中期报告

1. 前期进展情况（按阶段完成的主要任务、取得的阶段成果和存在的困难与不足）

在过去的两个月中，我会时不时思考个人博客系统有哪些可以优化的点，一想到某个功能，我便会将其加入到canny这个工具中去，做好相关的项目管理工作。[我的canny站点。](https://simple.canny.io/)

但是后来我还是弃用了canny，选择了github project来管理我的毕设项目。[我的毕设项目管理。](https://github.com/users/real-jiakai/projects/2/)因为github project可以直接对接到相应的仓库，这样子方便我将自己的灵感转换为github isssue，跟进我的开发。

开题汇报的时候，我的决定是采用Hugo框架来搭建个人博客系统。但是在我亲自实践的时候，我发现使用Hugo来实现毕设，工作量太小了，因为博客该有的功能Hugo框架的作者早就已经将其嵌入到框架内部，你仅需要调整相应的模板，就能实现一个简单的个人博客系统。

随着技术的发展，越来越多的第三方工具可以集成进静态博客系统，让静态博客功能堪比动态博客。比如评论区可以采用waline内嵌入静态博客系统，搜索功能可以使用algolia来实现。

既然如此，貌似毕设可以通过很多成熟的第三方来实现。但是我并不希望自己的毕设最终束缚于第三方工具。我想从零到一实现个人博客系统。

我经过了多方考虑，最终决定使用Next.js作为我的开发框架。其中，我选择Next.js的原因主要有以下三点：

首先，Next.js的生态系统成熟，可以媲美于后端框架如Django、Springboot。这意味着我可以轻松地找到相关的资料和工具，并且可以快速解决问题，加速开发周期。

其次，Next.js具有SSG（静态网站生成）功能，类似于Hugo等静态网站生成器。这意味着我可以在构建时生成静态HTML文件，从而使网站加载速度更快，同时还可以保留搜索引擎优化和动态交互性。

最后，Next.js具有动态交互能力，能够实现一些常见的交互需求，包括自定义页面布局、动态路由、API路由、代码拆分和CSS模块化等。这使得我可以轻松构建具有动态效果的Web应用程序，并为我的用户提供更好的体验。

综上所述，我选择使用Next.js作为我的开发框架，因为它具有成熟的生态系统、SSG功能和动态交互能力，这些特性使得我的博客系统更加高效、灵活和易于维护。

1. 下一阶段工作计划

优先实现MVP（Minimum Viable Product，最小可行产品），预计3月初部署上线。接着分阶段实现自己项目管理中罗列出来的功能，使用Git管理自己开发的个人博客系统，并同步到GitHub。