NATIONAL UNIVERSITY
OF COMPUTER AND EMERGING SCIENCES
PESHAWAR CAMPUS

PROJECT ASSIGNMENT 01

SOFTWARE REQUIREMENTS ENGINEERING

SUBMITTED TO: LECT SARA REHMAT

GROUP MEMBERS:
Muhammad Rehan (22P-9106) (BSSE-3A)
Hasnain Saleem (22P-9123) (BSSE-3A)

Vision and Scope Document

(CLOUD NATIVE)

1. Business Requirements

1.1 Background

The move to cloud-native solutions happened because of a few main reasons. First, old IT systems were inflexible and costly to run, so companies looked for cheaper and more flexible options. New technologies like containers and Kubernetes made it easier to build and manage software. Companies also wanted to work faster to keep up in a competitive world and needed systems that were reliable and secure. Many had old computer systems that were causing problems, so they decided to switch to cloud-native ways to modernize. A bunch of tools and help from the community made this shift easier for businesses aiming to do well in the digital world.

1.2 Business Opportunity

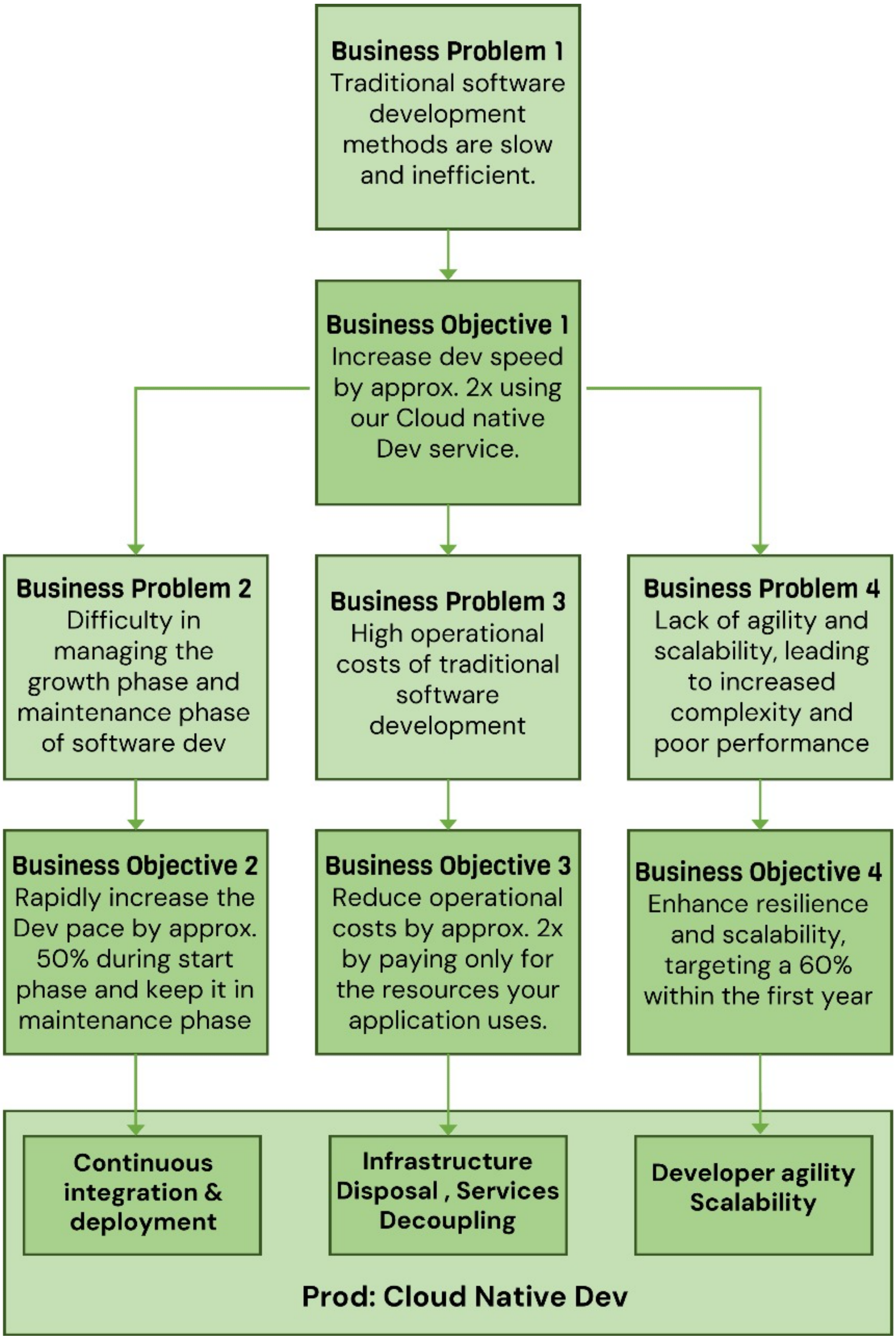
In the ever-evolving digital landscape, businesses seek to enhance scalability, efficiency, and agility in their corporate information systems. Traditional IT infrastructures struggle to meet these demands cost-effectively. Cloud-native solutions, aligning with market trends favoring agility and cost-efficiency, offer seamless scalability, cost savings, rapid software deployment, and improved system resilience. Key components for a complete cloud-native solution include microservices architecture, CI/CD pipelines, container orchestration (e.g., Kubernetes), robust security, DevOps practices, scalable cloud infrastructure, and skill development programs. These elements cater to customer needs such as scalability, cost-efficiency, agility, security, competitiveness, modernization, compliance, and skill development. The new cloud-native product effectively addresses these needs, facilitating seamless scalability, cost efficiency (e.g., server cost reduction during off-peak hours), rapid software updates, and other critical enhancements for a competitive edge.

1.3 Business Objectives

BO-1: During the initial development phase, aim to accelerate development speed by around 50%. Once the project transitions to the maintenance phase, maintain this increased pace to efficiently handle ongoing updates and support.

BO-2: Achieve a 2x reduction in operational costs by paying for resources based on your application's actual usage, optimizing expenditure while maintaining performance.

BO-3: This goal aims to improve system robustness and capacity, with a specific target of achieving a 60% enhancement within the initial year.



Business Problem 1
Traditional software development methods are slow and inefficient.
Business Objective 1
Increase dev speed by approx. 2x using our Cloud native Dev service.
Business Problem 2
Difficulty in managing the growth phase and maintenance phase of software dev
Business Problem 3
High operational costs of traditional software development
Business Problem 4
Lack of agility and scalability, leading to increased complexity and poor performance
Business Objective 2
Rapidly increase the Dev pace by approx. 50% during start phase and keep it in maintenance phase
Business Objective 3
Reduce operational costs by approx. 2x by paying only for the resources your application uses.
Business Objective 4
Enhance resilience and scalability, targeting a 60% within the first year
Continuous integration & deployment
Infrastructure Disposal , Services Decoupling
Developer agility Scalability
Prod: Cloud Native Dev

Figure 1 Business Objectives Model

1.4 Vision Statement

We offer a cloud-native development service **for** businesses and organizations seeking an edge in the competitive and rapidly evolving world of software development. This service is designed to provide a comprehensive understanding of the requirements and principles driving the shift toward cloud-based solutions. **Unlike** traditional software development methods, our approach is focused on scalability, enhanced customer experience, developer agility, continuous integration and deployment (CI/CD), disposable infrastructure, and decoupling of services.

By leveraging the expertise of the Cloud Native Computing Foundation (CNCF), we can deliver software development at approximately **twice (2x) the speed** of traditional methods. Our service thoroughly analyzes the tools and technologies used in cloud-native development, such as Kubernetes, Microservices, Containers, and Terraform. These tools help rapidly increase the development pace **~=50%** in the start phase of development, which continues even in the maintenance phase of the project.

Unlike traditional development methods, **our project** allows businesses to utilize cloud-native technologies' benefits fully. It provides a practical application of these technologies, showcasing the benefits of embracing this paradigm shift diversity.

Compared to the traditional approach, our cloud-native service ensures your applications' availability, resilience, and scalability. It also allows for **twice (2x) cost-effective** operations as you only pay for the resources your application uses. This makes it a more efficient and economical choice for businesses.

With **our project**, businesses will gain a deep understanding of the reasons behind the shift to cloud-native development. This will **enable them** to create innovative, scalable, and reliable software solutions that meet the demands of the modern digital landscape. By choosing **our service**, businesses can ensure they stay ahead of **10x times** the curve and are equipped to meet future challenges in the software development landscape.

1.5 Business Risks

RI-1: This risk pertains to the potential entry of new competitors or intensified competition in the market, which could lead to a loss of market share and reduced profitability. (Probability = 0.4; Impact = 4)

RI-2: This risk involves the possibility of project delays, resulting in missed market opportunities and potential revenue losses.
(Probability = 0.3; Impact = 3)

RI-3: This risk relates to the risk of the product not meeting user expectations, leading to decreased sales and potential damage to the brand's reputation. (Probability = 0.2; Impact = 4)

RI-4: This risk concerns the potential occurrence of a data security breach, resulting in data loss, legal consequences, and damage to the company's reputation. (Probability = 0.5; Impact = 5)

RI-5: This risk involves the possibility of failing to comply with relevant regulations and facing legal penalties, fines, and reputational harm. (Probability = 0.3; Impact = 4)

2. Scope and Limitations

2.1 Major Features

FE-1: Encapsulate applications and their dependencies in portable units, ensuring consistent performance and easy deployment across various environments, enhancing application reliability and scalability.

FE-2: Streamlines repetitive tasks and processes, making software development and operations more efficient and error-free, resulting in faster and more reliable services.

FE-3: Breaks down complex applications into smaller, independently deployable services, enabling quicker development, easier maintenance, and more flexible scalability.

FE-4: Provides standardized interfaces for software components to interact, allowing seamless integration between different services, applications, and systems.

FE-5: Fosters collaboration between development and operations teams, automating software delivery pipelines for faster feature releases and improved product quality.

FE-6: Efficiently manages and scales containers and services, ensuring applications run smoothly in dynamic, cloud-native environments, enhancing performance and resource utilization.

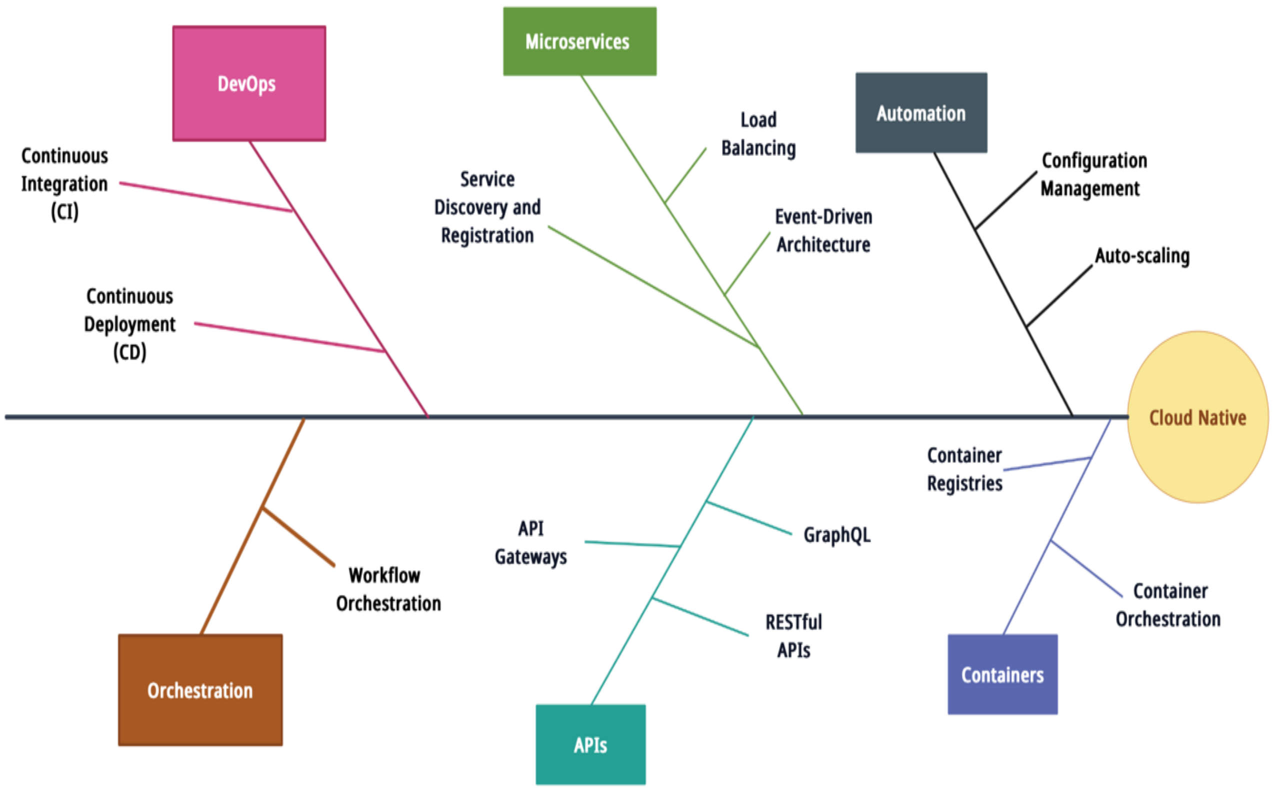


Figure 2 Feature tree for Cloud Native

2.2 Scope of initial and subsequent releases

Feature	Release 1	Release 2	Release 3
FE-1, Scalability	Basic horizontal scaling using load balancers	Advanced horizontal scaling with auto-scaling groups	Full implementation of horizontal and vertical scaling with auto-scaling, load balancing, and caching
FE-2, Enhanced Customer Experience	Basic user interface improvements	Advanced user interface enhancements with responsive design and accessibility features	Full implementation of user interface improvements with optimized performance and personalized experiences
FE-3, Developer Agility	Essential integration with version control systems	Advanced integration with CI/CD pipelines and automated testing	Full implementation of developer agility with continuous deployment, feature flagging, and automatic rollback mechanisms
FE-4, CI/CD	Basic setup of CI/CD pipelines for build and deployment	Advanced CI/CD pipelines with automated testing and release management	Full implementation of CI/CD with continuous integration, automated testing, deployment pipelines, and release orchestration
FE-5, Disposable Infrastructure	Not Implemented	Basic implementation of infrastructure as code using Terraform	Full implementation of disposable infrastructure with automated provisioning, scaling, and teardown using Terraform and Kubernetes
FE-6, Decoupling of Services	Essential service decoupling using API gateways	Advanced service decoupling with event-driven architecture and microservices	Full implementation of service decoupling with microservices, event-driven communication, and service mesh

FE-7, Cloud Native Technologies (Kubernetes, Microservices, Containers, Terraform)	Basic containerization using Docker	Advanced container orchestration with Kubernetes	Full implementation of cloud-native technologies with containerization, orchestration, and infrastructure as code using Kubernetes, Docker, and Terraform

2.3 Limitations

LI-1: The reliance on a stable internet connection can be a critical limitation, as it affects the accessibility and usability of cloud-native applications. Users may experience disruptions or loss of functionality without a reliable internet connection.

LI-2: Users place a high value on the privacy and security of their data. Any perceived or actual breaches of data privacy can have serious consequences, making this a top concern.

LI-3: Users expect cloud services to be highly reliable. Downtime or service outages can significantly impact their ability to access and use cloud-native applications, making service reliability a top priority.

LI-4: Security is paramount for users, as they want assurance that their data and transactions are secure. Any perceived or actual security vulnerabilities can erode trust and confidence in cloud-native solutions.

LI-5: Users are concerned about the financial aspect of cloud-native solutions. The uncertainty of variable costs and potential budgetary concerns can be a significant limitation, especially for businesses and organizations.