**JOB RECRUITMENT & HIRING MANAGEMENT SYSTEM**

**NAME: KOLLI JAGAN MOHAN RAO**

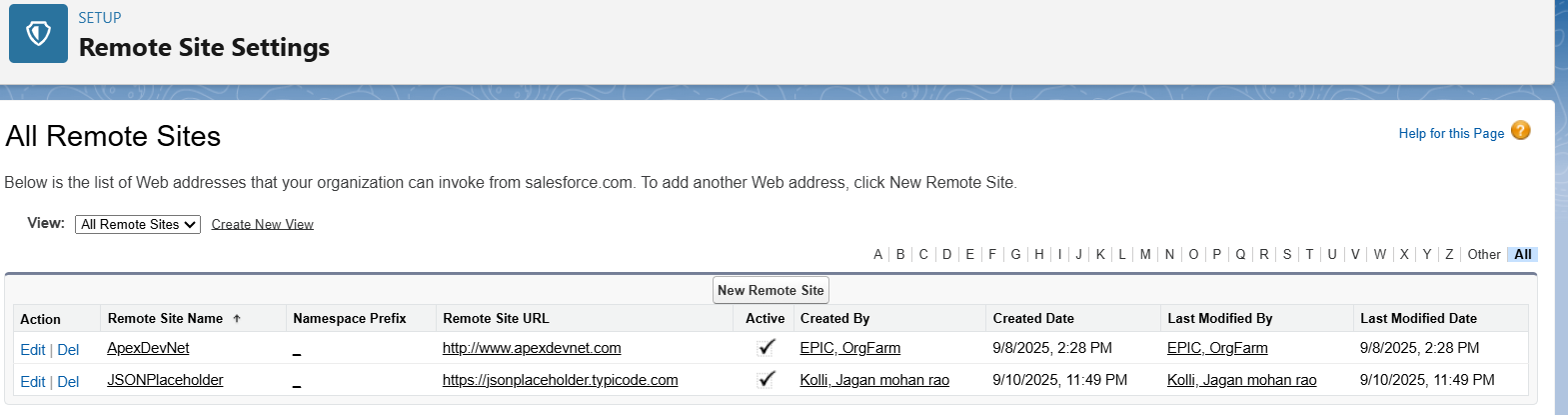
**Phase 7: Integration & External Access**

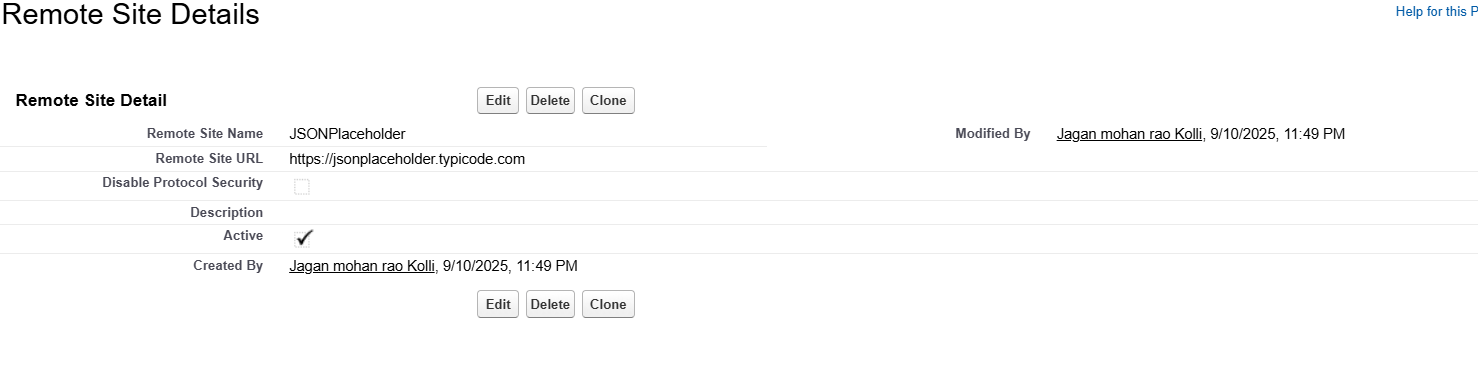
**Objective**: To connect the Salesforce application with an external web service to fetch data in real-time, demonstrating the platform's ability to integrate with other systems.

**Integration Strategy**: An **outbound REST API callout** was implemented. This involved an LWC on the front end initiating an action that called an Apex class on the backend. The Apex class then made an HTTP request to an external endpoint and returned the data to the LWC for display.

**Endpoint Configuration**:

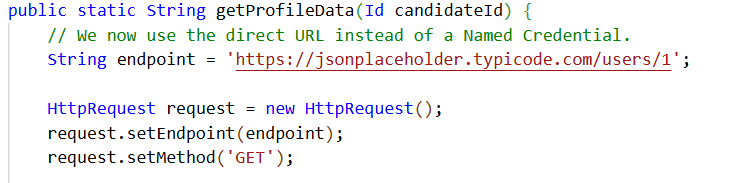
* **Initial Approach**: The initial plan was to use a Named Credential, which is the modern, secure method for storing endpoints and authentication details. We configured both an **External Credential** (Public API No Auth) with a No Authentication protocol and a **Named Credential** (JSONPlaceholder).
* **Troubleshooting & Pivot**: Due to persistent and unusual permission errors specific to the developer org that blocked the callout, a strategic pivot was made.
* **Final Approach**: A **Remote Site Setting** was configured instead. This is a standard but older method for whitelisting external URLs for callouts. The URL https://jsonplaceholder.typicode.com was added, explicitly authorizing Apex to make requests to this domain.





**Backend Development (Apex Callout)**:

* An Apex class, **CandidateProfileService**, was created to handle the callout logic.
* An @AuraEnabled(cacheable=true) static method, **getProfileData**, was written to be accessible from the LWC.



**Frontend Development (LWC Integration)**:

* A custom Lightning Web Component, **candidateApiProfile**, was built to provide a user interface for the integration.

