# 1.INTRODUCTION

Complaints are important drivers for improvement in service delivery for any kind of organisation. People complain because they want the undesirable state to be corrected. A grievance is an oppressive state of things caused due to any wrong or hardship suffered by an individual which forms legitimate grounds for complaint, and the complaint demands remedial action. The grievance redressed mechanism is a part of the prevalent machinery of any administration. Redressal of the grievances is considered as a parameter to measure the efficacy of an organization. No organization can claim to be responsive and user-friendly unless it has established a well-versed system of grievances/complaints redressal. A redressal mechanism would cover complaints of not only a refusal to return documents or certificates, any irregularities in the admission process but also complaints regarding harassment and victimization including harassment. The cell enables a student to express feelings by initiating and pursuing the grievance procedure by the rules and regulations of the college. “Student's Grievance Cell” enquires and analyses the nature and pattern of the grievances in a strictly confidential manner. We develop a web application that is interactive, responsive, and user-friendly. This project covers complaints of all sorts not only complaints regarding ragging and harassment but also these features are added in future complaints regarding irregularities in the admission process, finance, migration, revaluations, and conflicts in names and mark sheets if any and other issues faced by the students. The main purpose of the project is to provide redress to the complaints without time consumption.

A complaint becomes a grievance when the student/staff feels that an injustice has been committed. A grievance is usually more than a complaint. A grievance procedure is a means of internal dispute resolution by which a student may have his or her grievances addressed. Grievance generally give rise to unhappiness, frustration, discontent, and they ultimately lead to the inefficiency of students. The student’s grievance cell desires to promote and maintain a conductive and unprejudiced educational environment. The objectives of student’s grievance cell include the following. A grievance represents a situation in which a student feels that something unfavourable has happened. To support, those students who have been deprived of the services offered by the college, for which he/she is entitled. To make officials of the college responsive, accountable, and courteous in dealing with the students. To ensure effective solution to the student’s grievances with an impartial and fair approach. The registered students access the web application and login is provided for the redressal committee, institute and department with appropriate credentials, institute, and department with appropriate credentials. Student grievance support system functions for several purposes, which includes ensuring the secure environment and familiarizing all faculty and students about their rights and thus it, results in development of the organization. The administration gets thousands of grievances in a day, and it becomes difficult to respond to each one of them. The rate of incoming grievances tends to be high. So, attending to individual issues is not practical, especially when the population is huge. Sometimes there are similar complaints or grievance being registered multiple times, which means that the issue needs urgent attention. To the best of our understanding, there is no existing software application to analyse the grievances, in terms of the similarity of grievances and generate a summary. The Grievance Redressal Cell (GRC) aims to investigate the complaints lodged by any student and redress it as per requirement. The students can state their grievance regarding any academic and non- academic matter within the campus through the online and grievance/ suggestion box. The institution aims at solving the grievances of the students within stipulated academic and non-academic matter within the campus

through the online and grievance box. The institution aims at solving the grievances of the students within stipulated time. It is a formal channel of communication used to resolve grievances. A fair redressal mechanism would boost the morale of all employees greatly. Grievance system provides a sense of confidence and security to the university. It helps in establishing and maintaining a harmonious culture of the university.

Grievance is defined as an official statement of a complaint over something believed to be wrong or unfair. Our project focuses on developing a typical student grievance system, which works and functions for registering student issues. These issues include complaints regarding college environment, faculty feedback and fee collection. Thus, our project ensures a democratic campus environment, acquaints all the faculty and students about their rights, and provides a qualitative and quantitative development of the university. We are developing an online management system for submitting complaints online. By using complaint management system, a user can upload his complaint from anywhere by using this website on his phone or computer online. User can submit his complaint by easily creating his/her own profile; also, user can check status of their complaints and view what kind of action is taken. It is based on centralize management and only admin can check or solve the complaint, admin have authority to remove a user. The main objective of the complaints management system is to make complaints easier to resolve and to target problem areas. It is used to record, resolve, and respond to customer complaints.

Protection of human rights is essential for all round development of an individual's personality. To realize the primary needs of the students and staff and secure civil liberties for everybody, a Grievance Redressal Cell is needed. The function of the cell is to investigate the complaints lodged by any student at college and judge its merit. The Grievance Cell is also empowered to investigate matters of harassment. Anyone with a genuine grievance may approach the department members in person, or in consultation with the officer in-charge Students' Grievance Cell. The function of the cell is to investigate the complaints lodged by any student and judge its merit. The Grievance cell is also empowered to investigate matters of harassment. Anyone with a genuine grievance may approach the department members in person, or in consultation with the officer in-charge Student's Grievance Cell. The formation of Students and Staff Grievance Cell is to promote and maintain a conducive and unprejudiced educational environment where students and staffs are experiencing that; there has been an infringement of their rights. It is a measure to develop responsive and accountable attitude among college officials to ensure that, there is no laxity in terms of fair-deal with the students. It is to deal with the complex situations in a tactful manner to lessen the condition felt to be oppressive or dissatisfied.

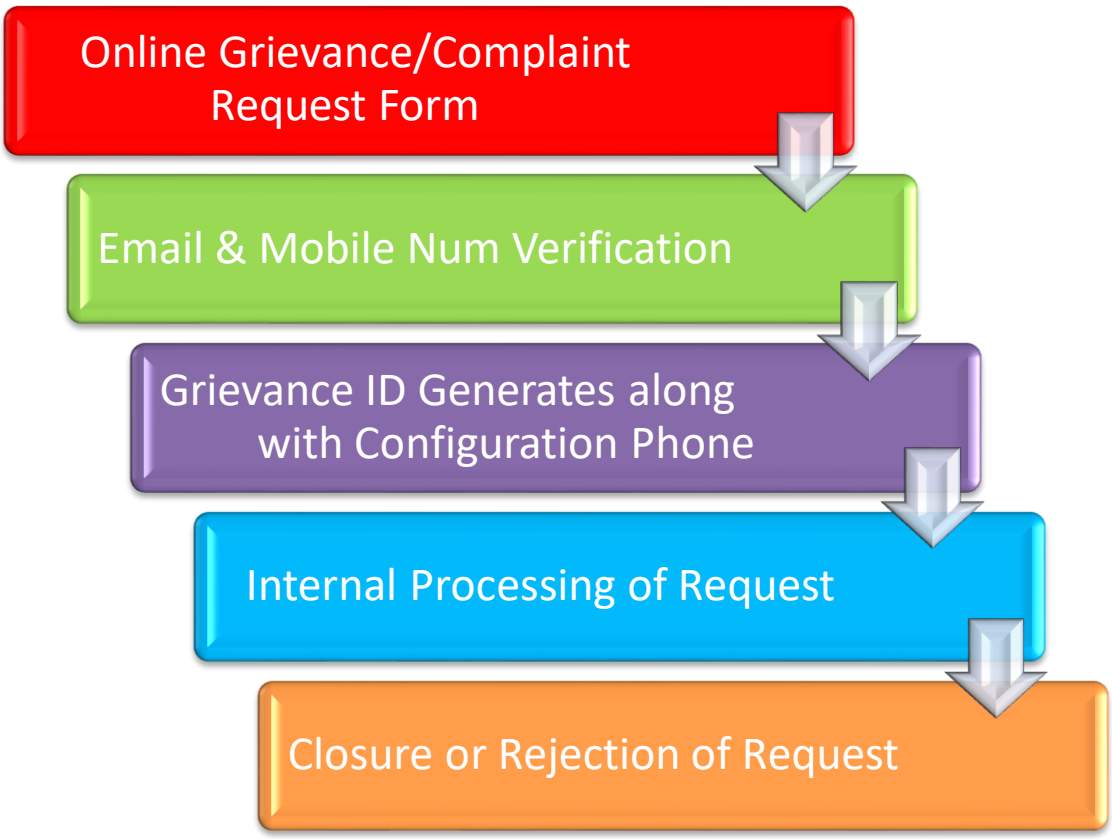


Fig.1. Grievance Procedure

Student Grievance Redressal provides an effective mechanism to promptly resolve all student grievances. The Grievance Redressal Cell was constituted by the college as per the regulations of University Grants Commission. A Grievance is any discontent or dissatisfaction, whether expressed or not, whether valid or not, arising out of anything that is directly connected to the institute, and which a student thinks, believes, or even feels, is unfair, unjust, or inequitable. All grievances of the student are redressed through a single window system. Any student with a genuine grievance may approach Associate Person - Grievances to submit his/her grievance in a proper format. All grievances are redressed in a systematic way by involving the respective department/person directly connected to the specific issue.

# 2.PROBLEM STATEMENT

Many of the students and staff are facing issues in the university, which can be resolved using the grievance. It may be any genuine or imaginary feeling of dissatisfaction or injustice which an employee or student experiences about his/her job and education about the management policies and procedures. Problem must be expressed by the employee and brought to the notice of the management and the organization. Grievances take the form of collective disputes when they are not resolved. In addition, they will then lower the morale and efficiency of the employees. Unattended grievances result in frustration, dissatisfaction, low productivity, lack of interest in work, absenteeism, etc. If the grievances are not within the purview of the committee, then the issues are brought to the notice of the authorities. The function of the cell is to investigate the complaints lodged by any student at college and judge its merit. The Grievance Cell is empowered to investigate matters of harassment. Grievances may be sent through e-mail to the member/officer in-charge of student's grievance cell. Grievance creates a platform where staff and students can raise their problems about academic and non-academic matters.


Fig. 2.1. Student Compliant

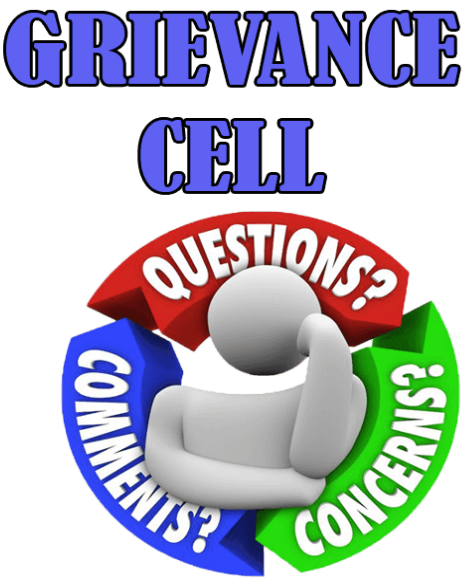


Fig. 2.2. Grievance Cell

## 2.1 OBJECTIVE

The main objective of the grievance cell is to promote and maintain a conducive and harmonious educational environment among the students. The student's grievance cell constitutes the redressed of the problems reported by the students at the college. This is with the following objectives in mind. Grievance cell formed to keep the healthy working atmosphere amongst staff, students, and parents. This cell helps students to record their complaints and solve their problems related to academics, resources, and personal grievances freely and frankly without any fear of victimization. To uphold the dignity of the College by ensuring strife free atmosphere in the College through promotion of cordial Student-Student relationship and Student-teacher relationship etc. To provide responsive, accountable, and easily accessible machinery for settlement of grievances and to take measures in the college undertakings to ensure expeditious settlement of grievances of Students to maintain a harmonious educational atmosphere in the institute. It is to deal with the complex situations in a tactful manner to lessen the condition felt to be oppressive or dissatisfied. Encouraging the students to express their grievances / problems freely and frankly, without any fear of being victimized.

Advising Students of the College to respect the right and dignity of one another and show utmost restraint and patience whenever any occasion of rift arises. Advising all the Students to refrain from inciting Students against other Students, teachers, and College administration. Advising all staffs to be affectionate to the students and not behave in a vindictive manner towards any of them for any reason. To keep the dignity of the college high by ensuring conflict free atmosphere in the college by promoting good student-student relationship and student-teacher relationship. To ensure effective solution to the student grievances with an impartial and fair approach. To advising students at the college to respect each other and be patient whenever any occasion of conflict arises. To advise all the students to refrain from stirring up students against other students, teachers, and college administration.

Ragging in any form is prohibited in and outside the institution. Any violation of ragging and disciplinary rules should be urgently brought to the notice of the principal. Ragging complaints will be resolved according to ragging rules. Woman harassment complaints are resolved with respective government guidelines by respective section. To ensure an effective solution to the student grievance with an impartial and fair approach. To prevent non-transparent or unfair practices adopted. Advising students of the institution to respect the rights and dignity of one another and show utmost restraint and whenever any occasion of the rift arises. To help the students to lodge their complaints and solve their problems related to academics, resources, and personal grievances freely without any fear of victimization. To make officials of the college responsive, accountable, and courteous in dealing with the students. To ensure effective solution to the student's grievances with an impartial and fair approach. The objective of the Grievance Redressal Cell is to develop a responsive and accountable attitude among all the stakeholders to maintain a harmonious educational environment on the campus.

## 2.2 SCOPE

The cell will deal with grievances received in writing from the students about any of the following matters are Academic matters: Related to timely issue of duplicate mark-sheets, transfer certificates, conduct certificates, or other examination related matters. Financial matters: Related to dues and payments for assorted items from library, hostels etc. Other matters: Related to certain misgivings about conditions of sanitation, preparation of food, availability of transport, victimization by teachers etc.

**A. ACADEMIC AND CURRICULAR ISSUES**

a. Matters pertaining to Admissions: Admission Grievance Committee

b. Matters pertaining to Teaching-Learning: Teacher-In-Charges of respective Departments

c. Matters pertaining to Internal Assessment: Internal Assessment Committee

**B. NON-ACADEMIC ISSUES**

a. Matters pertaining to general discipline: Discipline Committee

b. Matters pertaining to discrimination: Equal Opportunity Cell

c. Matters pertaining to infrastructure: Administrative Office

d. Matters pertaining to sexual harassment: Internal Complaints Committee

e. Matters pertaining to ragging: Anti-Ragging Committee and Anti-Ragging Cell

**C. GRIEVANCES RELATED TO HOSTEL**

a. Matters pertaining to Boys Hostel: Discipline Committee for Hostel (Boys Hostel)

b. Matters pertaining to Girls Hostel: Discipline Committee for Hostel (Girls Hostel)

# 3.SYSTEM ANALYSIS

## 3.1 SYSTEM

A system is an arrangement in which all its unit assemble work together according to a set of rules. It can also be defined as a way of working, organizing, or doing one or many tasks according to a fixed plan. For example, a watch is a time displaying system. Its component follows a set of rules to show time. If one of its parts fails, the watch will stop working. So, we can say, in a system, all its subcomponents depend on each other. This is a process of collecting and interpreting facts, identifying the problems and decomposition of a system into its components. System analysis is used in every situation where something is developed. Analysis specifies what the system should do.

## 3.2 PROPOSED SYSTEM

The idea is to automate the entire complaint process. Grievance can be lodged based on the level and based on the category. Students can be able to track the grievance once the complaint has been registered. Students and cell members are provided a platform where they can discuss regarding the grievance. Students can provide any document as a proof, which makes the complaint strong. Cell members can track the pending and completed grievances. User friendliness is provided in the application with various controls. The system makes the overall project management much easier and flexible. It can be accessed over the Internet. There is no risk of data mismanagement at any level while the project development is under process. The system provides higher level of security using various protocols.

## 3.3 FEATURES

The project makes work done at faster way. The software is applicable to view the student's complaints details when the student lodges it. Our project aims at a Grievance process automation, i.e., we have tried to computerize various process of Student Grievances. The system generates types of information that can be used for various purposes. It satisfies the user requirements.

**ADVANTAGES OF PROPOSED SYSTEM**

Security and satisfaction of the student

Increased accuracy and reliability

Easy maintenance and reliability

# 4.SOFTWARE REQUIREMENTS & SPECIFICATIONS

## 4.1 FUNCTIONAL SPECIFICATIONS

The system after careful analysis has been identified to be presented with the following modules:

The modules involved are:

1. Admin module.
2. User module.
3. Retrieve the Student Data.

- User Module: Used for maintaining the Users details.
- Login Module: Used for maintaining the login details.

**ADMIN**

Admin is a person whose responsibility is to maintain the database that contains every data regarding all students and staff. Admin can add student's details into the database can be able to delete the student's details but can't update the user's details.

**USER**

Here user means either student or staff. The responsibility of the student is to login into the grievance page and fill the details of the complaint. Whenever the student registered his/her name then the student will be given by one individual mail and password. When the student will type the correct user mail id and password, they will enter another page. In that page student can view the report.

## 4.2 REQUIREMENTS & SPECIFICATIONS

Requirement specification provides a high secure storage to the web server efficiently. Software requirements deal with software and hardware resources that need to be in on a server, which provides optimal functioning for the application. These software and hardware requirements need to be installed before the packages are installed. These are the most common set of requirements defined by any operation system. These software and hardware requirements provide a compatible support to the operation system in developing an application.

## 4.3 HARDWARE REQUIREMENTS

The hardware requirement specifies each interface of the software elements and the hardware elements of the system. These hardware requirements include configuration characteristics.

Hard disk speed : To improve performance, install windows and Xampp Server on a solid-state drive (SSD).

Hard disk space : Minimum of 800MB up to 210 GB of available space, depending on features installed, typical installations require 20-50 GB of free space.

Processor : 1.8 GHz or faster processor. Quad-core or better recommended

RAM: 4 GB to 8 GB of RAM recommended (2.5 GB minimum if running on a virtual machine)

Graphic card : Video card that supports a minimum display resolution of 720p (1280 by 720) Sublime Text will work best at a resolution of WXGA (1366 by 768) or higher.

## 4.4 SOFTWARE REQUIREMENTS

Operating System : Windows 7 or above

Coding Language : JavaScript, HTML, CSS, SQL

Software : NIVIDIA graphic drivers like CURD 11.2, CUDNN SDK8, 1.0

IDE : Sublime Text

## HTML

HTML, in full hypertext markup language, a formatting system for displaying material retrieved over the Internet. Each retrieval unit is known as a Web page, and such pages frequently contain hypertext links that allow related pages to be retrieved. HTML is the markup language for encoding Web pages. HTML markup tags specify document elements such as headings, paragraphs, and tables. They mark up a document for display by a computer program known as a Web browser. The browser interprets the tags, displaying the headings, paragraphs, and tables in a layout that is adapted to the screen size and fonts available to it.

## JS

JavaScript (JS) is a scripting language, primarily used on the Web. It is used to enhance HTML pages and is commonly found embedded in HTML code. JavaScript is an interpreted language. Thus, it doesn't need to be compiled. JavaScript renders web pages in an interactive and

dynamic fashion. This allowing the pages to react to events, exhibit special effects, accept variable text, validate data, create cookies, detect a user's browser, etc.

## 4.5 FUNCTIONAL REQUIREMENTS

The functional requirements refer to the system needs in an exceedingly computer code engineering method. The key goal of determinant "functional requirements" in an exceedingly product style and implementation is to capture the desired behaviour of a software package in terms of practicality and the technology implementation of the business processes

**PRE-PROCESSING**

In this module, dataset is organized in required manner and data can be split into different categories to determining the various type's data.

## 4.6 NON-FUNCTIONAL REQUIREMENTS

All the other requirements, which do not form a part of the above specification, are categorized as non-functional needs. A system perhaps needed to bestow a show upon the user of the quantity of records during info. If the quantity must be updated in real time, the system architects should make sure that the system is capable of change the displayed record count at intervals associate tolerably short interval pf the quantity of records dynamic. Comfortable network information measure may additionally be a non-functional requirement of a system.

**THE FOLLOWING ARE THE FEATURES:**

- Accessibility
- Availability
- Backup
- Configuration Management
- Documentation
- Disaster Recovery
- Efficiency

# 5.SYSTEM DESIGN

## 5.1 DEFINITION

The most creative and challenging face of the system development is System Design. It provides the understanding and procedural details necessary for implementing the system recommended in the feasibility study. Design goes through the logical and physical stages of development.

In designing a new system, the system analyst must have a clear understanding of the objectives, which the design is aiming to fulfill. The first step is to determine how the output is to be produced and in what format. Second, input data and master files have to be designed to meet the requirements of the proposed output. The operational phases are handled through program construction and testing.

Design of a system can be defined as a process of applying various techniques and principles for the purpose of defining a device, a process, or a system in sufficient detail to permit its physical realization. Thus, system design is a solution to "how to" approach to the creation of a new system. Thus, important phase provides the understanding and the procedural details necessary for implementing the system recommended in the feasibility study. The design step provides a data design, architectural design, and a procedural design.

## 5.2 INPUT DESIGN & OUTPUT DESIGN

### INPUT DESIGN

In the input design, user-oriented inputs are converted into a computer-based system format. It also includes determining the record media, method of input, speed of capture and entry on to the screen. Online data entry accepts commands and data through a keyboard. The major approach to input design is the menu and the prompt design. In each alternative, the user's options are predefined. The data flow diagram indicates logical data flow, data stores, source and destination. Input data are collected and organized into a group of similar data. Once identified input media are selected for processing.

In this software, importance is given to develop Graphical User Interface (GUI), which is an important factor in developing efficient and user-friendly software. For inputting user data, attractive forms are designed. User can also select desired options from the menu, which provides all possible facilities.

Also, the important input format is designed in such a way that accidental errors are avoided. The user must input only just the minimum data required, which also helps in avoiding the errors that the users may make.

**OUTPUT DESIGN**

In the output design, the emphasis is on producing a hard copy of the information requested or displaying the output on the CRT screen in a predetermined format. Two of the most output media today are printers and the screen. Most users now access their reports from a hard copy or screen display. Computer's output is the most important and direct source of information to the user, efficient, logical, output design should improve the systems relations with the user and help in decision-making.

As the outputs are the most important source of information to the user, better design should improve the system's relation and should help in decision-making. The output device's capability, print capability, response time requirements etc. should also be considered form design elaborates the way output is presented and layout available for capturing information. It's very helpful to produce the clear, accurate and speedy information for end users.

## 5.3 SOFTWARE DESIGN

It is a process of planning the new or modified system. Analysis specifies what a new or modified system does. Design specifies how to accomplish the same. Design is essentially a bridge between requirement specification and the final solution satisfying the requirements. Design of a system is essentially a blueprint or a solution for the system.

The design process for a software system has two levels. At first level the focus is on depending in which modules are needed for the system, the specification of these modules and how the modules should be interconnected. This is what is called system designing of top-level design. In the second level, the internal design of the modules, or how the specification of the module can be satisfied is described upon. This design level is often called detailed design or logic design.

The first level produces system design, which defines the components needed for the system, and how the components interact with each other. It focuses is on depending on in which those modules are needed for the system, the specification of these modules and how the module should be interconnected.

## 5.4 INTRODUCTION TO UML

United Modeling Language is the one of the most exciting tools in the world of system development today. Because UML enables system builders to create blueprints that capture their visions in a standard, easy to understand way and communicate them to others. The UML is brainchild of Grady Brooch, James Rumbaugh, and Ivar Jacobson.

The UML is a language for

Visualizing

Specifying

Constructing

Documenting

These are the artifacts of a software-intensive system. The abbreviation for UML is Unified Modeling Language and is being brought of a designed to make sure that the existing ER Diagram which do not serve the purpose will be replaced by this UML Diagram where in these language as its own set of Diagrams.

Some of the diagrams that help for the Diagrammatic Approach for the object-oriented software engineering are:

Class Diagram

Use case Diagram

Deployment Diagram

State chart Diagram

Using the above-mentioned diagram, we can show the entire system regarding the working of the system or the flow of control and sequence of flow the state of the system and the activities involved in the system.

## 5.5 COMPONENTS OF THE UML

The UML consists of several graphical elements that combine to form diagram. Because it's a language, the UML has rules for combining these elements. The purpose of the diagram to present multiple views of the system, and this set of multiple views is called a Model. A UML Model of a system is something like a scale model of building.

UML model describes what a system is supposed to do. It doesn't tell how to implement the system. The following are the main nine component diagrams of UML.

## 5.5.1 CLASS DIAGRAM

A class is a category or group of things that has similar attributes and common behavior. A Rectangle is the icon that represents the class it is divided into three areas. The upper most area contains the name; the middle area contains the attributes, and the lowest areas show the operations. Class diagrams provide the representation that developers work from, class diagram help on the analysis side, too.

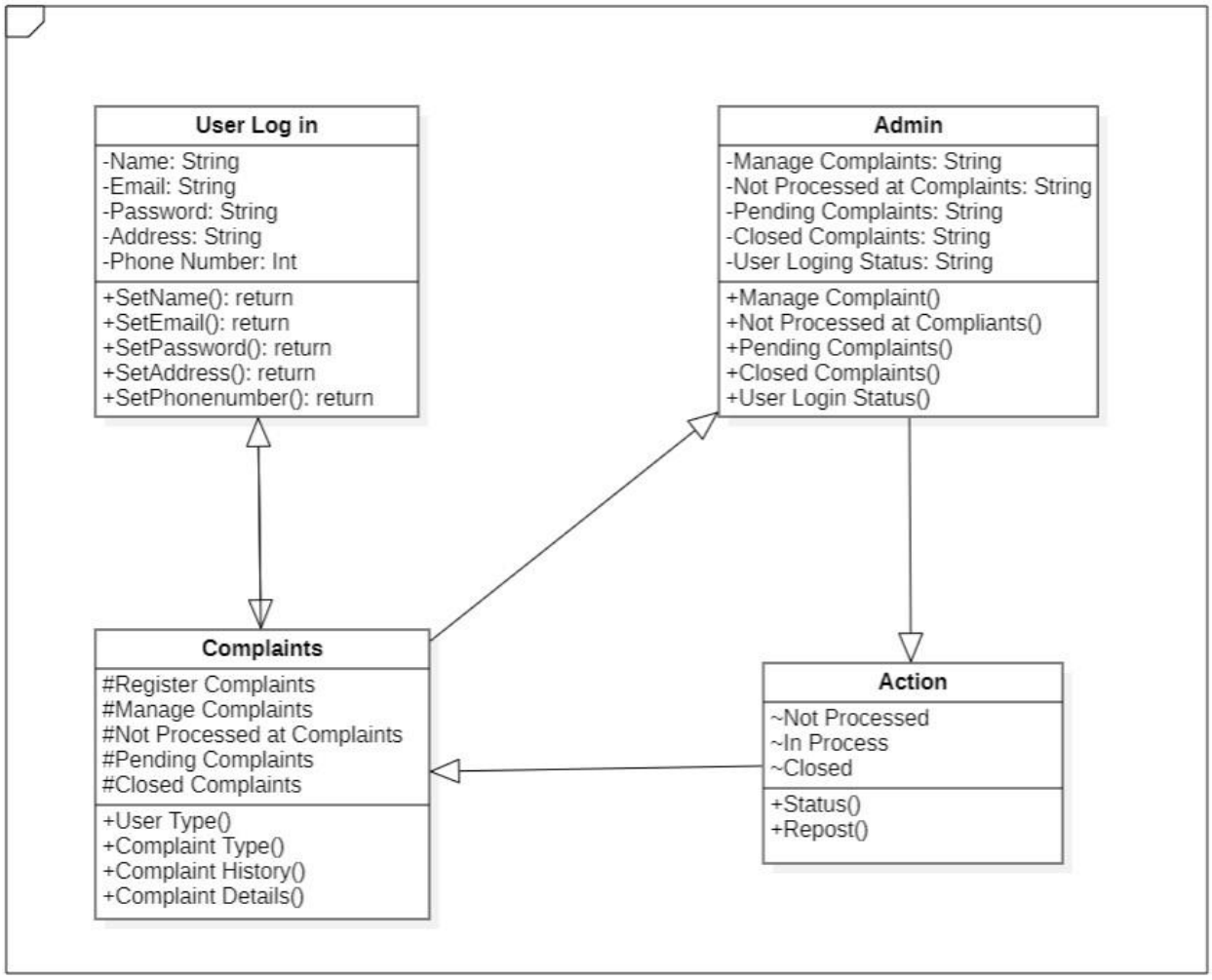


Fig. 5.5.1. Class Diagram for Grievance Cell

## 5.5.2 USE CASE DIAGRAM

A Use case is a description of a systems behavior from a user's standpoint. For system developer this is a valuable tool: it's a tired-and-true technique for gathering system requirements from a user's point of view. This is important if the goal is to build a system that real people can use. A little stick figure is used to identify an actor the ellipse represents use-cases.

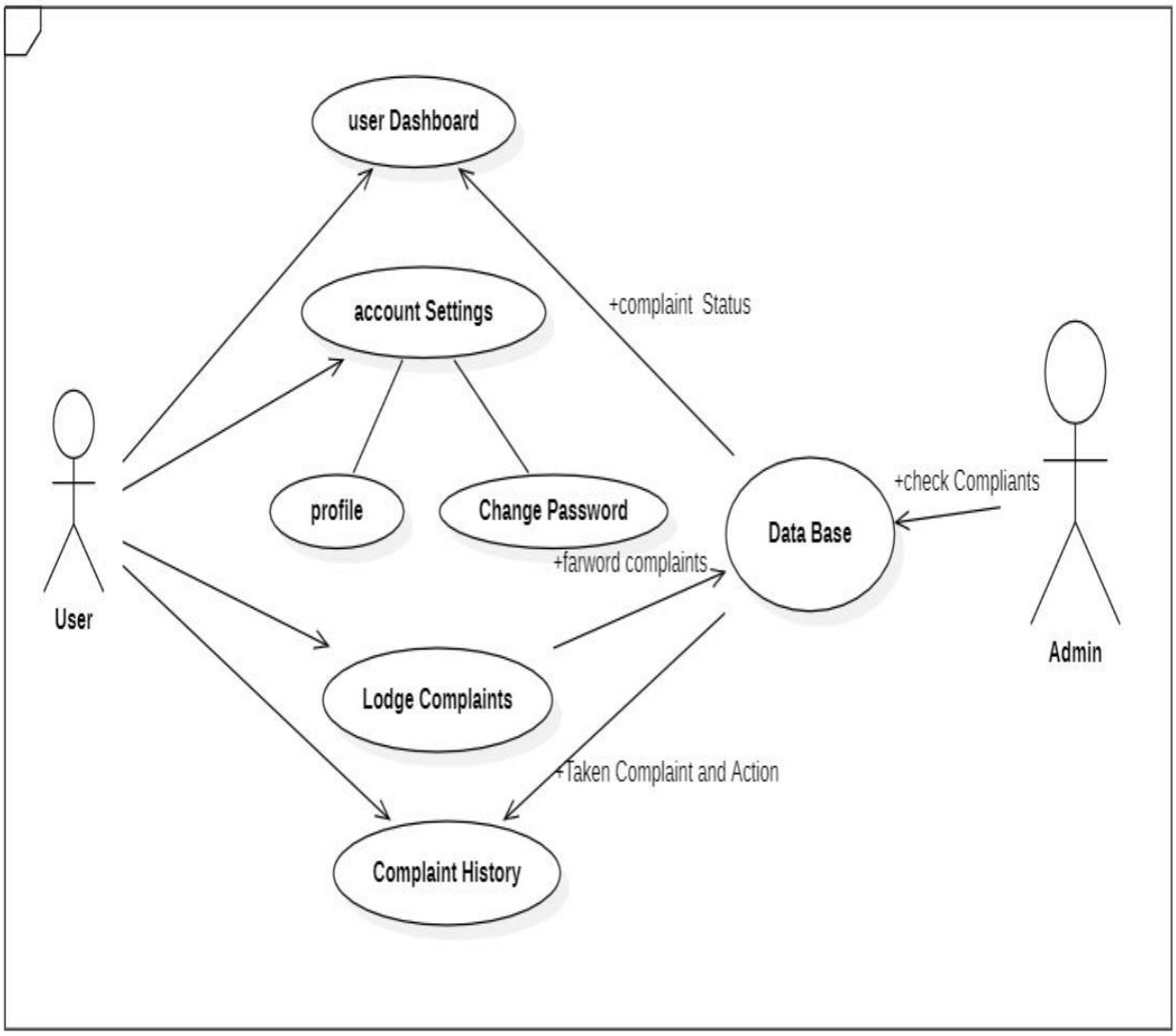


Fig. 5.5.2. Use Case Diagram for Grievance Cell

## 5.5.3 DEPLOYMENT DIAGRAM

A deployment diagram is a UML diagram type that shows the execution architecture of a system, including nodes such as hardware or software execution environments, and the middleware connecting them. Deployment diagrams are typically used to visualize the physical hardware and software of a system. Using it you can understand how the system will be physically deployed on the hardware. Deployment diagrams help model the hardware topology of a system compared to other UML diagram types which mostly outline the logical components of a system.

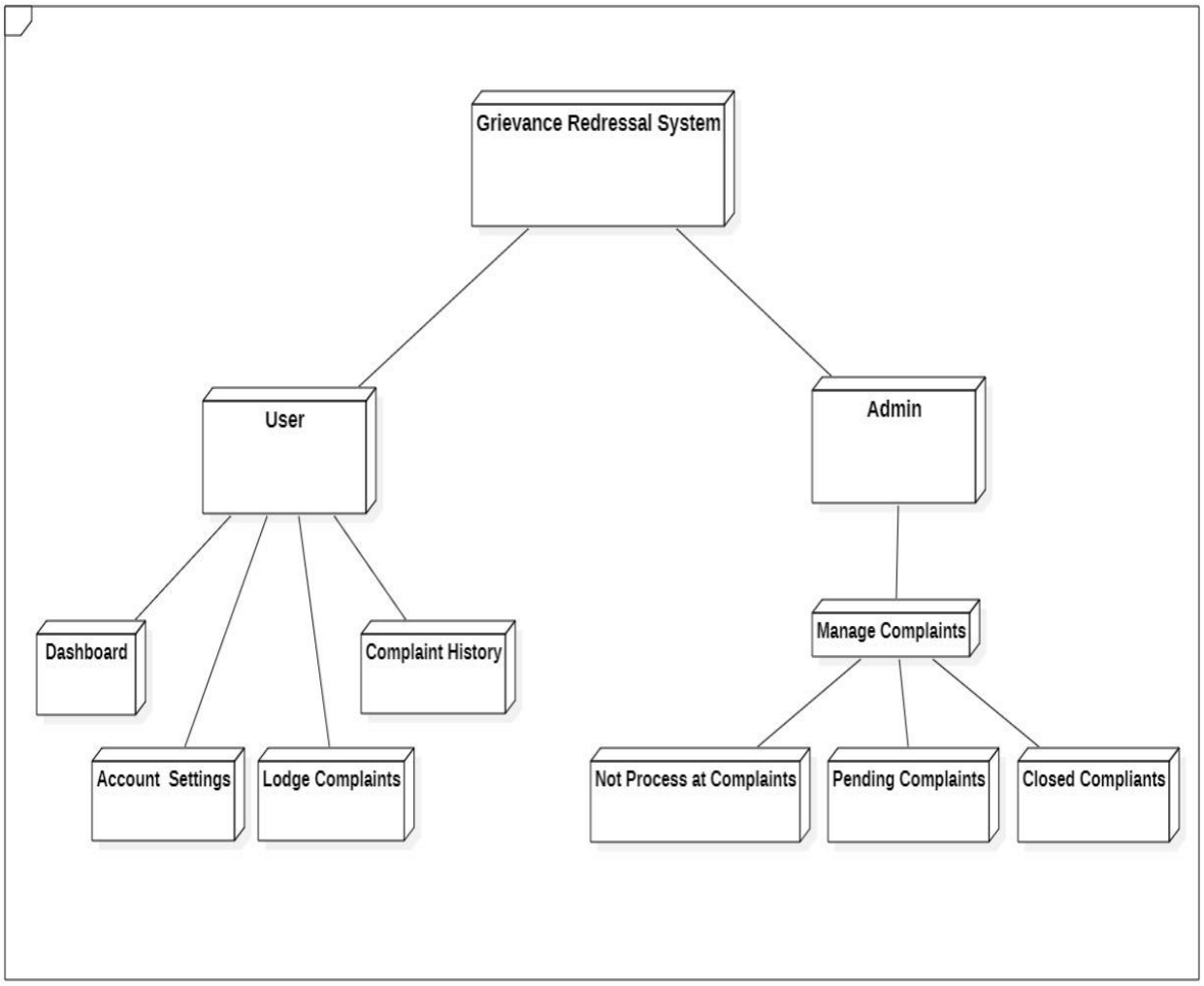


Fig. 5.5.3 Deployment Diagram for Grievance Cell

## 5.5.4 STATE CHART DIAGRAM

The name of the diagram itself clarifies the purpose of the diagram and other details. It describes different states of a component in a system. The states are specific to a component/object of a system. A State chart diagram describes a state machine. State machine can be defined as a machine which defines different states of an object, and these states are controlled by external or internal event. They define different states of an object during its lifetime and these states are changed by events. State chart diagrams are useful to model the reactive systems. Reactive systems can be defined as a system that responds to external or internal events. State chart diagram describes the flow of control from one state to another state. States are defined as a condition in which an object exists, and it changes when some event is triggered. The most important purpose of State chart diagram is to model lifetime of an object from creation to termination.

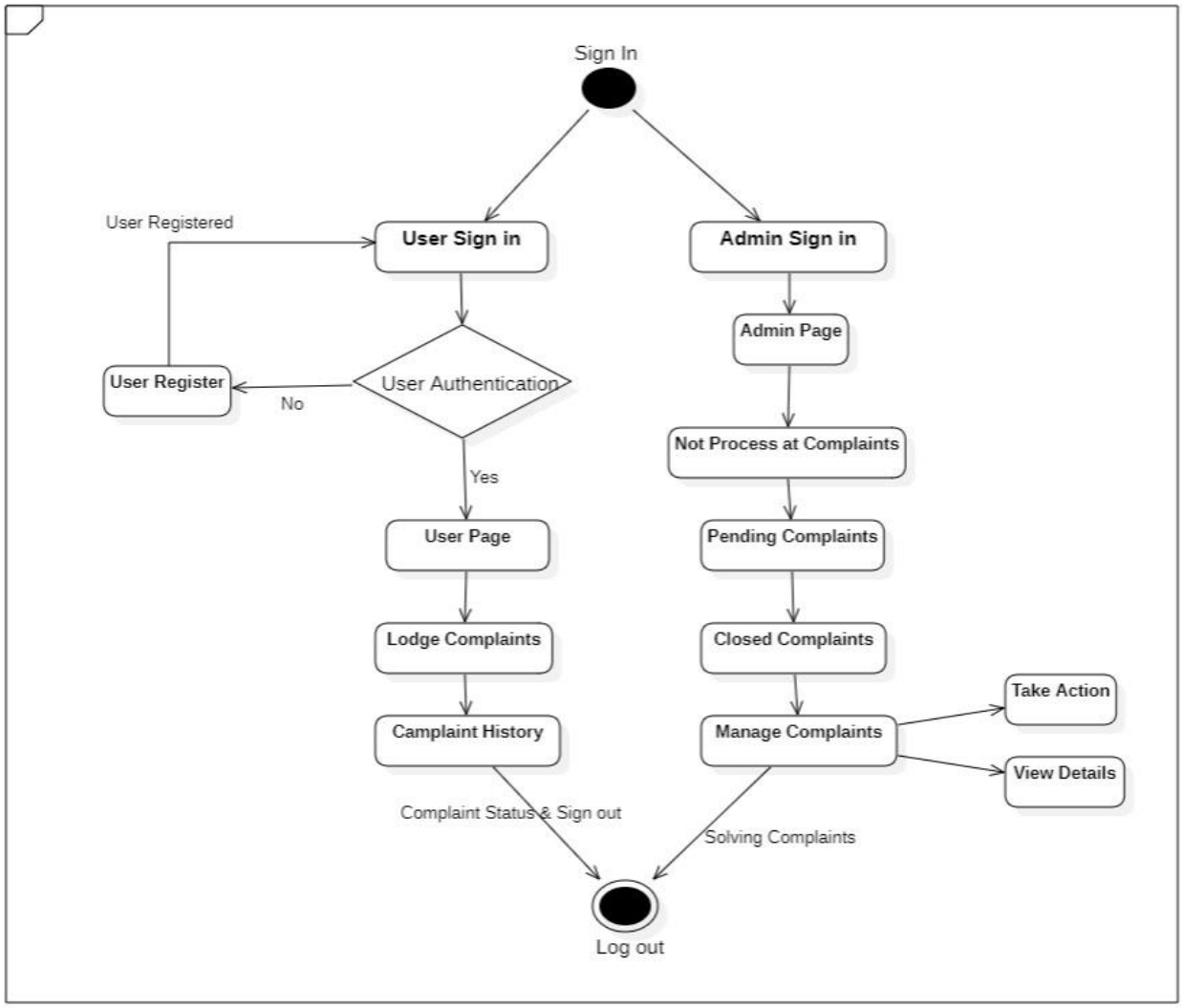


Fig. 5.5.4 State Chart Diagram for Grievance Cell

# 6.DESIGN METHODOLOGY

System architecture is the conceptual model that defines the structure, behavior, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that support reasoning about the structures and behaviors of the system.

- It is three-tier architecture.
- Students and cell members can access grievance services through the online application.

## PHP and My SQL

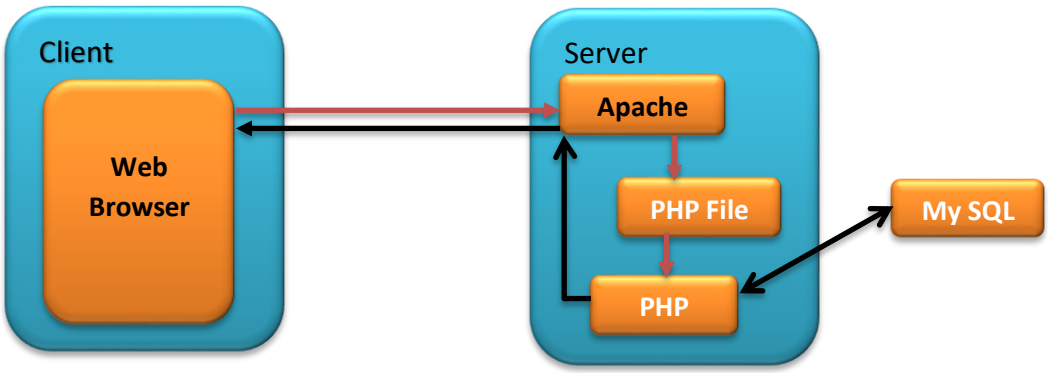


Fig. 6. Three Tier Architecture of Grievance Cell

## 6.1 DATABASE DESIGN

**DATABASE MANAGEMENT SYSTEM**: A database management system consists of a collection of interrelated data and a set of programs to access the data. The collection of data is usually referred to as the database. A database system is design to maintain large volumes of data. Management of data involves:
Definition the structure for the storage of data
Providing the mechanisms for the manipulation of the data
Providing for the security of the data against unauthorized access

**USERS OF THE DBMS**

Broadly there are three types of DBMS users:
The application programmer
The end user
The database administrator

The application programmer writes application that use the database. These programs operate on the data in the database. These operations include retrieving information, inserting data, deleting or changing data. The end user interacts with the system either by invoking an application program or by writing their queries in a database query language. The database query language allows the end user to perform all the basic operations on the data. The DBA has to coordinate the functions of collecting information about the data to be stored, designing and maintained to provide the right information at the right time to authorized people. These responsibilities belong to the DBA and his staff.

## 6.2 ADVANTAGES OF A DBMS

The major advantages that the database approach has over the conversation approach is that a database system provides centralized control of data. Most benefits accrue from this notion of centralized control

**REDUNDANCY CAN BE CONTROLLED**

Unlike the conventional approach, each application does not have to maintain its own data files. Centralized control of data by the DBA avoids unnecessary duplication of data and effectively reduces the total amount of data storage required. It also eliminates the extra processing necessary to trace the required data in a large mass of data present any redundancy that exist in the DBMS are controlled and the system ensure that these multiple copies are consistent.

**INCONSISTENCY CAN BE AVOIDED**

Since redundancy is reduced, inconsistency can also be avoided to some extent. The DBMS guarantee and that the database is never inconsistent, by ensuring that a change made to any entry automatically applies to the other entries as well. The process is known as propagating update.

**THE DATA CAN BE SHARED**

A database allows the sharing of data under its control by any number of application program or users. Sharing of data does not merely imply that existing applications can share the data the database, it also means that new applications can be developed to operate using the same database.

**STANDARDS CAN BE ENFORCED**

Since there is centralized control of data, the database administrator can ensure the standards are maintained in the representation of the stored data formats. This is particularly useful for data interchange, or migration of data between two systems.

**SECURITY RESTRICTIONS CAN BE APPLIED**

The DBMS guarantees that only authorized persons can access the database. The DBA defines the security checks to be carried out. Different checks can be applied to different operations on the same data. For instance, a person may have rights to query a file, but may not have the right to delete or update that file. The DBMS allows such security checks to be established for each piece of data in database.

**INTEGRITY CAN BE MAINTAINED**

Centralized control can also ensure that adequate checks are incorporated in the DBMS to provide data integrity. Data integrity means that the data contain in the database is both accurate and consistent. Inconsistency between two entries can lead to integrity problems.

**DATA INDEPENDENCE**

In non-database systems, the requirement of the application dictates the way in which the data is stored and the access techniques. Besides, the knowledge of organization of the data, the access techniques are built into the logic and code of the application. These systems are data dependent.

**DATA CATEGORISATION**

Data once gathered is grouped into separate categories depending on the database being created.

**DATA FIELD**

The fields or attributes of each table is determined after the data categorization. These include the columns that would hold the data to be stored. All data must be broken to its basic unit.

# 7.SYSTEM TESTING

## 7.1 FUNCTIONAL SPECIFICATION

Testing is the process of evaluating a system or its components with the intent to find that whether it satisfied the specified requirements or not. This activity results in the actual, expected and difference between their results. In simple words testing is executing a system to identify any gaps, errors or missing requirements in contrary to the actual desire or requirements. Testing can be defined as A process of analyzing a software item to detect the difference between existing and required conditions and to evaluate the features of the software item. In the IT industry, large companies have a team with responsibilities to evaluate the developed software in the context of the given requirements.

Moreover, developers also conduct testing which is called UNIT testing. In most cases, following professionals are involved in testing of a system within their respective capacities:

Software Tester

Software Developer

Project Lead/Manager

End User

## WHEN TO START TESTING

An early start to testing reduces the cost, time to rework and error free software that is delivered to the client. However, in Software Development Life Cycle (SDLC) testing can be started from the requirements gathering phase and lasts till the deployment of the software. However, it also depends on the development model that is being used. For example, in water fall model formal testing is conducted in the testing phase, but in incremental model, testing is performed at the end of every incremental/iteration and at the end of every incremental/iteration and at the end of whole application is tested. Testing is done in different forms at every phase of SDLC like during requirement gathering phase, the analysis and verifications of requirements are also considered testing. Reviewing the design in the design phase with intent to improve the design is also considered as testing. Testing performed by a developer on completion of the code is also categorized as UNIT type of testing.

## WHEN TO STOP TESTING

Unlike when to start testing it is difficult to determine when to stop testing, as testing is a never-ending process, and no one can say that any software is 100% tested. Following are the aspects which should be considered to stop the testing:

Testing Deadlines

Completion of test case execution

Completion of Functional and code coverage to a certain point

Bug rate falls below a certain level and no high priority bugs are identified

Management decision. Testing is the process of evaluating a system or its components with the intent to find that whether it satisfied requirements or not. This activity results in the actual, expected and difference between their results.

## 7.2 VERIFICATION AND VALIDATION

### VERIFIACTION

Are you building the system it right?

Ensure that the software system meets all the functionality.

Verification takes place first and includes the checking for documentation, code etc.

Done by Developers.

Have static activities as it includes the reviews, walkthroughs, and inspections to verify that software is correct or not.

It is an objective process, and no subjective decision should be needed to verify the software.

### VALIDATION

Are you building the right thing system?

Ensure that functionality meet the intended behavior

Validation occurs after verification and mainly involves the checking of the overall product.

Have dynamic activities as it includes executing the software against the requirements.

It is a subjective process and involves subjective decision on how well the software works.

## 7.3 SOFTWARE TESTING TYPES

The different types of testing which may be used to test software during SDLC are

**MANUAL TESTING**

This type includes the testing of the software manually i.e., without using any automated tool or any script. In this type of tester takes over the role of an end user and test the software to identify any un-expected behavior or bug. There are different stages for manual testing like unit testing, integration testing, system testing and user acceptance testing.

**UNIT TESTING**

Functional and reliability testing in an Engineering environment. Producing tests for the behavior of components of a product to ensure their correct behavior prior to system integration.

**INTEGRATION TESTING**

Testing in which modules are combined and tested as a group. Modules are typically code modules, individual applications, client, and server applications on a network etc. Integration Testing follows unit testing and process system testing.

**SYSTEM TESTING**

Testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black box testing, and as such, should require no knowledge of the inner design of the code or logic.

**ACCEPTANCE TESTING**

Testing to verify a product meets student specified requirements. A Student usually does this type of testing on a product that is developed externally. Testers use test plan, test cases or test scenarios to test the software to ensure the completeness of testing. Manual testing also includes exploratory testing as testers explore the software to identify errors in it.

**AUTOMATION TESTING**

Automation testing which is also known as test automation is when the tester writes scripts and uses another software to test the software. This process involves automation of a manual process. Automation Testing is used to re-turn the test scenarios that were performed manually, quickly, and repeatedly. Apart from regression testing, Automation testing is also used to test the application from load, performance, and stress point view. It increases the test coverage;improve accuracy, saves time and money in comparison to manual testing

## REGRESSION TESTING

Similar in scope to a functional test, a regression test allows a consistent, repeatable validation of each new release of a product or web site. Such testing ensures reported problems were introduced in the maintenance process. Though regression testing can be performed manually an automated test suite is often used to reduce the time and resources needed to perform the required testing.

## LOAD TESTING

Load testing is a generic term covering performance testing and stress testing.

## PERFORMANCE TESTING

Performance testing can be applied to understand your application or web site's scalability, or to benchmark the performance in an environment of third-party products such as servers and middleware for potential purchase. This sort of testing is particularly useful to identify performance bottlenecks in high use applications.

## STRESS TESTING

Testing conducted to evaluate a system or component at or beyond the limits of its specified requirements to determine the load under which it fails and how. A graceful degradation under load leading to non-catastrophic failure is the desired result. Often stress testing is performed using the same process as performance testing but employing a very high level of simulated load. It is not possible to automate everything in the software; however, the areas at which user can make transactions such as login form or registration forms etc., any area where large number of users can access the software simultaneously should be automated. Furthermore, all GUI items, connections with databases, field validations etc. can be efficiently tested by automating the manual process.

# 8.IMPLEMENTATION

## 8.1 SET UP INSTITUTIONS:

The resettlement plan defines the institutions that should be set up to deal with several types of grievances. The environmental management plan defines how and by whom environmental impacts are monitored. These institutions should have different levels, so that complainants can move to a higher level if they are not satisfied with the grievance redress suggested or provided, and they should be designed to provide appropriate responses to specific types of grievances. In extraordinarily complex and sensitive projects, for example, nine concerns are often raised at the early stages of project preparation questioning the project overall, its location, its impact, or its rationale. Trace-related grievances in the southern transport development project in the southern transport development project, for example, trace-related grievances were raised by families who were to lose their valuable assets as well as by activist groups and organizations such as environmentalists and human rights groups that were concerned with potential damage to ecological and social resources during the design stage. This type of grievance should not be managed at the field level but should be managed in a transparent manner and responded to professionally by the executing and aid agencies responsible for the project.

Grievance Redress Committees GRCs should be established at the division level to assure accessibility for APs. They are legitimized through a circular issued by the secretary of the ministry of highways. Except for me (i) grievances related to compensation for acquired land, (ii) issues related to road design and engineering aspects, and (iii) cases pending in courts, the GRC is mandated to deal with any other types of grievances arising at the community level. GRC meetings are held at the respective divisional secretariats, which are familiar and accessible to the APs. The members of the GRC include the assistant divisional secretary as chair, 10 the project manager or resettlement officer, the chair of the community mediation board, a member of a recognized nongovernment organization, and a community leader. A complainant has the right to appear in person, to be accompanied by a family member, and/or to request to be represented by a village elder.

# 9.SOURCE CODE

**ADMIN INDEX. PHP**

```
<?php
session_start();
error_reporting(0);
include("include/config.php");
if(isset($_POST['submit']))
{
        $username=$_POST['username'];
        $password=md5($_POST['password']);
$ret=mysqli_query($bd, "SELECT * FROM admin WHERE username='$username' and password='$password'");
$num=mysqli_fetch_array($ret);
if($num>0)
{
$extra="notprocess-complaint.php";//
$_SESSION['alogin']=$_POST['username'];
$_SESSION['id']=$num['id'];
$host=$_SERVER['HTTP_HOST'];
$uri=rtrim(dirname($_SERVER['PHP_SELF']),'/\\');
header("location:http://$host$uri/$extra");
exit();
}
else
{
```

```
$_SESSION['errmsg']="Invalid username or password";
$extra="index.php";
$host  = $_SERVER['HTTP_HOST'];
$uri  = rtrim(dirname($_SERVER['PHP_SELF']),'/\\');
header("location:http://$host$uri/$extra");
exit();
}
}
?>
<!DOCTYPE html>
<html lang="en">
<head>
        <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <title>CMS | Admin login</title>
        <link type="text/css" href="bootstrap/css/bootstrap.min.css" rel="stylesheet">
        <link type="text/css" href="bootstrap/css/bootstrap-responsive.min.css"
rel="stylesheet">
        <link type="text/css" href="css/theme.css" rel="stylesheet">
        <link type="text/css" href="images/icons/css/font-awesome.css"
rel="stylesheet">
        <link type="text/css"
href='http://fonts.googleapis.com/css?family=Open+Sans:400italic,600italic,400,600'
rel='stylesheet'>
</head>
<body>
        <div class="navbar navbar-fixed-top">
```

```
<div class="navbar-inner">
<div class="container">
<a class="btn btn-navbar" data-toggle="collapse" data-target=".navbar-inverse-collapse">
<i class="icon-reorder shaded"></i>
</a>
<a class="brand" href="index.html">
Admin
</a>
<div class="nav-collapse collapse navbar-inverse-collapse">
<ul class="nav pull-right">
<li><a href="/shyam/index.html">
Back to Portal
</a></li>
</ul>
</div> <!-- /.nav-collapse -->
</div>
</div><!-- /navbar-inner -->
</div><!-- /navbar -->
<div class="wrapper">
<div class="container">
<div class="row">
<div class="module module-login span4 offset4">
<form class="form-vertical" method="post">
<div class="module-head">
```

```
                    <h3>Sign In</h3>
                </div>
                <span style="color:red;" ><?php echo
htmlentities($_SESSION['errmsg']); ?><?php echo
htmlentities($_SESSION['errmsg']="");?></span>
                <div class="module-body">
                    <div class="control-group">
                    <div class="controls row-fluid">
                    <input class="span12" type="text"
id="inputEmail" name="username" placeholder="Username">
                        </div>
                    </div>
                    <div class="control-group">
                    <div class="controls row-fluid">
                <input class="span12" type="password"
id="inputPassword" name="password" placeholder="Password">
                        </div>
                    </div>
                </div>
                <div class="module-foot">
                    <div class="control-group">
                    <div class="controls clearfix">
                <button type="submit" class="btn btn-primary
pull-right" name="submit">Login</button>
                        </div>
                    </div>
                </div>
```

```
                                    </form>
                                </div>
                            </div>
                        </div>
                    </div><!--/.wrapper-->
                    <div class="footer">
                        <div class="container">
                            <b class="copyright">@Krishnna Universtiy</b> All rights
reserved.
                        </div>
                    </div>
                    <script src="scripts/jquery-1.9.1.min.js" type="text/javascript"></script>
                    <script src="scripts/jquery-ui-1.10.1.custom.min.js"
type="text/javascript"></script>
                    <script src="bootstrap/js/bootstrap.min.js" type="text/javascript"></script>
</body>
```

**USER INDEX.PHP**

```
<?php

session_start();

error_reporting(0);

include("includes/config.php");

if(isset($_POST['submit']))

{

$ret=mysqli_query($bd, "SELECT * FROM users WHERE
userEmail='".$_POST['username']."' and password='".md5($_POST['password'])."'");

$num=mysqli_fetch_array($ret);

if($num>0)

{

$extra="dashboard.php";//

$_SESSION['login']=$_POST['username'];

$_SESSION['id']=$num['id'];

$host=$_SERVER['HTTP_HOST'];

$uip=$_SERVER['REMOTE_ADDR'];

$status=1;

$log=mysqli_query($bd, "insert into userlog(uid,username,userip,status)
values('".$_SESSION['id']."','".$_SESSION['login']."','$uip','$status')");

$uri=rtrim(dirname($_SERVER['PHP_SELF']),'/\\');

header("location:http://$host$uri/$extra");

exit();

}

else

{
```

```
$_SESSION['login']=$_POST['username'];

$uip=$_SERVER['REMOTE_ADDR'];

$status=0;

mysqli_query($bd, "insert into userlog(username,userip,status)
values('".$_SESSION['login']."','$uip','$status')");

$errormsg="Invalid username or password";

$extra="login.php";

}}

if(isset($_POST['change']))

{

  $email=$_POST['email'];

   $contact=$_POST['contact'];

   $password=md5($_POST['password']);

$query=mysqli_query($bd, "SELECT * FROM users WHERE userEmail='$email' and
contactNo='$contact'");

$num=mysqli_fetch_array($query);

if($num>0)

{

mysqli_query($bd, "update users set password='$password' WHERE
userEmail='$email' and contactNo='$contact' ");

$msg="Password Changed Successfully";

}

else

{

$errormsg="Invalid email id or Contact no";

}}
```

```
?>
<!DOCTYPE html>
<html lang="en">
  <head>
    <link href="assets/font-awesome/css/font-awesome.css" rel="stylesheet" />
    <!-- Custom styles for this template -->
    <link href="assets/css/style.css" rel="stylesheet">
    <link href="assets/css/style-responsive.css" rel="stylesheet">
<script type="text/javascript">
function valid()
{
 if(document.forgot.password.value!= document.forgot.confirmpassword.value)
{
alert("Password and Confirm Password Field do not match  !!");
document.forgot.confirmpassword.focus();
return false;
}
return true;
}
</script>
  </head>
  <body>
            <div id="login-page">
                    <div class="container">
                        <form class="form-login" name="login" method="post">
```

```
                    <h2 class="form-login-heading">sign in now</h2>
                    <p style="padding-left:4%; padding-top:2%;  color:red">
                       <?php if($errormsg){
echo htmlentities($errormsg);
                                }?></p>
                    <p style="padding-left:4%; padding-top:2%;  color:green">
                    <?php if($msg){
echo htmlentities($msg);
                                }?></p>
                    <div class="login-wrap">
                       <input type="text" class="form-control" name="username"
placeholder="Email"  required autofocus>
                       <br>
                       <input type="password" class="form-control" name="password"
required placeholder="Password">
                       <label class="checkbox">
                          <span class="pull-right">
                             <a data-toggle="modal" href="login.html#myModal">
Forgot Password?</a>
                          </span>
                       </label>
                       <button class="btn btn-theme btn-block" name="submit"
type="submit"><i class="fa fa-lock"></i> SIGN IN</button>
                       <hr>
                    </form>
                    <div class="registration">
                       Don't have an account yet?
```

```
<br/>
                                <a class="" href="registration.php">
                                    Create an account
                                </a>
                            </div>
                        </div>
                        <!-- Modal -->
                        <form class="form-login" name="forgot" method="post">
                        <div aria-hidden="true" aria-labelledby="myModalLabel"
role="dialog" tabindex="-1" id="myModal" class="modal fade">
                            <div class="modal-dialog">
                                <div class="modal-content">
                                    <div class="modal-header">
                                        <button type="button" class="close" data-
dismiss="modal" aria-hidden="true">&times;</button>
                                        <h4 class="modal-title">Forgot Password ?</h4>
                                    </div>
                                    <div class="modal-body">
                                    <p>Enter your details below to reset your password.</p>
<input type="email" name="email" placeholder="Email" autocomplete="off"
class="form-control" required><br >
<input type="text" name="contact" placeholder="contact No" autocomplete="off"
class="form-control" required><br>
 <input type="password" class="form-control" placeholder="New Password"
id="password" name="password"  required ><br />
<input type="password" class="form-control unicase-form-control text-input"
placeholder="Confirm Password" id="confirmpassword" name="confirmpassword"
required >
```

```
                                    </div>
                                    <div class="modal-footer">
                                        <button data-dismiss="modal" class="btn btn-default"
type="button">Cancel</button>
                                        <button class="btn btn-theme" type="submit"
name="change" onclick="return valid();">Submit</button>
</div>
                                </div>
                            </div>
                        </div>
                        <!-- modal -->
                        </form>
                    </div>
                </div>
    <!-- js placed at the end of the document so the pages load faster -->
    <script src="assets/js/jquery.js"></script>
    <script src="assets/js/bootstrap.min.js"></script>
    <!--BACKSTRETCH-->
    <!-- You can use an image of whatever size. This script will stretch to fit in any
screen size.-->
    <script type="text/javascript" src="assets/js/jquery.backstretch.min.js"></script>
    <script>
        $.backstretch("../", {speed: 500});
    </script>
  </body>
</html>
```

**USER AUTHENTICATION**

```
<!DOCTYPE html>
<html>
  <head>
    <title>Phone Verification</title>
    <link href="https://fonts.googleapis.com/css?family=Roboto:300,400,500,700"
rel="stylesheet">
    <link href="main.css" rel="stylesheet">
  </head>
  <body>
    <form >
      <h1>Phone Verification</h1>
      <div class="formcontainer">
      <hr/>
      <div class="container">
        <label for="uname"><strong>Phone Number</strong></label>
        <input type="text" id="number" placeholder="Enter phone number"
name="uname" required>
      </div>
      <div id="recaptcha-container"></div>
      <button type="button" onclick="phoneAuth();">Send Otp</button>
    </form>
<form >
      <div class="formcontainer">
      <input type="text" id="verificationCode" placeholder="Enter verification code">
      </div>
```

```
    <button type="button" onclick="codeverify();">Verify code</button>
  </form>
  <script src="https://www.gstatic.com/firebasejs/8.3.1/firebase.js"></script>
  <script>
  // Your web app's Firebase configuration
  const firebaseConfig = {
  apiKey: "AIzaSyAaqovKT152FsMMwplLVNIwrPc7Mj1bUoI",
  authDomain: "connection-php-4614e.firebaseapp.com",
  projectId: "connection-php-4614e",
  storageBucket: "connection-php-4614e.appspot.com",
  messagingSenderId: "921432258357",
  appId: "1:921432258357:web:2d4bf86f1c7b3ef84b7e7c",
  measurementId: "G-HP5X36BCLN"
 };
  // Initialize Firebase
  firebase.initializeApp(firebaseConfig);
   firebase.analytics();
</script>
  <script src="firebase.js" type="text/javascript"></script>
 </body>
</html>
```

# 10. SCREENS & REPORTS

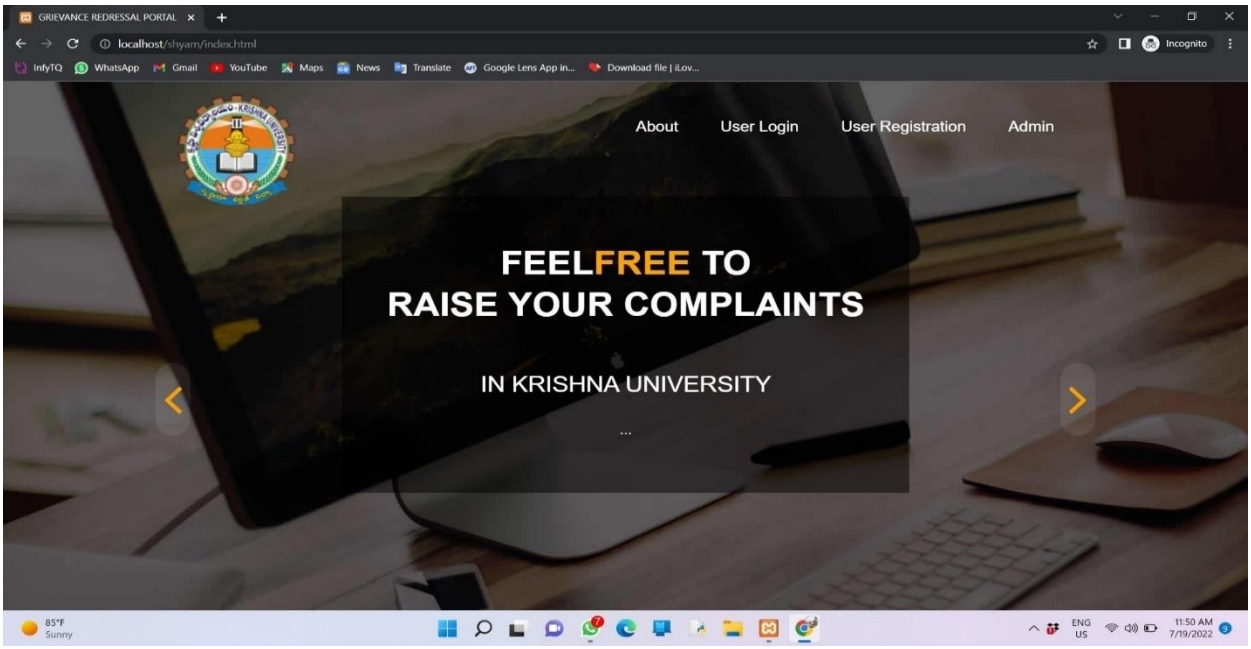


Fig. 10.1. Website Welcome Page for GRC

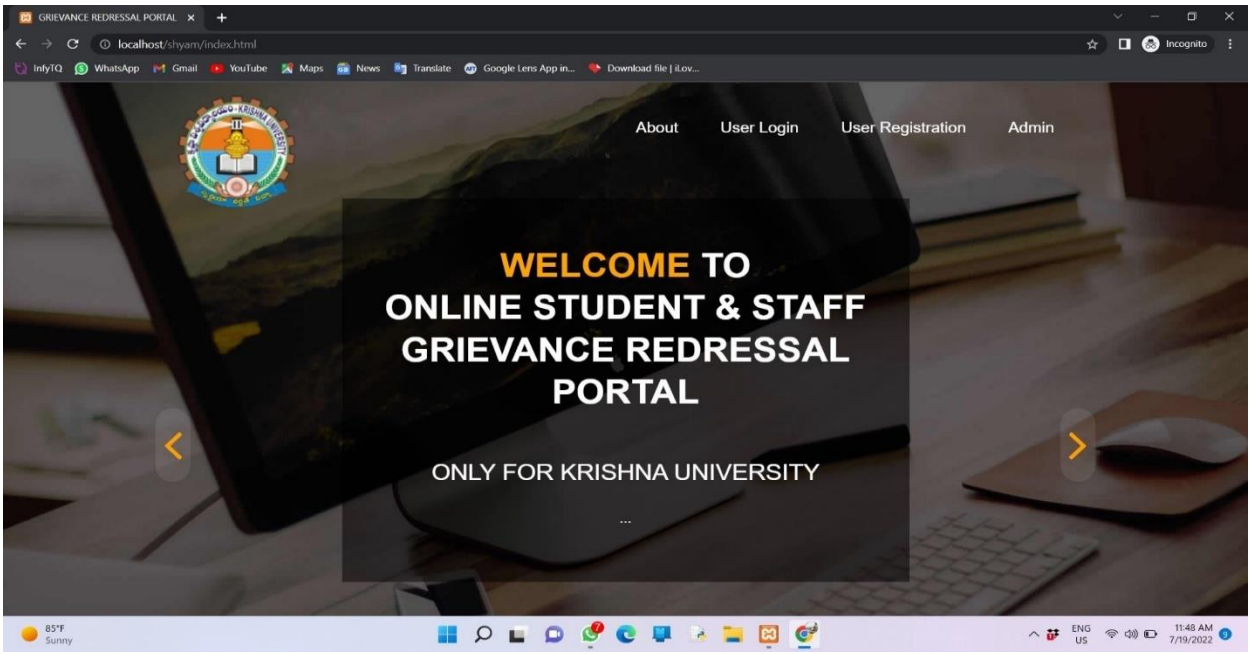


Fig. 10.2. Website Welcome Page for GRC

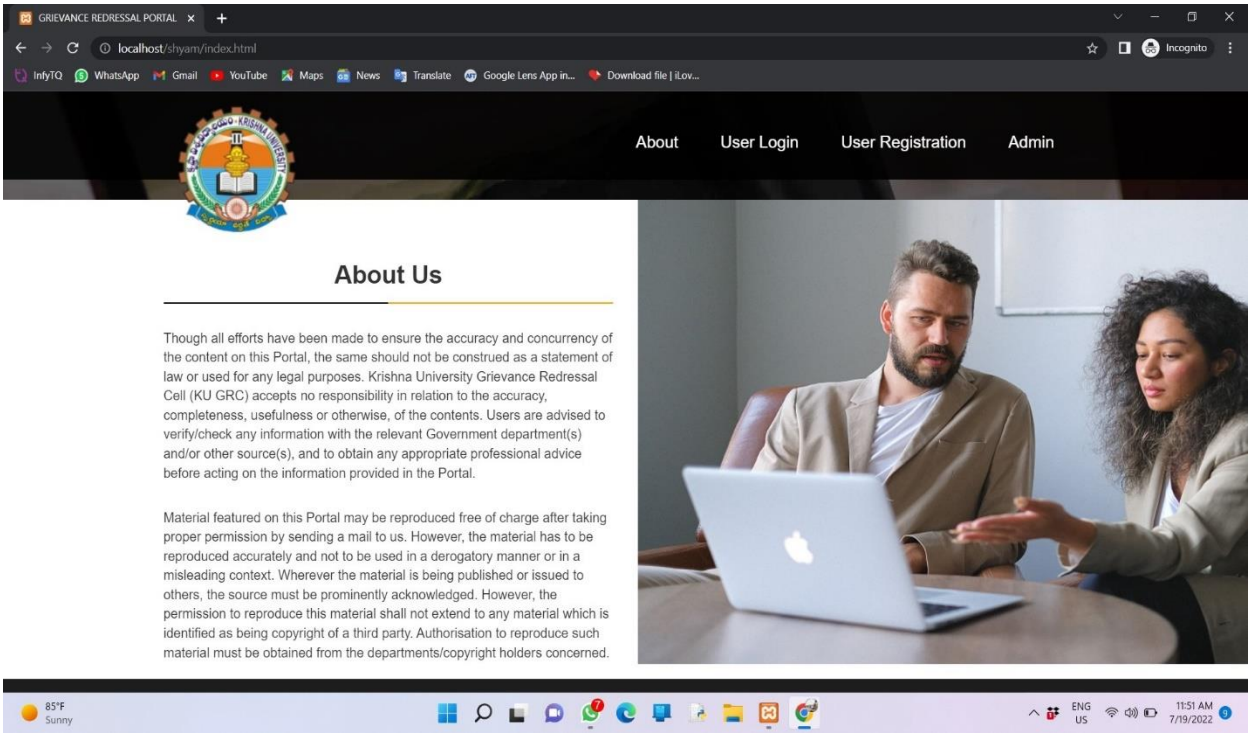


Fig. 10.3. About Us for Grievance Redressal Cell

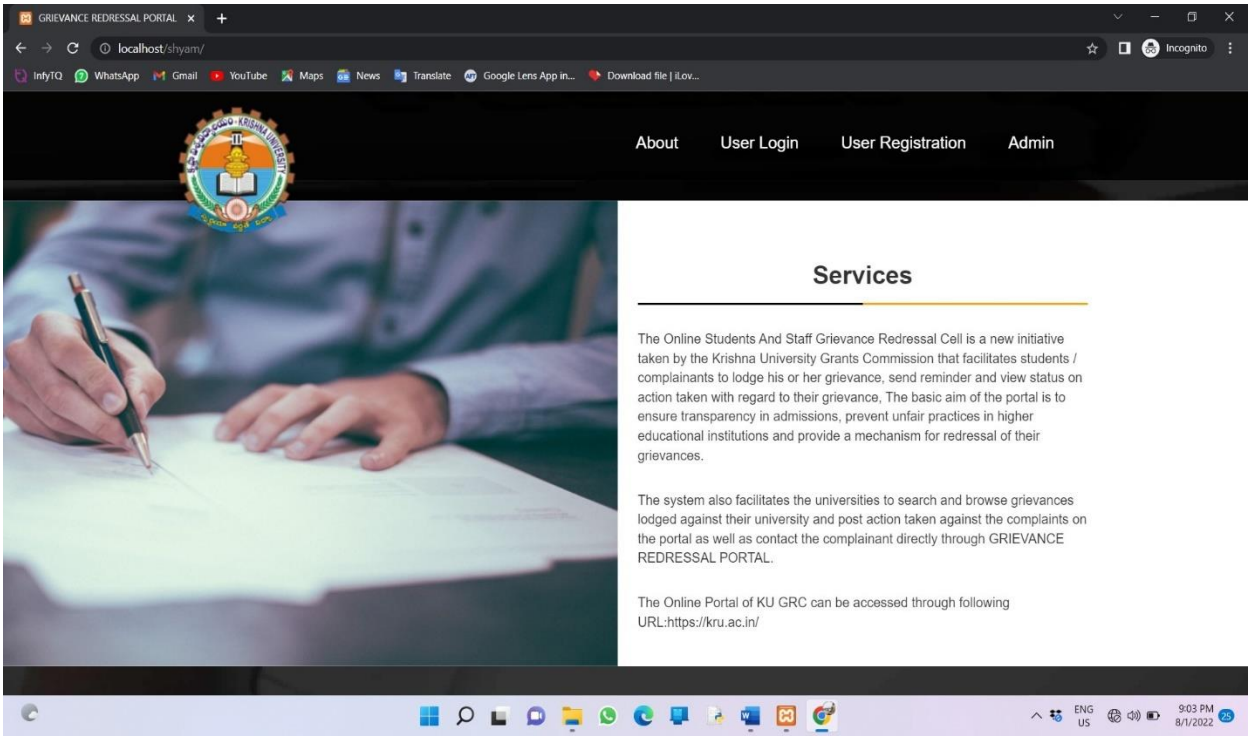


Fig. 10.4. Services for Grievance Redressal Cell

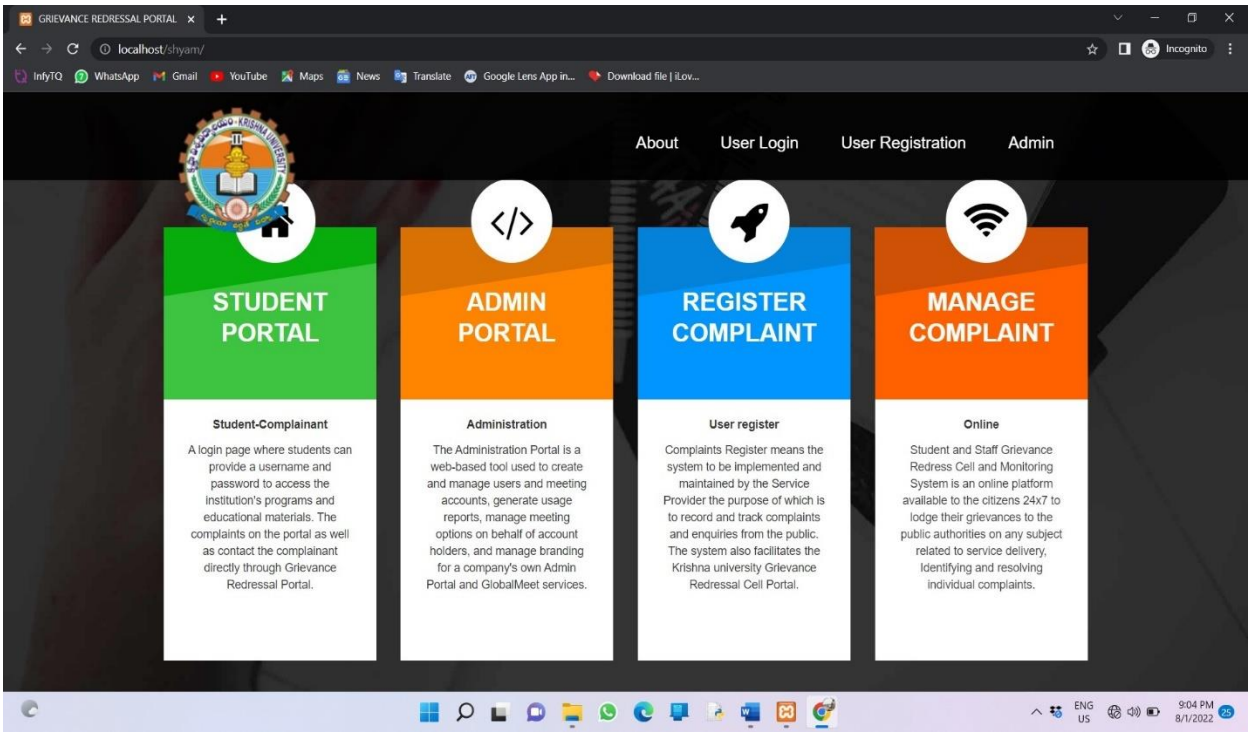


Fig. 10.5. Different Services for Grievance Redressal Cell

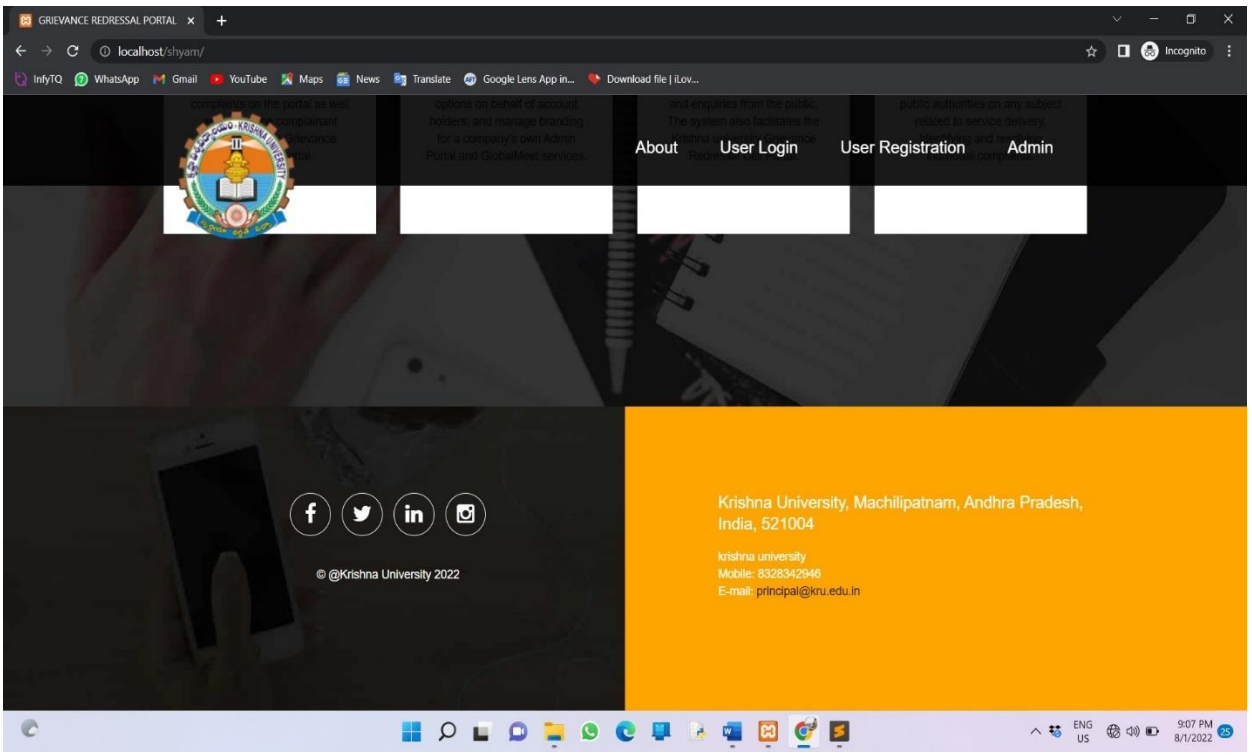


Fig. 10.6. Contact us for Grievance Redressal Cell

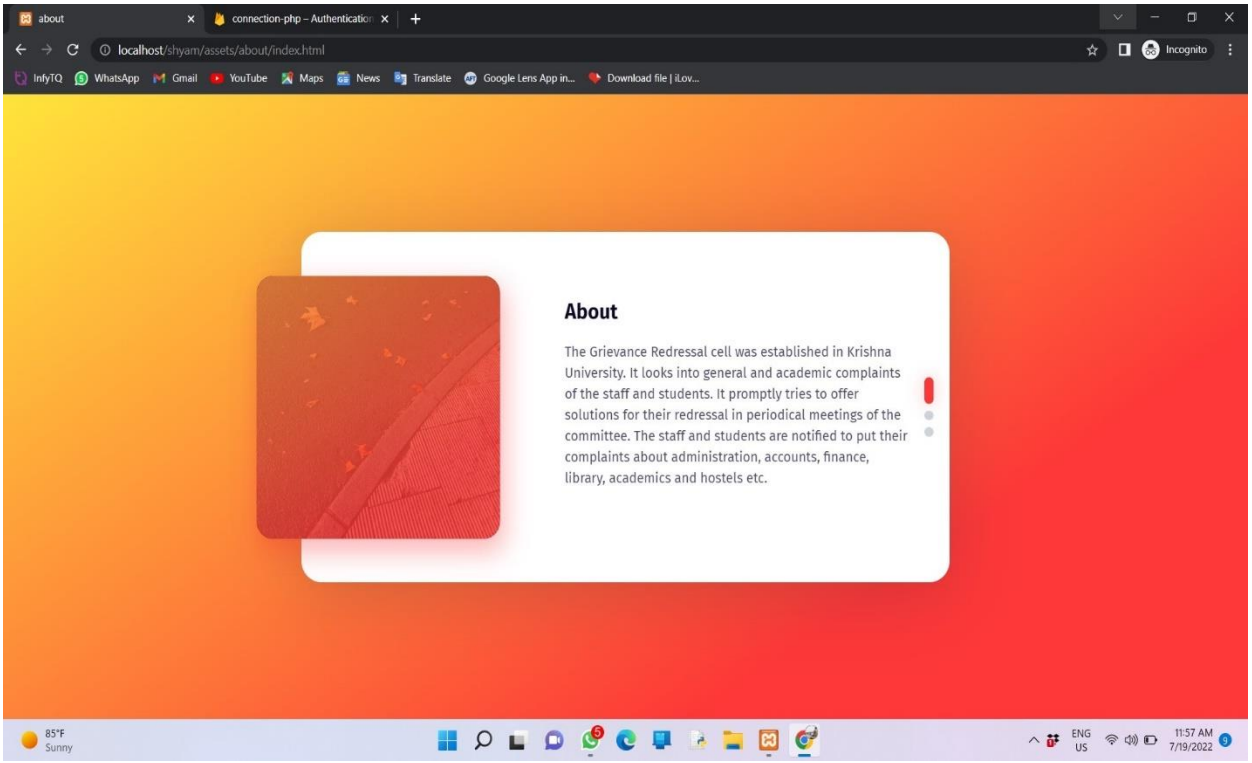


Fig. 10.7. About for Grievance Redressal Cell

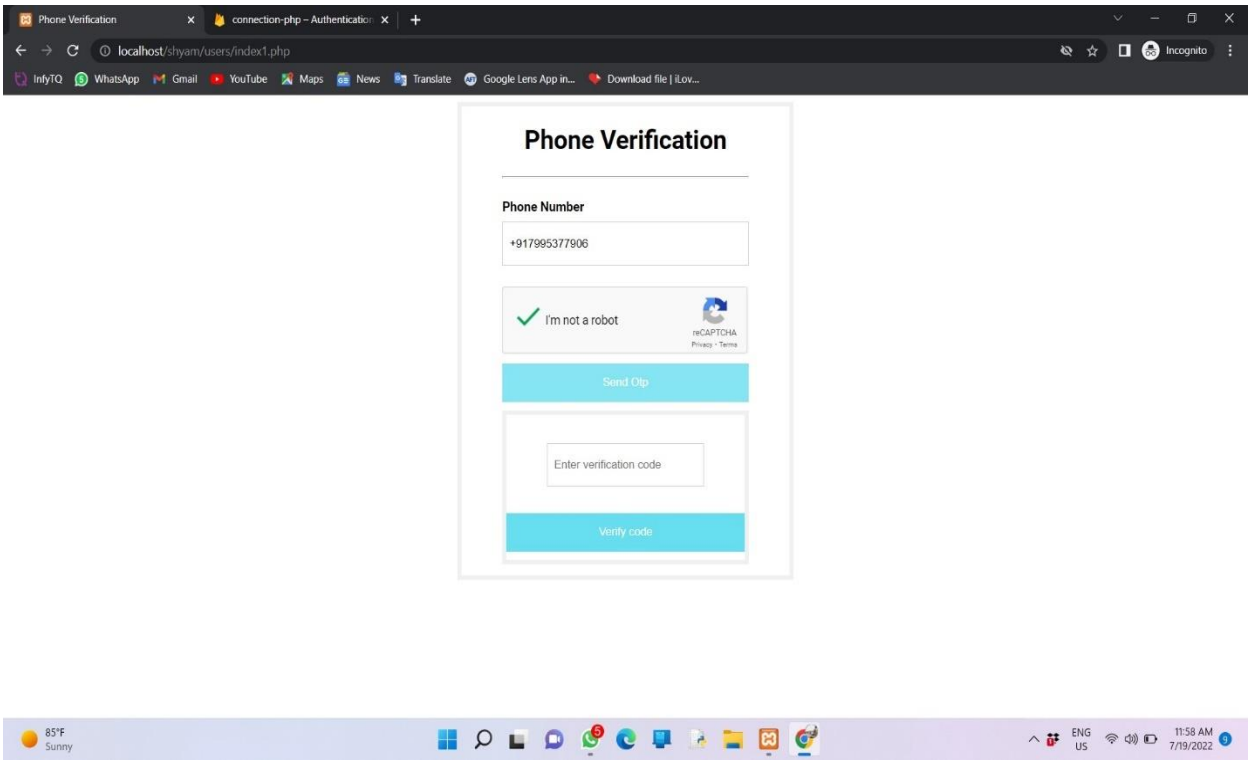


Fig. 10.8. User Authentication for Grievance Redressal Cell

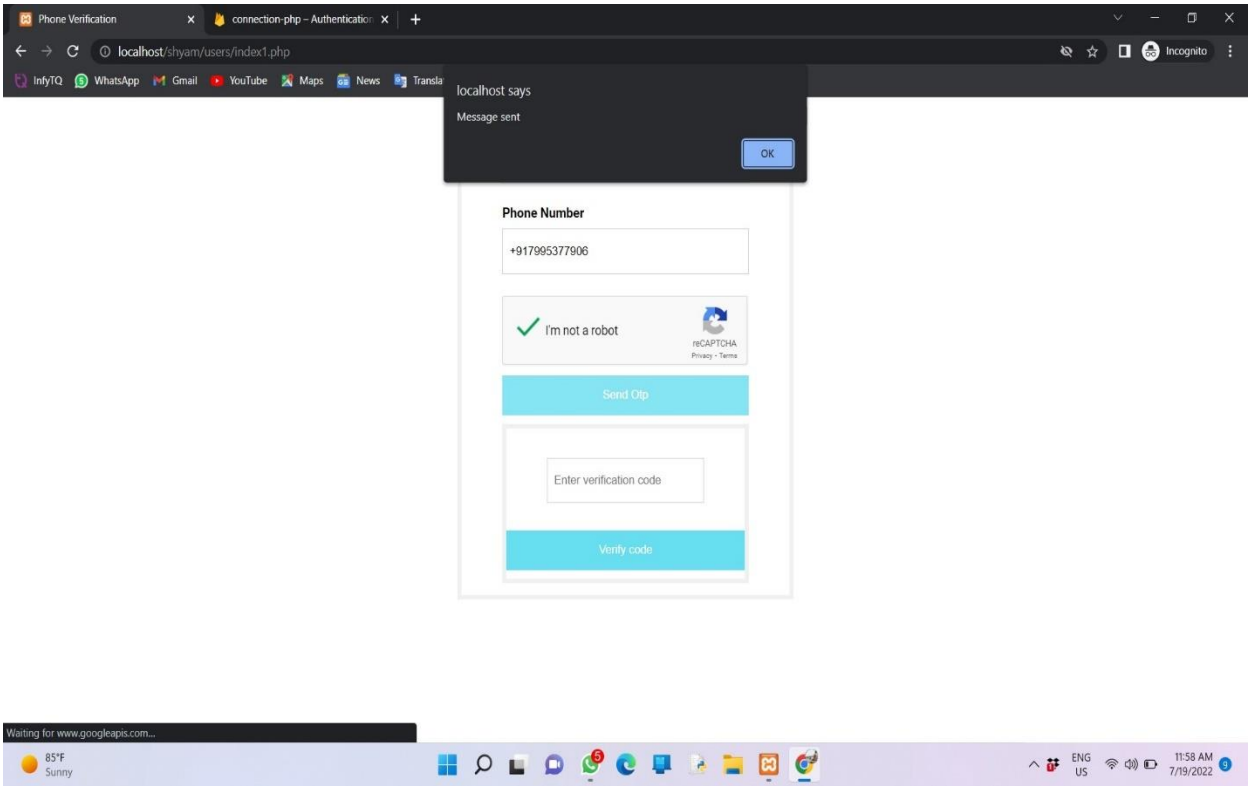


Fig. 10.9. Send OTP to Phone for User verification

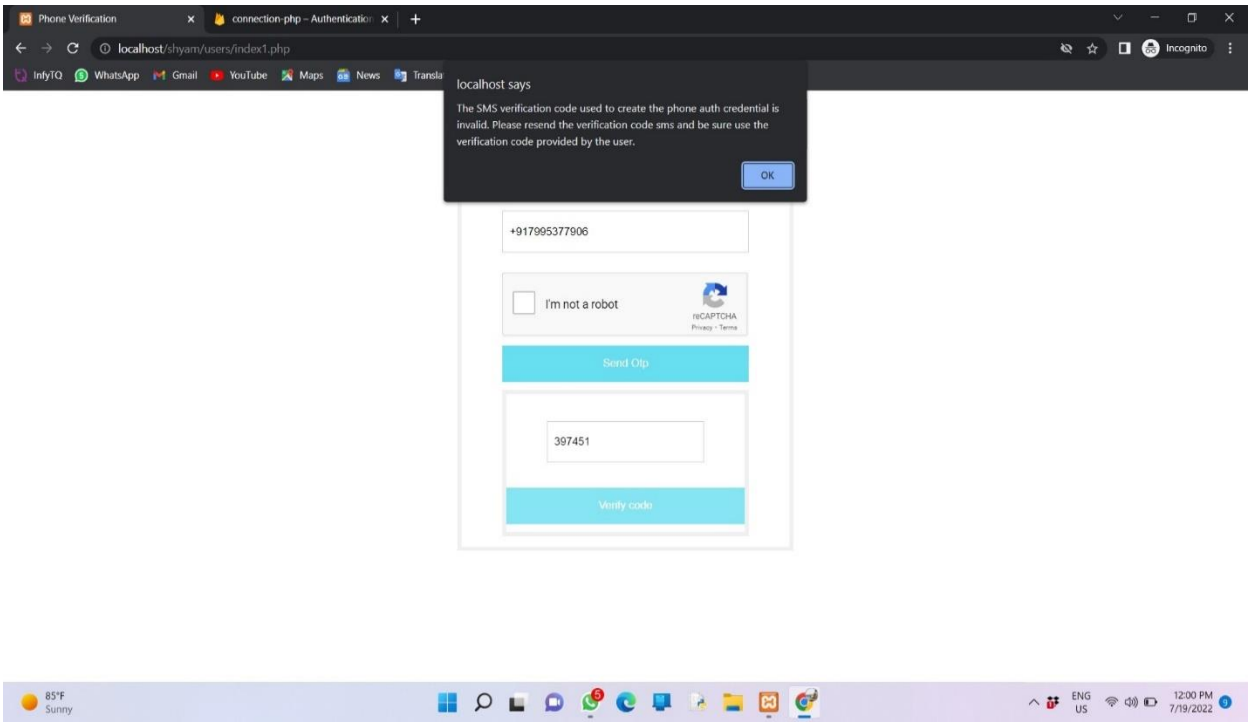


Fig. 10.10. Invalid OTP in User verification

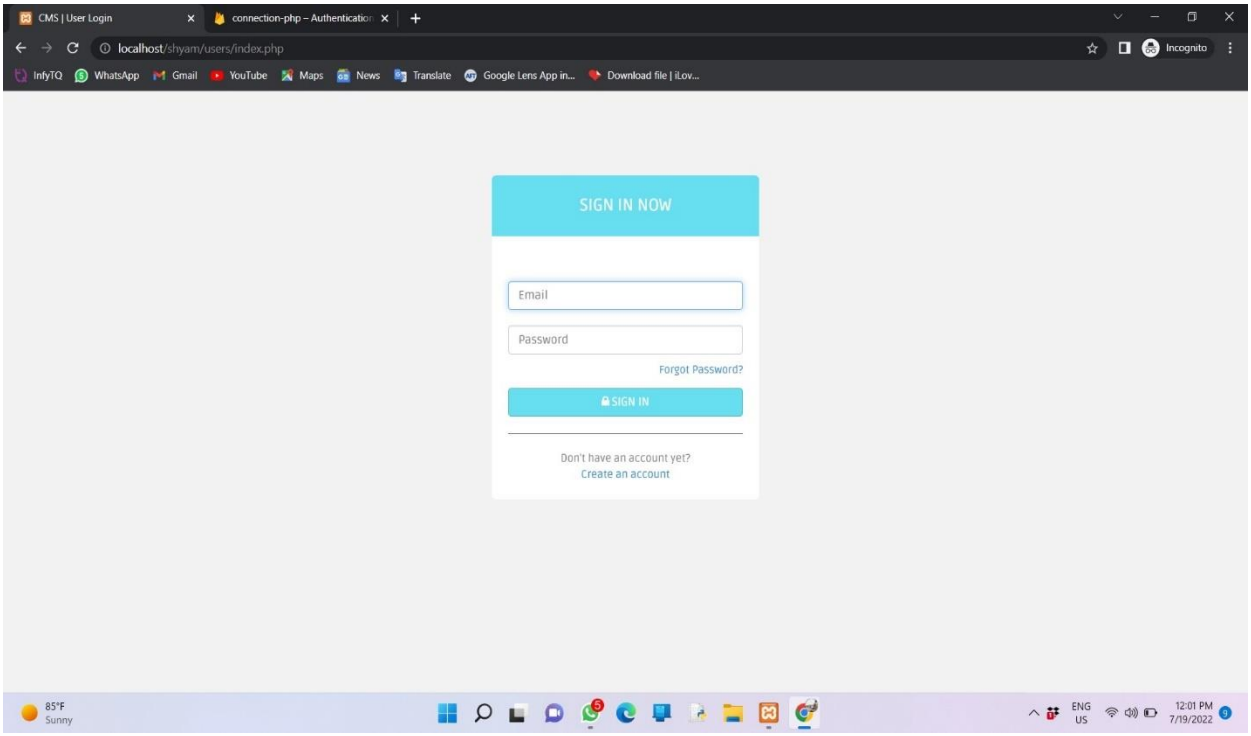


Fig. 10.11. Sign in for the User Portal in GRC

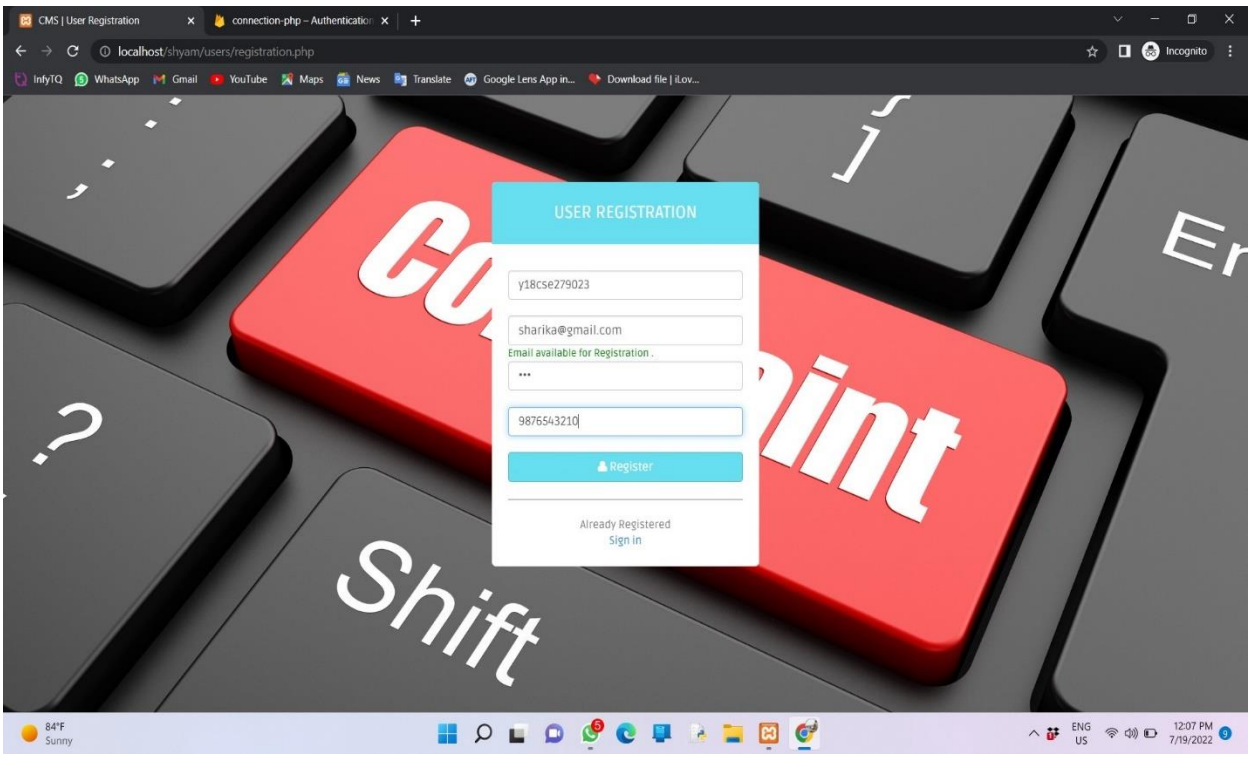


Fig. 10.12. User Registration for the User Portal in GRC

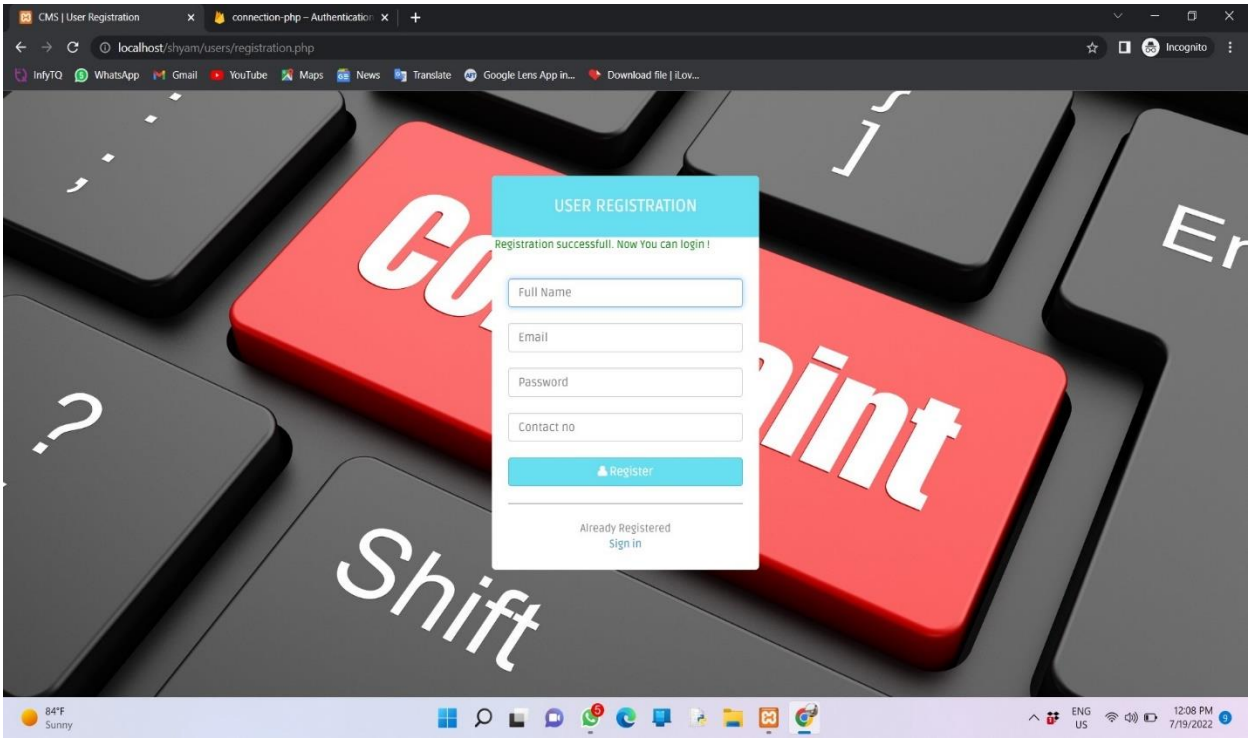


Fig. 10.13. Registration Successful for the User Portal in GRC

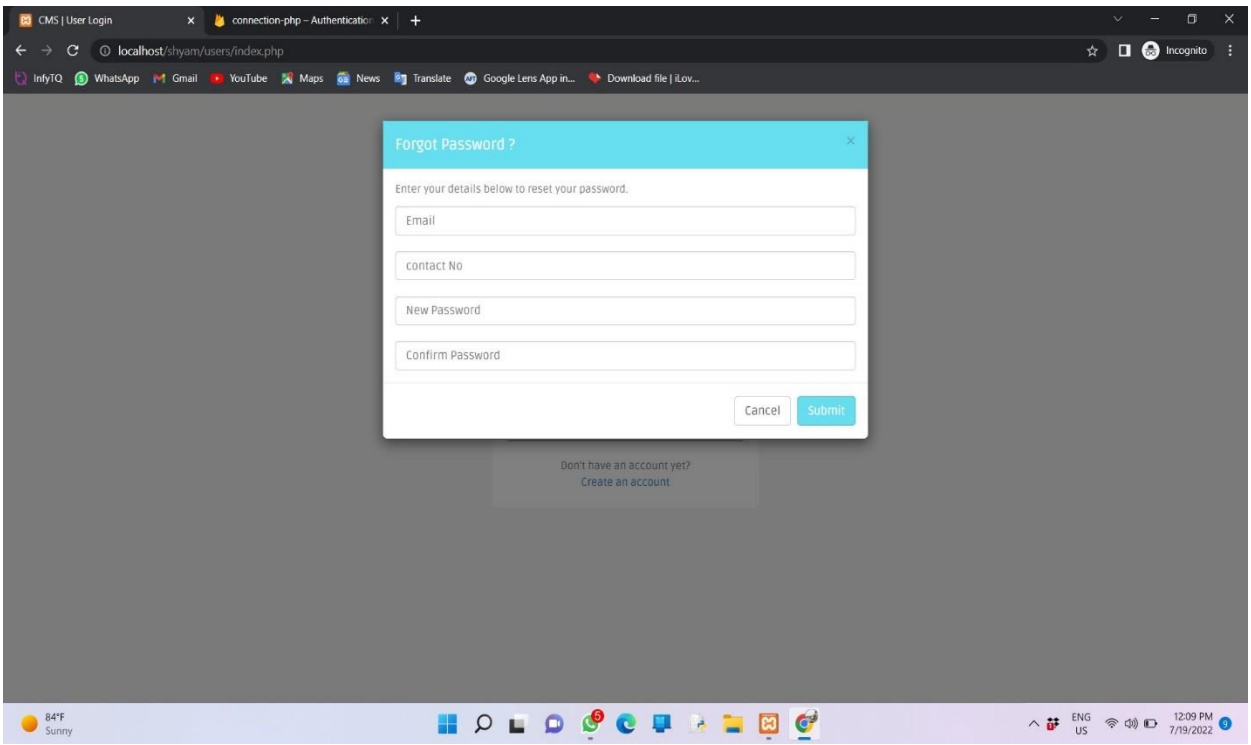


Fig. 10.14. Forgot Password feature for the User Portal in GRC

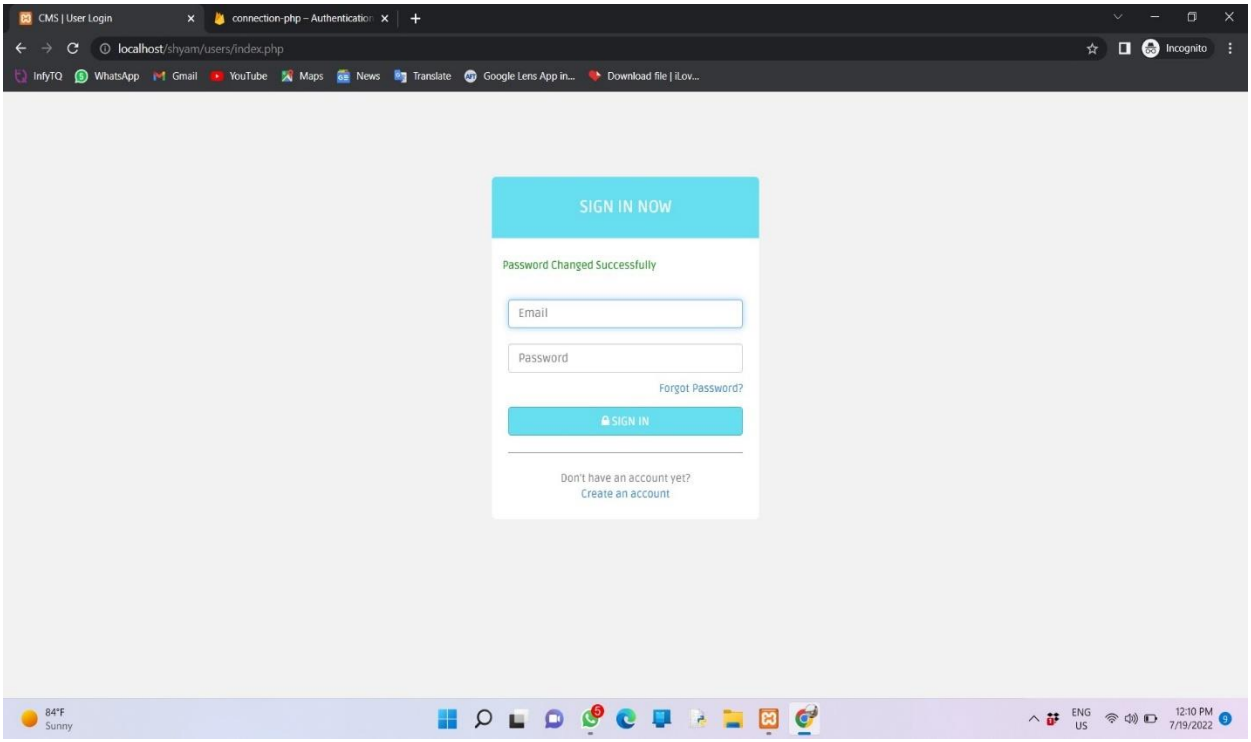


Fig. 10.15. Password Change Successfully for the User Portal in GRC

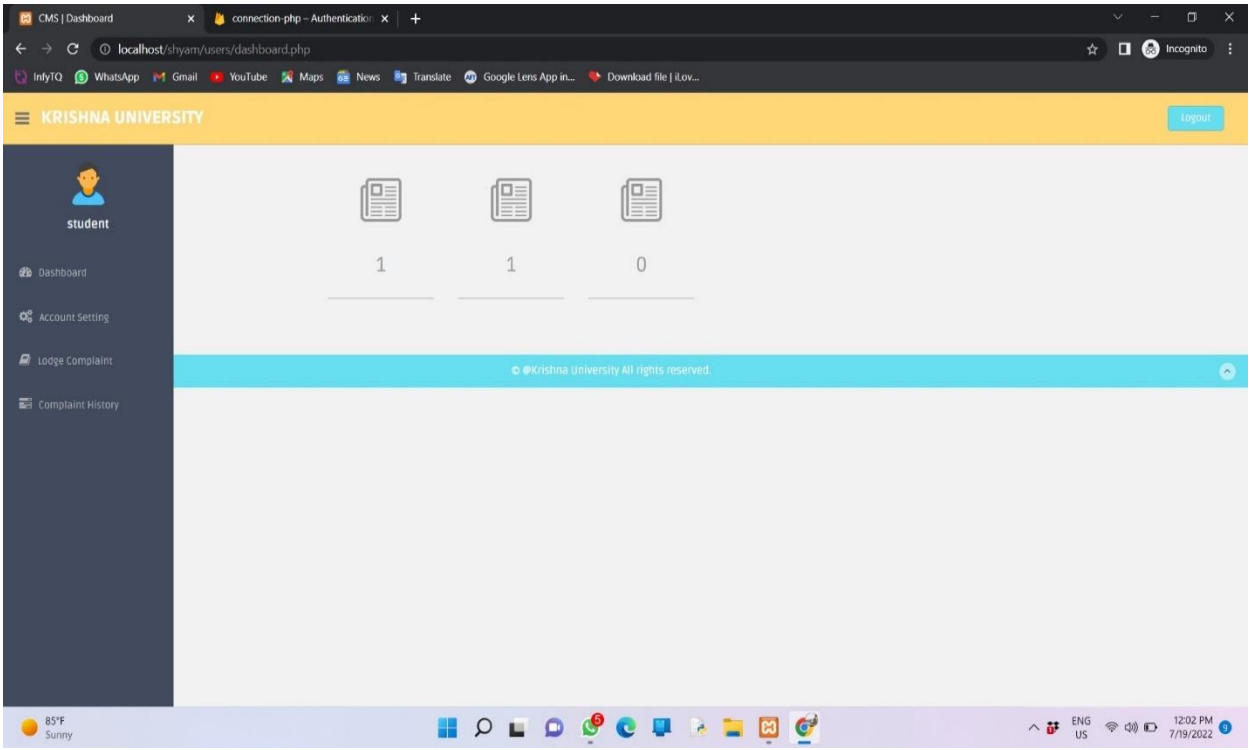


Fig. 10.16. User Dashboard for the User Portal in GRC

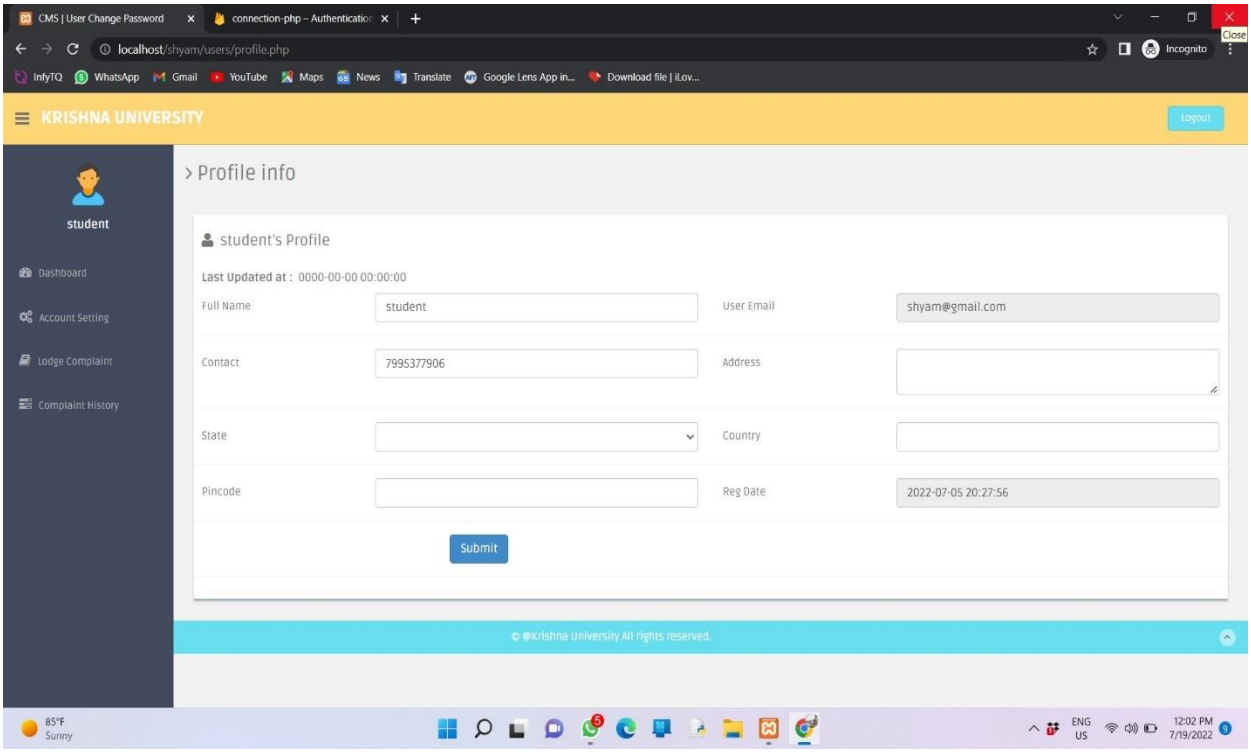


Fig. 10.17. User Profile for the User Portal in GRC

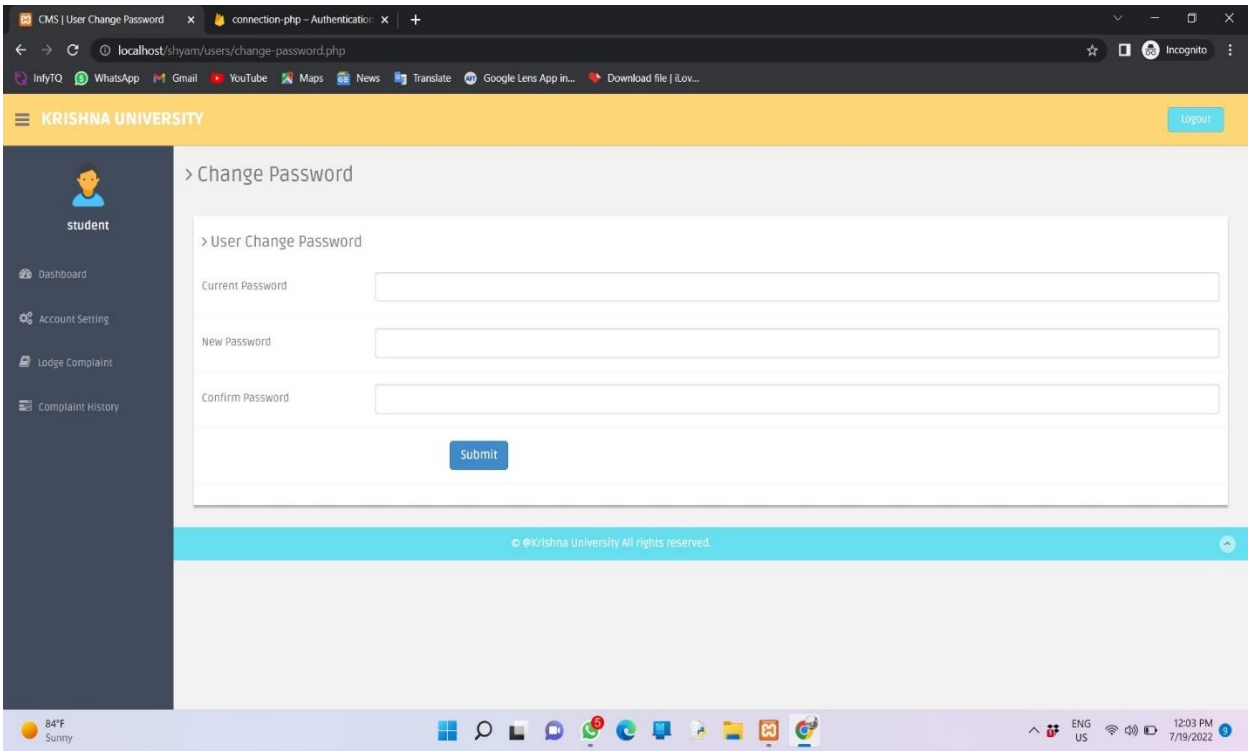


Fig. 10.18. Change Password for the User Portal in GRC

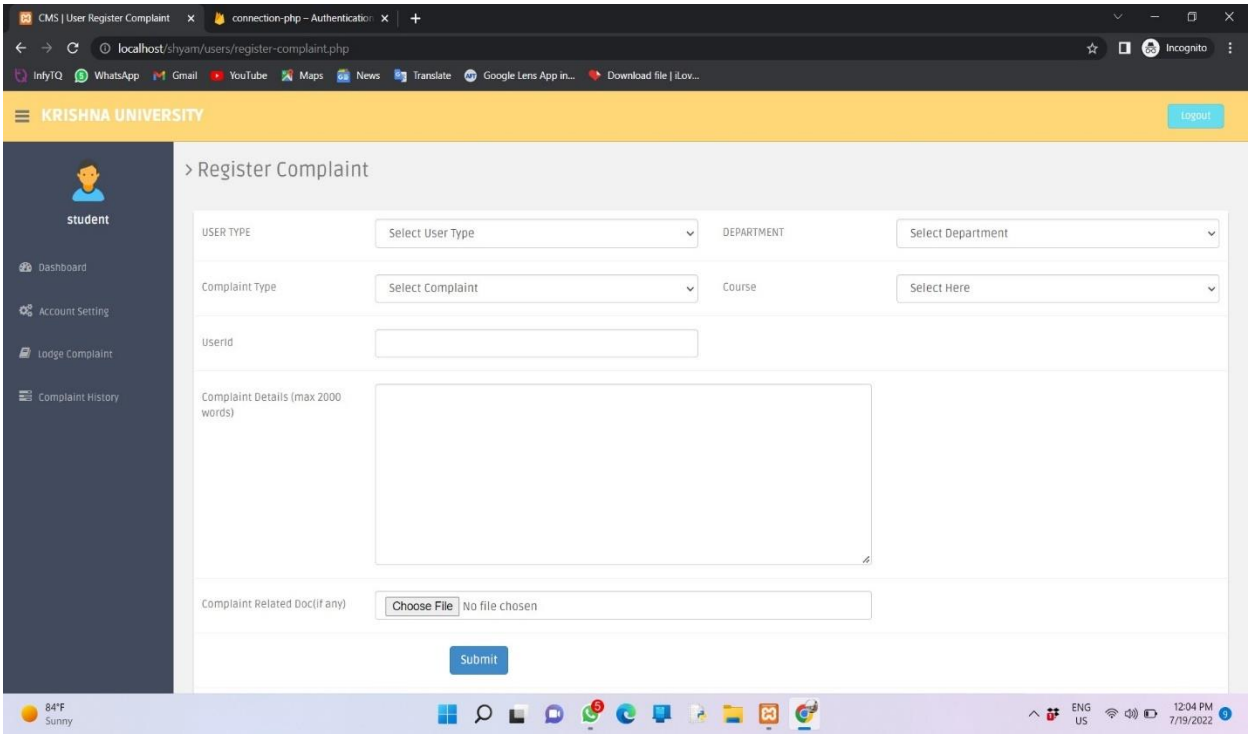


Fig. 10.19. Lodge Complaints for the User Portal in GRC

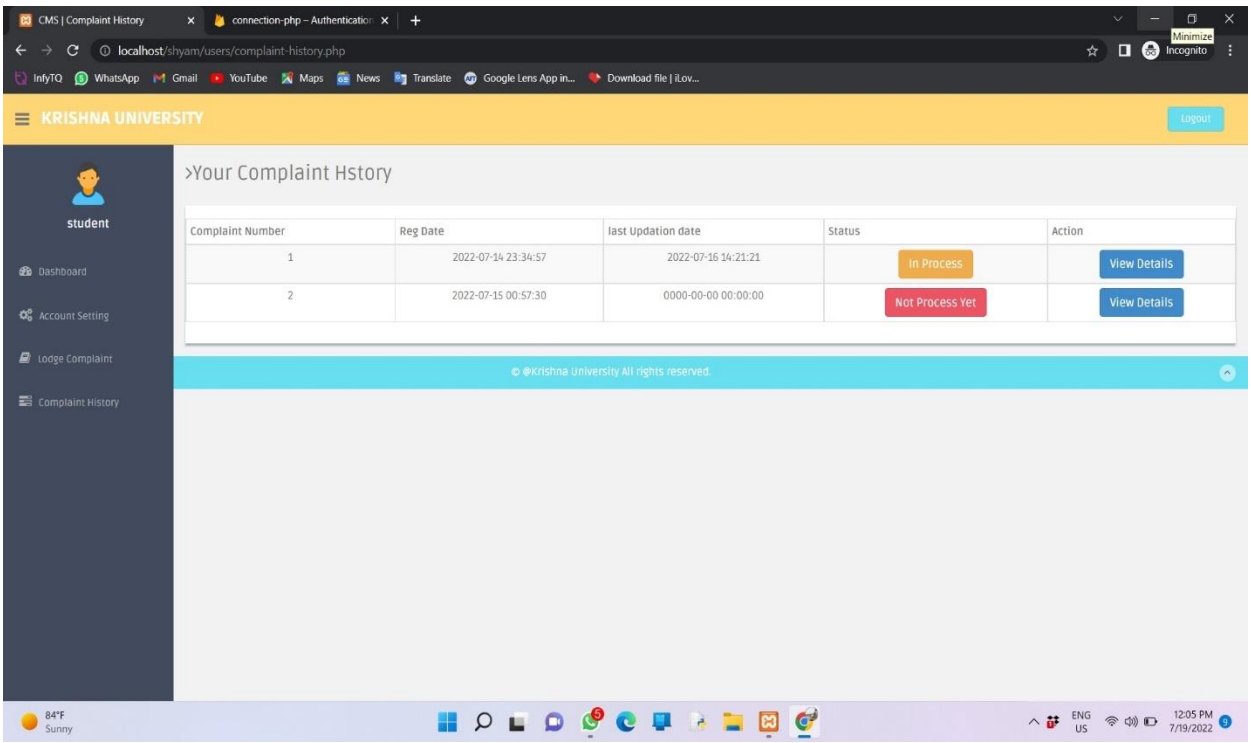


Fig. 10.20. Complaint History for the User Portal in GRC

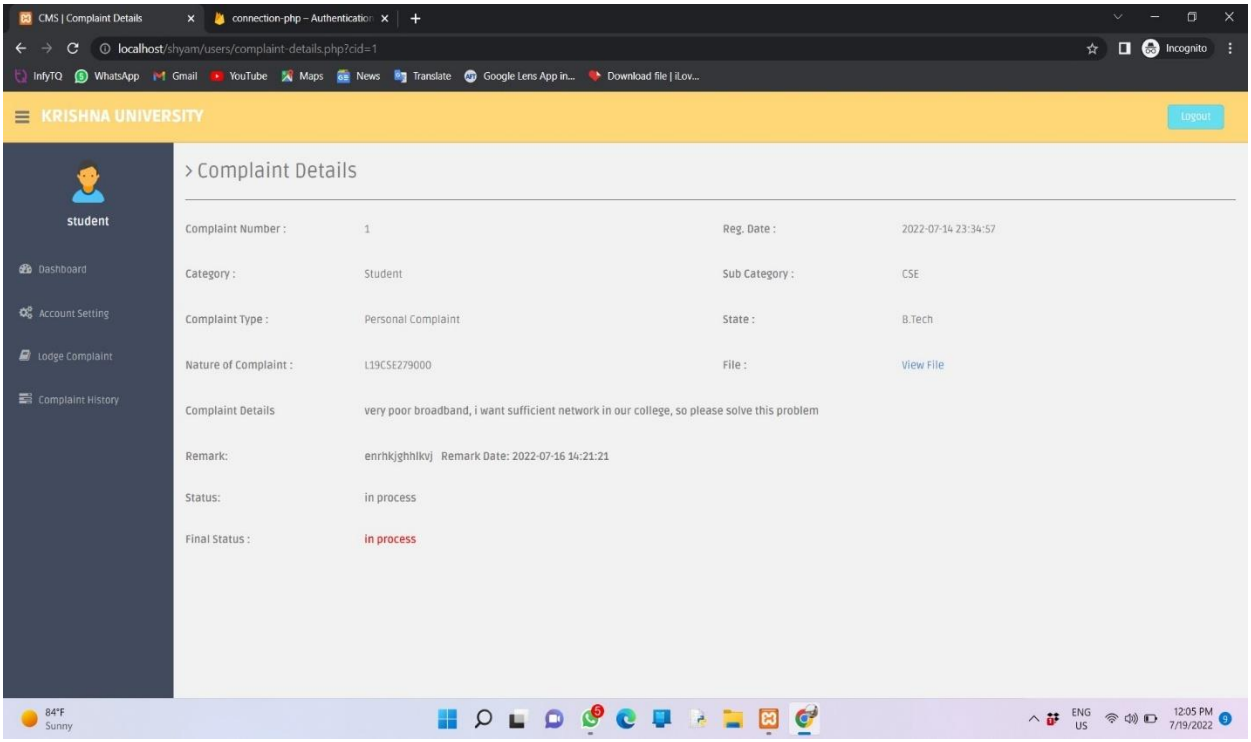


Fig. 10.21. Complaint Details for the User Portal in GRC

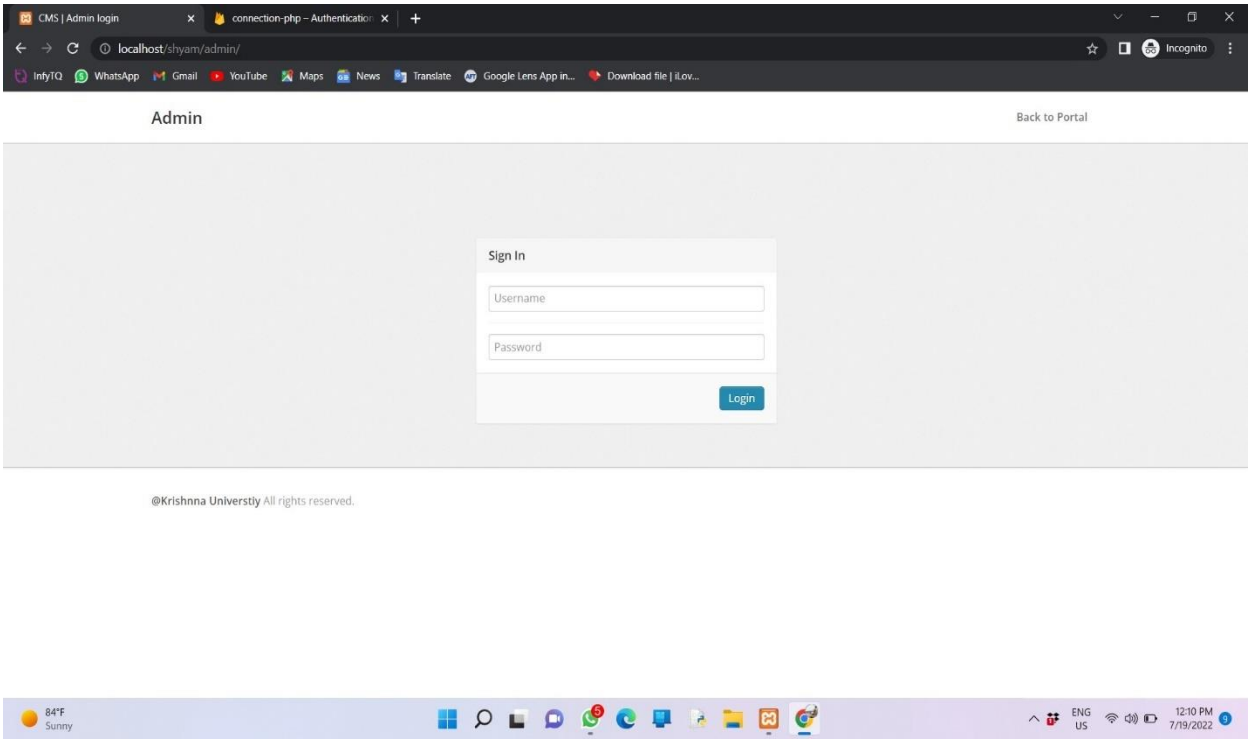


Fig. 12.22. Admin Login page for Grievance Redressal Cell

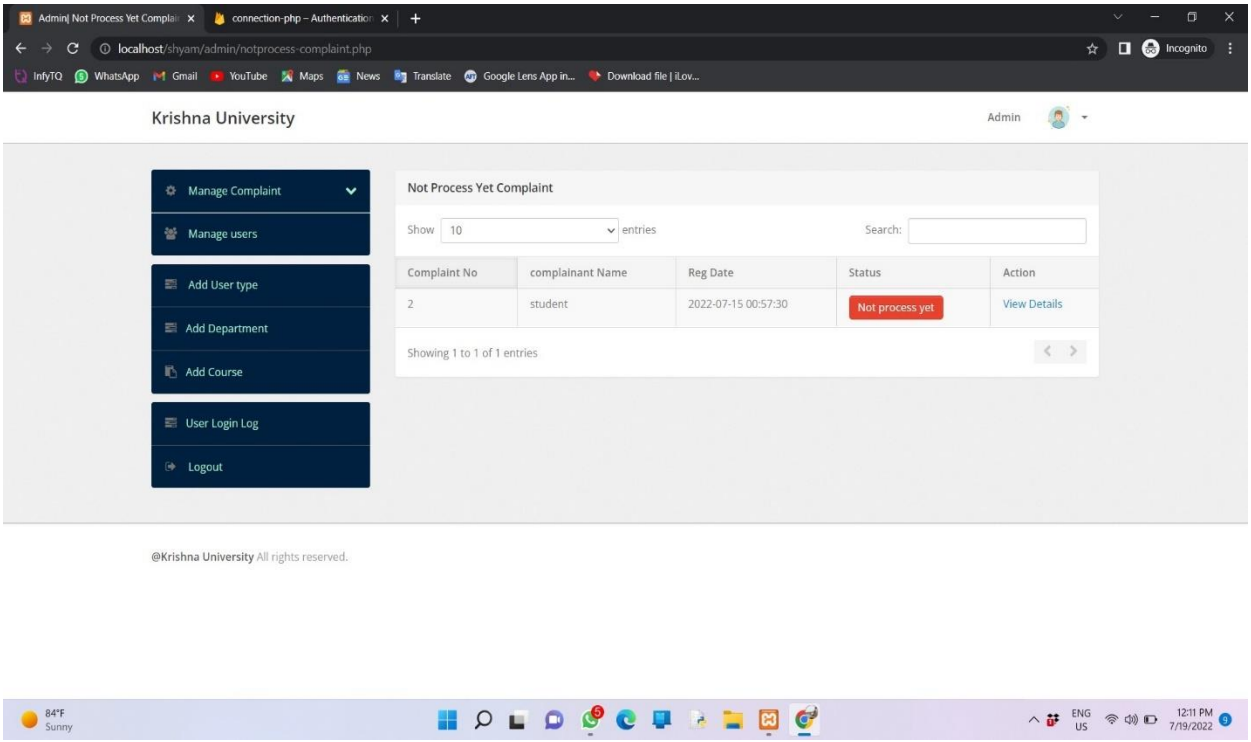


Fig. 10.23. Admin Portal for Grievance Redressal Cell

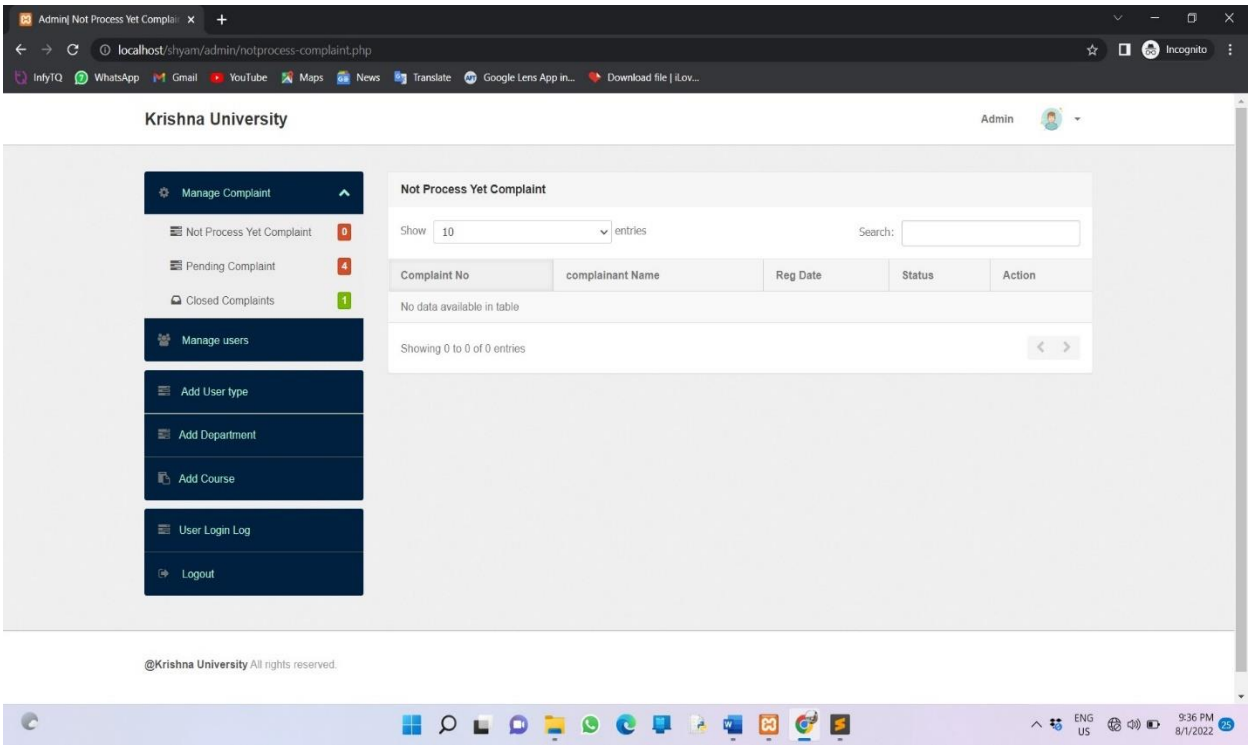


Fig. 10.24. Dashboard for Admin Portal in GRC

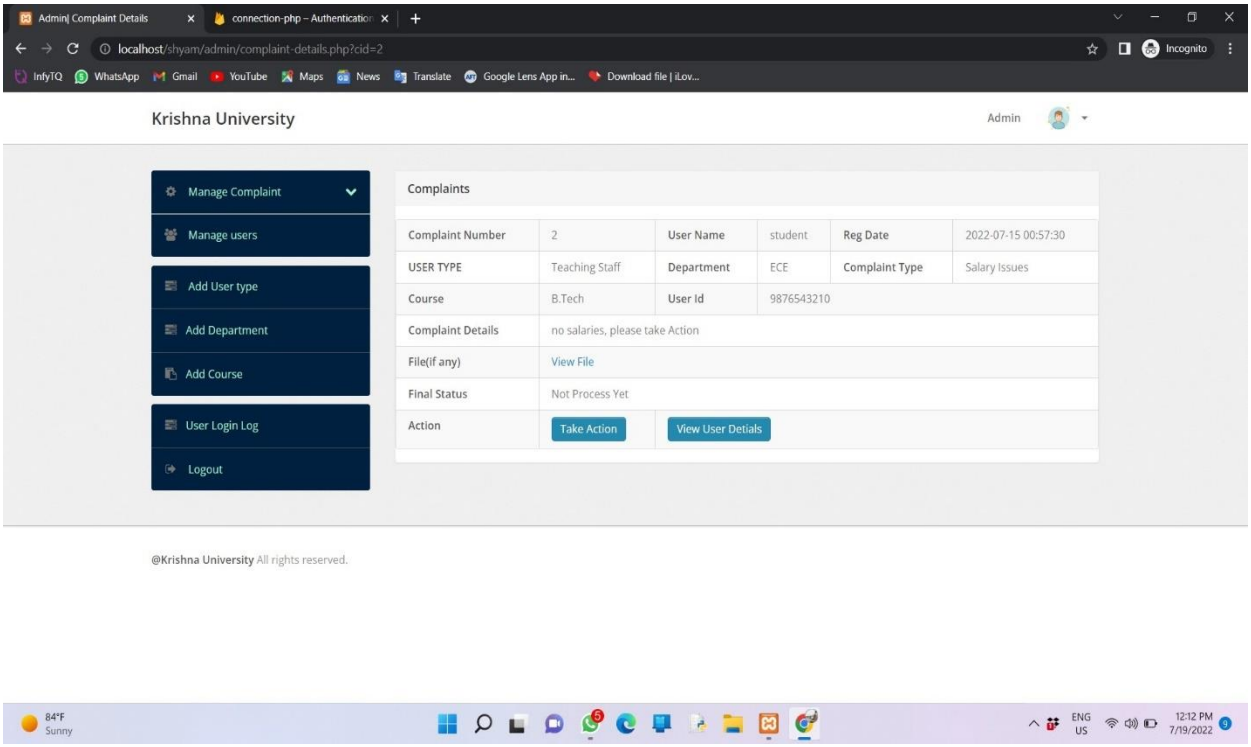


Fig. 10.25. Manage Complaints in Admin Portal in GRC

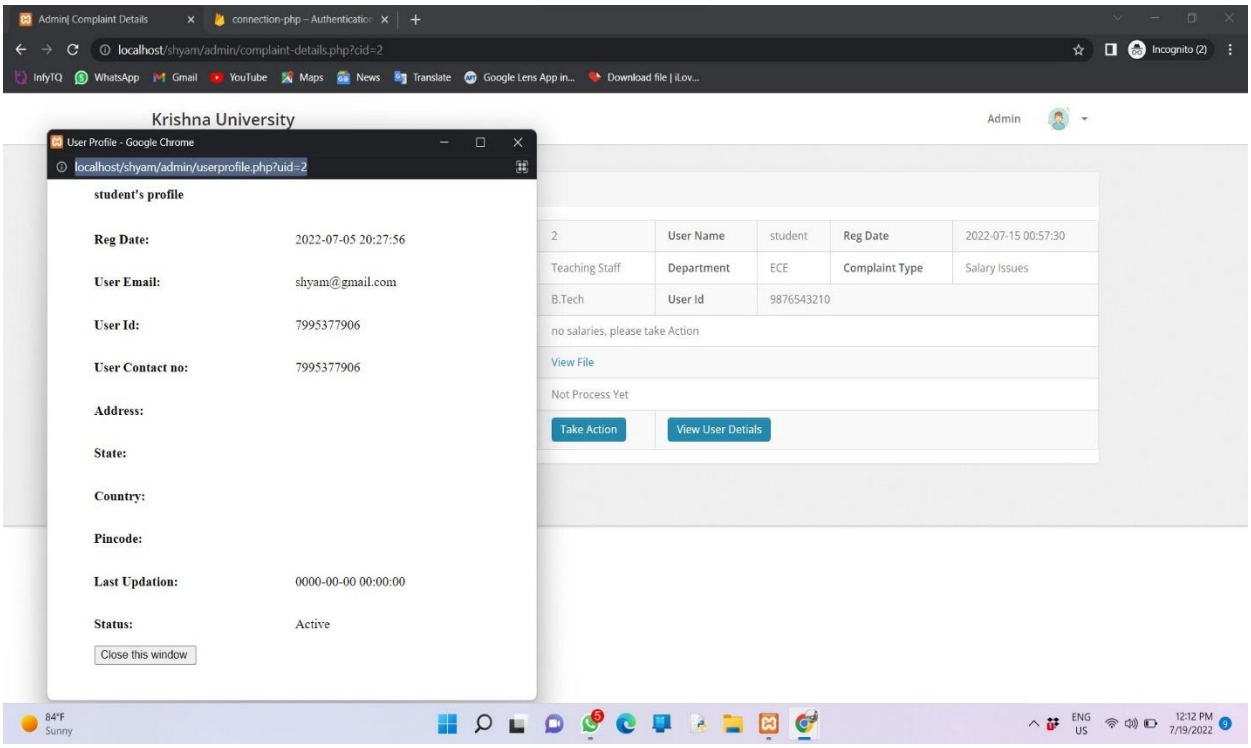


Fig. 10.26. View User Details in Admin Portal in GRC

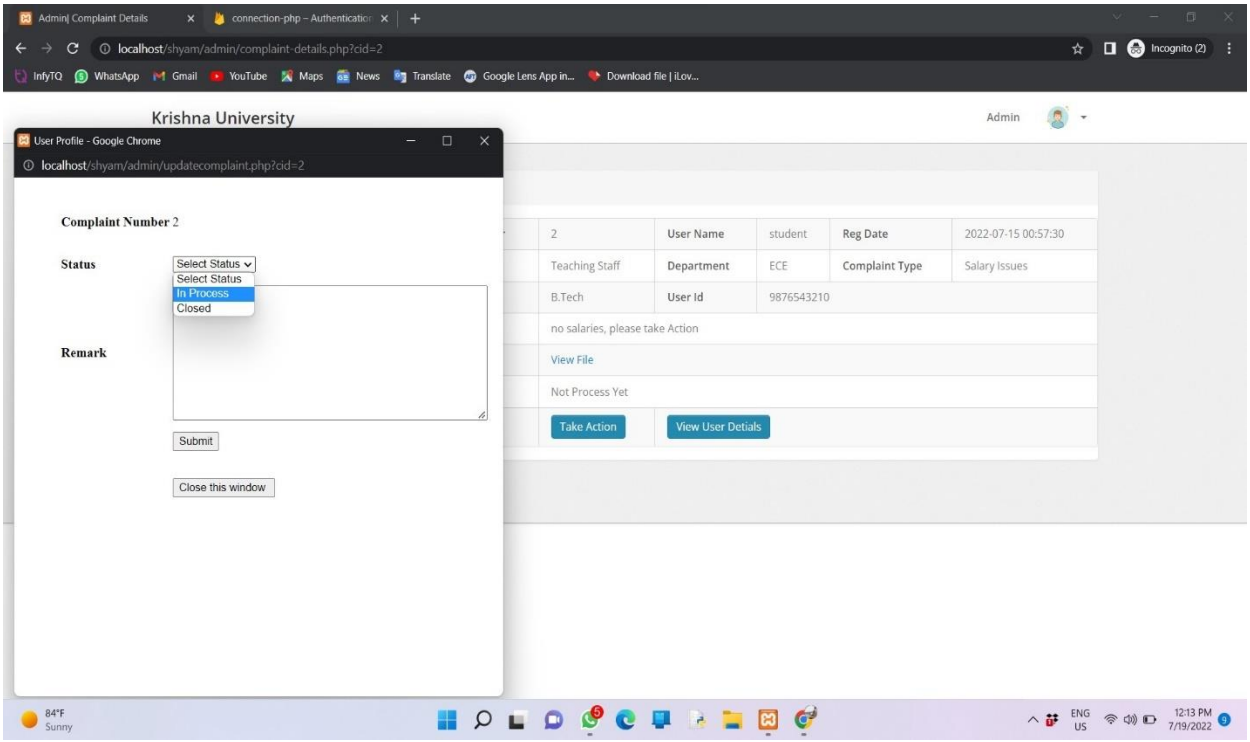


Fig. 10.27. Action against the Complaint in Admin Portal in GRC

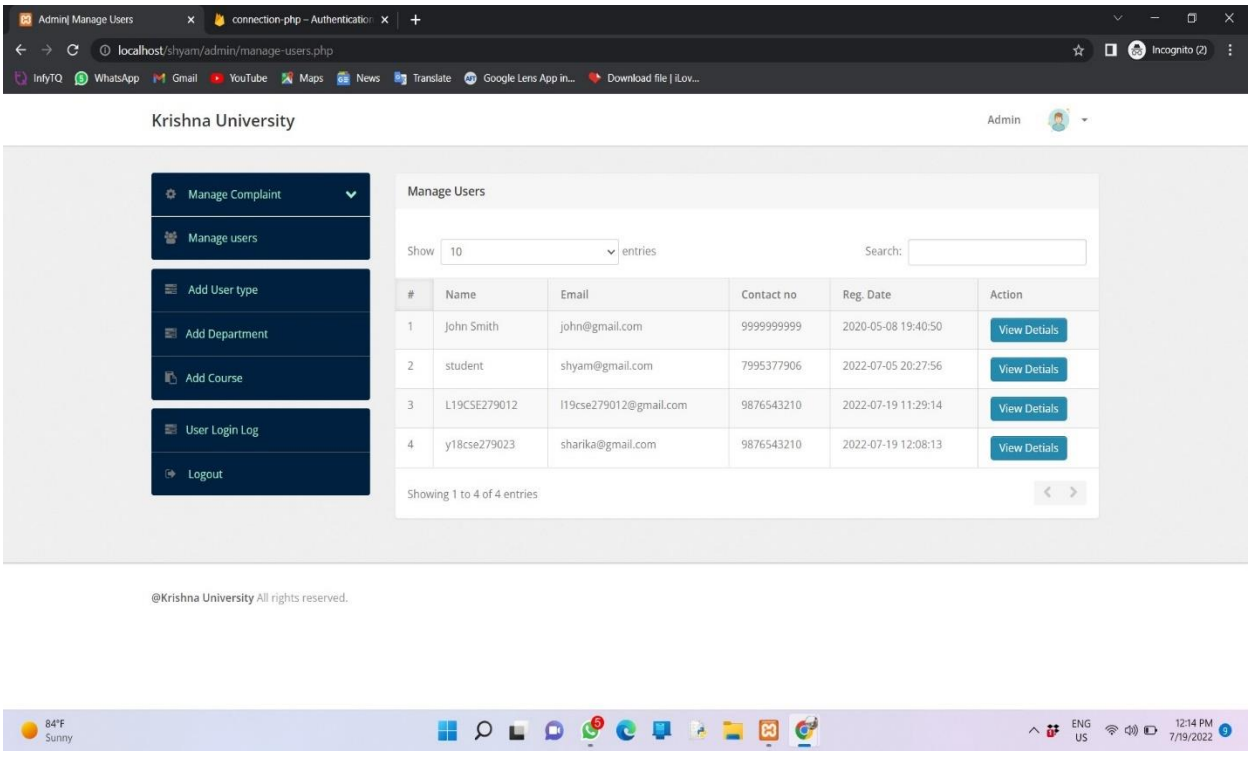


Fig. 10.28. Manage Users in Admin Portal in GRC

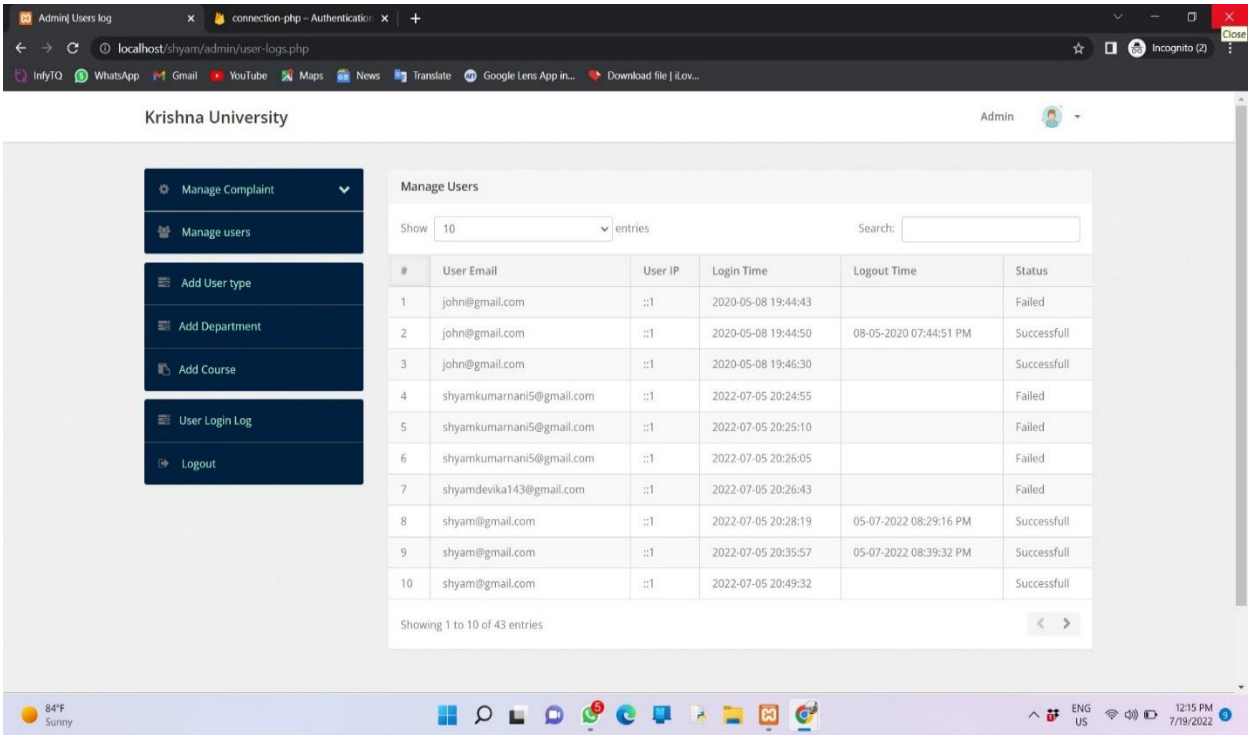


Fig. 10.29. User Login Log in Admin Portal in GRC

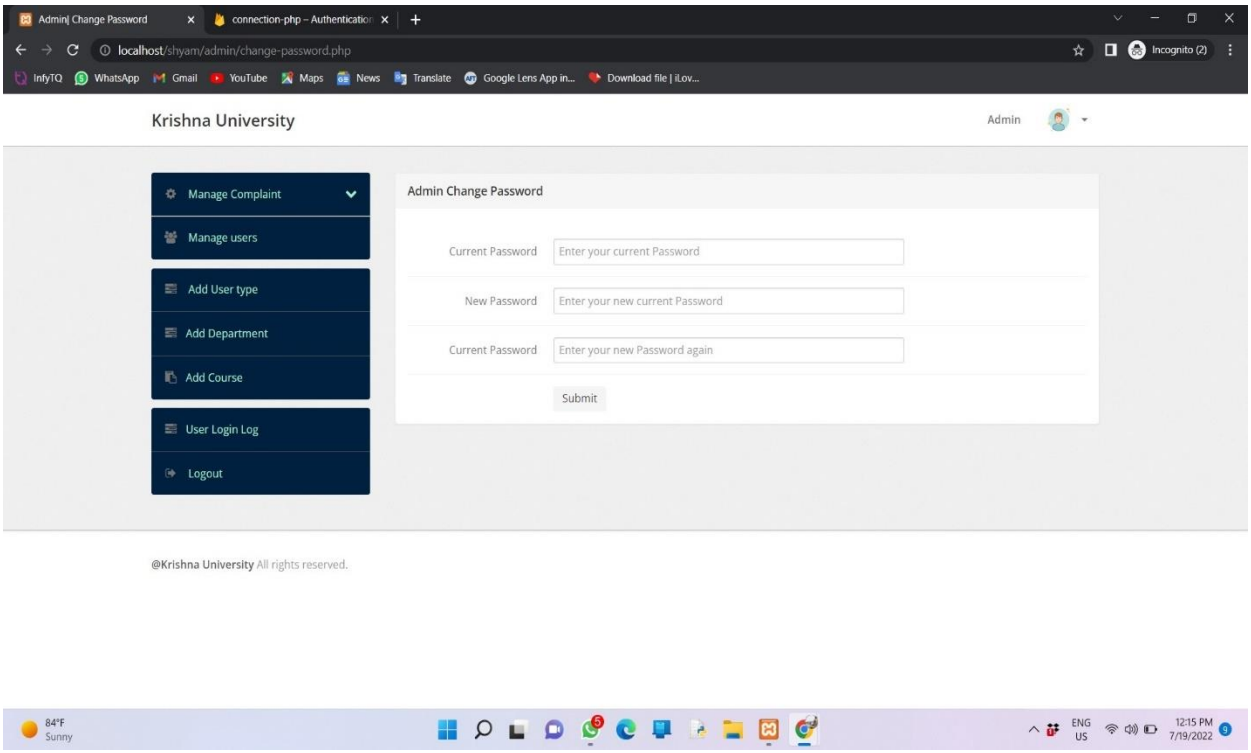


Fig. 10.30. Change Password in Admin Portal in GRC

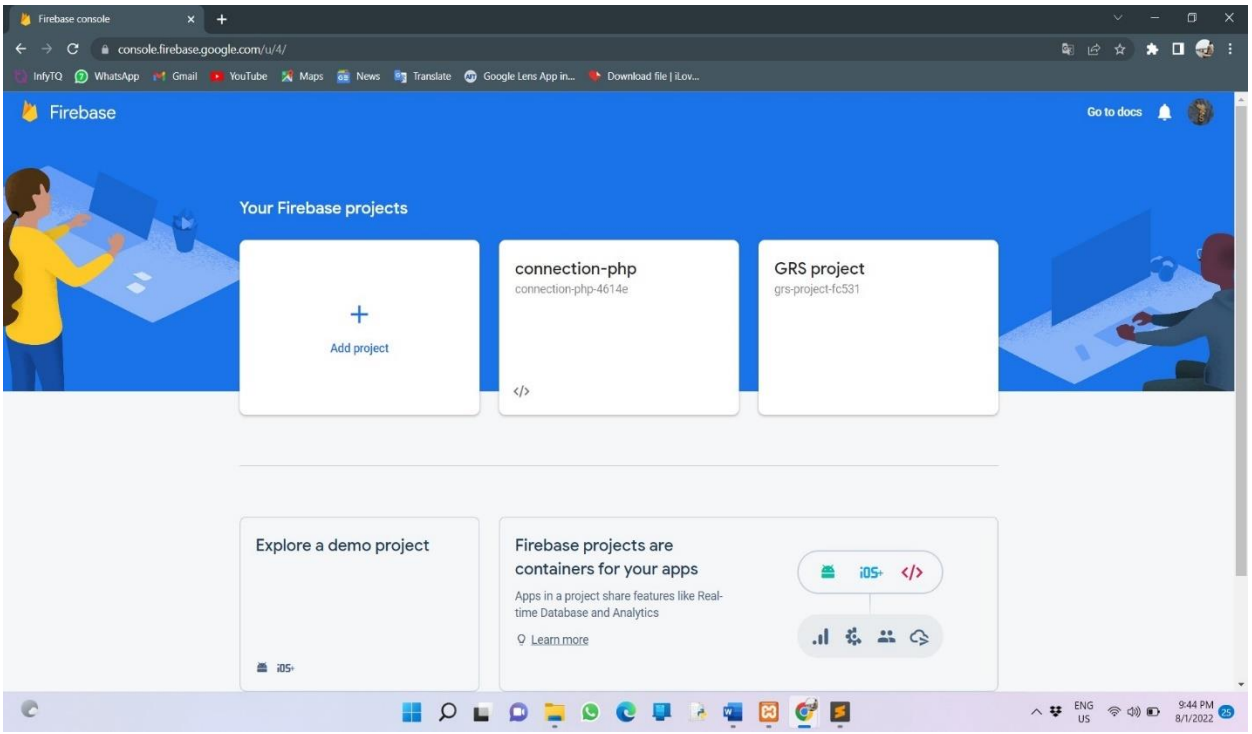


Fig. 10.31. User Authentication Data Base in Firebase

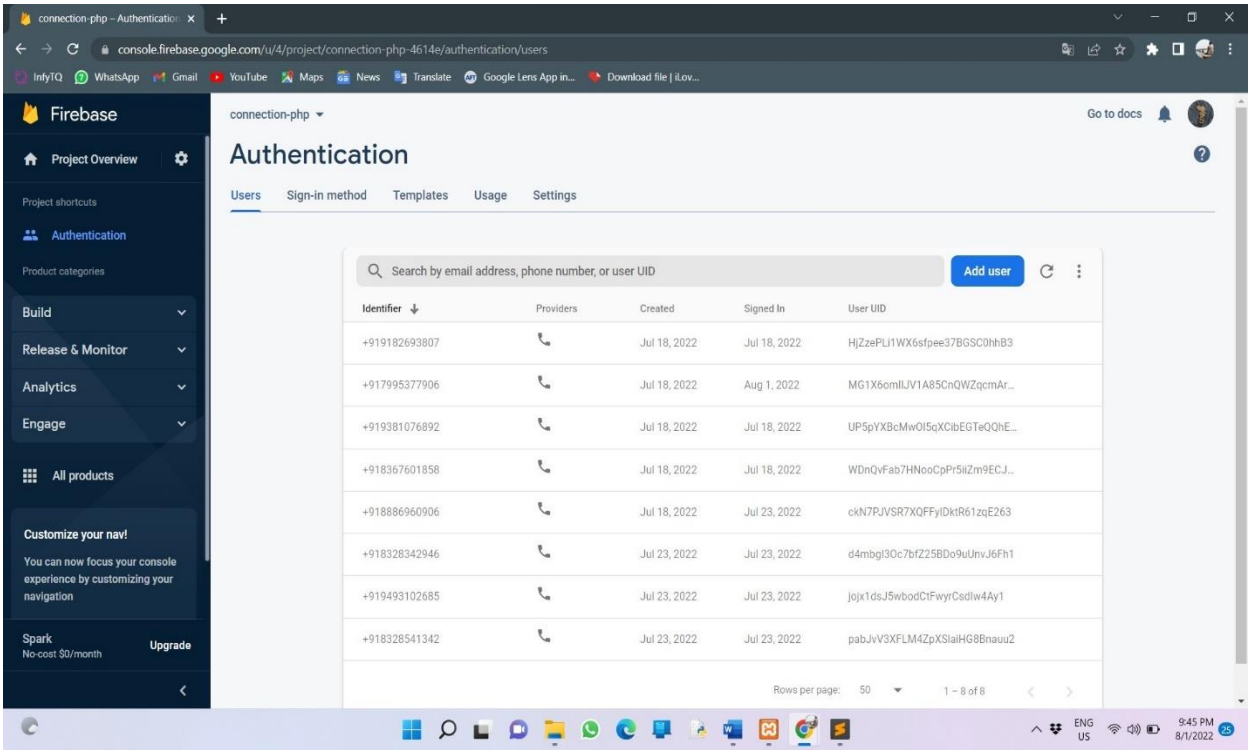


Fig. 10.32. User Authentication Login Data Base in Firebase

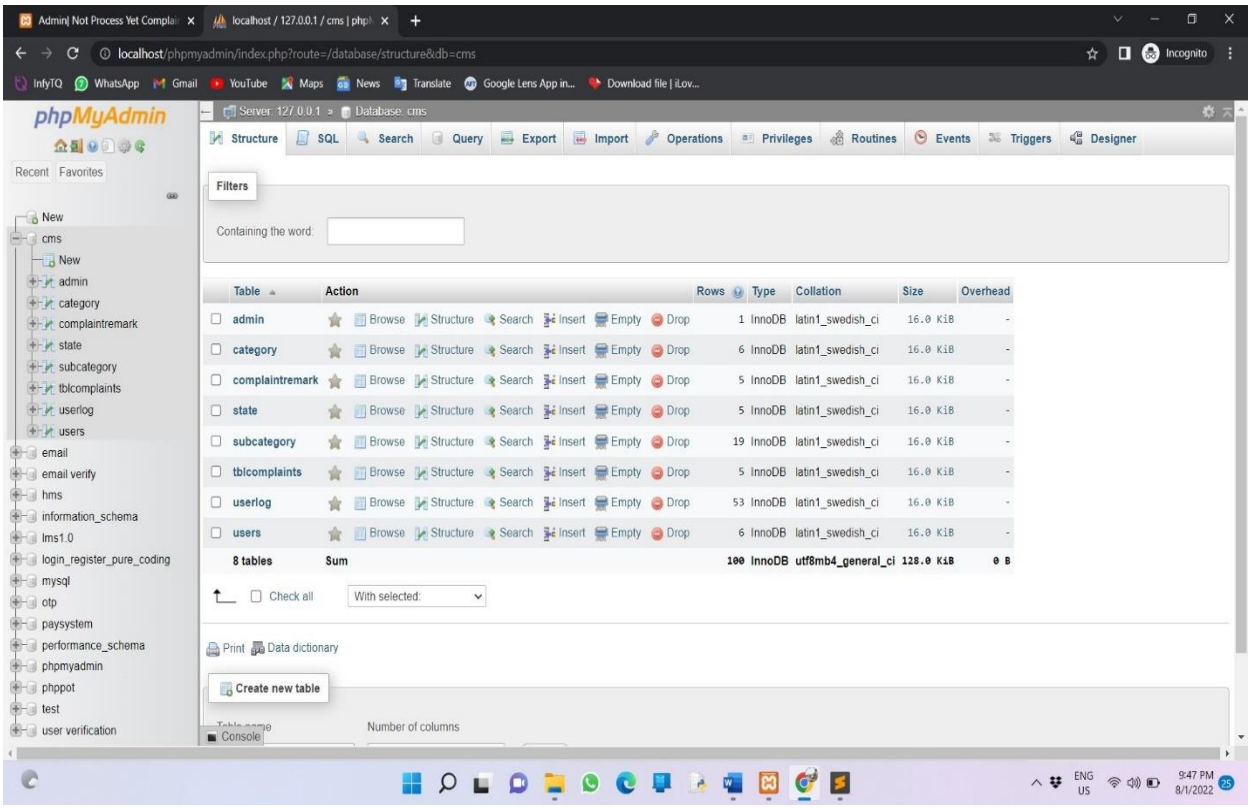


Fig. 10.33. Data Base for Xampp Server(php)

# 11.CONCLUSION

This project has demonstrated a proposed Grievance System for the grievance redressed of students covering various domains of complaints, which could be lodged easily, and thus leading to easy and sure solutions to the problems being faced by a student on a regular basis.

This project is an attempt to highlight the fact that there are hardly such systems prevailing curtailing to the complaint redressed for students enrolled in numerous organizations. It has demonstrated a proposed GRIEVANCE REDRESSAL SYSTEM for student for the grievance redressed of student covering various domains of complaints, which could be lodged easily, and thus leading to easy and sure solutions or redressed to the problems being faced by a student on a regular basis. The technologies used comprise of HTML and CSS to design a user-friendly Graphical User Interface, PHP, and SQL to keep track of the records at the back end. This system would be suitable for any organization for the resolution of complaints and thus lead to a qualitative and quantitative development of the organization.

## FUTURE SCOPE

The GRS working on the pretext of the grievance redressed for the student currently works as web application among the various members and the targeted audience. To extend this further to fulfill various requirements, following enhancements are below.

- However, many router enhancements of the system worked upon are possible. The prime focus includes the development of a mobile application to increase the mobility of the application. Since the future demarcates the usage of mobile applications and as seen portable devices are ubiquitous which will facilitate the receiving of all the notifications in the cell phone by the members and students associated with the application further increasing the reliability of the system and the rate of problem solving.
- The mobile application id targeted to enhance the user experience by providing the user with additional features for uploading the pictures the proofs in the form of audio or video files, which might enhance the case solving ability especially in such cases with a high rate of severity.
- A toll-free helpline could be made available on a 24x7 basis for the victims in an order to lodge complaints at emergency hours or to seek counsel in case of catastrophes.
- Above all, a tracker could be added as a part of the future perspectives to track the performance of various committee members involved into the process on the pretext of the provided feature of the report generation.

# 12.REFERENCES

https://www.w3schools.com/js/

https://www.w3schools.com/mysql/mysql_sql.asp

https://docs.microsoft.com/en-us/visualstudio/get-started/csharp/tutorial-console?view=vs-2022

https://stackoverflow.com

http://www.apachefriends.org/download.html

http://www.geeksforgeeks.org/7-best-books-for-sql/