# **Darshan Sharma**

+91-7009746321

thedarshansharma@gmail.com

https://linkedin.cm/in/darshansharmain

### **Education**

Panjab University. B.E (CS)

2014-2018

## **Skills**

Web: HTML5, CSS3

Frameworks: React.Js, Express Node.Js, Nest.Js, Flutter

Tech Stack: MERN, LAMP, PERN

DevOps: CI/CD, AWS-Lambda, EC2, ECS, EKS, S3, Docker, Kubernetes

Web API: REST API, GraphQL

**Database:** PostgreSQL, MongoDB, Redis **Load Balancer:** Apache Kafka, Rabbit MQ

Programming Languages: Java, Python Dart, JavaScript, TypeScript

## Work Experience (5+ Years)

### **MonkAl Technologies**

**Position: Senior Software Engineer** 

May 2020 - Dec 2023

- Improved video player load time in the app feed by pre-loading just 25% of the content, fetching the rest only after 90% viewership of the initial segment, guided by user data analysis.
- Reduced DB expenses by 10% by integrating Redis as a caching layer, resulting in a single definitive database call rather than 6-7 heavy Mongo calls.
- Developed 7-9 Flutter mobile and web apps featuring in-app purchases, animations, push alerts, auth, and maps. Launched on Play/App Store, leveraging Firebase for auth, database, and GCP cloud functions..

#### **Paxcom**

#### **Position: Software Engineer**

May 2019 - Oct, 2019

- Decreased the load on the node server backend by 34% as it was being used for heavy CPU computations like image processing, and CSV to HTML conversion using a RabbitMQ as a load balancer.
- Achieved a 27% decrease in server expenses by transitioning from a monolithic to microservices, used RabbitMQ for load distribution, coupled with the integration of automated scaling and efficient traffic management.

#### Block8

#### **Position: Software Engineer**

May 2019 - May, 2019

 Boosted a Node app's speed by tweaking database queries, markedly enhancing its operational efficiency.

# **Projects**

- MyStake: Reduced loading time of web page by 20% by optimizing database queries, specifically by selecting before joining operations.
- Paxcom: Reduced the memory usage by 70% of a CPU-intensive task of Excel to HTML conversion in Node.js by architecting a dedicated microservice