

## EDUCATION

- |  |  |
|--|--|
| • Master of Science in Computer Science, GPA 3.64/4.0<br><i>Saint Louis University</i> | <i>Jan 2023-Dec 2024</i><br><i>Saint Louis, MO</i> |
|--|--|

## EXPERIENCE

<b>Software Developer</b> <i>Metis Analytics</i>	<b>Jan 2025 - present</b> <i>Saint Louis, MO</i>
---	---

- Developed modular components, and reusable components, implementing custom hooks and HOCs, using JavaScript and React.js resulting in 40% faster development cycles for new features.
- Created CI/CD pipeline using GitHub Actions and AWS Code Pipeline with automated Docker image builds and deployments.

<b>Software Developer</b> <i>Gamyam</i>	<b>May 2024 – Dec 2024</b> <i>Saint Louis, MO</i>
--	--

- Developed high- availability notification microservices using JavaScript, Nodejs and express handling 50k + daily notifications with 99.9% uptime.
- Implemented virtualized list component with infinite scroll using intersection observer API and React window context handling 10k+ records while maintaining <2s load time.
- Developed message queue system using Docker, RabbitMQ and AWS ECS handling 50k+ messages/minute
- Implemented service discovery and load balancing using AWS ECS Service Discovery and Application Load Balancer.

<b>Assistant Systems Engineer</b> <i>Tata Consultancy Services</i>	<b>Nov 2021 -Dec 2022</b> <i>Hyderabad</i>
---	---

- Developed and maintained frontend monitoring dashboard using AppDynamics, creating custom metrics for JavaScript heap usage, component render times, and API latencies, maintaining 99% frontend availability.
- Engineered high-throughput message processing system using Apache Kafka and RabbitMQ, implementing dead letter queues and retry mechanisms.
- Implemented caching layer using Redis and Elasticsearch, invalidation strategies and data replication that reduced database load by 70% and improved query response times.
- Built scalable data pipeline using AWS Lambda, S3, and DynamoDB, implementing event-driven architecture that processes 5TB+ daily data with automated failover and recovery mechanisms achieving 99.9% data consistency

<b>Full Stack Developer Intern</b> <i>Naresh Technologies</i>	<b>Jul 2021 -Nov 2021</b> <i>Hyderabad</i>
--	---

- Developed full stack application using React.js and Node.js implementing responsive design patterns and cross-browser compatibility.
- Participated in code reviews and collaborated with senior developers to maintain coding standards and best practices.
- Optimized Database performance through connection pooling and prepared statements to prevent SQL injection.

## SKILLS

- **Frontend** : JavaScript/TypeScript, React.js, Next.js, Svelte, CSS, Tailwind CSS, Bootstrap, Web Standards
- **Backend** : JavaScript/TypeScript, Node.js, Express.js, Python,
- **Databases & Caching**: PostgreSQL, MySQL, MongoDB, PostgreSQL, Redis, Pocket base
- **Cloud & DevOps**: AWS (ECS, Lambda, S3), Docker, Kubernetes, CI/CD
- **Testing**: Jest, Cypress, Junit, Istanbul, React Testing Library, Swagger, Postman Testing
- **API Integration**: REST, Swagger, Postman, SOAP, XML, JSON

## PROJECTS

<b>Flagrant Fowl Football- club Web Application</b> <i>Principals of Software Development Course Project</i>	<b>16 weeks</b> <i>Saint Louis, MO</i>
---	---

- Designed a scalable web application using Next.js, TypeScript and PostgreSQL for managing football club operations.
- Implemented API authentication with using Next AUTH and developed Fan engagements features during live matches using Next.js

<b>Database Developer -Team lead</b> <i>Community Mediation Services of St. Louis</i>	<b>12 weeks</b> <i>Saint Louis, MO</i>
--	---

- Led of migrating, Legacy database to MySQL and conveying technical requirements and architecture to the business users.
- Automated PL/SQL reporting implemented monitoring solutions using Grafana dashboards for real performance metrics.

<b>Help – Web Application</b> <i>Computer Networks Project</i>	<b>12 Weeks</b> <i>Saint Louis, MO</i>
---	---

- Developed responsive frontend using React.js with custom hooks for geolocation services and real-time updates, implementing lazy loading for location-based components, built real-time chat interface using Socket.IO and React Context, implementing UI components for message threads, and user presence with optimized re-rendering patterns