A Minor Project Synopsis on

**AI-Based Health Report Summarizer – MediWay**

Submitted in the partial fulfillment for the award of the degree of

**BACHELOR OF TECHNOLOGY**

**in**

Computer Science & Engineering (IoT & Intelligent Systems)

2022-2026

By

Satyam Nautiyal

229302154



Under the guidance of

Dr. G.L. Saini

**School of Computer Science and Engineering**

**Department of IoT & Intelligent Systems**

**Manipal University Jaipur**

**Jaipur, Rajasthan**

**Jan-May 2025**

**1. Introduction**

In today’s healthcare ecosystem, patients frequently receive diagnostic lab reports containing complex medical jargon and numerical values that are difficult to interpret without professional assistance. Understanding these reports is crucial for early disease detection, treatment monitoring, and overall health awareness. However, many individuals struggle to comprehend the significance of their test results, often leading to confusion or unnecessary anxiety.

**MediWay** aims to address this issue by leveraging **Artificial Intelligence (AI) and Natural Language Processing (NLP)** to automatically extract, analyze, and summarize health reports in an easy-to-understand format. The system will allow users to upload medical reports in PDF or image format, extract relevant medical data using Optical Character Recognition (OCR), and generate a simplified summary that provides actionable insights. This tool will empower individuals to take charge of their health by bridging the gap between complex medical data and patient understanding

**2. Motivation**

The increasing digitization of healthcare services has improved accessibility to medical reports, yet the challenge of interpretation remains. Many individuals lack the medical expertise required to understand diagnostic results, which can lead to either undue panic or neglect of serious health conditions.

Current solutions include online medical forums and doctor consultations, but these methods are either time-consuming or inconsistent in accuracy. **MediWay** is motivated by the need for an automated, reliable, and user-friendly solution that provides immediate, accurate, and comprehensible insights from medical reports. By simplifying medical information, this project aims to empower users with knowledge about their health, leading to better medical decision-making and early interventions.

**3. Statement of Problem**

Lab reports contain crucial medical data, but their technical complexity makes them difficult for non-medical professionals to understand. While there are some existing solutions for medical data interpretation, they have notable limitations. The table below outlines the pros and cons of existing methods:

|  |  |  |
| --- | --- | --- |
| **Method** | **Pros** | **Cons** |
| **Manual Doctor Consultation** | Highly accurate and personalized interpretation | Expensive and time-consuming |
| **Online Medical Forums** | Community-based discussion and insights | Information may not be reliable or scientifically backed |
| **Google Search & Self-Research** | Quick access to general medical knowledge | Can be misleading or overly generic |
| **Existing AI-based Solutions** | Some apps provide partial explanations of reports | Limited accuracy and lack of personalization |

**MediWay** overcomes these challenges by providing a fully automated, AI-driven medical report summarization system that offers precise, easy-to-understand, and personalized insights.

4. **Methodology**

* **Phase 1: Data Extraction**
  + Use OCR tools (Tesseract or Google Vision API) to extract data from PDF and image files.
  + Preprocess and structure the extracted data for analysis.
* **Phase 2: Data Processing**
  + Use NLP models (spaCy, BERT) to identify key medical terms and values.
  + Compare test results against predefined normal ranges stored in a database.
* **Phase 3: Summarization**
  + Use pre-trained AI models like **T5** or **GPT** to generate easy-to-understand explanations of the test results.
* **Phase 4: User Interface**
  + Develop a web-based interface to allow users to upload reports and view the results.
* **Phase 5: Future Integration**
  + Once the summarizer is fully functional, integrate a **Doctor Recommendation System** to provide suggestions for specialists based on symptoms.

### 5. Gantt Chart for MediWay

| **Task** | **Week 1** | **Week 2** | **Week 3** | **Week 4** | **Week 5** | **Week 6** | **Week 7** | **Week 8** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Phase 1: Data Extraction (OCR)** | 🟢 | 🟢 |  |  |  |  |  |  |
| - PDF text extraction (PyMuPDF/pdfplumber) | 🟢 | 🟢 |  |  |  |  |  |  |
| - Image OCR (Tesseract/Google Vision) | 🟢 | 🟢 |  |  |  |  |  |  |
| - Text preprocessing & cleaning | 🟢 | 🟢 |  |  |  |  |  |  |
| **Phase 2: Data Processing** |  | 🟢 | 🟢 |  |  |  |  |  |
| - Identifying test names & values (Regex/NLP) |  | 🟢 | 🟢 |  |  |  |  |  |
| - Mapping values to normal reference ranges |  | 🟢 | 🟢 |  |  |  |  |  |
| **Phase 3: AI-Based Summarization** |  |  | 🟢 | 🟢 | 🟢 |  |  |  |
| - Fine-tuning NLP models (BERT/Gemini/Grok/T5) |  |  | 🟢 | 🟢 | 🟢 |  |  |  |
| - Generating explanations & insights |  |  | 🟢 | 🟢 | 🟢 |  |  |  |
| **Phase 4: UI & Frontend Development** |  |  |  | 🟢 | 🟢 | 🟢 |  |  |
| - Web/ UI design (Streamlit/React Js) |  |  |  | 🟢 | 🟢 | 🟢 |  |  |
| - Backend API integration (Flask) |  |  |  | 🟢 | 🟢 | 🟢 |  |  |
| **Phase 5: Testing & Deployment** |  |  |  |  |  | 🟢 | 🟢 | 🟢 |
| - Bug fixing & performance optimization |  |  |  |  |  | 🟢 | 🟢 | 🟢 |
| - Final deployment & project documentation |  |  |  |  |  |  | 🟢 | 🟢 |

🟢 **Indicates active work on that task during the given weeks.**

**6. Facilities Required for Proposed Work**

To develop **MediWay**, the following hardware and software resources are required:

**Software Requirements:**

* **Programming Languages:** Python, JavaScript
* **OCR Tools:** Tesseract OCR, Google Vision API
* **NLP Libraries:** spaCy, Hugging Face Transformers, BERT, Grok
* **Web Frameworks:** Flask
* **Frontend Development:** Streamlit/React.js
* **Database:** MySQL (for storing reference values and user data)
* **Cloud Services:** Google Cloud / AWS (for scalable processing)

**Hardware Requirements:**

* **Processor:** Minimum Intel i5 or equivalent (for development purposes)
* **RAM:** Minimum 8GB (16GB recommended for deep learning model training)
* **GPU (Optional):** NVIDIA GTX/RTX series (for AI model acceleration)
* **Storage:** Minimum 100GB SSD (for datasets and model storage)

These resources will ensure smooth development and deployment of the AI-powered health report summarization system.

**7. Bibliography / References**

1. **Optical Character Recognition (OCR)**
   * Smith, R. (2007). "An Overview of the Tesseract OCR Engine." *Proceedings of the Ninth International Conference on Document Analysis and Recognition.* IEEE.
   * Google Cloud. "Vision AI – Optical Character Recognition." https://cloud.google.com/vision
2. **Natural Language Processing (NLP)**
   * Jurafsky, D., & Martin, J. H. (2021). *Speech and Language Processing.* 3rd Edition. Pearson.
   * Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." arXiv preprint arXiv:1810.04805.
3. **Machine Learning & AI for Medical Applications**
   * Rajpurkar, P., et al. (2017). "CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning." *arXiv preprint arXiv:1711.05225.*
   * Topol, E. (2019). *Deep Medicine: How Artificial Intelligence Can Make Healthcare Human Again.* Basic Books.
4. **Medical Reference Ranges & Health Data**
   * Mayo Clinic. (2023). "Laboratory Reference Ranges." <https://www.mayoclinic.org>
   * WHO (World Health Organization). "Normal Hematological Ranges and Guidelines." <https://www.who.int>
5. **Software Development & APIs**
   * Grinberg, M. (2018). *Flask Web Development: Developing Web Applications with Python.* O'Reilly Media.
   * Mozilla. "MDN Web Docs – Web APIs and JavaScript Frameworks." <https://developer.mozilla.org>