

项目申请书

项目名称：改进前端渲染框架

项目 ID： 2019092

项目详细方案：

1.完善项目中 JavaScript 向 TypeScript 的迁移

被称为 JS 的超集的 TS，具有类型检查和更强的类型系统特性，这些特性使得 TS 与 JS 相比会更精简的代码和更低的出错率来适应大型应用的开发。此外，参数和返回值的类型定义，实现了代码即文档的功能，更有利于后期的维护。

2.添加 Blog/Changelog 功能

changelog 关于对接 github 中的 api 已经实现了站外链接跳转，但考虑到提升用户体验感，将获取 github commit 中的我们所需数据，为之生成符合 OI-Wiki 风格的页面。Blog 功能的实现，将以 wiki 中 md 文档相同的方式，实现展示。其思路可以是（仓库中借助的是 MDX，后续会看 MDX 的实现思路）：借助 gatsby-source-filesystem 访问 md 文件，使用 gatsby-transformer-remark 将 md 转换为 HTML。编写 template 文件，使用 graphql 查询我们所需要的 HTML 等数据，编写函数向我们已获取的 HTML 数据，增加额外的 HTML 代码，完善 post 页面。在 gatsby-node.js 中使用 createPages API 来生成静态页面。在 createPages 函数内，将 graph 查询结果，逐一 map，项目在 build 期间即可为 md 文件动态生成静态可访问页面。

3.实现国际化功能，提供中英双语支持。

通过 gatsby-plugin-i18n 和 Kontent source plugin 的配合使用可以定位用户位置，生成定制化语言静态页面。在 github 中有一个同样基于 gatsby 的开源 [demo](#) 通过这种方式较好的实现了国际化支持。大致实现思路是：通过 gatsby-plugin-i18n 识别用户所在地区所使用的语言。借助 Kontent source plugin，GraphQL 查询时增加 preferred_language 字段，使用 createPage 方法将生成的定制化语言页面放入同级文件夹内。提供一个 UI 按钮，当用户点击切换语言时 Link 到相应页面。

4.对接后端服务，添加代码运行功能。

在 gatsby 中，run time 和 build time 中 fetch data 的方式具有较大差异，build time 期间，使用 GraphQL 进行数据的获取。代码运行功能的实现，需要在 run time 期间动态获取。可以使用 react hooks 中的 useState 和 useEffect。定义一个 state 变量，将原有节点内容存入作为初始值，用户点击 run 后，fetch 后端数据后，set 存储相应的值，当 state 发生变化后，react 会对页面进行更新渲染，即可展示代码运行结果。

5.对用户提交的推荐链接根据 vote 高低的顺序进行展示

由于 blog 篇幅有限，如果能在 blog 末尾增加相关阅读链接，则可以更加便利用户。所以将提供一个推荐链接列表，用户可对其 vote，项目列表的顺序根据 vote 数量高低排序。其实现思路与第四点大致相同。对于链接列表的展示通过 react 的 fetch 方法对接后端，当用户投票后，向后端发送数据，更新链接的投票数。

项目开发时间计划:

第一周(6.30-7.7)	1.和导师进行相关事宜(blog 和 vote) 的沟通。 2.做相关练习, 学会使用 GraphQL 和 Gatsby 编程。 3.初步熟悉仓库代码, 弄清代码结构与逻辑。
第二周(7.8-7.15)	做 ts 练习, 学会使用 ts 来完善项目中 js 向 ts 的迁移
第三周(7.16-7.23)	1.学习 github 中基于 gatsby 的 demo 中的 i18n 的实现 2.通过 gatsby-plugin-i18n 完成中英双语支持
第四周(7.24-7.31)	1.查找获取 github commit 数据并引入到页面的方法 2.随着从原网站的众多 md 文件的迁移完成, 学会实现对剩余空白页面的补全, 建立一个初步完整的 Gatsby-OI-Wiki
第五周(8.1-8.8)	1.实现站内展示 changelog 功能 2.初步实现 blog 功能, 并进行测试
第六周(8.9-8.16)	1.完善 blog 功能 2.撰写中期项目报告
第七周(8.17-8.24)	对接后端, 添加代码运行功能
第八周(8.25-9.1)	1.与后端进行所需接口讨论 2.向页面中加入推荐列表, 以及用户 vote 功能
第九周(9.2-9.8)	对功能点进行测试, 解决遗留/隐藏 bug 和项目中的 issue
第十周(9.9-9.16)	对功能点进行测试, 解决遗留/隐藏 bug 和项目中的 issue
第十一周(9.17-9.24)	产出项目技术总结 blog
第十二周(9.24-9.30)	撰写终期项目报告