***AALIM MUHAMMED SALEGH COLLEGE OF ENGINEERING***

***TITTLE :***

*CHATBOT DEPLOYMENT WITH IBM CLOUD WATSON ASSISTANT*

***TEAM MEMBERS:***

*ABDUL SALAM (au110121104004)*

*ABUBUKAR SIDDIQ (au110121104005)*

*JAGAN (au110121104307)*

*KEVIN HARRIS (au110121104308)*

*SHARUKHAN (au110121104318)*

*SURYA PRAKASH (au110121104320)*

***INNOVATION:***

***1. Setup Your Environment:***

* *IBM Cloud Account: Ensure you have an active IBM Cloud account. If not, sign up for one.*
* *Watson Assistant Service: Create a new Watson Assistant instance in IBM Cloud.*

***2. Initialize Your Workspace:***

* *Within Watson Assistant, create a new workspace. This workspace will contain all the dialogues, entities, and intents for "Eddie."*

***3. Intents & Entities Configuration:***

* *Intents: These represent the user's intentions. Create intents like #ask\_event\_details, #RSVP, etc.*
* *Entities: These are specific details the user might mention. Create entities like @event\_name, @date, etc.*

***4. Dialog Creation:***

* *Use the visual dialog editor in Watson Assistant to start constructing the conversation flow.*
* *For each intent, create a new dialog node. Within each node, define the bot's response and any necessary conditional logic based on recognized entities or context.*

***5. Advanced Configuration with Slots and Context:***

* *Slots: If you need to collect multiple pieces of information in a structured manner (like RSVP details), use slots in your dialog nodes.*
* *Context Variables: Store and manipulate information during a conversation using context variables. For example, if a user mentions an event name, store it in a context variable for reference in later parts of the conversation.*

***6. Webhook Integration (if necessary):***

* *If "Eddie" needs to fetch real-time data (like available seats) or integrate with other systems (like an event management database), set up webhooks.*
* *This involves creating an external API (potentially using something like IBM Cloud Functions) that the Watson Assistant can call.*

***7. Testing the Chatbot:***

* *Use the "Try it" feature in Watson Assistant to test Eddie's interactions.*
* *Address any misunderstandings and refine the dialog to improve the user experience.*

***8. Integration with Channels:***

* *Decide where Eddie will be accessible from (e.g., a website, Slack, Facebook Messenger).*
* *Use Watson Assistant's built-in integrations or the provided API to connect Eddie to the desired channels.*

***9. User Interface (UI) Implementation:***

* *If integrating Eddie onto a website, design the chat interface. While Watson provides default UI components, you may want to customize this to fit your event theme or branding.*

***10. Deployment and Scaling:***

* *Once everything is set and tested, deploy Eddie to the chosen platform.*
* *Ensure backend scalability, especially if expecting high user volume during events.*

***11. Feedback Loop and Iteration:***

* *After deploying Eddie, gather feedback from actual users.*
* *Use this feedback to make enhancements, fix issues, and improve the overall user experience.*

***12. Documentation and Training:***

* *If event organizers or other stakeholders need to interact with Eddie or update event details, provide them with documentation or training sessions.*

***13. Ongoing Maintenance:***

* *Technology and user expectations change over time. Ensure Eddie is regularly updated based on the latest tech trends, user feedback, and any changes in event management processes.*