1.3 Software Developer

1 Brief Description

Software Developers design and implement individual software building blocks [components and modules].

2 Responsibilities

Software Developers are responsible for the cost-effective design and implementation to requirements of individual software components based on existing system, database and GUI designs. They specify technical details for the software components and define interfaces to other systems components.
Software Developers draw up algorithms, define data structures and convert programmes to a higher [3GL or 4GL] programming language. They draw up test specifications, test data and test environments and conduct unit tests at the level of software components. They devise solutions within the development team and in cooperation with experts from the applications environment.

3 Profile-typical Work Processes

The partial processes described below document the entire profile-typical work process carried out by the IT specialists. Mastery of this work process, combined with proficiency in the respective fields of competency and practical experience, forms the basis for professional competency.

3.0.1 Support IT systems analysts and IT systems developers in the tasks of systems analysis and systems design, for example, by production of prototypes.

3.0.2 Collaborate in defining the development framework and the development environment, estimation of costs, definition of milestones and identification of implementation risks.

3.0.3 Check requirements models and system design documents for accuracy, clarity and comprehensiveness, and that the systems requirements with respect to security and performance can be met. Agree functional modifications and extensions with IT systems analysts, IT systems designers and other specialists in the area of development.

3.0.4 Refine systems design by imaging the specified, larger systems components on to smaller software components, such as, for example, classes and objects. Specify the interaction and relationships of these software components by means of appropriate diagrams.

3.0.5 Agree concrete interfaces and data formats with the development team.

3.0.6 Derive test cases and scenarios from the specifications for the software components and produce test data for the unit test.

3.0.7 Draft database tables and mechanisms for remote call up, including using code generators.

3.0.8 Encapsulate existing systems, convert data, image complex communications and query protocols on to classes and methods.

3.0.9 Implement the software components and carry out unit tests, record test results.

3.0.10 Implement installation programs.

3.0.11 Support systems integration and systems testing, or, in the case of smaller projects, carry out the systems integration with the support of the project developer.
3.0.12 Collaborate in the production of manuals, installation instructions and training materials.

4 Characteristic Areas of Competency

The ability to perform the profile-typical work processes requires varying degrees of proficiency in the following areas of professional competency. The competencies are assigned various levels of knowledge and ability and a range of typical methods and tools.

Universal competencies requiring fundamental proficiency:
company objectives and customer interests,
problem analysis and solution,
communication, presentation,
conflict recognition, resolution,
foreign-language communication [English],
project organisation, cooperation,
time management, task planning and prioritising,
economical decision-making,
self-teaching, learning organisation,
innovation potentials,
data protection, security,
documentation, standards,
quality assurance.

Group-specific competencies requiring in-depth proficiency:
methods and tools of software development,
engineering processes,
systems analysis,
development standards [performance, security, availability, innovation],
quality standards,
databases, networks.

Profile-specific competencies requiring working knowledge:
Module design, design models,
programming and display languages,
program libraries,
algorithms and data structures,
interfaces,
data models, formats, types.

5 Requirements for Certification

Generally, the requirement for certification is adequate qualification based on relevant vocational training or job experience.