

# **AI Personal Health Advisor**

## **1. Title Page**

Project Title: AI Personal Health Advisor

Student Name: Kalpesh Dilip Dukhande

Course: TYIT

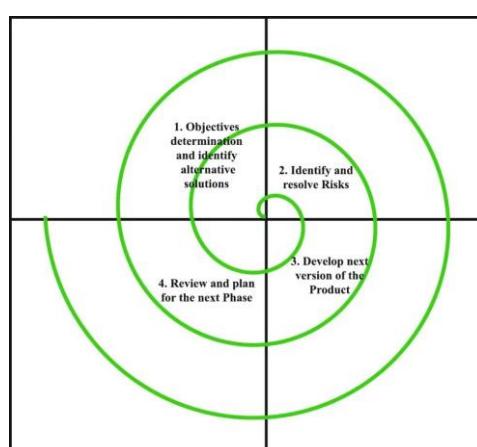
College: Abhinav Degree College

Technology Used: Python (Flask), HTML, CSS, JavaScript

## **2. Introduction**

The AI Personal Health Advisor is a smart web-based system that allows users to input their health symptoms and receive instant medical insights powered by logic-based predictions. The platform offers free features such as symptom checker, health tips, and medication reminders. Premium features include AI-driven health suggestions, doctor booking, and record management. The system ensures that users are guided toward better health management from the comfort of their home.

## **3. Software Model Used: Spiral Model**



# **AI Personal Health Advisor**

Phases in Spiral Model (Iterative with risk analysis):

## **I Planning Phases:**

- Define project goals like symptom checker, health tips, and emergency alerts.
- Decide technology stack: Flask (Python), HTML, CSS, JS.
- Finalize required and optional features.

## **II. Risk Analysis:**

- Identify issues like:
  - Incorrect symptom prediction
  - Poor user interface for patients
  - Data privacy concerns
- Plan solutions like:
  - Rule-based systems
  - Simple and responsive design
  - Basic encryption and secure routing

## **III. Engineering / Development:**

- Code core modules like symptom checker, dashboard, and alerts.
- Implement frontend pages and Flask backend routes.
- Add optional features like reminders and AI suggestions.

## **IV. Evaluation / Testing:**

- Test the system with different inputs.
- Get user feedback.
- Fix bugs and plan improvements for the next cycle.

The Spiral Model was chosen due to its flexibility, iterative development, and the ability to incorporate changes at each phase based on testing or feedback.

## AI Personal Health Advisor

### 4. Front End & Back End

Frontend:

- HTML5 for structure
- CSS3 for styling
- JavaScript for dynamic behavior
- Jinja2 templating for dynamic page rendering
- Python with Flask framework
- Uses Jinja to pass dynamic content to HTML

Backend:

- Handles routing, form data, and logic processing

### 5. Key Points of Project (Working)

- Symptom Checker: User inputs symptoms and gets probable conditions.
- Medicine Reminder (planned/optional): Let users set reminders.
- Premium Features:
  - AI-based suggestions (rule-based)
  - Secure health record uploads (to be implemented)
  - Doctor booking placeholder
- Dashboard: Placeholder for charts and personal insights
- Secure routing for features like signup and premium access

## **6. Advantages & Disadvantages**

Advantages:

- Easy access to preliminary health advice
- Reduces minor doctor visits
- Expandable for real-world integrations
- Clean UI with responsive design

Disadvantages:

- Not a replacement for actual diagnosis
- Limited to logic-based predictions (not ML-backed)
- No real-time doctor support or authentication system yet

## **7. References**

- Flask Official Documentation: <https://flask.palletsprojects.com/>
- W3Schools: <https://www.w3schools.com/>
- Font Awesome Icons: <https://fontawesome.com/>
- Bootstrap CSS (optional future use): <https://getbootstrap.com/>
- GitHub open-source health projects