<http://www.onlinedown.net/soft/1149183.htm>

CodematicDemoMVC 是一个基于三层架构的MVC模式应用的完整示例项目源码，基于Asp.net 3.5开发，  
结构更简洁，提供更多有效的示例源码参考。  
方便用户更好的理解和使用该架构进行开发，配合动软.Net[代码生成器](http://www.onlinedown.net/soft/63546.htm)，可以使开发[效率](http://www.onlinedown.net/sort/244/)事半功倍，倍感轻松。该项目涉及如下几项要点应用：  
1) 基于标准的三层结构。  
2) 表示层采用MVC模式。  
3) 数据的增、删、改、查操作。  
4) 列表数据显示及分页[处理](http://www.onlinedown.net/soft/96372.htm)。  
5) Linq的应用等。\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Lib：公用类库  
BLL: 业务层  
Model: 数据实体  
DAL: 数据层  
WebMVC: 表示层DB\_51aspx下为Sql2005数据库，附加即可  
默认管理帐号/密码：51aspx/51aspx通过对MVC的学习，我们可以看到MVC有如下特点：  
1) 业务处理与显示分离：ASPX页面与CS代码彻底分离。更好的复用和维护。  
2) [服务器](http://www.onlinedown.net/sort/52/)端的表单[控件](http://www.onlinedown.net/sort/132/)不再被提倡使用，取而代之使用传统的input，或直接让Html.TextBox生成控件标签。  
3) 没有了事件[驱动](http://www.onlinedown.net/soft/81870.htm)模型。在ASP.NET MVC中，当某个按钮被点击，你不要再习惯性想到应该在相应的aspx.cs中有个Click处理方法，你应该想到的是该有某个Controller中有个Action来处理这个事件。  
4) 没有了数据绑定。如果你习惯了GridView的数据绑定，在ASP.NET MVC中则需要改变思路，你自己动手解决。  
5) 增加了地址重写。MVC的Routing组件让我们可以很好的进行URL路由处理。  
6) 总体[来看](http://www.onlinedown.net/soft/867109.htm)，可以理解MVC是基于Web Form的一种编程方式模型扩展，是一种表示层的编程模型方案。