Set B

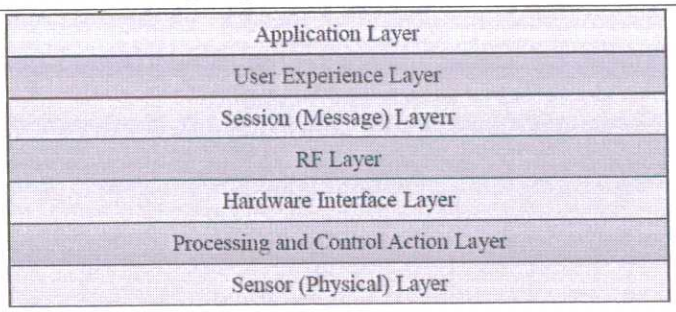
Scoring Indicators

COURSENAME: INTERNET OF THINGS

COURSECODE: 6131A

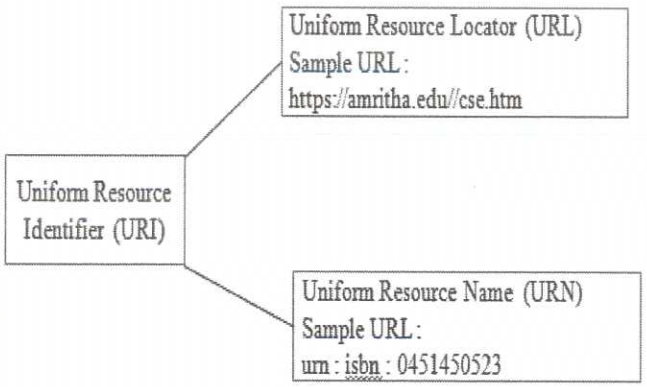
QID : 2102240128

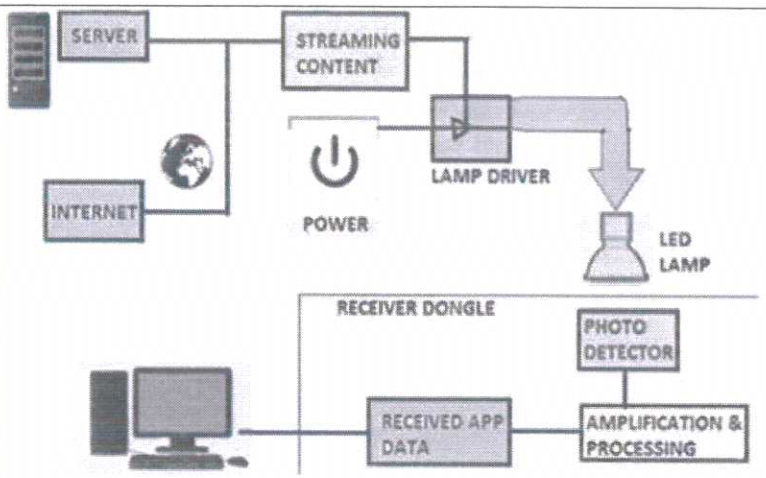
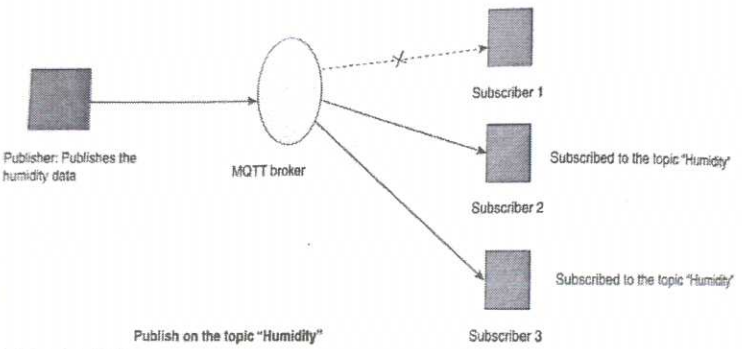
Q No	Scoring Indicators	Split score	Sub Total	Total score
	PARTA			9
I.1	▪ IoT refers to the interconnection via the Internet of computing devices embedded in everyday objects, enabling them to send and receive data. ▪ IoT can also define as the analysis of data to generate a meaningful action, triggered subsequently after the interchange of data.	1	1	9
I.2	Security/personal safety, privacy, data extraction with consistency from complex environment, connectivity, power requirements, complexity involved, storage **(any two)**	½ x 2	1	
I.3	IPV4 and IPV6	1	1	
I.4	128 bits (16 byte)	1	1	
15	Infrastructure as a service	1	1	
I.6	Software Security, Infrastructure Security, Storage Security, Network Security (Any two)	½ x 2	1	
I.7	An actuator is a device that converts energy into physical motion. An actuator is a component that can move or control a mechanism or system.	1	1	
I.8	Arduino, NodeMCU, ESP32, Raspberry PI	½ x 2	1	
I.9	Gas sensor, Obstacle sensor, Heartbeat Sensor, Ultra Sonic Sound Sensor etc	½ x 2	1	
	PART B			24
II.1	▪ Connectivity ▪ Intelligence and identity ▪ Scalability ▪ Dynamic and self-adapting (Complexity) ▪ Architecture ▪ Safety	½ x 6	3	3
II. 2	" THINGS = HARDWARE + SOFTWARE + SERVICE " ▪ Things in IoT refers to a variety of devices. ▪ Thing in a network can monitor/measure. Eg temperature sensor ▪ Things are capable of exchanging data with other connected devices in system. ▪ Data could be stored in cloud, processed there and a control action could be initiated and the devices involved in getting this accomplished is known as things. ▪ Eg: Industrial motors, wearables (watch), vehicles, shoes,	3	3	3

	heart monitoring implants etc			
II.3	 Explain any two	1 ½ x 2	3	3
II. 4	▪ IPv6 was developed to overcome the difficulties faced with IPv4 address allocations. ▪ Named as the next generation protocol for the Internet ▪ Works in layer 3 ▪ It has more addresses and multiple features than IPv4 ▪ IPv6 provides 2^{128} unique IP addresses. ▪ IPv6 addresses are 128 bits long, written in hexadecimal and separated by colons ▪ Eg: 3ffe:1900:4545:3:200:f8ff:fe21:67cf. ▪ Colons separate 16 bit fields. ▪ Leading zeros can be omitted in each field. Eg: 0003 is written as 3 ▪ A double colon :: can be used once in an address to replace multiple fields of zeros ▪ Ex: fe80:0:0:0:200:f8ff:fe21:67cf can be written as fe80::200:f8ff:fe21:67cf ▪ The two colons tell the operating system that everything in between is a zero	3	3	3
II. 5	• It is license free and hence does not add any costing related overhead to the system. • There is no restriction with respect to manufacturers. • BLE modules are inexpensive and much affordable. • Small in size and they do not occupy much space on board. • Power consumption is minimum. • The range offered by BLE is much higher.	3	3	3
II. 6	• The data and data confidentiality are the essential parameters for choosing cloud. • Based on the data being worked on, cloud is categorized into three deployment models 1. Public Cloud Deployment 2. Private Cloud Deployment 3. Hybrid Cloud Deployment (Explain each)	3	3	3
II. 7	• With fog computing it is possible to analyse the data at a place that is closer to where it was generated. • This prevents from sending all data to cloud and respond much faster since data travel is reduced considerably • It thus becomes a reality to process the data in milliseconds time frame. • Fog is below cloud, which is closer to the elements that	3	3	3

	generate data. • After analysis data stored is pushed into the cloud. • Sensors/devices generate data and transmit it to the middle layer which is very close to the data source. • The nodes in the middle layer are capable of handling the data • This requires minimum power and lesser resources. • All data need not go to the cloud at the instant it is generated • Sensitive data gets processed very fast which results in an instant response. • Fog is not meant for hefty (strong) storage. • It is still cloud that does the task of storing big data • Fog is just an intermediary layer that enables faster data processing , thereby facilitating faster response time			
II. 8	1. Data collection 2. Environmental monitoring 3. Asset tracking and Management 4. Health monitoring 5. Predictive Maintenance 6. Energy Management 7. Security and Surveillance 8. Smart agriculture 9. Consumer Electronics etc (explain any 3)	1 x 3	3	3
II. 9	Int, List, Dictionary, Bool, Str, Set, Tuple, Float etc (explain any 3)	1 x 3	3	3
II. 10	• The operating voltage is 5V. • The recommended input voltage will range from 7V to 12V • 14 Digital input/output pins. • 6 Analog input pins. • DC current for each input/output pin is 40mA • Flash Memory size is 32KB • SRAM size is 2KB • 1KB EEPROM. • The USB interface is also used to provide power supply to the board. USB enables the board to acts as a serial device. • The microcontroller used is ATMega328. • The board has also one LED fitted inboard to make the debugging process easy and help to find the bugs in the code. • The board has also one reset button that helps to restart the program using the board. (write any 3)	1 x 3	3	3
				42
III.	**Level 1** ▪ Level 1 IoT has minimum complexity and easy to build. ▪ The Level 1 IoT has normally one sensor. (Temperature / Pressure Sensor) ▪ The data sensed is stored locally and the data analysis is	7	7	7

	done locally. ▪ Data generated in this level is not a big data. ▪ All the controls happens through the internet. Eg: Smart home **Level 2** ▪ It is more complex than Level 1. ▪ Data are huge, hence cloud storage is preferred. ▪ Frequency of sensing is faster. ▪ Analysis is carried out locally. ▪ Control action can be triggered through web/mobile applications. ▪ Eg: Smart factory **Level 3** ▪ Data is huge/voluminous. ▪ Frequency of sensing done by the sensor is faster. ▪ Data stored in the cloud. ▪ Analysis is carried out on the cloud. ▪ The control action can be triggered trough the web/mobile application. ▪ Eg: Agriculture application **Level 4** ▪ Multiple nodes are present, which are independent of each other. ▪ These nodes uploads data to the cloud. ▪ Analysis also carried out in the cloud. Eg: Courier tracking system. **Level 5** ▪ The amount of data is extensive and sensed much faster. ▪ Multiple nodes are involved and are independent of each other. ▪ Sensing of data and storage is cloud based. ▪ The application is completely cloud oriented and it is computationally intensive in real time.			
IV	▪ Acquiring or sensing data ▪ Analyzing or processing data ▪ Taking control action ▪ Enhancing security or privacy (Briefly Explain)	7	7	7
V	. **Constrained Application Protocol (CoAP)** ▪ It is a specialized Internet Application Protocol for constrained devices. ▪ It is a lightweight protocol. ▪ It is designed by IETF (Internet Engineering Task Force) to work in constrained environments ▪ It is used to enable the smart devices to connect to the internet. ▪ It is a one-to-one communication protocol. ▪ It uses lesser resources than HTTP. ▪ It is a simplification of the HTTP protocol running on UDP and is connectionless. ▪ The CoAP has a Client – Server model architecture. ▪ The client will send a request followed by server's	7	7	7

	response to the request with an appropriate reply. ▪ CoAP is based on the REST API model known as RESTful (Representational State Transfer). ▪ REST ensures a secure, fault-tolerant and scalable system. ▪ The CoAP optimizes the datagram length. ▪ It also support multicast. ▪ CoAP is a two-layered protocol. ▪ Lower layer is the message layer and the upper layer owns the request-response process. ▪ The upper layer is dependent on UDP.			
VI	• It is a resource identifier protocol. • Uniform Resource Identifier(URI) is a sequence of characters used to identify logical resources. • All guidelines with respect to URI are issued by the IETF (Internet Engineering Task Force) • URI has Uniform Resource Name(URN)/ Uniform Resource Locator(URL) in it • URIs try to identify a resource and location by accessing its address and name through a primary access mechanism • URN defines an item's identity whereas URL renders a method for finding it • URL contains information on how to fetch or acquire a resource from its location • URL always begin with a protocol(http). • It will have the details of host name and path. • A URL is used when a client raises a request to the server for the service. • Uniform Resource Name (URN) gives the name of a resource and its identification  Sample URL  	Figure – 3 Explanation – 4	7	7

VII	 • Li – Fi needs a light source to transfer data. • It needs modified LED bulbs/lights, which can transmit data. • Led is preferred, because it is a semiconductor device and will have switching characteristics. • Li – Fi is constructed with many components which start with a modified LED bulb. • Data is transmitted over Li-Fi by modulating the intensity of light. • The light is received by the photodetector and demodulation(process) happens to generate the data stream. • All the LED lamps need an LED lamp driver. • The driver gets information from the server and encoding occurs. • After that, LED illuminates(flickers). • The photodetector will be able to read this and convert it into data after amplification.	Figure - 3 marks Explanation – 4 Marks	7	7
VIII	• Lightweight protocol. • It demands minimal resources for its functioning. • It does not require additional resources from its working environment. • IoT prefers MQTT due to resource constraints. • It follows the publish – subscribe pattern.  <u>MQTT Client</u> ▪ Collect information from sensors. ▪ Connect to the messaging server (broker) ▪ Topic is used to publish this message to let other clients	Figure-3 Explanation – 4	7	7

	understand. MQTT Broker ▪ Protocol is implemented in this case. ▪ Mediate and facilitate the data based on interest of the subscriber.			
IX	• There are many parameters and considerations to select the cloud service provider. • The following are some of the criteria for the considerations : 1. Certification and Standards Compliance 2. Financial Health of the Service Provider 3. Business and Technology Strength 4. Compliance Audit 5. Service Level Agreements 6. Reporting/Tracking 7. Costing and Billing 8. Maintenance Monitoring and Upgrade 9. Support 10. Security (Explain each)	7	7	7
X	The security of any computing platform including cloud computing depends on the following : Software Security Infrastructure Security Storage Security Network Security **1. Software Security** • If there are defects created/generated during the development phase, it is a software security threat. • Defects such as software implementation defects, memory allocation, design issues, and exception handling all contribute to security issues. • Care should be taken to write software for IoT without errors/defects. • It can be ensured by complete and comprehensive testing carried out at all stages **2. Infrastructure Security** • Ensure the infrastructure provided by CSP is safe. • The entire data is stored and is dependent on this infrastructure • The third party involved in infrastructure is to check the security vulnerabilities with the infrastructure. • All infrastructure related guidelines should be mentioned clearly in the agreements and should be transparent to the customer. • If data is damaged, everything is damaged and lost, hence care should be taken to protect the infrastructure. **3. Storage Security** • It is important to be in formed of who owns the data and the location where it is stored • Data leak, snooping, malware attacks, etc are all threats to the stored data and can be listed under	7	7	7

	storage security. • Appropriate antivirus software and periodic monitoring, should help protect data **4. Network Security** • Data is stored in the cloud via the Internet and hence all network threats become a possibility			
XI	**1. CPU** • It is the brain and carries out instructions using logical and mathematical operations. In raspberry Pi, the ARM11 series processor is used as CPU. **2. HDMI Port** • Raspberry Pi board has an HDMI or High Definition Multimedia Interface port. • It has video output options which are on the computer. • An HDMI cable connects the Raspberry Pi to a display device such as TV. • It also has an RCA port for other display options. **3. Graphic Processing Unit (GPU)** • Its primary purpose is to fasten the speed of image calculations. 4. **Memory (RAM)** • It is where real-time information is stored for easy access. • The Raspberry Pi had an initial 256MB RAM size. • The latest model (Raspberry Pi 4) with the maximum capacity is 8GB RAM . **5. Wi fi** • Wi-Fi allows devices such as mobile devices, computers, wearables, and other equipment to interface with the Internet wirelessly using radio waves. • Raspberry Pi supports 2.4/5 GHz dual-band Wi-Fi. **6. Bluetooth** • Bluetooth is a high-speed wireless connectivity technology that is used to transfer data within a small range. • It is used to connect one device with other devices wirelessly. The latest version of Raspberry supports Bluetooth 5 connectivity. **7. Ethernet Port** • Use Ethernet Port to establish an internet connection for the Raspberry Pi. • With the help of Ethernet ports, we can establish a wired internet connection to the Raspberry Pi. • Ue the RJ45 Ethernet Jack for the Ethernet port. 8. **SD Card Slot** • The Raspberry Pi boards contains SD card to act as a hard drive, and it can also contain the operating system necessary for turning the system on. (Write any 4 and explain)	7	7	7
XII	Explain briefly with two applications	4	7	7

XIII	• Connect the sensor's VCC pin to Uno's 5V pin using a red male-to-female jumper cable. • Connect the sensor's ground pin to Uno's ground pin using a black male-to-female jumper cable. • Then, connect the sensor's out pin to Uno's digital pin 2 using a green male-to-female jumper cable.	Example – 3 Explanation – 4	7	7
XIV	• Package in Python is a folder that contains various modules as files. • **Packages** allow for a hierarchical structuring of the module namespace using **dot notation**. • Choose a name for your package and create a directory with that name. • Create an init.py file in the package directory. ... • Create one or more Python modules (i.e., ... • Define the package's API in the init.py file. ... • Write documentation for your package, describing what it does and how to use it. • Choose a name for your package and create a directory with that name. • Create an **init**.py file in the package directory. This file can be empty, but it signals to Python that this directory should be considered a package. • Create one or more Python modules (i.e., .py files) in the package directory, containing the functions and classes you want to include in your package. • Define the package's API in the **init**.py file. This can include importing modules and sub-packages and defining variables and functions that should be exposed to users of the package. • Write documentation for your package, describing what it does and how to use it.	7	7	7