TEAM VY quad S | CSE-D | 5TH SEM

WEATHER APP DEPLOYMENT

PROJECT SUMMARY

REPORT DATE	PROJECT NAME	PROJECT MANAGER
09-11-2024	Continuous Integration, Delivery, and Deployment In AWS, Netlify, GitHub, Dockers and Vercel.	Yashwanth KR

EXECUTIVE SUMMARY

This weather app project aims to deliver a reliable, user-centric platform that provides real-time weather data with precision and accessibility. Designed with a responsive interface, the app ensures a consistent experience across all devices, meeting users wherever they are.

PROJECT OVERVIEW

TASK	% DONE	DUE DATE	DEVOPS FACILITATOR	MILESTONES
Planning Stage	100	5-11-24	Suhas U	Meetings and Discussions about our project selection
Development Stage	100	6-11-24	Vishal D	HTML, CSS, JS code development
Testing Stag	100	6-11-24	Sujay Matur	Testing the proper functionality of our codes
Deployment Stage	100	7-11-24	Sudarshan B	Successful deployment using AWS, Vercel, GitHub, Docker and Netlify
Monitoring Stage	100	8-11-24	Sanjay MM & Kiran KS	Successful documentation of all our processes
Feedback Stage	100	9-11-24	Yashwanth K R	Review and minor alterations after reviews and feedback

MAN-HOURS

CATEGORY	SPENT	% OF TOTAL	ON TRACK?	NOTES
Planning and Assessment	1 hr	2	YES	Importance of planning
Requirements gathering:	30 mins	1	YES	————————
Application assessment:	30 mins	1	YES	————————

DevOps strategy planning	1 hr	2	YES	Role of AI Tools
Tool selection and configuration	30 mins	1	YES	—————
Infrastructure Setup	30 mins	1	YES	—————
Cloud infrastructure setup (AWS/Azure/GCP)	1 hr	2	YES	Introduction to new services
Containerization (Docker):	4 hr	8	YES	—————
Orchestration (Kubernetes)	1 hr	2	YES	New service
Monitoring and logging setup	30 mins	1	YES	—————
Application Integration	1 hr	2	YES	
Code repository setup (Git)	1 hr	2	YES	Usage of GitHub
Continuous Integration/Continuous Deployment (CI/CD) pipeline setup	30 mins	1	YES	—————
Automated testing setup	30 mins	1	YES	—————
Vulnerability management	30 mins	1	YES	—————
Security and Compliance	30 mins	1	YES	—————
Deployment automation	30 mins	1	YES	—————
Security assessment	2 hr	4	YES	—————
Compliance setup	2 hr	4	YES	—————
Access control and identity management	3 hr	6	YES	—————
Testing and Quality Assurance	3 hr	6	YES	—————
Test planning	4 hr	8	YES	Need of work division
Test execution	3 hr	6	YES	—————
Defect tracking and resolution:	4 hr	8	YES	—————
Quality assurance	5 hr	10	YES	Need of scrum meetings
Deployment and Maintenance	6 hr	12	YES	—————
Deployment planning	1 hr	2	YES	—————
Deployment execution	30 mins	1	YES	Running our project online
Post-deployment monitoring	30 mins	1	YES	—————
Maintenance and support	30 mins	1	YES	—————

STAKEHOLDERS

STAKEHOLDER	USN	KEY RESPONSBILITY AREA
Yashwanth KR	4NI22CS254	Project Manager
Sudarshan B	4NI22CS218	DevOps Engineer (Docker, AWS)
Suhas U	4NI22CS222	DevOps Engineer (Vercel, Netlify)
Sujay Matur	4NI22CS225	DevOps Engineer (GitHub, Netlify)
Vishal D	4NI22CS247	Developer and QA Engineer
Sanjay MM	4NI22CS190	Developer
Kiran KS	4NI23CS411	Developer and Designer

PROJECT OVERVIEW

This weather app project leverages DevOps practices to enhance the delivery, reliability, and scalability of a real-time weather data platform. Through continuous integration and continuous deployment (CI/CD) pipelines, DevOps enables rapid updates, seamless testing, and efficient deployment to ensure the app is always up-to-date and stable. DevOps automation facilitates environment provisioning and configuration, making deployments consistent and reducing manual intervention

.

KEY OBJECTIVES:

The primary objective of the weather app project is to deliver accurate, real-time weather information through an intuitive and user-friendly interface. This includes implementing a responsive design to ensure seamless functionality and accessibility across various devices and screen sizes. Additionally, the project aims to leverage an efficient backend architecture to facilitate fast data retrieval and processing, enhancing performance and user experience. Overall, the goal is to create a reliable and engaging weather application that meets user needs with precision and efficiency.

BENEFITS:

The top three benefits of using DevOps for the weather app project are:

1. Faster Time-to-Market: DevOps enables rapid deployment cycles, allowing quicker releases of features and updates.

2. Enhanced Reliability: Continuous integration and automated testing improve app stability, reducing downtime and ensuring high performance.

3. Scalability: Automated scaling in a DevOps environment allows the app to handle increasing user demand seamlessly, adapting as needed.

LESSONS LEARNED:

The most valuable lessons learned from the weather app project are:

1. The Value of Automation: Automating testing and deployment processes through DevOps significantly improves development speed and reduces human error, proving essential for efficient, reliable releases.

2. The Importance of Scalability: Designing with scalability in mind ensures the app can handle fluctuating user demand without performance issues, a crucial factor for high-traffic applications like weather forecasting.

3. Continuous Monitoring's Role in Quality: Real-time monitoring provides valuable insights into app performance and user experience, highlighting the importance of proactive issue detection and timely optimizations.

FUTURE RECOMMENDATIONS:

Here are some future recommendations using DevOps practices for the weather app project:

1. Integrate Advanced DevOps Analytics: Use DevOps analytics tools to track app performance and user behavior, enabling data-driven improvements and tailored user experiences.

2. Strengthen DevOps Security Measures: Implement DevOps security practices, such as automated vulnerability scans and continuous compliance checks, to enhance data protection and build user trust.

3. Leverage Cloud-Native DevOps Tools for Scalability: Adopt cloud-native DevOps tools, like auto-scaling and serverless architecture, to streamline scaling processes, optimize costs, and improve app resilience under variable traffic loads.

CONCLUSION:

In conclusion, the weather app project demonstrates the powerful impact of DevOps practices in driving faster delivery, enhanced reliability, and scalability. By integrating continuous integration, automated deployment, and cloud-native solutions, DevOps enables seamless updates and performance optimization. The project's success hinges on the collaborative nature of DevOps, fostering efficient workflows, rapid issue resolution, and proactive monitoring. Looking ahead, further strengthening DevOps practices—through advanced analytics, security measures, and cloud-native tools—will continue to enhance the app's performance, user experience, and overall scalability.

METRICS:

Here are some key DevOps metrics for the weather app project:

1. Deployment frequency: Increased by 300%.
2. Deployment time: Reduced by 95%.
3. Application uptime: Improved by 90%.
4. Defect density: Reduced by 20%.
5. Customer satisfaction: Improved by 35%.