TA Management System

Deliverable-1 1st Iteration

02.03.2025

Course: CS 319

Section: 01

Group No.: 7

Instructor: Eray Tüzün

Teaching Assistant: Yahya Elnouby

Group Members: Bilge İdil Öziş [22102365]

Burak Kağan Gür [22203365]

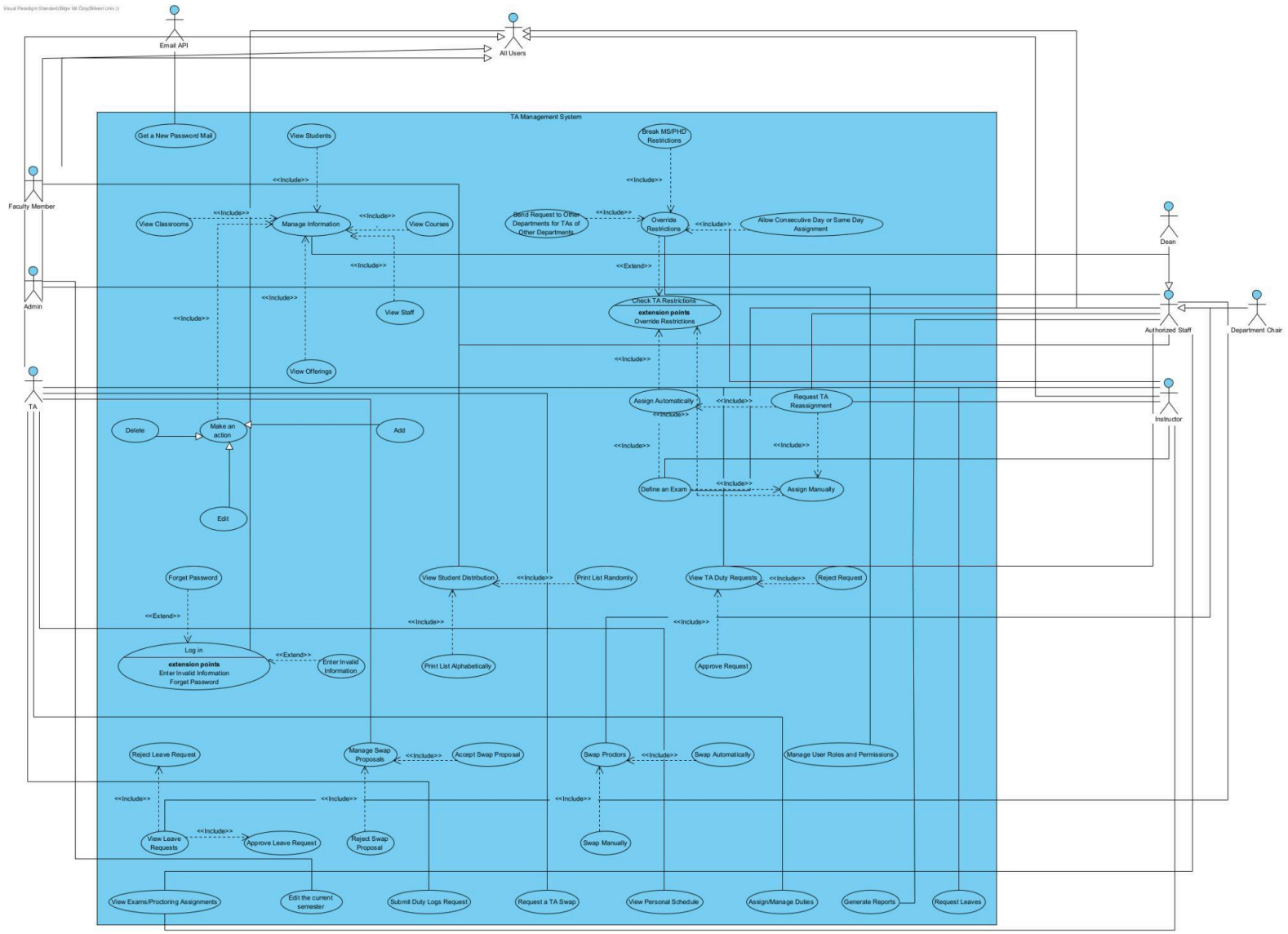
Daib Malik [22201013]

Yunus Günay [22203758]

Yusufbek Karamatov [22301586]

Contents

1. Use-Case Diagram



2. Use-Case Textual Description

2.1 Authentication & Home Page Package

Use Case Name: Login

1. Participating Actor: TA Management System User
2. Flow of Events:

   2.1 The actor opens the website.

   2.2 The actor enters their email and password.

   2.3 The actor presses the login button on the main page.

3. Entry Condition: The actor should be authorized to log in.
4. Exit Condition: The actor should enter their credentials correctly.
5. Quality Requirements: -

Use Case Name: Enter Invalid Information

1. Participating Actor: TA Management System User
2. Flow of Events:

   2.1 The actor opens the website.

   2.2 The actor enters their login credentials and presses the login button on the main page.

   2.3 (Extending the Login Case) If the credentials do not match, the system displays an error message.

3. Entry Condition: The actor should enter their information incorrectly.
4. Exit Condition: An error is shown telling the user about the incorrect login attempt.
5. Quality Requirements: The system should detect when an email does not exist or password does not match with the email, and provide a clear, user-friendly error message.

Use Case Name: Forget Password

1. Participating Actor: TA Management System User
2. Flow of Events:

   2.1 The actor clicks on the "Forgot Password?" link on the login page.

   2.2 The system prompts the actor to enter their registered email address.

   2.3 The actor enters their email and submits a request.

   2.4 The system checks if email exists. If the email is valid, the system generates a password reset link and sends an email (extends New Password Mail). If the email is not found, the system displays an error message.

   2.5 The actor is informed that an email has been sent (if valid).

3. Entry Condition: The actor clicks the "Forgot Password?" button on the login page.
4. Exit Condition: The system sends a password reset email or informs the user of an issue.

5. Quality Requirements: The password reset link should be secure and time-limited and the system should ensure that no security-sensitive information is exposed in error messages.

Use Case Name: Get a New Password Mail

1. Participating Actor: Email API
2. Flow of Events:

   2.1 The system sends an email to the actor containing a message explaining the reset process and a secure link to reset the password.

   2.2 The actor receives the email and clicks the reset link.

   2.3 The password of the user is changed with the new one.

3. Entry Condition: The actor whose email is registered in the system requests a password reset.
4. Exit Condition: The password of the actor is changed.
5. Quality Requirements: The email should be sent securely and the system should limit the number of failed attempts to prevent brute-force attacks.

2.2 TA Duties Package

Use Case Name: Assign/Manage Duties

1. Participating Actor: Authorized Staff, Instructor (for specific courses)
2. Flow of Events:

   2..1 The actor logs into the system.

   2.2 The actor views existing duty assignments for TAs or creates a new one.

   2.3 The actor enters details (course, task type, date/time, expected duration).

   2.4 The system checks for conflicts (schedule, TA availability, workload caps).

   2.5 The actor confirms the assignment, and the system notifies the assigned TA and updates records.

3. Entry Condition: The actor has the necessary role/permissions and TAs exist in the system with the assigned courses.
4. Exit Condition: A new duty is assigned or an existing assignment is updated.
5. Quality Requirements: The system must prevent double-booking and enforce workload limits.

Use Case Name: Submit Duty Logs Request

1. Participating Actor: TA
2. Flow of Events:

2.1 TA logs into the system and selects the course and the type of the duty (lab, grading, office hours, etc.)

2.2 TA enters date/time and duration spent on the duty.

2.3 TA submits the duty log request, and the system notifies the corresponding Instructor or Staff for approval.

3. Entry Condition: TA is assigned to at least one course.
4. Exit Condition: The duty log is recorded and pending approval.
5. Quality Requirements: The system should validate the data (non-negative duration, correct date format, etc.) and it should handle multiple submissions without duplicates.

Use Case Name: View TA Duty Requests

1. Participating Actor: Instructor or Authorized Staff
2. Flow of Events:

   2.1 The actor logs into the system and goes to the requests page.

   2.2 The system displays a list of duty logs submitted by TAs.

3. Entry Condition: TAs have submitted duty logs.
4. Exit Condition: The actor can view and proceed to approve or reject requests.
5. Quality Requirements: The system must show real-time status (approved, rejected, pending) and provide relevant details.

Use Case Name: Define an Exam

1. Participating Actor: Instructor, Authorized Staff
2. Flow of Events:

   2.1 The actor navigates to the exam management section.

   2.2 The actor enters exam details (course, date, time, duration, location).

   2.3 The actor specifies the number of proctors needed.

   2.4 The system saves the exam information.

3. Entry Condition: The actor has appropriate permissions.
4. Exit Condition: New exam is created in the system.
5. Quality Requirements: The system should check for scheduling conflicts with other exams or class activities.

Use Case Name: View Exams/Proctoring Assignments

1. Participating Actor: Instructor, Authorized Staff, or Department Chair
2. Flow of Events:

   2.1 The actor navigates to the exams/proctoring assignments page.

   2.2 The system displays upcoming exams, assigned TAs, and any open proctoring slots.

3. Entry Condition: Exams and proctoring data must exist in the system.

4. Exit Condition: The actor reviews exam schedules and proctor assignments.
5. Quality Requirements: The interface should clearly distinguish between filled and unfilled proctor positions.

2.3 Information Management Package

Use Case Name: View Students

1. Participating Actor: Instructor, Authorized Staff
2. Flow of Events:

   2.1 The actor navigates to the student management section.

   2.2 The system displays a list of students with relevant information.

   2.3 The actor can filter or search for specific students.

3. Entry Condition: The actor has appropriate permissions.
4. Exit Condition: Student information is displayed.
5. Quality Requirements: The system should provide efficient filtering and sorting options.

Use Case Name: View Courses

1. Participating Actor: TA Management System User
2. Flow of Events:

   2.1 The actor navigates to the courses section.

2.2 The system displays a list of courses based on the user's role.

2.3 The actor can select a course to view more details.

3. Entry Condition: The actor has permissions to view course information.
4. Exit Condition: Course information is displayed according to user role.
5. Quality Requirements: The interface should organize courses by semester and department.

Use Case Name: View Classrooms

1. Participating Actor: Instructor, Authorized Staff
2. Flow of Events:

   2.1 The actor navigates to the classroom management section.

   2.2 The system displays a list of available classrooms with capacity and equipment information.

   2.3 The actor can filter or search for specific rooms.

3. Entry Condition: The actor has appropriate permissions.
4. Exit Condition: Classroom information is displayed.
5. Quality Requirements: The system should indicate current and upcoming room availability.

Use Case Name: View Staff

1. Participating Actor: Authorized Staff, Department Chair

2. Flow of Events:

   2.1 The actor navigates to the staff management section.

   2.2 The system displays a list of staff members with relevant information.

   2.3 The actor can filter or search for specific staff members.

3. Entry Condition: The actor has appropriate permissions.
4. Exit Condition: Staff information is displayed.
5. Quality Requirements: Information should be organized by department and role.

Use Case Name: View Offerings

1. Participating Actor: TA Management System User
2. Flow of Events:

   2.1 The actor navigates to the course offerings section.

   2.2 The system displays current and upcoming course offerings.

   2.3 The actor can view details including instructors and assigned TAs.

3. Entry Condition: The actor has permissions to view offerings.
4. Exit Condition: Course offering information is displayed.
5. Quality Requirements: The interface should clearly show the status of TA assignments for each offering.

Use Case Name: Manage Information

1. Participating Actor: Authorized Staff, Department Chair
2. Flow of Events:

   2.1 The actor navigates to the information management section.

   2.2 The actor selects the type of information to manage (courses, classrooms, students, staff).

   2.3 The actor can view, add, edit, or delete information based on their permissions.

3. Entry Condition: The actor has appropriate permissions.
4. Exit Condition: Information is updated in the system.
5. Quality Requirements: Changes should be logged for audit purposes.

Use Case Name: Add

1. Participating Actor: Authorized Staff, Department Chair
2. Flow of Events:

   2.1 The actor selects the appropriate section (courses, students, staff, etc.).

   2.2 The actor clicks the "Add" button.

   2.3 The actor fills out the required information in the form.

   2.4 The system validates the information.

   2.5 The actor confirms, and new information is added to the system.

3. Entry Condition: The actor has permission to add information.
4. Exit Condition: New information is successfully added.
5. Quality Requirements: The system should check for duplicates and data integrity.

Use Case Name: Edit

1. Participating Actor: Authorized Staff, Department Chair
2. Flow of Events:

   2.1 The actor selects an existing record.

   2.2 The actor clicks the "Edit" button.

   2.3 The actor modifies information in the pre-filled form.

   2.4 The system validates the changes.

   2.5 The actor confirms, and the information is updated.

3. Entry Condition: The record exists and the actor has permission to edit.
4. Exit Condition: Information is updated in the system.
5. Quality Requirements: The system should maintain data history and audit trails.

Use Case Name: Delete

1. Participating Actor: Authorized Staff, Department Chair
2. Flow of Events:

   2.1 The actor selects an existing record.

2.2 The actor clicks the "Delete" button.

2.3 The system confirms the deletion request.

2.4 The actor confirms, and the record is removed or marked as inactive.

3. Entry Condition: The record exists and the actor has permission to delete.
4. Exit Condition: The record is removed from active use.
5. Quality Requirements: The system should check for dependencies before deletion.

Use Case Name: Make an Action

1. Participating Actor: TA Management System User
2. Flow of Events:

   2.1 The actor navigates to the relevant section.

   2.2 The actor selects an appropriate action (add, edit, delete, approve, reject, etc.).

   2.3 The actor provides necessary information or confirmation.

   2.4 The system processes the action and updates relevant records.

3. Entry Condition: The actor has permission to perform the action.
4. Exit Condition: The action is completed successfully.
5. Quality Requirements: The system should provide clear feedback about action results.

2.4 Request Management Package

Use Case Name: Request a TA Swap

1. Participating Actor: TA
2. Flow of Events:

   2.1 The actor logs into the system and goes to the duties page.

   2.2 The actor selects an assigned duty or proctoring slot they want to swap.

   2.3 The actor initiates a swap request, choosing another TA as the proposed swap partner.

   2.4 The system sends a swap proposal to the selected TA.

   2.5 If the second TA accepts, the system reassigns duties and updates workloads.

   2.6 Both TAs and any relevant instructor/staff are notified.

3. Entry Condition: TA has at least one active assignment.
4. Exit Condition: Duty/proctoring assignment is swapped or the request is declined.
5. Quality Requirements: The system must prevent scheduling conflicts or exceeding workload caps, and maintain a log of swaps to avoid repeated or cyclical swapping.

Use Case Name: Request Leaves

1. Participating Actor: TA
2. Flow of Events:

2.1 TA logs into the system and goes to the leaves page.

2.2 TA enters the date range and reason for leave (medical, conference, etc.).

2.3 TA submits the request and the system notifies the Department Chair or Authorized Staff.

3. Entry Condition: TA must be logged in.
4. Exit Condition: Leave request is recorded and pending approval/rejection.
5. Quality Requirements: The system must block assignment of duties during approved leave dates.

Use Case Name: Approve Request

1. Participating Actor: Instructor, Authorized Staff, or Department Chair
2. Flow of Events:

   2.1 The actor logs into the system and goes to the requests page.

   2.2 The actor reviews details of a request (duty log, leave, swap, assignment). 2.3 The actor clicks the approve button.

   2.4 The system updates the request status to "Approved."

   2.5 TAs are notified.

3. Entry Condition: A request is awaiting approval.
4. Exit Condition: Request status is updated to "Approved."

5. Quality Requirements: The system should log approval actions with timestamp and approver.

Use Case Name: Reject Request

1. Participating Actor: Instructor, Authorized Staff, or Department Chair
2. Flow of Events:

   2.1 The actor logs into the system and goes to the requests page.

   2.2 The actor reviews details of a request (duty log, leave, swap, assignment).

   2.3 The actor clicks the reject button and optionally provides a reason.

   2.4 The system updates the request status to "Rejected."

   2.5 TAs are notified.

3. Entry Condition: A request is awaiting approval.
4. Exit Condition: Request status is updated to "Rejected."
5. Quality Requirements: The system should require a reason for rejection.

Use Case Name: Manage Swap Proposals

1. Participating Actor: TA (receiving or initiating a swap)
2. Flow of Events:

2.1 TA receives a notification of a swap proposal.

2.2 TA reviews the proposed swap details (duty type, date/time, etc.).

2.3 TA accepts or rejects the proposal.

2.4 If accepted, the system updates both TAs' workloads.

2.5 Notifications are sent to both TAs and relevant staff.

3. Entry Condition: A swap request must have been initiated.
4. Exit Condition: The swap is completed or declined.
5. Quality Requirements: The system ensures no scheduling or workload conflicts.

Use Case Name: View Personal Schedule

1. Participating Actor: TA, Instructor
2. Flow of Events:

   2.1 The actor logs into the system and navigates to the personal schedule section.

   2.2 The system displays the actor's assigned duties, classes, and other commitments.

   2.3 The actor can filter by date range or duty type.

3. Entry Condition: The actor has a valid user account.
4. Exit Condition: The actor views their schedule.
5. Quality Requirements: The schedule should be visually clear and exportable.

Use Case Name: Manage Leave Request

1. Participating Actor: TA
2. Flow of Events:

   2.1 TA logs into the system and navigates to the leave management section.

   2.2 TA views current and past leave requests.

   2.3 TA can create new requests, modify pending requests, or cancel requests.

3. Entry Condition: TA has a valid user account.
4. Exit Condition: Leave request is created, modified, or canceled.
5. Quality Requirements: The system should prevent overlapping leave requests.

Use Case Name: View Approved Leaves

1. Participating Actor: Instructor, Authorized Staff, Department Chair
2. Flow of Events:

   2.1 The actor navigates to the leave management section.

   2.2 The system displays all approved leaves, filterable by date and TA.

   2.3 The actor can view details of each leave.

3. Entry Condition: The actor has appropriate permissions.

4. Exit Condition: Approved leaves are displayed.
5. Quality Requirements: The view should highlight leaves that affect upcoming duties.

Use Case Name: Submit Swap Proposal

1. Participating Actor: TA
2. Flow of Events:

   2.1 TA selects a duty they want to swap.

   2.2 TA chooses another TA to propose the swap to.

   2.3 TA submits the proposal with optional details/reason.

   2.4 The system notifies the selected TA.

3. Entry Condition: TA has an assigned duty eligible for swapping.
4. Exit Condition: Swap proposal is sent to another TA.
5. Quality Requirements: The system should verify eligibility of both TAs.

Use Case Name: Manage User Roles and Permissions

1. Participating Actor: Department Chair, System Administrator
2. Flow of Events:

   2.1 The actor navigates to the user management section.

   2.2 The actor selects a user to modify.

   2.3 The actor assigns or modifies roles and permissions.

2.4 The system updates the user's access rights.

3. Entry Condition: The actor has administrator privileges.
4. Exit Condition: User's roles and permissions are updated.
5. Quality Requirements: The system should enforce role separation and least privilege.

2.5 Assignment Management Package

Use Case Name: Assign Automatically

1. Participating Actor: Instructor, Authorized Staff, or Department Chair
2. Flow of Events:

   2.1 The actor logs in and chooses an exam or duty to assign TAs automatically.

   2.2 The actor sets the parameters (number of TAs needed, course, exam type, date/time).

   2.3 The system analyzes TAs' workload, availability, and restrictions.

   2.4 The system proposes an assignment list, prioritizing TAs with least workload and relevant qualifications.

   2.5 The actor confirms or modifies the proposal.

3. Entry Condition: The exam or duty details must be valid and the user has appropriate permissions..
4. Exit Condition: TAs are automatically assigned.

5. Quality Requirements: The system must respect departmental rules and workload caps.

Use Case Name: Assign Manually

1. Participating Actor: Instructor, Authorized Staff
2. Flow of Events:

   2.1 The actor navigates to the assignment section.

   2.2 The actor selects a specific duty, course, or exam.

   2.3 The actor views eligible TAs and their current workload.

   2.4 The actor manually selects TAs for the assignment.

   2.5 The system validates the selection and confirms the assignment.

3. Entry Condition: The actor has appropriate permissions.
4. Exit Condition: TAs are assigned to the specified duty.
5. Quality Requirements: The system should prevent assignments that violate constraints.

Use Case Name: Request TA Reassignment

1. Participating Actor: Instructor
2. Flow of Events:

   2.1 The instructor navigates to the TA assignments for their course.

2.2 The instructor selects a TA assignment they wish to change.

2.3 The instructor submits a reassignment request with justification.

2.4 The system forwards the request to the Authorized Staff or Department Chair.

3. Entry Condition: The instructor has an assigned TA.
4. Exit Condition: Reassignment request is submitted.
5. Quality Requirements: The system should track reassignment frequency and reasons.

Use Case Name: Override Restrictions

6. Participating Actor: Authorized Staff
7. Flow of Events:

2.1 The actor goes to the proctor assignment page for a specific exam.

2.2 The actor can not find the required number of TAs for that exam.

2.3 The actor makes an override to qualify the required number of TAs for that exam.

8. Entry Condition: The exam is already defined in the system.
9. Exit Condition: The system breaks the TA restrictions.
10. Quality Requirements: The system should not allow overriding restrictions unless there is need to.

Use Case Name: Process TA Assignments

1. Participating Actor: Department Chair, Authorized Staff
2. Flow of Events:

   2.1 The actor navigates to the TA assignment processing section.

   2.2 The system displays courses needing TA assignments.

   2.3 The actor can manually assign or use automatic assignment.

   2.4 The system validates assignments against constraints.

   2.5 The actor confirms finalized assignments.

3. Entry Condition: Courses requiring TAs must be defined in the system.
4. Exit Condition: TAs are assigned to courses.
5. Quality Requirements: The system should optimize for balanced workloads.

Use Case Name: View/Distribute Student List

1. Participating Actor: Instructor, TA
2. Flow of Events:

   2.1 The actor navigates to the course section.

   2.2 The actor selects a specific course.

   2.3 The system displays the student list.

2.4 For instructors, an option to distribute students among TAs is available.

3. Entry Condition: The actor is associated with the course.
4. Exit Condition: Student list is viewed or distributed.
5. Quality Requirements: Distribution should be balanced and configurable.

Use Case Name: Post List Announcement

1. Participating Actor: Authorized Staff
2. Flow of Events:

   2.1 The actor navigates to the announcement section.

   2.2 The actor creates a new announcement regarding TA assignments.

   2.3 The actor uploads or links to assignment lists.

   2.4 The actor publishes the announcement.

3. Entry Condition: TA assignments are finalized.
4. Exit Condition: Assignment announcement is published.
5. Quality Requirements: Announcements should be accessible to all relevant users.

2.6 Reporting & User Management Package

Use Case Name: Generate Reports

1. Participating Actor: Authorized Staff, Department Chair
2. Flow of Events:

   2.1 The actor navigates to the reporting section.

   2.2 The actor selects the type of report (workload, assignments, duty logs).

   2.3 The actor specifies parameters (date range, courses, TAs).

   2.4 The system generates and displays the report.

   2.5 The actor can export the report in various formats.

3. Entry Condition: The actor has permission to access reports.
4. Exit Condition: Report is generated and displayed/exported.
5. Quality Requirements: Reports should be accurate, comprehensive, and exportable.

Use Case Name: View User Duties and Permissions

1. Participating Actor: Authorized Staff, Department Chair
2. Flow of Events:

   2.1 The actor navigates to the user management section.

   2.2 The actor selects a user to view.

   2.3 The system displays the user's current duties, roles, and permissions.

3. Entry Condition: The actor has appropriate permissions.
4. Exit Condition: User information is displayed.
5. Quality Requirements: The interface should clearly distinguish between different roles.

Use Case Name: Manage User Roles and Permissions

1. Participating Actor: Department Chair, System Administrator
2. Flow of Events:

   2.1 The actor navigates to the user management section.

   2.2 The actor selects a user to modify.

   2.3 The actor assigns or modifies roles and permissions.

   2.4 The system updates the user's access rights.

3. Entry Condition: The actor has administrator privileges.
4. Exit Condition: User's roles and permissions are updated.
5. Quality Requirements: The system should enforce role separation and least privilege.

Use Case Name: Send Personal Invitation

1. Participating Actor: Authorized Staff, Department Chair
2. Flow of Events:

   2.1 The actor navigates to the user management section.

   2.2 The actor selects the option to invite a new user.

2.3 The actor enters the invitee's email and role.

2.4 The system generates and sends an invitation.

3. Entry Condition: The actor has permission to invite users.
4. Exit Condition: Invitation is sent to potential users.
5. Quality Requirements: Invitations should have secure, time-limited links.

Use Case Name: New Coordinator/Exam or Same Day Assignment

1. Participating Actor: Department Chair, Authorized Staff
2. Flow of Events:

   2.1 The actor navigates to the assignments section.

   2.2 The actor creates a new coordinator position or same-day assignment.

   2.3 The actor specifies details and requirements.

   2.4 The actor assigns a TA or leaves it open for application.

3. Entry Condition: The actor has appropriate permissions.
4. Exit Condition: New position/assignment is created.
5. Quality Requirements: The system should check for scheduling conflicts.

Use Case Name: Manage/View Roles and Permissions

1. Participating Actor: Department Chair, System Administrator
2. Flow of Events:

2.1 The actor navigates to the role management section.

2.2 The actor can view existing roles and their permissions.

2.3 The actor can create new roles or modify existing ones.

2.4 The system updates the role configuration.

3. Entry Condition: The actor has administrator privileges.
4. Exit Condition: Roles and permissions are updated.
5. Quality Requirements: The system should prevent circular dependencies in role hierarchies.

Conclusion: We designed each of these packages to organize related functionality within the TA Management System, making the system more modular and easier to understand.

3. Tech Stack

Backend:

We are planning to use **Flask** as our main backend framework. It is a lightweight tool that lets us quickly set up RESTful APIs and handle server-side logic. Moreover, it is easy to learn, which helps our team to get rolling fast. Flask also allows for easy integration of libraries and tools for authentication, security, and other middleware technologies.

Frontend:

The frontend of the application will be developed using **React**. React's component-based architecture allows for modular and reusable code, facilitating easier maintenance and scalability. There's also lots of resources and community support available.

Database:

For data management, **MySQL** will be used. It offers efficient handling of complex queries, and ensures data integrity. It is also a solid choice for pulling data from Excel files. This makes importing and managing data straightforward and reliable.

Version Control:

Our development workflow will be managed with **Git**. It ensures that every change is tracked, facilitating code integration and effective management of different development branches. Besides, we are all familiar with using it.