**ASSESSMENT 47**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | 14-07-2020 | **Name:** | Sheela Golasangi |
| **Course:** | Coursera | **USN:** | 4AL16EC068 |
| **Topic:** | Industrial IoT on Google Cloud Platform | **Semester & Section:** | VIII  ‘B’ |
| **Github Repository:** | Sheela-Course |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
| **REPORT**  C:\Users\india\Pictures\Screenshots\Screenshot (1652).png  C:\Users\india\Pictures\Screenshots\Screenshot (1664).png  **Sensors and Devices Introduction**  In this lesson you learn about sensors and devices. Often, sensors and devices are referred to as simply 'devices,' with the presence of sensors implied. The two will be discussed separately in this module, but throughout the rest of the course, 'devices' may refer to both the sensor and device combined.  A sensor is a module that observes changes in its environment and sends information about these changes to a device.  https://d3c33hcgiwev3.cloudfront.net/imageAssetProxy.v1/dtbBP_NXEeilxxL_ZeRz_A_729577cb437741a79f09f5bface6623b_Screen-Shot-2018-11-28-at-1.48.46-PM.png?expiry=1594857600000&hmac=js5-McvBHzf4e_C223wUCW0Td4rmpR0C_8V4p7yjQNI  Devices collect data from sensors and send it to the cloud. Devices can be very small and have very few resources in terms of compute, storage, and so on. They might be able to communicate only through networks that cannot reach a cloud platform directly, such as over Bluetooth Low Energy (BLE). Standard devices are more likely to resemble small computers and may have the ability to store, process, and analyze data before sending it to the cloud.  **Types of Sensors**  There are many sensors available for IoT and a number of ways of categorizing them. The categories discussed below are just a small sample of the ways sensors can be grouped.  Sensors can be divided by their external power requirements:   | **Type** | **Definition** | **Example** | | --- | --- | --- | | **passive** | Does not require external power to operate. They respond to input from their environment. | A temperature sensor that changes resistance in response to temperature changes | | **active** | Requires external power to operate. | A camera |   Type of signal the sensor produces:   | **Type** | **Definition** | **Example** | | --- | --- | --- | | **analog** | Outputs an analog continuous signal | Accelerometers, temperature sensors | | **digital** | The output is converted to discrete values (digital 1s and 0s) before transmitting to a device | Digital pressure sensor, digital temperature sensor |   Type of measuring device:   | **Type** | **Definition** | **Example** | | --- | --- | --- | | **chemical** | Responds to chemical changes in its environment | Gas sensor | | **mechanical** | Responds to physical changes in its environment | Microswitch | | **electrical** | Responds to electrical changes in its environment | Optical sensor |   Choosing sensors for your project requires a clear understanding of what you want to measure and what accuracy is required. Choosing a Sensor When selecting an IoT sensor, there are several things to consider. Typically, the goal for an IoT sensor and device is long life with little human interaction. You expect to place IoT sensors and devices into the desired environment and have them work for an extended period of time. They might be in a remote location or embedded deep within a system, inaccessible to humans. Replacing a sensor and device in this situation may be extremely costly, dangerous, or even impossible; all reasons to carefully consider your sensor and device decisions.  Your decision is based on many factors. As you design your system, you need to carefully consider the importance of each factor and its priority to the overall design.  The following list of considerations can be thought of as a starting point for any IoT sensor discussion. Durability Durability must be considered with regard to the environment of the sensor. You want to make sure your device is as durable as necessary to operate for a reasonable period of time, without incurring unnecessary costs.  For example, a water-resistant temperature sensor may be acceptable for a remote weather station, but it would be completely unsuitable for monitoring water temperature in a pool because it is not waterproof. Accuracy You want to have enough accuracy to correctly monitor an environment, but you don't want to pay for more than you need.  For example, if you are designing a system to regulate the temperature in a remote household storage unit, you are probably willing to accept a sensor that might be accurate with +/- 2 degrees. This accuracy would be completely unacceptable if you were designing a medical device system. A medical device temperature sensor would need to be accurate to +/- 0.2 degrees! Versatility Sensors must be able to operate within reasonable variations of environment. Because most IoT network designs have many sensors, in a variety of environments, it is important to have sensors that can function accurately in all variations of the environment.  For example, if you are building remote weather stations for wilderness areas, you will need to use sensors capable of handling extremes of summer and winter temperatures. It would not be practical to have sensors that only operate accurately at room temperature. Power Consumption Depending upon the situation, your requirements might be for a low-power, or even very low–power device. You will need to decide whether power-saving features (like sleep mode or fast wake up) are necessary.  For example, a sensor or device powered by solar-charged batteries may need to spend a great portion of its life in sleep mode to prolong battery life during low-light times. It may also need fast wake times to accurately capture data. Special Environmental Considerations Sensor choice can even affect the final system design.  For example, when designing a system for monitoring water quality, a sensor that can be placed within the main water supply piping is far more cost-effective and accurate than a sensor that requires diverting water samples. Cost IoT networks usually involve hundreds or even thousands of sensors and devices. Every aspect of sensor design must be scrutinized from a cost perspective. These costs involve more than just the price of the sensor. Consideration must be given to the cost of placement, maintenance, reliability, etc. Devices A "Thing" in the "Internet of Things" is a processing unit that is capable of connecting to the internet and exchanging data with the cloud. Devices are often called "smart devices" or "connected devices." They communicate two types of data: telemetry and state.  https://d3c33hcgiwev3.cloudfront.net/imageAssetProxy.v1/ubBjePPyEeiAgQrXx6bp4g_e2acfde90c42996b6b52e712e4b2aa4f_Screen-Shot-2018-11-29-at-8.20.34-AM.png?expiry=1594857600000&hmac=n8DBSNiRT2TIlOmLScUiQ1mC3qdO6tJT2RbfgNNj9jY ****Types of information**** Each device can provide or consume various types of information. Each form of information might best be handled by a different backend system, and each system should be specialized around the data rate, volume, and preferred API. ****Device metadata**** Metadata contains information about a device. Most metadata rarely, if ever, changes. Examples of metadata fields include:   * Identifier (ID) - An identifier that uniquely identifies a device. * Class or type * Model * Revision * Date manufactured * Hardware serial number  ****Telemetry**** Data collected by the device is called telemetry. This is the eyes-and-ears data that IoT devices provide to applications. Telemetry is read-only data about the environment, usually collected through sensors. ****State information**** https://d3c33hcgiwev3.cloudfront.net/imageAssetProxy.v1/12cMTPPyEeilxxL_ZeRz_A_61a75174e2c8e22ce1aaa88392f4bf3d_Screen-Shot-2018-11-29-at-8.21.26-AM.png?expiry=1594857600000&hmac=bjI2yfIXBkCh7df7w7lJ-u4pKyIz2myPExn35gOlHuU  State information describes the current status of the device, not of the environment. This information can be read/write. It is updated, but usually not frequently.  **Introduction to MQTT and HTTP**  When connecting devices to Google Cloud Platform, you will need to specify which communication protocol your devices will use. The choices are MQTT, HTTP, or both.  https://d3c33hcgiwev3.cloudfront.net/imageAssetProxy.v1/NAsqcu3EEeiAgQrXx6bp4g_cb7543986d240c527eff2a43fc0e2e54_Screen-Shot-2018-11-21-at-11.32.14-AM.png?expiry=1594857600000&hmac=ZsazptYEdAG3AdT_uPy68aBewFn0aMEQzpr-LUMWOjM  In this lesson you are introduced to both protocols, and you compare them. This should help you make an informed decision when designing your network. Introduction to Cloud Pub/Sub Cloud Pub/Sub is an integral Google Cloud IoT. It interacts with Cloud IoT Core, Cloud Functions, Cloud Dataflow.  https://d3c33hcgiwev3.cloudfront.net/imageAssetProxy.v1/N9aROezjEeiaxBKyA9PBAg_f2c94890b7f25dc606ec08b98486b9ea_Screen-Shot-2018-11-20-at-8.37.12-AM.png?expiry=1594857600000&hmac=l6Wg3nOs283fK5H_Pv17Iv-kLhBd_CZ2b9ura91ix3Y  The next node in the course is a Pub/Sub video from the course 'Elastic Cloud Infrastructure: Containers and Services, the instructor is Mylene Biddle, a technical curriculum developer at Google. The video covers the basics of Cloud Pub/Sub, including a detailed flow chart of messages as they pass through Pub/Sub. Introduction to Google Cloud IoT Core In this module we discuss another service in the ingest stage, Cloud IoT Core.  https://d3c33hcgiwev3.cloudfront.net/imageAssetProxy.v1/w8pb9u3BEeilxxL_ZeRz_A_5c8d8e338c3b35b1f085c856bc218f01_Screen-Shot-2018-11-21-at-11.14.55-AM.png?expiry=1594857600000&hmac=lD1T0RAYMLxZ3ETwsCKtaUm9jhcVKuxyluZGFFZHzXk  Cloud IoT Core is a 100% managed service. This means there is no need for you to do autoscaling, setup redundancy, database partitioning, or resource pre-provisioning. You can connect one or millions of devices, and Cloud IoT Core will scale to meet your needs.  https://d3c33hcgiwev3.cloudfront.net/imageAssetProxy.v1/AEsDPPGfEeiTKQ5ajE7PqA_3ac56b31bdce7b5f2b4b7628cb3e1a83_Screen-Shot-2018-11-26-at-9.16.05-AM.png?expiry=1594857600000&hmac=Rq-VGmAcgn_BvksOeDuq3F9yVzVGXCQfGmd2s-Jn2O0  Cloud IoT combines MQTT protocol with the highest level of security (TLS 1.2 with certificates), and it is a single GLOBAL endpoint (mqtt.googleapis.com). When communicating with a device, you don’t need to know the device location, and you don't have to replicate its configuration in each region. Data is automatically published to Cloud Pub/Sub and is accessible globally.  You can update and control devices using the Device Manager.  This course includes an introductory video on Cloud IoT Core. Heather Cross and Jenny Brown introduce Cloud IoT Core and do a quick demonstration.  **Introduction to Google Cloud Storage**  Google Cloud Storage is unified object storage. You can store and/or retrieve data from any where in the world, at any time.  https://d3c33hcgiwev3.cloudfront.net/imageAssetProxy.v1/bQLIke3BEeiTKQ5ajE7PqA_f79b3b9d76490b77f9ba16df0be4f068_Screen-Shot-2018-11-21-at-11.12.32-AM.png?expiry=1594857600000&hmac=zKF5koN2rYFPL7mpBZO5PZiLa-0pnsgsUtMyvjIwXfg  In this lesson you learn about Google's cloud storage options and how to create and access storage buckets.  The lesson begins with a video with Evan Jones and Heather Cross introducing Google Cloud Storage. The video is part of a series of videos on Google's Cloud Platform. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | 14-07-2020 | **Name:** | Sheela Golasangi |
| **Course:** | Sales force | **USN:** | 4AL16EC068 |
| **Topic:** | [Trailhead Playground Management](https://trailhead.salesforce.com/en/content/learn/modules/trailhead_playground_management?trail_id=learn_salesforce_with_trailhead) | **Semester & Section:** | VIII  ‘B’ |
| **Github Repository:** | Sheela-Course |  |  |
| **AFTERNOON SESSION DETAILS** | | | |
| C:\Users\User\Pictures\Screenshots\Screenshot (285).pngC:\Users\User\Pictures\Screenshots\Screenshot (286).png **Create a Trailhead Playground**  **Learning Objectives**  After completing this unit, you’ll be able to:  Create a Trailhead Playground.  Explain the difference between a Trailhead Playground and a Developer Edition org.  **What Is a Trailhead Playground?**  A Trailhead Playground is an org you can use to complete hands on challenges, and try out new features and customizations. Much like a real playground, a Trailhead Playground lets you play around and make customizations without impacting anything else (in this case, your production org).  The only difference is that in a playground, playing means swinging from the monkey bars and riding the merry-go-round. In a Trailhead Playground, it means writing Lightning web components and creating new custom objects. Which, if you ask us, is just as fun?  You can do almost anything to your Trailhead Playground, and it comes with a set of Trailhead-specific data that you can use when completing challenges. Trailhead Playgrounds have some limits, but for the most part they give you the same customization options as a production org. And although you can outgrow a real-life playground, your Trailhead Playground never expires, as long as you keep using it.  **What are the Difference Between a Trailhead Playground and a Developer Edition Org?**  If you’re used to trying out new Salesforce features and playing around in a development environment, you might already have a Developer Edition (DE) org. A DE org is an org that we provide for free to test new features and implementations in Salesforce without affecting a production org.  A Trailhead Playground is like a DE org, but specifically for Trailhead. Trailhead Playgrounds come with Trailhead-specific data, and a pre-installed package that we use to test your hands-on challenges. Trailhead Playgrounds also include tools to make some of the tasks you’ll find yourself completing often easier, such as finding your username and resetting your password, and installing managed packages.  Additionally, My Domain is already active in every Trailhead Playground. My Domain is required to create custom Lightning components and set up single sign-on (SSO) in an org. To learn more about My Domain, check out this help article. To learn how to activate it in your production org, see the User Authentication module.  If you’d rather use an existing DE org, though, we understand. Just choose Log in to a Developer Edition from the dropdown in any hands-on challenge, and enter the credentials for your DE org. Once you’ve linked your DE org to your Trailhead account, you’ll be able to launch it from any hands-on challenge.  **Create Your First Trailhead Playground**  Once you've created a Trailhead account with your Salesforce account or a linked social account, we’ve done all the hard work for you! A Trailhead Playground is created automatically and linked to your Trailhead account.  In every hands-on challenge and project step verification, there’s a dropdown menu. To create a new Trailhead Playground, click the dropdown and select Create a Trailhead Playground. And that’s it! Now you have an org that you can use to complete hands-on challenges and projects, and test new features and code. Note that if you're using Trailhead in a language other than English, your playground still needs to be set to English when you're working on hands-on challenges. Otherwise you may run into issues passing challenges.  Get Your Trailhead Playground Username and Password  Learning Objectives  After completing this unit, you’ll be able to:  Get your Trailhead Playground username and password.  Rename a Trailhead Playground.  Get Your Username and Reset Your Password  Most of the time, you won’t need to know the username and password of your Trailhead Playground. When a Trailhead Playground is linked to your Trailhead account, you can launch it with the click of a button, without logging in to it. You don’t even need to choose a username or fill out any information to create a new Trailhead Playground. You do need your username and password every once in a while, however. For example, if you’re authorizing your org for use with the Salesforce Command-Line Interface (CLI), or signing into it on your phone to see how something looks on mobile.  In most Trailhead Playgrounds, it’s easy to reset your password. First, launch your Trailhead Playground by clicking Launch from any hands-on challenge. If you see a tab in your playground that says Get Your Login Credentials, great! Follow the steps in the Your Playground Has the Playground Starter App section below.  If not, click App Launcher to launch the App Launcher, then click Playground Starter and keep reading. If you don’t see the Playground Starter app, that’s OK—skip to the Your Playground Doesn’t Have the Playground Starter App section. | | | |