

Day	Task	AI Tool Used	Output
1	<b>Architecture &amp; Setup</b> - Create Firebase project, set up local environment. Initialize Firebase functions and install dependencies (in the terminal) such as <code>npm</code> <code>axios</code> <code>dotenv</code> <code>firebase-admin</code> <code>firebase-functions</code> <code>node-fetch</code> and <code>openai</code>	Firebase WSL Node.js	Deployed Firebase Functions backend  CLI environment ready for firebase JavaScript function
2	<b>Poe AI Setup</b> – Create a chatbot backed by OpenAI in poe.com and utilize its embedded API JavaScript code. Ensure Poe AI is trained (Knowledge Base) with various media such as video, articles and websites regarding Oral Health. Ensure Poe is prompted specifically for Oral Health information.	Poe AI API	A dependable, intelligent AI backend to quickly respond to user queries.
	<b>Meta Developer Setup</b> - Create WhatsApp Business App, get phone number ID, generate access token.	Meta Developer Console Firebase Secret Manager	Configured WhatsApp with API credentials stored as Firebase Secrets
3	<b>Webhook Integration</b> - Code the Meta verification handshake and structure to receive incoming messages.	Meta Developer Tools	Successfully verified webhook URL with Meta. Proof of message reception in logs.
4	<b>Backend AI Integration</b> - Code the function to send user messages to Poe AI and return the response to user by integrating Firebase functions with the WhatsApp webhook function.	Firebase Poe AI API WhatsApp Business API	Functional connection between a user's message and the AI backend. Concept complete.
5	<b>Deployment and Debugging</b> – Final testing, solve deployment container errors and WhatsApp API authentication errors	Firebase Debug Logs WhatsApp API	Working prototype