

#### A

# MINOR PROJECT REPORT ON

### "STUDENT FEES MANAGEMENT SYSTEM-SFMS"

Project Report Submitted in partial fulfilment for

the Award of degree of

# MASTER OF COMPUTER APPLICATIONS



# Submitted By:

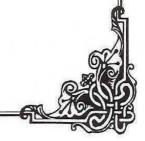
BATTULA VENKATESWARI	(6012263013)
<b>BOGIREDY VAISHANVI</b>	(6012263020)
CHELURI DINESH	(6012263026)
GAJJELA HAREESH	(6012263038)

*Under the esteemed guidance of* 

Prof. M. PADMAVATHAMMA Ph.D., M.phil., M.S., M.Ed., M.Sc

DEPARTMENT OF COMPUTER SCIENCE SRI VENKATESWARA UNIVERSITY TIRUPATI -517502 (2022-2024)





# DEPARTMENT OF COMPUTER SCIENCE S.V.U COLLEGE OF COMMERCE MANAGEMENT AND COMPUTERSCIENCE

# SRI VENKATESWARA UNIVERSITY TIRUPATI -517502

2022-2024

# **CERTIFICATE**

This is to Certify that the Project titled "STUDENT FEES MANAGEMENT SYSTEM-SRFS" Is a Bonafide work carried out by BATTULA **VENKATESWARI** (6012263013),**BOGIREDDY** VAISHNAVI (6012263020), CHELURI DINESH (6012263026), GAJJELA HAREESH (6012263038) of Sri Venkateswara University, Tirupati, for the award of MASTER OF COMPUTER APPLICATIONS is a record of bonafide minor project work carried out by the candidates under my supervision and Guidance. The minor project has reached the standard fulfilling the requirements of the regulation and guidance. The minor project has reached the standard fulfilling therequirements of the regulation for the award of the degree of SRI VENKATESWARA UNIVERSITY.

PROJECT GUIDE
Prof. M. PADMAVATHAMMA

Ph.D., M.phil., M.S., M.Ed., M.Sc

**Dept. of Computer Science** S V University, Tirupati

HEAD OF DEPARTMENT Prof. G. ANJAN BABU

Ph.D., M.Tech., M.Sc.

Dept. of Computer Science S V University, Tirupati

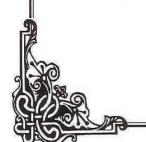
**Examiner Submitted for the practical Examination Held On: .....** 

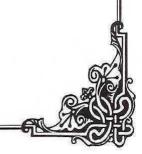
**Examiner** 

**External Examiner** 

1.

2.









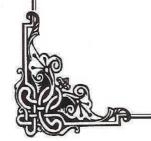
#### **DECLARATION**

We, here by declared that the these entitled "STUDENT RESULTS MANAGEMENT SYSTEM" under the esteemed guidance and supervision of Prof. M. PADMAVATHAMMA Ph.D., M.Phil., M.S., M.Ed., M.Sc Department of Computer Science, S V University, Tirupati, is submitted in partial fulfillment of the degree of Master of Computer Applications to Sri Venkateshwara University and this mini project is result of our own effort and has been submitted earlier for the award of Master of Computer Applications degree.

Place:

Date:

BATTULA VENKATESWARI (6012263013) BOGIREDDY VAISHNAVI (6012263020) CHELURI DINESH (6012263026) GAJJELA HAREESH (6012263038)





#### **ACKNOWLEDGEMENT**

We take this opportunity to express our sincere gratitude to all those who helped us in various capacities in undertaking this project and devising the report.

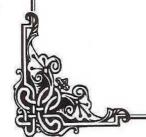
We are privileged to express our sense of gratitude to our respected Faculty **Prof. M. PADMAVATHAMMA**, whose unparalleled knowledge, moral fiber and judgment along with his know-how, was an immense support in completing the project.

We are also grateful to **Prof. G. ANJAN BABU**, the Head of Department, MCA Dept., for the brainwave and encouragement given.

I thank all my beloved Faculty Members, Department Of Computer Science, S V University, Tirupati.

I also extend my thanks to my Team Members for their co- operation during my course.

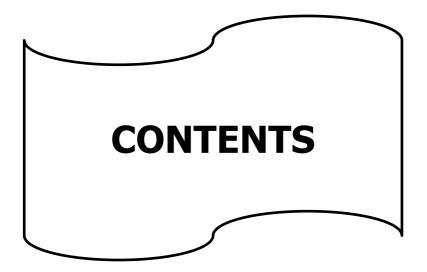
Finally I would like to thanks my friends and contemporaries for their co-operation and compliance.





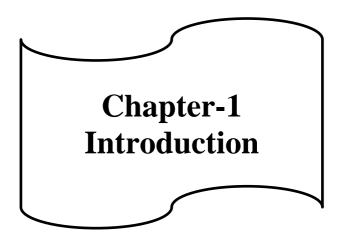
### **ABSTRACT**

The Student Fees Management System is a web-based application developed to maintain the results of the students. It contains three modules. Register/ login module, admin module and student module. Admin module is used to manage the student details. Student module is used to pay tuition fees or examination fees and can download the fee receipt. Registered admin can access the database of the students. Admin can only see and print student details and admin can track student details and Admin do not have access to edit student details. This is useful for the students.



# CONTENTS

<u>To</u>	Page No.		
1) Introdu	4-6		
1.1	Existing System	5	
	Proposed System	6	
2) System Analysis		7-11	
2.1	Feasibility Requirements	8	
2.2	Hardware Requirements	9	
2.3	Software Requirements	9-11	
	2.3.1 Programming Languages	10	
	2.3.2 Database Description	11	
3) System	12-24		
3.1	UML Diagrams	12-22	
3.2	ER Diagrams	23-24	
4)System	Implementation	25-59	
4.1	Sample Code	25-54	
4.2	Out Put Screens	55-59	
5) System	Testing	60-65	
6) Conclu	sion	66-67	
7) Bibliog	raphy	68-69	



#### 1.INTRODUCTION

Student Management System is software which is helpful for students as well as the school authorities. In the current system all the activities are done manually. It is very time consuming and costly. Our Student Management System deals with the various activities related to the students.

There are mainly 2 modules in this software

- ➤ Admin module
- > Student Module

In the Software we can register as a user and user has of two types, student And administrator. Admin can only see and print student details and admin can track student details and Admin do not have access to edit student details. A student can register as user and can add edit. All the student can see fees details.

#### 1.1 EXISTING SYSTEM:

In an existing system to check the results, the student should find the link over the website. For different courses and semesters they should find different links. There is no direct login for the admin or student. After entering the results it does not show the preview, due to that there is a chance of entering wrong results.

During analysis, data collected on the various files, decision points and transactions handled by the present system. The commonly used tools in the system are Data Flow Diagram, interviews, etc. Training, experience and common sense are required for collection of relevant information needed to develop the system. The success of the system depends largely on how clearly the problem is defined, thoroughly investigated and properly carried out through the choice of solution. A good analysis model should provide not only the mechanisms of problem understanding but also the frame work of the solution.

Thus it should be studied thoroughly by collecting data about the system. Then the proposed system should be analyzed thoroughly in accordance with the needs.

System analysis can be categorized into four parts.

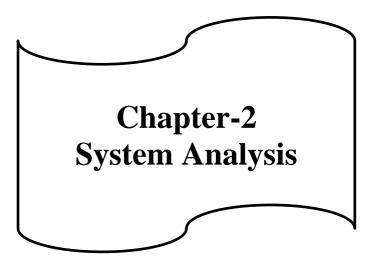
- ✓ System planning and initial investigation
- ✓ Information Gathering
- ✓ Applying analysis tools for structured analysis
- ✓ Feasibility study
- ✓ Cost/ Benefit analysis.

In the current system we need to keep a number of records related to the student and want to enter the details of the student and the marks manually. In this system only the teacher or the school authority views the mark of the student and they want to enter the details of the student. This is time consuming and has much cost.

#### 1.2 PROPOSED SYSTEM

In a proposed system, in the website the admin can easily login to admin module by using I'd and password. In this proposed system there is a featured called preview which helps Admin to see and print student details and admin can track student details and Admin do not have access to edit student details. The students can check any course in that semester results at one place by using result module no need to find different links Our proposed system has several advantages

- > User friendly interface
- > Fast access to database
- Less error
- More Storage Capacity
- > Search facility
- ➤ Look and Feel Environment



#### 2. SYSTEM ANALYSIS

#### 2.1 FEASIBILITY ANALYSIS

Whatever we think need not be feasible .It is wise to think about the feasibility of any problem we undertake. Feasibility is the study of impact, which happens in the organization by the development of a system. The impact can be either positive or negative. When the positives nominate the negatives, then the system is considered feasible. Here the feasibility study can be performed in two ways such as technical feasibility and Economical Feasibility.

#### **Technical Feasibility:**

We can strongly says that it is technically feasible, since there will not be much difficulty in getting required resources for the development and maintaining the system as well. All the resources needed for the development of the software as well as the maintenance of the same is available in the organization here we are utilizing the resources which are available already.

## **Economical Feasibility**

Development of this application is highly economically feasible. The organization needed not spend much money for the development of the system already available. The only thing is to be done is making an environment for the development with an effective supervision. If we are doing so, we can attain the maximum usability of the corresponding resources. Even after the development, the organization will not be in condition to invest more in the organization. Therefore, the system is economically feasible.

### 2.2 HARDWARE REQUIREMENTS

Processor : Pentium III 630MHz

RAM : 128 MB

Hard Disk : 20GB

Monitor : 15" Color monitor

Key Board : 104 Keys

# 2.3 SOFTWARE REQUIREMENTS

Operating System : Windows 11, Windows 10,

Frontend : HTML, JavaScript, CSS, React JS

Backend : Java, Spring Boot

Database : MySQL

# 2.3.1 Languages Used

# **HTML:** Hyper Text Mark Up Language

HTML is the standard markup language for documents designed to be displayed in a web browser. It defines the content and structure of web content. It is often assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.

# Java Script:

**JavaScript** often abbreviated as **JS**, is a programming language and core technology of the Web, alongside HTML and CSS. 99% of websites use JavaScript on the client side for webpage behavior.

#### **React JS:**

React JS is a free and open-source front-end library for building user interfaces. It is maintained by Facebook and a community of developers. React uses a declarative approach to building UIs, which makes it easier to write code that is maintainable and bug-free. React is used by many large companies, including Facebook, Netflix, and Airbnb. It is also popular among individual developers because it is easy to learn and use.

### Java:

Java is a programming language and computing platform first released by Sun Microsystems in 1995. It has evolved from humble beginnings to power a large share of today's digital world, by providing the reliable platform upon which many services and applications are built. New, innovative products and digital services designed for the future continue to rely on Java, as well.

### **Spring Boot:**

Spring Boot offers a fast way to build applications. It looks at your classpath and at the beans you have configured, makes reasonable assumptions about what you are missing, and adds those items. With Spring Boot, you can focus more on business features and less on infrastructure.

# **Functional Requirements**

The functional requirements of the system are to the implement the solution for finding the BMI value and Tips to follow for it's maintainance.

# 1. Input / Output:

The user have to give name, height, weight, age and gender of him with which finds the BMI value and his category.

# 2. Processing:

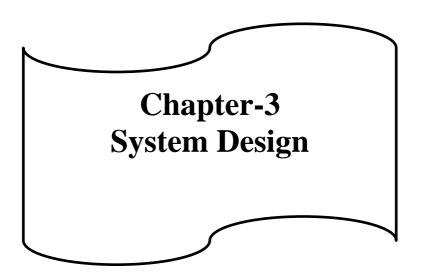
The information regarding user details are calculated through the formula which was inserted in the code.

# 3. Storage Requirements:

The information will be stored and retrieved from the database.

# 4. Control Requirements:

Alerts when any errors are there and when any of the field is not selected.



#### 3.SYSTEM DESIGN

#### Introduction

System design is a process through which requirements are translated into a representation of software. Initially the representation depicts a holistic view of software. Subsequent refinement leads to a design representation that is very close to source code. Design is a place where quality fostered in software development. Design provides us with representation of software that can be assessed for quality; this is the only way that can accurately translate the customer requirements into finished software product or system. System design serves as the foundation for all software engineering and software maintenance steps that follow.

We look the design process from three distinct perspectives:

- Conceptual Design
- Logical Design
- Physical Design

The higher view is the conceptual view, followed by the logical view and finally the physical view. In designing an application, we generally begin and end each phase in a sequentially order, although they may overlap one anotheralong the way.

#### **Conceptual Design:**

Conceptual Design is the process of acquiring and evaluating, documenting and then validating what the user envisions to be the business relation. It identifies the user and business requirements of the application andleads to a business solution as seen by the user.

All applications are built to solve business problems, and it is important to pay close attention to principle that the business need drives application development. At any point in the design process, the current state of the design should be directly traceable to a business problem and requirements.

To achieve this conceptual design is driven by developing usage scenarios. These scenarios are a direct representation of the user's view of the solution to a specific business problem. A conceptual view places the emphasize on solving a business problem and deriving a solution that corresponds to the needs and requirements of the users. It is based on deriving the behavior of the solution with a primary emphasizes on the user. Beginning with a emphasis on the activities of the business rather than aspects of software development, underscores the fact that systems exists to serve the business. A strong focus on the user in the beginning of the project will help in maintaining a proper perspective throughput the development lifecycle. The conceptual design results in the first description of what the system does to solve the business problemarticulated in the vision/scope document.

# **Logical Design**

Logical Design derives business objects and their related services directly from these usage scenarios. The logical view of the solution provides abasis for evaluating different physical options. It also formalizes the solution for the project team.

The idea of the application is that the system first emerges in logical design. Its boundaries and business objects and it contain the system definition. Logical design specifies the interfaces between the system and external entities, such as users and other systems. Within a system there may be a number of sub-systems, and these boundaries are also specified.

Logical System Design consists of the following steps:

- Input/Output Specifications
- File Specifications
- Processing Specifications

Logical design should be technologically independent as possible, inorder to separate system behavior issues from system implementation issues. Implementation constraints should only be considered only after the project team verifies that the essential behavior has been incorporated onto a logical design. This approach does not establish a technical direction until the system is well understood and documented.

# **Physical Design**

The purpose of Physical Design is to translate the logical design into a solution that can be implemented effectively, according to performance, administration and development process requirements. This physical view should correctly implement the desired system behavior while meeting the constraints imposed by the technology.

In Physical Design, the perspective shifts from an abstraction of system behavior to an implementation of the behavior. Whereas the logical designis largely technology independent, physical design is necessarily tied to chosen set of technologies, these being the hardware and software on which the applicationwill run.

The aim of physical design is to specify how to build portioned applications from software components. The interaction of these components through defined interfaces results in the desired behavior of the system as a whole. The rules for communicating between components are defined by interaction standards: what a component does and how it communicates are

major considerations in physical design.

Physical design consists of the following steps:

- 1. Design the physical media
  - Specify input/output media.
  - Design the database and specify backup procedures.
  - Design physical information flow through the system.
- 2. Plan the system implementation
  - Prepare a conversion schedule target date.
  - Determine training procedure, courses and timetable.
- 3. Device a test and implementation plan.
- 4. Specify any new Hardware/Software usage.
- 5. Update benefits, costs, conversion date and system constraints.

#### 3.1 UML Diagrams

#### Introduction

Design is the first step in the development phase for an engineered product or system. Design is the place where quality is fostered in software development. Design is the only way that we can accurately translate a user's requirements into a finished software product or system. Software design serves as the foundation for all software engineers and software maintenance steps that follow. Without design we risk building an unstable design -one that will fail when small changes are made, one that may be difficult to test, and one whose quantity cannot be accessed until late in the software engineering.

SFMS

Taking software requirements specification document of analysis phase as input to the design phase we have drawn.

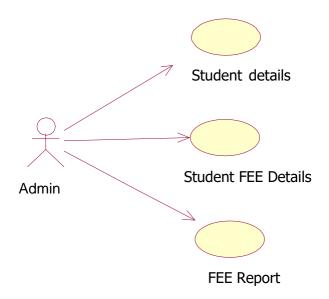
Unified Modelling Language: (UML) diagrams, UML depends on the visual modelling of the system. Visual modelling is the process of taking the information from the model and displaying it graphically using some sort of standards set of graphical elements.

UML Diagrams are drawn using the Pace Star UML Diagrammed Software. We seem too able to understand complexity better when it is displayed to us visually opposed to written textually. By producing visual models of a system, we can show how system works on several levels. We can model and the interactions between the users and the system.

### **Types of UML Diagrams**

Each UML diagram is designed to let developers and customers view a software system from a different perspective and in varying degrees of abstraction. UML diagrams commonly created in visual modeling tools include

Use Case Diagram displays the relationship among actors and use cases.



 $Fig \ 3.1.1 \ Use \ Diagram \ of \ administrator$ 

**Class Diagram** models class structure and contents using design elements such as classes, packages and objects. It also displays relationships such as containment, inheritance, associations and others.

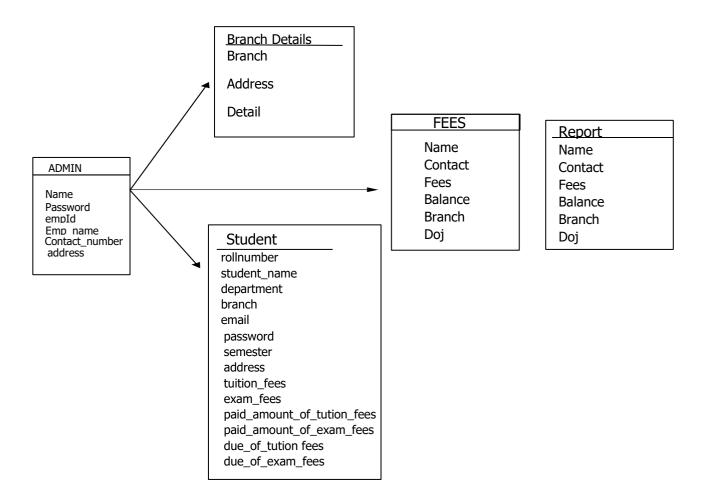


Fig 3.1.2 Class Diagram

# **Interaction Diagrams:**

**Sequence Diagram** displays the time sequence of the objects participating in the interaction. This consists of the vertical dimension (time) and horizontal dimension (different objects).

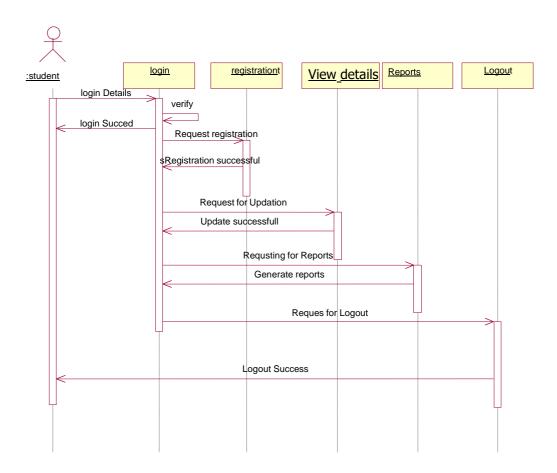


Fig 3.1.3 Sequence Diagram

**State Diagram** displays the sequence of states that an object of an interaction goes through during its life in response to received stimuli, together with its responses and actions.

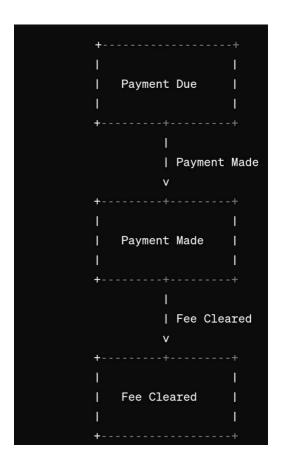


Fig 3.1.5 State Diagram

**Activity Diagram** displays a special state diagram where most of the states are action states and most of the transitions are triggered by completion of the actions in the source states. This diagram focuses on flows driven by internal processing.

### **Physical Diagrams:**

**Component Diagram** displays the high level packaged structure of the code itself. Dependencies among components are shown; include source code components, binary code components, and executable components. Some components exist at compile time, at link time, at run times well as at more than one time.

**Deployment Diagram** displays the configuration of run-time processing elements and the software components, processes, and objects that live on them. Software component instances represent run-time manifestations of code units.

#### 3.2 DATABASE DESIGN

The general theme behind a database is to handle information as an integrated whole. A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make information access easy quick and flexible for user. In database design several objectives are considered.

# **Control Redundancy:**

Redundant occupies space and therefore, is wasteful. If versions of the data are in different phases of updating the system often gives conflicting information. Aunique aspect of database design is storing only once, which controls redundancy and improves system performance.

#### **ER DIAGRAM:**

ER Diagram stands for entity relationship diagram, also known as ERD is a diagram that displays the relationship of entity sets stored in a database. In other words, ER diagrams help to explain the logical structure of database. ER diagrams are created based on the three basic concepts: Entities, attributes and relationship.

ER diagram contain different symbols that use rectangles to represent entities, ovals to define attributes and download shapes to represent relationships.

ER diagram looks very similar to the flow chart. However, ER Diagram includes many specialized symbols, and its meaning make this model unique. The purpose of ER diagram is to represent the entity framework infrastructure.

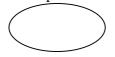
#### ER DIAGRAM SYMBOLS AND NOTATIONS:

Entity relationship diagram symbols and notations mainly contains three basic symbols which are rectangle, oval and diamond to represent relationship between elements, entities and attributes. There are some sub elements which are based on main elements in ERD diagram. ER diagram is a visual representation of the data that describes how data is related to each other using different ERD symbols and notations.

Following the main components and its symbols in ER Diagram:

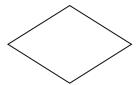
•	Rectangle:	This Entity	Relations	ship Diagram	symbol rep	resents entity
typ	pes			_		

• Ellipses: Symbol represent attributes.



• Diamonds: This symbol represents relationship types

SFMS



- Lines: It like attribute to entity types and entity types with other relationship types.
- Primary key: attributes are underlined.
- Double Ellipse: Represent Multi valued attribute.



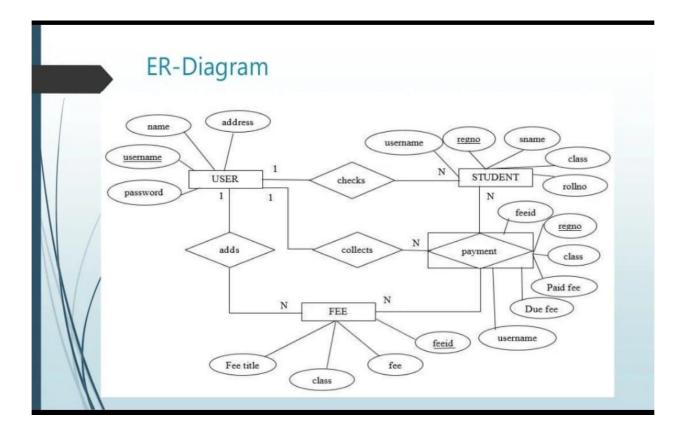
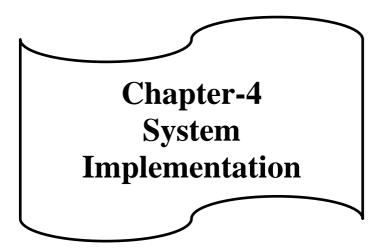


Fig. 3.1 ER Diagram of SVU FEE Payment System



#### **4 SYSTEM IMPLEMENTATION**

#### Introduction

Implementation is the stage in the project where the theoretical design is turned into a working system. The implementation phase constructs, installs and operates the new system. The most crucial stage in achieving a new successful system is that it will work efficiently and effectively.

There are several activities involved while implementing a new project. They are

- > End user training
- > End user Education
- > Training on the application software
- System Design
- Parallel Run and To New System
- > Post implementation Review

#### **End user Training:**

The successful implementation of the new system will purely upon the involvement of the officers working in that department. The officers will be imparted the necessary training on the new technology

#### **End User Education:**

The education of the end user start after the implementation and testing is over. When the system is found to be more difficult to understand and complex, more effort is put to educate the end used to make them aware of the system, giving them lectures about the new system and providing them necessary documents and materials about how the system can do this.

### **Training of application software:**

After providing the necessary basic training on the computer awareness, the users will have to be trained upon the new system such as the screen flows and screen design type of help on the screen, type of errors while entering the data, the corresponding validation check at each entry and the way to correct the data entered. It should then cover information needed by the specific user or group to use the system.

# **Post Implementation View:**

The department is planning a method to know the states of t he past implementation process. For that regular meeting will be arranged by the concerned officers about the implementation problem and success.

# **Project Module**

Our application deals with three modules

- Student module
- Admin module
- Data management Module.

#### **Student Module:**

- ➤ In the Software we can register as a student with his name and email id.
- Student can see his/her details to ensure that is the profile of same.
- > Student can pay his/her fees directly through this interface.
- > Students can pay their tuition fees, exam fees.
- > Students can update their information any time anywhere.
- After paying their tuition fees or exam fees they can download customized fees receipt with unique id of their payment.

SFMS

## **Admin Module:**

- ➤ In the Software we can register as a admin with his name and email id.
- > Admin can not be able edit student details
- Admin can see all details of student who has registered earlier.
- Admin can see the payment status, when the student has paid the fees, how much they have paid, and how much due they have.
- Admin can download all the details of the students as a image.
- Admin can search a particular student with roll number, he can fetch details of that student, and can download all those details as a image.

### **Data Management Module:**

- ➤ In this module a particular person can store his/her data.
- > User can check their data whenever they required.
- Every user data can be stored for multiple times.
- ➤ User can compare their data with their previous one.

# **SOURCE CODE**:

# **Code for all components:**

# Code of NavBar.js:

```
import React, { useContext } from 'react'
import './NavBar.css';
import SearchIcon from '@mui/icons-material/Search';
import NotificationsIcon from '@mui/icons-material/Notifications';
import { Button } from '@mui/material';
import { Link } from 'react-router-dom'
```

```
import { FormContext } from './FormContext';
import { UserContext } from './UserContext';
function NavBar() {
  const { user, setUser } = useContext(FormContext)
  const { student, setStudent } = useContext(UserContext)
  return (
     <div className='header'>
        <img className='header__logo' src={process.env.PUBLIC_URL + '/images/SV-logo.png'}</pre>
alt="Cognizant-logo" />
        <div className="header middle">
          <div className="header middle break1">
             <div className="header option">
                <a href="/" className='active'><b>Home</b></a>
             </div>
          </div>
          <div className="header__middle__break4">
             <div className="header option">
                <Link to="/administration"> <a href="/"
className='notactive'>Administration</a></Link>
             </div>
             <div className="header option">
                <Link to="/ourpredecessors"> <a href="/" className='notactive'>Our
Predecessors</a></Link>
             </div>
          </div>
          <div className="header middle break2">
             <div className="header option">
                <Link to="/vision"> <a href="/" className='notactive'>Vision & MIsion</a></Link>
             </div>
             <div className="header__option">
                <Link to="/campustour"> <a href="/" className='notactive'>Campus Tour</a></Link>
             <div className="header__option">
                <Link to="/notifications"> <a href="/" className='notactive'>Notifications</a></Link>
             </div>
          </div>
          <div className="header middle break3">
             <div className="header option">
                <Link to="/about"> <a href="/" className='notactive'>About Us</a></Link>
             </div>
             <div className="header option">
               <Link to="/contact"> <a href="/" className='notactive'>Contact Us</a></Link>
             </div>
          </div>
        </div>
        <div className="header right">
          <SearchIcon />
          <NotificationsIcon />
          <div className="circle">
             2
          </div>
          {(!user && !student) && <Link to="/departments"><Button variant="contained"
className='btn'>Login</Button></Link>}
```

## Code for NavBar.css

```
.header {
  z-index: 1;
  padding: 15px 5%;
  height: 10%;
  display: flex;
  justify-content: space-between;
  position: sticky;
  background-color: white;
  /* z-index: 100; */
  top: 0;
  align-items: center;
  box-shadow: 0px 5px 6px -9px rgba(0,0,0,0.75);
  font-family: Arial, Helvetica, sans-serif;
}
.header > img {
  width: 30%;
  margin-left: 0px;
  object-fit: contain;
}
.header__middle {
  margin: 5px;
  font-size: 16px;
  display: flex;
  justify-content: space-between;
  flex-direction: row;
.header__middle .header__middle__break1 {
  display: flex;
  flex-direction: column;
}
.header middle .header middle break2 {
  display: flex;
  flex-direction: column;
.header__middle .header__middle__break3 {
  display: flex;
  flex-direction: column;
```

```
.header__middle .header__middle__break1 .header__option Link .noactive {
  text-decoration: none;
  color: #000048;
  margin: 25px;
  font-weight: 500;
}
.header__option > a {
  text-decoration: none;
  color: #000048;
  margin: 25px;
  font-weight: 500;
  font-size: 12px;
}
.notactive {
  text-decoration: none;
  color: #000048;
  margin: 25px;
  font-weight: 500;
}
.header__option__with__icon > a {
  text-decoration: none;
  color: #000048;
  /* margin: 10px; */
  font-weight: 500;
}
.header__option__with__icon {
  margin-top: -3px;
  display: flex;
  justify-content: space-between;
   align-items: center;
}
.header__right {
  margin-right: -30px;
  display: flex;
  align-items: center;
  /* justify-content: space-between; */
.header__right .MuiSvgIcon-root {
  padding: 8px;
  font-size: 40px;
}
.circle {
  position: absolute;
  margin-left: 58px;
  margin-bottom: 13px;
  height: 11px;
  width: 11px;
  background-color: red;
  border-radius: 50%;
  font-size: 5px;
}
.btn {
```

```
padding: 10px;
  margin-left: 15px;
  text-align: center;
  height: 25px;
  background-color: #E9F2FE;
  width: 80px;
  border: none;
  border-radius: 20px;
  color: #000048;
  font-weight: bold;
  box-shadow: none;
  font-size: 12px;
  text-transform: none;
}
.header__right .btn > span {
   margin-bottom: -20px;
.btn:hover {
  background-color: #000048;
   color: white;
}
.header__right #plans {
  margin-left: 15px;
  border: none;
  height: 25px;
  width: 120px;
  color: #000048;
  font-weight: bold;
  background-color: #E9F2FE;
  border-radius: 20px;
  font-size: 12px;
  text-align: center;
}
.header__right #plans:focus {
  outline: none;
}
```

# **Code for HomePage:**

```
data-target="#carouselExampleIndicators" data-slide-to="1">
      data-target="#carouselExampleIndicators" data-slide-to="2">
     <div class="carousel-inner">
      <div class="carousel-item active">
        A. P. J. Abdul Kalam
        <img id='slider01' src={process.env.PUBLIC_URL + '/images/double.png'} alt="" />
        <img class="d-block w-100" src={process.env.PUBLIC_URL + '/images/slider.png'} alt="First slide"</pre>
/>
      </div>
      <div class="carousel-item">
        Dr. Babasaheb Ambedkar
        <img id='slider11' src={process.env.PUBLIC_URL + '/images/double.png'} alt="" />
        <img class="d-block w-100" src={process.env.PUBLIC_URL + '/images/slider1.png'} alt="Second"</pre>
slide" />
      </div>
      <div class="carousel-item">
        Nelson Rolihlahla Mandela
        <img id='slider31' src={process.env.PUBLIC URL + '/images/double.png'} alt="" />
        <img class="d-block w-100" src={process.env.PUBLIC_URL + '/images/slider3.png'} alt="Third
slide" />
      </div>
     </div>
     <a class="carousel-control-prev" href="#carouselExampleIndicators" role="button" data-
slide="prev">
      <span class="carousel-control-prev-icon" aria-hidden="true"></span>
      <span class="sr-only">Previous</span>
     <a class="carousel-control-next" href="#carouselExampleIndicators" role="button" data-
slide="next">
      <span class="carousel-control-next-icon" aria-hidden="true"></span>
      <span class="sr-only">Next</span>
     </a>
    </div>
    <div className="banner">
     <div className="latest updates">
      <span className='latest'>LATEST UPDATES</span>
      <marquee behavior="scroll" direction="left" className="update">
        <img src={process.env.PUBLIC_URL + "/images/new.gif"} alt="" /> MCA 1st Year - External
laboratory Examinations will be held on 22/03/2024
        <img src={process.env.PUBLIC_URL + "/images/new.gif"} alt="" /> MCA 2nd Year - External
laboratory Examinations will be held on 28/03/2024
        <img src={process.env.PUBLIC URL + "/images/new.gif"} alt="" /> ICET notification has been
released
      </marquee>
     </div>
     <div className="banner content">
      <div className="banner left">
        <h2>Experience Hassle-Free Fee Transactions!</h2>
        <h5>...No need to waste your time</h5>
        <div className="buttons">
         <Link to="/departments"><Button variant="contained" className='btn'>Pay
```

# **Code for HomePage CSS:**

```
h2{
  font-size: 40px;
  width: 70%;
  margin-left: 80px;
  font-weight: 1000;
}
h5 {
  color: black;
  font-weight: 550;
  margin-left: 30%;
}
.latest_updates {
  display: block;
  margin-left: 80px;
  padding-top: 30px;
  margin-bottom: -20px;
  /* border: 1px solid black; */
  width: 95%;
  height: 25px;
.latest__updates marquee img{
   margin-left: 20px;
}
.latest {
  background-color: aqua;
  height: 80px;
  border-right: 1px solid black;
}
.update {
  color: red;
  position: absolute;
   margin-bottom: 60px;
```

```
.buttons {
   margin-top: 50px;
.buttons h4 {
  margin-top: 20px;
  margin-left: 80px;
  font-size: 20px;
}
/* .student__btn {
  bor
} */
.buttons .btn{
  margin-left: 80px;
  padding: 10px;
  text-align: center;
  height: 50px;
  background-color: #000048;
  width: 150px;
  border: none;
  border-radius: 10px;
  color: #E2FEF9;
  font-weight: bold;
  box-shadow: none;
  font-size: 17px;
  text-transform: none;
  z-index: 0;
  cursor: pointer;
}
.buttons .btn:hover {
  background-color: #000048;
  box-shadow: none;
}
.banner {
  height: 500px;
  background: rgb(55, 182, 246);
  margin-bottom: -100px;
  z-index: -1;
}
.banner .banner__content{
  display: flex;
   align-items: center;
.banner__left {
  padding-top: 90px;
.banner__right img{
  height: 300px;
  width: 550px;
  margin-right: 100px;
  /* margin-bottom: 200px; */
}
```

```
#slider1 {
  position: absolute;
  margin-top: 230px;
  margin-left: 670px;
  color: white;
  font-size: 13px;
  font-weight: 800;
}
#slider0 {
  position: absolute;
  margin-top: 220px;
  margin-left: 660px;
  color: white;
  font-size: 13px;
  font-weight: 800;
}
#slider3 {
  position: absolute;
  margin-top: 220px;
  margin-left: 600px;
  color: white;
  font-weight: 800;
}
#slider01{
  position: absolute;
  margin-top: 180px;
  margin-left: 830px;
  z-index: 2200;
  opacity: 20;
  /* height: 150px; */
  /* background-color: red; */
}
#slider11{
  position: absolute;
  margin-top: 170px;
  margin-left: 900px;
  z-index: 2200;
  opacity: 9;
  /* height: 150px; */
  /* background-color: red; */
}
#slider31{
  position: absolute;
  margin-top: 140px;
  margin-left: 720px;
  z-index: 2200;
  opacity: 9;
  /* height: 150px; */
  /* background-color: red; */
}
```

### Code for StudentSignUp.js:

```
import axios from 'axios';
import React, { useContext, useState } from 'react'
import { DeptContext } from './DeptContext';
function StudentSignup() {
 const { dept, setDept } = useContext(DeptContext)
 const [studentDetails, setStudentDetails] = useState({
  rollnumber: ",
  studentName: ",
  department: dept,
  semester: ",
  email: ",
  password: ",
  contactNumber: ",
  address: "
 });
 const handleChange = (e) => {
  const { name, value } = e.target;
  setStudentDetails(prevState => ({
    ...prevState,
    [name]: value
  }));
 };
 const handleSubmit = async (e) => {
   e.preventDefault();
    // Make POST request to your API endpoint
    const response = await axios.post('http://localhost:8081/api/students/signup', studentDetails);
    console.log(response.data); // Handle response from server
    if (response.status === 200) {
     alert("Hey " + response.data.studentName + "! Thank you for signing in. ")
     window.location.href = '/student';
  } catch (error) {
    console.error('Error submitting form:', error);
    alert('Error submitting form:' + error)
  }
 };
 return (
   <div className="login-box">
    <h2>Student Registration</h2>
    <form onSubmit={handleSubmit} autoComplete="off">
     <div className="user-box">
       <input
        type="text"
```

```
id="rollnumber"
  name="rollnumber"
  value={studentDetails.rollnumber}
  autoComplete="false"
  onChange={handleChange}
  required
 />
 <label htmlFor="rollnumber">Roll Number</label>
<div className="user-box">
 <input
  type="text"
  id="studentName"
  name="studentName"
  value={studentDetails.studentName}
  autoComplete="false"
  onChange={handleChange}
  required
 />
 <label htmlFor="studentName">Student Name</label>
<div className="user-box">
 <input
  type="text"
  id="department"
  name="department"
  value={dept}
  onChange={handleChange}
  autoComplete="off"
  required
 />
 <label htmlFor="studentName">Department Name</label>
</div>
<div className="user-box">
 <input
  type="text"
  id="branch"
  name="branch"
  value={studentDetails.branch}
  autoComplete="false"
  onChange={handleChange}
  required
 />
 <label htmlFor="branch">Branch</label>
</div>
<div className="user-box">
 <input
  type="text"
  id="semester"
  name="semester"
  value={studentDetails.semester}
  autoComplete="false"
  onChange={handleChange}
  required
```

```
/>
   <label htmlFor="branch">Semester</label>
  </div>
  <div className="user-box">
   <input
    type="email"
    id="email"
    name="email"
    value={studentDetails.email}
    autoComplete="false"
    onChange={handleChange}
    required
   />
   <label htmlFor="email">Email</label>
  </div>
  <div className="user-box">
   <input
    type="password"
    id="password"
    name="password"
    value={studentDetails.password}
    autoComplete="false"
    onChange={handleChange}
    required
   />
   <label htmlFor="password">Password</label>
  </div>
  <div className="user-box">
   <input
    type="text"
    id="contactNumber"
    name="contactNumber"
    value={studentDetails.contactNumber}
    autoComplete="false"
    onChange={handleChange}
    required
   />
   <label htmlFor="contactNumber">Contact Number</label>
  </div>
  <div className="user-box">
   <input
    type="text"
    id="address"
    name="address"
    value={studentDetails.address}
    autoComplete="false"
    onChange={handleChange}
    required
   />
   <label htmlFor="address">Address</label>
  <button type="submit">Submit</button>
 </form>
</div>
```

```
)
}
export default StudentSignup
```

# **Code for StudentSignIn.js:**

```
import React, { useContext, useState } from 'react'
import './AdminGetInPage.css'
import axios from 'axios';
import { Link, useNavigate } from 'react-router-dom';
import { UserContext } from './UserContext';
import { FormContext } from './FormContext';
function StudentGetInPage() {
 const navigate = useNavigate();
 const { user, setUser } = useContext(FormContext)
 const { student, setStudent } = useContext(UserContext)
 const [errorMsg, setErrorMsg] = useState(false)
 const [adminDetails, setAdminDetails] = useState({
  email: ",
  password: "
 });
 const handleChange = (e) => {
  const { name, value } = e.target;
   setAdminDetails(prevState => ({
    ...prevState,
    [name]: value
  }));
 };
 const handleSubmit = async (e) => {
   e.preventDefault();
  try {
    // Make POST request to your API endpoint
    const response = await axios.post('http://localhost:8081/api/students/signin', adminDetails);
    console.log(response.status); // Handle response from server
    setErrorMsg(false)
    if (response.status === 200) {
     alert("Hey " + response.data.studentName + "! You have been signed in successfully ")
     setUser("")
     setStudent(response.data.studentName)
     navigate('/studentHome', { state: { name: response.data } });
     // window.location.href = '/studentHome';
   } catch (error) {
    console.error('Error submitting form:', error);
```

```
setErrorMsg(true);
   }
 };
 return (
   <div className="login-box">
    <h2>Student Login</h2>
    <form onSubmit={handleSubmit} autoComplete="off">
     <div className="user-box">
       <input
        type="email"
        id="email"
        name="email"
        value={adminDetails.email}
        autoComplete="false"
        onChange={handleChange}
        required
       />
       <label htmlFor="email">Email</label>
     <div className="user-box">
       <input
        type="password"
        id="password"
        name="password"
        value={adminDetails.password}
        autoComplete="false"
        onChange={handleChange}
        required
       />
       <label htmlFor="password">Password</label>
     </div>
     {errorMsg && Invalid Credentials}
     <button type="submit">Submit</button>
    </form>
    >Don't have an accout? <Link to="/studentSignup"><button>Sign Up</button></Link>
   </div>
)
}
export default StudentGetInPage
```

# **Code for StudentHome.js:**

```
import React, { useContext, useEffect, useState } from "react";
import { UserContext } from "./UserContext";
import { useLocation } from "react-router-dom";
```

```
import "./StudentHome.css";
import axios from "axios";
import jsPDF from "jspdf";
import { DeptContext } from "./DeptContext";
function StudentHome() {
 const { dept, setDept } = useContext(DeptContext)
 const [totalAmount, setTotalAmount] = useState(0);
 const [amountPaid, setAmountPaid] = useState(0);
 const [totalAmountE, setTotalAmountE] = useState(0);
 const [amountPaidE, setAmountPaidE] = useState(0);
 const [balance, setBalance] = useState(0);
 const [balanceE, setBalanceE] = useState(0);
 const [paymentAmount, setPaymentAmount] = useState(0);
 const [paymentAmountExam, setPaymentAmountExam] = useState(0);
 // const { student } = useContext(UserContext);
 const location = useLocation();
 const [feeType, setFeeType] = useState("");
 const [sem, setSem] = useState("");
 const [showTutionFields, setShowTutionFields] = useState(false);
 const [showExamFields, setShowExamFields] = useState(false);
 const handleSem = (e) => {
  setSem(e.target.value)
 const handleFeeTypeChange = (event) => {
  const selectedOption = event.target.value;
  setFeeType(selectedOption);
  if (selectedOption === "Tution Fees") {
    setShowTutionFields(true);
    setShowExamFields(false);
    setTotalAmount(15000);
    setAmountPaid(0);
    console.log(totalAmount, amountPaid)
    setBalance(totalAmount - amountPaid);
   } else if (selectedOption === "Exam Fees") {
    console.log("hello");
    setShowTutionFields(false);
    setShowExamFields(true);
    setTotalAmountE(1200);
    setAmountPaidE(0);
    setBalanceE(totalAmountE - amountPaidE);
  } else {
    setShowTutionFields(false);
    setShowExamFields(false);
  }
 };
 const data = location.state.name;
 let dataa;
 useEffect(() => {
  axios.get(`http://localhost:8081/api/students/getStudent/${data.rollnumber}`).then((res) => {
```

```
dataa = res.data
 });
 console.log(dataa)
}, [])
const [studentDetails, setStudentDetails] = useState({
 rollnumber: data.rollnumber,
 studentName: data.studentName,
 department: data.department,
 branch: data.branch,
 semester: data.semester,
 email: data.email.
 password: data.password,
 contactNumber: data.contactNumber,
 address: data.address,
 paidAmountOfTutionFees: data.paidAmountOfTutionFees,
 paidAmountOfExamFees: data.paidAmountOfExamFees,
 dueOfTutionFees: data.dueOfTutionFees,
 dueOfExamFees: data.dueOfExamFees
});
const handleChange = (e) => {
 const { name, value } = e.target;
 setStudentDetails((prevState) => ({
  ...prevState,
  [name]: value,
 }));
};
let studentName = data.studentName;
const receipt = studentName + data.contactNumber.toString().substring(1, 5);
const rollnumber = data.rollnumber;
const handlePayButtonClickTution = () => {
 const amountToPay = parseFloat(prompt("Enter the amount to pay:", "0"));
 studentDetails.paidAmountOfTutionFees = amountToPay
 studentDetails.dueOfTutionFees = totalAmount - amountToPay
 axios.patch(`http://localhost:8081/api/students/updatePayment/${data.rollnumber}`, studentDetails);
 if (!isNaN(amountToPay)) {
  setPaymentAmount(amountToPay);
 setAmountPaid(amountToPay);
 setBalance(totalAmount - amountToPay);
const handlePayButtonClickExam = () => {
 const amountToPay = parseFloat(prompt("Enter the amount to pay:", "0"));
 // setStudentDetails((prevState) => ({
 // ...prevState,
 // [paidAmountOfExamFees]: amountToPay,
 // [dueOfExamFees]: totalAmountE - amountToPay
 studentDetails.paidAmountOfExamFees = amountToPay
 studentDetails.dueOfExamFees = totalAmountE - amountToPay
 console.log(studentDetails)
 axios.patch(`http://localhost:8081/api/students/updatePayment/${data.rollnumber}`, studentDetails);
```

```
if (!isNaN(amountToPay)) {
  setPaymentAmountExam(amountToPay);
 setAmountPaidE(amountToPay);
 setBalance(totalAmountE - amountToPay);
};
const handleSubmit = async (e) => {
 e.preventDefault();
 try {
  // Make POST request to your API endpoint
  const headers = {
    "Content-Type": "application/json",
    "Access-Control-Allow-Origin": "*",
  };
  console.log(studentDetails);
  const response = await axios.patch(
     http://localhost:8081/api/students/update/${data.rollnumber}`,
    studentDetails,
    { headers }
  );
  console.log(response.data); // Handle response from server
  if (response.status === 200) {
    alert(
     "Hey " + response.data.studentName + "Your data has been updated"
    window.location.href = "/studentHome";
  }
 } catch (error) {
  console.error("Error submitting form:", error);
  // alert('Error submitting form:'+ error)
 }
};
const generateReceipt = () => {
 axios.get(`http://localhost:8081/api/students/getStudent/${data.rollnumber}`).then((res) => {
  dataa = res.data
 });
 const doc = new jsPDF();
 doc.setFontSize(22);
 doc.rect(10, 5, 180, 280);
 doc.text(
   `Your ${showTutionFields ? "Tution Fees" : "Exam Fees"} Receipt`,
  50,
  20
 );
 doc.setFontSize(16);
 doc.text(`Receipt No: ${receipt}`, 20, 30);
 doc.text(`Student Name: ${studentName}`, 20, 40);
 doc.text(`Roll Number: ${rollnumber}`, 20, 50);
 doc.text(`Department: ${dept}`, 20, 60);
```

```
doc.text(`Semester: ${sem}`, 20, 70);
 doc.text(`Fee Type: ${feeType}`, 20, 80);
 doc.text(`Total Amount to Pay for tution fees: ${totalAmount}`, 20, 90);
 doc.text(`Paid Amount for tution fees: ${amountPaid}`, 20, 100);
 doc.text(`Due for tution fees: ${balance}`, 20, 110);
 doc.text(`Total Amount to Pay for tution fees: ${totalAmountE}`, 20, 90);
 doc.text(`Paid Amount for tution fees: ${amountPaidE}`, 20, 100);
 doc.text(`Due for tution fees: ${balanceE}`, 20, 110);
 doc.text(`Email Address: ${data.email}`, 20, 120);
 doc.text(`Branch: ${data.branch}`, 20, 130);
 doc.text(`Contact Number: ${data.contactNumber}`, 20, 140);
 doc.text(`Exam Fees: ${data.examFees}`, 20, 150);
 doc.text(`Tution Fees: ${data.tutionFees}`, 20, 160);
 doc.text(`Paid amount of tution fees: ${data.paidAmountOfTutionFees}`, 20, 170);
 doc.text(`Paid amount of exam fees: ${data.paidAmountOfExamFees}`, 20, 180);
 doc.text(`Due of Tution Fees: ${data.dueOfTutionFees}`, 20, 190);
 doc.text(`Due of Exam Fees: ${data.dueOfExamFees}`, 20, 200);
 doc.text(`Payment Status: paid`, 20, 210);
 doc.text(`Paid Date: 2024-03-19`, 20, 220);
 // Save the PDF
 doc.save("Receipt.pdf");
 alert("Your receipt has been downloaded you can check that");
};
return (
 <div className="hello">
  <h1>Hi {studentName}! Welcome to SV University Fees Portal</h1>
  Receipt No.
      <input type="text" value={receipt} />
    Student Name:
      <input type="text" value={studentDetails.studentName} />
     Roll Number:
      <input type="text" value={studentDetails.rollnumber} />
     Department:
      <input type="text" value={studentDetails.department} />
     Semester:
```

```
<select value={sem} onChange={handleSem}>
   <option value="">Select Semester</option>
   <option value="I">I</option>
   <option value="II">II</option>
   <option value="I">III</option>
   <option value="II">IV</option>
  </select>
 Address
 <input type="text" value={studentDetails.address} />
 <select value={feeType} onChange={handleFeeTypeChange}>
   <option value="">Select Fees Type</option>
   <option value="Tution Fees">Tution Fees
   <option value="Exam Fees">Exam Fees
  </select>
  {showTutionFields && (
   <div>
     <h3>Tution Fees Details</h3>
      Total Amount to pay: <input type="text" value={totalAmount} />
     </label>
     <label>
     Paid amount: <input type="text" value={amountPaid} />
     </label>
     <label>
      Due: <input type="text" value={totalAmount - amountPaid} />
     <button onClick={handlePayButtonClickTution}>Pay</button>
    \{paymentAmount > 0 \&\& (
      <div>
       <h4>You have paid: {paymentAmount}</h4>
      </div>
    )}
   </div>
  )}
  {showExamFields && (
   <div>
     <h3>Exam Fees Details</h3>
     <label>
     Total Amount to pay: <input type="text" value={totalAmountE} />
     </label>
     <label>
     Paid amount: <input type="text" value={amountPaidE} />
     </label>
     <label>
```

```
Due: <input type="text" value={totalAmountE - amountPaidE} />
     </label>
     <button onClick={handlePayButtonClickExam}>Pay</button>
     {paymentAmountExam > 0 && (
      You have paid: {paymentAmountExam}
     )}
    </div>
   )}
  Email Address:
   <input type="text" value={data.email} />
  Branch:
  <input type="text" value={data.branch} />
 Contact Number:
   <input type="text" value={data.contactNumber} />
  <button
type="button"
class="btn btn-primary edit"
data-bs-toggle="modal"
data-bs-target="#exampleModal"
Edit Details
</button>
<div
class="modal fade"
id="exampleModal"
tabindex="-1"
aria-labelledby="exampleModalLabel"
aria-hidden="true"
 <div class="modal-dialog">
  <div class="modal-content">
   <div class="modal-header">
    <h1 class="modal-title fs-5" id="exampleModalLabel">
     Modal title
    </h1>
    <button
     type="button"
     class="btn-close"
```

```
data-bs-dismiss="modal"
  aria-label="Close"
 ></button>
</div>
<div class="modal-body">
 <form>
  <div className="user-box">
   <label htmlFor="rollnumber">Roll Number: </label>
   <input
    type="text"
    id="rollnumber"
    name="rollnumber"
    value={data.rollnumber}
    onChange={handleChange}
    readOnly={false}
    required
   />
  </div>
  <div className="user-box">
   <label htmlFor="studentName">Student Name: </label>
   <input
    type="text"
    id="studentName"
    name="studentName"
    defaultValue={studentName}
    onChange={handleChange}
    readOnly={false}
    required
   />
  </div>
  <div className="user-box">
   <label htmlFor="password">Department:</label>
   <input
    type="department"
    id="department"
    name="department"
    defaultValue={data.department}
    onChange={handleChange}
    required
   />
  </div>
  <div className="user-box">
   <label htmlFor="password">Semester:</label>
   <input
    type="semester"
    id="semester"
    name="semester"
    defaultValue={data.semester}
    onChange={handleChange}
    required
   />
  </div>
  <div className="user-box">
   <label htmlFor="email">Email: </label>
```

```
<input
    type="email"
    id="email"
    name="email"
    value={data.email}
    onChange={handleChange}
    required
   />
  </div>
  <div className="user-box">
   <label htmlFor="branch">Branch</label>
   <input
    type="text"
    id="branch"
    name="branch"
    defaultValue={data.branch}
    onChange={handleChange}
    required
   />
  </div>
  <div className="user-box">
   <label htmlFor="contactNumber">Contact Number</label>
   <input
    type="text"
    id="contactNumber"
    name="contactNumber"
    defaultValue={data.contactNumber}
    onChange={handleChange}
    required
   />
  </div>
  <div className="user-box">
   <label htmlFor="address">Address: </label>
   <input
    type="text"
    id="address"
    name="address"
    defaultValue={data.address}
    onChange={handleChange}
    required
   />
  </div>
  {/* <button type="submit">Submit</button> */}
 </form>
</div>
<div class="modal-footer">
 <button
  type="button"
  class="btn btn-secondary"
  data-bs-dismiss="modal"
 >
  Close
 </button>
```

```
<button
           onClick={handleSubmit}
           type="button"
           class="btn btn-primary"
           Save changes
          </button>
         </div>
       </div>
      </div>
     </div>
     <button class="btn btn-primary download" onClick={generateReceipt}>
      Download Receipt ( PDF )
     </button>
   </div>
);
}
```

export default StudentHome;

### **Code for StudentHome.css:**

```
color: white;
}
td {
   color: rgb(0, 0, 0);
button {
   margin-left: 20px;
.hello .download {
  font-size: 13px;
  height: 50px;
  width: 200px;
}
.hello .edit {
  margin-left: 40%;
  font-size: 13px;
  height: 50px;
   width: 200px;
}
h4 {
  font-size: 20px;
}
```

### Code for AdminSignUp.js:

```
import React, { useState } from 'react'
import './AdminGetInPage.css'
import axios from 'axios';
function AdminSignUp() {
 const [adminDetails, setAdminDetails] = useState({
  empld: ",
  empName: ",
  email: ",
  password: ",
  contactNumber: ",
  address: "
 });
 const handleChange = (e) => {
   const { name, value } = e.target;
  setAdminDetails(prevState => ({
    ...prevState,
    [name]: value
  }));
 };
 const handleSubmit = async (e) => {
   e.preventDefault();
  try {
    // Make POST request to your API endpoint
    const response = await axios.post('http://localhost:8081/api/admins/signup', adminDetails);
    console.log(response.data); // Handle response from server
    if (response.status === 200) {
     alert("Hey " + response.data.empName + "! Thank you for signing up with us. ")
     window.location.href = '/admin';
    }
   } catch (error) {
    console.error('Error submitting form:', error);
    alert('Error submitting form:' + error)
  }
 };
 return (
   <div className='login-box'>
    <h2>Admin Registration</h2>
    <form onSubmit={handleSubmit} autoComplete="off">
     <div className="user-box">
       <input
        type="text"
        id="empld"
        name="empld"
        value={adminDetails.empld}
        autoComplete="false"
```

```
onChange={handleChange}
  required
 <label htmlFor="empld">Employee ID</label>
</div>
<div className="user-box">
 <input
  type="text"
  id="empName"
  name="empName"
  value={adminDetails.empName}
  autoComplete="false"
  onChange={handleChange}
  required
 />
 <label htmlFor="empName">Employee Name</label>
</div>
<div className="user-box">
 <input
  type="email"
  id="email"
  name="email"
  value={adminDetails.email}
  autoComplete="false"
  onChange={handleChange}
  required
 />
 <label htmlFor="email">Email</label>
</div>
<div className="user-box">
 <input
  type="password"
  id="password"
  name="password"
  value={adminDetails.password}
  autoComplete="false"
  onChange={handleChange}
  required
 />
 <label htmlFor="password">Password</label>
</div>
<div className="user-box">
 <input
  type="text"
  id="contactNumber"
  name="contactNumber"
  value={adminDetails.contactNumber}
  autoComplete="false"
  onChange={handleChange}
  required
 />
 <label htmlFor="contactNumber">Contact Number</label>
<div className="user-box">
```

```
<input
       type="text"
       id="address"
       name="address"
       value={adminDetails.address}
       autoComplete="false"
       onChange={handleChange}
       required
      />
      <label htmlFor="address">Address</label>
     </div>
     <button type="submit">Submit</button>
    </form>
  </div>
 )
export default AdminSignUp
```

# Code for AdminSignIn.js:

```
import React, { useContext, useState } from 'react'
import './AdminGetInPage.css'
import axios from 'axios';
import { Link, useNavigate } from 'react-router-dom';
import { FormContext } from './FormContext';
import { UserContext } from './UserContext';
function AdminGetInPage() {
 const { user, setUser } = useContext(FormContext)
 const { student, setStudent } = useContext(UserContext)
 const [errorMsg, setErrorMsg] = useState(false)
 const navigate = useNavigate();
 const [adminDetails, setAdminDetails] = useState({
  email: ",
  password: "
 });
 const handleChange = (e) => {
  const { name, value } = e.target;
  setAdminDetails(prevState => ({
    ...prevState,
    [name]: value
  }));
 };
 const handleSubmit = async (e) => {
   e.preventDefault();
  try {
    // Make POST request to your API endpoint
```

```
const response = await axios.post('http://localhost:8081/api/admins/signin', adminDetails);
  console.log(response.data); // Handle response from server
  setErrorMsg(false)
  if (response.status === 200) {
    alert("Hey " + response.data.empName + "! You have been signed in successfully ")
    setStudent("")
   setUser(response.data.empName)
   navigate('/adminHome', { state: { name: response.data } });
   // window.location.href = '/studentHome';
 } catch (error) {
  console.error('Error submitting form:', error);
  setErrorMsg(true)
};
return (
 <div className="login-box">
  <h2>Admin Login</h2>
  <form onSubmit={handleSubmit} autoComplete="off">
    <div className="user-box">
     <input
      type="email"
      id="email"
      name="email"
      value={adminDetails.email}
      autoComplete="false"
      onChange={handleChange}
      required
     />
     <label htmlFor="email">Email</label>
    </div>
    <div className="user-box">
     <input
      type="password"
      id="password"
      name="password"
      autoComplete="false"
      value={adminDetails.password}
      onChange={handleChange}
      required
     />
     <label htmlFor="password">Password</label>
    </div>
    <button type="submit">Submit</button>
    {errorMsg && Invalid Credentials}
   </form>
```

```
On't have an accout? <Link to="/adminSignup"><button>Sign Up</button></Link></div>
)
}
export default AdminGetInPage
```

# **Code for AdminSignIn.css:**

```
html {
  height: 100%;
 body {
  margin:0;
  padding:0;
  font-family: sans-serif;
  /* background: linear-gradient(#141e30, #243b55); */
 .login-box {
  margin-bottom: -280px;
  background: rgb(177, 176, 176);
  /* position: absolute; */
  margin-top: 40%;
  margin-left: 50%;
  width: 400px;
  padding: 40px;
  transform: translate(-50%, -50%);
  /* background: rgba(0,0,0,.5); */
  box-sizing: border-box;
  box-shadow: 0 15px 25px rgba(214, 212, 212, 0.6);
  border-radius: 10px;
 }
 .login-box p {
  color: white;
 }
 .login-box h2 {
  margin: 0 0 30px;
  padding: 0;
  color: #fff;
  text-align: center;
 .login-box .user-box {
  position: relative;
 .login-box .user-box input {
   width: 100%;
  padding: 10px 0;
```

```
font-size: 16px;
 color: #fff;
 margin-bottom: 30px;
 border: none;
 border-bottom: 1px solid #fff;
 outline: none;
 background: transparent;
.login-box .user-box label {
 position: absolute;
 top:0;
 left: 0;
 padding: 10px 0;
 font-size: 16px;
 color: #fff;
 pointer-events: none;
 transition: .5s;
}
.login-box .user-box input:focus ~ label,
.login-box .user-box input:valid ~ label {
 top: -20px;
 left: 0;
 color: #03e9f4;
 font-size: 12px;
}
.login-box form button {
 cursor: pointer;
 background: linear-gradient(#141e30, #243b55);
 position: relative;
 display: inline-block;
 padding: 10px 20px;
 color: #03e9f4;
 font-size: 16px;
 text-decoration: none;
 text-transform: uppercase;
 overflow: hidden;
 transition: .5s;
 margin-top: 40px;
 letter-spacing: 4px
}
.login-box button:hover {
 background: linear-gradient(#141e30, #243b55);
 color: #fff;
 border-radius: 5px;
 box-shadow: 0 0 5px #fcfcfc,
          0 0 2px #b9b9b9,
          0 0 5px #ffffff,
          0 0 10px #ffffff;
}
```

```
.login-box button span {
 position: absolute;
 display: block;
.login-box button span:nth-child(1) {
 top: 0;
 left: -100%;
 width: 100%;
 height: 2px;
 background: linear-gradient(90deg, transparent, #03e9f4);
 animation: btn-anim1 1s linear infinite;
}
@keyframes btn-anim1 {
 0% {
  left: -100%;
 50%,100% {
  left: 100%;
}
.login-box button span:nth-child(2) {
 top: -100%;
 right: 0;
 width: 2px;
 height: 100%;
 background: linear-gradient(180deg, transparent, #03e9f4);
 animation: btn-anim2 1s linear infinite;
 animation-delay: .25s
}
@keyframes btn-anim2 {
 0% {
  top: -100%;
 50%,100% {
  top: 100%;
 }
}
.login-box button span:nth-child(3) {
 bottom: 0;
 right: -100%;
 width: 100%;
 height: 2px;
 background: linear-gradient(270deg, transparent, #03e9f4);
 animation: btn-anim3 1s linear infinite;
 animation-delay: .5s
}
@keyframes btn-anim3 {
 0% {
```

```
right: -100%;
 50%,100% {
  right: 100%;
.login-box button span:nth-child(4) {
 bottom: -100%;
 left: 0;
 width: 2px;
 height: 100%;
 background: linear-gradient(360deg, transparent, #03e9f4);
 animation: btn-anim4 1s linear infinite;
 animation-delay: .75s
}
@keyframes btn-anim4 {
 0% {
  bottom: -100%;
 50%,100% {
  bottom: 100%;
 }
```

# **Code for AdminHome.js:**

```
import React, { useEffect, useRef, useState } from "react";
import { useLocation } from "react-router-dom";
import "./AdminHome.css";
import axios from "axios";
import ArrowCircleRightIcon from '@mui/icons-material/ArrowCircleRight';
import html2canvas from 'html2canvas';
import jsPDF from "jspdf";
function AdminHome() {
 const [students, setStudents] = useState([]);
 const location = useLocation();
 const data = location.state.name;
 const [selectedStudent, setSelectedStudent] = useState(null);
 const [clickedOnSubmit, setClickedOnSubmit] = useState(false);
 const tableRef = useRef(null);
 // const [clickedOnSubmit, setClickedOnSubmit] = useState(false)
 // const [searchesStudent,/ setSearchedStudent] = useState()
 let rollnumber;
 // let searchesStudent;
 const [searchesStudent, setSearchesStudent] = useState({
  rollnumber: "",
  studentName: "",
  department: "",
  semester: 0,
```

```
email: ""
 branch: "".
 contactNumber: "",
 address: "",
 paymentStatus: ",
 paidDate: "",
 tutionFees: 0,
 examFees: 0,
 paidAmountOfTutionFees: 0,
 paidAmountOfExamFees: 0,
 dueOfTutionFees: 0,
 dueOfExamFees: 0
});
useEffect(() => {
 fetchStudents();
}, []);
const fetchStudents = () => {
 console.log(searchesStudent.rollnumber)
  .get("http://localhost:8081/api/students/allstudents")
  .then((res) => {
    console.log(res);
    setStudents(res.data);
  .catch((error) => {
    console.error("Error fetching students:", error);
  });
};
const getOneStudent = async () => {
 setClickedOnSubmit(true);
 const roll = document.getElementById('rollnumber').value;
 await axios.get(`http://localhost:8081/api/students/oneStudent/${roll}`).then((res) => {
  console.log(res.data.rollnumber);
  rollnumber = res.data.rollnumber
  // setSearchedStudent(res.data);
  searchesStudent.rollnumber = res.data.rollnumber
  searchesStudent.studentName = res.data.studentName
  searchesStudent.department = res.data.department
  searchesStudent.semester = res.data.semester
  searchesStudent.email = res.data.email
  searchesStudent.branch = res.data.branch
  searchesStudent.contactNumber = res.data.contactNumber
  searchesStudent.address = res.data.address
  searchesStudent.paymentStatus = res.data.paymentStatus
  searchesStudent.paidDate = res.data.paidDate
  searchesStudent.tutionFees = res.data.tutionFees
  searchesStudent.examFees = res.data.examFees
  searchesStudent.paidAmountOfTutionFees = res.data.paidAmountOfTutionFees
  searchesStudent.paidAmountOfExamFees = res.data.paidAmountOfExamFees
  searchesStudent.dueOfTutionFees = res.data.dueOfTutionFees
  searchesStudent.dueOfExamFees = res.data.dueOfExamFees
```

```
})
 console.log(fetchStudents())
const handleAll = () => {
 setClickedOnSubmit(false)
const generatePDF = async () => {
 try {
  const canvas = await html2canvas(tableRef.current);
  const image = canvas.toDataURL('image/png', 1.0);
  // You can now save or display the image as needed
  // For example, download the image:
  const link = document.createElement('a');
  link.href = image;
  link.download = 'student-details.png';
  link.click();
 } catch (error) {
  console.error('Error converting table to image:', error);
 }
}
return (
 <div>
  <h1>Welcome {data.empName}</h1>
  As a admin you can manage all students data
  <div className="ui left">
   <form action="">
    <input type="text" placeholder='Enter Student rollnumber' id='rollnumber' onKeyDown={(e) => {
     if (e.key === 'Enter') {
      e.preventDefault();
      getOneStudent();
    }} required />
    <abbr title='Submit NPI'><ArrowCircleRightIcon onClick={getOneStudent} /></abbr>
   </form>
   <button onClick={handleAll}>Show All Students</button>
  {!clickedOnSubmit && <>
   <thead>
    Roll Number
      Student Name
      Department
     Semester
      Email
     Branch
      Contact Number
      Address
      Tution Fees
      Exam Fees
      Paid Amount of tution fees
      Paid Amount of exam fees
```

```
Due of tution fees
  Due of exam fees
  Payment Status
  paid
 </thead>
{students.map((student) => (
  {student.rollnumber}
   {student.studentName}
   {student.department}
   {student.semester}
   {student.email}
   {student.branch}
   {student.contactNumber}
   {student.address}
   {student.tutionFees}
   {student.examFees}
   {student.paidAmountOfTutionFees}
   {student.paidAmountOfExamFees}
   {student.dueOfTutionFees}
   {student.dueOfExamFees}
   {student.paymentStatus}
   {student.paidDate}
  ))}
<button className="printBtn" onClick={generatePDF}>
 print student details
</button></>}
clickedOnSubmit && <>
 <thead>
  Roll Number
   Student Name
   Department
   Semester
   Email
   Branch
   Contact Number
   Address
   Tution Fees
   Exam Fees
   Paid Amount of tution fees
   Paid Amount of exam fees
   Due of tution fees
   Due of exam fees
   Payment Status
   paid
  </thead>
```

```
{/* {searchesStudent} */}
       {searchesStudent.rollnumber}
       {searchesStudent.studentName}
       {searchesStudent.department}
       {searchesStudent.semester}
       {searchesStudent.email}
       {searchesStudent.branch}
       {searchesStudent.contactNumber}
       {searchesStudent.address}
       {searchesStudent.tutionFees}
       {searchesStudent.examFees}
       {searchesStudent.paidAmountOfTutionFees}
       {searchesStudent.paidAmountOfExamFees}
       {searchesStudent.dueOfTutionFees}
       {searchesStudent.dueOfExamFees}
       {searchesStudent.paymentStatus}
       {searchesStudent.paidDate}
      <button className="printBtn" onClick={generatePDF}>print</button></>
   }
  </div>
);
}
```

#### **Code for AdminHome.css:**

export default AdminHome;

```
p {
   color: white;
}
.ui__left {
   display: flex;
.ui__left button {
   border: none;
   height: 30px;
   width: 300px;
   background-color: bisque;
}
h1{
   color: #000;
p{
   color: #000;
.printBtn {
margin-left: 45%;
```

```
border: none;
width: 150px;
background-color: bisque;
}
```

# Code for About.js:

Sri Venkateswara University was established in 1954, in Tirupati to cater the educational needs and aspirations of people of the Rayalaseema Region of Andhra Pradesh. After completing 68 years of excellence in teaching, research, extension and outreach activities, the University is committed to cater the needs of higher education offering a full range of post-graduate programs in Arts, Sciences, Law, Management, Education, Physical Education, Engineering and Pharmacy disciplines. From a humble beginning of one College with six departments, the University has now grown into the second largest University in Andhra Pradesh having five constituent Colleges Viz. College of Arts, College of Sciences, College of Commerce, Management & Computer Science, College of Pharmacy, and College of Engineering accommodating 52 departments offering 88 programs.

The University has made rapid strides in the field of higher education and research and is adjudged as one of the best Universities in the country and got ACCREDITED with 'A+' GRADE BY NAAC-2017.

The most commonly reported advantage of studying in Sri Venkateswara University are good teachers and "positive environment". The students are not only taught but given Hands-on training and career preparation. Sri Venkateswara University has been reviewed by Various National and international agencies and ranked as below.

<b>Various prestigious rankings and Academic credits of Sri Venkateswara University.

<b/p>

```
The Asia University Ranking -2022
                                    : 351-400
OS India Rankings-2020
                                       : 44
OS Asia Rankings-2021
                                       : 351-400
NIRF University category 2021
                                     : 54
THE Young University Rankings-2020
                                   : 101-150
Education world Rankings 2020
                                    : 23
THE world University Rankings-2021
                                   : 801-1000
The Asia University Ranking-2021
                                    : 251-300
```

Sri Venkateswara University is on top in research and innovation. The University has expertise and connections to help young researchers at every stage. From the inception of the University the faculty, students and research scholars, post-doctoral fellows have been extend the boundaries of knowledge which is evident from the

```
    Citations( Normalized) for 2014-2019 : 16,321
    Citations for 2017-2022 : 14,891
```

The Emerging Economies : 351-440
PhDs awarded : 217 FT + 129 PT
Patents filed : 4
Total research projects on going : 34
Total amount : Rs.53.14,94,271
Consultancy projects : 629
Money generated through consultancy : Rs. 766965/-

The University is also selected under RUSA component 4 for Rs.100 crore. Under this program Centres of Excellence in (i)Nano & Micro Satellite (ii) Earth & Atmospheric Sciences (iii)Material Sciences (iv)VLP Technologies (v) Herbal Drug Development (vi) Psycho & Bio Sciences (viii) Water Resources and (ix) Bio Energy, are established. University is having a National facility UGC-SVU center for MST Radar Applications selected under Centre for Potential of Excellence in a particular area (CPEPA) by

</div>
<div className="content\_\_right">

UGC, New Delhi and awarded with Centers for Advanced Study (UGC-CAS), Based on research output university was awarded DST-PURSE program.

Several departments have received Special Assistance Programmes (SAP) from UGC, New Delhi FIST program from DST, New Delhi to improve infrastructural facilities. Most of the faculties are receiving research grants from various National and International funding agencies like UGC, DST, DBT, CSIR, ICMR, ICSSR, BRNS, ISRO, MNES, MoES, MoEF, DRDO and foreign institutes such as UNESCO, UK-DFID, ICRISAT, European Commission Programme ERASMUS MUNDUS. Utilizing these impressive funding, the faculty of the University have proactively interacted with Industry, Academic and Research Institutes at National and International level and entered into collaborative research agreements through MoUs (Memorandum of Understating) to conduct research work in frontier areas of National & International importance.

The students belonging to socially and economically backward communities and get State and Central Government Welfare Scholarships. Fellowships and Scholarships are also offered to research students of the campus by the University and also by the UGC, CSIR, DAE, ICMR, ISRO, ICAR, ICPR, ICHR, ICSSR, DST, DRDO, SRNS and Rashtriya Sanskrit Sansthan.

The University central library is the major resource centre that provides information for research students and faculty. It provides unlimited access to the UGC INFONET e-resources and e-journals through UGC INFONET Digital Library Consortium. Besides the addition of latest books related to both teaching and research in the main library, most of the departments have developed departmental libraries by utilizing the grant from the University or from special assistance received from funding agencies like UGC, DST, etc. A Digital Library for PhD Theses submitted to the University has been designed using DSpace, an Open Source Software developed by HP and MIT, USA. 3,000 PhD Theses have been digitalized and uploaded to the Shodhganga database. This gives easy access to the PhD theses and research documents.

The University provides residential facilities to Graduate, Post Graduate Students research scholars, Post-Doctoral fellows and other researchers in the hostels specially earmarked for them on the campus. Summer fellows, research associates, visiting scientists and faculty from other universities and academics are also provided accommodation in the university guest house. The university campus is completely Wi-fi connected. The university provides computer and uninterrupted internet facilities for research students, fellows as well as to the faculty on the campus. Direct internet facility and common internet hubs are set-up near the hostels and quarters on the campus to provide internet access at free of cost.

To further enhance the status of the University and to compete with the top Universities at National and Global level through continued development during 2019-2020, additional Buildings for E-Class Rooms, Establishment of Incubation Centres, On-line Examination System/expansion of Laboratories to meet the requirements of new Courses are on the pipeline. Besides meeting the infrastructure facilities of Office, Cultural Centre and world Class Gymnasium and Sports facilities are available. Digital Library, Wi-Fi Connectivity, Networking, Smart Campus, Air-Conditioned Auditorium, Development of Greenery and State

```
of the modern art furniture in classrooms, faculty rooms, offices, Guest House and Auditorium are under
progress.
             >Besides strengthening the existing curriculum, new Courses are being planned under
MERU program.
          </div>
       </div>
        <section class="elementor-section elementor-top-section elementor-element elementor-element-
850e7da elementor-section-full width elementor-section-height-default elementor-section-height-default"
data-id="850e7da" data-element_type="section">
          <div class="elementor-container elementor-column-gap-default">
             <div class="elementor-row">
               <div class="elementor-column elementor-col-100 elementor-top-column elementor-
element elementor-element-f055ae8" data-id="f055ae8" data-element type="column">
                  <div class="elementor-column-wrap elementor-element-populated">
                    <div class="elementor-widget-wrap">
                       <div class="elementor-element elementor-element-09a8d60 elementor-widget"
elementor-widget-google_maps" data-id="09a8d60" data-element_type="widget" data-
widget type="google maps.default">
                          <div class="elementor-widget-container">
                            <div class="elementor-custom-embed">
                               <iframe loading="lazy"
src="https://maps.google.com/maps?q=13.6289%C2%B0%20N%2C%2079.3978%C2%B0%20E&t=
m&z=15&output=embed&iwloc=near" title="13.6289° N, 79.3978° E" aria-label="13.6289°
N, 79.3978° E" height="500px" width="1200px"></iframe>
                            </div>
                          </div>
                       </div>
                    </div>
                  </div>
               </div>
             </div>
          </div>
       </section>
     </div>
  )
}
```

### **Code for About.css:**

export default About

```
.ui {
   color: #002E3B;
   font-family: "Ubuntu", Sans-serif;
   font-size: 16px;
   font-weight: 400;
   margin-left: 20px;
}
.ui h2 {
   text-align: center;
   font-size: 30px;
```

```
margin-left: 220px;
  margin-top: 50px;
  margin-bottom: 50px;
}
.ui .content__grp {
   display: flex;
}
.ui .content__grp .content__left p{
  color: black;
  max-width: 600px;
  text-align: justify;
  line-height: 23px !important;
  display: block;
  margin-block-start: 1em;
  margin-block-end: 1em;
  margin-inline-start: 0px;
  margin-inline-end: 0px;
.ui .content__grp .content__right {
margin-left: 20px;
.ui .content__grp .content__right p{
  color: black;
  max-width: 600px;
  text-align: justify;
  line-height: 23px !important;
  display: block;
  margin-block-start: 1em;
  margin-block-end: 1em;
  margin-inline-start: 0px;
  margin-inline-end: 0px;
}
```

# **Code for Administration.js:**

```
import React, { useState } from 'react'
import './Administration.css'

function Administration() {
   const [about, setAbout] = useState(true);
   const [principal, setPrincipal] = useState(false);
   const [vice, setVice] = useState(false);
   const [depts, setDepts] = useState(false);

   const handleAbout = () => {
      setAbout(!about)
      setPrincipal(false)
      setVice(false)
      setDepts(false)
}
```

```
const handlePrincipal = () => {
     setAbout(false)
     setPrincipal(!principal)
     setVice(false)
     setDepts(false)
   }
  const handleVice = () => {
     setAbout(false)
     setPrincipal(false)
     setVice(!vice)
     setDepts(false)
   }
  const handleDepts = () => {
     setAbout(false)
     setPrincipal(false)
     setVice(false)
     setDepts(!depts)
  }
  return (
     <div className='main ui'>
        <div className="btns">
           <button className={`about ${about ? "active" : ""}`} onClick={handleAbout}
>About</button>
           <button className={`principal ${principal ? "active" : ""}`}</pre>
onClick={handlePrincipal}>Principal</button>
           <button className={`vice ${vice ? "active" : ""}`} onClick={handleVice}>Vice
principal</button>
           <button className={`depts ${depts ? "active" : ""}`}</pre>
onClick={handleDepts}>Departments</button>
        {about && <div className="respective">
           <h2>CMCS</h2>
           <img src={process.env.PUBLIC_URL + '/images/Department-of-Commerce-Management-</pre>
Computer-Science.jpg' alt="" />
research, extension and consultancy. It is a constituent campus college of Sri Venkateswara
```

Sri Venkateswara University College has been a premiere institution for higher learning, University.

The College is located serenely in a picturesque campus at the foot of the Seven Hills. With well planned and aesthetically designed buildings spread in a green, clean and expansive environment, the campus is a joyous place to work and live. Since its inception half century ago, the College has been successfully maintaining and meeting national and international standards in higher education.

```
1. COMMERCE <br></br>
```

Department of Commerce has excellence in commerce education is our mission and individual's holistic development and quality education in commerce to make effective contribution to the national economy in a dynamic environment. Making the students skillful, competitive, and communicative and grooming them to be fit for their placement in the production and service sectors in the global environment.

```
<b>COURSES OFFERED</b>
M. Com:
PhD (Full time / Part time)
 Accounting & Finance (A&F)
```

```
Financial Management (FM)2. BUSINESS MANAGEMENT <br></br>
```

Department of Management Studies makes every effort to dismantle the patterned mindsets and restructure the personality of the students. Great emphasis is laid on individual development and team work. The learning techniques like Role play, Case analysis, Group Discussions, Simulation exercises, Management Games, Business information sharing conferences, etc., are part of the pedagogy.

```
<b>COURSES OFFERED</b>

    PhD (Full time / Part time)
    MBA

3. COMPUTER SCIENCE <br></br>
```

Department of Computer Science has efficient and technical qualified faculty who continuously updating their knowledge as per requirement of present world technology and forwarding the fruitful knowledge to the students. The department is consistently achieving more than 99% results in all the courses offered. Along with education our students are participating in the development of projects as per the needs of various departments in the university and also department.

```
<b>COURSES OFFERED</b>
        PhD (Full time / Part time)
          MPhil
          MCA
          MSc
        </div>}
      {principal && <div className="principalContent">
        <div className="left">
          <imq src={process.env.PUBLIC_URL + '/images/PV-Narsaiah.png'} alt="" />
          <div className="leftContent">
            <h3><b>Dr. P.V. Narasaiah</b></h3>
            Principal, SVU CM & CS & Professor,<br></br> Department of
Commerce
            M.Com, M.Phil, Ph.D.
            Email
            drpvnarasaiah@gmail.com
          </div>
        </div>
        <div className="right">
```

<b>Prof. P.V. Narasaiah</b> did his M.Com, M.Phil, and Ph.D. in 1985, 1988 and 1990 respectively. He got first rank in M.Com. He joined S.V University Service as Lecturer at SVU PG Center, Kadapa during 1992 and promoted as Associate Professor by 2001. During 2007, he was repatriated to S.V University Campus, Tirupati when SVU PG Center was elevated to Y.V University. He became Professor during the year 2009. He has to his credit 92 articles published in reputed National and International Journals and 9 books. He successfully guided 11 Ph.Ds and 3 M.Phils. He has completed 2 ICSSR and UGC research projects. He has attended and presented 127 papers in different National and International Seminars. He has organized One National Seminar and acted as Deputy Director for International Conference, Co-Director for 3 National Seminars. He delivered 12 AIR Talks on different contemporary issues, more particularly relating to his area of research.

He is a life member, ICA, AIAA, Red Cross Society (Hyderabad Chapter). He is a editorial

member for i-manager and International Journal of Business Research (MITTS), Madanapalle. He acted as PRO, Co-ordinator for different courses and Deputy Warden at Y.V Universtiy. He was the Co-ordinator for HPIC for University examinations. After joining SVU, he served as Co-ordinator for M.Com (A&F) and MFM Courses, Deputy Warden for SVU CM&CS Hostel for Men. He was also member of BOS, SVU. At present, he is a member of BOS, Y.V University, Sri Padmavathi Mahila Visva Vidyalayam, Rayalaseema University. He was a member in the panel for paper setting relating to APSET. He is a resource person to many Universities.

```
He is currently the Principal, SVU College of Commerce, Management and Computer
Science.
           <h3>Dr. P.V. Narasaiah</h3>
           Principal, SVU CM & CS & Professor, Department of Commerce
           M.Com, M.Phil, Ph.D.
         </div>
       </div>}
      {vice && <div className="principalContent">
         <div className="left">
           <img src={process.env.PUBLIC_URL + '/images/M-Venkateswarlu.png'} alt="" />
           <div className="leftContent">
             <h3><b>Dr. M. Venkateswarlu</b></h3>
             Professor & Vice-Principal
             M.Com, MBA, M.Phil, Ph.D., PGDCA
             Email
             drmidasala@gmail.com
           </div>
         </div>
         <div className="right">
```

<b>Prof. M.Venkateswarlu</b> was born in Karavadi Village, Prakasam District. He obtained B.A. (AES) Degree in 1985, M.Com. Degree in 1987, M.Phil. Degree in the year 1990 and Ph.D. degree from S.V. University in the year 1994. He was appointed as Lecturer in S.V.University, Tirupati in October 1992, promoted as Associate Professor during 2001 and subsequently he became Professor during 2009. He got his PGDCA and MBA Degrees during the years 2011, 2013 respectively. He is having 28 years of P.G. teaching experience, 7 M.Phil. Degrees and 24 Ph.D. degrees were awarded and 9 post doctoral fellows under his able guidance. He has published 142 articles in various repeated National and International Journals. He has attended 53 National and International conferences and also attended 6 online webinars.

He has completed one UGC MRP major research project and presently he is having ICSSR research project. He has organized two national seminars; he worked as Co-ordinator for M.Com A&F Course. He is a Member of Board of Studies in Davangere University and Bangalore North University. He is editorial board member for SELP Journal in Tiruchirapalli. He is life member in various professional bodies like All India Commerce Association, All India Accounting Association, and Business Vision. He worked as Director, Directorate of Admissions (DOA), Sri Venkateswara University, Tirupati from 01-01-2020 to 07-05-2021. At present he is Vice-Principal, SVU CM & CS, Sri Venkateswara University, Tirupati.

He is currently the Principal, SVU College of Commerce, Management and Computer

export default Administration

## **Code for Administration.css:**

```
.main__ui {
  display: flex;
.btns {
  display: flex;
  flex-direction: column;
  padding: 50px;
}
.btns .active{
  background-color: orange;
.btns button {
  width: 300px;
  margin-top: 5px;
  border: none;
  height: 40px;
}
.btns button:hover {
  background-color: orange;
}
p{
  color: black;
}
.main__ui .respective h2 {
  font-size: 25px;
  margin-top: 30px;
}
.main__ui .respective img {
  height: 300px;
  width: 800px;
  object-fit: contain;
}
.main__ui .principalContent {
  margin-top: 30px;
```

```
display: flex;
  flex-direction: row;
.main__ui .principalContent .left .leftContent{
   padding: 20px;
background-color: rgb(230, 223, 223);
width: 330px;
.main__ui .principalContent .left .role{
  font-size: 15px;
.main__ui .principalContent .right p{
  margin-left: -20px;
  width: 450px;
  text-align: justify;
}
.main__ui .deptss{
  margin-top: 50px;
.main__ui .deptss .btn {
  padding: 20px;
  width: 200px;
  height: 150px;
  background-color: blue;
  text-align: center;
  align-items: center;
}
.main__ui .deptss .btn button {
  border: none;
  background-color: blue;
  height: 100px;
  width: 150px;
  text-align: center;
  color: white;
  font-weight: 800;
  font-size: 18px;
}
.main__ui .deptss .btn button:hover {
  background-color: rgb(98, 98, 158);
  border-radius: 10px;
}
```

# **Code for CampusTour.js:**

import React from 'react'

# **Code for ContactUs.js:**

```
import React from 'react'
import "./ContactUs.css"
import FacebookIcon from '@mui/icons-material/Facebook';
import InstagramIcon from '@mui/icons-material/Instagram';
import LinkedInIcon from '@mui/icons-material/LinkedIn';
import TwitterIcon from '@mui/icons-material/Twitter';
function ContactUs() {
  return (
     <div className='hey'>
        <div className="contact_content">
           <h2>Contact US</h2>
           <div className="address">
             <span class="material-symbols-outlined">
                location on
             </span>
             <h4>Sri Venkateswara University, Tirupati, Andhra Pradesh - 517502</h4>
           <button><a href="https://goo.ql/maps/wrdsiNe4me6xYdzdA">View on maps</a> </button>
           <div className="address">
             <span class="material-symbols-outlined">
                call
             </span>
             <h4>+91 (877) 2249472</h4>
           </div>
           <div className="address">
             <span class="material-symbols-outlined">
                mail
             </span>
             <h4>registrarsvu@gmail.com</h4>
           <div className="footer__icons">
             <FacebookIcon className='facebook' />
             <InstagramIcon className='instagram' />
             <LinkedInIcon className='linkdin' />
             <TwitterIcon className='twitter' />
           </div>
        </div>
     </div>
  )
}
```

export default ContactUs

## **Code for ContactUs.css:**

```
.hey .contact_content {
  margin-top: 50px;
.hey .contact_content h2 {
  margin-bottom: 30px;
  margin-left: 60px;
.hey .contact_content .address {
  display: flex;
  margin-left: 60px;
  align-items: center;
}
.hey .contact_content .address span{
   color: green;
.hey .contact_content .address h4 {
  margin-top: 10px;
  margin-left: 20px;
  font-size: 15px;
}
.hey .contact_content button {
  border: none;
  background-color: rgb(225, 194, 21);
  height: 40px;
  width: 150px;
  border-radius: 20px;
  margin-left: 385px;
}
.hey .contact_content button a {
  text-decoration: none;
  color: black;
}
.footer__icons {
  margin-top: 50px;
  margin-left: 60px;
}
```

# **Code for Departments.js:**

```
import React, { useContext } from 'react'
import { Link } from 'react-router-dom'
import './Departments.css'
```

```
import { DeptContext } from './DeptContext'
function Departments() {
  const { dept, setDept } = useContext(DeptContext)
  const handleBusiness = () => {
     setDept("Business Management")
  }
  const handleCS = () => {
     setDept("Computer Science")
  const handleCM = () => {
     setDept("Department of commerce")
  }
  return (
     <div className='departments'>
        <h4>Select your department</h4>
        <div className="btnns">
          <div className="btnn">
             <Link to='/student'><button onClick={handleBusiness}>Business
Management</button></Link>
          </div>
          <div className="btnn">
             <Link to='/student'><button onClick={handleCS}>Computer Science</button></Link>
          <div className="btnn">
             <Link to='/student'><button onClick={handleCM}>Department of
commerce</button></Link>
          </div>
        </div>
     </div>
  )
}
export default Departments
```

## **Code for Departments.css:**

```
.departments {
    padding: 100px;
}
.departments h4{
    text-align: center;
    font-weight: 800;
    font-size: 30px;
}
.departments .btnns {
    margin-top: 30px;
    display: flex;
}
.departments .btnns .btnn{
```

```
padding: 30px;
width: 1100px;
background-color: blueviolet;
}
.departments .btnns .btnn button{
  padding: 30px;
  border: none;
  color: #fff;
  height: 150px;
  background-color: rgb(193, 127, 255);
  border-radius: 15px;
}
.departments .btnns .btnn button:hover{
  background-color: rgb(217, 181, 251);
}
```

# **Code for DepartmentContext.js:**

## **Code for Notifications.js:**

```
<a href="https://svuniversity.edu.in/notification/aplawcet-2022-spot-
admissions">APLAWCET 2022 - SPOT ADMISSIONS</a>
          <a href="https://svuniversity.edu.in/notification/executive-quota-information-and-
extramural-category-information">The extended date for submission of application for Executive and
Extramural guota is extended upto 31-10-2022</a>
          3-8-2022
            <a href="https://svuniversity.edu.in/notification/svu-dde-admission-notification-for-
the-academic-session-2022-23">SVU DDE ADMISSION NOTIFICATION FOR THE ACADEMIC SESSION -
2022-23</a>
          18-6-2022
             <a href="https://svuniversity.edu.in/notification/aprcet-2021-ph-d-vaccancies-for-
spot-admissions-in-sv-universitytirupati">APRCET-2021 :: Ph.D VACCANCIES FOR SPOT ADMISSIONS IN SV
UNIVERSITY,TIRUPATI</a>
          May 2022
            <a href="https://svuniversity.edu.in/notification/i-sem-bcom-may2022-time-table">I
SEM BCOM MAY,2022 time Table</a>
          May 2022
             aprmay2022">Degree III sem Timetable (REG & Supply) exam Timetable May'2022</a>
          May 2022
             <a href="https://svuniversity.edu.in/notification/ug-i-sem-exam-
aprilmay2022">Degree I sem Timetable (REG & Supply) exam Timetable May'2022</a>
          17-03-2022
             <a href="https://svuniversity.edu.in/notification/pg-special-fee-structure-category-
seats">PG SPECIAL FEE STRUCTURE CATEGORY SEATS Extends Till 29-3-2022</a>
          </div>
  )
}
```

export default Notofications

## **Code for Notifications.css:**

```
.notifs {
   padding-left: 150px;
   padding-right: 150px;
   padding-top: 50px;
}
.notifs table thead {
   height: 50px;
   background-color: rgb(207, 209, 210);
}
.notifs table tbody tr{
   border: 1px solid rgb(205, 202, 202);
}
.notifs table tbody tr td{
   border: 1px solid rgb(205, 202, 202);;
}
```

# **Code for OurPredecessors.js:**

```
import React from 'react'
import './OurPredecessors.css'
function OurPredecessors() {
  return (
    <div className='predecessors'>
      <h1>SUCCESSIVE VICE-CHANCELLORS</h1>
       <table border="1"
         cellpadding="10"
         cellspacing="5">
         <h3>Prof. S. Govindarajulu, M.A.(Cantab)</h3>
             1954-64
           <h3>Prof. V.C. Vaman Rao, M.A., Ph.D.(Dublin)</h3>
             1964-69
           <h3>Prof. D. Jagannadha Reddy, M.D., M.G.(Pathology)</h3>
             1969-75
           <h3>Prof. K. Satchidanandamurthy, M.A., Ph.D.</h3>
```

```
1975-78
 <h3>Prof. M. Santappa, M.A., Ph.D. (London), Ph.D.(Manchester)</h3>
   1978-80
 <h3>Prof. M.V. Rama Sarma, M.A., Ph.D. (Wales)</h3>
   1980-84
 <h3>Prof. R. Ramamurthi, M.A.,M.Sc.,F.N.A., F.N.A.Sc.</h3>
   1994-97
 <h3>Prof. K. Enoch, M.A., Ph.D. </h3>
   1998-01
 <h3>Prof. P.Murali, M.Com., Ph.D.</h3>
   2001-04
 <h3>Prof. S. Jayarama Reddy, M.Sc., Ph.D., Dip.in German</h3>
   2004-07
 <h3>Prof. C. Ratnam, M.A.,Ph.D.</h3>
   2008
 <h3>Prof. N. Prabhakara Rao, M.Sc.,Ph.D.</h3>
   2008-11
 <h3>Prof. W. Rajendra, M.Sc.,Ph.D.</h3>
   2012-15
 <h3>Prof. Avula Damodaram, Ph.D.</h3>
   2015-18
 <h3>Prof. G. Janaki Ramaiah I/c, M.A.,Ph.D.</h3>
   2018-19
```

```
<h3>Prof. VVN Rajendra Prasad, M.A.,Ph.D.</h3>
           3-2-2019 to 19-06-2019
         <h3>Sri J.S.V. Prasad, IAS, (I/c Vice-Chancellor )</h3>
           19-6-2019 to 12-11-2019
         <h3>Sri Sathish Chandra, IAS, (I/c Vice-Chancellor)</h3>
           12-11-2019 to 25-11-2020
         <h3>Prof. K. Raja Reddy, M.A.,Ph.D.</h3>
           26-11-2020 to till date
         <h3></h3>
           <h3></h3>
           </div >
  )
}
```

export default OurPredecessors

# **Code for OurPredecessors.css:**

```
.predecessors {
    padding: 50px;
}
.predecessors h1{
    color: rgb(244, 194, 29);
    text-align: center;
}
.predecessors table tr td {
    border: 1px solid black;
}
.predecessors table tr td {
    margin-left: 5px;
    background-color: aqua;
}
```

```
.predecessors table tr td h3{
  font-size: 20px;
}
```

# **Code for Privacy.js:**

We shall collect personal information by lawful & fair means and wherever appropriate with the knowledge or consent of the individual concerned.

Before or at the time of collecting personal information, we will identify the purposes for which information is being collected.

We shall collect and use personal information solely for fulfilling those purposes specified by us and for other ancillary purposes, unless we obtain the consent of the individual concerned or as required by law.

Personal data should be relevant to the purposes for which it is to be used, and to the extent necessary for those purposes, should be accurate, complete, and up-to- date.

We shall protect personal information by using reasonable security safeguards against loss or theft, as well as unauthorized access, disclosure, copying, use or modification.

We shall make readily available to customers information about our policies and practices relating to the management of personal information.

<Ii>We shall only retain personal information for as long as necessary for the fulfillment of those purposes. We are committed to conducting our business in accordance with these principles in order to ensure that the confidentiality of personal information is protected and maintained. Sri Venkateswara Universitymay change this privacy policy from time to time at Sri Venkateswara University

```
</div>
)
```

export default Privacy

# Code for Term.js:

Welcome to our website Sri Venkateswara University. This agreement contains the complete terms and conditions that apply when you visit our site or use choose to browse the contents herein. By using this website, you agree to be bound by its terms of use and shall comply thereof. This agreement describes and encompasses the entire agreement between us and you, and supersedes all prior or contemporaneous agreements, representations, warranties and understandings with respect to the Site and the content provided by or through the Site, and the subject matter of this agreement.

<b>EDITING, DELETING AND MODIFICATION</b>

We may edit, delete or modify any of the terms and conditions contained in this agreement, at any time and in our sole discretion, by posting a notice or a new agreement on our site. Your continued visit in our website following our posting of a change notice or new agreement on our site will constitute binding acceptance of the change.

<b>ACCURACY, COMPLETENESS AND TIMELINESS OF INFORMATION</b>

Sri Venkateswara University(https://svuniversity.edu.in/) is not responsible, if information made available on this site is not accurate, complete or current. The material on this site is provided for general information only and should not be relied upon or used as the sole basis for making decisions without consulting primary, more accurate, more complete or timely sources of information. Any reliance on the material on this site is at your own risk. This site may contain certain historical information which is definitely not current and is provided for reference only. We reserve the right to modify the contents of this site at any time without obligation on our end to update any information on the site. You understand and agree that it is your responsibility to monitor changes to the site.

<b>LICENSE AND SITE ACCESS</b>

<b>Venkateswara University</b> (https://svuniversity.edu.in/) grants you a limited license to access and make personal use of this site and not to download or modify it, or any portion of it, except with express written consent from our company. This site or any portion of this site may not be reproduced, duplicated, copied, sold, resold, visited, or otherwise exploited for any commercial purpose without express written consent from our company.

The right to access our corporate website does not include any resale or commercial use of our site or its contents nor allows you to download or copy any account information for the benefit of another merchant. Any unauthorized use terminates the permission or license granted by Venkateswara University (https://svuniversity.edu.in/)

<b>LINKS AND FRAMINGS</b>

You shall not use our company logo or other proprietary graphic to link to this Site without the express written permission of Sri Venkateswara University (https://svuniversity.edu.in/). Further, you may not frame any trademark, logo or other proprietary information, including the Images Content, without our express written consent.

We make no claim or representation regarding, and accepts no responsibility for, directly or indirectly, the quality, content, nature or reliability of third-party websites accessible by hyperlink from the Site, or websites linking to the Site. Such sites are not under the control of our company and we shall not responsible for the contents of any linked site or any link contained in a linked site, or any review, changes or updates to such sites. You hereby acknowledge and agree that we are not responsible for the availability of such external websites or resources, its content, including, without limitation, any link contained in such content, or any changes or updates to such content. If you decide to access links to third-party Web sites,

you do so at your own risk. Any concerns regarding any external link should be directed to the relevant site administrator or web master.

<b>DISCLAIMERS</b>

Your use of this site is at your sole risk. The site is provided on an "as is" and "as available" basis. We reserve the right to restrict or terminate your access to the site or any feature or part thereof at any time. Venkateswara University (https://svuniversity.edu.in/) expressly disclaims all warranties of any kind, whether express or implied, including but not limited to the implied warranties of merchantability and fitness for a particular purpose and any warranties that materials on the site are no infringing, as well as warranties implied from a course of performance or course of dealing; that access to the site will be uninterrupted or error-free; that the site will be secure; that the site or the server that makes the site available will be virus-free; or that information on the site will be complete, accurate or timely. If you download any materials from this site, you do so at your own discretion and risk. You will be solely responsible for any damage to your computer system or loss of data that results from the download of any such materials. No advice or information, whether oral or written, obtained by you from Venkateswara University or through or from the site shall create any warranty of any kind. Venkateswara University does not make any warranties or representations regarding the use of the materials on this site in terms of their completeness, correctness, accuracy, adequacy, usefulness, timeliness, reliability or otherwise.

<b>CORPORATE INTELLECTUAL PROPERTY RIGHTS</b>

You hereby acknowledge that all rights, titles and interests, including but not limited to rights covered by the Intellectual Property Rights, in and to the site, and that You will not acquire any right, title, or interest in or to the site except as expressly set forth in this agreement. You will not modify, adapt, translate, prepare derivative works from, decompile, reverse engineer, disassemble or otherwise attempt to derive source code from any of our services, software, or documentation, or create or attempt to create a substitute or similar service or product through use of or access to the Program or proprietary information related thereto.

<b>CONFIDENTIALITY</b>

You agree not to disclose information you obtain from us and or from our clients, advertisers, suppliers and forum members. All information submitted to by an end-user customer pursuant to a Program is proprietary information of Venkateswara University (https://svuniversity.edu.in/). Such customer information is confidential and may not be disclosed. Publisher agrees not to reproduce, disseminate, sell, distribute or commercially exploit any such proprietary information in any manner.

<b>NON-ASSIGNMENT OF RIGHTS</b>

Your rights of whatever nature cannot be assigned nor transferred to anybody, and any such attempt may result in termination of this agreement, without liability to us. However, we may assign this agreement to any person at any time without notice.

<b>NON-WAIVER</b>

Failure of the Sri Venkateswara University (https://svuniversity.edu.in/) to insist upon strict performance of any of the terms, conditions and covenants hereof shall not be deemed a relinquishment or waiver of any rights or remedy that the we may have, nor shall it be construed as a waiver of any subsequent breach of the terms, conditions or covenants hereof, which terms, conditions and covenants shall continue to be in full force and effect.

No waiver by either party of any breach of any provision hereof shall be deemed a waiver of any subsequent or prior breach of the same or any other provision.

<b>SEVERABILITY OF TERMS</b>

In the event that any provision of these Terms and Conditions is found invalid or unenforceable pursuant to any judicial decree or decision, such provision shall be deemed to apply only to the maximum extent permitted by law, and the remainder of these Terms and Conditions shall remain valid and enforceable according to its terms.

<b>ENTIRE AGREEMENT</b>

With out any reference to conflict-of-laws principles. The agreement describes and encompasses the entire agreement between us and you, and supersedes all prior or contemporaneous agreements, representations, warranties and understandings with respect to the Site, the contents and materials provided by or through the Site, and the subject matter of this agreement.

<b>GOVERNING LAW</b>

Any dispute, controversy or difference which may arise between the parties out of, in relation to or in connection with this agreement is hereby irrevocably submitted to the exclusive jurisdiction of the courts of Sri Venkateswara University,

```
<b>Sri Venkateswara University</b>
Tirupati, Andhra Pradesh 517502
>0123456789
>svu@gmail.com
</div>
)
}
export default Term
```

# **Code for UserContext.js:**

# Code for Vision.js:

```
<div className="visionPart">
            <img src={process.env.PUBLIC_URL + "/images/vision.png"} alt="" />
            <h3>Our Vision</h3>
             To be a centre of excellence in Education and Technology committed towards Socio-
Economic advancement of the country.
            </div>
          <div className="misionPart">
            <img src={process.env.PUBLIC_URL + "/images/mission.png"} alt="" />
            <h3>Our Mission</h3>
            ul>
               To impart advanced knowledge in all branches of teaching and learning. • To
transform young minds towards professional competence by inculcating values and developing skills.
               To promote research and ensure continuous value addition among students and
employees.
               To strengthen association with industry, research organizations and alumni to enhance
knowledge on current technologies.
               To promote next generation technocracy and nurture entrepreneurial culture for
social-economic
                 growth.
            </div>
       </div>
     </div>
  )
}
```

## **Code for Vision.css:**

export default Vision

```
.vision {
    margin-top: 50px;
}
.vision .contentt {
    padding: 40px;
    display: flex;
}
.vision .contentt .visionPart{
    padding: 20px;
    background-color: rgb(222, 214, 214);
}
.vision .contentt .visionPart ul li {
    text-align: justify;
}
.vision .contentt .misionPart{
    padding: 20px;
    margin-left: 20px;
```

```
background-color: rgb(222, 214, 214);
}
.vision .contentt .misionPart ul li {
    text-align: justify;
}
.vision .contentt .visionPart img{
    max-height: 100px;
}
.vision .contentt .misionPart img{
    max-height: 100px;
}
```

# **Code for Footer.js:**

```
import React from 'react'
import './Footer.css'
import FacebookIcon from '@mui/icons-material/Facebook';
import InstagramIcon from '@mui/icons-material/Instagram';
import LinkedInIcon from '@mui/icons-material/LinkedIn';
import TwitterIcon from '@mui/icons-material/Twitter';
import { Link } from 'react-router-dom';
function Footer() {
  const scrollToContent = (id) => {
     const element = document.getElementById(id);
     if (element) {
        element.scrollIntoView({ behavior: 'smooth' });
     }
  };
  return (
     <div className="footer">
        <div className="footer__contents">
           <div className="footer content-left">
             <img className="logo" src={process.env.PUBLIC_URL + '/images/SV-logo.png'} alt="" />
             <div className="footer__cotent-left__options">
                <div className="footer cotent-left option">
                   <Link to="/home" onClick={() => scrollToContent('home')}> <a
href="/">Home</a></Link>
                  <Link to="/about" onClick={() => scrollToContent('about')}> <a href="/">About
Us</a></Link>
                  <Link to="/vision"><a href="/">Vision & Mission</a></Link>
                   <Link to="/vision"><a href="/">Administration</a></Link>
                <div className="footer cotent-left option2">
                   <a href={process.env.PUBLIC_URL + "/images/academic-calendar.pdf"}
target="_blank">Acadamic Calender</a>
                   <Link to="/notifications"><a href="/">Notifications</a></Link>
                   <Link to="/campustour"><a href="/">Campus Tour</a></Link>
                  <Link to="/privacy"><a href="/">Privacy Policy</a></Link>
                   <Link to="/term"><a href="/">Terms and Conditions</a></Link>
                </div>
```

```
</div>
          </div>
           <div className="hey">
             <div className="contact_content">
                <h2 className='contact'>Contact US</h2>
                <div className="address">
                  <span class="material-symbols-outlined">
                     location_on
                  </span>
                  <h4>Sri Venkateswara University, Tirupati, Andhra Pradesh - 517502</h4>
                <button><a href="https://goo.gl/maps/wrdsiNe4me6xYdzdA">View on maps</a>
</button>
                <div className="address">
                  <span class="material-symbols-outlined">
                     call
                  </span>
                  <h4>+91 (877) 2249472</h4>
                </div>
                <div className="address">
                  <span class="material-symbols-outlined">
                     mail
                  </span>
                  <h4>registrarsvu@gmail.com</h4>
                </div>
                <div className="footer__icons">
                  <FacebookIcon className='facebook' />
                  <InstagramIcon className='instagram' />
                  <LinkedInIcon className='linkdin' />
                  <TwitterIcon className='twitter' />
                </div>
             </div>
          </div>
        </div>
     </div >
  )
}
export default Footer
```

## **Code for Footer.css:**

```
.footer {
    margin-top: 100px;
    height: 400px;
    width: 100%;
    background-color: #E9F2FE;
    font-family: Arial, Helvetica, sans-serif;
}
.footer .footer__contents {
    display: flex;
```

```
/* justify-content: space-between; */
  position: relative;
  margin-top: 50px;
}
.footer contents .footer content-left {
  display: flex;
  flex-direction: column;
  justify-content: space-between;
  margin-left: 100px;
  margin-top: 40px;
  width: 800px;
}
.footer__contents .footer__content-left .logo {
  width: 250px;
  object-fit: contain;
.footer__cotent-left__options {
  margin-top: 20px;
  display: flex;
  justify-content: space-around;
}
.footer__cotent-left__options a {
  text-decoration: none;
  color: black;
  margin: 10px;
}
.footer__cotent-left__options .footer__cotent-left__option {
  /* padding: 20px; */
  /* margin: 50px; */
  display: flex;
  flex-direction: column;
}
.footer cotent-left options .footer cotent-left option1 {
  margin-left: 100px;
  /* margin: 20px; */
  display: flex;
  flex-direction: column;
.footer__cotent-left__options .footer__cotent-left__option2 {
   margin-left: 100px;
  /* margin: 20px; */
  display: flex;
  flex-direction: column;
}
.footer__content-middle {
  margin-left: 150px;
  margin-top: 40px;
.footer__content-middle a {
  text-decoration: none;
  color: black;
.footer__content-middle .top__insurances {
```

```
margin-top: 20px;
  /* margin: 10px; */
  display: flex;
  flex-direction: column;
}
.footer__content-middle .top__insurances:nth-child(1) {
  margin-top: 50px;
.footer__content-middle h2 {
  margin-left: -10px;
  font-weight: 700;
  font-size: 20px;
}
.footer__content-right {
  margin-left: 50px;
  margin-top: 40px;
}
.footer__content-right h2 {
  margin-left: -10px;
  font-weight: 700;
  font-size: 16px;
.footer__icons .facebook{
  margin: 3px;
  color: blue;
}
.footer__icons .instagram{
  margin: 3px;
  color: red;
}
.footer__icons .linkdin{
  margin: 3px;
  color: rgb(125, 125, 220);
.footer__icons .twitter{
  margin: 3px;
   color: skyblue;
}
.hey .contact_content {
  margin-top: 50px;
}
.hey .contact content h2 {
  margin-bottom: 30px;
  margin-left: 60px;
}
.hey .contact_content .address {
  display: flex;
  margin-left: 60px;
  align-items: center;
}
.hey .contact_content .address span{
  color: green;
}
```

```
.hey .contact content .address h4 {
  margin-top: 10px;
  margin-left: 20px;
  font-size: 15px;
}
.hey .contact_content button {
  border: none;
  background-color: rgb(225, 194, 21);
  height: 40px;
  width: 150px;
  border-radius: 20px;
  margin-left: 385px;
}
.hey .contact content button a {
  text-decoration: none;
   color: black;
}
.footer icons {
  margin-top: 50px;
   margin-left: 60px;
}
.contact {
  font-size: 25px;
```

# Code for App.js:

```
import './App.css';
import { BrowserRouter as Router, Route, Routes } from 'react-router-dom';
import AdminGetInPage from './components/AdminGetInPage';
import StudentGetInPage from './components/StudentGetInPage';
import InitialPage from './components/InitialPage';
import AdminSignUp from './components/AdminSignUp';
import StudentSignup from './components/StudentSignup';
import StudentHome from './components/StudentHome';
import UpdateStudent from './components/UpdateStudent';
import AdminHome from './components/AdminHome';
import NavBar from './components/NavBar';
import Footer from './components/Footer';
import About from './components/About';
import { FormContextProvider } from './components/FormContext';
import { ThemeProvider } from './components/ThemeContext';
import { UserProvider } from './components/UserContext';
import ContactUs from './components/ContactUs';
import CampusTour from './components/CampusTour';
import Vision from './components/Vision';
import Administration from './components/Administration';
import OurPredecessors from './components/OurPredecessors';
import Privacy from './components/Privacy';
```

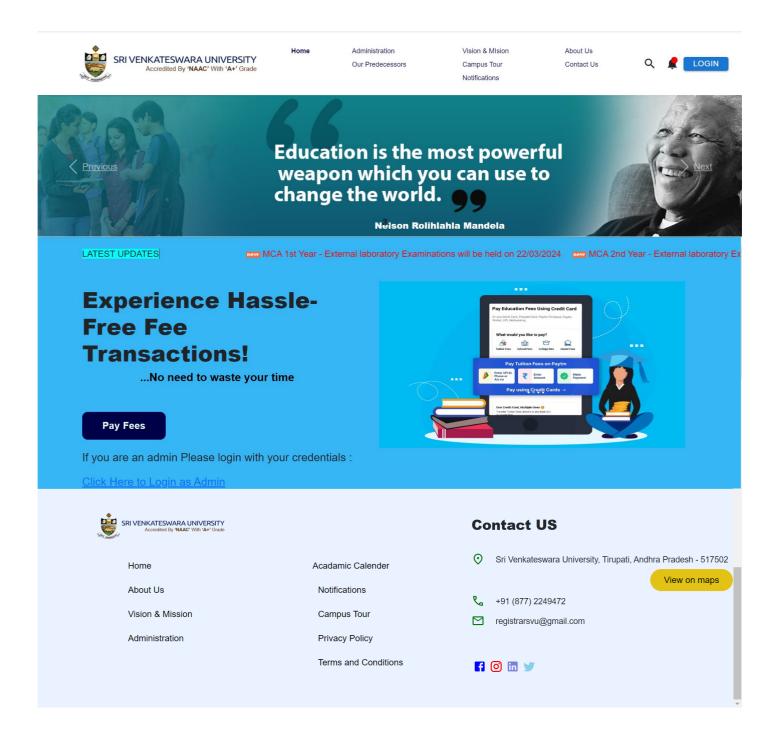
```
import Term from './components/Term';
import Notofications from './components/Notofications';
import Departments from './components/Departments';
import { DeptProvider } from './components/DeptContext';
// import { createBrowserHistory } from 'history';
function App() {
 // const history = createBrowserHistory();
 return (
  <div className="App">
    <DeptProvider>
     <UserProvider>
      <FormContextProvider>
        <NavBar />
        <Routes>
         <Route exact path="/" element={<InitialPage />} />
         <Route path="/admin" element={<AdminGetInPage />} />
         <Route path="/student" element={<StudentGetInPage />} />
         <Route path="/adminSignup" element={<AdminSignUp />} />
         <Route path="/studentSignup" element={<StudentSignup />} />
         <Route path="/studentHome" element={<StudentHome />} />
         <Route path="/updateStudent" element={<UpdateStudent />} />
         <Route path="/adminHome" element={<AdminHome />} />
         <Route path="/about" element={<About />} />
         <Route path="/contact" element={<ContactUs />} />
         <Route path="/campustour" element={<CampusTour />} />
         <Route path="/vision" element={<Vision />} />
         <Route path="/administration" element={<Administration />} />
         <Route path="/ourpredecessors" element={<OurPredecessors />} />
         <Route path="/privacy" element={<Privacy />} />
         <Route path="/term" element={<Term />} />
         <Route path="/notifications" element={<Notofications />} />
         <Route path="/departments" element={<Departments />} />
        </Routes>
        <Footer className="footer" />
      </FormContextProvider>
     </UserProvider>
    </DeptProvider>
  </div>
 );
export default App;
```

# Code for package.json:

```
"name": "studentfeesmanagement",
"version": "0.1.0",
"private": true,
"dependencies": {
 "@emotion/react": "^11.11.4",
 "@emotion/styled": "^11.11.0",
 "@mui/icons-material": "^5.15.12",
 "@mui/material": "^5.15.12",
 "@progress/kendo-react-animation": "^7.1.0",
 "@progress/kendo-react-notification": "^7.1.0",
 "@testing-library/jest-dom": "^5.17.0",
 "@testing-library/react": "^13.4.0",
 "@testing-library/user-event": "^13.5.0",
 "axios": "^1.6.7",
 "bootstrap": "^5.3.2",
 "history": "^5.3.0",
 "jspdf": "^2.5.1",
 "react": "^18.2.0",
 "react-dom": "^18.2.0",
 "react-router-dom": "^6.22.0",
 "react-scripts": "5.0.1",
 "web-vitals": "^2.1.4"
},
"scripts": {
 "start": "react-scripts start",
 "build": "react-scripts build",
 "test": "react-scripts test",
 "eject": "react-scripts eject"
"eslintConfig": {
 "extends": [
  "react-app",
   "react-app/jest"
"browserslist": {
 "production": [
   ">0.2%",
  "not dead",
  "not op mini all"
 "development": [
  "last 1 chrome version",
   "last 1 firefox version",
   "last 1 safari version"
}}
```

## 4.2 Output Screens

Fig.4.2.1 Home page



## Fig.4.2.2 Administration Page



ome

Administration
Our Predecessors

Vision & MIsion
Campus Tour
Notifications

About Us Contact Us

Q



### About

Principal

Vice principal

Departments

#### **CMCS**



Sri Venkateswara University College has been a premiere institution for higher learning, research, extension and consultancy. It is a constituent campus college of Sri Venkateswara University.

The College is located serenely in a picturesque campus at the foot of the Seven Hills. With well planned and

About

Principal

Vice principal

Departments



Dr. P.V. Narasaiah
Principal, SVU CM & CS & Professor,

About

Principal

Vice principal

Departments



Dr. M. Venkateswarlu
Professor & Vice-Principal

Prof. P.V. Narasalah did his M.Com, M.Phil, and Ph.D. in 1985, 1988 and 1990 respectively. He got first rank in M.Com. He joined S.V University Service as Lecturer at SVU PG Center, Kadapa during 1992 and promoted as Associate Professor by 2001. During 2007, he was repatriated to S.V University Campus, Tirupati when SVU PG Center was elevated to Y.V University. He became Professor during the year 2009. He has to his credit 92 articles published in reputed National and International Journals and 9 books. He successfully guided 11 Ph.Ds and 3 M.Phils. He has completed 2 ICSSR and UGC research projects. He has attended and presented 127 papers in different National and International Seminars. He has organized One National Seminar and acted as Deputy Director for International Conference , Co-Director for 3 National Seminars. He delivered 12 AIR Talks on different contemporary issues, more particularly relating to his area of research.

He is a life member, ICA, AIAA, Red Cross Society (Hyderabad Chapter). He is a editorial member for i-manager and Prof. M.Venkateswarlu was born in Karavadi Village, Prakasam District. He obtained B.A. (AES) Degree in 1985, M.Com. Degree in 1987, M.Phil. Degree in the year 1990 and Ph.D. degree from S.V. University in the year 1994. He was appointed as Lecturer in S.V.University, Tirupati in October 1992, promoted as Associate Professor during 2001 and subsequently he became Professor during 2009. He got his PGDCA and MBA Degrees during the years 2011, 2013 respectively. He is having 28 years of P.G. teaching experience, 7 M.Phil. Degrees and 24 Ph.D. degrees were awarded and 9 post doctoral fellows under his able guidance. He has published 142 articles in various repeated National and International Journals. He has attended 53 National and International conferences and also attended 6 online webinars.

He has completed one UGC MRP major research project and presently he is having ICSSR research project. He has organized two national seminars; he worked as Co-ordinator for M.Com A&F Course. He is a Member of Board of Studies in

Fig.4.2.3 Vision & Mission

## **Vision & Mission**



#### Our Vision

• To be a centre of excellence in Education and Technology committed towards Socio-Economic advancement of the country.



#### Our Mission

- To impart advanced knowledge in all branches of teaching and learning.
   To transform young minds towards professional competence by inculcating values and developing skills.
- To promote research and ensure continuous value addition among students and employees.
- To strengthen association with industry, research organizations and alumni to enhance knowledge on current technologies.
- To promote next generation technocracy and nurture entrepreneurial culture for social-economic growth.

Fig.4.2.4 Campus Tour Page

# **Campus Tour**



## Fig.4.2.5 Notifications Page

Date	Notifications
	APLAWCET 2022 – SPOT ADMISSIONS
	The extended date for submission of application for Executive and Extramural quota is extended upto 31-10-2022
3-8-2022	SVU DDE ADMISSION NOTIFICATION FOR THE ACADEMIC SESSION -2022-23
18-6-2022	APRCET-2021 :: Ph.D VACCANCIES FOR SPOT ADMISSIONS IN SV UNIVERSITY,TIRUPATI
May 2022	I SEM BCOM MAY,2022 time Table
May 2022	Degree III sem Timetable (REG & Supply) exam Timetable May'2022
May 2022	Degree I sem Timetable (REG & Supply) exam Timetable May'2022
17-03-2022	PG SPECIAL FEE STRUCTURE CATEGORY SEATS Extends Till 29-3-2022

Fig4.2.6 AboutUs Page



Administration

Vision & MIsion Campus Tour Notifications

About Us



## **Welcome to Sri Venkateswara University**

Sri Venkateswara University was established in 1954, in Tirupati to cater the educational needs and aspirations of people of the Rayalaseema Region of Andhra Pradesh, After completing 68 years of excellence in teaching, research, extension and outreach activities, the University is committed to cater the needs of higher education offering a full range of post-graduate programs in Arts, Sciences, Law, Management, Education, Physical Education, Engineering and Pharmacy disciplines. From a humble beginning of one College with six departments, the University has now grown into the second largest University in Andhra Pradesh having five constituent Colleges Viz. College of Arts, College of Sciences, College of Commerce, Management & Computer Science, College of Pharmacy, and College of Engineering accommodating 52 departments offering 88 programs.

The University has made rapid strides in the field of higher education and research and is adjudged as one of the best Universities in the country and got ACCREDITED with 'A+' GRADE BY NAAC-2017.

The most commonly reported advantage of studying in Sri Venkateswara University are good teachers and "positive environment". The students are not only taught but given Hands-on training and career preparation. Sri Venkateswara University has

UGC, New Delhi and awarded with Centers for Advanced Study (UGC-CAS), Based on research output university was awarded DST-PURSE program.

Several departments have received Special Assistance Programmes (SAP) from UGC, New Delhi FIST program from DST, New Delhi to improve infrastructural facilities. Most of the faculties are receiving research grants from various National and International funding agencies like UGC, DST, DBT, CSIR, ICMR, ICSSR, BRNS, ISRO, MNES, MoES, MoEF, DRDO and foreign institutes such as UNESCO, UK-DFID, ICRISAT, European Commission Programme ERASMUS MUNDUS. Utilizing these impressive funding, the faculty of the University have proactively interacted with Industry, Academic and Research Institutes at National and International level and entered into collaborative research agreements through MoUs (Memorandum of Understating) to conduct research work in frontier areas of National & International importance.

The students belonging to socially and economically backward communities and get State and Central Government Welfare Scholarships. Fellowships and Scholarships are also offered to research students of the campus by the University and also by the UGC. CSIR. DAE. ICMR. ISRO. ICAR. ICPR. ICHR. ICSSR. DST. DRDO. SRNS

Fig4.2.6 ContactUs Page

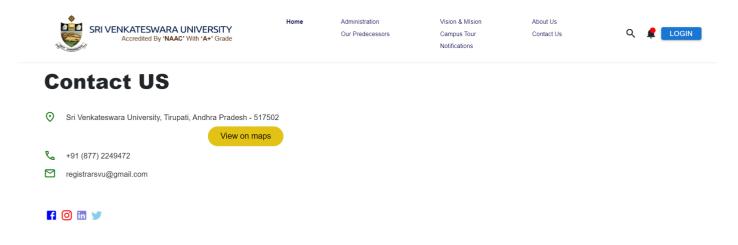


Fig.4.2.7 Student Signup Page





Fig.4.2.8 Student Signin Page





Vision & Mision

Campus Tour

Notifications

About Us Contact Us



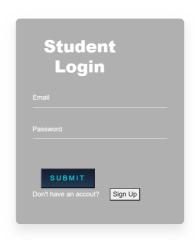
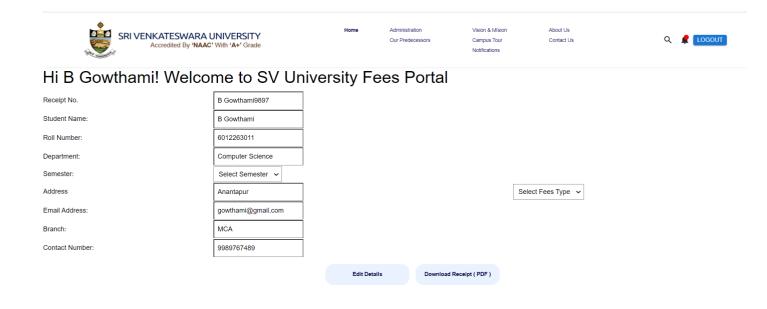


Fig.4.2.8 Student Home Page



#### Hi B Gowthami! Welcome to SV University Fees Portal Receipt No. B Gowthami9897 B Gowthami Student Name Roll Number: 6012263011 Computer Science Semester: Select Semester 🐱 Tution Fees **Tution Fees Details** Address Anantapur Total Amount to pay: 15000 Paid amount: 12000 Due: 3000 Pay You have paid: 12000 gowthami@gmail.com Email Address: Contact Number: 9989767489 Download Receipt ( PDF )

Fig.4.2.9 Payment Receipt

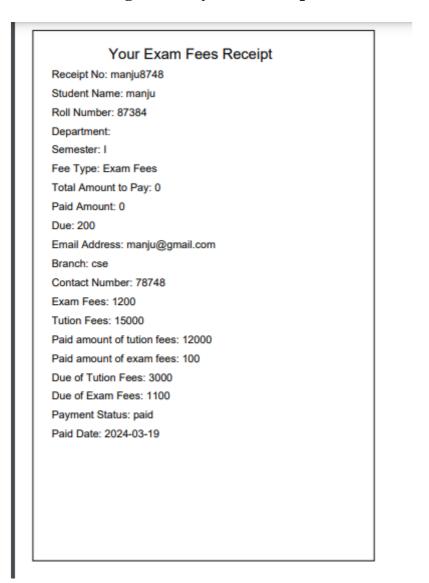


Fig.4.2.10 Admin Sign in Page



Administration
Our Predecessors

Vision & Mision
Campus Tour

About Us Contact Us

Q 🚅 LOGIN

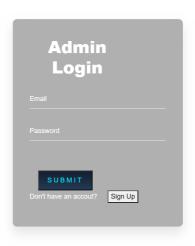
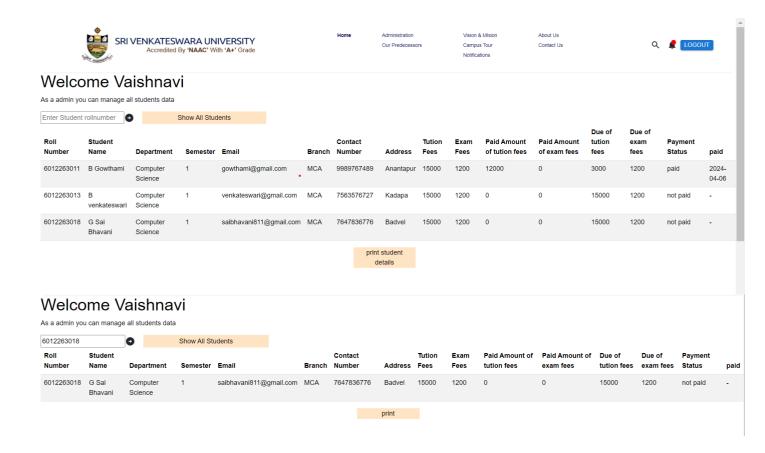
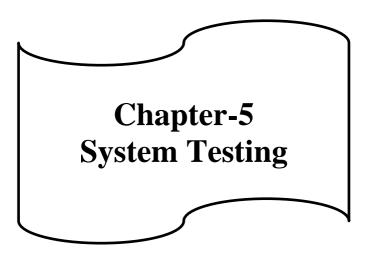


Fig.4.2.11 Admin Home Page



# **Fig.4.2.11 Printed Student Details**

Roll Number	Student Name	Department	Semester	Email	Branch	Contact Number	Address	Tution Fees	Exam Fees	Paid Amount of tution fees	Paid Amount of exam fees	Due of tution fees	Due of exam fees	Payment Status	paid
6012263011	B Gowthami	Computer Science	1	gowthami@gmail.com	MCA	9989767489	Anantapur	15000	1200	12000	0	3000	1200	paid	2024- 04-06
6012263013	B venkateswari	Computer Science	1	venkateswari@gmail.com	MCA	7563576727	Kadapa	15000	1200	0	0	15000	1200	not paid	-
6012263018	G Sai Bhavani	Computer Science	1	saibhavani811@gmail.com	MCA	7647836776	Badvel	15000	1200	0	0	15000	1200	not paid	



### 5.SYSTEM TESTING

Is the menu bar displayed in the appropriate contested some system related features included either in menus or tools? Do pull —Down menu operation and Tool-bars work properly? Are all menu function and pull down sub function properly listed?; Is it possible to invoke each menu function using a logical assumptions that if all parts of the system are correct, the goal will be successfully achieved .? In adequate testing or non-testing will leads to errors that may appear few months later.

This create two problem

- 1. Time delay between the cause and appearance of the problem.
- 2. The effect of the system errors on files and records within the system

The purpose of the system testing is to consider all the likely variations to which it will be suggested and push the systems to limits.

The testing process focuses on the logical intervals of the software ensuring that all statements have been tested and on functional interval is conducting tests to uncover errors and ensure that defined input will produce actual results that agree with the required results. Program level testing, modules level testing integrated and carried out.

## **Testing Methods**

There are two major type of testing they are

- 1) White Box Testing.
- 2) Black Box Testing.

## White Box Testing

White box sometimes called "Glass box testing" is a test case design usesthe control structure of the procedural design to drive test case.

Using white box testing methods, the following tests were made on the system

- a) All independent paths within a module have been exercised once. In our system, ensuring that case was selected and executed checked all case structures. The bugs that were prevailing in some part of the code where fixed
- b) All logical decisions were checked for the truth and falsity of the values.

## **Black box Testing**

Black box testing focuses on the functional requirements of the software. This is black box testing enables the software engineering to derive a set of input conditions that will fully exercise all functional requirements for a program. Black box testing is not an alternative to white box testing rather it is complementary approach that is likely to uncover a different class of errors that white box methods like..

- 1) Interface errors
- 2) Performance in data structure
- 3) Performance errors
- 4) Initializing and termination errors

## **Unit testing**

Unit testing is a software verification and validation method in which a programmer tests if individual units of source code are fit for use.

A unit is the smallest testable part of an application. In <u>procedural</u> <u>programming</u> a unit may be an individual function or procedure.

Ideally, each <u>test case</u> is independent from the others: substitutes like <u>method stubs</u>, objects, fakes and <u>test harnesses</u> can be used to assist testing a module in isolation.

## **Integration Testing**:

This testing is sometimes called Integration and Testing. Integration testingis the phase in software testing in which individual software modules are combinedand tested as a group. It occurs after unit testing and before system testing.

Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to thoseaggregates and delivers as its output the integrated system ready for system testing.

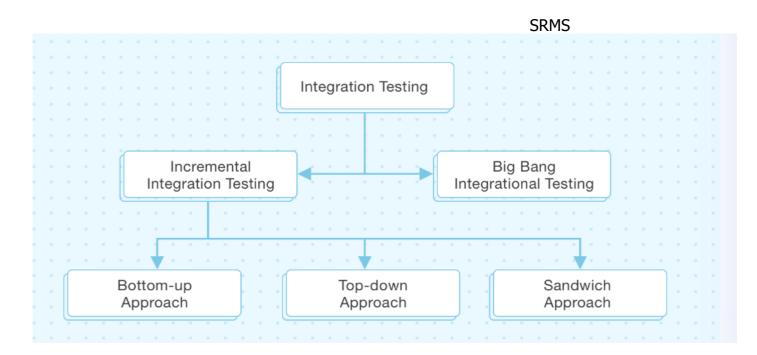


FIG 5.1 Integrating & Testing Stage

## **Validation Testing:**

Validation Testing can be defined in many ways, but a simple definition is that validation succeeds when the software functions in a manner that can reasonably expected by a customer. After validation test has been conducted, one of the following two possible conditions exists. The functions or performance characteristics confirm to specification and are accepted.

- In the administrator and marks modules, all the fields must be filled.
- In the student registration, mobile number should contain exactly 10 numbers.

## **User Acceptance Testing:**

User acceptance of a system is a key factor of any system. The systemunder consideration is tested for the acceptance by constantly keeping in

## SRMS

touch with the prospective system users at the same time of developing and marketing changes whenever required. This is done in regard to the following points:

- Input Screen Design
- Output Screen Design

## **6.2** Test Cases

NO	INPUT GIVEN	EXPECTED	ACTUAL	TEST PASS	ACTION			
		ОИТРИТ	ОИТРИТ		TAKEN			
			OCCURED					
1	Admin , pass	Admin Home	Admin Home	Yes	-			
		page	page					
2	bindu , bindu	student	student	Yes	-			
		Home page	Home page					
3	Admin, kumar	Admin Home	Invalid	No	The wrong			
		page	password for		password			
			user Admin		kumar is			
					given for user			
					Admin.			
4	phoneNumber	Student	Please enter	No	The phone			
		registration	a valid phone		number			
		successful.	number.		given is of 9			
					numbers.			
5	Adding of	Subject	Alredy	No	The subject			
	subject into	Allocated	Subject is		name given			
	the specified	Sucessfully	allocated		was already			
	branch				exists.			
	according to							
	semester wise							

### 6.CONCLUSION

The present research was based on the computerization and the implementation of a sophisticated Web-Based Student Fees Management System for the Catholic University of Mozambique. The main objective was to enhance and automate the Fee payments and monitoring of student details' using a computerized system. A well-defined, efficient, controlled and managed information system or software based on web technology storing, processing and providing information through the internet.

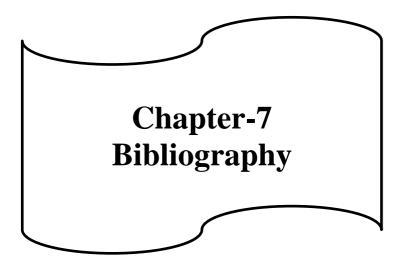
And the objectives were achieved by following a process model such as system analysis, design and system implementation.

## **Future Enhancements**

It is not possible to develop a system that makes all the requirements of theuser. Some of the future enhancement that can be done to this system are:

- As the technology emerges, it is possible to upgrade the system and can be adoptable to desired environment.
- Because it is based on object-oriented design, any further changes can be adaptable easily.
- Based on the future security issues, security can be improved using emerging technologies like single sign-on.

Develop an android application for access data and check their information through mobiles.



### **7.BIBLIOGRAPHY**

The following books were referred during the analysis and execution of the project.

### **Books Referred:**

- 1. HTML Black Book by Wiley Dreamtech press India Pvt Ltd.2000.
- 2. Spring Boot and MYSQL Web Development second Edition by Luke welling, LauraThomson was released in 2003.
- 3. Microsoft SQL Server –2000 Editions are nine and ten by Rankin, Paul & Jensen.
- 4. PP In a Nutshell by Paul Hudson released in October 2005.
- 5. Software Engineering by Roger Pressman, Elizabeth A. Jones, 12<sup>th</sup> march 1982.
- 6. Java: A Beginner's Guide by Herbert Schildt.

## Websites:

- <u>WWW.GOOGLE.COM</u>
- WWW.CODEWITHRANDOM.COM
- WWW.YOUTUBE.COM
- WWW.W3CSCHOOLS.COM