

Adam T. Geller

atgeller@cs.ubc.ca

(604) 440-2503

<https://cs.ubc.ca/~atgeller/>

EDUCATION:

University of British Columbia (Graduate Student)

University of Washington (Bachelor's)

2016- 2018

GPA: 3.63

Relevant Courses:

- CSE 311: Foundations of Computing I
- CSE 312: Foundations of Computing II
- CSE 331: Software Design and Implementation
- CSE 332: Data Structures and Parallelism
- CSE 333: Systems Programming
- CSE 341: Programming Languages
- CSE 351: The Hardware/Software Interface
- CSE 391: Software Tools
- CSE 401: Compilers
- CSE 451: Operating Systems
- CSE 452: Distributed Systems
- CSE 461: Introduction to Computer Communication Networks
- CSE 498: Senior Project / Research
- CSE 507: Computer-Aided Reasoning

Bellevue College (Associate's)

2014-2016

GPA: 3.98

Relevant Courses:

- CS 211: Fundamentals of CS II
- BA 240: Statistical Analysis
- ECON 199: Individual Studies
- ECON 201: Microeconomics
- ECON 202: Macroeconomics
- ENG 235: Technical Writing
- MAT 254: Calculus IV
- PHYS 121: General Engineering Physics I

PUBLICATIONS:

"Verifying that Webpages have Accessible Layout". P. Panckhka, A. Geller, M. Ernst, Z. Tatlock, S. Kamil. Accepted to PLDI 18.

As part of Adam's work on Cassius (see Projects below), he co-authored a paper that was accepted to appear at the PLDI 2018 conference. The paper describes a tool, VizAssert, that can verify assertions about the visual behavior of webpages. VizAssert is a DSL built on top of Cassius, allowing webpage developers to ensure their webpages satisfy a specification even with variable screen width, screen height, and font size. The webpage developer can write their own assertions, along with some example assertions we provide based on accessibility and usability best practices.

PROJECTS:

- Cassius Project: Adam is one of the key developers in a research group in the Programming Languages and Software Engineering group at the University of Washington working on The Cassius Project, <http://cassius.uwplse.org/>. He helped develop formal semantics for CSS floating elements based on the CSS informal specification. He wrote tests in Z3 to ensure that the specification is met for valid inputs. He added support for foreground and background color to the Cassius Framework, and extended the framework's treatment of text boxes by generating constraints based on font metrics. He also wrote a test-case minimizer to assist in the debugging of Cassius.
- Fencing Scorekeeper: Adam designed and built a Windows Phone application for use refereeing fencing tournaments. Fencing Scorekeeper keeps time and score, and understands the rules for different types of bouts that lead to different timekeeping and scoring behavior. It provides a simple and intuitive user interface for referees. Fencing Scorekeeper was developed in C#. It is multithreaded to support consistent and correct timekeeping while minimizing battery drain.
- Vectacular: Adam conceived, designed, and developed a linear algebra package to use for his high school math homework. The package included matrix and vector addition and multiplication, matrix-vector multiplication, vector dot and cross products, Gaussian elimination, matrix inversion using determinants, various geometric operations such as intersections between lines and planes, and a variety of other useful routines. Vectacular was originally developed in C++, and then ported to C#.

SKILLS:

- C#, Java, Racket; familiar with TypeScript/JavaScript, F#, CSS, HTML, SML, C++, C, x86 assembly
- Experienced with development tools such as Visual Studio and VS Code, Git, and Linux shell
- Familiar with the Z3 theorem prover
- Experience with Windows phone development
- Experienced with Microsoft Office

EXPERIENCE:

U.S. Fencing

2010-current

Referee

Adam is a national fencing referee in foil, epee, and saber. He has directed at local, national, and international tournaments, directing events at every level and age group. As a referee, he has learned valuable leadership skills such as communication, teamwork, and making tough decisions under pressure.