

Doctor Appointment System

Giselle Fernandes - 2308
Niyati Madgaokar - 2312
Swizel Monteiro - 2314

INTRODUCTION

In today's fast-paced world, access to healthcare services is paramount, yet often hindered by logistical challenges such as long wait times, scheduling conflicts, and administrative inefficiencies. To address these issues and enhance the patient experience, the Design and Development of an Online Doctor Appointment System becomes imperative. This project aims to revolutionize the way patients interact with healthcare providers by offering a comprehensive and user-friendly platform for scheduling appointments, managing medical records, and facilitating communication between patients and healthcare professionals.

OBJECTIVES

Our project aims to streamline the management of doctor, appointment, patient, booking, and doctor schedules through an administrative-end application. Access is restricted to administrators to ensure data security. The purpose is to automate manual processes related to managing doctors, appointments, and patients, tracking details such as doctor schedules, patient information, and bookings. Future functionalities include enhanced security, search capabilities, comprehensive monitoring of transactions, and integration of all doctor schedule records.

Key features we aim to provide:

- Secure Authentication: Robust authentication mechanisms, including encrypted passwords and multi-factor authentication, ensure secure access for administrators, doctors, and users, enhancing data security and privacy.
- Intuitive Search Functionality: Advanced search options allow users to find doctors based on various criteria such as location, specialization, and availability. This facilitates easy appointment scheduling and enhances user convenience.
- Seamless Appointment Management: Doctors can efficiently manage their schedules by accepting or canceling appointments. Real-time updates reflect in the system, minimizing scheduling conflicts and optimizing workflow efficiency.
- Privacy-Preserving Design: Patient details are securely managed, with only doctors and authorized personnel having access to relevant information. This maintains patient confidentiality and ensures compliance with privacy regulations.
- Mobile-Friendly Interface: A responsive design ensures optimal user experience across devices, enabling convenient access to appointment booking functionality on smartphones and tablets. This improves accessibility and engagement, allowing users to schedule appointments on-the-go.

MODULES

ADMIN:

Dashboard:

- Provides an overview of total registered doctors and available specializations.
- Specializations:
- Allows admin to manage specializations by adding, updating, or deleting them as needed.

Doctor List:

- Displays a list of doctors along with their respective appointments for easy monitoring.

Search Doctor:

- Enables admin to search for a specific doctor based on their mobile number, facilitating quick access to relevant information.

Doctor Reg Report:

- Generates reports of doctor registrations within a specified date range, aiding in administrative tasks and analysis.

Website Page:

- Allows admin to manage content on website pages such as About Us and Contact Us, ensuring up-to-date information is presented to users.

DOCTOR:

Dashboard:

- Provides a summary of new appointments, approved appointments, canceled appointments, and completed appointments, allowing doctors to track their workload effectively.

Appointment:

- Enables doctors to view appointment details and change appointment status based on the current situation, ensuring smooth management of patient appointments.

Patient List:

- Allows doctors to access their finalized appointments, prescribe medications, and recommend tests, streamlining patient management processes.

Appointment Completed:

- Provides a view of completed appointments for reference and follow-up purposes, aiding in patient care continuity.

Reports:

- Allows doctors to view appointment details within a specified period, facilitating analysis and performance evaluation.

Search:

- Enables doctors to search for appointments using user appointment numbers or names, improving accessibility to patient information.

Profile Management:

- Allows doctors to update their profiles, change passwords, and recover passwords as needed, ensuring account security and personalized user experience.

USER:(Working on it)

Login:

- Users can securely log in to their accounts using their credentials, ensuring access to personalized features and information.

Dashboard:

- Users have access to a personalized dashboard where they can manage appointments for themselves and their family members. They can add family members, view their profiles, and make appointments on their behalf.

Family Members Management:

- Users can add, edit, or remove family members from their dashboard, facilitating centralized management of healthcare needs for the entire family.

Appointment Booking:

- Users can easily schedule appointments for themselves or their family members based on location and doctor specialization. They can browse available doctors, select suitable appointment slots, and confirm bookings seamlessly.

Appointment Management:

- Users have the ability to view, update, or cancel appointments for themselves and their family members. This feature enables efficient scheduling and coordination of healthcare services.

Medical Records Management:

- Users can upload and maintain comprehensive medical histories, prescriptions, and other relevant documents for each family member. This centralized

repository ensures easy access to medical information during appointments and consultations.

Patient Profile:

- Users can create and manage individual patient profiles for each family member, including demographic information, medical history, allergies, and medications. This facilitates personalized care and improves communication with healthcare providers.

Prescription Upload:

- Users can upload prescriptions and medical reports directly to the platform, ensuring seamless communication with healthcare providers and maintaining accurate medical records.

Appointment History:

- Users have access to a detailed appointment history for themselves and their family members, allowing them to track past appointments, follow-up visits, and treatment plans.

Notifications:

- Users receive timely notifications for upcoming appointments, appointment confirmations, and any updates or changes to their appointments, ensuring they stay informed about their healthcare schedule.

FUNCTIONALITIES

DASHBOARD

This gives a brief overview of the healthcare facility, including its mission, vision, and values. It includes the along with some key statistics describing the facility. The navbar includes User Registration and Login , and Appointment

Good Health Is The Root Of All Happiness

123

Expert Doctors

1234

Medical Staff

12345

Total Patients

Search Appointment

REGISTRATION

Allows admin ,doctors , users to register accounts with additional information such as specialization and profile picture.



Sign Up
Access to our dashboard

Profile Pic
 Choose file No file chosen

First Name

Last Name

Email

Username

Mobile Number

Specialization

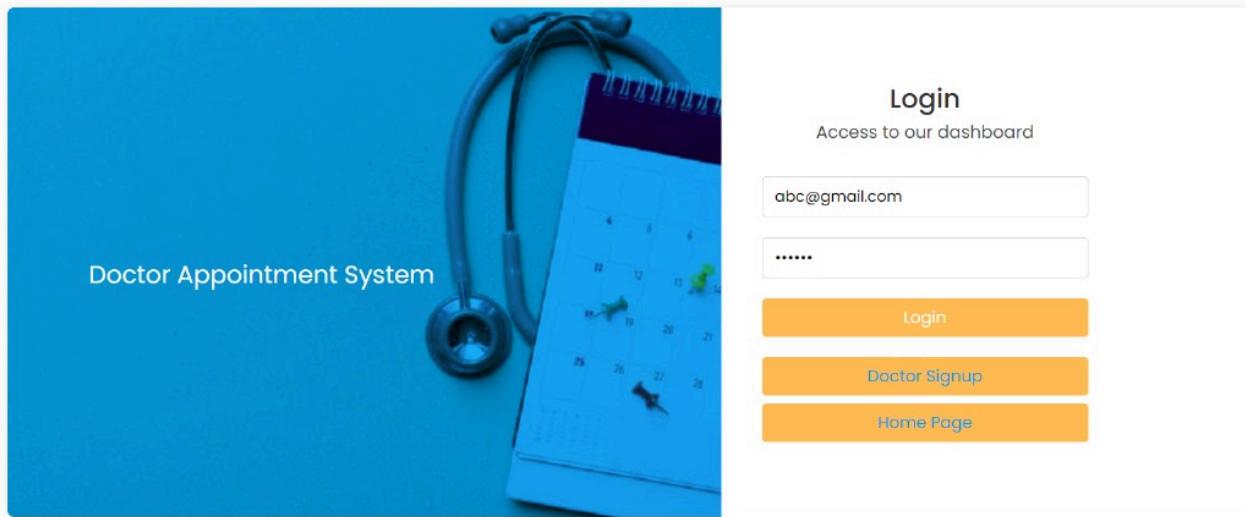
Password

Register

Do you have an account ?

LOGIN

Allows admin ,doctors , users to register accounts with additional information such as specialization and profile picture.



PROFILE UPDATE

This screenshot shows the "Profile Details" page within the Doctor Appointment System. The left sidebar has a logo and the acronym "DAS". The main menu includes "Main Menu", "Dashboard", "Specialization", "Doctor List", "Search Doctor", "Doctor Reg Report", and "Website Page". The central content area is titled "Profile Details" and shows the current profile information: "Profile Pic" (choose file, no file chosen), "First Name" (Admin), "Last Name" (Sample), "Email" (admin@gmail.com), and "Username" (admin). A "Update" button is at the bottom. The top right corner shows a user profile icon. At the bottom center, it says "Doctor Appointment System.".

ADMIN FUNCTIONALITIES

ADMIN DASHBOARD

The screenshot shows the Admin Dashboard of the Doctor Appointment System. On the left, there's a sidebar with a logo and the acronym 'DAS'. The main menu includes options like Main Menu, Dashboard, Appointment, Patient List, Appointment Completed, B/D Appointment Report, and Search Appointment. The central area displays a welcome message 'Welcome Kiran Yadu!!!!' followed by a dashboard summary: '9 Total Doctor' and '13 Total Specialization'. A footer at the bottom right says 'Doctor Appointment System.'

CHANGE PASSWORD

The screenshot shows the 'Change Password' page within the Admin Dashboard. The sidebar has a similar structure to the dashboard, with options like Main Menu, Dashboard, Specialization, Doctor List, Search Doctor, Doctor Reg Report, and Website Page. The main content area is titled 'Change Password' and shows a breadcrumb navigation: 'Dashboard / Change Password'. It features two input fields: 'Current Password' and 'New Password', each with a corresponding text input box. Below these fields is a dark blue 'Change' button. A footer at the bottom right says 'Doctor Appointment System.'

ADD SPECIALIZATION



DAS

Main Menu

- Dashboard
- Specialization
- Doctor List
- Search Doctor
- Doctor Reg Report
- Website Page

Add Specialization

Dashboard / Add Specialization

Add Specialization

Specialization Name

Add

Doctor Appointment System.

MANAGE SPECIALIZATION



DAS

Main Menu

- Dashboard
- Specialization
- Doctor List
- Search Doctor
- Doctor Reg Report
- Website Page

Manage Specialization

Dashboard / Manage Specialization

S.No	Specialization	Creation Date	Action
1	Orthopedics	Feb. 26, 2024, 6:24 a.m.	 
2	Internal Medicine	Feb. 26, 2024, 6:24 a.m.	 
3	Obstetrics and Gynecology	Feb. 26, 2024, 6:24 a.m.	 
4	Dermatology	Feb. 26, 2024, 6:24 a.m.	 
5	Pediatrics	Feb. 26, 2024, 6:24 a.m.	 
6	Radiology	Feb. 26, 2024, 6:25 a.m.	 

UPDATE SPECIALIZATION

 DAS

Main Menu

- Dashboard
- Specialization
- Doctor List
- Search Doctor
- Doctor Reg Report
- Website Page

Doctor Appointment System

Update Specialization

Dashboard / Update Specialization

Update Specialization

Specialization Name

Orthopedics

Update

Doctor Appointment System.

REGISTERED DOCTOR LIST

 DAS

Main Menu

- Dashboard
- Specialization
- Doctor List
- Search Doctor
- Doctor Reg Report
- Website Page

Doctor Appointment System

Registered Doctor List

Dashboard / Doctor List

S.No	Fullname	Mobile Number	Email	Registration Date	Action
1	Garima Singh	7894561236	garima@gmail.com	Feb. 26, 2024, 6:29 a.m.	View Appointment List
2	Abir Kumar	7897979878	abir@gmail.com	Feb. 26, 2024, 6:30 a.m.	View Appointment List
3	Soumya Yadav	6464646445	som@gmail.com	Feb. 26, 2024, 6:30 a.m.	View Appointment List
4	Harishankar Singh	4464564654	har@gmail.com	Feb. 26, 2024, 6:31 a.m.	View Appointment List
5	Renu Gupta	6545646546	renu@gmail.com	Feb. 26, 2024, 6:32 a.m.	View Appointment List
6	Anuj Kumar	1414141425	ak@gmail.com	March 6, 2024, 4:56 p.m.	View Appointment List
7	John Doe	1231231231	johndoe12@gmail.com	March 6, 2024, 5:22 p.m.	View Appointment List

Doctor Appointment System.

VIEW DOCTOR DETAILS

 DAS

Main Menu

- Dashboard
- Specialization
- Doctor List
- Search Doctor
- Doctor Reg Report
- Website Page

Doctor Appointment System

Doctor Details

Dashboard / Doctor Details

Name of Doctor	Garima Singh	Contact Number	7894561236
Email Address	garima@gmail.com	Doctor Specialization	Obstetrics and Gynecology
Profile Pic		Date of Registrations	Feb. 26, 2024, 6:29 a.m.

[Back](#)

Doctor Appointment System.

DOCTOR APPOINTMENT LIST

 DAS

Main Menu

- Dashboard
- Specialization
- Doctor List
- Search Doctor
- Doctor Reg Report
- Website Page

Doctor Appointment System

Appointments

Dashboard / View Appointment

Appointment Number	Patient Name	Date of Appointment	Time of Appointment	Creation Date	Status
925833328	Harish Kumar	2024-02-26	11:44	Feb. 26, 2024, 6:44 a.m.	Completed
566162414	Rakesh Sharma	2024-02-28	14:49	Feb. 26, 2024, 6:45 a.m.	Not Updated Yet

[Back](#)

Doctor Appointment System.

SEARCH DOCTOR

The screenshot shows the 'Search Doctor' page of the Doctor Appointment System. The left sidebar contains a logo and links for Main Menu, Dashboard, Specialization, Doctor List, Search Doctor, Doctor Reg Report, and Website Page. The main content area has a header 'Doctor Appointment System' and a sub-header 'Search Doctor'. It shows a search bar with placeholder 'Search(By Mobilenumber/Name)' and a button 'Search'. Below is a table with two rows of data:

S.No	Fullname	Mobile Number	Email	Registration Date	Action
1	Garima Singh	7894561236	garima@gmail.com	Feb. 26, 2024, 6:29 a.m.	View Appointment List
4	Harishankar Singh	4464564654	har@gmail.com	Feb. 26, 2024, 6:31 a.m.	View Appointment List

At the bottom, a footer bar says 'Doctor Appointment System.'

WEBPAGE DETAILS

The screenshot shows the 'Update Website Details' page of the Doctor Appointment System. The left sidebar is identical to the previous screenshot. The main content area has a header 'Doctor Appointment System' and a sub-header 'Update Website Details'. It shows a form with fields for Page Title (Webpage Details), Address (H-890, Ajanta Apartment), About Us (a text area containing a paragraph about medical specialties), Email (support@gmail.com), and Mobile Number (1234567890). A large orange 'Update' button is at the bottom.

At the bottom, a footer bar says 'Doctor Appointment System.'

DOCTOR FUNCTIONALITIES

DOCTOR DASHBOARD

 DAS

Main Menu

- Dashboard
- Appointment
- Patient List
- Appointment Completed
- B/D Appointment Report
- Search Appointment

Doctor Appointment System

Welcome Doctor Giselle Fernandes!

Dashboard

 5 All Appointment

 1 New Appointment

 1 Approved Appointment

 1 Cancelled Appointment

 2 Completed Appointment

Doctor Appointment System.

APPOINTMENT LIST

 DAS

Main Menu

- Dashboard
- Appointment
- Patient List
- Appointment Completed
- B/D Appointment Report
- Search Appointment

Doctor Appointment System

Appointments

Dashboard / View Appointment

Appointment Number	Patient Name	Date of Appointment	Time of Appointment	Creation Date	Status	Action
100989384	Gis	2024-05-03	16:15	April 30, 2024, 6:41 a.m.	Completed	<button>View</button>
954999688	niyati	2024-05-04	15:20	April 30, 2024, 6:47 a.m.	Completed	<button>View</button>
666377461	Swizel	2024-05-06	12:10	May 2, 2024, 5:38 a.m.	Approved	<button>View</button>
906127018	Daniel	2024-05-07	15:15	May 2, 2024, 5:40 a.m.	Cancelled	<button>View</button>
792059064	Sophia	2024-05-08	16:15	May 2, 2024, 5:40 a.m.	Not Updated Yet	<button>View</button>

Page 1 of 1.

APPOINTMENT DETAILS

Doctor Appointment System

Appointments

Dashboard / View Appointment

Appointment Number: 100989384

Patient Name	Gis	Patient Contact Number	1234567890
Patient Email	abc@gmail.com	Date of Appointment	2024-05-03
Date of Time	16:15	Message	Stomach Pain
Doctor Remark	OK	Status	Completed
Prescribed Medicine	ygewfgi		
Recommended Test	fjaksfbka		

APPROVE/ CANCEL APPOINTMENTS

Doctor Appointment System

Appointments

Dashboard / View Appointment

Appointment Number: 7920

Take Action

Remark :

Status :

Patient Name	Gis	Patient Contact Number	1234567890
Patient Email	abc@gmail.com	Date of Appointment	2024-05-08
Date of Time	16:15	Message	Cough and Cold
Doctor Remark	Not Updated Yet		
Prescribed Medicine			
Recommended Test			

SEARCHING AN APPOINTMENT

Doctor Appointment S/Y

HOME ADMIN DOCTOR REG DOCTOR LOGIN SEARCH APPOINTMENT

Search Appointment

Search(By Fullname/Appointment Number)

Search against Gis

Appointment Number	Patient Name	Date of Appointment	Time of Appointment	Creation Date	Status	Action
167032570	Gis	2024-05-02	10:20	April 30, 2024, 6:32 a.m.	Not Updated Yet	<input type="button" value="View"/>
100989384	Gis	2024-05-03	16:15	April 30, 2024, 6:41 a.m.	Completed	<input type="button" value="View"/>

VIEW NEW APPOINTMENT DETAILS

The screenshot shows a web-based Doctor Appointment System interface. On the left is a sidebar with a logo and navigation links: Main Menu, Dashboard, Appointment (with a dropdown arrow), Patient List, Appointment Completed, B/D Appointment Report, and Search Appointment. The main content area has a header "Doctor Appointment System" and a sub-header "Appointments". Below that is a breadcrumb "Dashboard / View Appointment". A table displays appointment details for "Appointment Number: 566162414":

Patient Name	Rakesh Sharma	Patient Contact Number	5454554646
Patient Email	rakesh@gmail.com <th>Date of Appointment</th> <td>2024-02-28</td>	Date of Appointment	2024-02-28
Date of Time	14:49	Message	sample
Doctor Remark	Not Updated Yet	Status	Not Updated Yet
Prescribed Medicine	Not Prescribed Yet		
Recommended Test	Not Recommended Yet		

A yellow "Take Action" button is at the bottom right of the table.

Doctor Appointment System.

APPROVED APPOINTMENTS

The screenshot shows the same web-based Doctor Appointment System interface. The sidebar and header are identical to the previous screenshot. The main content area shows a table of "Approved" appointments:

Appointment Number	Patient Name	Date of Appointment	Time of Appointment	Creation Date	Status	Action
566162414	Rakesh Sharma	2024-02-28	14:49	Feb. 26, 2024, 6:45 a.m.	Approved	<button>View</button>
if .						

A horizontal scrollbar is visible at the bottom of the table.

Doctor Appointment System.

VIEW APPROVED APPOINTMENTS

The screenshot shows the 'Doctor Appointment System' interface. On the left, a sidebar menu includes 'Main Menu', 'Dashboard', 'Appointment' (selected), 'Patient List', 'Appointment Completed', 'B/D Appointment Report', and 'Search Appointment'. The main content area is titled 'Appointments' and shows a detailed view of a recent appointment. The appointment number is 566162414. The patient's name is Rakesh Sharma, and the contact number is 5454554646. The email is rakesh@gmail.com, and the date of appointment is 2024-02-28. The time is 14:49, and the message is 'sample'. The doctor's remark is 'Approved', and the status is 'Approved'. The prescribed medicine is 'Not Prescribed Yet', and the recommended test is 'Not Recommended Yet'.

PATIENT LIST

The screenshot shows the 'Doctor Appointment System' interface. The sidebar menu is identical to the previous screenshot. The main content area is titled 'Appointments' and shows a list of appointments. The first appointment listed is for Rakesh Sharma, with the appointment number 566162414, creation date Feb. 26, 2024, 6:45 a.m., and status Approved. A 'View' button is next to the appointment details. Below the table, there is a pagination bar labeled 'Page of .'. At the bottom of the page, a footer message reads 'Doctor Appointment System.'

VIEW PATIENT DETAILS

 DAS

Main Menu

- Dashboard
- Appointment >
- Patient List
- Appointment Completed
- B/D Appointment Report
- Search Appointment

Doctor Appointment System

Appointments Lists

Dashboard / View Appointment

Appointment Number: 566162414		Date: 2024-02-28	
Patient Name	Rakesh Sharma	Patient Contact Number	5454554646
Patient Email	rakesh@gmail.com	Date of Appointment	2024-02-28
Date of Time	14:49	Message	sample
Doctor Remark	Approved	Status	Approved
Prescribed Medicine	Not Prescribed Yet		
Recommended Test	Not Recommended Yet		

[Take Action](#)

Doctor Appointment System.

APPOINTMENT COMPLETED

 DAS

Main Menu

- Dashboard
- Appointment >
- Patient List
- Appointment Completed
- B/D Appointment Report
- Search Appointment

Doctor Appointment System

Appointments

Dashboard / View Appointment

Appointment Number	Patient Name	Date of Appointment	Time of Appointment	Creation Date	Status	Action
1833328	Harish Kumar	2024-02-26	11:44	Feb. 26, 2024, 6:44 a.m.	Completed	View

of .

[View](#)

Doctor Appointment System.

APPOINTMENT DETAILS



DAS

Main Menu

- Dashboard
- Appointment >
- Patient List
- Appointment Completed
- B/D Appointment Report
- Search Appointment

Doctor Appointment System

Appointments Lists

Dashboard / View Appointment

Appointment Number: 925833328		Date: 2024-02-26	
Patient Name	Harish Kumar	Patient Contact Number	8979798797
Patient Email	harish@gmail.com	Date of Appointment	2024-02-26
Date of Time	11:44	Message	Pain in stomach since 1 week
Doctor Remark	Approved	Status	Completed
Prescribed Medicine	1. Blargin 2. Zincovit 3. Digene		
Recommended Test	No Test Required		

Doctor Appointment System.

USER FUNCTIONALITIES (still working on it)

📍 H-890, Ajanta Apartment 📩 support@gmail.com

📞 1234567890

[f](#) [t](#) [in](#) [@](#)

Doctor Appointment S/Y

HOME ADMIN DOCTOR REG DOCTOR LOGIN SEARCH APPOINTMENT

Appointment ➔

Appointment

HOME / PAGES / APPOINTMENT

Appointment

Make An Appointment To Visit Our Doctor

Tempor erat elitr rebum at clita. Diam dolor diam ipsum sit. Aliqu diam amet diam et eos. Clita erat ipsum et lorem et sit, sed stet lorem sit clita duo justo magna dolore erat amet

Call Us Now

+1234567890

Mail Us Now

support@gmail.com

Address

H-890, Ajanta Apartment

Your Name

Your Email

Your Mobile

Choose Doctor

mm/dd/yyyy



--:--



Describe your problem

Book Appointment

Address

📍 H-890, Ajanta Apartment
📞 1234567890
✉️ support@gmail.com

[Twitter](#) [Facebook](#) [YouTube](#) [LinkedIn](#)

About Us

There are around 60 medical specialties and 30 subspecialties that make it harder for doctors to choose their specialization. Based on interest, doctors can choose their medical specialties. However, there are aspects like demand, scope, salary, and availability of the PG medical seat. In such a situation, decision-making would be a tougher job. However, your decision is crucial to can set up a rewarding or dismaying career. Thus, be careful while choosing your medical specialties. The medical courses listed below are in high demand and have a greater scope in India.

© Doctor Appointment System, All Right Reserved.

Doctor Appointment System

MAKING AN APPOINTMENT

Appointment

Make An Appointment To Visit Our Doctor

Tempor erat elitr rebum at clita. Diam dolor diam ipsum sit. Aliqu diam amet diam et eos. Clita erat ipsum et lorem et sit, sed stet lorem sit clita duo justo magna dolore erat amet



Call Us Now

+1234567890



Mail Us Now

support@gmail.com

Your Name

Your Email

Your Mobile

Choose Doctor

dd-mm-yyyy



--:--

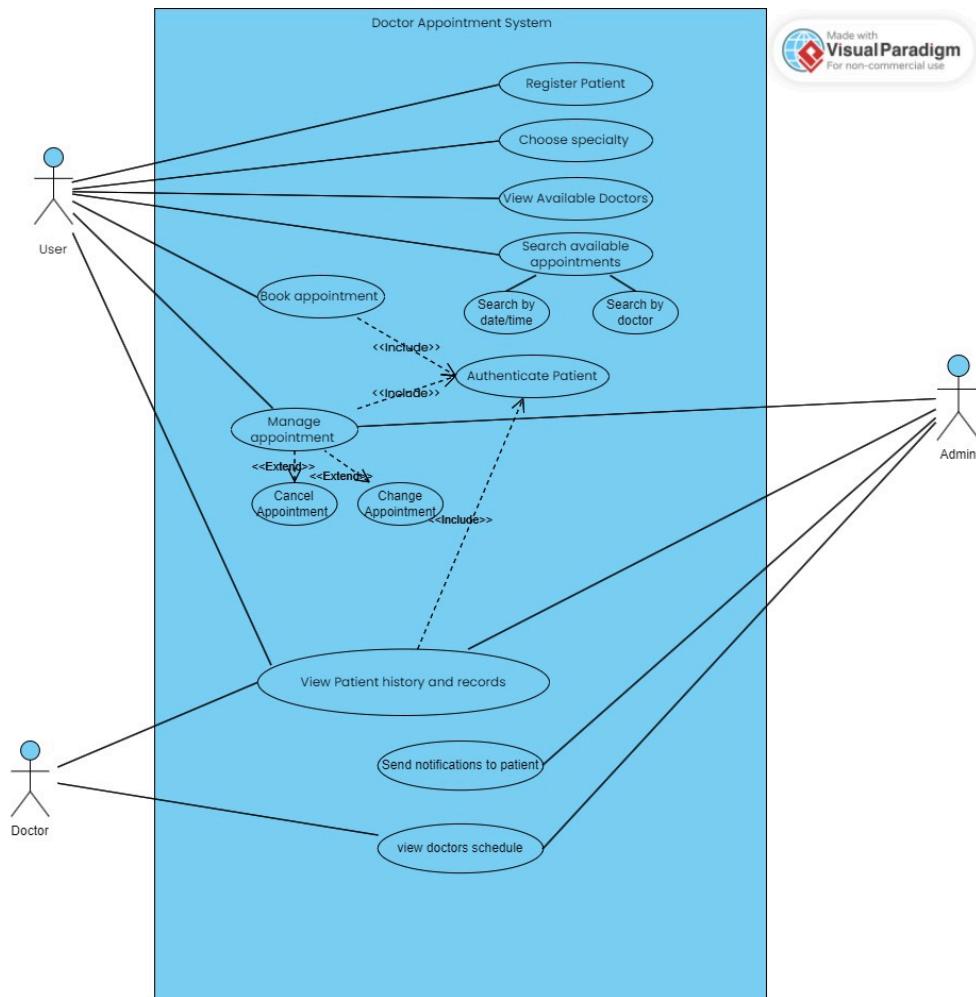


Describe your problem

Book Appointment

USE CASE DIAGRAM

A Use Case Diagram for a system like the one described could depict interactions between actors (users) and the system. Here's a simplified representation:



SOFTWARE ENGINEERING TECHNOLOGIES USED

Web Server	APACHE
Server side language	Python
Database Server	MYSQL
Web Browser	Google Chrome or any compatible browser
Operating System	Windows or any equivalent OS
Scripting Language	Java Script
IDE	PyCharm, VS Code
Source Control	GITHUB

Project management	Jira
Code analysis	Sonar Lint
Framework	Django

DJANGO

In Doctor Appointment Management System we use Python using Django framework and MySQL database. This project has three modules i.e., admin, doctor and user.

In developing our doctor appointment system, we utilized Django, a high-level Python web framework renowned for its efficiency and scalability. At the core of Django's architecture lies the MVT (Model-View-Template) pattern, a variation of the classic MVC (Model-View-Controller) design.

Model: In Django's MVT framework, the Model represents the data structure, essentially the database schema. For our appointment system, models were crafted to define entities like patients, doctors, appointments, and other relevant information. Django's ORM (Object-Relational Mapping) simplifies database interactions, allowing seamless integration with various database systems.

View: Views in Django handle the business logic, processing user requests, and generating responses. They bridge the gap between models and templates, orchestrating data retrieval, manipulation, and rendering. Within our appointment system, views manage tasks such as scheduling appointments, fetching patient records, and handling authentication.

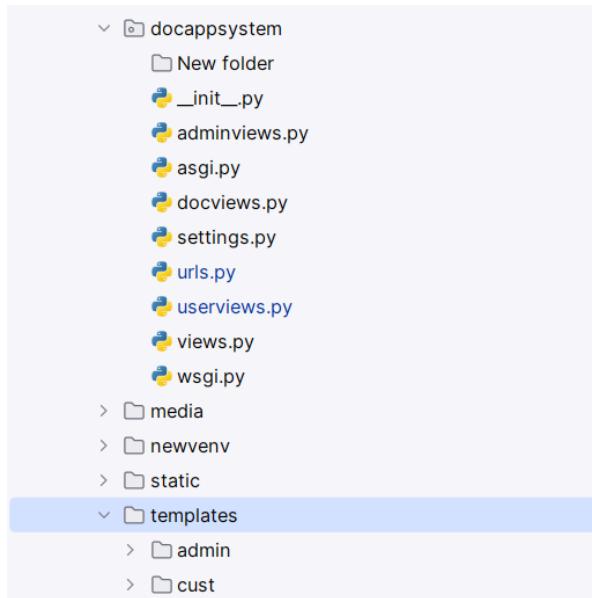
Template: Templates are responsible for the presentation layer, determining the layout and structure of the user interface. Django's template engine facilitates the creation of dynamic HTML content, enabling us to design interactive and user-friendly appointment booking interfaces. Templates seamlessly integrate with views, allowing the rendering of dynamic data fetched from models.

By leveraging Django's MVT framework, we were able to develop a robust and scalable doctor appointment system that streamlines the scheduling process,

enhances user experience, and ensures efficient management of patient-doctor interactions.

Django also provides a way to navigate around the different pages in a website. When a user requests a URL, Django decides which view it will send it to. This is done in a file called urls.py.

Django receives the URL, checks the urls.py file, and calls the view that matches the URL. The view, located in views.py, checks for relevant models. The models are imported from the models.py file. The view then sends the data to a specific template in the template folder. The template contains HTML and Django tags, and with the data it returns finished HTML content back to the browser.



SONAR LINT

SonarLint is a static code analysis tool that helps developers identify and fix code quality and security issues as they write code. It integrates seamlessly with popular integrated development environments (IDEs), providing real-time feedback on potential issues directly within the developer's workflow.

Overall, SonarLint is a valuable tool for developers looking to improve code quality, enhance security, and maintain consistency across their projects by identifying and addressing potential issues early in the development lifecycle.

The screenshot shows a code editor with two tabs: "userhome.html" and "userviews.py 9+, M". The "userviews.py" tab is active, displaying Python code related to a Django application. A SonarLint rule description window is open over the code, specifically for rule S1542: "Function names should comply with a naming convention". The rule states that function names should be lowercase with underscores, matching the PEP8 convention. It provides examples like `calculate_area`, `print_hello`, and `process_data`. A "Noncompliant code example" section shows a function definition with a red background: `def MyFunction(a,b):`. Below the editor, a "PROBLEMS" panel shows 10 issues, mostly from sonarlint, related to naming conventions and unused variables.

File	Source
doctorappointment\docappsystem\docappy...	sonarlint

This screenshot is similar to the first one but shows PyLance rule descriptions instead of SonarLint. The "userviews.py" tab is active, and a PyLance rule description window is open for rule reportMissingImport, which states that imports like `django.contrib.messages` could not be resolved. It also mentions rule reportUndefinedVariable for undefined variables like "context". The "PROBLEMS" panel shows 10 issues, with the first few being PyLance reports and the rest being sonarlint reports, mirroring the structure of the first screenshot.

File	Source
doctorappointment\docappsystem\docappy...	_generated_di...
doctorappointment\docappsystem\docappy...	_generated_di...
doctorappointment\docappsystem\docappy...	sonarlint
doctorappointment\docappsystem\docappy...	_generated_di...
doctorappointment\docappsystem\docappy...	sonarlint
doctorappointment\docappsystem\docappy...	_generated_di...
doctorappointment\docappsystem\docappy...	sonarlint
doctorappointment\docappsystem\docappy...	_generated_di...
doctorappointment\docappsystem\docappy...	sonarlint
doctorappointment\docappsystem\docappy...	_generated_di...

```

💡 Remove the unused local variable "appointmentdetails". sonarlint(python:S1481) [Ln 54, Col 9]
⚠️ "UserReg" is not defined Pylance(reportUndefinedVariable) [Ln 59, Col 16]
⚠️ Rename function "User_Search_Appointments" to match the regular expression ^[a-z_][a-z0-9_]*$. sonarlint(python:S1542) [Ln 75, Col 5]
⚠️ Define a constant instead of duplicating this literal 'search-appointment.html' 3 times. [+2 locations] sonarlint(python:S1192) [Ln 85, Col 36]
⚠️ "Patient" is not defined Pylance(reportUndefinedVariable) [Ln 89, Col 23]
⚠️ Rename function "View_Appointment_Details" to match the regular expression ^[a-z_][a-z0-9_]*$. sonarlint(python:S1542) [Ln 94, Col 5]
⚠️ "request" is not defined Pylance(reportUndefinedVariable) [Ln 104, Col 19]

```

GITHUB

GitHub is a web-based platform built for version control using Git. It's primarily used for hosting and sharing code repositories, but it also offers features for issue tracking, project management, and collaboration. Developers and teams use GitHub to store and manage their codebases, track changes, collaborate on projects, and coordinate their work. It's widely used in the software development community and has become an integral part of the development workflow for many organizations and open-source projects.

The screenshot shows a GitHub repository interface. On the left, there is a sidebar with a tree view of the repository's directory structure. The main area displays a list of commits from a user named 'giselle06'. Each commit includes the file or folder name, the last commit message, and the date of the commit.

Name	Last commit message	Last commit date
...		
idea	files added	2 days ago
dasapp	added urls to userpanel	12 hours ago
docappsystem	edits made at 7 am	4 hours ago
media/profile_pic	Create Screenshot_20221113_190423.png	16 hours ago
static	files added	2 days ago
templates	index update	1 hour ago
db.sqlite3	files added	2 days ago
manage.py	files added	2 days ago

JIRA

JIRA is a project management tool used for issue tracking, bug tracking, and agile project management. It offers customizable workflows, agile boards, integration with various tools, reporting, and analytics. It helps teams manage projects effectively, streamline workflows, and collaborate efficiently.

The screenshot shows the Jira software interface. At the top, there are navigation links: Jira, Your work, Projects, Filters, Dashboards, Teams, Plans, Apps, and a Create button. To the right are Upgrade and Search buttons. Below the header, the path 'Projects / Doc_App' is shown, followed by the word 'Issues'. There are buttons for Share, Export issues, and Go to all issues. A search bar labeled 'Search issues' is present, along with filters for Project (set to Doc_App), Type, Status, Assignee, More, and Save filter.

The main area displays a list of issues under the 'Created' filter. Issue DA-11 is selected, titled 'Doctor Appointment app'. It has a description: 'Doctor Appointment app that can make scheduling appointments convenient according to doctors availability. Also includes patients profile info and previous history making it easier for doctors to prescribe correct medication'. Below the main issue, there is a section for 'Child issues' with two items: DA-9 and DA-2, both marked as 'IN PROGRESS'.

APACHE

Apache is a widely used open-source web server software. It serves web content over the internet, supports multiple operating systems, and can be extended with modules for additional functionality. Its modular architecture, performance, scalability, and security features make it a popular choice for hosting websites and web applications.

The screenshot shows the phpMyAdmin interface. On the left is a sidebar with a tree view of databases: cent, docaspythondb, auth_group, auth_group_permissions, auth_permission, dasapp_appointment, dasapp_customuser, dasapp_customuser_groups, dasapp_customuser_user_permissions, dasapp_doctorreg, dasapp_page, dasapp_specialization, django_admin_log, django_content_type, and django_migrations. The main area has four tabs: General settings, Appearance settings, Database server, and Web server.

- General settings:** Server connection collation is set to utf8mb4_unicode_ci.
- Appearance settings:** Language is English and the theme is pmahomme.
- Database server:**
 - Server: 127.0.0.1 via TCP/IP
 - Server type: MariaDB
 - Server connection: SSL is not being used
 - Server version: 10.4.32-MariaDB - mariadb.org binary distribution
 - Protocol version: 10
 - User: root@localhost
 - Server charset: UTF-8 Unicode (utf8mb4)
- Web server:**
 - Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12
 - Database client version: libmysql - mysqld 8.2.12
 - PHP extension: mysqli curl mbstring
 - PHP version: 8.2.12

Use Case Diagrams:

MySQL Data Tables

Doctor Appointment System(DAS) contain 14 tables(In this MySQL 5 table is customized and 9 table made by default in django)

The screenshot shows the phpMyAdmin interface for the 'docaspythondb' database. On the left, a tree view lists various tables under the 'New' category, including 'auth_group', 'auth_group_permissions', 'auth_permission', 'dasapp_appointment', 'dasapp_customuser', 'dasapp_customuser_groups', 'dasapp_customuser_user_pe', 'dasapp_doctorreg', 'dasapp_page', 'dasapp_specialization', 'django_admin_log', 'django_content_type', 'django_migrations', 'django_session', and 'patients'. On the right, a main panel displays a table of all 14 tables with columns for 'Table' and 'Action'. Each row includes a checkbox, a star icon, and links for 'Browse', 'Structure', 'Search', and 'Insert'.

Table	Action
auth_group	Browse Structure Search Insert
auth_group_permissions	Browse Structure Search Insert
auth_permission	Browse Structure Search Insert
dasapp_appointment	Browse Structure Search Insert
dasapp_customuser	Browse Structure Search Insert
dasapp_customuser_groups	Browse Structure Search Insert
dasapp_customuser_user_pe	Browse Structure Search Insert
dasapp_doctorreg	Browse Structure Search Insert
dasapp_page	Browse Structure Search Insert
dasapp_specialization	Browse Structure Search Insert
django_admin_log	Browse Structure Search Insert
django_content_type	Browse Structure Search Insert
django_migrations	Browse Structure Search Insert
django_session	Browse Structure Search Insert

Doctor Table:(Table name is dasapp_doctorreg)

This store has doctor personal and login details.

The screenshot shows the 'Structure' tab for the 'dasapp_doctorreg' table. It displays the following columns:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	bigint(20)			No	None		AUTO_INCREMENT	Change Drop More
2	mobilenumber	varchar(11)	utf8mb4_general_ci		No	None			Change Drop More
3	regdate_at	datetime(6)			No	None			Change Drop More
4	updated_at	datetime(6)			No	None			Change Drop More
5	admin_id	bigint(20)			Yes	NULL			Change Drop More
6	specialization_id_id	bigint(20)			No	None			Change Drop More

Below the table, there are buttons for 'Check all', 'With selected:', and various column operations like 'Change', 'Drop', 'Primary', 'Unique', 'Index', 'Spatial', 'Fulltext', and 'Remove from central columns'.

Server: 127.0.0.1 » Database: docaspythondb » Table: dasapp_doctorreg

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking

Showing rows 0 - 11 (12 total, Query took 0.0012 seconds.)

SELECT * FROM `dasapp_doctorreg`

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	id	mobilenumber	regdate_at	updated_at	admin_id	specialization_id_id
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	1	7894561236	2024-02-26 06:29:39.005447	2024-02-26 06:29:39.005447	2	3
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	2	7897979878	2024-02-26 06:30:07.971414	2024-02-26 06:30:07.971414	3	2
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	3	6464646445	2024-02-26 06:30:55.169334	2024-02-26 06:30:55.169334	4	10
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	4	4464564654	2024-02-26 06:31:40.324768	2024-02-26 06:31:40.324768	5	11
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	5	6545646546	2024-02-26 06:32:24.094640	2024-02-26 06:32:24.095638	6	9
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	6	1414141425	2024-03-06 16:56:05.764937	2024-03-06 16:56:05.764937	7	13
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	7	1231231231	2024-03-06 17:22:41.637519	2024-03-06 17:22:41.637519	8	1
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	8	9999999999	2024-04-18 09:53:51.442072	2024-04-18 09:53:51.442072	9	11
	<input type="checkbox"/>	<input type="checkbox"/> Edit	<input type="checkbox"/> Copy	<input type="checkbox"/> Delete	9	1234567890	2024-04-18 10:49:26.575977	2024-04-18 10:49:26.575977	11	6

Specialization Table(Table name is dasapp_specialization)

This table stores the specialization of doctors.

Server: 127.0.0.1 » Database: docaspythondb » Table: dasapp_specialization

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#) [Privileges](#) [Op](#)

Showing rows 0 - 12 (13 total, Query took 0.0003 seconds.)

```
SELECT * FROM `dasapp_specialization`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

[Extra options](#)

			id	sname	created_at	updated_at
<input type="checkbox"/>	Edit	Copy	Delete	1 Orthopedics	2024-02-26 06:24:12.101985	2024-02-26 06:24:12.101985
<input type="checkbox"/>	Edit	Copy	Delete	2 Internal Medicine	2024-02-26 06:24:25.653848	2024-02-26 06:24:25.653848
<input type="checkbox"/>	Edit	Copy	Delete	3 Obstetrics and Gynecology	2024-02-26 06:24:35.592772	2024-02-26 06:24:35.592772
<input type="checkbox"/>	Edit	Copy	Delete	4 Dermatology	2024-02-26 06:24:45.212670	2024-02-26 06:24:45.212670
<input type="checkbox"/>	Edit	Copy	Delete	5 Pediatrics	2024-02-26 06:24:52.493340	2024-02-26 06:24:52.493340
<input type="checkbox"/>	Edit	Copy	Delete	6 Radiology	2024-02-26 06:25:00.364941	2024-02-26 06:25:00.364941
<input type="checkbox"/>	Edit	Copy	Delete	7 General Surgery	2024-02-26 06:25:09.110658	2024-02-26 06:25:09.110658
<input type="checkbox"/>	Edit	Copy	Delete	8 Ophthalmology	2024-02-26 06:25:16.661938	2024-02-26 06:25:16.662934

Appointment Table: (Table name is dasapp_appointment)

This table stores the details of appointments and doctor remark.

Server: 127.0.0.1 » Database: docaspythondb » Table: dasapp_appointment

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#) [Privileges](#) [Operations](#) [Tracking](#)

[Table structure](#) [Relation view](#)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	bigint(20)			No	None		AUTO_INCREMENT	Change Drop More
2	appointmentnumber	int(11)			No	None			Change Drop More
3	fullname	varchar(250)	utf8mb4_general_ci		No	None			Change Drop More
4	mobilenumber	varchar(11)	utf8mb4_general_ci		No	None			Change Drop More
5	email	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
6	date_of_appointment	varchar(250)	utf8mb4_general_ci		No	None			Change Drop More
7	time_of_appointment	varchar(250)	utf8mb4_general_ci		No	None			Change Drop More
8	additional_msg	longtext	utf8mb4_general_ci		No	None			Change Drop More
9	created_at	datetime(6)			No	None			Change Drop More
10	updated_at	datetime(6)			No	None			Change Drop More
11	doctor_id_id	bigint(20)			No	None			Change Drop More
12	prescription	longtext	utf8mb4_general_ci		No	None			Change Drop More
13	remark	varchar(250)	utf8mb4_general_ci		No	None			Change Drop More
14	status	varchar(200)	utf8mb4_general_ci		No	None			Change Drop More
15	recommendedtest	longtext	utf8mb4_general_ci		No	None			Change Drop More

Custom User Table(Table name is dasapp_customuser)

This table stores the login details of doctors and admin.

Server: 127.0.0.1 » Database: docaspythondb » Table: dasapp_customuser

[Browse](#) [Structure](#) [SQL](#) [Search](#) [Insert](#) [Export](#) [Import](#) [Privileges](#) [Operations](#) [Tracking](#) [Triggers](#)

Showing rows 0 - 12 (13 total, Query took 0.0005 seconds.)

```
SELECT * FROM `dasapp_customuser`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

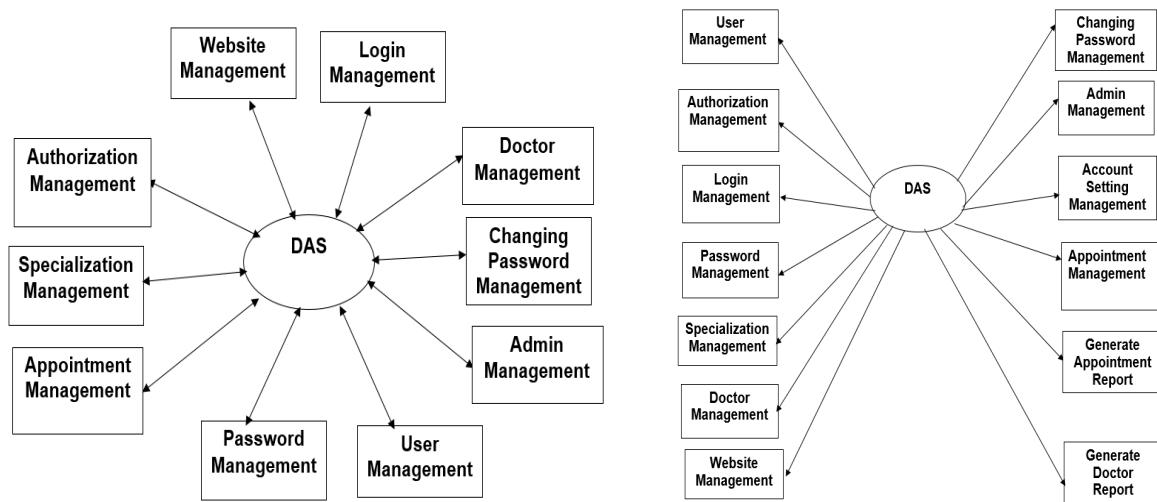
[Extra options](#)

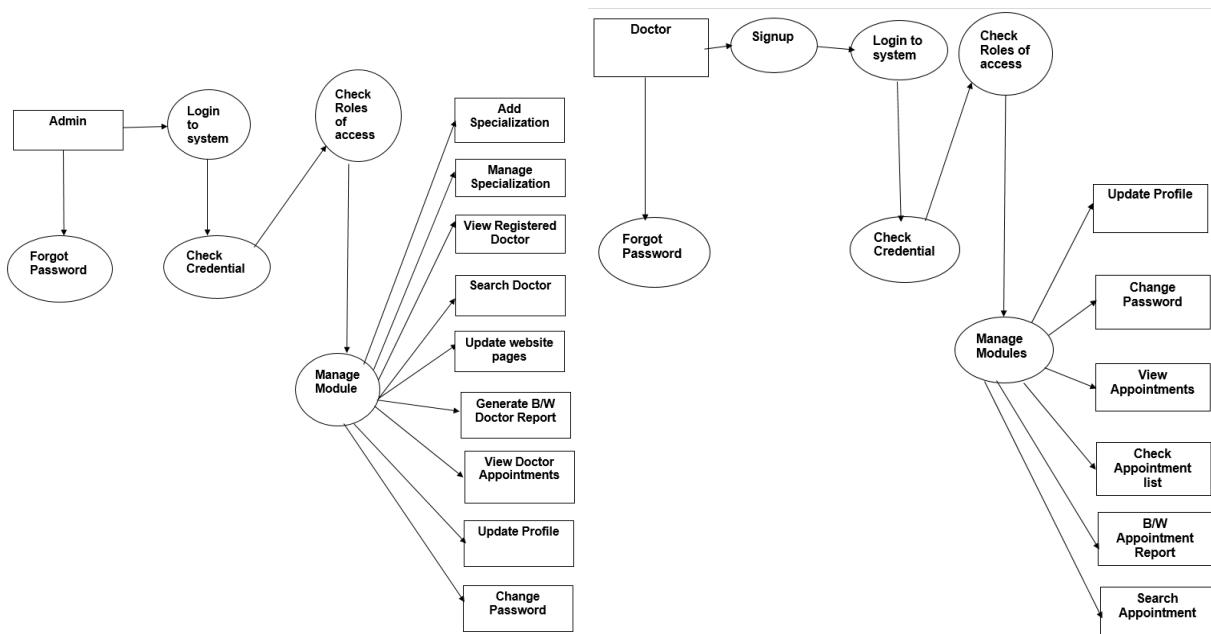
			id	password	is_superuser	username	first_name	last_name	email	is_staff	is_active	date_joined
<input type="checkbox"/>	Edit Copy Delete	1	pbkdf2_sha256	202	1	admin	Admin	Sample	admin@gmail.com	1	1	2024-02-2
<input type="checkbox"/>	Edit Copy Delete	2	pbkdf2_sha256	NU	0	gar123	Garima	Singh	garima@gmail.com	0	1	2024-02-2
<input type="checkbox"/>	Edit Copy Delete	3	pbkdf2_sha256	202	0	abir123	Abir	Kumar	abir@gmail.com	0	1	2024-02-2
<input type="checkbox"/>	Edit Copy Delete	4	pbkdf2_sha256	NU	0	som123	Soumya	Yadav	som@gmail.com	0	1	2024-02-2
<input type="checkbox"/>	Edit Copy Delete	5	pbkdf2_sha256	NU	0	har123	Harishankar	Singh	har@gmail.com	0	1	2024-02-2
<input type="checkbox"/>	Edit Copy Delete	6	pbkdf2_sha256	NU	0	renu123	Renu	Gupta	renu@gmail.com	0	1	2024-02-2
<input type="checkbox"/>	Edit Copy Delete	7	pbkdf2_sha256	202	0	drak30	Anuj	Kumar	ak@gmail.com	0	1	2024-03-0
<input type="checkbox"/>	Edit Copy Delete	8	pbkdf2_sha256	202	0	johndoe12	John	Doe	johndoe12@gmail.com	0	1	2024-03-0
<input type="checkbox"/>	Edit Copy Delete	9	pbkdf2_sha256	202	0	saas	ass	ssa	assssa@se.com	0	1	2024-04-1
<input type="checkbox"/>	Edit Copy Delete	11	pbkdf2_sha256	202	0	kiran	Kiran	Yadu	Kiran@gmail.com	0	1	2024-04-1

auth_permission table Structure : This table in Django is used to store information about the permissions defined.

Server: 127.0.0.1 » Database: docaspythondb » Table: auth_permission					
	Browse	Structure	SQL	Search	Insert
Extra options					
	Edit	Copy	Delete	id	name
<input type="checkbox"/>	Edit	Copy	Delete	1	Can add log entry
<input type="checkbox"/>	Edit	Copy	Delete	2	Can change log entry
<input type="checkbox"/>	Edit	Copy	Delete	3	Can delete log entry
<input type="checkbox"/>	Edit	Copy	Delete	4	Can view log entry
<input type="checkbox"/>	Edit	Copy	Delete	5	Can add permission
<input type="checkbox"/>	Edit	Copy	Delete	6	Can change permission
<input type="checkbox"/>	Edit	Copy	Delete	7	Can delete permission
<input type="checkbox"/>	Edit	Copy	Delete	8	Can view permission
<input type="checkbox"/>	Edit	Copy	Delete	9	Can add group
					content_type_id
					codename

DATA FLOW DIAGRAMS





SYSTEM TESTING

After all phases have been perfectly done, the system will be implemented to the server and the system can be used.

System Testing

The goal of the system testing process was to determine all faults in our project .The program was subjected to a set of test inputs and many explanations were made and based on these explanations it will be decided whether the program behaves as expected or not. Our Project went through two levels of testing

1. Unit testing

2. Integration testing

UNIT TESTING

Unit testing is commenced when a unit has been created and effectively reviewed .In order to test a single module we need to provide a complete environment i.e. besides the section we would require

- The procedures belonging to other units that the unit under test calls
- Non local data structures that module accesses
- A procedure to call the functions of the unit under test with appropriate parameters

1. Test for the admin module

- Testing admin login form-This form is used for login of administrator of the system. In this form we enter the username and password if both are correct, the administration page will open, otherwise if any of the data is wrong it will get redirected back to the login page and again ask the details.
- Report Generation: admin can generate reports from the main database.

INTEGRATION TESTING

In the Integration testing we test various combinations of the project module by providing the input.

The primary objective is to test the module interfaces in order to confirm that no errors are occurring when one module invokes the other module.

FUTURE IMPROVEMENTS

In addition to the existing functionalities, several enhancements can be incorporated into the system to further improve user experience, security, and efficiency. These future improvements include:

- User Authentication Enhancements:

Implement robust user authentication mechanisms, such as multi-factor authentication, to enhance security and protect user accounts from unauthorized access.

- Enhanced Appointment Booking:

Introduce advanced search filters for appointments, allowing users to search based on additional criteria such as preferred time slots, doctor availability, or treatment types. This will further streamline the appointment booking process and improve user satisfaction.

- Family Member Permissions:

Implement granular permissions for family members, allowing users to assign specific roles or access levels to each member. This feature will provide greater control over who can view, book, or manage appointments for each family member, enhancing privacy and security.

- Integration with Electronic Health Records (EHR):

Integrate the system with electronic health records (EHR) systems to enable seamless sharing of medical data between healthcare providers and users. This integration will improve coordination of care and provide healthcare professionals with access to comprehensive patient information.

- Telemedicine Support:

Introduce support for telemedicine appointments, allowing users to schedule and conduct virtual consultations with healthcare providers. Integration with

video conferencing platforms and secure messaging systems will enable users to receive medical advice and treatment remotely, increasing accessibility and convenience.

- **Data Encryption and Digital Signatures:**

Implement end-to-end encryption for sensitive data, such as medical records and personal information, to protect against unauthorized access or data breaches. Additionally, introduce digital signatures for verifying the authenticity and integrity of user data, further enhancing data security and trustworthiness.

- **Enhanced Reporting and Analytics:**

Develop comprehensive reporting and analytics capabilities, allowing users to track and analyze trends in appointment scheduling, medical conditions, and treatment outcomes. These insights will facilitate informed decision-making and support continuous improvement in healthcare delivery.

- **Mobile Application Development:**

Develop a dedicated mobile application for the system, providing users with on-the-go access to appointment booking, medical records, and other key features. A mobile app will enhance accessibility and usability, catering to users who prefer to manage their healthcare needs from their smartphones or tablets.

CONCLUSION

The Online Doctor Appointment System represents a transformative step towards enhancing healthcare accessibility, efficiency, and patient satisfaction. By leveraging cutting-edge technology and adhering to best practices in software development, the system embodies the future of healthcare delivery, empowering patients and healthcare professionals alike to navigate the complexities of modern healthcare with ease and confidence.

REFERENCES

[\(DOC\) Design and Development of Online Doctor Appointment System | Sharief Zamshed - Academia.edu](#)

<https://www.behance.net/search/projects/doctor%20dashboard>

<https://www.purrweb.com/blog/doctor-appointment-app-development-features-benefits-costs/>