

# Jacob T Fischer

[jacob.t.fischer@gmail.com](mailto:jacob.t.fischer@gmail.com)  
(636) 443-5129

[github.com/JacobFischer](https://github.com/JacobFischer)  
St. Louis, MO.

---

*I seek to temper a passion for technology by always experimenting with the latest the industry and academia have to offer through programming; specifically in software engineering, systems design, and web development.*

---

## Education

Master of Science

### Computer Science

[Missouri University of Science and Technology](#) – 2015 - 2017

Graduate work in Software Engineering & AI development, research, and implementations

Bachelor of Science

### Computer Science

[Missouri University of Science and Technology](#) – 2008 - 2012

---

## Experience

ePlata, LLC.

[eplata.com](http://eplata.com)

### Software Developer – 2017 - Current

Startup working primarily in Full Stack development using Node.js to make a cross platform financial app leveraging Cordova for Android, iOS, and Web platforms

- Full stack JavaScript/TypeScript development and DevOps for front end, back end, and cloud infrastructure
- Modern JavaScript, Babel, React, and Material Design via Material UI framework and styled components for the front end
- Node.js, Express, Redis, GraphQL, and MSSQL on the back end
- Cloud stack utilizing Amazon Web Services (AWS) with multiple ECS Docker containers across regions monitored with Sentry, Logz.io, and builds using AWS CodeBuild
- Wrote Webpack plugins to improve developer workflow by automating build steps and improve performance by moving runtime costs to build time
- Updated Express REST APIs to GraphQL queries for improved performance, type checking, and code quality
- Improved server and client speed by chunking build and utilizing loadable code and components
- Modernized JS code to use async/await syntax and Flow/TypeScript types
- Offloaded multi-lingual strategy to build time string compilation instead of run time string building
- Rewrote server-side rendering to stream responses to improve response time and pre-cache most highest impact queries
- Performed DevOps tasks such as improving continuous integration, deploying builds, and improving development workflows
- Wore many hats and did much outside a traditional software engineering role being a small startup with big ambitions

## Software Engineer – 2013 - 2014

Developed many different event management programs across utilizing web and Microsoft technologies

- Assisted with multiple larger teams to update deprecated WinForms software to VB.Net Silverlight full stack
- Completed performance profiling to identify slow forms and virtualize them
- Worked on small team to quickly create and deploy new event registration sites for clients with a variety of needs
- jQuery, JavaScript, HTML, CSS, and a proprietary framework for the front end
- ASP.Net back end using VB.Net, C#, and MSSQL

---

## Skills

### Programming

JavaScript (ES6+), TypeScript, HTML, CSS, SASS/SCSS, Python 2/3, JSON, SQL, C++, C#, Java, Go (Golang), YAML, Lua, RegEx, Shell, VB.Net  
*always eager to learn many more!*

### Frameworks

Android, iOS, Node.js, npm, React, Babel, Material UI, Webpack, Lerna, Mocha, Jest, GraphQL, PixiJS, Express, jQuery, Lodash, OpenGL, Moai, CodeIgniter, Wordpress, ASP.NET MVC, Silverlight

### Databases

MySQL, Microsoft SQL Server (MSSQL), SQLite, PostgreSQL, MongoDB, Redis

### Project Management

Git, GitHub, Trello, Jira, Slack, Travis CI, Circle CI, Jenkins, DevOps, OKR

### Other

Microsoft Office, Google Docs; Adobe's PhotoShop, Illustrator, and Premier

---

## Technical Activities

### Cadre AI Framework

[siggame.io/Cadre](http://siggame.io/Cadre)

Master Degree software engineering project(s), with continuing development and enhancements to this day

Designed and implemented a new framework for creating and playing games with AIs across a variety of programming languages and platforms

- Created AI clients in C++, C#, Go, Java, JavaScript, Python, Lua, and TypeScript that all must interact with JSON, networking sockets, reflection, threading, and documentation systems
- Game server implemented via Node.js with JavaScript/TypeScript and Express facilitating TCP/WS/HTTP connections over clustered threads
- Visualizer to playback game log files in browser using PixiJS, Webpack, jQuery, TypeScript, and SCSS
- Python metaprogramming tool to speed up development of common game logic between all projects via code generation and mutation
- Battle tested in multiple programming competitions with a wide range of games and competitions, some with human player(s)

### npm Packages

[npmjs.com/~jacobfischer](http://npmjs.com/~jacobfischer)

Maintains a set of JavaScript, TypeScript, and C++ (native) packages for Node.js

- Maintains a variety of packages across domains
- Performs expected maintenance, including addressing issues, merging pull requests, and publishing new releases
- Also contributes to other open source packages such as [DefinitelyTyped](https://github.com/DefinitelyTyped/DefinitelyTyped)