# 开发笔记Day1

## 技术选取

### 后端需要技术：

SpringBoot redis缓存

SpringCloud maven项目管理

SpringSecurity easyExcel处理excel文件

MybatisPlus jwt

oAuth2

### 前端需要技术：

Vue+Element-ui+axios+node.js..

### 其他技术

阿里巴巴oss 微信登陆和支付

阿里巴巴视频点播服务 docker Jenkins

阿里巴巴短信服务 git

### 第一天demo开发

**Mybatis-Plus**本质就是mybatis，是增强版，没有做改变。

使用：

1. 创建数据库，建表并且插入数据，用于mybatis-plus操作。
2. 创建springboot项目
3. 引入相关依赖(springboot, mybatis-plus, mysql, lombok)，并且安装插件Lombok
4. 配置application.properties文件(resource文件夹下 )的mysql
5. 写代码：完成模型层（实体层）和mapper层

实体层中给类加上**@Data**，相当于帮助实现了get和set方法

Mapper层在没有写下层mapper.xml文件时可以继承通用类**BaseMapper<Entity>**，实现了普通的增删改查，但是无法自定义SQL，所以少用。

写完之后需要回到启动类，加上**@MapperScan(“ 对应的包”)**，扫描包下的mapper接口，并且自动查询其实现类。

1. 简单测试:

@SpringBootTest

class WebprojectApplicationTests {

//自动装配，把mapper注入到当前类来使用,但是它是个interface，没有具体实现，会报"找不到UserMapper对象"，但是可以运行。

//如果要，如何解决？ 在UserMapper上加注解，申明对象，可以用component，service或者repository

@Autowired

private UserMapper userMapper;

//查询User表的所有数据

@Test

void findAll() {

List<User> users=userMapper.selectList(null);

System.*out*.println(users);

}

@Test

void test(){

User user = new User();

user.setAge(30);

user.setEmail("");

user.setName("K");

//如果不写id，则mybatis\_plus会随机生成id

int insert = userMapper.insert(user);

System.*out*.println(insert);

}

}