

**Enterprise Web Software Development**

Group 5: Trần Ngọc Nhất Linh (GCS18518), Ngô Triều Hảo(GCS18683), Trần Trung Tính(GCS18834), Mai Xuân Duy(GCS18380), Lê Thanh Đức (GCS18688), Phạm Mạnh Hải (GCS18586)

Class: GCS0704A

Subject code: 1640

Assessor name: **HO NGUYEN PHU BAO**

Assignment due: Assignment submitted

Table of content

[1. **Introduction** 1](#_Toc69418094)

[2. **Database** 1](#_Toc69418095)

[2.1. Entity relationship diagrams 1](#_Toc69418098)

[2.2. Relational schema 8](#_Toc69418099)

[3. **Functionality** 8](#_Toc69418100)

[3.1. Product backlogs 8](#_Toc69418104)

[3.2. Use case diagrams 8](#_Toc69418105)

[3.3. Use case Specification 12](#_Toc69418111)

[4. **Design** 19](#_Toc69418112)

[4.1. Website wireframes 19](#_Toc69418113)

[5. **Testing** 21](#_Toc69418114)

[5.1. Test Log 21](#_Toc69418116)

[5.2. Test Plan 25](#_Toc69418117)

[6. **Agile method** 28](#_Toc69418118)

[7. **Product** 31](#_Toc69418119)

[8. **Links** 43](#_Toc69418120)

[9. **References** 44](#_Toc69418121)

# **Introduction**

This project is a group coursework. The project is built with ASP.NET Core which allow us retain flexibility while constructing solutions. The project is a web-based secure role-based system for collecting student contributions for an annual university magazine in a large university. This project implements the processes according to criteria using scrum agile work.

In this project, group 5 will create a website where students in the school can talk to each other called… There will be 3 main users of this website: Admin, Staff, and Student. With this website, all students in the school can post and comment on their opinions below that post. Staff will manage the posts posted to the website. Admin will decentralize for each user object. Not only that, users who are not under the management of the school can also visit

Below is the requirement specification as obtained. A system that accomplishes the following:

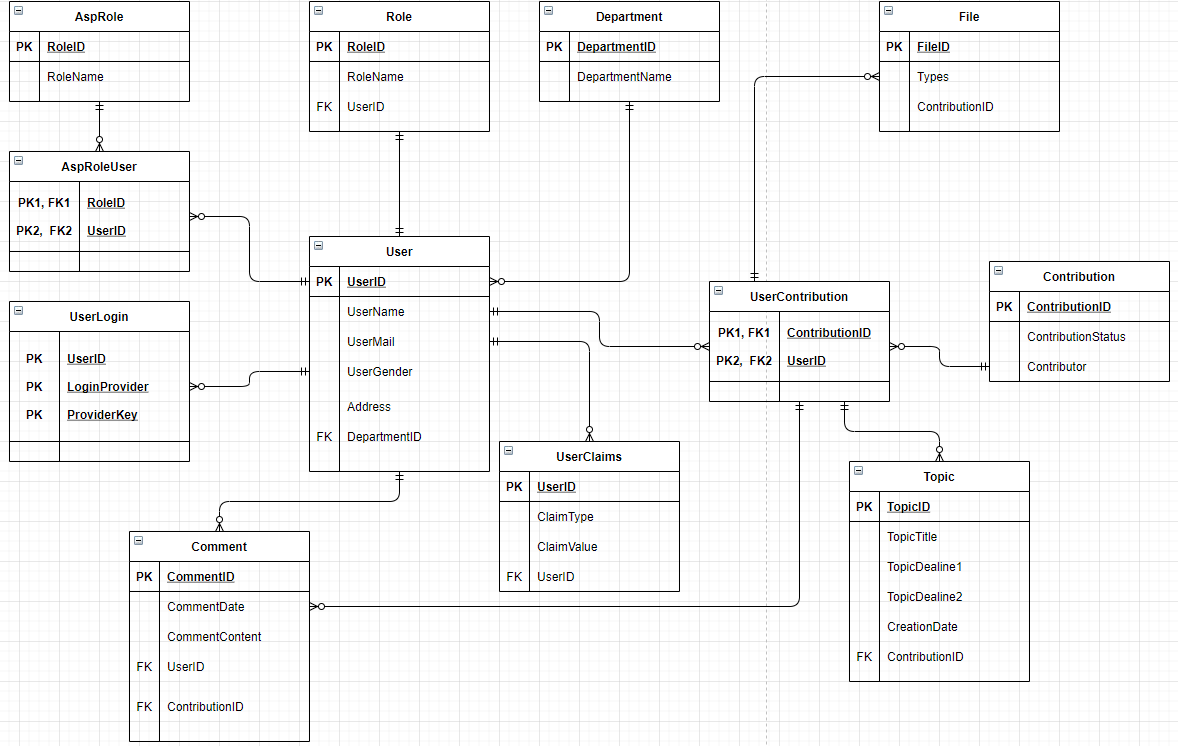
* Student can submit contribution on the system
* The coordinator can moderate the student's post before deciding whether to post it on the system or not
* Student and coordinator can comment in post.
* Admin can edit all data on the system as well as decentralize use for each object
* At every step an email notification must be sent to the relevant so action can be taken where required

# **Database**



## Entity relationship diagrams

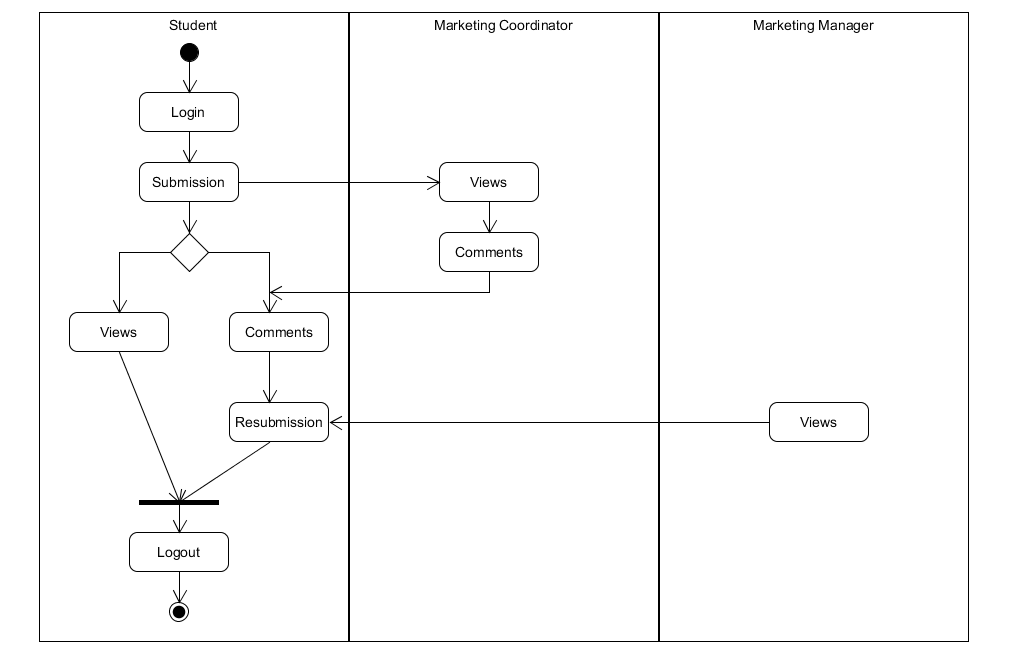
* ERD Diagram:



(Figure 1: ERD Diagram)

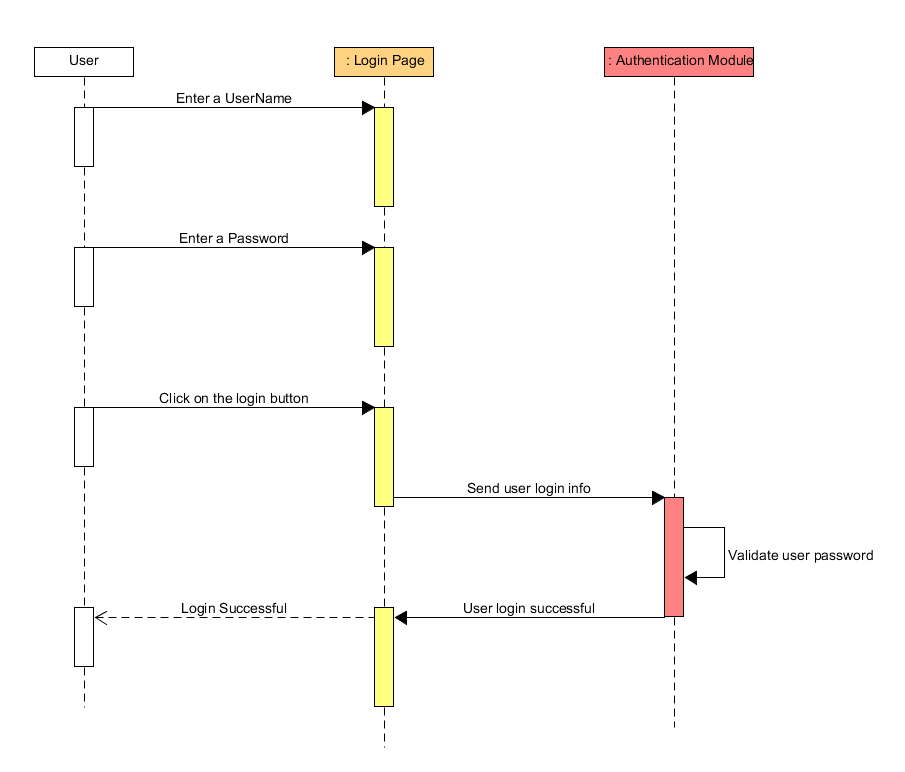
Above is the ERD Diagram table. This table shows the linkages between the functions in the system. Each table will have a primary key, and these primary keys will be subkeys in other tables and link the tables. (Figure 1)

* Activity Diagram:



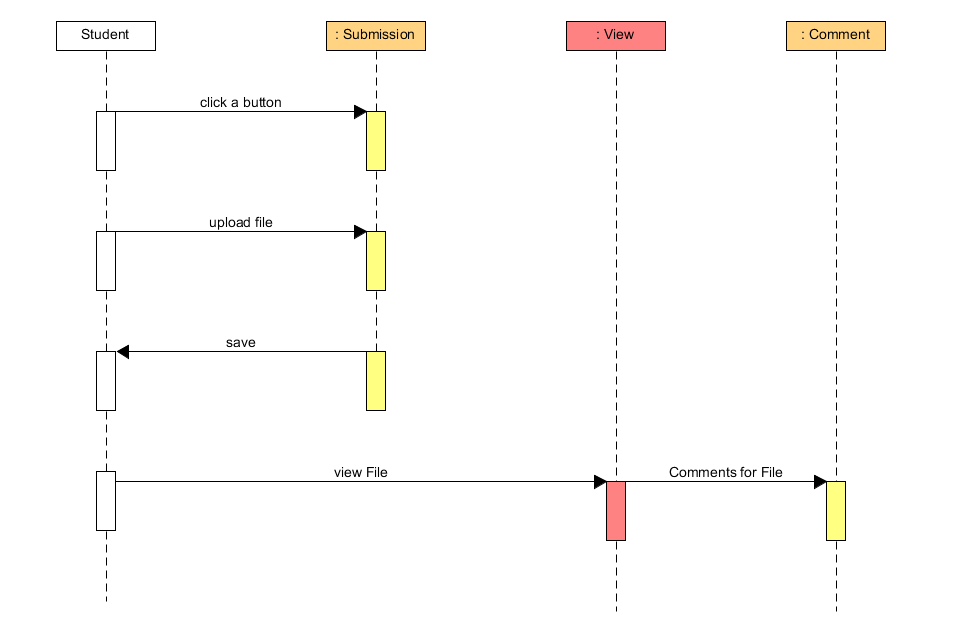
(Figure 2: Activity Diagram)

In the Activity Diagram, students will first log into the system with the account provided by the Admin. After successfully logging in, students can post to the system. From there, the Coordinator will view that post and then send the comment back to the students. Students will have 2 options: view the post then exit or view comments, then edit the post and then submit it again. Now the Manager will see and then decide whether that post is posted on the forum or not. (Figure 2)



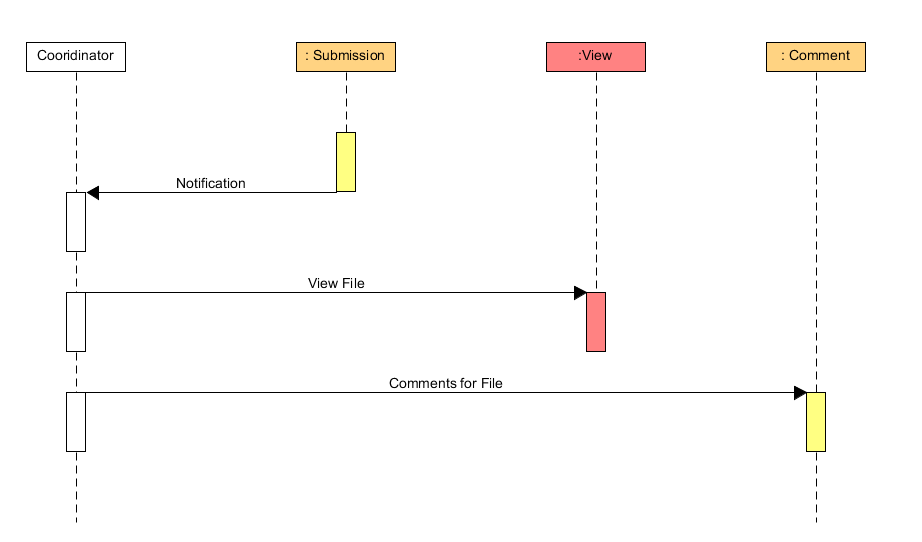
(Figure 3: Sequence Diagram User)

The above diagram shows in turn the steps that the user needs to do when starting to log in to the system such as: Enter User name, Enter Password and the system will give the results. (Figure 3)



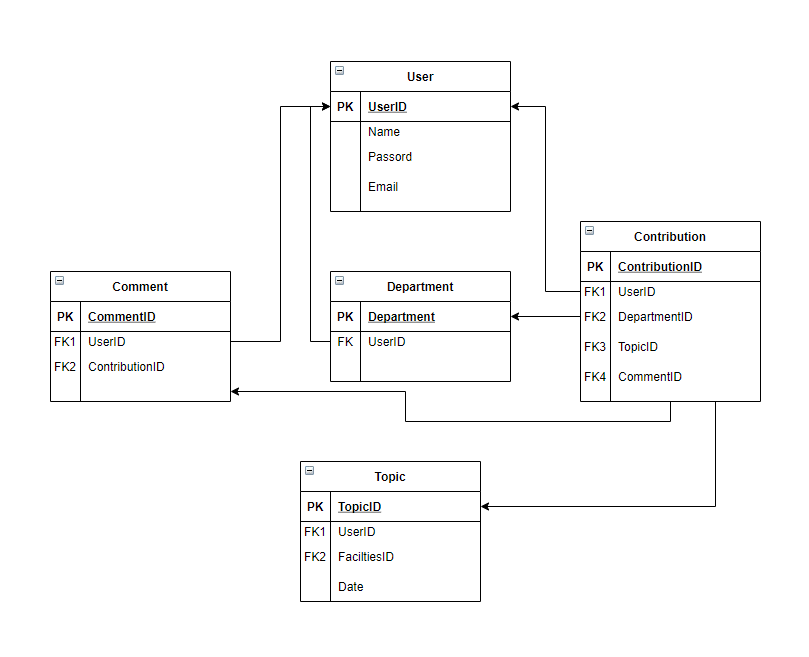
(Figure 4: Sequence Diagram Student)

Diagram shows the submission process of a student in the system. Students will choose "Submission" then upload the file and save the file. After saving the file, students can review and comment on it (Figure 4)



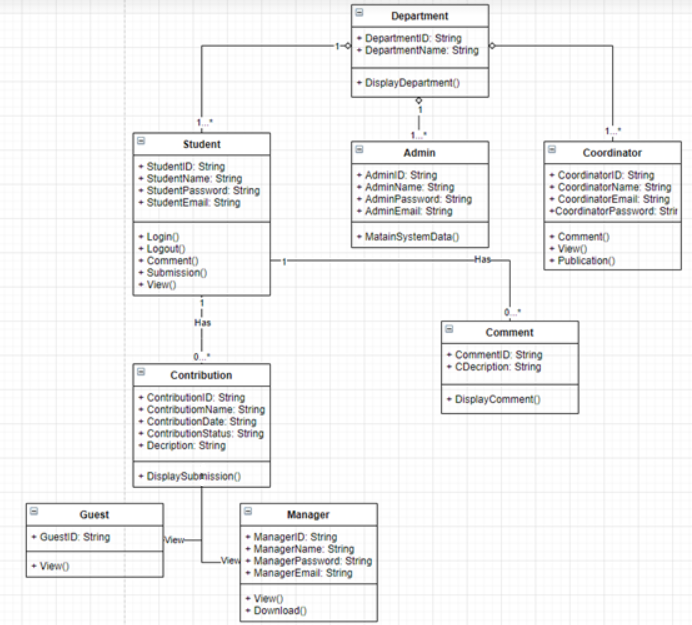
(Figure 5: Sequence Diagram Coordinator)

The table above shows the Coordinator's functions such as: View files uploaded by students and comment their opinions in the article. (Figure 5)



(Figure 6: Relational Schema Diagram)

Above is the Relational Diagram table. This table shows the linkages between the functions in the system. Each table will have a primary key, and these primary keys will be subkeys in other tables and link the tables. (Figure 6)



(Figure 7: Class Diagram)

Class Diagram shows the structure and relationship between the components that make up the software. Class diagram is mainly composed of classes, each Class describes a group of objects with the same properties and actions in the system. These classes are often associated with 4 relationships: Association, Aggregation, Composition, and Generalization. (Figure 7)

## Relational schema

* User( UserId, Firstname, Lastname)
* Faculty( FacultyId, FacultyName, Year)
* Submission
* Comment
* File
* Picture
* FacultiesUser( FacultyId, UserId)
* SubmissionUser

# **Functionality**

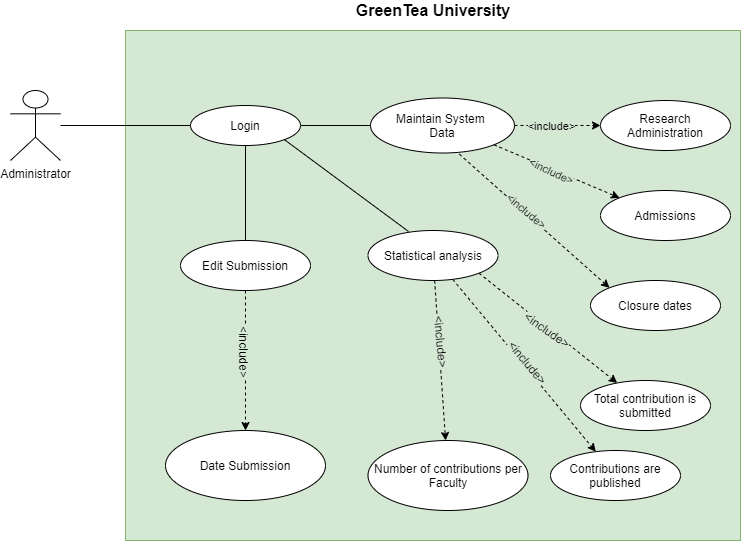


## Product backlogs

## Use case diagrams

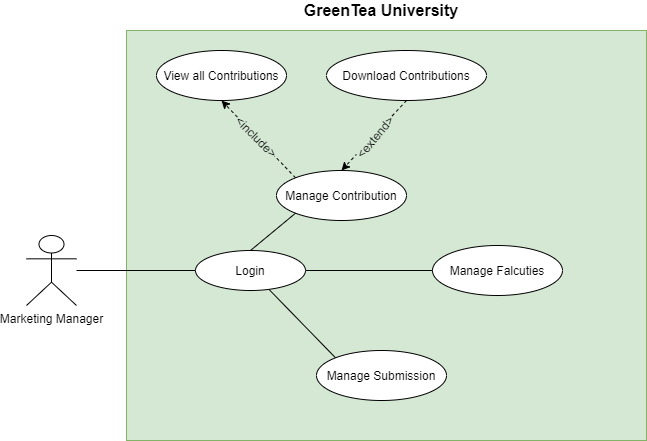
We choose use case diagrams for our design. The diagrams will describes the design of our project. Use case diagrams are made up of users, use cases, and their relationships.

### Admin



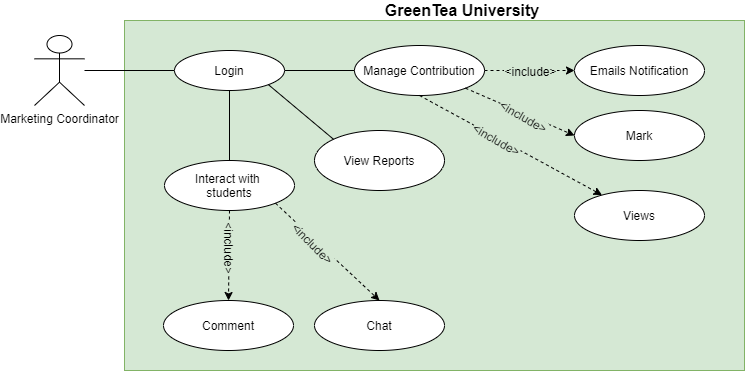
Admin has full rights to use the system such as logging into the system, correcting information on the system, and delegating authority to each type of system user

### Manager



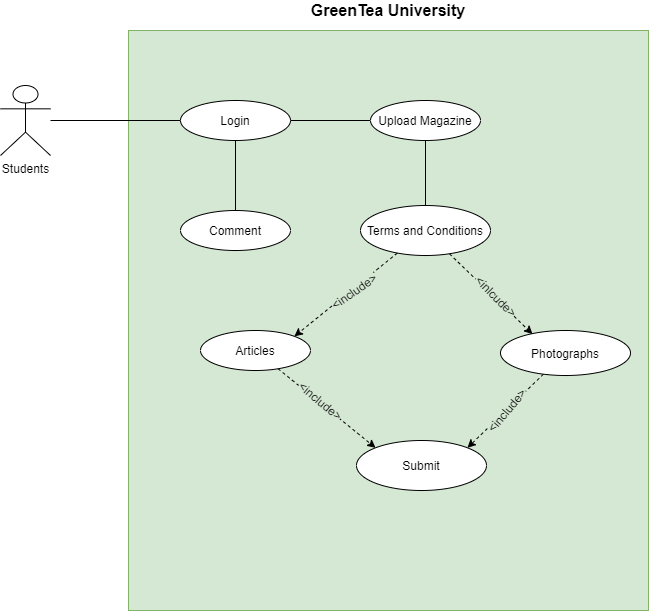
Manager can login to system and Manage Faculties, Manage Submission, Manage Contribution

### Coordinator



Coordinator will log into the system from which to manage Contribution including Emails Notification, Mark, Views. Coordinator can view reports, Interact with students. In Interact with student coordinator can comment and chat with student

### Student



Student can login to system and upload Magazine have a articles and photographs or comment to post.

|  |  |
| --- | --- |
| **Use case ID:** | LG-1.1 |
| **Use case name:** | Login |
| **Brief description:** | As a user, administrator, marketing coordinator, marketing manager, I want to log into the application to use the service from the application. |
| **Actor:** | Marketing coordinator, marketing manager, students, administrator |
| **Priority:** | High |
| **Trigger:** | User wants to log into the Green Tea University application |
| **Pre-condition(s):** | * User account has been created * The user account has been authorized * The user's device was connected to the internet when logging in |
| **Post-condition(s):** | * The user has successfully logged in to the application * The system records successful login activity |
| **Basic Flow:** | 1. Users access the Green Tea University application 2. The user chooses the method to log in with the account 3. The user enters the account and selects the login command 4. The system verifies login information successfully and allows users to access the application 5. The system records successful login activity |



## Use case Specification

### Login

### Create Account

|  |  |
| --- | --- |
| **Use case ID:** | CA-1.1 |
| **Use case name:** | Create Account |
| **Brief description:** | This function is used to create an account. For example, the admin will create an account for the students and marketing manager. |
| **Actor:** | Admin |
| **Priority:** | High |
| **Trigger:** | The required account is authorized to be an admin. |
| **Pre-condition(s):** | * Only Admin can use this function. * Admin can create account for Students, Marketing Manager and Marketing Coordinator. |
| **Post-condition(s):** | * The system will save the account data that Admin has created. |
| **Basic Flow:** | 1. The use case starts when the Admin chooses to create an account on the main System menu. 2. The system provides the Admin with a form that includes all the information to fill out. 3. If entered, the required fields are not valid    * The system requires the user to re-enter the incorrectly entered fields.    * The system validates the fields after re-entering. 4. If the information entered is valid, the account will be created successfully. 5. The system adds new accounts |

### Create Date Submission

|  |  |
| --- | --- |
| **Use case ID:** | CDS-1.1 |
| **Use case name:** | Create Date Submission |
| **Brief description:** | Coordinator created that date and student based on that submission. |
| **Actor:** | Marketing Coordinator |
| **Priority:** | High |
| **Trigger:** | Coordinator wants to create a place for students to submit |
| **Pre-condition(s):** | * The Coordinator account has been successfully logged into the system |
| **Post-condition(s):** | * Create Date Submission was successfully * The system records the successful creation of the date submission to the Activity Log. |
| **Basic Flow:** | 1. Coordinator access the Green Tea University website  2. Coordinator chooses the Create Date Submission method  3. Coordinator will fill in information such as:  - Name of class, Course, ID Course number  - Select a date for the student to submit and a date to finish submitting  4. Save and Create Date Submission was successfully |

|  |  |
| --- | --- |
| **Use case ID:** | AS-1.1 |
| **Use case name:** | Assign |
| **Brief description:** | This use case describes the function assign each account in the system. |
| **Actor:** | Admin |
| **Priority:** | High |
| **Trigger:** | Admin wants to authorize an account such as: This account will be Marketing Coordinator, Students, … |
| **Pre-condition(s):** | * The admin account has been successfully logged into the system |
| **Post-condition(s):** | * Authorization for each account is successful * The system records the successful creation of the date submission to the Activity Log. |
| **Basic Flow:** | 1. Admin access the website. 2. The administrator chooses the Assign method 3. Admin will assign authority to each account (One account will take over certain rights)    * Marketing Manager    * Marketing Coordinator    * Student 4. Assign and save was successful |

### Assign role for user

### Comments

|  |  |
| --- | --- |
| **Use case ID:** | CM-1.1 |
| **Use case name:** | Comments |
| **Brief description:** | This function will allow Marketing Coordinator to want to comment on submissions from students, so that students can comment from the Marketing Coordinator and vice versa |
| **Actor:** | Marketing Coordinator, Students |
| **Priority:** | Medium |
| **Trigger:** | Marketing Coordinator wants to give feedback to students in order to create interaction between coordinator and students  Students can respond to comments from Coordinator |
| **Pre-condition(s):** | * The Student and Marketing Coordinator account has been successfully logged into the system * The student must submit the previous articles |
| **Post-condition(s):** | * Marketing Coordinator successfully commented * Students have successfully commented * After a successful comment, the system will notify the parties to know who commented to whom. * The system will save to Activity Log |
| **Basic Flow:** | 1. The Marketing Coordinator (or Students) accesses the Green Tea University website  2. Select method:  2.1 Marketing Coordinator will choose View Submission  2.2 Students will select My Course and click on the Course they  submitted (or Students can click the notification from Coordinator comment on that course).  3. After you have finished commenting  3.1 Marketing Coordinator will send comments to students on that post. The system will save that activity and will send an email notification to that student  3.2 Student will submit again and this comment will be reviewed by Coordinator again  4. The data will be saved automatically and sent to the database |

### Views

|  |  |
| --- | --- |
|  | VW-1.1 |
| **Use case name:** | Views |
| **Brief description:** | When Students and Guests want to see the report and articles, this Views function will help them see all the articles and reports |
| **Actor:** | Students, Guests |
| **Priority:** | Medium |
| **Trigger:** | Students (or Guests) want to Views their articles or reports |
| **Pre-condition(s):** | * The students (or guests) account has been successfully logged into the system |
| **Post-condition(s):** | * The students view the articles (guest view reports) successfully |
| **Basic Flow:** | 1. Students and Guests access the Green Tea University website  2. Select the methods:  2.1 Students choose My Course and just click on the articles they want to see  2.2 Guests choose Views and just click on the reports they want to see  3. The system records student view the articles (guests view the reports) sending in the Activity Log. |

# **Design**

## Website wireframes

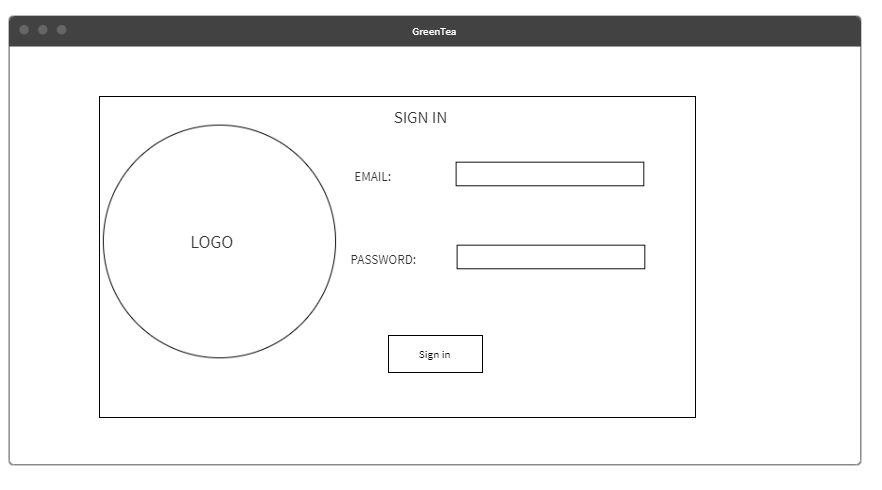
In this session, will show a design of the homepage, login pages and other pages

* Home Page:

(Figure 8: Home Page)

On the Homepage page we introduce our website as below we show the hot faculties on campus. (Figure 8)

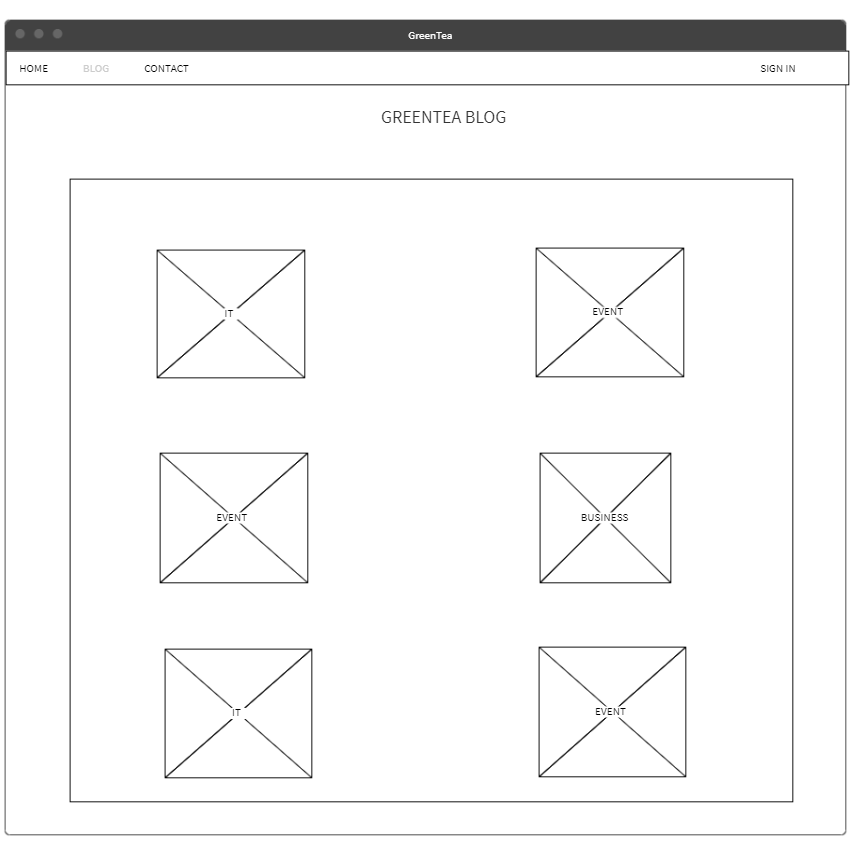
* Login:



(Figure 9: Login)

Above is the login page before you can log into the system. Users need to enter the Username and Password provided by the admin to log in to the system. (Figure 9)

* Blog:



(Figure 10: Blog)

The Blog Page makes it possible for everyone to post their articles on it for all to be read so that everyone can comment on it. (Figure 10)

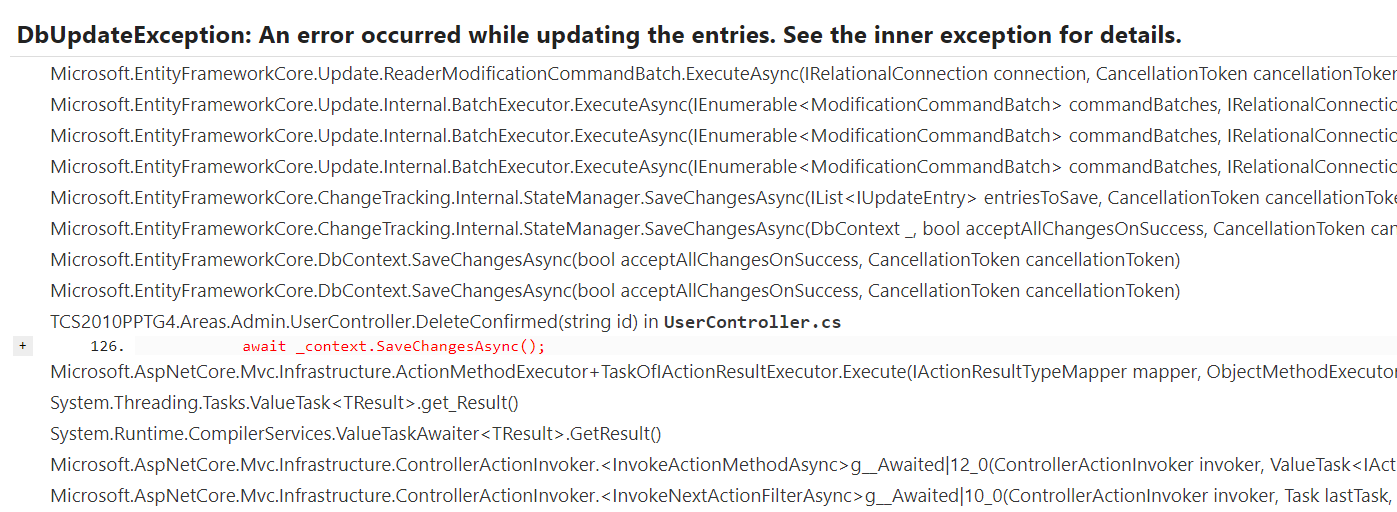
# **Testing**



## Test Log

Sprint 1 (28/1 - 10/2):

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No. | Function Testing | Description | Start Date | End Date | Duration | Note |
| 1. | Add Role | Check function Add Role for errors or not | 4/2/2021 | 4/2/2021 | 1 day |  |
| 2. | Delete Role | Check function Delete Role for errors or not | 5/2/2021 | 5/2/2021 | 1 day |  |
| 3. | Edit Role | Check function Edit Role for errors or not | 6/2/2021 | 6/2/2021 | 1 day |  |
| 4. | Assign Role | Check function Assign Role for errors or not | 7/2/2021 | 7/2/2021 | 1 day |  |
| 5. | Add Account | Try creating a new account and add it to the system | 8/2/2021 | 8/2/2021 | 1 day |  |
| 6. | Delete Account | Try delete a account | 9/2/2021 | 9/2/2021 | 1 day | Bug |
| 7. | Edit Account | Try Edit a account | 10/2/2021 | 10/2/2021 | 1 day | Bug |



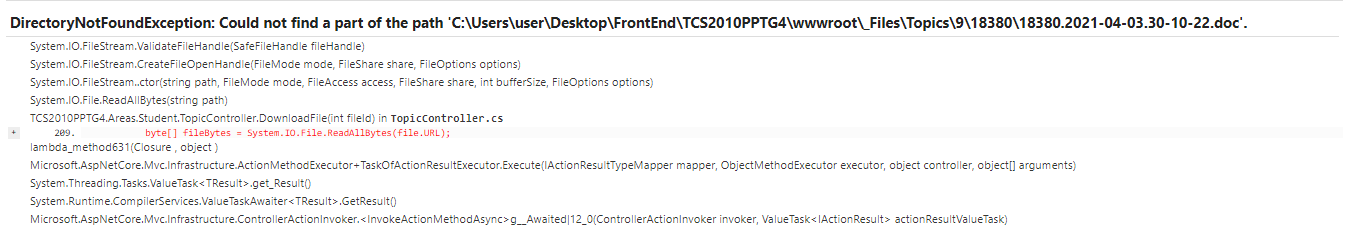
(Figure 11: Bug Delete,Add,Edit)

Sprint 2 (22/2 - 7/3):

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No. | Function Testing | description | Start Date | End Date | Duration | Note |
| 1. | Login | Try signing in with the account you created earlier | 22/2/2021 | 22/2/2021 | 1 day |  |
| 2. | Log out | Click Logout to logout | 23/2/2021 | 23/2/2021 | 1 day |  |
| 3. | Create Department | Create Department in the system | 24/2/2021 | 24/2/2021 | 1 day |  |
| 4. | Delete Department | Delete Department in the system | 25/2/2021 | 25/2/2021 | 1 day |  |
| 5. | Edit Department | Edit Department in the system | 26/2/2021 | 26/2/2021 | 1 day |  |

Sprint 3 (8/3 - 21/3):

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No. | Function Testing | description | Start Date | End Date | Duration | Note |
| 1. | Create Topic | Try Create topic in the system by account Coordinator | 8/3/2021 | 8/3/2021 | 1 day |  |
| 2. | Delete Topic | Try Delete topic in the system by account Coordinator | 9/3/2021 | 9/3/2021 | 1 day |  |
| 3. | Edit Topic | Try Edit Topic in the system by account Coordinator | 10/3/2021 | 10/3/2021 | 1 day |  |
| 4. | View Details Topic | Try view details in the system by account Coordinator | 11/3/2021 | 11/3/2021 | 1 day |  |
| 5. | Mark | Try Mark in the system by account Coordinator | 12/3/2021 | 12/3/2021 | 1 day |  |
| 6. | Comment | Try comment somethings in the box chat | 13/3/2021 | 13/3/2021 | 1 day |  |
| 7. | Delete Comment | Delete comment in system | 14/3/2021 | 14/3/2021 | 1 day |  |
| 8. | Download files | Download files by Coordinator account | 15/3/2021 | 15/3/2021 | 1 day | Bug |



(Figure 12: Bug Download Files)

Sprint 4 (22/3 - 5/4):

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No. | Function Testing | Description | Start Date | End Date | Duration | Note |
| 1. | Comment | Comment somethings by Student Account | 22/3/2021 | 22/3/2021 | 1 day |  |
| 2. | Delete Comment | Try delete comment by Student Account | 23/3/2021 | 23/3/2021 | 1 day |  |
| 3. | Upload files | Try upload files by Student account | 24/3/2021 | 24/3/2021 | 1 day |  |
| 4. | Download files | Try Download files by Student account | 25/3/2021 | 25/3/2021 | 1 day |  |
| 5. | View Mark | Try View mark by Student account | 26/3/2021 | 26/3/2021 | 1 day |  |

## Test Plan

Sprint 1:

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Function | Testing | Results |
| 1. | Add role by Admin | Login account Admin. Click “Users” to Add role | Add complete |
| 2. | Delete role by Admin | Login account Admin. Click “Users” to Delete role | Delete complete |
| 3. | Edit role by Admin | Login account Admin. Click “Users” to Edit role | Edit complete |
| 4. | Assign role by Admin | Login account Admin. Click “Users” to Edit role | Assign complete |
| 5. | Role by Admin | Login account Admin. Click “Users” to Role | Role complete |
| 6. | Add Account | Admin add account to System | Add complete |
| 7. | Delete Account | Admin Delete Account to System | Bug |
| 8. | Edit Account | Admin Edit Account to System | Bug |

Sprint 2:

|  |  |  |  |
| --- | --- | --- | --- |
| No | Function | Testing | Results |
| 1. | Login | Enter “UserName” and “Password” to Login | Login Complete |
| 2. | Logout | Click “Log Out” | Log out complete |
| 3. | Create Department | Login Account Admin. Click “ Department” and click “Create” to Create | Create Department complete |
| 4. | Delete Department | Login Account Admin. Click “ Department” and click “Delete” to Delete | Delete Department complete |
| 5. | Edit Department | Login Account Admin. Click “ Department” and click “Edit” to Edit | Edit Department complete |

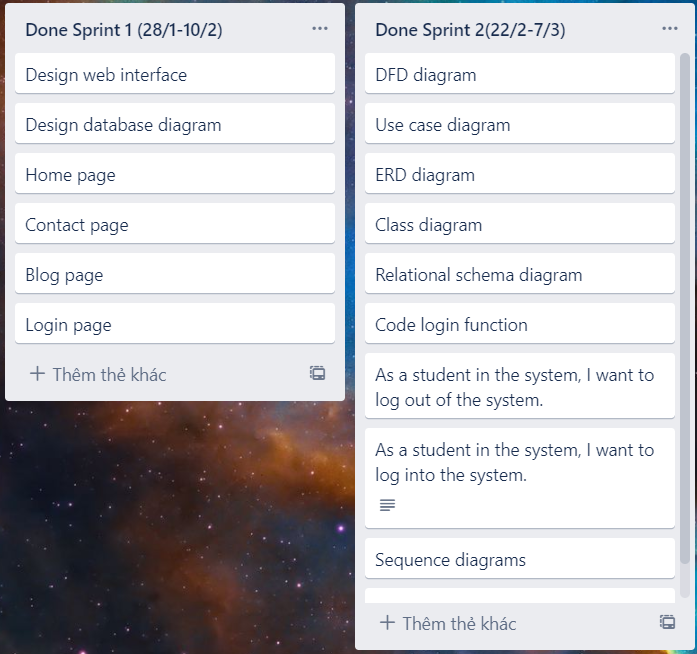
Sprint 3:

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Function | Testing | Results |
| 1. | Create Topic by Coordinator | Login account Coordinator. Click “Topic” and click “Create” to Create | Create Topic  complete |
| 2. | Delete Topic by Coordinator | Login account Coordinator. Click “Topic” and click “Delete” to Delete | Delete Topic complete |
| 3. | Edit Topic by Coordinator | Login account Coordinator. Click” Topic” and click “Edit” to Edit | Edit Topic complete |
| 4. | View Details by Coordinator Topic | Login account Coordinator. Click “Topic” to view Details topic | View Complete |
| 5. | Mark by Coordinator | Login account Coordinator. Click “Details” to Mark | Mark complete |
| 6. | Comment by Coordinator | Login account Coordinator. Click “Details” to Comment | Comment complete |
| 7. | Delete comment by Coordinator | Login account Coordinator. Click “ Details” to Delete Comment | Delete comment complete |
| 8. | View status details by Manager | Login account Manager. Click “Topic” and Click “Details” to View | View Complete |
| 9. | Download files by Coordinator | Login account Coordinator. Click “Topic” and click “Download files” | Bug |

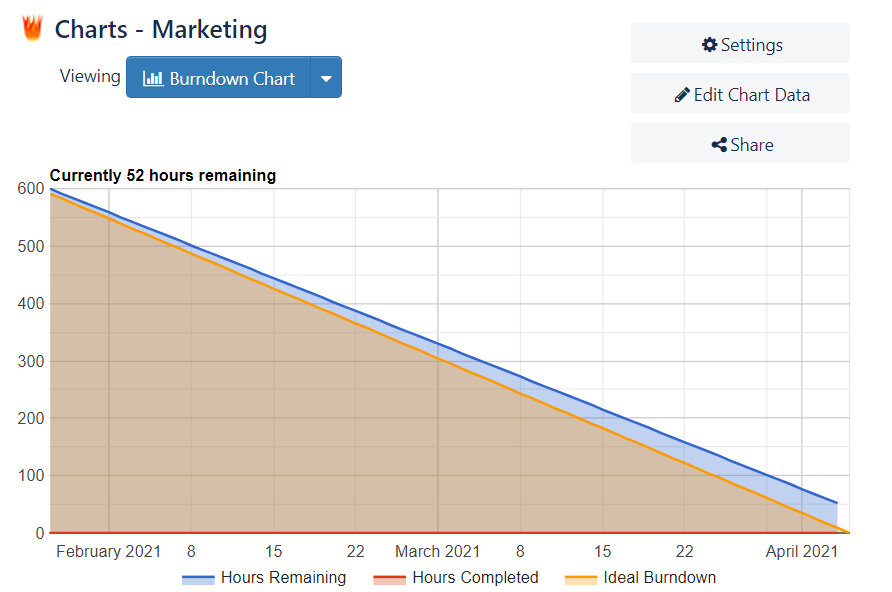
Sprint 4:

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Function | Testing | Results |
| 1. | Comment by Student | Login account Student. Click “Topic” and “Click “Details” to Comment | Comment complete |
| 2. | Delete comment by Student | Login account Student. Click “ Details” to Delete Comment | Delete comment complete |
| 3. | Upload files | Login account Student. Click “Details” and click “Upload files” | Upload files complete |
| 4. | Download files | Login account Student. Click “Details” and click “ Download files” | Bug |
| 5. | View Mark | Login account Student. Click “Details” to View Mark | View Mark complete |

# **Agile method**



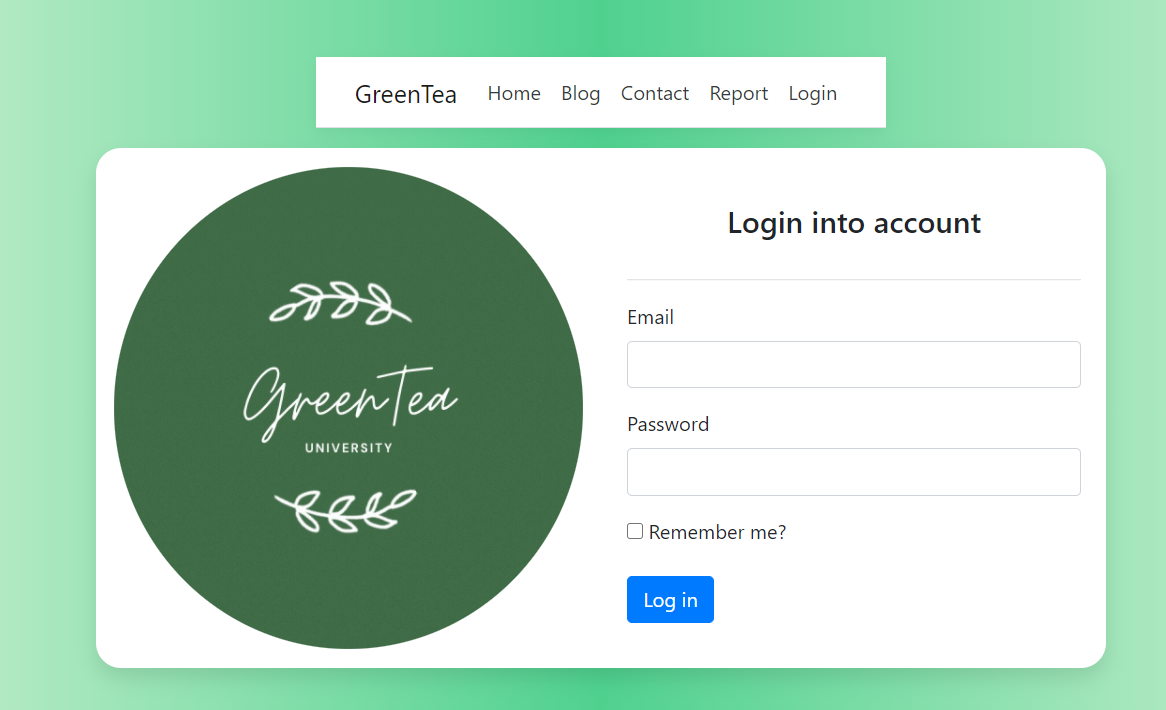




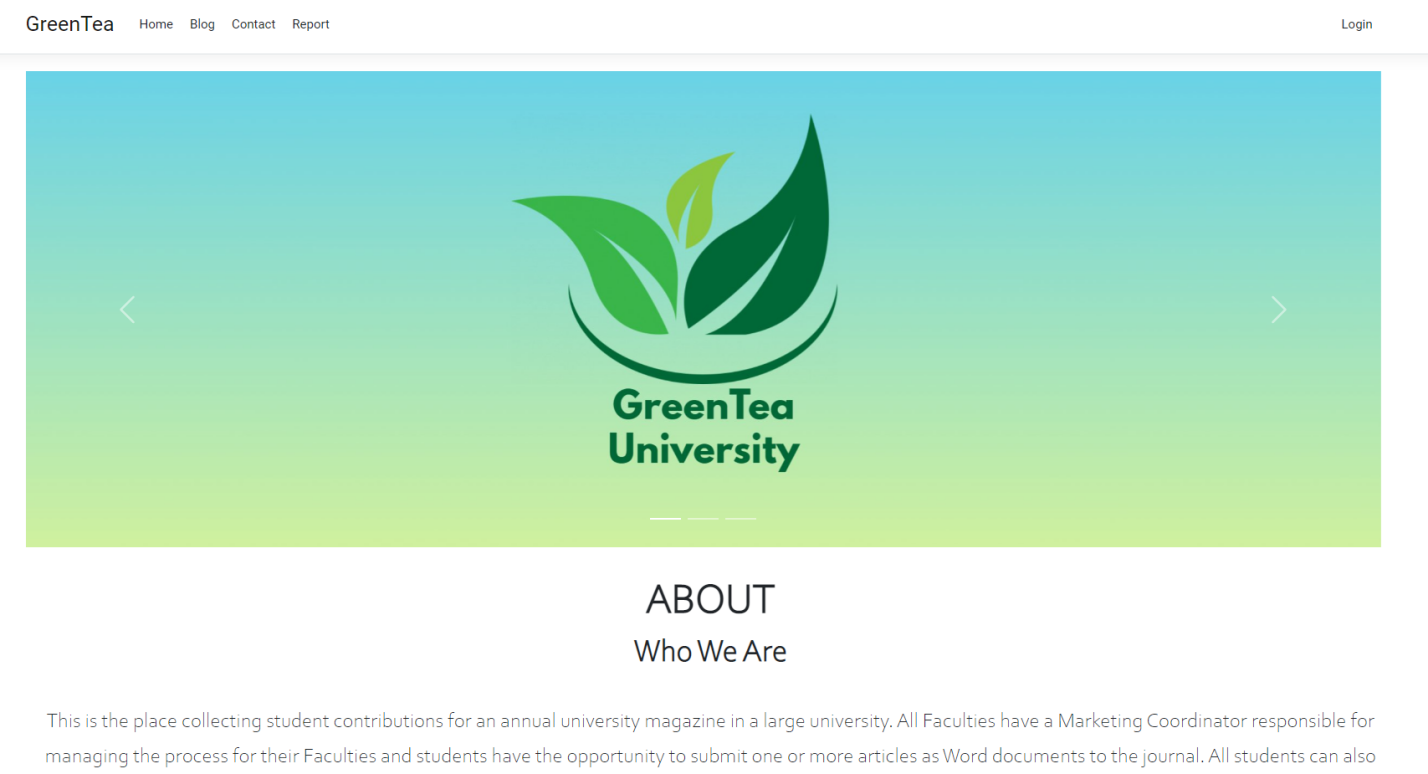
(Figure 13: Burndown Chart)

Burndown chart shows the speed of completing the job in the ideal time compared with the speed of completing the job in real time. (Figure 13)

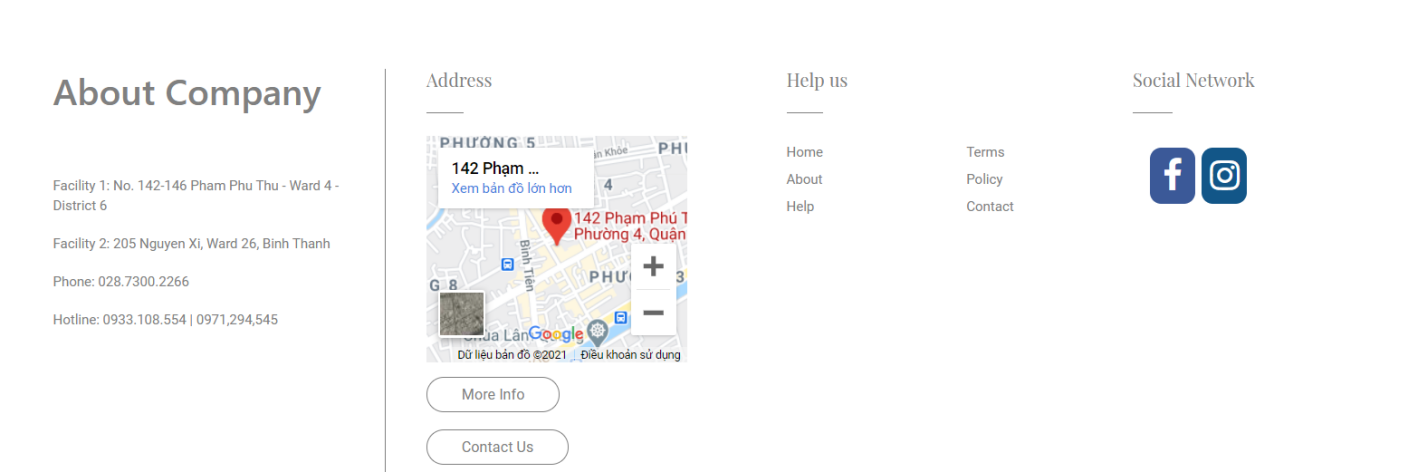
# **Product**



(Figure 14: Login)



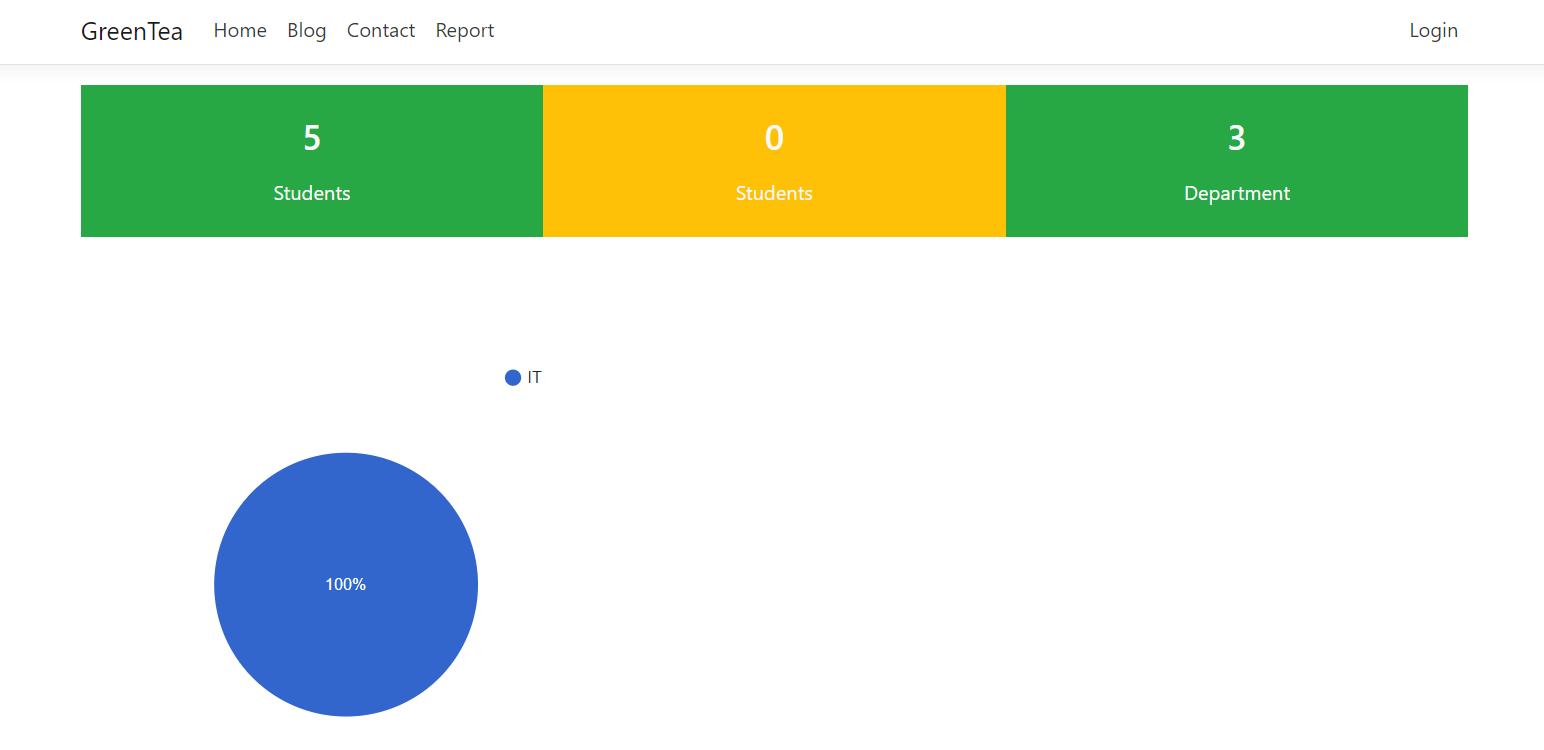
(Figure 15: Home Page)



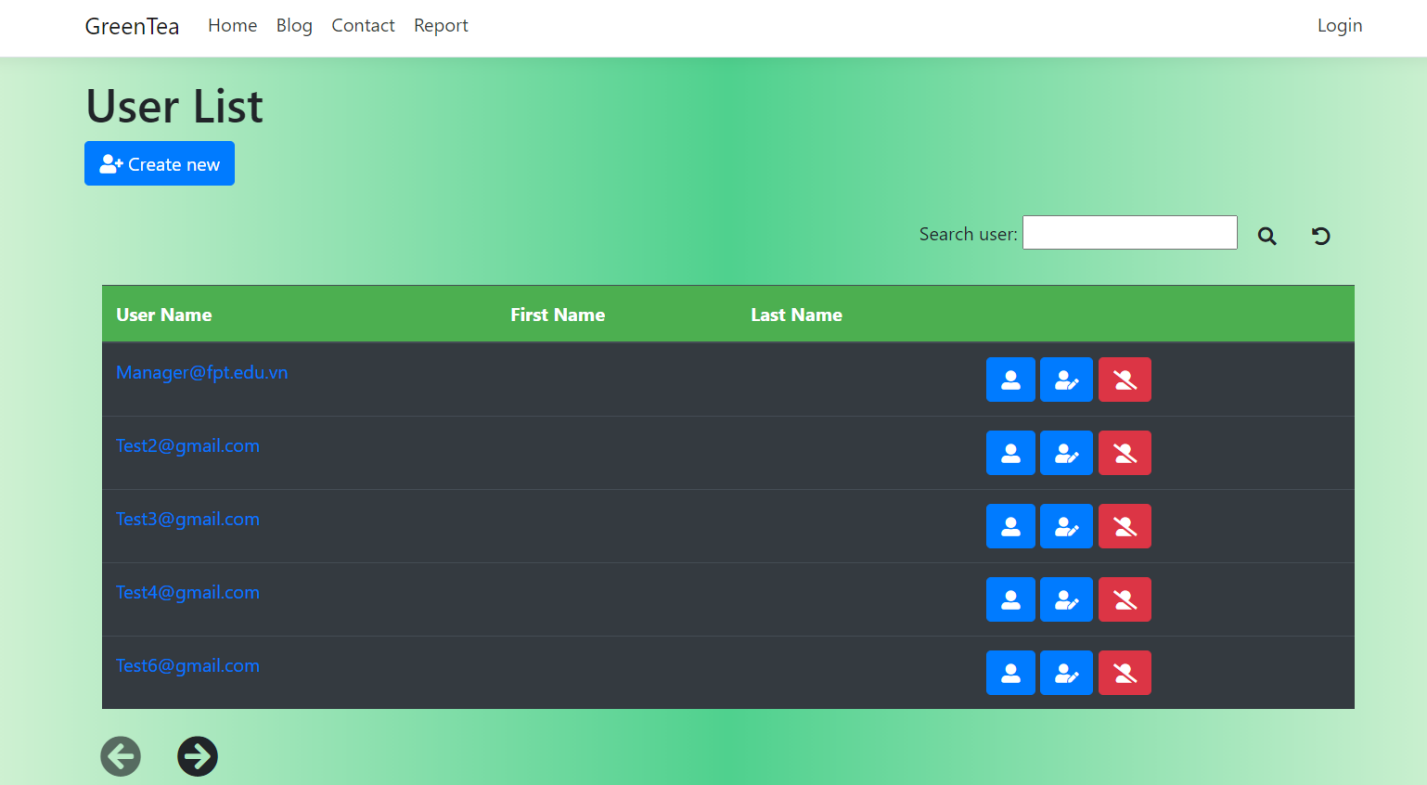
(Figure 16: About Company)



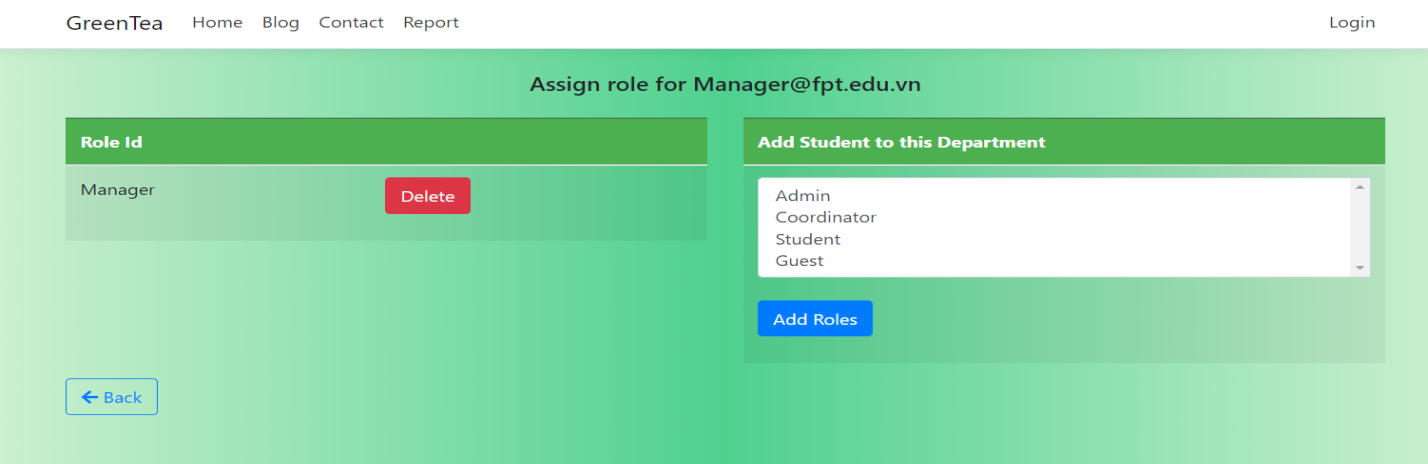
(Figure 17: Green Tea Blog)



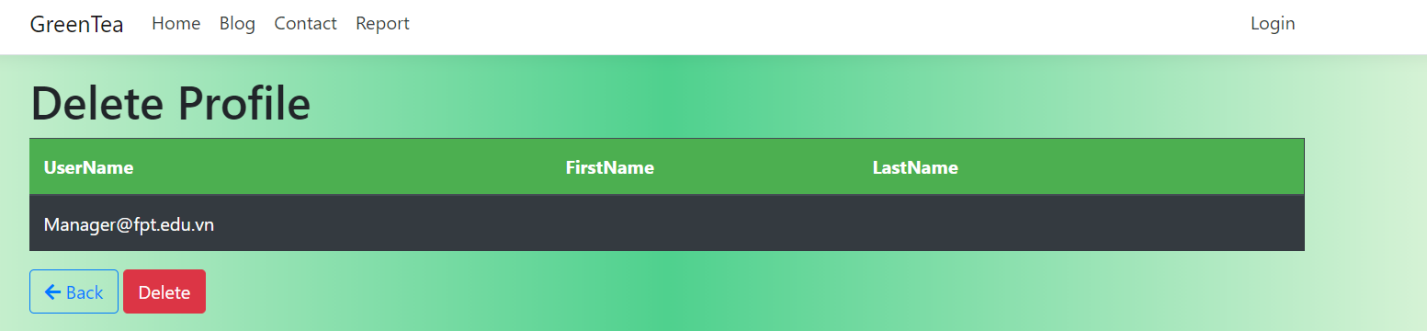
(Figure 18: Statistical)



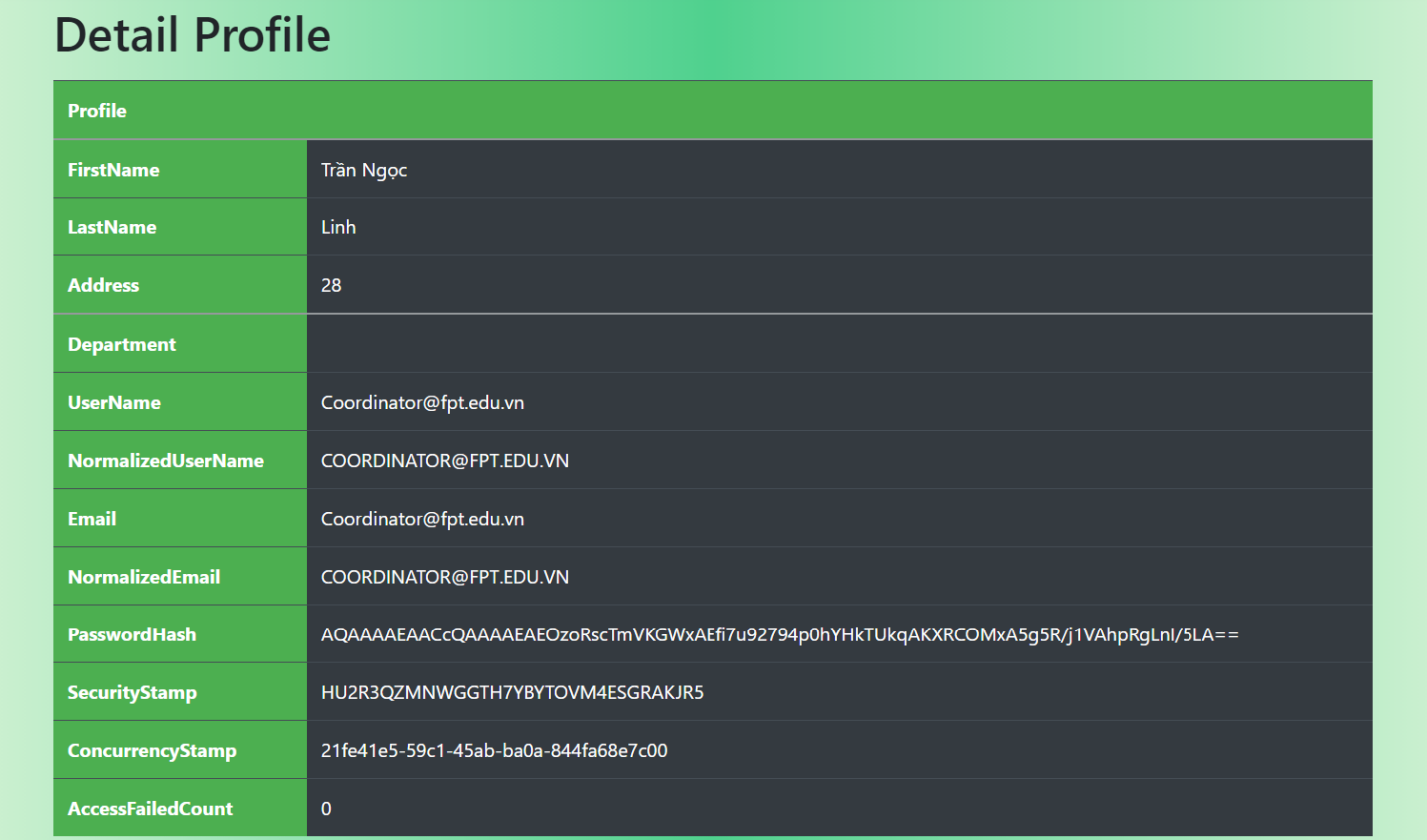
(Figure 19: User List)



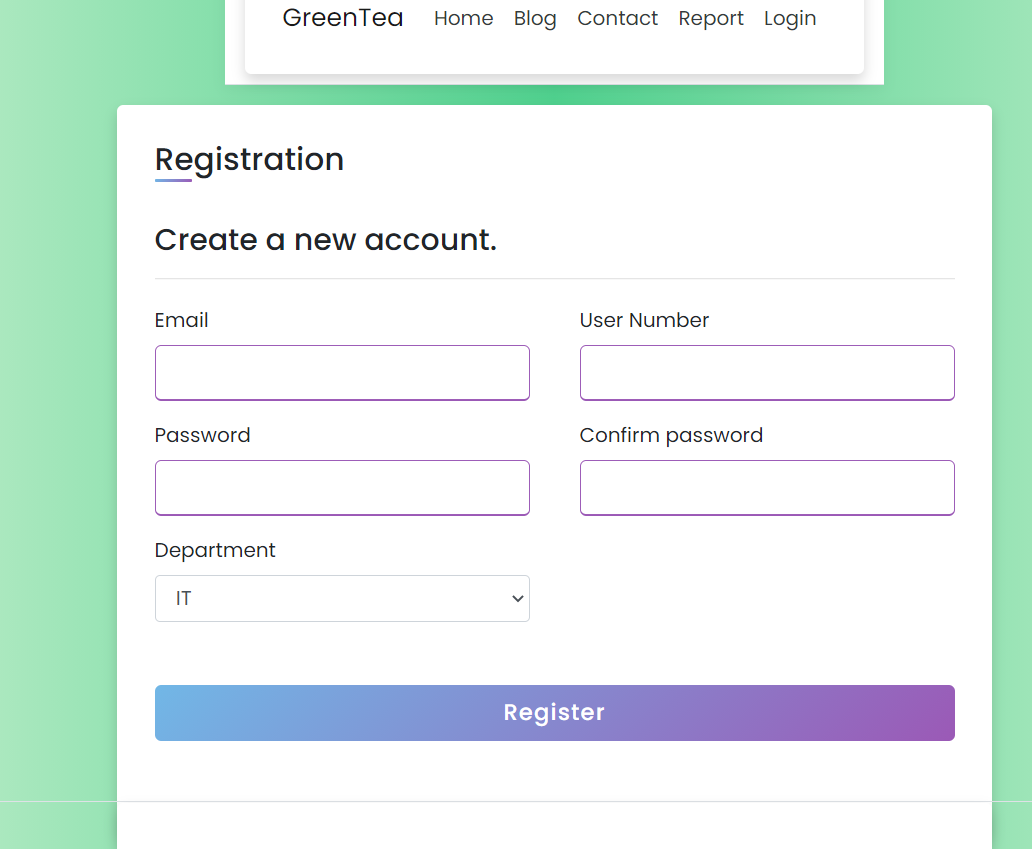
(Figure 20: Add Role Admin)



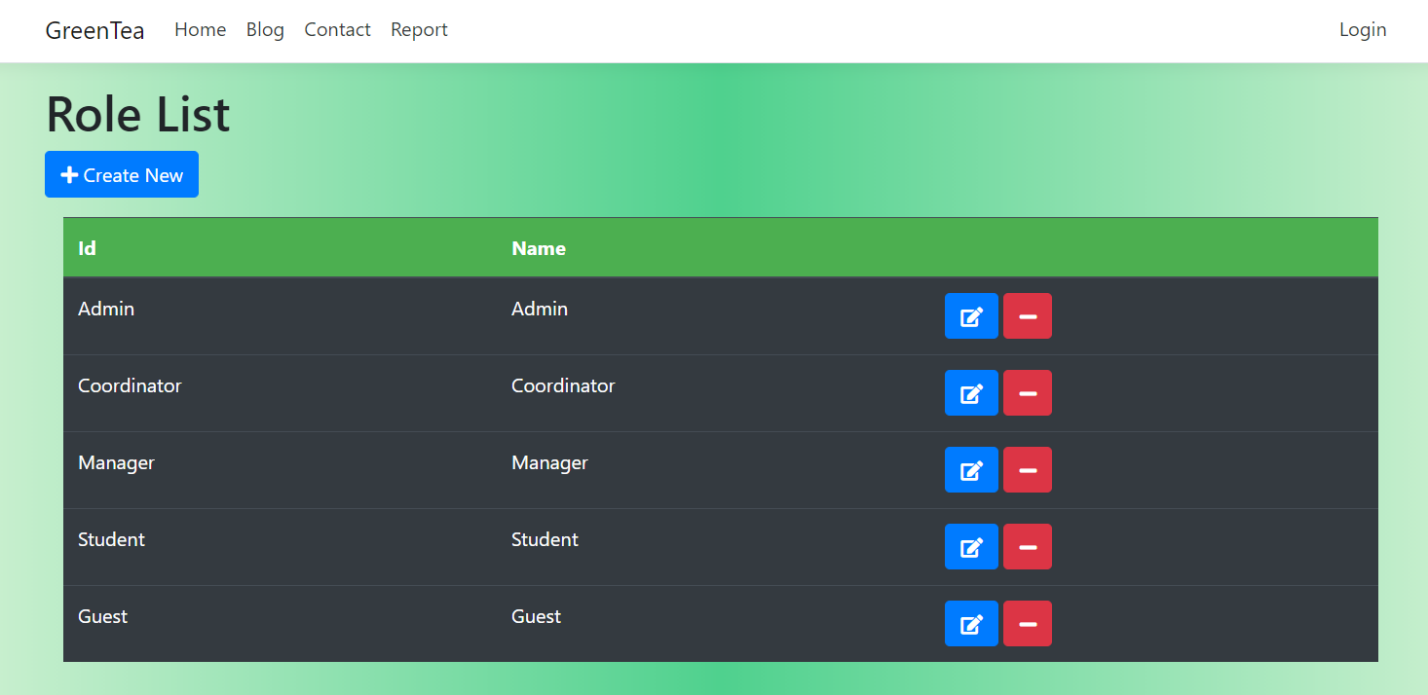
(Figure 21: Delete Profile Admin)



(Figure 22: Detail profiles Admin)



(Figure 23: Registration Admin)



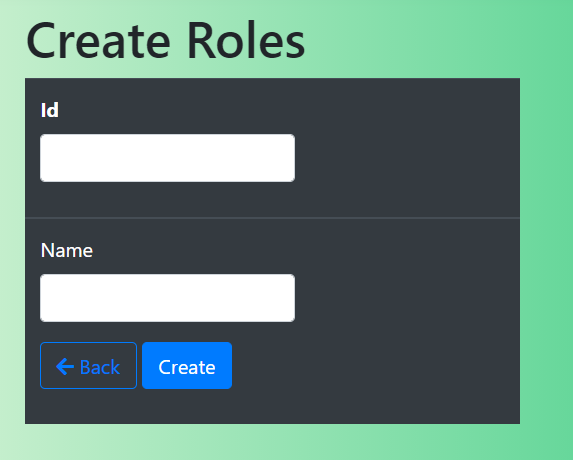
(Figure 24: Role List Admin)



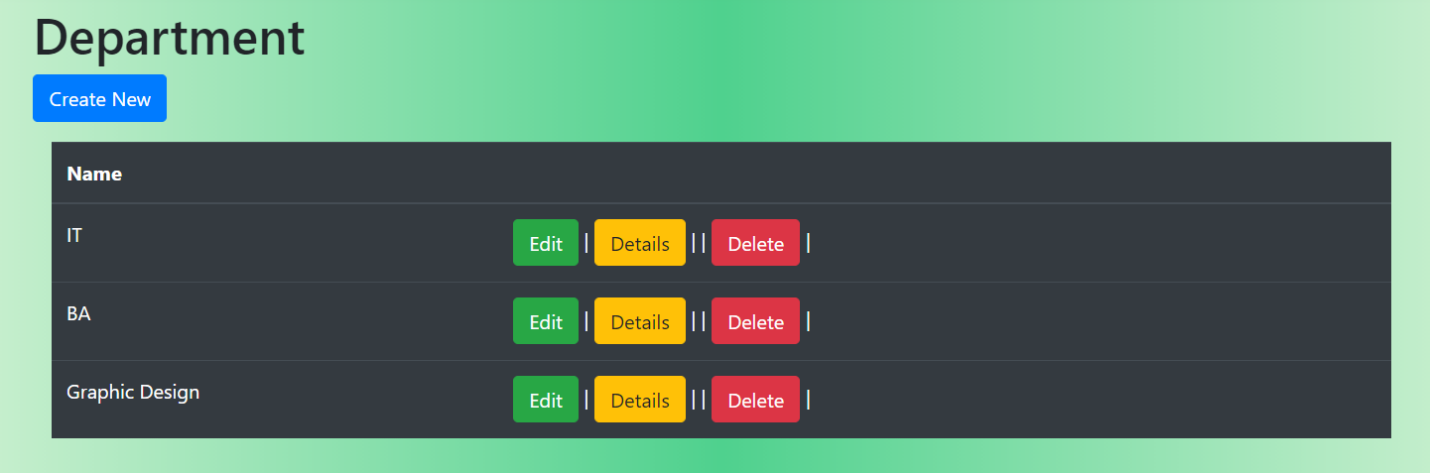
(Figure 25: Edit Roles Admin)



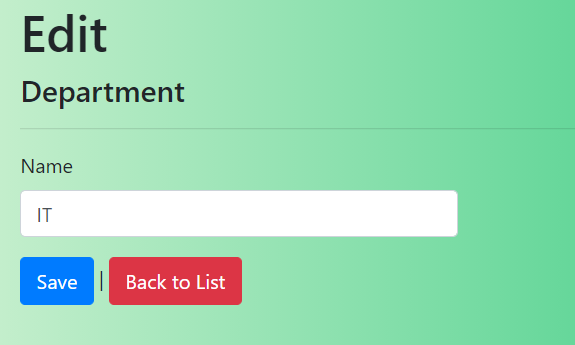
(Figure 26: Delete Roles Admin)



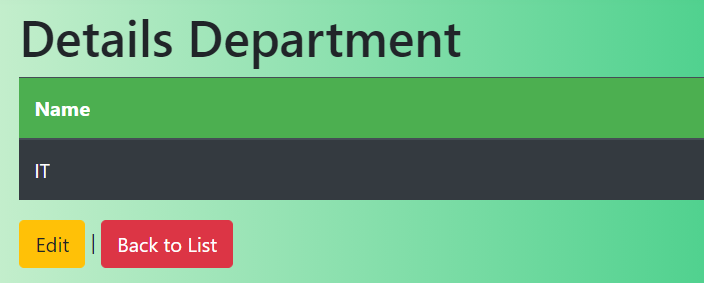
(Figure 27: Create Roles Admin)



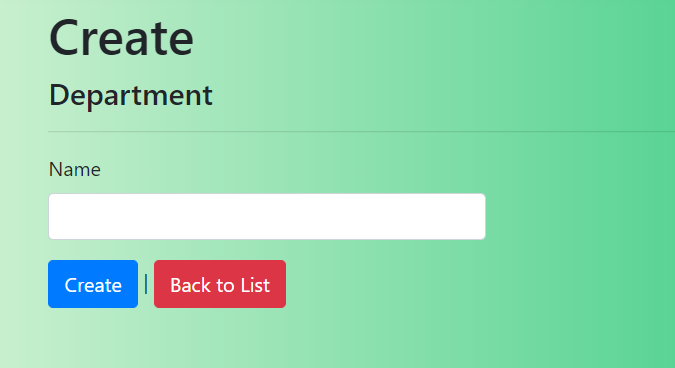
(Figure 28: Department Admin)



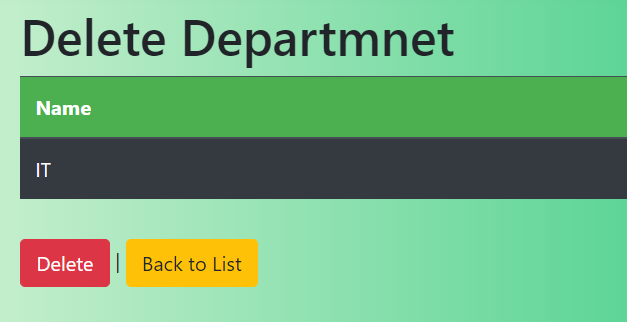
(Figure 29: Edit Department Admin)



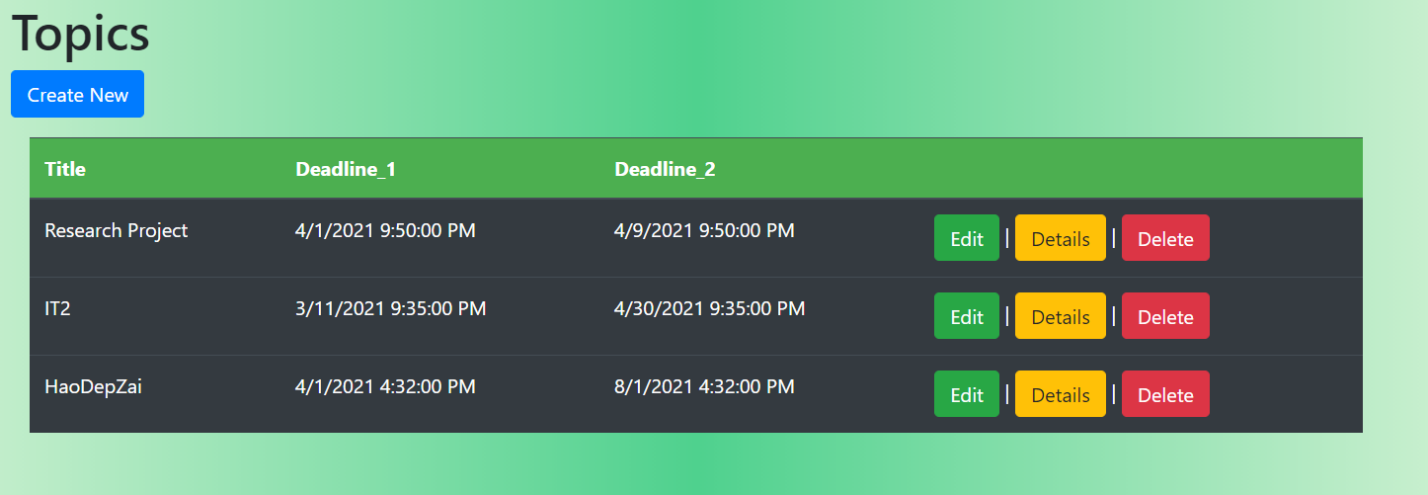
(Figure 30: Details Department Admin)



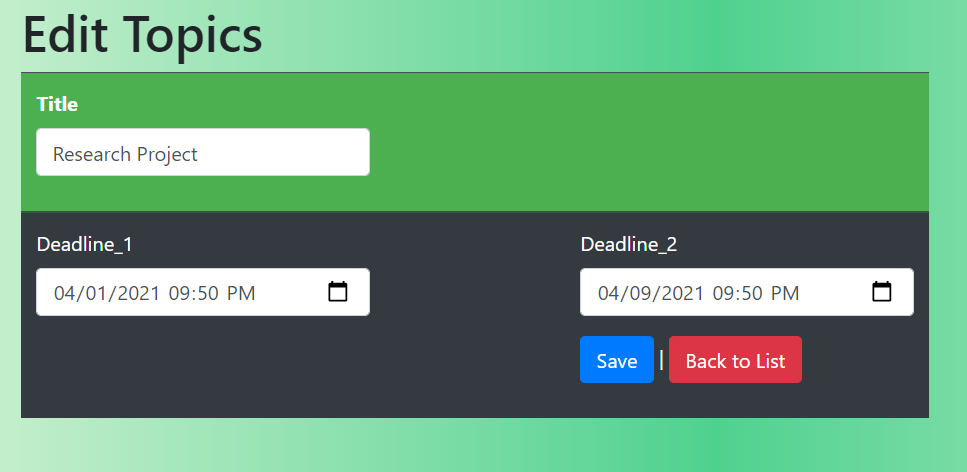
(Figure 31: Create Department Admin)



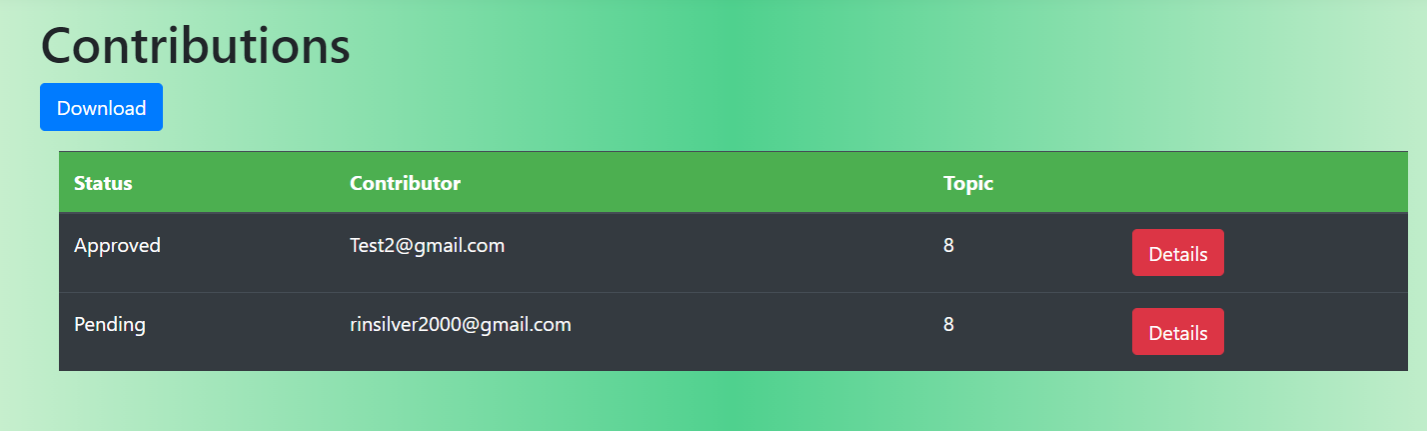
(Figure 32: Delete Department Admin)



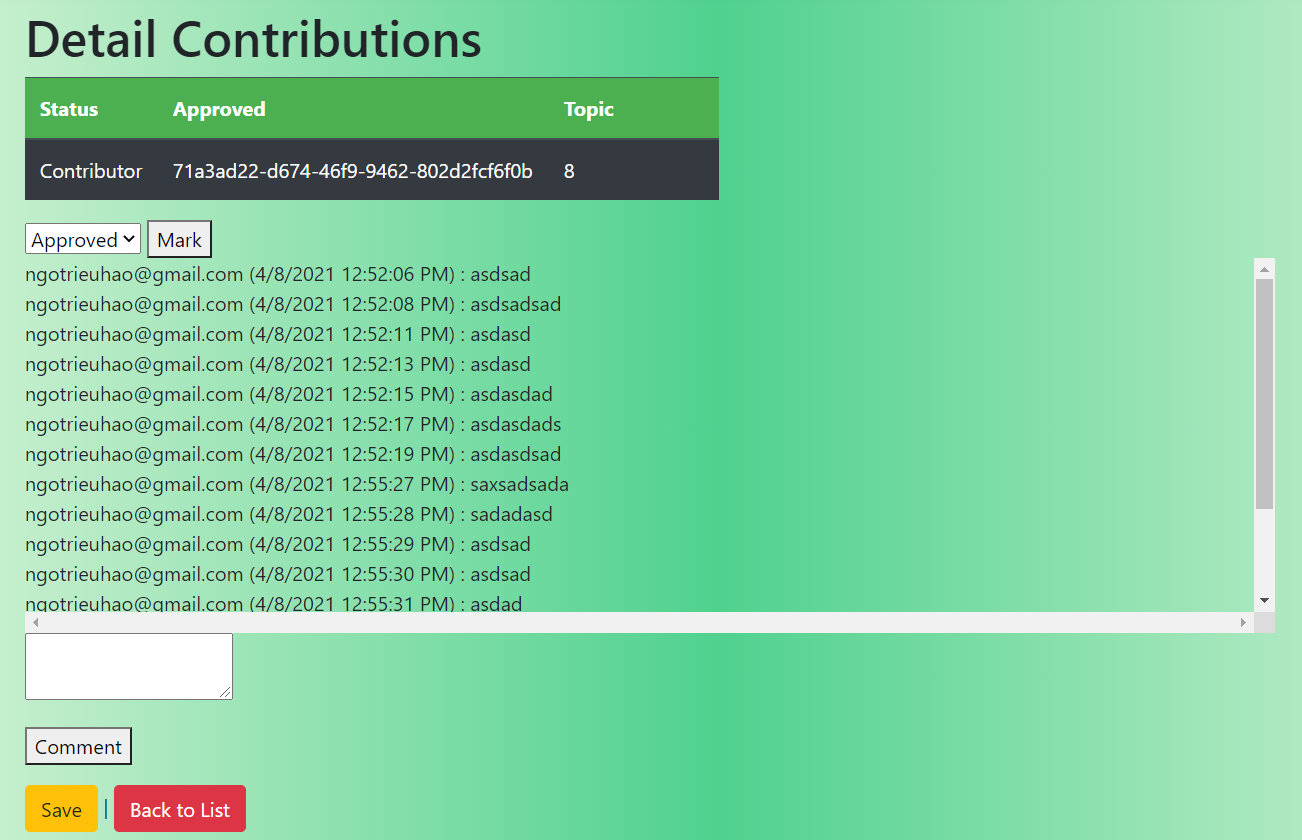
(Figure 33: Topics Coordinator)



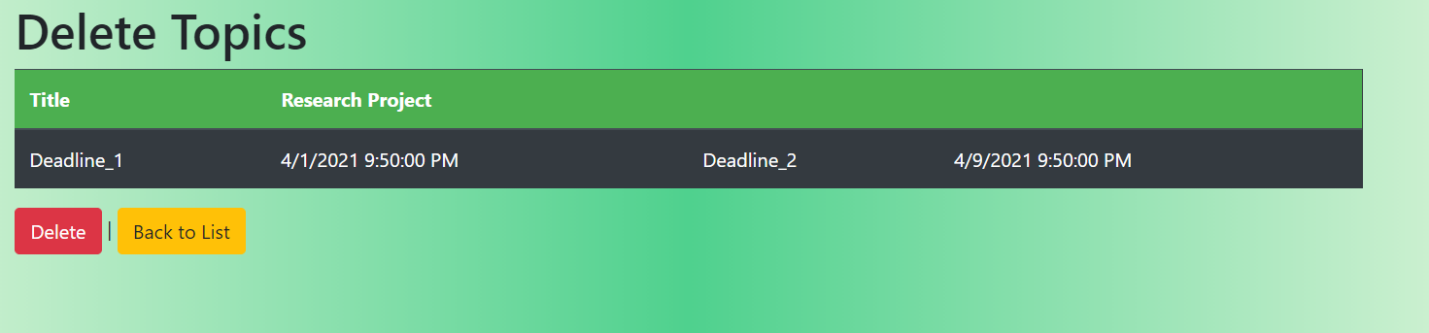
(Figure 34: Edit Topic Coordinator)



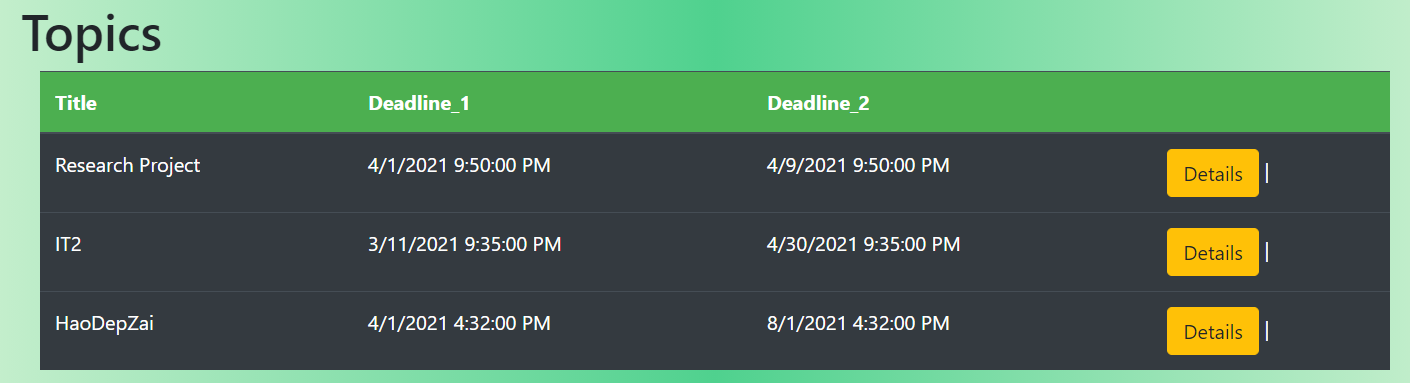
(Figure 35: Contributions Coordinator)



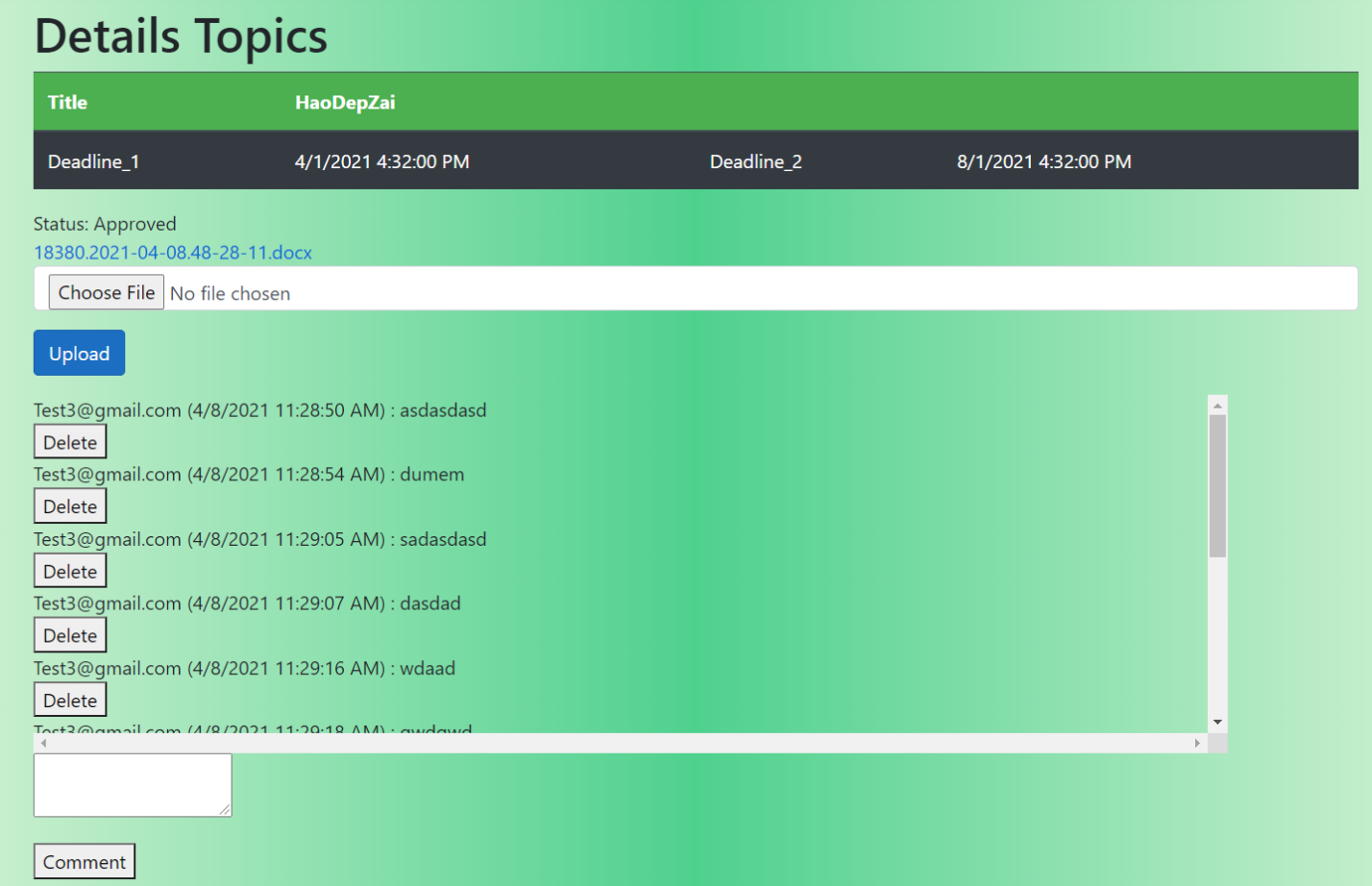
(Figure 36: Detail Contribution Student)



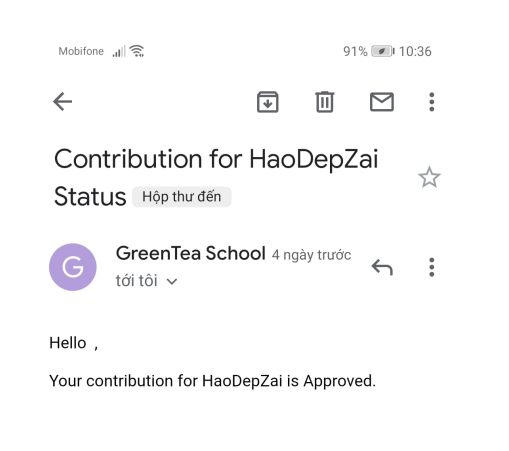
(Figure 37: Delete Topics Coordinator)



(Figure 38: Topics Student)



(Figure 39: Details Topics Student)



(Figure 40: Email)

# **Links**

Trello: <https://trello.com/invite/b/szNyXDlw/57996663f0a980685fc2549444faa4b3/marketing>

Google Drive: https://drive.google.com/drive/folders/1Qu4TJBjqZmPBbu-7kHf-2Bmi30HJDZSh?usp=sharing

GitHub: https://github.com/Lethanhduc/green?fbclid=IwAR3jzAQ4exBlIlaspt6dW4BtC1Q8LFd6djH-BHSq5dNoMV7qyhc8FZEC4hI

# **References**