Department of Computer Science

Batch - 2013

Online Appointment System

By

Syed Nazir Hussain 13B-049-BS

Syed Umer Wahab 13B-033-BS

Supervisor: Parkash Lohana

Project report submitted in partial fulfillment of the requirement for the award of the Degree of **BS (CS)**

Karachi
July – 2017

ST-13, Block 7, Gulshan-e-Iqbal, AbdulHasanIsphani Road,
Opposite Safari Park, P.O. Box – 75300, Karachi, Pakistan.
Phone: 34978274-5; 349904305, 34982476
http://www.uit.edu.pk

CERTIFICATE

The undersigned certify that they have read & accepted the Final Year Project Report entitled '**Online Appointment System'** submitted by **Syed Nazir Hussain 13B-049-BS** And **Syed Umer Wahab 13B-033-BS** in conformity with the requirements for the degree of Bachelor of Science in Computer Science.

ABSTRACT

This system online appointment idea can provide a great facilities with its existence. Now a day we are living in a 21th century, an era of digital technology or if I say today a man is incomplete without accessing the application of technology in his/her life keeping the technology factor in mind we are working on an online application system which can be beneficial for large group of people of different work nature. This system will also facilitate people according to their perception so our main goal is to meet the right client like (Doctor) to the right

User to gain good user experience and it's profitable for both of them.

CHAPTER 1

INTRODUCTION

Introduction

This online appointment system deal with a great deal of facilities with its existence. We are in an era of digital technology or if I say today a man is incomplete without accessing the application of technology in his/her life. Keeping the technology factor in mind, we are creating one online application, which can be beneficial for large group of people of different work nature. Especially the category of work people who requires prior appointment before meeting with the person for Instance like doctor by using our online system one can get the online appointment sitting in any part of the country by just following the simple process.

Problem Definition

For last few months, I had an experience of visiting hospital frequently. There I see a great rush of people who are coming for emergency who are coming to see their doctors and most of the people who are coming to with a specific doctor. There I feel very painful when see a people in a tired condition and just asking the advance appointment with the doctor. As now a day we are living a very busy society where everyone is busy in his daily earning. Everywhere a great deal of traffic jam, bad city condition, violence and what not. In such a miserable condition I feel we can have some system which can relief the people at least those who are visiting the hospital for just looking the advance appointment.
On the other hand if we keep the daily routine and busy life on one hand the other factors in our society is the traditional cultures. I am trying to point out the very important issue of my society which is, most of the ladies are not allowed frequently to visit hospitals.

Suggested solution

Keeping the mentioned problem in our mind we are creating an online appointment system, which will bring the relief in the life of users and also it can bring the relief for hospital staff who are making advance appointment at hospital.

In our online appointment system there will be two parts the first part will be from doctor side to register doctor in our system. The second part will be for users the user will also register them on the system. They will find the registered doctor to make an appointment

Process flow

The doctor will visit our online web-site and will register them on our site. Our web system will make proper validation by means of NIC no or PMDC ID that the registered users by doctor. By this validation on scan user can registered themselves as a doctor in our system. On the other end the users will see the registered doctor and they can make the appointment of any available dates after making an appointment they asked doctor will confirm the appointment and then users will get the notification of appointment is confirmed

Option for payment

The doctor will have the option of payment before meeting the user. They can take the payment via

- Easy paisa mobile account
- Mobi cash/Easy paisa etc.
- Credit card (assumption)

Maintain records

During appointment some option at doctor end is prescription and track a patient. The doctor can able to track his/her specific patient and can able to see patient medical history as how many time he/she has appointment with me and other is prescription .In prescription option can contain all medicine and its related test so the doctor can select and referred medicine according to their experience in prescription area and after that the doctor will send this prescription. The prescription will be received at patient end with corresponding doctor detail so he/she can easily access required medicine from any store and it can also be beneficial for his/her to maintain his medical history.

CHAPTER 2

LITERATURE REVIEWS

An online appointment system which will book prior appointment with the doctor.
The user will not need to visit the hospital to take the appointment. And Doctor can track user by viewing records which are mentioned by our system.

Existing system

Our system is not match any of the existing system. As there are many system or website which are now existing in our country to make online appointment.
E.g. Findadoctor.com.pk, onlinedoctor.com etc.
Our system is providing the feature of validating the doctor by means of PMDC or NIC number. The above mentioned online appointment system which are already existing does not contain any validation. Anyone can register himself as a doctor there will be also no mean of security When you are making the payment.

Hospital system existing

There are many big and renowned hospital website are available for maintaining their appointment online in advance like Aga khan university hospital, Al-shifa trust Islamabad and tabba heart etc. but these website are only managing and targeting the patient of their own domain. Our system is to facilitate or will target users from all over Pakistan.

Third party involvement

As in our system we are making payment option via automatically generated email or sms to the corresponding user to pay fees of that doctor if you want to confirm your appointment within time limit that is mentioned in term and condition at doctor end so we are depending on third party. We will make payment by

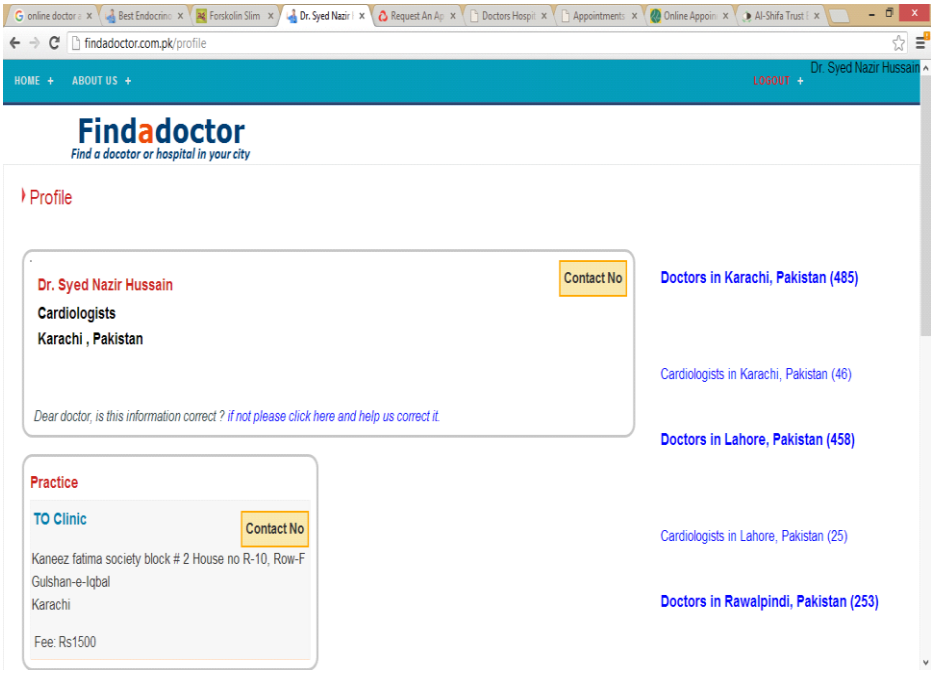
- Easy paisa mobile account
- Mobi cash/easy paisa etc.

So mobile companies like Telenor, mobilink will be our stakeholder.

Stake-Holder for project

The appointment stakeholder for system will be the user who will be benefited by taking advance appointment. The doctor are also our stake holder as they will be directly impacted by this system. Mobile service provider will also be stakeholder.

Available product



Feature:

-View a map of doctors in your insurance network.

-Read patient reviews to help you choose the right doctor.

-See any doctor's available times and click to book instantly

Pros: Findadoctor.com.pk is very simple site to search for doctors near to your home.

Cons: Findadoctor.com.pk does not make any warranties about the completeness, reliability and accuracy of this information.It doesn't maintain any history or record of any activity.

Company Name: TechGallop (Ptv) Ltd

CHAPTER 3

SOFTWARE AND HARDWARE REQUIREMENTS

Hardware Required

Operating System (OS)	Hardware
• Microsoft Windows XP/ 7 (32 or 64 Bit)	• 1.5 GHz 32-bit (x86)/64-bit (x64) or higher • 512 MB RAM (32-bit) or higher • 2GB HDD free • Internet modem

Table 2.1 Hardware Requirement

Internet Modem

This is required to use the internet.

Operating System

Minimum Requirement is WIN 7 & 8 b/c it is a web application so it required a high speed Browser like GOOGLE CHROME AND FIREFOX new versions these version only required to install in this operating systems.

Operating Environment

This product will be developed using Open Source server side language PHP so, we would preferably use Microsoft Windows Operating System for developing this software.

Hardware Interface

- **Display Screen**

  Screen resolution of at least 800*600 or above will be preferable for viewing.

- **Design and implementation constraints**

  For ensuring platform independence of the software the implementation will be on PHP so the end users system must have a Windows / Linux environment.

Software Required

Programming Environment

The following is the development environment of what we will be developed:

- Html
- CSS
- JQuery
- PHP
- MYSQL work bench
- TextEditor(Notepad ++,sublimetext)
- Web browser
- WAMP (Windows Apache MySQL PHP) Server.

Platform

Windows 7 or higher.

User Documentation

A user document should be provided at the end of the development. It should have the following

- A well-documented user manual.

Assumptions

Based on current scope of our Project, the assumptions are listed below. If an assumption is invalid later, then the activities and estimates in the project plan will adjust accordingly.

As our project is, allowing payment mechanism through Easy Paisa mobile account and Mobi Cash prior to the meeting with the doctor. An assumption is there to Integrate bank Online Payment System. If a particular bank allows us to use Its API for integration, we can add this feature.

Another assumption is that we can achieve direct verification from PMDC Pakistan against the PMDC ID obtain from doctors at the time of registration. This verification is subjected to accessing the API for PMDC and can be obtained through proper process with PMDC.

Dependencies

- This software would need PHP runtime environment and some additional packages for working.

- It will need a web browser for viewing.

CHAPTER 4

Methodology

Before starting the project we made the complete analysis for the people who will use the system and will be benefited by using this facility, after this we made the paper based prototyping and sequentially integrate the pages.

As now the prototype is completed, we figured out the inputs and possible outputs. We also marked those inputs which are required (**Mandatory**) by the system in order to complete the transactions like for example NIC number, mobile number are required for payment and tracking the User for any sort of query.

As the Doctor is the main business user of this system. Therefore, after the analysis process we have created the complete mechanism for the verification and validation of the Doctor. As currently we are developing the system for university FYP, the scope is smaller for now, therefore, the current scope of the project treat the current user as an admin to verify the Doctor. For future, we can extend the Admin users by adding the flow for registration process of admin.

After the complete analysis of the project, we entered the development phase. After starting the development phase, we started working for the design and structure by using HTML and CSS. Additionally we have taken care of modern use of technology as nowadays mobile and tablet users are in large quantity, so we have created the design that is user responsive and to be fitted, in either screen of mobile, tablet or computer.

After completion of front-end layout, we have started working over database. We have started created database structure; we have created paper based ERD model design and then performed normalization of all forms like 1NF, 2NF and 3NF. After normalization, we have started creating database on the structure, which has developed by normalization process.

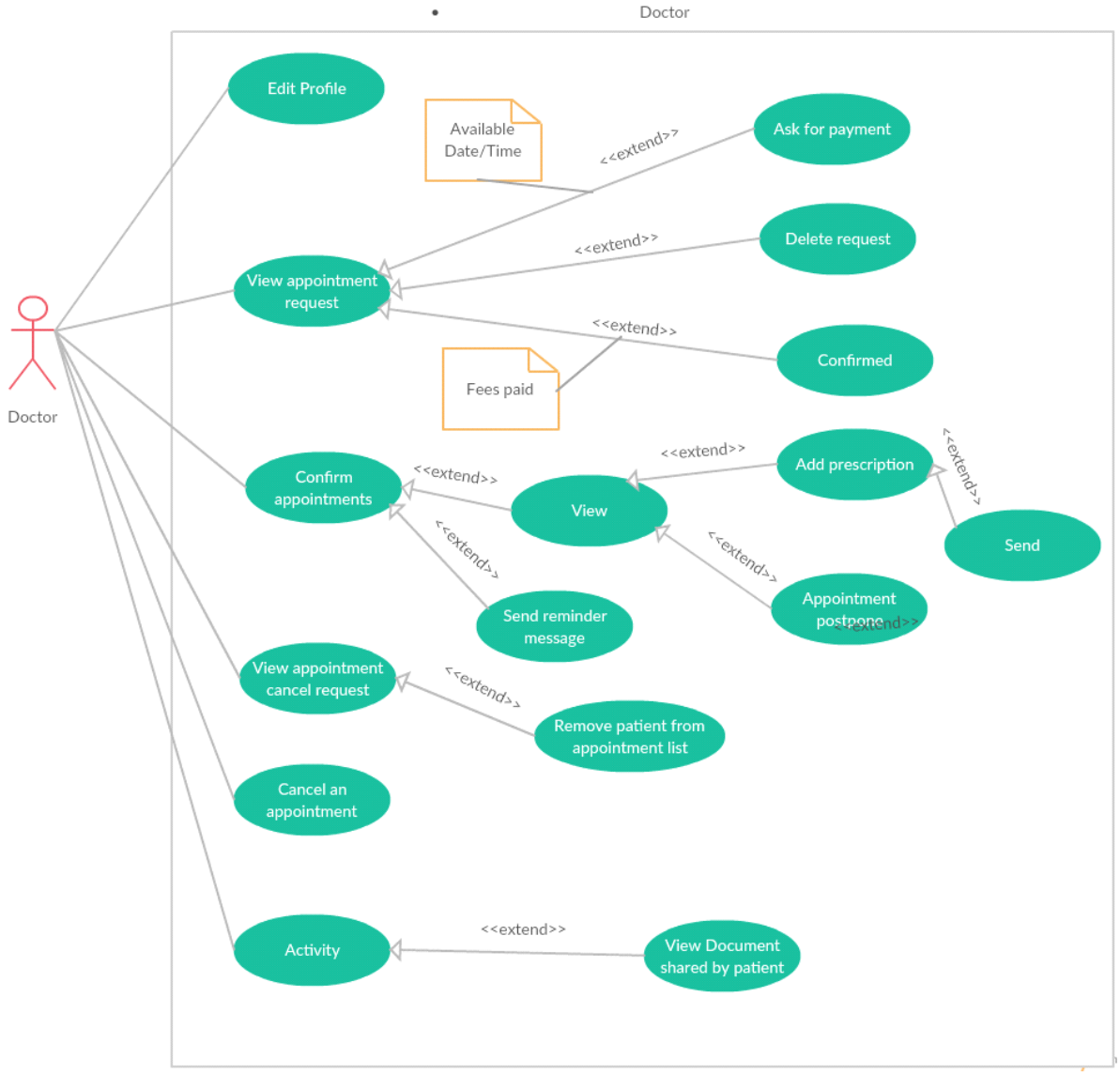
In the end after completion of database creation, we have inserted dummy data to test the queries and verified the field sizes to avoid runtime data lose.

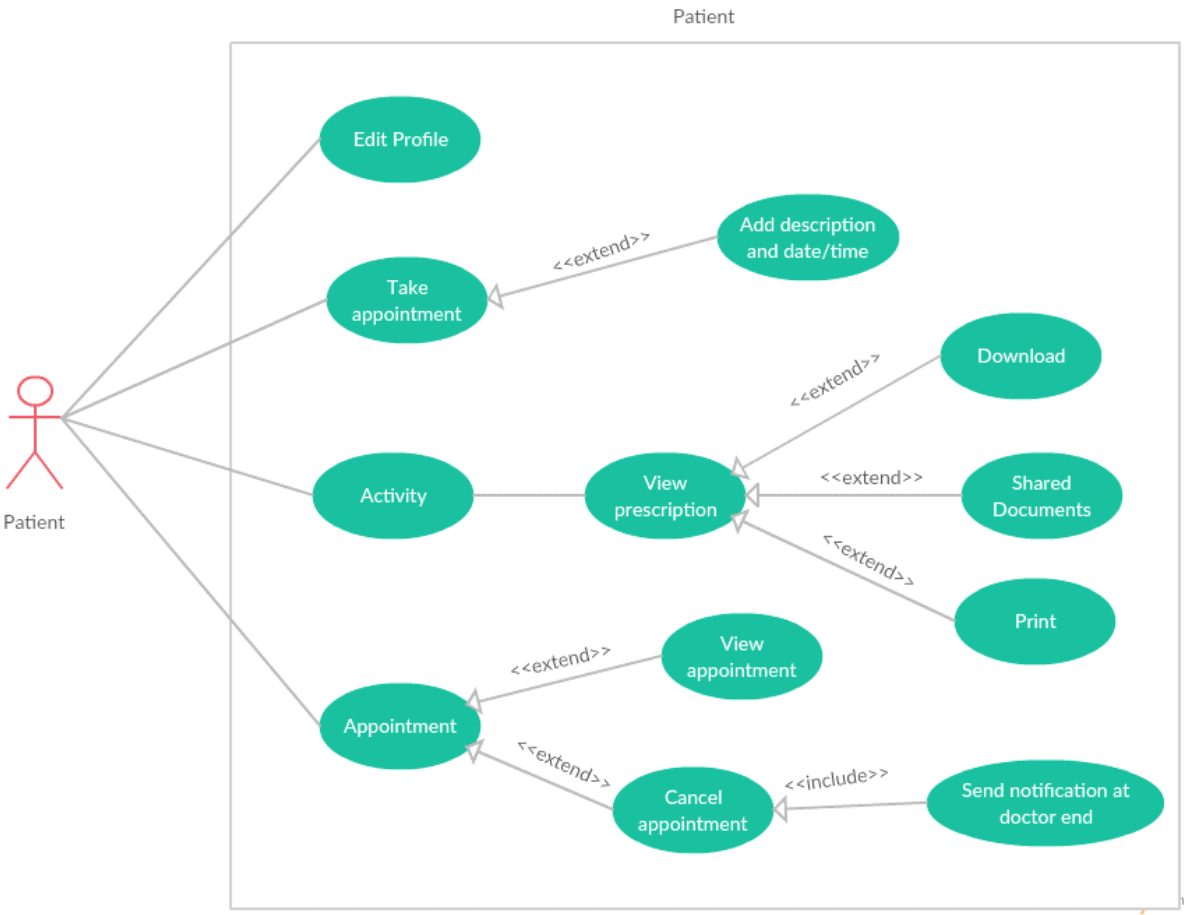
Then we started the coding phase, in coding phase we design the classes according to my database structure and defined the relationship among the classes, we have defined every method and properties, which will be used in database for every class. After this, we performed unit testing and fixed the issues and bugs appeared during unit test .We performed some basic operations like login Signup for the testing of flow.

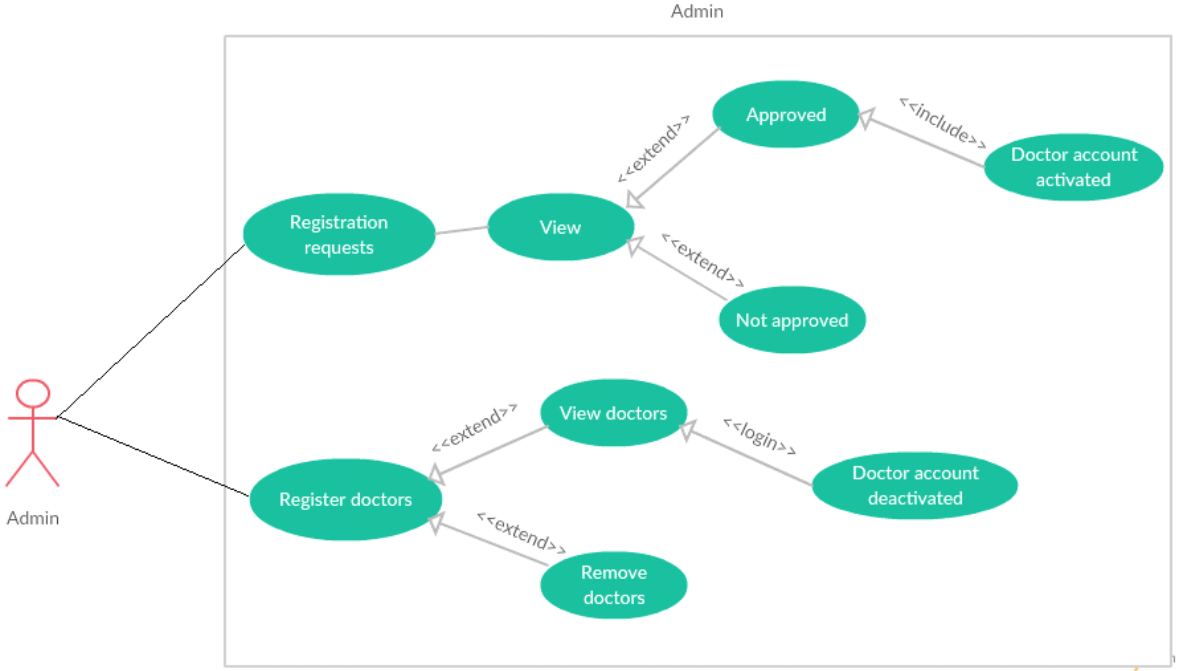
CHAPTER 5

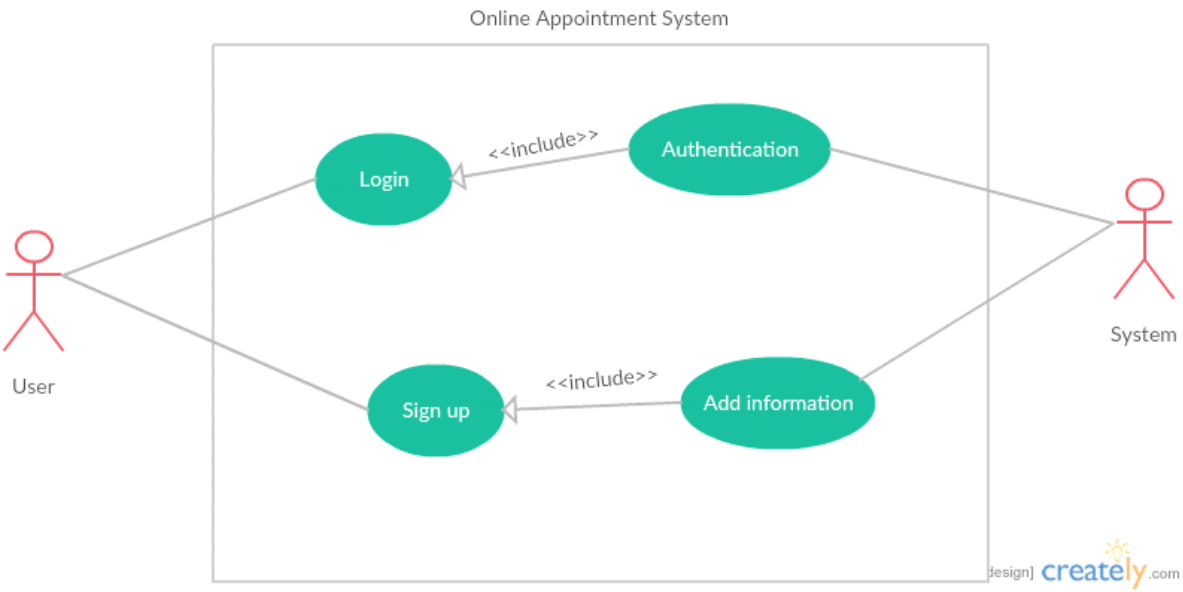
UML Diagrams

Use Case

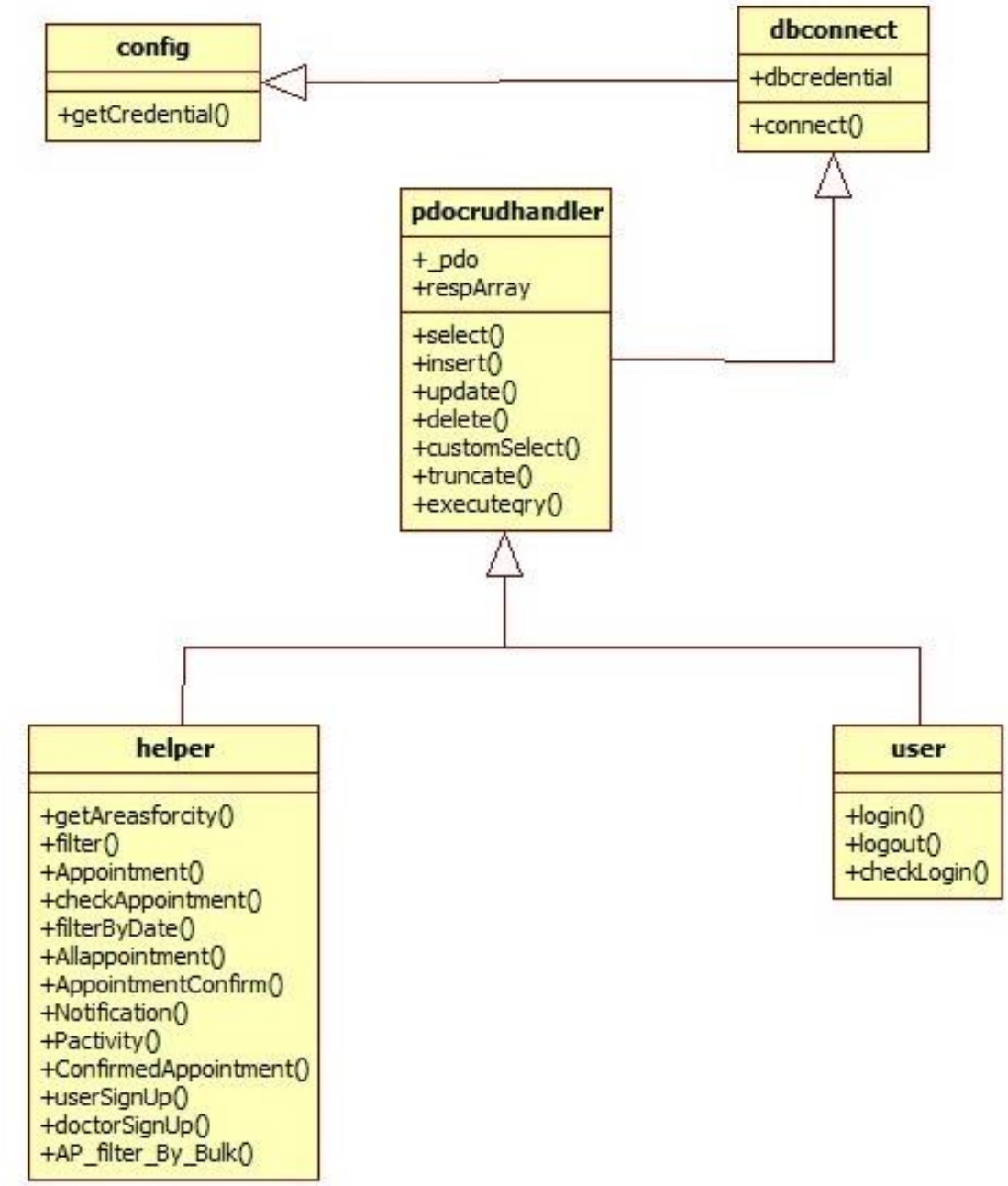

Doctor
Edit Profile
Available Date/Time
Ask for payment
<<extend>>
Delete request
<<extend>>
View appointment request
<<extend>>
Confirmed
Fees paid
Doctor
Confirm appointments
<<extend>>
<<extend>>
View
<<extend>>
Add prescription
<<extend>>
Send
<<extend>>
Send reminder message
<<extend>>
Appointment postpone
<<extend>>
View appointment cancel request
<<extend>>
Remove patient from appointment list
Cancel an appointment
Activity
<<extend>>
View Document shared by patient


Patient
Edit Profile
Take appointment
Add description and date/time
<<extend>>
Activity
View prescription
Download
Shared Documents
Print
Appointment
View appointment
Cancel appointment
Send notification at doctor end
<<include>>


Admin
Registration requests
View
Approved
Not approved
Doctor account activated
<<include>>
Register doctors
View doctors
Remove doctors
Doctor account deactivated
<<login>>


Online Appointment System
Login
<<include>>
Authentication
User
System
Sign up
<<include>>
Add information
[design] creately.com

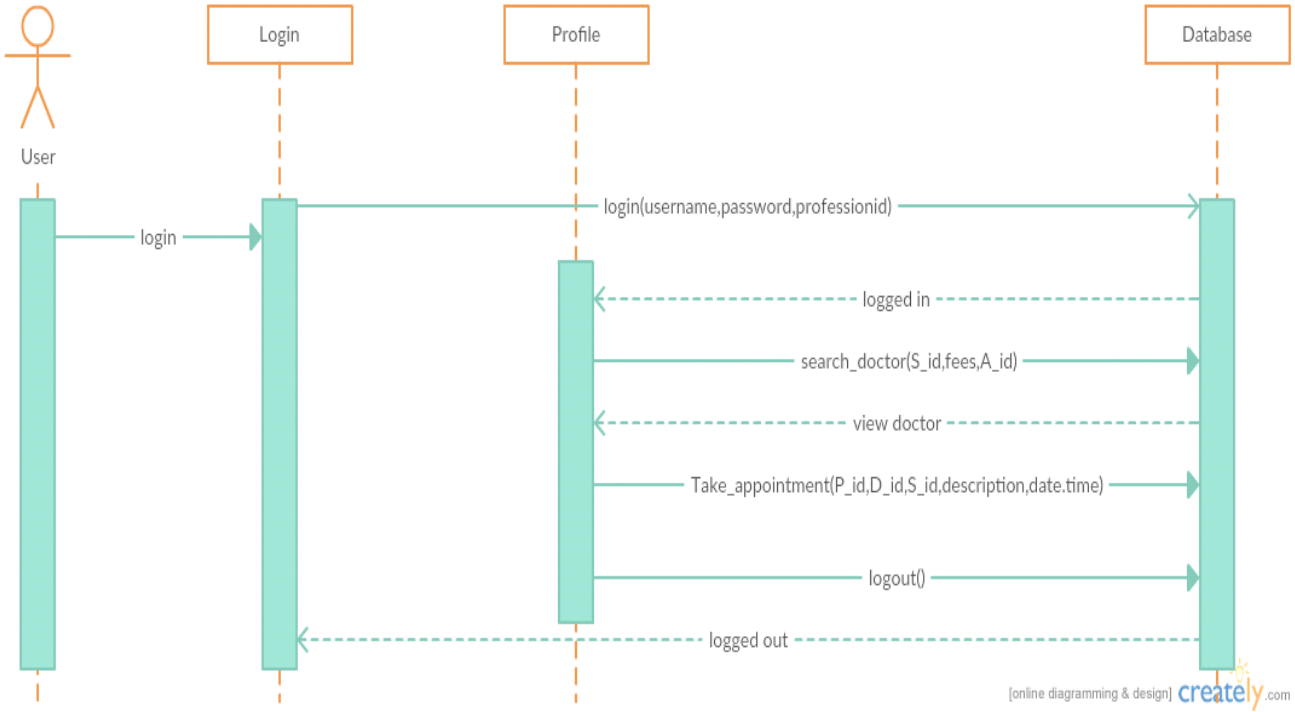
Class Diagram



The above mentioned class diagram contain the information and status of my current performance and project works.

Sequence diagram

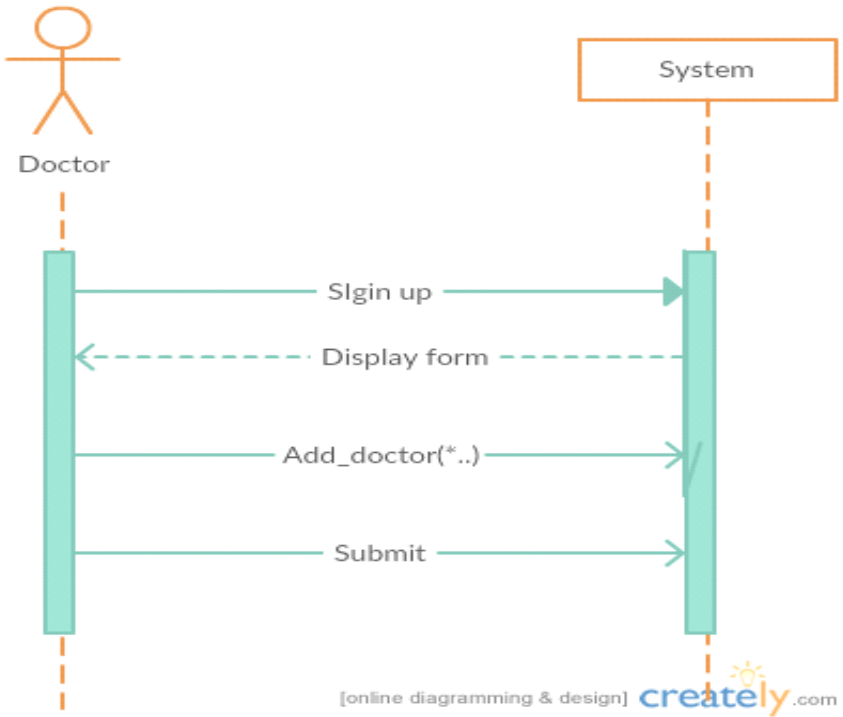
- **Take appointment**



- **Login** Will accept the User name and password.
- If the user is validated the Database will return the status of correct credentials and user will logged In
- **Serach Doctor** will be perform the searching of a doctor by means of Doctor fees , Doctor ID , Doctor Specializations

- **View Doctor**: User will view the list of all doctors which comes under his/her search criteria and can select the doctor.
- **Take Appointment**: User can finalize the appointment of selected Doctor.

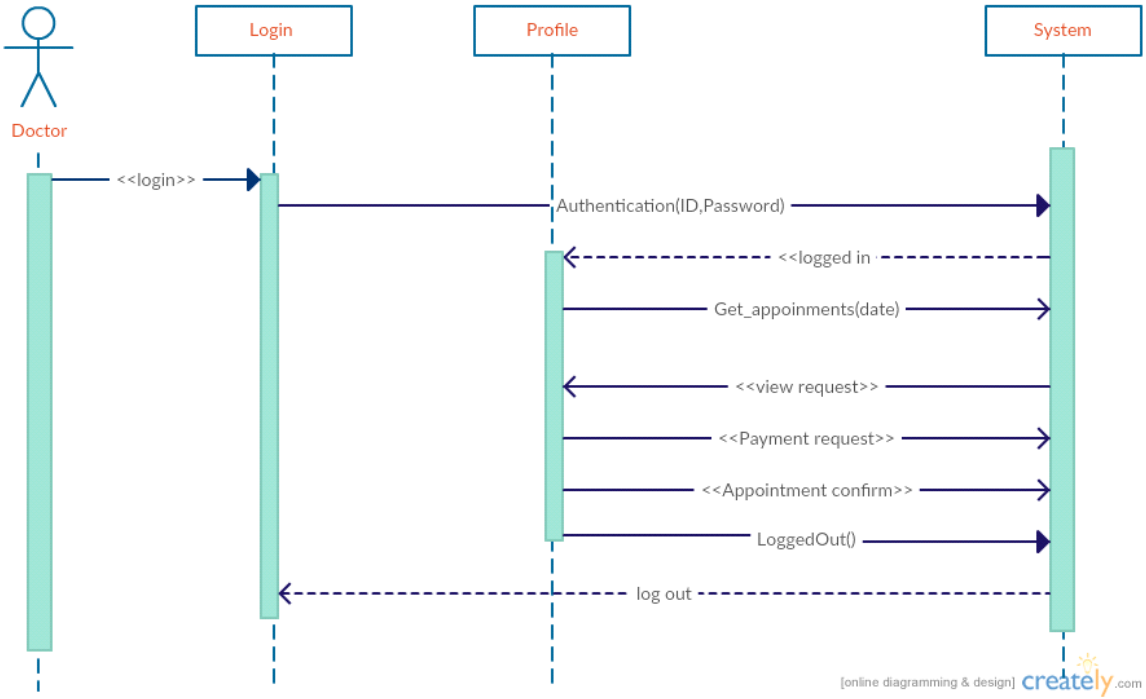
- **Doctor registration process**



- **Signup:** Doctor will sign up.
- **Display Form:** The registration form for doctor, which will contain all its basic information which is required by the system. And further details like

Specialty, Timings, Days Off

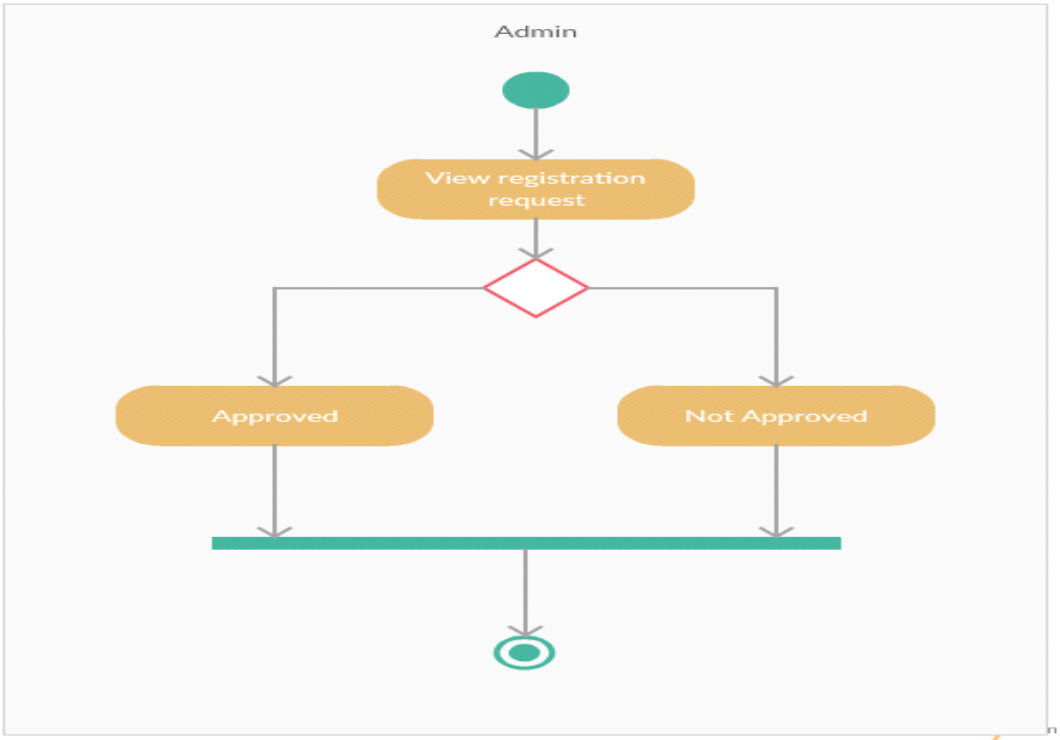
- **Doctor approved appointment request**



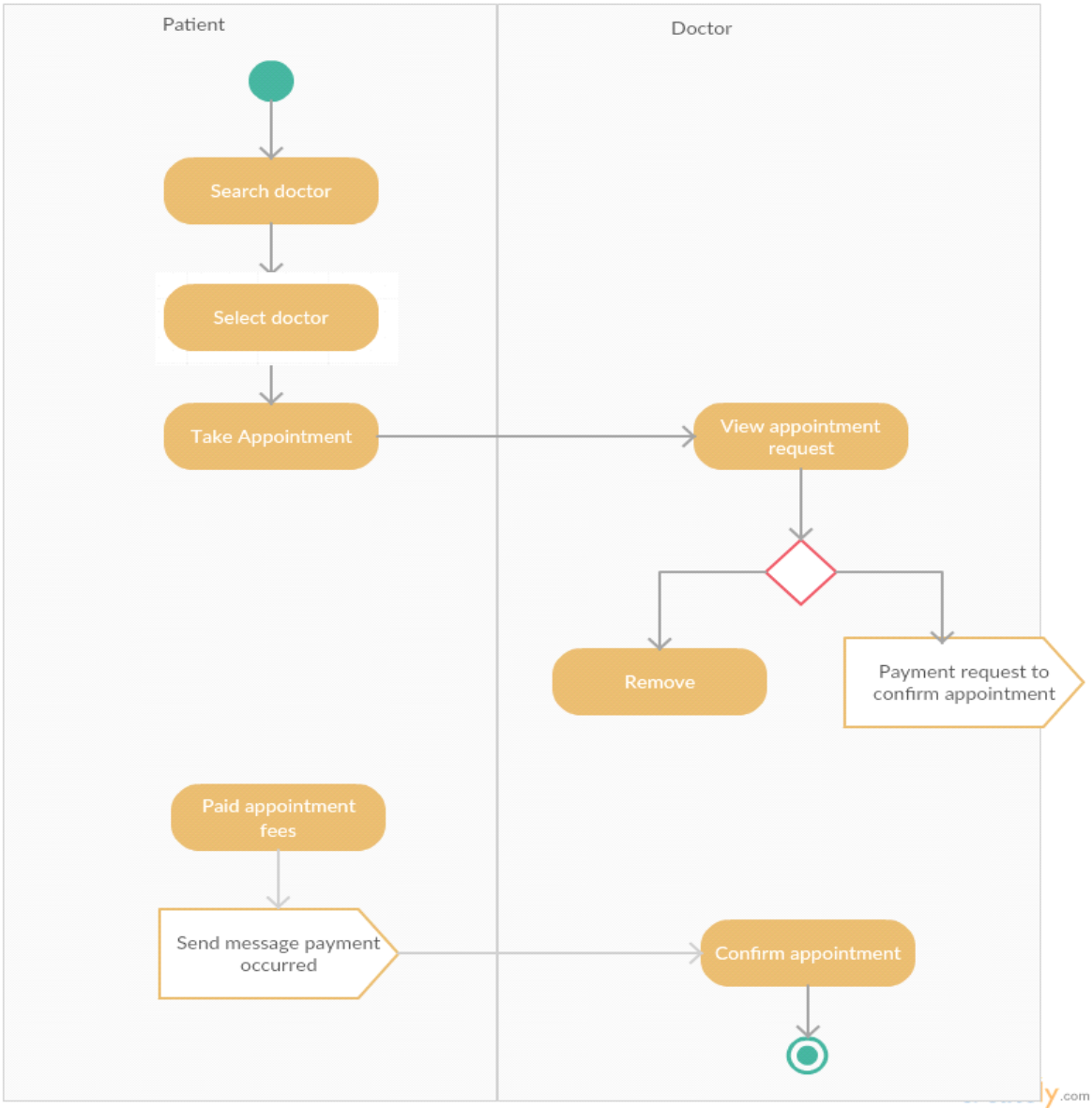
- **Login** Will accept the User name and password .
- If the user is **validated** the Database will return the status of correct credentials and user will logged In
- **Get Appointment :** Doctor will see the appointments for specific date for him
- **Payment Request:** Doctor will request the for payment.
- **Appointment Confirmation:** After payment has been received , doctor will confirm the appointment .

Activity diagram

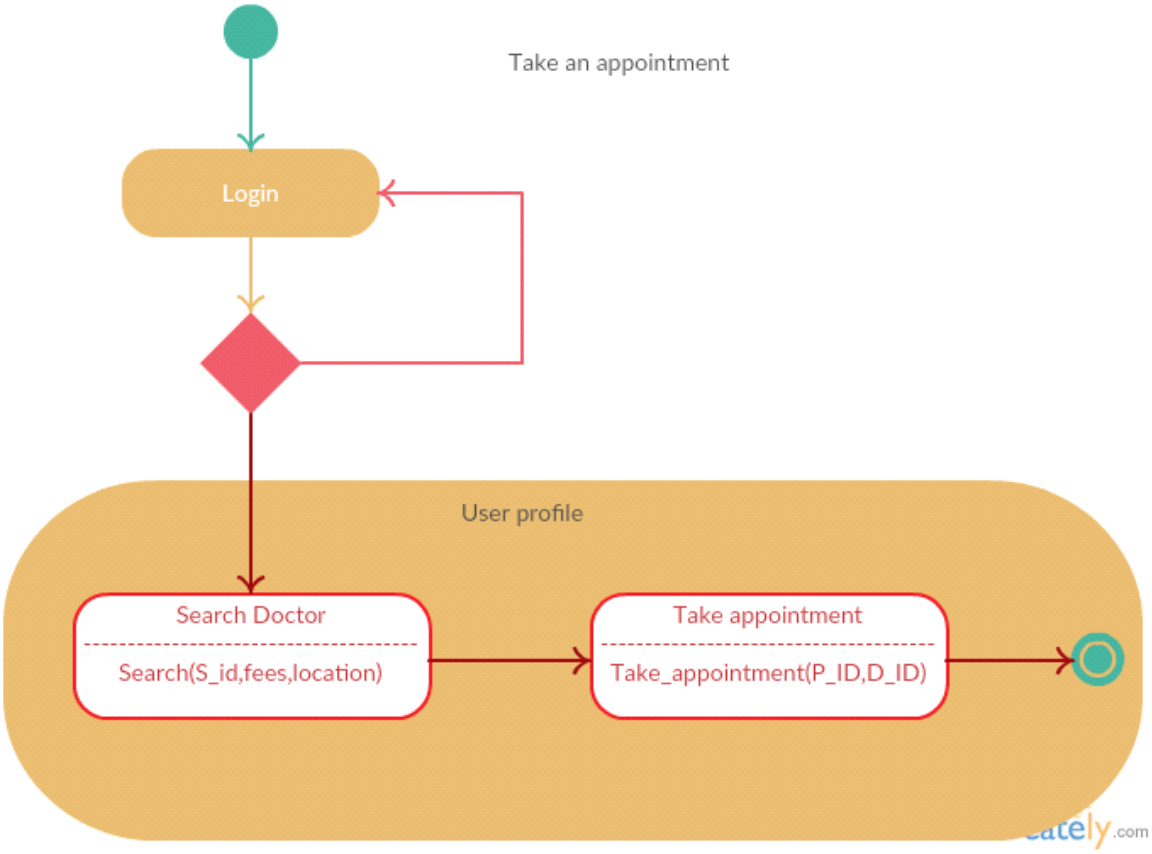
- **Registration checking process at admin end**



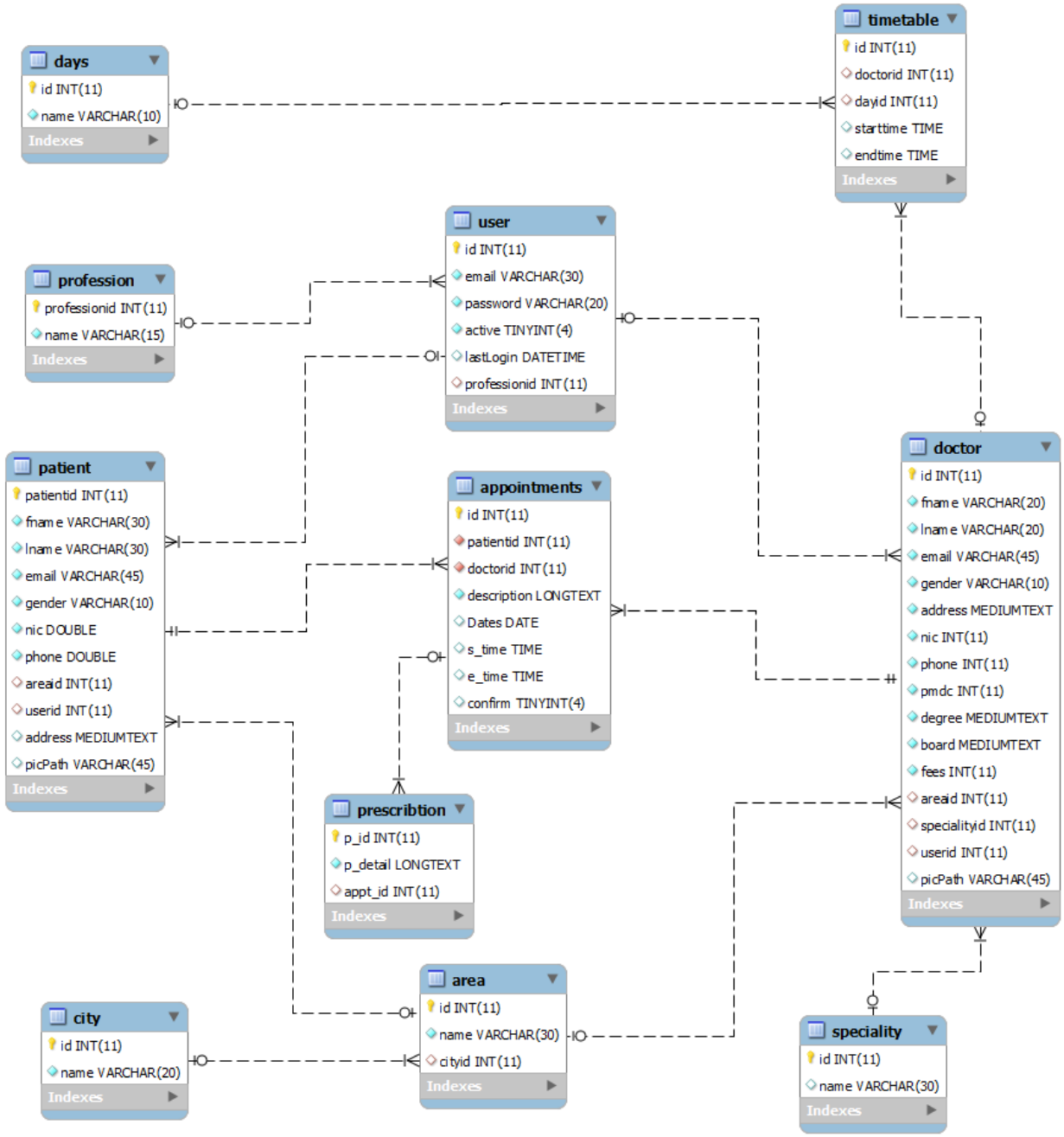
- **Taking appointment process**



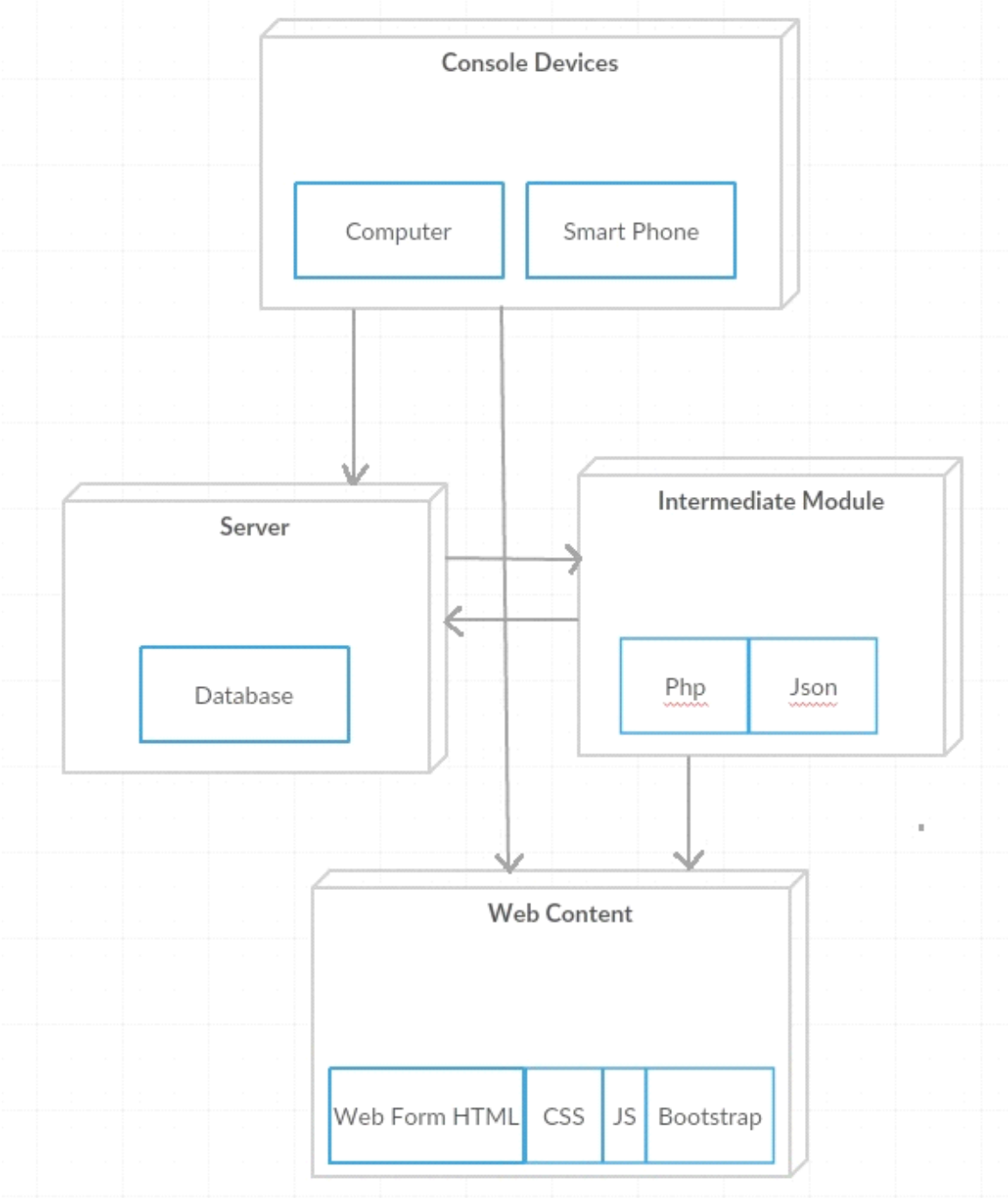
- **State Machine**



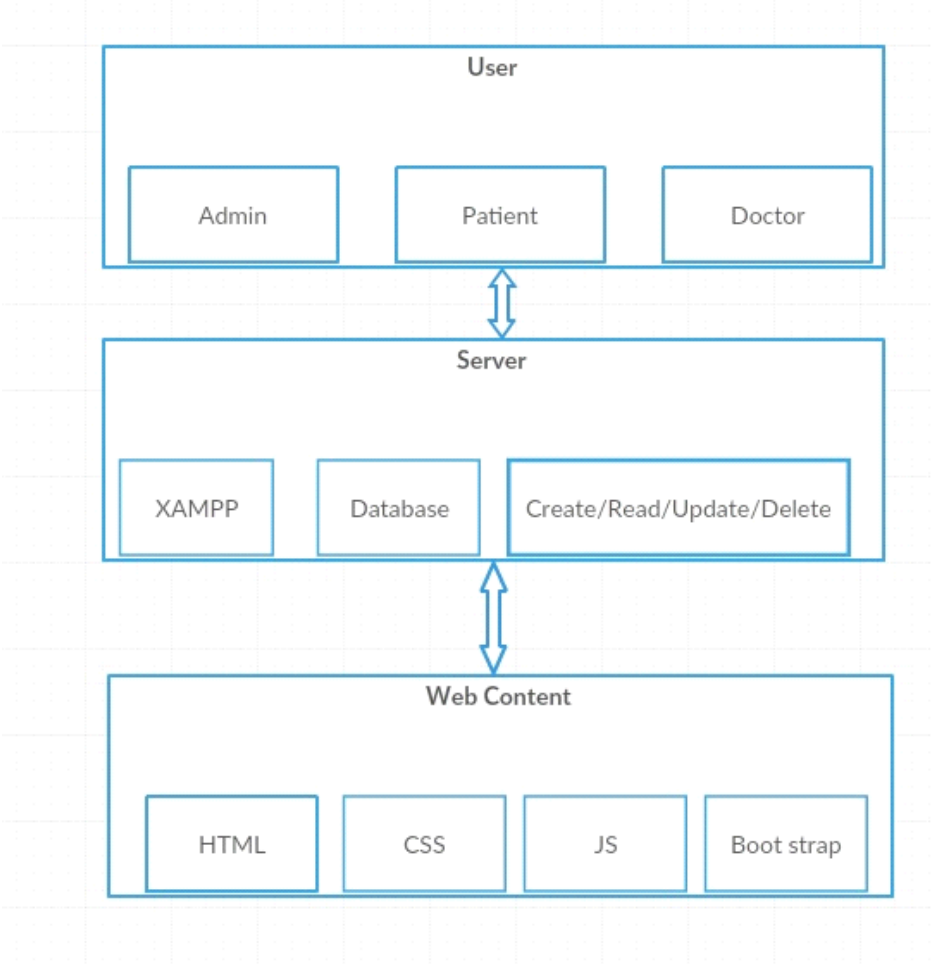
- **ERD model diagram**



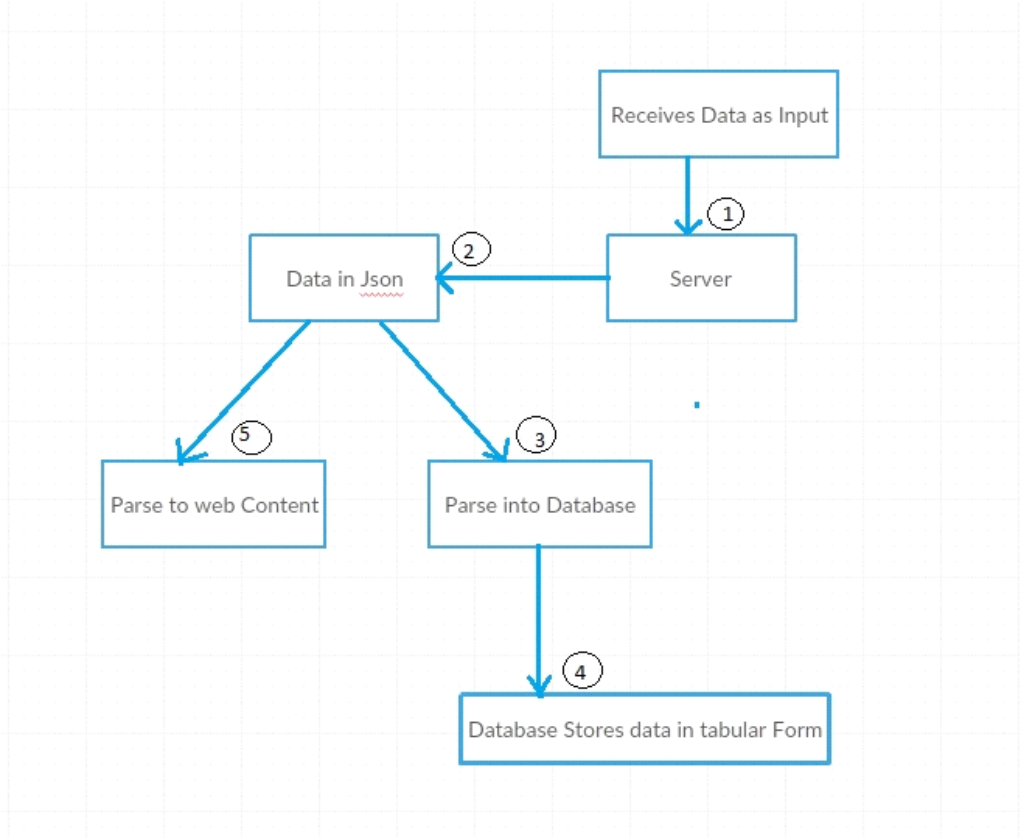
Deployment Diagram



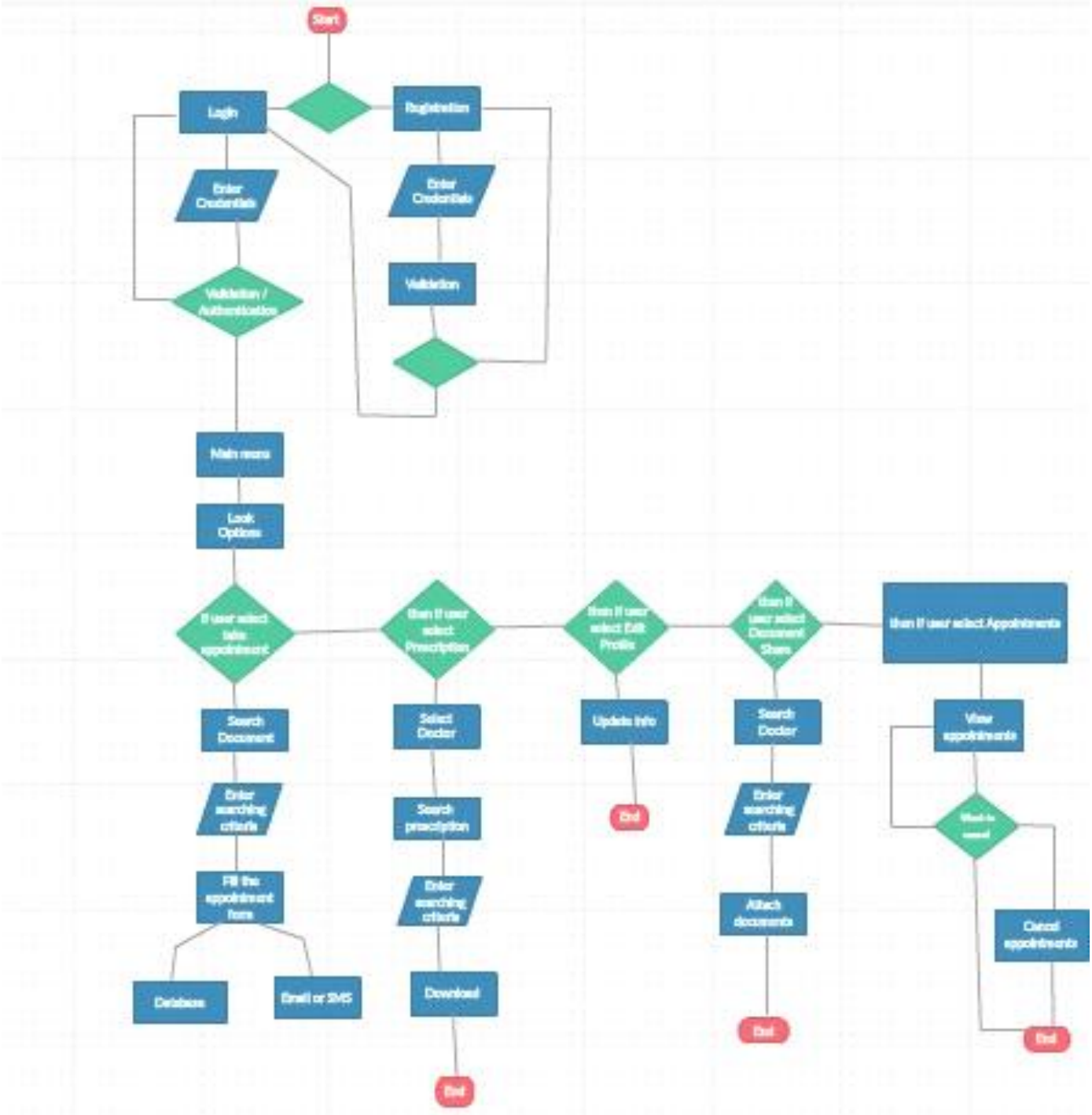
Architecture Diagram



Data Flow Diagram



Flow Chart (Admin)



CHAPTER 6

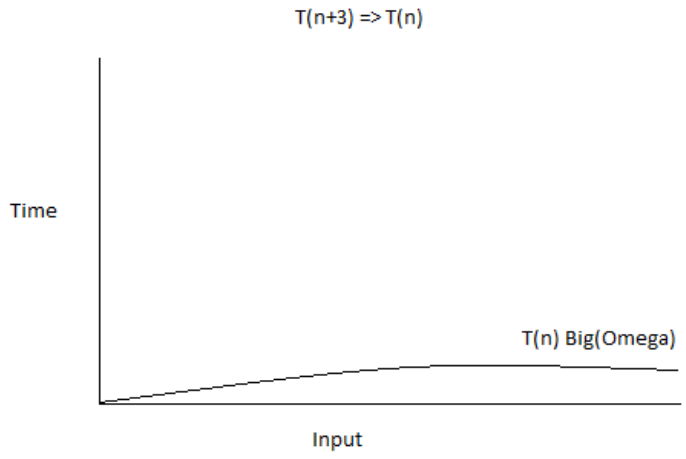
Algorithm Analysis & Pseudo Code Complexity

Algorithm pseudo code

Helper class check Mandatory Parameters

1. If set (POST['call']) AND set (mendetoryParam[POST['call']]))
2. declare data = array()
3. declare missingFields = array()
4. set flag = true
5. Repeat step 5 to 8 until mendetoryParam[POST['call']] has item
6. If not set POST[item]
7. Set flag = false;
8. missingFields[] = item;
9. if count(missingFields) > 0 then
10. set data['status'] = false
11. data['error'] = 'Required parameter(s) missing'
12. set data['missingParameters'] = implode(',',$missingFields)
13. print json_encode(data,true)
14. else
15. create helperObj = new helper()
16. set data = POST
17. print json_encode(helperObj reference call POST['call'](data)
18. end
19. elseif set(POST['call'] AND POST['call'] == 'doctorsignupTimetable' then
20. create helperObj = new helper()
21. print json_encode(helperObj reference call POST['call'](POST))
22. else
23. print json_encode(array('status'=>false,'error'=>'Invalid method called'))
24. end

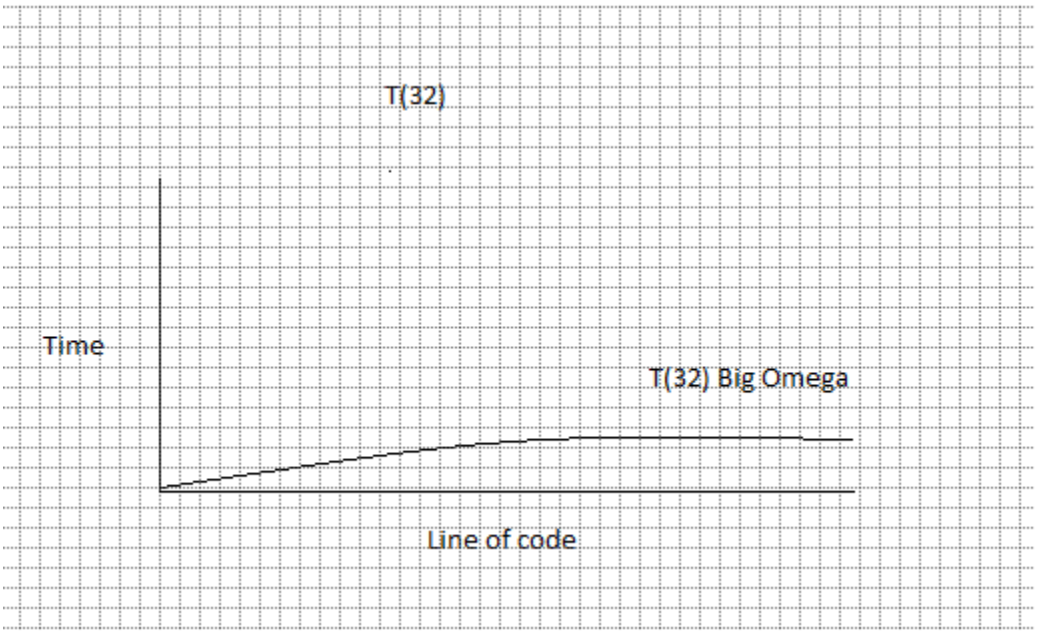
T(n+3)



Helper class filter(params) Method

1. If params then
2. Set whereClause = ""
3. If set params['speciality']) AND params['speciality'] not equal 0 then
4. If whereClause equals to " " then
5. whereClause concatenate = "where specialityid params['speciality']”
6. else
7. whereClause concatenate = " and specialityid = params['speciality']”
8. end
9. end
10. if (set(params['fees']) AND params['fees'] not equal 0 then
11. if whereClause not equal " "
12. whereClause concatenate = " and fees less then equals to $params['fees']”
13. else
14. whereClause concatenate = "where fees less than equals to params['fees']”
15. end
16. end
17. if (set params['Location'] AND params['Location'] not equals to 0 than
18. if whereClause not equal to empty than
19. whereClause concatenate = " and areaid = params['Location']”
20. else
21. whereClause concatenate ="where areaid = params['Location']”
22. end
23. end
24. set query = "select d.*,s.name as speciality,a.name as area from doctor d
25. inner join speciality s on d.specialityid = s.id
26. inner join area a on a.id = d.areaid concatenate whereClause
27. set result = current scope object call pdo call method customSelect(query)
28. return result
29. else
30. result = current scope object call pdo call method select('doctor',array('*'));
31. return result
32. end

T(32)



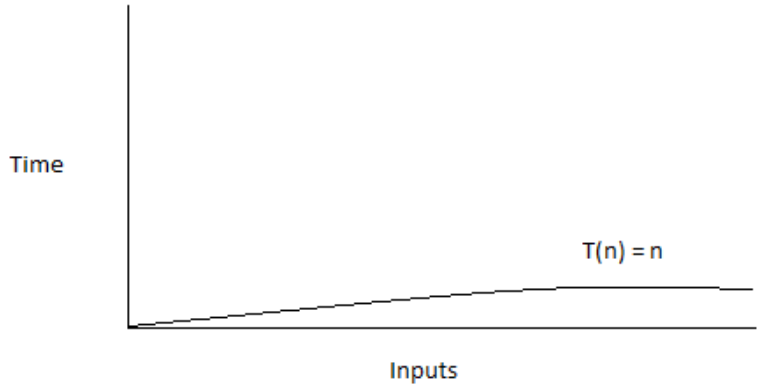
Helper class usersignup (params) Method

1. Set regex = '/^[_a-z0-9-]+(\.[_a-z0-9-]+)*@[a-z0-9-]+(\.[a-z0-9-]+)*(\.[a-z]{2,3})$/'
2. Declare missingFeild = array()
3. Set count = 0
4. Set message = ""
5. Repeat step 5 to 13 until params has key of all container in array contain some value
6. If params[key] not equal "" than
7. if key equal to "email" than
8. if not equal preg_match_all(regex , params[key]) than
9. set missingFeild[count] = "Email not valid"
10. count = count + 1
11. end
12. end
13. else
14. if key equal to "fname" than
15. set missingFeild[count] = "First Name";
16. count = count + 1
17. else if key equal to "lname"
18. set missingFeild[count] = "Last Name"
19. count = count + 1
20. else if key equal to "areaid")
21. set missingFeild[count] = "Area"
22. count = count + 1

23. else if key equal to "phone"
24. set missingFeild[count] = "Cell"
25. count = count + 1
26. else if key equal to "confirmpassword" than
27. set missingFeild[count] = "Confirm Password"
28. set count = count + 1
29. else if key equal to "nic"
30. set missingFeild[count] = "NIC"
31. count = count + 1
32. else
33. set missingFeild[count] = key;
34. count = count + 1
35. end
36. end
37. end
38. if empty(missingFeild) than
39. set response = current scope reference call method insert() and pass required attribute
40. return response
41. end
42. else
43. Repeat step 43 to 45 until i <= count
44. (i equal to 0) than message = "Error[] :- [".(i+1)."] ".missingFeild[i].", " : message.= "[".(i+1)."] ".missingFeild[i].", "
45. end
46. return message concatenate message = " Total error : concatenate field "count"
47. end

T(n) = 41+5n+n
T(n) = 41+6n
T(n) = 6n
T(n) = n

Linear time T(n) = O(n)



Response codes

RESPONSE CODE	RESPONSES
200 OK	7

Content size by content type

CONTENT TYPE	PERCENT	SIZE
Image	86.7 %	6.94 KB
HTML	13.3 %	1.06 KB
Total	100.00 %	8.00 KB

Requests by content type

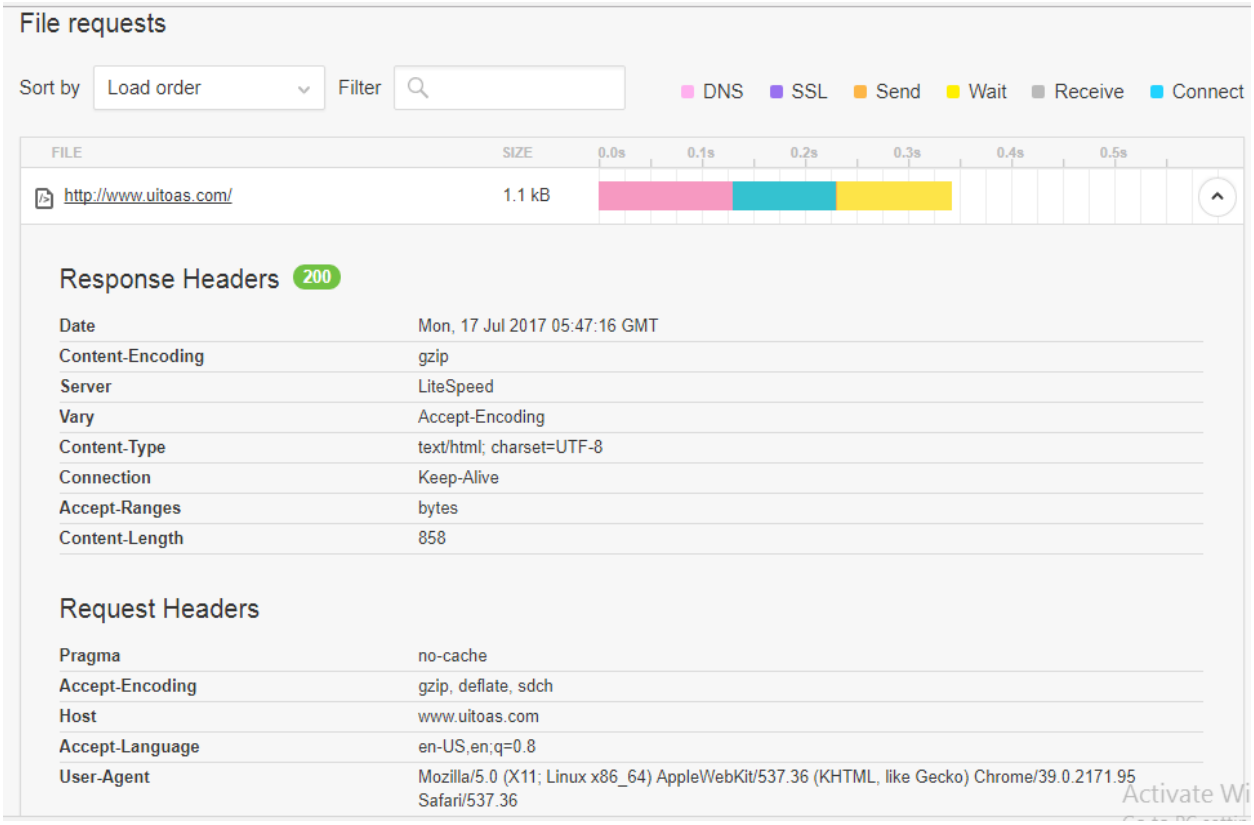
CONTENT TYPE	PERCENT	REQUESTS
Image	85.7 %	6
HTML	14.3 %	1
Total	100.00 %	7

Content size by domain

DOMAIN	PERCENT	SIZE
www.uitoas.com	100.0 %	8.00 KB
Total	100.00 %	8.00 KB

Requests by domain

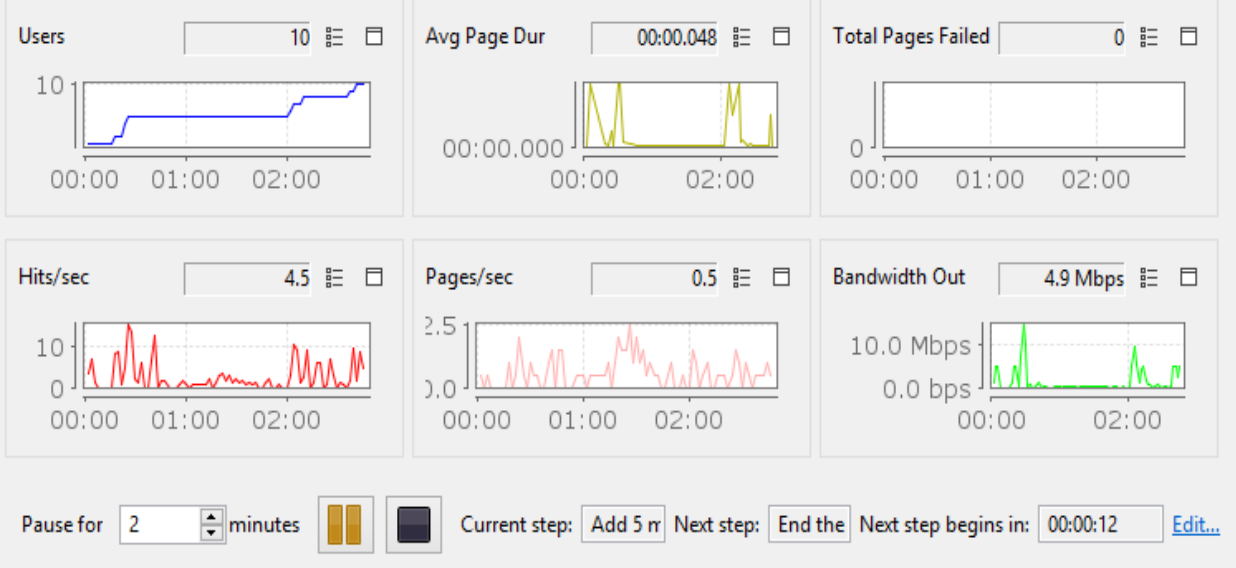
DOMAIN	PERCENT	REQUESTS
www.uitoas.com	100.0 %	7
Total	100.00 %	7


File requests
Sort by
Load order
Filter
DNS
SSL
Send
Wait
Receive
Connect
FILE
SIZE
0.0s
0.1s
0.2s
0.3s
0.4s
0.5s
http://www.uitoas.com/
1.1 kB
Response Headers
200
Date
Mon, 17 Jul 2017 05:47:16 GMT
Content-Encoding
gzip
Server
LiteSpeed
Vary
Accept-Encoding
Content-Type
text/html; charset=UTF-8
Connection
Keep-Alive
Accept-Ranges
bytes
Content-Length
858
Request Headers
Pragma
no-cache
Accept-Encoding
gzip, deflate, sdch
Host
www.uitoas.com
Accept-Language
en-US,en;q=0.8
User-Agent
Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/39.0.2171.95 Safari/537.36

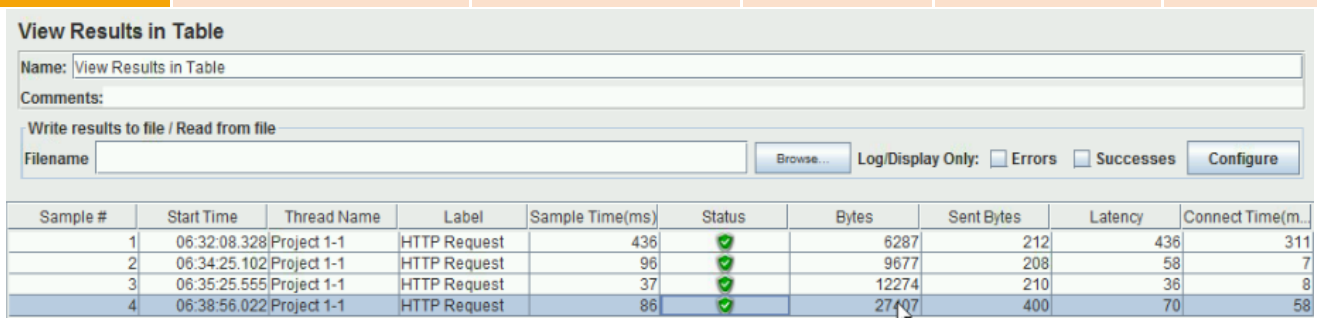
CHAPTER 7

Testing

Load Test



Page	Load time	Connect Time	Latency	Size in bytes	Sent bytes
Login	436	311	436	6287	212
Admin	37	11	45	6554	210
user-signup	96	7	58	9677	208
doctors-signup	37	8	36	12274	210
User-profile	70	58	70	21120	263
doctor-profile	14	1	14	6287	137



g

Response codes

RESPONSE CODE	RESPONSES
200 OK	7

Content size by content type

CONTENT TYPE	PERCENT	SIZE
Image	86.7 %	6.94 KB
HTML	13.3 %	1.06 KB
Total	100.00 %	8.00 KB

Requests by content type

CONTENT TYPE	PERCENT	REQUESTS
Image	85.7 %	6
HTML	14.3 %	1
Total	100.00 %	7

Content size by domain

DOMAIN	PERCENT	SIZE
www.uitoas.com	100.0 %	8.00 KB
Total	100.00 %	8.00 KB

Requests by domain

DOMAIN	PERCENT	REQUESTS
www.uitoas.com	100.0 %	7
Total	100.00 %	7

File requests

Sort by Load order Filter

DNS SSL Send Wait Receive Connect

FILE	SIZE	0.0s 0.1s 0.2s 0.3s 0.4s 0.5s
http://www.uitoas.com/	1.1 kB	

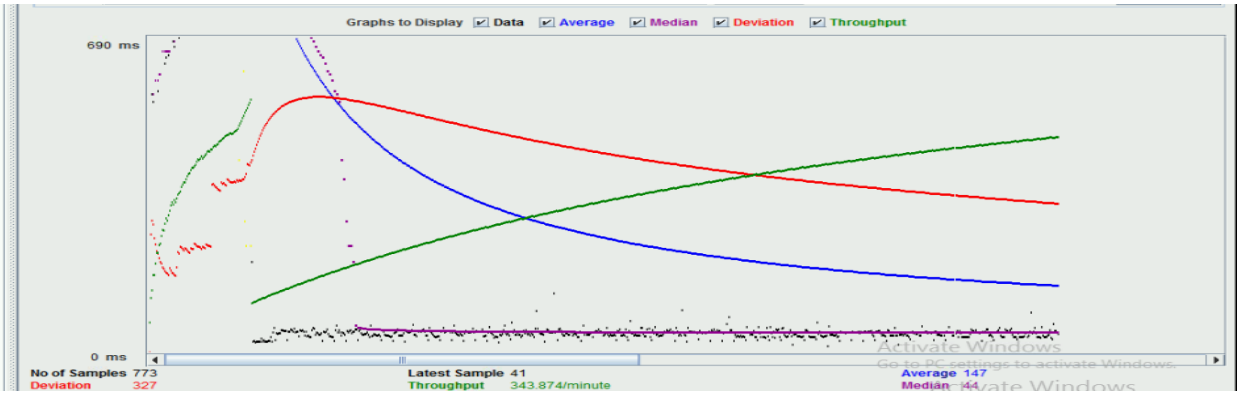
Response Headers 200

Date	Mon, 17 Jul 2017 05:47:16 GMT
Content-Encoding	gzip
Server	LiteSpeed
Vary	Accept-Encoding
Content-Type	text/html; charset=UTF-8
Connection	Keep-Alive
Accept-Ranges	bytes
Content-Length	858

Request Headers

Pragma	no-cache
Accept-Encoding	gzip, deflate, sdch
Host	www.uitoas.com
Accept-Language	en-US,en;q=0.8
User-Agent	Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/39.0.2171.95 Safari/537.36

Performance test



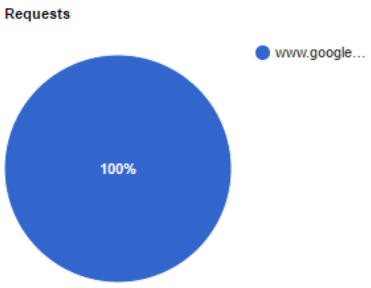
Request Details

Before Start Render	Before On Load	After On Load

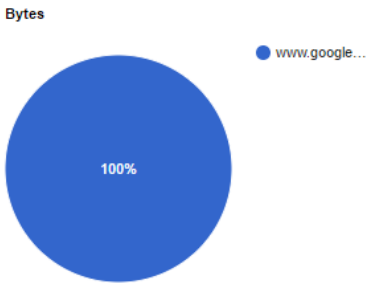
Request Details

Resource	Content Type	Request Start	DNS Lookup	Initial Connection	SSL Negotiation	Time to First Byte	Content Download	Bytes Downloaded	Certificates	Error/Status Code	IP
https://www.googleapis.com/rpc	application/json	0.314 s	40 ms	39 ms	202 ms	223 ms	-	0.8 KB	3231 B	200	172.217.3.106

Content breakdown by domain (First View)



Domain	Requests
www.googleapis.com	1



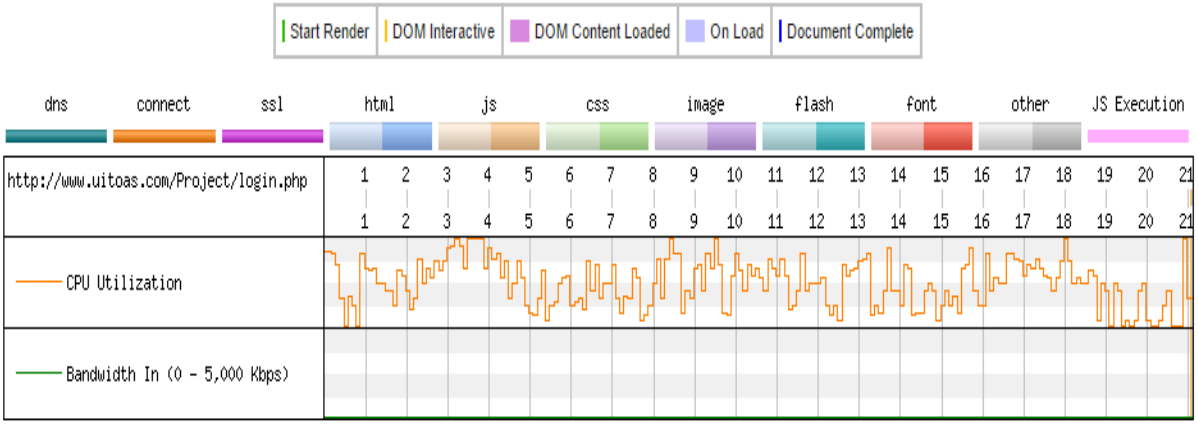
Domain	Bytes
www.googleapis.com	770

Stress test

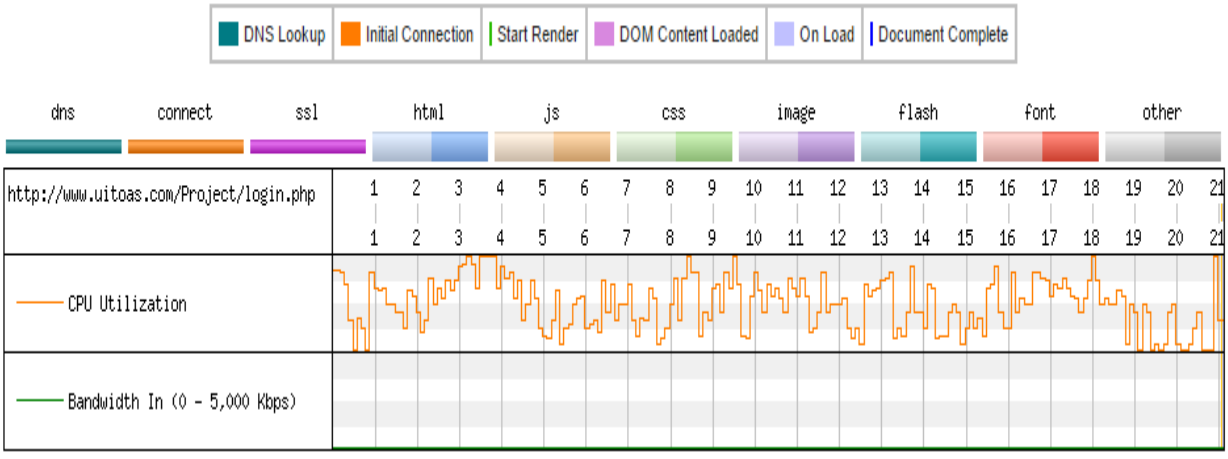
					Document Complete			Fully Loaded		
	Load Time	First Byte	Start Render	Result (error code)	Time	Requests	Bytes In	Time	Requests	Bytes In
First View (Run 1)	21.545s	-	-	12029	-	0	4 KB	21.545s	0	4 KB

Colordepth	Images	domInteractive	domContentLoaded	loadEvent
24	[]	21.060s	21.060s - 21.060s (0.000s)	21.065s - 21.065s (0.000s)

Waterfall View



Connection View



Load Test:

Project main page

Thread Name: Project 1-1
Sample Start: 2017-04-01 06:32:08 PDT
Load time: 436
Connect Time: 311
Latency: 436
Size in bytes: 6287
Sent bytes:212
Headers size in bytes: 421
Body size in bytes: 5866
Sample Count: 1
Error Count: 0
Data type ("text"|"bin"|""): text
Response code: 200
Response message: OK

user-signup page Test

Thread Name: Project 1-1
Sample Start: 2017-04-01 06:34:25 PDT
Load time: 96
Connect Time: 7
Latency: 58
Size in bytes: 9677
Sent bytes:208
Headers size in bytes: 232
Body size in bytes: 9445
Sample Count: 1
Error Count: 0
Data type ("text"|"bin"|""): text
Response code: 200
Response message: OK

doctors-signup page Test

Thread Name: Project 1-1
Sample Start: 2017-04-01 06:35:25 PDT
Load time: 37
Connect Time: 8
Latency: 36
Size in bytes: 12274
Sent bytes:210
Headers size in bytes: 232
Body size in bytes: 12042

Sample Count: 1
Error Count: 0
Data type ("text"|"bin"|""): text
Response code: 200
Response message: OK

User-profile page Test

Thread Name: Project 1-1
Sample Start: 2017-04-01 06:38:56 PDT
Load time: 70
Connect Time: 58
Latency: 70
Size in bytes: 21120
Sent bytes:263
Headers size in bytes: 451
Body size in bytes: 20669
Sample Count: 1
Error Count: 0
Data type ("text"|"bin"|""): text
Response code: 302
Response message: Found

main page redirect to user profile page

Thread Name: Project 1-1
Sample Start: 2017-04-01 06:38:56 PDT
Load time: 14
Connect Time: 1
Latency: 14
Size in bytes: 6287
Sent bytes:137
Headers size in bytes: 421
Body size in bytes: 5866
Sample Count: 1
Error Count: 0
Data type ("text"|"bin"|""): text
Response code: 200
Response message: OK

http://order.hostndomain.com/cart.php?a=view
https://www.youtube.com/watch?v=nZJv8FpR44A

Unit Test

I have performed the unit test of my system which is online appointment system. This test assures that every unit of my system is working properly as per requirement.

Alpha Test

I have done alpha testing of my system. From this alpha testing i found some small issues some of the issues are rectified and some will be rectified ASAP.

User Acceptance Test

Doctor and normal person are the user of my system and rite now my system is of small scope for university fyp so i have some test account for doctor and patient (normal user).form these test account we have performed user acceptance test.

ONLINE APPOINTMENT FORM				
Q1) Is the application helpful for medical purposes?				
1	2	3	4	5
Q2) Does the application have appropriate controls?				
1	2	3	4	5
Q3) does the patients have ease of access to book an appointment of doctor?				
1	2	3	4	5
Q4)Does the application is visually appealing ?				
1	2	3	4	5
Q5)How would you rate this application?				
1	2	3	4	5

	A	B	C	D	E
1	Timestamp	Username	Is this application actually helpful !	Is this application easy to use ?	Is there a strong link between the control and interface ?
2	2017/04/12 4:36:08 am	anasanwar@gmail.com	3	4	4
3	2017/04/12 4:37:17 am	kashanamir13@gmail.com	4	5	4
4	2017/04/12 4:38:22 am	amir1990@gmail.com	3	3	4
5	2017/04/12 4:39:12 am	ahmedraza675@gmail.com	4	4	3
6	2017/04/12 4:40:22 am	abdullahibrahim3@gmail.com	4	3	2
7	2017/04/12 4:41:17 am	sharukhmoin56@gmail.com	4	3	4
8	2017/04/12 4:42:37 am	waleedhussain14@yahoo.com	3	3	4
9	2017/04/12 4:43:23 am	faizanbaig@gmail.com	4	4	5
10	2017/04/12 4:44:54 am	asadibrahim@gmail.com	4	4	4
11	2017/04/12 4:45:44 am	samamauddin47@gmail.com	5	4	5
12	2017/04/12 4:47:34 am	Lailaamir@gmail.com	4	4	4
13	2017/04/12 4:48:21 am	munibazuberi@gmail.com	4	4	5
14	2017/04/12 4:48:52 am	hinabaig@gmail.com	4	5	4
15	2017/04/12 4:49:35 am	wasiqperzada@gmail.com	4	5	4
16	2017/04/12 4:50:58 am	kamranbaig@gmail.com	5	5	4
17	2017/04/12 4:52:07 am	waseembadami@gmail.com	5	4	4
18	2017/04/12 4:53:05 am	sameergul@gmail.com	4	5	4
19	2017/04/12 4:57:40 am	hamzazuberi@gmail.com	4	5	4
20	2017/04/12 5:03:51 am	shoaibkhan@gmail.com	4	5	5
21					

CHAPTER 8

Conclusion

As per requirement of completion of the project "The Online Appointment System" in a year, we have complete the core functionalities of the project and achieve the main goal of the project. Some functionalities required some modification but the main functionalities which are also the main goals of the project are properly working and we have done the testing after the deployment of the project on server.

CHAPTER 9

Future Enhancement

CHAPTER 10

References

Aga khan university hospital

https://hospitals.aku.edu/pakistan/Pages/Request-an-Appointment.aspx

Online Appointment Doctor Lahore

http://doctorshospital.com.pk

Al-Shifa Trust Islamabad

http://www.alshifaeye.org/onlineappointment.php

Online Appointment in pakistan

http://findadoctor.com.pk/

CHAPTER 11

PLAGIARISM REPORT

Plagiarism Scan Report

Summary	
Report Genrated Date	18 Jul, 2017
Plagiarism Status	**100% Unique**
Total Words	639
Total Characters	3837
Any Ignore Url Used	

Content Checked For Plagiarism:

CHAPTER 1

INTRODUCTION

Introduction
This online appointment system deal with a great deal o acilities with its existence. We are in an era o digital technology or i I say today a man is incomplete without accessing the application o technology in his/her li e. Keeping the technology actor in mind, we are creating one online application, which can be bene icial or large group o people o di ferent work nature. Especially the category o work people who requires prior appointment be ore meeting with the person or Instance like doctor by using our online system one can get the online appointment sitting in any part o the country by just ollowing the simple process.

Problem De inition

Plagiarism Scan Report

Summary	
Report Genrated Date	18 Jul, 2017
Plagiarism Status	**100% Unique**
Total Words	392
Total Characters	2507
Any Ignore Url Used	

Content Checked For Plagiarism:

CHAPTER 2

LITERATURE REVIEWS

An online appointment system which will book prior appointment with the doctor.
The user will not need to visit the hospital to take the appointment. And Doctor can track user by viewing records which are mentioned by our system.

Existing system
Our system is not match any o the existing system. As there are many system or website which are now existing in our country to make online appointment.
E.g. Findadoctor.com.pk, onlinedoctor.com etc.

Plagiarism Scan Report

Summary	
Report Genrated Date	18 Jul, 2017
Plagiarism Status	**83% Unique**
Total Words	391
Total Characters	2570
Any Ignore Url Used	

Content Checked For Plagiarism:

CHAPTER 3

SOFTWARE AND HARDWARE REQUIREMENTS

Hardware Required
Operating System (OS) Hardware

• Microso t Windows XP/ 7 (32 or 64 Bit) • 1.5 GHz 32-bit (x86)/64-bit (x64) or higher
• 512 MB RAM (32-bit) or higher
• 2GB HDD ree
• Internet modem
Table 2.1 Hardware Requirement
Internet Modem
This is required to use the internet.
Operating System
Minimum Requirement is WIN 7 & 8 b/c it is a web application so it required a high speed Browser like GOOGLE CHROME AND FIREFOX new versions these version only required to install in this operating systems.
Operating Environment
This product will be developed using Open Source server side language PHP so, we would pre erably use Microso t Windows Operating System or developing this so tware.
Hardware Inter ace
• Display Screen

Plagiarism Scan Report

Summary	
Report Genrated Date	18 Jul, 2017
Plagiarism Status	**100% Unique**
Total Words	400
Total Characters	2543
Any Ignore Url Used	

Content Checked For Plagiarism:

CHAPTER 4

Methodology

Be ore starting the project we made the complete analysis or the people who will use the system and will be bene ited by using this acility, a ter this we made the paper based prototyping and sequentially integrate the pages.

As now the prototype is completed, we igured out the inputs and possible outputs. We also marked those inputs which are required (Mandatory) by the system in order to complete the transactions like or example NIC number, mobile number are required or payment and tracking the User or any sort o query.

As the Doctor is the main business user o this system. There ore, a ter the analysis process we have created the complete mechanism or the veri ication and validation o

Plagiarism Scan Report

Summary	
Report Genrated Date	18 Jul, 2017
Plagiarism Status	100% Unique
Total Words	260
Total Characters	1744
Any Ignore Url Used	

Content Checked For Plagiarism:

CHAPTER 5

UML Diagrams

Use Case

CHAPTER 12

ANNEXURE

CHAPTER 12.1

PROPOSAL

CHAPTER 12.2

GANTT CHART

CHAPTER 12.3

PROJECT CHARTER

CHAPTER 12.4

PROJECT SRS