**前端开发准备**

1.开发框架文档：<https://panjiachen.gitee.io/vue-element-admin-site/zh/>

2.开发框架在线访问地址：[https://panjiachen.gitee.io/vue-element-admin/#/login](https://panjiachen.gitee.io/vue-element-admin/" \l "/login)

3.开发工具统一：Visual Studio Code <https://code.visualstudio.com/>

4.UI组件库：[https://element.eleme.cn/#/zh-CN](https://element.eleme.cn/" \l "/zh-CN)

5.参考实例： <https://panjiachen.gitee.io/vue-element-admin/#/table/complex-table> （完整增删改查及api调用）

6.图标库：[iconfont+](https://www.iconfont.cn/) (图标更新文件提取 @李少群 负责)

7.代码管理：git（vscode可安装插件 Gitlens、Git History 方便管理）

8.接口调试工具：[postMan](https://www.getpostman.com/)（接口调试时如有问题，先用postMan校验是否后端接口问题）

**项目git代码仓库**

地址：http://101.37.255.172:3000/jsonyy/silktonevue.git（帐号密码联系 @谢绵宝 添加）

注意：提交代码时如遇pre-commit...相关报错，可先把.git(隐藏文件)\hooks\pre-commin文件删除

**产品相关**

1.产品相关负责人：洪杨杨、柯圣杰、李少峰

2.产品原型：[https://next.modao.cc/app/b869264fe7add410b3e126e9b63d2843?#screen=s94657962E81567042128391](https://next.modao.cc/app/b869264fe7add410b3e126e9b63d2843?" \l "screen=s94657962E81567042128391)

3.UI设计稿：[https://lanhuapp.com/web/#/item/project/board?type=share\_mark&pid=38b1578d-d952-4db7-8e46-9cb115a892c8&activeSectionId=&teamId=47a2c4fa-515a-46d2-90e1-f0c234cb79f9&param=a990516a-a2b2-4543-8d4e-adc1b385192c](https://lanhuapp.com/web/" \l "/item/project/board?type=share_mark&pid=38b1578d-d952-4db7-8e46-9cb115a892c8&activeSectionId=&teamId=47a2c4fa-515a-46d2-90e1-f0c234cb79f9&param=a990516a-a2b2-4543-8d4e-adc1b385192c)

**后端协同**

1.主要负责人：谢绵宝

2.API文档：<http://api.meiyetongsoft.com/index.php?s=/54&page_id=1440>

**项目预览**

1.测试地址： <http://slt.meiyetongsoft.com/dist> （如无登陆账号可联系 @谢绵宝 添加）

2.构建发布：npm run build:prod （打包任务由 @刘洋 负责）

**开发规范统一 ★★★★★**

1.日期时间格式：timestamp 时间戳（单位秒）

2.前端调用后端接口时，字段严格按照所需参数进行传递，非必填项默认空值

3.增删改查模版统一按照标准框架（[src/views/table/complex-table](https://panjiachen.gitee.io/vue-element-admin/#/table/complex-table)）执行，根据业务需求在该基础上进行拓展

4.列表与弹框组件分开定义在不同的.vue文件中，尽量不要在一个组件中处理过多的逻辑

5.表单验证：各自表单各自校验，公共校验规则统一封装在 src/utils/validate.js 中

6.接口管理：统一存放在 src/api 下，按模块进行划分

7.命名规则：Component -> 大写开头驼峰命名法、JS文件/路由页面 -> 横线连接（kebab-case）

8.样式问题：每个页面的样式写在当前views下面，并加上scoped或者命名空间，避免造成全局的样式污染

9.代码提交：提交之前需对代码进行格式化（vscode右键“格式化文档”）

10.请求响应异常情况统一在 src/utils/request.js 响应拦截器中处理，特殊异常在catch中接收处理（后端需提供所有全局error\_code及对应解释方便前端异常情况统一处理，该部分由 @李少群 负责封装）

【2020/01/11补充】

**样式问题补充**：每个页面的最大的dom元素要加一个类名（**建议在类名末尾加上自己名字的缩写以确保类名唯一性，例：” pay-dialog-ly ”**），样式编写推荐用scss，在最外层写上最外层dom的类名，避免造成全局样式污染（**在带有scope的style标签内也要写上最外层的dom元素类名，否则也会造成样式污染**）