A
Project Report
On

Development of Web Based Portal for Central Authentication and Authorization Services(CAAS)

Developed At

Institute For Plasma Research

Near Indira Bridge,

Bhat, Gandhinagar-382428

Developed By

Patel Nil - Department Of IT, D. D. University

Patel Charmy - Department Of IT, D. D. University

Guided By

Internal Guide :
Prof. R. S.Chhajed
Dept. of Information Technology
Faculty of Technology
Dharmsinh Desai University

External Guide :
Vijay Patel, Scientific Officer-F
Sharad Jash, Scientific Assistant
Computer-Center
Institute for Plasma Research

Department of Information Technology
Faculty of Technology, Dharmsinh Desai University
College Road, Nadiad-387001
April - 2018

CANDIDATE'S DECLARATION

We declare that final semester report entitled " Development Of Web Based Portal For Central Aunthentication And Authorization Services(CAAS)" is our own work conducted under the supervision of the external guide Sharad Jash and Vijay Patel from Institute For Plasma Research.

We further declare that to the best of our knowledge the report for B.Tech. final semester does not contain part of the work which has been submitted for the award of B.Tech Degree either in this or any other university without proper citation.

Also we declare that following students also worked in this project.

Patel Nil - IT Department, D.D. University
Patel Charmy - IT Department, D.D. University

Candidate's Signature
Candidate's Name: Charmy Patel
Student ID: 14ITUOF048

Candidate's Signature
Candidate's Name: Nil Patel
Student ID: 14ITUOS090

DHARMSINH DESAI UNIVERSITY
NADIAD-387001, GUJARAT

CERTIFICATE

This is to certify that the project entitled " Development Of Web Based Portal For Central Authentication And Authorization Services" is a bonafide report of the work carried out by

NO	NAME	ID
1	NIL PATEL	14ITUOS090
2	CHARMY PATEL	14ITUOS048

of Department of Information Technology, semester VIII, under the guidance and supervision for the award of the degree of Bachelor of Technology at Dharamsinh Desai University, Nadiad (Gujarat) .They were involved in Project training during academic year 2017-2018.

Prof. R.S. Chhajed
(Project Guide),
Department of Information Technology,
Faculty of Technology,
Dharmsinh Desai University, Nadiad
Date:

Prof. R.S.Chhajed,
Head , Department of Information Technology,
Faculty of Technology,
Dharmsinh Desai University, Nadiad
Date:

ACKNOWLEDGEMENT

Every project big or small is successful largely due to the effort of a number of people who have always given their valuable advice or lent a helping hand. We sincerely appreciate the inspiration; support and guidance of all those people who have been instrumental in making this project a success.

We, Charmy Patel and Nil Patel students of Dharamsinh Desai University, are grateful to "The Department of Information Technology" and "Institute for Plasma Research" for the confidence bestowed in us entrusting us with the project entitled: "Development of Web based Portal for Central Authentication And Authorization".

It has been great honor and privilege to undergo training at Institute for Plasma Research – Bhat, Gandhinagar. I am very thankful to Mr. Vijay Patel, Engineer at Institute for Plasma Research and Mr. Sharad Jash, Scientific Assistant at Institute for Plasma Research for providing all the support and guidance to meet our project requirements. Their constant tutelage and willingness to share and instigate their knowledge made us understand this project and its manifestations in great depths and helped us to complete the assigned tasks.

At this juncture we feel great honor in expressing my deepest thanks to my Internal guide and mentor, Prof. R.S.Chhajed for providing significant advice, guidance, ideas and insights without which this project would have not been possible.

Thanking Everyone.

1. Patel Nil
2. Patel Charmy

TABLE OF CONTENTS

ABSTRACT

The proposed project would require Designing and Development of Central authentication and authorization service (CAAS) for Institute For Plasma Research (IPR). The proposed system is based on LDAP protocol. The project involves installation and configuration of LDAP tool and import the user's data including organization breakdown structure. Develop a web based application for user management, password reset and other administrator tasks via a single access and management tool.

The work involves development easy to use GUI for LDAP management system which allows users to manage their profiles and update them in controlled manner. It allows administrator and users to manage user accounts and different services of IPR's network such as Internet via wired or wireless or managing web-application roles via LDAP.

COMAPANY PROFILE

The Institute for Plasma Research can trace its roots back to early 1970's when a coherent and interactive program of theoretical and experimental studies in plasma physics with an orientation towards understanding space plasma phenomena was established at the Physical Research laboratory

The early studies were on simulation of E x B instabilities characteristic of the equatorial electrojet, plasma-neutral gas interaction with relevance to the cometary plasma-solar wind interaction and single particle confinement in non-adiabatic magnetic mirrors. Experiments on non-linear ion acoustic waves and double layers were added later. High power plasma experiments using intense electron beams to form compact toroids and electron rings in toroidal devices started in 1978 reflected a re-orientation to fusion-relevant experiments.

A proposal to the Government of India to initiate studies on magnetically confined high temperature plasmas was accepted in 1982 and resulted in establishment of the Plasma Physics Program (PPP) supported by the Department of Science and Technology. Design and engineering of India's first tokamak ADITYA started at the same time. In 1984 the activities moved into an independent campus at Bhat village in the outskirts of Ahmedabad city in 1984.

The PPP evolved into the autonomous Institute for Plasma Research under the Department of Science and Technology in 1986. With the commissioning of ADITYA in 1989, full-fledged tokamak experiments started. A dynamic experimental program focusing on transport due to edge turbulence has resulted in major discoveries in this field. This period also saw development of new programs in plasma processing and basic and computational plasma research.

With the decision to build the second generation superconducting steady state tokamak SST-1 capable of 1000 second operation in 1995, the institute grew rapidly and came under the administrative umbrella of the Department of Atomic Energy. Major new programs in pulsed power, advanced diagnostics, computer modelling, development of RF and neutral beam heating systems etc. also came into being. The industrial plasma activities were reorganised under the Facilitation Centre for Industrial Plasma Technologies and moved to a separate campus in Gandhinagar in 1998.

The Institute for Plasma Research can trace its roots back to early 1970's when a coherent and interactive program of theoretical and experimental studies in plasma physics with an orientation towards understanding space plasma phenomena was established at the Physical Research laboratory.

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1 PROJECT DETAILS

The proposed project would require Designing and Development of our existing Central authentication and authorization service (CAAS) having captive portal used for user management, password reset and other administrator tasks via a single access and management portal.

1.2 PURPOSE OF PROJECT

1. Development of web based captive Portal for central authentication/authorization of users.
2. RBAC based portal for management with Captcha login.
3. Guest user management portal.
4. UI Customization for identity management server.
5. Addition/deletion/updation of owners/members of group.
6. Auditing of all tasks executed by approvers/admins/users.
7. Paperless way to create users based on approvals on different levels.
8. On click assignment of roles for accessing different services.
9. Creation/management of users with groups/projects with the group owners.

1.3 PROJECT SCOPE

The Authentication System provides four types of user Administrator, Guest-Manager, IPR-User and Guest-User. The system will allow Administrator to access user profiles as well as activate/de-activate their account or edit their profile. It allows user to access and edit their profile. Allows Guest Manager to access and edit Guest User Profiles. Admin can see logs showing detailed information.

1.4 OBJECTIVE

1. Objective of this system is to give internet access to different users.
2. This system will maintain record of which type of internet access is being provided to the users/guest.
3. It supports mailing system through which division head as well as CC head can approve the user for getting access to the internet.
4. Admin can have full details of the user through Log-analyzer.

1.5 TECHNOLOGY AND LITERATURE REVIEW

Following section of this document will focus on describing the tools used for developing the application.

LDAP

LDAP stands for Lightweight Directory Access Protocol. It is an application protocol used over an IP network to manage and access the distributed directory information service. It provides systematic set of records, usually organized in hierarchical structure. It is organized in tree hierarchy consisting of levels like: Root, Country, Organization, etc.

A directory service is a kind of database or data store, but not necessarily a relational database. The structure is usually much simpler, storing hierarchical collections of name-value pairs, e.g. lastName=Smith, firstName=John. The structure of LDAP - the server is capable of holding subtree and it's childrens.

LDAP syntax :
CN=Bob,OU=Users,DC=Youtube,DC=Com
CN = Canonical Name (object or name)
OU = Organizational Unit (Folder in Active directory)
DC = Domain Controller (Where it is)

The main benefit of using LDAP is that information for an entire organization can be consolidated into a central repository. For example, rather than managing user lists for each group within an organization, LDAP can be used as a central directory accessible from anywhere on the network. And because LDAP supports Secure Sockets Layer (SSL) and Transport Layer Security (TLS), sensitive data can be protected from prying eyes.

The primary focus of LDAP is to define directory data records in a standard format so that any user can access and store information locally. The way LDAP achive this is through object classes, which are made up of attributes. Every entry in and LDAP server defines the object classes if accepts and the way it handle, request to those object classes through a schema or data definition.

The Lightweight Directory Access Protocol (LDAP) is cross-platform directory and set of protocols designed to access and maintain information directories. LDAP can be used for numerous purposes, like user and group management, system configuration management, or address management.

The LDAP contains major three parts they are:
1. Data format : Defines how the directory information is stored and recalled
2. The Protocol : Defines how clients and server in tract with each other
3. The API : Defines how programs in traction LDAP server , Novell LDAP API, sun's JNDI etc

The LDAP data format

The LDAP data format is most difficult part. The data format in LDAP is in cross culture comparing with DBMS. Each entry in LDAP participates in a global naming schema called namespace.

In LDAP, the top level is called the domain. There can be several domains in an LDAP implementation. This is because, like X.500, LDAP is designed to help provide a global directory service. Typically, the top-level domain will just be the root of the organisation or organisations that a particular company manages.

The branches are in the form of organisational units, which are usually departments in an organisation, but can be any convenient sub-division of that organisation. Each entry that is not a domain and not an organisational unit is called a leaf. It is wise to have people in an organisation entries in both an organisational unit called People, as well *** under their department released organisational unit, since this can speed up searches.

Each entry identified by unique dn.

Domains, organisational units and every other type of information that is contained in the LDAP server must have a unique entry, and these entries must be a member of a particular object class.

Schemas, Objects and LDAP

Each LDAP server defines the object classes and the object hierarchy in a schema. Remember the schema is the blue print of the LDAP server. The schema defines the format of the records in the database.

The entries in LDAP are in the form of Objects. The object paradigm was chosen because it allows any form of data to be stored in and LDAP server. Also, objects can be extended without having the completely redefined. To illustrate what I mean, let's consider a database. In traditional database design which uses the familiar table, field model, you are fixed to a particular design once the database has been created. If you have a table of some kind and your want to add some fields to it, you would normally have to redesign and recreate the table from scratch. Under the object paradigm, however, if you have an object A and you want to use it to create Object B, you can just declare that you want to use all of the attributes available under object A and then add some new attributes to make Object B. Another neat off-shoot of this is that now all objects that are of type B are also objects of type A because Object B is derived from Object A.

By using objects, it's possible to store a wide variety of data, including for instance
1. The organisation itself, such as where it is located.
2. People, like their names and passwords.
3. Particular machine attributes, including password management information, such as how many times a user can re-try to authenticate and how old the password can be before it expires.
4.Other computers on the Internet, similar to DNS.

Attributes and Distinguished Names

When building a distinguished name (DN) for a person object, you want to make the lead entry in the DN the user ID, because this entry is probably unique for the entire server. If you don't have a user ID field, you may have to combine entries to make a unique entry. A possible combination would be one of common name and employee number.

Here is an example DN for a person object:
dn: uid=priyankpatel,ou=users,dc=ipr,dc=res,dc=in.

The LDAP protocol

A client starts an LDAP session by connecting to an LDAP server, by LDAP port [default on TCP port 389]. The client then sends operation requests to the server, and the server sends responses in turn.

Operations on LDAP protocol
1. Connects to the server.
2. Performs a series of operations with the server.
3. Disconnects from the server

This directory access protocol which can either act as a gateway to other service or provide service itself. LDAP follows standardised way of managing directory

information solutions of X.500. LDAP, is based on the X.500 standard, but significantly simpler and more readily adapted to meet custom needs.

X.500

The X.500 directory service is a global directory service. Its components cooperate to manage information about objects such as countries, organisations, people, machines, and so on in a worldwide scope. It provides the capability to look up information by name and to browse and search for information.

Advantages:
1. Global namespace
2. A plat form and data independent way of describing our directory information.

Why LDAP , Even X.500 Service is available?

Disadvantages :
1. Runs on the OSI communications stack
2. Difficult in implementation, because based on OSI rather than internet protocols.

Uses of LDAP
1. Central user database for multiple systems.
2. LAN Access authentication with Username and password
3.Wifi Access Authentication
4.Mail Server Access Authentication
5.LAN Access authentication with MAC address

The LDAP an Approach for Directory Service

The LDAP is updated with removing some of confusing and clunkier parts of the X.500 protocol land moving the underlying network communications protocol from OSI to TCP/IP and revealed the light weight Directory Access Protocol.

The original version of LDAP was created at the University of Michigan. LDAP version 3(LDAPV 3) has now made it possible to give us the power we need to finally synchronise all of our networking and directory information into manageable system.

LDAP implementation

The LDAP used in more secure purpose. For example a list of users can only read the values, list of users can only update etc.,. The entry can be securely maintained.
LDAP is particularly useful for storing information that you wish to read from many locations, but update infrequently.
For example in a company the LDAP may be used for the following purposes
1. Company Employee information like address, phone number, email etc.
2. Company vendor, service provides etc.
3. Configuration information for distributed software packages
4. Public certificates and security keys

The LDAP can be implemented when we receive the following requirements
1. Data to be available cross-platform
2. Data accessed with different application
3.Data may not frequently updated (not a transitional information)
4.Instead of using Flat file we can implement LDAP

Advantages of LDAP for industries (Reasons for Choose LDAP)

1. Open Solution: It's and open standard any one allowed to develop LDAP servers/clients, providing suggestions for future enhancements of existing implementations without paying any fee.

2. Severity: LDAP uses the same object data format as X.500 where each object is made up of attributes that are indexed and referenced by a distinguished name. The name of each object is unique to fit into a global namespace that helps determine the relationship of the object and allows for the object to be references uniquely.
"In a directory no object stands alone. Each object has a relationship to every other object in the directory service."

- Standard API

There are lot of peoples providing APIS for LDAP like Michigan API, Netscape LDAP SDK's, Sun Java JNDI (Java Naming & directory Interfaces), Microsoft Achieve Directory Services Interfaces (AOSI), Per LOAP SOK etc.

3. Gateway Services: Through LDAP it's easy to make gateways like web, and have LDAP provide a gateway to other directory services. By providing gateway we make it very simple to use this client to access directory information.

4. Vendor Supports: There are so many vendors released their own LDAP serves Netscape, Novell, Sun, Microsoft.

5. LDAP Terminology: It's called an object or an object entry.

6. Entry: An entry in an LDAP, we call as a record in LDAP server. Its akin to record in a traditional data base.

7. Attributes: An entry is made up of attributes. It's divided into names value pairs. It's very similar to fields in database management system. An attribute will have only one name but it can have more than one value. The values may be text, Binary depends on the definition of and attribute.
dn: The entry is uniquely identified by distinguished name (DN).
RDN: Relative distinguished name, the dn is made up of components, which is called RDN.
8. Object classes: Identifies what attributes are available to an entry. These are very akin to tables in data base management system. It's differing from 'tables' in the way of there are extensible.

For example object class that allows a set of attributes like last name, a first name, email etc, and if we want add extra attributes to this object class, we can declare anew object class that extends from earlier one. This new object class would have the ability to recognize the older object class attribute as well as the new ones we added.

9. Schema: Each LDAP server have schema. The schema is a blueprint of the server and it specifies all of the object classes and attributes that are available to be searched and stored in LDAP server.

The directory server is accessed like a tree and is referred to on a directory information tree (DIT). The directory server itself is called as a directory service agent (DSA).

If an objectclass is not present in schema, then it's not possible to add the object class. For Example OpenLDAP, the Schema imported in the file slapd.conf
Include /usr/local/etc/openldap/schema/core.schema
include /usr/local/etc/openldap/schema/cosine.schema
include /usr/local/etc/openldap/schema/inetorgperson.schema

Application:

1. It can be used in authentication and authorization services.

- LDAP clients can use persistent connections in order to eliminate the need to create a new connection for each request, and the server can maintain state information for the connection so that it's not necessary to repeat things (e.g., authentication credentials, session cookies, etc.) in each request.
- LDAP is a staple of enterprise IT infrastructures for authenticating and storing data about users, and its deployment runs the gamut of business types, including heavy use in areas like telecommunications, finance, manufacturing, retail, education, and government.

2. Security Feature:

- Supports automatically encoding passwords so that they are stored in encrypted format making it very difficult to determine clear-text credentials for admin too.
- Supports extensible authentication via the SASL framework. This provides a range of authentication options that include non-password-based mechanisms like certificates and Kerberos tickets, and some servers provide a number of options for multi-factor authentication.
- Support for numerous password policy features like password expiration, password quality validation, and account lockout from too many failed attempts.

3. General-purpose data storage. LDAP servers are often used to hold information about users and groups.

4. Scalability. A data repository isn't any good if it can't cope with the volume of data or amount of load that your applications demand. Many modern LDAP servers can scale vertically to take advantage of systems with tons of memory and a lot of CPUs, and horizontally to spread the load across multiple systems.

5. High availability. Redundancy is a very good thing. All LDAP servers of any consequence support replication, and most support some form of multimaster replication that allow writes to any node in the environment. A number of vendors also offer LDAP proxy servers, which can help further improve performance by

balancing traffic across multiple servers (while also preferring local servers to those in other data centers), and can improve availability by automatically routing around servers that become unavailable or start to misbehave. Further, some LDAP client libraries (like the UnboundID LDAP SDK for Java) also provide load balancing and failover capabilities.

6. Disaster Recovery. No software is perfect, and neither is the hardware that it runs on. Redundancy is good for hiding failures from end users, but backup and restore capabilities are essential for recovering from them. Not only do all LDAP servers offer some kind of backup and restore mechanism, but they also support exporting data in the standard LDIF format that can be read by not only other LDAP servers (including those from other vendors), but also by a wide range of client software.

7.Logging. LDAP servers typically provide at least access and error log files so that administrators can keep track of what's going on in the directory environment and troubleshoot any problems that may arise. Many servers offer a number of other logging options.

8. LDAP is a well-defined public standard, with the core of the protocol defined in RFCs, and numerous other RFCs and IETF drafts describing other capabilities. Anyone can read the specs and create their own client and/or server (and in fact, if you want to really learn the ins and outs of the protocol, that's a great way to do it), and any standards-compliant client should be able to interact with any standards-compliant server. This means that there are several good LDAP directory servers out there, and there are LDAP client libraries out there for just about whatever programming language you want to use.

Laravel

Laravel is a free, open-source PHP web framework, created by Taylor Otwell and intended for the development of web applications following the model–view–controller (MVC) architectural pattern. Some of the features of Laravel are a modular packaging system with a dedicated dependency manager, different ways for accessing relational databases, utilities that aid in application deployment and maintenance, and its orientation toward syntactic sugar.

Laravel is a web application framework with expressive, elegant syntax. We believe development must be an enjoyable, creative experience to be truly fulfilling. Laravel attempts to take the pain out of development by easing common tasks used in the majority of web projects, such as authentication, routing, sessions, and caching.

Laravel aims to make the development process a pleasing one for the developer without sacrificing application functionality. Happy developers make the best code. To this end, we've attempted to combine the very best of what we have seen in other web frameworks, including frameworks implemented in other languages, such as Ruby on Rails, ASP.NET MVC, and Sinatra.

Laravel is accessible, yet powerful, providing powerful tools needed for large, robust applications. A superb inversion of control container, expressive migration system,

and tightly integrated unit testing support give you the tools you need to build any application with which you are tasked.

OpenDJ

OpenDJ is a directory server which implements a wide range of Lightweight Directory Access Protocol and related standards, including full compliance with LDAPv3 but also support for Directory Service Markup Language . Written in Java, OpenDJ offers multi-master replication, access control, and many extensions.

OpenDJ is a new LDAP compliant directory service, providing high performance, availability and secure store for identities managed by an enterprise. It is one of the fastest and simplest directory server to deploy and manage.

An open source, lightweight, embeddable directory that can easily share real-time customer, device, and user identity data across enterprise, cloud, social, and mobile environments.

- Massive data scale and high availability providings developers with ultra-lightweight ways to access identity data.
- High Performance - ms response times & tens of thousands of w/r per sec
- Multi Master replication for high availability

Table 1.1 Technology And Literature Review

Sr. No	Type	Description
1	Operating System	Linux
2	Application for Development Environment	PHP
3	Tools for Development	OpenDJ, Text Editor, Xamp
4	Data Access Technology	LDIF
5	Web Server	Apache
6	Database Server	Develop.ipr.res.in, MySql

2. PROJECT MANAGEMENT

Project management is the discipline of planning, organizing, motivating, and controlling resources to achieve specific goals. A project is a temporary attempt with a defined beginning and end (usually time constrained, and often constrained by funding or deliverables), undertaken to meet unique goals and objectives, typically to bring about beneficial change or added value. In practice, the management is managing such projects. The primary challenge of project management is to achieve all of the project goals and objectives. The primary constraints are scope, time, quality and budget.

2.1 FEASIBILITY STUDY

Preliminary investigation examine project feasibility, the likelihood the system will be useful to the organization. The main objective of the feasibility study is to test the Technical, Operational, Time-Schedule feasibility for providing internet access All system is feasible if there are unlimited resources and infinite time.

2.1.1 Technical Feasibility

Technical analysis evaluates technical merits of the system at the same time collection additional information about performance, reliability, maintainability and productivity. In some cases, an analysis step also includes research and design. The project being developed uses PHP and LDAP protocol. The softwares for them are available freely on the internet. As it is already developed project it meets all the aspect of technical feasibility.

2.1.2 Time Schedule Feasibility

Time schedule plays a vital role in the project. If the project is not delivered on due time then it can cause a project failure. Hence, time management should be taken care of by the project manager. It should also be taken care that the staff, which is related with the project, should be able to complete the technical tasks in given schedule. If the current staff is not sufficient in completing the project tasks, project manager should allot more technical persons

2.1.3 Operational Feasibility

The users of the client organization should be able to operate the application easily, for whom the application is developed, to gain the advantages of the application. This demands good user interface. During the application development process prototypes of the application are developed initially and are shown to client, so the client can give some additional changes as per their requirements. As the application is developed and modified as per the comments of the users, there is very little possibility that there will be resistance from end users.

2.1.4 Implementation Feasibility

Implementation feasibility is concerned with specifying external resources and software's that will successfully satisfy the user requirements. We have given more importance to external resources and configuration of the system rather than the actual map of the hardware. Financial consideration was also considered at this stage. A proper implementation is essential to provide a reliable system meet the requirement of the organization. Implementation is the stage in the project where the theoretical design is turned into a working system.

2.2 PROJECT PLANNING

2.2.1 Project Development Approach and Justification

For the project development we have utilized the spiral modal software development approach. Spiral model is similar to incremental model has four phases: Planning, Risk Analysis, Engineering and Evaluation.

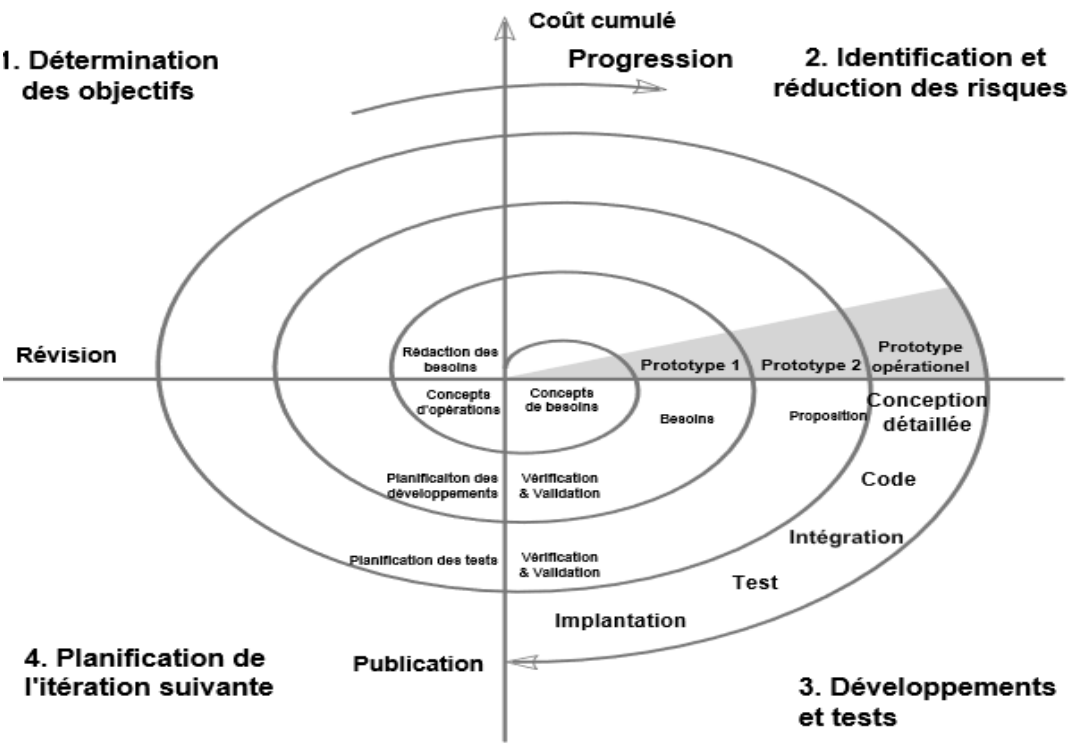


Fig 2.1 Spiral Model

Advantages of Spiral Model are:

- High amount of risk analysis hence, avoidance of risk is enhanced.
- It is good for large and mission-critical projects.
- Additional functionalities can be added at a later date
- When cost and risk evaluation is important.
- For medium to high-risk projects.
- Long-term project commitment unwise because of potential changes to economic priorities.
- Requirements are complex.

2.2.2 Project Plan

In managing any project the whole plan of the project is made before the actual implementation. The plan of the project helps team to work as per the schedule and helps to successfully complete the project.

To plan the project the ain requirements that are calculated are cost, duratin, effort, scheduling, man-power, resource allocation, risk management, etc. The plan of our project is a s follows:

1. Gather the definition.
2. Check whether the definition is feasible or not in given deadline.
3. Requirement gathering.
4. Analysis on gathered requirements.
5. Designing.
6. Coding.
7. Testing.

2.2.3 Milestones and Deliverables

Management needs information. As software is intangible, this information can only be provided as documents that describe that state of the software being developed. Without this information, it is impossible to judge progress and cost estimates and schedules cannot be updated. When planning a project series of milestones are established.

Milestone:

1. Milestone is an end-point of the software process activity.
2. At each milestone there should be formal output, such as report, that can be represented to the management. The weekly report is submitted to project guide, which include day to day work report.
3. Milestone represents the end of the distinct, logical stage in the project.

Deliverables:

1. Deliverables is a project report that is delivered to the administrator of the project.
2. Deliverables are delivered to the administrators of our organization at the end of some major project phase such as specification, design, etc.
3. Milestones may be internal project results that are used by the project manager to check progress but which are not delivered to the administrator.

2.2.4 Roles and Responsibilities

Generally proposed Project Management structure for the development/customization, implementation and maintenance of the project is as follows:

1.Team Member

- ✓ Carrying out the work allocated
- ✓ Report on a daily basis to Project Lead

2. Requirement Analysis
   - ✓ The requirement analyst or engineer is the individual who has the primary responsibility to elicit, analyze, validate, specify, verify, and manage the real needs of the project.
   - ✓ He needs organizational skills, to work with the vast array of information gathered during elicitation and analysis and to cope with rapidly changing information.

3. Developer
   - ✓ Develop software, architecture, specifications and technical interfaces.
   - ✓ Design, initiate and handle technical designs and complex application features.
   - ✓ Initiate and drive major changes in programs, procedures and methodology.

4. Tester
   - ✓ Preparing Test Scenarios
   - ✓ Preparing Test Cases for module, integration and system testing.

2.2.5 Group Dependencies

The members of the project should be dedicated to the project and should in turn help each other in whatever problems concerning the project. They should report periodically to the project leader or the concerned faculty of the project reporting.

2.3 PROJECT SCHEDULING

2.3.1Project Scheduling chart

Table 2.1 Scheduling Chart

Task	Start	End	Dec	Jan	Feb	March
Learning	11/12/2017	31/12/2017	✓			
Analysis	1/1/2018	15/1/2018		✓		
Design	16/1/2018	11/2/2018		✓	✓	
Coding	17/2/2018	10/3/2018			✓	✓
Testing	11/3/2018	17/3/2018				✓
Maintanence	18/3/2018	23/3/2018				✓
Documentation	24/3/2018	31/3/2018				✓

3. SYSTEM REQUIREMENT STUDY

The following sections will introduce the numerous requirements of the system from the point of view of different users and will introduce a number of decisions that have been made regarding implementation. These sections also attempt to somewhat describe the role of each user group in the system, discussing their individual roles through the functions they can perform.

3.1 STUDY OF CURRENT SYSTEM

The current system involved Configuring and integrating Single Sign on functionality using Shibboleth version 3.3.1.The Shibboleth Identity Provider was configured so that it can be support SAML 1.1 and SAML 2.0. Further implementation of Service Provider can be done and be integrated in authentication flow in the Single sign-on.

3.2 PROBLEMS AND WEAKNESSES OF CURRENT SYSTEM

1. Developed in traditional classic style with PHP/js/jquery/smarty.
2. Configuration file common for user/admin.
3. Two different procedures for management for user and admin.
4. Requires internet for Captcha(Google Re-captcha).
5. No records of user approvals.

3.3 USER CHARACTERISTICS

The type of users who is dealing with the system are as follows

1. Admin
2. User
    - 2.1 CC-Head
    - 2.2 Div-Head

3.4 HARDWARE AND SOFTWARE REQUIREMENTS

1. OS - Centos 7/RHEL 7
2. Hardware - Minimum 8GB RAM with at least cpu speed 2.0Ghz clock.
3.Db - Mysql 5.7 community
4.LDAP - Opensource Opends/389DS
5.Httpd Apache - 2.4
6.Browsers with cookie enabled.

3.5 CONSTRAINTS

3.5.1 Regulatory Policies

As per the Company's policy any developer has to maintain the Coding Standards.
Also each and every user should maintain the subversion and commit the modification with appropriate comment so to have track of work and also of the code modification.
From the client's perspective:
1. Developer should use well known technology.
2. Developer should know how to operate computer

3.5.2 Reliability Requirements

The application does demand much reliability and it is fully assured that the particular information about the user should be secured and flow is maintained and accessed according to the rights. LDAP have the ability to replicate information widely in order to increase availability and reliability, while reducing response time.

3.5.3 Safety and Security Consideration

The system provides a tight security to every user's account. Users' accounts are secured by password mechanism which are stored to database. Captcha is less about cracking logging details and more about spam and bot-ing. Open forums are vulnerable as a simple script can load the page, fill the form in with garbage and submit over and over and it becomes difficult to get the script to work out the captcha and so the spamming almost stops completely as people are forced to do it by hand.

3.5.4 Criticality of the Application

Criticality means any occurrence of malfunction of the system or any accidental event in software which can damage the resources of software as well as hardware. As per my knowledge there is no criticality in our application.

To assure criticality of the system, it provides replication through multi-master and master-slave. In the Multi-Master replication, two or more servers act as master and all these are authoritative for any change in the LDAP directory. Queries from the clients are distributed across the multiple servers with the help of replication. As the master-slave replication is a one-way replication (from master to slave), only the master database is used for the write operations, while read operations may be spread on multiple slave databases.

3.6 ASSUMPTIONS AND DEPENDENCIES

The details related to the end users needs to be provided manually. Admin in the system works as a main role and tasks are predefined for administrator. 'develop.ipr.res.in' server is used to store the database. Users and Groups are already present in our directory. Bootstrap/AJAX,JS is used to develop the front end.

4. SYSTEM ANALYSIS

4.1 REQUIREMENTS OF NEW SYSTEM (SRS)

4.1.1 User Requirements

The user requirements include attending certain input restrictions to the systems. The user must be aware of the situations and conditions that meet the reliable accurate and feasible working of the system. The following are the responsibilities of the user:

- He must request a new account through internet of IPR. The developed site is only accessible through IPR.
- Before getting access to the directory services ,he/she requires approval from his/her respective section/division head as well as from CC-head.
- He must read the Acceptable Usage Policy before requesting a new account.

4.1.2 System Requirements

When a user access the website login screen will be displayed. If a user is member of Directory Admin also his account is enabled by Computer Center and division head he can login as Admin.

R1: User Login
Input: Login Credentials are given along with captcha.
Output: Home page of user is shown along with all details of the user.
State: Initial page of Website.

R1.1: General
Input: Login credentials are given or general tab is clicked.
Output: Initials, first name, last name and display name is shown.

R1.2: Personal
Input: Personal tab is clicked.
Output: Employee information is shown.

R1.3: Group
Input: Group tab is clicked.
Output: Groups of users are shown if he is member of any.

R1.4: Access-Rights
Input: Access-Rights tab is clicked.
Output: Access-Rights of user are shown.

R1.5: Change Password
Input: Change Password navigation header is clicked.
Output: Page requesting for new password is shown.

R1.5.1: Change password
Input: Change Password button is clicked.
Output: Password is changed.

R1.6: Search Users
Input: Username is given.
Output: Details regarding given user is shown.

R1.7: Edit Profile Picture
Input: Edit Profile Picture is clicked.
Output: Profile Picture is changed after selecting image of required criteria.

R1.8: User Request Management
Input: User Request Management navigation header is clicked.
Output: Dropdown list is displayed.

R1.8.1: User Request Approval
Input: User Request Approval tab is clicked.
Output: Page showing list of users in user's department requesting for internet Access is shown where he can select yes/no.

R1.8.2: CC-Head Approval
Input: CC-Head Approval tab is clicked.
Output: Page showing list of users requesting for internet Access is shown where he can select yes/no.

R1.9: Logout
Input: Logout navigation header is clicked.
Output: User is logged out from his account.

R2: Admin Login
Input: Login credentials along with captcha.
Output: Home page of Admin is shown along with

R2.1: Log Viewer
Input: Log Viewer navigation header is clicked.
Output: Logs are shown for admin ease.

R2.2: User Management
Input: User Management navigation header is clicked.
Output: Dropdown list containing all users, locked users, unlocked users, Guest users, and change password is shown .

R2.2.1: All Users
Input: All Users named dropdown list is clicked.
Output: Page displaying all users is shown in tabular form.

R2.2.1.1: Search
Input: Username is given and auto-searching facility is provided to find users.
Output: The particular user is displayed.

R2.2.1.2: User Details
Input: Particular User is clicked.
Output: Details of particular user is shown.

R2.2.1.3: Create New
Input: User Details are given.
Output: New user is created.

R2.2.2: Locked Users
Input: Locked Users named dropdown list is clicked.
Output: Page displaying locked users is shown in tabular form.

R2.2.2.1: Search
Input: Username is given and auto-searching facility is provided to find users.
Output: The particular user is displayed.

R2.2.2.2: User Details
Input: Particular User is clicked.
Output: Details of particular user is shown.

R2.2.3: Guest Users
Input: Guest Users named dropdown list is clicked.
Output: Page displaying all guest users is shown in tabular form.

R2.2.3.1: Search
Input: Username is given and auto-searching facility is provided to find users.
Output: The particular user is displayed.

R2.2.3.2: User Details
Input: Particular User is clicked.
Output: Details of particular user is shown.
Processing: All details of user is fetched.

R2.2.4: Unlocked Users
Input: Unlocked Users named dropdown list is clicked.
Output: Page displaying all unlocked users is shown in tabular form.

R2.2.4.1: Search
Input: Username is given and ajax facility is provided to find users.
Output: The particular user is displayed.

R2.2.4.2: User Details
Input: Particular User is clicked.
Output: Details of particular user is shown.

R2.2.5: Change Password
Input: Change password named dropdown list is clicked.
Output: Password is changed by admin of any user.

R2.3: Groups
Input: Group navigation header is clicked.
Output: Dropdown list containing Org groups, security groups is shown .

R2.3.1: Org Groups
Input: Org Groups named dropdown list is clicked.
Output: Page displaying all Org Groups is shown in tabular form.

R2.3.1.1: Search
Input: Group name is given and auto-searching facility is provided to find groups.
Output: The particular group is displayed.

R2.3.1.2: Group Details
Input: Particular Group is clicked.
Output: Details of particular Group is shown.

R2.3.1.2.1: Rename
Input: New group name is given.
Output: Group name is changed.

R2.3.1.2.2: Add Group
Input: Sub group name and owner name is given.
Output: New group is created under selected group.

R2.3.1.2.3: Add Member
Input: Member name is selected from modal popup.
Output: New member is added under selected group.

R2.3.1.2.4: Remove
Input: Member name is selected from tabular form.
Output: Member is removed from selected group.

R2.3.1.2.5: Change owner
Input: Owner name is given.
Output: Group Owner is changed.

R2.3.2: Security Groups
Input: Security Groups named dropdown list is clicked.
Output: Page displaying all Security Groups is shown in tabular form.

R2.3.2.1: Search
Input: Group name is given and auto-searching facility is provided to find groups.
Output: The particular group is displayed.

R2.3.2.2: Group Details
Input: Particular Group is clicked.
Output: Details of particular Group is shown.

R2.3.2.2.1 Rename
Input: New group name is given.
Output: Group name is changed.

R2.3.2.2.2: Add New Host
Input: Sub group name and owner name is given.
Output: New group is created under selected group.

R2.3.2.2.3: Add Member
Input: Member name is selected from modal popup.
Output: New member is added under selected group.

R2.3.2.2.4: Remove Member
Input: Member name is selected from tabular form.
Output: Member is removed from selected group.

R2.4: Deadline
Input: Deadline navigation header is clicked.
Output: List of users who have date of Completion in current month as well as in previous month.

R2.5: User Request Approval
Input: User Request Approval navigation header is clicked.
Output: List of users waiting for Approval for internet access are shown where admin can select yes/no.

R2.6: Logout
Input: Logout navigation header is clicked.
Output: Admin is logged out from his account.

R3: Request New Account
Dropdown list for requesting new account as employee or guest is shown.

R3.1: Employee
Input: Employee is selected from dropdown list..
Output: Registration form for employee is displayed.

R3.1: Guest
Input: Guest is selected from dropdown list..
Output: Registration form for guest is displayed .

R4: Reset Password using mail
Input: Reset Password using mail navigation header is clicked.
Output: Page for reset password is shown where user need to provide Username.

R4.1: Change Password
Input: Username and email is given and send button is clicked.
Output: Mail is sent to user for new password.

R5: Status
Input: Status navigation header is clicked.
Output: Page is shown Requesting username where he can see status of his process is shown.

4.2 FEATURES OF NEW SYSTEM

- Paperless way to create users based on approvals on different levels.
- Login with TLS handshaking.
- User password reset features with CSRF and session tokens.
- Single portal for user/admin for different managements.
- Auditing of all tasks executed by approvers/admins/users.
- On click assignment of roles for accessing different services.
- Creation/management of users with groups/projects with the group owners.
- Dynamic workflow for expressing user "Status".
- RBAC based portal for management with Captcha login.
- Dynamic user search with limited user details.
- Addition/deletion/updation of owners/members of group.

4.3 NAVIGATION CHART

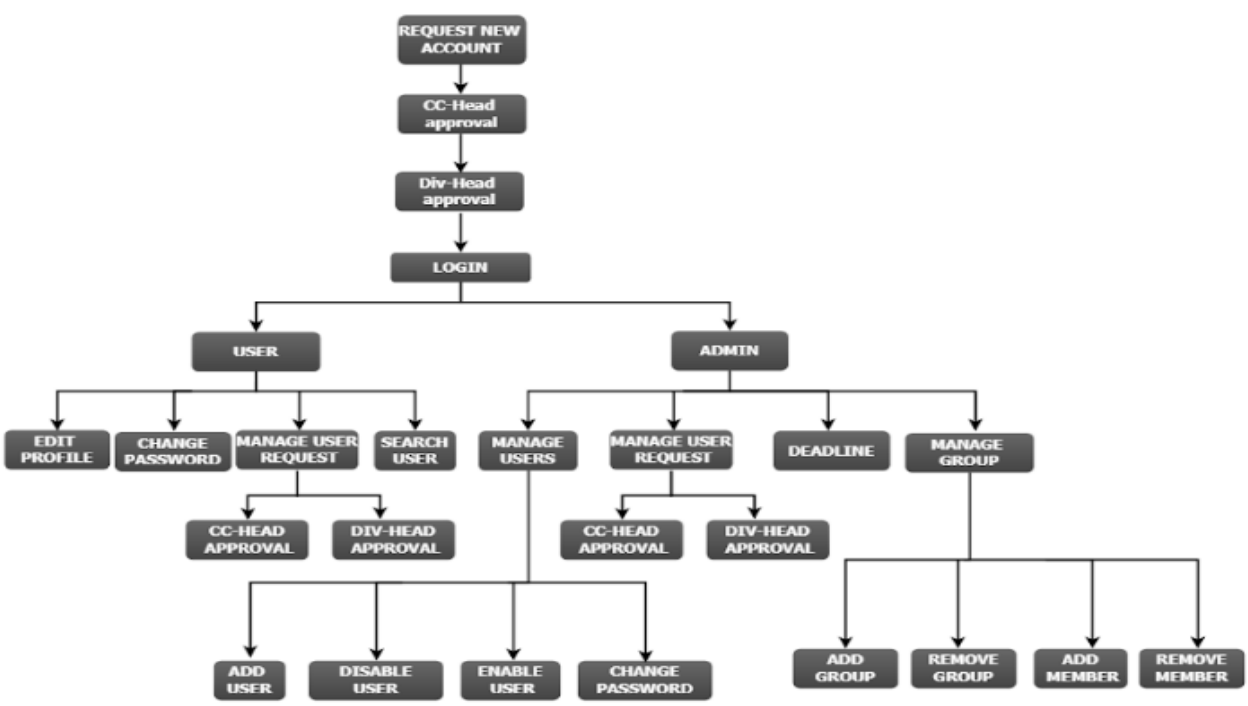


Fig 4.1 Navigation Chart

4.4 SYSTEM ACTIVITY

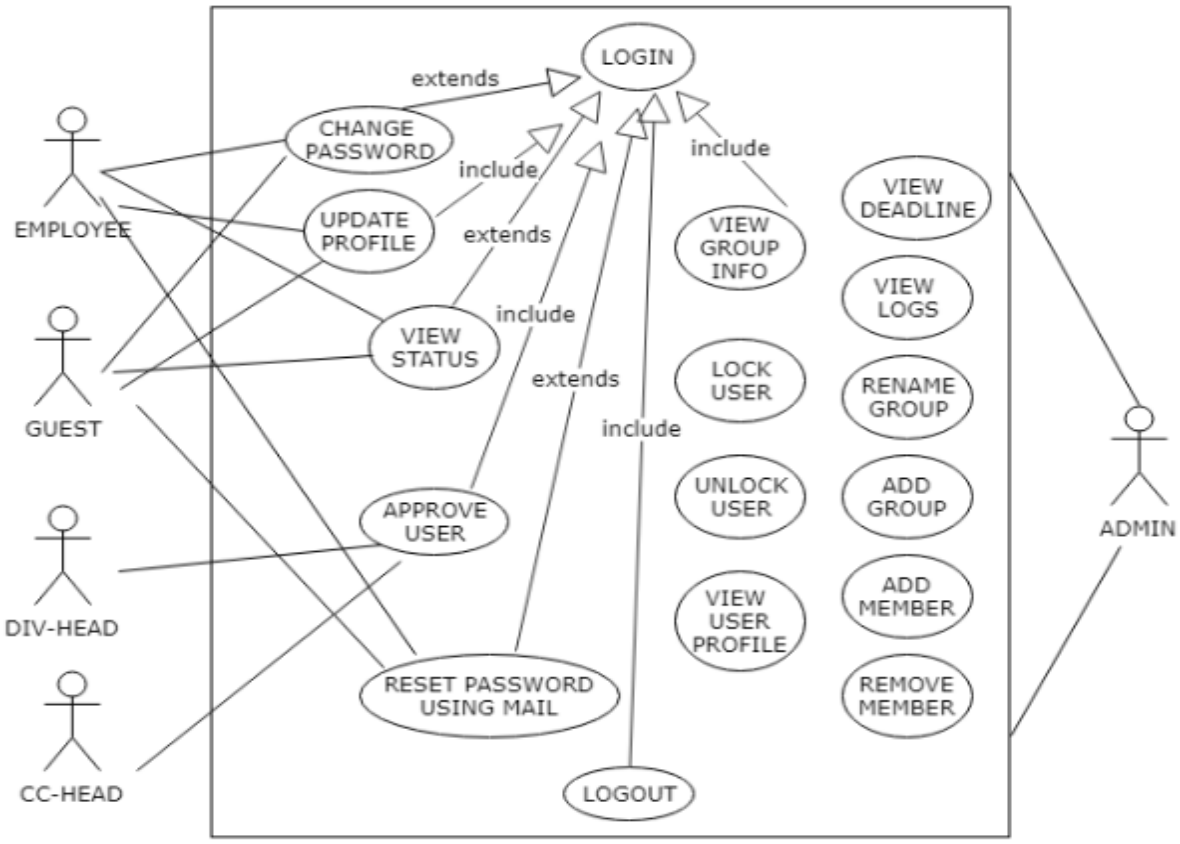


Fig 4.2 Use-Case

4.5 SEQUENCE DIAGRAM

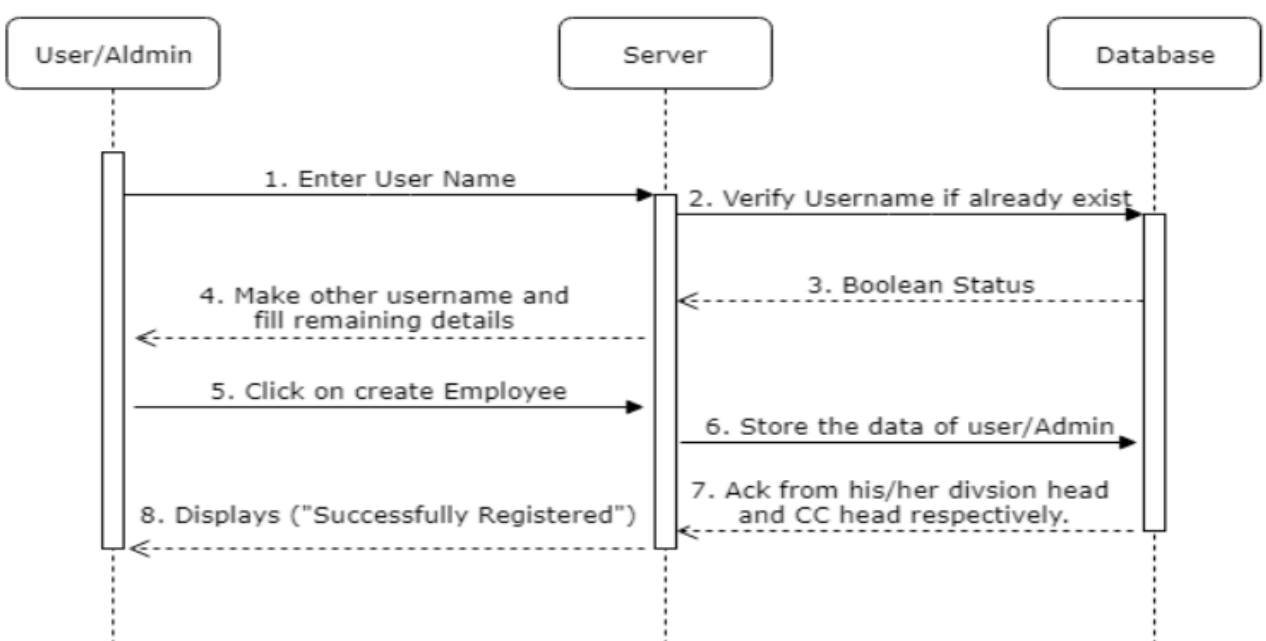


Fig 4.3 Create User/Admin (Sequence Diagram)

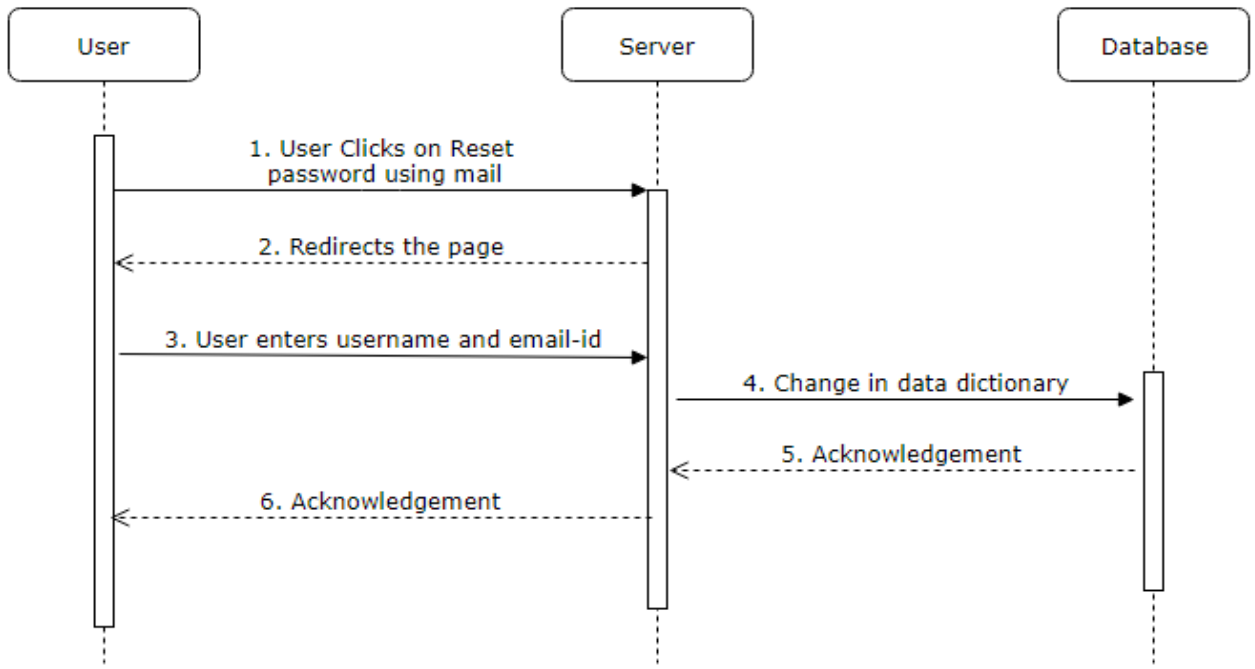


Fig 4.4 Reset Password Using mail(Sequence Diagram)

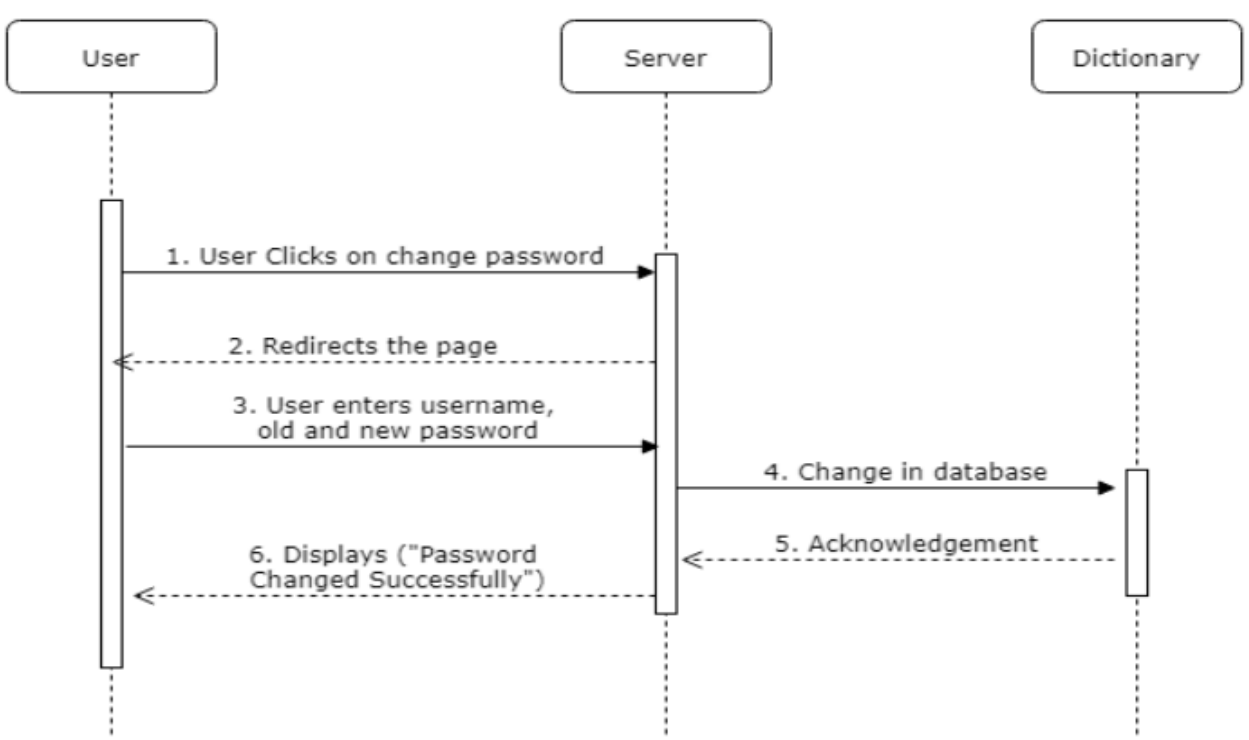


Fig 4.5 Change Password(Sequence Diagram)

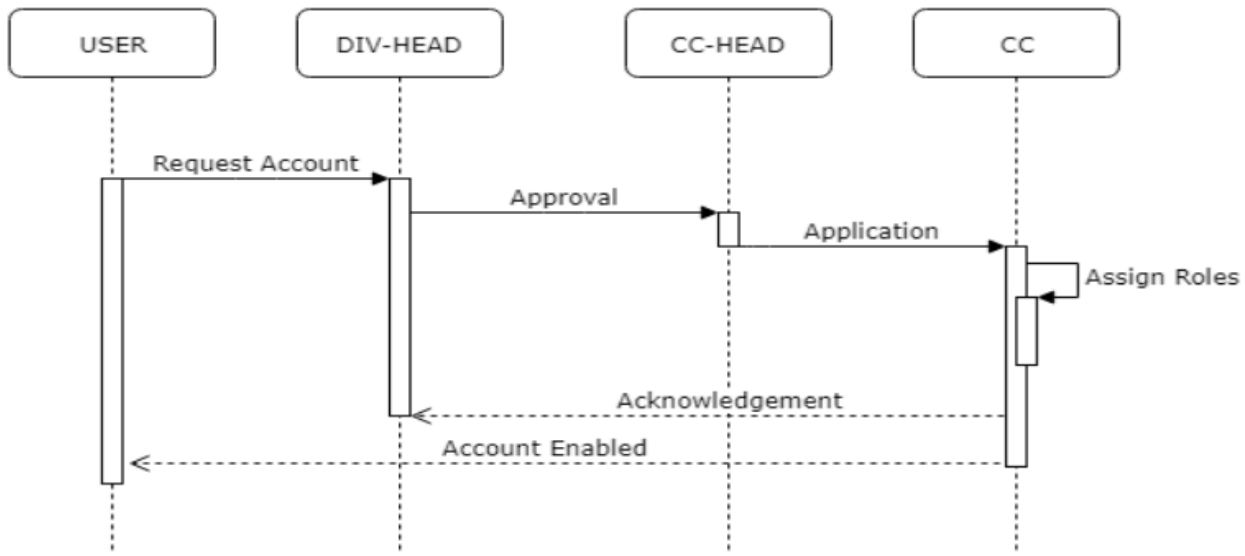


Fig 4.6 Enable User(Sequence Diagram)

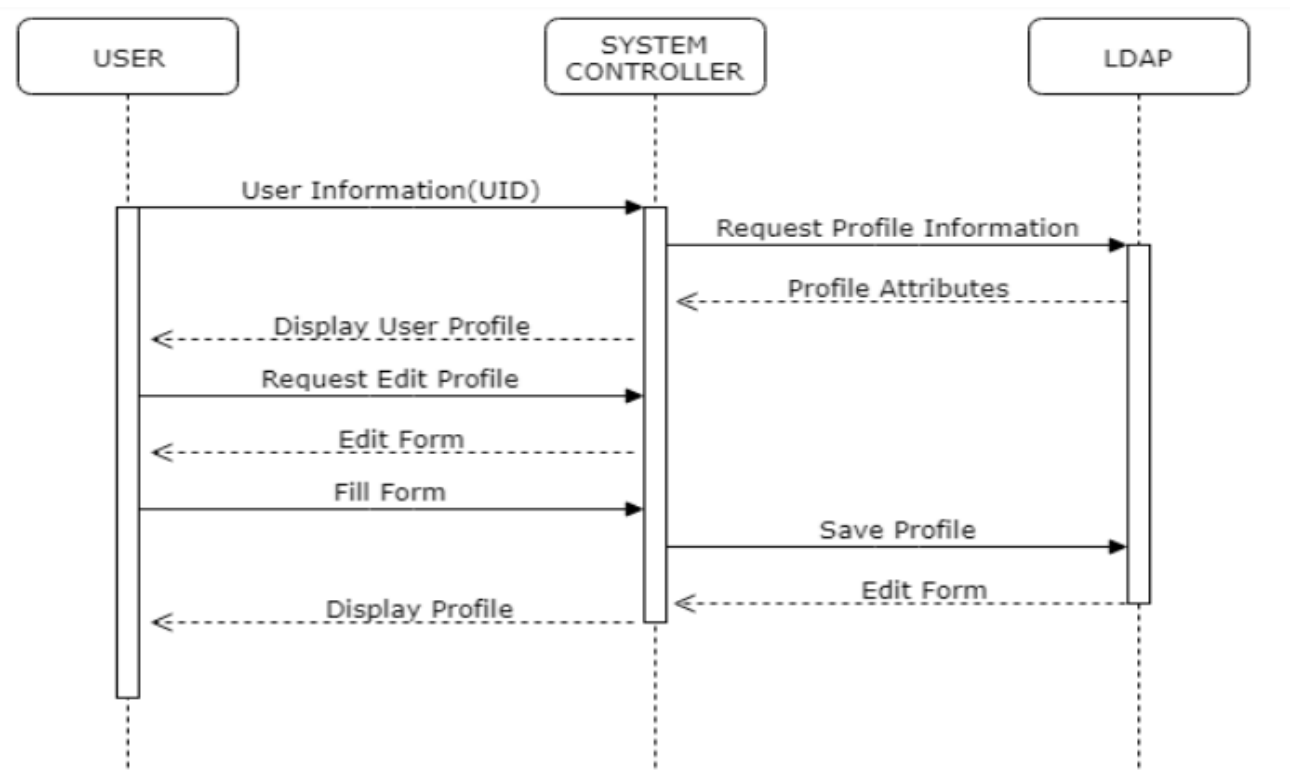


Fig 4.7 Display and Edit Profile(Sequence Diagram)

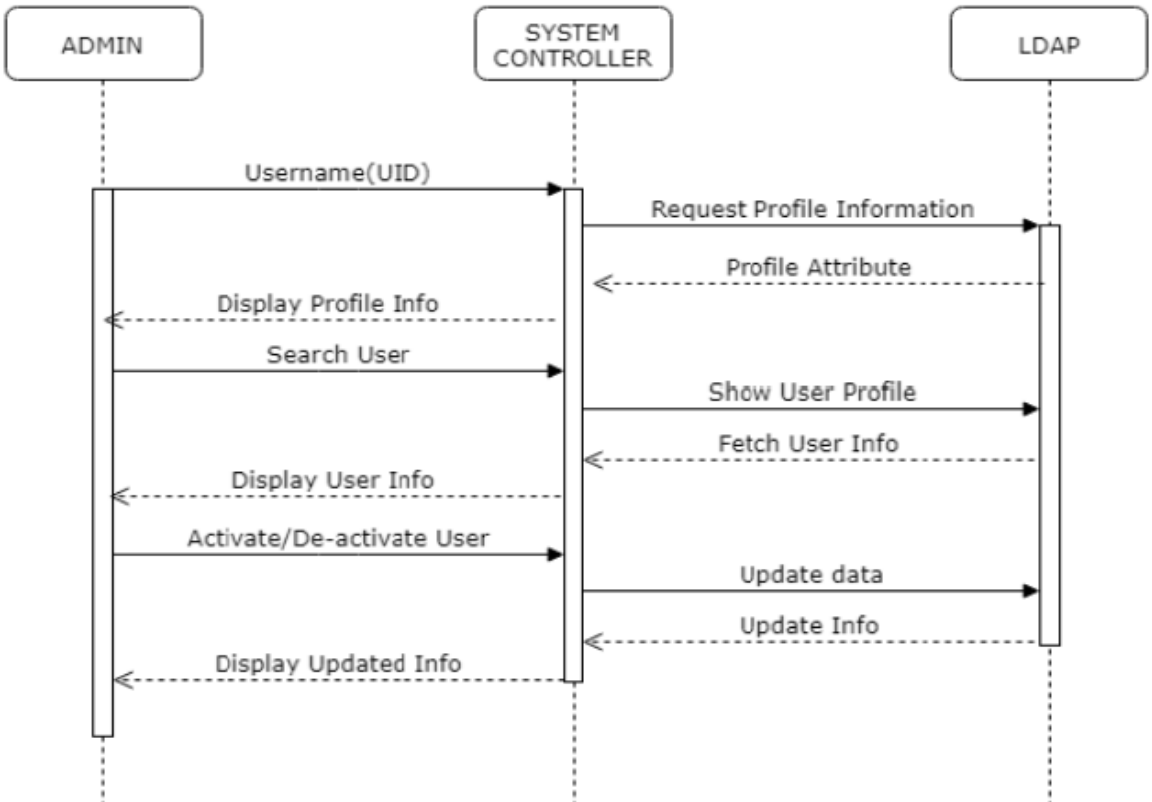


Fig 4.8 Activate/De-activate User(Sequence Diagram)

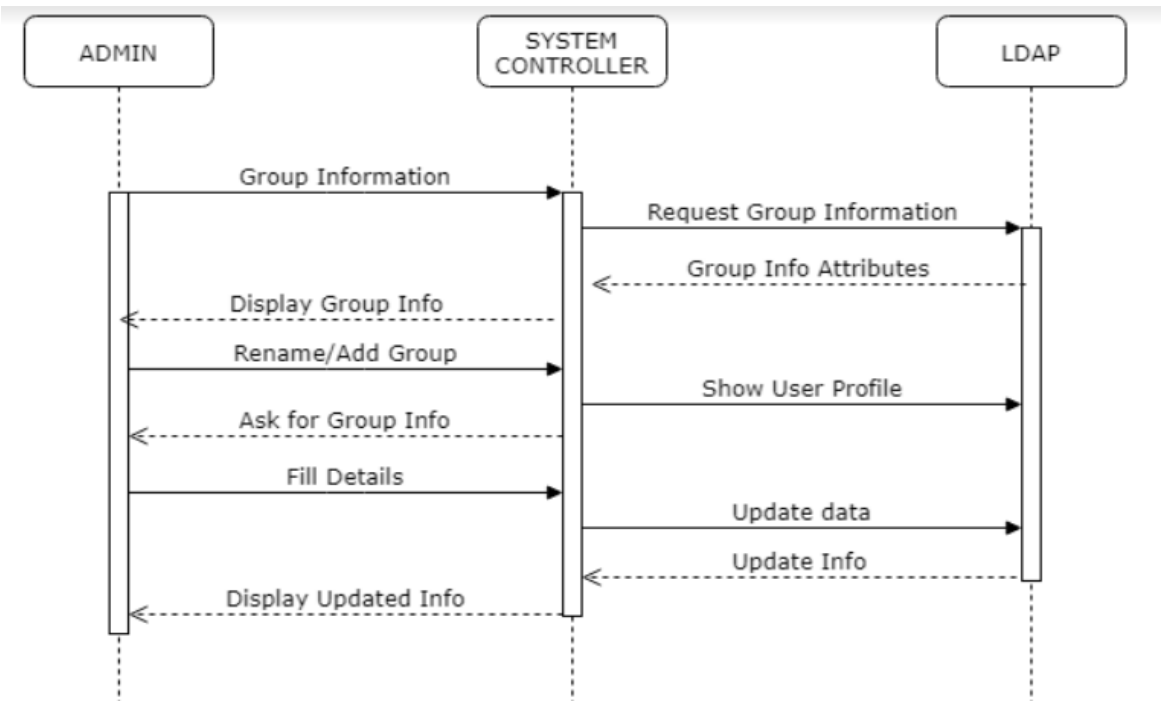


Fig 4.9 Add/Rename Group(Sequence Diagram)

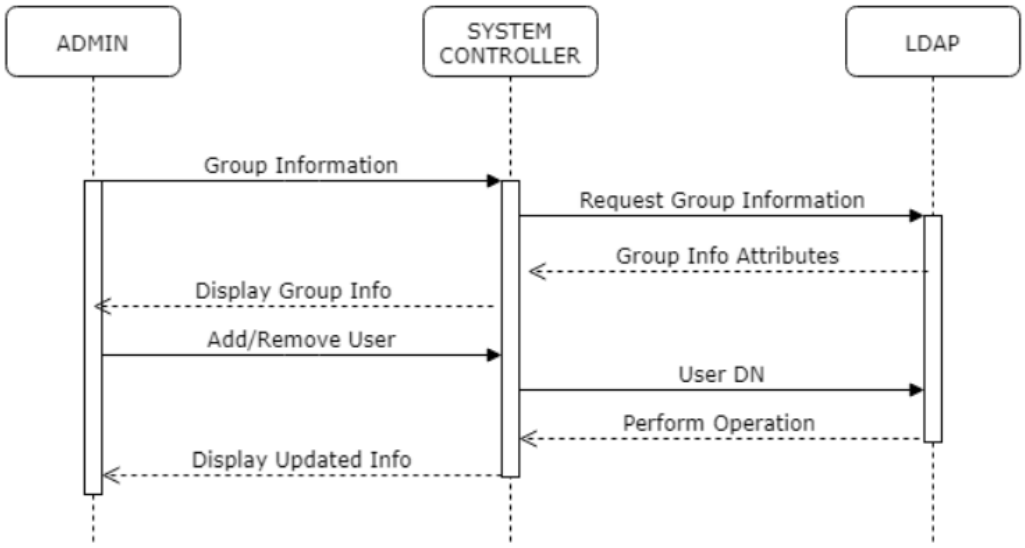


Fig 4.10 Add/Remove Member(Sequence Diagram)

4.6 TRANSITION DIAGRAM

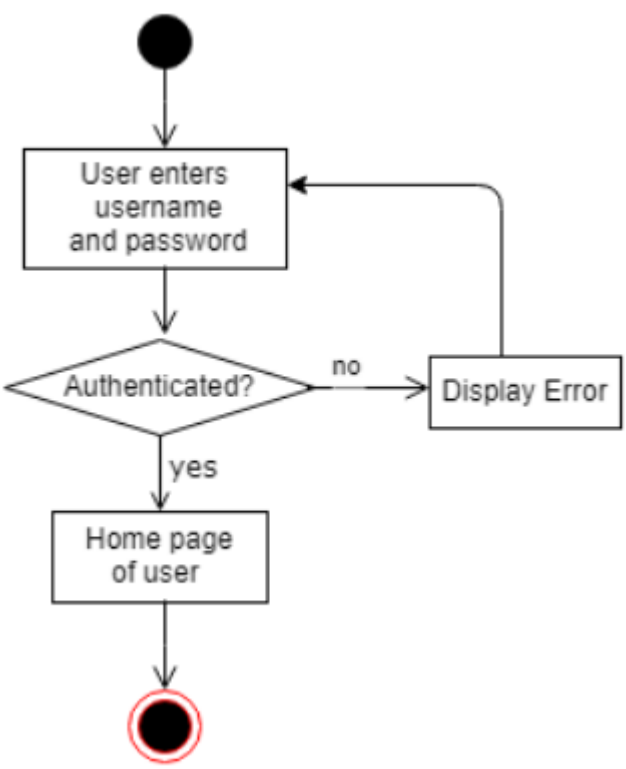


Fig 4.11 Login(Activity Diagram)

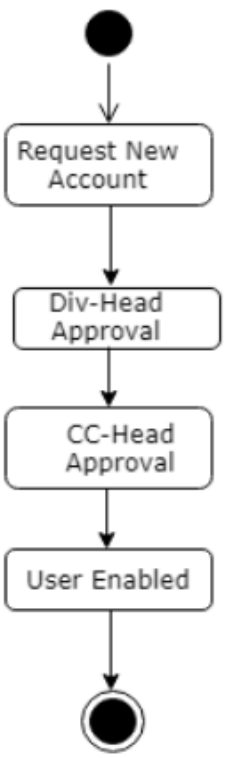


Fig 4.12 Enable User(Activity Diagram)

ADMIN:

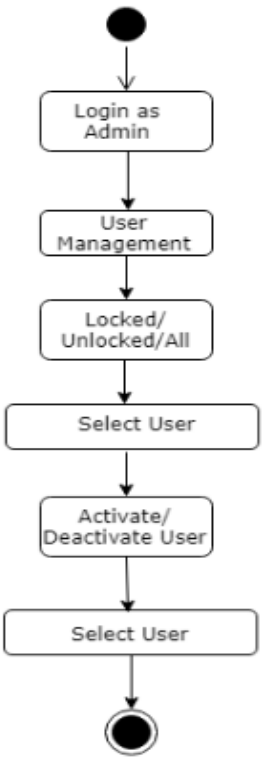


Fig 4.13 Activate/Deactivate User

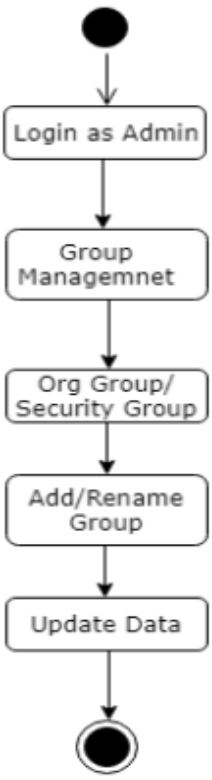


Fig 4.14 Add/Rename Group

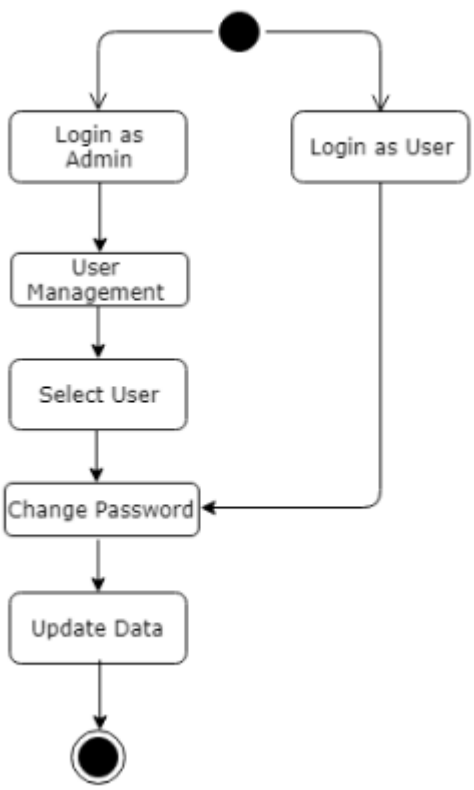


Fig 4.15 Change Password(Activity Diagram)

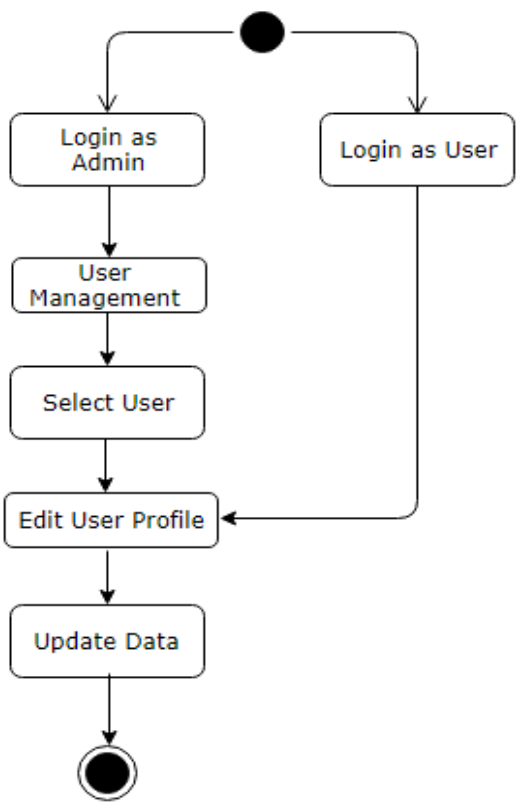


Fig 4.16 Edit User (Activity Diagram)

5. SYSTEM DESIGN

The Authentication system is built with rich GUI which provides the user a complete view of the entire topology to make changes in the system. Each operational component will consist of various GUI components, such as buttons, labels, text fields, etc. These components will be arranged in such a way that the user will be able to quickly grasp the purpose of each menu and perform whatever task it is designed for efficiency.

5.1 SYSTEM ARCHITECTURE DESIGN

5.1.1 Component Diagram

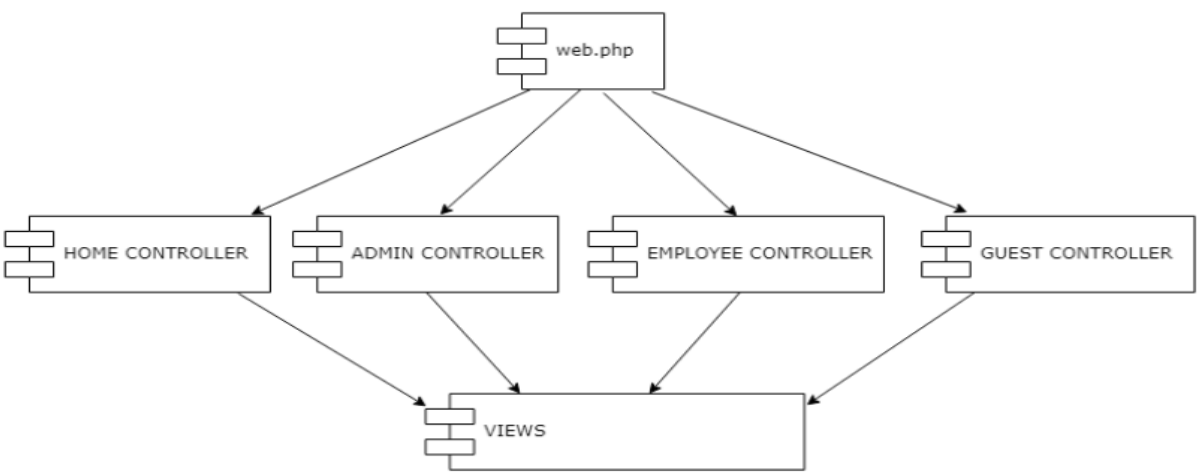


Fig 5.1 Component Diagram

5.1.2. Deployment Diagram

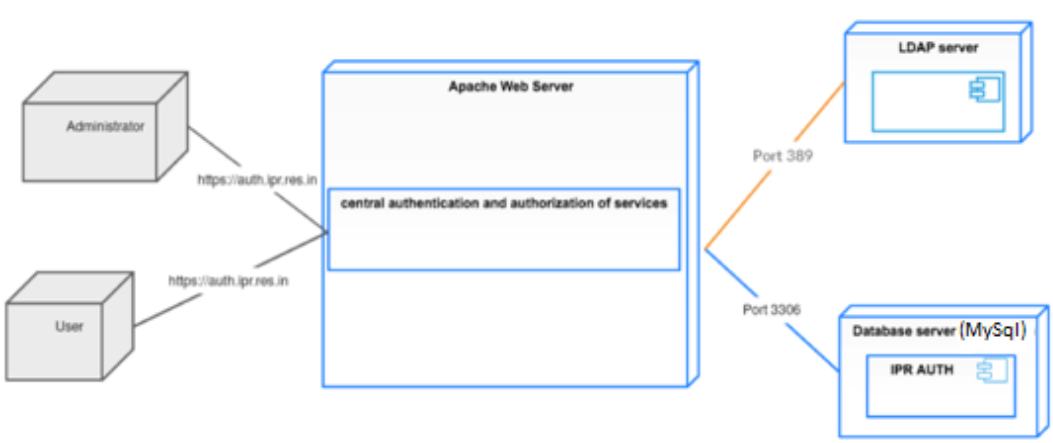


Fig 5.2 Deployment Diagram

5.2 DATABASE DESIGN

Table 5.1 Data Dictionary

USER	GROUPS
Display Name	entryDN
Given Name*	hasSubOrdinates
Business Category	Member
employeeNumber	Owner*
employeeType	Cn*
Uid*	
EmailAddress	
Aadharnumber	
Bankaccountnumber	
middleName	
Cchead	
Cn*	
Dateofcomplition	
Datefjoining	
departmentNumber	
displayName	
Divhead	
entryDN	
Gender*	
Mobile*	
Photo	

*=Required

6. IMPLEMENTATION PLANNING

The development phase involves converting design specifications into executable programs. Effective development standards include requirements that programmers and other project participants discuss design specifications before programming begins. The procedures help ensure programmers clearly understand program designs and functional requirements. Programmers use various techniques to develop computer programs.

6.1 IMPLEMENTATION ENVIRONMENT

The implementation environment used is Laravel Framework. Data is stored in data dictionary using develop.ipr.res.in server. One can retrieve and update data in database. System is implemented in PHP

6.2 PROGRAM/MODULES SPECIFICATION

The main modules of CAAS System are.
1. Admin module.
2. User module.

6.3 CODING STANDARDS

- Exceptions are properly handled and write within try-catch only.
- Comments are also specified to inform the working of the function.
- Proper messages are displayed when error occurs.
- Functions are properly written in respective controllers.

7. TESTING

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. In fact, testing is the one step in the software engineering process that could be viewed as destructive rather than constructive. Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design, and coding. And it needs to be done in almost every phase of product development life cycle not just before a product is handled to a customer.

7.1 TESTING PLAN

The testing sub-process includes the following activities in a phase dependent manner: Create test plans.
a) Create test Specifications.
b) Review Test Plans and Test Specifications
c) Conduct tests according to the Test Specifications, and log the defects
d) Fix defects, if any.
e) When defects are fixed continue from activity.

7.2 TESTING STRATEGY

The development process repeats this testing sub-process a number of times for the following phases.

- Unit Testing.
- Integration Testing

Unit Testing tests a unit of code (module or program) after coding of that unit is completed. Integration Testing tests whether the various programs that make up a system, interface with each other as desired, fit together and whether the interfaces between the programs are correct. System Testing ensures that the system meets its stated design specifications. Acceptance Testing is testing by the users to ascertain whether the system developed is a correct implementation of the Software Requirements Specification.

Testing is carried out in such a hierarchical manner to ensure that each component is correct and the assembly/combination of components is correct. Merely testing a whole system at the end would most likely throw up errors in components that would be very costly to trace and fix. We have performed both Unit Testing and System Testing to detect and fix errors.

7.3 TESTING METHODS

7.3.1 White Box Testing

White box testing, sometimes called glass box testing is a test case design method that uses the control structure of the procedural design to derive test cases. Using white box testing methods, the software engineer can derive test cases that: Guarantee that all independent paths within a module have been exercised at least once. Exercise all logical decisions on their true and false sides.

7.3.2 Black Box Testing

It focuses on the functional requirements of the software. That is black box testing enables the software engineer to derive sets of input conditions that will fully exercise all functional requirements for a program. Black box testing is not an alternative to white box techniques. Rather it is a complementary approach that is likely to uncover a different class of errors than white box methods. Black box testing attempts to find errors in their following categories:

- Incorrect or missing functions.
- Interface errors.
- Errors in data structures or external data base access.
- Performance errors.
- Initialization and termination errors.

7.4 TEST CASES

NO	PURPOSE	INPUT	OUTPUT
1	Employee Login	Username, password, captctha	Home page of employee
2	Admin Login	Username, password, captctha	Home page of admin
3	Login	Incorrect username or password	Invalid Credentials
4	Login	Incorrect Captcha	Incorrect Captcha
5	Edit Profile	Required Data	Profile Updated
6	Change Password	Username, old and new password	Password changed successfully
8	Rename Group	New Group name	Group renamed
9	Add group member	Select member	Member added
10	Create new user	Valid data	Registration successful
11	Create new user	Invalid data	Registration unsuccessful
12	Check status	Invalid Username	Username is Incorrect
13	Disable User	Select user	User disabled
14	Enable User	Select User	User enabled
15	Ldap Connectivity	Close Server Connection	Can't connect LDAP server

ZAP SCANNING REPORT

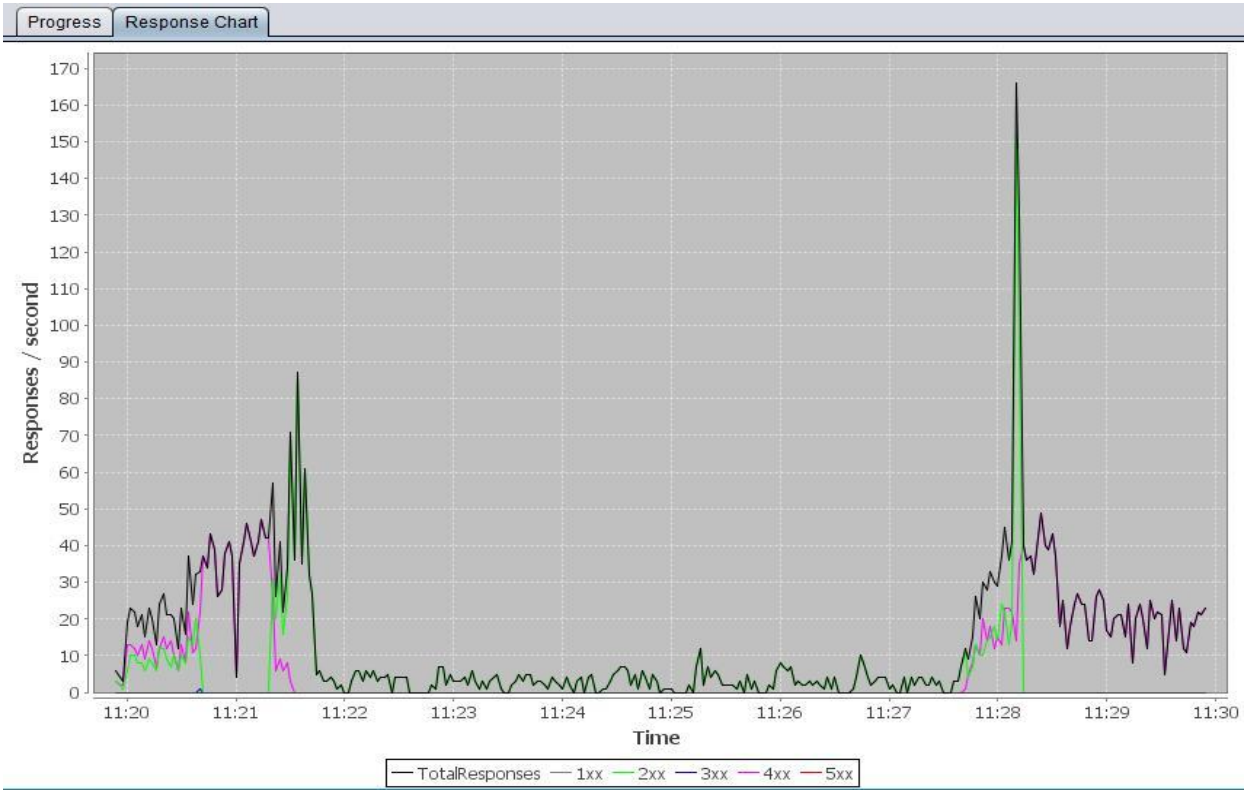
SUMMARY OF ALERTS

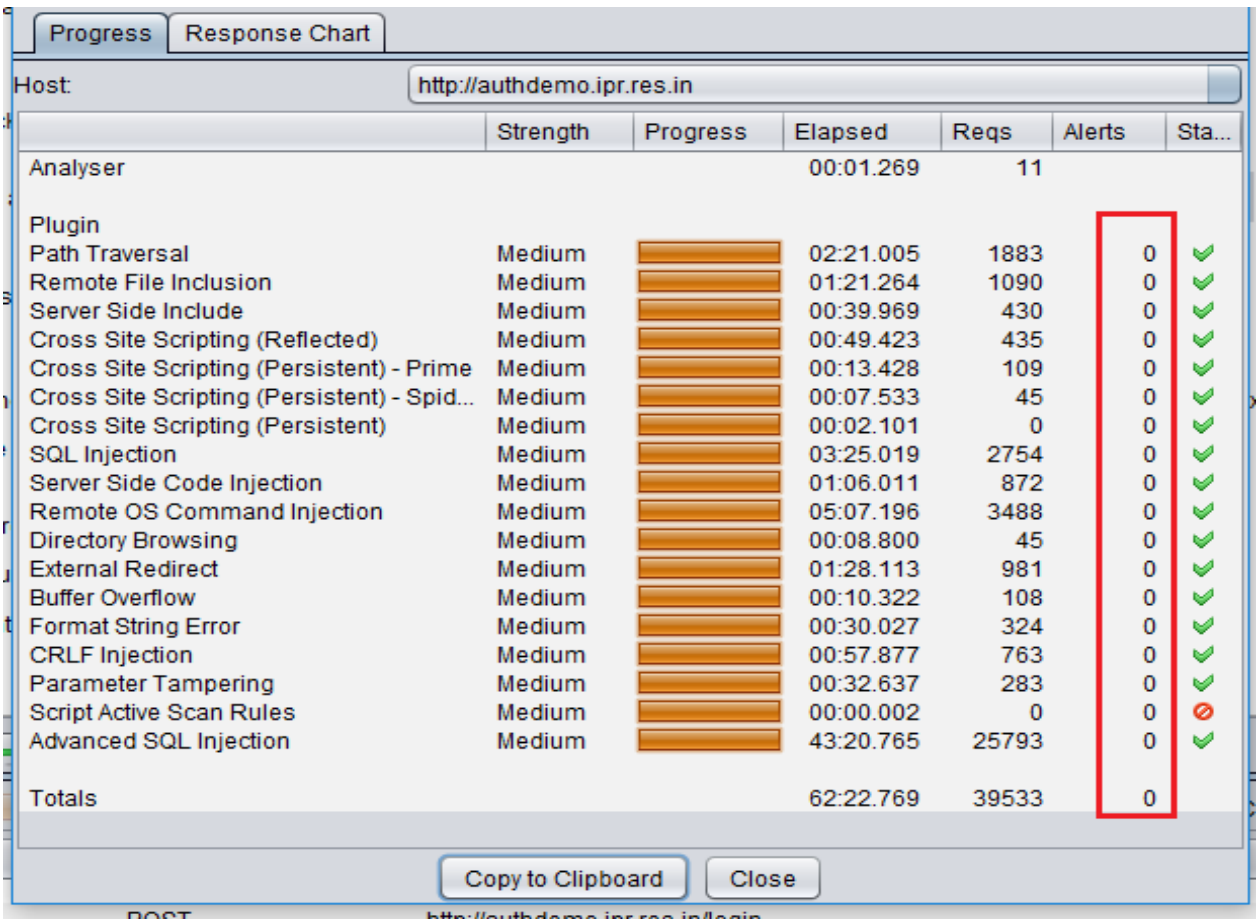
Risk Level	Number of Alerts
High	0
Medium	1
Low	2
Informational	0

ALERT DETAIL

Since the applicationis running only in IPR's intranet, disclosure of local IPs is not an major information disclosed.

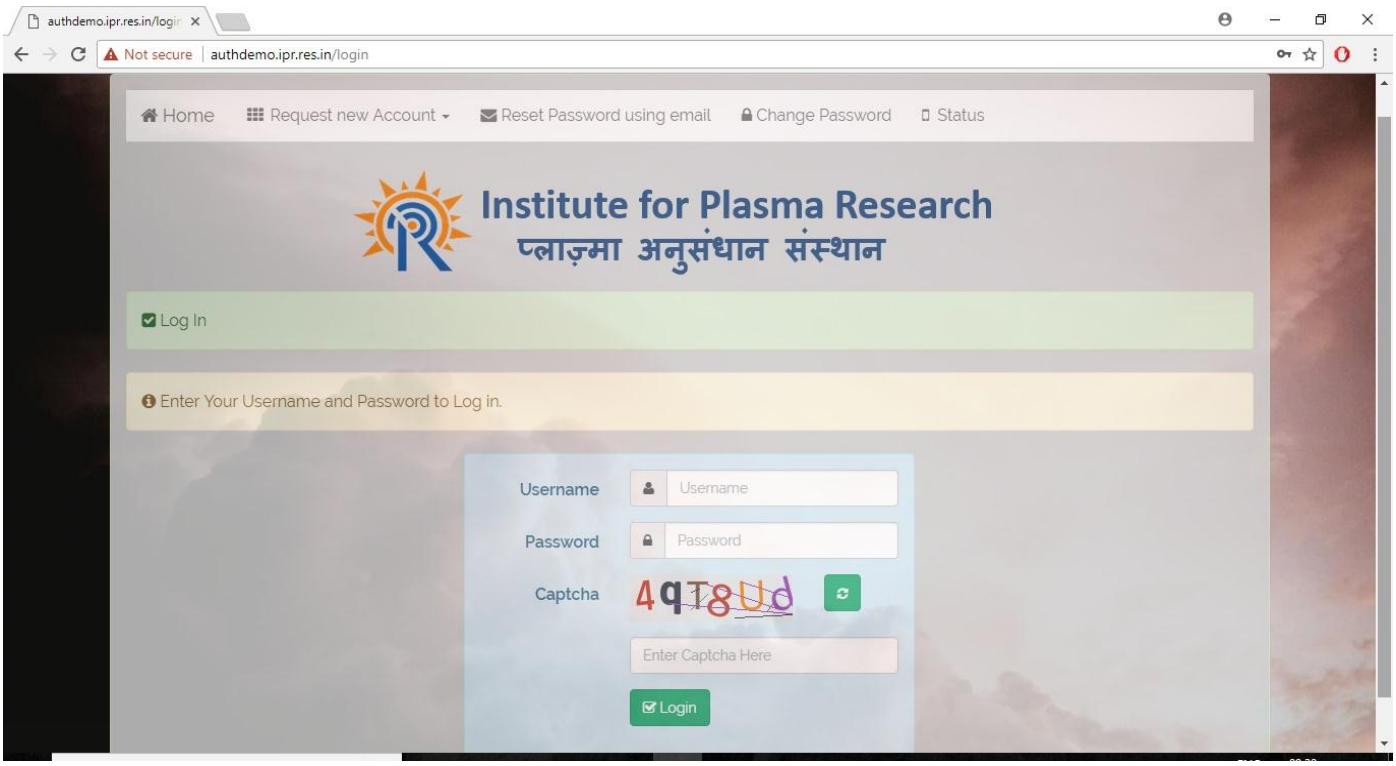
Low (Medium)	**Private IP Disclosure**
Description	A private IP (such as 10.x.x.x, 172.x.x.x, 192.168.x.x) or an Amazon EC2 private hostname (for example, ip-10-0-56-78) has been found in the HTTP response body. This information might be helpful for further attacks targeting internal systems.
URL	http://authdemo.ipr.res.in/index/SentResetPasswordLink
Method	GET
Evidence	10.20.4.12
URL	http://authdemo.ipr.res.in/index/statusView
Method	GET
Evidence	10.20.4.12
URL	http://authdemo.ipr.res.in/index/passwordchange
Method	GET
Evidence	10.20.4.12
Instances	3
Solution	Remove the private IP address from the HTTP response body. For comments, use JSP/ASP/PHP comment instead of HTML/JavaScript comment which can be seen by client browsers.
Other information	10.20.4.12, 10.20.4.25
Reference	https://tools.ietf.org/html/rfc1918
CWE Id	200
WASC Id	13
Source ID	3



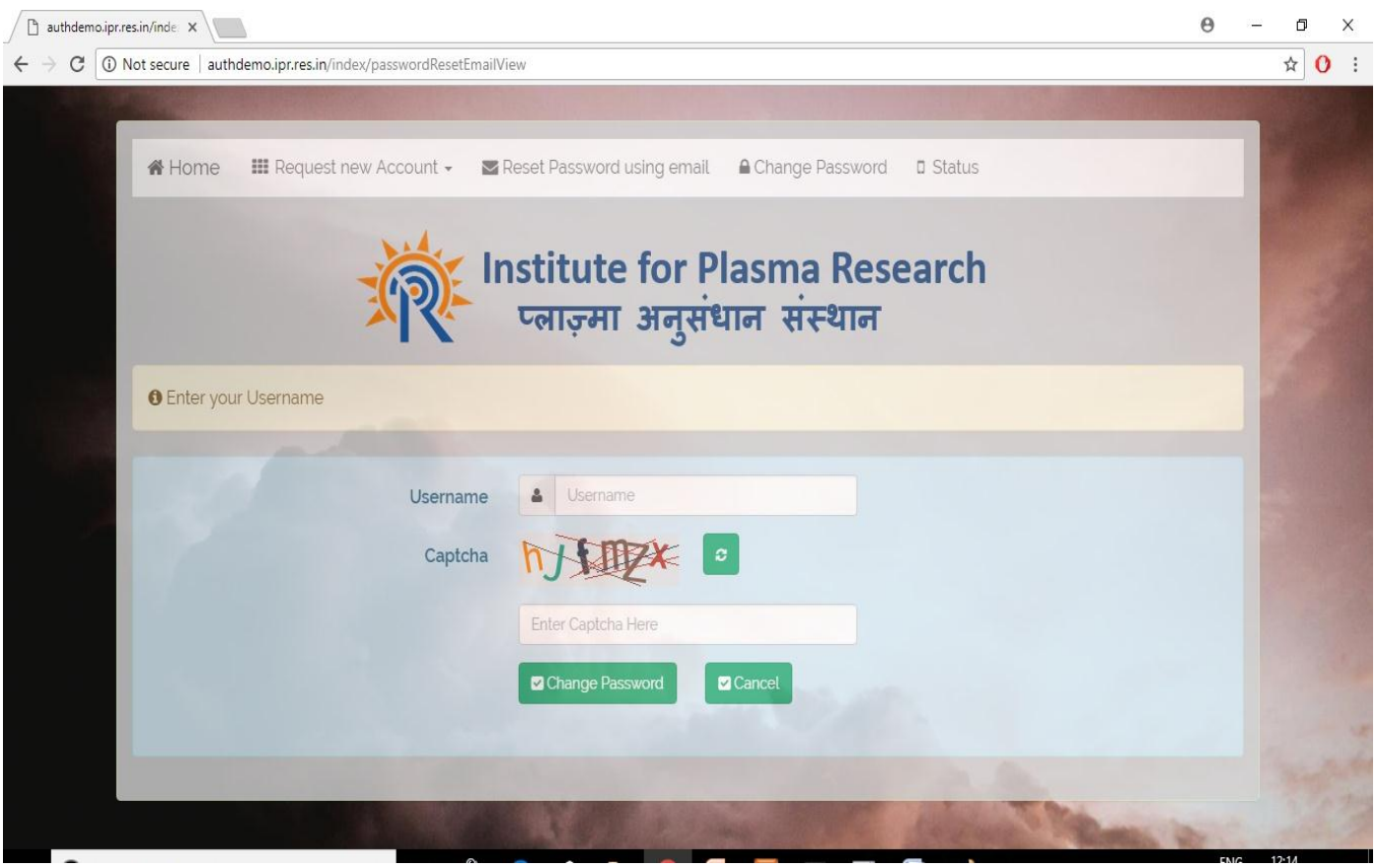


	Strength	Progress	Elapsed	Reqs	Alerts	Sta...
Analyser			00:01.269	11		
Plugin						
Path Traversal	Medium		02:21.005	1883	0	
Remote File Inclusion	Medium		01:21.264	1090	0	
Server Side Include	Medium		00:39.969	430	0	
Cross Site Scripting (Reflected)	Medium		00:49.423	435	0	
Cross Site Scripting (Persistent) - Prime	Medium		00:13.428	109	0	
Cross Site Scripting (Persistent) - Spid...	Medium		00:07.533	45	0	
Cross Site Scripting (Persistent)	Medium		00:02.101	0	0	
SQL Injection	Medium		03:25.019	2754	0	
Server Side Code Injection	Medium		01:06.011	872	0	
Remote OS Command Injection	Medium		05:07.196	3488	0	
Directory Browsing	Medium		00:08.800	45	0	
External Redirect	Medium		01:28.113	981	0	
Buffer Overflow	Medium		00:10.322	108	0	
Format String Error	Medium		00:30.027	324	0	
CRLF Injection	Medium		00:57.877	763	0	
Parameter Tampering	Medium		00:32.637	283	0	
Script Active Scan Rules	Medium		00:00.002	0	0	
Advanced SQL Injection	Medium		43:20.765	25793	0	
Totals			62:22.769	39533	0	

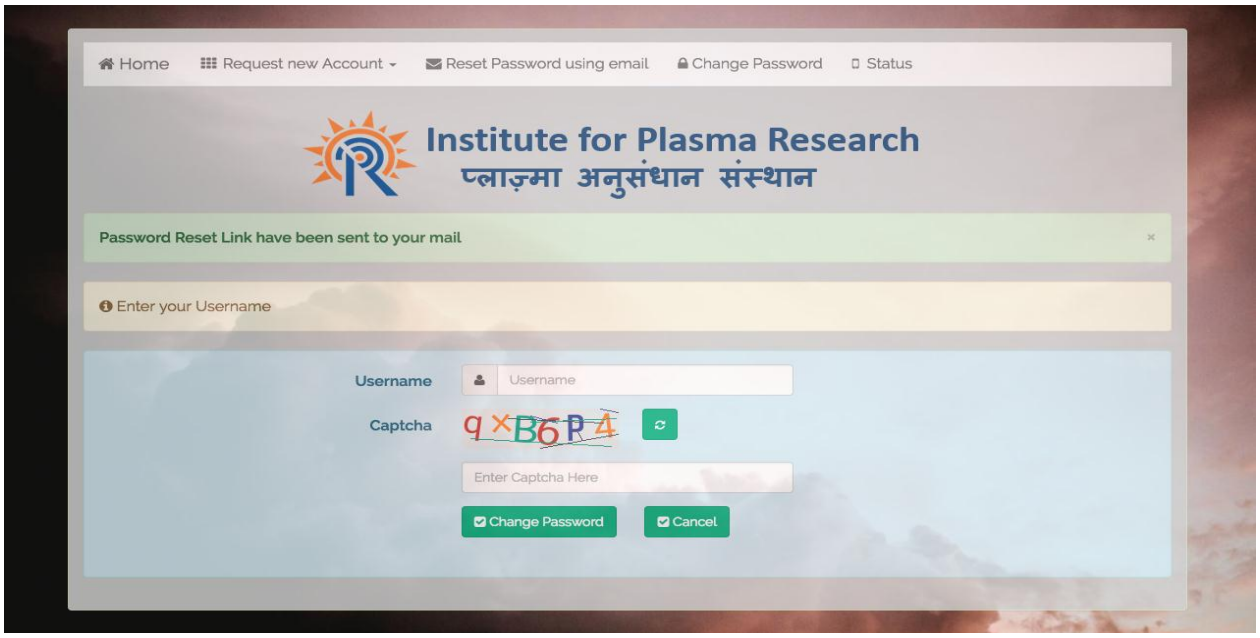
8. USER MANUALS



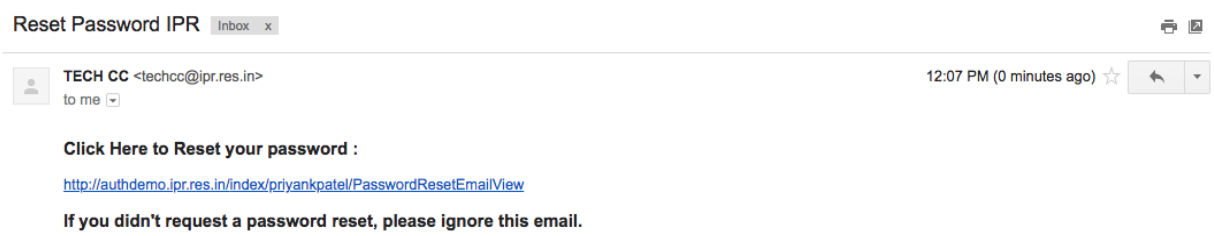
8.1 Login Page



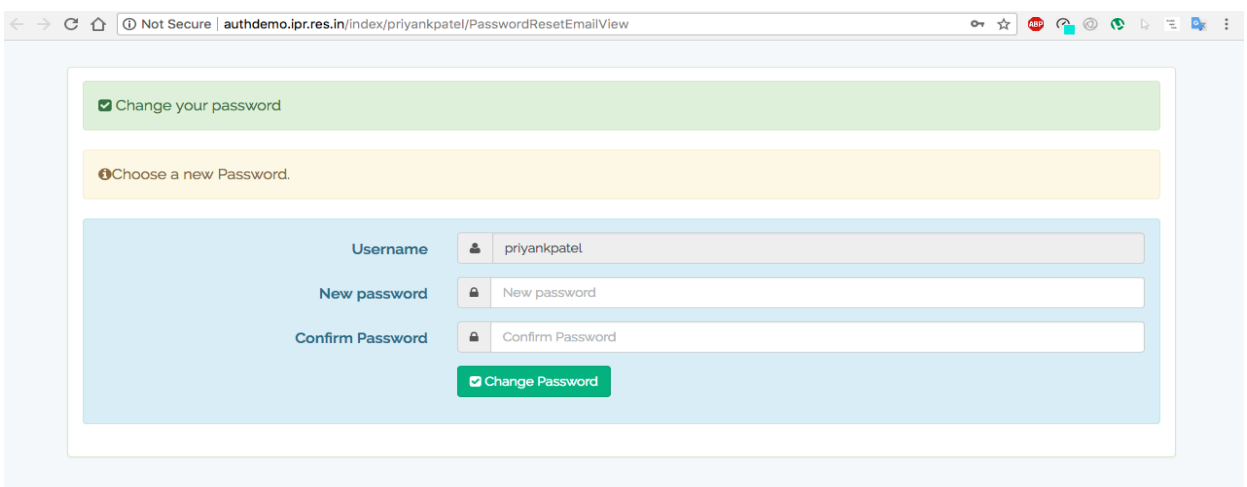
8.2(a) Reset password using mail



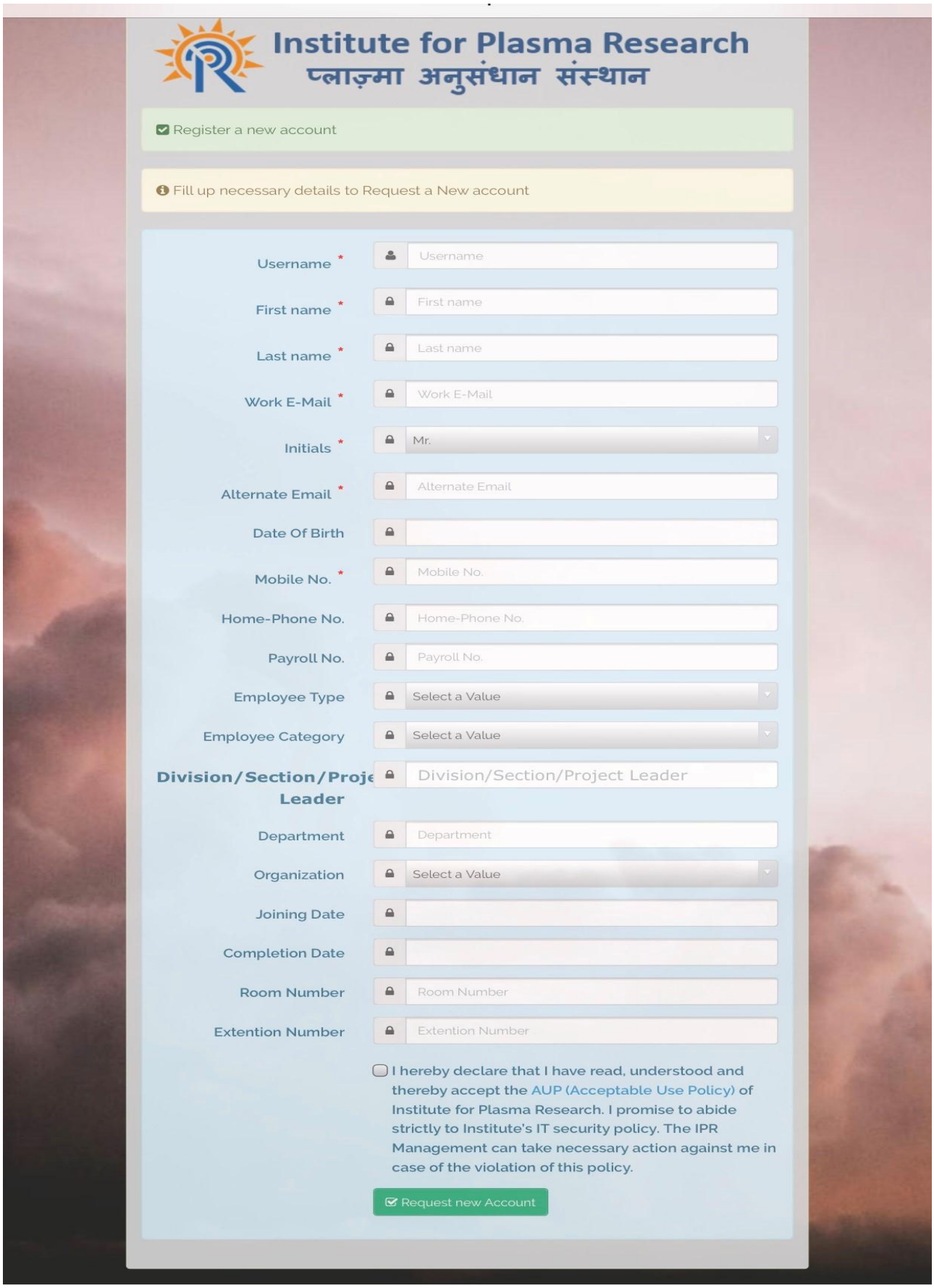
8.2(b) Password Link Sent



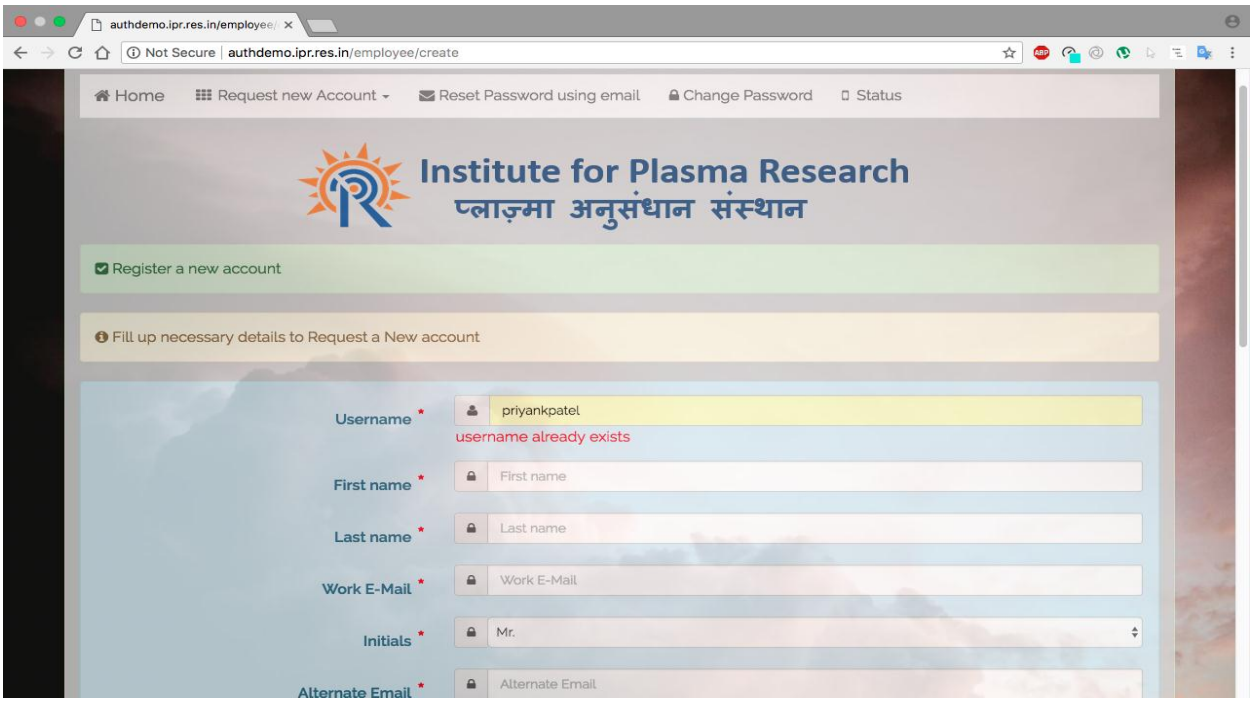
8.2 (c) Reset password Link



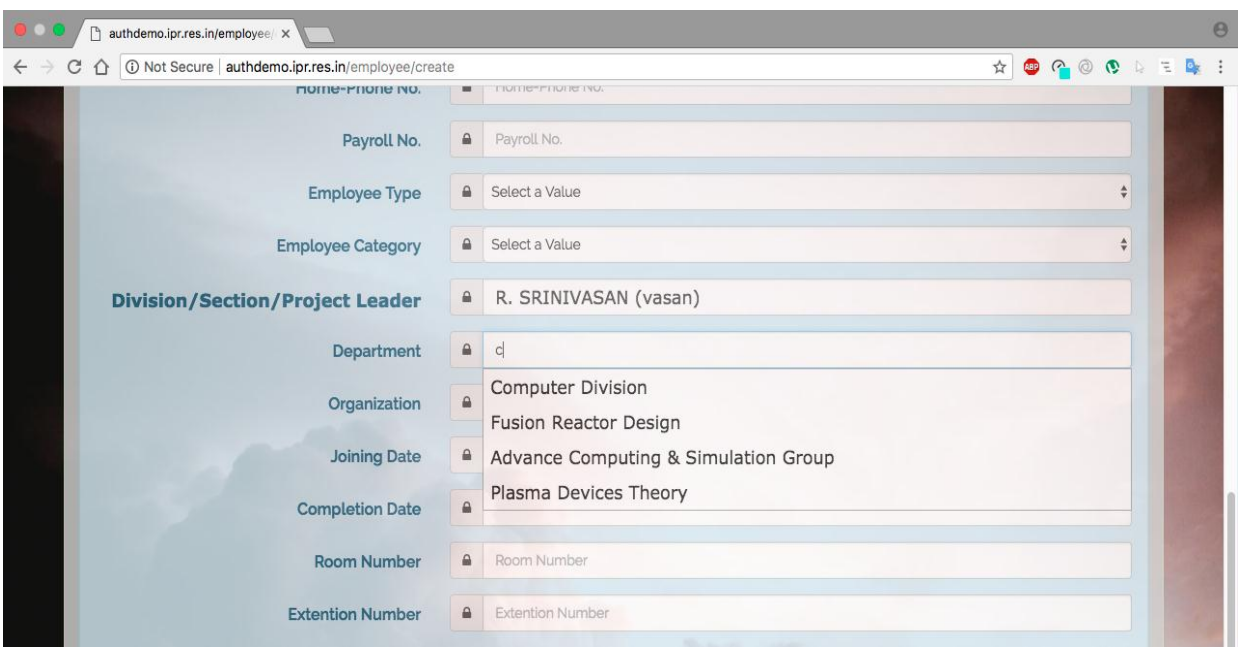
8.2(d) Password Link



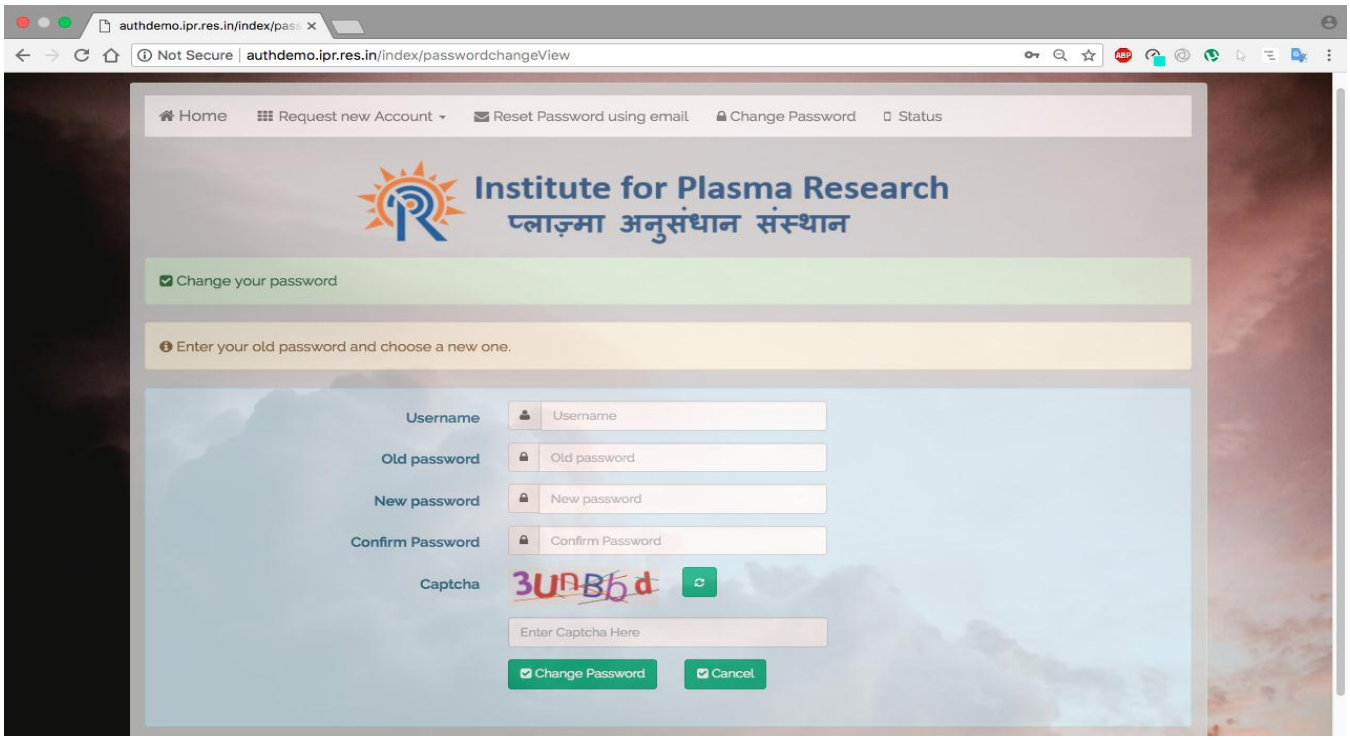
8.3(a) Create Employee



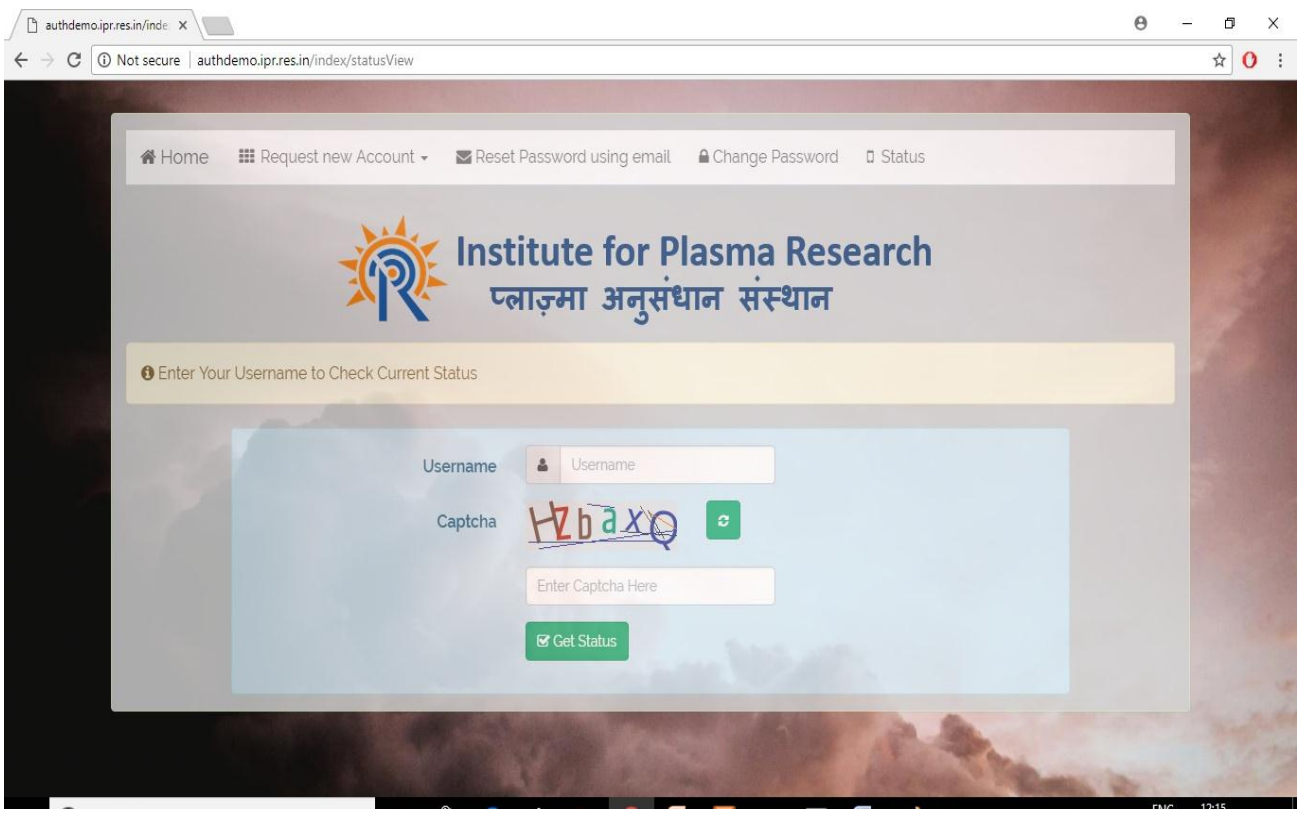
8.3(b) Create Employee (Username already exists)



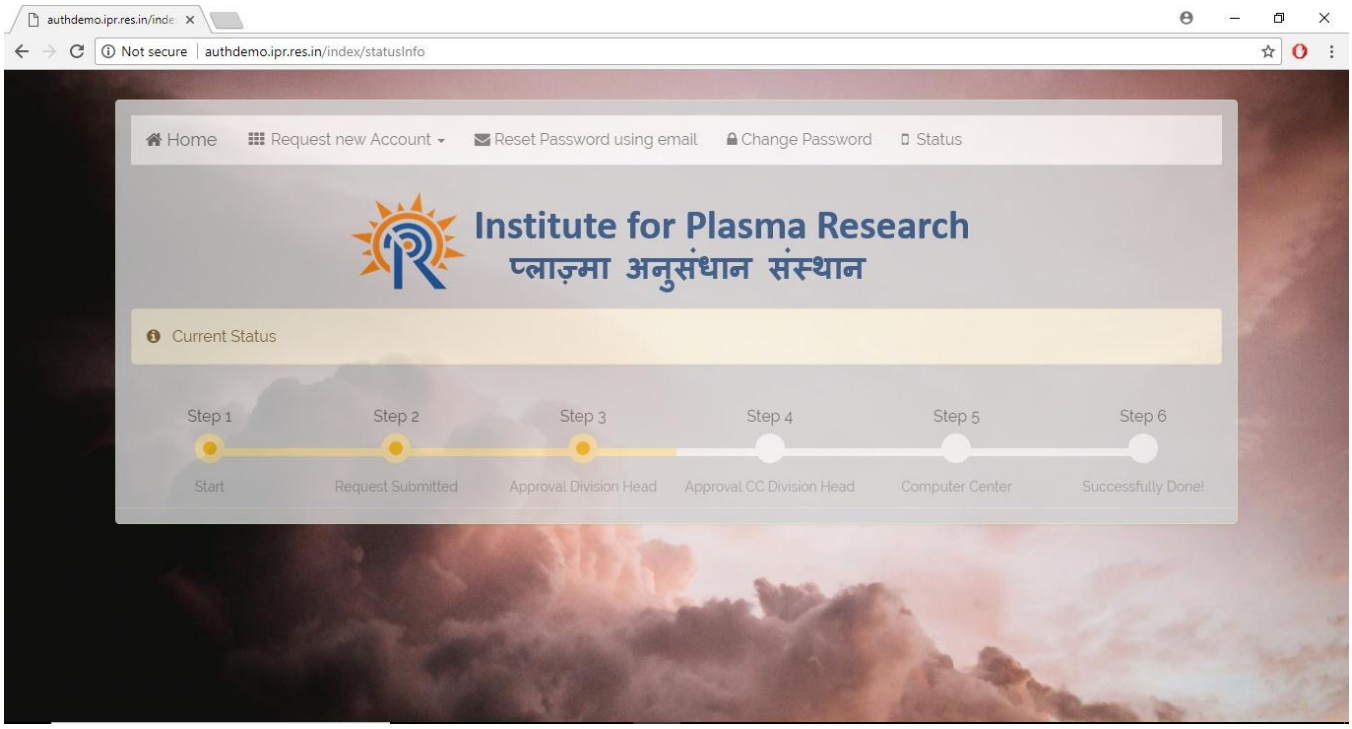
8.3 (c) Create Employee (Division Head with Department)



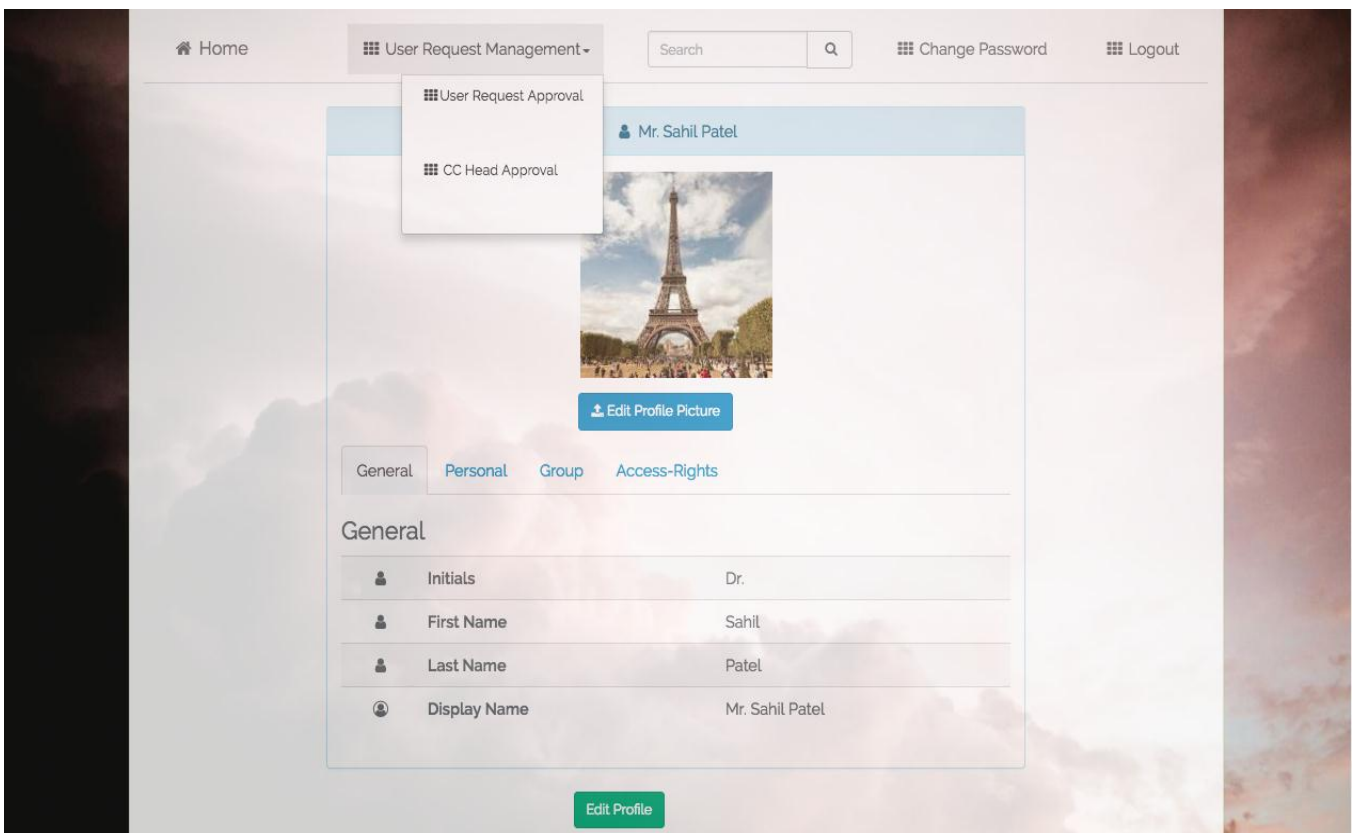
8.4 Change Password



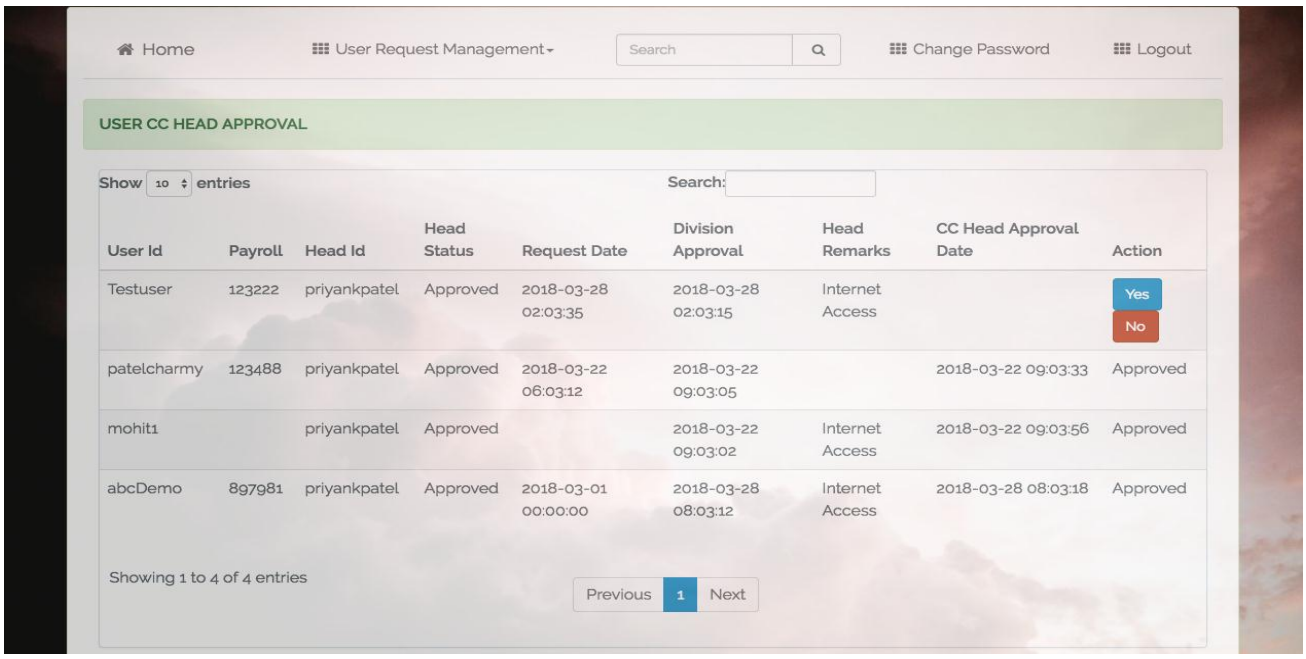
8.5(a) Status



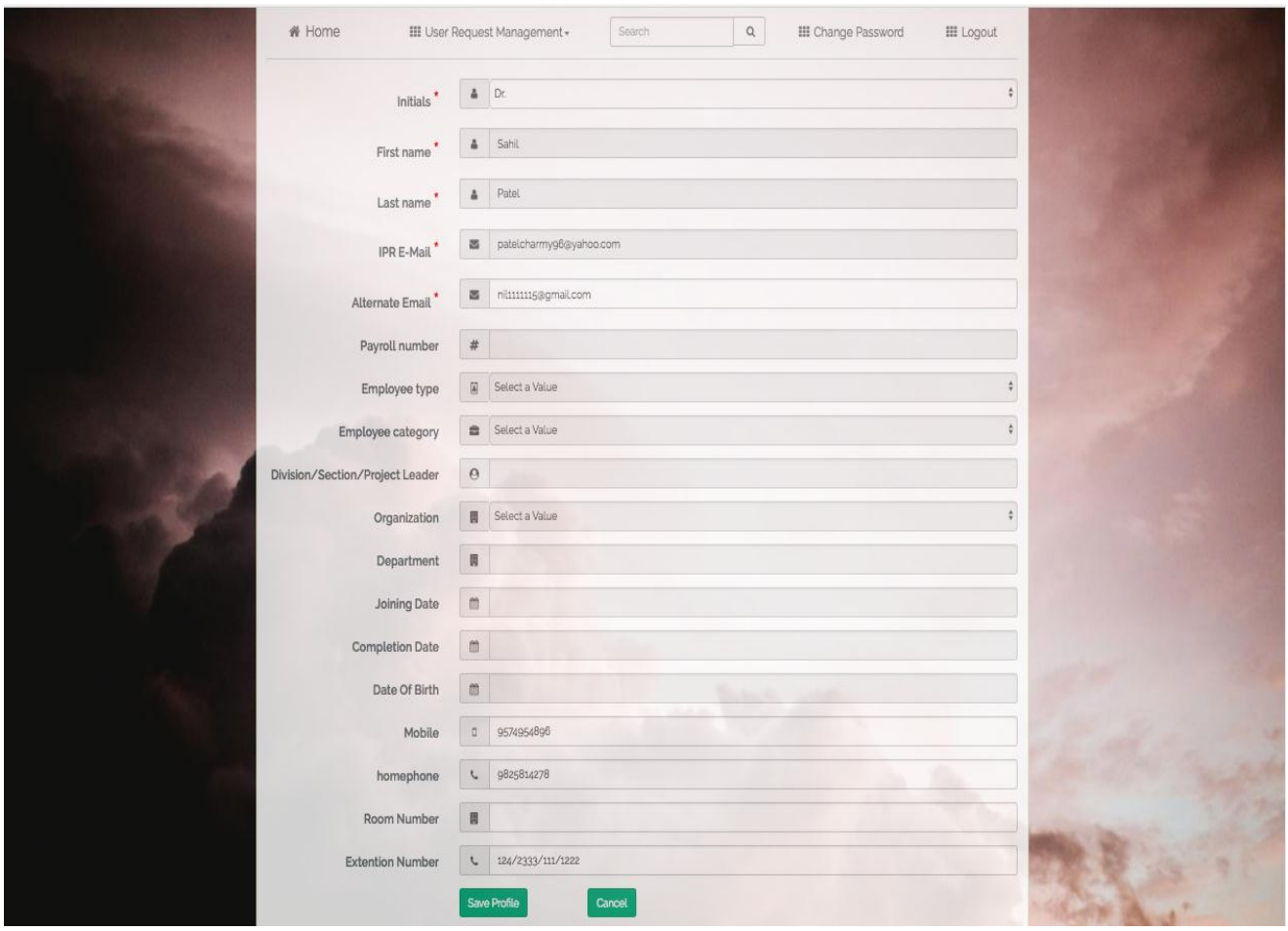
8.5(b) Status



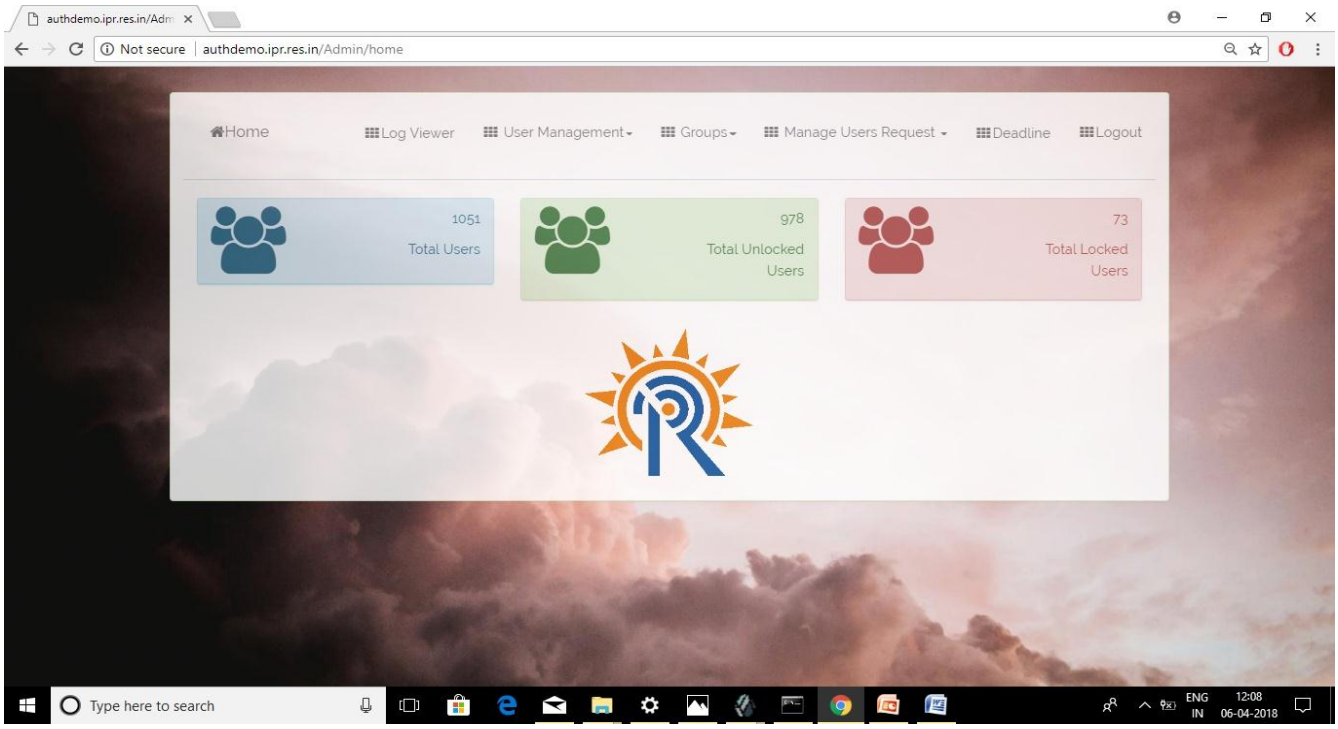
8.6 User Login



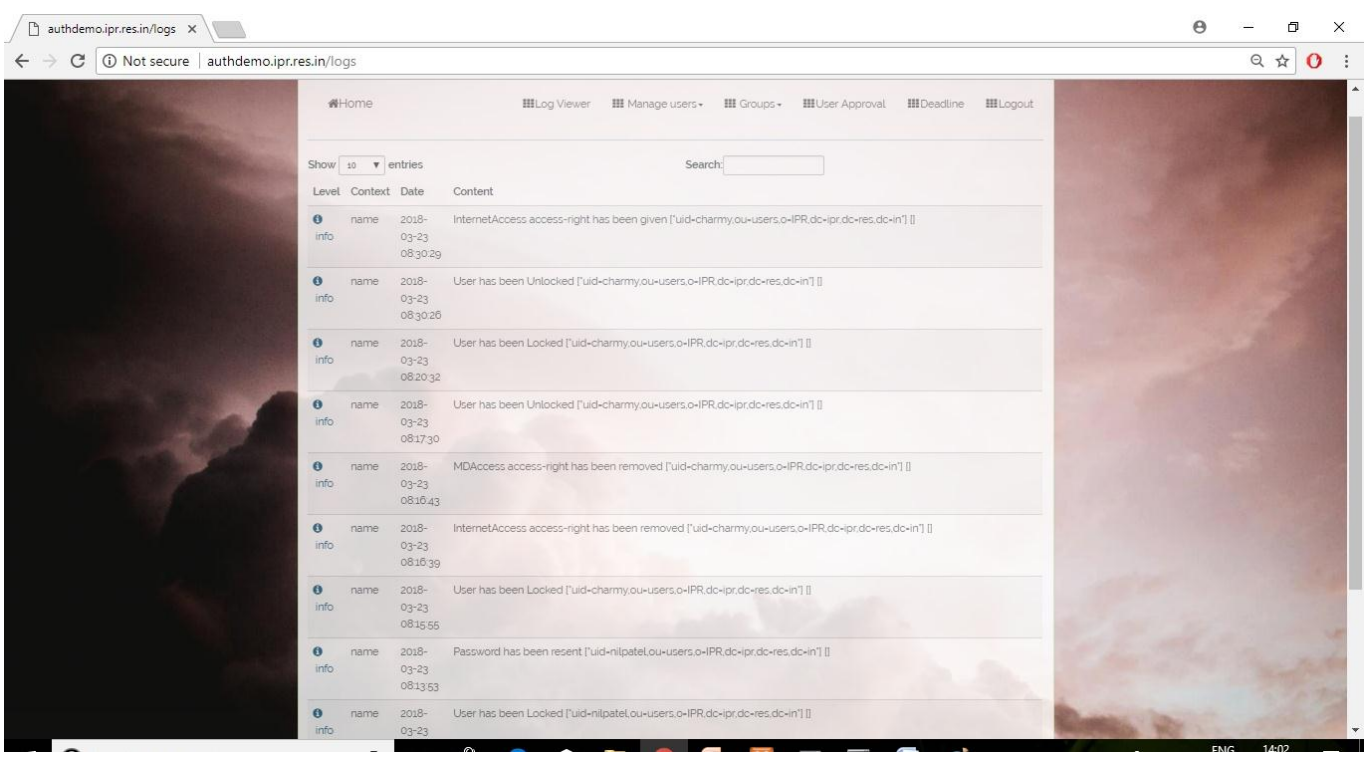
8.7 Request Approval



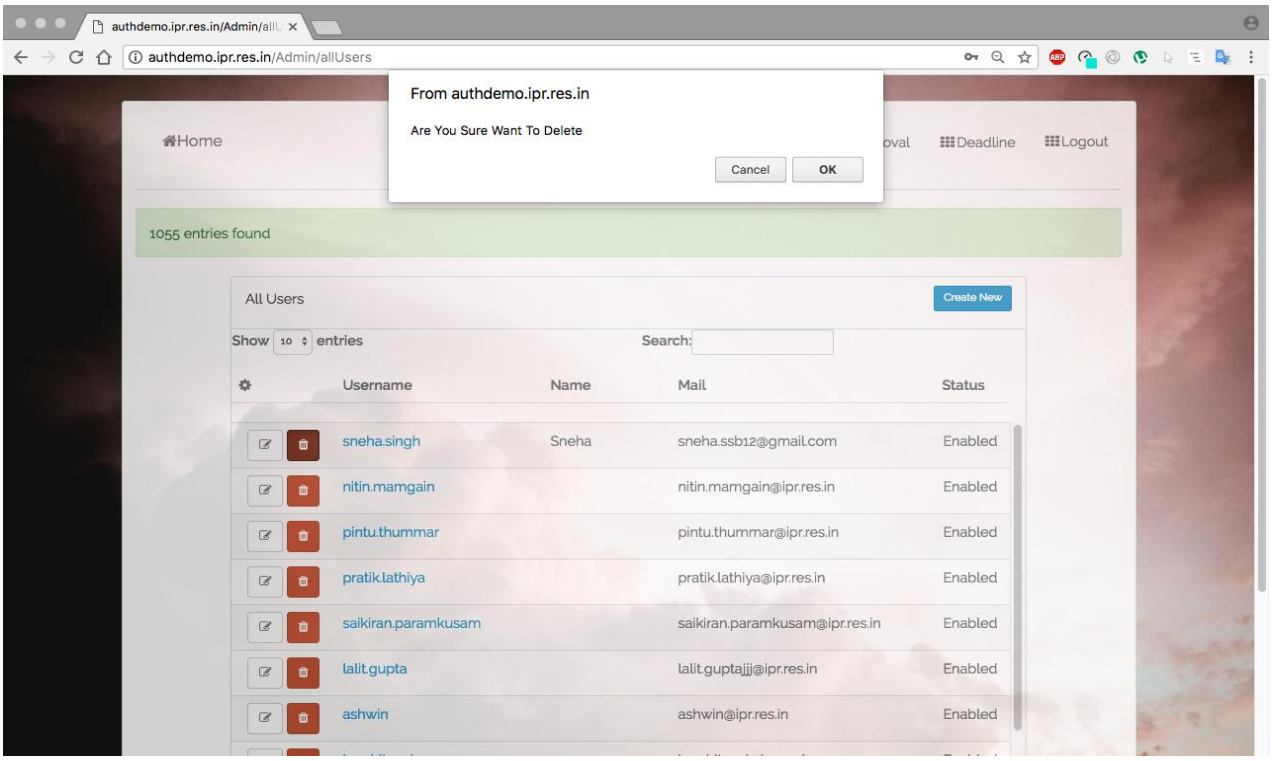
8.8 Edit Profile



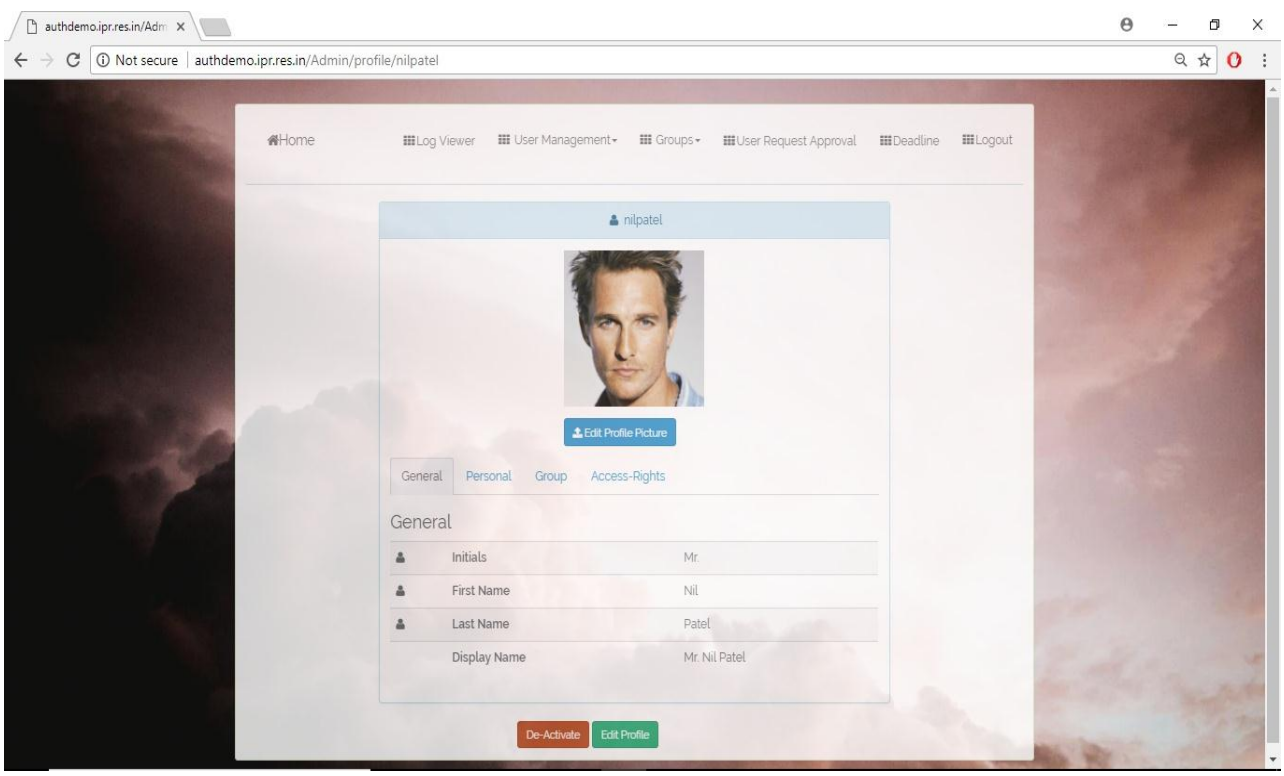
8.9 Admin Login



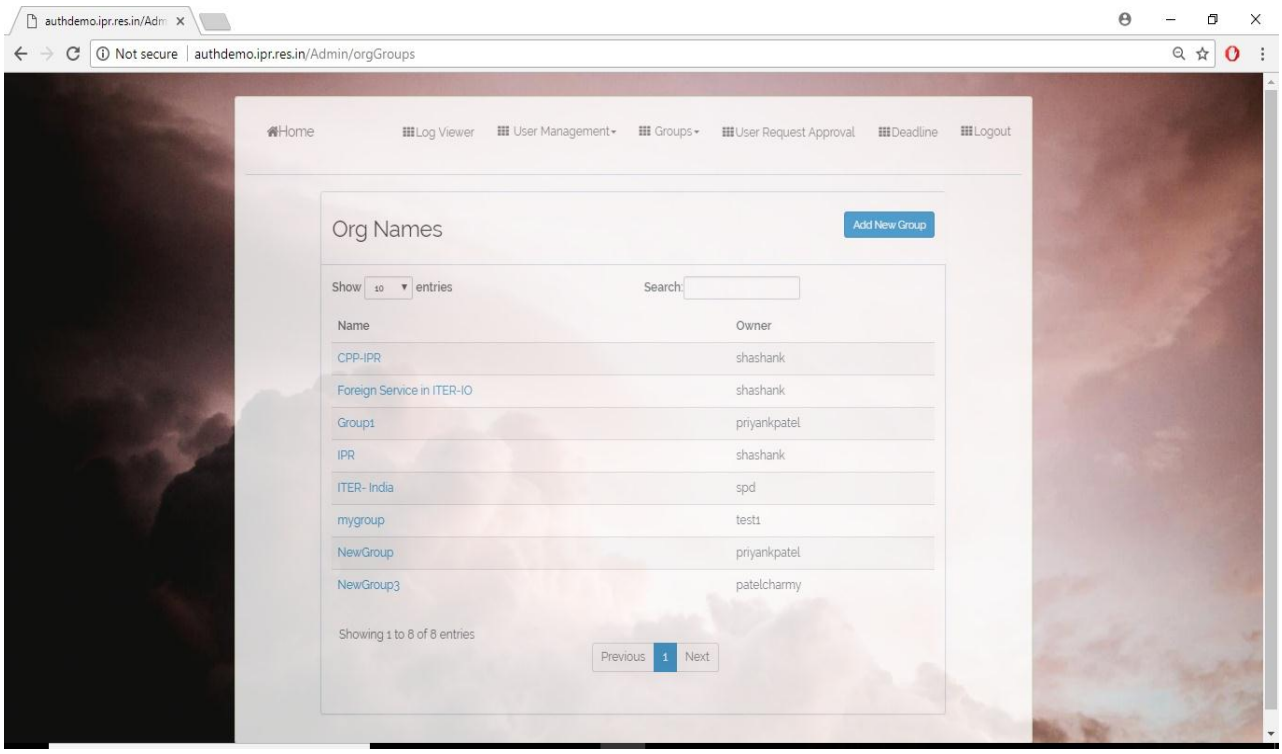
8.10 Log Viewer



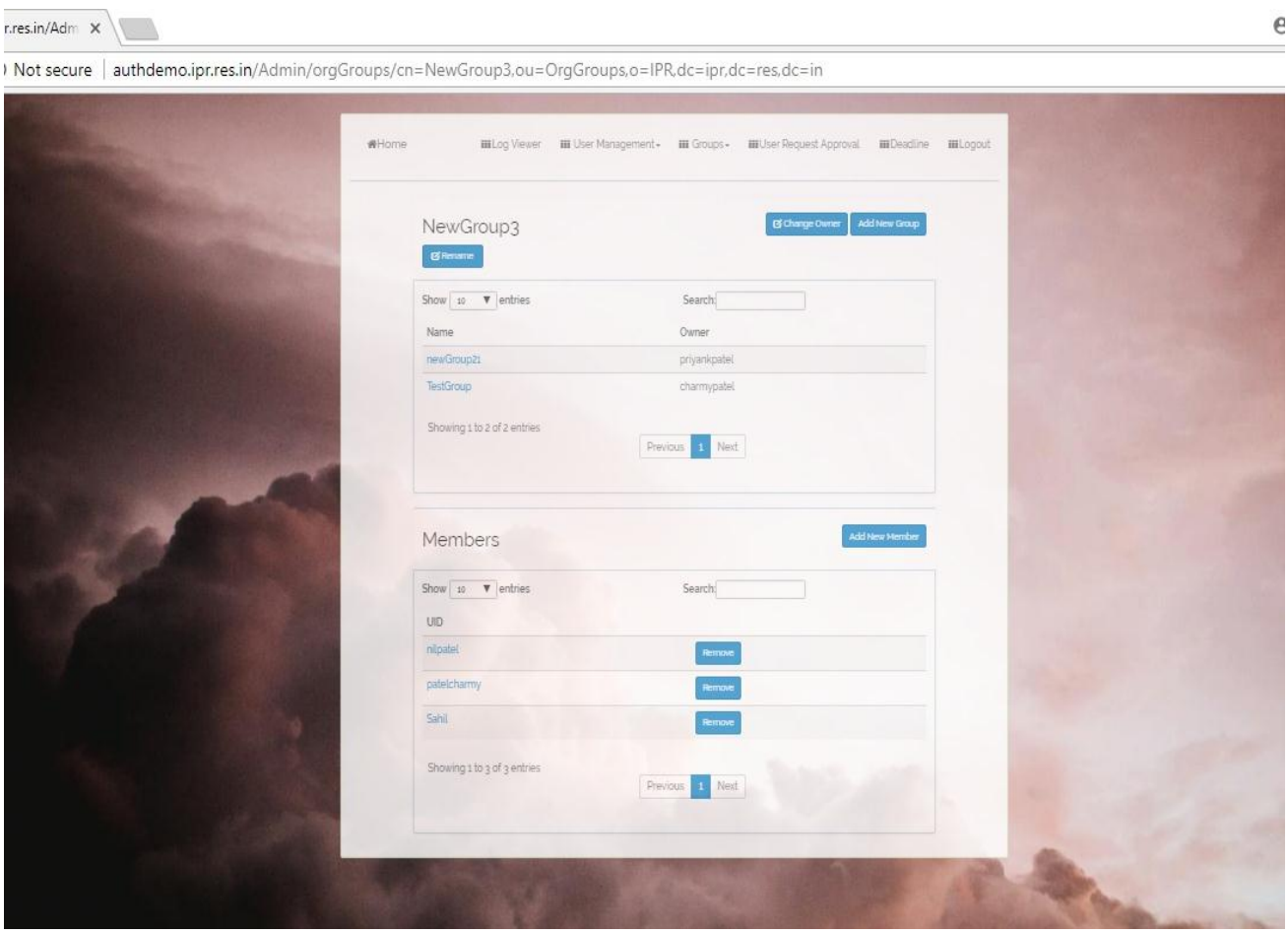
8.11 All Users with edit and delete functionality



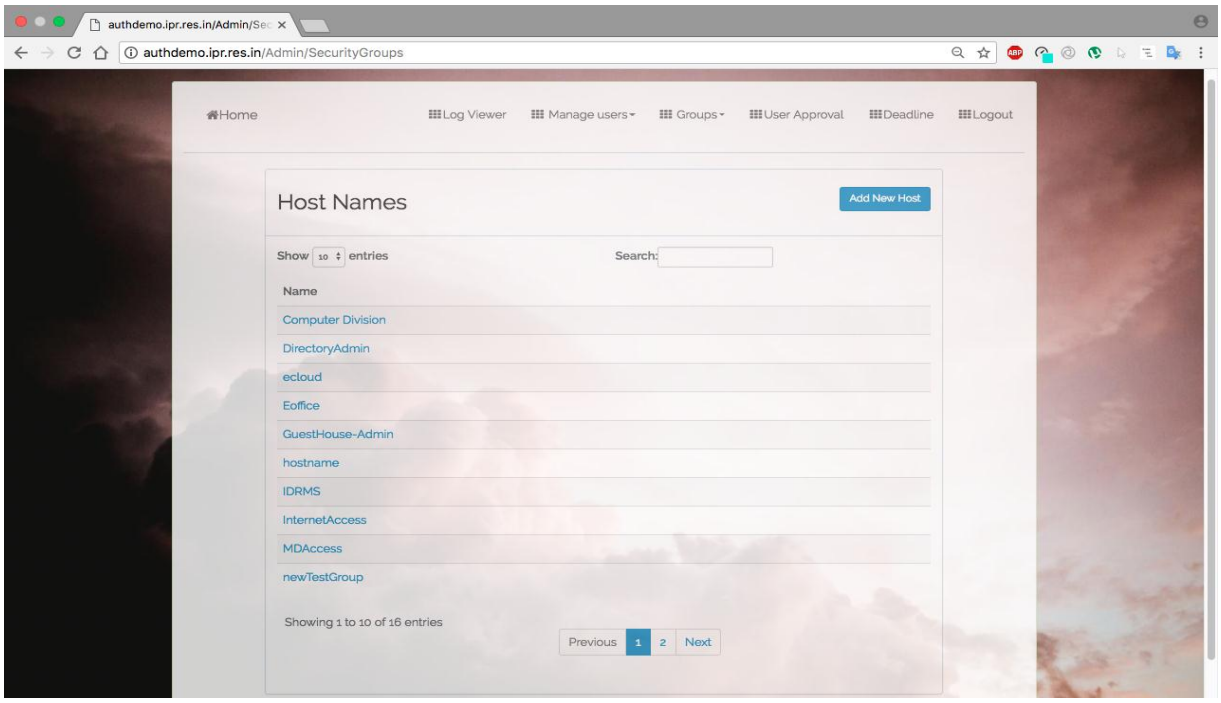
8.12 Activate/De-Activate User



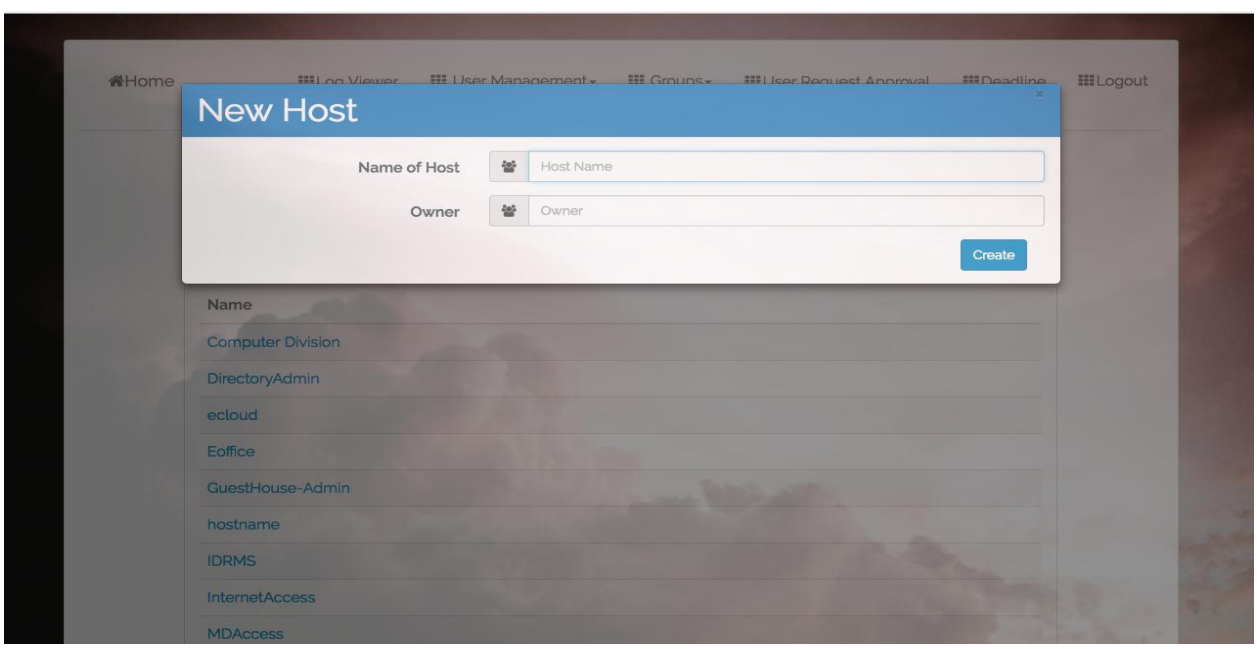
8.13 List of Groups



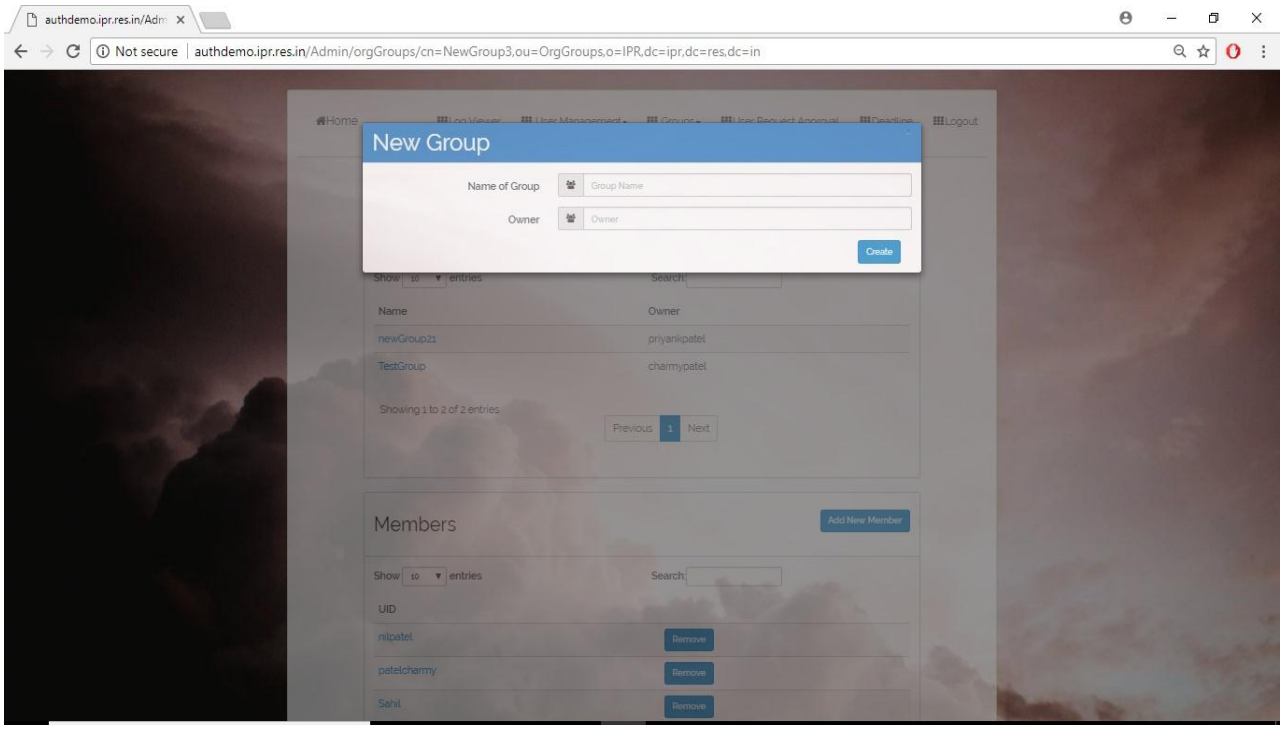
8.14 Group Details



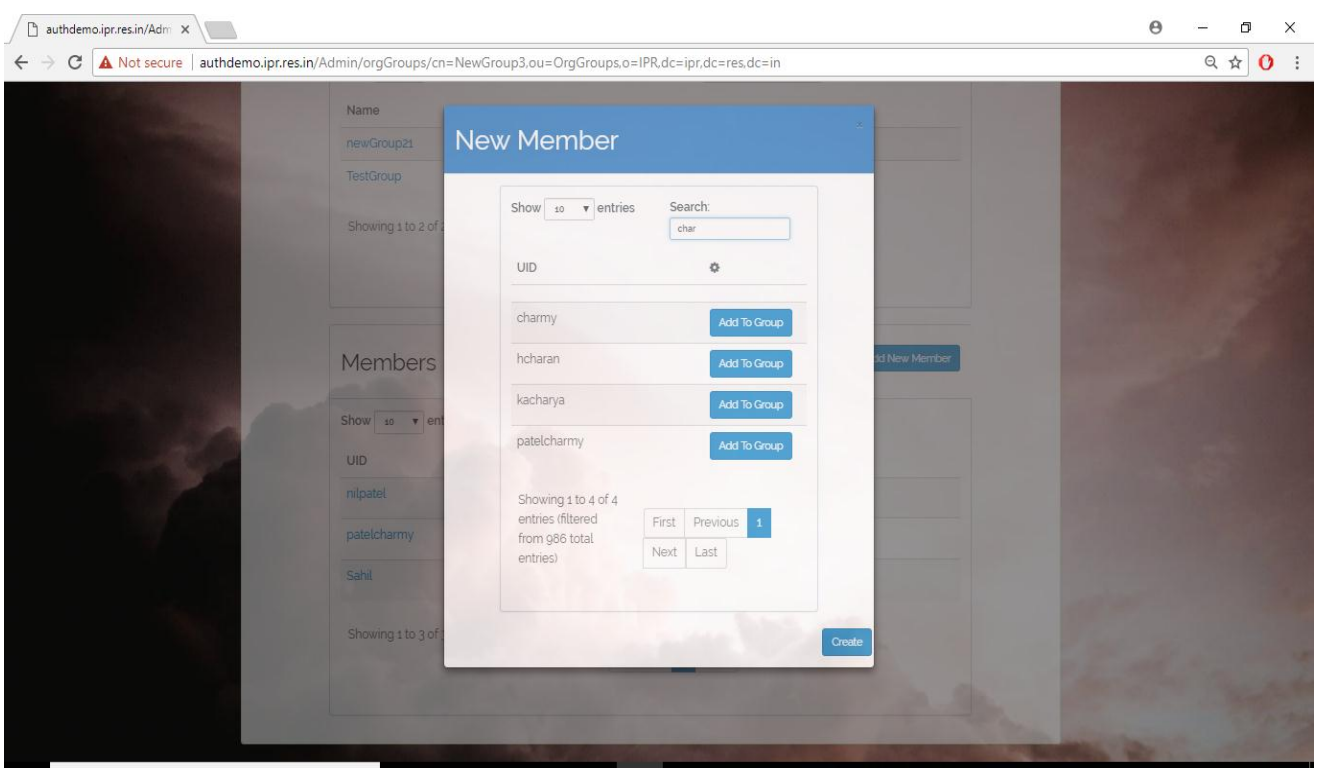
8.15 Host List



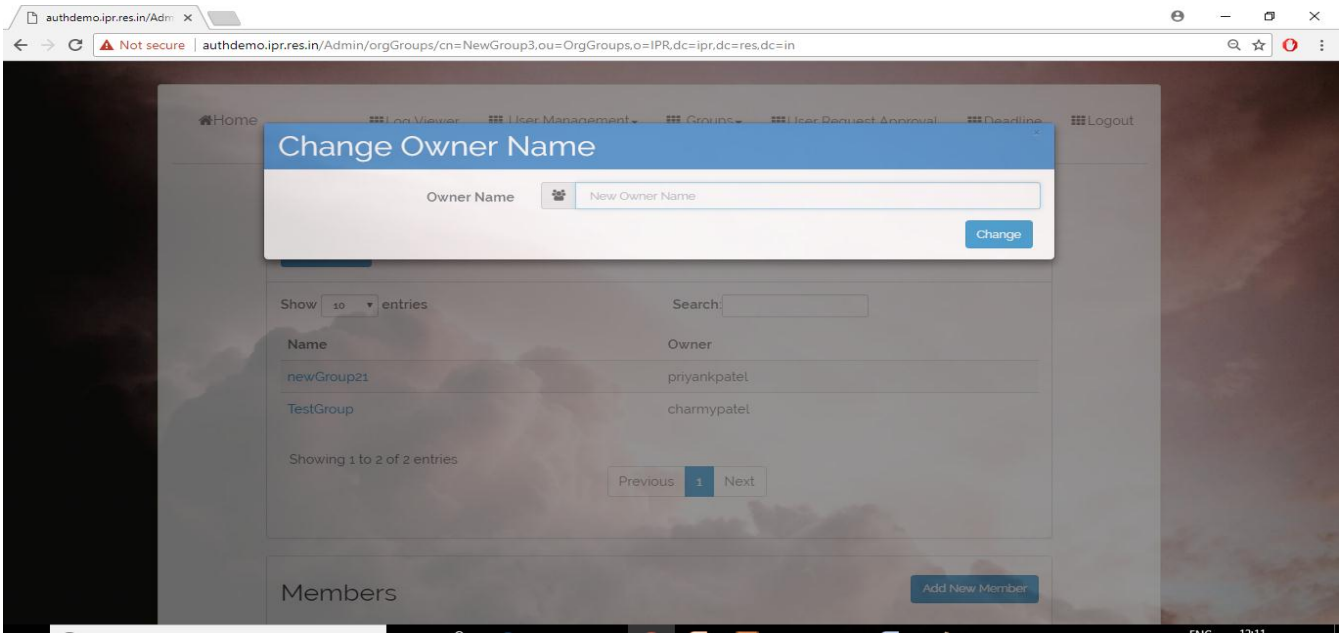
8.16 Add new Host



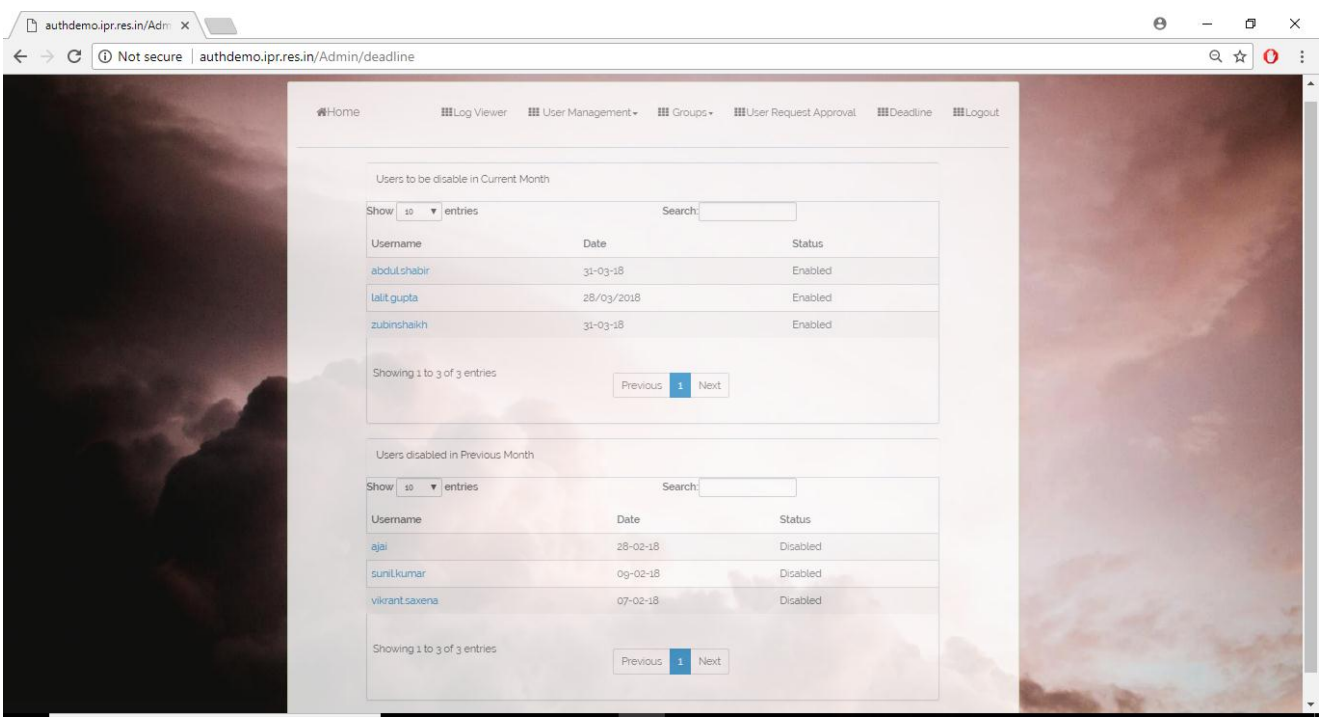
8.17 Add Group



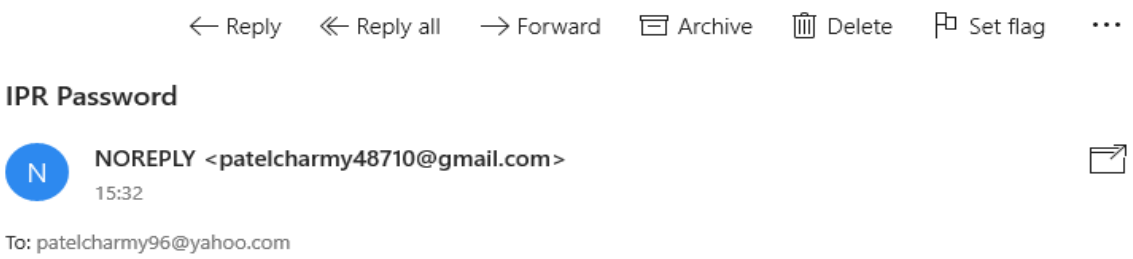
8.18 Add Member



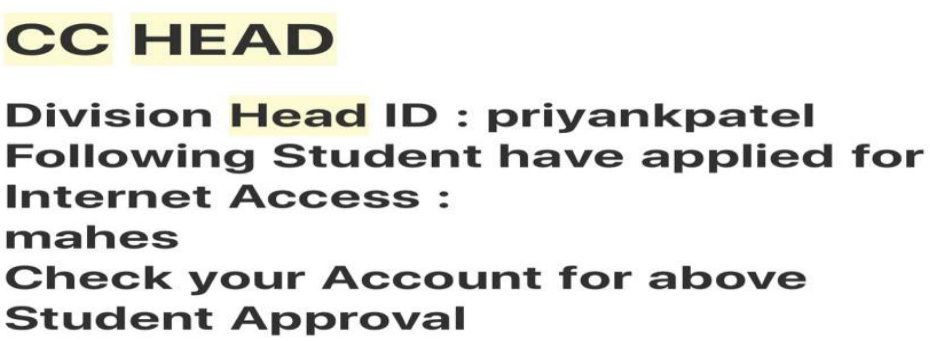
8.19 Change Owner



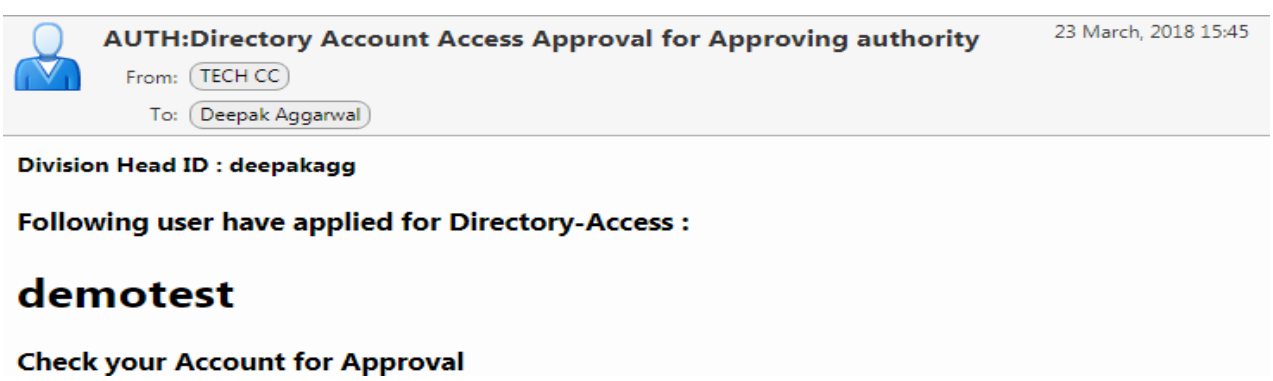
8.20 Deadline



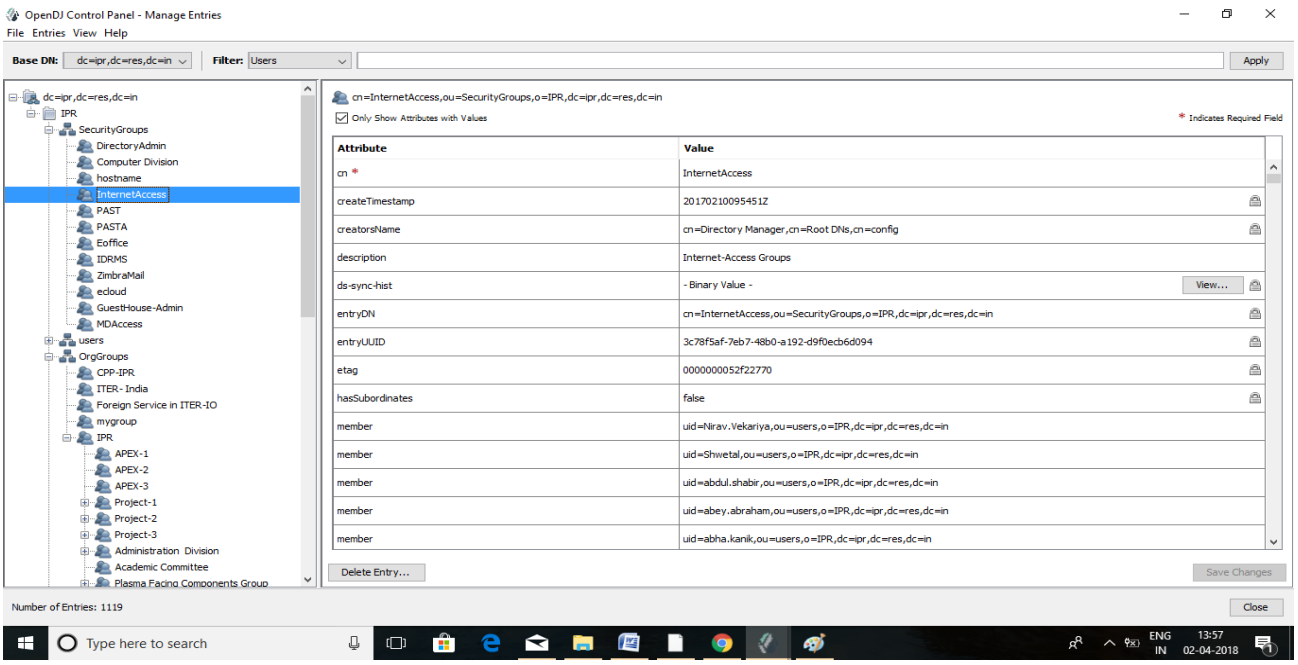
8.21 Temporary Password



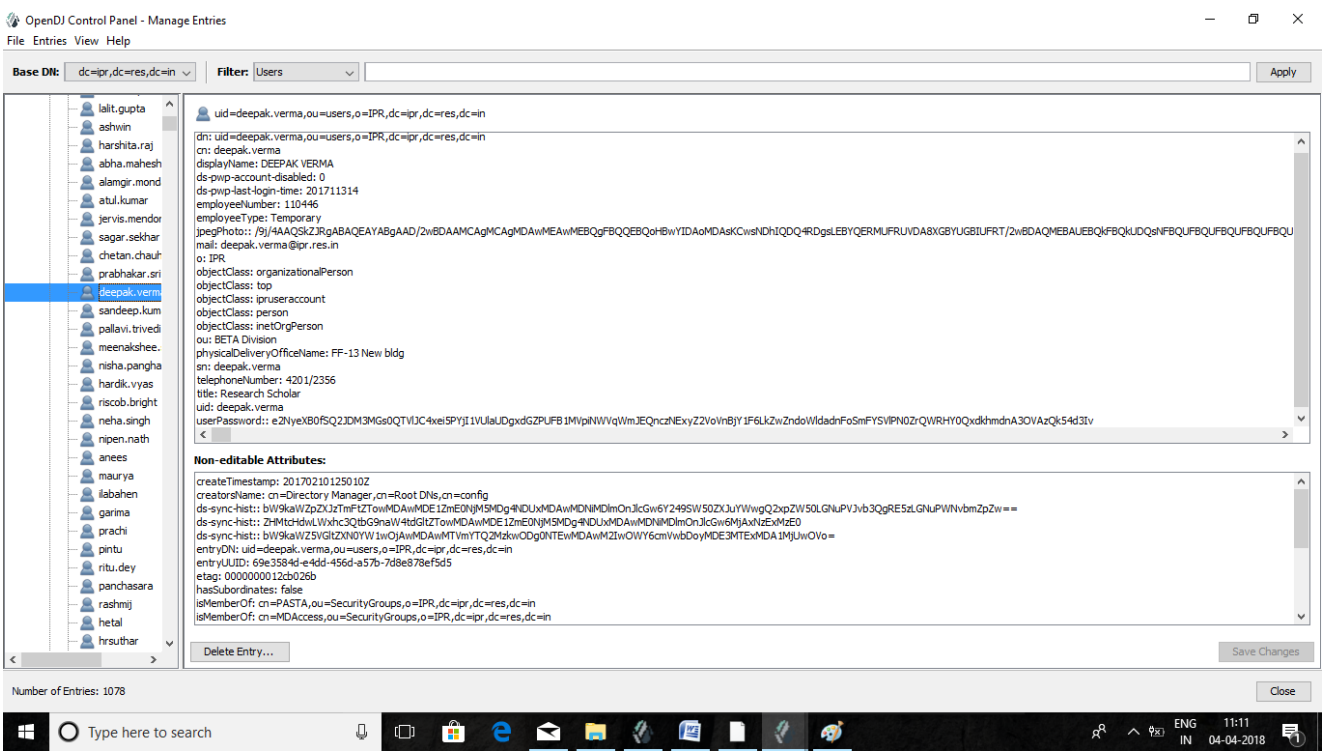
8.22 CC-Head Email (After Div-Head Approval)



8.23 Div-Head Email (After request is submitted)



8.24 LDAP Structure



8.25 LDAP User Structure

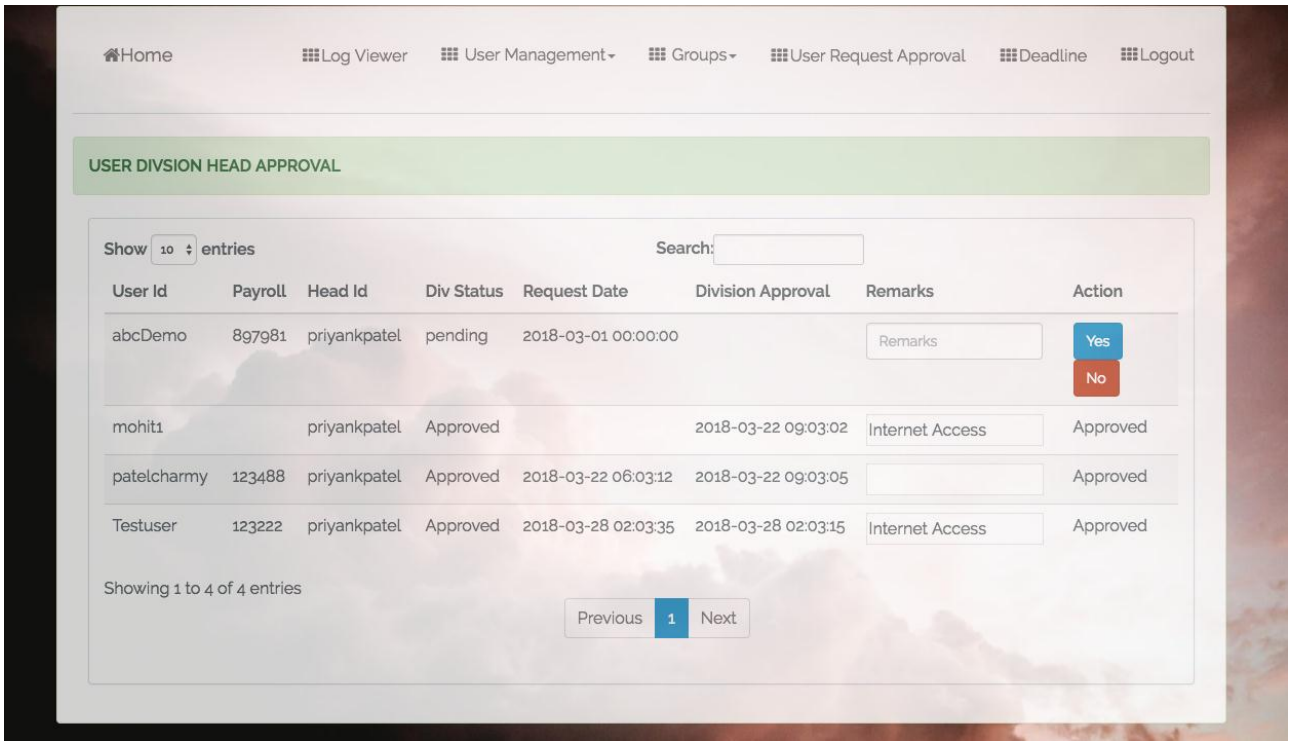


Fig 8.26 Div-Head User Request

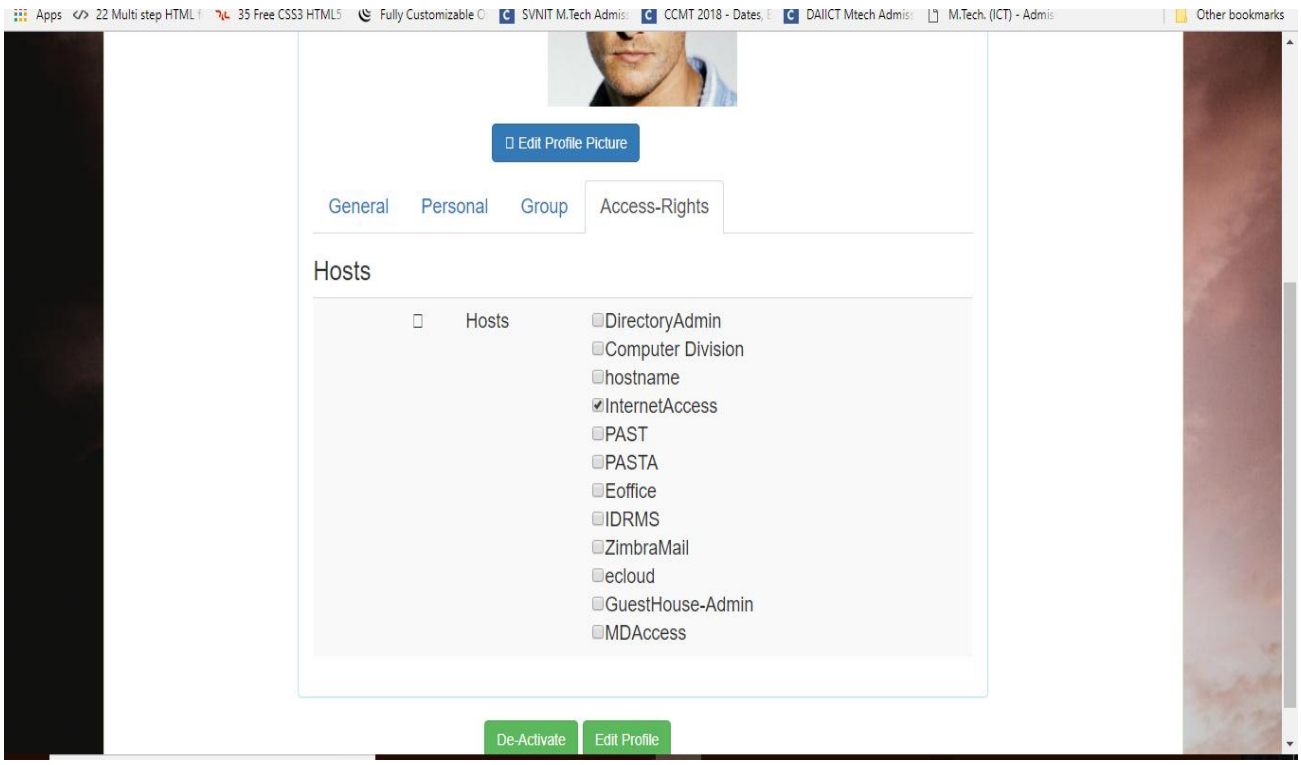


Fig 8.27 Access-Rights of User

9. LIMITATION AND FUTURE ENHANCEMENT

9.1 LIMITATION

1. As this website can only be accessed through IPR server. When new user comes, he obviously not have internet connectivity so he would find difficulty in registering his new account.
2. Group Management is clumsy. When we want to move sub-group under another group may face difficulty.

9.2 FUTURE ENHANCEMENT

1. Organization Group can be visualized as hierarchical form to have a clear vision of groups and their sub-groups.
2. Password change notification can be sent using SMS gate-way.
3. When user is enabled a message is sent to his mobile mentioning your account has been enabled.
4. In deadline, users going to be disable will get email notification before one month, two weeks before, and before one day.

10. CONCLUSION AND DISCUSSION

The project entitled "CAAS" was completed successfully. The system has been developed with much care and free of errors and at the same time it is efficient and less time consuming.

10.1 CONCLUSION

We think that the success of any project doesn't depend only on better development skills but also zeal to listen and help the clients. Thus keeping this perspective in mind we have tried to incorporate every functionality which can make our system to run with ease in future. External Guides is found to be very helpful. We may face difficulty without his proper support and guidance. New ideas of him help us to make our project with nice functionalities.

10.2 DISCUSSION

This project helped us in gaining valuable information and practical knowledge on several topics like Laravel framework which is totally new for us and designing UI of applications, and management of database using LDAP The entire system is secured. Also the project helped us understanding about the development phases of a project and software development life cycle. We learned how to test different features of a project. This project has given us great satisfaction in having designed an application which is going to be live at IPR

REFERENCES

a. Laravel : https://laravel.com/
b. LDAP commands :
https://docs.oracle.com/cd/B10501_01/network.920/a96579/comtools.htm
c. Crudbooster : https://github.com/crocodic-studio/crudbooster/blob/master/docs/en/installation.md
d. Adldap : https://github.com/Adldap2/Adldap2
e. LDAP Authentication :
https://github.com/Adldap2/Adldap2/blob/master/docs/configuration.md
f. Mail : https://laravel.com/docs/5.5/mail
g. Rap2hpoutre Log-viewer : https://github.com/rap2hpoutre/laravel-log-viewer
h. Captcha : https://github.com/mewebstudio/captcha