



Samhitha Patangi

Sr. Software Engineer – .NET

+1 (972)-379-8987 | saamhithaa831@gmail.com | [LinkedIn](#) | [GitHub](#)

PROFESSIONAL SUMMARY:

- 8+ years of experience as a **Sr. Software Engineer** delivering secure, scalable, and high-performance enterprise applications in healthcare, banking, financial services, and agriculture.
- Proficient in **C# .NET 6/7/8, REST API** development, Angular 16+ with RxJS, and SQL databases (MS SQL Server, Oracle, PostgreSQL). Strong Agile/Scrum contributor with expertise in CI/CD pipelines and service-oriented application development.
- Skilled in frontend development using **React, Angular (v7–16), Blazor, Razor views**, and modern JavaScript/TypeScript with responsive UI design.
- Expertise in **RESTful APIs, microservices, and SignalR** for real-time, event-driven communication across distributed systems.
- Strong database skills with **MS SQL Server (2012–2019), Azure SQL**, stored procedures, query optimization, indexing, and ETL workflows.
- Cloud experience in **Azure (App Services, AKS, Service Bus, Functions, Storage, API Management, Key Vault)** and **AWS (ECS, Lambda, S3)** deployments.
- Hands-on expertise in **Docker, Kubernetes, and Infrastructure as Code** using **Terraform, Bicep, and YAML** pipelines.
- Proficient in **CI/CD automation** using **Azure DevOps, GitHub Actions**, and **PowerShell** scripting for zero-downtime deployments.
- Skilled in **reporting & analytics** with **SSRS, Power BI, Crystal Reports**, and **SSIS** for data integration.
- Strong Agile/Scrum contributor with experience in sprint planning, backlog grooming, code reviews, and cross-functional collaboration.
- Utilized **Microsoft IIS** for hosting and managing enterprise-grade web applications with high uptime and performance.
- Proficient in development tools like **Microsoft Visual Studio, Azure DevOps**, and **Postman**, accelerating productivity and improving debugging.
- Skilled in **legacy migrations** from VB6/Java/C++/PLSQL to Angular & C# Web API services.
- Experienced in **microservices architecture, Splunk monitoring, IBM MQ messaging, WPF, WinForms, and Test Driven Development (TDD)** using MSTest, NUnit, Jasmine, and Karma.
- Collaborated in **Agile/Scrum environments**, working closely with cross-functional teams to deliver business-critical features on time.
- Demonstrated leadership by **mentoring junior developers**, conducting code reviews, and contributing to technical decision-making processes.
- Strong analytical skills with the ability to interpret complex business requirements and translate them into technical solutions.
- Committed to continuous learning, staying updated with the latest development trends and cloud technologies to deliver innovative solutions.
- Adept at breaking down technical topics for **non-technical stakeholders**, ensuring alignment between development and business objectives.
- Recognized for **problem-solving and debugging skills**, consistently resolving critical production issues with minimal downtime.

TECHNICAL SKILLS:

Languages & Frameworks:	C#, C++, VB6, VB.NET, ASP.NET Core, ASP.NET MVC, .NET Framework (4.5 – 4.8), .NET Core (2.2 – 8.0), Entity Framework (5, Core 2.1+), WCF, LINQ, Autofac, MediatR
Frontend Technologies	Angular (v6-16), React, Razor Views, Blazor, HTML5, CSS3, TypeScript, AJAX, SignalR, Bootstrap, jQuery, Javascripts
Web Technologies:	ASP.NET Core, ASP.NET MVC, Web API, REST, SOAP, N-Hibernate, Razor Pages, JSON, XML
Database Technologies:	MS SQL Server (2012–2019), Azure SQL, PostgreSQL, MongoDB, DynamoDB, Stored Procedures, Functions, Triggers, Query Optimization, ETL (SSIS), Oracle 12c/19c, PL/SQL, Stored Procedures, Functions, Triggers, Query Optimization, EXPLAIN PLAN
Cloud Services:	Azure (Functions, Blob Storage, API Management, AD, Service Bus, Event Hub, Application Insights), AWS (EC2, S3, Lambda, API Gateway, DynamoDB, ECS), Docker, Kubernetes, Apache Kafka
DevOps & CI/CD:	Azure DevOps, GitHub Actions, YAML Pipelines, Jenkins, Docker, Kubernetes, Terraform, Bicep, PowerShell Automation



Testing Tools	MSTest, NUnit, XUnit, Selenium, Postman, JMeter, Karma, Jasmine, Protractor, Swagger, Test-Driven Development (TDD)
Reporting & Analytics	SQL Server Reporting Services (SSRS), Power BI, Crystal Reports, SSIS
Development Tools:	Microsoft Visual Studio (2013, 2019), Microsoft IIS.
Other Technologies:	Visual Studio (2013–2022), Visual Studio Code, Microsoft IIS, SonarQube, Log4Net, NLog, UML, MS Visio, JIRA, WPF, WinForms, IBM MQ, Splunk

PROFESSIONAL EXPERIENCE:

Client: Palo Alto Networks, Texas, USA

Role: Sr. Full stack Developer

Jan 2024 to till date

Description:

Developed and secured a health insurance claims processing platform for women's health programs. WHCRT (ASP.NET Web Forms) managed claim submission, provider administration, and MOA workflows, while PHClaims (ASP.NET MVC) handled claims reporting, data analytics, and API integrations. In collaboration with Palo Alto Networks, integrated Medical IoT Security and Zero Trust policies to protect connected Internet of Medical Things (IoMT) devices—such as diagnostic machines, vaccine refrigerators, and clinical monitoring systems—against unauthorized access and cyber threats.

- Built REST APIs in .NET 6/7 with Entity Framework Core for secure claims processing.
- Designed service-oriented modules integrated with Angular 16+ components using RxJS for real-time updates.
- Implemented CI/CD pipelines via Azure DevOps ensuring automated deployments and quality gates.

Frontend

- Designed and developed **ASP.NET Web Forms** pages in WHCRT for provider management, claims submission, CPT code administration, and MOA agreement workflows, integrating **DevExpress** controls for rich UI functionality.
- Built responsive UI components in **ASP.NET MVC Razor views** for PHClaims, enhancing claims reporting dashboards and search/filter capabilities.
- Implemented **JavaScript/jQuery**-based validations, AJAX callbacks, and dynamic data binding to improve form usability and reduce page reloads.
- Customized **DevExpress ASPxGridView** and implemented **custom callbacks** for server-side filtering, sorting, and paging.
- Developed reusable **User Controls** for common UI features like session timeout warnings and role-based menu rendering.
- Embedded **AI Assistant Chatbot UI** using **JavaScript + SignalR/WebSockets** to allow real-time, HIPAA-compliant assistance for claim status inquiries, eligibility questions, and provider lookup.
- Designed **chat window components** with role-based visibility, ensuring different responses for claimants, providers, and administrators.

Backend

- Developed and maintained **C# code-behind** logic for WHCRT, handling CRUD operations, form processing, and integration with **SQL Server stored procedures**.
- Integrated **MongoDB** for high-volume claim event logging and unstructured provider communication records, enabling faster retrieval for audits and analytics.
- Implemented **MVC Controllers** in PHClaims to process HTTP requests, perform business logic, and return data to Razor views or APIs.
- Integrated **LINQ to SQL** and **Entity Framework** models for database operations, optimizing queries for claims and provider records.
- Added **session-aware conversation history storage** for better user experience and follow-up continuity.
- Developed **API endpoints** in PHClaims for secure data exchange with third-party healthcare systems.
- Utilized **NLog** for structured logging, error tracking, and performance monitoring across environments (DEVL, QUAL, PROD).
- Created and maintained environment-specific configurations (Web.DEVL.config, Web.PROD.config, Web.QUAL.config) for database connections, API endpoints, and logging.

Cloud / DevOps

- Used **Azure DevOps** for **version control, branch management, and CI/CD pipelines**, implementing gated pull requests and automated deployments to DEVL and QUAL environments.
- Managed multi-environmental deployments with environment-specific **Web.config transformations**.
- Participated in **code reviews**, ensuring compliance with **AppSec security scan** requirements before release.
- Monitored application logs and performance metrics in Azure for proactive issue detection and resolution.
- Coordinated with QA teams for functional, integration, and UAT testing prior to production release.
- Deployed applications on **Azure App Service** using Terraform, applying Infrastructure as Code (IaC) for consistent and scalable deployments.
- Participated in release management processes, including **Normal, Hotfix, Same-Day Hotfix, and Emergency releases**.



- Documented workflows, API endpoints, and deployment procedures for future maintenance and onboarding.

Environment: .NET Framework 4.8, Razor, Entity Framework Core, CI/CD pipelines, Azure DevOps, Docker, Agile, security best practices, SQL optimization, feature toggles, and documentation.

Client: Franklin Templeton, Hyderabad, India

Sep 2021 to July 2023

Role: Software Developer

Description:

Developed a cloud-based investment portfolio management platform for Franklin Templeton, enabling real-time visualization and analysis of mutual funds and private market assets. The solution integrated data from multiple acquisitions, providing advisors with a unified view of traditional and alternative investments. The system leveraged React for an interactive, responsive frontend, .NET Core and SQL Server for high-performance backend services, and Azure Kubernetes Service (AKS) with CI/CD pipelines for scalable, reliable cloud deployments. Real-time updates were delivered via SignalR, while Azure Service Bus facilitated secure, asynchronous communication between distributed microservices. Proactive monitoring, incident management, and optimized autoscaling ensured availability, optimized performance with loading, and system stability during peak at trading and critical reporting in periods.

Frontend

- Migrated core services to .NET 6 APIs with optimized SQL Server and Oracle queries.
- Integrated Angular 16+ UI with RxJS for reactive state handling and streaming portfolio updates.
- Automated deployments using CI/CD pipelines, reducing release time and ensuring high availability.
 - Developed dynamic Portfolio Fund (PF) tables and mutual fund dashboards using **React** and **SignalR** for real-time visualization of asset allocations during the transition from mutual funds to private markets.
 - Implemented lazy loading, modular component architecture, and reusable hooks to streamline performance and maintainability. Enhanced API security and modularity using **IOC containers** and **SOLID principles**.
 - Integrated data from newly acquired platforms (private credit, equity, and real estate) into unified portfolio views, enhancing advisor insights and decision-making.
 - Added cross-browser compatibility, state management with **Redux**, and improved error handling for a smoother user experience.
 - Enhanced advisor engagement by surfacing data from newly acquired platforms (e.g., private credit, equity, and real estate), improving portfolio transparency and decision-making speed.
 - Ensured cross-browser compatibility and added intuitive state handling and error feedback, improving UI stability and user experience.
 - Accelerated development speed through structured debugging and IDE-based tooling with Visual Studio 2019.

Backend

- Migrated large-scale **VB6/C++/PLSQL modules** to Angular & C# **Web API 2.0 REST services**, improving maintainability and performance.
- Optimized complex queries and stored procedures in SQL Server 2017 for faster execution of portfolio-level aggregations.
- Integrated Apache Kafka for real-time market data streaming and asynchronous processing of large trade event datasets, improving message throughput and reducing system latency.
- Used **Kafka** topics for distributing portfolio updates to microservices, ensuring consistent state across analytics and reporting modules.
- Integrated SignalR server-side logic to broadcast real-time fund updates, reducing polling needs and improving refresh latency.
- Implemented Autofac for dependency injection, decoupling core services and increasing unit test coverage; supported this with robust unit testing using MS Test.
- Developed and optimized **Oracle PL/SQL stored procedures** for claims processing, using **EXPLAIN PLAN** for query optimization.
- Contributed to the design of multiple key backend modules enabling integration of data from strategic acquisitions like Lexington, Clarion, and Benefit Street Partners.

Cloud / DevOps

- Deployed scalable microservices on Azure Kubernetes Service (AKS) to manage both legacy mutual fund and new private market data workloads, ensuring high availability during traffic spikes.
- Automated CI/CD pipelines using Azure DevOps, enabling faster release cycles with zero-downtime blue-green deployments.
- Leveraged **Azure Service Bus** for decoupled, reliable message passing across distributed systems, increasing communication resilience.
- Integrated **Splunk dashboards** for real-time monitoring, error tracking, and performance metrics across environments.
- Tuned AKS autoscaling and optimized resource utilization to reduce cloud infrastructure costs.
- Automated deployment and rollback workflows using YAML and PowerShell, saving significant manual deployment effort.
- Proactively monitored production environments and resolved incidents, improving mean time to resolution (MTTR).

Environment: C#, .NET Core 6.0, .NET Framework 4.8, MS SQL Server 2017, React, Redux, SignalR, Autofac, Azure AKS, Azure DevOps, Azure Service Bus, MSTest, PowerShell, YAML, Git.

Client: FNB Corporation

Jan 2020 to Aug 2021



Role: .Net Backend Developer

Description:

Built a secure, end-to-end platform to aggregate bank customer/transaction data, reconcile gaps, and produce quarterly regulatory submissions. The system generates “lost data” and “legal data” reports (e.g., legal holds, PII access logs, breach investigations), supports reviewer sign-offs and immutable audit trails, and exports regulator-ready files. Front end is Blazor Server for rich, real-time review workflows; back end is .NET Core APIs and workers; data stored in SQL Server with record-level lineage and encryption.

Frontend

- Built **Blazor** Server UI for quarterly submissions, with real-time review panels for “lost data” and “legal data” workflows using **SignalR**.
- Implemented reusable Razor components for filters, data grids, exception drill-downs, and attestation modals.
- Added role-aware UI states, optimistic updates, and accessible forms with validation summaries.
- Used `_Host.cshtml` and `_Layout.cshtml` to integrate theming and secure headers; configured CSP and anti-forgery.

Backend

- Designed CQRS handlers and REST endpoints for ingestion status, reconciliation results, approvals, and package generation.
- Wrote EF Core models and migrations; tuned stored procedures for quarterly aggregates and regulator extracts.
- Implemented background workers for scheduled ETL, exception rollups, and packaging (XML/CSV) with checksum manifests.
- Added comprehensive auditing (who/what/when), data lineage tracking, and immutable submission snapshots.
- Modeled PII domains with column-level encryption, data classification, and masking in non-prod.
- Enforced legal-hold tagging and export suppression rules; created audit reports for access and edits.
- Produced regulator-ready exports with schema validation and signed manifests; preserved artifacts in WORM storage.

Cloud / DevOps

- Deployed Web, API, and Worker to Azure (App Service/AKS); wired Azure Service Bus for decoupled processing.
- Set up YAML pipelines (build, test, security scans, infra deploy, app deploy); parameterized per environment.
- Configured observability with App Insights, custom traces for submission IDs, and alerting on SLAs & failures.

Environment: .NET Core 8.0, ASP.NET Core Web API, Blazor Server, Razor (.razor) Components, C#, Entity Framework Core, SQL Server 2019, Stored Procedures, Azure Service Bus, Azure App Service, Azure Kubernetes Service (AKS), Azure DevOps (YAML CI/CD), MediatR, AutoMapper, SignalR, Hangfire, Azure Key Vault, App Insights, Terraform/Bicep, Visual Studio 2022, Visual Studio Code, Git, REST APIs, JSON, Bootstrap, Fluent UI.

Client: Tangentia, Hyderabad, India

June 2018 to Jan 2019

Role: .NET Software Engineer

Frontend

- Developed and managed a SharePoint site with admin-level permissions, improving access control and usability.
- Built user management modules in Angular integrated with .NET backend, implementing secure login, registration, authentication, and authorization workflows.
- Utilized Angular(v7) Material Design with reactive forms to streamline validation, enhance UI interactions, and improve overall user experience.
- Designed and developed a full-featured Angular web application integrated with .NET APIs for seamless data flow and performance optimization.

Backend

- Designed and optimized RESTful APIs for authentication, authorization, and core business logic in .NET applications.
- Improved API performance and reliability through optimized data processing and secure coding practices.
- Developed Power Automate flows and basic Power Apps solutions to automate workflows and reduce manual intervention.

Cloud / DevOps

- Deployed and scaled applications on **Microsoft Azure**, improving uptime and performance for production and QA environments.
- Automated build and deployment processes using **Azure DevOps** and **PowerShell scripts**, reducing release cycles.
- Hosted web services on **IIS**, configuring load balancing and monitoring to ensure consistent service delivery across environments.
- Used **Visual Studio (2013/2015)** and **Azure DevOps** for collaborative development, integrated version control, and agile sprint planning.

Environment: .NET Framework, C#, Entity Framework, MS SQL Server, Windows Communication Foundation (WCF), Microsoft IIS, Microsoft Visual Studio, Azure, NUnit, Autofac, Azure DevOps.

EDUCATION

- Master's Science (M.S) in Computer Science from Northern Arizona University, Flagstaff, Arizona
- Bachelor of Technology (B.Tech) in Computer science from JNTUH, Hyderabad, Telangana, India.