目录

随着时代的进步和发展，无论在计算机还是通信领域，信息化的社会中信息获取和传递将起到越来越重要的地位。

学校拥有自己的信息媒介非常重要。在信息化时代网络横行的年代，建设符合学校背景，需求的招聘就业平台必将以指数倍的形式提高学生就业的效率和积极性。

现阶段学校的招聘就业信息网站已沦为鸡肋，其实质为新闻发布系统。因特网上如此多的招聘就业信息网站，使学生无从下手，海投简历，海量浏览无法保证学生就业的效率。同时，企业无法获得校园中求职学生的特长，技能等信息，在很大程度上失去了被“伯乐”发现的机会。

概述

信息筛选方面，用户能够订阅自己感兴趣的招聘信息，求职信息，校园活动，校园公告。系统将订阅的信息通过邮件，短信，Web Push 等方式传递给用户，从而使用户及时了解社会动向。

新的招聘就业网将主要采用互动形式的信息交互。分学生，教师，企业三个主要对象。其中学生和企业为主要用户，前者将能够浏览教师，就业指导员和企业发布的招聘就业信息。后者可以浏览学生的求职信息以及发布招聘信息。

学生与企业的互动采用流水形式的应聘、面试、试用、录用。学生对感兴趣的企业的职位发出应聘请求，企业查看该学生的自我简历与求职信之后决定是否给其面试机会；企业决定面试时间并接受应聘请求；学生收到面试通知参加企业的面试或者考试；企业在结束后给学生反馈考试（面试）信息以及是否录用；学生最终确定是否就职于该企业。

设计规范

系统架构

系统采用多层架构，数据、代码、视图分离方式开发。
N层结构从逻辑上相互独立其特点：是某一层的变动通常不影响其它层，具有很高的可重用性。减少整个系统的维护成本，具有良好的开放性，进行严密的安全管理，可支持异种数据库，有很高的可用性。
代价是前期的开发周期较长，成本较高。

模块结构

多层架构设计同时，利用.Net 简便的开发特性充分实现低耦合高内聚的模块结构。

计算处理

支持分布式云计算。
为保证在同时处理繁重任务时仍然稳定，系统支持分布式处理，信息处理和用户界面呈现可在不同地点，不同服务器上运行。信息处理服务能够在一台或者一台以上服务器平台上协同工作，效率为信息处理服务数量的倍数。
确保在因发生单点故障而卸载物理服务器时这些关键性消息处理仍能处于在线状态。

负载处理

支持服务器群集。
虚拟化的出现可以在硬件成本不变的情况下利用多核 CPU 运行多个操作系统，解决单一服务无法计时处理大量用户同时访问而性能下降乃至无响应的可能。
群集能够确保在因发生单点故障而卸载物理服务器时这些关键性应用程序仍能处于在线状态。实现这一功能丝毫不会增加用户使用上的复杂性。由于群集在终端用户、应用程序及网络面前呈现为一套单一系统，因此，他们可以像使用其它任意服务器那样来使用群集。

扩展性

提供内部公开 API。
提供外部标准 XML 数据源。

接口

数据源及 API 接口遵循 XML 标准。
XML 是基于 W3C 定制的开放标准，从而使得基于 XML 的应用具有广泛性。

数据标准

各层数据之间通信采用 XML 标准，同时遵循微软的 Web Service 2.0 标准，数据中包含各数据源类型的定义和说明，方便二次开发人员对系统进行开发

需求分析

所需设备

硬件环境

硬件基于 Intel Xeon。
多核多线程。

操作系统

Windows Server 2008 Release 2 RC。
华为短信信息机。

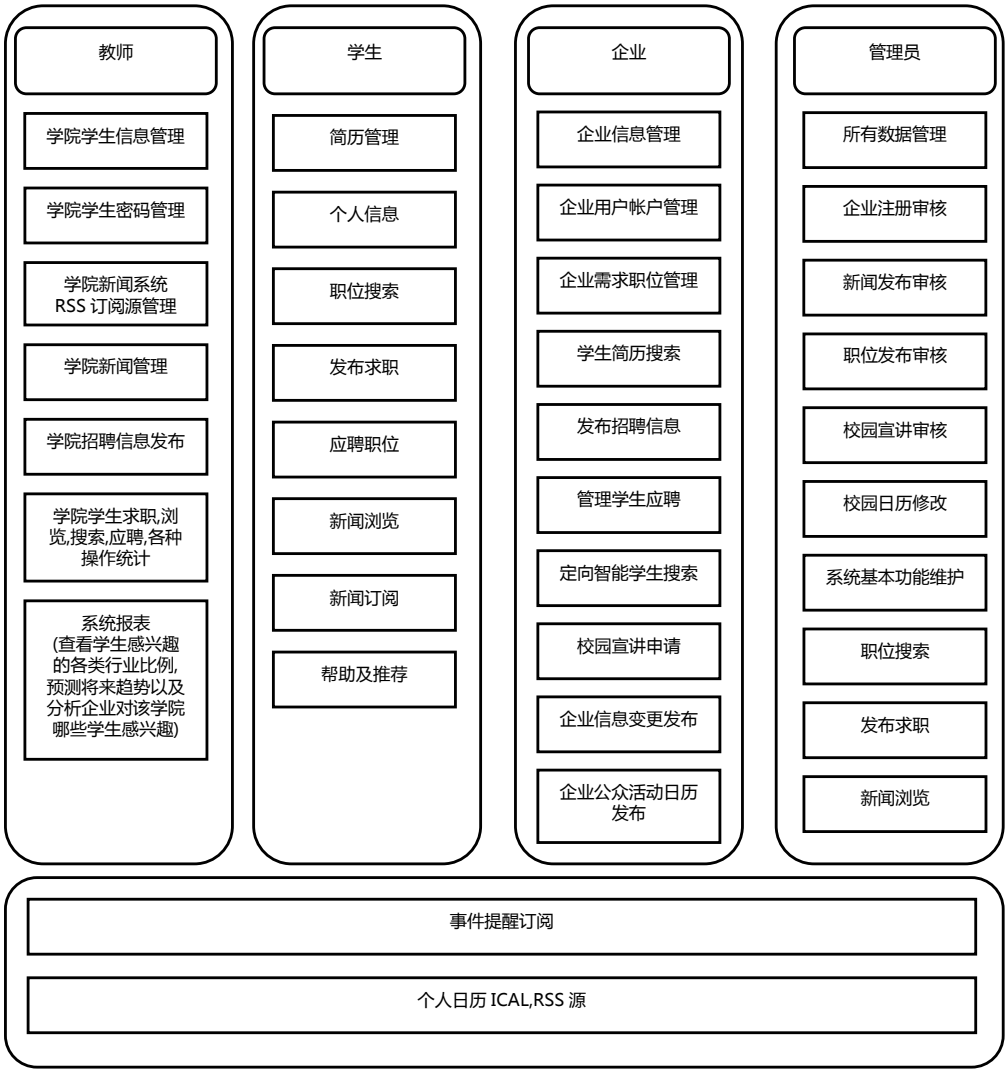
服务器软件

IIS 7.0 服务组件。
.Net Framework 4.0 可重组开发包。
Microsoft SQL Server 2008 数据库服务。
支持 POP3 及 SMTP 的邮件系统。

开发工具

Windows Server 2008 R2 RC
Microsoft Visual Studio 2010 Beta 1
Microsoft SQL Server 2008 企业版
VisualSVN 版本控制
Internet Express Browser 8.0 Beta

用户交互平台



交互平台基于 BS 架构（Server/Browse），用户和招聘就业系统
每单位用户对应一用户组，不同用户组的功能和操作也不同。

教师组

学院学生信息管理，学院学生密码管理，学院新闻系统 RSS 订阅源管理，学院新闻管理，学院招聘信息发布，学院学生求职,浏览,搜索,应聘,各种操作统计，系统报表(查看学生感兴趣的各类行业比例,预测将来趋势以及分析企业对该学院哪些学生感兴趣)。

学生组

简历管理；个人信息；职位搜索；发布求职；应聘职位；新闻浏览；新闻订

阅；帮助及推荐。

企业组

企业信息管理，企业用户账号管理，企业需求职位管理，学生简历搜索，发布招聘信息，管理学生应聘，定向智能学生搜索，校园宣讲申请，企业，息变更发布，企业公众活动日历发布。

管理员

所有数据管理，企业注册审核，新闻发布审核，职位发布审核，校园宣讲审核，校园日历修改，系统基本功能维护，职位搜索，发布求职，新闻浏览

消息发布方法

调查

结合校园学生生活习惯及调查，最有效快捷的方法将信息主动传到指定学生的贴身设备中，优选为短信传送。

结合企业工作人员及校园教师生活习惯及观察，电子邮件为其首选方式。

系统将内置 2 中消息通知方式，在用户未登录任何招聘就业系统终端时，即时的提醒事件、消息内容将主动发送至用户的手持设备或者电子邮件中。

短信

用户登录系统的交互平台或终端，给定界面允许用户输入手机号码，并向用户手机发送验证码。用户接收到验证码后，将其输入系统交互界面（终端）的手机验证码输入框，设置该用户为手机已绑定，并记录手机号码至数据库。

邮件

企业用户在注册时，输入邮件地址，首次注册信息将发送至企业用户电子邮箱。

学生，教师用户登录招聘就业系统交互界面后，随时可以更改邮件地址。

用户体验

使用户拥有操作招聘就业系统更好的体验，交互界面内容主要采用 Ajax 方式无刷新操作。

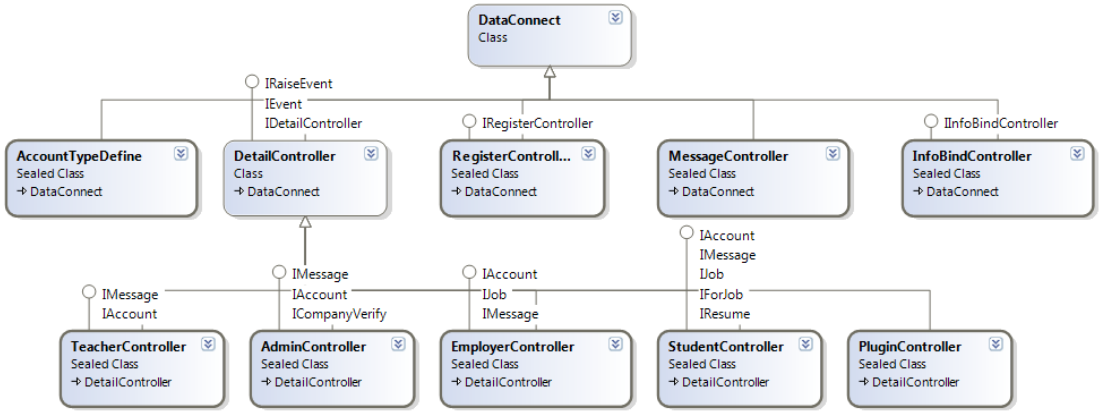
学生是该系统最大的用户群体，针对其时尚，追求新鲜的特性，系统提供个人日历订阅，新闻筛选订阅，校园活动订阅。这些订阅源可以采用 RSS，XML，邮件，短信，WAP Push 各种方式推送到用户设置的目的地。

提供内部 API 和外部 API 供开发人员使用。内部 API 用于开发扩展功能和第三方插件，增加用户体验指数。外部 API 可让学生制作类似 Vista 侧边拦的桌面、手持应用，突破仅有 BS 架构访问的局限。

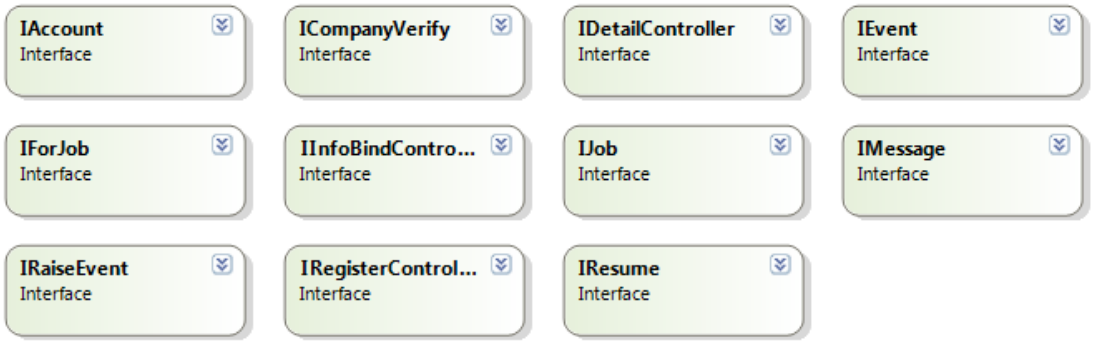
系统设计

核心部分

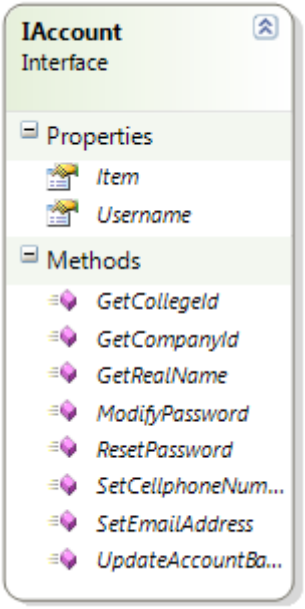
各类关系



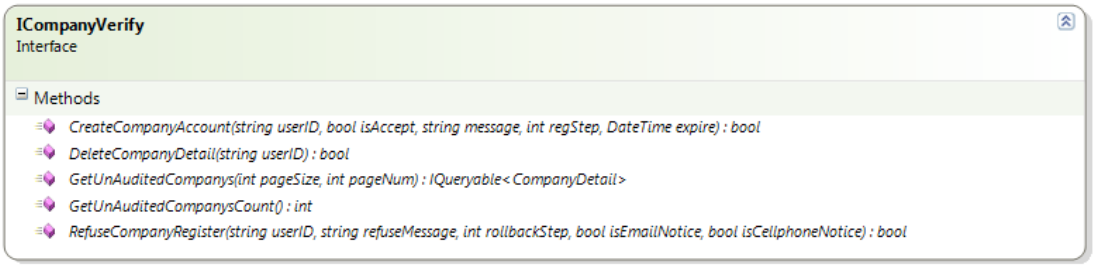
系统由教师，管理员，企业，学生，插件 5 个对象组成。这 5 个对象分别实现不同的接口功能。



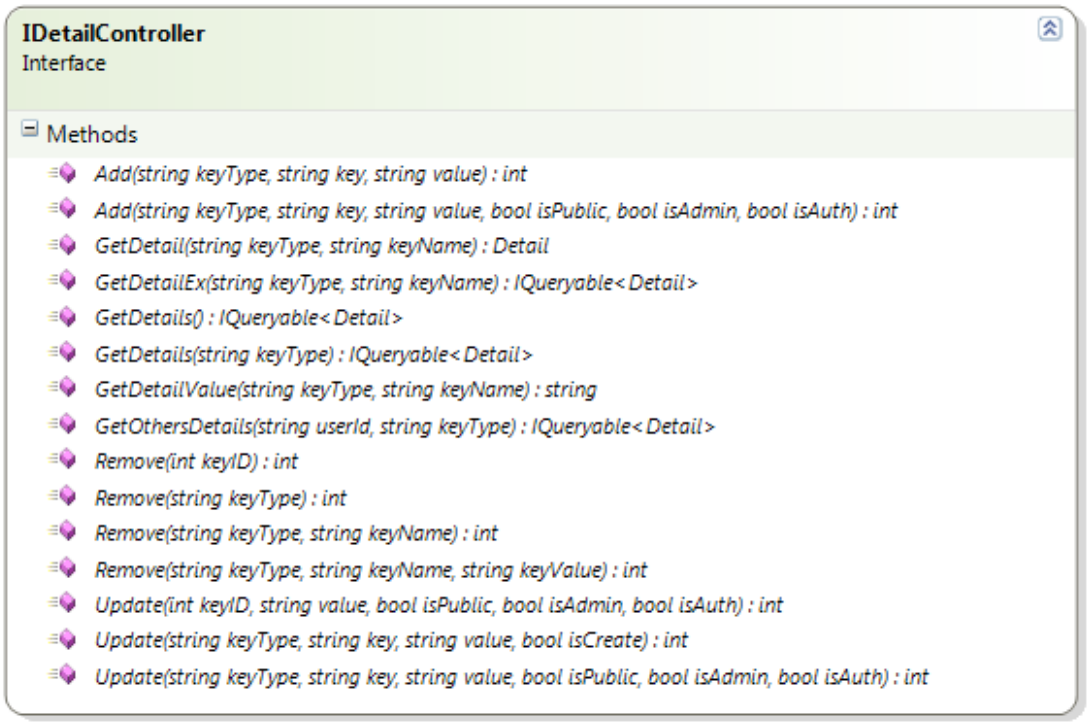
Namespace QzjCareer.Core 中共有 11 个接口，每个接口定义一组操作。



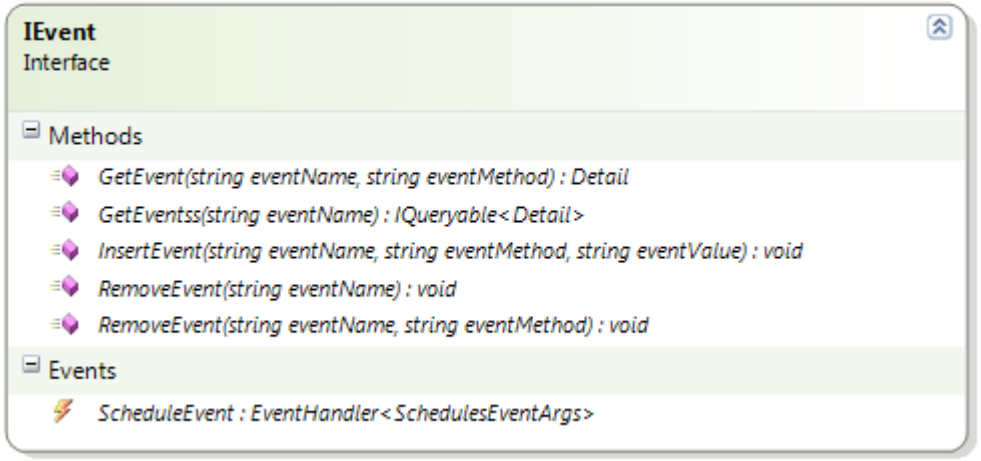
IAccount 定义账户操作，处理基本的账户信息。



ICompanyVerify 定义企业账号注册后管理员可对其执行的一系列操作。



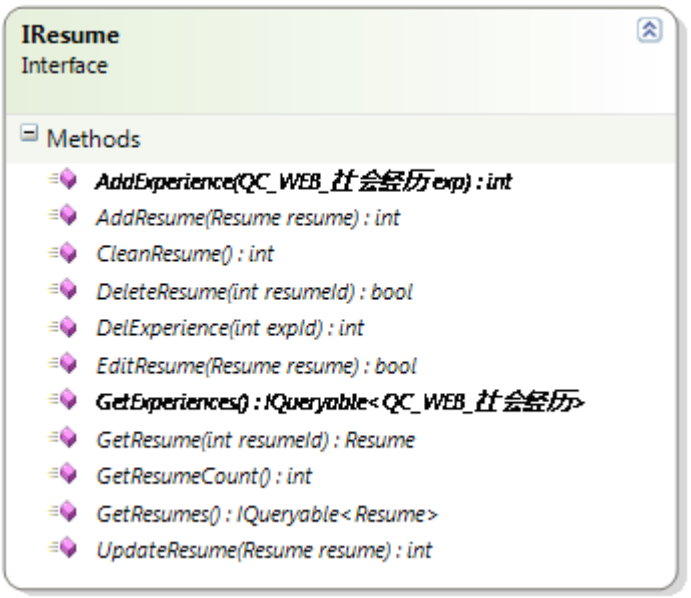
IDetailcontroller 定义所有对象拥有的参数数据库，该接口的实现通常用于第三方的数据操作中。



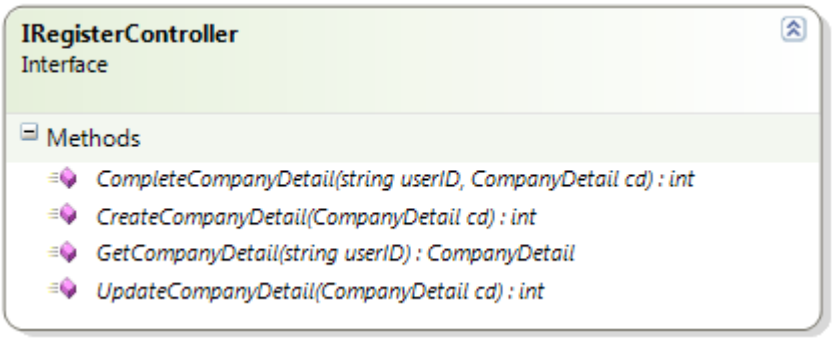
IEvent 定义系统内部产生的事件，该接口的实现用于消息队列和信息处理中。

IMessage 定义各用户之间的消息传送功能及用户组之间的消息传送权限。

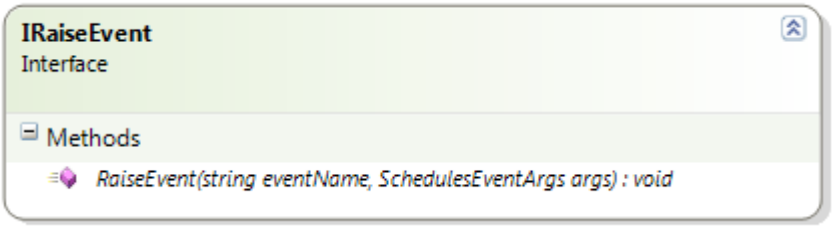
IJob 定义企业和学生招聘应聘中的方法。



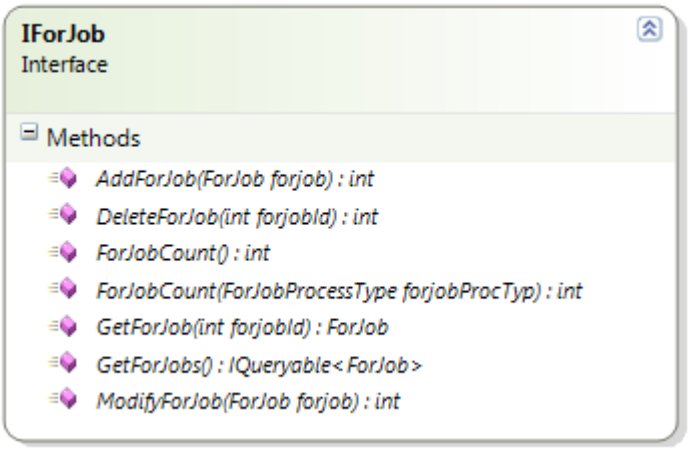
IResume 定义学生简历的操作。该接口将在更高级的版本中被使用。



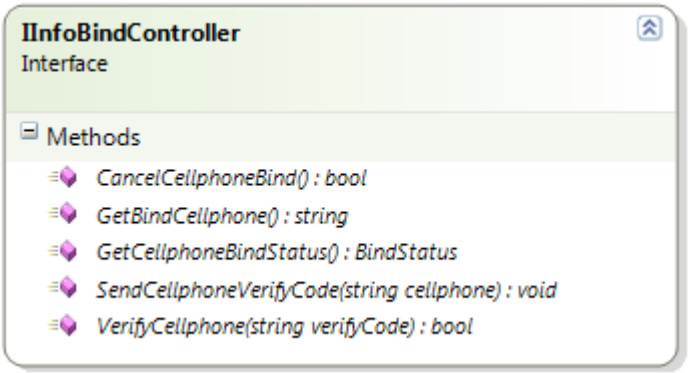
IRegisterController 定义管理员对注册账号进行审核的方法。



IRaiseEvent 定义事件处理类中的引发事件后的方法操作。

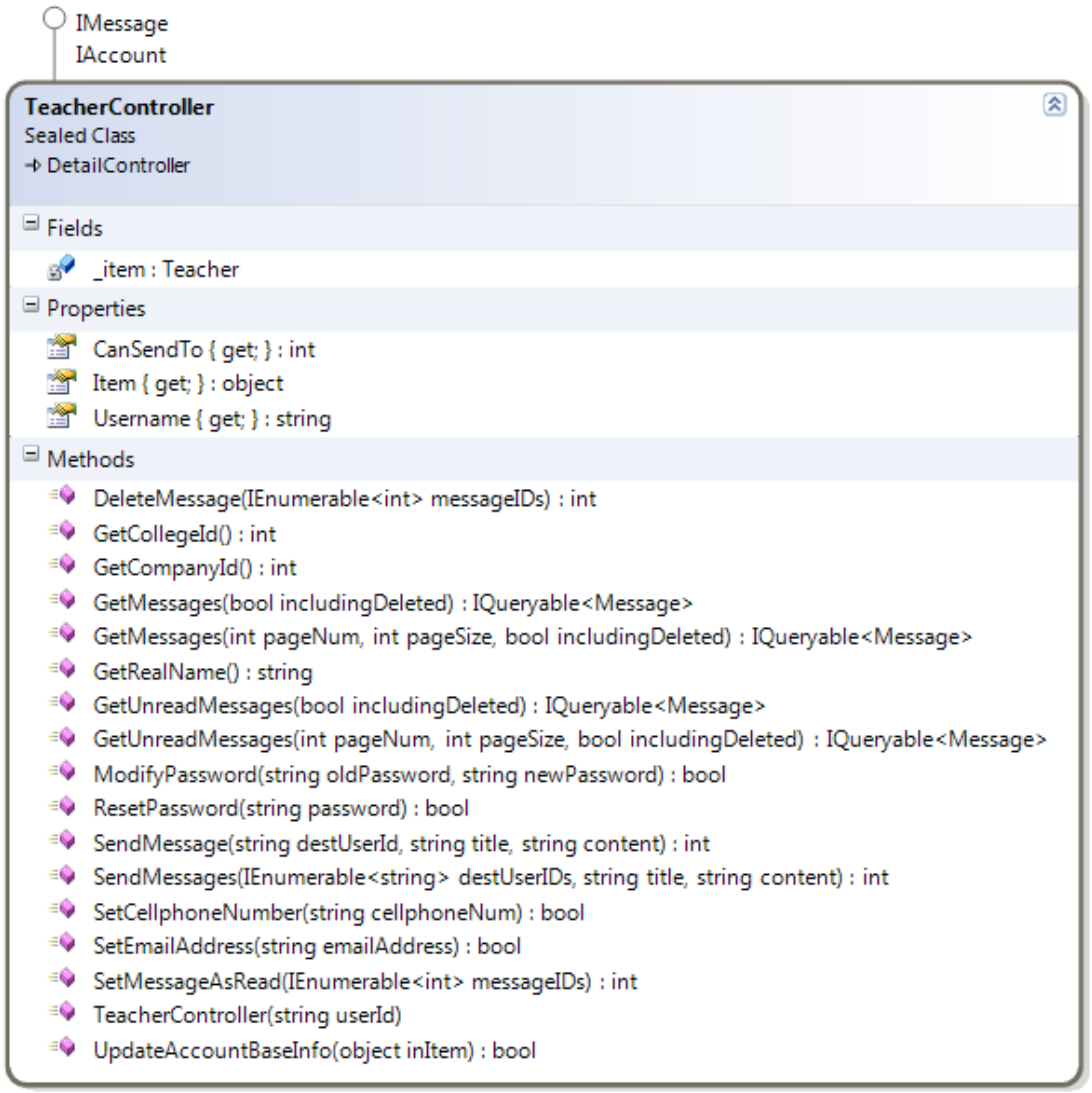


IForJob 定义学生发布求职的方法。

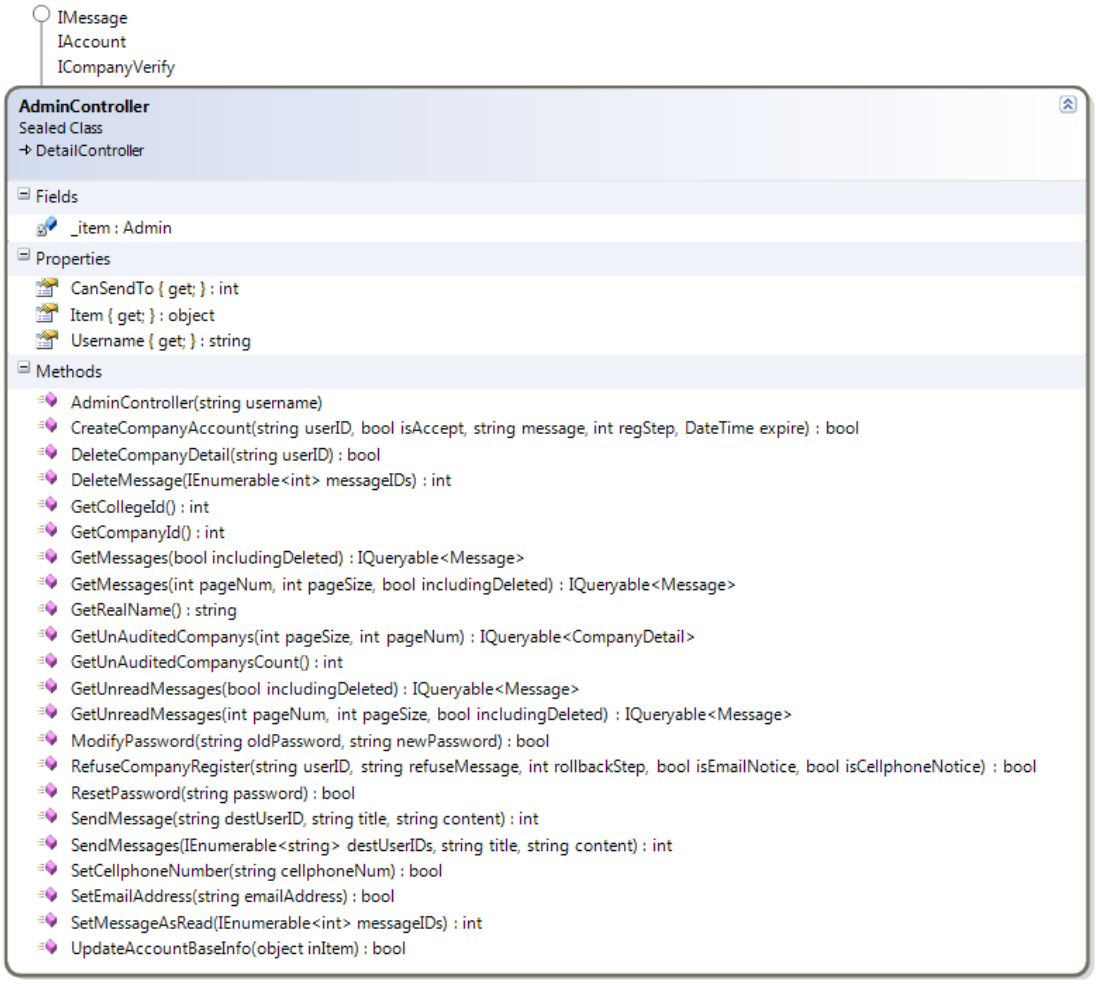


IInfroBindController 定义所有用户均包含的公共系统属性操作方法。

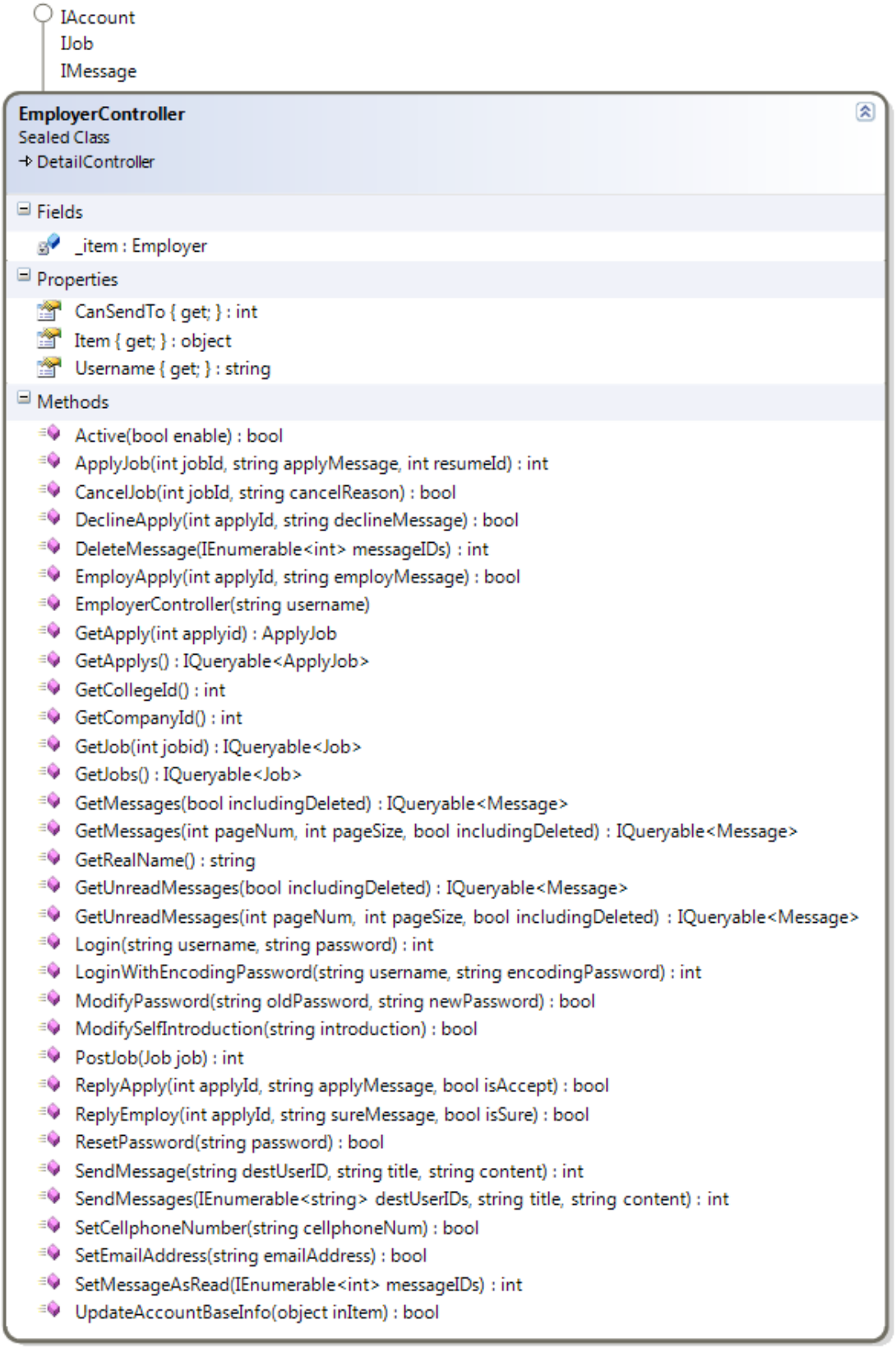
操作单位



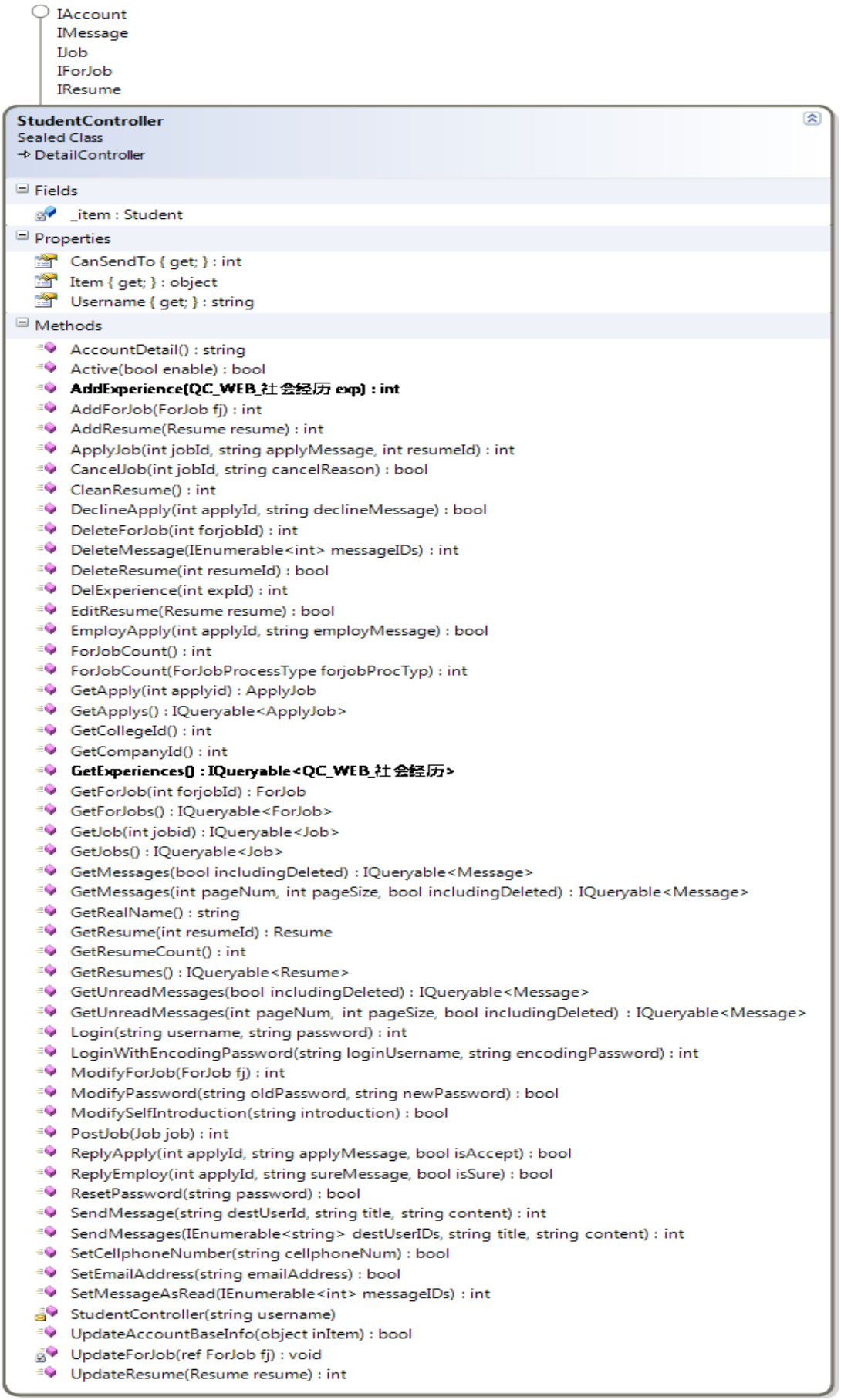
教师单位。实现 IMessage、IAccount 接口。继承于 DetailController 类。Sealed 类。



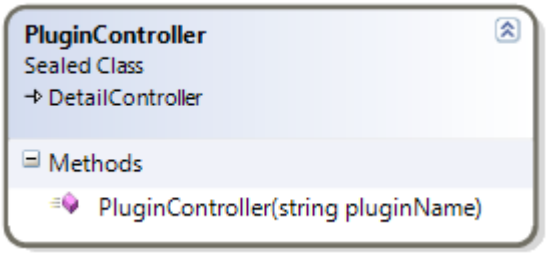
管理员单位。实现 IMessage， IAccount， ICompanyVerify 接口。继承于 DetailController 类。Sealed 类。



企业招聘者单位。实现 IAccount、IJob、IMessage 接口。继承于 DetailController 类。Sealed 类。

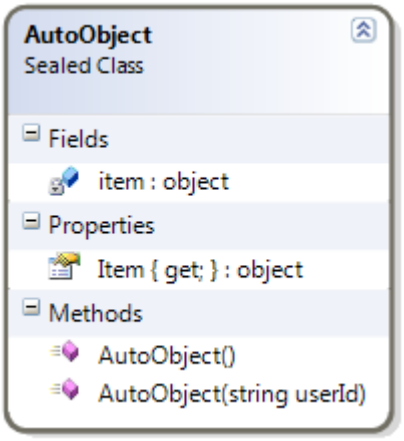

IAccount
IMessage
IJob
IForJob
IResume
StudentController
Sealed Class
DetailController
Fields
_item : Student
Properties
CanSendTo { get; } : int
Item { get; } : object
Username { get; } : string
Methods
AccountDetail() : string
Active(bool enable) : bool
AddExperience(QC_WEB_社会经历 exp) : int
AddForJob(ForJob fj) : int
AddResume(Resume resume) : int
ApplyJob(int jobId, string applyMessage, int resumeId) : int
CancelJob(int jobId, string cancelReason) : bool
CleanResume() : int
DeclineApply(int applyId, string declineMessage) : bool
DeleteForJob(int forjobId) : int
DeleteMessage(IEnumerable<int> messageIDs) : int
DeleteResume(int resumeId) : bool
DelExperience(int expId) : int
EditResume(Resume resume) : bool
EmployApply(int applyId, string employMessage) : bool
ForJobCount() : int
ForJobCount(ForJobProcessType forjobProcTyp) : int
GetApply(int applyid) : ApplyJob
GetApplys() : IQueryable<ApplyJob>
GetCollegeId() : int
GetCompanyId() : int
GetExperiences() : IQueryable<QC_WEB_社会经历>
GetForJob(int forjobId) : ForJob
GetForJobs() : IQueryable<ForJob>
GetJob(int jobid) : IQueryable<Job>
GetJobs() : IQueryable<Job>
GetMessages(bool includingDeleted) : IQueryable<Message>
GetMessages(int pageNum, int pageSize, bool includingDeleted) : IQueryable<Message>
GetRealName() : string
GetResume(int resumeId) : Resume
GetResumeCount() : int
GetResumes() : IQueryable<Resume>
GetUnreadMessages(bool includingDeleted) : IQueryable<Message>
GetUnreadMessages(int pageNum, int pageSize, bool includingDeleted) : IQueryable<Message>
Login(string username, string password) : int
LoginWithEncodingPassword(string loginUsername, string encodingPassword) : int
ModifyForJob(ForJob fj) : int
ModifyPassword(string oldPassword, string newPassword) : bool
ModifySelfIntroduction(string introduction) : bool
PostJob(Job job) : int
ReplyApply(int applyId, string applyMessage, bool isAccept) : bool
ReplyEmploy(int applyId, string sureMessage, bool isSure) : bool
ResetPassword(string password) : bool
SendMessage(string destUserId, string title, string content) : int
SendMessages(IEnumerable<string> destUserIDs, string title, string content) : int
SetCellphoneNumber(string cellphoneNum) : bool
SetEmailAddress(string emailAddress) : bool
SetMessageAsRead(IEnumerable<int> messageIDs) : int
StudentController(string username)
UpdateAccountBaseInfo(object inItem) : bool
UpdateForJob(ref ForJob fj) : void
UpdateResume(Resume resume) : int

学生单位。拥有各单位中最多的方法。实现 IAccount、IMessage、IForJob、IJob、IResume 接口。继承于 DetailController 类。Sealed 类。

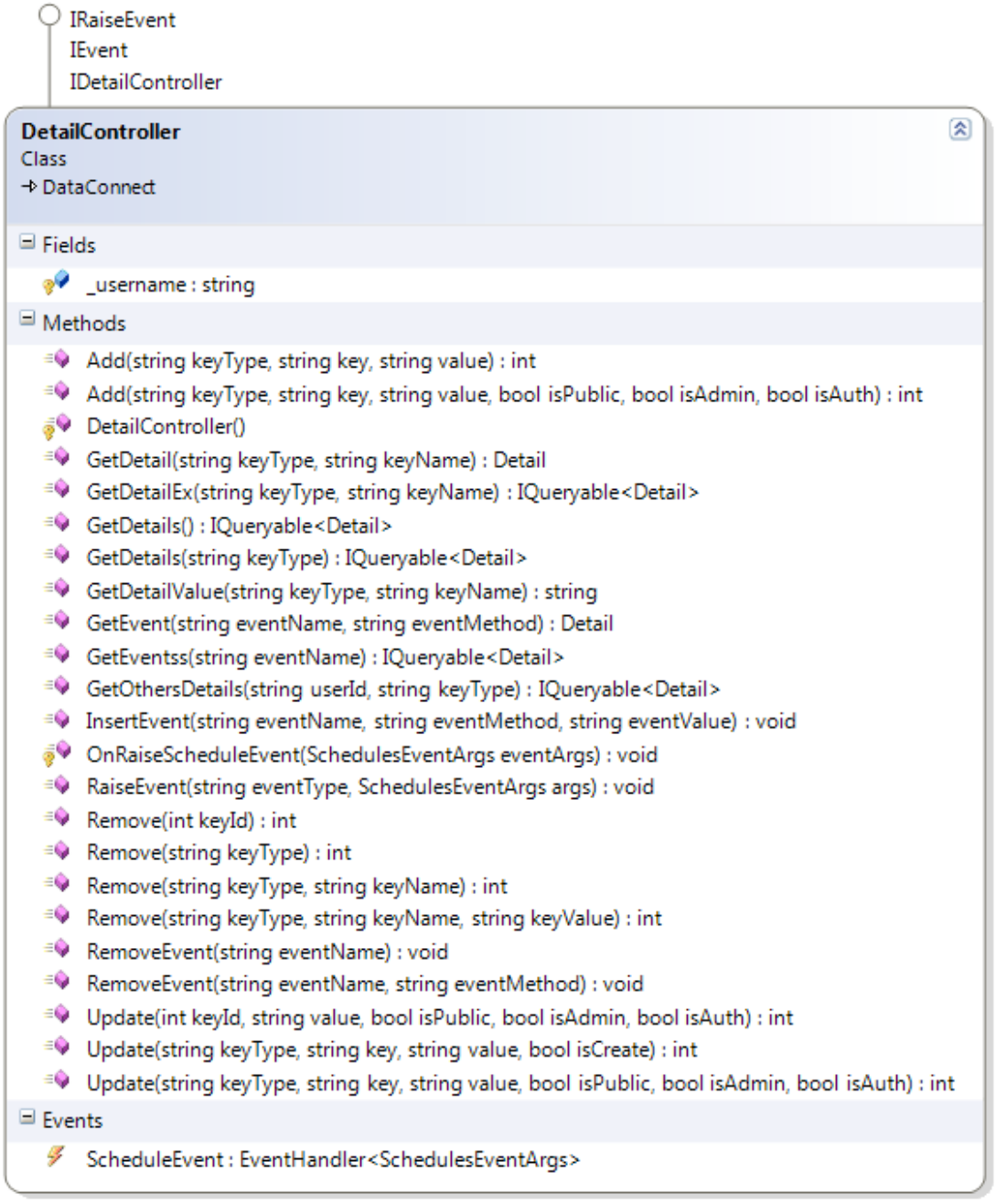


插件单位。继承于 DetailController。除了基本的数据操作外无其他功能，主要依赖插件（可扩展功能）的消息处理服务实现各种操作。Sealed 类。

独立控制

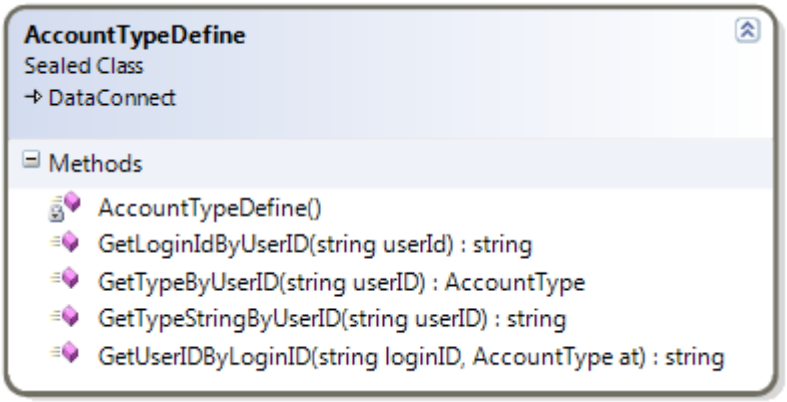


用于域名创建对应单位实例的类。根据传入用户的 ID 动态创建相应单位类的实例。（由于反射应用对系统开销过高而被摒弃）

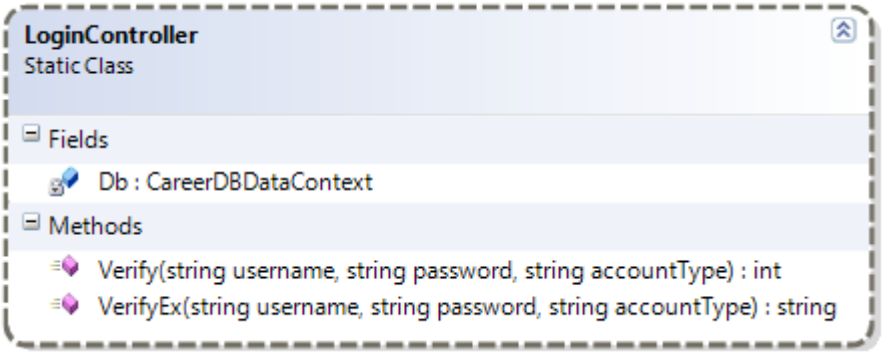


DetailController 类。各单位的父类，数据类 DataConnect 的子类。实现系统各单位的 Profile 数据存放和数据层各表操作的继承。

实例化该对象时，除插件单位外的其他单位必须使用已存在的用户 ID 作为构造函数的第一个参数传入，否则将引发类无法创建异常，该异常将被 QC（QzjCareer）代码捕获并抛出 User does not exist（用户不存在）异常，内联该类无法创建异常。



AccountTypeDefine 类。提供用户 ID、登录 ID、AccountType 结构体之间转换的各种方法。



LoginController 静态类。提供登录验证方法。

数据描述

数据层各表的封装。每一个类包含一个数据表的操作。使用 Visual Studio 2010 的

LinqToSql 工具创建。所有表封装在 CareerDBDataContext 类下。

结构体定义

QzjCareer.Core.Models

AccountType	描述各单位账号定义
BindStatus	描述手机绑定状态
DetailTimeType	描述用户私人信息临时数据存放时间
ForJobProcessType	描述学生求职状态
ProcessTypeIdType	描述学生和企业之间的应聘状态
ResumeDetailType	描述学生简历中各字段内容

QzjCareer.Core.Controller

SchedulesEventArgs 继承于 EventArgs 的 Sealed 类。
提供存放事件信息上下文数据的作用。

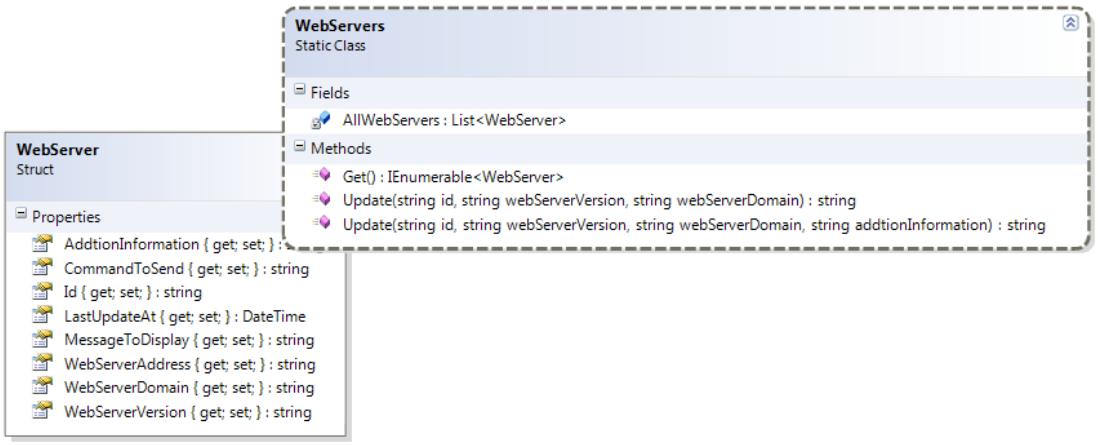
内部公开 API

功能接口

QzjCareer.Portal.CareerApi
系统功能实现的主要接口，依赖并调用核心模块的函数功能。
接口外部无需关心用户类型、操作过程、操作原理，仅传递基本参数和调用的方法即可。
开发手册请参见副页。

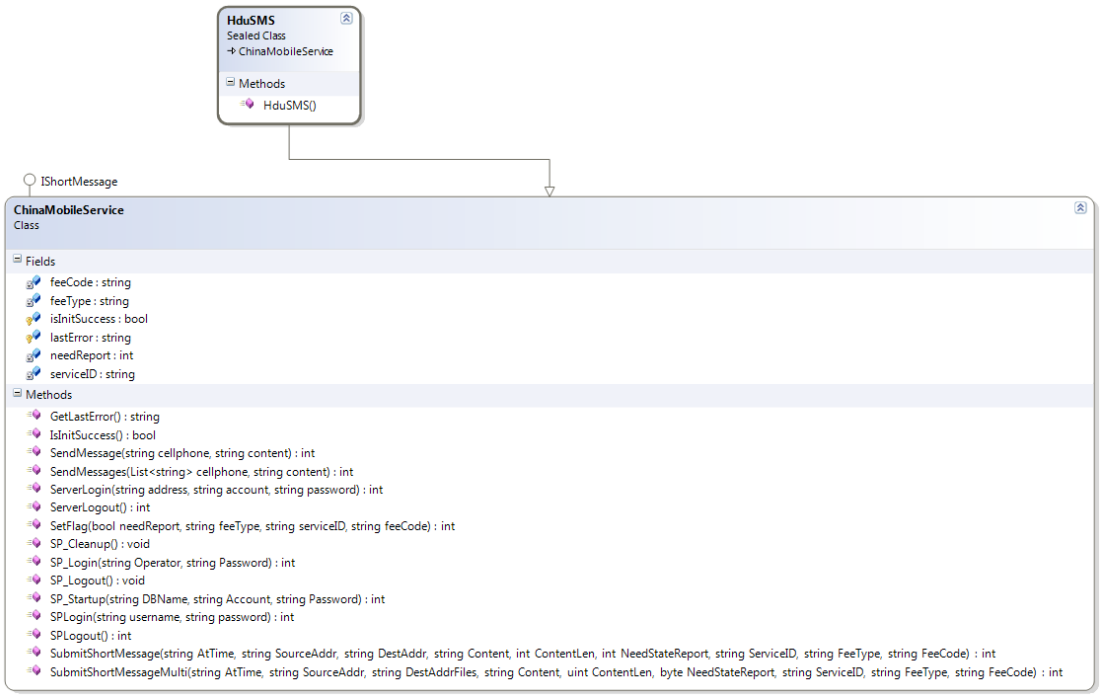
状态接口

QzjCareer.Portal.Ping
群集、分布式应用程序的 Alive 状态记录功能。提供各应用程序工作状态的监测和命令管理。各点（端）应用程序能够在管理员设定的 Ping 时间内响应中心数据库返回的命令。



QzjCareer.Portal.WebServer & QzjCareer.Portal.WebServers
以上两个类分别用于存放个在线的点（端）应用程序及服务的状态信息，例如：IP，SID，AppPoolID，AppPath 等。
遵循 QC 第三方插件规范的应用程序运行状态信息也将被存放并实时更新。

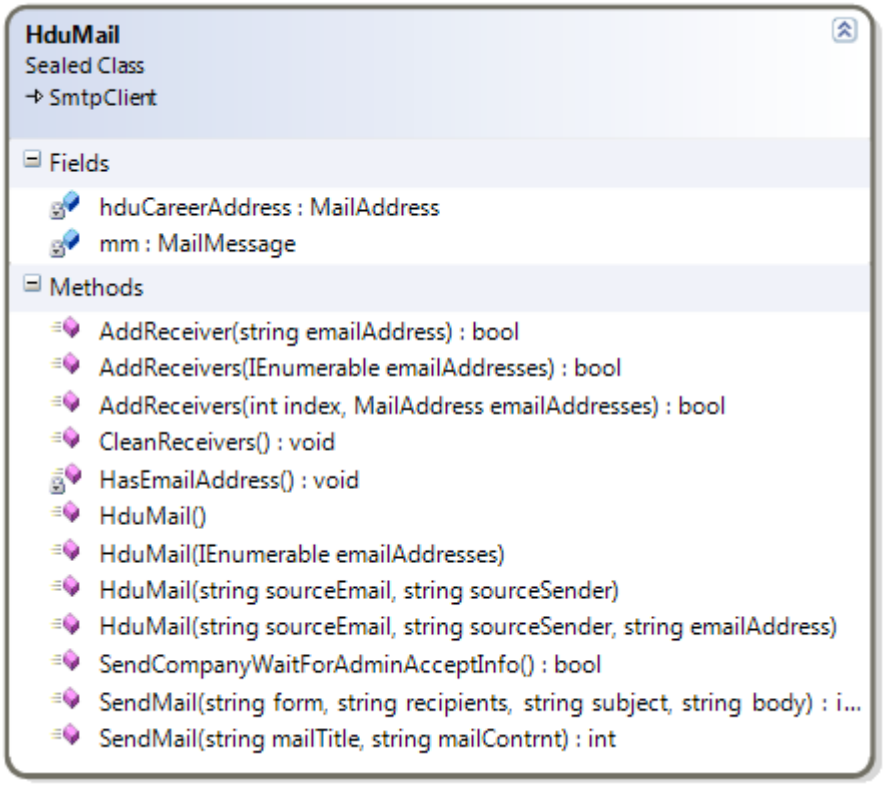
短信功能（仅 32 位）



QzjCareer.QZJSMS 封装了华为信息机的短信接口，提供更人性化的短信操作函数作为对外接口。

由于华为信息机提供的接口为 32 位非托管带指针的非安全类库，造成 QZJSMS 无法被 64 位 Runtime 调用，因此在这里短信接口只能编译为 32 位类库。

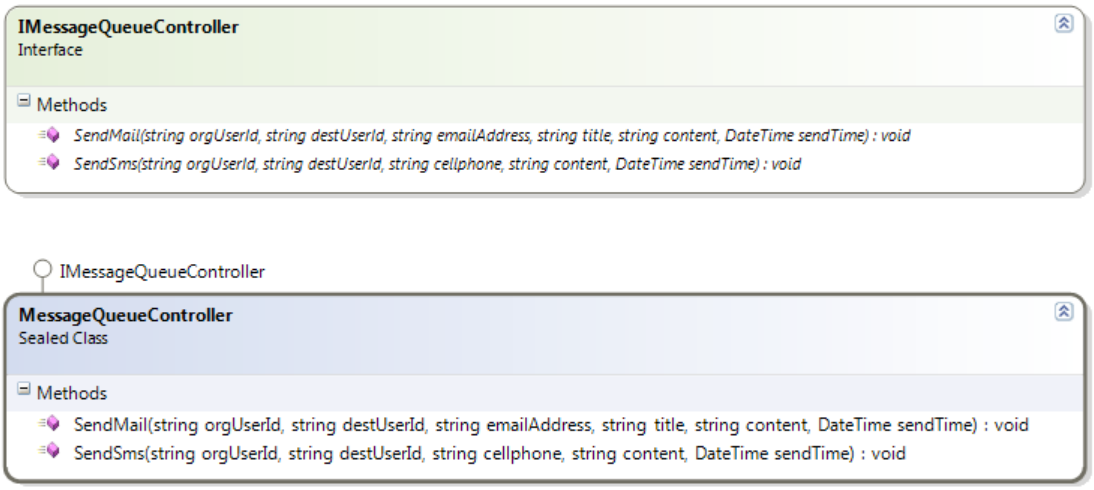
邮件功能



QzjCareer.QZJMail 封装了.NET Framework 中 SMTP 和 POP3 的邮件操作，并为 QzjCareer 对邮件发送的需求进行优化。

消息队列类库

QzjCareer.MessageQueue

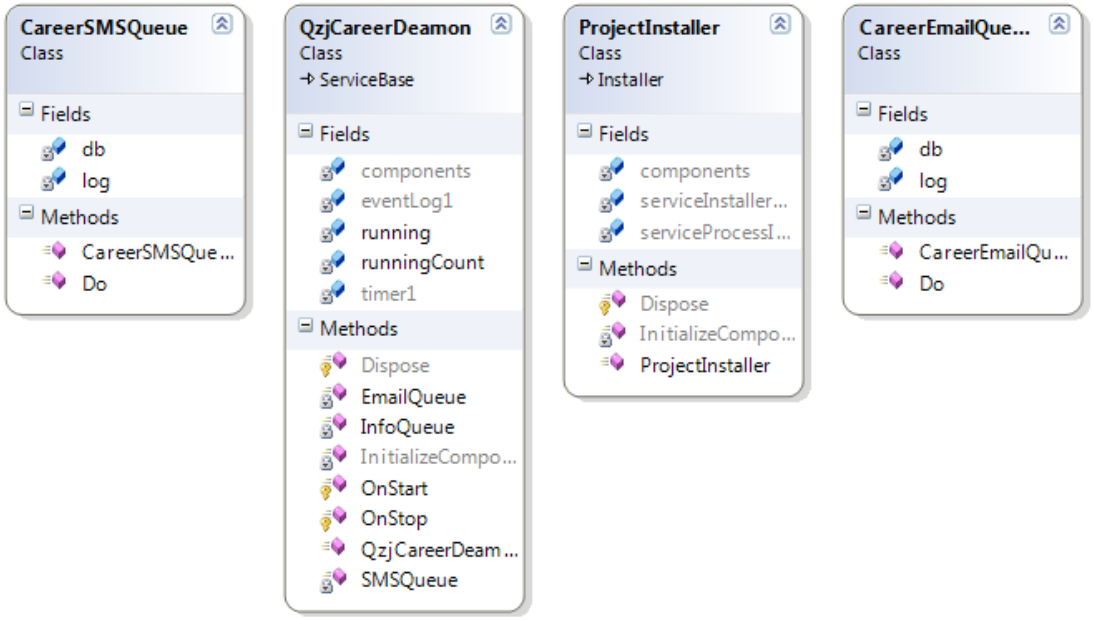


任何应用程序都可以调用 QzjCareer 消息队列类库发送邮件或者短信。

消息队列服务

结构

QzjCareer.ScheduleService



消息队列应用将作为 Windows 的服务运行。

作用

它将处理整个 QzjCareer 系统中的邮件，短信，计划任务，以及事件消息。
消息队列服务包含 QzjCareer.QZJSMS 和 QzjCareer.QZJMail，邮件和短信将通过它发送至目的地。通过 Portal 的内部 API，开发者可以二次开发自定义的消息队列。

云处理

当一个消息队列服务无法响应 QC 系统的大量消息处理时，可以在同一系统或者多个计算机节点上安装，无需单独配置，操作员可以复制在线的消息队列配置并安装在其他节点上，实现自动分布式消息处理。

分布式控制

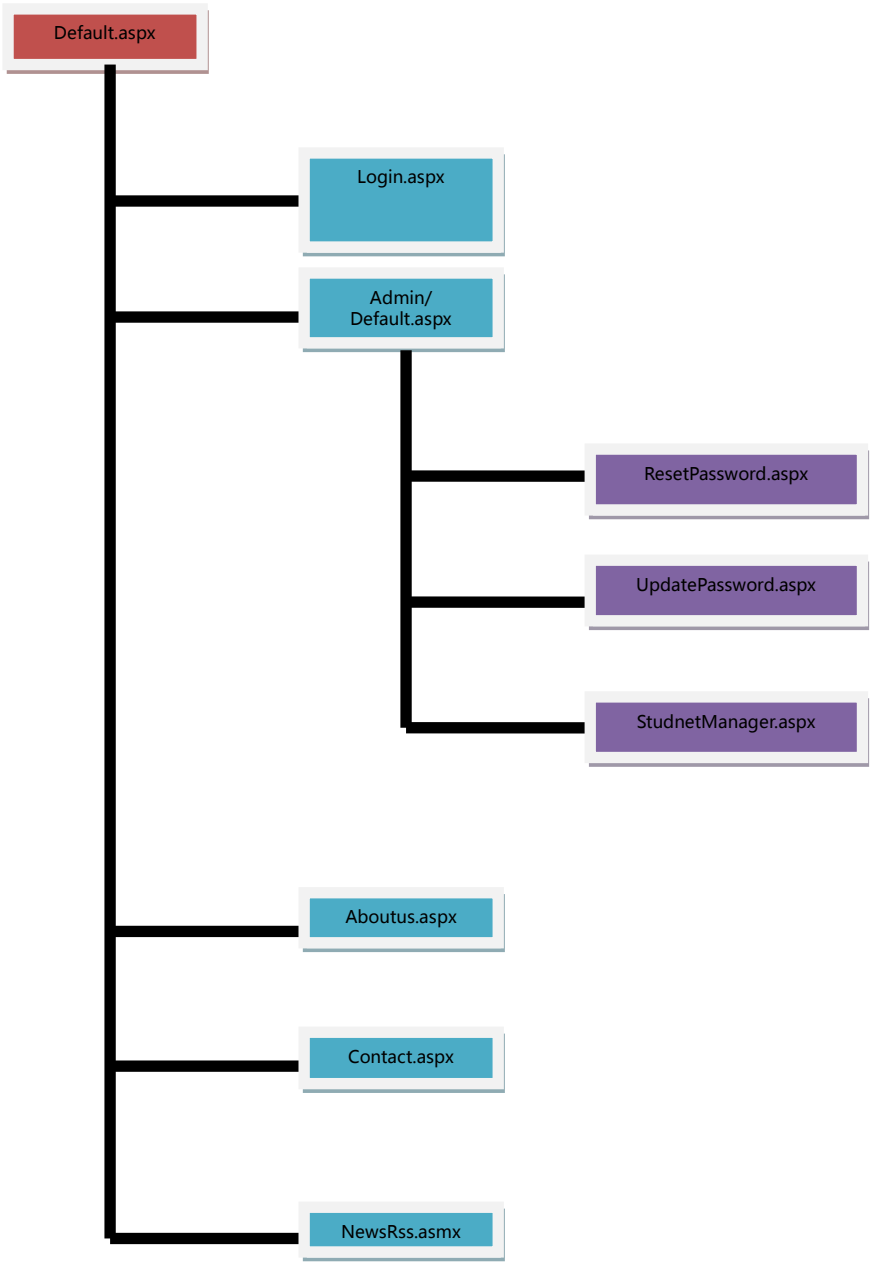
QC 消息队列服务支持热启动和关闭，无需暂停 QC 核心服务，无需考虑多个消息队列的数据冲突或者丢失。但是至少保证一个消息队列服务实例的在线以便能够及时处理 QC 系统的消息。

服务独立

QC 消息队列处理服务主要处理事件中的短信和邮件发送，另外监视系统运行状态消息，记录警告和错误状态到数据库中，因此该服务的在线对整个系统至关重要。消息队列服务的运行和数据处理不依赖 Portal，取而代之为直接连接数据库，同样使用了数据层，这样做的目的将保证 Portal 服务掉线或数据拥塞时不会影响到系统信息的处理。

关键技术

Sitemap 机制



使用到 Sitemap 主要功能有 页面标题，相关功能显示以及站点导航.

"相关功能"在页面的右上方，使用 Theme. master 进行控制，具体的读取算法为:

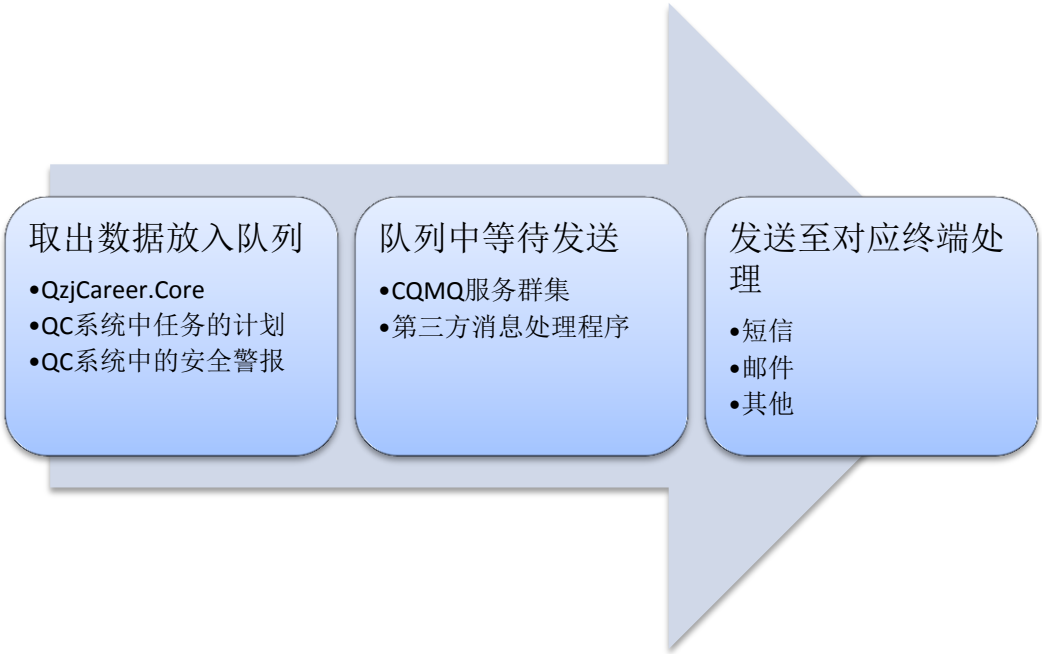
寻找当前节点的 Sitemap Node，并将其标题赋值给页面，判断当前节点是否有 1 个以上的子节点，如果有，则将该节点深度为 1 的所有子节点作为数据源提

供给"相关功能". 假如当前节点下没有 1 个以上的子节点, 则将父节点下深度为 1 的所有字节点作为数据源提供给"相关功能".

例如, 当访问 Admin/Default.aspx 时, 相关功能会显示 ResetPassword.aspx, UpdatePassword.aspx, StudnetManager.aspx 三个节点. 访问 Aboutus.aspx 时, 将显示 Login.aspx, Admin/Login.aspx, Contact.aspx, NewsRss.aspx 四个节点。

相关操作的提供有助于提高用户访问操作的体验。

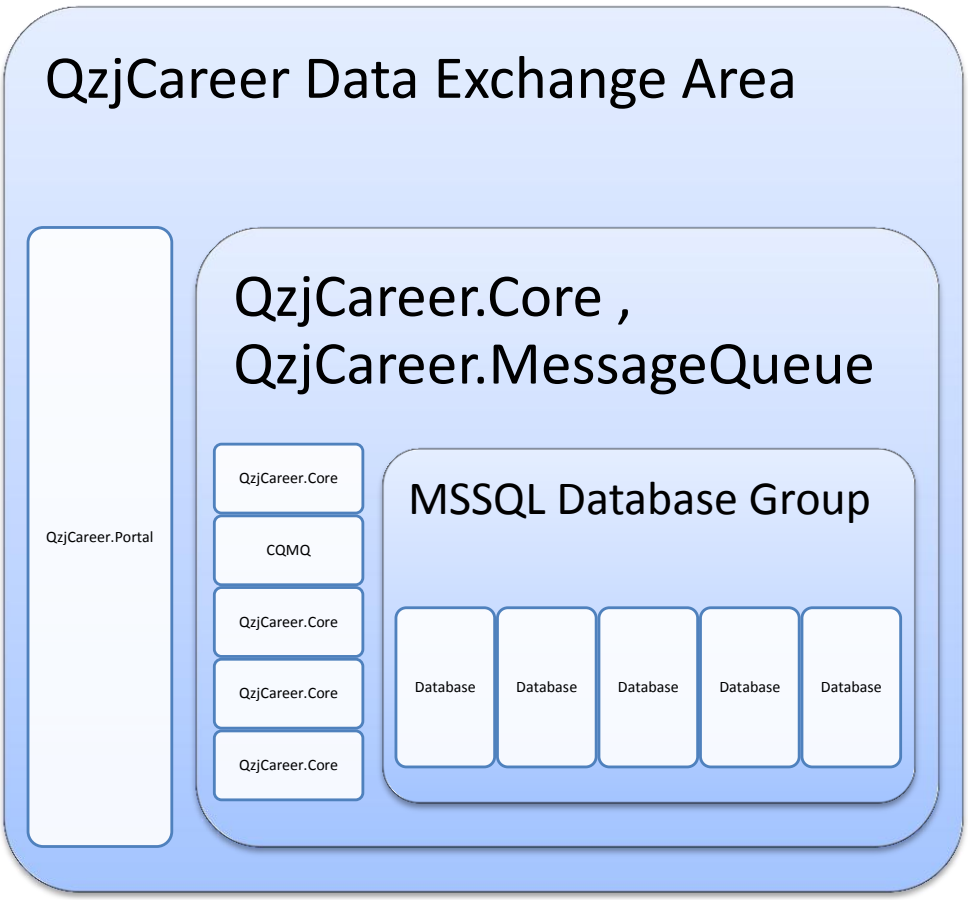
QCMQ 消息队列



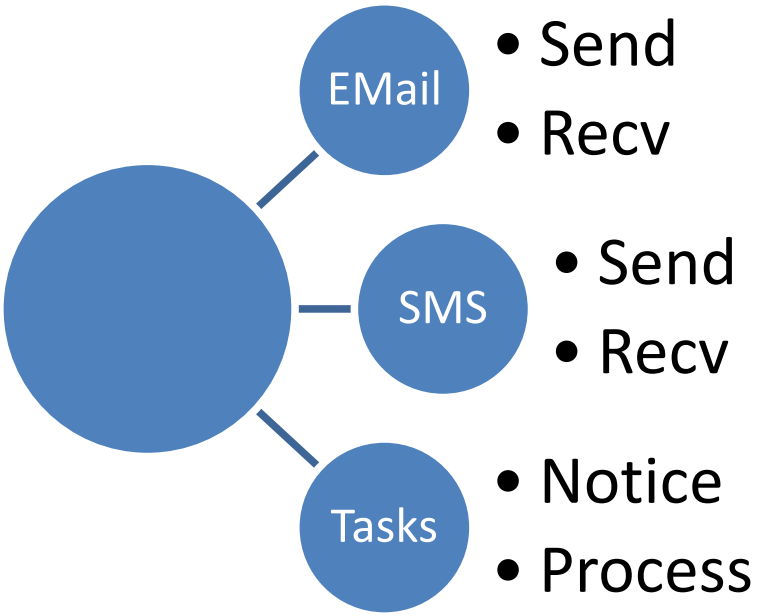
QC 系统中针对于第三方消息处理分离 消息队列以 Windows 服务方式运行，支持傻瓜式分布处理配置，无需配置服务器参数便能实现云处理。
分布式计算，云计算优点及搭建方法请参考附录。

- 取出数据放入队列
  - QzjCareer.Core
  - QC 系统中任务的计划
  - QC 系统中的安全警报
- 队列中等待发送
  - CQMQ 服务群集
  - 第三方消息处理程序
- 发送至对应终端处理
  - 短信
  - 邮件
  - 其他

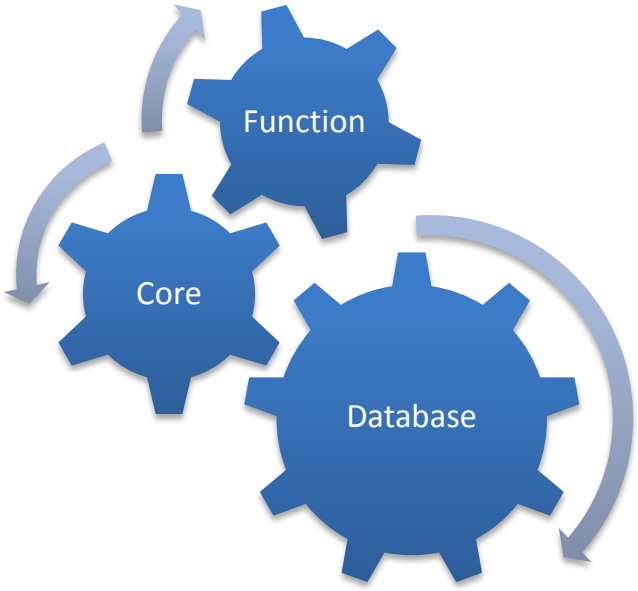
数据逻辑处理部分



为了提高系统的可扩展性和兼容性，同时为云处理技术而优化，系统采用分层设计方式。数据库采用多个 MSSQL 协通工作处理大量的数据信息。

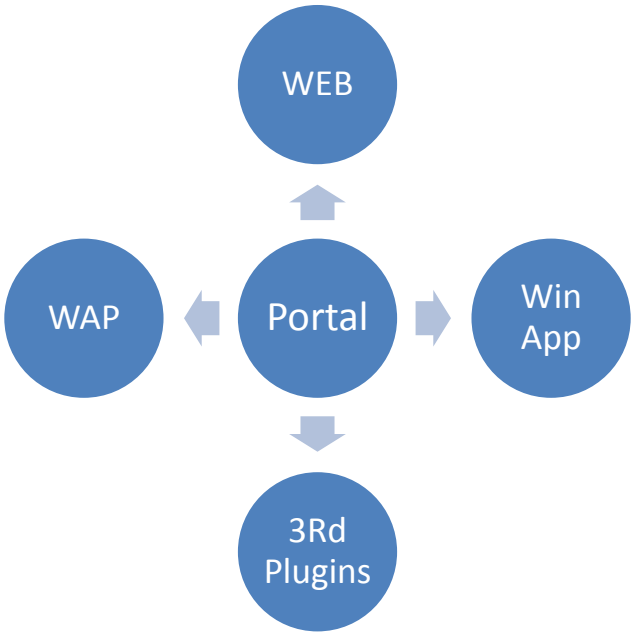


QCMQ (QzjCareer.MessageQuere)直接处理数据库中等待处理的短信，邮件等外

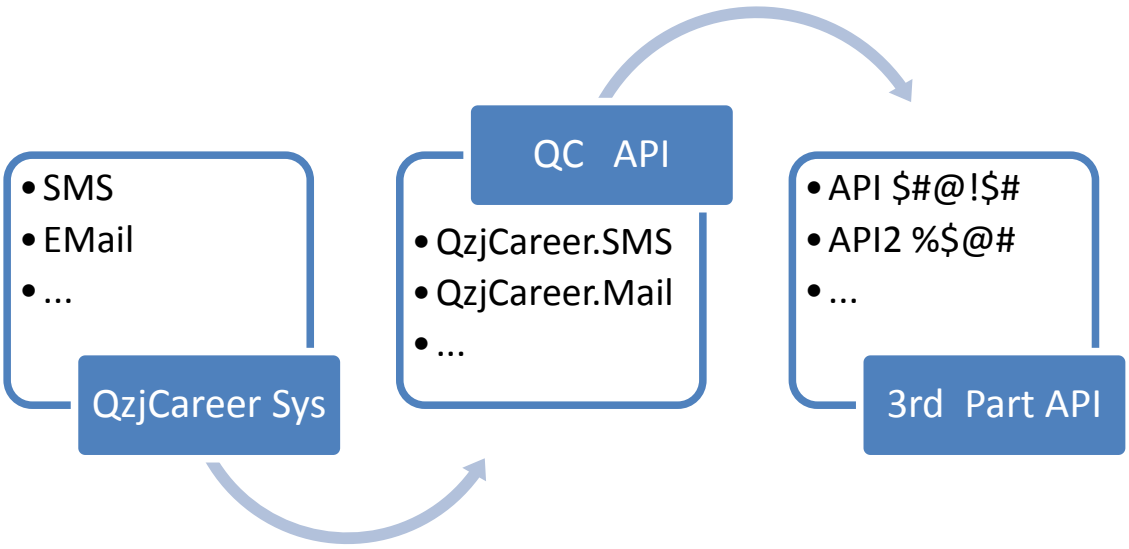


部信息，同时监视日历数据，发出提醒事件。有效解决处理大规模数据时终端拥塞线程假死。

QzjCareer.Core 作为整个系统功能与数据库连接的核心部分，操作用户数据库唯一的途径，内建数据错误处理确保数据完整。同时对部分程序员对其二次开发过程中可能错误度地调用方法造成潜在致命的数据逻辑错误前发出提醒。



QzjCareer.Portal 作为整个系统的核心对内功能 API，提供所有系统操作所需要的函数，集成用户帐户角色安全，通过 XML 数据与外部程序进行交互。让程序员对 QzjCareer 系统进行二次开发，扩展功能更轻松方便。

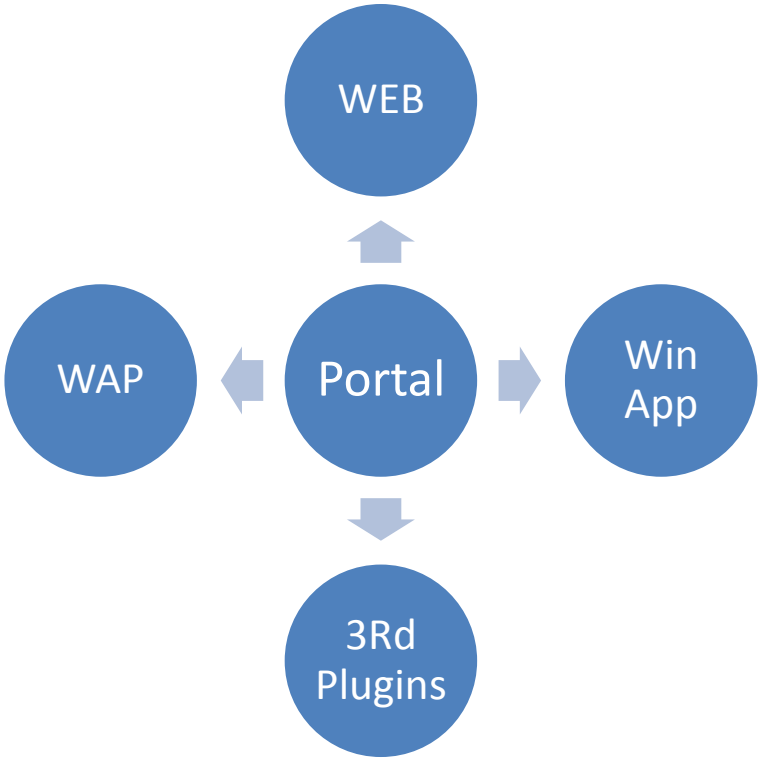


QzjCareer.SMS 、QzjCareer.Mail 给系统以及第三方开发人员提供了简单易懂的参数配置和友好的第三方接口。

简单配置便能使程序连接华为短信信息机接口和符合国际 ISO 标准的 SMTP，POP3 邮件服务器。

内部开放 API 接口

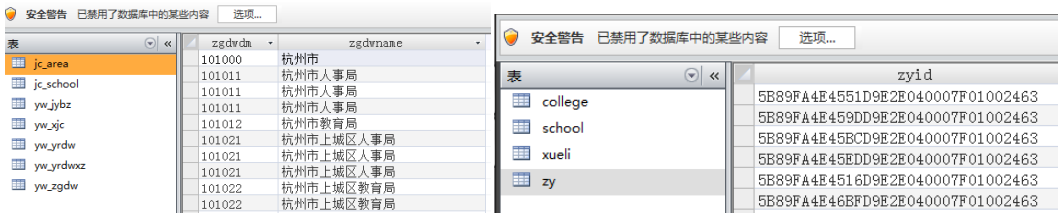
由于 QC 平台对内接口的开放性，致使第三方扩展功能都能在其基础上完成操作.



QC Portal 采用 XML 标准对数据进行序列与反序列化. XML 可以同时用来描述字段,定义字段,和包含字段的值，给第三方开发以及系统自身扩展提供了简便快捷的基础.
第三方插件（扩展）同样可以注册和获得自定义的 QC 系统事件。

数据规范

省级数据库格式支持



以上表格为浙江省大学生网上就业市场提供教师下载填写班级信息、学生信息、学院信息的数据库。

QC 系统支持并兼容该数据格式，将数据全部导入 QC 系统相同的表名后执行 QC 数据库工具的用户体验数据初始化
更新表数据时，删除上述所有表，重新添加。包括表结构和表数据，同时在所有表中添加以下两行字段
[chineseSpell] [varchar](50) NULL,
[chineseFirst] [varchar](50) NULL
QC 系统将用其存放优化用户体验的数据。

注意：[浙江省大学生网上就业市场]提供的 Access 数据库并没有设定主键，但是 QC 系统中必须设定主键！不能以存放重复数据的字段作为主键。

学生在 QC 系统录入的获奖信息，个人信息，个人爱好等个人简历信息可以导出为[浙江省大学生网上就业市场]的模板数据，免去手工录入信息的时间和精力。

Outlook 日历订阅

校园日历和个人日历均提供 ICal 与 XML 格式的数据源，任何兼容以上两种数据源的程序（例如 Outlook）能够实时更新电脑或手持设备的个人日历以及校园日历。

QC 系统事件

QC Event 规范
Event 使用 Details 表存放与读取数据。
Event 的动态属性有 事件名称，方法，值。
Event 静态属性为 用户标示 ID，用户类型。

Event 与 Detail 表对应关系

Event.EventName -> Detail.DetailType
Event.EventMethod -> Detail.DetailName
Event.EventValue -> Detail.DetailValue

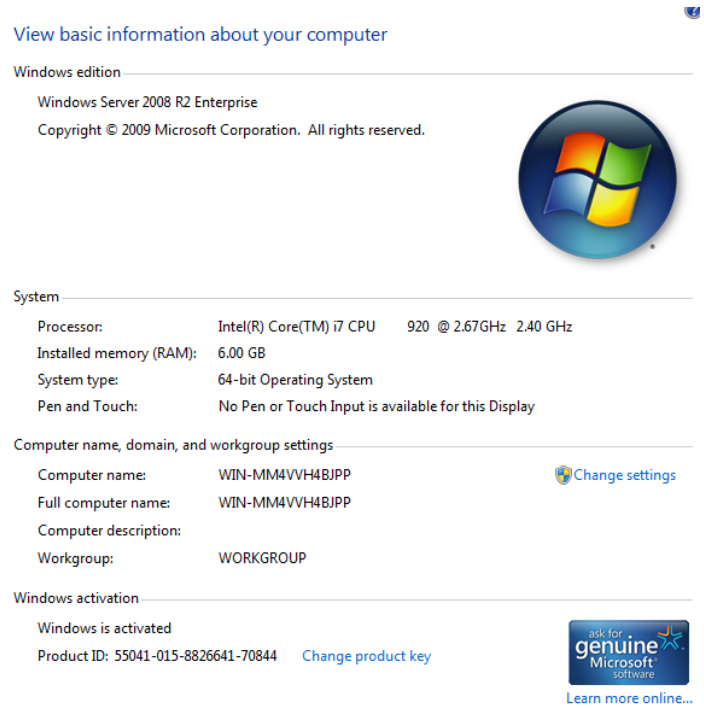
事件名称： 当发生该事件引发。（e.g. Event_XXX)
事件方法： 引发后对事件进行的操作。(e.g. SMS)
事件值： 对该事件操作的数据。
事件值格式：目标地址|内容[|...]
|为每项值分隔符，在添加事件处理条目时必须对值中已存在的|进行转义或者删除。
在第三方开发过程中，务必检查事件值格式的各项值中是否存有与分隔符相同的值。

例如：

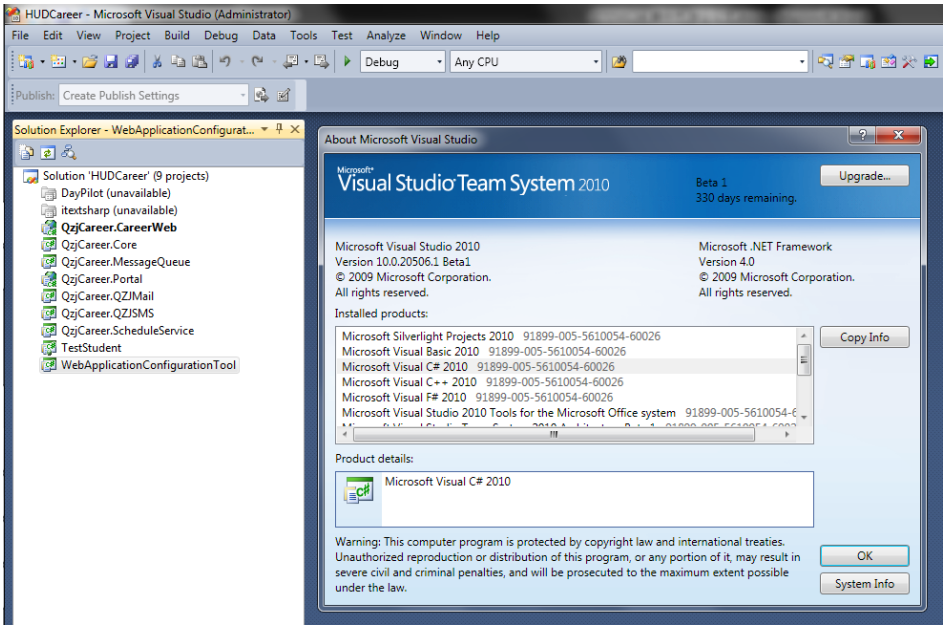
事件类型[引发时间源]	Event_ApplyJob
事件方法[消息处理类型]	SMS
事件值[事件发生时数据]	13588860000|维护人员的应聘状态已经改变为：初次应聘成功，等待面试。|进入面试状态

当学生或者企业中用户的应聘状态改变时，将引发名称为 Event_ApplyJob 的事件，系统将对订阅该事件的对象产生相关数据放入[SYS_Schedules]表中。消息队列将对支持的方法进行处理，第三方插件从[SYS_Schedules]表中处理自身支持的信息并将反馈信息和处理结果保存。
具体信息请参阅附录的开发手册。

开发过程

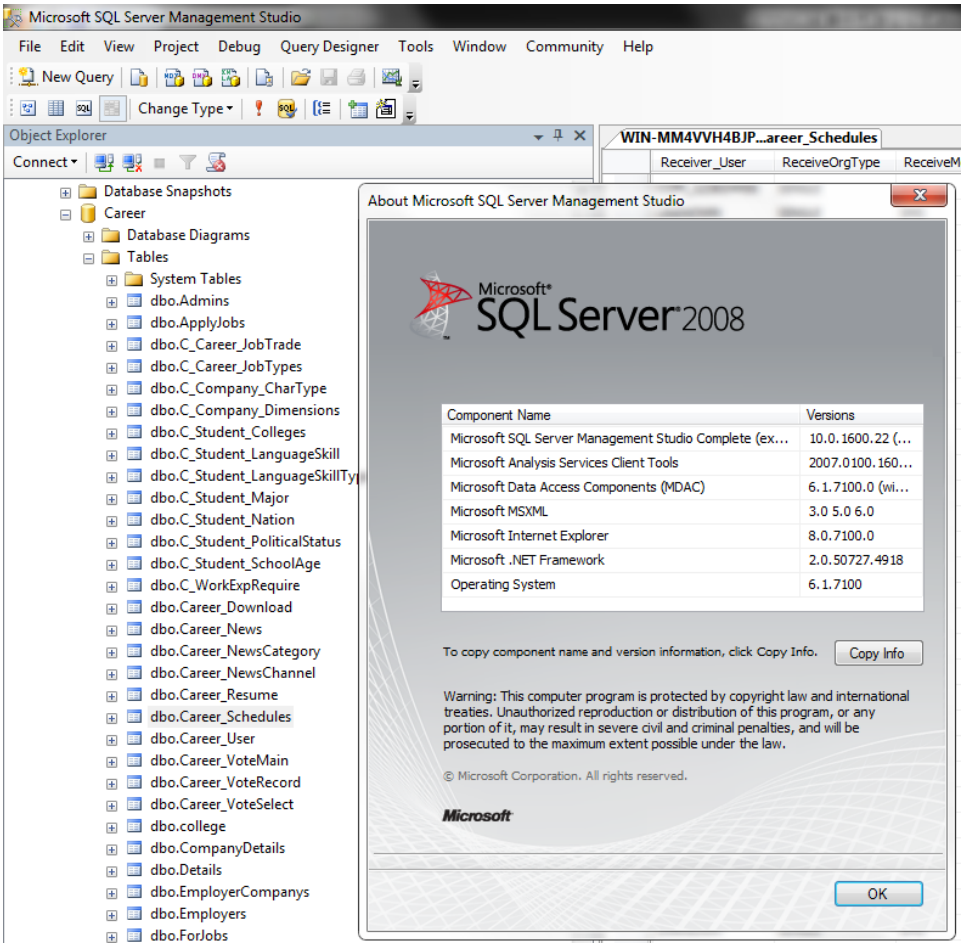


Windows 2008 R2 企业版操作系统。



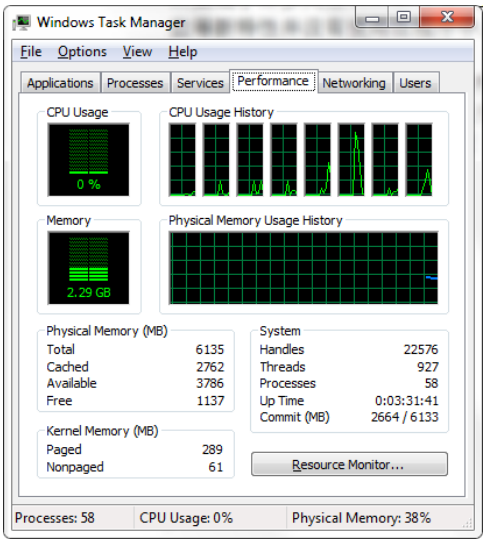
软件开发前期使用 Visual Studio 2008 Team System。

后期开发及调试基于 Visual Studio 2010 Team System，并转化为.Net 4.0 应用程序。



数据库使用 SQL Server 2008。

之所以选择 Visual Studio 2010 来做开发，因为其强大的.Net Framework 平台支持，高效的编程体验，同时拥有舒适的界面和完美的透明界面。.Net Framework 4.0 种集成了许多人性化的新特性，鉴于其现在仍然是测试版本，新的控件和域名函数等新特性并没有使用在程序中。



在开发就业招聘系统中，应用程序的调试启动速度一直是困扰本人开发效率的问

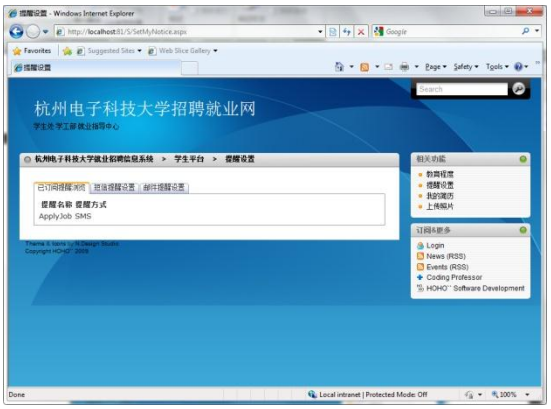
题之一。调试状态需要装载各托管代码的签名及符号，在笔记本上开发的 Debug 模式比 Release 模式要多消耗近 15 倍的时间。为此更换了台式电脑，优化代码支持多核处理器。

版本控制对开发大型项目有效管理开发进度及任务的方法之一。通过自己架设的 SVN，每次更新将更新的内容作为 Comments 传送到 SVN 服务器中。在未来浏览之前修改的代码会使过程思路非常清晰，不会因为文件、代码的繁多而晕头转向。

部分功能浏览

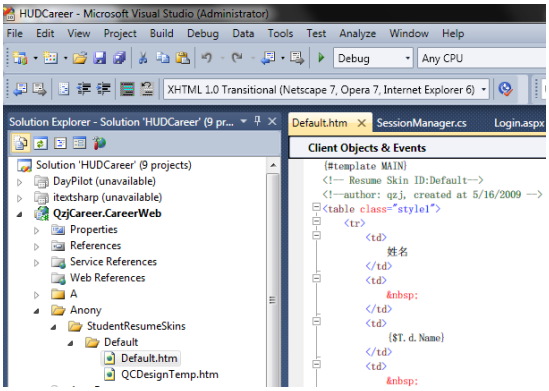
交互界面

采用无验证码的浏览器计算动态数据验证来判断登录异常。


杭州电子科技大学招聘就业网
学生处 学工部 就业指导中心
杭州电子科技大学就业招聘信息系统 > 学生平台 > 提醒设置
相关功能
订阅&更多
Login
News (RSS)
Events (RSS)
Coding Professor
HOHO'' Software Development

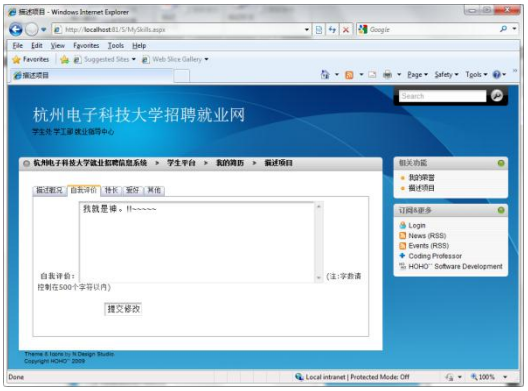

杭州电子科技大学招聘就业网
学生处 学工部 就业指导中心
杭州电子科技大学就业招聘信息系统 > 学生平台 > 提醒设置
应聘状态改变
私人消息接收
我的订阅
相关功能
订阅&更多
Login
News (RSS)
Events (RSS)
Coding Professor
HOHO'' Software Development


杭州电子科技大学招聘就业网
学生处 学工部 就业指导中心
杭州电子科技大学就业招聘信息系统 > 学生平台 > 我的简历
相关功能
订阅&更多
Login
News (RSS)
Events (RSS)
Coding Professor
HOHO'' Software Development

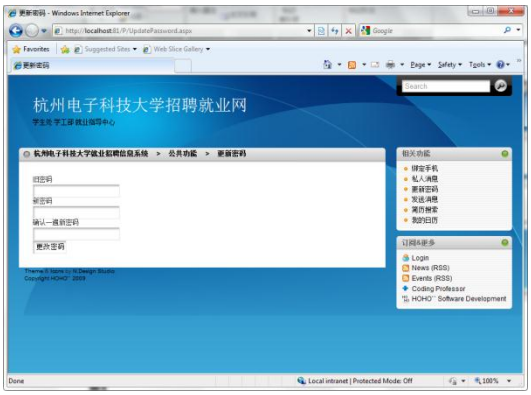

HUDCareer - Microsoft Visual Studio (Administrator)
File Edit View Project Build Debug Data Tools Test Analyze Window Help
Debug
Any CPU
XHTML 1.0 Transitional (Netscape 7, Opera 7, Internet Explorer 6)
Solution Explorer - Solution 'HUDCareer' (9 pr...
Solution 'HUDCareer' (9 projects)
DayPilot (unavailable)
itextsharp (unavailable)
QzjCareer.CareerWeb
Properties
References
Service References
Web References
A
Anony
StudentResumeSkins
Default
Default.htm
QCDesignTemp.htm
Default.htm
SessionManager.cs
Login.aspx
Client Objects & Events
<table class="style1">
姓名
{$T. d. Name}

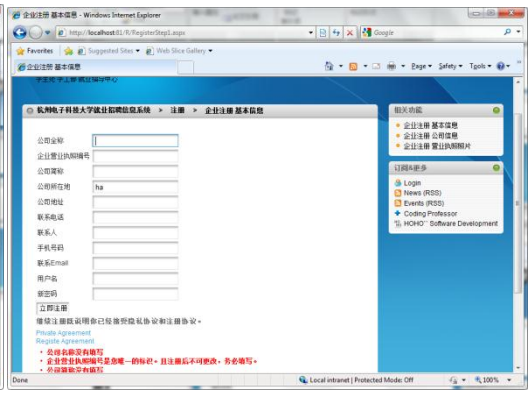

杭州电子科技大学招聘就业网
学生处 学工部 就业指导中心
杭州电子科技大学就业招聘信息系统 > 学生平台 > 我的简历 > 我的荣誉
相关功能
订阅&更多
Login
News (RSS)
Events (RSS)
Coding Professor
HOHO'' Software Development


杭州电子科技大学招聘就业网
学生处 学工部 就业指导中心
杭州电子科技大学就业招聘信息系统 > 学生平台 > 我的简历 > 我的荣誉
相关功能
订阅&更多
Login
News (RSS)
Events (RSS)
Coding Professor
HOHO'' Software Development

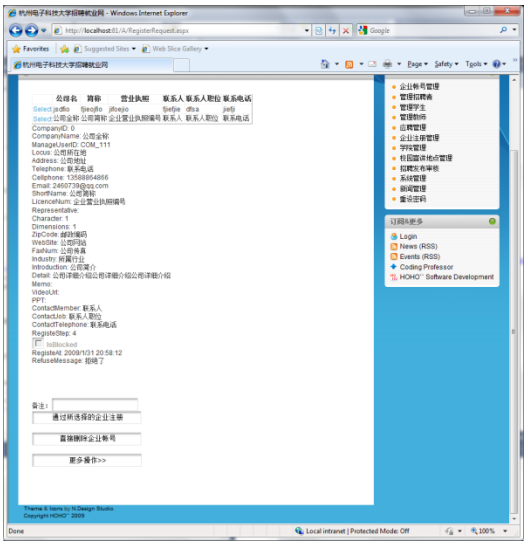

杭州电子科技大学招聘就业网
学生处 学工部 就业指导中心
杭州电子科技大学就业招聘信息系统 > 学生平台 > 我的简历 > 描述项目
相关功能
订阅&更多
Login
News (RSS)
Events (RSS)
Coding Professor
HOHO'' Software Development


杭州电子科技大学招聘就业网
学生处 学工部 就业指导中心
杭州电子科技大学就业招聘信息系统 > 学生平台 > 上传照片
相关功能
订阅&更多
Login
News (RSS)
Events (RSS)
Coding Professor
HOHO'' Software Development


更新密码 - Windows Internet Explorer
杭州电子科技大学招聘就业网
学生处 学工部 就业指导中心
杭州电子科技大学就业招聘信息系统 > 公共功能 > 更新密码
旧密码
新密码
确认一遍新密码
更改密码
相关功能
绑定手机
私人消息
更新密码
发送消息
简历搜索
我的日历
订阅&更多
Login
News (RSS)
Events (RSS)
Coding Professor
HOHO'' Software Development


企业注册 基本信息 - Windows Internet Explorer
杭州电子科技大学就业招聘信息系统 > 注册 > 企业注册 基本信息
公司全称
企业营业执照编号
公司简称
公司所在地
公司地址
联系电话
联系人
手机号码
联系Email
用户名
新密码
立即注册
继续注册就表明你已经接受隐私协议和注册协议。
Private Agreement
Registe Agreement
相关功能
企业注册 基本信息
企业注册 公司信息
企业注册 营业执照照片
订阅&更多
Login
News (RSS)
Events (RSS)
Coding Professor
HOHO'' Software Development


杭州电子科技大学就业招聘信息系统 > 管理员 > 企业注册管理
公司名 简称 营业执照 联系人 联系人职位 联系电话
相关功能
企业帐号管理
管理招聘者
管理学生
管理教师
应聘管理
企业注册管理
学院管理
校园宣讲地点管理
招聘发布审核
系统管理
新闻管理
重设密码
订阅&更多
Login
News (RSS)
Events (RSS)
Coding Professor
HOHO'' Software Development
重置企业注册
邮件通知
短信通知
重置企业注册

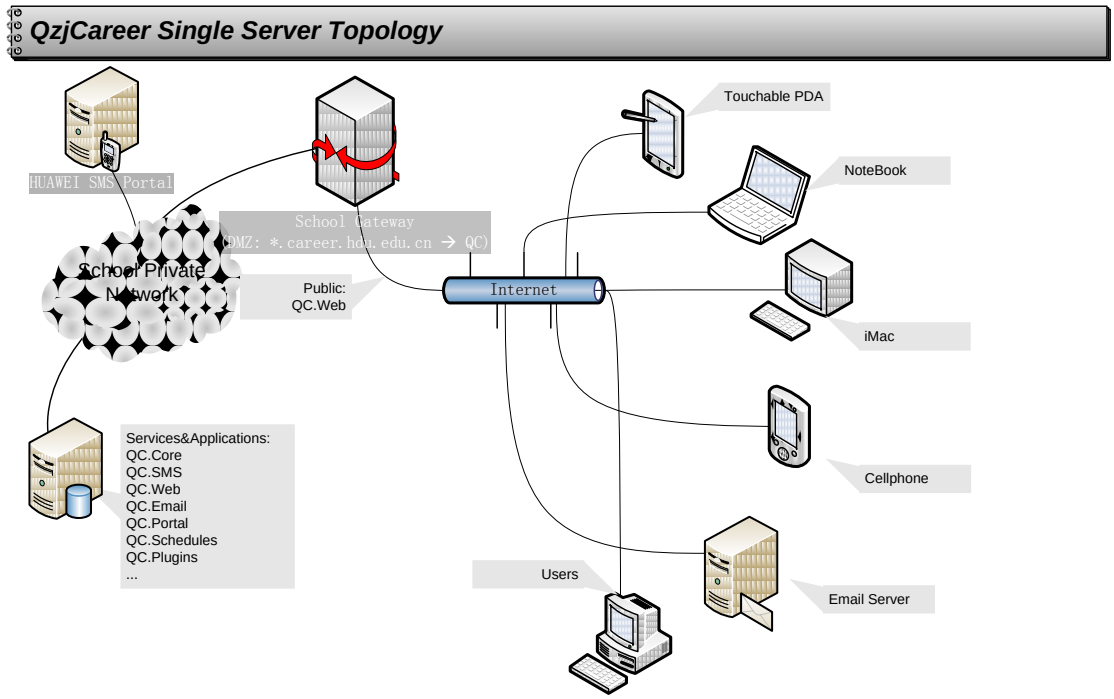

杭州电子科技大学招聘就业网 - Windows Internet Explorer
公司名 简称 营业执照 联系人 联系人职位 联系电话
相关功能
企业帐号管理
管理招聘者
管理学生
管理教师
应聘管理
企业注册管理
学院管理
校园宣讲地点管理
招聘发布审核
系统管理
新闻管理
重设密码
订阅&更多
Login
News (RSS)
Events (RSS)
Coding Professor
HOHO'' Software Development
通过所选择的企业注册
直接删除企业帐号
更多操作>>

安装及使用

架构选择

QzjCareer 系统可运行在多种网络拓扑方式以及多种系统架构,选择需依据用户同时访问量及硬件性能而定。
以下推荐三种配置方案供安装参考。

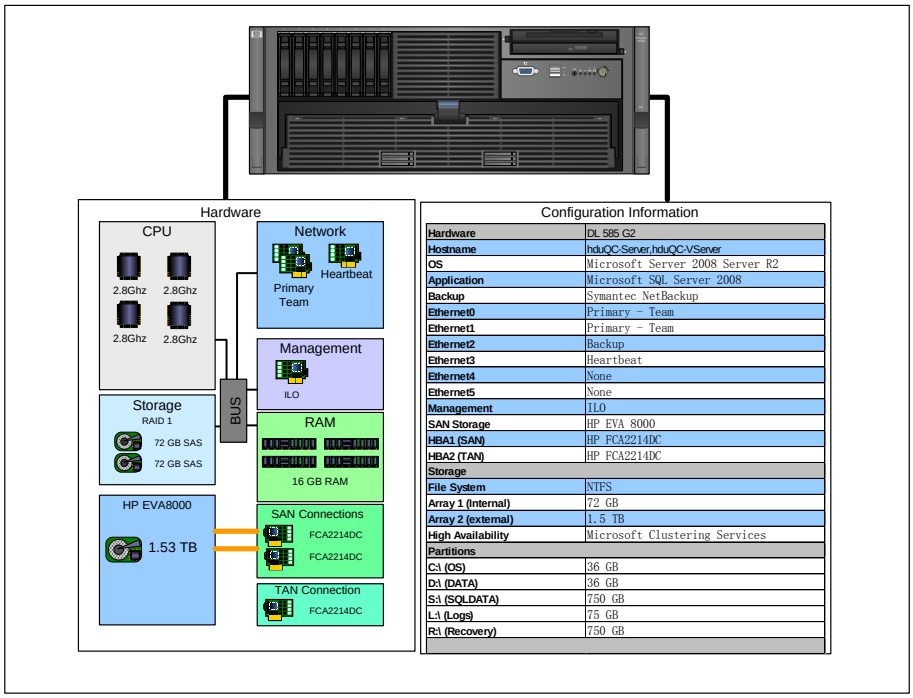
单一服务器无虚拟化配置



该方案适合访问量波动稳定,流量较小,硬件配置较好的单(双)核服务器.由单一的服务器上运行单一操作系统,同时在线 QC 核心服务，数据库服务，QCAPI 服务，Email 接口，SMS 接口，QC 消息队列服务，QC 计划任务服务，QC 扩展应用程序。

硬件配置最低要求:
CPU: Intel Xeon 2.0G
RAM: 4GB
Disk Space: 1TB

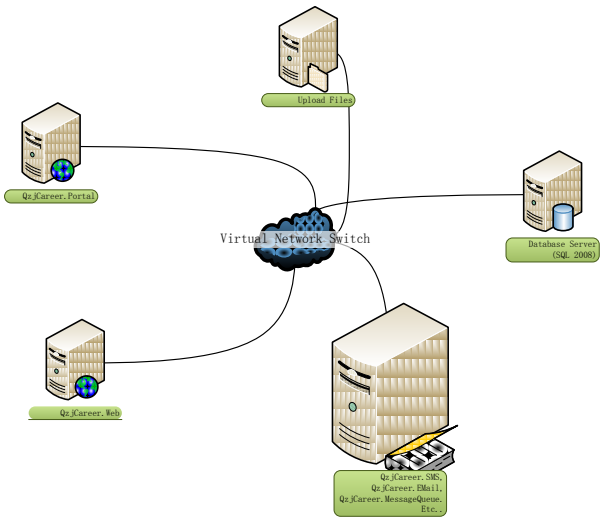
单一服务器虚拟化配置



Hardware	DL 585 G2
Hostname	hduQC-Server,hduQC-VServer
OS	Microsoft Server 2008 Server R2
Application	Microsoft SQL Server 2008
Backup	Symantec NetBackup
Ethernet0	Primary - Team
Ethernet1	Primary - Team
Ethernet2	Backup
Ethernet3	Heartbeat
Ethernet4	None
Ethernet5	None
Management	ILO
SAN Storage	HP EVA 8000
HBA1 (SAN)	HP FCA2214DC
HBA2 (TAN)	HP FCA2214DC
Storage	
File System	NTFS
Array 1 (Internal)	72 GB
Array 2 (external)	1.5 TB
High Availability	Microsoft Clustering Services
Partitions	
C:\ (OS)	36 GB
D:\ (DATA)	36 GB
S:\ (SQLDATA)	750 GB
L:\ (Logs)	75 GB
R:\ (Recovery)	750 GB

上图为假定的宿主机配置。在宿主机上的虚拟控制化设备中建立 5 个客户机，如下图。

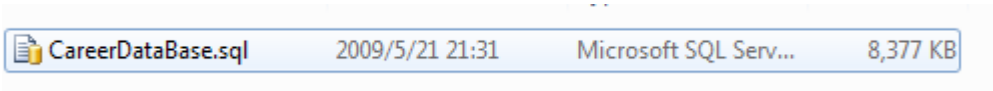
Hyper-v, QzjCareer



每个客户机分配虚拟网络的独立 IP 地址，通过 NAT 技术连接到宿主机的物理网络上。同时宿主机需要指定 NAT 上的应用程序服务端口对应内部虚拟主机的服务端口。在这个配置中，需要在虚拟交换机 NAT 映射设置物理网络 80 端口对应 QzjCareer.WEB 主机的 WEB 服务端口才能够接受外部的传入连接。

在该配置下的宿主机必须严格配置安全策略，当入侵进入宿主机时，他将轻易获得虚拟环境下的任何设备信息数据。

数据库搭建

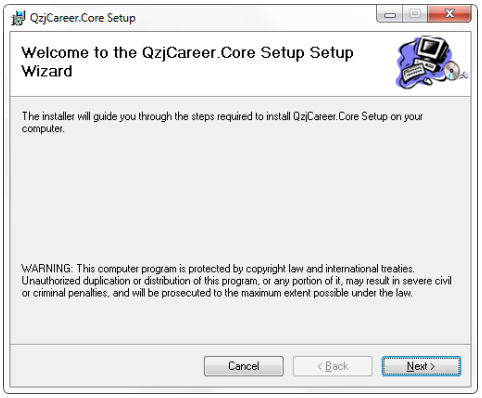


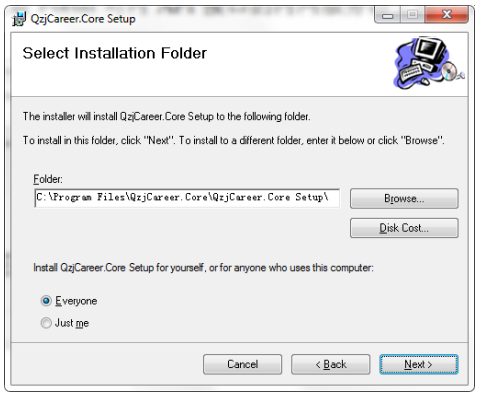
1. 安装 Microsoft SQL Server 2005 或者以上版本，推荐使用 Microsoft SQL Server 2008 以支持全部编录。
2. 进入 SQL 管理工具，New Database[1]并设定期望的数据库参数和文件存放位置。
3. 选择 New Query[2]数据脚本，并导入数据库脚本文件，文件位置在 InstallPacket\MSSQL Script 目录下的 CareerDataBase.Sql。
4. 导入完成后，生成对应的数据库连接语句准备其他应用程序配置时使用。

值得注意的是，数据库默认情况下使用 Windows 验证方式。如果维护人员希望使用 TCP 登录方式，请右键管理工具中的 Object Explorer[3]选择属性。在 Security[4]选项卡中更改 SQL 允许的验证方式为 SQL Server and Windows。完成后检查能够从外部该机的外部网络访问，如遇失败则检查防火墙设置。

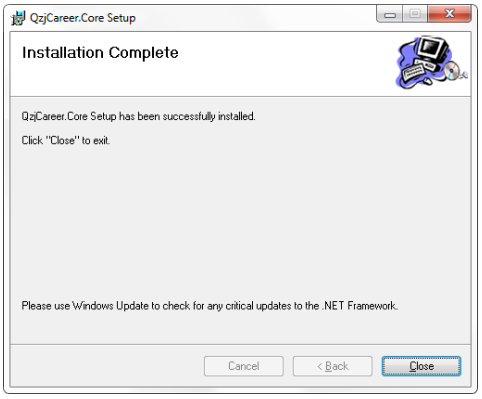
核心模块安装

在需要安装 Portal 对内 API 接口的内网服务器上执行 SetupQzjCareerCore.msi。





[1] New Database:新建数据库。
[2] New Query:新建查询。
[3] Object Explorer:对象浏览器。
[4] Security:安全。



安装完成。依赖 QzjCareer.Core 库的组件便能够在该系统平台上运行。

系统接口安装

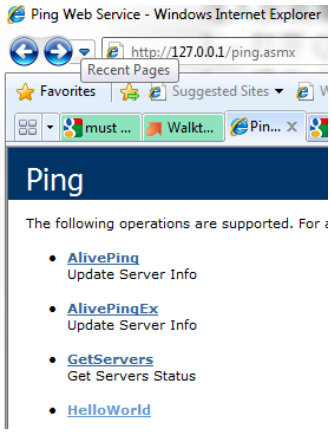
Portal 内部接口依赖 QzjCareer.Core 库，未安装 QzjCareer.Core 库的系统平台上无法正确运行借口。

进入安装媒体[5]中的 SetupQzjCareerPortal 目录，依次执行以下步骤：

1. 打开 QzjCareer.Portal.parameters.txt 文件。
2. 设定 QzjCareer.Portal_IisWebApplication 的值为 Portal 在 IIS 中应用程序的名称。
3. 设定 QzjCareer.Portal_CareerDB_ConnectionString 为数据库连接字符串。如果数据库存在数据需备份，安装 Portal 时将会清空之前的数据。
4. 打开 IIS 管理工具，创建以修改 QzjCareer.Portal_IisWebApplication 后的值为名称的网站项目，并配置好基本信息，将.Net 应用程序池的工作版本设置为 4.0。
5. 执行 QzjCareer.Portal.deploy.cmd [/T|/Y]。如需得到参数说明，直接执行 QzjCareer.Portal.deploy.cmd。
   提示：首先确保当前执行安装的用户的环境变量里添加有 msdeploy 的 path 变量，如果不存在，请查阅相关 Windows 使用技巧，加入 path 变量中 msdeploy 的路径。或者维护人员也可以在命令行模式下执行 set Path=%Path%;"C:\Program Files (x86)\IIS\Microsoft Web Deploy"[6]。
6. 如果没有出现错误提示，安装就完成了。

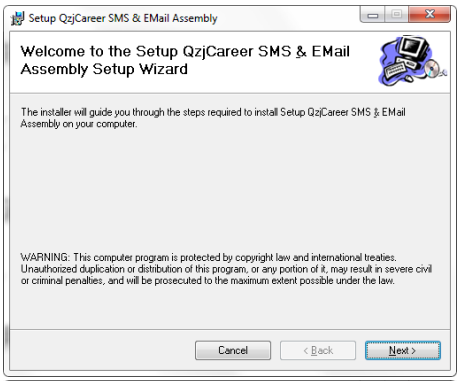
打开 http://localhost/ping.asmx 检查是否可以正常访问。Portal 在线将会出现如图中内容。注意，Portal 能够正常显示并不说明已正常连接到数据库。

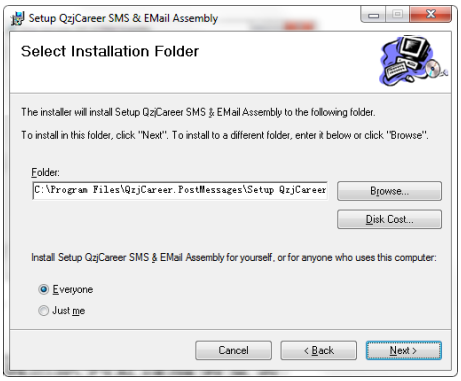
[5] 安装媒体为含有 QzjCareer 安装文件的光盘或者镜像文件。
[6] 系统中 msdeploy 的文件夹路径。

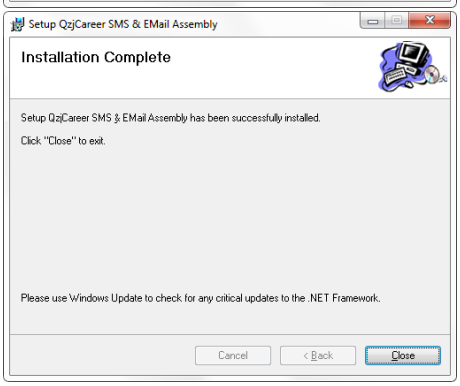


短信、邮件、消息接口的安装

在需要使用 QC 短信邮件及消息处理的服务器上执行 SetupQzjCareerSmsEmail\SetupQzjCareerSmsEmail.msi。







安装成功的操作系统平台将可处理 QC 中的短信邮件。

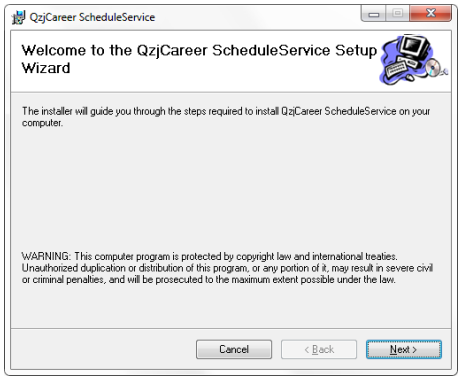
消息队列处理服务安装

QC 消息队列处理服务主要处理事件中的短信和邮件发送，另外监视系统运行状态消息，记录警告和错误状态到数据库中，因此该服务的在线对整个系统至关重要，安装的系统需要能够与数据库快速地通信。

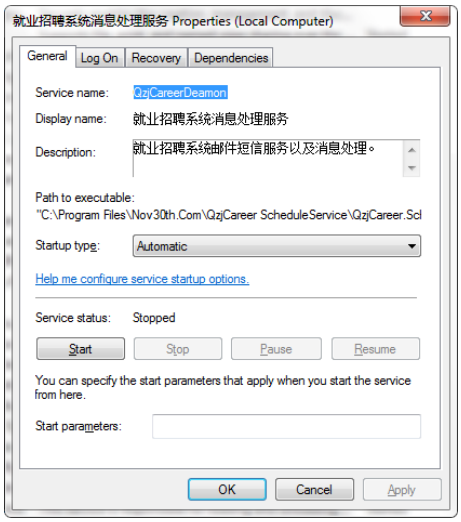
消息队列处理服务可以被安装在任何内部网络直接与数据库通信的操作系统平台上，一个操作系统只建议安装一个实例。

安装步骤：

1. 进入安装介质中的 SetupScheduleService 目录执行 Setup.Exe。



2. 安装完成后将在桌面创建一个名为“Install QzjCareer Message Queue Service”的快捷方式。双击运行。
3. 出现提示“The Commit phase completed successfully.”即安装成功。
4. 点击“开始”菜单的“运行”，输入 Services.msc，找到 QzjCareerDeamon 服务，点击开始。



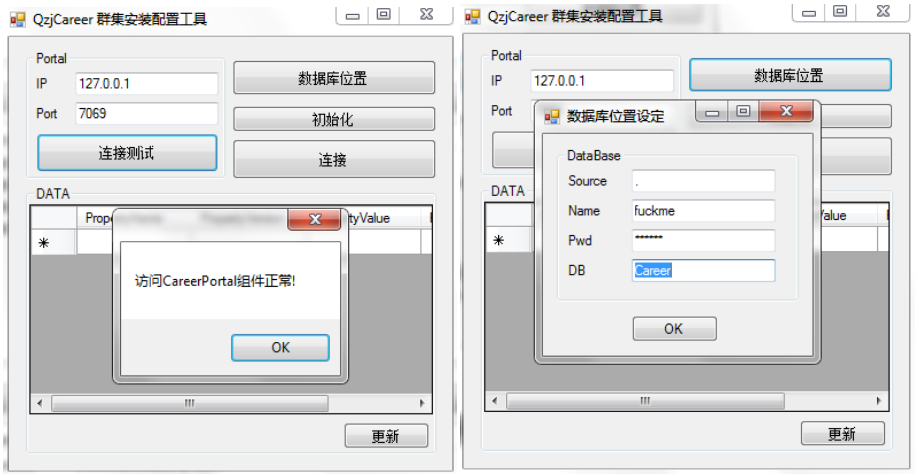
用户交互应用程序安装与配置

用户交互界面的安装与 Portal 相似，安装媒体的 SetupQzjCareerWeb 目录下，运行 QzjCareer.CareerWeb.deploy.cmd 开始安装过程。需要注意的是，安装后要完成配置文件修改、网络配置，包括但不限于：

1. 数据库连接字符串。
2. 用户文件上传根文件夹。
3. MachineKey SecretKey 的设置，分布式状态下必须设置相同。
4. 公用网络至 QC 网络服务端口 NAT 映射。

群集诊断检测工具的使用

在各 QC 内网主机上进行 Portal 的通讯测试，包括连接测试，访问测试，数据测试，功能测试。



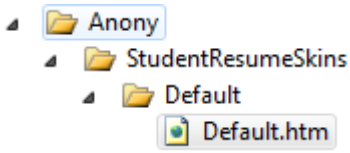
QC 系统扩展插件配置

QC 系统插件基于 Portal 内部接口，运行 QC 插件的系统平台上必须能够访问 Portal 接口。

配置方式参考样例程序“QC 插件 XiaoNei 小工具”。

学生简历模板方案

突出系统个性化、人性化功能以及吸引学生主动参加招聘就业系统上与企业进行交互，系统中许多功能都是定制方便修改的。以最简单的学生简历模板作为介绍：



交互应用程序目录~/Anony/StudentResumeSkins/下用于存放个性化的学生简历模板，维护员或者开发者能够在当前目录下建立以容易辨明模板类型的目录名称来存放模板文件。模板文件内容基于 JTemplate[7]中定义规范。详细开发规范可以从 QC 开发手册中获得。

[7] JTemplate 是基于 JavaScript 扩展库 JQuery 中的一个小插件。

系统诊断及维护

交互界面应用程序心跳

每个交互界面应用程序启动时都将提交其系统基本状态信息给 Portal，并在预先设定的时间间隔持续发送数据。维护人员可以访问 Portal 的 Ping GetServices 中方法来查看当前在线以及最后在线应用程序的时间。

例如本地运行的交互界面应用程序，访问该方法后将返回 XML 数据如下：

<?xml version="1.0" encoding="utf-8" ?>
- <ArrayOfWebServer xmlns:xsi="**http://www.w3.org/2001/XMLSchema-instance**" xmlns:xsd="**http://www.w3.org/2001/XMLSchema**" xmlns="**http://tempuri.org/**">
- <WebServer>
<WebServerAddress>**127.0.0.1**</WebServerAddress>
<WebServerVersion>**0.0001.7069**</WebServerVersion>
<WebServerDomain>**dd6576da-1-128874673611533984**</WebServerDomain>
<LastUpdateAt>**2009-05-22T20:03:23.6309375+08:00**</LastUpdateAt>
<Id>**127.0.0.1**</Id>
<AddtionInformation>**DEFID:WEB1|CLSID:0000-0000-0000-0000-FFFF-0001**</AddtionInformation>
</WebServer>
</ArrayOfWebServer>

开发者可根据该接口开发状态监视及下线报警程序。

内部公开数据接口状态

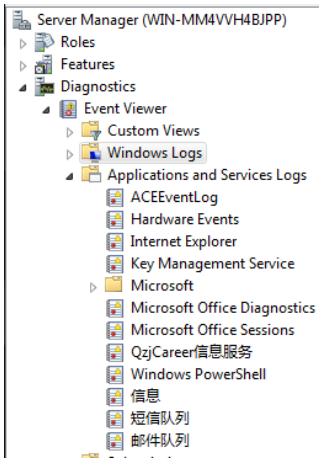
任何对 Portal 接口访问的应用程序都应该在第一次启动或者访问时确认 Portal 能够正常访问，否则停止退出应用程序并间隔尝试。第一次访问时对 Portal 进行 Ping 下的 HelloWorld 方法访问，返回值和期望值相同时，启动后台线程持续进行 AlivePing 方法的访问。

同时，必须对后台线程使用守护进程或者异常处理，在任何导致线程退出的情况及时停止应用程序服务。

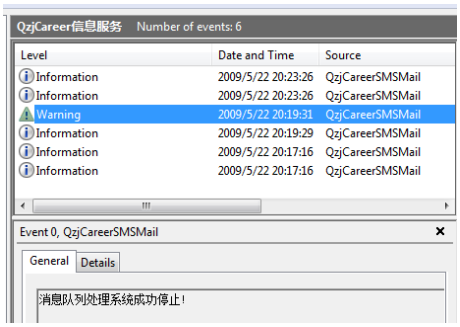
以上行为目的是为了保证在 Portal 接口不可用的情况下，不进行任何的业务操作，保证用户体验，同时确保 Portal 不会因下层应用程序单点的拒绝访问式攻击而瘫痪。

消息队列日志的查看

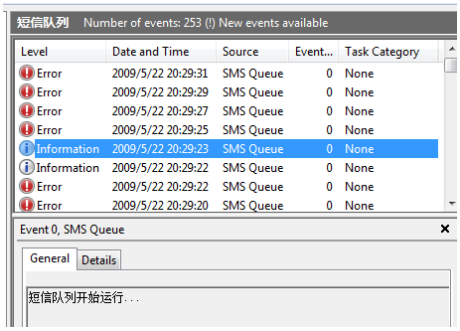
在消息队列处理服务的 Windows 系统平台上，维护人员能够在 Windows 系统管理工具的事件日志中查看服务状态以及服务所有进行的操作。日志分为四种，分别为：QzjCareer 信息服务程序日志，短信日志，邮件日志，系统信息日志。



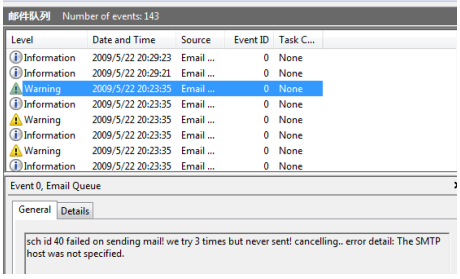
1. 信息服务日志包含服务启动，暂停，停止状态记录和服务内部异常、停止警告等。



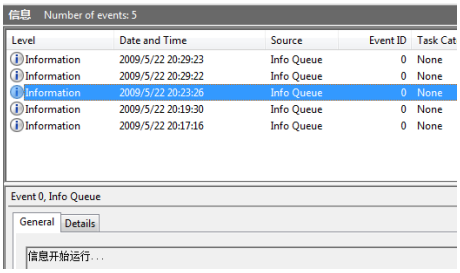
2. 短信日志包含短信发送状态消息、警告、错误。



3. 邮件日志包含邮件发送状态消息、警告、错误。



4. 信息队列日志包含 QzjCareer 系统中各种事件的处理过程和结果信息。



各节点参数独立配置方法

根据需求不同，维护人员可能希望两个交互应用程序运行在不同的环境中。例如，校园内部网络和校园外部网络，并且希望对其从不同网络位置访问的用户进行不同的限制。

使用独立参数配置可以解决类似问题。当交互应用程序[8]启动时，会试图读取其版本的初始化信息（该信息使用群集诊断工具配置，存放于数据库中）。读取的数据将用来覆盖应用程序 web.config 内基本配置信息。当然，除了数据库连接字符串之外，都可以被覆盖。

维护人员可以从群集诊断工具的使用中看到，QzjCareerVersion 节配置信息。该节配置用来告诉数据库取何版本，何种类型应用程序的数据。当前样例中该交互应用程序 QzjCareerVersion 节点值为 0.0001.7069，表示 主版本号 0，附版本号 0001，应用程序参数组编号 7069。

理解参数获得方式，维护人员便可以设置不同的参数组，修改应用程序下的节点配置以对应维护人员期望的目的。

软件环境相关问题

在组建该系统的任何应用时，务必注意使用的.Net Framework 版本。该系统基于.Net Framework 4.0 Beta 1 架构，也是当前编写该文档前 4 天微软对全球公开的第一个测试版本。微软于去年开始开发 Visual Studio 10，CLR 4.0，.Net 4.0 以及 AJAX Framework 4.0。

因此，有可能在微软.Net 运行库产生未知错误与问题。系统内代码已对所有涉及当前版本已知的.Net 平台 BUG 进行优化，但维护人员仍务必做好数据、错误日志的备份，以便及时应对未知状况。

[8] 即 QzjCareer.WEB

插件开发

概述

QzjCareer 系统依照分层，零耦合，分布处理的思想进行编写。因此开发人员可以在原有架构上编写简单的代码来增加新的功能。

例如：添加 Weblog Metabase 的新闻源支持将方便用户使用 Windows Live Writer 发布，修改，删除新闻。添加日历同步源，使用户在 Outlook 创建的内容显示在交互应用程序中，增加其访问就业招聘系统的频率及依赖性。提供个人订阅职位类型中的更新 RSS 订阅，使 Windows Vista 及以上版本用户在使用电脑时能够同时在侧边栏中观察各种新职位。等等。

样例程序

为了让开发人员能够快速，方便地编写 QC 系统上插件和扩展，这里提供一个样例程序。基于 QC 系统 Portal（内网公开 API）和“校内网”[9]两种 API 同步信息的样例。

样例介绍

“校内网”是国内比较庞大的校友录网站，其功能及特点模仿国外 Facebook 交友网，共同拥有操作人性化，信息处理响应延迟低，功能全面，自定义性强等特点。在需求分析阶段的调查显示，无论在校学生、教师、已经参与工作的上班族注册该网站并不定时更新个人资料的群体占调查总人数的 70%。按照预计发展，未来 5 年内如按照其现阶段的维护及功能更新，将会更大比例的学生、教师、企业工作人员使用其网站提供的功能。

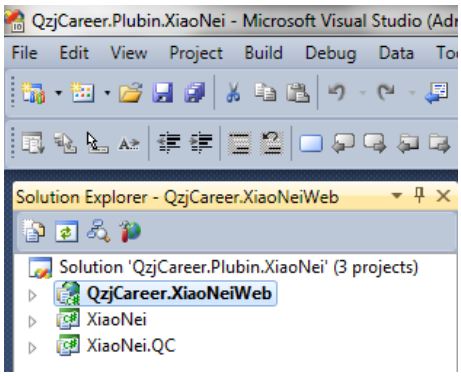
“校内网”提倡使用实名制及真实个人信息，学校资料，工作资料。也提供类似 Facebook 网站的小工具功能，对外开放 API 接口，网站上的用户将能够使用小工具进行交互。

对 QC 系统而言，最关心其网站提供的个人信息，学校信息，以及工作单位信息的接口。QC 可以使用其接口同步学校信息、个人爱好、个人照片、个人介绍，所有就读过的学校、工作单位、实习单位信息。

样例使用 QC 系统中 Portal 内部公开接口，除此之外，不使用其他 QC 系统资源。

[9] 校友录类型的网站。

样例使用



插件使用 3 个工程，分别为“校内网”的 XML 数据结构定义，“校内网”API 封装类，QC 在“校内网”交互界面呈现组成。

说明

开发原则

异常处理

内部 API 函数手册