



Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.

TELE-MEDICAL SYSTEM FOR REMOTE MONITORING OF PATIENTS WITH COVID 19 AND OTHER INFECTIOUS DISEASES

Mag. Eng. Angel B. Ivanov*, Mag. Kocho Hrisafov*, Assoc. Prof. Nayden Chivarov*,
Mag. Eng. Stefan Chivarov***, Prof. Ivana Budinska ***

*Institute of Information and Communication Technologies at the Bulgarian Academy of Science;

** Intelligent Handling devices and Robotics – IHRT, TU VIENNA;

*** Institute of Informatics, Slovak Academy of Sciences;

e-mails: angel.ivanov@iict.bas.bg, kocho.hrisafov@iict.bas.bg, nchivarov@gmail.com,
schivarov@gmail.com, budinska@savba.sk

Abstract: The COVID-19 virus has been highly infectious and one of the best strategies to contain the rapid progression is through social distancing. This pandemic gave an extreme rise in popularity and adoption of telemedicine systems. These systems can provide seamless patient monitoring while protecting the health workers from the disease. Full commercial patient monitoring systems can easily become costly because of the closed “vendor-specific” hardware and supported software. The goal of this paper is to prove that with combination of free and open-source software products in combination of already available on the market hardware can be created good telemedicine platform.

Copyright © 2021 The Authors. This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0/>)

Keywords: covid-19, telemedicine, cyber-physical system, internet of things, remote patient monitoring, open-source

INTRODUCTION

The novel Corona virus pandemic has thrown the world in chaos. Given that the virus is extremely contagious, the world’s economy was deeply shaken. Countries were locked down, factories were shut down and the hospitals were not able to keep up with the rising numbers of infected patients. These cataclysmic events led to the critical need of preserving people’s life in the most efficient way possible. It is becoming critical to use modern technology to bring medical care to patients while keeping the medical staff less exposed to the disease and preserving personal protective equipment. As discovered in [1], the telemedicine has catapulted during the COVID-19 pandemic as having critical role of providing essential medical service to patients and helping mitigate the spread of the disease. Telemedicine can supplement the health-care delivery when facing the absence of in-person visits [2]. The Indian government has launched a national teleconsultation service, which is being adopted by many state governments as mandatory for healthcare providers. Five factors which can affect the adoption of telemedicine: relative advantage, compatibility, trialability, observability and complexity have been identified in [3]. In this research, the author tested all five hypothesis and found that patients are more willing to try telemedicine than not. The patients found huge part of their motivation in the reduced cost of the overall medical service and reduced need of travel. Telemedicine is a technology which allow convenience, lower costs and ease of access to health-related information and communication using the Internet or other associated technologies [4]. The telemedicine is also referred to as the first line of defence for the health workers.

Currently are existing excellent commercial telemonitoring systems. [5,6,7].

The downside to all of these systems is that they are locked to the specified vendor of hardware and services. This can make the system implementation extremely costly, requiring new expensive equipment, software and implementation to be bought. In the developing countries, the hospitals might not have the budget to be able to afford such systems.

The goal of this paper is to prove that using free and open-source software products in conjunction with cheap hardware already generally available on the market, we can create fully functional telemonitoring system, suitable for hospitals with limited budget and/or home use in the cases of home quarantine.

2. PROOF OF CONCEPT

In fig.1 we are showing an example workflow on how the telemonitoring system will gather its data. For the sake of simplicity, in the scope of this paper we will use only the digital stethoscope as external hardware.



Fig. 1. Tele-medical system workflow for remote patient monitoring.

The workflow consists of the following parts: digital stethoscope, Bluetooth Low Energy collector, MQTT Publisher and Subscriber, Database and FrontEnd GUI.

2.1 Digital Stethoscope

For the purpose of this paper, we are using digital stethoscope from the brand Eko [8]. It has Bluetooth LE module connectivity which we utilize in the automatic lung screening. Main advantages are the low cost and extreme ease of use, superb performance and compatibility with all major acoustic stethoscope providers.

2.2 BLE collector and data transfer

In order to capture the data from the digital stethoscope and for the purpose of this proof of concept, we use a Raspberry Pi minicomputer [9]. The built in BLE 4.2 module is perfect for the connectivity with the EKO stethoscope. The main advantage of the Raspberry Pi is the ability to run native Linux on its ARM processor. Using the Linux OS, we install a message broker service called Mosquitto [10]. The Mosquitto message broker uses the MQTT message queue protocol, which is fast and reliable. It supports natively both text and binary data transfer. The .wav files produced by the digital stethoscope are being transferred as a binary stream.

The MQTT subscriber receives the binary data from the stethoscope measurements, sent over by the publisher. It will convert the binary stream into a .wav file and then begins the loading process into the database. As described in the next chapter, the database we use for the proof of concept is Microsoft SQL Server 2019 Express. In the Automated Interval Examination table shown in figure 4, a special field was created for the stethoscope recordings. The type of this field is varbinary. This ensures that the wav file from the stethoscope recording is compatible and will be stored in the database. The actual data load is done using MS OLE DB driver for MS SQL Server.

The automated measurements are taken every 30 minutes with recording time set to 15 seconds. The recording time is configurable and can be raised or decreased if the situation requires such changes.

Since this is telemedicine system, we are thinking also about the patient and medic cyber security. In order to protect the communication and prevent leaks of personal identifiable information the following measures were taken:

Vpn server was setup in order to connect both the mqtt publisher and subscriber. Firewall rules are created on both devices in order to filter any unwanted traffic. To further minimize the chance of information leak, ssl certificate is being used to encrypt the communication between the publisher and subscriber.

2.3 Storing the data

The data is stored in Microsoft SQL Server 2019 Express relational database [11]. It is a free edition of SQL Server, ideal

for development and production for desktop, web, and small server applications.

Fig. 2 shows all table and relations defined in the database in a diagram:

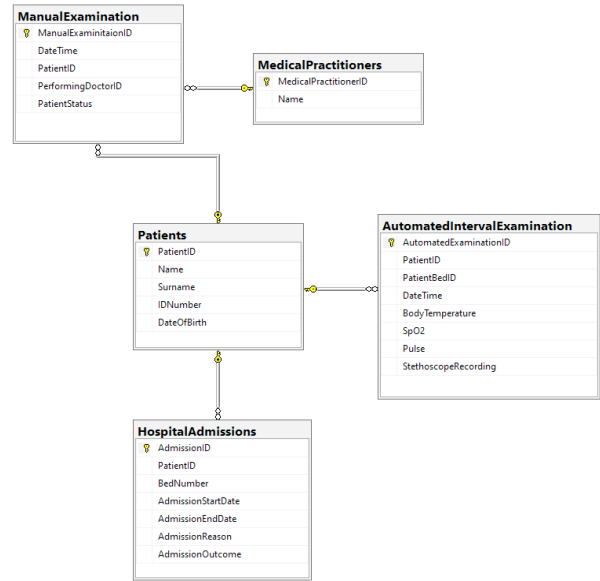


Fig. 2. Tables and relations in the database.

The table showed in fig. 2 contains the general information for the patients.

- Unique key is **PatientID**
- **PatientId** field contains ID number of the patient
- **Name** and **Surname** field contains patient’s name.
- **DateOfBirth** field contains the birth date of the patient

Its data is filled by human operators when entering new patients or modifying patient’s data (fig. 3).

Column Name	Data Type	Allow Nulls
PatientID	int	<input type="checkbox"/>
Name	nvarchar(50)	<input type="checkbox"/>
Surname	nvarchar(50)	<input type="checkbox"/>
IDNumber	int	<input type="checkbox"/>
DateOfBirth	smalldatetime	<input type="checkbox"/>

Fig. 3. Patients table

Table showed in Fig.4 contains hospital admissions data. It is entered by human operators.

- Unique key is **AdmissionID**

- Field **PatientID** contains patient ID and used to form a relationship to **PatientID** in Patients Table
- **BedNumber** field contains number of the hospital bed which is related to the corresponding measurement devices linked to that bed (pulse oximeter, stethoscope etc.) so the data collected by those devices will be linked to the correct patient.
- Fields **Admission Start Date** and **Admission End Date** contains dates when the patient was admitted and released from hospital
- Field **AdmissionReason** describes why the patient was admitted to hospital
- Field **AdmissionOutcome** describes in which state the patient was released from hospital

- Field **PatientBedID** contains data related to location of the acquired data so it can be linked with patient and bed location
- Field **DateTime** contains data when the measurements were taken
- Field **SpO2** contains data from the SpO2 sensor
- Field **Pulse** contains data from the patient heartbeat rate
- Field **StethoscopeRecording** contains sound file from the stethoscope

Column Name	Data Type	Allow Nulls
AdmissionID	int	<input type="checkbox"/>
PatientID	int	<input type="checkbox"/>
BedNumber	int	<input type="checkbox"/>
AdmissionStartDate	smalldatetime	<input type="checkbox"/>
AdmissionEndDate	smalldatetime	<input checked="" type="checkbox"/>
AdmissionReason	ntext	<input type="checkbox"/>
AdmissionOutcome	ntext	<input checked="" type="checkbox"/>

Fig. 4. Hospital Admissions table

Table showed in Fig.5 contains automated examination data. It is entered automatically on an interval basis by mqtt subscriber. Although this paper covers digital stethoscope only – the database is designed for full range of medical monitoring devices.

Column Name	Data Type	Allow Nulls
AutomatedExaminationID	int	<input type="checkbox"/>
PatientID	int	<input type="checkbox"/>
PatientBedID	int	<input type="checkbox"/>
DateTime	smalldatetime	<input type="checkbox"/>
BodyTemperature	decimal(2, 2)	<input type="checkbox"/>
SpO2	int	<input type="checkbox"/>
Pulse	int	<input type="checkbox"/>
StethoscopeRecording	varbinary(MAX)	<input checked="" type="checkbox"/>

Fig. 5. Automated Interval Examination table

- Unique key is AutomatedExaminationID
- Field **PatientID** contains patient ID and used to form a relationship to **PatientID** in Patients Table

Table showed in Fig.6 contains data collected by manual examination of the patients by medical personnel. It is entered manually.

- Unique key is ManualExaminationID
- Field **Datetime** contains date and time when the examination was performed
- Field **PatientID** contains patient ID and used to form a relationship to **PatientID** in Patients Table
- Field **PerformingDoctorID** contains doctor ID and used to form a relationship to **MedicalPractitionerID** in MedicalPractitioners Table
- Field **PatientStatus** contains data collected as a result of the medical examination

Column Name	Data Type	Allow Nulls
ManualExaminationID	int	<input type="checkbox"/>
DateTime	smalldatetime	<input type="checkbox"/>
PatientID	int	<input type="checkbox"/>
PerformingDoctorID	int	<input type="checkbox"/>
PatientStatus	ntext	<input type="checkbox"/>

Fig. 6. Manual Examination table

Table showed in Fig.7 contains data for medical personnel tending to patients.

Unique key is MedicalPractitionerID

Field **Name** contains the name of the medical practitioner

Column Name	Data Type	Allow Nulls
MedicalPractitionerID	int	<input type="checkbox"/>
Name	nvarchar(50)	<input type="checkbox"/>

Fig. 7. Medical Practitioners table

3 FRONTEND AND GRFICAL USER INTERFACE

Frontend is developed in Microsoft Visual Studio Community Edition 2019 using ASP.NET C# [12, 13]. It is a fully featured, extensible, free IDE for creating modern applications for Android, iOS, Windows, as well as web applications and cloud services. .NET is an open-source developer platform with no licensing costs and free development tools for Linux, macOS, and Windows.

The control centre of the telemonitoring solution is easy to use WEB GUI frontend.

It is designed for non-technical medical personnel.

It's developed in a client – server architecture.

It can be deployed in two scenarios depending on the resources and IT capabilities of the medical institution:

- hybrid cloud solution – backend (mqtt publisher and subscriber) runs on an on-premise server located in the medical facility; the SQL database engine and GUI runs in the cloud.
- Fully on-premise solution – all three components run locally.

The GUI frontend is WEB based so the medical personnel can access from different devices (computers, tablets etc.) without the need to install separate client software.

Also, the WEB technology will enable the use of different type of devices without compatibility problems (OS independent).

Its benefit is that medical practitioners to check remotely for patients' vitals without exposing themselves to risks.

It is consisted of several list and entry forms.

Figures 8 shows a web form which enables listing, searching and modifying patient data. Data from the DB table Patients is used.

Name	Surname	Date Of Birth
John	Doe	7/7/1994 12:00 AM
Eric	Doe	1/27/1970 12:00 AM
Jason	Doe	3/4/1971 12:00 AM

Fig. 8. Patients list table

Figures 9 shows a web form which enables inserting and modifying new/existing patients. It adds/modifies records in the Patients DB table.

Fig. 9. Patients' entry

Figures 10 shows a form which lists all the patients' hospitalizations. Data can be filtered and searched by patient name and date. Operators have quick overview of the reason's patients were hospitalized and the outcome of the hospitalization. There is quick link to remote monitoring of the selected patient data. Data is sourced from the Hospital Admissions DB table.

Patient	Bed Number	Admission Start Date	Admission End Date	Admission Outcome	Admission Reason
Eric	1	3/1/2021 12:00 AM			Persistent cough High temperature Pneumonia
Jason	2	4/8/2021 12:00 AM	4/29/2021 12:00 AM	Good overall condition	Pneumonia
John	4	4/21/2021 12:00 AM			Breathing problems

Fig. 10. Hospitalization's list

Figures 11 shows a form which enables to add/modify hospitalizations data. Operators select existing or add a new patient. Date of the admission, hospital bed number where patient is located is entered. An extensive reason for the hospitalization and its outcome is also supplied. Data is inserted/ modified in Admissions DB table.

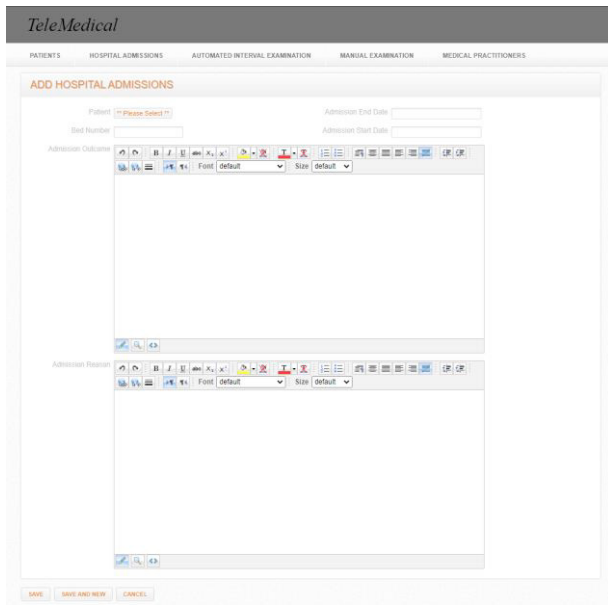


Fig. 11. Hospitalization's entry

Figure 12 shows the form for remote monitoring of the patients' vitals. Medical practitioners can filter data by patient and point in time. They can have an auto refreshing overview of the vitals of several patients on screen or they can drill down into detailed data for single patient. Data is read from Automated Interval Examination DB table.

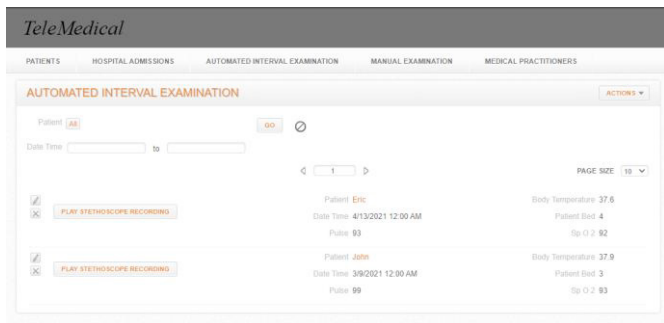


Fig. 12 Remote monitoring of patient's vitals

Figures 13 shows the form for listing of manual examinations. They can be searched/filtered/sorted by patient and date. Data from Manual Examination DB table is used.

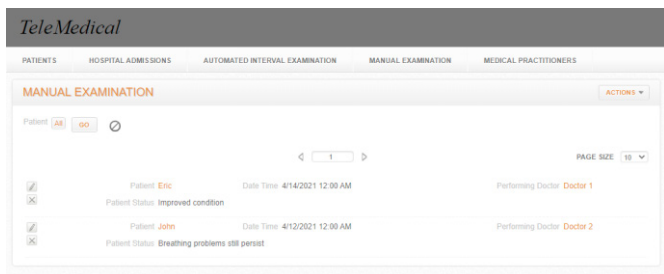


Fig. 13. Manual examination list

Figures 14 shows the form for entry/modify of manual examinations. Patient and performing doctor are selected. Time of the examination is entered. The results from the examination are entered in the last document style field. Data is inserted / modified in Automated Interval Examination DB table.

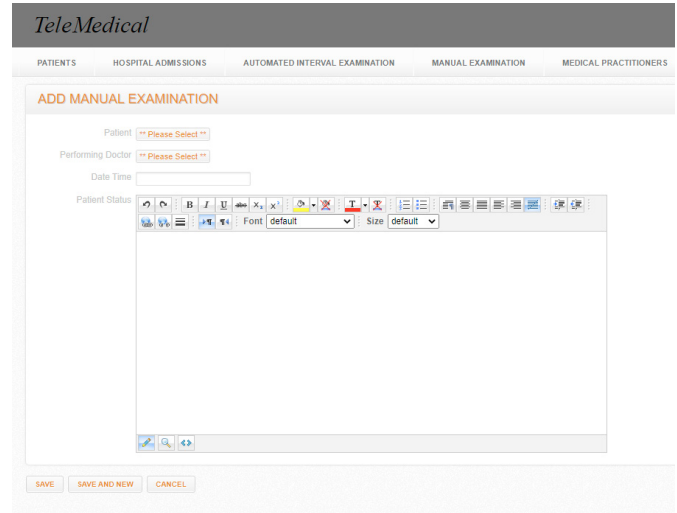


Fig. 14. Manual examination entry

Figures 15 shows the form for listing the medical practitioners. They can be sorted and searched by name. All related medical examinations performed by the selected medical practitioner can be shown. Data from Medical Practitioners DB table is used.

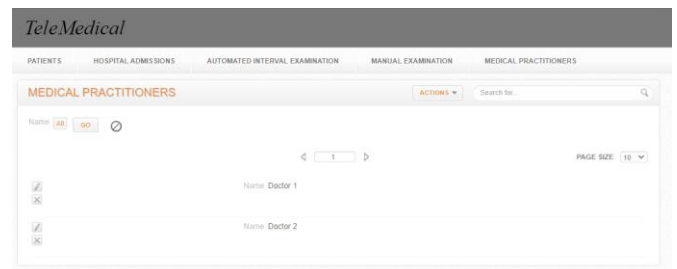


Fig. 15. Medical practitioners list

Figures 16 show the forms for the process of entry / modify of medical practitioners. Data is inserted / modified in Medical Practitioners DB table.

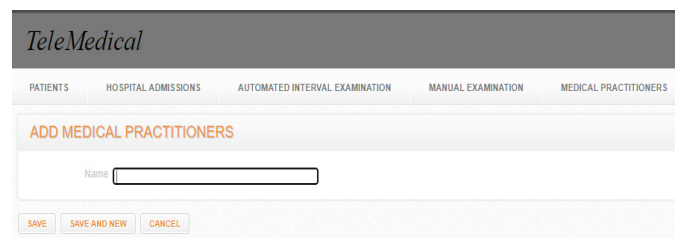


Fig. 16. Medical practitioners entry

4. TESTS AND RESULTS

Experiments were performed with the open-source Respiratory Sound Database, which is using audio recordings for detection of respiratory diseases [14]. Respiratory sounds are important indicators of respiratory health and respiratory disorders. The sound emitted when a person breathes is directly related to the movement of air, changes in lung tissue and the position of secretions in the lungs. Wheezing, for example, is a common sign that a patient has an obstructive airway disease such as asthma or chronic obstructive pulmonary disease (COPD). These sounds can be recorded using digital stethoscopes and another recording device. These figures reveal the possibility of using machine learning to automatically diagnose respiratory disorders such as asthma, pneumonia and bronchiolitis, to name a few. The Respiratory Sound Database [14] was created by two research teams in Portugal and Greece. It includes 920 annotated recordings of varying length - 10s to 90s. These recordings were taken from 126 patients. There is a total of 5.5 hours of recordings containing 6898 respiratory cycles - 1864 contain crackles, 886 contain wheezes and 506 contain both crackles and wheezes. The data includes both clean respiratory sounds as well as noisy recordings that simulate real life conditions. The patients span all age groups - children, adults and the elderly.

In the conducted experiment with the respiratory sound database, developed Tele-Medical System work properly and manage to monitor remotely vital parameters of the patients with Covid-19 and other infectious diseases.

CONCLUSION

The research, presented in this paper has the goal of creating low cost, yet fully functional telemedical system. The target users for this system are low-budget hospitals and developing countries. We have showed that using free and open-source tools, it is possible to achieve this goal. While in this paper we are showing only digital stethoscope, we are actively working on adding support for cheap pulse-oximeter devices and thermometers with BLE support. Future work includes adding AI functionalities to assist patient monitoring and diagnosis.

ACKNOWLEDGMENT

This research was carried out as part of the bilateral BAS-SAS project: “Cyber-Physical System for smart monitoring and tele-medicine for patient with COVID-19”.

REFERENCES

1. Calton, B., Abedini, N. and Fratkin, M. (2020). Telemedicine in the Time of Coronavirus. *Covid-19 articles fast tracked articles*, volume 60, issue 1, e12-e14. Elsevier. Online at [https://www.jpsmjournals.com/article/S0885-3924\(20\)30170-6/fulltext](https://www.jpsmjournals.com/article/S0885-3924(20)30170-6/fulltext)
2. Mahajan, V., Singh, T., and Azad, C. (2020). Using Telemedicine During the COVID-19 Pandemic. *Indian pediatrics*, 57(7), 652–657. Online at

- <https://www.indianpediatrics.net/COVID29.03.2020/SA-00182.pdf>
3. Mishra V. (2020). Factors affecting the adoption of telemedicine during COVID-19. *Indian J Public Health* 2020;64, Suppl S2:234-6, Online at <https://www.ijph.in/text.asp?2020/64/6/234/285604>
4. Vidal-Alaball, J., Acosta-Roja, R., Hernández, N., Luque, U., Morrison, D., Pérez, S., Perez-Llano, J., Vèrges, A., Seguí F. (2020). Telemedicine in the face of the COVID-19 pandemic. *Atención Primaria*. Vol. 52. No. 6., Pages 369-446. Online at <https://www.elsevier.es/es-revista-atencion-primaria-27-articulo-telemedicina-in-face-covid-19-pandemic-S0212656720301268>
5. Phillips (2021). Central patient monitoring systems. Online at <https://www.usa.philips.com/healthcare/solutions/patient-monitoring/central-monitoring-systems>
6. GE Healthcare (2021), Digital Centralized Monitoring Unit. Online at <https://www.gehealthcare.com/products/patient-monitoring/connectivity/digital-centralized-monitoring-unit>
7. Mindray (2021). Mindray BeneVision Central Monitoring System. Online at https://www.mindray.com/en/product/BeneVision_CMS.html
8. Eko (2021). Eko Core Digital Attachment. Online at <https://www.ekohealth.com/product/eko-core-digital-attachment>
9. Raspberry Pi (2021). Raspberry Pi 3 Model B+. Online at <https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/>
10. Mosquitto (2021). Mosquitto MQTT broker. Online at <https://mosquitto.org/download/>
11. Microsoft c (2021). SQL Server downloads. Online at <https://www.microsoft.com/en-us/sql-server/sql-server-downloads>
12. Microsoft b (2021). Visual studio community edition. Online at <https://visualstudio.microsoft.com/vs/community/>
13. Microsoft a (2021). Dot Net programming language. Online at <https://dotnet.microsoft.com/platform/free>
14. Respiratory Sound Database, <https://www.kaggle.com/vbookshelf/respiratory-sound-database>