1. **请描述下你做的项目**

纳税服务系统是国税协同办公平台的一个子系统；主要目的是给用户在移动端上可快速查询国税相关信息并可以在后台对发布的信息进行管理。整个系统包括了移动端和后台端：在移动端主要是查看国税局发布的信息、通知、维护个人信息等；而后台管理功能包括：会员管理、信息发布管理、纳税咨询管理、投诉受理管理、预约服务管理、易告知管理、服务调查管理。我做了其中的xxx。。。等模块。

1. **结合项目谈谈你对MVC的理解**

MVC是Model—View—Controler的简称。即模型—视图—控制器。MVC是一种设计模式，它强制性的把应用程序的输入、处理和输出分开。

    MVC中的模型、视图、控制器它们分别担负着不同的任务。

    视图: 视图是用户看到并与之交互的界面。视图向用户显示相关的数据，并接受用户的输入。视图不进行任何业务逻辑处理。

    模型: 模型表示业务数据和业务处理。相当于JavaBean。一个模型能为多个视图提供数据。这提高了应用程序的重用性

    控制器: 当用户单击Web页面中的提交按钮时,控制器接受请求并调用相应的模型去处理请求。

然后根据处理的结果调用相应的视图来显示处理的结果。

MVC的处理过程：首先控制器接受用户的请求，调用相应的模型来进行业务处理，并返回数据给控制器。控制器调用相应的视图来显示处理的结果。并通过视图呈现给用户。

如在项目中要对应MVC的话：View 对应项目中Jsp，Controler对应Action，Model 对应service+dao层的业务逻辑和持久层的操作。

1. **项目中为什么使用SSH**

1. 使用Struts是因为struts是基于MVC模式的,很好的将应用程序进行了分层，使开发者更关注于业务逻辑的实现；struts有着丰富的标签库，能大大提高开发效率。

2. 使用Hibernate：因为hibernate为Java应用提供了一个易用的、高效率的对象关系映射框架（ORM）。hibernate是个轻量级的持久性框架，功能丰富。

 3. 使用Spring：因为spring基于IoC(Inversion of Control，反向控制)和AOP构架多层j2ee系统的框架。

1. **挑选一个你做的功能说明SSH框架的应用原理**

① 在表示层中，首先通过JSP页面实现交互界面，负责传送请求(Request)和接收响应(Response)，然后Struts根据配置文件(\*-struts.xml)将ActionServlet接收到的Request委派给相应的Action处理。

② 在业务层中，管理服务组件的Spring IoC容器负责向Action提供业务模型(Model)组件和该组件的协作对象数据处理(DAO)组件完成业务逻辑，并提供事务处理、缓冲池等容器组件以提升系统性能和保证数据的完整性。

③ 在持久层中，则依赖于Hibernate的对象化映射和数据库交互，处理DAO组件请求的数据，并返回处理结果。

结合上面的概述；然后可以把我们做的信息发布管理模块用上述的方式实例化描述一遍。

1. **Struts工作原理**

-----------------------------------按照Struts工作原理图地说：

1.客户端发出一个指向servlet容器的请求；

2.请求会经过StrutsPrepareAndExecuteFilter过滤器；

3.过滤器和请求将访问struts2的底层框架结构。在web容器启动时，struts2框架会自动加载配置文件里相关参数，并转换成相应的类。

如：ConfigurationManager、ActionMapper和ObjectFactory。ConfigurationManager 存有配置文件的一些基本信息，ActionMapper存有action的配置信息。在请求过程中所有的对象（Action，Results，Interceptors，等）都是通过ObjectFactory来创建的。过滤器会通过询问ActionMapper类来查找请求中需要用到的Action。

4.如果找到需要调用的Action，过滤器会把请求的处理交给ActionProxy。ActionProxy为Action的代理对象。ActionProxy通过ConfigurationManager询问框架的配置文件，找到需要调用的Action类。

5.ActionProxy创建一个ActionInvocation的实例。ActionInvocation在ActionProxy层之下，它表示了Action的执行状态,或者说它控制的Action的执行步骤。它持有Action实例和所有的Interceptor。

6.ActionInvocation实例使用命名模式来调用，1. ActionInvocation初始化时，根据配置，加载Action相关的所有Interceptor。2. 通过ActionInvocation.invoke方法调用Action实现时，执行Interceptor。在调用Action的过程前后，涉及到相关拦截器(intercepetor)的调用。

7. 一旦Action执行完毕，ActionInvocation负责根据struts.xml中的配置找到对应的返回结果。

---------------------------------通俗地说：

🡺发送http请求

🡺Web服务器(tomcat/weblogic...)

🡺执行struts核心过滤器StrutsPrepareAndExecuteFilter

🡺加载struts配置文件中配置信息，找到对应的Action类并实例化

🡺执行各类拦截器和Action中对应方法

🡺配置文件中找到返回结果

🡺转发到具体页面或其它操作

1. **Struts有什么优缺点**

优点:

1. 实现MVC模式，结构清晰；

2. 丰富的struts的标记库，利用好能大大提高开发效率；

3. 全局结果与声明式异常；

4. 可使用OGNL进行参数传递

5. 各个类方便使用的拦截器

缺点:

1. 转到表现层时，需要配置结果页面；页面多了比较繁杂；

2. 对Servlet的依赖性过强

3. struts标签稍微比el表达式繁重

1. **为何使用Spring**

Spring是一个轻量级的控制反转（IoC）、面向切面（AOP）、面向接口、事务管理、包容促进其它框架；使系统中用到的其它框架耦合程度大大降低，拓展性强、简单易用好管理。

1. **spring在项目中如何充当粘合剂**

1、在项目中利用spring的IOC(控制反转或依赖注入),明确地定义组件接口(如UserDAO)，开发者可以独立开发各个组件, 然后根据组件间的依赖关系组装(UserAction依赖于UserService,UserService依赖于UserDAO)运行，很好的把Struts(Action)和hibernate(DAO的实现)结合起来；

2、spring的事务管理把hibernate对数据库的操作进行了事务配置

1. **描述在系统中如何使用了Spring的事务控制**

Spring事务包括编程式事务和声明式事务。在系统中使用了声明式的事务管理是用Spring的AOP来实现的；配置了只读事务和回滚事务（传播行为为REQUIRED）当出现错误后进行回滚操作。在项目中通过aop切入事务到serivce层，这样做能使一次业务逻辑操作如果包括几个数据库操作都控制在一个事务中。

1. **Hibernate工作原理及为什么要用**

原理： 1.读取并解析配置文件 2.读取并解析映射信息，创建SessionFactory 3.打开Session 4.创建事务Transation 5.持久化操作 6.提交事务 7.关闭Session 8.关闭SesstionFactory

为什么要用： 1. 对JDBC访问数据库的代码做了封装，大大简化了数据访问层繁琐的重复性代码。 2. Hibernate是一个基于JDBC的主流持久化框架，是一个优秀的ORM实现。他很大程度的简化DAO层的编码工作 3. hibernate使用Java反射机制，而不是字节码增强程序来实现透明性。 4. hibernate的性能非常好，因为它是个轻量级框架。映射的灵活性很出色。它支持各种关系数据库，从一对一到多对多的各种复杂关系。

1. **Hibernate在系统中使用的优化策略**

Hibernate 对数据的缓存包括两个级：一级缓存，在Session的级别上进行，主要是对象缓存，以其id为键保存对象，在Session的生命期间存在；二级缓存， 在SessionFactory的级别上进行，有对象缓存和查询缓存，查询缓存以查询条件为键保存查询结果，在SessionFactory的生命期间存 在。默认地，Hibernate只启用一级缓存。

关于这个问题答系统使用了Hibernate的二级缓存就好。

1. **列举你在开发中常用的Struts标签**

property、iterator、if、date、form、a、url、textfiled、textarea、select、radio、checkboxlist、hidden、param

1. **Jquery异步获取数据的方式有几种？有何区别？**

1、load 载入远程 HTML 文件代码并插入至 DOM 中

2、get

3、post

4、ajax

5、getJSON

6、getScript 通过 GET 方式请求载入并执行一个 JavaScript 文件

1. **简述UML并说明你使用Rose常于什么情况，画什么图？**

Unified Modeling Language (UML) 统一建模语言；用于在软件开发各个阶段中用图形的方式描述流程和功能需求；IBM Rational Rose是uml建模工具之一比较多的在需求阶段，常用于画用例图、类图、时序图。

1. **请简述导入导出的工具类；并说明你导出过的最大数据量**

POI（合并单元格对象和背景色的应用）

1. **统计图的实现方式**

Fusioncharts 图表采用SVG（可缩放矢量图形）实现，无需Flash；跨终端

1. **易告知信息推送方式；如何实现**？

在后台填写推送信息和推送名单后，系统将自动推送；后台EE开发人员只实现定时的调用由客户端开发人员写的接口推送信息。

1. **简述调查问卷的制作**

不在你填写的开发功能内的；只需要描述下功能。可以制定调查题目和提供单选和多选两种类型的选项；并可对调查结果进行统计。

1. **系统的并发量如何？如何解决大并发量的请求？**

系统并发量不大；但有在应用服务器中间件Weblogic上做负载均衡的配置。

1. **项目中遇到问题如何解决？**

自行解决——网络搜索——团队讨论解决

1. **简述你在项目中的开发流程**

项目课中的设计和开发都是比较贴近实际开发的，做了的话按照当时的开发情况描述。