 

HAROHALLI, KANAKAPURA ROAD – 562112

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

(DATA SCIENCE)

**MATLAB PROJECT REPORT**

ON

#### 

#### "MATLAB BASED RECOMMENDER SYSTEM FOR PERSONALIZED MEDICINE"

2024-2025

BACHELOR OF TECHNOLOGY IN

COMPUTER SCIENCE & ENGINEERING (DATA SCIENCE)

**Submitted by**

MANJUNATH V– ENG23DS0020

**Under The Supervision of:**

#### Prof. Shivamma D.

#### Assistant Professor

#### Department of CSE (Data Science), DSU

DAYANANDA SAGAR UNIVERSITY

#### DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

**(DATA SCIENCE)**

#### SCHOOL OF ENGINEERING

**HAROHALLI, KANAKAPURA ROAD – 562112**



CERTIFICATE

#### It is certified that the mini project work entitled “MATLAB BASED RECOMMENDER SYSTEM FOR PERSONALIZED MEDICINE” has been carried out at *Dayananda Sagar University*, Bangalore, by *MANJUNATH V (ENG23DS0020)*, Bonafide student of fourth Semester, B.Tech in partial fulfilment for the award of degree in *Bachelor of Technology in Computer Science & Engineering (Data Science)* during academic year *2024-25*. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in departmental library.

#### The project report has been approved as it satisfies the academic requirements in respect of project work for the said degree.

**Signature of the Guide Signature of the Chairperson**

**ACKNOWLEDGEMENT**

A project's successful completion offers a sense of satisfaction, but it is never finished without expressing gratitude to everyone who contributed to its accomplishment. We would like to convey our sincere gratitude to our esteemed university, Dayananda Sagar University, for offering the first-rate facilities.

We are especially thankful to our Chairperson, Dr. Shaila S G, for providing necessary departmental facilities, moral support and encouragement. The largest measure of our acknowledgment is reserved for Prof. Shivamma D. whose guidance and support made it possible to complete the project work in a timely manner.

We would want to thank everyone who has assisted us in successfully completing this project work, both directly and indirectly. The staff has provided us with a great deal of direction and cooperation.

MANJUNATH V- ENG23DS0020

**DECLARATION**

We hereby declare that the project entitled **" MATLAB BASED RECOMMENDER SYSTEM FOR PERSONALIZED MEDICINE"** submitted to Dayananda Sagar University, Bengaluru, is a bona fide record of the work carried out by me under the guidance of Prof. Shivamma D., Assistant Professor in the Dayananda Sagar University School of Engineering's Department of Computer Science and Engineering (Data Science). This work is submitted toward the partial fulfillment of the requirements for the award of a Bachelor of Technology in Computer Science and Engineering (Data Science).

MANJUNATH V- ENG23DS0020

**ABSTRACT**

The **Car Rental Management System** is a software application designed to automate and streamline the process of renting cars to customers. Traditional car rental services often rely on manual paperwork and processes, which can lead to inefficiencies, errors, and poor customer experience. This project aims to eliminate these challenges by developing a user-friendly, efficient, and secure digital system that handles car reservations, customer management, billing, and inventory control.

The system allows customers to view available cars, check rental rates, book vehicles, and manage their rentals through a web or desktop interface. Administrators and employees can manage vehicle data, track rentals, generate invoices, and monitor the status of each car in real-time. Key features include a centralized database, user authentication, car availability tracking, rental history, and automated billing.

Technologies used in the development of the system may include front-end interfaces (HTML/CSS/JavaScript), back-end logic (Java/PHP/Python), and database management (MySQL/SQLite). The system enhances operational efficiency, reduces manual workload, and improves overall customer satisfaction.

This project demonstrates the application of software development principles in solving real-world business problems and provides a scalable solution that can be adopted by car rental agencies of various sizes.

**TABLE OF CONTENTS**

|  |  |
| --- | --- |
| **CHAPTER** | **PG. NO.** |
| 1. INTRODUCTION | 7-7 |
| 1. OBJECTIVES AND SCOPE OF WORK | 8-9 |
| 1. DESCRIPTION OF WORK | 10-13 |
| 1. SOURCE CODE | 14-18 |
| 1. RESULT | 19-19 |
| 1. CONCLUSION | 20-20 |
| 1. REFERENCES | 21-22 |

**INTRODUCTION**

The healthcare industry is currently experiencing an unprecedented surge in data generation from diverse sources such as electronic health records (EHRs), genetic sequencing, wearable devices, medical imaging, and mobile health applications. While this abundance of data holds immense potential, the key challenge lies in converting raw data into actionable insights that can support clinical decision-making. Recommender systems powered by machine learning have emerged as a promising solution, capable of predicting user preferences and delivering personalized suggestions by analysing past behaviour and contextual information. Traditional analytical approaches often struggle to manage the complexity and heterogeneity of modern health data, which is where recommender systems—drawing on machine learning, data mining, and artificial intelligence—offer a powerful alternative. These systems can learn from patient histories and similarities to provide intelligent, data-driven treatment recommendations. The rise of precision medicine has further emphasized the need for individualized treatment plans that account for genetic, environmental, and lifestyle factors, moving beyond the "one-size-fits-all" model of traditional healthcare. Personalized medicine aims to improve treatment efficacy and reduce adverse effects, and recommender systems play a crucial role in this transformation by processing and interpreting large-scale clinical and biomedical datasets. This project focuses on developing a MATLAB-based personalized medicine recommender system that suggests optimal treatments by analysing patient profiles and historical medical data. MATLAB, known for its high-performance computing capabilities, offers a flexible environment for building such systems through its rich set of toolboxes for machine learning, statistics, and optimization. Additionally, MATLAB's robust data handling, integration with external sources, and powerful visualization tools make it particularly suited for developing healthcare informatics applications. Its ability to efficiently process matrix operations also supports the implementation of collaborative filtering and other recommendation algorithms, reinforcing its suitability for this project.

### OBJECTIVE AND SCOPE OF WORK

### Objective

### The primary objective of this project is to design and develop a MATLAB-based Recommender System that supports personalized medicine by suggesting individualized treatment options based on patient-specific characteristics and historical data. The recommender system aims to analyse patterns in past medical records, identify similarities among patient profiles, and generate accurate, data-driven treatment recommendations.

### The system is intended to function as a Clinical Decision Support Tool (CDST) to assist healthcare providers in choosing the most effective therapeutic strategies tailored to the unique needs of each patient. It leverages machine learning algorithms, data preprocessing techniques, and similarity metrics to ensure intelligent and interpretable recommendations.

### Specific Objectives

### The primary goal of this project is to develop a MATLAB-based recommender system capable of delivering personalized treatment recommendations using patient-specific data. To achieve this, several specific objectives have been identified and structured into logical development phases.

### The first objective is data collection and preprocessing. Since access to real patient data is often restricted due to privacy and ethical concerns, the system will rely on synthetic or publicly available medical datasets. These datasets will be carefully curated and pre-processed to ensure data quality and consistency. Preprocessing will include cleaning the data to remove errors and noise, normalizing numerical values, and encoding categorical variables. Special attention will be given to handling missing values and resolving heterogeneity within the data to ensure that it is suitable for training machine learning models.

### The next step involves feature engineering, which is crucial for the accuracy and relevance of the recommender system. This phase focuses on identifying and extracting meaningful features from the dataset. These features may include patient demographic information (such as age and gender), clinical diagnoses, treatment history, comorbid conditions, and laboratory test results. The goal is to select attributes that significantly influence treatment outcomes and personalize the recommendation process.

### Finally, the project will include thorough documentation and reporting. A complete record of the system’s design, implementation, testing procedures, and performance results will be compiled. This documentation will serve academic, educational, and potential clinical development purposes and will include detailed explanations of algorithms used, design decisions made, system architecture, limitations, and potential future work.

### Scope of Work

### The project focuses on the conceptualization, design, and prototype implementation of a personalized medicine recommender system within a MATLAB environment. The scope includes data processing, algorithm development, model evaluation, and user interface design, all aimed at delivering a functional prototype that demonstrates the core principles of personalized treatment recommendation.

### In-Scope Activities:

### Use of publicly available medical datasets or synthetic patient records.

### Development of core recommender algorithms within MATLAB.

### Application to a specific disease domain, such as diabetes, cancer, or hypertension.

### Development of a graphical user interface (GUI) for demonstration purposes.

### Performance benchmarking using multiple machine learning metrics.

### Designing a modular and scalable system architecture for future improvements.

### 14

### Real-time deployment in hospital or clinical settings.

### Integration with commercial Electronic Health Record (EHR) systems.

### Deep learning-based approaches requiring high computational resources.

### Legal, ethical, or regulatory compliance testing for clinical deployment.

### DESCRIPTION OF WORK

### This project involves the development of a MATLAB-based recommender system for personalized medicine. The main goal is to provide treatment suggestions tailored to individual patients using machine learning techniques and patient-specific data. Personalized medicine is an emerging field that uses data-driven approaches to optimize healthcare decisions for individual patients. The recommender system developed in this project aims to support clinical decision-making by analysing patient data and identifying suitable treatment options based on previous outcomes and patient similarities.

### The work began with a thorough understanding of the problem and identification of the system requirements. The need for personalized recommendations in healthcare was established, and objectives were clearly defined. This included acquiring relevant data, selecting appropriate algorithms, evaluating system performance, and building a user interface for interaction. Once the problem space was well-understood, the design and planning phases of the recommender system were initiated.

### The first practical step in the project was data collection and preprocessing. Due to ethical and privacy concerns, synthetic or publicly available datasets were used to simulate real-world patient information. These datasets typically contained variables such as patient age, gender, medical conditions, previous treatments, and clinical outcomes. Preprocessing was essential to ensure the data was clean, consistent, and ready for analysis. This involved removing duplicate or irrelevant entries, handling missing values through imputation or exclusion, normalizing numerical data, and encoding categorical variables into a machine-readable format.

### Feature engineering followed data preprocessing. This stage involved selecting the most important and clinically relevant attributes that would be used in the recommendation process. Key features included demographic information, diagnostic codes, treatment history, and lab results. In addition to these, derived features—such as the number of prior treatments or time between diagnoses—were created to enhance the model's predictive power. This process ensured that the recommender system could learn from meaningful and representative data.

### METHODOLOGY

### The development of a personalized medicine recommender system involved the following major steps: data collection, data preprocessing, design of the recommendation logic, system development using MATLAB App Designer, and evaluation. Each of these steps is outlined below.

### 1. Data Collection (continued)

### Diet Dataset: Recommended dietary guidelines tailored to specific diseases.

### Workout Dataset: Suggested physical exercises appropriate for various health conditions.

### Precaution Dataset: Provided preventive measures to minimize disease impact or progression.

### These datasets were selected to enable a holistic approach to personalized medicine, considering not only diagnosis but also lifestyle interventions.

### 2. Data Preprocessing

### The collected datasets underwent preprocessing to ensure consistency and usability in the recommendation engine:

### Data Cleaning: Removed missing values, duplicate entries, and corrected inconsistent formatting (e.g., symptom spelling variations).

### Normalization: Standardized the format of symptoms, diseases, and severity scores to allow for effective matching.

### Encoding: Symptoms and diseases were encoded numerically for efficient computation and model input.

### Integration: Merged datasets based on common attributes (e.g., disease name) to create a unified data structure suitable for personalized recommendations.

### 3. System Design and Architecture

### The recommender system was designed to function as a hybrid model incorporating both rule-based filtering and content-based recommendation techniques:

### Symptom Matching: Users input their symptoms via the GUI. These symptoms are matched with diseases from the training dataset using a similarity scoring method.

### Severity Assessment: Each symptom's severity is factored in to prioritize urgent conditions.

### Disease Prediction: A weighted rule-based classifier determines the most likely disease based on symptom match and severity.

### Personalized Recommendation: Once the disease is predicted, corresponding diet plans, workout routines, and precautions are retrieved and displayed to the user.

### 4. MATLAB App Development

### The front-end of the system was developed using MATLAB App Designer, providing an intuitive GUI for user interaction. Key components included:

### Input Panel: Allows users to select or type symptoms.

### Submit Button: Triggers the disease prediction algorithm.

### Output Display: Shows predicted disease, its description, severity level, and recommended actions (diet, workout, and precautions).

### Modular Code Structure: Backend logic is modularized for ease of maintenance and future expansion.

### 5. Evaluation and Testing

### The recommender system was tested using sample inputs to verify the following:

### Accuracy of Disease Prediction: Compared predicted diseases with ground truth from the dataset.

### Relevance of Recommendations: Ensured that diet, workout, and precautions matched expert guidelines for the predicted diseases.

### Usability Testing: Evaluated GUI functionality for responsiveness, clarity, and user-friendliness.

### 6. Future Improvements

### Machine Learning Integration: Future versions may include ML models (e.g., decision trees or SVMs) to improve prediction accuracy.

### User Feedback Loop: Adding a feedback system to refine recommendations based on user outcomes.

### API Connectivity: Integrating external APIs for up-to-date medical guidelines.

### Let me know if you’d like help formatting this into LaTeX, Word format, or need assistance with other report sections (like Abstract, Results, or Conclusion).

### SOURCE CODE

### The core functionality of the personalized medicine recommender system is implemented using MATLAB. The source code is designed to read multiple medical datasets, accept user input symptoms, predict the most probable disease, and provide tailored recommendations related to medications, diet, workout, and precautions. Below is an overview of the key components and logic in the code:

### “”””””””””””

### %% Disease Recommender System in MATLAB

### % Read CSV files with preserved original headers

### description = readtable('description.csv', 'VariableNamingRule', 'preserve');

### diets = readtable('diets.csv', 'VariableNamingRule', 'preserve');

### medications = readtable('medications.csv', 'VariableNamingRule', 'preserve');

### precautions = readtable('precautions\_df.csv', 'VariableNamingRule', 'preserve');

### symptoms\_df = readtable('symtoms\_df.csv', 'VariableNamingRule', 'preserve');

### workouts = readtable('workout\_df.csv', 'VariableNamingRule', 'preserve');

### severity = readtable('Symptom-severity.csv', 'VariableNamingRule', 'preserve');

### training = readtable('Training.csv', 'VariableNamingRule', 'preserve');

### % Convert all column names to lowercase for consistency

### description.Properties.VariableNames = lower(description.Properties.VariableNames);

### diets.Properties.VariableNames = lower(diets.Properties.VariableNames);

### medications.Properties.VariableNames = lower(medications.Properties.VariableNames);

### precautions.Properties.VariableNames = lower(precautions.Properties.VariableNames);

### symptoms\_df.Properties.VariableNames = lower(symptoms\_df.Properties.VariableNames);

### workouts.Properties.VariableNames = lower(workouts.Properties.VariableNames);

### severity.Properties.VariableNames = lower(severity.Properties.VariableNames);

### training.Properties.VariableNames = lower(training.Properties.VariableNames);

### %% Replace missing strings with 'None'

### vars = training.Properties.VariableNames;

### for i = 1:length(vars)

### col = training.(vars{i});

### if iscell(col) || isstring(col)

### idx = cellfun(@(x) isempty(x) || (isstring(x) && strlength(x) == 0), col);

### col(idx) = {'None'};

### training.(vars{i}) = col;

### end

### end

### %% Take user symptoms as input

### user\_input = input('Enter your symptoms (comma separated): ', 's');

### input\_symptoms = strtrim(strsplit(lower(user\_input), ','));

### %% Match symptoms to training data

### match\_scores = zeros(height(training), 1);

### for i = 1:height(training)

### row\_symptoms = lower(string(training{i, 1:end-1}));

### match\_scores(i) = sum(ismember(input\_symptoms, row\_symptoms));

### end

### [~, best\_match\_idx] = max(match\_scores);

### predicted\_disease = training{best\_match\_idx, end}{1};

### fprintf('\n🔍 Predicted Disease: %s\n', predicted\_disease);

### %% Display information

### % Description

### desc\_idx = strcmpi(description.disease, predicted\_disease);

### if any(desc\_idx)

### fprintf('📝 Description: %s\n', description.description{desc\_idx});

### end

### % Medications

### med\_idx = strcmpi(medications.disease, predicted\_disease);

### if any(med\_idx)

### meds = medications{med\_idx, 2:end};

### meds = meds(~cellfun('isempty', meds));

### fprintf('💊 Medications: %s\n', strjoin(meds, ', '));

### end

### % Precautions

### prec\_idx = strcmpi(precautions.disease, predicted\_disease);

### if any(prec\_idx)

### pre = precautions{prec\_idx, 2:end};

### pre = pre(~cellfun('isempty', pre));

### fprintf('⚠ Precautions: %s\n', strjoin(pre, ', '));

### end

### % Diets

### diet\_idx = strcmpi(diets.disease, predicted\_disease);

### if any(diet\_idx)

### diet = diets{diet\_idx, 2:end};

### diet = diet(~cellfun('isempty', diet));

### fprintf('🥗 Diets: %s\n', strjoin(diet, ', '));

### end

### % Workouts

### work\_idx = strcmpi(workouts.disease, predicted\_disease);

### if any(work\_idx)

### work = workouts{work\_idx, 2:end};

### work = work(~cellfun('isempty', work));

### fprintf('🏃 Workouts: %s\n', strjoin(work, ', '));

### end

### “”””””””””””

### This is the code we have used in MATLAB for doing the project and training the model.

### This recommender system is implemented using MATLAB and provides personalized medical suggestions based on user-input symptoms. The system utilizes multiple medical datasets loaded from CSV files, including:

### Disease descriptions

### Training dataset (symptom-disease mapping)

### Symptom severity

### Medications

### Diet plans

### Workouts

### Precautions

### Key Steps in the Code:

### Data Import: Reads all CSV files using readtable() with preserved headers.

### Column Standardization: Converts all column names to lowercase for uniformity.

### Data Cleaning: Replaces missing strings in the training data with 'None'.

### User Input: Accepts comma-separated symptoms from the user.

### Symptom Matching: Compares input symptoms with each row in the training set.

### Disease Prediction: Selects the disease with the highest number of matching symptoms.

### Information Retrieval: Fetches:

### Disease description

### Medications

### Recommended diets

### Workouts

### Precautions from respective datasets.

### Display Output: Presents the predicted disease and related recommendations to the user.

### The approach uses a simple rule-based method without machine learning, enabling easy interpretation and real-time interaction through MATLAB’s console or App Designer interface.

### RESULT

### 

### 

### CONCLUSION

### This project presents a MATLAB-based personalized medicine recommender system that predicts diseases based on user-input symptoms and provides tailored medical guidance. By integrating multiple healthcare datasets—covering disease descriptions, symptoms, severity, medications, diets, workouts, and precautions—the system delivers a holistic solution for preliminary health assessment.

### The system uses a simple rule-based symptom matching algorithm to identify the most probable disease, making it easy to interpret and implement. MATLAB’s App Designer was instrumental in creating a user-friendly interface for real-time interaction. Data preprocessing steps such as normalization and missing value handling significantly enhanced the reliability and consistency of the predictions.

### The recommender system not only identifies likely conditions but also suggests supportive actions like diet plans and exercise routines, promoting preventive healthcare. While not a substitute for professional diagnosis, it serves as an initial step toward self-awareness and early detection.

### This tool demonstrates the potential of data-driven healthcare and offers a foundation for future expansion. With the integration of machine learning models, the system could be further improved for better prediction accuracy and adaptability. Overall, the project showcases how technology can bridge the gap between data science and personalized medicine.

### REFERENCES

### 1. [A Systematic Review and Research Perspective on Recommender Systems](https://journalofbigdata.springeropen.com/articles/10.1186/s40537-022-00592-5)

### 2. [From Hype to Reality: Data Science Enabling Personalized Medicine](https://bmcmedicine.biomedcentral.com/articles/10.1186/s12916-018-1122-7)

### 3. [Improving Clinical Decision Making with a Two-Stage Recommender System](https://pubmed.ncbi.nlm.nih.gov/37738190/)

### 4. [Kaggle](https://www.kaggle.com/)

### 5. [Medical Recommender Systems Based on Continuous-Valued Logic](https://bmcmedinformdecismak.biomedcentral.com/articles/10.1186/s12911-021-01553-3)

### 6. [Recommendation Systems: Principles, Methods and Evaluation](https://www.sciencedirect.com/science/article/pii/S1110866515000341)

### 7. [Personalized Healthcare Recommendations System - Medium](https://medium.com/@konda.sumanayana1209/personalized-healthcare-recommendations-system-448a00a141b1)

### 8. [Building a Machine Learning System for Personalized Healthcare Recommendations](https://bgiri-gcloud.medium.com/building-a-machine-learning-system-for-personalized-healthcare-recommendations-10dbd3b563d8)

### 9. [Precision Medicine: AI and Big Data Use Cases in Healthcare](https://www.softeq.com/blog/precision-medicine-ai-and-big-data-use-cases-in-healthcare)

### 10. [Personalized Medicine: A Comprehensive Review](https://www.orientjchem.org/vol40no4/personalized-medicine-a-comprehensive-review/)

### 11. [Understanding Fairness in Recommender Systems](https://www.arxiv.org/abs/2409.03893)

### 12. [A Systematic Review and Research Perspective on Recommender Systems (Duplicate)](https://journalofbigdata.springeropen.com/articles/10.1186/s40537-022-00592-5)

### 13. [ChatGPT](https://chat.openai.com/)