



个人信息

- 叫天笑
- 男/1990
- 籍贯：湖南株洲
- 学历：本科/湘潭大学电子信息科学与技术专业
- 工作年限：3年
- 博客：<http://ttian99.github.io>
- Github：<http://github.com/ttian99>

联系方式

- 手机：13266814898
- Email：jiaotianxiao@qq.com
- QQ号：176553611
- 微信：jiaotianxiao

技能清单

- 开发语言：JavaScript/Lua/Java/NodeJs
- 游戏开发：Cocos2d-js/HTML5
- Web开发：React/html/css
- 常用框架：Express/Bootstrap/Amazeui/Ant-Design
- 常用第三方类：lodash/moment/http-server/
- 前端工具：Gulp/Webpack/SaSS/LeSS
- 数据库：Mongodb/Redis/PostgreSQL
- 版本管理工具：Svn/Git

- 常用工具：TexturePacker/TiledMap/BMFont/Spine

工作经历

深圳乐市乐天成科技公司（2014年11月~2017年3月）

- 1.负责公司游戏前端UI的构建。
- 2.负责公司现有游戏服务器(6台腾讯云服务器)的运营维护。
- 3.负责公司游戏各渠道平台sdk接入和apk的打包。
- 4.负责公司游戏休闲游戏后台的搭建。

深圳市盛元半导体公司（2013年9月~2014年10月）

- 1、MOS管与IC产品测试程序的编写和测试系统的搭建，解决产品测试异常，统计并分析产品测试数据。
- 2、MOS管和肖特基二极管产线的测试工序管控，拟定对应产品的测试工艺文件，指导生产部门生产。
- 3、协助应用工程师解决客户投诉主要应用问题，包括节能灯和LED驱动项目，测试并且分析产品异常原因。

主要项目经历

1.来消星星的你游戏项目

- 项目概述：h5消除游戏，前端使用cocos2d-js，后端前期使用java+redis，后期使用nodejs+mongodb。在玩吧平台注册用户超1000万用户。
- 工作职责：负责游戏新功能的添加，后端的重构和维护，关卡编辑器的功能实现。
- 遇到问题：
 - ○ 1.公司没有专门的java开发，且服务器缺失说明文档，后端更新步骤繁琐。
 - ○ 2.前后端协议混乱，修改或者添加游戏功能时，前后端耦合度较高，需要同时修改，开发效率低。
 - ○ 3.后端数据结构混乱，部分多余数量，占用存储过高，使用redis做数据存储，大数据量下占用服务器内存较高，使用性能较低。
- 解决方案：
 - ○ 1.使用nodejs重构服务器，便于其他程序员协助开发，保持技术栈的一致性。整个后端项目使用git做版本控制，在服务器后端代码可以通过git更新。用markdown格式的文件作为说明文件，一并放入版本控制。
 - ○ 2.约定好前后端协议调取接口和返回数据，统一前后端协议接口置于同一个配置文件，将该文件作为子项目，嵌入到前后端项目中，可供前后端共同调用。实现了一处修改，两端更新。
 - ○ 3.使用mongodb作为数据库，通过mongoose连接管理，使用Schema对象定义数据字段和类型、索引和验证，访问速度显著提高。
- 游戏地址(用手机打开)：http://h5.qzone.qq.com/gamebar/framework?appid=1101249290&source=gamebarshare&_wv=132096

2.厨侠小破孩项目

- 项目概述: Html5/Android多平台消除游戏, 前端使用cocos2d-js + reactjs, 后端使用nodejs + mongodb。
- 工作职责: 负责后端服务器的搭建, 前端部分ui的构建, android平台sdk的接入。
- 遇到问题:
 - 1.公司项目较多, 后端代码版本较多, 维护成本高。
 - 2.游戏接入渠道较多, 且每次都需要手动处理游戏资源、拷贝sdk文件、进行打包, 开发效率较低。
- 解决方案:
 - 1.使用express框架重新架构后端, 实现一套代码复用所有项目。不同项目通过一个json文件配置对应的项目信息: 项目名、数据库名, 数据库地址等。提取不同项目的后端逻辑, 对应调用不同的协议, 每一个协议对应单独的一个js文件。后端根据配置文件的项目信息, 调用对应项目的协议。
 - 2.利用gulp封装一套自动化打包工具, 实现不同项目调用, 图片资源的压缩、合并、调整尺寸等操作, 通过node调用cocos的命令行工具进行apk的生成, 支持apk的重命名以及生成路径的控制。每个渠道配置成可选的子任务, 实现一键选择打包。
- 体验地址(请用手机打开): <http://1251001828.cdn.myqcloud.com/1251001828/pobaby/>

3.美一间项目

- 项目概述:web单页应用, 前端使用Amaze-UI-Touch+Reactjs, 后端使用nodejs, 原始数据存于postSQL。
- 工作职责: 独立负责整个项目的前后端功能实现
- 遇到问题:
 - 1.开发周期短, 人手不足, 一个月内需要完成整个项目的功能。
 - 2.原始数据量大, 数据库搜索访问较慢。
- 解决方案:
 - 1.考虑到自己可以快速开发, 以及后面其他程序员的容易上手, 前端ui采用Amaze-UI-Touch框架搭建。该框架基于ES6 + reactjs的组件式开发, 各个组件的demo和说明文档齐全, 便于学习, 按需添加到项目内即可。后端服务逻辑与游戏后端无太大差异, 所以后端可以复用express框架的游戏后端, 与游戏后端有差异的搜索逻辑部分写入了新的协议接口, 供前端调用。
 - 2.由于原始数据将近有1200万的数据, 直接搜索数据库会造成响应缓慢, 用户体验差。所以使用nodejs写个提取工具, 将每条数据的学校、小区和楼栋名字添加进该条数据的索引关键字, 再按相同的索引关键字统计数据, 并存入新的数据库。搜索时先遍历索引关键字, 再根据索引关键字的数据, 搜索出对应的数据, 极大的缩短了搜索时间。
- 体验地址: <http://each1.cn>

4.雀悦麻将项目

- 项目概述:安卓、苹果多平台的app游戏广东麻将, 前端使用cocos-js引擎 + ES6 + mobx构建, 后端使用pomelo框架, 数据库使用mongodb。

- 工作职责：负责前端所有UI的构建，游戏apk的打包。
- 遇到问题：
 - 1.开发周期只有一个半月，麻将的玩法规则不熟悉，ui功能较多。
 - 2.游戏内ui刷新频繁，事件处理繁琐。
 - 3.测试时，需要同时开4个apk客户端。
- 解决方案：
 - 1.反复试玩多个麻将游戏，深度组件化各个UI部件，以layer为每个部件的载体，做到单个部件可单独调试。
 - 2.独立逻辑、数据、UI三大部分，确立数据驱动的原则，建立一个mobx对象存储数据，通过mobx的特性，当监视的数据有变化时，自动运行mobx.autorun的UI操作。这样可以减少许多事件的监听和发送等处理。
 - 3.搭建web端测试地址和模拟数据，提高开发效率。
- 体验地址：<http://118.89.38.129:8080/?uid=200111>

自我评价

- 游戏玩的比较多，自信技术还不错。无奈游戏贵又搓，就想自己写一个。
 - 写着写着爱上了，编程成了新工作。莫看平日话较少，聊起项目停不了。
 - 初入公司人手少，前端后端一起搞。服务域名要维护，还要接入渠道包。
 - 前端游戏十多个，后端一套搞定了。累并快乐两年多，技术提升倒不少。
 - 单身多年爱文艺，另类程序没得跑。立志全栈工程师，崎岖求索路迢迢