University of Asia Pacific | UAP

Department of CSE

Course Title: Artificial Intelligence and Expert Systems Lab

Course Code: CSE 404

Title: MedLog | An Expert System for Medical Diagnosis, Emergency Detection, and Lifestyle Advice.

Date of Submission: 12_08_2025

Submitted by:

Name: Bokhtear Md Abid

Reg: 22101124

Section: C

Group: C-1

Submitted to:

Bidita Sarkar Diba

Lecturer

Department of Computer Science and Engineering

University of Asia Pacafic

Problem Description

The *MedLog* project is a Prolog-based expert system designed to help diagnose common diseases, detect emergency health conditions, suggest treatments, and provide lifestyle advice based on a patient's symptoms, medical history, and risk factors.

The system can store and manage data for multiple patients, perform rule-based reasoning to identify possible diseases, and offer preventive measures and hospital referrals. It also integrates symptom severity tracking and multi-disease detection for more accurate assistance.

Main Features:

- **Multi-patient support**: Each patient's symptoms, history, and risk factors are stored separately.
- **Disease diagnosis**: Identifies diseases such as flu, COVID-19, hypertension, diabetes, and dengue based on symptom patterns.
- **Emergency detection**: Recognizes critical conditions that require immediate medical attention.
- **Treatment suggestions**: Provides recommended medical treatments for diagnosed diseases.
- **Lifestyle advice**: Gives preventive and long-term health improvement suggestions.
- **Risk factor tracking**: Stores risk factors like smoking, obesity, and family history.
- **Allergy database**: Prevents unsafe medication suggestions.
- **Symptom severity tracking**: Helps prioritize urgent medical cases.
- **Hospital referral system**: Suggests ICU, specialist, or home care based on condition severity.
- **Multi-disease detection**: Lists all possible diseases a patient may have.

Tools and Languages Used

- **Programming Language:** Prolog
    - Used to define the knowledge base, create rules for diagnosis, suggest treatments, and manage patient records.
- **Tools:**
    - **SWI-Prolog** — for writing, running, and testing the system.

Diagram / Figure

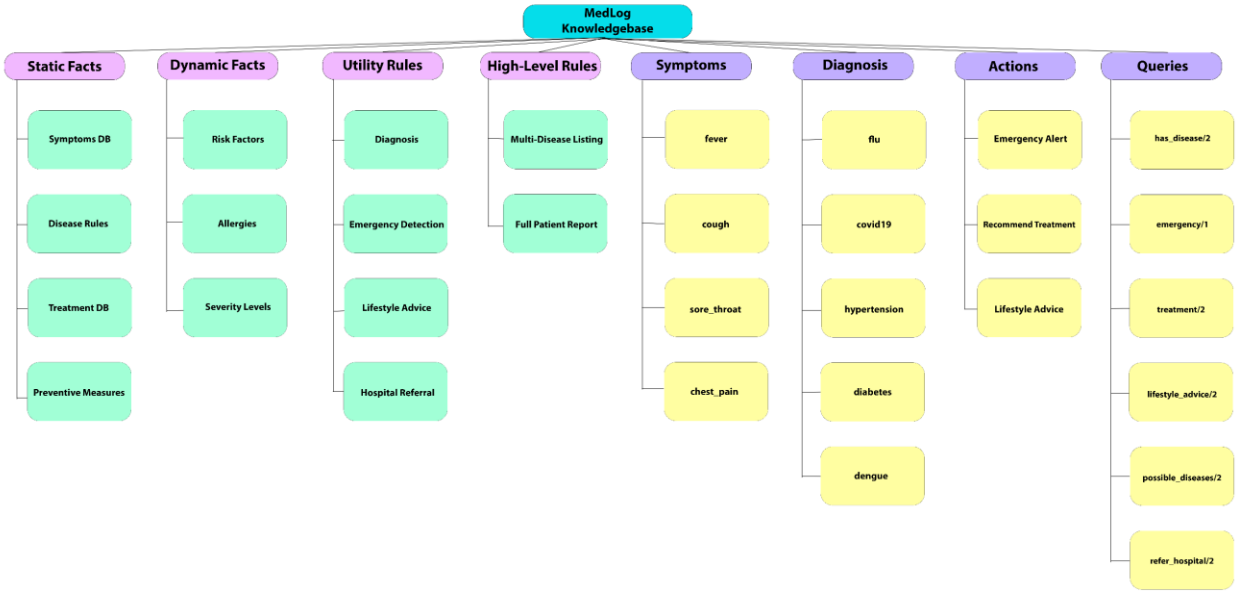

Fig: Knowledge Base Diagram of MedLog Expert System

- **Static Facts**:
    - Symptoms database (per patient)
    - Diseases and their symptom requirements
    - Treatment suggestions
    - Preventive measures
- **Dynamic Facts**:
    - Risk factors
    - Allergies
    - Symptom severity levels
- **Utility Rules**:
    - Disease diagnosis
    - Emergency detection
    - Lifestyle advice

- Hospital referral suggestions

- **High-Level Rules**:
  - Multi-disease listing
  - Full patient diagnosis with treatment and prevention plan

Sample Input/Output:

1. Check if Mukit has disease:

```
?- has_disease(mukit, Disease).
Disease = flu
```

2. List possible diseases for Rakib:

```
?- possible_diseases(rakib, Diseases).
Diseases = [covid19].
```

3. Get treatment for COVID-19:

```
?- treatment(covid19, Treatment).
Treatment = 'Isolate, monitor oxygen level, consult a doctor'.
```

4. Emergency detection for Sadman:

```
?- emergency(sadman).
true.
```

5. Hospital referral for Nirob:

```
?- refer_hospital(nirob, Referral).
Referral = 'Specialist Consultation Needed'
```

6. Preventive measure for dengue:

```
?- preventive(dengue, Measure).
Measure = 'Use mosquito nets, avoid stagnant water'.
```

7. Severity of chest pain:

```
?- severity(chest_pain, Level).
Level = high.
```

8. Risk factor for Lamia:

```
?- risk_factor(lamia, Risk).
Risk = family_history_diabetes.
```

Conclusion

The *MedLog* Prolog expert system successfully implements a rule-based diagnosis approach for multiple patients. By combining symptoms, risk factors, allergies, and severity tracking, the system can provide accurate disease predictions, emergency alerts, and lifestyle advice. The hospital referral system enhances its practical usability for triaging patients.

Challenges

- **Accuracy**: The system depends on correct and complete symptom entry.
- **Scalability**: Adding more diseases and symptoms increases rule complexity.
- **Maintenance**: Updating medical knowledge requires careful rule adjustments.

- **No real-time data**: The system currently does not connect to live health sensors.