**RESUME BUILDER**

The Project Code is for a Resume Builder UI and API which allows users to easily create Resumes.

**Tech Stack :-**

* Resume Builder UI :- HTML, CSS, JavaScript, Bootstrap, JQuery
* Resume Builder API :- Node JS, Express JS, npm packages

**BASIC WORKFLOW:-**

* The **app.js** is the main file from where execution begins. It creates an **express application** and listens on port 8080. Also, the required setup for the Document Generation SDK is done at the same time.
* If the application receives a **GET Request at ‘/’ endpoint**, it sends the **Index.html** file to load the **RESUME BUILDER UI.**

The UI follows Client-Side Rendering to Provide Smooth User Experience.

* On receiving **POST Request at ‘/resume’ endpoint**, the application calls the

**createResume** Middleware Function.

* This function first **verifies the Request Object** if it follows the expected format and all fields and subfields are present, their type is correct, they are non-empty and if there are no extra fields present.
* It then extracts all the fields data and **creates a valid JSON Object** to send to the Document Generation API.

After successful verification and extraction, the Document Merge Operation is instantized and executed. On receiving a success Response, the PDF file is sent in the response.

* Throughout the process, If the Application encounters any error or finds an invalid request object, it sends an error message in response.

**RESUME BUILDER API CODE DESCRIPTION:-**

* **app.js :-** This is the main file from where execution starts. It creates an express application, listen and serves the requests at PORT 8080 and imports all the necessary modules which sets up the SDK.
* **Controller :-** This Folder contains all the JS Modules.
  + **verifyRequest.js :-** This module takes as input a Request Object and then verifies it. It verifies the request headers, Presence of all the required fields and subfields and no extra fields. It also checks the type of each field and makes sure no field/sub field is empty.
  + **createJSON.js :-** This Module takes as input a Request Data Object and returns a JSON Object in the format to be sent to the Document Generation API.

It also trims the strings to remove unnecessary spaces at the beginning/end, converts the linkedin URL to HyperLink. For rest of the HyperLinks, they must be passed in the request Data directly.

* + **createResume.js :-** This Module first sets up the Document Generation API. It has a Middleware function that takes as input a request and response object, uses above Modules to verify the same and extract JSON Object.

It then sends a request to the Document Generation API and sends the PDF/Error response.

It uses a request count variable to name output PDFs of every request uniquely. As soon as the response is sent, the output PDF is removed to save space and the count variable is reset when no active requests are present.

* **Credentials Folder** contains the SDK credentials files(credentials.json file and secret.js file which contains client ID and Secret Key) .
* **ResumeTemplates** Folder stores the Templates in docx format and **TemporaryResumes** Folder stores the Document Generation API output pdf temporarily.
* **config folder** is for managing the default SDK logging.

**RESUME BUILDER UI CODE DESCRIPTION:-**

* **Views :-** This folder contains the index.html file that is the Homepage for the Resume Builder UI.

**MUST READ KEY POINTS :-**

* The API can Handle Multiple Requests Independently at the Same Time. The Rate Limit and Response Time is primarily dependent on that of Document Generation API.
* The PDF files generated by the SDK are deleted just after response is sent and so, no Information is cached between requests as well as Space is not wasted.
* **Error Handling :-** If Some PDF is not able to be extracted by the API due to any server side issue, time out, etc. The request is sent again in order to not miss any PDF in our final CSV File.

So, The Code gives an **accurate result** and has been verified several times.