PRACTICAL: 02

Practical 2.1:

Aim: To perform the system analysis task of your system

Prepare Software Requirements Specification Document.

Program:

SRS –Student Fee Management System.

Table Of Content:

1. Introduction

1.1 Purpose

The purpose of this system is to build an online fees system for the students of an institution to manage every data related to student and payment and providing short or detailed summary reports on that. It also keep real-time tracking of their ward's fee payment details.

1.2 Document Conventions

Font family: Times new roman
Heading:
    Font-size: 14
    Font-style: bold
Main content:
    Font-size: 12
    Font-style: normal

1.3 Intended Audience and Reading Suggestions

The document is intended for, such as software developers, project consultants, and team managers. Readers can start reading from introduction followed by overall description, External interface requirements, System features, Other Non-functional requirements and other requirements.

1.4 Product Scope

A student fees management system allows payment through different gateways. When students submit their fees online, it is easier for the admin to collect it and make records of it. The reports generated include all the important details which you can access and view anytime you want.

1.5 References

IEEE Recommended Practice for Software Requirements Specifications-IEEE Std 830-1999.

2. Overall Description

2.1 Product Perspective

Student Fees management system allows students to make online payments from anywhere at any time. They do not have to go through the hassles of making manual fee payment.

Student fees management systems are customizable and they also provide a user-friendly platform. It makes it much simpler for the students as well as the administrative to manage payments and process error-free transactions.

2.2 Product Functions

This project aims at developing an independent and easy to use software-based system that can help students to easily view details and pay fees online and this system will also help administration to keep track of their fees transaction.
The proposed software shall have the following modules or functions:

- Login module
- Update fees detail module
- View details module
- Online Payment module
- Report generation module

2.3 User Classes and Characteristics

Users	**Characteristics**
Admin	● Login ● Manage Student info. ● Manage fees details ● Manage user credentials
Student	● Login ● View Student details ● View Payment details ● Pay fees online ● Mode of payment ● Generate Receipt

2.4 Operating Environment

Operating environment for the Student fees management system is as listed below:

- distributed database
- client/server system

- Operating system: Windows, Linux.
- database: MYSQL
- Platform: Front End: HTML, CSS, JavaScript are utilized to implement the frontend. Back End: The back end is implemented using MySQL/PHP which is used to design the database.

2.5 Design and Implementation Constraints

The Student Fees Management System software is designed in such a way that the user can easily interact with the screen because of GUI. The admin and the user are the two users who use the project. The admin inserts the details of the students and the fee details that the students have paid. Student can view his/her details, check the fee details and can pay fees online.

Constraints

- The system will only manage the fees of college Students.
- The system will only run in Windows Desktop and Android Phone.
- A working payment gateway is needed to demonstrate the practical working of the system else the gateway will have to be simulated.

2.6 User Documentation

In our user manual, we are going to keep the information regarding system which can be understandable by a new person who is going to use it. If a new person is using it online help will be provided in that we are going to explain each and every step clearly by system can be useful for any user.

2.7 Assumptions and Dependencies

The system is dependent on the following assumptions:

- The database of registered students will be made available along with the login details of admin.
- The fee structure of the academic year will be made available beforehand.
- We are assuming access to the payment gateway service of any bank.
- The transaction history of the institution’s fee account will be made available to the admin for verification.
- The Admin should not be corrupt.

3. External Interface Requirements

3.1 User Interfaces

The user interface for the software shall be compatible to any browser such as Chrome, Internet Explorer, Mozilla, Netscape Navigator, etc by which user can access to the system. This website will provide very user-friendly interface and it is quite easy to use. First, the user is directed to the system's home page, which includes several filters such view fees details, online payment, generate receipt and so on. The user selects filters based on their needs. Although for making payment online the users require to login first. The system also provides an various mode of payment for paying fees. Users also can generate fees reports.

3.2 Hardware Interfaces

This application is supported by:

Windows devices, Android devices, Linux devices etc. through any browser.

3.3 Software Interfaces

Following are the software used for the Covid-Resource Management System.

Software used	Description
Operating System	Student Fees Management System is platform independent it can run on any device having we browser.
Database	To save the student records, fees records we have choose MySQL
Php and JavaScript	To develop backend of the system.
Html and CSS	To create frontend of our system.

3.4 Communication Interfaces

SMTP: Used for simple mail transfer

HTTP/HTTPS: For transmitting hyper-media documents

FTP: Used for file transfer

4. System Features

Student Info:

4.1.1 Adding Student details:

Admin will deal with all new students to add them with their personal details/information to database.

4.1.2 Updating details:

Admin will update student info, and fees details to database.

4.1.3 Deletion details:

Admin will delete student detail if require and also update in database.

4.1.4 Setting up fees structure:

The system should make it possible to collect fines or late payment fees from students.

Online Payment:

4.2.1 Payment Gateway:

The payment gateway will give student options for selecting payment mode through credit/debit card, Net banking, UPI, etc. Students have to select the mode and further enter details and make payment.

4.2.2 Fee Payment:

The user can select the type of fees he want to pay, i.e partial payment or complete payment.

Generating report:

4.3.1 Generate Receipt:

User can generate receipt as a proof of payment for every fees transaction mode.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

- The performance of the system should be fast and accurate.
- System shall handle expected and non-expected errors in way that prevent loss in information and long downtime period. Thus it should have inbuilt error testing to identify search or data check/fetch.
- The system should be able to handle large amount of data. Thus it should accommodate large number of data entry of a particular students without any fault

5.2 Safety Requirements

The database may get crashed at any certain time due to virus or operating system failure. Therefore, it is required take the database backup.

5.3 Security Requirements

- System will use secured database.
- Normal users can just read instruction and use this system but they cannot edit or modify anything except paying fees.
- Proper user authentication should be provided.
- No one should be able to hack user's details or any other information.
- There should be separate part for users that no users can access the database and only admin has the rights to update the database.

5.4 Software Quality Attributes

Availability:
The system should be available for 24 hours

Reliability:
The system should update and show transaction real-time.

Maintainability:
The system must be able to modify the stored records so that any change in the student information that can occur can be accommodated easily in the student database and also backup for database are available.

Usability:
As the system is implemented through a universal app the user can log into the system both from his mobile and desktop, anywhere.

Capacity:

The system must have the capacity to handle large amount of student data and should not slow down as the amount of data increases.

5.5 Business Rule

A business rule is anything that captures and implements business policies and practices. A rule can enforce business policy, make a decision, or infer new data from existing data. This includes the rules and regulations that the System users should abide by. This includes the cost of the project and the discount offers provided. The users should avoid illegal rules and protocols. Neither admin nor member should cross the rules and regulations.

6. Other Requirements

Appendix A: Glossary

SRS :
- Software Requirements Statement
- Statement clarifying the what a software project is supposed to be engineered to do

IEEE :
-The institute of Electrical and Electronics Enginees, Inc.

FTP :
- File Transfer Protocol
- Protocol to send/receive files to/from an FTP server

GUI :
-Graphical User Interface
- The interface which the user uses on graphics displaying hardware c. Refers to the layout of the dialog boxes, menu elements, etc

SQL :
-Structured Query Language
-A programming language used for making queries to a database, and setting/retrieving data to/from it

SMPT :
-Simple Mail Transfer Protocol
-Simple protocol for dealing with email processing online

OS :
-Operating system
-An operating system (OS) is system software that manages computer hardware, software resources, and provides common services for computer programs.

HTTP:
-Hypertext transfer protocol
-Hypertext Transfer Protocol is an application protocol for distributed, collaborative, hypermedia information systems that allows users to communicate data on the World Wide Web.

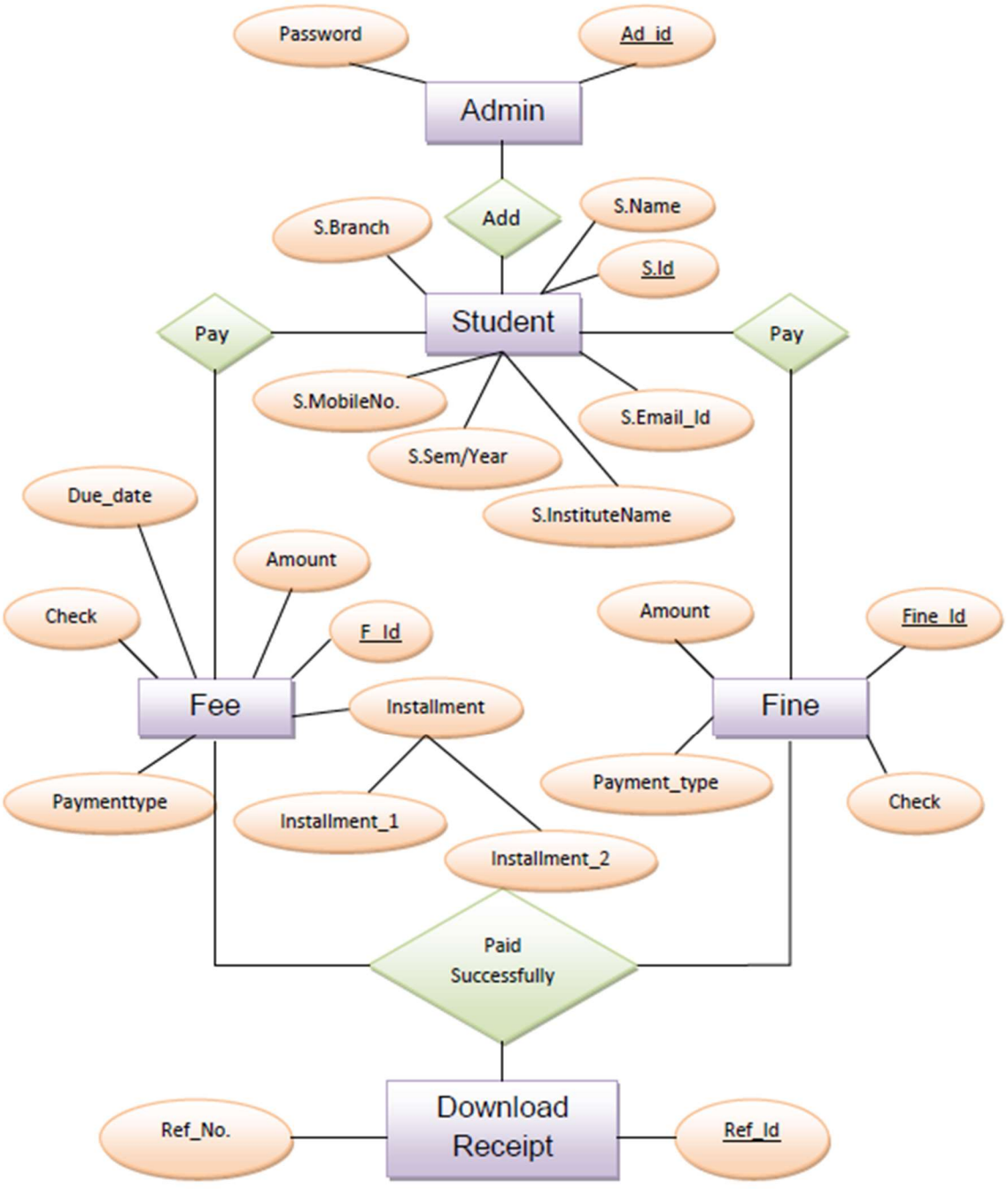
HTTPS:
-Hypertext Transfer Protocol Secure
-It is used for secure communication over a computer network, and is widely used on the Internet.

ERD:
-Entity Relationship Diagram
-An entity relationship diagram is a graphical representation that depicts relationships among people, objects, places, concepts or events within an information technology (IT) system.

DFD:
-Data-flow diagram
-A data flow diagram maps out the flow of information for any process or system.

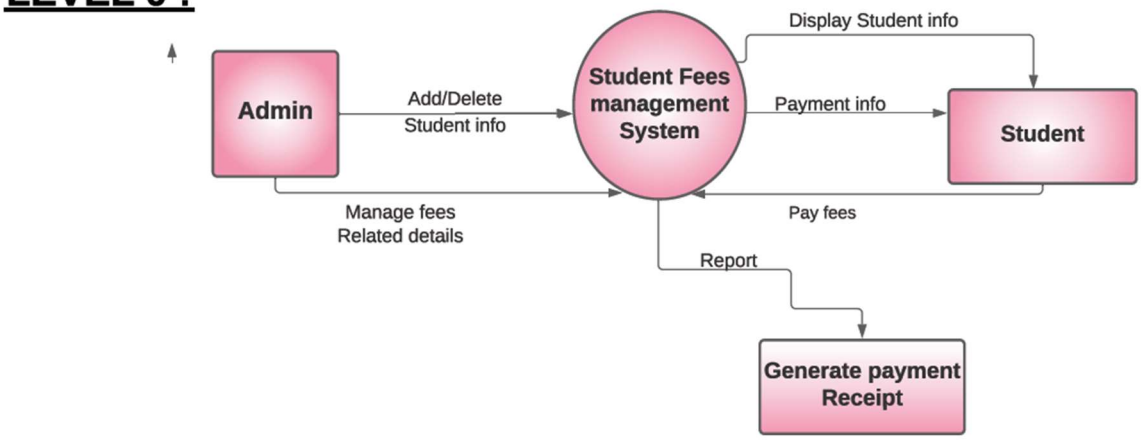
Appendix B: Analysis Models

ER-Diagram:

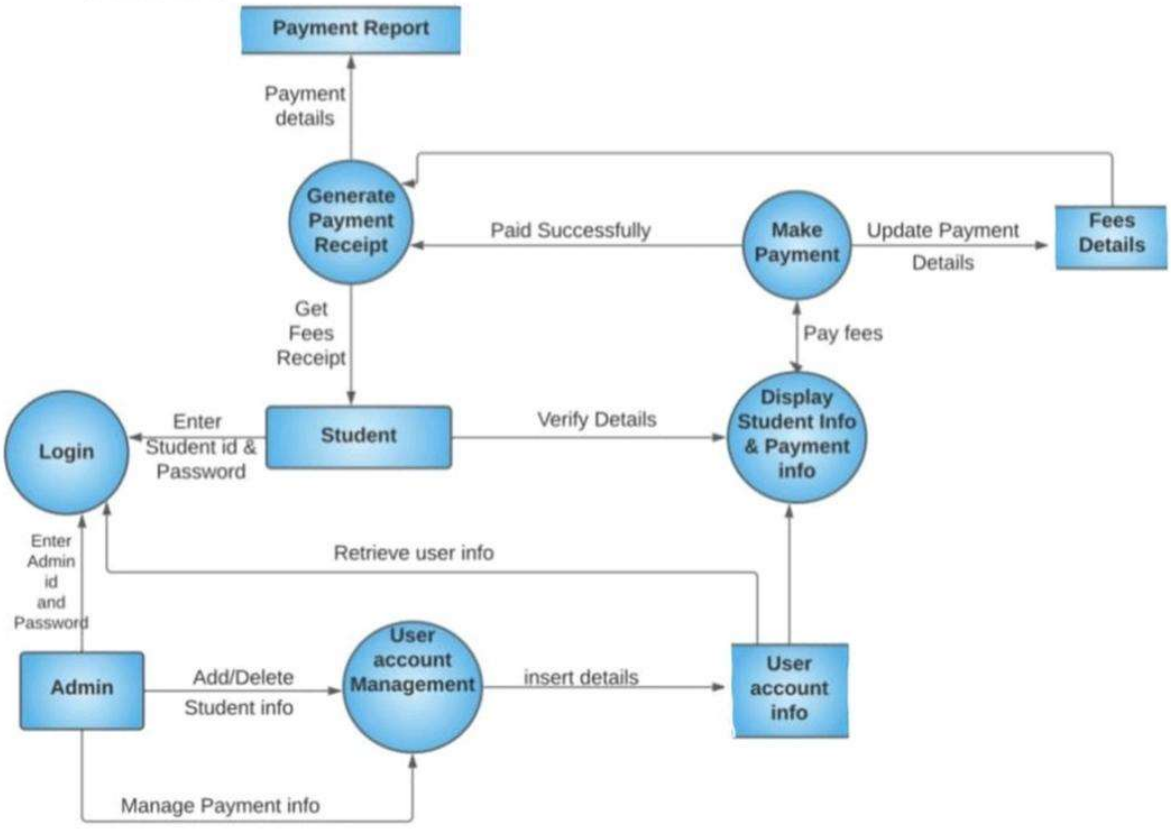


Data Flow Diagram:

LEVEL 0 :

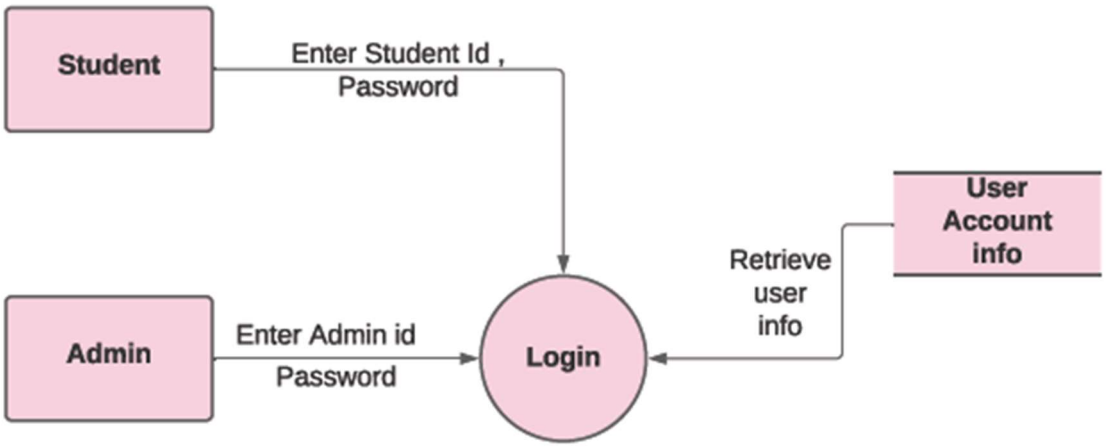


LEVEL 1 :

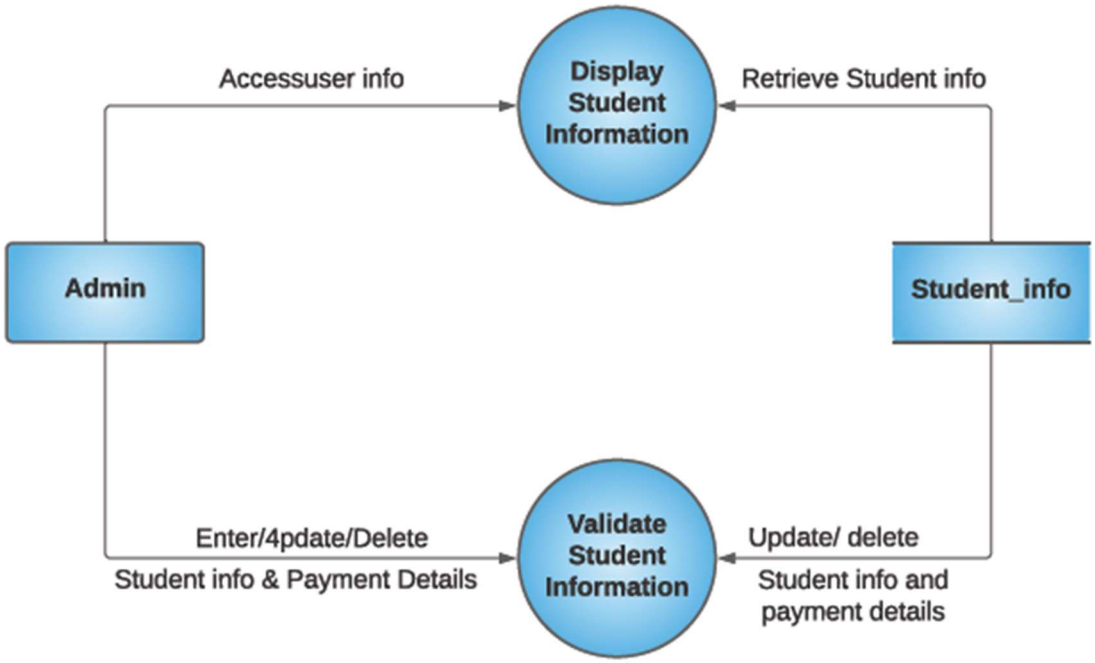


LEVEL 2 :

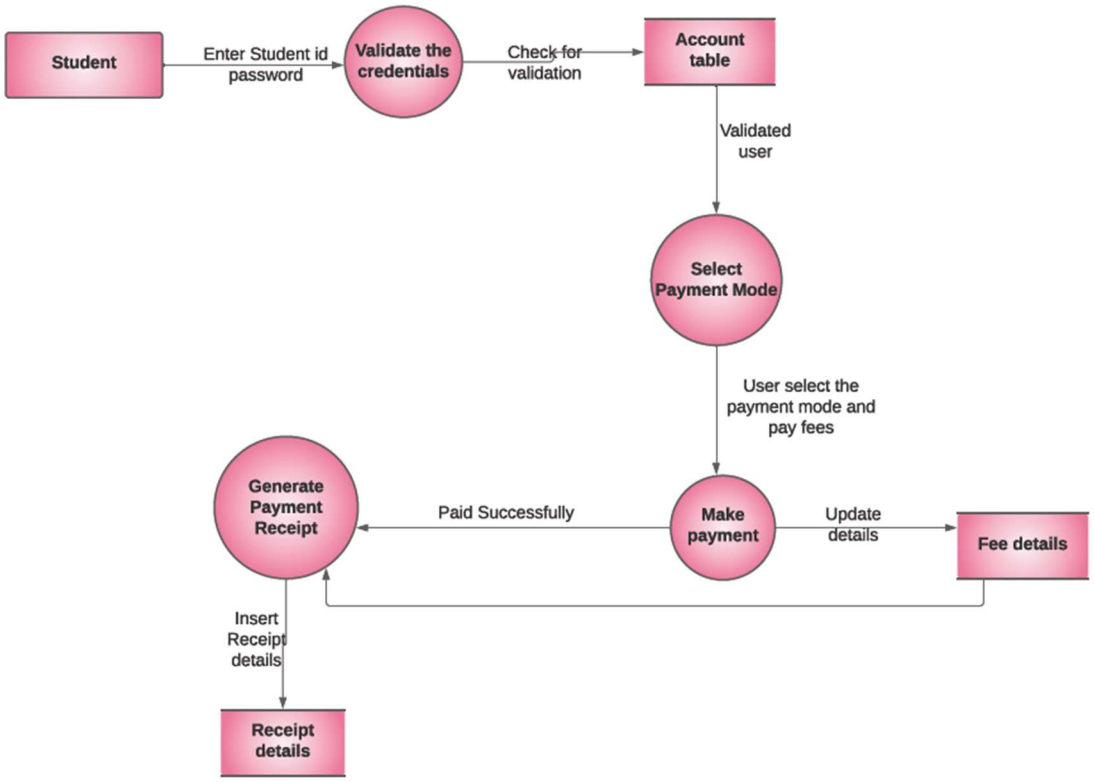
1. Login :



2. Student Information Management :



3 . Online Payment :



Appendix C: To Be Determined List

- IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications
- https://www.acs.ac.in/college-online-fees-payment/