

Assessment Evidence Guide

For “IoT Associate Engineer”

**Level-5
(Summative Assessment)**

Dec 2021



**National Vocational & Technical
Training Commission**

Title of Qualification: Level 5 National Qualification Certificate, in Internet of Things (IoT Associate Engineer)	CS Code:	Level: 5	Version: 01
Competency Standard Title: Identify and interface long range wireless technologies (Lora, NB IoT, MTC) with Microcontroller Apply MQTT, CoAP, HTTP on IoT nodes Configure IoT gateways (Wi-Fi/Lora/NB- IoT) Setup cloud server Develop Program in Python Deploy hardware protection Perform software protection Implement cryptography and network security	Assessment Date (DD/MM/YY): Assessment Time: 05 Hours		

Candidate Details	Name: Registration/Roll Number:
Guidance for Candidate	<p>To meet this standard, you are required to complete the following within the given time frame (for practical demonstration & assessment):</p> <p>Assessment Task 1: Candidate is required to setup a small weather station including temperature, and humidity sensor connect these sensors with long range sensor and test its reading wirelessly on device using browser.</p> <p>Assessment Task 2: Candidate is required to store data from IoT device (Temperature) to cloud server (IoT Hub) and secure data using encryption techniques.</p> <p>And complete:</p> <ol style="list-style-type: none"> Knowledge assessment test (Written or Oral) Portfolios at the time of assessment (if any)
Minimum Evidence Required	<p>During a practical assessment, under observation by an assessor, you will complete:</p> <p>Assessment Task 1</p> <p>Performance Criteria 1: Install the required library on Raspberry Pi IDE</p> <p>Performance Criteria 2: Setup a Raspberry Pi</p> <p>Performance Criteria 3: Configure LoRa settings</p>

	<p>Performance Criteria 4: Establish LoRa communication between two different modules</p> <p>Performance Criteria 5: Transmit and receive application data</p> <p>Performance Criteria 6: Install any required software and configure settings</p> <p>Performance Criteria 7: Setup network interface</p> <p>Performance Criteria 8: Establish MQTT Client</p> <p>Performance Criteria 9: Establish MQTT broker</p> <p>Performance Criteria 10: Use MQTT built-in libraries for MQTT communication in application codes</p> <p>Performance Criteria 11: Run a MQTT C client to read the data from broker.</p> <p>Performance Criteria 12: Run another MQTT client to send data to the MQTT broker on the cloud</p> <p>Performance Criteria 13: Confirm data receipt on gateway from multiple end nodes</p> <p>Assessment Task 2:</p> <p>Performance Criteria 1: Set up cloud account</p> <p>Performance Criteria 2: Launch cloud tool for required application</p> <p>Performance Criteria 3: Assign resources to host</p> <p>Performance Criteria 4: Install the required application as per instruction</p> <p>Performance Criteria 5: Determine the requirement and specification for your network</p> <p>Performance Criteria 6: Select the management tool to manage resources as per instruction</p> <p>Performance Criteria 7: Prepare IoT against Wi-Fi intruder attack</p> <p>Performance Criteria 8: Apply AES/TKIP on IoT gateway</p> <p>Performance Criteria 9: Apply MAC address filtering</p> <p>Performance Criteria 10: Isolate IoT devices by securing device to device communication through wireless PAN protocols</p> <p>Performance Criteria 11: Install OpenSSL library on server and client side.</p> <p>Performance Criteria 12: Secure cloud and IoT device connection by applying SSL</p> <p>Performance Criteria 13: Secure communication from device to gateway by encryption protocols</p>
--	--

	<p>Portfolios required at the time of assessment (if any) for</p> <p>Performance Criteria 1: Diary log or any other evidence of work completed on Identify and interface long range wireless technologies (Lora, NB IoT, MTC) with Microcontroller</p> <p>Performance Criteria 1: Diary log or any other evidence of work completed on Apply MQTT, CoAP, HTTP on IoT nodes</p> <p>Performance Criteria 1: Diary log or any other evidence of work completed on Configure IoT gateways (Wi-Fi/Lora/NB-IoT)</p> <p>Performance Criteria 1: Diary log or any other evidence of work completed on Setup cloud server</p> <p>Performance Criteria 1: Diary log or any other evidence of work completed on Develop Program in Python</p> <p>Performance Criteria 1: Diary log or any other evidence of work completed on Deploy hardware protection</p> <p>Performance Criteria 1: Diary log or any other evidence of work completed on Perform software protection</p> <p>Performance Criteria 1: Diary log or any other evidence of work completed on Implement cryptography and network security</p>
--	---

Continued on following page

Assessors Judgment Guide (to be completed by the Assessor and signed both by the assessor and the candidate after the assessment)

Candidate Details	Name: Registration/Roll Number: Candidate Signature:.....
Assessment Outcome	COMPETENT <input type="checkbox"/> NOT YET COMPETENT <input type="checkbox"/> Name of the Assessor: Assessor's code: Signature of the Assessor:

Assessment Summary (to be filled by the assessor)							
Activity	Method					Result	
Nature of Activity	Written	Oral	Observation	Portfolio	Role Play	Competent	Not Yet Competent
Practical Skill Demonstration			✓				
Knowledge Assessment	✓	✓					
Other Requirement							

Each Assessment Task (with performance criteria)				
Assessment Task 1		Description of assessment task 1 Candidate is required to setup a small weather station including temperature, and humidity sensor connect these sensors with long range sensor and test its reading wirelessly on device using browser.		
During the practical assessment, candidate demonstrated the following:		Yes	No	Remarks
1	Install the required library on Raspberry Pi IDE			
2	Setup a Raspberry Pi			
3	Configure LoRa settings			
4	Establish LoRa communication between two different modules			
5	Transmit and receive application data			
6	Install any required software and configure settings			
7	Setup network interface			
8	Establish MQTT Client			
9	Establish MQTT broker			
10	Use MQTT built-in libraries for MQTT communication in application codes			
11	Run a MQTT C client to read the data from broker.			
12	Run another MQTT client to send data to the MQTT broker on the cloud			
13	Confirm data receipt on gateway from multiple end nodes			
Competent <input type="checkbox"/>		Not Yet Competent <input type="checkbox"/>		

Each Assessment Task (with performance criteria)				
Assessment Task 2		Description of assessment task 2		
		Candidate is required to store data from IoT device (Temperature) to cloud server (IoT Hub) and secure data using encryption		
During the practical assessment, candidate demonstrated the following:		Yes	No	Remarks
1	Set up cloud account			
2	Launch cloud tool for required application			
3	Assign resources to host			
4	Install the required application as per instruction			
5	Determine the requirement and specification for your network			
6	Select the management tool to manage resources as per instruction			
7	Prepare IoT against Wi-Fi intruder attack			
8	Apply AES/TKIP on IoT gateway			
9	Apply MAC address filtering			
10	Isolate IoT devices by securing device to device communication through wireless PAN protocols			
11	Install OpenSSL library on server and client side.			
12	Secure cloud and IoT device connection by applying SSL			
13	Secure communication from device to gateway by encryption protocols			
Competent <input type="checkbox"/>		Not Yet Competent <input type="checkbox"/>		

KNOWLEDGE ASSESSMENT

Title of Qualification: Level 5 National Qualification Certificate, in Internet of Things (IoT Associate Engineer)	CS Code:	Level: 5	Version: 01
Competency Standard Title: Identify and interface long range wireless technologies (Lora, NB IoT, MTC) with Microcontroller Apply MQTT, CoAP, HTTP on IoT nodes Configure IoT gateways (Wi-Fi/Lora/NB- IoT) Setup cloud server Develop Program in Python Deploy hardware protection Perform software protection Implement cryptography and network security	Assessment Date (DD/MM/YY): Assessment Time: 30min		

Guidance for Candidate	To complete your assessment for this Competency Standard, you need to answer the questions on the following pages successfully.
------------------------------	--

Candidate Details	Name:Registration/Roll Number: Candidate Signature:.....
Written Assessment Outcome	COMPETENT <input type="checkbox"/> NOT YET COMPETENT <input type="checkbox"/> Name of the Assessor:Assessor's code: Signature of the Assessor:.....

Questions (Candidate confidently answered questions correctly and demonstrated understanding of the topics and their application)	
Define background threads?	
What is functionality of DBMS?	
Define database?	
What is Infrastructure as a service (IaaS) in cloud computing?	
What is Mobile "backend" as a service (MBaaS) in cloud computing?	
What is Duck typing?	
What are the Libraries in python programming language?	
What Risks and vulnerabilities of Cloud Computing have?	
What is Prevent access type of security?	

Questions (Candidate confidently answered questions correctly and demonstrated understanding of the topics and their application)

Why Cryptographic primitives important?

ANSWER KEY

Sr.	Answers
1	A background thread is a thread that runs behind the scenes, while the foreground thread continues to run. For instance, a background thread may perform calculations on user input while the user is entering information using a foreground thread.
2	A database management system is a software tool that makes it possible to organize data in a database. It is often referred to by its acronym, DBMS. The functions of a DBMS include concurrency, security, backup and recovery, integrity and data descriptions
3	A database is an organized collection of structured information, or data, typically stored electronically in a computer system
4	"Infrastructure as a service" (IaaS) refers to online services that provide high-level APIs used to abstract various low-level details of underlying network infrastructure like physical computing resources, location, data partitioning, scaling, security, backup, etc.
5	In the mobile "backend" as a service (m) model, also known as backend as a service (BaaS), web app and mobile app developers are provided with a way to link their applications to cloud storage and cloud computing services with application programming interfaces (APIs) exposed to their applications and custom software development kits (SDKs).
6	Duck typing in computer programming is an application of the duck test—"If it walks like a duck and it quacks like a duck, then it must be a duck"
7	Python's large standard library, commonly cited as one of its greatest strengths, provides tools suited to many tasks. For Internet-facing applications, many standard formats and protocols such as MIME and HTTP are supported. It includes modules for creating graphical user interfaces, connecting to relational

	databases, generating pseudorandom numbers, arithmetic with arbitrary-precision decimals, manipulating regular expressions, and unit testing.
8	While cloud computing is on the cutting edge of information technology there are risks and vulnerabilities to consider before investing fully in it. Security controls and services do exist for the cloud but as with any security system they are not guaranteed to succeed. Furthermore, some risks extend beyond asset security and may involve issues in productivity and even privacy as well.
9	The primary purpose of these types of systems is to restrict and often to completely prevent access to computers or data except to a very limited set of users. The theory is often that if a key, credential, or token is unavailable then access should be impossible. This often involves taking valuable information and then either reducing it to apparent noise or hiding it within another source of information in such a way that it is unrecoverable.
10	Much of the theoretical work in cryptography concerns cryptographic primitives algorithms with basic cryptographic properties and their relationship to other cryptographic problems. More complicated cryptographic tools are then built from these basic primitives. These primitives provide fundamental properties, which are used to develop more complex tools called cryptosystems or cryptographic protocols, which guarantee one or more high-level security properties.