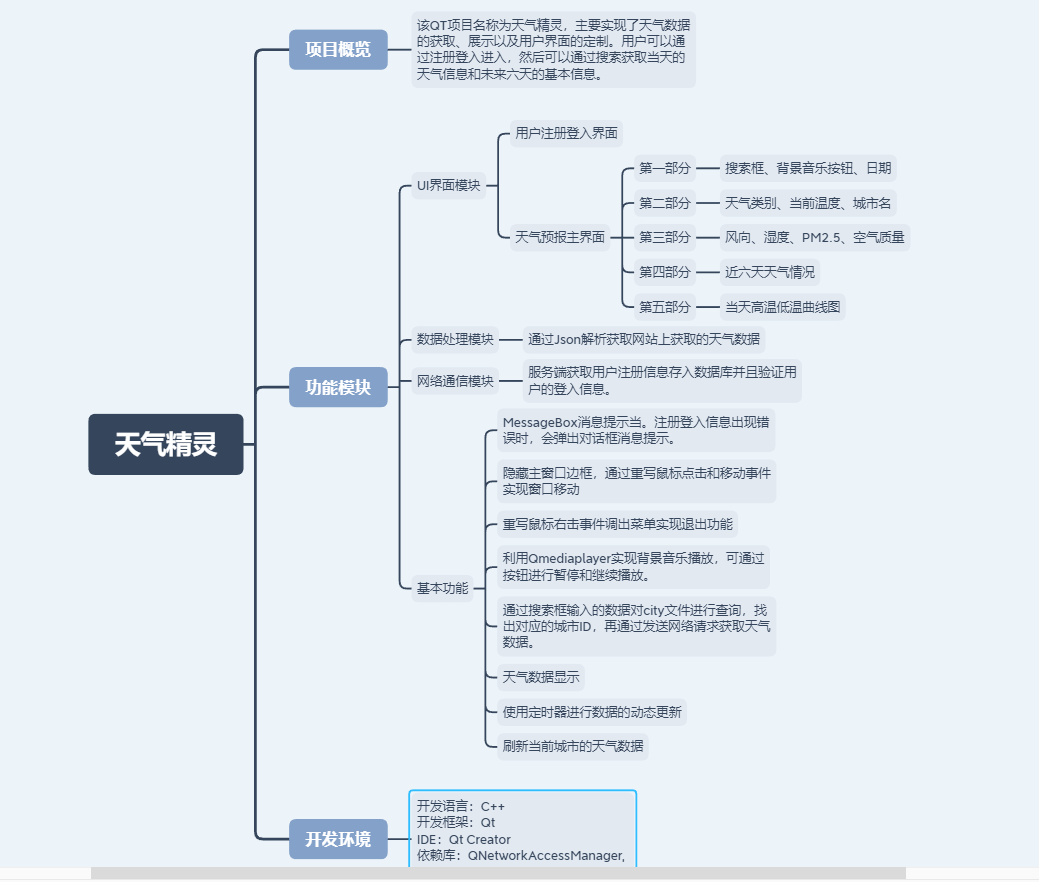
# 天气预报开发文档

## 1. 项目概述

本项目是一个基于Qt框架的桌面应用程序，包含主窗口MainWindow和天气窗口Weather。主窗口负责用户登录、注册以及与服务器的通信，而天气窗口则用于显示实时和未来的天气信息。



## 2. 主要功能

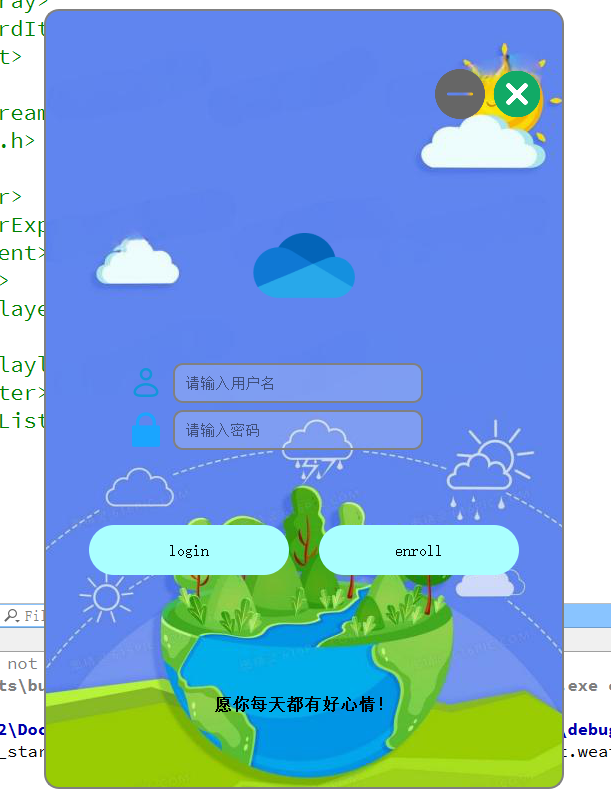
2.1 主窗口 (MainWindow)

用户界面：包含一个用于显示用户头像的标签、登录和注册表单以及几个按钮（关闭、最小化、登录、注册和切换天气窗口）。

网络通信：使用QTcpSocket与服务器进行连接，实现用户注册、登录以及接收天气信息的功能。

样式设置：通过加载style.qss文件来设置窗口的样式，包括边框、透明度和背景图片等。

窗口操作：支持窗口的无边框和透明背景设置，以及通过鼠标拖动来移动窗口。



2.2 天气窗口 (Weather)

定时器更新：使用QTimer每秒更新一次天气信息，包括温度、湿度、PM2.5、风级和空气质量等。

窗口操作：支持窗口的无边框设置，以及通过鼠标拖动来移动窗口。

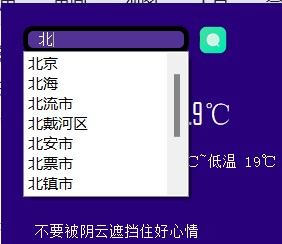
网络请求：通过QNetworkAccessManager向天气API发送GET请求，获取实时和未来的天气数据。

数据显示：使用QLabel控件显示天气信息，包括城市名、当前温度、湿度、PM2.5、风级、天气类型以及未来六天的天气预报。

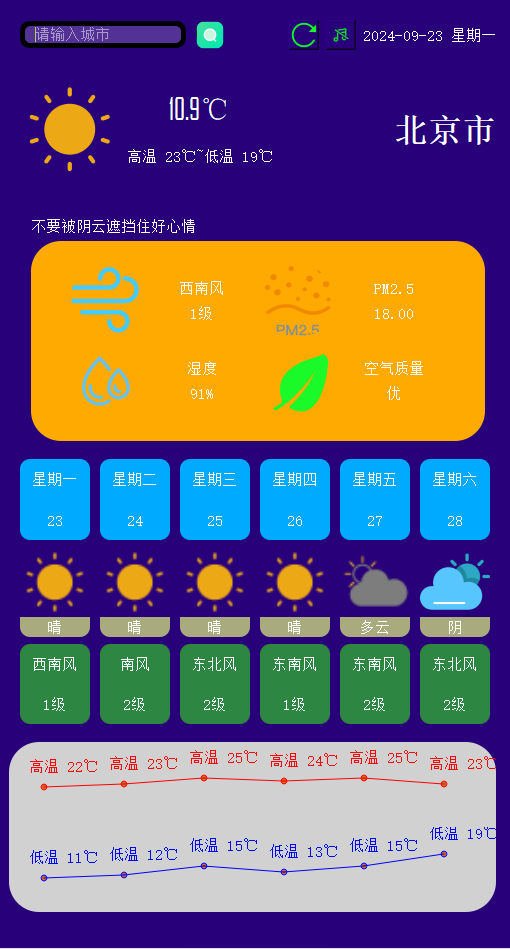
事件处理：通过重写eventFilter方法，在特定的widget上绘制最高温和最低温的折线图。

背景音乐播放：通过一个音乐按钮可以控制背景音乐的播放。

城市搜索：跟据搜索城市获取该城市的天气数据，且根据搜索内容显示相似城市。



天气数据刷新：可以跟据上次的搜索城市刷新天数据。



## 3.关键模块实现

3.1 构造函数（Weather::Weather）

功能：初始化界面、设置定时器（每秒更新）、发送网络请求获取天气数据、配置背景音乐播放等。

步骤：

调用setupUi初始化UI。

创建定时器，设置每秒更新界面。

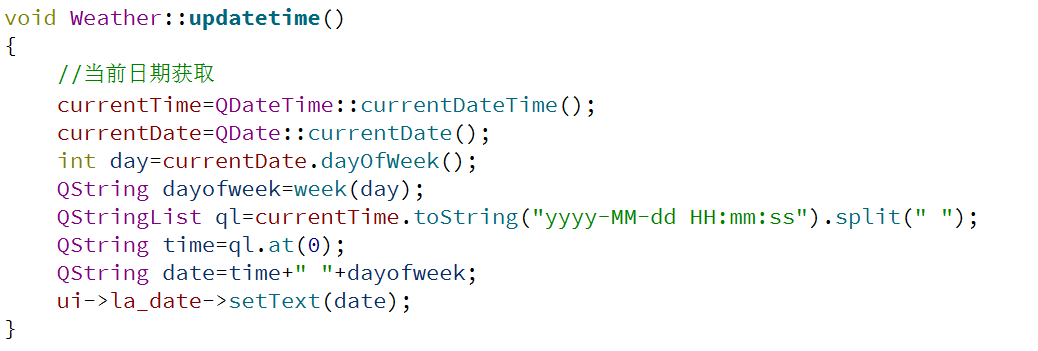
使用QNetworkAccessManager发送HTTP请求获取天气数据。

初始化背景音乐播放列表和播放器。

3.2 定时器更新（Weather::updatetime）

功能：更新当前时间和日期，并显示到界面上。

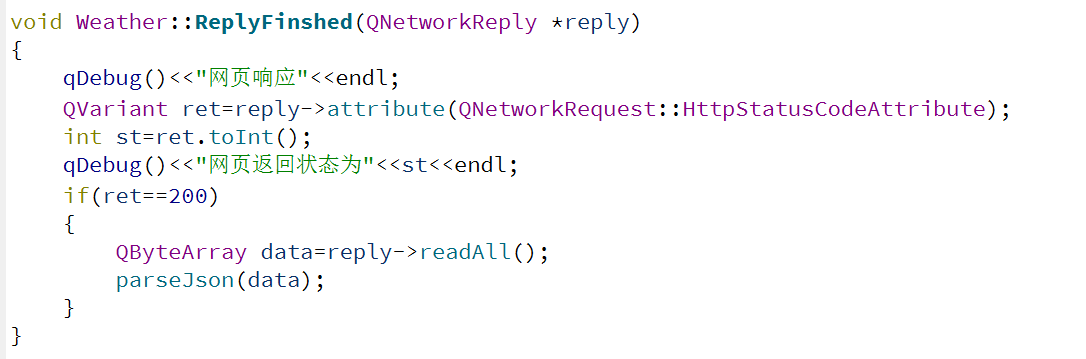
实现：使用QDateTime和QDate获取当前时间和日期，并转换成字符串格式显示。



3.3 网络请求响应处理（Weather::ReplyFinshed）

功能：处理网络请求返回的天气数据，并解析JSON格式的数据。

实现：检查HTTP状态码，读取返回的数据，并使用QJsonDocument和QJsonObject解析JSON。



3.4 JSON数据解析（Weather::parseJson）

功能：解析从服务器返回的JSON数据，提取城市信息、天气状况、温度、湿度等，并更新到界面上。

实现：使用Qt的JSON处理类（如QJsonDocument, QJsonObject, QJsonValue等）解析数据。

3.5 天气图标显示（Weather::setpicture）

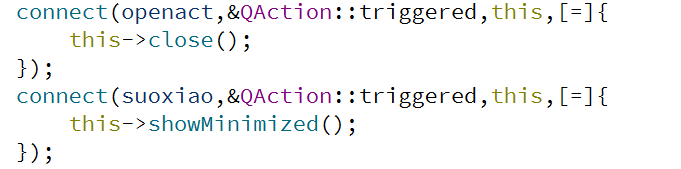
功能：根据当前天气类型，设置相应的天气图标到界面上。

实现：通过判断farr[0].type的值，设置picture控件的背景图片。

3.6 退出缩小菜单（鼠标右键事件处理）

功能：实现通过鼠标右键点击窗口，弹出退出菜单，并点击“退出”按钮关闭程序，点击“最小化”按钮缩小程序。

实现：重写mousePressEvent和mouseMoveEvent，并创建退出菜单和动作。







3.7 事件过滤器（Weather::eventFilter）

功能：处理特定控件的绘制事件，用于绘制天气数据的图形化展示（如温度曲线图）。

实现：重写eventFilter函数，对widget\_10和widget\_11的绘制事件进行特殊处理。



## 4.后续开发计划

增加更多天气信息的展示，如日出日落时间、紫外线强度等。

增加地图显示和天气预警功能。