

Hemil Ruparel

rahem027.gitlab.io | (+91) 7020582873 | linkedin.com/in/hemil-ruparel-2aa513166 | hemilr026@gmail.com | gitlab.com/rahem027 | github.com/HemilTheRebel/

A software developer trying to be an “engineer”, learning the formal foundations behind the code we write. A firm believer of formal modeling, my objective is to write provable code which models the business problems to deliver value to business with maximal ROI.

Skills

Languages - Go, Python, Rust, Java, Smalltalk, C#, Scheme (Lisp), C++, C, Dart

Frameworks - Flask, Flutter, FastAPI, SQLAlchemy

Databases - PostgreSQL, MySQL, MariaDB

Infrastructure - Google Cloud Platform (GCP), Amazon Web Services (AWS)

Paradigms - OOP, FP, declarative

Education

Bachelor of Technology in Computer Engineering

Bharati Vidyapeeth College of Engineering, Pune

June 2021 - June 2024

9.3 CGPA

Diploma in Computer Engineering

Government Polytechnic, Pune.

June 2018 - June 2021

9.5 CGPA

Experience

SDE-1 at Blox.xyz

June 17 2024 - Present

- Implemented bridge between multiple internal platforms to enable them to communicate with each other.
- Revamped the authentication service to use JWTs

SDE Intern at Blox.xyz

Jan 2023 - June 15 2024

- **Invent and Simplify**
 - Scrapped property data for analysis and future price prediction for essentially free which would have otherwise cost **Rs. 2 Lakh (~ \$2400) per month per user**.
 - Forked [metabase](#) and implemented a paid feature for free. We needed the ability to filter the data of the reports based on the current user. Spent 2 days hacking around and found a solution but it was discarded as it violated the licensing terms. Spent another week and found a workaround that only touches open source code. Allowed the company to **save \$1400 (~ Rs 1.15 Lakh)**.
- **Backend / Backend for frontend (BFF)**
 - Used [COWJ](#) to rapidly develop a Backend for Frontend (BFF) layer to location service unblocking front end team and relieving work from backend team. **P90 request latency was < 200ms**.
 - Used [COWJ](#) to rapidly develop the full backend for our CP App allowing us to **release it within 2 months without cross team dependencies**.
 - As the first user of [COWJ](#) and [Zoomba](#) in Blox, **reported multiple issues [here](#), [here](#), [here](#) and fixed a couple of them - [COWJ](#) and [Zoomba](#)**
 - Developed BFF for [tv.blox.xyz](#) with a high level overview of the sales team for a given day to **enable the sales team to analyze their performance in real time**.
 - Our in-house data ingestion engine and free text search only supported ingesting one file at a time. Added a queuing mechanism paired up with a small UI that allowed business

people to upload whatever data they want whenever they want **removing the reliance on tech to ingest data.**

- **Mobile Team**

- Set up **build variants** to enable distinction between apps in QA, UAT and Prod
- Set up the **app on iOS**. Even though it was built using flutter, no one had tried to release the iOS version.
- As a part of the mobile team, set up error reporting tools like **crashlytics and sentry** and analytics tools like **Firebase Analytics and Amplitude** to help product focus on the right areas.
- Fixed more than **30 bugs and 11 on call issues within first 3 months** of my tenure
- Introduced **feature flags, remote configuration, A/B testing** into the mobile team
- Set up an **event based notification service** using MoEngage to allow the business team to configure their own notification setup that they can add and maintain.
- Reduced an animation from 4 seconds to 1 making the **app startup time 3x faster within my first week.**

Freelancer at Lexicard

March 2021 - Jan 2023

- Built a spaced repetition word learning app using Flutter and Firebase with Python FastAPI Backend and PostgreSQL database
- Implemented Anki import/export
- Integrated Google Translate API to perform the translation
- Integrated in app purchases for both ios and android

Side Projects

- [No Nonsense](#) - A simple Static Site Generator (SSG) in under 60 SLOC. Important realization was: **SSG = template engine + config**. All it took was combining [wheezy.template](#) (as it can embed arbitrary python), and [yaml](#) for config (as it has very little syntax) and some boilerplate. For reference (Jekyll is 10k SLOC, astro is 135k SLOC, Hugo has 183k SLOC)
- [STFU](#) - (Simple Testing Framework for Unit Tests) - Single header **C++ unit testing framework in 700 lines of code including comments**. Just drop the header file in your project and get going. No installation. No need for package managers. Get straight to development. All other testing frameworks required way more integration than necessary

Talks

- [\[CppIndia\] Fun with Lambdas](#) by Hemil Ruparel
- [\[CppIndia\] The Tragedy of Testing](#) by Hemil Ruparel
- [\[CppIndia\] Overview of Asynchronous Programming](#) by Hemil Ruparel
- [\[CppIndia : N-Lightning Talks\] TDD - The Bad Parts](#) by Hemil Ruparel

References

- Nabarun Mondal - [linkedin.com/in/nabarunmondal](https://www.linkedin.com/in/nabarunmondal)
- Pravin Mehta - [linkedin.com/in/pravinmehta/](https://www.linkedin.com/in/pravinmehta/)
- Manoj Sarwa - [linkedin.com/in/manoj-sarwa/](https://www.linkedin.com/in/manoj-sarwa/)