

第六周周报-李美瑾-2019141420032

(一) 本周完成的任务:

参与了项目计划撰写, 负责项目应交付成果、项目开发环境、项目验收方式和依据、项目支持条件等部分的内容。

学习了拦截器如何事项, 使用注意事项, 以保证项目的安全性。java 里的拦截器是动态拦截 Action 调用的对象, 它提供了一种机制可以使开发者在一个 Action 执行的前后执行一段代码, 也可以在一个 Action 执行前阻止其执行, 同时也提供了一种可以提取 Action 中可重用部分代码的方式。

学习了 Redis 的基本概念: redis 是一个 key-value 存储系统。和 Memcached 类似, 它支持存储的 value 类型相对更多, 包括 string(字符串)、list(链表)、set(集合)、zset(sorted set --有序集合)和 hash(哈希类型)。这些数据类型都支持 push/pop、add/remove 及取交集并集和差集及更丰富的操作, 而且这些操作都是原子性的。

学习了 API fox 的基本功能, 阅读了官方文档, 学会了如何使用 APIFox 进行后端接口测试。因为本人负责餐厅管理员部分的后端开发, 因此与负责相应功能的前端程序员讨论了部分 API 接口如何定义、规范。

复习了 Springboot 和 mybatisplus, 学习了 Spring Cloud。最终, 我们决定使用 Springboot 作为后端框架, 因为 Spring Cloud 作为一系列框架的有序集合, 通过微服务部署简化了维护和开发, 适合本项目。我们可以利用 Springboot 的开发便利性巧妙地简化分布式系统基础设施的开发, 如服务发现注册、配置中心、消息总线、负载均衡、断路器、数据监控等。、

确定了后端需要保存在 Session 中的静态数据:

序号	数据	数据描述
1	系统名称	数据类型: 字符串
2	角色编号	数据类型: 整数, 取值范围: 1-3
3	角色名称	数据类型: 字符串, 取值集合: 餐厅管理员、餐厅服务员、后厨人员, 分别对应角色编号 1、2、3

(二) 下周计划

- 1) 设计并建立 MySQL 数据库
- 2) 根据 UI 设计原型完成管理员后端部分的 API 文档编程

- 3) 与前端人员交流，明确 API 设计数据传输规范
- 4) 开始员工管理部分的编程