Software Design Specification

Amigo

RAHUL ROY(IIT2019194)
AYUSH KHANDELWAL(IIT2019240)
NISCHAY NAGAR(IIT2019198)
ESHAN VAID(IIT2019230)

Table of Contents

Introduction

“Amigo” is basically a web application that offers a user friendly interface for management of academic affairs (Admissions, Assessments, Awards and Archives).

1.1 Purpose

The process of admission and academic management of our institution is less efficient and clumsy in our organisation (NU), as per one of the spokesperson from the Academic Department. Hence we aim to create a platform for smooth, easy and efficient interaction between the students, academic administration and FnA.

1.2 System Overview

The system’s aim is to create a platform for smooth, easy and efficient interaction between the students, academic administration and FnA.
The app should be suitable for all device screen sizes. The class-timetable and academic calendar should be updated on a regular basis efficiently.

1.3 Definitions and Acronyms

1) *Text highlighted in the srs means a special significance or importance to the reader or any institutional staff of IIIT Allahabad.*
2) *Following are the abbreviations used:*
   AM - Academic Management
   AMS - Academic Management System
   RPS - Registration Procedure System
   “Amigo©”;: Copyrighted app name.
   SRS: Software Requirement Specification
3) *“Amigo” is an independent group project which does not share any similarity with other app.*

2) Design Considerations

- *The app should be suitable for all device screen sizes*

2.1) Assumptions

Metrics and TaskView reports will be handled in the workflow reporting project.

2.2) Constraints

- should have separate windows/screens for handling HoD data and FnA data.
- *The app should be suitable for all device screen sizes.*
- *app should be time efficient (take optimized time for each operation)*
- *The user interface (UI) should support both modern and old forms of styles that can help users work according to their styles and usage.*

2.3) System Environment

The communication between the different parts of the system is important since they depend on each other. However, in what way the communication is achieved is not important for the system and is therefore handled by the underlying operating systems for web application. But, any transaction on the web deals with the following protocols. TCP/IP,HTTP

All the software required for a Web Browser needs to be installed and act as an interface for the product and the user.

All the hardware on which a Web Browser can be installed can act as an interface for the product and the user. The Hardware can be mobile, laptop, pc and tablet.

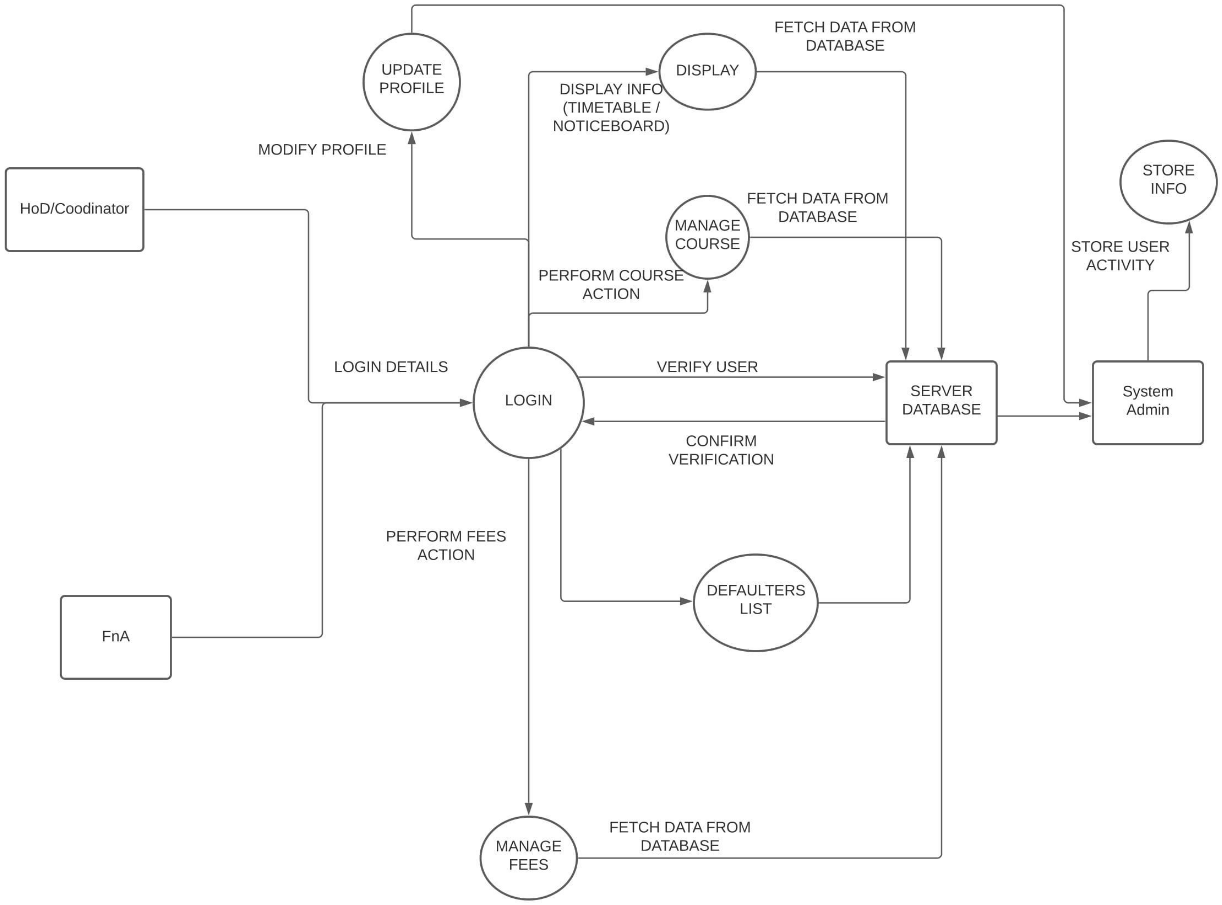
2.4) Design Methodology

We will be incorporating the following design architectures:

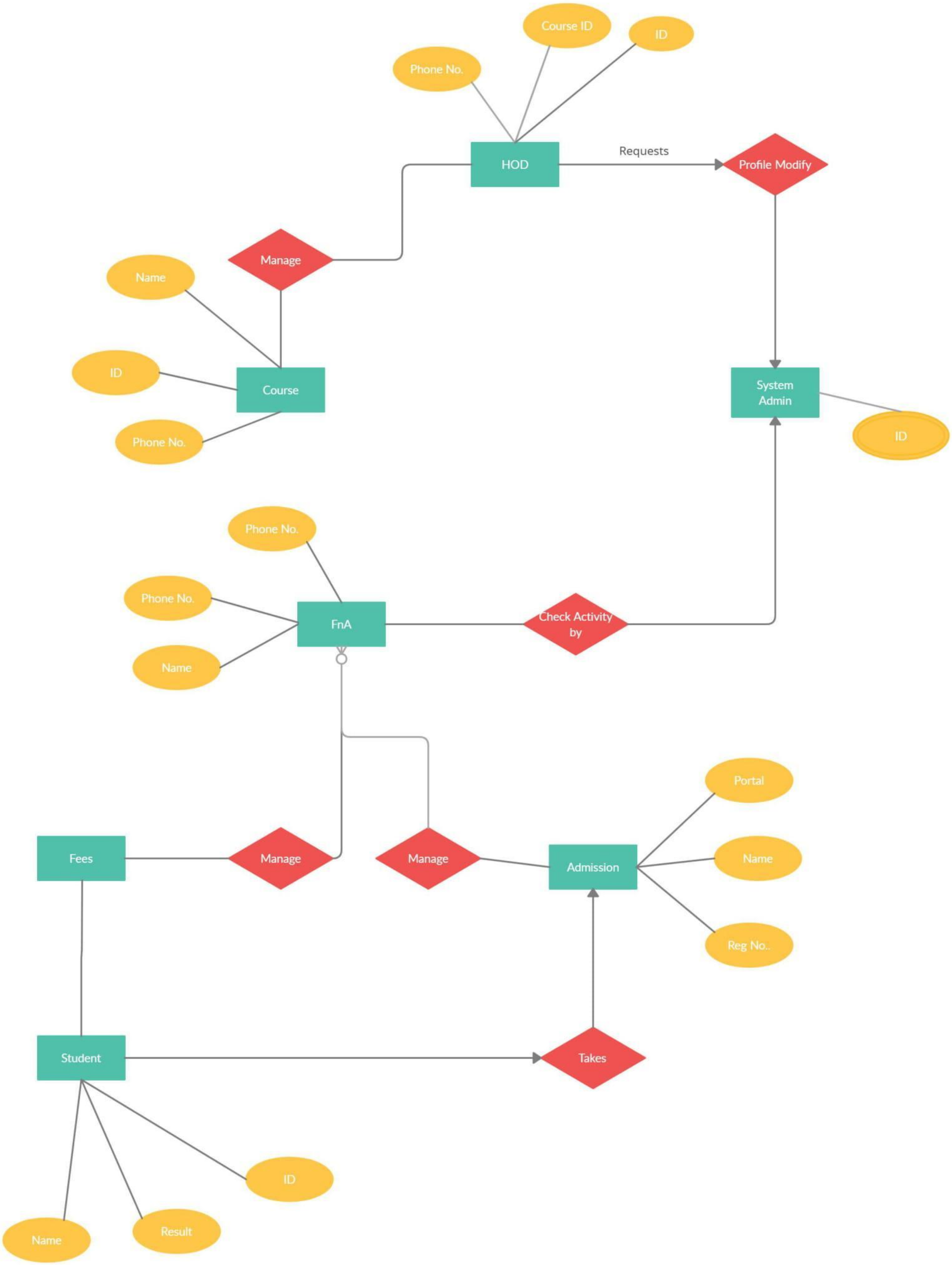
- *The app should be suitable for all device screen sizes.*
- *Designing caching, data access mechanism, and state management considering times of intermittent connectivity.*
- *The user interface (UI) should support both modern and old forms of styles that can help users work according to their styles and usage. It should have easy and simple implementation levels.*
- *Using Stacked navigation Bar.*
- *Bring in consistent flow to the architecture.*

3) Architecture

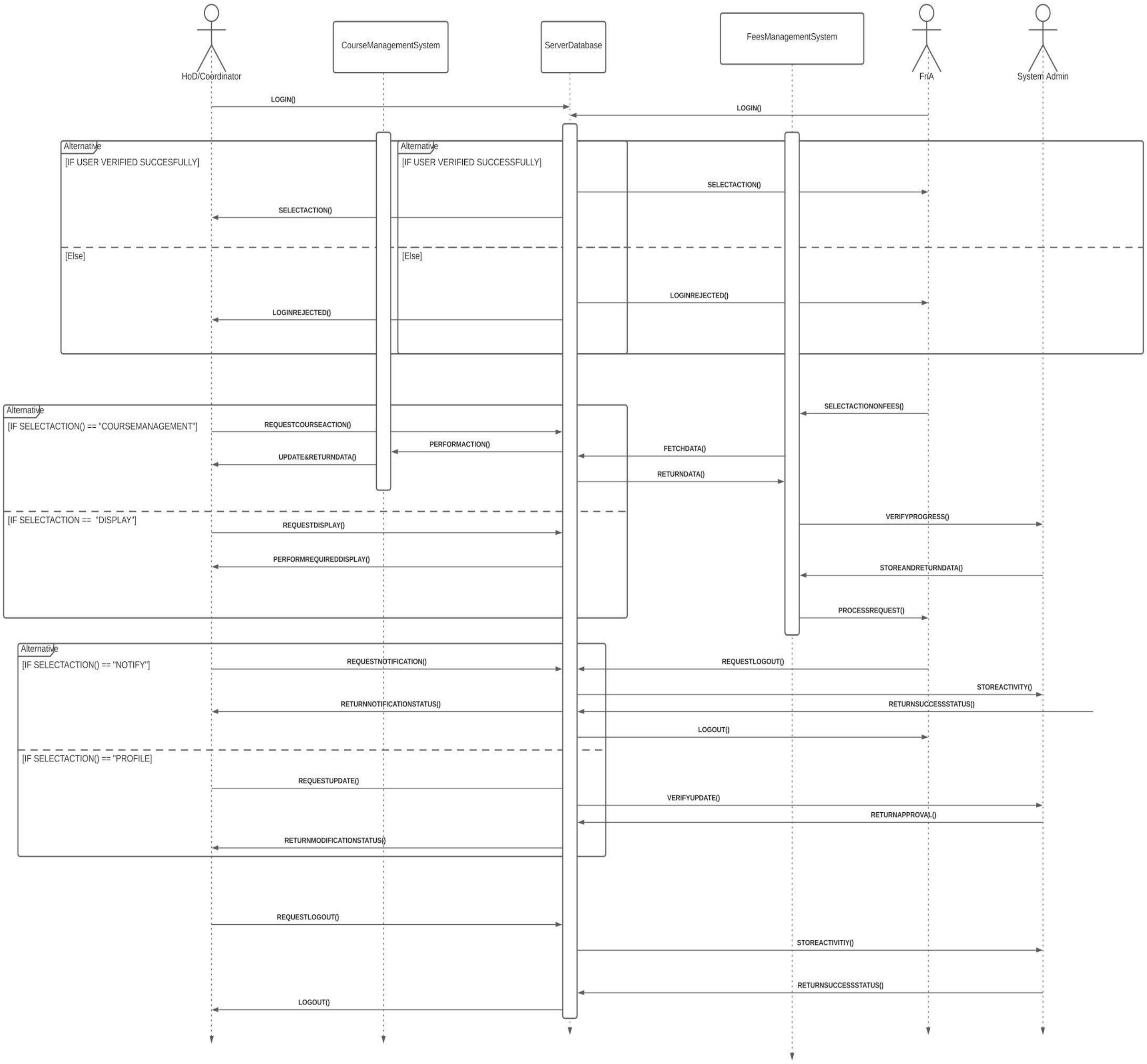
3.1) Data Flow Diagram



3.2) E-R Diagram


Course ID
ID
Phone No.
HOD
Requests
Profile Modify
Manage
Name
ID
Course
Phone No.
System
Admin
ID
Phone No.
Phone No.
FnA
Check Activity
by
Name
Portal
Fees
Manage
Manage
Admission
Name
Reg No..
Student
Takes
ID
Name
Result

3.3) Sequence Diagram



3.4 Overview

Our app basically serves the purpose of academic and management affairs in IIIT Allahabad. We have 3 main components or modules which are namely :

- *Student*
- *HoD*
- *FnA*

3.5 Modules/Components

Module 1 : STUDENT

We have prepared a user-friendly and efficient interface for students to manage their fees payment, get important notifications, check their result and performance and also set up plans for future events according to the academic calendar

- *Sub-Module 1 : Profile View*
  - *allows the student to check his/her profile.*

- *Sub-Module 2: Fees Payment*
  - *allows to check fee status*
  - *allows to pay the fees (redirected to fees portal separately)*
  - *gets notification regarding late fee and extended deadline*

- *Sub-Module 3: Semester Results*
  - *allows the student to check his/her current semester result.*
  - *gets notified on when the final result is released.*

Module 2 : HoD

We have prepared a dynamic and efficient interface for Hod to manage academic affairs, send important notifications, check and manage faculty and coordinator list for each subject and also modify profile.

- *Sub-Module 1 : Profile Modification*
  - *allows HoD to view and modify profile.*

- *Sub-Module 2 : Faculty List Management*
  - *allows the user to check and update the faculty list.*
  - *allows the user to assign coordinator for each course.*

- *Sub-Module 3 : Manage Class Time Table*
  - *allows the user to update the time table for each batch/program.*
  - *allows the user to upload the update the uploaded time table*

- *Sub-Module 4 : Manage Notice Board*
  - *allows the user to upload notice and update everyone about the same.*

- *Sub-Module 5 : Course Allotment and Management*
  - *allows the user to allot courses to various batches/programmes/degrees.*
  - *allows the user to create/modify/delete any course and update about the same.*

- *Sub-Module 6 : Send Notifications*
    - *allows the user to send notifications to anybody.*

Module 3 : FnA

We have prepared a precise, secure and efficient interface for the Fees and Admission department to manage their fees payment and registration, send important notifications, check the fee status of all students and also allow them to extend the late fee payment deadline and amount of late fee.

- *Sub-Module 1 : Fee Status*
    - *allows the user to check the fee status of all students.*

- *Sub-Module 2 : Late Fee Management*
    - *allows the user to manage late fee payment.*
    - *allows the user to extend the deadline for late fee payment*
    - *allows the user to notify the students regarding late fee*

- *Sub-Module 3 : Registration*
    - *allows the user to verify registration of the student.*
    - *allows the user to notify the student on successful registration.*

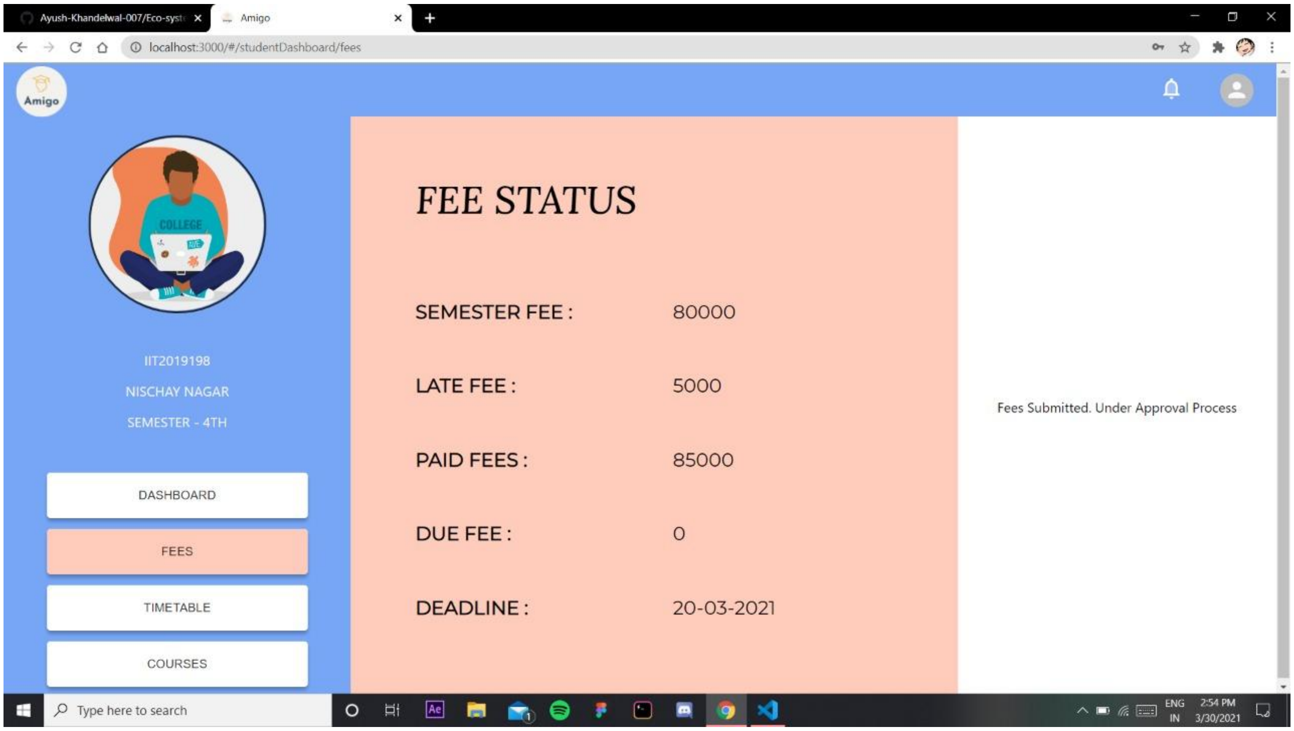
4) Database Schema

1.1 Tables, Fields and Relationships

1) ***Table - FnA***
    a) Fields : email, password
2) Table - HoD
    a) Fields : branch, email, name, password
3) Table - Students
    a) Fields : branch, dob, email, feeStatusAtAdmis, feeStatusAtReg, feeApproved, imageurl, name, password, roll, semester
    b) Table - fees
        i) Fields : due, latefee, paid, semfee
    c) Table - homeaddress
        i) Fields : address, city, country, state, zip
4) Table - Teachers
    a) Fields : name

Database used here for development, testing and production is ➢ Firebase.

5) High Level Design



IIT2019198
NISCHAY NAGAR
SEMESTER - 4TH

FEE STATUS

SEMESTER FEE : 80000

LATE FEE : 5000

PAID FEES : 85000

DUE FEE : 0

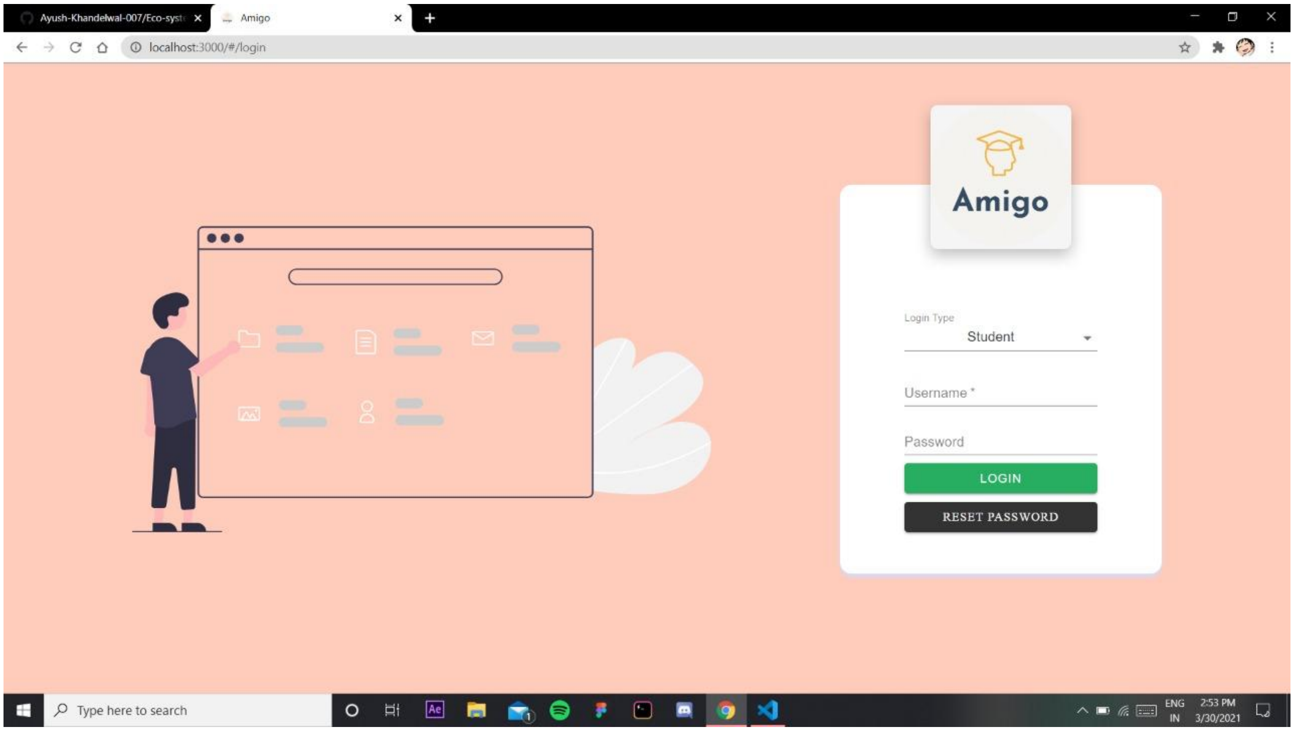
DEADLINE : 20-03-2021

Fees Submitted. Under Approval Process

The main aim was to manage fees and academics which we have achieved through a stable and interactive design.
The above design shows our color scheme and features added.
The student can pay the semester fees, see his/her fee status , see the late fee penalty and due fees along with the extended deadline.

6) UI DESIGN

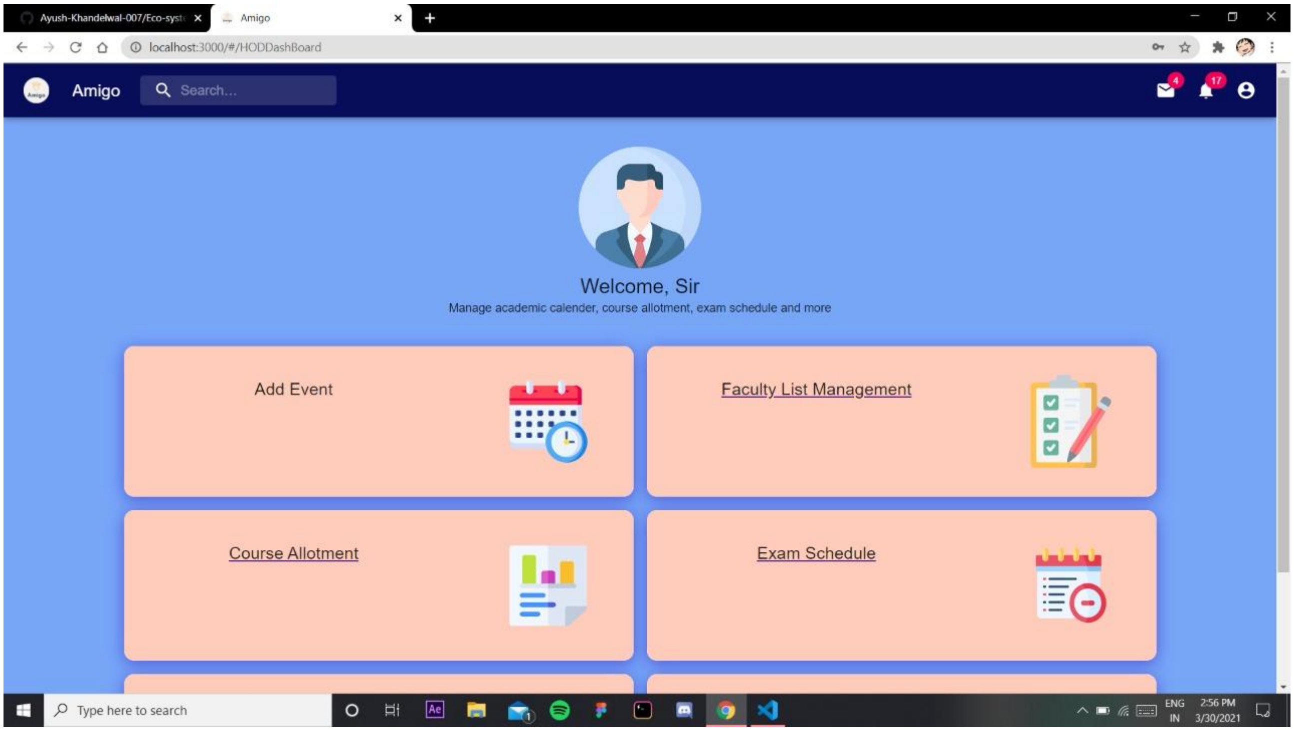
Home Page :-



Login Type

Student

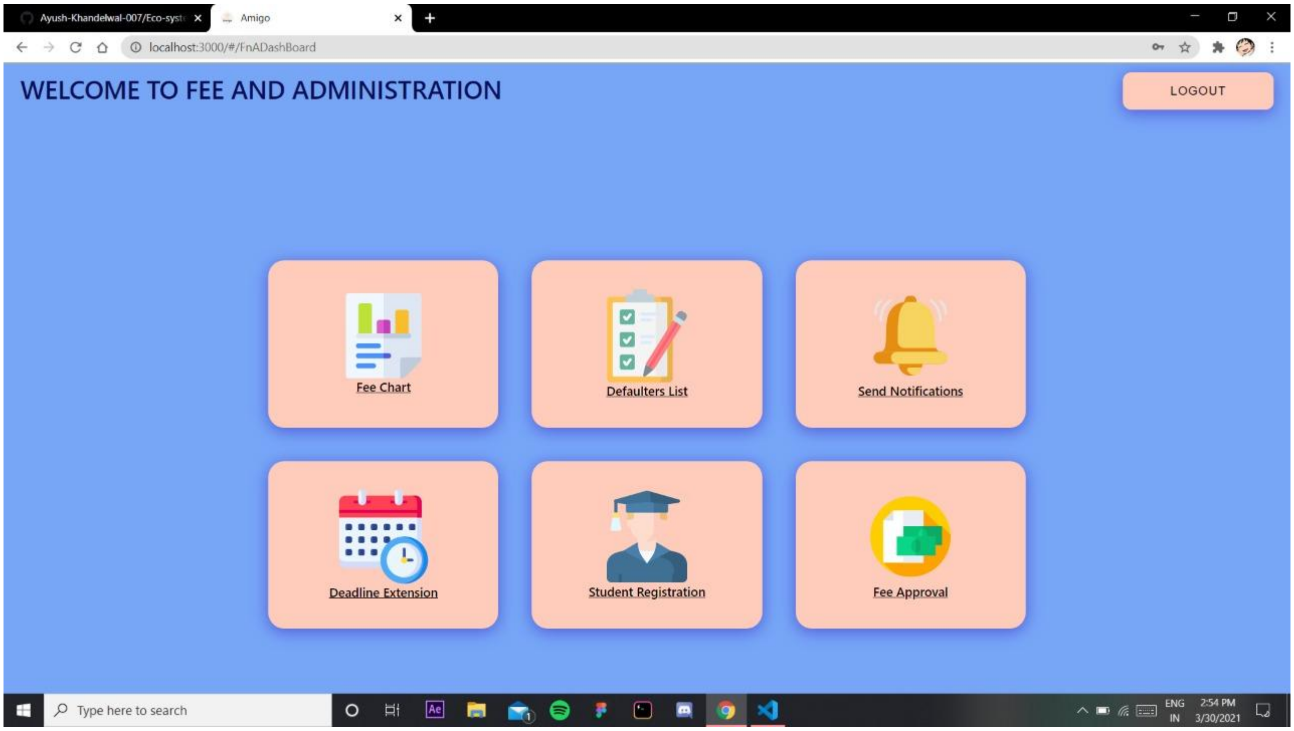
HoD :-



Welcome, Sir

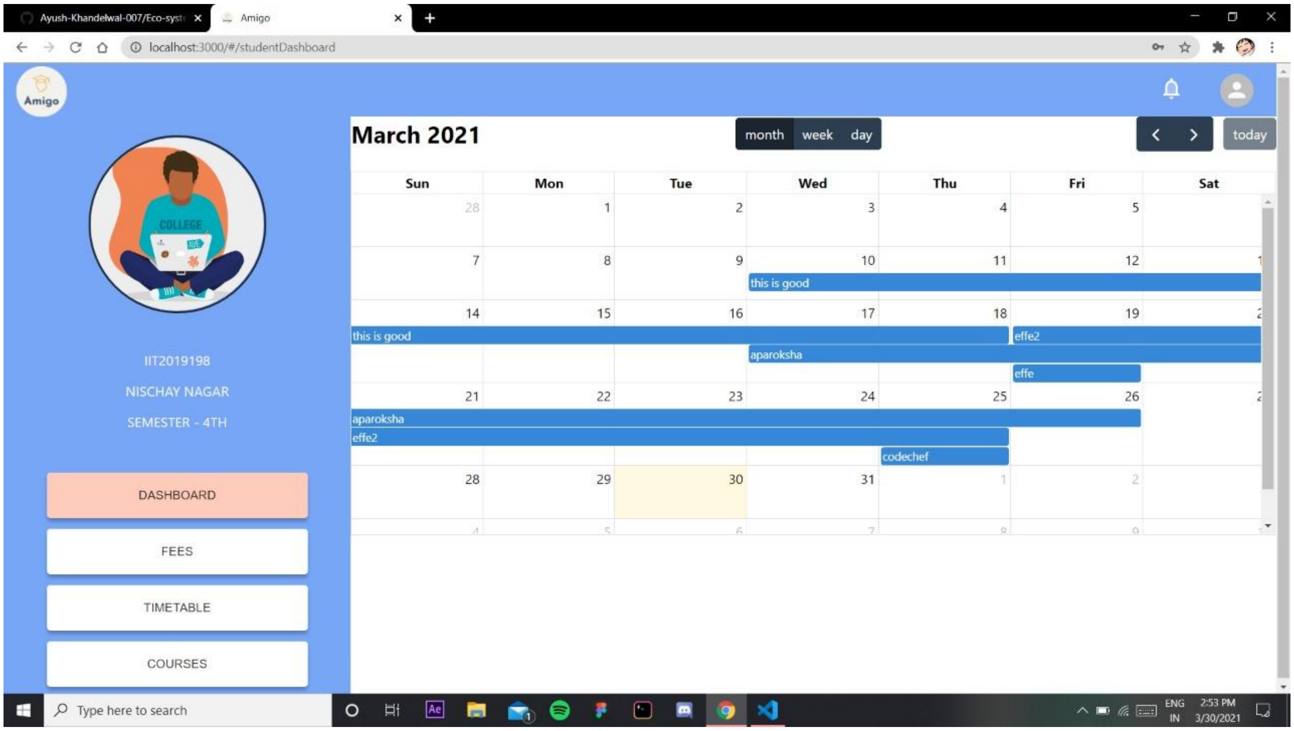
Manage academic calender, course allotment, exam schedule and more

FnA :-



WELCOME TO FEE AND ADMINISTRATION

Calender :-



IIT2019198

NISCHAY NAGAR

SEMESTER - 4TH

March 2021

Sun	Mon	Tue	Wed	Thu	Fri	Sat
28	1	2	3	4	5	
7	8	9	10 this is good	11	12	1
14 this is good	15	16	17 aparoksha	18	19 effe2 effe	2
21 aparoksha effe2	22	23	24	25 codechef	26	2
28	29	30	31	1	2	
4	5	6	7	8	9	1