GHENT UNIVERSITY

Course Specifications

Valid as from the academic year 2025-2026

Cloud Computing (E736040)

Course size *(nominal values; actual values may depend on programme)*

Credits 6.0 **Study time 180 h**

Course offerings in academic year 2026-2027

A (semester 2)	English	Gent
B (semester 2)	English	Gent

Lecturers in academic year 2026-2027

Sartor, Jennifer	TW06	staff member
Volckaert, Bruno	TW05	lecturer-in-charge

Offered in the following programmes in 2026-2027	crdts	offering
Bridging Programme Master of Science in Bioinformatics(main subject Engineering)	6	A
Master of Science in Electronics and ICT Engineering Technology(main subject Electronics Engineering)	6	A, B
Master of Science in Electronics and ICT Engineering Technology(main subject Embedded Systems)	6	A, B
Master of Science in Bioinformatics(main subject Engineering)	6	A
Master of Science in Electronics and ICT Engineering Technology(main subject ICT)	6	A
Micro-credential Cloud Computing	6	A
Exchange Programme in Computer Science (master's level)	6	A

Teaching languages

English

Keywords

Cloud, containers, Docker, Kubernetes, microservices, DevOps, agile, container orchestration, distributed storage, infrastructure-as-code, monitoring, web services, REST
For the partim version of 3 ECTS: Cloud, containers, Docker, Kubernetes, microservices, DevOps, agile, container orchestration, infrastructure-as-code, monitoring

Position of the course

As software systems are becoming more and more complex and fully intertwined with our daily lives, the need to operate these systems in a reliable and typically always-on manner is becoming more and more apparent. Companies with great ideas can, over the course of a few months' time, see global uptake of their software products, through proper use of cloud infrastructure in a pay-as-you-go model.
The goal of this course is to get to know how microservice-based applications can lead to more agile ways of developing software and how the ensuing embrace failure mantra of distributed systems can be used to increase the reliability of said applications. The increased complexity of these multi-service applications can be countered by means of automation on numerous levels: introducing DevOps culture, CI/CD pipelines, infrastructure-as-code, testing automation, etc. Container technologies are identified as a prime candidate to deploy such microservices, while managing the complexity of running thousands of container instances is shown to be the work of a dedicated container orchestration platform. The course will provide students both theoretical and hands-on extensive knowledge of state-of-the-art techniques to develop, operate and support modern cloud-native applications.

This course builds on the knowledge gained in earlier courses on computer architecture, operating systems and computer networking.

Contents

- Cloud concepts: pay-per-use, cloud offerings (IaaS, CaaS, PaaS, SaaS, FaaS/serverless, MaaS), public / private / hybrid / multi-cloud choices, standards
- Microservice based application design: concept of applications as loosely coupled services, strengths and drawbacks, communication methods, patterns to increase resilience in highly distributed environments, etc.
- DevOps: from source code to operational support
- Virtualisation: overview of container technologies (OS-level virtualisation, microVMs, sandboxing, trusted computing, WebAssembly, etc.)
- Container orchestration: Kubernetes architecture and functionality
- Infrastructure-as-Code, automation / configuration management and provisioning tools
- Cloud Native Computing Foundation developments: service meshes, monitoring, policy agents, artefact registries, etc.
- Distributed storage systems: file stores, object stores, block stores (not applicable for the partim version of 3 ECTS)
- Software as a service: web services, REST (not applicable for the partim version of 3 ECTS)

Initial competences

- Knowledge of computer architecture
- Knowledge of computer networking fundamentals
- Knowledge of web / service-oriented programming
- Knowledge of operating system internals
- Basic Linux (i.e. Bash) knowledge
- Programming in Python

Final competences

1 Understand and be able to use the terminology dealing with cloud-based distributed systems, OS-level virtualisation and orchestration
2 Knowledge of the drive and motivation behind microservice based applications
3 Deep understanding of OS-level virtualisation types, along with strengths and weaknesses in terms of security and performance
4 Deep understanding of container orchestration functionality, including attaching workloads to persistent storage
5 The ability to set up a Cloud-based application on a large public cloud vendor, using Infrastructure-as-Code tooling
6 The ability to set up a CI/CD pipeline transforming versioned source code into artefacts which automatically get tested and deployed on a container orchestration platform
7 The ability to make a basic design and implementation of web services running in a Cloud infrastructure (not applicable for the partim version of 3 ECTS)

Conditions for credit contract

Access to this course unit via a credit contract is determined after successful competences assessment

Conditions for exam contract

This course unit cannot be taken via an exam contract

Teaching methods

Seminar, Lecture

Study material

Type: Slides

Name: Slides on Ufora
Indicative price: Free or paid by faculty
Optional: no
Language : English
Number of Slides : 700
Oldest Usable Edition : Yearly update of slides
Available on Ufora : Yes
Online Available : Yes

Available in the Library : No
Available through Student Association : No

Type: Audiovisual Material

Name: Online videos van hoorcollege
Indicative price: Free or paid by faculty
Optional: no
Available on Ufora : Yes
Online Available : Yes

Type: Other

Name: Online labos en opdrachten + voorbereidingsmateriaal
Indicative price: Free or paid by faculty
Optional: no
Available on Ufora : Yes

Additional information: Practicums mostly on own laptops (which must be able to run x86 virtual machines) and provided Cloud infrastructure

References

- Building Microservices, 2nd edition, Sam Newman, 2021, ISBN: 978-1492034025
- Core Kubernetes, Jay Vyas and Chris Love, 2022, ISBN: 9781617297557
- Bootstrapping Microservices with Docker, Kubernetes, and Terraform, Ashley Davis, 2021, ISBN: 9781617297212

Course content-related study coaching

Interactive support and coaching through the electronic learning platform
appointments
Teachers and assistants are available for extra help before and after contact hours, and via e-mail, chat and conf calls

Assessment moments

end-of-term and continuous assessment

Examination methods in case of periodic assessment during the first examination period

Written assessment with open-ended questions

Examination methods in case of periodic assessment during the second examination period

Written assessment with open-ended questions

Examination methods in case of permanent assessment

Participation, Peer and/or self assessment, Written assessment, Assignment

Possibilities of retake in case of permanent assessment

examination during the second examination period is possible

Extra information on the examination methods

- Periodic evaluation: written examination with open and closed questions on theory, with closed-book
- During semester: graded lab sessions (written reports), graded on participation and project.

Calculation of the examination mark

- 50% on permanent evaluation, 50% on periodic exam.
- Lack of participation in permanent evaluation for no valid reason results in a zero for that part.
- In the case of group assignments, the students in a group get the same score by default. Only when there is a clear difference in contribution, the students will be given different scores.
- The student must pass (>=10/20) both parts to pass the whole course. If they fail for one part while still scoring >=10/20 on average, the final score becomes 9/20.
- Students who eschew one or more parts of the assessment can no longer obtain a pass mark for the course unit. Should the final mark be higher than 7/20, it will be reduced to the highest non-tolerable mark (i.e. 7/20).

Facilities for Working Students

Option to be freed from presence in labs after consultation with the responsible

teacher (deadline of at least a week for labs). Option for feedback by appointment during and after business hours.