AI-powered medical search engine

Senior Project

Primary Advisor: **< Dr. Nosheen Sabahat >**

Secondary Advisor: **< Rehma >**

Presented by:

Student Reg# 261943888 **Student Name:** Abdul Hanan

Department of Computer Science

Forman Christian College (A Chartered University)

AI-powered medical search engine

By

Asher David (261947376)

Talha Saleem (261940993)

Abdul Hanan (261943888)

Project submitted to

Department of Computer Science,

Forman Christian College (A Chartered University),

Lahore, Pakistan.

in partial fulfillment of the requirements for the degree of

BACHELOR OF SCIENCE
IN
COMPUTER SCIENCE (Honors)

Primary Project Advisor

Secondary Project Advisor

Senior Project Management
Committee Representative

Abstract

The AI-powered medical search engine project addresses the need for accurate, accessible, and reliable medical information for individuals who may not have immediate access to healthcare professionals. This project leverages advanced AI algorithms to analyze user queries and provide precise medical guidance, reducing the risks associated with misinformation and misinterpretation found in general online resources. This document outlines the project's objectives, system design, test specifications, and future work, demonstrating the efficacy and potential impact of the AI-powered medical search engine in improving healthcare accessibility and outcomes.

Acknowledgement

We would like to express our deepest gratitude to our primary advisor, Dr. Nosheen Sabahat, and our secondary advisor, Rehma, for their invaluable guidance and support throughout this project. Their expertise and encouragement were crucial in helping us bring this project to fruition.

We also extend our heartfelt thanks to our families and friends for their unwavering support and understanding during the many hours we invested in this project. Additionally, we are grateful to the Department of Computer Science at Forman Christian College for providing us with the resources and environment necessary to complete our work.

Lastly, we acknowledge the contributions of our classmates and peers, whose feedback and assistance were greatly appreciated. Without the collective effort and support of all these individuals, this project would not have been possible.

List of Figures

List of Tables

Contents

Chapter 1: Introduction

1.1 Introduction

In our daily fast-paced lives, most individuals do not have the luxury of time to go see a doctor for their illness, and another problem that has emerged in recent years is that there has been a shortage of healthcare workers. Due to these problems, many Individuals are increasingly turning to online resources for medical information and guidance. However, accessibility and credibility of such information remain major obstacles, frequently leading to confusion, misinformation, and mediocre healthcare outcomes, all of which can have dire consequences on the wellness of individuals. The creation of an AI-powered medical search engine aims to overcome these issues, allowing users to receive accurate, concise, and understandable solutions.

Many people in their lives are not able to find time to visit a doctor so they turn to online resources and while it may sometimes provide them with proper guidance but often it gives them vague or incorrect information, this is dangerous as it can lead to further health complications. They can also misinterpret the information and then spread the misinformation to their friends, family, and coworkers.

Another problem that is quite evident in 3rd world countries is cost, many people can not to afford go to the doctor for every ailment they might have so they just use medicines that their friends and family have told them work for them. While this may work for some people, that is not true for all people. They might take medication with more or less dose than the required amount.

Our AI-powered medical search engine aims to solve this problem by providing the people with a quick, accurate, reliable, easy to understand, and a free way to get medical information. Our mission is to help stop the spread of misinformation about medicine.

1.2 Objectives

The project's goal is to create an application that provides users with individualized information based on their personal data while also serving as a trustworthy and simple tool for symptom-based medical diagnosis. In addition, the application will monitor a user's medical history, recommend medication, and provide information on possible side effects. To effectively identify probable diseases, the software searches a massive collection of medical information. The fundamental goal of this app is to provide easy access to medical knowledge by reducing the transmission of erroneous information and making it available to people of various backgrounds.

The project's main goal is to create a cutting-edge web and mobile application that functions as a thorough and approachable tool for symptom-based medical diagnosis and health management. By giving users easy access to trustworthy medical information that is catered to their unique needs and traits, the app seeks to empower its users.

The ability of the app to do symptom based medical diagnosis, which analyzes user-provided symptoms and generates potential diagnoses using complex algorithms and machine learning techniques, lies at the heart of its functioning. Demographic data like age, gender, and medical history are included into the app to provide individualized insights and recommendations that improve the app's diagnostic capabilities' relevance and accuracy.

The app will act as a single system for managing users' medical histories in addition to aiding in diagnosis. Users will be able to track and safely retain health-related data, such as past prescription history, diagnoses, and treatments. This benefit of keeping thorough medical records not only encourages active control of one's health but also makes it easier to have more educated conversations with medical providers.

1.3 Problem Statement

The goal of our project is to create a medical search engine and diagnostic tool driven by AI. By entering their symptoms, users of this software will be able to receive potential diagnoses as well as individualized information based on demographic data. They will also be given a list of research the AI used for the diagnosis. Additionally, the tool will provide medication recommendations and maintain a user's medical history.

It will address the following problems:

1. It will allow people to quickly check any medical queries they have.
2. Allow people to get quick, accurate, and reliable medical information.
3. Give the information in an easy to consume and understand manner.

1.4 Scope

This Software Requirements Specification (SRS) covers the development of the AI-powered medical search engine and diagnosis tool, including its features, functionalities, and limitations.

This Software Requirements Specification (SRS) defines the scope of the AI-powered medical search engine and diagnosis tool project. It encompasses the development, features, functionalities, and limitations of the software, including:

1) **Feature Set:** Definition of core features such as symptom input, diagnosis generation, personalized insights, medication recommendations, and medical history maintenance.
2) **Functionality:** Description of essential functionalities, including symptom analysis, AI-driven diagnosis, personalized insights, medication recommendations, and secure medical history tracking.
3) **User Interface:** Requirements for a user-friendly and intuitive interface across different devices, covering symptom input, diagnosis presentation, medication recommendations, and medical history tracking.
4) **Integration:** Specifications for integration with external databases, AI algorithms, and medical resources to ensure accuracy and reliability.
5) **Scalability:** Considerations for handling increasing user demand, supporting future enhancements, and maintaining a growing database of medical information.

6) **Limitations:** Acknowledgment of limitations, such as reliance on user-provided data and the necessity for consulting healthcare professionals for definitive advice.
7) **Compliance:** Requirements for adherence to relevant regulations, standards, and best practices in healthcare information privacy, data security, and ethical AI use.
8) **Documentation:** Needs for user manuals, technical specifications, API documentation, and release notes to facilitate understanding and maintenance.
9) **Testing:** Testing requirements, including unit testing, integration testing, system testing, and user acceptance testing, to ensure reliability and usability.
10) **Deployment:** Considerations for deployment environments, installation procedures, configuration settings, and support mechanisms.
11) **Maintenance:** Requirements for bug fixes, feature enhancements, updates, and technical support to ensure ongoing usability and reliability.

Chapter 2: Requirements Analysis

2.1 Literature Review

The application of artificial intelligence (AI) to medical diagnostics has become a revolutionary approach in the field of healthcare, allowing for the precise and effective diagnosis of ailments. A study done by (Zeltzer et al., 2023) demonstrated that AI can exhibit high diagnostic accuracy, with consistent performance across different patient demographics and presenting symptoms.

WebMD's Symptom Checker offers a user-friendly interface for individuals to input their symptoms and receive information about potential health conditions. While it is a useful tool for initial information, it is not a substitute for expert medical guidance or diagnosis. Users should consult healthcare professionals for accurate recommendations, as it may not always provide personalized advice and carries the risk of misinterpretation leading to incorrect self-diagnosis or unnecessary concern.

WebMD's Symptom Checker, despite its utility, has its limits. It primarily relies on user-entered information, which may lead to inaccurate results if symptoms are not thoroughly described. Additionally, it does not ask for major contributing factors to an individual's health, such as race, region of birth, etc. A study showed that the first diagnosis was only correct for 26% of the cases and that the top 3 diagnoses were correct 36% of the time (Seminara, 2019). Users should be aware that the tool provides general information and not personalized medical advice, and it cannot consider individual medical histories. Thus, while WebMD's Symptom Checker serves as a useful informational tool, users need to consult with healthcare professionals for accurate diagnoses and advice tailored to their specific circumstances.

2.2 User Classes and Characteristics

- **General Users:**
  - Varied demographic backgrounds and levels of health literacy.
  - Seek short and reliable information approximately health checkups.
  - May have specific health issues or general fitness maintenance needs.
  - Require user-pleasant interfaces and clean, comprehensible information.
- **Healthcare Professionals:**
  - Have medical training and expertise.
  - Seek authoritative and evidence-primarily based data.
  - Use the search engine for professional research, patient education, or to stay updated on health trends.
  - May require advanced seek features and access to medical literature and journals.
- **Health Enthusiasts:**
  - Proactively in involved in health and wellness.
  - May have a deeper understanding of medical terminology and procedures.
  - Seek detailed information about different types of health checkups.
  - Value accuracy, comprehensiveness, and the latest medical research.
- **Doctor**:

  Technical Skill: Moderate AI and medical informatics.

  Frequency of Use: Regularly for patient care.

  Education Level: Medical degree.

  Privileges: Access to patient data and diagnostic tools.

  Experience with System: Proficient.

  Product Functions: Conduct health assessments, review patient data, prescribe treatments.

- **Patient:**

  Technical Skill: Basic internet literacy.

  Frequency of Use: Periodically for health inquiries.

  Education Level: Varied.

  Privileges: Access to personal health information.

  Experience with System: Mixed proficiency.

  Product Functions: Input health concerns, receive personalized recommendations, access educational content.

2.3 Design and Implementation Constraints

1. **Data Privacy Regulations**: Compliance with HIPAA and GDPR limits get entry to to sensitive patient statistics, requiring strong safety features and records encryption.

2. **Limited Training Data:** Availability of high-quality data for AI algorithms may be restrained, affecting the accuracy and effectiveness of search outcomes.

3. **Integration with Electronic Health Records (EHR):** Ensuring seamless integration with various EHR structures poses technical demanding situations and calls for adherence to interoperability standards.

4. **Ethical Considerations:** Balancing using AI for health checkups with moral standards inclusive of knowledgeable consent, equity, and transparency offers design dilemmas.

5. **Language and Cultural Diversity:** Accommodating diverse languages and cultural nuances in health inquiries necessitates complete linguistic and cultural fashions for accurate outcomes.

6. **Resource Constraints:** Limited computational assets and bandwidth may also impact the rate and scalability of the search engine, requiring optimization techniques.

7. **Accuracy and Reliability:** Ensuring the AI-powered seek engine offers correct and reliable health information requires continuous validation and refinement of algorithms.

8. **User Interface Accessibility:** Designing an intuitive and available user interface for people with varying tiers of virtual literacy and disabilities is important for inclusivity and usability.

2.4 Assumptions and Dependencies

1. **Quality of Data:** The effectiveness of the AI system heavily relies upon the satisfaction and reliability of the input data.

Assumption: The health checkup search engine has access to a widespread and various dataset of accurate medical records, including symptoms, diagnoses, remedies, and consequences.

2. **Medical Expertise:** While AI algorithms can assist in offering recommendations and insights, they may not be an alternative to professional scientific recommendation. Dependency: Users must acknowledge that

the search engine serves as a tool to complement, not replace, consultation with certified healthcare specialists.

3. **Regulatory Compliance:** Compliance with healthcare guidelines and privacy laws is crucial.

Assumption: The project adheres to applicable rules inclusive of HIPAA (in the United States) or GDPR (in the European Union), ensuring the security and confidentiality of affected patient data.

4. **User Accessibility:** The search engine should be user-friendly and accessible to individuals with various ranges of technical proficiency.

Assumption: The interface is designed with intuitive navigation and clear instructions, catering to both clinicians and sufferers.

5. **Continuous Learning:** AI models require continuous schooling and updating to stay relevant and accurate.

Dependency: There should be mechanisms in place for ongoing facts series, model refinement, and feedback integration from customers and medical experts.

6. **Integration with Healthcare Systems:** For clinicians, integration with current electronic health record (EHR) systems is important for seamless workflow integration.

Assumption: The search engine affords APIs or interoperability features to connect with EHR systems, permitting smooth get right of entry to patient data and incorporation of search outcomes into patient records.

7. **Hardware and Infrastructure:** The performance of the AI algorithms can be depending on the availability of robust hardware and infrastructure for processing massive amounts of data.

Assumption: Sufficient computational assets are allotted for running AI models efficiently and dealing with user queries in real-time.

8. **Ethical Considerations:** The project needs to prioritize moral ideas along with equity, transparency, and accountability in its development and deployment.

Assumption: Measures are taken to mitigate biases within the AI algorithms and ensure transparency in how recommendations are generated.

9. **User Trust and Adoption:** Success depends on constructing belief amongst customers, together with clinicians and patients, regarding the accuracy and reliability of the search engine's recommendations.

Assumption: Through powerful verbal exchange and demonstration of value, users will undertake and continue to use the system for their health checkup needs.

2.5 Functional Requirements

2.5.1 User Login

Identifier	User Login	
Purpose	To personalized content and functionalities within a system while ensuring security and accountability.	
Priority	High	
Pre-conditions	User must have a valid and already registered username and password.	
Post-conditions	The application will display a sign in success message. And user will be directed to dashboard.	
Typical Course of Action		
S#	**Actor Action**	**System Response**
1	User navigates to the login page.	System displays the login form
2	User enters their username and password.	The system validates the credentials entered and prompts the user to select their role: patient or doctor.
3	User selects their role by clicking the respective option.	System verifies the role selection and grants access accordingly.
4	User submits the login form.	System validates the credentials and logs the user to the main page of the AI search engine.
Alternate Course of Action		
S#	**Actor Action**	**System Response**
	N/A	**N/A**

Table 2.1: User Login

2.5.2 Search

Identifier	Search	
Purpose	To provide users with relevant and contextual information based on their search queries using AI-powered capabilities.	
Priority	High	
Pre-conditions	User must be logged into the system.	
Post-conditions	User receives a text-based response generated by AI, providing information or guidance related to their query.	
Typical Course of Action		
S#	**Actor Action**	**System Response**
1	User enters a medical query or symptoms in the search bar.	System validates the query and triggers the AI-powered search algorithm.
2	**N/A**	System processes the query using AI algorithms, including natural language understanding and generation.
3	**N/A**	System generates a text-based response that provides information.
4	User reads and interacts with the text-based response.	System may offer additional options or prompts for the user to explore related topics.
Alternate Course of Action		
S#	**Actor Action**	**System Response**
1	User enters a vague or ambiguous query.	System prompts the user for clarification or provide general information related to the query.

Table 2.2: Search

2.5.3 Monitor System Performance

Identifier	Monitor System Performance	
Purpose	To continuously monitor the performance of the system to ensure optimal functioning and identify potential issues or bottlenecks.	
Priority	High	
Pre-conditions	Monitoring tools or software must be deployed and configured within the system.	
Post-conditions	Real-time and historical data on system performance metrics are available for analysis and troubleshooting.	
Typical Course of Action		
S#	**Actor Action**	**System Response**
1	System administrators access the monitoring dashboard or interface.	System gather and displays real-time performance metrics in a user-friendly format, such as graphs, charts, or tables.
2	Administrators analyze performance trends and identify any deviations from normal behavior.	System highlights anomalies or triggers alerts based on predefined conditions.
3	Administrators investigate the root cause of performance issues if detected.	System provides diagnostic tools and detailed logs to facilitate troubleshooting.
4	Administrators take corrective actions to optimize system performance.	System may suggest adjustments to configuration settings, resource allocation, or software updates.
Alternate Course of Action		
S#	**Actor Action**	**System Response**
1	Administrators encounter difficulties accessing the monitoring interface.	System provides alternative access methods or performs automatic failover to redundant monitoring systems.

Table 2.3: Monitor System Performance

2.5.4 Patient Education

Identifier	Patient Education	
Purpose	To provide users with detailed and contextual medical information, including related research and past cases, based on their queries or symptoms, leveraging AI-powered capabilities.	
Priority	Medium	
Pre-conditions	User has entered a query, symptoms, or topic of interest.	
Post-conditions	User receives a descriptive and comprehensive answer with related research and past cases by clicking a button associated with their query or topic of interest.	
Typical Course of Action		
S#	**Actor Action**	**System Response**
1	User clicks the "View Details" button associated with the summarized response.	System retrieves and displays a detailed answer with comprehensive information, including related research studies, clinical trials, and past cases relevant to the query.
Alternate Course of Action		
S#	**Actor Action**	**System Response**
1	User encounters difficulties clicking the "View Details" button.	System provides troubleshooting steps.

Table 2.4: Patient Education

2.5.5 Drug Interaction Checker

Identifier	Drug Interaction Checker	
Purpose	To allow users to check for potential interactions between medications to ensure safe usage.	
Priority	High	
Pre-conditions	User clicks on the "Drug Interaction Checker" button.	
Post-conditions	User receives information about potential interactions between entered medications.	
Typical Course of Action		
S#	**Actor Action**	**System Response**
1	User clicks on the "Drug Interaction Checker" button.	System opens a prompt or dialog box for the user to enter the names or dosages of medications they are currently taking.
2	User enters the names or dosages of medications they are currently taking.	System validates and processes the entered medications and presents the user with a summary of potential interactions.
Alternate Course of Action		
S#	**Actor Action**	**System Response**
1	User enters incorrect or incomplete information about their medications.	System prompts the user to verify and correct the entered information.

Table 2.5: Drug Interaction Checker

2.5.6 NLP (Natural Language Processing)

Identifier	Natural Language Processing	
Purpose	To apply natural language processing techniques for enhancing user search experience and extracting relevant information from medical records.	
Priority	High	
Pre-conditions	User has access to the search interface.	
Post-conditions	User receives relevant search results based on their query.	
Typical Course of Action		
S#	**Actor Action**	**System Response**
1	User enters a search query in natural language.	System applies NLP algorithms to understand the user query and extract key concepts and retrieve the relevant information.
2	User interacts with the search results and refines the query if needed.	System may offer suggestions or refinements based on NLP analysis of user interactions.
Alternate Course of Action		
S#	**Actor Action**	**System Response**
1	User enters a vague or ambiguous query.	System prompts the user for clarification or suggests related topics based on context.

Table 2.6: Natural Language Processing

2.6 Use Case Diagram

2.6.1 Use Case Diagram (Patient)

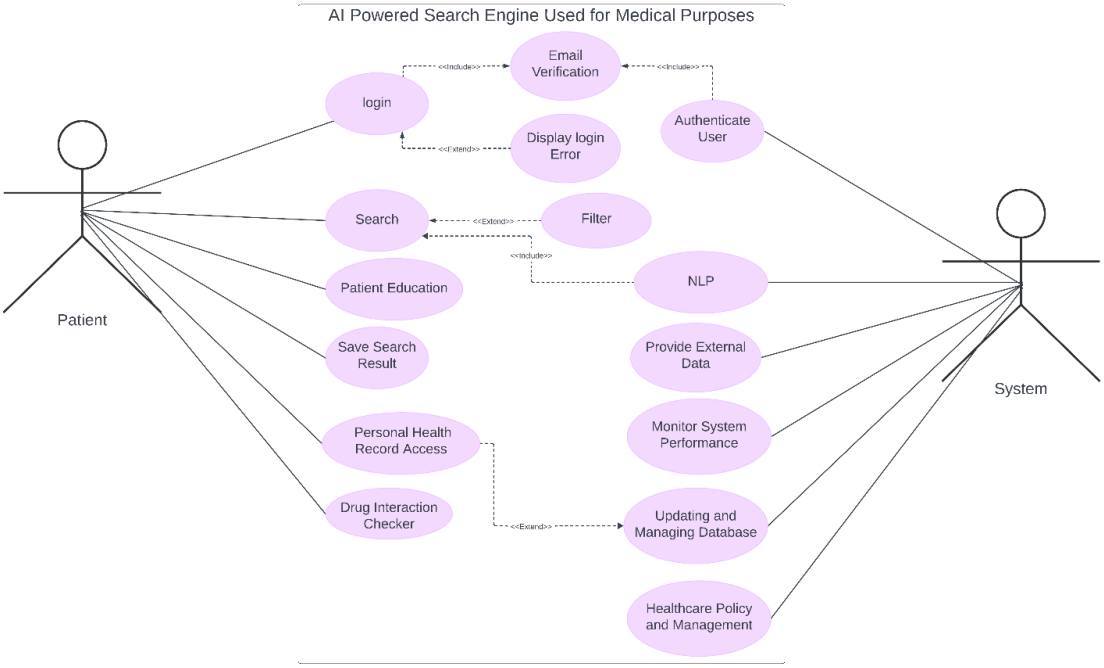

Figure 2.1 Use Case Diagram (Patient)

2.6.2 Use Case Diagram (Doctor)

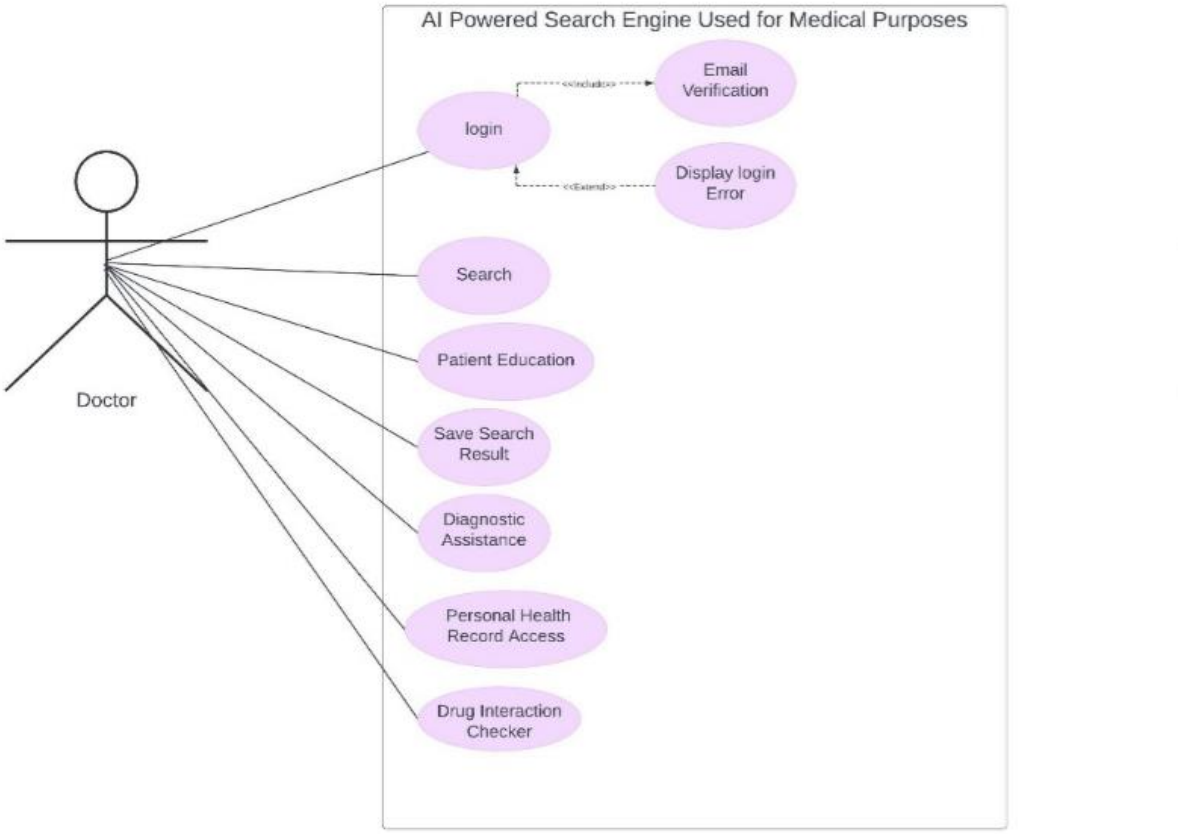

Figure 2.2 Use Case Diagram (Doctor)

2.7 Nonfunctional Requirements

2.7.1 Performance Requirements

Response Time:

The search engine ought to supply search consequences with a reaction time of below 5 seconds from the user query submission, ensuring set off gets right of entry to health-associated records.

Concurrent Users:

The search engine must guide no less than 1000 concurrent users while retaining superior response times and device balance.

Caching and Optimization:

- Implement caching mechanisms to store regularly accessed search results and reduce database queries.

- Use techniques consisting of query optimization and indexing to enhance the speed of search operations.

- Regularly monitor cache hit rates and optimize caching strategies to ensure maximum performance advantages.

Network Performance:

- The system ought to be optimized to limit network latency, especially for customers having access to the search engine from exclusive geographical locations.

- Content delivery networks (CDNs) or side caching should be utilized to serve static assets and decrease latency for worldwide users.

- Network performance metrics ought to be monitored continuously to become aware of and address any latency troubles.

2.7.2 Safety Requirements

Data Integrity: Ensure the accuracy and consistency of health information supplied to customers.

Reliability: The system need to always supply accurate and reliable outcomes to customers.

Privacy Protection: Safeguard sensitive user health information by implementing robust data encryption and access controls.

Error Handling: Provide mechanisms to discover and handle errors gracefully to prevent incorrect information or system failures.

Regulatory Compliance: Adhere to relevant healthcare guidelines and standards to make sure the legality and ethical use of health data.

2.7.3 Security Requirements

Data Confidentiality: Implement measures to save you unauthorized access to user health records and search records.

Authentication: Utilize secure authentication methods which includes multi-aspect authentication to affirm the identity of users having access to the system.

Authorization: Enforce access control regulations to restrict user access to sensitive health records primarily based on their roles and permissions.

Vulnerability Management: Regularly assess and mitigate safety vulnerabilities to protect against potential cyber threats.

2.7.4 Additional Software Quality Attributes

Compatibility: Ensure compatibility with extensive variety of devices, browsers, and operating systems to maximize accessibility for users regardless of their preferred platform.

Adaptability: Incorporate mechanisms to evolve to adjustments in healthcare recommendations, regulations, and user requirements over time, ensuring the search engine stays relevant and powerful within the ever-evolving healthcare panorama.

Testability: Implement robust trying out techniques, together with unit tests, integration tests, and user acceptance tests, to validate the functionality, reliability, and accuracy of the search engine's functions and algorithms.

Documentation: Provide complete documentation, consumer publications, and guide materials to assist users in successfully using the search engine and understanding its features and capabilities.

2.8 Other Requirements

This section outlines additional requirements not addressed elsewhere in the Software Requirements Specification (SRS), encompassing various aspects crucial for the functionality, usability, and compliance of the system.

1. **Database Requirements:**

    - The system shall utilize a relational database management system (RDBMS) to store user data, medical records, diagnostic information, and research references.
    - Database architecture shall be designed for efficient data storage, retrieval, and management, ensuring scalability and data integrity.
    - Regular data backups and disaster recovery procedures shall be implemented to mitigate the risk of data loss and ensure system reliability.

2. **External Interface Requirements:**

    - The system shall integrate seamlessly with external APIs or web services to access medical research databases, drug databases, and other relevant healthcare resources.
    - Compatibility with a variety of devices and platforms, including desktop browsers, mobile devices, and tablets, shall be ensured to provide a consistent user experience across different environments.
    - Communication protocols and data exchange formats shall comply with industry standards to facilitate interoperability with external systems and services.

3. **Internationalization Requirements:**

    - The system shall support multilingual functionality, enabling users from diverse linguistic backgrounds to access and interact with the platform in their preferred language.
    - Localization settings for date and time formats, currency symbols, and other locale-specific elements shall be customizable to accommodate regional preferences and conventions.

4. **Legal Requirements:**

- Compliance with relevant laws and regulations governing the storage, processing, and transmission of healthcare data, such as HIPAA (Health Insurance Portability and Accountability Act) in the United States, shall be ensured.
- Transparent privacy policies, terms of service, and data protection measures shall be provided to users, with mechanisms for obtaining consent and managing data sharing preferences in accordance with applicable regulations.

5. **Reuse Objectives:**

- The project shall adhere to modular design principles and coding standards to facilitate code reuse, maintainability, and extensibility.
- Common functions, algorithms, and components shall be modularized, documented, and organized into reusable libraries or modules to streamline future development efforts and enhance system scalability.

Chapter 3: System Design

3.1 Application and Data Architecture

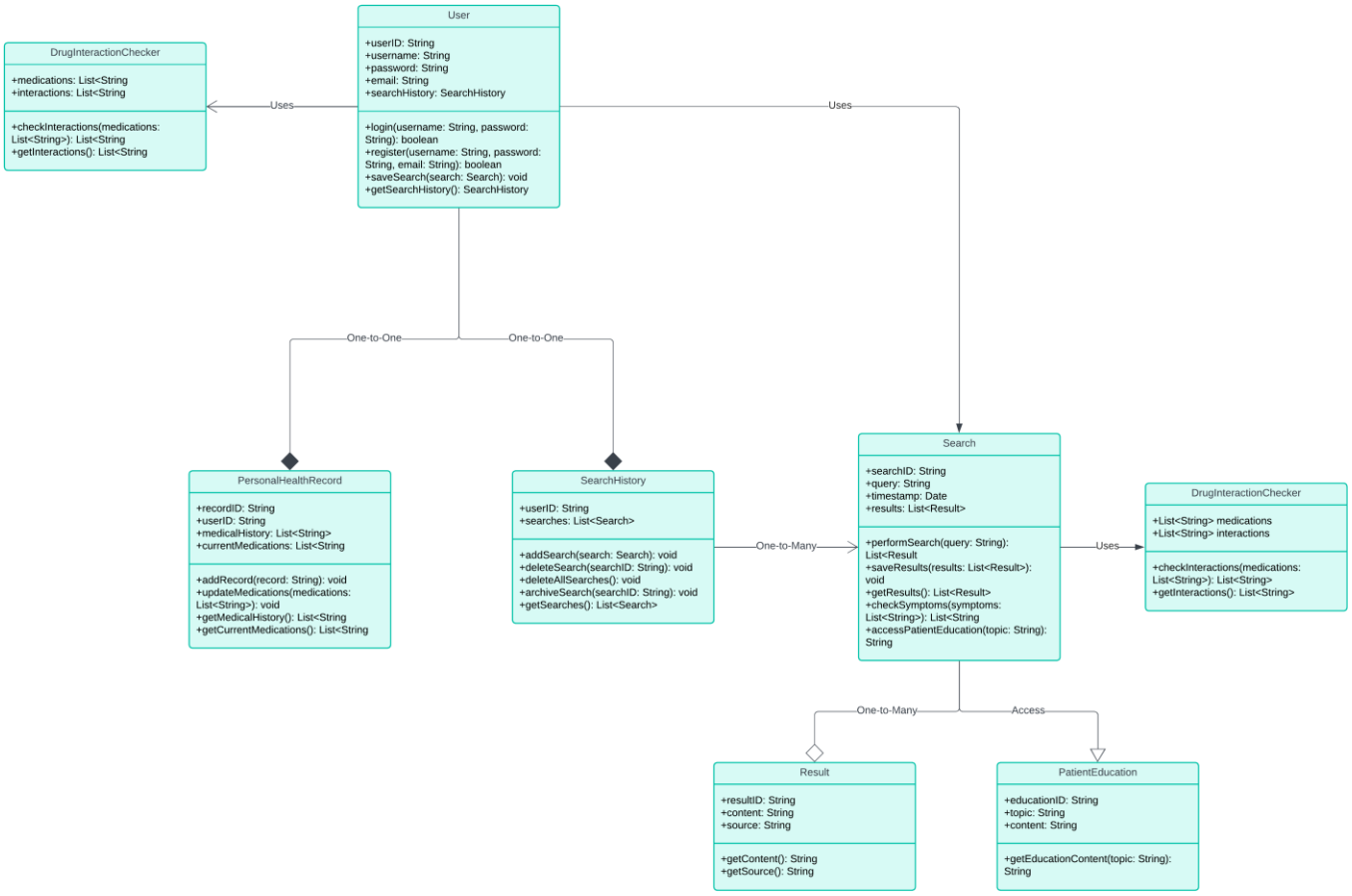

Figure 3.1 Class Diagram

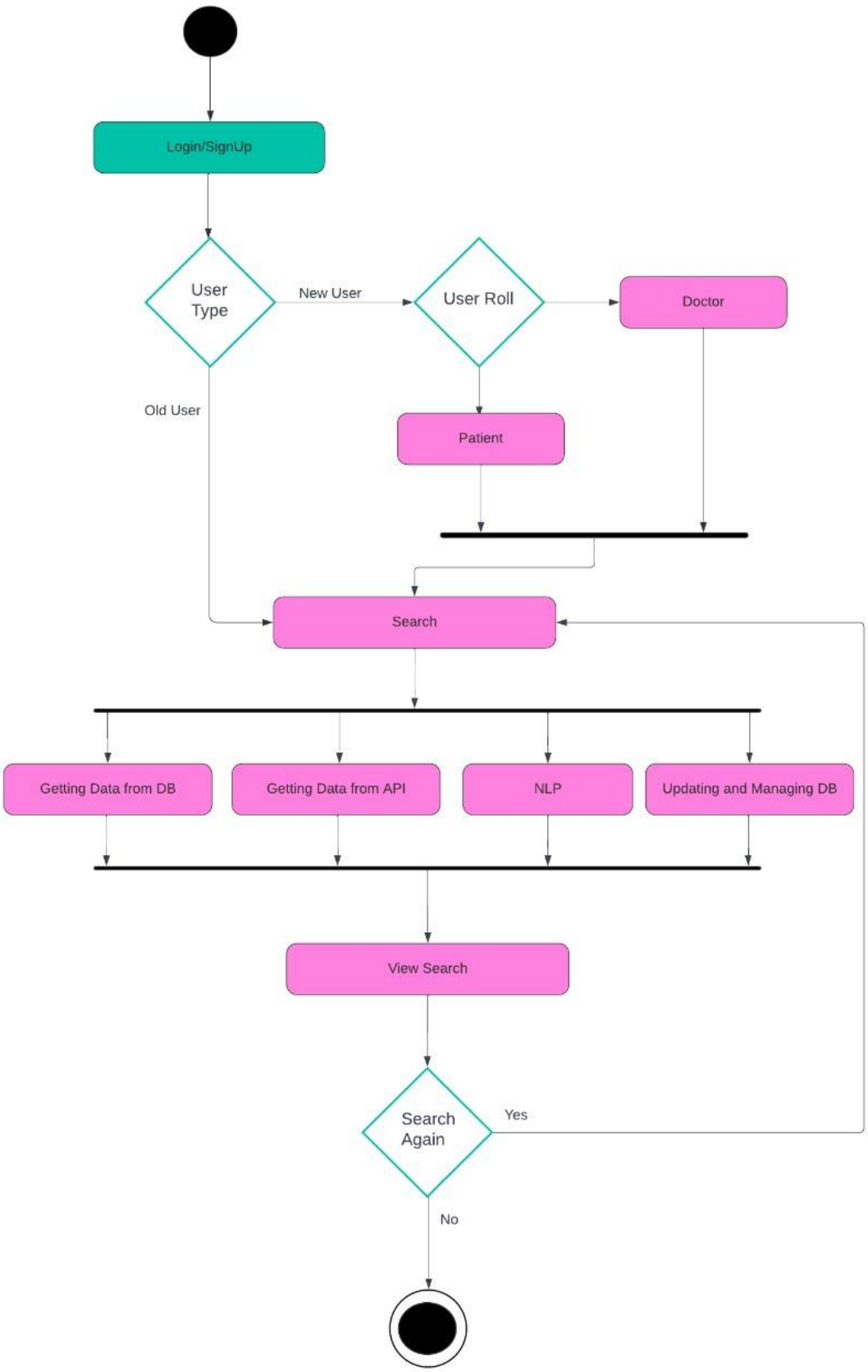

Figure 3.2 Activity Diagram (Search)

3.2 Component Interactions and Collaborations

Usecase: Personal health record access.
Actor: Patient

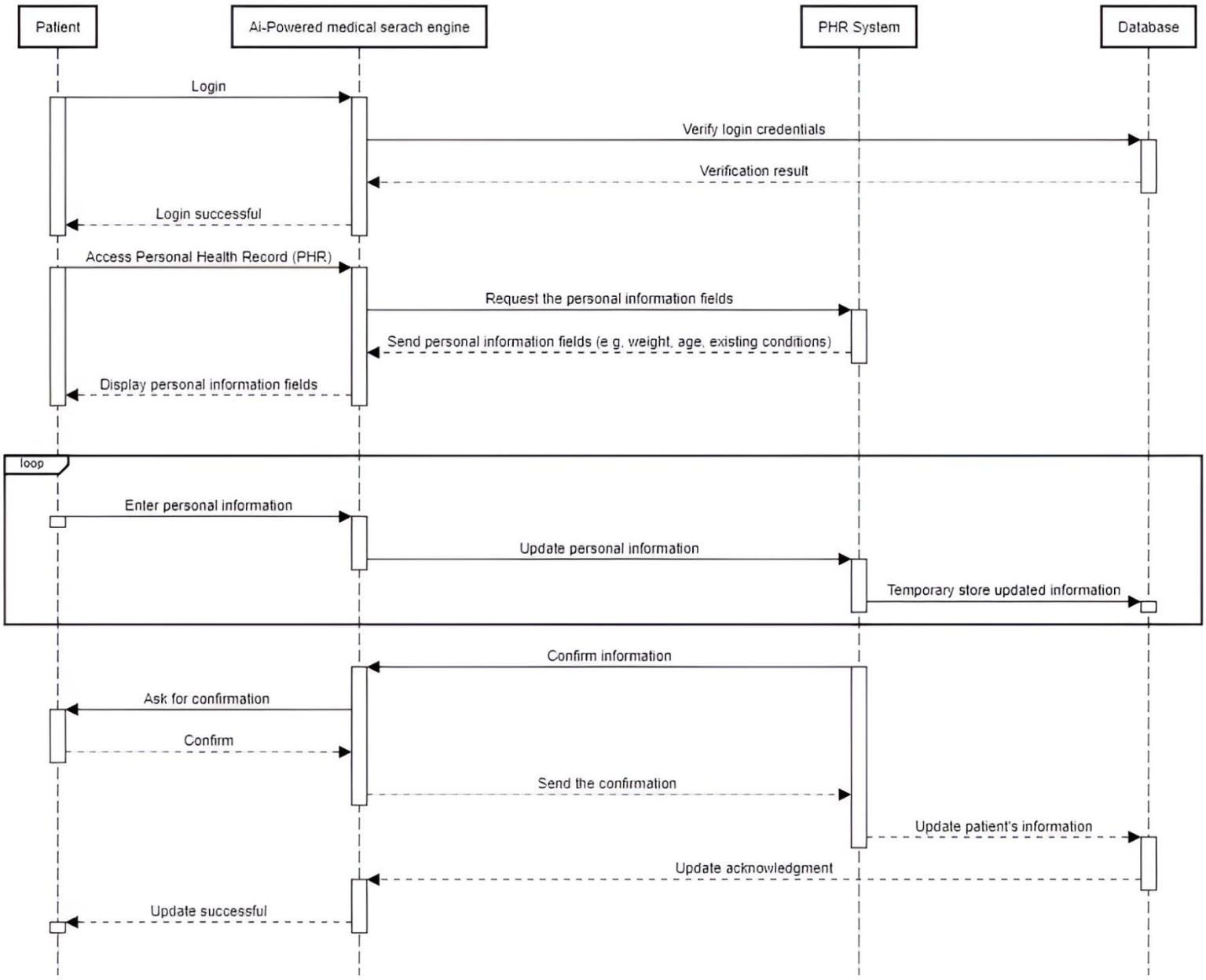

Figure 3.3 Sequence Diagram (PHR)

Usecase: Patients education (detailed information)
Actor: Patient

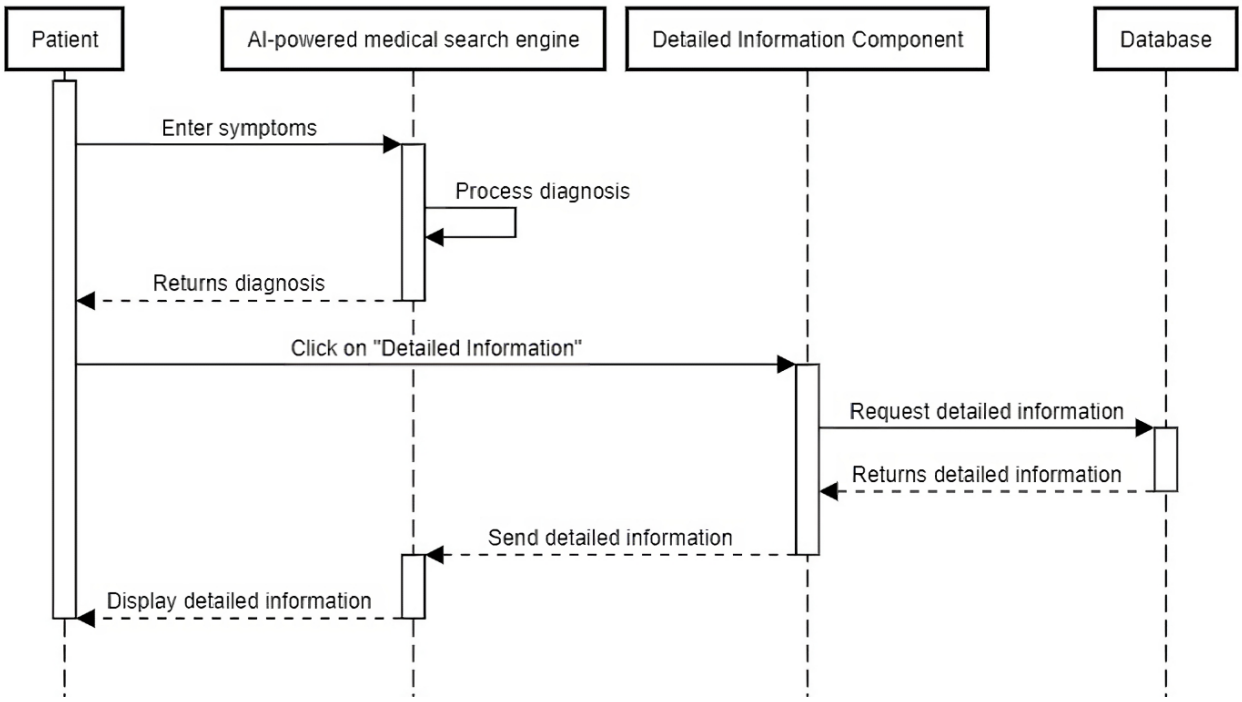

Figure 3.4 Sequence Diagram (Patient Education)

Usecase: Drug Interaction Checker
Actor: Doctor

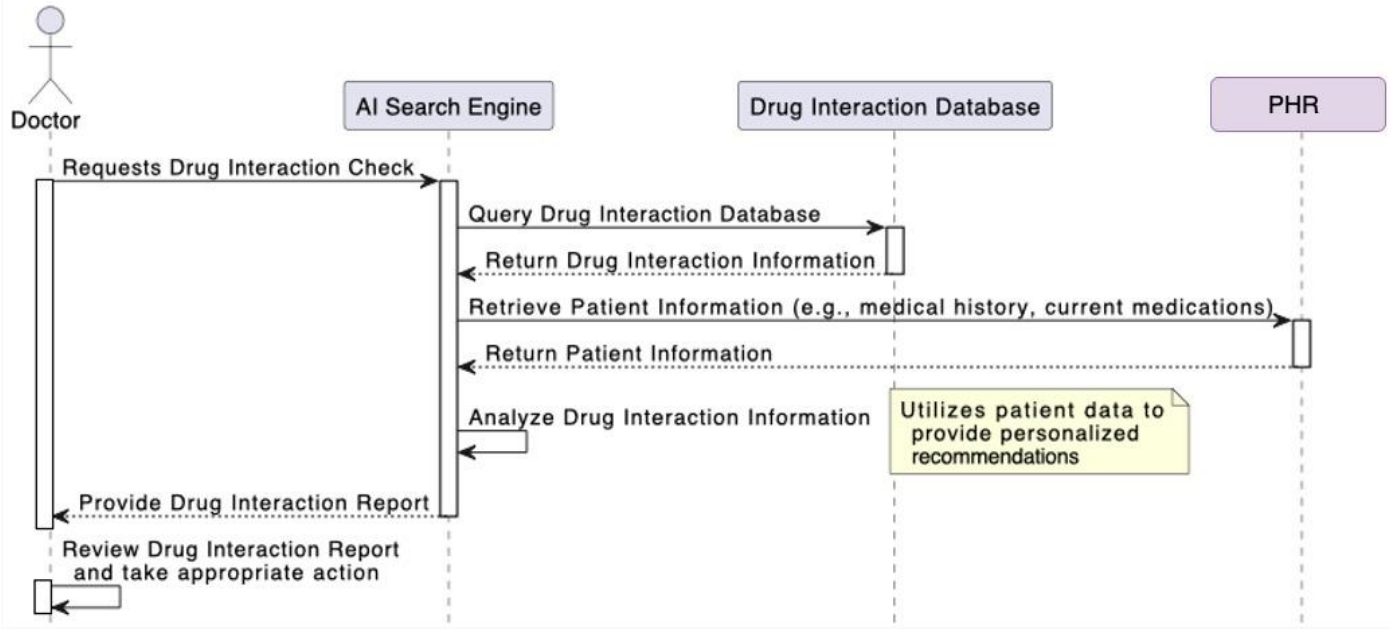

Figure 3.5 Sequence Diagram (Drug Interaction)

Usecase: Save Search Result
Actor: Doctor

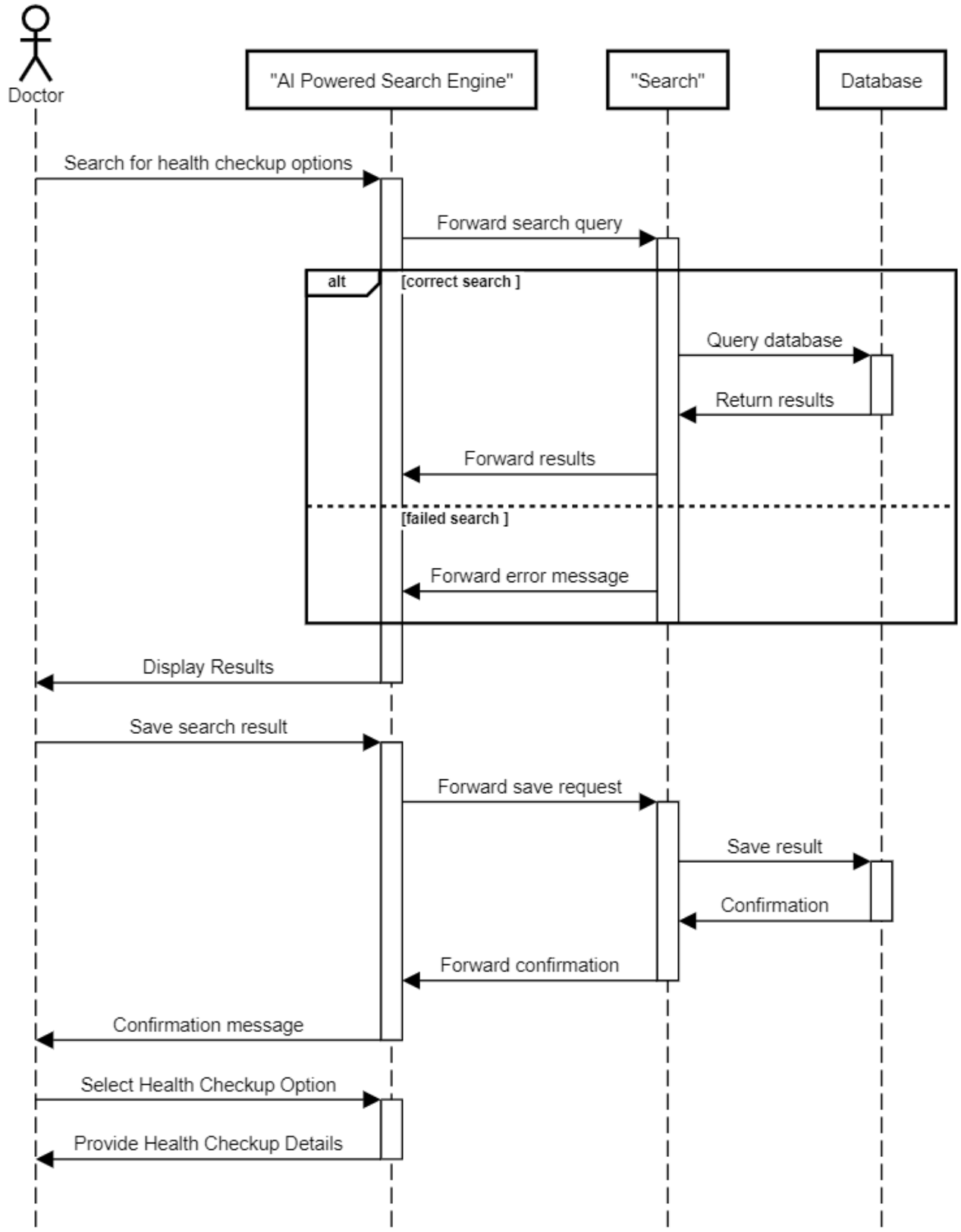

Figure 3.6 Sequence Diagram (Save Search)

Usecase: Diagnostic Checker
Actor: Doctor

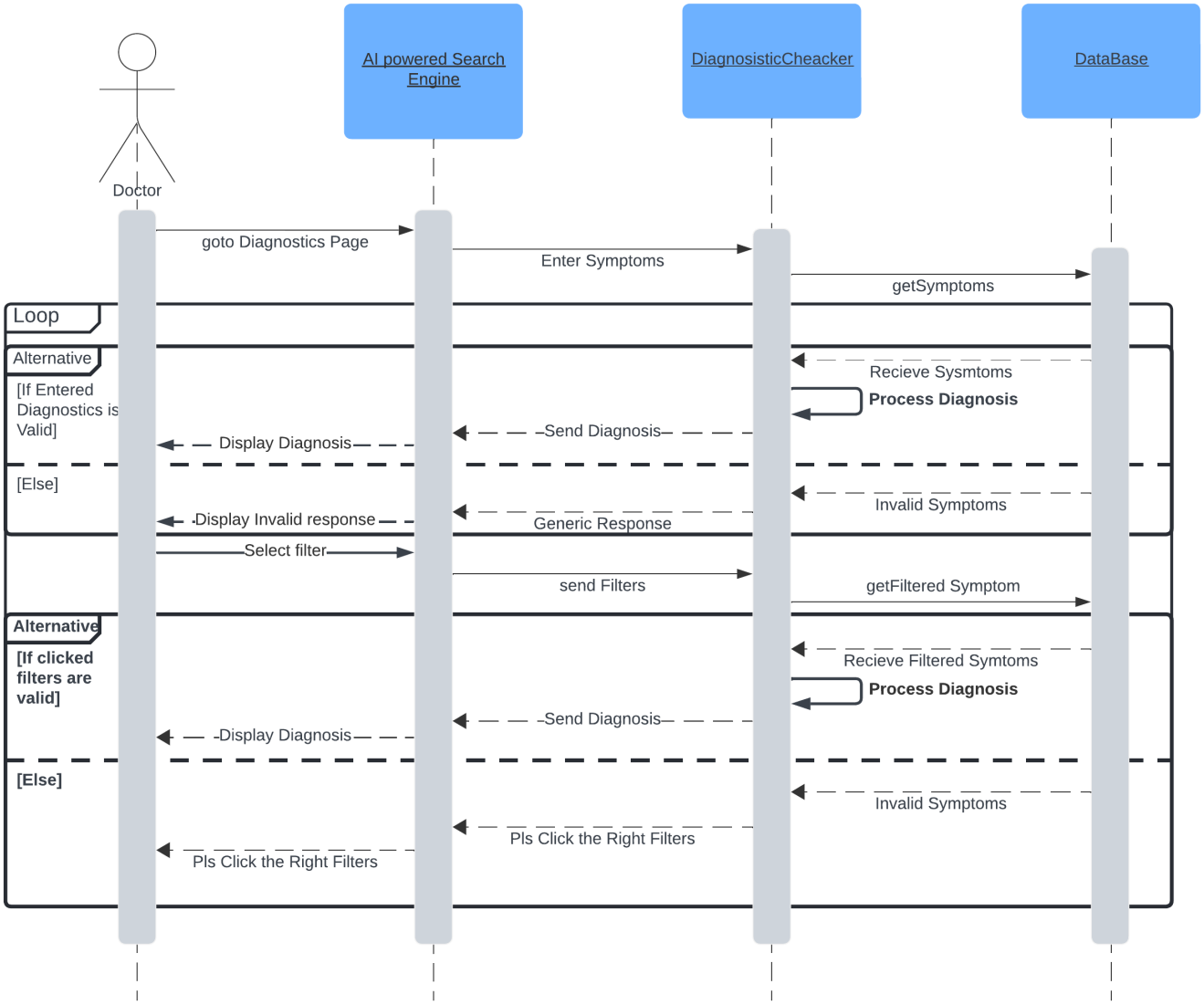

Figure 3.7 Sequence Diagram (Diagnosis Checker)

Usecase: Search
Actor: Patient

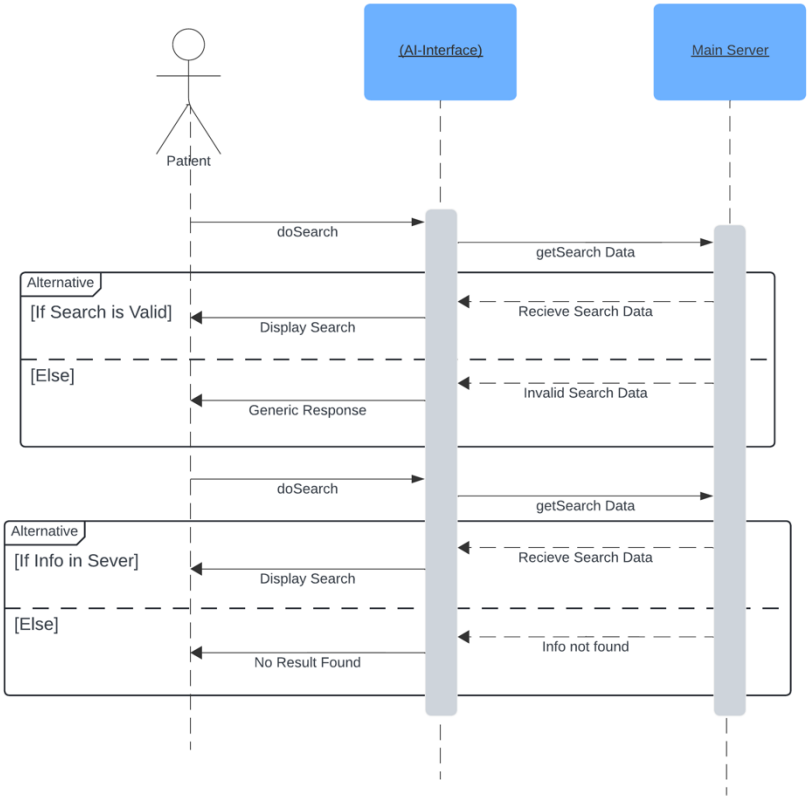

Figure 3.8 Sequence Diagram (Search)

3.3 System Architecture

<Provide the technical architecture of your system. Include a high-level architecture diagram that highlights major subsystems and components.>

Usecase: Patient education (Detailed information)

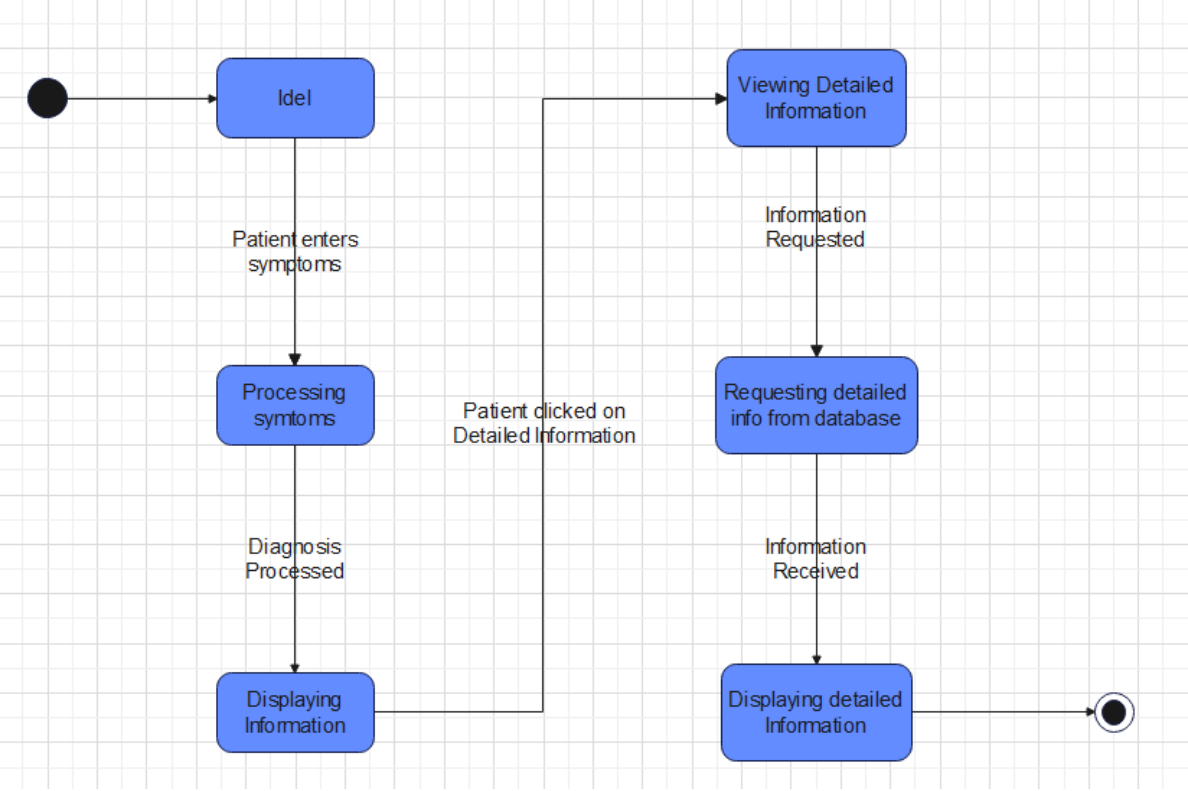

Figure 3.9 State Diagram (Patient Education / detailed info)

Usecase: Personal Health Record Access

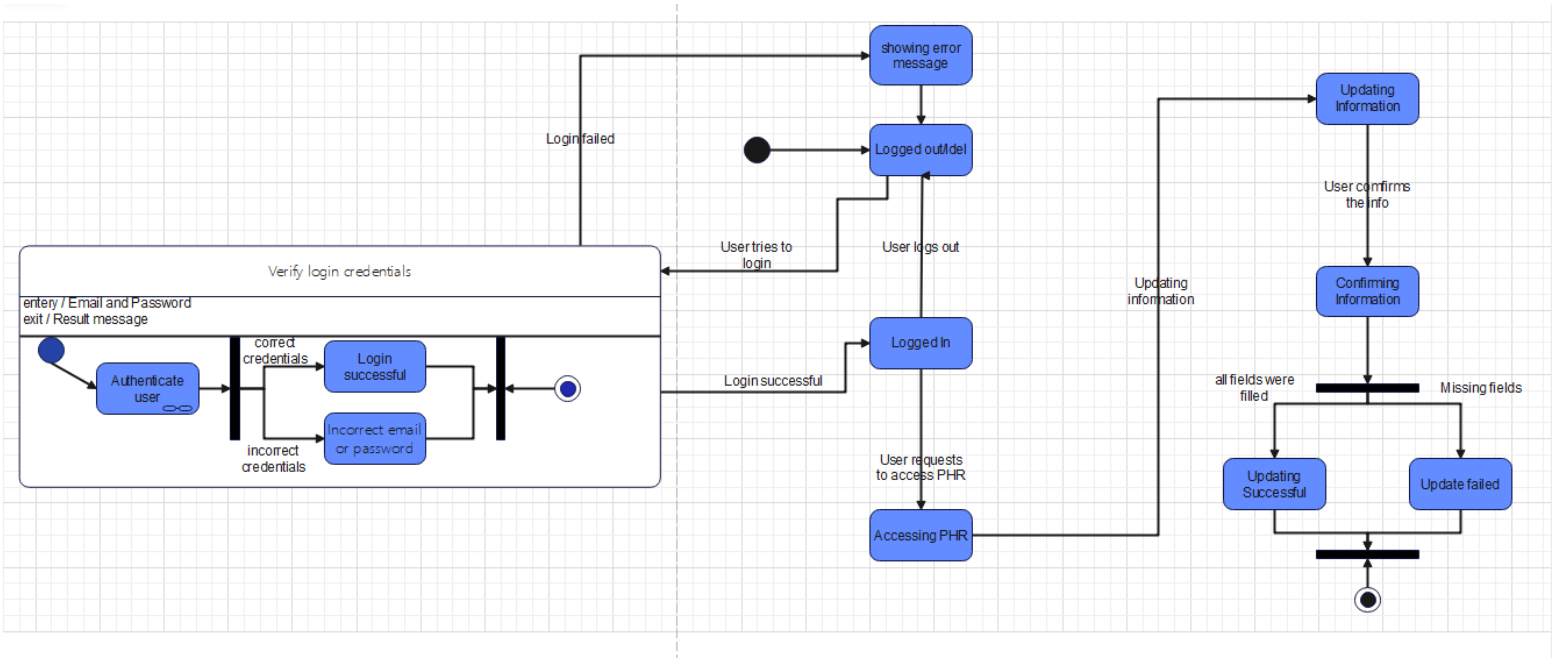

Figure 3.10 State Diagram (PHR system)

Usecase: Diagnosis Checker

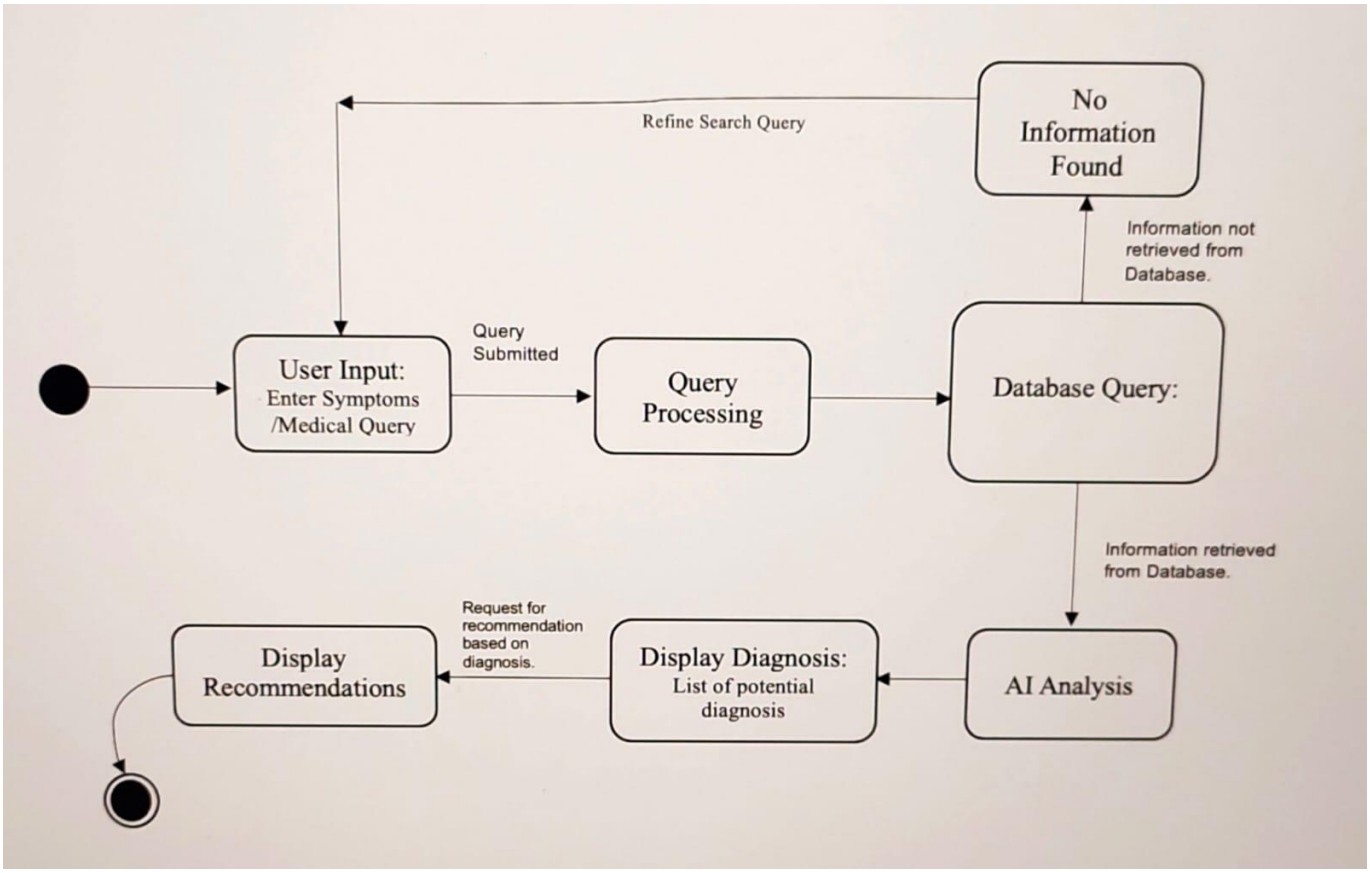

Figure 3.11 State Diagram (Diagnosis Checker)

Usecase: Search

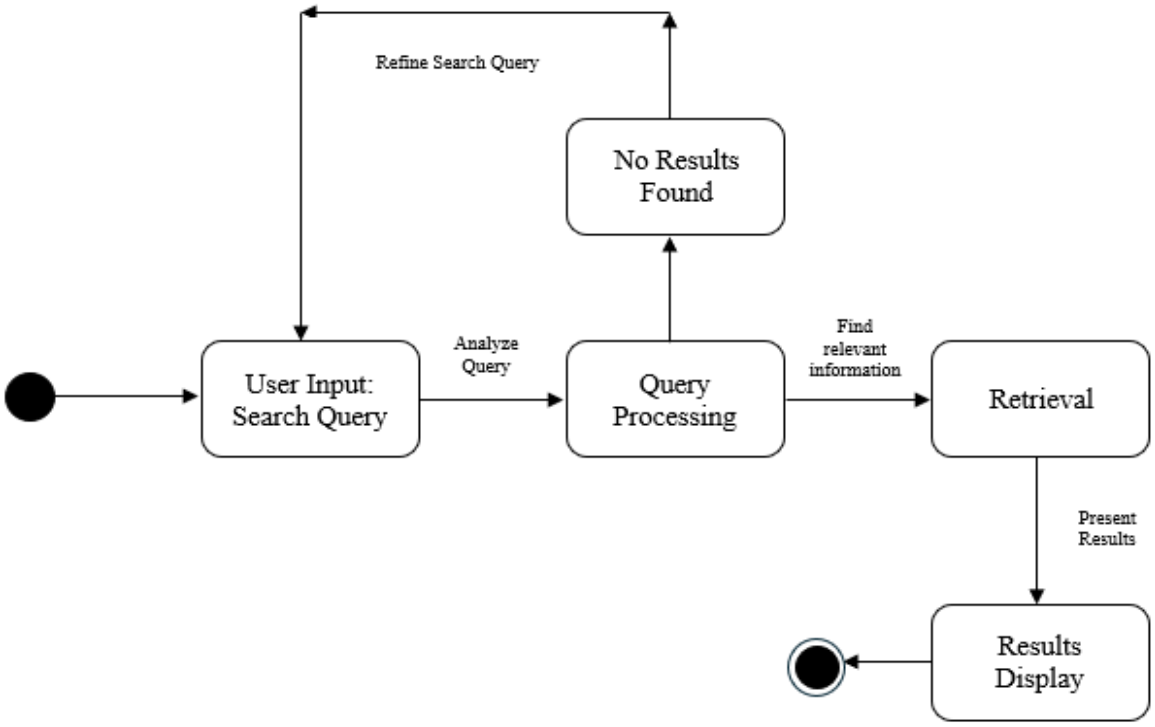

Figure 3.12 State Diagram (Search)

Usecase: Save search result

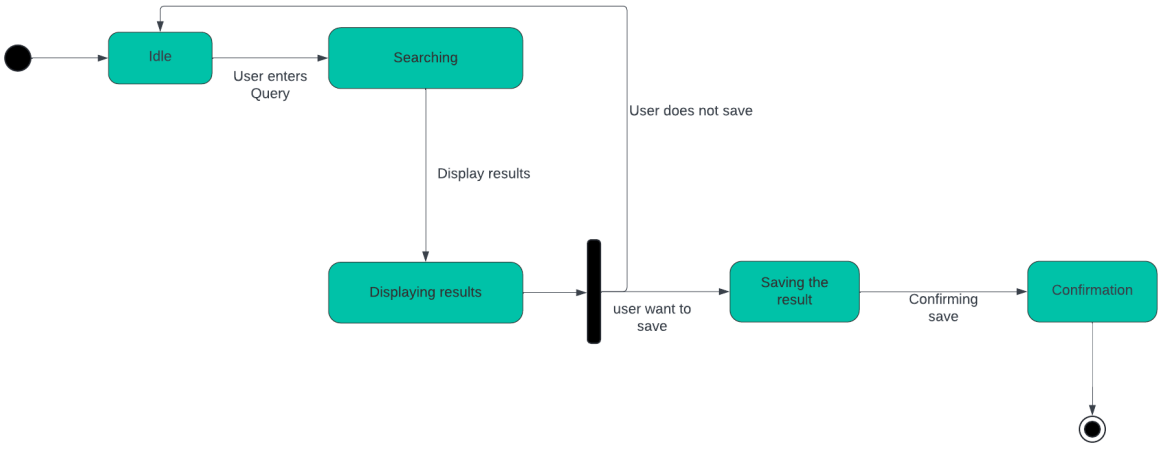

Figure 3.13 State Diagram (Save Search Result)

Usecase: Drug interactions checker

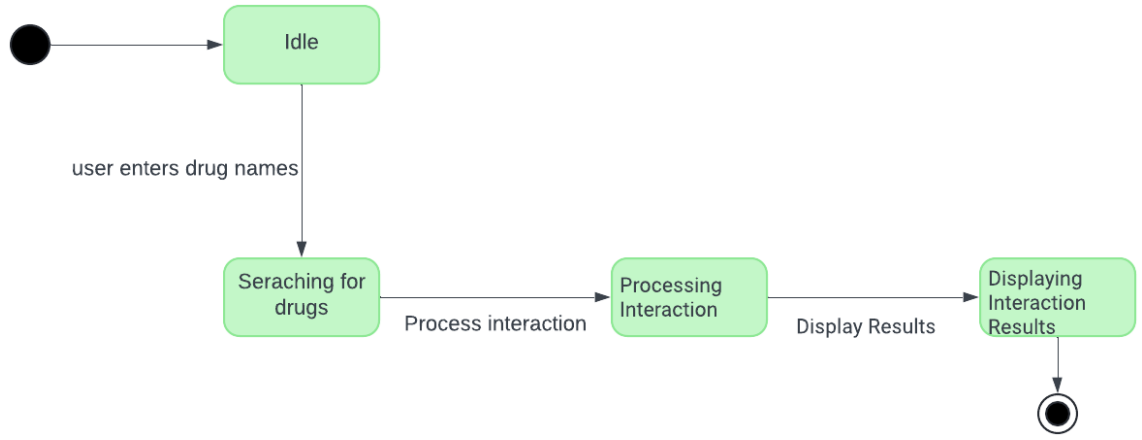

Figure 3.14 State Diagram (Drug interaction)

3.4 Architecture Evaluation

The architecture of the AI-powered medical search engine has been designed with careful consideration of the available infrastructure and technology options. This section evaluates the decisions made during the selection process, highlighting the reasons behind these choices, the advantages and disadvantages of the selected technologies, and the alternatives that were considered.

Selection of Infrastructure/Technology

1. Cloud-Based Infrastructure:

Reason for Selection: The cloud-based infrastructure was chosen due to its scalability, flexibility, and cost-effectiveness. Cloud platforms like AWS, Azure, and Google Cloud offer robust services that can handle the dynamic needs of an AI-powered application, providing computational resources on-demand and ensuring high availability and reliability.

Pros:

- **Scalability:** Ability to scale resources up or down based on demand, which is crucial for handling varying loads.
- **Cost-Effectiveness:** Pay-as-you-go pricing model reduces the upfront investment in hardware.
- **Reliability:** High availability and disaster recovery options ensure continuous service.
- **Flexibility:** Wide range of services and tools that support rapid development and deployment.

Cons:

- **Dependency on Internet Connectivity:** Requires stable internet connections, which may be a limitation in certain areas.
- **Security Concerns:** Storing sensitive health data on the cloud necessitates robust security measures to prevent breaches.
- **Compliance Challenges:** Ensuring compliance with healthcare regulations like HIPAA and GDPR requires careful management of data privacy and security.

Alternative Considered: On-Premises Infrastructure

- **Pros:** Greater control over hardware and data security, potentially lower long-term costs for large-scale operations.
- **Cons:** High initial investment, limited scalability, and increased maintenance burden.

2. Frameworks and Libraries:

Natural Language Processing (NLP):

Selected Technology: spaCy and NLTK

Reason for Selection: spaCy and NLTK were selected for their powerful NLP capabilities and ease of integration. These libraries support a wide range of NLP tasks essential for processing medical queries, such as tokenization, part-of-speech tagging, named entity recognition, and dependency parsing.

Pros:

- **Comprehensive Functionality:** Provides tools for a wide range of NLP tasks.
- **Performance:** Optimized for performance, handling large volumes of text efficiently.
- **Community and Support:** Strong community support and extensive documentation.

Cons:

- **Complexity:** Requires significant expertise to implement advanced NLP features.
- **Resource Intensive:** Can be computationally demanding, requiring robust hardware.

Alternative Considered: TensorFlow and PyTorch (for custom NLP models)

- **Pros:** Greater flexibility and customization for specific NLP tasks.
- **Cons:** More complex to implement and optimize, requiring deep expertise in machine learning.

Machine Learning Models:

Selected Technology: Scikit-Learn and TensorFlow

Reason for Selection*:* Scikit-Learn was chosen for its simplicity and efficiency in implementing traditional machine learning algorithms, while TensorFlow was selected for developing and deploying deep learning models, which are essential for handling complex medical queries.

Pros:

- **Versatility:** Both libraries cover a wide range of machine learning tasks.
- **Integration:** Easy to integrate with other Python-based tools and libraries.
- **Community and Support:** Extensive support and resources available for developers.

Cons:

- **Learning Curve:** Steep learning curve for TensorFlow, especially for developers new to deep learning.
- **Resource Demands:** Deep learning models require significant computational resources for training and inference.

Alternative Considered: PyTorch

- **Pros:** Dynamic computation graph and ease of use for research and development.
- **Cons:** Less mature deployment ecosystem compared to TensorFlow.

Conclusion

The selected architecture and technologies for the AI-powered medical search engine offer a balanced approach to scalability, performance, and ease of development. While alternatives were considered, the chosen infrastructure and tools provide the necessary capabilities to build a robust, reliable, and user-friendly system. Ongoing evaluation and potential integration of alternative technologies will ensure that the system remains at the cutting edge of AI and medical informatics.

3.5 Component-External Entities Interface

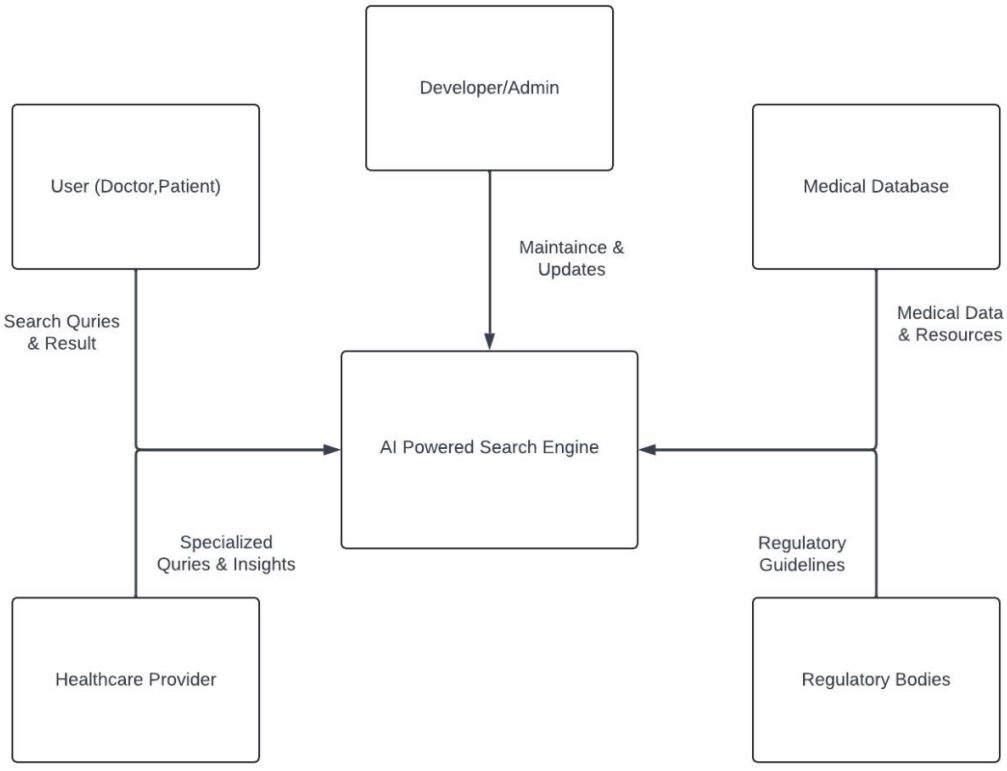

Figure 3.15 Context Diagram

3.6 Screenshots/Prototype

3.6.1 Workflow

<Describe complete workflow of your system. Swim-lane diagram may be used. This section should be an improved version of the section presented in previous phases>

The workflow of the AI-powered medical search engine is designed to provide a seamless and user-friendly experience for individuals seeking medical information. The process begins with the user logging into the system using their credentials. Once authenticated, the user can enter their medical query or symptoms into the search bar. The system then validates the query and triggers the AI-powered search algorithm, which processes the input using natural language understanding and generation. The AI algorithm analyzes the query, matches it with relevant medical data, and generates a text-based response providing the user with the requested information. The user can then review the information and, if needed, seek further details or clarification. The workflow ensures that users receive accurate and reliable medical guidance efficiently.

3.6.2 Screens

< Include all screenshots of your complete software application's graphical user interface.>

Login Page

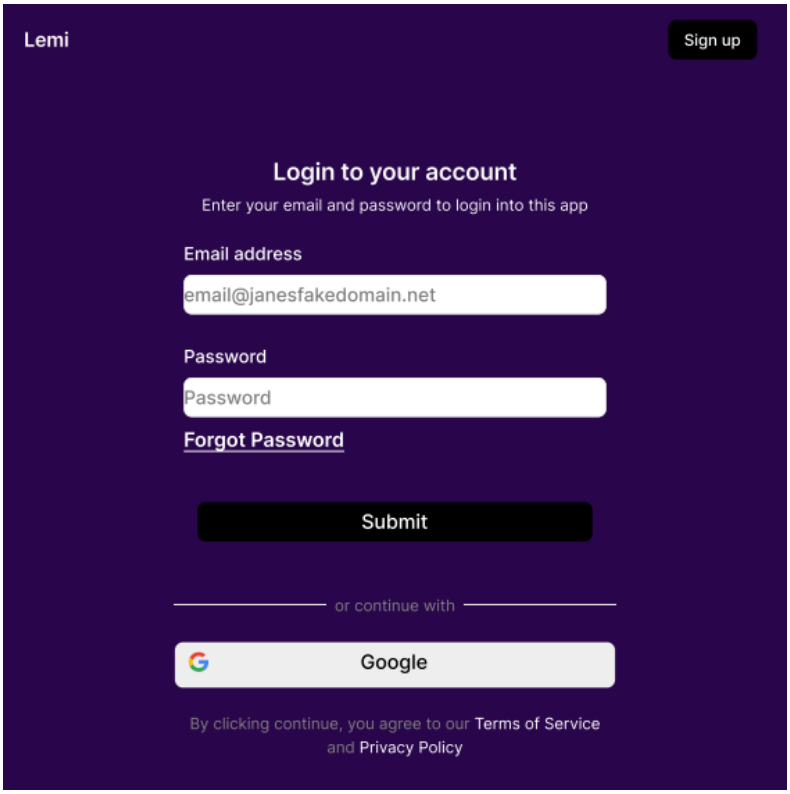

Figure 3.16 System UI (Login Page)

Sign up/ Create Account page.

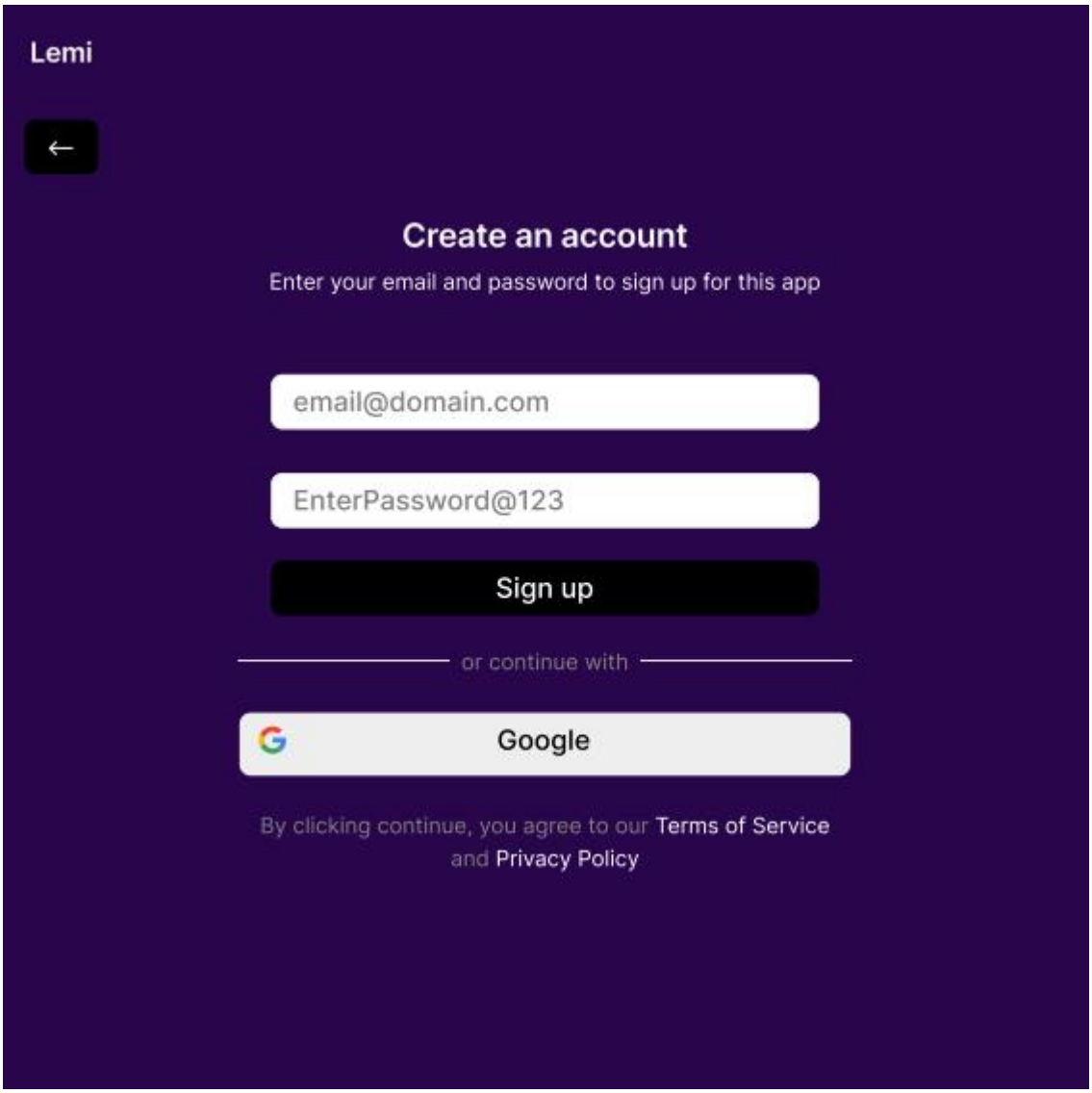

Figure 3.17 System UI (Sign up/ Create Account page)

Patient and doctor checker

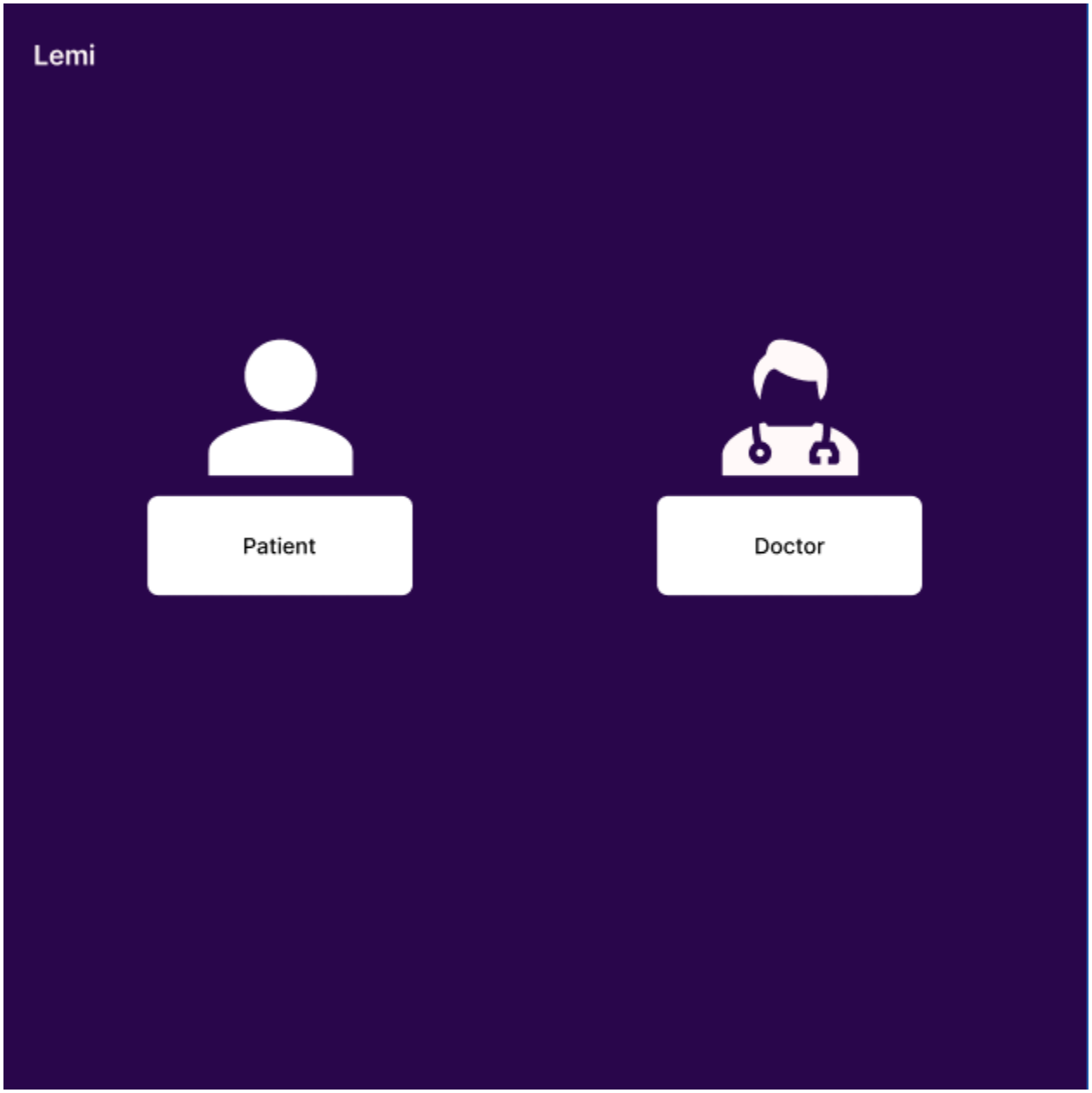

Figure 3.18 System UI (Patient and doctor checker)

Search (idle)

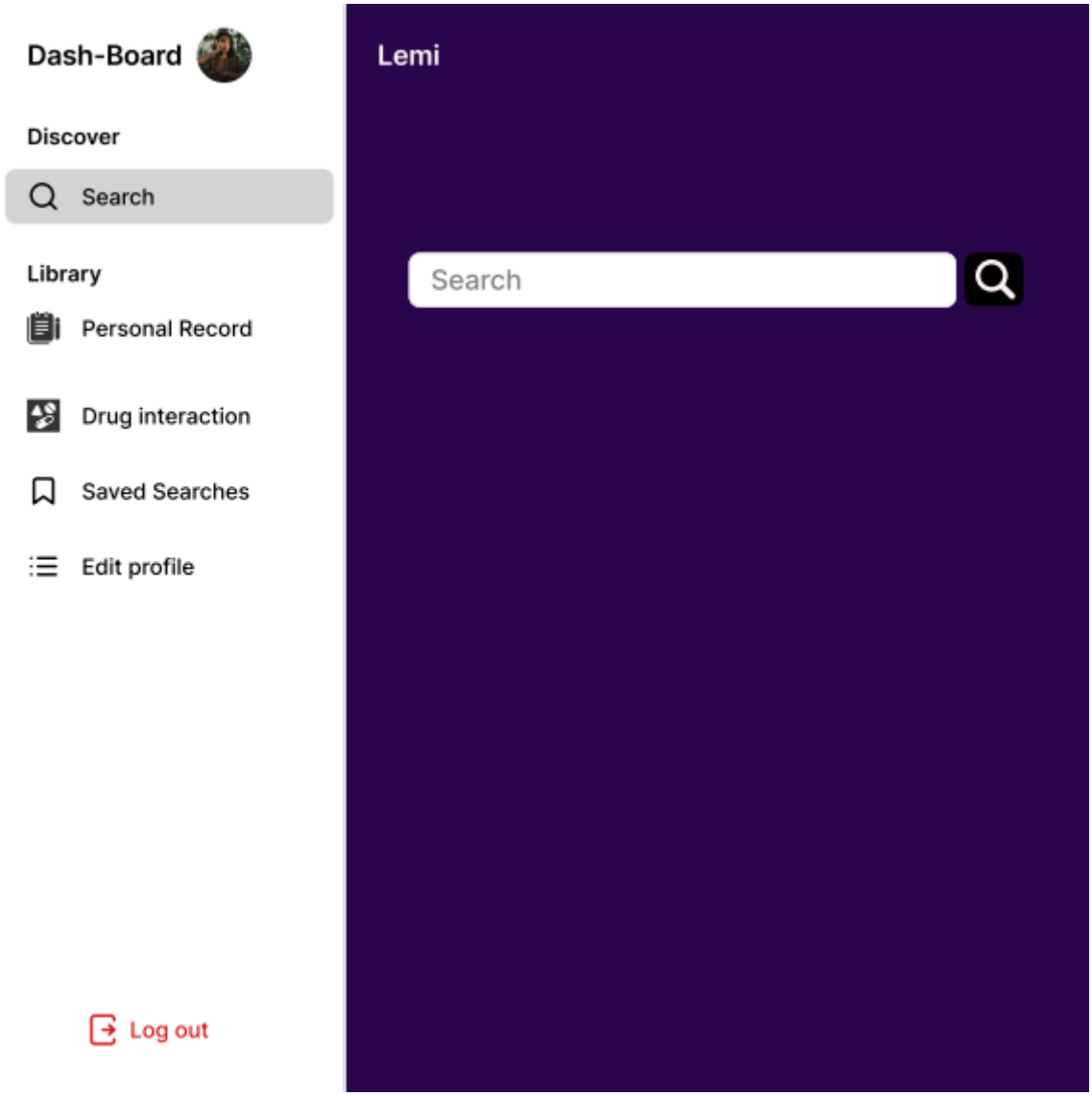

Figure 3.19 System UI [Search (idle)]

Search (failed)

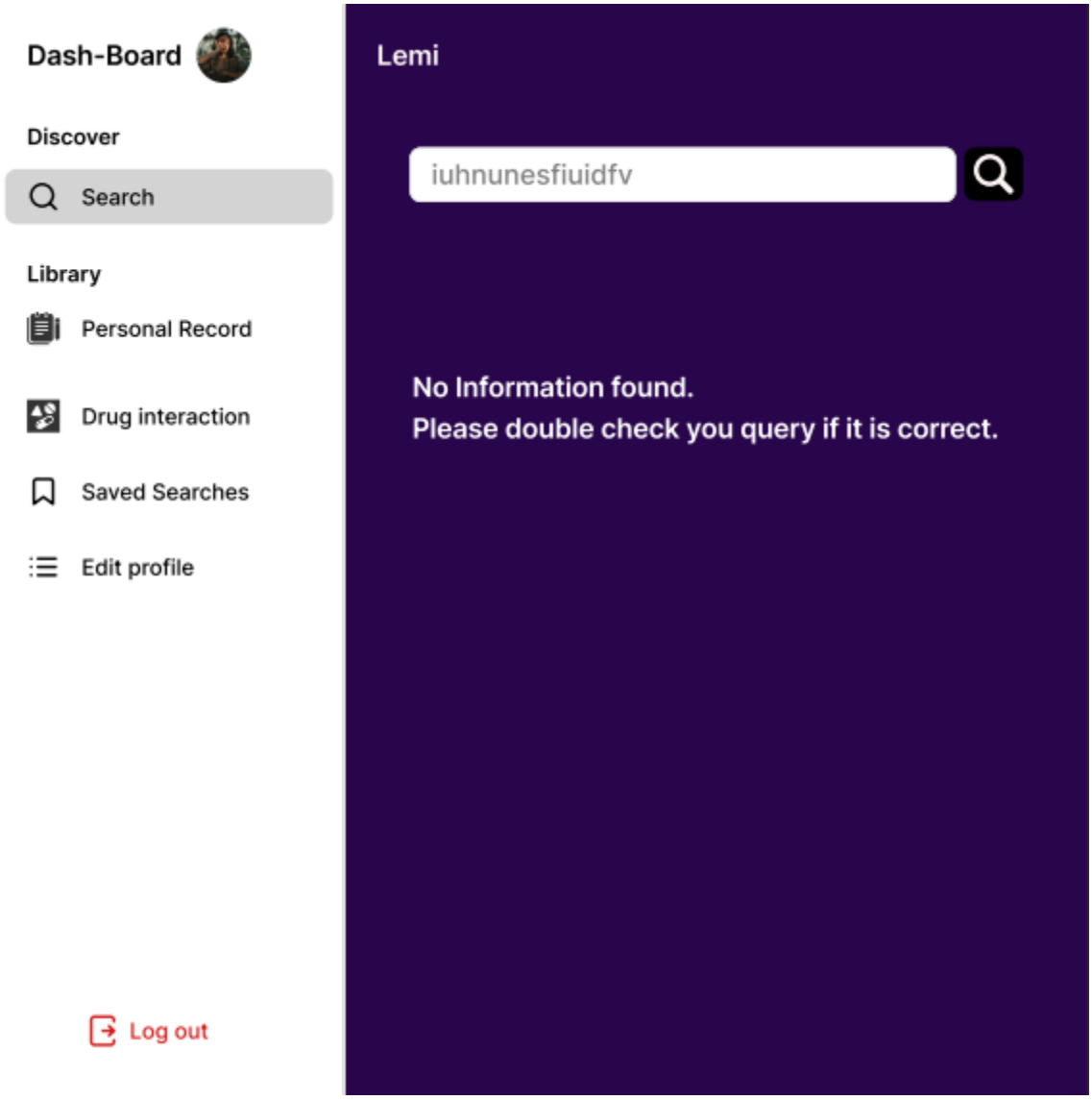

Figure 3.20 System UI [Search (failed)]

Search (Success)

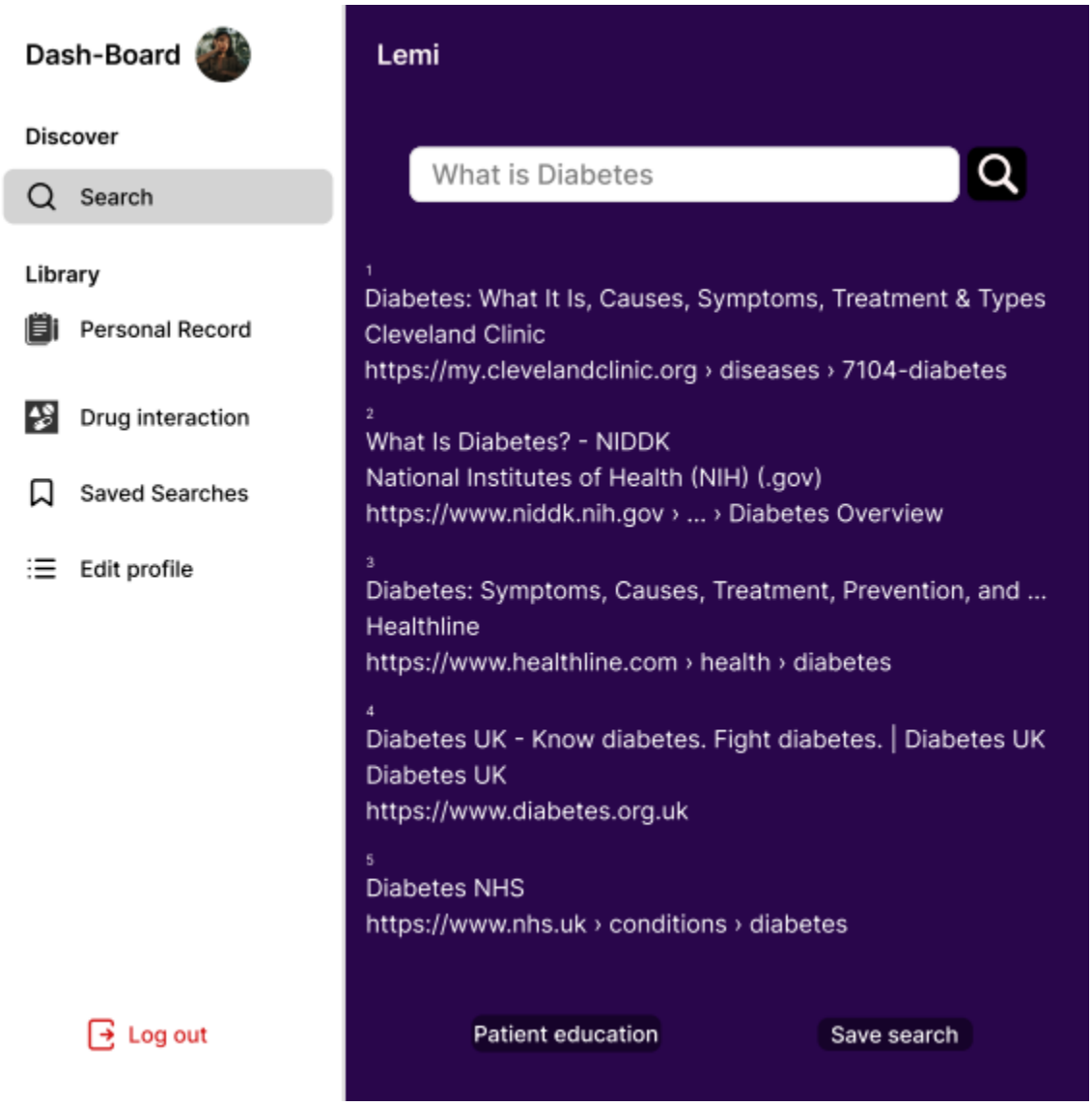

Figure 3.21 System UI [Search (Success)]

Patient education (detailed information)

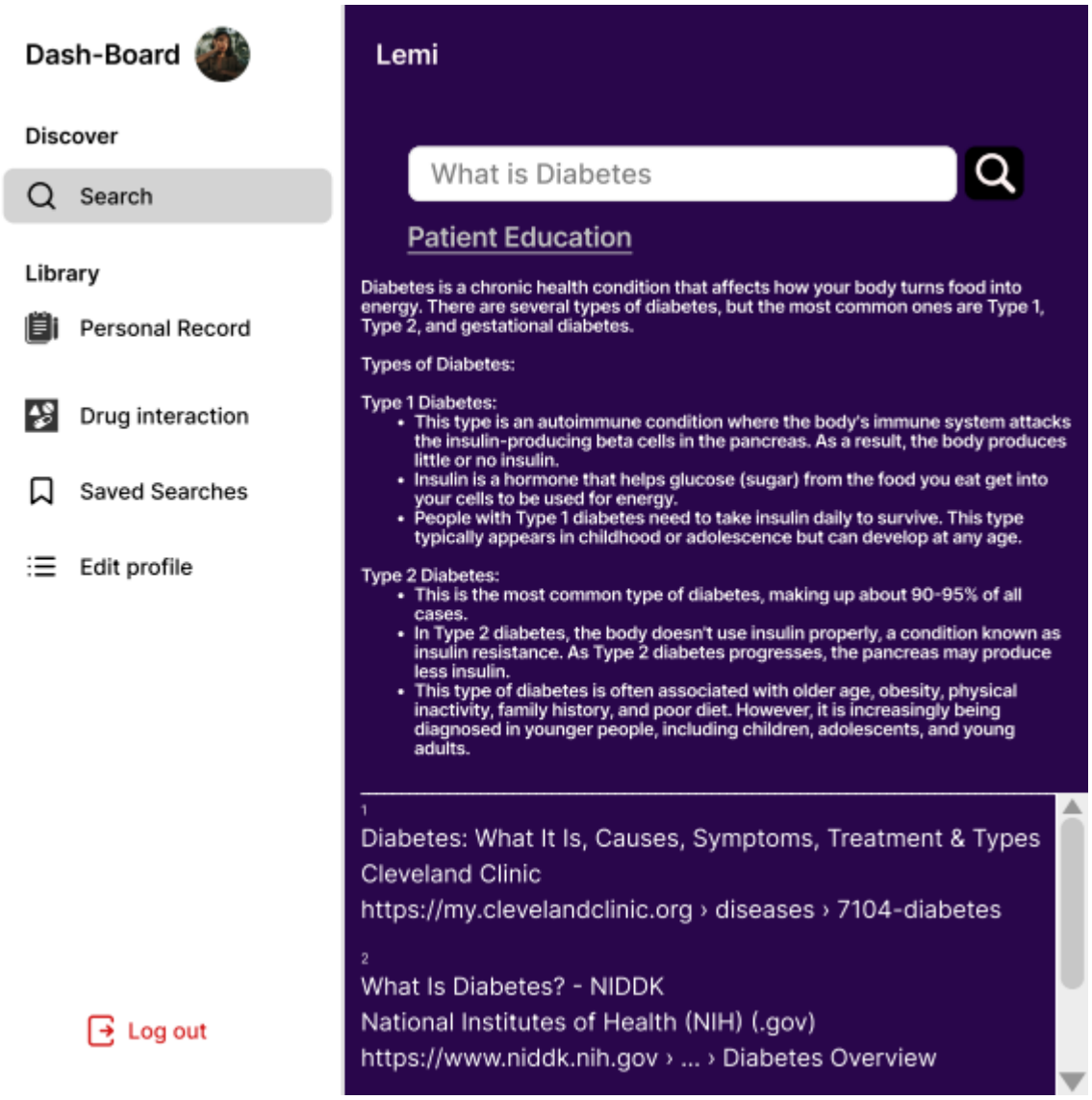

Figure 3.22 System UI [Patient education (detailed information)]

Saved Searches

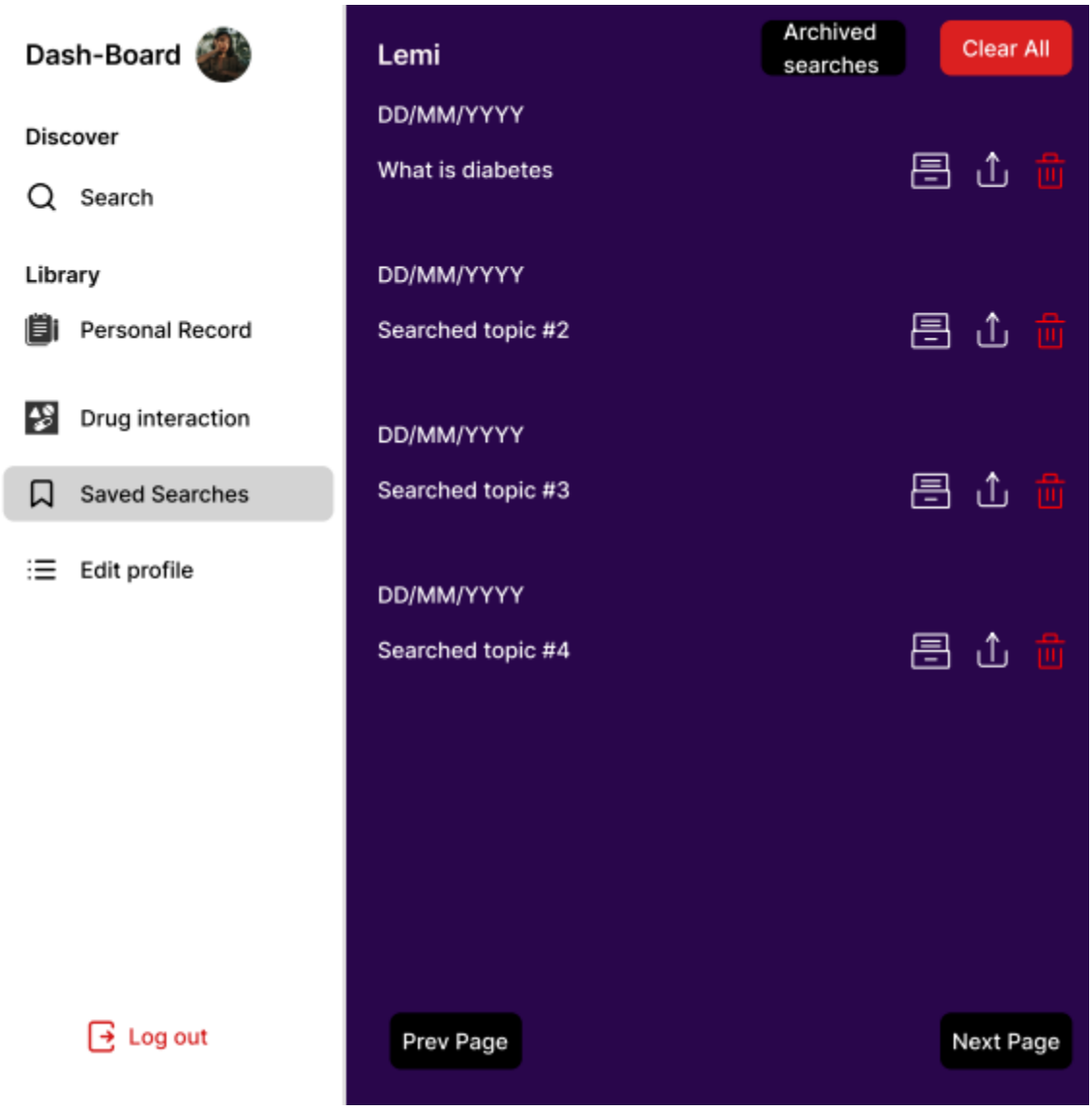

Figure 3.23 System UI (Saved Searches)

Saved Searches (Cleared)

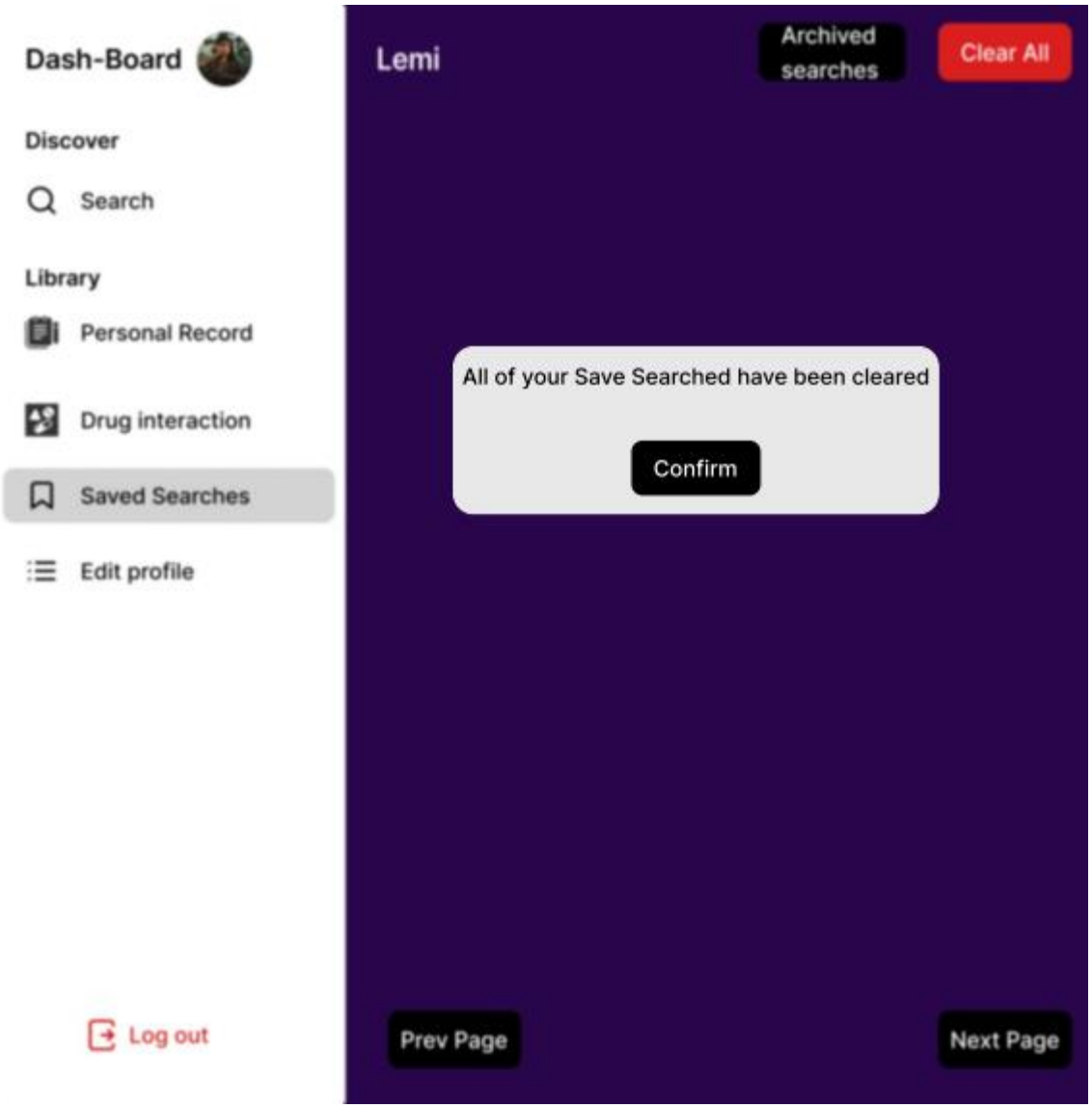

Figure 3.24 System UI [Saved Searches (Cleared)]

Drug interactions (idle)

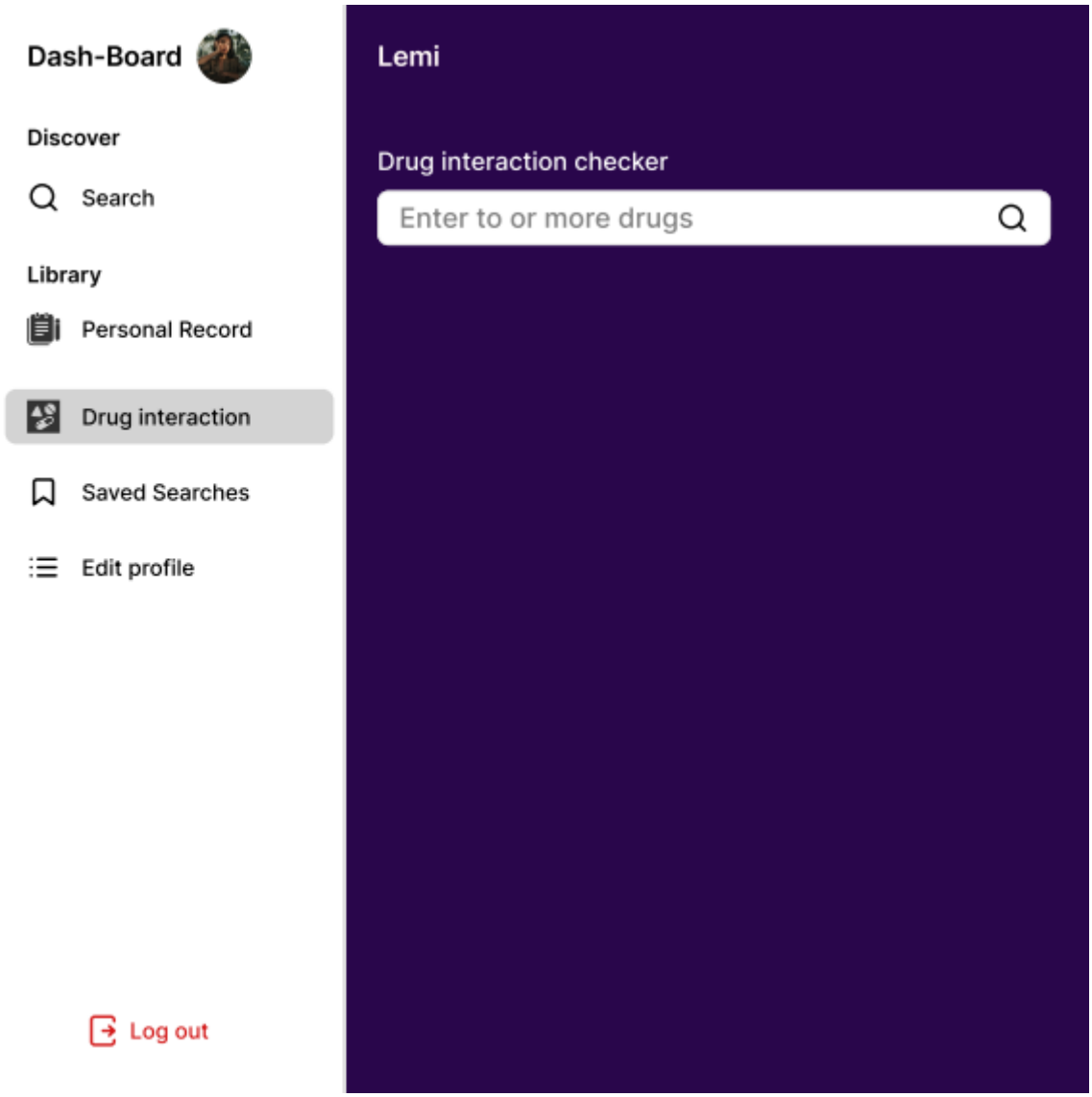

Figure 3.25 System UI [Drug interactions (idle)]

Drug interactions (Result)

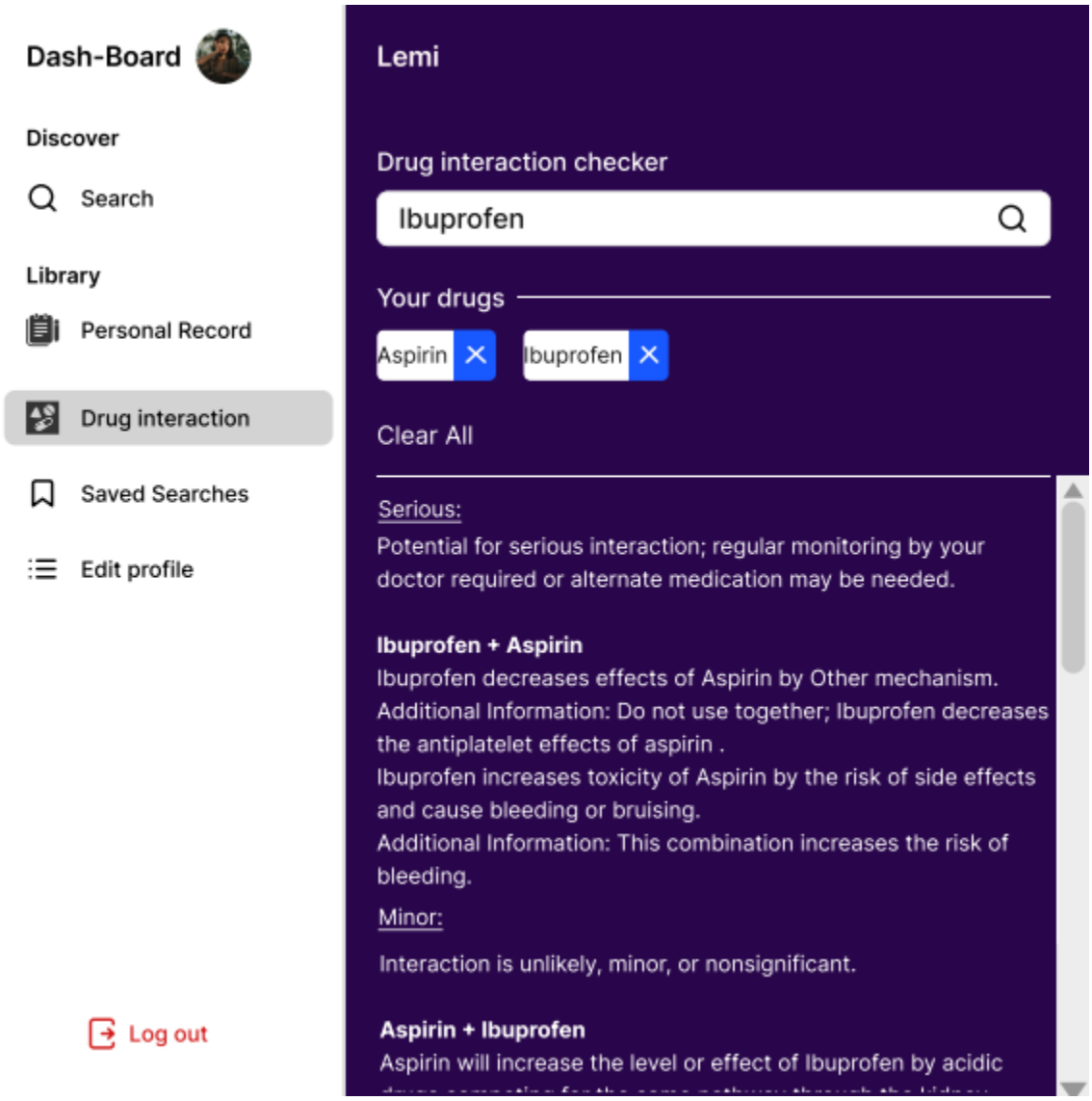

Figure 3.26 System UI [Drug interactions (Result)]

Personal Health Record (PHR)

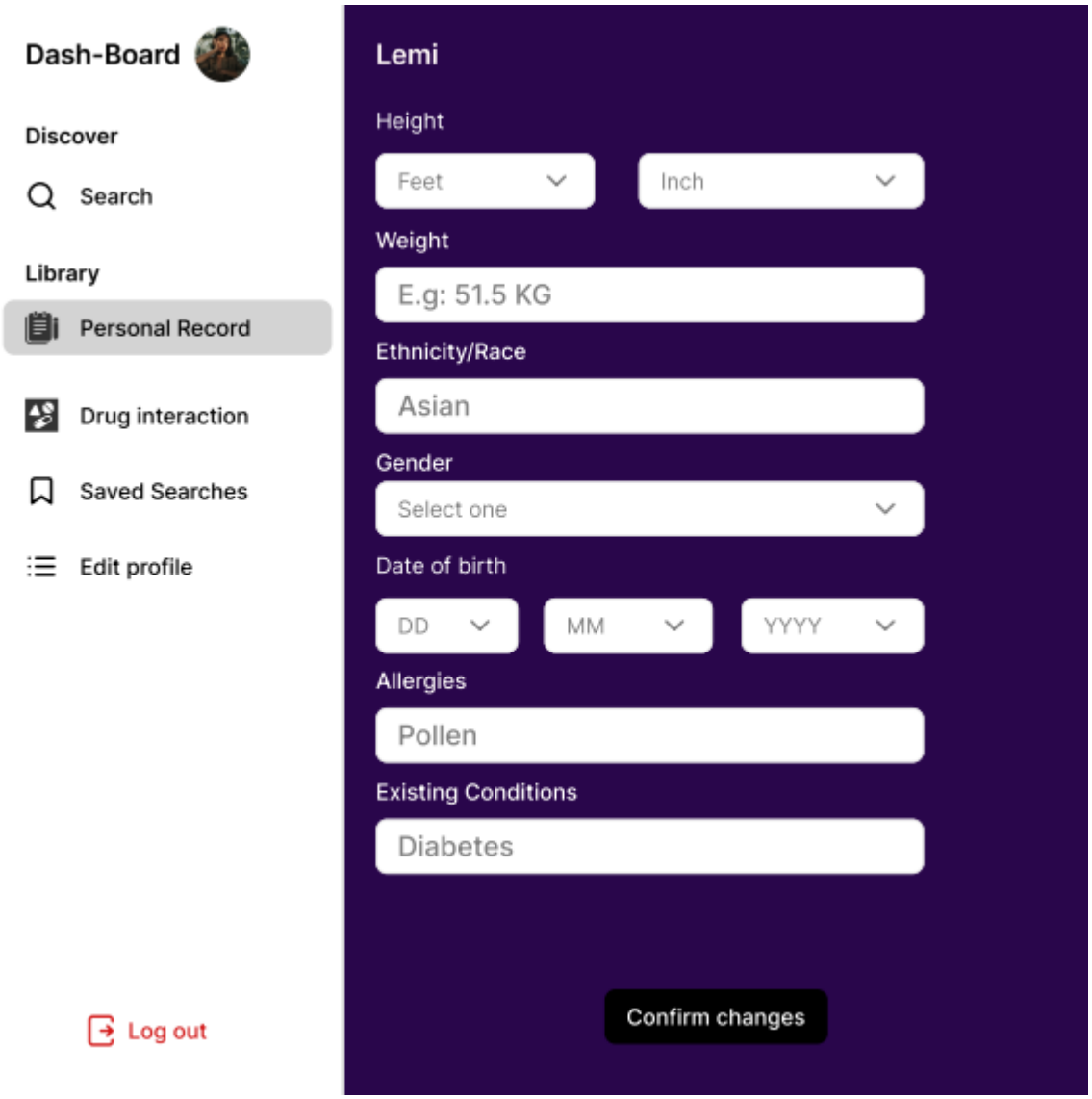

Figure 3.27 System UI [Personal Health Record (PHR)]

Edit profile.

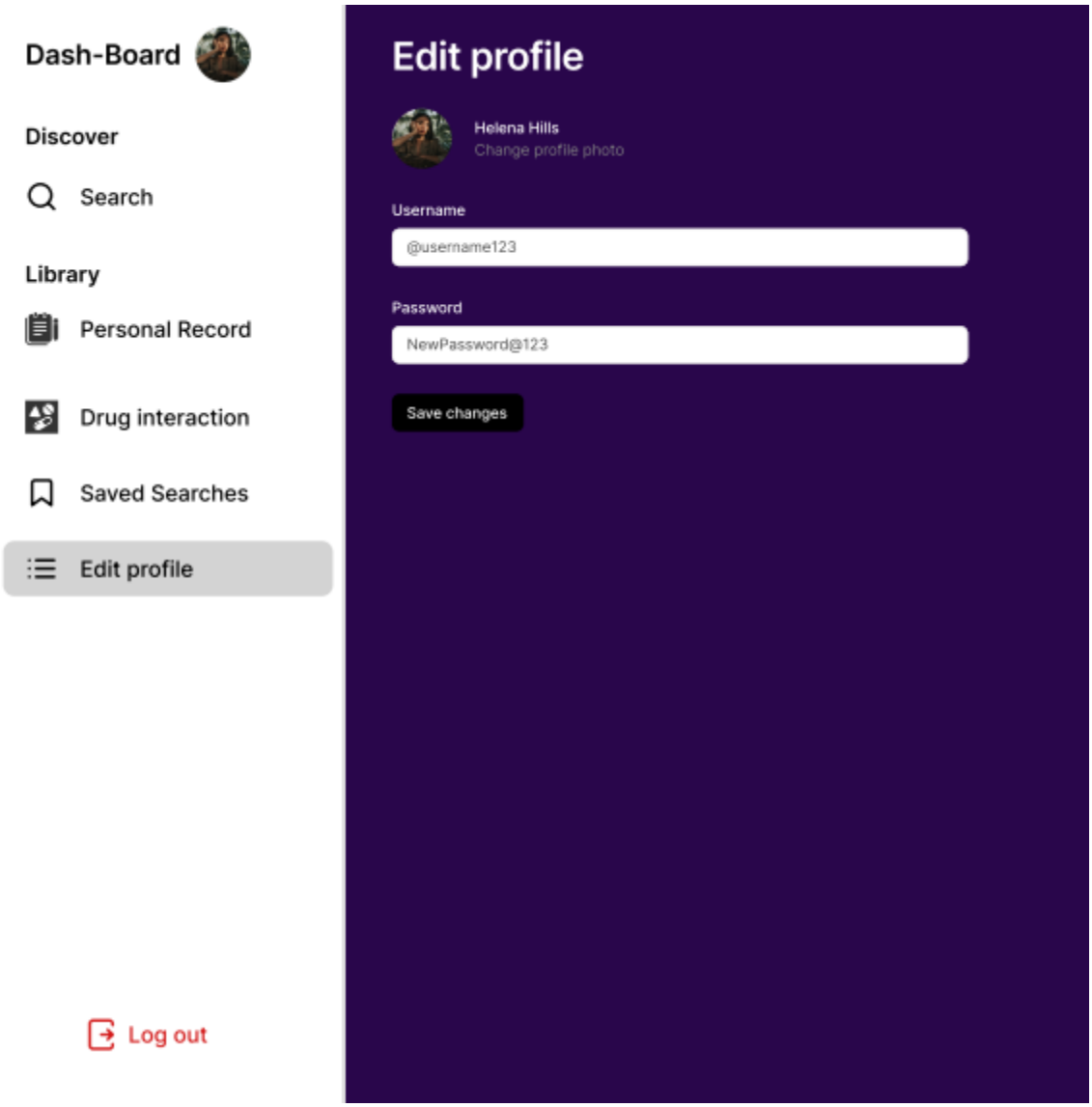

Figure 3.28 System UI (Edit profile)

Logout

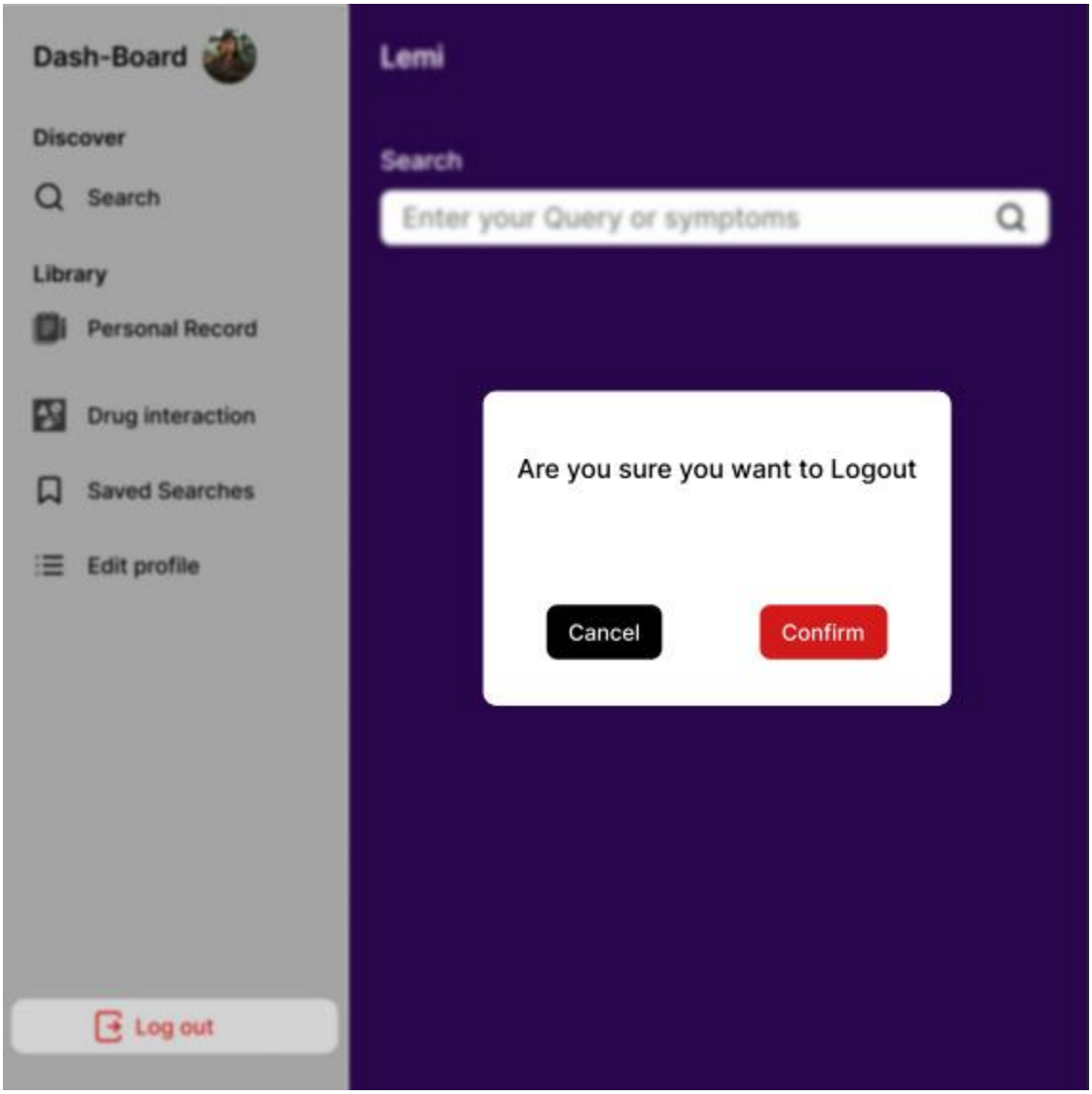

Figure 3.29 System UI (Logout)

3.7 Other Design Details

This section covers additional design details not previously addressed. The AI-powered medical search engine integrates several key features to enhance its functionality and user experience. These include:

1. **Natural Language Processing (NLP)**: The system uses advanced NLP techniques to understand and interpret user queries, ensuring accurate responses.
2. **Machine Learning Models**: Continuous training and updating of machine learning models are implemented to improve the system's accuracy and relevance over time.
3. **User Interface (UI)**: The UI is designed to be intuitive and accessible, catering to users with varying levels of digital literacy.
4. **Data Security**: Robust security measures, including data encryption and compliance with HIPAA and GDPR, are in place to protect user information.
5. **Interoperability**: The system supports integration with Electronic Health Records (EHR) to facilitate seamless data exchange and enhance the user experience.

Chapter 4: Test Specification and Results

4.1 Test Case Specification

Identifier	Patient Education (test case)
Related requirements(s)	Use Case: Patient Education
Brief description	Test the functionality of the "Detailed Information" component after the AI diagnosis.
Pre-condition(s)	The user has received a diagnosis from the AI-powered medical search engine.
Input data	None
Detailed steps	1. After receiving the diagnosis, click on the "Detailed Information" button. 2. Wait for the detailed information to be displayed.
Expected result(s)	Detailed information about the diagnosis is displayed, including frequency of occurrence among different demographics, most common age groups affected by the condition, common treatments for the condition, how long does it take to recover from it on average, and preventive measures.
Post-condition(s)	The user has access to detailed information to better understand their diagnosis and potential treatment options.
Actual result(s)	Detailed information about the diagnosis is displayed as expected.
Test Case Result	Pass

Table 4.1: Patient Education (test case)

Identifier	PHR
Related requirements(s)	Use Case: Updating Personal Health Record
Brief description	Test the functionality of updating personal health record information.
Pre-condition(s)	The user is logged in and accessing their personal health record.
Input data	New or updated personal health information (e.g., weight, height, existing conditions, allergies, race, etc.)
Detailed steps	1. Navigate to the personal health record section. 2. Enter new or updated personal health information. 3. Save the changes.
Expected result(s)	The personal health record is updated with the new or modified information.
Post-condition(s)	The user's personal health record reflects the changes made during the update process.
Actual result(s)	The personal health record is successfully updated with the new or modified information.
Test Case Result	Pass

Table 4.2: PHR

Identifier	Search (test case)
Related requirements(s)	Functional requirement: Search functionality
Short description	Test the search functionality of the AI-powered health checkup search engine.
Pre-condition(s)	The search engine is accessible and operational.
Input data	Symptom query: “headache”
Detailed steps	1. Open the AI-powered health checkup search engine application. 2. Navigate to the search bar. 3. Enter the symptom query "headache". 4. Submit the query by clicking the search button or pressing enter.
Expected result(s)	Relevant search results related to "headache" are displayed. Results include information on potential causes, remedies, and nearby healthcare providers.
Post-condition(s)	Search results are displayed to the user.
Actual result(s)	Relevant search results related to “headache” are displayed, including causes, remedies, and nearby healthcare providers.
Test Case Result	Pass

Table 4.3: Search (test case)

Identifier	Symptom Checker (test case)
Related requirements(s)	Functional Requirement: Symptom Checker
Short description	Evaluate the symptom checker functionality of the AI-powered health checkup search engine.
Pre-condition(s)	The search engine is accessible and operational.
Input data	Symptom information (e.g., type, severity)
Detailed steps	1. Open the AI-powered health checkup search engine application. 2. Access the symptom checker feature. 3. Input details regarding the symptoms experienced (e.g., type, severity). 4. Submit the symptom information.
Expected result(s)	The symptom checker accurately assesses the inputted symptoms. Relevant suggestions or recommendations for further action (e.g., seeking medical attention, self-care measures) are provided based on the symptoms.
Post-condition(s)	User receives accurate assessment and recommendations based on inputted symptoms.
Actual result(s)	The symptom checker provides accurate assessment and relevant recommendations based on the inputted symptoms.
Test Case Result	Pass

Table 4.4: Symptom Checker (test case)

Identifier	Drug interaction (test case)
Related requirements(s)	Functional requirement: Drug interaction
Short description	User can enter to drug names and the system checks if those 2 can be taken together
Pre-condition(s)	The drugs exist and are in the database. The search engine is accessible and operational.
Input data	User enters the name of the drugs into the system. [Optional] User can enter their allergies.
Detailed steps	1. User accesses the drug interaction checker feature. 2. User enters the names of two drugs into the system. 3. [Optional] User specifies any allergies they have. 4. User submits the query.
Expected result(s)	The system should display information indicating whether the two drugs can be taken together safely. If there are any potential interactions or contraindications, the system should provide relevant warnings or recommendations.
Post-condition(s)	The user receives guidance on the compatibility of the entered drugs and any associated precautions.
Actual result(s)	User receives correct guidance from the system on whether the drugs are compatible or not. If the drugs are not compatible with each other, it gives proper warnings and recommendations on substitutes.
Test Case Result	Pass

Table 4.5: Drug interaction (test case)

Identifier	Save Search (test case)
Related requirements(s)	Functional requirement: Save Search Results
Short description	Test the functionality of saving search results for patients and doctors.
Pre-condition(s)	The Patient/Doctor is logged into the AI-powered Search Engine.
Input data	Search results for a specific medical condition.
Detailed steps	1. Conduct a search for a specific medical condition. 2. Save the search results for future reference.
Expected result(s)	The search results are successfully saved to the patient's or doctor's account for future access.
Post-condition(s)	The patient/doctor can access the saved search results from their account dashboard.
Actual result(s)	The search results are successfully saved to the patient's/doctor's account and accessible from the account dashboard.
Test Case Result	Pass

Table 4.6: Save Search (test case)

4.2 Summary of Test Results

<Provide in tabular form the defects found in each of your software modules. For example see Table 6.2 below.>

Module Name	Test Cases Run	Number of Defects Found	Number of Defects Corrected So Far	Number of Defects Still Needing Correction
Search Module	Search (test case)	5	4	1
Symptom Checker	Symptom Checker (test case)	3	3	0
Drug Interaction	Drug interaction (test case)	4	3	1
Save Search	Save Search (test case)	1	1	0
Patient Education	Patient Education (test case)	2	2	0
Personal Health Record	PHR	2	2	0
Complete System	TC7	17	15	2

Table 6.2: Summary of All Test Results

Chapter 5: Conclusion and Future Work

5.1 Project summary

The AI-powered medical search engine project aimed to provide users with accurate and reliable medical information using advanced AI technologies. Throughout the project, we developed a system capable of interpreting user queries, processing them through AI algorithms, and generating relevant medical guidance. The project addressed key challenges such as data privacy, system integration, and user accessibility, resulting in a robust and user-friendly application.

5.2 Problems faced and lessons learned.

During the project, we encountered several challenges, including the integration of diverse data sources, ensuring data privacy compliance, and optimizing the AI algorithms for accuracy and speed. These challenges taught us the importance of meticulous planning, continuous testing, and iterative development. We also learned the value of user feedback in refining the system and enhancing its usability.

5.3 Future work

Future work on the AI-powered medical search engine will focus on expanding the dataset to include more diverse and comprehensive medical information. We aim to improve the natural language processing capabilities to handle more complex queries and languages. Additionally, we plan to enhance the user interface further and integrate more advanced machine learning models to continually improve the accuracy and relevance of the information provided. Collaboration with healthcare professionals and organizations will also be pursued to ensure the system remains up-to-date with the latest medical knowledge and standards.

References

Zeltzer, D., Herzog, L., Pickman, Y., Steuerman, Y., Ber, R. I., Kugler, Z., Shaul, R., & Ebbert, J. O. (2023). Diagnostic accuracy of artificial intelligence in virtual primary care. Mayo Clinic Proceedings Digital Health, 1(4), 480–489. https://doi.org/10.1016/j.mcpdig.2023.08.002

Seminara, L. (2019). Accuracy of the WebMD Symptom Checker for Ophthalmic Diagnoses. Retrieved from www.aao.org: https://www.aao.org/eyenet/article/accuracy-of-the-webmd-symptom-checker

Symptom Checker with Body from WebMD - Check Your Medical Symptoms. (n.d.). WebMD. https://symptoms.webmd.com/

Health workforce shortage - doing the right things or doing things right? Croatian medical journal, 63(2), 107–109. https://doi.org/10.3325/cmj.2022.63.107

Symptom Checker at Patient. Check common symptoms online. (n.d.). Patient. https://patient.info/symptom-checker

Symptom checker. (2020, April 23). Mayo Clinic. https://www.mayoclinic.org/symptom-checker/chest-pain-in-adults-adult/related-factors/itt-20009075

Appendix A Glossary

- **AI (Artificial Intelligence):** The simulation of human intelligence processes by machines, especially computer systems, to perform tasks such as reasoning, learning, and problem-solving.

- **API (Application Programming Interface):** A set of rules and protocols that allows different software applications to communicate with each other.

- **HIPAA (Health Insurance Portability and Accountability Act):** A US legislation that provides data privacy and security provisions for safeguarding medical information.

- **SRS (Software Requirements Specification):** A document that specifies the requirements of a software system, including its features, functionalities, and limitations.

- **Symptom-Based Diagnosis:** A diagnostic approach that relies on analyzing symptoms reported by patients to identify potential health conditions.

- **User Interface (UI):** The visual elements and controls through which users interact with a software application.

- **User Manual:** Documentation that provides instructions and guidelines for users on how to use a software application effectively.

- **Machine Learning:** A subset of artificial intelligence that enables systems to learn and improve from experience without being explicitly programmed.

- **Data Encryption:** The process of encoding data in such a way that only authorized parties can access it.

- **Compliance:** Adherence to relevant regulations, standards, and best practices in healthcare information privacy, data security, and ethical AI use.

- **Integration:** The process of combining different software components or systems to work together as a unified whole.

- **Personalized Recommendations:** Recommendations tailored to individual user preferences, behaviors, and characteristics.

- **Medical History:** A record of a patient's past medical conditions, treatments, and interventions.

- **Symptom Analysis:** The process of evaluating and interpreting symptoms reported by patients to identify patterns and potential health issues.

- **User Acceptance Testing (UAT):** Testing conducted to verify that a software system meets the requirements and expectations of end-users.

- **Feature Enhancements:** Improvements or additions made to existing software features to enhance functionality or user experience.

- **Technical Support:** Assistance provided to users of a software application to help them troubleshoot issues or navigate challenges encountered during usage.

- **Healthcare Informatics:** The intersection of healthcare, information technology, and data science aimed at improving healthcare delivery and outcomes.

- **Regulatory Compliance:** Compliance with laws, regulations, policies and procedures, standards, and the other rules issued by governments and regulatory bodies.

Appendix B Deployment/Installation Guide

Appendix C User Manual

1. **Introduction**:
    - Welcome to the AI-powered medical search engine. This user manual provides detailed instructions on how to use the system effectively.
2. **Logging In**:
    - Open the application and enter your username and password.
    - Click on the login button to access the main dashboard.
3. **Searching for Medical Information**:
    - Enter your query or symptoms into the search bar.
    - Click on the search button to initiate the AI-powered search.
    - Review the generated response and follow any additional instructions provided.
4. **Accessing Detailed Information**:
    - After receiving a diagnosis, click on the "Detailed Information" button.
    - Review the detailed information, including treatment options and preventive measures.
5. **Updating Personal Health Records**:
    - Navigate to the personal health record section.
    - Enter new or updated health information and save the changes.

Appendix D Student Information Sheet

Roll No	Name	Email Address (FC College)	Frequently Checked Email Address	Personal Cell Phone Number
261947376	Asher David	261947376@formanite.fccollege.pk.edu		

Appendix E Plagiarism Free Certificate

This is to certify that, I am _____ S/D/o _____, group leader of FYP under registration no _____ at Computer Science Department, Forman Christian College (A Chartered University), Lahore. I declare that my Final year project report is checked by my supervisor and the similarity index is _____% that is less than 20%, an acceptable limit by HEC. Report is attached herewith as Appendix F. To the best of my knowledge and belief, the report contains no material previously published or written by another person except where due reference is made in the report itself.

Date: _____ Name of Group Leader: _____ Signature: _____

Name of Supervisor: _____ Co-Supervisor (if any):_____
Designation: _____ Designation: _____
Signature: _____ Signature: _____

Senior Project Management Committee Representative: _____
Signature: _____

Appendix F Plagiarism Report