European Software Skills Alliance.

EDUCATIONAL PROFILE

ESSA Junior Developer

EQF 4/5

This educational profile belongs to the field of Software Engineering and is covered by relevant references in EN 16234-1:2019 (e-CF) and ISO/IEC/IEEE 90003:2018.

Software developers build and create computer programs, including mobile applications, desktop applications, hybrid applications, or even sometimes operating systems. They may also be involved in **other aspects of software development**, including identifying user needs, software design, testing new software, software implementation, and making changes to the system. **Software developers play a critical role in many different professional fields** such as computer systems, manufacturing, finance, and software publishing.

Junior developers support all aspects of software development processes: development, testing, implementing, and maintaining of basic software solutions. They **master the codebase**, **attend design meetings, write basic code, and fix bugs**. They have an inquiring attitude, oversee the coherence, and work in a structured manner within clear boundaries.

Learn more:

About this profile **2**
- Competences 2
- Deliverables 2
- Professional perspectives 2
- Educational perspectives 3

Programme Learning Outcomes (PLO) **4**
Assessments **7**

The Erasmus+ project ESSA (European Software Skills Alliance) aims to skill, upskill, and reskill individuals into high-demand software roles across the EU.

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

This project has been funded with support from the European Commission. The document reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

About this profile

This profile is for educating people, with or without previous ICT knowledge, to build and implement software components and applications based on specifications and designs using programming languages, tools, and platforms.

The profile has an EQF 4/5 level. No prior knowledge of the topic is required.
This level requires:

- **Autonomy:** Work independently within specified boundaries and resolve issues which arise from project activities.
- **Context:** Structured/predictable context, well-defined, concrete, and abstract problems.
- **Content**: Common factual and theoretical knowledge that underpins the field of work or vocation in the local situation.

Competences

e-CF competences (incl. e-CF level)	General competences (incl. EQF level)
A.6. Application Design [e-2]	P. Profession-related competences (EQF5)
B.1. Application Development [e-2]	I.S. Interpersonal soft competences (EQF5)
B.2. Component Integration [e-2]	P.S. Personal soft competences (EQF5)
B.3. Testing [e-2]	F.O. Functioning in organisations (EQF5)
B.5. Documentation Production [e-2]	
C.4. Problem management [e-2]	

Deliverables

Learners should master the following deliverables:

- Simple relational database
- Code and related documentation
- Simple working software component/ application
- Modified existing software component/ application
- Installation report
- (Automated) test
- Test result documentation/ test report
- (Parts of) common technical documents
- Solved incident
- Report on the application of a method, technique or tool related to a new technology
- Self-reflection report
- Report on functioning in an organisation

Professional perspectives

Upon completing this programme, the learner is eligible to apply for **junior developer positions** such as Web Developer - Computer Programmer - Database Administrator - Software Developer - Software Engineer - Java Developer - Full-stack Developer - Font-end Developer - Back-end Developer

Educational perspectives

The learner may continue in a **Bachelor's Developer programme** with a focus on more advanced application technologies and architectures.

The learner also has a proper base for further training and certification such as Specific programming languages (e.g., Java, Python, C++) - Cybersecurity - Data analytics - Software architecture - Blockchain - Machine Learning/AI - DevOps - IoT/Automation/Robotics - Testing

Programme Learning Outcomes (PLO)

1. PLO Application Design [e-2]	
The learner has demonstrated capability *→ to interpret a design for a software application or component*	
Unit learning outcomes	Explains and distinguishes basic principles and terminology of software design (e.g., phases in the design process, common techniques, deliverables)
	Describes principles of user interface design
	Reads design models and diagrams (e.g., ERD, UML)
	Interprets a basic database design
	Interprets a design for an application or software component

2. PLO Application Development [e-2]	
The learner has demonstrated capability *→ to systematically develop a simple software application or component* *→ to propose modifications to an existing solution* *→ to document the development activities*	
Unit learning outcomes	Explains and distinguishes common software development methods (e.g., waterfall, iterative, agile), techniques (e.g., object-oriented) and tools (e.g., IDE, CASE; multimedia integration tools; app development tools)
	Describes common programming principles and terminology (e.g., secure programming)
	Explains concepts and principles of databases, data structures and query languages (e.g., SQL)
	Participates in a development process and applies a common software development method (e.g., agile)
	Creates a **simple relational database**
	Writes **code** and related **documentation** to it, by using a common programming language and applying coding conventions (e.g., Java, Javascript, PHP, Python; clean coding principle)
	Creates a **simple working software component or application,** taking into account architecture, design requirements and other possible constraints (e.g., installability) applying relevant tools and techniques (e.g., object-oriented programming; IDE, CASE; editors, compilers, version control tools)
	Modifies an existing software application or component

3. PLO Component Integration [e-2]	
The learner has demonstrated capability *→ to integrate efficiently a software application or component into an existing system* *→ to document the installation activities*	
Unit learning outcomes	Explains and distinguishes common methods, techniques and tools related to efficient integration
	Describes the interplay between and compatibility of system components
	Carries out installation and configuration activities, applying common methods, techniques and tools related to efficient integration (e.g., packaging and distribution, virtualisation, containerisation)
	Monitors and tests the connectivity of integrated systems
	Writes an **installation report**

4. PLO Testing [e-2]	
The learner has demonstrated capability *→ to test a software application or component* *→ to document test outcomes*	
Unit learning outcomes	Explains and distinguishes principles of software testing, common testing methods, techniques, and tools
	Writes **an (automated) test** on a piece of code
	Performs common test activities, applying testing and debugging techniques and tools
	Records and interprets test outcomes and writes **test result documentation/ test report**

5. PLO Documentation Production [e-2]	
The learner has demonstrated capability *→ to draft technical documentation*	
Unit learning outcomes	Describes types of technical documentation
	Provides different **(parts of) common technical documents**, using appropriate tools (e.g., software documentation tools)

6. PLO Problem management [e-2]	
The learner has demonstrated capability *→ to act systematically in handling incidents and problems*	
Unit learning outcomes	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

7. PLO Profession related competences [EQF5]	
The learner has demonstrated capability *→ to apply profession related skills*	
Unit learning outcomes	Masters common **ICT knowledge**
	Explains the principles, related concepts, advantages and disadvantages of a **new technology**. Applies and reports on basic methods, techniques and tools related to a new technology.
	Applies and reports on measures, methods, tools and techniques related to **security**
	Applies and reports on measures, methods, tools and techniques related to **software lifecycle processes**
	Is aware of basic **ethical considerations** and issues

8. PLO Soft competences [EQF5]	
The learner has demonstrated capability *→ to apply soft skills*	
Unit learning outcomes	**Works** together with others **in a team**
	Communicates with peers, colleagues, supervisors and/or relevant others, appropriately to the context, using conventions that are relevant to professional practice. Explains and gives instruction.
	Masters the **English language** at level B2. Can understand the main ideas of complex text on both concrete and abstract topics, including technical discussions in his/her field of specialisation
	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.
	Exercises **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. Copes with pressure and stress

	setbacks and maintains composure. Shows some initiative and carries responsibility for the results of own activities, work and or study. Works correctly and carefully.
	Realises **learning and personal development** on request, where necessary with support, through self-reflection and external- and self-evaluation of own (learning) results.

9. PLO Functioning in organisations [EQF5]	
The learner has demonstrated capability *→ to function in an organisational context*	
Unit learning outcomes	Explains the basics of **organisation theory and behaviour**
	Describes the relationship between **business and IT**
	Works in an organisational context under specific direction with limited autonomy and responsibility e.g., at the level of a trainee, junior or assistant
	Works in project settings, applies **project management** methods and tools
	Writes **a report on functioning in the organisation**

Assessments

Unit learning outcome	Assessment method	Validation of prior acquired competences (skills and knowledge)
1.1	Exam	Certification
1.2	Exam	Certification
1.3	Exam	Certification
1.4	Exam	Certification
1.5	Exam	Certification
2.1	Exam	Certification
2.2	Exam	Certification
2.3	Exam	Certification
2.4	Practical assignment	Assessment (of skills)
2.5	Practical assignment	Assessment (of skills)
2.6	Practical assignment	Assessment (of skills)
2.7	Practical assignment	Assessment (of skills)
2.8	Practical assignment	Assessment (of skills)
3.1	Exam	Certification
3.2	Exam	Certification
3.3	Practical assignment	Assessment (of skills)
3.4	Practical assignment	Assessment (of skills)
3.5	Report	Assessment (of report)
4.1	Exam	Certification
4.2	Practical assignment	Assessment (of skills)
4.3	Practical assignment	Assessment (of skills)
4.4	Report	Assessment (of report)
5.1	Exam	Certification
5.2	Practical assignment	Assessment (of skills)
6.1	Practical assignment	Assessment (of skills)
7.1	Exam	Certification
7.2	Practical assignment	Assessment (of skills)
7.3	Report	Assessment (of report)
8.1	Exam	Certification
8.2	360º assessment	360º assessment
8.3	Practical assignment	Assessment (of skills)
8.4	Practical assignment	Assessment (of skills)
8.5	Exam	Certification
9.1	360º assessment	360º assessment
9.2	360º assessment	360º assessment
9.3	Exam	Certification
9.4	Practical assignment	Assessment (of skills)
9.5	Self-reflection report	Assessment (of report)
10.1	Exam	Certification
10.2	Exam	Certification
10.3	360º assessment	360º assessment
10.4	Report	Assessment (of report)