From Learning to Earning

Front-end Edition

Mohammad ali Ali panah - Spring 2023

Who Am I?

ZOOMIT

- Mohammad ali Ali panah
- Born in 2000
- 4 years working as software engineer(front-end, back-end)
- Zoomit org(currently leading team of 5), iGap messenger
- NIT Academy, React Workshop, Web programming TA
- Open source contributor(patches on lexical, dayjs, slatejs, …)

Chapter 1: Job

** I might be wrong but I have opinions **

This chapter is story telling with conclusion about my experiences

How I got my first job?

ZOOMIT

- First semester I got invited to play football. One of my friends said he gotten an offer from an startup.
  I said reach to me if you need a Front-end developer.
  TL;DR. They reached

- My experience at the time?

- NONE. C++(at university) was my first language.
  HTML, CSS, Bootstrap, JavaScript, React(less than 10h), git

- Generated resume with two or three projects

- The Company?

- It was trash at best but I was very exited and worked hard

How I got my second job?

ZOOMIT

- Summer of first year. jobinja.ir. Send resume to over 50 companies
- I'm still getting rejection emails
- I got couple of interview sessions and offers. which I chose iGap
- It was my breakthrough in software
- React and hooks, Reux, TypeScript, couple of Patterns and best practices, …
  Nodejs, express, mongodb, REST API, SOLID, …
- Multiple talented colleagues: Exposed to CI/CD, Docker, Kubernetes,
  Microservices, Kafka, …
- Very poor management and bad atmosphere
- Worked day and night for 10 months at the peak of my abilities

How I got my third job?

ZOOMIT

- jobinja.ir and in company websites. This time send resume to limited number of companies that I liked to work in
- Priorities: tech stack, atmosphere, pr, ability to grow
- I got 10+ offers from Zoomit, Alibaba, SabaIdea(Aparat, Filimoo), …
- Zoomit was and is the right move for me. Very chill and friendly
  If you feel the same join us https://zoomit.ir/hire
- Nextjs, Nx, Typescript, Monorepos, React query, Styled-components, Zustand, SEO, GitlabCI, …
- Agile, Jira, Leadership, Hiring, Project management, Testing, E2E
- First time getting raised

What is the trend? What am I trying to say?

ZOOMIT

- Have a plan for your near future and works towards it
- Small but continuous steps
- don't be too picky about the company at start
- Learn by doing
- embrace the challenge and work on hard tasks
- Work on side projects and learn new things
- It’s not always about coding . There are peoples to interact with or manage. Business to understand. Design and decisions to make

How you get your first job?

ZOOMIT

- Ask your friends to invite you to play football maybe!?
- jobinja.ir or other platforms(quera, jobvision, …)
- Connections are not a way into a company. They are a way to get easier interview and skip some process
- Internship programs or contests
- It might be hard at first but after a while the companies will reach you

When to know you are ready?

ZOOMIT

- As soon as learned most of the job requirements
- For Front-end after Git, HTML, CSS, JavaScript and React.js(6 month or 400+) - totally opinionated!
- Realistic: TypeScript, Tailwind, Nextjs, State management for higher chance
- Build multiple small projects and push the code with good commit messages to GitHub. The more unique project the better

شرح موقعیت شغلی

A glance at some of our engineering works:

- Building cutting-edge tools & infrastructure for our editors and consumer
- Modern and well-maintained mono-repo project with supporting tests.
- Designed and develop design system components from the ground up

We expect you to:

- Have Bachelor's Degree in Computer Science, Computer Engineering, or related field / equivalent experience
- Have good knowledge of computer science fundamentals (algorithms, data structures, and networks)
- Be Proficient in Web fundamentals (CSS, JavaScript, Web-API)
- Have an understanding of the Software Development Lifecycle (e.g., Requirements, Design, Development, Testing) and Exposure to Agile
- Be Experienced with one or more modern JavaScript libraries/frameworks (React, Next.js, Solid)
- Have a strong ability to work well with others in a highly collaborative team environment

Good to have experience with:

- Typescript and type wizardry (Conditional types, Template literals, Narrowing, etc.)
- Web performance and optimization (Core web vitals, bundle analyzing, Dynamic imports, etc.)
- Writing automated tests (Jest, Cypress, etc.)
- Docker and CI/CD Pipelines
- Rendering patterns and Next.js app directory
- Modern React ecosystem (TanStack Query, react-hook-form, Zustand, etc.)

Resume

- Good resume doesn't mean an offer. The only purpose of resume is to gets you an interview

- I reviewed 500+ resumes last year. I'm sick of seeing quera templates. Keep your resume very simple and focus on the content
  * Preferably single page
  * No photo of yourself
  * No starts. Don't show your ability with a few stars

- Stand on the shoulder of the giants. Read and learn from top employees in your field

- Bullets and bolding keyword are useful

- If you're just starting. Fill your resume with projects, university activities and course/ books. Don't mention your java exercise.

What to expect after you apply?

ZOOMIT

- Phone call for setting time
- Phone call interview or Home assignments
  * you can change the time of receiving home assignments
- Usually two or three technical interview rounds.
  - framework or technology
  - data-structure and algorithm
  * ask question - explain your thoughts - if you don't know just say I don't know but can I guess
- One HR interview round. Questions about your previous experiences. Your plan for future. Situational questions(like will you work overtime if needed)
  * think about common questions ahead of time
- Final interview with CTO. Might ask technical questions but usually for introduction and final decision
- Offer letter

University YES/NO?

ZOOMIT

- Concepts YES, execution NO
- Put time on things that make you eligible. You need to sacrifice something. A slight decrease in a couple of marks wouldn't harm you
- You can’t be jack of all trades. Focus on single fields. Companies pay for proficient
- You don’t know what you don’t know
  You don’t consider scale of users, legacy code, business, …
  If the technology for example Front-end seems easy for you, you should change your job. Simple blog post is front-end. Google map is front-end too

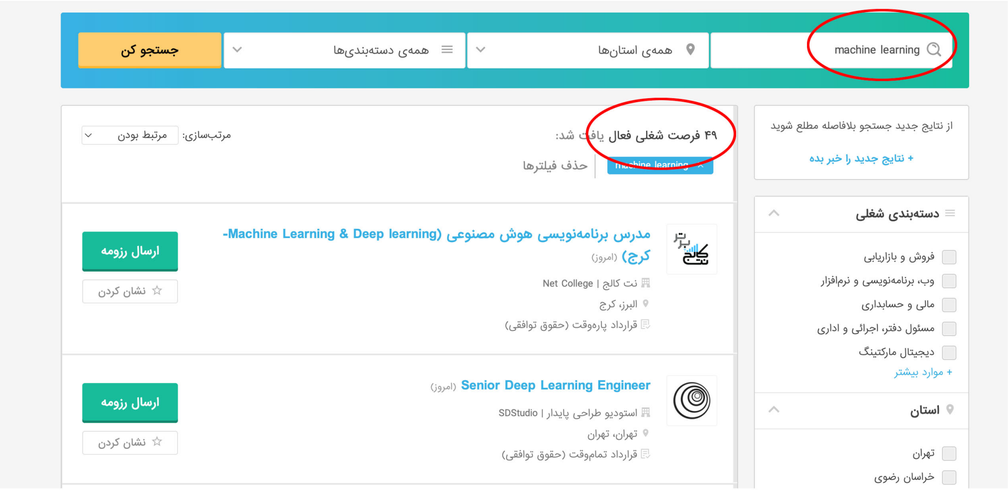
Conditions

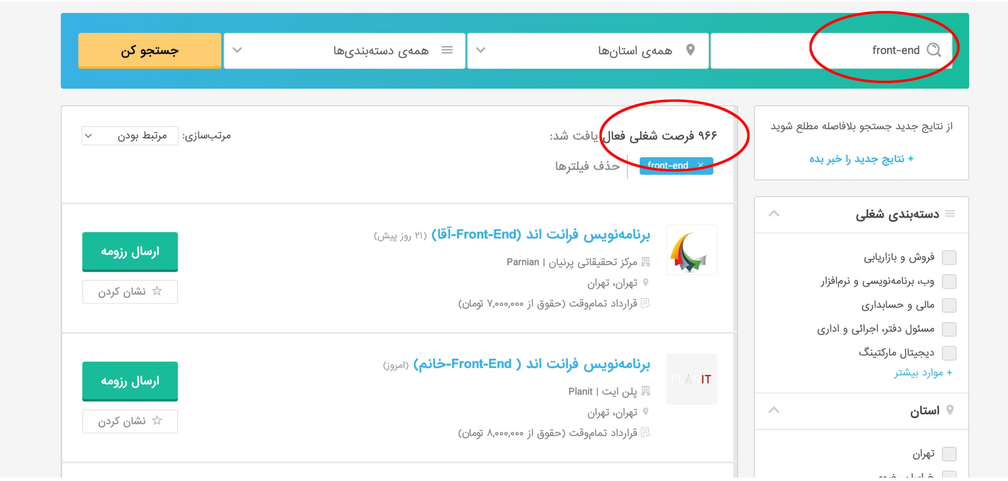
- AI and ML are the future. Rockets, Nanotech are too. If you a have plan to stay in Iran(don't) you should accept that we are consumer. Some fields require a higher education at top universities

- Iran job market is mostly around Web, Mobile, infra and analytics

- The income difference is not very big. Talking about rich and slightly richer but the job market might be in couple of years

- Chase your dreams and do what you passionate about but think and plan





Chapter 2: Web

Learning outside of the university

ZOOMIT

- University is easy-mode. I have never faced a question in my exams that the professor didn't talk about or mention it but the real world isn't like that. You should be versatile and react to problems and come up with a solution in reasonable time
- The difference is … it's not an exam. You can search. Read. Ask for help.
  Step into your journey with a fellow university student or friend and experience together
- Be patient reading article and docs
- Ask LLM(chatgpt, bard) to explain a concept or ask them to correct you. Read error messages and learn from them
- Read books and blog posts. Watch less video unless conferences
- Learn by doing. Always work on couple of side projects
- Watch short videos(YouTube) videos instead of long courses(30+ hours course is a hard commitment rather than an hour video)

- Git: you should know init, clone, status, add, commit, pull, push, branching, creating merge request but learn just a bit more. Stash, –amend commit, reset, rebase, merge, cherry pick.
  - www.atlassian.com/git

- Common data-structures and algorithm: array, hash map, linked-list, stack, tree, queue, priority-queue, heap, graph, tries. Sorting and searching.
  - frontendmasters.com/courses/algorithms
  - neetcode.io (solve leetcode.com by category)

- Get familiar with your editor and practice touch-typing:
  - www.typing.com
  - monkeytype.com

- HTML: elements(headings, ul, li, img, a, video), semantics, meta & seo, scripts tag

- CSS: box model, selectors, properties(margin & padding, font, color, border, lists), flex, grid, transition, animation, z-index, media queries and responsive design
  Learning tailwind is a really good idea

  - web.dev/learn/css
  - udemy.com/course/design-and-develop-a-killer-website-with-html5-and-css3/

Web Fundamentals (Cont.)

ZOOMIT

- JavaScript: Hoisting, Event Bubbling, Scope, Prototype, Shadow DOM and strict
  - javascript.info

- JavaScript(ES6): JavaScript new
  - es6 in hour: youtube.com/watch?v=NCwa_xi0Uuc

- DOM Manipulation: getElementById, innerText, onClick, …
  - www.youtube.com/watch?v=5fb2aPlgoys

- Forms: form element, inputs, onSubmit, FormData, actions, …

- Async & Fetch: asynchronous programming and remote data fetching
  - https://www.youtube.com/watch?v=vn3tm0quoqE

- React by far has the biggest ecosystem and job market.
  - https://www.udemy.com/course/react-the-complete-guide-incl-redux/
  - https://epicreact.dev/

- The docs are amazing
  - react.dev/

- Understand the need and mental model

- Just use vite and skip create-react-app
  - vitejs.dev

- Don’t waste time learning class-components. Skip it!
  Learn hooks and custom hooks

React Ecosystem

ZOOMIT

- State management: I recommend learning zustand but the company you join might need redux
  - https://github.com/pmndrs/zustand
  - https://redux.js.org/
  - https://redux-toolkit.js.org/

- UI libraries: Learn Material ui. It's really good idea to look the source code of some of these
  - https://mui.com/
  - https://github.com/nextui-org/nextui

- Tanstack query: helps data fetching states
  - https://tanstack.com/query/latest

- React-hook-form: helps building complex forms
  - https://react-hook-form.com/

- Just know these are exists
  - https://day.js.org/ - date helpers
  - https://www.framer.com/motion/ - animations
  - https://dndkit.com/ - drag and drop
  - https://zod.dev/ - schema validation

- Picking up TypeScript for university students shouldn't be hard task as they already worked with typed languages but don't judge too soon. TypeScript is a beast. It's very powerful and complex for library maintainers but for everyday use it's lovely
  - https://www.totaltypescript.com/tutorials/beginners-typescript
  - https://www.totaltypescript.com/tutorials/react-with-typescript

- Interfaces, types, utility types. Generics, classes, keyof, typeof, narrowing, indexed access, conditional types, mapped types, template literal

- Next.js is the framework for React. The docs and learn section is amazing
  - https://nextjs.org/learn/foundations/about-nextjs
  - https://nextjs.org/docs
- Next.js is in a weird place right now. I recommend ignore all the new stuff and head to Pages router doc
- Routing, rendering and data-fetching(ssr, ssg, isr, csr), middlewares, Image and Link components

Go Beyond

ZOOMIT

- CI
- Monorepo
- Bundeling
- Logging
- Performance Monitoring

End

Thanks everybody
esp. ygn <3