

European Software
Skills Alliance.

ESSA Learning programmes

ANNEX IV

DevOps expert EQF 6

30 November 2023
Status: Final version

di
SS
e



Co-funded by the
Erasmus+ Programme
of the European Union

Copyright © 2024 by the European Software Skills Alliance. The project resources contained herein are publicly available under the [Creative Commons license 4.0 B.Y](https://creativecommons.org/licenses/by/4.0/)

ESSA Learning Programme – DevOps expert EQF 6, 2024.

Deliverable 10 – ESSA Learning Programmes & Materials – ANNEX IV

This document is a draft version and is subject to change after review coordinated by the European Education and Culture Executive Agency (EACEA).

Authors: Federica D'Armini (Adecco/Mylia)

Editors/Reviewers: Federica D'Armini (Adecco/Mylia), Sabine Boesen – Mariani (Utrecht University of Applied Sciences), Gruden Zala (University of Ljubljana), Soltes Tjasa (University of Ljubljana), Niels Sellings (Digital Europe), Wanda Saabel (Global Knowledge), Ants Slid (BCS Koolitus), Jutta Breyer (Digital Europe), Paul Aertsen (Global Knowledge)

softwareskills.eu

Legal Disclaimer

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.



Co-funded by the
Erasmus+ Programme
of the European Union

About ESSA

The European Software Skills Alliance (ESSA) is a four-year transnational project funded under the EU's Erasmus+ programme. It ensures the skills needs of the rapidly evolving Software sector can be met — today and tomorrow.

ESSA provides current and future software professionals, learning providers and organisations with software needs with the educational and training instruments they need to meet the demand for software skills in Europe.

ESSA will develop a European Software Skills Strategy and learning programmes for Europe. It will address skill mismatches and shortages by analysing the sector in depth and delivering future-proof curricula and mobility solutions; tailored to the European software sector's reality and needs.

Project partners

The ESSA consortium is led by DIGITALEUROPE. It is composed of academic and non-academic partners from the education, training, and software sectors.

View all project partners: [ESSA Partners](#) | [ESSA Associated Partners](#)

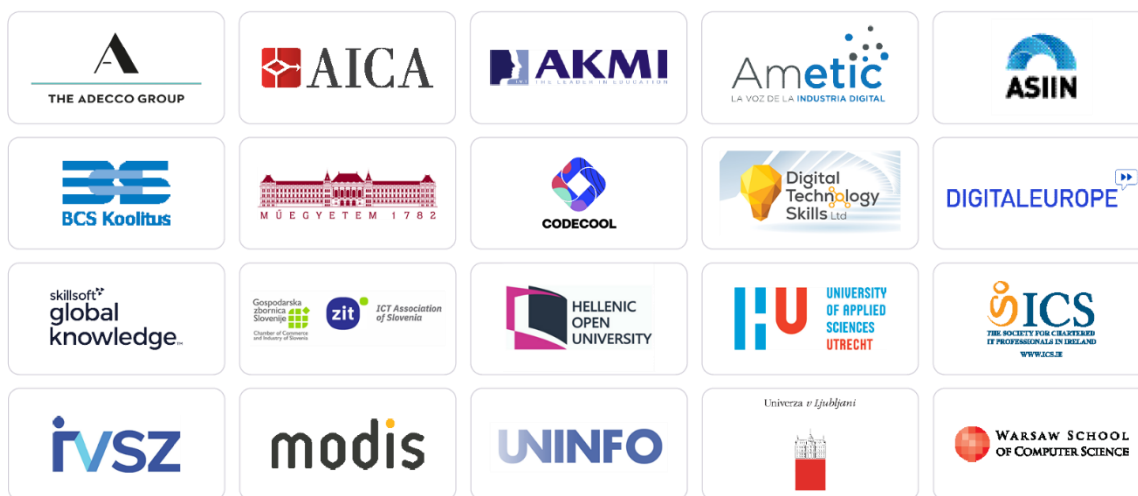


Table of Contents

1	DevOps expert EQF 6 - ESSA Learning Programme	7
1.1	IT-oriented students	7
1.1.1	PLO 1. Application Development [e-3]	7
1.1.2	Learning Resources - PLO 1. Application Development [e-3]	10
1.1.3	PLO 2. Component integration [e-3]	13
1.1.4	Learning Resources - PLO 2. Component integration [e-3]	15
1.2	People with ICT knowledge in need of reskilling to become DevOps expert..16	
1.2.1	PLO 1. Application Development [e-3]	18
1.2.2	Learning Resources - PLO 1. Application Development [e-3]	21
1.2.3	PLO 2. Component integration [e-3]	22
1.2.4	Learning Resources - PLO 2. Component integration [e-3]	24
1.2.5	PLO 3. Testing [e-3]	25
1.2.6	Learning Resources - PLO. 3 Testing [e-3]	27
1.2.7	PLO 4. Solution Deployment [e-3]	28
1.2.8	Learning Resources - PLO 4. Solution Deployment [e-3]	31
1.2.9	PLO 5. ICT Systems Engineering [e-3]	32
1.2.10	Learning Resources - PLO 5. ICT Systems Engineering [e-3]	35
1.2.11	PLO 6. Change support [e-3]	37
1.2.12	Learning Resources - PLO 6. Change support [e-3]	39
1.2.13	PLO 7. Professional related competences [EQF6]	40
1.2.14	Learning Resources - PLO 7. Profession related competence [EQF6]	43
1.2.15	PLO 8. Soft competences [EQF6]	44
1.2.16	Learning Resources - PLO 8. Soft competences [EQF6]	48
1.2.17	PLO 9. Functioning in organisations [EQF6]	50
1.2.18	Learning Resources - PLO 9. Functioning in organisation [EQF6]	53
1.2.19	EXTRA CURRICULAR PLO: New Technology [EQF6]	54
1.2.20	Learning Resources - EXTRA CURRICULAR PLO: New Technology [EQF6]	56
1.3	Students and professionals in need of upskilling/reskilling.....	57
1.3.1	PLO 1. Application Development [e-3]	58
1.3.2	Learning Resources - PLO 1. Application Development [e-3]	61
	PLO 2. Component integration [e-3]	63
1.3.3	Learning Resources - PLO 2. Component integration [e-3]	65
1.3.4	PLO 3. Testing [e-3]	69
1.3.5	Learning Resources - PLO 3. Testing [e-3]	71
1.3.6	PLO 4. Solution Deployment [e-3]	72
1.3.7	Learning Resources PLO 4. Solution Deployment [e-3]	74
1.3.8	PLO 5. ICT Systems Engineering [e-3]	78
1.3.9	Learning Resources - PLO 5. ICT Systems Engineering [e-3]	80
1.3.10	PLO 6. Change support [e-3]	82
1.3.11	Learning Resources - PLO 6. Change support [e-3]	84
1.3.12	PLO 7. Professional related competences [EQF6]	85
1.3.13	Examples of Curriculum Learning Resources - 7. PLO Profession related competence [EQF6]	87
1.3.14	PLO 8. Soft competences [EQF6]	91

1.3.15	Learning Resources - PLO 8. Soft competences [EQF6].....	93
1.3.16	PLO 9. Functionning in organisations [EQF6]	97
1.3.17	Learning Resources - PLO 9. Functionning in organisation [EQF6]	99
1.3.18	EXTRA CURRICULAR PLO: New Technology [EQF6].....	102
1.3.19	Learning Resources - EXTRA CURRICULAR PLO: New Technology [EQF6]...	104

List of abbreviations and acronyms

Abbreviation	Term
e-CF, EN 16234-1	European e-Competence Framework, European Norm 16234 - Part 1: Framework
ECTS	European Credit Transfer and Accumulation System
EQF	European Qualifications Framework
ESSA	European Software Skills Alliance
LO	Learning Outcome
PLO	Programme Learning Outcome

1 DevOps expert EQF 6 - ESSA Learning Programme

1.1 IT-oriented students

Executive summary

This Learning Programme is being designed by University of Applied Sciences Utrecht (NL). The curriculum proposed has been integrated in the current HBO Open-ICT at the Institute for ICT at Hogeschool Utrecht. This is a vocational course that trains university students to become an ICT specialist. Student learn smart and devise creative ICT solutions for business issues. The students don't get lessons nor exams, but work fully on challenging projects for real clients from the start of the studies – it is therefore a practice based learning approach. The students are coached in their learning, both on skills and competencies. The learning outcomes of the Open-ICT training program are based on the HBO-I professional tasks (elaborated by the HBO-I Foundation). This foundation is a partnership between the universities of applied sciences in the Netherlands that provide ICT education and the business community. The curriculum leverages a blended learning model, combining the presence classroom and virtual classroom. HBO Open-ICT lasts 8 semesters and the curriculum here present is addressing first and second year students.

1.1.1 PLO 1. Application Development [e-3]

1. PLO Application Development [e-3]

The learner has demonstrated capability

→ to creatively develop software applications and components, by interpreting the software design

→ to optimise the application development

Unit learning outcomes	Organises data and creates a structured dataset
	Writes code and related documentation to it, using programming languages (e.g., Java, Javascript, PHP, Python) and tools (e.g., GitHub), applying programming principles (e.g., clean coding, green coding, secure programming) and other relevant practices (e.g. continuous integration, test-driven development), principles or constraints (e.g., privacy legislation, intellectual property law)
	Efficiently creates a working software component/ application taking into account design requirements (e.g., scalability, availability) and other relevant constraints (e.g., architecture, efficiency, cost, quality, energy consumption) and applying relevant tools and techniques (e.g., object-oriented programming; IDE, CASE; editors, compilers; version control management and tools; multimedia integration tools; app development tools; reuse of proved solutions)
	Modifies an existing software component/ application, in order to optimise it (e.g., to improve scalability, maintenance, performance, security)

1.1.1.1 Duration of Study

Recommended duration: around 1 ECT

Often integrated with studies of PLOs: PLO2

1.1.1.2 Recommendations for Micro-credentials

- This PLO is currently deployed in a 4 year bachelor programme and delivered for students in the second year.

1.1.1.3 Recommendations on Didactical Approach, Delivery Methods and Training Environment

Recommended didactical approach:

- Presence Classroom
- Work placement

Additional comments

Recommended delivery methods:

- Lecture 20%
- Case study. Individual/team project 80+%

1.1.1.4 WBL and Follow-up Reinforcement

Open-ICT training program are based on the HBO-I professional tasks (elaborated by the HBO-I Foundation). This foundation is a partnership between the universities of applied sciences in the Netherlands that provide ICT education and the business community.

Open-ICT is characterized by agile project-driven education. Students therefore always work on real projects for our clients. Agile stands for short cyclical. Every two weeks the team thinks about what will be made and each student in the team looks at what he or she needs to learn for this. During the two weeks, making and learning alternate and at the end of each two weeks the work is delivered and you receive feedback on your work and your learning ability. Through this form of education, you learn new general and ICT skills every two weeks and deliver real products every two weeks. With this working method we are 100% in line with how a company works and learns later. The materials are supporting the students learning.

1.1.1.5 Important (new) approaches and technologies to consider

Open ICT is based on new approaches to education, based on intrinsic motivation. The intrinsic motivation is maximal when students are allowed to make their own choices: autonomy, when students feel included in a learning community: connectedness, and when they develop self-confidence by learning in challenging tasks: feeling competent. Every semester the student chooses a professional role they want to deepen in line with the HBO-I professional tasks. In a development team, together with the client, they determine what they will make.

Students work incorporating ways of working implemented in companies such as agile methods. The work and learning process of Open-ICT comes from the agile method of the software development industry, called SCRUM. Every two weeks, students think about what they are going

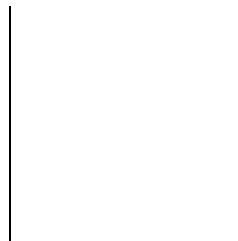
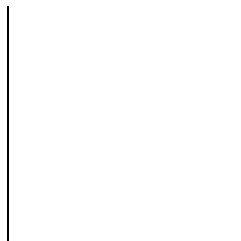
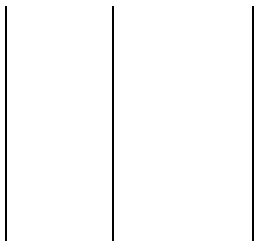
to create as a team, by user stories. They will think of the necessary tasks for their own contribution within the team and what they have to learn in order to be able to perform a certain task (learning stories). By dividing this into ‘sprints’ and properly guiding students, they can achieve learning objectives every two weeks and deliver working products. These quick results boosts confidence and motivation.

1.1.1.6 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Organises data and creates a structured dataset	Practical assessment & Portfolio	Year 1 ICT courses (foundations)
Writes code and related documentation to it, using programming languages and tools, applying programming principles and other relevant practices, principles or constraints	Practical assessment & Portfolio	Year 1 ICT courses (foundations)
Efficiently creates a working software component/ application taking into account design requirements and other relevant constraints and applying relevant tools and techniques	Practical assessment & Portfolio	Year 1 ICT courses (foundations)
Modifies an existing software component/ application, in order to optimise it	Practical assessment & Portfolio	Year 1 ICT courses (foundations)

1.1.2 Learning Resources - PLO 1. Application Development [e-3]

LEARNING UNIT	EQF	Duration	Didactical Approach	ASSESSMENT	Title of the Learning material	Delivery method of the learning material	Quick link to learning materials
<i>CISQ -1 Softwarekwaliteit versiebeheer en CI</i>	6	1 hour and 30 minutes	Live classes	Practical assessment & Portfolio	Introduction: Continuous Integration & Software Quality	Lecture and practical exercises	1. PLO Application development [e-3]
<i>CISQ - 3 Test principles and patterns</i>	6	1 hour and 30 minutes	Live classes	Practical assessment & Portfolio	Test principles and patterns	Lecture and practical exercises	1. PLO Application development [e-3]
<i>CISQ - 4 Code coverage</i>	6	1 hour and 30 minutes	Live classes	Practical assessment & Portfolio	Coverage and mutation testing	Lecture and practical exercises	1. PLO Application development [e-3]
<i>CISQ - 7 Security</i>	6	1 hour and 30 minutes	Live classes	Practical assessment & Portfolio	Security	Lecture and practical exercises	1. PLO Application development [e-3]
<i>CISQ - 8 Bonus</i>	6	1 hour and 30 minutes	Live classes	Practical assessment & Portfolio	Bonus lecture: Continuous Integration & Software Quality	Lecture and practical exercises	https://learn.softwareskills.eu/wp-content/uploads/2024/01/Lavbic-Designing-with-Swagger-and-OpenAPI-ESSA_Learning-Programmes-and-Materials.pptx https://learn.softwareskills.eu/wp-content/uploads/2024/01/Lavbic-Node-MVC-other-environments-ESSA_Learning-Programmes-and-Materials.pptx



[Angular-SPA-ESSA_Learning-Programmes-and-Materials.pptx](#)
<https://learn.softwareskills.eu/wp-content/uploads/2024/01/Data-organization-and-structures-xQuery.pptx>

1.1.3 PLO 2. Component integration [e-3]

2. PLO Component Integration [e-3]

The learner has demonstrated capability

→ to integrate efficiently a software application or component into an existing system, compliant with standards and procedures

→ to document the installation activities

Unit learning outcomes

Integrates a solution, software application or component applying relevant practices, methods, techniques and tools, compliant with appropriate standards and procedures (e.g. configuration management, version management, change control, packaging and distribution, virtualisation, containerisation)

Monitors, verifies and tests system capacity and performance, using appropriate techniques and tools

Writes an installation report/ installation documentation

1.1.3.1 Duration of Study

Recommended duration: 1 ECT

Often integrated with studies of PLOs: PLO 1

1.1.3.2 Recommendations for Micro-credentials

- This PLO is currently deployed in a 4 year bachelor programme and delivered for students in the second year.

1.1.3.3 Recommendations on Didactical Approach, Delivery Methods and Training Environment

Recommended didactical approach:

- Presence Classroom
- Work placement

Additional comments

Recommended delivery methods:

- Lecture up to 20%
- Case study. Individual/team project 80+%

Additional comments

n/a

1.1.3.4 WBL and Follow-up Reinforcement

Open-ICT training program are based on the HBO-I professional tasks (elaborated by the HBO-I Foundation). This foundation is a partnership between the universities of applied sciences in the Netherlands that provide ICT education and the business community.

Open-ICT is characterized by agile project-driven education. Students therefore always work on real projects for our clients. Agile stands for short cyclical. Every two weeks the team thinks about what will be made and each student in the team looks at what he or she needs to learn for this. During the two weeks, making and learning alternate and at the end of each two weeks the work is delivered and you receive feedback on your work and your learning ability. Through this form of education, you learn new general and ICT skills every two weeks and deliver real products every two weeks. With this working method we are 100% in line with how a company works and learns later. The materials are supporting the students learning.

1.1.3.5 Important (new) approaches and technologies to consider

Open ICT is based on new approaches to education, based on intrinsic motivation. The intrinsic motivation is maximal when students are allowed to make their own choices: autonomy, when students feel included in a learning community: connectedness, and when they develop self-confidence by learning in challenging tasks: feeling competent. Every semester the student chooses a professional role they want to deepen in line with the HBO-I professional tasks. In a development team, together with the client, they determine what they will make.

Students work incorporating ways of working implemented in companies such as agile methods. The work and learning process of Open-ICT comes from the agile method of the software development industry, called SCRUM. Every two weeks, students think about what they are going to create as a team, by user stories. They will think of the necessary tasks for their own contribution within the team and what they have to learn in order to be able to perform a certain task (learning stories). By dividing this into ‘sprints’ and properly guiding students, they can achieve learning objectives every two weeks and deliver working products. These quick results boosts confidence and motivation.

1.1.3.6 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Integrates a solution, software application or component applying relevant practices, methods, techniques and tools, compliant with appropriate standards and procedures	Practical assessment & Portfolio	Year 1 ICT courses (foundations)
Monitors, verifies and tests system capacity and performance, using appropriate techniques and tools	Practical assessment & Portfolio	Year 1 ICT courses (foundations)
Writes an installation report/ installation documentation	Practical assessment & Portfolio	Year 1 ICT courses (foundations)

1.1.4 Learning Resources - PLO 2. Component integration [e-3]

LEARNING UNIT	EQF	Duration	Didactical Approach	ASSESSMENT	Title of the Learning material	Delivery method of the learning material	Quick link to learning materials
<i>CISQ - 2 Testautomatisering</i>	6	1 hour and 30 minutes	Live classes	Practical assessment & Portfolio	Test automation	Lecture and practical exercises	2. PLO Component integration [e-3]
<i>CISQ - 3 Test principles and patterns</i>	6	1 hour and 30 minutes	Live classes	Practical assessment & Portfolio	Test principles and patterns	Lecture and practical exercises	2. PLO Component integration [e-3]
<i>CISQ - 5 Testen met collaborators</i>	6	1 hour and 30 minutes	Live classes	Practical assessment & Portfolio	Testing with collaborators	Lecture and practical exercises	2. PLO Component integration [e-3]
<i>CISQ - 6 Structuur en testaanpak</i>	6	1 hour and 30 minutes	Live classes	Practical assessment & Portfolio	Structure and test approach	Lecture and practical exercises	2. PLO Component integration [e-3]

1.2 People with ICT knowledge in need of reskilling to become DevOps expert

Executive summary

This Learning Programme is being designed by Global Knowledge France (FR). The main objective of the programme is to reskill people to become a DevOps expert at level EQF6. The programme target group are people with some ICT knowledge that want to reskill themselves quickly. The programme is well suited to be offered by VET and training institutes, both in more traditional settings, and also, for example, in more time- and location-independent settings, for example, in the case of distance learning institutes. As the programme focuses on a rapid reskilling and short route to the labour market, cooperation can be sought with labour market intermediaries.

The programme has a compact design and runs over a 3-month period, during which participants are actively engaged full-time. The advice is to keep the group of participants very small to ensure intensive guidance. The programme is made up of 8 learning units for a total duration of 406 hours, which can be delivered face-to-face, remotely or in a mix of the 2 modalities. By offering the programme in these different ways, flexibility and accessibility are maximised. The programme itself is a mix of theory and practical assignments, in which the participants have to work together to achieve results. The programme concludes with an overall team project in which all the knowledge and skills learned must be applied.

The programme starts by explaining the role of software development and IT in projects and organisations. It then follows a logical structure with DevOps Foundation, basic programming skills such as SQL, OO, and Python, followed by introductions of deployment tools such as Docker and Kubernetes and code-based infrastructure tools (Ansible, Puppet, Vagrant, Terraform). Testing, DevSecOps and change management are also covered. Also, more profession-related competences are addressed with project management, team working and time management. The programme concludes with an overarching software implementation assignment requiring teamwork.

In order to maximise the flow to business and employment and make the programme as attractive as possible for jobseekers, it is also highly recommended to work closely with companies or other institutions that can mediate in this process.

The programme that is offered by Global Knowledge France is part of employability and reskilling programs in France and is financed by Pôle emploi, the CPF (Compte Professionnel de Formation) and public funding bodies. Pôle emploi is the leading player in the French employment market, operating as an intermediate between companies and jobseekers. Global Knowledge France works closely together with Pôle emploi, assuring that the job guarantee of its learners in selected programmes is almost 100%. Besides this, offering programmes in-company, as a private class for a specific organisation adds to the attractiveness of the programme as well.

Learning units	PLO		Duration in hours (ECTS)	% practice
IT in the organisation	1,9	The role of development in IT projects	16 (0,64 ECTS)	Theoretical
Work in project situation	9	Introduction to project management	8 (0,32 ECTS)	Theoretical
		Introduction to Agile	16 (0,64 ECTS)	40%
Team working		Integration and teamwork	8 (0,32 ECTS)	80%
	8	Assertiveness and place in a team	8 (0,32 ECTS)	80%
		Communication	8 (0,32 ECTS)	80%
		Prevent and manage difficult situations	8 (0,32 ECTS)	80%
Get trained and informed		Managing time and priorities	8 (0,32 ECTS)	80%
	7,8	Developing creativity	8 (0,32 ECTS)	80%
		Technology watch	8 (0,32 ECTS)	Theoretical
Work in DevOps culture		DevOps Foundation	16 (0,64 ECTS)	Theoretical
	4,5,6	ITIL introduction	8 (0,32 ECTS)	Theoretical
		Change management	16 (0,64 ECTS)	80%
DevOps Fundamentals to develop and integrate a solution in DevOps mode [OO, SQL, PYTHON, Linux, Shell]		SQL basics	24 (0,96 ECTS)	40%
		Object design	16 (0,64 ECTS)	60%
	1,3,5	Unix, Linux: the essential basics and Shell programming	40 (1,6 ECTS)	40%
		Python programming	24 (0,96 ECTS)	50%
		Software and hardware architecture	24 (0,96 ECTS)	Theoretical
		Introduction to testing	8 (0,32 ECTS)	Theoretical
		DevSecOps	16 (0,64 ECTS)	Theoretical
Continuously develop and deploy [Testing, DevSecOps, SonarQube, Docker,		Implementing the software factory	40 (1,6 ECTS)	50%
	2,4,5	Production deployment tools: Docker and Kubernetes	40 (1,6 ECTS)	70%

Kubernetes, deployment tools]		Tools for continuous deployment and code-based infrastructure: Ansible, Puppet, Vagrant, Terraform	40 (1,6 ECTS)	50%
Team project: implementing a software factory	1,2,3,4,5,7,8,9	Project: Implementation of a software factory	40 (1,6 ECTS)	100%

Curriculum DevOps Expert EQF6 Global Knowledge France

1.2.1 PLO 1. Application Development [e-3]

1. PLO Application Development [e-3]

The learner has demonstrated capability

→ to creatively develop software applications and components, by interpreting the software design

→ to optimise the application development

Unit learning outcomes	Organises data and creates a structured dataset
	Writes code and related documentation to it, using programming languages (e.g., Java, Javascript, PHP, Python) and tools (e.g., GitHub), applying programming principles (e.g., clean coding, green coding, secure programming) and other relevant practices (e.g. continuous integration, test-driven development), principles or constraints (e.g., privacy legislation, intellectual property law)
	Efficiently creates a working software component/ application taking into account design requirements (e.g., scalability, availability) and other relevant constraints (e.g., architecture, efficiency, cost, quality, energy consumption) and applying relevant tools and techniques (e.g., object-oriented programming; IDE, CASE; editors, compilers; version control management and tools; multimedia integration tools; app development tools; reuse of proved solutions)
	Modifies an existing software component/ application, in order to optimise it (e.g., to improve scalability, maintenance, performance, security)

1.2.1.1 Duration of Study

Recommended duration: starting from 5 ECTS as an absolute minimum. If bigger and more complex practical assignments are used such as groupwork and a team project, then 7 ECTS is a minimum.

Often integrated with studies of PLOs: PLO 2. Component Integration, PLO 3. Testing, PLO 4. Solution Deployment and PLO 5. ICT Systems Engineering and if there are practical assignments, groupwork and a team project involved also PLO's: 7. Profession related competences, 8. Soft competences and 9. Functioning in organisations may be involved.

1.2.1.2 Recommendations for Micro-credentials

This PLO and its subsequent parts can be offered as a micro-credential as part of a modular (re)skilling programme for learners with no prior knowledge of application development. This PLO is also recommended as an independent stand-alone micro-credential for skilling and reskilling (ICT) professionals that are interested in learning the basics of application development. Parts of this PLO are also good candidates for micro-credentials, such as:

- Object design
- SQL basics
- Python programming
- Software & hardware architecture
- Testing introduction
- DevSecOps

1.2.1.3 Recommendations on Didactical Approach, Delivery Methods and Training Environment

Recommended didactical approach:

- | | |
|---------------------|-------------------------------------|
| • F2F classroom | <input checked="" type="checkbox"/> |
| • Virtual classroom | <input checked="" type="checkbox"/> |
| • Blended | <input checked="" type="checkbox"/> |
| • e-Learning | <input checked="" type="checkbox"/> |
| • In-company | <input checked="" type="checkbox"/> |

Additional comments

To maximise accessibility and flexibility it is recommended that different didactical approaches are used as much as possible, so that the individual learner can decide what suits best. Besides this, offering in-company courses and training supports accessibility and flexibility.

Recommended delivery methods:

- | | |
|--|-------------------------------------|
| • Lectures F2F | <input checked="" type="checkbox"/> |
| • Lectures virtual | <input checked="" type="checkbox"/> |
| • Lectures blended | <input checked="" type="checkbox"/> |
| • Virtual instructor-led training (VILT) | <input checked="" type="checkbox"/> |
| • Practical exercises | <input checked="" type="checkbox"/> |
| • Group/ teamwork | <input checked="" type="checkbox"/> |
| • Team project | <input checked="" type="checkbox"/> |

Additional comments

Lectures, e-learning and virtual instructor-led training are recommended for learning the basic principles, terminology, and methods of application development. These should be reinforced through practical tasks, individual and group assignments, and if possible, a team project. The ratio between on the one hand lectures/ VILT and on the other hand practical work done by the learners should be appr. 60% - 40% respectively.

1.2.1.4 WBL and Follow-up Reinforcement

After learning the basic principles, terminology, and models of application development, the programme should focus on analysing and simulating real work-life-like tasks as, for example:

- Practical exercises, based on real life situations, e.g., case studies

1.2.1.5 Important (new) approaches and technologies to consider

n/a

1.2.1.6 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Organises data and creates a structured dataset	Exam	n/a
Writes code and related documentation to it, using programming languages, applying programming principles and other relevant practices, principles, or constraints	Exam	n/a
Efficiently creates a working software component/ application taking into account design requirements and other relevant constraints and applying relevant tools and techniques	Practical assignment	n/a
Modifies an existing software component/ application, in order to optimize it	Practical assignment	n/a

1.2.2 Learning Resources - PLO 1. Application Development [e-3]

LEARNING UNIT	EQF	Duration in hrs (ECTS)	Didactical Approach	ASSESSMENT (formative+summative)	Title of the Learning material	Delivery method of the learning material	Quick link to learning materials
IT in the organisation	4	16 (0,64 ECTS)	F2F, virtual, blended	Exam	Course materials	Virtual instructor-led training (VILT), F2F lectures	-
DevOps Fundamentals: -Object design -SQL basics -Python Programming -Software & hardware architecture -Introduction to testing -DevSecOps	6	152 (6,1 ECTS)	F2F, virtual, blended, e-learning/ videos	Practical exercises, Exam	Course materials, workbook; "Object design and UML", "Object design exercise - Mini Bank", "Object design exercise solution - Mini Bank"	Training lectures, Virtual instructor-led training (VILT), Practical exercises	https://learn.softwareskills.eu/wp-content/uploads/2023/11/Object-design-and-UML-presentation-Dev.pdf https://learn.softwareskills.eu/wp-content/uploads/2023/11/Object-design-exercise-solution-Mini-Bank-Dev.pdf https://learn.softwareskills.eu/wp-content/uploads/2023/11/Object-design-exercise-Mini-Bank-Dev.pdf
Team project: Implementing of a software factory	6	40 (1,6 ECTS)	F2F, virtual, blended	Practical assignment, presentation	Workbook	Training lectures, Virtual instructor-led training (VILT), Team project, Solving a problem together	-

1.2.3 PLO 2. Component integration [e-3]

2. PLO Component Integration [e-3]

The learner has demonstrated capability

→ to integrate efficiently a software application or component into an existing system, compliant with standards and procedures

→ to document the installation activities

Unit learning outcomes

Integrates a solution, software application or component applying relevant practices, methods, techniques and tools, compliant with appropriate standards and procedures (e.g. configuration management, version management, change control, packaging and distribution, virtualisation, containerisation)

Monitors, verifies and tests system capacity and performance, using appropriate techniques and tools

Writes an installation report/ installation documentation

1.2.3.1 Duration of Study

Recommended duration: starting from 5 ECTS as an absolute minimum. If bigger and more complex practical assignments are used such as groupwork and a team project, then 7 ECTS is a minimum.

Often integrated with studies of PLOs: PLO 1. Application Development, PLO 3. Testing, PLO 4. Solution Deployment and PLO 5. ICT Systems Engineering and if there are practical assignments, groupwork and a team project involved also PLO's: 7. Profession related competences, 8. Soft competences and 9. Functioning in organisations may be involved.

1.2.3.2 Recommendations for Micro-credentials

This PLO and its subsequent parts can be offered as a micro-credential as part of a modular (re)skilling programme for learners with no prior knowledge of component integration or deployment. This PLO is also recommended as an independent stand-alone micro-credential for skilling and reskilling (ICT) professionals that are interested in learning the basics and tooling of CI/CD. Parts of this PLO are also good candidates for micro-credentials, such as:

- Production deployment tools (Docker, Kubernetes)
- Tools for continuous deployment and code-based infrastructure (Ansible, Puppet, Vagrant, Terraform)

1.2.3.3 Recommendations on Didactical Approach, Delivery Methods and Training Environment

Recommended didactical approach:

- F2F classroom
- Virtual classroom
- Blended
- e-Learning

- In-company

Additional comments

To maximise accessibility and flexibility it is recommended that different didactical approaches are used as much as possible, so that the individual learner can decide what suits best. Besides this, offering in-company courses and training supports accessibility and flexibility.

Recommended delivery methods:

- Lectures F2F
- Lectures virtual
- Lectures blended
- Virtual instructor-led training (VILT)
- Practical exercises
- Group/ teamwork
- Team project

Additional comments

Lectures, e-learning and virtual instructor-led training are recommended for learning the basic principles, terminology, and methods of application development. These should be reinforced through practical tasks, individual and group assignments, and if possible, a team project. The ratio between on the one hand lectures/ VILT and on the other hand practical work done by the learners should be appr. 60% - 40% respectively.

1.2.3.4 WBL and Follow-up Reinforcement

After learning the basic principles, methods and tools of component integration, the programme should focus on analysing and simulating real work-life-like tasks as, for example:

- Practical exercises, based on real life situations, e.g., case studies
- Working together in a team to integrate a solution

1.2.3.5 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Integrates a solution, software application or component applying relevant practices, methods, techniques and tools, compliant with appropriate standards and procedures	Practical assignment	n/a
Monitors, verifies and tests system capacity and performance, using appropriate techniques and tools	Practical assignment	n/a
Writes an installation report/ installation documentation	Practical assignment	n/a

1.2.4 Learning Resources - PLO 2. Component integration [e-3]

LEARNING UNIT	EQF	Duration in hrs (ECTS)	Didactical Approach	ASSESSMENT (formative+summative)	Title of the Learning material	Delivery method of the learning material	Quick link to learning materials
Continuously develop and deploy: - Production deployment tools with Docker and Kubernetes - Tools for continuous deployment and code-based infrastructure (Ansible, Puppet, Vagrant, Terraform) - Software factory implementation	6	120 (4,8 ECTS)	F2F, virtual, blended, e-learning/ videos	Practical exercises, Exam	Course materials, workbook	Training lectures, Virtual instructor-led training (VILT), Practical exercises	-
Team project: Implementing of a software factory	6	40 (1,6 ECTS)	F2F, virtual, blended	Practical assignment, presentation	Workbook	Training lectures, Virtual instructor-led training (VILT), Team project, Solving a problem together	-

1.2.5 PLO 3. Testing [e-3]

3. PLO Testing [e-3]

The learner has demonstrated capability

→ to construct and execute tests for solutions, software applications or components

→ to document test outcomes

Unit learning outcomes	Writes (parts of) test related documentation (e.g. test plan, test strategy/approach, test case, test script, test scenario, test conditions)
	Constructs tests, by selecting appropriate test methods, techniques, and tools (e.g. test automation tools)
	Defines and configures a test environment
	Executes test cases and performs test activities related to different sorts of tests
	Records and interprets test outcomes and writes test result documentation/ test report

1.2.5.1 Duration of Study

Recommended duration: starting from 2 ECTS as an absolute minimum. If bigger and more complex practical assignments are used such as groupwork and a team project, then 5 ECTS is a minimum.

Often integrated with studies of PLOs: PLO 1. Application development, PLO 3. Component Integration, PLO 4. Solution Deployment and if there are practical assignments, groupwork and a team project involved also PLO's: 6. Profession related competences, 7. Soft competences and 8. Functioning in organisations may be involved.

1.2.5.2 Recommendations for Micro-credentials

This PLO can be offered as a micro-credential as part of a modular (re)skilling programme for learners with no prior knowledge of software testing. This PLO is also recommended as an independent stand-alone micro-credential for skilling and reskilling (ICT) professionals that are interested in learning the basics of testing.

1.2.5.3 Recommendations on Didactical Approach, Delivery Methods and Training Environment

Recommended didactical approach:

- F2F classroom
- Virtual classroom
- Blended
- e-Learning
- In-company

Additional comments

To maximise accessibility and flexibility it is recommended that different didactical approaches are used as much as possible, so that the individual learner can decide what suits best. Besides this, offering in-company courses and training supports accessibility and flexibility.

Recommended delivery methods:

- Lectures F2F ☒
- Lectures virtual ☒
- Lectures blended ☒
- Virtual instructor-led training (VILT) ☒
- Practical exercises ☒
- Group/ teamwork ☒
- Team project ☒

Additional comments

Lectures, e-learning and virtual instructor-led training are recommended for learning the basic principles, terminology, and methods of application development. These should be reinforced through practical tasks, individual and group assignments, and if possible, a team project. The ratio between on the one hand lectures/ VILT and on the other hand practical work done by the learners should be appr. 60% - 40% respectively.

1.2.5.4 WBL and Follow-up Reinforcement

After learning the basic principles, terminology, and models of application development, the programme should focus on analysing and simulating real work-life-like tasks as, for example:

- Practical exercises, based on real life situations, e.g., case studies

1.2.5.5 Important (new) approaches and technologies to consider

n/a

1.2.5.6 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Writes (parts of) test related documentation	Practical assignment	n/a
Constructs tests, by selecting appropriate test methods, techniques, and tools	Practical assignment	n/a
Defines and configures a test environment	Practical assignment	n/a
Executes test cases and performs test activities related to different sorts of tests	Practical assignment	n/a
Records and interprets test outcomes and writes test result documentation/ test report	Practical assignment	n/a

1.2.6 Learning Resources - PLO. 3 Testing [e-3]

LEARNING UNIT	EQF	Duration in hrs (ECTS)	Didactical Approach	ASSESSMENT (formative+summative)	Title of the Learning material	Delivery method of the learning material	Quick link to learning materials
DevOps Fundamentals: <u>-Introduction to testing</u>	5	8 (0,32 ECTS)	F2F, virtual, blended	Exam	Course materials	Training lectures, Virtual instructor-led training (VILT)	-
Team project: Implementing of a software factory	6	40 (1,6 ECTS)	F2F, virtual, blended	Practical assignment, presentation	Workbook	Training lectures, Virtual instructor-led training (VILT), Team project, Solving a problem together	-

1.2.7 PLO 4. Solution Deployment [e-3]

4. PLO Solution Deployment [e-3]

The learner has demonstrated capability

→ to implement solutions and services

→ to complete release documentation

Unit learning outcomes	Writes a release plan (e.g., including solution verification and validation, documentation, supporting activities; deployment workflow and product roll-out activities)
	Writes a plan for data management/ migration
	Executes (parts of) a solution/ software release, applying appropriate methods, techniques, and tools (e.g., CI/CD tools; tools related to automated software release, software packaging and distribution)
	Writes (parts of) release documentation

1.2.7.1 Duration of Study

Recommended duration: starting from 5 ECTS as an absolute minimum. If bigger and more complex practical assignments are used such as groupwork and a team project, then 7 ECTS is a minimum.

Often integrated with studies of PLOs: PLO 1. Application Development, PLO 2. Component Integration, PLO 3. Testing, and PLO 5. ICT Systems Engineering and if there are practical assignments, groupwork and a team project involved also PLO's: 7. Profession related competences, 8. Soft competences and 9. Functioning in organisations may be involved.

1.2.7.2 Recommendations for Micro-credentials

This PLO and its subsequent parts can be offered as a micro-credential as part of a modular (re)skilling programme for learners with no prior knowledge of component integration or deployment. This PLO is also recommended as an independent stand-alone micro-credential for skilling and reskilling (ICT) professionals that are interested in learning the basics and tooling of CI/CD. Parts of this PLO are also good candidates for micro-credentials, such as:

- DevOps Foundation
- Production deployment tools (Docker, Kubernetes)
- Tools for continuous deployment and code-based infrastructure (Ansible, Puppet, Vagrant, Terraform)

1.2.7.3 Recommendations on Didactical Approach, Delivery Methods and Training Environment

Recommended didactical approach:

- F2F classroom
- Virtual classroom
- Blended

- e-Learning
- In-company

Additional comments

To maximise accessibility and flexibility it is recommended that different didactical approaches are used as much as possible, so that the individual learner can decide what suits best. Besides this, offering in-company courses and training supports accessibility and flexibility.

Recommended delivery methods:

- Lectures F2F
- Lectures virtual
- Lectures blended
- Virtual instructor-led training (VILT)
- Practical exercises
- Group/ teamwork
- Team project

Additional comments

Lectures, e-learning and virtual instructor-led training are recommended for learning the basic principles, terminology, and methods of application development. These should be reinforced through practical tasks, individual and group assignments, and if possible, a team project. The ratio between on the one hand lectures/ VILT and on the other hand practical work done by the learners should be appr. 60% - 40% respectively.

1.2.7.4 WBL and Follow-up Reinforcement

After learning the basic principles, methods and tools of solution deployment, the programme should focus on analysing and simulating real work-life-like tasks as, for example:

- Practical exercises, based on real life situations, e.g., case studies
- Working together in a team to deploy a solution

1.2.7.5 Important (new) approaches and technologies to consider

n/a

1.2.7.6 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Writes a release plan	Practical assignment	n/a
Writes a plan for data management/ migration	Practical assignment	n/a
Executes (parts of) a solution/ software release, applying appropriate methods, techniques, and tools	Practical assignment	n/a
Writes (parts of) release documentation	Practical assignment	n/a

1.2.8 Learning Resources - PLO 4. Solution Deployment [e-3]

LEARNING UNIT	EQF	Duration in hrs (ECTS)	Didactical Approach	ASSESSMENT (formative+summative)	Title of the Learning material	Delivery method of the learning material	Quick link to learning materials
Work in DevOps culture: <u>- DevOps Foundation</u>	5	16 (0,64 ECTS)	F2F, virtual, blended	Exam	Course materials	Training lectures, Virtual instructor-led training (VILT)	-
Continuously develop and deploy: <u>- Production deployment tools with Docker and Kubernetes</u> <u>- Tools for continuous deployment and code-based infrastructure (Ansible, Puppet, Vagrant, Terraform)</u> <u>- Software factory implementation</u>	6	120 (4,8 ECTS)	F2F, virtual, blended, e-learning/ videos	Practical exercises, Exam	Course materials, workbook	Training lectures, Virtual instructor-led training (VILT), Practical exercises	-
Team project: Implementing of a software factory	6	40 (1,6 ECTS)	F2F, virtual, blended	Practical assignment, presentation	Workbook	Training lectures, Virtual instructor-led training (VILT), Team project, Solving a problem together	-

1.2.9 PLO 5. ICT Systems Engineering [e-3]

5. PLO ICT Systems Engineering [e-3]

The learner has demonstrated capability

→ to create a system infrastructure that meets requirements

→ to ensure interoperability of system components

Unit learning outcomes	Describes software and hardware components, tools and architectures, including network components, topologies, protocols and interconnections
	Analyses existing system infrastructures, applying appropriate methods, techniques, and tools (e.g. related to interoperability of components, security, energy consumption)
	Proposes improvements to an existing system infrastructure, to better meet requirements (e.g., cloud solutions, Infrastructure as Code, Containers as a Service, virtual machines, load balancers)
	Designs (parts of) a DevOps pipeline, by formulating a set of practices and tools that the development and operations teams may implement to build, test, and deploy software

1.2.9.1 Duration of Study

Recommended duration: starting from 5 ECTS as an absolute minimum. If bigger and more complex practical assignments are used such as groupwork and a team project, then 7 ECTS is a minimum.

Often integrated with studies of PLOs: PLO 2. Component Integration, PLO 3. Testing, and PLO 4. Solution Deployment and if there are practical assignments, groupwork and a team project involved also PLO's: 7. Profession related competences, 8. Soft competences and 9. Functioning in organisations may be involved.

1.2.9.2 Recommendations for Micro-credentials

This PLO and its subsequent parts can be offered as a micro-credential as part of a modular (re)skilling programme for learners with no prior knowledge of systems engineering.

1.2.9.3 Recommendations on Didactical Approach, Delivery Methods and Training Environment

Recommended didactical approach:

- F2F classroom
- Virtual classroom
- Blended
- e-Learning
- In-company

Additional comments

To maximise accessibility and flexibility it is recommended that different didactical approaches are used as much as possible, so that the individual learner can decide what suits best. Besides this, offering in-company courses and training supports accessibility and flexibility.

Recommended delivery methods:

- Lectures F2F ☒
- Lectures virtual ☒
- Lectures blended ☒
- Virtual instructor-led training (VILT) ☒
- Practical exercises ☒
- Group/ teamwork ☒
- Team project ☒

Additional comments

Lectures, e-learning and virtual instructor-led training are recommended for learning the basic principles, terminology, and methods of application development. These should be reinforced through practical tasks, individual and group assignments, and if possible, a team project. The ratio between on the one hand lectures/ VILT and on the other hand practical work done by the learners should be appr. 60% - 40% respectively.

1.2.9.4 WBL and Follow-up Reinforcement

After learning the basic principles, methods and tools of component integration, the programme should focus on analysing and simulating real work-life-like tasks as, for example:

- Practical exercises, based on real life situations, e.g., case studies
- Working together in a team to deploy a solution

1.2.9.5 Important (new) approaches and technologies to consider

n/a

1.2.9.6 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Describes software and hardware components, tools and architectures, including network components, topologies, protocols and interconnections	Exam	n/a
Analyses existing system infrastructures, applying appropriate methods, techniques, and tools	Practical assignment	n/a

Proposes improvements to an existing system infrastructure, to better meet requirements	Practical assignment	n/a
Designs (parts of) a DevOps pipeline, by formulating a set of practices and tools that the development and operations teams may implement to build, test, and deploy software	Practical assignment	n/a

1.2.10 Learning Resources - PLO 5. ICT Systems Engineering [e-3]

LEARNING UNIT	EQF	Duration in hrs (ECTS)	Didactical Approach	ASSESSMENT (formative+summative)	Title of the Learning material	Delivery method of the learning material	Quick link to learning materials
Work in DevOps culture: - DevOps Foundation - ITIL introduction - Change management	5	40 (1,6 ECTS)	F2F, virtual, blended	Exam	Course materials, workbook	Training lectures, Virtual instructor-led training (VILT)	-
DevOps Fundamentals: - Software and hardware architecture - Unix, Linux: the essential basics and Shell programming	5	64 (2,6 ECTS)	F2F, virtual, blended	Exam	Course materials	Training lectures, Virtual instructor-led training (VILT)	-
Continuously develop and deploy: - Tools for continuous deployment and code-based infrastructure	6	80 (3,2 ECTS)	F2F, virtual, blended, e-learning/ videos	Practical exercises, Exam	Course materials, workbook	Training lectures, Virtual instructor-led training (VILT), Practical exercises	-

<u>(Ansible, Puppet, Vagrant, Terraform)</u> <u>- Software factory implementation</u>							
Team project: Implementing of a software factory	6	40 (1,6 ECTS)	F2F, virtual, blended	Practical assignment, presentation	Workbook	Training lectures, Virtual instructor-led training (VILT), Team project, Solving a problem together	-

1.2.11 PLO 6. Change support [e-3]

6. Change support [e-3]

The learner has demonstrated capability

→ to oversee and control system changes

→ to take into account procedures, requirements and restrictions

Unit learning outcomes

Describes, distinguishes, and applies change management methods, tools and techniques

Proposes and applies appropriate and efficient practices, measures and procedures to handle change and to reduce the impact of change (e.g., based on ITIL Change management, DevOps)

Writes a change request/ change proposal (for addition, modification or removal of a service or software application/component) (e.g., describing benefit, risk, and impact of change, taking into account requirements and restrictions such as information security regulations, budget, SLA, conditions for system integrity)

1.2.11.1 Duration of Study

Recommended duration: starting from 2 ECTS as an absolute minimum.

Often integrated with studies of PLOs: PLO 4. Solution Deployment and if there are practical assignments, groupwork and a team project involved also PLO's: 6. Profession related competences, 7. Soft competences and 8. Functioning in organisations may be involved.

1.2.11.2 Recommendations for Micro-credentials

This PLO can be offered as a micro-credential as part of a modular (re)skilling programme for learners with no prior knowledge of change support in ICT or change management in organisations.

1.2.11.3 Recommendations on Didactical Approach, Delivery Methods and Training Environment

Recommended didactical approach:

- F2F classroom
- Virtual classroom
- Blended
- e-Learning
- In-company

Additional comments

To maximise accessibility and flexibility it is recommended that different didactical approaches are used as much as possible, so that the individual learner can decide what suits best. Besides this, offering in-company courses and training supports accessibility and flexibility.

Recommended delivery methods:

- Lectures F2F ☒
- Lectures virtual ☒
- Lectures blended ☒
- Virtual instructor-led training (VILT) ☒
- Practical exercises ☒
- Group/ teamwork ☒
- Team project ☒

Additional comments

Lectures, e-learning and virtual instructor-led training are recommended for learning the basic principles, terminology, and methods of change support/ management. These should be reinforced through practical tasks, individual and group assignments. The ratio between on the one hand lectures/ VILT and on the other hand practical work done by the learners should be appr. 80% - 20% respectively.

1.2.11.4 WBL and Follow-up Reinforcement

After learning the basic principles, terminology, methods and techniques of testing, the programme should focus on analysing and simulating real work-life-like tasks as, for example:

- Practical exercises, based on real life situations, e.g., case studies

1.2.11.5 Important (new) approaches and technologies to consider

n/a

1.2.11.6 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Describes, distinguishes, and applies change management methods, tools and techniques	Practical exercises	n/a
Proposes and applies appropriate and efficient practices, measures and procedures to handle change and to reduce the impact of change	Practical exercises	n/a
Writes a change request/ change proposal	Practical exercises	n/a

1.2.12 Learning Resources - PLO 6. Change support [e-3]

LEARNING UNIT	EQF	Duration in hrs (ECTS)	Didactical Approach	ASSESSMENT (formative+summative)	Title of the Learning material	Delivery method of the learning material	Quick link to learning materials
Prevent and manage difficult situations	5	8 (0,32 ECTS)	F2F, virtual	Practical exercises	Course materials	Training lectures, Virtual instructor-led training (VILT), work in groups	-
Change management	6	16 (0,64 ECTS)	F2F, virtual	Practical exercises	Course materials, workbook	Training lectures, Virtual instructor-led training (VILT), work in groups	-

1.2.13 PLO 7. Professional related competences [EQF6]

7. PLO Profession related competences [EQF6]

The learner has demonstrated capability

→ to apply profession related skills

Unit learning outcomes	Masters common ICT knowledge
	Explains the principles, related concepts, advantages, disadvantages, limitations and possible societal, environmental, and ethical issues related to the application of a new technology. Applies and reports on basic methods, techniques and tools related to a new technology.
	Applies, evaluates, reports and provides advice on security standards, regulations, measures, methods, tools, and techniques
	Applies, evaluates and provides advice on appropriate methods, tools and techniques related to software lifecycle processes
	Applies, evaluates, reports and provides advice on sustainability standards, regulations, measures, and methods.
	Is aware of ethical considerations and issues and applies these in professional contexts and activities. Forms and communicates an opinion based on considerations of relevant social, professional, scientific and ethical aspects

1.2.13.1 Duration of Study

Recommended duration: starting from 5 ECTS as an absolute minimum. If bigger and more complex practical assignments are used such as groupwork and a team project, then 7 ECTS is a minimum.

Often integrated with studies of PLOs: This PLO addresses cross-cutting topics, that relate to many technical aspects in the field of ICT and DevOps. It is often integrated with PLO 1. Application Development, PLO 2. Component Integration, PLO 3. Testing, PLO 4. Solution Deployment, PLO 5. Systems Engineering and if there are practical assignments, groupwork and a team project involved also PLO's: 7. Soft competences and 8. Functioning in organisations may be involved.

1.2.13.2 Recommendations for Micro-credentials

This PLO and especially its subsequent parts can be offered as a micro-credential as part of a modular (re)skilling programme for learners with no prior knowledge of software development. This PLO is also recommended as an independent stand-alone micro-credential for skilling and reskilling (ICT) professionals that are interested in learning the basics of these topics. Parts of this PLO are also good candidates for micro-credentials, such as:

- ICT in organisations
- New technology watch

1.2.13.3 Recommendations on Didactical Approach, Delivery Methods and Training Environment

Recommended didactical approach:

- F2F classroom
- Virtual classroom
- Blended
- e-Learning
- In-company

Additional comments

To maximise accessibility and flexibility it is recommended that different didactical approaches are used as much as possible, so that the individual learner can decide what suits best. Besides this, offering in-company courses and training supports accessibility and flexibility.

Recommended delivery methods:

- Lectures F2F
- Lectures virtual
- Lectures blended
- Virtual instructor-led training (VILT)
- Practical exercises
- Group/ teamwork
- Team project

Additional comments

Lectures, e-learning and virtual instructor-led training are recommended for learning the basic principles, terminology, and methods related to professional competences. These should be reinforced through practical tasks, individual and group assignments, and if possible, a team project. The ratio between on the one hand lectures/VILT and on the other hand practical work done by the learners should be appr. 60% - 40% respectively.

1.2.13.4 WBL and Follow-up Reinforcement

After learning the basic principles, terminology, and models of application development, the programme should focus on analysing and simulating real work-life-like tasks as, for example:

- Practical exercises, based on real life situations, e.g., case studies
- Working together in a team to solve specific problems

1.2.13.5 Important (new) approaches and technologies to consider

n/a

1.2.13.6 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Masters common ICT knowledge	Exam	n/a
Explains the principles, related concepts, advantages, disadvantages, limitations and possible societal, environmental, and ethical issues related to the application of a new technology. Applies and reports on basic methods, techniques and tools related to a new technology.	Exam	n/a
Applies, evaluates, reports and provides advice on security standards, regulations, measures, methods, tools, and techniques	Exam	n/a
Applies, evaluates and provides advice on appropriate methods, tools and techniques related to software lifecycle processes	Practical assignment	n/a
Applies, evaluates, reports and provides advice on sustainability standards, regulations, measures, and methods.	Exam	n/a
Is aware of ethical considerations and issues and applies these in professional contexts and activities. Forms and communicates an opinion based on considerations of relevant social, professional, scientific and ethical aspects	Practical assignment	n/a

1.2.14 Learning Resources - PLO 7. Profession related competence [EQF6]

LEARNING UNIT	EQF	Duration in hrs (ECTS)	Didactical Approach	ASSESSMENT	Title of the Learning material	Delivery method of the learning material	Quick link to learning materials
The place of development/ ICT in an organisation	5	16 (0,64 ECTS)	F2F, virtual, blended	Exam	Course materials	Virtual instructor-led training (VILT), F2F lectures	-
Get trained and informed: Technology watch	5	8 (0,32 ECTS)	F2F, virtual, e-learning/ videos	Practical exercises	Course materials	Virtual instructor-led training (VILT), Training lecture, F2F lectures,	-
Team project: Implementing of a software factory	6	40 (1,6 ECTS)	F2F, virtual, blended	Practical assignment, presentation	Workbook	Training lectures, Virtual instructor-led training (VILT), Team project, Solving a problem together	-

1.2.15 PLO 8. Soft competences [EQF6]

8. PLO Soft competences [EQF6]

The learner has demonstrated capability

→ to apply soft skills

Unit learning outcomes	Manages teamwork processes and facilitates collaboration to reach common objectives, e.g., handles conflicts, negotiates, motivates, and persuades.
	Communicates with peers, colleagues, supervisors and or relevant others, specialists and non-specialists, and clients, appropriately to the scientific and professional community, using conventions which are relevant. Applies communication to the objective and the target group.
	Masters the English language at level B2. Can understand the main ideas of complex text on both concrete and abstract topics, including technical discussions in his/her field of specialisation
	Related to the occupation and knowledge domain, critically collects detailed professional and/or scientific information on a limited range of basic theories, principles and concepts, as well as limited information on some important current issues and topics. Analyses, evaluates, and combines critically this information, knowledge and insights and presents this. Critically applies/ translates/ interprets results of research (possibly executed by others) to the own context (the occupation and/or knowledge domain). Executes applied and practice-oriented research.
	Identifies and analyses complex and unpredictable problems. Solves these problems in a tactical, strategic and creative way by selecting and using data and by using one's creativity, flexibility and inventiveness.
	Exercises self-management in complex technical or professional activities or projects, taking responsibility for decision making in unpredictable work or study contexts. Is able to cope with change (positive or negative) and to adapt to a considerable level of variety in the workplace. Handles pressure and setbacks and maintains composure. Shows initiative, creativity and some originality and carries responsibility for the results of own activities, work and or study and for the work results of others. Works correctly and carefully, fully aware of the importance of trustworthiness and accountability.
	Realises learning and personal development on one's own initiative, by reflecting on and evaluating personal (learning) results. Selects and uses training/instructional methods and procedures appropriate for the situation when learning.

1.2.15.1 Duration of Study

Recommended duration: starting from 5 ECTS as an absolute minimum.

Often integrated with studies of PLOs: This PLO addresses competences that relate to more generic aspects in work related contexts. It is often integrated with: PLO 9. Functioning in organisations. The problem solving and critical analysis aspects of this PLO are often intertwined with PLO's 1. Application development, PLO 5. Systems Engineering and PLO 6. Change Support.

1.2.15.2 Recommendations for Micro-credentials

Due to its strong intertwining with other PLOs, it may be difficult to offer this PLO in its entirety as a stand-alone micro-credential. However, this PLO can be given explicit attention in combination with other PLOs such as mentioned above. Any more theoretical parts of this PLO could be offered separately as micro-credential for skilling (ICT) professionals, such as:

- Time management
- Creativity development

1.2.15.3 Recommendations on Didactical Approach, Delivery Methods and Training Environment

Recommended didactical approach:

- F2F classroom
- Virtual classroom
- Blended
- e-Learning
- In-company

Additional comments

To maximise accessibility and flexibility it is recommended that different didactical approaches are used as much as possible, so that the individual learner can decide what suits best. Besides this, offering in-company courses and training supports accessibility and flexibility.

Recommended delivery methods:

- Lectures virtual
- Virtual instructor-led training (VILT)
- Practical exercises
- Group/ teamwork
- Team project

Additional comments

Lectures, e-learning and virtual instructor-led training are recommended for learning the basic principles, terminology, and methods related to soft competences. These should be reinforced through practical tasks, individual and group assignments, and if possible, a team project. The ratio between on the one hand lectures/ VILT and on the other hand practical work done by the learners should be appr. 20% (or less) - 80% (or more) respectively.

1.2.15.4 WBL and Follow-up Reinforcement

After learning the basic principles of the different soft competences, the programme should focus on real work-life-like tasks as, for example:

- Practical exercises, based on real life situations, e.g., case studies
- Working together in a group or a team to analyse and solve specific problems

1.2.15.5 Important (new) approaches and technologies to consider

n/a

1.2.15.6 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Manages teamwork processes and facilitates collaboration to reach common objectives, e.g., handles conflicts, negotiates, motivates, and persuades.	Practical assignment	n/a
Communicates with peers, colleagues, supervisors and or relevant others, specialists and non-specialists, and clients, appropriately to the scientific and professional community, using conventions which are relevant. Applies communication to the objective and the target group.	Practical assignment	n/a
Masters the English language at level B2. Can understand the main ideas of complex text on both concrete and abstract topics, including technical discussions in his/her field of specialisation	Practical assignment	n/a
Related to the occupation and knowledge domain, critically collects detailed professional and/or scientific information on a limited range of basic theories, principles and concepts, as well as limited information on some important current issues and topics. Analyses, evaluates, and combines critically this information, knowledge and insights and presents this. Critically applies/ translates/ interprets results of research (possibly executed by others) to the own context (the occupation and/or knowledge domain). Executes applied and practice-oriented research.	Practical assignment	n/a
Identifies and analyses complex and unpredictable problems. Solves these problems in a tactical, strategic and creative way by selecting and using data and by using one's creativity, flexibility and inventiveness.	Practical assignment	n/a
Exercises self-management in complex technical or professional activities or projects, taking responsibility for decision making in unpredictable work or study contexts. Is able to cope with change (positive or negative) and to adapt to a considerable level of variety in the workplace. Handles pressure and setbacks and maintains composure. Shows initiative, creativity and some originality and carries responsibility for the results of own activities, work and or study and for the work results of others. Works correctly	Practical assignment	n/a

and carefully, fully aware of the importance of trustworthiness and accountability.		
Realises learning and personal development on one's own initiative, by reflecting on and evaluating personal (learning) results. Selects and uses training/instructional methods and procedures appropriate for the situation when learning.	Practical assignment	n/a

1.2.16 Learning Resources - PLO 8. Soft competences [EQF6]

LEARNING UNIT	EQF	Duration in hrs (ECTS)	Didactical Approach	ASSESSMENT (formative+summative)	Title of the Learning material	Delivery method of the learning material	Quick link to learning materials
Team working: - <u>Integration and teamwork</u> - <u>Communication</u> - <u>Assertiveness and place in a team</u> - <u>Preventing and managing difficult situations</u>	6	32 (1,3 ECTS)	F2F, virtual	Practical exercises	Course materials, workbook	Virtual instructor-led training (VILT), Training lecture, practical exercises, work in groups	-
Get trained and informed: - <u>Managing time and priorities</u> - <u>Developing creativity</u>	5	16 (0,64 ECTS)	F2F, virtual, e-learning/ videos	Practical exercises	Course materials: - Creative thinking for professional efficiency - Trainee booklet - Creative thinking for professional efficiency – Exercises	Virtual instructor-led training (VILT), Training lecture, practical exercises, work in groups	https://learn.softwareskills.eu/wp-content/uploads/2023/11/Creative-thinking-for-professional-efficiency-Trainee-booklet-Dev.pdf https://learn.softwareskills.eu/wp-content/uploads/2023/11/Creative-thinking-for-professional-efficiency-Exercises-Dev.pdf

					<ul style="list-style-type: none"> - Manage your time and priorities - Trainee booklet - Manage your time and priorities - Exercises 		
Team project: Implementing of a software factory	6	40 (1,6 ECTS)	F2F, virtual, blended	Practical assignment, presentation	Workbook	Training lectures, Virtual instructor-led training (VILT), Team project, Solving a problem together	-

1.2.17 PLO 9. Functioning in organisations [EQF6]

9. PLO Functioning in organisations [EQF6]

The learner has demonstrated capability

→ to function in an organisational context

Unit learning outcomes	Explains the basics of organisation theory and behaviour
	Describes the relationship between business and IT
	Works in an organisational context under specific direction with limited autonomy and responsibility e.g., at the level of a trainee, junior or assistant
	Manages a project, selects appropriate project management methods and tools
	Writes a report on functioning in organisation

1.2.17.1 Duration of Study

Recommended duration: starting from 5 ECTS as an absolute minimum. If bigger and more complex practical assignments are used such as groupwork and a team project, then 7 ECTS is a minimum.

Often integrated with studies of PLOs: This PLO addresses topics that relate to more generic aspects in work related contexts. It is often integrated with: PLO's 7. Profession related competences and 8. Soft competences.

1.2.17.2 Recommendations for Micro-credentials

This PLO and especially its subsequent parts can be offered as a micro-credential as part of a modular (re)skilling programme for learners with no prior knowledge of software development. This PLO is also recommended as an independent stand-alone micro-credential for skilling and reskilling (ICT) professionals that are interested in learning the basics of these topics. Parts of this PLO are also good candidates for micro-credentials, such as:

- Introduction to project management
- Introduction to agile

1.2.17.3 Recommendations on Didactical Approach, Delivery Methods and Training Environment

Recommended didactical approach:

- F2F classroom
- Virtual classroom
- Blended
- e-Learning
- In-company

Additional comments

To maximise accessibility and flexibility it is recommended that different didactical approaches are used as much as possible, so that the individual learner can decide what suits best. Besides this, offering in-company courses and training supports accessibility and flexibility.

Recommended delivery methods:

- Lectures F2F ☒
- Lectures virtual ☒
- Lectures blended ☒
- Virtual instructor-led training (VILT) ☒
- Practical exercises ☒
- Group/ teamwork ☒
- Team project ☒

Additional comments

Lectures, e-learning and virtual instructor-led training are recommended for learning the basic principles, terminology, and methods related to functioning in an organisation and project management. These should be reinforced through practical tasks, individual and group assignments, and if possible, a team project. The ratio between on the one hand lectures/ VILT and on the other hand practical work done by the learners should be appr. 60% - 40% respectively for this PLO overall as an average.

For learning units that address parts of this PLO (e.g. manages a project or works in an organizational context) this ratio should be appr. 20% (or less) - 80% (or more).

1.2.17.4 WBL and Follow-up Reinforcement

After learning the basic principles, terminology, and models of organisations and project management, the programme should focus on analysing and simulating real work-life-like tasks as, for example:

- Practical exercises, based on real life situations, e.g., case studies
- Working together in a team to solve specific problems

1.2.17.5 Important (new) approaches and technologies to consider

n/a

1.2.17.6 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Explains the basics of organisation theory and behaviour	Exam	n/a
Describes the relationship between business and IT	Exam	n/a

Works in an organisational context under specific direction with limited autonomy and responsibility e.g., at the level of a trainee, junior or assistant	Practical assignment	n/a
Manages a project, selects appropriate project management methods and tools	Practical assignment	n/a
Writes a report on functioning in organisation	Practical assignment	n/a

1.2.18 Learning Resources - PLO 9. Functioning in organisation [EQF6]

LEARNING UNIT	EQF	Duration in hrs (ECTS)	Didactical Approach	ASSESSMENT (formative+summative)	Title of the Learning material	Delivery method of the learning material	Quick link to learning materials
The place of development in an organization	5	16 (0,64 ECTS)	F2F, virtual, blended	Exam	Course materials	Virtual instructor-led training (VILT), F2F lectures	-
Work in project situation: - Introduction to project management - Introduction to Agile	5	24 (0,96 ECTS)	F2F, virtual, blended, e-learning/ videos	Practical exercises	Course materials	Virtual instructor-led training (VILT), F2F lectures, practical exercises, work in groups	-
Team project: Implementing of a software factory	6	40 (1,6 ECTS)	F2F, virtual, blended	Practical assignment, presentation	Workbook	Training lectures, Virtual instructor-led training (VILT), Team project, Solving a problem together	-

1.2.19 EXTRA CURRICULAR PLO: New Technology [EQF6]

PLO New Technology [EQF6]

*The learner has demonstrated capability
→ to keep up to date with new technologies*

Unit learning outcomes	Explains the principles, related concepts, advantages and disadvantages of a new technology
	Applies methods, techniques and tools related to a new technology
	Writes a report on a new technology or of a method, technique or tool related to it

1.2.19.1 Duration of Study

Recommended duration: starting from 1 ECTS

Often integrated with studies of PLOs: This PLO addresses topics that can relate to many aspects within the field of ICT, therefore also many different PLO's can be involved.

1.2.19.2 Recommendations for Micro-credentials

This PLO can be offered as a micro-credential as part of a modular (re)skilling programme for learners with no prior knowledge of software development. This PLO is also recommended as an independent stand-alone micro-credential for skilling and reskilling (ICT) professionals that are interested in learning about new technologies in the field of ICT.

1.2.19.3 Recommendations on Didactical Approach, Delivery Methods and Training Environment

Recommended didactical approach:

- F2F classroom
- Virtual classroom
- Blended
- In-company

Additional comments

To maximise accessibility and flexibility it is recommended that different didactical approaches are used as much as possible, so that the individual learner can decide what suits best. Besides this, offering in-company courses and training supports accessibility and flexibility.

Recommended delivery methods:

- Lectures F2F
- Lectures virtual
- Lectures blended
- Virtual instructor-led training (VILT)
- Practical exercises

Additional comments

Lectures and virtual instructor-led training are recommended for learning how to stay up to date with new tools and technologies. These should be reinforced through practical exercises.

1.2.19.4 WBL and Follow-up Reinforcement

n/a

1.2.19.5 Important (new) approaches and technologies to consider

n/a

1.2.19.6 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Staying up to date with new technologies	Exam	n/a

1.2.20 Learning Resources - EXTRA CURRICULAR PLO: New Technology [EQF6]

LEARNING UNIT	EQF	Duration in hrs (ECTS)	Didactical Approach	ASSESSMENT	Title of the Learning material	Delivery method of the learning material	Quick link to learning materials
Technology watch	5	8 (0,32 ECTS)	F2F, virtual, blended	Exam	Course materials	Virtual instructor-led training (VILT), F2F lectures	-

1.3 Students and professionals in need of upskilling/reskilling

Executive summary

This Learning Programme is being designed The University of Ljubljana, Faculty of Computer and Information Science (SI).

The University of Ljubljana, Faculty of Computer and Information Science (UL) identified the domain of software engineering as a key area where up-skilling would be necessary on domestic grounds. The targeted professionals are those that are already engaged in the software engineering industry and need to upgrade their software engineering skills, or in any other technical industry sector where professionals would need to acquire new software engineering skills that are currently needed due to business processes modernization. Hence, the UL's programme of Jr DevOps Expert EQF6.

In order to support our activities, UL recently launched AkademijaFRI (<https://akademijafri.si/>) a business entity that is an integral part of the Faculty of Information and Computer Science, and will be used to deliver the DevOps Expert EQF6 study programme. The study programme consists of 19 ECTS and may end up with a diploma as a final certificate. In order to achieve comparability among the courses, each course (that is, PLO) is designed as 2 ECTS course, except for the course Virtualisation that is designed as 1 ECTS. The 2 ECTS courses consist of 50 study and work hours, this includes 12 teaching hours, and 4 laboratory classes (1 class has the duration of 2hours, that is, altogether 8 hours), home study and project work (25 hours), and an examination and presentations of project work (5 hours), that is planned to extend throughout a period of 3 weeks. All courses are designed to be taken as integral study courses within a period of 2-3 years, depending on the upskilling requirements of the candidates.

The DevOps Junior Expert programme is designed to provide advanced knowledge in the field of DevOps, involving a critical understanding of important theories and principles, The courses are designed to develop advanced skills, demonstrating mastery and innovation, required to solve complex and unpredictable problems in the DevOps field, All attendees are expected to autonomously manage complex DevOps projects, taking responsibility for decision-making in unpredictable software engineering contexts; This study programme will enable the participants to responsibly apply the DevOps culture and practice within groups of developers.

- Following is a summary of the study programme:
- Educational profile name: DevOps Expert EQF6
- Targeted Institutions: VET, IT companies, digital enthusiasts
- Target Groups: professional in upskilling/reskilling path
- Overall number of the Learning Units proposed by your Curriculum Learning Program 10 units
- Overall number of hours concerned by your Curriculum Learning Program – 475 of lectures, lab sessions, real-world problem solving, seminars, individual students work and preparation of thesis.
- Overview of the delivery method/s recommended: Presence Classroom, Virtual Classroom, Blended, e-learning, Work placement

- Localization characteristics: The study programme is especially designed for upskilling in the Slovenian industrial (software engineering, electrotechnical engineering, machine engineering, and other related disciplines) sector.

1.3.1 PLO 1. Application Development [e-3]

1. PLO Application Development [e-3]

The learner has demonstrated capability

→ to creatively develop software applications and components, by interpreting the software design

→ to optimise the application development

Unit learning outcomes	Organises data and creates a structured dataset
	Writes code and related documentation to it, using programming languages (e.g., Java, Javascript, PHP, Python) and tools (e.g., GitHub), applying programming principles (e.g., clean coding, green coding, secure programming) and other relevant practices (e.g. continuous integration, test-driven development), principles or constraints (e.g., privacy legislation, intellectual property law)
	Efficiently creates a working software component/ application taking into account design requirements (e.g., scalability, availability) and other relevant constraints (e.g., architecture, efficiency, cost, quality, energy consumption) and applying relevant tools and techniques (e.g., object-oriented programming; IDE, CASE; editors, compilers; version control management and tools; multimedia integration tools; app development tools; reuse of proved solutions)
	Modifies an existing software component/ application, in order to optimise it (e.g., to improve scalability, maintenance, performance, security).

1.3.1.1 Duration of Study

Recommended duration: starting from 2 ECTS in order to offer a balanced programme that covers all the 10 PLOs. **Often integrated with studies of PLOs:** 2, 5, 9.

1.3.1.2 Recommendations for Micro-credentials

UL will provide qualitative microcredentials that represent the key competences acquired by the student within this PLO, subject to further refinement and agreement at project level. Examples include:

“Writes code and related documentation to it, using programming languages (e.g., Java, Javascript, PHP, Python) and tools (e.g., GitHub).”

“Applyies programming principles (e.g., clean coding, green coding, secure programming) and other relevant practices (e.g. continuous integration, test-driven development), principles or constraints (e.g., privacy legislation, intellectual property law).”

And similar to all other competences provided by this PLO.

1.3.1.3 Recommendations on Didactical Approach, Delivery Methods, and Training Environment

Recommended didactical approach:

- Presence Classroom
- Virtual Classroom
- Blended
- e-learning
- Work placement

Additional comments

N/A

Recommended delivery methods:

- Lectures and laboratory classes up to 50%
- Case study. Individual/team project (WBL) up to 50%

Additional comments

Lectures, e-learning are recommended for learning the basic principles, terminology, and models of software design. These should be reinforced through practical tasks, case studies, individual/team-projects.

1.3.1.4 WBL and Follow-up Reinforcement

After learning the basic principles, terminology, and models of software design, the study should focus on analysing and simulating real work-life-like tasks as, for example:

- Designing simple design models & diagrams for a real-life-like customer project
- Internship tasks of interpreting designs for applications

1.3.1.5 Important (new) approaches and technologies to consider

- Understanding the basics of application lifecycle
- Getting to know the process of application development
- Getting to know tools and applications used in the process of application development

1.3.1.6 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Organises data and creates a structured dataset	Exam	Assesment (of skills)
Writes code and related documentation to it, using programming languages (e.g., Java, Javascript, PHP,	Practical assignment	Assesment (of skills)

<p>Python) and tools (e.g., GitHub), applying programming principles (e.g., clean coding, green coding, secure programming) and other relevant practices (e.g. continuous integration, test-driven development), principles or constraints (e.g., privacy legislation, intellectual property law)</p>		
<p>Efficiently creates a working software component/application taking into account design requirements (e.g., scalability, availability) and other relevant constraints (e.g., architecture, efficiency, cost, quality, energy consumption) and applying relevant tools and techniques (e.g., object-oriented programming; IDE, CASE; editors, compilers; version control management and tools; multimedia integration tools; app development tools; reuse of proved solutions)</p>	<p>Presentation of practical assignment</p>	<p>Assesment (of skills)</p>

1.3.2 Learning Resources - PLO 1. Application Development [e-3]

LEARNING UNIT	EQF	Duration	Didactical Approach	ASSESSMENT	Title of the Learning material	Delivery method of the learning material	Quick link to learning materials
Web Development: Backend	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, problem-based learning, collaborative-based learning.	Exam, Practical assignment and Presentation of Practical assignment	Node, MVC, Other environments Rest API User Authentiation	Lectures, laboratory sessions	https://learn.softwareskills.eu/wp-content/uploads/2024/01/Lavbic-Node-MVC-other-environments-ESSA_Learning-Programmes-and-Materials.pdf https://learn.softwareskills.eu/wp-content/uploads/2024/01/Lavbic-Rest-API-ESSA_Learning-Programmes-and-Materials.f.pdf
Web Development: Frontend	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week	Lectures, problem-based learning, collaborative-based learning.	Practical assignment	Angular SPA Designing with Swagger and OpenAPI	Lectures, laboratory sessions	https://learn.softwareskills.eu/wp-content/uploads/2024/01/Lavbic-Angular-SPA-ESSA_Learning-

	exam presentations	and		Type script and Angular		Programmes-and-Materials.pptx
				User Authentiation		https://learn.softwareskills.eu/wp-content/uploads/2024/01/Lavbic-Type-script-and-Angular-ESSA_Learning-Programmes-and-Materials_f.pptx
						https://learn.softwareskills.eu/wp-content/uploads/2023/11/plo3Angular-React-Jest-Mocha-Selenium.pptx
						https://learn.softwareskills.eu/wp-content/uploads/2023/11/Angular-React-Jest-Mocha-Selenium.pptx

PLO 2. Component integration [e-3]

2. PLO Component Integration [e-3]

The learner has demonstrated capability

→ to integrate efficiently a software application or component into an existing system, compliant with standards and procedures

→ to document the installation activities

Unit learning outcomes

Integrates a solution, software application or component applying relevant practices, methods, techniques and tools, compliant with appropriate standards and procedures (e.g. configuration management, version management, change control, packaging and distribution, virtualisation, containerisation)

Monitors, verifies and tests system capacity and performance, using appropriate techniques and tools

Writes an installation report/ installation documentation

1.3.2.1 Duration of Study

Recommended duration: starting from 2 ECTS

Often integrated with studies of PLOs: Connected to PLO's 4, 5, 8, 9, 10

1.3.2.2 Recommendations for Micro-credentials

- *This PLO should be an integral part of the initial studies for students with no prior knowledge of software development.*
- *Recommended as an independent micro-credential for upskilling junior developers and junior DevOps (EQF 4, 5).*
- *Microcredentials will be assigned in qualitative form according to the competences acquired.*

1.3.2.3 Recommendations on Didactical Approach, Delivery Methods and Training Environment

Recommended didactical approach:

- Presence Classroom
- Virtual Classroom
- Blended
- e-learning
- Work placement

Additional comments

n/a

Recommended delivery methods:

- Lecture and laboratory session up to 50%

- Case study. Individual/team project up to 50%

Additional comments

Lectures, e-learning are recommended for learning the basic principles, terminology, and models of software design. These should be reinforced through practical tasks, case studies, individual/team-projects in the same amount (time and ECTS-wise) as theoretical work. Lectures should be supported by laboratory sessions.

1.3.2.4 WBL and Follow-up Reinforcement

After learning the basic principles, terminology, and models of components integration, the study should focus on analysing and simulating real work-life-like tasks as, for example:

- Each participant chooses a custom domain (from work or life) and create an application that can later be used in real life
- Internship tasks of interpreting designs for applications

1.3.2.5 Important (new) approaches and technologies to consider

- Upgrade of custom domain project
- Learning about modern approaches and demand on the labour market
- Learning about different tools available

1.3.2.6 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Integrates a solution, software application or component applying relevant practices, methods, techniques and tools, compliant with appropriate standards and procedures (e.g. configuration management, version management, change control, packaging and distribution, virtualisation, containerisation)	Practical assignment	Assessment of skills
Monitors, verifies and tests system capacity and performance, using appropriate techniques and tools	Exam	Assessment of skills
Writes an installation report/ installation documentation	Presentation	Assessment of skills

1.3.3 Learning Resources - PLO 2. Component integration [e-3]

LEARNING UNIT	EQF	Duration	Didactical Approach	ASSESSMENT	Title of the Learning material	Delivery method of the learning material	Quick link to learning materials
Web development: Frontend	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, laboratory sessions, problem-based learning, collaborative-based learning.	Exam, Practical assignment, Presentation of practical assignment	Angular SPA Designing with Swagger and OpenAPI Type script and Angular User Authentiation	Lectures, laboratory sessions	https://learn.softwareskills.eu/wp-content/uploads/2024/01/Lavbic-Angular-SPA-ESSA_Learning-Programmes-and-Materials.pptx https://learn.softwareskills.eu/wp-content/uploads/2024/01/Lavbic-Designing-with-Swagger-and-OpenAPI-ESSA_Learning-Programmes-and-Materials.pptx https://learn.softwareskills.eu/wp-content/uploads/2024/01/Lavbic-Type-script-and-Angular-ESSA_Learning-Programmes-and-Materials_f.pptx https://learn.softwareskills.eu/wp-content/uploads/2024/01/Lavbic-User-Authentication-ESSA_Learning-Programmes-and-Materials_f.pptx

<p>Web development: Backend</p>	<p>6</p>	<p>3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations</p>	<p>Lectures, laboratory sessions, problem-based learning, collaborative-based learning.</p>	<p>Exam, Practical assignment, Presentation of practical assignment</p>	<p>Node, MVC, Other environments Rest API User Authentiaction</p>	<p>Lectures, laboratory sessions</p>	<p>https://learn.softwareskills.eu/wp-content/uploads/2024/01/Lavbic-Node-MVC-other-environments-ESSA_Learning-Programmes-and-Materials.pdf</p> <p>https://learn.softwareskills.eu/wp-content/uploads/2024/01/Lavbic-Rest-API-ESSA_Learning-Programmes-and-Materials_f.pdf</p> <p>https://learn.softwareskills.eu/wp-content/uploads/2024/01/Lavbic-User-Authentication-ESSA_Learning-Programmes-and-Materials_f.pptx</p>
<p>Virtualization</p>	<p>6</p>	<p>3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations</p>	<p>Lectures, laboratory sessions, problem-based learning, collaborative-based learning.</p>	<p>Exam, Practical assignment, Presentation of practical assignment</p>	<p>Modern Infrastructures - Virtualization</p>	<p>Lectures, laboratory sessions</p>	<p>https://learn.softwareskills.eu/wp-content/uploads/2023/12/ESSA-Junior-Developer_Virtualization.pptx</p>

<i>Orchestration</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, laboratory sessions, problem-based learning, collaborative-based learning.	Exam, Practical assignment, Presentation of practical assignment	Modern Infrastructures - Orchestration	Lectures, laboratory sessions	https://learn.softwareskills.eu/wp-content/uploads/2024/01/Modern-infrastructures-DevOps-best-practices-ESSA-1.pdf
<i>Containerization</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, laboratory sessions, problem-based learning, collaborative-based learning.	Exam, Practical assignment, Presentation of practical assignment	Modern Infrastructures - Containers Modern Infrastructures - Kubernetes	Lectures, laboratory sessions	https://learn.softwareskills.eu/wp-content/uploads/2024/01/Modern-infrastructures-Containers-ESSA_f.pdf

1.3.4 PLO 3. Testing [e-3]

3. PLO Testing [e-3]

The learner has demonstrated capability

→ to construct and execute tests for solutions, software applications or components

→ to document test outcomes

Unit learning outcomes	Writes (parts of) test related documentation (e.g. test plan, test strategy/approach, test case, test script, test scenario, test conditions)
	Constructs tests, by selecting appropriate test methods, techniques, and tools (e.g. test automation tools)
	Defines and configures a test environment
	Executes test cases and performs test activities related to different sorts of tests
	Records and interprets test outcomes and writes test result documentation/ test report

1.3.4.1 Duration of Study

Recommended duration: starting from 2 ECTS

Often integrated with studies of PLOs: 3, 9, 10

1.3.4.2 Recommendations for Micro-credentials

Please provide specific indications, as the following example:

- Recommended as an independent micro-credential for upskilling junior developers and junior DevOps (for example EQF 4 and 5).

1.3.4.3 Recommendations on Didactical Approach, Delivery Methods and Training Environment

Recommended didactical approach:

- Presence Classroom
- Virtual Classroom
- Blended
- e-learning
- Work placement

Additional comments

n/a

Recommended delivery methods:

- Lecture and laboratory sessions up to 50%
- Project work/ WBL activity up to 50%

Additional comments

Lectures, e-learning are recommended for learning the basic principles, terminology, and models of software design. These should be reinforced through practical tasks, case studies, individual/team-projects in the same amount (time and ECTS-wise) as theoretical work. Lectures should be supported by laboratory sessions.

1.3.4.4 WBL and Follow-up Reinforcement

After learning the basic principles, terminology, and models of software design, the study should focus on analysing and simulating real work-life-like tasks as, for example:

- Each participant chooses a custom domain (from work or life) and create an application that can later be used in real life
- Internship tasks of interpreting designs for applications

1.3.4.5 Important (new) approaches and technologies to consider

- Creating a test plan
- Understanding the importance of testing
- Getting to know tools available for testing

1.3.4.6 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Writes (parts of) test related documentation (e.g. test plan, test strategy/approach, test case, test script, test scenario, test conditions)	Exam	Assessment of skills
Constructs tests, by selecting appropriate test methods, techniques, and tools	Practical assignment	Assessment of skills
(e.g. test automation tools)	Presentation of practical assignment	Assessment of skills

1.3.5 Learning Resources - PLO 3. Testing [e-3]

LEARNING UNIT	EQF	Duration	Didactical Approach	ASSESSMENT	Title of the Learning material	Delivery method of the learning material	Quick link to learning materials
<i>Software testing</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, problem-based learning, collaborative-based learning.	Exam, Practical assignment, Presentation of Practical Assignment	Testing	Lectures, laboratory sessions	https://learn.softwareskills.eu/wp-content/uploads/2023/11/DEVOPS-EXPERT-EQF7-TESTING-CONFIGURATION-MANAGEMENT-LU.docx

1.3.6 PLO 4. Solution Deployment [e-3]

4. PLO Solution Deployment [e-3]

The learner has demonstrated capability

→ to implement solutions and services

→ to complete release documentation

Unit learning outcomes	Writes a release plan (e.g., including solution verification and validation, documentation, supporting activities; deployment workflow and product roll-out activities)
	Writes a plan for data management/ migration
	Executes (parts of) a solution/ software release, applying appropriate methods, techniques, and tools (e.g., CI/CD tools; tools related to automated software release, software packaging and distribution)
	Writes (parts of) release documentation

1.3.6.1 Duration of Study

Recommended duration: starting from 2 ECTS

Often integrated with studies of PLOs: 2, 5, 8, 9

1.3.6.2 Recommendations for Micro-credentials

Please provide specific indications, as the following example:

- Recommended as an independent micro-credential for upskilling junior developers and junior DevOps (for example EQF 4, 5).

1.3.6.3 Recommendations on Didactical Approach, Delivery Methods and Training Environment

Recommended didactical approach:

- Presence Classroom
- Virtual Classroom
- Blended
- e-learning
- Work placement

Additional comments

n/a

Recommended delivery methods:

- Lecture and laboratory sessions up to 50%
- Case study. Individual/team project up to 50%

Additional comments

Lectures, e-learning are recommended for learning the basic principles, terminology, and models of software design. These should be reinforced through practical tasks, case studies, individual/team-projects in the same amount (time and ECTS-wise) as theoretical work. Lectures should be supported by laboratory sessions.

1.3.6.4 WBL and Follow-up Reinforcement

After learning the basic principles, terminology, and models of software design, the study should focus on analysing and simulating real work-life-like tasks as, for example:

- Internship tasks of interpreting designs for applications
- Each participant chooses a custom domain (from work or life) and create an application that can later be used in real life

1.3.6.5 Important (new) approaches and technologies to consider

- Understanding integration of information systems
- Understanding the pipeline in real-life contexts
- Learning about managing applications
- Getting to know approaches and most used tools

1.3.6.6 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Writes a plan for data management/migration	Exam	Assesment of skills
Executes (parts of) a solution/ software release, applying appropriate methods, techniques, and tools (e.g., CI/CD tools; tools related to automated software release, software packaging and distribution)	Practical assignment	Assessment of skills
Writes (parts of) release documentation	Presentation	Assessment of skills

1.3.7 Learning Resources PLO 4. Solution Deployment [e-3]

LEARNING UNIT	EQF	Duration	Didactical Approach	ASSESSMENT	Title of the Learning material	Delivery method of the learning material	Quick link to learning materials
Web Development: Frontend	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assessment, Presentation	Angular SPA Designing with Swagger and OpenAPI Type script and Angular User Authentiaction	Lectures, laboratory sessions	https://learn.softwareskills.eu/wp-content/uploads/2024/01/Lavbic-Angular-SPA-ESSA_Learning-Programmes-and-Materials.pptx https://learn.softwareskills.eu/wp-content/uploads/2024/01/Lavbic-Designing-with-Swagger-and-OpenAPI-ESSA_Learning-Programmes-and-Materials.pptx https://learn.softwareskills.eu/wp-content/uploads/2024/01/Lavbic-Type-script-and-Angular-ESSA_Learning-Programmes-and-Materials_f.pptx https://learn.softwareskills.eu/wp-content/uploads/2024/01/Lavbic-User-Authentication-ESSA_Learning-Programmes-and-Materials_f.pptx

<i>Virtualization</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assessment, Presentation assessment	Modern Infrastructures - Virtualization	Lectures, sessions laboratory	https://learn.softwareskills.eu/wp-content/uploads/2023/12/ESSA-Junior-Developer_Virtualization.pptx
<i>Orchestration</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assessment, Presentation	Modern Infrastructures - Orchestration	Lectures, sessions laboratory	https://learn.softwareskills.eu/wp-content/uploads/2024/01/Modern-infrastructures-DevOps-best-practices-ESSA-1.pdf
<i>Containerization</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1	Lectures, Laboratory sessions, problem-based learning,	Exam, Practical assessment, Presentation	Modern Infrastructures - Containers	Lectures, sessions laboratory	https://learn.softwareskills.eu/wp-content/uploads/2024/01/Modern-infrastructures-Containers-ESSA_f.pdf

		weeks of individual student work on practical assignment, 1 week exam and presentations	collaborative-based learning, WBL.		Modern Infrastructures - Kubernetes		
<i>Software release</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assessment, Presentation	-	Lectures, laboratory sessions	- .

1.3.8 PLO 5. ICT Systems Engineering [e-3]

5. PLO ICT Systems Engineering [e-3]

The learner has demonstrated capability

→ to create a system infrastructure that meets requirements

→ to ensure interoperability of system components

Unit learning outcomes	Describes software and hardware components, tools and architectures, including network components, topologies, protocols and interconnections
	Analyses existing system infrastructures, applying appropriate methods, techniques, and tools (e.g. related to interoperability of components, security, energy consumption)
	Proposes improvements to an existing system infrastructure, to better meet requirements (e.g., cloud solutions, Infrastructure as Code, Containers as a Service, virtual machines, load balancers)
	Designs (parts of) a DevOps pipeline, by formulating a set of practices and tools that the development and operations teams may implement to build, test, and deploy software

1.3.8.1 Duration of Study

Recommended duration: starting from 2 ECTS

Often integrated with studies of PLOs: 2, 4, 8, 9

1.3.8.2 Recommendations for Micro-credentials

- Recommended as an independent micro-credential for upskilling junior developers and junior DevOps (EQF 4 and 5).

1.3.8.3 Recommendations on Didactical Approach, Delivery Methods and Training Environment

Recommended didactical approach:

- Presence Classroom
- Blended
- Work placement

Additional comments

n/a

Recommended delivery methods:

- Lecture and laboratory sessions up to 50%
- Practical assignment up to 50%

Additional comments

Lectures, e-learning are recommended for learning the basic principles, terminology, and models of software design. These should be reinforced through practical tasks, case studies, individual/team-projects in the same amount (time and ECTS-wise) as theoretical work. Lectures should be supported by laboratory sessions.

1.3.8.4 WBL and Follow-up Reinforcement

After learning the basic principles, terminology, and models of software design, the study should focus on analysing and simulating real work-life-like tasks as, for example:

- Internship tasks of interpreting designs for applications
- Each participant chooses a custom domain (from work or life) and create an application that can later be used in real life

1.3.8.5 Important (new) approaches and technologies to consider

- Learning about modern approaches
- Getting to know different tools
- Learning structural and behavioural system specifications

1.3.8.6 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Describes software and hardware components, tools and architectures, including network components, topologies, protocols and interconnections	Exam	Assessment of skills
Analyses existing system infrastructures, applying appropriate methods, techniques, and tools (e.g. related to interoperability of components, security, energy consumption)	Practical assignment	Assessment of skills
Proposes improvements to an existing system infrastructure, to better meet requirements (e.g., cloud solutions, Infrastructure as Code, Containers as a Service, virtual machines, load balancers)	Presentation	Assessment of skills

1.3.9 Learning Resources - PLO 5. ICT Systems Engineering [e-3]

LEARNING UNIT	EQF	Duration	Didactical Approach	ASSESSMENT	Title of the Learning material	Delivery method of the learning material	Quick link to learning materials
<i>Orchestration</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical Assignment, Presentation	Modern Infrastructures - Orchestration	Lectures, laboratory sessions	https://learn.softwareskills.eu/wp-content/uploads/2024/01/Modern-infrastructures-DevOps-best-practices-ESSA-1.pdf
<i>Containerization</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of	Lectures, Laboratory sessions, problem-based learning,	Exam, Practical Assignment, Presentation	Modern Infrastructures - Containers	Lectures, laboratory sessions	https://learn.softwareskills.eu/wp-content/uploads/2024/01/Modern-infrastructures-Containers-ESSA_f.pdf

		individual student work on practical assignment, 1 week exam and presentations	collaborative-based learning, WBL.		Modern Infrastructures - Kubernetes		
Virtualizaton	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical Assignment, Presentation	Modern Infrastructures - Virtualization	Lectures, laboratory sessions	https://learn.softwareskills.eu/wp-content/uploads/2023/12/ESSA-Junior-Developer_Virtualization.pptx

1.3.10 PLO 6. Change support [e-3]

6. Change support [e-3]

The learner has demonstrated capability

→ to oversee and control system changes

→ to take into account procedures, requirements and restrictions

Unit learning outcomes	Describes, distinguishes, and applies change management methods, tools and techniques
	Proposes and applies appropriate and efficient practices, measures and procedures to handle change and to reduce the impact of change (e.g., based on ITIL Change management, DevOps)
	Writes a change request/ change proposal (for addition, modification or removal of a service or software application/component) (e.g., describing benefit, risk, and impact of change, taking into account requirements and restrictions such as information security regulations, budget, SLA, conditions for system integrity)

1.3.10.1 Duration of Study

Recommended duration: starting from 2 ECTS

Often integrated with studies of PLOs: 9

1.3.10.2 Recommendations for Micro-credentials

- Recommended as an independent micro-credential for upskilling junior developers and junior DevOps (EQF 4, 5).

1.3.10.3 Recommendations on Didactical Approach, Delivery Methods and Training Environment

Recommended didactical approach:

- Presence Classroom
- Blended

Additional comments

n/a

Recommended delivery methods:

- Lecture and laboratory sessions up to 50%
- Case study. Individual/team project up to 50%

Additional comments

Lectures, e-learning are recommended for learning the basic principles, terminology, and models of software design. These should be reinforced through practical tasks, case studies,

individual/team-projects in the same amount (time and ECTS-wise) as theoretical work. Lectures should be supported by laboratory sessions.

1.3.10.4 WBL and Follow-up Reinforcement

After learning the basic principles, terminology, and models of software design, the study should focus on analysing and simulating real work-life-like tasks as, for example:

- Each participant chooses a custom domain (from work or life) and create an application that can later be used in real life
- Internship tasks of interpreting application configurations

1.3.10.5 Important (new) approaches and technologies to consider

- Learning about application adaptation
- Learning about tools and approaches, and requirements of application adaptation
- Learning probabilistic methods for application adaptation

1.3.10.6 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Describes, distinguishes, and applies change management methods, tools and techniques	Practical assignment	Assessment of skills
Proposes and applies appropriate and efficient practices, measures and procedures to handle change and to reduce the impact of change (e.g., based on ITIL Change management, DevOps)	Presentation	Assessment of skills
Writes a change request/ change proposal (for addition, modification or removal of a service or software application/component) (e.g., describing benefit, risk, and impact of change, taking into account requirements and restrictions such as information security regulations, budget, SLA, conditions for system integrity)	Exam	Assessment of skills

1.3.11 Learning Resources - PLO 6. Change support [e-3]

LEARNING UNIT	EQF	Duration	Didactical Approach	ASSESSMENT	Title of the Learning material	Delivery method of the learning material	Quick link to learning materials
<i>Application Adaptation</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assignment, Presentation	-	Lectures, laboratory sessions	-

1.3.12 PLO 7. Professional related competences [EQF6]

7. PLO Profession related competences [EQF6]

The learner has demonstrated capability

→ to apply profession related skills

Unit learning outcomes	Masters common ICT knowledge
	Explains the principles, related concepts, advantages, disadvantages, limitations and possible societal, environmental, and ethical issues related to the application of a new technology. Applies and reports on basic methods, techniques and tools related to a new technology.
	Applies, evaluates, reports and provides advice on security standards, regulations, measures, methods, tools, and techniques
	Applies, evaluates and provides advice on appropriate methods, tools and techniques related to software lifecycle processes
	Applies, evaluates, reports and provides advice on sustainability standards, regulations, measures, and methods.
	Is aware of ethical considerations and issues and applies these in professional contexts and activities. Forms and communicates an opinion based on considerations of relevant social, professional, scientific and ethical aspects

1.3.12.1 Duration of Study

Recommended duration: starting from 2 ECTS

Often integrated with studies of PLOs: 1, 2, 3, 4, 5, 6

1.3.12.2 Recommendations for Micro-credentials

- This PLO should be an integral part of all courses, we recommend including this mainly in laboratory sessions and WBL

1.3.12.3 Recommendations on Didactical Approach, Delivery Methods and Training Environment

Recommended didactical approach:

- Presence Classroom
- Blended
- Work placement

Additional comments

n/a

Recommended delivery methods:

- Lecture and laboratory sessions up to 50%
- Case study. Individual/team project up to 50%

Additional comments

Professional related competences should be an integral part of all courses. We recommend including these skills mainly through laboratory sessions and WBL components.

1.3.12.4 WBL and Follow-up Reinforcement

WBL (alongside with laboratory sessions or practical work) is the main part acquiring these skills.

1.3.12.5 Important (new) approaches and technologies to consider

- Learning about different steps working on a project
- Learning about different available tools for project management

1.3.12.6 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Masters common ICT knowledge	Practical assignment and presentation	Assessment of skills
Explains the principles, related concepts, advantages, disadvantages, limitations and possible societal, environmental, and ethical issues related to the application of a new technology. Applies and reports on basic methods, techniques and tools related to a new technology.	Practical assignment and presentation	Assessment of skills

1.3.13 Examples of Curriculum Learning Resources - 7. PLO Profession related competence [EQF6]

LEARNING UNIT	EQF	Duration	Didactical Approach	ASSESSMENT	Title of the Learning material	Delivery method of the learning material	Quick link to learning materials
<i>Web Development Frontend</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assignment, Presentation	N/A	Practical assignment, laboratory sessions, lessons	*there is no learning material on this topic. It is an integral part of managing own project and collaborating with other participants
<i>Web Development Backend</i>		3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assignment, Presentation	N/A	Practical assignment, laboratory sessions, lessons	*there is no learning material on this topic. It is an integral part of managing own project and collaborating with other participants

<i>Orchestration</i>	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assignment, Presentation	N/A	Practical assignment, laboratory sessions, lessons	*there is no learning material on this topic. It is an integral part of managing own project and collaborating with other participants
<i>Virtualization</i>	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assignment, Presentation	N/A	Practical assignment, laboratory sessions, lessons	*there is no learning material on this topic. It is an integral part of managing own project and collaborating with other participants
<i>Containerization</i>	3 weeks altogether: 1 week of lectures and lab work, 1	Lectures, Laboratory sessions, problem-based learning,	Exam, Practical assignment, Presentation	N/A	Practical assignment, laboratory sessions, lessons	*there is no learning material on this topic. It is an integral part of managing own project

		weeks of individual student work on practical assignment, 1 week exam and presentations	collaborative-based learning, WBL.				and collaborating with other participants
<i>Project Management</i>		3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assignment, Presentation	-	Practical assignment, laboratory sessions, lessons	-
<i>Software Testing</i>		3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assignment, Presentation	N/A	Practical assignment, laboratory sessions, lessons	*there is no learning material on this topic. It is an integral part of managing own project and collaborating with other participants

	week and presentations	exam					
--	------------------------------	------	--	--	--	--	--

1.3.14 PLO 8. Soft competences [EQF6]

8. PLO Soft competences [EQF6]

The learner has demonstrated capability

→ to apply soft skills

Unit learning outcomes	Manages teamwork processes and facilitates collaboration to reach common objectives, e.g., handles conflicts, negotiates, motivates, and persuades.
	Communicates with peers, colleagues, supervisors and or relevant others, specialists and non-specialists, and clients, appropriately to the scientific and professional community, using conventions which are relevant. Applies communication to the objective and the target group.
	Masters the English language at level B2. Can understand the main ideas of complex text on both concrete and abstract topics, including technical discussions in his/her field of specialisation
	Related to the occupation and knowledge domain, critically collects detailed professional and/or scientific information on a limited range of basic theories, principles and concepts, as well as limited information on some important current issues and topics. Analyses, evaluates, and combines critically this information, knowledge and insights and presents this. Critically applies/ translates/ interprets results of research (possibly executed by others) to the own context (the occupation and/or knowledge domain). Executes applied and practice-oriented research.
	Identifies and analyses complex and unpredictable problems. Solves these problems in a tactical, strategic and creative way by selecting and using data and by using one's creativity, flexibility and inventiveness.
	Exercises self-management in complex technical or professional activities or projects, taking responsibility for decision making in unpredictable work or study contexts. Is able to cope with change (positive or negative) and to adapt to a considerable level of variety in the workplace. Handles pressure and setbacks and maintains composure. Shows initiative, creativity and some originality and carries responsibility for the results of own activities, work and or study and for the work results of others. Works correctly and carefully, fully aware of the importance of trustworthiness and accountability.
	Realises learning and personal development on one's own initiative, by reflecting on and evaluating personal (learning) results. Selects and uses training/instructional methods and procedures appropriate for the situation when learning.

1.3.14.1 Duration of Study

Recommended duration: starting from 1 ECTS

Often integrated with studies of PLOs: 1, 2, 3, 4, 5, 6, 7, 9

1.3.14.2 Recommendations for Micro-credentials

- We recommend this PLO for all students in all courses. Particularly, this PLO should be realized through practical work and laboratory sessions.

1.3.14.3 Recommendations on Didactical Approach, Delivery Methods and Training Environment

Recommended didactical approach:

- Presence Classroom
- Blended
- Work placement

Additional comments

Recommended delivery methods:

- Lecture and laboratory session up to 50%
- Case study. Individual/team project up to 50%

Additional comments

These skills should mostly be reinforced through practical work and laboratory sessions.

1.3.14.4 WBL and Follow-up Reinforcement

After learning the basic principles, terminology, and models of software design, the study should focus on analysing and simulating real work-life-like tasks as, for example:

- Internship tasks of interpreting designs for applications
- Any WBL

1.3.14.5 Important (new) approaches and technologies to consider

- Gaining communication skills

1.3.14.6 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Manages teamwork processes and facilitates collaboration to reach common objectives, e.g., handles conflicts, negotiates, motivates, and persuades	Practical assignment	Assessment of skills

1.3.15 Learning Resources - PLO 8. Soft competences [EQF6]

LEARNING UNIT	EQF	Duration	Didactical Approach	ASSESSMENT	Title of the Learning material	Delivery method of the learning material	Quick link to learning materials
<i>Web Development Frontend</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assignment, Presentation	N/A	Practical assignment, laboratory sessions, lectures	*there is no specific material on soft competences. This should be reinforced through WBL, laboratory sessions, and lectures If possible
<i>Web Development Backend</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assignment, Presentation	N/A	Practical assignment, laboratory sessions, lectures	*there is no specific material on soft competences. This should be reinforced through WBL, laboratory sessions, and lectures If possible

<i>Containerization</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assignment, Presentation	N/A	Practical assignment, laboratory sessions, lectures	*there is no specific material on soft competences. This should be reinforced through WBL, laboratory sessions, and lectures If possible
<i>Virtualization</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assignment, Presentation	N/A	Practical assignment, laboratory sessions, lectures	*there is no specific material on soft competences. This should be reinforced through WBL, laboratory sessions, and lectures If possible
<i>Orchestration</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1	Lectures, Laboratory sessions, problem-based learning,	Exam, Practical assignment, Presentation	N/A	Practical assignment, laboratory sessions, lectures	*there is no specific material on soft competences. This should be reinforced through WBL,

		weeks of individual student work on practical assignment, 1 week exam and presentations	collaborative-based learning, WBL.				laboratory sessions, and lectures If possible
<i>Software Testing</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assignment, Presentation	N/A	Practical assignment, laboratory sessions, lectures	*there is no specific material on soft competences. This should be reinforced through WBL, laboratory sessions, and lectures If possible
<i>Project Management</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assignment, Presentation	N/A	Practical assignment, laboratory sessions, lectures	*there is no specific material on soft competences. This should be reinforced through WBL, laboratory sessions, and lectures If possible

week exam
and
presentations

1.3.16 PLO 9. Functioning in organisations [EQF6]

9. PLO Functioning in organisations [EQF6]

The learner has demonstrated capability

→ to function in an organisational context

Unit learning outcomes	Explains the basics of organisation theory and behaviour
	Describes the relationship between business and IT
	Works in an organisational context under specific direction with limited autonomy and responsibility e.g., at the level of a trainee, junior or assistant
	Manages a project, selects appropriate project management methods and tools
	Writes a report on functioning in organisation

1.3.16.1 Duration of Study

Recommended duration: starting from 2 ECTS

Often integrated with studies of PLOs: 1, 2, 3, 4, 5, 6, 7, 8

1.3.16.2 Recommendations for Micro-credentials

- We recommend this PLO for all students in all courses. Particularly, this PLO should be realized through practical work and laboratory sessions.

1.3.16.3 Recommendations on Didactical Approach, Delivery Methods and Training Environment

Recommended didactical approach:

- Presence Classroom
- Blended
- Work placement

Additional comments

n/a

Recommended delivery methods:

- Lecture and laboratory sessions up to 50%
- Case study. Individual/team project up to 50%

Additional comments

These skills should mostly be reinforced through practical work and laboratory sessions.

1.3.16.4 WBL and Follow-up Reinforcement

After learning the basic principles, terminology, and models of software design, the study should focus on analysing and simulating real work-life-like tasks as, for example:

- Internship tasks of interpreting designs for applications

1.3.16.5 Important (new) approaches and technologies to consider

- Internship tasks of interpreting designs for applications
- Any WBL

1.3.16.6 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Explains the basics of organisation theory and behaviour	Practical assignment	Assessment of skills

1.3.17 Learning Resources - PLO 9. Functioning in organisation [EQF6]

LEARNING UNIT	EQF	Duration	Didactical Approach	ASSESSMENT	Title of the Learning material	Delivery method of the learning material	Quick link to learning materials
<i>Web Development Frontend</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assignment, Presentation	N/A	Practical assignment, laboratory sessions, lectures	*there is no specific material on functioning in organizations. This should be reinforced through WBL, laboratory sessions, and lectures If possible
<i>Web Development Backend</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assignment, Presentation	N/A	Practical assignment, laboratory sessions, lectures	*there is no specific material on functioning in organizations. This should be reinforced through WBL, laboratory sessions, and lectures If possible
<i>Virtualization</i>	6	3 weeks altogether: 1 week of	Lectures, Laboratory sessions,	Exam, Practical assignment, Presentation	N/A	Practical assignment, laboratory sessions, lectures	*there is no specific material on functioning in organizations. This

		lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	problem-based learning, collaborative-based learning, WBL.				should be reinforced through WBL, laboratory sessions, and lectures If possible
<i>Orchestration</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assignment, Presentation	N/A	Practical assignment, laboratory sessions, lectures	*there is no specific material on functioning in organizations. This should be reinforced through WBL, laboratory sessions, and lectures If possible
<i>Containerization</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assignment, Presentation	N/A	Practical assignment, laboratory sessions, lectures	*there is no specific material on functioning in organizations. This should be reinforced through WBL, laboratory sessions, and lectures If possible

		and presentations					
<i>Software Testing</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assignment, Presentation		Practical assignment, laboratory sessions, lectures	*there is no specific material on functioning in organizations. This should be reinforced through WBL, laboratory sessions, and lectures If possible
<i>Project Management</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assignment, Presentation	N/A	Practical assignment, laboratory sessions, lectures	*there is no specific material on functioning in organizations. This should be reinforced through WBL, laboratory sessions, and lectures If possible

1.3.18 EXTRA CURRICULAR PLO: New Technology [EQF6]

PLO New technology [EQF6]

The learner has demonstrated capability

→ to work with IoT, containerized AI, digital twins, blockchain, Cloud-to-Edge computing, quantum computing, and other upcoming technologies

Unit learning outcomes	Explains the principles, related concepts, advantages and disadvantages of a new technology
	Applies methods, techniques and tools related to a new technology
	Writes a report on a new technology or of a method, technique or tool related to it

1.3.18.1 Duration of Study

Recommended duration: starting from 2 ECTS

Often integrated with studies of PLOs: 8, 9

1.3.18.2 Recommendations for Micro-credentials

- Recommended as an independent micro-credential for upskilling junior developers and junior DevOps (for example EQF 4, 5).

1.3.18.3 Recommendations on Didactical Approach, Delivery Methods and Training Environment

Recommended didactical approach:

- Presence Classroom
- Blended
- Work placement

Additional comments

n/a

Recommended delivery methods:

- Lectures and laboratory sessions up to 50%
- Practical assignment up to 50%

Additional comments

Lectures, e-learning are recommended for learning the basic principles, terminology, and models of software design. These should be reinforced through practical tasks, case studies, individual/team-projects in the same amount (time and ECTS-wise) as theoretical work. Lectures should be supported by laboratory sessions.

1.3.18.4 WBL and Follow-up Reinforcement

After learning the basic principles, terminology, and models of software design, the study should focus on analysing and simulating real work-life-like tasks as, for example:

- Each participant chooses a custom domain (from work or life) and create an application that can later be used in real life
- Internship tasks of interpreting designs for applications

1.3.18.5 Important (new) approaches and technologies to consider

- Getting to know new technologies
- Learning about new approaches
- Learning about tools related to new technologies

1.3.18.6 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Explains the principles, related concepts, advantages and disadvantages of a new technology	Exam	Assessment of skills
Applies methods, techniques and tools related to a new technology	Practical assignment	Assessment of skills
Writes a report on a new technology or of a method, technique or tool related to it	Presentation	Assessment of skills

1.3.19 Learning Resources - EXTRA CURRICULAR PLO: New Technology [EQF6]

LEARNING UNIT	EQF	Duration	Didactical Approach	ASSESSMENT	Title of the Learning material	Delivery method of the learning material	Quick link to learning materials
<i>New technologies</i>	6	3 weeks altogether: 1 week of lectures and lab work, 1 weeks of individual student work on practical assignment, 1 week exam and presentations	Lectures, Laboratory sessions, problem-based learning, collaborative-based learning, WBL.	Exam, Practical assignment, Presentation	-	Lectures, laboratory sessions, practical assignment	-

www.softwareskills.eu



Co-funded by the
Erasmus+ Programme
of the European Union

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

www.softwareskills.eu



Co-funded by the
Erasmus+ Programme
of the European Union

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

www.softwareskills.eu



Co-funded by the
Erasmus+ Programme
of the European Union

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

