

ABHIRAG AWASTHI

PERSONAL INFORMATION

website abhirag.com
email hey@abhirag.com
phone (+91) 86 19 18 8322

EDUCATION

2008-2012 **Dhirubhai Ambani Institute of Information and
Communication Technology**
*Bachelor of
Technology* Specialized in Information and Communication Technology
*Selected
Coursework* Data Structures, Introduction to Algorithms, Systems Software, Computer
Networks, Software Engineering, Probability & Statistics, Stochastic Simulation
GPA 7.5/10

LANGUAGES

*Intermediate
Proficiency* Rust, Go, C, Python, SQL, C#, [Zig](#), Lisp (*various dialects*)
Basic Proficiency [Elixir](#), [Dhall](#), JavaScript, HTML, CSS, \LaTeX

TOOLS

Web Backend Go Standard Library, Flask, Django
Web Infrastructure Docker, AWS, [Nomad](#), [Consul](#), [Vault](#), [Packer](#), Terraform, [Envoy](#)
Data Analytics [Spark](#), NumPy, Pandas, Matplotlib, [Parquet](#), Jupyter Notebook
Database PostgreSQL, Redis, SQLite, [Bolt](#)
Observability [OpenTelemetry](#), Datadog, Sentry, [Tracy](#)
Build System [Meson](#), Make
Version Control Git
Game Engine/Libs [GLFW](#), [SDL](#), [LÖVE](#)
Code Editor Emacs, GoLand, CLion, VS Code, Visual Studio
OS Linux (Ubuntu, Debian, NixOS), Windows, macOS

EXPERIENCE

Aug 2012 - May 2016 **Hewlett-Packard**
*Systems Software
Engineer* · Maintained core firmware services like Power Management, Firmware
Installation Management, Reset, Backup/Restore, Time
· Increased the fault tolerance of firmware upgrades and rollbacks by reducing
the probability of boot partition corruption caused by power failure
· Exposed system calls as safe to consume APIs and SNMP OIDs
· Acted as the first point of contact for crash dump analysis
· Resolved defects faced by customers with multi-million dollar contracts

Jun 2016 - Dec 2018 **Mockint**
*Founder /
Programmer* · Designed and developed Mockint, a platform to prepare for technical
interviews

Site Reliability
Engineer

Aug 2019 - Aug 2021 **Shuttl**

- Developed and maintained large parts of service infrastructure, atop which a fleet of microservices were built
- Built developer tooling to improve the user experience of interacting with the service infrastructure
- Assumed responsibility of developing and maintaining web services with tighter than usual SLOs
- Created data processing pipelines that periodically transformed GPS data into relevant metrics
- Wrote the software service infrastructure necessary to run cloud-native Rust web services
- Decreased the time spent performing data serialization/deserialization in Python web services by a factor of ≈ 3.8 . Achieved this by rewriting the data serialization/deserialization layer in Rust while preserving binary compatibility using property based testing

PROJECTS

Professional

-
- 2012 · **Jedi Firmware** (C#, .NET Compact Framework, C, WinDbg)
· Firmware for Hewlett-Packard LaserJet printers
 - 2016
 - 2016 · **Mockint** (Python, Django, JavaScript, Microsoft Bot Framework)
· Platform to prepare for technical interviews
 - 2018
 - 2019 · **Shuttl Infrastructure** (AWS, Docker, Terraform, Nomad, Consul, Vault)
· Shuttl's AWS cloud infrastructure, declaratively defined using Terraform. Hosting containerized services orchestrated through Nomad and connected using Consul. Secrets managed by Vault.
 - 2021
 - **Shuttl Backend** (Python, Flask, PostgreSQL, Redis)
· Fleet of microservices powering the Shuttl apps
 - **Virgo** (Go, PostgreSQL, SQL)
· Platform as a Service(PaaS) built for providing an abstraction layer over the Shuttl Infrastructure
 - **Pick-up Point Adherence** (Python, Spark, Amazon EMR, Parquet)
· Periodic big data processing pipeline that analyzed GPS coordinates collected from the fleet of Shuttl buses to detect any deviance from the designated bus stops

Personal

-
- 2016 · **Interactive Résumé** (GameMaker, GML)
· Interactive version of my résumé in the form of a platformer game
 - 2017 · **Résumé Plot** (Python, Matplotlib, Jupyter Notebook)
· My résumé as a plot of Tupper's self-referential formula
 - **Interface to Indian Railways** (Python, Pandas, Folium)
· Visualization of route and live running status of trains in India
 - 2018 · **Literate Ray Tracer** (Clojure, Emacs)
· Ray tracer written in literate programming style
 - **Tachyons CSS Playground** (JavaScript, React)
· Web App for interactive web design
 - **Haxorville** (Type Design)
· A quirky monospaced font designed for programmers
 - **Scribe** (C, Lisp)
· A tool for writing better documentation of programs. It borrows ideas from literate programming and augments them with static analysis

· **Transmogrify** (C, Lisp)

Converts markdown mixed with a smidge of Lisp to \LaTeX , HTML and plaintext

RESEARCH

2014 Using Web Page Segmentation to Mitigate Negative Effects of a Subpar Web Design on Print-Quality

*(Intellectual
Property of
Hewlett-Packard)*

An attempt towards improving the typography and signal to noise ratio of printed web pages

March 20, 2022