George Steel

647-771-5414
george.steel@gmail.com
github.com/george-steel

Technologies and Languages

Languages: Go, Rust, C++, C, JavaScript, Typescript, Haskell, Python, SQL, TEX

Technologies: FIDO2/Webauthn, MFA, OAuth, Vulkan, Lit, WebAnimations, Blink, WPT testing, W3C specs, Postgres, CockroachDB, GTK, EPUB, Flask

Education

2010 – 2016 **University of Toronto**, *Hon. B. Sc. in Mathematics*

- Minors in Computer Science and Biology.
- Graduated with High Distinction (GPA 3.9).

Employment History

Dec 2024 - Feb 2025 **Software developer**, Pinkbot

- Created a new, automated billing system that replaced a previous process involving multiple, time-consuming manual steps per client.
- Helped to create a new authentication system for our merchant portal.

Apr 2021 – Jun 2023 **Software Developer**, LoginID Canada

- Created an OAuth2.1/OpenID Connect identity provider supporting configurable multi-factor authentication methods, including passwordless FIDO2, SMS, TOTP, and others.
- Led multiple refactoring projects to reduce tech debt and improve performance in related services, eventually architecting a move to a single-binary to reduce microservice overhead as well as helping to architect a rewrite of our core FIDO2 library in Go.
- Created a custom high-performance database client library for our Go services based on sqlx and pgxpool. This allowed queries to be performed in all transactional modes supported by CockroachDB (including the fast batch mode unsupported by database/sql and existing ORMs) with the results automatically mapped into slices of structs.
- Led code reviews for a large portion of our software stack.

Jun 2019 – Nov 2020 **Software Developer**, *Chrome Animations/Interactions*, Google Canada

- Helped complete and launch the WebAnimations API in Blink (the rendering and Javascript engine powering Chrome and Electron used on over 3 billion devices), giving a unified Javascript interface to declarative animations from all sources using an invalidatable keyframe model.
- Contributed a section to the WebAnimations spec (including the associated cross-browser *Web Platform Tests*) allowing the creation and manipulation of animations targeting pseudo-elements.
- Contributed numerous bugfixes and optimizations across the Blink animations stack (written in C++). This included allowing percentage transform animations (sidebars and some popups) to run off-thread, allowing them to run more smoothly despite the actions of other scripts on the same page.
- Participated in the Chrome Interactions bug triage rotation, which is the first point of contact for game developers (and others) dealing with possible engine problems.
- Contributions at `https://chromium-review.googlesource.com/q/owner:gtsteel%40chromium.org`

Nov 2017 – Feb 2019 **Full-stack Software Developer**, *Satsuma Labs*

- Created a prototype mobile application using a Haskell backend and a React-Native frontend.
- Developed a number of open-source libraries furthering the Haskell web service and react-native ecosystems, including a spatial indexing layer which we used with CockroachDB. (Available at `https://github.com/SatsumaLabs`)

May 2016 – May 2017 **Software Developer**, *Prof. Peter Jurgec*, Linguistics, University of Toronto

- Created browser-based educational software used in introductory phonology courses.
- Rewrote and further developed a research tool which uses a maximum-entropy machine learning model to analyze the relative frequency of sound patterns in speech based on sample text and generate random pronounceable gibberish based in the inferred constraints.

Sept 2013 – Mar 2014 **IT Assistant**, *ENAGB Youth Program*, Native Canadian Centre of Toronto

Created a responsive website for the ENAGB program (featuring a dynamic events calendar) along with a variety of promotional materials (posters, brochures, business cards, etc.) for the program.

Summer 2011, 2012 **Summer Research Assistant**, *Prof. Gilbert Walker*, Chemistry, University of Toronto

Performed spectroscopy and microscopy supporting research into creating nanoparticle based markers for medical diagnostic use, improving the sensitivity non-destructive procedures for determining particle shape.

Summer 2010 **Intern**, *Kerr Vayne Systems*

Created a web application to stream real-time data for schedule display in a television broadcast automation system.

Releases and Publications

2025 **pgxx**, `https://github.com/george-steel/pgxx`

A high-level Postgres client for go which supports ACID/serializable transactions as well as automatic mapping of query results and parameters to and from user-defined structs.

2024 **Bowfishing Blitz**, `https://github.com/george-steel/bowfishing-blitz`

A tech demo/minigame which demonstrates my new graphical technique of clip-space planar refraction, used to render water with physically-accurate refraction without having performance costs of raytracing.

2021 – present **Contributions to Ink/Stitch**, `https://github.com/inkstitch/inkstitch`

A tool used to program numerically-controlled embroidery machines. I added a new algorithm for running stitch along curves which gets much more uniform stitch spacing as well as a system for randomized satin stitch which stays stable under changes in path shape.

2021 **Ojibwe Dictionary App**, `https://ojibwe-dict-test.netlify.app/`

A searchable interface to the Ojibwe-English dictionary compiled by Weshki-ayaad, Charlie Lippert, and Guy Gambill. Used Rust and WebAssembly to create an approximate search function that works effectively despite local spelling differences.

2019 **persistent-spatial**, `https://hackage.haskell.org/package/persistent-spatial`

A data structure for storing and indexing geographic coordinates which can be used with any SQL database to provide fast area queries using a standard b-tree index. Originally used with CockroachDB, which had no spatial index support at the time of release.

2017 **Maxent Phonotactic Learner**, `https://github.com/george-steel/maxent-learner`

A machine-learning tool for automatically inferring phonotactic grammars from a lexicon and using those grammars to generate random text, based on Hayes and Wilson's *A Maximum Entropy Model of Phonotactics and Phonotactic Learning*.

2017 **frpnow-gtk3**, `https://hackage.haskell.org/package/frpnow-gtk3`

High-level interface for GTK3 using FRPNow for asynchronous, reactive event handling.

2016 – 2017 **PhonoApps**, *with Prof. Peter Jurgec*, `http://phonology.us/`

Computational and learning tools for phonologists.