TUTORIAL

1. **MAKE A COPY OF THIS**
2. **CREATE THE CONTENT OF THE DOCUMENT**
3. **SEND IT TO MENTORS TO DECIDE**

ANALYSIS AND DESIGN

INFORMATION SYSTEM FOR MyUni

System Request

Name of the project	MyUni
Project sponsor	GFOSS
Business Need	MyUni aims to streamline university-related tasks, including student enrollment, course management, and academic tracking. The platform will provide an intuitive and centralized system to improve efficiency for students and administrators.
Business Requirements	● Robust login authentication system for students and administrators ● Scalable backend architecture ● Role-Based Access Control(RBAC) ● Student portal that will allow students access their personal information and academic records like their grades. ● Professors will be able to record students' grades on spreadsheets ● Admin panel that will allow administrators define appearance ,content and functionality of the application independently, depending on their institutions. ● Rewriting existing codes using Typescript(TS) for maintainability. ● Content Management System(CMS) for generating dynamic content and handling frequently changing information.
Business Value	● **Efficiency**: Automates manual processes, reducing workload for university administrators. ● **Accessibility:** Allows students to access their information , academic information like their grades in different subjects, schedules and other useful information ● **Transparency:** Provides real-time updates on students' grades and schedules ● **Scalability:** Supports growing university needs with modular features.
Specific issues or limitations	● Integration with existing university systems. ● Data security and compliance with education regulations. ● Training and onboarding for new users.

g

Feasibility Study

Technical feasibility	**Tech Stack:** ● **Programing Languages:** JavaScript(JS) and Typescript(TS) ● **Frameworks**: Express.js(for Backend), React.js and Next.js(for Frontend) ● **Database:** MySQL
Financial feasibility	● **Initial Development Cost:** Estimation of budget based on resource allocation. ● **Operational Costs:** Hosting, maintenance, and security. ● **Revenue Model:** Subscription-based or university-funded model.
Corporate feasibility	● Collaboration with universities and educational institutions. ● Alignment with university policies and academic workflows.
Other •Legal feasibility	● Compliance with **GDPR** and **FERPA** regulations for student data privacy. ● Licensing agreements for third-party integrations.

ANALYSIS

BUSINESS PROCEDURES

Add more procedures here. All the following are examples how the document will look like

1. User Registration
2. User Login
3. Profile Management
4. Data Viewing and Personalisation
5. CMS Data Management
6. Admin Panel Management
7. Student Portal information
8. Sharing application instances for multiple domains
9. **Role-Based Access Control(RBAC)**:Define access levels for students,admins and other roles.
10. **Course Enrollment & Management**: Allows students to enroll in courses and manage their enrollments
11. **Notifications & Alerts**: Send automated emails, push notifications, or SMS alerts for important updates and announcements.
12. **Feedback & Ratings System:** Enable users to provide feedback on courses or other services.
13. **Reporting & Analytics**: Generate reports for administrators on student engagement, course performance, and more.
14. **Multi-Tenancy Support**: Enable the system to support multiple institutions or organizations under one platform.
15. **Integration with External APIs**: Connect with third-party services like payment gateways, authentication providers, or learning management systems (LMS).
16. **Security & Compliance Management**: Ensure adherence to data protection laws, GDPR, and other relevant regulations.

UML activity diagrams of the state of the new information system (AS-IS)

Check myUoM system (https://my.uom.gr/) and create a UML diagram.

UML activity diagrams of the state of the new information system (TO-BE)

Below are the UML activity diagrams of the state of the new information system.

UML Activity diagrams for the following processes:

User Registration and Login

Profile Management

Data Viewing and Personalization

CMS Data Management

Admin Panel Management

Student Portal Information

Sharing application instances for multiple domains

User Stories

USER

1. As a user I want to register to have access to all the features of the application.
2. As a user I want to login to my account to view personalized information.
3. As a user I want to be able to change my password
4. As a user I want to manage my profile to ensure I can see appropriate personalized information.
5. As a user I want to use the application on my Mobile, Tablet or Desktop.
6. As a user I want to change the theme of the application e.g: Dark,Light or System
7. As a user I want to be able to change language so that I can use the language which I feel comfortable with
8. As a user, I want to get meaningful messages when the error happens in the system.Whether they are caused by system or by my actions
   eg: If I try to login with invalid credentials, I should get a meaningful error message back to know what has exactly happened.

ADMIN

1. As an Admin I want to login to manage all the students information, and website design.
2. As an Admin I want to be able to manage a CMS for fast-changing information.
3. As an Admin I want to create website pages or design through the backend.
4. As an Admin I want to change the color schemes for the different instances.
5. As an Admin I want to control which type of data is fetched from each source.
6. As an Admin I want to be able to have multiple users with different access levels.
7. As an Admin I want to be able to reset the password of the users.
8. As an administrator, I want to approve student course registrations so that I can ensure compliance with academic policies.
9. As an administrator, I want to generate reports on student enrollment and academic performance so that I can make data-driven decisions.
10. As an administrator, I want to manage faculty assignments so that I can balance workloads effectively.

EPICS

EPIC: User Authentication and Management

1. User Story: As a user, I want to register for an account with my university credentials so that I can access personalized services.
2. User Story: As a user, I want to log in securely to my account so that my personal data is protected.
3. User Story: As a user, I want to manage my profile so that I can update my personal information.
4. As a user , I want that my credentials will be kept securely in the database with no alteration and free from cyber-attacks

EPIC: CMS Data Management

1. User Story: As an admin, I want to manage a CMS for fast-changing information.
2. User Story: As an admin, I want to create website pages or design through the backend.
3. User Story: As an admin, I want to define and schedule the publication of content.

EPIC: Student Portal information

1. User Story: As a student, I want to view my grades so that I can understand my performance in my courses.

2. User Story: As a student, I want to be informed about campus events.
3. As a student, I want to register for courses online so that I can plan my academic schedule efficiently.
4. As a student, I want to view my grades so that I can track my academic progress.
5. As a student, I want to pay my tuition fees online so that I can complete the registration process easily.
6. As a student, I want to receive notifications about exam schedules and deadlines so that I don’t miss important dates.
7. As a student, I want to access library resources online so that I can borrow and return books conveniently.

EPIC: Theme and Style Management

1. User Story: As an admin, I want to change the color schemes for the different instances.
2. User Story: As an admin, I want to manage all the assets of the page.
3. User Story: As a user(student or admin), I want to get the UI based on my university.

EPIC: Data Configuration Management

1. User Story: As an Admin I want to control which type of data is fetched from each source.
2. EPIC: Users Access Levels Management
3. User Story: As an Admin I want to be able to have multiple users with different access levels.

EPIC: Access management

User Story: As an Admin I want to be able to reset the password of the users.
User Story: As a user, I want to access the application with all features working correctly every time I need to based on my role

Initial Information System Backlog

EPIC	STORY_ID	Description Story	MoSCoW Priority	Dependencies	Ideal days estimate	Sprint
User Authentication and Management	1	As a user, I want to register for an account with my university credentials.	Must have	None	3	1
User Authentication and Management	2	As a user, I want to log in securely to my account.	Must have	Story 1	2	1
User Authentication and Management	3	As a user, I want to manage my profile.	Should have	Story 1, Story 2	3	2
CMS Data Management	4	As an admin, I want to manage a CMS for fast-changing information.	Must have	None	5	3
CMS Data Managem	5	As an admin, I want to create	Should have	Story 4	8	5

ent		website pages or design through the backend.				
CMS Data Management	6	As an admin, I want to define and schedule the publication of content.	Must have	Story 4	5	3
Student Portal information	7	As a student, I want to view my grades.	Must have	Story 2	3	2
Student Portal information	8	As a student, I want to register for courses.	Should have	Story 2	5	3
Student Portal information	9	As a student, I want to be informed about campus events.	Must have	None	2	1
Theme and Style Management	10	As an admin, I want to change the color schemes for the different instances.	Must have	Story 4	3	2
Theme and Style Management	11	As an admin, I want to manage all the assets of the page.	Should have	Story 4	5	3
Data Configuration Management	12	As an Admin I want to control which type of data is fetched from each	Must have	None	3	2

		source.				
Users Access Levels Management	13	As an Admin I want to be able to have multiple users with different access levels.	Must have	None	3	2
Access management	14	As an Admin I want to be able to reset the password of the users.	Must have	None	2	1

Requirements analysis

Functional

A/A	Requirement	High/Medium/Low
1 User Authentication		
1.1	User registration using university credentials.	High
1.2	Secure login process with password encryption.	High
1.3	Password reset functionality.	High
2. Profile Management		
2.1	Update personal information (name, email, etc.).	High
2.2	Manage notification preferences.	Medium
3. CMS Data Management		
3.1	Create new content pages.	High
3.2	Edit existing content pages.	High
3.3	Schedule content publication.	High
4. Student Portal Information		
4.1	Display student grades.	High
4.2	Course registration functionality.	High
4.3	Display campus events.	High
5. Theme and Style Management		
5.1	Customize application color scheme.	High
5.2	Upload university logo and branding assets.	High
6. Users Access Levels Management		
6.1	Access configuration levels	High
6.2	Reset user Password	High
7. Data Configuration Management		

7.1	Select source of data to be fetched from each source.	High

Non functional

Operational

A/A	Requirement	High/Medium/Low
1	Usability: The system should be easy to use and navigate, with a user-friendly interface.	High
2	Reliability: The system must operate without errors or interruptions.	High
3	Adaptability: The user interface should be adaptive for use on mobile devices and tablets.	High
4	Scalability: The system must be scalable to support additional functions in the future.	Medium
5	Access management: the system must have capabilities to manage user roles and rights.	High
6	Interoperability: The system must be compatible and able to integrate with other systems and APIs.	Medium
7	Environmental performance: the system must be optimised for minimum resource consumption.	Low
8	Monitoring and logging: The system shall provide monitoring and logging capabilities to diagnose problems.	Medium

Performance

A/A	Requirement	High/Medium/Low
1	Response time: The application must respond in <2 seconds for all basic operations (e.g., user login, search for actions).	High
2	Loading Page: Most pages of the application should load in <1 second.	High
3	Concurrent Users: The application must support up to 5000 concurrent users without performance degradation.	High
4	Recovery Time from Error: The application must recover from errors within 5 seconds.	High
5	Data processing: processing large volumes of data (e.g. usage statistics) should be completed in <5 seconds.	Medium

6	Search efficiency: search results should be displayed in <1 second.	High
7	Interface Reaction: The user interface must react directly to user actions (e.g., clicking buttons, changing pages).	High
8	Cache Memory Performance: The use of cache memory should be optimized to minimize requests to the server.	Medium
9	Data Update Speed: Data updates (e.g. form submission, feedback) should be completed in <2 seconds.	Medium
10	Resource Efficiency: the application must be resource efficient (CPU, memory) to support optimal performance.	Medium
11	Disaster Recovery Time: The application must recover its data and operations within 4 hours of catastrophic events.	High
12	Minimize User Difficulty: Users should not experience delays or glitches when navigating the app.	High
13	Support for Multilingualism: The performance of the application should not be affected by supporting multiple languages.	Low
14	Upgradeability: The application must be able to be upgraded without interrupting performance.	Medium
15	Performance monitoring: a performance monitoring system should be in place to diagnose problems and improve performance.	Medium
16	Availability: The system must be available 99.9% of the time.	High
17	Maintenance: The system shall be designed for easy maintenance and upgrades.	Medium

Security

A/A	Requirement	High/Medium/Low
1	User authentication: The application must use secure authentication methods (e.g., OAuth, 2FA).	High
2	Data encryption: all sensitive data must be encrypted during storage and transfer.	High
3	Data Security: user data must be stored and transmitted using encryption (e.g. SSL/TLS).	High
4	SQL Injection Prevention: the application must be protected against SQL injection attacks through appropriate input processing.	High
5	Secure Password Management: User passwords must be	High

	stored using secure hashing algorithms (e.g. bcrypt).	
6	Session Management: User sessions should automatically expire after a predefined period of inactivity.	High
7	Strong Code Policy: Users must create codes that meet specific complexity requirements.	High
8	Secure Password Recovery: The password recovery process must include secure authentication steps.	High
9	Secure API Access: all API calls must be secure and require authentication.	High
10	Recording and Monitoring: The application must record all significant user actions and support the tracking of these.	Medium
11	Access Control: Users should only have access to the functions and data they are authorized to view or edit.	High
12	Mobile Data Security: Data transferred on mobile devices must be encrypted and protected.	Medium
13	Preventing Malicious Activities: the application must detect and prevent potentially malicious activities (e.g. brute force attacks).	High
14	Network Security: The application must be protected from network attacks by firewall and other network security methods.	High
15	Ensuring Data Integrity: the application data must be secured so that it is not corrupted or destroyed.	Medium
16	Penetration tests: The application shall be subjected to regular penetration tests to identify and repair vulnerabilities.	Medium
17	Privacy Policy: The application must have a clear privacy policy that informs users about the management of their data.	High
18	Secure Software Updates: Software updates must be performed in a secure manner to avoid introducing new vulnerabilities.	Medium

Cultural & Political

A/A	Requirement	High/Medium/Low
1	Multilingual Support: The application must support multiple languages, starting with Greek and English, to accommodate different users.	High
2	Accessibility: The application must be accessible to people with disabilities, following WCAG 2.1 AA standards.	High

3	GDPR compliance: the application must comply with the EU General Data Protection Regulation (GDPR).	High
4	Diversity and Inclusion: The application must support diversity and inclusion, ensuring that content and interaction is appropriate for all users.	Medium
5	Avoiding Hate Content: The application must prevent the publication and dissemination of content that promotes hatred, violence or discrimination.	High
6	Compliance with the ePrivacy Directive: the application must comply with the ePrivacy Directive.	Medium
7	Transparency: The application must provide clear and transparent information on the use of users' data.	High
8	User Participation Policy: The application should encourage active user participation in polls and surveys to improve services.	Low
9	Ethical Use of Data: the application must ensure that users' data is used ethically and only for the purposes stated.	High
10	Support for Local Laws: the application must comply with local laws and regulations in each region where it operates.	High
11	User Content Policy: The application should have a policy for managing user-generated content, ensuring that it is appropriate and secure.	Medium
12	User Education: The application should provide resources and guidance to educate users on security and usage policies.	Low
13	Crisis management: the application must have a crisis management plan to deal with possible emergencies or security breaches.	Medium
14	User Acceptance Policy: The application should include clear terms and conditions for its use, as well as procedures for dealing with violations.	High
15	Anonymity policy: The application must allow users to use the services anonymously where possible and appropriate.	Medium
16	Policy flexibility: The application must be able to adapt to changes in policies and regulations, incorporating new requirements quickly and efficiently.	Medium
17	Consent Management Policy: The application must manage user consent for data collection and use by providing easy procedures for withdrawing consent.	High
18	Cooperation with Supervisory Authorities: The application shall cooperate with supervisory authorities to ensure compliance with regulations.	Medium

19	Informing Users about Policy Changes: Users should be notified in a timely manner of any changes to the application's policies.	Medium

UML use case diagram

Below is the UML use case diagram.

Verbal descriptions and indicative screens

1. User Registration

Description:

The user registers on the platform by creating a new account.

Regular Flow:

1. The user selects "Register".
2. Fill in the required fields: email, first name, last name, city of residence, password.
3. Confirms that he/she agrees with the terms and conditions.
4. The user clicks on the "Register" button.
5. The system checks if all fields are filled in correctly and if the email has not been used before.
6. The system creates the new account and sends a confirmation email to the user.

Alternative Flow:

5.1 If the data is not filled in correctly or the email has been used, the system displays the message "The data is not correct. Please correct it and try again".
5.2 The UC continues from step 2 of the regular flow.

Sub-Flow:

- If the user selects "Cancel", the user is returned to the home page.

2. User Login

Description:

User logs in.

Regular Flow:

1. User clicks "Login."
2. User enters credentials (email, password).
3. User clicks "Submit."
4. System verifies credentials and logs the user in.

Alternative Flow:

4.1 If the data is wrong, the system displays an error message and asks the user to correct it.
4.2 The UC continues from step 2 of the regular flow.

Sub-Flow:

- If the user selects "Reset password", the user goes to the reset password procedure (UC 3).

3. Reset Password

Description:

The user resets his password when he does not remember the login credentials.

Regular Flow:

1. The user receives an error message when logging into their account (UC 2).
2. The user selects "Reset password".
3. The user fills in their email.
4. The user clicks the "Send" button.
5. The system checks the email and, if it is correct, sends a password reset email to the user.

Alternative Flow:

5.1 If the email is not correct, the system displays the message "Your email is not correct. Please enter the correct email".
5.2 The UC continues from step 3 of the regular flow.

Sub-Flow:

- If the user selects "Cancel", the user is returned to the login page.

4. User Profile Management

Description:

The user manages his profile by updating his personal information.

Regular Flow:

1. The user logs in to his/her account (UC 2).
2. The user selects "Profile".
3. The user updates the fields: first name, surname, city of residence, profile photo.
4. The user presses the "Save" button.
5. The system updates the profile data and displays a confirmation message.

5. Account deletion

Description:

The user deletes his/her account from the application.

Regular Flow:

1. The user logs in to his/her account (UC 2).
2. The user selects "Profile".
3. The system returns the user's profile.
4. The user selects "Delete Account".
5. The user confirms the action.
6. The system deletes the account and all user data.

Alternative Flow:

5.1 If the user cancels the action, the user is returned to the previous page.

5.2 The UC continues from step 3 of the regular flow.

6. CMS Content Creation

Description:
Admin creates new content pages.

Regular Flow:

1. Admin logs in.
2. Admin navigates to CMS section.
3. Admin clicks "Create New Page."
4. Admin fills in title, content, and other metadata.
5. Admin clicks "Publish".
6. System publishes to be visible to the users.

7. Push Notifications Management

Description:
The administrator manages the platform notifications.

Regular Flow:

7. The administrator logs in to his/her account (UC 2).
8. The system displays the control panel.
9. The administrator selects 'Notifications'.
10. The administrator creates or edits alerts.
11. Specifies the recipients and the time of sending.
12. The system sends the notifications to the users.

38 User Logout

Description:
The user is logged out of the platform.

Regular Flow:

1. The user selects "Logout".
2. The system logs the user out and returns him to the home page.

9. Reset User Password by Administrator

Description:
The administrator resets a user's password.

Regular Flow:

1. The administrator enters his/her account (UC 2).
2. The system displays the control panel.
3. The administrator selects "Manage Users".
4. The administrator selects the user who needs a password reset.
5. The administrator presses the "Reset Password" button.
6. The system generates a new password and sends it to the user via email.

10. User Support

Description:
The administrator provides support to the users of the platform.

Regular Flow:

1. The administrator enters his/her account (UC 2).
2. The system returns the dashboard.
3. The administrator selects "Support".
4. A list of support requests is displayed.
5. The administrator selects a request to view details.
6. The administrator responds to the request and records the actions taken.
7. The system informs the user and closes the request.

11. Integration with Social Networks

Description:
The user connects their platform account with social networks for easy sharing.

Regular Flow:

1. The user logs in to his/her account (UC 2).
2. The user selects "Profile".
3. The user selects "Connect with Social Networks".
4. The user selects the social network and confirms the connection.
5. The system links the account and informs the user.

12. Connecting to External APIs

Description:
The platform connects to external APIs for additional functionality.

Regular Flow:

1. The administrator enters his/her account (UC 2).
2. The system returns the dashboard.
3. Admin selects "Connect with API".
4. Enter the necessary API keys and parameters.
5. The system checks the connection and informs the administrator.

Alternative Flow:
5.1 If the administrator has entered the wrong API key or the service is down, the system displays a message “Unable to connect to external service using API”.
5.2 The UC continues from step 4 of the regular flow.

13. Event and Announcement Notifications

Description:
The user receives notifications about new events or announcements on the platform.

Regular Flow:

1. The administrator creates a new event or announcement.
2. The administrator selects the audience (students, teachers, or all users).

3. The administrator publishes the event/announcement.
4. The system sends push notifications, emails, or in-app notifications to relevant users.
5. The user receives the notification and clicks on it to view details.

Alternative Flow:
5.1 If the user has disabled notifications, they will not receive real-time alerts but can still access announcements in the platform’s event section.

14. Student Marks and Course Results

Description:
Students can view their marks and results for different courses in their portal.

Regular Flow:

1. The student logs in to their account (UC 2).
2. The student navigates to the "Results" or "My Courses" section.
3. The system displays a list of enrolled courses.
4. The student selects a course to view marks and final results.
5. The system retrieves and displays the grades, including assessments and final scores.

Alternative Flow:
5.1 If results are not yet available, the system displays a message: “Your results are not yet published. Please check back later.”

15. Sharing Application Instances for Multiple Domains

Description:
The system allows institutions to run multiple instances of the platform under different domains.

Regular Flow:

1. The administrator logs in to the platform (UC 2).
2. The administrator selects "Instance Management" from the control panel.
3. The administrator enters details such as the domain name, institution name, and configuration preferences.
4. The system validates the domain and provides a separate instance.
5. The administrator customizes the instance settings and launches it for users.

Alternative Flow:
5.1 If the domain is invalid or already in use, the system displays a message: "The selected domain is unavailable. Please choose another domain."