

## EDUCATION

**University at Buffalo, State University of New York at Buffalo, NY**

*Master of Science, Computer Science Engineering*

February 2019

**Punjab Technical University, India**

*Bachelor of Technology in Computer Science Engineering*

June 2014

- 4-time winner of the best project in the semester.
- 3-time winner of the university-level coding competition.

## SKILLS

**Web Development:** REST API Development, Load balancing, Replication, Redis, Job queues, Cron Jobs, Docker

**Languages:** PHP, Python, Java, C, Bash Scripting, JavaScript, SQL, Django and Laravel Web Frameworks

**Others:** Microservices, Multithreading, Network programming, Design Patterns, Regular expressions

## EXPERIENCE

**Freelancing, Punjab, India**

*Software Engineer – LAMP Stack*

Jan 2016 – Mar 2017

- Implemented backend for a pollution monitoring system, deployed in 6 locations in India. Data partitioned based on location with a slave DB for each replica and interfaced through a REST API.
- Built back-end for a social network, deployed for 7 regions in Europe. Reduced server load by 33%. Microservices built for scaling the application, assisted by database replication.

**Infowiz, Chandigarh, India**

*Software Engineer – LAMP Stack*

Feb 2015 – Dec 2015

- Enhanced product search/listing functionality of an E-Commerce project and increased response time by 46%. Normalized table structure, join queries, indexes and in-memory caching used for optimization.
- Built document management module for an ERP, reducing paperwork and better sharing of documents. History of changes recorded for future rollbacks and efficient storage of documents.

**Newgen Software Technologies, Noida, India**

*Software Engineer*

July 2014 – Feb 2015

- Created front-end for a DIY enthusiast crafting machine, with the canvas created using Flex Framework.

## ACADEMIC PROJECTS

- **Distributed key-value store based on Amazon Dynamo with chain-replication and linearized transactions**  
Nodes organized in a ring structure and data partitioned using SHA hash of the keys. Chain replication implemented for complete linearization of transactions and load balancing of read-write operations. Nodes can join and leave dynamically, and the system is fault tolerant using a heartbeat style detector.
- **Distributed hash-table based on the Chord ring design with partitioning and dynamic join/leaving of nodes**  
Each node maintains a finger-table of the adjacent nodes which gives  $O(\log n)$  network lookup of nodes. Data is partitioned across the nodes and the replication between the nodes is causally consistent.
- **Message Multi-cast utility with a Total-FIFO ordering of messages in a cluster using ISIS algorithm**  
Messages are reliably multi-cast to a group of nodes and the ordering of messages is same for all the nodes. This is achieved by implementing the ISIS algorithm for total ordering.
- **Scheduler, System-call infrastructure, Virtual memory and File System subsystems for a x86 OS, Pintos**  
Multi-queue feedback scheduling algorithm implemented for the scheduler rotated in a round-robin manner. Read, write, exec and other Syscalls implemented. Paging and eviction policies implemented.