

《开源软件设计与开发》课程总结

姓名：林佳 学号：51195100035

1、开源理解

开源：开源即为“开放源代码”。通常来说，指的是将软件项目的源代码向大众开放，允许大众获取、使用、修改和发行。

开源项目：简单理解就是，开放源代码的软件项目。可以认为，开源项目的产出物是软件程序。在遵守开源协议的条件下，可以不断对一款开源软件进行维护和升级，或者有可能在一个开源项目中，可以通过修改代码定制出多款不同的个性化软件。

开源软件：顾名思义是指公开源代码的软件，换言之，如果软件的源代码是开源的，那么这个软件就可以称之为开源软件。软件的作者让源代码对其他人提供，需要的人都可以查看、复制、学习、修改或分享代码。公司和个人可以根据自己的需要基于开源的软件源代码，修改或衍生出真正适合自己的软件产品。

开源社区：为某个开源项目的开发成员提供的一个学习和交流的空间。由于开源项目常常需要散布在全世界的开发人员共同参与推进，所以开源社区就成了他们沟通交流的必要途径。社区，这个是我在上课的时候听到的一个非常富有情怀的词，是一群有相同爱好与信仰的人的集合。我常常觉得，程序员真的是一群很单纯很善良又有着自己坚持的群体。在很多开源项目中，都有很多程序员志愿者们在默默地付出。

2、开源贡献

第一次接触 Github 是在大学的时候想写自己的博客，咨询前辈听他说起这个网站，通过 Github 可以很方便的进行版本管理。那个时候的 Github 的私有仓库还需要收费，据说很多代码放在 Github 上公开，有些重要信息会泄露。因为这个原因，更多也因为懒，这个计划后来就不了了之了。后来我分别去两家公司参与了实习，在第一个公司的项目代码是通过 Gitlab 托管的，第二个公司用的是 Github。个人其实更喜欢 Gitlab 一些。

在本学期，我参与的项目是华为的 ServiceComb。为什么选择这个项目呢？这个项目对我来说和 Dubbo 似乎有些像，只是一个华为的，一个是阿里的。因为彼时华为正要发布 mate30 5G，当时觉得这个很棒，于是我成了华为的脑残粉。了解了这个项目之后，也越发觉得这个项目很厉害。

ServiceComb 取名寓意蜂巢，是微服务开源项目，Apache 软件基金会孵化项目，其源自 ServiceComb 商业版华为云微服务引擎 CSE，于 2017 年 5 月开源开放。原主体代码于 2017 年 12 月由华为捐赠给 Apache，是全球首个进入 Apache 软件基金会孵化的微服务开源项目。其提供一站式开源微服务解决方案，融合 SDK 框架级、0 侵入 ServiceMesh 场景并支持多语言。ServiceComb 基于自身的愿景，一直坚持“将复杂扔给自己，将极简留给用户”，在设计上不断追求能最简单化为用户带来价值，具有以下优点：1) 开箱即用：集服务注册、发现、通信和微服务治理能力为一体，并默认提

供集中化配置，结合脚手架能力，实现微服务治理能力 0 编码，开发人员简单配置即可使能微服务常见功能； 2) 第一个实现纯异步内核的微服务框架：实现超低调用时延和高吞吐能力，高性能。**ServiceComb** 底层基于 **vertx**，但在 **vertx** 之上进行了进一步封装，同时支持异步及同步模型； 3) 同时支持框架式 **SDK** 和 **Service Mesh** 0 侵入微服务化两种方式，实现两种微服务化方式运维管控统一，降低用户运维管控成本； 4) 唯一一个基于 **Swagger** 实现契约和代码双向自动生成； 5) 分布式事务最终一致性：

ServiceComb 目前拥有三个主要的子项目，分别为：

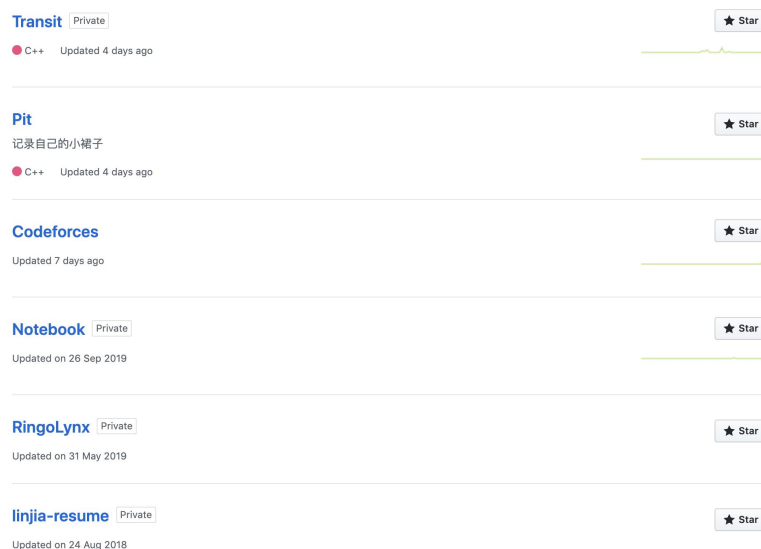
1) **Java chassis**：开箱即用 **Java** 语言 微服务 **SDK**，含服务契约、编程模型、运行模型与通信模型四个部分，具备负载均衡、容错熔断、限流降级、调用链追踪等全面微服务治理能力，服务治理能力与业务逻辑隔离。

2) **Service Center**—服务注册中心：基于 **Etcd** 的高性能、高可用、无状态的 **Golang** 版分布式服务注册与发现中心，可实时服务实例注册、实时服务实例推送和服务间契约测试等。

3) **Saga**—分布式事务解决方案：**ServiceComb Saga** 是提供了分布式事务最终一致性解决方案，用户只需要通过注解方式定义事务的执行方法以及撤销方法，**Saga** 框架会自动保证分布式事务执行的最终一致性。

由于这个项目对我来说过于庞大，我只能瞻仰，所以也没有作出贡献。

目前我在 **Github** 上主要存放自己日常的学习笔记、简历以及一些代码。公开的项目只有两个，处于自娱自乐的状态，也不敢说有什么贡献。其中一个仓库我储存了自己购买 **Lolita** 裙子、以及 **JK** 制服的心得。未来我将努力参与到他人的开源项目中去，感受开源的魅力。



3、课程反馈

1) 开源项目的设计与构建，对于很多新手来说，可能让大家分组做一个从无到有的项目构建，可能会学到更多。直接让我们参与到一个庞大的项目中，让人望而生畏。

代码这么多,我要从哪儿开始看?这个项目看起来已经实现的很不错了,我这么菜的人,又能改进哪里?直接让我提 pr,我真的有些懵逼。

2) 请了很多业界大佬,让我们见了见世面,也给出了很多实用的建议。但是很多老师的讲课内容之间让人感觉没什么联系,整个学期回忆起的课堂内容都是零零散散的,没有逻辑性,更像一场场分享经验的报告。

4、参考文献

无。