

## **README**

### **PROJECT : - DOCTOR ON DEMAND**

#### **-Installations**

1. Apache Tomcat Installed
2. MySQL database Installed
3. MongoDB Installed
4. Anaconda installed to work on jupyter notebook

#### **-End to End tasks and activities to start the website: -**

1. Start the Apache Tomcat server from cmd.exe from C:\apache-tomcat-7.0.34\bin with command startup.bat
2. Start the MySQL server by starting the MySQL notifier.
3. Start the SQL editor(MySQL workbench)-Database Name doctorondemand
4. Start the MongoDB server from C:\Program Files\MongoDB\Server\3.2\bin by executing mongod.exe
5. Start the Mongo DB shell from C:\Program Files\MongoDB\Server\3.2\bin by executing mongo.exe
6. Open cmd.exe, change path to  
C:\apache-tomcat-7.0.34\webapps\DoctorOnDemand\WEB-INF\classes
7. Compile all the java files by javac \*.java
8. Start the Anaconda Prompt and type jupyter notebook .
9. Run scripts DODDealMatches and ProductRecommendations to get deals and recommendations.
8. Open the browser and add the below address  
<http://localhost/DoctorOnDemand/Home>

#### **-Deployment:**

Once the website is opened Homepage will give idea of website and can get access to all the tabs and services.

#### **-Other Files:**

1. Recording which has a demo of project
2. output.pdf has screenshots of important interface added in website.

---

-

#### **-Functional Features implemented in the project:**

---

-

1. The intent of the project is to build servlet based web application that will allow patients to book appointments online from DoctorOnDemand website.
2. There are 3 users in our application-Patient, Doctor and Customer Specialist.

- 2.The patient can book appointment as per requirement in various categories namely
  - a)Physician -It includes subcategories namely Pediatrician,Dermatologist,Psychiatrist
  - b)EyeCare-It includes subcategories namely Optometrist,Optician,Orthoptist
  - c)Dental-It includes subcategories namely General Dentist,Orthodontist ,Endodontist
- 3.Patients can pay for the booking using the credit card, Name, Email, Phone Number and Address
- 4.Patients can view there bookings available by clicking on view order.
- 5.Patients can also write a review for a particular service offered and can also view the reviews given by other patients.
- 6.Patient can cancel and change the transaction before the appointment is scheduled .
- 6.Doctors can add the services according to the categories.
- 7.Doctors can add, modify and delete the existing products.
- 8.Customer Specialist can keep track of the appointments and offer support to both doctor and patient through email and phone.
- 9.The various services can be searched using the search tab implemented using autocomplete servlet and Ajax.

## **Role Information**

---

There are three ROLES:

- 1.Patient
- 2.Doctor
- 3.Customer Specialist

---

-

### **1.Patient**

- Create account
- Login
- Book appointments
- Make Transactions by checking out by giving user information like credit card info,Name ,Email ,Phone Number and Address
- Once appointment is placed can check confirmation through View order
- Cancel the appointment
- Write Review
- View Reviews
- View Trending Services

### **2.Doctor**

- All features of Patient are implemented here with additional features namely
- Add ,Update and Delete a service
- Can view the Inventory Report which
  - 1.Includes appointment information for services

2.Includes appointment chart for all services

-Can view the Sales Report which

1.Includes sales information for services

2.Includes sales for every service

-Can view Trending services

-Can view Data Analytics Report which shows

1.All services and their ratings

2.Shows services with ratings more than 3

3.Shows services with price above \$200 and rating 5

### **3.Customer Specialist**

-Same functionalities as patient are implemented plus can also keep track of Appointments and provide support to doctor and patient through email and phone

**Note:**Services added by doctor can be viewed by logging in as patient or customer specialist

### **Additional features implemented**

-Search Tab using Autocomplete Servlet and Ajax.

-Deal Matches

-Recommender Feature for recommendations to various users

-Google Maps to locate a clinic

- 
- Total java files-56(Around 6757 lines of code)
  - Total Java script files-2 (Visualization-chart-script and javascript- 206 lines of code)
  - Python files-2 (DODDealMatches and ProductRecommendations)
  - HTML Files-5
  - XML files-2(Product Catalog and web)

---

### **MYSQL Queries**

---

-

1.Database used is doctorondemand

#### **COMMAND:**

create database doctorondemand;

2.Check if it is created using

#### **COMMAND:**

show databases;

3. Use the created database

**COMMAND**

use doctorondemand;

4. Create a service details table

**COMMAND**

create table servicedetails(ServiceType varchar(100),Id varchar(200),servicename varchar(100),clinicname varchar(200),DoctorName varchar(200),Price double,TypeImage varchar(200),Type varchar(200),Location varchar(400),Discount double);

5. Check the service details table as and when a service is added or updated using following command.

**COMMAND:**

select \* from servicedetails;

6. Table to check the appointments booked

**COMMAND**

create table orders(orderId integer,userName varchar(100),orderName varchar(40),orderPrice double,userAddress varchar(100),creditcardNo varchar(100),date varchar(100),email varchar(100),phonenummer varchar(100),Primary key(orderId,userName,orderName));

7. To check orders

**COMMAND**

select \* from orders;

8. Registration table name insertuser

**COMMAND**

Create table insertuser (firstname varchar(40),lastname varchar(40),username varchar(100),password varchar(40),repassword varchar(40),usertype varchar(60));

9. To check insertuser

**COMMAND**

select \* from insertuser;

10. Record table to track appointment booked, sales and discount

**COMMAND**

create table records(Id varchar(300), servicename varchar(300),Price double, Appointments int,Sales double,Discount double);

insert into records(Id,servicename,Price,Discount) select Id,servicename,Price,Discount from servicedetails;

update records set Appointments = 0;

update records set Sales = 0;

11.To check record

**COMMAND**

select \* from records;

12.To import database

1. Click Manage **Import** / Export under Server Administration on the right of the **Workbench** window. ...
2. Select your **database** and click OK.
3. Enter your **database** password if prompted.
4. Select the **Import** from Disk tab

---

## MONGO DB COMMANDS

---

1.To see the existing database

**COMMAND**

show dbs;

2.Use the database

**COMMAND**

use CustomerReviews;

3.Create collection

**COMMAND**

db.createCollection("myReviewsDoctor")

4.To see collections

**COMMAND**

show collections;

5.To see the reviews

**COMMAND**

db.myReviewsDoctor.find()

6.Import the collection

**COMMAND**

mongoimport --db test --collection collectionname  
--authenticationDatabase admin --username <user> --password <password>  
--drop --file ~\filename.json