

B.Tech Project Report

VJTI Records System

*Submitted in partial fulfillment of
the requirements for the award of the degree of*

Bachelor of Technology in Information Technology

Submitted by

Student Name	University No
<hr/>	
Prashil Bhimani	131080010
Neel Kapadia	131080015
Mikin Shah	131080016
Vidhit Patni	131080020
Utsav Shah	131080021

Under the guidance of
Sandeep Udmale



Department of Information Technology
VEERMATA JIJABAI TECHNOLOGICAL INSTITUTE
Matunga, Mumbai

May 5, 2017

Statement of the Student

VEERMATA JIJABAI TECHNOLOGICAL INSTITUTE

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources.

I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea / data / fact / source in my submission.

I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Roll No	Names of Students	Signature
---------	-------------------	-----------

131080010	Prashil Bhimani	
-----------	-----------------	--

131080015	Neel Kapadia	
-----------	--------------	--

131080016	Mikin Shah	
-----------	------------	--

131080020	Vidhit Patni	
-----------	--------------	--

131080021	Utsav Shah	
-----------	------------	--

Date:

Certificate

This is to certify that

Roll No	Names of Students	Signature
---------	-------------------	-----------

131080010	Prashil Bhimani	
-----------	-----------------	--

131080015	Neel Kapadia	
-----------	--------------	--

131080016	Mikin Shah	
-----------	------------	--

131080020	Vidhit Patni	
-----------	--------------	--

131080021	Utsav Shah	
-----------	------------	--

of Bachelor in Technology in Information Technology have completed the thesis/ report/ dissertation entitled **VJTI Records System** to our satisfaction.

(Name and signature)

Guide/Supervisor

(Name and signature)

**Head, Department of
Computer Engineering
and IT**

(Name and signature)

Co-Guide/Co-Supervisor

(Name and signature)

Director, VJTI

Certificate

The thesis / dissertation / report, **VJTI Records System** submitted by

Roll No	Names of Students	Signature
---------	-------------------	-----------

131080010	Prashil Bhimani	
-----------	-----------------	--

131080015	Neel Kapadia	
-----------	--------------	--

131080016	Mikin Shah	
-----------	------------	--

131080020	Vidhit Patni	
-----------	--------------	--

131080021	Utsav Shah	
-----------	------------	--

is found to be satisfactory and is approved for the degree of Bachelor of Technology in Information Technology.

(Name and signature)
Supervisor/Guide

(Name and signature)
Examiner

Date:

Place:

Abstract

Veermata Jijabai Technological Institute has a glorious history of 128 years. One of the oldest engineering colleges in Asia. Founded in 1887 and formerly known as the Victoria Jubilee Technical Institute, it adopted its present name in 1998. VJTI is an academically and administratively autonomous institute, however it is a part of the University of Mumbai and its degrees and diplomas are issued by the university. Due to inundating creativity and thirst for innovation of our students and with the constant guidance of our unparalleled faculty, VJTI has grown by leaps and bounds. Every year, around 500 students graduate from the college paving their way into the future. However after the students graduate, it is very difficult for the college to record information about the achievements of the student which is a key factor in the progress of the institute. The amount of activities that the students collectively do over the four years is humungous, and to maintain all these records and proofs is no easy task. It is a cumbersome process for the professors to collect this information from various students and maintain all proofs and records during accreditation. In addition to this, there are times when students perform a lot of activities in events and other extra curricular and co curricular activities, but they forget to keep a track of these certificates or dont bother to ask them and document these proofs. Hence, we decided to digitize this procedure by building VJTI Record System, which will help to organize the entire procedure in an efficient and effective way, making the process a lot more convenient.

Contents

1	Introduction	1
1.1	Problem Defination	1
2	Literature Review	2
2.1	Exisiting System	2
2.2	Motivation	3
3	Report Of The Present Investigation	4
3.1	Proposed Solution	4
3.2	Entity Relationship Diagram	5
3.2.1	Use Case Specification	6
3.3	Technologies Used	17
3.3.1	HTML	17
3.3.2	Cascading Style Sheets (CSS)	18
3.3.3	JavaScript	20
3.3.4	PHP	20
3.3.5	TCPDF	21
3.3.6	Google Charts	23
3.3.7	MYSQL	23
4	Results and Discussions	26
4.1	Student User Manual	26
4.1.1	Sign in Page	26
4.1.2	Student Home Page	27
4.1.3	Edit Personal Information	29
4.1.4	View Event	29
4.1.5	Join Event	32
4.1.6	Create New Project	32
4.1.7	Project Details	33
4.2	Staff User Manual	34
4.2.1	Login	34

4.2.2	Staff Home Page	34
4.2.3	Analytical Reports	34
4.2.4	PDF Reports	37
4.2.5	Event List	40
4.2.6	View Student Information	41
4.2.7	Create Project	43
4.2.8	Create Event	44
4.2.9	Notifications	44
4.3	Future Scope	45
5	Summary and Conclusions	46
	References	47
	Acknowledgements	48

List of Figures

3.1	Entity Relationship Diagram	6
3.2	Use Case Student Sub-System Diagram	7
3.3	Use Case Staff Sub-System Diagram	8
4.1	Login and Sign Up Page Screenshot	27
4.2	Post Login and Sign Up Page Screenshot	27
4.3	Student Details on Student Home Page	28
4.4	Internal Activities on Student Home Page	28
4.5	External Activities on Student Home Page	29
4.6	Projects on Student Home Page	30
4.7	Edit Personal Information Page	30
4.8	Event List Page	31
4.9	Event Details Page	31
4.10	Join Event Page	32
4.11	Create New Project Page	33
4.12	Project Details Page	33
4.13	Staff Login Page	34
4.14	Staff Home Page	35
4.15	Analytical Reports List Page	35
4.16	Sample Analytical Report 1	36
4.17	Sample Analytical Report 2	36
4.18	Search Analytical Report	37
4.19	PDF Reports List Page	37
4.20	Sample PDF Reports Filter 1	38
4.21	Sample PDF Reports Filter 2	38
4.22	Sample PDF Reports Filter 3	39
4.23	Sample PDF Reports 1	39
4.24	Sample PDF Reports Filter 4	40
4.25	Sample PDF Reports 2	40
4.26	Sample PDF Reports 3	41
4.27	Sample PDF Reports 4	41
4.28	Sample PDF Reports 5	42

4.29	Event List Page	42
4.30	Event List Page	43
4.31	Create Project Page	43
4.32	Create Event Page	44
4.33	Notifications Page	44

Chapter 1

Introduction

1.1 Problem Defination

University Record System basically aims at collecting and preserving all the information about the students that can be used by the institute for its development. VJTI, being an autonomous institute is accredited every three years by National Board of Accreditation giving it a national ranking. The institute has to provide all information about the students which include academics, co-curricular and extra-curricular. Internal Accreditation Officer is the professor who is in-charge of this entire procedure. He can design various statistical reports showing trend analysis which can also be used to make improvements in the present curriculum structure. Also, he can request the teachers to provide him with the relevant information. So, basically the student will upload all the information about his co-curricular and extra-curricular activities along with the relevant documents which will be used by the teacher to authenticate the information. Curricular information includes projects, internships, technical competitions, hackathon, research activities and scholarships. Extra-curricular information includes sports, cultural activities and managerial activities. Also, the information regarding the student grading will directly be taken from the Examination Department to maintain authenticity. The teacher can also view information of a particular student to see his progress. The main aim is to provide all this information with security, concurrency, openness and transparency. There will be a strict control on access. This is just a brief overview of the system and its functionalities. The topics will be covered in detail in the following sections.

Chapter 2

Literature Review

2.1 Existing System

As discussed in the introduction above, VJTI has obtained the accreditation of National Board of Accreditation (NBA). The proposed project digitizes the current manual system of record maintenance for accreditation. Recently, various hardships were faced by students and faculty due to lack of records of student activities. Various student activities and organizations are active in the college throughout the year. These activities are completely managed and executed by student committees. Also, as a part of the Sports Festival, students play sports at inter and intra-college level throughout the year. Furthermore, student organizations like Society of Robotics and Automation (SRA), Society of Automobile Engineering (SAE) etc. represent college in national and international competitions. The college management finds it difficult to maintain the records of the students who participate in these activities and organizations. Recently, the professor in-charge of coordination with the accreditors asked all the students to provide a list of all the activities they were engaged in VJTI and a certificate of the same as a proof within a span of few days. The students submitted the certificates and the professor scanned them. Also, many students failed to submit the documents and thus all the activities were not recorded. The professor then generated a report of the activities of the students manually and sent it for accreditation. Secondly, when students approached the faculty members for recommendations for further studies, the concerned professors had no record of the activities of students. The students had to explicitly provide their certificates and class rank list which the professors referred to for writing recommendations. This led to discrepancies in the procedure and delayed the submission of students applications.

2.2 Motivation

On seeing these difficulties, we decided to ease out the process of accreditation. We observed that the root cause of these difficulties was unavailability of a centralized system to record the student activities. Thus, we came up with an idea to implement a University Record System for the college to ease the process of accreditation in future. After communicating with the professors, we realized the difficulties faced in the execution of the procedure. Also, the certificates which the students submitted were scanned and sent to accreditors but they were not recorded officially. Thus, in the next accreditation the same procedure would have to be repeated again. Thus we decided to implement University Record System so that it becomes easier for future accreditations. The remainder of the document defines the scope of the project and technologies used for implementation.

Chapter 3

Report Of The Present Investigation

3.1 Proposed Solution

The system proposed aims to solve the issues raised by the problem statement in the existing system. The system will log all student records by categorizing them in 4 major categories namely grades, internal activities, projects, and external activities. The grades of every student will be logged directly from the examination department. The academic record of the student will be mapped using the student ID. These records can be displayed in the student profile directly. Internal Activities are those which take place within the institution. College events, competitions, workshops and seminars fall under this category. External Activities are those which in which the student participates outside college. Other colleges events, hackathons, internships, coding competitions and outside workshops and seminars fall under this category.

Student internal activities broadly divided in 2 parts: co-curricular and extra-curricular. These 2 parts can be further sub-divided into sub-categories according to various festivals or sports or events or competitions. To standardize and group the participation of student in these activities, every activity will be represented as a set of events. The internal events can be created by a Faculty member. The event will be then open for students to participate / manage in. The student will then send a request via the system to the concerned faculty member with supporting documents to update the records. Any such requests issued by the students will be sent to approval to the concerned faculty members. The faculty member can then approve the request and that particular student's activity will be updated.

Student is given a provision to add an External Activity through forms. The student enters details of the event and his/her role in the event. Along with these, he/she also enters relevant document proof to his/her participation. This document proof is then verified by the concerned faculty (the class teacher) and necessary changes are made in the student profile.

For maintaining the records of the projects, a student will have to first send an approval request to the project mentor. If there is no mentor then the students can request any faculty who might be aware of the on-going project to be a stand-in mentor. Once the project request is approved other team members can request to join the project and the same can be approved by the project mentor. The project status can be updated by the mentor as in weather the project is pending or completed.

For accreditation purposes, the Teacher can select and filter the results according to various criteria. He can choose to generate reports only pertaining a particular batch or class. He can also group the records according to various events and activities. This will provide flexibility and will be able to carter to ever changing needs of the accreditation board. Two types of reports are generated by the system: Analytical and PDF reports.

Analytical Reports are statistical reports which show graphs of various parameters. These reports give information in the form of bar, pie and line graphs. Analysis of these charts give invaluable information regarding the performance of students in academics, extra-curricular and co-curricular activities. PDF reports are the type of reports which generate PDFs containing a table of all students activities and the document proofs which have been uploaded. These reports are in the form which NBA demands. These reports can be printed and submitted to serve the accreditation process.

3.2 Entity Relationship Diagram

Refer figure 3.1.

The major entities in the ER diagram are:

- Student: This entity will hold the basic details of every student. There will be a unique student id for every student
- Project : Every project entity will have the details regarding all individual and group projects that are under progress. Every project will have a mentor.
- Project Members This is a weak entity. This will hold a map for students that are working on every project.

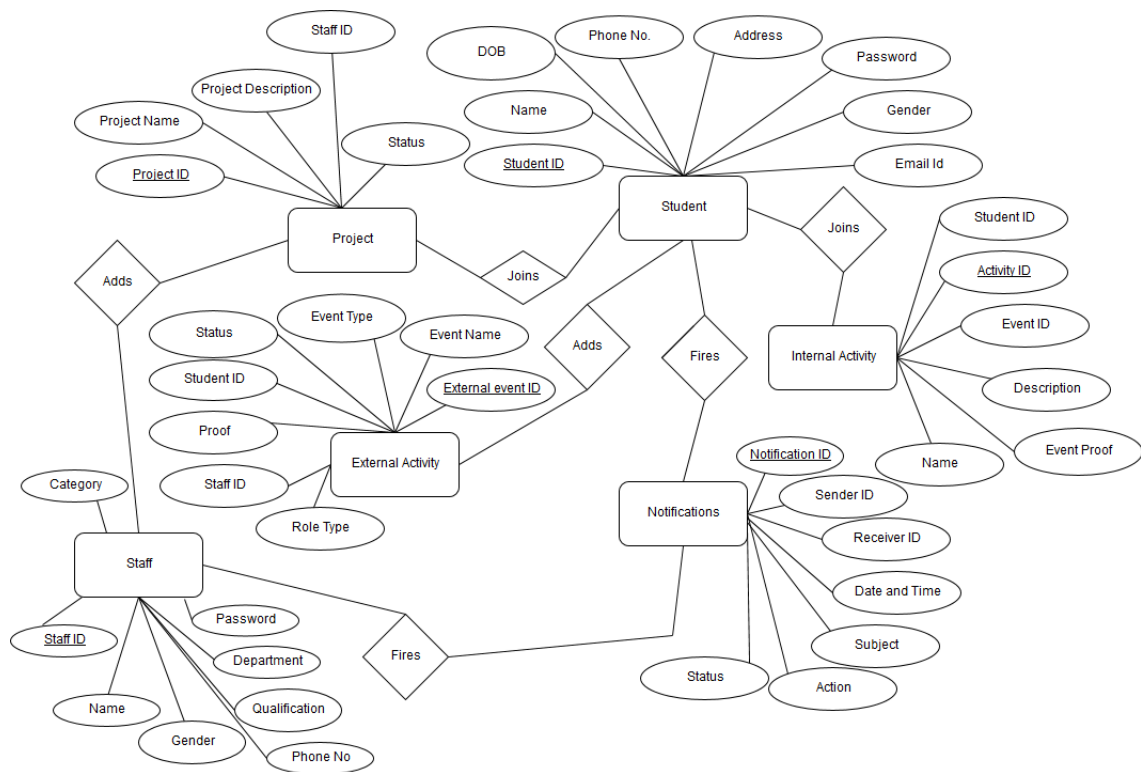


Figure 3.1: Entity Relationship Diagram

- Internal activities and External activities The students will log activities for every event occurring inside and outside the college campus.
- Staff A staff will have the authority to create and approve events and requests respectively.
- Events Every internal activity can be created by a teacher. A student that will participate in a particular event and has to forward the approval request to the concerned teacher.

3.2.1 Use Case Specification

Edit Profile

- Use Case Name : Edit Profile
- Primary Actors : Teacher, Student

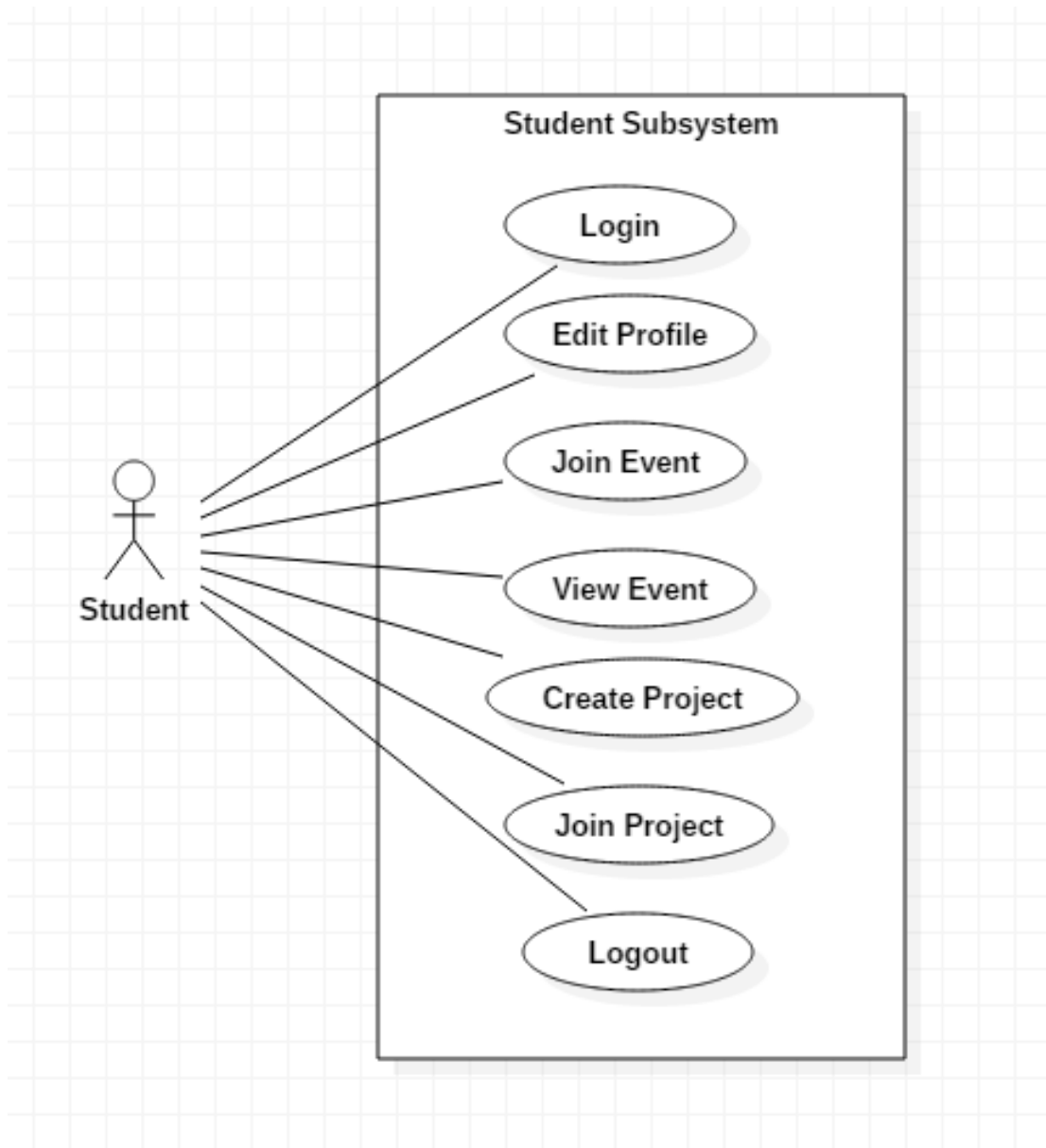


Figure 3.2: Use Case Student Sub-System Diagram

- Description : This use case allows the user to edit his/her own personal details like alternate email-id, phone number, etc
- Pre-Condition : The user should have been logged into the system to edit his/her details
- Basic Flow :

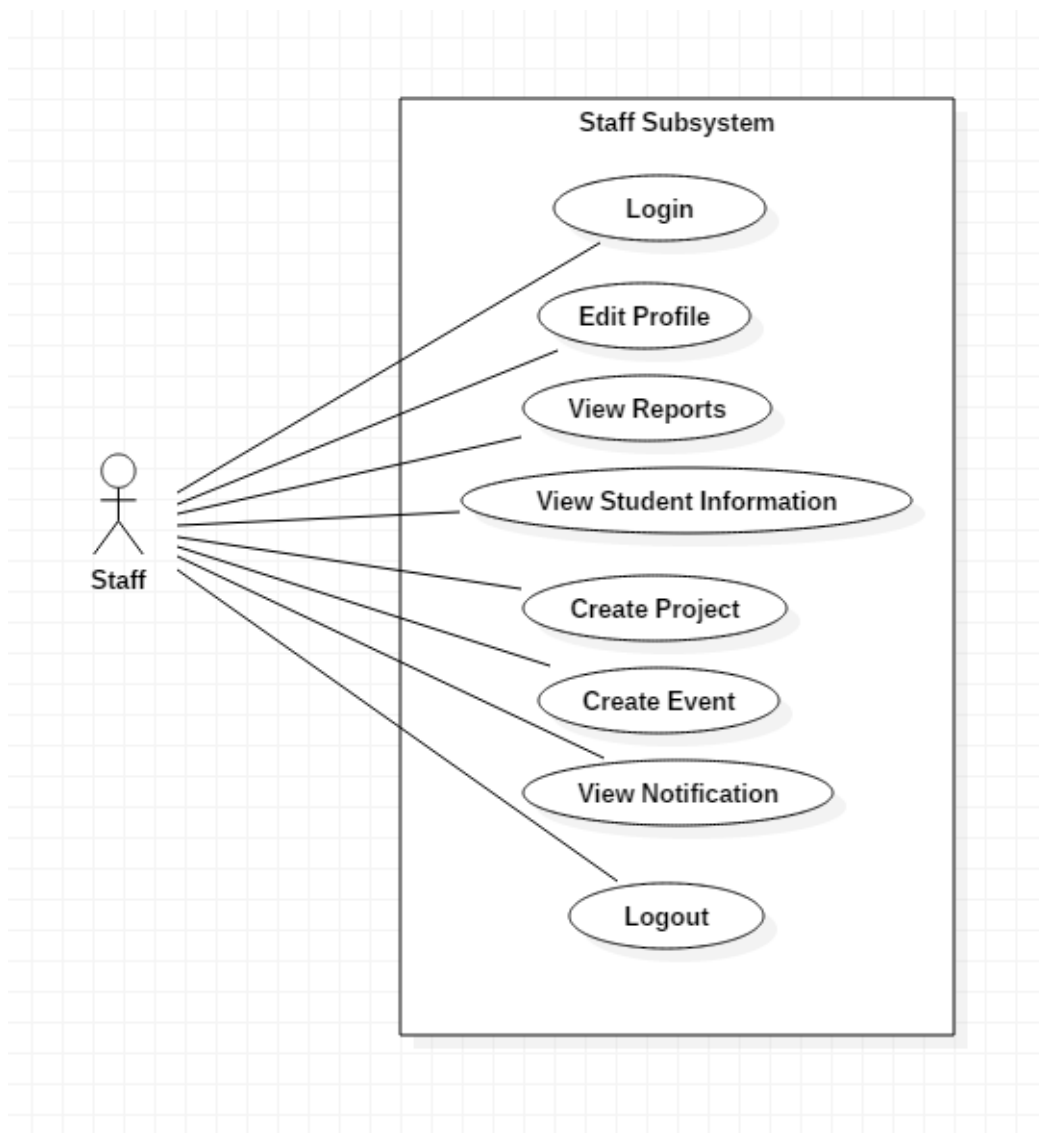


Figure 3.3: Use Case Staff Sub-System Diagram

1. The user clicks on edit profile on right hand corner of screen by user
 2. Edit profile page is displayed by system
 3. The user edits his/her details by user
 4. The user confirms the editing of the details by user
 5. Page is reloaded by system
- Post Condition : The users details have been edited and updated into

the database

- Alternate Flow :
 1. The details entered are not valid or in a proper format and alert is generated
 2. System is timed out
- Special Condition :
 1. 1000 concurrent users at a time
 2. System Failure
 3. No activity for 1440 seconds

Create Project

- Use Case Name : Create Project
- Primary Actors : Student
- Description : This use case allows the user to create a project that he/she will start working on or has started working on with the mentor
- Pre-Condition : The user should have been logged into the system and have access to the create project page
- Basic Flow :
 1. The user clicks on create project page on the side bar - user
 2. Create project page is displayed - system
 3. The user enters details like name of project, description, and mentor id with name - user
- Post Condition : The user has created project successfully and notification has been fired to concerned faculty
- Alternate Flow :
 1. Mentor Id entered is wrong, hence page is reloaded and no notification is fired.
 2. System is timed out
- Special Condition :

1. 1000 concurrent users at a time
2. System Failure
3. No activity for 1440 seconds

Join Project

- Use Case Name : Join Project
- Primary Actors : Student
- Description : This use case allows the user to join a project where he has performed a task
- Pre-Condition : The user should have been logged into the system and have access to the join project page
- Basic Flow :
 1. The user clicks on join project page on the side bar - user
 2. Join project page is displayed - system
 3. List of all projects being done are displayed - system
 4. The user joins the project he chooses to - user
 5. Join project page is displayed - system
 6. The user has to enter details like role, role type, and description.
- Post Condition : The user has joined project and notification has been fired to concerned faculty/mentor
- Alternate Flow :
 1. System is timed out
- Special Condition :
 1. 1000 concurrent users at a time
 2. System Failure
 3. No activity for 1440 seconds

View Reports

- Use Case Name : View Reports
- Primary Actors : Teacher
- Description : This use case allows the user to view reports, charts, and an analysis of all the data in the database
- Pre-Condition : The user should have been logged into the system and have access to the reports page
- Basic Flow :
 1. The user clicks on reports page on the side bar - user
 2. Reports page is displayed - system
 3. List of all reports are displayed - system
 4. The user chooses the report he wants to view - user
- Post Condition : The user has viewed the report
- Alternate Flow :
 1. System is timed out
- Special Condition :
 1. 1000 concurrent users at a time
 2. System Failure
 3. No activity for 1440 seconds

Log Out

- Use Case Name : Log Out
- Primary Actors : Teacher, Student
- Description : This use case allows the user to log out of the system
- Pre-Condition : The user should have been logged into the system
- Basic Flow :
 1. The user asks to log out of the system - user

- 2. Home page is displayed - system
- Post Condition:
 - 1. The user is logged out of the system
- Alternate Flow :
 - 1. System is timed out
- Special Condition :
 - 1. 1000 concurrent users at a time
 - 2. System Failure
 - 3. No activity for 1440 seconds

View Notifications

- Use Case Name : View Notifications
- Primary Actors : Teacher, Student
- Description : This use case allows the user to view the notifications that have been fired by student
- Pre-Condition : The user should have been logged into the system and have access to the view notifications page
- Basic Flow :
 - 1. The user clicks on view notifications page on the side bar - user
 - 2. The page is displayed - system
 - 3. The user either approves or rejects request - user
 - 4. Page is reloaded - system
- Post Condition:
 - 1. The student will know if request is approved or rejected
- Alternate Flow :
 - 1. System is timed out
- Special Condition :

1. 1000 concurrent users at a time
2. System Failure
3. No activity for 1440 seconds

Create Event

- Use Case Name : Create Event
- Primary Actors : Teacher, Student
- Description : This use case allows the user to create an event that is happening in college
- Pre-Condition : The user should have been logged into the system and have access to the create event page
- Basic Flow :
 1. The user clicks on create project page on the side bar - user
 2. Create event page is displayed - system
 3. The user enters details like name of event, description, start date, end date and event type - user
 4. Event is created with a unique id and entered into database- system
- Post Condition:
 1. Event is created with a unique id and entered into database
- Alternate Flow :
 1. System is timed out
 2. Event Already Exists
- Special Condition :
 1. 1000 concurrent users at a time
 2. System Failure
 3. No activity for 1440 seconds

Create Project

- Use Case Name : Create Project
- Primary Actors : Teacher, Student
- Description : This use case allows the user to create a project that is under him
- Pre-Condition : The user should have been logged into the system and have access to the create project page
- Basic Flow :
 1. The user clicks on create project page on the side bar - user
 2. Create project page is displayed - system
 3. The user enters details like name of project, description - user
- Post Condition:
 1. The user has created project successfully
- Alternate Flow :
 1. System is timed out
- Special Condition :
 1. 1000 concurrent users at a time
 2. System Failure
 3. No activity for 1440 seconds

View Student Information

- Use Case Name : View Student Information
- Primary Actors : Teacher
- Description : This use case allows the user to view information about a specific student
- Pre-Condition : The user should have been logged into the system and have access to the view students page
- Basic Flow :

1. The user clicks on student information page on the side bar - user
 2. View student information page is displayed - system
 3. The user enters student id for which he wishes to see details user
 4. The details of the student are displayed - system
- Post Condition:
 1. The user has viewed details and information of the student
 - Alternate Flow :
 1. System is timed out
 2. The teacher has entered a wrong student id
 - Special Condition :
 1. 1000 concurrent users at a time
 2. System Failure
 3. No activity for 1440 seconds

Sign In

- Use Case Name : Sign In
- Primary Actors : Teacher, Student
- Description : This use case allows the user to log into the system
- Pre-Condition : The user should have access to the home page of the website
- Basic Flow :
 1. The user enters username and password in the system - user
 2. The user also selects student or staff - user
 3. The system validates username and password - system
 4. The student home page is displayed - system
 5. Page is reloaded by system
- Post-Condition : The user is logged into the system
- Alternate Flow :

1. The username or password is wrongly entered
 2. System is timed out
- Special Condition :
 1. 1000 concurrent users at a time
 2. System Failure
 3. No activity for 1440 seconds

View Events

- Use Case Name : Join Events
- Primary Actors : Student
- Description : This use case allows the user to view all the events of the college that are undertaken
- Pre-Condition : The user should have been logged into the system to view the events displayed
- Basic Flow :
 1. The user clicks on view events on the left side bar - user
 2. The View events page is displayed - system
 3. The user has option of viewing more details of the event by clicking on view more
 4. Details of the event along with images are presented - system
- Post-Condition : The user has viewed all the events of his choice and all the details have been displayed to him
- Alternate Flow :System is timed out
- Special Condition :
 1. 1000 concurrent users at a time
 2. System Failure
 3. No activity for 1440 seconds

Join Events

- Use Case Name : Join Events
- Primary Actors : Student
- Description : This use case allows the user to join an event where he has performed a task as an organiser or a participant
- Pre-Condition : The user should have been logged into the system and have access to the view events page
- Basic Flow :
 1. The user clicks on view events page on the side bar - user
 2. View events page is displayed - system
 3. List of all events in college are displayed - system
 4. The user joins the event he chooses to - user
 5. Join event page is displayed - system
 6. The user has to enter details like role, role type, and description.
- Post-Condition : The user has joined event and notification has been fired to concerned faculty
- Alternate Flow :
 1. The event end date has passed, hence user cannot join
 2. System is timed out
- Special Condition :
 1. 1000 concurrent users at a time
 2. System Failure
 3. No activity for 1440 seconds

3.3 Technologies Used

3.3.1 HTML

Hypertext Markup Language (HTML) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS)

and JavaScript it forms a triad of cornerstone technologies for the World Wide Web. Web browsers receive HTML documents from a webserver or from local storage and render them into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects, such as interactive forms, may be embedded into the rendered page. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets. Tags such as `` and `<input />` introduce content into the page directly. Others such as `<p>...</p>` surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

HTML can embed programs written in a scripting language such as JavaScript which affect the behavior and content of web pages. Inclusion of CSS defines the look and layout of content. The World Wide Web Consortium (W3C), maintainer of both the HTML and the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997.

HTML is the standard markup language for creating Web pages.

- HTML stands for Hyper Text Markup Language.
- HTML describes the structure of Web pages using markup.
- HTML elements are the building blocks of HTML pages.
- HTML elements are represented by tags.
- HTML tags label pieces of content such as "heading", "paragraph", "table", and so on.
- Browsers do not display the HTML tags, but use them to render the content of the page.

3.3.2 Cascading Style Sheets (CSS)

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language. Although most often used to set the visual style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any XML document, including plain XML, SVG and XUL, and is applicable to rendering

in speech, or on other media. Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications.

CSS is designed primarily to enable the separation of document content from document presentation, including aspects such as the layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple HTML pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

Separation of formatting and content makes it possible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices. It can also display the web page differently depending on the screen size or viewing device. Readers can also specify a different style sheet, such as a CSS file stored on their own computer, to override the one the author specified.

Changes to the graphic design of a document (or hundreds of documents) can be applied quickly and easily, by editing a few lines in the CSS file they use, rather than by changing markup in the documents.

The CSS specification describes a priority scheme to determine which style rules apply if more than one rule matches against a particular element. In this so-called cascade, priorities (or weights) are calculated and assigned to rules, so that the results are predictable.

Before CSS, nearly all presentational attributes of HTML documents were contained within the HTML markup. All font colors, background styles, element alignments, borders and sizes had to be explicitly described, often repeatedly, within the HTML. CSS lets authors move much of that information to another file, the style sheet, resulting in considerably simpler HTML.

For example, headings (h1 elements), sub-headings (h2), sub-sub-headings (h3), etc., are defined structurally using HTML. In print and on the screen, choice of font, size, color and emphasis for these elements is presentational.

Before CSS, document authors who wanted to assign such typographic characteristics to, say, all h2 headings had to repeat HTML presentational markup for each occurrence of that heading type. This made documents more complex, larger, and more error-prone and difficult to maintain. CSS allows the separation of presentation from structure. CSS can define color, font, text alignment, size, borders, spacing, layout and many other typographic characteristics, and can do so independently for on-screen and printed views. CSS also defines non-visual styles, such as reading speed and emphasis for

aural text readers.

3.3.3 JavaScript

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

Although there are strong outward similarities between JavaScript and Java, including language name, syntax, and respective standard libraries, the two are distinct languages and differ greatly in their design. JavaScript was influenced by programming languages such as Self and Scheme

JavaScript is also used in environments that are not Web-based, such as PDF documents, site-specific browsers, and desktop widgets. Newer and faster JavaScript virtual machines (VMs) and platforms built upon them have also increased the popularity of JavaScript for server-side Web applications. On the client side, developers have traditionally implemented JavaScript as an interpreted language, but more recent browsers perform just-in-time compilation. Programmers also use JavaScript in video-game development, in crafting desktop and mobile applications, and in server-side network programming with run-time environments such as Node.js.

3.3.4 PHP

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

PHP code may be embedded into HTML or HTML5 markup, or it can be used in combination with various web template systems, web content management systems and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as

a Common Gateway Interface (CGI) executable. The web server software combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications.

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

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

The original, only complete and most widely used PHP implementation is powered by the Zend Engine and known simply as PHP. To disambiguate it from other implementations, it is sometimes unofficially referred to as "Zend PHP". The Zend Engine compiles PHP source code on-the-fly into an internal format that it can execute, thus it works as an interpreter. It is also the "reference implementation" of PHP, as PHP has no formal specification, and so the semantics of Zend PHP define the semantics of PHP itself. Due to the complex and nuanced semantics of PHP, defined by how Zend works, it is difficult for competing implementations to offer complete compatibility.

PHP's single-request-per-script-execution model, and the fact that the Zend Engine is an interpreter, leads to inefficiency; as a result, various products have been developed to help improve PHP performance. In order to speed up execution time and not have to compile the PHP source code every time the web page is accessed, PHP scripts can also be deployed in the PHP engine's internal format by using an opcode cache, which works by caching the compiled form of a PHP script (opcodes) in shared memory to avoid the overhead of parsing and compiling the code every time the script runs. An opcode cache, Zend OpCache, is built into PHP since version 5.5. Another example of a widely used opcode cache is the Alternative PHP Cache (APC), which is available as a PECL extension.

While Zend PHP is still the most popular implementation, several other implementations have been developed. Some of these are compilers or support JIT compilation, and hence offer performance benefits over Zend PHP at the expense of lacking full PHP compatibility.

3.3.5 TCPDF

Open Source PHP class for generating PDF documents. The major features are :

- No external libraries are required for the basic functions
- All standard page formats, custom page formats, custom margins and units of measure
- UTF-8 Unicode and Right-To-Left languages
- TrueTypeUnicode, OpenTypeUnicode, TrueType, OpenType, Type1 and CID-0 fonts
- Font subsetting
- Methods to publish some XHTML + CSS code, Javascript and Forms
- Images, graphic (geometric figures) and transformation methods;
- Supports JPEG, PNG and SVG images natively, all images supported by GD (GD, GD2, GD2PART, GIF, JPEG, PNG, BMP, XBM, XPM) and all images supported via ImagMagick ([http: www.imagemagick.org/www/formats.html](http://www.imagemagick.org/www/formats.html));
- 1D and 2D barcodes: CODE 39, ANSI MH10.8M-1983, USD-3, 3 of 9, CODE 93, USS-93, Standard 2 of 5, Interleaved 2 of 5, CODE 128 A/B/C, 2 and 5 Digits UPC-Based Extention, EAN 8, EAN 13, UPC-A, UPC-E, MSI, POSTNET, PLANET, RMS4CC (Royal Mail 4-state Customer Code), CBC (Customer Bar Code), KIX (Klant index - Customer index), Intelligent Mail Barcode, Onecode, USPS-B-3200, CODABAR, CODE 11, PHARMACODE, PHARMACODE TWO-TRACKS, Datamatrix ECC200, QR-Code, PDF417;
- ICC Color Profiles, Grayscale, RGB, CMYK, Spot Colors and Transparencies;
- Automatic page header and footer management;
- Document encryption up to 256 bit and digital signature certifications;
- Transactions to UNDO commands;
- PDF annotations, including links, text and file attachments;
- Text rendering modes (fill, stroke and clipping);
- Multiple columns mode;
- No-write page regions;

- Bookmarks and table of content;
- Text hyphenation;
- Text stretching and spacing (tracking/kerning);
- Automatic page break, line break and text alignments including justification;
- Automatic page numbering and page groups;
- Move and delete pages;
- Page compression (requires php-zlib extension);
- XObject templates;
- PDF/A-1b (ISO 19005-1:2005) support.

3.3.6 Google Charts

The Google Chart API is an interactive Web service (now deprecated) that creates graphical charts from user-supplied data. Google servers create a PNG image of a chart from data and formatting parameters specified by a user's HTTP request. The service supports a wide variety of chart information and formatting. Users may conveniently embed these charts in a Web page by using a simple image tag.

Originally the API was Google's internal tool to support rapid embedding of charts within Google's own applications (like Google Finance for example). Google figured it would be a useful tool to open up to web developers. It officially launched on December 6, 2007.

Currently, line, bar, pie, and radar charts, as well as Venn diagrams, scatter plots, sparklines, maps, google-o-meters, and QR codes are supported.

3.3.7 MYSQL

MySQL is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius' daughter, and "SQL", the abbreviation for Structured Query Language. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation.

For proprietary use, several paid editions are available, and offer additional functionality.

MySQL is a central component of the LAMP open-source web application software stack (and other "AMP" stacks). LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python". Applications that use the MySQL database include: TYPO3, MODx, Joomla, WordPress, phpBB, MyBB, and Drupal. MySQL is also used in many high-profile, large-scale websites, including Google (though not for searches), Facebook, Twitter, Flickr, and YouTube

Major features as available in MySQL 5.6:

- A broad subset of ANSI SQL 99, as well as extensions
- Cross-platform support
- Stored procedures, using a procedural language that closely adheres to SQL/PSM[62]
- Triggers
- Cursors
- Updatable views
- Online DDL when using the InnoDB Storage Engine.
- Information schema
- Performance Schema that collects and aggregates statistics about server execution and query performance for monitoring purposes.[63]
- A set of SQL Mode options to control runtime behavior, including a strict mode to better adhere to SQL standards.
- X/Open XA distributed transaction processing (DTP) support; two phase commit as part of this, using the default InnoDB storage engine
- Transactions with savepoints when using the default InnoDB Storage Engine. The NDB Cluster Storage Engine also supports transactions.
- ACID compliance when using InnoDB and NDB Cluster Storage Engines[64]
- SSL support
- Query caching

- Sub-SELECTs (i.e. nested SELECTs)
- Built-in replication support (i.e., master-master replication and master-slave replication) with one master per slave, many slaves per master. Multi-master replication is provided in MySQL Cluster,[66] and multi-master support can be added to unclustered configurations using Galera Cluster.
- Full-text indexing and searching[a]
- Embedded database library

Chapter 4

Results and Discussions

The system currently focuses on the digitisation of the accreditation procedure. The system will help all the faculty to get any information about any student at finger tips. It will also help to evaluate the overall progress of the institute in different sectors like academics, sports, extra-curricular activities and many others. The college management can accordingly plan the activities to strengthen the weaker areas. The system can also be expanded in future to cover other functionalities and more modules can be integrated into a single system.

4.1 Student User Manual

4.1.1 Sign in Page

Refer figure 4.1.

This is the dashboard for users where users can sign into their account by entering their username and password into the system. The system validates the same with the database and returns the login status. On successful login, the user will be redirected to the user home page else he will be redirected to the login page again. The user has to select the Student radio button to validate login.

A new user can also sign up from this screen. He enters the basic details on the dashboard and can add additional details from the user home page. Refer figure 4.2. After signing up, the basic details which were entered during sign up are displayed. And the rest of the details are shown as default values.

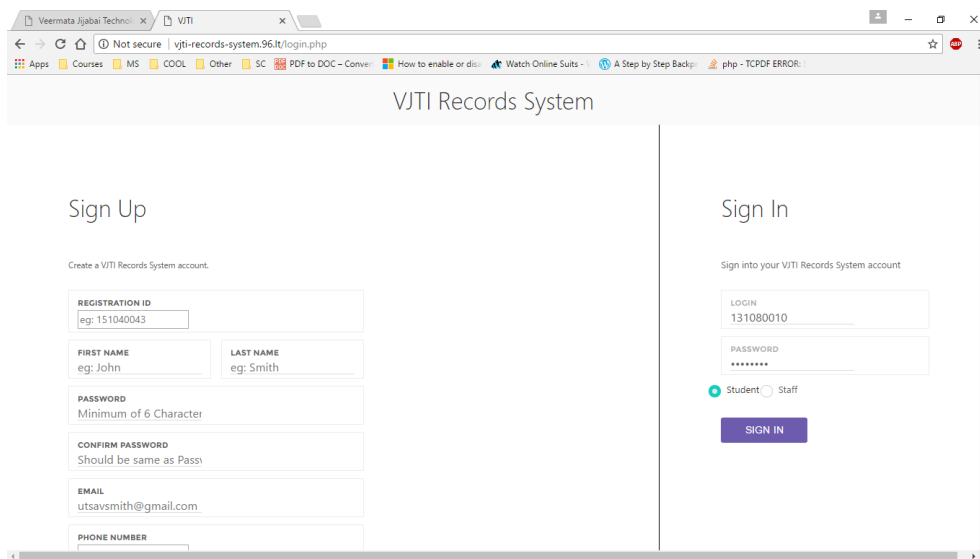


Figure 4.1: Login and Sign Up Page Screenshot

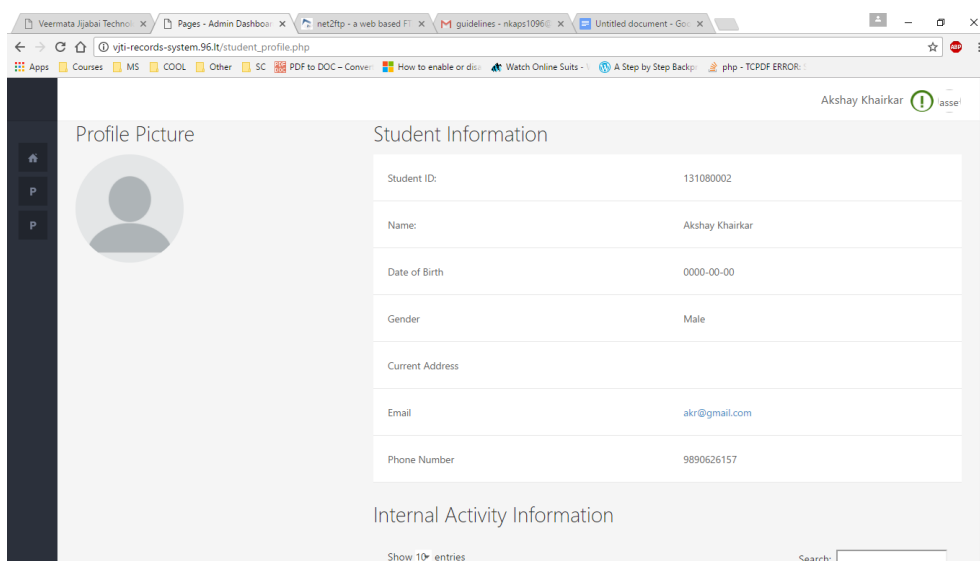


Figure 4.2: Post Login and Sign Up Page Screenshot

4.1.2 Student Home Page

Refer figure 4.3.

This is the user dashboard. This page shows all the basic user information such as Name, Address, Class, Email, Branch etc. The sidebar gives options of all the activities which the user can perform using the system. This also

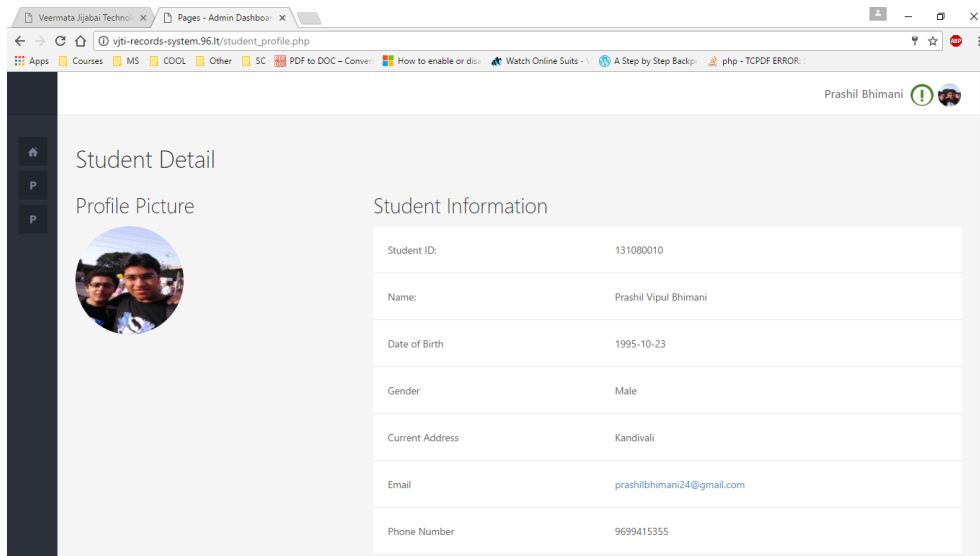


Figure 4.3: Student Details on Student Home Page

gives information about internal activities, external activities and projects which the user is/was a part of.

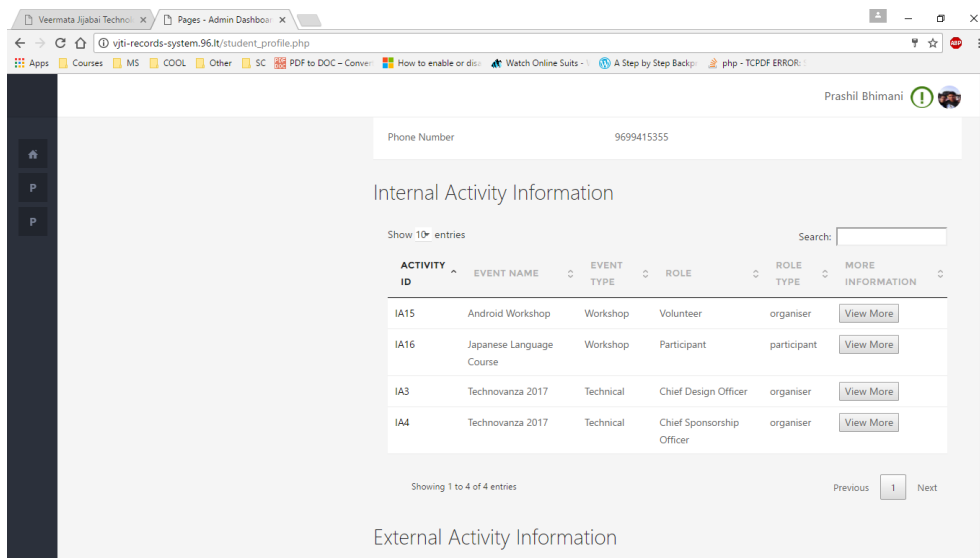


Figure 4.4: Internal Activities on Student Home Page

Refer figure 4.4.

This shows the list of internal activities which the user is/was a part of. Also there is a view more option which redirects to a page which gives

detailed information about the activity. Internal activities are the activities which the users participates in within college. Refer figure 4.5.

External Activity Information

Show 10 entries

Search:

ACTIVITY ID	EVENT NAME	EVENT TYPE	ROLE	ROLE TYPE	MORE INFORMATION
EA23	Technovanza 2017	Technical Festival	Chief Design Officer	organiser	View More
EA3	ACM ICPC	Competition	Team member	participant	View More
EA4	Infosys Hackathon	Hackathon	Team member	participant	View More
EA6	SAP Hackathon	Hackathon	Developer	organiser	View More
EA7	BSE Bull Run	Marathon	Participant	participant	View More
EA8	Mumbai Marathon	Marathon	Participant	participant	View More
EA9	Robowars COEP	Competition	Participant	participant	View More

Showing 1 to 7 of 7 entries

Previous 1 Next

Figure 4.5: External Activities on Student Home Page

This shows the list of external activities which the user is/was a part of. Also there is a view more option which redirects to a page which gives detailed information about the activity. External activities are the activities which are conducted outside the institute by other institutes/organizations. Refer figure 4.6.

It shows a list of all the projects which the user has done in the institute. The view more button redirects the user to a page which shows the detailed explanation of the project.

4.1.3 Edit Personal Information

Refer figure 4.7.

This page provides the functionality to edit the profile information. It shows the current data which is stored in the database. The user can change certain entries which are updated in the database when the user presses the Update Details button.

4.1.4 View Event

Refer figure 4.8.

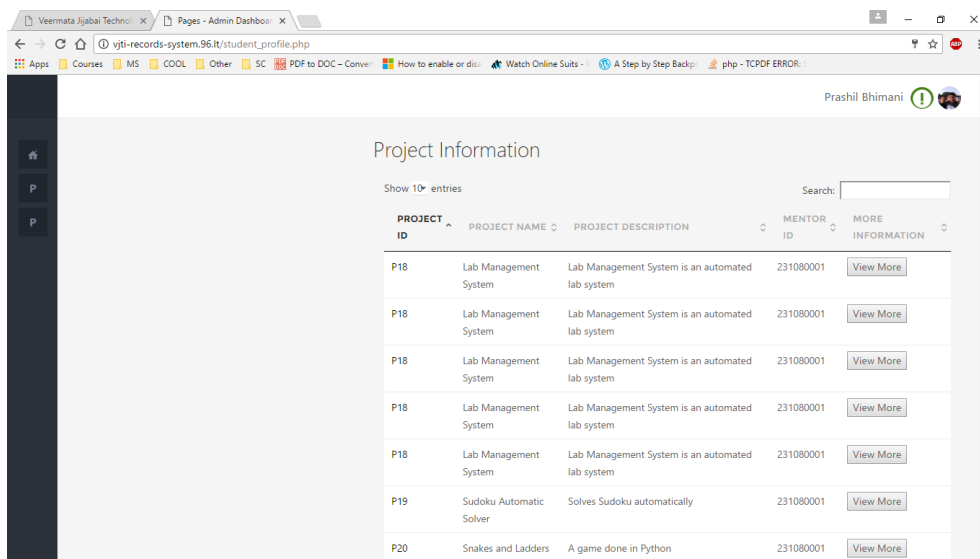


Figure 4.6: Projects on Student Home Page

Edit Personal Information

REGISTRATION ID: 131080010

FIRST NAME: Prashil MIDDLE NAME: Vipul LAST NAME: Bhimani

FATHER'S NAME: Vipul MOTHER'S NAME: Unnathi

DATE OF BIRTH: 23-10-1995

Gender: ☒ Male ☐ Female

EMAIL: prashilbhimani24@gmail.com

ALTERNATE EMAIL: prashilbhimani24@gmail.com

PHONE NUMBER: 9899413355

ALTERNATE PHONE NUMBER: 9899413355

Physically Disabled: ☒ Yes ☐ No

CATEGORY: Open

HIC PERCENTAGE: 100

HIC PERCENTAGE: 100

Figure 4.7: Edit Personal Information Page

This page provides a list of all the events which have been created by the faculty. The user can select the event in which he has participated/organized. The Join Event button redirects the user to a page where the user enters the details and relevant proof of his participation in the respective event. This proof of participation will be approved by the concerned faculty member who receives a notification of the action. After the request to join is approved,

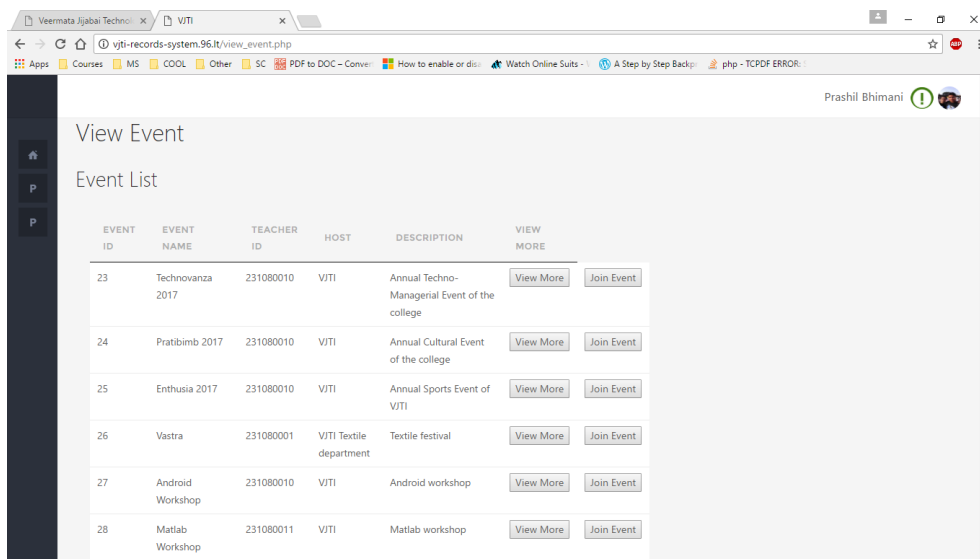


Figure 4.8: Event List Page

this event is added in the user activities in the database. Refer figure 4.9.

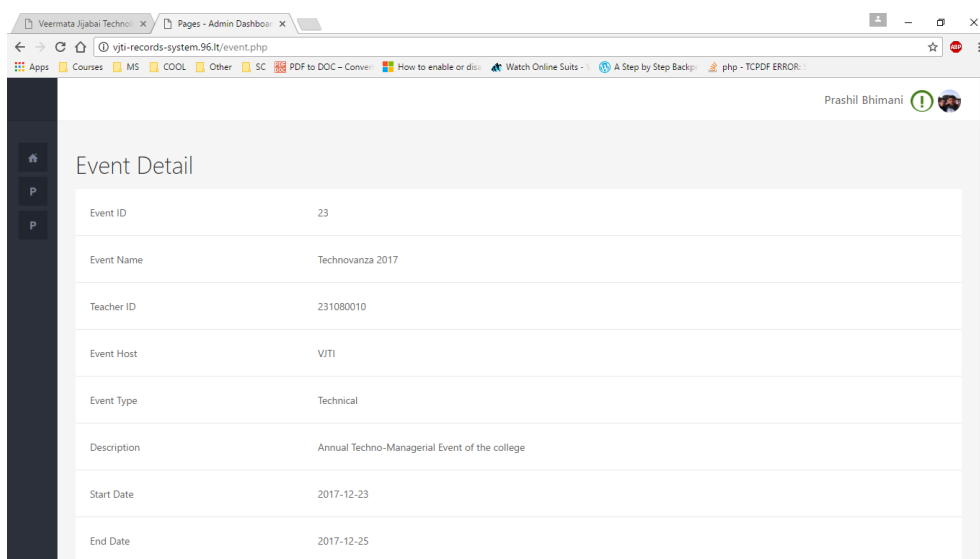


Figure 4.9: Event Details Page

On clicking the View More button on the previous page, the user is redirected to a page which shows the information of the corresponding event. This page will give you the information of the event such as the Event ID, Event name, Teacher ID who has created the event, the host of the event,

the type of the event, the description of the event and the start date and the end date of the event. This page also shows the photos of the event which have been uploaded by the faculty.

4.1.5 Join Event

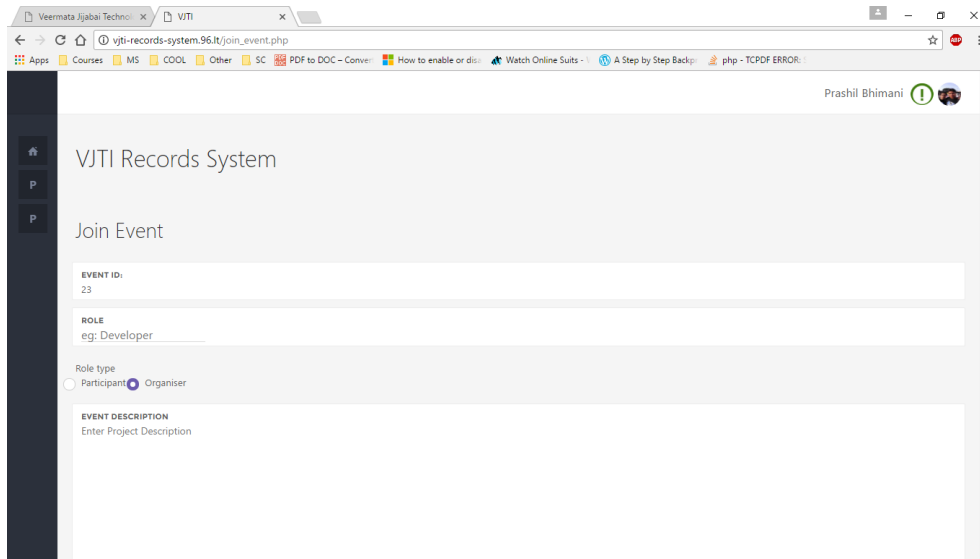
The screenshot shows a web browser window with the URL 'vjti-records-system.96.it/join_event.php'. The page title is 'VITI Records System'. On the left, there is a dark sidebar with a home icon and two 'P' icons. The main content area is titled 'Join Event'. It contains a form with the following fields: 'EVENT ID:' with the value '23', 'ROLE:' with the value 'eg: Developer', 'Role type:' with radio buttons for 'Participant' and 'Organiser' (the 'Organiser' button is selected), and 'EVENT DESCRIPTION:' with the placeholder text 'Enter Project Description'. In the top right corner, the user's name 'Prashil Bhimani' is displayed next to a green status icon and a profile picture.

Figure 4.10: Join Event Page

Refer figure 4.10.

This page provides the functionality to join the event, the user has to fill certain fields to join the event, and user inputs the event ID, the role which he has played in the event. The role type gives the whether the user was the participant or the organizer and the event description which gives the description of the event. The user has to upload the proof that he has participated in the event. The request for approval is sent to the concerned faculty.

4.1.6 Create New Project

Refer figure 4.11.

This is the page to create a new project. The user enters the details of the project and also the ID of the mentor who is guiding the project.

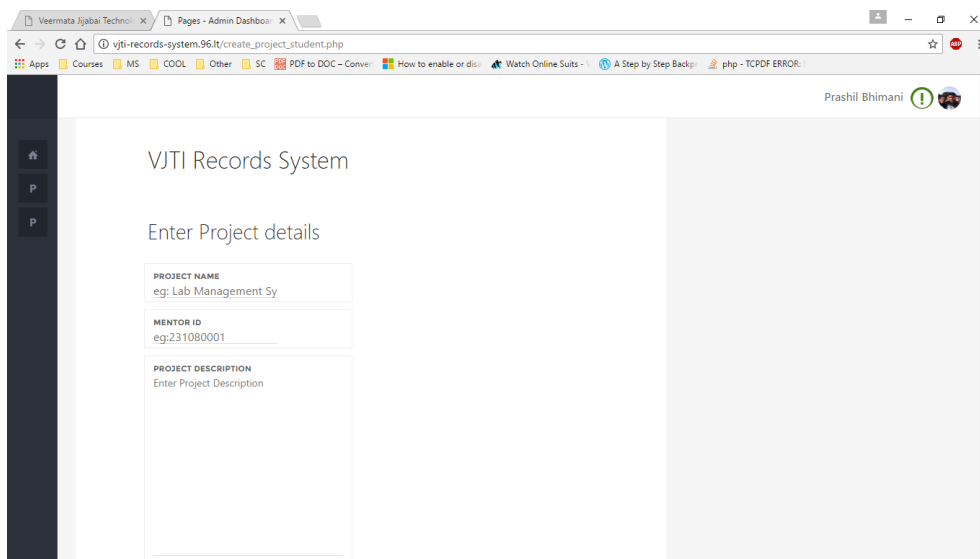


Figure 4.11: Create New Project Page

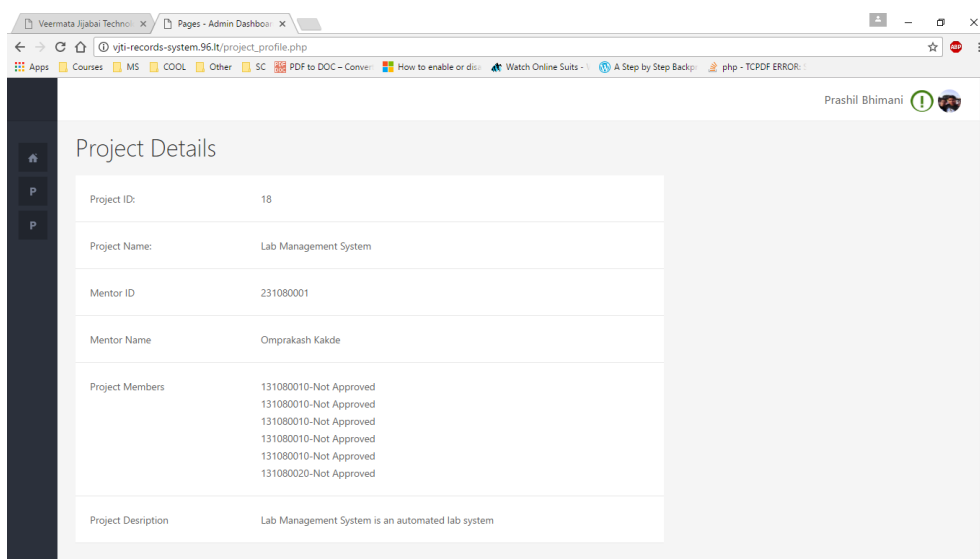


Figure 4.12: Project Details Page

4.1.7 Project Details

Refer figure 4.12.

This page shows the details of the project along with the Mentor details who is guiding the project. This also shows the status of the project. Once the Mentor the project, the status is changed to "Approved".

4.2 Staff User Manual

4.2.1 Login

VJTI Records System

Sign Up

Create a VJTI Records System account.

REGISTRATION ID
eg: 151040043

FIRST NAME
eg: John

LAST NAME
eg: Smith

PASSWORD
Minimum of 6 Character

CONFIRM PASSWORD
Should be same as Pass

EMAIL
utsavsmith@gmail.com

PHONE NUMBER

Sign In

Sign into your VJTI Records System account

LOGIN
131080010

PASSWORD

☒ Student ☐ Staff

SIGN IN

Figure 4.13: Staff Login Page

Refer figure 4.13.

This is the dashboard for users where users can sign into their account by entering their username and password into the system. The system validates the same with the database and returns the login status. On successful login, the user will be redirected to the user home page else he will be redirected to the login page again. The user has to select the Staff radio button to validate login.

4.2.2 Staff Home Page

Refer figure 4.14.

This is the user dashboard. This page shows all the basic user information such as Name, Address, Class, Email, Branch etc. The sidebar gives options of all the activities which the user can perform using the system. The user can also view reports from here, create event project and update his/her personal information.

4.2.3 Analytical Reports

Refer figure 4.15.

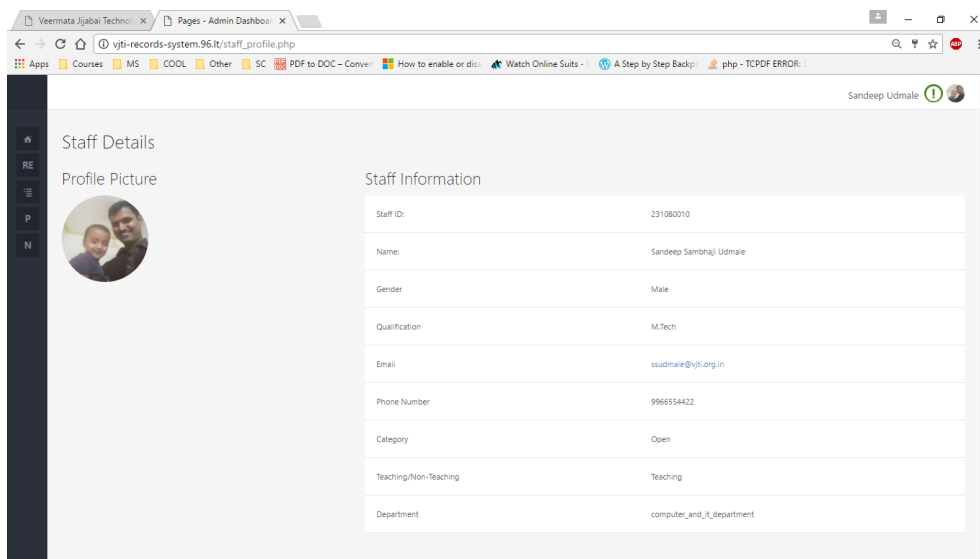


Figure 4.14: Staff Home Page

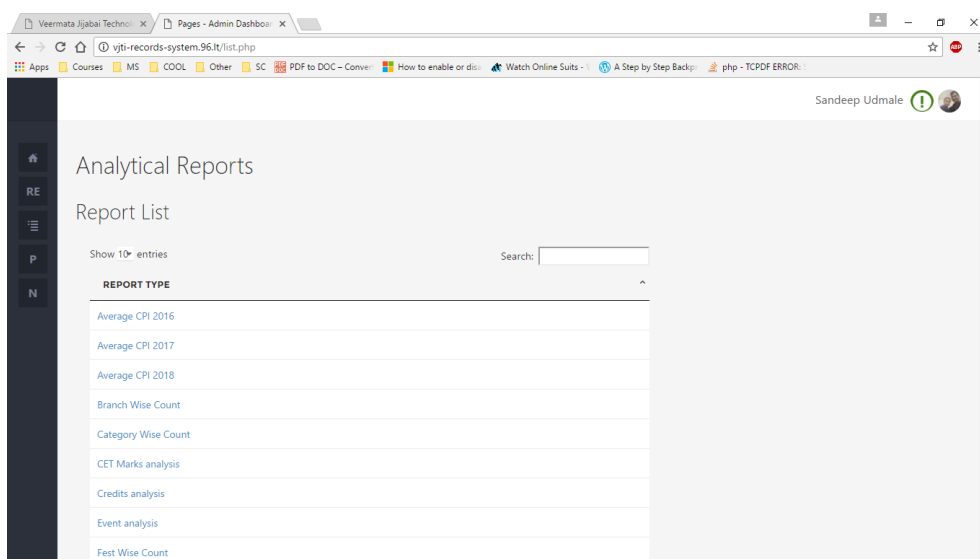


Figure 4.15: Analytical Reports List Page

This gives analytical reports of the data which give various statistics of the students and its information, these reports are generated using google charts. It contains various information such as the average CPI of a particular year, batch-wise count, event and festival statistics.

Refer figure 4.16 and 4.17.

Given above are some of the statistical reports of the data. These re-

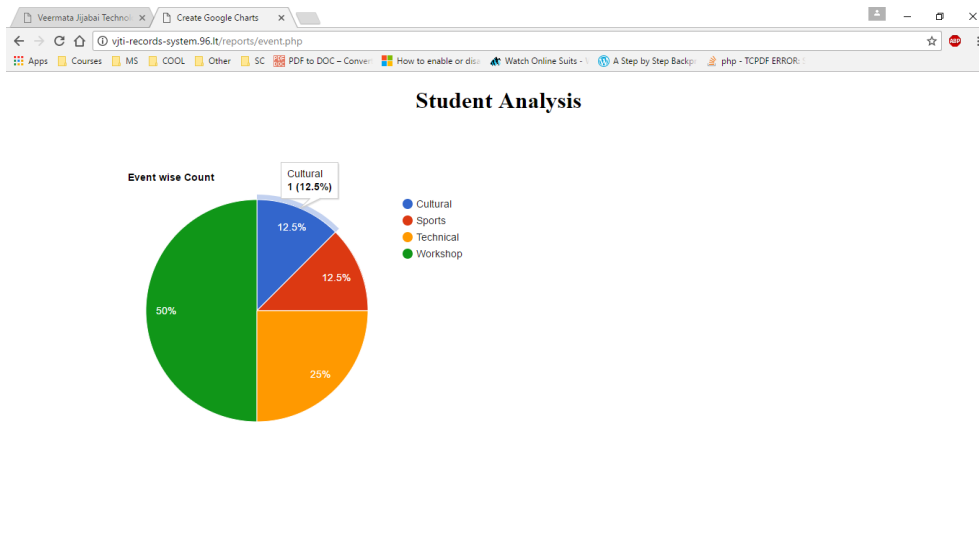


Figure 4.16: Sample Analytical Report 1

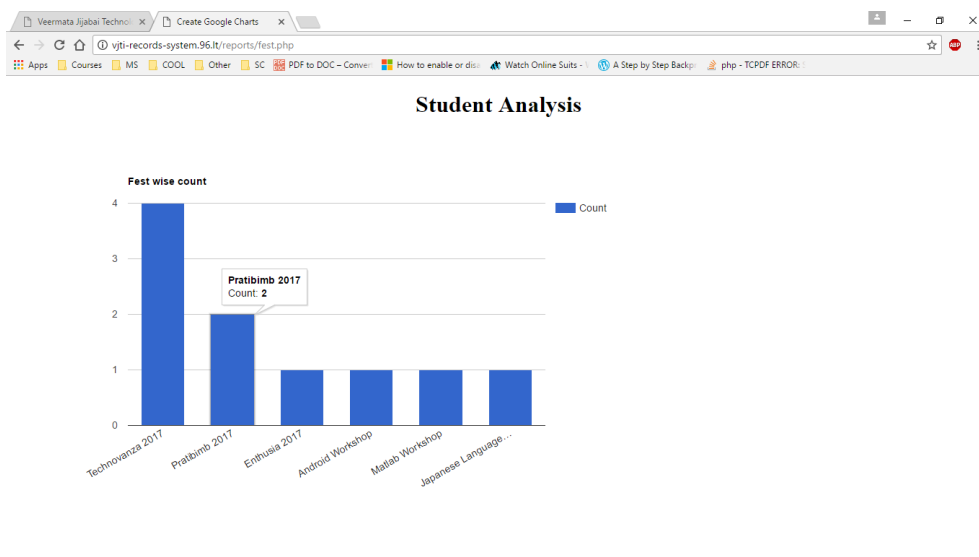


Figure 4.17: Sample Analytical Report 2

ports are useful for student analysis. These charts give an analysis of various parameters and provides segregation of data according to the created parameters.

Refer figure 4.18

The system also provides for searching of the reports, the required reports are available in the search space to get the report.

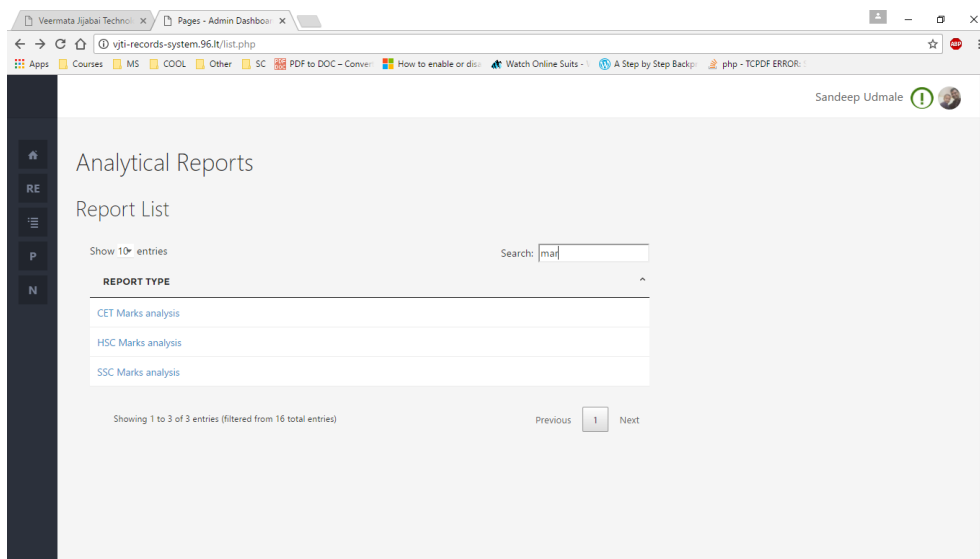


Figure 4.18: Search Analytical Report

4.2.4 PDF Reports

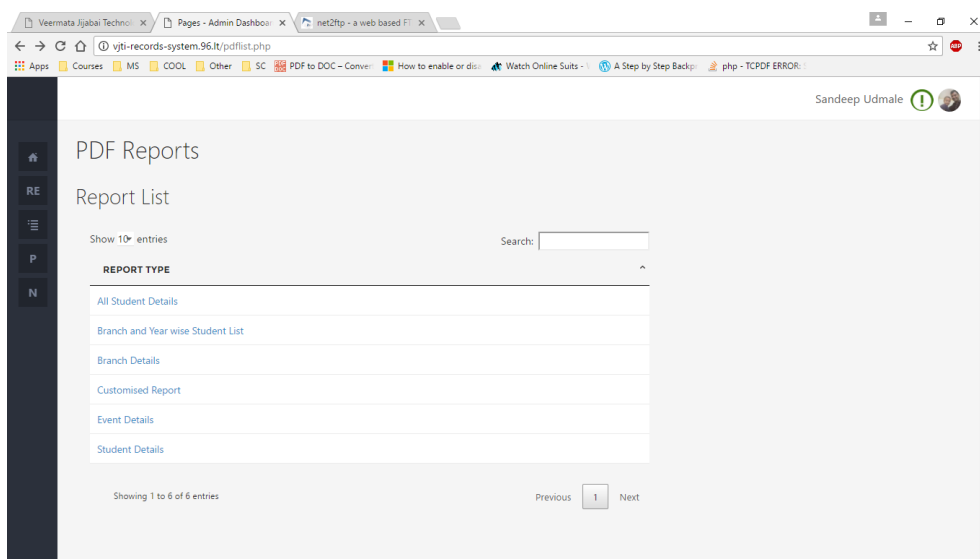


Figure 4.19: PDF Reports List Page

Refer figure 4.19.

This page provides the list of the PDF reports of the data. The PDF generator is used to generate PDFs of the required data by the user. They generate reports in such a way as it is required to be submitted to the NBA

committee. The reports contain data which is required for the accreditation process.

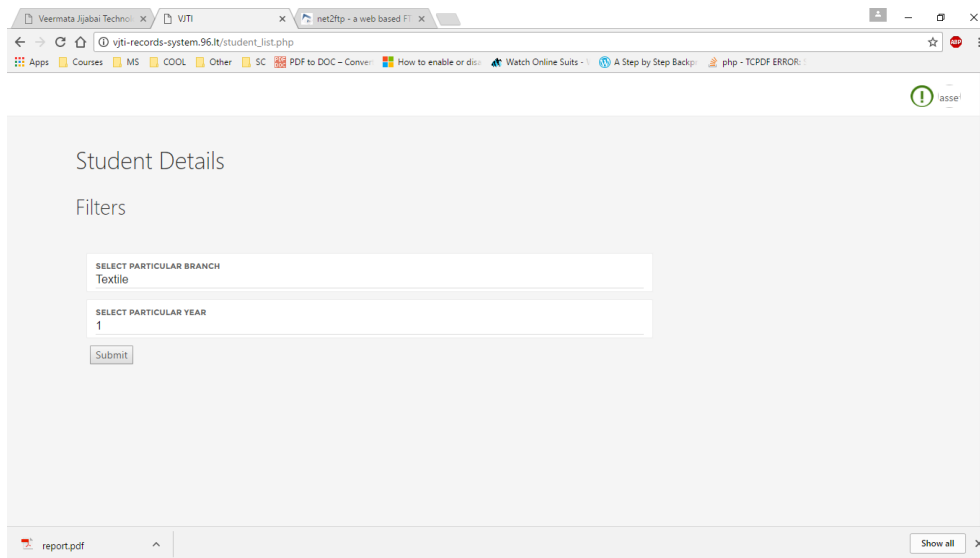


Figure 4.20: Sample PDF Reports Filter 1

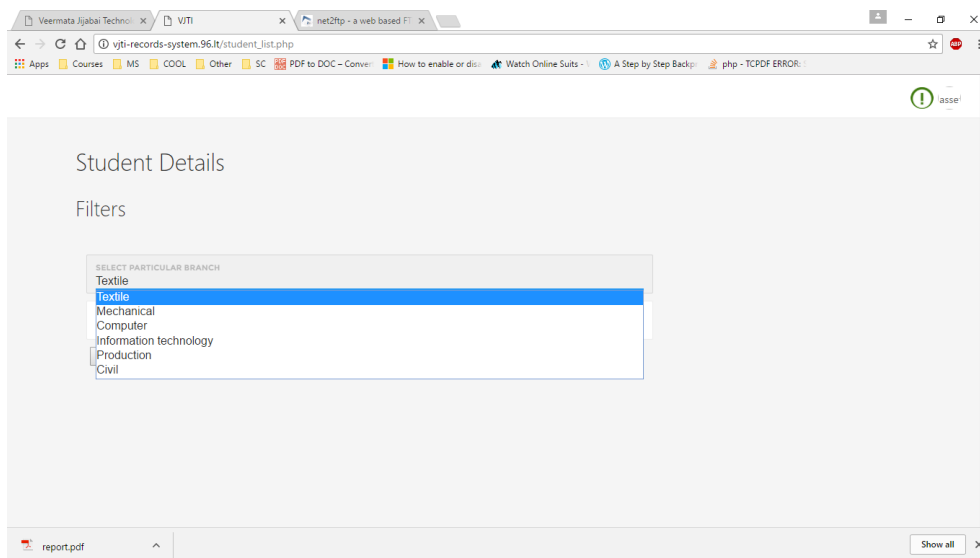


Figure 4.21: Sample PDF Reports Filter 2

Refer figure 4.20 , 4.21 and 4.22.

You can use the following filters to get the reports for whatever branch and year students the user needs.

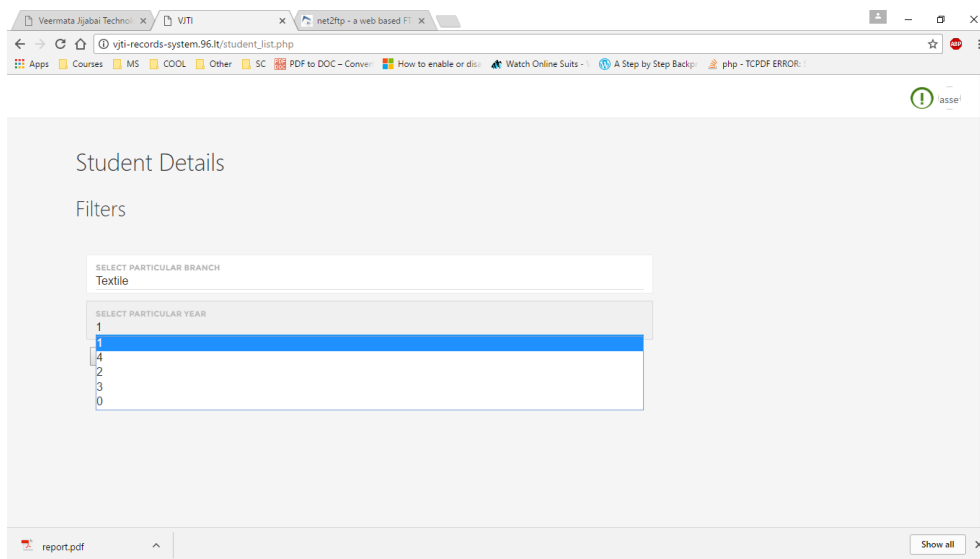


Figure 4.22: Sample PDF Reports Filter 3

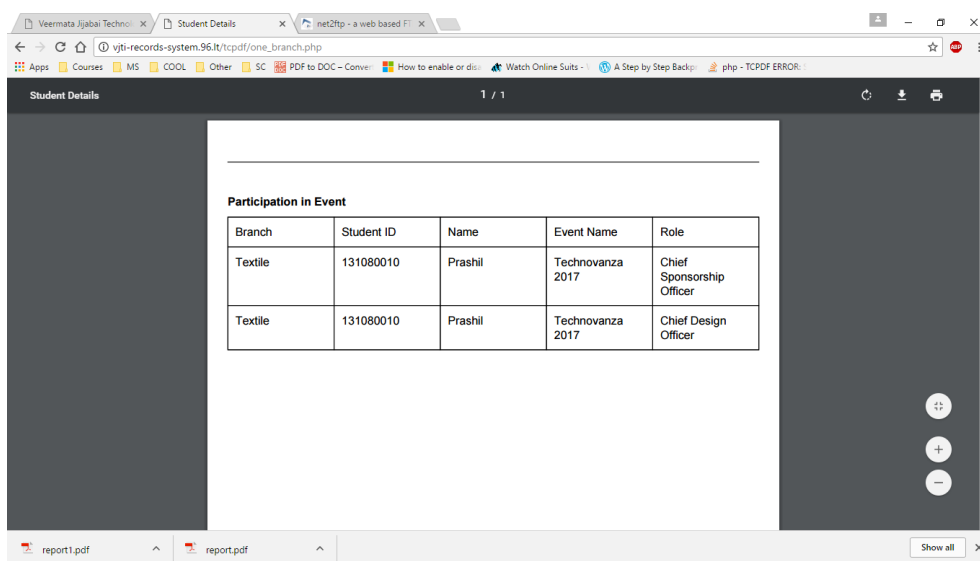


Figure 4.23: Sample PDF Reports 1

The report is generated as in figure 4.23.

Various reports can be generated by using filters in figure 4.24 and the reports generated will resemble fig 4.25 , 4.26, 4.27, 4.28

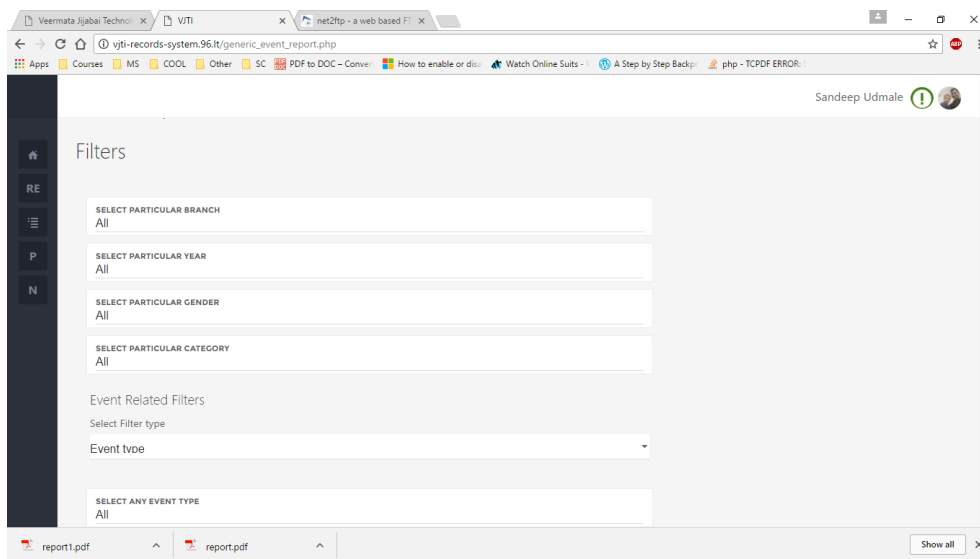


Figure 4.24: Sample PDF Reports Filter 4

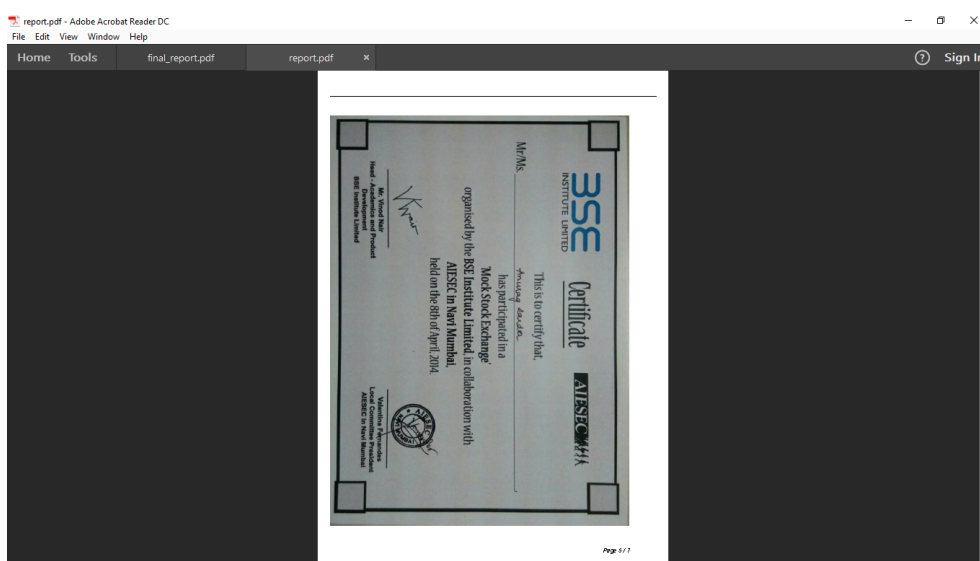


Figure 4.25: Sample PDF Reports 2

4.2.5 Event List

Refer fig 4.29

This page provides a list of all the events which have been created by the faculty. The user can view the events which has been created by the faculty.

report.pdf - Adobe Acrobat Reader DC
File Edit View Window Help

Home Tools final_report.pdf report.pdf x Sign In

ID	Name	Year	Branch	Event	Role	Role Type
131080021	Utkar	4	Computer	Jt Os	Organiser	
131080010	Prashil	1	Tecile	Participant	participant	
141080020	Etic	3	Production	Participant	participant	
141080020	Etic	3	Production	Pratigya Head	organiser	

Student Academics

ID	Name	Semester	GPI	CPI
131080010	Prashil	1	8.14	8.14
131080010	Prashil	2	8.25	8.2
131080015	Neel	1	8.38	8.38
131080015	Neel	2	8.5	8.44
131080015	Neel	3	8.22	8.33
131080021	Utkar	1	8.54	8.54
131080021	Utkar	2	8.94	8.74
131080020	Vishit	1	8.18	8.18
131080020	Vishit	2	8.38	8.28
131080016	Mikin	1	9.5	9.5
131080016	Mikin	2	9.3	9.4
131080016	Mikin	3	9.4	9.4

Page 1 / 1

Figure 4.26: Sample PDF Reports 3

report.pdf - Adobe Acrobat Reader DC
File Edit View Window Help

Home Tools final_report.pdf report.pdf x Sign In

Student External Activities

ID	Name	Year	Branch	Event	Role	Role Type
131080010	Prashil	1	Tecile	ACM KPC	Team member	participant
131080010	Prashil	1	Tecile	Techman za 2017	Chief Design Officer	organiser
131080010	Prashil	1	Tecile	Infoys Hackathon	Team member	participant
131080010	Prashil	1	Tecile	SSE Bull Run	Participant	participant
131080010	Prashil	1	Tecile	SAP Hackathon	Developer	organiser
131080010	Prashil	1	Tecile	Mumbai Marathon	Participant	participant
131080010	Prashil	1	Tecile	Roharwas COEP	Participant	participant
131080021	Utkar	4	Computer	ICPC	Participant	Participant

Student Internal Activities

ID	Name	Year	Branch	Role	Role Type
131080010	Prashil	1	Tecile	Chief Sponsorship Officer	organiser
131081036	Sakshi	3	Information Technology	Os	Organiser
131080010	Prashil	1	Tecile	Chief Design Officer	organiser
131080010	Prashil	1	Tecile	Volunteer	organiser
131080020	Vishit	1	Tecile	Chief Finance	Organiser

Page 1 / 1

Figure 4.27: Sample PDF Reports 4

4.2.6 View Student Information

Refer fig 4.30

The User can view any student information by feeding in the Student ID about whom it wants the information about. When the User inputs the Student ID it gets the information of the particular student. When the

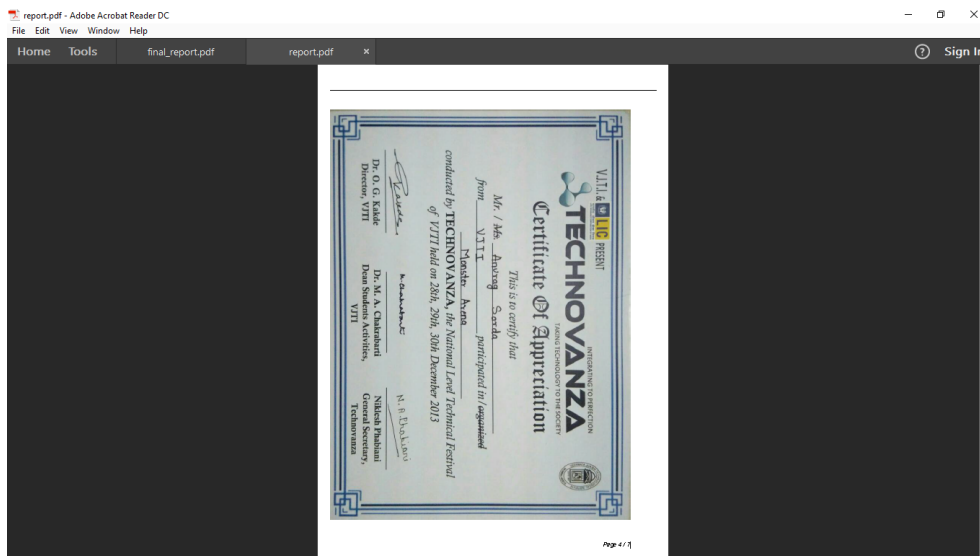


Figure 4.28: Sample PDF Reports 5

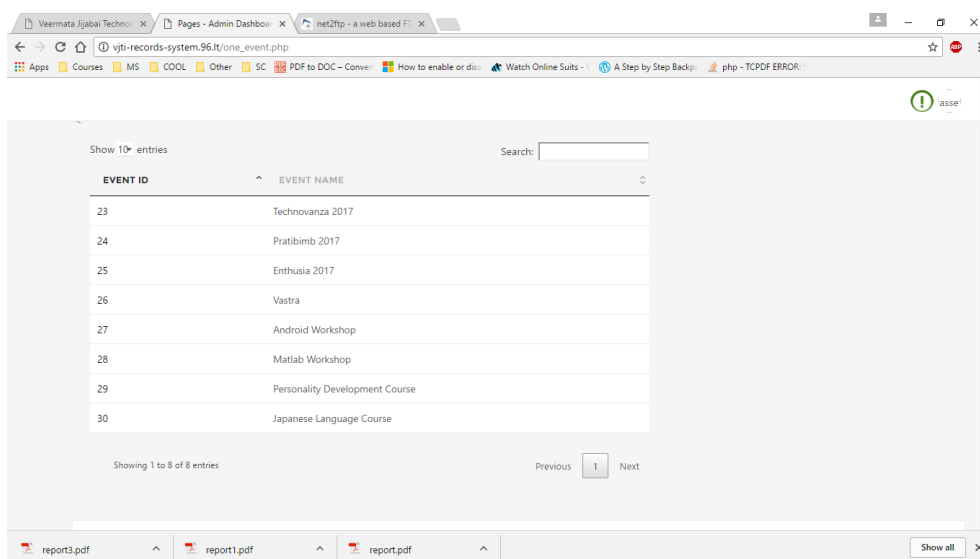


Figure 4.29: Event List Page

user inputs the Student ID it is redirected to that page which contains the student information like in 4.3, 4.4, 4.5 and 4.6. The user can also view all the activities in which the student has participated/organized.

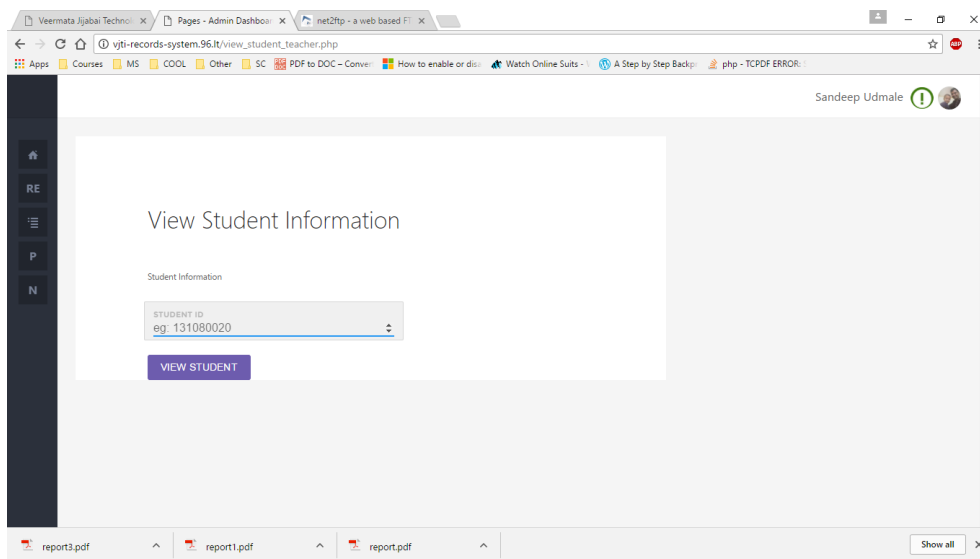


Figure 4.30: Event List Page

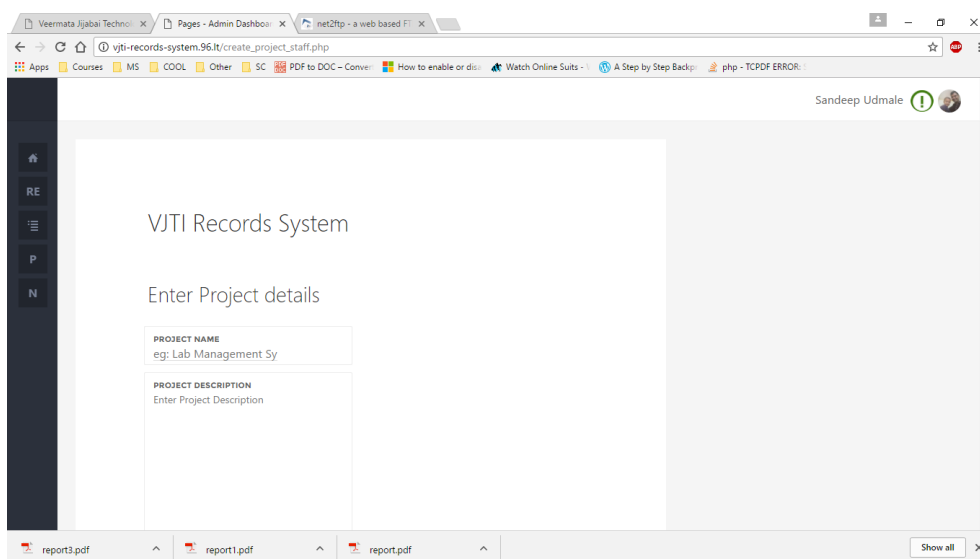


Figure 4.31: Create Project Page

4.2.7 Create Project

Refer fig 4.31

The user can create a project. The user has to enter the Project name, Project description to create the project. The student can join the project using the mentor ID of the faculty who has created the project.

4.2.8 Create Event

Figure 4.32: Create Event Page

Refer fig 4.32

The user can create a project. The user has to enter the Project name, Project description to create the project. The student can join the project using the mentor ID of the faculty who has created the project.

4.2.9 Notifications

Figure 4.33: Notifications Page

Refer fig 4.33

The notifications of approving the project or the event are sent to the user. These requests can be accessed in the notification sidebar button, these can be approved by the user. Notifications contains request for project approvals or event approval.

4.3 Future Scope

The University Records System provides important functionality, for the ease of maintaining the records online so that all the student information about their extracurricular activities, co-curricular activities, academics on a single platform so that the data can be evaluated and reports can be generated. This project has a scope to ease the difficulties faced by students and the faculty to maintain the records of the student online. The motivation for this project were the difficulties faced by the students and faculties during the process of accreditation. The base system can be used to integrate various in-house systems such as TPO system, TEQIP system, Lab Management System, Hostel Management System etc. This university records system can be used to store the records of these base systems, so that all the data of these systems can be clustered and evaluated to generate reports, of the student performance, and their data information about their placements. The base system can be collaborated, along with legacy systems to get all the processes online. The system has a very wide scope and can be extended to other systems, and makes the ease of having the entire system online. The data of the students about their extracurricular, co-curricular, academics can be used to have a clear view about the improvements in different regions according to the data. Like the external co-curricular activities can be used to keep different sessions for different departments. Extracurricular activities and their interests in the different domains can help to promote the activities so that it leads to overall development of the students. Also, when writing a letter of recommendation for a student going for graduate studies, the faculty can see the students profile and get a list of all the activities of the student. This integrates transparency in the institute and the faculty can promote students to participate in co-curricular and extra-curricular activities. Thus the system has a wide future scope because the base system can be integrated with legacy systems and there is a sufficient data for many of the institutes offline procedure to be digitized under a single system.

Chapter 5

Summary and Conclusions

To summarize, the project aims to ease the accreditation procedure by providing a centralized system for maintaining University Records online. The project will be implemented in HTML, CSS, PHP and SQL. The following list of functionalities have been implemented in the project:

- Maintaining academic record of all students of the university (current and past).
- Recording the extra-curricular activities of students who received certificates for participation.
- Tracking the activities of student organizations and their achievements.
- Maintaining co-curricular activities of students.
- Generating dynamic reports for various student activities.
- Generating analytical reports for student details and diversity
- Maintaing the records of projects undertaken under various various mentors

These functionalities have been combined with some legacy systems existing in college. Together they can be used to predict strengths and weaknesses of students and suggest changes in the curriculum accordingly. Also, the data maintained can be used for digitizing of Lab Management Systems, Training and Placement Office and TEQIP.

References

- [1] MySql Documentation, <https://dev.mysql.com/doc> <https://en.wikipedia.org/wiki/MySQL>
- [2] PHP Documentation, <http://php.net/docs.php> <https://en.wikipedia.org/wiki/PHP> <https://en.wikipedia.org/wiki/PHP#Implementations>
- [3] HTML References, <https://www.w3schools.com/php/> <https://www.w3schools.com/html/default.asp>
- [4] Java Script, <https://en.wikipedia.org/wiki/JavaScript>
- [5] Cascading Style Sheet, https://en.wikipedia.org/wiki/Cascading_Style_Sheets
- [6] JQuery Library for AJAX Documentation, <https://api.jquery.com/category/ajax>
- [7] Software Engineering: A Practitioner's Approach, Roger S. Pressman
- [8] Object-oriented Software Engineering: Practical Software Development using UML and Java, Robert Laganieri and Timothy Lethbridge
- [9] Database System Concepts, Henry F. Korth, Abraham Silberschatz and S. Sudarshan
- [10] Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe
- [11] Google Charts API, https://en.wikipedia.org/wiki/Google_Chart_API

Acknowledgements

This idea of building a University Records Management System for the institute came up due to the difficulties faced by the students and faculty during the Accreditation process in the college. We would like to thank Prof. S.S Udmale to give us the opportunity to take up this project of University Records Management System, and for guiding us for the project. We would also like to thank Dr. S.G.Bhirud, Prof. Dept. of CE and IT, for guiding us to design the functionalities to be integrated in the system.