

Role Based & Time Bound Access Control

A
MINOR PROJECT REPORT

Submitted for the partial fulfillment of the
BACHELOR OF TECHNOLOGY
IN
COMPUTER SCIENCE AND ENGINEERING

Submitted by-

Abhishek Kumar Gupta (9914103169)

Deepanshu Kathuria (9914103177)



**Under the Guidance
Of**

Mrs. Vartika Srivastava

(Dept. of CSE)

Department of Computer Science and Engineering
Jaypee Institute of Information Technology, Noida

DECEMBER 2016

Acknowledgement

We are overwhelmed in all humbleness and gratefulness to acknowledge our depth to all those who have helped us to put our ideas and assigned work, well above the level of simplicity and into something concrete.

We all thank whole heartedly **Mrs. Vartika Srivastava, Department of Computer Science and Engineering, Jaypee Institute of Information Technology, Noida, India** for being the mentor of this valuable project, constantly motivating us for doing better and showing complete confidence in our work. We are very thankful for her valuable suggestions, guidance and encouragement.

In the process, we learnt a lot of technical as well as non-technical things through this project.

We would like to place on record our deep sense of gratitude to **Mrs. Vartika Srivastava, (Department Of CSE), Jaypee Institute of Information Technology, India** for her generous guidance, help and useful suggestions.

Finally, we also like to thank all other colleagues working in different projects for helping us at small problems as well as critical junctures.

Signature(s) of Students

- 1) Abhishek Kumar Gupta (9914103169)
- 2) Deepanshu Kathuria (9914103177)

.

ABSTRACT.

Security and privacy are widely recognized as important requirements for access and management of data. Data needs to be managed with customizable access control in both spatial and temporal dimensions.

The main objective of our project ‘Role Based and Time Bound Access Control’ is to provide the authorities to the registered users and to control the access of sensitive data. RBTBAC model provides the flexibility in both roles and time dimensions to control the access of sensitive data. RBTBAC model has two salient features. First is to develop a privacy aware access and management of data, focusing on the consistency of access authorization with the activation role of the user. Second, we have implemented time-bound access authorization and control.

Within this document, readers can find information about our general project concepts, a list of functional and non-functional requirements.

Table of Contents

No.	Topics	Page No.
1	List of All Figures	1
2	Abbreviation and Definitions	2
3	Introduction	3
3.1	Product Scope	3
3.2	Product Prospective	3
3.3	Operating Environment	4
4	Background Study	4
5	Research Paper Summary	8
6	Specific Requirements	10
6.1	User Interfaces	10
6.2	Hardware Interfaces	10
6.3	Software Interfaces	11
6.4	Communication Interfaces	12
7	Functional Requirements	12
8	Behaviour Requirements	14
9	Non-Functional Requirements	20
9.1	Performance Requirements	20
9.2	Safety and Security Requirements	20
9.3	Software Quality Requirement	20
10	Testing Report	22
11	Conclusion and Future Scope	34
12	References	35

LIST OF ALL FIGURES

Figures	Description	Page no.
1	User Interface	8
2	User Interface	8
3	Use case	14
4	Schema	17
5	Class Diagram	18
6	Sequence Diagram of Appointment	19
7	Super-admin Dashboard	28
9	Change Role	28
10	Admin Dashboard	29
11	Add Counsellor	29
12	Counsellor Dashboard	30
13	Doctor Details	30
14	Patient Dashboard	31
15	Pie Chart	33
16	Gantt Chart	34

ABBREVIATIONS AND DEFINITIONS

- SRS – Software Requirement Specification.
- Front End – This stands for the interface that the user will see while using the application
- Client – This stands for all the end users using the application.
- Administrator – An individual responsible for managing user accounts, contents, and security of a website, database, or other system.
- Notes – Text added to content and stored in the database; will be private (viewable only by administrators or moderators)
- CLI: Command Line Interface
- API: Application Programming Interface
- ROM: Read-Only Memory
- HTML: Hypertext Mark-up Language
- RBTBAC: Role Based and Time Bound Access Control
- DAC: Discretionary Access Control
- MAC: Mandatory Access Control
- CSS: Cascading Style Sheets

INTRODUCTION

Role Based & Time Bound Access Control is a Database Security project. We are implementing this project on a website in which there are five roles-Doctor, Patient, Super Admin, Admin and Counsellor. And each of them have the different authorities.

Users can only access data when they have authorisation for that. Every role has different privileges and so, according to a user's role he/she can access the data for the privileges defined for that role.

Admin can change the password of authorized user, add or drop the user, Counsellor has the authority to assign time to patient for the consult from doctor. Super Admin has all the authorities as admin's authorities and in addition to those authorities, he can also change role of users.

Patient and Doctor have the authority to register them. After that, they can access their data. And doctor will be able to see all the required information of the patient assigned by the counsellor on his dashboard. Doctor will treat the patient and after successfully treatment, doctor will be having the record of the patient in treatment history.

PRODUCT SCOPE: The system will be partitioned as 2 definitive modules:

1. User Module- In this module user has to login to access the website content using any browser.
2. Administrator Module- Administrator can change the role of the user according to the privileges that should be given to user.

The Intended Audience of the project 'Role Based & Time Bound Access Control' is the counsellor, patient and doctor who want to register on the website and after being verified by the admin, they can view their details.

PRODUCT PROSPECTIVE: The proposed system will allow the access of the data to the authorized user. Authorized users shall be able to log on to view or modify the data. Role Based & Time Bound Access Control is meant to be one of the most flexible, elegant, customizable, simple yet powerful website content management and security system.

There will be some other facilities provided on the website like login, view details and register.

OPERATING ENVIRONMENT:

- Adobe Dreamweaver
- XAMPP Server
- Operating System (Windows, Linux, Unix)
- PHP Designer
- Web Browser
- Laptop or Desktop
- Sublime Text

BACKGROUND STUDY

This section includes the basic knowledge and background study, one requires to make this project. Following are the studies and researches which inspired us to make 'Role Based & Time Bound Access Control'.

HTML

HTML is a Mark-up language of describing web pages. HTML stands for Hyper Text Mark-up Language. Along with Bootstrap and JavaScript, HTML is a cornerstone technology, used by the most websites to create visually engaging webpages, user interfaces for web applications and user interfaces for many mobile applications. Web browsers can read HTML files and render them into visible or audible web pages. HTML describes the structure of a website semantically along with cues for presentation, making it a mark-up language, rather than a programming language. HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items.

A mark-up language is a set of mark-up tags. HTML documents are described by HTML tags. Each HTML tag describes different document content:

- The **DOCTYPE** declaration defines the document type to be HTML
- The text between **<html>** and **</html>** describes an HTML document
- The text between **<head>** and **</head>** provides information about the document
- The text between **<title>** and **</title>** provides a title for the document
- The text between **<body>** and **</body>** describes the visible page content
- The text between **<h1>** and **</h1>** describes a heading
- The text between **<p>** and **</p>** describes a paragraph

JavaScript

JavaScript is a high-level, dynamic and interpreted programming language. It has been standardized in the ECMA Script language specification. Alongside HTML and CSS, it is one of the three essential technologies of World Wide Web content production; most websites employ it and it is supported by all modern web browsers without plug-ins. JavaScript is prototype-based with first-class functions, making it a multi-paradigm language, supporting object-oriented, imperative, and functional programming styles. It has an API for working with text, arrays, dates and regular expressions, but does not include any I/O, such as networking, storage or graphics facilities, relying for these upon the host environment in which it is embedded.

Despite some naming, syntactic, and standard library similarities, JavaScript and Java are otherwise unrelated and have very different semantics. The syntax of JavaScript is derived from C, while the semantics and design are influenced by the self and Scheme programming languages.

JavaScript is also used in environments that are not web-based, such as PDF documents, site-specific browsers, and desktop widgets. Newer and faster JavaScript virtual machines (VMs) and platforms built upon them have also increased the popularity of JavaScript for server-side web applications. On the client side. JavaScript has been traditionally implemented as an interpreted language, but more recent browsers perform

just-in-time compilation. It is also used in game programming with run time environments such as Node.js.

PHP

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. Originally created by Rasmus Lerdorf in 1994, the PHP reference implementation is now produced by The PHP Group. While PHP originally stood for Personal Home Page, it now stands for the recursive backronym PHP: Hypertext Pre-processor.

PHP code may be embedded into HTML code, or it can be used in combination with various Web template systems and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications.

The standard PHP interpreter, powered by the Zend Engine, is free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers on almost every operating system and platform, free of charge.

The PHP language evolved without a written formal specification or standard until 2014, leaving the canonical PHP interpreter as *de facto* standard. Since 2014 work has been ongoing to create a formal PHP specification.

jQuery

jQuery is a cross-platform JavaScript library designed to simplify the client-side scripting of HTML. jQuery is the most popular JavaScript library in use today, with installation on 65% of the top 10 million highest-trafficked sites on the Web. jQuery is free, open-source software licensed under the MIT License.

jQuery's syntax is designed to make it easier to navigate a document, select DOM elements, create animations, handle events, and develop Ajax applications. jQuery also provides capabilities for developers to create plug-ins on top of the JavaScript library.

This enables developers to create abstractions for low-level interaction and animation, advanced effects and high-level, themeable widgets. The modular approach to the jQuery library allows the creation of powerful dynamic web pages and web applications.

The set of jQuery core features—DOM element selections, traversal and manipulation—enabled by its selector engine (named "Sizzle" from v1.3), created a new "programming style", fusing algorithms and DOM data structures. This style influenced the architecture of other JavaScript frameworks like YUI v3 and Dojo, later stimulating the creation of the standard Selectors API.

Microsoft and Nokia bundle jQuery on their platforms. Microsoft includes it with Visual Studio for use within Microsoft's ASP.NET AJAX framework and ASP.NET MVC Framework while Nokia has integrated it into the Web Run-Time widget development platform. jQuery has also been used in Media Wiki since version 1.16.

Bootstrap

Bootstrap is a free and open source front-end web framework for designing websites and web applications. It contains HTML and CSS based design template for typography, forms, buttons, navigation and other interface components, as well as optional JavaScript extensions. Unlike many web frameworks, it concerns itself with front-end development only. Bootstrap is a framework, because it provides structure instead of simply being a library of predefined elements and styles.

Bootstrap, originally named Twitter Blueprint was developed by Mark Otto and Jacob Thornton at Twitter as a framework to encourage consistency across internal tools. Before Bootstrap, various libraries were used for interface development, which led to inconsistencies and a high maintenance burden. It provides a set of stylesheets that provide basic style definitions for all key HTML components. These provide a uniform, modern appearance for formatting text, tables and form elements.

Bootstrap contains other commonly used interface elements. The components are implemented as CSS classes, which must be applied to certain HTML elements in a page.

Bootstrap comes with several JavaScript components in the form of jQuery plugins. They provide additional user interface elements such as dialog boxes, tooltips and carousels. They also extend the functionality of some existing interface elements, including for example an auto-complete function for input fields. In version 2.0, the following JavaScript plugins are supported: Model, Dropdown, Scroll spy, Tab, Tooltip, Popover, Alert, Button, Collapse, Carousel and Typeahead.

DATABASE MANAGEMENT SYSTEM

A Database Management System (DBMS) is a computer software application that interacts with the user, other applications and the database itself to capture and analyse data. A general-purpose DBMS is designed to allow the definition, creation, querying, update and administration on databases.

A DBMS makes it possible for end users to create, read, update and delete data in a database. The DBMS essentially serves as an interface between the database and end users or application programs, ensuring that data is consistently organized and remains easily accessible.

The DBMS is perhaps most useful for providing a centralized view of data that can be accessed by multiple users, from multiple locations, in a controlled manner. A DBMS can limit what data the end user sees, as well as how that end user can view the data, providing many views of a single database schema. End users and software programs are free from having to understand where the data is physically located or on what type of storage media it resides because the DBMS handles all requests.

PostgreSQL

PostgreSQL is an object relational database (ORDBMS). It is developed by the PostgreSQL Global Development Group, adverse group of many companies and individual contributors.

PostgreSQL supports transactions, subselects, triggers, views, foreign key referential integrity, and sophisticated locking. It runs on numerous platforms including Linux, most flavors of Unix, Mac OS X, Solaris, Tru64 and Windows. It supports text, images, sounds, and video, and includes programming interfaces for C/C++, JAVA, Perl, Python, Ruby, Open Database Connectivity (ODBC).

Research Papers: -

1. Role Based and Time Bound Access and Management of EHR Data.

Published On: 21 June 2013

Authors: Rui Zhang, Ling Liu, Rui Xue.

Summary: - Role Based Access Control (RBAC) is designed to simplify security administration by introducing the 'role' abstraction between principles (subjects) and privileges (objects).

Three phases of RBTBAC model are: -

- Role-Based Management
- Time-Bound Management
- Path -Invisible Access Control

Role based key structure is applied here.

- Encryption and Decryption Key Structure: -To encrypt and decrypt data, system uses key tree.
- Different keys for different patient.
- To avoid inconvenience, RBTBAC uses a master key.

For Time-bound management RBTBAC model uses Time Tree Method.

- Time Binary Tree: - Binary Tree index has two or less than two nodes for time.
- Time Multi Tree: -Multi Tree index has more than two nodes.
- Space Complexity: - It has effective space complexity.

2. Published in SEA '07 Proceedings of the 11th IASTED International Conference on Software Engineering and Applications, 11 June 2007

Summary: -This paper describes testing the applicability of Role Based Access Control(RBAC) within an existing medical database. We show how role hierarchies and RBAC rules are derived for this database and observe the outcomes of our RBAC implementation.

The traditional methods of DB access control are Mandatory Access Control(MAC) and

Discretionary Access Control(DAC).MAC has mainly been used for military applications and has been found to be too rigid for commercial use. DAC has been seen to be too weak.

RBAC is different to DAC and MAC in that access is based on job/task that a DB user is required to carry out rather than data ownership. MAC works by giving users security labels which ensures that the users only get access to objects with the same or lower security labels, while DAC allows users to assign or revoke specific privileges to or from other users.

RBAC controls access to information based on users' work activities. It is easy to change users' privileges without modifying the underlying access structure by simply adding or removing people from roles.

The role hierarchy created for RBAC can directly reflect an organization's structure and can evolve over time as the organization changes. In addition, roles can be composed of other roles using a role hierarchy, which prevents repetition of role-privilege assignments. In a typical role hierarchy used in RBAC "senior roles" inherit the (positive) privileges assigned to "junior" roles (but not conversely). Three types of digital certificates are used: identity certificates for authentication; attribute certificates for authorization; and access-rule certificates for propagation of an access control policy.

SPECIFIC REQUIREMENTS

This section covers the requirements of the project with all the interfaces, functional and non-functional requirements of the project with diagrams and specifications of the software and languages used.

USER INTERFACES

Once the program has been launched, the user will interact primarily with the webpage. Since the program will be designed with simpler mechanics and ease of use in mind, the GUI is not overly complex. Whatever interface is used is in fact kept readable and minimalist in-order to accommodate all the type of screens.

Figure1

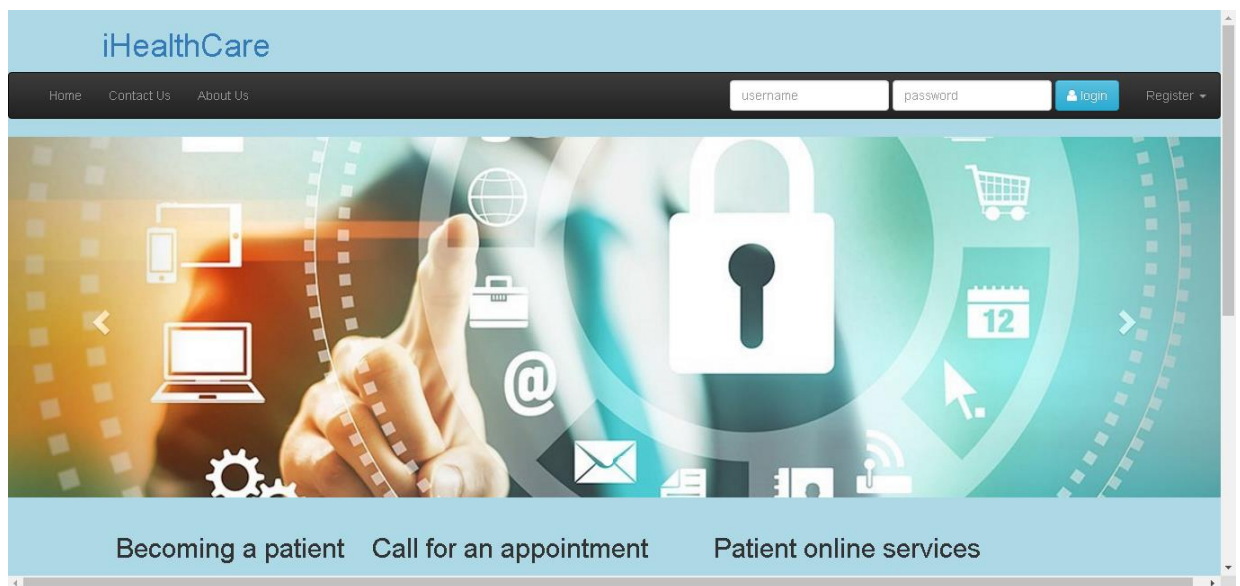


Figure2

The image shows a "Contact Us" form on a light blue background. The title "Contact Us" is centered at the top. Below the title are three input fields: "Your Name", "Your Email", and "Subject". Below these is a large text area for the message, with the placeholder text "Message for me *". At the bottom right of the form is a green "Send message" button.

HARDWARE INTERFACES

The users need to have access to a keyboard, monitor, and a computer to view and manage our webpage. Users are highly recommended to have a mouse, and keyboard for full functionality.

SOFTWARE INTERFACES

The users need to have a functional operating system that includes an up to date Internet browser with JavaScript enabled, such as Internet Explorer, Google Chrome, Firefox, Opera or Safari. In addition to a functional operating system, and Internet browser.

COMMUNICATION INTERFACES

- WAMP Server
- Web Browser

FUNCTIONAL REQUIREMENTS

1. System shall allow the administrator login

A possible threat exists if the system does not recognize a proper administrator login. To prevent this risk extensive testing with multiple user logins should find any problems.

2. System shall allow the administrator log out

A possible threat exists if the system does not recognize a proper log out and keeps administrator logged in, possibly giving someone unauthorized access. To prevent this risk extensive testing shall be done to insure there are no problems.

3. System shall allow the administrator to add Counsellor

1. Description

After a successful login, an administrator shall be able to add Counsellor.

2. Technical issues

Duplicate user id occurs.

3. Risks

A possible risk is a user is already added. The system shall notify the administrator for the risks of adding a user.

4. System shall allow users to view the details as per their authority.

5. System shall allow super administrator to change the role

1. Description

After successful login, Admin shall be able to change the role of the registered users.

2. Technical issues

Link is broken upon or after changing the role.

3. Risks

Server fails to change the roles.

7. System shall allow administrator to change user's password

1. Description

After a successful login, the administrator shall be able to change the password of the registered users to access the system.

2. Technical issues

Password is too long/short.

3. Risks

User's identity did not match.

8. Doctor should be able to access patient information for a specific time

1. Description

After successful login, doctor shall be able to view appointment details before appointment ends.

2. Technical issues

Page may not reload.

2. Risks

Doctor will be able to view appointment for more than specified time.

Behaviour Requirements

USE CASE DIAGRAM

Figure 3(a)

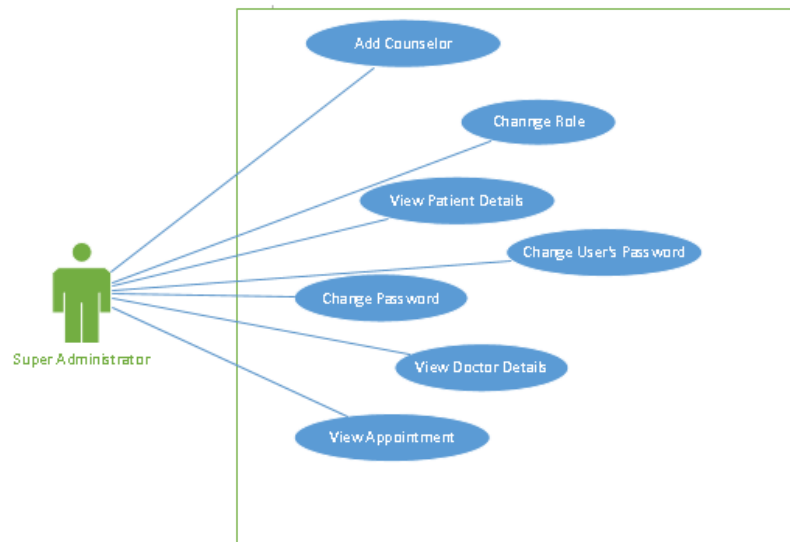


Figure 3(b)

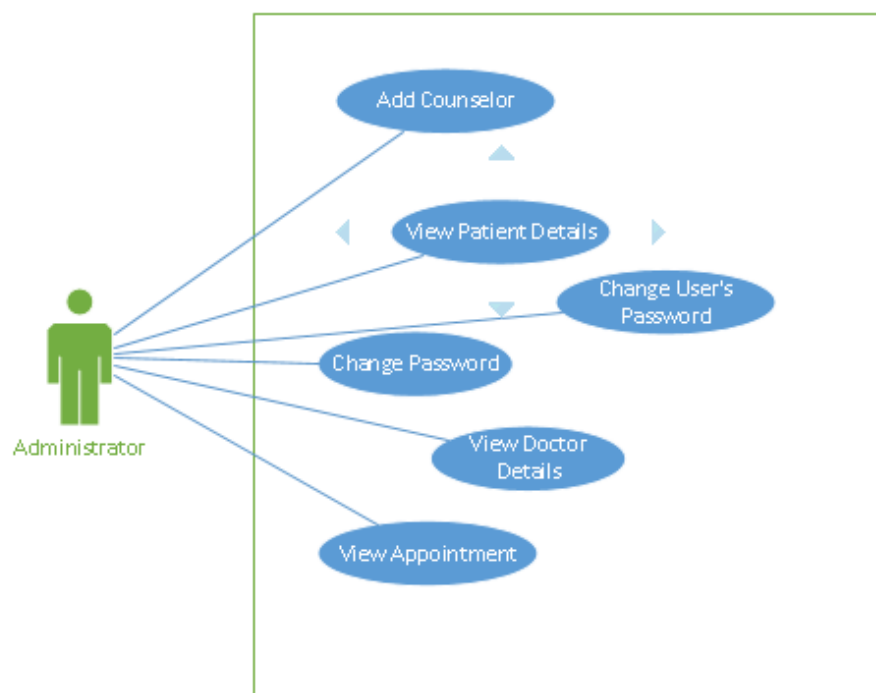


Figure 3(c)

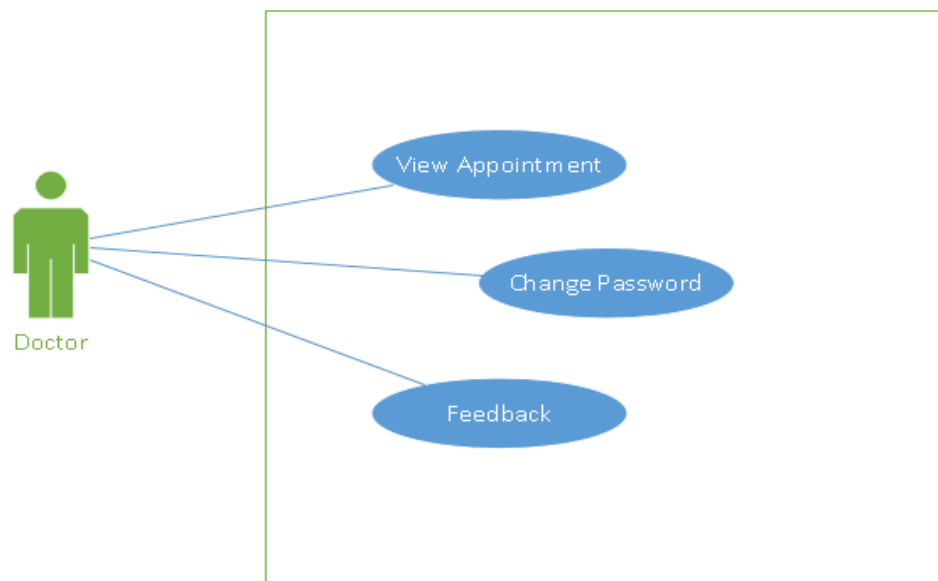


Figure 3(d)

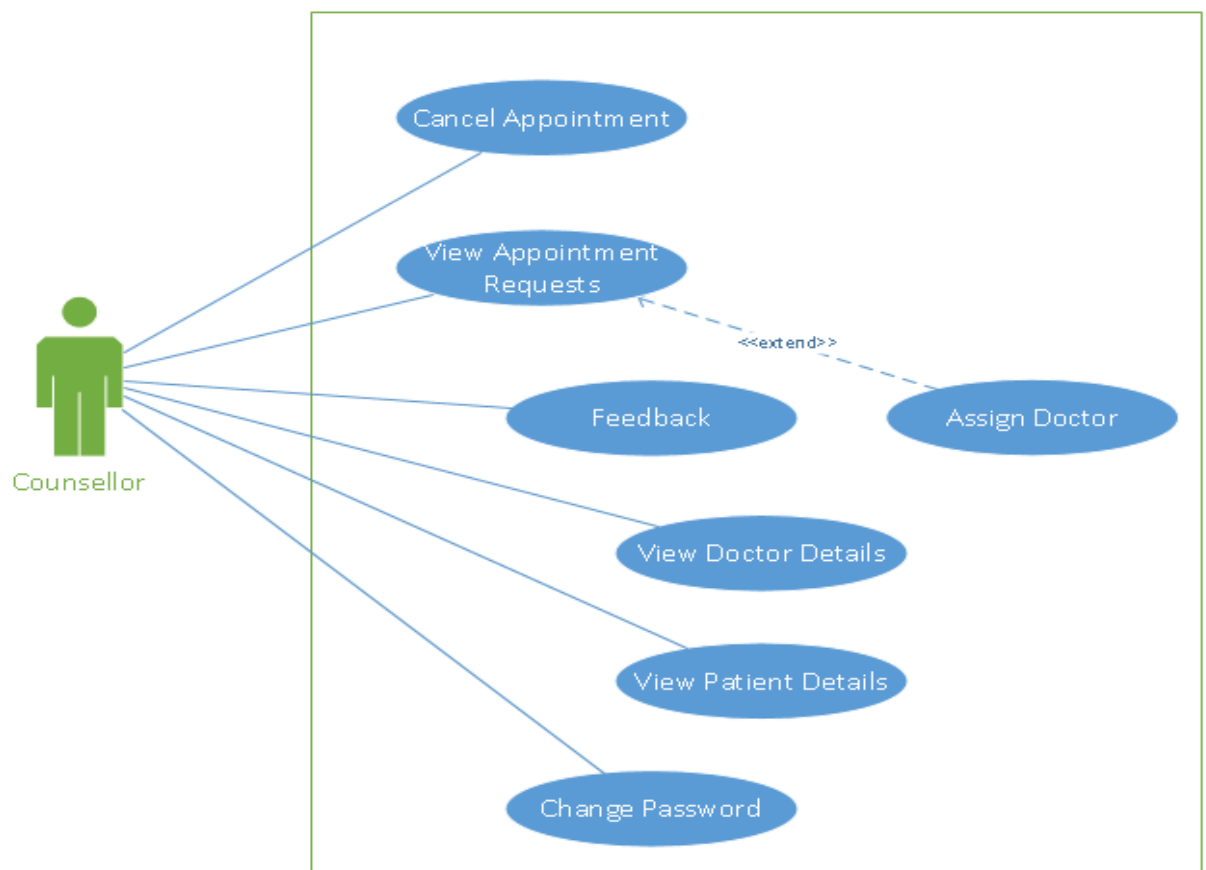
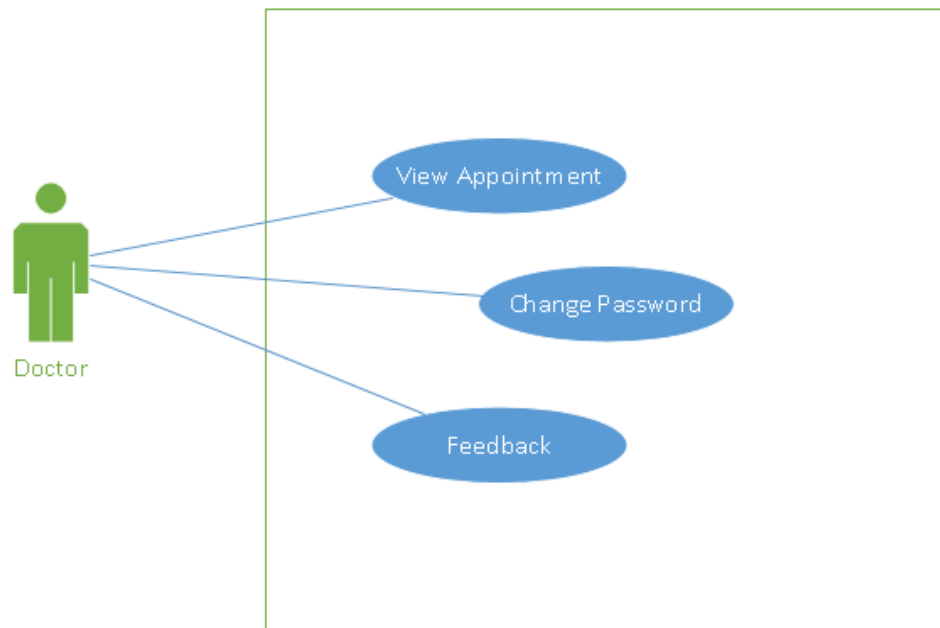
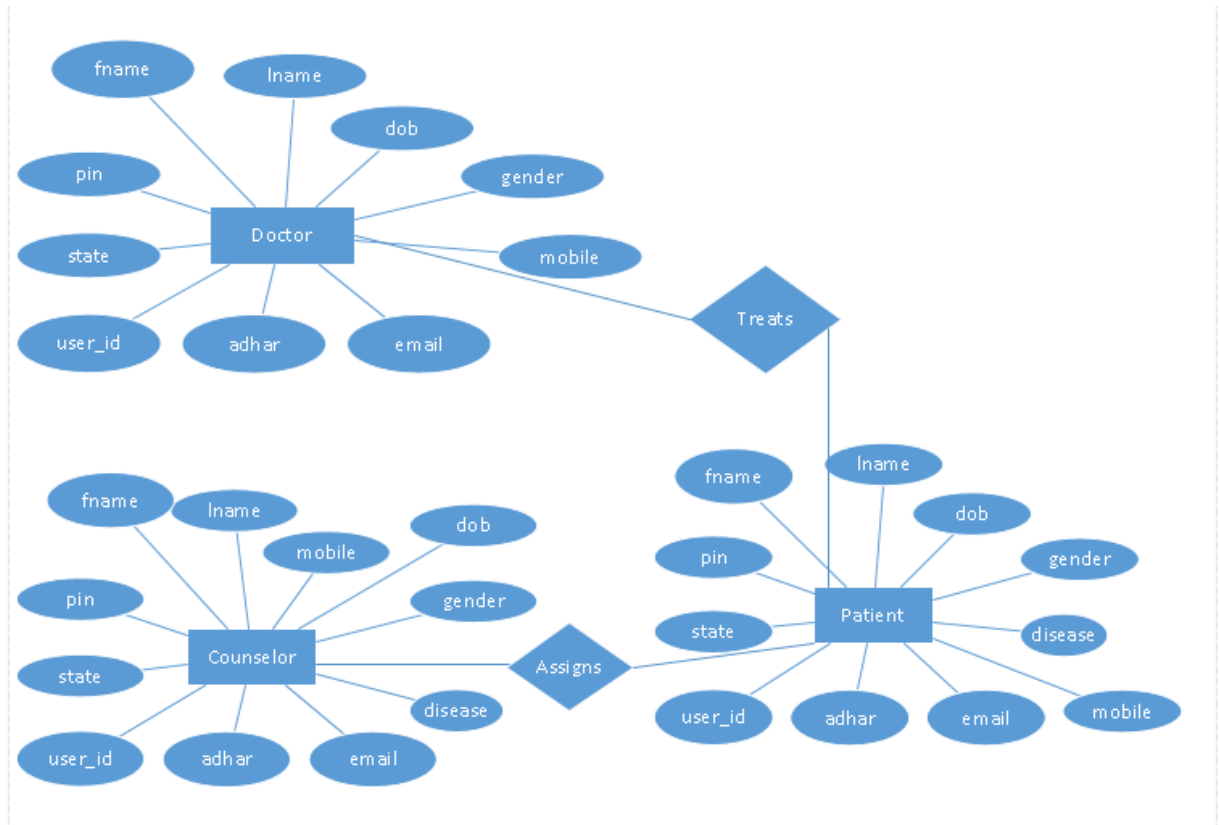


Figure 3(e)



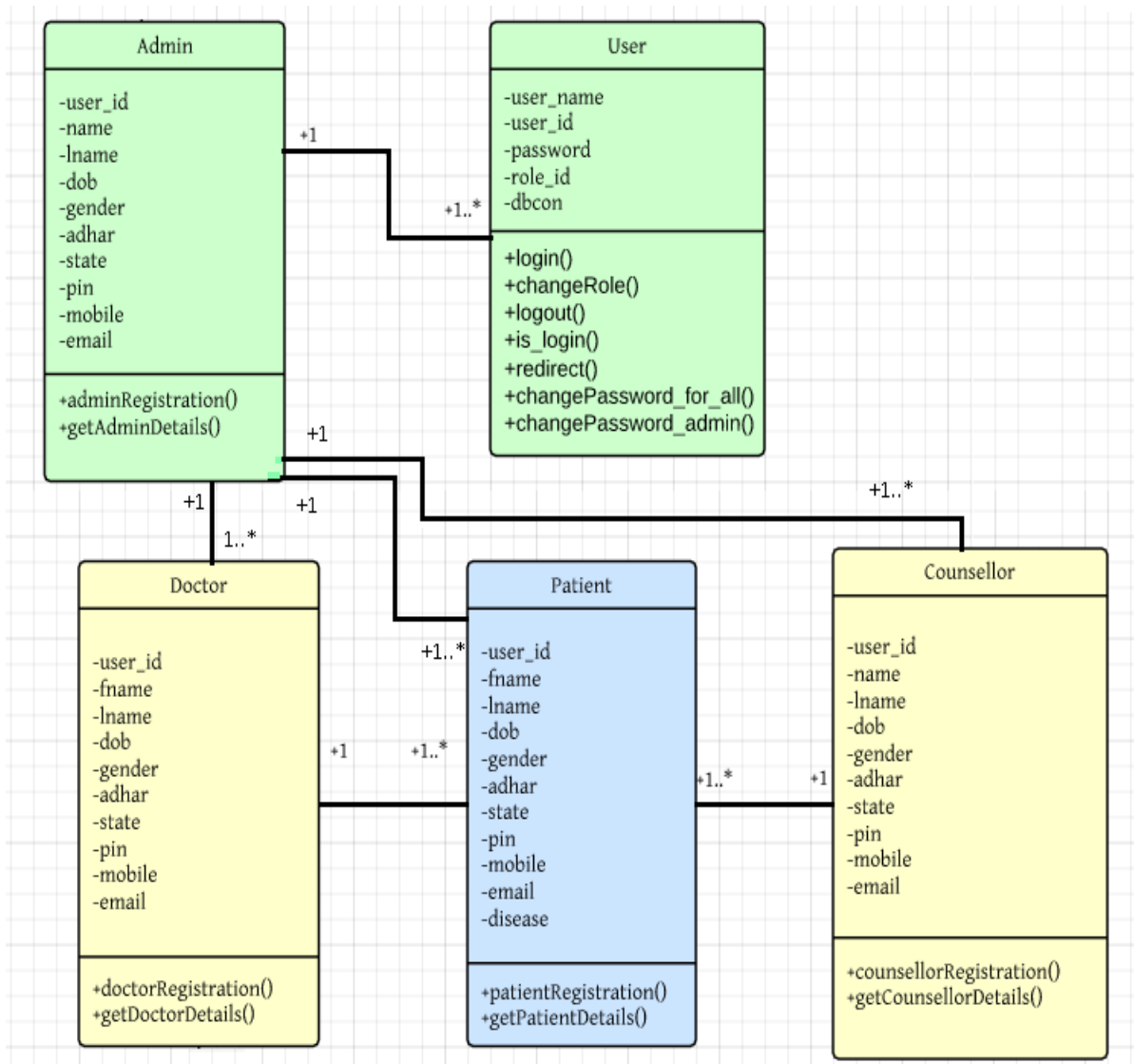
Schema Diagram

Figure 4



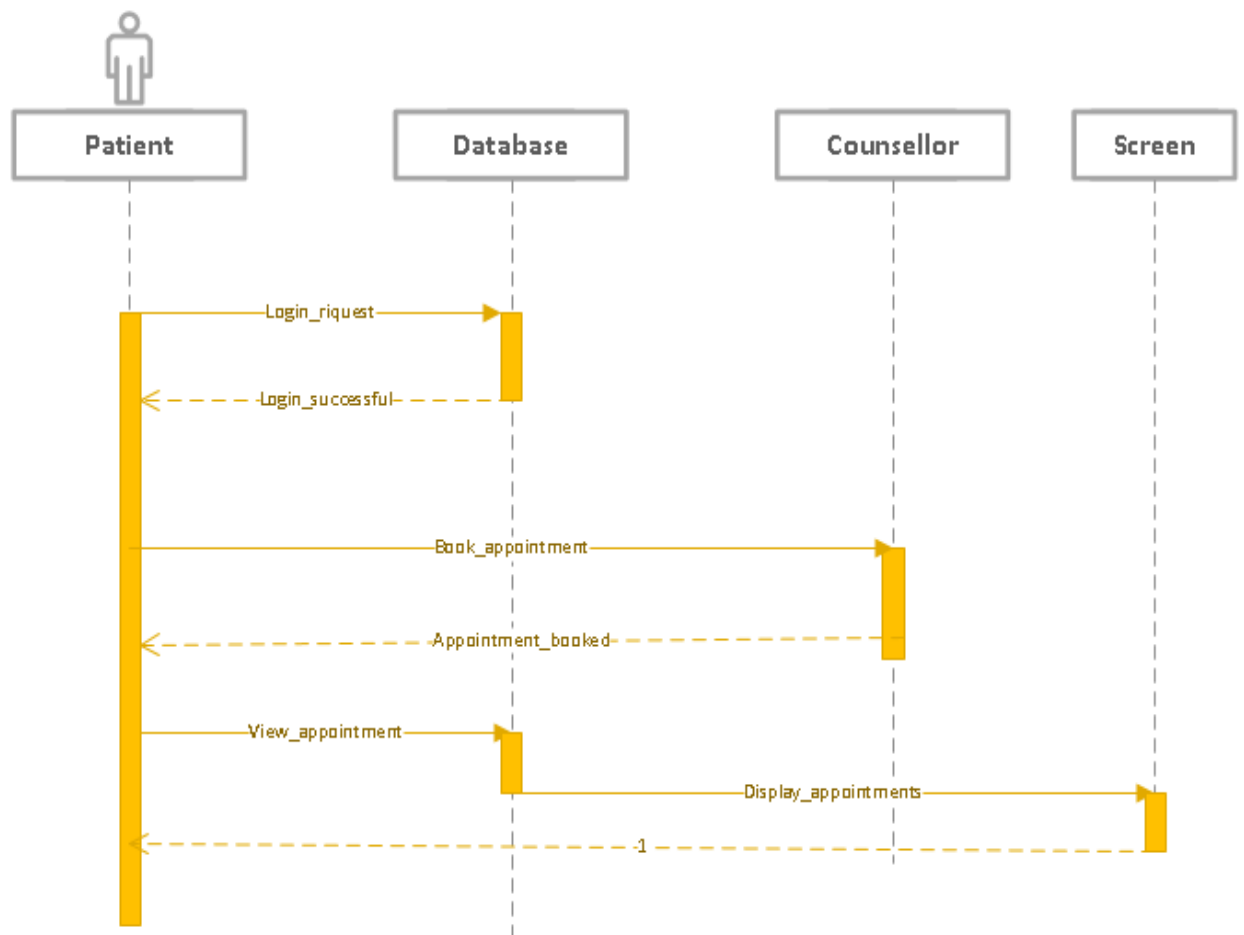
CLASS DIAGRAM

Figure 5



SEQUENCE DIAGRAM

Figure 6



NON-FUNCTIONAL REQUIREMENTS

Informally these are sometimes called the “ilities”, from attributes like stability and portability. Qualities, that are non-functional requirements, can be divided into two main categories: Execution qualities, such as security and usability, which are observable at run time.

PERFORMANCE REQUIREMENTS

The system should be accessible 24 hours a day, 7 days a week if the server is up and running correctly.

In the implementation of Role Based & Time Bound Access Control, users of the system shall be able to perform the activities or tasks according to their roles.

The system should be well documented and provide simple navigation, so future individuals can keep the site up-to-date.

SAFETY AND SECURITY REQUIREMENTS

The secure system shall be able to verify the users while login. In-order to maintain correct information and features on the site, only the admin has the full functionality to add, modify the users. After being verified, the doctor will be able to see the appointments on his dashboard and will treat the patient as the time slot is being assigned by the counsellor, and will not be allowed to access the patient details after the time has passed.

SOFTWARE QUALITY ATTRIBUTES

The non-functional requirements of the software used in Role Based & Time Bound Access Control includes Maintainability, Portability, Reusability, Compatibility, Resource Utilization and Serviceability.

Maintainability

The system should be easily maintainable for the average administrator and user.

Using the secure log-in, these users should be able to access the data as per their roles. In-order to do so, the system should be well documented and provide simple navigation, so future individuals can keep the site up-to-date.

Portability

The system should allow the authorized users to access the site from any web browser, as well as through a mobile device.

Reusability

The system will allow the administrator to access the database, in order to maintain and update the information as needed. The system will allow the registered users to login and view the information as per their authority and privileges assigned to them by the admin using the user interface.

Application Affinity/Compatibility

The system should be compatible with an internet connection of at least 56K and modern web browser such as Firefox, Internet Explorer, Safari and Google Chrome, regardless of what operating system that users is using.

Resource Utilization

The system shall require input from a mouse and keyboard, an output from a monitor and steady Internet connection to fulfil its user's requirements.

Serviceability

Any future modifications or services that Role Based & Time Bound Access Control requires, shall to be easily implemented and managed for the average administrator with this system due to the well-designed layout of the site and its corresponding documentation.

Testing Report

INTRODUCTION

Project Basic Information

Name of the Project: Role Based & Time Bound Access Control.

Table 1:

Start Date	23 August 2016
End Date	6 December 2016
Test Report Submission	7 December 2016

This report is intended to highlight the importance of result reporting in the context of software testing. Result reporting can be at various stages of testing like system, integration etc. This document addresses some of the areas of result reporting at a high level involving independent system testing (Black box testing) keeping in mind, the customer as one of the audience.

One of the important facts of software development life cycle is testing. Software testing is an area that is being considered and given utmost importance in the world of fast changing technology. There are various stages at which testing is done.

Reporting format varies depending on the stage of the testing in the development life cycle at each stage.

Transparency involved during testing - white box or black box testing. Types of testing involved like Functional, Performance/Load/Stress, Disaster recovery etc. Independent system testing takes about 20 to 50 % of the development time depending on various combinations of the types of tests conducted on a product. The complex the testing gets, more would be the effort required. Of course, the more a product is tested, the better the quality would be. Typically, test result reporting would consist of about 5 to 10% of this effort. Result reporting is very important and even more so when the product has failed testing.

Tools, Techniques and Methodologies

The following tools, techniques and methods would be used in the project for the specified purpose: -

Coding: PHP, PostgreSQL, HTML, JavaScript, Bootstrap.

Testing: Sublime Text3, Xampp server, SQLShell, Web browser.

Documentation: MS Word, Rational Rose, MS Visio.

Testing Standards

The following tools, techniques and methods would be used for the project for the specified purpose:

Tested Features- The following list describe the tested features:

DOCTOR:

- Registration (validation of all fields)
- Login
- View Appointment

SUPER ADMIN:

- Add admin
- Login
- Change password
- Change role
- Add counsellor
- View doctor details
- View patient details

ADMIN:

- Login
- Change password
- Add counsellor
- View doctor details
- View patient details

PATIENT:

- Registration (validation of all fields)
- Login
- Book Appointment
- View Appointment

COUNSELLOR:

- Login
- Cancel appointment
- View appointment requests
- View doctor details
- View patient details

Table 2:

S No.	Name	Enrolment No.
1	Abhishek Kumar Gupta	9914103169
2	Deepanshu Kathuria	9914103177

Table 3:

S No	Description	Expected O/p	Actual O/p	Remark
1	Empty text fields on the code insertion pages.	Alert message Validation to fill required details	Required alert Messages.	Correct
2	Transfer of data from frontend to database.	Values inserted in database.	Values inserted in database correctly.	Correct
3	Right click on the webpage.	Not allowed.	Could not right click.	Correct
4	All text fields are validated from the database.	Correct Values should be printed.	Correct Values are printed.	Correct
5	Every link directs to its respective web page.	All links correctly direct to their web pages.	All links correctly direct to their web pages.	Correct
6	Top to bottom linking in one go.	Move to top of the page.	Moving to the top of the page.	Correct
7	Change role activity performed by super admin.	Role should be changed.	Role changed correctly.	Correct
8	Time-period for access of Information by doctor.	Information should be accessed for the limited time.	Information had been accessed for specific time period.	Correct
9	Every page auto-refreshes itself after few seconds.	Every page auto-refreshes itself updates.	Every page does not auto-refreshes itself for updates.	Incorrect
10	Privilege changes after changing role.	Privilege should be changed after changing role.	Privilege changed after changing role	Correct

UNIT TESTING

Unit testing is a method of testing that verifies the individual units of source code are working properly. The goal of unit testing is to isolate each part of the program and show that the individual parts are correct.

LOAD TESTING

Load testing is the process of creating demand on a system or device and measuring its response. It generally refers to the practice of modelling the expected usage of a software program by simulating multiple users accessing the program concurrently. As such, this testing is most relevant for multi-user systems; often one built using a client/server model, such as web servers etc. The servers can bear up to 1000 hits per query.

SYSTEM TESTING

Once the entire system has been built then it must be tested against the Software Requirement Specification and System Specification to check if it delivers the features required. System testing can involve several specialist types of test to see if all the functional and non-functional requirements have been met.

PERFORMANCE TESTING

The system should meet the performance requirements as mentioned in the Vision document. The performance will be evaluated based on the response time of the GUI and the database commands.

The network performance has been tested on both low and high bandwidths and result are displayed in the pictures below. All the queries are redirected to an online web server and the acceptance of the request and the appropriate result depends upon the network speed.

MANUAL TESTING

Manual Testing will be done to ensure the correctness of various parts of the code using test cases generated by the tester.

PASS/FAIL CRITERIA

The system should satisfy all the functional requirements, in the Vision document. Each feature to be tested will be evaluated against its requirement as stated above in the Document. The pass or fail of a test depends on whether the system meets with all the post conditions.

OBSERVATIONS

Average response time is low for home page because it doesn't have many database interactions.

On changing role of a particular user, privileges also changed according to the role.

The doctor can access the information of patient for specific time period.


IMAGES

There are five roles in our project-Doctor, Super Admin, Admin, Patient and Counsellor. Different roles have different authorities and privileges. The users' dashboards and other pages will appear as below: -

SUPER ADMIN DASHBOARD

Figure 7


iHealthCare

Doctor Details	Patient Details	Change Password	Add Counsellor	View Appointment	Change Roles	 super_admin ▾
----------------	-----------------	-----------------	----------------	------------------	--------------	---

CHANGE ROLE

Figure 8

iHealthCare

Doctor Details	Change Password	Add Counsellor	Change Roles	 admin ▾
----------------	-----------------	----------------	--------------	---

Change Role

Username :

Present Role

Counsellor ▾

Change Role To

Counsellor ▾

Confirm

ADMIN DASHBOARD

Figure 9

iHealthCare

Doctor Details	Patient Details	Change Password	Add Counsellor	View Appointment	admin ▾
----------------	-----------------	-----------------	----------------	------------------	---------

ADD COUNSELLOR

Figure 10

Doctor Details	Patient Details	Change Password	Add Counsellor	View Appointment	admin ▾
----------------	-----------------	-----------------	----------------	------------------	---------


Counsellor Registration Form

First Name :	<input type="text" value="First Name"/>
Last Name :	<input type="text" value="Last Name"/>
Date of Birth :	<input type="text" value="dd-mm-yyyy"/>
Gender :	<input type="text" value="Male"/>
Username :	<input type="text" value="username"/>
Password :	<input type="text" value="Password"/>
Department :	<input type="text" value="Department 1"/>
Adhar No. :	<input type="text" value="adhar no."/>
State :	<input type="text" value="State"/>
Pin Code :	<input type="text" value="pin"/>
Email :	<input type="text" value="email id"/>
Mobile No. :	<input type="text" value="Mobile No."/>
	<input type="button" value="Register"/>

COUNSELLOR DASHBOARD

Figure 11

iHealthCare

[Appointments Request](#) [Feedback](#) [Cancel Appointment](#) [Doctor Details](#) 

Counsellor Information

Name : deepanshu kathuria

Date of Birth : 2016-12-10

Gender : Male

Mobile No. : 9999999999

Email id. : ya@yahoo.com

Adhar No. : 1234


State : haryana

Pin code : 125

Doctor Details

Figure 12

iHealthCare

[Doctor Details](#) 

Doctor Details

User-id :

[View Details](#)

PATIENT DASHBOARD

Figure 13

iHealthCare

[Book Appointment](#) [View Appointments](#) [Feedback](#) 👤 555 ▾

Patient Information

Name : jk kk

Date of Birth : 2016-12-16

Gender : Male

Mobile No. : 65656565

Email id. : da@yahoo.com

Adhar No. : 9999

State : hhh

Pin code : 5454

Book Appointment

Select Date :

Disease :

Confirm

Cancel

PROJECT EVALUATION

Introduction

This document evaluates the experience of implementation of Role Based and Time Bound Access Control. A brief description of the tools, process, techniques employed as well as the mistakes made is presented so that lessons are documented and learned.

Problems faced

The following are the problems faced during the implementation of project.

JavaScript in PHP

The main problem we faced was inexperience in JavaScript in PHP.

Time Bound Access Control

While implementing Time Bound Access Control, we faced the problems like auto refreshing the page using php.

So, we used to refer to YouTube video tutorials and some good website like stack overflow etc.

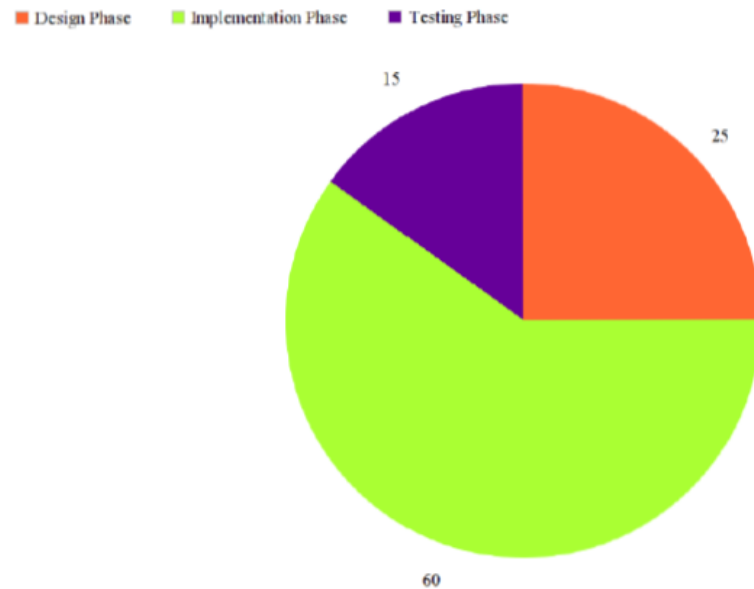
Pie chart below shows the time spent for each task at each phase. Design is the most important task that was performed at last two phases. Coding was performed in last two phases but mainly in the implementation phase.

Design Phase: 3 weeks

Implementation Phase: 8 weeks

Testing phase: 2.5 weeks

Figure 14



Lessons Learned

Programming

Role Based & Time Bound Project helped us to improve our confidence level in PHP Programming. Help us learn the basics in PostgreSQL and making us familiar with JavaScript.

Time Management

Since this Project is done by our team we have learnt how to manage time during the Software Life Cycle Process. We have also learned how to face tense situations and meet the deadlines. This would add as a good experience for our future job perspective.

UML and Software Lifecycle

As software student though we have good knowledge in UML and Software LIFE cycle we never had any good practical experience regarding them. Through this project, we have learnt how to develop a project following the various stages in Software Life Cycle.

Documentation

We always had a feeling that we are not good at documentation. But through this project we believe that we have improved our Documentation skills.

GANTT CHART

Task	Start Date	Task Data	Days to Complete
Task 1	8/23/2016	Study a lot of research papers in every field	13
Task 2	9/10/2016	Summary of Research Papers	7
Task 3	9/10/2016	Detailed Discussion on Research Papers	14
Task 4	10/1/2016	Topic Chosen: Role Based and Time Bound Access Control	11
Task 5	10/1/2016	Decide the title of website	10
Task 6	10/2/2016	Layout of The Website	18
Task 7	10/6/2016	Backhand and Database work	65

CONCLUSION AND FUTURE SCOPE

The main objective of our project ‘Role Based & Time Bound Access Control’ is to implementation of Role Based and Time Bound Access Control for database security purpose. This access control model emphasizes more on flexibility of roles and has the capacity to control the access of sensitive data from time dimension. For role based, we have developed a privacy-aware method. We have employed a time bound access, in which a doctor can view patient’s appointment for specific time-period. RBTBAC model is more suitable since it offers high efficiency and better security and privacy for users.

We can consider much future scope of this project. The following are: -.

- RBTBAC model can be applied to many different fields, especially sensitive information system such as government or military systems, banking systems and e-commerce systems.
- The RBTBAC model can be used in the systems storing a mass of data on untrusted remote DB or cloud.
- It can be used to guarantee better security and privacy of data sharing.

References

<https://en.wikipedia.org/>

<https://www.php.net/>

<https://www.youtube.com/>

<https://www.zentut.com/>

<https://www.postgresql.org/>

<http://www.w3schools.com/>

Research Papers: -

1. Role Based and Time Bound Access and Management of EHR Data.

Published on: 21 June 2013

Authors: Rui Zhang, Ling Liu, Rui Xue.

2. Published in SEA '07 Proceedings of the 11th IASTED International Conference on Software Engineering and Applications, 11 June 2007