International Journal of Electronic Healthcare

ISSN online: 1741-8461 - ISSN print: 1741-8453
https://www.inderscience.com/ijeh

Integrated e-healthcare management system using machine learning and flask

Sanjeev Kumar, Jaya Ojha, Mayank Mani Tripathi, Kirtika Garg

DOI: 10.1504/IJEH.2023.10052840

Article History:

Received:	29 June 2021
Accepted:	28 October 2022
Published online:	27 January 2023

Integrated e-healthcare management system using machine learning and flask

Sanjeev Kumar*, Jaya Ojha,
Mayank Mani Tripathi and Kirtika Garg

Department of Computer Science and Engineering,
ABES Institute of Technology,
Ghaziabad, India
Email: sanjeevsavy1@gmail.com
Email: jayaojha73@gmail.com
Email: sismmt09@gmail.com
Email: kirtikagarg2017@gmail.com
*Corresponding author

Abstract: Healthcare is one of the flourishing sectors in each developed and emerging economy. Due to this vast COVID-19 pandemic, the traditional healthcare system cannot provide adequate facilities due to a lack of interactions between doctors and patients. In such conditions, e-healthcare is contributing towards the accelerating growth within the healthcare industry by providing the latest information technology to support information search and communication processes. Besides this, a machine learning algorithm is used to intensify the smartness of the healthcare industry. The five major components of an e-healthcare system are cost-saving, virtual networking, electronic medical record physician-patient relationships and privacy concerns. Our proposed system provides location-based e-prescribing, e-reports, disease prediction, and suggesting treatments and emergency services with a single click, so it is better than another existing system.

Keywords: e-healthcare; location-based; disease prediction; treatments; e-reports; e-prescription.

Reference to this paper should be made as follows: Kumar, S., Ojha, J., Tripathi, M.M. and Garg, K. (2023) 'Integrated e-healthcare management system using machine learning and flask', *Int. J. Electronic Healthcare*, Vol. 13, No. 1, pp.71–90.

Biographical notes: Sanjeev Kumar is an Assistant Professor at the ABESIT Ghaziabad in the Department of Computer Science and Engineering. He received his Master of Technology degree from U.S.I.T. (Guru Govind Singh Inderprastha University Delhi). Currently, he is pursuing his PhD from the IIT (I.S.M.) Dhanbad. His research area is the security of the internet of things and machine learning. He has some research papers in SCI and Scopus journals. These journals are *IJBIC* (Inderscience), *ETT* (Wiley), *IJAIP* (Inderscience), etc.

Jaya Ojha is working as a Software Development Engineer at the Nagarro Software Pvt. Ltd. She received her BTech from the AKTU, Lucknow (ABESIT Ghaziabad) in Computer Science and Engineering.

Mayank Mani Tripathi is working as Senior Software Development Engineer in the SaaS Labs Software Pvt. Ltd. He received his BTech from the AKTU, Lucknow (ABESIT Ghaziabad) in Computer Science and Engineering.

Kirtika Garg is a Java web developer in the Erasmith Technologies Pvt. Ltd. She received her BTech from the AKTU, Lucknow (ABESIT Ghaziabad) in Computer Science and Engineering.

1 Introduction

Advancements in recent years in e-healthcare have led to opportunities for designing and implementing patient-centred e-healthcare systems across small to large-scale medical practices (Jayaratne et al., 2019). Earlier, two people used to visit doctors for regular checkups or some treats and get treated. But over the course of time and conditions, many changes occur. Earlier, people visited doctors, they interacted with doctors physically, but due to the vast COVID-19 pandemic outbreak, this interaction has somewhat lost its hold. We know that more than half of the population of this world is suffering from this vast COVID-19 pandemic. So, people avoid going to the doctor, and many doctors are also not taking new cases or discontinuing some of the previous instances considering the current horrible situation. As this outbreak is highly contagious and deadlier, hence doctors, as well as patients, are also avoiding visiting doctors. And apart from this COVID-19 outbreak, this Bird-flu has also started spreading its arms in our country. This Bird-flu is also very dangerous and deadlier. This avian influenza is mainly reported in birds but can potentially affect humans and other animals. The COVID-19 pandemic has taken the worst phase since March 2020 in India. The total number of people testing positive for the virus stood at 271, with 65 new cases on 21 March 2020, and it has increased to 10.4 M points and 151 K deaths, and worldwide it is 88.9 M cases and 1.91 M deaths. We know this doctor-patient meeting helped doctors to understand the problem more clearly, but this outbreak has made it all a mess. So, now there is a requirement for a proper healthcare system comprising no physical interaction between doctors and patients. Online health communities also allow patients to obtain medical services for various diseases through several online platforms (Wu and Lu, 2017). The internet and online media can resolve or diminish medical resources deficiency, and doctor-patient disputes emerge due to a lack of interaction to a certain extent (Wu and Lu, 2017). Due to its promising benefits, big corporations implement a little eye-opener that serves healthcare systems that have been designed (Guo et al., 2016).

Our proposed system eases the creation, storage, and retrieval of medical information and appropriate treatments for patients at the proper time (Guo et al., 2016). Ensuring the security of patients' electronic health records (EHRs) in the e-healthcare system is a hot research issue worldwide. Existing literature confirms that providing privacy-preserving e-health healthcare with fine-grained data access control in the presence of semi-trusted servers remains a research challenge. Unlike earlier, they had to consult the doctors using cellphones or laptops (PCs), saving the patients they need not wait for their turn. They just need to consult the doctors on the appointment time and get the appropriate treatment, and it also saves the travelling time of patients (Hu et al., 2012). Another work of our system is maintaining patient e-health records, and their appointment corresponding to related specialised doctors. It also aims to predict disease based on the symptoms and suggest medicines for common diseases. Implementing machine learning (ML) in the medical field reflects promising opportunities for considerably improving the healthcare system (Dalal, 2020). ML is revolutionising the medical industry by giving it a helping hand and intensifying its smartness. This powerful subset of artificial intelligence appears familiar in several three cases like personalised treatments, drug discovery, cancer detection, lung disease prediction, robotic surgery and many more. Here we have used this strong field of AI for disease prediction. Various algorithms can be developed specifically for their independent functions across the medical industry. Algorithms such as K-NN, Naïve Bayes, decision trees, and random forests classifiers can be used to for this purpose. For making doctors-patients interact efficiently through an e-healthcare system, location-based access to hospitals is a better way so that person can contact the known door and get treatment. With the vogue of powerful, location-enabled mobile devices, user interest in swiftly locating information about nearby places of interest has fattened. For example, identifying a mobile device's geographical location device that takes user input, such as by using global positioning system information. A server system uses this information to perform a 'local' search, wherein search outcomes near the device's reported location are preferred over the other works. Such techniques may be used to return the names of restaurants around a user; let's say the user submits 'continental restaurants' as a query on mobile (Cheng and Von Behren, 2013). A similar approach is applied here to locate nearby hospitals for booking virtual appointments or accessing emergency services.

The layout of our article is as follows; the working of our system is described in Section 2. Section 3 demonstrates the result and discussion of the proposed approach. Section 4 discusses a comparative study of our submitted article with existing articles. In Section 5, the future scope of the article is provided. The conclusion of this paper is mentioned in Section 6.

2 Working

The proposed system provides all the required facilities to the patient and doctor so that they can interact while sitting in the privacy of their cubicles (Guo et al., 2016; Mukherjee and McGinnis, 2007). As we all are aware, we all know the world is affected by different viruses, so everyone is very possessive about their health. Next, they fear clinic and hospital patients who might have been affected. So, our proposed system helps patients to get advice and medicines prescribed by the doctor without moving out (Sahi et al., 2017). After careful analysis, the proposed approach can be divided into the following modules:

- patient module
- doctor module
- billing module
- disease prediction
- suggesting medicines for diseases.

2.1 Patient module

This module benefits patients and doctors, as a doctor can also treat patients in case of a particular disease. This model is one of the essential modules because it allows the patient to search about an illness to get suggestions for medicines regarding some common conditions and help us to get some treatments in our home against diseases. Patients can also access location-based hospitals to fix appointments with doctors and get medical advice and e-prescriptions at their homes (Guo et al., 2016; Das and Alam, 2014; Ilorah et al., 2017; Cheng and Von Behren, 2013; Ahlers and Boll, 2007; Ortega et al., 2009). They can fix an appointment and take the advice the doctors who are far away (Malik et al., 2017; Wu and Lu, 2017; Hu et al., 2012; Yin et al., 2019; Cola and Valean, 2015). Doctors can check to pick up the patient using video, and patients can easily view the e-prescription prescribed by the doctor with a single click (Cola and Valean, 2015). Patients can easily avail of the emergency service of an ambulance with a single click to save in an emergency. The patient does not need to log in or stop to avail of the service (Guo et al., 2016; Das and Alam, 2014; Ilorah et al., 2017; Iqbal et al., 2018). The patient can also view their e-report time and send them to doctors for future treatments. The proposed system provides a speech recognition facility. It performs tasks by using listening commands for people in rural areas or people who do not know much about using websites and applications. In this way, they can avail of services from their homes.

2.2 Doctor module

In this module, the patient can avail appointment facility. The proposed module approved the patient's appointment request and provided the communication details, such as contact details and a Google Meet link for a video check-up (Wu and Lu, 2017; Cola and Valean, 2015). In this module, the doctor prescribes the medicine for the respective disease and proper planned dosage information regarding medications. After receiving the prescription of payment, the doctor updates the payment status so that patient can view the e-prescription and then follow the dosage plan as advised by the doctor.

2.3 Billing module

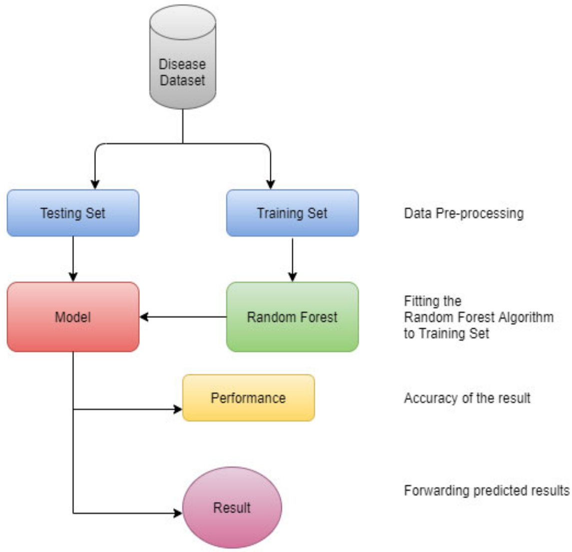
This module enables the patient to pay the bill for the doctor's advice and prescription and the security of both doctor and patient. Once the prescription is ready, the patient will receive a notification email on the proposed system. The patient can view the prescription when the doctor gets the payment (Liu and Park, 2012).

2.4 Disease prediction using ML

This module predicts diseases where the patient can check whether any disease infects him (Hussein et al., 2012; Beri, 2019; Dortmund and Friedman, 1989). ML algorithm random forest is used to predict diseases, as shown in Figure 1. The random forest algorithm is implemented as

- it is capable of handling large as well as high-dimensionality datasets
- it provides high accuracy and prevents overfitting issues
- it is pretty much capable of handling the underfitting case.

Figure 1 Working procedure of random forest algorithm (see online version for colours)



We used Python to implement the algorithm. Now the steps involved in the implementation are as

1 Data pre-processing step:
   - The scraped symptoms are pre-processed to remove similar symptoms with different names (for example, headache and pain in the forehead).
2 Fitting the random forest algorithm to the training set:
   - It constructs decision trees for every sample dataset chosen, predicting the test results.
   - Here, every predicted result will be compiled and voted on.
   - At last, the most voted prediction will be selected and be presented as a result of the classification
3 Test the accuracy of the results:
   - In this project, we have used a random forest classifier with 100 random samples, giving the result with ~95% accuracy.
4 Forwarding predicted result to front:
   - After the predicted results are achieved, the API embeds it into the response for the user.

2.4.1 Implementation of disease prediction

A CSV document containing diseases and symptoms is used to prepare the model depicted in Figure 2. There is a function called Read_csv() function which is utilised to store information in the data frame. Utilising replace() function and the column that is different diseases is replaced by the numbers from 0 to n – 1, where n is the number of conditions present in the .csv document.

Figure 2 Training set (see online version for colours)

Out[179]:

	itching	skin_rash	nodal_skin_eruptions	continuous_sneezing	shivering	chills
0	1	1	1	0	0	0
1	0	1	1	0	0	0
2	1	0	1	0	0	0
3	1	1	0	0	0	0
4	1	1	1	0	0	0

5 rows × 133 columns

This is the output produced, containing the initial five data frame rows. There is a function called plot per column distribution() which is used to build the distribution graph of the columns of the .csv file. This is the output for the distribution graph of the columns of the .csv file. To build the precision of the model, we used random forest algorithm.

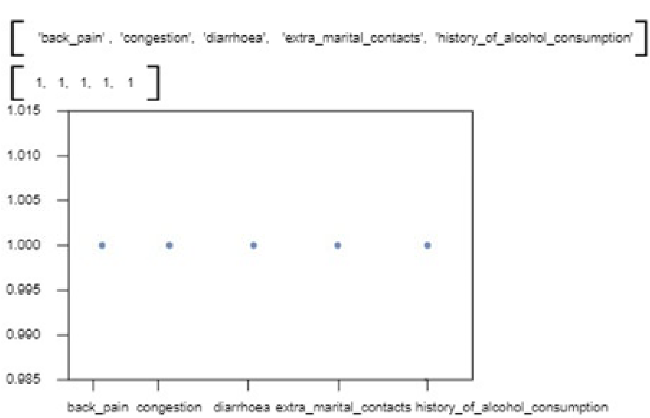
The definition of the random forest() function, ‘pred2’, is used to store the prediction disease using a random forest classifier mentioned in Figure 3.

Figure 3 Random forest function (see online version for colours)

```
pred2=StringVar()
def randomforest():
```

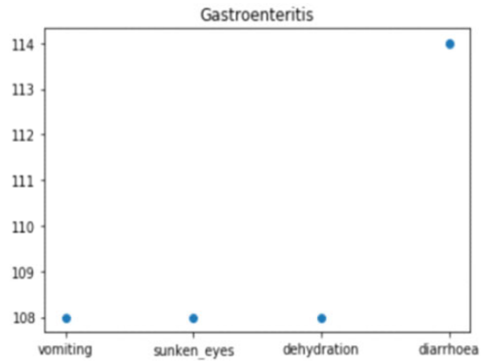
Random forest classifier() is used to train the model and predict the disease on the testing dataset according to symptoms provided by the user. The final disease for the random forest is stored in a variable named ‘pred2’. The accuracy of predicting the disease is calculated using accuracy_score, and the confusion matrix is created using confusion_matrix, which is imported from sklearn matrices. The functions scatterplot (disease) and scattering (sym1, sym2, sym3, sym4 and sym5) are created to plot the predictions of the diseases and the symptoms entered by the user.

Figure 4 Scatterplot for symptoms (see online version for colours)



The scatterplot for the symptoms that the user selects as input is explained in Figure 4.

Figure 5 Scatterplot for disease (see online version for colours)



This is the scatterplot of the diseases predicted based on the symptoms provided by the user as inputs are shown in Figure 5. The system takes symptoms as inputs from the user and provides the best possible result of whether a person is infected with a disease and, if yes, which disease. It is just a suggestion, and the patient can later book an appointment with the doctor and get proper treatment.

2.5 Suggesting medicines for diseases

This system also suggests some treatments for the diseases. The user just provides the name of the disease, and all the possible medicines and treatments related to that disease will be displayed; it will help the patient to know more about diseases and their treatments and can help them to treat themselves if they get some acute diseases like headache, fever, cold and many more which can be treated at home also (Ceyhan et al., 2017; Goyal et al., 2020; Hussein et al., 2012; Chen et al., 2010).

2.5.1 Working methodology

1 *Signup:* Users must sign up with all their email IDs, AADHAR CARD and other essential documents and details, as shown in Figure 6.

2 *Login:* Once users register on this portal, they need to log into our system using their credentials to access the services.

3 *Access the services:*
   - Patient: He can access the following services
   - Enter the disease: He can enter the disease to know the medicines for the illness.
   - Enter the symptoms: Patient can enter the symptoms of disease or symptoms of sickness, and the system will help them in the prediction of the disease if the patient is satisfied with the prediction of the system, then it can ask for the medicine from our system, but if not, then it may consult to the specialised and experienced doctors for the check-up from our system.

- Contact the respective doctor: Patients can opt for this option for direct contact with doctors regarding their disease. In this option, the Patient needs to send a virtual request to the respective doctor for a virtual meeting/to get the appointment. If an appointment gets fixed by the doctor, they can have a virtual, whose related information will be shared to the mode of contact, either SMS or email address.
- Doctor: Doctors receive appointment requests from patients. The doctor decides whether he has time for that patient on the day. If the appointment request is accepted, then the notification of acceptance and other details of the meeting will be sent to the patient, and after that payment, the procedure is done.

But if not accepted, notification of declined request is sent to the patient so that they must be aware of the declining request and may contact the doctor on some other day or can contact the other doctors from the system shown in Figure 7.

Figure 6 Working methodology (see online version for colours)

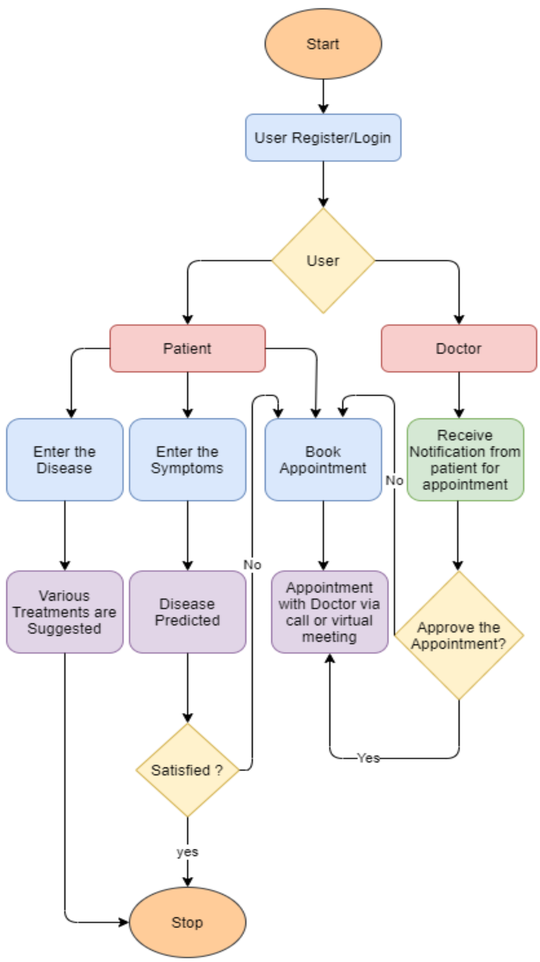
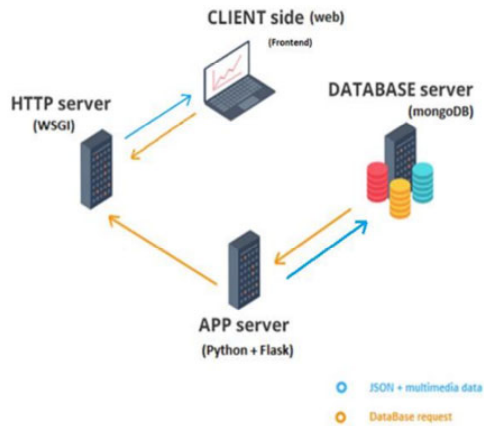


Figure 7 Data flow through various components (see online version for colours)



2.5.2 Frontend

We have created the front end of our project using HTML, CSS, and JavaScript, and we have tried to keep it very user-friendly and simple so that every person can interact with the system without any difficulties.

HTML or Hyper Text Markup Language is the standard markup language and is also referred to as the backbone of any website development process and is used to design documents to be displayed on web browsers. HTML is an acronym where hypertext indicates that text has linked embedded while the markup language shows that text can be converted to images, tables, links, and many more. With the help of HTML, the code site acquires its overall framework means how it will look.

CSS or cascading style sheet, controls the site's overall presentation and allows it to have its distinctive look by maintaining style sheets that lie on top of all other style rules. It can be triggered depending on inputs like device screen size and resolution.

JS is an acronym for JavaScript, an event-based imperative programming language that helps change a static HTML page to a dynamic interface. It uses document object model (DOM), delivered by HTML standard, to mould a web page in response to events such as user input.

2.5.3 Python + Flask

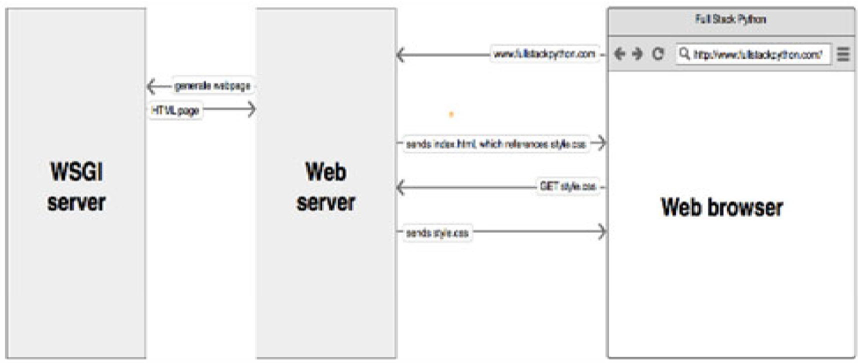
We have used Python + Flask to create APIs to retrieve data from the front end, send the required response to the front end, and process data. Also used for training data for ML algorithms, i.e., random forest, as discussed earlier for disease prediction. Python is an object-oriented, interpreted, high level and interactive programming language. It is incredibly readable and uses simple English keywords with fewer syntactical constructions than any other programming language. Its language constructs and object-oriented approach also help programmers quickly write clear and logical code for small and large-scale projects.

Flask is a famous Python web framework, which means that it is a third-party Python library utilised for developing web applications. Flask is based on the WSGI toolkit and Jinja2 template engine where both are Pocco projects.

2.5.4 WSGI

Any Python module needs web server gateway interface (WSGI) to get executed on the server, as shown in Figure 8.

Figure 8 Working of WSGI (see online version for colours)



WSGI acts as an interface between the Python raw code (business logic implemented over Python) and the standard Apache server. WSGI is implemented by modifying the core apache to increase the flexibility of the server and make the server cross-compatible with various frameworks and different scripting languages.

The business logic (generally implemented over Python) is used to process the data/ request over the top of WSGI, which then returns the rendered HTML pages.

It is a separately running container on the web server over a different port. The Apache server is configured to pass the HTTP request to the WSGI container.

2.5.5 MongoDB

MongoDB stores data in flexible JSON-like documents where fields differ depending on the documents and data structure can be changed when required. The document model maps the object in your code, making it easy to work with data. It is a distributed database at its core enterprise server. It is a source-available and cross-platform database program classified as a NoSQL program.

The main features of MongoDB are:

- ad hoc queries
- indexing
- replication
- load balancing
- file storage
- aggregation
- server-side JavaScript execution
- capped collections
- transactions.

Here, ad hoc queries, indexing, and aggregation facilitate full data access and analysis processes.

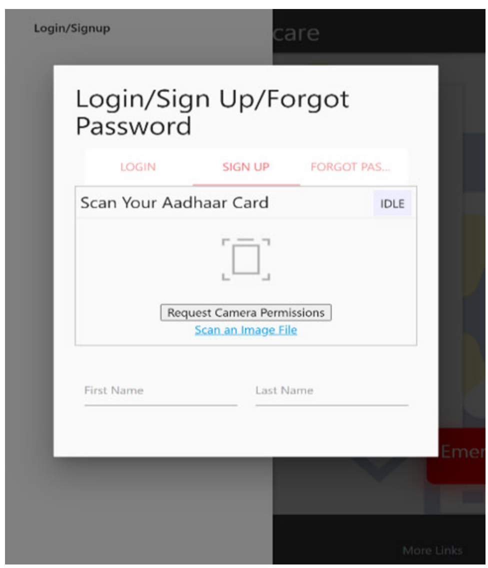
3 Results and discussion

The output of the e-healthcare management system shows how the system is working and what the outcomes look homogeneous to; below are some snapshots of the code, Aadhaar card scan for sign up, patient's side such as patient's dashboard, location-predicated access to hospitals, booking an appointment, emergency accommodations, health records, payment portal as well as doctor's side such as appointment corroborations and prescribing medicines.

3.1 Aadhar card scan

It is used to sign up immediately by scanning your Aadhar card, as depicted in Figure 9.

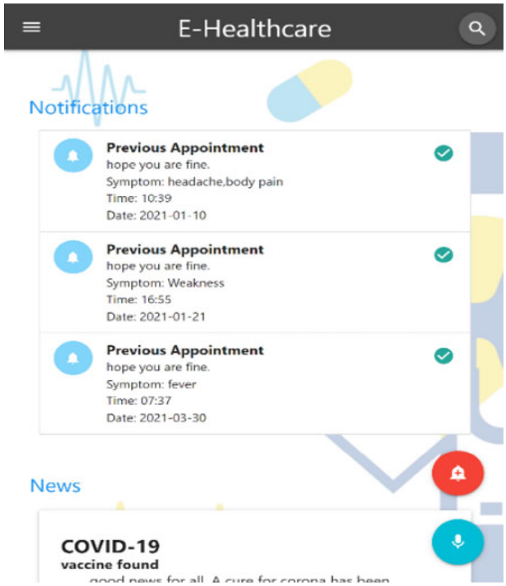
Figure 9 Aadhaar card scan (see online version for colours)



3.2 Patient's dashboard

It consists of notifications regarding prescription forms, appointments booked, payments, and news cognates to sundry medical departments shown in Figure 10. It additionally includes sundry functionalities such as edit/view profile, search for medicos or medicines, disease prognostication portal link, and link to doctor's portal in case of the patient is withal a medico.

Figure 10 Dashboard (see online version for colours)



3.3 *Making an appointment request*

Figure 11 represents the form which allows patients to book an appointment with the medicos depicted in Figure 11. They can book their appointments on a date and time congruous to them.

Figure 11 Appointment request (see online version for colours)

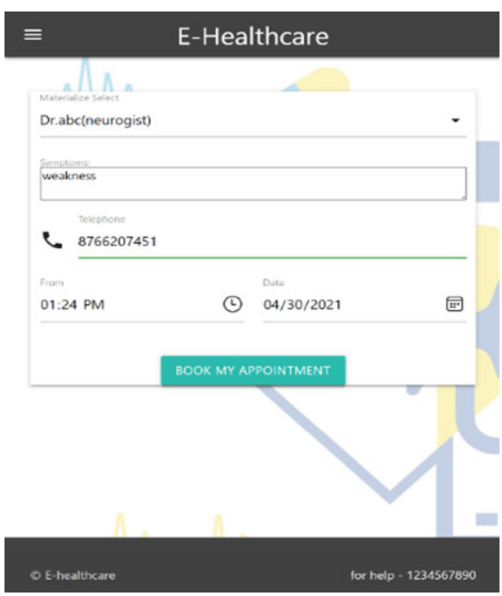
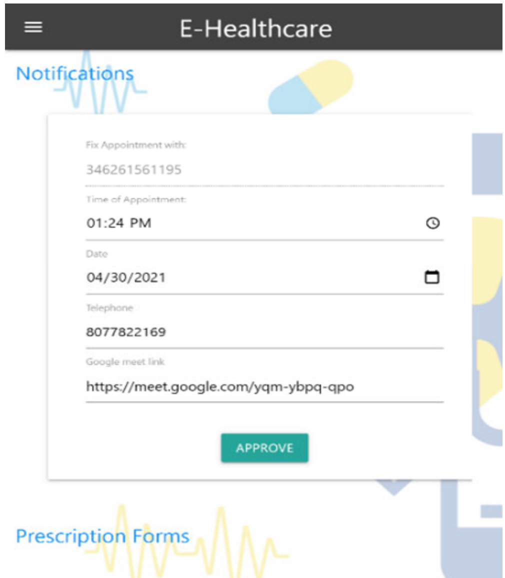


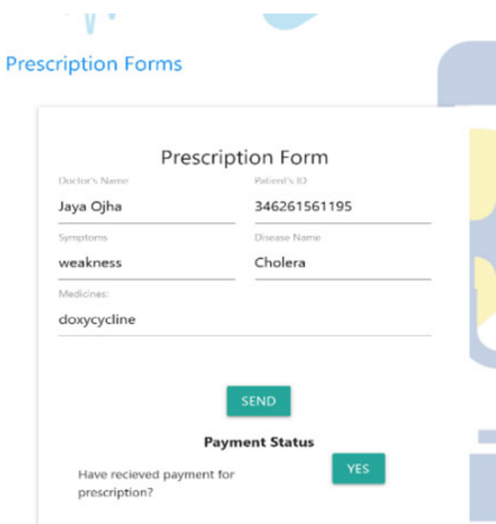
Figure 12 Appointment approval (see online version for colours)



3.4 Doctor approving appointment request

Figure 12 represents the appointment requests received by medicos from several patients with the time and date mentioned by patients. In contrast, medicos can approve or transmute the timing of the appointment according to their schedule and provide links for video checkups or contact details for an audio check-up as per their accommodation.

Figure 13 Prescription form (see online version for colours)



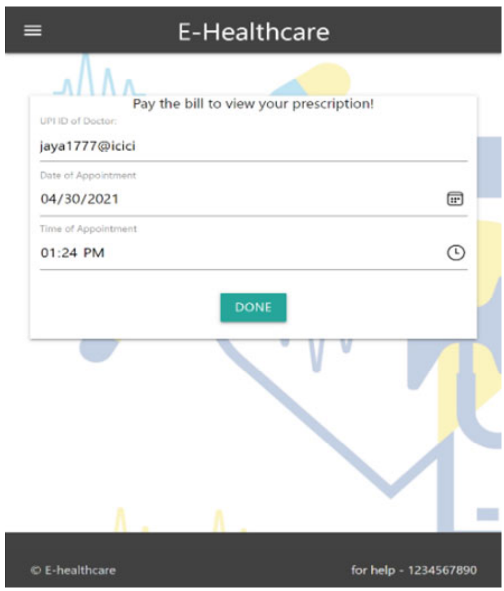
3.5 Doctor prescribing medicine after the appointment

After the video/audio check-up, a patient-medico needs to send the prescription form. The patient is notified by mail and on the dashboard regarding the equipollent shown in Figure 13.

3.6 Payment portal

In this portal, the patient needs to make the payments to the medico for a check-up depicted in Figure 14. Only after a medico update the payment status to affirmative will the patient be able to view his/her prescription.

Figure 14 Payment portal (see online version for colours)



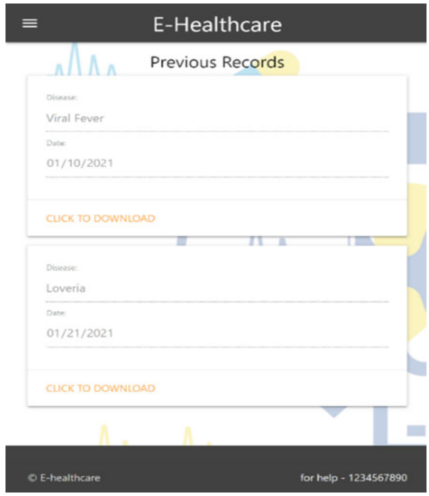
3.7 Viewing previous health records

Figure 15 represents a patient's anterior e-health records, including prescribed medicines from the medico with their respective dates and diseases.

3.8 Disease prediction

This portal is for patients whenever they encounter any symptoms and want to check whether they are suffering from any disease, and if yes, then which. But this is only for the patient's contentment, so if it shows any central quandary, then the patient can consult with medico.

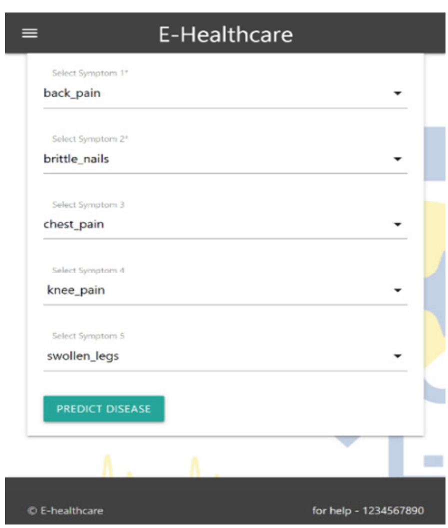
Figure 15 Previous records (see online version for colours)



3.8.1 Symptoms selected

This portal is for patients whenever they encounter any symptoms and want to check whether they are suffering from any disease, and if yes, then which. But this is only for the patient's gratification, so if it shows any central quandary, then the patient can consult with medico. Figure 16 represents the form that takes symptoms as input from the patient to soothe their disease. Here the patient has culled five symptoms: neck pain, back pain, knee pain, cramps, and painful ambulating.

Figure 16 Select symptoms (see online version for colours)



The patient needs to cull at least two symptoms, but for better results, they can provide up to five symptoms.

Figure 17 Predicted disease (see online version for colours)

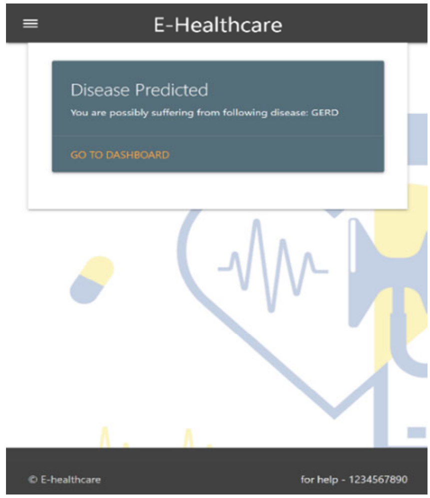
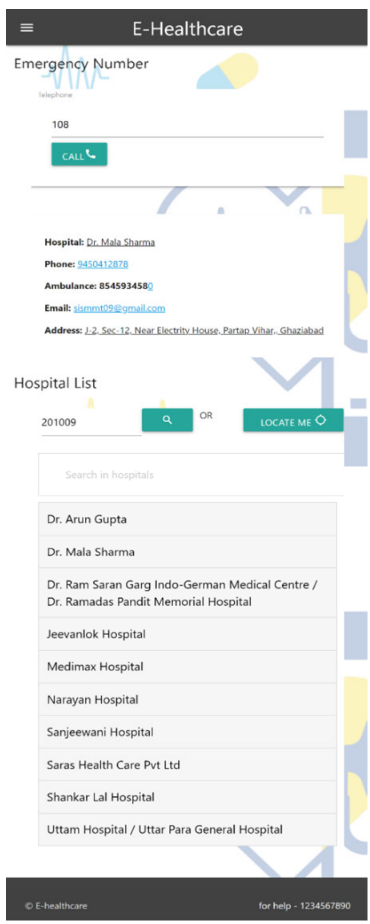


Figure 18 Emergency portal (see online version for colours)



3.8.2 Disease predicted

Figure 17 shows whether the symptoms lead to any disease and if yes, then which condition? After analysis of all the five symptoms provided by the patient, the patient gets the result of osteoarthritis.

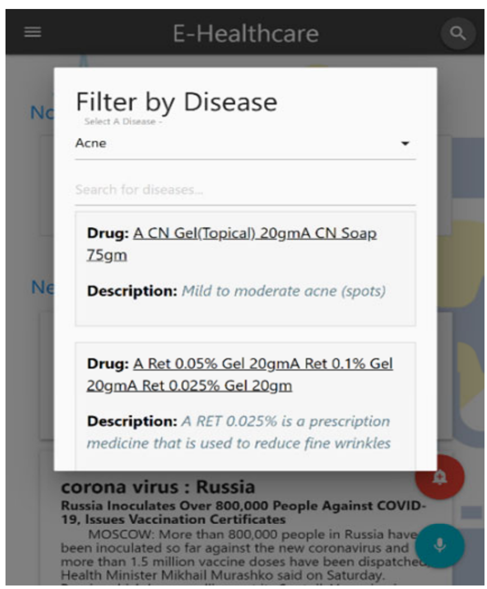
3.9 Emergency portal

It exhibits details of all the proximately located hospitals such as ambulance, phone number, email, and address by providing postal code or sanctioning location access as shown in Figure 18. These facilities are available to all patients, whether they are registered to this portal.

3.10 Suggesting medicines for disease

It exhibits all the possible treatments for a disease for the patient's erudition and contentment mentioned in Figure 19. For mundane quandaries like acne, gelid, headache, etc., most patients do not prefer to go to the medico, so they can get all the possible medications.

Figure 19 Suggesting medicines (see online version for colours)



4 Future scope

This system is not inhibited to medicos and patients. Still, it can involve a pharmacy module allowing patients location-predicated access to pharmacies and diagnostic labs near them. This way, they can avail all the facilities integrated within the same system. (Das and Alam, 2014; Ilorah et al., 2017; Cheng and Von Behren, 2013; Ahlers and Boll, 2007; Ortega et al., 2009) Concurrently, pharmacists can earn more mazuma as people often buy their medicines from online pharmacy management systems like pharmacies, needs, etc. Nowadays people do not ken where they are got medicines. They pay more mazuma. With the availability of another proposed system, they can get those medicines nearby medical shops only without paying the substantial amplitude of distribution charges. This system is a subsidiary for many people and comprises a website for end-utiliser. We developed a mobile application for end-utiliser, which would be facilely accessible and convenient for all the users, as the website may not be accessible everywhere. With a mobile application, the patient can avail the facility of emergency accommodations without the cyber world withal (i.e., offline mode) (Das and Alam, 2014; Ilorah et al., 2017; Iqbal et al., 2018).

5 Comparison with the existing system

5.1 Mobility

Less mobility as patients can get exhortation from medicos without dislocating from their house. The present system abstracts the importance of the subsisting system in which patients had to go to the doctor's clinic or hospital for fine-tuning appointments for checkups (Guo et al., 2016; Malik et al., 2017; Wu and Lu, 2017; Hu et al., 2012; Yin et al., 2019).

5.2 Medical records management

Facile maintenance of medical reports as e-records can be accessed at any time from anywhere. In subsisting system, patients have to carry precedent health records each time for checkups (Ceyhan et al., 2017; Jayaratne et al., 2019).

5.3 Time

Less time-consuming, and patients do not require to wait for their chance for a check-up. They can just attend the meeting with a medico at the time mentioned by the medico. It is more time-consuming as patients must wait in the queue and the medico provides no concrete time (Malik et al., 2017; Yin et al., 2019).

5.4 Burden on doctor

The abbreviated burden on medicos as they already have all the information afore approving appointments and can manage the appointments according to their time and cases. Medicos do not have any prior cognisance of appointments to be made the next day, so either they had to elongate their working hours or, in case of some meaningful work, they have to leave the patient (Saleem et al., 2015; Malik et al., 2017; Wu and Lu, 2017; Hu et al., 2012; Yin et al., 2019).

6 Conclusions

E-healthcare system is becoming more popular in internet-connected society due to modern digitisation. For this, we propose an integrated e-healthcare system that will able to provide miraculous facilities to patients Some of the primary facilities are virtual interaction of patient and doctor recommendation of medicine, giving prescription paper and reporting time to the doctor. Based on the patient's current health status, health history, and current symptoms, the present engine can look for personage with similar parameters in the database and suggest the drugs that are most successful for similar patients. With the proposed system's help, the doctor can make a better-informed decision through a virtual meeting. Apart from this present system provide a location-based emergency medical assistance system. Again, there is a database as a central server that comprises detailed information about healthcare centres. Users of the application can get information about the nearest healthcare centres from the user's current position in case of an emergency Hence these extra features make our system superior to another recently existing scheme.

References

Ahlers, D. and Boll, S. (2007) 'Location-based web search', *The Geospatial Web*, pp.55–66, Springer, London.

Beri, R. (2019) 'Machine learning for healthcare and pharma', *Journal of the Gujarat Research Society*, Vol. 21, No. 6, pp.500–504.

Ceyhan, M., Orhan, Z. and Domnori, E. (2017) 'E-medical test recommendation system based on the analysis of patients symptoms and anamneses', *CMBEBIH 2017*, Springer, Singapore, pp.654–659.

Chen, M. et al. (2010) 'A 2G-RFID-based e-healthcare system', *IEEE Wireless Communications*, Vol. 17, No. 1, pp.37–43.

Cheng, T.K. and Von Behren, J.R. (2013) *Location-based Searching using a Search Area that Corresponds to a Geographical Location of a Computing Device*, US Patent No. 8,386,514. 26 February.

Cola, C. and Valean, H. (2015) 'E-health appointment solution, a web-based approach', *2015 E-Health and Bioengineering Conference (EHB)*, IEEE.

Dalal, K.R. (2020) 'Analysing the implementation of machine learning in healthcare', *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)*, IEEE.

Das, R.C. and Alam, T. (2014) 'Location-based emergency medical assistance system using OpenStreetMap', *2014 International Conference on Informatics, Electronics & Vision (ICIEV)*, IEEE.

Dortmund, K. and Friedman, R.J. (1989) *Computer System and Method for Suggesting Treatments for Physical Trauma*, US Patent No. 4,839,822, 13 June.

Goyal, V.A. et al. (2020) 'Medicine recommendation system', *Medicine*, Vol. 7, No. 3, p.1658.

Guo, C. et al. (2016) 'Fine-grained database field search using attribute-based encryption for e-healthcare clouds', *Journal of Medical Systems*, Vol. 40, No. 11, p.235.

Hu, X. et al. (2012) 'The prepared patient: information seeking of online support group members before their medical appointments', *Journal of Health Communication*, Vol. 17, No. 8, pp.960–978.

Hussein, A.S. et al. (2012) 'Efficient chronic disease diagnosis prediction and recommendation system', *2012 IEEE-EMBS Conference on Biomedical Engineering and Sciences*, IEEE.

Ilorah, A.I. et al. (2017) 'Issues and challenges of implementing mobile e-healthcare systems in South Africa', *African Journal of Biomedical Research*, Vol. 20, No. 3, pp.249–255.

Iqbal, Z. et al. (2018) 'A comparative study of machine learning techniques used in non-clinical systems for continuous healthcare of independent living', *2018 IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE)*, IEEE.

Jayaratne, M. et al. (2019) 'A data integration platform for patient-centred e-healthcare and clinical decision support', *Future Generation Computer Systems*, Vol. 92, No. C, pp.996–1008.

Liu, W. and Park, E.K. (2012) 'E-healthcare security solution framework', *2012 21st International Conference on Computer Communications and Networks (ICCCN)*, IEEE.

Malik, S. et al. (2017) *Mr. Doc: A Doctor Appointment Application System*, arXiv preprint arXiv: 1701.08786.

Mukherjee, A. and McGinnis, J. (2007) 'E-healthcare: an analysis of key themes in research', *International Journal of Pharmaceutical and Healthcare Marketing*, Vol. 1, No. 4, pp.349–363.

Ortega, R.E., Frederick, R. and Dorfman, B. (2009) *System and Method for Providing Search Results Based on Location*, US Patent No. 7,565,157, 21 July.

Sahi, M.A. et al. (2017) 'Privacy preservation in e-healthcare environments: state of the art and future directions', *IEEE Access*, Vol. 6, pp.464–478.

Saleem, K. et al. (2015) 'Human-oriented design of secure machine-to-machine communication system for e-healthcare society', *Computers in Human Behavior*, Vol. 51, pp.977–985.

Wu, H. and Lu, N. (2017) 'Online written consultation, telephone consultation and offline appointment: an examination of the channel effect in online health communities', *International Journal of Medical Informatics*, Vol. 107, pp.107–119.

Yin, L. et al. (2019) 'Security-aware department matching and doctor searching for online appointment registration system', *IEEE Access*, Vol. 7, pp.41296–41308.