NATIONAL UNIVERSITY

OF COMPUTER AND EMERGING SCIENCES

PESHAWAR CAMPUS

PROJECT PROPOSAL

SOFTWARE REQUIREMENTS ENGINEERING

SUBMITTED TO: LECT SARA REHMAT

GROUP MEMBERS:

Muhammad Rehan (22P-9106) (BSSE-3A)

Hasnain Saleem (22P-9123) (BSSE-3A)

Requirements Analysis for Cloud Native Development with CNCF

Introduction:

The Cloud Native Computing Foundation (CNCF) is a significant player in the software industry, providing support and guidance for rapidly growing cloud-native projects such as **Kubernetes**, **Docker**, and **CI/CD**. In this project, we aim to study and analyze the requirements that led to the emergence of cloud-native development, why development has shifted towards the cloud, and why we are focusing on cloud-native for our project.

We will be exploring a variety of tools used in the cloud ecosystem, including:

- Kubernetes (container orchestration)
- Microservices (Node.js)
- Continuous integration and deployment (Gitlab CI/CD)
- Containers (Docker)
- Container infrastructure as code (Terraform)
- Certificates (cert-manager)
- API Gateway (Ambassador)

Our focus, however, will be on understanding the requirements and principles that necessitated the development and use of these tools and technologies.

Why Cloud-Native?

Cloud-native technologies offer several benefits that have driven their adoption. These include:

- **Scalability**: Cloud-native applications can quickly scale up or down to meet demand, which is crucial for growing businesses.
- **Improved Customer Experience**: Cloud-native technologies can help businesses deliver an exceptional customer experience by providing faster, more reliable services.

- **Developer Agility**: Cloud-native development tools can increase developer productivity, enabling faster delivery of new features.
- **CI/CD Integration**: Cloud-native technologies enable continuous integration and continuous deployment, speeding up the development process and reducing the risk of errors.
- **Disposable Infrastructure**: Cloud-native applications leverage the concept of disposable infrastructure, or Infrastructure as Code, which allows for the creation and destruction of resources as needed, laying the groundwork for scalability and elasticity.
- **Decoupling of Services**: Cloud-native applications often comprise decoupled microservices, each independent of the other. It allows for easier bug fixing and avoidance of side effects, as each component can be dealt with individually.

Project Approach:

Our project will conduct a thorough requirements analysis for cloud-native development, focusing on understanding the reasons behind the shift towards the cloud and the principles that underpin cloud-native technologies.

- **Analyzing the Shift to Cloud:** We will explore the reasons behind the shift to cloud-native development, such as the need for scalability, improved customer experience, and developer agility.
- **Exploring Cloud-Native Tools and Technologies:** We will study the tools and technologies used in cloud-native development, including microservices, containers, CI/CD, and more. We will analyze the requirements that necessitated the development and use of these tools.
- **Studying CNCF's Role**: We will study the high-level features of CNCF and how it supports cloud-native projects, focusing on its role in fostering cross-company collaboration, supporting cloud-native projects, and educating the community.
- **Case Study:** Finally, we will apply what we have learned to a case study, demonstrating the practical application of cloud-native technologies.

Conclusion:

In conclusion, our project aims to provide a thorough understanding of the requirements of cloud-native development and the role of CNCF in supporting this shift in the software industry. This knowledge will be invaluable for future software engineering projects, particularly in the area of requirements engineering, and will contribute to the ongoing growth and evolution of cloud-native technologies.

REFERENCES:

1. [The Definitive Guide to Cloud-native Development: All You Need to Know](#)
2. [IBM: What is Cloud Native?](#)
3. [Top 4 Skills to be a cloud-native Developer](#)
4. [7 Cloud Native Development Principles for Maximum Efficiency](#)