

TRAVIS WILSON

737 Haight Ave.
Alameda, CA 94501

(510) 684-4365
trav@majorleaguemarket.com

In brief

For the past twenty years I've built the architecture, code, and development lifecycle of web-based software. My clients and employers have ranged from startups seeking prototypes to Fortune 500 companies developing enterprise solutions. My development is tempered with a solid understanding of product design, user interface, and data modeling. My team management is based on a visible, realistic, ambitious development process, and respect for software developers as collaborative individuals.

My goal is to get your product built (via hands-on coding), production-quality (ensure decent test framework), and maintainable (offer architectural guidance and design along the way).

Weapons of choice: Python · Javascript · Java · bash · jQuery · Spring · Django · MySQL · Good source control habits

Inspirations: Jeff Atwood · Paul Graham · Vijay Vaitheeswaran · Steve Yegge · Alan Carter · Wayne Gretzky's dad

Experience

Google · San Bruno, CA · *Engineering Productivity*

June 2014 to October 2015

YouTube developers frequently built their own test or development environment from scratch, mocking someone else's complex services one function at a time and writing their own extensive test case data. Engineering productivity includes minimizing the time a developer must spend doing that kind of work, so the developer can focus on his own code.

So I wrote reusable mocks, tests, and test specs myself when appropriate, but more often I packaged someone else's good mocks, tests, or specs for broad reuse. My own product was a test spec framework and processor to streamline reuse. Test coverage went up; code got more legible.

Microbattle · *Creator* · Please see <http://microbattle.com>

2013 to 2014

Microbattle is a software engine that lets users play games by writing their own software agents to play the game. It's also a website that makes it easy for users to write those agents online (or upload after writing them offline), and play games against other users' agents.

The game APIs, available in multiple languages, incorporate yields and asynchronous events so that agent code is easy to write and to read. Two games – a real-time battle simulation and an iterated prisoners dilemma – demonstrate the concept on the site. All of this is my original work and has pushed my definition of “full stack developer” to its extreme.

The Walt Disney Company · Palo Alto, CA · *Platform Architect and Developer*

May 2009 to December 2012

Develop Disney's internal analysis platform. It's an asynchronous high-volume data logging system, coupled with collaborative web-based analysis that encourages users to share their queries across Disney. It engages non-analyst users by showing insightful results, offering next steps (*e.g.* “should we alert you when X happens?”) from directly within the resultant data, allowing users to embed visualizations in their own documents, and revealing the queries' source code.

When Disney bought Playdom, an online game startup, in 2010, I inherited the product. I worked on systems architecture, software development, and customer relations. I added features, improved security and reliability, reduced end-to-end data latency from 15 minutes to sub-second, and grew the product throughout the Disney organization – increasing its customer base by 150%.

My coordination between teams was generally peer-to-peer and meritocratic – design it well and they will come. I introduced and directed SDLC improvements such as deployment scripts, versioned components, and a configuration layer. I personally recruited six employees and two contractors onto my teams at Playdom/Disney.

Experience (continued)

Simple Dynamics · Oakland, CA · *Partner*

2003 to 2009

I was a founding partner in this software consulting group, specializing in early-stage development for startups. Simple Dynamics helped jump-start development often before the startup had its team in place. We led by example: delivered modular software, established visible and predictable development cycles, and so forth.

I developed business, mostly through repeat customers and referrals. I staffed projects by selecting from a small network of trusted software developers – we wouldn't accept the project if the developers weren't a match. And I led projects – by drafting interfaces, mentoring where necessary, and writing lots of code. We shipped on time and on budget. Here's a representative sample of the clients and projects I led:

Location Labs (acquired by AVG Technologies)

I led a three-developer team to create a simple e-commerce site dedicated to selling a single handheld device for Location Labs' new venture. The device required extensive backend integration with three different, unrelated third-party services. I wrote specs for the interaction between services and for the website.

Siri (acquired by Apple)

The server/client codebase had grown unwieldy and monolithic as this company scrambled to meet market demand. I led a two-developer team to redesign and refactor web application code to keep layers of functionality separate, meanwhile developing new features to meet business deadlines.

Trion World Network (still producing games independently)

I created an online store application to sell virtual items within a game world using the game's currency. The store app accommodates multiple games simultaneously and provides extensive administrative tools. I designed and documented the product, and managed a second developer. We both coded.

Adaptive Path (acquired by Capital One)

Leading user experience design firm needed a functional, aesthetic prototype website for a financial institution, to give users a friendlier way to manage their mutual funds and financial outlook. I wrote the functional code, and hired and managed the HTML coder on the project.

Facetmap · *Creator* · Please see <http://facetmap.com>

2002 to 2007

Originally a demonstration of a visual interface to navigate faceted data, this became a software product in its own right – as a web-based service and as downloadable software. This was a solo project; work included the SaaS portal, the software engine, and sample UIs to demonstrate its use. Facetmap has distinct recognition among information architects, is taught in university courses, and has been licensed to other software companies.

Online Alchemy · Austin, TX · *Software Architect*

2003 to 2004

This independent game company envisioned an artificial intelligence product, which I helped bring from concept to marketable prototype. I worked with the CEO to develop a new AI model realizable in software, and managed a development team of three. Also designed production architecture, where the team's modular components worked together in an efficient, asynchronous environment.

Earlier work experience information is available upon request.

Education

UC Berkeley · Berkeley, CA · *B.S., Electrical Engineering and Computer Science*

1990 to 1994