THE NATIONAL COUNCIL FOR TECHNICAL AND VOCATIONAL EDUCATION AND TRAINING

OCCUPATIONAL STANDARDS

OCCUPATION: SOFTWARE ENGINEER

LEVEL: NTA LEVEL 7

FEBRUARY 2024

TABLE OF CONTENTS

ABBREVIATIONS

API	Application Programming Interface
BRD	Business Requirement Document
CMM	Capability Maturity Model
CBET	Competency Based Education and Training
GUI	Graphical User Interface
MRD	Market Requirement Document
NACTVET	National Council for Technical and Vocational Education and Training
NOS	National Occupational Standards
OS	Occupational Standards
PRD	Product Requirement Document
RAM	Random Access Memory
RPC	Remote Procedure Call
SLC	Software Life Cycle
SOA	Service-Oriented Architecture
SRS	Software Requirements Specification
TET	Technical Education and Training
TVET	Technical and Vocational Education and Training
UML	Unified Modelling Language

GLOSSARY OF TERMS

Circumstantial Knowledge:	Detailed knowledge, which allows the decision-making in regard to different circumstances and cross cutting issues.
Competence:	The ability to use knowledge, understanding, practical, and thinking skills to perform effectively to the workplace standards required in employment.
Competency:	A description of the ability one possesses when able to perform a given occupational task effectively and efficiently.
Competency-based Education:	An instructional programme that derives its content from validated tasks and bases assessment on the learner's performance.
Curriculum:	A description or composite of statements about "what is to be learned" by the trainee/student in a particular instructional programme; a product that states the "intended learning outcomes".
Educational/Training Programme:	The complete curriculum and instruction (what and how) that is designed to prepare a person for employment in a job or other particular performance situation.
Occupation:	A specific position requiring the performance of specific tasks – essentially the same tasks are performed by all employees having the same title. (Example: baker)
Occupational Area:	This is a broad grouping of related jobs. (Example: food service)
Occupational Competence:	The application of knowledge and skills that consistently meet the standards required by the work context.
Occupational Standards:	Specific requirements of competences people are expected to demonstrate in a particular occupational area, including knowledge and relevant attitudes. They also act as a performance tool of assessment of the prescribed outcomes.
Occupational/Job analysis:	A process used to identify the tasks that are important to employees in any given occupation**.**
Performance Criteria:	Indicate expected end results or outcomes in the form of evaluative statements.
Skills:	The ability to perform occupational tasks with a high degree of proficiency within a given occupation. Skill is conceived of as a composite of three completely interdependent components: cognitive, affective, and psychomotor.

Standards:	A set of statements, which if proved true under working conditions, means that an individual is meeting an expected level and type of performance.
Task Analysis:	The process of analysing each task to determine the steps, circumstantial knowledge, attitudes, performance standards, tools and materials needed, as well as safety concerns required for the employees performing it.
Task:	A work activity that has a definite beginning and ending, is observable or measurable, and consists of two or more definite steps that leads to a product, service, or decision.
Underpinning Knowledge:	Crucial knowledge that an individual must acquire in order to demonstrate competences that are associated in performing a given task.
Verification Process:	The process of having experts review and confirm the importance of the task (competency) statements identified through occupational analysis. Other questions, such as the degree of task learning difficulty are also frequently asked. This process is also sometimes referred to as validation.

1.0 INTRODUCTION

Technical Education and Training (TET) is one of the most important education sub-sectors in Tanzania, responsible for developing a skilled workforce to support the country's industrialization economic agenda. Tanzania's *Development Vision 2025* intends to raise the country's economy to a middle-income status, with a high level of human development. This requires a skilled workforce that is aligned with the needs of the public and private sectors of the economy. The National Council for Technical and Vocational Education and Training (NACTVET) has begun the job of drafting Occupational Standards (OS) that will eventually be adopted as National Occupational Standards (NOS) for use in the delivery of TET that meets the needs of the labour market and the country's economic agenda.

Occupational Standards (OS) are performance criteria that are matched with labour market demands. Each of them describes the functions, performance standards, and understanding or knowledge underpinning a given occupation. They combine skills, knowledge, and attitudes to describe best practice. They are useful tools for establishing job roles, personnel recruitment, supervision, and appraisal, as well as TET Standards. They are also helpful for benchmarking and harmonizing job qualifications on a national and international level. Standards, in general, provide a solid framework for high-quality TET that is labour market-relevant, current, and consistent in application across all public and private institutions.

However, it must be noted that Occupational Standards are different from Training /Education Standards. Occupational standards are defined in terms of activities performed by a person in a selected occupation (e.g., an electrical engineer designs electrical circuits, performs troubleshooting in electrical circuits, etc.), and are usually defined by Employers following procedures as agreed upon by all the stakeholders. On the other hand, Training and Education Standards are developed from the activities defined in the occupational standards, and they specify learning objectives to ensure that the necessary skills and knowledge are developed by a person to enable him/her to function at an agreed level in an occupation. Training and Education Standards are used to define curricula in training institutions. It is critical, however, to establish a direct link between the occupational standards and the training standards for both of them to respond collaboratively to the demands of the labour market.

For the purpose of TET delivery, Tanzania has adopted the Competence Based Education and Training (CBET) approach. The CBET approach focuses on providing learners with the skills and knowledge required to meet the occupational standards. Occupational standards are thus the starting point for developing competency-based training (CBET) programmes. Therefore, it is quite pertinent

for TET institutions to use the relevant occupational standards as a benchmark for formulating their curricula.

Occupational Standards are developed based on a given occupation's current and future demands. As a result, they serve as a means of bridging the gap between the worlds of employment and technical education and training.

The document explains how the occupational standards were developed, as well as the scope, the occupational profile in the form of DACUM charts, and the Occupational Standards.

2.0 OCCUPATIONAL STANDARD DEVELOPMENT PROCESS

The process of developing these Occupational Standards involved both local and international expertise. The process began with an examination of major documents that guide Tanzanian skills development including the *10-year National Skills Development Strategy (2016-2026).* NACTVET labour market reports were also used in the literature review to determine the skills demand in the Tanzanian labour market as a whole.

After the literature review, a team of experts in consultation with practitioners developed draft occupational standards. The draft document was used to develop an occupational profile for each occupation (DACUM Chart), which is attached as an **Appendix** to every Occupational Standard.

The occupational standards were validated during the stakeholders' forum held on 22nd and 23rd February 2024 at Morogoro. The information from the stakeholders' forum provides insight from the workplace, professional bodies, regulatory bodies and sector ministries regarding trends and changes in the profession, including how well graduates are prepared for working in the occupation.

3.0 THE SCOPE AND OVERVIEW OF THE OCCUPATION STANDARDS FOR SOFTWARE ENGINEERS

The standards cover a broad range of duties and tasks that can be performed by a Software Engineer. However, the occupational standards are not meant to replace individual job descriptions. Instead, they are to be used for guidance in defining skill levels and knowledge for the technician in specific settings or positions. The Software Engineer may perform tasks in a number of key areas in the occupational standards, but not necessarily in all areas. For example, in large operations, other individuals may be employed or designated to perform specific tasks.

Software Engineers work under the supervision of senior software engineers to analyse user needs, complete the design and development, customized service development, software testing and quality control of enterprise-level software projects. Software Engineers complete the full cycle of development and management of enterprise-level software projects in the workroom or office in accordance with user requirements, including user requirement and feasibility analysis, high level

design and detailed design of the system, software coding, debugging and testing, acceptance and operation, maintenance and upgrading. Generally, the Software Engineer performs the following responsibilities:

a) User requirement analysis and management

b) High level design of software system

c) Software code writing

d) Enterprise-level software project development

e) Software test and quality control

f) Product delivery

g) Whole software development process management

The Occupational Standards have been clustered into NTA qualification levels, i.e. NTA level 7 and 8.

4.0 VALIDITY PERIOD

Due to the rapid development of technology, the validity period of occupational standards is 3-5 years. The review will proceed in the same manner as the one before it, with new occupational standards being developed based on current trends of the labour market.

5.0 OCCUPATIONAL STANDARDS

5.1 OCCUPATIONAL STANDARDS FOR SOFTWARE ENGINEER - NTA LEVEL 7

<table>
<tr><td>OCCUPATION</td><td>SOFTWARE ENGINEER</td><td>OCCUPATION CODE</td><td></td></tr>
<tr><td>DUTY TITLE</td><td>PERFORM USER REQUIREMENT ANALYSIS AND MANAGEMENT</td><td>DUTY NO.</td><td>701</td></tr>
<tr><td>TASK TITLE</td><td>CONDUCT USER REQUIREMENT COLLECTION AND ANALYSIS</td><td>TASK NO.</td><td>7011</td></tr>
<tr><td>PERFORMANCE CRITERIA</td><td colspan="3">The person performing this task must be able to prepare the guideline according to the computer software requirements specification, and complete the work related to the collection and analysis of user requirements.</td></tr>
<tr><td>RANGE STATEMENT</td><td colspan="3">The task can be performed in the workroom or office under the supervision of software engineers.
The tools and equipment to be used include:
1. Computer processors;
2. RAM;
3. Computers with hard disk storage;
4. Computer operating system;
5. Requirement questionnaire;
6. User requirement management tools;
7. BRDs and MRDs.
8. Safety gear</td></tr>
<tr><td colspan="4">EVIDENCE REQUIREMENT</td></tr>
<tr><td colspan="2">PRACTICAL PERFORMANCE</td><td colspan="2">UNDERPINNING KNOWLEDGE</td></tr>
<tr><td colspan="2">The person performing this task must be able to do the following:
1. Interview users and collect key points;
2. Obtain the user's existing business processes;
3. Understand the user's expectations;
4. Understand the user's business and goals;
5. Develop a demand survey plan, outline and questionnaire;
6. Organize and edit the survey results;
7. Interpret user requirements into tangible system features
8. Observe health, occupational and environmental safety, rules and regulations</td><td colspan="2">Detailed knowledge about:
1.0 Methods
The person performing this task must be able to explain how to:
1.1 Collect user requirements;
1.2 Analyse user requirements.

2.0 Principles
The person performing this task must be able to explain the following principles:
2.1 Requirement definition;
2.2 Hierarchy of needs;
2.3 Specifications for demand survey.

3.0 Theories
The person performing this task must be able to explain the following:
3.1 Purpose of requirements;</td></tr>
</table>

	3.2 Types of requirements; 3.3 Techniques for requirement elicitation; 3.4 Methods of requirement modelling. **4.0 Essential Skills** 4.1 Communication and report writing skills; 4.2 Problem analysis and positioning skills; 4.3 Expression skills; 4.4 Excellent interpersonal skills.
DESCRIPTION OF THE END PRODUCT / SERVICE	Collection and analysis of user requirements are completed.
CIRCUMSTANTIAL KNOWLEDGE	**Detailed knowledge about:** 1. Occupational health and safety.

OCCUPATION	SOFTWARE ENGINEER	**OCCUPATION CODE**	
DUTY TITLE	PERFORM USER REQUIREMENT ANALYSIS AND MANAGEMENT	**DUTY NO.**	701
TASK TITLE	CONDUCT USER REQUIREMENT DESIGN	**TASK NO.**	7012
PERFORMANCE CRITERIA	The person performing this task must be able to prepare the guideline according to the computer software requirements specification, and complete the work related to the design of user requirements.		
RANGE STATEMENT	The task can be performed in the workroom or office under the supervision of software engineers. The tools and equipment to be used include: 1. Computer processors; 2. RAM; 3. Computers with hard disk storage; 4. Computer operating system; 5. User requirement management tools; 6. BRDs and MRDs. 7. Safety gear		

EVIDENCE REQUIREMENT	
PRACTICAL PERFORMANCE	**UNDERPINNING KNOWLEDGE**
The person performing this task must be able to do the following: 1. Record key points of user communication; 2. Restate user requirements; 3. Create user portraits; 4. Classify user requirements; 5. Analyse and determine user requirement specifications. 6. Observe health, occupational and environmental safety, rules and regulations	**Detailed knowledge about:** **1.0 Methods** The person performing this task must be able to explain how to: 1.1 Create user portraits; 1.2 Design user requirements. **2.0 Principles** The person performing this task must be able to explain the following principles: 2.1 Definition of requirement elicitation; 2.2 Definition of requirement analysis; 2.3 Quality function deployment specification; 2.4 Software requirements specification; 2.5 Software architecture design. **3.0 Theories** The person performing this task must be able to explain the following: 3.1 Methods for obtaining software requirements; 3.2 Characteristics of requirements; 3.3 Objectives, processes, and methods of requirement analysis;

	3.4 Maslow's hierarchy of needs; 3.5 Kano model. **4.0 Essential Skills** 4.1 Communication and report writing skills; 4.2 Problem analysis and positioning skills; 4.3 Expression skills; 4.4 Excellent interpersonal skills.
DESCRIPTION OF THE END PRODUCT / SERVICE	The original requirement document is created through user requirement analysis and design in combination with the product.
CIRCUMSTANTIAL KNOWLEDGE	**Detailed knowledge about:** 1. Occupational health and safety.

OCCUPATION	SOFTWARE ENGINEER	**OCCUPATION CODE**	
DUTY TITLE	PERFORM USER REQUIREMENT ANALYSIS AND MANAGEMENT	**DUTY NO.**	701
TASK TITLE	WRITE WORK REQUIREMENT DOCUMENT	**TASK NO.**	7013
PERFORMANCE CRITERIA	The person performing this task must be able to complete the work-related requirement document in accordance with the standards for the preparation of the computer software requirements specification.		
RANGE STATEMENT	The task can be performed in the workroom or office under the supervision of software engineers. The tools and equipment to be used include: 1. Computer processors; 2. RAM; 3. Computers with hard disk storage; 4. Computer operating system; 5. User requirement management tools; 6. Requirement document writing software. 7. Safety gear		

EVIDENCE REQUIREMENT

PRACTICAL PERFORMANCE	**UNDERPINNING KNOWLEDGE**
The person performing this task must be able to do the following: 1. Analyse and design the architecture of the development software system; 2. Establish a baseline for system architecture; 3. Build a version functional requirement tree; 4. Establish the requirement document directory structure; 5. Design the way to implement requirements; 6. Validate the requirement analysis results; 7. Write the project requirement documents. 8. Observe health, occupational and environmental safety, rules and regulations	**Detailed knowledge about:** **1.0 Methods** The person performing this task must be able to explain how to: 1.1 Validate requirements; 1.2 Write requirement documents. **2.0 Principles** The person performing this task must be able to explain the following principles: 2.1 Writing of software requirements specification; 2.2 Specification for software requirement document preparation; 2.3 Definition of requirement validation. **3.0 Theories** The person performing this task must be able to explain the following: 3.1 Structure of the software module profile document; 3.2 Methods for detailed design of software modules; 3.3 Role of requirement documents; 3.4 Ideas and methods for writing requirement documents; 3.5 SRS templates;

	3.6 Data dictionary; 3.7 UML. **4.0 Essential Skills** 4.1 Communication and report writing skills; 4.2 Problem analysis and positioning skills; 4.3 Project management competence.
DESCRIPTION OF THE END PRODUCT / SERVICE	The work requirement document is completed.
CIRCUMSTANTIAL KNOWLEDGE	**Detailed knowledge about:** 1. Occupational health and safety.

OCCUPATION	SOFTWARE ENGINEER	**OCCUPATION CODE**	
DUTY TITLE	PERFORM USER REQUIREMENT ANALYSIS AND MANAGEMENT	**DUTY NO.**	701
TASK TITLE	CONDUCT CHANGE MANAGEMENT REQUIREMENTS	**TASK NO.**	7014
PERFORMANCE CRITERIA	The person performing this task must be able to complete the work related to change management requirements in accordance with the standards for the preparation of the computer software requirements specification.		
RANGE STATEMENT	The task can be performed in the workroom or office under the supervision of software engineers. The tools and equipment to be used include: 1. Computer processors; 2. RAM; 3. Computers with hard disk storage; 4. Computer operating system; 5. User requirement management tools; 6. BRDs, MRDs and PRDs. 7. Safety gear		

EVIDENCE REQUIREMENT

PRACTICAL PERFORMANCE	**UNDERPINNING KNOWLEDGE**
The person performing this task must be able to do the following: 1. Control the requirement baseline; 2. Manage the connection between requirements and the chain of connections; 3. Manage the dependency relationship between individual requirements and other deliverables of the project; 4. Track the status of requirements in the baseline; 5. Develop the change control process; 6. Deliver the version update instructions. 7. Observe health, occupational and environmental safety, rules and regulations	**Detailed knowledge about:** **1.0 Methods** The person performing this task must be able to explain how to: 1.1 Manage the change of software function module requirements. **2.0 Principles** The person performing this task must be able to explain the following principles: 2.1 Definition of requirement tracking; 2.2 Definition of requirement changes; 2.3 Basic process of and precautions for requirement changes; 2.4 Requirement change control. **3.0 Theories** The person performing this task must be able to explain the following: 3.1 Requirement tracking methods; 3.2 Requirement tracking tools; 3.3 Methods for handling requirement changes. **4.0 Essential Skills**

	4.1 Communication and report writing skills; 4.2 Problem analysis and positioning skills; 4.3 Excellent interpersonal skills; 4.4 Project management competence.
DESCRIPTION OF THE END PRODUCT / SERVICE	Change requirements are managed in a well-coordinated manner, and a standardized change control process is established.
CIRCUMSTANTIAL KNOWLEDGE	**Detailed knowledge about:** 1. Occupational health and safety.

OCCUPATION	SOFTWARE ENGINEER	**OCCUPATION CODE**	
DUTY TITLE	CARRY OUT SOFTWARE DESIGN	**DUTY NO.**	702
TASK TITLE	CARRY OUT HIGH LEVEL DESIGN OF SOFTWARE	**TASK NO.**	7021
PERFORMANCE CRITERIA	The person performing this task must be able to carry out the high-level design of software in accordance with the software requirements specification document in the CMM standard.		
RANGE STATEMENT	The task can be performed in the workroom or office under the supervision of software engineers. The tools and equipment to be used include: 1. Computer processors; 2. Computers with hard disk storage; 3. Servers that support the system; 4. Computer operating system; 5. Software project design and modelling tools; 6. Office software. 7. Safety gear		

EVIDENCE REQUIREMENT	
PRACTICAL PERFORMANCE	**UNDERPINNING KNOWLEDGE**
The person performing this task must be able to do the following: 1. Analyse the function and operating environment of the software in the system; 2. Determine the purpose and content of the software; 3. Design the overall structure of the software; 4. Divide software function modules and define data interfaces; 5. Define the control processes, data flow, concurrent execution, and other aspects between software modules; 6. Describe user interfaces, external interfaces, and internal interfaces in detail; 7. Assign a unique identifier to each interface; 8. Describe the purpose, development type, input/output data, and resource usage of each module; 9. Analyse and determine the definition of data structure, data	**Detailed knowledge about:** **1.0 Methods** The person performing this task must be able to explain how to: 1.1 Determine the software development and operation environment; 1.2 Divide software function modules and define data interfaces; 1.3 Draw software control flowcharts and data flow diagrams; 1.4 Determine software performance indicators; 1.5 Determine the software data storage structure; 1.6 Handle abnormal inputs. **2.0 Principles** The person performing this task must be able to explain the following principles: 2.1 System requirement analysis; 2.2 System function analysis; 2.3 Function module design; 2.4 Software functions. **3.0 Theories** The person performing this task must be able to explain the following:

variable types, and dynamic allocation/static allocation, and assign a unique identifier to each; 10. Describe the database design and use of the software; 11. Design ways to handle exceptions; 12. Make the requirements in the requirements specification document traceable; 13. Review the high-level design of software; 14. Design output documents. 15. Observe health, occupational and environmental safety, rules and regulations	3.1 Requirement engineering theory; 3.2 Software architecture model; 3.3 Data modelling methods; 3.4 Process modelling methods; 3.5 Design pattern theory; 3.6 System analysis and design methods. **4.0 Essential Skills** 4.1 Communication skills; 4.2 Document writing skills; 4.3 Problem analysis and positioning skills; 4.4 Management skills.
DESCRIPTION OF THE END PRODUCT / SERVICE	The high-level design of software is carried out in accordance with the software requirements specification document, and the high-level design report is prepared accordingly.
CIRCUMSTANTIAL KNOWLEDGE	**Detailed knowledge about:** 1. Occupational health and safety; 2. Confidentiality system; 3. Network security regulations.

OCCUPATION	SOFTWARE ENGINEER	**OCCUPATION CODE**	
DUTY TITLE	CARRY OUT SOFTWARE DESIGN	**DUTY NO.**	702
TASK TITLE	CARRY OUT DETAILED DESIGN OF SOFTWARE	**TASK NO.**	7022
PERFORMANCE CRITERIA	The person performing this task must be able to carry out the detailed design of software in accordance with the software requirements specification and high-level design in the CMM standard.		
RANGE STATEMENT	The task can be performed in the workroom or office under the supervision of software engineers. The tools and equipment to be used include: 1. Computer processors; 2. RAM; 3. Computers with hard disk storage; 4. Servers that support the system; 3. Computer operating system; 4. Software project design and modelling tools; 5. Office software. 6. Safety gear		

EVIDENCE REQUIREMENT	
PRACTICAL PERFORMANCE	**UNDERPINNING KNOWLEDGE**
The person performing this task must be able to do the following: 1. Consult software requirements specifications and software profiles; 2. Design documents; 3. Divide all sub-modules in the module and assign a unique identifier to each; 4. Define the execution control flow between sub-modules, the data flow and all interfaces between sub-modules and external links in the form of icon and text; 5. Analyse and clarify the purpose, function, performance, input/output, algorithm and logical flow design information of sub-modules; 6. Determine the design requirements and all constraints of sub-modules; 7. Use UML class diagrams to clarify the definitions and relationships between major classes for object-oriented programs;	**Detailed knowledge about:** **1.0 Methods** The person performing this task must be able to explain how to: 1.1 Divide sub-modules and their interfaces, and label them; 1.2 Draw control flowcharts and data flow diagrams of each sub-module; 1.3 Determine the functions and core algorithms of sub-modules; 1.4 Determine the performance indicators of sub-modules; 1.5 Handle abnormal inputs. **2.0 Principles** The person performing this task must be able to explain the following principles: 2.1 Function module design; 2.2 System function analysis. **3.0 Theories** The person performing this task must be able to explain the following: 3.1 Approaches to select data structure; 3.2 Data modelling methods;

8. Clarify the implication and operating procedures of each element in the main human-machine interface one by one, including the module and class corresponding to the interface; 9. Clarify the database table structure and table relationships; 10. Implement traceability between modules in the detailed design and those in the high-level design; 11. Determine deployment environment specifications; 12. Review the detailed design output document of the software; 13. Observe health, occupational and environmental safety, rules and regulations	3.3 Process modelling methods; 3.4 Design pattern theory. **4.0 Essential Skills** 4.1 Communication skills; 4.2 Document writing skills; 4.3 Problem analysis and positioning skills; 4.4 Management skills.
DESCRIPTION OF THE END PRODUCT / SERVICE	The detailed design of software is carried out in accordance with the high-level design document, and the detailed design report is prepared accordingly.
CIRCUMSTANTIAL KNOWLEDGE	**Detailed knowledge about:** 1. Occupational health and safety; 2. Confidentiality system; 3. Network security regulations.

OCCUPATION	SOFTWARE ENGINEER	**OCCUPATION CODE**	
DUTY TITLE	PERFORM ENTERPRISE-LEVEL SOFTWARE PROJECT DEVELOPMENT	**DUTY NO.**	703
TASK TITLE	CARRY OUT FRONT-END AND BACK-END SEPARATION PROJECT DEVELOPMENT	**TASK NO.**	7031
PERFORMANCE CRITERIA	The person performing this task must be able to complete the efficient collaborative development of front-end and back-end separation projects in accordance with the specification standards for the development of enterprise-level software projects.		
RANGE STATEMENT	The task can be performed in the workroom or office under the supervision of software engineers. The tools and equipment to be used include: 1. Computer processors; 2. RAM; 3. Computers with hard disk storage; 4. Servers that support the system; 5. Computer operating system; 6. Software project development and compilation tools; 7. Development frameworks and related components; 8. Database management system. 9. Safety gear		

EVIDENCE REQUIREMENT	
PRACTICAL PERFORMANCE	**UNDERPINNING KNOWLEDGE**
The person performing this task must be able to do the following: 1. Comply with the technical specifications for software project development when performing front-end and back-end separation project development; 2. Determine the workload and cycle length of front-end and back-end development based on product requirements; 3. Identify the modules that require joint debugging in the project; 4. Preliminarily negotiate the interface protocol; 5. Write and maintain interface documents, and update them when API changes occur; 6. Select appropriate components and development tools to synchronize the design, coding	**Detailed knowledge about:** **1.0 Methods** The person performing this task must be able to explain how to: 1.1 Evaluate the development cycle of front-end and back-end projects; 1.2 Write API documents in accordance with specifications; 1.3 Design function modules, and encode development and integration software; 1.4 Perform joint debugging and testing for front-end and back-end separation projects; 1.5 Release and maintain product projects. **2.0 Principles** The person performing this task must be able to explain the following principles: 2.1 Core theory of front-end and back-end separation project development;

development, and integration of front-end function pages and back-end service logic modules; 7. Conduct front-end and back-end joint debugging and adjust and update interface documents in real time; 8. Submit the product to experience the requirement functions; 9. Submit the product to testers and work with them to fix corresponding development issues; 10. Develop a review form indicating the procedures for project release; 11. Release the execution code and implement online testing. 12. Observe health, occupational and environmental safety, rules and regulations	2.2 Collaborative agile development; 2.3 Technical specifications for software project development. **3.0 Theories** The person performing this task must be able to explain the following: 3.1 Protocol specifications for front-end and back-end separation project interface documents; 3.2 Methods for preparing and maintaining API documents; 3.3 Functions and applications of components and development tools; 3.4 Joint debugging and testing plan and implementation of front-end and back-end separation projects; 3.5 Front-end and back-end code release procedures and implementation. **4.0 Essential Skills** 4.1 Communication skills; 4.2 Document writing skills; 4.3 Logic development skills; 4.4 Management skills; 4.5 Teamwork skills.
DESCRIPTION OF THE END PRODUCT / SERVICE	The development of front-end and back-end separation projects is completed through front-end and back-end collaboration in accordance with the product requirements of corresponding enterprise-level projects.
CIRCUMSTANTIAL KNOWLEDGE	**Detailed knowledge about:** 1. Page design; 2. Project scheduling; 3. Occupational health and safety; 4. Confidentiality system; 5. Network security regulations.

OCCUPATION	SOFTWARE ENGINEER	**OCCUPATION CODE**	
DUTY TITLE	PERFORM ENTERPRISE-LEVEL SOFTWARE PROJECT DEVELOPMENT	**DUTY NO.**	703
TASK TITLE	CONDUCT NETWORK PROGRAMMING PROJECT DEVELOPMENT	**TASK NO.**	7032
PERFORMANCE CRITERIA	The person performing this task must be able to complete the development of network programming projects in accordance with the specification standards for the development of enterprise-level software projects.		
RANGE STATEMENT	The task can be performed in the workroom or office under the supervision of software engineers. The tools and equipment to be used include: 1. Computer processors; 2. RAM; 3. Computers with hard disk storage; 4. Servers that support the system; 5. Computer operating system; 6. Project development and compilation tools; 7. Development frameworks and related components; 8. Database management system; 9. Server operating system. 10. Safety gear		

EVIDENCE REQUIREMENT	
PRACTICAL PERFORMANCE	**UNDERPINNING KNOWLEDGE**
The person performing this task must be able to do the following: 1. Comply with the technical specifications for software project development when performing network programming project development; 2. Select appropriate development tools; 3. Configure the platform environment; 4. Design service logic modules for client and server according to network programming project requirements; 5. Implement RPC and socket code development based on network protocols; 6. Develop distributed component programs;	**Detailed knowledge about:** **1.0 Methods** The person performing this task must be able to explain how to: 1.1 Design service logic modules for client and server; 1.2 Develop RPC and sockets based on network protocols; 1.3 Develop distributed components; 1.4 Implement Web Service programming; 1.5 Develop heterogeneous network communication programs; 1.6 Implement network information security reinforcement. **2.0 Principles** The person performing this task must be able to explain the following principles: 2.1 Interaction between client and server;

7. Write Web Service programs to respond to network services; 8. Develop heterogeneous network communication programs; 9. Reinforce network information security; 10. Test the functionality of network programming projects; 11. Write technical documentation. 12. Observe health, occupational and environmental safety, rules and regulations	2.2 Network security reinforcement; 2.3 RPC interface usage specifications; 2.4 Technical specifications for project development. **3.0 Theories** The person performing this task must be able to explain the following: 3.1 Working modes of network protocols; 3.2 Core service functions of client and server of network programming projects; 3.3 Working modes of distributed system components; 3.4 Heterogeneous network communication methods; 3.5 Network information security reinforcement technology; 3.6 Configuration of network operating systems; 3.7 Fundamentals of encryption technology. **4.0 Essential Skills** 4.1 Communication skills; 4.2 Document writing skills; 4.3 Management skills; 4.4 Teamwork skills.
DESCRIPTION OF THE END PRODUCT / SERVICE	Relevant development tools are used to complete the development of network programming projects in accordance with the requirements of enterprise-level network programming product projects.
CIRCUMSTANTIAL KNOWLEDGE	**Detailed knowledge about:** 1. Occupational health and safety; 2. Confidentiality system; 3. Network security regulations.

OCCUPATION	SOFTWARE ENGINEER	**OCCUPATION CODE**	
DUTY TITLE	PERFORM ENTERPRISE-LEVEL SOFTWARE PROJECT DEVELOPMENT	**DUTY NO.**	703
TASK TITLE	DEVELOP CLOUD APPLICATION	**TASK NO.**	7033
PERFORMANCE CRITERIA	The person performing this task must be able to complete the development of cloud applications in accordance with the specification standards for the development of enterprise-level software projects.		
RANGE STATEMENT	The task can be performed in the workroom or office under the supervision of software engineers. The tools and equipment to be used include: 1. Computer processors; 2. RAM; 3. Computers with hard disk storage; 4. Cloud servers that support system development; 5. Computer operating system; 6. Public cloud, private cloud or hybrid cloud container environment; 7. Development tools and related frameworks. 8. Safety gear		

EVIDENCE REQUIREMENT	
PRACTICAL PERFORMANCE	**UNDERPINNING KNOWLEDGE**
The person performing this task must be able to do the following: 1. Determine the functional and non-functional requirement of cloud applications; 2. Design descriptions and deployment strategies for cloud applications; 3. Build a cloud development environment; 4. Use development tools to develop the display layer for cloud applications; 5. Use development tools to develop the service layer for cloud applications; 6. Use development tools to develop the storage layer for cloud applications; 7. Review and test codes; 8. Perform load testing and continuous integration; 9. Evaluate security;	**Detailed knowledge about:** **1.0 Methods** The person performing this task must be able to explain how to: 1.1 Design cloud applications; 1.2 Cloud application development; 1.3 Deploy cloud applications; 1.4 Maintain cloud applications; 1.5 Migrate cloud applications. **2.0 Principles** The person performing this task must be able to explain the following principles: 2.1 Service-Oriented Architecture (SOA); 2.2 Cloud-native applications; 2.3 DevOps; 2.4 Security. **3.0 Theories** The person performing this task must be able to explain the following: 3.1 Public cloud, private cloud, and hybrid cloud deployment methods;

10. Identify and fix potential security vulnerabilities; 11. Deploy applications on public cloud, private cloud, or hybrid cloud containers; 12. Track and analyse system logs for continuous monitoring and optimization; 13. Troubleshoot and debug network, storage, software, and external faults; 14. Respond quickly and deal with problems; 15. Migrate and maintain cloud applications according to requirements. 16. Observe health, occupational and environmental safety, rules and regulations	3.2 Microservices architecture theory; 3.3 Automation test, deployment, operation and maintenance theory; 3.4 Horizontal scaling, vertical scaling, and load balancing methods; 3.5 Continuous integration and continuous delivery methods. **4.0 Essential Skills** 4.1 Communication and report writing skills; 4.2 Problem analysis and positioning skills; 4.3 Time management ability; 4.4 Schedule control skills; 4.5 Teamwork skills.
DESCRIPTION OF THE END PRODUCT / SERVICE	The cloud applications are developed in accordance with the specification standards for the development of enterprise-level software projects.
CIRCUMSTANTIAL KNOWLEDGE	**Detailed knowledge about:** 1. Data storage; 2. Webpage front-end; 3. Occupational health and safety; 4. Confidentiality system; 5. Network security regulations; 6. Quality control methods.

OCCUPATION	SOFTWARE ENGINEER	**OCCUPATION CODE**	
DUTY TITLE	CONDUCT SOFTWARE TESTING AND PRODUCT DEBUGGING	**DUTY NO.**	704
TASK TITLE	PERFORM SOFTWARE TESTING	**TASK NO.**	7041
PERFORMANCE CRITERIA	The person performing this task must be able to carry out the software unit testing and integration testing in accordance with the requirements of software testing in the CMM standard.		
RANGE STATEMENT	The task can be performed in the workroom or office under the supervision of software engineers. The tools and equipment to be used include: 1. Computer processors; 2. RAM; 3. Computers with hard disk storage; 4. Servers that support the system; 5. Computer operating system; 6. Software project design and modelling tools; 7. Office software. 8. Safety gear		

EVIDENCE REQUIREMENT	
PRACTICAL PERFORMANCE	**UNDERPINNING KNOWLEDGE**
The person performing this task must be able to do the following: 1. Consult the software requirements specification document and software; 2. Perform detailed design of documents; 3. Develop a software test plan; 4. Prepare software and hardware items for testing; 5. Write test cases; 6. Define the conditions for all tests or a group of tests; 7. Write system integration test cases; 8. Execute test cases; 9. Analyse data record documents; 10. Organize the results; 11. Fill in a software problem report for any software problems that arise; 12. Summarize and submit the problems found during the testing process to the developer for confirmation;	**Detailed knowledge about:** **1.0 Methods** The person performing this task must be able to explain how to: 1.1 Develop a software test plan; 1.2 Write software test cases; 1.3 Execute software test cases; 1.4 Give feedback on software test bugs. **2.0 Principles** The person performing this task must be able to explain the following principles: 2.1 Software test methods; 2.2 Software testing process; 2.3 Software engineering principles. **3.0 Theories** The person performing this task must be able to explain the following: 3.1 Data modelling methods; 3.2 Process modelling methods; 3.3 Unit test methods; 3.4 Integration test methods.

13. Review software testing reports. 14. Observe health, occupational and environmental safety, rules and regulations	**4.0 Essential Skills** 4.1 Communication skills; 4.2 Document writing skills; 4.3 Problem analysis and positioning skills; 4.4 Management skills.
DESCRIPTION OF THE END PRODUCT / SERVICE	Software testing reports are prepared in accordance with the requirements of software testing in the CMM standard..
CIRCUMSTANTIAL KNOWLEDGE	**Detailed knowledge about:** 1. Occupational health and safety; 2. Confidentiality system; 3. Network security regulations.

OCCUPATION	SOFTWARE ENGINEER	**OCCUPATION CODE**	
DUTY TITLE	CONDUCT SOFTWARE TESTING AND PRODUCT DEBUGGING	**DUTY NO.**	704
TASK TITLE	CONDUCT PRODUCT DEBUGGING	**TASK NO.**	7042
PERFORMANCE CRITERIA	The person performing this task must be able to debug products loaded with the software in accordance with the requirements of product debugging in the ISO9001 standard.		
RANGE STATEMENT	The task can be performed in the workroom or debugging workshop under the supervision of system engineers. The tools and equipment to be used include: 1. Computer processors; 2. RAM; 3. Computers with hard disk storage; 4. Servers that support the system; 5. Computer operating system; 6. Software download tools, and product debugging instruments and equipment; 7. Product simulation data source. 8. Safety gear		

EVIDENCE REQUIREMENT

PRACTICAL PERFORMANCE	**UNDERPINNING KNOWLEDGE**
The person performing this task must be able to do the following: 1. Consult software requirement documents, product technical conditions, technical instructions, and other documents; 2. Develop the product debugging outline; 3. Build the hardware and software environment for product debugging; 4. Download or deploy the software to the product; 5. Carry out normal test according to product technical documents; 6. Test the products that operate in a special environment in simulation scenarios; 7. Analyse the test data and record documents, and organize the results; 8. Timely fill in the problem report for any problems found;	**Detailed knowledge about:** **1.0 Methods** The person performing this task must be able to explain how to: 1.1 Write the product debugging outline; 1.2 Carry out normal product debugging; 1.3 Debug the products in special environments; 1.4 Give feedback on product debugging problems. **2.0 Principles** The person performing this task must be able to explain the following principles: 2.1 Product debugging methods; 2.2 Product debugging process; 2.3 Product engineering in the industry. **3.0 Theories** The person performing this task must be able to explain the following: 3.1 Functions and implication of performance indicators of to-be-tested products; 3.2 Test methods for functions and performance indicators of to-be-tested products;

9. Summarize and submit the problems found during the testing process to the developer for confirmation. 10. Observe health, occupational and environmental safety, rules and regulations	3.3 Unit test methods; 3.4 Integration test methods. **4.0 Essential Skills** 4.1 Communication skills; 4.2 Document writing skills; 4.3 Problem analysis and positioning skills; 4.4 Management skills.
DESCRIPTION OF THE END PRODUCT / SERVICE	The products debugging is completed in accordance with the product specification document.
CIRCUMSTANTIAL KNOWLEDGE	**Detailed knowledge about:** 1. Occupational health and safety; 2. Confidentiality system; 3. Network security regulations.

OCCUPATION	SOFTWARE ENGINEER	**OCCUPATION CODE**	
DUTY TITLE	CARRY OUT PRODUCT DELIVERY	**DUTY NO.**	705
TASK TITLE	DEVELOP USER MANUAL AND TECHNICAL DOCUMENTATION	**TASK NO.**	7051
PERFORMANCE CRITERIA	The person performing this task must be able to complete the work related to the development of user manual and technical documentation in accordance with the specification standards for the delivery of software project products.		
RANGE STATEMENT	The task can be performed in the workroom or office under the supervision of software engineers. The tools and equipment to be used include: 1. Computer processors; 2. RAM; 3. Computers with hard disk storage; 4. Computer operating system; 5. Requirement document writing software. 6. Safety gear		

EVIDENCE REQUIREMENT

PRACTICAL PERFORMANCE	**UNDERPINNING KNOWLEDGE**
The person performing this task must be able to do the following: 1. Analyse user characteristics and usage scenarios; 2. Foresee potential problems that users may encounter and solutions thereto; 3. Plan document layout; 4. Pay attention to user feedback; 5. Optimize document content. 6. Observe health, occupational and environmental safety, rules and regulations	**Detailed knowledge about:** **1.0 Methods** The person performing this task must be able to explain how to: 1.1 Develop user manual and technical documentation. **2.0 Principles** The person performing this task must be able to explain the following principles: 2.1 Definition of user manual and technical documentation; 2.2 Specifications for preparing the user manual and technical documentation. **3.0 Theories** The person performing this task must be able to explain the following: 3.1 Methods for writing the user manual and technical documentation. **4.0 Essential Skills** 4.1 Communication skills; 4.2 Problem analysis and positioning skills; 4.3 Project management competence.

DESCRIPTION OF THE END PRODUCT / SERVICE	User manual and technical documentation are completed.
CIRCUMSTANTIAL KNOWLEDGE	**Detailed knowledge about:** 1. Occupational health and safety; 2. Scope of duties; 3. Confidentiality system.

OCCUPATION	SOFTWARE ENGINEER	**OCCUPATION CODE**	
DUTY TITLE	CARRY OUT PRODUCT DELIVERY	**DUTY NO.**	705
TASK TITLE	CARRY OUT SOFTWARE PROJECT DEPLOYMENT AND DELIVERY	**TASK NO.**	7052
PERFORMANCE CRITERIA	The person performing this task must be able to complete the work related to the deployment and delivery of software projects in accordance with the specification standards for the delivery of software project products.		
RANGE STATEMENT	The task can be performed in the workroom or office under the supervision of software engineers. The tools and equipment to be used include: 1. Computer processors; 2. RAM; 3. Computers with hard disk storage; 4. Computer operating system; 5. Software deployment tools; 6. Software delivery tools; 7. Automated deployment tools; 8. Software system deployment plan templates; 9. Software delivery list. 10. Safety gear		

EVIDENCE REQUIREMENT	
PRACTICAL PERFORMANCE	**UNDERPINNING KNOWLEDGE**
The person performing this task must be able to do the following: 1. List updates; 2. List deployment steps; 3. List the matters to be dealt with; 4. Take note of update records; 5. Notify business personnel to validate; 6. Notify business personnel of the update plan; 7. Notify business personnel of the update results; 8. Submit the products stipulated in the contract, such as the developed target installer, database data dictionary, *User Installation Manual*, *User Guide*, requirement report, design report, and test report; 9. Guide customers to install software;	**Detailed knowledge about:** **1.0 Methods** The person performing this task must be able to explain how to: 1.1 Deploy software projects; 1.2 Deliver software projects. **2.0 Principles** The person performing this task must be able to explain the following principles: 2.1 Software deployment steps and precautions; 2.2 Standards for software delivery; 2.3 Software delivery content and process; 2.4 Principles of software delivery. **3.0 Theories** The person performing this task must be able to explain the following: 3.1 Software deployment methods; 3.2 Technical implementation elements of software deployment;

10. Guide customers on the actual operation methods and usage process of the software. 11. Observe health, occupational and environmental safety, rules and regulations	3.3 Definition of software delivery; 3.4 Definition of software deployment; 3.5 Software development and delivery modes. **4.0 Essential Skills** 4.1 Project management competence; 4.2 Communication skills.
DESCRIPTION OF THE END PRODUCT / SERVICE	An installation package that allows users to deploy software products to their work environment through simple execution is created by the development team.
CIRCUMSTANTIAL KNOWLEDGE	**Detailed knowledge about:** 1. Occupational health and safety; 2. Scope of duties.

OCCUPATION	SOFTWARE ENGINEER	**OCCUPATION CODE**	
DUTY TITLE	CARRY OUT PRODUCT DELIVERY	**DUTY NO.**	705
TASK TITLE	PROVIDE AFTER-SALES TECHNICAL SUPPORT	**TASK NO.**	7053
PERFORMANCE CRITERIA	The person performing this task must be able to complete the work related to providing after-sales technical support in accordance with the specification standards for the delivery of software project products.		
RANGE STATEMENT	The task can be performed in the workplace or office under the supervision of software engineers. The tools and equipment to be used include: 1. Computer processors; 2. RAM; 3. Computers with hard disk storage; 4. Computer operating system. 5. Safety gear		

EVIDENCE REQUIREMENT	
PRACTICAL PERFORMANCE	**UNDERPINNING KNOWLEDGE**
The person performing this task must be able to do the following: 1. Provide product after-sales consulting services; 2. Provide product technical support services; 3. Guide user requirements and propose suggestions and solutions; 4. Demonstrate products; 5. Develop product requirement strategies; 6. Collect customers' comments and suggestions on the product; 7. Understand and track the use of products by customers; 8. Collect new customer requirements. 9. Observe health, occupational and environmental safety, rules and regulations	**Detailed knowledge about:** **1.0 Methods** The person performing this task must be able to explain how to: 1.1 Provide after-sales technical support to meet requirements. **2.0 Principles** The person performing this task must be able to explain the following principles: 2.1 Acceptance and continuous support. **3.0 Theories** The person performing this task must be able to explain the following: 3.1 Product after-sales consulting and requirement tracking; 3.2 Work content of after-sales technical support. **4.0 Essential Skills** 4.1 Communication and report writing skills; 4.2 Problem analysis and positioning skills; 4.3 Excellent interpersonal skills; 4.4 Expression skills; 4.5 Project management competence.

DESCRIPTION OF THE END PRODUCT / SERVICE	After-sales technical support is provided in accordance with the specification standards for the delivery of software project products.
CIRCUMSTANTIAL KNOWLEDGE	**Detailed knowledge about:** 1. Occupational safety and health.

APPENDIX: DACUM CHART FOR SOFTWARE ENGINEER – NTA LEVEL 7

DUTIES	TASKS	ENABLERS
1.0 Perform user requirement analysis and management	1.1 Conduct user requirement collection and analysis. 1.2 Conduct user requirement design. 1.3 Write work requirement document. 1.4 Conduct change management requirements.	**General skills and knowledge** · Hierarchy of needs · Quality function deployment · Requirement elicitation · Requirement analysis · Software requirements specification · Requirement validation · Requirement tracking · Requirement changes · Software requirement document preparation knowledge · Communication and report writing skills · Problem analysis and positioning skills · Expression skills · Project management competence **Tools and equipment** · Computer processors · RAM · Computers with hard disk storage · Computer operating system · User requirement management tools · Virtual machine environments · Development tools and related frameworks · Requirement document writing software **Materials** · BRDs, MRDs and PRDs. **Requirements for employees** · Teamwork spirit · Integrity · Excellent interpersonal skills · Time management
2.0 Carry out software design	2.1 Carry out high level design of software. 2.2 Carry out detailed design of software.	**General skills and knowledge** · Common development frameworks · Object-oriented analysis and design, and design patterns · Use of design tools such as UML

DUTIES	TASKS	ENABLERS
		· Commonly-used task operating systems, real-time operating systems and embedded operating systems · Container deployment skills · Common interface protocols · Common UI design tools · Database application · Software engineering process · Document preparation **Tools and equipment** · Computer processors · RAM · Computers with hard disk storage · Development tools such as eclipse/ide/WebStorm, qt, and vs · Databases, such as MySQL, sqlserver, and oracle · Commonly-used computer operating systems · Virtual machine environments · Related framework dependencies **Requirements for employees** · Professional dedication · Teamwork spirit · Strong learning ability · Communication skills · Good occupational quality · Strong sense of self-improvement · Thirst for knowledge · Adeptness in learning and applying new knowledge · Good coding practices and writing habits
3.0 Perform enterprise-level software project development	3.1 Carry out front-end and back-end separation project development. 3.2 Conduct network programming project development. 3.3 Develop cloud application.	**General skills and knowledge** · Reliability design methods for network applications · Data structure and algorithms · Front-end webpage component development methods · Back-end database application optimization methods

DUTIES	TASKS	ENABLERS
		· Server component development methods · User interface component design methods · Methods for network programming in heterogeneous network environment · Characteristics and development methods of distributed component programs · Enterprise-level software project coding · Enterprise-level software design and development modes · Distributed cloud-based software development · Development of enterprise virtual network functions · Team communication and collaboration skills · Problem analysis and positioning skills **Tools and equipment** · Programming devices · Computer processors · RAM · Computers with hard disk storage · Commonly-used operating systems, such as Windows, Mac, and Linux · Virtual machine environments · Development tools and related frameworks **Requirements for employees** · Teamwork spirit · Integrity · Time management
4.0 Conduct software testing and product debugging	4.1 Perform software testing.	**General skills and knowledge** · Common programming languages · Use of common operating systems · Common front-end technologies · Database programming · Test theory and techniques · Use of various black-box test case design methods

DUTIES	TASKS	ENABLERS
		· Use of tools for performance test, stress test, and function test · Use of version control systems · Software engineering process · Report design · Document preparation **Tools and equipment** · Programming devices · Computer processors · RAM · Computers with hard disk storage · Testing tools such as LoadRunner and uft · Databases, such as MySQL, sqlserver, and oracle · Commonly-used operating systems, such as Windows and Linux · Virtual machine environments **Requirements for employees** · Professional dedication · Teamwork spirit · Strong learning ability · Communication skills · Good occupational quality · Strong sense of self-improvement · Thirst for knowledge · Adeptness in learning and applying new knowledge · Good scripting practices and habits
	4.2 Conduct product debugging.	**General skills and knowledge** · Common task operating systems, real-time operating systems and embedded operating systems · Basic network knowledge · Knowledge of computer hardware interfaces · Test theory and techniques · Use of various black-box test case design methods · Database operation and configuration · Common troubleshooting capabilities · Product debugging process

DUTIES	TASKS	ENABLERS
		· Product debugging **Tools and equipment** · Programming devices · Computer processors · RAM · Computers with hard disk storage · Databases, such as MySQL, sqlserver, and oracle · Commonly-used computer operating systems, such as Windows and Linux · Software download tools **Materials** · Product simulation data **Requirements for employees** · Professional dedication · Teamwork spirit · Strong learning ability · Communication skills · Good occupational quality · Strong sense of self-improvement · Thirst for knowledge · Adeptness in learning and applying new knowledge
5.0 Carry out product delivery	5.1 Develop user manual and technical documentation. 5.2 Carry out software project deployment and delivery. 5.3 Provide after-sales technical support.	**General skills and knowledge** · Knowledge related to software project deployment · Knowledge related to software project delivery · Acceptance and continuous support · Communication and report writing skills · Problem analysis and positioning skills · Expression skills · Project management competence **Tools and equipment** · Computer processors · RAM · Computers with hard disk storage · Commonly-used operating systems, such as Windows, Mac, and Linux

DUTIES	TASKS	ENABLERS
		· Virtual machine environments · Development tools and related frameworks **Materials** · User manual and technical documentation development **Requirements for employees** · Teamwork spirit · Integrity · Interpersonal skills · Time management ability