YASH DAXINI

LinkedIn | 91730 66249 | yashdaxini.com | yashdaxini2003@gmail.com | GitHub

Summary

- Software Development Engineer with experience designing and implementing multi-tier distributed applications using Java, C#, NET, Node.js, Angular and React.
- Hands-on in building scalable RESTful microservices, optimizing backend performance, and integrating with cloud and on-prem systems across the full software development life cycle.
- Experienced in agile methodologies, CI/CD, automated builds, and code reviews, with a strong focus on reliability, performance, and Extensible system design.

Skills

- **Languages:** C# | Java | PHP | JavaScript (ES6+) | TypeScript | jQuery
- **Databases:** MSSQL | MongoDB | MySQL | Redis | Firebase
- **Frameworks & Tools:** React.js | Angular | Node.js | Express.js | ASP.NET MVC | .NET WebAPI | Slim (PHP) | Ionic Framework
- **DevOps & Build Systems:** Docker | TeamCity | IIS Deployment | CI/CD Pipelines | Mobile Build Automation (Android APKs) | Linux
- **Core Engineering:** REST APIs | Microservices | System Design | Scalable Systems | Data Structures | Algorithms | OOP
- **Version Control:** Git | GitHub
- **Other:** HTML5 | CSS3 | Bootstrap

Experience

Jr. Software Engineer — **Tark technologies** — **07/2024 - Current**

- Designed and implemented **CI/CD pipelines** in **TeamCity** for multi-tier applications, including **automated build steps**, **artifact generation**, and **IIS deployment** workflows, **reducing manual deployment time** and improving release consistency across QA and Staging environments.
- Generated **Android APKs** for Angular (Ionic) applications and **automated mobile build** and deployment workflows, accelerating testing cycles and reducing turnaround time for QA.
- **Improved performance** and reliability of backend services by refactoring inefficient database logic and API endpoints, reducing redundant queries and improving response times under increased load.
- Implemented and maintained hybrid deployment architecture **connecting on-premises** and **cloud-based systems**, **reducing data duplication** and enabling reliable, scalable bi-directional synchronization.
- Architected and delivered a **new microservice** that communicates directly with cloud applications, **reducing overall synchronization load**, **eliminating multi-database duplication**, and introducing structured logging and end-to-end authentication for secure, observable operations.
- **Developed plugins for HomeSeer** to integrate IoT devices (smart plugs, locks, Wi-Fi bulbs) using React, JavaScript, and C#, enhancing interoperability and robustness of automation workflows for customers.
- Contributed to an algorithmic trading platform as a full stack engineer, leading frontend implementation in React and optimizing backend performance using Redis caching, resulting in faster data refreshing and lower server load during peak trading periods.

Software trainee — **Tark technologies** — **01/2024 - 07/2024**

- Built and maintained backend modules using C#, .NET Core, and EF Core, following **clean code principles**, **unit testing**, and **code review practices** as part of the full software development life cycle.
- Implemented a customizable event scheduling system with recurring event logic and optimized algorithms, improving scheduling performance and reliability for high-volume usage scenarios.
- Collaborated with cross-functional teams to design and implement an AWS-based billing engine, applying object-oriented design, data modeling, and scalability best practices to meet evolving business requirements.

Education

Bachelor of Engineering — **Darshan University** — **08/2020 - 06/2024**

- **CGPA**: 9.31

Mentorship

- **Computer Science Tutor:** Programming | Data Structure and Algorithms | Database management system | React JS | Node JS

Projects

- **Customizable Event Calendar:** Built a **Microsoft Outlook-inspired calendar** application with **recurring event** logic and optimized overlap calculation algorithms, improving scheduling performance and handling edge cases reliably for large numbers of events. Link to Frontend Code and Backend Code.
- **Distributed Rate Limiter:** Developed a high-performance **distributed rate limiter** to manage API traffic **across multiple servers** using Node.js, TypeScript, React, and Redis, apply Lua scripts for atomic operations and robust edge-case handling. Implemented **sliding window** and **lazy refill** techniques to improve accuracy and performance under high concurrency, Demonstrating experience with distributed system design and reliability. Link to Code.
- **Comment System:** Created a dynamic comment system where users can create posts, comments, and replies, including Upvote/downvote functionality, using React, PHP-Slim, and MySQL, emphasizing scalable backend design and responsive UI. Link to Code.

Others

- **1st rank**: Achieved **1st place** in the 30-Hour **.NET Hackathon organized by Gateway Group**. **(03/2023)**
- **Top 0.4%**: Ranked **38th** out of 10,205 participants in the Code chef December Long Challenge 2022 and a **global rank of 260 in Starters 99**.