

马腾飞

男 | 年龄: 29岁 | ✉ m_mone@163.com

7年工作经验 | 求职意向: Python | 期望薪资: 30-35K | 期望城市: 上海



个人优势

- 1.熟悉 Linux 操作环境;
- 2.Python 熟悉, JS, GO, Ruby 了解;
- 3.熟练使用 MySQL 数据库, 熟悉 Redis, DynamoDB, MongoDB等 NoSQL 数据库;
- 4.熟练使用 Fastapi, nameko等 Web 框架, 熟悉 apollo-server等 graphql 框架, 熟悉 scrapy 等爬虫框架;
- 5.熟悉计算机基础数据结构
- 6.熟悉微服务架构及无服务架构
- 7.熟悉web常见爬取及反爬方式, 熟悉JS逆向
- 8.带领过3-8人的团队

工作经历

北京瑞达恒建筑咨询有限公司上海分公司	Python	2022.07-至今
担任角色: 抓取团队技术负责人		
负责工作: 抓取系统从0到1设计与搭建, 新人培养, 技术团队规范制定		
获得业绩:		
搭建抓取管理平台, 日均抓取量年翻6倍		
团队技术培养, 辅导新人10+		
江曲投资咨询(上海)有限公司	Python开发工程师	2019.01-2022.06
1. Student.com的开发与维护;		
2. CMS后台系统的搭建与维护迭代;		
3. ETL脚本编写, 数据分析支持;		
4. Integration系统, Public API的设计与支持维护;		
5. Support测试工程师, 协助其编写自动化测试脚本;		
上海拜聚科技有限公司	python工程师	2018.08-2019.01
1、使用 Python 完成后台服务开发、维护和技术支持等;		
2、与后台 JAVA 团队进行沟通调试;		
3、按项目计划完成代码编写、内部测试等工作;		
4、根据公司产品需求, 参与讨论技术解决方案;		
5、参与新技术研究学习及应用。		

- 1、使用Python完成后台服务开发、维护和技术支持等；
- 2、按项目计划完成代码编写、内部测试等工作；
- 3、负责API的开发、设计和维护；
- 4、根据公司产品需求，参与讨论技术解决方案；
- 5、参与数据模型设计和实现；
- 6、参与新技术研究学习及应用。

项目经历

font-obfuscator 开发

2024.08-至今

字体混淆工具是一个开源的 Python 库，旨在通过混淆字体文件中的字符到字形映射，防止网页内容被爬虫轻松提取。该工具允许您在 TrueType Font (TTF) 文件中打乱或加密字符映射，从而增加网站文本的安全性。

功能特点

- 字符混淆: 打乱或加密 TTF 字体文件中的字符到字形映射。
- 字符集筛选: 只保留字体文件中的指定字符, 移除所有其他字符。
- 易于集成: 作为独立的 Python 函数使用, 可集成到现有项目中。

langchain-wechat 开发工程师

2024.03-至今

内容:

软件环境: python3.10+langchain+fastapi+itchat

项目描述：基于 langchain + fastapi + itchat 搭建的微信聊天机器人，灵感来自于 chatgpt-on-wechat

默认 Agent 方式执行，自动调用 DuCKDuckGo 搜索引擎，对搜索后的结果进行总结，返回给用户。

支持的消息类型

- 接收单人文本消息回复
- 接收群消息@当前用户 文本消息回复

支持的模型:

- openai

tools:

- duckduckgo 搜索

业绩：

star数量40+

RCC爬虫项目 爬虫工程师

2023.02-至今

内容:

软件环境: python

3.10+scrapy+redis+crawlab+k8s

项目描述:

1. 基于 scrapy 的通用爬虫项目
2. 基于 scrapy 的特殊爬虫项目
3. 爬虫公共组件库
4. 基于 fastapi + pytorch 的验证码识别服务
5. 基于 fastapi + langchain 的自动解析xpath服务

业绩:

1. 日均抓取量年翻6倍, 获得公司特殊贡献奖
2. 构建通用爬虫、抽离公共组件库, 提高开发效率
3. 搭建验证码识别服务, 月均减少公司开销2k+
4. 搭建自动解析xpath服务, 减少人力成本

RCC爬虫管理平台 后端开发工程师

2013.12-至今

内容:

软件环境: ruby3.1+ rails + node18 +vue

项目描述: 基于 rails的 web 项目, 主要用于爬虫网站、定时任务、抓取数据、抓取配置的管理,

1. 负责后端服务的架构设计与实现, 确保系统的高可用性和可扩展性。
2. 利用Ruby on Rails框架开发核心功能模块, 包括爬虫任务管理、数据抓取及配置管理等。
3. 协同前端团队, 通过Vue.js实现用户友好的界面交互, 提升操作效率。
4. 优化数据库设计, 确保数据处理的高效性和准确性, 通过定期维护保障系统性能。
5. 设计页面结构, 分析系统需求, 提升用户体验

业绩:

同时负责

- 产品设计70%工作, 输入原型图10+
- 后端开发80%工作, 接口 40+
- 前端开发10%工作, 页面 2+
- 测试工作100%工作, 测试 bug100+

Student.com 开发工程师

2019.01-2022.06

内容:

软件环境: python3.6+nameko+typescript+graphql+mysql+redis+k8s+aws

项目描述:

基于nameko+graphql的微服务架构

- student.com, 留学生公寓出租网站
- content.student.com 内部管理系统, 内部员工管理房源信息, 对账等
- landlord.student.com 房东管理系统, 房东管理房源信息, 对账等
- api.student.com student.com的API, 第三方可以通过API创建订单, 获取房源信息等
- integration系统, 与第三方系统(salesforce, student suite, sturents, slack等)进行交互

业绩:

- 通过integration系统和房东管理系统提升平台公寓总数, 实现10倍增长
- 维护开发公共组件库, 优化代码结构, 提升开发效率
- 引入缓存系统和反爬机制, 降低延迟, 提高系统响应速度
- 编写自动化脚本, 优化support流程

Student.com ETL系统 开发工程师

2019.01-2022.06

内容:

软件环境: python3.6+serverless+Pentaho Data Integration+Matillion+snowfalke+lambda+aws

项目描述:

基于pentaho+aws lambda的无服务架构

将student.com,salesforce,mixpanel等系统数据进行ETL, support数据分析部门

业绩:

- 从boomi integration(付费)迁移到pentaho integration(开源版)，降低系统维护成本
- 编写自动化集成脚本，提高集成效率
- 优化代码结构，提高开发效率和代码共用

CC数字货币交易所

python开发工程师

2018.08-2018.12

数字货币交易所平台项目，通过平台对接各大交易所进行交易，后端由java和python两个团队构成，java负责平台自身逻辑的实现，包括资管，交易，行情等模块.python负责平台与各大交易所之间的匹配交互业绩：

- 1.通过rest接口对接交易所交易、资管相关接口
- 2.通过websocket接口对接交易所行情相关接口
- 3.通过kafka与java后台进行数据交互，接收用户的订单数据，调用对应交易所rest接口
- 4.通过mysql数据库保存订单信息，与平台数据进行匹配
- 5.将各大交易所webscoket推送的行情数据通过kafka传给java后端

乐家电商平台

后端工程师

2017.12-2018.06

B2C模式的电商项目，该项目主要是完成网上购物系统的开发，该系统分为用户、商品、购物车、订单等模块。

分别在用户模块实现了登录、注册、收货地址等功能；商品模块实现了商品详情，商品搜索等功能；购物车模块实现了购买、登录合并等功能；订单模块实现了保存等功能。

业绩：

该项目整体采用Django REST framework框架。本人在项目中主要完成了用户和订单模块的开发，参与用户、订单、商品数据表的设计，参与商品模块的开发。

用户注册基于Celery实现异步发送短信，登录状态保存使用基于token的鉴权机制。订单保存使用乐观锁解决并发处理。使用Docker部署Elasticsearch实现商品搜索功能

华天金融资讯项目

后端工程师

2017.12-2018.01

一款资讯展示的Web项目，主要为用户提供最新的金融资讯、数据.以抓取其他网站数据和用户发布作为新闻的主要来源 业绩：

该项目整体采用Flask框架.本人在项目中主要参与了用户和新闻模块的开发

- 1.参与前期需求分析、技术方案选型，项目的架构设计、开发环境搭建、数据库设计；
- 2.注册登录模块：参与设计用户模型类，完成用户注册登录，采用手机号注册；
- 3.新闻模块：点击排行，新闻显示，新闻详情页，用户收藏新闻，用户信息显示等

教育经历

青岛农业大学

本科

生物技术

2013-2017