European Software
Skills Alliance.

Work-based Learning

DELIVERABLE 12 – ESSA Work-based Learning

30 November 2023

Status: Final version

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

Work-based Learning, 2024, Final version.

Deliverable 12: "ESSA Work-based Learning"

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

Authors: Merje Vaide (BCS Koolitus), Ants Sild (BCS Koolitus), Regina Mathisen (BCS Koolitus),

Editors/Reviewers: Guido Ongena (HU), Niels Selling (DE), Mateja Pucihar Baebler (CCIS)

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

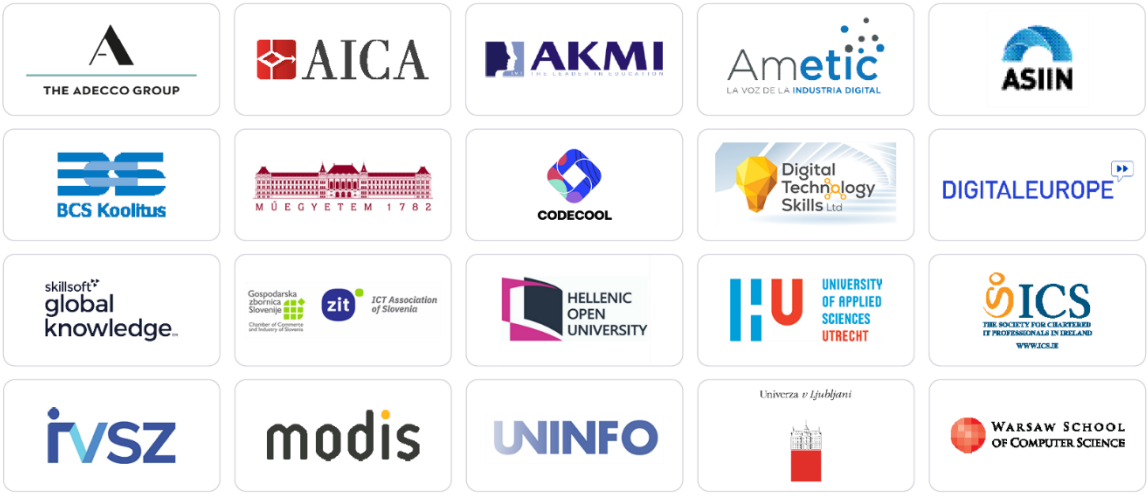

Table of Contents

List of Tables

List of Figures

List of abbreviations and acronyms

Abbreviation	Term
ESSA	European Software Skills Alliance
VET	Vocational Education and Training
PLO	Programme Learning Outcome
ULO	Unit Learning Outcome
WBL	Work-based Learning
Adecco	Adecco Formazione
BCS	BCS Koolitus AS, Estonia
GK	Global Knowledge France
HOU	Hellenic Open University
HU	University of Applied Sciences Utrecht
UL	University of Ljubljana
WSCS	Warsaw School of Computer Science

1 Executive Summary

1.1 Introduction

This report "ESSA Work-Based Learning" (WBL) presents the set of educational activities designed and organized to achieve the ESSA learning objectives and accomplish a specific set of tasks. Work-based learning has been designed and adopted into the ESSA learning programmes to ensure the acquisition of knowledge and skills through carrying out – and reflecting on – tasks in a vocational context, either at the workplace or in a learning institution.

The document has been produced as a deliverable related to Work Package 4 "Learning Programme, Testing & Rollout" of the European Software Skills Alliance (ESSA) project. It refers, specifically, to task 4.3 "Work-based Learning & Continuous Professional Development".

The contents provided are focused on the presentation of the ESSA work-based learning and related learning materials designed to be integrated into all ESSA curricula and learning programmes.

Work-based Learning (WBL) means a continuum of awareness, exploration, preparation, and training activities that combine structured learning and authentic work experiences implemented through industry and education partnerships. The primary purpose of WBL is to provide learners with learning experiences that integrate knowledge and theory with practical real-life work activities. Work-based learning is playing an important role in improving learners' professional development and learning. As the nature of work and careers changes, work-based learning can prepare students to engage in active learning both at work and in the classroom and develop new skills throughout their careers. WBL is a fundamental part of the curricula.

The "ESSA Work-based Learning" defines the term WBL in ESSA project as an integrated learning, that is focused on defining and solving real life tasks and practical work-based problems. WBL is an important part of ESSA learning materials because it is equipping learners with the practical skills and industry insights, necessary for a successful career in this constantly evolving field of software development.

This document describes the primary types of WBL, elucidating the way these components are described and integrated into the ESSA learning programmes at EQF levels 4/5, 6 and 7 (Task 4.3). It categorises WBL components according to ESSA profiles, offering an overview of the most frequently used ones. Furthermore, the document provides recommendations on designing WBL components and elucidates the best practises to follow in ESSA learning programmes.

Contents and information made available are structured as follows. Chapter 2, and 3 explain the ESSA approach, context, and methodology in dealing with work-based learning components as integral parts of each Learning Programme and curriculum.

Chapters 4, and 5 present the analysis of the WBL components developed and piloted in ESSA. Recommendations, and conclusions, what are the elements of the WBL components that deliver the best learning outcomes are presented.

In Annexes the exemplary WBL components are presented. Each Annex presents in detail one ESSA Work-based Learning component and is structured as follows:

- WBL Title. Learning Profile. Learning Provider /ESSA Partner
- Related Learning Units
- Description. Learning Outcomes
- Expected Output Format
- Time Estimation
- Recommended To-Do List
- Recommended Work Environment, Hardware, and Software List
- Assessment Description and Criteria
- Recommendations and Technical Instruction for the Teacher

Up to the purpose, this document must be considered in its entirety - the Annexes are a full part of the document.

1.2 Objective

By describing and analysing the best practices of ESSA partners in developing and using WBL components for ESSA Programmes, we can provide learning providers and learners with a better understanding and support on how to implement WBL to reach the best learning outcomes. From the recommendations and presented examples interested parties can select and use the components best fit for their needs.

The purpose of this document is to support:

- The piloting of the learning programmes as defined under task 4.2 “Pilot the ESSA learning programme” how to set up and integrate WBL components into the learning programme.
- The evaluation and reviewing of the ESSA Learning Programme and ESSA material proposed for each Educational Profile (task 4.1.3).
- Associated Partners how to use WBL in running the soft pilots of the ESSA learning programmes.
- All learning providers how to develop and use WBL components for their learning programmes.
- General audience to consult, use and transfer the contents according to their specific needs.

1.3 Methodological Approach

Effectively aligning learning programmes with workplace needs remains the most critical challenge faced by learning providers in today's dynamic business landscape. ESSA determines Work-based Learning as a powerful tool for effectively aligning learning

programmes with workplace needs because it provides students with the opportunity to apply their knowledge and skills in a real-world setting. The methodological question for ESSA thus was not if and when to introduce WBL in ESSA Learning Programmes, but how to support learning providers to include as much WBL as possible and what are the best methods and practices of introducing WBL components.
ESSA started with defining and providing a comprehensive understanding of Work-Based Learning for our programmes. The ESSA focus is on developing Learning Programmes that best and most effectively address the desired learning outcomes. Out of the types of WBL (see Chapter 2.2), analysis showed that integrated learning is the type of WBL, that provides the most suitable opportunities and tools for such programmes as ESSA.
To get a detailed understanding of the practices of using WBL components by ESSA partners, a WBL template was developed and introduced. Each partner selected the most important and productive best-practice WBL examples they have developed and used and described each of these examples using the WBL template.
The collected WBL components' descriptions were analysed and systemised by different aspects (WBL type, learning outcomes, learning profiles). The criteria to select the best-of-breed examples of WBL components were defined and recommendations were developed for using WBL in learning programmes.
The methodological approach and work process is explained in more detail in Chapter 3.

1.4 Results

The results of the ESSA Task 4.3. "Design Work-based Learning Component" are the following:

- Work-based Learning is an indispensable component of ESSA Software Strategy and of each learning programme defined and described. As such an irreplaceable tool, WBL is recognised and acknowledged by all ESSA partners.
- A comprehensive amount (26) of best-practice WBL examples from ESSA learning programmes are described, analysed, and systemised for use by learning providers in developing the WBL components for their learning programme descriptions.
- Out of the 26 selected WBL examples, 5 are presented in detail in the Annexes 1-5 of this report. These can be used by learning providers to support them in developing WBLs for their learning profile, needed learning outcomes, and type of WBL needed. Descriptions of all the 26 WBL examples are available in the full documentation of ESSA Work Package 4.
- Analysis of what are the criteria of WBL components delivering the best learning outcomes (see Chapter 5.1).
- Recommendations on how to develop, implement and use WBL components in software skills learning programmes (see Chapter 5.2).

1.5 Conclusions

The defined ESSA work-based learning components, recommendations and examples serve as a foundation for the design and use of WBL components of curricula, and effective learning programmes for skilling, upskilling, and reskilling people into in-demand software roles.

The described WBL components have been used and tested in ESSA pilots. The overviews of example WBL materials serve to support learning providers in developing comprehensive WBL components for any software skills curriculum and learning programme.

The main conclusions and takeaways will be amended based on analysis of the results of the work during the following period.

1.6 Use of This Document

This document furnishes an overview of the Work-Based Learning (WBL) components within the ESSA learning programme. It further offers recommendations on the establishment of WBL components, thereby elucidating the optimal practices for integrating practical, hands-on learning experiences into the learning programme.

The document will also support the European Mobility Programme (Task 6.2) component of the project in facilitating the identification of contents and learning activities that might be part of a transferrable element.

This document can be adopted by trainers, teachers, education providers and the key actors of the sector to obtain content and information on the practices of employing work-based learning in different learning programmes for software roles at different EQF levels (4-7). This document gives learning providers tools to identify and develop WBL components best suitable for their needs and objectives.

2 Context of Work-based Learning within ESSA

2.1 Work-based Learning Definition

ESSA views Work-based learning (WBL) as a critical component of all learning programmes and as an indispensable tool for aligning learning programmes with workplace needs.

Work-based learning (WBL) means a continuum of awareness, exploration, preparation, and training activities that combine structured learning and authentic work experiences implemented through industry and education partnerships. WBL is imperative as it contributes to 1) organisational productivity and innovation, 2) value as a form of learning, 3) value in improving youth transitions, and 4) in career development (Sweet, 2013). Work-based learning is often undertaken in conjunction with classroom or related learning and may take the form of work placements, work experience, workplace mentoring, instruction in general workplace competences, and broad instruction in all aspects of the industry.

The work-based learning component in the ESSA context means mostly an integrated training model i.e., practical workplace-related tasks and assignments as an integral part of the learning process. The goal is to provide practical work experience to support theoretical studies. It should also provide wider opportunities for personal and professional development.

2.2 Types of WBL

Existing research and overviews of implementing practices of work-based learning define three main types of work-based learning. The most widely used is **integrated learning**, where integral components of the curriculum and learning programme are designed to focus on defining and solving real-life tasks and practical work-based problems. It can take place in the working environment or at a training institution. It is learning through work context problem-solving. Such learning takes place through tasks usually defined by industry practitioners and can use a large variety of setups like real business/industry problem-solving tasks and project assignments, workshops, workplace simulations, project-based learning, student-led enterprises etc. The time and the amount of work needed may vary, starting from quick tasks, e.g. brainstorming in a classroom during a lecture, to a multi-month onsite industry project.

The second type of WBL is **on-the-job training or internships in companies.** On-the-job training or internship typically covers an internship at workplaces as a compulsory or optional element of study programmes. It provides students with knowledge and skills that they need in real work life (including soft skills). The internship must have clearly defined learning tasks and learning outcomes. It also has an important role in students' professional development with the important part of students' reflection on work-based learning experience, professional and self-development. The success of an internship is always based on a good relationship of three parties: school-student-company. During the internship, students need guidance and support from both, the school, and the company. Especially important are the skills of the supervisor in the company. Special training and support from the training institutions are usually needed for the supervisors to achieve the best results. For the training institutions, the critical task here is the partnership relations management with companies and the supervisors.

While an internship is a short-term work experience focused on defined tasks and outcomes, longer periods of work during which the goal is to experience all aspects of a working position, form the third type of WBL – **apprenticeships**. In apprenticeships, the learning is guided mostly by the employer and is less (or not) integrated with the learning programme at an educational institution. The apprenticeship period should provide the learner (apprentice) with certainty about the occupation's suitability for him/her and give the employer a clearer understanding of the apprentice's capabilities to perform the job as needed.

In ESSA learning programmes the focus is to design learning components that would address the desired learning outcomes most directly and effectively. For that goal, the first WBL type — integrated learning — offers the best opportunities. For that reason, most of the examples and recommendations deal with integrated learning. Internships and shorter on-the-job training periods fit also well with ESSA programmes and can be successfully integrated.

Apprenticeships follow the structure and logic of a working position and as a rule, do not need prior design of the learning process. For that reason, ESSA does not focus on analysing and planning apprenticeship programmes. Nevertheless, apprenticeship programmes can

be successfully used as an extension of ESSA programmes after successful completion of the curriculum.

ESSA learning programme modules contain WBL components —- work-based learning tasks (real-life-problem-centred tasks, -projects, -assignments), their assessment methodology and general guidelines how WBL components are integrated into the curricula.

Integrated learning WBL components can also be differentiated according to the complexity, length of the task, and involvement of business representatives. The shorter and most simple ones are **practical exercises and tasks** simulating real-life work scenarios, integrated into the learning process at learning institution (without direct involvement of business representatives as customers). These can start from simple exemplary exercises simulating real-life tasks focusing on one specific learning outcome. But they can be complex projects to be completed during the learning process at the learning institution. More complex are **business/industry defined projects** where the task and needed learning outcome is defined by a business representative. Those projects can be integral parts of the learning programme to address specific learning, where gaining first-hand business experience is the main task. Those projects can be led by students themselves to obtain learning outcomes of real-life project work.

Internships, as more complex and long-lasting assignments address a wider set of learning outcomes. For that format, the learning programme defines the learning outcomes to be tackled but does not prescribe the scope and content of the work tasks or projects to reach these outcomes. For that reason, internships do not need as much design and development work from the learning providers. That is also why most examples in this report focus on integrated learning. For designing internship programmes, the examples of integrated learning can be combined and used.

2.3 WBL and Scenario Based Learning

Scenario-based learning stands as an engaged and participatory pedagogical strategy, entailing the presentation of students with a scenario or contextual situation necessitating resolution. In this learning approach, learners are tasked with navigating through a carefully constructed narrative, thereby stimulating their capacities for problem-solving and critical thinking. These scenarios are underpinned by authentic, real-world problems or situations, imbuing the learning experience with practical relevance.

The majority of WBL components prominently feature elements of **scenario-based learning**. These WBL components typically present an initial task within a contextual scenario, alongside a set of tools to facilitate the solution process. Learners, thus, are tasked with the discerning responsibility of selecting the most suitable techniques to resolve the presented challenge. In certain WBL instances, while the instructions may offer a detailed, step-by-step framework for the exercise, they are inherently designed to empower the learner to configure a solution that aligns with their individual approach.

Main elements (5C-s) of scenario-based learning https://www.thinglink.com/blog/what-is-scenario-based-learning/:

- Context: Setting up the initial situation/environment for learners to start with.
- Challenge: A problem or situation set within the context including the learning outcomes and limits.
- Choices: Usually, the scenarios are not definitively prescribed, a lot depends on the context, on what tools and technologies are used, or what are the restrictions.
- Consequences: Results (solutions) may be usable, incomplete, or not cover the requirements of the initial task. The results arise from the choices learners have made and how well these choices correspond to the situation.
- Contemplate/Retrospective: Assess the effectiveness and usability of the solution. What did the learners learn from this experience?

Within this educational framework, learners are entrusted with the autonomy to chart their own trajectory, a course that is fundamentally contingent upon the decisions they make. Embracing a scenario-based learning approach not only captivates learners' interest but also actively involves them in the learning process. Furthermore, it offers expeditious feedback mechanisms that cater to the needs of both educators and learners.

Scenario-based learning extends beyond just learning professional competences, fostering the development of a versatile set of soft skills and is also an important part of professional development.

The integration of scenario-based learning within the educational framework is adaptable, finding application within specific modules as well as permeating the entirety of the curriculum. In the latter instance, the comprehensive curriculum is characterised by a predominant reliance on practical, work-based activities.

In the case of a scenario-based learning curriculum, the whole learning process is designed consisting of scenario-based modules. Such an approach is gaining more and more popularity, especially in upskilling and reskilling programmes.

A scenario-based approach to learning programmes is presented in Annexes 5 and 6. In Annex 5, Developer EQF 7 programme by Warsaw School of Computer Science is presented.

In the example of a full scenario-based learning programme for Junior Software Developer EQF 4/5 by ESSA project partner Codecool, the whole learning happens as working on lifelike tech projects, in Scrum development teams, in a workplace simulation environment. The logic of building such a full scenario-based learning programme is described in Annex 6.

3 Methodology

The work in Work Package 4 started with a careful design of the methodological approach also including the methodology to deal with work-based learning. This was described for ESSA partners in a detailed document “ESSA WP4 Methodology”.

According to the plan, ESSA WP4 team started by defining the term Work-Based Learning. It was a serious discussion comparing the different understandings and developments that exist today in the communities of formal and non-formal education. A clear understanding was the ESSA should focus on the WBL components allowing the best alignment with needed learning outcomes. This did put ESSA's focus on **integrated learning** but did not exclude the other types of WBL.

The next step is gathering examples and data about different WBL components for further systematization and analysis. In May 2023 the team developed the WBL template. To describe the WBLs, each partner had to provide the title of the WBL component, names of related learning modules and topics in the learning programme. The description of the WBL component contains short descriptions of WBL, addressed learning outcomes and expected output format. If needed, there should be guidelines on how the planned task should be fulfilled.

All the partners focused on the curricula they developed under ESSA WP3 and described the WBL components integrated with their curricula. Out of the WBL components developed and used in each learning programme, each partner should select the most important and productive best-practice WBL examples and describe each of these on the WBL template.
The data collection period was from May to August when the partners provided their descriptions of the developed WBL components.

In the analysis of the collected WBL components' descriptions, the WP4.3 team systemised WBL components by different aspects (types, learning outcomes, main profiles). Based on this analysis the team defined the criteria on which recommended examples of WBL components were selected and presented in a unified format. These results are presented in Chapters 4.4 and 5 and in the Annexes.

As one main result of the Work Package 4, exemplary WBL components of best practices were chosen, and recommendations for using WBL components were developed. These recommendations are presented in Chapter 5 and serve as guidelines for learning programme developers in all learning organisations from formal learning to non-formal and even in-formal learning.

The process involved an analysis across different aspects to gain a holistic understanding of using WBL in ESSA learning materials. By examining various elements of WBL descriptions, the WP4.3 team systemised the descriptions based on different aspects (types, profiles, learning outcomes). The results showed the importance to consider multiple aspects or views to analyse the subject thoroughly.

ESSA methodology foresees that the development of learning programmes is followed by piloting them. This allows us to get real-life feedback also on WBL components and recommendations developed.

At this stage of the ESSA project, the piloting is still underway. All the piloting partners have declared that they will use WBL in their learning programmes and pilots. During the

piloting partners' meetings and the following evaluation of piloting, use of WBLs will be discussed and WP4 team responsible for WBL development will give feedback and support to piloting partners. These discussion points will be generalised and the recommendations on using WBL amended as needed.

4 Analysis of the Described WBL Components

4.1 General Overview of the WBL Examples

During the WBL components development and collection period, the partners provided 26 descriptions of WBL components introduced also into their programmes (See Table 1). The number of WBL examples presented were for software developers' profiles (12 WBL component examples), and DevOps (10), for Solution Designers were three WBL components and one for Test Specialist. Several DevOps WBL examples included also testing tasks. Thus, the examples cover the learning profiles and EQF levels planned.

The task of developing, describing and detailly documenting the WBLs was not an easy one, even for experienced learning providers as ESSA consortium. Some of the WBL components are described in more detail, but some are more general. Mainly the common practice is that WBL descriptions are more generic. The advantage of more generic descriptions is, that it makes these examples and best practices easier to use and transfer for similar ESSA profiles. Also, the planned amounts of learning cover a wide range, with WBL components starting from the duration of 2-8 hours.

The results show that while WBL and scenario-based learning approach is more and more discussed and promoted in the education community, there is still a shortage of common and unified understanding and learning practices to implement WBL. While all ESSA partners and programmes use practical work-driven tasks and components, there still are significant opportunities to make these tasks more workplace and business-driven, and transferable between programmes and disciplines. Good WBLs anticipate intense cooperation with industry and quick changes at the same pace as the working environment changes. This is not an easy task for learning providers. Thus, developing a unified approach and recommendations for WBL integration is an important task in making the learning and workplace better aligned, especially in dynamic fields like software development.

The following table presents all the WBL components developed and collected. They are presented by the learning profile they address, and the partner who developed that example. Mainly the examples are practical exercises, mostly hybrid exercises and industry-defined projects and one internship example. In the table, the WBLs are presented starting from the lower EQF levels.

These WBL components encompass a range of deliverables and activities such as software prototypes, web development, project work, mentoring sessions, documentation, reports, continuous delivery planning, background research, design and create documentation, innovation ideas, and personal reflections. They represent a diverse set of components essential in the software development lifecycle and learning process.

In the next chapter, these examples are analysed in more detail by the learning outcomes they address.

Nr	Title	Partner	Description	Output format
Developer EQF4-5				
1.	Application Design	Adecco	Practical exercise	UI/UX Prototype
2.	Application Development	Adecco	Practical exercise	Web application
3.	Creating a simple webpage	BCS	Practical exercise	Webpage
4.	Creating a landing page	BCS	Practical exercise	Webpage
5.	Creating a website for SME	BCS	Practical exercise	Website
6.	Creating a web application	BCS	Practical exercise	Web application
7.	Final Project	BCS	Practical exercise	Software Project
8.	Mentor Programme	DTSL	Internship	Mentoring Sessions, Software Project
9.	Industry Software Project	DTSL	Business/industry defined projects	Software Project
Test Specialist EQF4-5				
10.	Testing	Adecco	Practical exercise	Web application
Developer EQF6				
11.	Banking application	GK	Business/industry defined projects	Software Project
DevOps EQF6				
12.	Containerization	UL	Practical exercise	Deployment code in Git repo
13.	Software Testing	UL	Practical exercise	Test Documentation
14.	Web Development-Backend	UL	Practical exercise	Web application
15.	Web Development-Frontend	UL	Practical exercise	Web application
16.	Orchestration	UL	Practical exercise	Deployment code in Git repo
17.	Virtualization	UL	Practical exercise	Scripts, installation, documentation
18.	Implementing a software factory	GK	Business/industry defined projects	Software Project
Developer EQF7				
19.	Industry Practitioner-Led Sessions	WSCS	Business/industry defined projects	Learning sessions
20.	Implementation Business Industry Project	WSCS	Business/industry defined projects	Software Project
DevOps EQF7				
21.	Component Integration DevOps in a Large-scale Information System	HOU	Research project	Evaluation report, research article/report, case studies

22.	Implementing DevOps in a Large-scale Information System	HOU	Research project	Continuous Delivery Roadmap, feedback mechanism
23.	Testing DevOps in a Large-scale Information System	HOU	Research project	Evaluation report, research article/report, case studies
Solution Designer EQF7				
24.	Cloud-Based Ecommerce Architecture Design	HOU	Research + design project	Design document, presentation
25.	Cloud-Based Ecommerce Solution Design	HOU	Research project + idea generation	Needs identification report, innovation proposal
26.	Cloud-Based Solution Design Technology	HOU	Research project	Report, Personal Critical Reflection

Table 1. List of WBL components by ESSA profiles

4.2 Analysis of the WBL Examples Based on the Addressed Learning Outcomes

The logic behind developing WBL follows the same basic logic followed throughout the ESSA project and European Software Skills Strategy. From the competence needs and frameworks, the Educational Profiles are designed with the needed programme and unit learning outcomes (PLOs and ULOs). From these, the curricula and the learning programme modules and components are developed. So, the input for developing ESSA WBL components directly derives from the previous stages of ESSA work – like ESSA Deliverable 5 "European Software Skills Strategy", Deliverable 10 – "ESSA Learning Programme and Materials" and others.

Integrating work-based learning into a learning programme involves creating opportunities for learners to gain practical experience and apply their knowledge in real work settings. While for developing WBL components the first task is to clearly define the learning objectives – specific skills, competences, and knowledge that learners should gain through work-based learning experiences. The work-based learning tasks should be well aligned with the PLOs of the curriculum. The tasks should support diverse learning styles and complement and reinforce other learning activities to achieve learning outcomes. Effective WBL should encourage learners to reflect on their own experiences, decisions, and outcomes.

In this chapter, the components of WBL are presented according to the ESSA profiles and by PLOs of the ESSA learning programme they address. This classification manifests a graduated spectrum, commencing with single-PLO-centric WBL initiatives and culminating in multifaceted undertakings that comprehensively engage with nearly all PLOs. Such a structured analysis empowers ESSA partners and prospective users with a toolset to discern and develop their most suitable WBL components.

In the following section the WBL components are presented in the order of EQF levels and ESSA profiles they address. The presentation of the learning outcomes of the components allows the reader to determine the WBL examples most appropriate for further study for the learning modules and learning outcomes of their particular interest. The detailed descriptions of all ESSA 26 WBL examples collected are presented in ESSA full documentation.

4.2.1 Junior Developer EQF4-5:

Most of the WBL component examples for this profile address the technical PLOs related directly to the domain of software development. This emphasises the main goal of the learning programmes at that EQF level, which is developing the most important basic software development competences.

Table 2 is the list of smaller-scale WBL tasks. These examples are related directly to web development and address two fundamental learning outcomes (PLOs): Application Design and Application Development.

WBL components	Learning Outcomes
Application Development (Adecco) (Table 1, example 2) **Creating a simple webpage** (BCS) (Table 1, example 3) **Creating a landing page** (BCS) (Table 1, example 4) **Creating a website for SME** (BCS) (Table 1, example 5) **Creating a web application** (BCS) (Table 1, example 6)	1. PLO Application design 2. PLO Application Development

Table 2. WBL examples with related learning outcomes for Junior Developer EQF4-5

Examples 7, 8, and 9 represent software projects, and they all cover almost all the PLOs of the learning programme. This strategic approach ensures that learners can acquire a wide range of knowledge and skills and they are more prepared for the dynamic and collaborative nature of the software development industry. List of the project-based learning WBL components:

- **Final project** (Table 1, example 7) (BCS). Learners must prepare a final project to create a web application for a small business using basic technologies (HTML, CSS, JavaScript). The use of PHP and MySQL is recommended.
- **Industry Software project** (Table 1, example 9) (DTSL). Each pod (team) creates the design and develops the software application. At the end of the six weeks, they present their project to the panel and are asked questions on their technical approach, how they organised as a team, how they agreed and assigned tasks and how well they worked together in this practical and reflective presentation.
- **Mentor Programme** (Table 1, example 8) (DTSL). Learners must participate in mentoring sessions, and they get support for an industry-related software development project.

4.2.2 Test Specialist EQF4/5

For Test Specialist on this EQF level, in the WBL example "**Testing**" (Table 1 example 10) by Adecco Formazione, the learners must perform a software test using the ISTQB® certified tester framework. This example addresses two main PLOs (1. PLO Component Integration, 2. PLO Testing). The task allows participants to understand the main procedures of software testing, using practical examples, such as viewing and analysing software and programming code.

4.2.3 Developer EQF6

The WBL example "**Banking Application**" (Table 1, example 11) by Global Knowledge France is a description of a team project. In this task the learners work in a team to setup a CI/CD Pipeline and integrate several tools into that pipeline. They must organise themselves in a project and work according to the agile methodology, performing under the same conditions as in real-life.

The WBL component is small scaled but addresses most of the PLOs (see Table 3).

WBL components	Learning Outcomes
Banking Application (Table 1, example 11)	1. PLO Application Design 2. PLO Application Development 3. PLO Component Integration 4. PLO Testing 5. PLO Documentation Production 6. PLO Problem Management 7. PLO Profession related competences 8. PLO Soft competences – interpersonal and personal 9. PLO Functioning in an organisation.

Table 3. WBL example with related learning outcomes for Developer EQF6

4.2.4 Developer EQF7

For Developer EQF 7 the described WBL components are **"Industry Practitioner-Led Sessions"** (Table 1, example 19) and "**Implementation Business Industry Project"** (Table 1, example 20) by Warsaw School of Computer Science. In the "Industry Practitioner-Led Sessions", students will benefit from the direct knowledge transfer from seasoned professionals, enhancing their understanding of Full stack Development and bridging the gap between academic concepts and practical industry applications. In the other WBL component students will be engaged in projects conducted in collaboration with IT companies. Their task will involve working on specific application or system projects, from design to deployment. They will collaborate in teams, analyse client requirements, design system architecture, and implement both frontend and backend components. Throughout the project, students will need to consider real-world constraints and business challenges.

In these WBL components systematically a substantial majority of the PLOs are engaged. This comprehensive alignment underscores the concerted effort invested in ensuring that the entirety of the PLO spectrum is effectively addressed within WBL.

WBL components	Learning Outcomes
1. **Industry Practitioner-Led Sessions** (Table 1, example 19)	1. PLO Application Design 2. PLO Application Development 3. PLO Component Integration 4. PLO Testing 5. PLO Continuous Learning 7. PLO Soft competences 6. PLO Profession related competences
2. **Implementation Business Industry Project** (Table 1, example 20)	1. PLO Application Design 2. PLO Application Development 3. PLO Component Integration 4. PLO Testing 5. PLO Continuous Learning 7. PLO Soft competences 6. PLO Profession related competences

Table 4. WBL components and related learning outcomes, Developer EQF7

4.2.5 DevOps expert EQF6

In the WBL examples "**Web Development**" (Table 1, examples 14 and 15) by the University of Ljubljana (See table 5), learners must develop an interactive web page with backend and frontend components. The learner can freely select the problem domain for the application. In the WBL component "**Software testing**" (Table 1, example 13) by Global Knowledge France through integrated learning the learners will have to develop all deliverables that are needed for software testing.

The other WBL examples address specific areas of work of DevOps. In the WBL task "**Virtualization**" (Table 1, example 17) learners have to automate the creation of a virtual machine image, with two different technologies: Vagrant and Cloud-init. In the WBL component named "**Containerization**" (Table 1, example 12) the learners must containerize the application stack and automate deployment with docker-compose (single server). "**Orchestration**" (Table 1, example 16) is a WBL component where learners must automate the creation and deployment of complex containerized application stack in Kubernetes.

Most of the PLOs of the learning programme are addressed across all WBL components. The focus is on tasks directly linked with the different activities of the software development life cycle. Furthermore, each of the WBL components incorporates PLOs pertaining to soft competences, thereby affirming a dedicated commitment towards the holistic development of learners.

The WBL components in focus are not of large scale and address foundational PLOs. They can be predominantly characterised as frameworks, which can be easily updated by changing the tasks they address as needed by the changing working environment. These examples furnish learners with explicit tasks and tools, thereby necessitating independent

exploration and resolution. This pedagogical approach is designed to cultivate self-reliance and encourage learners to proactively engage in problem-solving processes.

WBL components	Learning Outcomes
University of Ljubljana	
1. **Web Development-Backend** (Table 1, example 14) 2. **Web Development-Frontend** (Table 1, example 15)	1. PLO Application Development 2. PLO Component Integration 8. PLO Soft competences 9. PLO Functioning in organisations
3. **Software testing**	3. PLO Testing 8. PLO Soft competences 9. PLO Functioning in organisations
4. **Orchestration** (Table 1, example 16) 5. **Virtualization** (Table 1, example 17) 6. **Containerization** (Table 1, example 12)	2. PLO Component Integration 5. PLO ICT Systems Engineering 8. PLO Soft competences 9. PLO Functioning in organisations
Global Knowledge France	
7. **Implementing a software factory** (Table 1, example 18)	1. PLO Application Development 2. PLO Component Integration 3. PLO Testing 4. PLO Solution Deployment 5. PLO ICT Systems Engineering 7. PLO Profession related competences 8. PLO Soft competences 9. PLO Functioning in organisations

Table 5. WBL examples and related learning outcomes for DevOps EQF 6

Figure 1 depicts the presentation of the results of the Web Development WBL–project, which was part of piloting the ESSA DevOps Learning Programme in 2023.

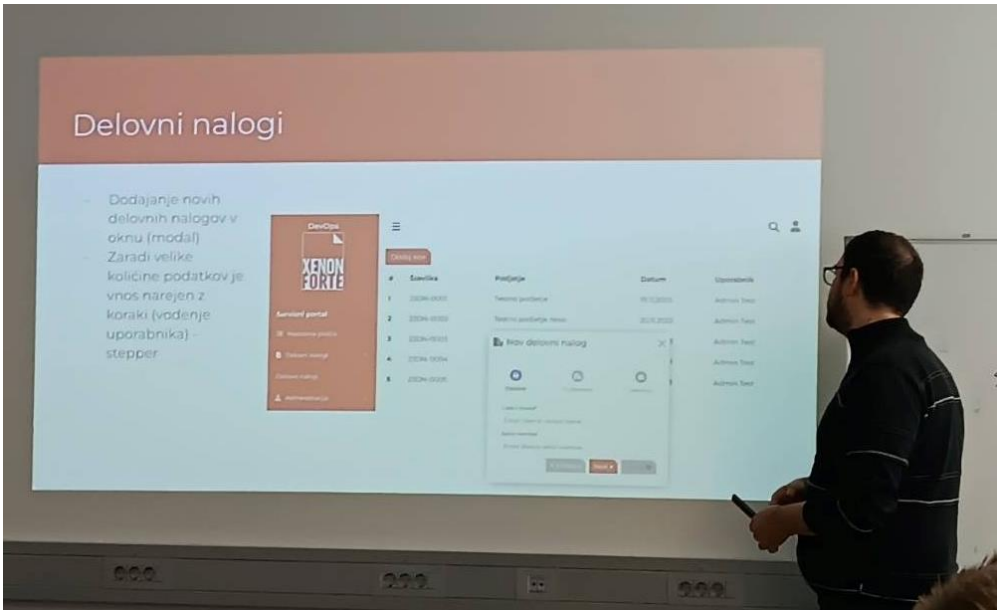

Figure 1. Presentation of Web Development Front-End project

4.2.6 DevOps EQF7

The WBL examples exhibited for this profile are closely related to specific PLOs of that learning programme. In the WBL component "**Component Integration DevOps in a Large-scale Information System**" (Table 1, example 21) the learners start by establishing standards of practice for continuous integration, to ensure that the innovative solution can be implemented and maintained over time. In the next WBL "**Testing DevOps in a Large-scale Information System**" (Table 1, example 23) the learners must design test automation strategies tailored for innovative solutions and generate a comprehensive report on configuration management. In the third WBL "**Implementing DevOps in a Large-scale Information System**" (Table 1, example 22) DevOps experts must collaborate for the transition of a large-scale information system to a continuous delivery mode.

The WBL component descriptions do not go into much detail, consequently, learners are encouraged to leverage their existing work experience, thereby fostering an environment conducive to autonomous research and independent application of knowledge.

WBL component	Learning Outcomes
1. **Component Integration DevOps in a Large-scale Information System**	1. PLO Component Integration
2. **Testing DevOps in a Large-scale Information System**	2. PLO Testing
3. **Implementing DevOps in a Large-scale Information System**	7. PLO Functioning in organisations

Figure 2. WBL examples and related learning outcomes for DevOps EQF7

4.2.7 Solution Designer EQF7

These WBL are deliberately tailored to address the fundamental PLOs. WBL tasks address various facets of e-commerce solutions, spanning from conceptual design to the impact of the deployed solution. In the WBL component "**Cloud-Based Ecommerce Solution Design**" (Table 1, example 25) by Hellenic Open University the learners must design and develop a cloud-based e-commerce solution. WBL component "**Cloud-Based Ecommerce Architecture Design**" (Table 1, example 24) is about the design of a cloud-based e-commerce architecture for a hypothetical business. In the WBL component "**Cloud-Based Solution Technology Design** " (Table 1, example 26), the learners have to research and analyse cloud-based technology's potential impact on an organisation or society, addressing security and ethical concerns, and writing a comprehensive report with recommendations. In all those WBL components learners are entrusted with the responsibility of conducting small research, writing a report or documenting the solution.

WBL components	Learning Outcomes
1. **Cloud-Based E-commerce Solution Design** (Table 1, example 25)	1. PLO Needs Identification 3. PLO Innovating
2. **Cloud-Based Ecommerce Architecture Design** (example 24)	2. PLO Architecture Design
3. **Cloud-Based Solution Technology Design** (example 26)	4. PLO New Technology

Figure 3. WBL components and related learning outcomes, Solution Designer EQF7

4.3 Soft Skills in WBL Learning Outcomes

For all collected WBL examples the importance in soft skills is an equally vital facet alongside technical expertise and professional development. This acknowledges that developers who have managed to combine both technical and soft skills can do better in the rapidly evolving and challenging world of software development.

The most important soft skills in ESSA programme learning outcomes are:

- Communication
- Teamwork
- Problem solving
- Self-management
- Critical thinking
- English language

All these soft skills are represented in WBL examples. The most focus has been given to:

- Communication
- Teamwork/collaborative work
- Problem-solving
- Critical reflection and thinking

Additionally, WBL components are addressing soft skills:

- Creativity
- Innovation, brainstorming, idea creation
- Presenting ideas
- Documenting, writing reports, making presentations.

The majority of WBL examples demonstrate a discernible incorporation of soft skills PLOs. Only in the instances where the WBL example is focused explicitly on the acquisition of specific technical proficiencies the soft skills PLOs were not explicitly emphasised.

In bigger-scale WBL examples, where the complexity is higher and there is a direct reference to industry practices, the integration of soft skills is constant, recognising their substantive significance within the professional landscape.

4.4 General Conclusions of ESSA WBL Components' Analysis

Based on the review and analysis of WBL components' descriptions, several key conclusions have emerged:

- Alignment with learning outcomes.

The extent to which WBL components exhibit a good alignment with PLOs is primarily contingent upon the scope and scale of the assigned tasks. When tasks are small scaled,

there is a corresponding need to focus on a pivotal PLO, typically focusing on one or two primary objectives. WBLs of larger volume are as a rule targeting most of the needed learning outcomes and attempt to integrate both technical and general skills in the best possible way.

- WBL components are mainly described generally without too much detail. In this way they serve like frameworks allowing to adopt dynamic changes and specifications as needed.

As common practice most of the examples demonstrate a propensity towards generality, primarily functioning as frameworks that provide recommendations for the application of specific technologies or tools.

- Emphasis on Scenario-Based Learning for detailed WBL components

Notably, the more detailed WBL components adopt a scenario-based approach, underlining the efficacy of this method in facilitating comprehensive learning experiences. Using Scenario-based Learning is natural to software development because the domain of software development itself is in some sense based on a predetermined scenario (software development process or life cycle).

- Profile-dependent characteristics of WBL components

Lower EQF profile levels exhibit smaller, more detailed, and specific components, while higher-profile levels feature larger components that are predominantly more general and open, targeting wider research and implementation tasks.

- Scenario-based Learning facilitates personalised learning and continuous professional development.

The utilisation of scenario-based learning emerges as an important factor, providing an enriched environment that fosters personalised learning experiences and a culture of continuous professional development.

Within the higher-scale WBL components, a notable emphasis is placed on the cultivation of the understanding of the need for continuous professional development. This is particularly evident in contexts such as internships and business/industry-defined projects, wherein there is a pronounced focus on fostering lifelong learning, personal growth, self-directed learning, and self-management. The integration of WBL within the learning programme manifests as a catalyst for experiential learning, a process that inherently engages and motivates learners. By allowing learners to grapple with authentic, real-world tasks, WBL cultivates a heightened state of preparedness for future professional work.

5 Recommendations for WBL Development

5.1 Criteria for Effective WBL Components

This subsection brings out and explains the criteria used by the team to select the finest WBL examples. These criteria are derived from the analysis of WBL components and discussions with partners who developed and described these components. These criteria and examples are chosen as a commendable guide for developing WBL and possibly integrating components of these examples into learning practices.

The selection of the exemplary cases is based on the assessment of WBL component descriptions meeting and utilising the following criteria:

- **Realistic and Authentic Situations or Scenarios**

WBL endeavours are meticulously designed to immerse learners in authentic and industry-relevant experiences, encompassing mentorship programmes and industry-led projects, thus facilitating a seamless transition into practical work placements (See Annex 1). If the WBL tasks are either too general or the description of the real-life working environment is left completely open, it is not possible to assess the authenticity of the tasks. For the learners it is hard then to relate the tasks with working environment and such WBL does not serve its purpose.

- **Relevance to Programme Learning Outcomes** (PLOs)

The alignment between WBL tasks and defined PLOs demonstrates a well-calibrated integration (See Annexes 4 and 5). For developing a comprehensive learning programme and curriculum, it is very important to choose and set WBL tasks focusing on the most important learning outcomes. The specific programme learning outcomes should be determined, what can be best learned in the working environment.

- **Comprehensive Skill Coverage**

Beyond technical proficiencies, WBL endeavours are thoughtfully curated to encompass a diverse spectrum of skills, including the pivotal PLOs related to soft skills (See Annexes 1 and 2).

- **Facilitates Continuous Professional Development**

The structured inclusion of WBL initiatives lends substantial support to the sustained professional growth and development of learners (See Annexes 3 and 5).

5.2 Recommendations for Integrating Work-Based Learning into Learning Programmes:

1. **Incorporating WBL as an intrinsic element in software development education**

WBL should be seamlessly woven into the fabric of software development education, warranting the inclusion of a dedicated WBL component within the learning programmes.

Following and covering the software development process steps and life-cycle logic is the best and most natural approach (See Annexes 1 and 2).

2. **Collaborative engagement with stakeholders and industry partners**

Active involvement of stakeholders and industry partners is imperative in crafting purposeful WBL components. Their expertise and insights are invaluable in tailoring components that resonate with real-world demands. Always relate the WBL to specific work situations and job positions where these tasks are relevant. (See Annexes 1).

3. **Alignment with programme learning outcomes**

A judicious alignment with the most critical learning outcomes is essential for the efficacy of WBL components. Tasks, scenarios, and situations within WBL should mirror authentic and pragmatic settings (See Annexes 4 and 5).

4. **Leveraging Scenario-Based Learning for continuous professional development**

Scenario-based learning stands out as an effective means to fortify learners' professional growth and development. Its application in WBL can offer a robust platform for dynamic learning experiences. Understand and follow the working scenarios of the learning profiles when designing the learning programmes. Integrate the theoretical and WBL components to follow real-life work scenarios. (See Annex 1, 3, 5).

5. **Fostering soft skills development through WBL**

The incorporation of WBL provides a base for the cultivation and development of soft skills. This not only augments learning but also offers a platform for the practical application and refinement of these vital competences. When designing WBLs include and explain the soft skills importance and their relevance to acquiring technical skills. (See Annexes 1 and 2).

6 Annexes

6.1 Annex 1: WBL example: Industry-related Software Development Project

INDUSTRY RELATED SOFTWARE DEVELOPMENT PROJECT

Junior Developer (EQF4/5)

DTSL

1. Related Learning Unit(s)
List the Programme Learning Outcomes (PLOs) and Unit Learning Outcomes (ULOs) addressed by this Work-based Learning (WBL)

PLO1: Application Design
Interprets a basic database design.
PLO2: Application Development
Participates in a development process and applies a common software development method (e.g., agile).
Creates a simple working software component or application**,** taking into account architecture, design requirements and possible other constraints (e.g., installability) applying relevant tools and techniques (e.g., object-oriented programming; IDE, CASE; editors, compilers, version control tools).
PLO3: Component Integration
Creates a simple working software component or application**,** taking into account architecture, design requirements and possible other constraints (e.g., installability) applying relevant tools and techniques (e.g., object-oriented programming; IDE, CASE; editors, compilers, version control tools).
PLO4: Testing
Writes an (automated) test on a piece of code.
Performs common test activities, applying testing and debugging techniques and tools.
PLO5: Documentation Production
Provides different (parts of) common technical documents, using appropriate tools (e.g., software documentation tools).
PLO6: Problem Management
Systematically resolves or escalates incidents and problems, resulting in a solved incident (e.g., by applying techniques and tools for troubleshooting such as diagnostic tools).
PLO6: New Technology
Explains the principles, related concepts, advantages and disadvantages of a new technology.
Applies basic methods, techniques and tools related to a new technology.
Writes a report on the application of a method, technique or tool related to a new technology.
PLO7: Profession related Competences
[Project skills] Works in project settings, applies project management methods and tools.

[Security skills] Applies and reports on methods, tools and techniques related to security.
[Software life cycle skills] Applies and reports on methods, tools and techniques related to software lifecycle processes.
[Ethical awareness skills] Is aware of basic ethical considerations and issues.

PLO8: Soft Competences

[Teamwork skills] Works together with others in a team.
[Communication skills] Communicates with peers, colleagues, supervisors and or relevant other, appropriately to the context, using conventions that are relevant to professional practice. Explains and gives instruction.
[Problem solving skills] Distinguishes and analyses fairly complex and unpredictable problems. Solves these problems systematically and in a creative way, using existing procedures and guidelines and own solutions by identifying and using data.
[Self-management skills] Realises personal development on request, where necessary with support, through self-reflection and external- and self-evaluation of own (learning) results.

PLO9: Functioning in Organisations

Explains 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.
Writes a report on functioning in organisation.

2. Description/Learning Outcomes
A short description of the task and planned learning outcomes

On successful completion of this learning unit, students' will be able to:

- Work together with others in a team.
- Communicate with peers, colleagues, supervisors and/or relevant others, appropriately to the context, using conventions that are relevant to professional practice. Explains and gives instruction.
- Master 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.
- Distinguish and analyse fairly complex and unpredictable problems. Solve these problems systematically and in a creative way, using existing procedures and guidelines and own solutions by identifying and using data.
- Exercise self-management within the guidelines of contexts that are usually predictable, but are subject to change. Is able to cope with limited change and to adapt to a certain level of variety in the workplace. Cope with pressure and stress setbacks and maintains composure. Show some initiative and carries responsibility for the results of own activities, work and or study. Works correctly and carefully.
- Realise learning and personal development on request, where necessary with support, through self-reflection and external- and self-evaluation of own (learning) results.

3. Expected Output Format
Indicate the format output criteria, i.e.,

Each pod (team) create the design and develop the software application. At the end of the six weeks, they present their project to the panel and take questions on their technical approach, how they organised as a team, how they agreed and assigned tasks and how well they worked together in this practical and reflective presentation.

4. Time Estimation
Estimate the learning time (in study hours)
60 hours

5. Recommended To-Do List
Please add a short instruction here. This is optional if described elsewhere or not relevant for this WBL

Developing a program involves steps like any problem-solving task. There are five main steps in the programming process:

1. Defining the problem
2. Planning the solution
3. Coding the program
4. Testing the program
5. Documenting the program

6. Recommended Work Environment, Hardware and Software List
Please add a short instruction. This is optional if described elsewhere or not relevant for this WBL

The first version will be written using C# and the Net Framework and will save data into a relational database. The second version will be written in Python and store the data as serialized objects in programmer-defined files.

7. Assessment Description and Criteria
Please add a short instruction. This is optional if described elsewhere or not relevant for this WBL
Project Evaluation Criteria:

1. Participation in the project (seeking information, giving information, offering an opinion, offering help to others, clarifying, applying information/formulating solutions)
2. Understanding of Technical Material (content of questions, quality of answers, ability to articulate ideas clearly and coherently, able to explore topic in greater depth, intelligent arguing based on technical knowledge, technical outcome of project, project design)

3. Interpersonal skills (able to work constructively with others, challenge the views of others without animosity, accept criticism with composure, ability to communicate ideas in the group).

8. Recommendations and Technical Instruction for the Teacher
Please add a short instruction. This is optional if described elsewhere or not relevant for this WBL

Project instructions and evaluation criteria is outlined in the document *ESSA_Software_Pathway_Project_JUNIOR_DEVELOPER_(EQF4_7).docx*

Teachers should:

- Assign learners to their team /pod.
- Provide the document.
- ESSA_Software_Pathway_Project_JUNIOR_DEVELOPER_(EQF4_7).docx containing the project to the students.
- Be prepared to answer questions related to the project.
- Teams can seek guidance from their mentor (if they have one)
- Teachers will need to bring together a review board comprising a representative from Industry, Government and Academia who will assess the project and interview the students on completion of the project.

6.2 Annex 2: WBL example: Cloud-based e-commerce Architecture Design

Cloud-Based e-Commerce Architecture Design

Solutions Designer (EQF7)

Hellenic Open University

1. Related Learning Unit(s)
List the Programme Learning Outcomes (PLOs) and Unit Learning Outcomes (ULOs) addressed by this Work-based Learning (WBL)

PLO2: Architecture Design:
Proposes a coherent architecture design for an innovative/ advanced solution or technology, taking into account relevant business and technological issues (e.g., business evolution and needs, budget and other resources; current technology, obsolescent equipment)
PLO2: Architecture Desing:
Develops a technology roadmap; an approach or strategy to implement a solution or technology (e.g., identifies change requirements, components affected/ involved by the implementation of specific solutions/ services)

2. Description/Learning Outcomes

A short description of the task and planned learning outcomes

Design a cloud-based ecommerce architecture for a hypothetical business. The participants will be required to consider various aspects of e-commerce such as user interface, product catalogue management, shopping cart functionality, payment processing, order management, customer relationship management, and data analytics. They will also need to consider the scalability, reliability, and security of the system. The design should leverage cloud technologies to optimise cost, performance, and scalability.

Upon completion of this task, participants should be able to:

- Understand the key components and operations of an e-commerce system.
- Apply principles of cloud computing to design scalable, reliable, and secure e-commerce systems.
- Make informed decisions about the use of cloud services and technologies in e-commerce.
- Evaluate the trade-offs in different design choices.

3. Expected Output Format

Indicate the format output criteria, i.e.,

- **Design Document:** A comprehensive document detailing the cloud-based e-commerce architecture design, including diagrams, descriptions, and justifications.
- **Presentation Slides:** A set of slides used for the presentation, including visuals illustrating the design and key points of the presentation.

4. Time Estimation

Estimate the learning time (in study hours)

8 hours

5. Recommended To-Do List

Please add a short instruction here. This is optional if described elsewhere or not relevant for this WBL

- Research: Begin by researching the key components of an e-commerce system and the principles of cloud-based architecture design. Familiarise yourself with different cloud services and technologies that can be used in e-commerce.
- Design: Design a cloud-based architecture for an e-commerce system. The design should include all the key components of an ecommerce system and leverage cloud technologies to optimise cost, performance, and scalability. Make sure to consider the scalability, reliability, and security of the system.
- Document: Write a comprehensive design document detailing your design. The document should include diagrams, descriptions, and justifications. Make sure the document is clear, concise, and well-organised.

6. Recommended Work Environment, Hardware and Software List

Please add a short instruction. This is optional if described elsewhere or not relevant for this WBL

7. Assessment Description and Criteria

Please add a short instruction. This is optional if described elsewhere or not relevant for this WBL

- Design Document: The quality, comprehensiveness, and clarity of the design document. The document should effectively communicate the design, the rationale behind the design decisions, and the evaluation of the design's scalability, reliability, and security.
- Presentation: The effectiveness of the presentation in communicating the design and justifications. The presentation should be engaging, well-structured, and supported by appropriate visual aids.

6.3 Annex 3: WBL example: Industry Practitioner-led Sessions

Examples of different types of WBLs considered by partners most relevant and most useful will be presented in Annexes.

Developer EQF7: Industry Practitioner-led Sessions

Warsaw School of Computer Science

1. Related Learning Unit(s)

List the Programme Learning Outcomes (PLOs) and Unit Learning Outcomes (ULOs) addressed by this Work-based Learning (WBL)

Programme Learning Outcomes (PLOs)

PLO 1: Application Design: Industry Insights and Trends

- Gain valuable insights into the latest trends, technologies, and best practices directly from industry specialists.
- Understand how these trends are shaping the field of Fullstack Development using .NET Core and React.

PLO 2: Application Development: Real-World Application

- Relate theoretical concepts to real-world scenarios presented by industry practitioners.
- Recognize the practical application of skills learned through exposure to actual industry challenges.

PLO 3: Component Integration: Innovative Problem-Solving (PLO 3)

- Develop innovative problem-solving skills by learning how industry professionals address complex issues.
- Apply creative thinking to adapt industry-proven solutions to unique development challenges.

PLO 4: Testing: Network and Professional Interaction (PLO 4)

- Establish connections with industry practitioners, fostering opportunities for mentorship and collaboration.
- Engage in meaningful discussions with professionals to broaden perspectives on Fullstack Development.

PLO 5: Continuous Learning (PLO 5)

- Embrace a culture of lifelong learning by staying updated with industry trends shared by experts.
- Cultivate a commitment to ongoing development through exposure to new tools and techniques.

PLO 7: Functioning in Organisation

- Learn how to functioning and behave in organizations from practitioners, based on real case studies and stories from real projects.

Unit Learning Outcomes (ULOs):

PLO 1: Industry Insights and Trends

- Attend sessions conducted by industry specialists to gain up-to-date insights into current and emerging trends.
- Analyse how the presented trends influence the development landscape and the adoption of .NET Core and React.

PLO 2: Real-World Application

- Apply theoretical knowledge to real-world case studies presented by industry practitioners.
- Recognize the practical relevance of concepts learned in the classroom within actual industry contexts.

PLO 3: Innovative Problem-Solving

- Observe industry professionals addressing complex problems and challenges.
- Discuss innovative approaches taken by specialists to tackle unique development scenarios.

PLO 4: Network and Professional Interaction

- Engage in interactive sessions and discussions with industry experts.
- Build professional relationships and gain insights from experienced practitioners.

PLO 5: Continuous Learning

- Stay updated with the latest tools, methodologies, and practices used in the field.
- Embrace a mindset of continuous learning and self-improvement based on insights from industry sessions.

2. Description/Learning Outcomes

A short description of the task and planned learning outcomes

In this task, students will have the opportunity to participate in sessions led by industry practitioners who are experts in Fullstack Development using .NET Core and React. These sessions will involve presentations, discussions, and interactive Q&A sessions, allowing students to directly learn from professionals actively working in the field. Industry practitioners will share their insights, experiences, and the latest trends, providing valuable real-world perspectives on the application of JAVA or .NET Core and React in various projects.

Planned Learning Outcomes:

- Industry Insights: Gain firsthand insights into the latest industry trends, technologies, and best practices directly from industry practitioners.
- Real-World Relevance: Understand how theoretical concepts taught in the classroom are applied in real-world scenarios through case studies and practical examples.
- Problem-Solving Acumen: Develop innovative problem-solving skills by learning about the challenges faced by professionals and the strategies they employ to address them.
- Networking and Mentorship: Establish connections with industry experts, facilitating networking opportunities and potential mentorship relationships.
- Lifelong Learning: Cultivate a commitment to continuous learning by staying updated with current industry developments and adapting to evolving trends.

Through these Industry Practitioner-Led Sessions, students will benefit from the direct knowledge transfer from seasoned professionals, enhancing their understanding of Fullstack Development and bridging the gap between academic concepts and practical industry applications.

3. Expected Output Format

Indicate the format output criteria, i.e.,

- Session Notes and Summaries: Students are required to take comprehensive notes during each session. These notes should capture key insights, trends, best practices, case studies, and any practical examples shared by the industry practitioners.
- Reflections and Insights: Following each session, students should provide reflective write-ups summarizing their main takeaways from the session. They should highlight how the information presented aligns with their existing knowledge and its relevance to the field of Fullstack Development.
- Questions and Discussion Points: Students should prepare thoughtful questions and discussion points based on the session content. These questions can be used during interactive Q&A segments to engage with industry practitioners and gain deeper insights.
- Application in Course Content: Students should analyse how the information from the practitioner-led sessions relates to the course content. They should identify areas where the industry insights can enhance their understanding of concepts learned in class.
- Networking and Professional Interaction: Students should actively engage with industry practitioners during the sessions, fostering connections and potentially initiating post-session discussions to expand their professional network.
- Personal Growth and Learning: Students should write reflections on how attending these sessions has contributed to their personal growth, understanding of industry trends, and commitment to continuous learning.

The combination of these outputs will demonstrate active engagement, critical thinking, and a proactive approach to leveraging the insights gained from the Industry Practitioner-Led Sessions. It will showcase the students' ability to connect theory with practical applications and their commitment to professional development.

4. Time Estimation
Estimate the learning time (in study hours)
15 hours

5. Recommended To-Do List
Please add a short instruction here. This is optional if described elsewhere or not relevant for this WBL

Suggested list of tasks that students should complete before, during, and after Industry Practitioner-Led Sessions. These tasks will help students maximise their learning experience and engagement with industry experts:

Preparation Before the Session:

- Research the background and expertise of the industry practitioner.
- Review relevant course content to have a foundational understanding of the topics to be discussed.
- Prepare questions and discussion points based on the session theme.

During the Session:

- Actively listen and take detailed notes on key insights, trends, and practical examples shared by the industry practitioner.
- Participate in interactive discussions and ask questions to clarify doubts and gain deeper insights.
- Engage in networking opportunities by introducing yourself and expressing interest in further discussions.

Post-Session Reflections:

- Reflect on the main takeaways from the session and how they align with your current knowledge.
- Write a summary of the session, highlighting key points, case studies, and innovative approaches discussed.
- Identify how the information from the session can enhance your understanding of Full-stack Development concepts.

Application to Course Content:

- Analyse how the insights gained from the session can be applied to course projects, assignments, and practical exercises.
- Consider how industry best practices and trends can be integrated into your development practices.

Networking and Professional Interaction:

- Connect with the industry practitioner on professional networking platforms if appropriate.
- Reach out with any follow-up questions or discussions to continue learning from their expertise.

Personal Growth and Continuous Learning:

- Reflect on how attending the session has contributed to your personal and professional growth.

- Commit to incorporating the learnings from the session into your ongoing journey of continuous learning.

Completing this recommended to-do list will help you derive the maximum benefit from Industry Practitioner-Led Sessions, allowing you to bridge the gap between theory and practice, expand your network, and stay up-to-date with industry trends.

---

6. Recommended Work Environment, Hardware and Software List

Please add a short instruction. This is optional if described elsewhere or not relevant for this WBL

Work Environment:

- Quiet and comfortable space conducive to focused learning and participation.
- Reliable internet connection to stream sessions and engage in online discussions.
- Notetaking materials such as a notebook or digital note-taking app.

Hardware:

- Personal computer or laptop with sufficient processing power and memory.
- Webcam and microphone for interactive participation and discussions.
- Headphones or earphones to ensure clear audio quality.

Software:

- Updated web browser (e.g., Google Chrome, Mozilla Firefox) for accessing online sessions and materials.
- Communication tools such as Zoom, Microsoft Teams, or a similar platform for virtual sessions.
- Note-taking applications for digitally capturing session insights and reflections.

Additional Tips:

- Ensure your computer's operating system and software are up to date to prevent technical issues.
- Test your webcam, microphone, and headphones in advance to ensure they are functioning properly.
- Disable any notifications or distractions during the session to maintain focus.

Having a suitable work environment, hardware, and software setup will enable you to fully engage with the Industry Practitioner-Led Sessions, interact with experts, and derive maximum value from the learning experience.

7. Assessment Description and Criteria

Please add a short instruction. This is optional if described elsewhere or not relevant for this WBL

Assessment Description: The assessment for Industry Practitioner-Led Sessions aims to evaluate students' active participation, comprehension of industry insights, and their ability to apply gained knowledge to their studies. These sessions provide valuable opportunities for students to engage with industry professionals and enhance their understanding of real-world practices in Fullstack Development using JAVA or .NET Core and React.

Assessment Criteria:

Active Participation (20%)

- Actively engage in discussions, ask questions, and provide thoughtful insights during the practitioner-led sessions.
- Demonstrate an eagerness to learn, contribute to discussions, and show respectful interaction with industry experts.

Comprehension of Insights (30%)

- Reflect on and summarize the key insights, trends, and case studies shared by industry practitioners.
- Showcase an understanding of the main takeaways from the sessions and their implications for Fullstack Development.

Application to Studies (25%)

- Analyse how the knowledge gained from the sessions can be integrated into course content and assignments.
- Present ideas on how industry best practices and trends can enhance their academic studies and practical exercises.

Critical Reflection (15%)

- Write post-session reflections highlighting personal learnings, areas of interest, and potential areas for further exploration.
- Offer insights into how attending the sessions has contributed to their understanding of industry dynamics.

Professional Interaction (10%)

- Demonstrate professionalism by engaging in networking with industry practitioners, if appropriate.
- Initiate follow-up discussions or questions, displaying a proactive approach to learning from industry experts.

8. Recommendations and Technical Instruction for the Teacher

Please add a short instruction. This is optional if described elsewhere or not relevant for this WBL

Recommendations:

- Provide clear explanations of key concepts related to .NET Core and React or JAVA, ensuring students have a solid foundation.
- Encourage collaborative learning by assigning group projects that require teamwork and cooperation.
- Offer guidance on best practices for project organization, version control, and documentation.
- Foster a problem-solving mindset by challenging students with real-world scenarios and coding challenges.
- Emphasise the importance of staying up-to-date with industry trends and technologies.

Technical Instructions:

- Ensure students have access to the required development tools, including .NET Core SDK, Node.js, and a code editor (e.g., Visual Studio Code).

- Provide instructions for setting up a development environment for both .NET Core and React projects on various operating systems (Windows, macOS, Linux).
- Guide students through the process of creating a new .NET Core or JAVA project and setting up a fronted f.e React application from scratch.
- Teach the process of integrating frontend and backend components and demonstrate effective communication between them.
- Instruct on testing methodologies for JAVA or both .NET Core APIs and React user interfaces.
- Explain deployment options for JAVA or .NET Core applications and React apps, including hosting and server configuration.
- These recommendations and technical instructions will contribute to a comprehensive learning experience for students and aid the teacher in facilitating an effective and engaging JAVA or .NET Core and React curriculum.

6.4 Annex 4: WBL example: Creating a Web Application

Junior Software Developer EQF 4: WBL 4_Creating a web application to support training workouts

BCS Training

1. Related Learning Unit(s)
List the Programme Learning Outcomes (PLOs) and Unit Learning Outcomes (ULOs) addressed by this Work-based Learning (WBL)
• PLO 1: Creates a simple working software component or application. • PLO 1: Describes principles of user interface design. • PLO 2: Uses HTML basic elements and CSS for designing. • PLO 2: Writes code and related documentation to it using a common programming language and applying coding conventions (JavaScript, PHP; clean coding principle). • PLO 2: Explains concepts and principles of databases, data structures, and query languages (MySQL)

2. Description/Learning Outcomes
A short description of the task and planned learning outcomes
The learner must create a training app where the user inserts the workouts and it is possible to get summaries, such as how many km were completed, how many workouts were done, how many minutes/hours of training, how many calories were spent, etc.

3. Expected Output Format
Indicate the format output criteria, i.e.,
Web application based on main technologies: HTML, CSS, JavaScript, PHP, MySQL.

4. Time Estimation
Estimate the learning time (in study hours)

8 hours

5. Recommended To-Do List
Please add a short instruction here. This is optional if described elsewhere or not relevant for this WBL
1. Determine what data is needed to save in the database, i.e., what is the data describing one workout. Create a database table structure and describe the data fields. To create a database structure, use the PHP program. 2. Create an HTML form to insert the workout. To design the form, create a separate CSS file. Create a JavaScript function to validate the data entered in the form, save as a separate file. 3. Create an HTML page to output workouts in table form. To design the table, create a separate CSS file. 4. Create a PHP program to connect to the database and add data to the database. 5. Create PHP programs that output: a. all workouts for the period in chronological order; b. number of workouts by type; c. summary of all types (workout type, time, km, calories); d. what kind of workout has been done most often.

6. Recommended Work Environment, Hardware and Software List
Please add a short instruction. This is optional if described elsewhere or not relevant for this WBL
Visual Studio Code, XAMPP

7. Assessment Description and Criteria
Please add a short instruction. This is optional if described elsewhere or not relevant for this WBL
Sprint review: Working app, Presentation

8. Recommendations and Technical Instruction for the Teacher
Please add a short instruction. This is optional if described elsewhere or not relevant for this WBL
For beginners the teacher should create a PHP program to create the database table, also example programs to connect and insert data and retrieve data from the database.

6.5 Annex 5: Scenario-based Learning Programme Example: Implementation of Business/Industry Projects

Scenario-based approach to learning programme is presented in this example.

Developer EQF7: Implementation of Business/Industry Projects

Warsaw School of Computer Science

1. Related Learning Unit(s)

List the Programme Learning Outcomes (PLOs) and Unit Learning Outcomes (ULOs) addressed by this Work-based Learning (WBL)

Programme Learning Outcomes (PLOs)

PLO 1: Application Design:

- Understand and apply architectural principles for designing scalable and efficient web applications.
- Demonstrate proficiency in creating structured application designs using Java or .NET Core and React.

PLO 2: Application Development

- Develop complete web applications by integrating frontend and backend components using JAVA or .NET Core and React.
- Implement essential features, user interfaces, and data interactions for comprehensive application functionality.

PLO 3: Component Integration

- Integrate diverse components within a web application to ensure smooth data flow, communication, and interaction.
- Demonstrate the ability to connect frontend and backend modules seamlessly.

PLO 4: Testing

- Design and implement comprehensive testing strategies, including in most unit tests, and some other f.e. integration tests, and end-to-end testing.
- Ensure application functionality, reliability, and responsiveness through rigorous testing practices.

PLO 5: Profession Related Competencies

- Analyse and evaluate real-world projects that effectively utilized JAVA or .NET Core and React technologies.
- Understand the role of these technologies in addressing industry challenges and demands.

PLO 6: Soft Competences

- Enhance soft skills such as teamwork, communication, critical thinking, and adaptability through collaborative projects.
- Successfully work in diverse teams, adapting to varying roles and responsibilities.

PLO 7: Functioning in Organisation

- Simulate professional development environments, following project plans and adhering to industry best practices.
- Develop an understanding of the workflow, standards, and expectations within a professional development setting.

Unit Learning Outcomes (ULOs)

ULO 1: Application Architecture Mastery

- Understand architectural patterns commonly used in web applications.
- Apply architectural principles to design a web application using .NET Core and React.

ULO 2: Fullstack Development Proficiency

- Develop a complete web application by integrating frontend and backend components.
- Implement CRUD operations and user authentication for a fullstack application.

ULO 3: Seamless Component Integration

- Integrate a React frontend with a .NET Core backend to enable data flow and interaction.
- Combine different modules to create a cohesive and functioning application.

ULO 4: Quality Assurance and Testing

- Design and implement unit tests for React components and .NET Core APIs.
- Perform end-to-end testing to ensure the overall functionality and user experience.

ULO 5: Professional Adaptability

- Analyse case studies of successful projects that used .NET Core and React.
- Discuss the practical applications and benefits of these technologies in professional contexts.

ULO 6: Effective Collaboration

- Collaborate effectively within a team to develop a full-stack application.
- Present findings on teamwork dynamics and problem-solving strategies.

ULO 7: Industry-Aligned Execution

- Create project plans for building a full-stack application using JAVA or .NET Core and React.
- Follow coding standards and version control practices commonly used in the industry.

2. Description/Learning Outcomes

A short description of the task and planned learning outcomes

Students will be engaged in projects conducted in collaboration with IT companies. Their task will involve working on specific application or system projects, from design to deployment. They will collaborate in teams, analyse client requirements, design system architecture, and implement both frontend and backend components. Throughout the project, students will need to consider real-world constraints and business challenges.

Planned Learning Outcomes:

- Ability to practically apply knowledge in creating comprehensive complex IT solutions.
- Skill in effective teamwork on projects with real business applications.
- Development of the ability to analyze client requirements and design systems.
- Understanding of the application deployment process and the necessity of incorporating business aspects.

Gaining experience in working in an environment close to a real business setting.

3. Expected Output Format

Indicate the format output criteria, i.e.,

Students are expected to deliver results in the following format:

- JAVA/.NET Core Projects: Compiled and functional Java or .NET Core applications in the form of executable files or containers, along with documentation for installation and execution.
- Frontend Projects: Compiled and functional JSP or React applications in the form of static files (HTML, CSS, JavaScript), along with instructions for running in a web browser.

All documentation should be written in a clear and understandable manner, enabling individuals unfamiliar with the project to understand its functionality and steps for execution.

4. Time Estimation

Estimate the learning time (in study hours)

80 hours

5. Recommended To-Do List

Please add a short instruction here. This is optional if described elsewhere or not relevant for this WBL

A suggested list of tasks that students can complete as part of the Work-Based Learning program.

- Business Project Analysis: Conduct analysis and assessment of a business project related to frontend and backend application development.
- System Architecture Design: Create a system architecture plan, considering both frontend and backend components and their interactions.
- User Interface Implementation: Engage in creating a responsive user interface based on the design concept, including WCAG 2.1 standard requirements.
- Backend Function Creation: Develop backend functions to handle business logic and interaction with the database, including possible high demand usage, identify and reduce systems bottlenecks.
- Component Integration: Connect the frontend and backend, ensuring smooth operation of the application as a whole.
- Testing and Debugging: Test the application, identify and fix errors, and ensure its high quality.
- Performance Optimization: Optimize application performance by refining code and resources.
- Deployment and Monitoring: Deploy the application on a server, monitor its performance, and respond to any issues.
- Progress Reporting: Prepare a progress report of the project, highlighting the stages of application development and challenges encountered.
- Results Presentation: Showcase your achievements in application development to fellow students and mentors

6. Recommended Work Environment, Hardware and Software List

Please add a short instruction here. This is optional if described elsewhere or not relevant for this WBL

Recommended Work Environment:

- Operating System: Windows 10 or newer, macOS, or Linux (for .NET Core and React)
- Integrated Development Environment (IDE): Visual Studio Code, Visual Studio 2019, or JetBrains Rider (for .NET Core) or Java and Netbeans 12.5
- Web Browser: Google Chrome, Mozilla Firefox, or Microsoft Edge (for testing React applications)

Hardware List:

- Personal computer or laptop with sufficient computational power
- Processor: Multi-core processor, preferably Intel Core i5 or higher
- RAM: Minimum 8 GB RAM, recommended 16 GB or more
- Hard Drive: SSD (Solid State Drive) for fast data access and application startup

Recommended Software:

- .NET Core SDK: For creating and running .NET Core applications or JAVA SDK for java solutions.
- Node.js: For running framework e.g., React/Angular (or other) applications and managing dependencies.
- Visual Studio Code: Integrated development environment for code editing (optional: Visual Studio or JetBrains Rider) or Netbeans 12.5
- Git or GitHub: Version control system for tracking code changes

7. Assessment Description and Criteria

Please add a short instruction. This is optional if described elsewhere or not relevant for this WBL

Assessment Description:

This assessment aims to evaluate students' progress in understanding and practically applying skills related to JAVA or .NET Core and React/Angular (or other). The assessment will be based on practical projects and effective utilization of tools and technologies.

Assessment Criteria:

- Technology Understanding: The assessment is based on the degree of understanding of JAVA or .NET Core and React demonstrated in project documentation and presentations.
- Design and Implementation: Assessment considers the quality of projects, including project structure, architecture, and source code quality.
- Functionality and Performance: Evaluated are the application's functionality, correctness of components, as well as performance and responsiveness.
- Tool Utilization: Assessed is the proficient use of development tools such as IDEs, version control, and debugging.
- Dependency Management: Assessed is the effective management of dependencies and library packages in the React project.
- Documentation: Evaluated is the completeness and clarity of project documentation and the ability to present work.

- Team Collaboration: Assessed is engagement in teamwork, cooperation, and knowledge sharing.
- Creativity and Innovation: Evaluated is the demonstration of creativity in problem-solving and an innovative approach to projects.
- Note: Criteria may change depending on the course's specifics and instructor's guidelines.

8. Recommendations and Technical Instruction for the Teacher
Please add a short instruction. This is optional if described elsewhere or not relevant for this WBL

Recommendations:

- Provide clear explanations of key concepts related to .NET Core and React or JAVA, ensuring students have a solid foundation.
- Encourage collaborative learning by assigning group projects that require teamwork and cooperation.
- Offer guidance on best practices for project organization, version control, and documentation.
- Foster a problem-solving mindset by challenging students with real-world scenarios and coding challenges.
- Emphasize the importance of staying up-to-date with industry trends and technologies.

Technical Instructions:

- Ensure students have access to the required development tools, including .NET Core SDK, Node.js, and a code editor (e.g., Visual Studio Code).
- Provide instructions for setting up a development environment for both .NET Core and React projects on various operating systems (Windows, macOS, Linux).
- Guide students through the process of creating a new .NET Core or JAVA project and setting up a fronted f.e React application from scratch.
- Teach the process of integrating frontend and backend components, and demonstrate effective communication between them.
- Instruct on testing methodologies for JAVA or both .NET Core APIs and React user interfaces.
- Explain deployment options for JAVA or .NET Core applications and React apps, including hosting and server configuration.
- These recommendations and technical instructions will contribute to a comprehensive learning experience for students and aid the teacher in facilitating an effective and engaging JAVA or .NET Core and React curriculum.

6.6 Annex 6: Description of a Full Scenario-based Approach to Building a Learning Programme - Example by Codecool

The logic of a scenario-based approach to designing a full learning programme for Junior Software Developer EQ is presented in this example.

In one-year full-stack developer course learners move through 4 modules. In this course learners only move to a new module once they have mastered the material in the previous one.

Learners are working on projects in groups and getting individual mentoring in each module. Learners work together with new team members from time to time, while the supporting environment stays the same all the way.

Learning speed may differ from student to student, but everyone will have the same solid foundation by the time they graduate.

With this learning model, learners master the study material and complete the projects at a pace that is comfortable for them. Learners oversee their own learning process; they develop a growth mindset and professional confidence — all key factors for long-term success.

Learners are working on lifelike tech projects, in Scrum development teams, in a workplace simulation environment. They can learn the latest technologies, but also develop their soft skills, and get hands-on experience in product development and project management (https://codecool.com/en/about/mastery-based-learning-mentors/).

www.softwareskills.eu

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