

GLOBAL JOURNAL OF ENGINEERING SCIENCE AND RESEARCHES

HOSPITAL BOT

A. Greeshma^{*1}, A. Priyanka², A.M. L Vasavi³, Mr. Rajasekar Sastry⁴, Dr. B V Ramana Murthy⁵ & Mr. C Kishor Kumar Reddy⁶

^{*1,2,3,4,5&6}Stanley College of Engineering and Technology for Women, Hyderabad

ABSTRACT

Life is becoming so busy in day to day life to get medical appointments in the hospitals. In this competitive world medical services with quality is more critical in the health industry. The major source of the patient dissatisfaction is waiting long time, thus to improve the medical service quality and to reduce waiting time is the most important. The bot which is well designed schedule of appointment can decrease the waiting time of the people and can increase the satisfaction of the people. This study is meant to create a framework for the individualized outpatient appointment schedule(OAS)in some of the hospitals. To make it, firstly model of the treatment duration of a patient is confirmed by using artificial neural network. Secondly, Simulation method is developed for simulation model which imitates the operations done by the mentioned hospitals. The main moto of this is to provide ease and comfort while taking appointment from doctors and it also resolves the problems that the patients has to face while taking an appointment by going to the required place.

Keywords: Individualized outpatient, Appointment, Hospital, Health, Simulation.

I. INTRODUCTION

A machine is introduced to replace humans in the works they perform. They work in the place of humans which reduces the extra burden in their day to day life. Later researcher didn't restrict the applications of machines and they introduced chatbots which will chat with humans and give the information what they want. They answer to all the requested demands by human. It has an alert expression which is maintained constantly for appointing or booking any hospital requirements. The chatbot in today's generation are quite unique having inbuilt functionality and predefined functions to perform day to day functionality of streamline and any other medical practices. Mainly used for scheduling and appointment.

Think of medical chatbots as virtual assistants for your clinic, hospital or healthcare centre. They are inexpensive. Easy to add to your website, and highly efficient at what they do. To make the picture a little clearer, lets run through some of the benefits you get when you create a chatbot to make appointments at your place. When patients are strapped for time, the last thing they want is to visit the doctor's clinic in person to make an appointment. Even calling to the clinic required and waiting in the line for hours also makes the people irritate and make their time waste. In any case, neither of these methods matches the simplicity and convenience of typing a quick request into a chat window and getting a confirmed appointment. The chatbots that are used by the majority of people now a days for assistance is Apple Siri, Google Allo, Microsoft cortana etc. These assistants answer to all the questions like a question and answer session. They are used for asking queries, and getting the information about the things or places that they want. There are number of approaches between human and computer interaction. One of them is Natural Language Processing. Chatbots need to be trained regularly so that they are shaped to tackle the daily queries from and to the users. During medical chatbot development, the doctors and healthcare can incorporate basic FACs in the chat script, so that visitors and patients can get the information that they wanted without any problem and without any calls and any delay.

About the project

In this project, we created a chatbot which gives the information about the hospital as well as books the appointment for the one who are in need. The information given in this project is about the hospital timings, infrastructure, location, branches of some of the few top hospitals and the doctors who are available in the specified hospitals.

II. LITERATURE SURVEY

The assistant, a human is inefficient and takes a long time to process a single request such as booking tickets ([source](#)), ordering food ([source](#)), getting beauty-services ([source](#)) etc. This reduces the throughput and business performance drastically. Obviously there is an increase in the demand of chat automation because it removes the human factor and it can give a 24 hour service which will have a multiplicative effect on the revenue generation.

There are so many chatbots that were developed in order to help the people. We can see them in messengers and some chat channels. The need for chatbot is increasing as the queries from people also increasing day by day. The chatbot accepts natural language input from users, navigates through the Information Repository and responds with information in natural language.

If we look at some papers we can see there are so many chatbot developments, but we rarely see chatbot that are developed for medical use. So our project aims to develop a chatbot that will be useful for medical purpose.



III. PROPOSED CHATBOT

DESIGN

Existing System

Currently, one can make a chatbot of various types -

- One that follows a basic set of menu-driven tasks, but with user input in the form of text.
- One that attempts to have a natural conversation with the user.
- One that performs tasks based on user input and returns output of query.

The first type does not require any form of NLP as the input can be processed as in a regular C program. There are existing services that allows a way to implement this type of bots in a convenient user interface. The others require some form of NLP, which can be done in-house or outsourced from an existing service.

One of a prominent option for the latter is DialogFlow (previously known as api.ai), now owned by Google. The SDKs contain voice recognition, natural language understanding, and text-to-speech. It offers a web interface to build and test conversation scenarios. Voice and conversational interfaces created with Dialogflow works with a wide range of devices including phones, wearables, cars, speakers and other smart devices. It supports 14+ languages including Brazilian Portuguese, Chinese, English, Dutch, French, German, Italian, Japanese, Korean, Portuguese, Russian, Spanish and Ukrainian. Dialogflow also includes an analytics tool that can measure the engagement or session metrics like usage patterns, latency issues, etc.

In this Code Pattern, we will use the Watson Assistant Slots feature to build a friendly chatbot.

The things we use to create the conversations are

- Intents
- Extents
- Dialogs

Intents are purposes or goals expressed in a customer's input, such as answering a question or processing a bill payment. By recognizing the intent expressed in a customer's input, the Watson Assistant service can choose the correct dialog flow for responding to it.

Plan the intents for your application.

Consider what your customers might want to do, and what you want your application to be able to handle on their behalf.

Teach Watson about your intents.

Once you decide which business requests you want your application to handle for your customers, you must teach Watson about them.

Creating intents

Use the Watson Assistant tool to create intents.

- In the Watson Assistant tool, open your dialog skill. The skill opens to the **Intents** page.
- Select **Create new**.
- In the **Intent name** field, type a name for the intent.
- The intent name can contain letters (in Unicode), numbers, underscores, hyphens, and periods.
- The name cannot consist of or any other string of only periods.
- Intent names cannot contain spaces and must not exceed 128 characters.
- Select **Create intent** to save your intent name. Next, in the **Add user examples** field, type the text of a user example for the intent. An example can be any string up to 1024 characters in length. Click **Add example** to save the example.
- Repeat the same process to add more examples. You can tab between examples. Provide at least 5 examples for each intent. The more examples you provide, the more accurate your application can be.
- To get help with user example creation, see Get intent user example recommendation.
- When you have finished adding examples, click to finish creating the intent.
- The system begins to train itself on the intent and user examples you added.

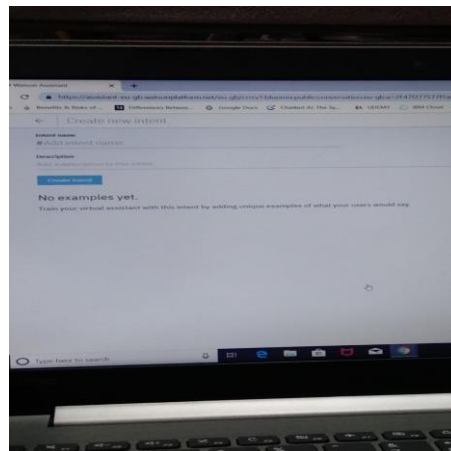


Figure2: Creating Intents

Creating entities

Entities represent information in the user input that is relevant to the user's purpose. Recognizing entities in the user's input helps you to craft more useful, targeted responses.

Synonym entity: You define a category of terms as an entity (color), and then one or more values in that category (blue). For each value you specify a bunch of synonyms (aqua, navy). You can also pick synonyms to add from recommendations made to you by the service. At run time, the service recognizes terms in the user input that exactly match the values or synonyms that you defined for the entity as mentions of that entity.

Pattern entity: You define a category of terms as an entity (contact info), and then one or more values in that category (email). For each value, you specify a regular expression that defines the textual pattern of mentions of that value type. For an email entity value, you might want to specify a regular expression that defines a text@text.com pattern.

At run time, the service looks for patterns matching your regular expression in the user input, and identifies any matches as mentions of that entity.



Figure 3: Creating entity

Dialog

The dialog uses the intents that are identified in the user's input, plus context from the application, to interact with the user and ultimately provide a useful response. The dialog matches intents (what users say) to responses (what the bot says back).

Dialog nodes

Each dialog node contains, at a minimum, a condition and a response.

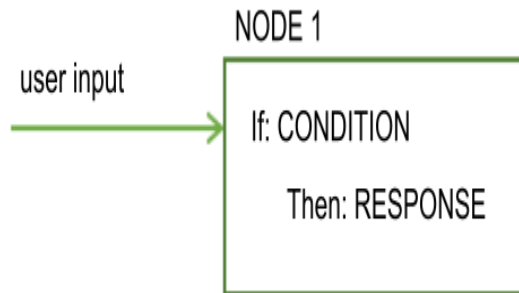


Figure 4: Using conditions in dialogs

- Condition: Specifies the information that must be present in the user input for this node in the dialog to be triggered. The information is typically a specific intent. It might also be an entity type, an entity value, or a context variable value. See Conditions for more information.

Response: The utterance that the service uses to respond to the user. The response can also be configured to show an image or a list of options, or to trigger programmatic actions. See Responses for more information.

You can think of the node as having an if/then construction: if this condition is true, then return this response.

Dialog flow

The dialog that you create is processed by the service from the first node in the tree to the last.

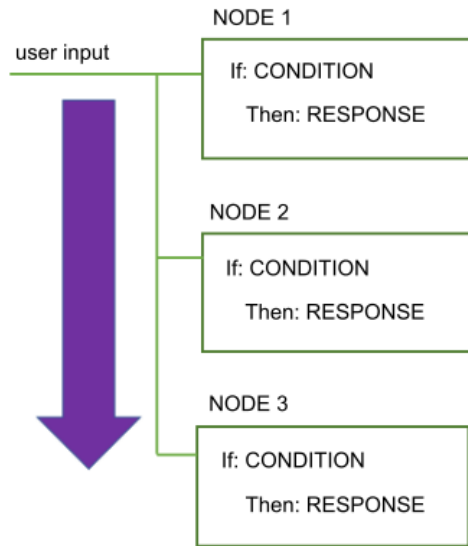
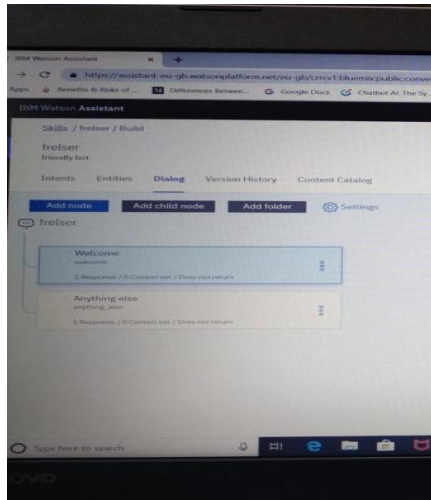


Figure 5: Responses to conditions

As it travels down the tree, if the service finds a condition that is met, it triggers that node. It then moves along the triggered node to check the user input against any child node conditions. As it checks the child nodes it moves again from the first child node to the last.



IV. ARCHITECTURE OF NODE-RED

Once the node red is created, the following can be done.

1. Add an Inject node

The Inject node allows you to inject messages into a flow, either by clicking the button on the node, or setting a time interval between injects. Drag one onto the workspace from the palette. Open the sidebar and select the Info tab. select the newly added Inject node to see information about its properties and a description of what it does.

2. Add a Debug node

The Debug node causes any message to be displayed in the Debug sidebar. By default, it just displays the payload of the message, but it is possible to display the entire message object.

3. Wire the two together

Connect the Inject and Debug nodes together by dragging between the output port of one to the input port of the other.

4. Deploy

At this point, the nodes only exist in the editor and must be deployed to the server.

Click the Deploy button. With the Debug sidebar tab selected, click the Inject button. You should see numbers appear in the sidebar. By default, the Inject node uses the number of milliseconds since January 1st, 1970 as its payload. Let's do something more useful with that.

5. Add a Function node

The Function node allows you to pass each message through a JavaScript function. Wire the Function node in between the Inject and Debug nodes. You may need to delete the existing wire.

The connections are completed by connecting the flows to each other.

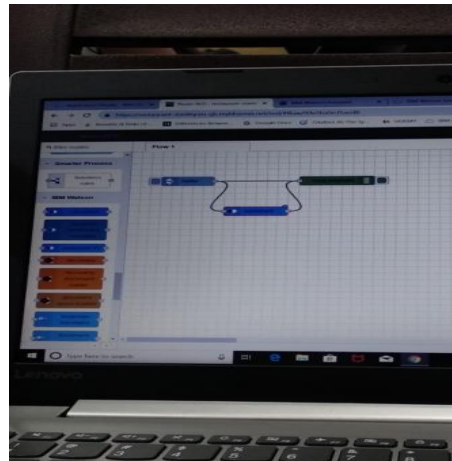
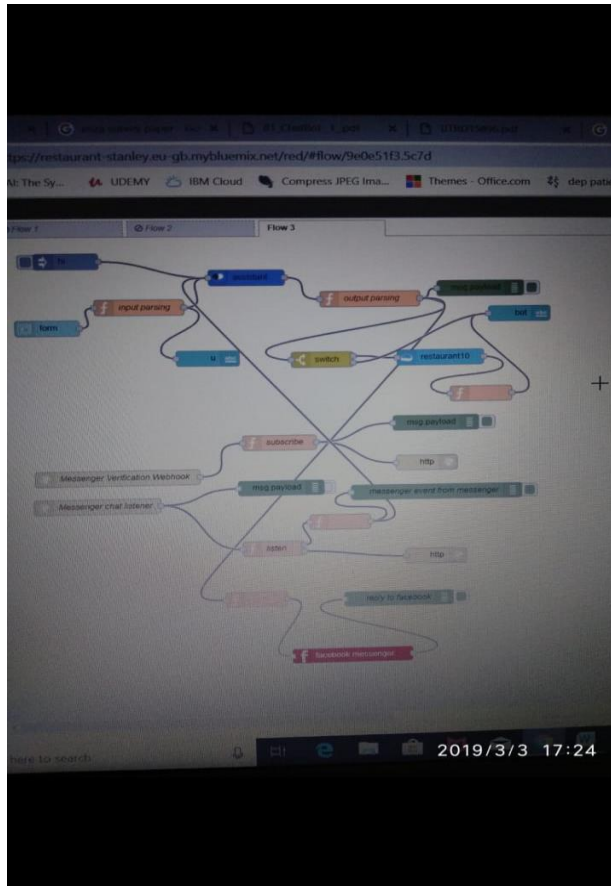
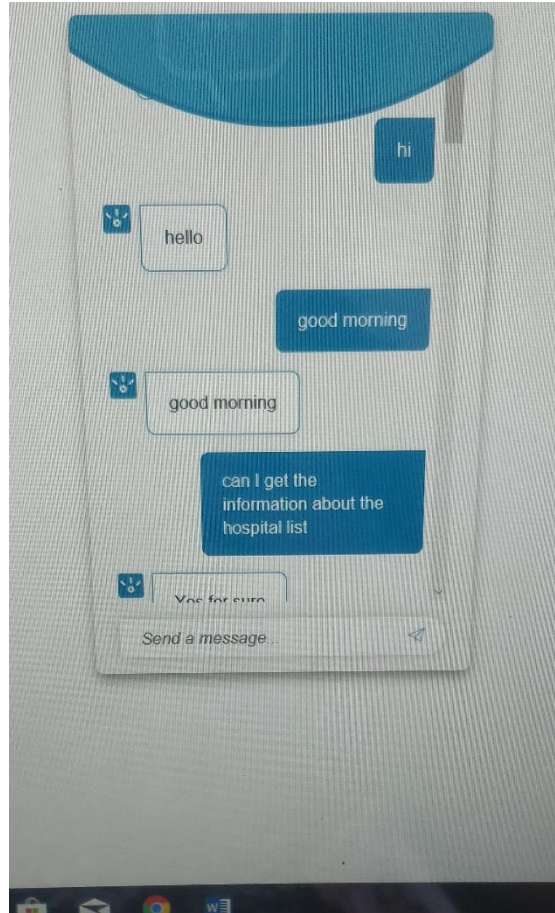


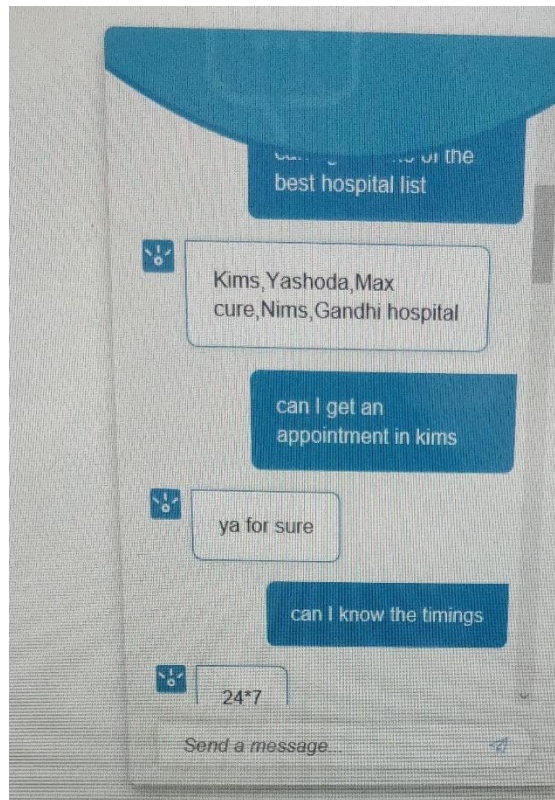
Figure 7: Creating basic flow



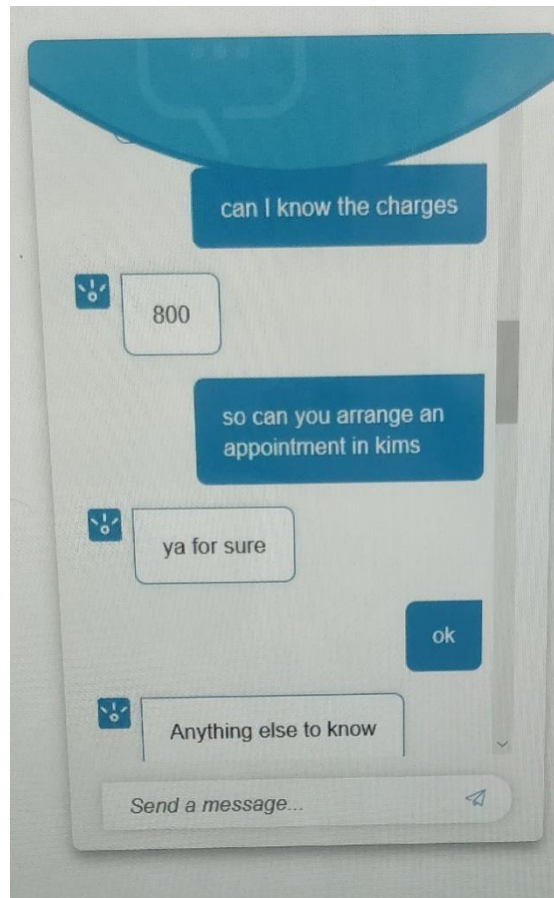
V. RESULTS AND DISCUSSION



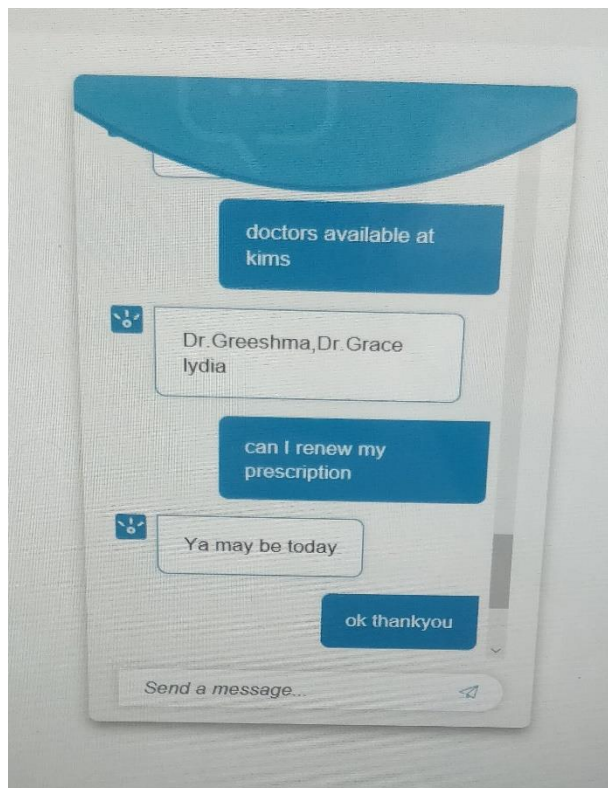
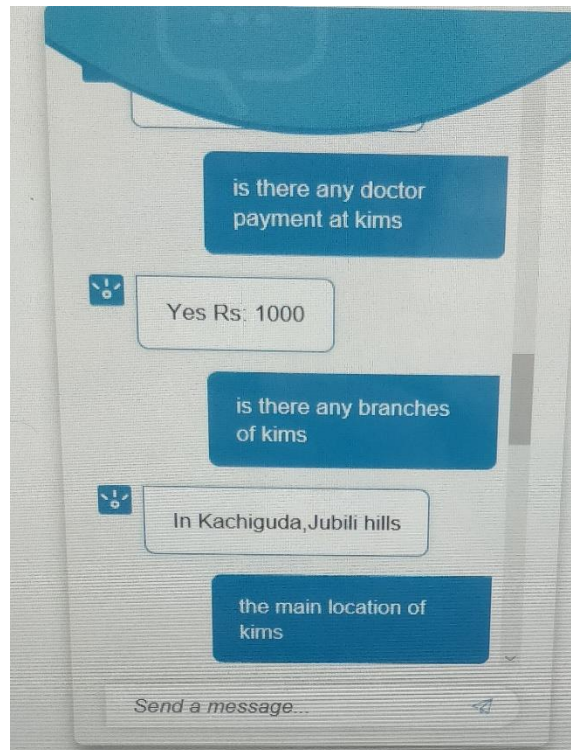
Using IBM Watson assistant, we have created dialogs which responds to the user the very next moment the user opens the preview link. The bot gives the warm welcome to the user when they greet it. Here the chatbot starts working. And then the bot puts the further questions and depending on the user's response the bot suggests the user things like which hospital they want to select and it gives all the information they want for booking an appointment.



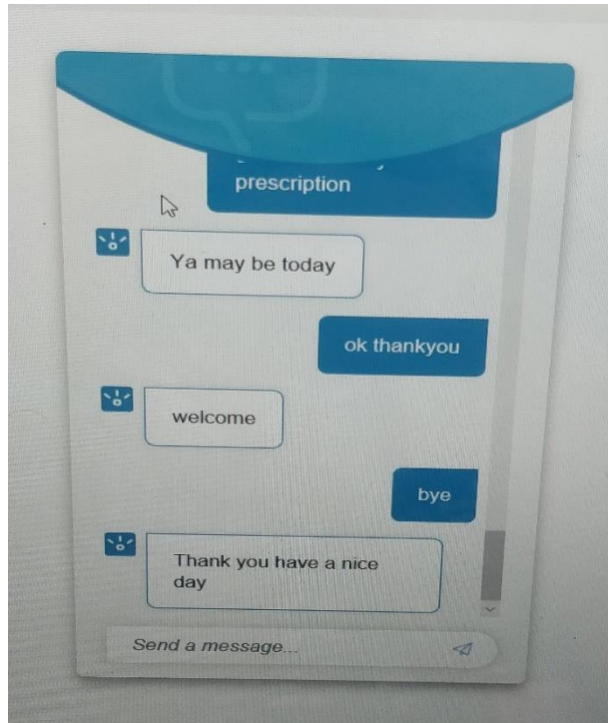
Here it gives the information about some of the top hospitals and it helps to fix the appointment in the required hospital and it also gives the timing about the hospitals.



Here it gives the information about the charges that the hospital takes for booking an appointment. And it also asks the user that anything else that the user want to know.



Here it gives the information when the user ask about the extra consultant fees in the required hospitals. And it also gives the hospital branches if required and it also gives the main location of the hospital and it also gives the information about the doctors available at the hospitals and even we can ask about our renew of the prescription.



Here at last it ends with giving a warm wish by thanking the users and by fixing their appointments.

VI. CONCLUSION

In this booking an appointment system the user can see the consultation fee and if any extra fees to the doctor. And the user can find out which hospital is best for the treatment. The user can book the appointment according to his free time. Now the world is finding an easy way for booking an appointment. But finding the best hospital is quite difficult. All of us know that they charge high cost and does not give a better treatment. So, user can choose a better hospital by this chatbot as it gives the information about some of the best hospitals.

REFERENCES

1. Arthur Hylton III and Suresh Sankaran arayanan "Application of Intelligent Agents in Hospital Appointment Scheduling System", *International Journal of Computer Theory and Engineering*, Vol. 4, August 2012, pp. 625-630.
2. Deepti Ameta, Kalpana Mudaliar and Palak Patel "Medication Reminder And Healthcare – An Android Application", *International Journal of Managing Public Sector Information and Communication Technologies (IJMPICIT)* Vol. 6, June 2015, pp. 3948.
3. Yeo Symey, Suresh Sankaran arayanan, Siti Nurafifah binti Sait "Application of Smart Technologies for Mobile Patient Appointment System", *International Journal of Advanced Trends in Computer Science and Engineering*, august 2013.
4. Jagannath Aghav, Smita Sonawane, and Himanshu Bhambhani "Health Track: Health Monitoring and Prognosis System using Wearable Sensors", *IEEE International Conference on Advances in Engineering & Technology Research 2014*, pp. 1-5.

5. YoeSyMey and Suresh Sankaranarayanan "Near Field Communication based Patient Appointment", *International Conference on Cloud and Ubiquitous Computing and Emerging Technologies*, 2013, pp.98-103.
6. RashmiA.Nimbalkar and R.A. Fadnavis "Domain Specific Search of Nearest Hospital and Healthcare Management System", *Recent Advances in Engineering and Computational Sciences (RAECS)*, 2014, pp.1-5.
7. A. Luschi, A. Belardinelli, L. Marzi, F. Frosini, R. Miniati and 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, pp.125-128.
8. Choi, J. ; Biomed lab Co., Seoul, South Korea ; Kang, W.Y. ; Chung, J. ; Park, J.W. "Development Of An Online Database System For Remote Monitoring Of Artificial Heart Patient", *Information Technology Applications in Biomedicine*, 2003. 4th International IEEE EMBS Special Topic Conference, 2426 April 2003
9. [9] Prof. S. B. Choudhari, ChaitanyaKusurkar, RuchaSonje, ParagMahajan, Joanna Vaz "Android Application for Doctor's Appointment", *International Journal of Innovative Research in Computer and Communication Engineering*, January 2014
10. S.Gavaskar, A. Sumithra, A.Saranya "Health Portal-An Android Smarter Healthcare Application", *International Journal of Research in Engineering and Technology*, Sep-2013.
11. Frank Sposaro and Gary Tyson, "iFall: An android application for fall monitoring and response", *31st Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 1:6119–22, 2009.
12. Pei-Fang Tsai, I-sheng Chen, and Keven Pothoven "Development of Handheld Healthcare Information System in an Outpatient Physical Therapy Clinic", *proceedings of the 2014 IEEE 18th International Conference on Computer Supported Cooperative Work in Design*, pp. 559-602.
13. Jin Wang, Richard Y.K. Fung "adaptive dynamic programming algorithms for sequential appointment scheduling with patient preferences", *Science Direct, Artificial Intelligence in Medicine* January 2015, Pages 33–40
14. Bin Mu, Feng Xiao, Shijin Yuan "A Rule-based Disease Self-inspection and Hospital Registration Recommendation System", *Software Engineering and Service Science (ICSESS)*, 2012 IEEE 3rd International Conference, 22-24 June 2012.
15. Agnese Augello, Giovanni Pilato, Alberto Machi' ICAR Istituto di Calcolo e Reti ad Alte Prestazioni CNR - Consiglio Nazionale delle Ricerche Viale delle Scienze , 978-0-76954859-3/12 \$26.00 © 2012 IEEE . "An Approach to Enhance Chatbot Semantic Power and Maintainability: Experiences within the FRASI Project".
16. Emanuela Haller, Traian Rebedea Faculty of Automatic Control and Computers university Politehnica of Bucharest, 978-0-7695-4980-4/13 \$26.00 © 2013 IEEE. "Designing a Chat-bot that Simulates a Historical Figure".
17. "Real World Smart Chatbot for Customer Care using a Software as a Service (SaaS) Architecture" Godson Michael D'silva1, *, Sanket Thakare2, Sharddha More1, and Jeril Kuriakose1, *International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)*
18. Bayu Setiaji, Ferry Wahyu Wibowo , Department of Informatics Engineering STMIK AMIKOM Yogyakarta, Yogyakarta, Indonesia, 2166-0670/16 \$31.00 © 2016 IEEE "Chatbot Using A Knowledge in Database-Human-to-Machine Conversation Modeling".
19. Chatbot Using A Knowledge in Database,"Bayu Setiaji,Ferry Wahyu Wibowo",2016 7th International Conference on Intelligent Systems, Modelling and Simulation.2016 IEEE.
20. "Novel Approach for Medical Assistance Using Trained Chatbot",Divya Madhu, Neeraj Jain C,*International Conference on Inventive Communication and Computational Technologies*.
21. V.Manoj Kumar"Sanative Chatbot For Health Seekers",*JECS Volume 05 Issue 3 March 2016 Page No.16022-16025*.
22. Imran Ahmed and Shikha Singh"AIML Based Voice Enabled Artificial Intelligent Chatterbot",*International Journal of u-and e-Service, Science and Technology Vol.8, No.2 (2015)*.
23. Ercan Canhasi,"Measuring the sentence level similarity "35 *ISCIM 2013*, pp. 35-42.
24. Yuhua Li, Zuhair Bandar, David McLean and James O'Shea "A Method for Measuring Sentence Similarity and its Application to Conversational Agents "Intelligent Systems Group.