

WeatherForecast文档

需求分析文档

项目名称：天气预报工具

1. 简介

天气预报工具是一个基于 PyQt 的 GUI 应用程序，旨在为用户提供实时的天气信息，并支持城市切换、手动刷新和保存天气数据的功能。该工具具有用户友好的界面设计，帮助用户轻松获取所需的天气信息，以便做出准确的出行决策。

2. 功能要求

2.1 显示当前城市的天气信息

- 显示当前城市的温度、湿度、风力等实时天气数据。
- 通过直观的图标或符号展示天气状况，如晴天、多云、雨天等。
- 提供一个易于阅读的界面，清晰显示天气信息，以满足用户的信息需求。

2.2 支持切换城市

- 提供一个下拉菜单或类似的选择界面，允许用户从预定义的城市列表中选择所需的城市。
- 当用户选择不同的城市时，自动更新显示该城市的天气信息。

2.3 手动刷新天气信息

- 提供一个刷新按钮，允许用户手动更新当前城市的天气信息。
- 当用户点击刷新按钮时，应使用最新的数据更新天气信息，以确保显示的信息是最准确的。

2.4 保存天气信息到文件

- 提供一个保存按钮，允许用户将当前城市的天气信息保存到文件。
- 当用户点击保存按钮时，弹出文件保存对话框，允许用户选择保存的文件路径和文件名。
- 将天气信息以适当的格式（如JSON、CSV等）保存到用户指定的文件中，以便用户可以随时查看之前的天气数据。

3. 用户界面设计

- 使用现代化的界面设计风格，包括清晰的布局、合适的颜色搭配和易于阅读的字体。
- 使用图标或符号来表示不同的天气状况，以增加界面的可视化效果。

- 提供明确的标签和按钮，以使用户能够轻松理解和操作工具的各项功能。

4. 性能要求

- 响应快速：工具应以快速的速度获取天气数据并更新界面，以确保用户能够实时获取最新的天气信息。
- 稳定性：工具应具有良好的稳定性和健壮性，能够处理不同的用户操作和异常情况，并提供适当的错误处理和反馈机制。

5. 兼容性要求

- 支持跨平台：工具应能在主流操作系统（如Windows、MacOS和Linux）上运行，并提供相同的功能和用户体验。

以上是天气预报工具的需求分析，旨在提供一个功能完善、用户友好的应用程序，帮助用户获取准确的天气信息，并在日常生活中做出更好的决策。通过实现这些功能和要求，我们将为用户提供一种便捷、可靠的方式来获取天气预报数据。

设计文档

1. 技术选择

编程语言：Python

GUI 框架：PyQT

2. 整体设计

- 用户界面：使用 Qt designer 构建用户界面，包括主窗口和相关控件。
- 数据获取：使用天气数据的 API（'http://t.weather.itboy.net/api/weather/city/' + cityCode）获取天气信息。
- 数据处理：对获取的 JSON 数据进行解析和处理，提取所需的天气信息。
- 数据展示：将提取的天气信息显示在 GUI 中的相应控件中。
- 用户交互：实现用户切换城市、刷新天气、保存天气等功能。
- 文件操作：实现将天气信息保存到文件和从文件加载天气信息的功能。

3. 运行步骤

- 可通过终端python WeatherForecast.py运行程序
- 程序已打包，可通过直接打开\WeatherForecast2.0\dist\WeatherForecast中的WeatherForecast.exe文件运行程序

4. 界面展示与交互

启动程序后，用户将看到一个简洁而直观的用户界面，该界面由主窗口和相关控件组成。主窗口设计采用现代化风格，使用户可以轻松地进行操作。

用户可以通过以下方式与应用程序进行交互：

1. 切换城市: 用户可以通过选择城市下拉菜单中的不同城市来切换当前显示的天气信息。每个城市都有一个对应的城市代码，用于从天气数据 API 中获取该城市的天气信息。
2. 查询天气: 用户点击“查询”按钮后，应用程序将使用所选的城市代码从天气数据 API 中获取最新的天气信息。天气数据以 JSON 格式返回，并在应用程序中进行解析和处理。
3. 显示天气信息: 应用程序将从获取的天气数据中提取关键信息，并将其显示在界面的相应控件中。包括当前温度、最高温度、最低温度、湿度、风向和风速等。
4. 保存天气信息: 用户可以选择将当前城市的天气信息保存到文件中，以便随时查看之前的天气数据。保存功能将通过文件操作实现，用户可以指定保存的文件名和路径。
5. 刷新天气: 用户可以在任何时候点击“刷新”按钮来获取最新的天气数据，以更新界面显示的天气信息。

应用程序还提供了错误处理和提示机制，以确保用户在操作过程中能够获得准确和实时的反馈。例如，在网络连接失败时，用户将收到相应的错误提示，以便及时调查和解决问题。

5. 扩展功能

为了进一步改善用户体验和提供更多有用的功能，可以考虑以下扩展：

- 天气图标: 在界面中显示天气图标，根据天气状态（如晴天、多云、雨天等）显示相应的图标，使界面更加生动和直观。
- 多城市支持: 允许用户添加多个城市，并在界面中切换显示不同城市的天气信息。可以通过添加城市选择下拉菜单或使用标签页等方式实现多城市的支持。
- 预警信息: 对于有预警信息的天气，可以在界面中显示相应的预警信息，例如暴雨预警、大风预警等，以便用户及时了解并采取相应的措施。

6. 展望未来

以下是一些可以增强用户体验的功能，可以在将来的版本中考虑实现：

- 实时数据更新: 添加定时器功能，定时刷新天气数据，以保持界面显示的天气信息是最新的。
- 国际化支持: 添加多语言支持，允许用户切换不同的语言界面，提供英语、中文等多种语言选项，以满足不同用户的语言需求。
- 历史天气查询: 允许用户通过选择日期或时间范围，查询过去某一天或某一段时间的历史天气信息。这样用户可以回顾过去的天气情况，并进行比较和分析。

- 个性化设置: 允许用户进行个性化设置，例如选择界面主题颜色、调整字体大小、设置温度单位等，以满足用户的个性化需求。

通过以上设计，我们将构建一个功能完善、用户友好的天气预报工具，提供准确、实时的天气信息，并具备一定的扩展性和可定性。