

INTERNSHIP REPORT

*A Report Submitted in partial fulfilment of the requirements
for the award of the degree of*

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING – DATA SCIENCE

Submitted by,

SAI BHARGAV SHIVAMPETA

[Reg. No: 20J41A6751]

Under the Supervision of

Ms. Sahithi Reddy

Full Stack Engineer, Path Creators, Hyderabad

(Duration: 15th May, 2023 to 30th May, 2023)



COMPUTER SCIENCE AND ENGINEERING - DATA SCIENCE

MALLA REDDY ENGINEERING COLLEGE

(An Autonomous Institution)

Maisammaguda, Secunderabad, Telangana, India 500100

May - 2023

MALLA REDDY ENGINEERING COLLEGE

Maisammaguda, Secunderabad, Telangana, India 500100



CERTIFICATE

This is to certify that the “Internship Report” submitted by **SAI BHARGAV SHIVAMPETA (Roll No: 20J41A6751)** is work done by him and submitted during 2023 – 2024 academic year, in partial fulfilment of the requirements for the award of the degree of **BACHELOR OF TECHNOLOGY** in Computer Science and Engineering - Data Science, at Path Creators, Hyderabad, Telangana.

SIGNATURE

Mrs. R. Sravani

INTERNSHIP COORDINATOR

Asst. Professor

CSE – Data Science

Malla Reddy Engineering College

Secunderabad, 500 100

SIGNATURE

Dr. S. Shiva Prasad

HOD

Professor

CSE – Data Science

Malla Reddy Engineering College

Secunderabad, 500 100


OVERVIEW OF INTERNSHIP ACTIVITY

WEEK – 1

Date	Day	Details of the Internship Activity Performed/ Completed
May 15	1	Basics of Python Programming and Data Types
May 16	2	Python Conditional Statements and Loops
May 17	3	Python Functions and Modules
May 18	4	Basics of Full Stack
May 19	5	HTML, CSS, JavaScript
May 20	6	Basics of JavaScript

NUGATORY INHERITS
LUMINESCENCE
TECHNOLOGIES PVT. LTD.
CIN: U74999UP2017PTC097008

Organization Seal


Signature of the Supervisor

WEEK – 2

Date	Day	Details of the Internship Activity Performed/ Completed
May 21	7	Integration of Backend with Frontend
May 22	8	Project Oriented Modules
May 23	9	Flask Modules
May 24	10	APIs
May 25	11	APIs
May 26	12	Python Data Base

NUGATORY INHERITS
LUMINESCENCE
TECHNOLOGIES PVT. LTD.
CIN: U74999UP2017PTC097008

Organization Seal



Signature of the Supervisor

WEEK – 3

Date	Day	Details of the Internship Activity Performed/ Completed
May 27	13	Python Problem Solving
May 28	14	Python Problem Solving
May 29	15	Project Explanation
May 30	16	Project Explanation

NUGATORY INHERITS
LUMINESCENCE
TECHNOLOGIES PVT. LTD.
CIN: U74999UP2017PTC097008

Organization Seal



Signature of the Supervisor



PATH CREATORS

Sri Valkunta Technologies Pvt Ltd.

www.PathCreators.in



Certificate ID: PTHCRT_1658

INTERNSHIP CERTIFICATE

This certificate is presented to SAI BHARGAV SHIVAMPETA of MREC (MALLA REDDY ENGINEERING COLLEGE) bearing Hall Ticket No. 20J41A6751 from CSE-DS Department for successful completion of Internship during the Duration of 2023-05-15 to 2023-05-30. SAI BHARGAV SHIVAMPETA was very keen to learn and understand the concepts during the Internship.

The Internship was focused on Web Technologies like HTML, CSS, Javascript, Python Programming, Flask Server and more. SAI BHARGAV SHIVAMPETA also successfully completed selected Projects.

We wish SAI BHARGAV SHIVAMPETA all the best for future endeavors.

PATH CREATORS



Sahithi Reddy
Full Stack Engineer

ACKNOWLEDGEMENT

I want to express my sincere gratitude to **Mr. Ashish Chinthakayala**, the Head of the Company at Hyderabad, for granting me the invaluable opportunity to intern within their esteemed organization. Additionally, I extend my heartfelt thanks to all the colleagues who collaborated with me during my time at **Path Creators**. Their patience and openness fostered a truly enjoyable work environment.

I would like to extend my sincere thanks to **Ms. Sahithi Reddy**, a Full Stack Engineer at Path Creators. She played a pivotal role in our internship experience by serving as a mentor throughout the duration of our time at the organization. Her expertise, guidance, and unwavering support significantly contributed to our professional growth and understanding of the intricate aspects of full-stack development.

I am filled with a profound sense of pleasure and gratitude as I reflect on the invaluable assistance provided by these individuals. I owe a great deal of thanks to our Respected Director **Dr. A. Ramaswamy Reddy** for facilitating this internship, as well as to my Department Head, **Dr. S. Shiva Prasad**, for offering constructive feedback throughout my internship journey.

I am also deeply appreciative of **Mrs. R. Sravani**, the Department's Internship Coordinator, for their unwavering support and guidance in securing and successfully completing my internship at the aforementioned organization. My heartfelt thanks also go out to the members of my department and my friends who played a crucial role in ensuring the successful culmination of this internship experience.

Sai Bhargav Shivampeta

20J41A6751

ABSTRACT

In today's rapidly evolving healthcare landscape, effective doctor-patient communication and streamlined appointment scheduling stand as cornerstones of quality patient care. This paper presents a revolutionary Flask-based web application known as the Doctor-Patient Interaction and Appointment Management WebApp (DPI WA), specifically engineered to revolutionize the way healthcare providers and patients engage with one another. By offering an intuitive, user-friendly platform, DPI WA aims to transform the healthcare experience into a seamless, efficient, and personalized journey.

Through DPI WA, both healthcare providers and patients gain access to a secure environment, where they can easily register and log in. Once inside, users are greeted by personalized dashboards that grant them comprehensive control over their healthcare information. This includes the ability to schedule appointments at their convenience and establish direct lines of communication with healthcare providers. This innovative system not only simplifies the process of healthcare data management but also empowers individuals to take charge of their own well-being.

The significance of DPI WA extends far beyond its convenience. By democratizing access to comprehensive healthcare tools, it bridges gaps between medical professionals and patients, fostering an environment of mutual understanding and trust. Through this application, patients can actively participate in their own care, leading to improved health outcomes. Moreover, medical professionals can allocate more time to patient care, alleviating the administrative burdens that often accompany traditional appointment scheduling methods.

In essence, DPI WA represents a quantum leap forward in healthcare management. It prioritizes patient-centric care, ensuring that individuals receive the attention and resources they need to lead healthy, fulfilling lives. By harnessing the power of technology, this application transcends geographical barriers and time constraints, ultimately paving the way for a more connected, efficient, and effective healthcare system.

INDEX

S.No.	Contents	Page No.
	ACKNOWLEDGEMENT	7
	ABSTRACT	8
1.	About the Organization (Background Information)	10
2.	Objective of the Internship	12
3	Requirement Analysis	13
4	Software Requirement Specifications	14
5	Activity Diagram	15
6	System Analysis	17
7	Feasibility	19
8	Python	23
9	Flask	26
10	HTML	28
11	CSS	30
12	Runtime Output Screenshots	32
13	Conclusion	34
	APPENDICES	35
	REFERENCES	37

1. ABOUT THE ORGANIZATION



Path Creators is a Hyderabad based start-up which provides mentorship on latest technologies emerging in the market by experienced resources. Providing world class infrastructure to our associates, we also give them a huge platform to explore, interact and showcase their ideas with various industries and investors. Working with us gives you a vision on how real time practises occur. We train both fresher's and experienced corporates by conducting various hackathons, workshops and also provide facility of Innovation club in various institutions. At Path Creators, We thrive on bringing associate's ideas into reality, helping them to achieve their mission with our vision.

Key aspects of our approach at Path Creators include:

1. World-Class Infrastructure: We understand the importance of a conducive work environment. That's why we provide world-class infrastructure to our associates. This sets the stage for productive learning and interaction.

2. Platform for Exploration: Our associates are not just mentees; they are individuals with ideas and innovations waiting to be explored. We provide a significant platform for them to interact with various industries and investors. This interaction helps them showcase their ideas, seek feedback, and potentially find avenues for funding and collaboration.

3. Real-World Exposure: Working with us isn't just about theoretical knowledge. We aim to provide a practical understanding of how real-time practices occur in the tech industry. This real-world exposure is invaluable for those looking to make a meaningful impact in the technology sector.

4. Diverse Training: We cater to a wide audience, including freshers and experienced corporate professionals. Our training methodology includes hackathons, workshops, and the

establishment of Innovation clubs in various institutions. This diverse training approach ensures that individuals at different stages of their careers can benefit from our expertise.

5. Idea Implementation: We are passionate about turning ideas into reality. We actively support our associates in bringing their ideas to life. Our goal is to help them achieve their mission, and we do this by providing a clear vision and the resources necessary for success.



Our vision is to educate & promote all entrepreneurship related stakeholders, including start-ups, investors, incubators and accelerators, research centres, start up resources, projects developing a nurturing environment to technology based business ideas to prosper and become viable contributors to our community and economy. We also bring together partnerships from different verticals to help you shorten the sprint towards your goal.

We also understand that partnerships are vital in the journey toward success. Therefore, we actively bring together partners from various verticals to help our associates shorten the distance to their goals. By facilitating collaborations and connections, we aim to expedite the journey of entrepreneurs and innovators.

In summary, Path Creators is more than a startup; it's a launchpad for aspiring technologists and entrepreneurs. We are driven by the vision of fostering innovation and providing the necessary resources for the growth of the technology sector, contributing to our community and economy. We are here to guide, mentor, and support those who dare to innovate and make a difference.

2. OBJECTIVES OF THE INTERNSHIP

- **Gain Practical Experience:** One of the primary objectives is to gain hands-on experience in applying data science techniques to real-world problems. This includes working with real data, using data analysis tools, and implementing ML models.
- **Learn Industry-specific Skills:** Interns often aim to acquire industry-specific skills and knowledge. This could include understanding the domain knowledge relevant to the organization they are interning with, such as healthcare, finance, e-commerce, etc.
- **Networking:** Building professional relationships and within the industry is another important. This includes connecting with colleagues, mentors, who can provide guidance and potentially open up future job opportunities.
- **Project Work:** Completing meaningful projects is often a key component of a data science internship. These projects allow interns to demonstrate their skills and contribute to the organization's goals.
- **Problem Solving:** Developing problem-solving skills is crucial. Interns should learn how to identify and define data-driven problems, design appropriate solutions, and iterate through the problem-solving process.
- **Teamwork and Collaboration:** Learning to work effectively in a team is important in many data science roles. Interns may aim to improve their teamwork and collaboration skills during their internship.
- **Exposure to Tools and Technologies:** Familiarity with data science tools, libraries, and technologies commonly used in the industry, such as Python, R, TensorFlow, or scikit-learn, is often a goal of data science internships.
- **Feedback and Self-Improvement:** Interns should seek feedback from supervisors and mentors to continuously improve their skills and knowledge in data science.

3. REQUIREMENT ANALYSIS

EXISTING SYSTEM

In the current landscape, there are indeed various applications designed to facilitate doctor-patient communication and appointment scheduling. However, a prevailing issue is that a significant portion of these solutions operate on a paid subscription model, which may limit accessibility for both healthcare providers and patients. Additionally, while there are open-source alternatives available, there is a notable absence of options that offer a combination of user-friendly customization and an easily modifiable interface. This gap in the existing system underscores the need for an innovative and adaptable solution that caters to the diverse needs of healthcare professionals and patients alike.

PROPOSED SYSTEM

The Flask-Based Doctor-Patient Interaction and Appointment Management WebApp (DPIMWA) addressed in this paper represents a pivotal advancement in healthcare technology. By seamlessly integrating healthcare data management and appointment scheduling within a user-friendly interface, DPIMWA is poised to revolutionize the way healthcare providers and patients engage with each other. This proposed system addresses the shortcomings of the existing solutions by offering a comprehensive, accessible, and highly customizable platform that can be tailored to individual preferences and organizational requirements.

The introduction of DPIMWA signifies a paradigm shift in healthcare management. It transcends the limitations of paid applications, democratizing access to critical healthcare tools. Moreover, its open-source nature ensures that it can be easily modified and adapted to suit the specific needs of various healthcare settings.

By fostering more efficient and effective doctor-patient interactions, DPIMWA has the potential to significantly enhance the overall quality of healthcare delivery. In essence, the proposed system not only bridges the gaps in the existing landscape but also paves the way for a new era of healthcare excellence.

4. SOFTWARE REQUIREMENT SPECIFICATIONS

SYSTEM CONFIGURATIONS

The software requirement specification can produce at the culmination of the analysis task.

The function and performance allocated to software as part of system engineering are refined by established a complete information description, a detailed functional description, a representation of system behavior, and indication of performance and design constrain, appropriate validate criteria, and other information pertinent to requirements.

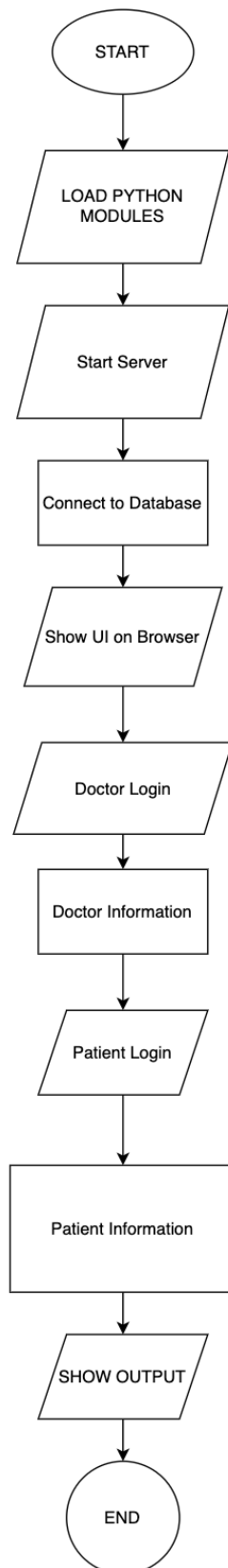
HARDWARE REQUIREMENTS

- **Processor** : i3 7th Gen or better
- **RAM** : 8GB or better
- **Storage** : 120GB or more

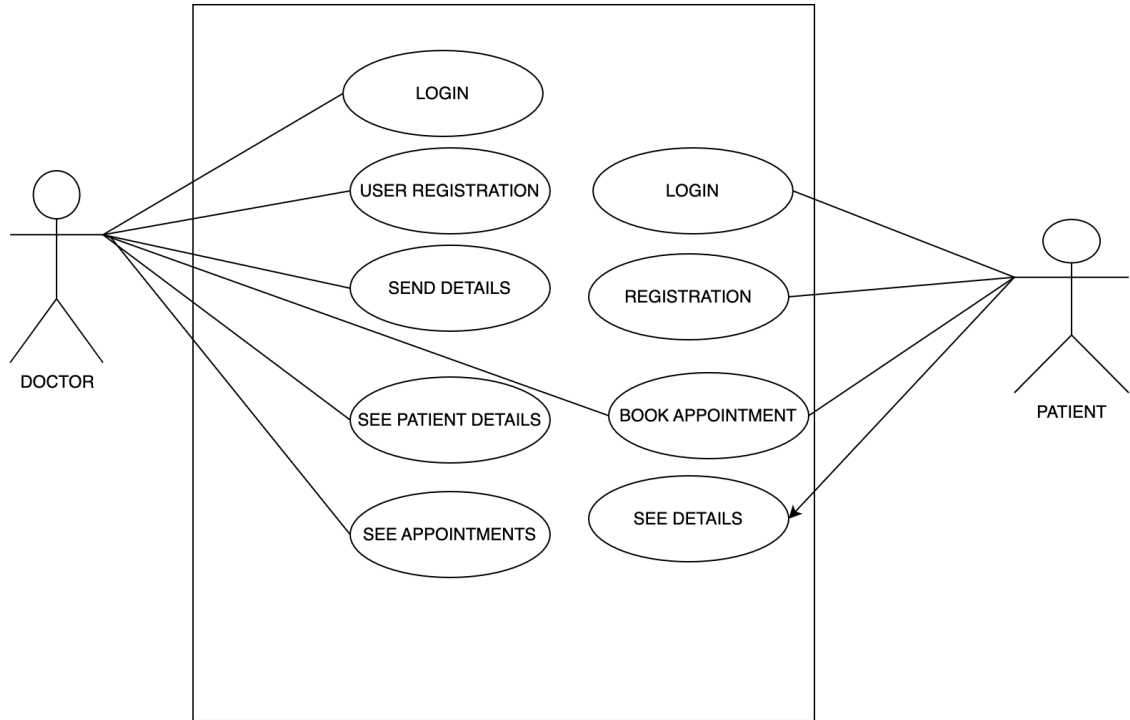
SOFTWARE REQUIREMENTS

- Windows 11
- Python 3.9 or newer
- Flask
- Web Browser
- Necessary Python Modules

5. ACTIVITY DIAGRAM



USE - CASE DIAGRAM

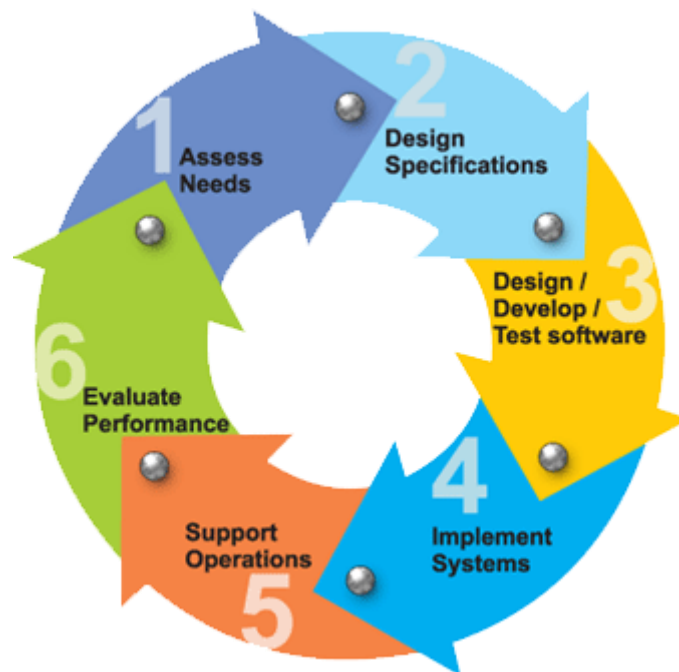


6. SYSTEM ANALYSIS

INTRODUCTION

Software Development Life Cycle:-

There is various software development approaches defined and designed which are used/employed during development process of software, these approaches are also referred as "Software Development Process Models". Each process model follows a particular life cycle in order to ensure success in process of software development.



Requirements:-

Business requirements are gathered in this phase. This phase is the main focus of the project managers and stake holders. Meetings with managers, stake holders and users are held in order to determine the requirements. Who is going to use the system? How will they use the system? What data should be input into the system? What data should be output by the system? These are general questions that get answered during a requirements gathering phase. This produces a nice big list of functionality that the system should provide, which describes functions the system should perform, business logic that processes data, what data

is stored and used by the system, and how the user interface should work. The overall result is the system as a whole and how it performs, not how it is actually going to do it.

Design

The software system design is produced from the results of the requirements phase. Architects have the ball in their court during this phase and this is the phase in which their focus lies. This is where the details on how the system will work is produced. Architecture, including hardware and software, communication, software design (UML is produced here) are all part of the deliverables of a design phase.

Implementation

Code is produced from the deliverables of the design phase during implementation, and this is the longest phase of the software development life cycle. For a developer, this is the main focus of the life cycle because this is where the code is produced. Implementation may overlap with both the design and testing phases. Many tools exist (CASE tools) to actually automate the production of code using information gathered and produced during the design phase.

Testing

During testing, the implementation is tested against the requirements to make sure that the product is actually solving the needs addressed and gathered during the requirements phase. Unit tests and system/acceptance tests are done during this phase. Unit tests act on a specific component of the system, while system tests act on the system as a whole.

So in a nutshell, that is a very basic overview of the general software development life cycle model. Now let's delve into some of the traditional and widely used variations.

7. FEASIBILITY STUDY

INTRODUCTION

Preliminary investigation examines project feasibility, the likelihood the system will be useful to the organization. The main objective of the feasibility study is to test the Technical, Operational and Economical feasibility for adding new modules and debugging old running system. All system is feasible if they are unlimited resources and infinite time. There are aspects in the feasibility study portion of the preliminary investigation:

- Technical Feasibility
- Operational Feasibility
- Economical Feasibility

TECHNICAL FEASIBILITY

The technical issue usually raised during the feasibility stage of the investigation includes the following:

- Does the necessary technology exist to do what is suggested?
- Do the proposed equipment have the technical capacity to hold the data required to use the new system?
- Will the proposed system provide adequate response to inquiries, regardless of the number or location of users?
- Can the system be upgraded if developed?
- Are there technical guarantees of accuracy, reliability, ease of access and data security?

Earlier no system existed to cater to the needs of ‘Secure Infrastructure Implementation System’. The current system developed is technically feasible. It is a web-based user interface for audit workflow at NIC-CSD. Thus, it provides an easy access to the users. The database’s purpose is to create, establish and maintain a workflow among various entities in order to facilitate all concerned users in their various capacities or roles. Permission to the users would be granted based on the roles specified. Therefore, it provides the technical guarantee of accuracy, reliability and security. The software and hard requirements for the development of

this project are not many and are already available in-house at NIC or are available as free as open source. The work for the project is done with the current equipment and existing software technology. Necessary bandwidth exists for providing a fast feedback to the users irrespective of the number of users using the system.

OPERATIONAL FEASIBILITY

Proposed projects are beneficial only if they can be turned out into information system. That will meet the organization's operating requirements. Operational feasibility aspects of the project are to be taken as an important part of the project implementation. Some of the important issues raised are to test the operational feasibility of a project includes the following:

-

- Is there sufficient support for the management from the users?
- Will the system be used and work properly if it is being developed and implemented?
- Will there be any resistance from the user that will undermine the possible application benefits?

This system is targeted to be in accordance with the above-mentioned issues. Beforehand, the management issues and user requirements have been taken into consideration. So, there is no question of resistance from the users that can undermine the possible application benefits.

The well-planned design would ensure the optimal utilization of the computer resources and would help in the improvement of performance status.

Operations Issues	Support Issues
<ul style="list-style-type: none"> • What tools are needed to support operations? • What skills will operators need to be trained in? • What processes need to be created and/or updated? • What documentation does operations need? 	<ul style="list-style-type: none"> • What documentation will users be given? • What training will users be given? • How will change requests be managed?

Very often you will need to improve the existing operations, maintenance, and support infrastructure to support the operation of the new application that you intend to develop. To determine what the impact will be you will need to understand both the current operations and support infrastructure of your organization and the operations and support characteristics of your new application. To operate this application END-TO-END VMS. The user no need to require any technical knowledge that we are used to develop this project is Python Programming. That the application providing rich user interface by user can do the operation in flexible manner.

ECONOMICAL FEASIBILITY

A system can be developed technically and that will be used if installed must still be a good investment for the organization. In the economic feasibility, the development cost in creating the system is evaluated against the ultimate benefit derived from the new systems. Financial benefits must equal or exceed the costs.

The system is economically feasible. It does not require any addition hardware or software. Since the interface for this system is developed using the existing resources and technologies available at NIC, there is nominal expenditure and economic feasibility for certain.

Determining Economic Feasibility:

Assessing the economic feasibility of an implementation by performing a cost/benefit analysis, which as its name suggests compares the full/real costs of the application to its full/real financial benefits. The alternatives should be evaluated on the basis of their contribution to net cash flow, the amount by which the benefits exceed the costs, because the primary objective of all investments is to improve overall organizational performance.

Type	Potential Costs	Potential Benefits
Quantitative	<ul style="list-style-type: none">• Hardware/software upgrades• Fully-burdened cost of labor (salary + benefits)• Support costs for the application• Expected operational costs• Training costs for users to learn the application• Training costs to train developers in new/updated technologies	<ul style="list-style-type: none">• Reduced operating costs• Reduced personnel costs from a reduction in staff• Increased revenue from additional sales of your organizations products/services
Qualitative	<ul style="list-style-type: none">• Increased employee dissatisfaction from fear of change	<ul style="list-style-type: none">• Improved decisions as the result of access to accurate and timely information• Raising of existing, or introduction of a new, barrier to entry within your industry to keep competition out of your market• Positive public perception that your organization is an innovator

8. PYTHON

Python is an interpreted, high-level, general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace. It provides constructs that enable clear programming on both small and large scales. Van Rossum led the language community until stepping down as leader in July 2018. Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural. It also has a comprehensive standard library. The Python interpreter and the extensive standard library are freely available in source or binary form for all major platforms from the Python Web site, <https://www.python.org/>, and may be freely distributed. The same site also contains distributions of and pointers to many free third party Python modules, programs and tools, and additional documentation.

Python is simple to use, but it is a real programming language, offering much more structure and support for large



programs than shell scripts or batch files can offer. On the other hand, Python also offers much more error checking than C, and, being a very-high-level language, it has high-level data types built in, such as flexible arrays and dictionaries. Because of its more general data types Python is applicable to a much larger problem domain than Awk or even Perl, yet many things are at least as easy in Python as in those languages.

Invoking the Interpreter

The Python interpreter is usually installed as `/usr/local/bin/python3.7` on those machines where it is available; putting `/usr/local/bin` in your Unix shell's search path makes it possible to start it by typing the command:

```
python3.7
```

to the shell. [1] Since the choice of the directory where the interpreter lives is an installation option, other places are possible; check with your local Python guru or system administrator. (E.g., `/usr/local/python` is a popular alternative location.)

On Windows machines, the Python installation is usually placed in `C:\Python37`, though you can change this when you're running the installer. To add this directory to your path, you can type the following command into the command prompt in a DOS box:

```
set path=%path%;C:\python37
```

Typing an end-of-file character (Control-D on Unix, Control-Z on Windows) at the primary prompt causes the interpreter to exit with a zero exit status. If that doesn't work, you can exit the interpreter by typing the following command: `quit()`.

The interpreter's line-editing features include interactive editing, history substitution and code completion on systems that support readline. Perhaps the quickest check to see whether command line editing is supported is typing Control-P to the first Python prompt you get. If it beeps, you have command line editing; see Appendix Interactive Input Editing and History Substitution for an introduction to the keys. If nothing appears to happen, or if `^P` is echoed, command line editing isn't available; you'll only be able to use backspace to remove characters from the current line.

Interactive Mode

When commands are read from a tty, the interpreter is said to be in interactive mode. In this mode it prompts for the next command with the primary prompt, usually three greater-than

signs (`>>>`); for continuation lines it prompts with the secondary prompt, by default three dots (`...`). The interpreter prints a welcome message stating its version number and a copyright notice before printing the first prompt:

```
$ python3.7
Python 3.7 (default, Sep 16 2015, 09:25:04)
[GCC 4.8.2] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

Continuation lines are needed when entering a multi-line construct. As an example, take a look at this if statement:

```
>>> the_world_is_flat = True
>>> if the_world_is_flat:
...     print("Be careful not to fall off!")
...
Be careful not to fall off!
```

In the following examples, input and output are distinguished by the presence or absence of prompts (`>>>` and `...`): to repeat the example, you must type everything after the prompt, when the prompt appears; lines that do not begin with a prompt are output from the interpreter. Note that a secondary prompt on a line by itself in an example means you must type a blank line; this is used to end a multi-line command.

9. FLASK



Flask is a lightweight WSGI web application framework. It is designed to make getting started quick and easy, with the ability to scale up to complex applications. It began as a simple wrapper around Werkzeug and Jinja and has become one of the most popular Python web application frameworks.

Flask offers suggestions, but doesn't enforce any dependencies or project layout. It is up to the developer to choose the tools and libraries they want to use. There are many extensions provided by the community that make adding new functionality easy.

```
from flask import Flask, escape, request

app = Flask(__name__)

@app.route('/')

def hello():

    name = request.args.get("name", "World")

    return f'Hello, {escape(name)}!'

$ env FLASK_APP=hello.py flask run

* Serving Flask app "hello"

* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

The code snippet provided demonstrates a basic Flask application with a single route ("/") that accepts a query parameter "name" and responds with a personalized greeting. This example showcases the framework's simplicity and readability. Additionally, the use of the escape function from the Flask module helps prevent potential security issues by escaping HTML characters in the user-supplied data.

Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.

Flask's popularity is further emphasized by its adoption in notable applications such as Pinterest and LinkedIn. Its lightweight nature and flexibility make it well-suited for a wide range of projects, from small prototypes to larger, more complex applications. As a testament to its effectiveness, Flask has gained a substantial following within the Python web development community.

10. HTML5

HTML5 is a markup language used for structuring and presenting content on the World Wide Web. It is the fifth and last major HTML version that is a World Wide Web Consortium (W3C) recommendation. The current specification is known as the HTML Living Standard. It is maintained by a consortium of the major browser vendors (Apple, Google, Mozilla, and Microsoft), the Web Hypertext Application Technology Working Group (WHATWG).



HTML5 was first released in a public-facing form on 22 January 2008, with a major update and "W3C Recommendation" status in October 2014.[2][5] Its goals were to improve the language with support for the latest multimedia and other new features; to keep the language both easily readable by humans and consistently understood by computers and devices such as web browsers, parsers, etc., without XHTML's rigidity; and to remain backward-compatible with older software. HTML5 is intended to subsume not only HTML 4 but also XHTML 1 and DOM Level 2 HTML.

HTML5 includes detailed processing models to encourage more interoperable implementations; it extends, improves, and rationalizes the markup available for documents and introduces markup and application programming interfaces (APIs) for complex web applications. For the same reasons, HTML5 is also a candidate for cross-platform mobile applications because it includes features designed with low-powered devices in mind.

Many new syntactic features are included. To natively include and handle multimedia and graphical content, the new <video>, <audio> and <canvas> elements were added, and support for scalable vector graphics (SVG) content and MathML for mathematical formulas was also added. To enrich the semantic content of documents, new page structure elements such as <main>, <section>, <article>, <header>, <footer>, <aside>, <nav>, and <figure> are added. New attributes were introduced, some elements and attributes were removed, and others such as <a>, <cite>, and <menu> were changed, redefined, or standardized. The APIs and Document Object Model (DOM) are now fundamental parts of the HTML5 specification, and HTML5 also better defines the processing for any invalid documents.

In conclusion, HTML5 represents a significant milestone in web development, addressing the evolving needs of the digital landscape. Its continuous evolution and widespread adoption by major browser vendors demonstrate its pivotal role in shaping the future of web technologies. As the HTML Living Standard, HTML5 stands as a testament to collaborative efforts, ensuring that the language not only adapts to contemporary multimedia demands but also maintains readability for developers and compatibility across diverse devices. With its enriched semantic elements, multimedia handling capabilities, and support for complex web applications, HTML5 remains a cornerstone for building modern and interactive online experiences, emphasizing both innovation and backward compatibility. As technology advances, HTML5's influence continues to shape the way content is structured and presented on the World Wide Web.

11. CSS (CASCADING STYLE SHEET)

CSS is used to control the style of a web document in a simple and easy way. **CSS** is the acronym for "**Cascading Style Sheet**". This tutorial covers both the versions CSS1,CSS2 and CSS3, and gives a complete understanding of CSS, starting from its basics to advanced concepts.

Why to Learn CSS?

Cascading Style Sheets, fondly referred to as **CSS**, is a simple design language intended to simplify the process of making web pages presentable.

CSS is a **MUST** for students and working professionals to become a great Software Engineer specially when they are working in Web Development Domain. I will list down some of the key advantages of learning CSS:

- **Create Stunning Web site** - CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs,variations in display for different devices and screen sizes as well as a variety of other effects.
- **Become a web designer** - If you want to start a carrer as a professional web designer, HTML and CSS designing is a must skill.
- **Control web** - CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the markup languages HTML or XHTML.
- **Learn other languages** - Once you understands the basic of HTML and CSS then other related technologies like javascript, php, or angular are become easier to understand.

Applications of CSS

As mentioned before, CSS is one of the most widely used style language over the web. I'm going to list few of them here:

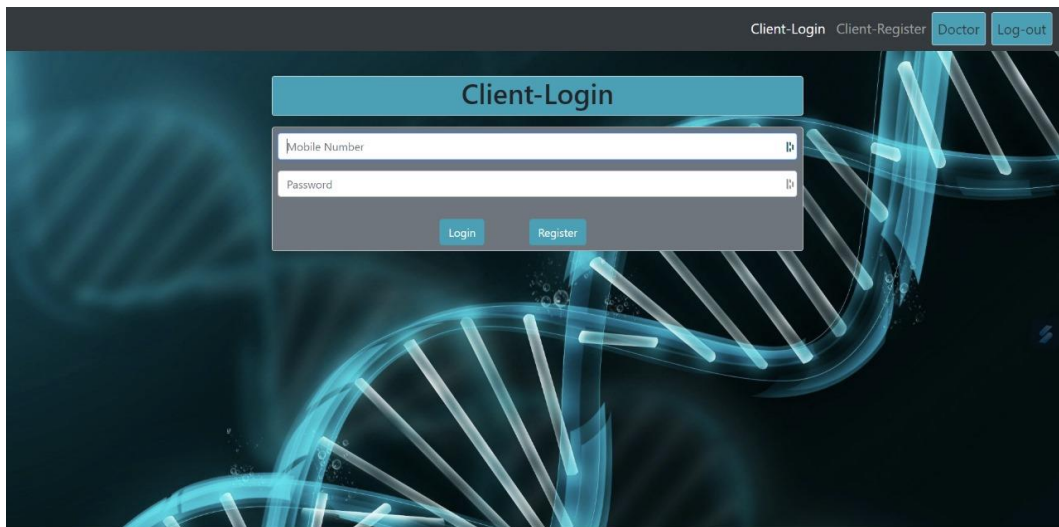
- **CSS saves time** - You can write CSS once and then reuse same sheet in multiple HTML pages. You can define a style for each HTML element and apply it to as many Web pages as you want.
- **Pages load faster** - If you are using CSS, you do not need to write HTML tag attributes every time. Just write one CSS rule of a tag and apply it to all the occurrences of that tag. So less code means faster download times.
- **Easy maintenance** - To make a global change, simply change the style, and all elements in all the web pages will be updated automatically.
- **Superior styles to HTML** - CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.
- **Multiple Device Compatibility** - Style sheets allow content to be optimized for more than one type of device. By using the same HTML document, different versions of a website can be presented for handheld devices such as PDAs and cell phones or for printing.
- **Global web standards** - Now HTML attributes are being deprecated and it is being recommended to use CSS. So its a good idea to start using CSS in all the HTML pages to make them compatible to future browsers.

Prerequisites

You should be familiar with:

- Basic word processing using any text editor.
- How to create directories and files.
- How to navigate through different directories.
- Internet browsing using popular browsers like Internet Explorer or Firefox.
- Developing simple Web Pages using HTML or XHTML.

12. RUN TIME OUTPUT SCREENSHOTS



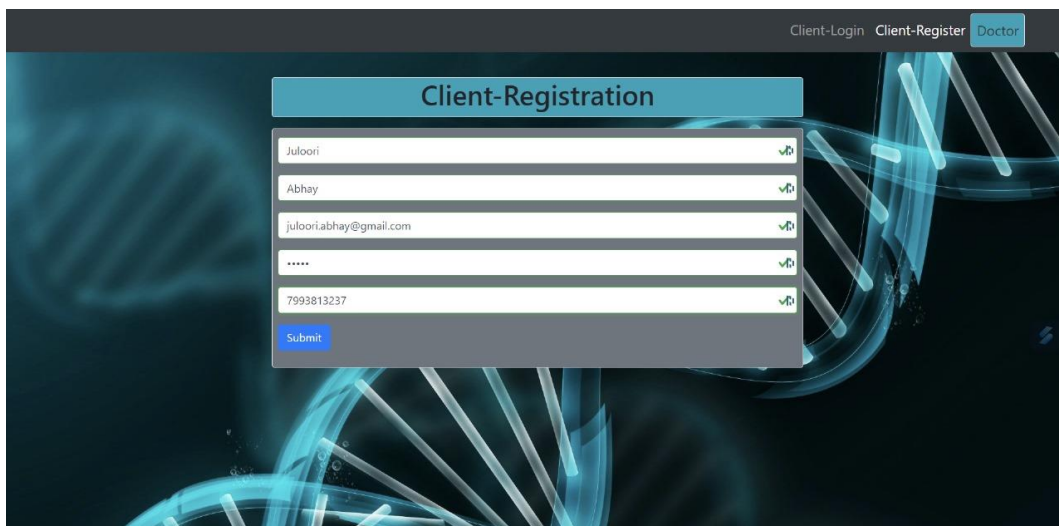
Client-Login Client-Register Doctor Log-out

Client-Login

Mobile Number

Password

Login Register



Client-Login Client-Register Doctor

Client-Registration

Juloori

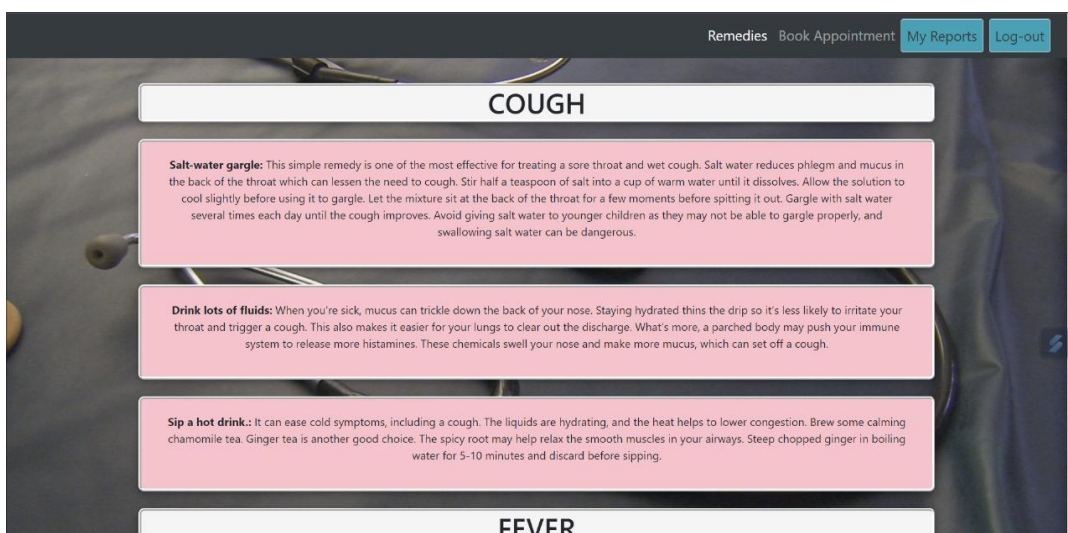
Abhay

juloori.abhay@gmail.com

.....

7993813237

Submit



Remedies Book Appointment My Reports Log-out

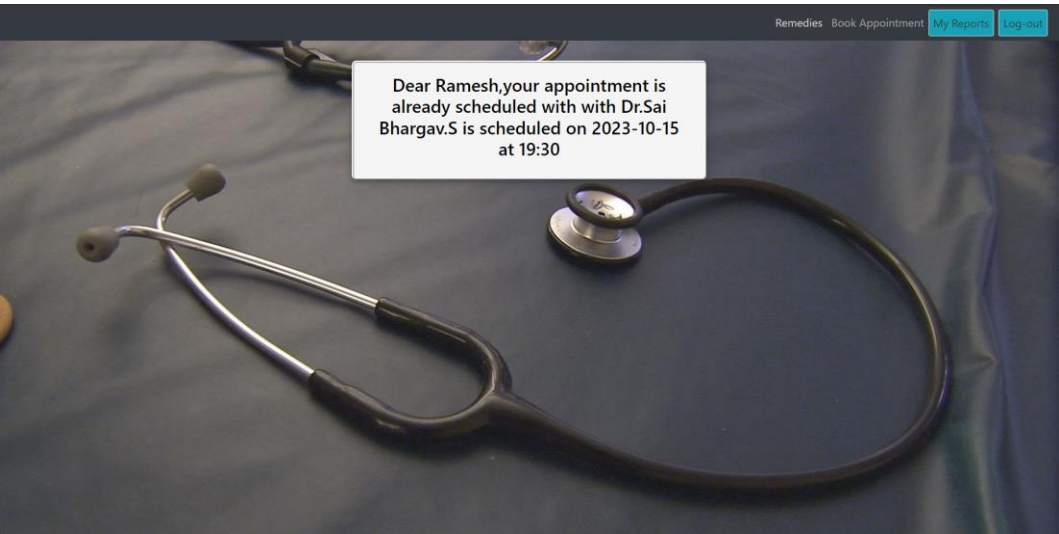
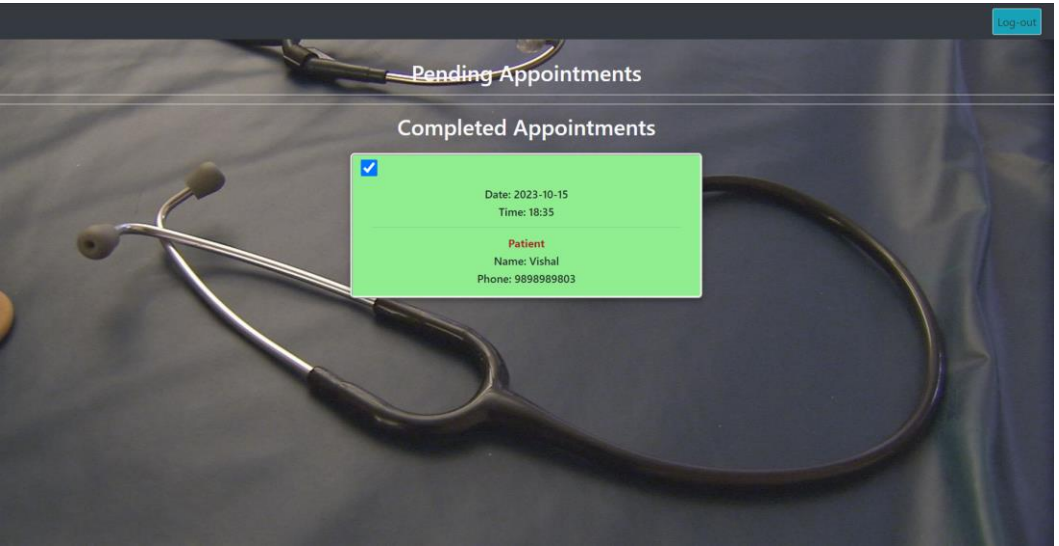
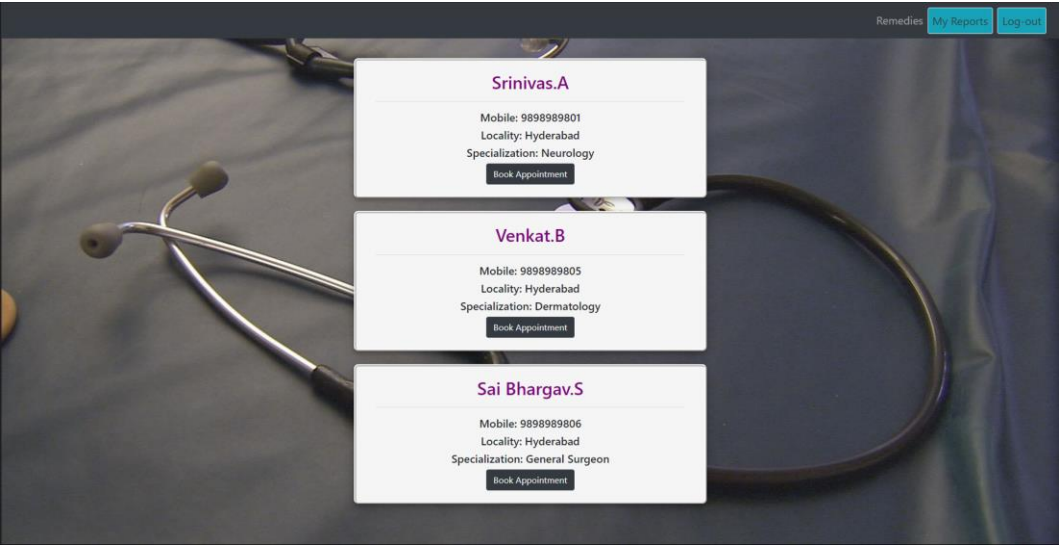
COUGH

Salt-water gargle: This simple remedy is one of the most effective for treating a sore throat and wet cough. Salt water reduces phlegm and mucus in the back of the throat which can lessen the need to cough. Stir half a teaspoon of salt into a cup of warm water until it dissolves. Allow the solution to cool slightly before using it to gargle. Let the mixture sit at the back of the throat for a few moments before spitting it out. Gargle with salt water several times each day until the cough improves. Avoid giving salt water to younger children as they may not be able to gargle properly, and swallowing salt water can be dangerous.

Drink lots of fluids: When you're sick, mucus can trickle down the back of your nose. Staying hydrated thins the drip so it's less likely to irritate your throat and trigger a cough. This also makes it easier for your lungs to clear out the discharge. What's more, a parched body may push your immune system to release more histamines. These chemicals swell your nose and make more mucus, which can set off a cough.

Sip a hot drink: It can ease cold symptoms, including a cough. The liquids are hydrating, and the heat helps to lower congestion. Brew some calming chamomile tea. Ginger tea is another good choice. The spicy root may help relax the smooth muscles in your airways. Steep chopped ginger in boiling water for 5-10 minutes and discard before sipping.

FEVER



13. CONCLUSION

In conclusion, the development of the Flask-based Doctor-Patient Logger App represents a significant step towards improving the healthcare system's efficiency and accessibility. This project was designed to address the critical need for an organized and secure platform for doctors and patients to communicate, schedule appointments, and keep track of medical records.

Throughout this project, we successfully implemented several crucial features, including user authentication, appointment scheduling, and record-keeping. The app offers a user-friendly interface that ensures both doctors and patients can easily navigate and use its features.

One of the most notable achievements of this project is the enhanced communication and convenience it provides to both doctors and patients. Doctors can manage their appointments, access patient information, and keep records up-to-date efficiently. Patients, on the other hand, can easily schedule appointments, view their medical history, and communicate with their healthcare providers without the need for physical visits.

The security of sensitive medical data is of utmost importance, and we have integrated security measures to protect patient information. We have employed encryption and authentication mechanisms to ensure data privacy and maintain compliance with healthcare data security standards.

While this project represents a significant milestone, there is always room for improvement. Future work could focus on enhancing the user experience, adding more features such as telemedicine capabilities, and expanding the app's scalability to accommodate more users and medical practitioners.

In conclusion, the Flask-based Doctor-Patient Logger App holds the potential to revolutionize the healthcare sector by streamlining the communication process between doctors and patients, reducing administrative burdens, and ultimately improving patient care. With continued development and integration into healthcare systems, this application could become an invaluable tool for both medical professionals and patients alike.

APPENDICES

- **Code Snippets**

In this section, we provide essential code snippets from our Flask-based Doctor-Patient Logger App to give readers a glimpse of our implementation. Please note that this is not an exhaustive code listing, but a selection of key portions of our codebase.

```
# User authentication with Flask-Login
from flask_login import LoginManager, UserMixin, login_user, login_required,

# Route for user login
@app.route('/login', methods=['GET', 'POST'])
def login():
    if request.method == 'POST':
        user = User.query.filter_by(email=request.form['email']).first()
        if user is not None and user.check_password(request.form['password']):
            login_user(user)
            flash('Logged in successfully.')
            return redirect(url_for('dashboard'))
        flash('Invalid email or password.')
    return render_template('login.html')
```

- **Database Schema**

This section includes the database schema used in the project, outlining the structure of our database tables and their relationships. We have used SQLAlchemy as our ORM (Object-Relational Mapping) tool for this project.

```
# Database schema using SQLAlchemy
from flask_sqlalchemy import SQLAlchemy

db = SQLAlchemy()

class User(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    username = db.Column(db.String(80), unique=True, nullable=False)
    email = db.Column(db.String(120), unique=True, nullable=False)
    # ... other fields ...

class Appointment(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    doctor_id = db.Column(db.Integer, db.ForeignKey('user.id'))
    patient_id = db.Column(db.Integer, db.ForeignKey('user.id'))
    appointment_date = db.Column(db.DateTime)
    # ... other fields ...
```

- **Installation and Usage Guide**

To set up and run the Flask-based Doctor-Patient Logger App, follow these steps:

Clone the project repository from GitHub: [GitHub Repository URL]

Navigate to the project directory.

Create a virtual environment and activate it.

Install the required dependencies using `pip install -r requirements.txt`.

Configure the application, including database settings and secret keys, in the `config.py` file.

Initialize the database using Flask-Migrate: `flask db init`, `flask db migrate`, `flask db upgrade`.

Run the Flask application: `flask run`.

- **Data Security and Privacy Considerations**

The Flask-based Doctor-Patient Logger App places a strong emphasis on data security and privacy. We use encryption to protect sensitive data, and user authentication is implemented for secure access. Additionally, the app complies with healthcare data security standards, ensuring the confidentiality of patient information.

- **Future Work and Enhancements**

The project offers promising features and functionalities for the healthcare sector. Future enhancements and development opportunities include:

Implementation of telemedicine capabilities to enable virtual consultations.

Integration with external health record systems for data synchronization.

Enhanced data visualization for medical professionals.

Expansion of the app's scalability to accommodate a larger user base.

REFERENCES

- [1] Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for Smart Cities," *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 22-32, 2014.
- [2] M. Hassanaliyagh et al., "Health Monitoring and Management Using Internet-of-Things (IoT) Sensing with Cloud-Based Processing: Opportunities and Challenges," 2015 *IEEE International Conference on Services Computing*, New York, NY, pp. 285-292, 2015.
- [3] V. Vipplapalli and S. Ananthula, "Internet of things (IoT) based smart health care system," 2016 *International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, Paralakhemundi, pp. 1229-1233, 2016.
- [4] J. Saha et al., "Advanced IOT based combined remote health monitoring, home automation and alarm system," 2018 *IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*, Las Vegas, NV, pp. 602-606, 2018.
- [5] N. Gupta, H. Saeed, S. Jha, M. Chahande and S. Pandey, "IOT based health monitoring systems," 2017 *International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, Coimbatore, pp. 1-6, 2017.
- [6] R. Jayswal, R. Gupta and K. K. Gupta, "Patient health monitoring system based on Internet of Things," 2017 *Fourth International Conference on Image Information Processing (ICIIP)*, Shimla, pp. 1-4, 2017.
- [7] G. V. Kumar, A. Bharadwaja and N. N. Sai, "Temperature and heart beat monitoring system using IOT," 2017 *International Conference on Trends in Electronics and Informatics (ICEI)*, Tirunelveli, pp. 692-695, 2017.
- [8] M. Pereira and K. K. Nagapriya, "A novel IoT based health monitoring system using LPC2129," 2017 *2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, Bangalore, pp. 564-568, 2017.