# 客服数据报表系统需求说明书

**目录**

[客服数据报表系统需求说明书 1](#_Toc7472)

[一、 项目背景 3](#_Toc12113)

[二、 需求调研 4](#_Toc14374)

[三、 系统架构 5](#_Toc7021)

[四、 功能概括 7](#_Toc1521)

[五、 预期工时 8](#_Toc8266)

[六、 附件内容 9](#_Toc16066)

[七、 问题 10](#_Toc26951)

[八、 有待补充 10](#_Toc11422)

1. **项目背景**

业务数据增长迅速，但用于经营分析和业务管控比较少信息化过程中，企业应用越来越多，数据量也越来越大，企业数据是一笔巨大财富，多数公司认可此观点却不知怎么利用，所以数据仅仅是堆积在一起，而参与运营决策，进行经营分析和业务管控的数据还非常少。

报表展现过于明细和简单，不利于管理者掌握企业全局信息业务系统带有的报表一般功能都很低，只能实现基本的数据汇总、展示功能，不能直观的、丰富的展现分析数据，更不能通过模型进行业务的预警、预测，支撑企业多种业务运用，不利于管理者掌握全局信息，全面决策。

当前系统需定期生成信息摘要并呈报征信中心管理层，但传统架构仅支持Excel文件生成功能，仍需人工二次转存至Word文档进行报送。系统尚存查询周报、月报功能模块待开发，资产登记与工单管理模块存在遗留Bug缺陷。现有架构存在响应效率亟待优化、数据可视化界面需进一步优化调整等技术瓶颈。鉴于此，上述痛点催生了报表项目2.0的立项推进。

1. **需求调研**

## 项目需求

1. 生成数据进行周月报送。
2. 数据在页面可视化展示。
3. 系统在使用时安全稳定。

## 报表种类

**查询：**查询周报、查询月报。

**异议：**异议周报、异议月报。

**电话：**电话周报、电话月报。

**其他：**部门月统计数据表、工单管理表、资产登记表。

（异议数据因数据实时变化，可以延迟3~5天取数据）

## 业务需求

通过系统可以生成出对应逻辑加工后的数据excel，通过这份数据再加上人工加工生成出对应的周报、月报。

**后续方案**：可以分别两种不同文件类型，分别为word（报送表）与excel（数据表）。直接生成word直接可以减轻人工操作，生成excel可以用于人工核对数据的正确性，查询数据的细节。报送的word格式固定，基本不会发生改变，因为格式固定，可以通过程序来生成。（还需要再讨论）

1. **系统架构**

## 技术架构

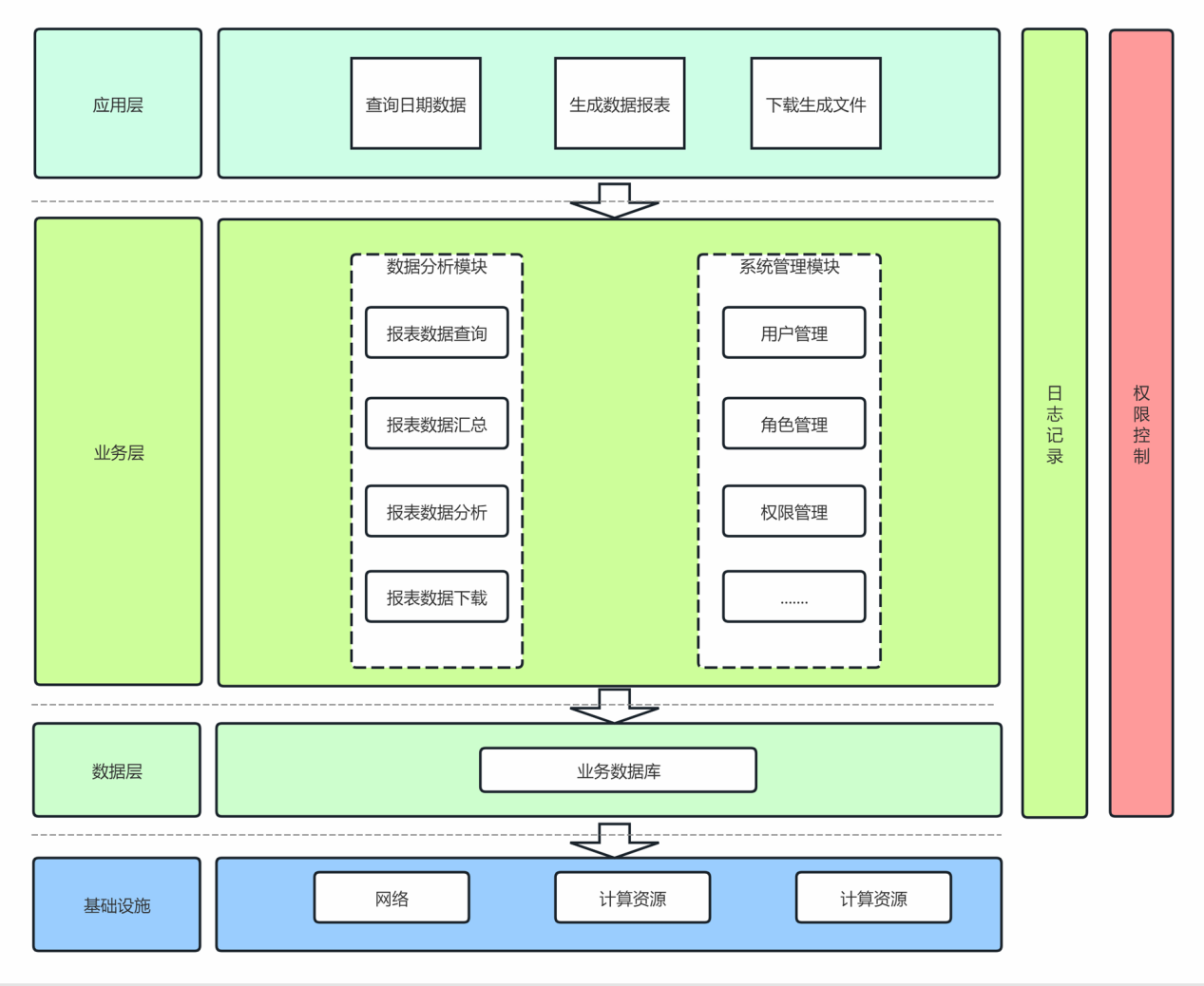
采取前后端分离式开发，前端采用Vue.js框架构建可视化界面，使用Element UI用于统一前端样式，后端基于Spring Boot单体架构搭建业务逻辑层。通过RESTful API规范实现数据传输交互，数据库选用Oracle部署保障事务处理能力，配合Redis缓存热点数据提升查询性能，引入Docker容器化部署方案，实现开发环境与生产环境的高度一致性。

## 技术概述表

|  |  |
| --- | --- |
| 技术名 | 概述 |
| Vue.js | 前端技术/框架Vue.js是一个渐进式JavaScript框架，专注于视图层，采用MVVM模式，支持双向数据绑定。 |
| Spring Boot | 后端技术/框架Spring Boot是一个简化Spring应用开发的框架，提供强大的配置和简化了的依赖管理。 |
| Redis | 数据库/缓存技术Redis是一种支持多种数据结构的NoSQL数据库，用于缓存热点数据以提升查询性能。 |
| Docker | 软件/容器化技术Docker是一种容器化技术，用于部署和管理应用程序的容器。 |
| Oracle | 数据库/软件Oracle是一个广泛使用的数据库管理系统，用于事务处理和数据存储。 |
| Element UI | 前端技术/组件库Element UI是一个基于Vue的UI组件库，提供预定义组件来快速搭建用户界面。 |
| Nginx | 服务器/负载均衡Nginx是一个高性能的HTTP和反向代理服务器，用于负载均衡和反向代理。 |

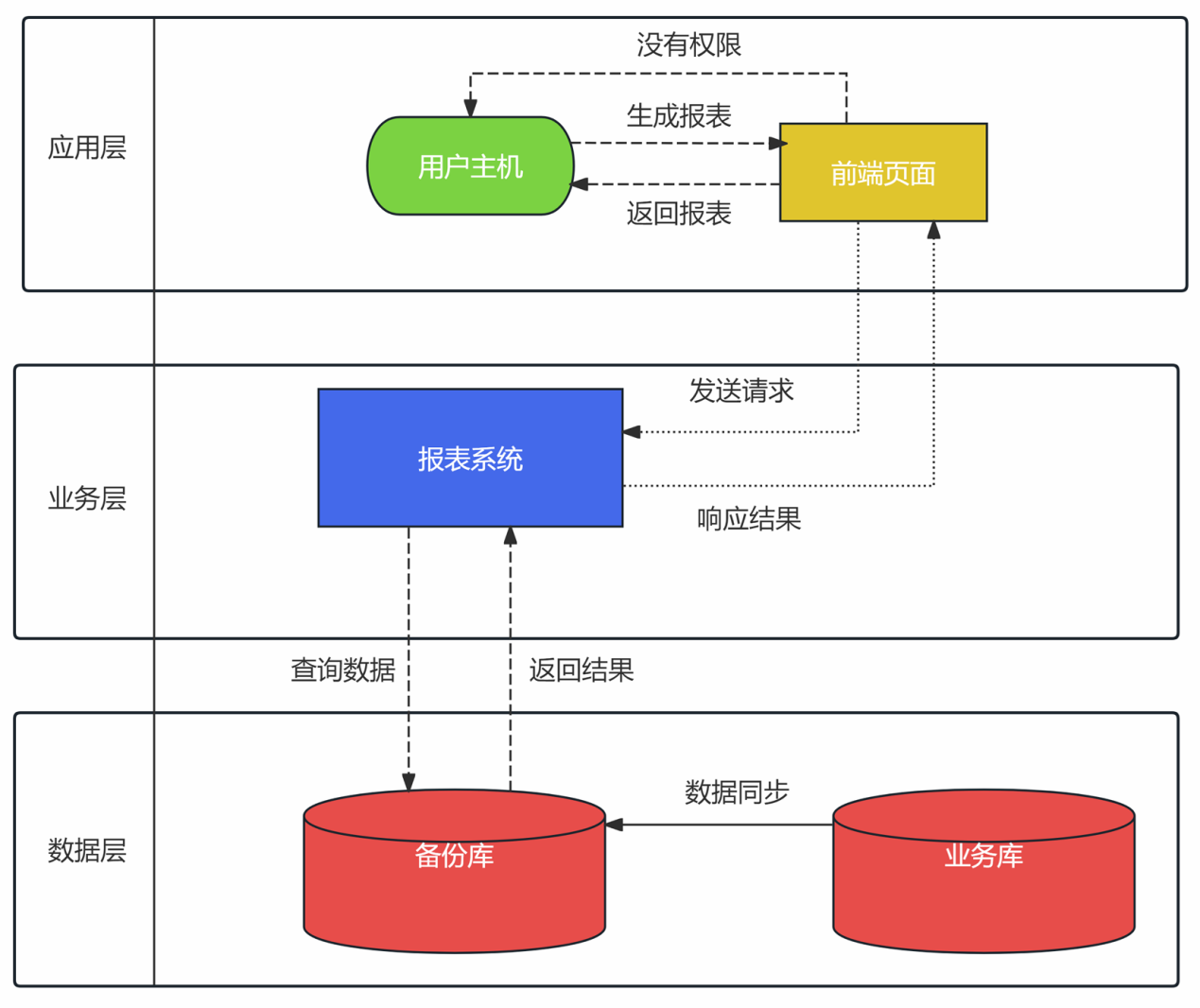
**表1 技术概述表**

## 系统架构图



**图1 系统架构图**

## 系统流程图



**图2 系统流程图**

1. **功能概括**

## 数据清洗

把数据表数据，通过特定的条件进行处理得到需要的数据。

## 在线图表

用户在系统中可以直观的看到统计图表，并保存图片。

## 文件导出

可以将加工后的数据通过excel导出。

## 操作日志

记录操作账号的各种操作（报表导出、账户登录时间）。

## 权限管理

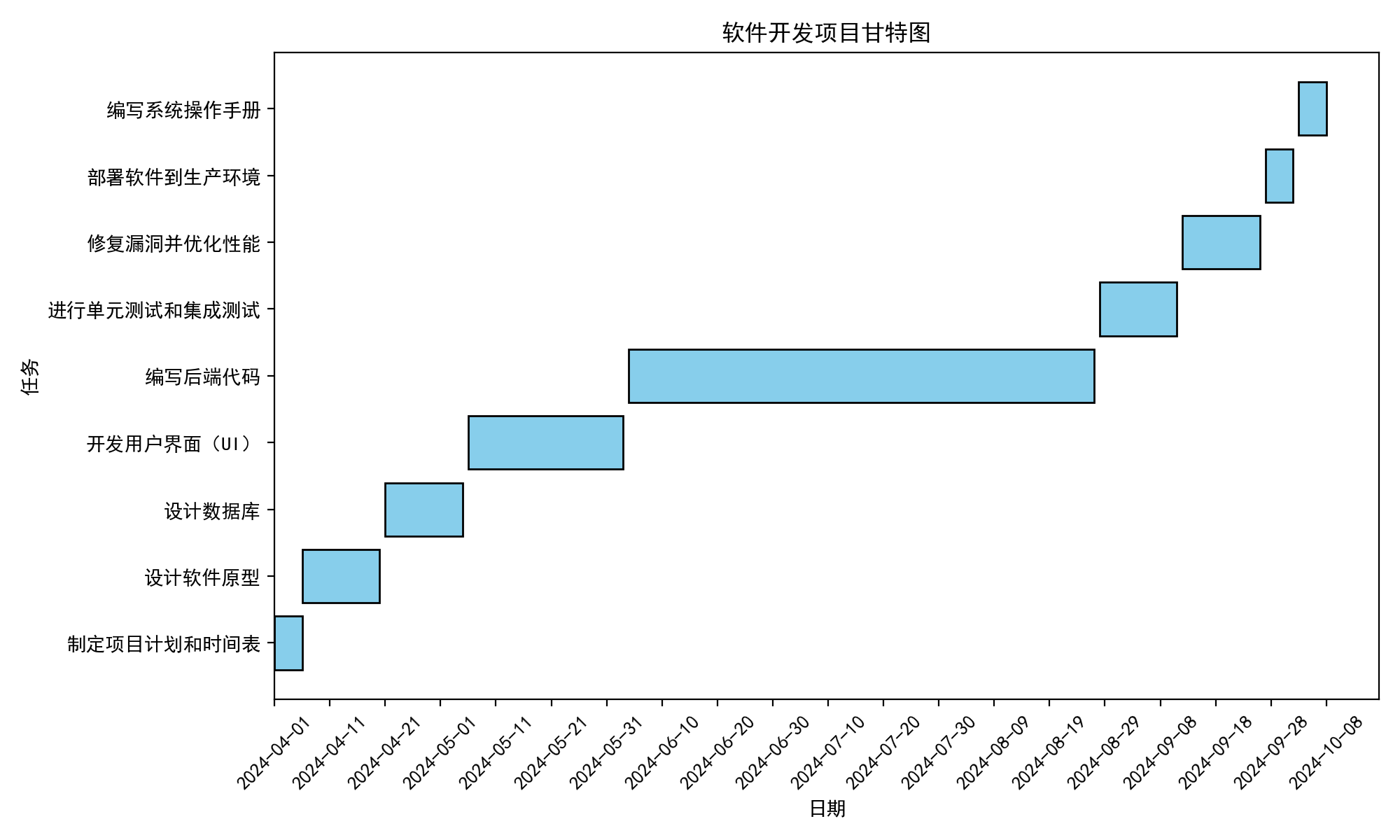
功能页面可以配置账户权限（使用MD5加密前后端数据）。

1. **预期工时**

## 项目排期

1. 制定项目计划和时间表（一周）
2. 设计软件原型（两周）
3. 设计数据库（两周）
4. 开发用户界面（四周）
5. 编写后端代码（十二周）
6. 进行单元测试和集成测试（两周）
7. 修复漏洞并优化性能（两周）
8. 部署软件到生产环境（一周）
9. 编写系统操作手册（一周）

## 项目甘特图



**图3 甘特图**

1. **附件内容**

## 系统生成报表

1. 异议周报
2. 异议月报
3. 电话周报
4. 电话月报
5. 综合日报
6. 部门月统计数据
7. 客服日报917
8. 客服日报510
9. 工单管理
10. 资产管理
11. **常见问题**

## 数据库需不需要重新设计？

我认为是需要的，因为目前的数据库，设计的并不是很好。有至少180张表没有任何存在的意义，但删除后会影响老系统功能的使用（问题原因还没有排查），提高了后续系统的上手成本、维护成本。

## 目前系统有哪些功能没开发完？亦或是无法使用？

绩效表已经移动至其他部门（无需开发）；工单管理表、资产登记表 无法正常使用（不确定是否需要开发）；查询周报、查询月报 目前在旧系统中（没有开发完成），需要咨询业务逻辑。