ENZIGMA- Assignment for AI Intern Position

ASSIGNMENT 01

Name- Akshay Bhosale

PRN- 202101040158

MIT Academy of engineering, Pune

Project Documentation: Automated Onboarding System with OCR and Database Integration

GitHub Link:- https://github.com/letsdoitbycode/OCR\_detection

Demo drive links:- https://drive.google.com/drive/folders/1JQt4TOC1ZH\_4Xh9YR4lZYCMIM6eo1ORG?usp=sharing

Overview of the Solution

This project will develop an onboarding automation process, during which users can upload a scanned form in the format of an image. Solution: The textual information is captured from these forms, further consolidated into structured data, placed in a MySQL database, and provided with an easy-to-navigate user interface for uploading files as well as managing records.

Key Features:

1. File Upload: Upload scanned images in .jpg, .jpeg, or .png formats.
2. OCR Processing: Extract both printed and handwritten text from images.
3. Data Consolidation: Process and clean extracted text, converting it into a structured tabular format.
4. Database Storage: Save structured data into a MySQL database for easy management and retrieval.
5. Record Management:
   * View all records in a tabular format.
   * Search records by name or email.

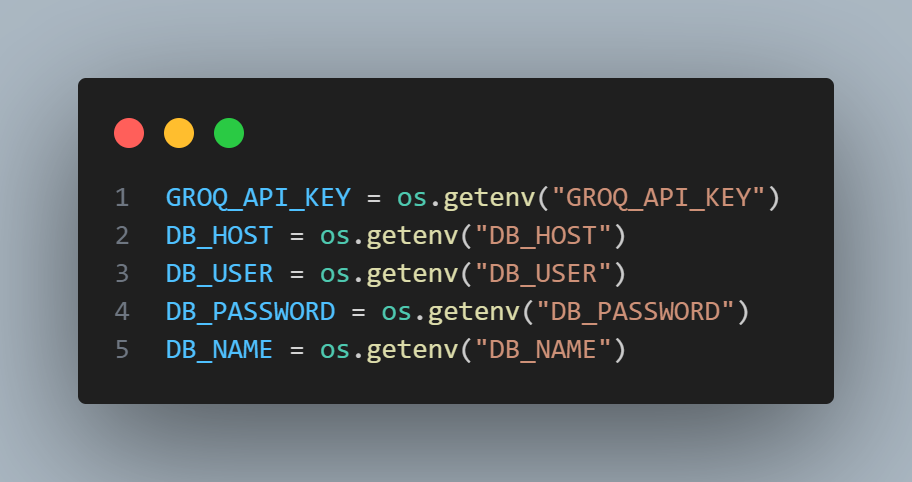
Steps to Set Up and Run the System

Prerequisites:

1. Python 3.x installed on your machine.
2. MySQL server running locally or on a remote server.

Environment Variables:

Create a .env file in the project root directory with the following variables:



Running the Application:

1. Start the MySQL server and ensure the database is accessible.
2. Run the Streamlit application using the command:

* streamlit run app.py

Database Schema

Database Table: Record

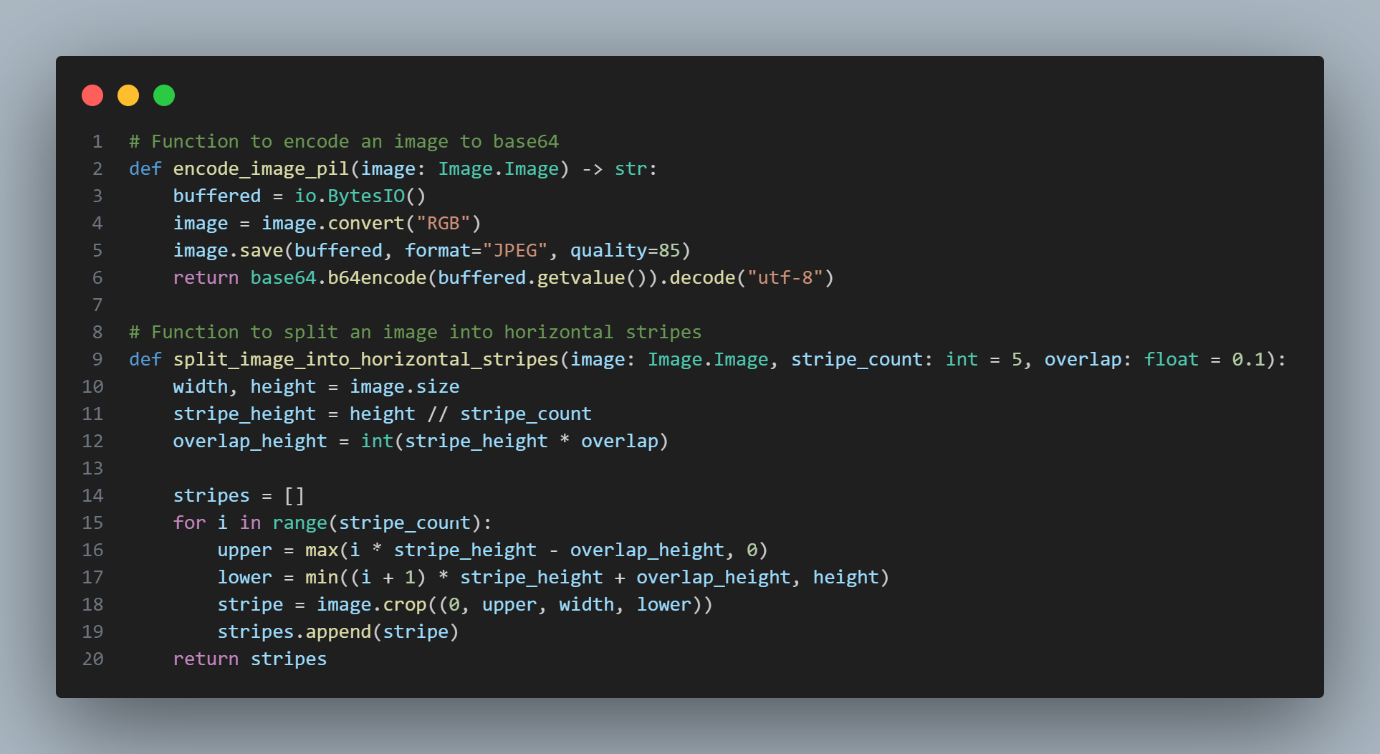
The Record table stores extracted details in a structured format. Below is the schema:

| Field | Type | Description |
| --- | --- | --- |
|  |  |  |
| name | VARCHAR(255) | Full name of the candidate. |
| email | VARCHAR(255) | Email address of the candidate. |
| address | TEXT | Residential address. |
| dob | DATE | Date of birth. |
| age | INT | Age of the candidate. |
| gender | VARCHAR(50) | Gender of the candidate. |
| mobile | VARCHAR(15) | Contact number. |
| education | TEXT | Educational qualifications. |
| profile | TEXT | Professional profile or job title. |

Code and Functionalities

1. Image Upload and Preprocessing

The system allows users to upload scanned forms and preprocesses them for OCR. Preprocessing includes resizing the image and splitting it into horizontal stripes for better accuracy during text extraction.



2. OCR Functionality

Extracts textual content from uploaded images. Both printed and handwritten elements are captured and processed.



3. Data Consolidation

The extracted text from image sections is combined into a structured format, eliminating duplicates and resolving conflicts between overlapping sections.



4. Data Parsing

Converts the consolidated tabular data into a Python dictionary format for database insertion.



5. Database Operations

* Insert Records: Saves extracted details into the MySQL database.
* Fetch All Records: Retrieves all stored records.
* Search Records: Searches for records by name or email.



6. Streamlit Interface

The user interface built with Streamlit includes:

1. Sidebar for Uploading Files: Allows users to upload images and displays a preview.
2. Main Section: Displays OCR results and buttons to save data to the database.
3. Record Management: Provides options to view all records or search specific entries.

Instructions for Using the User Interface

1. Uploading an Image:
   * Use the Upload Image section in the sidebar.
   * Preview the uploaded image to confirm its accuracy.
2. Processing and Saving Data:
   * After uploading, the image is processed to extract text.
   * Review the consolidated tabular data in the OCR and Results section.
   * Click Insert Records into Database to save the data.
3. Managing Records:
   * Use the View All Records button to see all stored entries.
   * Search for specific records using the Search Records input field.