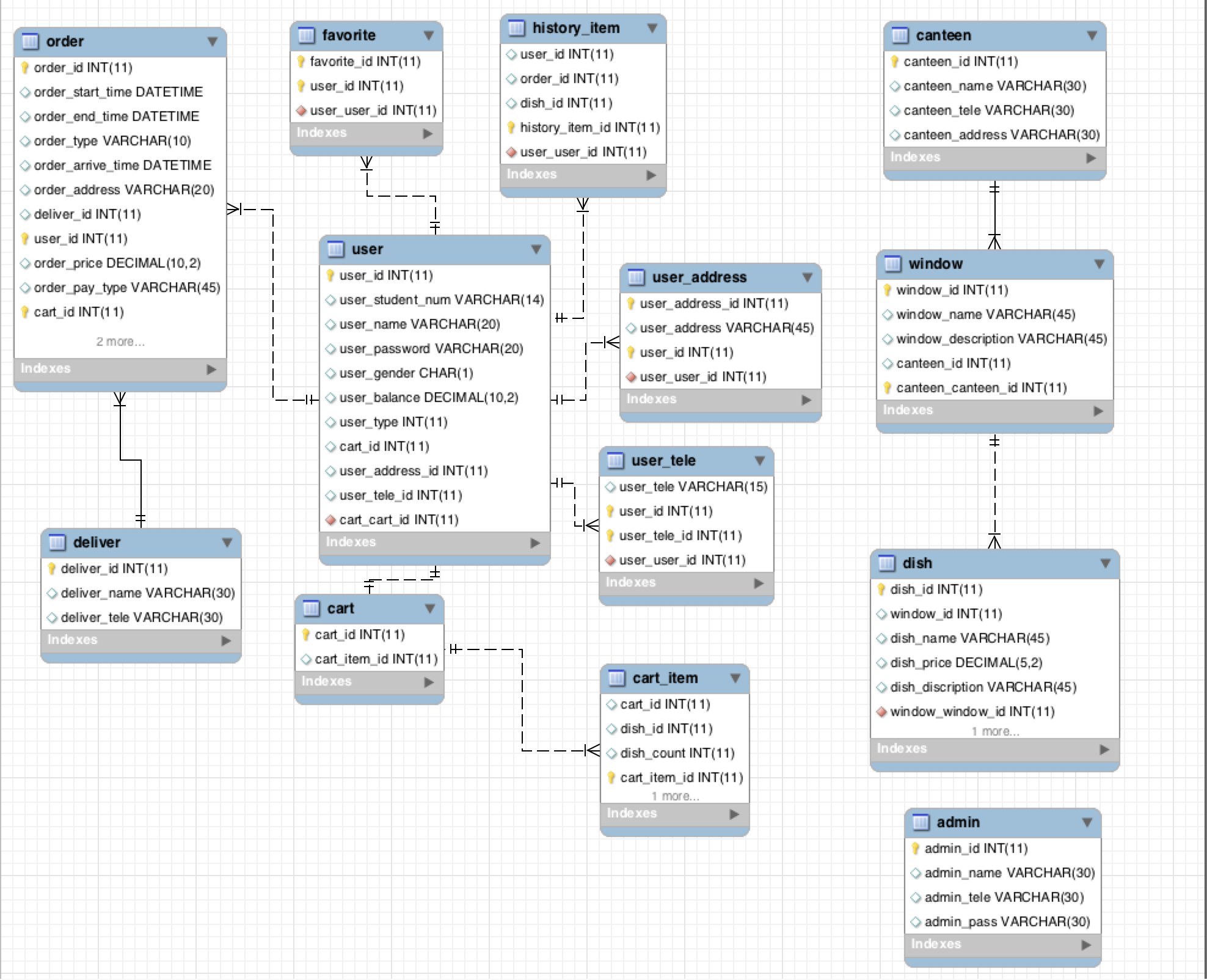
作品思路：

从底层数据库做起，使用SSM（Spring+SpringMVC+Mybatis）框架搭建Maven工程，整个项目分为POJO（Plain Ordinary Java Object）层，DAO（Data Access Object）层，Service层，Web层，其中Web层又分为View层（jsp页面，css/js等静态资源）和Controller层。

数据库：先做出各种类的model模型，画出ER关系图，理清不同模型之间的映射关系。一对一关系：在主表中添加从表的主键。一对多关系：在从表中添加主表的主键。多对多关系：添加联合主键表方便查询，主从表均设置对方表主键，不过内容为空。在做好各类表的主键外键工作后进一步填充模型属性，将金融字段设为decimal(10,2)，时间字段设为datetime，各表属性名遵循下划线命名规则。设置好所有模型的属性后填充虚拟数据。以便做CRUD测试。



POJO层&DAO层：通过数据库表利用Mybatis-Generator逆向工程插件生成POJO实体，Mapper.xml和Mapper.java。对于数据库中的每个表都有与之对应的POJO类，而对于每个POJO类都会有与之对应的mapper接口和xml文件，利用mybatis自动实现mapper接口的原理，对每个对象进行CRUD操作。再在此基础上将实体类的某些简单数据类型的属性替换成实体类型，同时更改Mapper.xml中的resultMap和sql语句与Mapper.java方法中的参数，根据自己实际需求增加或者删减方法。在DAO层的test方法中逐一测试各类mapper的方法，均通过后再开始Service层的搭建。

Service层：

一个service对象可以拥有多个mapper对象，也可以同时操作多个POJO类。例如User实体对象属性里有userTele,userAddress这两个实体类，但是在做Service层时只有一个UserService对象而没有UserTeleService，UserAddressService对象，Service层的目的是对DAO层的进一步封装，一个UserService里有多个mapper类进行同时操作，封装完成后的Service在外部看来就像User是一个整体模块，囊括了所有相关的操作。如user实体的删除操作，需要userTeleMapper,userAddressMapper,cartMapper等删除数据库对应含userId的表，所以一个user删除的方法里就有各种mapper的删除方法。在Service层的test方法中逐一测试各类的service方法，通过后开始搭建web层。

Web层：

Controller层：Controller负责页面请求的跳转，View层则是呈现给用户的前端页面，在controller里面调用自动装配好的Service对象，当请求发送成功后，dispatcher-servlet自动拦截请求，随后交付给对应的controller，controller再找到对应请求的方法将前端传入的参数处理，返回带数据模型的新的jsp页面，处理过程中会涉及到数据库的CRUD操作，因此也会根据需要运用Service对象。由于Service封装了很多细节，在调用方法的时候也会便捷很多。

View层：View层的调试则在浏览器的html检查器进行，遵循css盒模型原理，将组键位置调到合适位置，设置宽高比，在<meta>标签里设置等比例放缩和最小设备的宽高，做到响应式布局。通过JS,Jquery添加事件，JSTL，EL进行数据渲染，发送 .do 请求Controller。

将项目部署到服务器：

主要分为五个步骤：

1.购买服务器

2下载xshell和Xftp 6

3.在服务器上安装 jdk，mysql，tomcat。

4.数据库准备数据

5.maven项目生成war文件。

(6.有条件为公网IP备案一个域名)

前3步略过

将数据库搭载在服务器上:

在本地mysql workbeen 和 navicate 上新建连接，把公网IP和数据库对应的端口既数据库的root 账号和密码输进去，把所需要的database在本地导出为sql文件，再在服务器的mysql数据库上导入一遍，最后在web层的db.properties文件中配置为服务器端的url，用户名和密码。

将项目部署到服务器上:

将项目maven install 生成 的 .war 文件,传入服务器tomcat9下面的wabapps目录中。还是使用Xftp6, 修改 tomcat 中 conf 目录下的配置文件 server.xml，在配置文件中找到Host标签，新增Context。这里使用的是相对路径，doBase 属性的值是我们所打 war 包的名称。进入 tomcat 的 bin 目录，点击 startup.bat 启动我们的tomcat。显示如下结果则说明部署成功。这时候我们打开浏览器输入 http://fuke0204.top:8080/login/test.do，就成功访问到了我们的登录主页。