# Jason Horsley - Developer

I am a web developer based out of Austin Texas. I have been writing code full time since 2012. Most of those years have been remote, working by myself or in a very small team.

I can effectively communicate with designers, management, or end users. I can also take decent personal notes during meetings and collaborate with third party companies. But I am at my best and happiest when writing actual code to solve the actual problems at hand.

I enjoy VS Code as my editor, Linux as my OS, Agile as a loose philosophy, and any reasonable MVC or MVVM framework.

## Main Skills

### Frontend code

- Vue
- React
- jQuery
- Webix

### Frontend design

- Tailwind
- Bootstrap
- Bulma
- Scss

### Mobile & Desktop

- Vue Native
- Cordova
- Livecode
- Electron

### Backend code

- Laravel
- Wordpress
- EmberJS
- MySQL / Mongo

### Infrastructure

- AWS / G-Suite
- Apache / Nginx
- MailChimp / Iterable

### Soft skills

- Git
- Agile
- Shell

- Paypal / Synapse
- Vim

## Professional experience

### MicroVenture, Web Developer

MicroVentures is an Austin FinTech company that sells investment opportunities to their active user base of 100,000+ investors.

I am currently employed as a full stack developer, working on any and all company code. The MicroVentures codebase consists of a client facing investment platform, an in house management console, a native IOS app, an electron based 'secure document storage' desktop application, a covid relief charity site, a zendesk help center and a wordpress marketing site.

Most of the projects are Laravel and Vue, one is Scala and React, all communicate with a centralized MySQL database via a dedicated API server. Our CSS libraries are fragmented between Bootstrap, Bulma, Tailwind, and handwritten scss. We are attempting to unify to a single tailwind install. We have excellent back end test coverage, and are working to develop a cypress driven front end test suite to match.

Early 2019 - Present

### Pubbly, Lead Developer

Pubbly was a startup bankrolled by CCI-NY. The original vision was a simple desktop application to be used by in-house artists and authors, to modernize much of the parent organization's content library. After initial development, the tools were used to create a winning entry in the Global Learning Xprize competition. The Xprize was a contest to see who could best teach Swahili literacy to native Tanzania children using android tablets. We won a cash prize, and were able to field test our submission during an 18 month trial run.

I worked to create and improve the Pubbly code base from day one. Their content authoring environment was a desktop application written in LiveCode. It exported XML files which could be read and interpreted as "interactive books" via our javascript engine. The books were managed by a custom LAMP content management system. That in-house management tool was used to structure, map, and bundle each "interactive book" into cordova wrapped android applications, and was later retooled to manage external Laravel "marketplace" applications we sold to partner companies.

I managed two other full time developers, and was responsible for all production code

2015 - 2019

### CCI-NY, Developer

CCI NY is a traditional publishing house. Their main source of revenue continues to be the licencing of their proprietary educational content, and the sale of the physical books.

I was hired as a general web developer in their attempt to modernize. I worked on a project they did for McGraw Hill converting old flash based games to HTML5. I acted as their site manager, maintaining and improving their wordpress based homepage. I also wrote a web app to convert raw textbook lesson+quiz modules into SCORM compliant javascript quizlets.

2012-2015

## Projects

### MicroVentures: Stack separation

MicroVenture's core product, a client facing investment platform, was partially fragmented between a full stack Laravel 4.2 and a backend API Laravel 5 server. The older project accessed the newer's Eloquent models and database via a custom private composer SDK running guzzle commands. The SDK was an attempt to partially rewrite the backend without having to fully finish before pushing to production, but had been in the "temporary fix" state for quite some time.

My responsibility was to fully retire the Laravel 4.2 front end, served mostly through blade template, by adapting the existing HTML and styles into a standalone Vue2 SPA which would talk directly to the API server.

The project was much larger than originally anticipated. We were forced to change our authentication methods from the somewhat built in session based laravel way to an oauth generated token, rewriting much of the previous middleware. A lot of the old "front end" made assumptions about always having fresh data, so we needed to be meticulous in our implementation of the vuex store. And while some of the API calls were "working", the fact that it now ran on client machines meant that API calls needed to bundle together to avoid long waits while sequential ajax requests resolved

The project did finish and has been live for some months. I am proud to say it significantly reduces page to page load times, and code overhead for our team

### MicroVentures: Future acquisition integration

MicroVenture's new business model is in acquiring third party projects and integrating them into the existing ecosystem. Adapting to this change required a large restructuring of the base user model. Instead of people signing up as "investors" to make investments, they signup as "customers" with "invest property access". While the change seemed somewhat pedantic at first, it soon became clear that it would require much fiddling.

We decided to route all logins and singups through a "customer" subdomain, allowing for the shared functionalities of account management and preferences to be accessible from any external site. We then created a private NPM package, so that each independent project would start from the same base "shell", abstracting away default server error handling and unifying our common components to create a better brand image.

The work paid off, and our recently acquired "digital will" code base is now in use by our beta test group of customers. While it was done in react with a scala backend, it is able to share the same customer model and authentication logic.

### Pubbly: Javascript engine

Pubbly's main deliverable asset was a javascript based interactive book. It somewhat resembled the old "escape the room" flash games, but more polish. Each "book" had pages filled with user interaction zones, requiring a user touch gesture or a drag drop. Each zone would trigger a sequence of timed events such as animating an image across the page, playing an audio file, starting a video, or manipulating the visible objects.

As the project progressed, the authors and artists designing the books continued to push the envelope. They would create extraordinarily large books, packed with hundreds or even thousands of individual audio files. And since they were developing in a desktop environment, they were blissfully unaware of potential load times.

We used a number of different tricks to keep things under control. We concatenated all the audio files in a given book into a single file, and 'seeked' to different locations to play each unique sounds. This cut down the number of network requests needed, and solved an additional audio timing issue in safari. We also carefully chose which assets to load next based on what pages could be quickly accessed.

I won't claim it was a perfect system. In hindsight, we should have simulated network load times from within the authoring tools, or simply worked more on communicating browser based limitations to the rest of the team. But it did get the job done, and I am happy to say that Pubbly is still using much of my original codebase.

## Personal details

I am married and the proud father of a 3 month old son. I am interested in a career that provides stability for my family as well as new and interesting learning opportunities in this ever changing field. I play church pipe organ, enjoy wood working, and love long and overly-complicated board games.

Name: Jason Horsley

Email: Jason@JasonHorsley.tech

Git: JasonHorsleyTech

Phone: 512-202-6952