



Software Engineering IT314
Project: Student leave and TA assistantship
management
Group: 9

● **Group Members:**

- Preksha Anand - 202001061
- Maharth Thakar - 202001069
- **Meet Patel - 202001074**
- Kartik Dangi - 202001079
- Harsh Kathiriya - 202001088
- Kuldipsinh Gohil - 202001092
- Neha Miglani - 202001097
- Harsh Anand - 202001101
- Darshan Kheni - 202001109

Lab : 06

Date : 22-03-2023

Domain Analysis

INTRODUCTION:

The domain would be the management of the requested leaves within the institute.

❖ Entity Objects

1. **Student:** A person who is enrolled in one or more courses in the university who can request for leave of absence and also for TAsip.
2. **Faculty:** A person who can approve or deny the request of TAsip made to them of their specific course.
3. **HOD:** A person who is responsible for approving the request of leave of absence made to them by a student, TA or faculty.
4. **TA:** A person who is doing TAsip in a specific course and can request for leave to HOD.
5. **Leave Request:** A formal request submitted by student take leave of absence for a certain period of time
6. **Leave Approval:** A decision made by HOD to approve or deny the request made by student, TA or faculty
7. **TAsip request:** A request made by student to faculty for doing TAsip in their course
8. **TAsip Approval:** A decision made by Faculty to approve or deny the request of TAsip made by student

❖ Attributes

1. **Student:** Student ID, Name, Email
2. **Faculty:** Faculty ID, Name, Email, Courses
3. **HOD:** HoD ID, Name, Email
4. **TA:** TA ID, Name, Email, Course
5. **Leave Request:** Request Date, Status, Reason
6. **Leave Approval:** Approval Date
7. **TAship request:** Request Date, Status, Course
8. **TAship approval:** Approval Date

❖ Boundary Objects

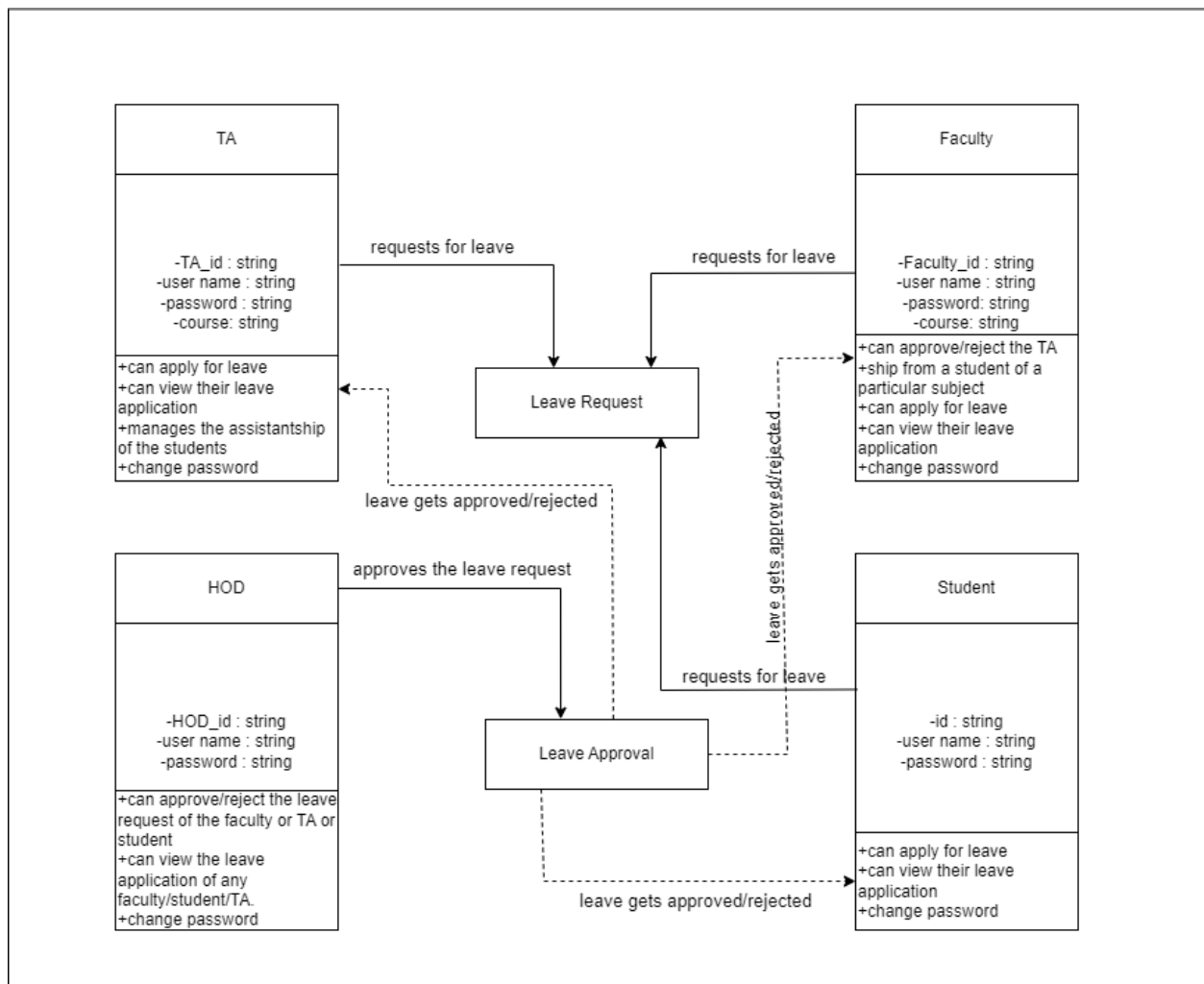
- Boundary object is an interface between a system and external entities such as faculty, student, HOD or TA.
1. **Leave Application From:** This is an input boundary object which captures information provided by students, faculty or TA. It may contain fields of start date and end date for leave, reason for leave, name and ID of student.
 2. **Leave Approval Notification:** This is an output boundary object which notifies the students of the decision made by HOD. It may contain status of request, reason for approval or denial, and other information.
 3. **TAship Application Form:** This is an input boundary object which stores information provided by students to become TA in a specific course. It may contain academic information about students, request date, eligibility information, course name, etc.
 4. **TAship Approval Notification:** This is an output boundary object which notifies students of the decision made by faculty. It may contain status of request, reason for approval or denial, and other information.

❖ Control Objects

- The control object represents the internal processes and rules that govern the behavior of the system. It coordinates the interactions between different objects and manages the flow of data within the system.
- Some examples of control objects in this project could be:
 1. **Leave Request Manager:** This control object is responsible for receiving and processing leave requests submitted by students. It may validate the inputs, store the data in the appropriate format, and initiate the review process by assigning the request to the HoD or faculty member.
 2. **Leave Approval Manager:** It is responsible for managing the review and approval process for leave requests. It may retrieve the request from the database, assign it to a HoD or faculty member for review, and track the status of the request until a decision is made.
 3. **TA Request Manager:** It is responsible for receiving and processing TA applications submitted by students. It may validate inputs, store the data and initiate the review process by assigning the request to the concerned faculty member.
 4. **TA Approval Manager:** It is responsible for managing the review and approval process for TA applications. It may retrieve the application from the database, assign it to a concerned faculty for review, and track the status of the request until a decision is made.
 5. **Reporting Manager:** This control object is responsible for generating reports on leave requests for a given academic term. It

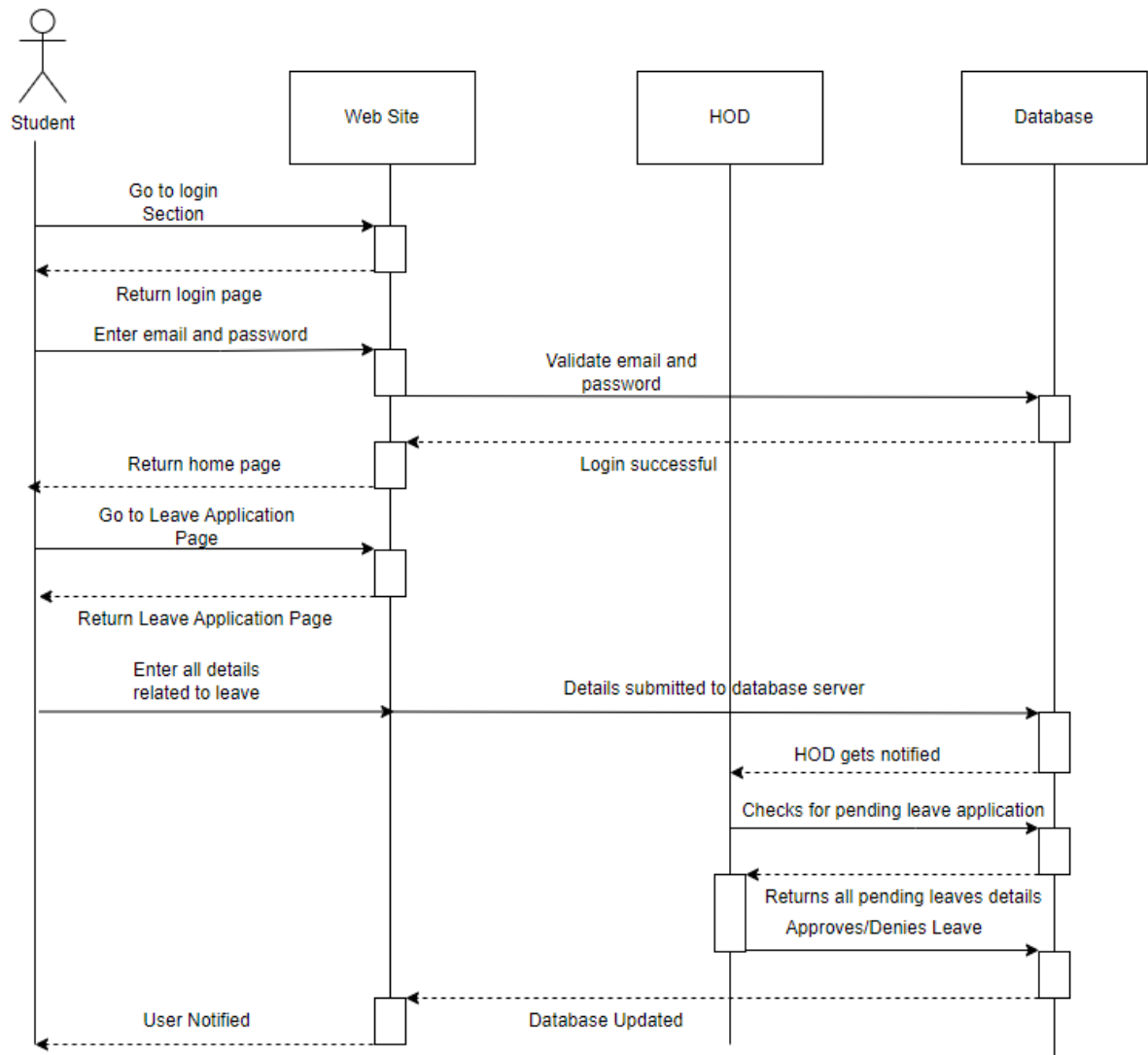
may retrieve the relevant data from the database, apply filters and aggregations as necessary, and present the results in a user-friendly format.

❖ Class Diagram:



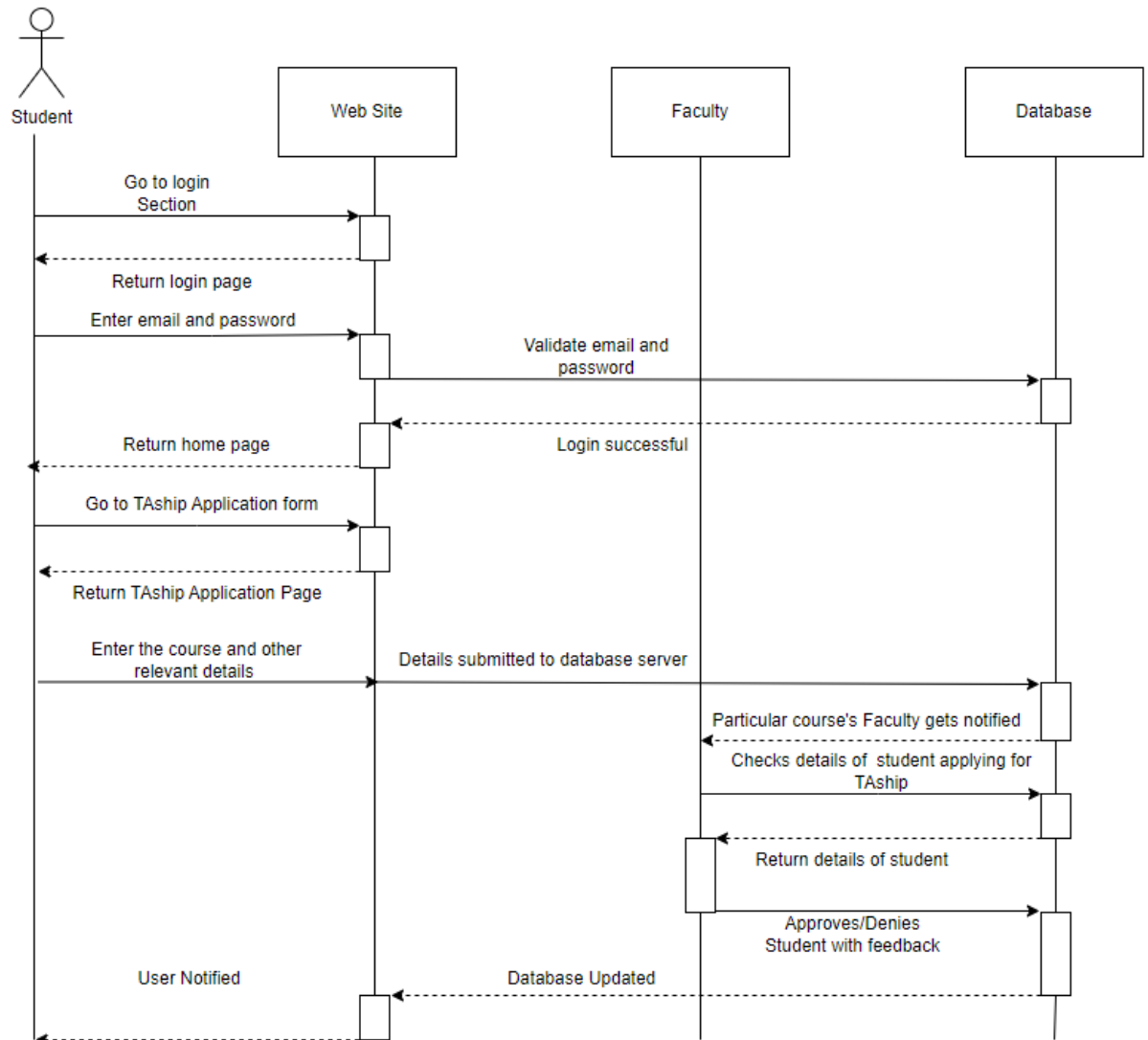
❖ Sequence Diagram:

1.Student Applying For Leave:



<https://drive.google.com/file/d/1Owl-dA064l7kAA7-FJDGWq-QpQvNtjAT/view?usp=sharing>

2. Student Applying For TAship:



https://drive.google.com/file/d/1HBoNneIUWkXnfkQ4pqZwj89yHIGj0jm_/view?usp=sharing

❖ Design Goals

MAINTENANCE AND DEPENDABILITY CRITERIA

1. **Functionality:** The software should have all the necessary features to allow stakeholders(faculty, TAs, to request leave, managers to approve or reject leave requests, and HR personnel to track and manage leave data.
2. **Availability :** The system should run constantly, with little downtime for upgrades and maintenance.
3. **Scalability :** Users should be confident that the system can adapt to changing requirements and requests.
4. **Security:**The system should secure sensitive data from unauthorized access and use suitable safeguards such as Authentication,access control and data encryption.
5. **Robustness:**Errors that occur during system execution and Incorrect input should be handled by the system.
6. **Flexibility:** It should be suitable for a wide range of user interfaces.

PERFORMANCE CRITERIA

7. Response Time:The system should acknowledge user requests as soon as possible.

8. Throughput:The system should accomplish specific amount of tasks in a certain time.

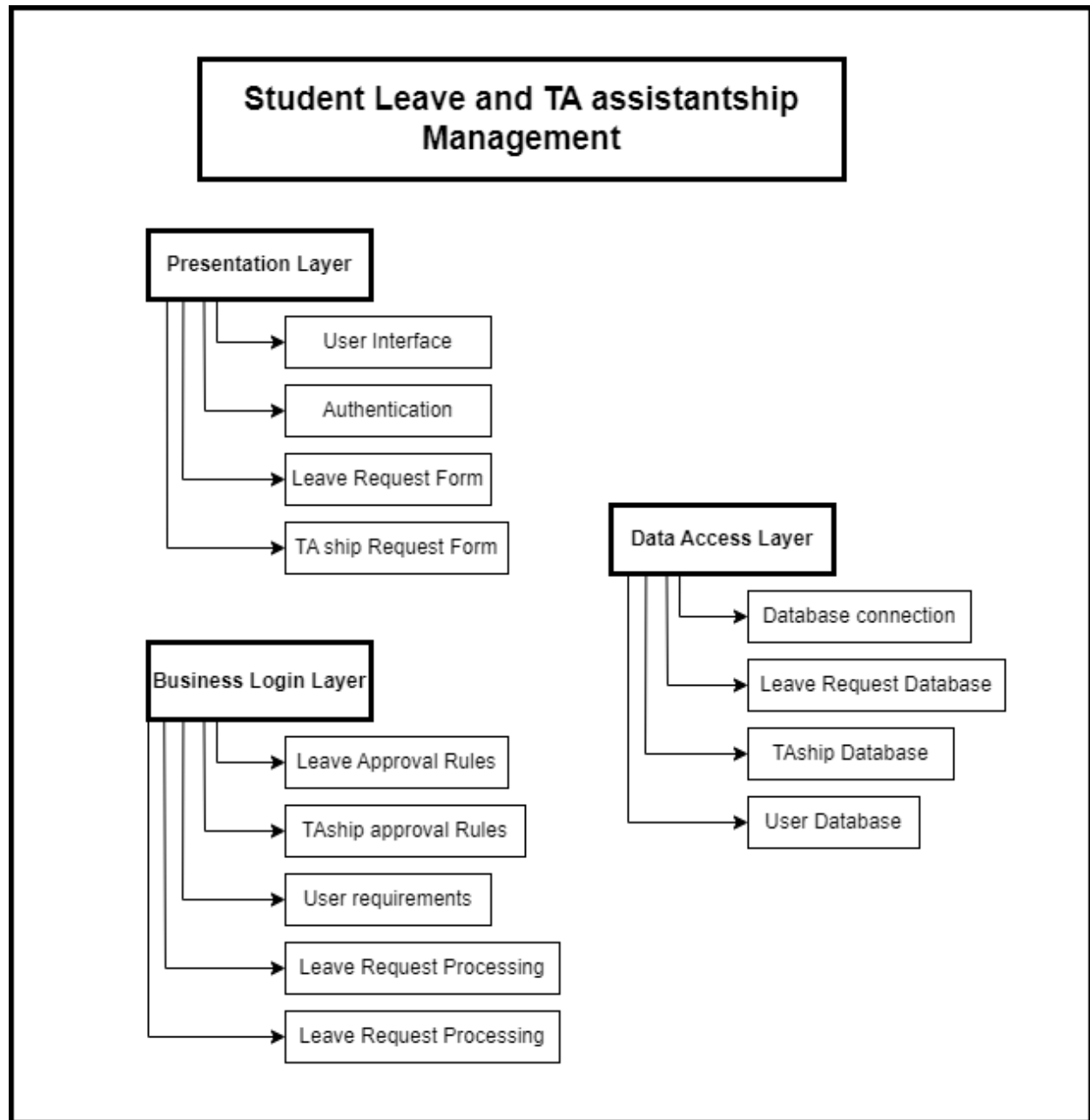
END-USER CRITERIA

9. Usability: The system ought to have a simple and intuitive user interface that can be used by anyone without requiring special training or technical knowledge, making it easy to use.

10. Utility-The system should be easily able to handle end-user's work.

High Level System Design

→ For a student leave management system, **three-tier architecture** is according to our model, with a presentation layer, a business logic layer, and a data access layer.



→ The subsystems of the three-tier architecture are:

1. Presentation Layer:

- This subsystem handles the user interface and authentication for the system.
- It includes the user interface components that allow users to interact with the system, the authentication subsystem that verifies user credentials, the TAship request from the students, and the leave request form component that allows users to submit leave requests.

2. Business Logic Layer:

- This subsystem contains the core business logic of the system.
- It includes the leave approval rules that determine whether a leave request is approved or denied, the user permissions subsystem that controls access to the system, and the leave request processing component that handles the processing of leave requests.
- It also includes the rules that determine the approval or denial of the TA ship request from the student to the faculty.

3. Data Access Layer:

- This subsystem handles the storage and retrieval of data for the system.
- It includes the database connection component that establishes a connection to the database, the leave request database component that stores information about leave requests, and the user database component that stores information about users.
- The database of the TA assistantship is also maintained which stores the subject for which TA is assigned and more.

Overall, this architecture provides a clear separation of concerns and allows for easy maintenance and scalability of the system.