**一、项目案例介绍分析模板**

**项目名称**：中国招标投标公共服务平台

**项目组名称**：中国招标投标公共服务平台项目组

**项目背景**：该项目是经国家发改委调和组建的B2B平台，统筹管理国家及各个省市地区 有需要招标、投标的项目。

**项目如何分工**：项目组根据人员等级，分给项目经理、开发经理、架构师、设计师、Java开发程序员、UI设计、前端开发程序员。

框架介绍及分析：springmvc+mybatis，

**项目具体介绍**：门户主要负责客户的操作、信息读取存储，系统管理主要负责后台信息的维护，前置机属于数据交换中心，是连接门户和系统管理的接口。

**模块介绍**：整个项目模块分为 门户+系统管理+前置机（接口）

**技术实现方式及有何优势及对比可以举例说明（要非常详细）**：

我在项目中，主要负责用户信息管理和数据的展示。用户信息的管理，即对用户数据的操作，比如用户添加招标信息，关注多少招标信息等。数据的展示即对从数据库中查询出来的数据，在页面进行展示，是一个查询和统计的功能。查询即根据用例，输入相应的条件，利用ajax异步提交的方式，利用.do方法，寻址url，在controller层加入注解，调用service层方法，调用dao层的公共查询方法，控制mybatits中mapper文件中对应id的sql语句。操作数据库，得到需要展示的数据，返回到前端对应的jsp页面，在js中再利用页面拼接的方式（el表达式等），将要展示的数据进行展示。

中间出现的bug类型：空指针异常、sql注入（写一个sql关键字过滤器，将可能会引起sql变化的关键字比如select and where 等进行白名单过滤。如果前端页面输入这些关键字，要将其替换别的，防止对我们写的sql造成影响）

所做项目里面需要注意的细节问题：一个成熟的程序员和刚入门的最大的区别就是经验，这个经验就是代码的健壮性和可维护性。

首先要熟记开发工具的快捷键，记住常用的十几二十几个快捷键，尽量脱离鼠标的开发，是一个开发老鸟的必备技能。

一个健壮的程序，能在各种异常情况下，给程序的使用者以友好的提示，比如简单的空指针异常，或者服务器宕机，更或者出现黑客的试图攻击。一个功能的开发完成，做到流程的顺利走通，是开发的第一步。简单的说，工作几年的程序员要比新手想的更多的 不仅仅是快捷的功能实现，更多的是，正常执行的程序在何时会出现不可控的异常。程序能够启动，功能能够实现，流程能够有一次走通，说明这个功能我们做出来了，功能实现是没有问题的。但是为何会经常出现问题，bug出现在哪里，有可能是数据，有可能是环境，甚至有可能是人品？于程序开发而言，数据库读取、客户端输入、配置文件读取、文件上传下载得到的内容 等 都属于不可信任来源，因为他们都可能出现人为造成数据异常。对于不可信来源的数据调用，我们是否做了判断？出现异常不做异常处理，程序始终会有bug，程序员的任务就不算完成，因为后期测试出来，都需要我们去改。

说到不可信任来源 这里，我就想到一个需要注意的问题，我们在听课时，老师讲过最多的就是登陆注册，我们是否认为登陆注册是最熟悉的，感觉最好写的，面试官询问时、这个最好回答？错！登陆注册是整个系统最复杂，最考验程序员对程序安全架构理解的部分。有经验的面试官，稍微问一些密码加密传输，数据库存储，页面跳转方面的问题，我们就可能挂菜。因为很多项目，对于这部分都会让专业的设计团队，专门去做。甚至很多大公司，对于这块都是属于绝密的部分，外围人员根本涉及不到的部分，不是真的开发过大型登陆注册的，千万不要在项目介绍中写这些。

再回到刚才说的出现异常。一段程序 分为三个使用者，1程序本身，这个说的是函数可被调用，2维护人员或开发人员3客户。

对于1 简单说，函数a，需要调用函数b,如果b函数本身在参数传值过程中，不对值进行校验，有可能出现空指针异常，而b没有做异常处理，a在调用时，a的后续操作就可能直接断掉。这也是程序最可能出现异常的地方，也是最容易忽略的部分。

对于2，就是基于现在的操作系统的大环境，多数公司都会将程序部署到Linux服务器，将程序部署后，是不能在看到程序的源代码的，也就是说，开发人员不能再进入debug模式进行调试，这时候我们需要准确的定位的到程序问题的所在。初学者都可能想到利用system.out将有可能出现错误的地方输出个记号。但是system.out是不安全的，公司都会配置log日志，这也就要求我们在面试过程中，被问到我们 部署到服务器之后，是如何调试时，我们要记住，在每个函数的产生部位，打上log日志，在函数的使用部位加上异常处理，一个功能的实现，应该有7成的代码在写log和异常处理。

对于3，就是一种互联网的思维，写程序需要以人为主体，这个程序客户用着不顺手，我们自然是要改的。明明是可能的重要文件，我们不能一个删除按钮就提交完事。必须考虑，给客户交互提示。

**二、项目案例介绍分析模板**

**项目名称：**

气象观测中央站

**项目如何分工：**

项目人数：3人

项目分工：

项目经理：负责整体项目流程监制，负责项目关键性技术开发和项目需求确定。

本人负责：负责站点管理模块和数据补发模块功能开发和后期功能修改的后台功能开发。研究mongodb，并将mongodb应用到本系统上。研究apache activemq源码，开发mq插件，消除超过指定数量的消息数量的消息数据。

另外一人：负责项目后台管理模块和数据查询模块功能开发，以及项目前端页面开发。

**框架介绍及分析：**

项目使用框架：SpringMVC

框架介绍及分析：

springmvc实现了系统的低耦合高聚合，便于代码的重用，而且面向接口的编程可以让api的访问不用知道底层的实现，如jdbctemplate，它并不知道是访问什么类型的数据库，只是配置上的差异，这样以后想从mysql换到oracle只需要换下驱动就好了，dao层代码都不用改。

**项目具体介绍：**

项目运行环境：windows7，

项目使用数据库：mongodb数据库、mysql数据库，

项目浏览器：IE8，

项目介绍：气象观测中央站系统主要用来完成对边海防地面自动气象站上传信息的接收、处理。对接收后的报文进行校验、解析、显示、反馈，对校验通过的数据进行转发，对台站数据接收状态的监控、监测、统计，对缺报、漏报的数据进行统计、补报，对边海防地面自动气象站站点进行注册、管理，对注册结果进行反馈等功能。系统具备信息接收、信息处理、信息转发、信息反馈、信息补报、站点管理、状态监测等功能。

**模块介绍：**

站点管理模块：

站点管理模块主要用于维护站点的基本信息、站点设备信息、及站点数据转发地址的基本信息，以便对接收到的数据进行归类匹配及转发。站点管理模块包括站点信息维护、设备维护、数据转发地址维护等功能。

站点状态监测模块：

站点状态监测模块主要实现数据查询和站点状态监测功能。数据接收处理系统接收到来自台站的数据后，对数据进行解码分类，之后把数据存储在数据库中，同时将数据转发给其他数据接收系统。系统实时对台站的通信状态进行监测，实时的更新台站的通信状态。接收到综合采集传输设备的数据，系统把最新的观测场综合通信供电防雷设备的信息及通信状态、台站的相应信息展示给用户。若发生数据缺报情况，系统能够实现半自动化数据补报操作。站点状态监测模块包括表格视图、GIS视图、数据补报、数据查询、到报情况查询、站点数据接收监控等功能。

后台管理模块：

后台管理模块主要是日志管理功能,用于存储台站的注册请求的操作日志、台站的补发请求的补发日志，以便用户查询相应的操作的结果情况。

**技术实现方式及有何优势及对比可以举例说明（要非常详细）：**

1. 使用mongodb数据库接收站点数据：

由于本系统站点数据是一分钟发送一条数据，而本系统要求需要保存一年的数据，mysql数据库满足不了项目需求。Mongo最大的特点是他支持的查询语言非常强大，其语法有点类似于面向对象的查询语言，几 乎可以实现类似关系数据库单表查询的绝大部分功能，而且还支持对数据建立索引。Mongo主要解决的是海量数据的访问效率问题能够满足项目需求。

**中间出现的bug类型：**

一、Could not toggle autocommit

com.mysql.jdbc.exceptions.jdbc4.MySQLNonTransientConnectionException: No operations allowed after connection closed.

com.mysql.jdbc.exceptions.jdbc4.CommunicationsException: Communications link failure

原因：Mysql服务器默认的“wait\_timeout”是8小时，即connection超过8小时，mysql自动断开connection。

解决方案：

1. 增加wait\_timeout时间；
2. 减少connection pool中connection的lifetime；
3. 测试connection pool中connection的有效性；

二、Java.lang.OutOfMemoryError: Java heap space

分析：启动参数里设置JVM大小 ，eclipse.ini文件设置的jvm大小不是具体某个程序运行的JVM大小。

解决方案：（网上）tomcat\_home/bin下catalina.bat(win)或catalina.sh(linux)执行代码前加上：set JAVA\_OPTS=%JAVA\_OPTS%-Xms128m-Xmx512m

三、JDBC begin failed

com.mysql.jdbc.exceptions.jdbc4.CommunicationsException: The last packet successfully received from the server was 86,383,365 milliseconds ago. The last packet sent successfully to the server was 86,383,365 milliseconds ago. is longer than the server configured value of 'wait\_timeout'. You should consider either expiring and/or testing connection validity before use in your application, increasing the server configured values for client timeouts, or using the Connector/J connection property 'autoReconnect=true' to avoid this problem.

Caused by: java.net.SocketException: Software caused connection abort: socket write error

分析：此问题同问题一相同，都是由timeout时间引起的，

解决方案：

在Mysql安装目录下找到my.ini文件添加超时时间限制：在最后一行添加：wait\_timeout=31536000

**所做项目里面需要注意的细节问题：**

正则表达式验证字段输入框。

结合使用apache activemq消息传输平台时消息不能积累过多不进行接收，若消息积累过多会造成消息阻塞。

接收站点数据时项目解码时要严格按照数据协议进行解析数据。

**三、项目案例介绍分析模板**

1. **项目名称**

新一代资产负债管理系统

1. **项目组名称**

流动性管理二期

1. **项目背景：**

通过资产负债管理系统对利率、资金流动、同业融资、外汇业务等银行业务进行有效管理，实现银行业务的优化，准确、及时、全面计量、监测和报告流动性风险状况，以便更好地实现对银行相关信息的管理，提高流动性风险管理能力，增强流动性管理的系统支撑;避免或减少各种可能的损失；提高现金流量的稳定性和资金的使用效率;加强人民币资金管理，提高银行资金营运效益，实现银行业务的不断发展及进步。

1. **项目如何分工：**

客户负责底层接口、核心功能及相关业务的商榷，外协人员负责页面及后台相关功能的实现

1. **框架介绍及分析**

前台框架为Extjs，后台框架为ssi(spring,struts,ibatis)

1. **项目具体介绍**

系统总体功能可分为数据区、功能区、展现区三个组成部分。数据区通过与源系统的接口获得原始数据，按照设定的规则进行转换处理及校验后，形成按主题与数据类别组织的资产负债管理基础数据集。功能区为用户提供主要的操作功能，并满足核心业务规则的实现需要，功能可主要分为人民币跨境支付管理、流动性成本分摊管理、总行资金头寸表管理、流动性监测指标管理、系统内上存下借管理。

1. **模块介绍**

本系统包括人民币资金一体化管理、流动性监测预警体系、CIPS账户管理功能、上存下借等模块，同时还包括现金流量管理、流动性预报和成本分摊优化、头寸管理模块。

1. **技术实现方式及有何优势及对比可以举例说明（要非常详细）**
2. 通过前台框架Extjs、自定的标签及相关的其他标签等实现前台页面的设计，相对于直接用JQuery等前台技术,方便了很多。如常用的分页功能，只需要在后台稍加设置，前台使用<P:paging/>即可实现分页功能，相对于传统的技术更加方便快捷。
3. 后台通过SSI(Spring,Struts,IBatis)框架实现，相对于SSH效率提高很多，用IBatis替代hibernate,后台SQL操作通过手工书写，以便有效的控制效率。
4. 报表的制作使用润乾报表插件，方便快捷。
5. 文件的上传下载使用POI技术等。
6. 例如系统内上存申请：该功能主要包含增加、修改、删除、重新发送、启动流程、查看流程图等功能。页面通过Extjs实现，布局为上下两部部分，上部分为查询条件及查询按钮的相关内容，其中查询是通过AJAX进后台查询，将结果返回至前台页面。下部分为其他功能的按钮、效果内容显示及相关功能的操作并通过Extjs的分页技术实现结果分页显示。以增加为例：点击增加按钮，弹出增加内容的弹窗口（p:window），点击保存则携带相关的信息通过Ajax的URL，进入struts寻找对应的Action及对应的方法，通过action调用service的方法，通过service再调AL中的方法，通过AL再调用dao中的方法，通过dao层调用对应的配置文件，执行对应SQL，将数据插入数据库。另外其他功能流程类似。其中重新发送功能：当数据插入成功且符合条件将会自动把对应的数据发送至COS，发送失败后可重新发送。启动流程：启动工作流，实现层层审批。查询流程图：查看审批流程图。
7. **中间出现的bug类型**

项目中出现的bug的类型有：提示信息类、建议类、需求变动

1. **所做项目里面需要注意的细节问题**
2. 权限问题：菜单权限，按钮权限，数据操作权限等，通过判断不同的登录用户实现菜单及按钮是否显示，并根据不同的用户对数据进行相应的筛选
3. 实现功能的一些必须的配置，如工作流，文件上传下载等，需要在application ,struts,bean等相关文件进行配置。
4. 前台页面显示：页面美观，风格统一，提示准确具体