Streamlining Healthcare: Multiple Disease Detection Web App for Convenient and Accurate Diagnoses

Niraj Sunil Bharambe, Sanjana Ramesh Bhangale, Ajay Namdev Phadke
Associate Professor, Student
University Of Mumbai

Abstract:

This application has been developed to facilitate individuals in finding suitable medical services with utmost convenience. In many instances, seeking medical assistance necessitates visiting multiple healthcare professionals, which can prove to be highly inconvenient. However, this inconvenience is now a thing of the past, thanks to the Multiple Disease Detection web app.

With this web app, people can easily obtain a diagnosis for their medical condition from the comfort of their own space, requiring only a few medical parameters. The Multiple Disease Detection web app harnesses the power of AI technologies to provide highly accurate results to users based on their input parameters. It determines whether a person is infected with a disease or not. This web app can diagnose diseases such as Cancer, Heart Disease, Lung Disease, Kidney Disease, Malaria, and Pneumonia. For Malaria diagnosis, the web app requires an image of the cell to predict whether it is infected with Malaria. Similarly, for Pneumonia diagnosis, it expects an X-ray image from the user.

Moreover, this web app also connects users with reliable and trustworthy medical professionals at affordable prices if a disease is diagnosed. If a user is disease-free, the website will provide the reassuring output, "You are healthy!"

The Multiple Disease Detection web app is designed with an easy-to-use User Interface (UI) and offers accurate predictions, significantly simplifying people's lives. In many cases, individuals seek a doctor's diagnosis, whether they have a medical issue or not. This can sometimes result in situations where those without any medical condition receive more attention and medical services than those who are affected by a disease. To prevent such occurrences and ensure swift disease diagnosis, this website can be a valuable resource.

Furthermore, this web app alleviates the burden on healthcare professionals, enabling them to focus more on patients who are indeed afflicted by specific diseases. The target audience for this application is mature adults who take responsibility for their health and actions when using Multiple Disease Detection.

This web app can also be a valuable tool for hospitals dealing with a high volume of patients daily. Manual diagnosis of all patients would be time-consuming, and many might not receive timely diagnoses. Multiple Disease Detection is a time-friendly solution, available for use at any time.

As of 2022, approximately 2.9 billion people are using medical websites. This underscores the potential impact and significance of this website in the healthcare landscape.
1. Free of Charge: The website is accessible to users without any associated costs.

2. Enhanced Safety and Security: Users can trust the website for its robust security measures, ensuring the protection of their personal information.

3. Minimal Investment: The website demands only a low investment from users, making it a cost-effective option.

4. Time and Cost Efficiency: Users can save both time and money by utilizing this efficient platform for their medical needs.

5. User-Friendly Accessibility: The website offers an easy-to-navigate interface, making it simple for users to access its services.

Chapter 1

1.1 preface

The operation of machine literacy in multiple complaint discovery represents a new approach to healthcare. This innovative approach aims to work machine literacy algorithms to give accurate prognostications regarding the liability of cases developing colorful conditions, taking into account their inheritable makeup, medical history, and other applicable factors. The primary ideal of this system is to enable early complaint opinion, thereby easing further effective treatments and perfecting patient issues. Machine literacy algorithms are particularly well-suited for complaint vaticination tasks because they can dissect expansive datasets of patient information, uncovering intricate patterns and connections that may not be incontinently apparent to mortal croakers. By checking data from different sources, including electronic medical records, medical imagery, and inheritable biographies, these algorithms can identify implicit complaint pointers that could else go undetected through conventional individual processes. The operation of machine literacy in complaint vaticination holds tremendous eventuality to revise the field of healthcare. It promises lesser perfection and personalization in treatment, before intervention, and bettered chances of recovery. still, this approach does come with certain challenges and limitations, including the necessity for different and representative data, enterprises related to algorithmic bias, and the demand for a focus on ethical considerations. Despite these challenges, the use of machine literacy in the vaticination of colorful conditions is a fleetly evolving field with a promising future. As technology continues to advance and as further data becomes available, machine literacy algorithms will come deceptively sophisticated and accurate. This, in turn, will affect in better case and healthcare issues. Machine literacy is at the van of rapid-fire advancements in computer wisdom, with a wide range of operations across colorful disciplines. It entails the birth of precious perceptive from expansive datasets and finds operations in different fields, including diagnostics, business, marketing, and scientific exploration. Machine literacy encompasses colorful ways, including bracket, retrogression, and clustering. In this environment, our primary focus lies on bracket ways, which are employed to classify data into predefined groups and read unborn events or information with a high degree of delicate and effectiveness.

1.2 Background

Healthcare and medical diagnostics have undergone remarkable advancements in recent years, with a growing emphasis on creating integrated solutions for detecting multiple diseases either simultaneously or sequentially. These advancements are driven by several significant factors:

1. Rising Disease Burden: Diseases such as diabetes, cancer, cardiovascular disease, kidney disease, liver disease, infectious diseases (e.g., malaria, pneumonia), and more have been on the rise globally. This growing burden places substantial pressure on healthcare systems, necessitating more efficient diagnostic and management strategies.
2. **Critical Role of Early Detection:** Research consistently demonstrates the benefits of early disease detection, which not only leads to improved patient survival rates but also reduces treatment costs and the overall impact on individuals and communities.

3. **Advancements in Medical Technology:** Innovations in medical technology, including highly sensitive diagnostic tests, advanced medical imaging techniques, genomics, proteomics, and the integration of artificial intelligence (AI) and machine learning in healthcare, have significantly expanded the possibilities for disease detection and risk assessment.

4. **Personalized Medicine:** Personalized medicine has gained prominence, recognizing that an individual's genetic makeup, lifestyle, and health history influence disease risk and response to treatment. Comprehensive detection of multiple diseases is a fundamental step in tailoring medical care to individual needs.

5. **Data-Driven Healthcare:** The availability of extensive healthcare data, encompassing electronic health records, data from wearable devices, and genomics information, has enabled the development of data-driven diagnostic and predictive models. These models use patient data to assess disease risk and guide clinical decisions.

6. **Public Health Priorities:** Addressing infectious diseases like malaria and pneumonia is pivotal for public health. Early detection, monitoring, and surveillance play essential roles in disease control and outbreak management.

7. **Efficiency and Resource Optimization:** Integrated systems for detecting multiple diseases offer the potential to streamline healthcare processes. They can reduce the need for redundant testing, shorten the time to diagnosis, and optimize resource allocation in healthcare facilities.

8. **Global Health Security:** The emergence of new and potentially pandemic-causing infectious diseases, exemplified by the COVID-19 pandemic, underscores the significance of rapid, accurate, and widespread disease detection for global health security.

9. **Access to Healthcare:** There is a growing acknowledgment of the necessity to expand healthcare access to underserved and remote populations. Technologies for multiple disease detection have the potential to democratize healthcare by providing cost-effective and efficient solutions deployable in various healthcare settings.

10. **Collaborative Research and Development:** Collaboration among healthcare providers, technology developers, research institutions, and public health agencies has propelled the development of integrated diagnostic systems, fostering innovation and technological advancements in the field.

The background of multiple disease detection is shaped by these factors, depicting a dynamic landscape in healthcare where innovation, data-driven approaches, and a focus on early detection and personalized care are the key drivers in addressing the diverse and complex healthcare challenges posed by a wide range of diseases. As technology and medical knowledge continue to advance, the development and deployment of systems for multiple disease detection become pivotal in improving healthcare outcomes and enhancing public health.
1.3 Purpose & Scope

1.3.1 Purpose

The purpose of developing a system for the detection of multiple diseases is to enhance healthcare outcomes, streamline medical diagnoses, and ultimately improve the well-being of individuals and communities. This technology aims to achieve the following critical objectives:

1. **Early Disease Identification:** Detecting a variety of diseases, including infectious, chronic, and rare conditions, at their earliest stages allows for timely and effective medical intervention. This early detection can significantly improve patient prognosis and reduce treatment costs.

2. **Personalized Medicine:** Tailoring treatment plans to an individual's specific disease profile and genetic makeup can optimize therapeutic outcomes and minimize adverse effects. Accurate disease detection is a crucial step in achieving this goal.

3. **Public Health Surveillance:** Comprehensive disease detection systems can aid in monitoring and managing outbreaks, thus safeguarding public health. Timely identification of communicable diseases can help control their spread and reduce the associated burden on healthcare systems.

4. **Cost Efficiency:** By diagnosing multiple diseases with a single system, healthcare providers can streamline diagnostic processes, reduce redundant testing, and optimize resource allocation, ultimately reducing healthcare costs.

5. **Improved Access to Healthcare:** These systems can be deployed in underserved and remote areas, extending healthcare access to populations that may have limited resources or medical expertise. This democratizes healthcare and reduces health disparities.

6. **Data-Driven Insights:** Multiple disease detection technologies generate valuable data that can be used for epidemiological research, tracking disease trends, and improving healthcare policies. These insights support evidence-based decision-making and public health strategies.

7. **Enhancing Telemedicine:** With the rise of telehealth and remote patient monitoring, integrated disease detection tools can empower healthcare professionals to remotely assess patients' health, make informed decisions, and offer timely guidance and treatment recommendations.

8. **Preventive Health Measures:** By identifying diseases early, patients can take proactive measures to manage their conditions, make necessary lifestyle changes, and reduce the risk of complications, thus promoting preventive healthcare.

9. **Global Health Security:** In an interconnected world, rapid detection of diseases, particularly emerging infectious threats, is essential for global health security. Early detection and containment can help prevent pandemics and protect the global population.

10. **Innovation and Research:** The development of multiple disease detection systems spurs innovation in the fields of medical technology, artificial intelligence, and data analytics. These advances have far-reaching implications beyond diagnostics, opening doors for breakthroughs in healthcare and medical research.

In summary, the purpose of multiple disease detection is to revolutionize healthcare by enabling early, accurate, and comprehensive disease identification, ultimately leading to improved patient outcomes, resource efficiency, and global health security. It is a critical tool in the ongoing mission to enhance healthcare access, quality, and effectiveness.
1.3.2 Scope

The scope for multiple disease detection, targeting diseases such as Diabetes, Breast Cancer, Heart Disease, Kidney Disease, Liver Disease, Malaria, and Pneumonia, is comprehensive and multifaceted, encompassing a range of objectives and considerations for improving healthcare and disease management. This scope includes:

1. **Disease-Specific Detection Methods**: Develop and implement specialized diagnostic methods and technologies tailored to each of the mentioned diseases. These methods may include blood tests, imaging techniques (e.g., mammography for breast cancer detection), and risk assessment tools specific to the disease in question.

2. **Cross-Disease Integration**: Explore opportunities for integrated diagnostics, where a single platform or system can detect multiple diseases simultaneously or sequentially. This approach can enhance efficiency and reduce the burden of separate diagnostic tests.

3. **Early Detection and Prevention**: Focus on early disease detection to enable timely intervention and prevention. Implement strategies for identifying risk factors and markers that indicate disease susceptibility, empowering healthcare professionals to take preventive measures.

4. **Data-Driven Predictive Models**: Develop data-driven models that use patient data, such as medical history, genetics, lifestyle factors, and clinical data, to predict the risk of developing these diseases. These models can facilitate personalized preventive care.

5. **Point-of-Care Testing**: Design and deploy point-of-care testing solutions for convenient and rapid diagnosis in various healthcare settings, including clinics, pharmacies, and even at-home testing kits.

6. **Remote Monitoring**: Implement remote monitoring technologies for managing chronic conditions (e.g., diabetes, heart disease, kidney disease) to track patients' health and enable timely intervention when needed.

7. **AI and Machine Learning**: Utilize artificial intelligence and machine learning algorithms for accurate disease detection, risk assessment, and predictive analytics. These technologies can enhance the speed and accuracy of diagnoses.

8. **Public Health Initiatives**: Integrate multiple disease detection into public health programs and surveillance systems to monitor and control the spread of diseases like Malaria and Pneumonia, particularly in regions with high prevalence.

9. **Population Screening**: Conduct systematic population screening programs for early detection of diseases with widespread impact, such as liver disease and breast cancer, with a focus on underserved communities.

10. **Treatment Tailoring**: Use disease detection data to guide personalized treatment plans for patients, optimizing therapeutic outcomes and minimizing adverse effects.

11. **Ethical Considerations**: Address ethical issues related to data privacy, consent, and responsible use of patient data in the context of multiple disease detection.

12. **Regulatory Compliance**: Ensure that all diagnostic methods and technologies comply with relevant medical and ethical regulations and standards.

13. **Research and Development**: Encourage ongoing research and development efforts to refine and expand the capabilities of multiple disease detection methods, fostering innovation and advancement in healthcare.

14. **Healthcare Collaboration**: Promote collaboration between healthcare providers, researchers, technology developers, and governmental health agencies to facilitate the development, deployment, and continuous improvement of these disease detection systems.
In summary, the scope for multiple disease detection of Diabetes, Breast Cancer, Heart Disease, Kidney Disease, Liver Disease, Malaria, and Pneumonia is diverse and extensive. It encompasses a range of diagnostic methods, preventive measures, data-driven technologies, and public health initiatives, all with the overarching goal of improving healthcare outcomes, reducing disease burden, and enhancing overall public health.

1.4 Objectives

Due to significant advancements in medical science, many cures for diseases have been discovered. Proper treatment at the right time can help individuals overcome dangerous diseases. However, with the increasing number of patients, hospitals are often overcrowded, making it challenging to receive a timely diagnosis. To address this issue and reduce the death rate resulting from late disease diagnosis, we have developed Mediconnect. This platform allows individuals to determine their potential illnesses with just a few medical parameters, enabling them to seek early treatment. In densely populated cities like Jharkhand and Uttar Pradesh, where there is often just one doctor for every 1200 people, this technology can significantly reduce the burden on healthcare professionals. Mediconnect facilitates rapid and early diagnosis, allowing doctors to focus their attention on patients who are infected, rather than those who only exhibit symptoms but aren't infected.

![Fig 1 - Solution for unavailability of doctors](image)

1.5 Existing System Feasibility Study

A feasibility study serves as a condensed overview of the entire system analysis and design process. It starts with defining the problem. The feasibility study's objective is to determine whether the project is worth pursuing. Once the problem definition is accepted, a logical system model is developed, and alternative solutions are thoroughly analyzed. The feasibility study consists of three key parts:

1.5.1 Operational Feasibility

Operational feasibility assesses how well a proposed system addresses identified problems, takes advantage of opportunities identified during scope definition, and fulfills the requirements from the requirements analysis phase of system development. This assessment focuses on the degree to which the development project fits into the existing business environment, considering development schedule, delivery date, corporate culture, and existing business processes. It is crucial to impart desired operational outcomes during the design and development phases, including parameters like reliability, maintainability, supportability, usability, productibility, disposability, sustainability, affordability, and others.
1.5.2 Technical Feasibility

Technical feasibility explores whether the required technology for the system exists, how challenging it will be to build, and whether the organization possesses sufficient experience with that technology. The assessment is based on the system's outline design, considering input, processes, output, fields, programs, and procedures. This can be expressed in terms of data volume, trends, and update frequency to provide an introduction to the technical system. In this case, the application is built on the Windows XP platform with high configuration, making it technically feasible. The assessment focuses on understanding the present technical resources of the organization and their suitability for the proposed system.

1.5.3 Economical Feasibility

Economical feasibility aims to establish whether the proposed system is cost-effective. It evaluates if the benefits of the system outweigh the costs. Given the growing need for online social networking facilities in today's fast-paced world, this project is economically feasible. The purpose of this assessment is to determine the positive economic benefits the system will provide to the organization. It involves quantifying and identifying all expected benefits and typically includes a cost/benefit analysis.

Chapter 2: System Analysis

2.1 Feasibility Study

A feasibility study is a systematic and thorough examination conducted to determine the possibility or probability of enhancing an existing system or creating an entirely new one. This study involves a comprehensive review of the existing system, aiming to identify its limitations and disadvantages. Once the management accepts the proposal for the study, it initiates an investigation into the existing system or problem area. This investigation is conducted in close collaboration with user management and provides in-depth insights to establish the technical, operational, and economic feasibility of the proposed changes.

The purpose of an AI-based application is to keep users up-to-date and assist them in their day-to-day activities. It enables users to work more efficiently and rapidly, saving both time and money. An AI-based medical website allows users to perform self-diagnosis by simply entering relevant information, eliminating the need for lengthy queues in hospitals. Our website simplifies the process of self-diagnosis with just a single click, providing users with the convenience of finding disease-related information and potential cures from the comfort of their homes.

2.2 Developed Systems

System development is the process of defining, designing, testing, and implementing new software applications or programs. It encompasses activities such as the internal development of customized systems, the creation of database systems, or the acquisition of third-party developed software. The system development process includes practices related to designing and implementing the desired software systems, aiming to achieve specific goals and objectives. This process is structured to realize the development goals and undergo system verification for a successful outcome. The organization of the system development process is oriented toward the classification of system development objectives at various levels.
Models with their Accuracy of Prediction

<table>
<thead>
<tr>
<th>Disease</th>
<th>Type of Model</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diabetes</td>
<td>Machine Learning Model</td>
<td>98.25%</td>
</tr>
<tr>
<td>Breast Cancer</td>
<td>Machine Learning Model</td>
<td>98.25%</td>
</tr>
<tr>
<td>Heart Disease</td>
<td>Machine Learning Model</td>
<td>85.25%</td>
</tr>
<tr>
<td>Kidney Disease</td>
<td>Machine Learning Model</td>
<td>99%</td>
</tr>
<tr>
<td>Liver Disease</td>
<td>Machine Learning Model</td>
<td>78%</td>
</tr>
<tr>
<td>Malaria</td>
<td>Deep Learning Model(CNN)</td>
<td>96%</td>
</tr>
<tr>
<td>Pneumonia</td>
<td>Deep Learning Model(CNN)</td>
<td>95%</td>
</tr>
</tbody>
</table>

2.3 Requirements

A Software Requirements Specification (SRS) is a comprehensive document that outlines the description of a software system to be developed. It is typically created after the Business Requirements Specification (CONOPS) or Stakeholder Requirements Specification (STRS) and may also be related to the System Requirements Specification (SYRS).

2.4 Hardware and Software Requirements

All computer software requires specific hardware components or other software resources to be available on a computer. These prerequisites are commonly referred to as system requirements and are usually considered as guidelines rather than strict rules. Software often specifies two sets of system requirements: minimum and recommended. As software evolves and demands higher processing power and resources, the system requirements tend to increase. This trend is a significant driver for upgrading existing computer systems, alongside technological advancements. Another interpretation of system requirements is a broader one, defining the requirements for the design of a system or subsystem.

2.4.1 Hardware Requirements
- System Processor: Intel Core i3.
- Hard Disk: 512 GB HDD.
- Monitor: 15" LED.
- Mouse: Optical Mouse.
- RAM: 4.0 GB.

2.4.2 Software Requirements
- Coding Language: Python 3.7.
- Front-End: Flask, Python.
- Python Modules: TensorFlow.

(i) Hardware Interface
Hardware interfaces define the necessary hardware devices for the website to run, such as the type of processor and required memory.

a) Android phones, iOS devices, laptops, PCs, and other compatible devices.
b) A minimum of 35 MB of storage on the device.
Software interfaces define the required software components for our website to operate, primarily including the operating system and internet connectivity.

a) Smartphones or other devices with a suitable operating system.
b) Internet access for fetching results from the API server.
c) Adequate disk space.
d) Sufficient battery percentage to continue making predictions.

2.5 Functional Requirements

Functional requirements define the internal workings of the software, encompassing calculations, technical details, data manipulations, processing, and specific functionalities. They specify how user requirements are implemented and how the website interacts with users, making it more user-friendly and accessible. Functional requirements detail a specific behavior or operation of a system. Typically, each functional requirement includes a unique identifier, a brief summary, and a rationale. This information helps readers understand why the requirement is necessary and enables tracking throughout the development process. Functional requirements ensure the proper functioning of the website, including its behavior, readability of fonts, and other user-oriented aspects. This behavior can be derived from organizational and business rules or identified by experts. Functional requirements must be clear, correct, unambiguous, specific, and verifiable. The fundamental functional requirements of the system are as follows:

- **Website View**: The website view is highly interactive, displaying disease details with proper names and symptoms. This feature assists users in easily matching their symptoms with specific diseases for further prediction tests.

- **Navigation View**: The navigation view is a crucial component of our website, containing links that redirect users to specific disease pages when clicked.

- **Predict Through Images**: Our website allows users to upload images, such as X-rays or cell images from sonography reports, to determine whether they are infected or not. This functionality is particularly designed for two diseases: Pneumonia and Malaria.

2.6 Non-Functional Requirements

In systems engineering and requirements engineering, non-functional requirements specify the criteria for evaluating the system’s performance rather than its specific behaviors. They differ from functional requirements, which detail specific functional behaviors. Common non-functional requirements include factors like reliability, scalability, and cost. Non-functional requirements are often referred to as the utilities of a system. The primary non-functional requirements for the system are:

- **Secure Access to Confidential Data**: Ensuring secure access to users' confidential information.

- **24x7 Availability**: The system must be available around the clock.

- **Better Component Design for Peak Performance**: Optimized component design to achieve superior performance during peak usage times.

- **Flexible Services-Based Architecture**: A flexible architecture for future extensions.

- **Regular Website and Data Updates**: Keeping the website and its data up-to-date.

- **Effective User Information Management**: Proper management of user information.

- **User Engagement**: Keeping users informed and engaged with the website.
- Data Transfer to Medical Professionals: Sending data in the form of Excel sheets to the relevant medical professionals after accurate disease prediction.

2.7 Software Environment

Python:
Python is a high-level, interpreted, interactive, and object-oriented scripting language. Python is designed for readability, using English keywords extensively, and it has a simpler syntax compared to many other languages. Some key characteristics of Python are:

- Interpreted: Python is processed at runtime by the interpreter, so you don't need to compile your program before executing it.
- Interactive: Python allows you to interact directly with the interpreter, making it easy to write and test your code.
- Object-Oriented: Python supports object-oriented programming, allowing you to encapsulate code within objects.
- Beginner-Friendly: Python is an excellent language for beginners, accommodating a wide range of applications from simple text processing to web browsers and games.

History of Python:
Python was developed by Guido van Rossum in the late eighties and early nineties. It was created at the National Research Institute for Mathematics and Computer Science in the Netherlands. Python is influenced by several other languages, including ABC, Modula-3, C, C++, Algol-68, Smalltalk, and Unix shell scripting. It is now available under the GNU General Public License (GPL) and maintained by a core development team.

Python Features:
Python's features include:

- Easy-to-learn: Python has a minimal number of keywords, simple structure, and clear syntax, making it easy for beginners to pick up.
- Easy-to-read: Python code is highly readable and well-structured.
- Easy-to-maintain: Python's source code is easy to maintain.
- Broad standard library: Python has a rich library that is portable and cross-platform.
- Interactive Mode: Python supports an interactive mode for testing and debugging code.
- Portability: Python can run on various hardware platforms and maintains the same interface.
- Extendability: Python allows low-level module integration for added efficiency.
- Databases: Python offers interfaces to major commercial databases.
- GUI Programming: Python supports GUI applications across various systems, including Windows MFC, Macintosh, and X Window system.
- Scalable: Python provides a better structure for large programs than shell scripting.

Python also supports functional and structured programming, OOP, can be used as a scripting language, and provides dynamic data types, automatic garbage collection, and integration with other languages.

Getting Python:
The most current Python source code, binaries, documentation, and updates can be found on the official Python website at [https://www.python.org](https://www.python.org).

Windows Installation:
To install Python on a Windows machine, follow these steps:

2. Click on the Windows installer (python-XYZ.msi) where XYZ is the version you want to install.
3. Download the installer file to your local machine and run it to determine if your system supports MSI.
4. Run the downloaded file, which launches the Python installation wizard. Accept the default settings, wait for the installation to complete, and you're done.
First Python Program:
Python programs can be executed in different modes. In interactive mode, invoking the interpreter without a script file parameter brings up a prompt.

```
$ python
Python2.4.3(#1,Nov112010,13:34:43)
[GCC 4.1.220080704(RedHat4.1.2-48)] on linux2
Type"help","copyright","credits"or"license"for more information.
```  

Type the following text at the Python prompt and press the Enter –

```
>>> print"Hello, Python!"
```  

If you are running new version of Python, then you would need to use print statement with parenthesis as in print ("Hello, Python!");. However in Python version 2.4.3, this produces the following result –

```
Hello, Python!
```  

Flask Framework:

Flask is a web application framework written in Python. It is developed by Armin Ronacher, who leads an international group of Python enthusiasts named Pocco. Flask is built on the Werkzeug WSGI toolkit and uses the Jinja2 template engine, both of which are also Pocco projects. The HTTP protocol forms the foundation of data communication on the World Wide Web, defining various methods for retrieving data from specified URLs.

What is Flask?

Flask is a Python API that enables the development of web applications. It was created by Armin Ronacher. Flask's framework is more explicit than Django's and is easier to learn because it requires less base code to build a simple web application. A web application framework, or web framework, is a collection of modules and libraries that aids developers in writing applications without having to deal with low-level coding such as protocols and thread management. Flask is built on the WSGI (Web Server Gateway Interface) toolkit and uses the Jinja2 template engine.

Getting Started with Flask:

To install Flask, you need Python 2.6 or a higher version. You can start by importing Flask from the Flask package in any Python IDE. For installation on any environment, you can follow the installation link provided below. To test if the installation is working, you can use the following code.
2.8 JUSTIFICATION OF SELECTED TECHNOLOGY

In the present age, applications have gained significant importance in the fields of Public Health and healthcare. However, the creation of websites has become a challenging task. In our AI-Based Mediconnect website, we have chosen to use the Flask framework to address this challenge. We have developed a user-friendly interface that allows users to input their symptoms and receive a diagnosis from the AI model.

Flask is an application framework in Python that has become a standard for Python web application development. It is built on the Werkzeug WSGI toolkit and uses the Jinja2 template engine. The Web Server Gateway Interface (WSGI) has a templating engine for Python, which combines a template with data sources to render dynamic web pages.

Flask is often referred to as a "micro framework" as it aims to keep the core of an application simple yet extensible. Unlike some other frameworks, Flask does not have a built-in abstraction layer for database handling or form validation support. Instead, it supports the use of extensions to add such functionality to the application. Flask is Pythonic in nature and is known for its simplicity and readability, making it a great choice for getting started quickly with web development.

When properly designed, implemented, and utilized, Health Information Technology (HIT) can play a pivotal role in transforming digital public health delivery. HIT interventions, including health apps, have the potential to enhance the performance and quality of healthcare services, reduce costs, and actively engage patients in managing their own healthcare.

For our Flask application, we set up a development environment, which involved installing necessary components. Flask's framework simplifies the process of predicting diseases and we also utilize the RandomForest for prediction. Flask is a collection of libraries and modules that empowers web application developers to create applications without the need to be concerned about low-level details such as protocols and thread management, making it an ideal choice for our project.
Chapter 3: ANALYSIS AND DESIGN

3.1 INFORMATION GATHERING

Information gathering is the process of collecting various types of data and details relevant to the targeted individual or system. The primary objective of information gathering is to identify the information needs of an organization or its users. In our project, information gathering primarily involves the analysis of patient information to diagnose diseases. This begins with the patient sharing their clinical history and symptoms for diagnosis.

![Fig 2: Work flow of algorithm](image)

Information gathering encompasses the following areas:

1. **Functionalities** – This involves analyzing the essential requirements of our project, such as the core functionalities of our website. It covers the overall operation of webpages and the website.

2. **APIs** – In this section, we provide information about the APIs used in our website.

3. **Front-end Design** – The analysis gives us insights into better interactivity and helps us create a user-friendly interface. It guides us in maintaining a logical flow. Front-end design focuses on the user interface elements, including components, menus, home pages, links, icons, images, and more that a webpage must have for improved interactivity. It also includes considerations for color combinations to enhance user interaction. We aim to address all the user expectations for a website to make it user-friendly.

In summary, this section delves into the core functionalities, APIs, and the front-end design of our project. It also takes into account the rules and regulations associated with systems and tools, ensuring that the project will elicit a positive response from its users.
3.2 UML DIAGRAM

The Unified Modeling Language (UML) is a methodology for detailing system architecture with a blueprint that provides a comprehensive representation of the system's structure. UML encompasses a set of best engineering practices proven successful for modeling complex and large-scale systems. It plays a vital role in developing object-oriented software and the software development process. UML primarily employs graphical notations to convey the design of software projects, allowing project teams to communicate, explore potential designs, and validate the architectural design of the software.
3.3 ACTIVITY DIAGRAM

Activity diagrams serve to describe the workflow behavior of a system. They are akin to state diagrams in that activities represent the state of actively doing something. These diagrams illustrate the state of activities by presenting the sequence of performed activities. Activity diagrams can depict conditional or parallel activities, thus providing a visual representation of the workflow.
This is basic Machine Learning and Deep Learning based WebApp. These Machine learning models and Deep learning models are trained on large datasets, and thousands of images.

**Model Accuracies:**
- Diabetes Model: 98.20%
- Breast Cancer Model: 98.31%
- Heart Disease Model: 98.33%
- Kidney Disease Model: 98.29%
- Liver Disease Model: 98%
- Pneumonia Model: 98%

**Information about the Diseases which this webApp can predict.**

**Diabetes**
Diabetes is a disease that occurs when your blood glucose, also called blood sugar, is too high. Most glucose comes from the food you eat. Insulin, a hormone made by the pancreas, helps glucose from food get into your cells to be used for energy. Sometimes, your body doesn’t make enough—or any—insulin or doesn’t use insulin well. Glucose then stays in your blood and doesn’t get into your cells.

**Symptoms:**
- Unusual tiredness,
- Feeling very thirsty,
- Extreme fatigue,
- Blurry vision.
### Heart Disease Predictor

- **Age**
- **Systolic blood pressure in mm Hg**
- **Diastolic blood pressure in mm Hg**
- **Serum cholesterol in mg/dL**
- **Serum triglyceride in mg/dL**
- **Serum glucose level in mg/dL**
- **Heart rate in beats per minute**
- **Hypertension diagnosis**
- **Family history of heart disease**
- **Excess weight index (BMI)**
- **Physical activity level**
- **Smoking status**
- **Alcohol consumption**
- **Other risk factors**

### Kidney Disease Predictor

- **Age**
- **Sex**
- **Height (in cm)**
- **Weight (in kg)**
- **Blood pressure (in mm Hg)**
- **Glomerular filtration rate (GFR)**
- **Urine albumin (in mg/L)**
- **Urine creatinine (in mg/dL)**
- **Urinary tract infection history**
- **Family history of kidney disease**
- **Diabetes mellitus diagnosis**
- **Obesity**
- **Other risk factors**

### Liver Disease Predictor

- **Age**
- **Direct Bilirubin**
- **Indirect Bilirubin**
- **Alkaline Phosphatase**
- **Aspartate Aminotransferase**
- **Alanine Aminotransferase**
- **Total Proteins**
- **Albumin**
- **Glucose and G lucose Ratio**
- **Gender (Male: Female)**

### Multiplier Disease Predictor

- **Age**
- **Gender**
- **Body Mass Index (BMI)**
- **Blood pressure (Systolic/Diastolic)**
- **Cholesterol (Total/LDL/HDL)**
- **Glucose level (in mg/dL)**
- **Uric Acid level (in mg/dL)**
- **Liver function tests**
- **Kidney function tests**
- **Smoking status**
- **Exercise level**
- **Other health conditions**
5.0 Conclusion

In conclusion, the preceding chapters provide a comprehensive overview of the project, "AI-Based Mediconnect - Multiple Disease Detection." This project focuses on utilizing machine learning and web-based technologies to enable early disease diagnosis and improve healthcare outcomes. **Chapter 1** introduced the project by emphasizing the significance of using machine learning to predict various diseases and the challenges associated with it. The purpose, scope, and objectives of the project were outlined, highlighting the potential to revolutionize healthcare by enhancing early diagnosis, personalized medicine, and global health security. **Chapter 2** delved into the system analysis, requirements, and software environment. It detailed the hardware and software prerequisites, and emphasized the importance of using Python, Flask, and other technologies to build the project. It also justified the choice of Flask as the web application framework for its simplicity and extensibility. **Chapter 3** focused on the analysis and design phase. Information gathering was highlighted, which involved understanding the functional requirements, APIs, and front-end design. The use of UML diagrams and activity diagrams for system modeling was explained, underscoring their significance in visualizing the project's architecture and workflow. Throughout these chapters, it became clear that the AI-Based Mediconnect project aims to bring about a transformative change in the healthcare sector by providing a user-friendly, efficient, and interactive platform for disease detection and diagnosis. The culmination of these project aspects sets the stage for the development and implementation phases, where the vision of early disease identification and improved healthcare will come to fruition.
6.0 References


43. Singh H., Navaneeth N., Pillai G. Multisurface Proximal SVM Based Decision Trees For Heart Disease Classification; Proceedings of the TENCON 2019–2019 IEEE Region 10 Conference (TENCON); Kerala, India. 17–20 October 2019; pp. 13–18. [Google Scholar]


111. Sun M., Huang Z., Guo C. Automatic Diagnosis of Alzheimer’s Disease and Mild Cognitive Impairment Based on CNN+ SVM Networks with End-to-end Training; Proceedings of the 2021 13th International Conference on Advanced Computational Intelligence (ICACI); Wanzhou, Chongqing, China. 14–16 May 2021; Manhattan, NY, USA: IEEE; 2021. pp. 279–285. [PMC free article] [PubMed] [Google Scholar]


