

**龚雨健**

## 电话：17626046621 邮箱：[924325679@qq.com](mailto:924325679@qq.com)

**教育背景**



**2013-2017 常熟理工学院 软件工程(本科) 苏州**

**个人网站**



**个人博客：<https://death00.github.io/>**

**工作经历**



**2018.8-至今 Java 工程师 南京呆萌猫网络科技有限公司 南京**

**2016.11-2018.7 软件工程师 高伟达软件股份有限公司 南京**

**工作内容**



1. 负责在线对战小游戏服务器的更新和优化。
2. 负责微信小游戏的后台数据统计服务。

# 项目经历



## 2018.11-至今 小游戏监控后台

项目内容：

该项目初始只是为了管理公司内部的小游戏（配置基础数据），并查看小游戏的各项统计指标（新增、日活、留存率等）。现在随着与外部渠道商的合作更加深入，正在转变成一个更加全面的统计平台（类似阿拉丁统计，包括各渠道统计数、事件漏斗分析等）。

本人的主要工作为：

1、利用 Spring Boot 实现微服务（包括接收小游戏上传的数据、调用微信api、定时调度生成统计数据、后台管理服务）。

2、针对后台统计服务器，利用 nginx 实现负载均衡，满足现在最高 1000 并发下 95%的请求，服务器响应速度均在 200ms 内。

3、通过慢日志优化 MongoDB 数据库索引，从而减轻了数据库服务器的压力，并减少了请求的响应时间。

4、利用AOP 和分布式锁（ MongoDB 和 Redis ），升级定时调度服务为分布式。

## 2018.10-2019.1 星 空 大 乱 斗

项目内容：

1、这是一款机甲题材的io 游戏，在游戏里玩家可以选择各种奇怪的机甲互相对战。

2、游戏服务器整体采用Spring Boot，与客户端的通信服务采用netty 框架，数据格式使用

Protobuf，服务器之间的共享缓存利用redis，数据库采用MongoDB。

本人的主要工作为：

1、主要负责游戏服务器中的升级和优化，以及相应的后台监管系统的开发。

2、梳理逻辑，利用线程池，将同一房间内的所有消息放在一个线程中执行，从而将多线程问题简化成单线程。

3、优化DDL 层（脏数据层，利用本地缓存构建），将订单、玩家相关的信息等，不在业务逻辑中直接入库，从而可以提高响应速度，并模拟出事务的功能，补足低版本 MongoDB 没有事务的缺陷。

4、利用 ReentrantLock 和 Condition，灵活控制任务的执行。5、利用Junit 和Mock，完成单元测试，辅助开发。

## 2018.8-2018.11 滑小稽赛跑

项目内容：

1、这是一款跑酷对战类小游戏，玩家可以与ai 对战或与好友对战，完成任务，获得奖励。

2、游戏服务器整体采用Spring Boot，与客户端的通信服务采用netty 框架，数据格式使用

Protobuf，服务器之间的共享缓存利用redis，数据库采用Mongodb。

本人的主要工作为：

1、主要负责游戏服务器的开发（包括提供玩家登录的门服务器、提供非游戏中的所有功能的大厅服务器、以及提供游戏中所有功能的游戏服务器），以及后台监管系统的开发（监控当前在线玩家情况、游戏情况）

2、解决由于内部逻辑问题引发的影子用户与影子房间问题。

3、利用Junit 和Mock，完成单元测试，辅助开发。

## 2017.11-2018.8 安徽农商行产创平台

项目内容：

1、该项⽬是为安徽省 84 家农商⾏及省联社打造的产品管理平台，从创意到需求再到产品，层层递进，精细管理。

2、该项⽬前端使⽤bootstrap 框架，后台使⽤了SSM 框架、安全框架Apache Shiro、⼯作流引擎Activity、分布式rpc 服务框架dubbo，数据库使⽤DB2，服务器采⽤的是WebSphere。

本人的主要工作为：

1、业务上，主要负责产品发布与产品模板业务中基本的增删改查。

2、利⽤quartz 框架完成项⽬的定时调度任务，并扩展为分布式部署。3、将原来的单机zookeeper 配置，扩展为分布式配置。

4、该造原本的文件上传功能，使其能支持大文件上传。

## 2016.11-2017.4 江苏银行贷前贷后调查报告

项目内容：

1、本系统主要是为银行借贷报告而做的系统，记录客户贷前情况及贷后还款情况。由银行告诉我们需要的报告类型及其详细信息，我们将报告作为模板的形式进行创建，供行方可以。

2、系统前端采用Bootstrap 框架，后端主要采用SSM 框架，数据库采用Oracle，日志记录主要采用Logback 组件。

本人的主要工作为：

1、熟悉SSM 框架，完成基础的增删改查。

2、仿照他人写的Java 自定义注解，结合SpringAOP 拦截了Controller、Service，将用户操作及系统报错存进数据库中，再展示到页面上，方便日常上线时对于日志的查看。

3、仿照他人写的JSP 标签，自己写一个可以访问数据库的标签，主要解决系统中前端页面对于一些常量数据的读取和写入。

4、快速学习并使用新控件（包括Ueditor 和echarts）。