



15850770332  
xwang1024@126.com  
wxppt1220  
<https://github.com/xwang1024>

# 王鑫

Web前端工程师 / JS全栈工程师

## 教育经历

EDUCATION

2012 ~ 2016

南京大学 学士 软件工程 (本科)

- 2012 - 2013 软件学院院服设计大赛一等奖
- 2013 - 2014 国家励志奖学金
- 2014 南京大学“励志模范”
- 2016 南京大学优秀毕业生

2016 ~ 2018

南京大学 学士 软件工程 (硕士)

## 技能栈

SKILLS

前端

### HTML / CSS / JavaScript

- 熟悉 HTML 的基本规范和常见标签，熟悉 HTML5 的新标签和语义化规范，熟悉 CSS3 样式
- 熟悉 Jade(Pug), handlebars 等模板语言
- 有强迫症，能精准地还原设计图
- 熟悉 JavaScript，了解浏览器 JavaScript 的基本工作原理，了解基本的模块化方式，熟悉ES6规范
- 熟悉 jQuery，熟练使用 \$.ajax 或 fetch 和服务器进行数据交互
- 会使用 npm, yarn 或 bower 管理项目依赖

### AngularJS

- 了解 Module, Controller, Service, Filter, Directive 的概念和编写方式，熟悉基本的内置指令和过滤器
- 了解 AngularJS 的依赖注入原理
- 使用 config 对 \$stateProvider 配置定义路由

### React / Webpack

- 了解 Webpack 配置文件编写方式，以搭建 React 开发以及部署环境
- 熟悉基本的 JSX 语法，了解 Component 的生命周期，了解 Component 间的数据传递方式
- 了解使用 Redux 进行状态管理的过程

### LESS

- 了解变量定义方式、基本语法和嵌套规则
- 了解常见的 LESS 文件组织形式（类 Bootstrap）
- 了解 mixin 的编写方式和常见的内建函数

### gulp

- 了解 gulp 基于 Stream 和 Task 的思想
- 熟悉常用的 gulp 插件，能独立编写 gulp 脚本，完成源文件的编译、合并、压缩、sourceMap、hash 等操作



后台

### Node.js

- 了解 node 的基本原理，能够使用 express 实现 Web 服务，了解 async, request 等常用库的使用方式
- 熟悉 node 的异步模型，习惯使用回调或 Promise 的方式处理问题
- 会使用 PM2 部署，进行负载均衡和进程守护

### Mysql

- 了解 Mysql 的常见命令，了解数据库设计的基本原则
- 了解常见的 sql 数据库优化方式

### MongoDB

- 了解 MongoDB 和 Mysql 的区别及其适用场景
- 了解 MongoDB 的常用操作，会使用 aggregate 对数据进行统计

### Linux (Debian/Ubuntu)

- 了解常见的 Linux 命令，会搭建 Gitlab、gogs、NGINX 等服务，会配置常见的执行环境



其他

### Git

- 熟练使用 git 基本命令以及常见的图形化 git 工具
- 了解团队使用的 Merge-based 或 Pull-based 等协作方式，解并使用过 Git Flow 工作方法

### Photoshop

- 有长时间的 PS 使用经验，可熟练完成 PS 基本操作
- 熟悉图层、通道、智能对象等概念，可独立完成简单的图像处理工作

### 科学上网

- 善于通过 Google 获取信息，有较强的学习能力
- 在 Stack Overflow、Github 等网站可以正常英文交流



项目与工作经验

EXPERIENCE

南京一可米互联网有限公司 [创业] 2016-09 ~ 2017-02

### Web前端开发（实习）

- 对已上线的系统“米盒”进行前端的模块化重构，处理原有前端存在的 Bug
- 在“米盒”原有 PHP 后台和前端之间搭建 Node 渲染层，使用 Handlebars 进行后台渲染，使用 Redis 对常用数据进行缓存，加快前端渲染速度，对于已有 ajax 接口使用透明代理的方式进行透传
- 负责 SQL 数据库设计



📞 15850770332  
✉️ xwang1024@126.com  
🌐 wxppt1220  
🔗 <https://github.com/xwang1024>

# 王鑫

Web前端工程师 / JS全栈工程师

- 15天内独立完成了“Course+ 项目”的运营平台（CMS系统）。前端使用 **H5 + less + jQuery**，通过 **sea.js** 组织模块，编写 **Gulp** 脚本：使用 **Babel** 支持 **ES6**，编译 **less**，生成 **sourceMap**，代码 **minify/uglify**，生产模式文件名 hash，开发模式 **browserSync** 自动刷新等；后台使用 **Express.js** 提供 Restful 接口
- 负责服务器管理和日常维护，为团队搭建 Gogs 开发平台，负责“Course+”项目部署，配置 HTTPS 协议
- 使用**React**开发新版“Course+ 项目”，使用 **Redux** 进行状态管理，使用 **fetch** 获取数据，并对 **fetch** 进行了封装，完成了时间轴、URL 解析等组件和工具类
- 独立完成“Course+ 项目”的直播页面，使用 **WebSocket** 完成弹幕功能的前后台，使用 **React** 封装 H5 原生播放器，并通过 **flv.js** 支持了直播 FLV 流

南京亿丰嘉玮新网络科技 [创业] 2016-05 ~ 2016-08

## Web / 微信前端开发（实习）

- 使用 **H5 + less + jQuery** 完成“帕不怕公众号”微信管理后台，使用 **ECharts** 进行数据的图形化展示，独立完成高度可定制的量表（问卷）生成模块，可根据多种评分规则向用户提供建议
- 使用 **H5 + less + 原生JS** 完成“帕不怕公众号”微信移动端界面，完成通过采集设备加速度和陀螺仪信息进行的“摇钱树” **HTML5 小游戏**

杭州数梦工场网络科技有限公司 2015-06 ~ 2016-04

## Web前端开发（实习）

- 首次接触到工程化和组件化的前端开发
- 1周时间快速上手 **AngularJS, bower, npm, less, gulp**，之后从事缺陷修复工作，项目结束时由我修复的 Bug 占总量的一半以上
- 参加“PaaS 平台”中虚拟机转售模块、插件模块的开发，主要使用技术为 **AngularJS**。同时也认识了 **Cloud Foundry** 和 **Docker**
- 在“政采云”项目中，进行结算管理、发票管理、待办模块、消息模块的开发，完成了 Session 管理、Cookie 管理、URL 解析、Query 解析等工具类，完成了搜索组件、表格组件、Panel 组件、Alert 组件供整个开发小组使用，主要使用的技术为 **HTML5, Babel-ES6, less**
- “政采云”项目使用 **jigglypuff** 为前端提供数据 mock 以及模拟后台渲染以进行前后台分离开发，使用 **linner** 作为前端编译工具。在使用前者时我发现模拟渲染和后台渲染不一致的问题，我定位了问题并联系开发者，通过向其提交代码改进了该工具
- “政采云”项目工期紧，业务复杂，在开发过程中我面临了从未感受过的压力，在一起实习的同学都选择放弃时，我坚持了下来，我第一次实习最大的收获就是学会了如何面对巨大的压力



个人项目

源代码

### SIF-Resource-Explorer

- 手游“Lovelive 学员偶像祭”图片资源合成和导出工具
- 反编译手游的 Android 包和 C# 编写的工具包，分析该手游自定义图片格式中数据的组织形式，将自定义图片格式转换为常见的图片格式
- 通过反编译得到的代码，确定图片合成逻辑
- 使用 Java8 开发，使用 **JavaFX** 编写 UI，用户可通过选项查看资源或下载资源



外包项目

源代码

### 南京大学后勤集团 - 人力资源管理系统

- 3个月独立完成整个项目的开发，包括需求获取，系统设计、部署和测试，以及后续的维护演进工作
- 后台使用 **Node.js + Express.js + MongoDB**，提供 Restful 接口，使用 **EventEmitter** 解决 **Callback Hell** 问题，使用 **async** 解决同步问题，使用 **aggregate** 完成数据的分析和统计；后台通过使用 **node-schedule** 进行数据备份和定期统计
- 前台使用 **AngularJS + less + Jade** 进行开发，编写辅助脚本将 **swagger** 文档直接转换为 **ngService** 与后台的 Restful 接口对接，大幅减少了工作量。项目中使用 **ngGrid** 展示大量数据，使用 **\$formatter** 和 **\$parser** 完成树形下拉选择框的 directive