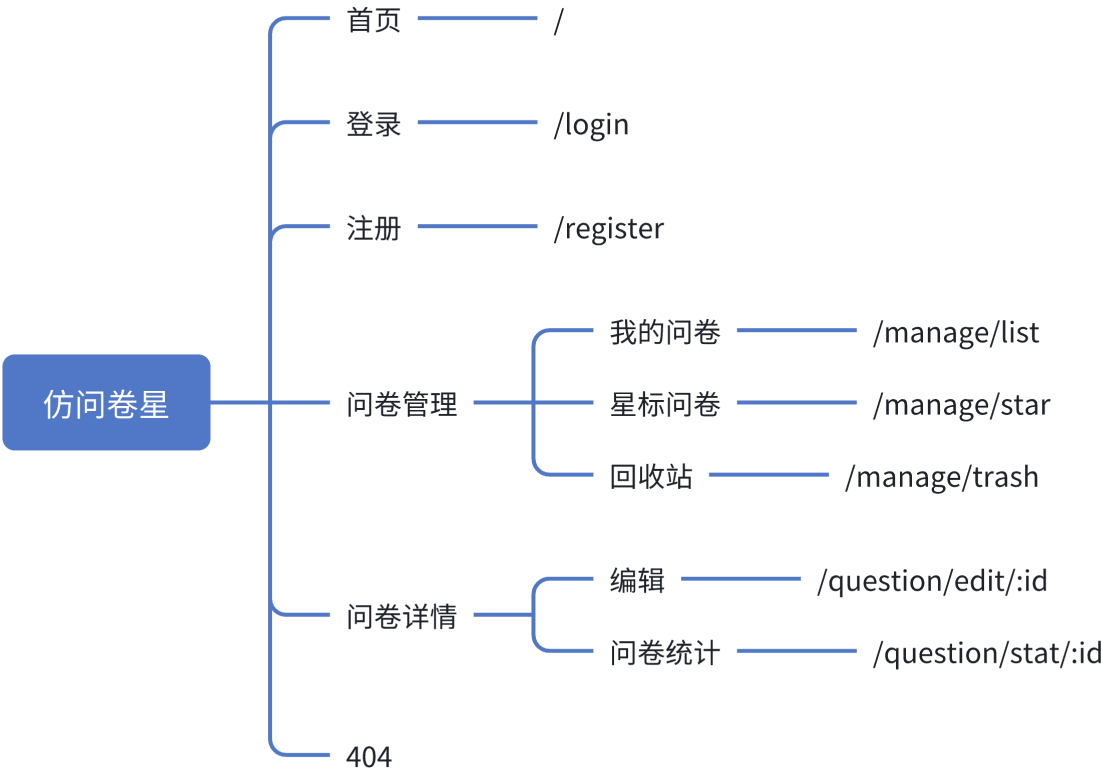
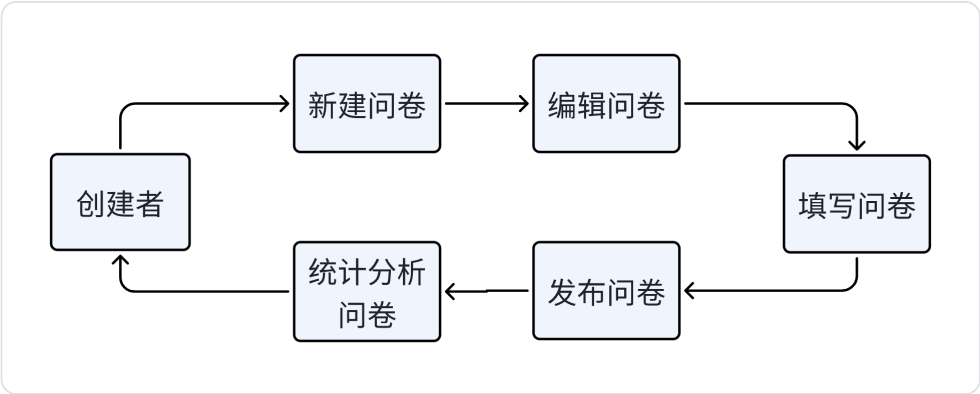


07 React Router

路由设计



业务闭环（有输入有输出）



Layout 模板

- **MainLayout** 包含大部分页面都需要的 header 和 footer（除了编辑、统计）
- **ManageLayout** 问卷管理页面所需要的左侧导航菜单，嵌套在 MainLayout 之下
- **QuestionLayout** 用于问卷编辑和统计

开发流程

1. 按照预定设计，**新建若干页面**，即 React 组件
2. 为每个页面**分配对应的路由**
3. 为每个页面**分配相应的 Layout 模板**

设置标题和 icon

- 设置标题
 - 在 index.html 中设置 title 标签的内容
 - 在组件中使用 ahooks 的 `useTitle`
- 设置 icon：在 index.htm 通过 meta 标签设置

react-router-dom 核心 API

通过 `npm install react-router-dom --save` 安装。

- `createBrowserRouter` 根据路由规则创建路由器
- `RouteObject` 路由规则类型
- `RouterProvider` 使用路由器，此时根据页面 URL 渲染对应的路由组件，一般在 App 中使用
- `lazy`, `Suspense` 路由的懒加载
- `useNavigate` 程式化路由导航
- `Link` 实现路由的跳转
- `useParams` 接收 params 类型的参数
- `useSearchParams` 接收 query 类型的参数

关于 Vite 配置别名的注意事项

Step 1. 配置 vite.config.ts。此时，代码中的别名可以被 vite 解析，项目也可以成功启动。

```
1 import { defineConfig } from 'vite';
2 import react from '@vitejs/plugin-react';
3 import { resolve } from 'path';
4
5 // https://vitejs.dev/config/
6 export default defineConfig({
7   plugins: [react()],
8   resolve: {
```

```

9     alias: {
10       '@': resolve(__dirname, './src'),
11       '@components': resolve(__dirname, './src/components'),
12       '@pages': resolve(__dirname, './src/pages'),
13       '@layouts': resolve(__dirname, './src/layouts'),
14     },
15   },
16 });

```

Step 2. 配置 tsconfig.app.json。此时，代码中的别名在 vscode 中才会有提示，可以实现点击跳转，智能提示等功能。

```

1 {
2   "compilerOptions": {
3     // ...
4     /* 设置别名提示 */
5     "baseUrl": ".",
6     "paths": {
7       "@/*": [
8         "src/*"
9       ],
10      "@components/*": [
11        "src/components/*"
12      ],
13      "@pages/*": [
14        "src/pages/*"
15      ],
16      "@layouts/*": [
17        "src/layouts/*"
18      ],
19    },
20  },
21  "include": [
22    "src"
23  ]
24 }

```