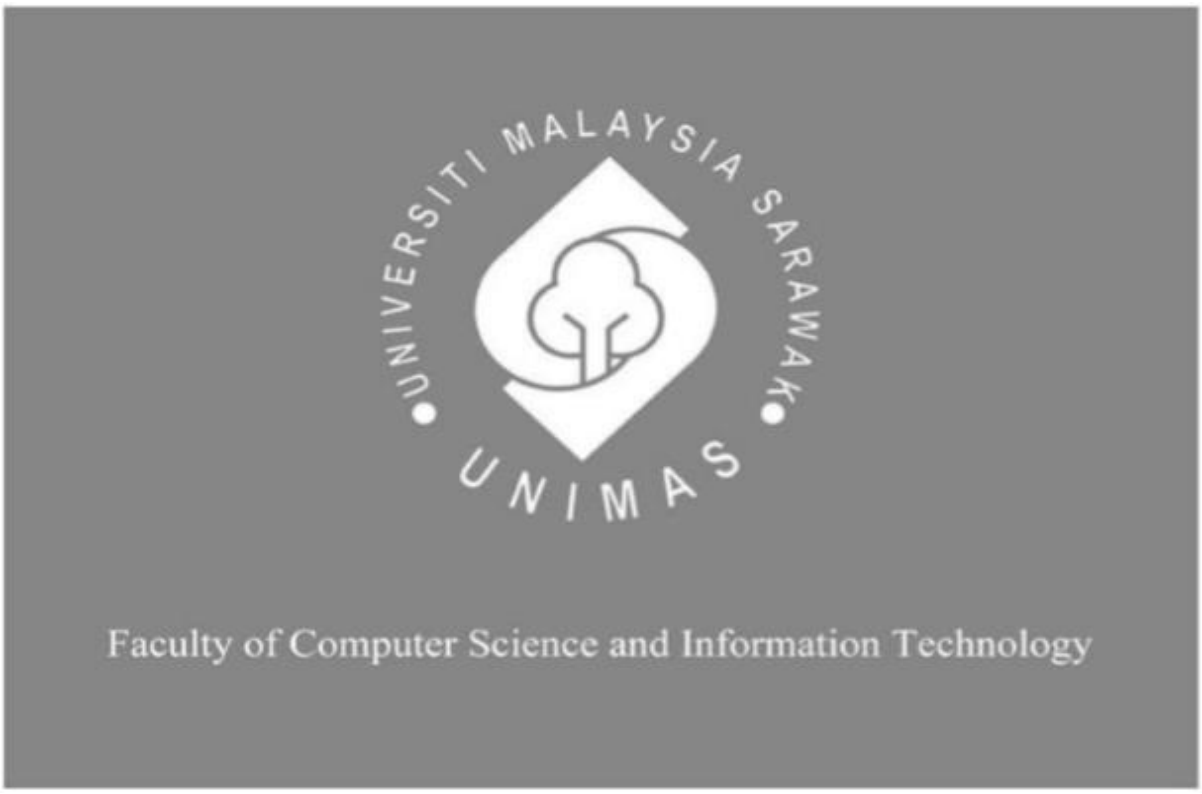


Automatic Text Extraction Using Optical Character Recognition (OCR) for Blood Test Report Management

Kueh Hui Tieng

79802

Bachelor of Computer Science with Honours

(Information System)

2025

Automatic Text Extraction Using Optical Character Recognition (OCR) for Blood Test Report Management

KUEH HUI TIENG

This project is submitted in partial fulfilment of the

requirements for the degree of

Bachelor of Computer Science and Information Technology

Faculty Computer Science and Information Technology

UNIVERSITI MALAYSIA SARAWAK

2025

UNIVERSITI MALAYSIA SARAWAK

THESIS STATUS ENDORSEMENT FORM

TITLE Automatic Text Extraction Using Optical Character Recognition (OCR) for Blood Test Report Management

ACADEMIC SESSION: SESSION 24/25

KUEH HUI TIENG

(CAPITAL LETTERS)

hereby agree that this Thesis* shall be kept at the Centre for Academic Information Services, Universiti Malaysia Sarawak, subject to the following terms and conditions:

1. The Thesis is solely owned by Universiti Malaysia Sarawak
2. The Centre for Academic Information Services is given full rights to produce copies for educational purposes only
3. The Centre for Academic Information Services is given full rights to do digitization in order to develop local content database
4. The Centre for Academic Information Services is given full rights to produce copies of this Thesis as part of its exchange item program between Higher Learning Institutions [or for the purpose of interlibrary loan between HLI]
5. ** Please tick (√)

☐	CONFIDENTIAL	(Contains classified information bounded by the OFFICIAL SECRETS ACT 1972)
☐	RESTRICTED	(Contains restricted information as dictated by the body or organization where the research was conducted)
☑	UNRESTRICTED	

(AUTHOR'S SIGNATURE)

Permanent Address

85, Lorong2B,
Three Hills Parks, 93300,
KUCHING, SARAWAK

Date:23 June 2025

Validated by

(SUPERVISOR'S SIGNATURE)

Ts. Dr Lim Phei Chin
Senior Lecturer
Faculty of Computer Science and Information Technology
Universiti Malaysia Sarawak

Date: 24 June 2025

Note * Thesis refers to PhD, Master, and Bachelor Degree
** For Confidential or Restricted materials, please attach relevant documents from relevant organizations / authorities

ACKNOWLEGEMENT

I would like to express my deepest gratitude to my supervisor, Ts. Dr. Lim Phei Chin, for her invaluable guidance, constructive feedback, and unwavering support throughout the development of this project. Her expertise and encouragement have been instrumental in helping me stay focused and successfully complete the project. I would also like to extend my sincere appreciation to my examiner, Ts. Dr. Hamimah Ujir, for her insightful suggestions and thoughtful feedback, which greatly contributed to enhancing the quality of my work. In addition, I am truly thankful to my beloved family for their continuous support, encouragement, and understanding. Their unwavering belief in me has been a constant source of strength and motivation throughout this journey. Finally, I would like to thank my course mates for their support and assistance during the development of the system. Their collaboration and willingness to help were truly appreciated.

ABSTRACT

The Automatic Medical Data Extraction System is to reduce the process of medical data entry by automating the extraction of data from physical blood test reports. This system utilizes Optical Character Recognition (OCR) and Natural Language Processing (NLP) techniques to efficiently extract, structure, and standardize medical data. Currently, medical staff manually input data from blood test reports into electronic systems, a process prone to human error and inefficiency. This project will eliminate the manual data entry process by developing a system to extract blood test report data from various laboratories. The system will extract relevant data, convert test units to a standardized format, and ensure the accuracy and consistency of the information stored in medical record systems. By automating the extraction and data structuring process, the system will significantly reduce human error, improve the efficiency of medical record management, and enhance the overall quality of patient care. The prototype's performance will be evaluated through metrics such as Character Error Rate (CER), Word Error Rate (WER), and Named Entity Recognition (NER), ensuring the reliability of the extracted data.

ABSTRAK

Sistem Automatik Medical Data Extraction dihasilkan bagi mengurangkan proses kemasukan data perubatan secara manual dengan mengautomasikan pengekstrakan data daripada laporan ujian darah fizikal. Sistem ini menggunakan teknik Pengecaman Aksara Optik (OCR) dan Pemprosesan Bahasa Semula Jadi (NLP) untuk mengekstrak, menyusun, dan menyeragamkan data perubatan dengan lebih cekap. Pada masa ini, kakitangan perubatan perlu memasukkan data daripada laporan ujian darah ke dalam sistem elektronik secara manual, yang terdedah kepada kesilapan manusia dan kurang cekap. Projek ini bertujuan untuk menghapuskan proses kemasukan data secara manual dengan membangunkan sistem yang boleh mengekstrak data laporan ujian darah daripada pelbagai makmal. Sistem ini akan mengekstrak data yang berkaitan, menukar unit ujian kepada format piawai, dan memastikan ketepatan serta konsistensi maklumat yang disimpan dalam sistem rekod perubatan. Dengan mengautomasikan proses pengekstrakan dan penyusunan data, sistem ini dapat mengurangkan kesilapan manusia, meningkatkan kecekapan pengurusan rekod perubatan, dan seterusnya mempertingkatkan kualiti penjagaan pesakit secara keseluruhan. Prestasi prototaip ini akan dinilai melalui metrik seperti Kadar Ralat Aksara (CER), Kadar Ralat Perkataan (WER), dan Pengecaman Entiti Dinamakan (NER) bagi memastikan kebolehpercayaan data yang diekstrak.

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

Abbreviation	Meaning
OCR	Optical Character Recognition
NLP	Natural Language Processing
CER	Character Error Rate
WER	Word Error Rate
NER	Named Entity Recognition
AI	Artificial Intelligence
ML	Machine Learning
API	Application Programming Interface
TTS	Text-To-Speech
PDF	Portable Document Format
PNG	Portable Network Graphics
CSS	Cascading Style Sheets
HTML	Hypertext Markup Language
PHP	Hypertext Preprocessor
MySQL	My Structured Query Language
Perl	Practical Extraction and Reporting Language
SQL	Structured Query Language
IC	Identity Card

Table of Contents

CHAPTER 1 INTRODUCTION

1.1 Introduction

The Automatic Medical Data Extraction System is to enhance the efficiency and accuracy of medical data entry by automated the data extraction from physical documents, such as blood test reports. Medical staff are often facing problem when input the data into record systems manually. The blood test report is from multiple laboratories, the test will be a lot. The medical staff must key in one by one for the test and the results. It is very wasting time and facing human error.

The system will support advanced technologies such as Optical Character Recognition (OCR) and Natural Language Processing (NLP) to develop a system that allows users to easily upload or scan the blood test report. Based on the research by Zhou et al. (2023), NLP and OCR can be used for automated data extraction process. According to Bhattacharjee et al. (2022), Natural Language Processing (NLP) is a branch of artificial intelligence (AI) that helps computers to understand, process, and analyse for large amounts of human language data. OCR techniques can be used to convert image to text. (Fleischhacker et al., 2024) The system will automatically extract key medical information, and structure these data using standardized format.

By eliminating the need for manual data entry, the system will significantly reduce the time required for medical staff to process blood test document while enhancing the accuracy of patient records. (Guo et al., 2024) Additionally, the generated reports will be formed in a standard format, that can help for healthcare manners. This will improve not only for record management but also support better decision-making and faster response times in critical healthcare.

The system is designed to be user-friendly, to ensure medical professionals can navigate the application easily and generate reports without extensive training. The intuitive interface will minimize the learning and ensuring seamless into a healthcare organization's existing workflow. By incorporating real-time processing capabilities, the system will further increase the speed and accessibility of patient data.

Finally, the Automatic Text Extraction for Blood Test Report Management System aims to improve medical data management, improve patient care, and enhance healthcare efficiency. This system is to reduce the operation work on medical staff, allowing them to dedicate more time to patient care. Additionally, it aligns with the growing trend of digital in the healthcare sector, promoting innovation and giving future advancements in medical data management.

1.2 Problem statement

At present, there is a lot of OCR scanner, but all of that will be for receipt scanner, bill scanner, QR scanner, License scanner and other template. There is no existing OCR scanner for the blood test report. There is no specific template for OCR scanner to detect the blood test report. There is multiple blood report from different laboratory. Different laboratory report has different template and layout. Even though the test that made are the same but different layout for the report will make the computer cannot detected accurately. This will be the challenges for this system. For example, Figure 1.1 show the report layout from Borneo Medical Centre Laboratory. Figure 1.2 show the report layout from Gribbles Laboratory. Figure 1.3 show the report layout from Pathlab Laboratory. These three examples illustrate that different laboratory have their own layout and template, but the test that they perform are the same. The test units will be also different depends on the laboratory.

Medical staff currently manual key in the medical data from physical documents such as blood test reports into electronic systems. This process is not only time wasting but also lead

to human error, which can also lead to inaccuracies in patient records and potentially affect the quality of care provided. The inefficiency of this manual method will be increasing the work for medical staff, and they will make mistakes while making decision. There is no automated solution for accurately extracting and structuring data from physical medical slips. An automated system that can streamline the process, reduce human error, and generate standardized reports is needed to enhance the accuracy and efficiency of medical record management and improve overall patient care.

FULL BLOOD COUNT

Haemoglobin 血红蛋白	*11.9	g/dl	12.0 - 14.0
RBC 红血球计数	4.1	x10^12/L	4.1 - 5.5
Hematocrit (HCT) 红血球比积	*0.36	L/L	0.36 - 0.44
MCV 平均红血球体积	87	fL	73 - 89
MCH 平均红血球蛋白质	29	pg	24 - 30
MCHC 平均红血蛋白浓度	33	g/dL	30 - 37
Platelet Count 血小板	*424	x10^9/L	150 - 400
White Blood Cells 白血球	*19.4	10^9/L	5.0 - 14.5

DIFFERENTIAL COUNT

Neutrophils 嗜中性球	*68	%	13 - 33
Lymphocytes 淋巴球	*21	%	46 - 76
Monocytes 单核球	*10	%	0 - 10
Eosinophils 嗜伊红球	1	%	0 - 6
Basophils 嗜硷性白血球	0	%	0 - 2

Figure 1. 1 Report from Borneo Medical Centre Laboratory.

HAEMATOLOGY

	BLOOD GROUP		A Rh(D) POSITIVE	
	Haemoglobin		144 g/L	(115-165)
	RBC		4.87 x 10^{12}/L	(3.80-5.50)
	PCV		0.46 L/L	(0.35-0.47)
	MCV		94 fL	(78-99)
	MCH		30 pg	(27-32)
	MCHC		314 g/L	(300-360)
	RDW		13.9 %	(11.0-15.0)
**	**White Cell Count**		**12.2** x 10^9/L	(4.0-11.0)
	Neutrophils	63 %	7.7 x 10^9/L	(2.0-8.0)
	Lymphocytes	29 %	3.5 x 10^9/L	(1.0-4.0)
	Monocytes	6 %	0.7 x 10^9/L	(< 1.2)
	Eosinophils	2 %	0.2 x 10^9/L	(< 0.8)
	Basophils	0 %	0.0 x 10^9/L	(< 0.2)
*	**Platelets**		**444** x 10^9/L	(150-400)

Film: The red cells show There is a mild leucocytosis. An occasional atypical lymphocyte seen. There is a mild thrombocytosis.

Validated by Muhammad Zaid Jinis (Dip.MLT-SEGI University)

Figure 1. 2 Report from Gribbles Laboratory.

PATHLAB FB HEALTH SCREEN .....

HAEMATOLOGY .....

ESR	紅血球沉澱率	10	MM/HR	0-20
RBC	紅血球	4.8	X10^12/L	3.9-5.6
HAEMOGLOBIN	血紅素	14.7	G/DL	11.5-16.5
PCV (HCT)	红血球容積量	44	%	35-47
MCV	平均球容積	91	FL	76-96
MCH	平均紅血球血色蛋白	31	PG	27-32
MCHC	平均紅血球血色蛋白濃度	34	G/DL	32-36
PLATELET COUNT	血小板	316	X10^9/L	150-400
WBC	白血球	10.9	X10^9/L	4.0-11.0
DIFFERENTIAL COUNT		.....		
NEUTROPHIL	多形核白血球	71	%	40-75
LYMPHOCYTE	淋巴球	20	%	20-45
MONOCYTE	單核白血球	8	%	2-10
EOSINOPHIL	嗜伊紅白血球	1	%	0-6
BASOPHIL	適鹼性白血球	0	%	0-2
ATYPICAL LYMPHOCYTE		0	%	
PBF	血片檢驗	~		

RBCS ARE NORMOCHROMIC AND NORMOCYTIC.
PLATELETS APPEAR ADEQUATE IN FILM.
NO EARLY WBCS SEEN.

Figure 1. 3 Report from Pathlab Laboratory.

The three figures shows that different laboratory will have distinct layout and template to presenting the blood test report.

1.3 Objectives

1. To study and analyse the techniques for converting blood test report images into structured data to support automated medical data extraction.
2. To develop a semi-automated systems to extract medical data from physical document using OCR.
3. To evaluate accuracy using Character Error Rate, Word Error Rate and Named Entity Recognition.

1.4 Project Scope

The project will focus on developing an automated system to extract and process data from blood test reports using Optical Character Recognition (OCR) techniques and Natural Language Processing (NLP) techniques. This system will be designed to handle multiple formats of blood test report and convert them into structured and standardized data. This project can be detected and process for various test, which is the common blood test over 50 types. To achieve this, the project will involve studying existing techniques for document digitization, text recognition, and data structuring to ensure a comprehensive understanding of the current methodologies. OCR will be implemented to extract text information from blood test report, while NLP will be used to analyse, interpret, and organize the extracted data into a structured format that is suitable to save in the database for the management systems. The accuracy of the system will be evaluated through the measurement of Character Error Rate, Word Error Rate and Named Entity Recognition.

1.5 Significance of project

The project will significantly enhance the efficiency of medical data entry by semi-automating the extraction and processing the information from blood test report, it will help to reduce the work for medical staff and minimize human error. This will help more accuracy for the patient records saved in the management systems and improved the overall patient care. Additionally, the generated reports will support better data management and healthcare decision-making.

By generating standard and structured reports, the project will support better data management, enable medical staff to access and analyse medical information efficiently. The system able to handle various layouts and templates of blood test report with different laboratory. The system will be able to detect different units and convert the units to the standard units and save to the database.

Overall, this system will reduce the work on manual data entry, minimizing the human error and improving the overall efficiency of medical data management. The project holds significant value in improving healthcare services, providing standard format, and advancing the medical technology.

1.6 Expected outcome

A prototype that can automatically extract medical data from physical documents using OCR will be developed as the key outcome of this project. By integrating advanced OCR techniques, the system will efficiently process blood test report, converting the document text into structured digital data. The accuracy of this prototype will be evaluated using metrics such as Character Error Rate (CER), Word Error Rate (WER) and Named Entity Recognition (NER) to ensure that the extracted information is high standard.

This system will serve as a proof of concept for automating blood test text extraction, reducing human error, and speeding up data entry tasks, ultimately improving the accuracy and efficiency of medical record management systems.

1.7 Project schedule

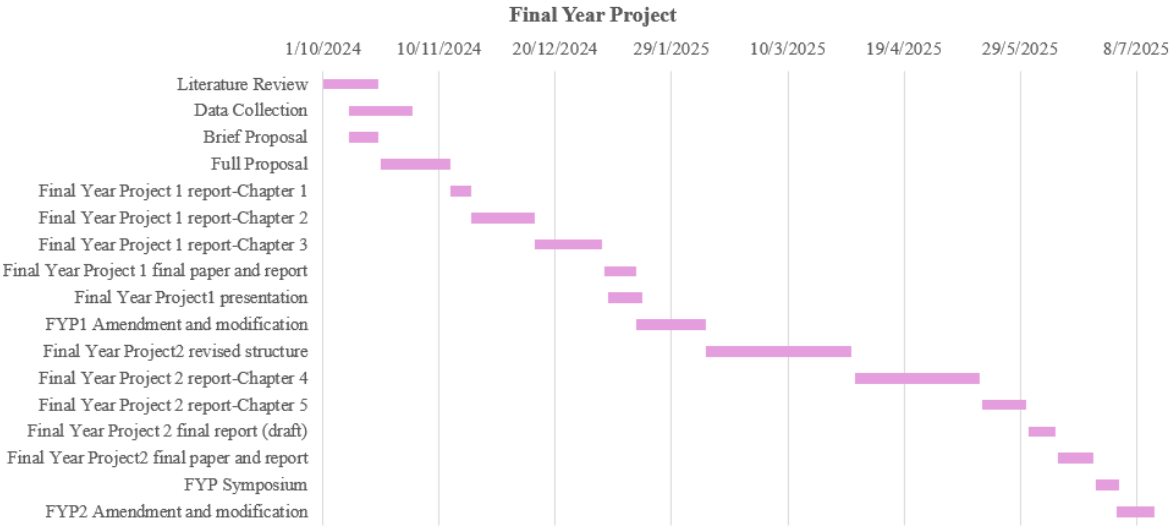


Figure 1. 4 Gantt Chart for final year project.

The Gantt chart illustrates the timeline for the Final Year Project, outlining from October 2024 to July 2025. The project begins with foundational activities such as literature review, data collection, and proposal writing, followed by the development of the Final Year Project report chapters. Subsequent stages include the final paper submission, presentation, and amendments. The second phase of the project (FYP2) involves further, report structuring, and the completion of the final draft. The process concludes with the FYP symposium and final amendments, ensuring the project is fully refined by July 2025.

1.8 Thesis outline

i. Chapter 1: Introduction

This chapter provides an overview of the report, starting with the background of the problem and the motivation behind the project. It includes a clear definition of the problem that needs

to be addressed, the objectives that are to be achieved, the limitations and scope of the project, and an outline of the report structure.

ii. Chapter 2: Literature Review

This chapter presents the terms and concepts related to the project, explaining them through a review of previous studies and published work from year 2019 to 2024. It also includes an analysis of current and existing systems that are look similar to this project, highlighting their approaches and how they relate to the development of the proposed system.

iii. Chapter 3: Methodology

This chapter present the details of the methodology used for creating the automated data extraction medical system. This explains the procedures involved in the development process, including the implementation of OCR and NLP, and outlines the steps taken for testing the usability and accuracy of the system.

iv. Chapter 4: Implementation

This chapter presents the implementation process of the system, the technologies and tools used.

v.Chapter 5: Testing

It also presents the results of system testing, which include the performance metrics and testing form.

vi.Chapter 6: Conclusion and future works

This concludes the report by summarizing the project. It reviews how the project met its objectives, discusses contributions to the field, and outlines possible future improvements or extensions to enhance the system’s capabilities further.

1.9 Summary

This chapter is the main introduction for the projects. It introduces the project background, defines the problem statement, states the objectives, details the methodology, project scope, significance of project, expected outcome, project timeline, and provides thesis outline of the report.

CHAPTER 2 LITERATURE REVIEW

2.1 Introduction

This chapter presents the previous study on the term and concept relevant to the development of this system. It explains several keywords that are important to this project. Section 2.1 is the introduction to Chapter 2 of this report. Section 2.2 explains the Review of Related Work which is the article review. Under Section 2.2, three small sections explain different techniques used. Section 2.2.1 presents for Optical Character Recognition (OCR), Section 2.2.2 presents for Natural Language Processing (NLP), and Section 2.2.3 will be presents Combination of OCR and NLP.

For Section 2.3, it contains the Review of existing scanner application. There are two small sections explain the application. Section 2.3.1 presents mobile application; Section 2.3.2 introduces web application. For Section 2.4, it is the discussion part. Section 2.4.1 introduces Comparison OCR and NLP Tools while Section 2.4.2 discuss about the comparison for the applications. Section 2.5 will be the summary of this section.

2.2 Review of Related Work

2.2.1 Optical Character Recognition (OCR)

Based on the research by Fleischhacker et al. (2024), OCR techniques are used to convert image to text. For this system, OCR techniques will be used to detect the blood test report and convert the data of the report to digital format. Optical Character Recognition (OCR) is a technology that can converts scanned documents into machine readable format text. OCR techniques are commonly used for data extraction. OCR is a powerful tool that transforms printed or handwritten text into digital formats, making it easier to manage and utilize information. The OCR process typically involves two main phases, which are Character Detection and Word Detection. The Character Detection phase focuses on identifying

individual characters within the image. For the Word Detection, it is following character detection, the system groups characters into words, which is essential for understanding the text's context. (Muthusundari et al., 2024) The four articles were selected based on their relevance, focus on OCR technology, and their diverse applications in various domains.

According to Pandey et al. (2023), this article was selected for its focus on Intelligent Document Management Systems (IDMS) and the integration of OCR with other advanced technologies like NLP and CV. According to the research conducted by Pandey et al. (2023), Intelligent Document Management System (IDMS) is a system which leverages advanced technologies such as Artificial Intelligence (AI), Machine Learning (ML), Natural Language Processing (NLP), and Optical Character Recognition (OCR). The OCR techniques will be important to extract data. This research will be focused on Easy OCR and Hybrid approach. Easy OCR is utilized as a technique for text recognition from images. Additionally, a Hybrid Approach, which combines Natural Language Processing (NLP) with Computer Vision (CV), were applied.

The research by Sugiyono et al. (2024) was chosen due to its application of OCR in Automatic Number Plate Recognition (ANPR). Sugiyono et al. (2024) focus on designing an Automatic Number Plate Recognition (ANPR) system using OCR techniques. The study employs the Tesseract library, an open-source OCR engine, to convert images of text into machine-readable text. This implementation is particularly effective for recognizing characters on vehicle registration plates. The research highlights the capability of Tesseract in recognizing characters on segmented license plates. The system's automatic recognition of characters underscores the OCR algorithm's precision and efficiency for this application.

The study by Ponnuru et al. (2024) was selected due to its direct application of OCR algorithms to prescription documents, which often feature varied handwriting styles and

printed text. The research by Ponnuru et al. (2024) utilizes OCR algorithms for text extraction from prescription documents. The study specifically employs Tesseract, renowned for its versatility in recognizing text in diverse fonts and formats. This makes Tesseract an optimal choice for processing the varying handwriting and printed text styles typically found in prescription forms.

According to the research by Chawla et al. (2020), the discussions in the paper lay a foundation for future research in the field of OCR and image processing. By exploring the capabilities of the Tesseract algorithm, the paper encourages further investigation into improving OCR technologies and their applications in various domains. This paper presented the using of the Tesseract algorithm that makes easier to extract text from images to help in detection of text from the captured image. The Tesseract algorithm is highlighted as a significant tool for extracting text from images. It is known for its effectiveness in recognizing and converting printed text into machine-readable formats.

2.2.2 Natural Language Processing (NLP)

According to Bhattacharjee et al. (2022), Natural Language Processing (NLP) is a branch of artificial intelligence (AI) that helps computers to understand, process, and analyse large amounts of natural human language data. Natural Language Processing (NLP) is a field within artificial intelligence (AI) that focuses on enabling computers to understand, process, and analyse human language data. It bridges the gap between human communication and machine understanding, allowing computers to interpret text and speech in a way that is meaningful and actionable. NLP techniques will be handling large volumes of unstructured language data, extracting insights, and automating various tasks, including text summarization, sentiment analysis, and language translation. (Supriyono et al., 2024) The four articles were selected based on their focus on utilizing Natural Language Processing (NLP) techniques to enhance various tasks within automated text extraction, summarization, and understanding.

According to article by Nair & Thusharab (2024), this article was discussed about it proposes an in-depth look at two widely used NLP libraries, SpaCy and NLTK. According to Nair & Thusharab (2024), SpaCy and Natural Language Toolkit (NLTK) are used in this research. SpaCy is a popular NLP library known for its speed and efficiency. It provides pre-trained models that can perform various NLP tasks, including part-of-speech tagging, named entity recognition, and dependency parsing. NLTK is another widely used library in the field of NLP. It offers a comprehensive suite of tools for text processing, including tokenization, stemming, and tagging. NLTK is particularly useful for educational purposes and research, as it provides a rich set of resources and documentation.

According to Prakash et al. (2022), this study was chosen because it explores how SpaCy, a NLP library, can be applied to automatic text summarization. Prakash et al. (2022) examined the use of SpaCy algorithm to analyze both extractive and abstractive summarization methods. It is known for its efficiency in natural language processing tasks, making it a suitable choice for the summarization process. The emphasis is on how SpaCy facilitates the summarization process, resulting in fewer iterations and a more focused summary.

According to Prakash et al. (2022), this article was selected as same as the previous study, as it focuses specifically on extractive text summarization, a key NLP technique for selecting the most important sentences or phrases from a text. In another survey by Prakash et al. (2022) indicates that SpaCy plays a crucial role in the growing field of automated text processing within Natural Language Processing (NLP). Its capabilities in scoring sentences and extracting relevant information are essential for creating meaningful summaries from extensive textual content.

Based on Jugran et al. (2024), this article was focus on extractive summarization using SpaCy and its comparison with manual summarization. Based on Jugran et al. (2024), this

paper contrasts manual summarization with automated methods, asserting that automatic text summarization is not only essential but also more accurate and efficient. It highlights the cost-effectiveness and time-saving benefits of using automated systems to summarize large volumes of text. The paper emphasizes that NLP techniques enable machines to analyze and interpret large volumes of text data. This paper used the NLP technologies, SpaCy, which provides tools for various NLP tasks such as tokenization, part-of-speech tagging, and named entity recognition.

2.2.3 Combination of OCR and NLP

Based on the research by Zhou et al. (2023), Optical Character Recognition (OCR) and Natural Language Processing (NLP) can be used for automated data extraction process. The four articles were selected based on their focus on combining Optical Character Recognition (OCR) with Natural Language Processing (NLP) techniques, which is key to automating data extraction from documents.

According to Malashin et al. (2024), this research was selected for its focus on improving the handling of unstructured data in the medical field, specifically medical reports. According to Malashin et al. (2024), the research is focused on image text extraction and natural language processing (NLP) from medical reports has several key purposes aimed at improving the handling of unstructured data in the medical field. The study highlights the use of a genetic algorithm (GA) for OCR techniques while Named Entity Recognition (NER) for NLP techniques.

According to Oehm et al. (2024), this research was chosen for its comparison of three approaches for extracting medication data from semi-structured prescriptions. According to Oehm et al. (2024), this research is to compare three approaches to automatically extract medication data from semi-structured German prescriptions, which is Levenshtein-based

Approach, Rule-based Approach, and CRF (Conditional Random Field)-based Approach. The paper aims to compare different approaches for extracting medication data from these semi-structured prescriptions.

According to Kumar and Prasad (2024), this article was selected for its novel approach to plagiarism detection using OCR and NLP. According to Kumar and Prasad (2024), this research is to present a novel approach to detecting plagiarism in text by scanning the images. The research designed to be efficient and automated, making it applicable in multiple environments. The system developed in this research is to compare the extracted text against a comprehensive database of existing sources to avoid plagiarism. The research approach to OCR techniques and NLP techniques used. The Optical Character Recognition (OCR) is to extract text from images. The Natural Language Processing (NLP) techniques is to evaluate the originality of the extracted content. The OCR technologies used is Tesseract OCR, Google Cloud Vision API, Cloud-based OCR, Adobe OCR, ABBYY FineReader. The NLP Technologies are spaCy, NLTK (Natural Language Toolkit), Gensim, BERT (Bidirectional Encoder Representations from Transformers.

According to Hegghammer (2021), this paper was selected for its detailed benchmarking of multiple OCR technologies (Tesseract, Amazon Textract, Google Document AI) and their application to historical documents. Based on Hegghammer (2021), this paper focuses on Optical Character Recognition (OCR), a technology that helps convert images of text into machine-readable text. This is particularly useful for analyzing historical documents that have not been studied. This paper focuses the potential of Optical Character Recognition (OCR) technology to transform historical documents into formats suitable for computational analysis. The OCR techniques used is Tesseract, Amazon Textract, Google Document AI. OCR tools convert images to text The NLP techniques used is SpaCy. NLP techniques can be applied to analyze the text

2.3 Review of existing scanner application

2.3.1 Mobile Application

Microsoft Office Lens

Microsoft Office Lens is a mobile scanning app to digitize documents, notes, whiteboards, and images. It enhances the scanned images and converts them into editable text by using OCR. It allows users to save them as PDFs, Word, or PowerPoint files. The app is widely used for organizing notes and managing documents efficiently. The app uses the device's camera to capture documents or notes. It can adjust brightness, contrast, and alignment for better reading. Microsoft Office Lens is free to use but it will have charges with 5 GB of free storage and integrates with Microsoft 365 services (Microsoft Corporation, 2024).

Logo:

Logo Source: (Microsoft Corporation, 2024)

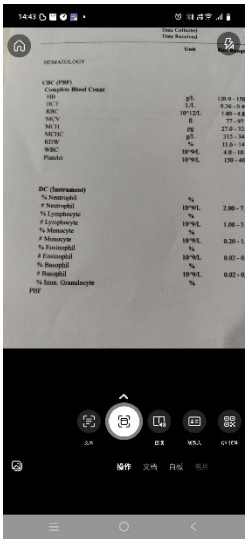


Figure 2. 1 Screenshot of Microsoft Office Lens

Figure 2. 2 Screenshot of Microsoft Office Lens

Figure 2.1 and 2.2 shows the screenshot for Microsoft Office Lens tools. Figure 2.1 is the capture function and figure 2.2 is the text extracted.

NaturalReader Text to Speech

NaturalReader is a text-to-speech app that converts written text into natural-sounding speech. It supports a variety of formats, including PDFs, eBooks, Word documents, and web pages. The app is ideal for users with reading difficulties, professionals looking for text-to-audio solutions, or anyone who prefers listening over reading. It converts written text into audio using advanced TTS algorithms. It is customizable voices and playback speeds. It is supporting multiple languages. The app is free and charges for premium versions with additional features. (Naturalsoft Ltd., 2024).

Logo:

Logo Source: (Naturalsoft Ltd., 2024)

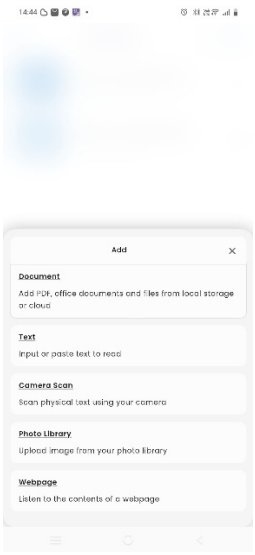


Figure 2. 3 Screenshot of NaturalReader

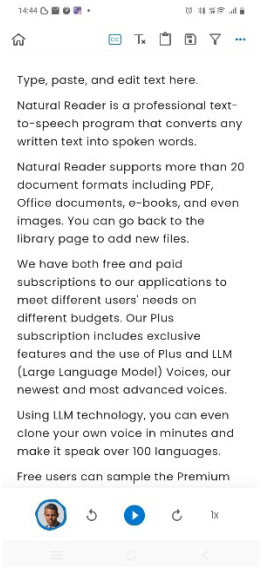


Figure 2. 4 Screenshot of NaturalReader

Figure 2.3 and 2.4 shows the screenshot for NaturalReader tools. Figure 2.3 is the upload function and figure 2.2 is the text to speech.

GradeUp: Homework Scanner

This app is designed to help students scan and manage their homework and notes. It can scan handwritten or printed materials. The Homework Scanner app removes unnecessary elements like scribbles or irrelevant notes, making the scanned content clean and ready for submission or revision. It supports basic OCR functionalities to extract text from handwritten or printed materials. It is free but with optional in-app purchases for advanced features. (Pixelcell Pte Ltd., 2024).

Logo:

Logo Source: (Pixelcell Pte Ltd., 2024)

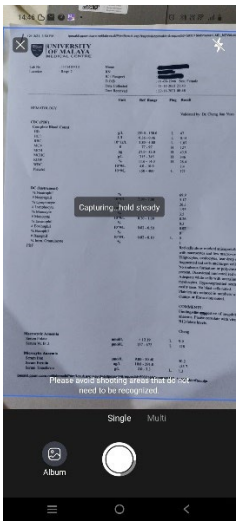


Figure 2. 5 Screenshot of GradeUp: Homework Scanner

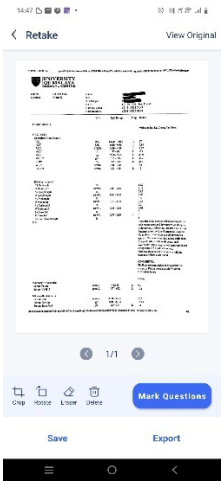


Figure 2. 6 Screenshot of GradeUp: Homework Scanner

Figure 2.5 and 2.6 shows the screenshot of GradeUp: Homework Scanner tools. Figure 2.5 is the capture function and figure 2.6 is the text recognized page.

Google Lens

Google Lens is a visual search tool that uses artificial intelligence to extract and interpret information from images. Users can scan text, identify objects, search for products, translate languages, and even solve equations. It integrates seamlessly with Google services, offering a versatile utility for daily tasks. It uses AI to analyse images captured by the device's camera. It is free with integration into Android devices and Google services (Google LLC, 2024).

Logo:

Logo Source: (Google LLC, 2024)

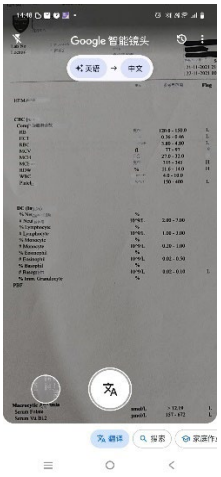


Figure 2. 7 Screenshot of Google Lens

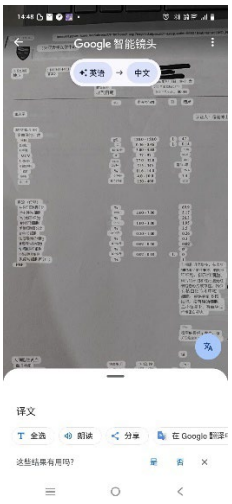


Figure 2. 8 Screenshot of Google Lens

Figure 2.7 and 2.8 shows the screenshot of Google Lens tools. Figure 2.7 is the capture function and figure 2.8 is the text translate to mandarin page.

Text Fairy (OCR Text Scanner)

Text Fairy is a lightweight OCR app that converts images of printed text into editable and searchable content. It captures printed text using the device's camera. It supports multiple languages and allows users to create PDFs from scanned documents. The app is particularly

useful for digitizing books, notes, and other printed materials efficiently. It is free to use (Renard Wellnitz, 2023).

Logo:

Logo Source: (Renard Wellnitz, 2023)

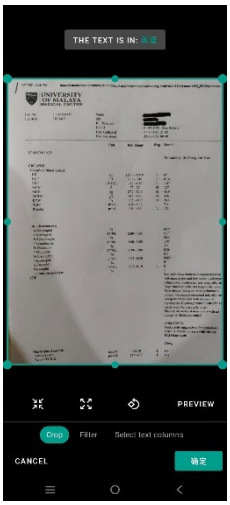


Figure 2. 9 Screenshot of Text Fairy

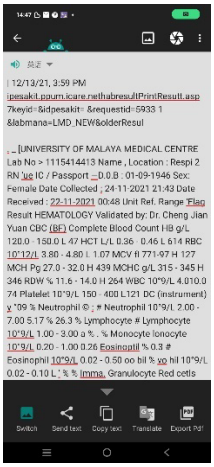


Figure 2. 10 Screenshot of Text Fairy

Figure 2.9 and 2.10 shows the screenshot of Google Lens tools. Figure 2.9 is the capture function and figure 2.10 is the text extracted page.

2.3.2 Web Application

Klippa

Klippa is using OCR and advanced technology to offer a web-based platform for data extraction and document management. By using OCR and advanced AI technologies, Klippa will extracts text and data from scanned or uploaded documents. It is automated the digitization of invoices, receipts, contracts, and other documents into structured digital data. It supports multiple languages and integrates seamlessly with business tools like accounting software systems. For NLP technologies, it extracts specific data fields like names, dates, and amounts from documents by using NLP techniques to recognizes context to identify structured and unstructured data. It is free and premium version available (Klippa, 2025).

Logo:

Logo Source: (Klippa, 2025)

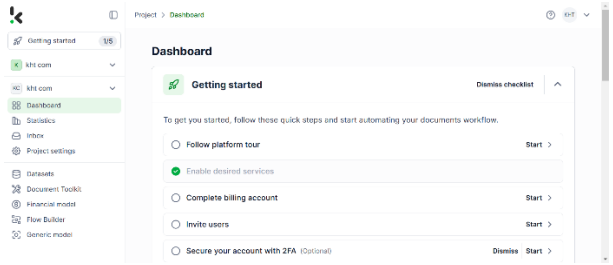


Figure 2. 11 Screenshot of klippa

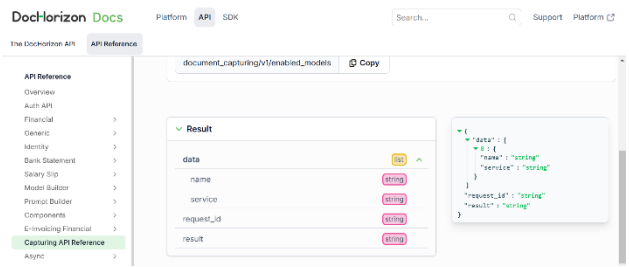


Figure 2. 12 Screenshot of klippa

Figure 2.11 and 2.12 shows the screenshot of klippa tool. Figure 2.11 is the dashboard for klippa and Figure 2.12 is the API page.

Rossum.ai

Rossum is a data extraction platform designed for processing invoices and other business documents. It uses OCR technologies to extracts data from documents without templates. It leverages NLP and machine learning, intelligently extracts relevant data fields, even from unstructured layouts. Rossum also allow validation workflows to improve data accuracy. Rossum uses NLP to process unstructured data and identify the meaningful fields within documents. It uses algorithms to classify data, improving extraction accuracy for invoices and forms. It is free and premium version available (Rossum, 2025).

Logo: ROSSUM

Logo Source: (Rossum, 2025)

Figure 2. 13 Screenshot of Rossum.ai

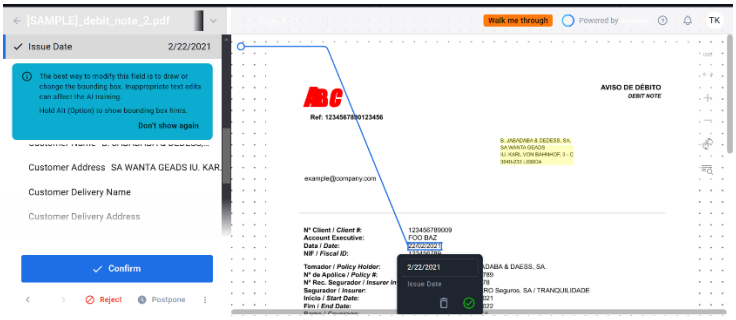


Figure 2. 14 Screenshot of Rossum.ai

Figure 2.13 and 2.14 shows the screenshot of Rossum.ai tool. Figure 2.13 is the homepage and Figure 2.14 is the invoice generated page.

Docsumo

Docsumo is special in automating data extraction from semi-structured documents like invoices, receipts, and bank statements. It uses OCR technologies to extract text and numeric data from uploaded documents. By using advanced OCR and AI techniques, it identifies key data fields and integrates the extracted data into workflows like accounting systems. The NLP technologies used to analyse text and numbers to identify key-value pairs. It is free and premium version available (Docsumo, 2025).

Logo: DOCSUMO Document AI

Logo Source: (Docsumo, 2025)

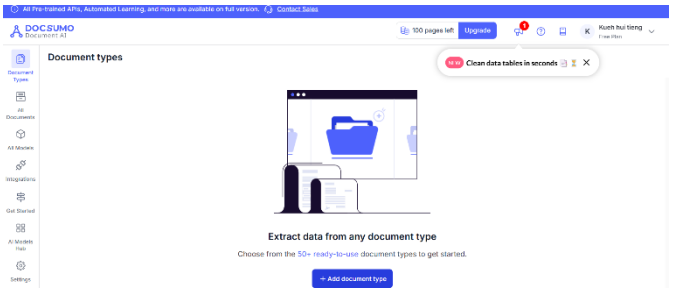


Figure 2. 15 Screenshot of Docsumo

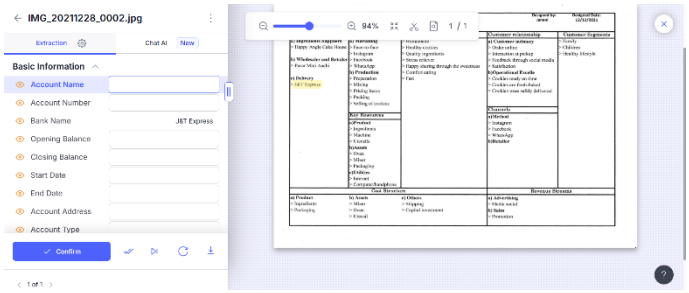


Figure 2. 16 Screenshot of Docsumo

Figure 2.15 and 2.16 shows the screenshot of Docsumo tool. Figure 2.15 is the upload page and Figure 2.16 is the text extracted page.

Nanonets

Nanonets is a web platform for OCR and data extraction. It uses AI to automate workflows, allowing users to train models on custom datasets. The platform supports a huge range of document types, including invoices, receipts, and contracts. NLP techniques use in Nanonets will be allows users to train custom models, leveraging NLP for language processing and recognize context sensitive data. It can handle multilingual data extortion and text analyse. It is free and premium version available (Nanonets, 2025).

Logo: Nanonets

Logo Source: (Nanonets, 2025)

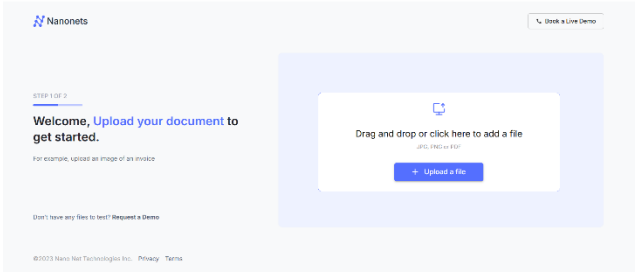


Figure 2. 17 Screenshot of Nanonets

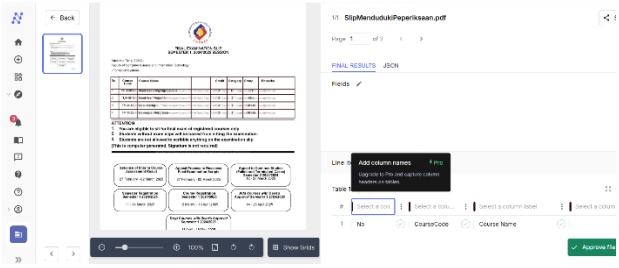


Figure 2. 18 Screenshot of Nanonets

Figure 2.17 and 2.18 shows the screenshot of Nanonets tool. Figure 2.17 is the upload page and Figure 2.18 is the text extracted page.

ABBYY FineReader

ABBYY FineReader is a OCR and document conversion tool that combines AI-powered text recognition with PDF editing capabilities. It is converting scanned documents into editable formats and extracting the data for further use. The OCR techniques used to convert scanned images and PDFs into editable formats. ABBYY FineReader utilizes AI and NLP to analyse and understand document layouts and text patterns. The NLP contextual the text extraction to improve the quality of output in structured formats like tables. It is free and premium version available (ABBYY, 2025).

Logo: ABBYY

Logo Source: (ABBYY, 2025)

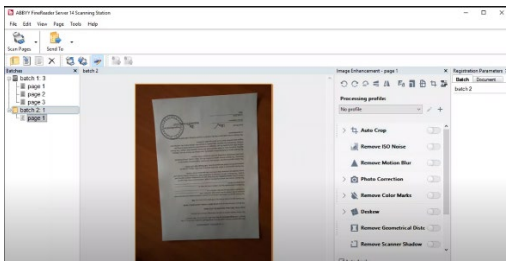


Figure 2. 19 Screenshot of ABBYY FineReader.

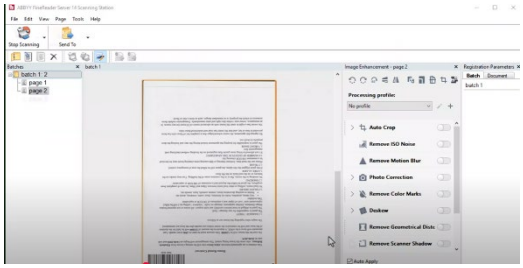


Figure 2. 20 Screenshot of ABBYY FineReader.

Figure 2.19 and 2.20 shows the screenshot of ABBYY FineReader tool. Figure 2.19 is the upload page and Figure 2.20 is the text extracted page.

2.4 Discussion

2.4.1 Comparison OCR and NLP Tools

Feature/ Tool	Tesseract	Easy OCR	Google Vision API	Amazon Textract	Google Document AI	BERT	Gensim	SpaCy	NLTK	Stanford NLP
Source	Chawla et al. (2020), Kumar and Prasad (2024), Ponnuru et al. (2024), Sugiyono et al. (2024),	Pandey et al. (2023)	Kumar and Prasad (2024)	Heggammer (2021)	Heggammer (2021)	Kumar and Prasad (2024)	Kumar and Prasad (2024)	Heggammer (2021), Jugran et al. (2024), Kumar and Prasad (2024), Nair & Thusharab (2024) Prakash et al. (2022)	Nair & Thusharab (2024)	Malashin et al. (2024)
Tools Type	OCR	OCR	OCR	OCR	OCR	NLP	NLP	NLP	NLP	NLP
OCR Accuracy	High	High	Very high	Very high	Very high	No	No	No	No	No
Multilanguage Support	100+	80+	50+	50+	50+	100+	100+	100+	100+	100+
Text Extraction	Yes	Yes	Yes	Yes	Yes	No	No	No	No	No

Convert image to text	Yes	Yes	Yes	Yes	Yes	No	No	No	No	No
Recognize structured text	Yes	Yes	Yes	Yes	Yes	No	No	Yes	Yes	Yes
NLP Performance	No	No	No	No	No	Very high	High	Very high	High	High
Tokenization	No	No	No	No	No	Yes	Yes	Yes	Yes	Yes
Text classification	No	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Ease of Use	Easy	Easy	Easy	Easy	Easy	Moderate	Easy	Moderate	Moderate	Moderate
Cost	Free (Open Source)	Free (Open Source)	Free and Paid (Premium)	Free and Paid (Premium)	Free and Paid (Premium)	Free (Open Source)	Free (Open Source)	Free (Open Source)	Free (Open Source)	Free (Open Source)
Customizability (Trainability)	High	High	Low	Low	Low	High	Moderate	High	High	High
Integration	Easy (via libraries)	Easy (via libraries)	Easy (via API)	Easy (via API)	Easy (via API)	Moderate (setup)	Easy (via API)	Easy (via API)	Moderate (setup)	Easy (via API)
Document Layout Handling	Moderate	Moderate	High (Optimized for forms)	High (Optimized for forms)	High (Optimized for forms)	No	No	Moderate	Moderate	High
Pre-Trained Models (NER)	N/A	N/A	N/A	N/A	N/A	Yes	Yes	Yes	Yes	Yes
Text Processing Capabilities	Moderate	Moderate	Moderate	Moderate	Moderate	Very High	Moderate	Very High	Very High	High
Speed	High	High	High	High	High	High	High	High	Moderate	High

Table 2. 1 Comparison for tools.

Table 2.1 comparison for 10 tools for OCR and NLP. Selecting five for OCR techniques and five for NLP based on the articles. These tools were selected for their ability to convert

images or scanned documents into machine-readable text (OCR tools) and to analyze and process text to extract useful information (NLP tools). OCR tools are particularly useful when working with documents like blood test reports, where text extraction is required from scanned images. NLP tools are excellent for extracting important entities, such as patient names, test results, and dates, from that extracted text.

Based on the table 2.1, for OCR, Tesseract is an open-source OCR engine that has gained significant recognition for its text extraction capabilities. It supports over 100 languages, making it a good tool for documents in various languages. It is easy to use. One of its major advantages is its customizability, especially when working with unstructured or varied data. Tesseract can be particularly effective when processing documents such as blood test reports that may contain multi-language content, handwritten notes, or varying layouts. However, it does require some technical knowledge to integrate, to make it suitable for flexibility and control.

Another OCR tool, EasyOCR, is also open-source and supports more than 80 languages. It is easy to use and provides competitive OCR accuracy, making it a strong choice for text extraction tasks. EasyOCR's capabilities make it suitable for converting scanned images of documents into text with different text types and formats. Like Tesseract, it's open-source and offers significant customization options.

Google Vision API, a cloud-based service, offers high OCR accuracy with advanced features, include text extraction from images and document layout analysis. It is especially effective for structured documents. This tool able to handle complex documents with high precision makes it ideal for cases where the document that has formatting. However, while the API is easy to use, it may cost for frequent use due to its cloud-based nature.

Similarly, Amazon Textract is another cloud-based OCR service that specializes in extracting text from documents with complex structures. Textract is useful for structured data extraction, making it an good choice for medical reports or forms like blood test reports, where there are multiple fields and layout variations.

Google Document AI is a tool for extracting, analyzing, and processing text from documents. It handles complex document layouts effectively. Google Document AI excels in parsing structured data and is well-suited for use cases that involve medical test reports or other forms requiring intelligent document classification. It offers APIs for seamless integration but, like other cloud-based tools, comes with a pricing model.

For NLP, BERT is a powerful model that help on understanding the context of words in sentences. It uses a bidirectional approach to analyse the relationships between words and is high effective for tasks like Named Entity Recognition (NER), sentiment analysis, and text classification. BERT can be very useful for extracting specific entities from text. BERT offers highly accurate text analysis, making it a good choice for NLP tool.

Gensim, another NLP tool, specializes for document similarity, and word embedding techniques. It is particularly effective for unsupervised learning tasks. For analyzing large volumes of text, Gensim can uncover patterns and relationships that are not immediately apparent, providing valuable insights into the underlying data. It is relatively easy to use and integrates well into Python-based systems for NLP applications.

SpaCy is a high-performance NLP library known for its speed and accuracy. It is especially good for tasks like tokenization, part-of-speech tagging, and NER. SpaCy's efficiency in extracting structured entities from text makes it ideal for real-time applications, such as processing medical records where specific entities like test results, dates, and patient information need to be identified quickly. It is also highly customizable and provides tools for

fine models for domain-specific tasks, making it an excellent choice for working on specialized NLP tasks.

The Natural Language Toolkit (NLTK) is one of the most widely used libraries for NLP in Python. It offers a broad range of tools for tasks like tokenization, classification, parsing, and stemming. While NLTK is highly versatile and provides great functionality for research and educational purposes, it may not be as fast as other libraries, such as SpaCy, when handling large datasets. However, for small text datasets, NLTK is a valuable tool for text analysis.

Stanford NLP provides a pre-trained models that can be used for a variety of NLP tasks, including part-of-speech tagging, dependency parsing, and Named Entity Recognition. It is high accuracy and understanding the relationships between words in sentences. However, Stanford NLP implementation may make it more difficult for Python users to integrate, and it requires more configuration than some of the other tools.

In conclusion, Tesseract and EasyOCR are both open-source OCR tools, which means they are free to use, but Tesseract being highly customizable. Tesseract provides solid performance for text extraction from images and supports multiple languages, making it highly customizable and cost-effective. So, it will be better for unstructured data, like blood test report. Tesseract has text extraction function, convert image to text function, and recognize structured text function, which are all the function needed for the systems. Additionally, Tesseract supports multiple languages above 100 languages, which will be important in case the blood test reports are in different languages. Tesseract is the easiest for integration into the system via libraries. SpaCy is a high-performance NLP library which offers very high text processing capabilities, like NER (Named Entity Recognition). SpaCy can recognize structured text, have tokenization techniques, and text classification techniques. SpaCy offer high speed, which means it can handle large volumes of data quickly. SpaCy provides high customizability. SpaCy

integrates easily with other tools and libraries. Thus, Tesseract and SpaCy is the best choice for OCR and NLP tools.

2.4.2 Comparison for application

App Name	Microsoft Lens	Natural Reader	Grade Up: Homework Scanner	Google Lens	Text Fairy	Klippa	Rossum.ai	Docsumo	Nanonets	ABBYY FineReader
Source	(Microsoft Corporation, 2024)	(Naturalsoft Ltd., 2024)	(Pixelcell Pte Ltd., 2024)	(Google LLC, 2024)	(Renard Wellnitz, 2023)	(Klippa, 2025)	(Rossum, 2025)	(Docsumo, 2025)	(Nanonets, 2025)	(ABBYY, 2025)
OCR Library	Microsoft (Azure Service)	Tesseract	Tesseract	Google Cloud Vision	Tesseract	Tesseract, Klippa OCR	Tesseract	Tesseract	Tesseract	ABBYY FineReader OCR
Text Extract from images	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Text Recognition	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
NLP Library	Microsoft Word's built-in NLP features	Open-source NLP libraries	spaCy, NLTK	Google NLP tools, Google's BERT	Open-source NLP libraries	spaCy, NLTK	spaCy, NLTK	spaCy, NLTK	spaCy, NLTK	ABBYY Text Analytics
Spelling, grammar checking	Yes	No	No	No	No	Yes	No	No	No	Yes
Text-to-Speech (TTS)	No	Yes	Yes	Yes	Yes	No	No	Yes	No	No
Extract meaning	No	No	No	Yes	No	Yes	Yes	Yes	Yes	Yes
Answer question & explanation	No	No	Yes	No	No	No	Yes	Yes	No	No
Main Features	OCR, text extraction, docu	OCR, text-to-speech, read docum	OCR, homework assistance,	OCR, object recognition, translati	OCR text extraction, text	OCR, invoice process, receipt scannin	OCR, document data extrac	OCR, document data extract	OCR, document data	OCR, text extract, document

	ment editing.	ent, customize voices.	question answering	on, search.	edit, multilangauge.	g, text extract	t, AI-driven	, invoice processing	extract	convert.
Charges	Free of charge	Free and premium version	Free and premium version	Free of charge	Free with ads, premium version	Free trial, subscribe for premium version	Free trial, subscribe for premium version	Free trial, subscribe for premium version	Free trial, subscribe for premium version	Free trial, subscribe for premium version

Table 2. 2 Comparison for mobile and web applications

Based on table 2.2, the 10 applications have been chosen. Five applications for mobile based, five applications for web based. By limiting the selection to five applications each based allows for a more focused and manageable comparison. The primary focus for selection was applications that provide effective OCR functionalities to convert scanned images into machine-readable text. Additionally, these applications should feature relevant NLP tools to enhance the user experience. Each selected application utilizes a distinct OCR engine (e.g., Microsoft OCR, Tesseract OCR, Google Cloud Vision OCR, Klippa OCR, ABBYY FineReader OCR), and incorporates NLP tools in unique ways. The rating, which rating above 4/5 is one of the reasons to choose as comparison for mobile applications. The selected mobile applications have ratings ranging from 4.2 to 4.9. Based on the comparison Microsoft Lens is best for the users who need professional document scanning and integration with Microsoft Office. NaturalReader is best for users who need OCR combined with text-to-speech (TTS) capabilities for reading documents aloud. Homework Helper best for students who need OCR to solve homework problems and explanations. Google Lens is best for users who need real-time object/landmark recognition and text extraction from images. Text Fairy is best for users who need a simple OCR tool for extracting text from images. By selecting Klippa, Rossum.ai, Docsumo, Nanonets, and ABBYY FineReader for comparison, I am focusing on platforms that provide highly accurate OCR, intelligent data extraction tools for comparing. Klippa is ideal for businesses that need invoice processing and receipt scanning. The tool provides OCR

functionality combined with data extraction. Rossum.ai offers AI-driven document extraction that goes beyond traditional OCR. Docsumo excels at invoice processing and form data extraction. Nanonets provides easy-to-use API integrations and supports multi-language OCR for global operations. ABBYY FineReader is for text extraction, document conversion, and PDF management.

2.5 Summary

Based on the study, Tesseract is more used for OCR techniques, and SpaCy is the mostly used for NLP techniques. Based on the comparison of applications, the Tesseract is more in used and open-source NLP tools will be a lot of used. The techniques that choose to use for the system for OCR is Tesseract tools. Tesseract is free to use, making it accessible for developers and researchers. It can recognize text in various languages, it will be recognized for the mandarin in blood test report. The techniques used for NLP in systems is SpaCy libraries. SpaCy is highly optimized for speed, making it suitable for processing large datasets quickly. SpaCy offers pre-trained models for various NLP tasks, such as named entity recognition, part-of-speech tagging, and dependency parsing. For accuracy measurement, Character Error Rate (CER), Word Error Rate (WER) and Named Entity Recognition (NER) will be used.

CHAPTER 3 METHODOLOGY

3.1 Introduction

In this chapter, the methodology to be applied on this project will be discuss. The methodology used is the Agile methodology to rapidly prototype and implement for automating the extraction of medical data from blood test reports. The steps to be applied for Agile methodology is introduces with each task to be done for each step in this project.

Under Chapter 3, there are several sections. The first section (Section 3.1) is the introduction to this chapter. Section 3.2 shows the systems architecture. Section 3.3 further explains Agile methodology to be applied in this project. In Section 3.3, the three sub-sections will be introduced. The subsections will be each phase of Agile methodology. Section 3.3.1 will be Requirements Section 3.3.2 will be Plan and Design. Section 3.3.3 proposes Planning for Implementation & Testing. Section 3.4 is the Summary to conclude this chapter.

3.2 Architecture

The system architecture is designed to handle the end-to-end process of extracting, structuring, and storing medical data from scanned blood test reports. The system flow can be represented by the following diagram:

System Flowchart

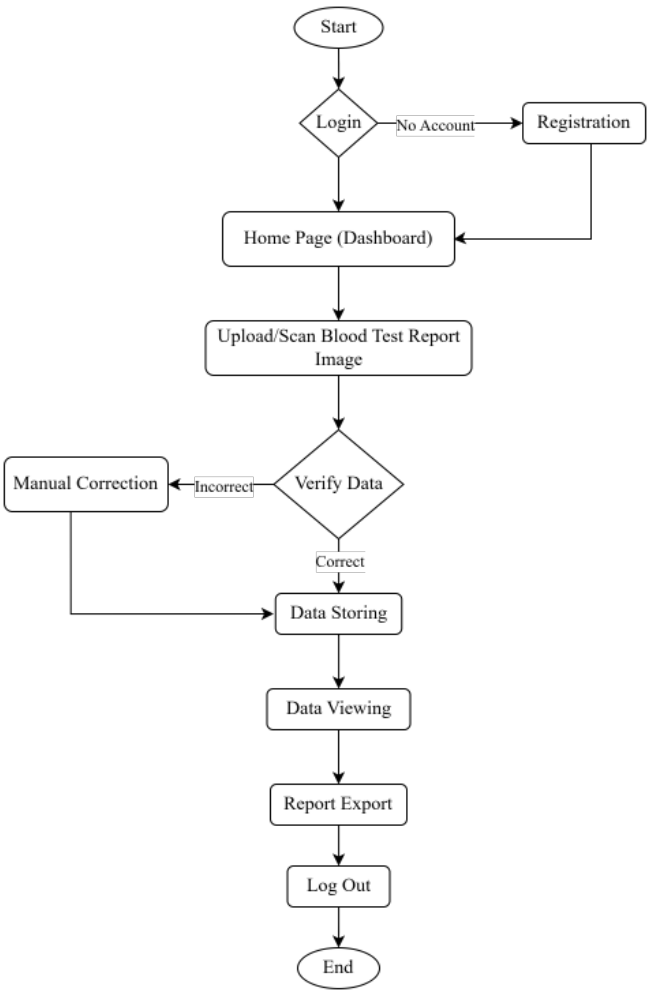


Figure 3. 1 Flowchart of System

The system includes for 12 stages. The 12 stages will be:

- **Start**

  This is the initiation of the process for the system. The user will begins interacting with the system to manage blood test reports.

- **Login**

  The user is prompt to log in page for the system. The user will log in their account using their credentials. If the user has no account, they are redirected to the Registration process to create an account. This step is to ensure secure access to the platform.

- **Registration**

For users without an account, this step allows them to register by providing the required information. Once the registration is complete, they can proceed to log in.

- **Home Page (Dashboard)**

  After logging in, the user will be directed to the dashboard. The main functionalities of the system are accessible, including the option to upload or scan blood test report images.

- **Upload/Scan Blood Test Report Image**

  The user can upload the blood test report image from their device or scan the report using the system's scanning feature. This image is then processed for data extraction.

- **Verify Data**

  The system processes the image and extracts the data. The user can verify the accuracy of the extracted data. If the data incorrect Data, the user is redirected to the Manual Correction step to edit and fix errors. If the data is correct, the user confirms and moves to the next stage.

- **Manual Correction**

  If the extracted data contains errors, the user manually corrects the information. Once corrected, the process loops back to the Verify Data step.

- **Data Storing**

  After verified, the data is saved in the system's database for future reference.

- **Data Viewing**

  The user can view the stored data in a structured and organized format. This step helps with report analysis and validation.

- **Report Export**

  The system allows users to export the stored reports in various formats for offline use.

- **Log Out**

The user logs out of the system to end the session securely, ensuring data protection.

- **End**

  The process concludes, and the user exits the system.

3.3 Agile Methodology

Agile methodology is a user-centric, iterative approach that emphasizes flexibility, collaboration, and frequent delivery of functional components. This approach is particularly suitable for the Automated Blood Test Data Extraction System of medical record management that includes OCR and NLP technologies for extracting medical data. In this methodology, a questionnaire feature will also be integrated to allow users to collect requirements regarding the system's functionality and effectiveness. Agile methodology has is one of the most widely adopted strategies in software development. Unlike traditional software development models, Agile emphasizes flexibility and iterative progress throughout the development lifecycle. Agile methodology will enhance customer satisfaction and accelerate product delivery by leveraging an incremental and iterative approach. (Omonije, A., 2024)

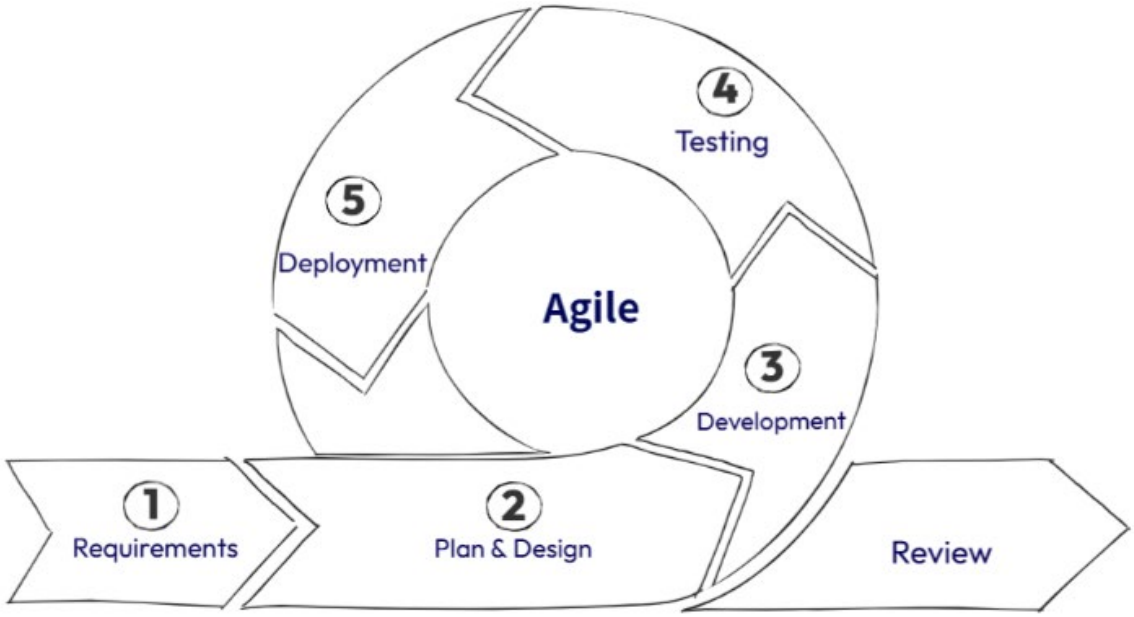


Figure 3. 2 Agile Methodology process

Figure 3.2 will be the Agile methodology process. Agile methodology consists of six key phases: Requirements, Plan and Design, Development, Testing, Deployment, and Review. The Requirements stage is the initial stage where requirements are gathered and analysed. It is to understand what the project aims to achieve and document the needs of stakeholders. For Plan and Design, the system architecture and design are created based on the requirements. Development is the stage involves actual software development, where coding is performed according to the design specifications. The system is built. For Testing stage, after development, the software undergoes testing to identify and fix any bugs or issues. This phase ensures that the system meets the requirements and improve the system. For Deployment, once the software passes testing, it is deployed to a production environment. Review stage is the final phase, the project is evaluated. User review the product to ensure it meets the requirements and objectives. Feedback is gathered to identify any improvements or changes needed for future iterations.

3.3.1 Requirements

Analyse for user requirements, system requirements, and understand the system's purpose and gather specific requirements.

User Requirements

Gather essential requirements for both the OCR and NLP based system and the questionnaire feature (Appendix A). For this user requirements, I have conducted Google Form to collect the requirements. The question will be asked to the medical staff. The result of the survey that has been conducted is shows in figures 3.1 until 3.20 below. The main reason of this survey is to ask the respondents for their views on this system and the features available, including by asking for their suggestions. This response is by the doctor that require for the digital system to decrease the time wasting for manually entry the blood test data.

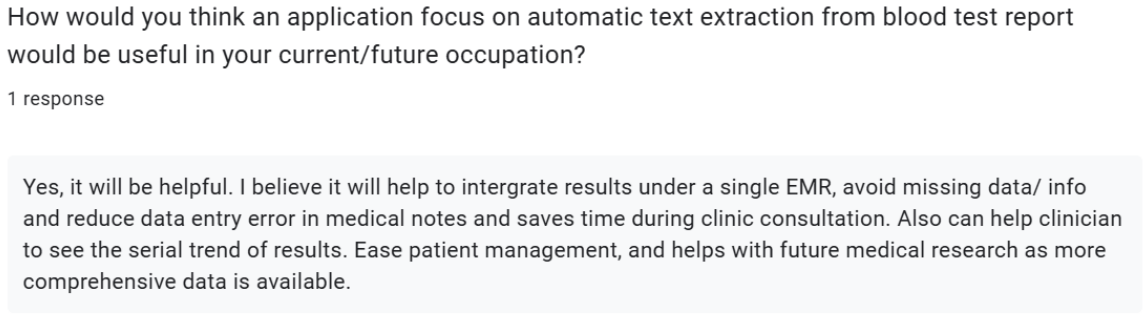


Figure 3. 3 User Requirement 1.

From figure 3.3, this question is to ask for the reason and helpfulness for the future work. Regarding the response by the doctor, it is very helpful and decrease the doctor's work and help to ease the patient management. It is also helpful for future searching patient's data.

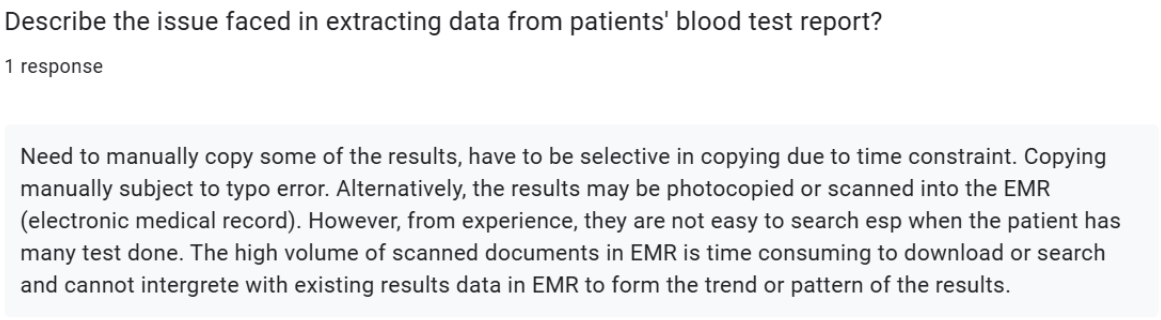


Figure 3. 4 User Requirement 2.

According to figure 3.4, this question is to describe for the doctor issue faced, and it will be also the problem statement that needs to be solved. Based on the doctor's response, medical staff currently manual key in the medical data from physical documents such as blood test reports into electronic systems. This process is not only time wasting but also lead to human error. The search for the patient will be also difficult as the report are all in hardcopy.

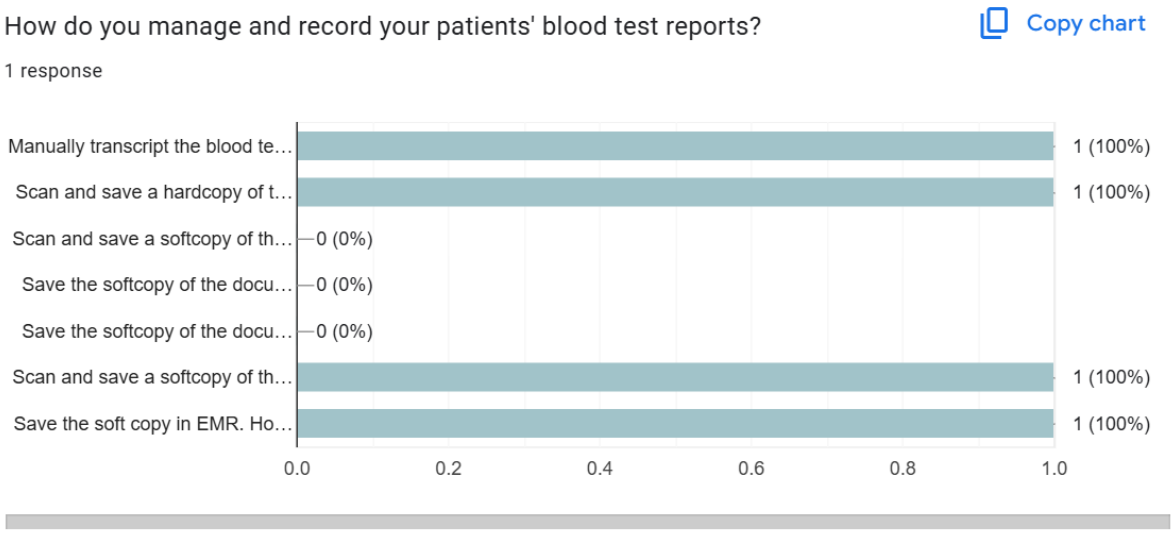


Figure 3. 5 User Requirement 3.

Based on the figure 3.5, the question asks for manage and record for blood test report. The doctor requires manually transcript blood test data on paper, scan and save the hardcopy, scan and save softcopy of document, save the softcopy in EMR.

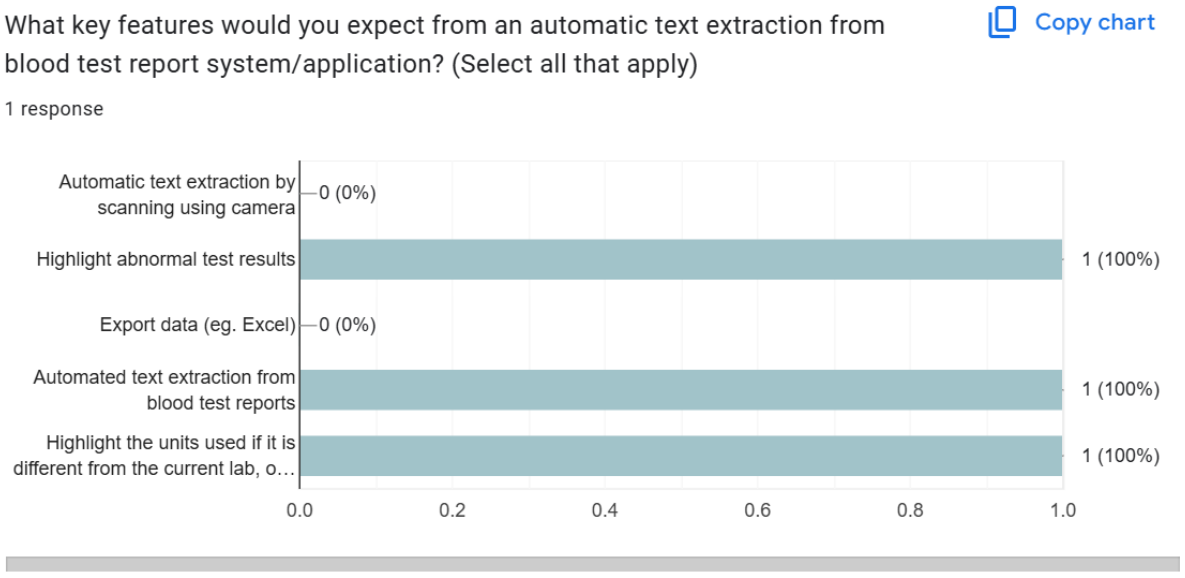


Figure 3. 6 User Requirement 4.

Based on figure 3.6, the question is to ask for key features for the system. From the response by doctor, the system would need to have highlight abnormal results, automated text extraction, highlight for units if different unit extracted features.

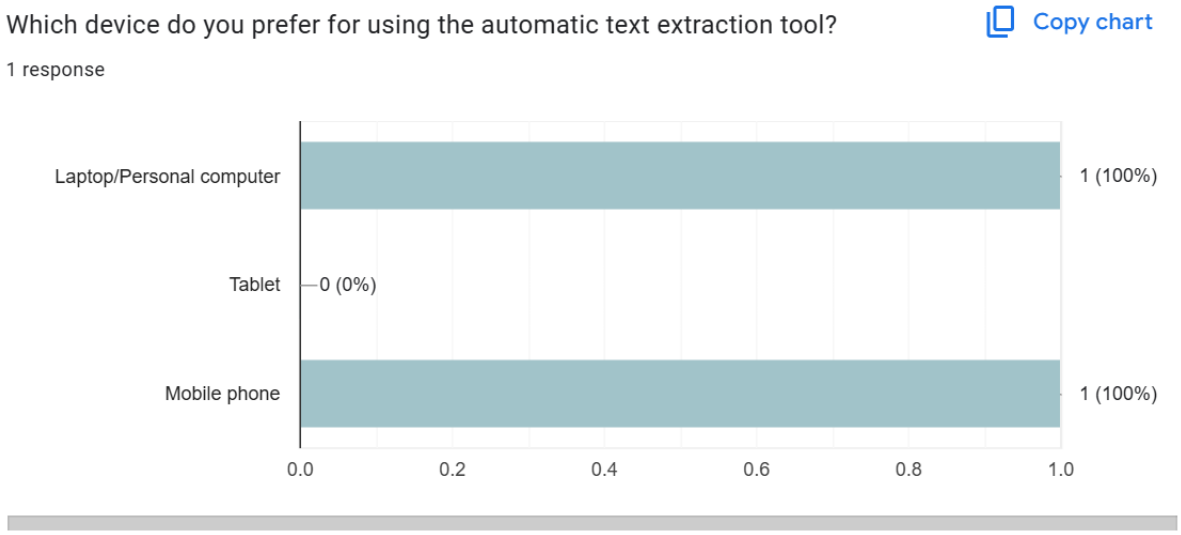


Figure 3. 7 User Requirement 5.

From figure 3.7, the prefer device for automatic text extraction system will be ask, the prefer device will be laptop/personal computer, mobile phone.

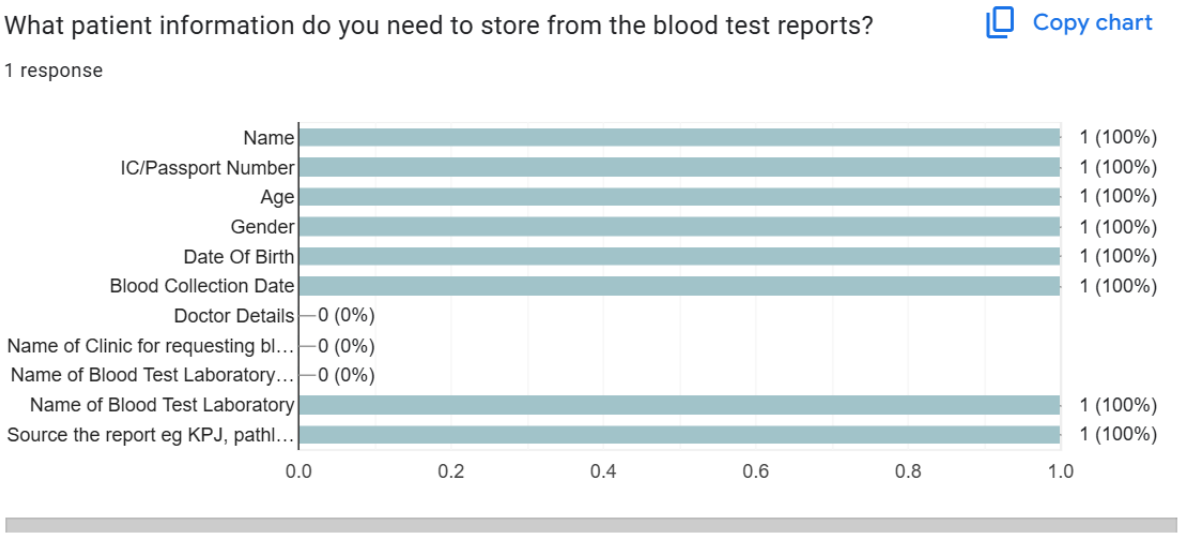


Figure 3. 8 User Requirement 6.

Figure 3.8 is the patient information needed for storage. Patient's Name, Identity Card Number/Passport Number, Age, Gender, Date of Birth, Blood Collection Date, Name of the Blood Test lab (such as Pathlab, Innoquest), Name of Clinic for requesting blood test report (eg. Prime Care Clinic, Lim medical Clinic).

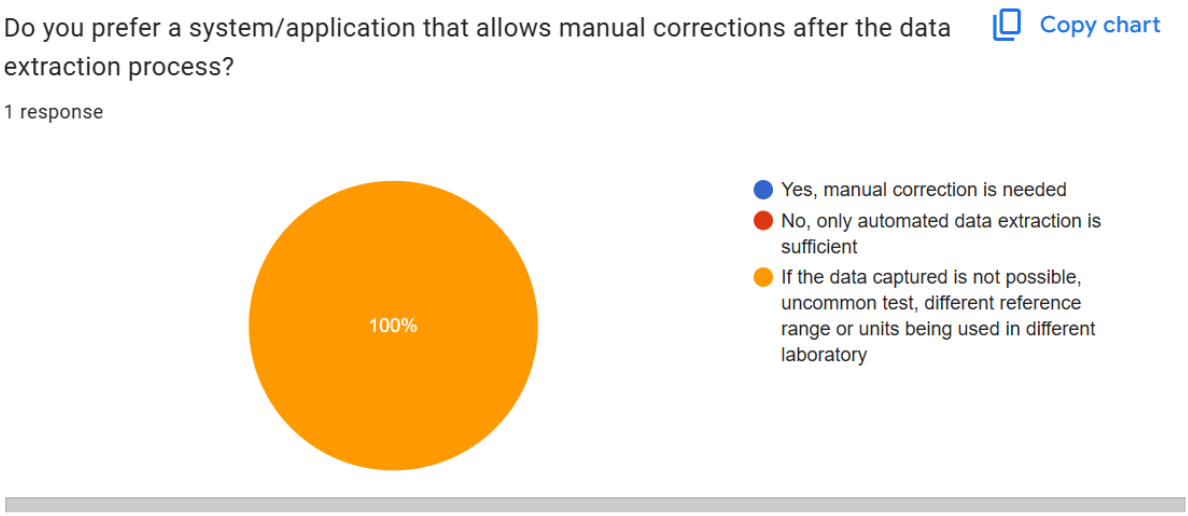


Figure 3. 9 User Requirement 7.

Figure 3.9 shows the manual corrections needed in the system. From response, manual corrections when the data capture is not possible, uncommon test.

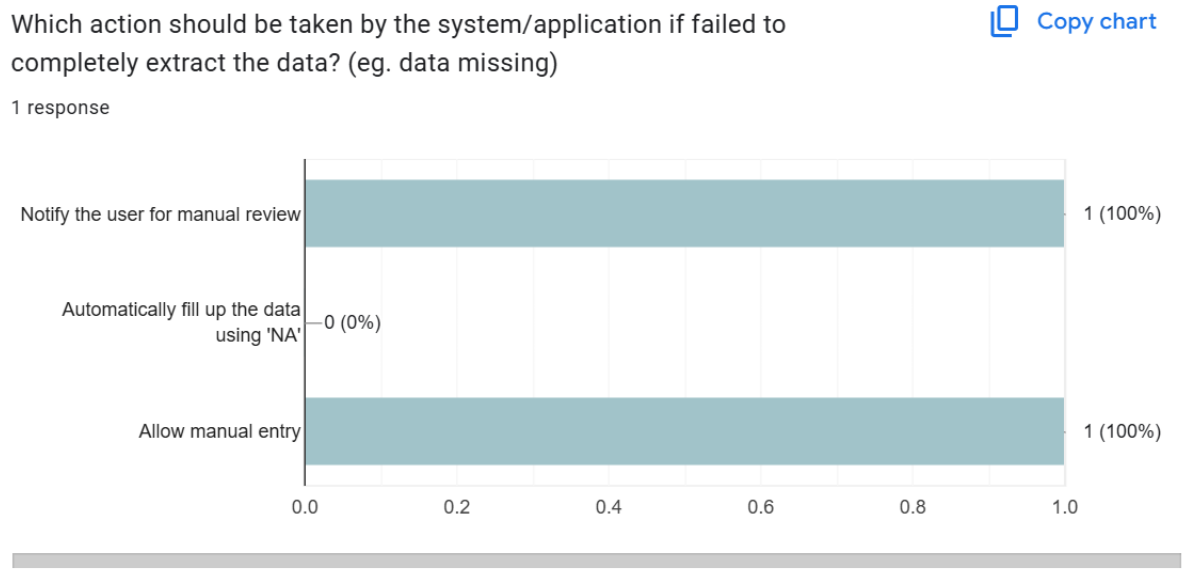


Figure 3. 10 User Requirement 8.

Figure 3.10 is to ask if the system failed to completely extract data. From the response, the doctor wants the notify the user for manual review function and allow for manual entry function.

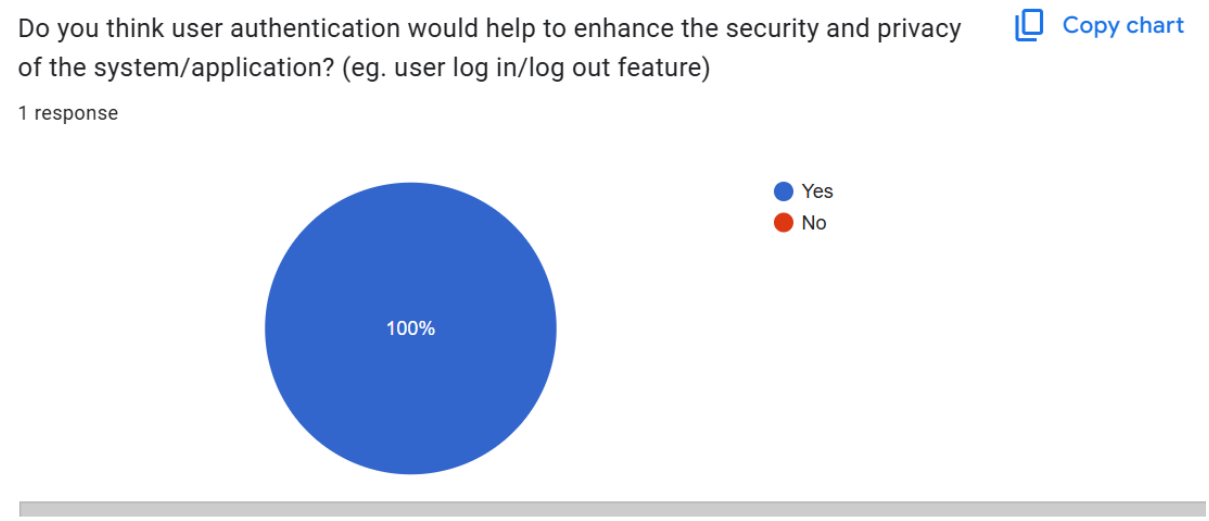


Figure 3. 11 User Requirement 9.

From figure 3.11, the user authentication as log in and log out function is needed.

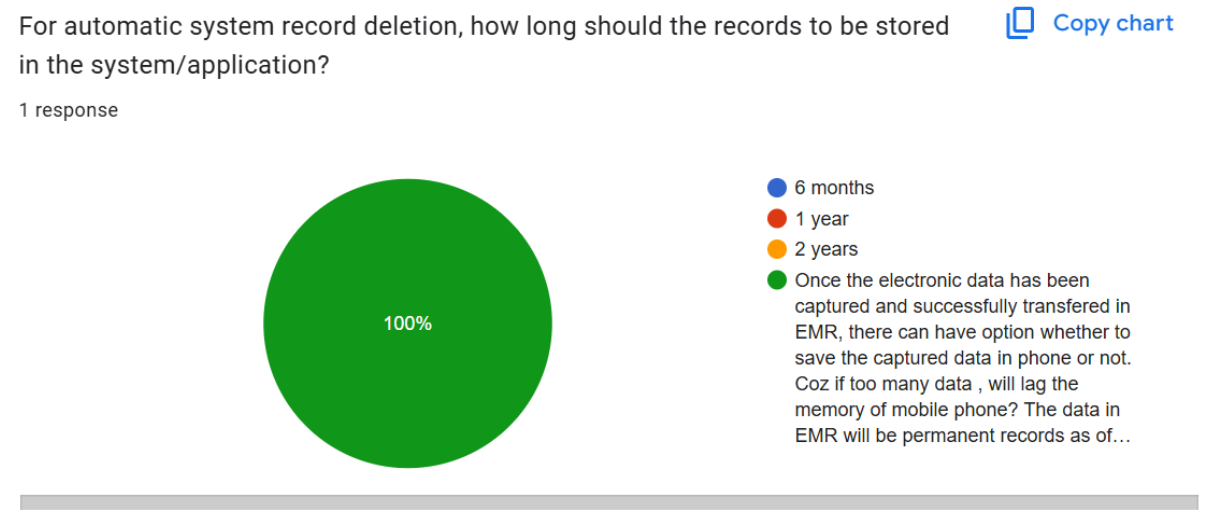


Figure 3. 12 User Requirement 10.

In figure 3.12, the question how long the record should save, the doctor request for cloud storage to have a bigger storage.

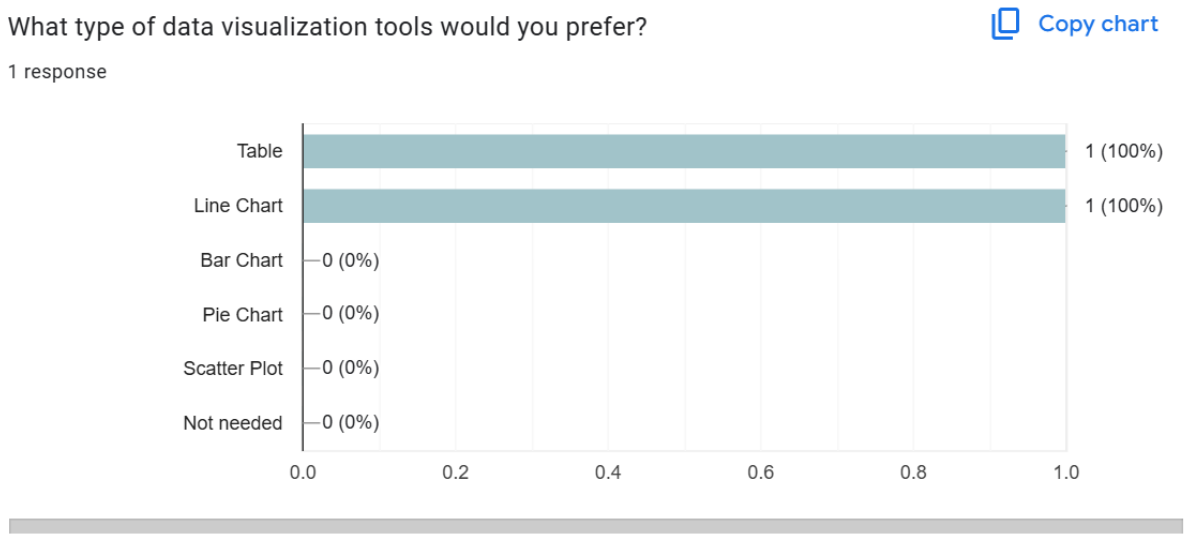


Figure 3. 13 User Requirement 11.

From figure 3.13, the data visualization tools “table”, “line chart” needed for the doctor.

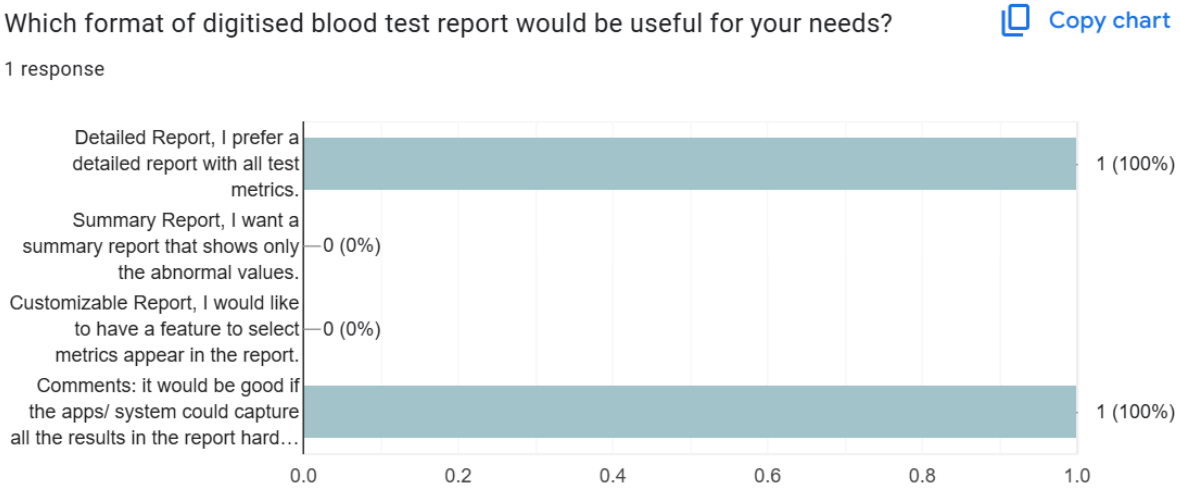


Figure 3. 14 User Requirement 12.

Figure 3.14 is the report format for the needs by the doctor. Detailed report with all test metrics needs.

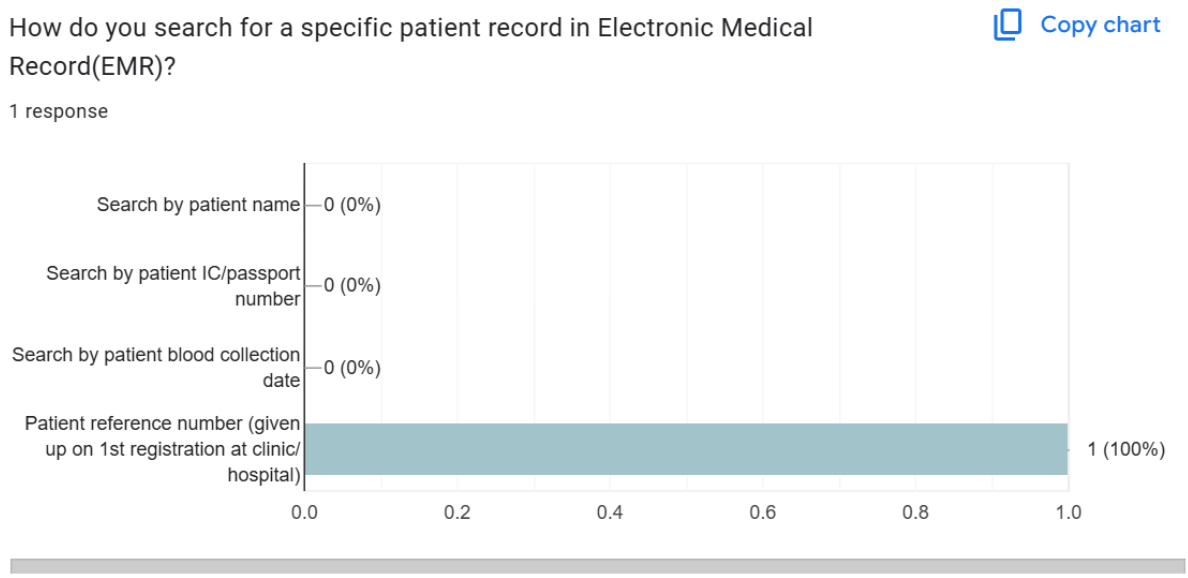


Figure 3. 15 User Requirement 13.

The figure 3.15, searching method for patient record is based on patient reference number, which is the patient_id.

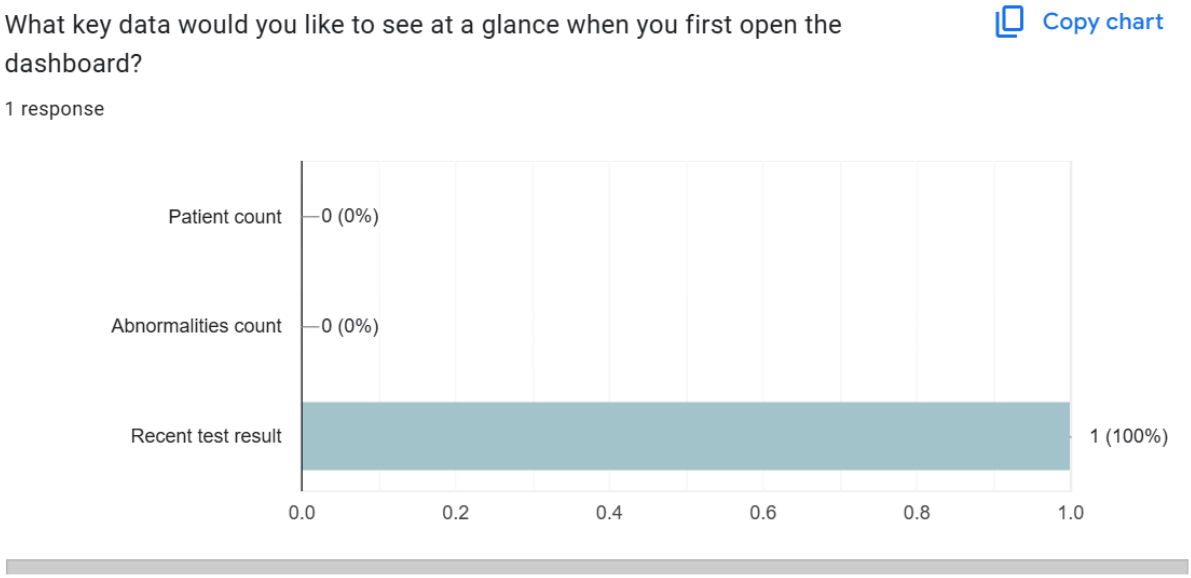


Figure 3. 16 User Requirement 14.

Figure 3.16 is about dashboard. The doctor needs to see the recent test result when first open the dashboard.

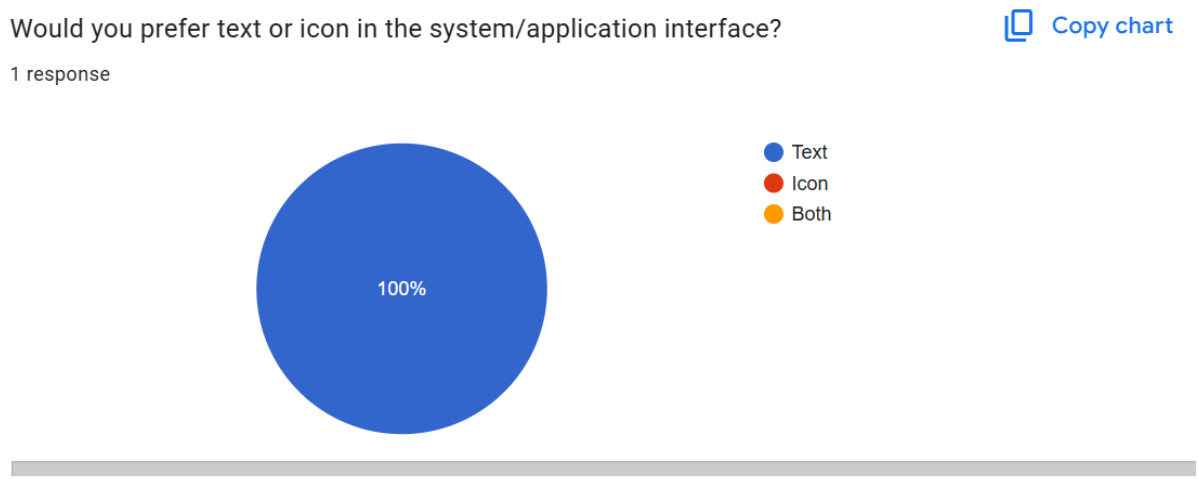


Figure 3. 17 User Requirement 15.

Figure 3.17 show that the system interface used is the text.

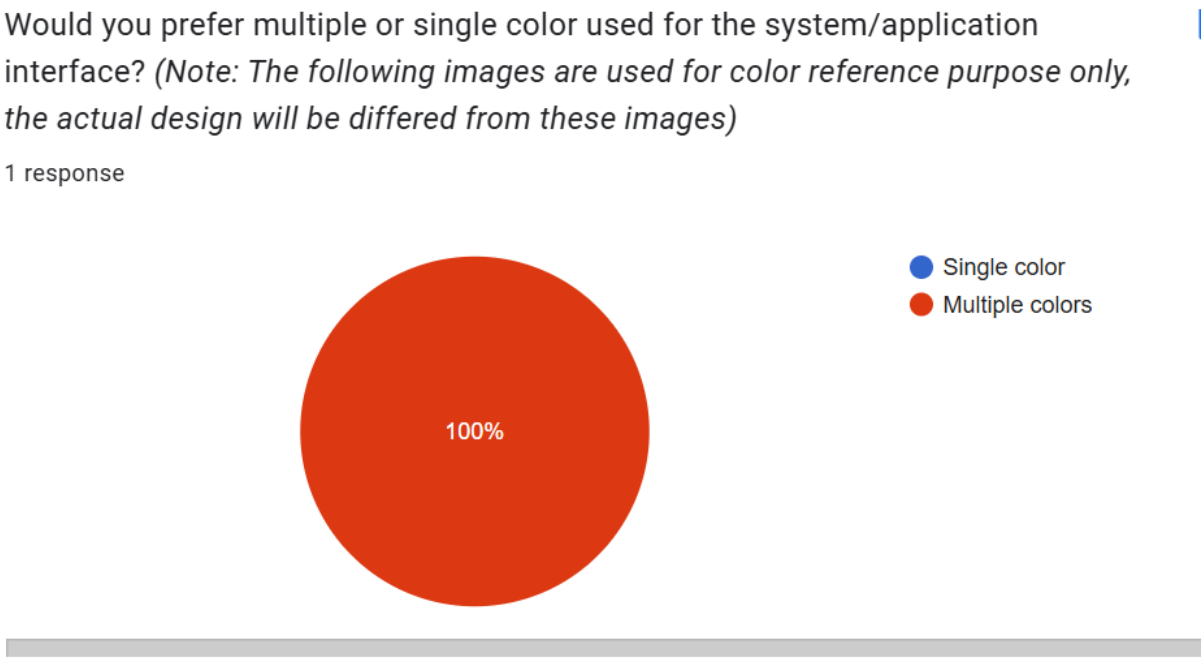


Figure 3. 18 User Requirement 16.

Figure 3.18 shows that multiple colours used for the system interface.

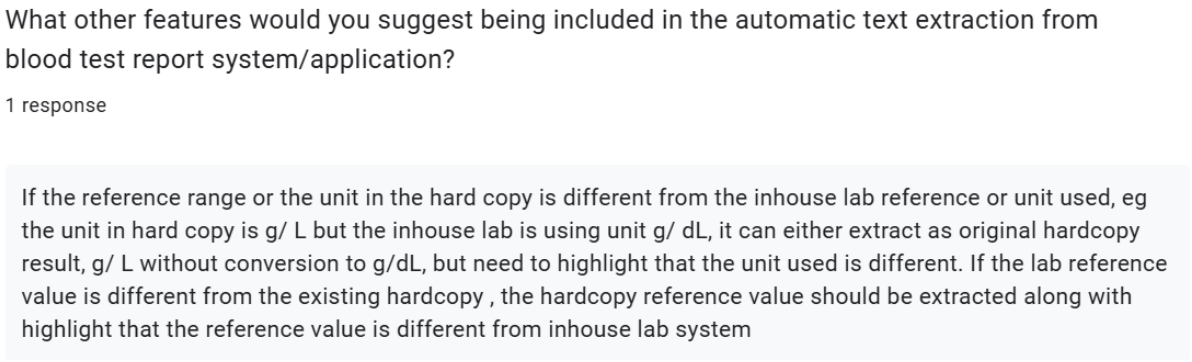


Figure 3. 19 User Requirement 17.

Figure 3.19 is another feature suggested by the doctor, the doctor needs the unit converter as the unit will be different from different laboratory. For example, BMC lab used g/dL unit for haemoglobin (Figure 3.20) while Gribbles lab used g/L unit for haemoglobin (Figure 3.21).

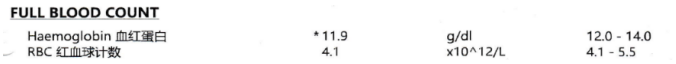


Figure 3. 20 Example from BMC lab

Figure 3. 21 Example from Gribbles Lab

System Requirements

Software

SOFTWARE	PURPOSE
Canva	The reason of using Canva is to design the UI/UX interfaces of this project
Draw.io	Used to create diagrams, flowcharts.
Drawify	Used to create diagram for Agile
Google Chrome	Web Browser for all searching
Navicat Database	Managed database to store the data in this project
Google Form	Create questionnaire, collect data, and store responses in Google Sheets.
Google Scholar	Searching for research paper to review.
Microsoft Office 365	Main purpose of using Microsoft 365 Office is to use Microsoft Words for the report documentation and Microsoft Powerpoint for the slide presentation

Python	The focus of Python is to generate code for this project
SpaCy	processing and structuring extracted data
Tesseract OCR	text extraction from images of blood test reports.
Visual Studio Code	The focus of Visual Studio Code is to generate code for this project

Table 3. 1 Software used for system.

Hardware

HARDWARE	SPECIFICATION	PURPOSE
Laptop	Operating System: Windows 11 Home Laptop Brand: HP Laptop 14s dq2xxx Processor: 11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40GHz Memory: 12.0 GB RAM	The purpose of using a laptop is to develop the whole process of the system and access the system through website
Printer	Printer Brand: Canon Pixma E410	The printer is used to print out the hardcopy for the paper.

Table 3. 2 Hardware used for system.

3.3.2 Plan and Design

Context Diagram

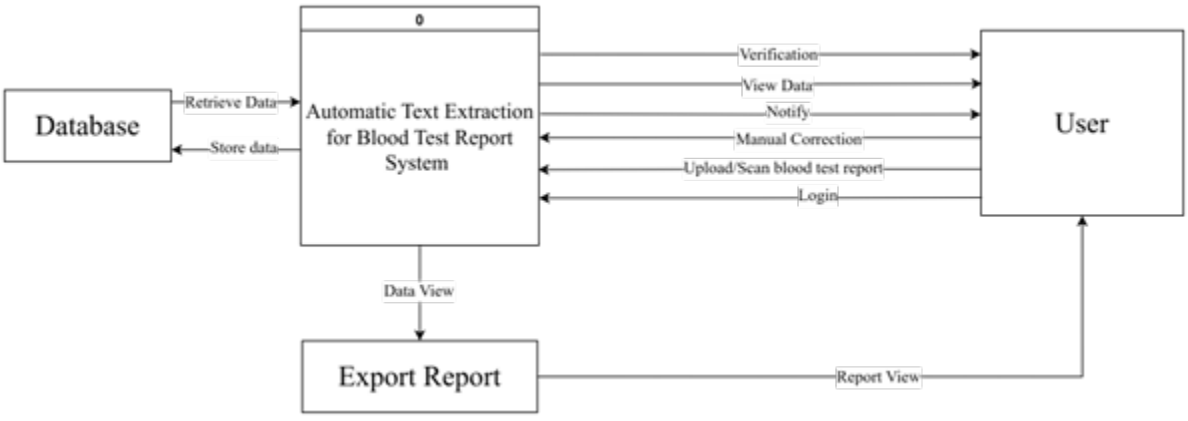


Figure 3. 3. 1 Context Diagram

Figure 3.3.1 shows the context diagram. The inputs and outputs of user in automatic Text Extraction for Blood Test Report System. These shows the general views of function of user can used to interact with system.

Data Flow Diagram

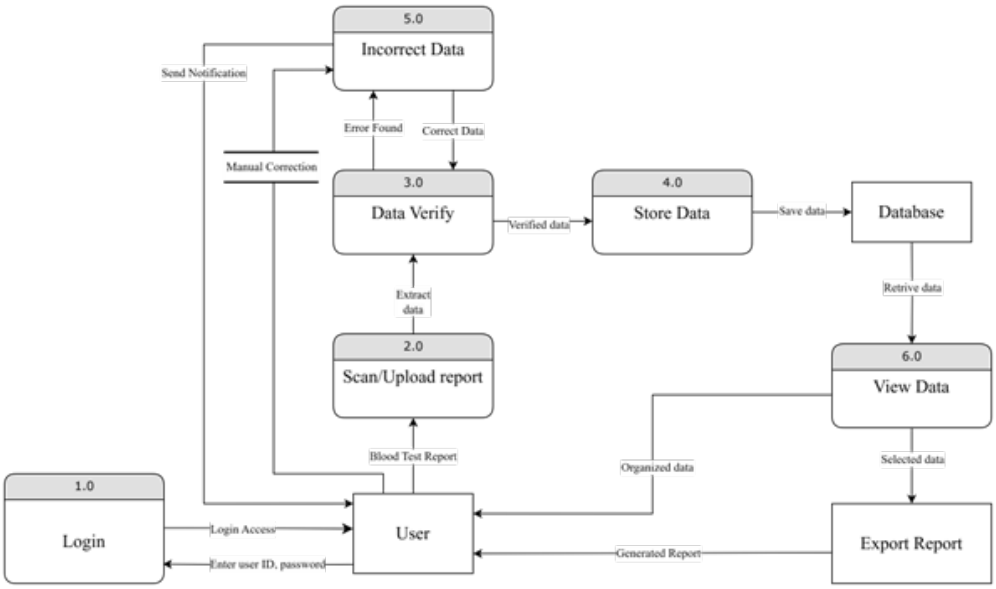


Figure 3. 3. 2 Data flow Diagram Level 1

Figure 3.3.2 shows the data flow diagram level 1. Login (Process 1.0) is the user enters their credentials (ID and password) to access the system. Upon successful login, the user gains access to the system. Scan/Upload Report (Process 2.0) is the user can either upload a blood test report image or scan it directly using the system. This process extracts the report's raw data. Data Verify (Process 3.0) is the extracted data is reviewed for accuracy. If errors are found, the user is notified and can proceed to manually correct the incorrect data. Store Data (Process 4.0) is the data verified and corrected; it is saved securely in the database for future use. Incorrect Data (Process 5.0), If errors are detected during verification, this process alerts the user and facilitates manual correction to ensure the data's accuracy. View Data (Process 6.0) is the user can retrieve and view the organized data from the database for analysis or validation. After selecting the data, the system generates a structured report for offline use or sharing.

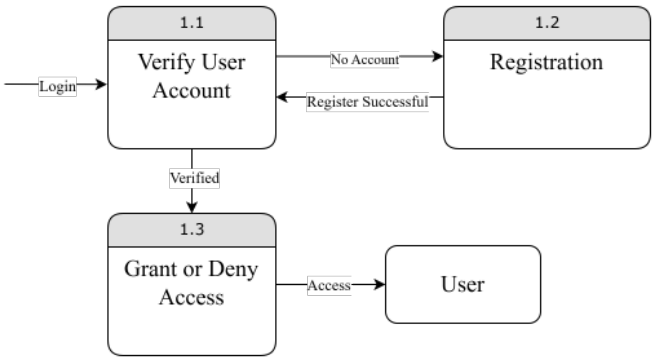


Figure 3. 3. 3 Data flow Diagram Level 2 for Process 1.0 Login.

Figure 3.3.3 is Data flow Diagram Level 2 for Process 1.0 Login. 1.1 is verify user account, the user provides login credentials, which are validated against the system's database. 1.2 is registration, if the user has no account registration is required. 1.3 is Grant/Deny Access. If the credentials are valid, access is granted; otherwise, an error message is sent to the user.

Figure 3. 3. 4 Data flow Diagram Level 2 for Process 2.0 Scan/Upload Report.

Figure 3.3.4 is Data flow Diagram Level 2 for Process 2.0 Scan/Upload Report. 2.1 is Upload Report. The user uploads the report image from their device. 2.2 is Extract Text. The system extracts data from the uploaded image using OCR (Optical Character Recognition). 2.3 is Preprocess Data. The extracted text is cleaned and formatted for verification.

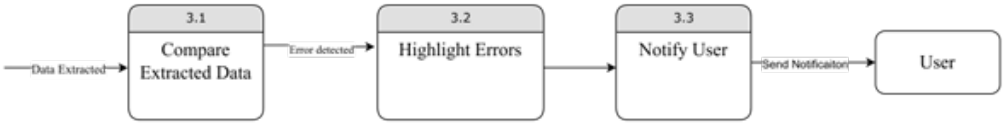


Figure 3. 3. 5 Data flow Diagram Level 2 for Process 3.0 Data Verify.

Figure 3.3.5 is Data flow Diagram Level 2 for Process 3.0 Data Verify. 3.1 is Compare Extracted data. Extracted data is compared against predefined standards or rules. 3.2 is

Highlight Errors. If discrepancies are found, errors are highlighted for correction. 3.3 is Notify User. The system notifies the user of errors or confirms that data is correct.

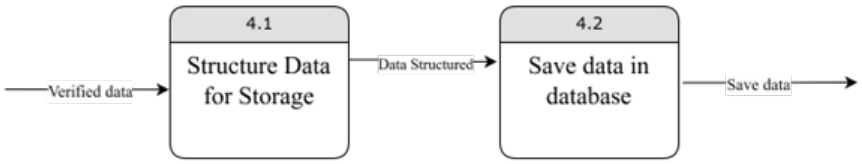


Figure 3. 3. 6 Data flow Diagram Level 2 for Process 4.0 Store Data.

Figure 3.3.6 is Data flow Diagram Level 2 for Process 4.0 Store Data. 4.1 is Structure Data. The verified data is organized into a structured format. 4.2 is Save to Database. The structured data is stored in the database for future retrieval.

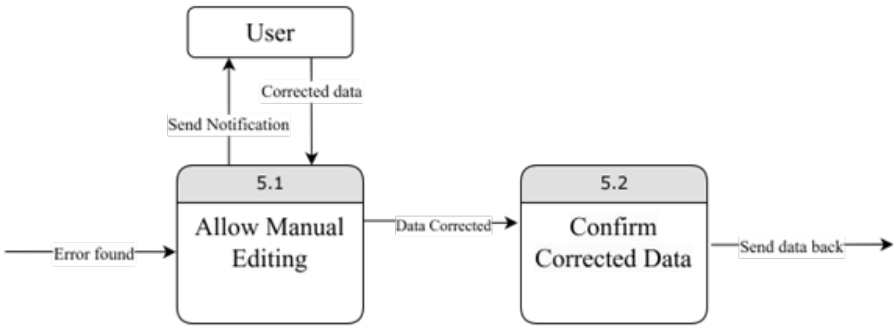


Figure 3. 3. 7 Data flow Diagram Level 2 for Process 5.0 Incorrect Data.

Figure 3.3.7 is Data flow Diagram Level 2 for Process 5.0 Incorrect Data. 5.1 is Manual Editing. The user manually edits the incorrect data. 5.2 is Confirm Corrected Data. The corrected data is reviewed and confirmed before returning to the verification step.

Figure 3. 3. 8 Data flow Diagram Level 2 for Process 6.0 View Data.

Figure 3.3.8 is Data flow Diagram Level 2 for Process 6.0 View Data. 6.1 is Retrieve Data. The system retrieves data from the database based on the user's request. 6.2 is Display Information. The retrieved data is presented to the user in a clear and organized format.

Database design

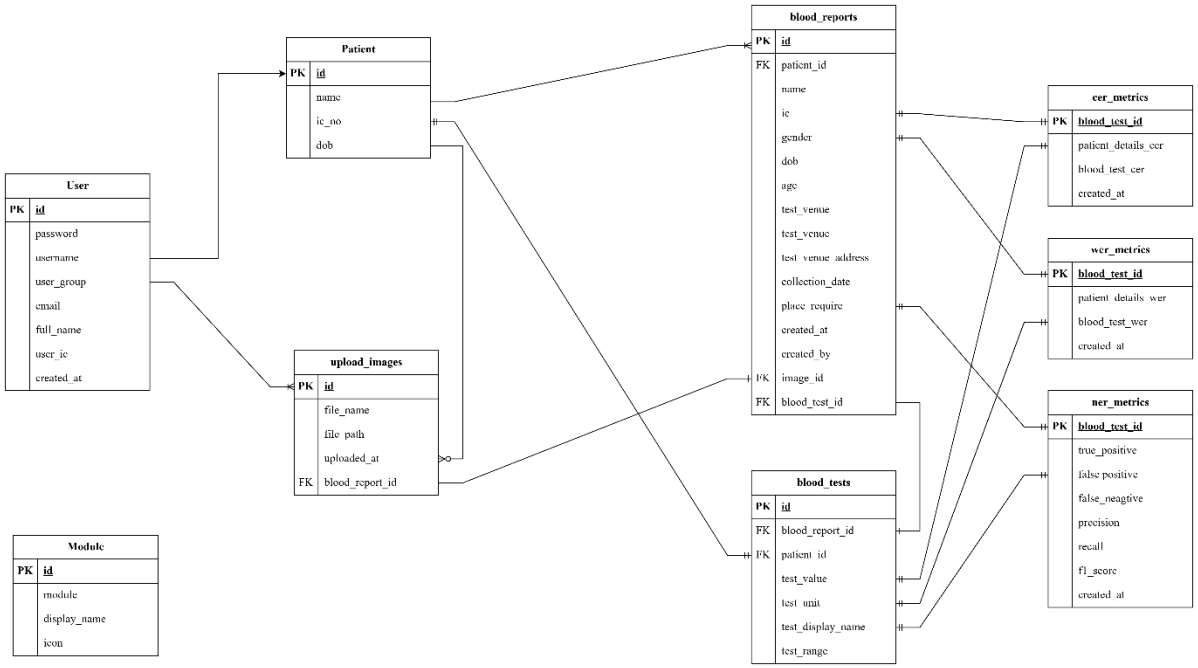


Figure 3. 3. 9 Entity Relationship Diagram

Figure 3.3.9 shows the Entity Relationship Diagram. It is a database design.

Data Dictionary

User

Attribute	Data Type	Description	Constraints
id	INT	Unique ID for the user, for log in used.	Primary Key, Null
password	VARCHAR(255)	Unique password for the user, for log in used.	Not Null
username	VARCHAR(255)	Login username	Unique
user_group	VARCHAR(255)	Role or group of the user (e.g., admin)	Not Null
email	VARCHAR(255)	User email address	Unique
full_name	VARCHAR(255)	User's full name	Not Null
user_ic	VARCHAR(255)	User's IC number	Not Null

created_at	timestamp	Account creation date	DEFAULT CURRENT_TIMESTAMP

Table 3. 3. 1 Database for user table

Patient

Attribute	Data Type	Description	Constraints
id	INT	Unique ID for the patient	Primary Key, Not Null
name	VARCHAR(100)	Full name of the patient	Not Null
ic_no	INT	Identity Card number or passport number	Not Null
dob	DATE	Date of birth	Not Null

Table 3. 3. 2 Database for patient table

Module

Attribute	Data Type	Description	Constraints
id	INT	Unique ID for the module	Primary Key, Not Null
module	VARCHAR(100)	Unique module	Not Null
display_name	VARCHAR(100)	Display Name for the module	Not Null
icon	VARCHAR(100)	Icon for the module	Not Null

Table 3. 3. 3 Database for module table

Upload_images

Attribute	Data Type	Description	Constraints
id	INT	Unique ID for the uploaded file	Primary Key, Auto_Increment, Not Null
file_name	VARCHAR(255)	Name of the uploaded file	Not Null
file_path	VARCHAR(255)	Path to the file on the server	Not Null
uploaded_at	TIMESTAMP	Timestamp when file was uploaded	DEFAULT CURRENT_TIMESTAMP, Not Null

blood_report_id	INT	Foreign key linking to blood_reports	Foreign Key, Not Null

Table 3. 3. 4 Database for upload_image table

Blood_reports

Attribute	Data Type	Description	Constraints
id	INT(11)	Unique ID for the blood report	Primary Key, Auto_Increment, Not Null
name	VARCHAR(100)	Patient name (if not using patient_id)	Null
ic	VARCHAR(20)	Patient IC/passport number	Null
gender	ENUM	Patient gender	Null
dob	DATE	Date of birth	Null
age	INT(10)	Age (as per report)	Null
test_venue	VARCHAR(150)	Venue name where test was done	Null
test_venue_address	VARCHAR(255)	Full address of test venue	Null
collection_date	DATE	When the blood was collected	Null
place_require	VARCHAR(255)	Clinic or place that requested the test	Null
created_at	DATETIME	Report creation datetime	DEFAULT CURRENT_TIMESTAMP, Not Null
created_by	VARCHAR(100)	ID of user who entered the data	Not Null
image_id	INT(11)	Id of uploaded image	Foreign Key
patient_id	INT(11)	Id from patient table	Foreign Key
blood_test_id	INT(11)	Id form blood test table	Foreign Key

Table 3. 3. 5 Database for blood_reports table

Blood_tests

Attribute	Data Type	Description	Constraints

id	INT	Primary key	Primary Key, Not Null
blood_report_id	INT	Id from blood_reports table	Foreign Key
patient_id	INT	Id from patient table	Foreign Key
test_value	VARCHAR(50)	e.g., 'haemoglobin_value'	Null
test_unit	VARCHAR(20)	e.g., ‘haemoglobin_unit’	Null
test_display_name	VARCHAR(100)	e.g., 'haemoglobin_display_name'	Null
test_range	VARCHAR(50)	e.g., 'haemoglobin_range'	Null

Table 3. 3. 6 Database for blood_tests table

CER_metrics

For counting Character Error Rate.

Attribute	**Data Type**	**Description**	**Constraints**
blood_test_id	VARCHAR(50)	ID for the blood test report	Primary Key, Not Null
patient_details_cer	Decimal(10,2)	CER for patient details	
blood_test_cer	Decimal(10,2)	CER for blood tests details	
created_at	DATETIME	CER creation datetime	DEFAULT CURRENT_TIMESTAMP, Not Null

Table 3. 3. 7 Database for CER_metrics table

WER_metrics

For counting Word Error Rate.

Attribute	**Data Type**	**Description**	**Constraints**
blood_test_id	VARCHAR(50)	ID for the blood test report	Primary Key, Not Null
patient_details_wer	Decimal(10,2)	WER for patient details	
blood_test_wer	Decimal(10,2)	WER for blood tests details	
created_at	DATETIME	WER creation datetime	DEFAULT CURRENT_TIMESTAMP, Not Null

Table 3. 3. 8 Database for WER_metrics table

NER_metrics

For counting Named Entity Recognition.

Attribute	Data Type	Description	Constraints
blood_test_id	VARCHAR(50)	ID for the blood test report	Primary Key, Not Null
true_positive	INT	Count of true positives	true_positives
false_positive	INT	Count of false positives	false_positives
false_negative	INT	Count of false negatives	false_negatives
precision	DECIMAL(5,2)	Precision percentage	precision
recall	DECIMAL(5,2)	Recall percentage	recall
f1_score	DECIMAL(5,2)	F1 Score percentage	f1_score
created_at	DATETIME	WER creation datetime	DEFAULT CURRENT_TIMESTAMP, Not Null

Table 3. 3. 9 Database for NER_metrics table

Interface Design

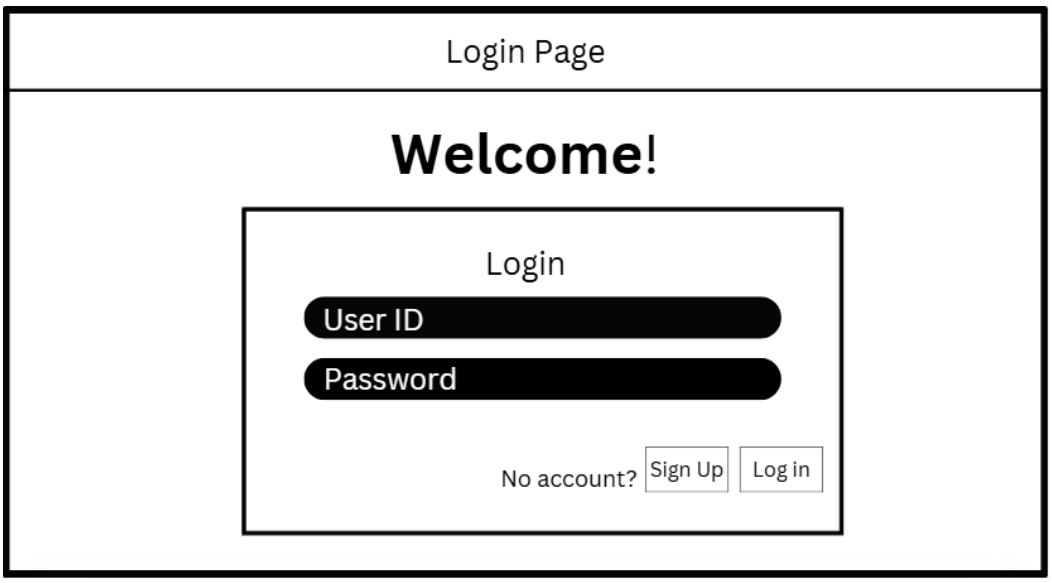


Figure 3. 3. 10 Login Page

Figure 3.3.10 is the login page, it serves as the entry point for users. It provides fields for the user's credentials, including username and password, to access the system. If the user doesn't

have an account, there is an option to register. The page ensures secure authentication to protect user data.

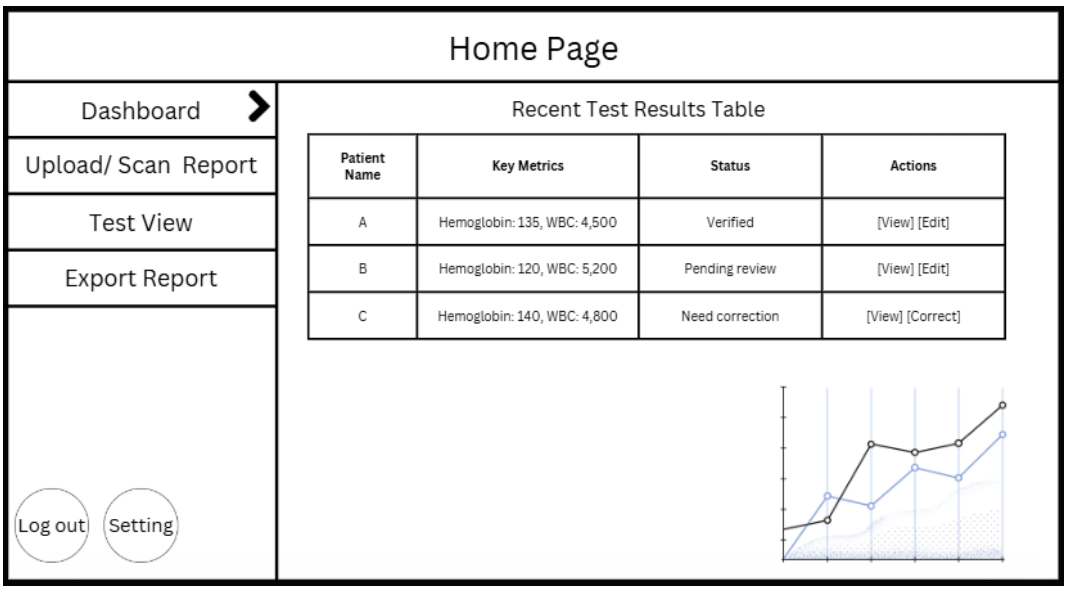


Figure 3. 3. 11 Home Page

Figure 3.3.11 is the home page of the system. The dashboard or home page offers a summary of key functionalities and information. It includes recent test result table and line chart.

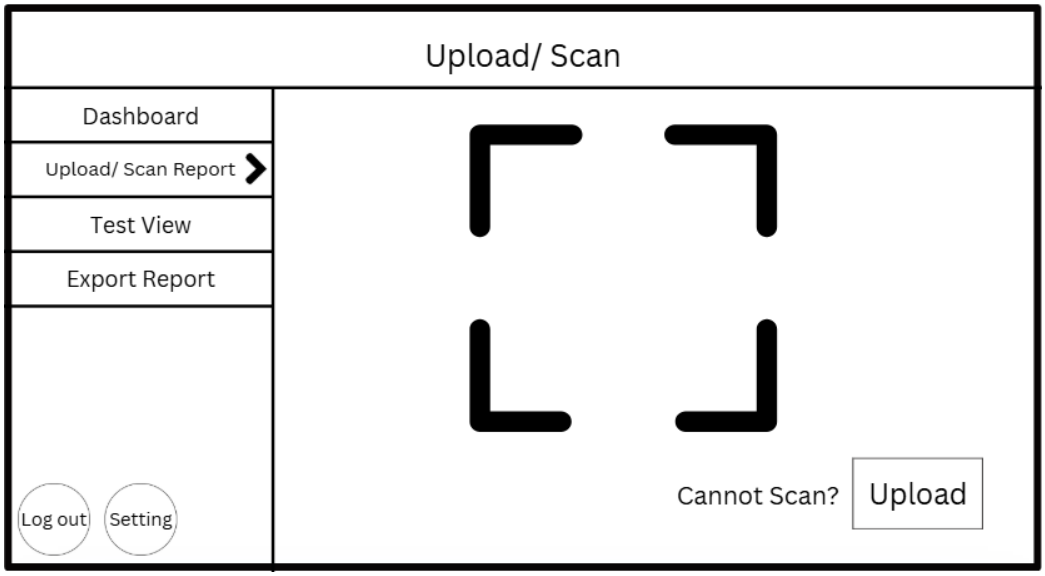


Figure 3. 3. 12 Upload Page

Figure 3.3.12 is the upload page. This page allows users to upload or scan blood test reports. Users can either drag and drop an image or select a file from their system.

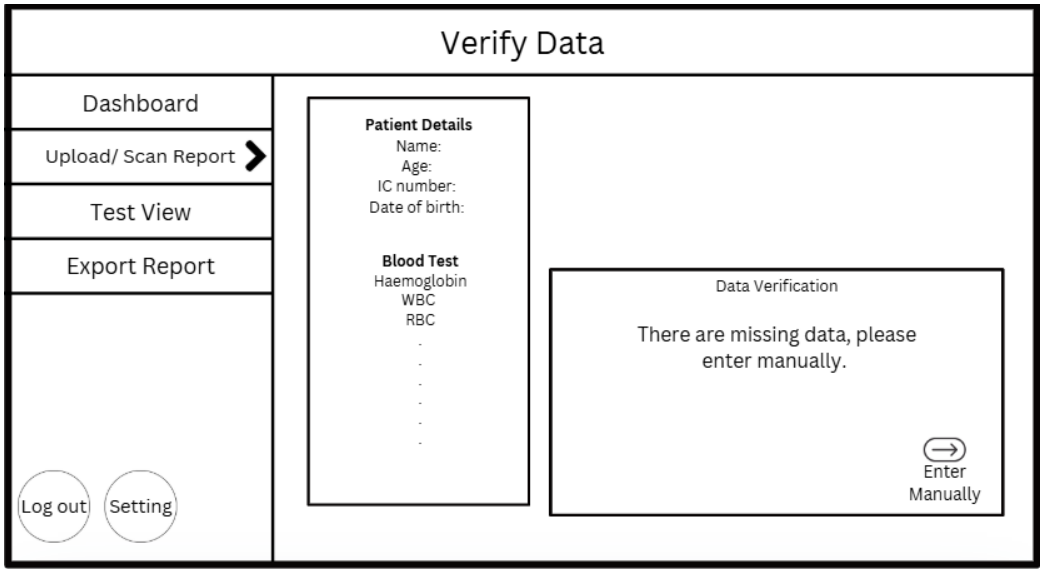


Figure 3. 3. 13 Data Verification page with data missing or incorrect data.

Figure 3.3.13 is the data verification page with data missing or incorrect. This page displays extracted data from the uploaded report. When data is missing or incorrect, it highlights these issues, prompting users to correct or manually input the missing information. The manual entry provided.

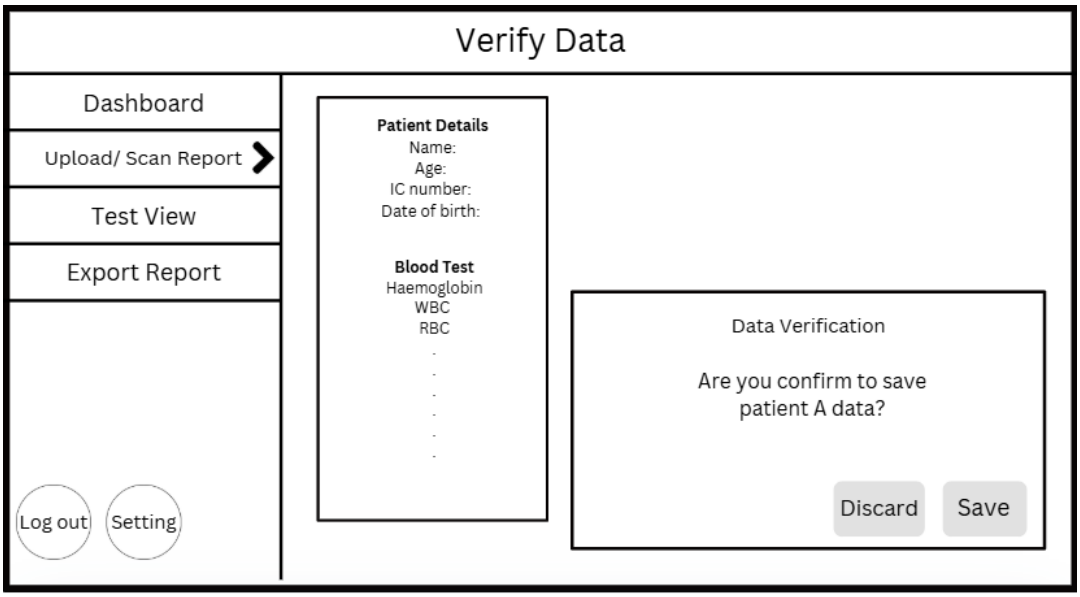


Figure 3. 3. 14 Data Verification page.

Figure 3.3.14 is the data verification page with normal data extract. This page shows the extracted data that has been successfully verified. The user will need to confirm before saving the data into database.

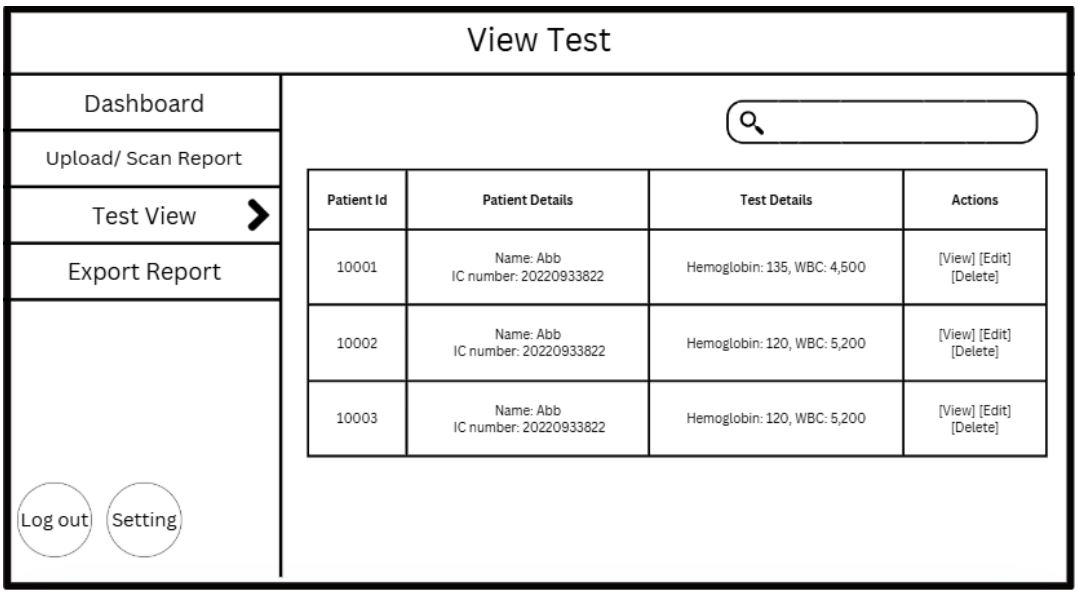


Figure 3. 3. 15 View test page.

Figure 3.3.15 is the view test page. This page lists all the stored blood test reports in tabular form. Users can search the records by various criteria such as patient ID, patient name.

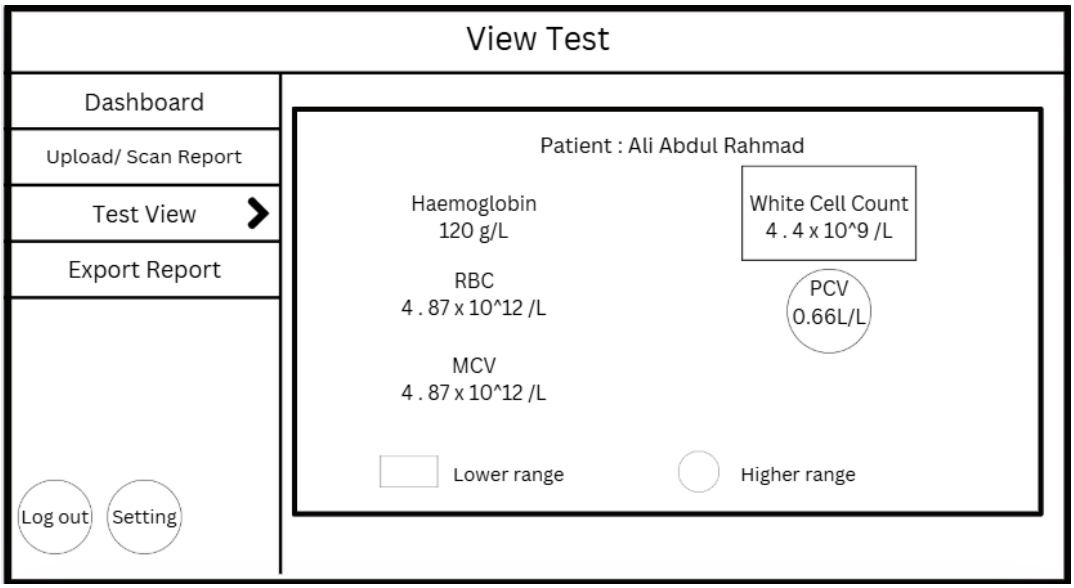


Figure 3. 3. 16 View test for selected patient page.

Figure 3.3.16 is the view selected patient blood test page. A detailed view for a specific patient, displaying all their test results. It includes comprehensive data, trends, and insights about the patient's health over time. Users can analyse individual records or compare results. The value of test lower or higher the range will be highlighted in different method.

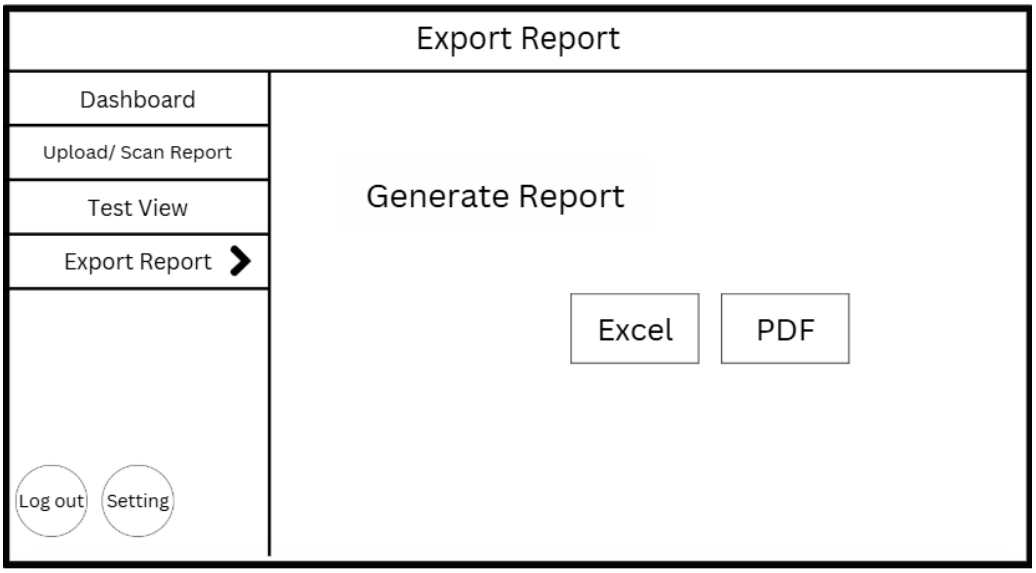


Figure 3. 3. 17 Export report page.

Figure 3.3.17 is the export report page. This page enables users to export blood test reports in various formats (e.g. PDF, Excel). Users can share or store reports externally.

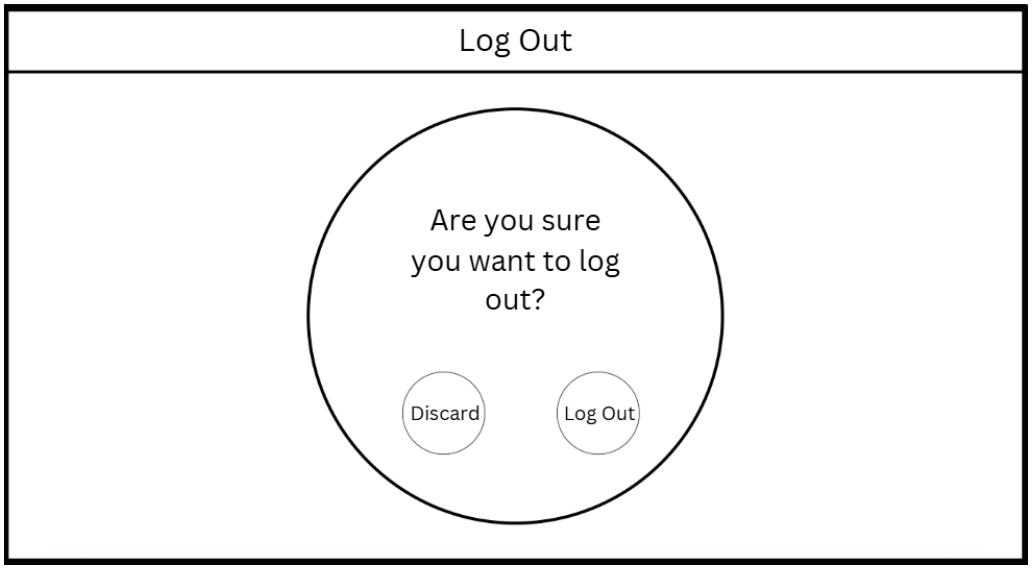


Figure 3. 3. 18 Log Out page.

Figure 3.3.18 is the log out page. The log out page provides a secure way for users to exit the system. It confirms that the session has been successfully ended

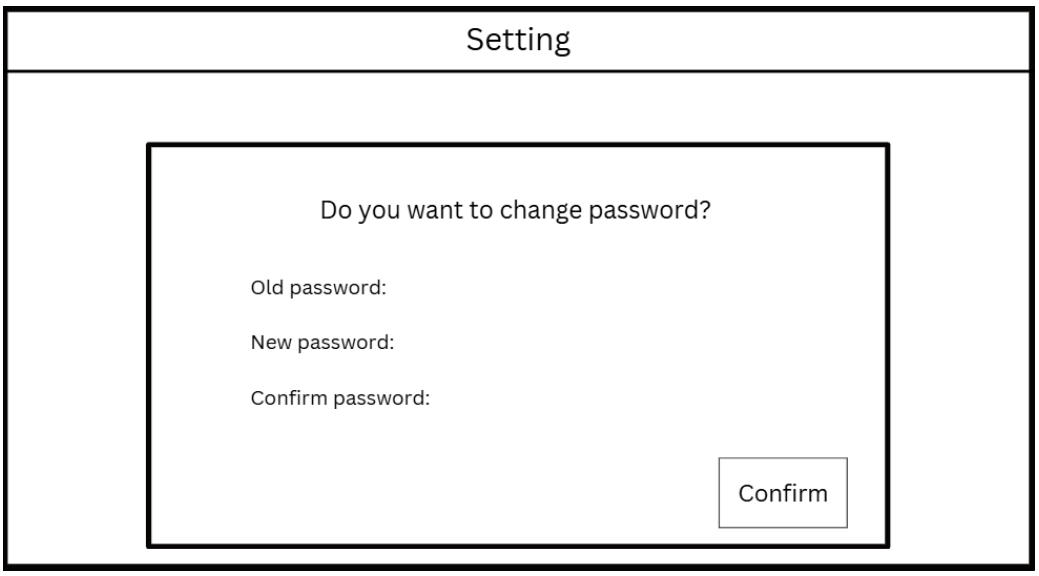


Figure 3. 3. 19 Setting page.

Figure 3.3.19 is the setting page. The settings page allows users to customize their password for the account.

3.3.3 Planning for Implementation & Testing

The implementation phase involves coding the system based on the design specifications, while the testing phase includes functional, performance, and testing to ensure system reliability and accuracy.

Tesseract OCR

Tesseract OCR is an open-source Optical Character Recognition (OCR) engine used to extract text from images. It is useful for converting printed or handwritten text into machine-readable data. Tesseract OCR plays a crucial role in processing scanned blood test reports by Preprocessing Images, Text Extraction, Data Structuring. Preprocessing images is to enhance the image quality through binarization and thresholding to improve text recognition accuracy. Text extraction is to identify and extract textual content from the scanned reports. Data

structuring is formatting the extracted text to align with predefined medical data fields for accurate storage and retrieval. Tesseract OCR is integrated with Python and used in conjunction with image processing libraries.

SpaCy

SpaCy is a popular, fast, and efficient natural language processing (NLP) library in Python that is used for a variety of NLP tasks, such as part-of-speech tagging, named entity recognition (NER), dependency parsing, and text classification. It is powerful for processing large volumes of text quickly and accurately. SpaCy plays a significant role in processing and structuring the extracted text from Tesseract OCR, such as text preprocessing, Named Entity Recognition, syntax analysis, text classification, data structuring. In text preprocessing, tokenization, lemmatization, common words removal will be process. For example, Tokenization, SpaCy can be used to split the raw text into individual tokens, which is essential for understanding the structure and meaning of the text. Lemmatization reduces words to their base or dictionary form (e.g., "running" becomes "run"). Common words removal is the common words like "the," "is," and "in" will be removed as they don’t provide much value. Named Entity Recognition (NER) will be used to identify and classify key information such as medical terms like cholesterol, glucose, patient details, testing dates or times of report generation, measurement units. For Syntax Analysis, SpaCy can help to analyze sentence structures to understand how words in a sentence are related. SpaCy can help to classify the type of report. After extracting the text from Tesseract OCR and using SpaCy to process it, SpaCy help to format the extracted information into structured data such as tables. This structured data can then be stored in a database or used for reporting.

In summary, Tesseract OCR is responsible for extracting the text from images, while SpaCy can help process and structure that text into meaningful data. SpaCy’s NLP capabilities,

especially NER and text classification, enhance the accuracy and efficiency of transforming raw OCR text into machine readable data.

Testing Form

NO	MODULE	STATUS		COMMENT
1	Log In	Pass	Fail	
2	Home Page	Pass	Fail	
3	Upload Page	Pass	Fail	
4	Data Verification Page (with missing or incorrect Data)	Pass	Fail	
5	Data Verification Page (with all data correct)	Pass	Fail	
6	View Test Page	Pass	Fail	
7	Search Bar	Pass	Fail	
8	View Test for Selected Patient Page	Pass	Fail	
9	Export Report Page	Pass	Fail	
10	Log Out	Pass	Fail	
11	Setting Page	Pass	Fail	

This test performed by:

Name: _____

Signature: _____

Date: _____

Table 3. 3. 10 Implementation & Testing

Performance Metrics

To evaluate the performance of the integrated OCR and NLP system, the following metrics are commonly used:

1. Character Error Rate (CER): CER measures the number of character-level errors in the OCR output compared to the ground truth. It is calculated by counting the number of substitutions, insertions, and deletions made when comparing the OCR result to the actual text. A lower CER is a sign of better OCR performance. (Thennal et al., 2024)

   Formula:

   $$CER = (S + I + D) / N$$

   Where:

   S = Number of substituted characters

   I = Number of inserted characters

   D = Number of deleted characters

   N = Total number of characters

2. Word Error Rate (WER): WER is a similar metric but evaluates the errors at the word level. It is calculated as the sum of substitutions, insertions, and deletions at the word level, divided by the total number of words in the reference text. A lower WER is a sign of better OCR performance. (Ali and Renal, 2018)

   Formula:

   $$WER = (S + I + D) / W$$

   Where:

S = Number of substituted words

I = Number of inserted words

D = Number of deleted words

W = Total number of words

3. Named Entity Recognition (NER): For the NLP part, NER is a critical task where the system identifies and classifies entities such as names of medications, dosages, or patient identifiers. The accuracy of NER is measured by precision, recall, and F1 score, which help in evaluating how well the system can identify relevant entities in the processed text. A low precision means the system identifies many incorrect entities. A low recall means the system misses important entities. A high F1-score indicates a balance between precision and recall, improving the quality of extracted structured data. (Sun et al., 2021)

   Formula:

$$\text{Precision} = TP / TP + FP$$

$$\text{Recall} = TP / TP + FN$$

$$\text{F1 Score} = 2 \times [(\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall})]$$

   Where:

   TP (True Positives) = Correct identified entities

   FP (False Positives) = Incorrect identified entities

   FN (False Negatives) = Missed entities

3.4 Summary

In this chapter, the Agile methodology was introduced as the development approach for the Automated Blood Test Data Extraction System for Medical Record Management Using OCR. The Agile approach emphasizes iterative development and continuous improvement, allowing the system to evolve quickly in response to user feedback and changing requirements. Agile ensures that both technical accuracy and user satisfaction are prioritized throughout the development process.

To gather detailed requirements from medical staff, a questionnaire feature has been integrated into the system. This feature enables the collection of specific preferences from key stakeholders, such as doctors, nurses, pharmacists, specialists, and others to get their unique needs of each user group effectively. The questionnaire will serve as a valuable tool for capturing requirements related to system usability, functionality, and overall performance.

CHAPTER 4 IMPLEMENTATION

4.1 Introduction

This chapter will introduce the implementation of Automatic Text Extraction Using Optical Character Recognition (OCR) for Blood Test Report Management. This project will adopt the Agile approach methodology. The development of the proposed web application will be based on the upfront design and planning in an iterative phase. Both the front-end and back-end components have been developed as a fully functional, user-friendly, gamified web application designed to help learn programming. In the front-end development, it utilizes HTML, CSS, PHP, and others to create a responsive and interactive user interface, including features such as dashboard page, upload/scan page, view page, export page with different content types. The back end is built using PHP, Python, Javascript, swal.fire and MySQL database to handle OCR, handel NLP, data management, and other functions.

4.2 Installation and Configuration of System's Components

In this project, some installation and configuration are required before starting the implementation of the proposed system. The necessary software tools are XAMPP, Navicat and Visual Studio Code.

4.2.1 XAMPP

XAMPP is a free and open-source cross-platform web server solution stack package. It includes components such as Apache, MariaDB (or MySQL), and interpreters for scripting languages like PHP and Perl. XAMPP is useful for developers to set up a local development environment to test and run web applications. In this project, XAMPP will be used to host the PHP-based system locally using Apache, manage the database using MariaDB/MySQL, and provide a simplified setup and control environment for development.

Steps to Install XAMPP:

1. Visit the official XAMPP website and download the installer suitable for Windows as Figure 4.2.1.1.
2. Run the installer and follow the setup instructions.
3. Choose a folder for installation (e.g., C:\xampp\) as Figure 4.2.1.2.
4. Once installed, launch the XAMPP Control Panel.

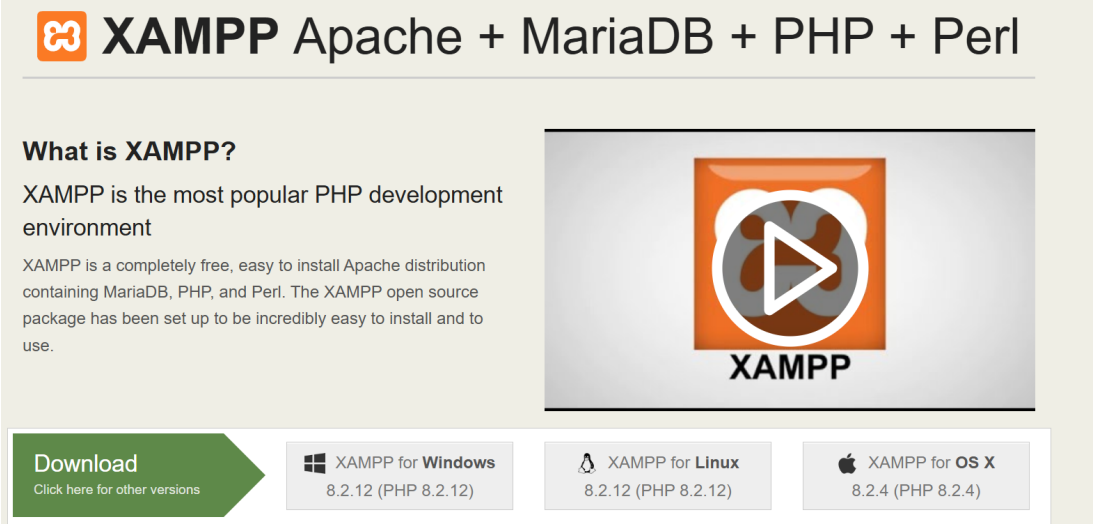


Figure 4. 2. 1. 1 Page to Install XAMPP

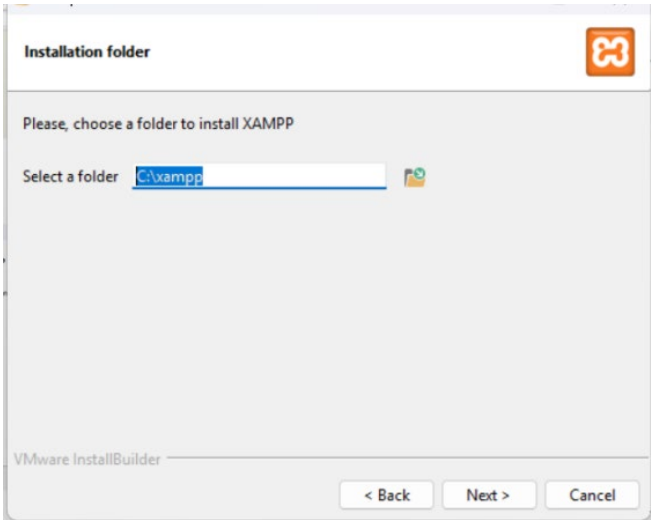


Figure 4. 2. 1. 2 Choose a folder to install.

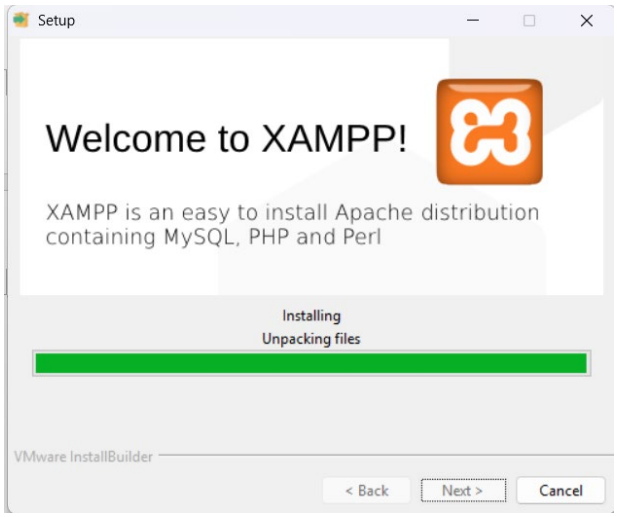


Figure 4. 2. 1. 3 Successfully download.

After successfully download and install the XAMPP, the XAMPP control panel will be opened as Figure 4.2.1.4 and start the services. This project will use Apache and MySQL. Then, create a project folder to “C:/xampp/htdocs/” as figure.

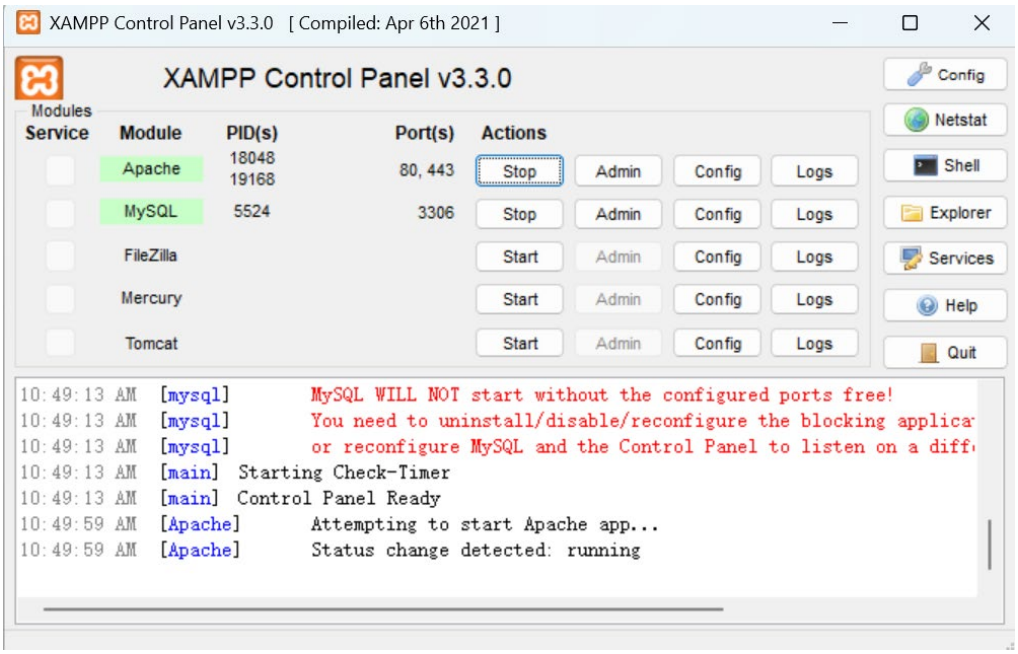


Figure 4. 2. 1. 4 XAMPP Control Panel

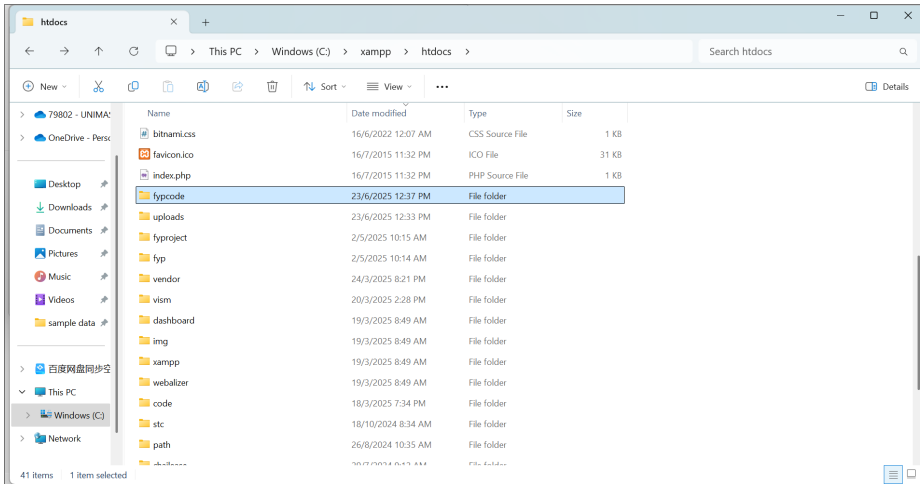


Figure 4. 2. 1. 5 Create project folder.

Once the project set up in the path successfully, users can view the website by entering "http://localhost/fypcode/" in the web browser. Apache will allow the user to access and interact with website.

4.2.2 Navicat

Navicat is a database administration and development tool that simplifies the management of MySQL or MariaDB databases through a graphical interface. It is useful for developers who prefer not to work directly with SQL commands via the command line. In this project, Navicat is used to create and manage databases and tables, insert, update, and delete records in an intuitive way, run SQL queries for advanced data manipulation, backup and restore the database easily.

Steps to Install Navicat:

Download and install Navicat for MySQL from the official website as Figure 4.2.2.1. Open Navicat and click "Connection > MySQL" to create a new connection as Figure 4.2.2.2. For this project, the connection details as Figure 4.2.2.3. The connection name "fypdatabase", host is "localhost", port will be "3306" which is default, username is root and blank password as the project password is not set in XAMPP.

Figure 4. 2. 2. 1 Download Navicat from website

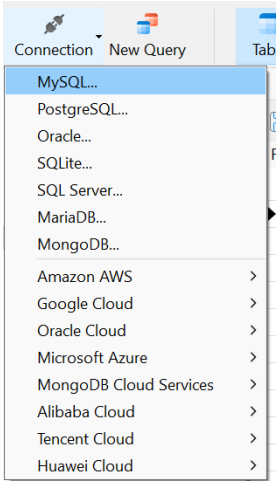


Figure 4. 2. 2. 2 Click “Connection > MySQL” to create new connection

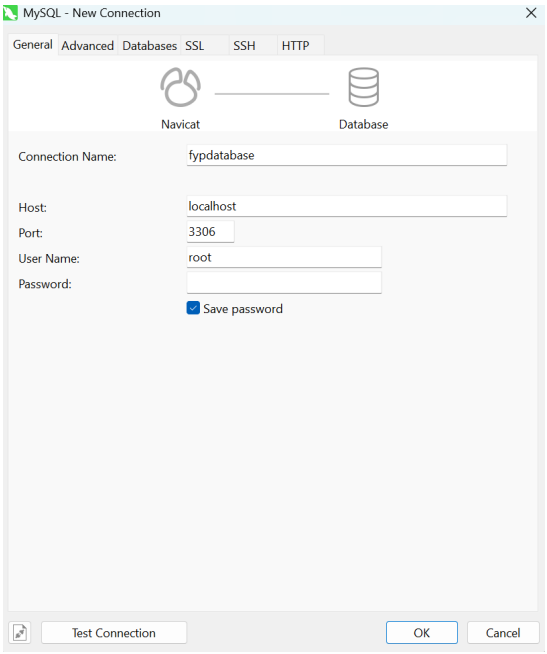


Figure 4. 2. 2. 3 Connection details

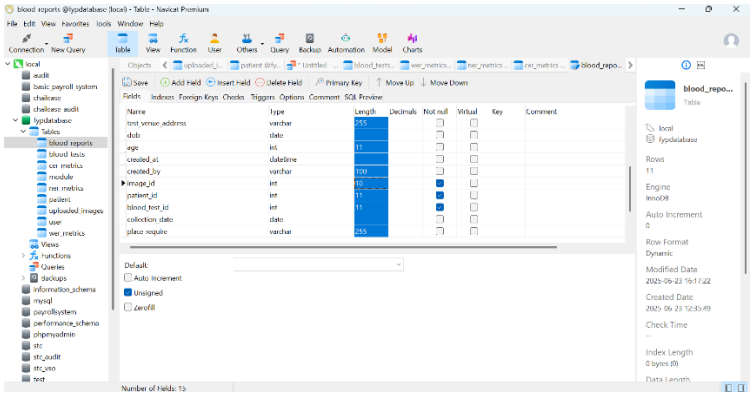


Figure 4. 2. 2. 4 Database of project

The connection is test and successfully created. The tables need for this project are “user”, “patient”, “blood_reports”, “blood_tests”, “module”, “uploaded_images”, “cer_metrics”, “wer_metrics”, “ner_metrics”. The fields will be created according to the data dictionary. The data management is through Navicat’s user-friendly interface as in Figure 4.2.2.4. Navicat works alongside XAMPP’s MySQL server to provide a more efficient and visual approach to database operations during development.

4.2.3 Visual Studio Code

Visual Studio Code (VS Code) is a powerful integrated development environment that is widely used for software development, including web, desktop, and data processing applications. Its clean interface, support for multiple programming languages, and extensive extensions make it ideal for OCR based projects. In this project, Visual Studio Code is used to develop the full functionality of the OCR-based blood test report management system. This includes building the interface for file uploads, integrating the OCR engine, processing extracted data, and implementing features for data verification, patient record management, and reporting. To get started, download Visual Studio Code from the official website: https://code.visualstudio.com/.

Figure 4. 2. 3. 1 Official website for downloading Visual Studio Code.

After installation, launch the application and follow the setup instructions. Once completed, open the project folder in Visual Studio Code, as shown in the Figure 4.2.3.2 to begin coding and managing the entire development process.

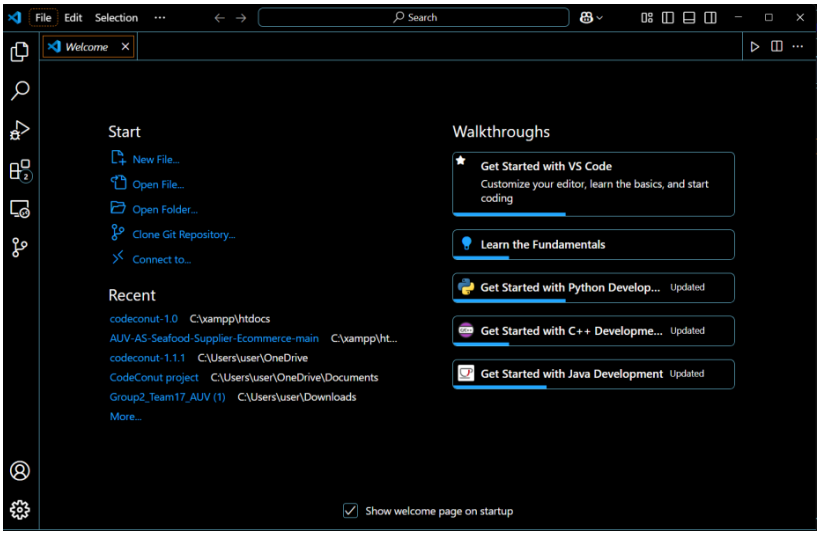


Figure 4. 2. 3. 2 Homepage opened in Visual Studio Code.

4.2.4 Configuration Database Connection

To connect the database for the project, a file “connect.php” is created. This file contains the neccessary configuration settings required to connect the system to the database server. The code shown in Figure 4.2.4.1 demonstrates how the connection is established using PHP.

Figure 4. 2. 4. 1 Written code in connect.php for connection between the project and database.

Figure 4. 2. 4. 2 Included the connect.php file in all PHP files that needs to interact with database.

All PHP files that need to interact with the database should include the connect.php file using its file path as Figure 4.2.4.2. Separating the database connection code into a file like connect.php is recommended, as it avoids repeating the same code in multiple files. This approach improves makes the project easier to maintain and simplifies updates to the database connection settings.

4.3 Workflow and modules

4.3.1 Registration page

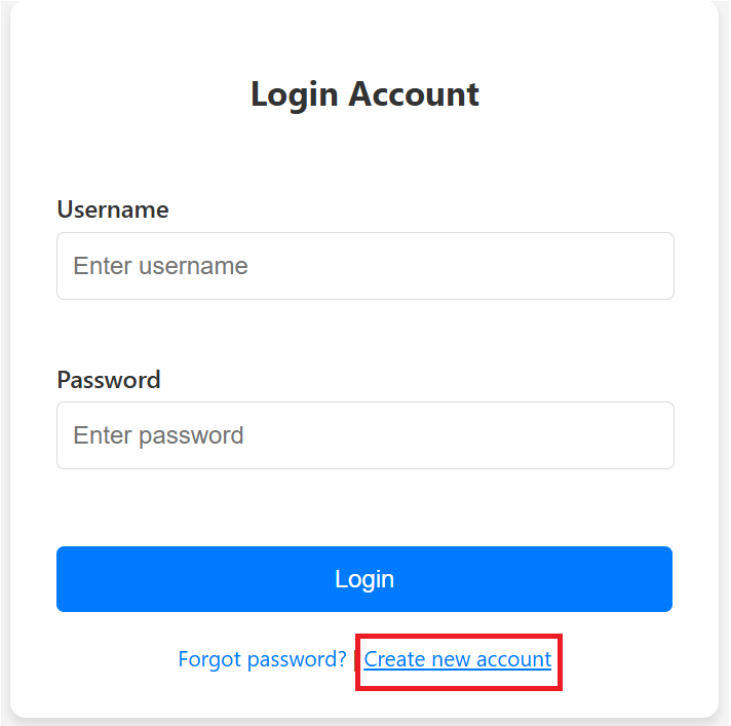


Figure 4. 3. 1. 1 Create new account

By clicking on the “Create new account” as in Figure 4.3.1.1 to register page.

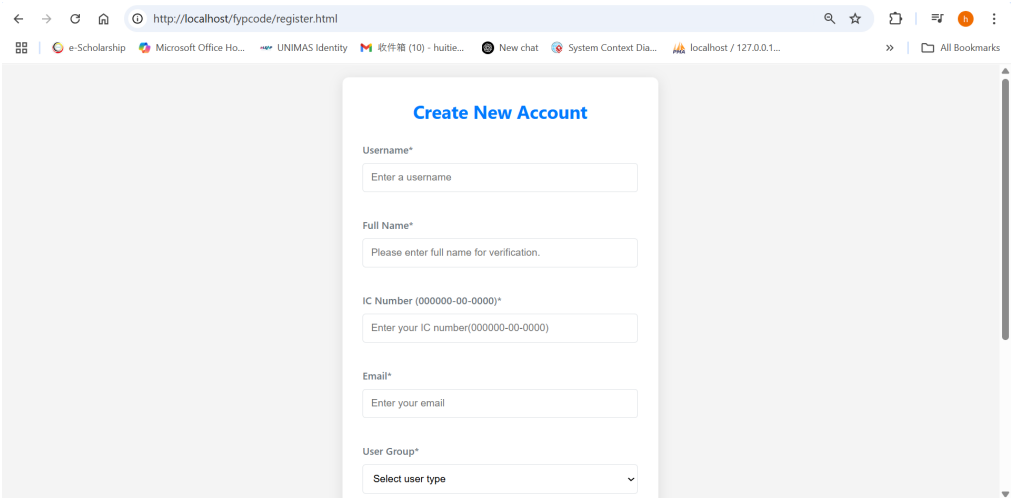


Figure 4. 3. 1. 2 Registration page

Figure 4.3.1.2 shows the registration page. New user must register before accessing the system. The user needs to fill in the username, full name, IC number, email address, and password. User group should be selected by the user when register.

4.3.2 Login/Logout page

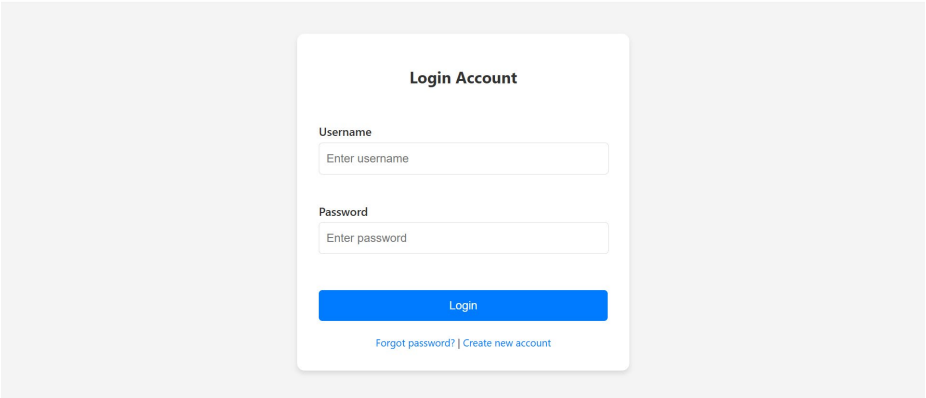


Figure 4. 3. 2. 1 Login Page

Figure 4.3.2.1 shows the Login page for the user. If the user already has an account, user can log in by filling in the registered username and password.

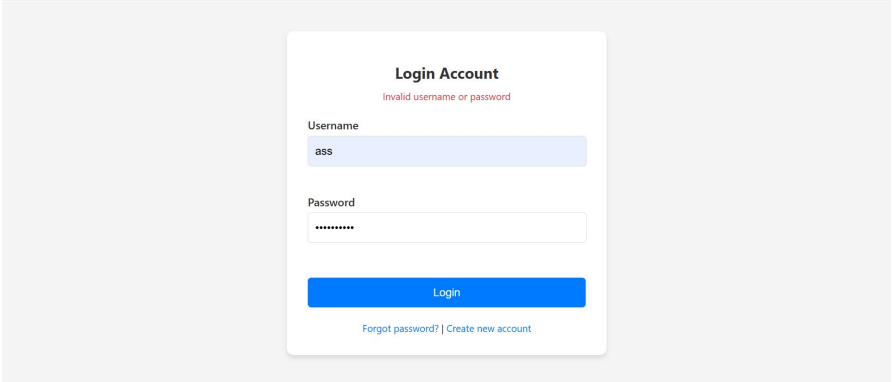


Figure 4. 3. 2. 2 Login page with invalid username and password

Figure 4.3.2.2 shows the login page with invalid username and password. The system will be warning when the user wrong entered for the username and password.

4.3.3 Forgot Password Page

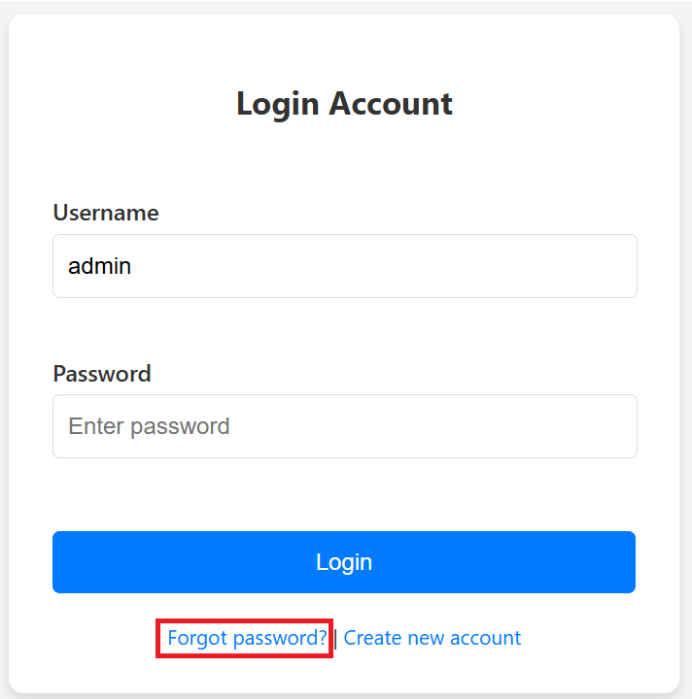


Figure 4. 3. 3. 1 Forgot password

By clicking on the “forgot password”, forgot password page will be shown as Figure 4.3.3.1.

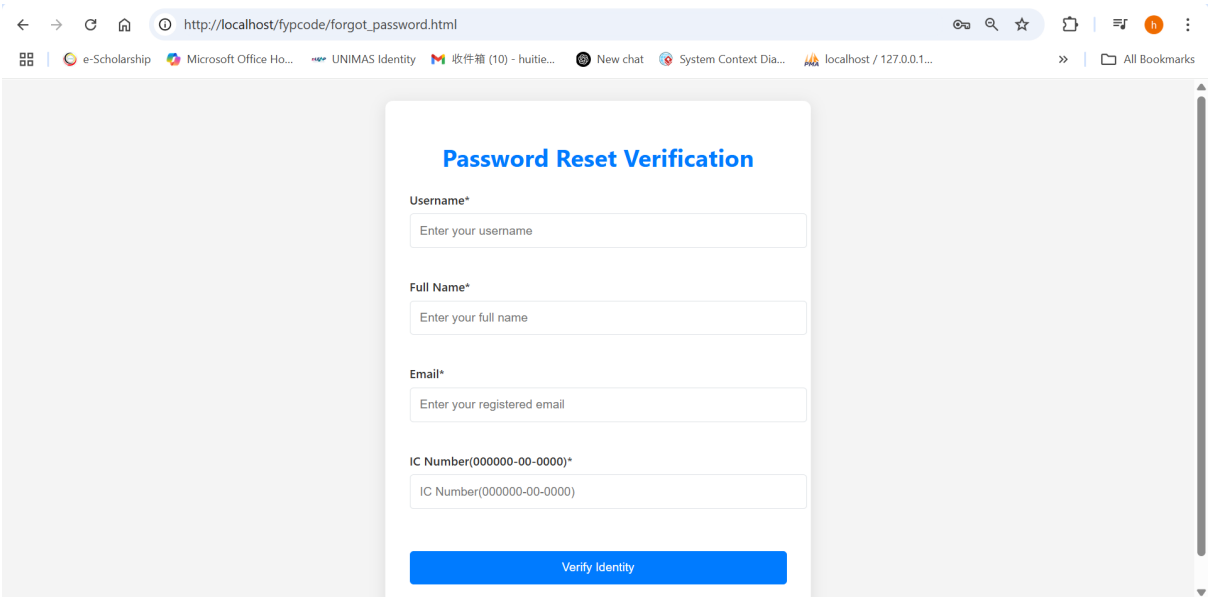


Figure 4. 3. 3. 2 Forgot password page

Figure 4.3.3.2 is forgot password page. User must enter the username, full name, email address, and IC number for verification.

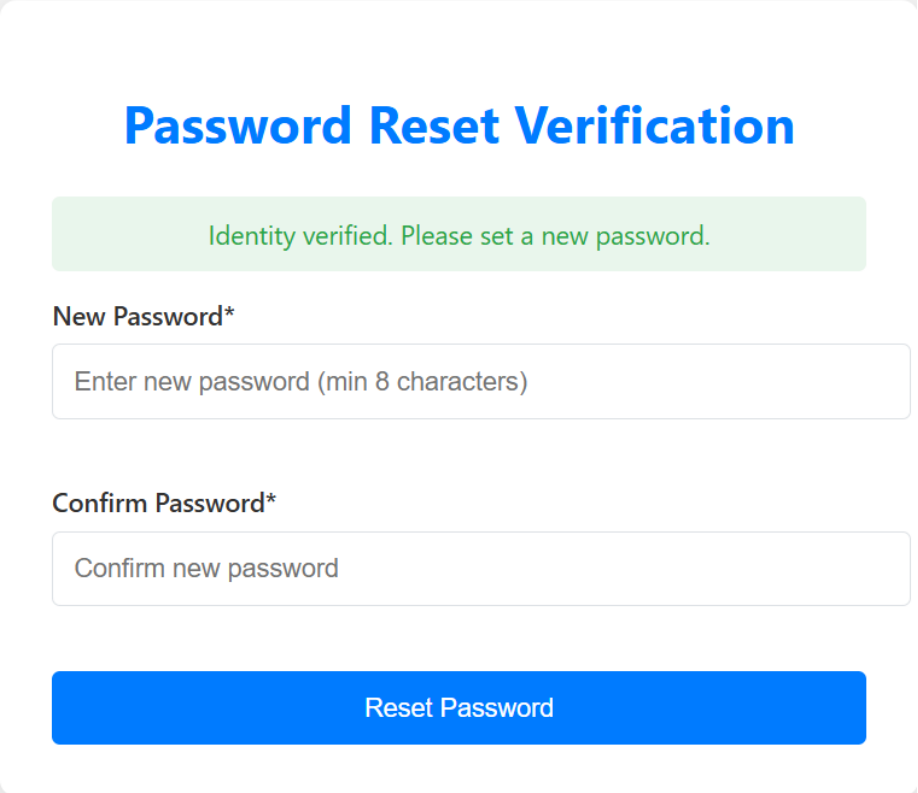


Figure 4. 3. 3. 3 Reset password

After verification, new password can be entered, and password will be reset.

4.3.4 Dashboard

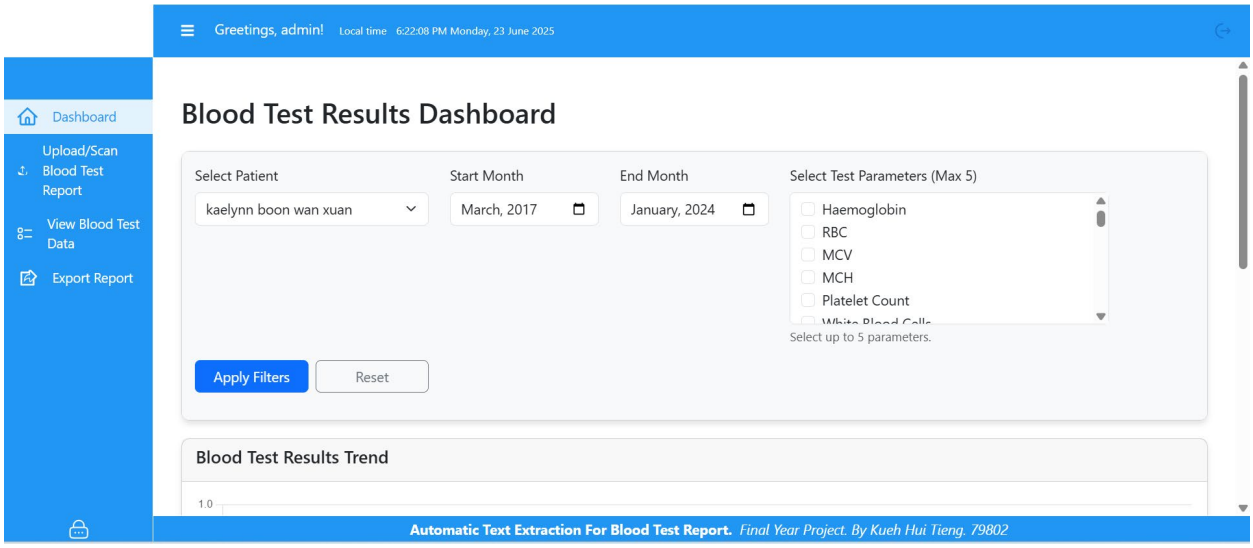


Figure 4. 3. 4. 1 Dashboard page

After user log in, the system will display a dashboard page as Figure 4.3.4.1. The user can select patient name, filter for month and select for test parameters to get the trend of the patients result. The line graph will be shown as Figure 4.3.4.2, and test data will be display using data table as Figure 4.3.4.3.

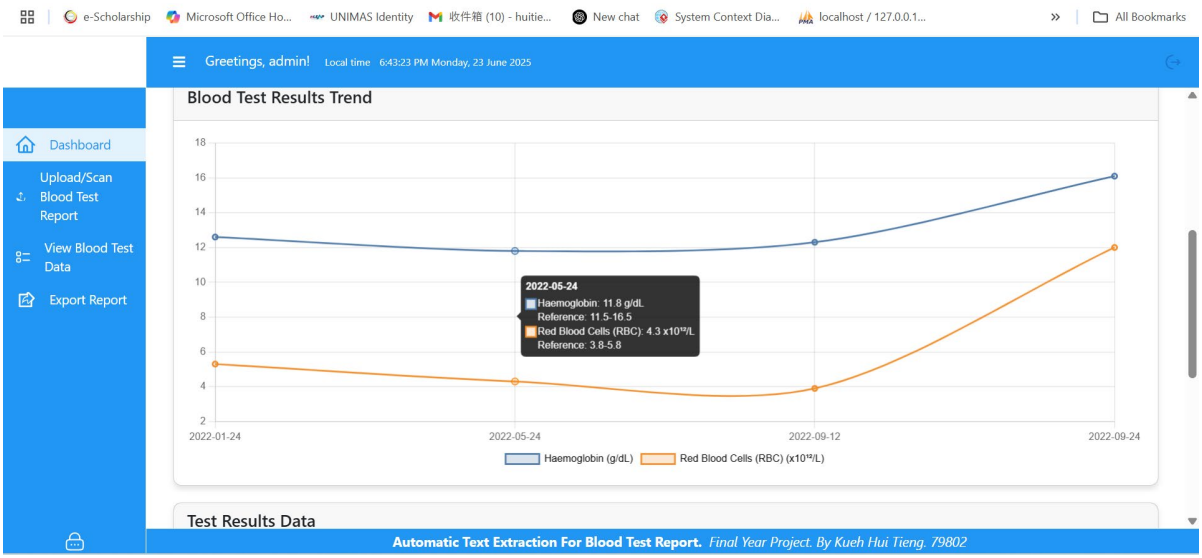


Figure 4. 3. 4. 2 Blood Test Results Trend

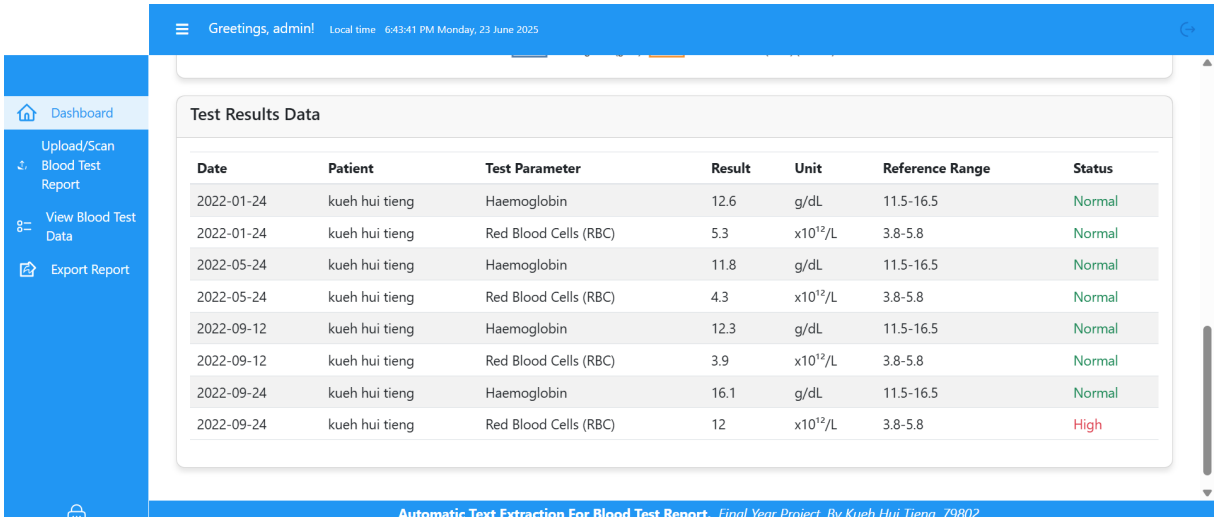


Figure 4. 3. 4. 3 Data table for test results.

4.3.5 Upload/Scan Blood Test Report

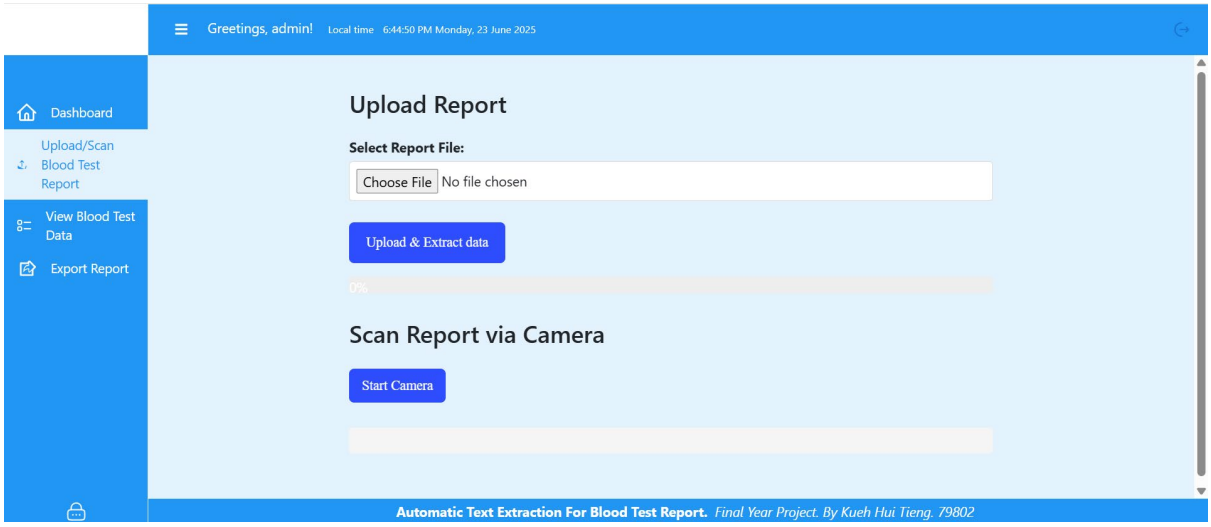


Figure 4. 3. 5. 1 Upload blood report page

Figure is the upload blood report page. The user can select the file or they can use scan function by clicking on start camera button.

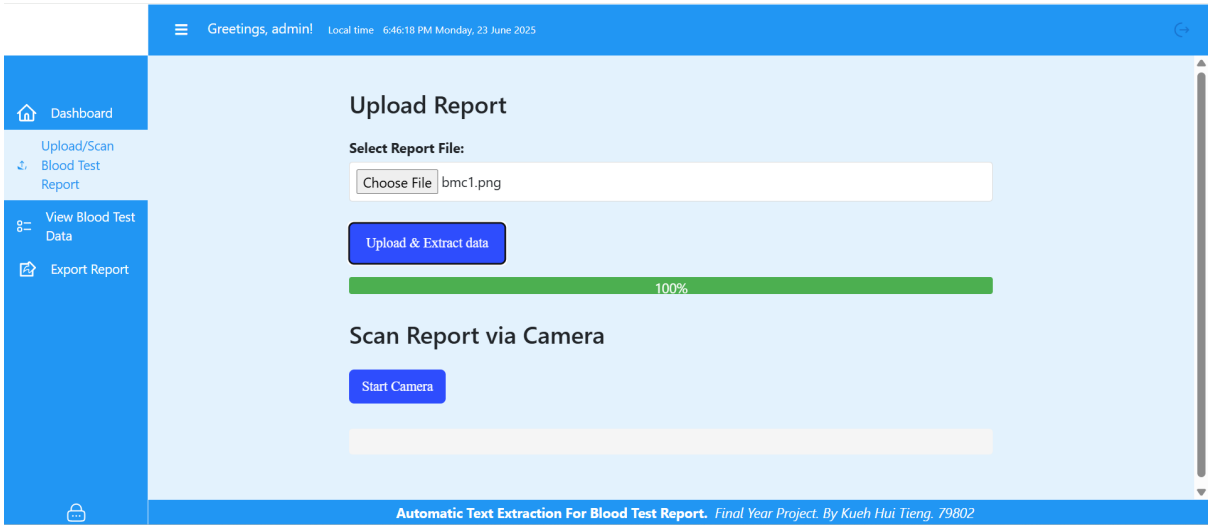


Figure 4. 3. 5. 2 OCR and NLP scanning

OCR scanning when the loading bar is green. NLP will be detected for the data of the reports.

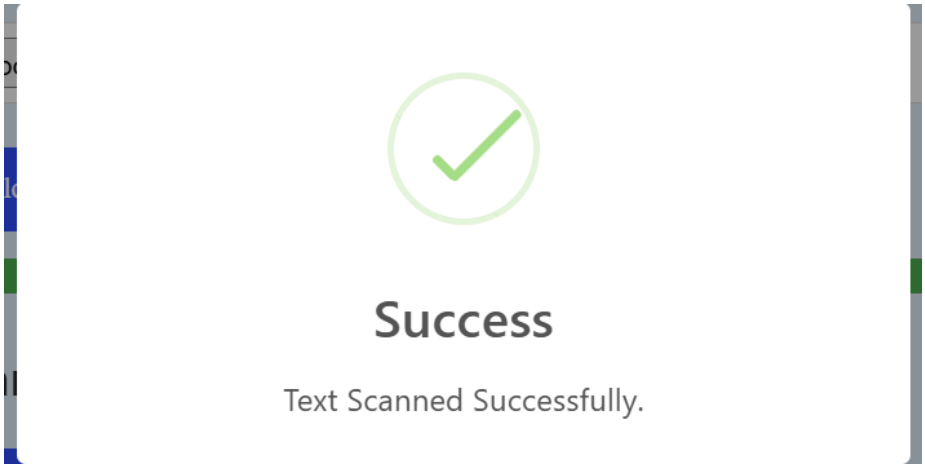


Figure 4. 3. 5. 3 Success scanning

Figure 4.3.5.3 is success scanning. When the text is scanned and detected by the NLP tools (SpaCy). The success scanned sweet alert will be shown.

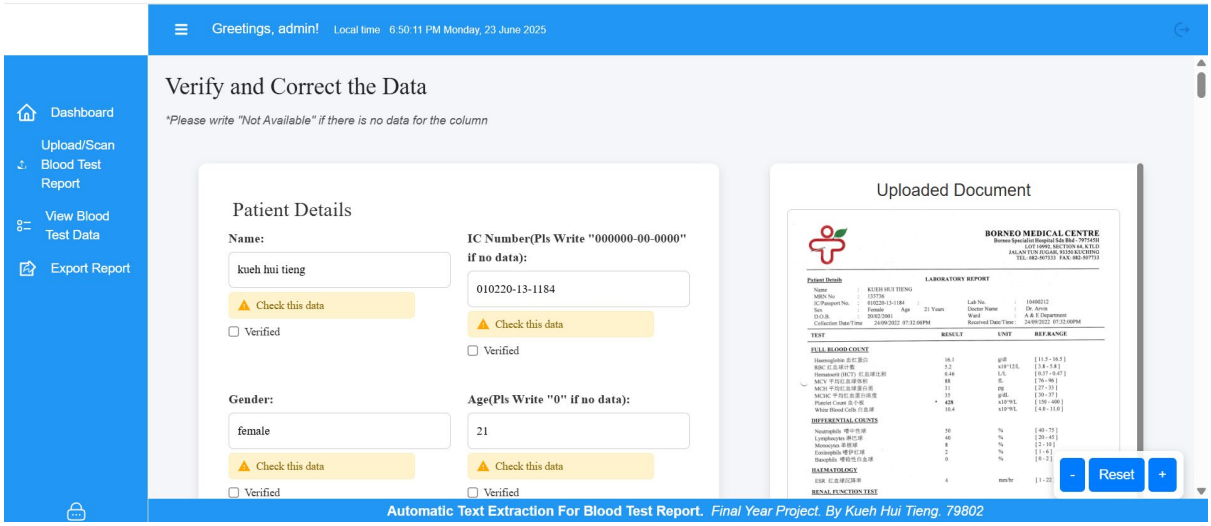


Figure 4. 3. 5. 4 Verification page

Figure 4.3.5.4 shows the verification page. The user can verify the data for the patient details and blood test details. The user needs to click on verify checkbox when they are verifying the data is correct. The user can delete for the test if the test is wrong detected.

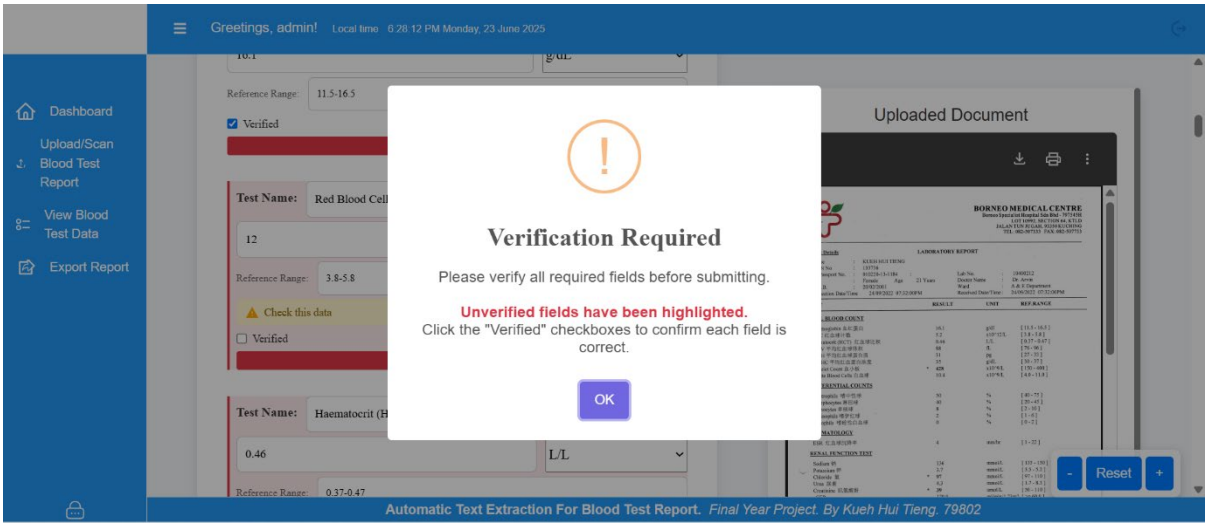


Figure 4. 3. 5. 5 Verificatoin Required Warning

All the data must be checked, if there is no check, verification required warning will be shown. The unverified data will be highlighted so that the user will know where they missed check.

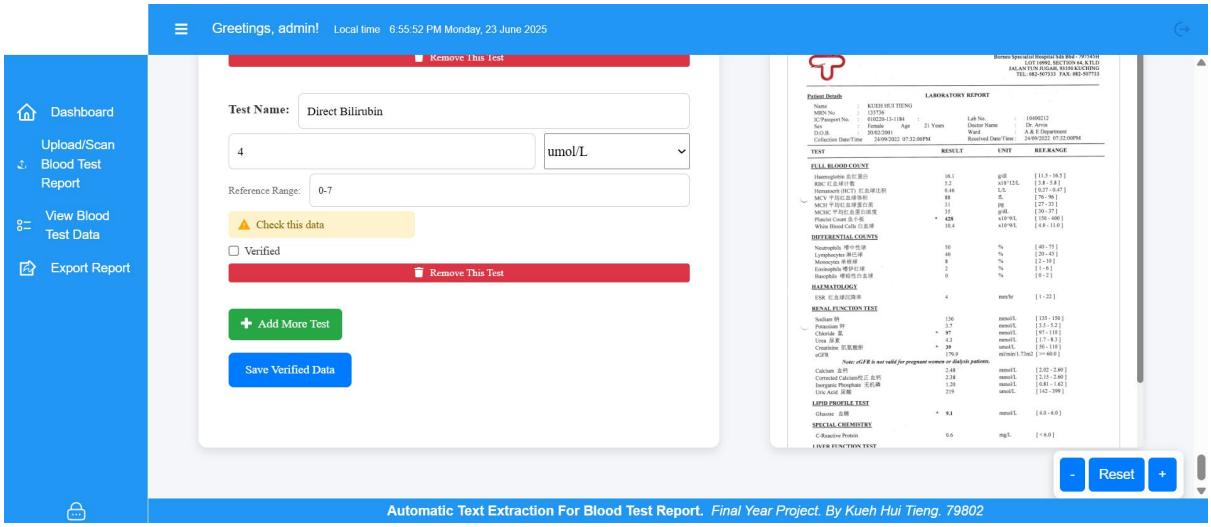


Figure 4. 3. 5. 6 Add more test

Figure shows add more test button. User can click on the add more test button to enter the data manually.

```
-- Upload Form --
DISPLAY a file upload form:
   - INPUT: file (image or PDF), required
   - BUTTON: "Upload & Extract data"

ON form submit:
   PREVENT default form action
   CREATE a new FormData object with form inputs
   SEND AJAX POST request to 'upload_process.php' with the FormData

   ON success:
      PARSE the JSON response
      IF upload is successful:
         CREATE a new hidden form
         ADD hidden inputs: file ID, parsed data, extracted text
         SUBMIT the form to 'main.php?page=verify_data'

-- Scan via Camera --
DISPLAY camera frame:
   - VIDEO element (live preview)
   - CANVAS element (used for image capture, hidden)

DISPLAY buttons:
   - "Start Camera"
   - "Capture" (initially hidden)
   - "Upload Scan" (initially hidden)

ON "Start Camera" button click:
   ACCESS user's camera using getUserMedia
   SET video stream as source of VIDEO element
   START playing the video

ON "Capture" button click:
   DRAW the current video frame onto CANVAS
   CONVERT canvas image to base64 PNG format
   STORE as capturedImage

ON "Upload Scan" button click:
   CONVERT base64 image (capturedImage) to BLOB
   CREATE new FormData object
   APPEND blob to FormData with filename "scan.jpg"
   SEND AJAX POST request to 'upload_process.php' with the FormData

   ON success:
      PARSE response
      IF upload successful:
         REDIRECT or POST to verification page (same as file upload)
```

Figure 4. 3. 5. 7 Pseudocode for upload and scan page, "upload_scan.php"

Figure 4.3.5.7 show the Pseudocode for the upload and scan page, the blood test report will be uploaded and will be passed to “upload_process.php” The code includes the camera function for the user to scan the blood test report. The image will upload into database.

```
IMPORT required libraries and establish database connection
IMPORT Tesseract OCR library
RECEIVE uploaded file from user
IF upload has error OR file size > 5MB:
   RETURN JSON error message and EXIT

INSERT placeholder record into `uploaded_images` table
GET the newly inserted file ID

DETERMINE file extension
DEFINE upload directory path
SET saved file path as "uploads/blood_reports/{file_id}.{ext}"
MOVE uploaded file to that path

IF file is a PDF:
   CONVERT PDF pages to images using external Python script
   STORE resulting image paths in a list
ELSE:
   USE the uploaded image as-is

FOR each image path in the list:
   RUN Tesseract OCR (English) on the image
   APPEND extracted text to an array

JOIN all extracted texts into one string

CALL external Python script to perform NLP parsing (spaCy) on the extracted text
   - PASS the combined text via STDIN
   - RECEIVE parsed result in JSON format

RETURN a JSON response containing:
   - success flag
   - file ID
   - raw extracted text
   - structured parsed data
```

Figure 4. 3. 5. 8 Pseudocode for process data, “upload_process.php”

Figure 4.3.5.8 show the pseudocode to process data. The Tesseract is used for OCR. The NLP and convert PDF will be converting by python. It will convert the .pdf file to .png file as the

tesseract can proceed OCR with picture. The function “callpythonparser” will be used to get the text that extracted by SpaCy in the python format.

```
IMPORT required libraries: sys, os, pdf2image, logging
CONFIGURE logging settings
DEFINE main() function:
   IF command-line arguments are fewer than 3:
      LOG error "Usage: python pdf_to_png_multi.py <input_pdf> <output_pattern>"
      EXIT with error code

   SET input_pdf = 1st argument
   SET output_pattern = 2nd argument

   IF input PDF file does not exist:
      LOG error and EXIT

   TRY:
      LOG start of conversion process

      CONVERT input PDF to list of image objects (one per page)

      IF no images were returned:
         LOG error and EXIT

      INITIALIZE empty list saved_files

      FOR each image and its index (starting from 1):
         REPLACE "%d" in output pattern with current index to form output_path
         SAVE image as PNG to output_path
         ADD output_path to saved_files list
         LOG success message for that page

      PRINT all saved file paths, one per line (for PHP to read)
      EXIT with success code

   EXCEPT any error:
      LOG the error with traceback
      EXIT with error code

IF this script is run directly:
   CALL main()
```

Figure 4. 3. 5. 9 Pseudocode for pdf_to_png.py

Figure 4.3.5.9 is the psuedocode for converting pdf to png file. It is for the tesseract OCR detection as the Tesseract cannot detect for PDF file.

```
IMPORT modules: re, sys, json, spacy, difflib
LOAD spaCy English model (en_core_web_sm)
DEFINE known test names:
   - LIST of standard test names (e.g., "Haemoglobin", "RBC", etc.)
   - CREATE a dictionary KNOWN_TESTS_MAP with lowercase keys for fuzzy matching
---
DEFINE FUNCTION fuzzy_match_test(line):
   CONVERT line to lowercase
   FOR each known test in lowercase:
      IF test name is directly in the line:
         RETURN the properly capitalized test name from KNOWN_TESTS_MAP

   IF no direct match:
      USE difflib to find a close match (cutoff = 0.75)
      RETURN the closest match if found
      OTHERWISE return None
---
DEFINE FUNCTION parse_test_result(line, test_name):
   USE regular expression to extract:
      - numeric result
      - optional unit
      - optional reference range

   IF match found:
      RETURN dictionary with:
         - test_name
         - result
         - unit
         - reference_range
   OTHERWISE:
      RETURN None
---
DEFINE FUNCTION extract_patient_name(text):
   RUN spaCy NLP on the entire text
   SEARCH for the first named entity labeled "PERSON"
   RETURN the person's name if found, else return empty string
---
DEFINE FUNCTION extract_tests(text):
   SPLIT the input text into lines
   INITIALIZE an empty SET called `seen` (to avoid duplicates)
   INITIALIZE an empty LIST `results` to store parsed test info

   FOR each line:
      TRY to fuzzy match a test name
      IF matched and not already seen:
         PARSE the result from the line
         IF successful:
            ADD result to results list
            MARK test as seen
```

```
    RETURN the list of parsed test results
---
IN MAIN:
    READ all input text from standard input
    CALL extract_patient_name() and extract_tests()
    CREATE a dictionary `parsed` with:
        - "name": extracted patient name
        - "tests": list of test result dictionaries
    PRINT parsed data as formatted JSON
```

Figure 4. 3. 5. 10 Pseudocode parsed_blood_report.py

Figure 4.3.5.10 show the pseudocode parsed_blood_report.py. The test will be detected and parsed the data into the file and NLP function to detect the word for tests.

4.3.6 View Blood Test Data

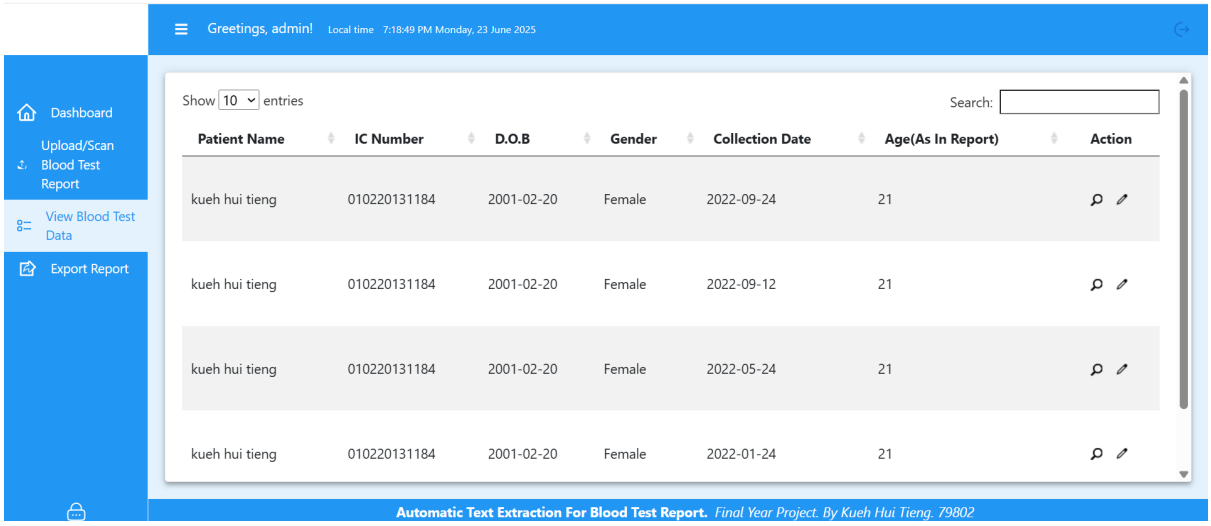


Figure 4. 3. 6. 1 View blood test data page

Figure 4.3.6.1 show the blood test data page for every patient. By clicking on the view and edit button, user can view each patient blood test details.

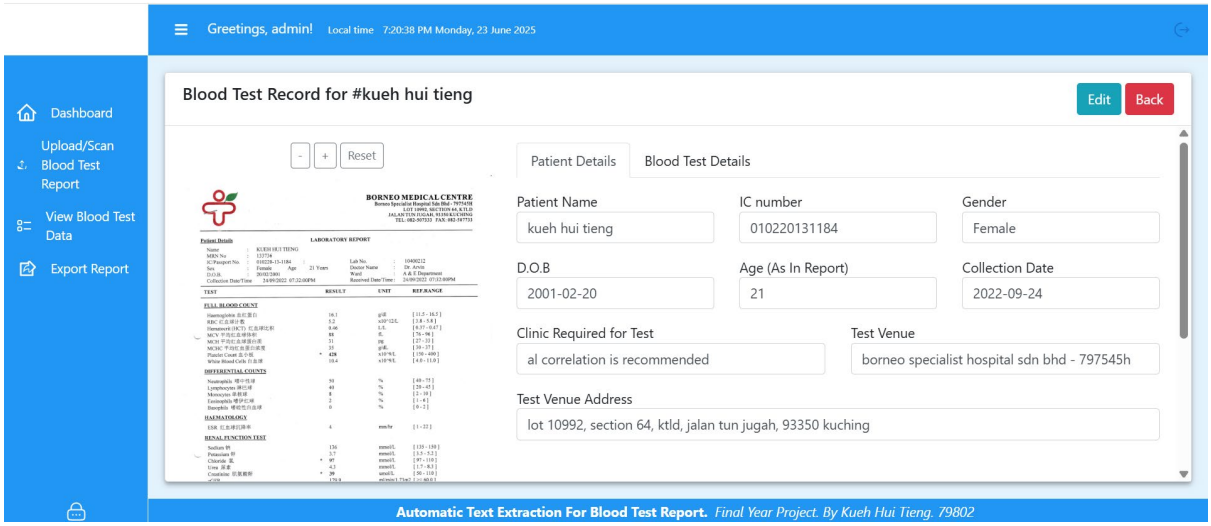


Figure 4. 3. 6. 2 View selected blood test report data page

Figure 4.3.6.2 show the blood test data page for the selected blood test report. The user can view for the data. The data will be displaying a red colour box if the data is above the range. The data will be displaying a orange colour box if the data is lower the range.

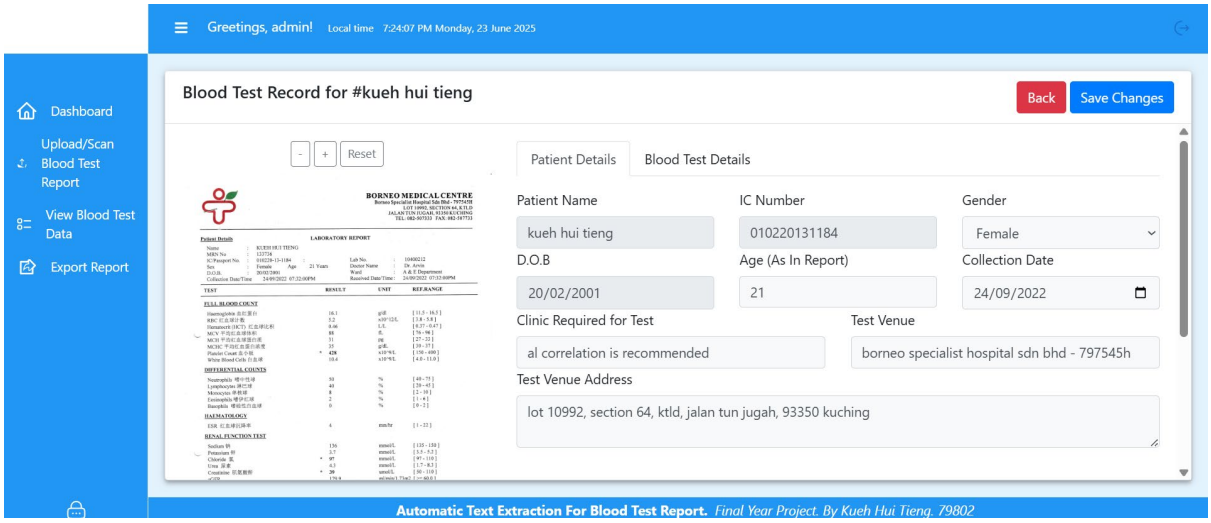


Figure 4. 3. 6. 3 Edit selected blood test report data page

Figure 4.3.6.3 show the edit page for blood test data page for the selected blood test report. The user can edit for the data. The data will be displaying red colour box if the data is above the range. The data will be displaying orange colour box if the data is lower the range. The user can change the unit, as the data and range will be change according to the unit.

4.3.7 Export Blood Test Report

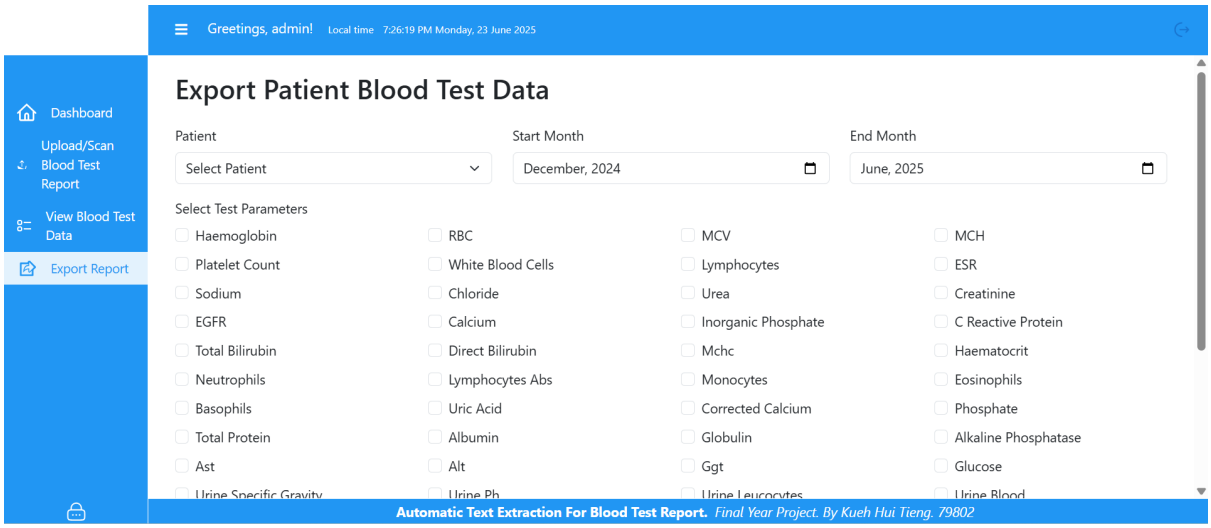


Figure 4. 3. 7. 1 Export patient blood test data page

Figure 4.3.7.1 shows export patient blood test data page. The user can select patient and filter the month. The user can select for test parameters for the report. The report will be in PDF format and EXCEL format.

4.4 Summary

This chapter presented the implementation process of the proposed system, Automatic Text Extraction Using Optical Character Recognition (OCR) for Blood Test Report Management. The project adopts the Agile methodology to iteratively develop both front-end and back-end components, ensuring a functional, user-friendly, and interactive web application. The installation and configuration of essential tools such as XAMPP, Navicat, and Visual Studio Code were described to set up the local development environment. XAMPP is used to host the system and manage the database, while Navicat simplifies database operations through a graphical interface. Visual Studio Code is the main used for coding and managing the overall project.

This chapter also detailed the main modules and workflows, including user registration, login/logout, forgot password, dashboard for viewing test trends, and upload/scan functionality

using OCR and NLP. It covers how scanned data is verified, edited, and stored, and explains how reports can be exported in PDF and Excel formats. The integration of Tesseract OCR and SpaCy NLP demonstrates how text from blood test reports is extracted, processed, and presented in a meaningful and structured way.

CHAPTER 5 Testing

5.1 Introduction

This chapter outlines the testing of the Automatic Text Extraction Using Optical Character Recognition (OCR) for Blood Test Report Management system. The purpose of this phase is to evaluate the accuracy, reliability, and usability of the proposed web application through three key types of testing which are functional testing, integration testing, and evaluation metrics.

Unit Testing is conducted to ensure that all major features of the system such as file uploading, OCR processing, data verification, and report exporting are all operate correctly. It verifies that each feature performs as expected and meets the system requirements.

Integration Testing focuses on the interaction between individual modules. This ensures that data flows smoothly from one part of the system to another, such as from OCR output to NLP parsing, and from parsed data to database storage and user interface.

Evaluation Metrics are used to assess the accuracy and effectiveness of the OCR and NLP components. Specific metrics such as Character Error Rate (CER), Word Error Rate (WER), and Named Entity Recognition (NER) scores which is Precision, Recall, and F1-Score are calculated to determine how well the system extracts and recognizes key information from blood test reports.

5.2 Unit Testing

Unit testing is carried out to test and evaluate each module in the proposed system. This testing is to ensure that all modules are run smoothly. It also helps to identify and resolve issues early before moving on to subsequent testing phases such as integration testing. If the integration testing is conducted directly without unit testing, critical bugs may occur, making it more difficult to identify, resulting in wasted time and effort in debugging, ultimately delaying the entire testing process.

Module: Registration		**Tested Date:** 21 May 2025		
Title: Register new account		**Pre-condition(s):** The user has no account. **Post-condition(s):** Redirect to login page.		
Test Case ID	**Description**	**Test Step(s)**	**Expected Output**	**Status**
TC1_1	User register with valid data.	Input valid username, email and password. **Sample Data:** Username: Ali Full name: Ali Smith Email: alis@gmail.com IC Number: 021211-12-2231 Usergroup: Doctor Password: admin1234	Success message: *Registration successful* and redirect to login page.	PASS
TC1_2	User register with existing/registered username/email.	Input the existing or registered email. **Sample Data:** Username: Ali Full name: Ali Smith Email: alis@gmail.com IC Number: 021211-12-2231 Usergroup: Doctor Password: admin1234	Error message: *Email already exists.*	PASS
TC1_3	Invalid email format	Input wrong email format. **Sample Data:** alis@gmail	Error message: *Invalid email format.*	PASS
TC1_4	Invalid IC format	Input wrong email format. **Sample Data:** 121212121212	Error message: *IC format must be like 000000-00-0000*	PASS
TC1_5	Unselect user group	User Group did not select.	Error message: Please select a user group	PASS
TC1_6	Password confirmation mismatch.	Input a different password and confirm password. **Sample Data:** Password: admin123 Confirm Password: Admin1234	Error message: *Password do not match.*	PASS

Table 5. 2. 1 Registration Testing Table

Module: Login		**Tested Date:** 21 May 2025		
Title: User login.		**Pre-condition(s):** The user has successfully registered an account. **Post-condition(s):** Redirect to dashboard page.		
Test Case ID	**Description**	**Test Step(s)**	**Expected Output**	**Status**
TC2_1	Successfully login for user with valid credentials.	Input correct username and correct password. **Sample Data:** Username: admin Password: admin1234	Redirect the user to homepage.	PASS
TC2_2	Login failed for user with incorrect username.	Input invalid username and correct password. **Sample Data:** Username: adminqq Password: admin1234	Error message: *Invalid username.*	PASS
TC2_3	Login failed for user with incorrect password.	Input correct username and invalid password. **Sample Data:** Username: admin Password: admin123444	Error message: *Invalid password.*	PASS

Table 5. 2. 2 Login Testing Table

Module: Forgot Password		**Tested Date:** 23 May 2025		
Title: User forgot password.		**Pre-condition(s):** The user has an account. **Post-condition(s):** Reset password.		
Test Case ID	**Description**	**Test Step(s)**	**Expected Output**	**Status**
TC3_1	Submit all valid data that have previously registered.	Input Username, Full Name, Email, and IC Number. **Sample Data:** Username: ali Full Name: Ali Bin Baka Email: ali@gmail.com IC Number: 121212-12-2321	Data verified, to reset password.	PASS
TC3_2	Submitting an invalid username.	Input invalid username. **Sample Data:** Username: aliwe	Error message: *Invalid Username!*	PASS
TC3_3	Submitting an invalid full name.	Input invalid full name. **Sample Data:** Full name: ali Bin	Error message: *Invalid Full name!*	PASS
TC3_4	Submitting an invalid email.	Input invalid email. **Sample Data:**	Error message: *Invalid Email!*	PASS

		Username: ali@email.my		
TC3_5	Submitting an invalid IC number.	Input invalid username. **Sample Data:** Username: 1233324242	Error message: *Invalid IC number!*	PASS

Table 5. 2. 3 Forgot Password Testing Table

Module: Reset Password		**Tested Date:** 23 May 2025		
Title: User reset password.		**Pre-condition(s):** The user has successfully verified data. **Post-condition(s):** Successfully reset the password and redirect to the login page.		
Test Case ID	**Description**	**Test Step(s)**	**Expected Output**	**Status**
TC4_1	Successfully reset new password.	Input valid password and confirm password. **Sample Data:** Password: admin123 Confirm Password: admin123	Redirect to the login page.	PASS
TC4_2	New password confirmation mismatch.	Input a different new password and confirm password. **Sample Data:** Password: admin123 Confirm Password: admin1234	Error message: *Passwords do not match.*	PASS

Table 5. 2. 4 Reset Password Testing Table

Module: Dashboard		**Tested Date:** 21 May 2025		
Title: View Patient Test Result Trends.		**Pre-condition:** User is logged in **Post-condition:** Blood test results are displayed in a line graph and data table		
Test Case ID	**Description**	**Test Step(s)**	**Expected Output**	**Status**
TC5_1	View blood test data trend	Select patient, filter month, select test parameters	Display line graph and data table with results	PASS
TC5_2	No test parameter selected	Select patient and date only, leave test unchecked	No data display	PASS
TC5_3	Empty month filter	Empty month selected	Data display for all	PASS

Table 5. 2. 5 Dashboard Testing Table

Module: Upload/Scan Blood Test Report		**Tested Date:** 21 May 2025		
Title: Upload and Scan Blood Test Report		**Pre-condition:** User is logged in **Post-condition:** File is uploaded and data extracted via OCR and NLP		
Test Case ID	**Description**	**Test Step(s)**	**Expected Output**	**Status**
TC6_1	Upload valid blood report file	Upload the file with valid format	File uploaded and scanned successfully	PASS
TC6_2	Start camera and scan report	Click "Start Camera", scan visible report	Text extracted and success message shown	PASS
TC6_3	Upload unsupported file format	Upload file type other than PDF or image	Error message: *Invalid file format.*	PASS

Table 5. 2. 6 Upload/Scan Blood Test Report Testing Table

Module: Data Verification Page		**Tested Date:** 21 May 2025		
Title: Verify Extracted Data		**Pre-condition:** OCR and NLP data extraction completed **Post-condition:** Data verified and saved		
Test Case ID	**Description**	**Test Step(s)**	**Expected Output**	**Status**
TC7_1	Verify all extracted data	Check all verification boxes, click "Verify"	Data saved and success message shown	PASS
TC7_2	Unverified data exists	Leave at least one box unchecked, click "Verify"	Warning: *Verification required*; unverified data highlighted.	PASS
TC7_3	Delete incorrect test	Click delete on wrongly detected test	Test removed from the list	PASS
TC7_4	Add new test manually	Click "Add More Test", fill in fields	New test added successfully	PASS
TC7_5	Invalid format for IC	Invalid IC format **Sample Data:** IC Number: Not Available	Data cannot be save	PASS

Table 5. 2. 7 Data Verification Testing Table

Module: View/Edit Blood Test Data		**Tested Date:** 21 May 2025		
Title: View and Edit Blood Test Report		**Pre-condition:** Test reports exist for selected patient **Post-condition:** Display and update test data		
Test Case ID	**Description**	**Test Step(s)**	**Expected Output**	**Status**
TC8_1	View patient report list	View patient list in data table	Display all test reports for all patients	PASS
TC8_2	Searching function	Type in the data in the search bar to search for unique patient	Search bar function.	PASS
TC8_3	Open specific report	Click "Edit/View" on one report	Report data displayed correctly	PASS
TC8_4	Highlight abnormal values	Load report with values out of range	Values in red (above) or orange (below)	PASS
TC8_5	Edit test result or unit	Modify result or select different unit	Value updated with converted range and data	PASS

Table 5. 2. 8 View and Edit Blood Test Data Testing Table

Module: Export Blood Test Report		**Tested Date:** 21 May 2025		
Title: Export Patient Report		**Pre-condition:** Test data available for selected patient **Post-condition:** Export file is downloaded in PDF/Excel format		
Test Case ID	**Description**	**Test Step(s)**	**Expected Output**	**Status**
TC9_1	Export with filters applied	Select patient, filter month, choose tests	Downloadable PDF/Excel file generated	PASS
TC9_2	No data for selected filter	Apply filter with no test data	No data available for export file	PASS
TC9_3	Select only PDF or Excel	Choose export type and click export	Correct file format downloaded	PASS
TC9_4	Data colouring in export	Export report with out-of-range values	Colours applied: red (above), orange (below), purple (null), blue (empty)	PASS

Table 5. 2. 9 Export Report Testing Table

5.3 Integration Testing

After the unit tests were successfully completed, integration testing was carried out to verify that different modules of the system work seamlessly together. This testing ensures that data flows correctly between components

Module: Upload, OCR/NLP, Verification		**Tested Date:** 23 May 2025		
Title: Integration of Upload, OCR, and Verification Modules		**Pre-condition:** User is logged in **Post-condition:** Uploaded file is processed and displayed for verification		
Test Case ID	**Description**	**Test Step(s)**	**Expected Output**	**Status**
INT1_1	Upload valid file and process	Upload PDF, pass to upload_process.php	OCR processes image; NLP extracts text; redirect to verification page	PASS
INT1_2	Missing extracted data from OCR	Upload low-quality image	System shows partial data; verification screen highlights missing fields	PASS
INT1_3	Python fails to respond	Simulate failure in callPythonParser()	Error message shown; upload blocked from proceeding	PASS

Table 5. 3. 1 Integration Table for Upload**,** OCR/NLP**,** Verification

Module: Verification, Save to Database		**Tested Date:** 23 May 2025		
Title: Integration of Verification and Data Saving		**Pre-condition:** OCR/NLP extraction is completed **Post-condition:** Verified data is saved into the database		
Test Case ID	**Description**	**Test Step(s)**	**Expected Output**	**Status**
INT2_1	All data verified	Check all boxes, click submit	Data stored in blood_reports and blood_tests tables	PASS
INT2_2	Data partially verified	Leave some fields unverified	Warning message: Verification required	PASS

INT2_3	Add custom test and save	Add new test manually, verify and save	Custom test saved along with extracted tests	PASS

Table 5. 3. 2 Integration Table for Verification**,** Save to Database

Module: Database, Dashboard, View report		**Tested Date:** 30 May 2025		
Title: Integration of Database and Display Modules		**Pre-condition:** Patient blood test data exists in database **Post-condition:** Data is fetched and displayed correctly		
Test Case ID	**Description**	**Test Step(s)**	**Expected Output**	**Status**
INT3_1	Load dashboard after upload	Login, go to dashboard, select patient	Line graph and table display with uploaded data	PASS
INT3_2	View single report	Click view on selected test report	Report data loaded from database and displayed	PASS
INT3_3	Edit report and view changes	Edit unit/value in report and save	Dashboard updates with new values on reload	PASS

Table 5. 3. 3 Integration Table for Database, Dashboard, View report

Module: Database, Export to PDF/Excel format		**Tested Date:** 02 June 2025		
Title: Integration of Database and Export Functionality		**Pre-condition:** Verified test data exists in the database **Post-condition:** PDF/Excel file exported with latest data		
Test Case ID	**Description**	**Test Step(s)**	**Expected Output**	**Status**
INT4_1	Export with valid filters	Select patient, date, test, click export	File downloaded with selected test data	PASS
INT4_2	Export with missing data	Export report with some null values	Values shown as purple/blue in export format	PASS

Table 5. 3. 4 Integration Table for Database, Export to PDF/Excel format

5.4 User Acceptance Testing

User Acceptance Testing (UAT) is the final phase of system testing where real users test the system to ensure it meets their needs, expectations, and business requirements before it is

launched or deployed. It is to verify that the system works as intended in a real-world environment, identify usability issues, bugs, or missing features, getting feedback from actual users, and ensure the software is ready for release. The User Acceptance Testing (UAT) is filled by 30 respondents and the question for the questionnaire is in Appendix B.

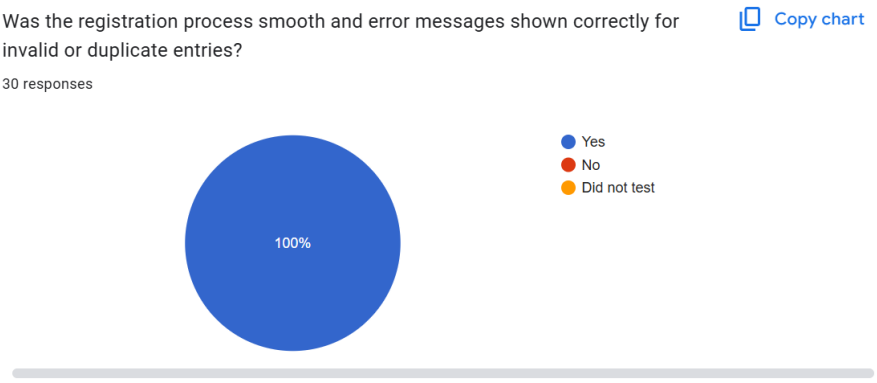


Figure 5. 4. 1 Pie Chart of registration process smoothness

Figure 5. 4. 1 show the pie chart of registration process smoothness. 100% of the respondents, which is 30 respondents think that the registration process is very smooth and error message shown correctly for invalid and duplicate errors.

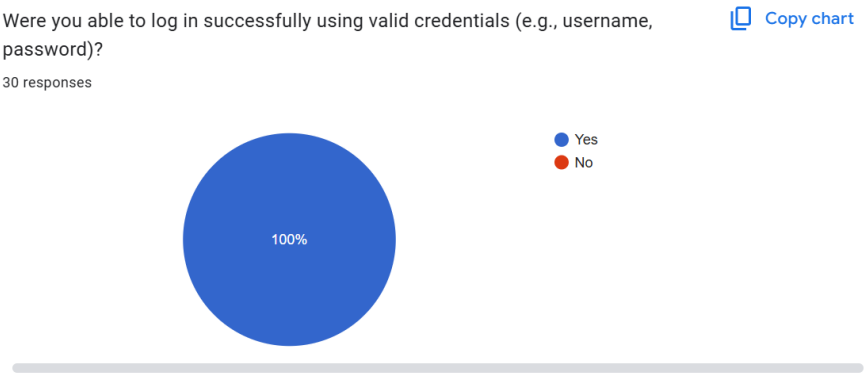


Figure 5. 4. 2 Pie Chart for successful log in.

Figure 5. 4. 2 show the pie chart of successful log in. 100% of the respondents, which is 30 respondents think that they are able to log in to the system successfully using valid credentials.

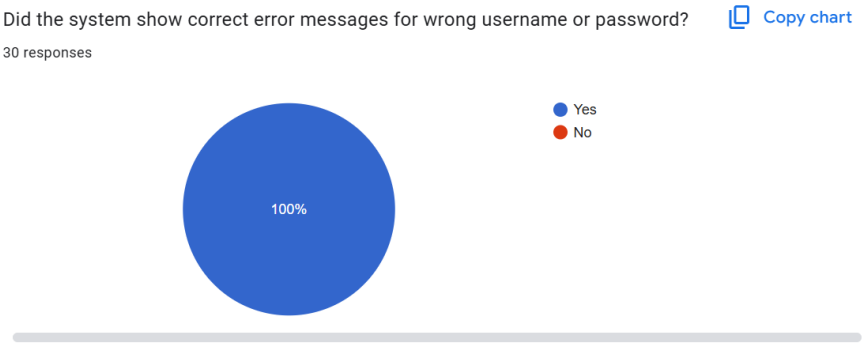


Figure 5. 4. 3 Pie Chart for correct error message.

Figure 5. 4. 3 show the pie chart for correct error message. 100% of the respondents, which is 30 respondents think that the system can show the correct error messages for wrong username and password.

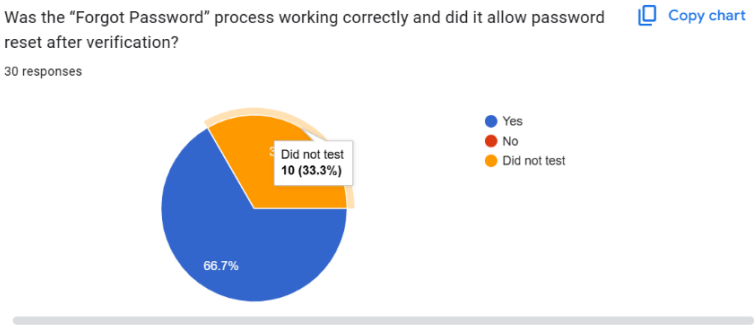


Figure 5. 4. 4 Pie Chart for testing forgot password function.

Figure 5. 4. 4 show the pie chart for forgot password function. 66.67% of the respondents, which is 20 respondents think that the forgot password function is working correctly while 33.3% of the respondents which is 10 respondents are not testing for the forgot password function.

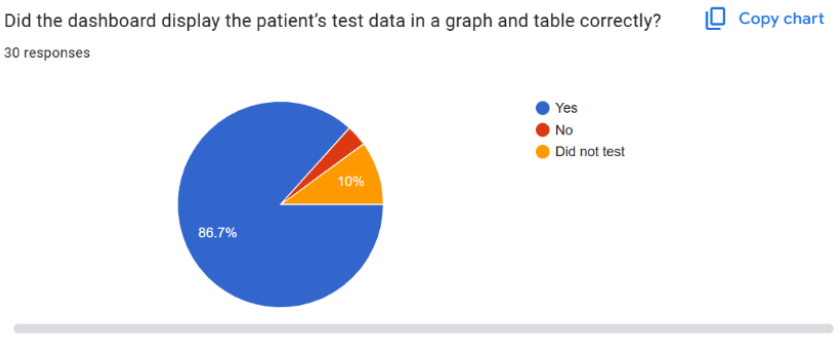


Figure 5. 4. 5 Pie Chart for testing dashboard display function.

Figure 5. 4. 5 show the pie chart for dashboard function. 86.7% of the respondents, which is 26 respondents think that the dashboard displays the patient's test data in a graph and table correctly. 3.3% of the respondents which is 1 respondent think that the function is not well and 10.0% of the respondents which is 3 respondents are not testing for the dashboard display function.

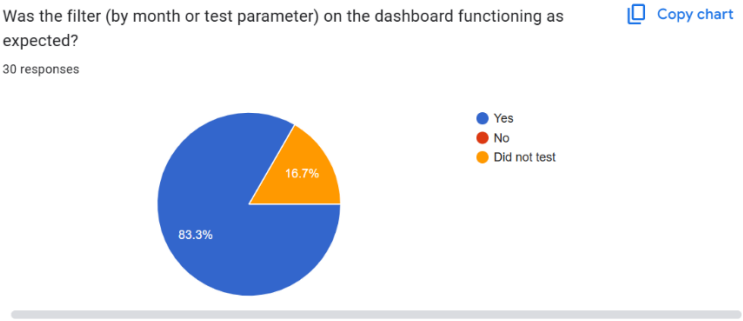


Figure 5. 4. 6 Pie Chart for testing dashboard filter function.

Figure 5. 4. 6 show the pie chart for dashboard filter function. 83.3% of the respondents, which is 25 respondents think that the dashboard filter is well function and display correctly. 16.7% of the respondents which is 5 respondents are not testing for the dashboard filter function.

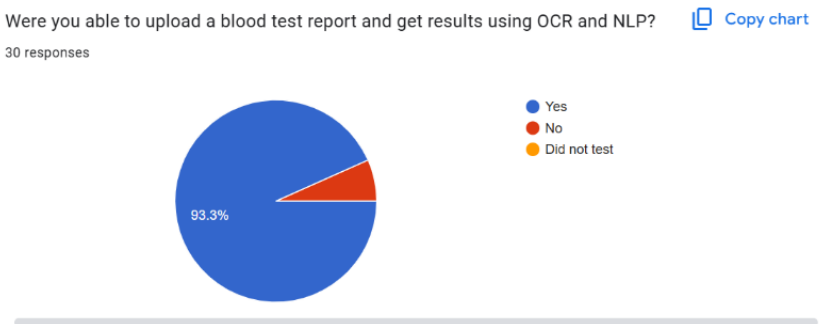


Figure 5. 4. 7 Pie Chart for upload blood test report function.

Figure 5. 4. 7 show the pie chart for upload blood test report function. 93.3% of the respondents, which is 28 respondents think that they able to use the upload function. 6.7% of the respondents which is 2 respondents cannot upload blood test report and get the results using OCR and NLP.

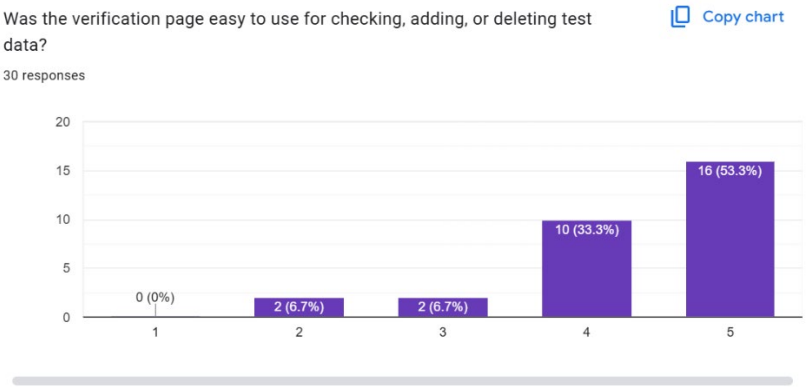


Figure 5. 4. 8 Graph for easy of verification page to use.

Figure 5. 4. 8 show the graph for easy of verification page to use. 53.3% of the respondents, which is 16 respondents think it is very easy to use for checking, adding or deleting test data. 33.3% of the respondents, which is 10 respondents thinks that it is easy to use. 6.7% of the respondents which is 2 respondents think that it is neutral to use. 6.7% of the respondents which is 2 respondents think that it is difficult to use.

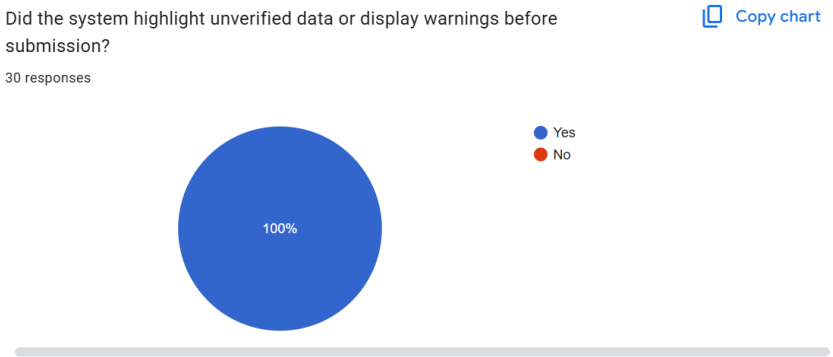


Figure 5. 4. 9 Pie Chart for system highlight unverified data.

Figure 5. 4. 9 show the pie chart for system highlight unverified data. 100% of the respondents, which is 30 respondents think that the system highlights unverified data and display warning before submission.

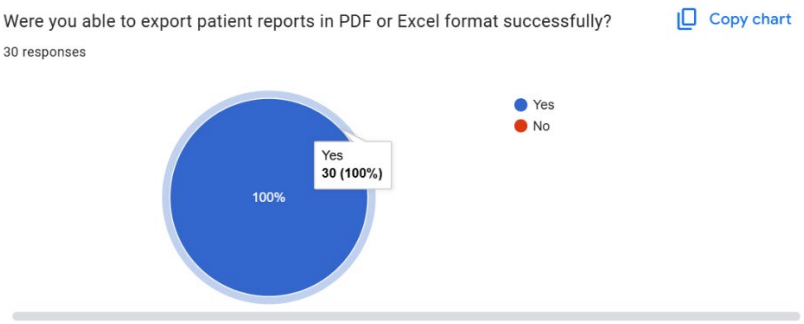


Figure 5. 4. 10 Pie Chart for export function in format successfully.

Figure 5. 4. 10 show the pie chart for export function in format successfully. 100% of the respondents, which is 30 respondents think that the system export patient report in PDF or Excel format successfully.

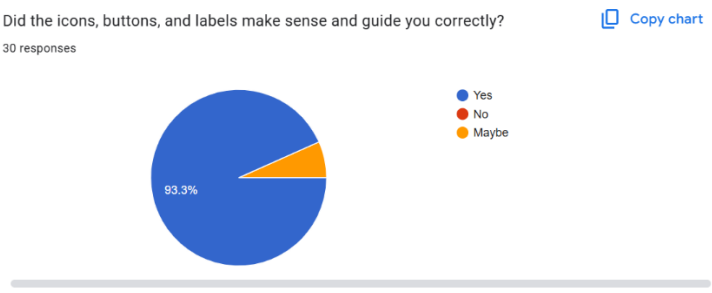


Figure 5. 4. 11 Pie Chart for icon, buttons and label guide correctly.

Figure 5. 4. 11 show the pie chart for icon, buttons and label guide correctly. 93.3% of the respondents, which is 28 respondents think that the icon, buttons and label guide correctly. 6.7% of the respondents which is 2 respondents did not sure that the icon, buttons and label guide are all correctly function.

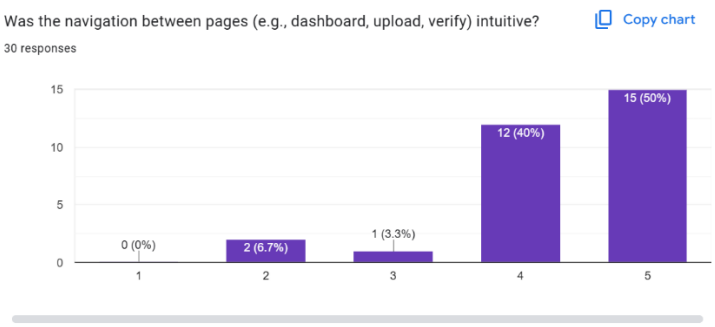


Figure 5. 4. 12 Graph for navigation between pages to use.

Figure 5. 4. 12 show the graph for navigation between pages to use. 50.0% of the respondents, which is 15 respondents think it is very easy to use for navigation between pages. 40.0% of the respondents, which is 12 respondents thinks that it is easy to use. 3.3% of the respondents which is 1 respondent think that it is neutral to use. 6.7% of the respondents which is 2 respondents think that it is difficult to use.

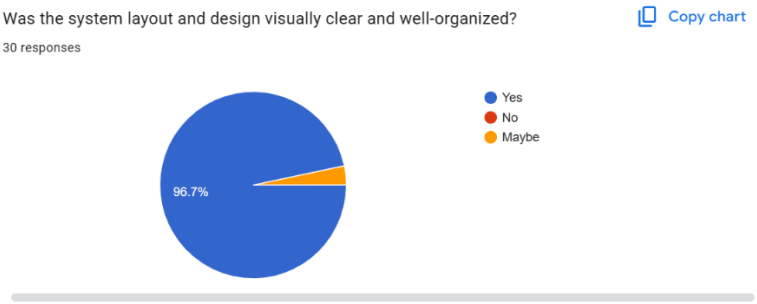


Figure 5. 4. 13 Pie Chart for system layout and design visually clear.

Figure 5. 4. 13 show the pie chart for system layout and design visually clear. 96.7% of the respondents, which is 29 respondents think that the system layout and design visually is clear. 3.3% of the respondents which is 1 respondent did not sure that system layout and design visually is clear and well organized.

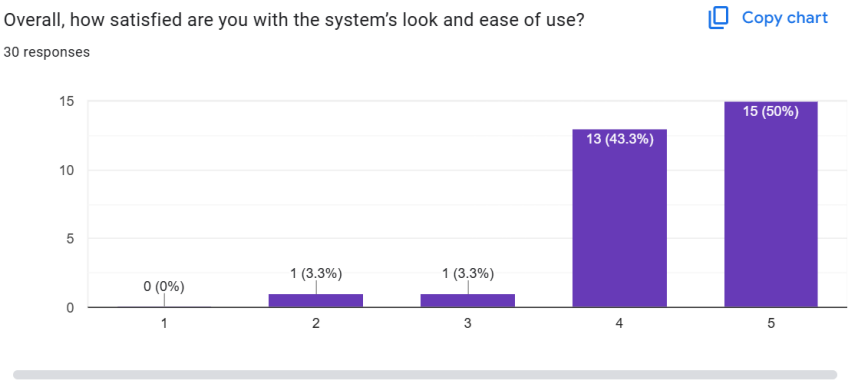


Figure 5. 4. 14 Graph for satisfied with the system's look and easy to use.

Figure 5. 4. 14 show the graph for satisfied with the system's look and easy to use. 50.0% of the respondents, which is 15 respondents think it is very satisfied with the system's look and easy to use. 43.3% of the respondents, which is 13 respondents thinks that it is satisfied. 3.3% of the respondents which is 1 respondent think that it is neutral to use. 3.3% of the respondents which is 1 respondent think that it is unsatisfied.

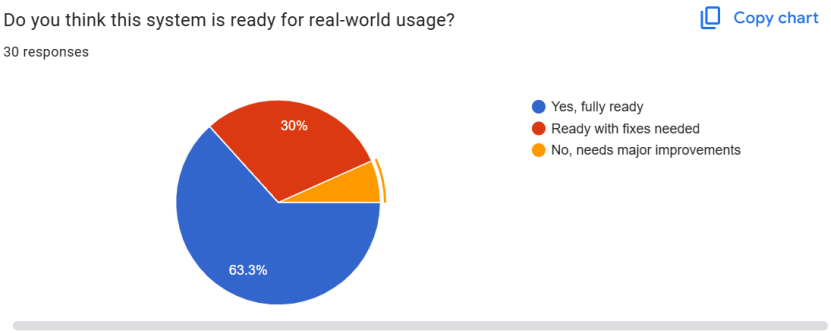


Figure 5. 4. 15 Pie Chart for system readiness for real-world usage.

Figure 5. 4. 15 show the pie chart for system readiness for real-world usage. 63.3% of the respondents, which is 19 respondents think that the system is ready to use for real world. 30.0% of the respondents which is 9 respondents think that the system is ready to use but the system will need to fixed and 6.7% of the respondents which is 2 respondents are not ready for real world usage.

5.5 Evaluation Metrics

Evaluation metrics were used to measure the performance and accuracy of the OCR and NLP components in the Automatic Text Extraction Using OCR for Blood Test Report Management system. These metrics help assess how accurately the system extracts and identifies information from blood test reports. The key evaluation metrics include Character Error Rate (CER), Word Error Rate (WER), and Named Entity Recognition (NER) scores.

Character Error Rate (CER)

BMC is lab report from Borneo Medical Centre, GRIBBLES is from Gribbles Lab, INNOQUEST is from Innoquest lab, TMC is lab report from Timberland Medical Centre, PANTAI is from Pantai Lab, NAVIPATH is from Navipath lab, UM is lab report from University of Malaya. Each report will be extracted using three type which are full-length PDF reports are the standard format received from laboratories, one page of the report PDF report, and PNG images represent scanned documents. By including all three formats, the system is tested for real-world diversity in document types, ensuring that it performs reliably across different input sources. This comprehensive approach improves the overall usability, adaptability, and reliability of the system.

Tester	Lab	File Type	File No	CER (%)
T01	BMC	PDF (Full)	FPDF_1	10.95
T02	BMC	PDF (Full)	FPDF_2	8.97
T03	GRIBBLES	PDF (Full)	FPDF_3	20.53
T04	GRIBBLES	PDF (Full)	FPDF_4	15.86
T05	INNOQUEST	PDF (Full)	FPDF_5	23.65
T06	INNOQUEST	PDF (Full)	FPDF_6	30.46
T07	TMC	PDF (Full)	FPDF_7	20.76
T08	PANTAI	PDF (Full)	FPDF_8	21.48
T09	NAVIPATH	PDF (Full)	FPDF_9	22.87
T10	UM	PDF (Full)	FPDF_10	17.95
T11	BMC	PDF (1 Page)	PDF_1	7.86
T12	BMC	PDF (1 Page)	PDF_2	5.77
T13	GRIBBLES	PDF (1 Page)	PDF_3	16.08
T14	GRIBBLES	PDF (1 Page)	PDF_4	23.76
T15	INNOQUEST	PDF (1 Page)	PDF_5	16.32

T16	INNOQUEST	PDF (1 Page)	PDF_6	15.37
T17	TMC	PDF (1 Page)	PDF_7	14.73
T18	PANTAI	PDF (1 Page)	PDF_8	15.84
T19	NAVIPATH	PDF (1 Page)	PDF_9	19.42
T20	UM	PDF (1 Page)	PDF_10	16.86
T21	BMC	PNG	PNG_1	7.86
T22	BMC	PNG	PNG_2	5.77
T23	GRIBBLES	PNG	PNG_3	16.08
T24	GRIBBLES	PNG	PNG_4	23.76
T25	INNOQUEST	PNG	PNG_5	16.32
T26	INNOQUEST	PNG	PNG_6	15.37
T27	TMC	PNG	PNG_7	14.73
T28	PANTAI	PNG	PNG_8	15.84
T29	NAVIPATH	PNG	PNG_9	19.42
T30	UM	PNG	PNG_10	16.86

Table 5. 4. 1 Character Error Rate for each report.

Total = 10.95 + 8.97 + 20.53 + 15.86 + 23.65 + 30.46 + 20.76 + 21.48 + 22.87 + 17.95 + 7.86 + 5.77 + 16.08 + 23.76 + 16.32 + 15.37 + 14.73 + 15.84 + 19.42 + 16.86 + 7.86 + 5.77 + 16.08 + 23.76 + 16.32 + 15.37 + 14.73 + 15.84 + 19.42 + 16.86 = 496.53

Average CER = 496.53 / 30 = 16.55

From Table 5.4.1, 16.55% of characters in the OCR output were either inserted, deleted, or substituted incorrectly. This indicates a moderate level of character recognition accuracy. Character Error Rate (CER) evaluates how many individual characters were incorrectly recognized during OCR, and it's highly influenced by the visual clarity and structure of the document. A high CER typically results from poor scan quality, low image contrast, unclear fonts, or complex report designs. For example, reports with small, tightly packed fonts or faded text can cause the OCR engine to misread or miss characters entirely, increasing the error rate. This is evident in reports like INNOQUEST, FPDF_6, which has the highest CER at 30.46%, possibly due to poor scan quality or unclear backgrounds. On the other hand, reports like BMC, PNG_2 show a low CER of 5.77%, suggesting that the image was sharp, with clear font and minimal background interference. Consistency in formatting and text alignment also

contributes to better OCR results. Therefore, differences in CER across reports likely reflect variations in scanning resolution, document cleanliness, and formatting complexity.

Word Error Rate (WER)

BMC is lab report from Borneo Medical Centre, GRIBBLES is from Gribbles Lab, INNOQUEST is from Innoquest lab, TMC is lab report from Timberland Medical Centre, PANTAI is from Pantai Lab, NAVIPATH is from Navipath lab, UM is lab report from University of Malaya. Each report will be extracted using three type which are full-length PDF reports are the standard format received from laboratories, one page of the report PDF report, and PNG images represent scanned documents. By including all three formats, the system is tested for real-world diversity in document types, ensuring that it performs reliably across different input sources. This comprehensive approach improves the overall usability, adaptability, and reliability of the system.

Tester	Lab	File Type	File No	WER (%)
T01	BMC	PDF (Full)	FPDF_1	16.34
T02	BMC	PDF (Full)	FPDF_2	10.97
T03	GRIBBLES	PDF (Full)	FPDF_3	35.76
T04	GRIBBLES	PDF (Full)	FPDF_4	26.98
T05	INNOQUEST	PDF (Full)	FPDF_5	43.65
T06	INNOQUEST	PDF (Full)	FPDF_6	50,63
T07	TMC	PDF (Full)	FPDF_7	35.79
T08	PANTAI	PDF (Full)	FPDF_8	35.65
T09	NAVIPATH	PDF (Full)	FPDF_9	36.85
T10	UM	PDF (Full)	FPDF_10	31.36
T11	BMC	PDF (1 Page)	PDF_1	12.76
T12	BMC	PDF (1 Page)	PDF_2	10.65
T13	GRIBBLES	PDF (1 Page)	PDF_3	30.46
T14	GRIBBLES	PDF (1 Page)	PDF_4	16.83
T15	INNOQUEST	PDF (1 Page)	PDF_5	33.85
T16	INNOQUEST	PDF (1 Page)	PDF_6	38.65
T17	TMC	PDF (1 Page)	PDF_7	31.97
T18	PANTAI	PDF (1 Page)	PDF_8	29.53
T19	NAVIPATH	PDF (1 Page)	PDF_9	36.75
T20	UM	PDF (1 Page)	PDF_10	29.59
T21	BMC	PNG	PNG_1	12.76
T22	BMC	PNG	PNG_2	10.65

T23	GRIBBLES	PNG	PNG_3	30.46
T24	GRIBBLES	PNG	PNG_4	16.83
T25	INNOQUEST	PNG	PNG_5	33.85
T26	INNOQUEST	PNG	PNG_6	38.65
T27	TMC	PNG	PNG_7	31.97
T28	PANTAI	PNG	PNG_8	29.53
T29	NAVIPATH	PNG	PNG_9	36.75
T30	UM	PNG	PNG_10	29.59

Table 5. 4. 2 Word Error Rate for each report

Total = 16.34 + 10.97 + 35.76 + 26.98 + 43.65 + 50.63 + 35.79 + 35.65 + 36.85 + 31.36 + 12.76 + 10.65 + 30.46 + 16.83 + 33.85 + 38.65 + 31.97 + 29.53 + 36.75 + 29.59 + 12.76 + 10.65 + 30.46 + 16.83 + 33.85 + 38.65 + 31.97 + 29.53 + 36.75 + 29.59 = 884.76

Average WER = 884.76 / 30 = 29.49

From Figure 5.4.2, we can see that roughly 29.49% of the words in the OCR results were wrong (wrong words, missing, or extra). This shows a high rate of word-level errors, suggesting that sentence structure and context are often affected. Improving segmentation and language modelling could help reduce this. Word Error Rate (WER) is measuring how many entire words are incorrect due to insertion, deletion, or substitution errors. WER is generally higher than CER because a single character mistake can corrupt an entire word, making the sentence difficult to interpret. WER is influenced not only by image quality but also by how well the text is segmented into words. If the OCR engine fails to recognize spaces properly or misinterprets units like “x10⁹/L” or symbols like “%,” it can dramatically increase WER. For instance, INNOQUEST, FPDF_6 recorded a very high WER of 50.63%, which may reflect not just visual issues but also difficulty in handling scientific or medical terms and symbols. BMC, PDF_2 had a low WER of 10.65%, indicating clear formatting, properly spaced words, and simple language structure. Reports with standard layout and minimal noise allow OCR to more accurately detect word boundaries, resulting in lower WER. Therefore, higher WER values

typically point to problems with sentence structure recognition and context understanding during OCR processing.

Named Entity Recognition (NER)

BMC is lab report from Borneo Medical Centre, GRIBBLES is from Gribbles Lab, INNOQUEST is from Innoquest lab, TMC is lab report from Timberland Medical Centre, PANTAI is from Pantai Lab, NAVIPATH is from Navipath lab, UM is lab report from University of Malaya. Each report will be extracted using three type which are full-length PDF reports are the standard format received from laboratories, one page of the report PDF report, and PNG images represent scanned documents. By including all three formats, the system is tested for real-world diversity in document types, ensuring that it performs reliably across different input sources. This comprehensive approach improves the overall usability, adaptability, and reliability of the system.

Tester	Lab	File Type	File No	NER (%)
T01	BMC	PDF (Full)	FPDF_1	80.98
T02	BMC	PDF (Full)	FPDF_2	88.89
T03	GRIBBLES	PDF (Full)	FPDF_3	67.53
T04	GRIBBLES	PDF (Full)	FPDF_4	76.19
T05	INNOQUEST	PDF (Full)	FPDF_5	66.67
T06	INNOQUEST	PDF (Full)	FPDF_6	55.56
T07	TMC	PDF (Full)	FPDF_7	66.67
T08	PANTAI	PDF (Full)	FPDF_8	65.87
T09	NAVIPATH	PDF (Full)	FPDF_9	66.66
T10	UM	PDF (Full)	FPDF_10	70.59
T11	BMC	PDF (1 Page)	PDF_1	87.10
T12	BMC	PDF (1 Page)	PDF_2	88.89
T13	GRIBBLES	PDF (1 Page)	PDF_3	68.75
T14	GRIBBLES	PDF (1 Page)	PDF_4	80.13
T15	INNOQUEST	PDF (1 Page)	PDF_5	70.27
T16	INNOQUEST	PDF (1 Page)	PDF_6	62.34
T17	TMC	PDF (1 Page)	PDF_7	68.67
T18	PANTAI	PDF (1 Page)	PDF_8	73.27
T19	NAVIPATH	PDF (1 Page)	PDF_9	66.66
T20	UM	PDF (1 Page)	PDF_10	72.34
T21	BMC	PNG	PNG_1	87.10
T22	BMC	PNG	PNG_2	88.89

T23	GRIBBLES	PNG	PNG_3	68.75
T24	GRIBBLES	PNG	PNG_4	80.13
T25	INNOQUEST	PNG	PNG_5	69.54
T26	INNOQUEST	PNG	PNG_6	65.87
T27	TMC	PNG	PNG_7	68.67
T28	PANTAI	PNG	PNG_8	70.58
T29	NAVIPATH	PNG	PNG_9	66.66
T30	UM	PNG	PNG_10	72.34

Table 5. 4. 3 Named Entity Recognition for each report

Total = 80.98 + 88.89 + 67.53 + 76.19 + 66.67 + 55.56 + 66.67 + 65.87 + 66.66 + 70.59 + 87.10 + 88.89 + 68.75 + 80.13 + 70.27 + 62.34 + 68.67 + 73.27 + 66.66 + 72.34 + 87.10 + 88.89 + 68.75 + 80.13 + 69.54 + 65.87 + 68.67 + 70.58 + 66.66 + 72.34 = 2095.71

Average NER = 2095.71 / 30 = 69.86

The system correctly identified nearly 70% of named entities. This show that the system is potential for improvement, especially in labs with complex or low-quality OCR output as some NER scores are in 50% range. Named Entity Recognition (NER) measures the system's ability to correctly identify key pieces of information such as patient names, test names, units, dates, and values. NER performance is directly affected by the quality of the OCR output, if the extracted text contains errors, especially in medical terms or numbers, the NER engine might fail to recognize the correct entities. Reports with a high NER score, such as BMC, PNG_2 with 88.89%, benefit from both clean text extraction and consistent lab report structure, allowing the system to detect entities with high accuracy. In contrast, INNOQUEST, FPDF_6 had a lower NER of 55.56%, which could be due to poor OCR results, unusual formatting, or more complex medical terminology that the entity recognition model struggled with. Templates that use abbreviations or irregular spacing between test names and values can also cause entity detection to fail. Thus, a low NER score often reflects a combination of OCR noise and mismatches with expected patterns or known entity formats, highlighting the importance of clean input and consistent structure for effective NER.

5.6 Summary

This chapter provided a comprehensive overview of the testing phase for the Automatic Text Extraction Using Optical Character Recognition (OCR) for Blood Test Report Management System. The objective was to evaluate the system's functionality, integration, and accuracy across its key components. Each core module of the system such as user registration, login, password reset, dashboard, report uploading, data verification, and exporting was individually tested. All test cases passed successfully, demonstrating that each function worked as intended and handled both valid and invalid inputs appropriately.

Integration testing confirmed that inter-module communication and data flow function correctly. Key interactions, such as between uploading, OCR/NLP, verification, saving to database, were tested. The system handled typical and edge cases effectively, such as missing OCR data or Python parser failures, with appropriate error handling and fallback mechanisms.

To assess the accuracy of OCR and NLP performance, three evaluation metrics were applied which are Character Error Rate (CER), average 16.55% which indicates moderate character-level recognition accuracy. Word Error Rate (WER), average 29.49% suggests relatively high word-level inaccuracies, affecting sentence structure and context. Named Entity Recognition (NER) average 69.86% Reflects moderate to good performance in identifying key information.

CHAPTER 6 Conclusion and future work

6.1 Introduction

This project successfully developed an Automatic Medical Data Extraction System that uses Optical Character Recognition (OCR) and Natural Language Processing (NLP) to extract and process data from blood test reports. The system aimed to reduce manual data entry, improve efficiency, and support better clinical data management. Various modules were implemented and tested thoroughly, and the results show that the system performs its intended functions effectively.

6.2 Strengths and Weakness

Strengths

The Automatic Text Extraction Using OCR for Blood Test Report Management system offers several key strengths that enhance its usability and effectiveness in a clinical setting. Firstly, it incorporates semi-automated data extraction from scanned blood test reports, which significantly reduces the burden of manual data entry by medical staff. This not only streamlines the workflow but also minimizes the risk of human error in maintaining accurate medical records. Additionally, the system is designed with multi-format compatibility, supporting various input types including PDF documents and image files. This enables it to handle diverse laboratory report formats from different healthcare institutions, thus increasing its adaptability to real-world use. Furthermore, the system features integrated upload and scanning functions, allowing users to either upload existing files or scan physical documents directly within the platform. This dual functionality makes the system convenient and suitable for various operational environments. Another feature is the interactive data verification process, where users can review, edit, or manually add data after OCR extraction. This step is crucial for ensuring the accuracy and integrity of extracted data, especially in cases where OCR

misreads information. Lastly, the system includes a user-friendly dashboard that allows medical professionals to conveniently view patient test details. These features make the system informative, accessible, and efficient for users, particularly in supporting decision-making and record-keeping in a clinical environment.

Weaknesses

The Automatic Text Extraction Using OCR for Blood Test Report Management system, while effective in many areas, also presents several limitations that impact its deployment in real-world clinical settings. One of the limitations is the lack of advanced security features. The current implementation does not include essential mechanisms such as data encryption, audit trails, or role-based access control. Another weakness is the lack of mobile responsiveness. The system is not optimized for mobile devices, which limits accessibility for healthcare professionals who may need to review or upload reports remotely or while on the move. Additionally, the system has limited support for diverse test formats and units. While it handles common test names and units effectively, it may struggle with non-standard test structures, less common units, or newly introduced lab templates. Finally, the system relies on a local database, which restricts scalability and collaborative access. For broader implementation, especially in multi-user healthcare environments, the system would benefit from migrating to a cloud-based database.

6.3 Future Work

For future work, several enhancements can be implemented to strengthen the system. Firstly, security features should be improved by introducing stricter user permission levels and access control to protect sensitive patient information. Secondly, improving the accuracy of the OCR and NLP modules is crucial to minimize errors and make the extraction process more efficient. Additionally, incorporating an audit trail feature would help track data changes and ensure

accountability in the system. The system should also be expanded to handle a wider variety of blood test report formats and support a broader database of test parameters. Finally, developing a mobile-friendly version would allow users to upload and review blood test data conveniently via smartphones, making the system more accessible and user-centric.

6.4 Summary

This project successfully developed an automatic system for extracting blood test data using OCR and NLP, aimed at reducing manual work, improving efficiency, and supporting clinical data management. The system demonstrated strong performance through tested modules including uploading, scanning, verification, and data visualization. Key strengths include semi-automated data extraction, support for multiple input formats, integrated scanning/upload features, interactive data verification, and a user-friendly dashboard with export options. However, weaknesses remain, such as the lack of advanced security, limited mobile support, restricted compatibility with non-standard formats, and reliance on a local database. Future improvements should focus on enhancing security, improving OCR/NLP accuracy, adding audit trails, expanding format support, transitioning to a cloud database, and developing a mobile-responsive version to make the system more secure, scalable, and accessible.

References

ABBYY. (2025). ABBYY FineReader. https://www.abbyy.com/company/media-and-brand-center/

Ali, A., & Renals, S. (2018). Word error rate estimation for speech recognition: e-WER. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), 20–24. Association for Computational Linguistics. https://doi.org/10.18653/v1/P18-2004

Altitude Accelerator. (2023). RAD Methodology process. Agile software development methodologies. Altitude Accelerator. https://altitudeaccelerator.ca/agile-software-development-methodologies/

Bhattacharjee, S., Delen, D., Ghasemaghaei, M., Kumar, A., & Ngai, E. W. T. (2022). Business and government applications of text mining & Natural Language Processing (NLP) for societal benefit: Introduction to the special issue on "text mining & NLP": Decision Support Systems 162 https://doi.org/10.1016/j.dss.2022.113867

Chawla, M., Jain, R., & Nagrath, P. (2020). Implementation of Tesseract algorithm to extract text from different images. Proceedings of the International Conference on Innovative Computing and Communication (ICICC 2020). https://doi.org/10.2139/ssrn.3589972

Docsumo. (2025). Docsumo Document AI. https://www.docsumo.com/

Fleischhacker, D., Goederle, W., & Kern, R. (2024). Improving OCR quality in 19th century historical documents using a combined machine learning based approach: Computer

Vision and Pattern Recognition https://doi.org/10.48550/arXiv.2401.07787

Google LLC. (2024). Google Lens (Version [1.17]) [Mobile app]. Google Play Store. https://play.google.com/store/apps/details?id=com.google.ar.lens

Hegghammer, T. (2022). OCR with Tesseract, Amazon Textract, and Google Document AI: A benchmarking experiment. Journal of Computational Social Science, 5, 861–882. https://doi.org/10.1007/s42001-021-00149-1

Jugran, S., Kumar, A., Tyagi, B. S., & Anand, V. (2021). Extractive Automatic Text Summarization using SpaCy in Python & NLP. In Proceedings of the 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE) (pp. 1-3). https://doi.org/10.1109/ICACITE51222.2021.9404712

Klippa App B.V. (2025). Klippa. https://www.klippa.com/en/home-en/

Kumar, P. S., & Prasad, K. (2024). Integrating OCR and NLP techniques for accurate text extraction and plagiarism detection in image-based content. Library Progress International, 44(3), 2986-2996. https://doi.org/10.48165/bapas.2024.44.2.1

Malashin I., Masich I., Tynchenko V., Gantimurov A., Nelyub V., & Borodulin A. (2024). Machine Learning & Knowledge Extraction: Image Text Extraction and Natural Language Processing of Unstructured Data from Medical Reports *6*(2), 1361 – 1377 https://doi.org/10.3390/make6020064

Microsoft Corporation. (2024). Microsoft Office Lens (Version [16.0]) [Mobile app]. Google Play Store. https://play.google.com/store/apps/details?id=com.microsoft.office.officelens

Muthusundari, M., Velpoorani, A., Kusuma, S. V., L, T., & Rohini, O. K. (2024). Optical character recognition system using artificial intelligence https://doi.org/10.62486/latia202498

Nair R.P. & Thushara M.G. (2024). Procedia Computer Science: Investigating Natural Language Techniques for Accurate Noun and Verb Extraction 235 2876 – 2885 https://doi.org/10.1016/j.procs.2024.04.272

Nanonets. (2025). Nanonets. https://nanonets.com/

Naturalsoft Ltd. (2024). NaturalReader Text to Speech (Version [8.0]) [Mobile app]. Google Play Store. https://play.google.com/store/apps/details?id=com.naturalsoft.personalweb

Oehm J.B., Wenning O., Storck M., Jiang X., & Varghese J. (2024). Digital Health and Informatics Innovations for Sustainable Health Care Systems: Automatic Extraction of Medication Data from Semi-Structured Prescriptions 1694 – 1698 https://doi.org/10.3233/SHTI240749

Omonije, A. (2024). *Agile methodology: A comprehensive impact on modern business operations*. *International Journal of Science and Research (IJSR)*, 13(2). https://doi.org/10.21275/SR24130104148

Pixelcell Pte Ltd (2024). Homework Scanner: Remove Notes (Version [1.0]) [Mobile app]. Google Play Store. https://play.google.com/store/apps/details?id=com.homework.helper

Pandey, M., Arora, M., Arora, S., Goyal, C., Gera, V. K., & Yadav, H. (2024). Procedia Computer Science: AI-based Integrated Approach for the Development of Intelligent Document Management System (IDMS) 230 725 – 736 https://doi.org/10.1016/j.procs.2023.12.127

Prakash, N. C., Narasimhaiah, A. P., Nagaraj, J. B., Pareek, P. K., Sedam, R. V., & Govindhaiah, N. (2022). A survey on NLP based automatic extractive text summarization using spacy. International Journal of Health Sciences,6(S8), 1514–1525 https://doi.org/10.53730/ijhs.v6nS8.10526

Prakash, N. C. P., Narasimhaiah, A. P., Nagaraj, J. B., Pareek, P. K., Maruthikumar, N. B., & Manjunath, R. I. (2022). Implementation of NLP based automatic text summarization using spacy. International Journal of Health Sciences,6(S5), 7508–7521 https://doi.org/10.53730/ijhs.v6nS5.10574

Ponnuru, M., Ponmalar, S. P., A, L., B, T. S., & G, G. C. (2024). Procedia Computer Science: Image-Based Extraction of Prescription Information using OCR-Tesseract 235 1077 – 1086 https://doi.org/10.1016/j.procs.2024.04.102

Renard Wellnitz. (2023.). Text Fairy (OCR Text Scanner) (Version [5.4]) [Mobile app]. Google Play Store. https://play.google.com/store/apps/details?id=com.renard.ocr

Rossum. (2025). Rossum.ai. https://rossum.ai/

Sugiyono, A. Y., Adrio, K., Tanuwijaya, K., & Suryaningrum, K. M. (2023). Procedia Computer Science: Extracting Information from Vehicle Registration Plate using OCR Tesseract 227 932-938 https://doi.org/10.1016/j.procs.2023.10.600

Sun, C., Yang, Z., Wang, L., Zhang, Y., Lin, H., & Wang, J. (2021). Deep learning with language models improves named entity recognition for PharmaCoNER. BMC Bioinformatics, 22(Suppl 1), Article 602. https://doi.org/10.1186/s12859-021-04260-y

Supriyono, A., Wibawa, A. P., Suyono, & Kurniawan, F. (2024). Telematics and Informatics Reports: Advancements in natural language processing: Implications, challenges, and future directions 16 100173 https://doi.org/10.1016/j.teler.2024.100173

Zhou, X., Zeng, T., Zhang, Y., Liao, Y., Smith, J., Zhang, L., Wang, C., Li, Q., Wu, D., Chong, Y., & Li, X. (2024). Automated data collection tool for real-world cohort studies of chronic hepatitis B: Leveraging OCR and NLP technologies for improved efficiency: New Microbes and New Infections 62 https://doi.org/10.1016/j.nmni.2024.101469

APPENDIX

A . Google Form Questionnaire for requirements.

2. Occupation *

Mark only one oval.

- ◯ Doctor
- ◯ Nurse
- ◯ Other: _____

3. How many blood test reports do you process daily? *

Mark only one oval.

- ◯ 1 - 5
- ◯ 6 - 10
- ◯ 11 - 20
- ◯ 21 and above

4. Describe the issue faced in extracting data from patients' blood test report? *

5. How would you think an application focus on automatic text extraction from blood test report would be useful in your current/future occupation? *

6. How do you manage and record your patients' blood test reports? *

Tick all that apply.

- ☐ Manually transcript the blood test data on paper
- ☐ Scan and save a hardcopy of the document (eg. photostat, printed report)
- ☐ Scan and save a softcopy of the document (eg. pdf, image)
- ☐ Save the softcopy of the document in local device (mobile phone, tablet, laptop, etc.)
- ☐ Save the softcopy of the document on cloud storage service
- ☐ Other: _____

7. What key features would you expect from an automatic text extraction from blood test report system/application? (Select all that apply) *

Tick all that apply.

- ☐ Automatic text extraction by scanning using camera
- ☐ Highlight abnormal test results
- ☐ Export data (eg. Excel)
- ☐ Other: _____

8. Which device do you prefer for using the automatic text extraction tool? *

Tick all that apply.

- ☐ Laptop/Personal computer
- ☐ Tablet
- ☐ Mobile phone

9. What patient information do you need to store from the blood test reports? *

Tick all that apply.

- [ ] Name
- [ ] IC/Passport Number
- [ ] Age
- [ ] Gender
- [ ] Date Of Birth
- [ ] Blood Collection Date
- [ ] Doctor Details
- [ ] Name of Clinic for requesting blood test
- [ ] Name of Blood Test Laboratory (eg.Pathlab, Innoquest)
- [ ] Other: _____

Skip to question 10

Section C: Data Accuracy and Validation

10. Do you prefer a system/application that allows manual corrections after the data extraction process? *

Mark only one oval.

- ◯ Yes, manual correction is needed
- ◯ No, only automated data extraction is sufficient
- ◯ Other: _____

11. Which action should be taken by the system/application if failed to completely extract the data? (eg. data missing) *

Tick all that apply.

- [ ] Notify the user for manual review
- [ ] Automatically fill up the data using 'NA'
- [ ] Other: _____

12. Do you think user authentication would help to enhance the security and privacy of the system/application? (eg. user log in/log out feature) *

Mark only one oval.

- ◯ Yes
- ◯ No

13. For automatic system record deletion, how long should the records to be stored in the system/application? *

Mark only one oval.

- ◯ 6 months
- ◯ 1 year
- ◯ 2 years
- ◯ Other: _____

Skip to question 14

Section D: Data Export and Reporting

14. What type of data visualization tools would you prefer? *

Tick all that apply.

- [ ] Table
- [ ] Line Chart
- [ ] Bar Chart
- [ ] Pie Chart
- [ ] Scatter Plot
- [ ] Not needed

15. Which format of digitised blood test report would be useful for your needs? *

Tick all that apply.

- [ ] Detailed Report, I prefer a detailed report with all test metrics.
- [ ] Summary Report, I want a summary report that shows only the abnormal values.
- [ ] Customizable Report, I would like to have a feature to select metrics appear in the report.
- [ ] Other: _____

16. How do you search for a specific patient record in Electronic Medical Record(EMR)? *

Tick all that apply.

- [ ] Search by patient name
- [ ] Search by patient IC/passport number
- [ ] Search by patient blood collection date
- [ ] Other: _____

Skip to question 17

Section E: Additional Requirements

17. What key data would you like to see at a glance when you first open the dashboard? *

Tick all that apply.

- [ ] Patient count
- [ ] Abnormalities count
- [ ] Recent test result
- [ ] Other: _____

18. Would you prefer text or icon in the system/application interface? *

Mark only one oval.

Text

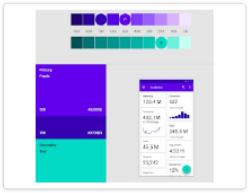
Icon

Both

19. Would you prefer multiple or single color used for the system/application interface? *(Note: The following images are used for color reference purpose only, the actual design will be differed from these images)* *

Mark only one oval.

Single color



Multiple colors

20. What other features would you suggest being included in the automatic text extraction from blood test report system/application? *

B . Google Form Questionnaire for User Acceptance Testing (UAT)

Functional Testing

Was the registration process smooth and error messages shown correctly for invalid or duplicate entries?

- ◯ Yes
- ◯ No
- ◯ Did not test

Were you able to log in successfully using valid credentials (e.g., username, password)?

- ◯ Yes
- ◯ No

Did the system show correct error messages for wrong username or password?

- ◯ Yes
- ◯ No

Was the “Forgot Password” process working correctly and did it allow password reset after verification?

- ◯ Yes
- ◯ No
- ◯ Did not test

Dashboard and Data Display

Did the dashboard display the patient’s test data in a graph and table correctly?

- ◯ Yes
- ◯ No
- ◯ Did not test

Was the filter (by month or test parameter) on the dashboard functioning as expected?

- ◯ Yes
- ◯ No
- ◯ Did not test

Upload, data processing, export

Were you able to upload a blood test report and get results using OCR and NLP?

- ◯ Yes
- ◯ No
- ◯ Did not test

Was the verification page easy to use for checking, adding, or deleting test data?

	1	2	3	4	5	
Very Difficult	◯	◯	◯	◯	◯	Very Easy

Did the system highlight unverified data or display warnings before submission?

- ◯ Yes
- ◯ No

Were you able to export patient reports in PDF or Excel format successfully?

- ◯ Yes
- ◯ No

User Interface & Experience (UI/UX)

Did the icons, buttons, and labels make sense and guide you correctly?

- ◯ Yes
- ◯ No
- ◯ Maybe

Was the navigation between pages (e.g., dashboard, upload, verify) intuitive?

	1	2	3	4	5	
Very Difficult	◯	◯	◯	◯	◯	Very Easy

Was the system layout and design visually clear and well-organized?

- ◯ Yes
- ◯ No
- ◯ Maybe

Overall, how satisfied are you with the system's look and ease of use?

	1	2	3	4	5	
Very unsatisfied	◯	◯	◯	◯	◯	Very Satisfied

Do you think this system is ready for real-world usage?

- ◯ Yes, fully ready
- ◯ Ready with fixes needed
- ◯ No, needs major improvements