

Automated Medical Health Diagnosis System

Ms. Akanksha Yadav
M. Tech Student

*Department of Computer Science and Engineering
Goel Institute of Technology & Management, Lucknow, India*

Mr. Shivam Shukla
Assistant Professor

*Department of Computer Science and Engineering
Goel Institute of Technology & Management, Lucknow, India*

Abstract

Mostly in large cities such as Mumbai, Bangalore, Delhi, most patients prefer to go to large hospitals to visit the doctors. As a result there is a congestion in the large hospitals. Consequently most doctors usually only had roughly 5 minutes on average to make their diagnosis on the patients. Doctor may not think about all diseases at once by listening to patient's symptoms. This problem grew even worse for radiologists also. In the large hospitals there are usually more than thousand examinations including images being performed (e.g., CT, MR, DR/CR) per day. Radiologist has to read more than 50 studies daily. They have only around 10 to 15 minutes to read images and to write a report for each study (e.g., CT or MR examination). Basically there is no time to review the patient's history if such patients had multiple historical studies. So there is a big need to store all the valuable data and the symptoms on a web platform portal to make diagnosis efficient and accurate. An automated medical health diagnosis system is a web platform portal to facilitate doctors to enter patients' symptoms and get the list of probable diseases, which enables doctors to start diagnosis quickly. In this system security is maintained as only the registered doctors can enter or update the system accordingly. The system enables the registered doctors to record the new diseases and symptoms if they find any. The project reduces the diagnosis time as well. Data handling technique is dealing with large amount of data with a great accuracy and reliability. This system has a database containing data about all diseases and their corresponding symptoms. When the patient details are entered, the system indicates the possible diseases the patient may be suffering from. The performance is measured by taking input from the registered doctor and from a patient who enter his symptoms and then analyzed. An Automated Medical Health Diagnosis System compares all entries in the database and update it if needed and finally gives the result in the form of diagnosed disease. Moreover it act as disease and symptoms repository.

Keywords- Microsoft Visual Studio 2017 Community Edition, GUI – ASP.NET MVC 5.0, C#, WEB API (as service) for Business Layer, Entity Framework v6 for Database Layer, Hosting – IIS 7

I. INTRODUCTION

Due to the generation of a large volume of Medical diagnostic records in the healthcare world, Medical diagnostic records have become the main component of medical big data, which brings huge value to healthcare services, professionals, and administration. But this large volume of Medical diagnostic records presented in hospitals also brings new challenges to healthcare professionals and services in terms of storing and managing the data.

Automated Medical Health Diagnosis System is a database management system (DBMS), which is based on computer networks and using the advanced database technology to construct, maintain and manipulate various kinds of data in a database system (DBS). The DBMS can track and update all the information of recorded patients in the healthcare center during a particular time span. An (AMHDS) Automated Medical Health Diagnosis System is a web platform portal to facilitate doctors to enter patients' symptoms and get the list of probable diseases, which enables doctors to start diagnosis quickly. The objectives of the system are-

- To record diseases and symptoms
- To reduce diagnosis time
- To Increase accuracy and reliability
- To record patients' data and diagnosis
- To act as disease and symptoms repository

This system has a database containing data about all diseases and their corresponding symptoms. When the patient details are entered, the system indicates the possible diseases the patient may be suffering from. This list of diseases help to clear the doctor's minds what to prescribe the medicine to the patient. Actually first they analyze the patient. This project reduces the diagnosis time, even though it can be deployed for online treatment of the patient.

II. CONCEPT

Now- a- days a web-based application is not adequate to reach its customers. People are using smart gadgets like iPhone, mobile, tablets etc. in its daily life. These devices also contain a lot of apps for making the life easy. Literally now we are moving from the web towards apps world. Web API is the great framework for exposing the data and service to the different devices. Web API is

open source and is an ideal platform for building REST-ful services over the .NET Framework. It uses all the full features of HTTP (like URIs, request/response headers, caching, versioning, various content formats) and there is no need to define any extra configuration or settings for different devices unlike WCF Rest service. It supports convention-based CRUD Actions since it works with HTTP, verbs GET, POST, PUT and DELETE. Here Web API is hosted on Microsoft IIS 7 server. Microsoft IIS server is more simple and robust as it also supports the MVC features like routing, controllers, action results, filter, model binders, IOC container or dependency injection. The concept of the system can be understood by fig 1.

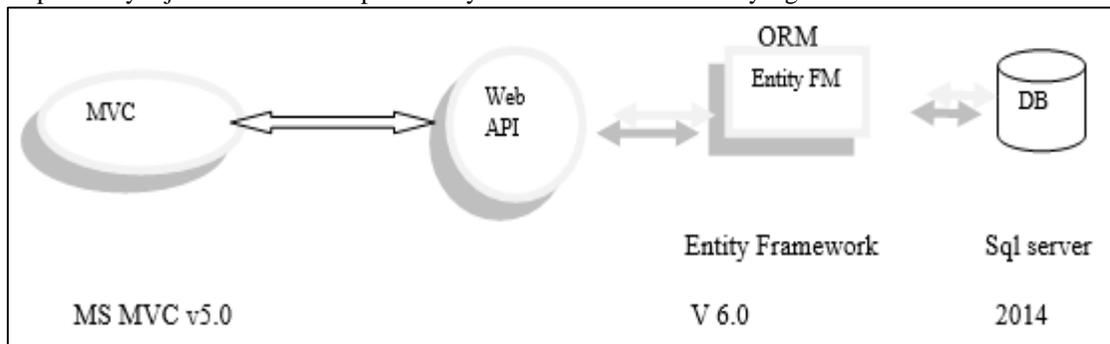


Fig. 1: conceptual diagram

ORM is an Object Relational Mapping technique, also known as O / R or Object-Relation mapping tool. It converts the data between incompatible type systems in object-oriented programming languages. This creates a "Virtual Object Database "(VOD) that can be used within the programming language.

In object-oriented programming data-management tasks operate on object-oriented (OO) object that are almost always non-scalar values.

The Entity Framework enables the developers to work with data in the form of domain-specific objects and properties such as customers and customer addresses. They don't have any concern with the underlying database tables and columns where this data is stored. Developers can work at a higher level of abstraction with the Entity Framework when they deal with data. They can create and maintain data-oriented applications with less code as compared in traditional applications

Model-View-Controller (MVC) is a software architectural pattern which separates an application into three main components:

- Model
- View
- Controller

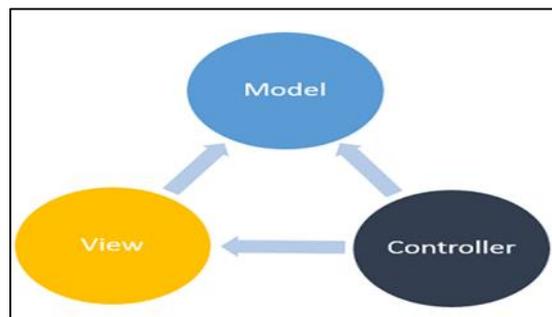


Fig. 2: MVC Structure

A. Components of MVC

The ASP.NET MVC framework is the most recent framework for creating web applications. It is developed by Microsoft.

B. Model

- MVC model is a C# class or VB.NET class.
- A model is accessible by both the components - controller and view.
- A model can be used to pass data from Controller to view component

C. View

- MVC view is the component that display the application's User Interface (UI)
- A view show model data in the page.

D. Controller

- Controller is basically a C# or VB.NET class. It inherits system.mvc.controller

- Controllers handle user interaction.
- Controller can access and use model class to pass data to view.

Internet Information Server-IIS7 is the most powerful web server. It is developed by Microsoft. IIS7 is used to host ASP.NET Web applications. IIS has its own ASP.NET Process Engine to handle the ASP.NET request. When a client send a request to server, IIS7 takes that request and process it and then send the response back to the clients.

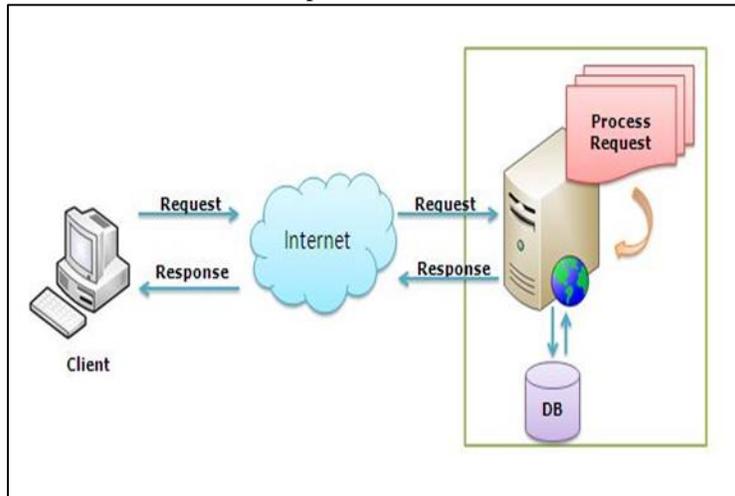


Fig. 3: Request handling by IIS

When any request comes from client to the server a lot of operation is performed before sending the response to the client. There is two main components in the web garden--1. Worker Process 2. Application Pool

- 1) Worker Process: Worker Process is responsible to manage all the requests and responses which are coming from client. All the ASP.Net functionality runs under the scope of it. When a client send the request to the server, worker process is responsible to generate the request and response. Actually worker process is the heart of ASP.NET Web Application which runs on IIS.
- 2) Application Pool: Application pool is the container of worker process. Application pools are used to contain the separate sets of IIS worker processes that share the same configuration. Application pools enable a better security, reliability, and availability for any web application. The worker process serves as the process margin that separates each application pool because of not affecting the other worker process when one worker process or application is having an issue or recycles. This assures that a particular web application doesn't not crash other web application as they are configured into different application pools.

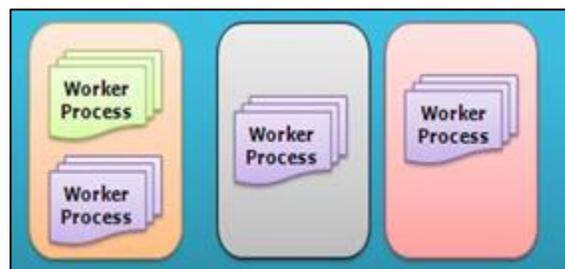


Fig. 4: Web Garden

Application Pool with multiple worker process is called "Web Garden". All the business models and business rules deal with Web API. ASP.NET Web API is a framework that makes it easy to build HTTP (Hyper Text Transfer Protocol) services that accomplish a broad range of clients including browsers and mobile devices. For building RESTful applications on the .NET Framework ASP.NET Web API is an ideal platform. It is a part of the core ASP.NET platform. It can be used with MVC and other types of Web applications like Asp.Net WebForms. It can also be used as a stand-alone Web service application.

ASP.Net Web API services are consumed through Rest Sharp.

Asp.Net Web API is a framework for building HTTP services. That HTTP services can be consumed by a broad range of clients including browsers, mobiles, iphone and tablets. It is not a part of the MVC Framework but it is a part of the core ASP.NET platform and can be used with MVC and other types of Web applications like Asp.Net Web Forms. It can also be used as a stand-alone Web services application.

III. PLAN, ANALYZE AND DESIGN

A. LLD (Low Level Diagram)

In the low level diagram (LLD) fig 5, the project concept with the help of flow chart is explained. First of all the system checks the doctor registration whether he is registered or not. If he is not registered, his registration will be done first. Further he enters the patient's records. He entered the patient's symptoms if it is not matched in our database, He enter the new symptoms and corresponding diseases into system database.

If entered symptoms are matched to the database, Automated Medical Health Diagnostic System gives the result in the form of probable diseases. At last the Automated Medical Health Diagnostic System enters the diagnosis data in the database as a record of the said patient.

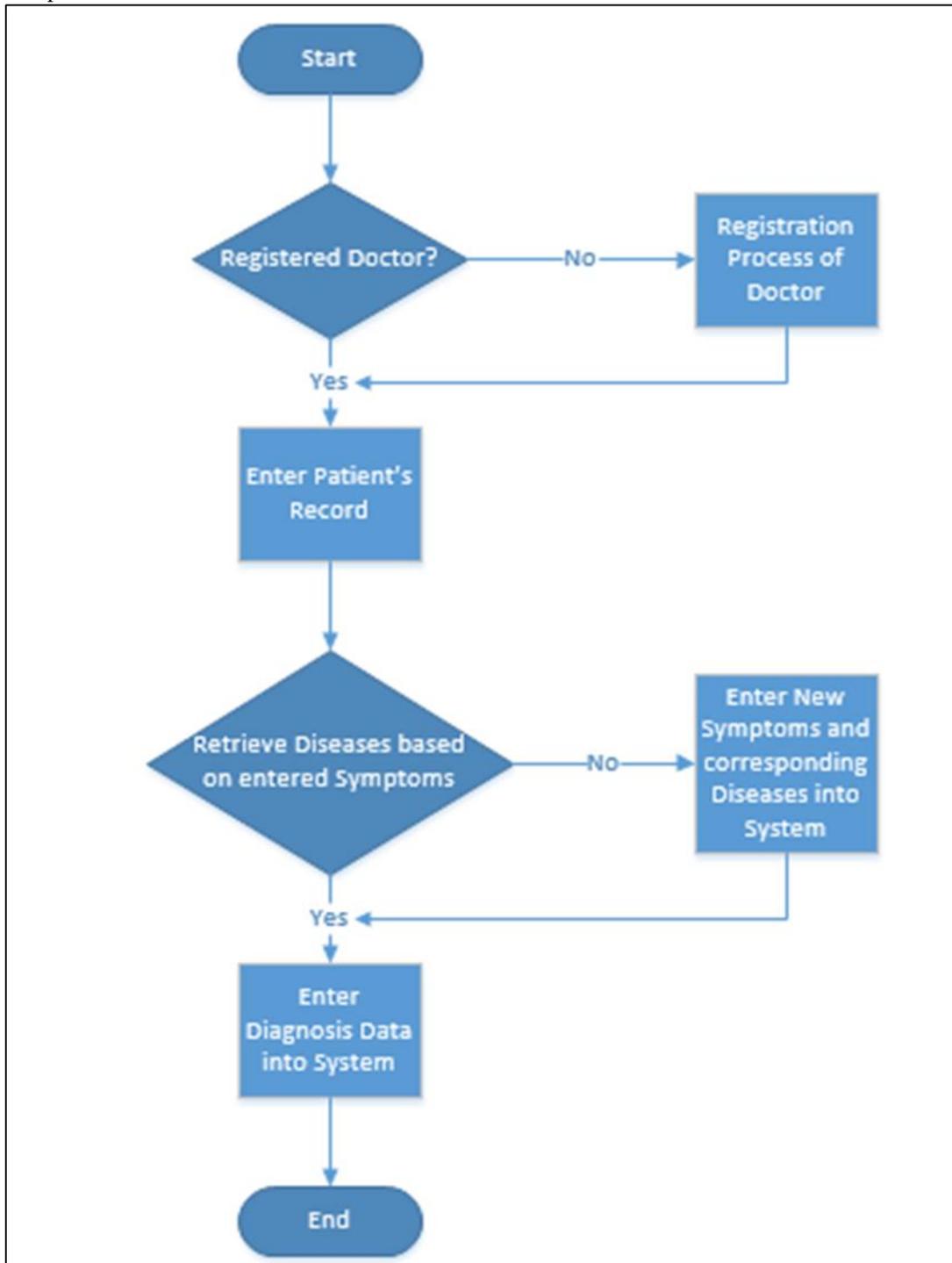


Fig. 5: Flow Chart of Automated Medical Health Diagnosis System

B. High Level Overview

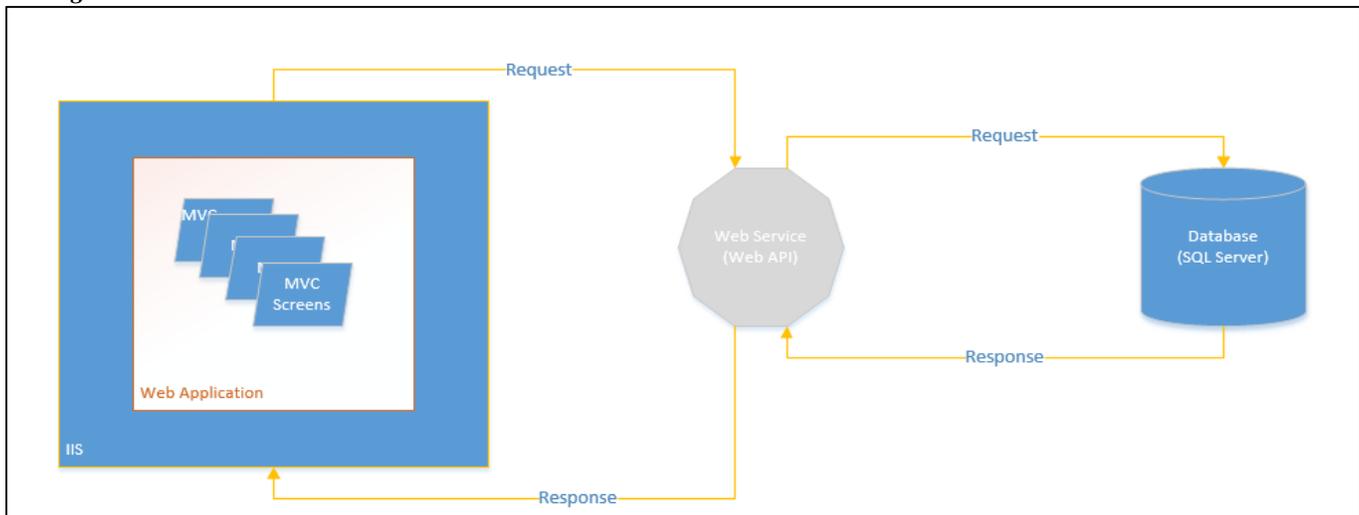


Fig. 6: General Overview of High Level Diagram

In order to design Automated Medical Health Research System, I used these approaches:

- 1) Registered doctors can enter patient's record like name, address, emergency contact and symptoms first create the database on SQL server of registered doctors, of patients' records and symptoms of diseases.
- 2) Based on said symptoms, registered doctor can retrieve the list of probable diseases. Registered doctors send the request to the server get the response according that through Web API. Here we can update the database for the new symptoms and diseases.
- 3) Doctor can save patient's diagnosis report date and time wise in database. For creating the database Microsoft SQL Server can be opted.

IV. RESULT

The result of Automated Medical Health Diagnosis System can be shown in these Use cases which are as follows:

A. Use Case 1: Add Disease Details

The input screen contains the following fields:

- Name: text value (e.g. "malaria")
- Severity: radio buttons for the severity (e.g. "high", "medium", "low")
- Cause: single-select list box containing the values "external factors" and "internal disorder"
- Description: multi-line text value
- Add Disease: button
- Cancel: button.

The user provides the required details in the form and clicks the "Add Disease" button. On successfully saving the details, the user is redirected to the home page, and the following message is displayed there: "The details of the new disease have been saved successfully".

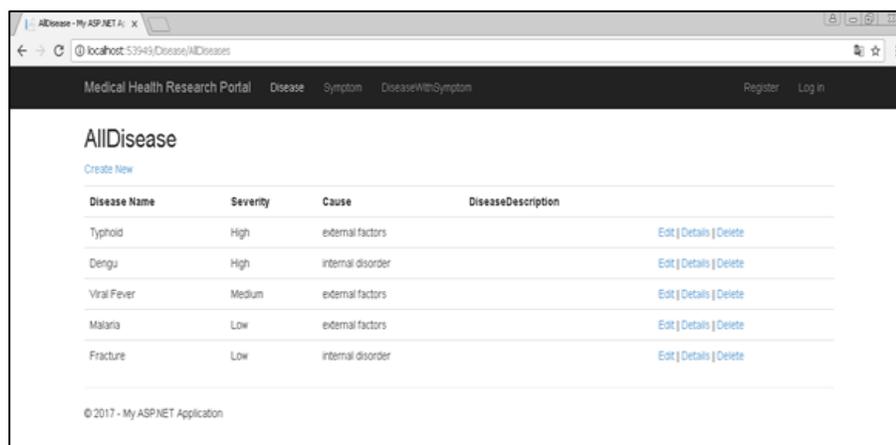


Fig. 7: Disease Portal

B. Use Case 2: Add Symptoms to the Disease

The input form contains the following fields:

- Disease Name: single-select list containing names of all diseases from the database.
- Symptom Name: text value (e.g. “fever”)
- Description: multi-line text value (e.g. “body temperature exceeds 102oC”)
- Save Symptom: button
- Cancel: button

The user provides the required details in the form and clicks the “Save Symptom” button. On successfully saving the details, the user is redirected to the home page, and the following message is displayed there: “The symptom <symptom-name> has been added to the disease <disease-name> successfully”.

Business Rules: A unique identifier is generated for the booking. One disease could have multiple symptoms (e.g. “fever” and “fits of shivering are both symptoms of the “malaria” disease). The combination of the symptom-name and disease-name should be unique.

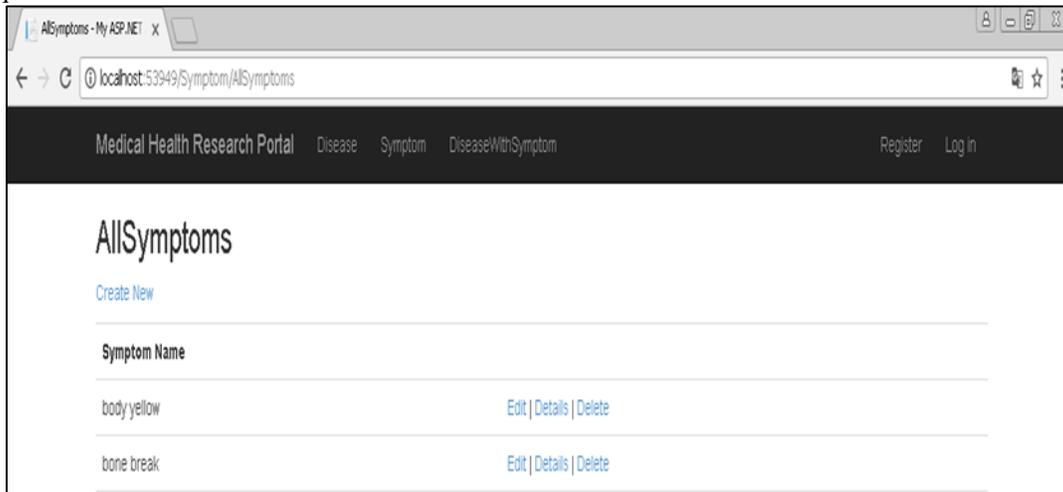


Fig. 8: Disease Symptoms Portal

C. Use Case 3: Patient Registration

Any patient who visits the Health care center will have to register and a unique ID will be generated which will be used for further correspondence. The patient must provide the details about him including:

- Patient Name: text Box
- Patient Address: Multi line text Box.
- Patient Telephone no:
- Emergency Contact No:
- Developers can choose few more fields based on the requirement.

Business rules: If the Patient name and the Address is already available, it should display the message saying that the Patient is already registered.

D. Use Case 4: Check the Patient

This part is used by the Doctors of the Application. NOTE: Only Doctors can visit this page, U should not allow patients or any other users to visit this page.

The user provides the following details:

- Patient Name: text value (e.g. “kumar”)
- Symptom Name: multi-select list, containing names of all the unique symptoms in the database (e.g. “Fever”).
- Diagnose: button

The user specifies the values for the input fields and clicks on the “Diagnose” button. The diseases with all the chosen symptoms should be listed as probable diseases the patient is suffering from. In the sample given (UI Screen Details), “fever” is a one of the symptom for both “malaria” and “common cold”. Finally the Doctor would analyze the patient based on the Symptoms and prescribes the required medicines.

1) *Input*

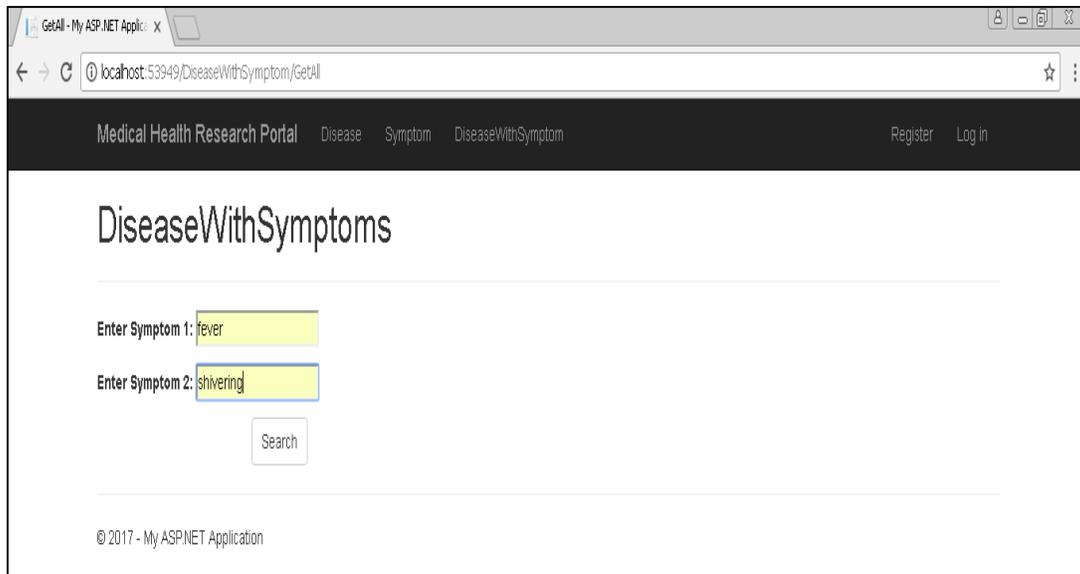


Fig. 9: Search Details by Disease Symptoms

2) *Output*

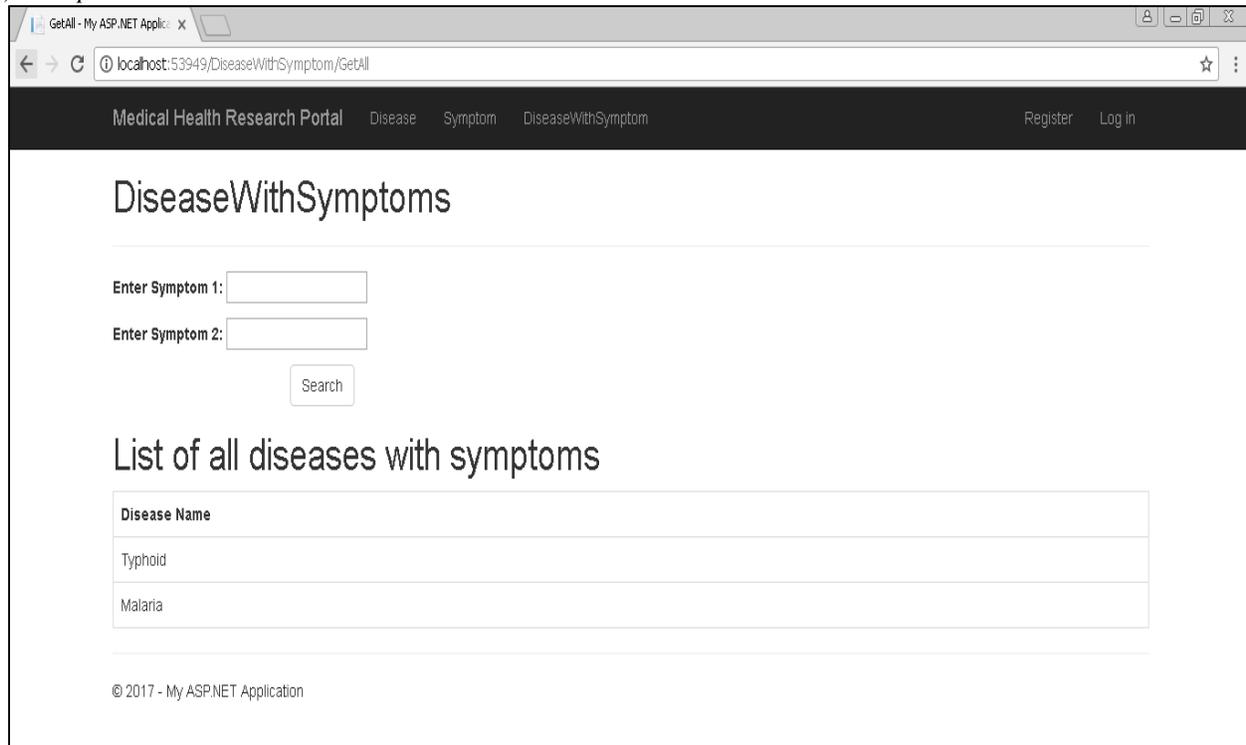


Fig. 10: Portal Result Showing All Diseases with Symptoms

V. CONCLUSION

The project Automated Medical Health Diagnosis System approaches the great level of success. Whatever the results desired has come with a great accuracy. It may happen sometimes that registered doctors login takes two attempts. Suppose when the doctor logins, his logins can be failed. After one or two attempt his login can be successful.

This system is secured and efficient enough as only registered doctors can enter or update data from anywhere. The system was given input by the registered doctor with some simple diseases. Accurate results were shown by this system. All the use cases are working well and the system can be deployed in many ways in health care industries.

REFERENCES

- [1] M. Thangamani, Dept. of Computer Science & Eng., Kongu Eng. Coll., Erode, India, Information Communication and Embedded Systems (ICICES), 2013 International Conference on Automatic medical disease treatment system using datamining, 29 April 2013
- [2] Hiroshi Takeuchi, Senior Member, IEEE, Department of Healthcare Informatics, Takasaki University of Health and Welfare. Engineering in Medicine and Biology Society, 2006. EMBS '06. 28th Annual International Conference of the IEEE, 15 December 2016, New York, NY, USA, Automated Healthcare Data Mining Based on a Personal Dynamic Healthcare System
- [3] Liehang Shi, Jianyong Sun, Yuanyuan Yang, Tonghui Ling, Mingqing Wang, Yiping Gu, Zhiming Yang+, Yanqing Hua+, and Jianguo Zhang, Three Dimensional visual patient based on electronic medical diagnostic records
- [4] http://en.wikipedia.org/wiki/Apache_Hadoops
- [5] Lovell NH, Magrabi F, Celler BG, Huynh K, and Garsden H, Web-based acquisition, storage, and retrieval of biomedical signals, IEEE Eng. Medicine and Biology, vol. 20(3), pp. 38-44, 2001.
- [6] Mazzi C, Ganguly P, and Kidd M, Healthcare application based on software agents, in Medinfo 2001 Proceedings, 2001, pp. 136-140.
- [7] Takeuchi H, Kodama N, Hashiguchi T, and Hayashi D, Automated Healthcare Data Mining Based on a Personal Dynamic Healthcare System, the 28th IEEE EMBS Annual International Conference, New York City, USA, 2006.
- [8] Giudici P, Applied Data Mining: Statistical Methods for Business and Industry: Wiley Publications, 2003.
- [9] Prather J, Lobach D, Goodwin L, Hales J, Hage M, and Hammond W, Medical data mining: knowledge discovery in a clinical data warehouse, in Proc AMIA Annu Fall Symp., 1997, pp. 101-5.
- [10] Churilov L, Bagirov A, Schwartz D, Smith K, and Dally M, Improving Risk Grouping Rules for Prostate Cancer Patients with Optimization, the 37th International Conference on System Sciences, Hawaii 2004
- [11] ASP.NET – Wikipedia
<https://en.wikipedia.org/wiki/ASP.NET>
- [12] ASP.NET Web API | The ASP.NET Site