



## Semesterarbeit Einreichungsformular / Klasse ITCNE23 – 4. Semesterarbeit

Im Folgenden beschreibt der Studierende das geplante Thema seiner Zertifikatsarbeit. Der Studierende orientiert sich dabei an dem Bewertungsraster. Der begleitende Dozent entscheidet, ob es sich dabei um ein geeignetes Thema handelt und fügt seine Kommentare und Überlegungen hinzu.

Bitte die *kursiven Textteile* durch die konkreten Angaben ersetzen.

### Name und Vorname des Studierenden

*Noah Luchsinger*

### Titel der Semesterarbeit

*Bereitstellung von Microservices mit Kubernetes und CI/CD-Pipeline*

### Zu behandelnde Themenfelder / Module (bitte ankreuzen)

Pkt.	Themenfeld / Modul	
1.	Projektmanagement	Pflicht
2.	DevOps*	Pflicht
3.	CNC - Cloud-native Core*	Pflicht
<b>Themen aus dem 3. Semester</b>		
4.	Relationale Datenbanken	<input type="checkbox"/>
5.	NoSQL, Key/Value, Time Series	<input type="checkbox"/>
6.	Microservices mit Python	X
7.	Datensicherheit / Datenschutz	<input type="checkbox"/>
8.	Netzwerk	<input type="checkbox"/>

\* Als Infrastruktur wird Kubernetes mit mindestens einer CI/CD Pipeline erwartet.

## Kurzbeschreibung des Zertifikatsarbeit Themas (1 bis max. 2 Seiten)

Im 3. Semester habe ich ein Galerieverwaltungssystem mit Gesichtserkennung entwickelt, das auf einem Server in diversen Containern läuft. Das System ist zwar funktional, aber es fehlen wichtige Funktionen wie automatische Skalierbarkeit und Ausfallsicherheit. Ziel dieser Arbeit ist es, das bestehende System Kubernetes-fähig zu machen und eine CI/CD-Pipeline einzurichten, um eine effiziente und fehlerfreie Bereitstellung der Microservices sicherzustellen.

### **Need:**

Das bestehende Microservice-basierte Galerieverwaltungssystem benötigt eine zukunftssichere Plattform, die eine automatische Skalierung und Ausfallsicherheit ermöglicht. Derzeit erfolgt das Deployment manuell, was zu potenziellen Fehlern und Verzögerungen führt. Da das System mit wachsenden Datenmengen und steigender Nutzerzahl umgehen muss, ist eine flexible, automatisierte Lösung notwendig.

### **Approach:**

Das System wird auf Kubernetes migriert, um eine automatische Orchestrierung und Skalierung der Microservices sicherzustellen. Kubernetes ermöglicht es, die Ressourcen je nach Auslastung dynamisch anzupassen. Zudem wird eine CI/CD-Pipeline implementiert, die sicherstellt, dass der Code nach jeder Änderung automatisch getestet und in den Produktions-Cluster integriert wird. Dies erfolgt entweder mit GitHub oder Azure DevOps als CI/CD-Tool.

### **Benefit:**

Durch die Kubernetes-Migration wird das System widerstandsfähiger und kann auf veränderte Lastanforderungen reagieren. Die CI/CD-Pipeline automatisiert den gesamten Bereitstellungsprozess und reduziert menschliche Fehler. Die Nutzung von Kubernetes und CI/CD sorgt dafür, dass die Anwendung hochverfügbar ist, einfach zu warten und flexibel auf zukünftige Anforderungen skalierbar bleibt.

### **Competition:**

Im Vergleich zu herkömmlichen Server- und Containerlösungen bietet Kubernetes eine native Unterstützung für automatische Skalierbarkeit und Ausfallsicherheit. Andere Lösungen erfordern oft manuelle Skalierung oder den Einsatz zusätzlicher Tools, um ähnliche Ergebnisse zu erzielen. Kubernetes bietet eine umfassende Plattform, die nicht nur die Verwaltung erleichtert, sondern auch die Effizienz im Betrieb erheblich steigert.

## Zielsetzung der Semesterarbeit

- Bereitstellung von Microservices auf Kubernetes: Die bereits entwickelten Microservices werden in einem Kubernetes-Cluster deployed und orchestriert, um ihre Skalierbarkeit und Verfügbarkeit zu gewährleisten.
- Automatisierung des Deployment-Prozesses: Eine CI/CD-Pipeline wird eingerichtet, um den Quellcode bei jeder Änderung zu testen und direkt in den Produktions-Cluster zu integrieren. Dies garantiert kontinuierliche Verbesserungen und die zeitnahe Implementierung von Änderungen.
- Sicherstellung der Skalierbarkeit und Ausfallsicherheit: Kubernetes ermöglicht die automatische Skalierung der Microservices basierend auf der Nachfrage, und die Pipeline sorgt für eine problemlose Aktualisierung der Services.

## Terminplan mit den wesentlichen Arbeitsschritten

Sprint	Zeitraum	Ziele
<b>Sprint 1</b>	28.10.24 - 14.11.24	Analyse des bestehenden Systems, Einrichtung des Kubernetes-Clusters und Vorbereitung der Microservices. Laufende Dokumentation der Einrichtung.
<b>Sprint 2</b>	15.11.24 - 15.12.24	Integration der Microservices in den Kubernetes-Cluster und Aufbau der CI/CD-Pipeline. Dokumentation der Integration und CI/CD-Pipeline.
<b>Sprint 3</b>	16.12.24 - 26.01.24	Tests, Optimierungen, automatische Skalierung, Dokumentation der Tests und Abschlussvorbereitungen für die Präsentation.
<b>Abgabe</b>	29.01.2025	Abgabe der Arbeit und Schlusspräsentation.

## Sachmittel / Rahmenbedingungen

- Kubernetes Cluster (in der Cloud oder lokal).
- GitHub oder Azure DevOps für CI/CD.
- Entwicklungsumgebung mit Zugriff auf Testinfrastruktur.
- Jira für die Organisation des Projektplans und der Sprints.

## Vorgaben, Methoden und Werkzeuge

- Kubernetes für die Orchestrierung der Microservices.
- CI/CD-Tools wie GitHub oder Azure DevOps für das automatische Deployment.
- Git zur Versionskontrolle.
- Jira für die Sprint-Planung und -Verwaltung.

## Risiken

- Technische Herausforderungen bei der Einrichtung der Kubernetes-Umgebung und der CI/CD-Pipeline.
- Zeitmanagement: Verzögerungen aufgrund unerwarteter Probleme bei der Implementierung, besonders angesichts des 50-Stunden-Limits.
- Komplexität des Kubernetes-Clusters und der Skalierungsmechanismen könnten zu Schwierigkeiten führen.

## Entscheid des begleitenden Dozenten

Bitte ankreuzen

- Genehmigt ☐
- Zu verbessern ☐
- Abgelehnt ☐

## Begleitender Dozent

Ort und Datum: \_\_\_\_\_

Name & Unterschrift: \_\_\_\_\_

## Beurteilung des Antrages

Kriterium	Kommentare	Erfüllt
<b>Anforderungen an die Form (Strukturierung) der Semesterarbeit</b>		
Problemstellung		<input type="checkbox"/>
Ziele (mindestens drei!)		<input type="checkbox"/>
Risiken bezogen auf Zertifikatsarbeit		<input type="checkbox"/>
Themenfelder (mindestens zwei!)		<input type="checkbox"/>
<b>Anforderungen an Qualität der Semesterarbeit</b>		
Machbar		<input type="checkbox"/>
Praxisnah		<input type="checkbox"/>
Herausfordernd		<input type="checkbox"/>
Lehrgangsbezug		<input type="checkbox"/>

Damit die Semesterarbeit angenommen wird, müssen alle Kriterien erfüllt sein.

Die Kommunikation wird über den jeweiligen Teams-Kanal geführt.