# Jack Wrenn


| | |
|---|---|
| Phone | 401-793-6774 |
| Email | jack@wrenn.fyi |
| Website | jack.wrenn.fyi |
| Github | jswrenn |


## Education

### Brown University

*Computer Science*

**Ph.D.** (Anticipated) 2021

**Sc.M.** 2018

### University of Rhode Island

*Computer Science, History*

**B.A.** 2015

## Employment

### Systems Operator & Technician

*University of Rhode Island* 2011–2015

Provided technical support and maintained key infrastructure (Windows and Linux servers) for the ~40 staff members of URI's student center.

### Course Staff

*Brown University, URI* 2012–2020

Provided design, instructional, and infrastructure support to various courses in OOP, functional programming, language design, and model-finding.

### Researcher

*Brown University* 2015–2021

Contributed to the development of the Pyret programming language, and published five research papers on error reporting, software testing, and human factors.

## Selected Awards

### Client Bug Bounty

*Mozilla* 2019

Discovered a security issue in Firefox's XSLT engine that enabled attackers to indefinitely execute JavaScript and send network requests (even with JS disabled) *after* the exploited tab was closed.

## Selected Projects

### Rust Programming Language

Passionate about expanding and leveraging Rust's language features to improve the safety and ergonomics of programming.

#### Safe Transmute Working Group

Co-lead the working group tasked with making bit-reinterpretation casts (e.g., `union`, `mem::transmute`) memory safe. Lead designer and author of the WG's inaugural RFC-2981.

#### Compiler Development

Implementor of RFC-2363, which permits fine-grained control over the memory layout of complex `enum` types; useful for C-interop and zero-copy parsing of network packets.

#### Open Source Libraries

Author and contributor to numerous Rust libraries and utilities. Lead maintainer of itertools, a popular (>16M downloads) library for ergonomic transformations and summarization of streams of data.

### Computing Education

Contributor to Pyret, a programming language designed by computer science educators, for computer science education.

#### Error Message Design

Developed Pyret's unique hypertext error messages, which leverage hyperlinks, highlights and in-line code snippets, that guide novice programmers towards deeply understanding their errors.

#### Data Science Support

Developed Pyret's language-level support for manipulating tabular data, used by the Bootstrap Data Science curriculum.

#### Example-Driven Development

Developed a cloud-hosted IDE for Pyret that encourages students to write input–output examples *before* they begin programming, and conducted research to assess its impact.

#### Evaluation at Scale

Developed infrastructure for assessing the quality of students' Pyret programs and test suites at massive scale on a distributed super-computing cluster, and developed an analysis to validate the robustness of these assessments.

### Digital Archival

#### Natural Language Processing

Published a digital remix of Brown University's authoritative dead-tree encyclopedia, which leveraged natural language processing to extract timelines and insert hyperlinks.

#### Digital Asset Management

Currently developing a digital asset management system in Rust and Typescript to catalogue my archive of >5,000 photographs, oral histories and written accounts of Brown University alumni.