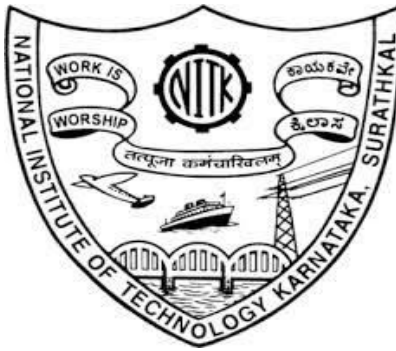# CO303 DATABASE MANAGEMENT SYSTEMS LAB

A REPORT ON THE PROJECT ENTITLED

ONLINE ATTENDANCE MANAGEMENT SYSTEM

SUBMITTED BY

SHWETA HARIHARAN IYER - 171CO245

NIWEDITA - 171CO227

V SEMESTER B-TECH

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA, SURATHKAL

2019-2020

INDEX

1.   ABSTRACT

Online Attendance Management System is an innovative tool to maintain and manage the attendance of students for the courses they register. It helps to ease the process of maintaining student and faculty attendance on paper. It generates attendance of a student on the basis of his/her presence in class. To provide attendance on a daily basis, each faculty member is provided with a username and password, using which they log in to their portal and give attendance to the students who have registered for a course taken by that faculty member.

This system is an application developed in python3.7. It is user-friendly and very intuitive. It is fast and can perform many operations which are necessary for a university/college/school. It is simple to understand and can even be used by people who are not even familiar with the workings of an attendance system.

This software package has been developed using HTML, CSS, Bootstrap and Javascript at Front End and Django at Back End with MySQL database. This version of the software has a multi-user approach.

2.   INTRODUCTION

2.1 Purpose

Online Attendance Management System is a software developed for daily student attendance in schools, colleges and institutes. It provides access to attendance information of a particular student registered for a particular course. This system helps in evaluating the eligibility attendance criteria of a student.

2.2 Scope

The scope and use of an online attendance management system is to computerize the traditional way of taking attendance in educational institutions.

The system will automate various tasks associated with handling student attendance details and better organising the stored information and optimum performance, thus helping educational institutions to ensure smooth working of these processes.

2.3 Requirement Specifications

2.3.1 Software Requirements

DBMS : MySQL Server version: 5.7.27

Backend : Django version 2.2.6

Development tool : PyCharm

Operating System : Ubuntu 16.04/18.04

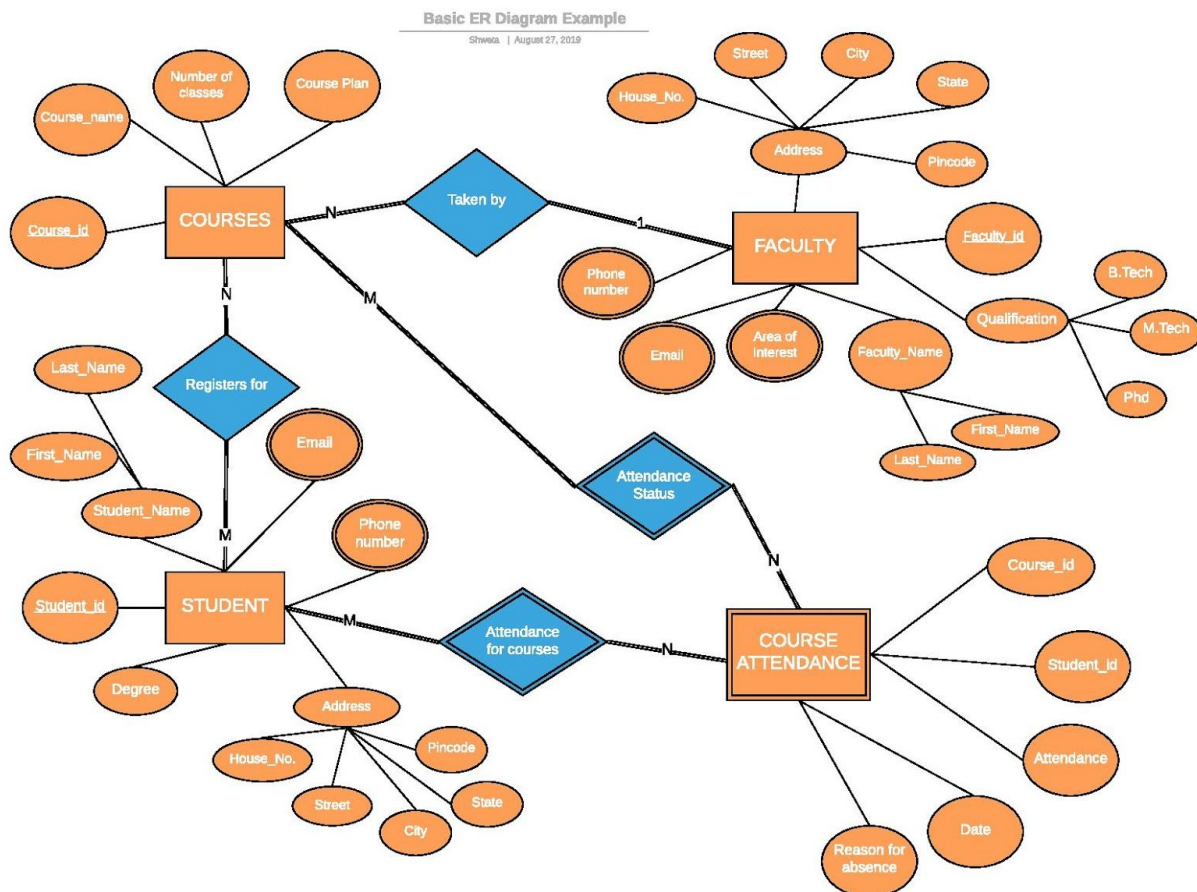### 2.3.2 Hardware Requirements

Processor: Intel Core

CPU: i5-7200U CPU @ 2.50GHz × 4

RAM: 8GB. , Hard Disk space : 1000GB

Architecture : x86-64

3. DATABASE DESIGN

3.A   ER DIAGRAM

This ER diagram represents the model of Attendance management System. The entity-relationship diagram shows all the visual instruments of database tables and relations between Student, Faculty, Course, Course Attendance etc. It uses structured data and defines the relationship between structured data groups of Attendance management System functionalities. The main entities of Airport Management System are Student, Faculty, Course, Course Attendance.

Faculty Entity

Attribute Name	Meaning	Value
fid	Faculty ID	varchar(8) [0-9]{2}[A-Z]{2}[0-9]{3}
f_fname	Faculty first name	varchar(15)
f_lname	Faculty Last name	varchar(15)
b-tech	B-tech degree	varchar(20)
m-tech	M-tech degree	varchar(20)
phd	PHD degree	varchar(20)
house_no	house number	varchar(10)
street	street name	varchar(15)
city	city name	varchar(20)
state	state name	varchar(20)
pin-code	pin-code number	integer(6)
phone	phone number	integer(10)
email	email id	email
area of interest	area of interest	varchar(20)

Student Entity

Attribute Name	Meaning	Value
sid	Student ID	varchar(8) [0-9]{2}[A-Z]{2}[0-9]{3}
s_fname	Student first name	varchar(15)
s_lname	Student last name	varchar(15)
degree	Degree	varchar(20)

house_no	House number	varchar(10)
street	Street name	varchar(15)
city	city name	varchar(20)
state	state name	varchar(20)
pin-code	pin-code number	integer(6)
phone	phone number	integer(10)
email	email id	email

Course Entity

Attribute Name	Meaning	Value
cid	Course ID	varchar(5)
cname	Course name	varchar(15)
no_of_classes	no of classes required to complete the course	Integer
course_plan	course plan	varchar(100)
fid	Faculty ID related to the course	Foreign key to fid in faculty info table.

Course Registration

Attribute Name	Meaning	Value
cid	Course ID	varchar(5)
sid	Student ID	varchar(8) [0-9]{2}[A-Z]{2}[0-9]{3}

Course Attendance

Attribute Name	Meaning	Value
cid	Course ID	varchar(5)
sid	Student ID	varchar(8) [0-9]{2}[A-Z]{2}[0-9]{3}
date	Date	Date
attendance	Attendance	varchar(3)
reason_for_absence	Reason for absence	varchar(100)

Description of Attendance Management System :

The entity is a concept or object in which the piece of information can be stored. There are three types of relations between entities. They are as follows:

● One to One(1-1): This relationship specifies that one instance of an entity is associated with another instance of an entity.

● One to Many(1-N): This relationship specifies that one instance of an entity is associated with zero or many other instances of another entity.

● Many to Many(M-N): This relationship specifies that one instance of an entity is associated with zero or many other instances of another entity.

Description of the entity relationship in ER-Diagram:

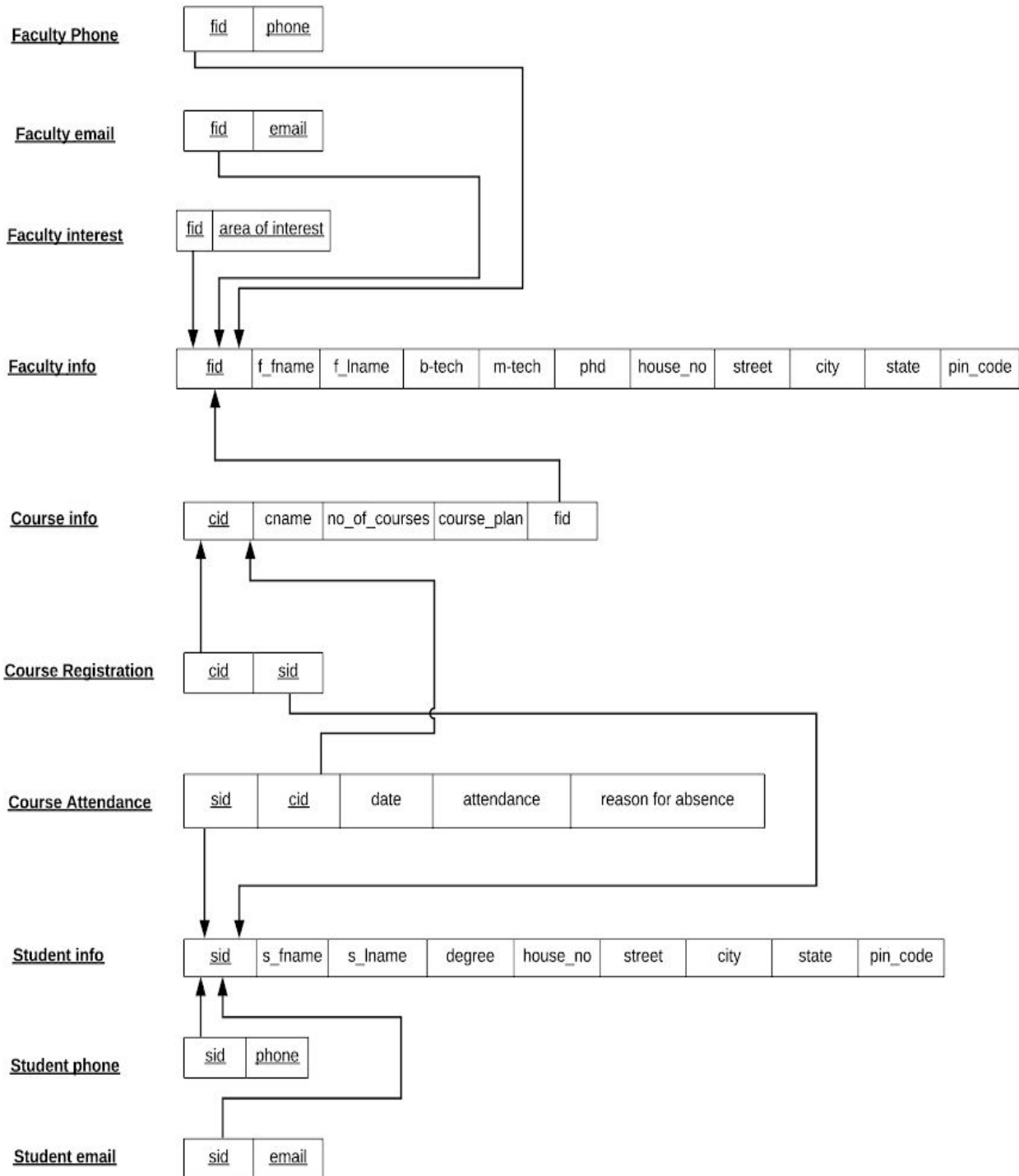| Entity 1	Entity 2	Relationship Name	Cardinality Ratio	Description
Courses	Faculty	Taken by	N:1	One faculty can take many courses.
Courses	Student	Registers for	N:M	One student can register for many courses.. One course can be registered by many students.

Courses	Course Attendance	Attendance Status	M:N	One course can have many attendance entries. An attendance can be updated by many courses.
Student	Course Attendance	Attendance for courses	M:N	One student can make many attendance entries. An attendance can be updated by many students.

3. B   RELATIONAL SCHEMA

In this section we give a description of Mapping of ER diagram with Relation schema along with schema diagram of Attendance Management System.

A relational database schema is the tables, columns and relationships that make up a relational database. There are two steps to creating a relational database schema: creating the logical schema and creating the physical schema. The logical schema depicts the structure of the database, showing the tables, columns and relationships with other tables in the database and can be created with modeling tools or spreadsheet and drawing software. The physical schema is created by actually generating the tables, columns and relationships in the relational database management software (RDBMS). Most modeling tools can automate the creation of the physical schema from the logical schema, but it can also be done by manually.

Online Attendance Management System (Relational Schema)

Faculty Phone

fid	phone

Faculty email

fid	email

Faculty interest

fid	area of interest

Faculty info

fid	f_fname	f_lname	b-tech	m-tech	phd	house_no	street	city	state	pin_code

Course info

cid	cname	no_of_courses	course_plan	fid

Course Registration

cid	sid

Course Attendance

sid	cid	date	attendance	reason for absence

Student info

sid	s_fname	s_lname	degree	house_no	street	city	state	pin_code

Student phone

sid	phone

Student email

sid	email

The application Attendance Management System consists of the entities described below. Some entities are related to other entities with the help of primary key - foreign key pair. The foreign key is used in establishing relations with the other table. Hence it is called a relational database system.

1. Faculty info

Attributes: fid, f_fname, f_lname, b-tech, m-tech, phd, house_no, street, city, state, pin-code, phone, email, area of interest

Primary key: fid

Foreign Key:

2. Student info

Attributes: sid, s_fname, s_lname, degree, house_no, street, city, state, pin_code, phone, email

Primary key: sid

Foreign Key:

3. Course info

Attributes: cid, cname, no of courses, course plan

Primary Key: cid

Foreign Key: fid

4. Course Registration

Attributes: cid, sid

Primary Key: (cid, sid)

Foreign Key: cid, sid

5. Course Attendance

Attributes: cid, sid, date, attendance, reason for absence

Primary Key: (cid, sid)

Foreign Key: cid, sid

3.C Key Constraints

Entity	Primary key	Foreign key
Faculty info	fid	
Student info	sid	
Course info	cid	fid
Course Registration	(cid, sid)	cid, fid
Course Attendance	(cid, sid)	cid, fid

3.D Participation Constraints in Relation Schema

Entity 1	Entity 2	Relationship Name	Participation Constraint	Description
Courses	Faculty	Taken by	1:0	Each course must be taken by a faculty. But there can be faculty who haven't taken any course.
Courses	Student	Registers for	0:0	Not every student registers for the course. Not every course id registered by students.
Courses	Course Attendance	Attendance Status	1:1	Every course must have an attendance entry. Every attendance is updated for a course.
Student	Course Attendance	Attendance for courses	0:1	Each course attendance must be updated for a student. But there can be student with no attendance entry.

4. MODULES

4.1 Generation of Student / Faculty profile :

The attendance management system admin is responsible for creating verified accounts for the student and faculty members. The details and ID of each user is entered during profile creation. The password entered by the user is converted to a hash and stored in the database so that even if the database is compromised, external entity cannot access the user's details as a hash cannot be traced back to the password the user entered.

4.2 Login as Student / Faculty :

After the admin creates profiles for students and faculty members, they can login to access their personal portals using their individual credentials and they will be redirected to their respective pages. The django-auth system takes care of authenticating users and verifying passwords. Passwords are stored in a secure manner using hashing.

4.3 Home page :

This page is accessible without logging in. It contains links to student and faculty portals as well as a course archive that lists the courses currently being taught in that institute. It also displays the features of the system and provides links to navigate to the social media pages (facebook, instagram, linkedin etc.) of that institute.

4.4 Student portal :

This module displays the details of the student who logged in. It also contains a list of courses the student has registered for, which in turn navigate him/her to attendance list of that particular course, to check attendance status.

4.5 Faculty portal :

This module displays the details of the faculty who logged in. It is used by the faculty member to give attendance to a particular student in a particular course that is taken up by the faculty member.

4.6 Course Archive :

It is a page that contains the list of courses being taught in that educational institute. Each course can be clicked on, to see the details of that particular course (course id, course name, course instructor, course plan etc.)

4.7 Attendance list :

This page is accessed by students from their respective portals, when they want to check their attendance for the courses they have registered.

5. TOOLS

5.1 FRONT END TOOLS

The application framework we have used is Django. It is a free and open-source web framework written in Python, which follows the Model-View-Template (MVT) architectural pattern. Django's primary goal is to ease the creation of complex, database-driven websites. Python is used throughout, even for settings files and data models.

The <u>Model</u> maintains the relationship between the objects and the database and handles validation, association, transactions, and more.

<u>View</u> is the facility within the application that directs traffic, on the one hand, querying the models for specific data, and on the other hand, organizing that data (searching, sorting, messaging it) into a form that fits the needs of a given template.

<u>Template</u> is a presentation of data in a particular format, triggered by a view's decision to present the data. For frontend design we have used HTML, CSS and Bootstrap. To introduce some dynamic elements in our application we have used Javascript and jQuery.

File Structure for frontend is as follows:

Templates : All the HTML,CSS,JavaScript files are stored in this folder. These are used to render the web pages when a particular view calls them.

Models.py: This file contains the definition of the structure of the Attendance management system database. It defines all the entities and its attributes, along with primary - foreign key pairs.

Views.py: Variable declaration , querying the database etc. is done in this file. The functionality for each URL is defined here

5.2 BACKEND TOOLS

MySQL is a relational database management system (RDBMS).

Data in the database is organised into tables, each table being organised as rows and columns. Each row in a table is called a record and each column in a table is called a field.

MySQL allows us to insert, retrieve, modify or delete records.

MySQL is a DBMS that relates information stored in one table to information stored in another table by looking for elements common to each of them. It has efficient storage and retrieval mechanisms for data and uses normalisation process during design of RDBMS.

Features of MySQL :

> Speed : MySQL runs very fast and supports clustered servers for demanding applications.

> Ease of use : MySQL is a high-performance, relatively simple database system. It can be configured, monitored and managed from command line.

> Capability : MySQL is a multi-threaded SQL server that supports different backends, client programs, libraries, administrative tools and programming interfaces.

> Connectivity : MySQL is fully networked and can be accessed from anywhere on the internet.

> Portability : MySQL runs on many UNIX and non-UNIX systems (eg. Windows). It runs on hardware from PC to high-end server.

6. CONCLUSION

This online Attendance Management System has 3 modules for proper functioning:

1. Admin : Has the rights to create a new entry for faculty, student and course details
2. Student : Can check attendance through his/her portal
3. Faculty : Can give attendance through his/her portal

The system is designed in such a way that only authorised people are allowed to access modules. Records of database are modified by the admin only. The User is always in control of the application and not vice versa.

The user interface is consistent so that user can handle the application with ease and speed.

The application is visually and conceptually clear.

7. RESULTS

Home Page :
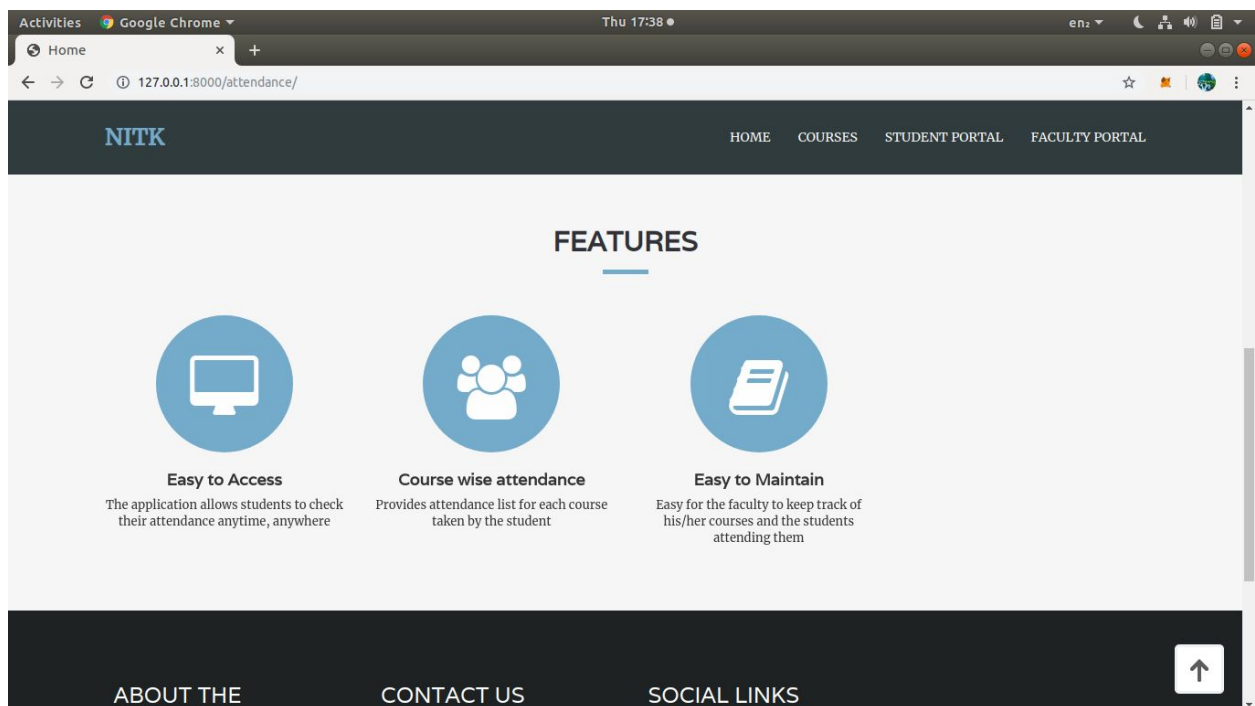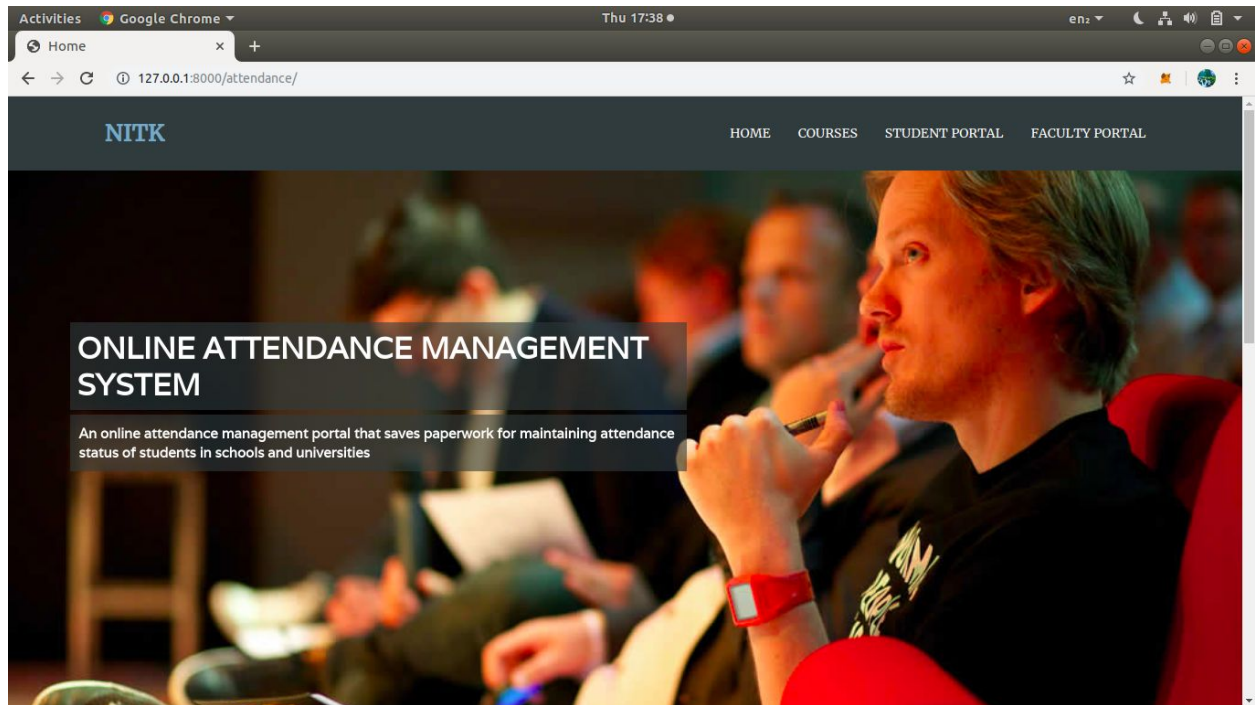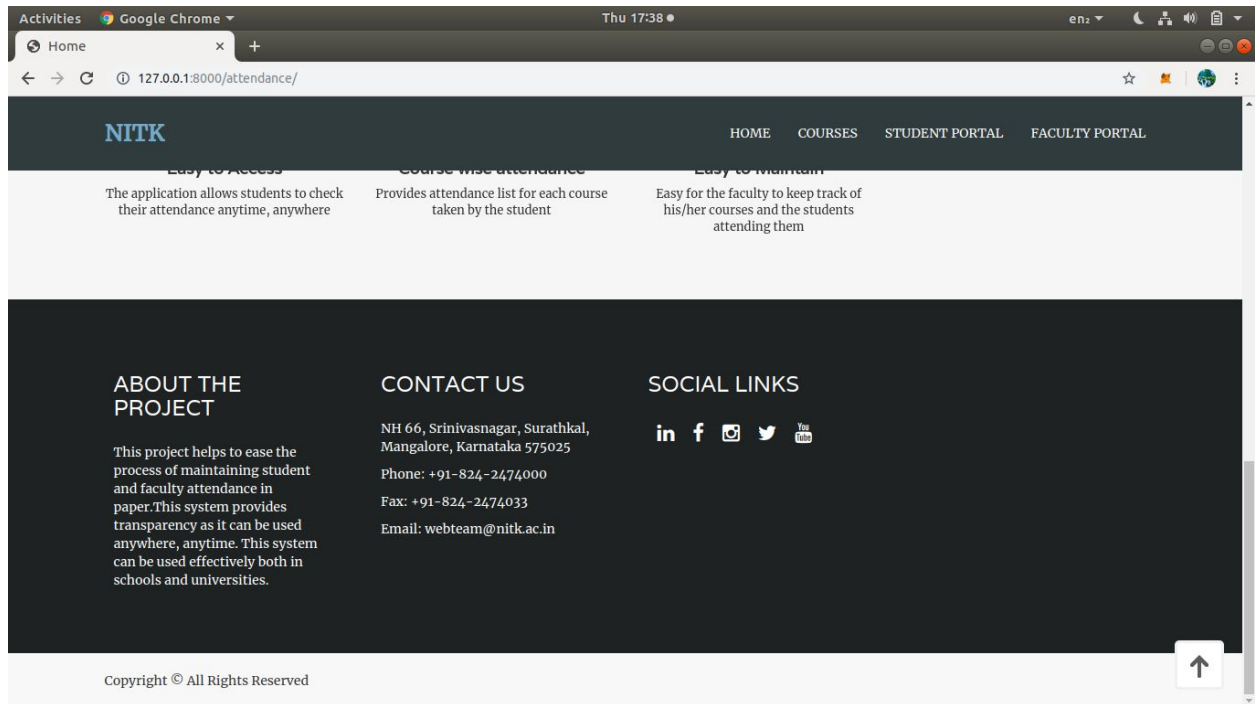
Course Archive Page :
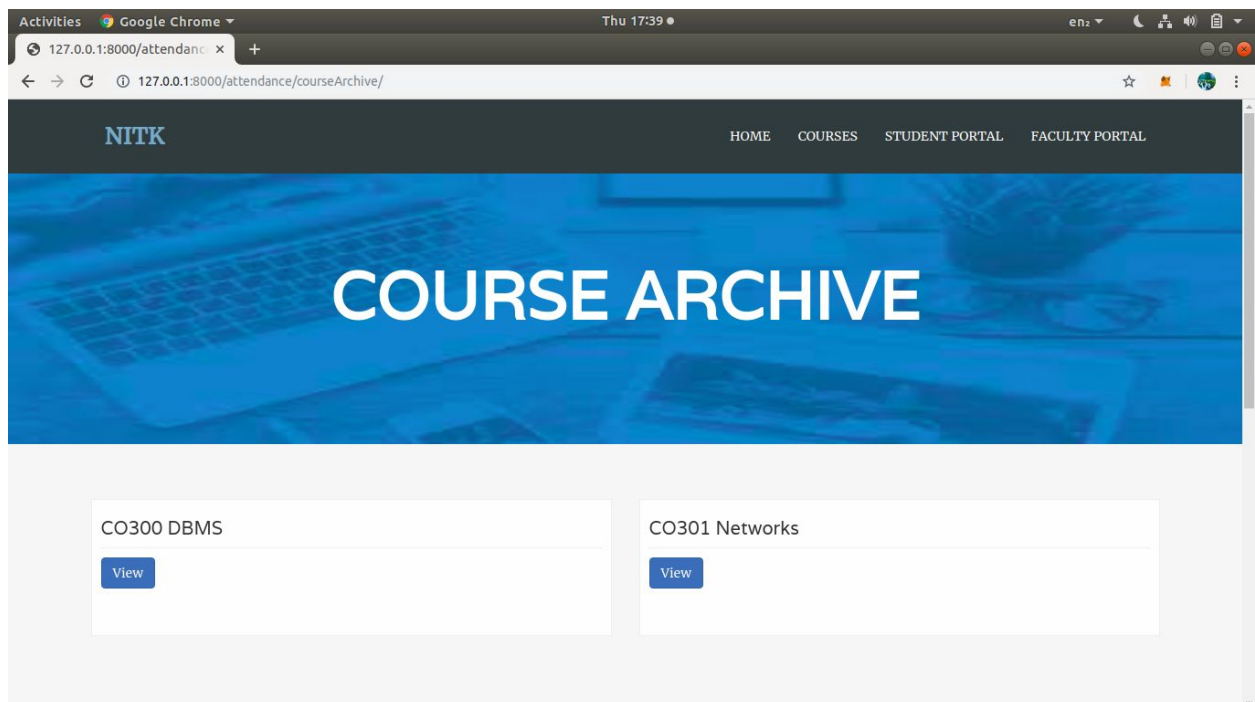
Course Details Page:
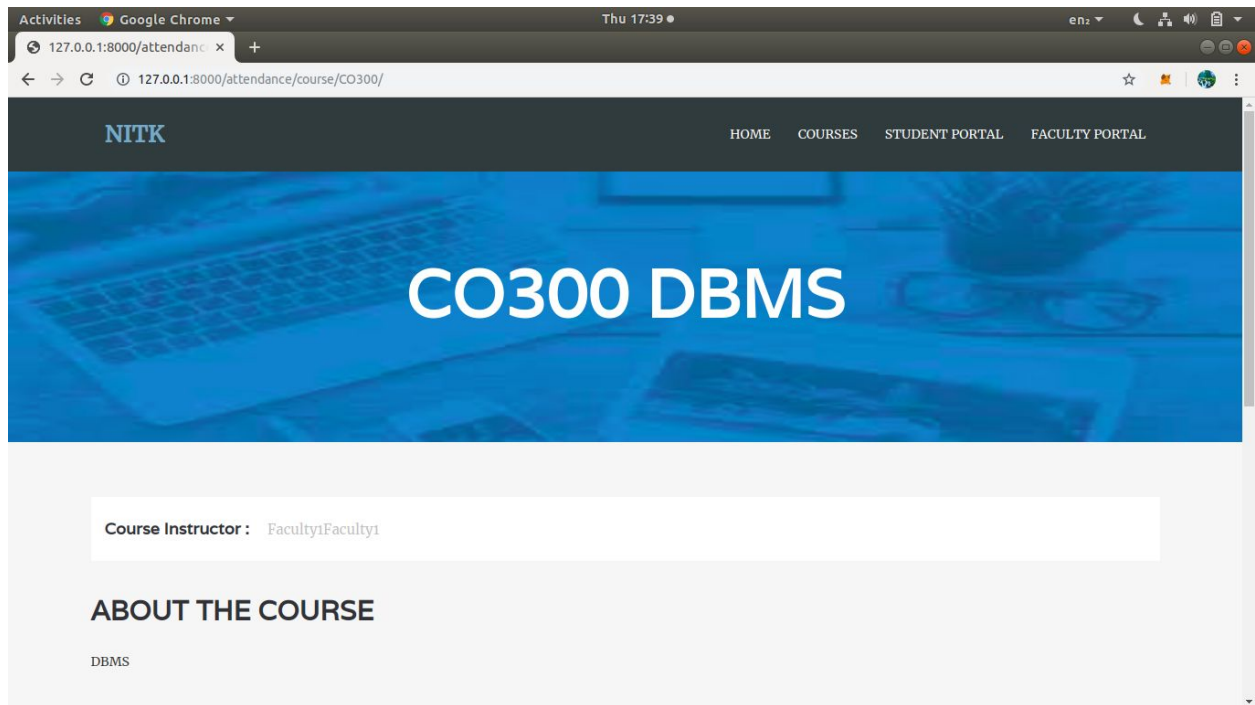
Student Login :
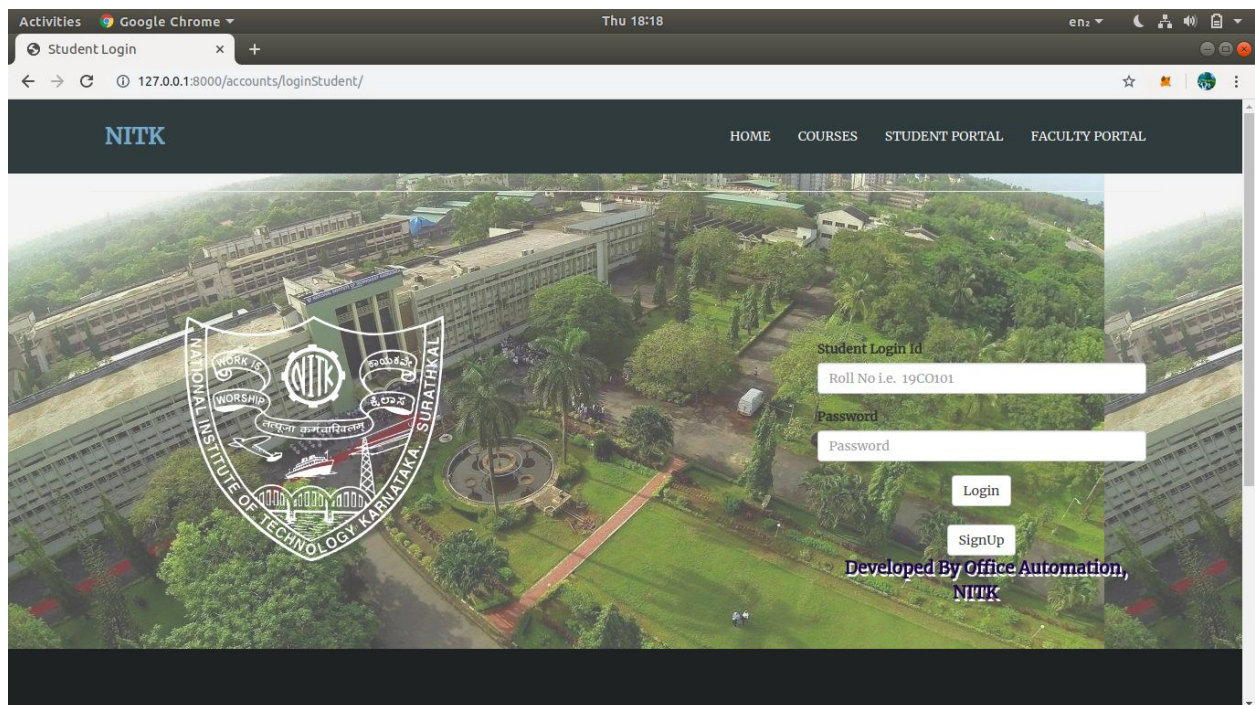
Student SignUp :

Student Portal :

Attendance List :

Faculty Login :

Faculty SignUp :

127.0.0.1:8000/accounts/ ×   +

← → C   ① 127.0.0.1:8000/accounts/signupFaculty/

NITK   HOME   COURSES   STUDENT PORTAL   FACULTY PORTAL

Area of interest:

This field is required.

Btech:

This field is required.

Mtech:

This field is required.

Phd:

This field is required.

House no:

This field is required.

Street:

This field is required.

127.0.0.1:8000/accounts/ ×   +

← → C   ① 127.0.0.1:8000/accounts/signupFaculty/

NITK   HOME   COURSES   STUDENT PORTAL   FACULTY PORTAL
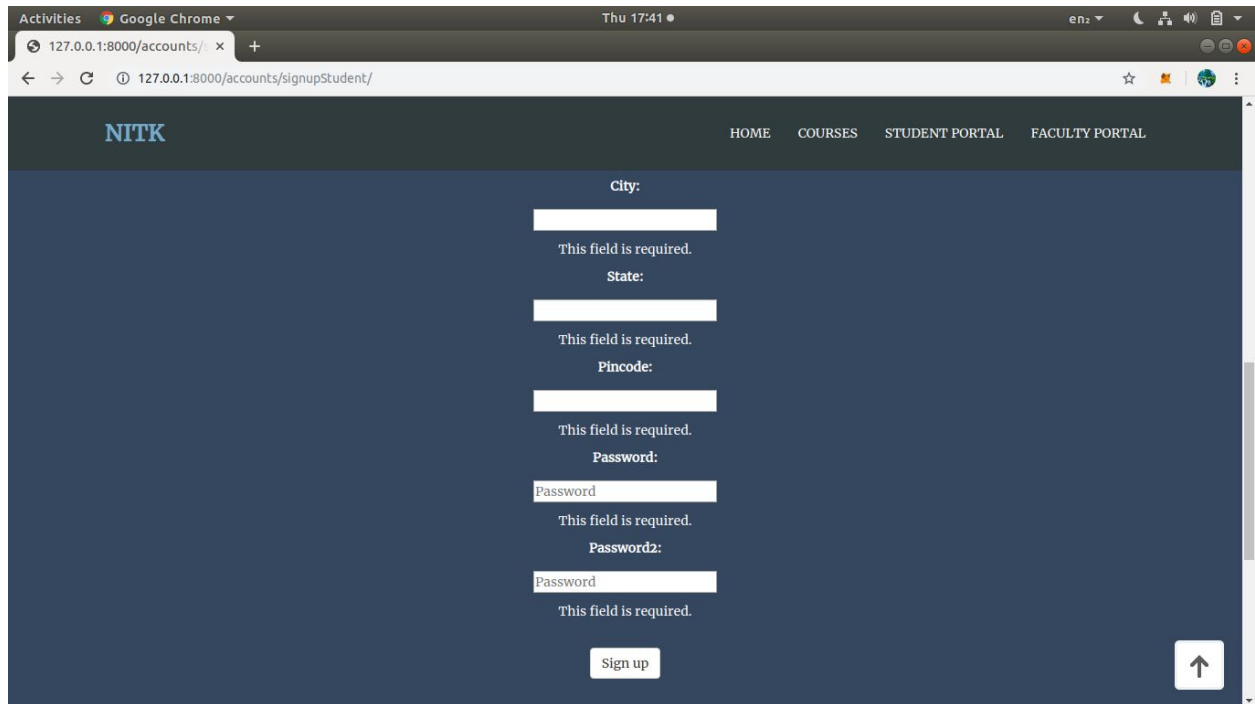
City:

This field is required.
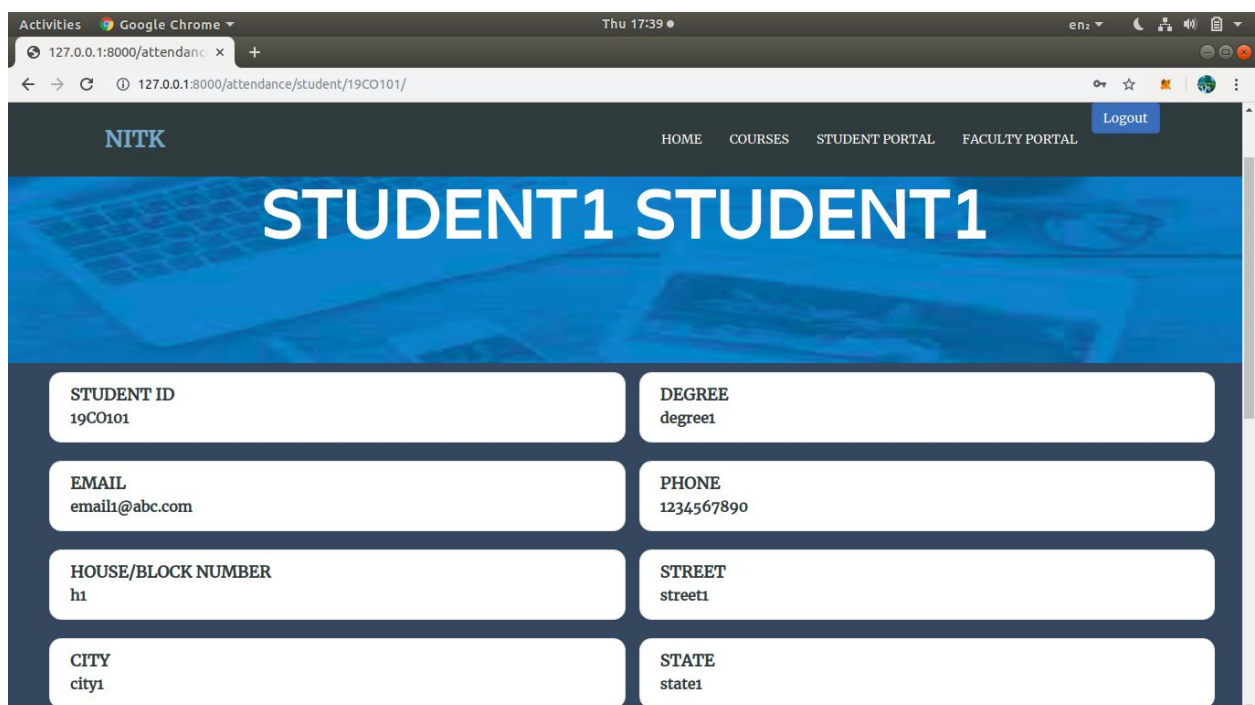
State:

This field is required.

Pincode:

This field is required.
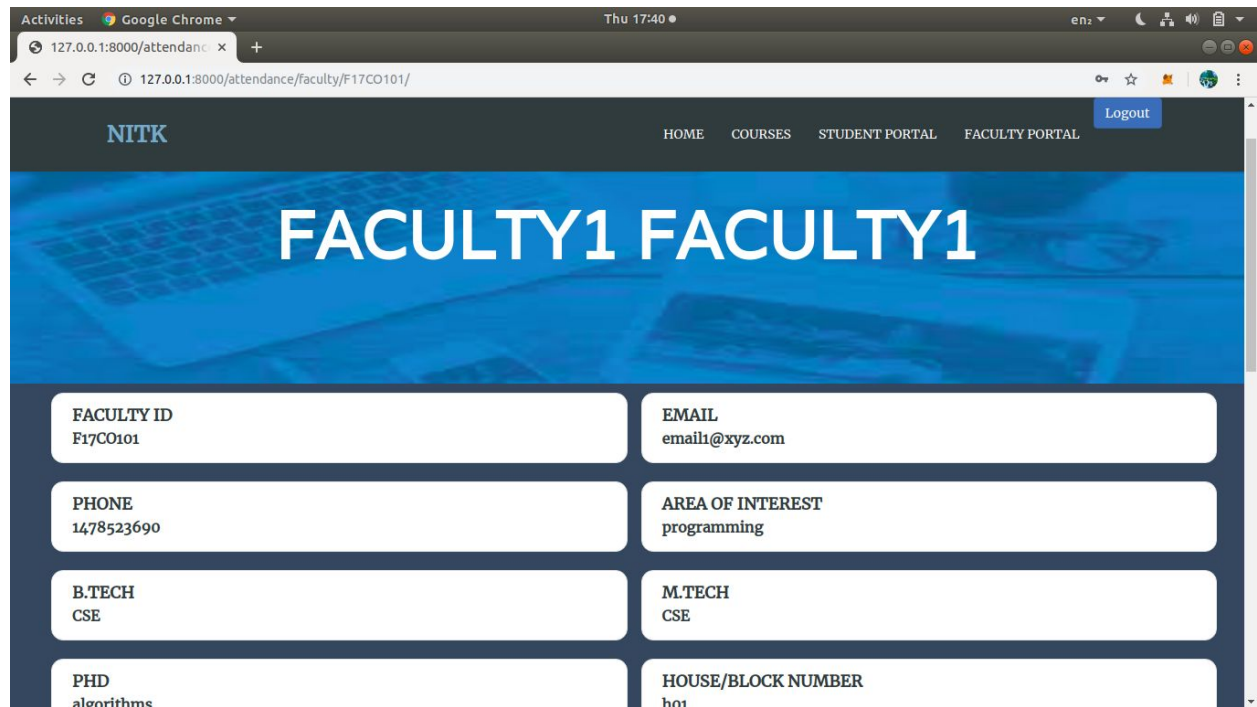
Password:

Password

This field is required.

Password2:

Password

This field is required.

Sign up

Faculty Portal :
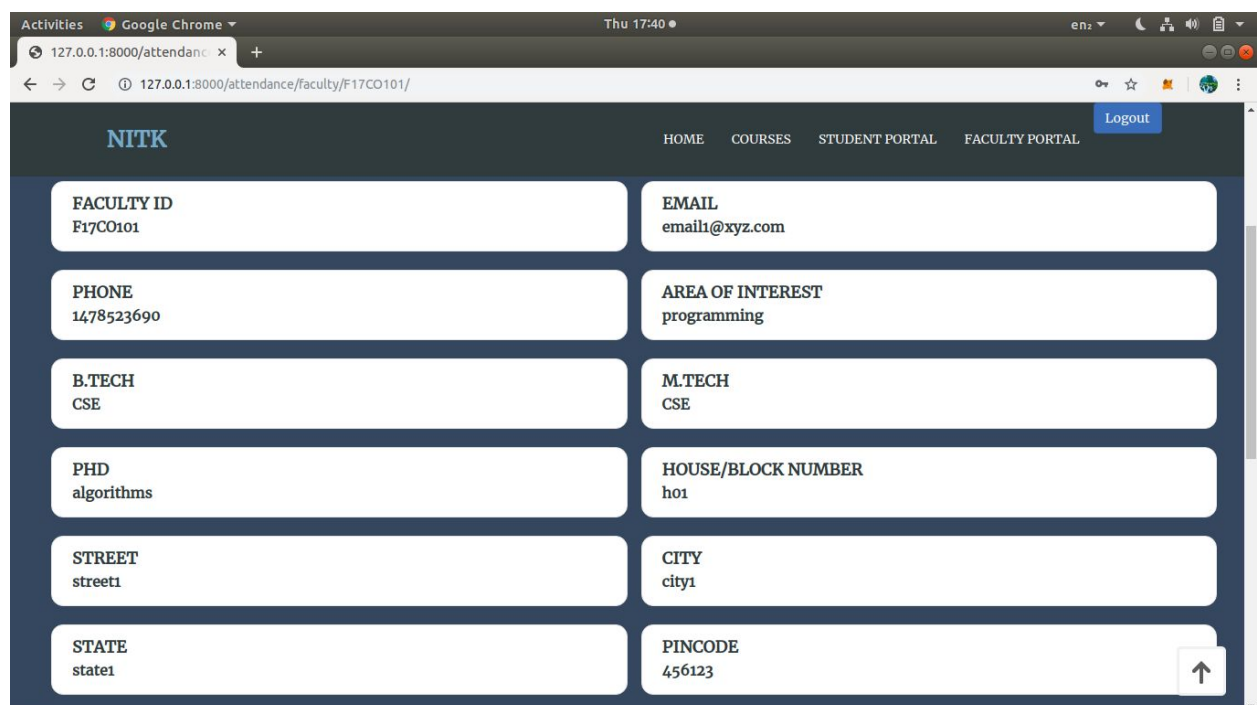
127.0.0.1:8000/attendance ×   +

← → C   ⓘ 127.0.0.1:8000/attendance/faculty/F17CO101/   🔑 ☆ 🐱 🌐 ⋮

NITK   HOME   COURSES   STUDENT PORTAL   FACULTY PORTAL   Logout

PHD
algorithms

HOUSE/BLOCK NUMBER
h01

STREET
street1

CITY
city1

STATE
state1

PINCODE
456123

CO300

Enter cid

Enter sid

Enter attendance

Give Attendance   ↑