# 商城项目

## 一.商城项目简介

### 1.技术特点

技术性;

技术范围广;

分布式;

高并发,负载均衡,高可用,集群

海量数据--🡪大数据,云计算;

业务负责;

系统安全

....

### 2.天天商城项目简介

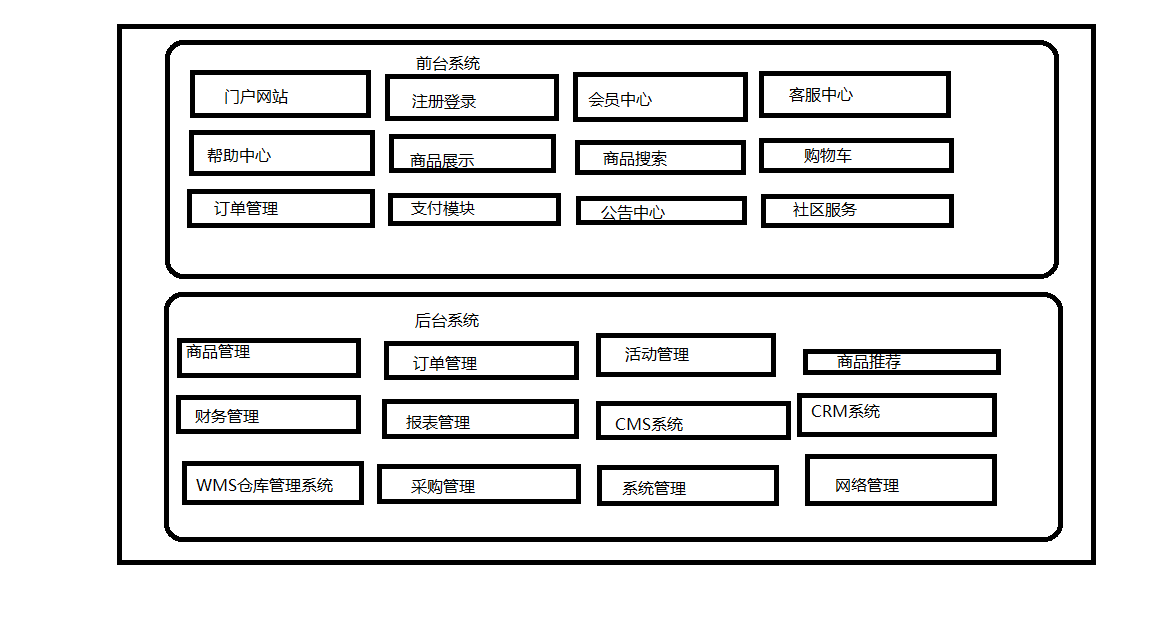
仿淘宝,京东等电商项目的一个B2C平台.

会员可以在该系统中浏览商品,购买商品,下订单;

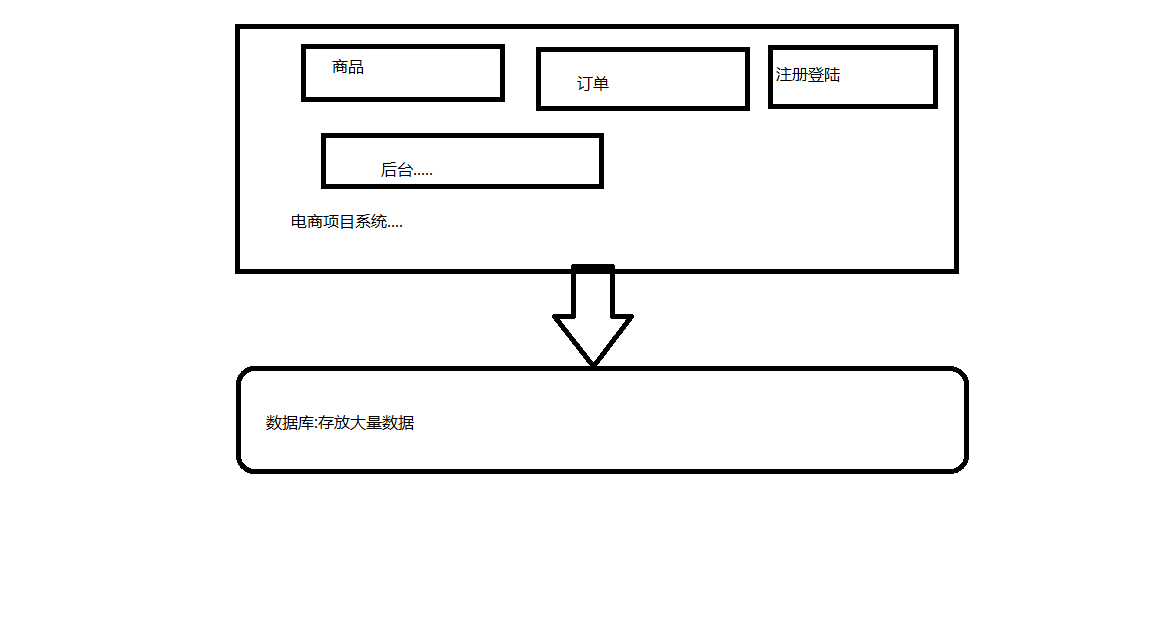
管理员和运维人员可以管理商品,订单,会员等...

客服可以与会员在系统中进行交流...(极光推送)

### 3.项目架构



### 4.架构分析



#### 传统架构缺点分析:

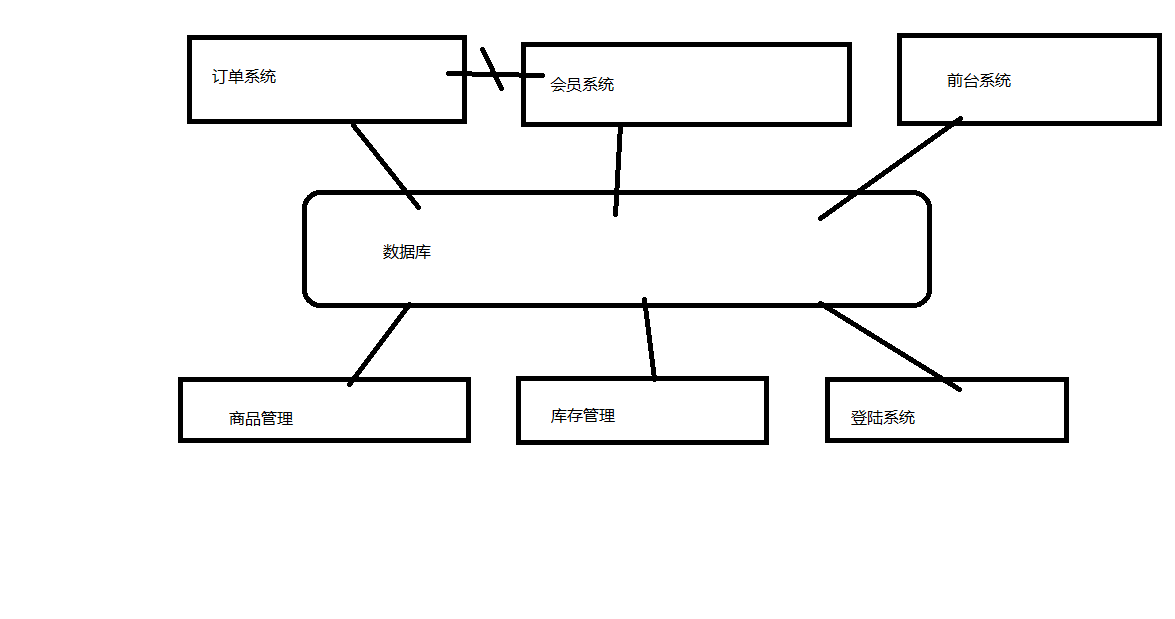
##### .模块之间的耦合度太高,其中一个模块升级,其他模块可能也要跟着升级;

##### .团队之间的合作非常困难,增加了团队开发的难度;

##### .系统的扩展性特别差;

##### .不能灵活的进行分布式部署.

### 5.分布式项目架构



#### 分布式系统架构的优点:

##### .模块之间进行了拆分,降低了模块之间的耦合性,模块之间使用接口通信;

##### .一个大项目拆分成了若干个子项目,不同的子项目由不同的团队来开发,清晰的划分责任,保证项目的进度;

##### .后期要进行系统的扩展,比如要增加功能,执行调用系统提供的接口即可.

##### .可以灵活的进行分布式部署.

##### 缺点:因为模块之间使用接口通信,增加了项目的开发工作量.

### 6.技术选择分析

maven(项目构建;依赖管理)

springmvc+spring+mybatis----SSM

jsp,jstl,css+div,jQuery,jQuery plugin,EasyUI,富文本编辑器(KindEditor);

mysql;

Redis;

Solr(搜索);

jsonp(跨域访问);

httpclient(模块之间远程通信);

Nginx(Linux,实现负载均衡的web服务器);

ActiveMQ,Quartz....

### 7.开发工具和环境

windows,linux

jdk1.8

Eclipse+Maven

SVN;

Tomcat;

Mysql;

Redis;

### 8.项目人员配置情况

产品经理:1-3个人,确定产品需求,给出产品原型.

项目经理(总监):1个,管理团里 TM leader(tl)

UI工程师:1-3个,做图片...

前端团队:3-5个左右,h5根据原型图编写静态页面

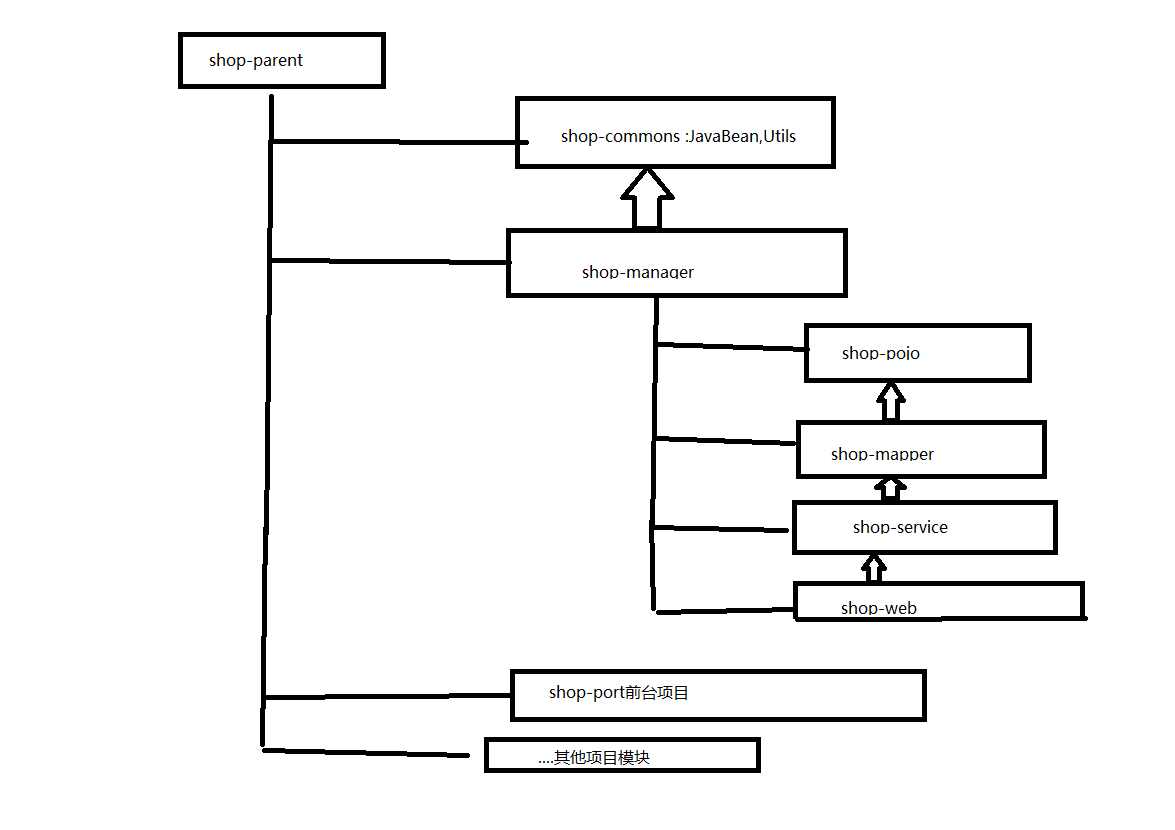
后端团队:5-10个左右,实现项目的所有核心功能.

测试团队:3-5个左右,测试项目中的bug

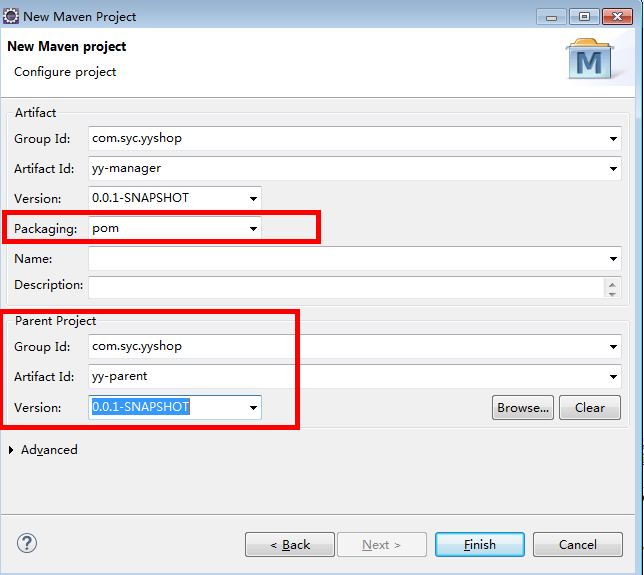
运维团队:3个左右,项目的发布及后期维护

程序员鼓励师....

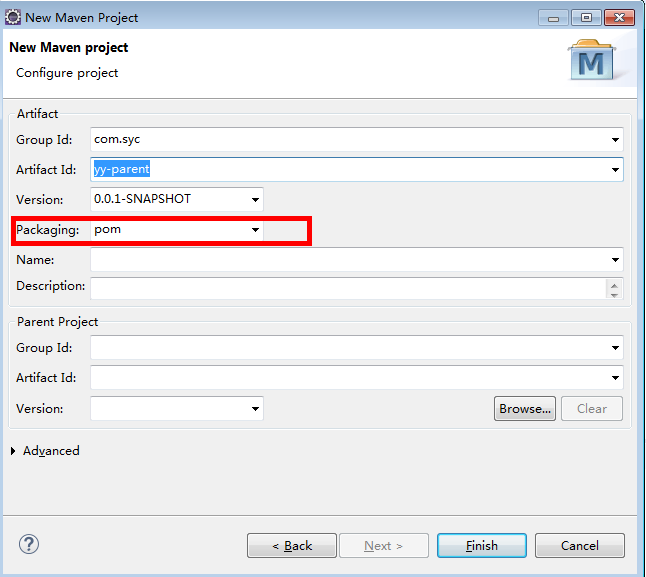
## 二.搭建项目开发环境



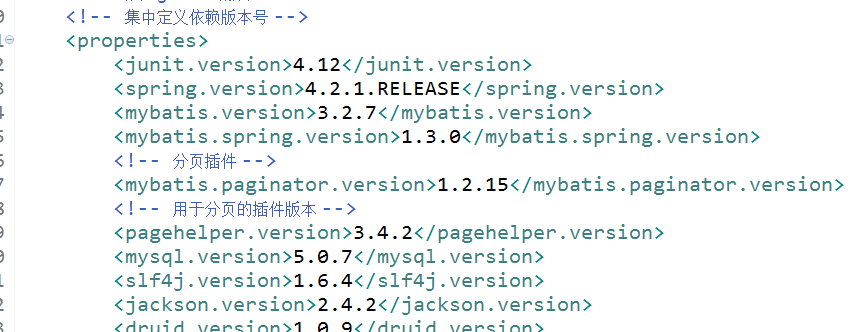
### 7.创建yy-manager项目



### 1.创建yy-parent项目



### 2.在父项目的pom.xml文件中,对依赖的jar和插件进行统一的配置管理



...



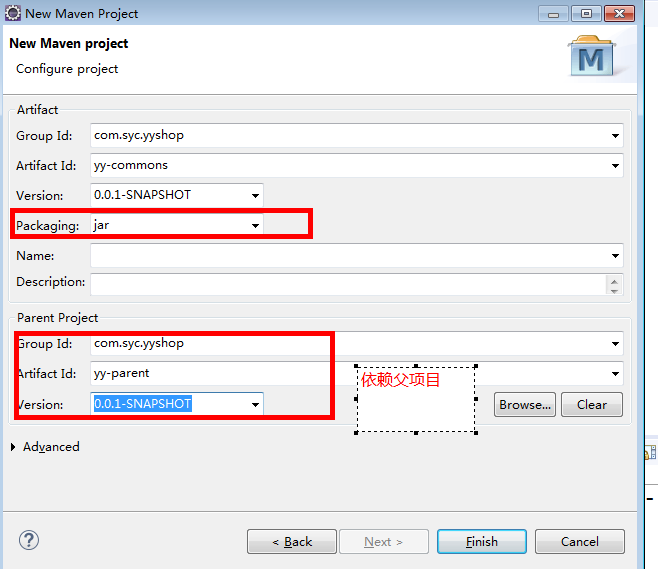
...



### 3.将yy-parent项目按照到本地参考

右键项目----run as---maven----install

### 4.创建yy-commons项目



### 5.修改yy-commons项目的pom.xml文件



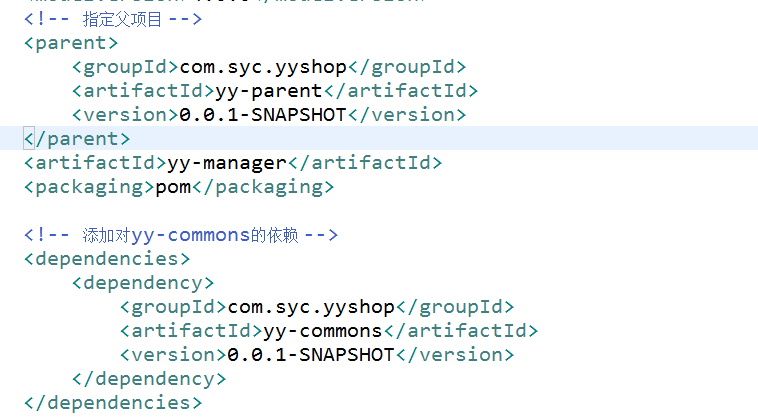
....

### 6.更新和按照yy-commons项目到本地仓库

右键项目----run as---maven----install

### 7.创建yy-manager项目

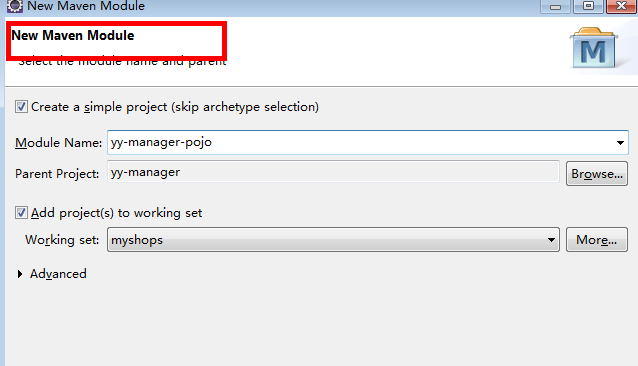
### 8.修改yy-manager项目的pom.xml文件

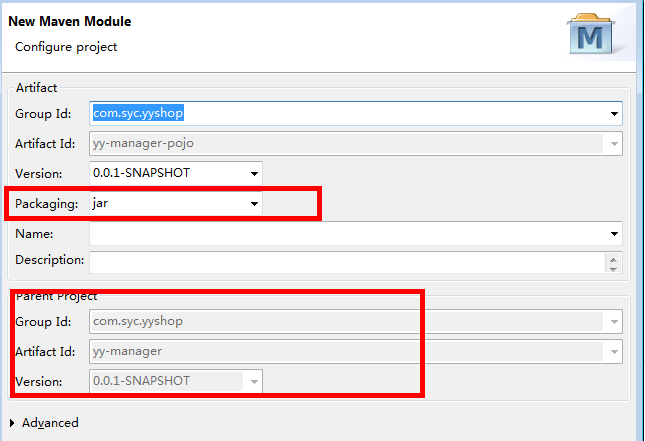




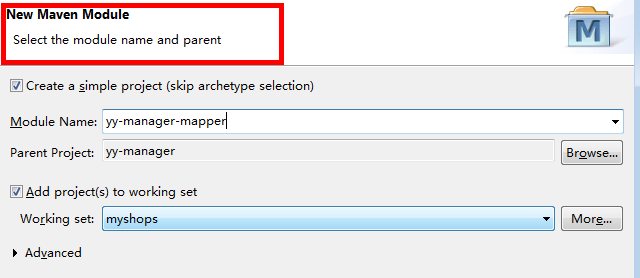
添加对yy-commons项目的依赖,并且配置tomcat插件的端口号和访问路径!

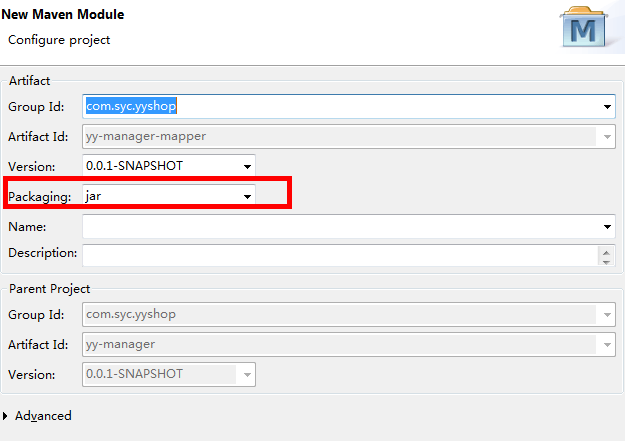
### 9.创建yy-manager项目中的子模块



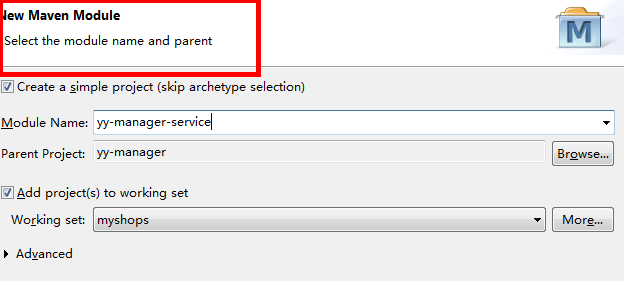


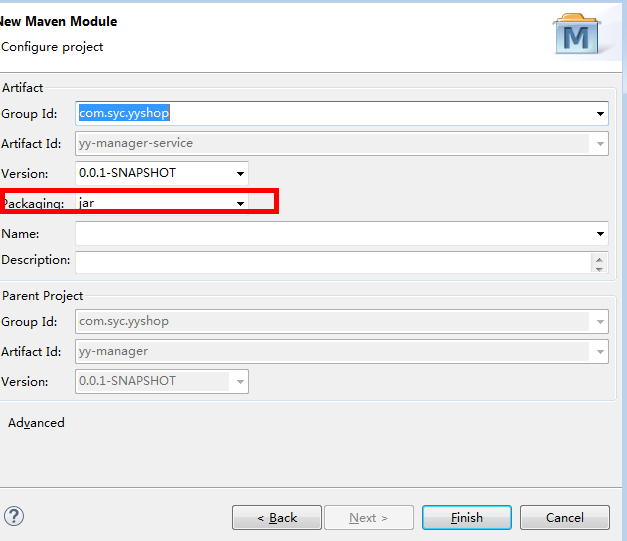
### 10.创建yy-manager里的mapper子模块



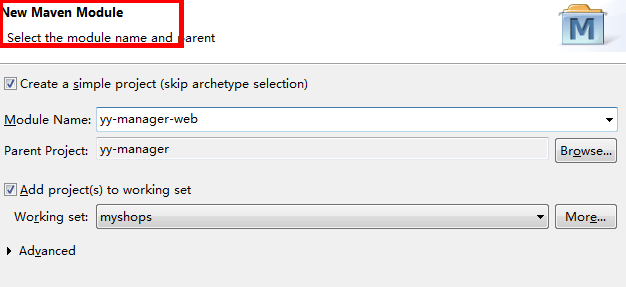


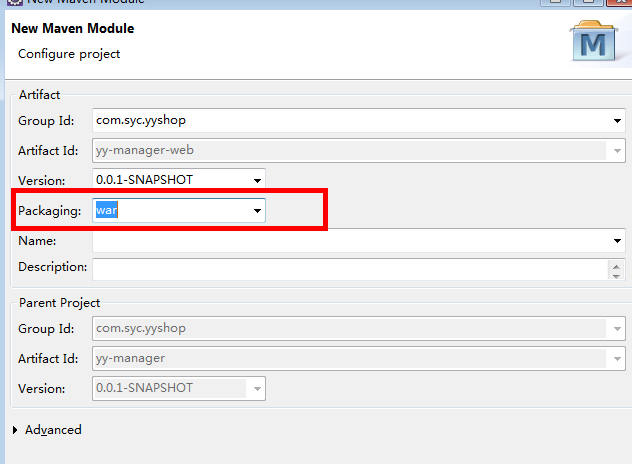
### 11.创建yy-manager里的service子模块





### 12.创建yy-manager里的web子模块

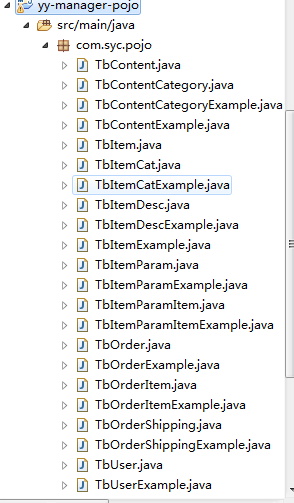




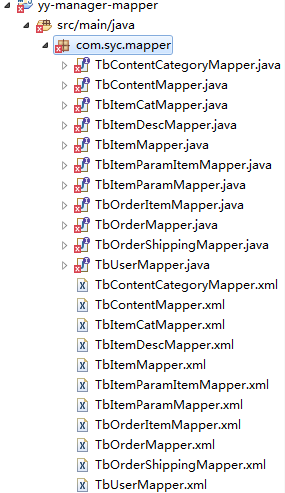
### 13.建立起个子模块之间的依赖关系

web--🡪service-🡪mapper-🡪pojo

### 14.根据逆向工程生成一列性的文件,把JavaBean与对应的xxxExample类添加到yy-manager-pojo项目中



### 15.把逆向工程生成的Mapper接口类和Mapper.xml文件拷贝到yy-manager-mapper项目中



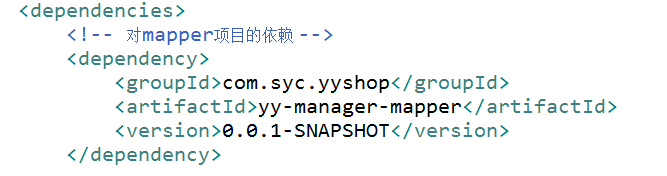
修改pom.xml文件,添加相关的依赖包和相关的插件的配置.

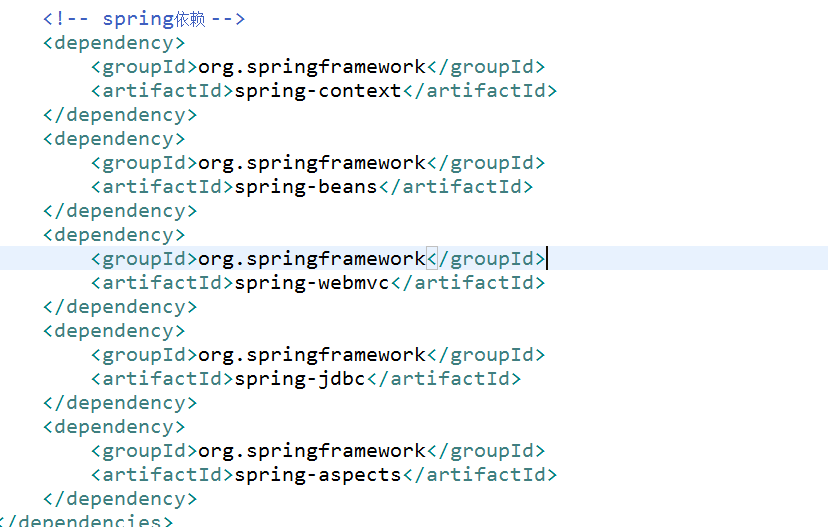






### 16.配置yy-manager-service项目的pom.xml文件





### 17.创建yy-manager-service项目中的接口和实现类

### 18.配置yy-manager-web项目,在pom.xml文件中添加相关依赖





### 19.在yy-manager-web项目中进行SSM整合

