

# Prajakta Pikale

(217) 200-5065 | Champaign, IL | [ppikale2@illinois.edu](mailto:ppikale2@illinois.edu) | [LinkedIn](#) | [GitHub](#) | [Portfolio](#)

## EDUCATION

**University of Illinois, Urbana-Champaign** | Illinois, United States Aug 2024 - Dec 2025  
Master of Computer Science | Graduate Teaching and Course Assistant GPA: 3.8/ 4  
Coursework: Cloud Computing Applications, Distributed Systems, Database Systems, ML for Software Engineering

**Vishwakarma Institute of Information Technology** | Pune, India Aug 2017 - May 2021  
Bachelor of Technology in Computer Engineering GPA: 9.49/ 10

## TECHNICAL SKILLS

**Programming:** C++, JavaScript, TypeScript, Python, SQL  
**Web Development:** React, Next.js, Three.js, HTML, CSS/SASS, Tailwind, RESTful APIs, Node.js, Express.js, Django, Flask  
**Databases:** MySQL, Microsoft SQL, PostgreSQL, MongoDB, DynamoDB, Neo4j, Firebase  
**Cloud and Platforms:** AWS, GCP, GitHub Actions, Docker, Kubernetes, Linux, Git, Jira, Postman, Figma, VS Code, WebStorm

## EXPERIENCE

**Expedia Group** | Software Development Engineer May 2023 – July 2024

- Developed a full-stack **data analytics web platform**, with an interactive **React** frontend featuring **custom-built data visualization charts and dashboards**, including **Tableau** integration, supported by a **Node.js** backend for API management.
- Built an interactive dashboard to visualize core business **KPIs** (e.g. monthly and weekly recurring data) by integrating data into **Material-UI React Table** and using **Highcharts**, while **saving 10 hours** of manual work per week and enabling **data-driven business decisions**, collaborated with cross-functional teams.
- Designed and maintained **new features** and **APIs**, optimized APIs ensuring response times to stay under **100ms** for real-time insights, and effectively addressed **software bugs** to optimize user experience.
- Automated **GitHub Actions** for CI/CD pipelines, collaborated on version control via GitHub, and streamlined workflows for **code deployment** using **Jenkins**, **Docker** and **Kubernetes**.

**Cognizant Technology Solutions** | Programmer Analyst Sep 2021 – Mar 2023

- Contributed to building the **frontend** of a web application for one of the largest Swedish **insurance companies**, streamlining internal team management processes using **React**.
- Implemented **Redux** and **React Hooks** for state management and handled **API calls**, optimizing an API endpoint to achieve a **15% faster** round-trip time.

## PROJECTS

**3D Web Developer Portfolio** | React, Three.js, React Three Fiber, Framer Motion, Tailwind Jan 2025

- Developed an interactive **3D portfolio** using **React** and **Three.js**, integrating **GLTF** models for **3D rendering**.
- Used **React Three Fiber** to create **dynamic animations** with **Framer Motion** for fluid visual and responsive web experience.

**Data-Driven Forecasting and Visualization of Fresh Food Demand** | React, Chart.js, Python, MySQL, Django, ML Dec 2024

- Developed a web application using linear regression **machine learning** model **to predict** fresh food demand, resulting in a significant reduction in inventory waste by optimizing stock levels and **improving supply chain efficiency**.
- Designed interactive charts and dashboards with **real-time data visualizations**, enabling analysts to track inventory, **analyze demand** patterns, and make **data-driven decisions** for better stock management.
- Implemented secure, role-based access control and **automated alerts** for perishable inventory management.

**Quora Clone** | React, Redux, Node.js, Express.js, MongoDB Apr 2024

- Built a full-stack Quora Clone, a **social question-and-answer application** replicating key features of **Quora**.
- Integrated **authentication** using **Nodemailer** and **Passport.js** JWT strategy with **OAuth2 Google Sign-In**.
- Utilized **Context API** and **Redux** for state management, **react-router** for routing, and **React Hooks** for reusable logic.
- Developed **API logic** with **Node.js**, employing **Mongoose** to manage schemas, and perform **CRUD** operations in **MongoDB**.