European Software
Skills Alliance.

ESSA Learning Programmes

ANNEX VI
Solution Designer
EQF 6

30 November 2023

Status: Final version

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

ESSA Learning programme – Solution Designer EQF 6, 2024

Deliverable 10 – ESSA Learning Programmes & Materials – ANNEX VI

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), Niels Sellings (Digital Europe), Ants Slid (BCS Koolitus), Jutta Breyer (Digital Europe), Paul Aertsen (Global Knowledge)

softwareskills.eu

Legal Disclaimer

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

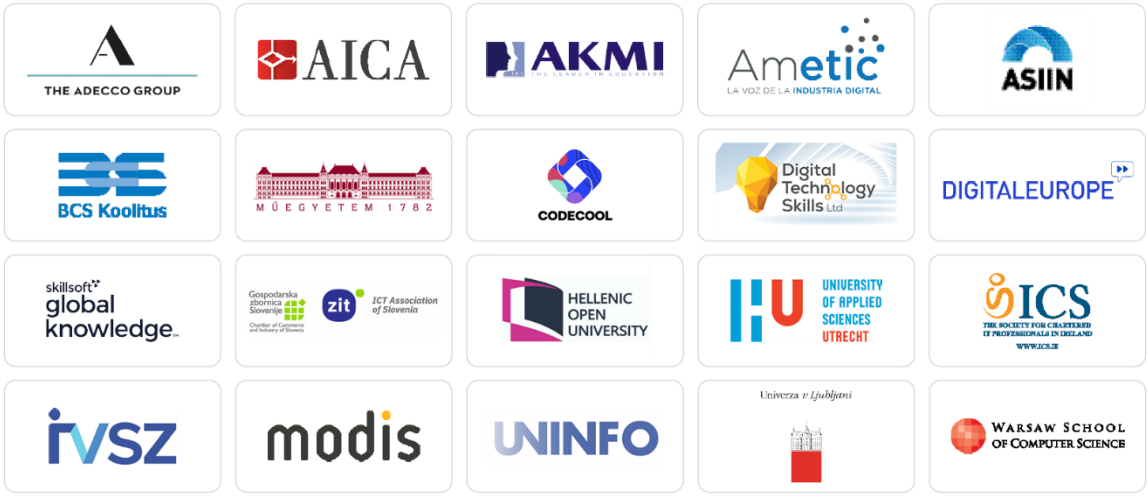

Table of Contents

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 Solution Designer EQF 6 – ESSA Learning Programme

1.1 IT-oriented students

Executive summary

This Learning Programme is being designed by University of Applied Sciences Utrecht (HU). 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.

1.1.1 PLO 2. Architecture Design [e-3]

2. PLO Architecture Design [e-3]	
The learner has demonstrated capability *→ to identify and align relevant ICT technology and specifications*	
Unit learning outcomes	Describes architecture frameworks and standards such as TOGAF
	Explains system architecture requirements (e.g., performance, maintainability, extendibility, scalability, availability, security, accessibility)
	Aligns an IT solution with a certain architecture and formulates (relevant parts of) an IT architecture design, for a relatively straightforward situation applying common design techniques and tools

1.1.1.1 Duration of Study

Recommended duration: 5 ECTS

Often integrated with studies of PLOs: PLO 3

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 first 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)

Describes architecture frameworks and standards such as TOGAF	Practical assessment & Portfolio	n/a
Explains system architecture requirements (e.g., performance, maintainability, extendibility, scalability, availability, security, accessibility)	Practical assessment & Portfolio	
Aligns an IT solution with a certain architecture and formulates (relevant parts of) an IT architecture design, for a relatively straightforward situation applying common design techniques and tools	Practical assessment & Portfolio	n/a

Continuous feedback is given on the learning and creation process by other students, senior students, teachers in the role of coach and experts from the field. This takes place during the planning of the sprint, the execution of the work, the peer review of products, the delivery to the client, coaching sessions and knowledge sharing. We have continuous contact with the student from within the program and during the final assessment that takes place every ten weeks. As a result, we know exactly how the student is doing.

In the final assessment, we look at the complete development of the student. We mainly ask ourselves whether the student is ready for the next phase. The complexity of projects increases every six months and students must be able to successfully fulfil their own role in a team more independently. Together with the development that the student has gone through in his general and substantive skills, we make a decision whether the student is allowed to continue to the next phase.

1.1.2 Learning Resources - PLO 2. Architecture Design [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
MaO - 01 Introductie & BPMN	6	1 hour and 30 minutes	Live classes	Practical assessment & Portfolio	BPMN part 1	Lecture and practical exercises	https://learn.softwareskills.eu/wp-content/uploads/2024/01/MaO-01-Introductie-BPMN-ENG.pptx
MaO - 02 BPMN deel 2	6	1 hour and 30 minutes	Live classes	Practical assessment & Portfolio	BPMN part 2	Lecture and practical exercises	https://learn.softwareskills.eu/wp-content/uploads/2024/01/MaO-02-BPMN-deel-2-ENG.pptx
MaO - 03 Feedback BPMN	6	45 minutes	Live classes	Practical assessment & Portfolio	Feedback BPMN	Lecture and practical exercises	https://learn.softwareskills.eu/wp-content/uploads/2024/01/MaO-03-Feedback-BPMN-ENG.pptx
MaO - 04 UML & Use case descriptions	6	1 hour 30 minutes	Live classes	Practical assessment & Portfolio	UML & Use case descriptions	Lecture and practical exercises	https://learn.softwareskills.eu/wp-content/uploads/2024/01/MaO-04-UML-Use-case-descriptions-ENG.pptx

1.1.3 PLO 3. Application Design [e-3]

3. PLO Application Design [e-3]	
The learner has demonstrated capability *→ to specify a design for a software application or component that meets requirements* *→ to organize the planning of the design of an application or software component*	
Unit learning outcomes	Explains and distinguishes principles and terminology of software design (e.g., phases in the design process, techniques, deliverables)
	Describes principles of usability, UI/UX design, accessibility, privacy, security
	Creates functional and data modelling diagrams, using common languages and techniques (e.g., DFD, IDEF0, ERD, and UML)
	Designs a simple system architecture and interfaces using familiar technologies
	Compares alternatives for a design and selects the most promising alternative(s), optimising the balance between cost and quality
	Specifies a design for a software application or component, taking into account certain constraints/ requirements (e.g., the development environment, programming language, technology, requirements related to performance, security, accessibility, usability, privacy, ethics, safety, IS policy, cost, quality)
	Designs and organises the overall plan for the design of an application or software component

1.1.3.1 Duration of Study

Recommended duration: around 5 ECTS

Often integrated with studies of PLOs: PLOs 2

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 first year.

1.1.3.3 Recommendations on Didactical Approach, Delivery Methods and Training Environment

Recommended didactical approach:

- Presence Classroom ☒
- Work placement ☒

Additional comments

n/a

Recommended delivery methods:

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

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)
Explains and distinguishes principles and terminology of software design (e.g., phases in the design process, techniques, deliverables)	Practical assessment & Portfolio	n/a-
Describes principles of usability, UI/UX design, accessibility, privacy, security	Practical assessment & Portfolio	n/a-
Creates functional and data modelling diagrams, using common languages and techniques (e.g., DFD, IDEF0, ERD, and UML)	Practical assessment & Portfolio	n/a-

Designs a simple system architecture and interfaces using familiar technologies	Practical assessment & Portfolio	n/a-
Compares alternatives for a design and selects the most promising alternative(s), optimising the balance between cost and quality	Practical assessment & Portfolio	n/a-
Specifies a design for a software application or component, taking into account certain constraints/ requirements (e.g., the development environment, programming language, technology, requirements related to performance, security, accessibility, usability, privacy, ethics, safety, IS policy, cost, quality)	Practical assessment & Portfolio	n/a-
Designs and organises the overall plan for the design of an application or software component	Practical assessment & Portfolio	n/a-

1.1.4 Learning Resources - PLO 3. Application Design [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
MaO - 01 Introductie & BPMN	6	1 hour and 30 minutes	Live classes	Practical assessment & Portfolio	BPMN part 1	Lecture and practical exercises	https://learn.softwareskills.eu/wp-content/uploads/2024/01/MaO-01-Introductie-BPMN-ENG.pptx
MaO - 02 BPMN deel 2	6	1 hour and 30 minutes	Live classes	Practical assessment & Portfolio	BPMN part 2	Lecture and practical exercises	https://learn.softwareskills.eu/wp-content/uploads/2024/01/MaO-02-BPMN-deel-2-ENG.pptx
MaO - 03 Feedback BPMN	6	45 minutes	Live classes	Practical assessment & Portfolio	Feedback BPMN	Lecture and practical exercises	https://learn.softwareskills.eu/wp-content/uploads/2024/01/MaO-03-Feedback-BPMN-ENG.pptx
MaO - 03 requirements & use cases afleiden	6	45 minutes	Live classes	Practical assessment & Portfolio	Requirements & Use case diagram	Lecture and practical exercises	https://learn.softwareskills.eu/wp-content/uploads/2024/01/MaO-03-Feedback-BPMN-ENG-1.pptx
MaO - 04 UML & Use case descriptions	6	1 hour 30 minutes	Live classes	Practical assessment & Portfolio	UML & Use case descriptions	Lecture and practical exercises	https://learn.softwareskills.eu/wp-content/uploads/2024/01/MaO-04-UML-Use-case-descriptions-ENG.pptx

1.1.5 PLO 10. Functioning in organisation [EQF6]

10. 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 the organisation

1.1.5.1 Duration of Study

Recommended duration: 1 ECTs

1.1.5.2 Recommendations for Micro-credentials

- This PLO is a part of a 4 year bachelor programme. It is aimed at students from the first year (HBO-startniveau)

1.1.5.3 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

The module discusses organization and management aspects that are relevant for IT professionals to understand their environment and organization. The student can describe the current business model of the organization, analyze it and based on it. the conclusions of this analysis provide advice on a possible future business model for the organization and more specifically for the (business) ICT domain.
Flow of the sessions:

- Preparation: studying literature and creating assignment meeting (upload reflection / case study results of assignment (ppt format) via Canvas)
- Plenary introduction by teachers (depending on content 1 or 2 hours)
- Break (15')

- Collecting feedback on homework in learning teams
- Presentations and feedback from fellow students and teacher
- Questions / theory
- Assignment for the next session

1.1.5.4 WBL and Follow-up Reinforcement

- During the 4 sessions, there is a combination of lecture and practical cases and exercises. Before each session, students must prepare (homework) working on the business case and reading relevant literature

1.1.5.5 Important (new) approaches and technologies to consider

n/a

1.1.5.6 Assessment

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
Explains the basics of organisation theory and behaviour	Professional product	n/a
Describes the relationship between business and IT	Professional product	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	Professional product	n/a
Manages a project, selects appropriate project management methods and tools	Professional product	n/a
Writes a report on functioning in the organisation	Professional product	n/a

Professional product: Description and analysis of, preferably, your own organization and its environment, translated into the design of IT based on similar elements of the CANVAS model and associated theories, models and instruments with clear conclusions and recommendations for the customer side and internally business model and a substantiated vision of a possible future, alternative business fashion model culminating in a discussion paper, i.e. advice for the entire organization.

The student follows the work cycle for practice-oriented research (Verhoeven, 2010)

1. Problem analysis: preliminary investigation and determining the problem definition, question, objective and definition

2. Research design: choice of the research strategy and the research methods

3. Data collection: collecting data /information you need to make your answer research questions

4. Data analysis: analyzing the data obtained data / information to draw conclusions

5. Conclusions & recommendations: providing an answer (conclusion) to the question basis of the data analysis and the interim conclusions and passing on recommendations to the client

6. Reporting and presentation: writing it research report and giving a presentation to stakeholders.

1.1.6 Learning Resources - PLO 10. 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
External and internal environment organizations	6	6 hours	Live classes	Professional product	External and internal environment organizations	Lecture and practical exercises	10 PLO Functioning in organisation
Strategy - SWOT, BCG, T&W and Ansoff	6	6 hours	Live classes	Professional product	*Strategy - SWOT, BCG, T&W and Ansoff Article: business model navigator)*	Lecture and practical exercises	10 PLO Functioning in organisation
Business model canvas	6	6 hours	Live classes	Professional product	*Business model canvas*	Lecture and practical exercises	10 PLO Functioning in organisation
Business model patterns & Blue Ocean strategy	6	6 hours	Live classes	Professional product	*business model patterns & Blue Ocean strategy*	Lecture and practical exercises	10 PLO Functioning in organisation
Extra - Modeling and orientation							
Modeling and Orientation - 03 requirements & use cases afleiden	6	1 hour and 30 minutes	Live classes		Requirements & Use case diagram	Lecture and practical exercises	10 PLO Functioning in organisation
Modeling and Orientation - 06 Business Rules & UI Design	6	1 hour and 30 minutes	Live classes		Business rules & UI Design	Lecture and practical exercises	10 PLO Functioning in organisation

www.softwareskills.eu

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