作品名称： 城科微信新闻小程序

作 者： 李建立

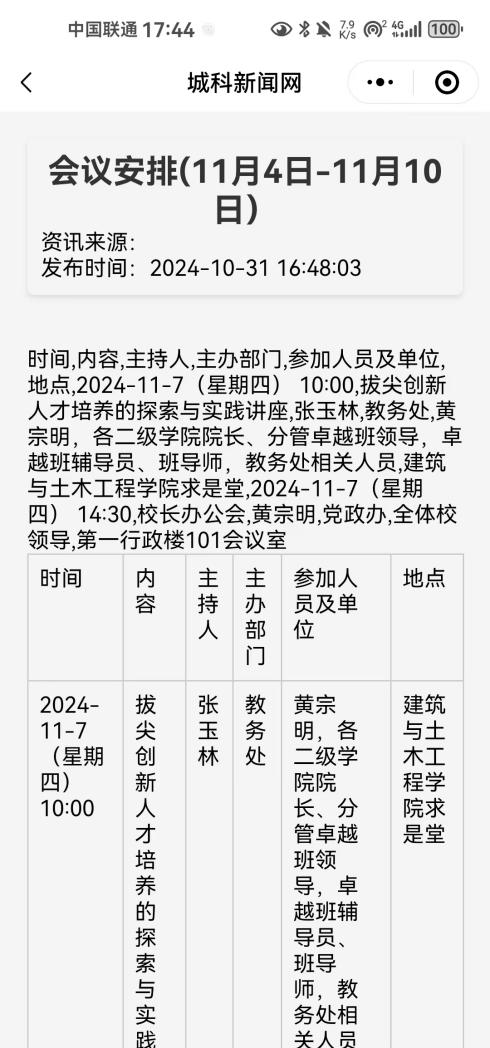
版本编号： V1.0

# 一 项目详情及使用方式

## 1.1项目详情： 项目名称：城科微信新闻小程序 项目简介：本项目是一个为本校学生和教师打造的新闻通知平台，帮助用户实时接收学校通知和新闻。 项目背景：为了使本校学生和教师接收学校新闻和通知更加便捷高效，避免了只能访问官网才能获取通知的痛楚，同时在特殊情况下，也缓解了学校官网的访问压力。

## 2.1使用方式 项目目前处于体验版本未进行备案上线，可通过扫描下方体验二维码进行申请访问： 项目内容： 以下展示项目的各个界面： 主界面（通知界面）： 新闻动态： 4fe34e2796506b6a001f91e520959a0 b7302665591ee3df9b9d132902ba032 个人主页： 新闻动态搜索后的界面： a8b8def253f78164814d78620a78364 25e7501a34b50e9a5ce2b34573e924d

详细内容展示：



二 项目的亮点

## 2.1 独立的后端框架 后端数据采集：本项目没有使用微信小程序云开发中的云数据库，项目展示的所有数据来源均来自于服务器数据库，服务器数据库的数据则是通过数据采集脚本爬取的，服务器端的数据采集脚本则是定时运行；在小程序端，利用js中的request请求模块，通过访问指定的服务器安全域名网址接收到数据，并渲染到页面上。

## 2.3 高效且完善的数据传输模式



上图的代码就是用于获取校园动态数据的部分代码，我们可以用游览器访问这个链接看看：



这是其中部分数据，可以看到，访问该链接后返回的数

据列表嵌套字典的的格数，这有利于我们在小程序端对数据进行遍历渲染处理。

三 项目的核心技术  
3.1 后端（云服务器端）

云服务器操作系统选择了 Ubuntu，主要基于以下优势：

开源免费：Ubuntu作为开源系统，免去了授权费用，尤其适合成本敏感的项目开发。

易于使用：Ubuntu系统界面简洁直观，命令操作便捷，文档资源丰富，且社区支持广泛，便于新手快速上手。

高安全性：Ubuntu定期提供安全更新和漏洞修复，安全性与稳定性广受认可。

软件包支持丰富：Ubuntu的APT包管理系统支持大量软件包的快速安装和管理，为应用开发和部署提供了极大便利。

广泛的兼容性：Ubuntu在云服务市场上拥有较高的支持度，与AWS、Azure等主流云平台的兼容性良好，便于服务迁移和扩展。

服务器代理选用了 Nginx。作为高性能的HTTP和反向代理服务器，Nginx以非阻塞的异步架构而闻名，能够高效处理大量并发请求，非常适合用于动态Web应用和API网关等需要高并发的场景。Ubuntu与Nginx的组合在云服务器上能有效平衡性能、灵活性和成本。

3.2 如何实现数据请求接口  
 在小程序的请求模块中，如果要实现客户端对服务端的 request 请求并支持手机端调试，则需要配置一个合法的 request 请求域名。整个过程可以分为以下几个步骤：

1. 申请域名并进行备案

首先，去域名注册商处申请一个域名（例如 example.com）。如果你计划在中国境内的服务器上部署该服务，域名需要进行ICP备案。备案完成后，域名才可以在国内通过合法的请求配置在微信小程序中使用。

2. 安装 SSL 证书

为了使小程序可以使用 HTTPS 请求并满足微信小程序对合法域名的要求，你需要在服务器端安装 SSL/TLS 证书，使域名支持 HTTPS 访问。

证书可以通过\*\*可信的证书颁发机构（如 Let’s Encrypt、DigiCert）\*\*获取，也可以选择一些免费的证书服务。获取证书后，将证书文件和密钥文件上传到服务器，并进行安装。

3. 配置 Nginx

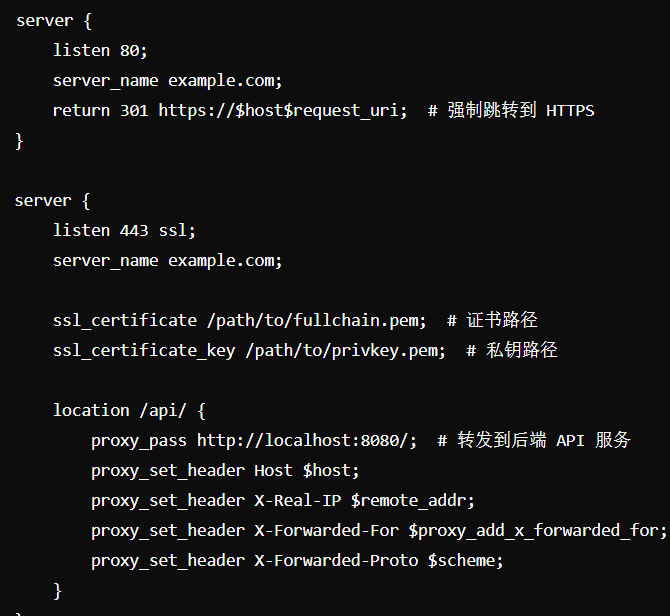
配置Nginx以使服务器能够处理域名请求。编辑Nginx 的配置文件（例如 /etc/nginx/sites-available/default 或 /etc/nginx/nginx.conf），并按以下步骤进行设置：

绑定域名和公网 IP：在配置文件中绑定你的域名（example.com）和服务器的公网 IP，确保所有访问该域名的请求都指向这台服务器。

设置HTTPS和HTTP重定向：强制所有HTTP请求自动跳转至HTTPS，确保数据传输安全。

端口转发与反向代理：根据实际应用的端口，将请求的 80（HTTP） 或 443（HTTPS） 端口转发至后端脚本的端口（例如 8080），使客户端请求能够正确转发到服务器端的 API。

具体的 Nginx 配置示例：

4. 配置微信小程序中的 request 合法域名

登录微信小程序管理后台，进入 “开发管理” -> “开发设置” 页面。

在“服务器域名”配置中，添加你的 HTTPS 域名（如 https://example.com）为合法的 request 域名。

注意，微信小程序要求域名必须以 HTTPS 开头，且需要通过 ICP 备案（中国境内服务器）并具备有效的 SSL 证书。

5. 请求数据与自定义 URL 地址

配置完成后，在小程序的脚本中可以通过 wx.request 调用服务器端API，例如调用 https://example.com/api/your\_custom\_path 以获取数据。

自定义 URL 地址可以方便地组织不同的 API 接口，返回指定的数据。通过在 Nginx 中的 location 配置，用户的自定义 URL 请求可以灵活地映射到相应的后端脚本，确保数据能够顺利返回。

## 3.3 数据采集和传输框架

在本项目中，我选择了 Python 作为编程语言，Flask 作为框架。Python 以其简洁易学、功能强大的特性，成为了许多开发者的首选语言。得益于 Python 丰富的第三方库和高效的模块化设计，开发者可以快速集成现成的功能，避免从头编写底层代码，这在提高开发效率的同时，也确保了代码的稳定性和可维护性。

具体而言，本项目的数据源为我校官网上的通知和新闻动态，包括标题、内容等信息。我利用 Python 中的 requests 模块，直接请求目标网页，获取所需的 HTML 数据。接下来，通过解析网页内容并进行必要的数据清洗与筛选，我能够提取出有价值的信息。最终，将处理过的数据以合适的格式返回并保存到我服务器端的数据库中，这样在其他客户端就可以通过指定的url进行获取指定数据了。  
以下是框架的部分代码：

## 数据采集

## 数据处理函数部分代码 数据库查询申请

## 3.4 小程序界面端数据渲染

在小程序界面端的数据渲染流程中，首先通过JavaScript中的wx.request接口向服务器发送请求，获取所需的数据。接收到数据后，可以在逻辑层进行初步的处理与筛选，如数据格式化、去重、分页等，以便更好地满足页面显示的需求。随后，将处理后的数据传递给wxml模板，利用wxml模块进行数据的遍历和动态渲染，使界面实时展示最新数据。

在样式方面，可以使用wxss模块（微信小程序的CSS）对渲染数据进行精细化的样式调整，优化视觉效果和用户体验。例如，通过合理的间距、对齐方式、字体颜色等设置，使信息展示更清晰、界面更整洁。同时，可以引入响应式设计和自适应布局，以适应不同尺寸的设备屏幕，为用户提供更舒适的交互体验。

为了进一步提升性能，使用setData来减少数据更新次数，并优化复杂数据的渲染逻辑。此外，还结合了懒加载、虚拟列表等技术手段，优化数据加载和显示的性能，避免大量数据一次性渲染造成的卡顿或加载延迟。

以下是部分功能代码：  
