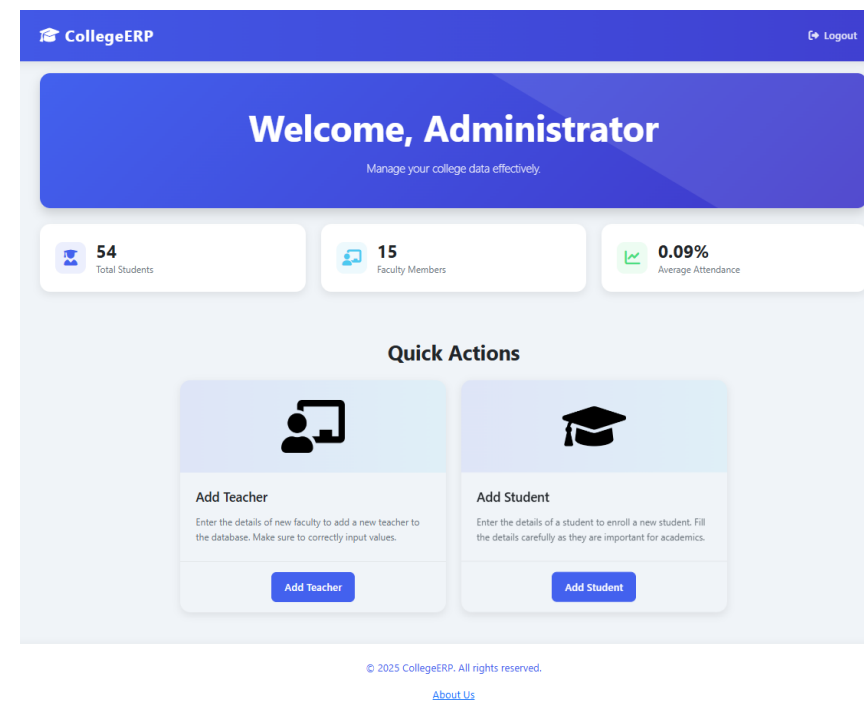


Leon Lobo
Jay Vankar
Jay Kapadia



College-ERP (Education Records Portal)



The simplest way to manage
information.

A Website which can:-

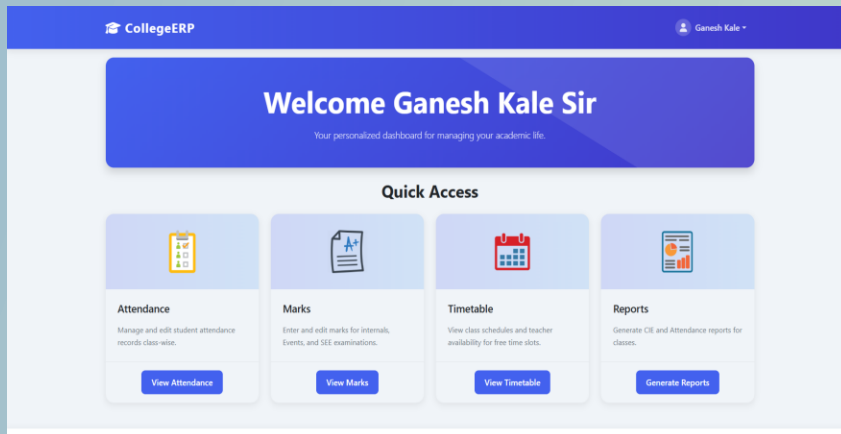
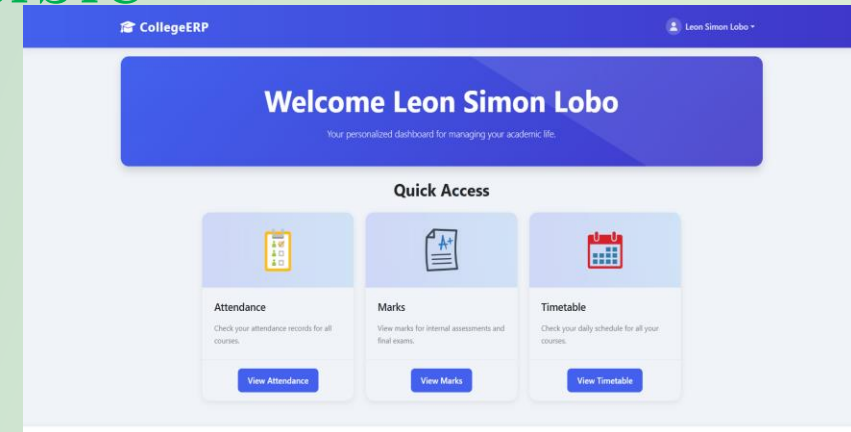
- View TimeTable of students and teachers
- Seamlessly check attendance and update it for any day and any class
- Cancel and add classes whenever you want
- Check and update marks at any time
- View each students personal attendance and marks
- Generate a report for all students in class

Can be accessed by Student, Teacher or Admin



- Can add new Students or Teachers as admin
- Total no. Of students, faculty and overall attendance is also visible

- ❑ Check attendance and marks of each subject
- ❑ View Timetable for the week



- ❖ Add extra classes, mark attendance and give scores for test
- ❖ Generate report for class

Simple, Clean and Effective UI

- Header which shows website name and dropdown menu to logout.
- A very simplistic sidebar which provides options to directly go to any page you desire without any extra fuss.

Course ID	Course name	Attended classes	Total classes	Attendance %	Classes to attend
AI102	Data Structures	1	2	50.0	2
AI104	Web Programming and Python	0	0	0	0
EC106	Digital Electronics and Logic Design	0	0	0	0
EG110	Energy and Environmental Engineering	0	0	0	0
MA106	Linear Algebra and Statistics	0	0	0	0

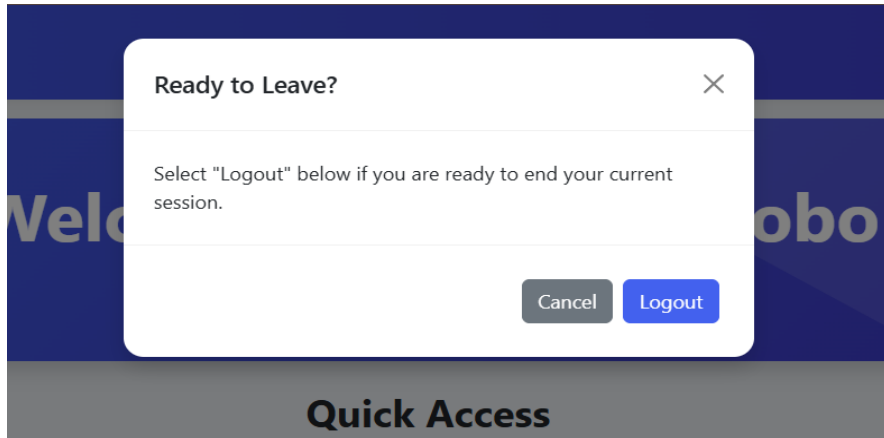
© 2025 CollegeERP. All rights reserved.
[About Us](#)

- Light colors and consistent theme.
- Bare but effective footer.

Login and Logout

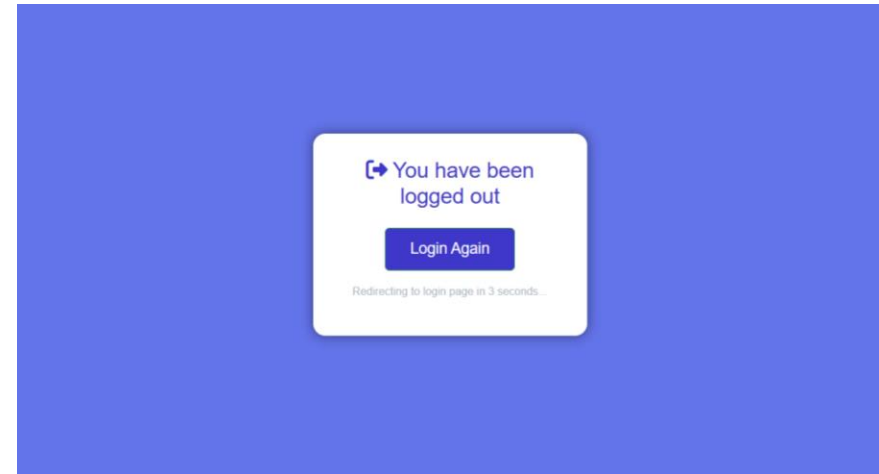


- Only asks for username and password, no extra info



- A pop-up for confirmation

- Log out page which after 3 seconds automatically redirects to login





The Architecture

- **Frontend:**
 - Responsive web pages using HTML, CSS, and Bootstrap.
 - Timetable grid and course display.
- **Backend:**
 - Django framework for managing URLs, views, and templates.
 - CRUD functionality for managing timetable entries.
 - Authentication and session management.
- **Database:**
 - SQLite or MySQL to store data for courses, teachers, and timetable assignments.

Login/Logout Functionality

django.contrib.auth is
django's built-in
authentication system which
contains views for log in and
logout

```
from django.contrib.auth import views as auth_views

path('accounts/login/',
     auth_views.LoginView.as_view(template_name='info/login.html'), name='login'),
path('accounts/logout/',
     auth_views.LogoutView.as_view(template_name='info/logout.html'), name='logout'),
```

Only HTML pages of login and logout have to be made
which are mostly frontend, however,

```
<form method="post" action="/accounts/logout/" style="display:inline;">
  {% csrf_token %}
  <button type="submit" class="btn btn-primary">Logout</button>
</form>
```

- ✓ Login and logout must be done using POST http request, not GET which is standard for urls
- ✓ This is because of safety issues and updation of pages/data

Database Model

CollegeERP Leon Simon Lobo ▾

Home Attendance Attendance By Subject Marks Time Table About Us

Timetable

	8:30 - 9:20	9:30 - 10:20	10:30 - 11:20	11:30 - 12:20	Lunch	2:00 - 4:00	4:00 - 6:00
Monday		AI102	EG110	AI104		EG110	AI102
Tuesday	MA106	EG110	AI102			EC106	
Wednesday	AI102	EG110	EC106	MA106			
Thursday	AI104	EC106	MA106			AI104	
Friday	EC106	AI102	AI104	MA106			

© 2025 CollegeERP. All rights reserved. [About Us](#)

Django model field(tuple of tuples):

```
time_slots = (
    ('8:30 - 9:20', '8:30 - 9:20'),
    ('9:30 - 10:20', '9:30 - 10:20'),
    ('10:30 - 11:20', '10:30 - 11:20'),
    ('11:30 - 12:20', '11:30 - 12:20'),
    ('2:00 - 4:00', '2:00 - 4:00'),
    ('4:00 - 6:00', '4:00 - 6:00'),
)

DAYS_OF_WEEK = (
    ('Monday', 'Monday'),
    ('Tuesday', 'Tuesday'),
    ('Wednesday', 'Wednesday'),
    ('Thursday', 'Thursday'),
    ('Friday', 'Friday'),
)
```

Django class
timetable view:

```
@login_required()
def timetable(request, class_id):
    asst = AssignTime.objects.filter(assign__class_id=class_id)
    matrix = [['' for i in range(8)] for j in range(6)]

    for i, d in enumerate(DAYS_OF_WEEK):
        t = 0
        for j in range(8):
            if j == 0:
                matrix[i][0] = d[0]
                continue
            if j == 5:
                continue
            try:
                a = asst.get(period=time_slots[t][0], day=d[0])
                matrix[i][j] = a.assign.course_id
            except AssignTime.DoesNotExist:
                pass
            t += 1

    context = {'matrix': matrix}
    return render(request, 'info/timetable.html', context)
```

HTML
Table:

```
<table class="col-sm-12 table table-bordered table-striped table-condensed cf">
  <thead class="cf">
    <tr style="color: var(--purp);">
      <th> </th>
      <th>8:30 - 9:20</th>
      <th>9:30 - 10:20</th>
      <th>10:30 - 11:20</th>
      <th>11:30 - 12:20</th>
      <th>Lunch</th>
      <th>2:00 - 4:00</th>
      <th>4:00 - 6:00</th>
    </tr>
  </thead>
  <tbody>
    {% for i in matrix %}
      <tr>
        {% for j in i %}
          {% if forloop.counter == 1 %}
            <td style="color: var(--primary);"><b>{{ j }}</b></td>
          {% else %}
            <td style="color: var(--danger);">{{ j }}</td>
          {% endif %}
        {% endfor %}
      </tr>
    {% endfor %}
  </tbody>
</table>
```


Challenges Faced	Solutions
Timetable slot mapping issues (index errors, etc.)	Refined the time slot structure and handled edge cases properly.
Old records not matching updated models (test names)	Implemented bulk update scripts to synchronize old database records.
Injecting mass data using admin page	Implementing buttons and features to make it easier
Tricky Layout Design	Implementation of Tailwind

Concluding this Presentation..

🎓 CollegeERP aims to ease the life of students and teachers by consolidating all their information into one simple place

We hope you enjoyed our website! Feel free to try a demo and experience a little less stress for a change

Developers:
Leon(Frontend) Jay V.(Backend) Jay K.(Backend)