Verity Tsai

(TSAI Hsien-Kuei)

Email: veritymissed@gmail.com
Phone: (+886) 983-216-098

26 years old, Male, Taiwanese.

Expectation

Apply for full stack software engineer.

I have been a web full stack developer about 4 years. Responsible for front-end UI design and the program implementation, back-end server (the MVC design pattern), APIs design (authentication/ data accessing/ pagination), NoSQL database operating, and some DevOps parts (CI/CD system upgrade/ Reverse proxy configs…) .

I am interested in the web-scale service design, want to participate in the product serving large quantity of users.

Education

B.A., Information Management, 2016, National Taiwan University, Taipei

Skills

Web development - full stack.

Front-end part :

RWD design and implement (HTML / CSS / LESS / Native JS / jQuery)
Front-end framework (AngularJS / ReactJS / Redux)
Design the UI flow, components, and the calling APIs processes.

Backend-part :

NodeJS / ExpressJS / EggJS / PM2 / API design / data modeling

Database :

SQL: MySQL / PostgreSQL / SQLite
NoSQL: MongoDB / Redis

DevOps :

Shell Script/ Reverse proxy (HAProxy or Traefik) / Crontab / Mongo clusters / Jenkins / Gerrit / Gitlab

Experiences

ucfunnel.com – 2017/9 - 2019/5

ucfunnel.com is a global RTB (Real Time Bidding) exchange - which means our customers are online advertising related industries.

As mentioned above, our business partners are global DSPs (such as Google/Yahoo) and many publishers
(such as news web media in Taiwan, Pixnet.net - the 3rd website on Alexa in Taiwan - which is a social media, JOOX - a music platform in China)

We are primarily engaged in ad exchange and SSP. We give our SDK to our publishers. Once publisher install our SDK on websites, our ad exchange will transfer the ads from our DSPs to that publisher according to the RTB specification.

The SDK will transfer the ad performance data into our Mongo database all the time, which is high QPS.

The DSPs and publishers can see the performance chart or download the report for revenues, costs, and other statistical data on our dashboards.

The experiences in ucfunnel :

DSP & Publisher dashboards re-design

I have renewed the UI flow and the layout for the dashboards, implemented the RWD (different style for different screen width by media queries) design.

DSP chart/report querying UI and backend API design

Design the UI for querying report with multiple parameters and time range, interval. Implement the backend controller for querying the mongoDB with Promise in JS asynchronously.

Authentication server design and implement

There are more than two dashboards for different user types in ucfunnel. There is one one data model for user, with one parameter ‘user type’ for distinguishing different types of user.

Our dashboards need authentication for the same user data collections in database server.

I design a SSO (Single Sign On) server for those dashboards and access our MongoDB cluster. This server is deployed on two VMs on our inner network. A I raised up a HAProxy server pointed to those VMs and configured the domain name.

CI/CD process optimizing

We choose Gerrit (hosting & code review) + Jenkins as CI system. Each time we push a commit to our own branch, or merge the commit into master branch on Gerrit, Jenkins would trigger the building process of that commit.

I redesign the process that Jenkins would run the process on the other VM in our inner network. That process would create a Docker container in another VM, building the commit in that container.

Automated process

Our DSPs and publishers can register the interval performance report (daily/ weekly/ monthly).

We use NodeJS's cluster module to run the report generating process in multi-thread mode.

I have implement the Crontab and run the program according to the intervals. That tasks will run the program, send reports via email, and send the success/failure notification to our slack group.

Experiences

Wavenet Tech – 2019/5 -

Advertising agency and marketing company.

Design and implement tools for other departments.

1. Marketing report system
2. Budget adjustment system

Marketing report system - performance report of Facebook/Google marketing Ad.

- Call the Facebook Graph API
- Call the Google Adwords API
- Aggregate the marketing data from different ad levels from Google/Facebook API .
- Format the performance data for the marketing analysts of other department .
- Display reports with visualization charts (D3.js) .
- Deploy with docker-compose, configure the reverse-proxy with Traefik and Clouflare domain service .

Related framework & libraries
- Egg.js/Mongoose
- React/Redux/Rx.Js(for async request)
- Material UI
- Egg's native MQ
- Docker compose

- Traefik as reverse-proxy
- D3.js

Budget adjustment system

Design for our clients. Manage the Facebook/Google ads from each level (campaign -> group or set -> ad) budgets and the automation of budget adjustment according to the conditions our clients' marketing analyst's configuring .

- Call the Facebook Graph API .
- Call the Google Adwords API .
- Optimize the UI flow for the budget adjustment of each level .
- Add and adjust the new columns in the reports and chart (D3.js) .

Related framework & libraries
- Angular 1
- Material UI
- ExpressJs
- Docker compose
- Traefik as reverse-proxy
- D3.js