OCR and Text Extraction API Using FastAPI

# **Introduction**

This report provides a comprehensive overview of the OCR (Optical Character Recognition) and text extraction API developed using FastAPI. The API is designed to accept images, extract text from these images using either the Veryfi or Google Vision OCR services, and then convert the extracted raw text into a hierarchical JSON format. Additionally, it populates a predefined template with precise data extracted from the hierarchical JSON.

# **Overview**

**Project Objectives**

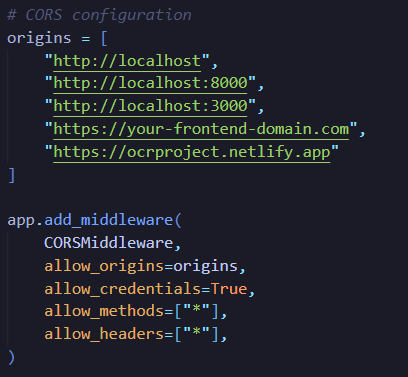
1. **Text Extraction**: Provide an API that supports text extraction from images using leading OCR services.
2. **Flexible Service Selection**: Allow users to choose between Veryfi and Google Vision for OCR, catering to different use cases and preferences.
3. **Data Structuring**: Convert the extracted text into a hierarchical JSON format for better data organization.
4. **Template Population**: Populate a user-defined template with accurately extracted data, streamlining data processing and integration into other systems.

**Libraries and Dependencies**

1. **FastAPI and Related Libraries**
   * **FastAPI**: A modern, high-performance web framework for building APIs with Python. It is built on standard Python type hints, ensuring type safety and ease of development.
   * **HTTPException**: A FastAPI utility used to handle HTTP errors and return appropriate HTTP status codes and error messages.
   * **File** and **UploadFile**: FastAPI utilities for handling file uploads via HTTP POST requests, allowing users to upload images for OCR processing.
   * **Query** and **Depends**: Used for extracting and validating query parameters and for dependency injection, which simplifies code and enhances reusability.
   * **FileResponse**: Provides a convenient way to return files as HTTP responses.
   * **CORS Middleware**: Implements Cross-Origin Resource Sharing (CORS) to control resource access across different domains, critical for security and functionality in web applications.
2. **Additional Libraries**
   * **Pydantic**: A data validation and settings management library using Python type annotations. It is heavily integrated into FastAPI, ensuring that the API's input data is validated before processing.
   * **dotenv**: A library for loading environment variables from a .env file, which is particularly useful for managing sensitive data like API keys in development and production environments.
   * **OpenAI API**: The OpenAI library is used to interact with OpenAI's GPT models, which are critical for converting raw text into hierarchical JSON and populating templates with extracted data.
   * **Veryfi API**: Provides the necessary tools to interact with Veryfi's OCR API, enabling the extraction of structured data from receipts, invoices, and other documents.
   * **Google Cloud Vision API**: Google’s OCR service for analyzing images, detecting text, and returning structured results.
   * **hashlib**: A Python standard library module used for generating secure hash values, specifically SHA-256, which is used in this project to create API keys.
   * **datetime**: Python's standard library for working with dates and times, crucial for generating and validating time-based API keys.
   * **io** and **tempfile**: Used for file handling operations, including temporary storage of uploaded files during processing.
   * **typing**: Python’s standard library module for type hinting, ensuring clarity in function signatures and aiding in code maintenance.

**CORS Configuration**

Cross-Origin Resource Sharing (CORS) is essential in web applications where resources need to be shared between different domains. This API uses CORS middleware to specify which domains are allowed to access the API, preventing unauthorized access and ensuring secure data exchange.



* **allow\_origins**: Specifies the list of origins (domains) that are allowed to access the API. In this case, both local development environments and production domains are permitted.
* **allow\_credentials**: Allows cookies and authentication headers to be included in the requests, enabling stateful sessions and secure API access.
* **allow\_methods**: Grants permission for all HTTP methods (GET, POST, etc.) to be used in requests to the API.
* **allow\_headers**: Ensures that all headers, including custom ones, are accepted by the API, facilitating the transfer of metadata like API keys and content types.

**API Endpoints**

The API is structured around several key endpoints, each with specific functionality related to OCR processing, API key generation, and data conversion.

**1. Root Endpoint**

The root endpoint is a simple health check to ensure the FastAPI service is running correctly. It returns a basic JSON message confirming the API's status.

A black background with white text

Description automatically generated

**2. Generate API Key**

This endpoint is responsible for generating a time-based API key. The key is created using the SHA-256 hash of the current date and time, ensuring uniqueness and security. The API key can be validated for up to 30 days, after which it becomes invalid.

A screen shot of a computer code

Description automatically generated

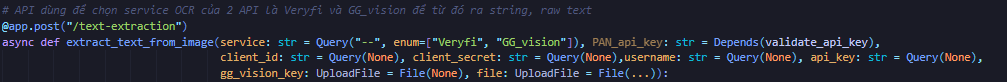
**A computer screen with colorful text

Description automatically generatedImplementation Details:**

* **SHA-256 Hash Generation**: The current date and time are converted to a string and then hashed using the SHA-256 algorithm. This hash serves as the API key.
* **Time-based Validation**: API keys are validated by comparing them with the potential keys generated within the last 30 days, allowing for a balance between security and usability.

**3. Text Extraction Endpoint**

This is the core endpoint for text extraction. Users can choose between the Veryfi and Google Vision services. The endpoint accepts image files, processes them using the selected OCR service, and returns the extracted text.

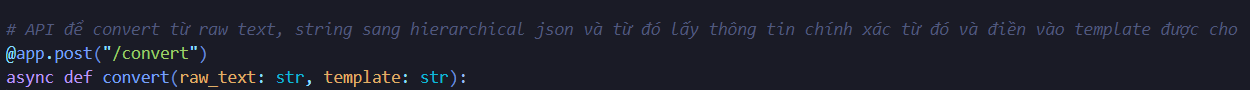


**Processing Logic:**

1. **Veryfi OCR**:
   * **Credentials Check**: Ensures that all required Veryfi credentials (client\_id, client\_secret, username, api\_key) are provided.
   * **Temporary File Storage**: The uploaded image is stored temporarily to facilitate processing.
   * **OCR Processing**: The Veryfi client processes the document, extracting text and returning it in a structured format.
2. **Google Vision OCR**:
   * **Service Account Key Handling**: The Google Vision service requires a service account JSON key, which is uploaded and stored temporarily.
   * **OCR Processing**: The image is analyzed by Google Vision, and the extracted text is returned.
3. **Error Handling**: The API includes robust error handling to manage missing credentials, service selection errors, and any processing failures, returning appropriate HTTP status codes and error messages.

**4. Convert Text Endpoint**

This endpoint is designed to transform the raw text extracted from the image into a hierarchical JSON structure and then use this JSON to fill a predefined template. It leverages OpenAI's GPT-3.5-turbo model to ensure the accuracy and relevance of the data.



**Conversion Process:**

1. **Raw Text to Hierarchical JSON**:
   * **Initial Processing**: The raw text is first processed into a hierarchical JSON format. This structuring is crucial for organizing the data in a way that is easy to manipulate and understand.
   * **OpenAI GPT-3.5-turbo**: The API sends the raw text to OpenAI's model, which processes it into a structured JSON format. This step is critical for tasks that require understanding the context and extracting meaningful data from unstructured text.
2. **Template Population**:
   * **Template Parsing**: The provided template is parsed, and placeholders within the template are identified.
   * **Data Insertion**: The structured data from the hierarchical JSON is inserted into the corresponding placeholders in the template, ensuring that the final output is correctly formatted and contains accurate information.
3. **Error Handling**:
   * **Exception Management**: The API captures any errors during the conversion process, including issues with template formatting or data extraction, and returns a detailed error message.

**API Key Generation and Validation**

The security of the API is paramount, and the generation and validation of API keys are central to this. The keys are generated based on the current date and time, ensuring that each key is unique and time-bound.

A computer screen shot of a code

Description automatically generated

A computer screen shot of code

Description automatically generated

**Key Features:**

* **Unique API Keys**: Each API key is unique, based on the exact date and time it was generated.
* **30-Day Validation Window**: API keys are valid for 30 days, after which they are automatically invalidated, balancing security with user convenience.

**Text Extraction using Veryfi and Google Vision**

The API supports two major OCR services: Veryfi and Google Vision. Users can select the service that best fits their needs, providing flexibility in text extraction.

A screen shot of a computer program

Description automatically generated**Service-Specific Details:**

1. **Veryfi**:
   * **Use Case**: Best suited for processing receipts, invoices, and other documents requiring categorization.
   * **OCR Process**: After validating credentials, the image is processed, and the OCR result is returned as structured text.
2. **Google Vision**:
   * **Use Case**: Suitable for general-purpose text extraction from a variety of image types.
   * **OCR Process**: Google Vision processes the image and returns the full text annotation, which includes the extracted text.
3. **Error Handling**:
   * **Credential Validation**: Ensures that all necessary credentials are provided before processing.
   * **File Handling**: Temporary files are used for processing, and all file operations are managed securely.

**Conversion to Hierarchical JSON and Template Filling**

One of the API's advanced features is converting extracted text into a structured JSON format and using that to populate a user-defined template.

A screenshot of a computer program

Description automatically generated

**A computer screen shot of a program

Description automatically generatedProcess Breakdown:**

1. **Raw Text Conversion**:
   * **Initial Processing**: The API takes the raw text and appends it to a predefined prompt, sending it to OpenAI's model.
   * **Hierarchical JSON**: The GPT model returns a structured JSON representation of the text, organizing the data hierarchically.
2. **Template Filling**:
   * **Template Parsing**: The provided template is analyzed, and the JSON data is used to fill in the placeholders.
   * **Final Output**: The filled template is returned as a response, ready for further use.
3. **OpenAI Model Usage**:
   * **Model Selection**: Different fine-tuned models are used for JSON conversion and template filling, ensuring that each task is handled by a model best suited to it.
   * **Temperature and Token Settings**: These settings are adjusted to balance the creativity and accuracy of the model's responses.
4. **Error Handling**:
   * **Catch-all Exception Handling**: The API captures any errors that occur during processing and returns a detailed error message, making debugging easier.

# **Conclusion**

The OCR and text extraction API developed using FastAPI is a powerful tool for extracting and processing text from images. By supporting both Veryfi and Google Vision OCR services, it provides flexibility for different use cases. The conversion of raw text into hierarchical JSON and subsequent template filling streamlines data processing, making this API an invaluable resource for applications that require accurate and structured text extraction.

**Key Features Recap:**

* **Flexible OCR Service Selection**: Users can choose between Veryfi and Google Vision, depending on their needs.
* **Secure API Key Generation**: Time-based API keys provide robust security while remaining user-friendly.
* **Advanced Text Processing**: Integration with OpenAI's models ensures that the text is not only extracted but also meaningfully structured and formatted.
* **CORS Protection**: Ensures that only authorized domains can interact with the API, preventing unauthorized access.
* **Error Handling**: Comprehensive error management ensures that users receive clear and actionable feedback in case of issues.

This API can be integrated into a variety of applications, including document processing systems, automated data entry platforms, and any other scenario where text extraction from images is required. Its combination of flexibility, security, and advanced processing capabilities makes it a standout solution in the field of OCR and text extraction.