项目名称：希腊神祇探索记

1. 项目概述

“希腊神祇探索记”是一款创新的文化探索平台，旨在让用户通过多种渠道和信息源来探索希腊神话中的角色及其文化背景。该平台集成了与希腊神话角色相关的多种类型的内容，包括角色详细信息、现代解读、相关书籍、音乐及视频内容，为用户提供全方位的探索体验。项目设计目标是通过互动式的探索工具，为用户构建一个既有趣又富有教育意义的神话知识库。

1. 系统需求
   1. 环境要求

Node.js：v20.17.0

Npm：10.8.2

浏览器：Chrome

* 1. 项目依赖

React：前端框架

jQuery：用于处理Ajax请求

2.3. 补充说明

由于api选择的问题，需要挂vpn才能正常使用。

1. 功能模块

3.1. 搜索希腊神话角色

用户通过简单的搜索功能输入希腊神话角色的名称（中文），系统将自动检索并呈现有关该角色的多方面信息。该功能是整个平台的核心入口。

3.2. 人物简介

平台将展示希腊神话角色的详细背景信息，这部分内容为用户提供基础的角色理解。

3.3. 相关讨论

平台会为用户展示与该希腊神话角色相关的现代解读。

3.4. 相关书籍

用户可以查阅与希腊神话角色相关的书籍推荐。

3.6. 相关音乐

用户还可以通过平台找到与希腊神话角色相关的艺术家、歌曲或专辑，探索角色在音乐作品中的影响。

3.7. 相关视频

平台聚合了与该角色相关的讲解、分析视频内容。无论是学术讨论、影视解读还是动画讲述，用户可以观看这些内容，获得不同视角的分析和探讨。

1. 目标用户

4.1. 古典文化与历史爱好者

这类用户对古代希腊文化、神话故事及其背后的历史充满浓厚兴趣。他们希望通过探索不同角色的背景故事，进一步了解希腊神话在文化、宗教、历史中的重要性。

4.2. 学生与学术研究者

希腊神话是许多学术研究的基础，尤其是在历史、文学等学科中占有重要地位。这类用户可能是大学生、研究生或学者，他们需要可靠的信息来帮助他们进行学术研究。

4.3. 教师与教育工作者

教师和教育工作者可以利用该平台为他们的课程和教育内容提供丰富的背景信息。

4.4. 艺术家、作家与创意工作者

希腊神话中的角色和故事常常成为艺术、文学、音乐和电影等领域的灵感来源。

4.5. 影视、游戏和娱乐产业从业者

希腊神话是许多影视作品和电子游戏的背景来源。对于这些产业中的编剧、导演和游戏设计师来说，平台能够提供神话角色的详细信息、相关书籍、视频内容等资源，以帮助他们为作品设定更丰富的背景，创造更真实的故事世界。

1. 实现步骤

5.1. 项目配置

5.1.1 创建React应用

npx create-react-app my-app

在名为my-app的文件夹中创建一个新的React应用，自动安装所有必要的依赖项。

5.1.2 安装jQuery依赖

npm install jquery

在项目中安装jQuery并使用Ajax

5.2. 具体实现

5.2.1 初始页面

1）界面设计

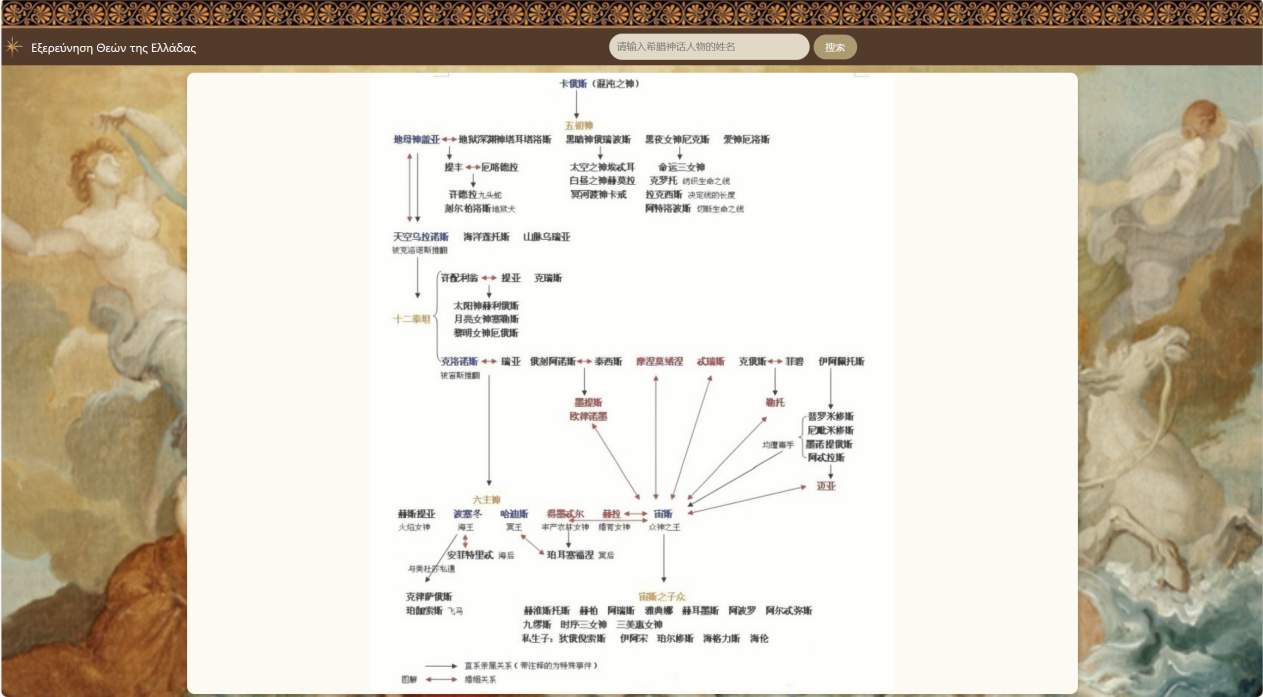
界面包含搜索框以及希腊神话人物关系图。用户可以根据希腊神话人物关系图上自身感兴趣的角色进行搜索。

1. 界面实现

SearchBox.jsx 用于定义搜索框的样式

App.js 判定用户是否已经搜索角色。若无，展示初始页面。若是，展示具体页面。

1. 界面展示



5.2.2 人物简介

1）界面设计

界面包含顶栏和具体人物信息，以及跳转链接。

2）界面实现

SearchBox.jsx 用于定义搜索框的样式

App.js 判定用户是否已经搜索角色。若无，展示初始页面。若是，展示具体页面。

Banner.jsx 用于定义顶栏样式

MediaWikiSearch.jsx 用于调用MediaWiki API

3）界面展示



5.2.3 相关视频

1）界面设计

界面包含顶栏和相关视频，以及跳转链接。

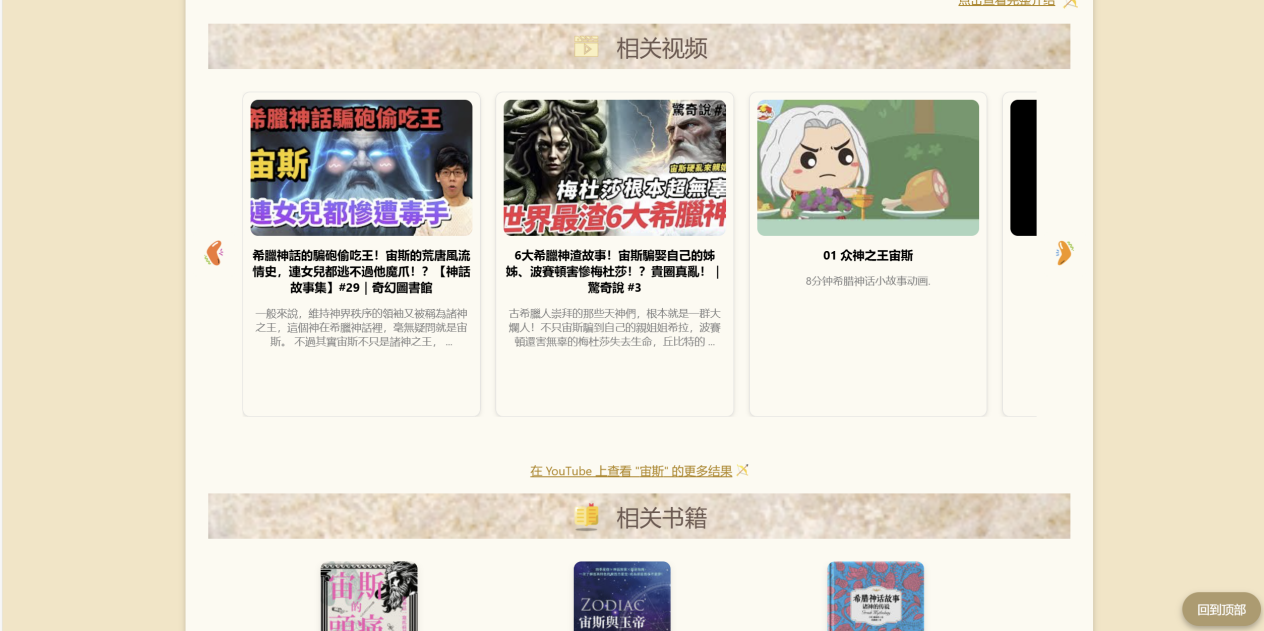
2）界面实现

App.js 判定用户是否已经搜索角色。若无，展示初始页面。若是，展示具体页面。

Banner.jsx 用于定义顶栏样式

YoutubeVideos.jsx 用于调用YouTube API

3）界面展示



5.2.4 相关书籍

1）界面设计

界面包含顶栏和相关书籍，以及跳转链接。

2）界面实现

App.js 判定用户是否已经搜索角色。若无，展示初始页面。若是，展示具体页面。

Banner.jsx 用于定义顶栏样式

GoogleBooks.jsx 用于调用Google Books API

3）界面展示



5.2.5 相关音乐

1）界面设计

界面包含顶栏和相关音乐（相关歌曲、相关专辑、相关艺术家），以及跳转链接。

2）界面实现

App.js 判定用户是否已经搜索角色。若无，展示初始页面。若是，展示具体页面。

Banner.jsx 用于定义顶栏样式

SpotifyMusic.jsx 用于调用Spotify API

3）界面展示





5.2.6 相关讨论

1）界面设计

界面包含顶栏和相关讨论，以及跳转链接。

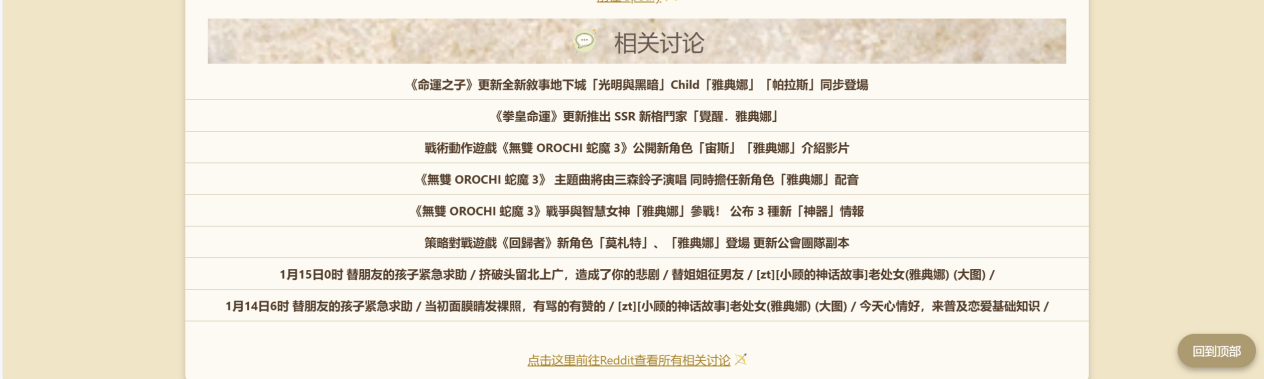
2）界面实现

App.js 判定用户是否已经搜索角色。若无，展示初始页面。若是，展示具体页面。

Banner.jsx 用于定义顶栏样式

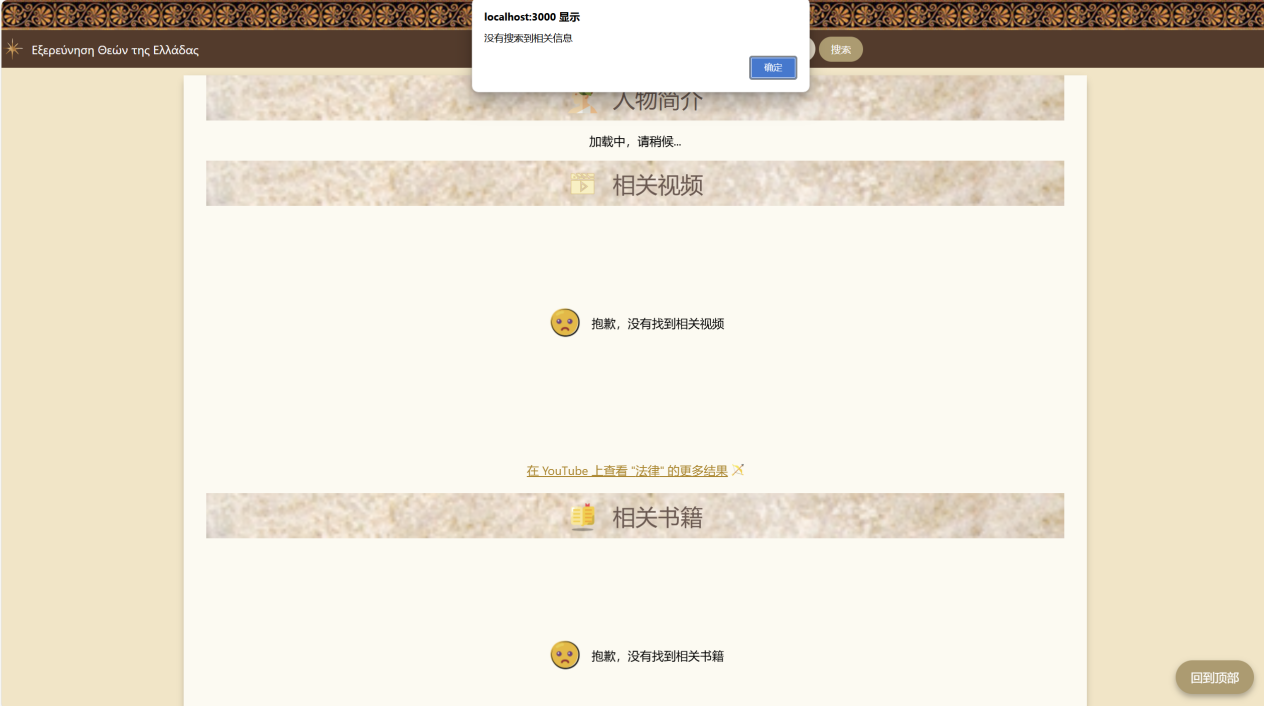
Reddit.jsx 用于调用Reddit API

3）界面展示



5.2.6 没有结果





5.3. 项目部署

5.1. 构建生产版本

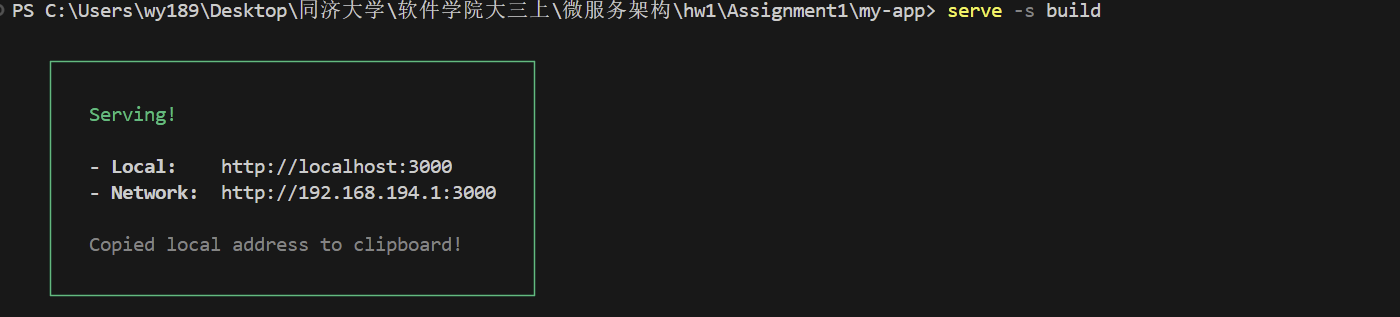
npm run build

5.2. 使用本地服务器托管生产版本

npm install -g serve

serve -s build

5.3. 访问应用



1. 未来工作

6.1. 部署到云平台

研究将应用部署到云平台（如Netlify）以提高可访问性和扩展性。

6.2. 功能扩展

根据用户反馈或需求，添加新功能或改进现有功能。

1. 解决方案中使用的Web API

MediaWiki API <https://www.mediawiki.org/wiki/API:Tutorial>

YouTube API <https://developers.google.com/youtube/v3/docs/search/list?hl=zh-cn>

Google Books API <https://developers.google.com/books/docs/v1/libraries?hl=zh-cn>

Spotify API <https://developer.spotify.com/documentation/web-api/concepts/apps>

Reddit API <https://developers.reddit.com/docs/api>