Al Alamein International University

Faculty of Computer

Science & Engineering

Software Engineering

Course Code: CSE251

University Management System

Report 2

Project Manager:

Adham Ayman Farouk Ibrahim – 21100782

Team Members:

Mazen Mohamed elsayed – 21100912

Abdelrahman Mohamed Abdelaziz – 21100799

Ahmed Mokhtar Mokhtar - 21100781

Ahmed Mohamed Mahdi – 21100822

DR. Islam Elgedawy

Functional User & System Requirements:

1. Student Management System:
    1.1. Ability to view and manage student records.
    1.2. Ability to generate reports on attendance, grades, and academic progress.
    1.3. Ability to register for classes and pay fees.
2. Faculty Management System:
    2.1. Ability to view and manage faculty records.
    2.2. Ability to generate reports on faculty workload, research output, and teaching evaluations.
    2.3. Ability to submit grades and request time off.
3. Course Management System:
    3.1. Ability to create and manage course schedules.
    3.2. Ability to manage course materials.
    3.3. Ability to generate reports on course enrollment, attendance, and performance.
4. Exam Management System:
    4.1. Ability to create and manage exam schedules.
    4.2. Ability to manage exam materials.
    4.3. Ability to generate reports on exam performance.
5. Fee Management System:
    5.1. Ability to manage fee collection and payment processing.
    5.2. Ability to generate reports on fee collection and payment history.
6. Library Management System:
    6.1. Ability to manage library materials and inventory.
    6.2. Ability to generate reports on library usage and loan history.
    6.3. Easy communication with library users.
7. Housing Management System:
    7.1. Ability to manage housing applications and assignments.
    7.2. Ability to generate reports on housing occupancy and availability.
8. Transportation Management System:
    8.1. Ability to manage transportation services and scheduling.
    8.2. Ability to generate reports on transportation usage and maintenance history.
9. Security Management System:
    9.1. Ability to manage access control and security measures on campus.
    9.2. Ability to generate reports on security incidents and response times.
10. Career Services Management System:
    10.1. Ability to manage job postings and career counseling services for students.
    10.2. Ability to generate reports on job placement rates and employer feedback.
11. Alumni Management System:
    11.1. Ability to manage alumni records and communication.
    11.2. Ability to generate reports on alumni engagement and giving.

12. Health Services Management System:
    12.1. Ability to manage health services and medical records for students.
    12.2. Ability to generate reports on health outcomes and usage.
13. Research Management System:
    13.1. Ability to manage research projects and grants.
    13.2. Ability to generate reports on research output and funding.
14. Facilities Management System:
    14.1. Ability to manage maintenance and repairs for campus facilities.
    14.2. Ability to generate reports on facility usage and maintenance costs.
15. Athletics Management System:
    15.1. Ability to manage athletic programs and schedules.
    15.2. Ability to generate reports on athletic performance and attendance.
    15.3. Diversity and Inclusion Management System:
    15.4. Ability to manage diversity and inclusion initiatives on campus.
    15.5. Ability to generate reports on diversity metrics and outcomes.
16. IT Management System:
    16.1. Ability to manage campus IT infrastructure and support services.
    16.2. Ability to generate reports on IT usage and service requests.
17. Accreditation Management System:
    17.1. Ability to manage accreditation processes and documentation.
    17.2. Ability to generate reports on accreditation status and progress.
18. Compliance Management System:
    18.1. Ability to manage compliance with laws and regulations.
    18.2. Ability to generate reports on compliance metrics and activities.
19. Event Management System:
    19.1. Ability to manage campus events and scheduling.
    19.2. Ability to generate reports on event attendance and feedback.
20. Student Organizations Management System:
    20.1. Ability to manage student organizations and events.
    20.2. Ability to generate reports on student organization activities and engagement.
21. International Student Management System:
    21.1. Ability to manage international student applications and enrollment.
    21.2. Ability to generate reports on international student demographics and outcomes.
22. Campus Store Management System:
    22.1. Ability to manage campus store inventory and sales.
    22.2. Ability to generate reports on store revenue and popular items.

23. Food Services Management System:
    23.1. Ability to manage campus dining options and meal plans.
    23.2. Ability to generate reports on food services usage and satisfaction.
24. Student Health Insurance Management System:
    24.1. Ability to manage student health insurance enrollment and claims.
    24.2. Ability to generate reports on insurance usage and costs.
25. Grants and Scholarships Management System:
    25.1. Ability to manage grants and scholarships awarded to students.
    25.2. Ability to generate reports on grant and scholarship activities and outcomes.
26. Environmental Sustainability Management System:
    26.1. Ability to manage campus sustainability initiatives and activities.
    26.2. Ability to generate reports on sustainability metrics and outcomes.
27. Student Counseling and Mental Health Management System:
    27.1. Ability to manage student counseling and mental health services.
    27.2. Ability to generate reports on service usage and outcomes.

Non-Functional User & System Requirements:

1. The system must be available 24/7.
2. The system must be designed to protect the privacy of users.
3. The system must be highly secure.
4. The system should offer all required functions to the user.
5. The system must have a response time of under 10 seconds.
6. The system screen refresh time should be less than 5 seconds.
7. The system must have mechanisms in place to prevent unauthorized access and ensure data integrity.
8. The system should be compatible with a wide range of devices and platforms such as PCs, smartphones, and tablets.
9. The system should be easy to maintain and update, with clear documentation and well-organized code.
10. The system should be optimized to consume minimal computing resources and energy.
11. The probability of events leading to failure shouldn't be higher than 90%.
12. The system should handle errors, provide error messages to users, and log errors.
13. The system must have a backup and recovery plan to ensure that data can be restored in the event of a system failure.
14. The system should be designed to allow for future upgrades.
15. The probability of non-availability shouldn't be higher than 5%.
16. The system should be scalable to handle an increase in user traffic and data volume.
17. The system should have a user-friendly interface that is easy to navigate and understand.
18. The system should be designed with accessibility in mind, including features for users with disabilities.
19. The system should have a comprehensive user manual and online help resources.
20. The system should have a reliable and secure authentication and authorization system.
21. The system should be compliant with relevant data protection and privacy laws and regulations.

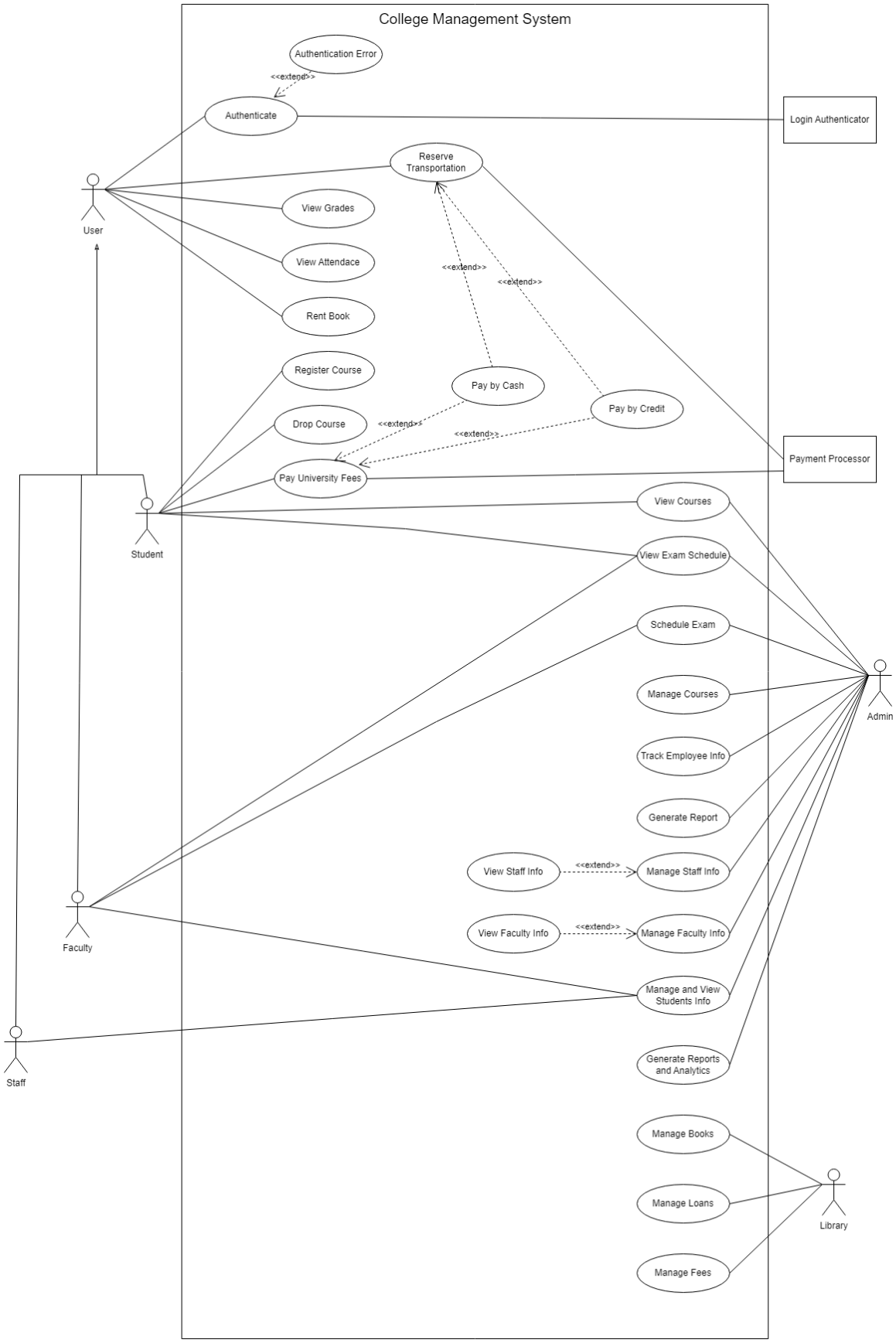
22. The system should have robust data backup and disaster recovery mechanisms in place.
23. The system should have a monitoring and alerting system to detect and respond to system failures or anomalies.
24. The system should provide accurate and timely reporting and analytics to support decision-making.
25. The system should be designed with modularity and flexibility in mind, allowing for easy customization and integration with other systems.
26. The system should have a testing and quality assurance process to ensure that updates and changes do not cause unexpected issues or disruptions.
27. The system should have a mechanism for collecting and incorporating user feedback to continuously improve the system.
28. The system should be designed with sustainability in mind, minimizing its impact on the environment and promoting energy efficiency.
29. The system should have a clear and transparent data retention and deletion policy.
30. The system should have a plan for continuous monitoring and improvement of its security and privacy features.

Test Requirements:

1. Mange Student information:
    1.1. Verify that a new student can be added to the system.
    1.2. Verify that a student can be updated with new information.
    1.3. Verify that a student can be removed from the system.
    1.4. Verify that a specific student can be retrieved from the system by their ID.
    1.5. Verify that a list of all students can be retrieved from the system.
2. Mange Faculty Information:
    2.1. Verify that a new faculty member can be added to the system.
    2.2. Verify that a faculty member can be updated with new information.
    2.3. Verify that a faculty member can be removed from the system.
    2.4. Verify that a specific faculty member can be retrieved from the system by their ID.
    2.5. Verify that a list of all faculty members can be retrieved from the system.
3. Mange courses:
    3.1. Verify that a new admission can be created in the system.
    3.2. Verify that an existing admission can be updated with new information.
    3.3. Verify that a specific admission can be retrieved from the system by its ID.
    3.4. Verify that a list of all admissions can be retrieved from the system.
    3.5. Verify that a list of admissions can be retrieved based on their status (e.g. accepted, rejected, pending).
    3.6. Verify that a list of admissions can be retrieved based on the program they are applying to.
4. Examination Management System:
    4.1. Verify that a new exam can be scheduled in the system.

4.2. Verify that a student can be registered for an exam.
4.3. Verify that an exam can be graded.
4.4. Verify that a list of exam schedules can be retrieved based on the course they are associated with.
4.5. Verify that a list of exam results can be retrieved for a specific student.

5. Mange Books:
   5.1. Verify that a new book can be added to the system.
   5.2. Verify that a book can be updated with new information.
   5.3. Verify that a book can be removed from the system.
   5.4. Verify that a specific book can be retrieved from the system by its ID.
   5.5. Verify that a list of all books can be retrieved from the system.
   5.6. Verify that books can be searched by title, author, and category.
6. Mange room:
   6.1. Verify that a new room can be added to the system.
   6.2. Verify that a room can be updated with new information.
   6.3. Verify that a room can be removed from the system.
   6.4. Verify that a specific room can be retrieved from the system by its ID.
   6.5. Verify that a list of all rooms can be retrieved from the system.
   6.6. Verify that rooms can be searched by type.
   6.7. Verify that a student can be allocated to a room.
   6.8. Verify that hostel fee can be collected for a specific student.
   6.9. Verify that a hostel receipt can be generated for a specific student.
   6.10. Verify that a list of hostel fee payments can be retrieved for a specific student.


College Management System
Authentication Error
<<extend>>
Authenticate
Login Authenticator
Reserve Transportation
User
View Grades
View Attendace
Rent Book
<<extend>>
<<extend>>
Register Course
Pay by Cash
Pay by Credit
Drop Course
<<extend>>
<<extend>>
Pay University Fees
Payment Processor
Student
View Courses
View Exam Schedule
Schedule Exam
Admin
Manage Courses
Track Employee Info
Generate Report
View Staff Info
<<extend>>
Manage Staff Info
Faculty
View Faculty Info
<<extend>>
Manage Faculty Info
Manage and View Students Info
Staff
Generate Reports and Analytics
Manage Books
Manage Loans
Library
Manage Fees

Use Case:	Generate Reports
Goal:	Generate Students' and User's reports, including but not limited to; Attendance, GPA, CGPA, Fees Report, Employee report etc..
Preconditions:	User must be authenticated on the System.
Success End Condition:	User generates the needed report.
Primary Actor:	User.
Secondary Actor:	Bank.
Trigger:	User is generating a report.

Use Case:	Manage Courses
Goal:	User and Staff are able to Add, Edit and Delete courses.
Preconditions:	User must be authenticated on the System.
Success End Condition:	Course(s) are managed.
Primary Actor:	User.
Secondary Actor:	
Trigger:	User is trying to manage a course or their courses.

Use Case:	Track Employee.
Goal:	Track Employee's Information.
Preconditions:	Staff member must be Authenticated on the System.
Success End Condition:	Employee's need is displayed.
Primary Actor:	Staff.
Secondary Actor:	
Trigger:	Staff Members wants to track their info.

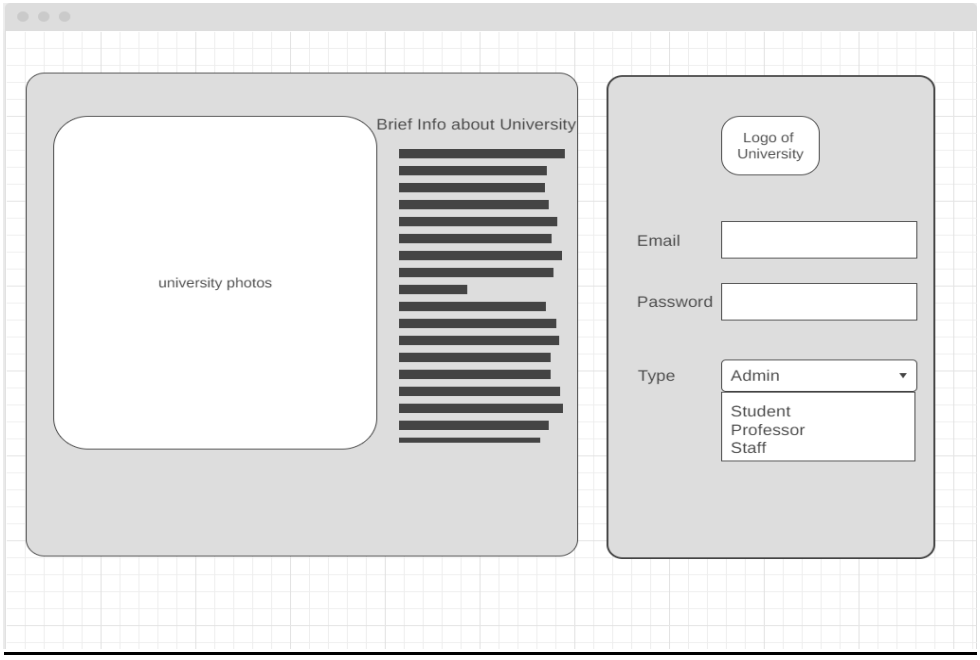
Use Case:	Register Course
Goal:	Course is registered.
Preconditions:	User must be authenticated on the system, and not pass his credit hour requirements.
Success End Condition:	Course is registered Successfully.
Primary Actor:	User.
Secondary Actor:	Student.
Trigger:	User is trying to register a course.

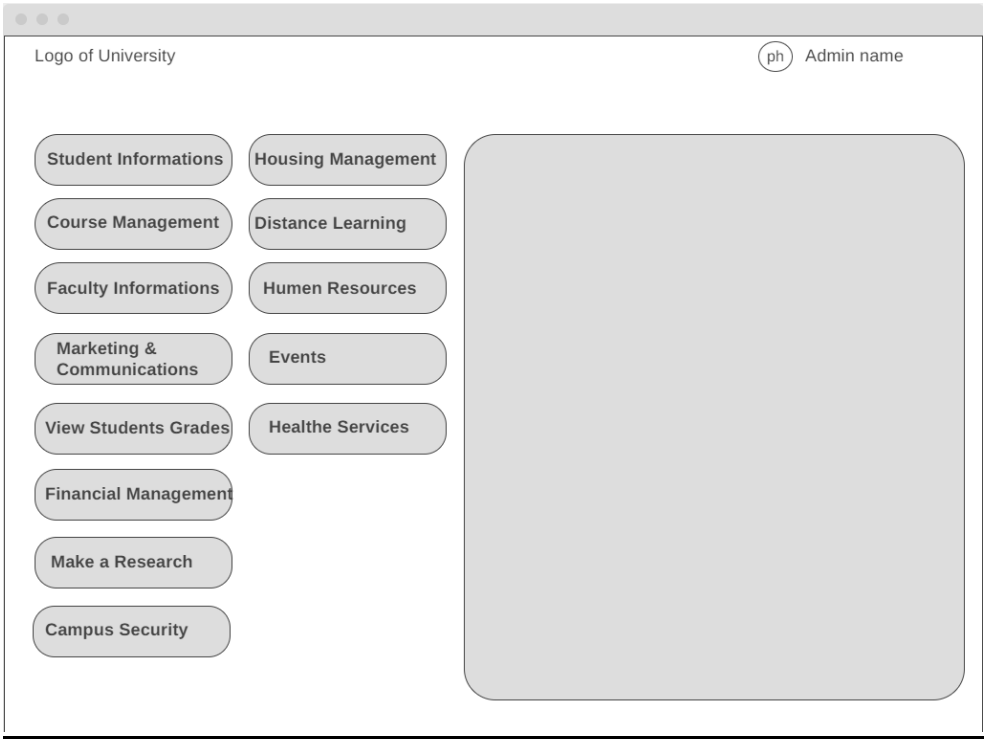
Use Case:	Pay University Fees
Goal:	Pay the University Fees.
Preconditions:	User must be authenticated on the system and must have enough money in payment method.
Success End Condition:	Fees are paid.
Primary Actor:	Student.
Secondary Actor:	Payment Processor.
Trigger:	User is trying to pay the University Fees.

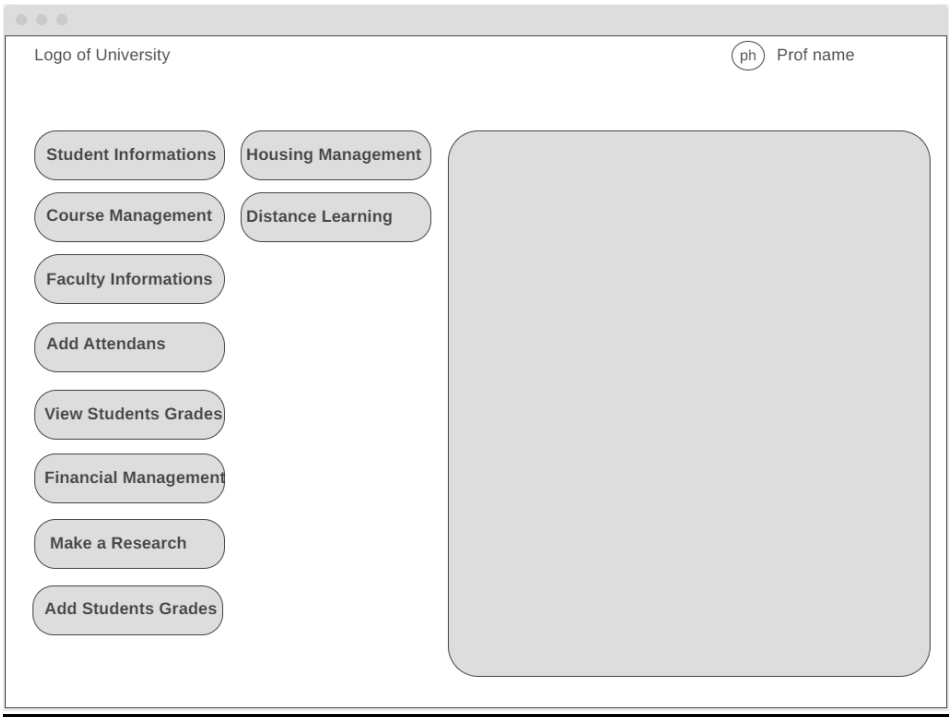
A	Student Management System:	• Student profile management • Student attendance tracking • Student grade management • Course registration and scheduling • Fee payment processing • Communication with faculty and advisors	• Student getStudentProfile(int studentId); • void updateStudentProfile(int studentId, Student student); • void trackAttendance(int studentId, Attendance attendance); • void submitGrades(int facultyId, int courseId, List<StudentGrade> grades); • void registerForCourse(int studentId, int courseId); • List<Course> getCourseSchedule(int studentId); • void processFeePayment(int studentId, FeePayment feePayment); • void communicateWithFaculty(int studentId, int facultyId, String message);
B	Faculty Management System:	• Faculty profile management • Course management and scheduling • Grade submission • Research output tracking • Time off requests • Communication with administrators and students	• void updateFacultyProfile(int facultyId, Faculty faculty); • void manageCourse(int facultyId, Course course); • void submitGrades(int facultyId, int courseId, List<StudentGrade> grades); • void trackResearchOutput(int facultyId, ResearchOutput researchOutput); • void requestTimeOff(int facultyId, TimeOffRequest timeOffRequest); • void communicateWithAdmin(int facultyId, int adminId, String message); • void communicateWithStudents(int facultyId, List<Integer> studentIds, String message);
C	Course Management System:	• Course scheduling and management • Course material management • Enrollment tracking and management • Attendance tracking and reporting • Grade tracking and reporting • Communication with faculty and students	• void manageCourse(Course course); • void scheduleCourse(int courseId, List<Session> sessions); • void manageCourseMaterials(int courseId, List<CourseMaterial> materials); • void enrollStudent(int studentId, int courseId); • void trackAttendance(int courseId, Attendance attendance); • void submitGrades(int facultyId, int courseId, List<StudentGrade> grades); • List<Student> getEnrolledStudents(int courseId);
D	Exam Management System:	• Exam scheduling and management • Exam material management • Exam grading and reporting • Communication with faculty and students	• void scheduleExam(int courseId, Date examDate); • void manageExamMaterials(int courseId, List<ExamMaterial> materials); • void gradeExam(int facultyId, int courseId, Exam exam); • List<Exam> getExamSchedule(int courseId);
E	Fee Management System:	• Fee collection and payment processing • Fee tracking and reporting • Communication with students regarding fees and payments	• void collectFee(int studentId, Fee fee); • void trackFee(int studentId); • void communicateWithStudents(int adminId, List<Integer> studentIds, String message);

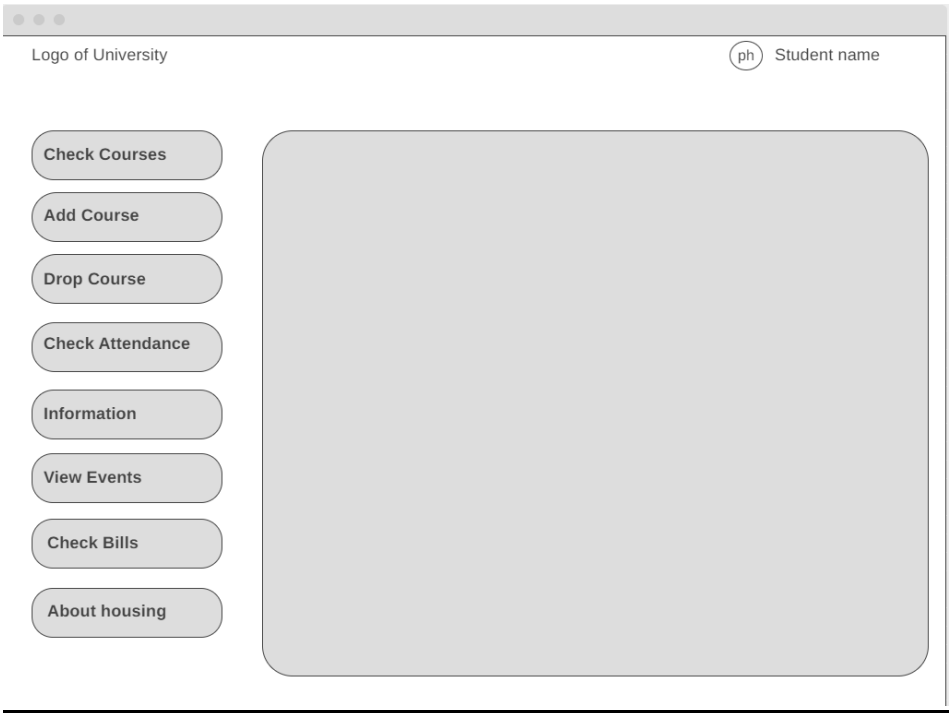
F	Library Management System:	• Library material management and inventory tracking • Loan management and tracking • Patron management and communication • Analytics and reporting on library usage and materials	• void addMaterial(Material material); • void removeMaterial(int materialId); • void checkoutMaterial(int patronId, int materialId); • void checkinMaterial(int patronId, int materialId); • List<Material> searchMaterials(String query); • List<Patron> getPatrons(int materialId);
G	Housing Management System:	• Housing application and assignment management • Occupancy tracking and management • Housing fee collection and reporting • Communication with students regarding housing options	• void applyForHousing(int studentId, HousingApplication application); • void assignHousing(int adminId, int studentId, Housing housing); • void trackOccupancy(int housingId); • void collectHousingFee(int studentId, HousingFee fee);
H	Transportation Management System:	• Transportation scheduling and management • Transportation vehicle maintenance tracking • Transportation usage tracking and reporting • Communication with transportation users	• void scheduleTransportation(int courseId, List<Session> sessions); • void manageTransportationVehicles(List<Vehicle> vehicles); • void trackTransportationUsage(int courseId); • void communicateWithTransportationUsers(int adminId, List<Integer> userIDs, String message);

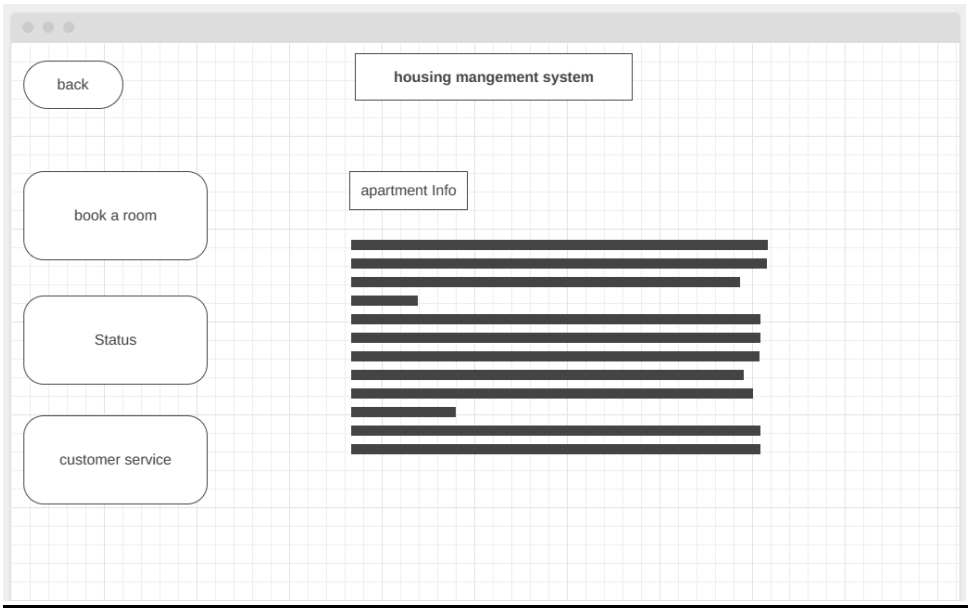
GUI

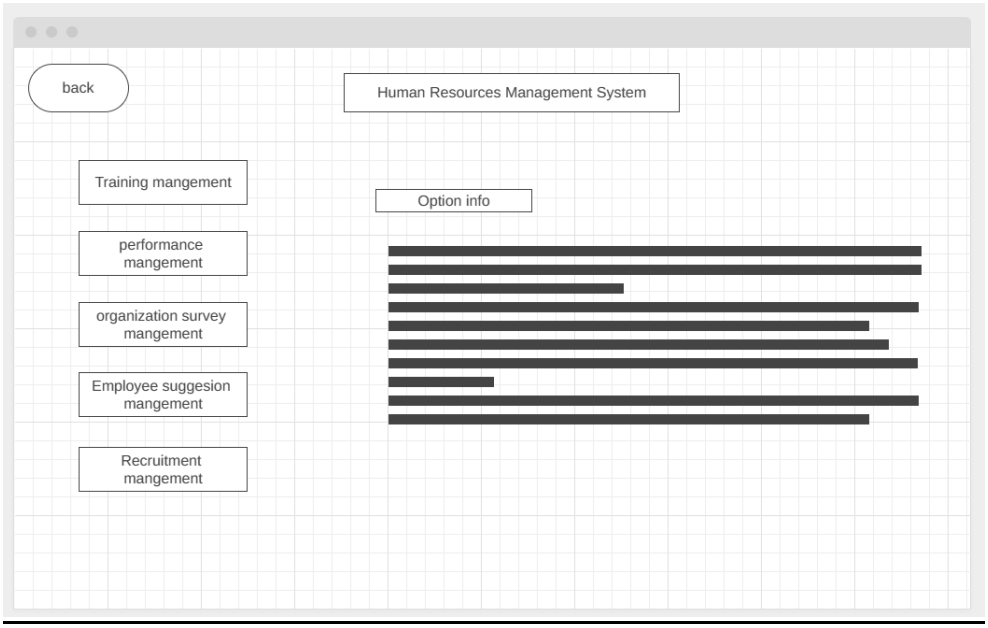

Brief Info about University
university photos
Logo of University
Email
Password
Type
Admin
Student
Professor
Staff

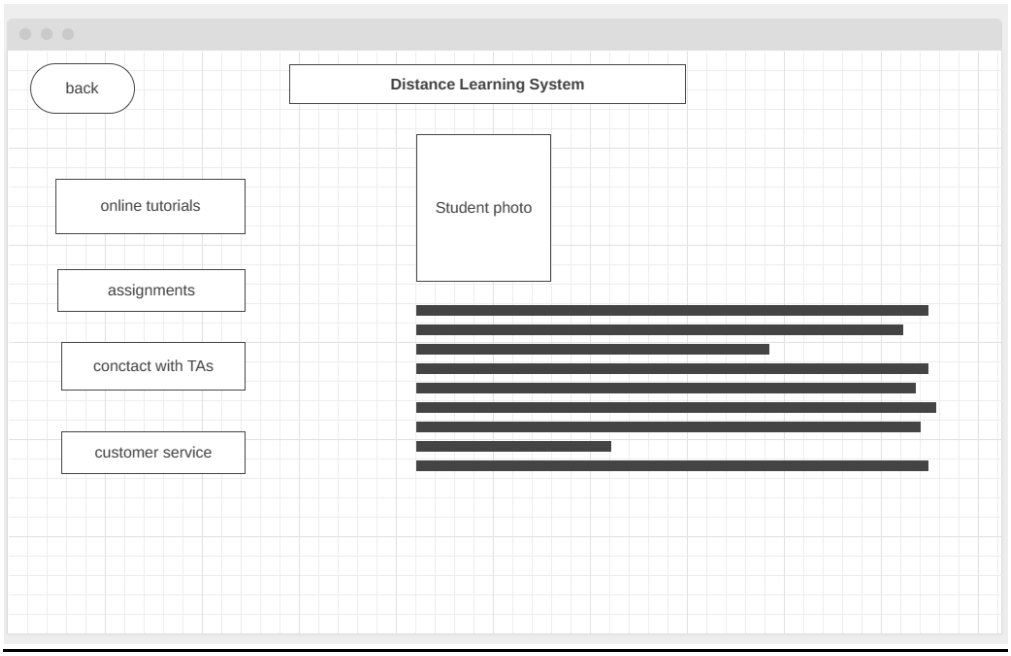

Logo of University
ph
Admin name
Student Informations
Housing Management
Course Management
Distance Learning
Faculty Informations
Humen Resources
Marketing & Communications
Events
View Students Grades
Healthe Services
Financial Management
Make a Research
Campus Security


Logo of University
ph
Prof name
Student Informations
Housing Management
Course Management
Distance Learning
Faculty Informations
Add Attendans
View Students Grades
Financial Management
Make a Research
Add Students Grades

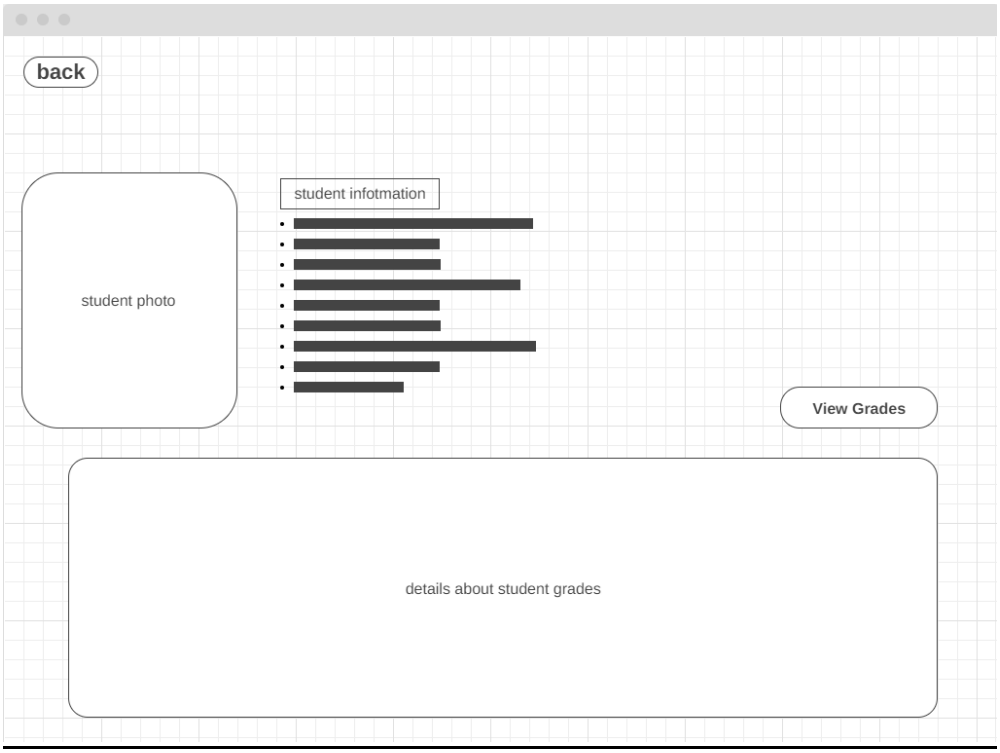

Logo of University
ph
Student name
Check Courses
Add Course
Drop Course
Check Attendance
Information
View Events
Check Bills
About housing

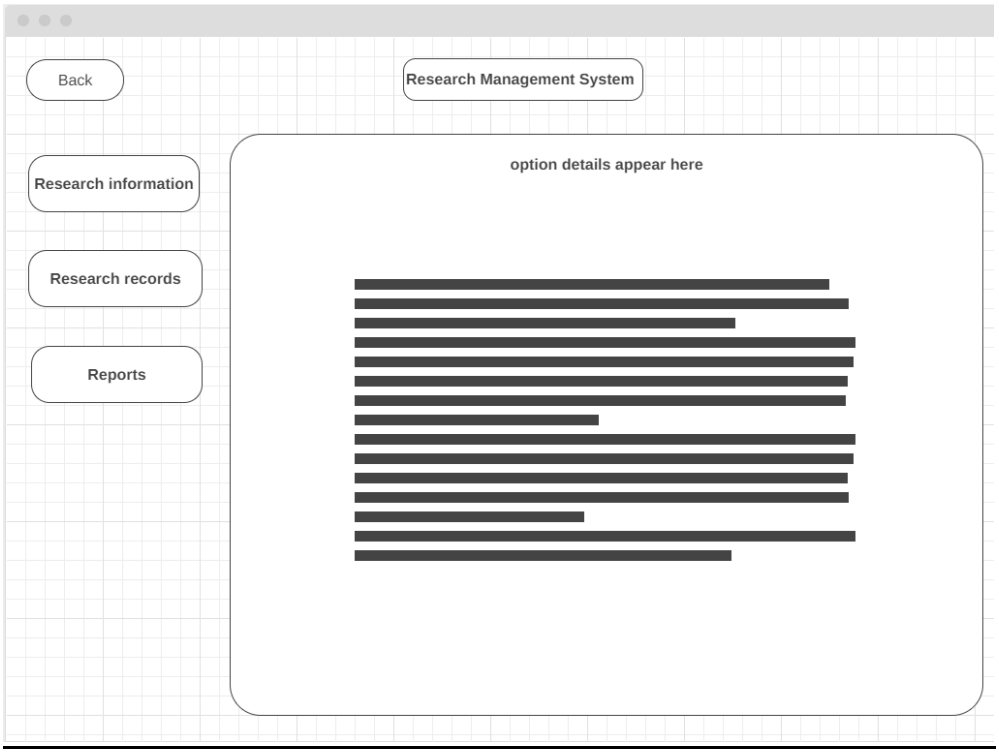

back
housing mangement system
book a room
apartment Info
Status
customer service

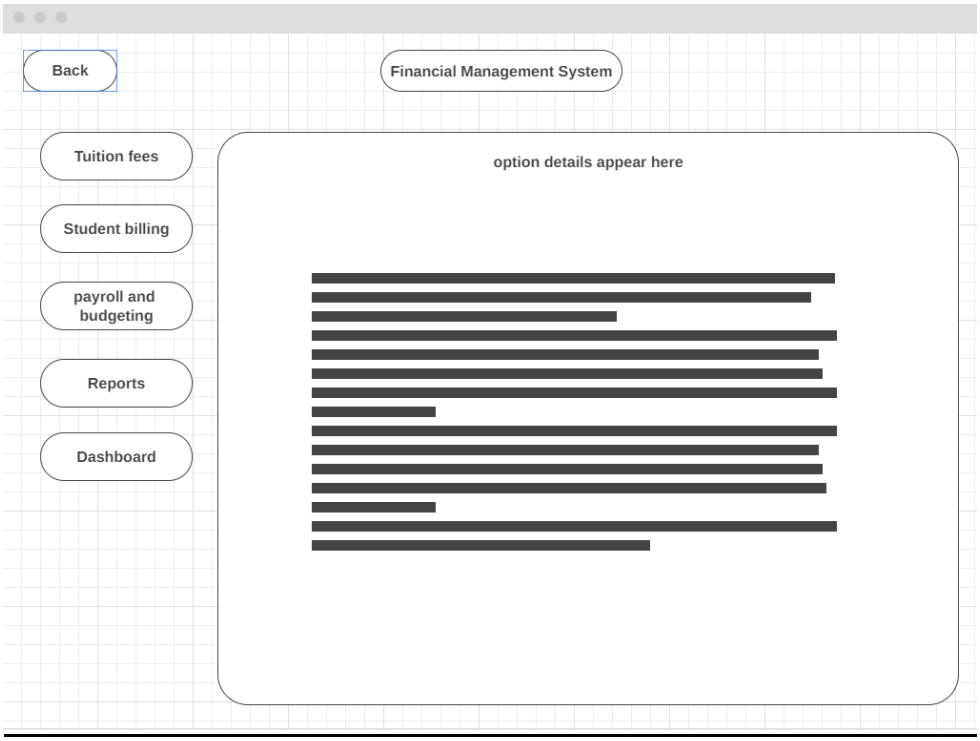

back
Human Resources Management System
Training mangement
Option info
performance mangement
organization survey mangement
Employee suggesion mangement
Recruitment mangement

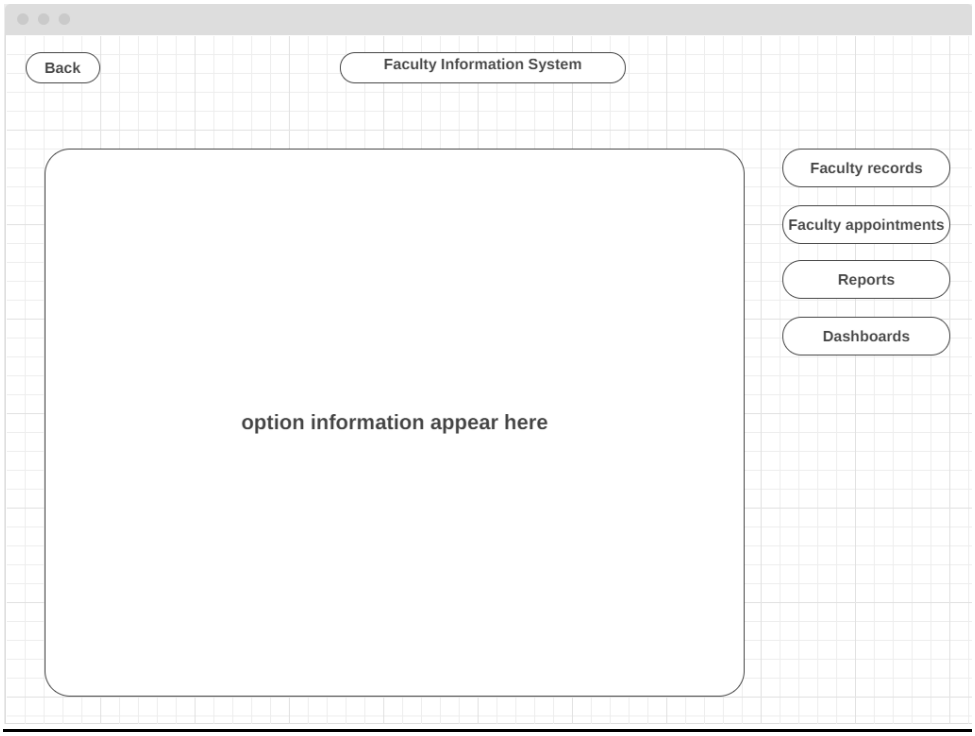

back
Distance Learning System
online tutorials
assignments
conctact with TAs
customer service
Student photo

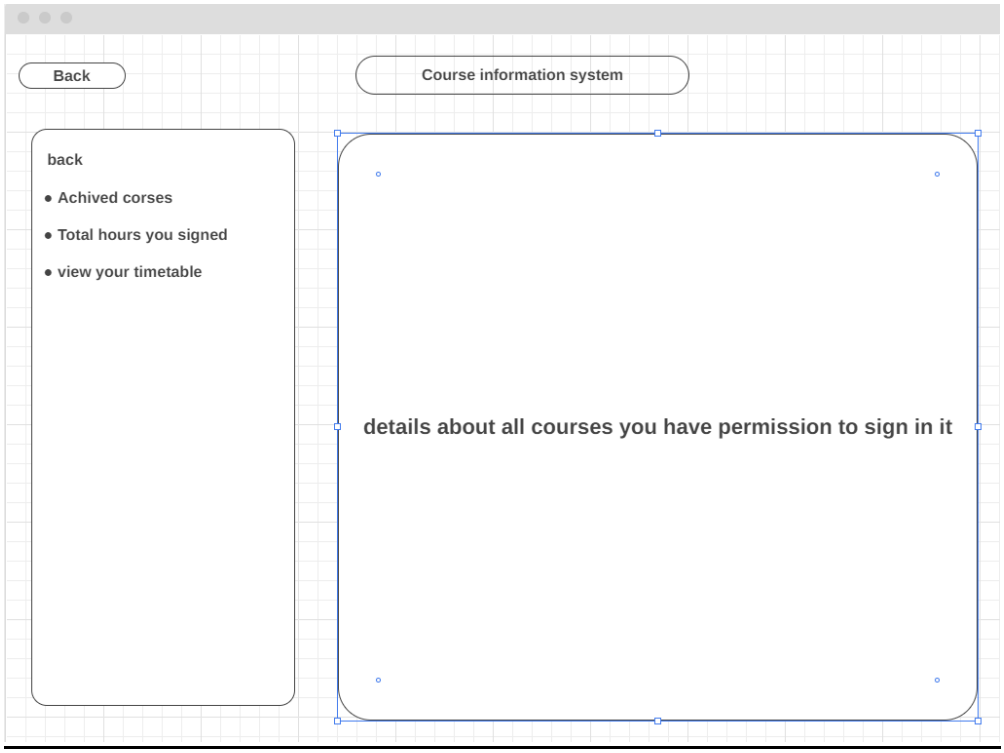

back
campus security system
lost belongings
campus map
criminal records
generate reports
threats
Option info

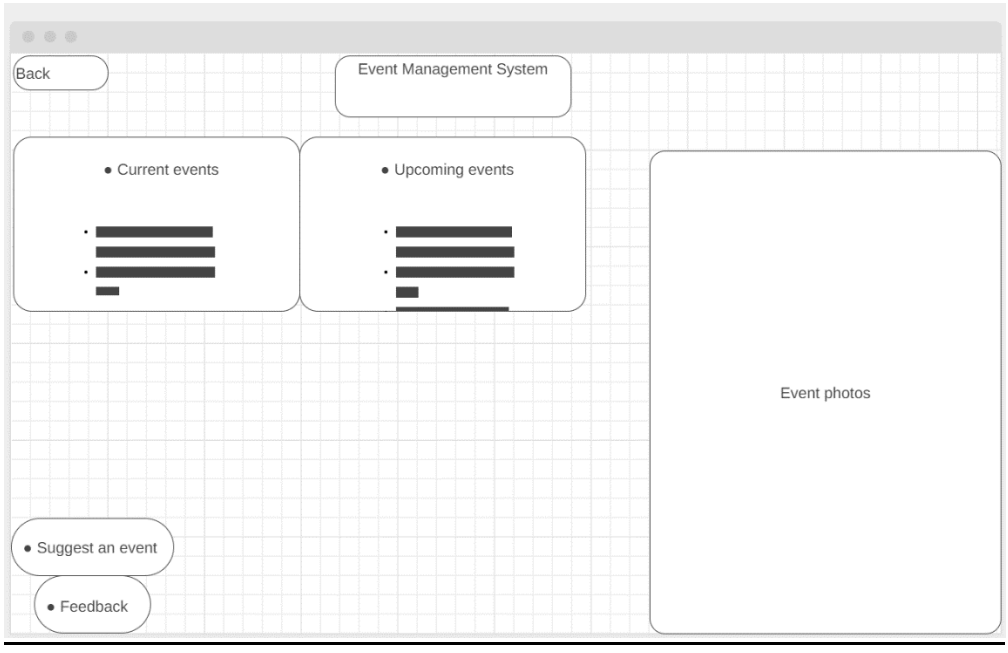

back
student photo
student infotmation
View Grades
details about student grades


Back
Research Management System
Research information
Research records
Reports
option details appear here


Back
Financial Management System
Tuition fees
Student billing
payroll and budgeting
Reports
Dashboard
option details appear here


Back
Faculty Information System
option information appear here
Faculty records
Faculty appointments
Reports
Dashboards


Back
Course information system
back
• Achived corses
• Total hours you signed
• view your timetable
details about all courses you have permission to sign in it


Back
Event Management System
• Current events
• Upcoming events
Event photos
• Suggest an event
• Feedback

User Manual:

Welcome to the University System User Manual! This guide is designed to help you navigate and utilize the many features of the university system.

Logging In:

To access the university system, you will need to log in with your university credentials. Your username and password should have been provided to you upon acceptance to the university. If you are having trouble logging in, please contact the IT help desk for assistance.

Dashboard:

Once you have logged in, you will be taken to your dashboard. Here you can access your course schedule, grades, financial aid information, and more. You can customize your dashboard to display the information that is most relevant to you.

Course Registration:

To register for courses, click on the registration link in your dashboard. Here you can view available courses and select the ones you wish to take. Be sure to pay attention to prerequisites and course requirements before registering.

Course Materials:

Once you have registered for your courses, you will need to access course materials. These can be found on the course page within the university system. Here you can view lecture notes, assignments, and other important course materials.

Grades:

Your grades will be posted within the university system. You can access your grades by clicking on the grades link in your dashboard. Be sure to check your grades regularly to stay on top of your academic progress.

Financial Aid:

If you have applied for financial aid, you can view the status of your application and awards within the university system. Be sure to check your financial aid status regularly to stay informed about your financial situation.

Communication:

The university system includes various communication tools that allow you to stay connected with your professors and classmates. These tools include email, discussion forums, and chat. Use these tools to ask questions, collaborate with classmates, and stay on top of important course announcements.

Help:

If you are having trouble using the university system, don't hesitate to reach out for help. Contact the IT help desk for technical assistance or your academic advisor for assistance with course-related questions.

We hope this user manual has been helpful in getting you started with the university system. Good luck with your studies!

Requirements		Student Management System:	Faculty Management System:	Course Management System:	Exam Management System:	Fee Management System:	Library Management System:	Housing Management System:	Transportation Management System
1		1	0	1	1	1	1	1	1
2		1	1	0	1	0	1	1	1
3		0	0	1	0	0	0	0	0
4		0	0	0	1	0	0	0	0
5		0	0	0	0	1	0	0	0
6		0	0	0	0	0	1	0	0
7		0	0	0	0	0	0	1	0
8		1	1	0	0	1	0	0	1
9		1	0	0	0	0	0	0	1
10		1	1	0	0	1	0	0	0
11		1	0	0	0	1	0	0	1
12		1	1	0	0	0	0	0	0
13		1	1	0	0	1	0	0	0
14		0	1	0	0	0	0	0	0
15		1	0	0	0	1	0	0	0
16		1	1	1	1	0	0	0	0
17		1	1	0	0	0	0	0	0
18		1	1	0	0	0	0	0	0
19		1	0	0	0	0	0	0	0
20		1	1	0	0	1	0	0	0
21		1	0	0	0	1	0	0	1
22		0	0	0	0	1	0	0	1
23		0	0	0	0	1	0	0	0
24		1	0	0	0	1	0	0	0
25		1	0	0	0	1	0	1	1
26		0	0	0	0	0	0	1	1
27		1	0	0	0	1	0	0	0