection des candidats, on retrouve des hiérarchies dans le rôle des gestionnaires :

**Les administrateurs** du site qui ont le droit d'effectuer des modifications au niveau des statuts du concours, ainsi que l'ajout, l'édition et la suppression des concours.

**Les coordinateurs** ont pour tâche de départager les inscriptions entre l'ensemble des gestionnaires. Ils ont le droit aussi d'établir des changements au niveau des confirmations et refus des inscriptions (revérification du travail des gestionnaires).

- La page d'accueil doit contenir un bloc d'inscription pour les utilisateurs, aussi une partie pour effectuer des recherches sur les concours qui les intéressent. Aussi la page web doit contenir un bloc de connexion pour les gestionnaires.
- Ce travail doit être effectué chez moi en vue de l'épidémie qui impose la délimitation du nombre d'effectifs au sein de l'établissement ainsi que la période du stage qui corrèle avec celle de mes études.

## **B. Objectifs personnels**

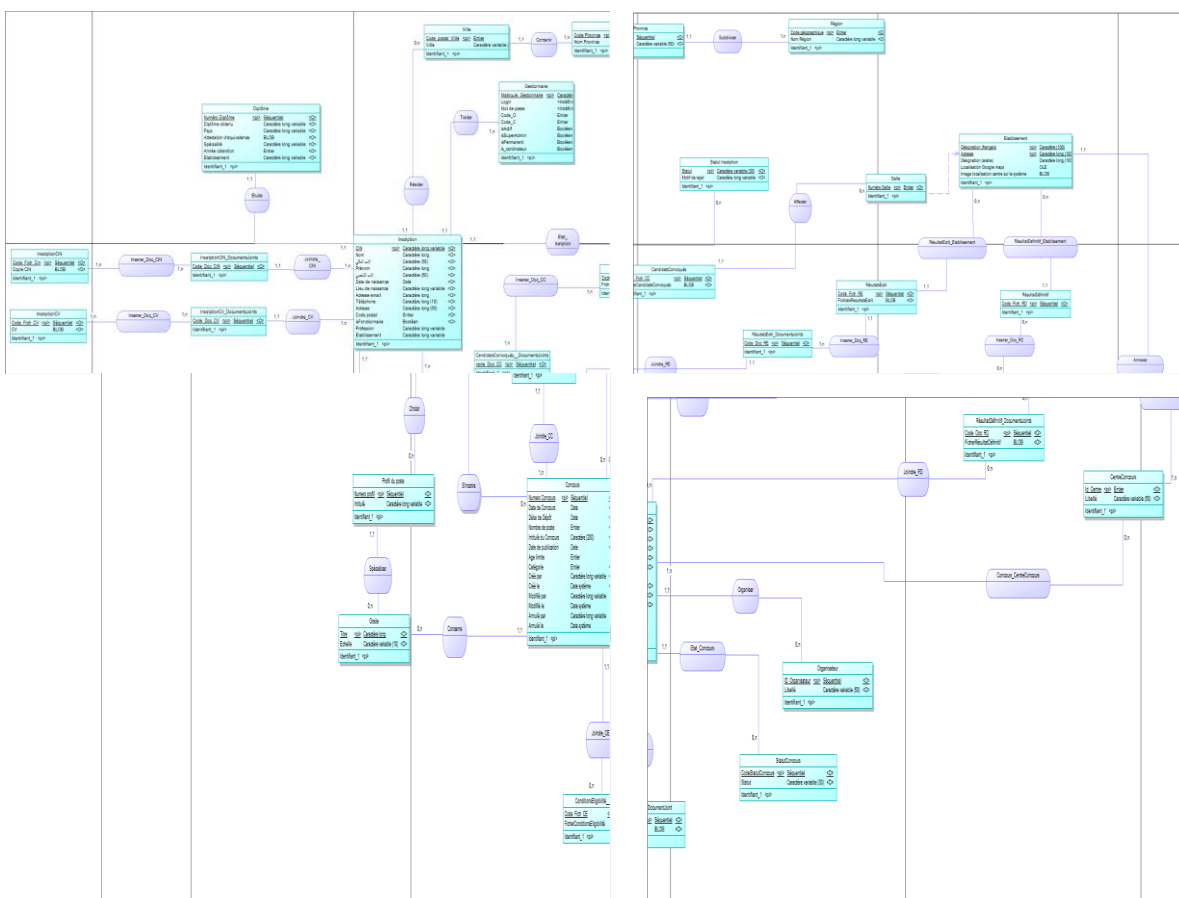
Je souhaite acquérir par le biais de ce stage de nouvelles techniques de travail pour la réalisation d'un site web. En effet, étudier et utiliser pour la première fois le dot Net s'est avéré formateur d'autant plus que dans un milieu professionnel, c'est une technologie très demandée de nos jours.

## IV. Réalisations

### A. Volet Conception

#### a) Modèle Conceptuel de données (MCD)

Sur Power AMC le schéma ci-dessous à été créé :



## b) Modèle Logique de données (MLD)

La génération d'un MLD ainsi qu'un MPD sur Power AMC n'a pas été possible en vue du temps limité offert pour une utilisation non licenciée du logiciel. Afin de créer une base de données sur SQL Server, j'ai procédé directement en écrivant les requêtes SQL suivantes :

### Creation des tables :

```
CREATE DATABASE Projet_eRecrutement;
go
use Projet_eRecrutement;
```

```
CREATE TABLE Inscription(
CIN varchar(7) CONSTRAINT pk_Inscrip PRIMARY KEY,
Nom varchar(20) NOT NULL ,
Prénom varchar(20) NOT NULL ,
date_de_naissance date NOT NULL ,
Lieu_de_naissance varchar(20) NOT NULL ,
Adresse varchar(100) NOT NULL ,
Téléphone varchar(10) NOT NULL ,
Adresse_email varchar(50) NOT NULL ,
Code_postal int NOT NULL ,
isFonctionnaire bit NOT NULL ,
Profession varchar(100) NOT NULL ,
Etablissement varchar(100) NOT NULL
);
```

```
CREATE TABLE Diplôme(
Numero_diplôme int identity(1,1) CONSTRAINT pk_Diplôme PRIMARY KEY,
Diplôme_obtenu varchar(100) NOT NULL,
Spécialité varchar(100) NOT NULL,
Année_obtention int NOT NULL,
Pays varchar(50) NOT NULL,
Attestation_equivalence varbinary(max),
Etablissement varchar(100) NOT NULL
) ;
```

--DOCUMENTS JOINTS A INSCRIPTION-----

```
CREATE TABLE InscriptionCINDocumentsJoins(
Code_Doc_CIN int identity(1,1) CONSTRAINT pk_InscriptionCINDocumentsJoins PRIMARY KEY
);
```

```
CREATE TABLE InscriptionCIN(
Code_Fich_CIN int identity(1,1) CONSTRAINT pk_InscriptionCIN PRIMARY KEY,
Copie_CIN varbinary(max) NOT NULL

);
```

```
CREATE TABLE InserrerDocCIN(
Code_Fich_CIN int references InscriptionCIN(Code_Fich_CIN),
Code_Doc_CIN int references InscriptionCINDocumentsJoints(Code_Doc_CIN),
```

```
CONSTRAINT pk_InserrerDocCIN PRIMARY KEY(Code_Fich_CIN,Code_Doc_CIN)
```

```
);
```

```
CREATE TABLE InscriptionCVDocumentsJoints(
Code_Doc_CV int identity(1,1) CONSTRAINT pk_InscriptionCVDocumentsJoints PRIMARY KEY
);
```

```
CREATE TABLE InscriptionCV(
Code_Fich_CV int identity(1,1) CONSTRAINT pk_InscriptionCV PRIMARY KEY,
Copie_CV varbinary(max) NOT NULL

);
```

```
CREATE TABLE InserrerDocCV(
Code_Fich_CV int references InscriptionCV(Code_Fich_CV),
Code_Doc_CV int references InscriptionCVDocumentsJoints(Code_Doc_CV),
CONSTRAINT pk_InserrerDocCV PRIMARY KEY(Code_Fich_CV,Code_Doc_CV)
```

```
);
```

```
CREATE TABLE Gestionnaire(
Matricule_Gestionnaire int identity(1,1) CONSTRAINT pk_Gestionnaire PRIMARY KEY,
UserName varchar(10) NOT NULL UNIQUE ,
Mot_de_passe varchar(10) NOT NULL,
Code_O varchar(10) NOT NULL,
Code_C varchar(10) NOT NULL,
isActif bit NOT NULL,
isSuperAdmin bit NOT NULL,
isCordinateur bit NOT NULL ,
isPermanent bit NOT NULL
```

```
);
```

```
CREATE TABLE Ville (
Code_posatl_ville int CONSTRAINT pk_Ville PRIMARY KEY,
ville varchar(20) NOT NULL
);
```

```
CREATE TABLE Province (
Code_province int CONSTRAINT pk_Province PRIMARY KEY,
province varchar(20) NOT NULL
```

```
);
```

```
CREATE TABLE Region (
Code_region int CONSTRAINT pk_Region PRIMARY KEY,
region varchar(20) NOT NULL
);
```



```

CREATE TABLE StatutInscription (
Numero_Statut_Inscription int identity(1,1) CONSTRAINT pk_StatutInscription PRIMARY KEY,
Statut varchar(20) NOT NULL,
Motif_de_rejet text

);

CREATE TABLE Etablissement (
Designation_français varchar(50) NOT NULL ,
Designation_arabe varchar(50) NOT NULL,
Adresse varchar(100) NOT NULL ,
Localisation_Google_Maps varbinary(max) NOT NULL,

Image_localisation_centre_sur_système varbinary(max) NOT NULL,
Constraint pk_Etablissement PRIMARY KEY(Designation_français,Adresse)
);

CREATE TABLE CentreConcours (
Id_Centre int identity(1,1) CONSTRAINT pk_CentreConcours PRIMARY KEY,
Libellé varchar(50) NOT NULL

);

CREATE TABLE Concours(
Numero_Concours int identity(1,1) CONSTRAINT pk_Concours PRIMARY KEY,
Date_Concours Date NOT NULL,
Délai_dépôt timestamp NOT NULL,
Nombre_postes int NOT NULL,
Intitulé_Concours varchar(50) NOT NULL,
Date_Publication date NOT NULL,
Age_limite int NOT NULL,
Catégorie char NOT NULL,
Créé_par varchar(100) ,
Créé_le date ,

Modifié_par varchar(100) ,
Modifié_le date ,
Annulé_par varchar(100) ,
Annulé_le date

);

CREATE TABLE ConcoursCentreConcours (
Numero_Concours int references Concours(Numero_Concours),
Id_Centre int references CentreConcours(Id_Centre),
CONSTRAINT pk_ConcoursCentreConcours PRIMARY KEY(Numero_Concours,Id_Centre)

);

CREATE TABLE Inscrire(
Numero_Concours int references Concours(Numero_Concours),
CIN varchar(7) references Inscription(CIN),
CONSTRAINT pk_Inscrire PRIMARY KEY(Numero_Concours,CIN)

) ;

CREATE TABLE Organisateur(
Id_Organisateur int identity(1,1) CONSTRAINT pk_Organisateur PRIMARY KEY,
Libellé varchar(100) NOT NULL
);

```

```

CREATE TABLE StatutConcours (
Numero_Statut_Concours int identity(1,1) CONSTRAINT pk_StatutConcours PRIMARY KEY,
Statut varchar(20) NOT NULL,

);

--DOCUMENTS JOINTS A CONCOURS-----

CREATE TABLE ConditionsEligibilitéDocumentsJoints(
Code_Fich_CE int identity(1,1) CONSTRAINT pk_ConditionsEligibilitéDocumentsJoints PRIMARY KEY,

Fichier_CE varbinary(max) NOT NULL
);

-----

CREATE TABLE CandidatsConvoquésDocumentsJoints(
Code_Doc_CC int identity(1,1) CONSTRAINT pk_CandidatsConvoquésDocumentsJoints PRIMARY KEY
);

CREATE TABLE ConcoursCandidatsConvoqués(
Code_Fich_CC int identity(1,1) CONSTRAINT pk_ConcoursCandidatsConvoqués PRIMARY KEY,
Fichier varbinary(max) NOT NULL

);

CREATE TABLE InserrerDocCC(
Code_Fich_CC int references ConcoursCandidatsConvoqués(Code_Fich_CC),
Code_Doc_CC int references CandidatsConvoquésDocumentsJoints(Code_Doc_CC),
CONSTRAINT pk_InserirDocCC PRIMARY KEY(Code_Fich_CC,Code_Doc_CC)
);

-----

CREATE TABLE ResultatsEcritDocumentsJoints(
Code_Doc_RE int identity(1,1) CONSTRAINT pk_ResultatsEcritDocumentsJoints PRIMARY KEY
);

CREATE TABLE ConcoursResultatsEcrit(
Code_Fich_RE int identity(1,1) CONSTRAINT pk_ResultatsEcrit PRIMARY KEY,
Fichier varbinary(max) NOT NULL

);

CREATE TABLE InserrerDocRE(
Code_Fich_RE int references ConcoursResultatsEcrit(Code_Fich_RE),
Code_Doc_RE int references ResultatsEcritDocumentsJoints(Code_Doc_RE),
CONSTRAINT pk_InserirDocRE PRIMARY KEY(Code_Fich_RE,Code_Doc_RE)
);

-----

CREATE TABLE ResultatsDefinitifDocumentsJoints(
Code_Doc_RD int identity(1,1) CONSTRAINT pk_ResultatsDefinitifDocumentsJoints PRIMARY KEY
);

CREATE TABLE ConcoursResultatsDefinitif(
Code_Fich_RD int identity(1,1) CONSTRAINT pk_ResultatsDefinitif PRIMARY KEY,
Fichier varbinary(max) NOT NULL

);

```

```
CREATE TABLE InserrerDocRD(
Code_Fich_RD int references ConcoursResultatsDefinitif(Code_Fich_RD),
Code_Doc_RD int references ResultatsDefinitifDocumentsJoints(Code_Doc_RD),
CONSTRAINT pk_InserrerDocRD PRIMARY KEY(Code_Fich_RD,Code_Doc_RD)

);
```

```
-----
CREATE TABLE Grade(
Titre varchar(50) CONSTRAINT pk_Grade PRIMARY KEY,
Echelle varchar(10)

);
```

```
CREATE TABLE ProfilDuPoste(
Numero_Profil int identity(1,1) CONSTRAINT pk_ProfilDuPoste PRIMARY KEY,
Intitulé varchar(100) NOT NULL

);
```

```
CREATE TABLE Salle(
Designation_français varchar(50) NOT NULL ,
Adresse varchar(100) NOT NULL,
Numero_Salle int NOT NULL,
CONSTRAINT pk_Salle PRIMARY KEY(Numero_Salle,Designation_français,Adresse),
CONSTRAINT fk_Etab FOREIGN KEY(Designation_français,Adresse) references
Etablissement(Designation_français,Adresse)

);
```

## Associations entre tables(Clés étrangères)

```
ALTER TABLE Inscription ADD
```

```
Code_postal_ville int NOT NULL CONSTRAINT fk_InscripVille FOREIGN KEY(Code_postal_ville) references
Ville(Code_postal_ville),
```

```
Matricule_Gestionnaire int NOT NULL CONSTRAINT fk_InscripGestionnaire FOREIGN
KEY(Matricule_Gestionnaire) references Gestionnaire(Matricule_Gestionnaire) ,
```

```
Numero_Profil int NOT NULL CONSTRAINT fk_InscripProfil FOREIGN KEY(Numero_Profil) references
ProfilDuPoste(Numero_Profil),
```

```
Numero_Statut_Inscription int NOT NULL CONSTRAINT fk_InscripStatutInscription FOREIGN
KEY(Numero_Statut_Inscription) references StatutInscription(Numero_Statut_Inscription),
```

```
Numero_diplôme int NOT NULL Constraint fk_Diplôme FOREIGN KEY references Diplôme(Numero_diplôme) ;
```

```
-----
ALTER TABLE InscriptionCINDocumentsJoints ADD
```

```
CIN varchar(7) NOT NULL CONSTRAINT fk_inscripCIN FOREIGN KEY(CIN) references Inscription(CIN) ;
```

```
-----
ALTER TABLE InscriptionCVDDocumentsJoints ADD
```

```
CIN varchar(7) NOT NULL CONSTRAINT fk_inscripCV FOREIGN KEY(CIN) references Inscription(CIN) ;
```

```

ALTER TABLE CandidatsConvoquésDocumentsJoints ADD
Numero_Concours int NOT NULL CONSTRAINT fk_ConcoursCC FOREIGN KEY(Numero_Concours) references
Concours(Numero_Concours) ;

```

```

ALTER TABLE ResultatsEcritDocumentsJoints ADD
Numero_Concours int NOT NULL CONSTRAINT fk_ConcoursRE FOREIGN KEY(Numero_Concours) references
Concours(Numero_Concours) ;

```

```

ALTER TABLE ResultatsDefinitifDocumentsJoints ADD

```

```

Numero_Concours int NOT NULL CONSTRAINT fk_ConcoursRD FOREIGN KEY(Numero_Concours) references
Concours(Numero_Concours) ;

```

```

ALTER TABLE ConcoursResultatsDefinitif ADD
Code_Etablissement int NOT NULL CONSTRAINT fk_EtabRD FOREIGN KEY(Code_Etablissement) references
Etablissement(Code_Etablissement) ;

```

```

ALTER TABLE ConcoursResultatsEcrit ADD
Code_Etablissement int NOT NULL CONSTRAINT fk_EtabRE FOREIGN KEY(Code_Etablissement) references
Etablissement(Code_Etablissement) ;

```

```

ALTER TABLE Ville ADD
Code_province int NOT NULL CONSTRAINT fk_Province FOREIGN KEY(Code_province) references
Province(Code_province) ;

```

```

ALTER TABLE Province ADD
Code_region int NOT NULL CONSTRAINT fk_Region FOREIGN KEY references Region(Code_region) ;

```

```

ALTER TABLE Etablissement ADD
Id_Centre int NOT NULL CONSTRAINT fk_Centre FOREIGN KEY references CentreConcours(Id_Centre) ;

```

```

ALTER TABLE Concours ADD
Id_Organisateur int NOT NULL CONSTRAINT fk_Organisateur FOREIGN KEY references
Organisateur(Id_Organisateur),

```

```

Titre varchar(50) NOT NULL CONSTRAINT fk_GradeConcours FOREIGN KEY references Grade(Titre),
Code_Fich_CE int NOT NULL CONSTRAINT fk_CE FOREIGN KEY references
ConditionsEligibilitéDocumentsJoints(Code_Fich_CE),
Numero_Statut_Concours int NOT NULL CONSTRAINT fk_StatutConcours FOREIGN KEY references
StatutConcours(Numero_Statut_Concours);

```

```

ALTER TABLE ProfilDuPoste ADD

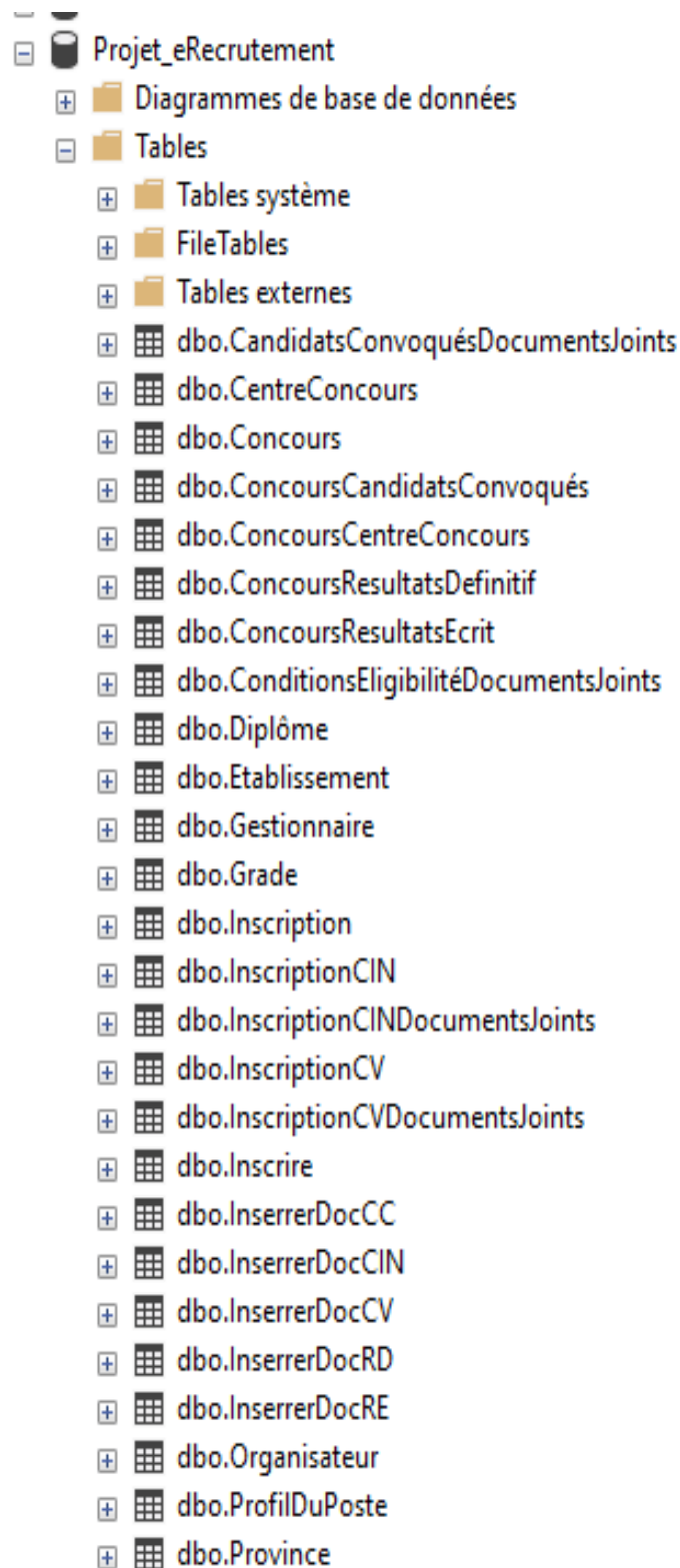
```

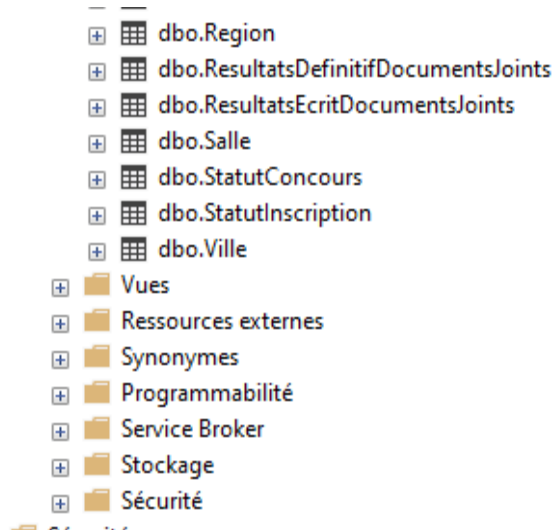
```

Titre varchar(50) NOT NULL CONSTRAINT fk_GradeProfil FOREIGN KEY references Grade(Titre);

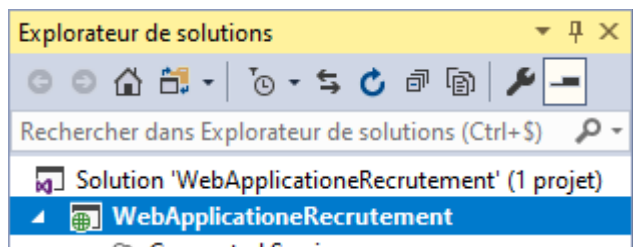
```

Après execution, le résultat ci-dessous s'affiche :

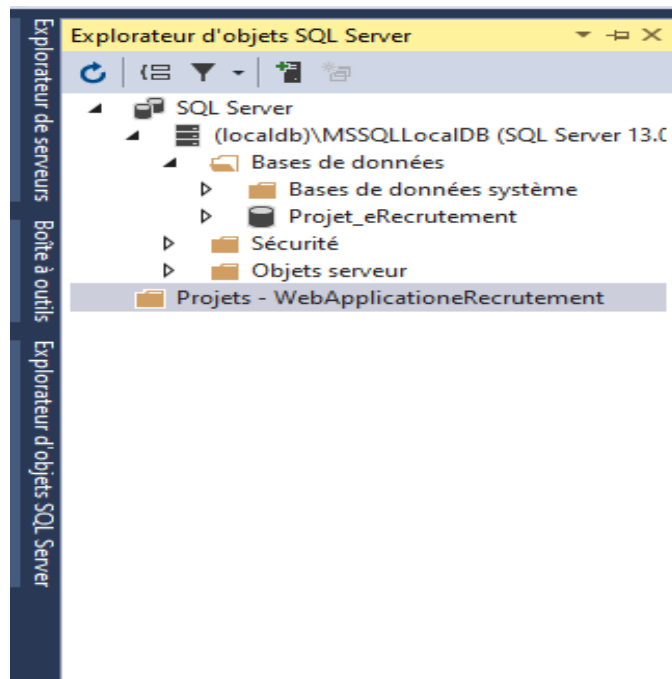




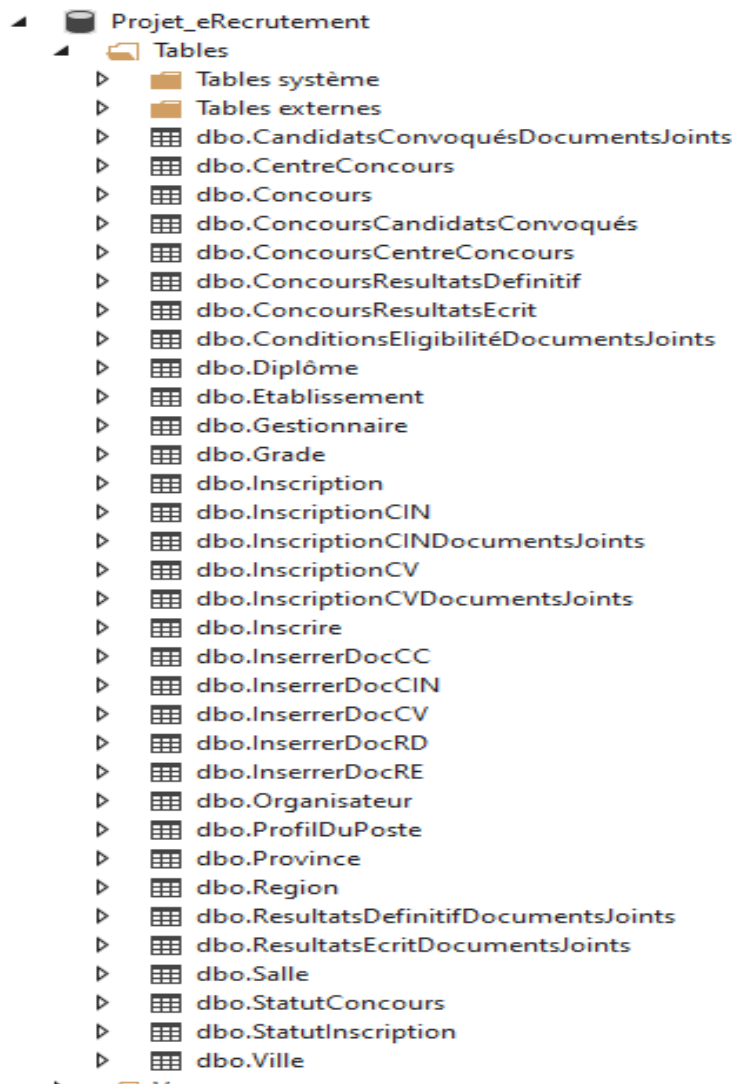
Sur Visual Studio, On crée le projet suivant **WebApplicationRecrutement** :



L'outil « **Explorateur d'objet SQL Server** », on remarque que la connexion avec la base de données SQL Server a été établie :



La base de données « **Projet-eRecrutement** » destinée au site apparait :



## B. Choix de l'architecture

### a. ASP.NET MVC 5

Le choix du socle de développement de l'application s'est porté sur ASP.NET MVC car il m'offre la possibilité d'aiguiser mes connaissances basiques en HTML et découvrir de nouvelles particularités langages tels que AJAX et C#. Vous aurez quelques particularités spécifiques à apprendre, mais la logique sera globalement la même. La maîtrise du HTML généré est m'a également permis de créer un site qui fonctionne

sur des PC et des tablettes ou des Smartphones. Le « responsive design », consiste à adapter un site web au dispositif de support (Smartphone, tablette, ordinateur etc.), implique d'avoir un bon contrôle sur le HTML généré, ce qui est plus compliqué à faire avec ASP.NET WebForms. Ceci implique, par contre, d'écrire soi-même tout le HTML nécessaire.



## b. Entity Framework 6

Entity Framework (EF) est un Framework ORM open source pour .les applications NET prises en charge par Microsoft. Il simplifie la correspondance entre les objets du logiciel et les tables et colonnes d'une base de données relationnelle.

Il permet aussi de travailler avec des données en utilisant des objets de classes spécifiques au domaine sans se concentrer sur les tables et colonnes de base de données sous-jacentes où ces données sont stockées.

Il prend soin de créer des connexions de base de données et d'exécuter des commandes, ainsi que de prendre les résultats des requêtes et de les matérialiser automatiquement en tant qu'objets dans l'application.

Il permet également de garder une trace des modifications apportées à ces objets. Après actualisation, les modifications seront transmises à la base de données.

## c. ADO.NET

Il s'agit d'un module (ensemble de classes) dans .NET Framework exposant les services d'accès et de gestion de données situées sur une base de données relationnelle (SQL Server, MySQL...) ou non SQL constitue de deux composants : un premier pour la connexion et un second pour la gestion. Dans ce projet j'ai été amené à utiliser quelques classes qu'offre ce module. Entre autres j'ai utilisé :

**Connection** : pour assurer la connectivité avec une source des données.

**Command** : pour définir des requêtes SQL afin de lire ou écrire des données, ou exécuter des procédures stockées.

**DataAdapter** : afin d'utiliser la requête définie dans un objet Command pour charger le DataSet avec des données.

**DataSet** : c'est un tableau conçu pour accueillir des données venant d'une source de données relationnelle.

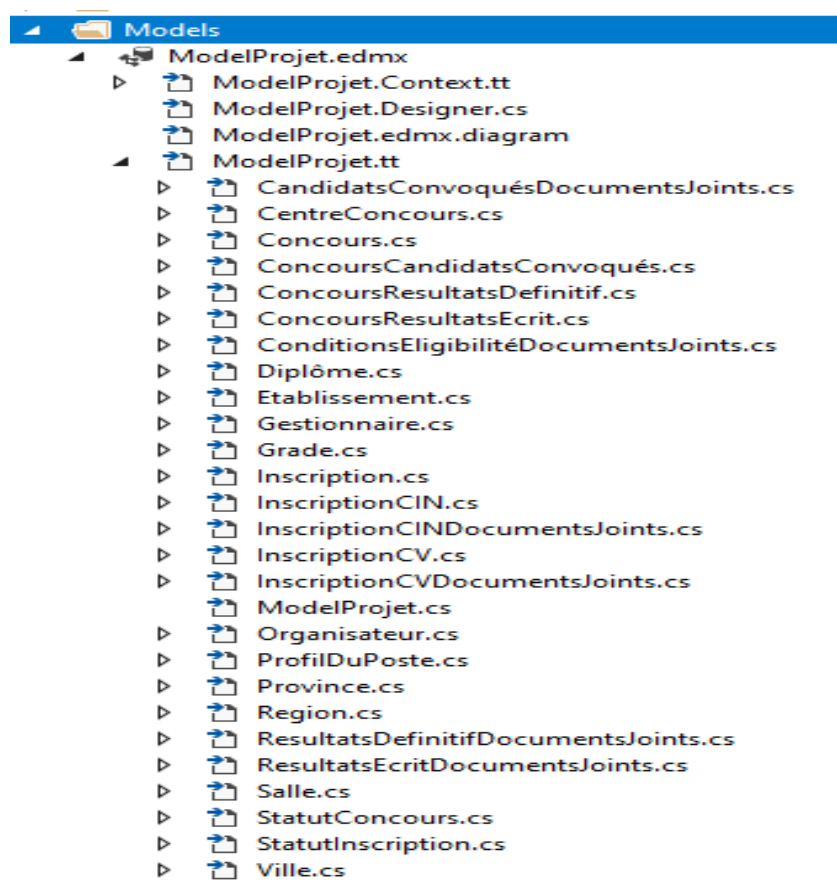
Le fonctionnement est assuré par l'ajout de la référence MySql.Data.

## d. Modèle

Cette partie gère les données du site. Son rôle est d'aller récupérer les informations « brutes » dans la base de données, de les organiser et de les assembler pour qu'elles puissent ensuite être traitées par le contrôleur. On y trouve donc les requêtes SQL. Il s'agit d'un moyen de transmission des données ainsi que de leur organisation enregistrées dans la base de données SQL Server à Visual Studio afin de pouvoir les manipuler.

On retrouve alors chaque table dans la base de données transmise sous forme de classe dans un fichier code extension \*.cs .

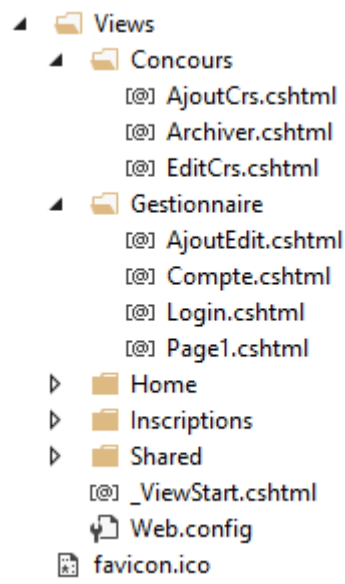
Le fichier extension \*.edmx contient le schéma conceptuel des données apportés de Power AMC à SQL Server puis à Visual Studio.



## e. View

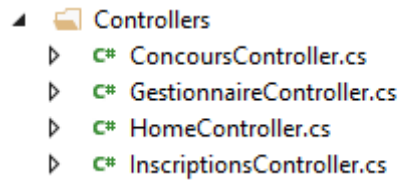
Cette partie se concentre sur l'affichage. Elle se contente de récupérer des variables pour savoir ce qu'elle doit afficher. On y trouve essentiellement du code HTML mais aussi quelques boucles et conditions PHP très simples, pour afficher par exemple une liste de messages.

Les vues permettent en outre la manipulation de l'affichage de la page web. Elles incluent les méthodes que l'on souhaite montrer dans le cas de saisies d'informations par exemple. Elles sont alors, toujours liées à un Controller qui contient les méthodes de manipulations d'un ou plusieurs modèles. On retrouve alors les vues dans des dossiers de même noms que les Controllers concernés sous forme de fichiers extension \*.cshtml.



## f. Controller

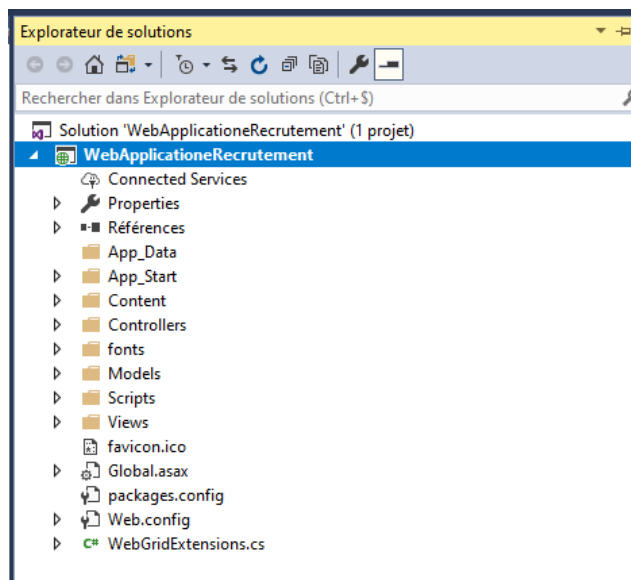
Cette partie gère la logique du code qui prend des décisions. C'est en quelque sorte l'intermédiaire entre le modèle et la vue : le contrôleur va demander au modèle les données, les analyser, prendre des décisions et renvoyer le texte à afficher à la vue. Le contrôleur contient exclusivement du C#. C'est notamment lui qui détermine si le visiteur a le droit de voir la page ou non (gestion des droits d'accès).



## g. Explorateur de solutions

Cet outil permet d'afficher tous les dossiers utilisés dans la création du site ainsi que la modification des propriétés qui les caractérisent.

Il offre aussi une vue organisée des fichiers du projet ainsi qu'un accès aux commandes qui s'y rapportent.



## C. Programmes développés

### a) Le moteur de vue RAZOR

Employée dans les Vues, la syntaxe Razor permet d'avoir un bloc C# au même endroit que des balises html. Celle-ci a été conçue pour faciliter la conception des pages ASP.NET.

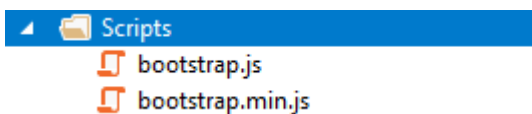
Razor est inclus par défaut dans le Framework .NET MVC et est pris en charge par Visual Studio qui offre des fonctionnalités supplémentaires telles que l'IntelliSense,

débogage et la coloration syntaxique. La syntaxe des vues alors repose sur du C# dans un fichier qui contient à la fois du HTML, CSS et JavaScript.

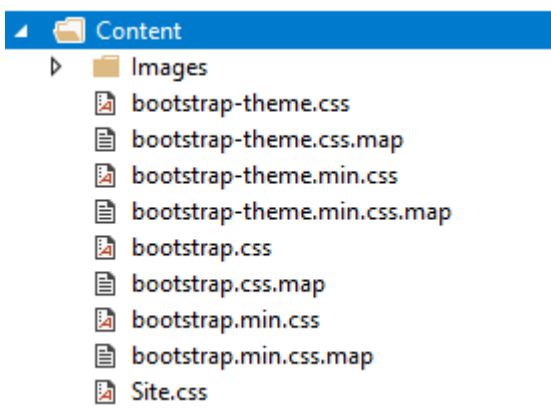
## b) Bootstrap :

Bootstrap est une collection d'outils utiles à la création du design (graphisme, animation et interactions avec la page dans le navigateur, etc.) de sites et d'applications web. C'est un ensemble qui contient des codes HTML et CSS, des formulaires, boutons, outils de navigation et autres éléments interactifs, ainsi que des extensions JavaScript en option.

Le Bootstrap est intégré par défaut dans les projets de développement d'application Web. On le retrouve par ailleurs dans le dossier Scripts en tant que code JavaScript :



On trouve aussi des fichiers bootstrap d'extension .css dans le dossier Content pour l'ajustement du design.



Les fichiers bootstrap\*.css ont été très utiles pour la réalisation du design du site. En plus pour des ajustements personnalisés j'ai trouvé nécessaire d'ajouter le fichier « Site.css ».

### c) WebGrid

Le WebGrid est un paquet permettant la simplification de l'affichage des données d'une table de la base de données dans une page ASP.NET.

Le WebGrid Helper retourne un tableau HTML qui affiche les données d'une table. Il affiche par défaut toutes les colonnes retournées par la requête. La possibilité de sélection de certaines colonnes est permise.

L'assistant prend en charge les options de mise en forme pour créer un moyen de parcourir ces données, et pour laisser aux utilisateurs la latitude de trier en cliquant sur un titre de la colonne.

Il est possible aussi grâce au WebGrid de créer une pagination du tableau pour des raisons d'économie dans l'espace de la page Web ainsi que pour le design.

Le WebGrid a été utile dans la partie du projet impliquant l'affichage du tableau de recherche des concours par les utilisateurs du site.

### d) Authentication

Le modèle de code Visual Studio a généré un ensemble de méthodes qui sont là pour implémenter de manière automatique toutes les fonctionnalités habituelles d'une authentification par login et mot de passe.

- L'inscription, qui se dit '**register**' en anglais.
- L'authentification en elle-même.
- La possibilité de récupérer son mot de passe en cas de perte.
- La désinscription.

Le contrôleur contient un certain nombre de méthodes simples et qu'il est possible de modifier afin d'arranger le résultat selon le cahier de charges.

### e) LINQ

LINQ est le sigle pour **Language-Integrated Query**. C'est un langage de requête puissant introduit avec **.Net 3.5** dans **Visual Studio**. LINQ peut être utilisé avec **C#** ou **Visual Basic** pour interroger différentes sources de données.


Connaissant **SQL**, il s'agit d'un langage de requête structuré utilisé pour enregistrer et récupérer des données d'une base de données. De la même manière, **LINQ** est une syntaxe de requête structurée construite dans **C#** et **VB.NET** pour récupérer des données de différents types de sources de données telles que les collections, **ADO.Net DataSet**, **XML Docs** et **MS SQL Server** ainsi que d'autres bases de données.

De ce fait, pour les interrogations de la base de données je me suis intéressée au langage **LINQ**.

## V. Présentation du site Web

- **Espace Gestionnaire**

### Page de connexion



The screenshot shows a web login page with a light blue background. In the center, there is a white rectangular box with a thin border. Inside this box, the title 'Connexion' is centered at the top. Below the title, there are two input fields. The first field is labeled 'Nom Utilisateur' in blue text above it. Below the first field, there is a red label 'Champ obligatoire' and a blue label 'Mot de passe' above the second field. The second field is labeled 'Mot de passe' in blue text above it. Below the second field, there is a red label 'Champ obligatoire'. At the bottom of the box, there is a dark blue button with the word 'Connexion' in white text.

## Inscription

Inscription

Nom Utilisateur

Champ obligatoire

Mot de passe

\*\*\*\*

Le mot de passe doit contenir au moins 8 caractères

Confirmation du mot de passe

\*\*\*

La confirmation n'est pas compatible

Code organisateur

Code coordinateur

SuperAdmin

☐

Coordinateur

☐

Enregistrer

Login

## Affichage du compte

---

[Archives concours](#)   [Deconnexion](#)

---



## Ajout du Concours

### Concours

Date_Concours jj/mm/aaaa	Age limite 0
Nombre de postes disponibles 0	Catégorie A
Intitulé	Créé par
Date_Publication jj/mm/aaaa	Créé le jj/mm/aaaa
Modifié par	Organisateur Selectionner
Modifié le	Conditions d'éligibilité Upload file Choisir un fichier   Aucun fichier choisi
Annulé par	Statut Selectionner
Annulé par	Grade Selectionner
	Délai_dépôt 01/01/0001 00:00:00

Créer

## Archives des concours

### Archives

Numero	Intitulé du Concours	Date du Concours	Date de publication	Délai de dépôt	Catégorie	Statut	Organisateur	Grade	Nombre de postes	Age limite	Créé le	Créé par	Modifié le	Modifié par	Annulé le	Annulé par	Conditions d'éligibilité	Action
8	Cocours Chef de division Inspection financière	02/12/2020 00:00:00	13/10/2020 00:00:00	13/11/2020 00:00:00	A	Publié	La Trésorerie Générale du Royaume	Inspecteur des finances	24	35	09/09/2020 12:20:33	Anbar Hamri					1	<a href="#">Modifier</a>

Ajouter

# VI. Bilan

## A. Bilan des actions réalisées

### 1. Parties exécutées :

- Le formulaire d'inscription pour chaque gestionnaire.
- Le formulaire d'inscription pour le candidat.
- Les sessions de connexion pour chaque gestionnaire.
- L'archive des concours (Liste).
- Ajout, modification et suppression des concours de la liste.

### 2. Parties non exécutées :

- Affichage de la page d'inscription pour le candidat.
- Enregistrement des fichiers dans la base de données.
- Exécution de la requête de recherche.

## B. Problèmes rencontrés

- Infamiliarité avec le langage C# ainsi qu'avec le langage LINQ.
- Difficultés dans la gestion de temps entre étude et projet de stage.

## C. Conclusion

Ce projet de stage a été très fructifiant en terme de découvertes des technologies d'actualités dans le monde du développement Web. L'approche du logiciel Visual Studio était intéressante et m'a ouverte les yeux sur plusieurs outils professionnels. Malgré le faible rendement en vue du manque de pratique de ces technologies ainsi que la contrainte du temps, j'évalue personnellement cette expérience comme très enrichissante en terme de connaissances.

## VII. Bibliographie

### Documents PDF

- [www.cours-gratuit.com--id-2287.pdf](http://www.cours-gratuit.com--id-2287.pdf)
- **Université d'Aix-Marseille -génie Logiciel (cours De Pr. Achref El Mouelhi)**
  - Cours aspSessions.pdf
  - GestionSession.pdf

### Sites Web

- [www.clubic.com](http://www.clubic.com)
- [www.tutorialsteacher.com](http://www.tutorialsteacher.com)
- [www.openclassrooms.com](http://www.openclassrooms.com)
- [www.lsis.org](http://www.lsis.org)
- [www.entityframework.net](http://www.entityframework.net)
- [www.dotnetawesome.com](http://www.dotnetawesome.com)
- [www.w3bai.com](http://www.w3bai.com)
- <https://www.w3schools.com/>
- <https://stackoverflow.com/>

## VIII. Annexe

### Table de l'annexe :

- Animation de page
  - WebGrid
  - Pagination
- Authentification
  - Partie concernant la connexion

- Partie concernant la déconnexion
- Partie concernant la session d'ouverture
- Requête de recherche
- Ligne de feuille de style ajoutées

# **Annexe**

# Animation de page

# WebGrid

## + HomePage View

### Entête :

```
@{
    Layout = null;
    var grid = new WebGrid (canPage: true, rowsPerPage: 10);
    grid.Bind(source: Model, rowCount: ViewBag.TotalRows , autoSortAndPage: false);
}
```

### Lister des concours avec pagination :

```
<h3>Liste des concours recherchés</h3>
    @grid.Table(
        tableStyle:"table table-responsive table-bordered" ,
        columns: grid.Columns(
            grid.Column(columnName:"db.Concours.Numero_Concours",header:"Numero"),
            grid.Column(columnName:"db.Concours.Intitulé_Concours", header:
"Intitulé"),
            grid.Column(columnName:"db.Concours.Date_Concours", header: "Date du
concours"),
            grid.Column(columnName:"db.Concours.db.Concours.Catégorie", header:
"Catégorie"),
            grid.Column(columnName:"db.Concours.StatutConcours.Statut", header:
"Statut"),
            grid.Column(columnName:"db.Concours.Age_limite", header: "Age limite"),
            grid.Column(columnName:"db.Concours.Grade.Titre", header: "Grade"),
            grid.Column(columnName:"db.Concours.ConditionsEligibilitéDocumentsJoint.Fichier_CE", header:
"Conditions d'eligibilité")
        )
    )

    <div class="row">
        <div class="col-sm-6">
            @grid.PagedList(mode: WebGridPagerModes.All)
        </div>
    </div>

    @Html.Hidden("dir",grid.SortDirection)
    @Html.Hidden("col",grid.SortColumn)

</div>
</div>
```

```

<script src=~/Scripts/jquery-3.3.1.js"></script>
<script src=~/Scripts/bootstrap.min.js"></script>
<script>

    $(function()
    {

        try {
            var dir = $('#dir').val();
            var col = $('#col').val();
            var header = $("th a[href*=" + col + "]");

            if (dir == "descending")
            {
                header.text(header.text() + "▼");
            }
            if (dir== "ascending")
            {
                header.text(header.text() + "▲");
            }

        }
        catch (e) {}
    }
    )
</script>

```

## Pagination :

```

public ActionResult PageHome(DateTime datec1 , DateTime datec2, DateTime dated1, DateTime
dated2, int page = 1, string selectedtitre="", string selectedcategorie = "", string
sort="Numero", string sortdir="asc")
{

    int pageSize = 10;
    int totalRecord = 0;
    if (page < 1) { page = 1 ;}
    int skip = (page * pageSize)-pageSize;
    var data = Rechcrslist(selectedtitre, selectedcategorie, sort, sortdir, datec1,
datec2, dated1, dated2,skip,pageSize,out totalRecord);
    ViewBag.TotalRows = totalRecord;

    return View(data);
}

```



# Authentication

## + GestionnaireController

### Partie concernant la connexion

```
[HttpPost]
public ActionResult Login(Gestionnaire GestionnaireModel)
{
    Projet_eRecrutementEntities db = new Projet_eRecrutementEntities();

    var GestionnaireLoggedIn = db.Gestionnaire.SingleOrDefault(x => x.UserName ==
GestionnaireModel.UserName && x.Mot_de_passe == GestionnaireModel.Mot_de_passe);

    if (GestionnaireLoggedIn != null)
    {
        ViewBag.Message = "Connexion établie";
        ViewBag.Ressayer = "oui";

        Session["UserName"] = GestionnaireModel.UserName;

        return RedirectToAction("Compte", "Gestionnaire", new { UserName =
GestionnaireModel.UserName });
    }
    else
    {
        ViewBag.Ressayer = "oui";
        return View();
    }
}
```

### Ouverture de session :

```
public ActionResult Login()
{
    if (Session["UserName"] != null)
    {
        return RedirectToAction("Compte", "Gestionnaire", new { UserName =
Session["UserName"].ToString() });
    }
    else
    {
        return View();
    }
}
```

```
[HttpPost]
public ActionResult Login(Gestionnaire GestionnaireModel)
```

```

{
    Projet_eRecrutementEntities db = new Projet_eRecrutementEntities();

    var GestionnaireLoggedIn = db.Gestionnaire.SingleOrDefault(x => x.UserName ==
    GestionnaireModel.UserName && x.Mot_de_passe == GestionnaireModel.Mot_de_passe);

    if (GestionnaireLoggedIn != null)
    {
        ViewBag.Message = "Connexion établie";
        ViewBag.Ressayer = "oui";

        Session["UserName"] = GestionnaireModel.UserName;

        return RedirectToAction("Compte", "Gestionnaire", new { UserName =
GestionnaireModel.UserName });
    }
    else
    {
        ViewBag.Ressayer = "oui";
        return View();
    }
}

public ActionResult Compte(string UserName)
{
    if (Session["UserName"] == null)
    {
        return RedirectToAction("Login", "Gestionnaire");
    }
    else
    {
        ViewBag.Message = Session["UserName"];

        return View();
    }
}
}

```

## Partie concernant la déconnexion :

### La page compte

```

<li><a href="/~/Home/PageHome">Deconnexion</a></li>

```

# La requête de recherche

## + HomeController

```
public List<Concours> Rechcrslist(string selectedtitre , string selectedcategorie ,string
sort,string sortdir, DateTime datec1 , DateTime datec2, DateTime dated1, DateTime dated2,
int skip , int pageSize,out int totalRecord)
{
    using (Projet_eRecrutementEntities db = new Projet_eRecrutementEntities())
    {
        var v = (from x in db.Concours
                 where x.StatutConcours.Statut.Contains(selectedcategorie)
                     && x.Grade.Titre.Contains(selectedtitre)
                     && datec1 <= x.Date_Concours && x.Date_Concours <=
datec2
                     && dated1 <= x.Délai_dépôt && x.Délai_dépôt <= dated2
                 select x
                );
        totalRecord = v.Count();
        v = v.OrderBy(sort + " " + sortdir);

        if (pageSize > 0)
        {
            v = v.Skip(skip).Take(pageSize);
        }

        return v.ToList();
    }
}
```

Lignes de feuille de style ajoutées

Dans cette partie, je mentionne l'ensemble des lignes que j'ai dû créer pour modifier dans la feuille de style ajoutée par défaut dans le dossier « Content » afin d'améliorer la mise en site web et la cohérence des couleurs.

## +Fichier Site.css

```
/* Régler le padding pour empêcher le contenu de heurter les bords */

.body-content {
    padding-left: 15px;
    padding-right: 15px;
}

/* Depasser le comportement bootstrap par défaut où les listes de description horizontales
tronqueront les elements trop longs pour tenir dans la colonne de gauche*/
.dl-horizontal dt {
    white-space: normal;
}

/* Etablir la largeur de la forme des elements une fois atteints 100% de largeur par default*/
input,
select,
textarea {
    max-width: 280px;
}

.fiche {
    margin-left: auto;
    margin-right: auto;
    font-size: 40px;
    width: 1400px;
    height: 900px;
    padding-top: 100px;
}

.p a {
    font-size: 50px;
    font-family: Cambria;
    color: rgba(91, 157, 218, 0.678);
}

.blocformLogin{

    margin-left: auto ;
    margin-right: auto;
    padding-top: 250px;
    height: 800px;
```

```

    width: 600px;
}

.blocformAjoutEdit {
    margin-left: auto;
    margin-right: auto;
    padding-top: 150px;
    margin: 0px;
}

.blockArchive {
    margin-left: 0px;
    margin-right: 10px;
    margin-top: 100px;
}

}

.blockhome {
    height: 200px;
    width: 400px;
    background-color: rgb(92, 25, 136);
    color: rgb(108, 68, 128);
    border-color: rgba(60, 17, 88, 0.555);
    margin-top: 300px;
    margin-left: 1000px;
    margin-right: 0px;
    padding: 10px;
}

```