**DevOps**

**Programme name-  MSc Cloud and Network Security and MSc Software**

**Engineering**

**Module title - SWE7303 DevOps**

**Assessment title - SWE7303**

**Student number – 2333995,2332568,2315205**

**Marking tutor – Aamir Abbas**

Contents

[Introduction 3](#_Toc163957566)

[Project Requirement 4](#_Toc163957567)

[Use case Diagram 6](#_Toc163957568)

[Activity Diagram 7](#_Toc163957569)

# Introduction

This is the discussion about, how we can improve a blog application using DevOps tool. The blog app is build using Java, Spring Boot, Maven, and other technologies like AWS, Docker, JWT, React, and PostgreSQL.

The main aim is to use DevOps to simplify the process of building, testing, and deploying the app. We'll use Docker to package the app and its dependencies, AWS to handle the servers where the app runs, and Maven to automate building tasks. To make things even smoother, we'll set up Continuous Integration and Continuous Deployment (CI/CD) pipelines. These pipelines will automatically test new changes and deploy them, so developers don't have to spend time doing it manually. using AWS services like EC2 for scaling computing power, RDS for managing databases easily, S3 for storing and getting blog content, and IAM for controlling who can access what.

# Project Requirement

Design an online platform, where people can share their thoughts and stories with the world. With social media and other web pages, this blog allows you to talk about things in more detail and connect with readers in an interesting way. Whether you might have hobby in sharing your innovative ideas about technology, cooking or a business tips in your specialized area by blogging to helps you reach a wider audience and build a community around your interests.

**Key Features**

The project required to develop a blog website where users can post article, and interact with blog posts. The website aims to provide a platform for users to share their ideas, experiences with a people. main features such as user registration and authentication, blog post creation and management, commenting, and search functionality. The website should be built in modern web technologies, like Java/Spring Boot for the backend, React for the frontend, and PostgreSQL for the database. the website will be hosted on a cloud platform such as AWS, ensuring scalability, reliability, and security. The project must utilize best practices of DevOps tool and web development, technique including building, testing, and deployments.

1. **Functional Requirements:**

* **User Management:**
* User registration.
* User login/logout functionality.
* User profile management.
* **Blog Management**:
* Create, edit, and delete blog posts.
* View published blog posts.
* Commenting on blog posts.
* **Search Functionality:**
* Allow users to search for blog posts based on keywords or categories.
* User Interaction:
* Ability to like/start rating blog posts.
* Can share blog posts on social media platforms.
* **Administration:**
* Admin dashboard to managing users, blog posts, comments, etc.
* Admin should have access for approving/rejecting user comments.
* **Security:**
* Implementation of authentication and authorization mechanisms.
* Protection against common security threats such as SQL injection, XSS, CSRF, etc.

1. **Non-Functional Requirements:**

* **Performance:**
* Fast loading times for web pages.
* Should handle traffic and content.
* **Usability:**
* User friendly interface.
* Responsive design for mobile devices.
* **Reliability:**
* Minimal downtime and high availability of the website.
* Regular backups of data to prevent data loss.
* **Security:**
* Encryption user passwords and sensitive user data.
* Protection against unauthorized access.

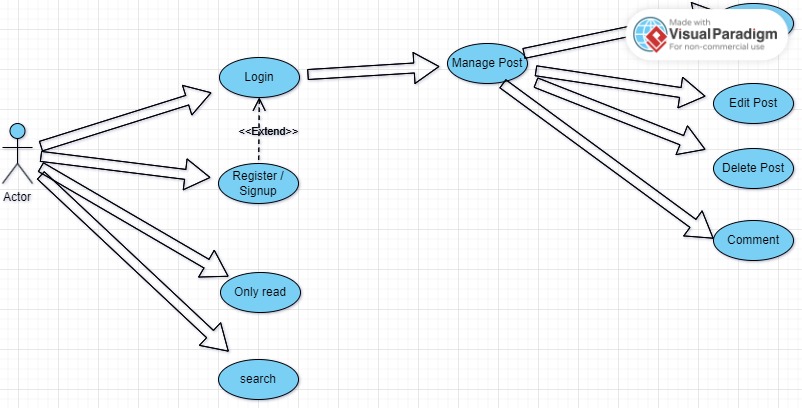
1. **Technology Stack:**

* **Backend**:
* Programming languages and frameworks (e.g., Java/Spring Boot).
* Database management system (e.g., PostgreSQL).
* Web API authentication tool (e.g., JWT for authentication).
* **Frontend:**
* JavaScript libraries/frameworks (e.g., React).
* HTML/CSS for styling.
* **Infrastructure:**
* Cloud platform (e.g., AWS) for hosting.
* Continuous Integration/Continuous Deployment (CI/CD) tools. (Jenkins)

1. **Testing**:

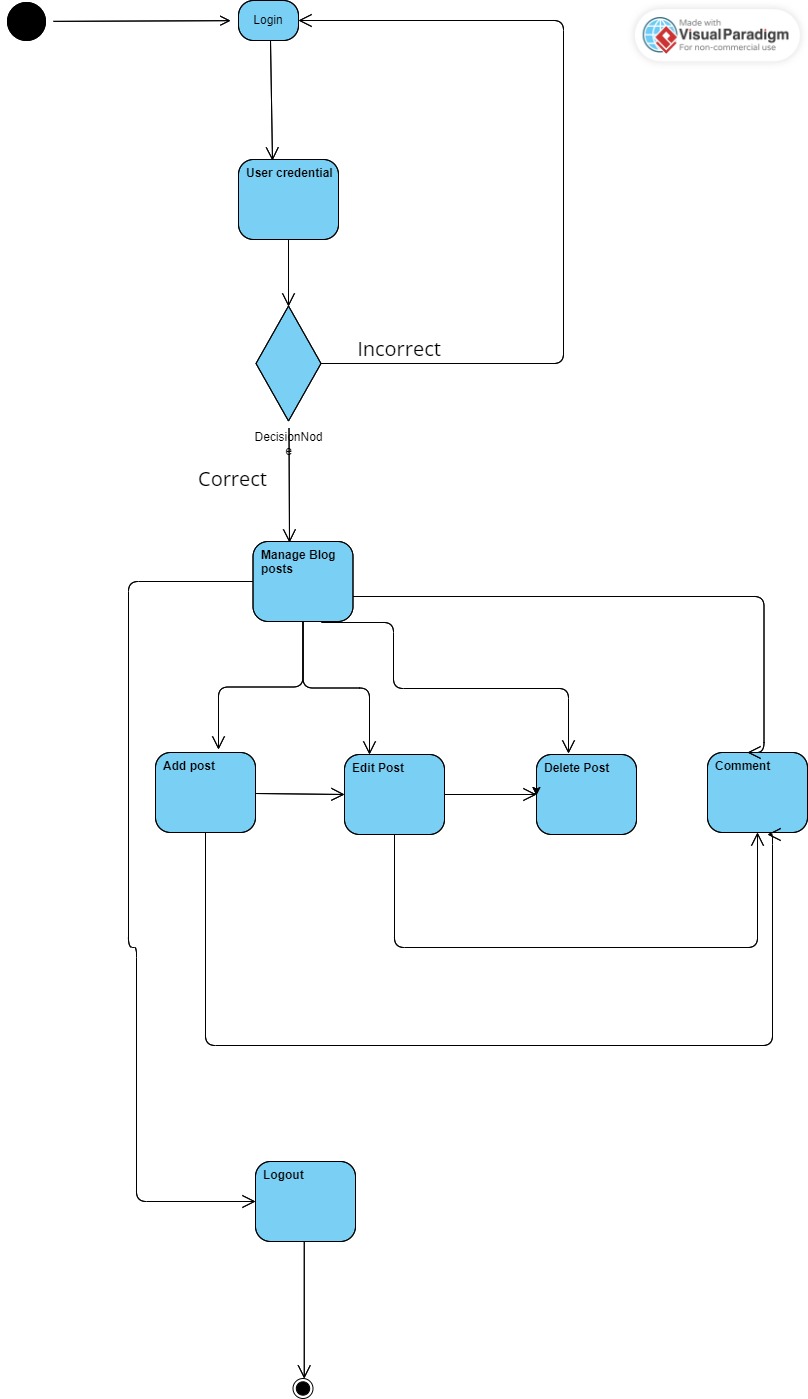
* Unit testing of backend and frontend.
* Integration testing to make sure different parts of the system working fine.
* End-to-end testing to validate the entire user flow.

# Use case Diagram



Use case diagram 1.1

# Activity Diagram



Activity Diagram 1.1