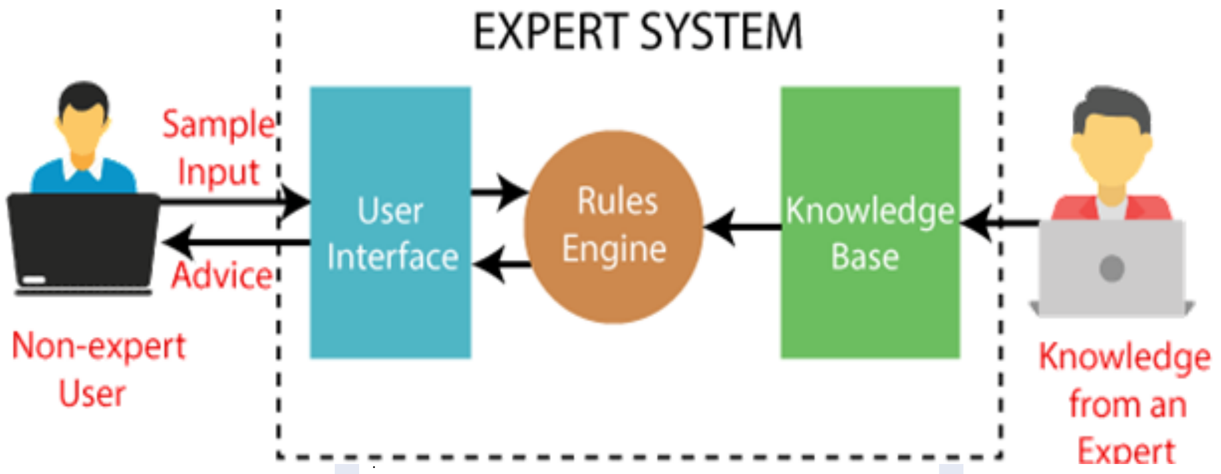


[ADVANCING HEALTHCARE: DEVELOPMENT OF A MEDICAL EXPERT SYSTEM FOR ACCURATE DIAGNOSIS]

Supervisors:
-Dr.Lekehali Somia
-M. Chawki ABBES

Members:G3
-Batiche Chourouq
-Boufroura Rania

[2023-2024]

Medical Expert System Report

Topic Description:

This report details the development of a Medical Expert System aimed at assisting in diagnosing common illnesses based on symptoms presented by patients. Leveraging a knowledge base of medical rules and employing backward chaining inference, the system offers comprehensive diagnoses to users.

Problem Formulation:

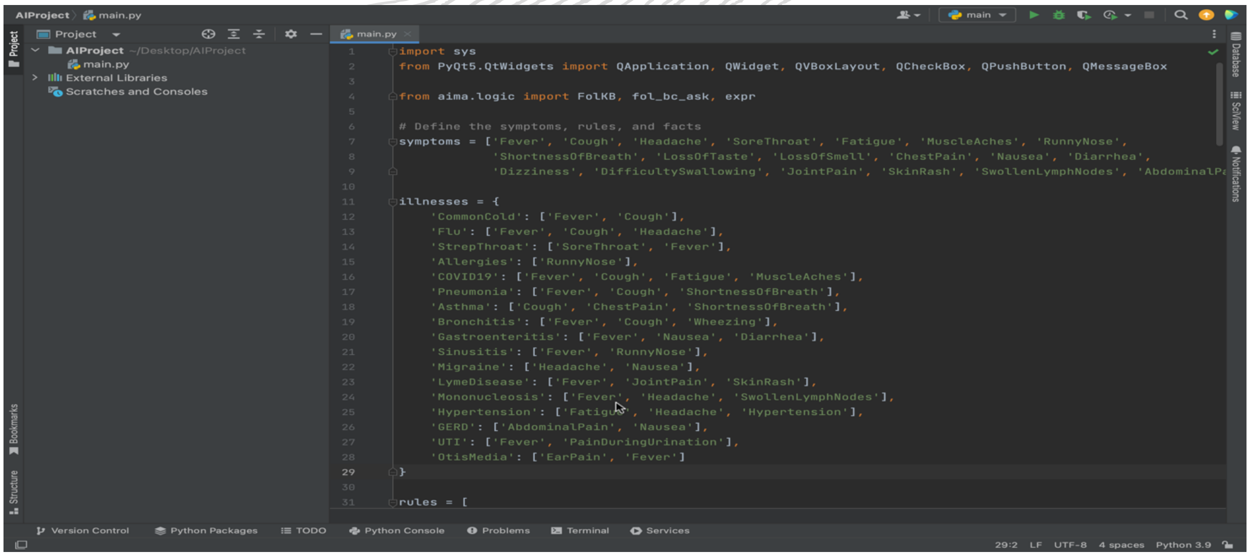
Medical diagnosis often necessitates the consideration of multiple symptoms to accurately identify potential illnesses. The Expert System addresses this complexity by:

Knowledge Base: Incorporating a repository of medical rules linking symptoms to specific illnesses.
Inference Engine: Utilizing backward chaining inference to analyze user-provided symptoms and deduce potential illnesses.
User Interface: Employing a PyQt5-based interface, as demonstrated in the provided Python code, for users to input symptoms and receive detailed diagnosis reports.
System Development:

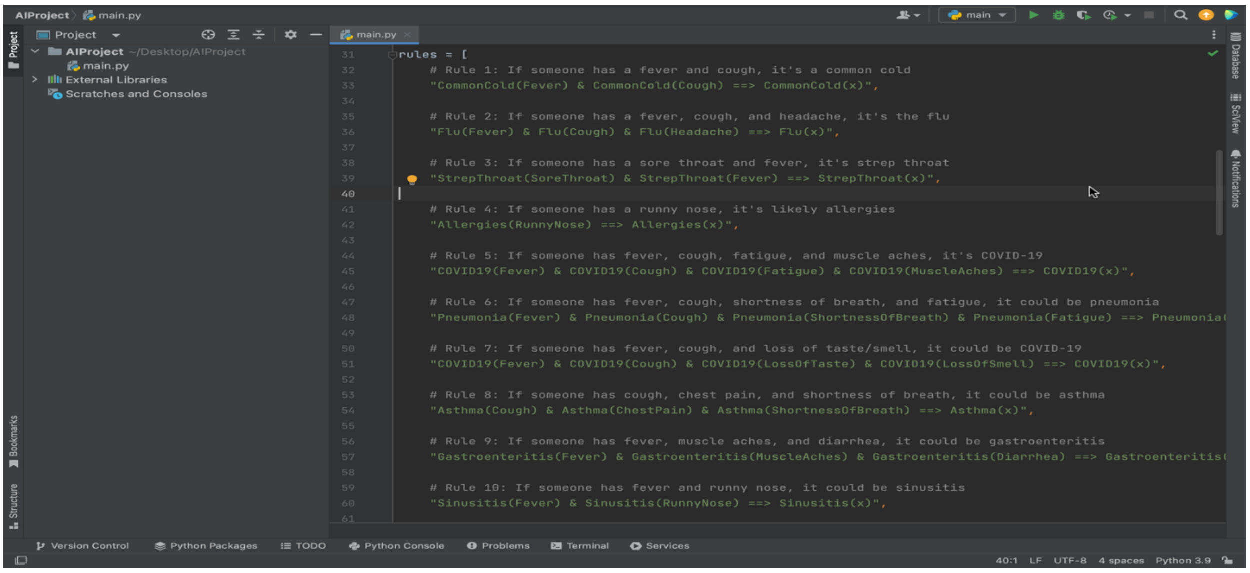
Knowledge Acquisition: Gathering symptom-illness associations from medical literature and consulting with healthcare professionals.
Knowledge Representation: Utilizing first-order logic (FOL) within the AIMA framework to encode medical knowledge systematically.
Inference Engine Selection: Utilizing backward chaining for reasoning, as demonstrated in the provided Python code, to deduce potential illnesses from symptom inputs.
Working Memory: Implementing a working memory component, as showcased in the provided code window, to store intermediate data during the inference process.
Agenda: Utilizing an agenda mechanism, as outlined in the code window, to manage diagnostic tasks systematically.
Utilizing PyQt5 to design a user-friendly interface, as depicted in the provided Python code, enabling users to input symptoms and receive clear diagnosis reports.

Conclusion:

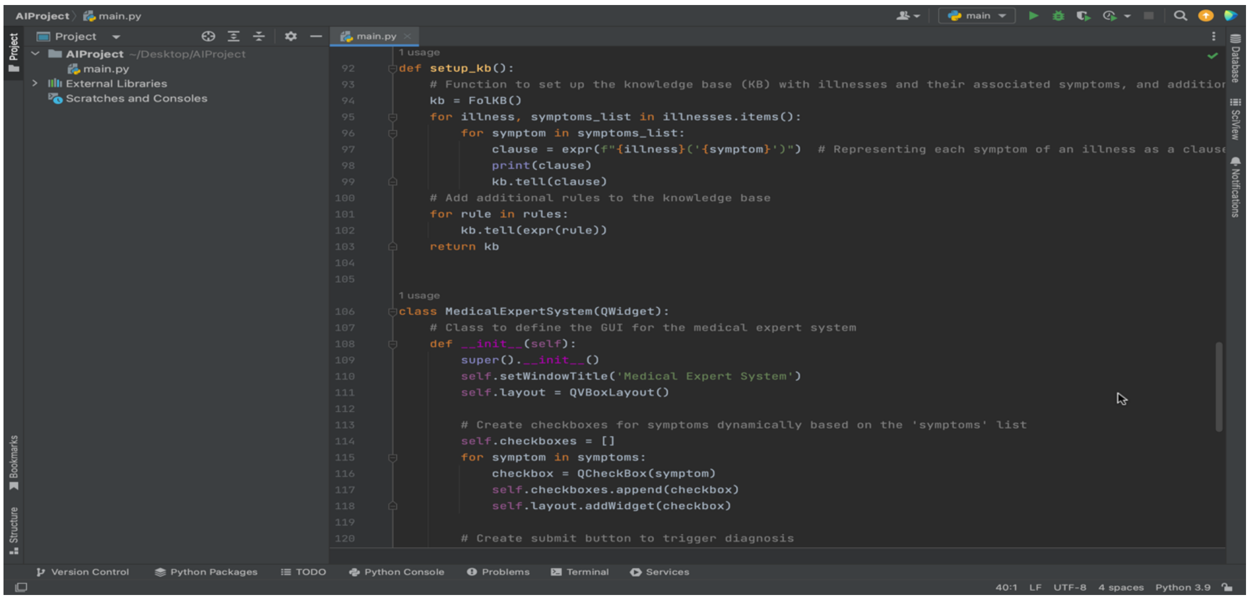
The Medical Expert System represents a valuable tool for medical professionals and individuals seeking accurate diagnoses. Through the integration of advanced inference techniques and a robust knowledge base, the system delivers personalized diagnosis reports, empowering users to make informed healthcare decisions.



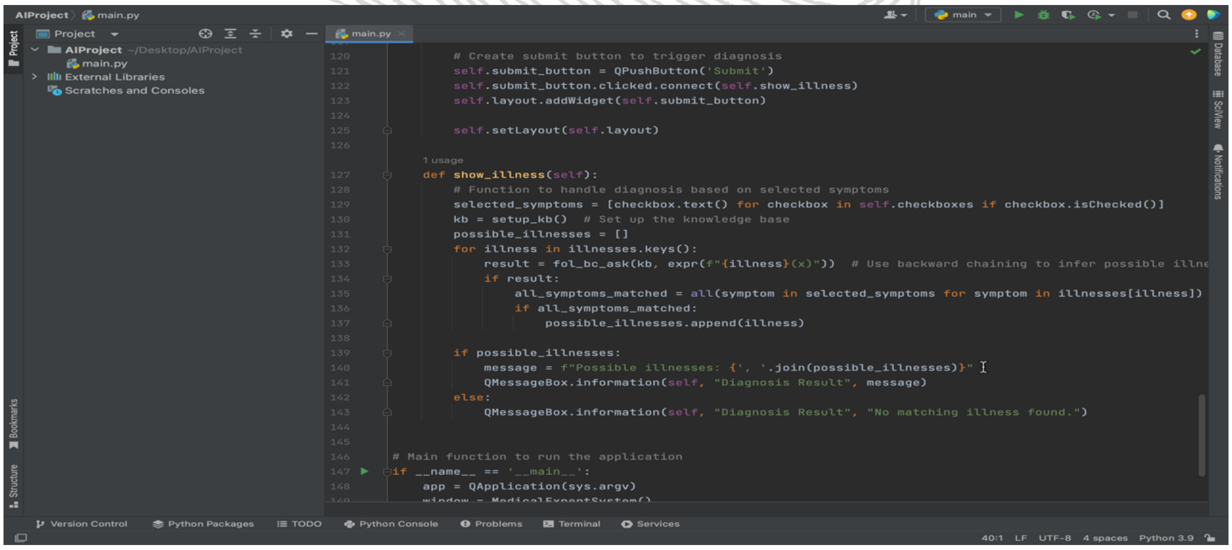
```python
import sys
from PyQt5.QtWidgets import QApplication, QWidget, QVBoxLayout, QCheckBox, QPushButton, QMessageBox

from aima.logic import FolKB, fol_bc_ask, expr

# Define the symptoms, rules, and facts
symptoms = ['Fever', 'Cough', 'Headache', 'SoreThroat', 'Fatigue', 'MuscleAches', 'RunnyNose',
            'ShortnessOfBreath', 'LossOfTaste', 'LossOfSmell', 'ChestPain', 'Nausea', 'Diarrhea',
            'Dizziness', 'DifficultySwallowing', 'JointPain', 'SkinRash', 'SwollenLymphNodes', 'AbdominalPa

illnesses = {
    'CommonCold': ['Fever', 'Cough'],
    'Flu': ['Fever', 'Cough', 'Headache'],
    'StrepThroat': ['SoreThroat', 'Fever'],
    'Allergies': ['RunnyNose'],
    'COVID19': ['Fever', 'Cough', 'Fatigue', 'MuscleAches'],
    'Pneumonia': ['Fever', 'Cough', 'ShortnessOfBreath'],
    'Asthma': ['Cough', 'ChestPain', 'ShortnessOfBreath'],
    'Bronchitis': ['Fever', 'Cough', 'Wheezing'],
    'Gastroenteritis': ['Fever', 'Nausea', 'Diarrhea'],
    'Sinusitis': ['Fever', 'RunnyNose'],
    'Migraine': ['Headache', 'Nausea'],
    'LymeDisease': ['Fever', 'JointPain', 'SkinRash'],
    'Mononucleosis': ['Fever', 'Headache', 'SwollenLymphNodes'],
    'Hypertension': ['Fatigue', 'Headache', 'Hypertension'],
    'GERD': ['AbdominalPain', 'Nausea'],
    'UTI': ['Fever', 'PainDuringUrination'],
    'OtisMedia': ['EarPain', 'Fever']
}

rules = [
```



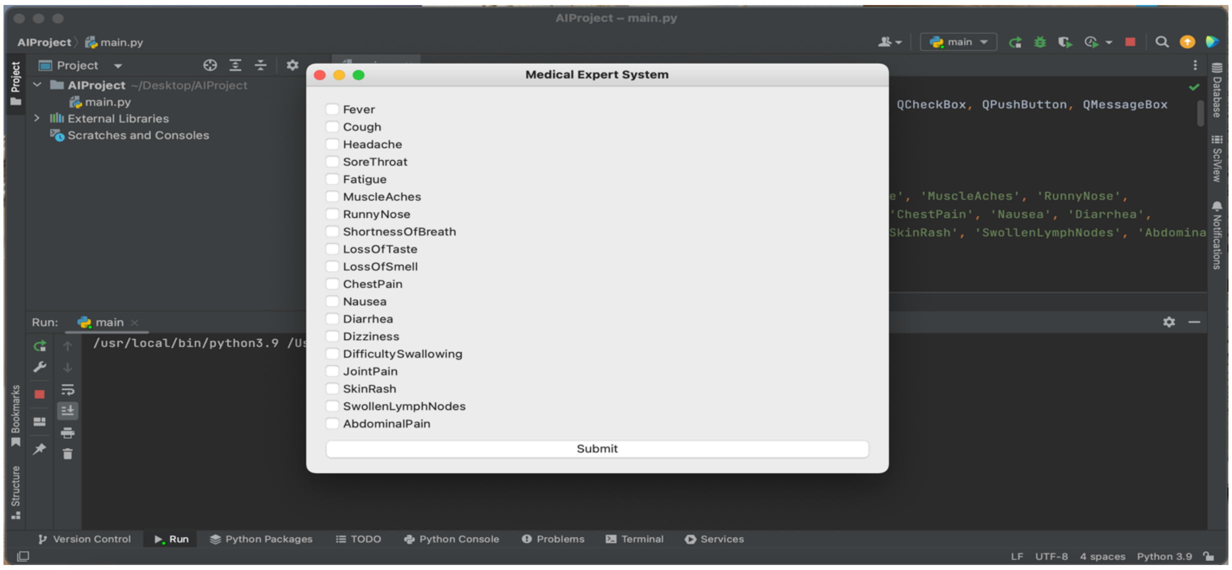
```python
rules = [
    # Rule 1: If someone has a fever and cough, it's a common cold
    "CommonCold(Fever) & CommonCold(Cough) ==> CommonCold(x)",

    # Rule 2: If someone has a fever, cough, and headache, it's the flu
    "Flu(Fever) & Flu(Cough) & Flu(Headache) ==> Flu(x)",

    # Rule 3: If someone has a sore throat and fever, it's strep throat
    "StrepThroat(SoreThroat) & StrepThroat(Fever) ==> StrepThroat(x)",

    # Rule 4: If someone has a runny nose, it's likely allergies
    "Allergies(RunnyNose) ==> Allergies(x)",

    # Rule 5: If someone has fever, cough, fatigue, and muscle aches, it's COVID-19
    "COVID19(Fever) & COVID19(Cough) & COVID19(Fatigue) & COVID19(MuscleAches) ==> COVID19(x)",

    # Rule 6: If someone has fever, cough, shortness of breath, and fatigue, it could be pneumonia
    "Pneumonia(Fever) & Pneumonia(Cough) & Pneumonia(ShortnessOfBreath) & Pneumonia(Fatigue) ==> Pneumonia(

    # Rule 7: If someone has fever, cough, and loss of taste/smell, it could be COVID-19
    "COVID19(Fever) & COVID19(Cough) & COVID19(LossOfTaste) & COVID19(LossOfSmell) ==> COVID19(x)",

    # Rule 8: If someone has cough, chest pain, and shortness of breath, it could be asthma
    "Asthma(Cough) & Asthma(ChestPain) & Asthma(ShortnessOfBreath) ==> Asthma(x)",

    # Rule 9: If someone has fever, muscle aches, and diarrhea, it could be gastroenteritis
    "Gastroenteritis(Fever) & Gastroenteritis(MuscleAches) & Gastroenteritis(Diarrhea) ==> Gastroenteritis(

    # Rule 10: If someone has fever and runny nose, it could be sinusitis
    "Sinusitis(Fever) & Sinusitis(RunnyNose) ==> Sinusitis(x)",

```



```python
1 usage
def setup_kb():
    # Function to set up the knowledge base (KB) with illnesses and their associated symptoms, and additior
    kb = FolKB()
    for illness, symptoms_list in illnesses.items():
        for symptom in symptoms_list:
            clause = expr(f"{illness}('{symptom}')")  # Representing each symptom of an illness as a clause
            print(clause)
            kb.tell(clause)
    # Add additional rules to the knowledge base
    for rule in rules:
        kb.tell(expr(rule))
    return kb


1 usage
class MedicalExpertSystem(QWidget):
    # Class to define the GUI for the medical expert system
    def __init__(self):
        super().__init__()
        self.setWindowTitle('Medical Expert System')
        self.layout = QVBoxLayout()

        # Create checkboxes for symptoms dynamically based on the 'symptoms' list
        self.checkboxes = []
        for symptom in symptoms:
            checkbox = QCheckBox(symptom)
            self.checkboxes.append(checkbox)
            self.layout.addWidget(checkbox)

        # Create submit button to trigger diagnosis
```



```
        # Create submit button to trigger diagnosis
        self.submit_button = QPushButton('Submit')
        self.submit_button.clicked.connect(self.show_illness)
        self.layout.addWidget(self.submit_button)

        self.setLayout(self.layout)

    def show_illness(self):
        # Function to handle diagnosis based on selected symptoms
        selected_symptoms = [checkbox.text() for checkbox in self.checkboxes if checkbox.isChecked()]
        kb = setup_kb()  # Set up the knowledge base
        possible_illnesses = []
        for illness in illnesses.keys():
            result = fol_bc_ask(kb, expr(f"{illness}(x)"))  # Use backward chaining to infer possible illne
            if result:
                all_symptoms_matched = all(symptom in selected_symptoms for symptom in illnesses[illness])
                if all_symptoms_matched:
                    possible_illnesses.append(illness)

        if possible_illnesses:
            message = f"Possible illnesses: {', '.join(possible_illnesses)}"
            QMessageBox.information(self, "Diagnosis Result", message)
        else:
            QMessageBox.information(self, "Diagnosis Result", "No matching illness found.")


# Main function to run the application
if __name__ == '__main__':
    app = QApplication(sys.argv)
    window = MedicalExpertSystem()
```

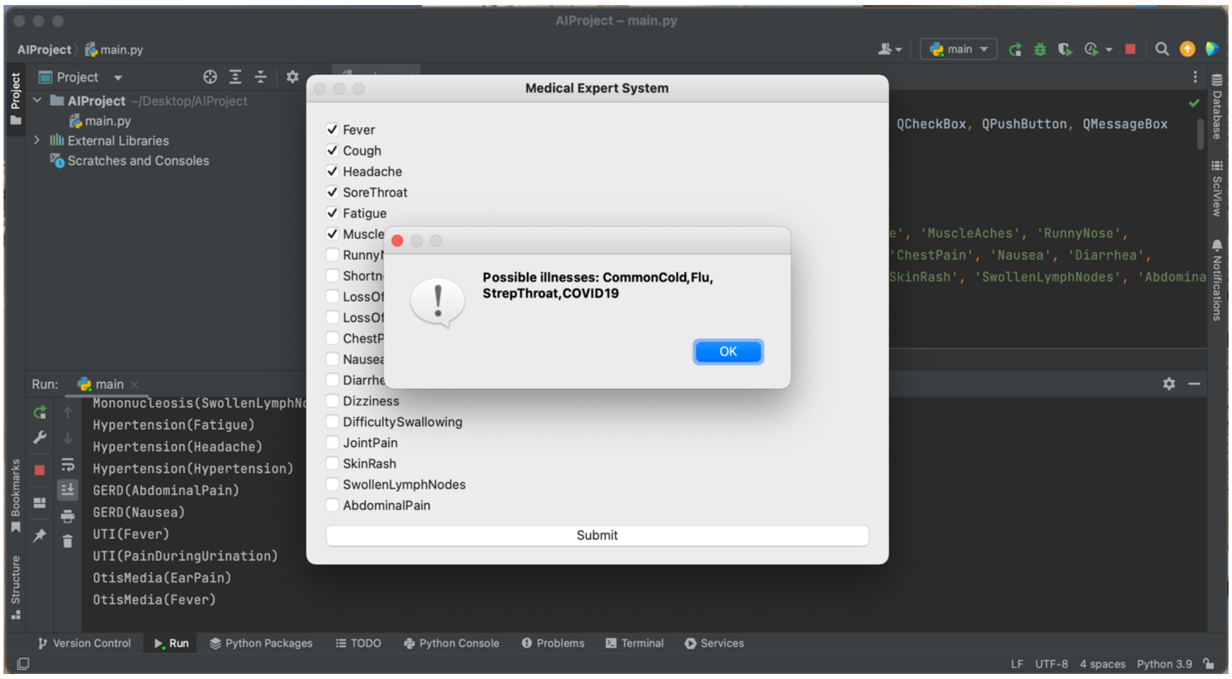


Medical Expert System

- Fever
- Cough
- Headache
- SoreThroat
- Fatigue
- MuscleAches
- RunnyNose
- ShortnessOfBreath
- LossOfTaste
- LossOfSmell
- ChestPain
- Nausea
- Diarrhea
- Dizziness
- DifficultySwallowing
- JointPain
- SkinRash
- SwollenLymphNodes
- AbdominalPain

Submit

```
/usr/local/bin/python3.9 /Us
```



Medical Expert System

- ✓ Fever
- ✓ Cough
- ✓ Headache
- ✓ SoreThroat
- ✓ Fatigue
- ✓ Muscle

Possible illnesses: CommonCold,Flu, StrepThroat,COVID19

OK

```
Mononucleosis(SwollenLymphNo
Hypertension(Fatigue)
Hypertension(Headache)
Hypertension(Hypertension)
GERD(AbdominalPain)
GERD(Nausea)
UTI(Fever)
UTI(PainDuringUrination)
OtisMedia(EarPain)
OtisMedia(Fever)
```