

MOBILE APP DEVELOPMENT FOR TELEMEDICINE APPLICATION

Submitted in partial fulfillment of the requirements for the award of Bachelor of
Engineering Degree in Electronics and Communication Engineering

By

MOHAMED FAYAS.M (40130701)

MOHAMED ABDUL RAHAMAN S (40130274)



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
SCHOOL OF ELECTRICAL AND ELECTRONICS**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

CATEGORY -1 UNIVERSITY BY UGC

**Accredited with Grade "A++" by NAAC | 12B Status by UGC | Approved by AICTE
JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119**

APRIL- 2024



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited "A++" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this project Report is the bona-fide work of **MOHAMED FAYAS.M (40130701)**, **MOHAMED ABDUL RAHMAN.S (40130274)**, who have done the Project as a team who carried out the project entitled "**MOBILE APP DEVELOPMENT FOR TELEMEDICINE APPLICATION**" under my supervision from November 2023 to April 2024

Internal Guide

Dr.I. REXILINE SHEEBA, M.E., Ph.D.,

Head of the Department

Dr. T. RAVI, M.E., Ph.D.

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

We **MOHAMED ABDUL RAHMAN.S (40130274), MOHAMED FAYAS.M (40130701)** hereby declare that the Project Report entitled "**MOBILE APP DEVELOPMENT FOR TELEMEDICINE APPLICATION**" done by us under the guidance of **Dr.I.REXLINE SHEEBA M.E.,Ph.D** Associate professor at SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY is submitted in partial fulfillment of the requirements for the award Bachelor of Engineering Degree in DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING.

DATE:17.4.2024

PLACE: Chennai

SIGNATURE OF THE CANDIDATES

MOHAMED FAYAS.M(40130701) *M. Mohamed Fayas (40130701)*

MOHAMED ABDUL RAHMAN.S(40130274)

ACKNOWLEDGEMENT

I am pleased to acknowledge our sincere thanks to the **Board of Management of SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr.N.M.Nandhitha M.E., Ph.D., Dean, School of Electrical and Electronics** and **Dr.T.Ravi, M.E.,Ph.D, Head of the Department, Dept. of Electronics and Communication Engineering** for providing us necessary support and details at the right time during the progressive reviews.

I would like to express our sincere and deep sense of gratitude to my Project Guide **Dr.REXILINE SHEEBA,M.E,Ph.D** for his valuable guidance, suggestions and constant encouragement paved the way for the successful completion of our projectwork.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Electronics and Communication Engineering** who were helpful in many ways for the completion of the Project

ABSTRACT

Aim of this project develop of a mobile application designed to enhance the experience for patients and streamline telemedicine practice management. The rapid advancements in mobile technology have opened up new possibilities for improving healthcare services, and dentistry is no exception. The proposed mobile app aims to address various challenges faced by both dental practitioners and patients by providing a user-friendly, accessible, and comprehensive solution.

In these task, the modules of client and specialist in these application, the client register and login in to the application. The client login with legitimate verification, in these application the client catch picture of skin and transfer to specialist, the specialist register and login in to the application, in client side the specialist list view and the client select the specialist to send pictures with his information. The specialist get and add the after effect of his sickness. The specialist send the installment add up to the client, the client once pay the sum to the specialist the specialist confirm and send the remedy report in these application. At long last the client get the record to download.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO
	ABSTRACT	V
	LIST OF ABBREVIATIONS	IX
	LIST OF FIGURES	VIII
1	INTRODUCTION	1
	1.1 METHODOLOGIES	1
	1.2 MODULE DESCRIPTION	1
	1.2.1 USER REGISTER	2
	1.2.2 DOCTOR REGISTER AND LOGIN	2
	1.2.3 USER APPLICATION	2
	1.3 MODULE DIAGRAM	3
	1.3.1 DOCTOR	3
2	LITERATURE SURVEY	4
	2.1 BACKGROUND AND NEED	4
	2.2 LITERATURE SUMMARY	5
	2.3 STANDARDS	8
	2.4 CONTRASTS	9
	2.5 TRANSOFF	11
3	PROPOSED SYSTEM	12
	3.1 PROPOSED METHOD	12
	3.3 REQUIREMENTS ENGINEERING	13
	3.3.1 GENERAL	14
	3.3.2 HARDWARE REQUIREMENTS	14
	3.3.3 SOFTWARE REQUIREMENTS	14
	3.3.4 USE CASE DIAGRAM	15

	3.3.5 STATE DIAGRAM	15
	3.3.6 ACTIVITY DIAGRAM	16
	3.3.7 DATA FLOW DIAGRAM	17
	3.3.8 LAYOUT	17
	3.3.9 WIDGETS	18
	3.3.10 FIREBASE	19
	3.3.11 SOFTWARE TESTING	20
	3.3.12 ECONOMIAL FEASIBILITY	20
	3.3.13 VARIOUS LEVELS OF TESTING	21
	3.3.14 WHITE BOX TESTING	21
	3.3.15 BLACK BOX TESTING	22
	3.3.16 UNIT TESTING	22
	3.3.17 FUNCTIONAL TESTING	23
	3.3.18 PERFORMANCE TESTING	24
	3.3.19 INTEGRATION TESTING	24
	3.3.20 VALIDATION TESTING	25
	3.3.21 SYSTEM TESTING	25
	3.3.22 OUTPUT TESTING	26
	3.3.23 USER ACCEPTANCE TESTING	26
4	SIMULATIONS RESULTS	27
	4.1 CODING	27
	4.2 OUTPUT	30
5	CONCLUSION	32
	5.1 FUTURE ENHANCEMENTS	32
	REFERENCE	33

LIST OF FIGURES

FIGURE NO.	TITLE	PAGE NO.
1.1	METHODOLOGIES	1
1.2	MODULE DESCRIPTION	1
1.3	MODULE DIAGRAM	3
2.1	BACKGROUND AND NEED	4
2.2	LITERATURE SUMMARY	5
2.3	STANDARDS	8
2.4	CONSTRAINTS	9
2.5	TRANSDOFF	10
3.1	PROPOSED METHOD	12
3.2	EXISTING METHOD	13
3.3	REQUIEMENTS	14
4.1	CODING	27
4.2	OUTPUT	30
5.2	FUTURE ENHANCEMENTS	31

LIST OF ABBREVIATIONS

PDA	PERSONAL DIGITAL ASSISTANTS
SMS	SHORT MESSAGING SERVICE
WHO	WORLD HEALTH ORGANISATION
EHR	ELECTRONIC HEALTH RECORDS
HSCA	HASH – SOLOMON CODE ALGORITHM
DFD	DATA FLOW DIAGRAM
SGML	STANDARD GENERALISED MARKUP LANGUAGE
BAAS	BACKEND AS A SERVICE
QA	QUALITY ASSURANCE
FRS	FUNCTIONAL REQUIRMENTS SPECIFICATION
SRS	SYSTEM REQUIRMENTS SPECIFICATION

CHAPTER 1

INTRODUCTION

In these undertaking, the modules of client and expert in this application, the client register and login in to the application. The client login with authentic check, in this application the client gets image of skin and move to trained professional, the expert register and login in to the application, in client side the expert rundown view and the client select the expert to send pictures with his data. The expert gets and add the eventual outcome of his affliction. The expert sends the portion amount to the client, the client once pays the aggregate to the expert the expert affirms and send the cure report in this application. Finally, the client gets the record to download. The adoption of mobile and wireless technologies in healthcare has the potential to transform health service delivery on a global perspective. This transformation is a result of the following achievements: rapid advances in mobile networks and applications, the trend of integrating mobile health solutions into institutional health systems, and the increase in funding, regulation and evidence. Medical and public health practice supported by mobile devices, such as mobile phones, patient monitoring devices, personal digital assistants (PDAs), and other wireless devices. mHealth involves the use and capitalization on a mobile phone's core utility of voice and short messaging service (SMS) as well as more complex functionalities.

1.1 METHODOLOGIES

Methodologies is the process of analyzing the principles or procedure of a Progressive Anonymous Database management system.

1.2 MODULE DESCRIPTION

1.2.1 USER REGISTER AND LOGIN

This is the first module in our project, here symbolizes a unit of work performed within a database management system (or similar system) against a database, and treated in a coherent and reliable way independent of other transactions. A transaction generally represents any change in database user will transfer the

amount to provider. They registered with datas the verification of email once verified they able to login in to the application.

1.2.2 DOCTOR REGISTER AND LOGIN

This is the first module in our project, here symbolizes a unit of work performed within a database management system (or similar system) against a database, and treated in a coherent and reliable way independent of other transactions. A transaction generally represents any change in database user will transfer the amount to provider. They registered with datas the verification of email once verified they able to login in to the application.

1.2.3 USER APPLICATION:

The user applies the form of application of the patient details with the location of datas with capture live image from android mobile device to upload with other datas in this application to share to particular doctor.

- **DOCTOR ACCEPT:**

If accept the application form of the user with view the images in this application. The doctor updates the disease datas in this application.

- **UPLOAD PRESCRIPTION FILE:**

In they upload the prescription of doctor to user of patient of particular datas of management in these application to database.

- **PAYMENT TO DOCTOR:**

In these modules, the user pays the amount of doctor requested amount and gpay number, they successfully payed details pass to doctor of this application.

- **DOWNLOAD PRESCRIPTION:**

After successfully payed, they automatically view the download option and the user download the prescription documents.

1.3 MODULE DIAGRAM

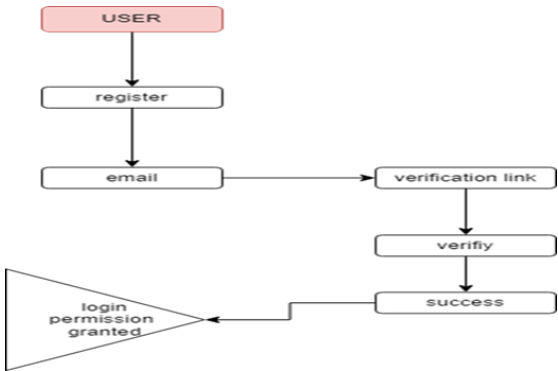


Fig:1.1: Module Diagram

1.3.1 DOCTOR

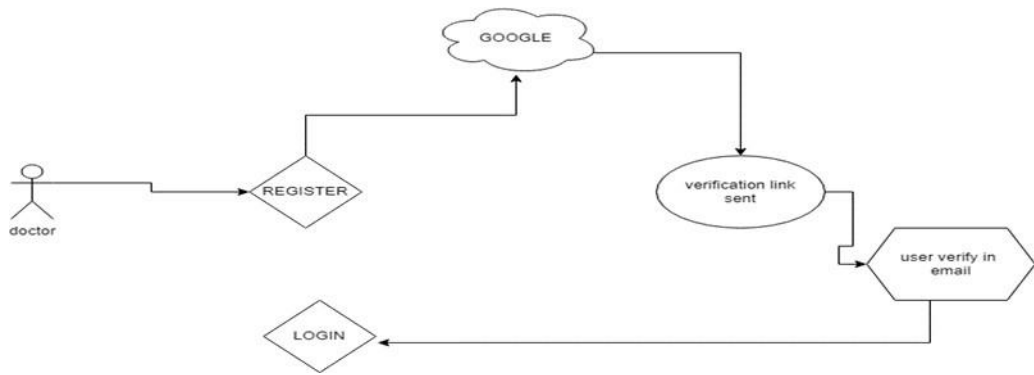


Fig 1.2: Doctor

CHAPTER 2

LITERATURE SURVEY

2.1 BACKGROUND AND NEED

The word taxonomy originates from the Greek word taxis, which means arrangement or order, and nomos, meaning law or science. In the context of information management, the term taxonomy is used in both a specific context to refer to a hierarchical classification or categorization system and in a broad sense, referring to any means of organizing concepts of knowledge. When developing a taxonomy, it is important to consider appropriately separating elements of a group into subgroups that are mutually exclusive, unambiguous, and as a whole, include all possibilities. For a taxonomy to be applicable in the real world, it must also be uncomplicated and easy to understand and use. We do not intend to present our taxonomy as an absolute classification scheme, but rather as a starting point to examine core components of the dimensions and categories of mHealth applications that could lead to a widely accepted taxonomy. The World Health Organization published an article discussing barriers to implementing mHealth technologies into health practices. Of those identified, lack of knowledge was the second highest rated barrier to implementation. Research and publication of reliable findings is needed for all identified barriers. We propose that our taxonomy may be a helpful foundation for addressing the lack of knowledge inhibiting the adoption of mHealth. Rapidly emerging mHealth technologies has resulted in an overwhelming amount of unorganized information; information that is necessary to understand before successful implementation can take place. Categorizations of available medical applications in meaningful groupings with definitions with intended purpose of uses have not yet been suggested. In identifying this gap, our research goal was to review available literature about existing and emerging medical applications and in doing so, systematically place them into appropriate categories with definitions and purpose of use in an understandable organizational scheme.

2.2 LITERATURE SUMMARY

Eka Miranda Information Systems Department, School of Information Systems Bina Nusantara University Jakarta, Indonesia 11480 Based on Census Population Survey in 2015, the population of citizens in Indonesia is 255.18 million people, while according to data from the Ministry of Health of the Republic of Indonesia, the number of health workers is 1,182,808 people. There is a gap between the number of health workers and the population of citizens in Indonesia, which is still far from ideal proportion. Moreover, in addition to the lack of health workers, many Indonesians have difficulty in accessing health services, especially in the current situation as the impact of the pandemic caused by the COVID-19 virus. The pandemic has caused all sectors in Indonesia. One affected sector is health sector. There are many people need health services in COVID-19 pandemic situation, but they have difficulties to get access health services. The health services limit the number of face-to-face visits due to break the chain of COVID-19 spread. Therefore, the health services in Indonesia are not adequate in providing their services, especially for low- and middle-class communities. All citizens have the equal right to get health services started from Community Health Service in subdistrict. The objective of this research is to develop mobile-based telemedicine application for community health center to ease the community to access health services from the community health center in subdistrict.

Telemedicine is broadly defined by the World Health Organization as the remote delivery of health services using electronic means for diagnosis of treatment, prevention of disease and injury, research and evaluation, health care education. The success of telemedicine programs can be seen from the increasing level of consultations, positive acceptance, and feedback from patients. Public health informatics is defined as the systematic application of information, computer science and technology in the field of public health, including health monitoring, prevention, anticipation, and promotion. In simple terms, telemedicine has been applied when there is a discussion between two doctors discussing patient problems via mobile.

mHealth taxonomy: mobile health applications Phillip Olla Springer-Verlag Berlin Heidelberg 2015. mHealth, or Mobile Health, is a global phenomenon that has received significant attention over the last decade. mHealth is a rapidly advancing field that uses mobile technologies such as mobile phones, software applications, and devices to support the achievement of health objectives. The adoption of mobile and wireless technologies in healthcare has the potential to transform health service delivery on a global perspective. This transformation is a result of the following achievements: rapid advances in mobile networks and applications, the trend of integrating mobile health solutions into institutional health systems, and the increase in funding, regulation and evidence. mHealth is accepted as a component of eHealth. A standardized definition of mHealth has not been established. The WHO Global Observatory for eHealth (GOe) defines mHealth as: "Medical and public health practice supported by mobile devices, such as mobile phones, patient monitoring devices, personal digital assistants (PDAs), and other wireless devices. mHealth involves the use and capitalization on a mobile phone's core utility of voice and short messaging service (SMS).

A Survey of Telecommunication Technologies, Developments, and Challenges by Caroline Omoanitse Alenoghena, 2023. The concept of telemedicine encompasses the use of information and telecommunication technology to render medical services irrespective of geographical separation between physicians and patients [1]. Telemedicine has been in practice as far back as the 1900s. It covers any form of electronic communication between health workers and patients from a remote location [1,2]. Recently, researchers have focused more on wireless communication technologies for telemedicine to provide effective and reliable health care service delivery from remote location especially during emergencies. Various communication technologies have been proposed and implemented for providing expert medical services to patients without the need for the conventional face-to-face encounters with patients. This has greatly reduced the cost of medical diagnosis and the need to travel long distances in search of professional consultations. Available studies on telemedicine implementations suggest the need for continuous research to address several issues and challenges. There is a need

to compare relevant studies in the field in order to provide a broad overview of available communication technologies suitable for modern designs as well as to identify the most viable means of practical implementation. This is not to say that telemedicine should completely replace the conventional practice of physical diagnostic and other medical processes, as certain services require physical face-to-face contact. Nonetheless, the deployment of telemedicine could greatly reduce congestion in hospitals, and consequently limit the spread of infectious diseases.

Content-Based MobileHealth Youcef Benferdia,2015. The use of mobile computing has exploded and reached the commercial industry and mainstream consumers via smartphones, personal digital assistants (PDAs) [16], mobile phones and tablets. According to Gartner Inc, worldwide mobile connections will grow up to 7.4 billion by 2015, and mobile applications offer benefits that cannot be matched by desktop products. In a recent example, Apple launched new mobile phone features on the iPhone 5S that could benefit mobile health developers through its movement sensor to detect a user's movement. A variety of sensors such as microphones and cameras can be greatly employed by mobile health (m-health) apps. Moreover, mobile guide systems have become significantly advanced and offer context-based personalisation, user collaboration and social interaction. These examples of the current advantages of mobile computing drive the healthcare community to seek ways to efficiently utilize the technology to better manage people's wellbeing. One of the solutions that has been proposed by healthcare providers is to make use of the available internet-based technologies, such as mobile phones, which offer tremendous access to information in order to help people manage their health. The technologies allow healthcare providers to upload medical records, lab results, images and drug information to handheld devices such as PDAs, tablets, push-to-talk devices, cell phones or smartphones.

mobile health apps in the management of chronic conditions Maria Cucciniello, Francesco Petracca, Oriana Ciani & Rosanna Tarricone, (2021). Smartphone users in the world have steadily increased, surpassing the 3.5 billion mark in 2020 with a further expected growth of several hundred million in the coming years,

fuelling the expanding interest in mHealth apps. The coronavirus disease 2019 (COVID-19) pandemic has contributed to further accelerate the reliance on digital health. Therefore, apps are being increasingly incorporated into healthcare (HC) services owing to their portability, instantaneous access and direct communication, inspiring new models of remote HC delivery and cost-effective solutions for chronic diseases, whose long-term nature and need for continuous monitoring can be positively impacted. mHealth apps may be particularly effective in self-management, one of the components of the eHealth Enhanced Chronic Care Model⁸, intended as the ‘individual’s ability to manage the symptoms, treatment, physical and psychosocial consequences and lifestyle changes inherent in living with a chronic disease’⁹. Apps could as well improve patient empowerment, the process of gaining knowledge of one’s health and ability and motivation to influence.

2.3 STANDARDS

- Client Registration and Login: Allow patients to register and log in securely to the application.
- Image Upload: Enable clients to capture images of their dental issues and upload them to the application.
- Specialist Registration and Login: Allow dental specialists to register and log in securely to the application.
- Specialist List View: Provide clients with a list of available specialists, allowing them to select the one they want to consult with.
- Image Review and Diagnosis: Allow specialists to view the images uploaded by clients and provide a diagnosis or treatment plan.
- Payment Processing: Facilitate payment transactions between clients and specialists for consultations and services rendered.
- Prescription and Report Delivery: Enable specialists to generate and send prescription reports to clients after consultation.

- **Record Management:** Allow clients to access and download their consultation records and reports.

Regarding standards, it's crucial to ensure compliance with relevant healthcare regulations and standards, such as HIPAA (Health Insurance Portability and Accountability Act) for patient data privacy and security, as well as any other applicable regulations in your region. Additionally, you may want to adhere to usability and accessibility standards to ensure the application is user-friendly for both patients and specialists. Continuous testing and iteration based on user feedback will be essential to refine and improve the application's functionality and user experience over time.

2.4 CONSTRAINTS

- **Regulatory Compliance:** Ensure compliance with healthcare regulations and standards, such as HIPAA (Health Insurance Portability and Accountability Act), to safeguard patient data privacy and security.
- **Data Security:** Implement robust measures to protect sensitive patient information during transmission, storage, and access within the application.
- **Authentication and Authorization:** Implement secure authentication mechanisms to verify the identities of both clients and specialists, ensuring only authorized individuals can access patient data and perform actions within the app.
- **Image Transfer:** Develop secure methods for clients to capture and transfer images of their dental issues to specialists, maintaining image quality and integrity while adhering to privacy regulations.
- **Payment Processing:** Utilize secure payment gateways to facilitate transactions between clients and specialists, ensuring financial data is encrypted and protected.
- **User Experience:** Design the application with a user-friendly interface and intuitive navigation to enhance the experience for both patients and practitioners, considering accessibility guidelines to accommodate users with diverse needs.

- Scalability: Build the application with scalability in mind to accommodate potential growth in user base and data volume over time.
- Interoperability: Consider integration capabilities with existing healthcare systems or platforms to facilitate seamless communication and data exchange between the telemedicine app and other healthcare providers or systems.
- Resource Constraints: Account for potential limitations in device capabilities, network connectivity, and system resources, particularly in regions with varying levels of technological infrastructure.
- Feedback and Iteration: Plan for continuous feedback collection from users and stakeholders to identify areas for improvement and iterate on the application's features and functionality over time.

2.5 TRADEOFF

- Security vs. Usability: Implementing stringent security measures, such as multi-factor authentication and encryption, may enhance data protection but could also add complexity to the user experience. Finding the right balance between security and usability is crucial.
- Features vs. Performance: Including a wide range of features to cater to diverse user needs may enhance the application's functionality but could also impact performance, especially on devices with limited resources. Prioritizing essential features and optimizing performance is essential.
- Customization vs. Standardization: Offering customizable options for both patients and specialists may enhance user satisfaction but could also complicate the development process and increase maintenance efforts. Finding a balance between customization and standardization can help streamline development while meeting user needs.

- Integration vs. Independence: Integrating the application with existing healthcare systems and platforms may improve interoperability but could also introduce compatibility issues and dependencies. Deciding the extent of integration versus maintaining independence is critical.
- Cost vs. Quality: Investing in high-quality development, security, and user experience may result in a superior application but could also incur higher costs. Balancing cost considerations with the desired level of quality is essential for project success.
- Scalability vs. Complexity: Designing the application to scale effectively to accommodate a growing user base may require a more complex architecture and infrastructure. Balancing scalability needs with the complexity of implementation is crucial to avoid over-engineering.
- Feedback vs. Time-to-Market: Soliciting feedback from users and stakeholders throughout the development process is valuable for iterative improvements but could also extend the time-to-market. Finding ways to gather feedback efficiently without compromising development timelines is important.

CHAPTER 3

3.1 PROPOSED METHOD

The proposed system involves the development of a dedicated mobile app for a telemedicine application, aiming to build upon and enhance the existing telemedicine ecosystem. The primary goal is to provide a user-friendly, secure, and comprehensive platform for telemedicine services, catering to the needs of both patients and healthcare providers. Here's an outline of the key components and improvements in the proposed system. The heart of the proposed system is a feature-rich mobile app available on both iOS and Android platforms. This app is designed to be user-friendly and accessible to individuals seeking telemedicine services.

Patients and healthcare providers can create profiles by providing essential information, including personal details, medical credentials, and preferences. User profiles are a central part of the app, helping users manage appointments, access health records, and communicate securely. Patients can schedule appointments with healthcare providers of their choice using a User-friendly calendar interface. Healthcare providers can manage their availability and appointment schedules through the app.

The mobile app offers secure, end-to-end encrypted video consultations. This ensures the privacy of patient data and promotes trust in the telemedicine process. The app incorporates a secure messaging system to facilitate non-emergency communication between patients and healthcare providers. This messaging system ensures the swift and secure exchange of medical information.

3.2 EXISTING METHOD

Before introducing a new mobile app for a telemedicine application, it's essential to understand the existing system, if any, to identify its strengths and weaknesses. As of my last knowledge update in September 2021, here's an overview of the typical components and characteristics of an existing telemedicine system. Many healthcare providers use web-based platforms for telemedicine. These platforms allow patients and healthcare professionals to connect through web browsers for video consultations. However, these platforms may lack the convenience and mobility of a dedicated mobile app. Some healthcare providers and patients might use third-party video conferencing apps like Zoom, Skype, or Google Meet for telemedicine consultations. While these tools are widely available, they may not be tailored to the specific needs of healthcare, such as integration with Electronic Health Records (EHR) or secure messaging. Electronic Health Record systems are used for storing patient information, including medical history, diagnoses, medications, and treatment plans. These systems might not be integrated with telemedicine solutions, requiring manual data entry during or after telemedicine appointments.

3.3 REQUIREMENTS ENGINEERING

3.3.1 GENERAL

These are the requirements for doing the project. Without using these tools and software's we can't do the project. So, we have two requirements to do the project. They are

1. Hardware Requirements.
2. Software Requirements.

3.3.2 HARDWARE REQUIREMENTS

- PROCESSOR : GREATER THAN I3
- RAM : 4GB DD RAM

- HARD DISK : 1TB

3.3.3 SOFTWARE REQUIREMENTS

- FRAMEWORK : FLUTTER
- OPERATING SYSTEM : WINDOWS 10
- IDE : ANDROID STUDIO
- DATABASE : FIREBASE

3.3.4 USE-CASE DIAGRAM

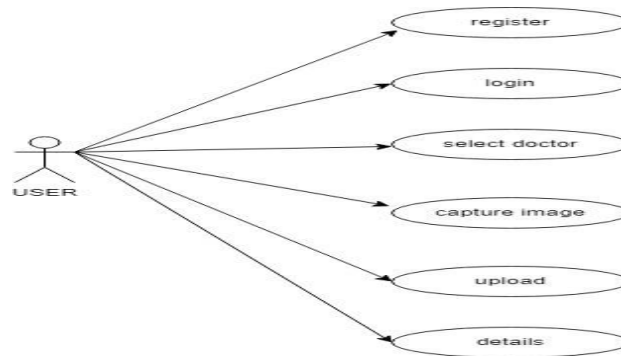


Fig :3.3.4: Use Case Diagram

The use case diagram is the main building block of object oriented modeling. It is used both for general conceptual modeling of the systematic of the application, and for detailed modeling translating the models into programming code. For this in our component diagram first propose a data. In this proposed method we are using Hash-Solomon Code Algorithm to encrypt the data.

3.3.5 STATE DIAGRAM

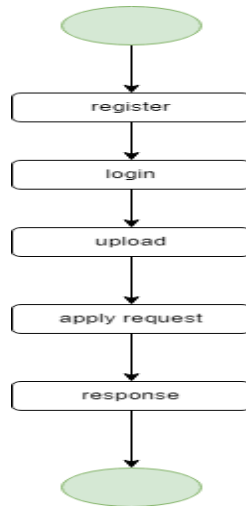


Fig: 3.3.5: State Diagram

State diagrams require that the system described is composed of a finite number of states; sometimes, this is indeed the case, while at other times this is a reasonable abstraction. Many forms of state diagrams exist, which differ slightly and have different semantics. In our state diagram first propose a. For this in our component diagram first propose a data. In this proposed method we are using Hash-Solomon Code Algorithm to encrypt the data.

3.3.6 ACTIVITY DIAGRAM:

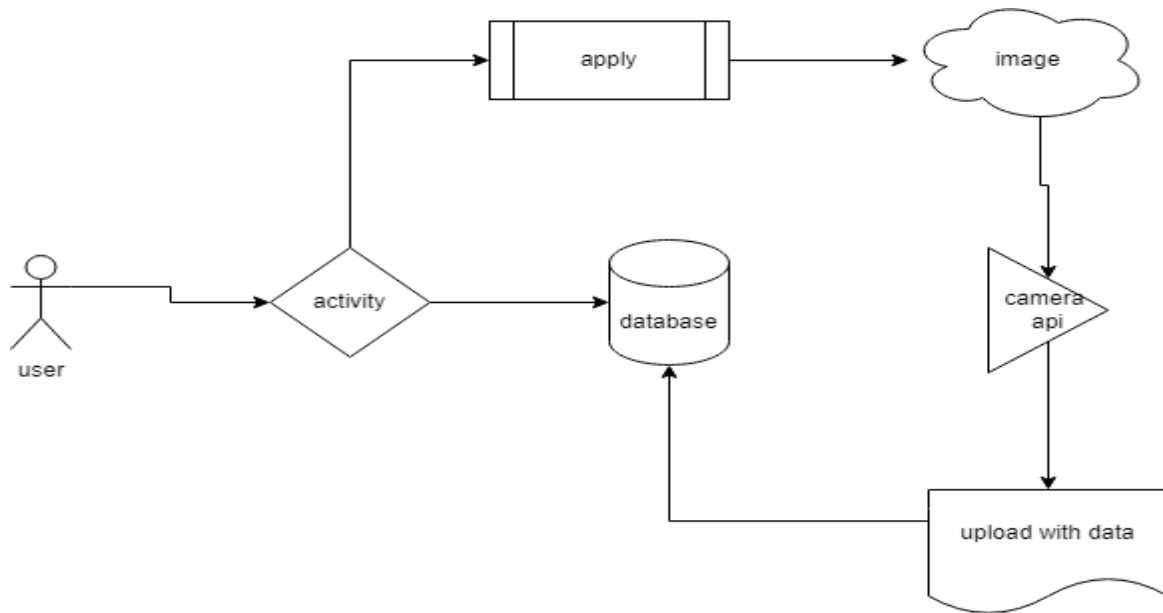


Fig: 3.3.6: ACTIVITY DIAGRAM

Activity diagram is a loosely defined diagram to show workflows of stepwise activities and actions, with support for choice, iteration and concurrency. UML, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. UML activity diagrams could potentially model the internal logic of a complex operation. In many ways UML activity diagrams are the object-oriented equivalent of flow charts and data flow diagrams (DFDs) from structural development.

3.3.7 DATAFLOW DIAGRAM

A data flow diagram (DFD) is a graphical representation of the “flow” of data through an information system. It differs from the flowchart as it shows the data flow instead of the control flow of the program. A data flow diagram can also be used for the visualization of data processing. The DFD is designed to show how a system is divided into smaller portions and to highlight the flow of data between those parts.

3.3.8 Layout

Earlier in client- server computing, each application had its own client program and it worked as a user interface and need to be installed on each user's personal computer. Most web applications use HTML/XHTML that are mostly supported by all the browsers and web pages are displayed to the client as static documents. A web page can merely display static content and it also lets the user navigate through the content, but a web application provides a more interactive experience. Any computer running Servlets or JSP needs to have a container. A container is nothing but a piece of software responsible for loading, executing and unloading the Servlets and JSP. While servlets can be used to extend the functionality of any Java- enabled server. They are mostly used to extend web servers, and are efficient replacement for CGI scripts. CGI was one of the earliest and most prominent server-side dynamic content solutions, so before going forward it is very important to know the difference between CGI and the Servlets.

3.3.9 WIDGETS

XML stands for Extensible Markup Language. XML is a markup language much like HTML used to describe data. It is derived from Standard Generalized Markup Language (SGML). Basically, the XML tags are not predefined in XML. We need to implement and define the tags in XML. XML tags define the data and used to store and organize data. It's easily scalable and simple to develop. In Android, the XML is used to implement UI-related data, and it's a lightweight markup language that doesn't make layout heavy. XML only contains tags, while implementing they need to be just invoked.

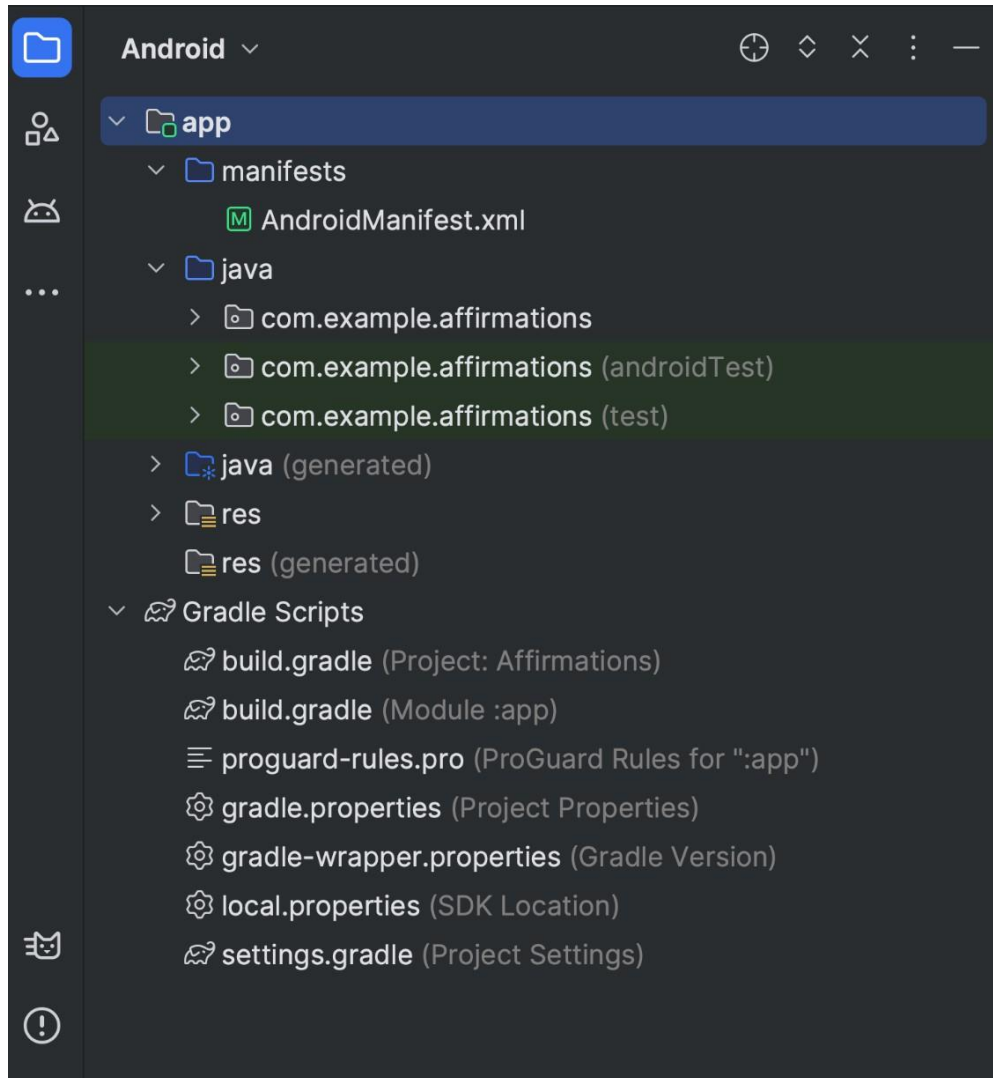


Fig 3.3.9: Widgets in Android project view

Each project in Android Studio contains one or more modules with source code files and resource files. The types of modules include:

- Android app modules
- Library modules
- Google App Engine modules

By default, Android Studio displays your project files in the Android project view, as shown in figure 1. This view is organized by modules to provide quick access to your project's key source files. All the build files are visible at the top level,

under Gradle Scripts. The build system can help you create different versions of the same app from a single project. This is useful when you have both a free version and a paid version of your app or if you want to distribute multiple APKs for different device configurations on Google Play.

Fig 3.3.10 FIREBASE

Firebase is a Backend-as-a-Service (Baas). It provides developers with a variety of tools and services to help them develop quality apps, grow their user base, and earn profit. It is built on Google's infrastructure. Firebase is categorized as a NoSQL database program, which stores data in JSON-like documents.



Fig.3.3.10: FireBase

In Firebase, a document is a set of key-value pairs defined by a schema. A group of documents makes up a collection.

3.3.11 SOFTWARE TESTING:

Feasibility studies aim to objectively and rationally uncover the strengths and weaknesses of the existing business or proposed venture, opportunities and threats as presented by the environment, the resources required to carry through, and ultimately the prospects for success.

In its simplest term, the two criteria to judge feasibility are cost required and value to be attained. As such, a well-designed feasibility study should provide a historical background of the business or project, description of the product or service, accounting statements, details of the operations and management, marketing research and policies, financial data, legal requirements and tax obligations. Generally, feasibility studies precede technical development and project implementation.

They are 3 types of Feasibility

- Economical feasibility
- Technical feasibility
- Operational feasibility

3.3.12 ECONOMICAL FEASIBILITY

The assessment is based on an outline design of system requirements in terms of Input, Processes, Output, Fields, Programs, and Procedures. This can be quantified in terms of volumes of data, trends, frequency of updating, etc. in order to estimate whether the new system will perform adequately or not. This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The software, which has been developed, has to be tested to prove its validity. Testing is considered to be the least creative phase of the whole cycle of system design. In the real sense it is the phase, which helps to bring out the creativity of the other phases makes it shine.

3.3.13 VARIOUS LEVELS OF TESTING

1. White Box Testing
2. Black Box Testing
3. Unit Testing
4. Functional Testing
5. Performance Testing
6. Integration Testing
7. Validation Testing
8. System Testing
9. Output Testing
10. User Acceptance Testing

3.3.14 WHITE BOX TESTING

White-box testing, sometimes called glass-box, is a test case design method that uses the control structure of the procedural design to derive test cases. Using White Box testing methods, we can derive test cases that

- Guarantee that all independent paths within a module have been exercised at least once
- Exercise all logical decisions on their true and false sides.
- Execute all loops at their boundaries and within their operational bounds.
- Exercise internal data structures to assure their validity.

3.3.15 BLACK BOX TESTING

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box. You cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

In this testing by knowing the internal operation of a product, test can be conducted to ensure that “all gears mesh”, that is the internal operation performs according to specification and all internal components have been adequately exercised. It fundamentally focuses on the functional requirements of the software.

3.3.16 UNIT TESTING

Unit testing is a method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures are tested to determine if they are fit for use. Intuitively, one can view a unit as the smallest testable part of an application. In procedural programming, a unit could be an entire module, but it is more commonly an individual function or procedure. In object-oriented programming, a unit is often an entire interface, such as a class, but could be an individual method. Unit tests are short code fragments created by programmers or occasionally by white box testers during the development process. Unit testing is software verification and validation method in which the individual units of source code are tested fit for use. A unit is the smallest testable part of an application. In this testing, each class is tested to be working satisfactorily.

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration.

3.3.17 FUNCTIONAL TESTING

Functional testing is a quality assurance (QA) process and a type of black box testing that bases its test cases on the specifications of the software component under test. Functions are tested by feeding them input and examining the output, and internal program structure is rarely considered (not like in white-box testing). Functional Testing usually describes what the system does. Functional testing differs from system testing in that functional testing "verifies a program by checking it against ... design document(s) or specification(s)", while system testing "validate a program by checking it against the published user or system requirements" (Kane, Falk, Nguyen 1999, p. 52). Functional testing typically involves five steps. The identification of functions that the software is expected to perform

1. The creation of input data based on the function's specifications
2. The determination of output based on the function's specifications
3. The execution of the test case
4. The comparison of actual and expected outputs.

3.3.18 PERFORMANCE TESTING

In general testing performed to determine how a system performs in terms of responsiveness and stability under a particular workload. It can also serve to investigate, measure, validate or verify other quality attributes of the system, such as scalability, reliability and resource usage. Performance testing is a subset of performance engineering, an emerging computer science practice which strives to build performance into the implementation, design and architecture of a system.

3.3.19 INTEGRATION TESTING

Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with. Individual modules, which are highly prone to interface errors, should not be assumed to work instantly when put together. The problem of course, is “putting them together”- interfacing. There may be the chances of data lost across on another’s sub functions, when combined may not produce the desired major function; individually acceptable impression may be magnified to unacceptable levels; global data structures can present problems. Integration testing is the phase in software testing in which individual software modules are combined and tested as a group. Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready. All the errors found in the system are corrected for the next phase.

The purpose of integration testing is to verify functional, performance, and reliability requirements placed on major design items. These "design items", i.e. assemblages (or groups of units), are exercised through their interfaces using black box testing, success and error cases being simulated via appropriate parameter and data inputs. Simulated usage of shared data areas and inter-process communication is tested and individual subsystems are exercised through their input interface. Test cases are constructed to test whether all the components

within assemblages interact correctly for example across procedure calls or process activations, and this is done after testing individual modules, unit testing.

3.3.20 VALIDATION TESTING

Verification and Validation are independent procedures that are used together for checking that a product, service, or system meets requirements and specifications and that it full fills its intended purpose. These are critical components of a quality management system such as ISO 9000. The words "verification" and "validation" are sometimes preceded with "Independent" (or IV&V), indicating that the verification and validation is to be performed by a disinterested third party. It is sometimes said that validation can be expressed by the query "Are you building the right thing?" and verification by "Are you building it right?". In practice, the usage of these terms varies. Sometimes they are even used interchangeably.

3.3.21 SYSTEM TESTING

System testing of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black box testing, and as such, should require no knowledge of the inner design of the code or logic. As a rule, system testing takes, as its input, all of the "integrated" software components that have passed integration testing and also the software system itself integrated with any applicable hardware system(s). The purpose of integration testing is to detect any inconsistencies between the software units that are integrated together (called assemblages) or between any of the assemblages and the hardware. System testing is a more limited type of testing; it seeks to detect defects both within the "inter-assemblages" and also within the system as a whole. System testing is performed on the entire system in the context of a Functional Requirement Specification(s) (FRS) and/or a System Requirement Specification (SRS). System testing tests not only the design, but also the behaviour and even the believed

expectations of the customer. It is also intended to test up to and beyond the bounds defined in the software/hardware requirements specification.

3.3.22 OUTPUT TESTING

After performing the validation testing, next step is output testing of the proposed system since no system could be useful if it does not produce the required output generated or considered in two ways. One is on screen and another is printed format. The output comes as the specified requirements by the user. Hence output testing does not result in any correction in the system.

3.3.23 USER ACCEPTANCE TESTING

User acceptance of a system is the factor for the success of any system. The system under consideration is tested for the user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes wherever required.

- Input screen design.
- Output screen design.
- Online message to guide user.
- Format of the ad-hoc reports and other outputs.

Taking various kinds of test data does the above testing. Preparation of test data plays a vital role in the system testing. After preparing the test data the system under study is tested using the test data. While testing the system by using test data errors are again uncovered and correct.

CHAPTER 4

SIMULATIONS RESULTS

4.1 CODINGS

```
Progress) {  
    if (inProgress) {  
        progressBar.setVisibility(View.VISIBLE);  
        login.setVisibility(View.GONE);  
        reg.setVisibility(View.GONE);  
    } else {  
        progressBar.setVisibility(View.GONE);  
        login.setVisibility(View.VISIBLE);  
        reg.setVisibility(View.VISIBLE);  
    }  
}
```

```
Path getPath(double width, double height, double innerRatio) {  
    Path path = new Path();  
  
    double dx = width / 2;  
    double dy = height / 2;  
  
    double outerRadius = width / 2;  
    double innerRadius = outerRadius * innerRatio;  
  
    double angle = -90;  
    double angleIncrement = 360 / 5;  
  
    path.moveTo(dx + outerRadius * cos(degreesToRadians(angle)),  
        dy + outerRadius * sin(degreesToRadians(angle)));  
  
    for (int i = 0; i < 5; i++) {  
        angle += angleIncrement;  
        path.lineTo(dx + innerRadius * cos(degreesToRadians(angle)),  
            dy + innerRadius * sin(degreesToRadians(angle)));  
  
        angle += angleIncrement;  
        path.lineTo(dx + outerRadius * cos(degreesToRadians(angle)),  
            dy + outerRadius * sin(degreesToRadians(angle)));  
    }
```

```

    }

    path.close();
    return path;
}

double degreesToRadians(double degrees) {
    return degrees * (pi / 180.0);
}
}

class stats extends StatefulWidget {
    const stats({super.key});

    @override
    State<stats> createState() => _MyAppState();
}

class HeartRateData {
    final int heartRate;
    final String date;

    HeartRateData({required this.heartRate, required this.date});
}

class _MyAppState extends State<stats> {
    double _iconSize = 70.0;

```

```

bool _isZoomed = false;
Timer? _timer;
final FirebaseAuth _auth = FirebaseAuth.instance;
final DatabaseReference _database = FirebaseDatabase.instance.reference();
@override
void initState() {
  super.initState();
  _fetchHeartRateData();
  Timer(Duration(seconds: 1), () {
    _startZoomAnimation();
  });
}

void _fetchHeartRateData() {
  User? user = _auth.currentUser;
  if (user != null) {
    String userId = user.uid;

    _database.child('heartRates').child(userId).onValue.listen((event) {
      // Check if the event snapshot exists and has data
      if (event.snapshot.value != null) {
        Map<dynamic, dynamic> heartRateDataMap =
          event.snapshot.value as Map<dynamic, dynamic>;

        groupedData.clear();

        heartRateDataMap.forEach((key, value) {
          HeartRateData heartRateData = HeartRateData(
            heartRate: value['heartRate'],
            date: value['date'],
          );

          if (groupedData.containsKey(heartRateData.date)) {
            groupedData[heartRateData.date]!.add(heartRateData);
          } else {
            groupedData[heartRateData.date] = [heartRateData];
          }
        });

        setState(() {});
      }
    });
  }
}

```

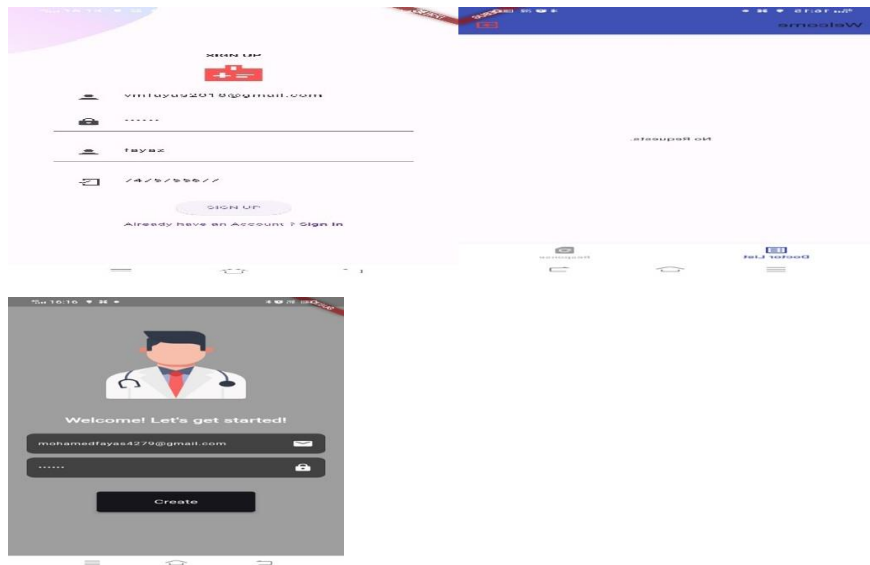
```
decoration: BoxDecoration(  
  gradient:
```

4.2 OUTPUT:

STEP 1:



STEP 2:



STEP 3:



Apply

Doctor Name

Hospital Name

Description

Specialist

Address

Location

Mobile

Experience

Submit Successfully

CHAPTER 5

CONCLUSION

The client prepared to login in this application. In essential page the game plan of ppg assessment using camera programming connection point the working principle of spot a finger in camera remembers for streak light flash and take review to assessment of picture treatment of red pixels of circulation system to recognize the derived heart beat. The communication is live acknowledgment of heart beat. The circulatory system of ECG is by and large show and scrutinizing show live scrutinizing of circulation system assessment. They show the ordinary truly check out at results with worth of area.

Xml and layout are gaining rapid acceptance as means to provide dynamic content on the Internet. With full access to the Java platform, running from the server in a secure manner, the application possibilities are almost limitless. When JSPs are used with Enterprise JavaBeans technology, e-commerce and database resources can be further enhanced to meet an enterprise's needs for web applications providing secure transactions in an open platform. J2EE technology as a whole makes it easy to develop, deploy and use web server applications instead of mingling with other technologies such as CGI and ASP. There are many tools for facilitating quick web software development and to easily convert existing server-side technologies to JSP and Servlets.

5.1. FUTURE ENHANCEMENTS

Implementing a real-world database system and evaluate more simultaneous connection. Improving the efficiency of protocols, in terms of number of messages exchanged and in terms of their sizes, as well. Implement using two are more algorithms.

REFERENCE

- [1] Smith. Human Resources for Health Country Profile. WHO Reg Off South-East Asia... WHO country profile 2(1): 123-5 2015.
- [2] Joja M L. Antiretroviral therapy in resource-poor settings: barriers to access and promoting adherence. JAIDS 43(1):s123-126. 2016.
- [3] Varghese S SR. Categorizing the Telehealth Policy Response of Countries and Their Implications for Complementary of Telehealth Policy. Tele J E-Health 10(1):32-72014.
- [4] Developing Smart Telehealth System in Indonesia Progress and Challenge.pdf>.
- [5] Kamsu-Foguem B, Tiako PF, Fotso LP, Foguem C. Modeling for effective collaboration in telemedicine. Telematics and Informatics.2015;32(4):776-86.
- [6] Bosworth A, York C, Kotansky H, Berman MA. An inventor's perspective on consumer health informatics. Am J Prev Med. 2011;40(5 Suppl2): S241-4.
- [7] Suksmono A SU, Mengko T. Overview of telemedicine activities inIndonesia: Progress and Constraints. TDB 14-12(2):1022014.
- [8] Hariyati TS SJ. Perception ofNursing Care for Cardiovascular Cases,Knowledge on Telehealth and Telecardiology in Indonesia. Int JColab Res Intern Med Public Health 4(2):116--21 2014.
- [9] Mohan J R, Raja, Yacoob, Raja. The Malaysian Telehealth FlagshipApplications Approach to Health Data Protection Utilization and Consumer Rights. Int J Med Inf2004;2(73):217-27 2004.
- [10] Wiweko B, Aulia SN, Agung PG. Indonesian Kalkulator of Oocytes(IKO): A Smart Application to Determine Our Biological Age.Indonesian Reproductive Medicine. 2016.
- [11] Wiweko B, Agung PG, Aulia SN, Zesario A. SMART APPLICATION: One Stop Services Bridging the Gap Between Doctor and Patients.

RE-2022-227363-plag-report

ORIGINALITY REPORT

17 %

SIMILARITY INDEX

11 %

INTERNET SOURCES

8 %

PUBLICATIONS

9 %

STUDENT PAPERS

PRIMARY SOURCES

1	i-scholar.in Internet Source	4 %
2	sist.sathyabama.ac.in Internet Source	2 %
3	www.rame.org.in Internet Source	<1 %
4	vpmpme.files.wordpress.com Internet Source	<1 %
5	repository.sustech.edu Internet Source	<1 %
6	www.biz4intellia.com Internet Source	<1 %
7	alkej.uobaghdad.edu.iq Internet Source	<1 %
8	jartel.polinema.ac.id Internet Source	<1 %
9	www.coursehero.com Internet Source	<1 %