1、每个人本地搭建好开发环境

（1）JDK

（2）开发IDE：eclipse、IntelliJ IDEA

（3）maven：配置nexus私服的地址，settings.xml

（4）git

2、完成资源申请

对你的技术架构中，所有需要使用的技术相关的资源进行申请

如果是在大公司里面：需要跟对应的dba、redis运维团队、RabbitMQ运维团队，申请你需要使用的数据库、redis集群、RabbitMQ资源

还有你的java服务需要部署的机器资源，也需要跟SRE（运维团队）来申请

而且要申请好所有环境的资源

（1）开发环境（dev）：在这个环境中，部署需要使用的mysql、redis、rabbitmq，等等，然后各个开发人员在自己本地，是直接连接统一的一个开发环境中的数据库、缓存、mq，来进行开发和测试的。单元测试，不跟外部依赖的基础设施打交道的；但是在本地冒烟测试，是需要将自己负责的程序跑起来，跟开发环境中的基础设施交互，确保主流程没问题

（2）集成测试环境（beta）：在这个环境中，其实就是所有人对一个版本都开发好之后，需要进行联调，系统集成。将所有人负责的东西全部集成到一起，如果是单块应用，将每个人的代码分支全部集成到一个统一的分支上去，将那个分支上集成好的代码，部署到集成测试环境的机器上去，同时连的是集成测试环境中的数据库、缓存、MQ。集成测试，部署好之后，由QA来执行，有时候RD自己来做，要将整个系统的所有部分集成在一起，主流程，主功能，跑通。

（3）系统测试环境（test）：在这个环境中，同样，是需要将集成好的代码，部署到系统测试环境中的机器上去，同时连的是系统测试环境中的数据库、缓存、MQ。主要就是测试工程师，QA介入，进行黑盒测试，他们会写各种测试用例，非常非常的细，这一步重点是测试系统的所有功能包括各个细节点，全部都正确。

（4）验收测试环境（staging）：在这个环境中，尽量保证跟线上环境一致的情况。比如说mysql，主从架构；开发环境、集成测试环境、系统测试环境里，可能就是单实例mysql；在验收测试环境中，就需要根线上尽量保持一致，必然线上规划的是mysql一主三从的架构，在staging环境，可以简化为一主一从。在这一步的重点意义是，模拟线上环境，甚至有时候会将部分线上环境的数据，或者是请求流量，导入到这个环境中来。这一步重点是需求方来做验收，比如说产品经理，准备好一堆测试用例，在这个环境去使用你的系统，看是否正确

（5）线上环境（prod）：线上环境，所有的基础设施全部按照规划好的线上环境配置来走

可以看一下，这里各个环境的资源怎么申请，申请多少，怎么部署，是根据什么来的？

概要设计里面的物理架构：配置图，硬件层面去考虑，需要哪些机器，部署哪些基础设施，多少台机器，多少个集群，跨几个机房

3、数据库初始化

基于详细设计中的数据库建模，PowerDesigner，里面设计好了各种物理表结构，导出来一份SQL文件

直接基于那份SQL文件，在开发环境中的数据库里，建好需要的库，初始化好所有的表结构

4、 工程搭建

一般会指定一个人，架构师，也可能是项目组中的其他人

负责在自己本地的IDE中，建立好一个工程，对这个工程中完成系统需要的技术框架的整合（Spring Boot + Spring MVC + Spring + MyBatis）

（1）maven的pom.xml中，引入所有的依赖

（2）对所有需要使用的技术，完成对应的配置，注解 / xml

（3）其他的工程结构，完成搭建：maven父子工程，maven pom.xml中配置跟nexus集成、用来打包的插件、用于单元测试的依赖、数据库连接池的依赖

（4）本地写一个最基本的测试代码，将框架技术的整合全部跑通：接收http请求 -> 插入数据库，读写redis，读写rabbitmq，单元测试，日志的本地打印

5、代码上传到仓库

负责搭建工程的那个哥儿们，将搭建好的工程代码，上传到GitLab的仓库中去

项目组中其他的同学，就需要从GitLab中拉取代码下来，到自己本地，代码导入IDE中

将负责搭建工程的那个哥儿们写好的那套测试代码，在本地运行一下，确保说，所有技术的整合以及运行全部能跑通