概要设计已经完成，而且通过了评审853769620

（1）逻辑架构 -> 技术架构，已经确定ok了

（2）运行架构 -> 时序图，已经确定ok了，系统跑起来是什么样子的

（3）物理架构 -> 系统开发、测试和线上环境部署的

团队内部每个人认领部分模块或者子系统，然后每个人自己做详细设计

说一下，概要设计主要确定的是架构，详细设计主要确定的是如何去编码实现

也就是说，一份良好的详细设计文档，写好之后，基本你看着文档，就是傻瓜式的将设计翻译为代码了

实际上来说，你写代码的过程，就是一个傻瓜式的过程，因为你怎么写这个代码，在详细设计文档里基本都已经定义清楚了

所以我这边强调一下，不光是架构师，一个好的高级工程师，就应该能够做一份非常好的详细设计

最终落实到编码的时候，你可以发挥的空间就比较少了，发挥一点点，尽量把代码写的漂亮一些，注释，编码规范，代码格式

如果后面我们落实到一些技术上的东西的话，编码的时候，实际上会用到一些新技术，当然也是不错的

只可意会不可言传，之前有一个同学，拿到一个项目，不知道怎么做数据库的设计

你就没有去做需求分析（用例图、每个用例的泳道图），概要设计（时序图、逻辑架构图），你都没有考虑清楚

如果你把前面两个步骤做的很好，那么其实你就应该对系统整体的技术架构、业务架构、功能逻辑、业务流程，非常非常的清晰了

然后就根据你对上面那些东西清晰而且透彻的理解，去考虑，你负责的子系统中的各个功能，如果要实现，需要哪些数据库表来承载数据，才能实现这些

经验足够多的同学，会有一种多年经验积累和沉淀的作用，看到业务需求，能非常好的将其快速的转换为技术设计，资深工程师，架构师的价值，这也是那些培训机构刚培训出来的生瓜蛋子，对于这些同学来说，绝对是刚开始最多只能做到初中级的工程师，初中级，在北京，10k以下初级，10k~20k，中级，20k~30k，高级，30k~35k之间，算是资深，35k~50k之间，算是架构师了

如果是那种经验不足的生瓜蛋子类的工程师而言，我见过很多人，技术没问题，j2ee的技术都很熟练了，但是写出来的代码很烂

就是在详细设计，几乎从来不好好设计，或者不懂怎么设计，实现类图，面向对象的设计，要去实现业务功能，设计哪些类，每个类有哪些东西

我举个反例，我之前带过本科刚毕业1年的小伙子，在我团队里面，看过他写的代码，实现类图的设计，设计的很烂，就是完全没有思考一个业务需求下，要拆分为哪些类，哪些对象，哪些接口，每个类负责哪些职责，类与类之间如何互相交互。

可能一个较为复杂的功能，就设计了一两个类出来，每个类里就两三个方法，每个方法囤积了大量的代码 -> 垃圾代码

因为面向对象做的太烂了，后面让人根本就看不懂代码，然后也没法维护代码

维护，往代码里面加入各种新功能的时候，难以下手。。。。。。

每个方法里大量的代码，没有注释，没有良好的划分不同逻辑的一块代码之间要有空行，代码全都堆积在一起

像这种垃圾代码，那个小伙子，他的技术是没有问题的，j2ee那套东西，redis内核，好学，偏偏写出来一手垃圾代码

基本功！！！很重要很重要，基本功，连面向对象设计都没掌握好，就胡乱去看那些redis内核分析之类的东西，本末倒置了，你做出来的都是垃圾