ACADEMIC ISSUE TRACKING SYSTEM (AITS)

Presented by:

Mawande John Paul	24/U/0678	787 130 130
Econgu Paul	24/U/22570	789 513 778
Lyazi Patrick	24/U/06571/PS	782 726 299
Namuli Angel Rebecca	24/U/09332/PS	756 055 966
Namayanja Mary Prechard	24/U/09046/PS	773 366 103

Introduction and Background

- Managing academic records, resolving student issues, and maintaining efficient communication between students, lecturers and administrators is very critical in a university setting. However, many institutions rely on manual processes, scattered communication channels, or inefficient methods which lead to delays in issue resolution and lack of transparency.
- The Academic Issue Tracking System (AITS) aims to provide a centralized, digital platform that enables students log academic-related issues, track their progress, and receive real time updates concerning the issues raised while keeping close communication with staff. This system enhances accountability, speeds up issue resolution, and ensures seamless communication between the stakeholders.

Problem Statement

- Delayed Responses from lecturers regarding missing marks, course work complaints and issues in that line.
- Lack of transparency in the issue resolution process.
- Difficulty in communication between students, lecturers and the administration as some of the lecturers and admins are hard to get since most of them don't spend much time in their offices.
- No structured system for documenting and tracking reported issues.

❖ With this, we found it necessary to create an automated solution that creates seamless issue reporting, tracking and resolution while ensuring accountability. This will also improve the institution's image on how students' results are handled.

Project Objectives

The AITS project aims to:

- Develop a full stack web application using Django, Python(backend) and React, CSS, JavaScript (frontend)
- Implement role-based access control (students, lecturers, administrators (Academic Register))
- Enable issue logging and tracking through a user-friendly interface
- Provide automated notifications concerning the resolution of issues brought forward. This will be mainly through the interface and emails.
- Ensure System Security and authentication using JWT authentication.
- Enable cloud deployment for accessibility across multiple devices.

Project scope:

❖ The AITS will focus on the following core functionalities.

Student Portal:

- *Log an Issue:* Students can report academic issues (e.g. missing marks, incorrect course registration, timetable conflicts)
- *Track Issue progress:* View real-time updates on the resolution status of their complaints.
- *Receive Notifications:* Get updates via email or SMS when an issue status changes.
- Lecturer and Administrator Portal:
- *View Reported Issues:* An admin will be able to view the complaint and filter it accordingly.
- *Update issue status:* Change status from "open" to whether "in progress" or "resolved"
- *Assign Issues to Responsible Lecturers:* Delegate specific issues to relevant departments.

System Management

- *Role-based access control:* Ensure only authorized users to access specific functions.
- *Audit logs:* Keep track of actions performed by different users for record keeping and accountability.
- *Cloud Deployment:* Host the platform online using AWS or Heroku to allow access from multiple users.

Key Deliverables

The project will be delivered in three phases.

- **Phase 1:** Planning and initial System design
  Defining project requirements and use cases, designing database schema and the data models, developing wireframes for the UI, setting up a version control system using Git/GitHub.
- **Phase 2:** Prototyping
  Presenting a working prototype showcasing user authentication and role-based access, issue logging and storage functionality and basic dashboard interfaces for students and academic register

- **Phase 3:** Deployment and Finalization
  A final version of the application has to be delivered having feedback from the prototype earlier presented.

Functional Requirements

Student Portal

- Students should be able to log academic issues for example missing marks, incorrect course registration, timetable conflicts.
- Students should be able to track the progress of their reported issues in real-time.
- Students should receive notifications (email/in portal) when their issue status changes.

Lecturer and Administrator Portal

- Lecturers and administrators should be able to view reported issues and filter them accordingly.
- Administrators should be able to assign issues to the relevant lecturers or departments.
- Lecturers should be able to update the issue status.
- Lecturers should provide constant accountability.

System Management

- The system must implement role-based access control (RBAC) to restrict functionalities based on user roles which include the students, lecturers, administrators.
- The system must maintain audit logs to track user actions and ensure accountability.
- The system should support secure authentication using JSON Web Token.
- The system should support cloud deployment for accessibility from multiple devices.

Non-Functional Requirements

Performance & Usability

- The system should have a user-friendly interface for easy navigation.
- The system should provide real-time updates on issue statuses.
- The system should process issue tracking within seconds to avoid delays.

Security

- The system should use JWT authentication to ensure only authorized users access it.
- Sensitive data like student records, academic issues etc should be encrypted in transit and at rest.
- The system should implement role-based access control to prevent unauthorized access.

Scalability & Reliability

- The system should be scalable to accommodate a large number of students and lecturers.
- It should handle simultaneous requests without crashing.
- The system should have regular backups to prevent data loss.

Maintainability & Portability

- The system should be easily maintainable, with clear documentation.
- The system should be deployable on the cloud like AWS, Heroku, etc. for multi-device access.

USER STORIES

User Story 1: Report an Academic Issue

As a student, I want to be able to report an academic issue (e.g., grade dispute, course scheduling conflict) so that I can get help resolving the issue.

Acceptance Criteria:

- The system allows students to submit a report with details of the issue.
- The report is sent to the relevant academic administrator (e.g., department chair, academic advisor).

User Story 2: Track and View Academic Issues

As an academic administrator, I want to be able to track and view all reported academic issues so that I can prioritize and resolve them efficiently.

Acceptance Criteria:

- The system provides a dashboard for administrators to view all reported issues.
- Administrators can filter and sort issues by category, status, and date.

User Story 3: Assign and Reassign Academic Issues

As an academic administrator, I want to be able to assign and reassign reported academic issues to other administrators or staff members so that issues can be resolved efficiently.

Acceptance Criteria:

- The system allows administrators to assign issues to other administrators or staff members.
- The system sends notifications to the assigned administrator or staff member.

User Story 4: Update and Resolve Academic Issues

As an academic administrator, I want to be able to update the status of reported academic issues and mark them as resolved so that students and other administrators can track progress.

Acceptance Criteria:

- The system allows administrators to update the status of issues (e.g., "in progress," "resolved").
- The system sends notifications to students and other administrators when an issue is resolved.

User Story 5: Generate Reports and Analytics

As an academic administrator, I want to be able to generate reports and analytics on reported academic issues so that I can identify trends and areas for improvement.

Acceptance Criteria:

- The system provides reporting and analytics tools for administrators.
- Administrators can generate reports on issue categories, resolution rates, and other metrics.

User Story 6: Integrate with Existing Systems

As an academic administrator, I want the Academic Issues Tracking System to integrate with our existing student information system (SIS) and learning management system (LMS) so that we can streamline processes and reduce data entry.

Acceptance Criteria:

- The system integrates with the SIS and LMS through APIs or other integration methods.
- The system can import and export data from the SIS and LMS.

User Story 7: Provide User Notifications and Alerts

As a student or academic administrator, I want to receive notifications and alerts when there are updates on reported academic issues so that I can stay informed and take action.

Acceptance Criteria:

- The system sends notifications and alerts to students and administrators via email or in-app notifications.
- Users can customize their notification preferences.

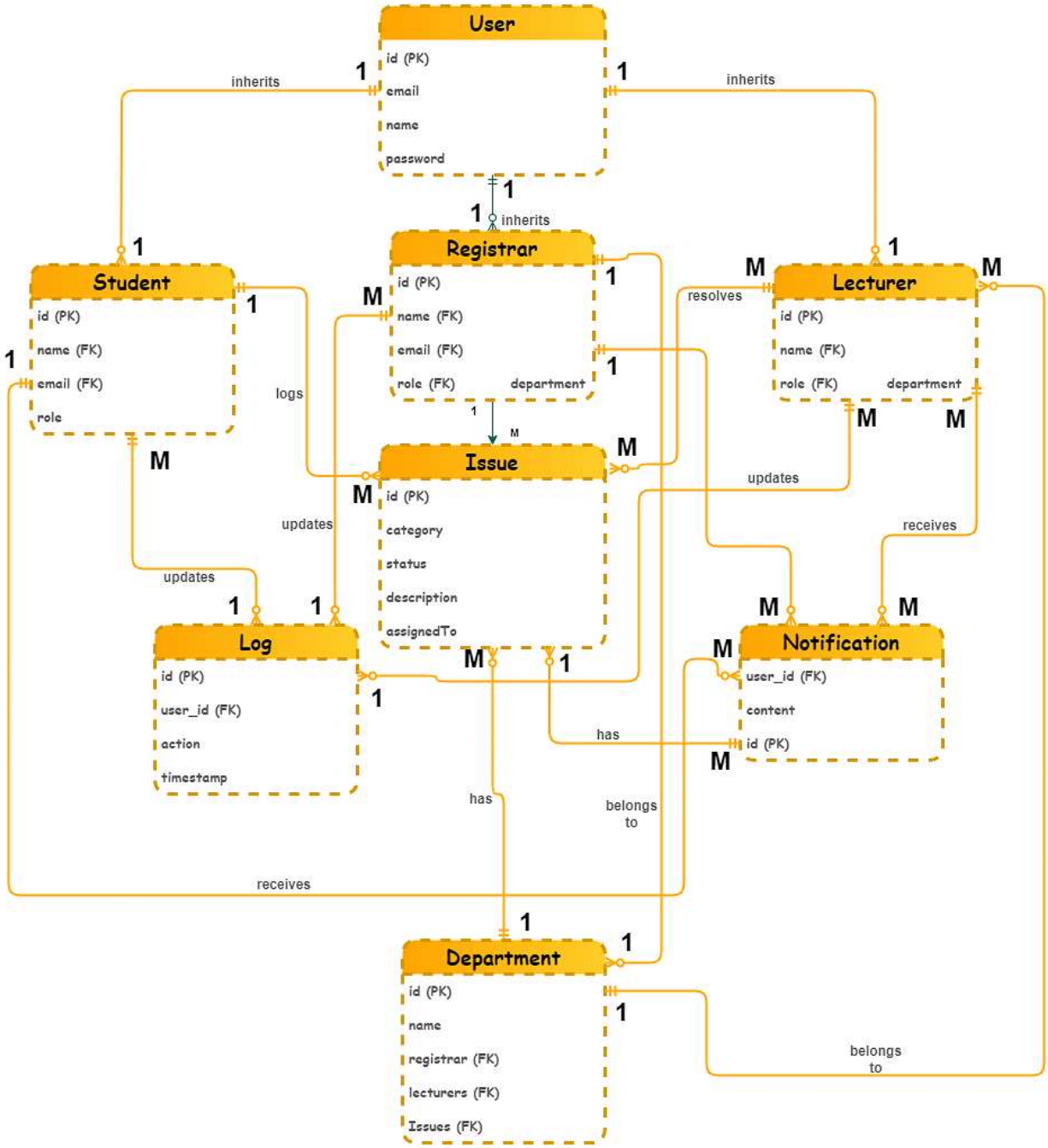
User Story 7: Provide User Notifications and Alerts

As a student or academic administrator, I want to receive notifications and alerts when there are updates on reported academic issues so that I can stay informed and take action.

Acceptance Criteria:

- The system sends notifications and alerts to students and administrators via email or in-app notifications.
- Users can customize their notification preferences.

ENTITY RELATIONSHIP DIAGRAM


User
id (PK)
email
name
password
inherits
inherits
inherits
Registrar
id (PK)
name (FK)
email (FK)
role (FK)
department
Student
id (PK)
name (FK)
email (FK)
role
Lecturer
id (PK)
name (FK)
role (FK)
department
resolves
logs
Issue
id (PK)
category
status
description
assignedTo
updates
updates
updates
receives
Log
id (PK)
user_id (FK)
action
timestamp
Notification
user_id (FK)
content
id (PK)
has
has
belongs to
receives
Department
id (PK)
name
registrar (FK)
lecturers (FK)
Issues (FK)
belongs to

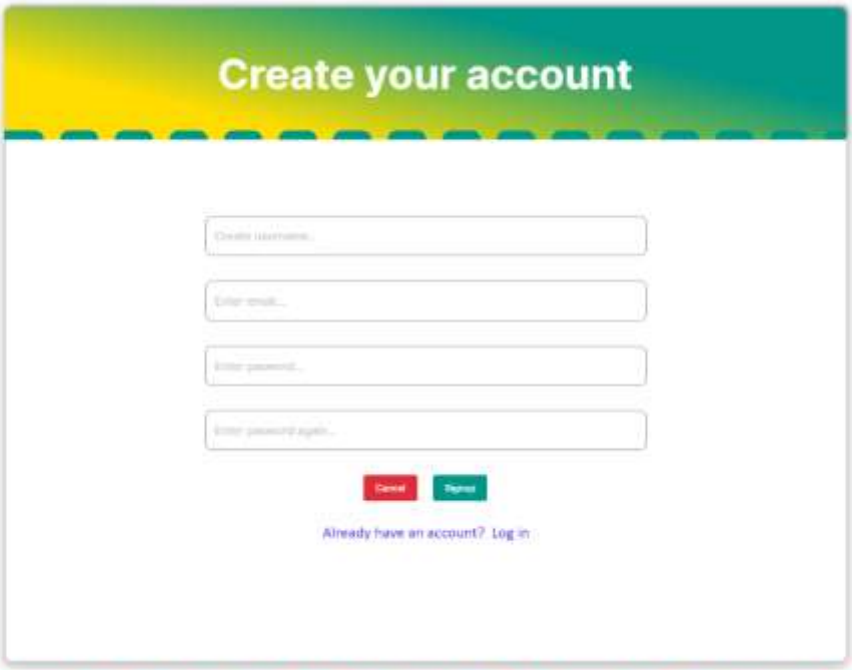
WIREFRAMES

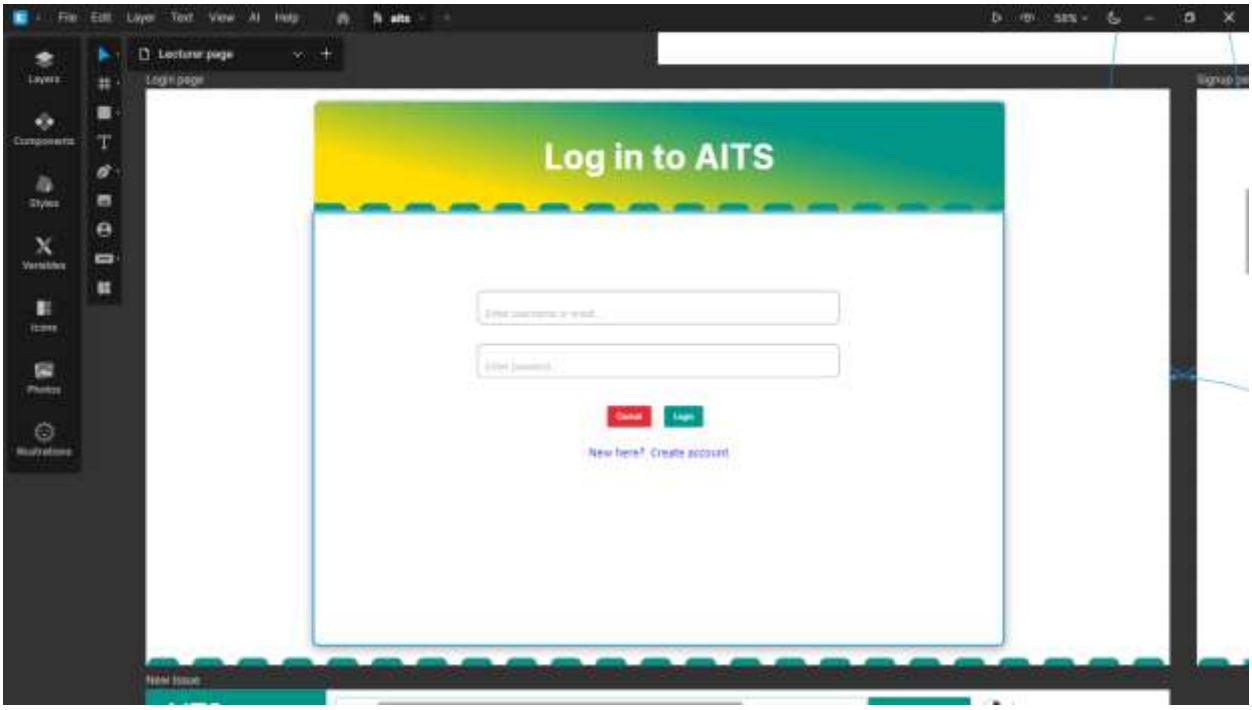
Disclaimer: Slight inconsistencies in UI components due to export errors in the wireframing program of choice.

For complete wireframe, refer to the document ;

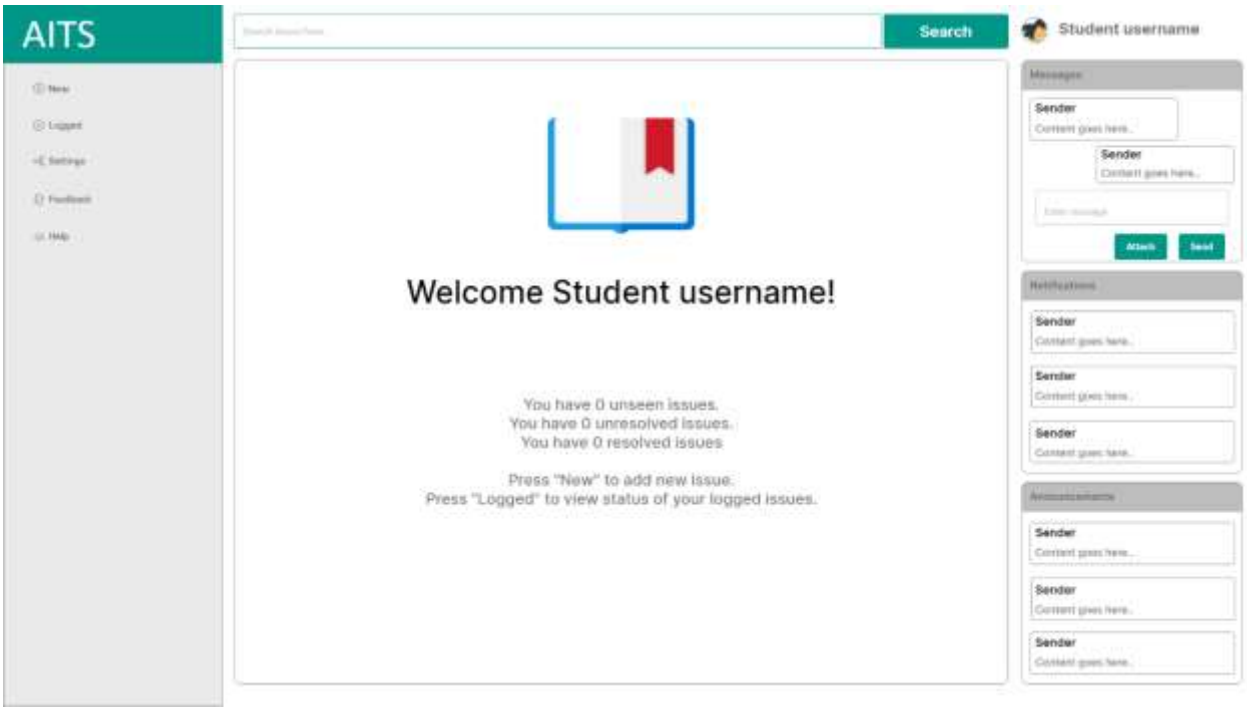
aits.free

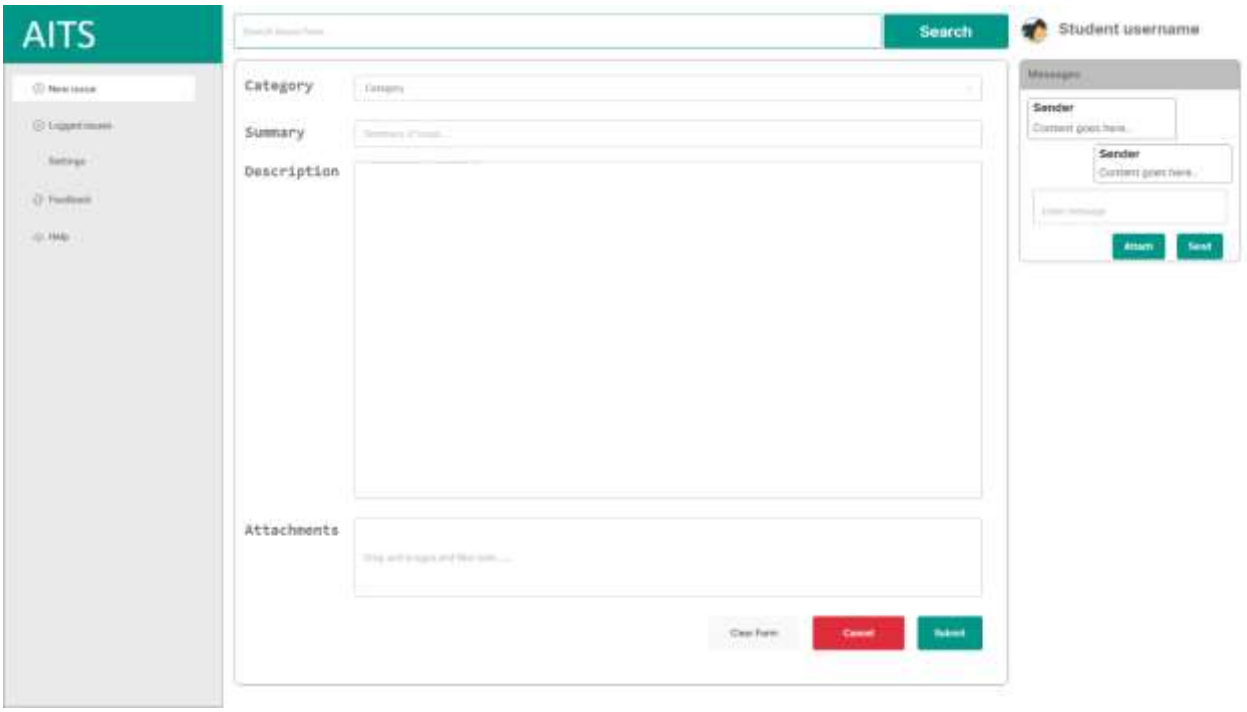
Can be opened using Lunacy, available in MS Store.

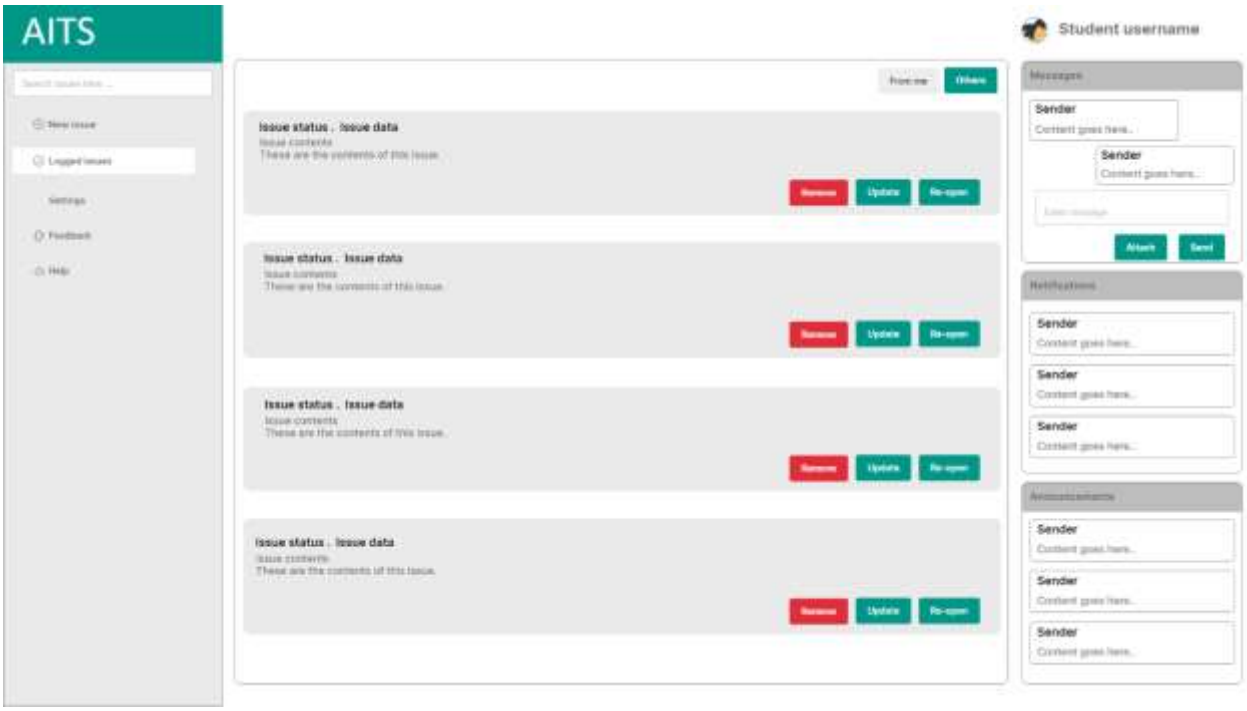

Create your account
Already have an account? Log in


Log in to AITS
New here? Create account

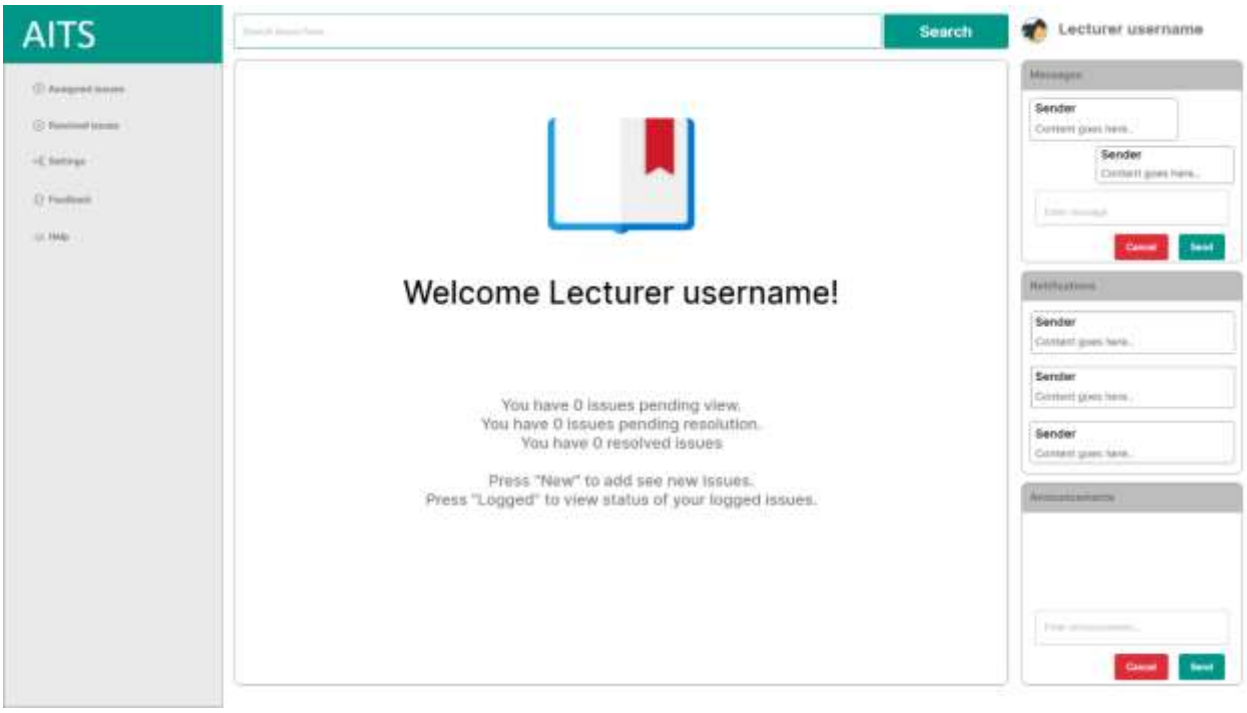
STUDENT

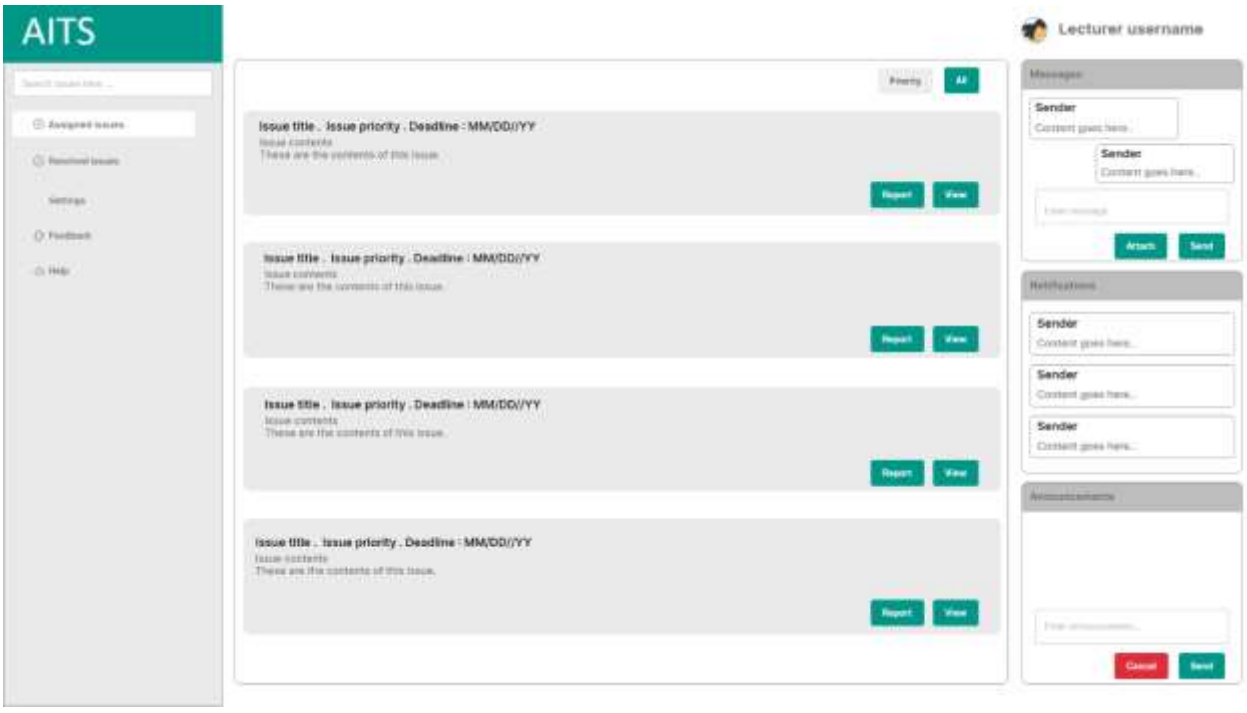

AITS
Search
Student username
Welcome Student username!
You have 0 unseen issues.
You have 0 unresolved issues.
You have 0 resolved issues
Press "New" to add new issue.
Press "Logged" to view status of your logged issues.
Sender

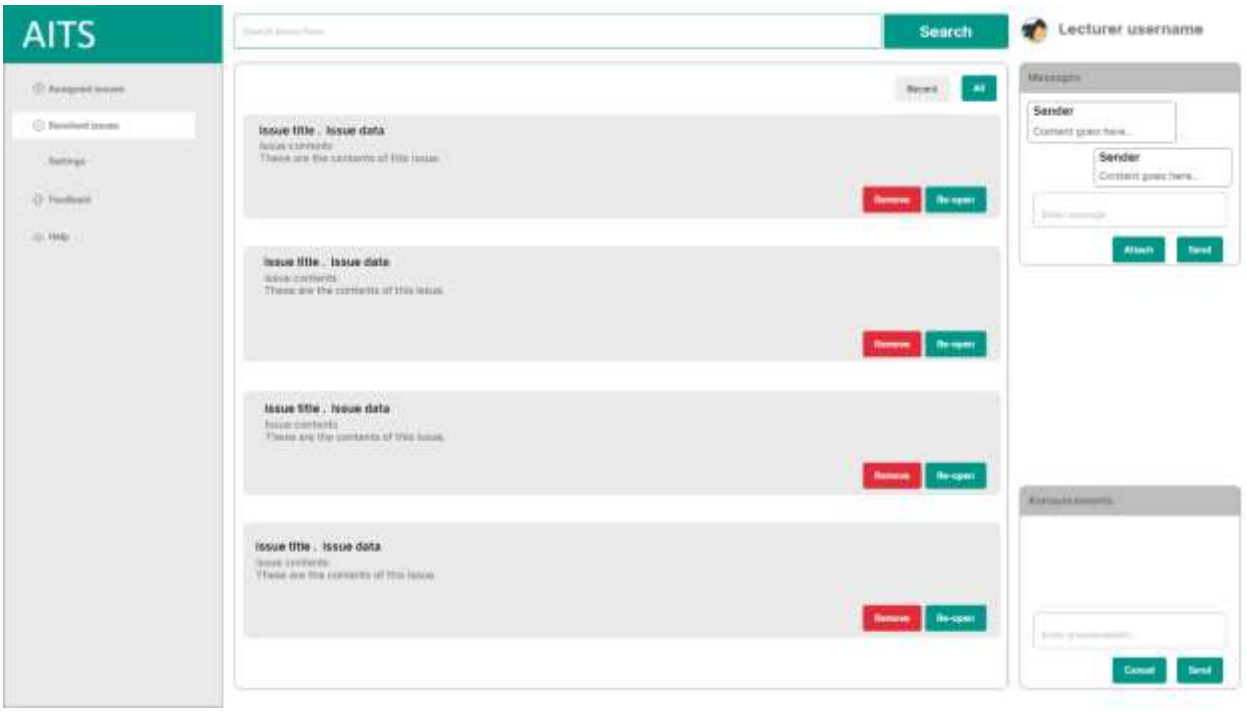

AITS
Search
Student username
Category
Summary
Description
Attachments
Sender

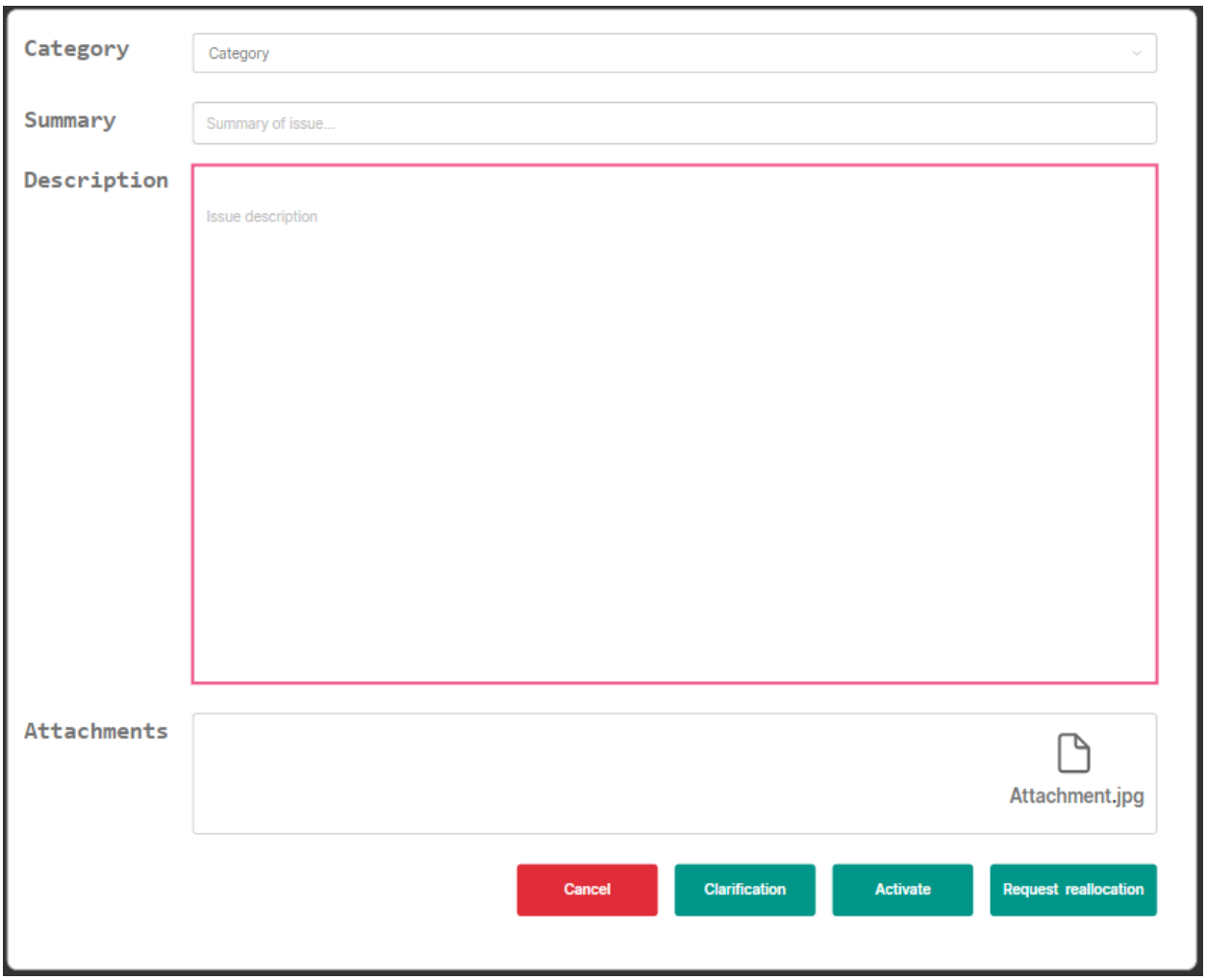

AITS
Student username
Issue status . Issue data
Issue contents
These are the contents of this issue.
Sender
Content goes here..

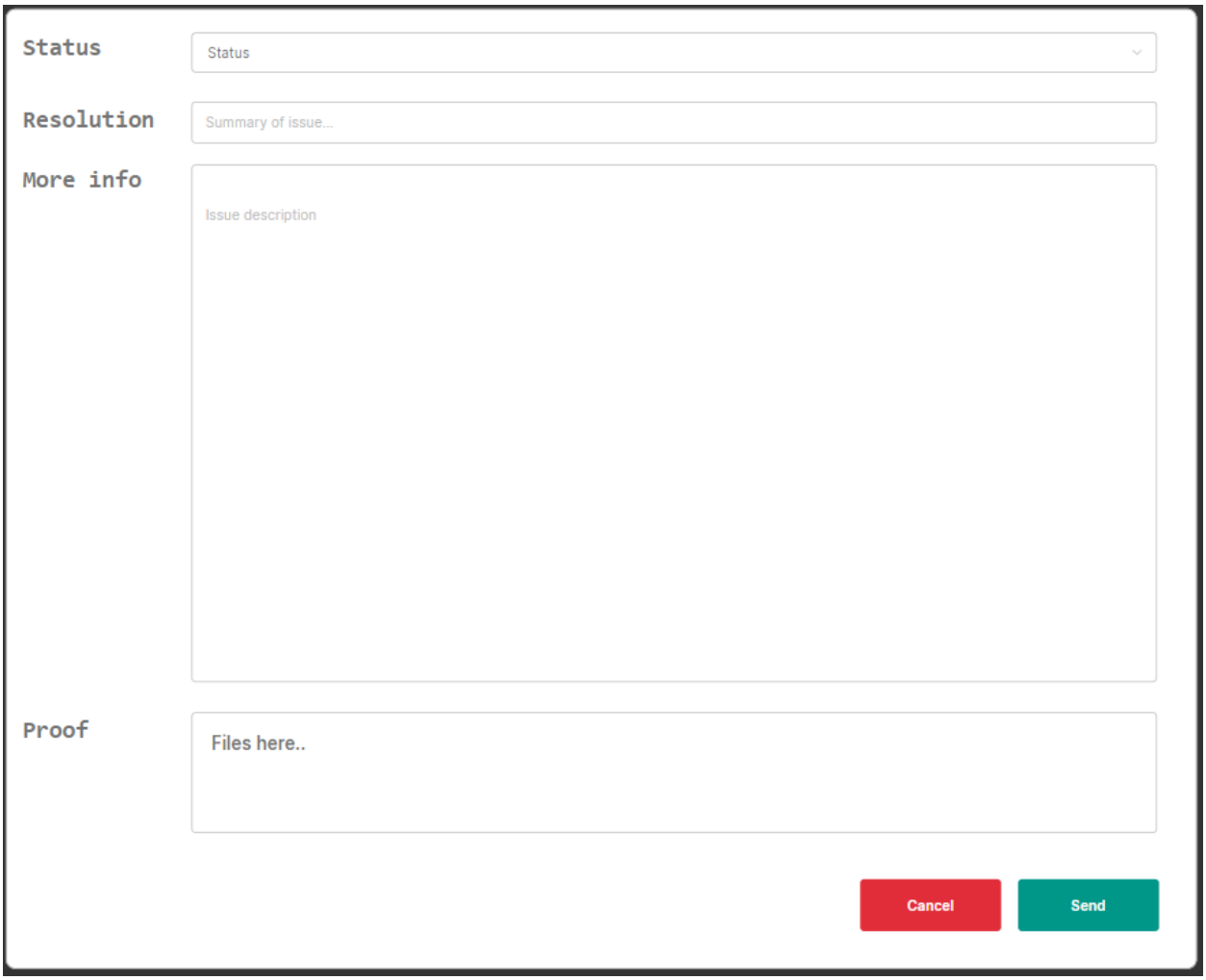
LECTURER


AITS
Search
Lecturer username
Welcome Lecturer username!
You have 0 issues pending view.
You have 0 issues pending resolution.
You have 0 resolved issues
Press "New" to add see new issues.
Press "Logged" to view status of your logged issues.
Sender
Content goes here..

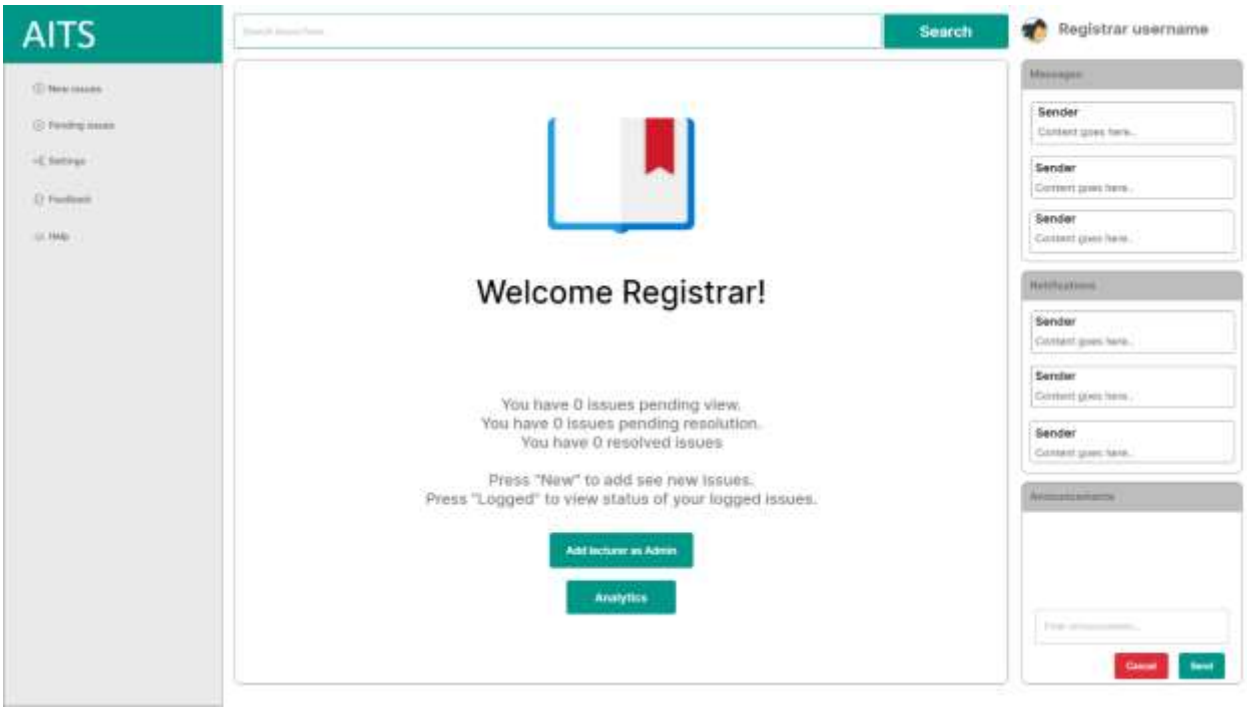

AITS
Lecturer username
Issue title . Issue priority . Deadline : MM/DD//YY
Issue contents
These are the contents of this issue.
Report
View
Issue title . Issue priority . Deadline : MM/DD//YY
Issue contents
These are the contents of this issue.
Report
View
Issue title . Issue priority . Deadline : MM/DD//YY
Issue contents
These are the contents of this issue.
Report
View
Issue title . Issue priority . Deadline : MM/DD//YY
Issue contents
These are the contents of this issue.
Report
View
Sender
Content goes here...
Sender
Content goes here...
Attach
Send
Sender
Content goes here...
Sender
Content goes here...
Sender
Content goes here...
Cancel
Send

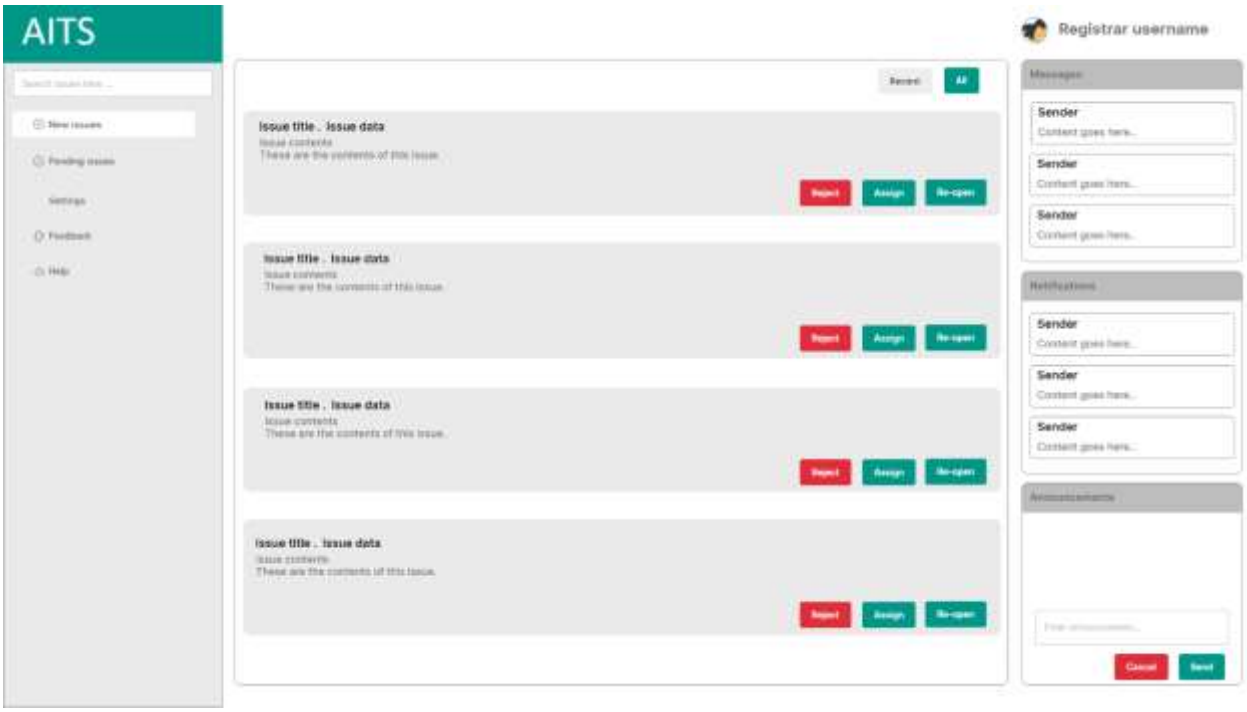

AITS
Search
Lecturer username
Issue title . Issue data
Issue contents
These are the contents of this issue.
Remove
Re-open
Issue title . Issue data
Issue contents
These are the contents of this issue.
Remove
Re-open
Issue title . Issue data
Issue contents
These are the contents of this issue.
Remove
Re-open
Issue title . Issue data
Issue contents
These are the contents of this issue.
Remove
Re-open
Sender
Content goes here...
Sender
Content goes here...
Attach
Send
Cancel
Send

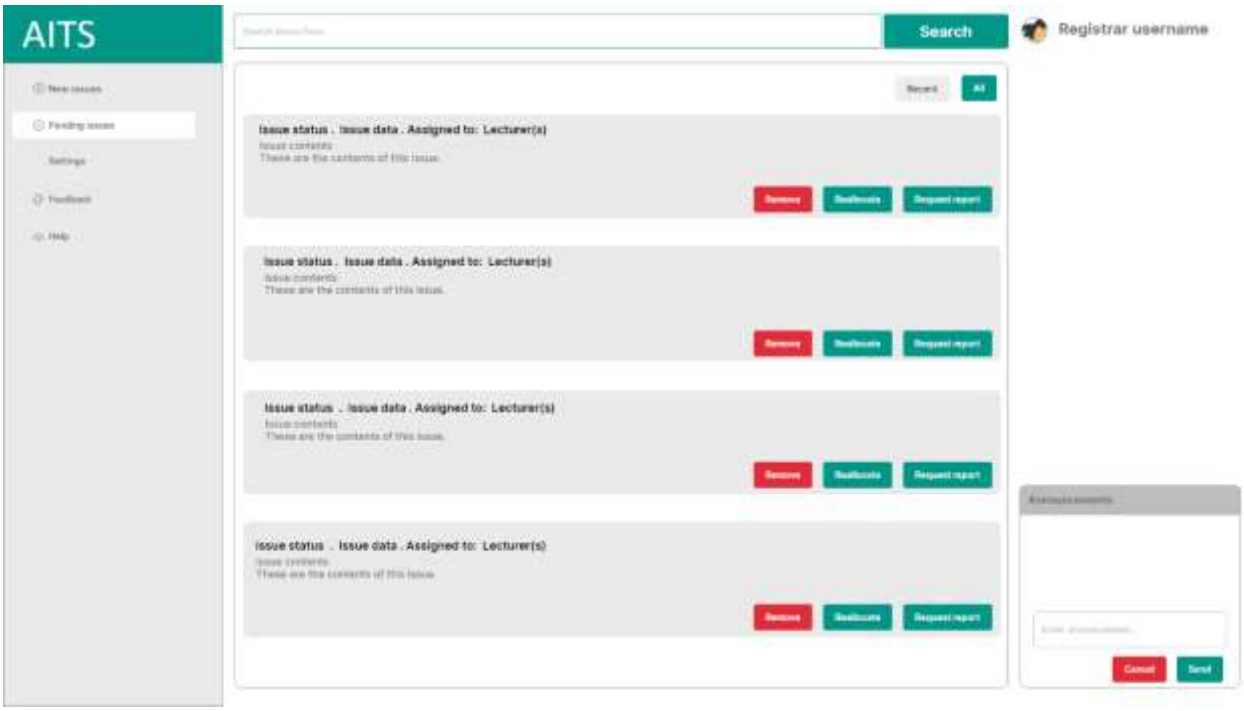

Category
Category
Summary
Summary of issue...
Description
Issue description
Attachments
Attachment.jpg
Cancel
Clarification
Activate
Request reallocation

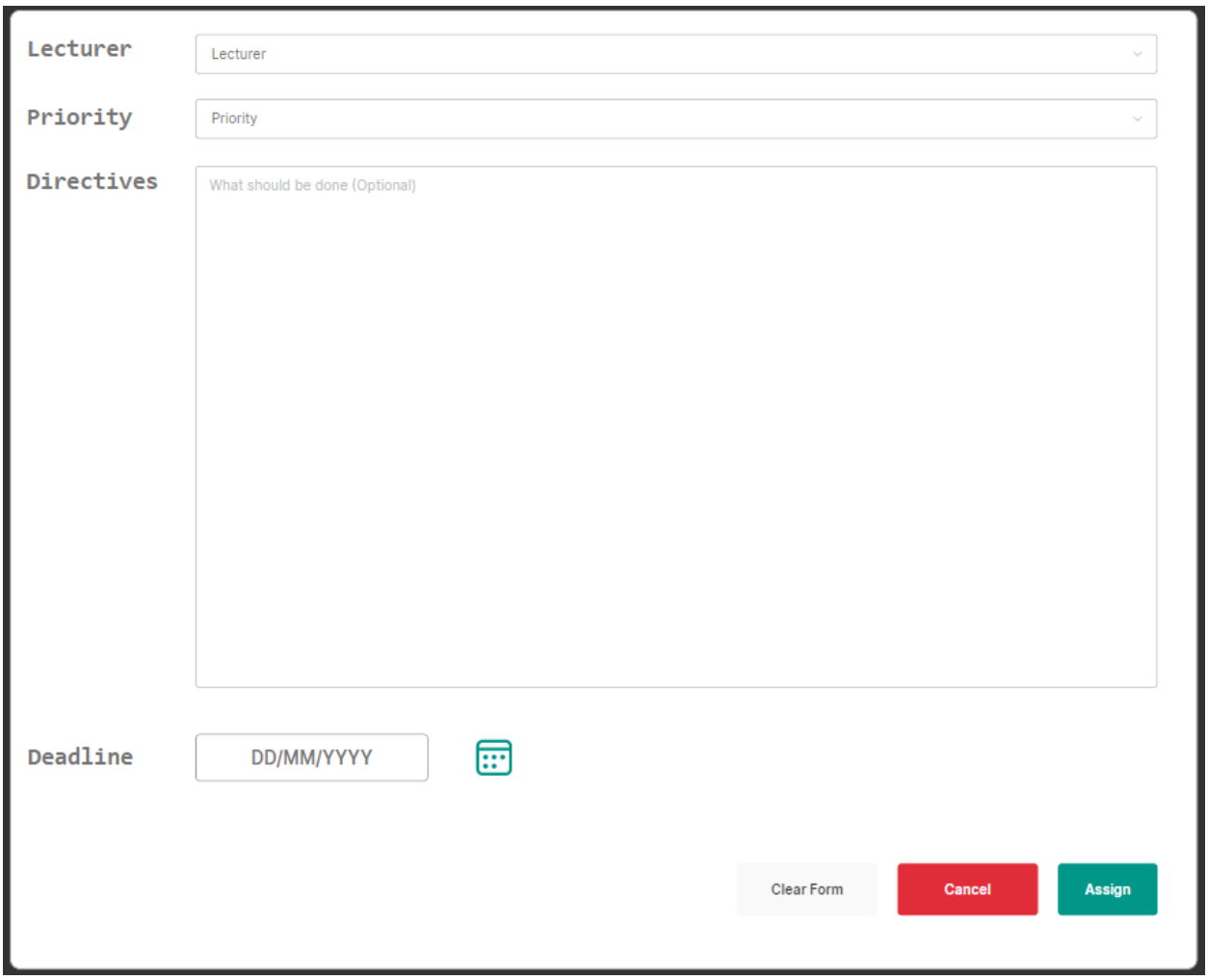

Status
Status
Resolution
Summary of issue...
More info
Issue description
Proof
Files here..
Cancel
Send

REGISTRAR


AITS
Search
Registrar username
Welcome Registrar!
You have 0 issues pending view.
You have 0 issues pending resolution.
You have 0 resolved issues
Press "New" to add see new issues.
Press "Logged" to view status of your logged issues.
Add lecturer as Admin
Analytics
Sender
Sender
Sender
Sender
Sender
Sender


AITS
Registrar username
Issue title . Issue data
Issue title . Issue data
Issue title . Issue data
Issue title . Issue data
Sender
Sender
Sender
Sender
Sender
Sender


AITS
Search
Registrar username
Recent
All
Issue status . Issue data . Assigned to: Lecturer(s)
Issue contents
These are the contents of this issue.
Remove
Reallocate
Request report
Issue status . Issue data . Assigned to: Lecturer(s)
Issue contents
These are the contents of this issue.
Remove
Reallocate
Request report
Issue status . Issue data . Assigned to: Lecturer(s)
Issue contents
These are the contents of this issue.
Remove
Reallocate
Request report
Issue status . Issue data . Assigned to: Lecturer(s)
Issue contents
These are the contents of this issue.
Remove
Reallocate
Request report
Cancel
Send


Lecturer
Lecturer
Priority
Priority
Directives
What should be done (Optional)
Deadline
DD/MM/YYYY
Clear Form
Cancel
Assign