**SOFTWARE ENGINEERING PROJECT REPORT**

**SMART CAMPUS MANAGEMENT SYSTEM**

**Student Name**

Astha Patel

**University’s Name**

Manhattan College

**Date of Submission**

12/10/2024

**Table of Content:**

[**1. Abstract** 3](#_Toc184479715)

[**2. Introduction** 3](#_Toc184479716)

[**3. Discussion** 3](#_Toc184479717)

[**4. Functional requirements** 4](#_Toc184479726)

[**4.1 Functional requirement 1**](#_Toc184479727)

[**4.2 Functional requirement 2**](#_Toc184479728)

[**4.3 Functional requirement 3**](#_Toc184479729)

[**4.4 Functional requirement 4**](#_Toc184479730)

[**5. Non-functional requirements** 14](#_Toc184479732)

[**5.1 Non-functional requirement 1**](#_Toc184479733)

[**5.2 Non-functional requirement 2**](#_Toc184479734)

[**5.4 Non-functional requirement 3**](#_Toc184479735)

[**6. System Functional requirements** 16](#_Toc184479738)

[**7. System Non-functional requirements** 16](#_Toc184479738)

[**8. Domain Requirements** 17](#_Toc184479742)

[**9. Application of IEEE/ACM code of Ethics** 17](#_Toc184479742)

[**10. Conclusion** 18](#_Toc184479745)

[**11. References** 19](#_Toc184479746)

**Abstract:**

The SmartCampus Management System (SCMS) is designed to make campus management easier and more efficient for students, teachers, and staff. It focuses on simplifying tasks like managing courses, student profiles, and administrative work. This document explains what the system should do, how it should perform, and why each feature is important. It also ensures the system follows ethical guidelines based on the IEEE/ACM Code of Ethics, supporting responsible and fair development practices

**Introduction:**

The Smart Campus Management System (SCMS) is developed to facilitate the efficient management of campus activities, enhancing the experience for students, faculty, and administration.

The system aims to streamline key functions such as course management, student profiles, and administrative processes. This specification document outlines the functional and non-functional requirements for the users and the system, ensuring clarity on what the system is expected to deliver. Additionally, the document covers possible domain requirements and provides a rationale for each user functional requirement.

The IEEE/ACM Code of Ethics is also applied to ensure that ethical standards are followed throughout the specification and development process.

For the creation of this project I used HTML(Hyper text markup language), CSS(cascading style sheet), Bootstrap, Python, Jinja2(Python framework) and for database storing I used pgAdmin4.

**Discussion:**

* System Efficiency:

The SmartCampus Management System has been designed to streamline campus operations, offering key functionalities like secure login, real-time notifications, and course management. The system effectively manages students, faculty, and administrators' needs, enhancing overall user experience. However, its scalability and performance may be impacted when dealing with very large user bases or complex requests.

* System Limitations:

Although the system meets current requirements for scalability and performance, its responsiveness may degrade as user numbers rise beyond the designed threshold of 10,000 users. Future upgrades to cloud infrastructure may be necessary to ensure seamless performance as the system grows.

* Future Enhancements:

To further enhance the system's capabilities, future integration of artificial intelligence for personalized notifications and analytics could improve user engagement. Additionally, implementing features like smart classroom management and integrating IoT-based campus monitoring tools will increase operational efficiency.

* Real-Time Notification System:

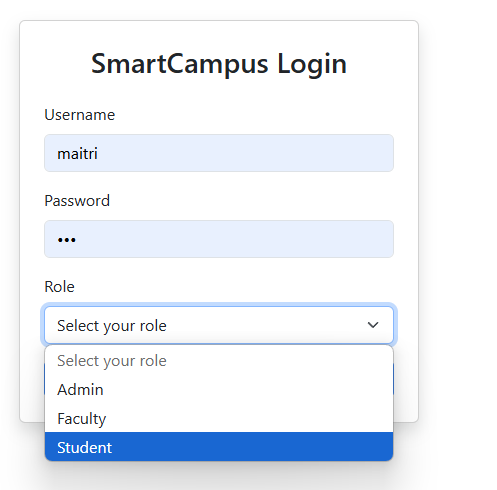
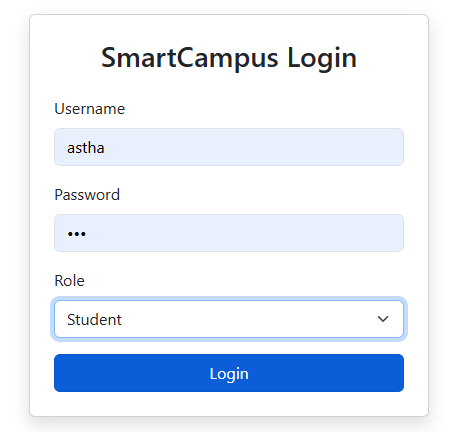
The real-time notification feature has been a major asset, keeping students informed of deadlines, class schedules, and other important events. While effective, future iterations might include AI-driven suggestions for more personalized notifications based on user activity and preferences

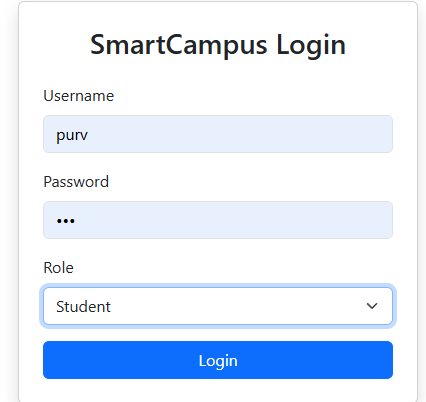
**Functional requirements:**

1. **User Functional Requirement 1:**

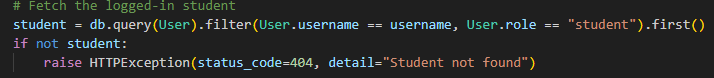
**The system shall allow students to log in securely using their unique student ID and password.**

* + For example, here as we can see three students are able to login with their unique ID and password.

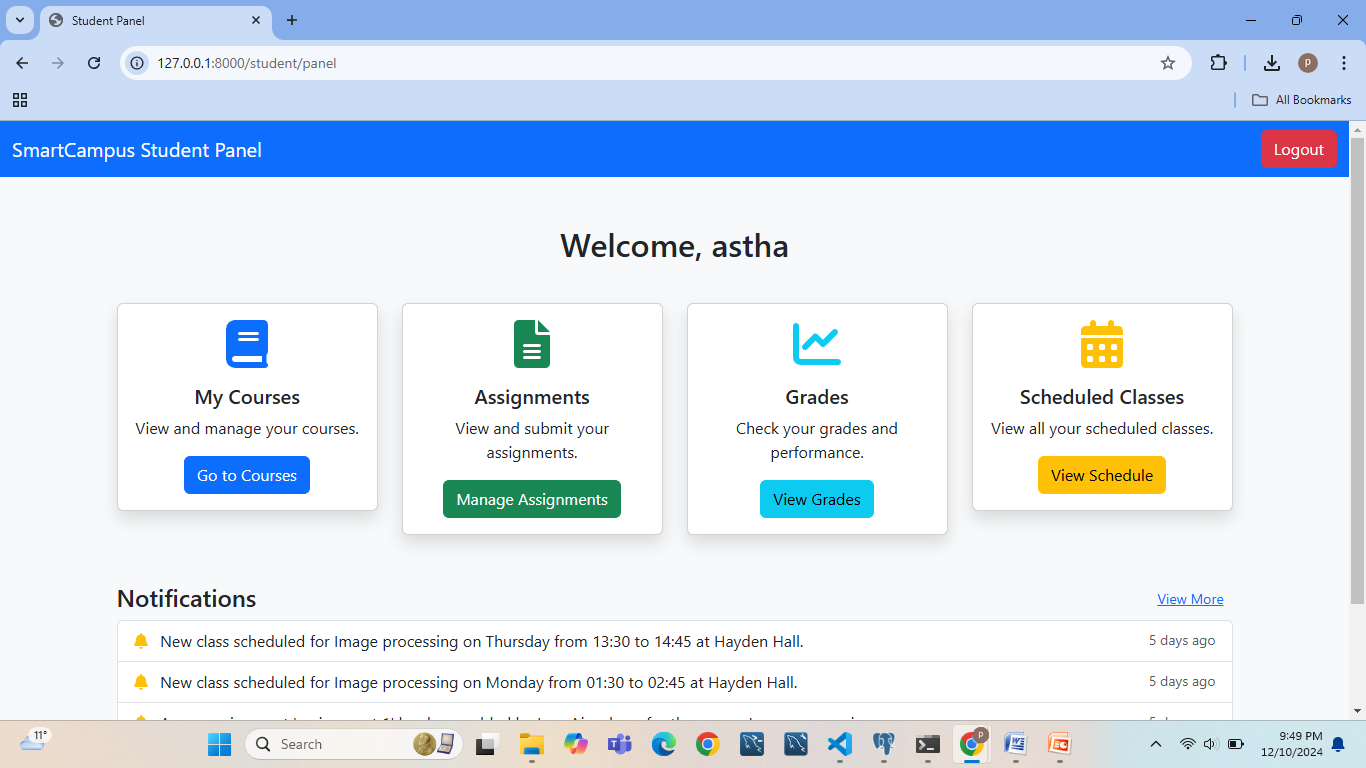
 



* + Following is the code which identify particular student and login to his/her particular profile.



* + After login, student profile will look like this where student can see My courses through which he/she is able to see enrolled courses, Number of Assignments, Grades for all subjects, Scheduled Classes and notification of all assignments, new upcoming events and all.



The code to achieve this is following:

<!-- Courses -->

            <div class="col-md-3 mt-3">

                <div class="card text-center shadow">

                    <div class="card-body">

                        <i class="fas fa-book fa-3x text-primary mb-3"></i>

                        <h5 class="card-title">My Courses</h5>

                        <p class="card-text">View and manage your courses.</p>

                        <a href="/student/courses" class="btn btn-primary">Go to Courses</a>

                    </div>

                </div>

            </div>

            <!-- Assignments -->

            <div class="col-md-3 mt-3">

                <div class="card text-center shadow">

                    <div class="card-body">

                        <i class="fas fa-file-alt fa-3x text-success mb-3"></i>

                        <h5 class="card-title">Assignments</h5>

                        <p class="card-text">View and submit your assignments.</p>

                        <a href="/student/assignments" class="btn btn-success">Manage Assignments</a>

                    </div>

                </div>

            </div>

            <!-- Grades -->

            <div class="col-md-3 mt-3">

                <div class="card text-center shadow">

                    <div class="card-body">

                        <i class="fas fa-chart-line fa-3x text-info mb-3"></i>

                        <h5 class="card-title">Grades</h5>

                        <p class="card-text">Check your grades and performance.</p>

                        <a href="/student/grades" class="btn btn-info">View Grades</a>

                    </div>

                </div>

            </div>

            <div class="col-md-3 mt-3">

                <div class="card text-center shadow">

                    <div class="card-body">

                        <i class="fas fa-calendar-alt fa-3x text-warning mb-3"></i>

                        <h5 class="card-title">Scheduled Classes</h5>

                        <p class="card-text">View all your scheduled classes.</p>

                        <a href="/student/scheduled-classes" class="btn btn-warning">View Schedule</a>

                    </div>

                </div>

            </div>

        </div>

        <!-- Notifications -->

        <div class="mt-5">

            <h4>

                Notifications

                <a href="/student/notifications" class="btn btn-link btn-sm float-end">View More</a>

            </h4>

            <ul class="list-group" id="notification-list">

                {% for notification in notifications[:3] %}

                <li class="list-group-item">

                    <i class="fas fa-bell text-warning me-2"></i>

                    {{ notification.message }}

                    <small class="text-muted float-end" data-timestamp="{{ notification.created\_at }}">{{ notification.created\_at }}</small>

                </li>

                {% else %}

                <li class="list-group-item text-center">No notifications.</li>

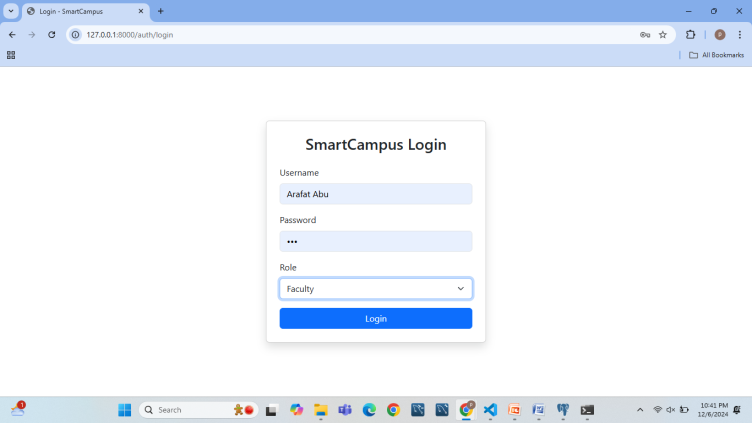
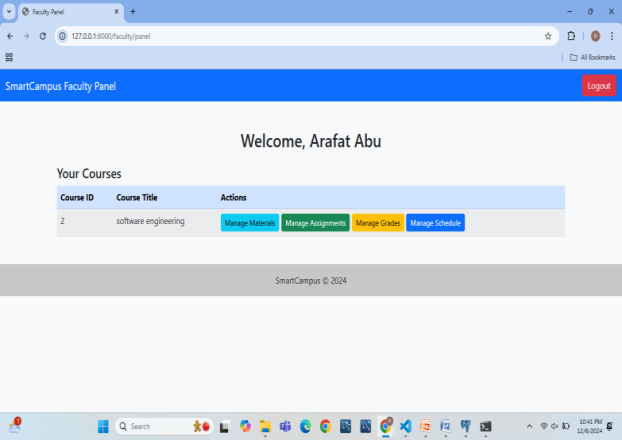
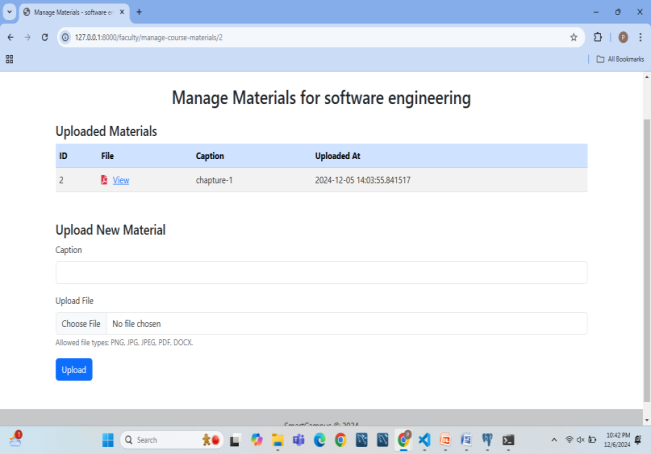
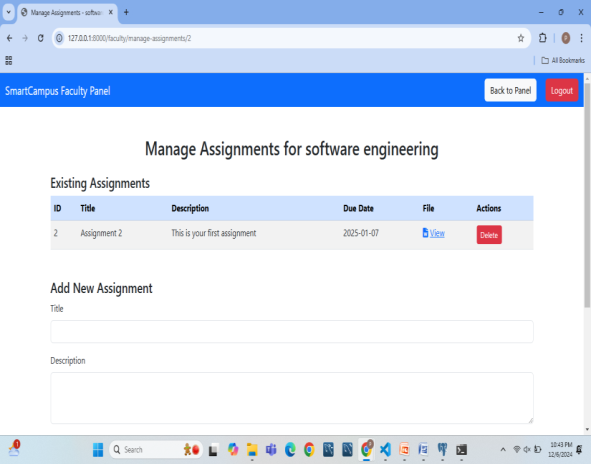
                {% endfor %}

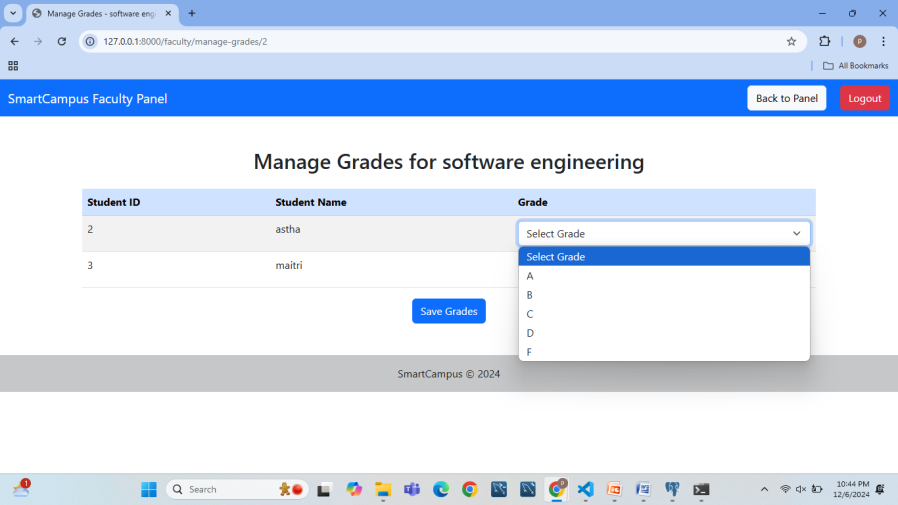
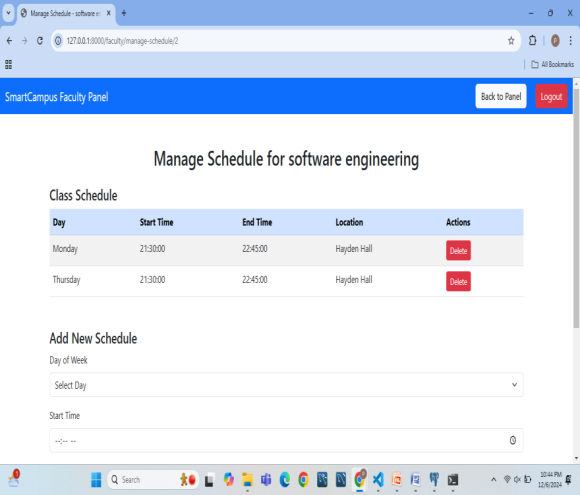
            </ul>

        </div>

    </div>

1. **User Functional Requirement 2**: Faculty members shall be able to upload and manage course materials, assignments, and grades.
   * Following is the screenshot to show faculty page access ad their rights like they can upload and manage course materials, assignments, schedules and grades.

* + Following is the code to provide secure login for professors:

# Fetch the logged-in faculty member

    faculty = db.query(User).filter(User.username == username, User.role == "faculty").first()

    if not faculty:

        raise HTTPException(status\_code=404, detail="Faculty member not found")

* + Then by using this code faculty can access the subjects they are going to teach:

# Fetch courses allocated to this faculty member

    courses = db.query(Course).filter(Course.faculty\_id == faculty.id).all()

    return templates.TemplateResponse(

        "faculty\_panel.html",

        {"request": request, "faculty": faculty, "courses": courses},

    )

* + What faculty page will include can be seen by following code:

<div class="container mt-5">

        <h2 class="text-center mb-4">Welcome, {{ faculty.username }}</h2>

        <div class="table-container">

            <h4>Your Courses</h4>

            <table class="table table-striped">

                <thead class="table-primary">

                    <tr>

                        <th>Course ID</th>

                        <th>Course Title</th>

                        <th>Actions</th>

                    </tr>

                </thead>

                <tbody>

                    {% for course in courses %}

                    <tr>

                        <td>{{ course.id }}</td>

                        <td>{{ course.title }}</td>

                        <td>

                            <a href="/faculty/manage-course-materials/{{ course.id }}" class="btn btn-info btn-sm">Manage Materials</a>

                            <a href="/faculty/manage-assignments/{{course.id }}" class="btn btn-success btn-sm">Manage Assignments</a>

                            <a href="/faculty/manage-grades/{{ course.id }}" class="btn btn-warning btn-sm">Manage Grades</a>

                            <a href="/faculty/manage-schedule/{{ course.id }}" class="btn btn-primary btn-sm">Manage Schedule</a>

                        </td>

                    </tr>

                    {% else %}

                    <tr>

                        <td colspan="3" class="text-center">No courses allocated to you yet.</td>

                    </tr>

                    {% endfor %}

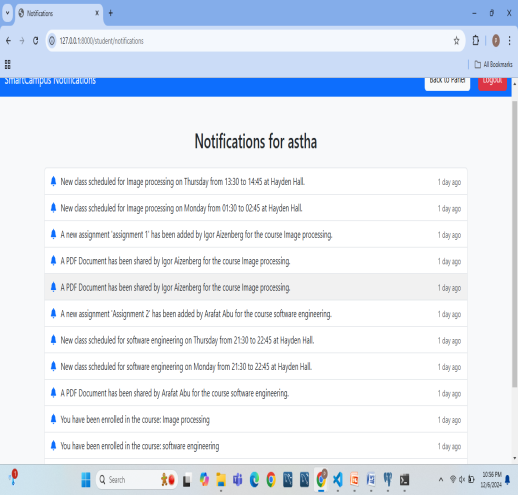
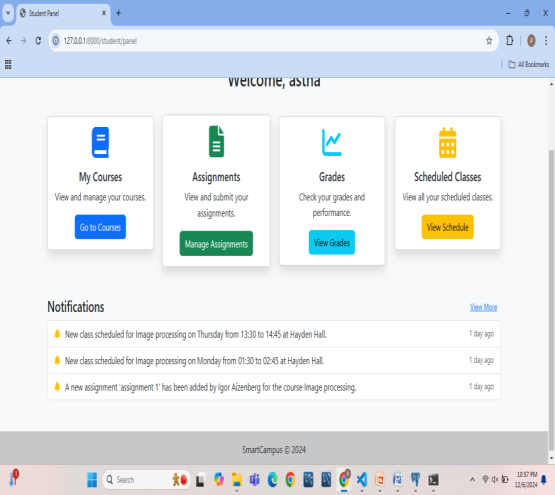
                </tbody>

            </table>

        </div>

    </div>

1. **User Functional Requirement 3:** The system shall provide real-time notifications to students regarding class schedules, assignment deadlines, and exam results.
   * Following is the notification page where students can see real time notifications about class schedules, assignment deadlines and exam result.



* + This is the notification page code where

<nav class="navbar navbar-expand-lg navbar-dark bg-primary">

        <div class="container-fluid">

            <a class="navbar-brand" href="#">SmartCampus Notifications</a>

            <div class="ms-auto">

                <a href="/student/panel" class="btn btn-light me-3">Back to Panel</a>

                <a href="/auth/logout" class="btn btn-danger">Logout</a>

            </div>

        </div>

    </nav>

    <div class="container mt-5">

        <h2 class="text-center mb-4">Notifications for {{ student.username }}</h2>

        <ul class="list-group">

            {% for notification in notifications %}

            <li class="list-group-item">

                <div>

                    <i class="fas fa-bell text-primary me-2"></i>

                    {{ notification.message }}

                </div>

                <small class="text-muted" data-timestamp="{{ notification.created\_at }}"></small>

            </li>

            {% else %}

            <li class="list-group-item text-center">

                <i class="fas fa-info-circle text-secondary me-2"></i>

                No notifications available.

            </li>

            {% endfor %}

        </ul>

    </div>

// Update notification timestamps dynamically

        document.addEventListener("DOMContentLoaded", () => {

            const timestamps = document.querySelectorAll('[data-timestamp]');

            timestamps.forEach(element => {

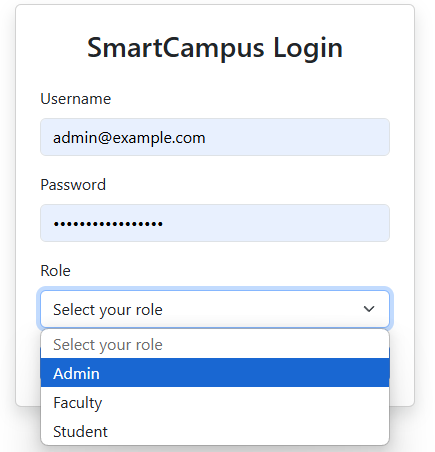
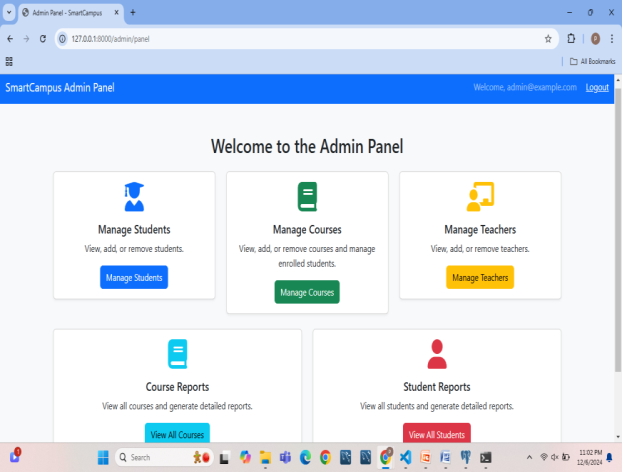
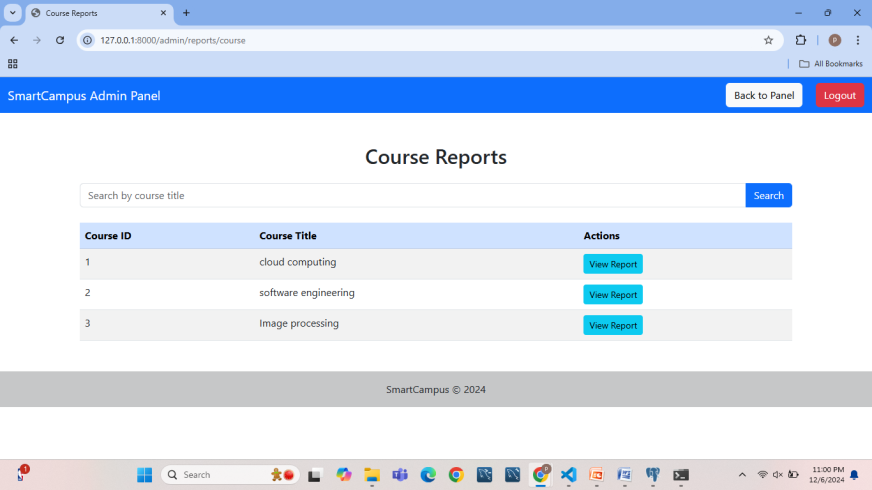
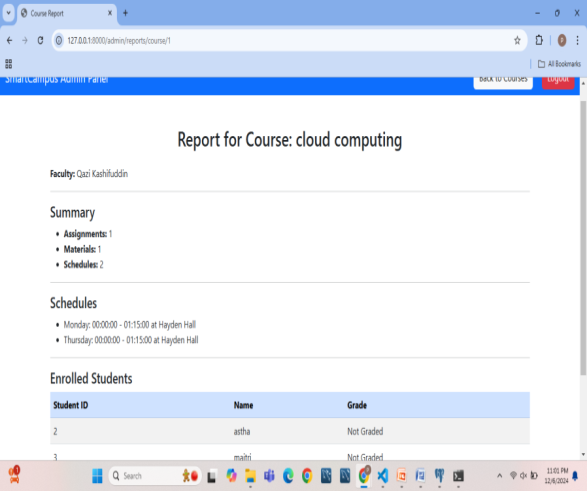
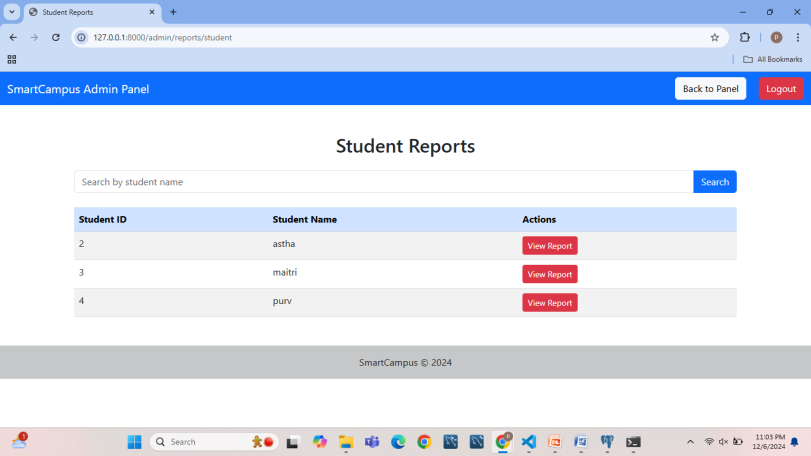
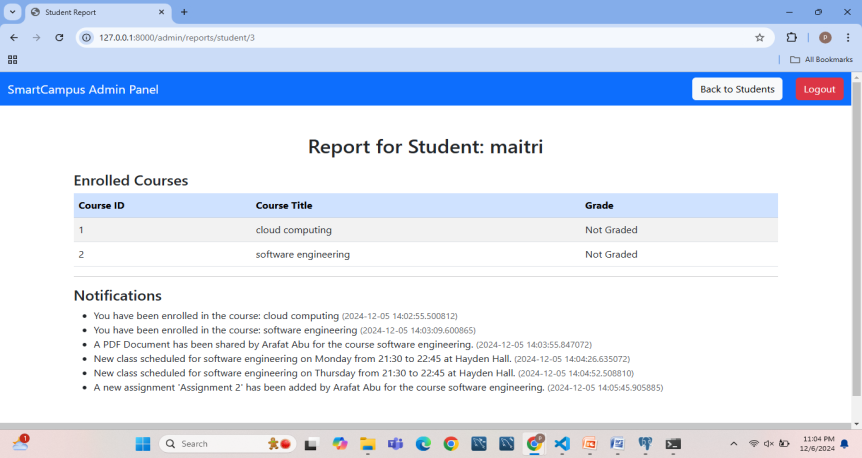
                const timestamp = element.getAttribute("data-timestamp");

                element.textContent = timeAgo(timestamp);

            });

        });

1. **User Functional Requirement 4:** Administrators shall be able to generate reports on student performance and campus activities.
   * Following is the administration page login page and its main page content(Admin has all authority to add new students, professor, assign subject to professor and students for that subject and also can generate student report and course report):

* + Code for admin panel is:

<nav class="navbar navbar-expand-lg navbar-dark bg-primary">

        <div class="container-fluid">

            <a class="navbar-brand" href="#">SmartCampus Admin Panel</a>

            <div class="ms-auto">

                <span class="navbar-text me-3">Welcome, {{ username }}</span>

                <a href="/auth/logout" class="logout-btn">Logout</a>

            </div>

        </div>

    </nav>

    <div class="container mt-5">

        <h2 class="text-center mb-4">Welcome to the Admin Panel</h2>

        <!-- Management Sections -->

        <div class="row">

            <div class="col-md-4">

                <div class="card shadow-sm">

                    <div class="card-body text-center">

                        <i class="fas fa-user-graduate fa-3x text-primary mb-3"></i>

                        <h5 class="card-title">Manage Students</h5>

                        <p class="card-text">View, add, or remove students.</p>

                        <a href="/admin/manage-students" class="btn btn-primary">Manage Students</a>

                    </div>

                </div>

            </div>

            <div class="col-md-4">

                <div class="card shadow-sm">

                    <div class="card-body text-center">

                        <i class="fas fa-book fa-3x text-success mb-3"></i>

                        <h5 class="card-title">Manage Courses</h5>

                        <p class="card-text">View, add, or remove courses and manage enrolled students.</p>

                        <a href="/admin/manage-courses" class="btn btn-success">Manage Courses</a>

                    </div>

                </div>

            </div>

            <div class="col-md-4">

                <div class="card shadow-sm">

                    <div class="card-body text-center">

                        <i class="fas fa-chalkboard-teacher fa-3x text-warning mb-3"></i>

                        <h5 class="card-title">Manage Teachers</h5>

                        <p class="card-text">View, add, or remove teachers.</p>

                        <a href="/admin/manage-teachers" class="btn btn-warning">Manage Teachers</a>

                    </div>

                </div>

            </div>

        </div>

        <!-- Report Section -->

        <!-- Report Section -->

        <div class="row mt-4">

            <div class="col-md-6">

                <div class="card shadow-sm">

                    <div class="card-body text-center">

                        <i class="fas fa-book fa-3x text-info mb-3"></i>

                        <h5 class="card-title">Course Reports</h5>

                        <p class="card-text">View all courses and generate detailed reports.</p>

                        <a href="/admin/reports/course" class="btn btn-info">View All Courses</a>

                    </div>

                </div>

            </div>

            <div class="col-md-6">

                <div class="card shadow-sm">

                    <div class="card-body text-center">

                        <i class="fas fa-user fa-3x text-danger mb-3"></i>

                        <h5 class="card-title">Student Reports</h5>

                        <p class="card-text">View all students and generate detailed reports.</p>

                        <a href="/admin/reports/student" class="btn btn-danger">View All Students</a>

                    </div>

                </div>

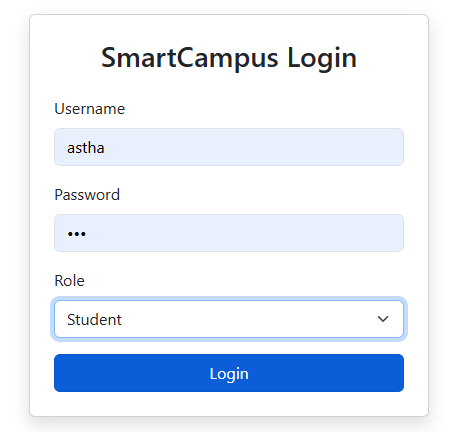
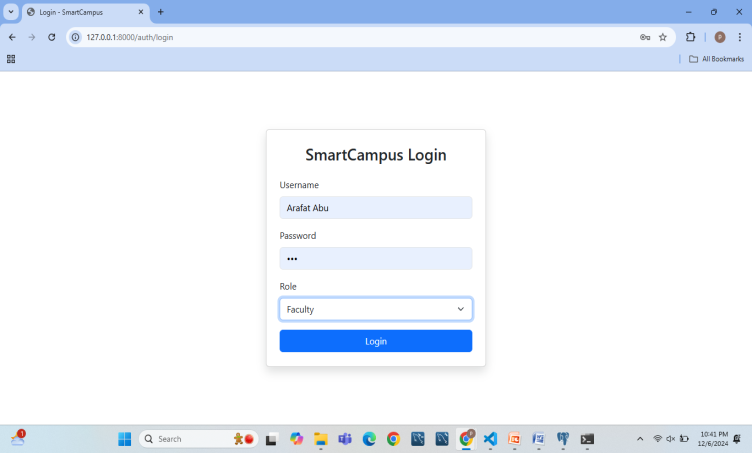
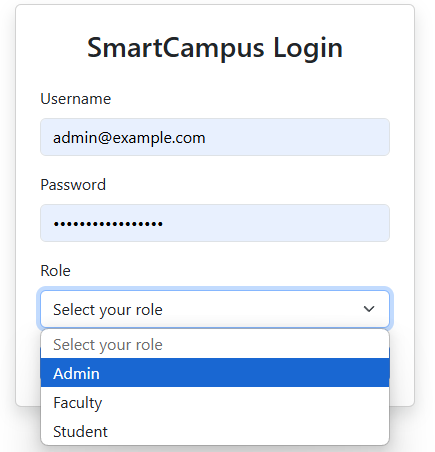
            </div>

        </div>

    </div>

**Non-functional requirements:**

1. **User Non-Functional Requirement 1:** The system interface shall be user-friendly and easy to navigate for all users (students, faculty, administrators).
   * Our system is easy to navigate for all users(students, faculty and admin panel) which is shown by following screenshots:

* + Based on role how to navigate to different profiles:

# Redirect to appropriate panel based on role

    if role == "admin":

        response = RedirectResponse(url="/admin/panel", status\_code=302)

    elif role == "faculty":

        response = RedirectResponse(url="/faculty/panel", status\_code=302)

    elif role == "student":

        response = RedirectResponse(url="/student/panel", status\_code=302)

    else:

        raise HTTPException(status\_code=403, detail="Unauthorized role")

* + How to redirect to the admin panel:

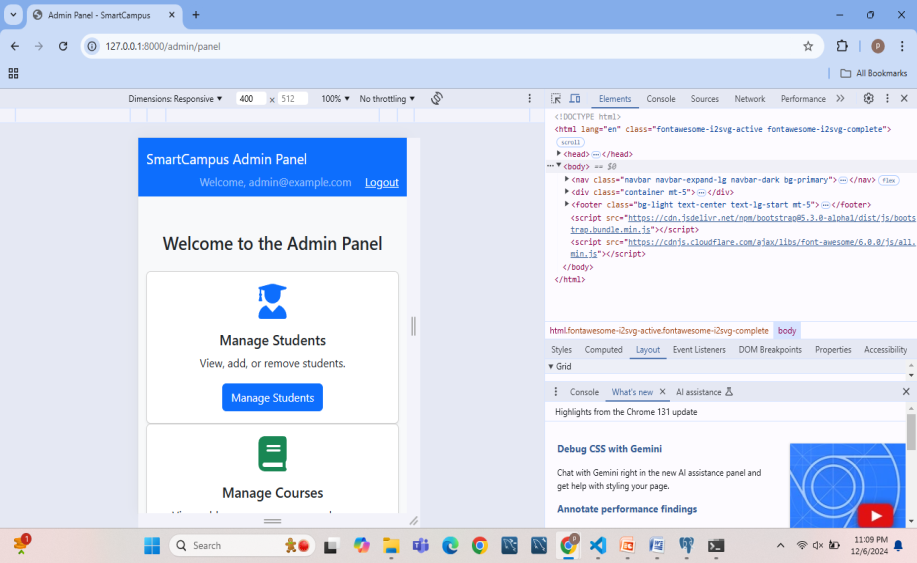
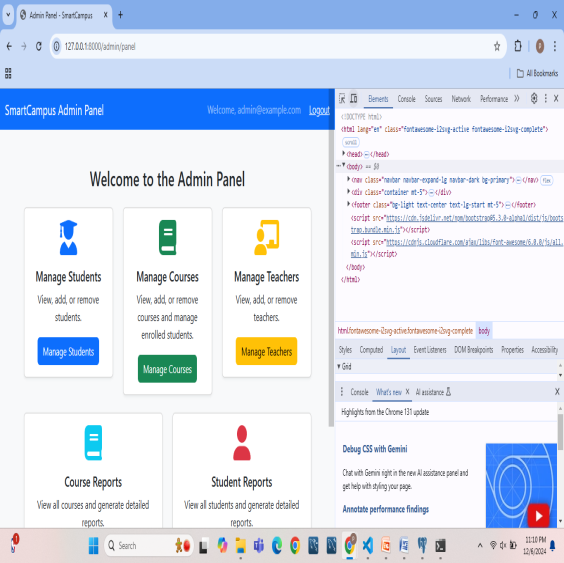
@auth\_router.get("/admin/panel", response\_class=HTMLResponse)

async def admin\_panel(request: Request, username: str = Depends(check\_session)):

    """Admin Panel: Protected route for admin users."""

    return templates.TemplateResponse("admin\_panel.html", {"request": request, "username": username})

1. **User Non-Functional Requirement 2**: The system shall be accessible on multiple devices, including desktops, tablets, and smartphones.

1. **User Non-Functional Requirement 3:** The system shall ensure a response time of fewer than 3 seconds for each user request.
2. **User Non-Functional Requirement 4:** The system shall maintain high security standards by encrypting all sensitive user data.

**System-functional requirements:**

1. **System Functional Requirement 1:** The system shall authenticate user credentials against a secure database using encryption protocols.
2. **System Functional Requirement 2:** The system shall store and manage course-related data, including assignments, grades, and feedback, in a structured database.
3. **System Functional Requirement 3**: The system shall generate real-time alerts for upcoming deadlines, class schedules, and campus events.
4. **System Functional Requirement 4:** The system shall allow for data exports, enabling administrators to generate performance reports for analysis.
5. **System Functional Requirement 5:** The system shall maintain logs of all user activities for auditing purposes

**System non-functional requirements:**

1. **System Non-Functional Requirement 1:** The system shall support up to 10,000 simultaneous users without any noticeable performance degradation.
2. **System Non-Functional Requirement 2:** The system shall be scalable, allowing for future expansion as the campus grows.
3. **System Non-Functional Requirement 3:** The system shall have an uptime of 99.9%, ensuring continuous availability to users.
4. **System Non-Functional Requirement 4:** The system shall adhere to industry-standard security protocols, including AES-256 encryption for all stored data.
5. **System Non-Functional Requirement 5:** The system shall comply with data privacy laws and regulations, such as GDPR.

**Domain Requirements:**

The Smart Campus Management System must adhere to educational regulations, including accreditation standards, and data privacy requirements, such as the Family Educational Rights and Privacy Act (FERPA). The system must also comply with local, regional, and international data protection laws, ensuring that sensitive information like student grades, personal information, and administrative data is protected.

**Application of IEEE/ACM code of Ethics:**

The following principles from the IEEE/ACM Code of Ethics are applied during the specification phase of the project:

**Principle 1: Public**

the system is developed with the public good in mind, ensuring the privacy, security, and reliability of user data. By implementing secure login protocols and encrypted data storage, the system protects user information and ensures public trust.

**Principle 2: Client and Employer**

the project ensures that the requirements of the stakeholders—students, faculty, and administrators—are fully met. Ethical communication is maintained with the client, providing transparency about project progress and ensuring their needs are addressed.

**Principle 3: Product**

The Smart Campus Management System is designed with high standards of quality, ensuring it is reliable, secure, and user-friendly. The system meets the specified functional and non-functional requirements to deliver a robust product.

**Principle 4: Judgment**

The specification process includes evaluating technical constraints and potential risks. Ethical decisions are made throughout the project to prioritize user safety, privacy, and data security.

**Principle 5: Management**

Ethical project management ensures that the team adheres to timelines and delivers a product that aligns with the client’s objectives. Regular reviews and assessments are conducted to ensure ethical compliance in design and development.

**Principle 6: Profession**

The development team adheres to professional standards, ensuring compliance with educational regulations and data privacy laws. The team remains committed to continuous professional development and adherence to ethical standards.

**Principle 7: Colleagues**

Collaborative efforts among team members are fostered, ensuring that ethical concerns raised during development are addressed. Peer reviews and consultations ensure that the best ethical practices are followed.

**Principle 8: Self**

Individual developers commit to ongoing learning and ethical improvement, staying updated with industry best practices and regulatory changes. Each team member is responsible for upholding the ethical integrity of the project.

**Conclusion:**

The Smart Campus Management System (SCMS) effectively enhances campus operations by providing secure, user-friendly, and scalable solutions tailored to the needs of students, faculty, and administrators. Key advantages include streamlined workflows, real-time notifications, and robust performance under current user limits. However, challenges such as potential scalability issues for larger user bases and the need for further infrastructure improvements highlight areas for growth.

Future enhancements, including AI-driven personalization, IoT-based campus monitoring, and advanced reporting tools, could further elevate system capabilities. By addressing these opportunities, SCMS can remain a cutting-edge solution for modern campus management.

**References:**

1. **Real-Time Notification Systems**  
   [Gama, J., Zaki, M. J., & Spiliopoulou, M. (2014). Data Stream Mining: Real-Time Decision Making. Springer.](Gama,%20J.,%20Zaki,%20M.%20J.,%20&%20Spiliopoulou,%20M.%20(2014).%20Data%20Stream%20Mining:%20Real-Time%20Decision%20Making.%20Springer.)
2. **Scalability and Performance Challenges**  
   [Armstrong, T. G., Ponnekanti, V., Borthakur, D., & Callaghan, M. (2013). "LinkBench: A database benchmark for the social graph." Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data.](Armstrong,%20T.%20G.,%20Ponnekanti,%20V.,%20Borthakur,%20D.,%20&%20Callaghan,%20M.%20(2013).%20%22LinkBench:%20A%20database%20benchmark%20for%20the%20social%20graph.%22%20Proceedings%20of%20the%202013%20ACM%20SIGMOD%20International%20Conference%20on%20Management%20of%20Data.)
3. **Security and Data Privacy**  
   [Stallings, W., & Brown, L. (2018). *Computer Security: Principles and Practice* (4th ed.). Pearson.](Stallings,%20W.,%20&%20Brown,%20L.%20(2018).%20Computer%20Security:%20Principles%20and%20Practice%20(4th%20ed.).%20Pearson.)
4. **Educational Management Systems**  
   [Al-Busaidi, K. A., & Al-Shihi, H. (2012). "Key factors to instructors’ satisfaction of learning management systems in blended learning." *Computers in Human Behavior*, 28(2), 225–234.](Al-Busaidi,%20K.%20A.,%20&%20Al-Shihi,%20H.%20(2012).%20%22Key%20factors%20to%20instructors’%20satisfaction%20of%20learning%20management%20systems%20in%20blended%20learning.%22%20Computers%20in%20Human%20Behavior,%2028(2),%20225–234.)
5. **AI and Machine Learning Integration**  
   [Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.](Goodfellow,%20I.,%20Bengio,%20Y.,%20&%20Courville,%20A.%20(2016).%20Deep%20Learning.%20MIT%20Press.)