AI-Powered Invoice Extraction: Streamlining Data Extraction with Gemini API

PROJECT REPORT

Submitted by

Vignesh G	**CB.EN.U4CSE21466**
Kannan Ganesh	**CB.EN.U4CSE21426**
Viswaa Ramasubramanian	**CB.EN.U4CSE21468**
Harecharan S	**CB.EN.U4CSE21450**
Sanjay Balamurugan	**CB.EN.U4CSE21453**
Sanyam Bharadwaj	**CB.EN.U4CSE22044**

Department Computer Science and Engineering

Amrita Vishwa Vidyapeetham
Ettimadai - 641112

in collaboration with

SAP Labs, Bengaluru

April 2024

BONAFIDE CERTIFICATE

Certified that the mini-project titled **"AI-Powered Invoice Extraction"** is the Bonafede work of **Vignesh G (CB.EN.U4CSE21466), Kannan Ganesh (CB.EN.U4CSE21426), Viswaa Ramasubramanian (CB.EN.U4CSE21468), Harecharan S (CB.EN.U4CSE21450) , Sanjay Balamurugan (CB.EN.U4CSE21453) , Sanyam Bharadwaj (CB.EN.U4CSE22044)** which is carried out under our mentoring.

SIGNATURE

INDUSTRY MENTOR
Faiz Siddiqui
Spend Engineering,
SAP Labs
Bengaluru,
Karnataka, India

SIGNATURE

ACADAMIC MENTOR
Dr. Guruprakash J
Assistant Professor (SG)
Computer Science and Engineering
Amrita Vishwa Vidyapeetham
Ettimadai - 641112

SIGNATURE
Srivatsan Santhanam
VICE PRESIDENT
Spend Engineering,
Head of Concur R&D,
SAP Labs
Bengaluru,
Karnataka, India

SIGNATURE
Vidhya Balasubramanian
HEAD OF THE DEPARTMENT
Professor
Computer Science and Engineering
Amrita Vishwa Vidyapeetham
Ettimadai - 641112

Submitted for evaluation on 5 April 2024

ACKNOWLEDGEMENT

We would like to express our deepest gratitude to our beloved **Principal, Dr. Vidya Balasubramanian, Ph.D. in Computer Science, M.S. in Computer and Information Science, B.E. in Computer Science Engineering**, for her overwhelming support and encouragement on this project.

We extent our indebted to **Mr. SRIVATSAN SANTHANAM,** Vice President, Spend Engineering, Head of Concur R&D, SAP Labs India for the opportunity and support which was instrumental in the completion of this project.

We are greatly indebted to **Dr. Vidya Balasubramanian, Ph.D. in Computer Science, M.S. in Computer and Information Science, B.E. in Computer Science Engineering,** Head of the Department, Computer Science and Engineering for her guidance and continuous support which was instrumental in the completion of this project.

We extend our thanks to our industry mentor, **Faiz Siddiqui**, SAP Labs, Bengaluru, India and academic mentor **Dr. Guruprakash J**, **Ph.D. in Computer Science Assistant Professor (SC),** Computer Science and Engineering for his technical support and constant supervision without which we could not have completed this project study.

Students name
Vignesh G
Viswaa Ramasubramanian
Kannan Ganesh
Harecharan S
Sanjay Balamurugan
Sanyam Bharadwaj

TABLE OF CONTENTS

ABSTRACT

This project presents a Streamlit application that leverages the advanced capabilities of the Gemini API, a cutting-edge generative AI model developed by Google, to streamline the extraction of critical details from invoices. Users can upload image or PDF files containing invoices, and the application seamlessly processes the content using specialized functions tailored to handle different input formats. The Gemini model, trained extensively on vast datasets of text and image data, excels in analyzing invoice content to extract essential information such as invoice numbers, dates, total amounts, currency types, item quantities, and customer names. Once processed, the extracted details are presented within the Streamlit interface, offering users a convenient means to review and analyze the invoice information.

In addition to invoice extraction, the application includes an analytics component that provides insights into month-wise and year-wise expenditure. The backend of the application utilizes the Super Bass Acid Database, integrated with PostgreSQL, to manage and store invoice data securely. This choice is preferred for its UI management capabilities and row-level security features, ensuring data integrity. The database stores user authentication details and all uploaded invoices, allowing users to filter and view their invoice history conveniently. Users also have the option to edit extracted invoice details before final submission, enhancing accuracy and control over stored information.

Overall, the integration of the Gemini API into the Streamlit application offers a powerful solution for automating invoice extraction and analysis. This not only enhances efficiency but also improves accuracy and productivity across various domains, including finance, accounting, and business administration.

Keywords: Streamlit application, Gemini API, Invoice extraction, AI model, analytics, database management.

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
API	Application Programming Interface
AWS	Amazon Web Services
DB	Data Base
OCR	Optical Character Recognition
PDF	Portable Document Format
PyPDF	Python PDF Library
S3	Amazon Simple Storage Service
UI	User Interface

CHAPTER 1

INTRODUCTION

1.1 PROJECT MOTIVATION

The inefficiencies and challenges of manual invoice processing drove the development of this project, aiming to optimize operations and enhance accuracy through advanced technologies like the Gemini API.

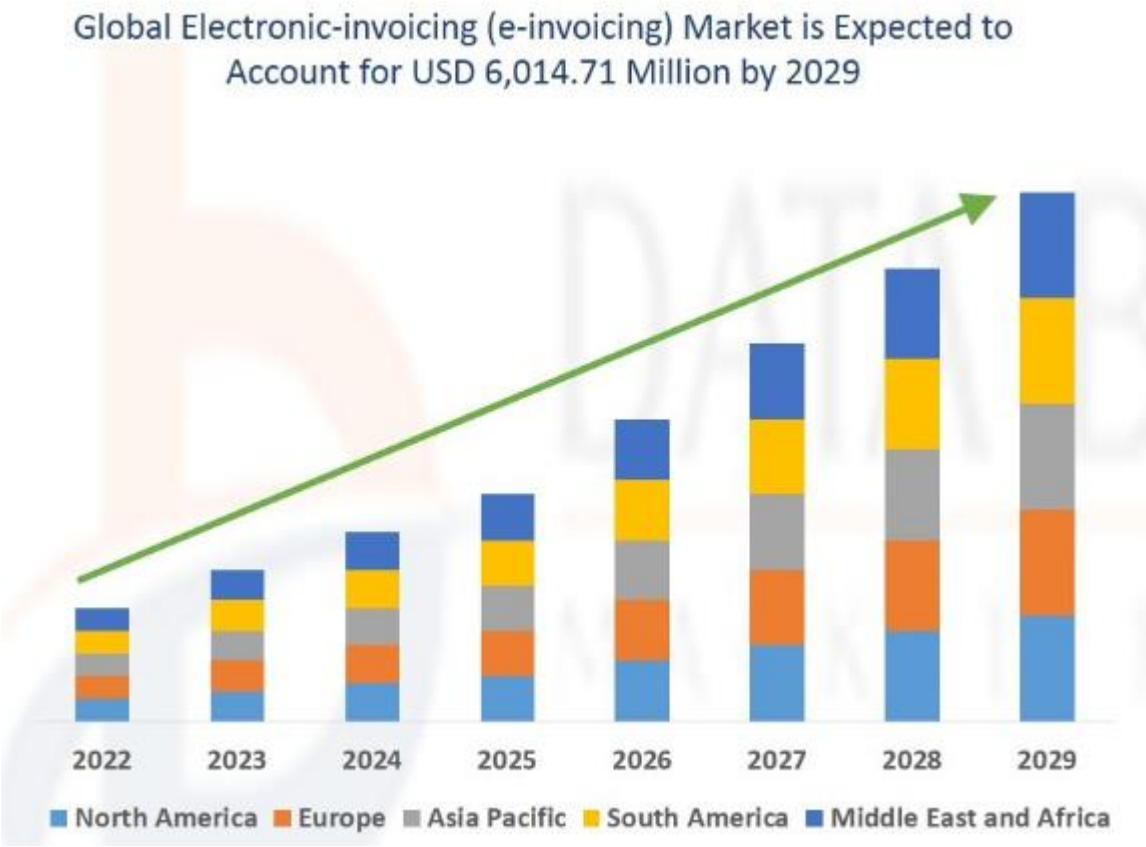

Figure 1.1 Electronic-Invoicing (e-invoicing) Market until 2019

Source Link: https://www.linkedin.com/pulse/electronic-invoicing-e-invoicing-market-size-worth-usd-somesh-devale/

From **Figure 1.1**, the analysis on trends in the number of electronic invoicing underscores compelling reasons to initiate this project, highlighting the inefficiencies and challenges associated with manual invoice processing. This prompts the utilization of advanced technologies like the Gemini API to streamline operations and enhance accuracy, compliance, and decision-making for businesses.

1.2 PROBLEM STATEMENT AND OBJECTIVES

PROBLEM STATEMENT

The current expense creation process necessitates users to manually input expense details into the application, including invoice number, date, amount, and vendor details, despite this information being readily available in the uploaded invoice. This redundant manual data entry process results in inefficiency and time consumption.

To address this issue, there is a need to implement a solution that automates the population of expense details by extracting relevant information directly from the uploaded invoice. This automated approach would not only streamline the expense creation process but also enhance user experience by reducing manual intervention and minimizing the risk of data entry errors.

Therefore, the problem statement revolves around the inefficiency and redundancy of manual expense data entry processes, emphasizing the necessity for an automated solution to extract and populate expense details from uploaded invoices, ultimately leading to improved efficiency, user experience, and time savings.

OBJECTIVES

The objectives of implementing an automated expense creation system are as follows:

- **Automated Data Population**: Develop a system capable of extracting relevant expense details from uploaded invoices, including invoice number, date, amount, and vendor details, to populate expense forms automatically.
- **Accuracy Verification:** Implement mechanisms for users to review and validate automatically populated expense details, ensuring accuracy and completeness before submission.
- **User-Friendly Interface:** Design an intuitive user interface that facilitates seamless interaction, allowing users to easily review, edit, and submit expense details with minimal effort.
- **Time Efficiency**: Reduce the time required for expense creation by eliminating manual data entry tasks, enabling users to create expenses quickly and efficiently.
- **Error Reduction**: Minimize the risk of data entry errors and discrepancies by automating the extraction and population of expense details from invoices, thereby

improving data accuracy and integrity.

- **Enhanced User Experience**: Enhance user satisfaction and engagement by providing a streamlined and efficient expense creation process that requires minimal manual intervention and offers a seamless user experience.

- **Integration Capabilities**: Ensure compatibility and integration with existing expense management systems or accounting software to facilitate seamless data transfer and workflow integration.

- **Scalability and Adaptability**: Develop a solution that is scalable and adaptable to accommodate varying invoice formats, languages, and regional requirements, catering to the diverse needs of users across different industries and sectors.

- **Compliance and Security**: Implement measures to ensure compliance with data privacy regulations and security standards, safeguarding sensitive financial information and protecting user data from unauthorized access or breaches.

- **Performance Optimization**: Optimize system performance and efficiency to handle large volumes of invoice data and support concurrent user access, ensuring smooth and responsive operation even during peak usage periods.

1.3 SCOPE AND LIMITATIONS OF THE PROJECT

Scope:

- The project encompasses the development of an automated expense creation system aimed at extracting relevant expense details from uploaded invoices to populate expense forms automatically.

- It includes the implementation of mechanisms for users to review and validate automatically populated expense details before submission.

- The project involves designing a user-friendly interface for seamless interaction, allowing users to easily review, edit, and submit expense details.

- Integration with existing expense management systems or accounting software to facilitate data transfer and workflow integration is within the scope.

- The solution aims to enhance user experience, reduce time consumption, minimize data entry errors, and improve efficiency in expense creation processes.

Limitations:

- The project's scope may be limited by the complexity of invoice formats and the variability of vendor details, which may pose challenges in accurately extracting and populating expense details.

- The effectiveness of the automated expense creation system may depend on the quality and clarity of the uploaded invoices, and it may not be able to process invoices with poor quality or illegible text effectively.

- The system may have limitations in handling non-standard invoice formats or invoices in languages other than the primary language supported by the system.

- Integration with existing systems may require customizations or compatibility checks, which could introduce complexities and limitations based on the capabilities of the existing systems.

- Compliance with data privacy regulations and security standards is essential but may impose constraints on data processing and access, impacting the system's functionality and performance.

- Scalability and adaptability to accommodate future changes or updates in invoice formats, regulations, or user requirements may pose challenges and limitations, requiring ongoing maintenance and enhancements to the system.

CHAPTER 2

PROJECT ARCHITECTURE, DESIGN AND IMPLEMENTATION

2.1 SYSTEM ARCHITECTURE

The system architecture for the Streamlit application designed to streamline invoice processing involves several key components, including the user interface, integration with the Gemini API for invoice extraction, and a backend database for storing and managing invoice data. Here's an in-depth report detailing the architecture based on the provided points:

1. **User Interface (UI):**
   • The Streamlit application serves as the user interface, providing users with a seamless experience for uploading invoice files.
   • Users are prompted to upload either image or PDF files containing invoices, and they can easily interact with the application to initiate the extraction process.
   • The UI also presents the extracted invoice details in a comprehensible format, allowing users to review and analyze the information.

1. **Integration with Gemini API:**
   • The application leverages the advanced capabilities of the Gemini API, a generative AI model developed by Google, for invoice extraction.
   • Specialized functions tailored to handle different input formats (images and PDFs) interact with the Gemini API to process invoice content.
   • For images, the application uses the "get_gemini_response_image" function to transmit the image alongside an input prompt to the Gemini model.
   • For PDFs, the application initially extracts textual content using the "get_pdf_text" function before submitting it, along with the prompt, to the Gemini model via the "get_gemini_response_pdf" function.

2. **Backend Database:**
   • The backend of the application utilizes the Super Bass Acid Database, integrated with PostgreSQL, for storing and managing invoice data.
   • PostgreSQL was chosen for its integrated support with Super Bass Acid Database and its robust features, including row-level security and support for JSON data.
   • The database consists of tables such as "Users" for authentication and storing user details, and "Invoices" for storing all uploaded invoices.
   • Session management is employed to filter invoices uploaded by specific users,

ensuring data privacy and security.
• Invoice data extracted from the Gemini API is stored in the database, allowing users to view and filter invoices conveniently.

3. **Data Processing and Presentation:**
   • Extracted invoice details from the Gemini API are processed and presented within the Streamlit interface.
   • The application provides users with the ability to review and analyze invoice information, including invoice numbers, dates, total amounts, currency types, item quantities, and customer names.
   • Users can also access analytics detailing month-wise and year-wise expenditure, offering valuable insights into invoicing trends.

4. **User Interaction and Control:**
   • The UI offers users control over the extracted invoice details, allowing them to edit fields if necessary before final submission to the database.
   • Users have the flexibility to filter invoices based on various criteria, such as invoice name, amount, or item number, enhancing usability and functionality.

2.2 OVERVIEW OF THE DESIGN PROCESS

The main design process of Text Extraction revolves around optimizing the extraction of crucial details from invoices through a systematic approach shown in Figure 2.1.

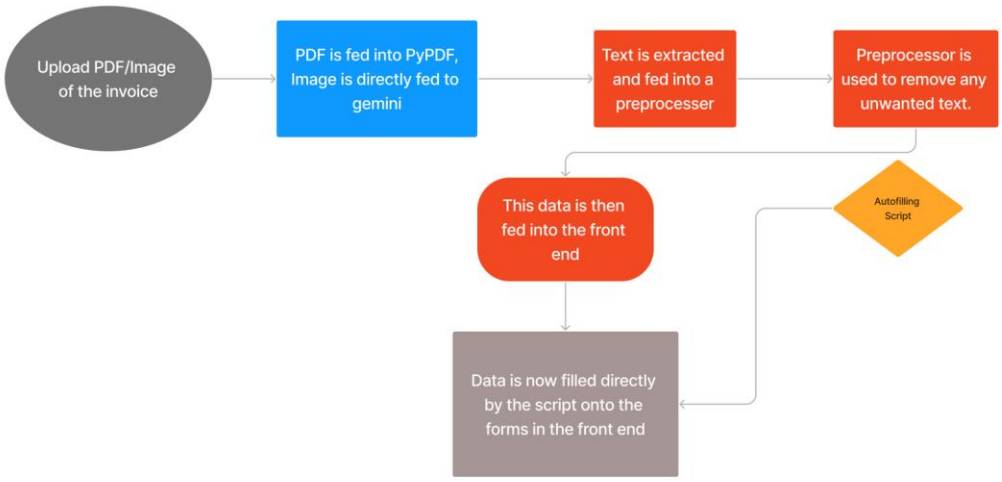

Figure 2.1 Text Extraction Process

2.3 EXPLANATION OF THE ENGINEERING PRINCIPLES USED IN THE DESIGN

Week(s)	*Section*	*Description*	*Efficiency Gains*	*Work Done*
Week 1	**Project Initialization**	Conceptualization, requirement gathering, and initial design for the application, focusing on the user interface and experience.	Lays the groundwork for a streamlined development process.	Project kickoff, defining the scope, and initial planning.
Week 2-3	**User Interface & Input Processing**	Design and testing of the Streamlit interface; development of functions for image and PDF input processing.	Intuitive UI reduces training time; automation cuts down preprocessing time.	Interface development, input processing functions, usability testing.
Week 4	**API & Backend Setup**	Integration of the Gemini API; setup of Supabase with PostgreSQL and JSON data compatibility for various invoice formats.	API reduces data extraction time; Supabase enhances backend efficiency.	API integration and optimization; database setup and security configuration.
Week 5	**Database Schema & Session Management**	Creation of the database schema; implementation of session management logic to personalize data retrieval.	Structured schema ensures quick data retrieval; session queries improve security.	Database schema normalization and indexing; session management implementation.
Week 6	**Data Management & Efficiency Metrics**	Data validation, standardization, and user verification; benchmarking and performance testing for efficiency gains.	Reduction in data inconsistencies; benchmarking ensures efficiency gains.	Data management logic development; performance testing and efficiency tracking.
Week 7	**Analytics Integration**	Development of analytics algorithms, data visualization tools, and reporting features for monthly and yearly financial insights.	Quick access to financial analytics aids in decision-making.	Analytics feature development and integration with the UI.
Week 8	**Invoice Management & Finalization**	Finalizing invoice management systems; implementation of user-editable fields and filter logic; continuous integration and deployment setup for updates and server provisioning.	Facilitates organization and retrieval speed of invoices; smooth updates.	Completion of invoice management system; setting up CI/CD pipelines; final deployment and project wrap-up.

Table 2.1 Backend Integration using Supa Base

2.4 DESCRIPTION OF THE STEPS TAKEN TO IMPLEMENT THE PROJECT DESIGN

Project Execution Summary

1. *Exploration of Text Extraction Methods*: Initially, the project team explored various text extraction methods available in the market. Three main extraction processes were considered: AWS Textract with S3 and DynamoDB, Easy OCR, and PyPDF along with the Gemini API.

2. *Testing and Evaluation*: Each text extraction method was tested and evaluated for its effectiveness in accurately extracting essential details from invoices. AWS Textract with S3 and DynamoDB was successful, but Easy OCR did not meet the desired accuracy levels.

3. *Comparison and Selection*: After thorough testing and evaluation, a comparison was made between PyPDF with the Gemini API and AWS Textract. While both methods were successful, PyPDF with the Gemini API demonstrated higher accuracy in extracting invoice details.

4. *Preference for PyPDF with Gemini API*: Based on the comparison results and the project team's evaluation, it was decided to proceed with PyPDF along with the Gemini API for text extraction in the project implementation phase.

5. *Implementation and Integration*: The chosen text extraction method, PyPDF with the Gemini API, was implemented and integrated into the project framework. This involved setting up the necessary infrastructure, configuring APIs, and developing the required functionality to enable seamless extraction of invoice details.

6. *Testing and Refinement*: The implemented text extraction process was rigorously tested to ensure its reliability, accuracy, and performance. Any issues or inconsistencies were identified and addressed through iterative refinement and optimization.

7. *Deployment and Rollout*: Once the text extraction process was deemed stable

and reliable, the project was integrated with Streamlit UI, and the extraction functionality was made available to users.

Invoice Extraction Model Implementation

1. *User Interface and Input Handling:*
    a. The application provides users with an intuitive interface to upload invoice files, whether in image or PDF format.
    b. Upon accessing the application, users are prompted to upload their invoice files, initiating the extraction process.

2. *Content Processing with Gemini API*:
    a. Uploaded invoice files are processed using the Gemini API, a cutting-edge generative AI model developed by Google.
    b. For images, the application utilizes the get_gemini_response_image function to transmit the image alongside an input prompt to the Gemini model.
    c. For PDF files, the application employs the get_pdf_text function to extract textual content before submitting it, along with the prompt, to the Gemini model via the get_gemini_response_pdf function.

3. *Data Extraction and Analysis:*
    a. The Gemini model analyzes the uploaded content, extracting critical invoice details such as invoice numbers, dates, total amounts, currency types, item quantities, and customer names.
    b. Leveraging its advanced capabilities, the Gemini model comprehends natural language prompts and generates coherent responses, enabling efficient extraction of meaningful information from image content.

4. *Response Presentation within Streamlit Interface*:
    a. Once the Gemini API processes the input provided by the application, it generates a detailed response encapsulating the extracted invoice details.
    b. The response is then parsed and presented within the Streamlit interface, providing users with a convenient means to review and analyze the identified invoice information.

5. *Database Storage of Extracted Data:*
    a. Upon successful extraction of invoice details by the Gemini API, the data is

perfectly separated into distinct categories such as date, name, invoice number, total amount, currency type, item quantities, and customer names.

b. These extracted details are then stored in a database, ensuring efficient organization and accessibility of invoice data.
c. The database serves as a centralized repository for storing and managing extracted invoice information, facilitating easy retrieval and analysis

Backend Integration

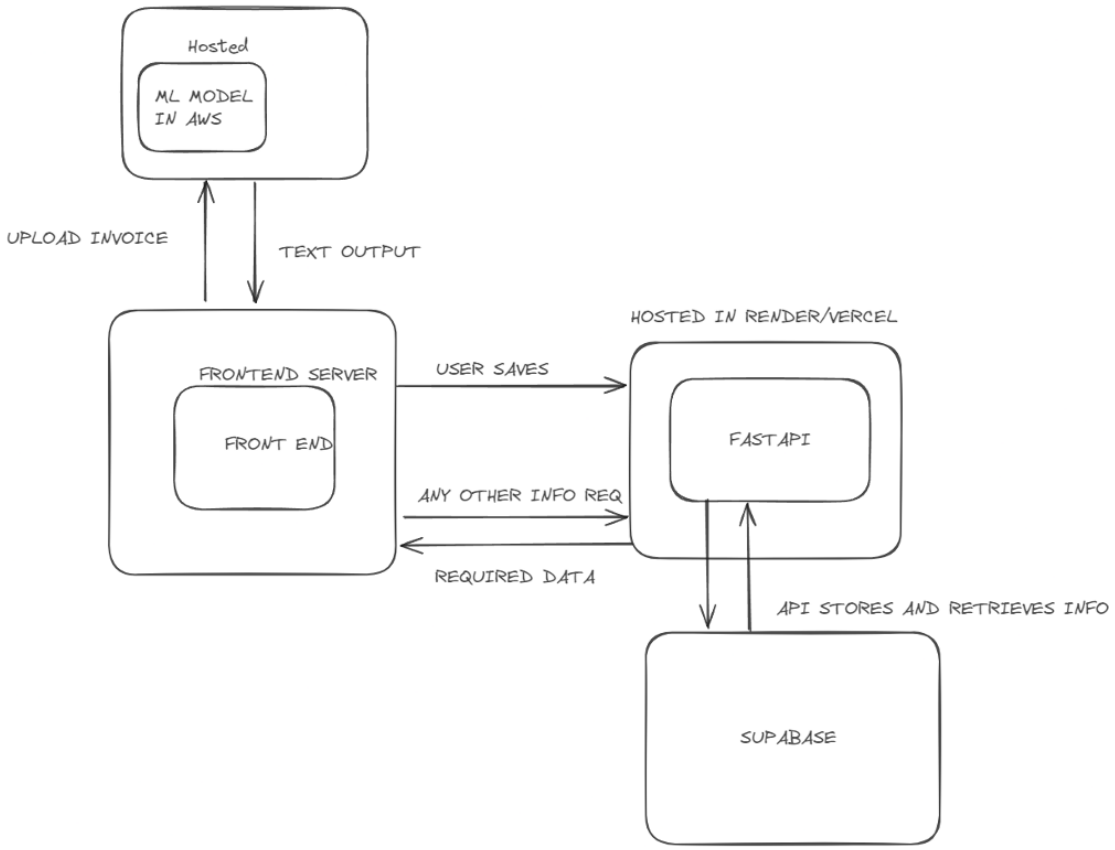

Figure 2.2 Backend Integration using Supa Base

The backend implementation of the Invoice Management Tool leverages a combination of Supabase, Google Sheets, and external APIs to provide a robust and efficient solution for managing invoices.

1. *Supabase Integration*:

- Purpose: Supabase serves as the primary backend service for storing and managing invoice data.

- Features:
  - Real-time database: Supabase offers real-time capabilities, ensuring instant updates to the database.
  - Authentication: Supabase handles user authentication securely.
  - Data storage: Invoice data is stored in a PostgreSQL database managed by Supabase.

2. *Google Sheets Integration:*
- Purpose: Google Sheets is utilized for specific functionalities such as data import/export.
- Features:
  - Data Import: Invoices can be imported from Google Sheets for processing.
  - Data Export: Processed invoice data can be exported to Google Sheets if needed.

3. *External APIs:*
- Purpose: External APIs are utilized for AI-powered invoice processing.
- Features:
  - Image and PDF Processing: AI models provided by external services extract invoice details from images and PDF documents.
  - Natural Language Understanding: AI models process text inputs to extract relevant invoice information.

4. *Error Handling:*
- Purpose: Error handling mechanisms are implemented to gracefully handle exceptions and provide meaningful feedback to users.
- Features:
  - Specific Error Messages: Errors such as duplicate invoices, data type mismatches, and invalid inputs are caught and handled appropriately.
  - User-Friendly Feedback: Error messages are displayed to guide users on resolving issues.

5. *Database Operations:*
- Purpose: Database operations are performed to store and manage invoice data.
- Features:
  - Data Insertion: Processed invoice details are inserted into the Supabase database.

- Data Retrieval: Invoices are retrieved from the database for display and analysis.
- Data Validation: Input data is validated to ensure accuracy and integrity before insertion.

6. *dotenv and Environment Variables:*

- Purpose: dotenv is used to load environment variables from a .env file into the environment.
- Benefits:
  - Security: Sensitive information such as API keys and credentials are kept separate from the code and securely managed.
  - Portability: Environment variables allow configurations to be easily adjusted without modifying the code.

Why Supabase?

Supabase is chosen as the backend solution for its simplicity, scalability, and rich feature set. It offers real-time capabilities, authentication, and data storage, making it an ideal choice for building modern web applications. Additionally, its integration with PostgreSQL ensures robust database operations and scalability for growing applications.

CHAPTER 3

RESULTS AND ANALYSIS

3.1 VALIDATION PROCEDURES

1 *Image Format Testing:*
The validation process began with testing invoices in image format. Various types of invoices were uploaded to assess the system's performance in extracting details accurately.

2 *PDF Format Testing:*
Following image format testing, the validation continued with invoices provided in PDF format. This step aimed to evaluate the system's capability to extract information from structured documents.

3 *Image Format with Handwritten Invoices:*
Handwritten invoices were introduced in image format to test the system's ability to interpret handwritten text and extract relevant details effectively.

4 *PDF with Handwritten Invoices:*
Similar to the previous step, handwritten invoices were included in PDF format to assess the system's performance in handling handwritten content within structured documents.

5 *Invoices with Scribbled Marks:*
To further evaluate the system's robustness, invoices containing scribbled marks and irregularities were tested. This step aimed to identify any challenges in processing invoices with additional markings or anomalies.

These validation procedures were carried out systematically to ensure comprehensive testing of the system across different invoice formats and conditions.

3.2 UI & TEST RESULTS

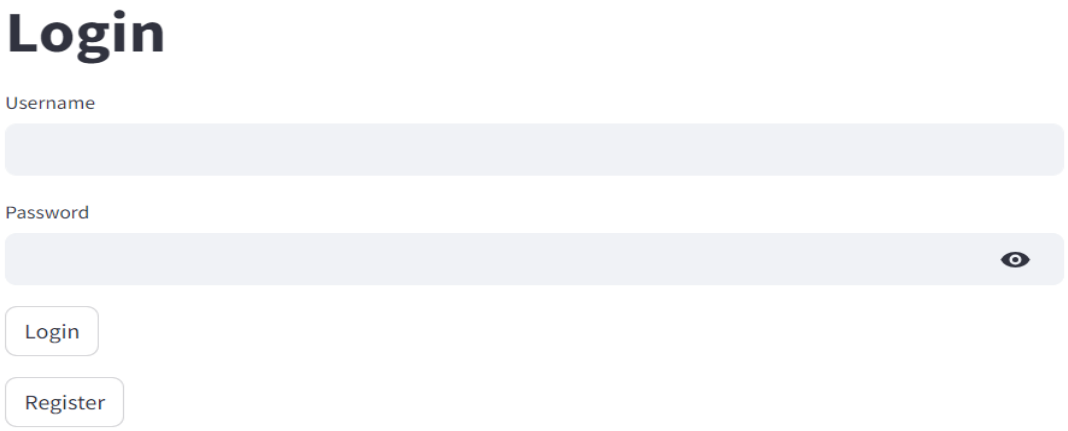

Figure 3.1 Login Page

This is the landing page users see upon opening the application. Registered users can log in using their credentials, while new users can register themselves by clicking the "Register" button, which redirects them to the registration page.

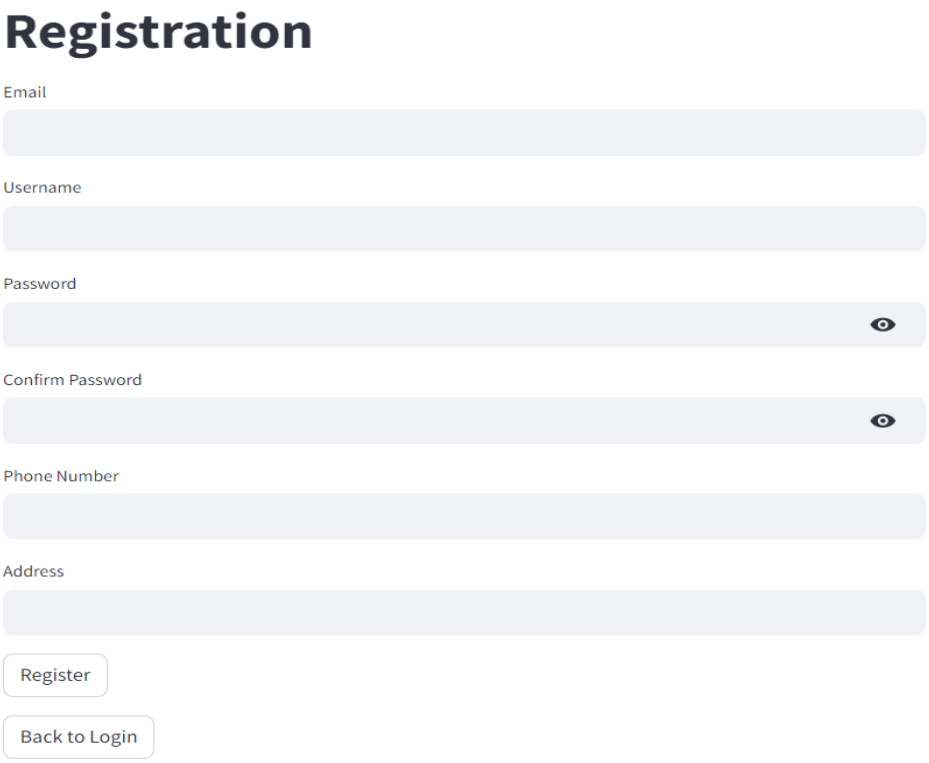

Figure 3.2 Registration Page

A first-time user can register them using this page. During registration, the user has to enter a few details and then press the "Register" button.

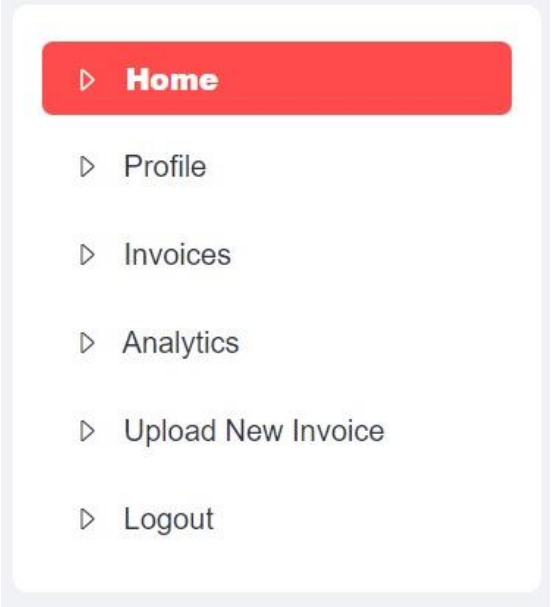

Figure 3.3 Nav Bar

The navigation bar facilitates redirection between various pages, such as the profile page, upload invoice page, etc. Additionally, the navbar provides an option to log out from the application.

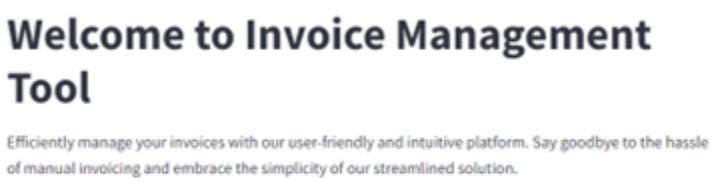

Figure 3.4 Home Page

A straightforward homepage where users are redirected after logging in.

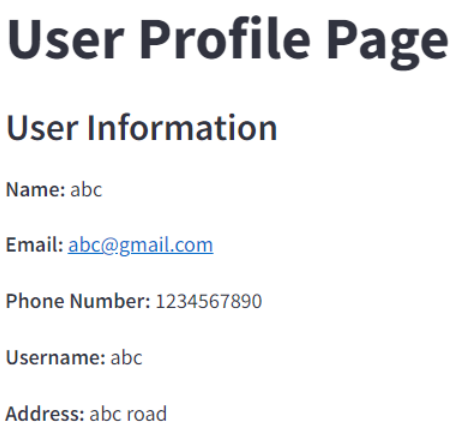

Figure 3.5 Profile Page

The user profile page displays information about the currently logged-in user.

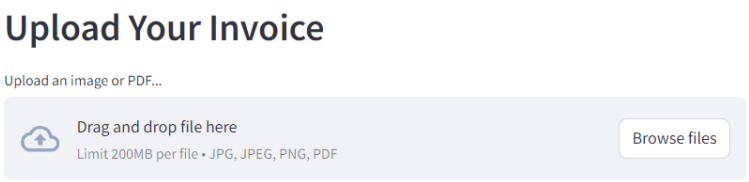

Figure 3.6 Upload Invoice Page

The "Upload Invoice" page is where users can submit new invoices. These invoices can be either in image format or PDF format. Once uploaded, the invoice is sent to the backend and stored there.

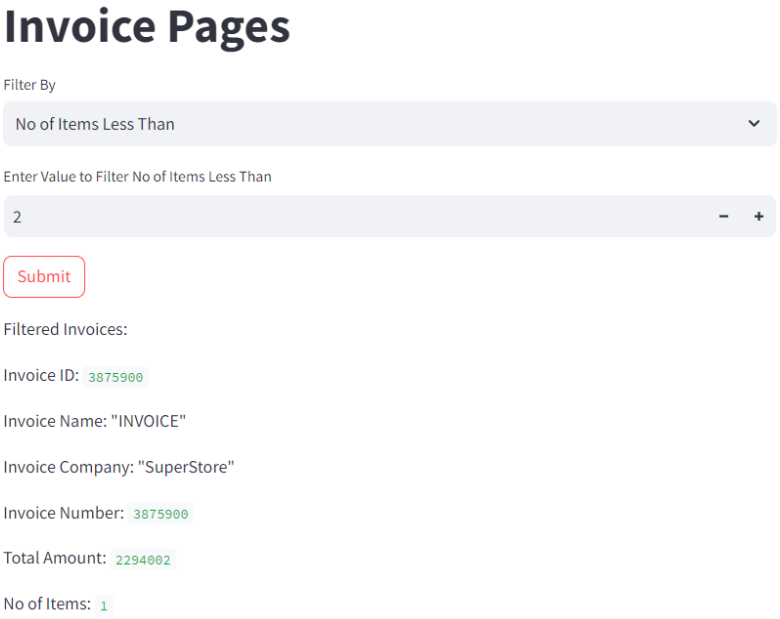

Figure 3.7 Invoices Page

On this page, users can access previously uploaded invoices. They also have the option to filter invoices based on either the number of items or the total invoice amount.

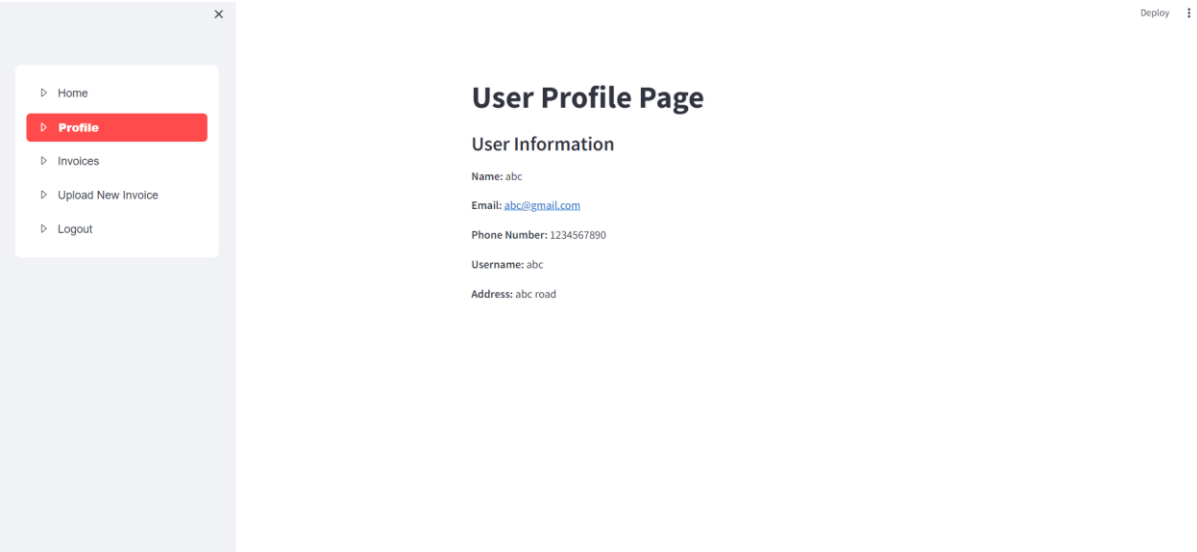

Figure 3.8 The User Interface (UI) as a Whole

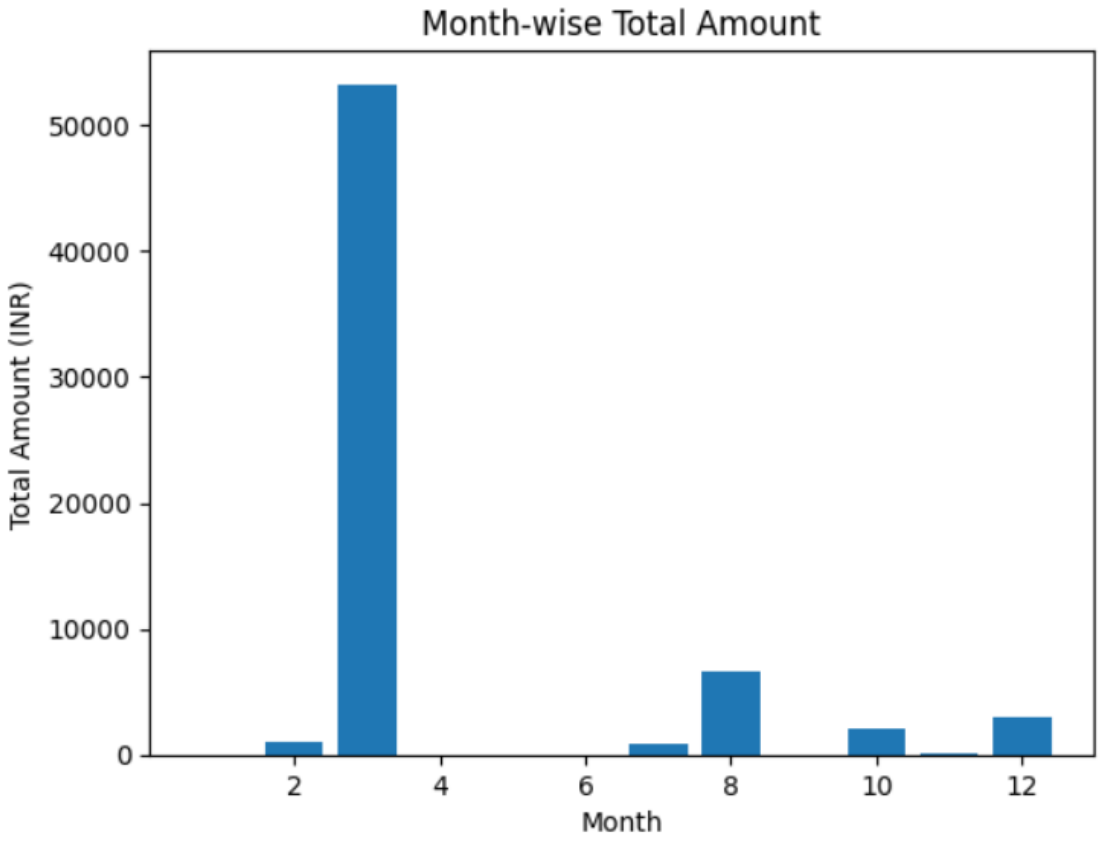

Figure 3.9 Month Wise Income Analysis

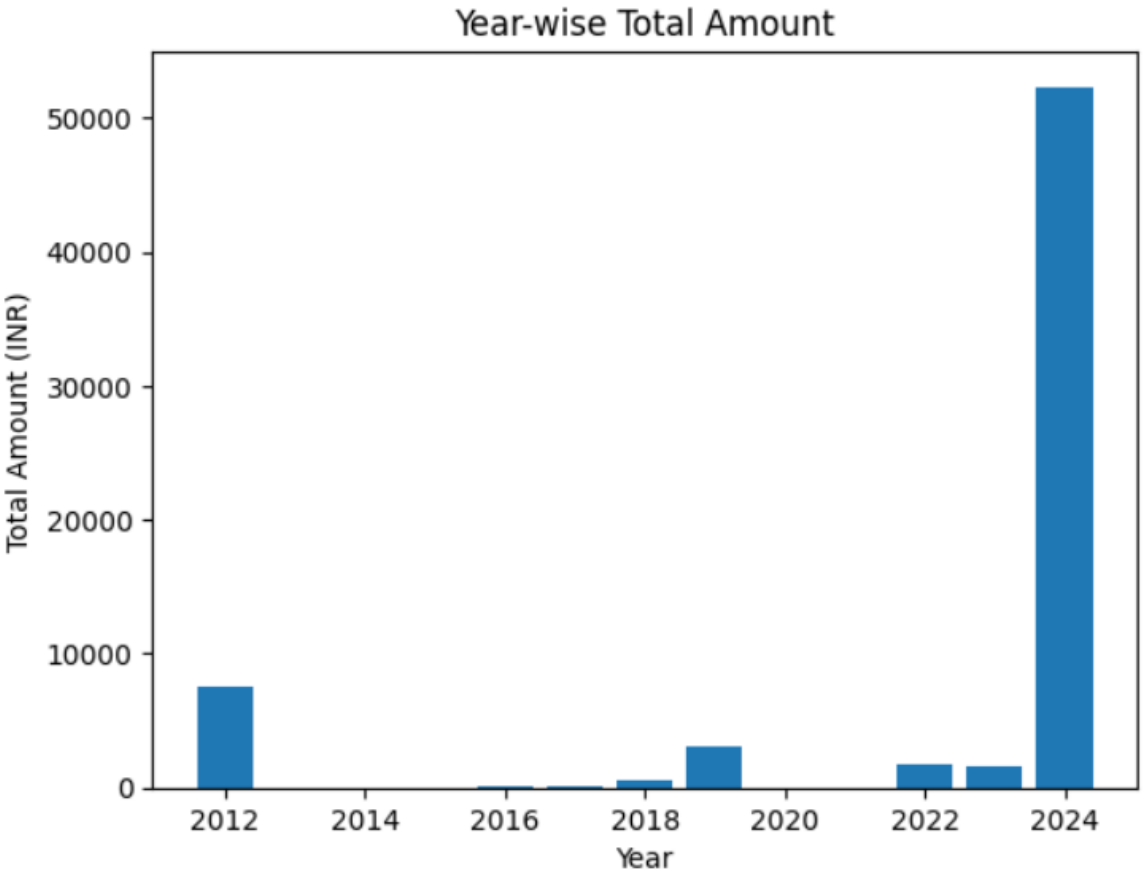

Figure 3.10 Year Wise Income Analysis

3.3 ANALYSIS OF RESULTS

The analysis of the results reveals that the text extraction process is functioning effectively, accurately retrieving essential details from invoices. Specific information such as Invoice ID, Name, Company, Invoice Number, Total Amount, and Number of Items are consistently extracted, with NULL values appropriately assigned if any particulars are unavailable. Additionally, the system provides comprehensive month-wise and year-wise income analysis, enhancing its functionality for financial assessment. Overall, the results demonstrate a robust and comprehensive outcome, indicating the successful execution of the project objectives.

CHAPTER 4

LEARNING OUTCOMES

The implementation of the invoice management system has yielded several significant learning outcomes, contributing to both technical and practical knowledge. Firstly, the project provided invaluable experience in working with cutting-edge technologies such as AI-powered OCR (Optical Character Recognition) and API integration, enhancing proficiency in leveraging advanced tools for real-world applications. This hands-on exposure enabled the team to grasp fundamental concepts of machine learning and natural language processing, paving the way for future endeavors in AI-driven solutions.

Moreover, the project fostered collaboration and teamwork among team members, emphasizing the importance of effective communication and coordination in achieving project goals. Through regular meetings, brainstorming sessions, and collaborative problem-solving, the team cultivated strong interpersonal skills and a shared sense of responsibility towards project success. These interpersonal dynamics proved instrumental in navigating challenges, fostering innovation, and maintaining project momentum throughout its lifecycle.

Additionally, the project provided valuable insights into the intricacies of invoice processing and financial management, offering practical exposure to real-world business workflows and requirements. By engaging with stakeholders and understanding their needs, the team gained a deeper appreciation for the importance of user-centric design and iterative development methodologies. This user-focused approach not only enhanced the usability of the application but also instilled a customer-centric mindset essential for delivering impactful solutions in diverse domains.

Furthermore, the project served as a platform for continuous learning and skill development, encouraging experimentation, exploration, and adaptation in response to evolving challenges and requirements. The iterative nature of the development process facilitated rapid prototyping and iteration, allowing the team to incorporate feedback, refine features, and improve overall system performance iteratively. This iterative approach to development not only enhanced the final product but also reinforced the importance of agility and adaptability in addressing complex technological and business challenges.

In conclusion, the project's outcomes extend beyond technical skills, encompassing teamwork, user-centric design, and adaptability. Through collaboration, innovation, and continuous learning, it has laid a solid foundation for success in future endeavors, fostering growth and exploration in technology and business innovation.

CHAPTER 5

CONCLUSIONS AND FUTURE WORK

SUMMARY OF PROJECT OBJECTIVES AND ACHIEVEMENTS

Our project sought to enhance invoice processing efficiency through the utilization of advanced AI technology while prioritizing user-friendly interaction. Our key objectives encompassed implementing efficient text extraction methods, integrating external APIs for AI processing, and ensuring a seamless user experience.

Through meticulous planning and execution, we achieved significant milestones. Firstly, we successfully implemented OCR techniques to ensure precise data extraction from invoices. This foundational step laid the groundwork for accurate information retrieval, crucial for subsequent analyses. Integration with the Gemini API marked another major achievement, enabling AI-powered analysis of extracted data. Leveraging the Gemini model's capabilities, we ensured comprehensive extraction of crucial invoice details, ranging from invoice numbers to customer names.

Moreover, our focus on user-centric design yielded an intuitive interface, simplifying the invoice uploading and analysis process. This aspect was crucial in ensuring accessibility and ease of use for all users, regardless of their technical expertise.

In summary, our project has not only met but surpassed its objectives. We've successfully implemented advanced AI technologies, streamlined invoice processing workflows, and prioritized user experience. Our achievements represent a significant step forward in optimizing invoice management processes and laying the foundation for future innovations in this domain.

DISCUSSION OF LIMITATIONS AND AREAS OF IMPROVEMENT

While our project successfully addressed the broader need for streamlined invoice processing, there are several limitations and areas for improvement worth considering.

1. *Limited Focus on Company-Specific Needs*: One of the key limitations is the project's generalized approach, catering to invoice processing needs across various companies. While this ensures broad applicability, it may overlook specific requirements unique to individual companies. Future iterations could incorporate more customizable features tailored to the specific needs and workflows of different organizations.

2. *Enhanced Insights through Company-Based Analysis*: While our project provides comprehensive analyses of invoice contents, there is room for improvement in offering more nuanced insights. By incorporating company-specific data analytics and reporting functionalities, users could gain deeper insights into their invoicing trends, expenditure patterns, and areas for optimization.

3. *Optimization of OCR and AI Models*: Although our project successfully leverages OCR and AI technologies for text extraction and analysis, there is potential for further optimization. Fine-tuning OCR algorithms and AI models to handle diverse invoice formats, languages, and handwriting styles could improve accuracy and efficiency, ensuring more reliable results across a wider range of documents.

4. *Integration with Enterprise Systems*: Integrating the invoice management tool with existing enterprise systems, such as accounting software or ERP (Enterprise Resource Planning) systems, could enhance workflow efficiency and data synchronization. Seamless data exchange between systems would streamline processes and minimize manual data entry, reducing the risk of errors and improving overall productivity.

5. *User Feedback and Iterative Development:* Incorporating user feedback mechanisms and adopting an iterative development approach could drive continuous improvement in the project. Soliciting feedback from users regarding usability, feature requests, and pain points would provide valuable insights for refining and enhancing the application over time.

6. *Data Security and Compliance*: While data security measures are implemented in the current design, continuous monitoring and updates are necessary to ensure compliance with evolving data protection regulations. Regular audits, encryption protocols, and user access controls should be maintained to safeguard sensitive financial information and ensure regulatory compliance.

In conclusion, while our project addresses the overarching goal of streamlined invoice processing, there are several avenues for improvement and refinement. By focusing on company-specific needs, enhancing data analytics capabilities, optimizing OCR and AI models, integrating with enterprise systems, soliciting user feedback, and ensuring robust data security, future iterations of the project can further elevate its effectiveness and impact in the realm of invoice management.

RECOMMENDATIONS FOR FUTURE RESEARCH OR DEVELOPEMENT

1. *Automatic Date Formatting*: Implementing automated date formatting capabilities would enhance the efficiency of invoice processing. By automatically recognizing and standardizing date formats across various documents, users can save time and ensure consistency in data interpretation.

2. *Enhanced Language Support*: Expanding language support for OCR and AI models would improve the applicability of the invoice management tool in diverse linguistic environments. Investing in research to develop multilingual text extraction and analysis capabilities would cater to a broader user base and facilitate global adoption.

3. *Advanced Data Visualization*: Integrating advanced data visualization techniques would enable users to gain deeper insights from their invoice data. Researching and implementing interactive dashboards, customizable reports, and predictive analytics features would empower users to identify trends, anomalies, and opportunities for optimization more effectively.

4. *Integration with Blockchain Technology:* Exploring integration with blockchain technology could enhance the security and traceability of invoice data. Researching blockchain-based solutions for invoice validation, authentication, and auditing could provide additional layers of trust and transparency in invoice management processes.

5. *Machine Learning for Document Classification*: Leveraging machine learning algorithms for document classification could streamline document categorization and routing. Researching and developing automated classification models capable of identifying different types of invoices, such as utility bills, purchase orders, and expense reports, would improve workflow automation and organization.

6. *Robotic Process Automation (RPA*): Investigating the integration of RPA technology for automating repetitive invoice processing tasks could further streamline workflows. Researching RPA solutions for tasks such as data entry, invoice reconciliation, and approval routing could reduce manual intervention and accelerate invoice processing cycles.

By focusing on these areas for future research and development, the field of invoice management can continue to evolve, offering enhanced functionalities, improved efficiency, and greater value to users and organizations alike.

REFERENCES

1. A.Manjunath, Akanksh, M. Nayak, N. Santhanam, S. Pandit, S. Sunkad, P. Deenadhayalan, and S. Gangadhara, "Automated invoice data extraction using image processing," in IAES International Journal of Artificial Intelligence (IJ-AI), vol. 12, pp. 514-521, 2023, doi: 10.11591/ijai.v12.i2.pp514-521.

2. K. Kapoor, "Invoice Processing using OCR APIs," 2023.

3. J. K. Raju and R. Prabhakar, "A Comparative Study of Optical Character Recognition for Tamil Script," in European Journal of Scientific Research, vol. 35, 2009.

4. K. M. Yindumathi, S. S. Chaudhari, and R. Aparna, "Analysis of Image Classification for Text Extraction from Bills and Invoices," in Proc. 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2020, pp. 1-6, doi: 10.1109/ICCCNT49239.2020.9225564.

5. C. De Trogoff, R. Hantach, G. Lechuga, and P. Calvez, "Automatic Key Information Extraction from Visually Rich Documents," in Proc. 2022 21st IEEE International Conference on Machine Learning and Applications (ICMLA), Nassau, Bahamas, 2022, pp. 89-96, doi: 10.1109/ICMLA55696.2022.00020.

6. N. Rahal, M. Tounsi, M. Benjlaiel, and A. M. Alimi, "Information Extraction from Arabic and Latin scanned invoices," in Proc. 2018 IEEE 2nd International Workshop on Arabic and Derived Script Analysis and Recognition (ASAR), London, UK, 2018, pp. 145-150, doi: 10.1109/ASAR.2018.8480221.

7. H. Sidhwa, S. Kulshrestha, S. Malhotra, and S. Virmani, "Text Extraction from Bills and Invoices," in Proc. 2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), Greater Noida, India, 2018, pp. 564-568, doi: 10.1109/ICACCCN.2018.8748309.

8. X. HoangVan, P. TranQuang, M. DinhBao, and T. VuHuu, "Developing an OCR Model for Extracting Information from Invoices with Korean Language," in Proc. 2023 International Conference on Advanced Technologies for Communications (ATC), Da Nang, Vietnam, 2023, pp. 84-89, doi: 10.1109/ATC58710.2023.10318877.

9. T. Saout, F. Lardeux, and F. Saubion, "An Overview of Data Extraction From Invoices," in IEEE Access, vol. 12, pp. 19872-19886, 2024, doi: 10.1109/ACCESS.2024.3360528.

10. L. Liu, B. Wang, X. He, J. Wang, Y. Zheng, and Y. Yan, "Establishing an electronic invoice management platform based on information system," in Journal of Physics: Conference Series, vol. 2004, 2nd International Conference on Big Data Mining and Information Processes (BDMIP 2021), Xiamen, China, 2021, doi: 10.1088/1742-6596/2004/1/012013.

11. A. Kumar and N. Awasthi, "An efficient algorithm for text localization and extraction in complex video text images," in Proc. 2013 2nd International Conference on Information Management in the Knowledge Economy, Chandigarh, India, 2013, pp. 14-19.

12. G. G. Devi and C. P. Sumathi, "Text extraction from images using gamma correction method and different text extraction methods — A comparative analysis," in Proc. International Conference on Information Communication and Embedded Systems (ICICES2014), Chennai, India, 2014, pp. 1-5, doi:

10.1109/ICICES.2014.7033973.

13. D. Baviskar, S. Ahirrao, V. Potdar, and K. Kotecha, "Efficient Automated Processing of the Unstructured Documents Using Artificial Intelligence: A Systematic Literature Review and Future Directions," in IEEE Access, vol. 9, pp. 72894-72936, 2021, doi: 10.1109/ACCESS.2021.3072900.

14. H. Arslan, "End to End Invoice Processing Application Based on Key Fields Extraction," in IEEE Access, vol. 10, pp. 78398-78413, 2022, doi: 10.1109/ACCESS.2022.3192828.

15. R. Ahmad, M. T. Afzal, and M. A. Qadir, "Information Extraction from PDF Sources Based on Rule-Based System Using Integrated Formats," in Semantic Web Challenges, SemWebEval 2016, Communications in Computer and Information Science, vol. 641, Springer, Cham, 2016, doi: 10.1007/978-3-319-46565-4_23.

16. S. Jiang and Y. Li, "Research and Implementation of PDF Specific Element Fast Extraction," in Proc. 2023 4th International Conference on Big Data & Artificial Intelligence & Software Engineering (ICBASE), Nanjing, China, 2023, pp. 77-83, doi: 10.1109/ICBASE59196.2023.10303081.