Ajay KUMAR

CAREER OBJECTIVE

Phone:	**7048963056**
Address:	**VILLAGE&POST-TAURA** **DIST-UNNAO** **PIN-209825**
Email:	**ajay32852@gmail.com**

A Full Stack Developer with 5 years of experience, I specialize in both backend and frontend web development, with a strong foundation in RESTful API development, React.js, and Angular.js. I have successfully led projects and handled both the backend and frontend sections of web applications. My primary focus has been on backend development, creating REST APIs, and managing admin panels, while also handling the frontend development using React.js and Angular.js. and Dev Express, And I have an experience AWS 2yr With Jenkins Docker

Experience

P S Intelegencia Analytics Pvt Ltd in Noida

28/JAN/2024 TO 07/JAN/2025

Software Engineer-II

Project Name: TBS (Total Benefit Solution) Insurance

Project URL: TBS Insurance Project -URL:https://stage-qubeupgrade.tbsmga.com/

Project Description:

TBS (Total Benefit Solution) Insurance is a comprehensive ASP.NET Core web application designed to streamline the management and delivery of insurance solutions. This platform integrates a robust set of backend and frontend technologies, ensuring a scalable, secure, and high-performing application.

The project leverages Entity Framework for ORM (Object Relational Mapping), utilizing a DB First approach with Power Tools and AutoMapper for efficient data management and object mapping. The application supports dependency injection, logging with Serilog, and includes advanced transaction management using SQL Server and REST APIs.

Additionally, the platform is enhanced by DevExpress for building responsive, data-driven frontend components and integrates AWS S3 buckets for file storage, enabling zip file upload and download functionalities.

Backend Technologies:

- ASP.NET Core Web Application Development
- REST APIs using Design Patterns (e.g., 3-Tier Architecture)
- Entity Framework (ORM) with DB First Approach
- SQL Server for data storage and management
- Dependency Injection for loose coupling and easier testing
- Serilog for centralized logging and application monitoring
- Transaction Management: Using BeginTransactionAsync, Commit, Rollback for handling database transactions
- AutoMapper for efficient object-to-object mapping
- NuGet Dependency Installer for managing third-party libraries

- Swagger UI for API documentation and testing
- Postman for API testing and client requests
- Synchronous & Asynchronous Programming for optimized performance
- Unit of Work & Repository Pattern for data access abstraction
- Event Streaming: Apache Kafka for real-time data processing.
- Machine Learning: AIML techniques, predictive analytics, customer segmentation, fraud detection.

Frontend Technologies:

- DevExpress: Rich UI components for building responsive and dynamic web interfaces
- HTML, CSS, jQuery for frontend design and functionality
- Responsive Design: Ensuring compatibility with various screen sizes and devices

AWS Services:

- Amazon S3: File storage for managing zip files, enabling upload and download capabilities for large file management

Key Features & Functionalities:

- API-First Development: Developed robust REST APIs with Swagger UI for easy access and testing
- Database Management: Automated database migrations for table creation and structure changes
- File Handling: Integrated AWS S3 for file uploads and downloads, enhancing storage and accessibility
- Logging and Monitoring: Integrated Serilog for error tracking, performance monitoring, and auditing
- Scalable Architecture: Used 3-Tier Architecture and Design Patterns to ensure code reusability, maintainability, and scalability
- Transaction Management: Ensured data integrity with SQL Server using DB transactions

- Based on the project details you provided, here's an optimized description of your TBS (Total Benefit Solution) Insurance project, highlighting the technologies used, and incorporating relevant keywords for backend, frontend, AWS services, full-stack development, and API development.

  ## Project Name: TBS (Total Benefit Solution) Insurance

  Project URL: TBS Insurance Project

  ### Project Description:

  TBS (Total Benefit Solution) Insurance is a comprehensive ASP.NET Core web application designed to streamline the management and delivery of insurance solutions. This platform integrates a robust set of backend and frontend technologies, ensuring a scalable, secure, and high-performing application.

  The project leverages Entity Framework for ORM (Object Relational Mapping), utilizing a DB First approach with Power Tools and AutoMapper for efficient data management and object mapping. The application supports dependency injection, logging with Serilog, and includes advanced transaction management using SQL Server and REST APIs.

Additionally, the platform is enhanced by DevExpress for building responsive, data-driven frontend components and integrates AWS S3 buckets for file storage, enabling zip file upload and download functionalities.

Backend Technologies:

  - ASP.NET Core Web Application Development
  - REST APIs using Design Patterns (e.g., 3-Tier Architecture)
  - Entity Framework (ORM) with DB First Approach
  - SQL Server for data storage and management
  - Dependency Injection for loose coupling and easier testing
  - Serilog for centralized logging and application monitoring
  - Transaction Management: Using BeginTransactionAsync, Commit, Rollback for handling database transactions
  - AutoMapper for efficient object-to-object mapping
  - NuGet Dependency Installer for managing third-party libraries
  - Swagger UI for API documentation and testing
  - Postman for API testing and client requests
  - Synchronous & Asynchronous Programming for optimized performance
  - Unit of Work & Repository Pattern for data access abstraction
  - Event Streaming: Apache Kafka for real-time data processing.
  - Machine Learning: AIML techniques, predictive analytics, customer segmentation, fraud detection.

- Frontend Technologies:
  - DevExpress: Rich UI components for building responsive and dynamic web interfaces
  - HTML, CSS, jQuery for frontend design and functionality
  - Responsive Design: Ensuring compatibility with various screen sizes and devices
- AWS Services:
  - Amazon S3: File storage for managing zip files, enabling upload and download capabilities for large file management
- Key Features & Functionalities:
  - API-First Development: Developed robust REST APIs with Swagger UI for easy access and testing
  - Database Management: Automated database migrations for table creation and structure changes
  - File Handling: Integrated AWS S3 for file uploads and downloads, enhancing storage and accessibility
  - Logging and Monitoring: Integrated Serilog for error tracking, performance monitoring, and auditing
  - Scalable Architecture: Used 3-Tier Architecture and Design Patterns to ensure code reusability, maintainability, and scalability
  - Transaction Management: Ensured data integrity with SQL Server using DB transactions
- **Technologies**
  - Backend Development: ASP.NET Core, Entity Framework, SQL Server, REST

    - APIs, Serilog, AutoMapper, Dependency Injection, 3-Tier Architecture, DB First, Transaction Management
    - Frontend Development: DevExpress, HTML, CSS, jQuery, Responsive Web Design
    - Cloud Integration: AWS S3 Bucket, File Storage, Zip File Upload & Download
    - API Development: Swagger UI, Postman, API Documentation, API Testing, RESTful Services
    - Database & ORM Tools: Entity Framework, DB First Approach, SQL Server, Migration, Power Tools
    - Version Control & Dependency Management: NuGet, Git,bitbucket
    - Design Patterns: Repository Pattern, Unit of Work, Dependency Injection, MVC

Dotsquares Technology Pvt Ltd

JUNE-2022 TO JUNE 2023

Programmer(AS Full Stack Developer)

Project Name: E-Commerce Portal (B2B, B2C)

Duration: June 2022 - June 2023

Technologies Used:

- Backend: ASP.NET Core Web Application, Entity Framework (ORM), SQL (PostgreSQL), REST APIs, Stripe Payment Gateway Integration, JWT Token Authentication, Dependency Injection, Transaction Management (BeginTransactionAsync, Commit, Rollback)
- Frontend: HTML, jQuery, CSS, AJAX, Bootstrap, React.js
- Database: PostgreSQL,Sql Server
- Version Control & Dependency Management: NuGet, Git,bitBucket
- CI/CD: Jenkins for continuous integration and continuous deployment, automated testing and deployment pipelines
- Version Control & Dependency Management: NuGet, Git
- Design Patterns: Repository Pattern, Unit of Work, Dependency Injection, MVC
- Event Streaming: Apache Kafka for real-time data processing.
- Machine Learning: AIML techniques, predictive analytics, customer segmentation, fraud detection.

Project Description:

The E-Commerce Portal (B2B, B2C) is a robust online platform designed for both customers and vendors. The project provides services for customers to browse products, add them to their cart, and complete purchases via a streamlined checkout process. The system integrates the Stripe Payment Gateway API for secure payment handling and uses JWT tokens for secure API authentication.

As a backend developer, I worked on building and maintaining RESTful APIs for the portal's React.js frontend, ensuring smooth data flow and user interactions. I also integrated migration-based database management using Entity Framework to create, update, and manage the database schema. The system supports database transaction management, which ensures that all order processes are consistent and reliable.

Key Responsibilities:

- Designed and developed REST APIs to support B2B and B2C operations (product browsing, order management, cart management, etc.).

- Integrated Stripe Payment Gateway for secure payment processing.
- Implemented secure JWT authentication to protect sensitive API endpoints.
- Developed database migration scripts for seamless schema updates in the PostgreSQL database.
- Utilized Swagger UI for API documentation and testing.
- Worked closely with the React.js frontend to ensure smooth integration and API consumption.

Project Name: E-Commerce (B2C)

Technologies Used:

- Backend: ASP.NET Core, Entity Framework, SQL Server, REST APIs, JWT Token Authentication, Swagger UI
- Frontend: Angular 11, HTML, jQuery, CSS
- CI/CD: Jenkins for automating deployments and CI/CD pipelines.

Project Description:

The E-Commerce (B2C) platform facilitates seamless interactions between admin and customers. The admin can manage product listings, orders, and promotional coupons, while customers can view products, place orders, and apply coupons for discounts during checkout.

I developed and deployed secure REST APIs using ASP.NET Core and Entity Framework, integrated with SQL Server. I also implemented JWT-based authentication to ensure secure user sessions. The frontend is built with Angular 11, ensuring an interactive and modern user experience.

Key Responsibilities:

- Designed APIs to handle product management, order management, and coupon services.
- Integrated JWT for secure user and admin authentication.
- Developed automated testing and deployment pipelines using Jenkins for CI/CD.
- Created Swagger UI for API documentation, ensuring transparency and ease of use for API consumers.

Project Name: E-Commerce Portal (MVC)

Technologies Used:

- Backend: ASP.NET MVC, C#, Stripe Payment Gateway, Entity Framework
- Frontend: HTML, jQuery, Bootstrap, Ajax
- Database: MySQL

Project Description:

The E-Commerce Portal (MVC) is a classic e-commerce solution that provides a simple, secure platform for customers to shop online. The portal supports product browsing, checkout, and payment processing through Stripe Payment Gateway.

The backend uses ASP.NET MVC with Entity Framework for data access and Stripe for payment processing. The frontend utilizes jQuery, Bootstrap, and Ajax to create a responsive and interactive shopping experience.

Key Responsibilities:

- Integrated the Stripe Payment Gateway for payment processing.
- Used ASP.NET MVC for building the backend architecture and data models.
- Developed the frontend with HTML, jQuery, Ajax, and Bootstrap for a seamless

user interface.

Project Name: LEELOS Supermarket LLC (B2C)

Technologies Used:

- Backend: .NET MVC, C#, Stripe Payment Gateway, Entity Framework, SQL Server
- Frontend: HTML, jQuery, Bootstrap, Ajax
- CI/CD: Jenkins for building deployment pipelines

Project Description:

LEELOS Supermarket LLC (B2C) is an e-commerce platform designed for a supermarket chain, offering B2C services. Customers can browse products, add them to the cart, and complete their purchase using the integrated Stripe Payment Gateway. The project was developed using .NET MVC for the backend and utilizes SQL Server for database management.

Key Responsibilities:

- Developed backend services using .NET MVC to manage product inventory and handle customer orders.
- Integrated Stripe Payment Gateway to enable secure online payments.
- Used Entity Framework for database management and SQL Server for product and transaction storage.
- Built the frontend with HTML, jQuery, Bootstrap, and Ajax to provide an interactive and user-friendly experience.

- Here is a well-structured project description along with technologies used based on the details you provided. This will work well for your resume or LinkedIn profile and incorporates CI/CD tools like Jenkins.

  ## Project Name: E-Commerce Portal (B2B, B2C)

  Duration: June 2022 - June 2023

  ### Technologies Used:

  - Backend: ASP.NET Core Web Application, Entity Framework (ORM), SQL (PostgreSQL), REST APIs, Stripe Payment Gateway Integration, JWT Token Authentication, Dependency Injection, Transaction Management (BeginTransactionAsync, Commit, Rollback)
  - Frontend: HTML, jQuery, CSS, AJAX, Bootstrap, React.js
  - Database: PostgreSQL
  - CI/CD: Jenkins for continuous integration and continuous deployment, automated testing and deployment pipelines

- ### Project Description:

  The E-Commerce Portal (B2B, B2C) is a robust online platform designed for both customers and vendors. The project provides services for customers to browse products, add them to their cart, and complete purchases via a streamlined checkout process. The system integrates the Stripe Payment Gateway API for secure payment handling and uses JWT tokens for secure API authentication.

  As a backend developer, I worked on building and maintaining RESTful APIs for the portal's React.js frontend, ensuring smooth data flow and user interactions. I also integrated migration-based database management using Entity Framework to create, update, and manage the database schema. The system supports database transaction management, which ensures that all order processes are consistent and reliable.

Key Responsibilities:

- Designed and developed REST APIs to support B2B and B2C operations (product browsing, order management, cart management, etc.).
- Integrated Stripe Payment Gateway for secure payment processing.
- Implemented secure JWT authentication to protect sensitive API endpoints.
- Developed database migration scripts for seamless schema updates in the PostgreSQL database.
- Utilized Swagger UI for API documentation and testing.
- Worked closely with the React.js frontend to ensure smooth integration and API consumption.

- ## Project Name: E-Commerce (B2C)

  Technologies Used:

  - Backend: ASP.NET Core, Entity Framework, SQL Server, REST APIs, JWT Token Authentication, Swagger UI
  - Frontend: Angular 11, HTML, jQuery, CSS
  - CI/CD: Jenkins for automating deployments and CI/CD pipelines.

- Project Description:

  The E-Commerce (B2C) platform facilitates seamless interactions between admin and customers. The admin can manage product listings, orders, and promotional coupons, while customers can view products, place orders, and apply coupons for discounts during checkout.

  I developed and deployed secure REST APIs using ASP.NET Core and Entity Framework, integrated with SQL Server. I also implemented JWT-based authentication to ensure secure user sessions. The frontend is built with Angular 11, ensuring an interactive and modern user experience.

  Key Responsibilities:

  - Designed APIs to handle product management, order management, and coupon services.
  - Integrated JWT for secure user and admin authentication.
  - Developed automated testing and deployment pipelines using Jenkins for CI/CD.
  - Created Swagger UI for API documentation, ensuring transparency and ease of use for API consumers.

- ## Project Name: E-Commerce Portal (MVC)

  Technologies Used:

  - Backend: ASP.NET MVC, C#, Stripe Payment Gateway, Entity Framework
  - Frontend: HTML, jQuery, Bootstrap, Ajax
  - Database: MySQL

- Project Description:

  The E-Commerce Portal (MVC) is a classic e-commerce solution that provides a simple, secure platform for customers to shop online. The portal supports product browsing, checkout, and payment processing through Stripe Payment Gateway.

The backend uses ASP.NET MVC with Entity Framework for data access and Stripe for payment processing. The frontend utilizes jQuery, Bootstrap, and Ajax to create a responsive and interactive shopping experience.

Key Responsibilities:

- Integrated the Stripe Payment Gateway for payment processing.
- Used ASP.NET MVC for building the backend architecture and data models.
- Developed the frontend with HTML, jQuery, Ajax, and Bootstrap for a seamless user interface.

- **Project Name: LEELOS Supermarket LLC (B2C)(USE DESGIN PATTERN CQRS)**

  Technologies Used:

  - Backend: .NET, C#, Stripe Payment Gateway, Entity Framework, SQL Server
  - Frontend: HTML, jQuery, Bootstrap,Angular 11
  - CI/CD: Jenkins for building deployment pipelines
  - Design Pattern: Command and Query Responsibility Segregation (CQRS)
  - USE ROW QUERY FOR GET DATA AND Add and Update Record

- Project Description:

  LEELOS Supermarket LLC (B2C) is an e-commerce platform designed for a supermarket chain, offering B2C services. Customers can browse products, add them to the cart, and complete their purchase using the integrated Stripe Payment Gateway. The project was developed using .NET CQRS for the backend and utilizes SQL Server for database management.

  Key Responsibilities:

  - Developed backend services using .NET to manage product inventory and handle customer orders.
  - Integrated Stripe Payment Gateway to enable secure online payments.
  - Used Entity Framework for database management and SQL Server for product and transaction storage.
  - Built the frontend with HTML, jQuery, Bootstrap, and Angular 11 to provide an interactive and user-friendly experience.

- **CI/CD and Automation:**

  - Jenkins: Implemented Jenkins for automating continuous integration and continuous deployment, ensuring that each project was tested and deployed in a seamless, automated pipeline. This process helped reduce deployment time and errors while improving overall project efficiency.

- **Technologies :**

  - Backend: ASP.NET Core, .NET MVC, C#, Entity Framework, REST APIs, Stripe Payment Gateway, SQL Server, PostgreSQL, MySQL, DB Transaction Management, Swagger UI, JWT Authentication
  - Frontend: React.js, Angular 11, HTML, CSS, jQuery, Bootstrap, Ajax, Responsive Web Design
  - Payment Integration: Stripe Payment Gateway, Payment API Integration
  - Database: SQL Server, PostgreSQL
  - CI/CD: Jenkins, Automated Testing, Deployment Pipelines, Continuous

  Integration
  - Cloud: AWS S3, File Storage, Zip File Upload & Download
  - API Testing: Swagger UI, Postman, RESTful APIs
  - Security: JWT Token Authentication, Secure Payment Processing
  - Command and Query Responsibility Segregation (CQRS)
  - use Dapper For Row Query Execution

Next Olive Technologies

DURATION: JANUARY 2019 - JUNE 2022

Role: Associate Software Engineer

Project Name: Visa Code Freelancer Project

Technologies Used:

- Backend: ASP.NET MVC 4, Entity Framework, Web API, C#
- Frontend: CSHTML, , HTML, CSS, Angular 11
- Routing: Route Configuration, API Routing

Project Description:

The Visa Code Freelancer Project is an online marketplace designed to connect freelancers with potential clients in various industries. The project allows freelancers to create profiles, offer their services, and manage their projects while clients can post jobs, browse freelancers, and hire based on skills.

The system was developed using ASP.NET MVC and Entity Framework, with a focus on building and managing RESTful APIs for handling marketplace functionalities. The frontend was built using CSHTML and Razor Pages, providing dynamic content rendering for seamless user interactions.

The routing system was carefully managed through Route Config files, ensuring efficient navigation for both views and APIs.

Key Responsibilities:

- Developed and maintained REST APIs to support user interactions and backend logic for freelancer and client operations.
- Implemented Route Config for proper routing of views and API calls.
- Worked with Entity Framework for database interactions and data management.
- Developed the frontend using Razor Pages, CSHTML, HTML, CSS, and jQuery to create a responsive and user-friendly interface.

Project Name: Code Game Management

Technologies Used:

- Backend: Asp .Net MVC 4 USE Entitiy Use
- Frontend: HTML, CSS, jQuery,Angular 11
- Database: SQL Server, DB Transaction Management

Project Description:

Code Game Management is an internal platform designed for managing game-related data, including game listings, game statuses, and user interactions. The project supports managing large-scale game information and automating various backend processes using ASP.NET MVC and Web API.

The Entity Framework was used to handle data operations efficiently, and DB Transaction Management was implemented for consistency during complex database operations.

Key Responsibilities:

- Developed backend services using ASP.NET MVC and Web API for managing games, game data, and interactions.
- Integrated Entity Framework to manage data transactions and streamline database interactions.
- Ensured proper data consistency with DB Transaction Management during database operations.
- Built a responsive frontend interface using HTML, CSS, and jQuery for optimal user experience.

Project Name: School Management System

Technologies Used:

- Backend: ASP .NET MVC 4
- Frontend: Ajax, jQuery, Bootstrap,Angular 11
- Database: SQL Server

Project Description:

The School Management System was developed to help educational institutions manage various aspects of their operations. The system includes modules for adding students, teachers, and managing academic records, attendance, and grades. It also facilitates administrative tasks like scheduling, reporting, and fee management.

Developed using ASPX and C#, the application supports integration with SQL Server for managing student and teacher data, utilizing ADO.NET for data access.

Key Responsibilities:

- Developed backend functionality for handling student and teacher records using ASP .NET MVC 4 and C#.
- Integrated ASP .NET MVC 4 for efficient data access and management in SQL Server.
- Used Ajax, jQuery, and Bootstrap to enhance the frontend, providing a responsive and smooth user interface.
- Implemented essential school management features like student registration, teacher management, and attendance tracking.

Education

Kanpur University

JUNE 2018 TO AUG 2020

Bachelor Of Science

A Bachelor of Science in Computer Science (B.Sc. CS) provided a strong foundation in computer systems, software development, programming, and information technology. The program covered a wide range of topics, including data structures, algorithms, database management, networking, software engineering, and computer architecture.

Throughout the course, I gained proficiency in key programming languages such as Java, C++, Python, and C#, and honed skills in problem-solving, debugging, and system design. Additionally, I worked on various projects that involved full-stack development, web application development, and software testing, which further strengthened my practical and theoretical knowledge in IT and computer science.

The program also offered a deep dive into data analysis, artificial intelligence, machine learning, cloud computing, and cybersecurity, equipping me with the latest industry-relevant tools and techniques to solve complex technical challenges.

GPS

AUG 2014 TO JUNE 2017

Diploma of Computer Science in Computer Science

Diploma of Computer Science in Computer

Rps Inter Collage Lucknow

JULY 2011 TO 2013

12th Grade | intermediate in Science

12th Grade | intermediate in Science

MGIC COLLAGE UNNAO

JULY 2009 TO MARCH 2011

10th in Science

10th in Science

Skills

- **Technical Skills**

  Programming Languages & Frameworks:

  - C# (Expert in OOP, asynchronous programming, and multi-threading)
  - ASP.NET Core (Advanced knowledge of MVC, Web API, and performance optimization)
  - ASP.NET MVC 4/5 (Experience in building robust web applications)
  - .NET Web API (Designing and implementing RESTful APIs with authentication and authorization)
  - Entity Framework (Code-first and database-first approaches, LINQ, migrations, performance tuning)
  - LINQ (Efficient querying and manipulation of data)
  - Dapper (Advanced data handling and database connections)
  - **Asynchronous Programming**:
    - Implementing asynchronous programming patterns using async and await and Task to improve application responsiveness and scalability, especially for I/O-bound operations.
    - use Multithreading in my application
    - To achieve low latency in .NET Core, focus on efficient code, optimized garbage collection, and network performance, using techniques like asynchronous operations, in-memory caching, and efficient serialization
  - **Background Processing:**
    - Implementing background tasks using IHostedService or BackgroundService in .NET Core for long-running operations that do not block the main application thread.

- **Frontend Technologies:**
  - Angular 11 (Component-based architecture, state management, services, and directives)
  - React.js (Component-driven UI development, hooks, context API, and state management)
  - HTML5, CSS3 (Responsive design, Flexbox, Grid, preprocessors like SASS)
  - JavaScript (Advanced JavaScript ES6+ features, DOM manipulation, event handling)
  - jQuery (AJAX requests, DOM traversal, and manipulation)
  - Dev Express
- **Cloud & DevOps:**
  - AWS (S3, EC2, DynamoDB, Lambda functions, and CloudWatch for monitoring and scaling)
  - Jenkins (CI/CD pipeline setup, automation, integration with Git repositories)
  - Docker (Containerization of applications for local and production environments)
  - Terraform (Infrastructure as Code, provisioning AWS resources)
- **Database Technologies:**
  - SQL Server (Stored procedures, triggers, optimization, and query tuning)
  - PostgreSQL (Schema design, indexing, and performance optimization)
  - DynamoDB (NoSQL database management, data modeling)
  - Entity Framework Core (Database migrations, repository pattern, unit of work,cqrs)
- **Version Control & Collaboration:**
  - Git (Branching strategies, merge conflicts, and pull requests)
  - Bitbucket (Source control and collaboration with teams)
  - GitHub (Public and private repository management)
  - Turtle Git Tool (Git operations, commit management)
  - JIRA (Task tracking, sprint planning, Agile project management)
- **API Development & Testing:**
  - Swagger (API documentation, specification generation, and testing)
  - Postman (API testing, integration, and automation)
  - JWT Authentication (Implementing secure user authentication for APIs)
  - OAuth 2.0 (API security, role-based access control)
- **Software Architecture & Design:**
  - Design Patterns (Singleton, Factory, Observer, Repository.)
  - Microservices Architecture (Service-based architecture, API gateways)
  - Clean Architecture (Separation of concerns, maintainability, scalability)
  - Dependency Injection (DI container setup, inversion of control)
  - CQRS (Command Query Responsibility Segregation) (Pattern for read/write separation)
  - Event-Driven Architecture (Message queues, event sourcing, and domain-driven design)
  - Use Kafka For Messaging
- **Leadership & Management Skills:**

  - Team Leadership (Mentoring and guiding junior developers, team management)
  - Agile Methodologies (Scrum,, Sprint Planning, Standups,Jira)
  - Project Management (Risk management, estimation, client interaction, backlog grooming)
  - Code Reviews (Enforcing coding standards, best practices, and peer reviews)
  - Collaboration Tools (Microsoft Teams, Confluence)
- **Testing & Quality Assurance:**
  - Unit Testing (xUnit, NUnit, MSTest)
  - Test-Driven Development (TDD) (Creating tests before writing business logic)
  - Integration Testing (Ensuring different modules work together)
  - Mocking Frameworks (Moq, NSubstitute)
- **Machine Learning:** AIML techniques, predictive analytics, customer segmentation, fraud detection.
- **Event Streaming:** Apache Kafka for real-time data processing.
- **Soft Skills:**
  - Problem Solving (Solving complex technical issues and proposing innovative solutions)
  - Communication (Clear and concise communication with stakeholders, product owners, and teams)
  - Mentoring & Coaching (Helping junior developers grow and upskill in technologies)
  - Time Management (Managing priorities and meeting deadlines)
  - Stakeholder Management (Interfacing with clients, business leaders, and internal teams)
- **Key Achievements**:
  - Successfully led the implementation of DDD principles in a major insurance project, resulting in a 30% reduction in system complexity and improved maintainability.
  - Developed bounded contexts and aggregates that streamlined insurance workflows, enhancing user experience and operational efficiency.