**用户需求书**

# 项目背景及目标

## 项目背景

随着港口业务的持续增长和数字化转型的推进，新轮渡生产系统和新散杂货TOS系统建设项目近期将上线运行。商服平台作为港航对外服务的重要窗口，目前尚未与这两个新的生产系统进行业务数据对接，如海峡专道预约、开票申请、船代业务、货代业务等，为了确保新生产系统上线后，商服平台业务的连续性和稳定性，须对商服平台与新生产系统之间的接口进行重新开发和对接。商服平台将能够更好地支持港航的数字化转型，为港航的长期发展提供有力的技术支持。

## 项目建设的必要性

为确保港口业务在新轮渡生产系统和新散杂货TOS系统上线后的平稳过渡与高效运行，本项目建设显得尤为关键，具体表现在如下三个方面。

一是保障业务连续性，新轮渡生产系统和新散杂货TOS系统的上线，将带来数据接口的变化。为确保商服平台业务在系统切换期间不受影响，商服平台须与新系统进行对接，保障业务的连续性。

二是适应业务发展需求，随着港口业务的不断拓展和数字化转型的推进，对商服平台的功能和性能提出了更高要求。通过项目建设为港航的长期发展奠定基础。

三是提升数据管理效率，通过将基础代码查询接口迁移到ESB平台，实现数据的集中管理和统一查询，提高数据管理效率。

## 建设目标

本项目建设旨在通过与新轮渡生产系统和新散杂货TOS系统的数据对接，实现以下具体目标：

完成海峡专道预约的数据对接：开发接口获取车辆到港时间和始发港口信息，确保用户在办理海峡专道预约时可选择对应的预约信息。同时，实现预约状态的查询，用户可通过车牌号或状态在台账中查询车辆预约状态数据。开发数据查询接口，使用户能够通过车牌号、状态、预约到港日期查询新轮渡生产系统中的具体预约记录。

实现开票申请功能的对接：完成与新轮渡系统开票申请数据的对接，用户在进行申请开票时可通过接口向新轮渡系统发送申请。同时，定时通过接口查询新轮渡生产系统中的开票结果，更新订单开票状态，并在商服平台列表中展示已发起开票的记录。

实现货车代扣业务的对接，用户在C端发起支付，商服校验代扣条件并返回结果，C端通知新轮渡系统改订单为支付状态；发起退款时，商服校验代扣数据并还原金额至企业钱包，C端通知系统改订单为未支付；商服通过接口查询用户代扣订单列表及明细，用户勾选订单后通过接口申请开票，支持接口申请合并开票。

实现散杂货子系统船代业务的对接：在船舶申报的未提交、已提交、已受理、已拒绝阶段，通过接口查询当前港区的订单数据，并在商服平台中根据港区筛选查询。船舶档案申报的待审核、审核通过、审核驳回阶段，通过接口获取船舶档案申报审核状态数据，并在商服平台中展示。车辆备案功能通过接口查询TOS系统中的车辆备案数据，并在商服平台中展示。

实现散杂货子系统货代业务的对接：在磅单查询、车辆预约、存栈委托、进口委托单申报、出口委托单申报等功能中，通过接口查询新通用TOS系统中的数据，并在商服平台中根据港区筛选查询。

实现基础代码查询接口的更换：将现有的基础服务接口（如港区代码、国籍代码等）迁移到ESB平台，通过ESB提供统一的接口服务，实现数据的集中管理和统一查询，提高数据管理效率。

# 建设内容

## 新轮渡生产系统对接

### 海峡专道预约

#### 预约信息

开发接口实现与新轮渡生产系统对接获取车辆到港时间、始发港口信息，以便用户在办理海峡专道预约时可选择对应的预约信息，确保预约信息完整准确。

#### 预约状态

开发接口实现获取推送至新轮渡生产系统的车辆预约状态信息，以便用户在台账中通过车牌号或状态查询车辆预约状态数据。

#### 数据查询

开发数据查询接口，实现通过车牌号、状态、预约到港日期查询新轮渡生产系统中具体预约记录。

### 开票申请

#### 申请开票

完成与新轮渡系统开票申请数据的对接，用户在进行申请开票时通过接口向新轮渡系统发送申请，同时提供合并开票方式，方便用户对多个订单进行统一开票。

#### 开票记录

完成定时通过接口查询新轮渡生产系统中开票结果，商服平台更新订单开票状态，列表展示已发起开票的记录。

### 货车代扣

#### 车辆代扣

完成与C端轮渡管家的对接。用户过海时在c端轮渡管家发起支付，通过接口向商服发送校验，商服校验其满足代扣条件并将代扣结果返回，C端根据结果通知新轮渡系统修改订单为支付状态。

#### 车辆退款

完成与C端轮渡管家的对接。用户过海时在c端轮渡管家发起退款，通过接口向商服发送校验，商服校验其是代扣数据并将代扣金额还原回企业钱包，C端根据结果通知新轮渡系统修改订单为未支付状态。

#### 订单查询

完成与新轮渡系统在查询货车代扣订单的对接。商服通过接口向新轮渡系统查询属于该用户的代扣订单列表与明细。

#### 代扣开票

完成与新轮渡系统在申请货车代扣发票的对接。用户勾选货车代扣订单通过接口向新轮渡系统发送申请。

#### 合并开票

通过接口向新轮渡系统申请合并开票，方便用户对多个订单进行统一开票。

## 新散杂货TOS系统对接

### 船代业务（PC端）

#### 船舶申报-未提交

船舶申报时，需选择作业区，在船舶申报未提交列表中根据港区筛选查询（必填项），商服平台通过接口查询当前港区的未提交订单。

#### 船舶申报-已提交

在船舶申报已提交列表中根据港区筛选查询（必填项），商服平台通过接口查询当前港区的已提交订单。

#### 船舶申报-已受理

通过接口获取各港区已受理申报信息，在船舶申报已受理列表中根据港区查询（必填项），可筛选对应港区的已受理订单。

#### 船舶申报-已拒绝

通过接口获取各港区已拒绝申报信息，在船舶申报已拒绝列表中根据港区查询（必填项），可筛选对应港区的已拒绝订单。

#### 船舶档案申报-待审核

商服平台通过接口获取船舶档案申报审核状态数据，船舶档案申报的待审核阶段，不区分港区将待审核申报数据进行展示。

#### 船舶档案申报-审核通过

商服平台通过接口获取船舶档案申报审核状态数据，船舶档案申报的审核通过列表，不区分港区将审核通过申报数据进行展示。

#### 船舶档案申报-审核驳回

商服平台通过接口获取船舶档案申报审核状态数据，船舶档案申报的审核驳回列表，不区分港区将审核驳回的申报数据进行展示。

### 货代业务（PC端）

#### 磅单查询

在磅单查询功能中，港区查询设置为必填项，商服平台通过接口查询当前港区所有提单号及其磅单记录。

#### 车辆预约-我要预约

在车辆预约的“我要预约”功能中，港区查询设置为必填项，商服平台通过接口查询新通用TOS系统当前港区提单号数据。

#### 车辆预约-我要预约(拆单)

在车辆预约的“我要预约（拆单）”功能中，港区查询设置为必填项，商服平台通过接口查询新通用TOS系统当前港区提单号数据。

#### 车辆预约-预约记录

在查询车辆预约记录时，港区查询设置为必填项，商服平台通过接口查询新通用TOS系统当前港区预约记录数据。

#### 磅单查询（拆单）

在磅单查询（拆单）功能中，港区查询设置为必填项，商服平台通过接口查询当前港区所有提单号及其磅单记录。

#### 存栈委托-未提交

存栈委托未提交列表中，港区查询设置为必填项，商服平台通过接口可筛选对应港区的未提交存栈委托申请。

#### 存栈委托-已提交

存栈委托已提交列表中，港区查询设置为必填项，商服平台通过接口查询当前港区已提交存栈委托申请。

#### 存栈委托-已受理

存栈委托已受理列表中，港区查询设置为必填项，商服平台通过接口查询当前港区已受理存栈委托申请。

#### 存栈委托-已拒绝

存栈委托已拒绝列表中，港区查询设置为必填项，商服平台通过接口查询当前港区已拒绝存栈委托申请。

#### 进口委托单申报-未提交

进口委托单申报未提交列表中，港区查询设置为必填项，商服平台通过接口可筛选对应港区的未提交进口委托单申报。

#### 进口委托单申报-已提交

进口委托单申报已提交列表中，港区查询设置为必填项，商服平台通过接口查询当前港区已提交进口委托单申报。

#### 进口委托单申报-已受理

进口委托单申报已受理列表中，港区查询设置为必填项，商服平台通过接口查询当前港区已受理进口委托单申报。

#### 进口委托单申报-已拒绝

进口委托单申报已拒绝列表中，港区查询设置为必填项，商服平台通过接口查询当前港区已拒绝进口委托单申报。

#### 出口委托单申报-未提交

出口委托单申报未提交列表中，港区查询设置为必填项，商服平台通过接口可筛选对应港区的未提交出口委托单申报。

#### 出口委托单申报-已提交

出口委托单申报已提交列表中，港区查询设置为必填项，可筛选对应港区的已提交出口委托单申报。

#### 出口委托单申报-已受理

出口委托单申报已受理列表中，港区查询设置为必填项，可筛选对应港区的已受理出口委托单申报。

#### 出口委托单申报-已拒绝

出口委托单申报已拒绝列表中，港区查询设置为必填项，可筛选对应港区的已拒绝出口委托单申报。

#### 车辆备案

商服平台通过接口查询TOS系统中车辆备案数据，不区分港区将车辆备案数据进行展示。

### 数据查询（PC端）

#### 库存查询

在库存查询功能中，港区查询设置为必填项，商服平台通过接口查询当前港区库存数据。

#### 船舶作业动态查询

在船舶作业动态查询功能中，港区查询设置为必填项，商服平台通过接口查询当前港区船舶作业动态数据。

#### 理货记录查询

在理货记录查询功能中，港区查询设置为必填项，商服平台通过接口查询当前港区理货记录数据。

#### 历史作业单量查询

在历史作业单量查询功能中，港区查询设置为必填项，商服平台通过接口查询当前港区历史作业单量数据。

### 船代业务（移动端）

#### 船舶申报-未提交

在移动端船舶申报时，需选择作业区，在船舶申报未提交列表中增加港区查询功能（必填项），可筛选对应港区的未提交订单。

#### 船舶申报-已提交

在移动端船舶申报已提交列表中增加港区查询功能（必填项），可筛选对应港区的已提交订单，可进行查看、取消提交操作。

#### 船舶申报-已受理

通过接口获取各港区已受理申报信息，在移动端船舶申报已受理列表中增加港区查询功能（必填项），可筛选对应港区的已受理订单。

#### 船舶申报-已拒绝

通过接口获取各港区已拒绝申报信息，在移动端船舶申报已拒绝列表中增加港区查询功能（必填项），可筛选对应港区的已拒绝订单。

#### 船舶档案申报-待审核

商服平台通过接口获取船舶档案申报审核状态数据，在移动端船舶档案申报的待审核列表，不区分港区将待审核申报数据进行展示。

#### 船舶档案申报-审核通过

商服平台通过接口获取船舶档案申报审核状态数据，在移动端船舶档案申报的审核通过列表，不区分港区将审核通过申报数据进行展示。

#### 船舶档案申报-审核驳回

商服平台通过接口获取船舶档案申报审核状态数据，在移动端船舶档案申报的审核驳回列表，不区分港区将审核驳回的申报数据进行展示。

### 货代业务（移动端）

#### 进口委托单申报-未提交

在移动端进口委托单申报未提交列表中，港区查询设置为必填项，商服平台通过接口可筛选对应港区的未提交进口委托单申报。

#### 进口委托单申报-已提交

在移动端进口委托单申报已提交列表中，港区查询设置为必填项，商服平台通过接口查询当前港区已提交进口委托单申报。

#### 进口委托单申报-已受理

在移动端进口委托单申报已受理列表中，港区查询设置为必填项，商服平台通过接口查询当前港区已受理进口委托单申报。

#### 进口委托单申报-已拒绝

在移动端出口委托单申报已拒绝列表中，港区查询设置为必填项，可筛选对应港区的已拒绝出口委托单申报。

#### 车辆预约

在移动端车辆预约功能中，港区查询设置为必填项，商服平台通过接口查询新通用TOS系统当前港区提单号数据或预约记录。

#### 磅单查询

在移动端磅单查询功能中，港区查询设置为必填项，商服平台通过接口查询当前港区所有提单号及其磅单记录。

#### 出口委托单申报-未提交

在移动端出口委托单申报未提交列表中，港区查询设置为必填项，商服平台通过接口可筛选对应港区的未提交出口委托单申报。

#### 出口委托单申报-已提交

在移动端出口委托单申报已提交列表中，港区查询设置为必填项，可筛选对应港区的已提交出口委托单申报。

#### 出口委托单申报-已受理

在移动端出口委托单申报已受理列表中，港区查询设置为必填项，可筛选对应港区的已受理出口委托单申报。

#### 出口委托单申报-已拒绝

在移动端出口委托单申报已拒绝列表中，港区查询设置为必填项，可筛选对应港区的已拒绝出口委托单申报。

#### 存栈委托-未提交

在移动端存栈委托未提交列表中，港区查询设置为必填项，商服平台通过接口可筛选对应港区的未提交存栈委托申请。

#### 存栈委托-已提交

在移动端存栈委托已提交列表中，港区查询设置为必填项，商服平台通过接口查询当前港区已提交存栈委托申请。

#### 存栈委托-已受理

在移动端存栈委托已受理列表中，港区查询设置为必填项，商服平台通过接口查询当前港区已受理存栈委托申请。

#### 存栈委托-已拒绝

在移动端出口委托单申报已拒绝列表中，港区查询设置为必填项，商服平台通过接口查询当前港区已拒绝存栈委托申请。

#### 过户申请-我的单证

在移动端过户申请我的单证列表中，港区查询设置为必填项，可筛选对应港区当前登录人的单证数据。

#### 过户申请-已提交

在移动端过户申请已提交列表中，港区查询设置为必填项，商服平台通过接口查询当前港区已提交过户申请。

#### 过户申请-已受理

在移动端过户申请已受理列表中，港区查询设置为必填项，商服平台通过接口查询当前港区已受理过户申请。

#### 过户申请-已拒绝

在移动端过户申请已拒绝列表中，港区查询设置为必填项，商服平台通过接口查询当前港区已拒绝过户申请。

#### 停泊费

在移动端停泊费账单查询中，港区查询设置为必填项，商服平台通过接口查询当前港区停泊费账单数据。

#### 停泊费-我的缴费记录

在移动端停泊费我的缴费记录中，港区查询设置为必填项，商服平台通过接口查询当前港区当前登录人的缴费记录数据。

#### 单证拆分-我的单证

在移动端单证拆分我的单证中，港区查询设置为必填项，商服平台通过接口查询当前港区当前登录人的拆单数据。

#### 单证拆分-已提交

在移动端单证拆分已提交列表中，港区查询设置为必填项，商服平台通过接口查询当前港区已提交的拆单数据。

#### 单证拆分-已审核

在移动端单证拆分已审核列表中，港区查询设置为必填项，商服平台通过接口查询当前港区已审核的拆单数据。

#### 单证拆分-已拒绝

在移动端单证拆分已拒绝列表中，港区查询设置为必填项，商服平台通过接口查询当前港区已拒绝的拆单数据。

#### 车辆预约（拆单）

在移动端车辆预约（拆单）功能中，商服平台通过接口查询新通用TOS系统当前港区拆单的车辆预约数据。

#### 磅单查询（拆单）

在移动端磅单查询（拆单）功能中，港区查询设置为必填项，商服平台通过接口查询当前港区所有拆单提单号数据。

#### 磅单查询（拆单）-磅单记录

在移动端磅单查询（拆单）磅单记录功能中，港区查询设置为必填项，商服平台通过接口查询当前港区磅单记录。

#### 散货预约记录

在移动端散货预约记录功能中，港区查询设置为必填项，商服平台通过接口查询当前港区散货预约记录。

#### 车辆备案-全部

在车辆备案全部列表中，商服平台通过接口查询新通用TOS系统中车辆备案的全部数据（不区分港区）。

#### 车辆备案-待审核

在车辆备案待审核列表中，商服平台通过接口查询新通用TOS系统中车辆备案的待审核数据（不区分港区）。

#### 车辆备案-审核通过

在车辆备案审核通过列表中，商服平台通过接口查询新通用TOS系统中车辆备案的审核通过数据（不区分港区）。

#### 车辆备案-审核驳回

在车辆备案审核驳回列表中，商服平台通过接口查询新通用TOS系统中车辆备案的审核驳回数据（不区分港区）。

### 数据库查询（移动端）

#### 库存查询

在移动端库存查询功能中，港区查询设置为必填项，商服平台通过接口查询当前港区库存数据进行展示。

#### 船舶作业动态查询

在移动端船舶作业动态查询功能中，港区查询设置为必填项，商服平台通过接口查询当前港区船舶作业动态数据进行展示。

#### 理货记录查询

在移动端理货记录查询功能中，港区查询设置为必填项，商服平台通过接口查询当前港区理货记录数据进行展示。

#### 历史作业单量查询

在移动端历史作业单量查询功能中，港区查询设置为必填项，商服平台通过接口查询当前港区历史作业单量数据进行展示。

#### 货物称重汇总查询

开发货物称重汇总查询接口，在移动端货物称重汇总查询中港区查询设置为必填项，商服平台通过接口查询当前港区货物称重汇总数据进行展示。

### 配置管理

商服平台提供一个切换新旧TOS系统配置功能，能够有效地支持新旧TOS系统的数据接口切换操作。

### 基础代码查询接口更换ESB

现有的基础服务接口将替换为通过ESB提供统一的接口服务，具体如下：

1.港区代码

2.国籍代码

3.船舶代理代码

4.委托人代码

5.联系人代码

6.申请人代码

7.货代代码

8.货主代码

9.收货人代码

10.货名代码

11.贸易性质代码

12.进出口性质代码

13.包装代码

14.作业方式代码

15.集港方式代码

16.船舶档案

17.港口代码

18.操作过程代码

19.单证类型代码

20.车队代码

21.轴数代码

22.库场代码

23.船型代码

24.船公司代码

25.规格代码

# 技术方案

## 设计原则

### 业务驱动原则

架构以业务发展、需求为导向，能够满足、支持业务的需要。以业务驱动为原动力，同时能够根据业务的发展而不断提出新的需求，进行动态、灵活地调整。

### 基于组件/服务为基础

采用面向服务的架构（SOA）进行设计。通过提供可重用的业务单元级组件，以及规则驱动的模型，解耦各个业务功能，目的是要达到架构的灵活性。避免由于应用的改变所产生的重复劳动成本，通过模块重用、代码重构，加快响应需求的时间。

### 标准统一原则

整体架构遵从标准统一的原则， 应用系统遵循架构统一原则，遵守统一的架构规范和第三方对接标准，避免造成系统资源过于庞杂，难以管理，难以共享。遵循标准统一原则和每个资源种类最少化原则。

### IT资产重用原则

充分利用现有的硬件、软件、服务组件等IT资产。为了满足、适应业务需求的快速变化和在尽可能短的时间内完成IT系统对业务的有力支持，使现有的大部分IT资产可以充分地利用。即对现有的进行适当重组合、再分配和更新配置等方式实现，加强对可重用资产的维护和管理流程，支持对这些可重用的组件的管理、定位。

### 多层次应用体系原则

每个具体的应用被分成多个层次，至少包括：

* 表示层：提供用户界面表现和获取其输入
* 应用层：包含大多数业务逻辑
* 数据管理层：管理业务数据及完整约束
* 应用系统架构采用JAVA架构进行设计，使系统具有很好的框架和灵活性。
* 表示层与应用层和数据管理层分隔，在增加一个新的访问渠道时，仅增加渠道驱动，改变内容展示格式，而交易处理和与后台的数据通讯及处理不作任何修改。
* 应用层与数据管理层的分隔，可以在业务处理不作大量的改动的情况下，连接到不同的后台。

因此，分层设计可以使得系统更加灵活、易于维护。

## 建设模式

本项目在设计开发时将采用以下的技术路线：

1. 保持商服平台现有技术路线延续性。

2. 严格的面向对象的分析和设计方法；系统采用开放性框架体系；全面采用XML技术与基于XML的Webservice技术作为文件、接口传输的规范和瘦客户端应用，支持JSON数据传输格式。

3. 系统实现跨时区操作，实现当地时区和标准时间的转换和显示。

4. 本系统设计中充分考虑系统每个模块的可扩充接口，保证系统能随时加挂各种应用模块；能够实现跨平台和跨数据库；采用多服务器集群和动态负载平衡技术，随时随地可通过增加服务器达到线性地提高系统整体性能的目的。

## 技术架构

### 微服务架构

统一采用B/S架构，便于维护更新。在技术路线方面采用微服务架构设计，微服务设计方案是一种将软件系统划分为多个独立且自治的小型服务的方法。每个微服务都专注于完成一个特定的业务功能，并通过轻量级的通信机制进行交互。

设计方案步骤：

1、需求分析：对系统的需求进行详细的分析和定义。确定系统中的各个业务功能，并将其拆分成适合微服务的粒度。

2、服务边界划分：根据需求分析的结果，将系统的各个业务功能划分为不同的服务。每个服务应该有清晰的边界，只关注自身的业务逻辑，而不干涉其他服务。

3、服务通信机制：确定不同服务之间的通信方式。常见的通信机制包括RESTful API、消息队列、事件驱动等。

4、数据管理：在微服务架构中，每个服务可以有自己的数据存储。确定每个服务所需的数据模型和数据库，并考虑数据的一致性和可靠性。

5、安全和认证：确保每个服务都有适当的安全措施，例如访问控制和身份验证。使用标准的认证和授权机制来保护对敏感数据和功能的访问。

6、容错与监控：设计容错机制，使系统能够处理服务不可用或故障的情况。同时，建立监控系统来跟踪服务的性能和健康状态，并及时发现和解决问题。

7、部署与扩展：选择适当的部署策略，例如容器化技术和自动化部署工具。确保系统能够根据需求进行水平扩展，以应对不断增长的流量和负载。

8、持续集成与交付：使用持续集成和交付工具，确保每个微服务都可以独立地进行开发、测试和部署。这有助于减少发布时间和风险。

9、前后端分离：将前端和后端分离开来。前端可以通过API调用不同的微服务来获取数据和执行业务逻辑。

### 引迈基础开发框架

商服平台管理后台延续使用引迈基础开发框架实现，集成到运营管控平台。jnpf-java-cloud 采用 J2EE 技术体系，基于 Spring Cloud Alibaba 微服务框架进行封装，包含多种开箱即 用功能、通用技术组件与服务、微服务治理等，具备RBAC功能、网关统一鉴权、Xss防跨站攻击、自动代码生成、 多种存储系统、分布式事务、分布式定时任务等多个模块，平台设计灵活可扩展、可移植、可应对高并发需求。同时兼顾本地化、私有云、公有云部署，支持SaaS模式应用。核心技术采用 Spring Boot , Spring Cloud Alibaba , Mybatis-plus 等相关核心组件，采用 Nacos 注册和配置中心，集成流量卫兵 Sentinel ，前端基 于 vue-element-admin 框架定制开发。为企业服务化中台整合、数字化中台转型提供强力支撑。

### 基于松耦合设计

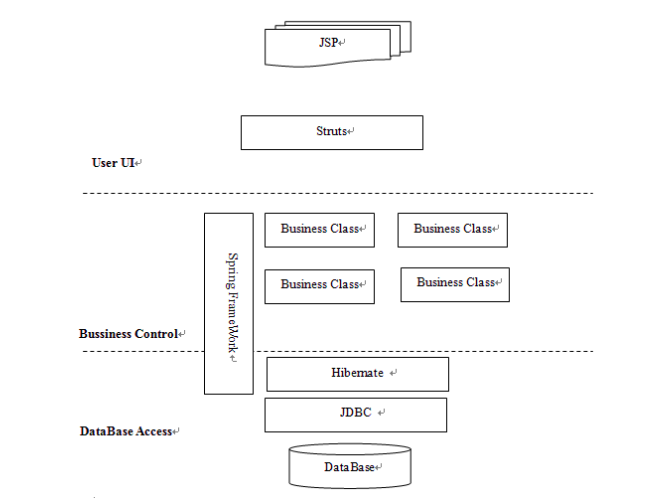
本项目使用松耦合设计符合平台和业务系统的开发需求。松耦合系统是基于消息的系统，松耦合通讯机制提供了紧耦合机制所没有的许多优点，有助于降低系统和模块之间的依赖性。理想情况下，各个业务条线关系应当互不影响：在建立新型业务关系时，不对已有的业务关系造成影响。为一个业务系统提供的功能或许不应当供给另一个系统；与一个业务相关的更改不应对其他合作伙伴造成影响。一个业务不应为了等待一个同步响应，而阻塞另一个系统应用或模块。松耦合系统的优点在于更新一个模块不会引起其它模块的改变。基于这种思路，本次项目建设在系统设计时采用松耦合的系统设计思路，保障系统负荷实际使用要求。

### 基于数据总线设计

本项目基于数据总线设计，为各业务系统提供统一的数据服务出口。数据总线规范了应用系统中同构系统、异构系统等方面进行数据共享和交换实现方法。系统间数据交换在架构分层上都有业务实体层，数据交换机制在业务实体层建立了一层对所有应用系统透明的层。子系统无论何种技术方案，均可通过业务实体层进行共享和交互，从而建立子系统间的持续集成和业务扩展，实现一个可扩展的完整的一体化信息系统。

### 高扩展性的后台设计

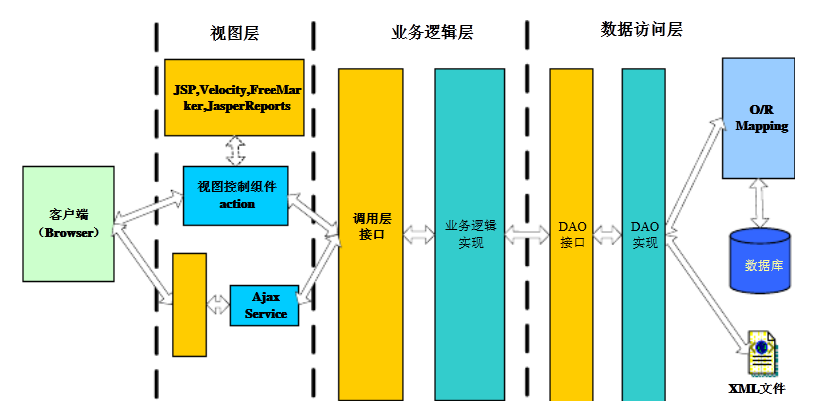
以当前JAVA WEB开发最流行的框架spring+hibernate+struts为基础，并通过大量项目实施的验证，形成的一套高效、扩展性强、灵活、稳定的技术框架。



在整个系统架构中，我们从逻辑上将系统分为多层来进行控制，分为界面上的流程控制和业务上的流程控制。

在业务组件的流程控制上，本架构利用了Spring容器来保证组件与组件之间的松耦合性，利用Spring的依赖注入机制来保证组件之间的完全解耦。由于Spring对原有的代码的无侵入性，能够最大程度上保护原有的开发工作，同时Spring能有效地降低系统的开发量，提高业务类的可重用性，更加有效的保证系统能按时按质的完成开发工作。

在持久层的应用，本架构利用Hibernate来实现对持久层的管理以及作为ORM（对象、关系映射）工具。在今日的企业环境中，把面向对象的软件和关系数据库一起使用可能是相当麻烦、浪费时间的。Hibernate是一个面向Java环境的对象/关系数据库映射工具。对象/关系数据库映射(object/relational mapping (ORM))这个术语表示一种技术，用来把对象模型表示的对象映射到基于SQL的关系模型数据结构中去。



此架构的优点：

**提高了系统性能：**

通过一些先进技术的引用，能够使基于web的b/s项目出现的一些性能瓶颈得到很好的解决，考虑到项目的用户量和数据量，采用此架构完全能够较好地满足未来系统的需求。

**技术成熟，层次清晰：**

从架构示意图和此架构采用的技术分析可以看出，控制层、视图层、业务层、持久层之间的层次关系非常清楚。所采用的技术能够很好的实现各层次的松散耦合。对系统的维护性和扩展性提供了很好的基础，做到了一定的界面灵活性，

页面展示可以通过装载页面模版实现，包含了实现页面动态展示的各种技术。能大幅度提高系统的异构性。在开发层面上彻底分离页面组件和前台页面的开发。

### 基于构件技术的系统搭建模式

面向构件技术整合了传统的构件技术、模型驱动技术和面向服务技术。构件的意义超越了技术层面的代码集合，而是真正做到与业务层面的内容相映射。构件将成为表达业务需求的最小单元，在特定行业中，通过积累可以形成相对完备的客户业务需求构件库，以构件组装的形式快速搭建客户的应用；以构件修改和构件增减的方式快速满足客户业务需求的变化。其次，从构件的技术实现角度来说，可以用四个要素来陈述软件构件模型：构件本身、该构件的插口、构件与其他构件协作的能力、以及构件的使用者。

面向构件的开发是一种软件开发手段，在开发周期的不同阶段和不同方面，包括需求分析、结构、设计、建立、测试、上线、支撑性技术架构、项目管理等，都以构件为基础，同时为了使构件可随时用于组装，这些构件必须作为项目的零件来建造。

本项目将采用面向构件的系统分析与设计方法，以减少代码量和复杂性，增加构件的可重用性，从而增加系统的灵活性和扩展性，使得系统能够稳定运行。

# 项目实施方案

## 项目进度控制

项目的建设与项目的建设有关管理规定、技术架构、标准规范等方面要求。在实施过程中将发生的重大质量保证活动或由此将产生的质量记录和产品，项目管理与开发阶段划分密切相关，因此主要按照项目实施的具体阶段划分说明。



## 项目进度保证

在本项目进行项目进度管理工具使用要求如下：

（1）建立完整的项目管理进度计划，形成基线；

（2）项目人力资源全部纳入系统中进行管理，做到人岗匹配，责任到人；

（3）及时更新和跟踪项目进度，调配分配的资源；

（4）定期向采购单位汇报工作进展和人员使用状况。(频度为每周一次，或根据实际需求安排)

（5）软件开发项目涉及面广较多，需要多个开发组同时进行，为保证项目的有效开展，项目间的沟通管理，将根据项目计划管理、项目跟踪与监控机制，协调各开发组、开发人员之间的进度计划。

### 进度跟踪与监控

项目监控的目的是为软件项目开发过程提供足够的可见性，以便当项目严重偏离计划时及时采取纠正措施。

项目监控包括根据估计、承诺和进度表跟踪软件计划执行情况，监控软件项目风险，当实际工作与计划有严重偏离时及时采取纠正措施。

项目计划是软件活动跟踪、汇报项目状态和项目计划修改的基础。项目经理监控软件活动，主要通过在所选出的软件工作产品完成时和在所选择的里程碑处，将实际的软件工作产品大小、工作量、成本和进度与计划相比较，来确定进展情况。当确定软件项目计划没有得到满足时，采取纠正措施。这些措施可以包括修订软件开发计划以反映实际的完成情况，以及重新策划遗留的工作或者采取措施改进性能。变更项目计划必须遵循《变更控制规程》。

1. 项目计划跟踪

项目计划用于跟踪项目活动和通报项目状态。项目计划的修改反映实际完成情况，有效协调软件工程组、项目经理、高级管理者和其他相关组的活动。

项目计划跟踪的内容包括：

1.软件规模(WBS)

2.项目进度(里程碑、甘特图)

3.风险、问题

4.下列情况需要修改项目计划：

当计划有明显改变时，或软件项目任务书的要求与项目不符时，需修改项目计划。在需求分析阶段，项目进度与计划相差10%以上，在设计阶段，项目进度与计划偏差25%以上，或者各阶段累积的进度偏差在10%以上。

修改后的项目计划应注意：

1.反映所有新的软件项目承诺和承诺的变更

2.对修改的项目计划进行评审并重新批准

3.项目计划要进行版本控制

4.项目估计的变更是在事件驱动的情况下进行的，例如额外的项目特征增加了原始估计的软件规模，因此需要更新估计的软件规模

5.项目经理应该定期组织项目评审，其评审的主要内容是确认计划的执行情况，评审会议要有会议纪要。

1. 软件规模跟踪

项目经理根据项目计划负责软件工作产品规模的跟踪，在必要时采取纠正措施。

1.比较实际代码量和项目计划中估计的代码量

2.明确与项目计划中估计值的偏离程度

3.监控影响软件工作产品大小的主要因素

4.监控每个软件工作元素大小的上下限，当其超出计划中的估计值时，需采取纠正措施

5.评价偏离造成的影响

6.定期监控和调整项目规模估计

7.软件规模估计的调整需形成文档，在影响软件承诺时，需得到相关组的同意

8.必要时，修改项目计划

1. 项目进度跟踪

1.跟踪项目进度，必要时采取纠正措施

2.比较活动的实际完成情况、里程碑和其它约束

3.确定与项目计划中明确要求的偏离

4.估计软件活动、里程碑是否延期和提前的工作量，其它的约束是否影响以后的活动和里程碑等

5.软件进度表的修改影响软件约束时，要与相关组商议并达成一致

6.本项目开发的软件分为几个子系统，要跟踪每个阶段单元的完成百分比，也就是说要跟踪每个单元的设计、编码、单元测试、集成测试和系统测试等

7.定期跟踪项目的关键依赖和关键路径

8.监控项目进度时间上下限，超出时要采取相应的措施

9.必要时，修改项目计划

1. 风险跟踪

风险跟踪的目的是监控识别出与成本、资源、项目进度、项目技术有关的软件项目的风险的状态，确保风险减轻措施的正确执行。先前计划的减缓活动的有效性取决于是否减缓了风险的可能性或潜在影响，或是否风险真的发生了。

1.在项目会议上，负责减缓识别的风险的项目组成员报告减缓活动状态及风险本身。对每个风险和减缓活动的状态都要进行跟踪和审查。

2.在项目会议上或里程碑会议评审项目风险状态。

3.如果风险仍然存在，审查可能性和影响值，必要的话进行更新风险管理计划。

4.当有其它信息可利用时(如：附加的风险和消除的风险等)修改风险的优先级和发生的可能性。

5.如果风险已发生或者正在发生，那么风险从潜在问题变成了一个实际的问题，应该对其进行跟踪和控制。把风险状态改变成为问题，并且把问题与措施一同记录到《项目问题日志》，必要时更新《风险管理计划》。

1. 项目会议

项目会议是沟通项目情况的主要机制之一，是常规项目管理过程的一部分，所有的项目组成员都有责任呈现其活动状态并对潜在的和实际的问题和风险进行沟通。

项目组定期评审项目技术进展、结果和问题，项目经理在项目计划里确定项目的时间和频率，并负责提供一个能够开放的、公正的进行状态报告和讨论的环境，项目会议应形成会议记要，并分发给相关人员。

1.项目组成员要以项目计划为基础执行活动。

2.定期(建议每周进行)与项目经理和个人交流某一项目活动和软件工作产品的技术状态。

3.收集并分析用于控制软件项目的度量数据。

4.识别严重问题和偏离项目计划的问题。

5.记录对软件工作产品和过程的更改请求和问题报告单。

6.跟踪变更请求和问题报告，直到问题被解决。

7.必要时，修改项目计划。

1. 正式里程碑评审

在里程碑处必须组织软件产品评审，提供的证据和考虑的问题如下：

1.软件工作产品是否完成

2.软件工作产品符合规范

3.软件活动符合项目的进度要求

4.项目组准备进行下一步活动

5.项目开发和维护活动按照项目计划、进度表和组织的标准进行项目经理负责组织正式的评审(如：设计准备评审、编码准备评审、测试准备评审、发布准备评审等)，评审项目计划中指定的里程碑处项目的结果和完成情况。项目组成员、软件质量保证员、软件配置管理员应参加里程碑评审会议。

在项目进度表中的某一时间点组织评审，如可以选择在阶段(需求、设计、编码/单元测试、子系统测试、系统测试等)完成时进行。在适当时，客户、最终用户、软件供应商、组织内其它相关人员参加评审会。里程碑评审包括以下讨论议题：

1.评审会涉及项目承诺、计划、状态和项目风险等内容；

2.明确严重问题；

3.估计解决问题的工作量；

4.记录评审结果、行动计划和决定；

5.必要时，修改项目计划。

1. 项目报告

项目组成员等通过每周填写《项目周报》报告他们的活动结果，由项目经理负责收集和整理成项目报告。

1.内部报告：项目例会根据需要每周召开一次，形成正式的进度报告。

2.外部报告：与客户每两周召开一次工作例会；与协调小组不定期召开会议。

1. 有效输出

1.更新的项目计划(项目计划、软件质量保证计划、软件配置管理计划)。

2.非正式项目评审、正式里程碑评审、每月项目评审的评审记录或会议纪要。

3.纠正措施。

## 项目质量管理

### 编制和评审质量计划

为加强项目质量管理和界定产品质量标准，项目实施过程中需制订适应于项目的检查验收规定和质量评定标准，确保项目质量。

制定质量保证计划：依据项目计划及项目质量目标确定需要检查的主要过程和工作产品，识别项目过程中的干系人及其活动，估计检查时间和人员，并制定出本项目的质量保证计划。

质量保证计划的主要内容包括：例行审计和里程碑评审，需要监督的重要活动和工作产品，确定审计方式，根据项目计划中的评审计划确定质量保证人员需要参加的评审计划。明确质量审计报告的报送范围。

质量保证计划的评审：质量保证计划需要经过评审方能生效，以确保质量保证计划和项目计划的一致性。经过批准的质量保证计划需要纳入配置管理。当项目计划变更时，需要及时更改和复审质量保证计划。

### 过程和产品的质量检查

根据质量保证计划进行质量的审计工作，并发布质量审计报告。

审计的主要内容包括：是否按照过程要求执行了相应的活动，是否按照过程要求产生了相应的工作产品。本项目中对质量的控制主要体现在不同阶段的审计当中。

### 不符合项的跟踪处理

对审计中发现的不符合项，要求项目组及时处理，质量保证人员需要确认不符合项的状态，直到最终的不符合项的状态为“完成”为止。

### 质量管理责任分配

在开发项目上需按照规范化软件的生产方式进行生产。每个项目除配备了项目开发所需角色外，还需配备质量保证小组、配置管理小组、测试小组等来确保质量管理的实施，下面针对这三种角色进行说明；

#### 质量保证小组

质量保证小组作为质量保证的实施小组，在项目开发的过程中几乎所有的部门都与质量保证小组有关。质量保证小组的主要职责是：以独立审查方式，从第三方的角度监控软件开发任务的执行，分析项目内存在的质量问题，审查项目的质量活动，给出质量审计报告。就项目是否遵循已制定的计划、标准和规程，给开发人员和管理层提供反映产品和过程质量的信息和数据，使他们能了解整个项目生存周期中工作产品和过程的情况，提高项目透明度，从而支持其交付高质量的软件产品。

质量保证人员依据质量保证计划，通过质量审计报告向项目经理及有关人员提出已识别出不符合项，并跟踪不符合项的解决过程，通过审计周报或审计月报向项目经理提供过程和产品质量数据，并与项目组协商不符合项的解决办法。

质量保证小组的检测范围主要包括：项目的进度是否按照项目计划执行，用户需求是否得到了用户的签字确认，软件需求是否正确的反映了用户的需求，是否将每一项用户需求都映射到软件需求；系统设计是否完全反映了软件需求；实现的软件是否正确的体现了系统设计；测试人员是否进行了较为彻底的和全面的测试；客户验收和交接清单是否完备；对于系统运行中出现的问题，维护人员是否记录了详细的维护记录；配置管理员是否按照配置管理计划建立了基线，是否严格控制变更过程，是否对配置库进行了维护。

#### 配置管理小组

配置管理活动的目的是通过执行版本控制、变更控制、基线管理等规程，借助配置管理工具的使用，来保证整个生命周期过程产生的所有配置项的完整性、一致性和可追溯性。配置管理是对工作成果（阶段工作成果和产品成果、进展状态成果）的一种有效保护形式，是反映项目及其工作产品的过去、现在、动态的资料和数据集中管理体现。

配置管理小组的主要职责包括：根据项目计划制定配置管理计划，建立配置库，为项目组人员分配配置库权限，创建需求、设计、开发、测试、交付阶段的基线。当纳入基线库的工作产品发生变更时，严格按照配置项变更控制过程执行变更，变更后建立新的基线。

#### 测试小组

作为质量控制的主要手段，如同软件开发一样，测试在执行之前，测试小组制定软件测试计划、测试用例的编写和执行工作。

本项目中，测试可以分为如下几种类型：代码走查、单元测试、集成测试、系统测试。为了保证程序的质量，开发人员需要对同伴的代码进行代码走查，同时对自己编写的程序进行单元测试，确保程序编译、运行正确。

测试人员根据软件需求分析报告进行软件集成测试用例和系统测试用例的编写。对编写完成的测试用例提交项目组进行评审，同时质量保证人员对评审过程和工作产品进行监测。

测试人员根据测试计划和测试用例执行测试用例，并对发现的缺陷进行记录，只有这样才能确保项目组开发的软件产品满足用户需求。在完成集成测试之后，可以进行软件系统测试，系统测试包括对软件进行功能测试、性能测试、安全测试、压力测试。只有进行了系统测试软件测试才是完整的。系统测试在本项目中占有重要的地位，性能要求有可能改变软件的设计，为避免造成软件的后期返工，测试在性能上需要较大的侧重。

### 质量控制过程

本项目中，由项目经理制订质量控制计划，项目质量控制组进行审核。审核方面包括：质量控制措施是否足够、各个成员的质量责任是否明确合理，测试方法是否适用。

### 实施需求管理

通过实施需求管理，建立对软件需求的评审机制，保证对软件需求的版本进行控制，实现对用户需求，软件需求，测试需求的跟踪。

使用户与设计人员之间建立良好的沟通机制，使用户和软件项目人员之间达成共识，建立维护和管理软件项目中的客户需求。

需求管理强调：

控制对需求基线的变更；

保持项目计划与需求一致；

控制单个需求和需求文档的版本；

管理需求与软件开发过程中其他产品之间的依赖关系

跟踪基线中需求的状态。

## 项目管理

### 项目定义

项目定义的主要工作，即制定《项目任务书》。项目任务书应该明确任务起始日期，角色及其职责。项目任务书必须经过高层管理者的批准，批准就等于授权允许推进到项目的初始阶段。

### 定义项目生命周期模型

在项目策划的早期阶段，应该确定项目生命周期模型，只有确定了项目的生命周期模型，才能做下一步的任务拆分结构及估计。

### 识别需求

确定待产生的需求及计划完成的日期这一步很重要。因为在项目策划期间，要定义许多项目任务，其中的一部分就是根据需求定义的。项目执行时，进度度量的基础就是需求的完成情况。

### 建立工作拆分结构(WBS)

工作拆分结构(WorkBreakdownStructure)是将项目的活动和任务以层次化的方式表示的一种方法。把整个项目分解为可独立安排、易实施、易跟踪的工作元素，作为计划、组织和控制项目工作的基本框架。工作任务的拆分可以减少重要的项和活动被忽略的可能性，有助于在逻辑上识别所有必须的项目活动和其相互关联关系。可以利用工作拆分结构识别和提高项目模块的可重用性。

工作拆分的目的是精确地估计项目规模并建立详细的进度表，并为估算和制定日程表提供基础。工作拆分结构应体现软件项目任务书要求，从而达到产品目标，工作拆分的程度应达到能有效计划和控制每一工作元素的程度，最底层的工作产品应相关连。

工作拆分结构应描述工作元素之间的内部关系并为每个工作元素指派负责人。

在项目的早期阶段，可能没有足够的信息为以后的阶段做详细工作拆分，但必须确定关键里程碑。WBS可能随着项目的进行而修改，WBS的更改必须通知所有相关人员，并得到相关人员的认可。

### 规模估计

在建立了工作拆分结构后，应对项目的规模、工作量和成本、关键计算机资源和支持工具、进度等进行估计。

### 制定项目进度表

在确定项目的项目生命周期模型，并进行工作拆分及对软件的规模、工作量、关键计算机资源等进行估计后，应制定项目进度表。项目进度表包括以下内容：

1.项目里程碑、任务、承诺、关键依赖、人员及工作量；

2.确定各活动的时间段；

3.对影响项目进度表的问题应进行评审，并达成一致，在项目进行过程中，可修订项目时间进度表。

### 识别并分析项目风险

风险指在项目过程中可能发生的对产品结果产生不利影响的事件。

1.项目策划阶段，项目经理负责识别、分析、降低、消除并跟踪项目风险，使其不影响项目目标的实现；

2.项目风险时，项目经理组织尽可能多的与项目成员进行讨论，以获得多方面的意见。完成项目风险的识别后，项目经理制定风险列表，为项目风险跟踪提供依据。

### 制定项目基准计划

此处的项目基准计划是一个统称，包括项目开发计划、质量保证计划、软件配置管理计划、测试计划等。制定的项目基准计划必须获得相关组和个人的承诺，相关的组和个人包括项目组成员、质量保证员、高级管理者、测试人员、工程人员及其它有项目有关人员。项目基准计划要进行版本控制。

### 有效输出

项目基准计划：包括工作拆分、工作量估计、项目进度表、配置管理计划、质量保证计划、风险管理计划、项目培训计划等。