# 性能数据管理系统

概要设计文档

Version：1.2

Date：2021-09-19

目 录

[性能数据管理系统 1](#_Toc82808070)

[1. 背景介绍 1](#_Toc82808071)

[2. 系统分析 1](#_Toc82808072)

[2.1 可行性分析 1](#_Toc82808073)

[2.2 难点 1](#_Toc82808074)

[2.3 项目目标 1](#_Toc82808075)

[3. 系统架构设计 2](#_Toc82808076)

[3.1 架构设计 2](#_Toc82808077)

[3.2 主要功能模块设计 3](#_Toc82808078)

[3.2.1 用户功能 3](#_Toc82808079)

[3.2.2 管理员功能 5](#_Toc82808080)

[Revision History 6](#_Toc82808081)

# 1. 背景介绍

外场测试数据与GNSS测试数据非常庞大，测试人员阶段性提供的性能测试数据比较分散，想要对比类似的数据难以查找，将各项测试数据统计比较是非常有必要的，为了能更直观的展示测试数据以及对比版本迭代、不同城市、不同运营商等之间的差异，发掘数据之间隐藏的不同关系，项目小组开发一个性能数据平台系统，将以前的测试数据统计到性能平台，后续新的测试数据也可以通过性能平台输入，通过大数据性能平台系统可以方便的将这些数据作以统计和对比。性能数据平台可以方便查找和对比测试数据，更直观的了解类似测试数据之间的差异性，为企业带来更高的效率。

# 2. 系统分析

## 2.1 可行性分析

应用可行性: 本Web网站的目的是对外场和GNSS性能测试数据进行管理，将测试数据统计到性能平台，通过大数据性能平台系统可以方便的查询到测试数据，有效管理数据，提升工作效率。

技术可行性: 当前存在多种组合开发工具，综合考虑团队成员对各工具的技术能力。语言上，Python能简单有效地面向对象编程，所以该项目采用python语言进行开发。 Django 是 Python 开发的一个免费开源的 Web 框架，提供了许多网站后台开发经常用到的模块，使开发者能够专注于业务部分，并且囊括的 Web 应用部分可以用于快速搭建高性能、优雅的网站，故采用Django框架。此外，mysql是一种开放源代码的关系型数据库管理系统，使用最常用的数据库管理语言–结构化查询语言进行数据库管理。mysql因为其速度、可靠性和适应性而备受关注。大多数人都认为在不需要事务化处理的情况下，mysql是管理内容最好的选择。团队成员对上述均有一定的了解，故相信技术上能够达到要求。

## 2.2 难点

外场和GNSS的性能数据不一，数据量庞大，因此需要整理出统一标准。

## 2.3 项目目标

我们的系统旨在通过网站对性能测试数据进行管理，实现一键导入、一键导出、按需查询、在线可视化分析等功能增、删、改、查等功能。此外，能够通过关键字查询，获取测试数据，实现导入导出。

# 3. 系统架构设计

## 3.1 架构设计

此项目选用Django框架中的MVT模型，如图1所示。其中，MVT分为三个模块，Model，View和Template。Model，缩写M，主要封装对数据库层的访问，实现对数据库数据增、删、查、改等操作。View，即V，接收来自客户端的请求，进行业务处理，返回应答。Template，即T，负责封装结果，生成页面所需展示的html内容。

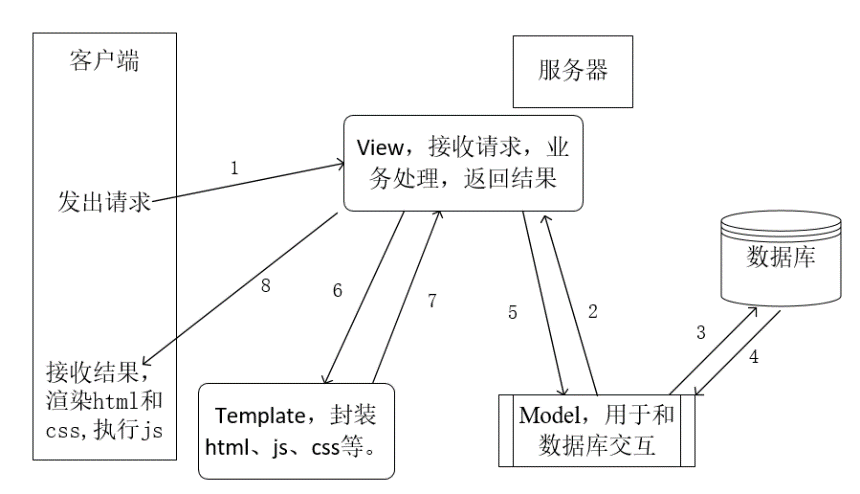


图1: MVT模型

整个系统如图2所示，主要分为前端，后端，数据库设计三个部分，前端的开发采用Bootstrap框架，将网站页面开发常用的样式、组件、特效集中在一起，极大提高页面开发的效率。数据库设计采用mysql对性能数据系统的基本需求进行相关字段设计。后端控制前端的内容，主要负责程序架构设计，管理数据库，并与服务器进行交互，实现用户和管理员在平台上的操作。

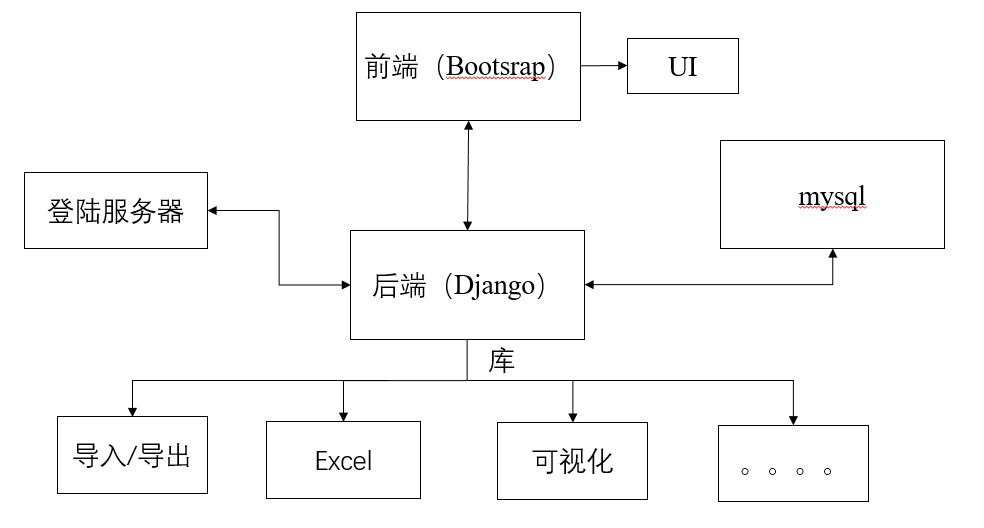


图2: 系统设计框架

## 3.2 主要功能模块设计

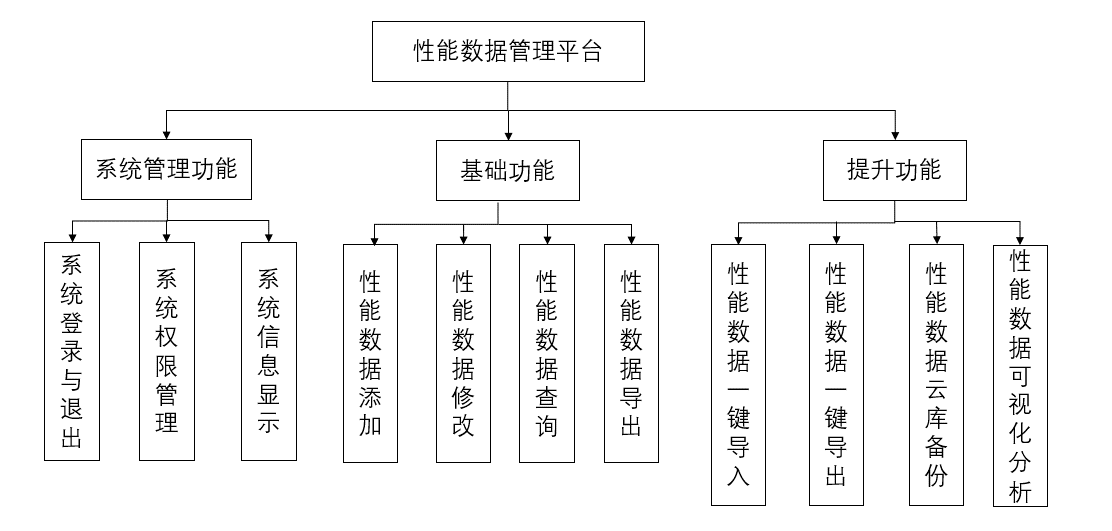
性能数据管理平台主要分为系统管理功能，基础功能和提升功能，如图3所示。

图3: 性能数据管理系统功能类别

系统管理功能分为系统登录/退出，系统权限管理和系统信息显示；系统登录和退出主要考虑到平台使用者的数据安全性。系统权限管理主要针对管理员和普通用户，管理员相比较于普通用户拥有更大的权限，壁如管理员可以对性能系统上的数据进行修改和删除，普通用户则没有。系统信息显示可以项目，版本信息等。

基础功能分为性能数据添加，修改，查询，导出，方便使用者对性能数据(GNSS&FT场测)进行整合，筛选出关键信息。

提升功能分为一键功能，云库备份，可视化分析，其中操作者使用一键功能(导入/导出)，无需繁琐的操作就可以将数据提交整合，极大的便利了使用者；云库备份，可以将本地的性能数据上传至云测服务器中，避免了因为数据丢失产生了安全隐患。可视化分析，可以将关键数据直观形象的展示给使用者，挖掘数据之间的关联性。

系统基于用户和管理员两个角度考虑，受众群体主要有两大类：一是性能数据管理员，二是普通用户。根据用户类型的不同，将系统划分为普通用户端和管理员端。

3.2.1 用户功能

用户是我们整个系统的服务对象，其各功能的有效、便利直接关系到用户体验。为方便对FT和GNSS数据均适用，需制定统一的数据excel模板。用户按照标准模板进行数据上传，更便于数据管理。当用户需要历史数据，按照自己需求筛选条件进行一键导出，方便有效。

（补充了一下功能，将登录这个功能作为了其一，个人觉得用户功能这个分类不是很好，直接写主要功能怎么样？）登陆：设置登录用户名与密码，其默认值可分别为个人PC名和PC密码，支持自定义修改密码，支持记住密码以方便下次登录。

系统可自动将相关人员的账号识别，自动登录。首次登录时，若用户名和密码正确，则显示用户登录成功；若出现输入用户名错误/密码错误的情况，则会显示登录失败/用户不存在。

表1:用户登录功能

|  |  |
| --- | --- |
| 功能名称 | 用户登录 |
| 输入 | 用户名和密码 |
| 输出 | 登录成功、登录失败(密码错误、用户不存在) |

3.2.1.1 登录系统及界面设计

功能：

表1:用户登录功能

|  |  |
| --- | --- |
| 功能名称 | 用户登录 |
| 输入 | 用户名和密码 |
| 输出 | 登录成功、登录失败(密码错误、用户不存在) |

UI: 具有紫光展锐特色；

3.2.1.2 主页及功能按钮设计

主页：主页UI、功能按钮确定、功能按钮UI、非功能UI设计。

3.2.1.3 一键导入

*说明：考虑到手动输入数据会给测试人员带来二次记录的不必要性，因此设计一键导入功能，由后台服务器自动识别测试人员按模板进行上传的数据文件并记录到相应的数据库中。*

设计原则：

后端：1.通过Pandas库读取上传的数据文件，并暂时保存。

2. 设计数据库接口，将1中数据按照主键录入相应数据库。

前端：1. 在显示界面生成一键导入功能；2. 设计前段与后端的文件传输接口（暂时仅支持excel格式）

3.2.1.4 一键导出

*说明：此功能目的为：按需将测试人员所需数据按照.xlsx、.csv、.xml等文件格式输出到本地。*

设计原则：

后端：1. 编写脚本将mysql数据文件转换为需要输出的相应格式文件；2. 设计输出接口。

前端：1. 设置“一键导出”功能按钮；2. 设置多种格式选择提示框；3.设置下载功能，搭配与后端的接口设计；

3.2.1.5 按需查询

*说明：此功能目的是为了测试人员能够对想要进行纵向或横向比较的数据标签按需组合并查询。*

设计原则：

后端：1.数据库的多个子表精准查询；2. 按所筛选数据生成新的临时数据表；

3. 设计缓存与清除功能，避免服务器爆仓；

前端：1. 设计查询筛选框；2.设计输出界面展示方案；

3.2.1.6 在线可视化分析

*说明：此功能旨在按照筛选标签，在线进行可视化展示，帮助测试人员快速、直观分析数据。*

设计原则：

后端：1. 按所选标签将筛选出的数据构建到新表中；2. 利用Python将数据表转化为利于进行在线可视化的数据类型；3. 生成多种（所选）类型的图形文件。

前段：1. 设计可视化选择及展示界面；2. 待补充。。

查询：用户可以通过平台查询测试数据。查询字段维度：按搜索条件进行查询；项目维度；版本维度，既可以看多个版本数据，也可以选择看完整最新数据。

表2:用户查询功能

|  |  |
| --- | --- |
| 功能名称 | 按条件查询 |
| 输入 | 项目、版本、Case名称等条件下的数据 |
| 输出 | 对应条件下的信息 |

上传：用户可以通过平台输入数据，上传方式可在线编辑后保存，也可将数据根据excel统一模板上传。

表3:上传数据功能

|  |  |
| --- | --- |
| 功能名称 | 上传测试数据 |
| 输入 | 按Excel模板统一数据 |
| 输出 | 上传成功 |

下载：用户根据各自需求，在需求筛选条件下将数据选择出来并可将数据下载到本地（excel标准模板）。

表4:下载数据功能

|  |  |
| --- | --- |
| 功能名称 | 下载数据 |
| 输入 | 将查询到的数据一键导出 |
| 输出 | 下载成功 |

3.2.2 管理员功能

管理员系统对整个系统进行管理，拥有最高权限，主要用于为全体用户服务，保证良好的用户体验和系统的平稳运行，主要包括用户管理和性能数据的管理。

用户管理：用户信息管理主要管理不同角色用户的信息，包括增加用户、修改用户信息、管理登录权限、查看操作日志等。

表5:用户管理功能

|  |  |
| --- | --- |
| 功能名称 | 用户管理 |
| 说明 | 对用户进行增加、删除 |
| 下属功能 | 修改用户信息 |
| 查看操作日志 |

性能数据管理：管理员可对用户上传的数据进行审核，如对错传的数据进行删除，对传错的数据进行修改，对补充的数据进行增加。

表6:性能数据管理功能

|  |  |
| --- | --- |
| 功能名称 | 性能数据管理 |
| 功能说明 | 对数据进行增、删、改、查 |

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Owner | Notes |
| 1.0. | 2021-09-17 | Yuhui.wan | 起草系统概要设计文档 |
| 1.1 | 2021-09-17 | Zepeng.zhao | 系统概要设计文档 |
| 1.2 | 2021-9-19 | Pei.wu | 完善概要设计文档 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |