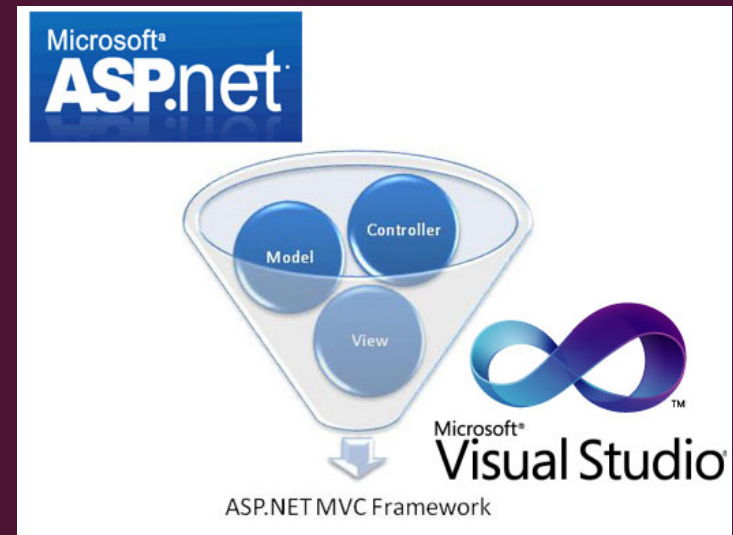


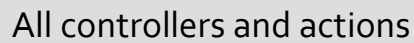
ASP.NET MVC

INTRODUCTION À ASP.NET MVC

- MVC : Models, Views, et Controllers
- MVC dans le Framework .NET



Static files (CSS, Images, etc.)



All models

JavaScript files (jQuery, Modernizr, knockout, etc.)

[View templates](#)

_Layout.cshtml – master page (main template)

Application_Start() – The entry point of the application

Configuration file

```
public class HomeController : Controller
{
    public ActionResult Index()
    {
        ViewBag.Message = "Index page.";
        return View();
    }
}
```

Modèles, vues et contrôleurs

- Les modèles représentent les entités d'une application et les classes correspondantes au logique métier.
- Les contrôleurs interagissent avec les requêtes utilisateur et se comportèrent comme le cerveau de l' application,
- Les vues construisent les interfaces utilisateur.

Modèles et données

- Un modèle est un ensemble de classes .NET qui représentent les objets manipulés par un site web.
- Comme toute classe .NET, les classes de modèle peuvent inclure un constructeur, qui est une procédure qui s'exécute lorsqu'une nouvelle instance de cette classe est créée.

Modèles, vues et contrôleurs

- Elle peuvent également inclure d'autres procédures, si nécessaire. Ces procédures encapsulent la logique métier.
- La plupart des sites Web stockent des informations dans une base de données. Dans une application MVC, le modèle inclut le code d'accès aux données.
- ASP.NET MVC fonctionne avec de nombreux frameworks d'accès aux données. cependant, Le framework le plus couramment utilisé est Entity Framework

Contrôleurs et Actions

- Un contrôleur est une classe .NET qui répond aux requêtes du navigateur Web dans une application MVC.
- Il y a habituellement une classe de contrôleur pour chaque classe de modèle.

Modèles, vues et contrôleurs

- Les contrôleurs incluent des actions, qui sont des méthodes qui s'exécutent lors d'une réponse à une requête utilisateur.
- Par exemple, le contrôleur de la classe produit peut inclure une action d'achat qui s'exécute lorsque l'utilisateur clique sur le bouton Ajouter du panier de l'application Web.
- Les contrôleurs héritent de la classe de base `System.Web.Mvc.Controller`.
- Les actions retournent généralement un Objet de type `System.Web.Mvc.ActionResult`.

Modèles, vues et contrôleurs

Vues et Razor

- Une vue est, par défaut, un fichier .cshtml ou .vbhtml contenant à la fois un code HTML et un code en c# ou en vb.net.
- Un Moteur de vues interprète les fichiers « Views », exécute le code côté serveur et rend le code HTML au navigateur Web.
- Razor est le moteur de vue par défaut dans ASP.NET.
- ASP.NET MVC prend également en charge le moteur de vue ASPX.
- De plus on peut installer d'autres moteurs de vue tels que Spark et NHaml.
- Le moteur de vue Razor identifie le code côté serveur en recherchant le symbole @

Modèles, vues et contrôleurs

- Exemple d'une vue utilisant La syntaxe Razor

```
<h2>Details</h2>
<fieldset>
  <legend>Comment</legend>
  <div class="display-label">
    @Html.DisplayNameFor(model => model.Subject)
  </div>
  <div class="display-field">
    @Html.DisplayFor(model => model.Subject)
  </div>
  <div class="display-label">
    @Html.DisplayNameFor(model => model.Body)
  </div>
  <div class="display-field">
    @Html.DisplayFor(model => model.Body)
  </div>
</fieldset>
```

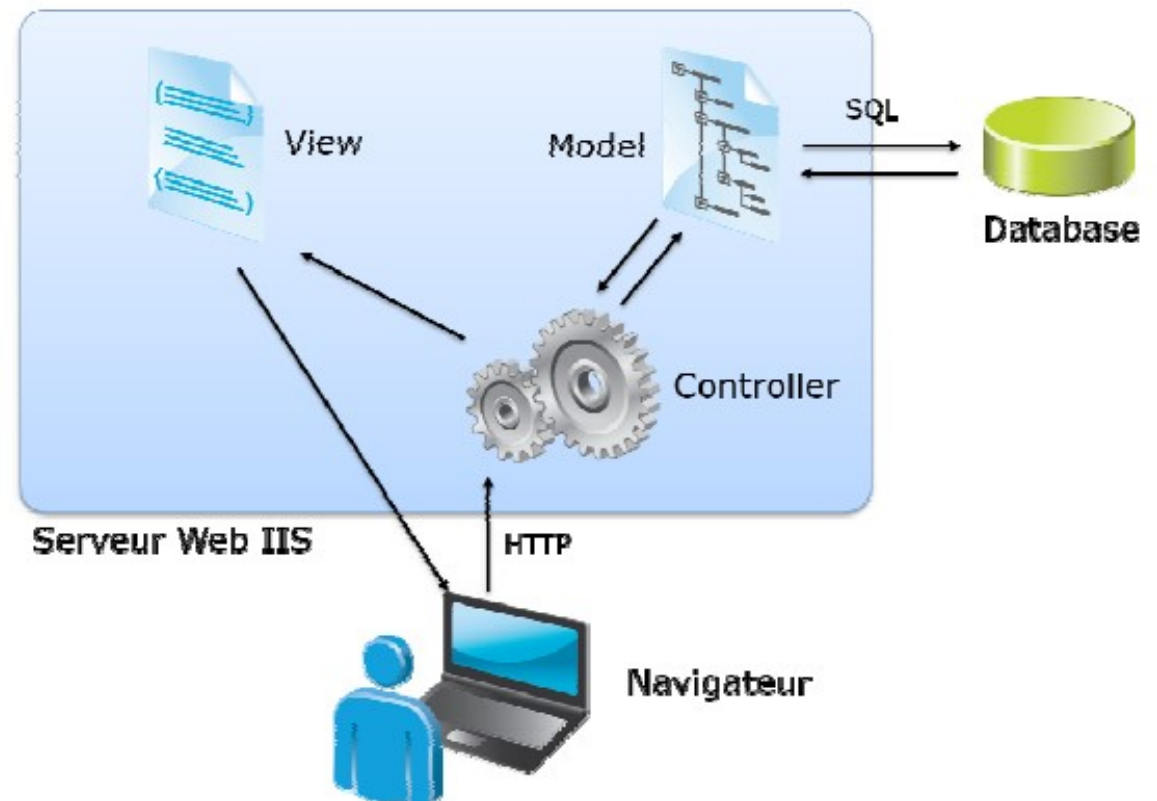
Modèles, vues et contrôleurs

Cycle de vie d' une requête

- La demande concerne les détails d'un produit portant l'ID « 1 »:
- 1. L'utilisateur demande l'adresse Web: <http://www.adventureworks.com/Product/Display/1>
- 2. Le moteur de routage MVC examine la demande et détermine qu'il doit la transmettre au Contrôleur « Product » et à l'action « Display ».
- 3. L'action « Display » dans le contrôleur « Product » crée une nouvelle instance de la classe modèle « Product ».
- 4. La classe modèle « Product » interroge la base de données pour obtenir les informations concernant le produit ayant l'ID « 1 ».
- 5. L'action « Display » crée également une nouvelle instance de la vue « Display » et lui transmet les données nécessaires.

Modèles, vues et contrôleurs

6. Le moteur de vue Razor exécute le code côté serveur dans la vue « Display » pour générer le code HTML.
7. La page HTML une fois terminée est renvoyée au navigateur pour affichage.



MVC DANS LE FRAMEWORK .NET

