1. 正常流程
2. 登录->获取token，将token存储于Cookie中
3. 选择项目->将项目信息存储于url和vuex中
4. 进行页面跳转前通过全局路由守卫，使用token和url中的项目信息访问后端，获取完整的可访问路由列表，进行权限控制。
5. 其他情况

1.用户不经登录页面直接访问路径

1.1 无toekn

直接跳转登录页面

1.2有token

1.2.1访问包含projectID的路径：请求两次权限

1.2.2访问不包含projectID的路径：请求一次权限

2.刷新页面：等同于情况1.2

3.修改url，但不刷新页面

3.1修改url中的projectID：添加函数判断项目ID是否修改，若修改，则重新发起两次权限请求（同1.2.1）

3.2修改url中非projectID的部分：同1.2

4.用户访问不存在的路径：跳转至404页面

5. 用户修改token

5.1.用户在登录前修改token会直接报错跳转到首页，

5.2.用户在登录之后删除token会在下一次页面跳转时直接跳转到首页

5.3.用户在登录之后修改token但不删除，则会保留当前用户权限，直到下一次刷新页面（考虑在api响应拦截器处添加判断token是否有效的函数（后端返回token是否有效的函数），若token无效或过期则显示报错信息并直接跳转到首页）（待讨论）

6. 后台用户权限变更：未考虑此种情况，只有当用户重新刷新页面或选择新项目时才会更新权限。（待讨论）

解决方案1:使用Mqtt实时同步用户权限

解决方案2：同5.3

1. 关于cookie
2. 存储内容：用户token
3. 注意：为实现单个用户在不同的窗口访问多个项目，projectID存储于url中
4. 用户点击注销后会清除所有的token（也可以不清除token，使用户下次登录时可以直接访问页面），但用户直接关闭浏览器则会保留token。
5. Token时效性：未考虑
6. 待定事项
7. 用户注销后是返回登录页面还是首页展示页面
8. 路由的push和replace设置
   1. 是否可以通过浏览器回退键返回login页面
   2. 是否可以通过浏览器回退键返回上一个选中的项目页面
9. 使用Cookie、localstorage还是session来存储token和其他信息