Telemedicine for Additional Diagnosis

Chirantan Mahipal

USN: 1BM13IS017, Department of ISE, BMS College of Engineering , Bangalore, India

ABSTRACT

Many organizations in India have been lately involved in developing and deploying various Healthcare based initiatives to make healthcare more accessible and affordable. Telemedicine for Additional Diagnosis (TAD) is another such initiative. It aims to make better healthcare assistance available to people in all parts of the country with limited local healthcare. It helps in collaborating with multiple super specialty hospitals to bring their healthcare experts onboard this platform. TAD is to help the people deal with the very common scenario encountered when going through some medical condition. Many a times, when suffering through a medical condition, there seems to be no relief from it. The condition seems to prolong even after medication prescribed from the local doctor. This generally happens due to the misdiagnosis of the underlying condition or hasty judgments made by the local doctor, inexperience of the doctor or just due to the lack of the relevant data. On many occasions the doctor may also suggest some invasive procedure which may turn out to be expensive and leave lasting side effects. In such situations, what normally people tend to do is to approach another doctor for an additional opinion on the diagnosis provided by the previous doctor. The doctor re-examines the patient along with the details provided by the previous doctor and suggests an alternative track of treatment or corroborates what the first doctor had already said. Our aim is to assist this additional opinion. Generally, the second opinion doctor tend to be selected based on word of mouth. It becomes especially hard to find such a doctor in smaller tier-2 and tier-3 cities. That's where our platform comes into picture. To provide a medium to connect people in any part of the country to healthcare experts to gain an additional diagnosis.

Keywords—*Telemedicin, m-health, TeleHomeCare.*

---

1. INTRODUCTION

According to Medical Council of India data, the doctor-population ratio in the country stands at 1:1681. In addition, this too is based on the assumption that 80% of these doctors are available on any given day. A staggering 70% of the population still lives in rural areas and has no or limited access to hospitals and clinics. Another key driver of India's healthcare landscape is the high out-of-pocket expenditure (roughly 70%). This means that most Indian patients pay for their hospital visits and doctors' appointments with straight up cash after care with no payment arrangements. In medical terms: A second opinion can be visit to a physician other than the one a patient has previously been seeing in order to get more information or to hear a differing point of view. With TAD, we're trying to create a system that forms a layer between super-specialty hospitals and users, seeking an expert opinion concerning a particular health ailment. The system uses machine learning algorithms to improve and generate more efficient recommendations for patients over time by analyzing the effectiveness of current recommendation model and user data. TAD platform keeps the user data in the system in an encrypted form so that sensitive information to users is not in the clear. The front end serves as a data collection layers that feeds data to the ML system at the back-end.

Qiang CZ et al. talk about the potential of mobile use in the healthcare market [1]. He starts off by talking about the challenges that the mobile health sector faces due to the lack of data on the societal impact that 'm-health' has, as well as the inability to interoperate between devices. This poses difficulties for the government to expand into mobile health and witness its true potential. In this report, three case studies are described (Haiti, India and Kenya) which explore the impact of m-health services across a variety of contexts, i.e. with different kinds of financial backgrounds, technological services and health sectors. It demonstrates new remedies that are more cost-saving and interactive, thus increasing the power that these applications have in terms of impact. It further talks about the possible goals for mobile health in the future, such as patient tracking, improving the quality of healthcare and helping patients with treatments when healthcare is expensive or unavailable in the area, particularly in the context of developing countries.

Mahmud Kahlid et al. explore other alternatives to care delivery in order to remain competitive in future [2]. He proposes the use of 'TeleHomeCare' in order to reduce costs, both at the consumer end, as well as the service provider end. But, he also talks about the various challenges that the proposed system might face, such as the lack of data available for analysis before physicians, healthcare companies and public policy makers can become advocates of such a system. The impact of such a system needs to be properly documented and analysed, according to his observations.

Ilias Maglogiannis et al. proposed a mobile healthcare system that can store patient records on the cloud [3]. They talk about how cloud computing can provide its edge, in terms of the functionality it offers with respect to managing data in a distributed, and portable manner and its ability to interoperate between different platforms. The paper presents a mobile system that stores health information and patient records on the cloud, thus enabling easy data storage and retrieval. The mobile application was built on the Android operating system, and the cloud platform used for evaluation was Amazon S3. The system allows proper documentation of patient records, medical images and presents the initial results that it has achieved.

Yonglin Ren et al. talk about the use of mobile healthcare through wearable devices [4]. This enables powerful surveillance and increased convenience for patients as well as doctors, nurses hospitals and policy makers. One of the challenges faced by this system is the difficulty in providing round-the-clock services for patients who require it. Privacy is another major concern, as many patients are reluctant to present their health data in a form that can be easily accessed by people without their knowledge. As a consequence, making sure that the channels are made secure and data confidentiality is maintained becomes a necessity. The article presents methods to enable efficient electronic patient monitoring and improve the functionality of telemedicine systems, along with solving the problem of privacy by enhancing the security measure that are taken.

E. Iadanza et al. present their application "Careggi Smart Hospital" which was made for Careggi Hospital in Florence [5]. The application was built on the Android platform and enables storage of patient and hospital information, and also allows access to regional health information for research purposes.

2. DESIGN

Design is viewed as a considerably more innovative process than investigation. Design is the initial phase in the improvement of any item or framework. Design can be characterized as "the process of applying different standards and methods with the end goal of characterizing a procedure, a gadget or a framework in detail which will be adequate to allow its physical acknowledgment".

A. Database Design

The Entity Framework Database was designed using the Code First Methodology. Code-First is mainly useful in Domain Driven Design. With the Code-First approach, you can focus on the domain design and start creating classes as per your domain requirement rather than design your database first and then create the classes which match your database design. Code-First APIs will create the database on the fly based on your entity classes and configuration.

We first start by writing C# or VB.net classes and context class. When you run the application, Code First APIs will create the new database (if it does not exist yet) and map your classes with the database using default code-first conventions. You can also configure your domain classes to override default conventions to map with database tables using Data Annotation attributes or fluent API.

Fig-1: Codefirst Workflow

2. IMPLEMENTATION

A user patient comes to the mobile health application with his health-related query to seek an additional diagnosis on his first level diagnostic tests. He/She authenticates to the system using a social login which provides their general personal information like age, gender etc. It creates a case then chooses the specialty depending on their condition. In the next step they choose the hospital based on the chosen specialty, the patient is made to fill up relevant illness related details (known as History Taking) Any documents pertaining to the case are uploaded to be reviewed by the doctor. After the details are filled the case is submitted This is where the TAD Web Server kicks in. The case is allotted to a doctor of the requested specialty in the given hospital based on availability and past performance of the doctor. The doctor gets a push notification informing him of the arrival of a new case At this point, the doctor can review the case details and choose to take up the case, or decline it or if he feels it must be forwarded to another specialty he makes a recommendation to do so. In case the patient choose to accept the case, he can further review the case details and engage with the patient to gather more information and conduct his/her own investigation and make recommendations to the patient. In case the patient rejects the case, the case goes back to the system to reassign to the next available doctor. During the course of the case the doctor may choose to have a conversation with the patient or request for additional documents or details. Once the doctor has completed the investigation he can give the final verdict and choose to close the case.

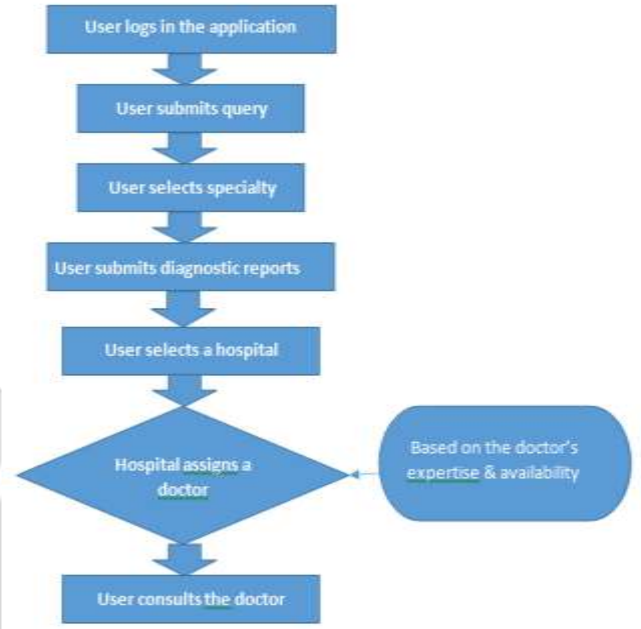

Fig-2: Workflow of TAD Android Application

3. RESULTS

B. Patient Application

- **Login Activity**

Fig-3: The first three screens of the application which serves as the Sign in/Sign up Screens

In these screens, what essentially happening is, Using the Google API's, the user authenticates to the Google Server which returns with a validation token. This validation token is sent as a GET request to the TAD backend server to validate for its authenticity and return an access token which can be used by the application to make the subsequent authorized server calls. At this point the google validation token is also used to authenticate to the Firebase System. Firebase is responsible for providing the capabilities of a NoSql Real time database and Push Notifications which up to now was being handled via Google Cloud Messenger. The user is also registered on to Azure Notification Hubs. Azure Notification Hub is a cloud service offering by Azure which links a .NET backed with the Firebase Cloud Messaging Service to provide a streamlined mechanism to send push notifications to a device.

- **Landing screen**

Fig- 4. Landing screen

Figure 4 is the main landing page of the patient. He/she can choose to create a new case here or check the status of the previous cases. Any status updates on any cases is visible here.

- **New Case Creation Activity**

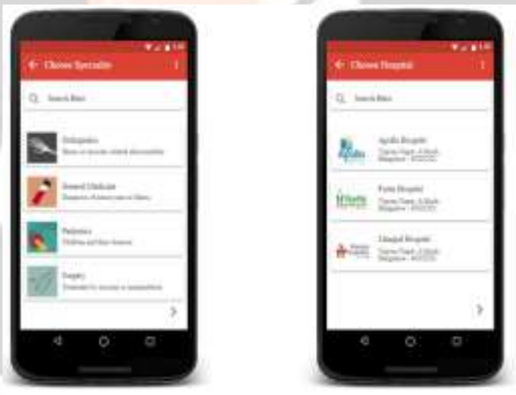

Fig-5: New case creation Activity

The user selects the relevant specialty. He can choose to enter the specialty he needs in the search bar or directly select them. If he isn't aware of his specialty, he can also choose to enter a few keywords pertaining to his illness and the system suggest a specialty. This suggestion is based on recommendations provided by a Naïve Bayes Classifier. Once the Specialty is chosen hospitals catering to those hospitals are populated. The ordering is dependent on the proximity of the hospitals to the user and their ratings in the recent past. Once the Hospital is chosen, the user is directed to a Specialty specific History Taking. The History Taking forms one of the most vital parts of the application as this automates the process of collecting the relevant and pertinent information for the doctor in a highly-structured manner. The History Taking Activity has been designed to be highly flexible and scalable recognizing the fact that multiple conditions can follow a similar line of inquiry, so instead of individually designing layouts and writing logic for all of them, History taking was further divided into Clusters of information being collected. Further, under every cluster there were Fields added, a field refers to a single query. With this fluid structure in place it gets incredibly easy to scale the application to cater to other Specialties without requiring much dev effort. Each query which is asked in the due course of history taking has been added by referring to doctors in the particular field and based on standard medical practices choosing questions which minimize form filling yet not compromise on the data collected. All the fields and clusters that are to be queried for a given specialty are dynamic in nature and therefore are queried at the server.

Fig-6: History Taking for General Medicine

- **Ongoing Case**

Figure 7. Details and Discussion Views

In figure 7, the ongoing case view can be seen. It comprises of the Details tab which shows all the details and attachments submitted. The Discussion chat which is a chat session controlled by the doctor to gain additional information. The chat functionality has been built from ground up into the system by relying on the NoSql offering by Google: Firebase. Firebase is a real-time database which stores data in Json form. It follows a tree structure, to gain access to information to the database, it's possible to attach Event Listeners. These Event Listeners are also triggered when there is any change in the information of the node. The use of such a database allows the messages to always be in a perfect sync without polling continuously for changes.

C. Doctor Application

- **Landing Page**

Fig-8: Landing Page for Doctors' Application

In figure 8, it gives an overview of all the current cases a doctor is handling and the new cases that are available. For every new case the doctor is provided with three choices: 1)Accept the case 2)Refer the doctor to another specialty,3)Decline the case. For every ongoing case the doctor can do the following: 1)Give his final verdict and close the case,2)Refer the case to another Specialty,3)Engage with the patient and ask for details and suggest a plan of action.

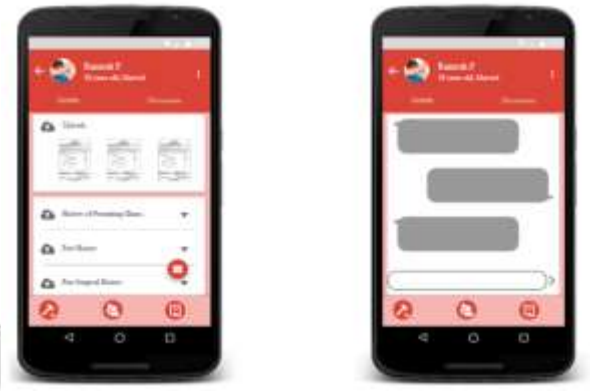

Fig-9: Case View

4. CONCLUSION

At the time of making of this report, the structure of the entire back end was up and running. The primary functionality of the android application had been built in. The beta testing of the application is on schedule to begin around the middle of May. To test it out with a test audience, to get feedback and make improvements and changes to the interface and structure of the platform.

Future Work

With such shortage of doctors, this project has a future vision of shrinking this ratio by bridging the gap between super specialty hospitals and patients. We achieve this by making expert opinions available for users at a minimal cost and in a timely manner. The system should be able to efficiently utilize the case history of existing users in order to recommend additional health assessments if required for a patient and detect a health concern at a very early stage. Our future aspiration is to use Machine Learning with TAD platform to analyze historical data in order to find recurring patterns that can be used for earlier detection of critical ailments such as cancer.

5. ACKNOWLEDGEMENT

I would like to express my sincere thanks to Prof. Sreelatha R, Asst. Prof., Dept. of ISE, BMSCE, for her constant support and encouragement to complete this project.

6. REFERENCES

1. Qiang CZ, Yamamichi M, Hausman V, Altman DG. Mobile applications for the health sector. 2011. https://pdfs.semanticscholar.org/ac01/b885bdc1f6d9e533afa3b96c27c92cfe6821.pdf

2. Mahmud K, Lenz J. The personal telemedicine system. A new tool for the delivery of health. DOI:10.1177/ 1357633X9500100307

3. Charalampos Doukas, Thomas Pliakas and Ilias Maglogiannis, Mobile Healthcare Information Management utilizing Cloud Computing and Android OS. Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 2010.

4. Yonglin Ren, Richard Werner Nelem Pazzi, Azzedine Boukerche: Monitoring patients via a secure and mobile healthcare system, IEEE Wireless Communications, 2010.

5. A. Luschi, A. Belardinelli, L. Marzi, F. Frosini, R. Miniati, E. Iadanza, Careggi Smart Hospital: a mobile app for patients, citizens and healthcare staff, IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI), 2014.