Wang Xiaotao

<Senior Front-end Engineer>

Profile

Email:	wxt2005@gmail.com	Location:	Kanagawa, Japan
GitHub:	https://github.com/wxt2005	LinkedIn:	https://www.linkedin.com/in/wxt2005

Skills

- JavaScript
- Figma / Sketch
- CSS / Less / Sass / PostCSS
- Vue + Vuex
- Node.js
- Semantic HTML
- React + Redux
- Nuxt.js

Certifications

- JLPT N1
- TOEIC 950
- Qualification Certificate of Computer and Software Technology Proficiency - Programmer
- Qualification Certificate of Computer and Software Technology Proficiency - Software Design Engineer

Summary

Self-driven front-end engineer with 7+ years of experience in building web applications with React and Vue.

Passionate about user experience and constantly chasing the new possibilities of front-end technologies.

Using Node.js and server-side knowledge to help build a better front-end development experience.

Experience

Rakuten UI/UX Engineer

December 2020 - Present | Tokyo, Japan

Vue, Nuxt, Figma, Node.js, Docker, Kubernetes

- Develop Rakuten GORA (Golf Booking) applications.
- Collaborate with UI designers to create design guidelines on Figma and implement common components.
- Investigate UX best practices and provide design suggestions to UI designers.
- Migrate legacy PHP pages to Nuxt application.
- Monitor and improve page performance for better SEO ranking.
- Audit page accessibility and ensure primary functions are keyboard accessible.

- Handle operations for front-end servers and manage deployment scripts.
- Designed and implemented the CDN release process.
- Migrated Nuxt applications to internal Kubernetes clusters.
- Improved CI/CD's performance and reduced cost by using CI dynamic agents on Kubernetes.

Cainiao Network Senior Front-end Developer

August 2017 - January 2020 | Hangzhou, Zhejiang, China

React, Redux, Sketch, Node.js, Docker, MongoDB

- Built a bunch of single-page applications for customer-oriented services by my own.
- Collaborated with UI designers and created team's UI component library.
- Migrated legacy payment service to React and released without any critial bugs.
- Created team's front-end scaffold and constantly updated it to follow industry best practices.
- Designed and implemented architecture that enabled several applications to share payment process.
- Maintained front-end deployment system which can deploy applications by only several clicks.
- Implemented Word document conversion service for internal staff to publish online documents by one click.

DXY Front-end Developer

October 2014 - August 2017 | Hangzhou, Zhejiang, China

React, Redux, jQuery, Sketch, Backbone, Node.js

- Led development team of 3 members to develop medicine database service.
- Refactored 3 legacy PHP + jQuery applications to React + Redux applications.
- Designed JSON schema for the back-end to configure rendering of the page.
- Implemented forum rich-text editor image edit & upload plugin.
- Implemented image compression function for mobile image upload.
- Worked with designers, standardized and implemented the company-wise prefecture selector component.
- In charge of all webview pages inside the company's mobile application.
- Collaborated with mobile engineers and designed JS bridge protocol.
- Created an internal library for opening native iOS or Android application from webpage.
- Deployed company's private npm repository for internal package sharing.

Side Projects

Browser Opener

Swift, AppKit, CoreData

Browser Opener is a macOS application to help you open browsers based on URL rules.

As a front-end engineer, I usually use Chrome as debugging tool but use Safari for other kinds of stuff. It's painful to switch between browsers when opening different URLs, for example, localhost or 127.0.0.1.

Browser Opener can determine which browser to open when clicking links.

It started to support browser profiles in the latest version.

App Store:	https://apps.apple.com/app/browseropener/id1387480496
Product Hunt:	https://www.producthunt.com/posts/browseropener

Reading List

Swift, UIKit, Catalyst, CoreData

Reading List is an application for collecting web pages for later reading.

Most RIL applications require a user account and iPad support is often incomplete. For instance, none supports drag and drop back then.

So I created my alternative. It's a universal application that supports iOS, iPadOS, and macOS. All data is synced to iCloud and no user account is needed.

App Store: https://apps.apple.com/app/reading-list-read-later/id1501647738

Bangumi List

TypeScript, React, Redux, Node.js, Express, SQLite

Bangumi List is a Japanese anime information site.

Back in China, due to the fragmentation of copyrights, it was hard to check which anime is available on which video site. So I created this site to help Chinese anime fans to easilly watch their favorite anime series.

This site is also my playground for front-end technologies. It has been rewritten with several frameworks: plain HTML, jQuery, Angular, and React. Recently it's refactored to using TypeScript and added the login system.

Bangumi List:	https://bgmlist.com
Source Code (GitHub):	https://github.com/wxt2005/bangumi-list-v3

Education

- B.E. Civil Engineering - Zhejiang Shuren University 2009-2013