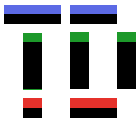

Tim Doerzbacher

Software Engineer

412-758-0179
tim@tim-doerzbacher.com
tim-doerzbacher.com

Skills

- Agile Development
- LAMP Environments
- Linux Administration
- Database Design
- Graphic Design

Operating Systems

- GNU/Linux
- Mac OS X
- Windows

Languages

- Bash
- C/C++
- Erlang
- HTML/CSS
- JavaScript
- LESS
- PHP
- SASS/SCSS
- SQL
- Yaml

Technologies

- Ansible
- Apache
- Bootstrap
- CodeIgniter
- DataTables
- Durandal
- FreeCAD
- Git
- Gitlab
- Jira
- jQuery
- Knockout
- MariaDB
- MySQL
- Nightwatch.js
- NodeJS
- PagedJS
- PHPUnit
- PostgreSQL
- QUnit
- Raspberry Pi
- React
- RequireJS
- Symfony
- Ubuntu
- WebDriver
- WordPress

Noteworthy Accomplishments

- I always clean up code bases and enjoy writing comments. There is no such thing as "self documenting" code.
- Creation of continuous integration and deployment pipelines in Gitlab. Reduced the release process from several hours of work, down to a couple seconds typing a single command and followed by 30 minutes of waiting.
- Overseeing migration and rewriting of legacy CodeIgniter code to a more modern Symfony/Laravel based project. All the controllers and routing remained in CI, while large swathes were hollowed out and replaced with calls to PSR-based code using Composer libraries.
- Replaced legacy Tablesaw based list views with DataTables. Event list views were previously hard coded into the application. The new architecture allowed users to configure everything. I later added custom in-place editors for most table cells to allow users to quickly edit rows without needing to open the event editor.
- Created new flexible and user customizable view-based rendering pipeline for easily filtering and transforming data into various formats. The new rendering pipeline now supports output in JSON, CSV, and PDF. This allows the same code path to handle feeding data to the DataTables based event list, and the report generators.
- Created new user customizable report functionality for events. Replaced hardcoded system-wide and limited user reports with a visual report editor and a more flexible backend.
- Completely re-implemented PDF generation to overcome existing limitations and to allow customers more flexibility in the layout of their documents. This was to remove an abandoned project (TCPDF) that had caused innumerable problems. Now all clients have near unlimited control of page layout and branding.
- Created new release process to automatically generate and distribute assets in various package formats (NuGet, WebJAR, etc.). My team provided a shared UI toolkit internally for multiple teams, all using different technologies. Automating the release into formats suitable for myriad development environments saved all of us time and effort.
- Implemented multiple visual regression systems. The first visual regression system I setup was used to spot hard crashes in toolkit demos. The second time was for prevention of regressions in PDF generation.
- Lead the project to divaricate a large, monolithic toolkit into independent projects. This was to mitigate frequent regressions and compatibility issues with downstream teams.
- Created an application to collect employment applications along with signatures. This was Flash based and way before it was cool to digitally sign documents.
- Created a privacy respecting user matching algorithm for a messaging app. The algorithm that I developed would normalize and hash contact information in a user's address book. This hashed data was then used to match new friends.
- Designed and implemented a new product that would allow multiple DVRs to archive and sync video data between nodes. The system was designed to have a large array of disks to store all the data in a central location. It was then indexed to allow quickly searching, sharing, and restoring to any DVR.
- Created a customer service and inventory management software. This software was designed to prevent unauthorized copies of the ETC system while allowing sales and support to upsell new features.
- Designed a video packaging format to allow exchange of data between DVRs. We needed a way to store up a dozen or more video streams along with other event data and other metadata. This would then allow easy backups and transferring of recording sessions.

Professional History

- Senior Software Engineer @ Opus One Interactive 2016 – Now
- Front End Engineer / Release Engineer @ Intermedix 2014 – 2016
- Subcontractor @ PiCon Web Design & Marketing 2007 – 2015
- Full Stack Developer @ Walking Thumbs 2013 – 2014
- Server & Web Developer @ Kb Port 2008 – 2013