

IZMA ANWAR

Senior Full Stack Developer

+92 331 7475 663 • izmaanwar005@gmail.com • <https://www.linkedin.com/in/izma-anwar-2a0125164/>

Education

Punjab University College of Information Technology,Lahore	Lahore, Pakistan
Bachelor's Degree in Computer Science CGPA: 3.45 / 4.0	2015 - 2019
Punjab Group Of Colleges	Sahiwal, Pakistan
F.Sc (Pre-Engineering)	2013 - 2015
• Bronze Medalist – Awarded for academic excellence among top board performers	

Programming Skills

Technologies And Architecture: C#, .NET Core, ASP.NET Core, MVC/Web API, ASP.NET, Entity Framework Core, LINQ, REST APIs, GraphQL ,Microservices, Event-Driven Architecture, DDD, CQRS, SOLID, Background Services, Cron Jobs , Mediator Pattern
Frontend/UI And Frameworks: Angular, HTML, CSS, Razor, Bootstrap, TypeScript, JavaScript, Blazor
Messaging and Databases: RabbitMQ, SQL Server, MySQL, Redis , Postgre SQL
Cloud And DevOps: AWS (EC2, Lambda, S3, RDS, CloudWatch , Redshift , EKS), Azure (DevOps, Pipelines, and App Service deployments), Docker, GitHub Actions, OpenShift, Nginx, API Gateway, IIS , Jenkins , Kubernetes
Testing: Unit Testing(xUnit) Code Coverage, Integration Testing, E2E API Testing
Code Quality & Analysis: SonarQube
Logging: NLog, Serilog, Log4Net, CloudWatch, Application Insights, OpenTelemetry, File-based Logging
Tools: Visual Studio, Visual Studio Code, SQL Server Management Studio, Git, Jira , Bitbucket , Postman/Swagger
Agile & Collaboration: Scrum, Kanban, Jira, Confluence, Code Reviews, Mentorship, Sprint Planning

Experience

Folio3	Lahore, Pakistan
Senior Software Engineer	2023 - Present
Leading the development of scalable livestock and dairy management systems, using .NET Core , Blazor, and modern cloud architectures, improving operational efficiency, automation, and analytics across multiple projects.	
Rolustech	Lahore, Pakistan
Senior Software Engineer	2019 - 2023
Led backend and frontend development using .NET Core, Angular, and MySQL, architecting microservices, implementing CI/CD pipelines, and enhancing system scalability, performance, and reliability.	
AIMRL	PUCIT, Lahore , Pakistan
Internee	2018 - 2019
Developed a student-faculty portal to streamline final year project allocation, enabling smooth coordination between students and supervisors using C#, ADO.NET, and SQL stored procedures.	

Projects

Cargill	USA
	08/2025 - Present
<ul style="list-style-type: none">• Architecture Design: Implemented CQRS pattern separating read and write pipelines, improving scalability and reducing API response time by approximately 20%.• Clean Architecture: Applied the Repository Pattern to maintain modular, clean, and reusable data access logic.• Monitoring & Tracing: Configured OpenTelemetry with Azure Monitor for distributed tracing and structured logging, reducing debugging time by approximately 30%.• Cloud Deployment: Containerized services using Docker and deployed to AWS EKS (Kubernetes).• Kubernetes Readiness: Implemented liveness/readiness probes ensuring stable deployments and faster recovery.• CI/CD Automation: Created Azure DevOps pipelines for automated build, test, and deployment, reducing manual effort by approximately 50%.• API Development: Developed high-performance REST APIs using ASP.NET Core 8 and EF Core with optimized query execution.• Dual Database Integration: Integrated PostgreSQL for transactional operations and AWS Redshift for forecasting and analytics data pipelines.• Code Quality & Analysis (SonarQube): Integrated SonarQube with Jenkins pipelines for continuous code quality checks, significantly improving maintainability and reducing bugs.• Tech Stack: C#, .NET 8, ASP.NET Core, EF Core, Clean Architecture, CQRS, Repository Pattern, Mediator Pattern, PostgreSQL, AWS Redshift, SQL, AWS EKS, Docker, Kubernetes, Azure DevOps, CI/CD, Azure AD, JWT, OAuth 2.0, RBAC, OpenTelemetry, Azure Application Insights, SonarQube.	

Projects

LexCheck

USA

07/2019 - 10/2023

- **Founding Backend Engineer:** Founding Backend Engineer for LexCheck, an enterprise contract management platform, built on .NET Core and Clean Architecture.
- **Document Parsing Engine:** Built a parsing engine in .NET Core using ANTLR and **OpenXML** SDK, enabling automated extraction and transformation of structured data from Word/PDF documents.
- **Document Conversion:** Implemented PDF document conversion using the **Ghostsript** library for reliable and high-quality file processing.
- **API Migration:** Migrated over 420 APIs from ASP.NET MVC 4.8 to .NET 6 microservices, improving scalability, maintainability, and reducing response times by more than 30%.
- **Microservices Architecture:** Architected microservices using Clean Architecture, **RabbitMQ**, and event-driven design to enhance performance, flexibility, and reliability.
- **Design Patterns:** Applied the **Factory Design** Pattern in the email module to eliminate conditional logic, improve code readability, and enhance maintainability by approximately 40%.
- **Email System Migration:** Migrated the email handling system from **MailKit** to **MailGun** to overcome bulk email processing limitations, achieving 80% improvement in reliability and delivery performance.
- **Resilience & Reliability:** Implemented retry and resilience mechanisms using Polly, resolving issues caused by system load and improving client satisfaction by 50%.
- **Real-Time Communication:** Integrated **SignalR** to enable real-time push notifications on the client dashboard for instant updates on incoming emails.
- **Revamp Client Side:** Revamped UI/UX design leveraging the latest **Angular** version, incorporating pivotal concepts such as modular components, reactive forms, Directives and efficient routing. Employed advanced Angular features like dependency injection and Observables and RxJS, enhancing user experiences through a streamlined and responsive application interface .
- **CI/CD & Cloud Deployment:** Implemented CI/CD pipelines using GitHub Actions and **Docker**, managing deployments on AWS EC2 and Azure to streamline build, test, and release workflows.
- **Code Quality and Review (SonarQube):** Used **SonarQube** to perform PR-level code analysis and maintain clean, reliable, and compliant code by 90%.
- **Leadership & Agile Delivery:** Led a cross-functional team of five (four developers, one QA), managing sprint planning, daily standups, and client discussions. Actively participated in requirement analysis, task assignment, and mentoring, ensuring timely delivery of production-ready releases.
- **Business Impact:** Delivered multiple sub-products that exceeded client expectations and contributed to LexCheck's **\$17M** Series A funding led by Mayfield Fund.
- **Tech Stack:** Angular, .NET Core, ASP.NET Core, Web API, C#, MYSQL, AWS Services (EC2, Lambda, S3 Bucket, Cognito, CloudWatch, Redis, RabbitMQ, AWS) , Azure services (Azure DevOps, CI/CD pipeline)

Agri Beef

USA

10/2023 - 08/2024

- **UI Development (Blazor & Syncfusion):** Utilized advanced **Blazor** features to enable customizable user settings, grid configurations, and layout preferences, and integrated **Syncfusion** components for interactive dashboards, advanced dropdowns, and responsive UI elements.
- **Caching & Performance (Redis Cache):** Implemented **Redis Cache** to store and share Admin app data across multiple applications, improving system performance and response time by **35%** and reducing redundant requests.
- **Clean Architecture:** Applied the Repository Pattern in .NET Core to maintain clean, modular, and scalable data access logic
- **CI/CD Automation (Azure DevOps):** Implemented automated build, test, and deployment pipelines via Azure DevOps, improving deployment efficiency and reducing manual intervention by **70%**.
- **Code Quality & Analysis (SonarQube):** Integrated SonarQube with Jenkins pipelines to continuously analyze, review, and improve code quality, resulting in a **90% improvement in code maintainability** and cleaner, more reliable builds.
- **Tech Stack:** NET Core, ASP.NET, ASP.NET Core MVC/Web API, C#, SQL Server, Blazor, Azure services (Azure DevOps, CI/CD pipeline), Redis , Jenkins , SonarQube

Shopify - DAX Integrations

USA

09/2024 - 03/2025

- **Data migration:** Utilized **GraphQL** APIs to extract and migrate data from Shopify (Products, Orders, Customers, and Locations) into SQL-based staging tables for downstream processing.
- **Worker services:** Created background workers to automate data synchronization between **Shopify** and **DAX**, handling scheduled updates, error retries, and large-volume data transfers efficiently.
- **Inventory and order sync:** Implemented logic for managing available inventory, order status updates, and shipment confirmations by integrating DAX inventory staging tables with Shopify fulfillment APIs.
- **Accounting and journal entries:** Assisted in the integration of financial transaction workflows, transforming Shopify fulfillment data into DAX journal entries for accurate revenue and cost tracking.
- **System optimization:** Improved data reliability and performance through batch processing and staged API calls, ensuring smooth communication between **multiple 3PL systems** and **DAX**.
- **Tech Stack :** GraphQL, C#, .NET Core, SQL, DAX, Shopify API, Worker Services

Projects

QA System

UK

04/2025 - 07/2025

- **Performance optimization:** Diagnosed and resolved major performance bottlenecks in API endpoints where multiple database calls were executed inside loops.
- **Data access optimization:** Rewrote inefficient queries using LINQ projections and AsNoTracking, fetching all required data in a single optimized call, which improved API **performance** by up to **90%**.
- **Deployment automation:** Created multiple environment-specific appSettings configurations and implemented an Azure DevOps pipeline for automated build, test, and deployment processes.
- **Testing and validation:** Conducted end-to-end testing in Azure environments to validate performance gains and ensure stability before production rollout.
- **Tech Stack :** NET Core, Entity Framework Core, SQL, Azure DevOps, LINQ, CI/CD Pipeline

Student Faculty Portal

Pakistan

2018 - 2019

- **Requirement gathering:** Collaborated with faculty members to gather and analyze requirements for a student–faculty portal to manage final year project allocations and supervision.
- **Front-end development:** Built responsive and user-friendly interfaces using HTML, CSS, JavaScript, and jQuery, ensuring smooth navigation and interaction.
- **Back-end development:** Developed backend logic in ASP.NET using C#, integrating with SQL Server through ADO.NET and stored procedures for efficient data handling.
- **Database design:** Designed and optimized SQL tables and stored procedures to manage student, faculty, and project information securely and efficiently.
- **Tech Stack :** C#, ASP.NET (Razor), ADO.NET, SQL, HTML, CSS, JavaScript, jQuery