📄 Project Report – CI/CD Pipeline with GitHub Actions & Docker

🔹 Introduction

This project was developed as part of the **DevOps Internship Program**. The primary goal was to design and implement a **CI/CD pipeline** for a simple Node.js To-Do application. The pipeline ensures that every code change is automatically tested, containerized, and deployed, following modern DevOps practices.

🔹 Abstract

The project demonstrates the **end-to-end DevOps workflow** starting from code development, containerization using Docker, automated CI/CD pipelines with **GitHub Actions**, and deployment using **Kubernetes (Minikube)**. This hands-on implementation helped gain real-world exposure to industry-standard DevOps tools and workflows.

🔹 Tools Used

- **Node.js & Express.js** → Application development
- **Docker** → Containerization of the app
- **GitHub Actions** → CI/CD automation pipeline
- **Kubernetes (Minikube)** → Deployment and scaling
- **kubectl** → Kubernetes cluster management
- **Git & GitHub** → Version control and collaboration

🔹 Steps Involved in Building the Project

1. **App Development:** Built a modern Node.js To-Do application with task management and leaderboard features.

2. **Containerization:** Created a Dockerfile to package the app into a portable Docker image.

3. **CI/CD Setup:** Configured GitHub Actions workflow to build and push the Docker image to DockerHub on every push to the `main` branch.

4. **Kubernetes Deployment:**

    - Created `deployment.yaml` for pod replicas.

    - Created `service.yaml` for exposing the app.

    - Verified running pods and services using `kubectl`.

5. **Scaling & Rollout:** Scaled the app to 3 replicas and tested successful rollout.

6. **Access Application:** Exposed the service using Minikube and accessed the app in the browser.

🔹 Conclusion

This project successfully implemented a **CI/CD pipeline** integrated with containerization and Kubernetes deployment. It enhanced practical knowledge in DevOps by covering key areas such as automation, container orchestration, and scalable deployments. The experience also strengthened understanding of how modern organizations streamline software delivery using DevOps practices.

📌 Author: **Chetan Vani**