|  |  |
| --- | --- |
| Resume Parser Project: Meeting minutes | |
| PROJECT: | Building a Resume Parser in Python using NLP and deploying the model on a cloud based platform using streamlit application. |
| DATE: | February 27 – March 31, 2023 |
| MEMBERS: | SIDDHARTHA PATRA (0787887) |
|  | USAMA NORAT (0792225) |
|  | AMIR DAHYA (07912521) |

# Specific Activities from the prior 4 weeks

The following work activities that were completed:

1. **Streamlit App:**
   * Defined a function called "pred" that takes the text of a resume as input and uses Spacy's pre-trained model to extract entities from the text.
   * Developed a user interface using Streamlit, where the user can upload a resume in PDF or DOCX format, and the system processes the file and extracts important information from it.
   * If the uploaded file is a PDF, our code uses ***pdfplumber*** library to extract text from each page of the PDF and passes it to the "pred" function.
   * If the uploaded file is a DOCX, our code uses ***docx2txt*** library to extract text from the file and passes it to the "pred" function.
   * The extracted information is displayed in the user interface using Streamlit's "***st.write***" function.
2. **Cloud Source Repository:**
   * A new repository was created or connected to an external repository.
   * The repository was cloned.
3. **Git Commands:**
   * The necessary files were uploaded to the repository in zip format.
   * The folder was unzipped and moved to the git folder.
   * Steps (git commands) were followed in order to add the files to the cloud source repository.
4. **Cloud Build Trigger:**
   * A new cloud build trigger was created in the GCP console.
   * The trigger was run.
5. **Docker File Creation:**

* A Docker file was created.
* Google Container Repository (GCR) was used to store the deployed images.

1. **Virtual Machine Creation:**
   * A VM instance was created.
2. **Server Deployment 1 (Server-Based Deployment):**
   * The team connected to the VM instance and set up the VM instance using the given code lines provided in ‘setup-new-vm.sh’ file.
   * Once the above steps were successfully completed, the team went to the particular URL (copying the external IP code and port) to access the resume parse page and extract the details.
3. **Server Deployment 2 (Docker Deployment):**
   * The team connected to the VM instance and installed the Docker file using the given code lines provided in ‘install-docker.sh’ file.
   * The Docker image was pulled from the container and run on the VM.
   * Once the above steps were successfully completed, the team went to the particular URL (copying the external IP code and port) to access the resume parse page and extract the details.

# Specific Outcome from prior 4 weeks

1. We developed a code is for a Streamlit app that allows users to upload their resume (in either PDF or DOCX format) and process it using Spacy model to extract key information (or entities) from it. The extracted information includes entities such as Name, Location, Companies worked at, Designation, College & Degree, which are displayed on the Streamlit app interface. The app could be useful in a capstone project where the team needs to process quickly and easily resumes of potential job candidates.
2. Two server-based deployments were carried out:

* **Server deployment 1** involved connecting to the VM instance, updating and upgrading the system, cloning the repository, installing required dependencies, creating and activating a virtual environment, installing necessary Python libraries, and running the Streamlit application. The external IP code and port were then copied and used to access the resume parser page.
* **Server deployment 2** involved connecting to the VM instance, installing Docker, pulling the docker image from the container, and running it on the VM. The external IP code and port were then copied and used to access the resume parser page.

# On Target

The project is on target and the status is green, as we have successfully developed an web based application on Streamlit and deployed it on the Google Cloud using two different methods i.e., a server-based deployment and a docker based deployment.

# Challenges/Disagreements

We mostly faced technical difficulties such as:

* Setting up and connecting to the cloud source repository
* Uploading and adding files to the repository using git commands
* Configuring and running the cloud build trigger
* Creating and configuring the docker file
* Choosing a virtual machine provider and configuring the instance
* Installing and running the necessary packages and dependencies for server-based deployment
* Configuring and integrating the docker file with docker-based deployment
* Addressing technical difficulties and ensuring technical expertise, effective communication, and collaboration among team members.

# Planned Activities for coming weeks

The team has planned the following activities for the coming week:

* + - Refining the model to improve its accuracy.
    - Testing the app on more unseen dataset & resumes.
    - Working on the Project Report and the Project Presentation.