|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 王玉龙/男/30王玉龙王玉龙/男/30/王玉龙/男/30男王玉龙/男/30/30王玉龙/男/30  |  | | --- | | **武汉理工大学/航运学院/硕士** | | C:\Users\ERON\Desktop\icons8-secured-letter-60.png | | wangyulong.eron@outlook.com | | C:\Users\ERON\Desktop\tel.png | | 153 6498 3973 | | C:\Users\ERON\Desktop\github.png | | <https://github.com/NAVERON> | | 求职意向求职意向求职意向求职意向 软件开发工程师  **Java**, Linux, **Python** | | 技能技能技能技能 Springboot后端, Linux等  后端开发, 算法研究, Shell脚本 | |  | |  | | --- | | **工作经验和求职意向** **工作经验：小米应用商店(2019.7-2022.02) :** 应用库管理及入库自动化测试平台后端开发, 2年; **华为OD工业软件云项目(2022.12-2023.12)** : 仿真云处理平台用户认证鉴权后端负责人, 1年; **实习(2017.9-2017.11)**：武汉创业公司, 机器学习基本理论, Python爬虫实现, 3个月.  **求职意向：**程序/软件系统的开发, 努力成为一名优秀的全栈工程师. | | **项目实践和开发技能**Java Web**后端**项目 **+ 避碰算法研究** **Springboot后端项目：**开发和完善安桌应用入库自动检测系统，实现APP基础性能的自动化检测和入库; 独立实现仿真云平台用户认证鉴权系统, 微服务后台软件开发等; 开源Springboot实战项目, Python/Shell工具脚本, 实际问题解决项目等.  **无人艇避碰仿真**：JavaFx实现无人艇避碰仿真客户端验证, Python实现机器学习避碰和效果仿真展示. | | **教育背景和兴趣爱好** **本科**： 武汉理工大学，航海技术专业  **本科辅修**： 华中科技大学，计算机科学与技术专业  **硕士**：武汉理工大学，交通信息工程及控制研究生  **兴趣爱好** : 绘画艺术设计, 游泳、跑步、爬山等运动 | |

## **教育背景：**

**1. 交通信息工程及控制 研究生 | 武汉理工大学 航运学院**

**起止日期**：2016.9 – 2019.7

**主要研究方向**：搭建无人艇避碰仿真平台和实验平台，无人艇避碰方面的问题研究以及解决办法

**主要课程**：水上交通建模与仿真，数学模型，船舶控制理论与技术，船舶交通流实验等

**发表论文**: 群无人艇避碰算法研究, 群无人艇模拟避碰软件著作一项

**2. 航海技术 | 武汉理工大学 航运学院**

**起止时间**：2012.9 – 2016.9

**主要课程**：船舶原理，航海学（地文+天文），船舶操纵与海上搜救等

**获奖情况**：获得学校二等奖学金，优秀学生干部、优秀毕业生、课外培养优秀学生

**3. 计算机科学与技术 双学位辅修 | 华中科技大学 计算机学院**

**起止时间：**2014.3 – 2015.10

**主要课程：**离散数学，数据结构，操作系统，编译原理, 软件工程等

**主要成就：**基于毕业设计论文实现并完善软件共计4个版本

## **项目经验：**

**1. 小米Java后端开发2年 | Java Web开发**

**(1) 应用自动化检测系统开发和优化**

**项目技术栈** : Springboot, MQ, MySQL , Zookeeper

1. 使用SpringBoot Schedule 调度器实现业务内容的定时运行. 实现对测试进度和异常状态的监测, 如有测试积压和测试时间过长的情况, 根据紧急程度发送邮件和短信, 内容包含告警应用的详细信息
2. 异步处理的设计. 任务拉取, 任务分发, 任务回调, 任务结果处理等过程相互之间不干扰, 使用数据状态保证测试幂等性; 任务分配借鉴一致性Hash算法思想. 计算空闲设备和待测试应用之间的比值, 根据比值确定分配位置, 在资源有限的情况下尽可能保证测试效率和可靠性
3. 测试和结果评判标准分离. 从配置入手, 在应用检测的两端逻辑考虑, 实现应用检测标准和应用合格标准的分离, 实现检测但不作为标准的功能
4. 特殊配置优化. 使用zookeeper节点保存Json格式的特殊配置, 如黑名单等, 本地注册对节点的事件监听, 实现配置上一定的灵活性

**(2) 兼容性标签**

**项目技术栈** : JSP, Spring, API

1. 实现应用绑定标签的增删改查基本功能. 在实现删除功能中, 使用假删除/批量方式实现数据比对的功能; 使用分段查询避免查询阻塞和影响数据库性能, 对于返回大量数据结果集合的情况下性能提升显著
2. RESTful风格接口. 对外提供RESTful风格接口, 根据筛选条件查询标签数据; 同时, 实现从手机部门数据中心周期同步数据到本地数据库, 采用全量, 增量, 每日相结合的方式同步

**(3) 基础需求开发**

1. 实现完整的有害应用库后台管理的开发. 功能包含有害应用入库, 信息修改, 审核流程记录等
2. 审核后台的日常bug修复, 如审核记录内容缺失问题, 页面多语言显示问题等

**2. 仿真云平台后端开发 1年 | Java后端开发**

(1) 独立设计和实现用户管理, 认证, 鉴权体系, 基于Oauth2协议提供用户三方登录

(2) 为平台各个微服务提供认证底层实现, 无感知接入自定义认证鉴权系统

(3) 在用户管理基础上, 实现团队管理功能, 并于前端联调上线, 软件开发全过程

**3. 创业公司实习 3个月 | Python**

(1) 机器学习理论内容, 协助搭建edx平台 ; 使用python学习爬虫并爬取英文网站的文章单词处理

(2) Linux 平台命令使用, 技术选型资料整理等

**4. 无人艇避碰仿真软件 | JavaFx**

**(1)** 本科毕业设计以无人艇避碰为题, 实现避碰仿真，基于Java Swing搭建主界面，解决多线程动画抢占问题并在后续版本中逐步优化 ;

**(2)** 总结第一个版本运行卡顿问题，使用总体线程调度的方式/串行实现仿真流程，定制化UI组件 ;

**(3)** 理解Java Socket编程实现客户端与服务端通信过程, 使用ServerSocket实现通信监听，Swing客户端模拟雷达扫描效果，实现局部区域可视化。推导演算实现局部/全局坐标系转换计算、相对方位、相对速度和速度矢量的计算等 ;

**(4)** 运用JavaFx实现客户端界面，充分借鉴之前版本的利弊, 合理布局和设计逻辑结构, 实现基本的避碰对象管理和避碰信息实时显示功能.

**仿真系列链接**：<https://github.com/NAVERON/ShipSimulation>

**4. 实用工具软件和开源项目等 | Linux Shell, Java, JavaFx, Python, C**

(1) 小工具开源软件 : <https://github.com/NAVERON/ArbitraryCoding>

(2) Linux Shell 脚本 : <https://github.com/NAVERON/ERON/tree/main/configuration>

(3) Python 机器学习应用 : <https://github.com/NAVERON/MachineLearningNotes>



**GITHUB 开源代码**

## **技能：**

**1. 编程基础和数学基础：**

* 编程技能 : Java, JavaFx, Python, Linux Shell, C, Maven, Git
* 具备Java Web的编程经验, 独立制作多个demo项目, 对解决java web中的问题有一定的见解 ;
* JavaFx客户端开发开源项目, 实现了工作打卡软件, 学校通知搜索软件等 ;
* Python 实现机器学习库的使用和应用, 独立学习并应用强化学习理论实现无人艇避碰 ;
* Shell脚本实现便捷工具, 远程服务部署, 开发环境重建, 数据导出预处理等脚本 ;

**2. Java软件仿真设计：**

* 独立完成无人艇避碰仿真软件的迭代开发, 会根据具体需要学习软件体系并实现 ;
* 使用FXGL框架, 并借鉴整体逻辑实现仿真软件的总循环逻辑 ;

**3. 数学工具软件Mathematica使用：**

* 简单使用Mathematica 完成公式求解 , 数据图形绘制等 ;
* Maple 类似的简单使用和图形绘制 ;
* Matlab实现过落水人员图像自动检测脚本 ;

**4. 艺术设计：**

* 具备一定的艺术绘画基础, 专业学习过绘画基础, 写生和素描 ;
* 会使用Flash制作动画, Photoshop 简单处理图像, CorelDraw 制作Logo, GoldWave, 视频编辑软件完成多媒体制作和投放, 热爱自由创作 : https://naveron.github.io