**GRIEVANCE REDRESSAL PORTAL**

**Project Report Submitted in partial fulfilment of the**

**Requirements for the award of the Degree of**

# BACHELOR OF ENGINEERING IN INFORMATION TECHNOLOGY

**By**

**Greta Koutika**

**1602-18-737-069**

**Vyuha Kandula**

**1602-18-737-119**

**Syed Ayaz**

**1602-18-737-111**

**Under the guidance of**

**Dharma reddy R**

**Assistant Professor**



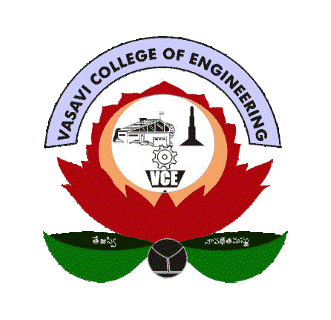
**Department of Information Technology**

**Vasavi College of Engineering (Autonomous)**

# (Affiliated to Osmania University)

# Ibrahimbagh, Hyderabad-31

# 2020



**Vasavi College of Engineering, Ibrahimbagh**

# Department of Information Technology

Report on

# Grievance Redressal Portal - VCE

**Submitted by**

Greta Koutika (1602-18-737-069)

Vyuha Kandula (1602-18-737-119)

Syed Ayaz (1602-18-737-111)

**Under the guidance of Internal guide:**

Mr. Dharma Reddy

**Mentor:**

K. Sai Krishna Reddy (1602-17-737-033)

**Vasavi College of Engineering**

**(Affiliated to Osmania University)**

**Hyderabad-500 031**

**Department of Information Technology**



**DECLARATION BY THE CANDIDATE**

We **GRETA KOUTIKA, VYUHA KANDULA, SYED AYAZ** bearing hall ticket number, **1602-18-737-069 and 1602-18-737-119**, **1602-18-737-111** hereby declare that the project report entitled **GRIEVANCE REDRESSAL PORTAL** Department of Information

Technology, Vasavi College of Engineering, Hyderabad, is submitted in partial fulfilment of the requirement for the award of the degree of **Bachelor of Engineering** in **Information Technology.**

This is a record of bonafide work carried out by me and the results embodied in this project report have not been submitted to any other university or institute for the award of any other degree or diploma.

**Greta Koutika**

**1602-18-737-069**

**Vyuha Kandula**

**1602-18-737-119**

**Syed Ayaz**

**1602-18-737-111**

**ABSTRACT**

**GRIEVANCE REDRESSAL PORTAL**

A Grievance is that the dispute that arises at any level of organization. In an education organization many students have vulnerable situations where they fail to state their problems or sometimes fail to seek out proper support for the issue they are facing in an organization this may lead them to dejection.

The purpose of this project is to provide optimised solutions for the student grievances.

Grievance Redressal System is an online platform to receive and act on complaints reported by students of private or public institutions, enabling prompt actions on any issue raised by them and to avail services more effectively.

On scrutinizing the above mentioned problem we culminates this Student Grievance Support System with the following features.

1. The web application builds a platform for the students to lodge the arising conflicts in their daily walk of lives.

2. In the web application students can address their complaints which are forwarded to the Grievance Redressal Committee.

3. The Committee will forward the valid complaints to the Department supporting the sensitivity of the matter. The Institute or Department will take action and update the status which can be viewed by the students.

4. This project provides transparency to the students which can be incorporated to supply solutions to the students.

This solution will reduce time, energy to solve the problems.

**TABLE OF CONTENTS**

**CHAPTER** **PAGE**

ABSTRACT

LIST OF FIGURES

CHAPTERS

CHAPTER 1 – Introduction.............................................................................................................1

CHAPTER 2 – Related Work ..........................................................................................................2

2.1 Existing System

2.2 Proposed System

2.3 Need for computerization

CHAPTER 3 – Proposed Work .......................................................................................................3

3.1 Use Case Diagram..........................................................................................................3

3.2 Screenshots ....................................................................................................................4

3.3 Architecture and Technology used ................................................................................9

3.4 Design ..........................................................................................................................11

3.5 Implementation ............................................................................................................13

3.6 GitHub links and Folder Structure ...............................................................................17

3.6 Testing..........................................................................................................................19

CHAPTER 4 – Results...................................................................................................................20

CHAPTER 5 – Discussions and Future Work ...............................................................................22

REFERENCES ..............................................................................................................................22

**LIST OF FIGURES**

**FIGURE** **PAGE**

Fig 1. Use case diagram of GRS………...............................................................................................9

Fig 2. Homepage (header) of GRS ..................................................................................................10

Fig3.Student Profile ......................................................................................................................10

Fig 4. Academic Complaints ..........................................................................................................12

Fig 5. Non Academic Complaints ...................................................................................................12

Fig 6.Committee Page ...................................................................................................................12

Fig 7. Faculty Page .........................................................................................................................13

Fig 8. Architecture of GRS..............................................................................................................15

Fig 9. Class Diagram ......................................................................................................................16

Fig 10. Sequence Diagram .............................................................................................................16

Fig 11. Student Flowchart Diagram................................................................................................18

Fig 12. Committee Flowchart Diagram..........................................................................................18

Fig 13. Faculty Flowchart Diagram.................................................................................................19

Fig15. Folder structure ..................................................................................................................20

## INTRODUCTION

We develop a Web Application which is dynamic, interactive, responsive and user-friendly. This project covers complaints in all sorts, complaints regarding

* Ragging
* Irregularities in admission process
* finance (fee payment)
* Conflicts in names and mark sheets if any and also other issues faced by the students.

The main objective of the project is to provide redressal to the complaints without time consumption.

This Web Application is accessed by the registered students and login is provided for the Redressal Committee, Institute and Department with appropriate credentials.

## 2. RELATED WORK

### 2.1 EXISTING SYSTEM

The existing way to get our queries solved is, going to college and enquiring the respective faculty. But it becomes very difficult for us to manually go and resolve our problems.

When we approach them, sometimes they might be engaged in some other tasks and may not be available. Though we succeed in approaching them, there will be some situations where they need some time to solve it. But later there are odds that they might forget. This will not help us in getting our problems solved.

This scenario is old-fashioned in this advanced technological world.

### 2.2 PROPOSED SYSTEM

The proposed system fully interactive and efficient, which every can use easily and anything can found easily by using search bars from the website.

Students, Faculty, Committee members should login with their respective credentials.

**Student:** Once they login, they can choose a particular domain, faculty they want to address according to their problem and raise a complaint. This complaint will be further forwarded to that respective faculty of that domain.

**Committee Member:** After logging in, they can see all the complaints registered by students. They will forward the complaints to respective faculty.

**Faculty**: After logging in, they can view all the issues posted by students.

As soon as the complaint is rectified, the status of the complaint is updated by the faculty which can be directly viewed by the student.

### 2.3 NEED FOR COMPUTERIZATION

All the complaints can be solved easily. Since there will be a record of every grievance no issued will be missed. It saves time, energy.

**3. PROPOSED WORK**

### 3.1 Use Case Diagram

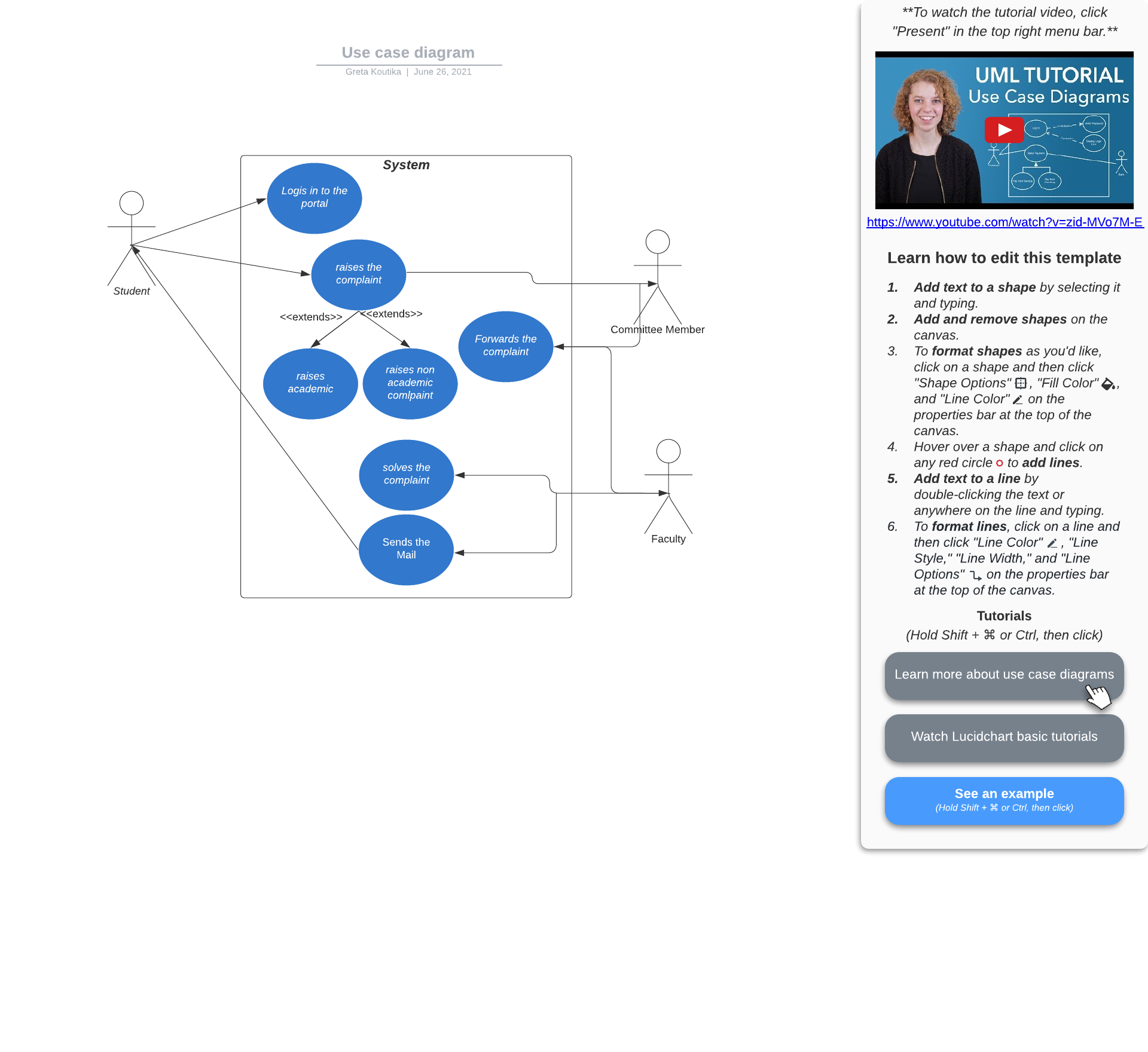


Fig 1. Use case diagram

### 3.2 Screenshots

#### 3.2.1 Homepage – Login Page(General view)

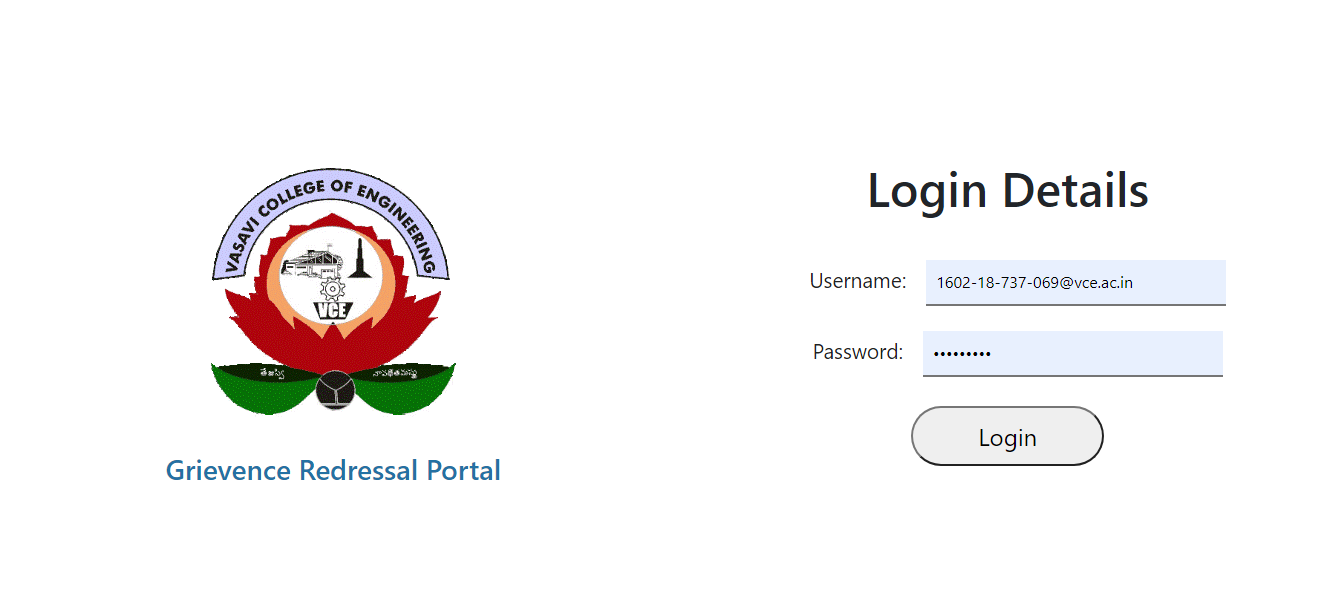


Fig 2. Home Page

#### 3.2.2 Student Profile

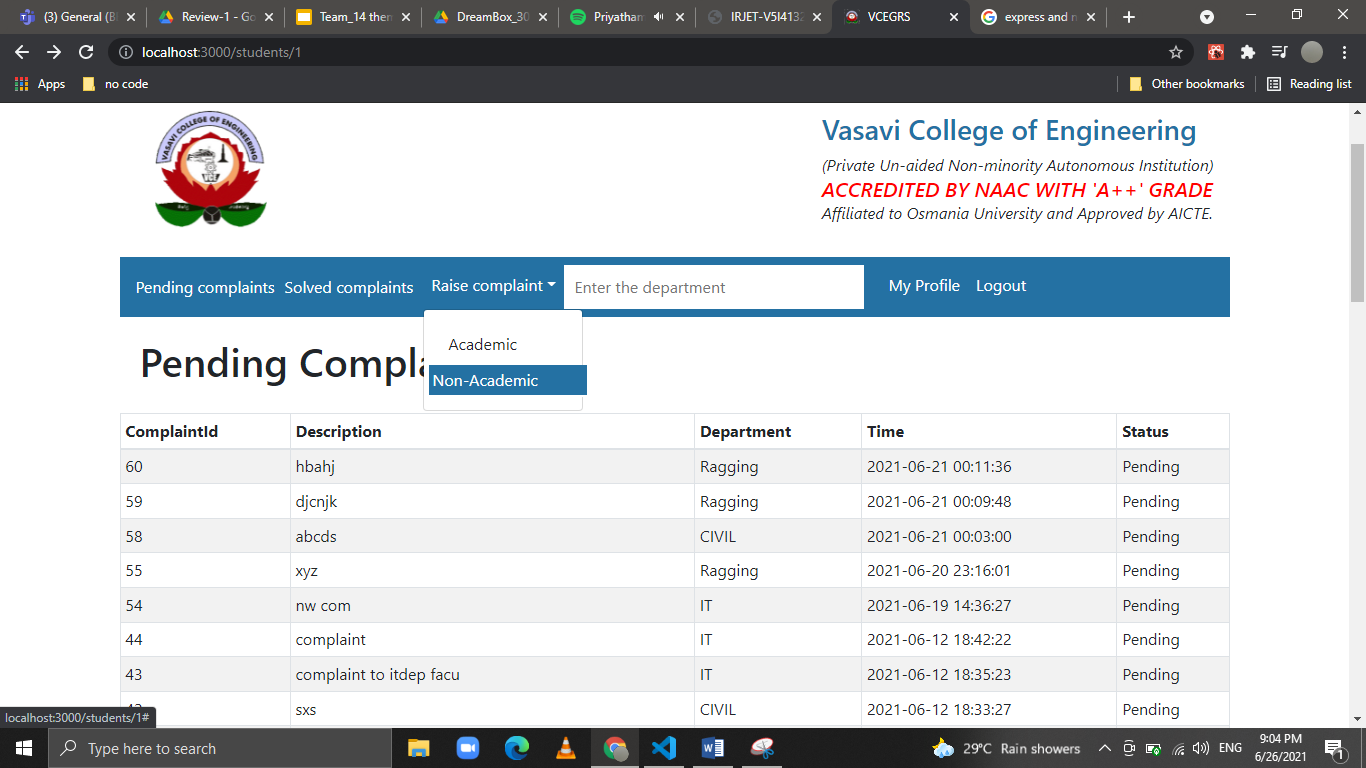
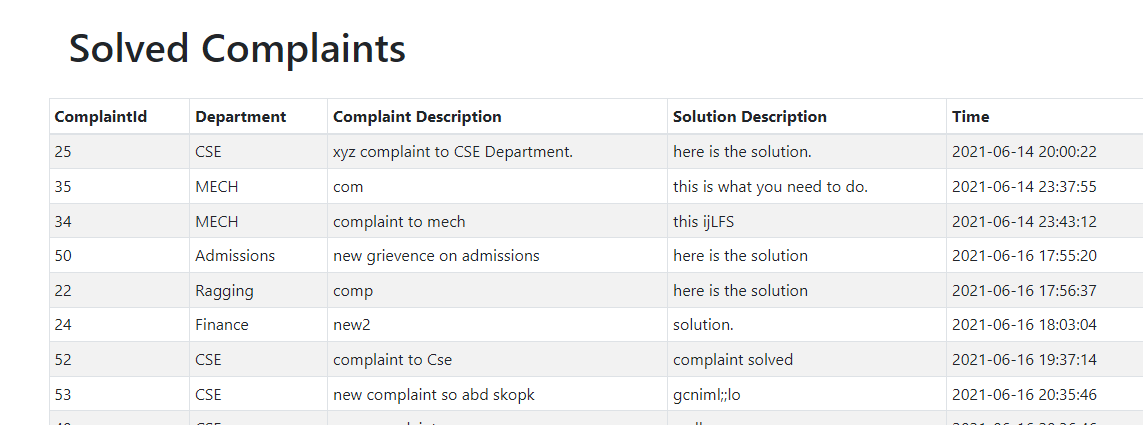
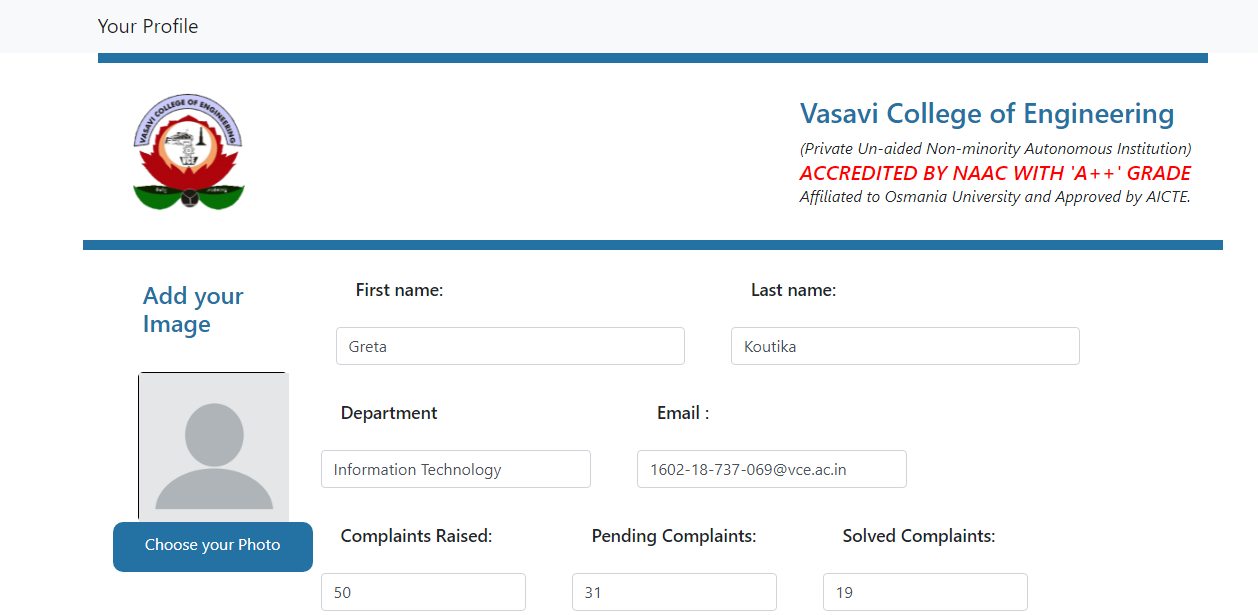


Fig 3. Student Profile page





**3.2.3 Academic Complaint Page:**

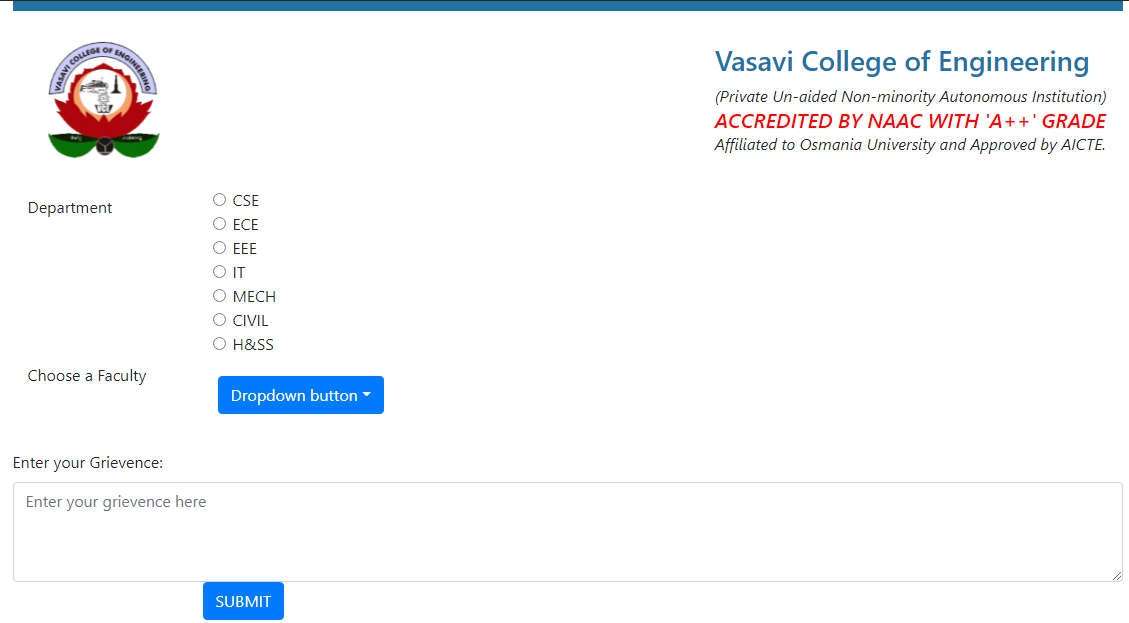


Fig 4. Academic complaints

**3.2.4** **Non Academic Complaint Page:**

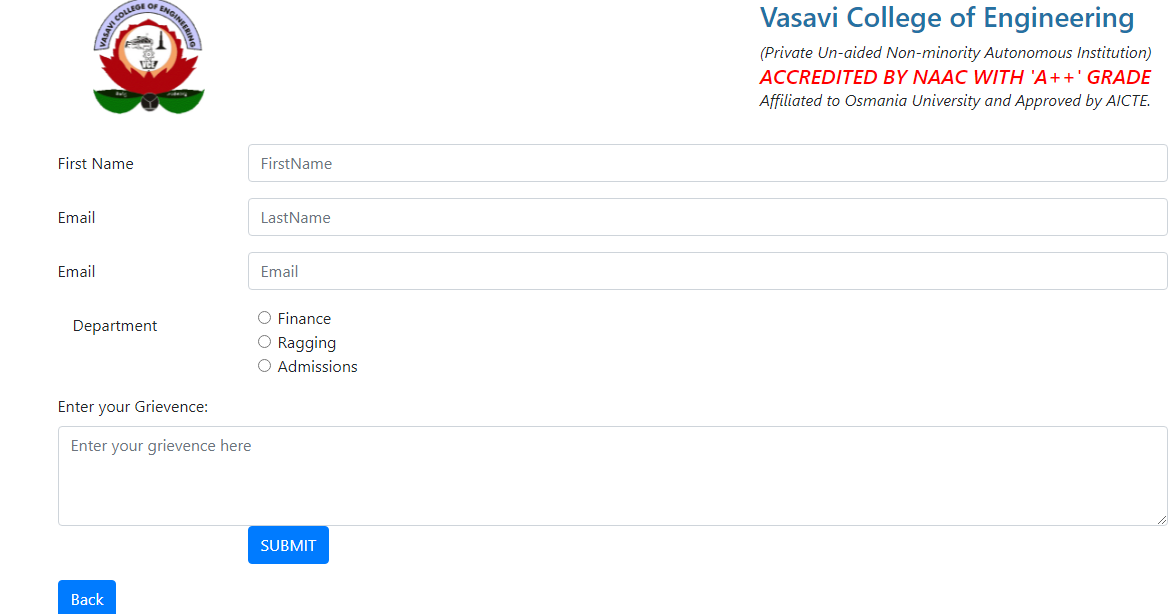
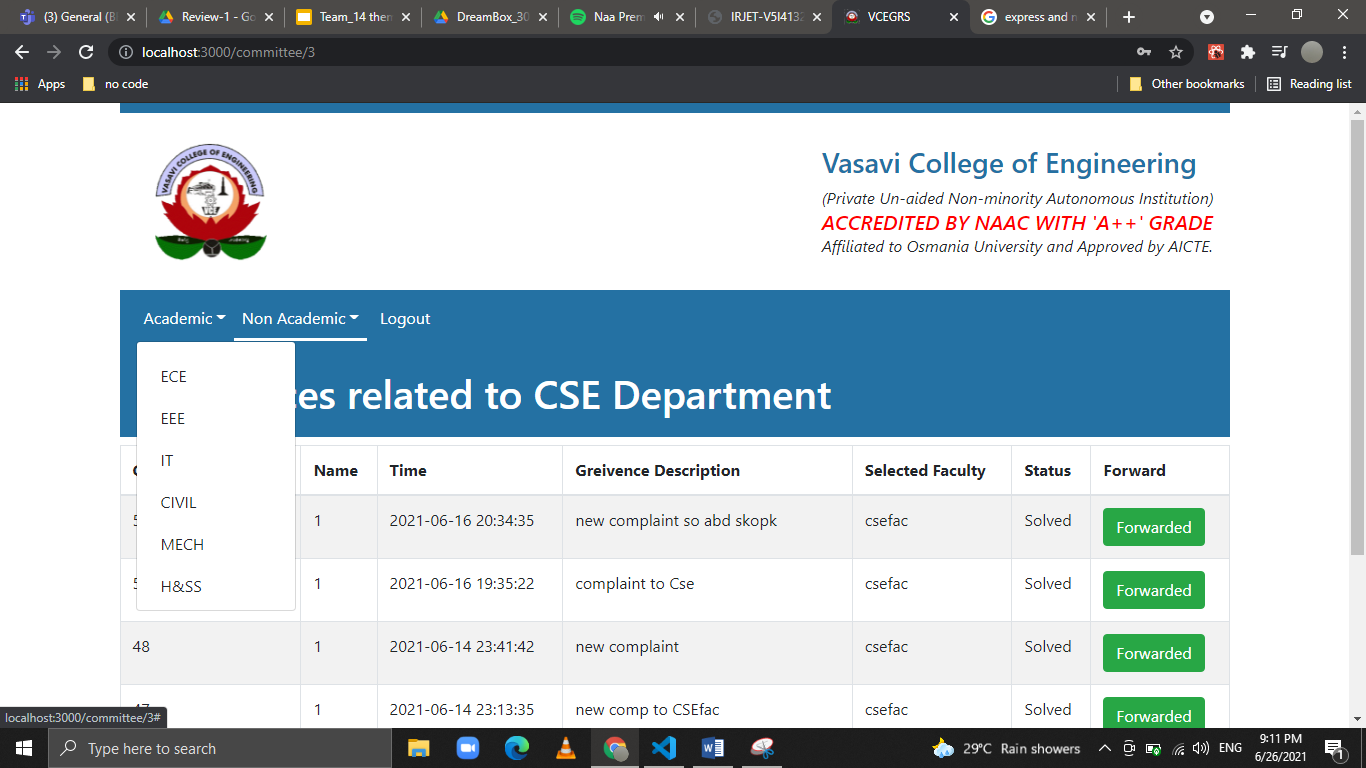


Fig 5. Non Academic complaints

#### 

#### 3.2.5 Committee Profile



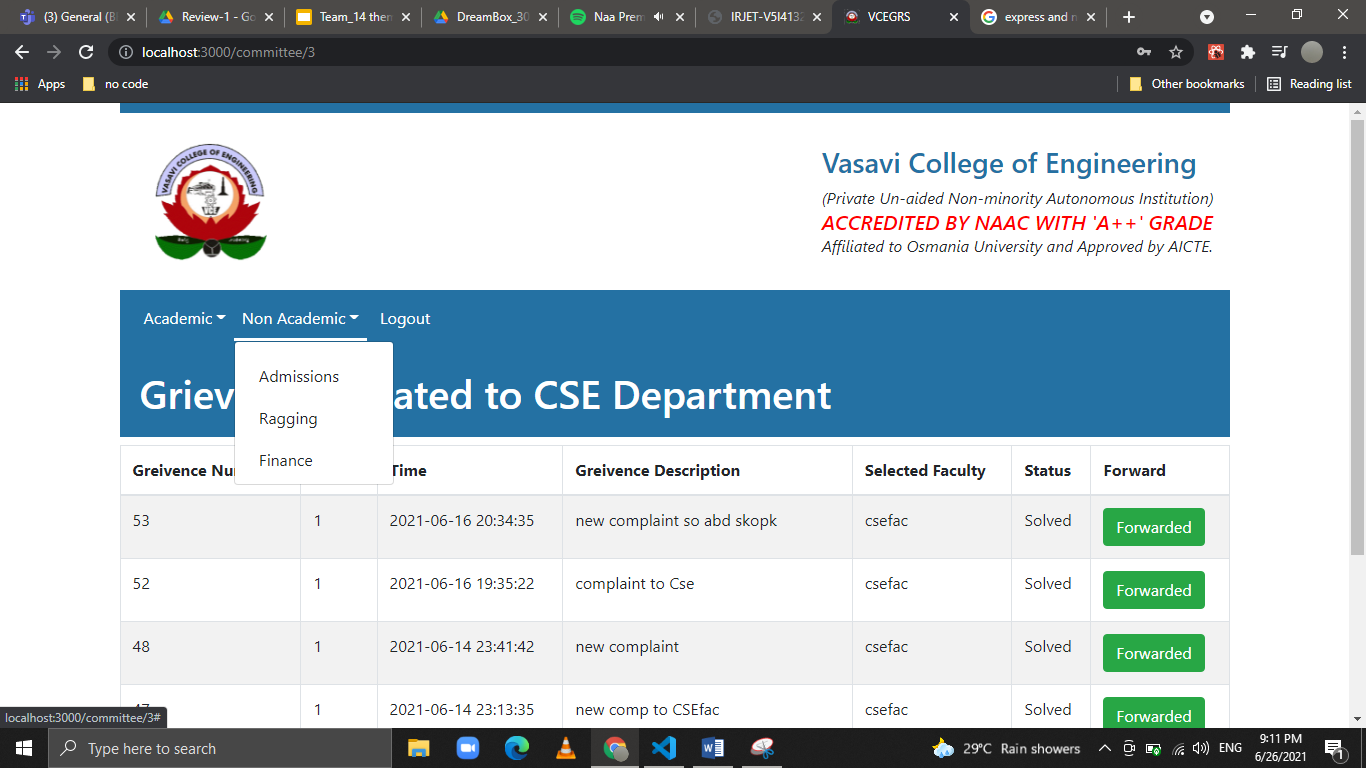


Fig 6: Committee page

#### 3.2.5 Faculty profile

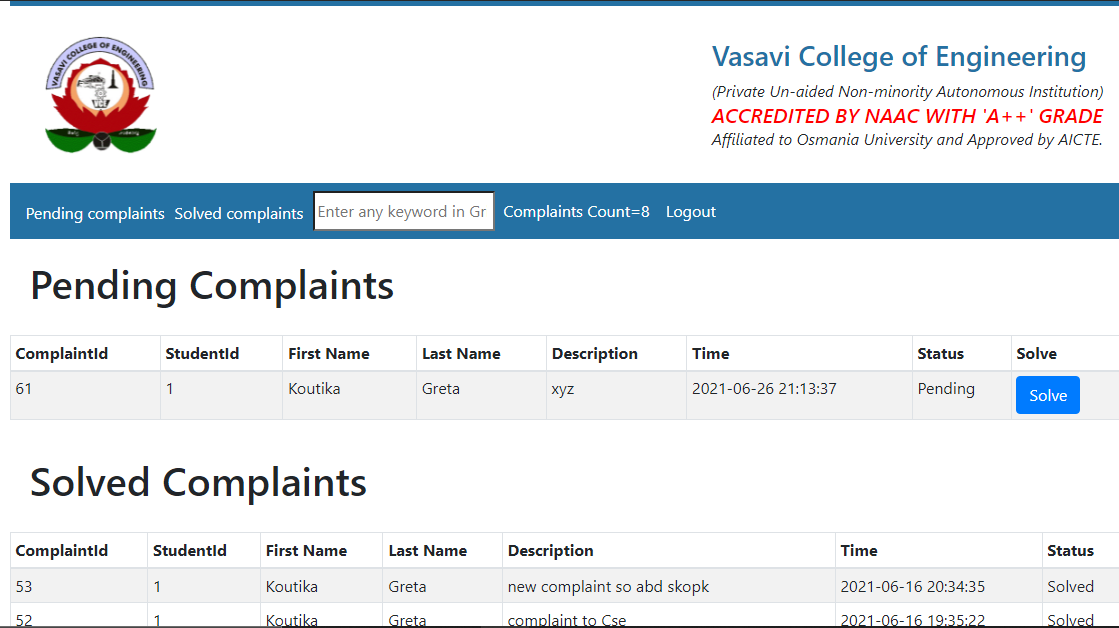
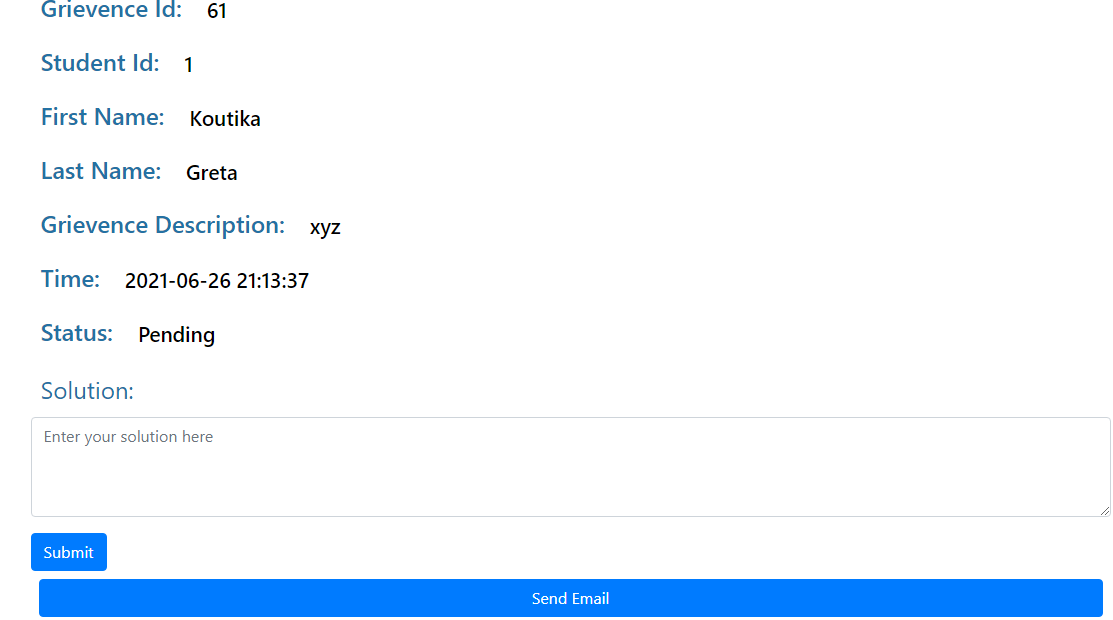


Fig 7: Faculty page

#### 3.2.6 Solve Complaint page



### 3.3 Architecture and Technology

**Technology used:**

Frontend technologies:

* Html
* Bootstrap CSS
* JS
* React JS

Backend technologies:

* SQLite (Database)
* Node JS
* Express JS

**Architecture:**

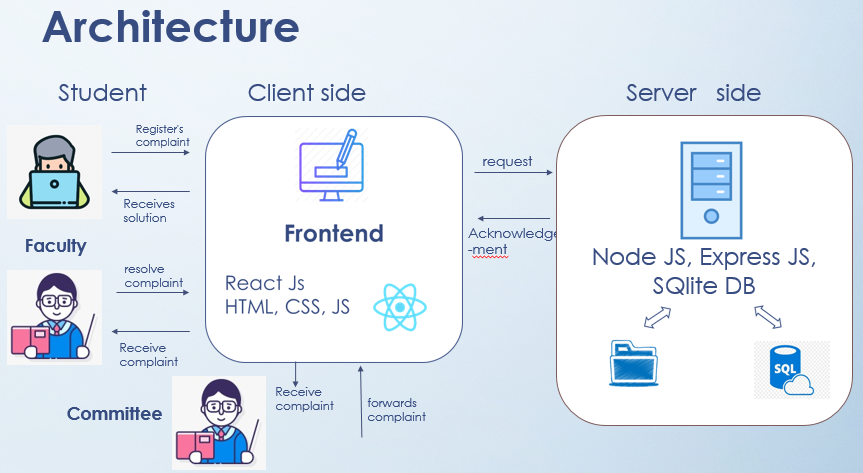


Fig 8: Architecture

#### 3.3.1 React JS Library

### Declarative

React makes it painless to create interactive UIs. Design simple views for each state in your application, and React will efficiently update and render just the right components when your data changes.

Declarative views make your code more predictable and easier to debug.

### Component-Based

Build encapsulated components that manage their own state, then compose them to make complex UIs.

Since component logic is written in JavaScript instead of templates, you can easily pass rich data through your app and keep state out of the DOM.

#### 3.3.2 Express JS and Node JS

### Web Applications

Express is a minimal and flexible Node.js web application framework that provides a robust set of features for web and mobile applications.

### APIs

With a myriad of HTTP utility methods and middleware at your disposal, creating a robust API is quick and easy.

### Performance

Express provides a thin layer of fundamental web application features, without obscuring Node.js features that you know and love.

.

### 3.4 Design

#### 3.4.1 UML static diagram

##### 3.4.1.1 Class Diagram

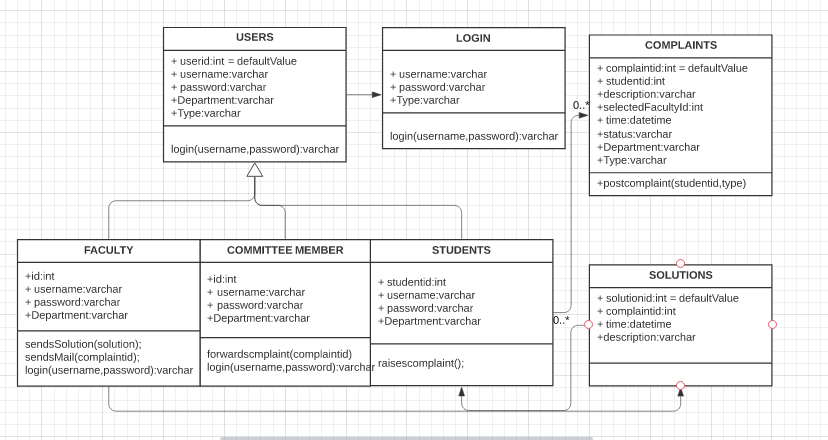


Fig 9. Class Diagram

**3.4.1.2 Sequence Diagram**

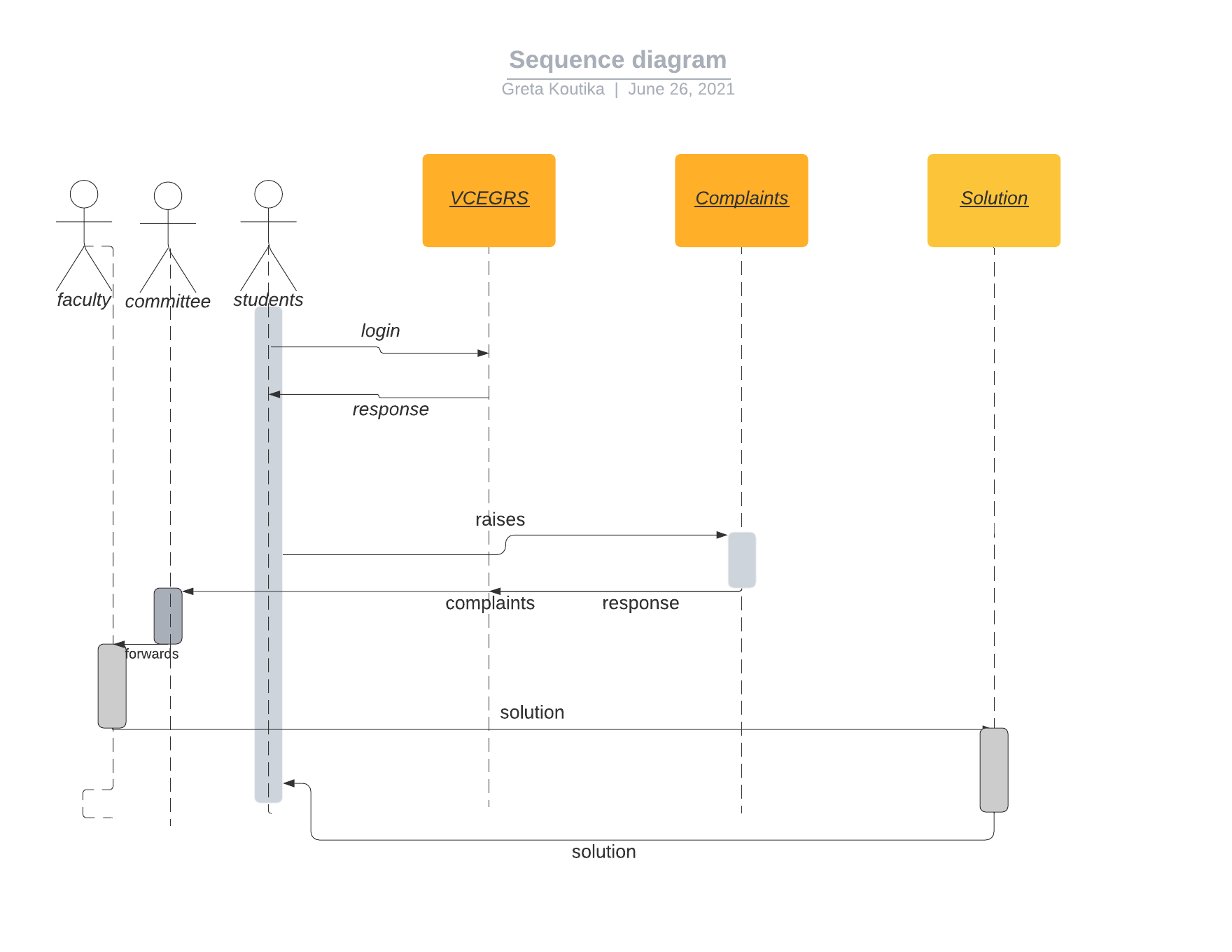
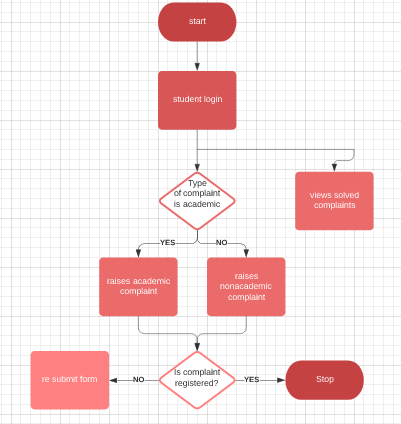


Fig 10. Sequence Diagram

##### 3.4.1.3 Flow Chart Diagram

**Student:** Fig 11. Student Flow Chart



**Committee:**

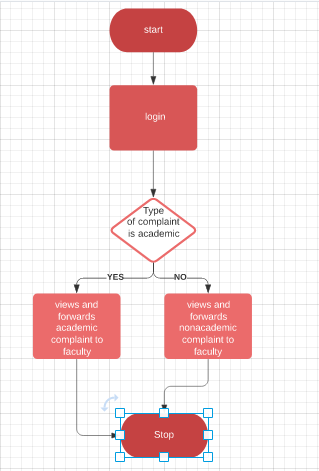
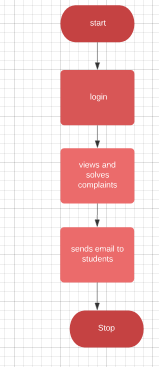


Fig 12.Committee member Flow Chart

**Faculty:**

 Fig 13. Faculty Flow Chart

**3.5 Implementation**

**Frontend:**

**Components Used:**

* 1. **Header:** It helps in rendering same header for all pages.
  2. **Login Form:** There are three types of Users- Student, Facultymember, Committee member. According to the details given by the user, they will be sent to their respective pages.
  3. **Student Profile Component:** It is responsible for managing data of students.
  4. **Academic Complaints:** It is responsible for submitting academiccomplaints registered by students.
  5. **Non Academic Complaints:** It is responsible for submitting non- academic complaints registered by students.
  6. **Committee:** It is responsible for managing grievances of students.
  7. **Academic Department**: Based on the department choosed by the dropdown by the committee member it retrieves all the complaints related to that Department.

Here the Departments are CSE, ECE, EEE, IT, MECH, CIVIL, H&SS

* 1. **Non-Academic Department**: Based on the department choosed by the dropdown by the committee member it retrieves all the complaints related to that Department.

Here the Departments are Ragging, Admissions and Ragging.

* 1. **Faculty Profile:** It helps in handling all the complaints received by the faculty member.
  2. **Protected Route:** It helps in managing the secured redirection between pages with the help of Cookies.

**Backend:**

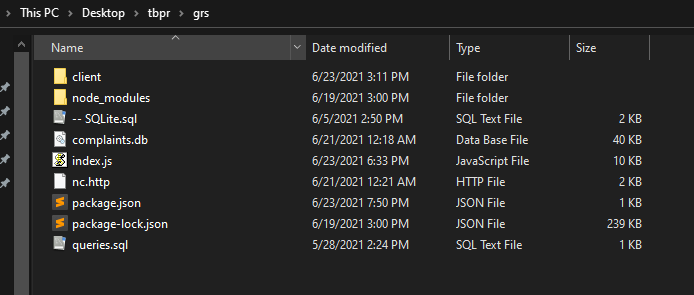
We have index.js which manages all the API calls with the help of express JS.

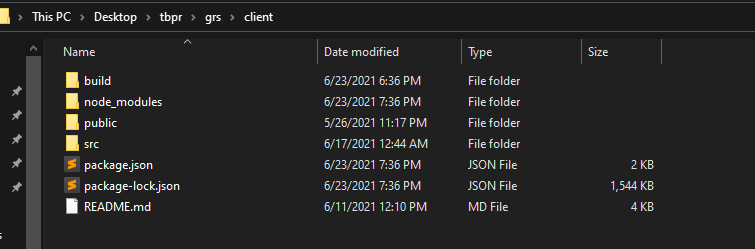
#### 3.6 GitHub links

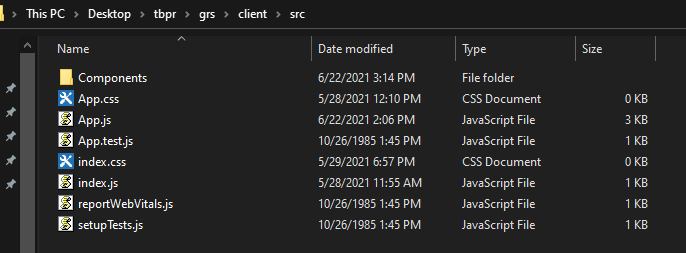
Greta Koutika (1602-18-737-069)

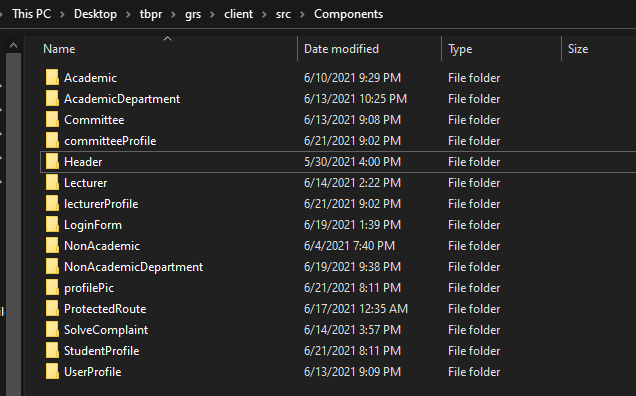
 **https://github.com/Gretakoutika/GRS**

##### 3.6.1 Folders Structure









#### 3.7 Testing

1. The testing part could be done on raising the complaint and checking whether it is received by committee or not.

2. If yes then the committee member will forward it to respective faculty member.

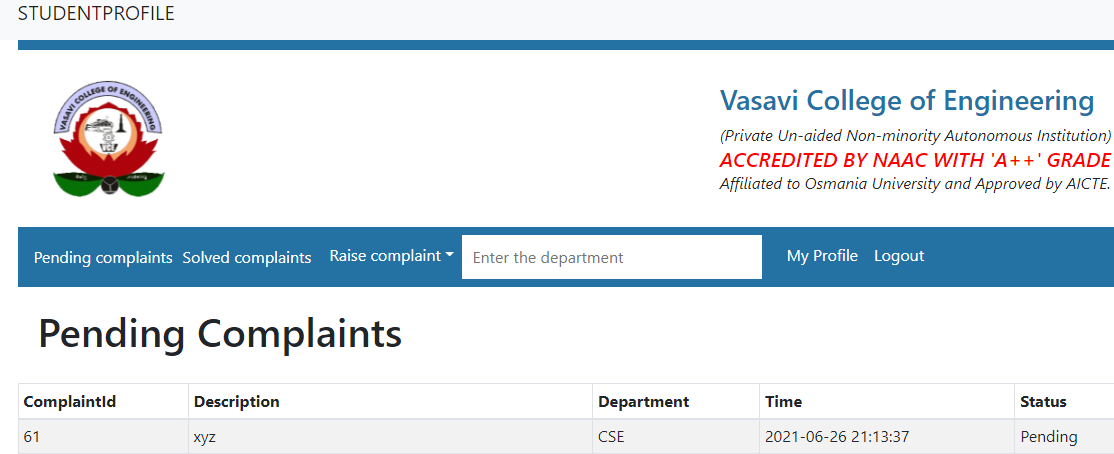
3. Faculty should be able to receive the complaint and provide a solution and send mail.

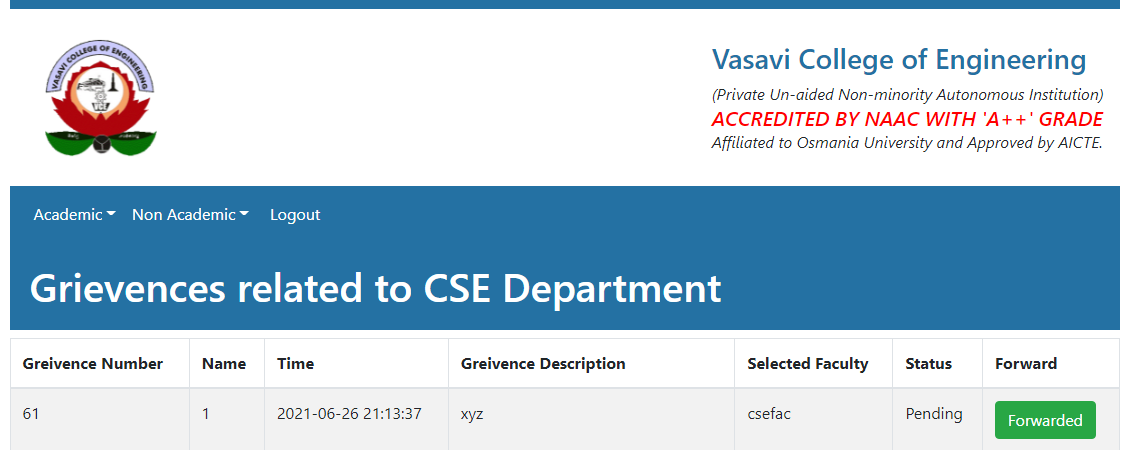
4. Complaint status should be changed.

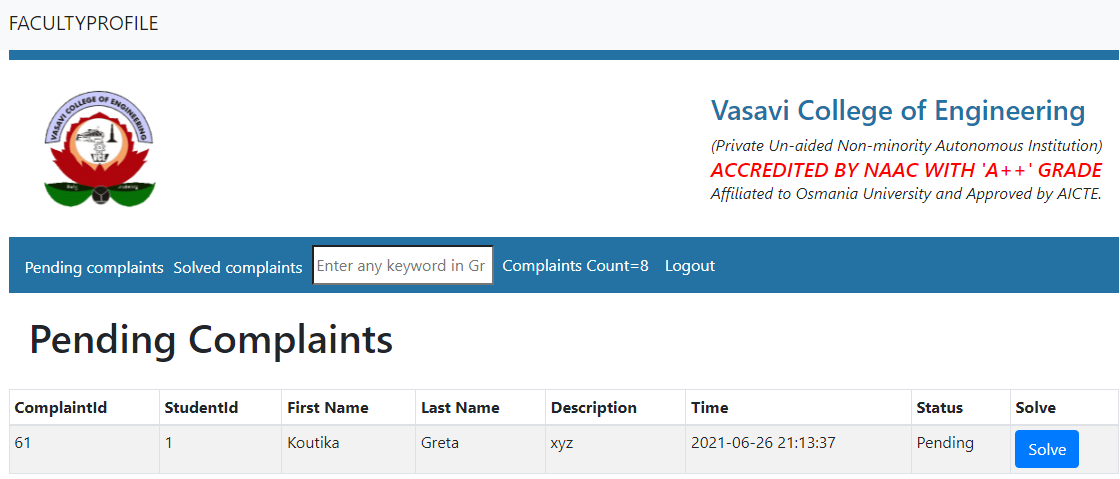
### 4.Results

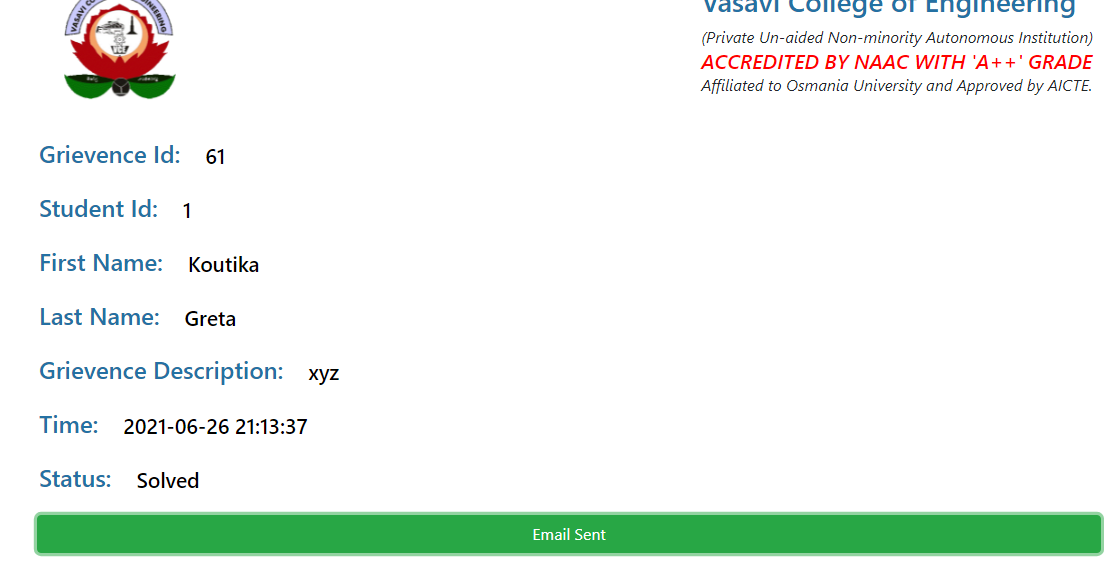
The results you can see in the form of posts. By watching the full post of a company you can understand the everything about the company. In the post you can see company information, eligibility criteria, skills required to crack the interview, recruiting process of the company and mostly asked questions of that company. These are the details i included in every post.

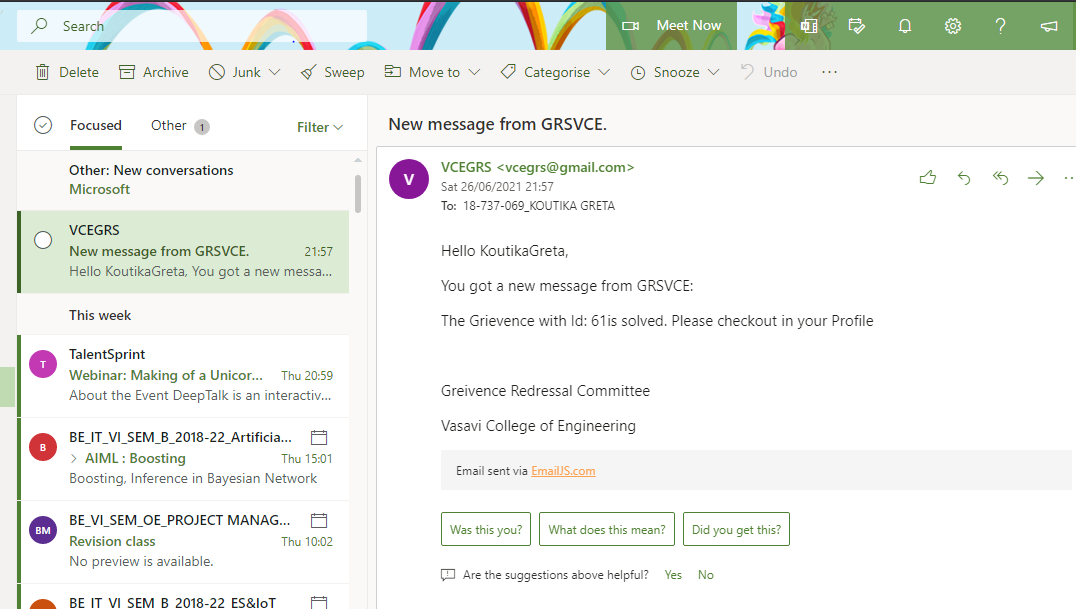
You can see the results of the posts by using below picture.



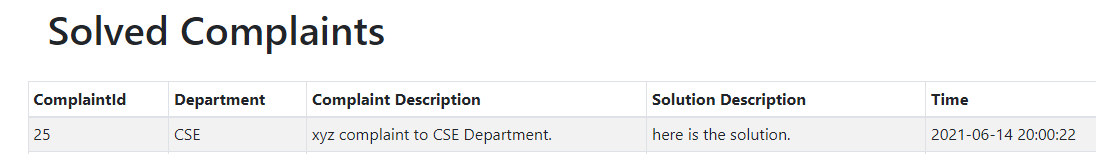








In Student Profile



**5. Discussion and Future Work**

Every application has its own merits and demerits. The project has covered almost all the requirements. Further requirements and improvements can easily be done since the coding is mainly structured or modular in nature. Changing the existing components or adding new components can append improvements. Further enhancements are

1. Student should be able to post complaint using different media.
2. Want to Use data Science and add Principal portal so that he can view the statistics of Grievances.
3. Create a mobile app.

### References

1. <https://reactjs.org/>
2. <https://youtu.be/w7ejDZ8SWv8> - React tutorial
3. <https://youtu.be/Oe421EPjeBE> - Node and Express js tutorial