

MEDWEB - A FULL STACK TELEMEDICINE APPLICATION USING CLOUD COMPUTING

Project report submitted in partial fulfillment of the requirement for the degree of

Bachelor of Technology

in

Computer Science and Engineering

By

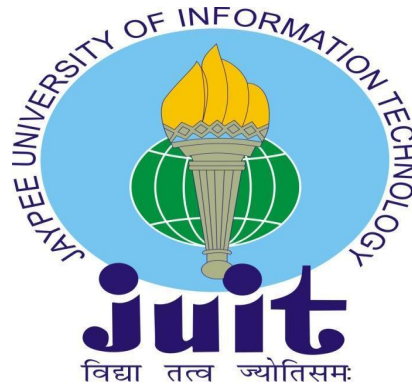
ANANYA MISHRA (191213)

VARSHA SINGH (191423)

UNDER THE SUPERVISION OF

Dr. Jagpreet Sidhu

Assistant Professor, Deptt. Of CSE & IT



Department of Computer Science & Engineering and Information Technology

**Jaypee University of Information Technology Waknaghat, Solan-173234, Himachal
Pradesh**

DECLARATION

I hereby declare that the work presented in this report entitled “**MedWeb- A Full Stack Telemedicine Application Using Cloud Computing**” in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology in Computer Science and Engineering/Information Technology** submitted in the department of Computer Science & Engineering and Information Technology, Jaypee University of Information Technology Waknaghat is an authentic record of my own work carried out over a period from January 2023 to May 2023 under the supervision of **Dr. Jagpreet Sidhu** (Assistant Professor(SG), Department of CSE Jaypee University of Information Technology). I also authenticate that I have carried out the above-mentioned project work under the proficiency stream **Cloud Computing**. The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Ananya Mishra

191213

Varsha Singh

191243

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

(Supervisor Signature)

Dr. Jagpreet Sidhu

Assistant Professor (SG)

Department of CSE Jaypee University of Information Technology

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT
PLAGIARISM VERIFICATION REPORT

Date:

Type of Document (Tick): PhD Thesis M.Tech Dissertation/ Report B.Tech Project Report Paper

Name: _____ Department: _____ Enrolment No _____

Contact No. _____ E-mail. _____

Name of the Supervisor: _____

Title of the Thesis/Dissertation/Project Report/Paper (In Capital letters): _____

UNDERTAKING

I undertake that I am aware of the plagiarism related norms/ regulations, if I found guilty of any plagiarism and copyright violations in the above thesis/report even after award of degree, the University reserves the rights to withdraw/revoke my degree/report. Kindly allow me to avail Plagiarism verification report for the document mentioned above.

Complete Thesis/Report Pages Detail:

- Total No. of Pages =
- Total No. of Preliminary pages =
- Total No. of pages accommodate bibliography/references =

(Signature of Student)

FOR DEPARTMENT USE

We have checked the thesis/report as per norms and found **Similarity Index** at(%). Therefore, we are forwarding the complete thesis/report for final plagiarism check. The plagiarism verification report may be handed over to the candidate.

(Signature of Guide/Supervisor)

Signature of HOD

FOR LRC USE

The above document was scanned for plagiarism check. The outcome of the same is reported below:

Copy Received on	Excluded	Similarity Index (%)	Generated Plagiarism Report Details (Title, Abstract & Chapters)	
	<ul style="list-style-type: none"> • All Preliminary Pages • Bibliography/Images/Quotes • 14 Words String 		Word Counts	
Report Generated on			Character Counts	
		Submission ID	Total Pages Scanned	
			File Size	

Checked by
Name & Signature

Librarian

Please send your complete thesis/report in (PDF) with Title Page, Abstract and Chapters in (Word File) through the supervisor at plagcheck.juit@gmail.com

ACKNOWLEDGEMENT

Firstly, I express my heartiest thanks and gratefulness to Almighty God for His divine blessing that made it possible to complete the project work successfully. I am grateful and wish my profound indebtedness to Supervisor **Dr. Jagpreet Sidhu**, Assistant Professor(SG), Department of CSE Jaypee University of Information Technology, Wakhnaghat. His deep knowledge & keen interest in the field of "Cloud Computing and Technologies" helped us to carry out this Project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts, and correcting them at all stages have made it possible to complete this Project. I would also generously welcome each one of those individuals who have helped me straightforwardly or in a roundabout way in making this project a win. In this unique situation, I want to thank the various staff individuals, both educating and non-instructing, which have developed their convenient help and facilitated my undertaking. Finally, I must acknowledge with due respect the constant support and patience of my parents.

Ananya Mishra

191213

Varsha Singh

191423

Table of Contents

Content	Page no.
Certificate	i
Plagiarism Certificate	ii
Acknowledgement	iv
Table of Figures	v
Abstract	vi
Chapter-1 INTRODUCTION	1-12
1.1 Introduction	1-5
1.2 Problem Statement	6
1.3 Objectives	7
1.4 Methodology	8-12
Chapter-2 LITERATURE SURVEY	13-23
Chapter-3 SYSTEM DEVELOPMENT	24-37
Chapter-4 PERFORMANCE ANALYSIS	38-53
Chapter-5 CONCLUSIONS	54-60
5.1 Conclusions	54-58
5.2 Future Scope	59-60
REFERENCES	61
APPENDICES	62-65

List of Figures

Figure 1	Ignored minor Health issues Due to Pandemic?
Figure 2	Dependent on your wards for Health Care
Figure 3	Healthcare providers and patients using smartphones as a result
Figure 4	Desire to use mobile apps as a result
Figure 5	Interest in the cloud and Virtual Technology as a result
Figure 6	Results of the plan regarding Cloud Based technology
Figure 7	User Flow diagram of discussed paper
Figure 8	Use case diagram
Figure 9	Login page
Figure 10	Sign up page(for new registrations)
Figure 11	Landing Page
Figure 12	How to book appointments
Figure 13	Doctor can view his upcoming appointments
Figure 14	Join Appointment Call
Figure 15	Uploading files
Figure 16	Database in Firestore showing userinfo
Figure 17	Database in Firestore showing Appointments
Figure 18	HTTP Request-Response Cycle
Figure 19	Communication pattern between a service provider and a service consumer..

ABSTRACT

In today's healthcare world, Genuine health advice, as well as easy accessibility, is the key and this is one of the most attractive advantages of our telemedicine application. In times like these, when the entire world is fighting the pandemic and people are trying to protect themselves from the virus in any way possible, all of us are going through tough changes that we never thought of.

This pandemic has not only altered our way of living in general but has also changed our attitude and perspective to today's every smaller aspect of livelihood. It is high time now that we make efficient use of the enormous available technology and data in order to make our lives easier and much safer. Thus we came up with the vision of a cloud-based telemedicine application, 'MedWeb'.

It will not only make the Health Care system of the country highly efficient but also it will make it 1000 times faster and more convenient, and at the same time it will also help to prevent the spread of communicable diseases such as COVID-19, flu, and other infectious diseases. It also aims towards the better treatment of Old Aged People and also towards the people who need OPD/Doctors Assistance quite often from their birth or due to an accident. Also, this can be very helpful for pregnant women.

With the help of this application, it will be possible to make a video or phone appointment between a patient and their healthcare individual, thus benefiting both health and convenience. More and more people are now showing interest in virtual diagnosis, and doctors too are finding it pretty convenient to see patients by computer and smartphones. Improved technology has made this possible and even easier, even for those who don't consider themselves computer savvy.

Chapter 01: INTRODUCTION

1.1 Introduction

Meeting specialists actually or taking arrangements by remaining in line or doing customs at the medical clinic is a very tedious and excruciating interaction. On the off chance that a crisis happens, then you could be sitting tight for extended periods of time until the specialist comes. So you need to skirt an entire day on the off chance that you became ill on an especially bustling day. One can obtain greater physical relief or save time by consulting with a professional doctor online anywhere, at any time, by seeing an online doctor, or through a virtual appointment with a doctor.

Today, you can get anything on request, from cabs to food. While these administrations have been well known for quite a while, on-request specialist meetings came into the scene as of late. The World Wellbeing Association (WHO) makes sense of virtual determination as "mending from good ways". It is the utilization of broadcast communications innovation and IT to give distant medical care administrations to patients.

Today, people never again need to book an in-person visit to a specialist to get treatment. The utilization of exceptionally secure video and sound associations makes it feasible for specialists to counsel patients who live where it is difficult to connect consistently every day. You don't need to head to the specialist's office or center, park, walk or sit in a lounge area when you're wiped out. You can see your primary care physician from the solace of your home.

The utilization of the phone for patient-clinician counsels and the utilization of radio to interface crisis clinical experts to clinical offices fall inside the

classification of ordinary use. On the furthest edge of the telemedicine range are improvements that are still, to a great extent, in the exploratory stages, including telesurgery, in which a specialist utilizes material and viewable prompts to coordinate mechanical devices as they carry out procedures at an alternate area. There is a wide range of visual, sound, and information transmission innovations and applications that fall between these two limits. For "constant" analytic and remedial purposes, some, as to some degree expensive intelligent video conferencing, empower experts to see, hear, look at, question, and direct far-off patients.

Others, in view of "store and forward" advances, empower the saving and sensibly modest transmission of computerized pictures and other data to specialists who can get and decipher them at whatever point is helpful, giving really booking adaptability to those on the two closures of the correspondences. These numerous advances have many ebb and flow and are expected to involve proficient training, research, general well-being, and organization, notwithstanding tolerant treatment. With such countless applications, it could be feasible to all the more equally disseminate the cost of significant data and correspondence speculations.

The shortage of exhaustive examinations of telemedicine applications' intolerant consideration filled in as the impulse for this paper. It offers a far-reaching structure for surveying clinical telemedicine applications and presents the defense for additional calculated and exhaustive assessments of their effects on medical care quality, openness, expenses, and worthiness in contrast with elective choices. Such assessments are urgent for telemedicine, similarly with any well-being innovation or administration, for various reasons. They could: Furnish doctors and patients with fitting affirmation or mindfulness in regards to telemedicine applications; Direct government officials who are choosing whether to advance telemedicine by speeding up foundation improvement, supporting specific

telemedicine projects, or bringing down authoritative boundaries; Assist the people with having put resources into telemedicine track down strategies to make it more powerful by teaching wellbeing plan the board who are discussing whether clinical telemedicine is functional, reasonable, and adequate to patients and doctors.

Today, the vast majority approach fundamental shrewd gadgets like cell phones and PCs. With further developed openness, people in country regions and occupied metropolitan regions can interface with a medical services supplier effectively and helpfully.

Telemedicine is now acquiring fame beyond provincial regions as the country's medical services framework goes through massive changes. Overseas care plans might discover some telemedicine applications engaging in the metropolitan and rural regions they, by and large serve, to the degree that telemedicine offers components for concentrating subject matter experts, bringing down costs for particular consideration, and helping essential consideration experts. Because of diminished subsidizing and avoidance from neighborhood oversight care organizations, some college clinical focuses are investigating telemedicine arrangements with an end goal to make new local, public, and overall business sectors for their exceptionally qualified doctors. As they search remotely for their administrations, autonomous expert gatherings, multi organization clinical consortia, and different associations are taking a gander at telemedicine.

Many types of clinical telemedicine are still distant from being consistently coordinated into most parts of medical care conveyance, in spite of its growing purposes and clients. Given the absence of intensive and reliable information and the pace of progress, a summed-up evaluation of the condition of telemedicine today is essential. Think at the size of the Indian medical services framework, for example.

250 million patients and imminent patients, generally. Most people have, without a doubt, called for clinical exhortation or data. An ever-increasing number of Americans currently own PCs and programming that empower them to get to clinical data sets (some of which were made for doctors or scientists as opposed to patients) and to email clinicians and others. The level of individuals who have encountered a "electronic house call," a video conference with a distant clinical master, or some other intelligent, varying media telemedicine application is hazy, however, it is undeniably tiny.

1.5 million medical caretakers, over a portion of 1,000,000 specialists, and a few other medical care laborers. Once more, most professionals have certainly utilized the telephone to examine patients' therapy; many have gone to video chats for proceeding with clinical training; and a few subject matter experts, similar to radiologists, are securing a ton of involvement by sending pictures for meetings. More specialists presently have online admittance to information bases like Medline from the Public Library of Medication and different devices that let them search clinical writing. On the Internet, a rising number of medical care establishments have home sites that offer data and connections to data accessible at different destinations.

A large number of medical clinics, nursing homes, facilities, and other medical services establishments. The number of medical services organizations that have progressed telemedicine limits (e.g., video as well as phone and fax) isn't legitimate. A review of roughly 2,400 provincial clinics led by the government Office of Country Wellbeing Strategy found that almost 20% revealed some telemedicine benefits however that 60% detailed no designs for telemedicine (Jones, 1996). Scholarly clinical focuses, local area clinics, and different organizations have made Internet pages that serve both as data sources and as advertising instruments. To address interior issues and outside requests,

workplaces, and emergency clinics are being renovated to more readily oblige data innovations that require distinctively designed space for individuals and gear. The electronic patient record is progressively perceived to be a need, albeit commonsense deterrent to execution finding an opportunity to survive.

The reception of telemedicine advancements is as yet hampered by specialized, clinical, hierarchical, and social issues, as well as by strategy obstructions and ambiguities encompassing installment, permitting clinical obligation, and different issues. Although many projects actually depend on financing from public authority and business, some applications seem to have a superior opportunity of long haul independence than others.

Telemedicine is a creative arrangement that uses innovation to offer virtual clinical conferences and medicines. MEDWEB is one such stage that associates patients with clinical experts, medical caretakers, or specialists for virtual therapy plans and analyses from any place and whenever. Patients can present a video or message depiction of their side effects to transfer to the clinical expert, and MEDWEB helps plan arrangements among patients and clinical experts in light of accessibility. Secure video conferencing is likewise a possibility for these collaborations. The two patients and clinical experts can pursue a record, with patients being able to see arrangement history, plan new arrangements, and access their electronic clinical record (EMR) from any place. MEDWEB gives a simple and incorporated way for patients to get to their EMR and required records, making the method involved with getting to clinical history practically immediate for the doctor, as patients can transfer their EMR and related archives.

Generally, telemedicine stages like MEDWEB give a proficient and helpful way for patients to get medical services, particularly for the people who might have restricted admittance to clinical offices or can't truly visit a clinical expert.

1.2 Problem Statement

In today's healthcare world, genuine health advice, as well as easy accessibility, is the key. One can get anything on demand, from taxis to food. On-demand doctor consultations came into the scene recently.

This project allows patients to use a web application to send/receive medical treatment in the comfort of their homes. The primary purpose of this project is to use the internet and cloud features to provide healthcare facilities in the comfort of your home and helps you avoid any traveling cost and also makes these health facilities available to old aged people or pregnant women, and to allow patients to use a web application to send/receive some information, and avoid or at least reduce the number of times per year that a patient must travel.

As in a portion of the current frameworks, the Framework UI is basic or lacks extra usefulness or highlights. As some framework is connected with the singular clinic, excludes every one of the clinics, or shows generally closed medical clinics or close specialists according to our area. Finding or looking at the close-by specialists or emergency clinics, and on the premise that classifications are displayed to patients according to the area, this component isn't accessible in a few existing frameworks. Video calling and talking with specialists by getting legitimate conferences or remedies for the fulfillment of patients. This component is likewise not accessible in a few existing frameworks. Arrangement booking or planning is straightforward, just getting the fundamental subtleties of patients, or then again, and information security is less, not having extra highlights of getting pictures from the patient at the hour of arrangement whether the patient has a sensitivity or skin-related issue. Information base taking care of and keeping up with and dependable working of the framework or ease of use is likewise an issue in a few existing frameworks. MEDWEB will beat these highlights.

1.3 Objectives

The goals of this project are:

1. **Virtual Prescription:** We have taken the doctor-patient relationship into a new way of interaction by facilitating virtual prescriptions with reminders for your medicines and many more.
2. **Storage of all previous Health Records:** All patients' medical information will be stored in highly secured servers with the help of cloud technology, which is known for its reliable storage and access control options and features.
3. **Consult Your Doctor via Video Conferencing:** You can schedule a call with your very doctor at the convenience of you both. It's up to the patient and the doctor when they can schedule an appointment.
4. **Encryption and Security Protocols:** Using security and encryption protocols, this project aims to allow patients and doctors to keep confidential patients' data secure and safe.
5. **Appointment Management Appointment:** scheduling allows patients to select an available slot from the doctor's calendar and schedule appointments.
6. **Stay away from contact with debilitated patients:** By limiting patients' exposure to one another, the application can also aid in stopping the spread of infectious diseases.

1.4 Methodology

There are numerous real-world uses for the application. It can be utilized for maintaining and monitoring patient records, monitoring healthcare, and digitally connecting with patients. Doctors have access to patient's test results and medication histories, which helps them provide more effective care in less time and at a lower cost. Demographic-based customization is conceivable.

A telemedicine application with the following salient characteristics:

- Patients can receive expert observation and consultation for optimal health.
- You can install a web application on the landing page.
- Can provide a message pop-up, making it unique to use.
- Patients can schedule an appointment with the doctor based on their needs.
- Through video chatting, doctors can provide prescriptions to patients and make appointments.

Prerequisites Analysis: 64% of the people surveyed reported that due to this pandemic, there has been a lot of ignorance of their health condition as it was not feasible to go and check out a Doctor for minor health issues.

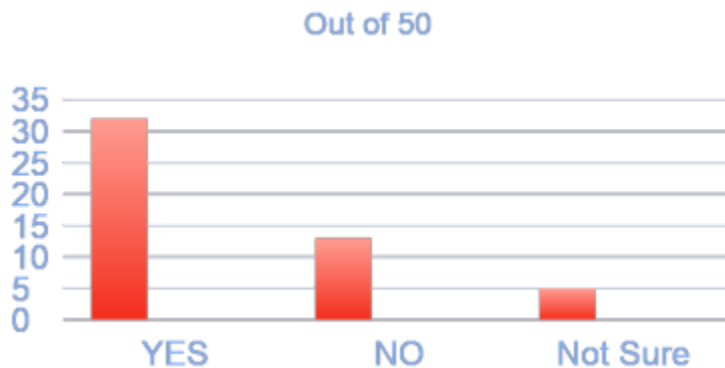


Figure 1: Ignored minor Health issues Due to the Pandemic? [2]

48% of Old Aged people agreed that, at times, they and their health care were being ignored six because of lack of time with their wards, lack of previous Records, and lack of continuous guidance.

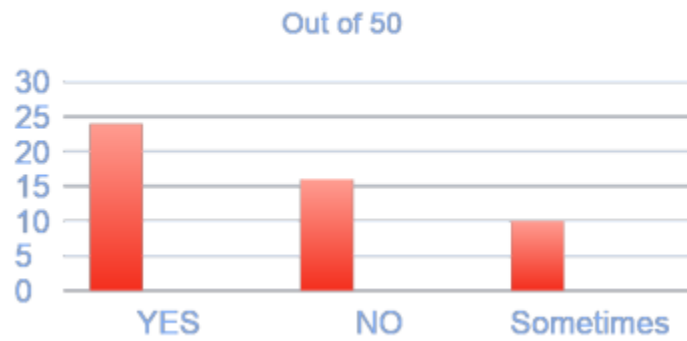


Figure 2: Dependent on your wards for Health Care [2]

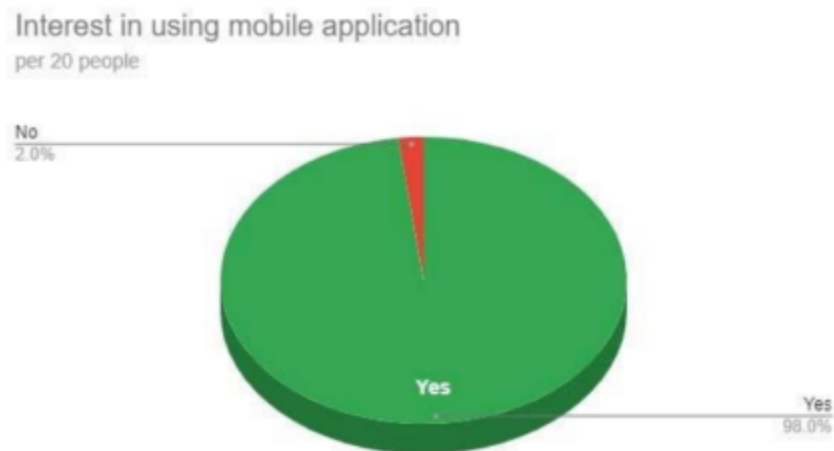


Figure 3: Healthcare providers and patients use smartphones as a result. [2]

From this survey, we all know from healthcare workers to patients, the majority of them own a smartphone.



Figure 4: Desire to use mobile apps as a result [2]

From this survey, principally, they like to get services through mobile apps.

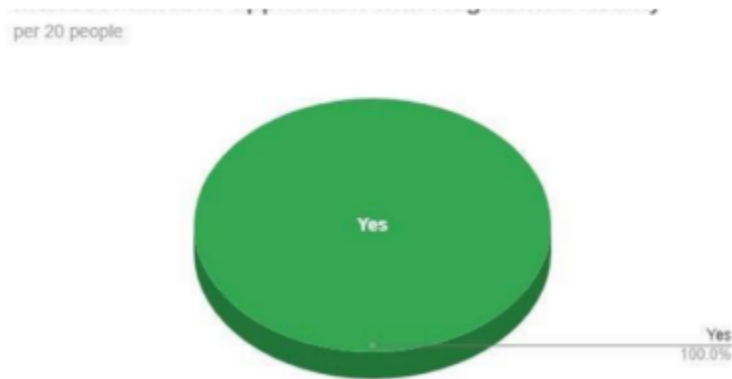


Figure 5: Interest in cloud and Virtual Technology as a result [2]

From this survey, we all know that everybody loves the cloud-based application in their smartphone. They surprisingly like this technology and find it beneficial.



Figure 6: Results of plan regarding Cloud Based technology [2]

From this survey, we realize that most clients need to utilize cloud technology, and the greater part of them are now presented with this technology.

Functional Requirements

- A Web Application that interacts with the user intelligently and provides them with all the requirements that would be hosted on a private domain with MongoDB as a backend Database.
- A 24*7 Service will be provided, and the patient's calls will be transferred to the Doctor On-Board at that present moment.
- The whole procedure would be even more efficient if the doctors make the necessary updates in the patients' database simultaneously, and one can do so by maintaining a unique patient ID.

- When a user logs into his/her account, he/she must be able to see his/her personal information on the screen.
- It must also prompt the user to ask for an appointment with the doctor.
- For each patient, the doctor must be able to see all the medical history and prescriptions and if there is any report, that the patient has sent.

Non-Functional Requirements

- Since the application must be accessible 24 hours a day, 7 days a week. So a PC with stable internet connection will be required.
- A good quality camera to ensure smooth video appointments.
- The system is accessed by multiple users simultaneously, so it has to offer a good response time (real time interaction), user interface. For that one must own a smart device or computer with good enough HD space, RAM memory, CPU etc.

MedWeb is a self-contained product made using full stack web technologies. The components to be used may include:

1. Frontend Design: HTML, CSS, JS and ReactJS
2. Backend: Firebase
3. User Authentication: Google Auth
4. Scheduling Appointments and Chats: Google Firebase/Calendly
5. Video Conferencing: Meet

Chapter 02: LITERATURE SURVEY

P. Matlani and N. D. Londhe, "A cloud computing based telemedicine service," 2019 IEEE Point-of-Care Healthcare Technologies (PHT), 2019, pp. 326-330, doi:10.1109/PHT.2013.6461351.

"The best belonging is well-being." Different strategies are created to more readily treat society's well being. As innovation arrives at its apex, it rapidly and effectively enters clinical medicines, leading to "telemedicine," a creation of innovation in medication. Quick internet providers are utilized to supplant sluggish and off-base tasks with exact and mistake-free methodology. At the point when appropriately verified, these systems empower ongoing information openness. The idea is based on constant video real-time and distributed computing.

A legitimate organization for the data is made accessible on the web, where it tends to be seen by qualified clinical staff. Telemedicine has been changed by distributed computing. Numerous clinical experts as of now, utilize advanced telehealth distributed computing applications. The lack of experts in the provincial clinical field is a main consideration in the underfunding of the rustic medical care framework. Then again, the usability and modern development of medical care administrations in metropolitan regions is a vital calculation of their prosperity.

With the extension of the Web, various medical services sites are currently open to offer appropriate answers for regular issues. It has been discussed how to really analyze a patient's condition during telemedicine interviews, where a specialist talks with them from a distance (through video gathering). Is the clinical expert on the opposite finish of the telemedicine interface responsible for the findings made?

The way that telemedicine frameworks are housed inside clinics makes it difficult for the clinical faculty to work or answer the telemedicine framework's down-to-earth tasks now and again. The shortfall of appropriate correspondence advances has obstructed various potential telemedicine programs.

1) most telemedicine applications seldom get sufficient data transfer capacity from phone lines.

2) The link wiring or different sorts of broadcast communications access fundamental for the utilization of telemedicine applications are absent in numerous country regions.

Issues in telemedicine applications:

1) Issues with the framework: A few spots have little data transmission, while others have exorbitant costs.

2) Issues with execution: to carry out telemedicine, experts, IT staff, and neighborhood doctors should get prepared at the town level.

3) Acknowledgment Issues: Utilizing trend-setting innovation can be excessively troublesome for the town specialist and local people.

The acknowledgment rate, notwithstanding, will presumably be high once the benefits are clear, as has been the situation with versatile communication and provincial internet providers.

4) Issues with feasibility: In India, tremendous clinic networks have commonly given telemedicine at no expense for patients. Furthermore, it expands their bed inhabitants if a tele patient needs hospitalization and gets exceptionally restless.

K. S. Gill, S. Saxena, and A. Sharma, "Taxonomy of Security Attacks on Cloud Environment: A Case Study on Telemedicine," 2019 Amity International Conference on Artificial Intelligence (AICAI), 2019, pp. 454-460, doi: 10.1109/AICAI.2019.8701363.

Several detective control measures have been put in place to lessen the harm done to telemedicine systems as a result of the aforementioned security threats. These standards aid in fostering trust between healthcare practitioners and patients in regard to the security and dependability of the provided services. Some governments develop tight policies to manage patient information transmission on telemedicine networks since the data communicated through telemedicine systems is very private and sensitive.

A law like the Health Insurance Portability and Accountability Act (HIPAA) was developed in the US to lessen the effects of security breaches. HIPAA exclusively regulates electronic health information and aims to improve the effectiveness of healthcare systems. By creating health standards, it creates health care systems.

The internet is bringing people together, and emerging technologies aim to improve the comfort, convenience, and security of human life. One such technology is cloud computing, which offers pay-per-use access to a wide range of services, including data access everywhere. The major topic of this study is IoT-based cloud computing telemedicine.

Using sensors, actuators, and cloud computing, telemedicine intends to deliver medical treatments from a remote location. In order to provide patients with medical care anywhere, anytime, specialized doctors who are unable to see patients in person can get patient information in the form of photos or real-time videos.

According to NIST, cloud computing offers five crucial features. Due to these qualities, cloud computing can be very useful in telemedicine, as shown in Table II. Additional traits include the ability to be used: The services are continuously offered; universal accessibility: Anyone in the world can use the services; improved communication and collaboration: scattered individuals from around the globe can easily connect and communicate in real-time; ecologically sound.

The use of green computing techniques by cloud service providers reduces negative environmental effects; Pay-per-use: Customers can gauge how much of a service they actually use and pay accordingly; Disaster recovery: Because backups are kept at various data centers, it is simple to restore data in the event of a catastrophe.

Cross-site scripting: This attack involves inserting malicious code into a web application from the client side, enabling the attacker to run harmful scripts. The most vulnerable web application to this specific attack is XSS.

Scenario: In telemedicine, an attacker may get a patient's cookies and impersonate him or her by accessing the patient's geolocation, certain files, webcam, and microphone.

Motive: The motivation includes accessing specific patient medical records as well as specific doctor records.

SQL injection: When a data-driven application is attacked, malicious SQL queries are injected into the user-filled entry fields (most often, forms).

Cookie manipulation: This dishonest tactic involves the attacker forging and manipulating cookies in order to steal the user's identity. Cookies store data such as IDs, passwords, and private information.

R. A. Razali and N. Jamil, "A Quick Review of Security Issues in Telemedicine," 2020 8th International Conference on Information Technology and Multimedia (ICIMU), 2020, pp. 162-165, doi: 10.1109/ICIMU49871.2020.9243549.

Due to the freedom that it provides with the help of cutting-edge technologies, telemedicine has become popular. The majority of developed nations use telemedicine services to give patients and clinicians quick, easy access to medical information and services. However, as more information and services are available online, data security is a growing problem.

Without adequate security measures in place, the weaknesses of telemedicine systems can be exploited, having detrimental effects on patients and the medical industry as a whole, including incorrect treatment and the loss of personal information. In order to manage the security concerns in telemedicine, this paper analyzes the security issues in telemedicine as an IOT system and identifies relevant security countermeasures from related works.

By combining remote technology-based virtual platforms to transfer medical data from one location to another via electronic communication, telemedicine is a system that supports healthcare, public health, and health administration. The system offers to help, care for, and manage patient recovery utilizing the system, which is another way it helps patients.

The field of health care has advanced thanks to telemedicine, which incorporates smart devices (also known as the Internet of Things (IoT)) and cloud computing. The system is amazing, but one drawback is that the IoT itself may pose a threat to the system. By 2020, it is predicted that there will be about 25 billion connected objects worldwide, making it harder to secure IoT networks.

Given that these systems are frequently used in hostile and unregulated environments, it will become an easy target for hackers. The security threats to telemedicine systems must therefore be analyzed in order to determine what security countermeasures can be taken to lessen the risks and enhance telemedicine's security. In non-emergency situations, telemedicine has been utilized to consult a doctor over the phone for medical advice. Additionally, the patient did not need to meet with the doctor, which reduced costs because the patient did not need to drive to the hospital or wait for follow-up cases for a longer period of time. A Webcam, a secure patient portal connecting the doctor to a database of medical records, and Internet access are the three essential components needed to run telemedicine.

The development of the Internet has significantly altered the practice of medicine. The ability to provide medical care remotely, or telemedicine, is a major change. Doctors and patients no longer need to be in the same physical location for medical treatment to be provided, thanks to telemedicine. Telemedicine has been made possible by a number of technologies, including the use of real-time video and remote sensing equipment. With the help of telemedicine, persons who find it challenging to travel to the hospital, such as those who reside in rural locations, the elderly, or physically challenged people, can now receive medical care.

The advantages of medical facilities have increased thanks to the integration of telemedicine and cloud computing. The precision of collecting patient data for telemedicine and remotely monitoring patients' health has risen with the use of IoT devices such as sensors, smart phones, computers, and home/work appliances. The drawback of employing IoT systems is that because they are utilized in uncontrolled environments, they must be secured to prevent becoming an easy target for hackers.

Bishwajeet Roy; P.W.C. Prasad; Angelika Maag, “A Review on Wireless Telemedicine Technology Challenges and Possible Solution” by IEEE in 2021 6th International Conference on Innovative Technology in Intelligent System and Industrial Applications (CITISIA)

The study provides a comprehensive review of the challenges and potential solutions in the field of wireless telemedicine technology.

The article opens by emphasizing the growing importance of telemedicine as a crucial tool in healthcare delivery. With the growing demand for telemedicine services, there is a greater requirement for effective wireless technology to facilitate data and information transmission.

The authors then go into the numerous obstacles connected with wireless telemedicine technology, such as bandwidth, signal interference, security, and privacy concerns. They also emphasize the difficulties associated with power consumption, pricing, and user approval.

The paper then proposes different potential answers for conquering these difficulties. These incorporate the utilization of cutting-edge pressure procedures, signal handling calculations, and conventions to work on the productivity of remote transmission. The creators likewise talk about the significance of guaranteeing security and protection in remote telemedicine innovation, using encryption, verification, and access control.

The authors additionally propose the utilization of energy-productive remote innovation, like Bluetooth Low Energy (BLE), to resolve the issue of force utilization. They additionally talk about the significance of client acknowledgment and propose the requirement for client-driven plan ways to deal with work on the ease of use and availability of remote telemedicine innovation.

The paper likewise talks about the developing significance of ongoing checking in medical services, which can work on the nature of care and decrease medical care

costs. The creators then, at that point, feature the capability of web-associated gadgets to help constant observing, yet in addition, note the difficulties related to this methodology.

The authors examine different specialized difficulties, including the requirement for a solid and secure network, as well as the requirement for precise and convenient information assortment and examination. They additionally feature the significance of interoperability, as patients might utilize various gadgets from various makers, and these gadgets might utilize different correspondence conventions and information designs.

The paper then talks about the lawful and moral difficulties related to continuous checking. The creators feature the significance of information protection and security, as well as the requirement for clear rules on information proprietorship, access, and use. They additionally talk about the potential for unseen side effects, like the defamation of patients or the making of new well-being imbalances.

All in all, the paper gives important bits of knowledge into the difficulties related with the constant checking of patients utilizing web-associated gadgets. The authors feature the significance of addressing these difficulties to guarantee that ongoing observation can be utilized really and morally to further develop medical care results.

The paper gives significant experiences into the difficulties and potential arrangements in the field of remote telemedicine innovation. The authors feature the requirement for research with innovative work around here to guarantee that telemedicine administrations can be conveyed productively and actually utilizing remote innovation.

Piyush Rathod, Gajanan Bhagat, Prof. Nikita Mohod, “Online OPD Appointment & Hospital Information System” in International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 09 Issue: 03 | Mar 2022

The paper focuses on the development of a web-based system for managing online appointments and hospital information. The framework plans to give a proficient and easy-to-understand way to deal with overseeing arrangements for short-term patients in emergency clinics. The paper begins by talking about the current arrangement frameworks in emergency clinics, which are, in many cases, manual and tedious. The creators feature the requirement for a mechanized framework that can oversee arrangements on the web and give constant data to patients.

The proposed framework is intended to permit patients to book arrangements on the Internet utilizing a web interface. The framework can likewise give data about the accessibility of specialists, their specialization, and the administrations given by the emergency clinic. The framework utilizes a unified data set to store patient data, arrangement plans, and other important information.

The authors likewise examine the advantages of the proposed framework, including its capacity to diminish hanging tight times for patients, limit the responsibility of medical clinic staff, and work on the general proficiency of the clinic. The framework can likewise give information investigation and announcing abilities, which can help clinic overseers to screen the exhibition of the clinic and distinguish regions for development.

The paper additionally depicts the execution of the framework, including the utilization of different advancements like HTML, CSS, PHP, and MySQL. The creators likewise examine the assessment of the framework, which included

testing the framework's usefulness and convenience with a gathering of emergency clinic staff and patients.

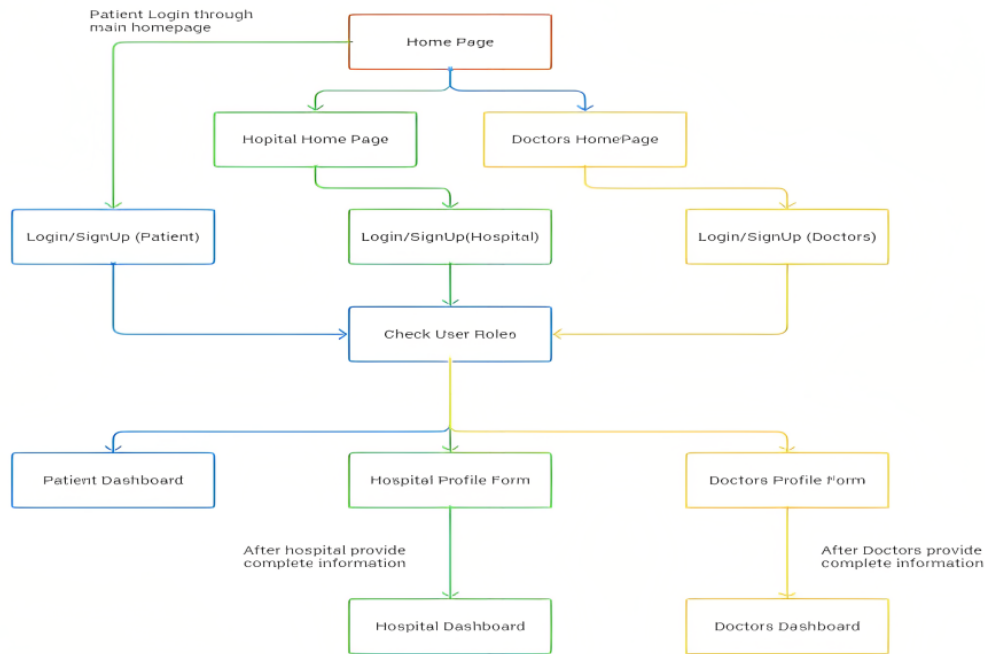


Figure 7: User Flow diagram of discussed paper [5]

The intention is to construct a web application to diminish the manual work of dealing with the specialist, arrangements, tests, and patient data. It tracks every one of the insights regarding the patient, their booking, and medication. Gives the looking through offices based on different factors like a specialist, patient, booking, and medication. The framework additionally deals with the arrangement subtleties & medication subtleties of patients and tracks all data on arrangements, tests, and appointments. It manages to observe the data and exchanges of booking. Altering, adding, and refreshing records is further developed, which brings about legitimate assets: the board of specialists, medical clinic and patient information. All in all, the paper gives significant bits of knowledge into the improvement of a web-based arrangement and emergency clinic data framework.

Table of comparison

Author(s)	Journal year	Published By	Methodology	Disadvantage
R. A. Razali and N. Jamil	2020	ICIMU	Authentication	Security issues in major telemedicine applications
K. S. Gill, S. Saxena and A. Sharma	2019	AICAI	Watermarking	Issues in Watermarking.
P. Matlani and N. D. Londhe	2019	IEEE	IOT	Not able to connect doctors and patients
Piyush Rathod, Gajanan Bhagat	2022	IRJET	Web-app to register hospitals, doctors, and their data.	Additional functionality, improved UI, and performance are possible.

Chapter 03: SYSTEM DEVELOPMENT

Medweb offers virtual connections between patients and medical professionals, allowing patients to receive diagnoses and treatment plans remotely. By submitting a video or text description of their symptoms, patients can schedule interactions with doctors based on availability and their own needs, including the option of secure live video conferencing.

Both patients and doctors can sign up for accounts, with patients having access to a calendar showing past, present, and future appointments, the ability to schedule new appointments, and view their visit history including notes and reasons for past visits. Doctors can also schedule appointments and create patient appointments.

To simplify access to patients' electronic medical records (EMRs), Medweb enables users to easily centralize and upload their EMRs and other necessary documents, allowing for rapid access by physicians without requiring patients to fill out additional forms.

Patients have the ability to register and sign in to the web application, enabling them to: schedule appointments or consult with a doctor online via video. On the other hand, doctors can log in to the web application to accept and view appointments. The system aims to provide users with only the necessary and pertinent information. When using telemedicine, clinical data typically needs to be shared among clinical experts and shown on real testing equipment. Consequently, the server side was developed with the explicit goal of obtaining, storing, and disseminating the vital sign data from patients. Firebase was used for the backend of the application, and React JS was used to develop the front end.

In order to create a platform for a home healthcare service provider, we conducted thorough research into areas that could benefit from digitization. Through this research, we identified various services including nursing care, therapy care, companionship, counseling, domestic help, and doctor's visits.

Our platform design allows for both new and existing users to create a patient form, select the desired service type, city, service provider, and schedule date and time. After confirmation of the booking, the request is sent to the selected service provider's admin for approval. Once approved, the user receives an automated confirmation email with the details of their assigned care provider.

We initially built the basic workflow and then broke down the development of the platform's functionality into stages. We later focused on editing the portal for both the user and admin experiences. Finally, we created operator IDs for each service provider and tested the end-to-end flow with different selections of user types and service providers.

Use Case Diagram

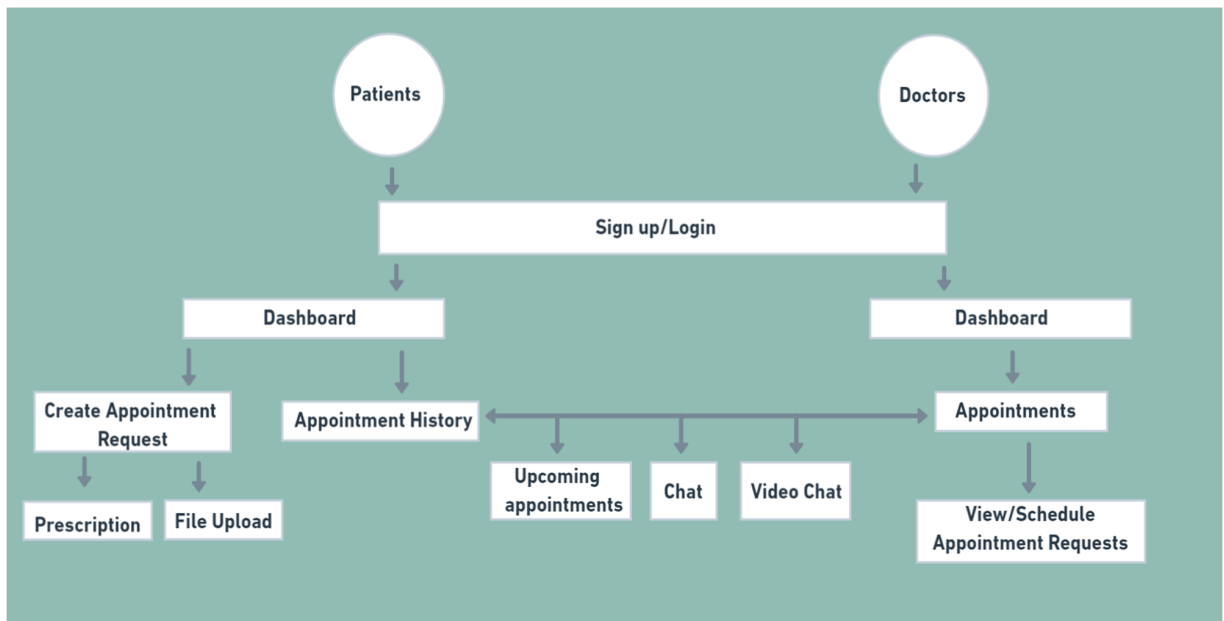


Figure 8: Use case diagram

Use case 1: A Patient

A patient will be given a special user ID, a password when using the application.

After logging in, the patient will be able to:

- View the prescription the doctor sent
- Set up an appointment
- Upload his prescription
- Upload a document stating his illness
- Upload a video of what his illness is
- View a calendar that mentions when his appointment is scheduled

Use case 2: A Doctor

The doctor has to first login or sign up. Afterward, he can do the following:

- Read the patient's information
- Review all the data sent by a single patient
- View appointment statuses and schedule or reschedule an appointment
- Create a google meet link
- View a calendar when his appointments are scheduled

System Requirements:

The telemedicine application has the following system functionalities:

1. Register User

Preconditions: None

Main flow:

- The use case begins when a client demonstrates that he needs to register.
- The form demands a name, email, telephone, and password.
- The user has to enter a name, email, telephone, and password.

- On user click/enter activity, the server checks if this client is already registered or not.
- If yes, an error message is tossed.
- If no, then the client is registered effectively, and the system begins a login session and shows the welcome page.

Alternative flow:

- In the event that the email now exists, the framework shows a message, and the utilization case returns to step 2.
- If the client doesn't enter a required field, a message is shown, and the use case repeats step 3.

2. Login Client

Preconditions: The user is enrolled.

Main flow:

- The utilization case begins when a client shows that he needs to log in.
- The framework demands the email and password.
- The client enters an email and password.
- The framework checks the id, email, and password against every single enrolled client.
- The framework begins a login meeting.
- Alternative Flows
- In the event that the username is invalid, the use case returns to stage 2.
- Assuming the password is invalid the system demands that the user re-enters the password. At the point when the client enters one more password the use case goes on with step 4, utilizing the first username and new secret phrase.

3. Book Appointment

Preconditions: The client should be registered, Signed in.

Main flow:

- The user is presented with two choices for which kind of appointment he wants.
- The user can either Consult Doctors for his disease or Book an appointment
- The user is then redirected to the 'consult doctors' page.
- Users are asked to enter the date and time of the appointment.
- User is asked to input the symptoms he is experiencing
- The user can upload his prescription
- The user can upload a video stating his issues

MedWeb Features

- It's convenient.: Patients no longer need to wait in line or sit in a waiting room to receive a diagnosis. Instead, individuals can connect at home by making an appointment with a doctor using a web application. which in turn will aid in efficiently and swiftly initiating the required treatment.
- Maintaining Medical Records: It makes it much simpler and more comfortable for patients, doctors, and other healthcare professionals to immediately access medical records and histories. It solves the interoperability issue by enabling many clinicians to simultaneously access the same data and instantly retrieve patient prior information.
- Patients can also provide their doctors, family, and friends access to their medical records and histories. It is also better for the environment because

there is no need to retain paper medical records or reports in lockers or drawers.

- **System for the Unified Monitoring and Management of Patients:** It will make it easier for patients and doctors to stay current on prescriptions, make changes as needed, schedule follow-up appointments, monitor chronic diseases, check vital signs, etc.

System Design:

Front-end Development: The front end of the telemedicine application will be developed using ReactJS, a popular JavaScript library for building user interfaces. The front end will have the following pages:

a. Login and Signup Page: This page will allow doctors and patients to create their new accounts and log in to the application.

b. Home Page: This page is the main page of the application, where the doctors and patients can see their profiles, appointments, and prescription history.

c. Video Call Page: This page helps the doctors and patients to talk through video calls. It assists to discuss about the medical conditions and treatments.

d. Appointment Booking Page: This page allows the patients to book appointments with whichever doctor.

e. Prescription Upload Page: This page permits the doctors to upload prescriptions for their patients they are diagnosing.

f. Calendar Page: This page is responsible for the patients and doctors to see their upcoming appointments' date and time.

Back-end Development: The backend of medweb is made using Firestore. Its a NoSQL database provided by Google Cloud. The back-end has the following features:

a. User Authentication: Firestore handles our user authentication for both the doctors as well as patients.

b. Appointment Booking: Firestore is managing the appointment booking system. Here patients can book appointments, and doctors can view and manage that.

c. Prescription Uploading: Firestore is storing the prescriptions uploaded by the doctors and allowing patients to view and download them.

d. Video Call: Firestore is also handling the video call feature. Here doctors and patients can connect to discuss their medical treatments.

e. Security: Firestore provides robust security measures to protect the patients' data. It also ensures secure communication between doctors and patients.

Medweb is designed to be used as an online web portal and a mediator for patients and doctors to create appointments for online doctor visits. The application serves the following purposes:

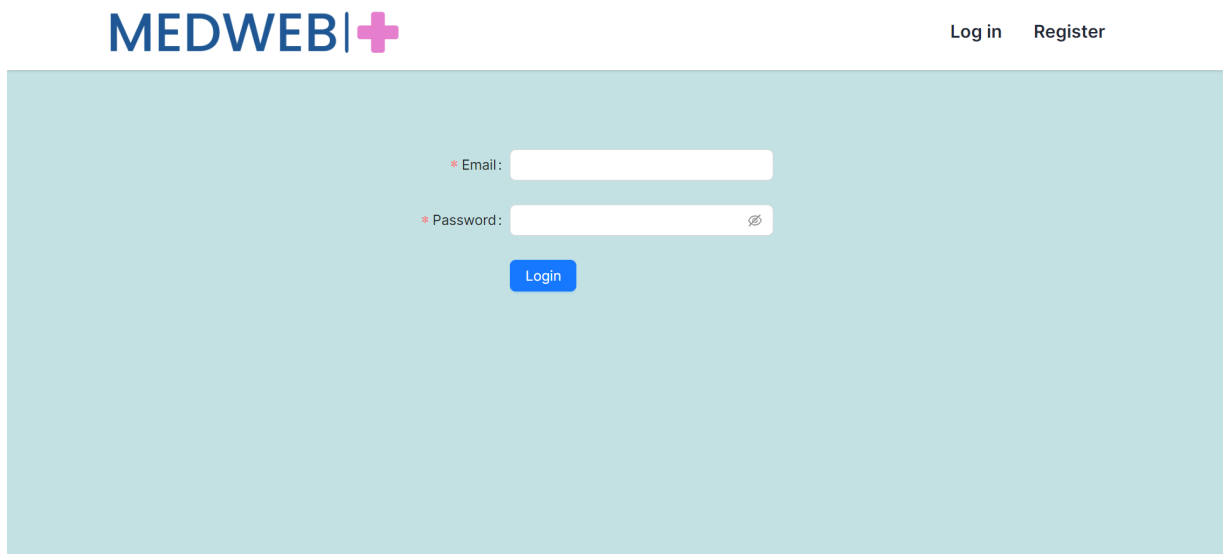
User Authentication

Patients have to make a new account with their email id and password. Doctors have to signup with their emails and password, and then choose the radio button present at the bottom to be a doctor.

No two users can have one same account, and credentials must be validated with Google in order to gain access to the application.

There are separate login and signup pages made in our application.

a. Login Form: The login form has fields for the user's email address and password. Users will enter their login credentials, and the application verifies their information to grant them access to their account made on medweb.



The image shows a screenshot of the MEDWEB login page. At the top left is the MEDWEB logo with a pink plus sign. At the top right are the links 'Log in' and 'Register'. The main content area is a light blue rectangle containing a login form. The form has two input fields: 'Email:' and 'Password:'. The 'Email:' field is a simple white text box. The 'Password:' field is a white text box with a small eye icon on the right side. Below the password field is a blue 'Login' button.

Figure 9: Login Page

a. Signup Form: The signup form has fields for the user's name, email address, password, phone number. Users will enter their information, and the application will create their account.

* First Name:

* Last Name:

* Email:

* Phone Number:

* Password:

* Confirm Password:

* Role: Patient Doctor

[Sign Up](#)

Figure 10: Sign up page(for new registrations)

Landing page

Delivers clinical excellence with compassion and care to the comfort of your home.

For Patients:

- Connect to a medical professional no matter where you are.
- Easily access your electronic medical record (EMR).

For Doctors:

- Guarantees the safety of you and your patient while providing authentic and accurate treatments.
- A way to have a flexible schedule.



Figure 11: Landing Page

Appointments

Both patients and doctors can create meetings with each other. Doctors can only create appointments with patients, and patients can only create appointments with doctors. This is to guarantee our project is being used for the right purposes.

Arrangements can be dropped whenever by either the underlying source or collector. Canceled voids the option for video chat but keeps the option open for text chat.

Appointments are automatically deleted by the server of our Firestore by the next day. Canceled appointments still show after they are canceled and are deleted along with the outdated appointments.

Patients and Doctors can also see their upcoming appointments in a calendar view. This is implemented by us using Calendly.

Upcoming Appointments

Date	Patient	Meeting Link
2023-02-13	ananya mishra	https://meet.google.com/tmi-iccu-ftc

< 1 >

Current Submissions

Patient	Preferred Date	Preferred Time	Video Link	Description	Appointment
fq ef	2023-02-10	06:00:00	https://firebasestorage.googleapis.com/v0/b/telemedicine-b4645.appspot.com/o/6309aab8-ec05-4726-9108-98007654e748?alt=media&token=38fdb4eb-b1b7-4caf-8bef-adf67cf98976	View	Schedule
hrithik	2023-02-	15:00:00	https://firebasestorage.googleapis.com/v0/b/telemedicine-b4645.appspot.com/o/a4f31a3a-f246-47ba-a848-5dfc051fdd8d?	View	Schedule

Next appointment on 2023-02-13 at 18:00:00

[Join Now](#)

New Submission

View History

2023 ▾ Feb ▾ Month Year

Su	Mo	Tu	We	Th	Fr	Sa
29	30	31	01	02	03	04

Figure 12: How to book appointments

Current Submissions						
Patient	Preferred Date	Preferred Time	Video Link	Description	Appointment	
fq ef	2023-02-10	06:00:00	https://firebasestorage.googleapis.com/v0/b/telemedicine-b4645.appspot.com/o/6309aab8-ec05-4726-9108-98007654e748?alt=media&token=38fdb4eb-b1b7-4caf-8bef-adf67cf98976			View Schedule
hrithik raichand	2023-02-13	15:00:00	https://firebasestorage.googleapis.com/v0/b/telemedicine-b4645.appspot.com/o/a4f31a3a-f246-47ba-a848-5dfc051fdd8d?alt=media&token=faaa2a30-244b-4d56-848e-837fab5f13a0			View Schedule
vasundhara pandey	2023-02-10	04:00:00	https://firebasestorage.googleapis.com/v0/b/telemedicine-b4645.appspot.com/o/cb8c99e7-34f2-4e67-93d8-42aec746a5b?alt=media&token=a33f7435-6ff9-41f8-a7a3-ffb265c250fd			View Schedule
ananya mishra	2023-02-13	18:00:00	https://firebasestorage.googleapis.com/v0/b/telemedicine-b4645.appspot.com/o/d1deacd2-ba21-4571-8224-fec411b720cb?alt=media&token=6698e6a4-3599-421e-ad12-8db58f443ac2			View Schedule
dummy user	2023-02-11	20:00:00	https://firebasestorage.googleapis.com/v0/b/telemedicine-b4645.appspot.com/o/e8323dfa-4640-4ef1-b3d7-52c7b2690e4a?alt=media&token=6200570e-81d4-4670-8100-1510057200f			View Schedule

Figure 13: Doctor can view his upcoming appointments

Video Chat

- Video chat is handled by Google Meet, which allows a computer to send a video and audio feed to another computer.

- When an appointment is created, so is a room that only allows the patient and doctor access to that specific room.

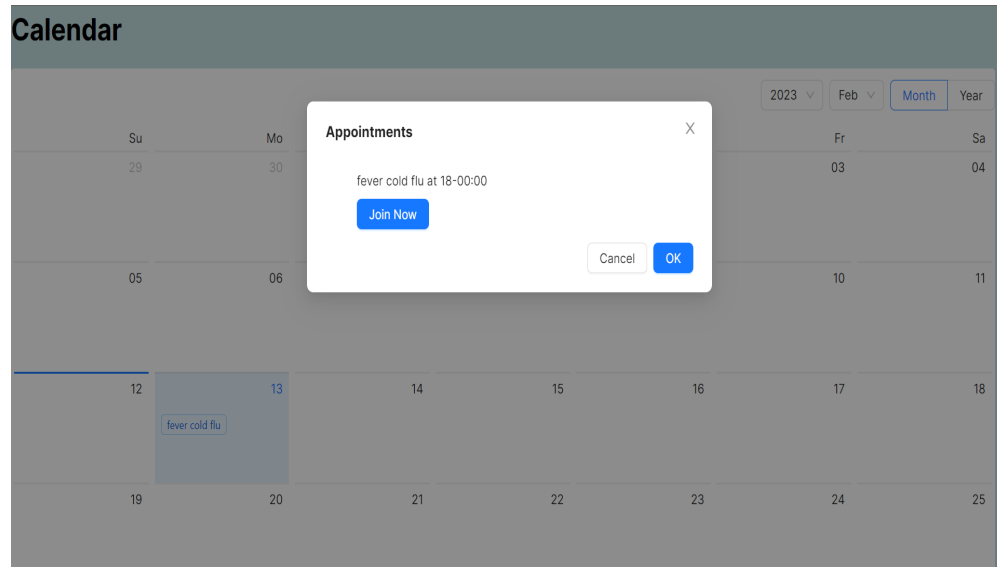



Figure 14: Join Appointment Call

File Uploads

- File Uploads feature allow patients to upload their own files, which can be viewed by Doctors. Doctors also have their own file storage.
- Files can be sent from Doctors to patients' personal file folders for them to view later. Only a patient can see their respective uploads.

New Submission

Enter a short description of your video and symptoms here



Click or drag file to this area to upload

Upload a valid video file with a thorough description of your symptoms, conditions, etc.

test.mp4

Preferred Date Preferred Time

Medical History

Date	Doctor	Notes	Video Link
2023-02-10	dev tyagi	<input type="button" value="View"/>	https://meet.google.com/tmi-iccu-ftc
2023-02-10	dev tyagi	<input type="button" value="View"/>	https://meet.google.com/tmi-iccu-ftc
2023-02-10	dev tyagi	<input type="button" value="View"/>	https://meet.google.com/tmi-iccu-ftc

< 1 >

Figure 15: uploading files

Firestore

- All the data has been stored in Firestore.
- As soon as a new user is registered, his details are added to this database, as shown below.

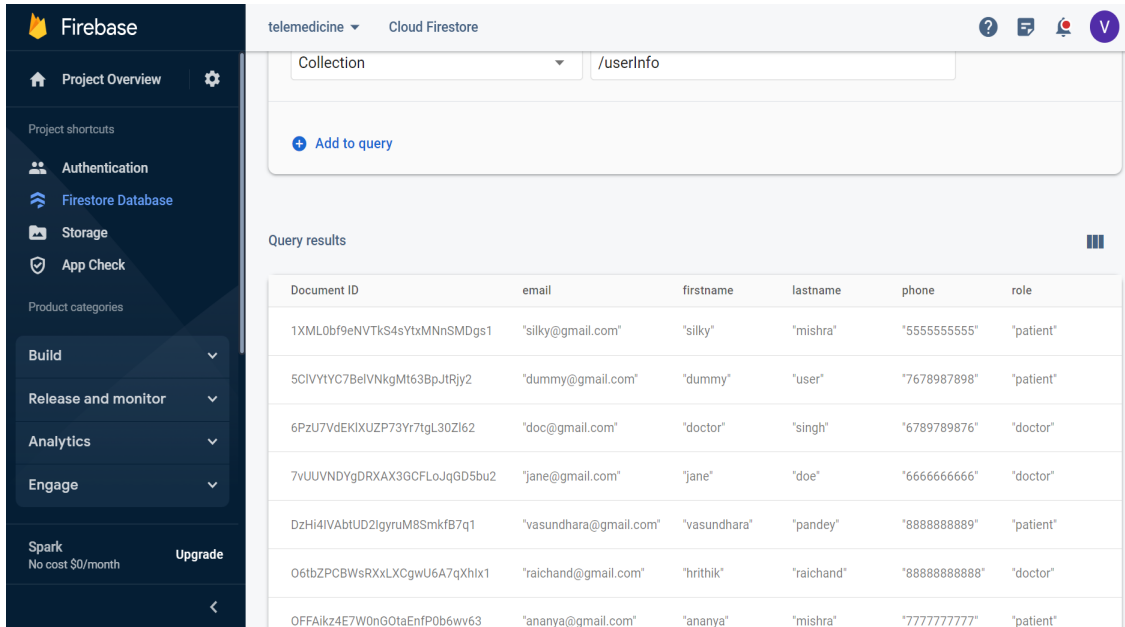


Figure 16: Database in Firestore showing userinfo

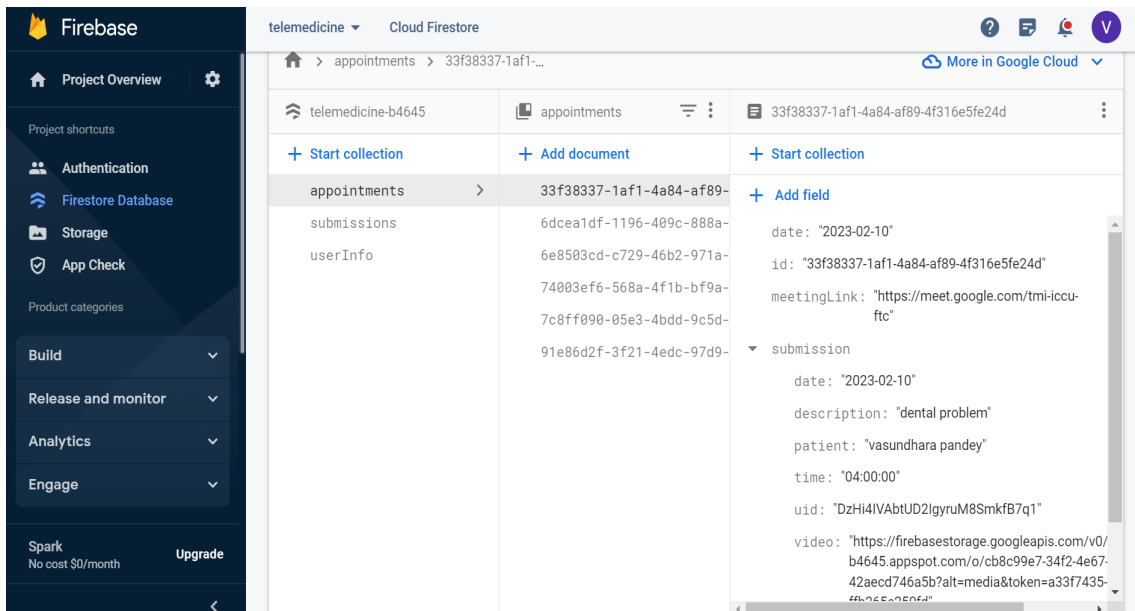


Figure 17: Database in Firestore showing Appointments

Chapter 04: PERFORMANCE ANALYSIS

4.1 Backend Programming - Firebase

The utilization of Firebase in building portable and web applications is normal these days. As per StackShare, around 2216 organizations are utilizing this backend as a help (BaaS) stage. Without a doubt, engineers depend on this stage due to broad instruments, a quick turn of events, and continuous information bases to make intelligent applications.

Crash detailing, Google Examination, secure facilitating, and confirmation are additional advantages of utilizing Firebase that you might want to be aware of.

Firebase is fundamentally a Google-supported application improvement stage which was at first evolved by James Tamplin and Andrew Lee in 2011. It was formally sent off in 2012, and just after two years of send-off, Google obtained this stage. Before all else, Firebase was just planned as a Real-time information base yet after its obtaining by Google, it began giving more administrations.

In basic words, Firebase is a product improvement stage that assists in building web and portable applications with its 18 administrations. These 18 administrations of this BaaS arrangement additionally incorporate intentional APIs and four beta items. Also, it is viable to incorporate Android, web, iOS, and Solidarity arrangements. Tech and business goliaths which are utilizing the cloud and BaaS administrations of Firebase are Alibaba Travels, Stack, Twitch, and Instacart.

Reasons for Firebase being so popular:

Dependable and Broad Information bases

Firebase works under the banner of Google, and that is the reason it gives strong information bases to the web and versatile application improvement. We will talk about significant Firebase data sets with their benefits beneath:

Most likely, the Realtime database was the principal item that didn't lose its appeal till now. Firebase Realtime database permits applications to move toward cross-stage information inside real-time subsequent to joining NoSQL distributed storage. This Real-time data set additionally empowers you to work without web availability. Indeed, even information is as yet being stored in the memory of your gadget when you are disconnected and begin synchronizing after the web network.

Cloud Firestore

One more benefit of Firebase is its Cloud Firestore. This NoSQL data set makes it simple for software engineers to move and store information for front and backend improvement. This cloud data set is likewise known for its Realtime refreshes, adaptable information models, disconnected help, and fast information requests.

All the same, Cloud Firestore likewise gives smooth incorporation of Google Cloud and other Firebase items. With these items, you can get broad information for your application advancement projects. To defeat the security-related worries of software engineers, Cloud Firestore is utilizing Character and Access The executives (IAM) and severe information assurance rules.

Quick and Safe Hosting

One more engaging benefit of Firebase is its solid and quick facilitating administration. Firebase facilitates every substance type, including web applications, and dynamic and static substances. Additionally, possibly you need to have your Express.js microservices, HTML, CSS, or APIs; the facilitating backing of Firebase is dependably there. It implies Firebase has a different assortment of content.

Likewise, zero-setup SSL upgrades the security of content conveyance. To guard your custom area against outer dangers, the utilization of free SSL confirmation of Firebase is likewise useful. Likewise, Firebase CLI helps software engineers to make their applications live and run in no time. SSD and CDN support make the substance conveyance exceptionally quick when you use Firebase.

The review of content and URLs is likewise one more extraordinary advantage of utilizing Firebase facilitating. However these URLs last just a brief time, yet you actually have the opportunity to impart it to partners to get their perspectives prior to going live.

4.2 Calendly

Calendly is a cloud-based planning program that empowers clients to plan gatherings, arrangements, and occasions rapidly and without any problem. The product is intended to wipe out the requirement for to and fro messages, calls, and messages to carve out a reasonable opportunity to meet. With Calendly, clients can share their accessibility and permit others to choose a period that works for the two players, simplifying booking and proficiency.

Calendly was established in 2013 by Tope Awotona, fully intent on working on the booking system. Today, the product is utilized by a huge number of individuals around the world, from specialists and entrepreneurs to enormous companies.

One of the primary elements of Calendly is its adaptability. Clients can alter their accessibility in light of their timetable, including setting normal working hours, time regions, and shutting out unambiguous times for individual occasions or arrangements. This adaptability guarantees that clients can keep up with their balance between serious and fun activities while being accessible for gatherings and arrangements.

One more key element of Calendly is its incorporation with well-known schedule applications, for example, Google Schedule, Office 365, and Standpoint. This mix guarantees that clients can, without much of a stretch, sync their timetables, view their arrangements, and get warnings for impending occasions.

Calendly likewise offers a scope of adaptable gathering types, remembering one-for-one gatherings, bunch gatherings, and online classes. Clients can add custom inquiries to their booking structure, permitting them to gather significant data from members, for example, their name, email address, and telephone number. They can likewise send programmed updates and follow-up messages to members, guaranteeing that everybody is in total agreement and that gatherings run as expected. One of the champion elements of Calendly is its straightforwardness. The product is not difficult to use, with a perfect and instinctive connection point that empowers clients to explore the stage rapidly and proficiently. The product likewise offers a scope of instructional exercise recordings and backing documentation, making it simple for clients to get everything rolling and investigate any issues they might experience.

Calendly's estimating plans are likewise intended to suit the necessities of many clients, from people to huge groups. The product offers a free arrangement that incorporates essential planning highlights, as well as superior plans that offer extra elements, for example, group booking, custom marking, and examination.

By and large, Calendly is a strong planning instrument that improves on the booking system and assists clients with saving investment. Its adaptability, mix with famous schedule applications, and adaptable gathering types make it a flexible device for people, groups, and organizations, everything being equal. Its straightforwardness and usability make it an astounding decision for anybody hoping to smooth out their planning cycle and work on their efficiency.

Notwithstanding the elements referenced above, Calendly likewise offers a scope of cutting-edge includes that can assist clients with enhancing their booking interaction. One such element is the capacity to make custom occasion types, permitting clients to fit their booking choices to their particular necessities. Clients can likewise set up cradle times between occasions, guaranteeing that they have sufficient opportunity to get ready for their next gathering or arrangement.

Calendly likewise offers strong incorporations with a scope of outsider applications, like Salesforce, Zapier, and Slack. These reconciliations permit clients to robotize undertakings, for example, sending follow-up messages or making scheduled occasions, and smooth out their work process.

For bigger groups, Calendly offers a scope of coordinated effort highlights, for example, group booking and client consent. This permits groups to cooperate flawlessly, guaranteeing that everybody is in total agreement and that planning clashes are limited.

One of the greatest advantages of utilizing Calendly is the efficient part of the product. Via mechanizing the planning system, clients can save critical measures of the significant investment that would somehow or another be spent on to and fro messages and messages. This expanded proficiency can prompt expanded efficiency and can assist clients with zeroing in on additional significant undertakings.

One more advantage of utilizing Calendly is the capacity to further develop client commitment. By making it simple for clients to plan gatherings and arrangements, organizations can further develop their client experience and increment consumer loyalty. Moreover, Calendly's customization choices permit organizations to fit their booking interaction to their particular image and client base, guaranteeing that their planning cycle lines up with their general image technique.

All in all, Calendly is a strong planning device that offers a scope of highlights intended to work on the booking system and save clients' investments. Its adaptability, mix with well-known schedule applications, and adjustable gathering types make it a flexible instrument for people, groups, and organizations, all things considered. Its high-level elements, cooperation choices, and strong incorporations pursue a magnificent decision for organizations hoping to advance their planning interaction and work on their efficiency.

Microservices

Because of the shortcomings of the solid example of programming improvement, microservices became essential. Every product application highlighted in a microservice is separated from the others, typically speaking with their own servers and data sets. Applications created using this type of engineering are roughly connected, also known as appropriated applications.

Assume we're building a web-based company store. Models will be needed for an installation highlight, truck, clients, administrator, and request. Each of these features will have its own set of servers and databases.

Our web-based business microservices will communicate via the REST Programming interface framework. Because we will separate our store includes independently from one another, if our framework has a flaw, we can easily discern which component to debug and avoid having to shut down the entire program.

Microservices-based applications are more adaptable than traditional applications. You may use any programming language to create a microservice; in fact, you can use several dialects to provide different features in a microservice application.

In general, microservices provide a better designer experience. Another designer joining the team will not need to understand the whole codebase, only the features they are working on, increasing overall efficiency.

Finally, microservices offer unit testing; you can write a unit test to evaluate a specific usefulness.

It's important to note that designing a microservice demands competence because reconciliation and end-to-end testing might be difficult. Furthermore, microservices can be extremely inconvenient, resulting in enormous support expenses.

Finally, it is not always straightforward to move code written with good engineering to a microservice, and it may be difficult for apps to discover one another inside a mind-boggling network.

Communication between microservices

Choosing a microservice compositional example comes with a few challenges, one of which is administered to support correspondence. Administrations are inextricably linked components of an application that contribute to the overall display of the application.

To achieve compelling execution, the microservices must communicate with one another. Correspondence is made possible in a microservice application by an administrative correspondence convention like HTTP(s), gRPC, or message representatives.

A part of the ways administrators organize correspondence in microservice engineering.

HTTP communication

HTTP correspondence is a type of concurrent communication design in which one service relies on another to perform:



Figure 18: HTTP Request-Response Cycle [6]

The image above represents the HTTP request-response cycle, where the client makes a request and waits for a response from the server-side application.

Event-driven communication pattern

The event-driven correspondence design entails a conversation between a professional organization and a help buyer. The assistance purchaser requires an asset from a third party. At that time, it does computations and sends the result to the client:

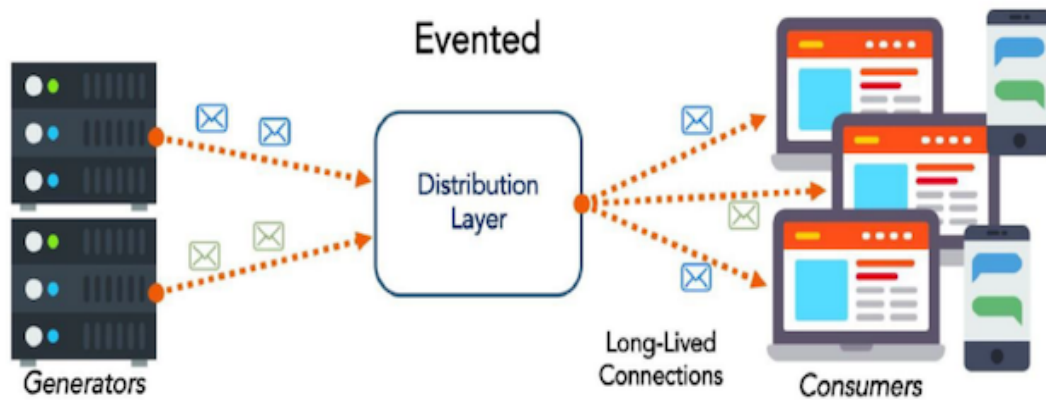


Figure 19: Communication pattern between a service provider and a service consumer. [6]

4.3 Frontend Web Development Framework - React.js

React.js is a famous front-end web improvement system created by Facebook. It is an open-source JavaScript library utilized for building UIs and dealing with the condition of web applications. React.js utilizes a part-based design, where the UI is separated into little, reusable parts that can be effortlessly overseen and refreshed.

React.js permits engineers to make complex UIs by composing basic, revelatory code. It utilizes a virtual DOM (Report Item Model) to effectively refresh the UI, making it quicker and more responsive than conventional DOM control methods. React.js likewise gives various helpful elements, for example, props and state, which permit parts to convey and interface with one another.

One of the critical advantages of React.js is its adaptability. It may very well be utilized to construct an extensive variety of web applications, from straightforward single-page applications (SPAs) to complex venture-level applications. React.js can likewise be handily coordinated with different libraries and structures, for example, Revival and Respond Local.

In short, React.js is a strong frontend web improvement system that empowers designers to fabricate complicated and responsive UIs utilizing part-based engineering. Its adaptability and usability pursue it, a famous decision for building web utilizations of all sizes and intricacy levels.

Miguel Ramos, Marco Tulio Valente, and Ricardo Land of Brazil conducted a survey to determine what practicing ReactJS (note: not Precise) software developers feel are the framework's basic presentation difficulties.

While a review does not provide concrete data like a benchmark, it does provide a few intriguing indicators of what the difficulties are, and such data is scarce:

While React.js has many advantages, it additionally has a few normal issues that designers might confront. Here is a portion of the issues that engineers might experience while working with React.js:

Steep expectation to learn and adapt: React.js has an intricate engineering that might require some investment to see completely. The expectation to learn and adapt can be steep for fledglings, particularly individuals who are new to JavaScript and frontend advancement.

State management: React.js depends vigorously on state management, which can become muddled as applications fill in size and intricacy. Managing state can likewise be troublesome while managing offbeat occasions, for example, Programming interface calls.

Execution issues: React.js can have execution issues while managing huge informational collections or complex UIs. The virtual DOM can turn out to be delayed to refresh, prompting slack and unfortunate client experience.

JSX intricacy: JSX is a grammar expansion utilized by React.js that can be challenging to peruse and comprehend, particularly for engineers who are used to conventional HTML and JavaScript sentence structure.

Troubleshooting: Debugging React.js applications can be troublesome, particularly while working with settled parts or complex state of the board.

All in all, there are many motivations behind why engineers might decide to work with Respond JS over other front-end systems. Responds adaptability, local area

support, execution, simplicity of coordination, and incredible assets and elements pursue it a magnificent decision for building complex web applications. Whether you are a fledgling or an accomplished designer, Respond JS is a structure that can assist you with building top-caliber, viable code and working on your general efficiency.

4.4 Twilio

Twilio is a cloud correspondence stage that permits organizations and designers to incorporate correspondence capacities into their applications without any problem. With Twilio, engineers can add voice, video, and informative usefulness to their web and portable applications, empowering them to speak with clients and clients in new and creative ways.

One of the vital advantages of utilizing Twilio is its usability. The stage offers a scope of APIs and SDKs that make it simple for engineers to incorporate correspondence capacities into their applications. These APIs and SDKs permit engineers to fabricate custom correspondence arrangements that are customized to their particular necessities while likewise lessening how much time and assets are expected to foster these arrangements.

One more benefit of utilizing Twilio is its adaptability. The stage offers a scope of correspondence channels, including SMS, voice, video, and talk, making it simple for organizations to speak with their clients in the manner that best suits their necessities. This adaptability likewise permits organizations to scale their correspondence capacities as their necessities advance over the long haul.

Twilio likewise offers a scope of cutting-edge includes that can assist organizations with improving their correspondence processes. For instance, Twilio's astute steering capacities permit organizations to naturally course

approach calls and messages to the most suitable specialist or division, lessening stand-by times and further developing the general client experience.

One more benefit of utilizing Twilio is its dependability and versatility. The stage is based on a profoundly versatile foundation that can deal with a huge number of calls and messages each day, guaranteeing that organizations can undoubtedly scale their correspondence capacities as their client base develops. Furthermore, Twilio offers work in overt repetitiveness and failover capacities, guaranteeing that organizations can keep on speaking with their clients even in case of a blackout or disturbance.

Notwithstanding its center correspondence capacities, Twilio likewise offers a scope of extra administrations and devices that can assist organizations with streamlining their correspondence processes. For instance, Twilio's Flex stage gives an adaptable contact community arrangement that permits organizations to construct a custom contact place that meets their particular necessities.

Twilio likewise offers a scope of engineer instruments and assets, including documentation, instructional exercises, and backing gatherings, that make it simple for designers to begin with the stage and construct custom correspondence arrangements.

All in all, Twilio is a strong cloud correspondence stage that offers a scope of correspondence channels and high-level elements that can assist organizations with upgrading their correspondence processes. Its usability, adaptability, unwavering quality, and versatility go with it a great decision for organizations hoping to incorporate correspondence abilities into their applications. Also, Twilio's scope of extra administrations and designer instruments make it simple for organizations to construct custom correspondence arrangements that meet their particular necessities.

Some use cases for Twilio include:

Customer care: Twilio can be utilized to construct a custom contact community that permits organizations to give great client care by means of voice, SMS, and talk. The stage's astute directing capacities can assist organizations with diminishing stand-by times and further develop the general client experience.

Marketing campaigns: Twilio can be utilized to send designated SMS messages, and voice calls to clients as a feature of a showcasing effort. This can assist organizations with further developing client commitment and driving deals.

Reminder for updates: Twilio can be utilized to send robotized SMS and voice suggestions to clients before planned arrangements. This can assist with decreasing flake-outs and work on the productivity of arrangement booking processes.

Two-factor verification: Twilio can be utilized to add an extra layer of safety to applications by giving two-factor confirmation by means of SMS or voice. This can assist organizations with safeguarding client accounts and forestall unapproved access.

IoT correspondence: Twilio can be utilized to empower correspondence between IoT gadgets and backend frameworks. This can help organizations screen and control IoT gadgets from a distance and mechanize processes.

In general, Twilio is a strong and adaptable cloud correspondence stage that offers a scope of correspondence channels and high-level elements that can assist organizations with enhancing their correspondence processes. Its convenience,

dependability, and adaptability go with it an amazing decision for organizations of all sizes, while its scope of extra administrations and designer instruments make it simple to construct custom correspondence arrangements that address explicit issues.

4.6 Google Meet

We integrated our website with Google meet. Integrating Google Meet with a telemedicine website can provide numerous advantages for healthcare providers and patients. These include:

Enhanced healthcare access: Telemedicine enables patients to receive medical care from their homes or workplaces, eliminating the need for travel and reducing wait times. This is especially useful for patients with mobility or transportation problems, those in remote or rural areas, or those who live far from their healthcare provider.

Improved convenience: The integration of Google Meet with a telemedicine website can simplify the scheduling and conducting of virtual appointments. Patients can join video conferences from their smartphones, tablets, or computers, eliminating the need for in-person visits.

Cost savings: Virtual consultations through telemedicine are usually less expensive than in-person appointments. Integrating Google Meet with a telemedicine website removes the need for costly video conferencing software, hardware, and IT support.

Better patient outcomes: Telemedicine consultations can boost patient outcomes by providing consistent care, reducing hospital readmissions, and improving patient satisfaction.

Security: Google Meet's secure video conferencing platform offers a safe communication channel for healthcare providers and patients. The integration can adhere to HIPAA regulations, safeguarding patient information.

Increased efficiency: Integrating Google Meet with a telemedicine website can save time and improve the efficiency of healthcare providers. Virtual consultations can be scheduled and conducted swiftly, reducing wait times and enabling providers to attend to more patients in a shorter period.

Overall, integrating a telemedicine website with Google Meet can enhance healthcare access, convenience, cost savings, patient outcomes, security, and efficiency for healthcare providers.

Chapter 05: CONCLUSIONS

The rise of the medical services business has had a vital impact, driving excellent development, owing mostly to digitization and telecare of therapy and discussion.

The buyer is content, the expert organization benefits and all parties are happy.

The relevance of Telehealth innovations stems from the fact that they may have a positive impact on patients' medical services lives by simplifying Time components, comfort, and accessibility in remote areas. It now ensures favorable medical treatments on the patient's time and accommodation, reducing travel time, and exploring deeper into provincial regions.

The greater impact is on decreasing the cost of care for the poor and improving access to experts for patients, which was previously confined to major cities. Video consultation via mobile applications, as well as certain other tech tools like sensors and monitors, are currently improving healthcare quality throughout the world, providing life-saving measures and preventative healthcare for everybody. When compared to in-house client-server systems, cloud computing technologies have played a big part in simplifying digital/virtual healthcare by driving economic, operational, and functional advantages in the sector. Cloud computing not only reduces expenses, but it also adds much-needed scalability and flexibility to operations.

In conclusion, telemedicine has become an important tool in healthcare delivery, particularly during the COVID-19 pandemic. Telemedicine applications have the potential to improve access to healthcare services, reduce healthcare costs, and provide personalized care to patients. However, there are also limitations to telemedicine, such as technical issues, lack of physical examination, and privacy concerns.

The future work of telemedicine applications is vast, and it includes expanding telemedicine services, integrating with EHRs and AI, remote monitoring, wearable technology integration, and public health applications. As technology continues to advance, it is likely that telemedicine will become an increasingly important part of healthcare delivery.

Overall, telemedicine has the potential to revolutionize healthcare delivery by improving patient outcomes, reducing healthcare costs, and increasing access to care. However, it is important for healthcare providers and patients to be aware of the limitations of telemedicine and use it appropriately to ensure that patients receive the highest quality of care possible.

Furthermore, it improves data security and privacy, which is an urgent necessity and a growing concern in society. These advantages will have a far-reaching influence on the healthcare business, boosting its growth and saving citizens' lives.

This cloud-based telemedicine application technology is still in its early stages and is developing by the day and hour. Healthcare professionals and patients will be prepared with more knowledge and informed on how Telemedicine may transform their lifestyle, thanks to more sophisticated research on the dangers and advantages.

To summarize, numerous technologies may be tested to provide patients more control over their healthcare, but current research requires far deeper examination and research to investigate all of the potentials that developing technology might offer society at large.

The project's purpose was to not only delve into the new technologies being developed on a daily basis but also to explore ways to shift patients' mindsets to a different type of healthcare backed by the power of technology.

5.1 Application of the Major Project

There are several practical applications that can be used, some of which are as mentioned below:

1. **Digital Connect:** This App will entice new tech-savvy customer groups and digitally connect healthcare with customers.
2. **Monitoring healthcare:** An Eagle's Eye View for Medical Organizations to monitor each and every action taken.
3. **Patient Records and Monitoring:** Doctors can see patients' history of medication and test reports, thereby reducing time and cost with effective treatment.
4. **Ease and Convenience:** Ease for doctors to Monitor daily data and improvements of a chronic patient.
5. **Treatment Cost:** No Extra costs of keeping a patient on observation in hospitals.
6. **Customisation based on demographics:** Say custom diet based on Location, Age, Sex, and other parameters.
7. **Authenticity:** Prescriptions will be authentic and be provided at our platform only so that people can purchase medicines from their locality itself.

5.2 Limitations of the Major Project

1. The login method is the only security measure provided in this project at the moment. It necessitates more security checks and controls. Only users with a valid username and password may now access the application's private features.
2. It does not check if the patient's numbers or parameters are within the given range. When a patient provides information, for example, there should be a mechanism in place to guarantee that the values the patient enters are correct.
3. Performance: It is likely to suffer from greater latency, which reduces and creates performance bottlenecks while accessing data.
4. Data security and governance: The greater threat is to data privacy and security, as well as exposing it to hackers. When some sorts of data wind up in cloud platforms, compliance difficulties may arise.
5. Poor storage management and inefficient database setup.
6. We cannot check the authenticity of the doctor.
7. Telemedicine has certain limitations that need to be considered. For example, it relies on technology such as high-speed internet, video conferencing software, and medical devices, which can be unreliable and lead to communication difficulties and poor quality of care.
8. Additionally, telemedicine consultations may not allow for thorough physical examinations, which can result in missed diagnoses and inaccurate

assessments, while the lack of personal interaction may make it difficult to build rapport between patients and healthcare providers.

9. Privacy concerns are also present as personal health information needs to be shared over the Internet, and legal and regulatory challenges may limit the availability of telemedicine services in certain areas or for certain patients due to differing laws and regulations across states or countries.
10. Telemedicine applications may present barriers to access for certain patients and healthcare providers who require technical expertise to use them.
11. Insurance coverage for medweb or likewise apps be restricted or inaccessible, especially for patients dwelling in remote or underserved regions.
12. The absence of actual presence during telemedicine discussions can at times make it provoking for medical services suppliers to make precise evaluations of a patient's condition, especially when particular gear is required.
13. Telemedicine does not always provide access to best equipment like radiology or lab equipment, which can be important for accurate diagnoses and treatments.
14. Telemedicine is also not fine for treating emergencies, and patients should seek immediate medical attention in person if such situations arise.
15. Although telemedicine has been essential during the COVID-19, patients and healthcare people should be aware of its restrictions and use it accordingly to ensure the quality of health.

5.2 Future work in the field of Telemedicine

In the future, telemedicine will possibly change medical care conveyance by giving more prominent admittance to medical care administrations and diminishing medical care costs. Future work includes:

Expansion of Telemedicine Services: Telemedicine administrations can be extended to incorporate a more extensive scope of claims to fame and administrations like psychological wellness, restoration, and preventive consideration.

Integration with Electronic Health Records (EHRs): Telemedicine applications can be coordinated with EHRs to furnish medical care suppliers with admittance to patient information during telemedicine conferences. This could work on the nature of care and diminish blunders in treatment.

Artificial Intelligence (AI) Integration: Telemedicine applications can be coordinated with AI to improve finding and treatment suggestions. AI could examine patient information to give altered treatment designs and robotize regulatory undertakings.

Remote Monitoring: Telemedicine applications can be utilized to screen patients with constant circumstances from a distance. This could empower medical services suppliers to screen patients' circumstances if necessary.

Wearable Technology Integration: Telemedicine applications can be integrated with latest technology like smartwatches and fitness trackers to collect the info about patient in real-time. This data could be utilized to provide customized treatment plans and improve health outcomes.

Telemedicine for Public Health: Telemedicine could be used for public health purposes like disease monitoring and outbreak controlling. Telemedicine could also be utilized to provide education and information to the public on public health issues.

Overall, the potential benefits of telemedicine to patients and healthcare providers are numerous, and as technology continues to advance, telemedicine will likely become an increasingly important aspect of healthcare delivery.

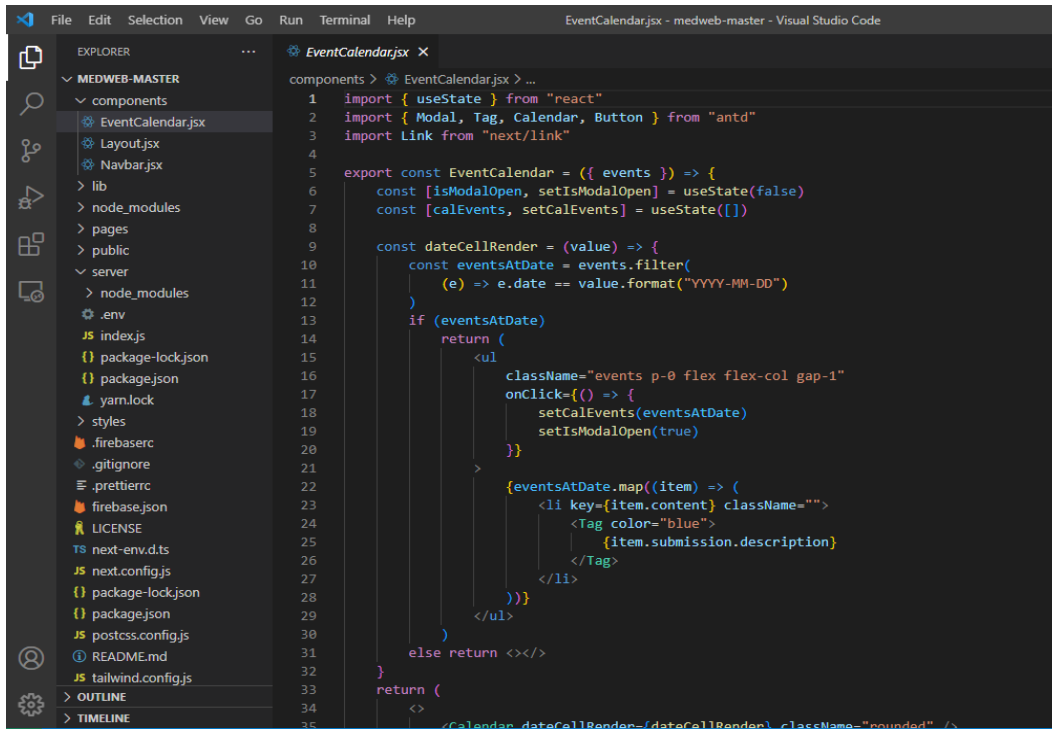
5.3 Future work of Major Project

1. Launching this application for the benefit of mankind and the larger society, especially those in rural and inaccessible locations.
2. Building a mobile application to expand the user base and be able to provide the service on all the platforms available.
3. Collaborating with different Medical Organizations and NGOs to drive better healthcare
4. Enhancing the Security of the Storage Servers - encryption, double layer security, etc.
5. Improving all in all interaction and verification of the specialists on-board.

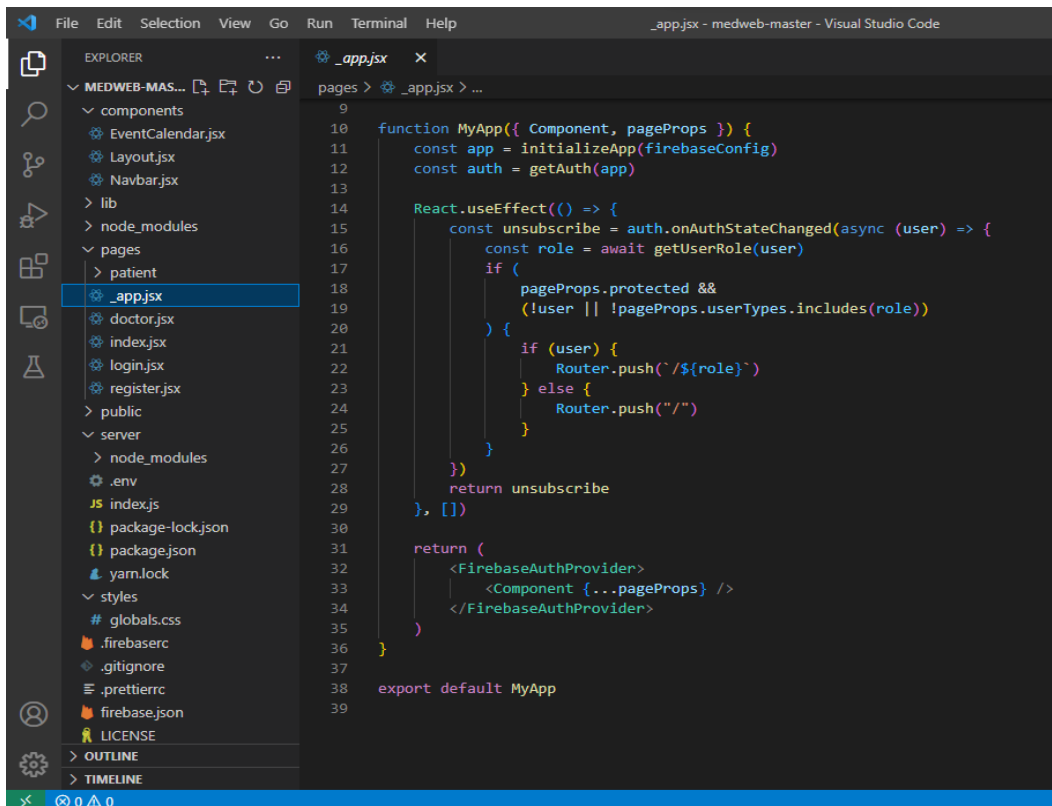
REFERENCES

- [1] P. Matlani and N. D. Londhe, "A cloud computing based telemedicine service," 2019 IEEE Point-of-Care Healthcare Technologies (PHT), 2019, pp. 326-330, doi: 10.1109/PHT.2013.6461351.
- [2] Z. Jin and Y. Chen, "Telemedicine in the Cloud Era: Prospects and Challenges," in IEEE Pervasive Computing, vol. 14, no. 1, pp. 54-61, Jan.-Mar. 2015, doi: 10.1109/MPRV.2015.19.
- [3] K. S. Gill, S. Saxena and A. Sharma, "Taxonomy of Security Attacks on Cloud Environment: A Case Study on Telemedicine," 2019 Amity International Conference on Artificial Intelligence (AICAI), 2019, pp. 454-460, doi: 10.1109/AICAI.2019.8701363.
- [4] Piyush Rathod, Gajanan Bhagat, Prof. Nikita Mohod, "Online OPD Appointment & Hospital Information System" in International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 09 Issue: 03 | Mar 2022
- [6] Á. Garai and I. Péntek, "Adaptive services with cloud architecture for telemedicine," 2015 6th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2015, pp. 369-374, doi: 10.1109/CogInfoCom.2015.7390621.
- [7] C. O. Rolim, F. L. Koch, C. B. Westphall, J. Werner, A. Fracalossi and G. S. Salvador, "A Cloud Computing Solution for Patient's Data Collection in Health Care Institutions," 2010 Second International Conference on eHealth, Telemedicine, and Social Medicine, 2010, pp. 95-99, doi: 10.1109/eTELEMED.2010.19.

APPENDICES



```
1 import { useState } from "react"
2 import { Modal, Tag, Calendar, Button } from "antd"
3 import Link from "next/link"
4
5 export const EventCalendar = ({ events }) => {
6   const [isModalOpen, setIsModalOpen] = useState(false)
7   const [calEvents, setCalEvents] = useState([])
8
9   const dateCellRender = (value) => {
10     const eventsAtDate = events.filter(
11       (e) => e.date == value.format("YYYY-MM-DD")
12     )
13     if (eventsAtDate)
14       return (
15         <ul
16           className="events p-0 flex flex-col gap-1"
17           onClick={() => {
18             setCalEvents(eventsAtDate)
19             setIsModalOpen(true)
20           }}
21         >
22           {eventsAtDate.map((item) => (
23             <li key={item.content} className="">
24               <Tag color="blue">
25                 {item.submission.description}
26               </Tag>
27             </li>
28           ))}
29         </ul>
30       )
31     else return <</>
32   }
33   return (
34     <<
35     <Calendar dataCellRender={dateCellRender} className="rounded" />
```



```
9
10 function MyApp({ Component, pageProps }) {
11   const app = initializeApp(firebaseConfig)
12   const auth = getAuth(app)
13
14   React.useEffect(() => {
15     const unsubscribe = auth.onAuthStateChanged(async (user) => {
16       const role = await getUserRole(user)
17       if (
18         pageProps.protected &&
19         (!user || !pageProps.userTypes.includes(role))
20       ) {
21         if (user) {
22           Router.push(`/${role}`)
23         } else {
24           Router.push("/")
25         }
26       }
27     })
28     return unsubscribe
29   }, [])
30
31   return (
32     <FirebaseAuthProvider>
33       <Component {...pageProps} />
34     </FirebaseAuthProvider>
35   )
36 }
37
38 export default MyApp
39
```

```

Navbarjsx - medweb-master - Visual Studio Code
EXPLORER
MEDWEB-MAS...
components > Navbarjsx > ...
  24 return (
  25   <div className="bg-white shadow">
  26     <div className="container mx-auto px-4">
  27       <div className="flex items-center justify-between py-4">
  28         <div>
  29           <Link href={`/${role}`}>
  30             
  36           </Link>
  37         </div>
  38
  39         {user !== null ? (
  40           <div className="hidden sm:flex sm:items-center">
  41             <a className="text-gray-800 text-lg font-semibold border px-4 py-2">
  42               <Button variant = "contained"
  43                 onClick={() => {
  44                   auth.signOut()
  45                     .then(function() {
  46                       Router.push("/login")
  47                     })
  48                   .catch(function(error) {
  49                     console.log(error)
  50                   })
  51                 }}
  52             >
  53               Log Out
  54             </Button>
  55           </a>
  56         </div>
  57       ) : (
  58         <div className="hidden sm:flex sm:items-center">
  59           <Link href="/login" className="text-gray-800 text-lg font-semibold border px-4 py-2">
  60             Log In
  61           </Link>
  62         </div>
  63     )
  64   </div>
  65 )
  66
  67 ) : (
  68   <div className="bg-white shadow">
  69     <div className="container mx-auto px-4">
  70       <div className="flex items-center justify-between py-4">
  71         <div>
  72           <Link href={`/${role}`}>
  73             
  79           </Link>
  80         </div>
  81
  82         {user !== null ? (
  83           <div className="hidden sm:flex sm:items-center">
  84             <a className="text-gray-800 text-lg font-semibold border px-4 py-2">
  85               <Button variant = "contained"
  86                 onClick={() => {
  87                   auth.signOut()
  88                     .then(function() {
  89                       Router.push("/login")
  90                     })
  91                   .catch(function(error) {
  92                     console.log(error)
  93                   })
  94                 }}
  95             >
  96               Log Out
  97             </Button>
  98           </a>
  99         </div>
  100       ) : (
  101         <div className="hidden sm:flex sm:items-center">
  102           <Link href="/login" className="text-gray-800 text-lg font-semibold border px-4 py-2">
  103             Log In
  104           </Link>
  105         </div>
  106     )
  107   </div>
  108 )
  109 )
  110 )
  111 )
  112 )
  113 )
  114 )
  115 )
  116 )
  117 )
  118 )
  119 )
  120 )
  121 )
  122 )
  123 )
  124 )
  125 )
  126 )
  127 )
  128 )
  129 )
  130 )
  131 )
  132 )
  133 )
  134 )
  135 )
  136 )
  137 )
  138 )
  139 )
  140 )
  141 )
  142 )
  143 )
  144 )
  145 )
  146 )
  147 )
  148 )
  149 )
  150 )
  151 )
  152 )
  153 )
  154 )
  155 )
  156 )
  157 )
  158 )
  159 )
  160 )
  161 )
  162 )
  163 )
  164 )
  165 )
  166 )
  167 )
  168 )
  169 )
  170 )
  171 )
  172 )
  173 )
  174 )
  175 )
  176 )
  177 )
  178 )
  179 )
  180 )
  181 )
  182 )
  183 )
  184 )
  185 )
  186 )
  187 )
  188 )
  189 )
  190 )
  191 )
  192 )
  193 )
  194 )
  195 )
  196 )
  197 )
  198 )
  199 )
  200 )
  201 )
  202 )
  203 )
  204 )
  205 )
  206 )
  207 )
  208 )
  209 )
  210 )
  211 )
  212 )
  213 )
  214 )
  215 )
  216 )
  217 )
  218 )
  219 )
  220 )
  221 )
  222 )
  223 )
  224 )
  225 )
  226 )
  227 )
  228 )
  229 )
  230 )
  231 )
  232 )
  233 )
  234 )
  235 )
  236 )
  237 )
  238 )
  239 )
  240 )
  241 )
  242 )
  243 )
  244 )
  245 )
  246 )
  247 )
  248 )
  249 )
  250 )
  251 )
  252 )
  253 )
  254 )
  255 )
  256 )
  257 )
  258 )
  259 )
  260 )
  261 )
  262 )
  263 )
  264 )
  265 )
  266 )
  267 )
  268 )
  269 )
  270 )
  271 )
  272 )
  273 )
  274 )
  275 )
  276 )
  277 )
  278 )
  279 )
  280 )
  281 )
  282 )
  283 )
  284 )
  285 )
  286 )
  287 )
  288 )
  289 )
  290 )
  291 )
  292 )
  293 )
  294 )
  295 )
  296 )
  297 )
  298 )
  299 )
  300 )
  301 )
  302 )
  303 )
  304 )
  305 )
  306 )
  307 )
  308 )
  309 )
  310 )
  311 )
  312 )
  313 )
  314 )
  315 )
  316 )
  317 )
  318 )
  319 )
  320 )
  321 )
  322 )
  323 )
  324 )
  325 )
  326 )
  327 )
  328 )
  329 )
  330 )
  331 )
  332 )
  333 )
  334 )
  335 )
  336 )
  337 )
  338 )
  339 )
  340 )
  341 )
  342 )
  343 )
  344 )
  345 )
  346 )
  347 )
  348 )
  349 )
  350 )
  351 )
  352 )
  353 )
  354 )
  355 )
  356 )
  357 )
  358 )
  359 )
  360 )
  361 )
  362 )
  363 )
  364 )
  365 )
  366 )
  367 )
  368 )
  369 )
  370 )
  371 )
  372 )
  373 )
  374 )
  375 )
  376 )
  377 )
  378 )
  379 )
  380 )
  381 )
  382 )
  383 )
  384 )
  385 )
  386 )
  387 )
  388 )
  389 )
  390 )
  391 )
  392 )
  393 )
  394 )
  395 )
  396 )
  397 )
  398 )
  399 )
  400 )
  401 )
  402 )
  403 )
  404 )
  405 )
  406 )
  407 )
  408 )
  409 )
  410 )
  411 )
  412 )
  413 )
  414 )
  415 )
  416 )
  417 )
  418 )
  419 )
  420 )
  421 )
  422 )
  423 )
  424 )
  425 )
  426 )
  427 )
  428 )
  429 )
  430 )
  431 )
  432 )
  433 )
  434 )
  435 )
  436 )
  437 )
  438 )
  439 )
  440 )
  441 )
  442 )
  443 )
  444 )
  445 )
  446 )
  447 )
  448 )
  449 )
  450 )
  451 )
  452 )
  453 )
  454 )
  455 )
  456 )
  457 )
  458 )
  459 )
  460 )
  461 )
  462 )
  463 )
  464 )
  465 )
  466 )
  467 )
  468 )
  469 )
  470 )
  471 )
  472 )
  473 )
  474 )
  475 )
  476 )
  477 )
  478 )
  479 )
  480 )
  481 )
  482 )
  483 )
  484 )
  485 )
  486 )
  487 )
  488 )
  489 )
  490 )
  491 )
  492 )
  493 )
  494 )
  495 )
  496 )
  497 )
  498 )
  499 )
  500 )
  501 )
  502 )
  503 )
  504 )
  505 )
  506 )
  507 )
  508 )
  509 )
  510 )
  511 )
  512 )
  513 )
  514 )
  515 )
  516 )
  517 )
  518 )
  519 )
  520 )
  521 )
  522 )
  523 )
  524 )
  525 )
  526 )
  527 )
  528 )
  529 )
  530 )
  531 )
  532 )
  533 )
  534 )
  535 )
  536 )
  537 )
  538 )
  539 )
  540 )
  541 )
  542 )
  543 )
  544 )
  545 )
  546 )
  547 )
  548 )
  549 )
  550 )
  551 )
  552 )
  553 )
  554 )
  555 )
  556 )
  557 )
  558 )
  559 )
  560 )
  561 )
  562 )
  563 )
  564 )
  565 )
  566 )
  567 )
  568 )
  569 )
  570 )
  571 )
  572 )
  573 )
  574 )
  575 )
  576 )
  577 )
  578 )
  579 )
  580 )
  581 )
  582 )
  583 )
  584 )
  585 )
  586 )
  587 )
  588 )
  589 )
  590 )
  591 )
  592 )
  593 )
  594 )
  595 )
  596 )
  597 )
  598 )
  599 )
  600 )
  601 )
  602 )
  603 )
  604 )
  605 )
  606 )
  607 )
  608 )
  609 )
  610 )
  611 )
  612 )
  613 )
  614 )
  615 )
  616 )
  617 )
  618 )
  619 )
  620 )
  621 )
  622 )
  623 )
  624 )
  625 )
  626 )
  627 )
  628 )
  629 )
  630 )
  631 )
  632 )
  633 )
  634 )
  635 )
  636 )
  637 )
  638 )
  639 )
  640 )
  641 )
  642 )
  643 )
  644 )
  645 )
  646 )
  647 )
  648 )
  649 )
  650 )
  651 )
  652 )
  653 )
  654 )
  655 )
  656 )
  657 )
  658 )
  659 )
  660 )
  661 )
  662 )
  663 )
  664 )
  665 )
  666 )
  667 )
  668 )
  669 )
  670 )
  671 )
  672 )
  673 )
  674 )
  675 )
  676 )
  677 )
  678 )
  679 )
  680 )
  681 )
  682 )
  683 )
  684 )
  685 )
  686 )
  687 )
  688 )
  689 )
  690 )
  691 )
  692 )
  693 )
  694 )
  695 )
  696 )
  697 )
  698 )
  699 )
  700 )
  701 )
  702 )
  703 )
  704 )
  705 )
  706 )
  707 )
  708 )
  709 )
  710 )
  711 )
  712 )
  713 )
  714 )
  715 )
  716 )
  717 )
  718 )
  719 )
  720 )
  721 )
  722 )
  723 )
  724 )
  725 )
  726 )
  727 )
  728 )
  729 )
  730 )
  731 )
  732 )
  733 )
  734 )
  735 )
  736 )
  737 )
  738 )
  739 )
  740 )
  741 )
  742 )
  743 )
  744 )
  745 )
  746 )
  747 )
  748 )
  749 )
  750 )
  751 )
  752 )
  753 )
  754 )
  755 )
  756 )
  757 )
  758 )
  759 )
  760 )
  761 )
  762 )
  763 )
  764 )
  765 )
  766 )
  767 )
  768 )
  769 )
  770 )
  771 )
  772 )
  773 )
  774 )
  775 )
  776 )
  777 )
  778 )
  779 )
  780 )
  781 )
  782 )
  783 )
  784 )
  785 )
  786 )
  787 )
  788 )
  789 )
  790 )
  791 )
  792 )
  793 )
  794 )
  795 )
  796 )
  797 )
  798 )
  799 )
  800 )
  801 )
  802 )
  803 )
  804 )
  805 )
  806 )
  807 )
  808 )
  809 )
  810 )
  811 )
  812 )
  813 )
  814 )
  815 )
  816 )
  817 )
  818 )
  819 )
  820 )
  821 )
  822 )
  823 )
  824 )
  825 )
  826 )
  827 )
  828 )
  829 )
  830 )
  831 )
  832 )
  833 )
  834 )
  835 )
  836 )
  837 )
  838 )
  839 )
  840 )
  841 )
  842 )
  843 )
  844 )
  845 )
  846 )
  847 )
  848 )
  849 )
  850 )
  851 )
  852 )
  853 )
  854 )
  855 )
  856 )
  857 )
  858 )
  859 )
  860 )
  861 )
  862 )
  863 )
  864 )
  865 )
  866 )
  867 )
  868 )
  869 )
  870 )
  871 )
  872 )
  873 )
  874 )
  875 )
  876 )
  877 )
  878 )
  879 )
  880 )
  881 )
  882 )
  883 )
  884 )
  885 )
  886 )
  887 )
  888 )
  889 )
  890 )
  891 )
  892 )
  893 )
  894 )
  895 )
  896 )
  897 )
  898 )
  899 )
  900 )
  901 )
  902 )
  903 )
  904 )
  905 )
  906 )
  907 )
  908 )
  909 )
  910 )
  911 )
  912 )
  913 )
  914 )
  915 )
  916 )
  917 )
  918 )
  919 )
  920 )
  921 )
  922 )
  923 )
  924 )
  925 )
  926 )
  927 )
  928 )
  929 )
  930 )
  931 )
  932 )
  933 )
  934 )
  935 )
  936 )
  937 )
  938 )
  939 )
  940 )
  941 )
  942 )
  943 )
  944 )
  945 )
  946 )
  947 )
  948 )
  949 )
  950 )
  951 )
  952 )
  953 )
  954 )
  955 )
  956 )
  957 )
  958 )
  959 )
  960 )
  961 )
  962 )
  963 )
  964 )
  965 )
  966 )
  967 )
  968 )
  969 )
  970 )
  971 )
  972 )
  973 )
  974 )
  975 )
  976 )
  977 )
  978 )
  979 )
  980 )
  981 )
  982 )
  983 )
  984 )
  985 )
  986 )
  987 )
  988 )
  989 )
  990 )
  991 )
  992 )
  993 )
  994 )
  995 )
  996 )
  997 )
  998 )
  999 )
  1000 )

```

```

utils.js - medweb-master - Visual Studio Code
EXPLORER
MEDWEB-MASTER
components
lib
  auth-context.jsx
  config.js
  hooks.js
  JS utils.js
  node_modules
  pages
  public
  server
  node_modules
  .env
  index.js
  package-lock.json
  package.json
  yarn.lock
  styles
  .firebaserc
  .gitignore
  .prettierrc
  firebase.json
  LICENSE
  next-env.d.ts
  next.config.js
  package-lock.json
  package.json
  postcss.config.js
  README.md
  tailwind.config.js
  OUTLINE
  TIMELINE
lib > JS utils.js > getUserRole
  15 export const getUserRole = async (user) => {
  16   if (!user) return ""
  17   const app = initializeApp(firebaseConfig)
  18   const db = getFirestore(app)
  19   const docSnap = await getDoc(doc(db, "userInfo", user.uid))
  20
  21   if (docSnap.exists()) {
  22     return docSnap.data().role
  23   } else {
  24     return ""
  25   }
  26 }
  27
  28 export const getFullName = async (user) => {
  29   if (!user) return ""
  30   const app = initializeApp(firebaseConfig)
  31   const db = getFirestore(app)
  32   const docSnap = await getDoc(doc(db, "userInfo", user.uid))
  33
  34   if (docSnap.exists()) {
  35     return `${docSnap.data().firstname} ${docSnap.data().lastname}`
  36   } else {
  37     return ""
  38   }
  39 }
  40
  41 export const getFullNameById = async (uid) => {
  42   if (!uid) return ""
  43   const app = initializeApp(firebaseConfig)
  44   const db = getFirestore(app)
  45   const docSnap = await getDoc(doc(db, "userInfo", uid))
  46
  47   if (docSnap.exists()) {
  48     return `${docSnap.data().firstname} ${docSnap.data().lastname}`
  49   } else {
  50     return ""
  51   }
  52 }

```

```

File Edit Selection View Go Run Terminal Help      utils.js - medweb-master - Visual Studio Code
EXPLORER
MEDWEB-MASTER
  components
  lib
    auth-context.jsx
    config.js
    hooks.js
    utils.js
  node_modules
  pages
  public
  server
    node_modules
    .env
    index.js
    package-lock.json
    package.json
    yarn.lock
  styles
  .firebaserc
  .gitignore
  .prettierrc
  firebase.json
  LICENSE
  next-env.d.ts
  next.config.js
  package-lock.json
  package.json
  postcss.config.js
  README.md
  OUTLINE
  TIMELINE

lib > JS utils.js > [0] getUserRole
54 export const getPhoneNumber = async (uid) => {
55   if (!uid) return ""
56   const app = initializeApp(firebaseConfig)
57   const db = getFirestore(app)
58   const docSnap = await getDoc(doc(db, "userInfo", uid))
59
60   if (docSnap.exists()) {
61     return `${docSnap.data().phone}`
62   } else {
63     return ""
64   }
65 }
66
67 export const getAppointmentsBy = async (uid) => {
68   if (!uid) return []
69   const app = initializeApp(firebaseConfig)
70   const db = getFirestore(app)
71   const q = query(collection(db, "appointments"), where("uid", "==", uid))
72   const querySnapshot = await getDocs(q)
73   return Array.from(querySnapshot.docs)
74     .map((doc) => doc.data())
75     .filter((value) => JSON.stringify(value) !== "{}")
76 }
77
78 export const getAppointmentsFor = async (uid) => {
79   if (!uid) return []
80   const app = initializeApp(firebaseConfig)
81   const db = getFirestore(app)
82   const q = query(collection(db, "appointments"))
83   const querySnapshot = await getDocs(q)
84   // inefficient, find out faster way
85   return Array.from(querySnapshot.docs)
86     .map((doc) => doc.data())
87     .filter((value) => JSON.stringify(value) !== "{}")
88     .filter((apt) => apt.submission_uid === uid)

```

```

File Edit Selection View Go Run Terminal Help      utils.js - medweb-master - Visual Studio Code
EXPLORER
MEDWEB-MASTER
  components
  lib
    auth-context.jsx
    config.js
    hooks.js
    utils.js
  node_modules
  pages
  public
  server
    node_modules
    .env
    index.js
    package-lock.json
    package.json
    yarn.lock
  styles
  .firebaserc
  .gitignore
  .prettierrc
  firebase.json
  LICENSE
  next-env.d.ts
  next.config.js
  package-lock.json
  package.json
  postcss.config.js
  README.md
  OUTLINE
  TIMELINE

lib > JS utils.js > [0] getUserRole
90
91 export const getAllSubmissions = async () => {
92   const app = initializeApp(firebaseConfig)
93   const db = getFirestore(app)
94   const q = query(collection(db, "submissions"))
95   const querySnapshot = await getDocs(q)
96   return Array.from(querySnapshot.docs).map((doc) => doc.data())
97 }
98
99 export const getSubmissions = async (user) => {
100   if (!user) return {}
101   const app = initializeApp(firebaseConfig)
102   const db = getFirestore(app)
103   const q = query(collection(db, "submissions"), where("uid", "==", user.uid))
104   const querySnapshot = await getDocs(q)
105   return Array.from(querySnapshot.docs)
106 }
107
108 export const getNextApt = async (uid) => {
109   const apts = await getAppointmentsFor(uid)
110   if (apts.length !== 0) {
111     const apt = apts.reduce((a, b) =>
112       new Date(a.date) > new Date(b.date) ? a : b
113     )
114     return apt
115   }
116 }
117
118 export const textTo = async (phone, msg) => {
119   await fetch(`http://localhost:8080/text?phone=${phone}&msg=${msg}`, {
120     method: "POST",
121     headers: {
122       Accept: "application/json",
123       "Content-Type": "application/json",

```

```
server > JS index.js > ...
1  const express = require("express")
2  const dotenv = require("dotenv")
3  const bodyParser = require("body-parser")
4  const cors = require("cors")
5  dotenv.config()
6
7  const accountSid = process.env.TWILIO_ACCOUNT_SID
8  const authToken = process.env.TWILIO_AUTH_TOKEN
9  const client = require("twilio")(accountSid, authToken)
10
11  const app = express()
12
13  app.use(cors())
14  app.use(bodyParser.urlencoded({ extended: false }))
15  app.use(bodyParser.json())
16
17  app.post("/text", (req, res) => {
18    const { phone, msg } = req.body
19    client.messages
20      .create({ body: msg, from: "+12057976004", to: phone })
21      .then((message) => `Sent ${message.sid}`)
22  })
23
24  app.listen(8080, () => {
25    console.log("Express server listening on port 8080")
26  })
```

```
firebase.json > ...
1  {
2    "hosting": {
3      "public": "public",
4      "ignore": ["firebase.json", "**/.*", "**/node_modules/**"],
5      "rewrites": [
6        {
7          "source": "**",
8          "destination": "/index.html"
9        }
10     ]
11   },
12   "extensions": {
13     "send-message": "twilio/send-message@0.2.1"
14   }
15 }
16
```