Development of a Web Application for Managing Visitors Record of an Institute

Mr. Chirayu Mehta, Ms. Vaishnavi Junghare, Ms. Kajal Bagde,
Mrs.Sujata Wankhede

Department Of Information Technology
Bachelor Of Engineering
S.B Jain Institute of Technology,
Management & Research, Nagpur.

ABSTRACT

College Visitor Management system is a web-based technology that will revolutionize the way your college manages visitors. This web application provides a way to effectively control record & track college visitor traffic. Nowadays, visitor management consists of visitors scribbling their name in a paper book.Instead, visitor management system will assist you the professionalized way in which you welcome your visitors. This software is a complete visitor management service to improve the efficiency, productivity and security.The data of the new visitors visiting the campus will be stored in the database. With this system the visitor's information can be stored in the database, so that in the future it shall be used for the reference. It is a process of managing the information of the visitors visiting the College campus. For any organization maintaining the information of the visitor is a very major issue. This system will replace the manual maintenance system and will help the authorities to keep track of the visitors visiting the campus. This software is being build with help of ASP.NET. The data of the new visitors visiting the campus will be stored in the database. With this system the visitor's information can be stored in the database, so that in the future it shall be used for the reference. The data will be stored in the SQL SERVER as the backend. This software will help an organization to do the job effectively and give an all round development of the project.

Index Terms – Visitors management, less efforts, information stored for future reference,professionalized way.

CHAPTER 1

INTRODUCTION

1.1 PROJECT BACKGROUND

College Visitor Management system is a web-based technology that will revolutionize the way your college manages visitors. This web application provides a way to effectively control record & track college visitor traffic. Nowadays, visitor management consists of visitors scribbling their name in a paper book.Instead, visitor management system will assist you the professionalized way in which you welcome your visitors. This software is a complete visitor management service to improve the efficiency, productivity and security.The data of the new visitors visiting the campus will be stored in the database. With this system the visitor's information can be stored in the database, so that in the future it shall be used for the reference.

1.2 PROBLEM STATEMENT

"College Visitor Management System" is a process of managing the information of the visitors visiting the College campus. For any organization maintaining the information of the visitor is a very major issue. This system will replace the manual maintenance system and will help the authorities to keep track of the visitors visiting the campus.

1.3 PURPOSE OF STUDY

In present all visitor work done on the paper. The whole year visitor is stored in the registers. We can't generate reports as per our requirements. The purpose of developing visitor management system is to computerized the traditional way of visitor.Another purpose for developing this application is to generate the report automatically.

Objectives

- The in and out time of the visitors is stored in the database.
- The reason for which the visitor is going to visit the campus.
- Name, address and all other information regarding the visitor.
- When the visitor visits the campus for the first time he/she needs to show Id proof.
- Appointments of the Visitors with the Faculty, principle or etc .will also be stored in the database.

1.4 TECHNOLOGICAL BASE

This Project can be implemented by using various technologies like HTML, CSS, BOOTSTRAP for frontend and PHP for backend.

Front End:

1) HTML-

- HTML stands for Hyper Text Markup Language
- HTML is the standard markup language for creating Web pages
- HTML describes the structure of a Web page
- HTML consists of a series of elements
- HTML elements tell the browser how to display the content
- HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.

2) CSS-

- CSS stands for Cascading Style Sheets
- CSS describes how HTML elements are to be displayed on screen, paper, or in other media
- CSS saves a lot of work. It can control the layout of multiple web pages all at once
- External stylesheets are stored in CSS files

3) BOOTSTRAP-

It is an open-source and free CSS framework, which helps in directing a responsive device-friendly mobile-first front-end web page development tool. Bootstrap includes the CSS (Cascading Style Sheets), and an optional JavaScript supported design template (plug-ins) that deals with typography, implementation of buttons, forms, and various other components user interface. This framework helps in faster web development and supports developers in creating responsive web pages faster.

Back End:

1. PHP-

PHP(short for Hypertext PreProcessor) is the most widely used open source and general purpose server side scripting language used mainly in web development to create dynamic websites and applications.

- PHP is an acronym for "PHP: Hypertext Preprocessor"
- PHP is a widely-used, open source scripting language
- PHP scripts are executed on the server
- PHP is free to download and use

CHAPTER 2

LITERATURE SURVEY

2.1 HISTORICAL STUDY

[1]There are varieties similar systems available in the market when you search visitor managementsystem in any of search engine available. Those systems come with various features to offer to theircustomer with different price range.

[2]The Visitor Management system by applying the model of UTAUT was discussed by Norizan Anwar,Mohamad Noorman Masrek, Yanty Rahayu Rambli in year 2012. They proposed the system byadopting technology model (UTAUT) to determine the user acceptance of visitor application system.The main motto of this system was “Handling your visitor at your fingertips”.

[3]Another system for Visitor Pass was discussed in the paper by Prof. Abhay Gaidhani, Suraj Sahijwani,Parag Jain, Shantanu Jadhav, Ankush Jain in year 2015. This paper aims to develop a system for Gatepass using Raspberry Pi. The main aim was to save paper with the help of Internet Connectivity tosend Email for verification of user.

[4]Digital Visitor Information Management System (VIMS) Application and Design. This applicationenables capturing new visiting record by auto-clock in/out, and assignment of visitor pass. Visitorinformation are recorded in a centralized database server, which provides data management

andmanipulation through searching and report generating. E-VIMS able to record visitor informationduring visitor registration by using visitor's Malaysia Government Multipurpose Card (MyKad).

2.2 REAL-TIME SURVEY

- **YCCE College,Nagpur**

YCCE is the **prominent institution acknowledged for remarkable engineering education and research**. Since 38 years the institution has magnificently cherished and encouraged the forthcoming engineering professionals across the country making us one of the most opted engineering colleges in Maharashtra.

Problems:

1. Paper-based visitor management provides a false sense of security.
2. Anyone can enter in college without registration. It may cause problem in the future.
3. There is no database to check.

Solution:

1.The in and out time of the visitors is stored in the database.

2.Name, address and all other information regarding the visitor.

3.There will also be provision to retrieve the data from the database for the future references.

4.The reason for which the visitor is going to visit the campus.

5.Appointments of the Visitors with the Faculty will also be stored in the database.

- **SB Jain college, Nagpur**

S B Jain College of Engineering, is a college in Nagpur, India. It was established in 2008 by the Shantilal Badjate Charitable Trust.

Problems:

1. Paper-based visitor management provides a false sense of security.
2. Anyone can enter in college without registration. It may cause problem in the future.
3. There is no database to check.

Solution:

1.The in and out time of the visitors is stored in the database.

2.Name, address and all other information regarding the visitor.

3.There will also be provision to retrieve the data from the database for the future references.

4.The reason for which the visitor is going to visit the campus.

5.Appointments of the Visitors with the Faculty will also be stored in the database.

6.provide an alert if any unknown person enter in the campus

CHAPTER NO 3

METHODOLOGY / PROPOSED SOLUTION

3.1 PROPOSED WORK

In present all visitor work done on the paper. The whole year visitor is stored in the registers. We can't generate reports as per our requirements. The purpose of developing visitor management system is to computerized the traditional way of visitor.Another purpose for developing this application is to generate the report automatically. This web application provides a way to effectively control record & track college visitor traffic. Nowadays, visitor management consists of visitors scribbling their name in a paper book.Instead, visitor management system will assist you the professionalized way in which you welcome your visitors. This software is a complete visitor management service to improve the efficiency, productivity and security. With this system the visitor's information can be stored in the database, so that in the future it shall be used for the reference.

3.2SYSTEMARCHITECTURE

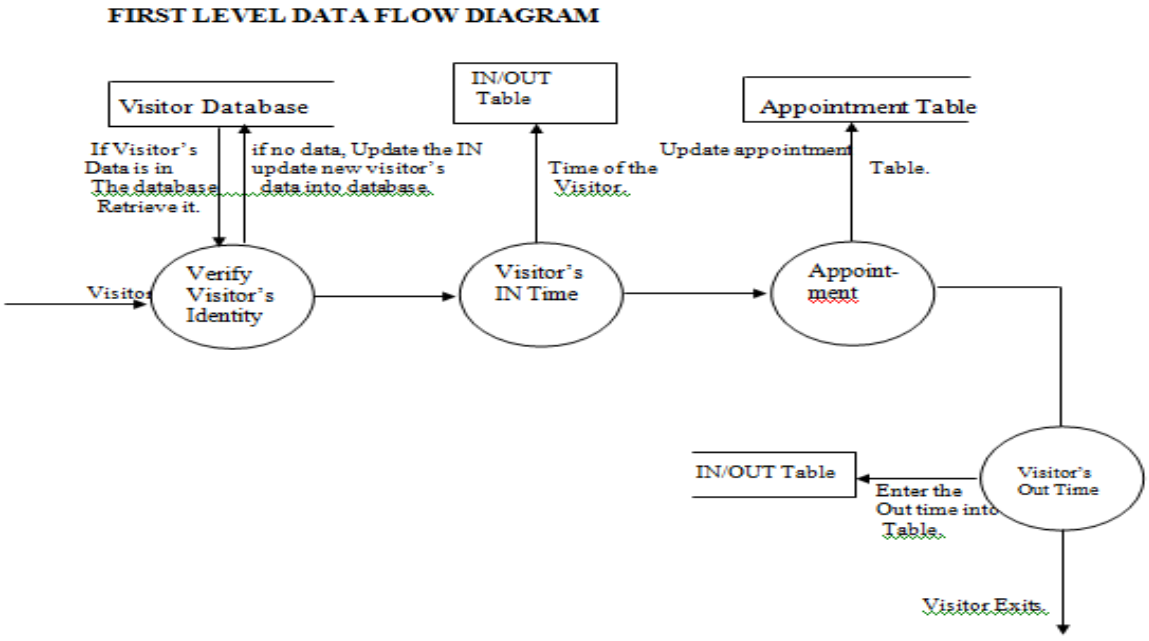


Fig 3.2 System Architecture

The purpose of developing visitor management system is to computerized the traditional way of visitor.Another purpose for developing this application is to generate the report automatically. The best thing of the software that

it is a process of managing the information of the visitors visiting the College campus. For any organization maintaining the information of the visitor is a very major issue. This system will replace the manual maintenance system and will help the authorities to keep track of the visitors visiting the campus.

Here the software provides the complete view of the software and its usage in the perfect manner. It also shows the working of all the actors.

3.3 FLOWCHART

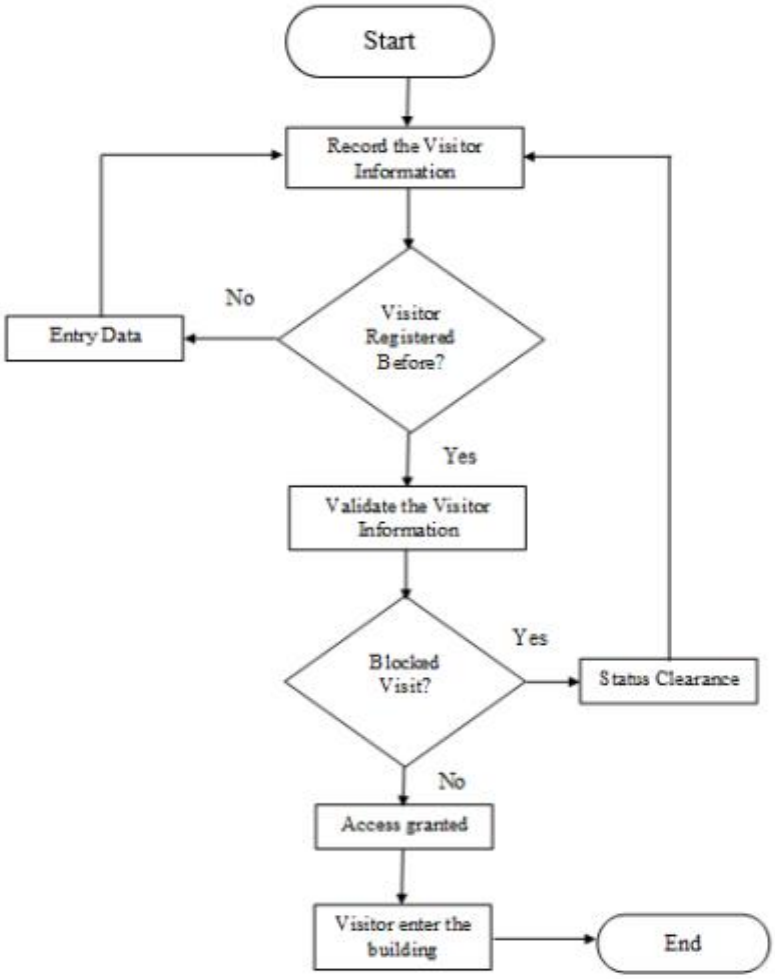


Fig 3.3 Flow Chart

Above we have shown the flow of our project. The guard is the main user which will use the software to manage visitors visiting in campus.

The major work of the admin is to generate report and manage it. Visitors management is use to generate report regularly.

CHAPTER 4

TOOLS/PLATFORM

4.1 SOFTWARE REQUIREMENT

Software:-

1. **OS** - Windows 7 or Higher , Xampp software
2. **Langauges** - Html , Css, Bootstrap , Php
3. **Testing tool** – Ms Excel

1. OPERATING SYSTEM: Any Operating System (preferably windows 7 or higher) which is having architecture of 32-bit or higher is supported. We have used Windows 10 64-bit.

2. Language:

HTML:

HTML stands for Hyper Text Markup Language. It is used to design web pages using markup language HTML is the combination of Hypertext and Markup language. Hypertext defines the link between the web pages. Markup language is used to define the text document within tag which defines the structure of web pages. This language is used to annotate (make notes for the computer) text so that a machine can understand it and manipulate text accordingly. Language uses tags to define what manipulation has to be done on the text.

Features:

- It is easy to learn and easy to use.
- It is platform independent.
- Images, video and audio can be added to a web page.
- Hypertext can be added to text.

Advantages :

HTML helps to build structure of a website and is a widely used Markup language.

- It is easy to learn.
- Every browser supports HTML Language.

- HTML is light weighted and fast to load.
- Storage of big files are allowed because of the application cache feature.
- Do not get to purchase any extra software because it's by default in every window.
- Loose syntax (although, being too flexible won't suit standards).
- HTML is simple to edit as being a plain text.
- It integrates easily with other languages such as JavaScript, CSS etc.
- HTML is that it is easy to code even for novice programmers.
- HTML also allows the utilization of templates, which makes designing a webpage easy.
- It is fast to download as the text is compressible.
- Very useful for beginners in the web designing field.

CSS:

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technolog of the World Wide Web, alongside HTML and JavaScript.

Features:

• You can control the- color of the text, the style of fonts and etc.
• Variations in display for different devices and screen sizes as well as a variety of other effects.

Advantages :

- CSS plays an important role, by using CSS you simply got to specify a repeated style for element once & use it multiple times as because CSS will automatically apply the required styles.
- The main advantage of CSS is that style is applied consistently across variety of sites. One instruction can control several areas which is advantageous.
- Web designers needs to use few lines of programming for every page improving site speed.
- Cascading sheet not only simplifies website development, but also simplifies the maintenance as a change of one line of code affects the whole web site and maintenance time.
- It is less complex therefore the effort are significantly reduced.
- It helps to form spontaneous and consistent changes.
- CSS changes are device friendly. With people employing a batch of various range of smart devices to access websites over the web, there's a requirement for responsive web design.

BOOTSTRAP:

Bootstrap is a free and open-source front end development framework for the creation of websites and web apps. The Bootstrap framework is built on HTML, CSS, and JavaScript (JS) to facilitate the development of responsive, mobile-first sites and apps. Responsive design makes it possible for a web page or app to detect the visitor's screen size and orientation and automatically adapt the display accordingly.

Features:
• Easy to Begin With
• LESS as Well as CSS Files
• Easily Customizable
• Responsive Utility Classes
• Some of the components come pre-styled in Bootstrap

Advantages :

- Easy to prevent repetitions among multiple projects
- Responsive design that can be used to adapt screen sizes and choose what shows and what doesn't on any given device
- Maintaining consistency among projects when using multiple developer teams
- Quick design of prototypes
- Cross-browser compatibility

PHP:

PHP started out as a small open source project that evolved as more and more people found out how useful it was. Rasmus Lerdorf unleashed the first version of PHP way back in 1994.

- PHP is a recursive acronym for "PHP: Hypertext Preprocessor".
- PHP is a server side scripting language that is embedded in HTML. It is used to manage dynamic content, databases, session tracking, even build entire e-commerce sites.
- It is integrated with a number of popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server.
- PHP is pleasingly zippy in its execution, especially when compiled as an Apache module on the Unix side. The MySQL server, once started, executes even very complex queries with huge result sets in record-setting time.

- PHP supports a large number of major protocols such as POP3, IMAP, and LDAP. PHP4 added support for Java and distributed object architectures (COM and CORBA), making n-tier development a possibility for the first time.
- PHP is forgiving: PHP language tries to be as forgiving as possible.
- PHP Syntax is C-Like.

Advantages :

- The most important advantage of PHP is that it's open-source and free from cost. It can be downloaded anywhere and is readily available to use for events or web applications.
- It is platform-independent. PHP-based applications can run on any OS like UNIX, Linux, Windows, etc.
- Applications can easily be loaded which are based on PHP and connected to the database. It's mainly used due to its faster rate of loading over slow internet speed than other programming language.
- It has less learning curve because it is simple and straightforward to use. Someone familiar with C programming can easily work on PHP.
- It is more stable for a few years with the assistance of providing continuous support to various versions.
- It helps in reusing an equivalent code and not got have to write lengthy code and sophisticated structure for events of web applications.
- It helps in managing code easily.
- It has powerful library support to use various function modules for data representation.
- PHP's built-in database connection modules help in connecting databases easily reducing trouble and time for the development of web applications and content-based sites.
- The popularity of PHP gave rise to various communities of developers, a fraction of which may be potential candidates for hire.
- Flexibility makes PHP ready to effectively combine with many other programming languages in order that the software package could use foremost effective technology for every particular feature.

4.2HARDWARE REQUIREMENT

PROCESSOR: Any processor supporting Xampp software

HARD DISK: Minimum 128GB

RAM: Minimum 4GB , 500GB HDD

CHAPTER 5

DESIGN & IMPLEMENTATION

5.1 SYSTEM DESIGN

5.1.1 USE-CASE DIAGRAM

Actors –

1. Admin
2.User

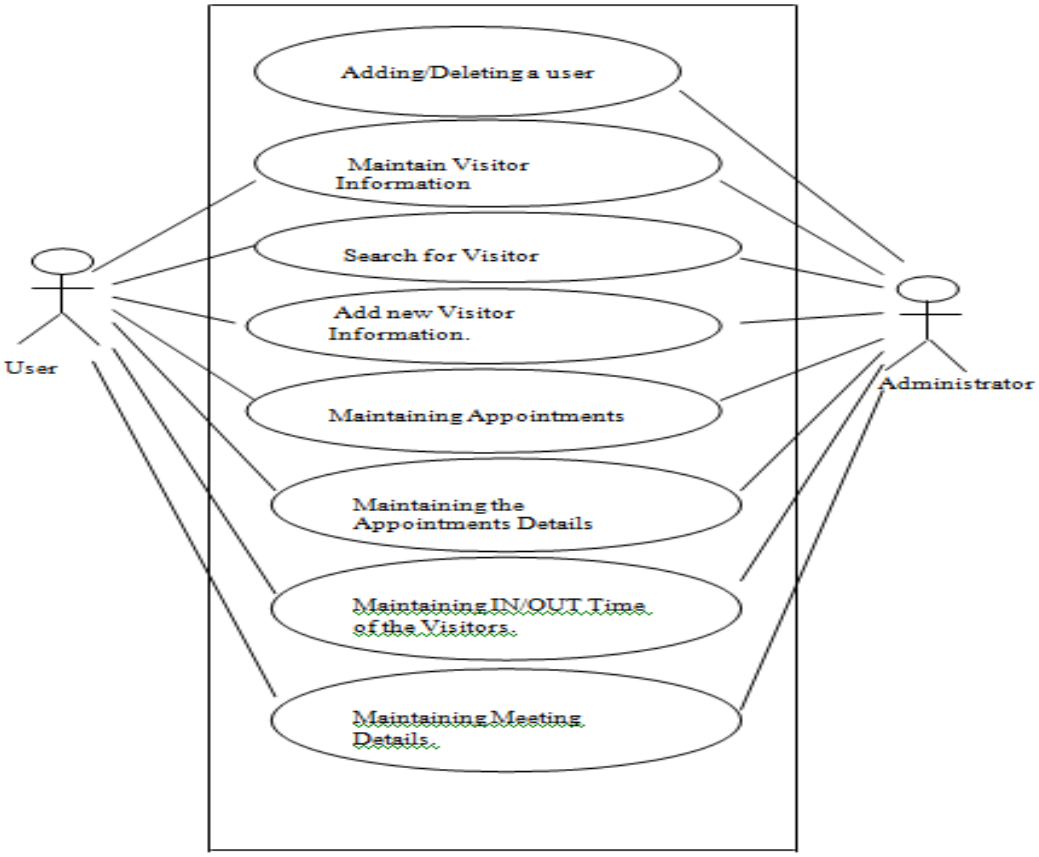


Fig. – 5.1.1: Use-Case Diagram

In Use-Case diagrsm, the tasks performed by the users are listed below,

1.Admin – Adding/Deleting s user, Maintain visitor information, Search for visitors , Add new visitors Information, Maintaining Appointments , Maintaining the Appointments Details, Maintaining IN/OUT Time of the visitors, Maintaining Meeting Details.

2.User - Maintain visitor information, Search for visitors , Add new visitors Information, Maintaining Appointments , Maintaining the Appointments Details, Maintaining IN/OUT Time of the visitors, Maintaining Meeting Details.

5.1.2. CLASS DIAGRAM

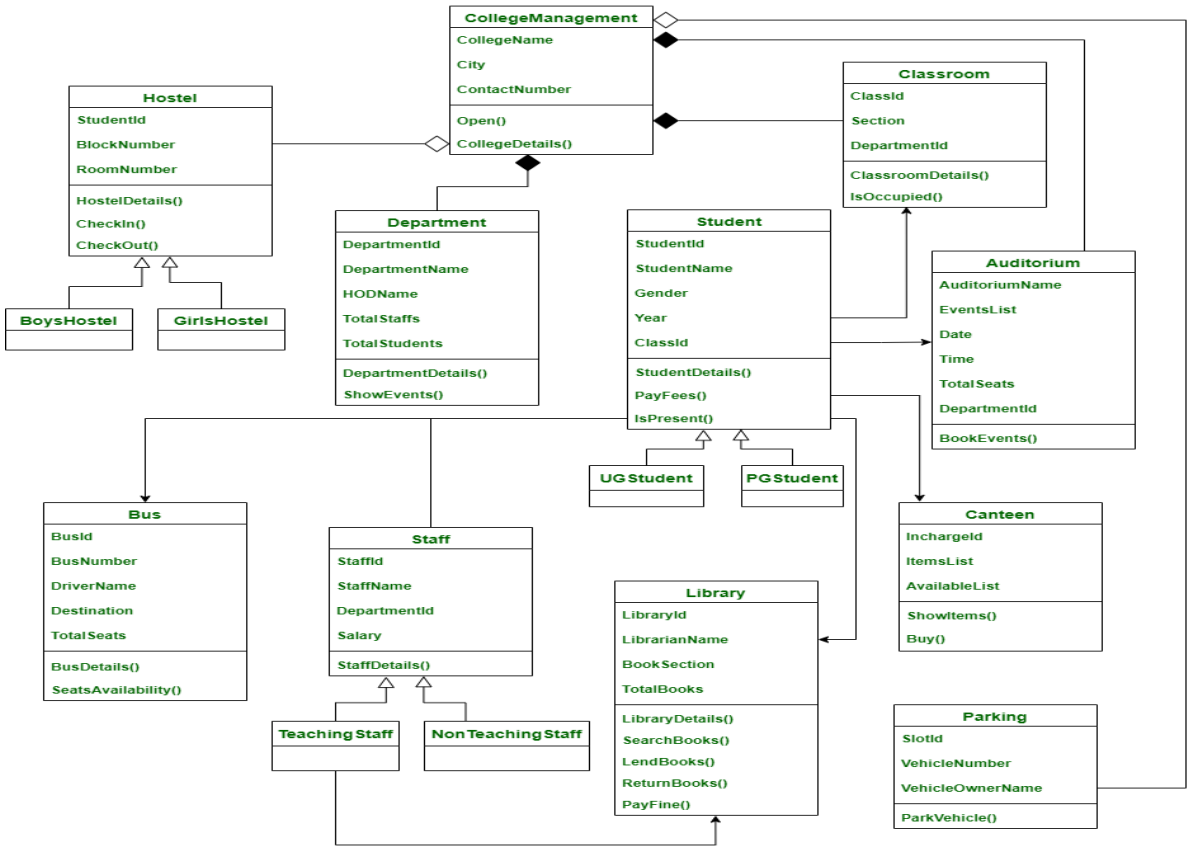


Fig. – 5.1.2: Class Diagram

- **CollegeManagement** – This class is the overall main class of the whole system.
- **Department** – This class contains the details of various departments in the college.
- **Student** – This class is for students and it is the base class for two child classes – UGStudent and PGStudent. Since UGStudent is a Student and PGStudent is a Student
- **UGStudent** – This class is the child class of Student and it contains the details of UGStudent.
- **PGStudent** – This class is the child class of Student and it contains the details of PGStudent.
- **Staff** – There are two types of staff in the college. So this class is the base class of two child classes – TeachingStaff and NonTeachingStaff
- **TeachingStaff** – This class is the child class of Staff. Since TeachingStaff is a Staff.
- **NonTeachingStaff** – This class is the child class of Staff. Since NonTeachingStaff is a Staff.
- **Classroom** – This class contains the details of each and every classroom in the whole college.
- **Canteen** – This class is for storing Canteen details inside the college
- **Library** – This class contains the details of a particular library in the college
- **Bus** – This class contains the details of a bus along with the bus driver details
- **Hostel** – Hostel can be of two types. So this class is the base class of two child classes – BoysHostel and GirlsHostel.
- **BoysHostel** – This class is the child class of the Hostel. Since BoysHostel is a Hostel.
- **GirlsHostel** – This class is the child class of the Hostel. Since GirlsHostel is a Hostel.
- **Parking** – This class contains the details of the parking area in a college. The parking area can be used by students, staff, visitors etc.
- **Auditorium** – Auditorium is a place where any events or guest lecture happens. This class contains the details of it.

5.1.3 SEQUENCE DIAGRAM

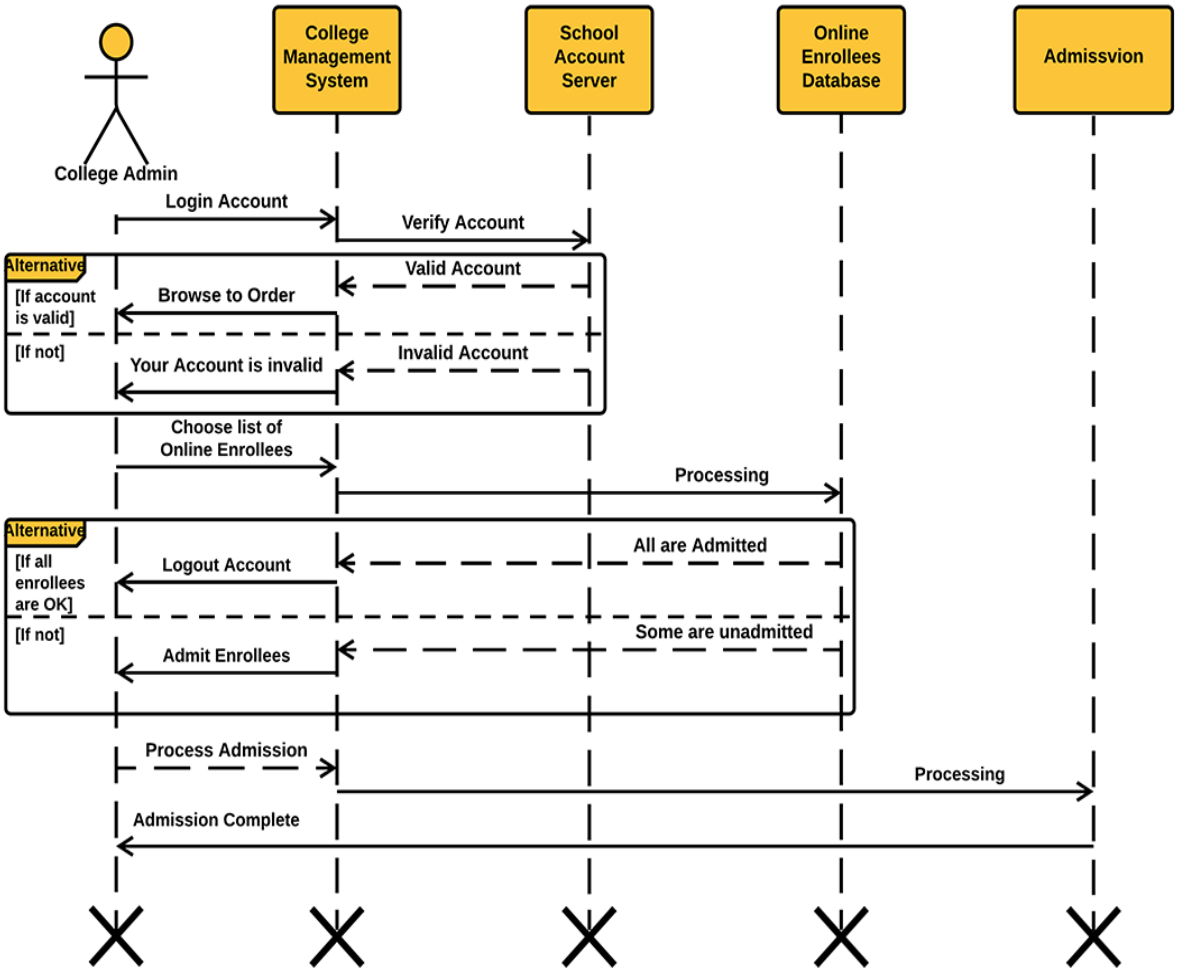


Fig. -5.1.3: Sequence Diagram

5.2 Implimentation Of System

5.2.1 Implemented Modules

Module:

1.Dashboard: In this section, admin can briefly view how many visitors visited in a particular period.

2.Visitors:In this section, admin adds new visitors by filling their information in add visitors sections and also view and manage visitors records.Admin also put visitors out time in the manage records section.

3.Search: In this bar, admin can search a particular person by their name and phone number.

4.Reports: In this section admin can generate visitors reports between two dates.

5.2.2 SAMPLE Code:

```
<(function ($) {
 // USE STRICT
 "use strict";

 try {
  //WidgetChart 1
  var ctx = document.getElementById("widgetChart1");
  if (ctx) {
   ctx.height = 130;
   var myChart = new Chart(ctx, {
    type: 'line',
    data: {
     labels: ['January', 'February', 'March', 'April', 'May', 'June', 'July'],
     type: 'line',
     datasets: [{
      data: [78, 81, 80, 45, 34, 12, 40],
      label: 'Dataset',
      backgroundColor: 'rgba(255,255,255,.1)',
      borderColor: 'rgba(255,255,255,.55)',
     },]
    },
    options: {
     maintainAspectRatio: true,
     legend: {
      display: false
     },
     layout: {
      padding: {
       left: 0,
```

```
      right: 0,
      top: 0,
      bottom: 0
    }
  },
  responsive: true,
  scales: {
    xAxes: [{
      gridLines: {
        color: 'transparent',
        zeroLineColor: 'transparent'
      },
      ticks: {
        fontSize: 2,
        fontColor: 'transparent'
      }
    }],
    yAxes: [{
      display: false,
      ticks: {
        display: false,
      }
    }]
  },
  title: {
    display: false,
  },
  elements: {
    line: {
      borderWidth: 0
    },
    point: {
      radius: 0,
      hitRadius: 10,
      hoverRadius: 4
    }
  }
 }
});
```

```
}
//WidgetChart 2
var ctx = document.getElementById("widgetChart2");
if (ctx) {
  ctx.height = 130;
  var myChart = new Chart(ctx, {
    type: 'line',
    data: {
      labels: ['January', 'February', 'March', 'April', 'May', 'June'],
      type: 'line',
      datasets: [{
        data: [1, 18, 9, 17, 34, 22],
        label: 'Dataset',
        backgroundColor: 'transparent',
        borderColor: 'rgba(255,255,255,.55)',
      },]
    },
    options: {
      maintainAspectRatio: false,
      legend: {
        display: false
      },
      responsive: true,
      tooltips: {
        mode: 'index',
        titleFontSize: 12,
        titleFontColor: '#000',
        bodyFontColor: '#000',
        backgroundColor: '#fff',
        titleFontFamily: 'Montserrat',
        bodyFontFamily: 'Montserrat',
        cornerRadius: 3,
        intersect: false,
      },
      scales: {
        xAxes: [{
          gridLines: {
            color: 'transparent',
            zeroLineColor: 'transparent'
```

```
        },
        ticks: {
          fontSize: 2,
          fontColor: 'transparent'
        }
      }],
      yAxes: [{
        display: false,
        ticks: {
          display: false,
        }
      }]
    },
    title: {
      display: false,
    },
    elements: {
      line: {
        tension: 0.00001,
        borderWidth: 1
      },
      point: {
        radius: 4,
        hitRadius: 10,
        hoverRadius: 4
      }
    }
  }
 });
}
//WidgetChart 3
var ctx = document.getElementById("widgetChart3");
if (ctx) {
 ctx.height = 130;
 var myChart = new Chart(ctx, {
  type: 'line',
  data: {
   labels: ['January', 'February', 'March', 'April', 'May', 'June'],
   type: 'line',
```

```
  datasets: [{
    data: [65, 59, 84, 84, 51, 55],
    label: 'Dataset',
    backgroundColor: 'transparent',
    borderColor: 'rgba(255,255,255,.55)',
  },]
},
options: {
  maintainAspectRatio: false,
  legend: {
    display: false
  },
  responsive: true,
  tooltips: {
    mode: 'index',
    titleFontSize: 12,
    titleFontColor: '#000',
    bodyFontColor: '#000',
    backgroundColor: '#fff',
    titleFontFamily: 'Montserrat',
    bodyFontFamily: 'Montserrat',
    cornerRadius: 3,
    intersect: false,
  },
  scales: {
    xAxes: [{
      gridLines: {
        color: 'transparent',
        zeroLineColor: 'transparent'
      },
      ticks: {
        fontSize: 2,
        fontColor: 'transparent'
      }
    }],
    yAxes: [{
      display: false,
      ticks: {
        display: false,
```

```
          }
         }]
        },
        title: {
         display: false,
        },
        elements: {
         line: {
          borderWidth: 1
         },
         point: {
          radius: 4,
          hitRadius: 10,
          hoverRadius: 4
         }
        }
       }
      });
    }

    //WidgetChart 4
    var ctx = document.getElementById("widgetChart4");
    if (ctx) {
     ctx.height = 115;
     var myChart = new Chart(ctx, {
      type: 'bar',
      data: {
       labels: ['January', 'February', 'March', 'April', 'May', 'June', 'July', 'August',
'September', 'October', 'November', 'December'],
        datasets: [
         {
          label: "My First dataset",
          data: [78, 81, 80, 65, 58, 75, 60, 75, 65, 60, 60, 75],
          borderColor: "transparent",
          borderWidth: "0",
          backgroundColor: "rgba(255,255,255,.3)"
         }
        ]
       },
```

```
      options: {
        maintainAspectRatio: true,
        legend: {
          display: false
        },
        scales: {
          xAxes: [{
            display: false,
            categoryPercentage: 1,
            barPercentage: 0.65
          }],
          yAxes: [{
            display: false
          }]
        }
      }
    });
  }

  // Recent Report
  const brandProduct = 'rgba(0,181,233,0.8)'
  const brandService = 'rgba(0,173,95,0.8)'

  var elements = 10
  var data1 = [52, 60, 55, 50, 65, 80, 57, 70, 105, 115]
  var data2 = [102, 70, 80, 100, 56, 53, 80, 75, 65, 90]
  var ctx = document.getElementById("recent-rep-chart");
  if (ctx) {
    ctx.height = 250;
    var myChart = new Chart(ctx, {
      type: 'line',
      data: {
        labels: ['January', 'February', 'March', 'April', 'May', 'June', 'July', 'August',
'September', ''],
        datasets: [
          {
            label: 'My First dataset',
            backgroundColor: brandService,
            borderColor: 'transparent',
```

```
      pointHoverBackgroundColor: '#fff',
      borderWidth: 0,
      data: data1
    },
    {
      label: 'My Second dataset',
      backgroundColor: brandProduct,
      borderColor: 'transparent',
      pointHoverBackgroundColor: '#fff',
      borderWidth: 0,
      data: data2

    }
  ]
},
options: {
  maintainAspectRatio: true,
  legend: {
    display: false
  },
  responsive: true,
  scales: {
    xAxes: [{
      gridLines: {
        drawOnChartArea: true,
        color: '#f2f2f2'
      },
      ticks: {
        fontFamily: "Poppins",
        fontSize: 12
      }
    }],
    yAxes: [{
      ticks: {
        beginAtZero: true,
        maxTicksLimit: 5,
        stepSize: 50,
        max: 150,
        fontFamily: "Poppins",
```

```
          fontSize: 12
        },
        gridLines: {
          display: true,
          color: '#f2f2f2'
        }
      }]
    },
    elements: {
      point: {
        radius: 0,
        hitRadius: 10,
        hoverRadius: 4,
        hoverBorderWidth: 3
      }
    }
  }
 });
}
// Percent Chart
var ctx = document.getElementById("percent-chart");
if (ctx) {
 ctx.height = 280;
 var myChart = new Chart(ctx, {
  type: 'doughnut',
  data: {
   datasets: [
    {
     label: "My First dataset",
     data: [60, 40],
     backgroundColor: [
      '#00b5e9',
      '#fa4251'
     ],
     hoverBackgroundColor: [
      '#00b5e9',
      '#fa4251'
     ],
     borderWidth: [
```

```
            0, 0
          ],
          hoverBorderColor: [
            'transparent',
            'transparent'
          ]
        }
      ],
      labels: [
        'Products',
        'Services'
      ]
    },
    options: {
      maintainAspectRatio: false,
      responsive: true,
      cutoutPercentage: 55,
      animation: {
        animateScale: true,
        animateRotate: true
      },
      legend: {
        display: false
      },
      tooltips: {
        titleFontFamily: "Poppins",
        xPadding: 15,
        yPadding: 10,
        caretPadding: 0,
        bodyFontSize: 16
      }
    }
  });
 }

} catch (error) {
 console.log(error);
}
try {
```

```
// Recent Report 2
const bd_brandProduct2 = 'rgba(0,181,233,0.9)'
const bd_brandService2 = 'rgba(0,173,95,0.9)'
const brandProduct2 = 'rgba(0,181,233,0.2)'
const brandService2 = 'rgba(0,173,95,0.2)'
var data3 = [52, 60, 55, 50, 65, 80, 57, 70, 105, 115]
var data4 = [102, 70, 80, 100, 56, 53, 80, 75, 65, 90]

var ctx = document.getElementById("recent-rep2-chart");
if (ctx) {
  ctx.height = 230;
  var myChart = new Chart(ctx, {
    type: 'line',
    data: {
      labels: ['January', 'February', 'March', 'April', 'May', 'June', 'July', 'August',
'September', ''],
      datasets: [
        {
          label: 'My First dataset',
          backgroundColor: brandService2,
          borderColor: bd_brandService2,
          pointHoverBackgroundColor: '#fff',
          borderWidth: 0,
          data: data3

        },
        {
          label: 'My Second dataset',
          backgroundColor: brandProduct2,
          borderColor: bd_brandProduct2,
          pointHoverBackgroundColor: '#fff',
          borderWidth: 0,
          data: data4

        }
      ]
    },
    options: {
      maintainAspectRatio: true,
```

```
legend: {
  display: false
},
responsive: true,
scales: {
  xAxes: [{
    gridLines: {
      drawOnChartArea: true,
      color: '#f2f2f2'
    },
    ticks: {
      fontFamily: "Poppins",
      fontSize: 12
    }
  }],
  yAxes: [{
    ticks: {
      beginAtZero: true,
      maxTicksLimit: 5,
      stepSize: 50,
      max: 150,
      fontFamily: "Poppins",
      fontSize: 12
    },
    gridLines: {
      display: true,
      color: '#f2f2f2'

    }
  }]
},
elements: {
  point: {
    radius: 0,
    hitRadius: 10,
    hoverRadius: 4,
    hoverBorderWidth: 3
  },
  line: {
```

```
          tension: 0
        }
      }

    }
  });
 }

} catch (error) {
 console.log(error);
}

try {

 // Recent Report 3
 const bd_brandProduct3 = 'rgba(0,181,233,0.9)';
 const bd_brandService3 = 'rgba(0,173,95,0.9)';
 const brandProduct3 = 'transparent';
 const brandService3 = 'transparent';

 var data5 = [52, 60, 55, 50, 65, 80, 57, 115];
 var data6 = [102, 70, 80, 100, 56, 53, 80, 90];

 var ctx = document.getElementById("recent-rep3-chart");
 if (ctx) {
  ctx.height = 230;
  var myChart = new Chart(ctx, {
   type: 'line',
   data: {
    labels: ['January', 'February', 'March', 'April', 'May', 'June', 'July', ''],
    datasets: [
     {
      label: 'My First dataset',
      backgroundColor: brandService3,
      borderColor: bd_brandService3,
      pointHoverBackgroundColor: '#fff',
      borderWidth: 0,
      data: data5,
      pointBackgroundColor: bd_brandService3
```

```
        },
        {
          label: 'My Second dataset',
          backgroundColor: brandProduct3,
          borderColor: bd_brandProduct3,
          pointHoverBackgroundColor: '#fff',
borderWidth: 0,
          data: data6,
          pointBackgroundColor: bd_brandProduct3

        }
       ]
      },
      options: {
       maintainAspectRatio: false,
       legend: {
        display: false
       },
       responsive: true,
       scales: {
        xAxes: [{
         gridLines: {
          drawOnChartArea: true,
          color: '#f2f2f2'
         },
         ticks: {
          fontFamily: "Poppins",
          fontSize: 12
         }
        }],
        yAxes: [{
         ticks: {
          beginAtZero: true,
          maxTicksLimit: 5,
          stepSize: 50,
          max: 150,
          fontFamily: "Poppins",
          fontSize: 12
         },
```

```
            gridLines: {
              display: false,
              color: '#f2f2f2'
            }
          }]
        },
        elements: {
          point: {
            radius: 3,
            hoverRadius: 4,
            hoverBorderWidth: 3,
            backgroundColor: '#333'
          }
        }

      }
    });
  }

 } catch (error) {
  console.log(error);
 }

 try {
  //WidgetChart 5
  var ctx = document.getElementById("widgetChart5");
  if (ctx) {
    ctx.height = 220;
    var myChart = new Chart(ctx, {
      type: 'bar',
      data: {
        labels: ['January', 'February', 'March', 'April', 'May', 'June', 'July', 'August',
'September', 'October', 'November', 'December'],
        datasets: [
          {
            label: "My First dataset",
            data: [78, 81, 80, 64, 65, 80, 70, 75, 67, 85, 66, 68],
            borderColor: "transparent",
            borderWidth: "0",
```

```
          backgroundColor: "#ccc",
        }
       ]
     },
     options: {
      maintainAspectRatio: true,
      legend: {
       display: false
      },
      scales: {
       xAxes: [{
        display: false,
        categoryPercentage: 1,
        barPercentage: 0.65
       }],
       yAxes: [{
        display: false
       }]
      }
     }
    });
  }

} catch (error) {
  console.log(error);
}

try {

  // Percent Chart 2
  var ctx = document.getElementById("percent-chart2");
  if (ctx) {
   ctx.height = 209;
   var myChart = new Chart(ctx, {
    type: 'doughnut',
    data: {
     datasets: [
      {
       label: "My First dataset",
```

```
      data: [60, 40],
      backgroundColor: [
        '#00b5e9',
        '#fa4251'
      ],
      hoverBackgroundColor: [
        '#00b5e9',
        '#fa4251'
      ],
      borderWidth: [
        0, 0
      ],
      hoverBorderColor: [
        'transparent',
        'transparent'
      ]
    }
  ],
  labels: [
    'Products',
    'Services'
  ]
},
options: {
  maintainAspectRatio: false,
  responsive: true,
  cutoutPercentage: 87,
  animation: {
    animateScale: true,
    animateRotate: true
  },
  legend: {
    display: false,
    position: 'bottom',
    labels: {
      fontSize: 14,
      fontFamily: "Poppins,sans-serif"
    }
```

```
      },
      tooltips: {
        titleFontFamily: "Poppins",
        xPadding: 15,
        yPadding: 10,
        caretPadding: 0,
        bodyFontSize: 16,
      }
    }
  });
 }

} catch (error) {
 console.log(error);
}

try {
 //Sales chart
 var ctx = document.getElementById("sales-chart");
 if (ctx) {
  ctx.height = 150;
  var myChart = new Chart(ctx, {
   type: 'line',
   data: {
    labels: ["2010", "2011", "2012", "2013", "2014", "2015", "2016"],
    type: 'line',
    defaultFontFamily: 'Poppins',
    datasets: [{
     label: "Foods",
     data: [0, 30, 10, 120, 50, 63, 10],
     backgroundColor: 'transparent',
     borderColor: 'rgba(220,53,69,0.75)',
     borderWidth: 3,
     pointStyle: 'circle',
     pointRadius: 5,
     pointBorderColor: 'transparent',
     pointBackgroundColor: 'rgba(220,53,69,0.75)',
    }, {
     label: "Electronics",
```

```
    data: [0, 50, 40, 80, 40, 79, 120],
    backgroundColor: 'transparent',
    borderColor: 'rgba(40,167,69,0.75)',
    borderWidth: 3,
    pointStyle: 'circle',
    pointRadius: 5,
    pointBorderColor: 'transparent',
    pointBackgroundColor: 'rgba(40,167,69,0.75)',
  }]
},
options: {
  responsive: true,
  tooltips: {
    mode: 'index',
    titleFontSize: 12,
    titleFontColor: '#000',
    bodyFontColor: '#000',
    backgroundColor: '#fff',
    titleFontFamily: 'Poppins',
    bodyFontFamily: 'Poppins',
    cornerRadius: 3,
    intersect: false,
  },
  legend: {
    display: false,
    labels: {
      usePointStyle: true,
      fontFamily: 'Poppins',
    },
  },
  scales: {
    xAxes: [{
      display: true,
      gridLines: {
        display: false,
        drawBorder: false
      },
      scaleLabel: {
        display: false,
```

```
          labelString: 'Month'
        },
        ticks: {
          fontFamily: "Poppins"
        }
      }],
      yAxes: [{
        display: true,
        gridLines: {
          display: false,
          drawBorder: false
        },
        scaleLabel: {
          display: true,
          labelString: 'Value',
          fontFamily: "Poppins"

        },
        ticks: {
          fontFamily: "Poppins"
        }
      }]
    },
    title: {
      display: false,
      text: 'Normal Legend'
    }
  }
 });
}
```

CHAPTER 6

TESTING , RESULTS & DISCUSSION

6.1 TESTING

6.1.1 TYPES OF TESTING

Manual Testing

Manual testing includes testing a software manually, i.e., without using any automated tool or any script. In this type, the tester takes over the role of an end-user and tests the software to identify any unexpected behavior or bug. There are different stages for manual testing such as unit testing, integration testing, system testing, and user acceptance testing .

Testers use test plans, test cases, or test scenarios to test a software to ensure the completeness of testing. Manual testing also includes exploratory testing, as testers explore the software to identify errors in it.

Following are the testing techniques that are performed manually during the test life cycle:

- Acceptance Testing
- White Box Testing
- Black Box Testing
- Unit Testing
- System Testing
- Integration Testing

Automation Testing

Automation testing, which is also known as Test Automation, is when the tester writes scripts and uses another software to test the product. This process involves automation of a manual process. Automation Testing is used to re-run 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 of view. It increases the test coverage, improves accuracy, and saves time and money in comparison to manual testing

Test Automation should be used by considering the following aspects of a software:

- Large and critical projects
- Projects that require testing the same areas frequently
- Requirements not changing frequently
- Accessing the application for load and performance with many virtual users
- Stable software with respect to manual testing
- Availability of time

6.1.2 LEVELS OF TESTING

There are four levels of testing: Unit, Integration, System and Acceptance

1. Unit Testing: A level of the software testing process where individual units/components of a software/system are tested. The purpose is to validate that each unit of the software performs as designed.

2. Integration Testing: A level of the software testing process where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units.

3. System Testing: A level of the software testing process where a complete, integrated system/software is tested. The purpose of this test is to evaluate the system's compliance with the specified requirements.

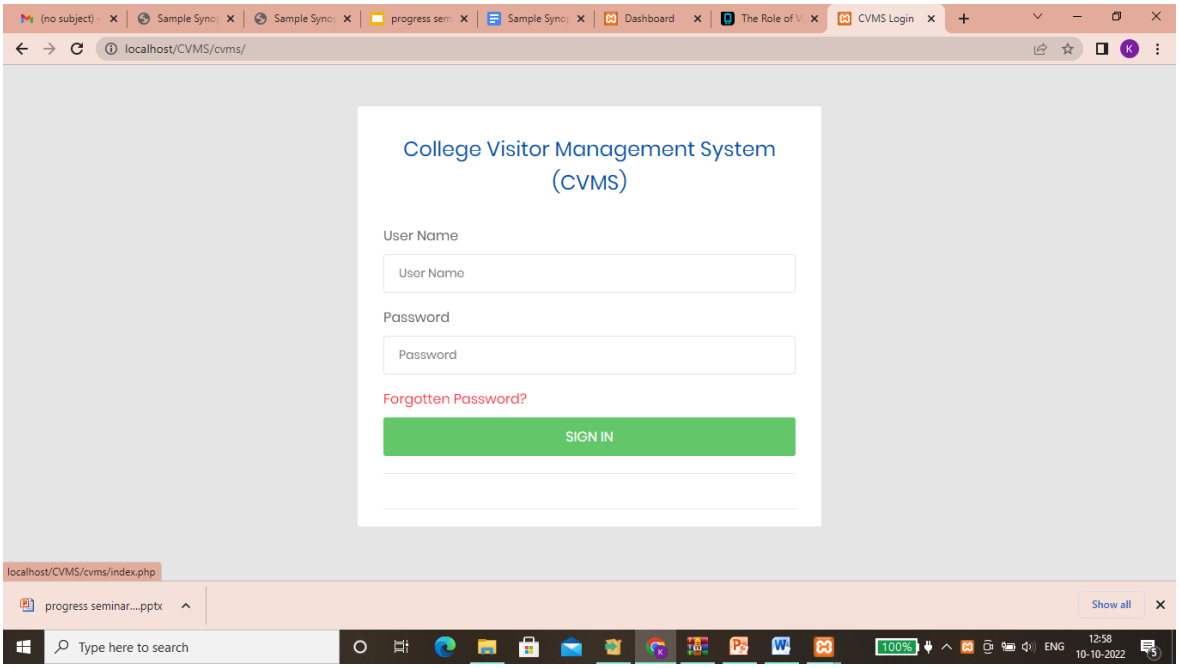
4. Acceptance Testing: A level of the software testing process where a system is tested for acceptability. The purpose of this test is to evaluate the system's compliance with the business requirements and assess whether it is acceptable for delivery.

6.1.3 TESTING REPORT

PROJECT:	college visitor management system	
MODULE:	Visitor	
PREPARED BY:	vaishnavi junghare kajal bagde chirayu mehta	

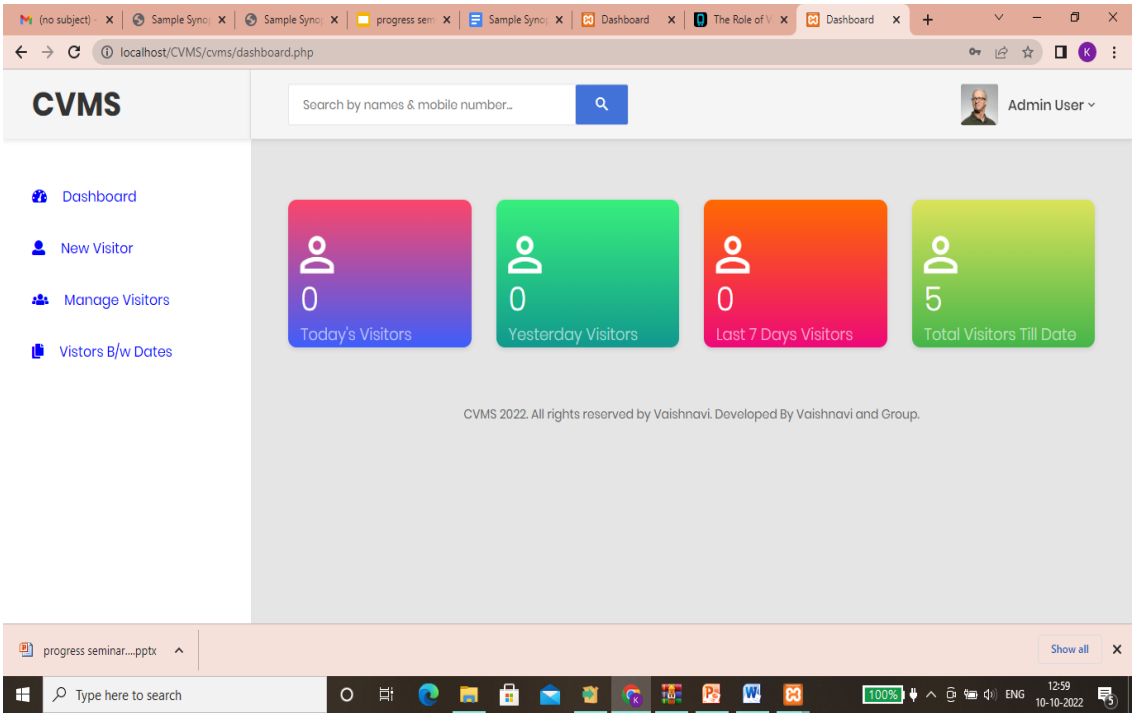
SR.NO.	TEST CASE ID	TEST OBJECTIVE	STEPS	DATA	PREREQUISITE	EXPECTED RESULT	ACTUAL RESULT	STATUS
1	TC_LOGIN_PAGE	to check the visitors detail.	1. open application 2.Enter username password for login	Username and Password		after successful login, it shoul go to the HOME section of an application.	It is working properly	Pass
2	TC_DASHBORD_02	to view the visitors details.	1.Go to drawable section in application	visitors should be login	there must be atleast one visit available of the visitors.	If there are any multiple logins,then it should be display the details of all those visitors.similarlly,it should also display the detailes of complete visitors.	Able to see complete logins.	Pass
3	TC_TO CHECK ALL THE DETAILS_03	to check whether the all information is filled or not.	1. Go to drawable section in application.	we can see the whole data of the visitors.		List of visitors with details done should be displayed.	Able to see detailed completed.	Pass
4	TC_PROFILE_04	To view the visitors details	1. Go to drawable section in application 2. Choose Profile tab		visitor should be logged in.	Details of visitors should be displayed on the screen.	Able to see the details.	Pass
5	TC_VISITORS _FORM_05	To fill the details in the form.	1. Go to drawable section in application 2. Choose entry pass			the visitors should fill the form properly.	It is working properly	Pass
6	TC_DATEWISE_ENTRY_06	To ckeck the datewise entry of the visitors.	1. Go to drawable section in application 2. Choose rntry b/t dates tab.	DATA	visitors all detail between specific dates.	visitors need to fill the form correctly where anyone can check the details betwenn the specific dates.	visitors name are present then its work properly.	Pass
7	TC_SEARCH_BAR_07	To Check the visitors detail..	1. Go to drawable section in application 2. Choose the search bar and search..	ALL DETAILS.	The search bar used to search a specific visitors details by their name.	the visitor should have the visit entry.	It is working properly	Pass
8	TC_VISITOR_08							
9	TC_VISITOR_09							
10	TC_VISITOR_10							
11	TC_VISITOR_11							
12	TC_VISITOR_12							
13	TC_VISITOR_13							
14	TC_VISITOR_14							
15	TC_VISITOR_15							
16	TC_VISITOR_16							
17	TC_VISITOR_17							

6.2 RESULTS AND DISCUSSIONS



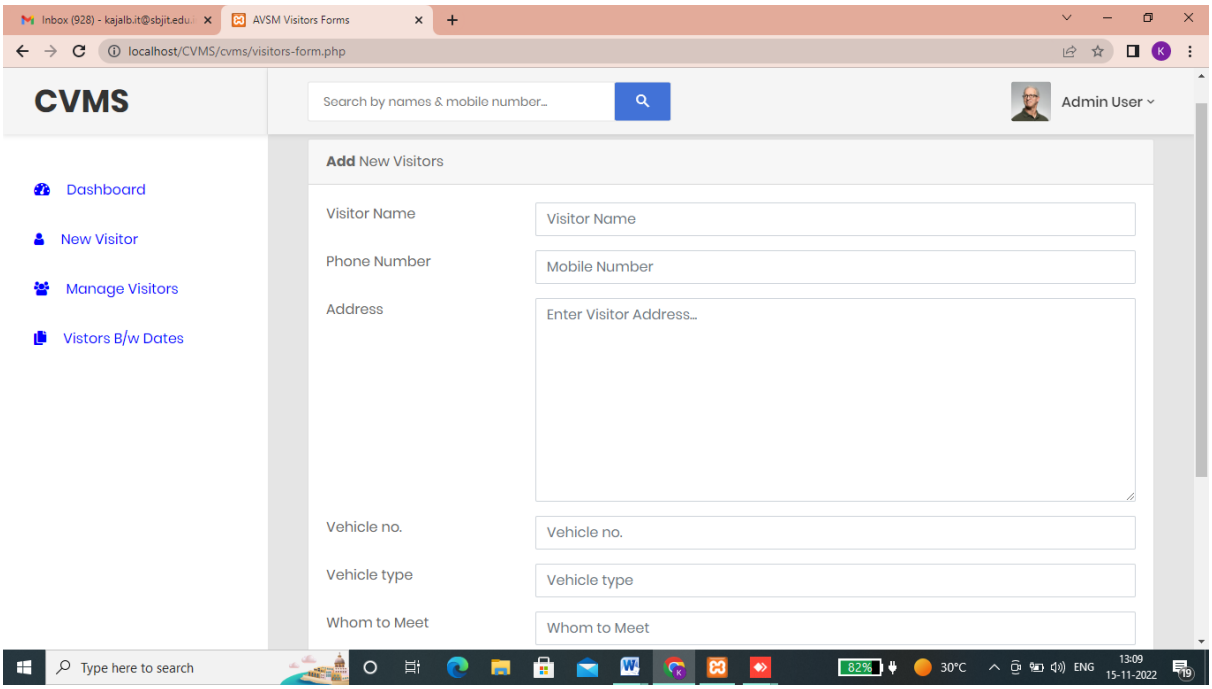
6.1.1 Login page

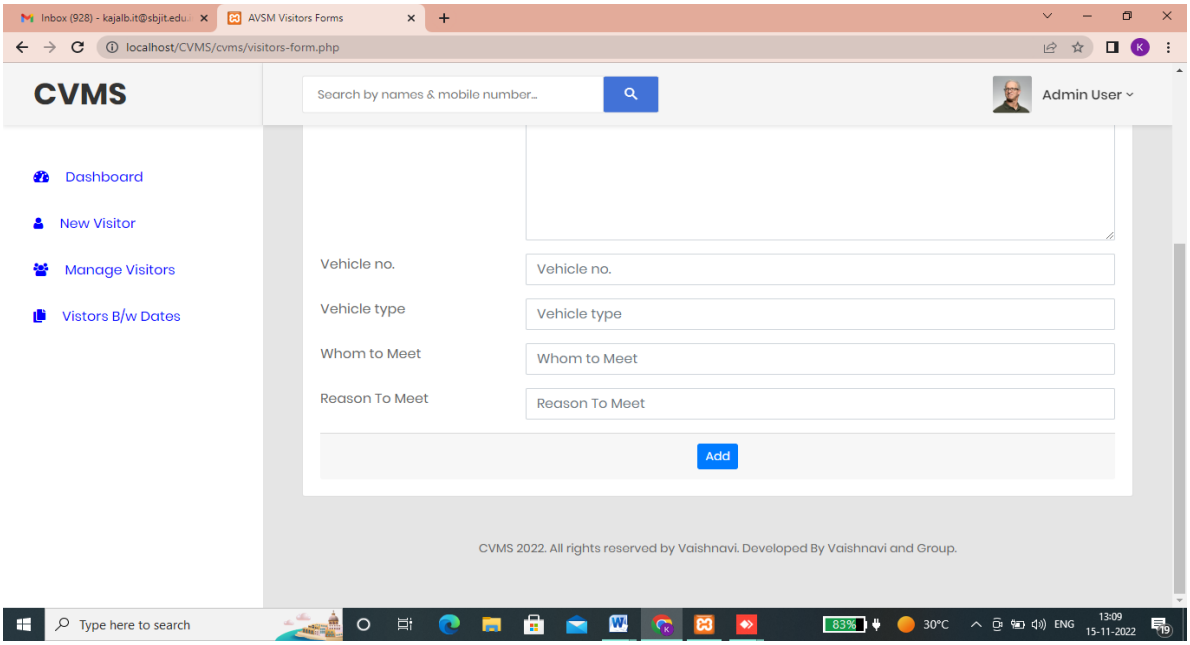
On starting the software, the first screen that appears is an shown in fig 6.1.1. Here After starting admin need to signin by adding username and password.



6.1.2 Dashboard

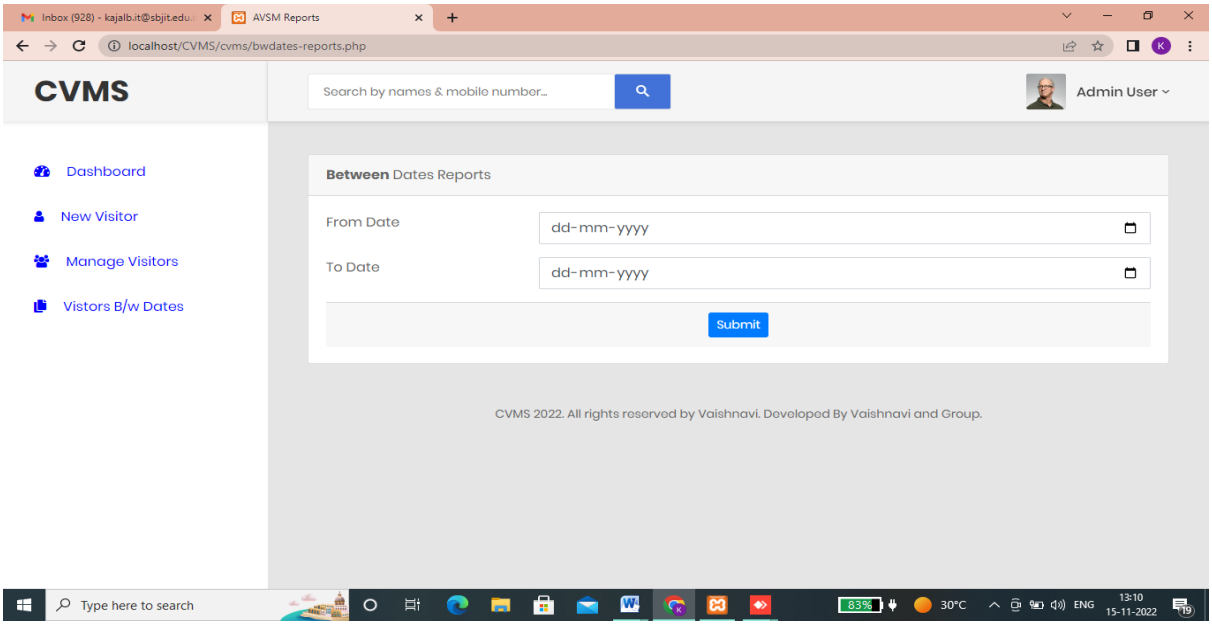
After entering username and password ,In this module admin can briefly view how many visitors visited in a particular period.





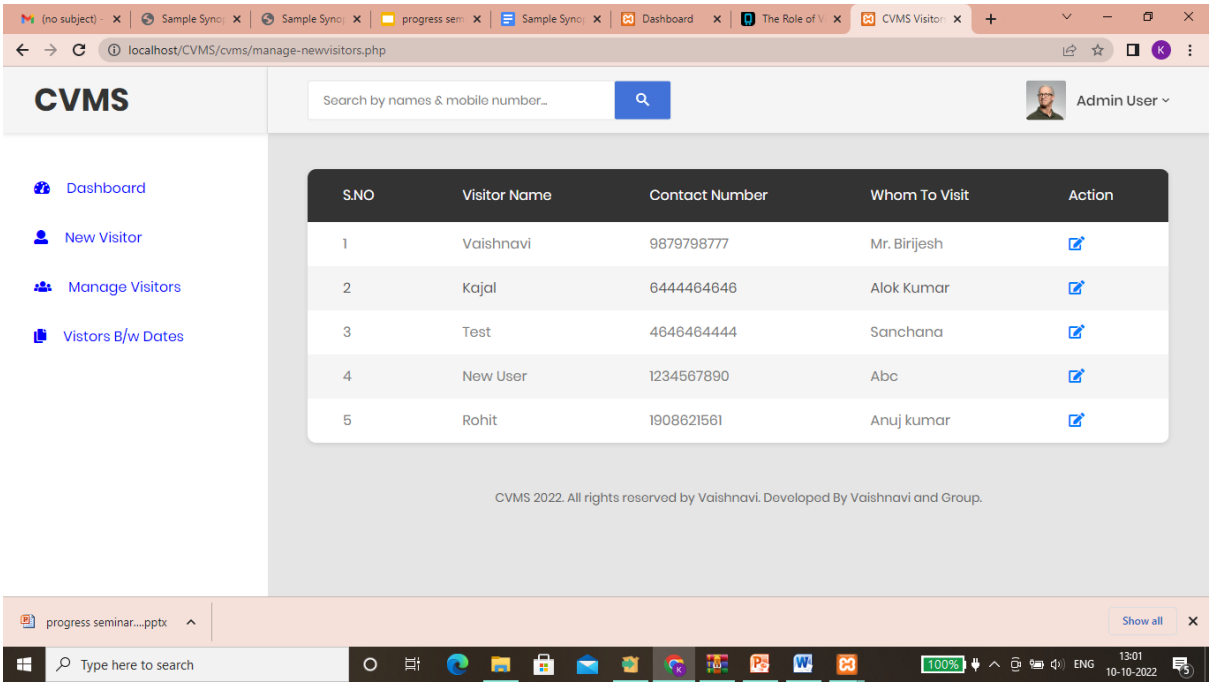
6.1.3 Add New Visitors

In this module, admin adds new visitors by filling their information in add new visitors section and also view and Manage visitors records. Admin also put visitors out and in time.



6.1.4 Visitors Search By Date

In this module admin can see how many visitors visits in the campus by adding dates or by adding name in search bar.



6.1.5 Report Generation

In this module, admin can generate visitors reports between two dates , admin can also update his profile, change password and phone number

CHAPTER 7

ADVANTAGES AND APPLICATIONS

7.1 ADVANTAGES

- Tracking of entry & exit of visitors- With SecurePass, entry time, exit time, & other details related to visitor visits are recorded in a cloud server.
- Contactless entry.
- Regular reports.
- Allows visitors to fill in their details quickly and easily.

7.2 APPLICATIONS

- There will also be provision to retrieve the data from the database for the future references.
- The reason for which the visitor is going to visit the campus.
- Appointments of the Visitors with the Faculty will also be stored in the database.

CHAPTER 8

CONCLUSION

8.1 CONCLUSION

We have **designed** a software and completed its partial development by applying **engineering knowledge** which provides an way to effectively control record & track college visitor traffic. Nowadays, visitor management consists of visitors scribbling their name in a paper book. thats why visitor management system solves the **societal problem** it will assist you the professionalized way in which you welcome your visitors. it **analyzes** the data of the new visitors visiting the campus will be stored in the database. With this system the visitor's information can be stored in the database, so that in the future it shall be used for the reference. We have used **modern tool** like visual studio to implement the project. During the development of the project we understood the importance of **individual and team work** while **project development and management.** While presenting our project in various seminars we have enhanced our **communication skills** and displayed **professional ethics** which will result in **lifelong learning**.

REFERENCES

PAPERS

1. N. Z. Haron, M. B. Mohter, S. A. M. J. Yunus, M. S. A. M. Isa and Rahim, "SAFe: A secure and fast auto filling form system," in ISIEA 2013 - 2013 IEEE Symposium on Industrial Electronics and Applications, 2013, 206–209.

2. H. Al Ghaithi, and U. Eaganathan, "A Brief Study and Implementation of Visitor Management System for Asia Pacific University, Malaysia," International Journal of Advance Research in Science and Engineering, vol. 5, no. 4, pp. 317–326, 2016.

3. B. S. Satari, N. A. A. Rahman, and Z. M. Z. Abidin, "Face recognition for security efficiency in managing and monitoring visitors of an organization." in Proceedings - 2014 International Symposium on Biometrics and Security Technologies, ISBAST 2014, 2015, pp 95–101.

4. N. K. M. Noor, J. Sulaiman and H. C. Khor, "Development of Visitor Management System Using Smart Card: UMP Case Study," in National Conference on Software Enginystems 2007 NaCSES'07, 2007.

WEBSITES

1. https://www.w3schools.com/php/default.asp [1]
2. https://www.sitepoint.com/php/ [2]
3. https://www.php.net/ [3]
4. https://www.mysql.com/ [4]
5. http://www.mysqltutorial.org [5]
6. https://www.apachefriends.org/download.html [6]