**DAILY ASSESSMENT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **21-July-2020** | **Name:** | **Raziya Banu** |
| **Course:** | **Coursera** | **USN:** | **4AL16EC058** |
| **Topic:** | **IOT** | **Semester & Section:** | **8th sem & ‘B’ section** |
| **Github Repository:** |  |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
| **Image of session** |
| **Report –** In my first session today I have studied about - IOT IoT architectures must be capable of scaling connectivity of devices, data ingestion, data processing, and data storage. They must be able to do this quickly while still producing real-time data insights. Sending ever-increasing amounts of data to the cloud slows processor times and requires more bandwidth to transfer and store data.  To mitigate this demand, distributed computing known as fog or edge computing is gaining popularity. The edge refers to the geographic distribution of computing nodes in the network as Internet of Things devices, which are at the "edge" of a network. This in turn increases the demand for devices that are capable of cleaning, processing, and analyzing data locally. The result is that only cleaned metadata is sent to the cloud.  Scaling also means that the ability to easily monitor and maintain thousands of devices must also scale. An asynchronous, scalable communication stack is crucial in bidirectional communication with devices. For example, what happens when thousands of devices must be updated or if they transmit data at the same time? A system that allows for asynchronous communication would be less brittle. A communication protocol that separates sending and receiving, such as MQTT, is a necessity in IoT architecture. At the same time, there are cases when commands sent to a device must be accomplished immediately, resulting in a need for synchronous (or near synchronous) behavior.  Google Cloud IoT platform includes the three stages necessary for an IoT pipeline: data ingestion, data processing, and data analysis. For each of the three stages in the pipeline, several smaller tasks need to be completed. Google Cloud IoT platform has created a fully integrated tool set for each stage.  **Ingesting** includes managing and optimizing IoT device data through secure device connections. Real-time data is collected with sensors. Devices are authorized through Cloud IoT Core. Then the data is uploaded to the cloud through Cloud Pub/Sub.  **Processing** includes cleaning and storing the data with on-demand solutions that scale. You use a Cloud Dataflow pipeline to direct data to Cloud Storage or BigQuery.  **Analysis** includes visualizing and predicting outcomes to generate actionable outcomes. You use BigQuery, Cloud Dataprep, and Cloud Machine Learning Engine to analyze data and gain valuable insights. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **21-July-2020** | **Name:** | **Raziya Banu** |
| **Course:** | **SalesForce** | **USN:** | **4AL16EC058** |
| **Topic:** | **Trailhead** | **Semester & Section:** | **8th sem & ‘B’ section** |
| **AFTERNOON SESSION DETAILS** | | | |
| **Image of session** | | | |
| **Learning Objectives**  After completing this unit, you’ll be able to:   * Describe user engagement as a key part of a user’s experience. * List the key user engagement scenarios. * Explain the Salesforce components that you can use to engage users.   **The Art of User Engagement Scenarios**  User engagement is the process of onboarding, empowering, assisting, and educating users through in-app guidance. You’ve probably encountered in-app guidance in Salesforce without even knowing it. Providing in-app guidance components, such as infobubbles and tooltips, is one way that Salesforce helps users work more effectively. And now, you can bring your unique knowledge of your org and your users to customize those components. But before you get going, you should understand why and when to add in-app guidance.  There are four key scenarios for user engagement. To explain them, we’re going to use an example that takes us outside the office and into an art museum. Picture yourself there. After getting your ticket and checking your backpack, the first things you want to know are what exhibits are on display and where to start. You pick up the gallery guide and map. You discover all the current exhibits, review how the general collection is laid out, and even identify the pieces of art you want to see most. Well, when users are in onboarding mode, they are asking similar questions. User engagement during this phase shows users where to begin and highlights what’s new or changed.  After lunch, you sign up for the audio tour. But as you’re working your way around the museum, you get stuck behind a large group of students on a tour. You can’t quite see the details of the art described in the audio tour, and you want to replay a segment of the tour. Luckily for you, the audio tour equipment has a helpful card attached that tells you how to do just that. The card serves the same purpose as troubleshooting help in a Salesforce app. It’s a good idea to provide help for common tasks and problems so users can get on with their work without too much disruption.  After listening to the audio tour, you want to learn even more about one of the photography exhibits. So you decide to join the next guided tour for the exhibit. The guide is pretty knowledgeable about the artist’s life, her influences, and how her art fits into the larger photography movement. This is the idea behind deeper learning guidance. This is guidance that takes the user on a learning journey to a better understanding of the concepts they need to be more productive and innovative at work. Push Method vs Pull Method Use the push method when users may not notice or seek out help, but would benefit from assistance. Present content to the user even though they don’t seek it out. For example, show a welcome mat upon first login or display a prompt when a user lands on the Account homepage. Prompts, popovers, and walkthroughs are examples of the push method.  Some push components “hide” in plain sight. Empty state messages (also known as zero state or first use states) and field-level help messages are considered push mechanisms because they are presented in the user interface even though the user didn’t specifically seek help.  Use the pull method when the user is motivated to seek help. This is where the Help Menu shines because it’s always sitting, waiting to spring open and help users when they need it. Another common example is the infobubble, which opens a tooltip when you hover over an icon.  **Even More User Engagement Options**  There are some additional features, options, and components that you might not even have considered as user engagement.   * Utility Bar Notes—Give your users quick access to common productivity tools in a fixed footer. Utilities open in docked panels. * Rich Text Component—Add text and simple HTML markup to your Lightning pages. * Guidance for Success on Path—At each step on the path, help users succeed with step-specific guidance, such as tips, links, and information about company policies. * Custom Notification from a Process—Send customized notifications when important events occur. For example, alert an account owner when a new support case is logged. * Einstein Analytics In-Dashboard Videos—Drive adoption and engagement with educational videos. Provide customized instruction that helps users get the most out of each dashboard and its charts.   At this point, you must be pretty excited about the world of user engagement at Salesforce. Next, you learn to quickly and easily set up one of the most powerful tools: in-app guidance prompts. | | | |