

# Jacob Zwang

[zwang.jacob@gmail.com](mailto:zwang.jacob@gmail.com) — <https://zwa.ng>

Los Angeles, California

## Profile

Advanced web applications engineer with 3+ years programming experience built on top of 10+ years of technical experience in hard surface 3d modeling and precision vector graphics. Capable working full-stack, most experienced in frontend. Excel at solving difficult architectural problems and comfortable working with cutting-edge tech that has little to no documentation or guidance.

Have design background; Achieve best results when working in place of, closely with, or alongside designers. Passionate about finding simple and elegant solutions to both design and engineering problems.

Love working with newer tech like SvelteJS, CSS Houdini, and WebGPU, but comfortable working with industry standards like ReactJS, Bootstrap, and WebGL.

## Key Skills

- Finding new solutions to old problems
- TypeScript, JavaScript, HTML, CSS, WebGL, WebGPU, and other web tech
- Unification on projects and ideas
- Ideation and brainstorming
- Fast learner
- Teaching others
- Quality control
- Forward-thinking

## Job Experience

### Product Architect, Co-Founder

Backyard Events, 2020 - 2022

- Designed UI (user interface) in Figma and during development.
- Led development of user interface in SvelteJS (SvelteKit) and established front-end architecture.
- Built custom Svelte transitions and router.
- Built reusable and SSR (server-side rendering) compatible UI components in SvelteJS (JavaScript, TypeScript).
- Built development environment including PostgreSQL, NodeJS, and Redis in Docker Compose.
- Built multi-stage Docker container for production deployment on AWS.
- Built session middleware using TypeScript (JavaScript), SvelteKit, and Redis.
- Built REST API endpoints using TypeScript, pg, and SvelteKit.
- Built automated testing in Cypress.
- Contributed to CI/CD (Continues Integration, Continues Development) in Github Actions, AWS CodePipeline, and Husky.
- Switched raw SQL queries to PgTyped queries for automated TypeScript interface generation.

### ThreeJS Developer

Genopets — Monarch Gaming, 2021

- Built ThreeJS abstraction layer in JavaScript for swapping vertex buffers in an out of components.
- Built animation phase abstraction using Tween.js for blending between models.
- Built Blender to ThreeJS automation pipeline in Python.

- Optimized GLTF 3D models for the web in Blender.
- Designed, modeled, and animated example "Genopet" for artists to reference.
- Wrote documentation for recreating efficient and aesthetically pleasing 3d models for the web.
- Built custom glow shader in GLSL.

### **Full Stack Web Developer, Salesforce Automation Engineer**

Alden Pacific Investments, 2021

- Lead development of custom questionnaire and calendar form with Salesforce integration.
- Taught Alden Pacific Investments internal engineer how to maintain the application.
- Set up CI/CD using Github and Google Cloud Build.
- Built Docker Compose development environment and production Docker containers.
- Mail automation using Mailgun and JavaScript.

### **Full Stack Web Developer**

Gustav Carroll, 2021 - 2022

- Designed and discussed the website with the client using Figma.
- Built development pipeline CI/CD (Continues Integration, Continues Development) in Google Cloud Build (GCP).
- Hosted Git repository on Google Cloud Source Repositories.
- Stored images, product index, and website backups using Google Cloud Storage.
- Built frontend UI in SvelteJS (SvelteKit) and TypeScript.
- Wrote scoped and global CSS styling.
- Built automated response contact form in Mailgun and TypeScript.
- Hosted website on Google Cloud Run (GCP).

### **C Programming Tutor**

Self Employed, 2020

### **Event Coordinator, Web Content Manager, Video Editor**

Yale Open Climate, 2019 - 2020

## **Personal Projects**

### **Organic Cluster Communication**

- Built high-performance metaball user interface in WebGPU and WGSL (WebGPU Shader Language).
- Rendered core user interface in fragment shader using only 2 triangles (to cover screen).
- Wrote buffer memory layout in TypeScript.
- Wrote isosurface extraction algorithms in TypeScript.
- Built video and audio data tunnels in WebRTC.
- Built WebRTC signaling server in SocketIO.
- Built complementary user interface in SvelteJS.
- Built REST API in SvelteKit and TypeScript.

### **CSS Framework**

- Built general purpose CSS framework in SCSS (for use in my own projects).
- Built custom superellipse corner rendering in CSS Paint (CSS Houdini).

- Incorporated tailwind for use of their utility classes alongside my semantic classes.

### **Duo Mono**

- Built chrome extension to automate Duo 2 factor authentication.
- Used Web Crypto API to generate HMAC tokens in the browser.
- Stored client secret in Chrome Storage Sync for cross device support.

## **Community Work**

### **BalotNav**

- Helped build initial prototype in ReactJS and JavaScript.
- Designed icon used in logo.

### **Lucky Parking**

- Built heatmap in MapBox and ReactJS.
- Provided design feedback and consultation.

### **[confidential internal tooling]**

- Built user interface in VueJS.

## **Certifications**

### **LinkedIn**

- Frontend Development
- Google Cloud Platform (GCP)
- HTML
- Git
- Cascading Style Sheets (CSS)
- JavaScript
- Python (Programming Language)
- Adobe Photoshop
- Adobe Illustrator

## **Skills and Tools (Extensive)**

*see next page*

## Design Skills

- ★ Branding
- ★ 3D Modeling
- ★ UI/UX
- ★ Grid Systems
- ★ Infographic
- ☆ Form
- ☆ User Story
- ☆ 2D/3D Animation
- ☆ Typography
- ☆ Architecture
- ☆ Video Editing
  - Compositing
  - Automotive

## Design Tools

- ★ Blender
- ☆ Figma
- ☆ Premier
- ☆ Photoshop
- ☆ Illustrator
- ☆ PowerPoint
- InDesign
- Miro
- Canva

## Graphics Programming Tools

- ☆ GLSL

## Programming Skills

- ★ Full Stack Architecture
- ★ Web Development
- ★ Advanced Type Definitions
  - ★ Generic
  - ★ Mapped
  - ★ Conditional
  - ★ Indexed
- ☆ RESTful API Design
- ☆ Modularity
- ☆ Packaging
- ☆ Semantic Versioning
- ☆ Modularity
- ☆ Schema Modeling
- ☆ Key Value
- ☆ Document
- ☆ Relational
- ☆ End to End Testing
- ☆ Unit Testing

- ☆ Build Pipeline
- ☆ Web Scraping
- ☆ Continuous Integration
- ☆ Mail Automation
- ☆ DNS Configuration
- ☆ Signaling Server Design
- ☆ Graphics Programming
  - Push Notifications (Web)
  - Data Visualization
  - Jamstack
  - Code Generation

## Workspace Tools

- ★ VSCode
- ☆ Extension Development
- ★ GitHub
- ☆ Actions
- ☆ Organizations
- ☆ Access
- ☆ SSH (Client & Server)
- ☆ Linux
  - Visual Studio

## Web Programming Tools

- ★ CSS
  - ★ Advanced Selectors
  - ★ Media Queries
  - ☆ Tailwind
  - ☆ Animation
  - ☆ SCSS / SASS
    - Stylus
    - Bootstrap
- ★ JavaScript (ES2022)
  - ★ TypeScript
  - ★ ESBuild
  - ★ Rollup
  - ★ Vite
    - Babel
    - WebPack
- ★ Svelte
  - ★ SvelteKit
  - ★ Sapper
- ★ HTML
- ☆ SVG
- ☆ DOM API
- ☆ ThreeJS
- ☆ WebGL / WebGPU
- ☆ SocketIO
- ☆ WebRTC (Audio Video)

- ☆ Vue
- ☆ Jest
  - GraphQL
  - Hugo
  - Web Assembly (Rust)
  - TensorFlowJS
  - React / JSX
  - Angular
  - Yew

## Cloud Programming Tools

- ★ Docker
- ☆ Google Cloud
- ☆ Cloudflare
- ☆ Mailgun
- ☆ Cypress
- ☆ Caddy
- ☆ Let's Encrypt
  - Kubernetes
  - AWS
  - Pion ION

## Other Programming Tools

- ☆ Git
- ☆ Python
- ☆ SQL (PostgreSQL)
- ☆ ElectronJS
  - OpenAI
  - Dart (Flutter)
  - Swift
  - Scala
  - Lua
  - Go
  - C#
  - Elm
  - Haskell
  - F#
  - Windows APIs
  - Idris
  - Common Lisp

## Systems Programming Tools

- ★ NodeJS
- ☆ Rust
- ☆ C
  - Java

## Other Skills

- ☆ Competitive Analysis
- ☆ 3D Printing (FDM)