

# Punit Lodha

✉ [punitlodha@pm.me](mailto:punitlodha@pm.me) 🌐 [punitlodha.com](https://punitlodha.com) 🐙 [PunitLodha](https://github.com/PunitLodha) 🌐 [PunitLodha](https://www.linkedin.com/company/punitlodha)

## EDUCATION

---

### Saarland University

*Masters in Computer Science*

Saarbrücken, Germany

*Oct 2022 – Present*

- **Courses:** Data Networks, Compilers, Distributed Systems, Databases
- **Thesis:** Enhancing video streaming quality of experience using partially reliable transport and QUIC

## EXPERIENCE

---

### Software Engineering Intern

*Lumeo*

Jan 2023 – July 2024

*Remote, San Fransisco*

- Enabled 1,000+ new users to perform custom video analytics by designing and deploying a Python service and an Axum backend in Rust, leveraging Kubernetes to offer free cloud gateways.
- Increased system reliability by implementing the MQTT transactional outbox pattern to ensure consistent and fault-tolerant communication between the database and message broker.
- Implemented background jobs to delete orphaned deployments upon source deletion, to clean up expired events, and automate stream renaming for consistency.
- Designed and implemented APIs for dashboards, tagging entities, and other quality of life features.

### Software Engineering Intern

*Emotech.ai*

Dec 2021 – Sept 2022

*Remote, London*

- Improved media decoding performance by 10% by developing a Rust-based streaming media library using FFmpeg to efficiently extract audio samples and video frames, and integrating it into a multimodal pronunciation assessment engine.
- Boosted text normalization efficiency by 6% by optimizing resource loading in the text normalizer module.
- Implemented an API using axum to render animations into 3D models using the FBX format.

### Student Developer

*Google Summer Of Code 2021 w/ CCEXtractor* [View Project](#)

Mar 2021 – Aug 2021

*Remote*

- Reduced subtitle delay by 90% by fixing critical bugs and porting the CEA-708 decoder from C to Rust, with enhanced documentation for easier maintenance.
- Increased subtitle format support by adding writers for SRT, SAMI, transcripts, and other formats, improving the decoder's versatility.
- Streamlined cross-platform compatibility by integrating Rust code into various build systems, including CMake, Autotools, and Visual Studio, for both Linux and Windows platforms.

## VOLUNTEER

---

### Mentor

*Google Summer Of Code 2022, 2023 and 2024 with CCEXtractor*

May 2022 – Present

*Remote*

- Guided 5+ GSoC students across three years, leading to successful project completions by taking regular updates and mentoring on best coding practices and project management.
- Ported a large part of the codebase to rust, improving code stability and maintainability.

## PROJECTS

---

### Bitr | *Rust, tokio*

- Implemented the Bittorrent Protocol Specification to create a bittorrent client in rust.
- Used the rarest first piece picking algorithm, ensuring better piece distribution across the swarm and increasing tolerance to peers leaving the network.

## TECHNICAL SKILLS

---

**Languages:** Rust, Javascript, C, C++, Python, Go, Java, HTML/CSS, SQL

**Frameworks:** Postgres, Axum, React, Node.js, Material-UI

**Developer Tools:** Git, Docker, Kubernetes, GitHub actions, Figma, GCP, Azure, VS Code, Visual Studio, Neovim