在线学习考试系统

开发文档

# 概述

MVVM架构，C#、Sqlserver、Vue。

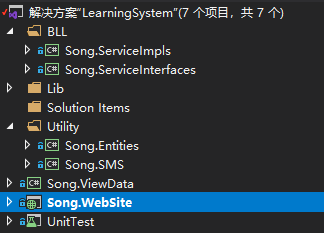
基于DotNet 4.6.2框架，C#语言，开发工具为Microsoft Visual Studio Community 2019；数据库Sqlserver2008。

前端部分主要采用Vue，web端采用ElementUI组件库，手机端采用VantUI，管理后台采用WebdeskUI（自主组件库）。

整个系统基于B/S架构，数据层采用了ORM，使用的是Mysoft.Data组件；业务层与表现层之间用了spring.net的IOC组件；相较于1.0版本，增加了RESTful API接口层（Song.ViewData项目）。

# 解决方案描述

在解决方案中，共分为七个项目。如下图



## 业务层

BLL解决方案文件夹下，是业务层的两个项目；

Song.ServiceInterfaces是所有业务层的接口

Song.ServiceImpls 是业务层（Song.ServiceInterfaces）的接口实现。

之所以分成两个项目，是为了基于面向接口编程，例如当业务层代码需要变动时，只需要改动实现，不用改动接口。实口与实现类的关系是在Song.Site项目中的web.config中配置的。如下图



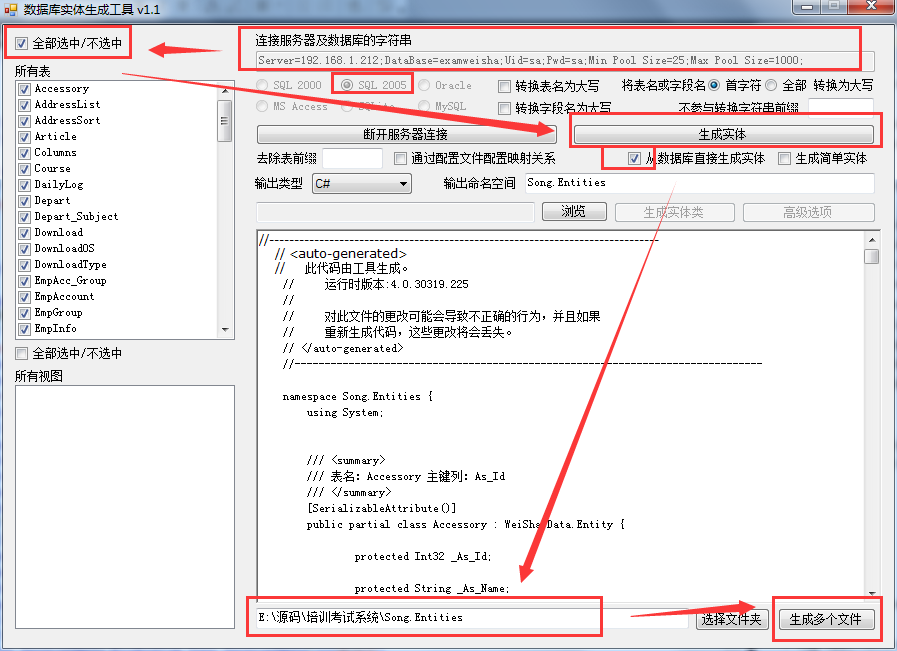
上图定义了接口名称，以及其实现类。这里是用了Spring.net的IOC（控制反转）组件。

## 数据层

Song.Entities是数据实体。

数据层的处理采用了ORM方式处理，这里是用了Mysoft.Data开源组件，网上可以找到下载。使用方法如下：

在Lib解决方案文件夹下，有WeiSha.Data.Generete.exe，双击打开。如下图步骤操作。



生成实体后，在Song.ServiceImpls项目中引用Song.Entities

在编写时，例如增加一个记录

/// <summary>

/// 添加

/// </summary>

/// <param name="entity">业务实体</param>

public void Add(Accessory entity)

{

entity.As\_CrtTime = DateTime.Now;

Song.Entities.Organization org = Business.Do<IOrganization>().OrganCurrent();

if (org != null)

{

entity.Org\_ID = org.Org\_ID;

entity.Org\_Name = org.Org\_Name;

}

Gateway.Default.Save<Accessory>(entity);

}

其中关键性语句是 Gateway.Default.Save<Accessory>(entity);

## 表现层

Song.WebSite是前端项目，主要有路由控制器，以及html网页，Js，css等文件构成。

Song.ViewData是所有 RESTful API 接口方法。

## 其它项目

Song.SMS短信发送的项目。

UnitTest单元测试。