**应用系统设计文档**

1. 引言

本文档描述了气象数据模拟分析系统的设计，包括软件架构、模块设计、数据库设计和用户界面设计等方面的内容。

2. 软件架构

气象数据模拟分析系统采用三层架构（Three-Tier Architecture），以实现展示逻辑、业务逻辑和数据访问逻辑分离。

2.1 模块划分

气象数据模拟分析系统包含以下主要模块：

展示层（Presentation Layer）：负责与用户进行交互，通常包括用户界面（UI）组件；处理用户的输入和输出，将用户的操作传递给应用层。

应用层（Application Layer）：接受展示层的请求，执行业务规则，并调用数据层进行数据存取。

数据层（Data Layer）：负责与数据库或其他数据存储系统进行交互；执行数据的存储、检索、更新和删除操作。

2.2 模块间通信

展示层（前端）通过HTTP请求（采用restful API）与应用层（后端）通信；

应用层通过直接调用数据层提供的函数或方法来访问和操作数据。

3. 模块设计

3.1 展示层（Presentation Layer）

设计用户界面，包括当日天气、历史天气、数据分析与可视化、气温预测、个人模拟分析、管理员界面等。

将处理逻辑写在前端，此处采用Qt的槽函数实现各种功能，并通过接口实现与数据库的数据交换等操作。

3.2 应用层（Application Layer）

采用flask框架编写restful API并挂在云服务器上运行，时刻响应前端发送的请求，并调用python编写的代码块，如爬虫、机器学习、多算法等。

3.3 数据层（Data Layer）

设计数据库结构，分为历史天气、当日天气、预测天气、用户数据等表格。

设计相应接口，相应API调用请求。

挂载在云服务器的固定端口上，使得API能够访问。

4. 数据库设计

气象数据模拟分析系统使用MySQL数据库来存储历史天气数据、预测天气数据、

用户数据、当日天气。

4.1 历史天气数据

字段：最高气温、最低气温、天气、风力、风向。

主键：城市、年、月、日。

4.2 预测天气数据

字段：年、月、日、温度、城市。

4.3 用户数据

字段：账号、密码、身份。

主键：账号。

4.4当日天气

字段：月、日、小时、温度、湿度、天气、风向、风力。

5. 用户界面设计

设计用户界面以提供用户友好的操作和交互体验。具体如下：

5.1密码设置

在注册账号时，用户需要重复输入密码以验证此密码是否输入正确。

可在注册页面返回至登录页面。

5.2可视化

在呈现历史天气数据时，并没有一味地采用文字形式赘述，而是结合图像形式给予用户更加直观的观感。

5.3等待界面

在机器学习预测天气时，因为云服务器上的内存限制和代码学习数据时间较长，需要用户耐心等待，因此设置读条以便于提醒用户。

6. 总体流程

整体流程如下：

6.1用户可以查看当日天气，并得到相关的小tips。

6.2用户可以查看历史天气数据。

6.3用户可以查看历史天气数据的可视化。

6.4用户可以预测某地的未来天气数据。

6.5用户可以调用多种算法进行数据模拟。

6.6管理员可以管理用户账号。

7. 接口定义

详细描述模块之间的接口和函数调用方式，包括展示层、应用层和数据层之间的接口。

展示层需调用Qt相关的库函数发送HTTP请求，API开放端口并连接数据库。