

ENTERPRISE WEB SOFTWARE DEVELOPMENT

Group report

This is group report of group 3 Enterprise Web Software Development

Group 3

Table of Contents

Table of Contents.....	1
List of Figures	4
List of Tables	9
Cover page	10
I. Introduction	11
II. Requirement	11
1. Functional 1:.....	13
2. Functional 2:.....	14
3. Functional 3:.....	15
4. Functional 4:.....	15
5. Functional 5:.....	16
6. Functional 6:.....	17
7. Functional 7:.....	17
8. Functional 8:.....	18
9. Functional 9:.....	19
10. Functional 10:.....	19
11. Functional 11:.....	20
12. Functional 12:.....	22
13. Functional 13:.....	22
14. Functional 14:.....	23
III. Database	24
1. Introduction.....	24
2. Physical design:.....	25

IV.	Design	33
1.	Sitemap.....	33
2.	Wireframes.....	34
a.	Home page.....	34
b.	Login page.....	35
c.	Admin.....	36
d.	Marketing Manager	44
e.	Marketing Coordinator	49
f.	Student.....	54
g.	Guest.....	58
3.	Interface	60
a.	Home page.....	60
b.	Login page.....	60
c.	Admin.....	61
d.	Marketing Manager	66
e.	Marketing Coordinator	69
f.	Student.....	71
g.	Guest.....	73
V.	Implementation	74
1.	Techniques	74
a.	Back-end	74
b.	Front-end	75
c.	Database	75
d.	Hosting.....	76

2.	Flowcharts and features	77
a.	Marketing Manager	77
b.	Marketing Coordinator	84
c.	Student	90
d.	Admin.....	94
e.	Guest.....	97
VI.	Testing.....	100
1.1.	Test plan.....	100
1.1.1.	Team members	100
1.1.2.	Scope.....	100
1.1.3.	Test Approach	101
1.1.4.	Test Environment	101
1.1.5.	Assumptions/Risks	102
1.1.6.	Test schedule.....	104
1.1.7.	Deliverables.....	105
1.2.	Sprint 1.....	105
1.2.1.	Test Cases and Test Logs of Sprint 1	105
1.2.2.	Sprint 1 backlog.....	121
1.2.3.	Sprint 1 burndown chart	121
1.3.	Sprint 2	123
1.3.1.	Test Cases and Test Logs of Sprint 2	123
1.3.2.	Sprint 2 backlog.....	128
1.3.3.	Sprint 2 burndown chart	133
1.4.	Sprint 3	134

1.4.1. Test Cases and Test Logs of Sprint 3	134
1.4.3. Sprint 3 burndown chart	143
1.5. Test evidence	144
VII. Agile	156
2.1. Product backlog.....	156
2.2. Product burndown chart	159
2.3. Team challenges.....	159
2.4. Minutes of meetings	162
VIII. Conclusion.....	164

List of Figures

Figure 1: Functional 1.....	14
Figure 2: Functional 2.....	14
Figure 3: Functional 3.....	15
Figure 4: Functional 4.....	16
Figure 5: Functional 5.....	16
Figure 6: Functional 6.....	17
Figure 7: Functional 7.....	18
Figure 8: Functional 8.....	18
Figure 9: Functional 9.....	19
Figure 10: Functional 10.....	20
Figure 11: Functional 11.1.....	20
Figure 12: Functional 11.2.....	21
Figure 13: Functional 12.....	22

Figure 14: Functional 13.....	22
Figure 15: Functional 14.....	23
Figure 16: ERD of system.....	24
Figure 17: use case Student of system	26
Figure 18: use case Marketing manager of system	27
Figure 19: use case Marketing coordinator of system	27
Figure 20: use case Admin of system	28
Figure 21: use case Guest of system	29
Figure 22: data security code	30
Figure 23: data validation code	30
Figure 24: fire base authentication database online	31
Figure 25: fire base real-time database online	31
Figure 26: fire base storage database online.....	32
Figure 27: Sitemap	33
Figure 28: Wireframes homepage.....	34
Figure 29: Wireframes Login page	35
Figure 30: Wireframes Admin Index.....	36
Figure 31: Wireframes Admin create Student account	37
Figure 32: Wireframes Admin create Marketing Coordinator account	38
Figure 33: Wireframes Admin create Guest account	39
Figure 34: Wireframes Admin create Marketing Manager account	40
Figure 35: Wireframes Admin Edit account.....	41
Figure 36: Wireframes Admin Index course	42
Figure 37: Wireframes Admin Edit course.....	43
Figure 38: Wireframes Marketing Manager Index	44
Figure 39: Wireframes Marketing Manager View Course	45
Figure 40: Wireframes Marketing Manager View Submissions.....	46
Figure 41: Wireframes Marketing Manager View Contributions	47
Figure 42: Wireframes Marketing Manager View Exceptional.....	48

Figure 43: Wireframes Marketing Coordinator Index	49
Figure 44: Wireframes Marketing Coordinator Mark	50
Figure 45: Wireframes Marketing Coordinator Marking student	51
Figure 46: Wireframes Marketing Coordinator Marking Index chat with student	52
Figure 47: Wireframes Marketing Coordinator Marking chat with student	53
Figure 48: Wireframes Student Index	54
Figure 49: Wireframes Student Submit file	55
Figure 50: Wireframes Student Index chat Marketing Coordinator	56
Figure 51: Wireframes Student chat Marketing Coordinator	57
Figure 52: Wireframes Guest Index	58
Figure 53: Wireframes Guest view submission	59
Figure 54: Interface Home page	60
Figure 55: Interface Login page	60
Figure 56: Interface Admin Index	61
Figure 57:Interface Admin create Admin account	61
Figure 58: Interface Admin create student account	62
Figure 59: Interface Admin create Marketing Coordinator account	62
Figure 60: Interface Admin create Guest account	63
Figure 61: Interface Admin create Marketing Manager account	63
Figure 62: Interface Admin Edit account	64
Figure 63: Interface Admin Index course	65
Figure 64: Interface Admin Edit course	65
Figure 65: Interface Marketing Manager Index	66
Figure 66: Interface Marketing Manager View Courses	66
Figure 67: Interface Marketing Manager View Submissions	67
Figure 68: Interface Marketing Manager View Contributions	67
Figure 69: Interface Marketing Manager View Exceptional	68
Figure 70: Interface Marketing Coordinator Index	69
Figure 71: Interface Marketing Coordinator Mark	69

Figure 72: Interface Marketing Coordinator Marking student	70
Figure 73: Interface Marketing Coordinator Marking Index chat with student.....	70
Figure 74: Interface Marketing Coordinator Marking chat with student	71
Figure 75: Interface Student Index.....	71
Figure 76: Interface Student Submit file	72
Figure 77: Interface Student Index chat Marketing Coordinator.....	72
Figure 78: Interface Student chat Marketing Coordinator	73
Figure 79: Interface Guest Index	73
Figure 80: Interface Guest view submission.....	74
Figure 81: Flowcharts View exceptional reports	77
Figure 82: Features View exceptional reports.....	78
Figure 83: Flowcharts View all selected submissions	79
Figure 84: Features View all selected submissions.....	80
Figure 85: Flowcharts Download submissions by ZIP format	81
Figure 86: Features Download submissions by ZIP format.....	82
Figure 87: Flowcharts View statistics	83
Figure 88: Flowcharts View submissions from students in specific faculty	84
Figure 89: Features View submissions from students in specific faculty.....	85
Figure 90: Flowcharts Receive notifications (emails) for new submissions	86
Figure 91: Features Receive notifications (emails) for new submissions	87
Figure 92: Flowcharts Give comments for each submission.....	88
Figure 93: Feature Give comments for each submission.....	88
Figure 94: Flowcharts Determine to accept or reject submissions.....	89
Figure 95: Feature Determine to accept or reject submissions.....	89
Figure 96: Flowchart Submit article in Word format (1 or more) before first closure date, can edit before final closure date (after first closure date)	90
Figure 97: Feature Submit article in Word format (1 or more) before first closure date, can edit before final closure date (after first closure date)	91
Figure 98: Flowchart Upload images.....	92

Figure 99: Feature Upload images	92
Figure 100: Flowchart Read comments for own submissions	93
Figure 101: Feature Read comments for own submissions.....	93
Figure 102: Flowchart Create new accounts	94
Figure 103: Feature Create new accounts.....	95
Figure 104: Flowchart Set and edit closure date	96
Figure 105: Feature Set and edit closure date	97
Figure 106: Flowchart View selected reports for each faculty	98
Figure 107: Feature View selected reports for each facult.....	99
Figure 108: Sprint 1 Burndown Chart.....	122
Figure 109: Sprint 2 Burndown Chart.....	133
Figure 110: Sprint 3 Burndown Chart.....	143
Figure 111: Test evidence of the login function	144
Figure 112: Test evidence of the logout function.....	145
Figure 113: Test evidence of the function to create and edit update time, closing time	145
Figure 114: Test evidence of the function to create account.....	146
Figure 115: Test evidence of the function to store and preserve all system data	147
Figure 116: Test evidence of the function to submit articles	147
Figure 117: Test evidence of the function to read comments for own submissions	148
Figure 118: Test evidence of the function to update contributions until a final closure date ...	149
Figure 119: Test evidence of the function to accept the terms and conditions	149
Figure 120: Test evidence of the function to receive email notifications.....	150
Figure 121: Test evidence of the function to make comments for student's articles.....	151
Figure 122: Test evidence of the function to view contributions from students in faculty	151
Figure 123: Test evidence of the function to interact with the students in faculty.....	152
Figure 124: Test evidence of the function to select contributions to publish	152
Figure 125: Test evidence of the function to view all the selected contributions	153
Figure 126: Test evidence of the function to view the exceptional reports	154
Figure 127: Test evidence of the function to download all selected contributions as zip file ...	154

Figure 128: Test evidence of the function to keep track of faculty data	155
Figure 129: Test evidence of the function to view faculty selected reports.....	155
Figure 130: Product Burndown Chart.....	159

List of Tables

Table 1: Requirement.....	13
Table 2: Physical design.....	26
Table 3: Team members and Roles	100
Table 4:Risk	103
Table 5: Test schedule.....	104
Table 6: Sprint 1	117
Table 7: Test case and Test log for student account	120
Table 8: Sprint 1 backlog	121
Table 9: Test Cases and Test Logs of Sprint 2 for admin account and marketing coordinator account	127
Table 10: Sprint backlog 2	132
Table 11: Test Cases and Test Logs of Sprint 3 for admin account and marketing coordinator account	135
Table 12: Test Cases and Test Logs for marketing manager account and guest account	137
Table 13: Sprint 3 backlog	142
Table 14: Product backlog.....	158
Table 15: Minutes of meetings.....	164

Cover page

Group members:

- Nguyen Hoang Thuyen, ID: 001181275, Role: Programmer, Information architect, Product owner
- Pham Ngoc Bao, ID: 001181141, Role: Web designer, Programmer
- Do Ho Nguyen, ID: 001181241, Role: Tester
- Nguyen Van Tuan, ID: 001181305, Role: Scrum master, Tester
- Do The Vinh, ID: 001181111, Role: Database designer, Web designer
- Ha Ngoc Vi, ID: 001143934, Role: Web designer

Repository: <https://github.com/Thuyen21/EW>

Screen Cast: <http://www.youtube.com/watch?v=wBGnxqTTZpU>

Public Site: <http://crt.somee.com/>

Backlog:

- <https://docs.google.com/spreadsheets/d/1jufthGaESFc8EW44YYNgoAw5PF4Eu8OZur5H6tMhUDY/edit?usp=sharing>

User account:

- Admin: admin@gmail.com Password: adminadmin
- Marketing Manager: TuanMM1@gmail.com Password: TuanMM1@gmail.com
- Marketing Coordinator: cje.madlife2000@gmail.com Password: cje.madlife2000@gmail.com
- Student: studentbao@gmail.com Password: studentbao@gmail.com
- Student2: pnb@gmail.com Password: pnb@gmail.com
- Guest: TuanGuest1@gmail.com Password: TuanGuest1@gmail.com

I. Introduction

We formed a group to collaborate on a project. The goal of this project is to create a collection site as well as a place for students to submit papers. The goal was to design a website for the prestigious university. With such a goal in mind, we convene and quickly divide the positions in order to begin project implementation. We need a database designer, an information architect, a programmer, a web designer, a tester, and a scrum specialist before we can spend any money. It is necessary to prepare the necessary knowledge for the project. We must ensure that the website is built to meet the 16 criteria set forth during the website concept and construction process. Divide the implementation phases and implement continuous progress checks to ensure the construction is moving in the right direction, adhering to the requirements, and progressing.

II. Requirement

In order to build and create websites, we have to adhere to the requirements for which the criteria are given and they must meet the system. Below are the criteria given. From the given criteria, which one has we classified as the Functional Requirement and which one is the Non-functional Request.

- The University has a Marketing Manager to oversee the process.
- All Faculties have a Marketing Coordinator who is responsible for managing the process for their Faculty.
- All students have the opportunity to submit one or more articles as Word documents to the magazine.
- All students can also upload high quality images, e.g. photographs.
- All new contributions are disabled after a closure date for new entries, but updates can continue to be done until a final closure date.
- All students must agree to Terms and Conditions before they can submit.
- Once a contribution is submitted the system emails a notification to the Faculty's Marketing Coordinator, who must make a comment within 14 days.
- A Marketing Coordinator can only access contributions by students in their Faculty.
- Each Marketing Coordinator needs to be able to interact with the students in their Faculty in order to edit the contributions and to select those for publication.
- The University Marketing Manager can view all the selected contributions but cannot edit any. They need to be able to download all the selected contributions after the final closure date in a ZIP file for transfer out of the system.
- An administrator maintains any system data, e.g. closure dates for each academic year.
- A guest account for each Faculty can be used to view the selected reports.
- Statistical analysis (e.g. number of contributions per Faculty) needs to be available.
- The interface must be suitable for all devices (e.g. mobile phones, tablets, desktops).

ID	Type	Reason why I decide it belongs to functional or non-functional requirement
1	Functional	Because this requirement belongs to the technical, decentralized and functional part of the system
2	Functional	Because the part's functional decentralization, technical
3	Functional	Because of the requirements and performance of the system requirements, user interaction, description of the function that needs to be performed.

4	Functional	Because of the quality properties of the system. Standards specified by the system
5	Functional	Because it belongs to the functional and technical part of the system
6	Functional	Because it belongs to the functional and technical part of the system
7	Functional	Because belonging to user interaction, capture the behavior to perform the required task
8	Functional	Because of the technical part, the decentralized function
9	Functional	Because of the user interaction, the request to capture data to request performance
10	Functional	Because it's technical, first use decentralization so that the director can perform his own function. The second is the requirement to submit the correct form
11	Functional	Because it belongs to the functional requirements of the system
12	Functional	Because belonging to the system's functional requirements, capture and statistic data.
13	Functional	Drugs on reviews, quality attributes
14	Non-Functional	Because of the properties and quality, namely the user interface, tailored to the devices

Table 1: Requirement

1. Functional 1:

It works. In my project I have role Marketing Manager View statistics, view exceptional reports
View all selected submissions from every faculty

Marketing Manager Index

Course Name	Coordinator	Coordinator mail	Date End	Date Final	View Mark	Exceptional	Contributions
Bao pn	P2qkxRm62bYO3fNiMCMUE8JAGKJ3	baopn@gmail.com	13/04/2021 09:20:00 AM	15/04/2021 09:20:00 AM	View	View	View
HaVy1	ja928ne2QzN0pB1lpUlNxugDkXq1	havy2@gmail.com	01/04/2021 04:18:00 PM	01/04/2021 04:18:00 PM	View	View	View
QTKD	ouDVOH5EVsVfzJIYiyIBXairffC2	cje.madlife2000@gmail.com	03/04/2021 06:03:00 PM	10/04/2021 06:03:00 PM	View	View	View

© 2021 - Edu Application by ASP.NET Web hosting by Somee.com

Figure 1: Functional 1

2. Functional 2:

It works. Each faculty have 1 Marketing Coordinator, and Marketing Coordinator have function mark student, view submission and comment student

Coordinator Index

Coordinator	Date end	Date final	Marking
ouDVOH5EVsVfzJIYiyIBXairffC2	03/04/2021 06:03:00 PM	10/04/2021 06:03:00 PM	Mark

© 2021 - Edu Application by ASP.NET Web hosting by Somee.com

Figure 2: Functional 2

3. Functional 3:

It works. Student can submit word files

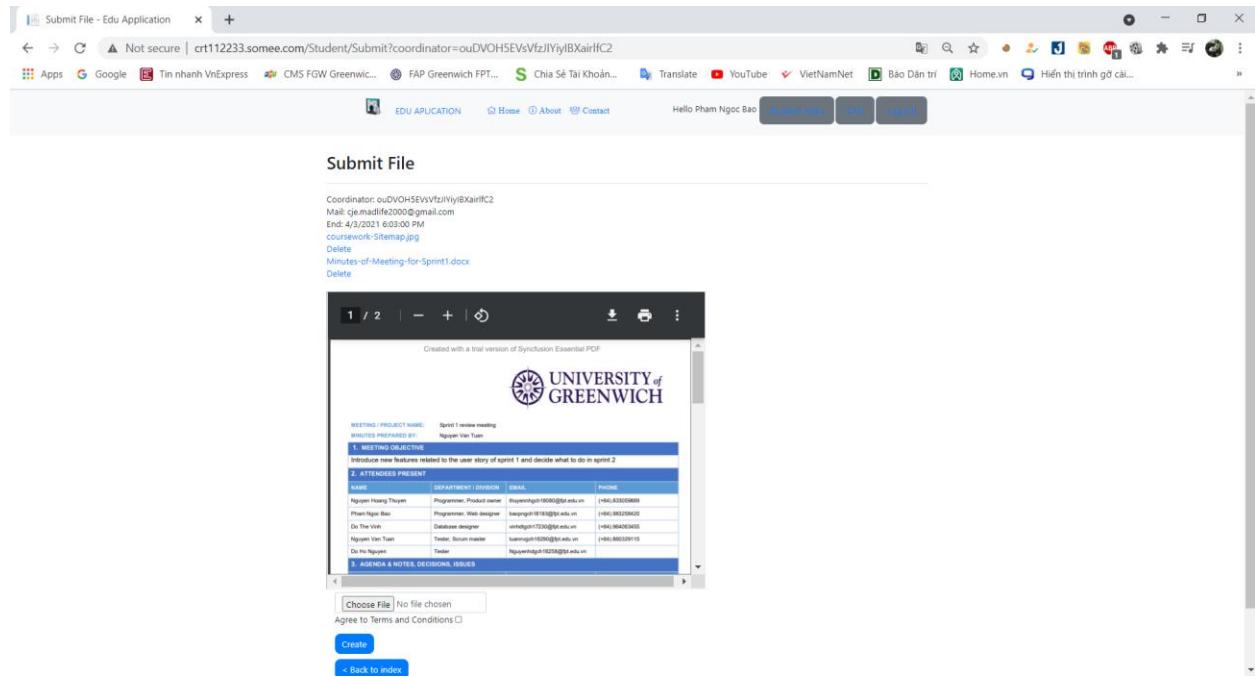


Figure 3: Functional 3

4. Functional 4:

It works. Student can upload images:

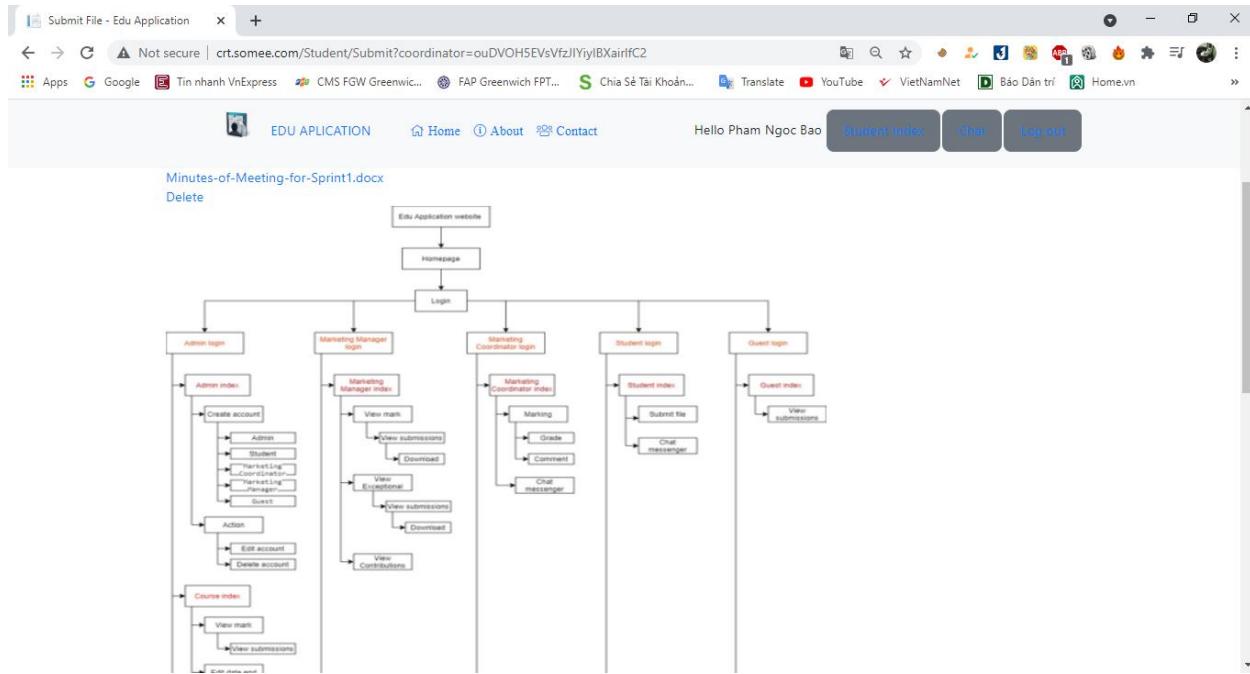


Figure 4: Functional 4

5. Functional 5:

It works. Student can submit before date end and can edit submission before date final

Continue development user interface	Pham Ngoc Bao	Friday, February 26, 2021
Edit database and use case	Do The Vinh	Friday, February 26, 2021
Test the features that are in development	Nguyen Van Tuan	Friday, February 26, 2021
Do Ho Nguyen		
5. NEXT MEETING (if applicable)		
DATE	Saturday, March 13, 2021	TIME 2:30 PM
LOCATION		
OBJECTIVE Introduce new features related to the user story of sprint 2 and decide what to do in sprint 3		

SUBMITTED BY: Nguyen Van Tuan APPROVED BY:

< Back to index

Figure 5: Functional 5

6. Functional 6:

It works. Student must tick in Agree to Terms and Conditions to submit

The screenshot shows a web browser window titled "Submit File - Edu Application". The URL is "crt.somee.com/Student/Submit?coordinator=ouDVOH5EVsVfzJlYrylBXairffC2". The page header includes "EDU APPLICATION", "Home", "About", "Contact", "Hello Pham Ngoc Bao", "Student Index", "Chat", and "Log out". Below the header is a table with two rows:

Topic 3 – Introduce the database	Do The Vinh	10 min
Topic 4 – Introduce the process of testing features	Nguyen Van Tuan, Do Ho Nguyen	10 min

Below the table is a section titled "4. ACTION ITEMS" containing a table:

ACTION	ACTION TO BE TAKEN BY	DATE TO BE ACTIONED BY
Continue development new features	Nguyen Hoang Thuyen	Friday, February 26, 2021
Continue development user interface	Pham Ngoc Bao	Friday, February 26, 2021
Edit database and use case	Do The Vinh	Friday, February 26, 2021
Test the features that are in development	Nguyen Van Tuan	Friday, February 26, 2021
Do Ho Nguyen		

Below this is a section titled "5. NEXT MEETING (if applicable)" with fields for DATE (Saturday, March 13, 2021), TIME (2:30 PM), and LOCATION.

At the bottom of the form, there is a file upload field labeled "Choose File: favicon.ico", a checkbox labeled "Agree to Terms and Conditions" with a tooltip "Please check this box if you want to proceed.", and a "Create" button.

At the very bottom of the page, there is copyright information: "© 2021 - Edu Application by ASP.NET" and "Web hosting by Somee.com".

Figure 6: Functional 6

7. Functional 7:

It works. When their students submit work, the Marketing Coordinator receives mail.

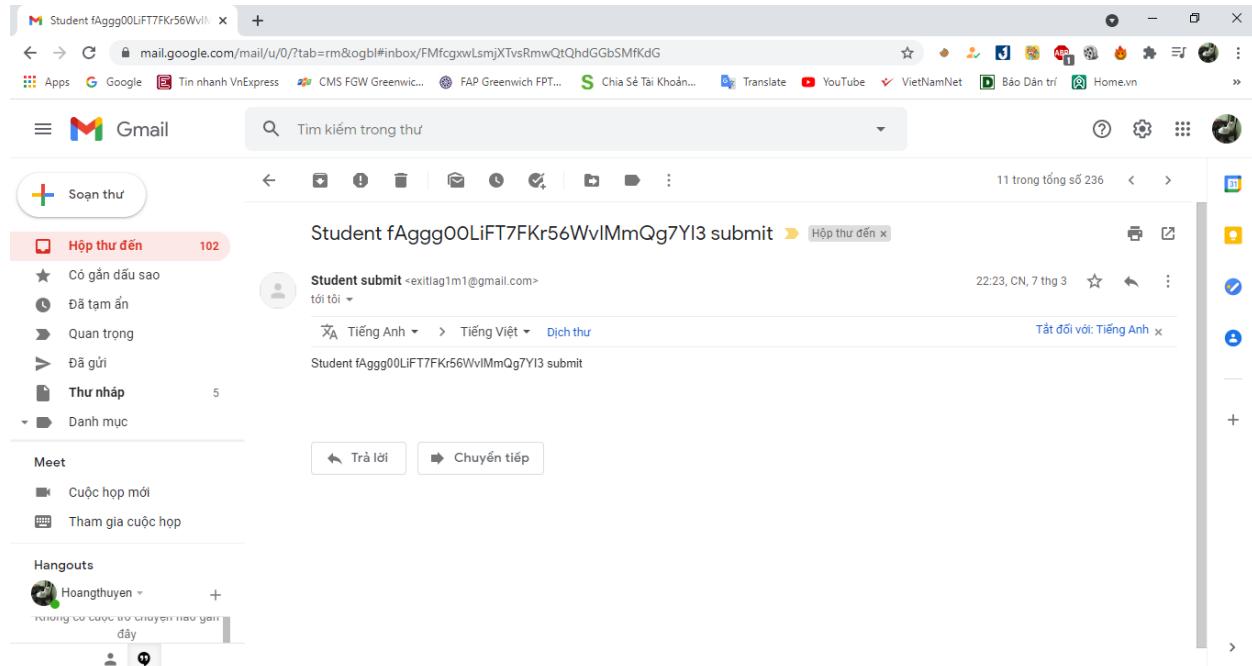


Figure 7: Functional 7

8. Functional 8:

It works, only the Marketing Coordinator can mark, comment and view submission in their faculty

Coordinator	Date end	Date final	Marking
ouDVOH5EVsVfzJlYiyIBXairfC2	03/04/2021 06:03:00 PM	10/04/2021 06:03:00 PM	Mark

Coordinator Index

Coordinator	Date end	Date final	Marking
ouDVOH5EVsVfzJlYiyIBXairfC2	03/04/2021 06:03:00 PM	10/04/2021 06:03:00 PM	Mark

© 2021 - Edu Application by ASP.NET

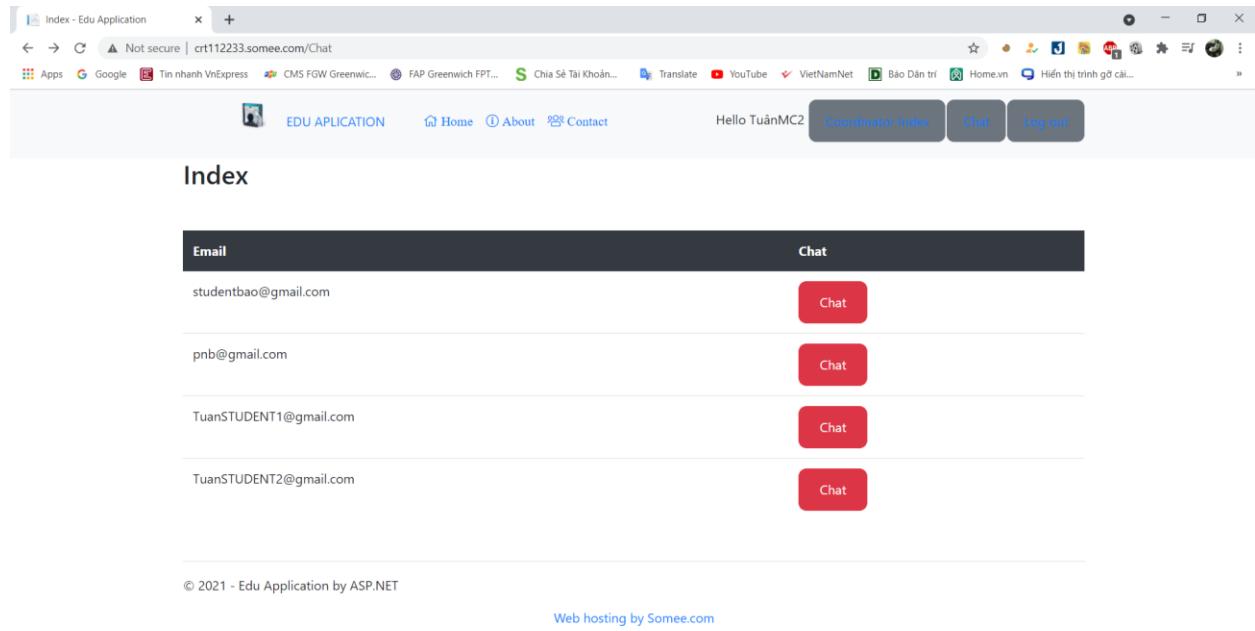
Web hosting by Somee.com

Waiting for crt112233.somee.com...

Figure 8: Functional 8

9. Functional 9:

It works. Marketing Coordinator can chat with their students



The screenshot shows a web browser window titled "Index - Edu Application". The address bar indicates the site is not secure. The page header includes a logo for "EDU APPLICATION", navigation links for "Home", "About", and "Contact", and a user greeting "Hello TuanMC2". On the right, there are buttons for "Coordinator Index", "Chat", and "Log out". The main content area is titled "Index" and contains a table with two columns: "Email" and "Chat". The "Email" column lists four student email addresses: "studentbao@gmail.com", "pnb@gmail.com", "TuanSTUDENT1@gmail.com", and "TuanSTUDENT2@gmail.com". To the right of each email is a red "Chat" button. At the bottom of the page, there is a copyright notice "© 2021 - Edu Application by ASP.NET" and a link to "Web hosting by Somee.com".

Email	Chat
studentbao@gmail.com	<button>Chat</button>
pnb@gmail.com	<button>Chat</button>
TuanSTUDENT1@gmail.com	<button>Chat</button>
TuanSTUDENT2@gmail.com	<button>Chat</button>

Figure 9: Functional 9

10. Functional 10:

It works. Marketing Manager can see all of the selected contributions and can download file zip contributions

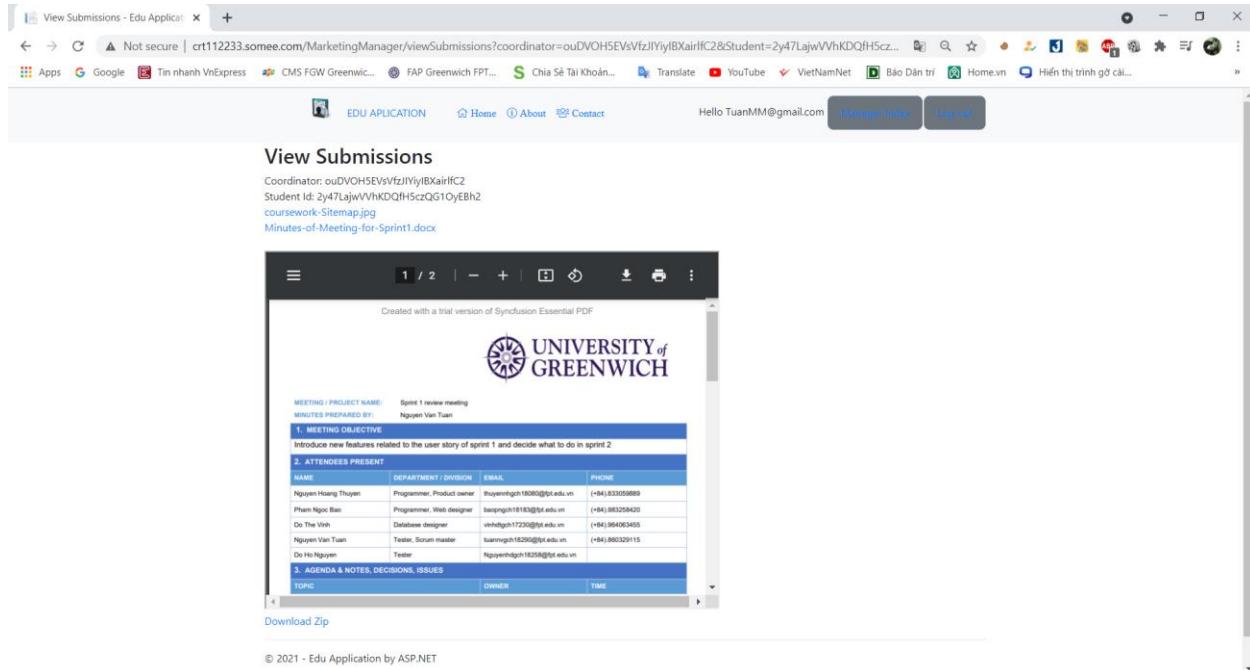


Figure 10: Functional 10

11. Functional 11:

Admin can edit all the account and all the faculty.

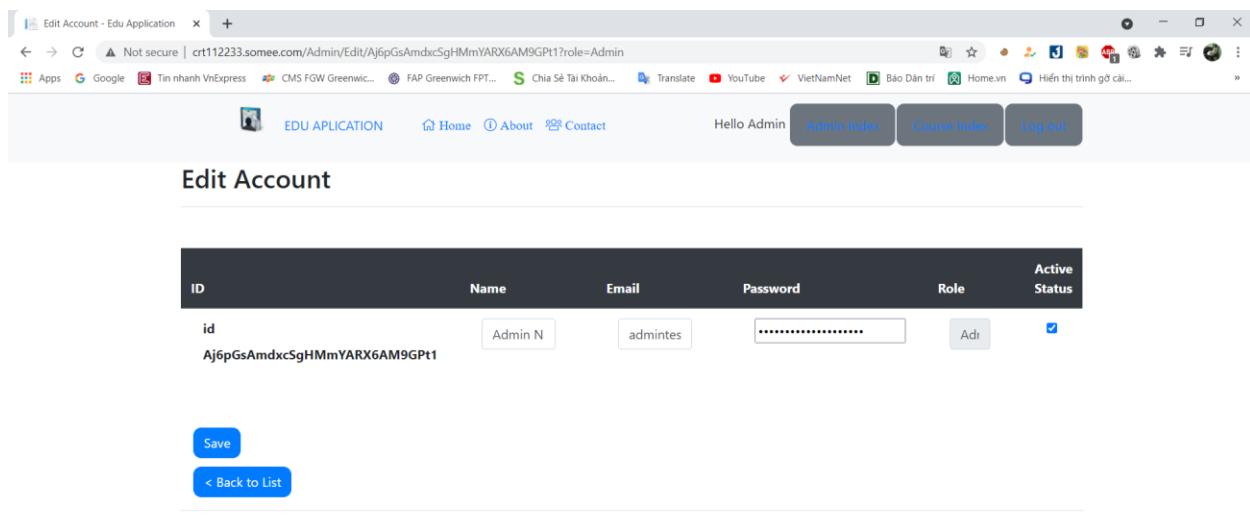


Figure 11: Functional 11.1

The screenshot shows a web browser window titled "Edit Course - Edu Application". The URL is crt12233.somee.com/Admin/EditCourse?coordinator=P2qkxRm62bYO3fNiMCMUE8JAGKJ3. The page has a header with "Hello Admin" and navigation links for "Home Index", "Course Index", and "Log out". Below the header, the title "Edit Course" is displayed, along with a "Back to List" button.

Coordinator	Old date end	Old date Final	nameCourse	New date End	New date Final
P2qkxRm62bYO3fNiMCMUE8JAGKJ3	13/04/2021 09:20:00 AM	15/04/2021 09:20:00 AM	Bac	<input type="text" value="mm/dd/yyyy --:-- --"/>	<input type="text" value="mm/dd/yyyy --:-- --"/>

Save

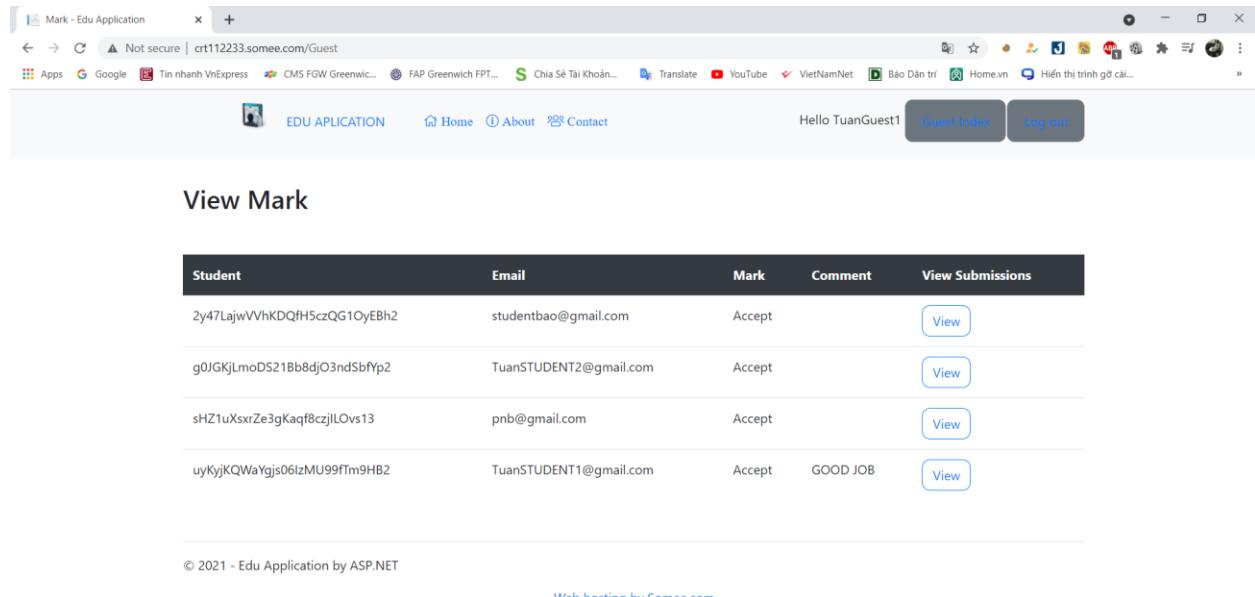
© 2021 - Edu Application by ASP.NET

Web hosting by Somee.com

Figure 12: Functional 11.2

12. Functional 12:

Guest can only view selected report of Faculty which admin assign guest in.



The screenshot shows a web browser window titled "Mark - Edu Application". The URL is "crt112233.somee.com/Guest". The page displays a table titled "View Mark" with columns: Student, Email, Mark, Comment, and View Submissions. There are four rows of data:

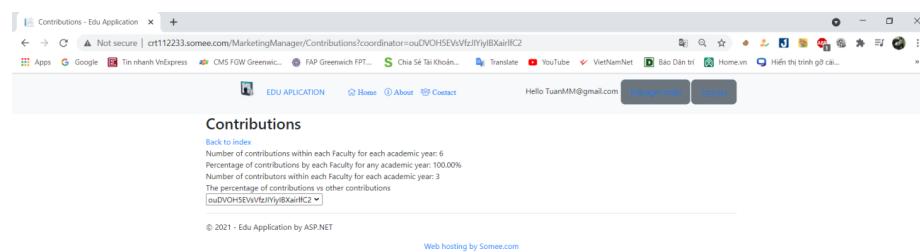
Student	Email	Mark	Comment	View Submissions
2y47LajwVvhKDQfh5czQG1OyEBh2	studentbao@gmail.com	Accept		View
g0JGKjLmoDS21Bb8djO3ndSbfYp2	TuanSTUDENT2@gmail.com	Accept		View
sHZ1uXsxrZe3gKaqf8czjLOvs13	pnb@gmail.com	Accept		View
uyKjKQWaYgjs06izMU99ftm9HB2	TuanSTUDENT1@gmail.com	Accept	GOOD JOB	View

At the bottom of the page, there is a copyright notice: "© 2021 - Edu Application by ASP.NET" and a link: "Web hosting by Somee.com".

Figure 13: Functional 12

13. Functional 13:

Marketing Manager can pick a Faculty to view Statistical analysis



The screenshot shows a web browser window titled "Contributions - Edu Application". The URL is "crt112233.somee.com/MarketingManager/Contributions?coordinator=ouDV0HSEVvIzIYlYlBXairIC2". The page displays a table titled "Contributions" with the following data:

Faculty	Number of contributions	Percentage of contributions
Faculty A	6	100.0%
Faculty B	1	16.67%
Faculty C	3	50.0%
Faculty D	2	33.33%
Faculty E	1	16.67%

At the bottom of the page, there is a copyright notice: "© 2021 - Edu Application by ASP.NET" and a link: "Web hosting by Somee.com".

Figure 14: Functional 13

14. Functional 14:

This web can view in all devices

The screenshot shows a mobile browser interface with the URL crt.somee.com/Admin. The top status bar indicates the time is 20:19, signal strength, battery level at 94%, and a notification for 19 new messages. The main content area is titled "Admin Index". It features a navigation bar with links: Create Account, Admin, Student, Marketing Coordinator, Marketing Manager, and a "Logout" button. Below this is a table with columns: ID, Name, Email, Password, Role, and Action (Delete, Edit). The table lists 19 entries, each corresponding to a user account with their name, email (e.g., admin@gmail.com), role (e.g., Admin, Marketing Coordinator, Marketing Manager, Guest), and options to delete or edit the record. At the bottom of the table, there is a note: "© 2021 - Edu Application by ASP.NET" and "Web hosting by Somee.com".

ID	Name	Email	Password	Role	Action
AjhyGzAmdedSighMmrfARX6AMSGP1	Admin Name after111	admin@test1@gmail.com	*****	Admin	<button>Delete</button> <button>Edit</button>
hnyAkDf5gDM755NU2sWgybicW2	Admin	admin@gmail.com	*****	Admin	<button>Delete</button> <button>Edit</button>
rIQhyt1T7AWdtaWhRJnMd81MCv2	Vinh	admin@gmail.co	*****	Admin	<button>Delete</button> <button>Edit</button>
P2qubRn62bY03NAMCMUEJAGKJ3	Pham Ngoc Bao	tuanbao@gmail.com	*****	Marketing Coordinator	<button>Delete</button> <button>Edit</button>
bIUSZ37jkWFs9kor874fyqeiw2	TuanMC3	TuanMC3@gmail.com	*****	Marketing Coordinator	<button>Delete</button> <button>Edit</button>
je72Bne2QzQ0p0ipnNugDjXc1	Hu Vy2	havy2@gmail.com	*****	Marketing Coordinator	<button>Delete</button> <button>Edit</button>
oxOVOHrEVVftzJYy/BxaifC2	TuanMC2	cje.madifc2000@gmail.com	*****	Marketing Coordinator	<button>Delete</button> <button>Edit</button>
qParKwQr50RbdIBR1SU6Byq1b2	Thuyien Nguyen Hoang	MarketingCoordinator@gmail.com	*****	Marketing Coordinator	<button>Delete</button> <button>Edit</button>
24EDF5fwCTXYtp0jCAoekrJwM33	TuanMM1	TuanMM1@gmail.com	*****	Marketing Manager	<button>Delete</button> <button>Edit</button>
HomtbRdzXakc7JoiTrNgpuv9e93	Ha Vy 4	havy4@gmail.com	*****	Marketing Manager	<button>Delete</button> <button>Edit</button>
PMSGOlyFraJg4YZ8Uhknu7Vgu1	Pham Ngoc Bao	phamngocba@gmail.com	*****	Marketing Manager	<button>Delete</button> <button>Edit</button>
SSEIGfjBrDyaSADhzNNNNV7hsh1	TuanMM@gmail.com	TuanMM@gmail.com	*****	Marketing Manager	<button>Delete</button> <button>Edit</button>
XR54eqp05aPx3qZ2bYH9tpZz62	Do The Vinh	jy.shn@gmail.com	*****	Marketing Manager	<button>Delete</button> <button>Edit</button>
1TDix1clc5kYPkID7cb4D7qWY2	TuanGuest1	TuanGuest1@gmail.com	*****	Guest	<button>Delete</button> <button>Edit</button>
Lrfu0PvBDY1G9BhnCOBuLPh1	Thuyien Nguyen Hoang	guest@gmail.com	*****	Guest	<button>Delete</button> <button>Edit</button>
bUli2lMRWMD6ipQ5L3uprVgE92	Ha Vy	havy@gmail.com	*****	Guest	<button>Delete</button> <button>Edit</button>
moUuwJTMd4YrWmkod3qz0U1	TuanGuest3	TuanGuest3@gmail.com	*****	Guest	<button>Delete</button> <button>Edit</button>
Zy47LajwVVhKDGPHsczGG1OyEBz2	Pham Ngoc Bao	studentbao@gmail.com	*****	Student	<button>Delete</button> <button>Edit</button>
Kaf5imC76OnyGikXn72Pz2YeaLg2	TuanSTUDENT3	TuanSTUDENT3@gmail.com	*****	Student	<button>Delete</button> <button>Edit</button>
g0J0KJLnoD521BsdqO3ndStyYp2	TuanSTUDENT2	TuanSTUDENT2@gmail.com	*****	Student	<button>Delete</button> <button>Edit</button>
ihaeLvgwfld4InhMa20XzEx03	TuanSTUDENT4	TuanSTUDENT4@gmail.com	*****	Student	<button>Delete</button> <button>Edit</button>
shZtuXseZelgKaqfbczjL0vs13	Pham Ngoc Bao	phnb@gmail.com	*****	Student	<button>Delete</button> <button>Edit</button>
uyKyKQWaYgs06izMu99fTrn9HB2	Tuan	TuanSTUDENT1@gmail.com	*****	Student	<button>Delete</button> <button>Edit</button>

Figure 15: Functional 14

III. Database

1. Introduction

This report will provide users with an overview of the system and usage purposes. This report covers ERDs, use case diagrams, an introduction to firebase databases used in projects and data security, data validation, followed by understanding the system and clarifying how operation of the system. Analyzing the network journal system of the University of Greenwich, introducing and learning about the use case diagram to help users better understand how the system works and the tasks of each actor, giving advantages and disadvantages and Things to improve in the system

Why choose a firebase database for the project:

- firebase helps you build and run successful apps
- Supported by Google and popular with companies and users by convenience
- Easily use and integrate Firebase with open technology tools like google marketing platform, data studio
- Trusted used for high-security safety.

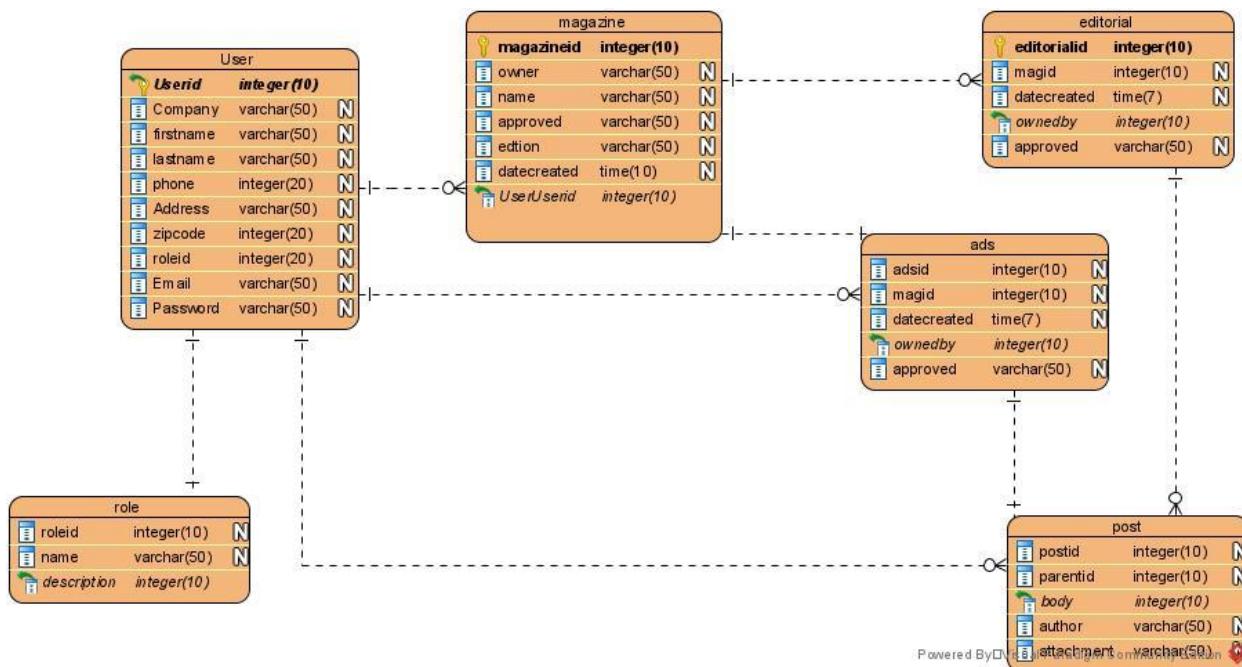


Figure 16: ERD of system

Collecting user criteria, the database was created into 6 tables, for each table followed standardized rules to minimize duplication. First look at the visible entities that are magazine and user, then we can access the ads and editorial, so there are 4 entities in total. From there I will add the role for the user and post for the ads as it is an important part of the way the system works. In general, a simple ERD will have 6 entities: user, magazine, ads, editorial, role, and post with each of the properties below.

2. Physical design:

	Approved	Varchar (50)		
Post	Postid Parented Body Author attachment	Integer (20) Varchar (50) Varchar (50) Varchar (50) Varchar (50)	PK	User table

Table 2: Physical design

First, for students, it is possible to upload high-quality images, read comments for their submissions, update contributions until the final closing date.

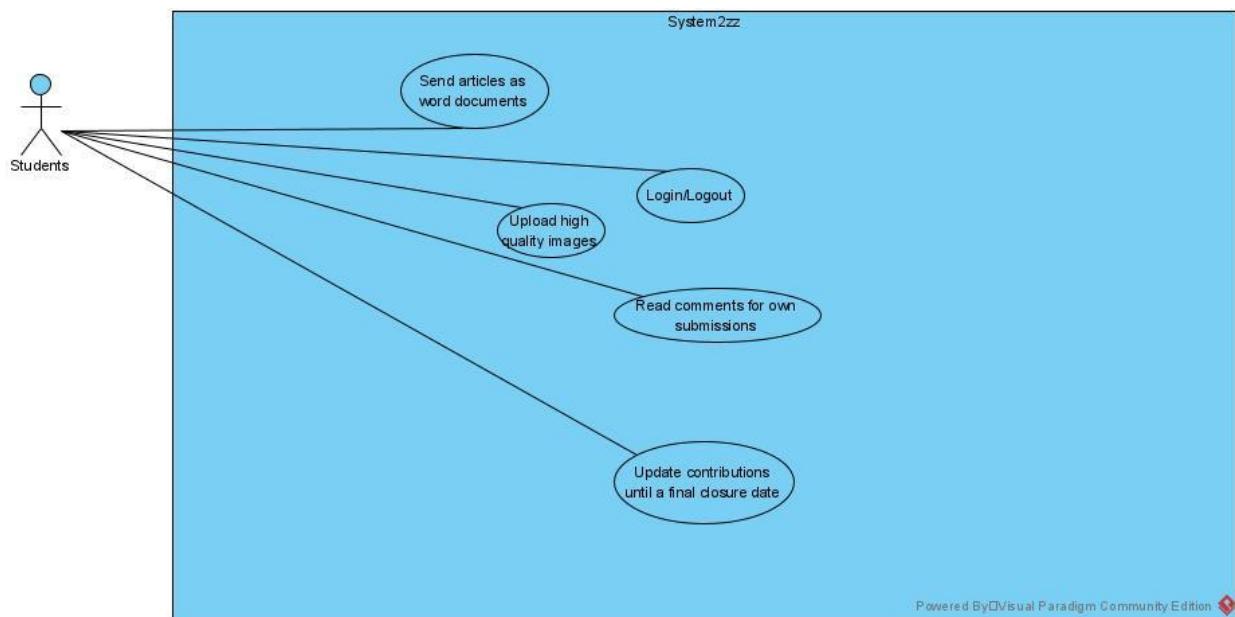


Figure 17: use case Student of system

- For the marketing director that can view all selected contributions, special reports can be viewed, all selected contributions can be downloaded as a zip file, track contribution

count, plus contributors, and percentage contributions of many faculties.

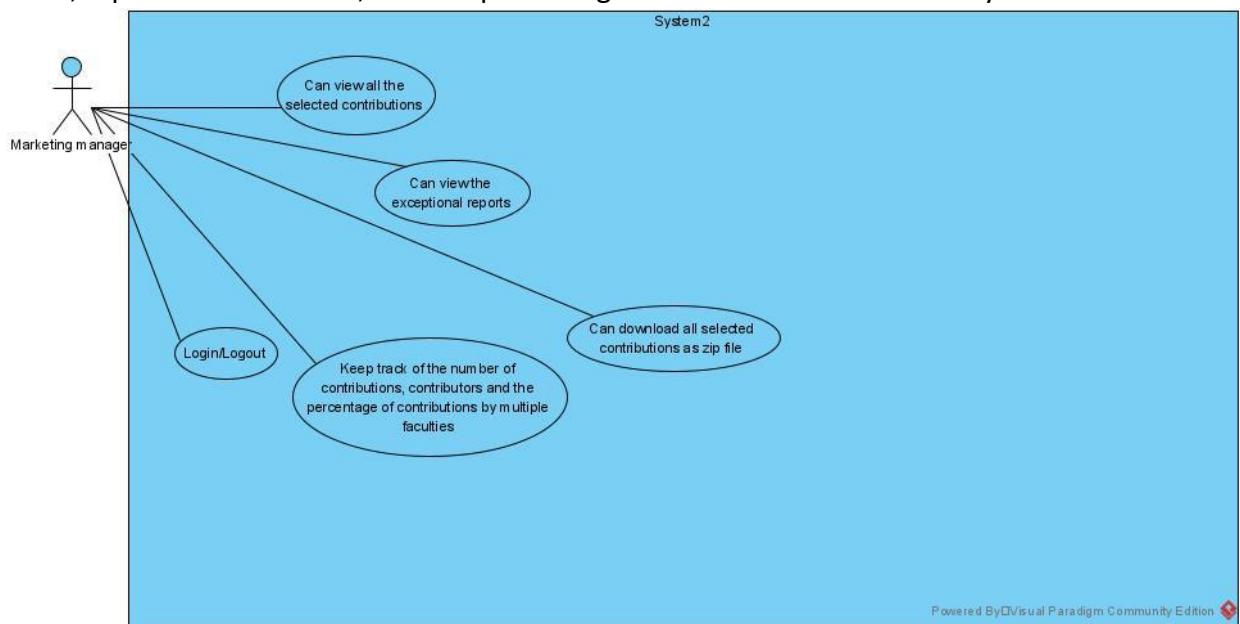


Figure 18: use case Marketing manager of system

- The marketing coordinator can log in, log out of accounts on the system, receive email notifications from the system, make comments for students' posts, within 14 days after the system stops accepting new posts, view contributions from students in my faculty, interact with students in my faculty, select contributions to publish.

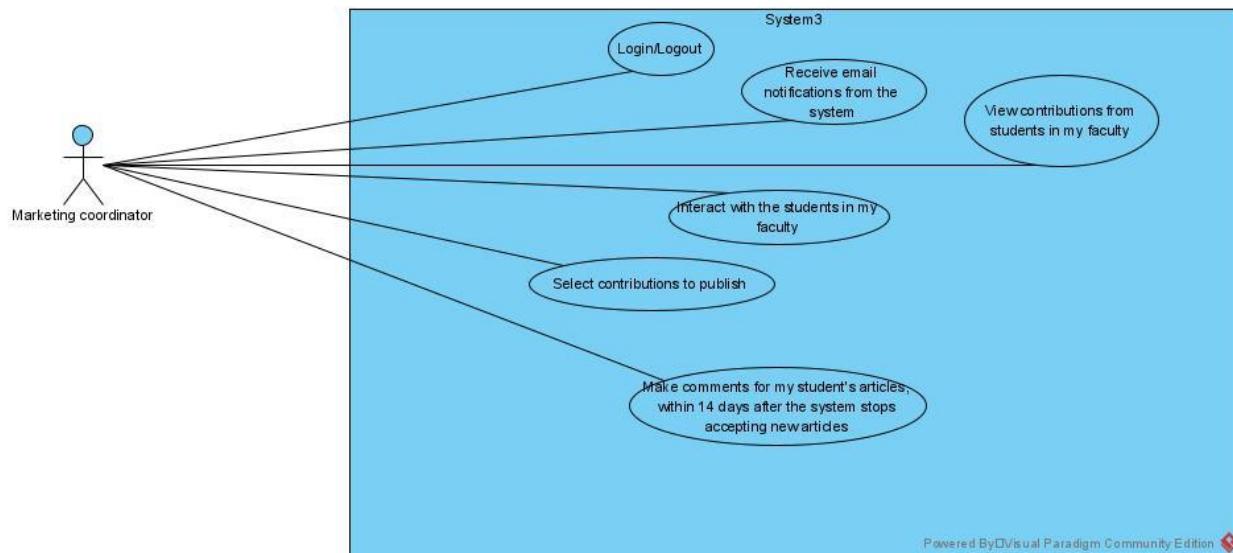


Figure 19: use case Marketing coordinator of system

- For administrators can maintain all system data, control update time, close time, create new accounts and control login permissions, create approval terms and conditions before students submit their articles.

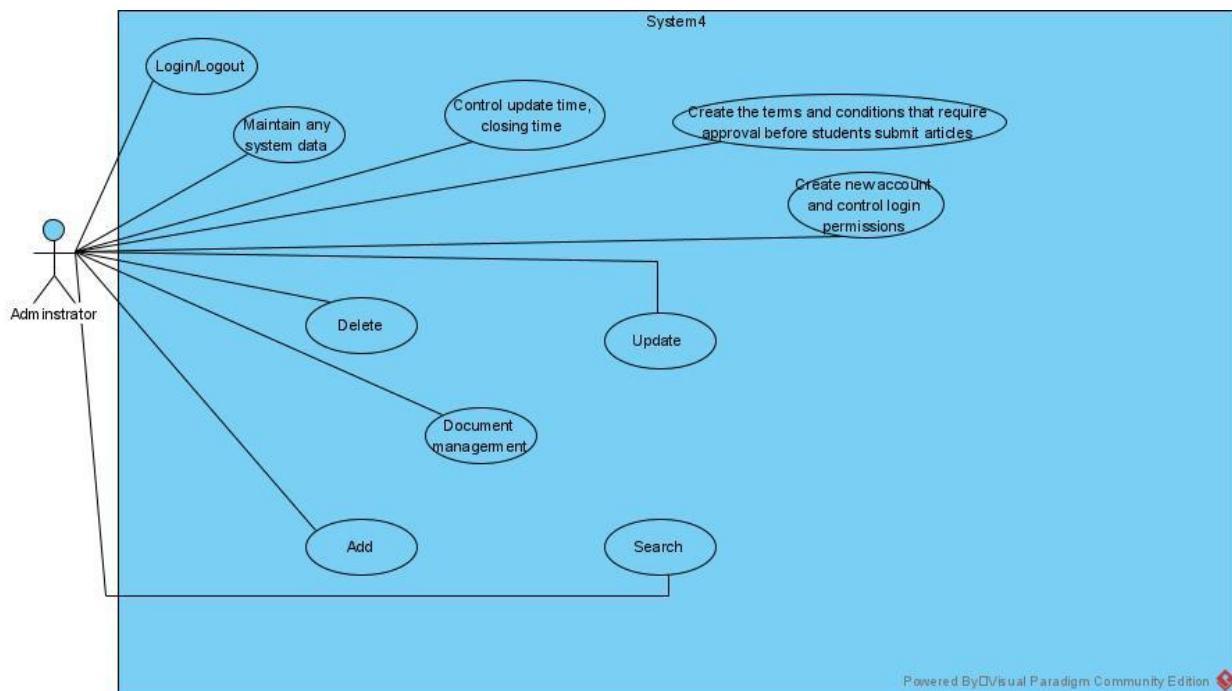


Figure 20: use case Admin of system

- Guests can log in, log out of accounts on the system, view the reports selected for each faculty.

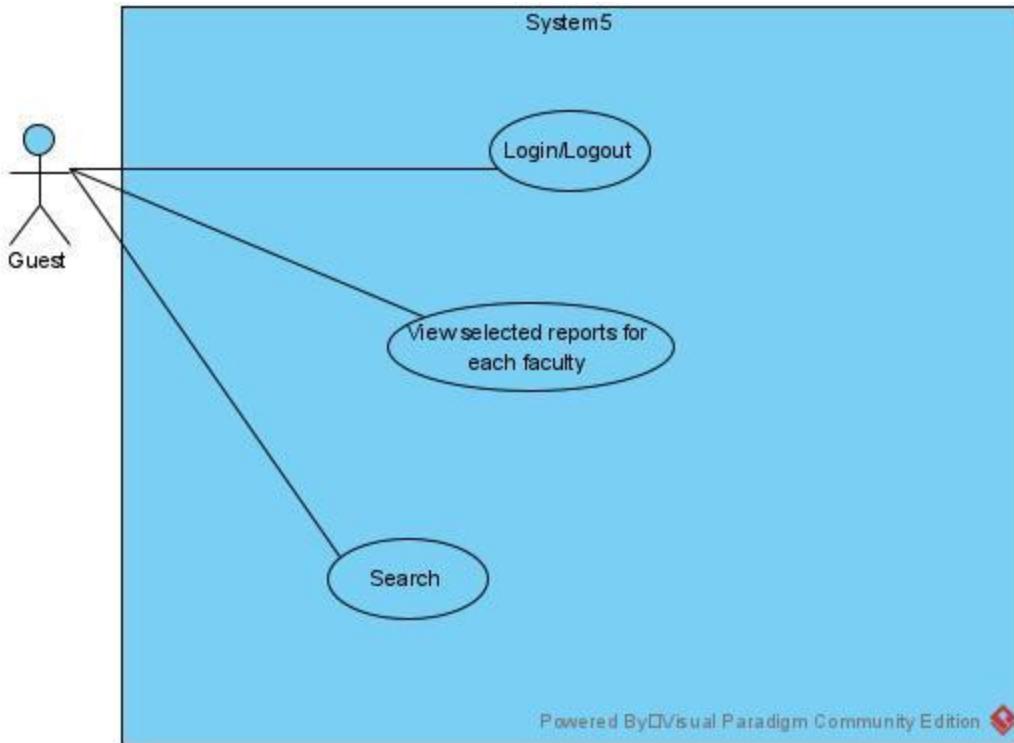


Figure 21: use case Guest of system

- What is data security:
- Data security is the process of protecting data from unauthorized access and theft throughout its entire life cycle. Similar to the project we deploy to have encrypted information and data. For example, when the user creates an account and logs in the system, the password will be encrypted in is fire base authentication to avoid status steal passwords from users.

```

try
{

    if (ModelState.IsValid)
    {
        var auth = new FirebaseAuthProvider(new FirebaseConfig(ApiKey));
        var ab = await auth.SignInWithEmailAndPasswordAsync(model.Email, model.Password);
        string token = ab.FirebaseToken;

        var user = ab.User;

        //if(ab.User.EmailVerified == false)
        //{
        //    ModelState.AddModelError(string.Empty, "Please Verify your email then login Plz.");
        //    return View(model);
        //}
        if (token == null)
        {
            ModelState.AddModelError("error", "Invalid username or password.");
            return View(model);
        }
    }
}

```

Figure 22: data security code

- What is data validation:
- Data validation is checking the accuracy of the source data before using, importing, or otherwise using the data. To be more precise, data validation is a way of data cleansing. For example, the system developed by this project for users when registering an account to log into the system, when the user enters the password must have enough from 8 to 16 characters including letters and numbers, Registered user account must be @gmail.com.



The screenshot shows a code editor with validation messages for three input fields: Name, Email, and Password. The validation messages are generated by the `Html.ValidationMessageFor` method, which adds a red text-danger class to the label and a red border to the input field if the validation fails.

```

<hr />


@Html.ValidationMessageFor(model => model.Name, "", new { @class = "text-danger" })
    <label for="name" class="form-label">Full name</label>
    <input type="text" id="name" class="form-input zoom" placeholder="Ex: John Doe" required name="name" style="border-radius:10px; padding:10px">



@Html.ValidationMessageFor(model => model.Email, "", new { @class = "text-danger" })
    <label for="email" class="form-label">Email</label>
    <input type="email" id="email" class="form-input zoom" placeholder="Ex: johndoe@email.com" required name="email" style="border-radius:10px; padding:10px">



@Html.ValidationMessageFor(model => model.Password, "", new { @class = "text-danger" })
    <label for="password" class="form-label">Password</label>
    <input type="password" id="password" class="form-input zoom" placeholder="*****" required name="password" style="border-radius:10px; padding:10px">


```

```

if (token == null)
{
    ModelState.AddModelError("error", "Invalid username or password.");
    return View(model);
}
var claims = new List<Claim>();
client = new FireSharp.FirebaseClient(config);

```

Figure 23: data validation code

The screenshot shows the Firebase Authentication console for the project 'aspdata'. The left sidebar includes 'Project Overview', 'Build' (Authentication, Firestore Database, Realtime Database, Storage, Hosting, Functions, Machine Learning), 'Release and monitor' (Crashlytics, Performance, Test Lab), and 'Extensions'. The main 'Authentication' tab is selected, showing a table of users:

Identifier	Providers	Created	Signed in	User UID
studentbao@gmail.com	✉️	30 Mar 2021	30 Mar 2021	2j47LajwVVhKD0fh5czQG1OyEBh2
tuanmc1@gmail.com	✉️	31 Mar 2021	31 Mar 2021	5JUeYSPMAaTcSLlwRFEdtGcSWd2
baopn@gmail.com	✉️	30 Mar 2021	30 Mar 2021	P2qkxRm62bY03fNlMCMUE8JAG...
pnbmktmng@gmail.com	✉️	30 Mar 2021	30 Mar 2021	PIMSG0tyFraJg4YZBUhKruYJVpu1
tuanstudent1@gmail.com	✉️	31 Mar 2021	31 Mar 2021	fMtuswJZjabFRew9JJdVzY4JSGj1
admin@gmail.com	✉️	24 Mar 2021	31 Mar 2021	hnyAkDffgDM7S5NUj2isWGybicIW2
1630202020@gmail.com	✉️	26 Mar 2021	30 Mar 2021	ouDVOH5EVsVfzJIYiyIBXairfc2

Figure 24: fire base authentication database online

- The system contains accounts, passwords, including students, guests, admin, and marketing management.

The screenshot shows the Firebase Realtime Database console for the project 'aspdata'. The left sidebar includes 'Project Overview', 'Build' (Authentication, Firestore Database, Realtime Database, Storage, Hosting, Functions, Machine Learning), 'Release and monitor' (Crashlytics, Performance, Test Lab), and 'Extensions'. The main 'Realtime Database' tab is selected, showing a hierarchical tree structure of data nodes under 'aspdata-8d746-default-rtdb':

```

aspdata-8d746-default-rtdb
├── Account
├── Chat
├── Comment
├── Course
├── Exceptional
├── Link
├── Mark
└── Notifications
  
```

Figure 25: fire base real-time database online

- It is a place for all data of the system such as accounts, comments, messages, courses, documents, etc.

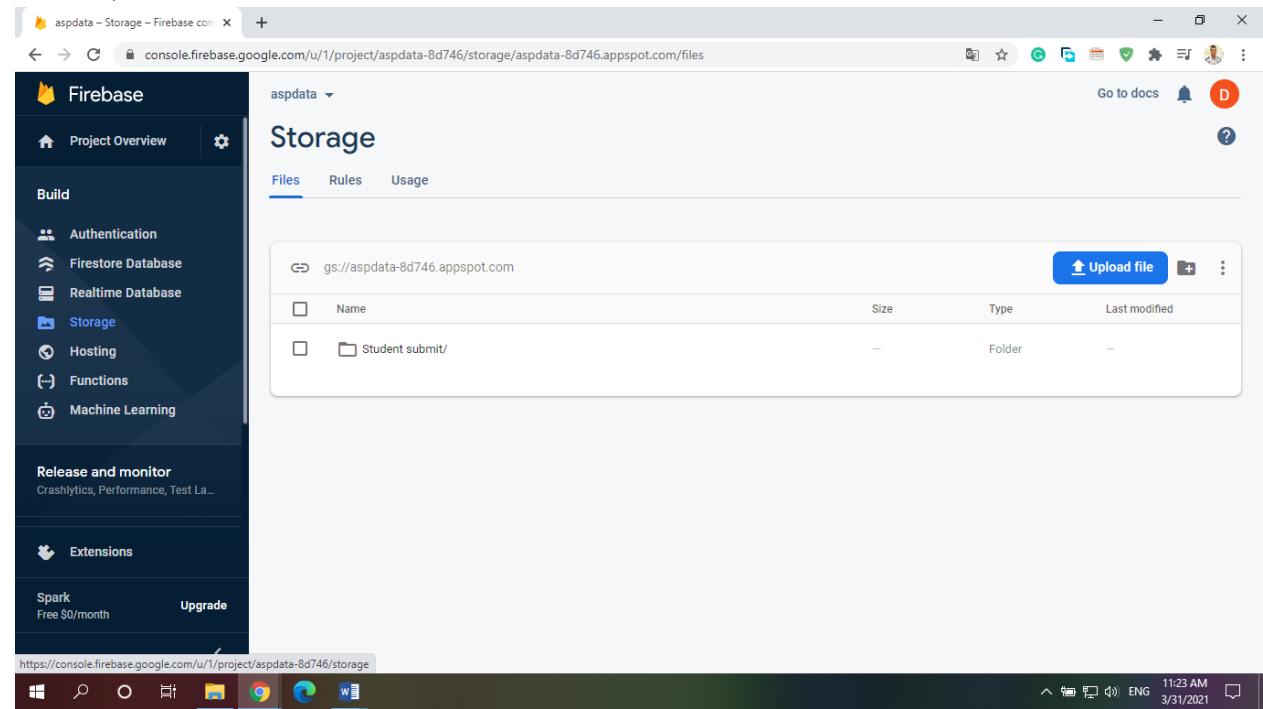


Figure 26: fire base storage database online

A place to store documents submitted by students such as photos, documents, reports in zip format.

IV. Design

1. Sitemap

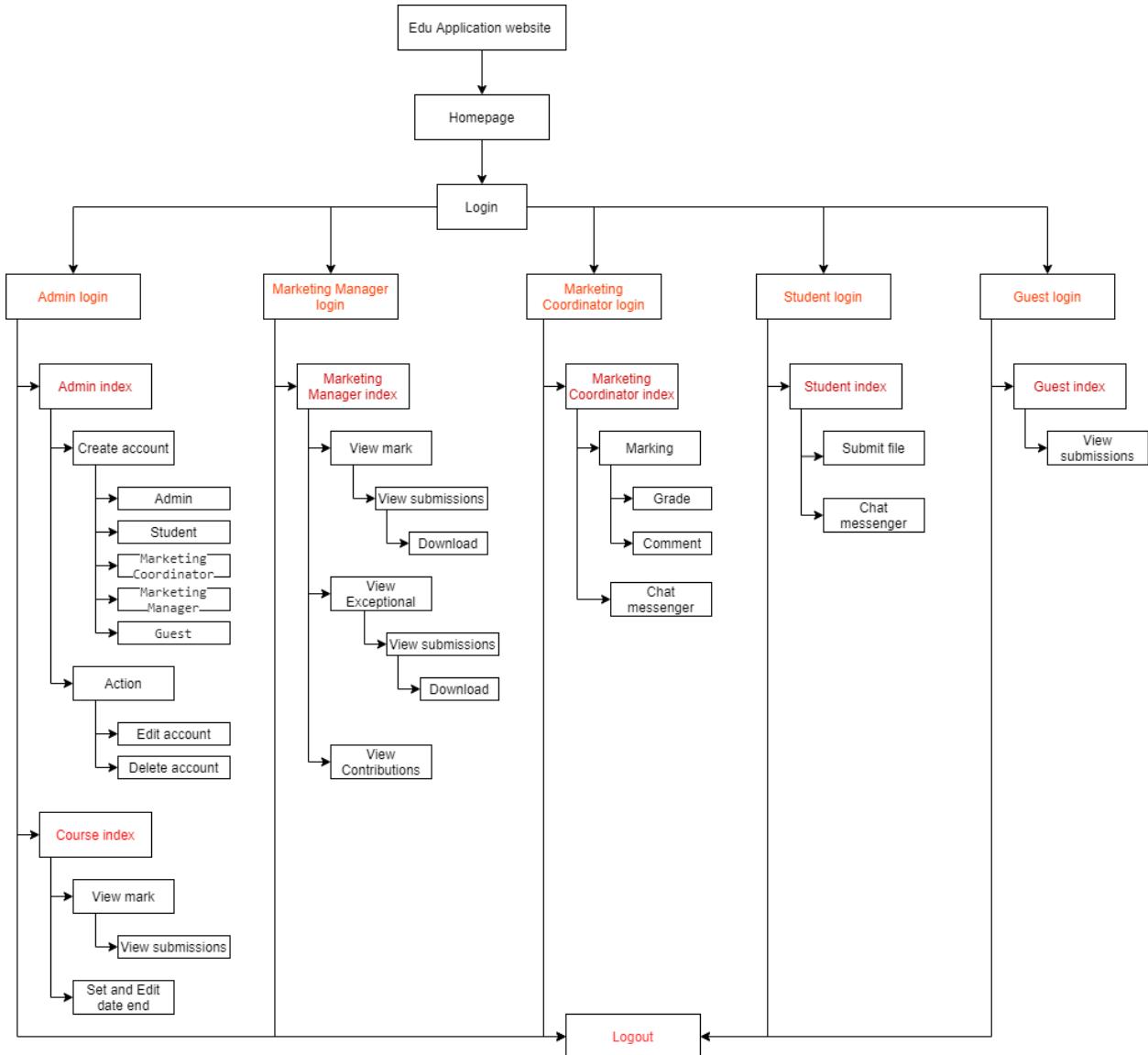


Figure 27: Sitemap

2. Wireframes

a. Home page

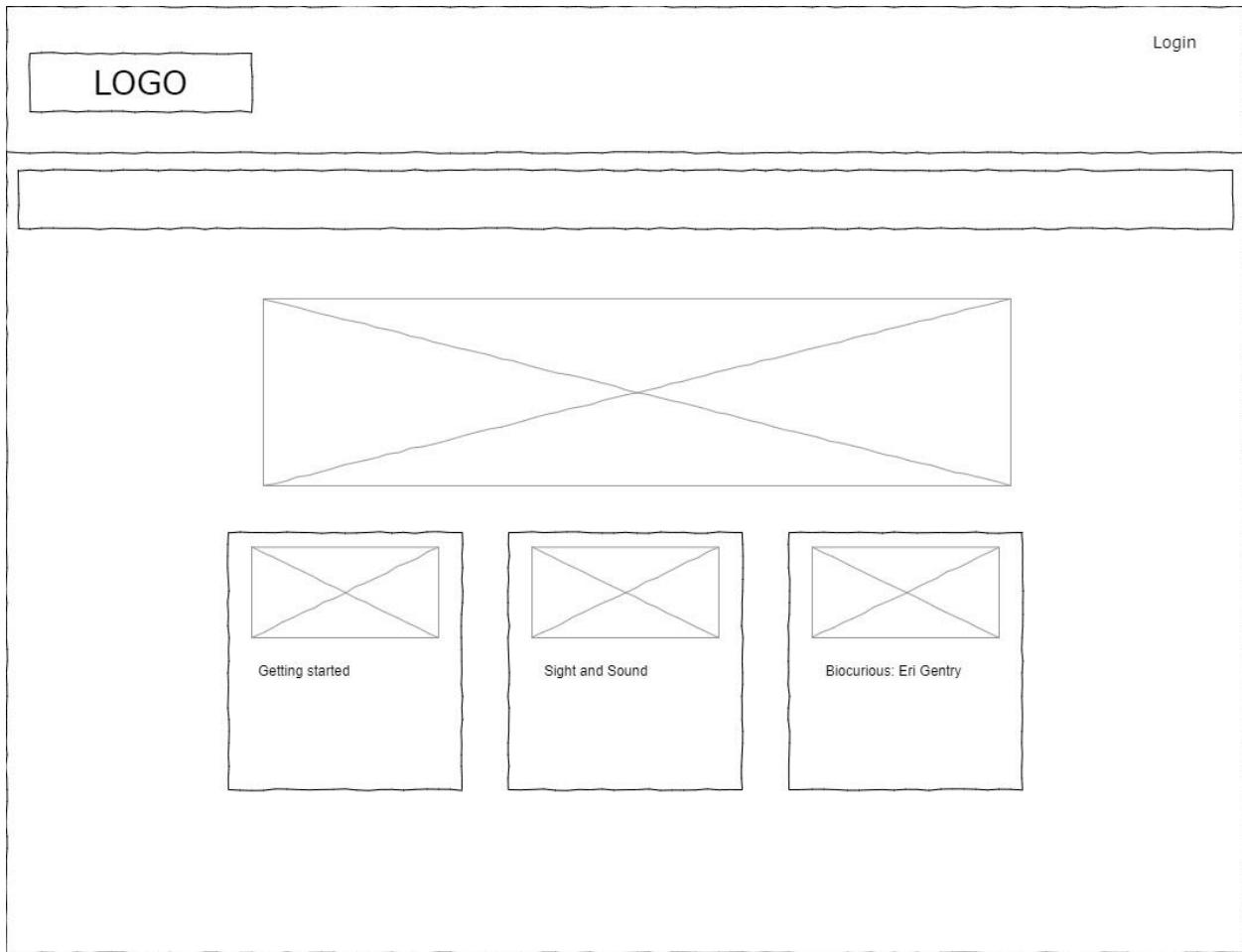


Figure 28: Wireframes homepage

b. Login page

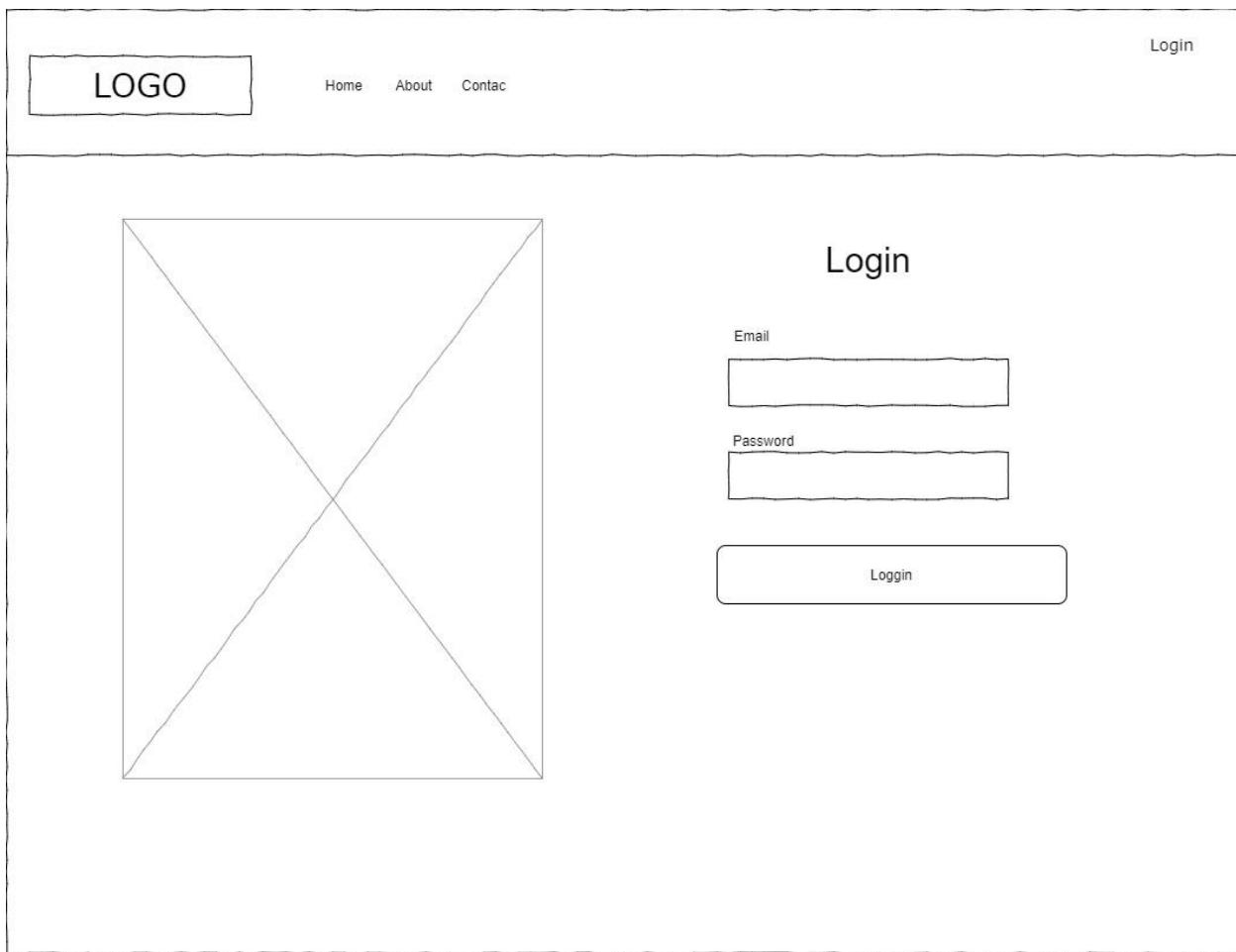


Figure 29: Wireframes Login page

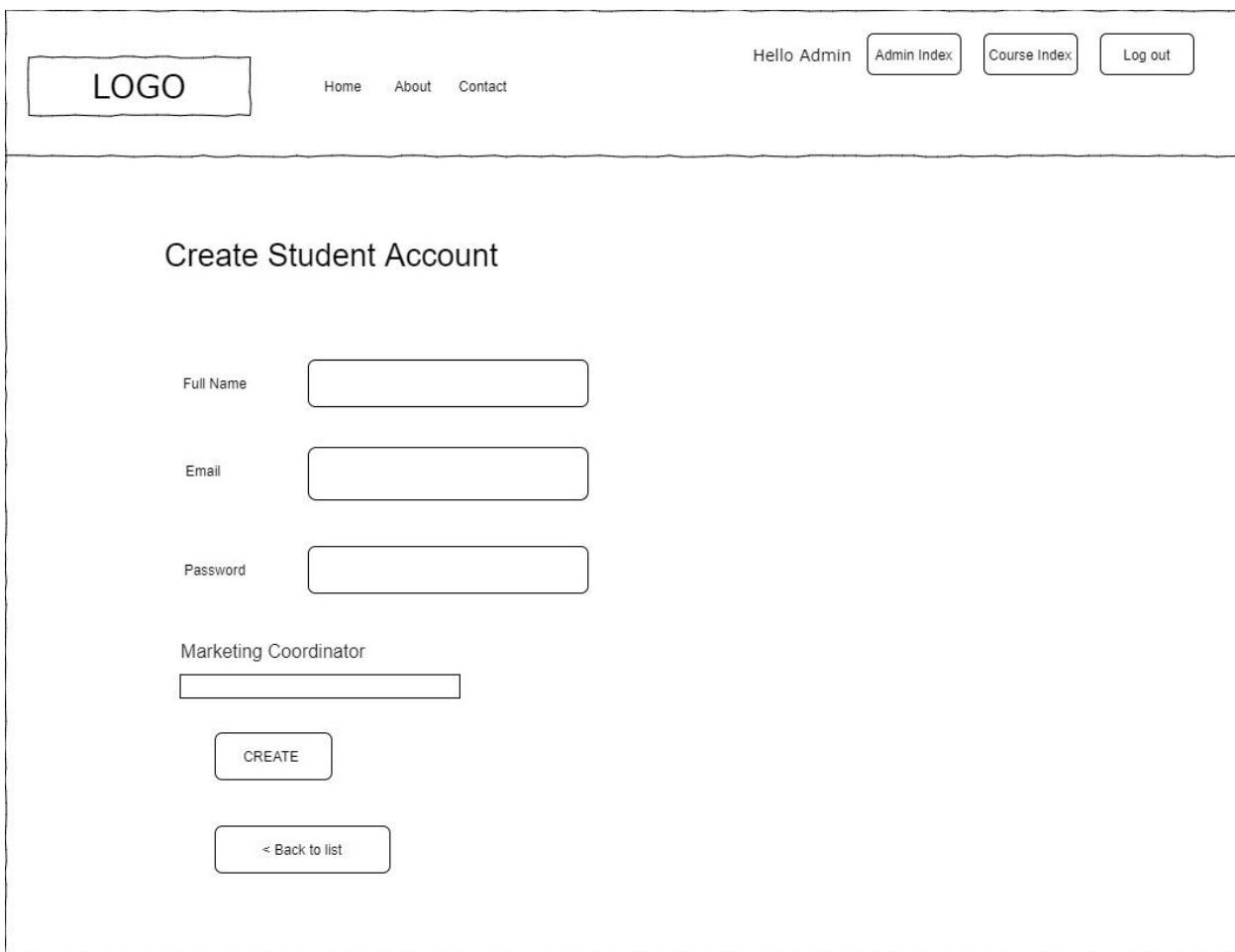
c. Admin

Admin Index

The wireframe shows the layout of the Admin Index page. At the top left is a placeholder for a logo. To its right is a horizontal navigation bar with links: Home, About, Contact, Hello Admin, Admin Index (which is highlighted), Course Index, and Log out. Below this is the main content area titled "Admin Index". On the left side of the content area, there is a "Create account:" label followed by five buttons: Admin, Student, Marketing Coordinator, Marketing Manager, and Guest. Below these buttons is a table with a single row and six columns, labeled ID, Name, Email, Password, Role, and Action. The entire central content area is enclosed in a large rectangular frame, which contains a large "X" drawn from corner to corner, indicating that this section is currently empty or a placeholder.

Figure 30: Wireframes Admin Index

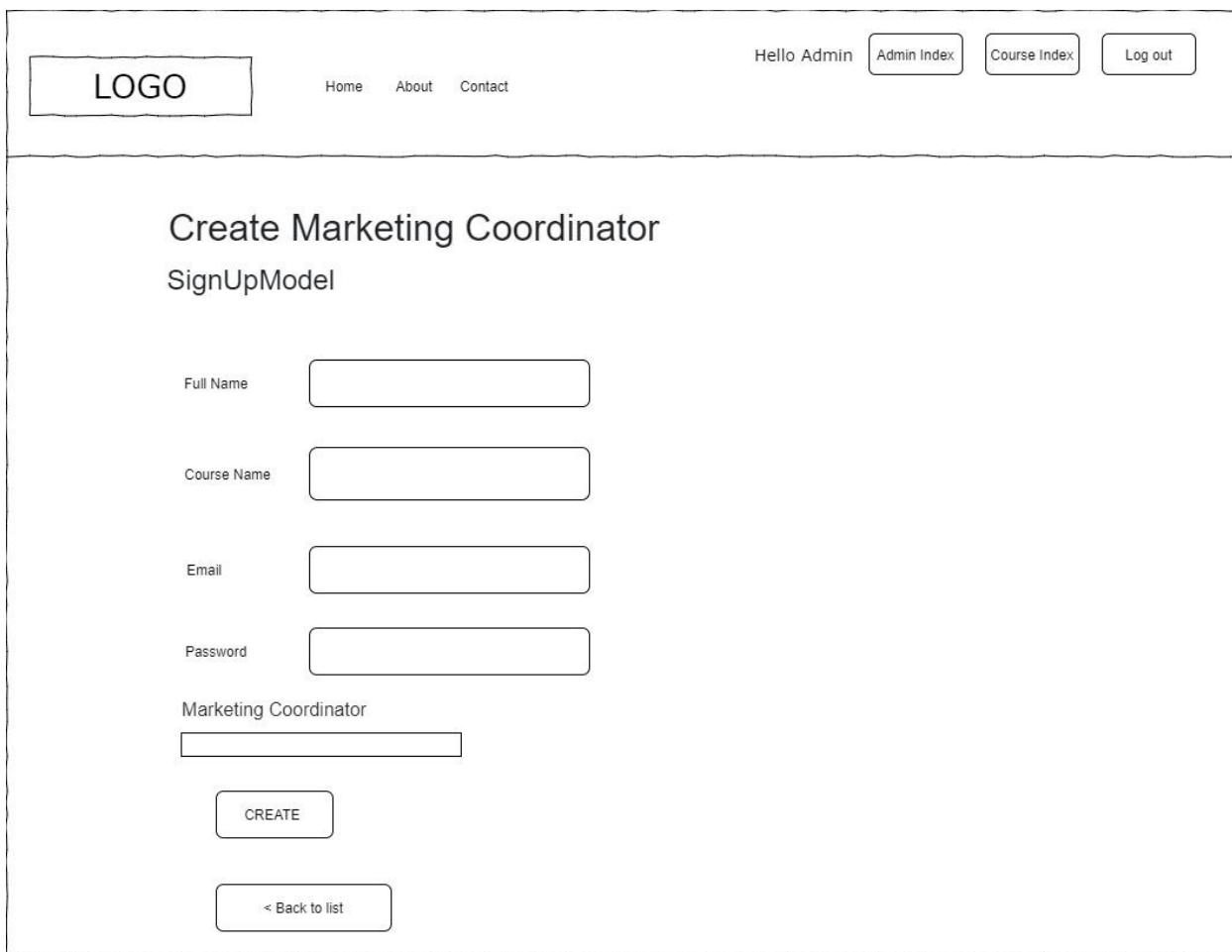
Admin create student account



The wireframe for the 'Admin create student account' page is enclosed in a rectangular frame. At the top left is a placeholder 'LOGO'. To its right is a horizontal navigation bar with links: 'Home', 'About', 'Contact', 'Hello Admin', 'Admin Index', 'Course Index', and 'Log out'. Below this is a large, thin-lined rectangular area. In the center of this area, the title 'Create Student Account' is displayed. To the left of the form fields, labels are aligned vertically: 'Full Name', 'Email', and 'Password', each followed by a corresponding empty rectangular input field. Below these fields is a label 'Marketing Coordinator' followed by a larger empty rectangular input field. At the bottom of the form area are two buttons: a small one labeled 'CREATE' and a larger one labeled '< Back to list'.

Figure 31: Wireframes Admin create Student account

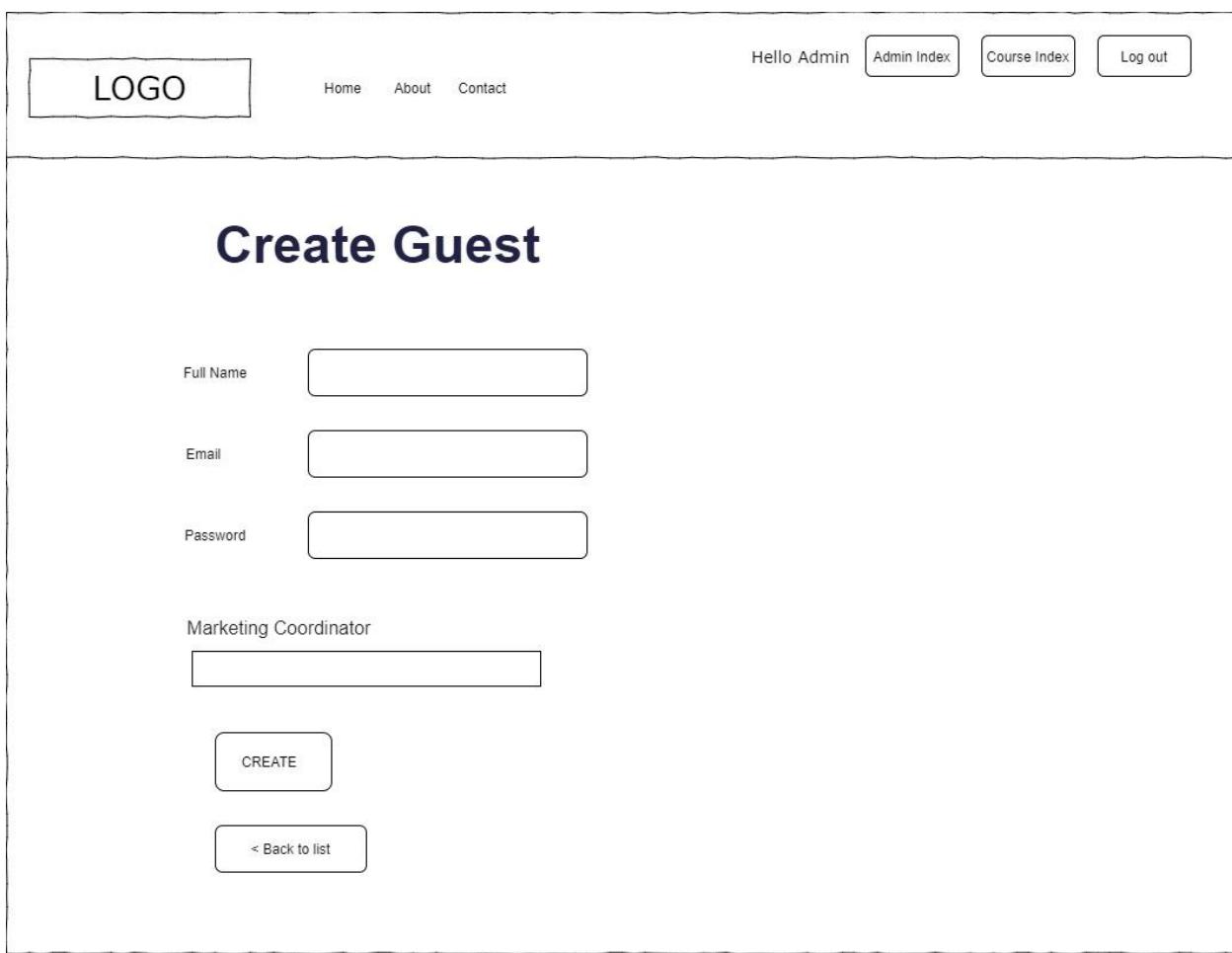
Admin create Marketing Coordinator account



The wireframe shows a user interface for creating a Marketing Coordinator account. At the top left is a placeholder for a logo. To its right are navigation links: Home, About, Contact, Hello Admin, Admin Index, Course Index, and Log out. The main title "Create Marketing Coordinator" is centered above the form fields. Below it, the text "SignUpModel" is displayed. The form consists of four text input fields labeled "Full Name", "Course Name", "Email", and "Password". Below these is a checkbox labeled "Marketing Coordinator". A large rectangular input field follows, and at the bottom is a "CREATE" button. At the very bottom is a link "[< Back to list](#)".

Figure 32: Wireframes Admin create Marketing Coordinator account

Admin create Guest account



The wireframe shows a top navigation bar with a logo placeholder, 'Hello Admin' greeting, and links for 'Admin Index', 'Course Index', and 'Log out'. Below this is a main title 'Create Guest'. The form consists of four input fields for 'Full Name', 'Email', and 'Password', followed by a single-line input for 'Marketing Coordinator'. At the bottom are two buttons: 'CREATE' and '< Back to list'.

LOGO

Hello Admin Admin Index Course Index Log out

Create Guest

Full Name

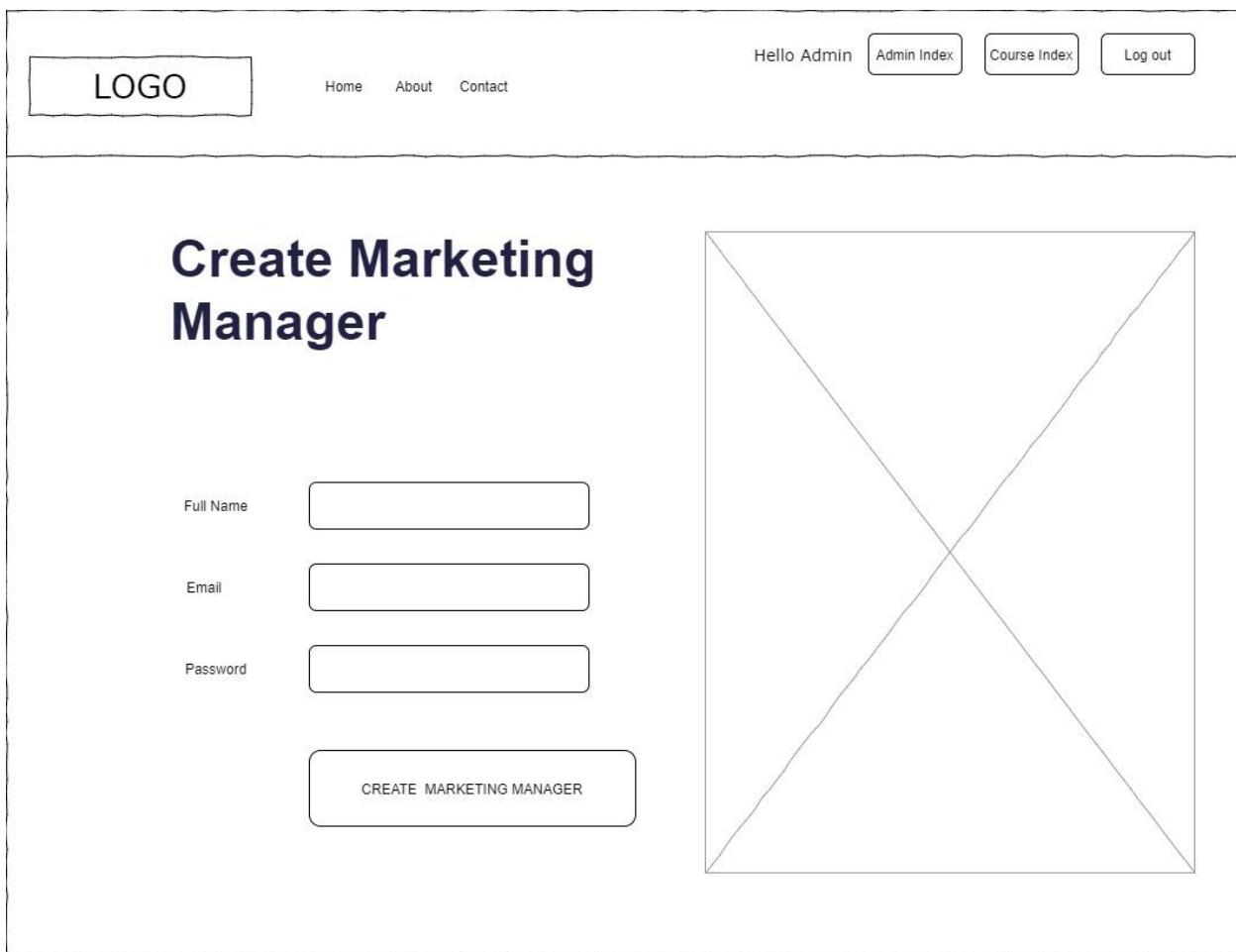
Email

Password

Marketing Coordinator

Figure 33: Wireframes Admin create Guest account

Admin create Marketing Manager account



The wireframe shows a top navigation bar with a logo placeholder, 'Hello Admin' and three buttons ('Admin Index', 'Course Index', 'Log out'). Below this is a large title 'Create Marketing Manager'. To the right is a large rectangular area with a large 'X' drawn through it. On the left, there are four input fields labeled 'Full Name', 'Email', 'Password', and a final button labeled 'CREATE MARKETING MANAGER'.

LOGO

Hello Admin [Admin Index](#) [Course Index](#) [Log out](#)

Home About Contact

Create Marketing Manager

Full Name

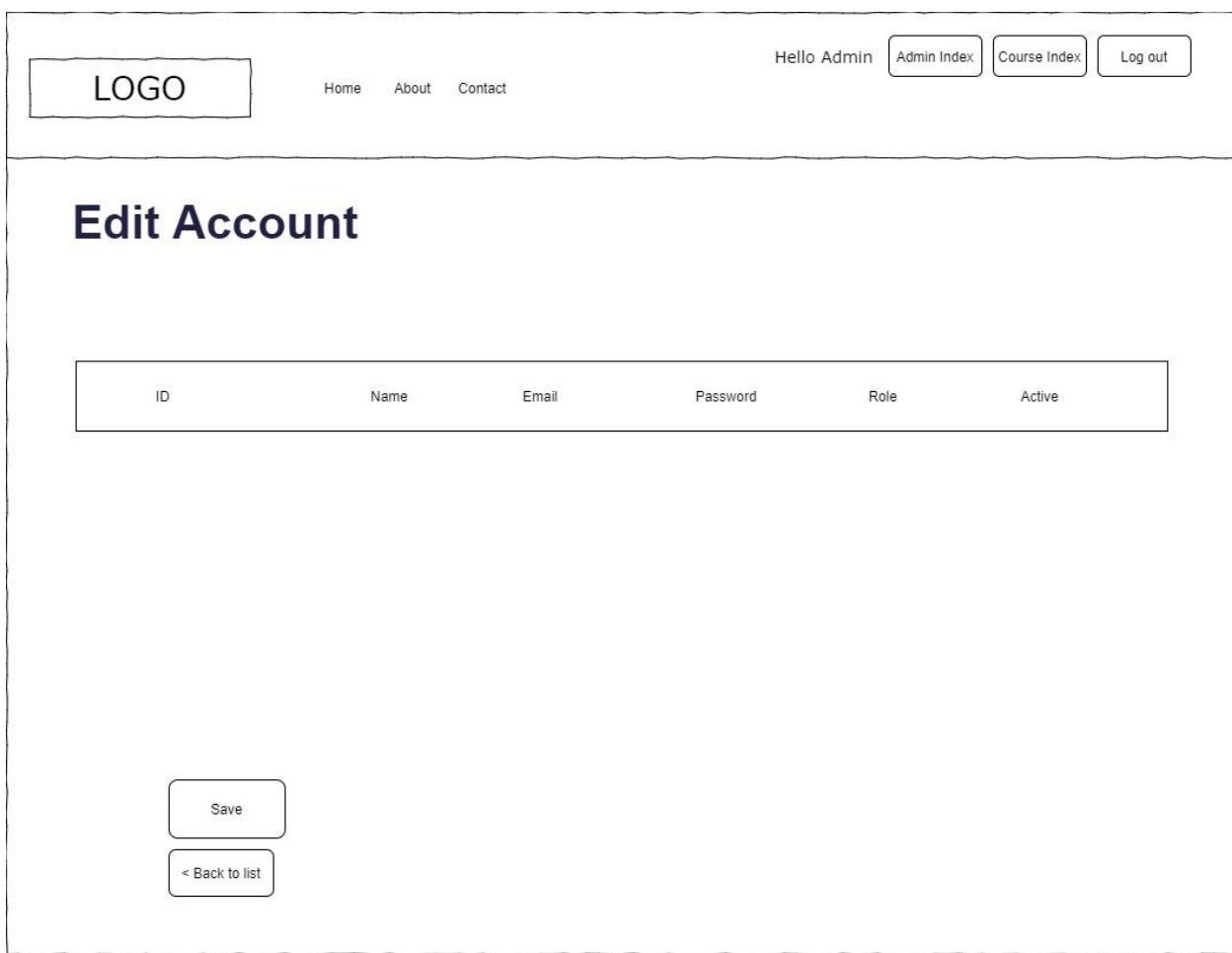
Email

Password

CREATE MARKETING MANAGER

Figure 34: Wireframes Admin create Marketing Manager account

Admin Edit account



The wireframe for the Admin Edit account page is structured as follows:

- Header:** A horizontal navigation bar at the top. It includes a "LOGO" placeholder box on the left, followed by links for "Home", "About", and "Contact". On the right, it shows a greeting "Hello Admin" and three buttons: "Admin Index", "Course Index", and "Log out".
- Title:** A large, bold heading "Edit Account" centered above the form.
- Form:** A rectangular input area containing six fields labeled "ID", "Name", "Email", "Password", "Role", and "Active". These fields are likely represented by input boxes or dropdown menus.
- Buttons:** Two rounded rectangular buttons at the bottom left:
 - A larger button labeled "Save".
 - A smaller button labeled "< Back to list".

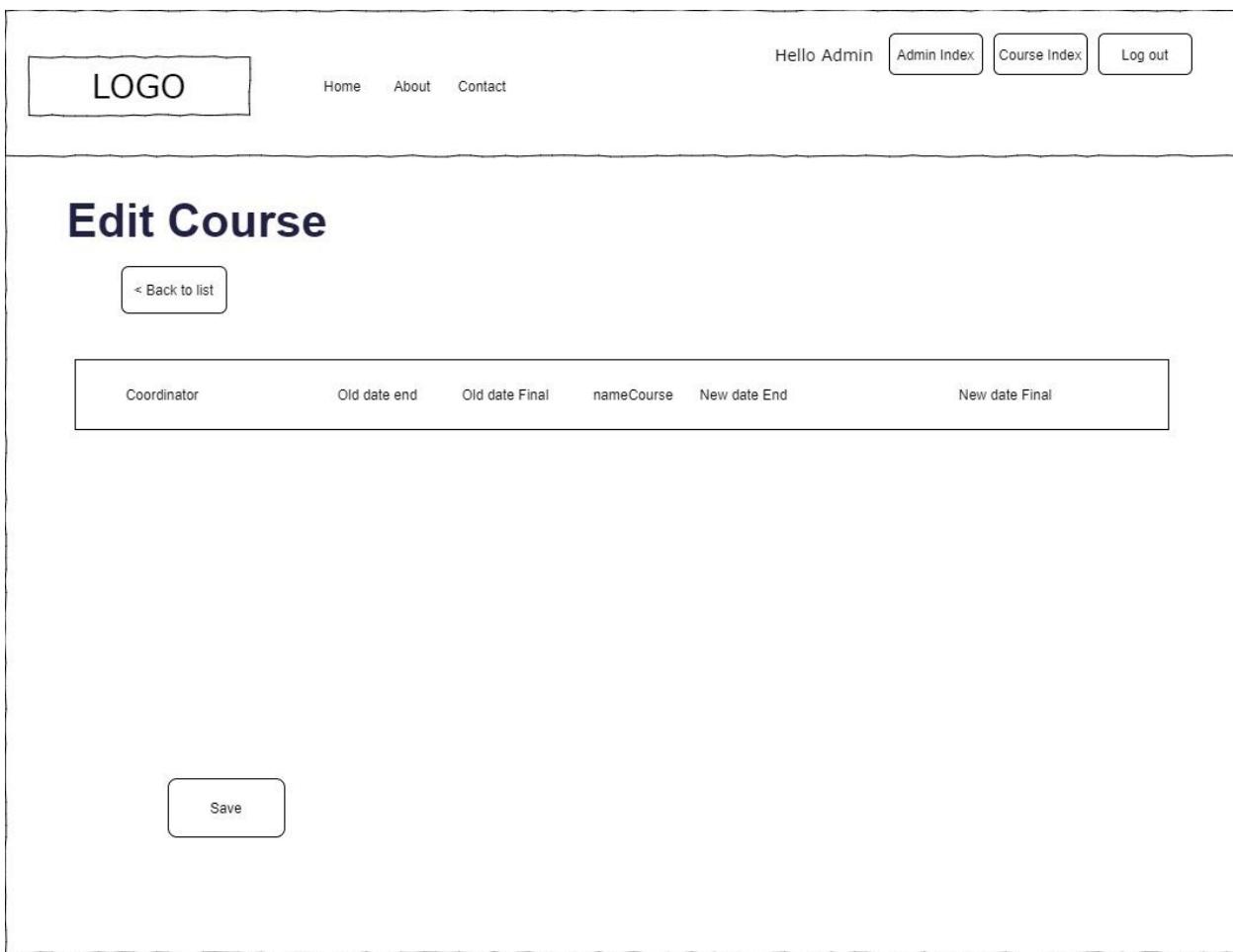
Figure 35: Wireframes Admin Edit account

Admin Index course



Figure 36: Wireframes Admin Index course

Admin Edit course



The wireframe illustrates the layout of the 'Admin Edit course' page. At the top left is a placeholder for a logo. To its right are navigation links: 'Home', 'About', and 'Contact'. On the far right, the text 'Hello Admin' is displayed, followed by three buttons: 'Admin Index', 'Course Index', and 'Log out'. Below this header, the main title 'Edit Course' is centered. Underneath the title is a button labeled '< Back to list'. A horizontal row of input fields follows, containing the labels 'Coordinator', 'Old date end', 'Old date Final', 'nameCourse', 'New date End', and 'New date Final'. At the bottom left of the page is a single button labeled 'Save'.

Figure 37: Wireframes Admin Edit course

d. Marketing Manager

Marketing Manager Index

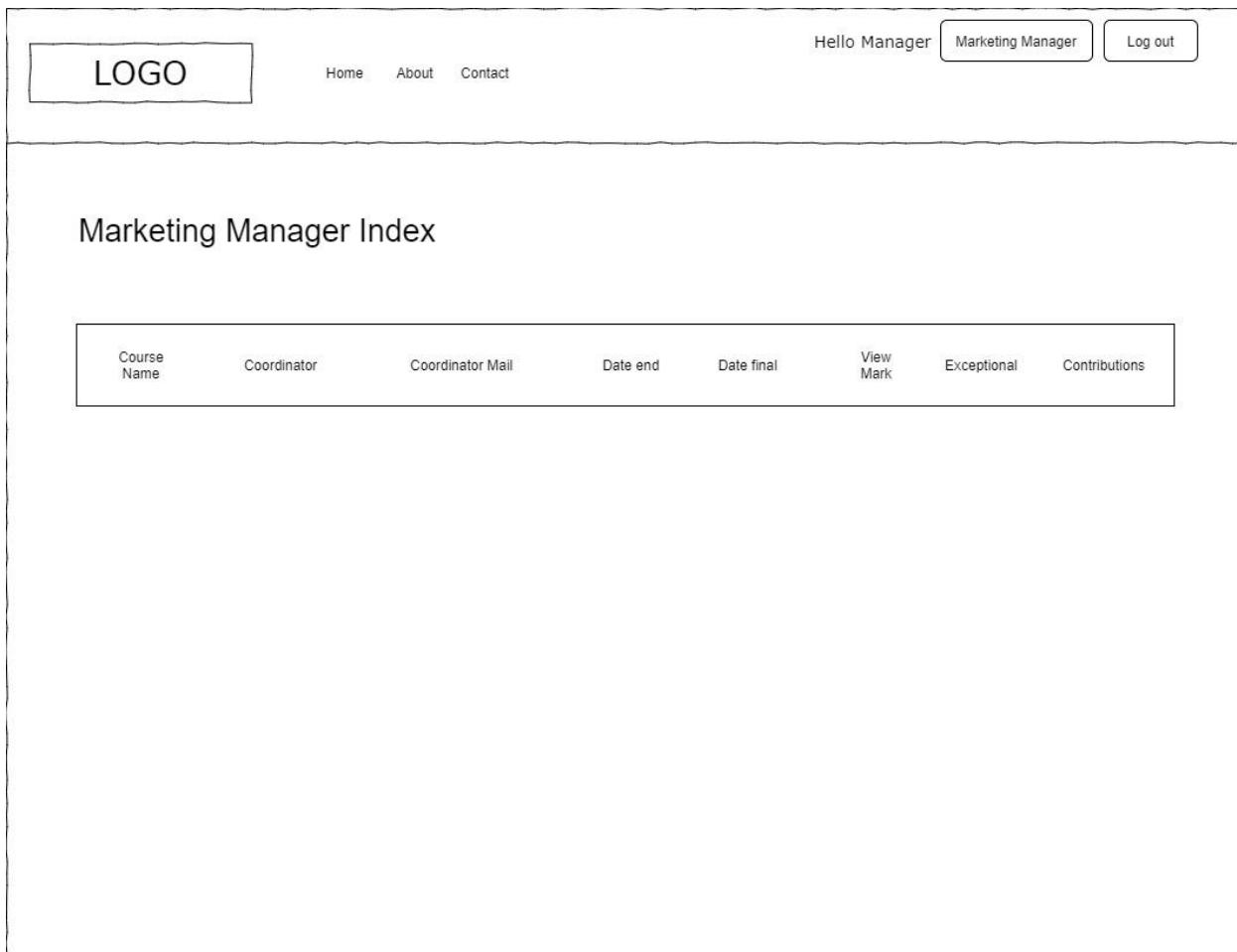


Figure 38: Wireframes Marketing Manager Index

Marketing Manager View Course

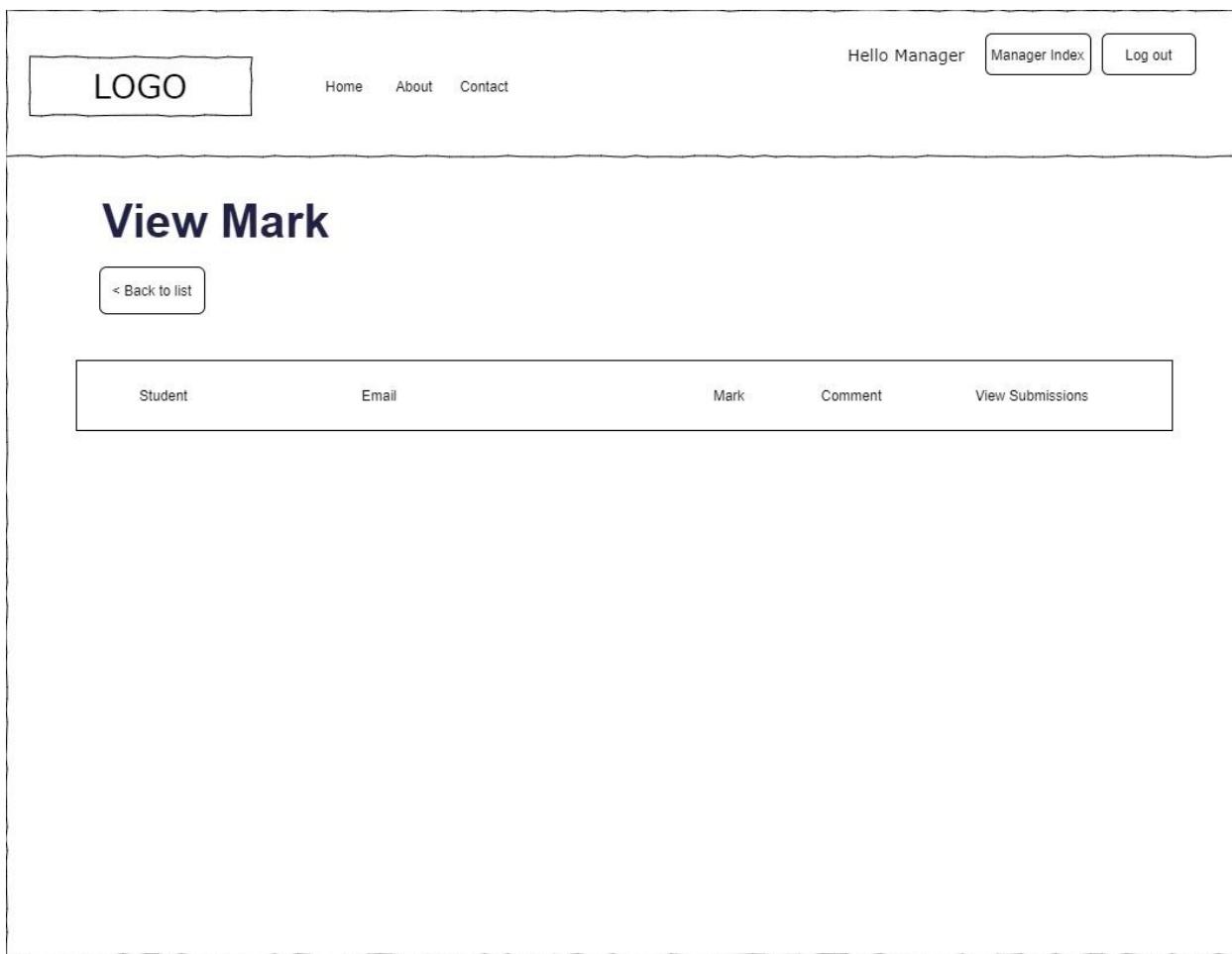


Figure 39: Wireframes Marketing Manager View Course

Marketing Manager View Submissions

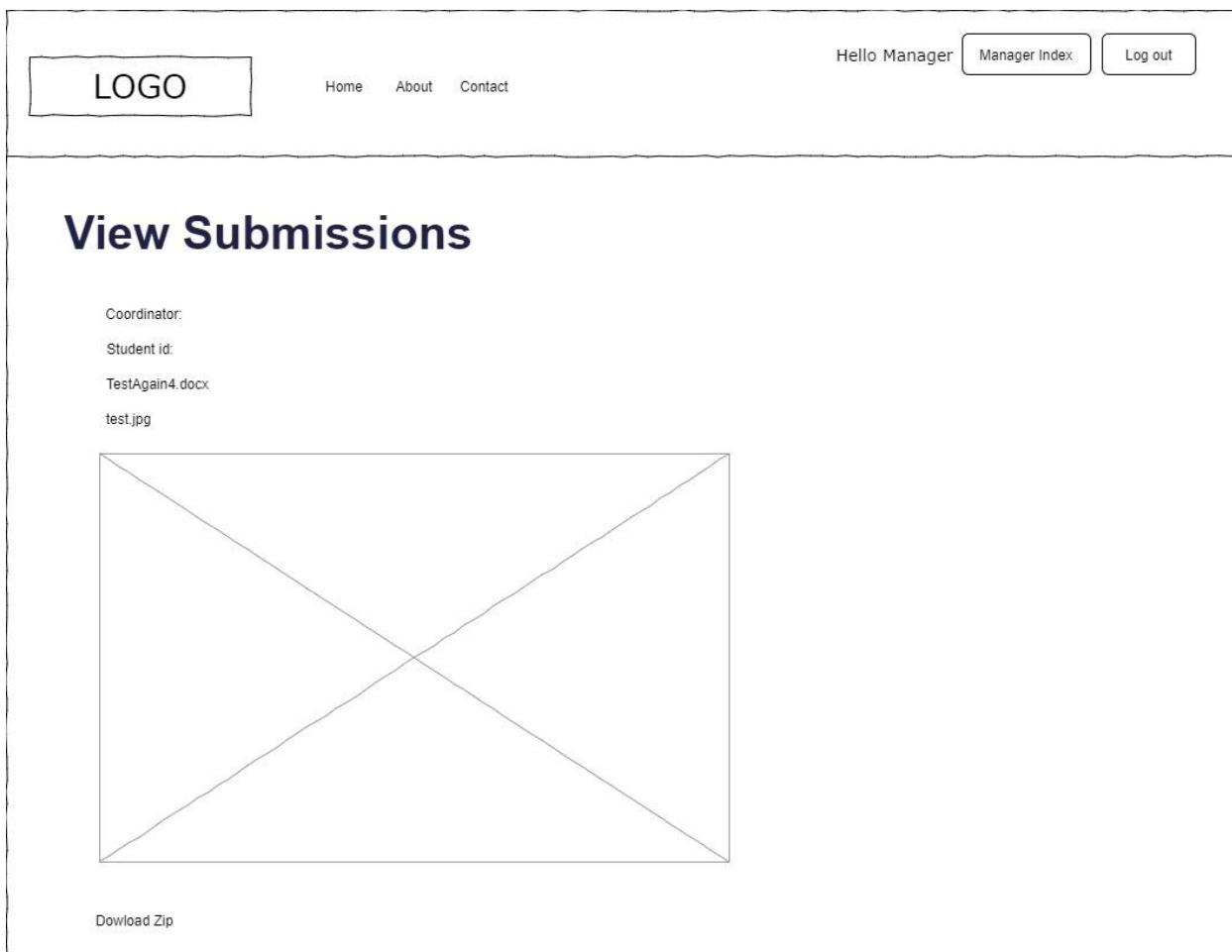


Figure 40: Wireframes Marketing Manager View Submissions

Marketing Manager View Contributions

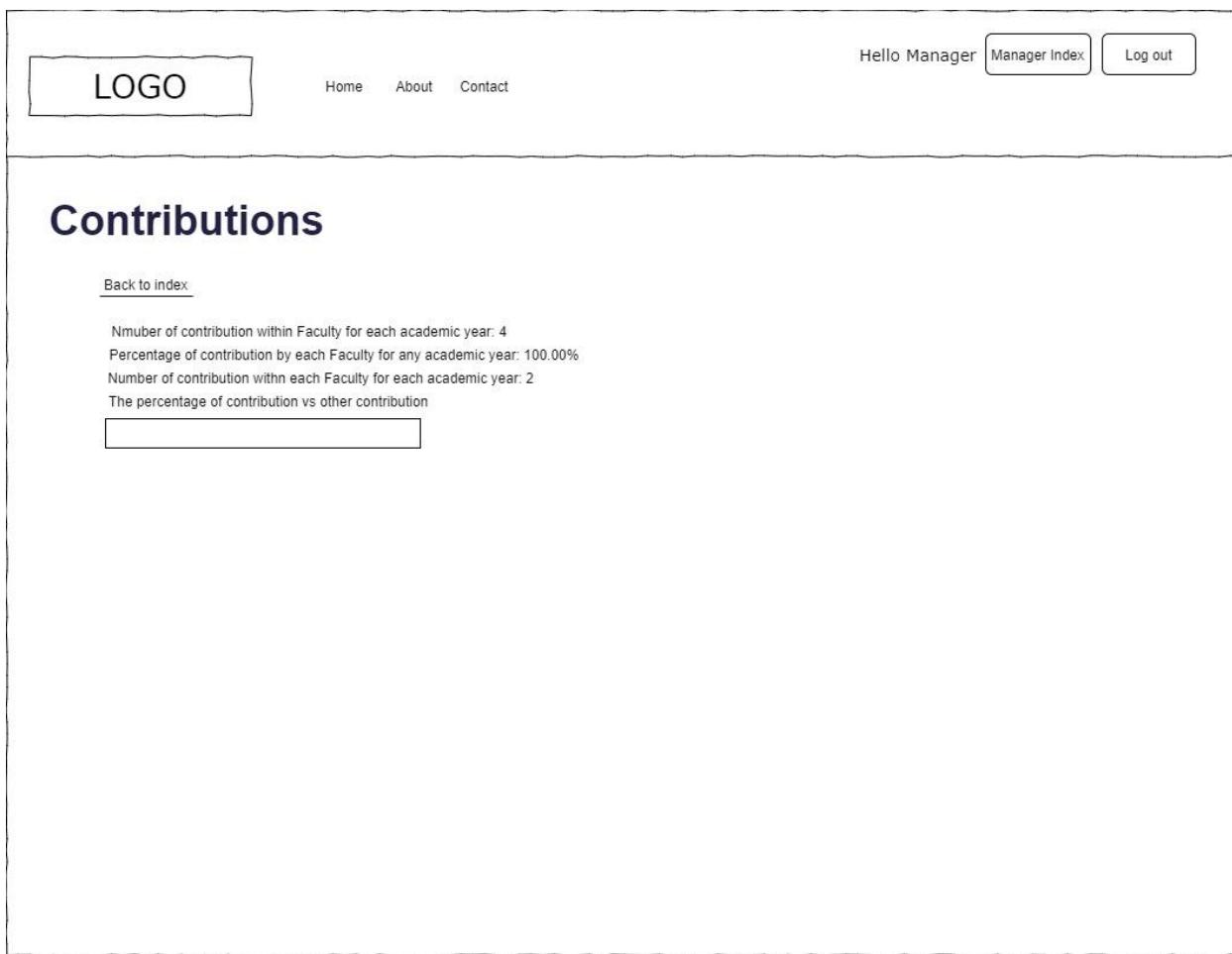


Figure 41: Wireframes Marketing Manager View Contributions

Marketing Manager View Exceptional

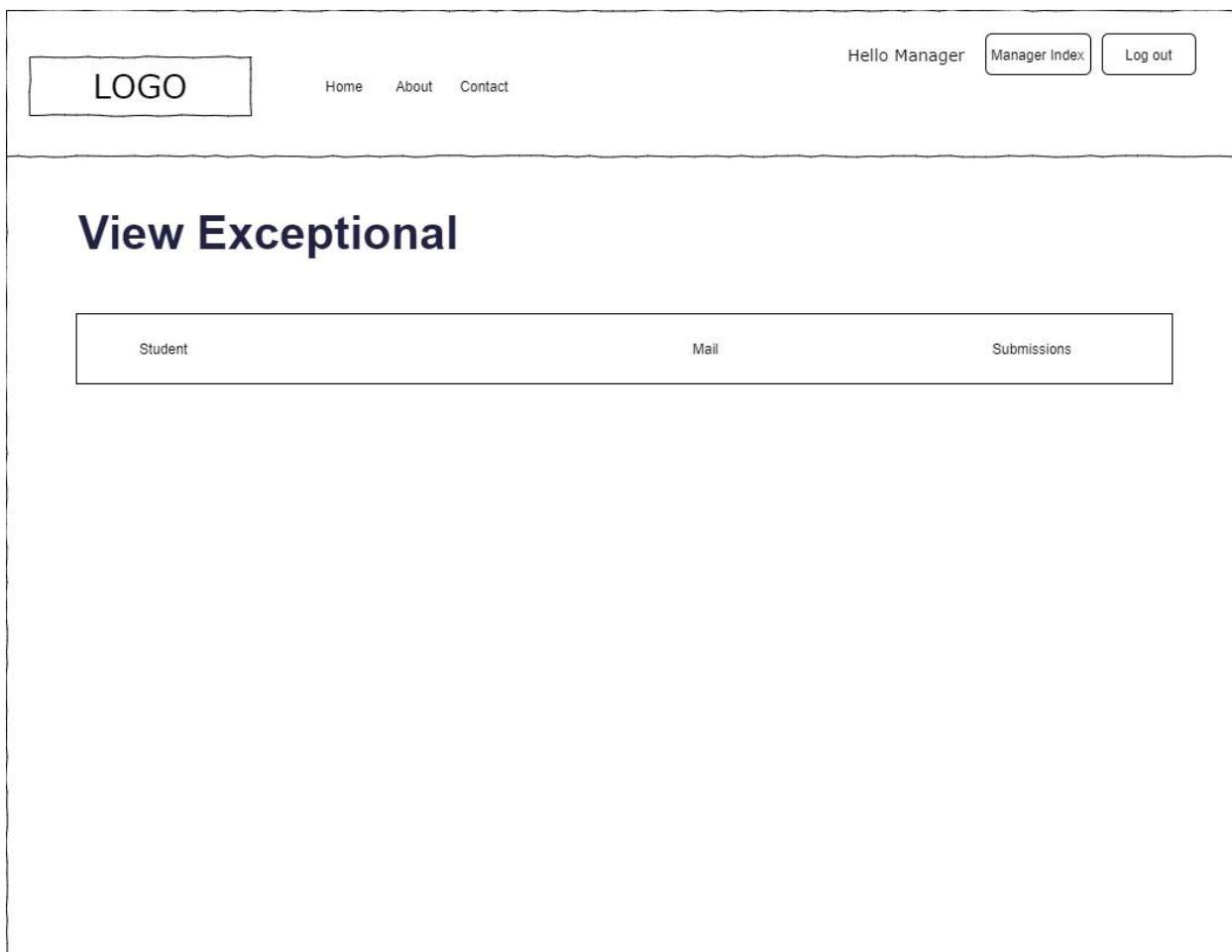


Figure 42: Wireframes Marketing Manager View Exceptional

e. Marketing Coordinator

Marketing Coordinator Index

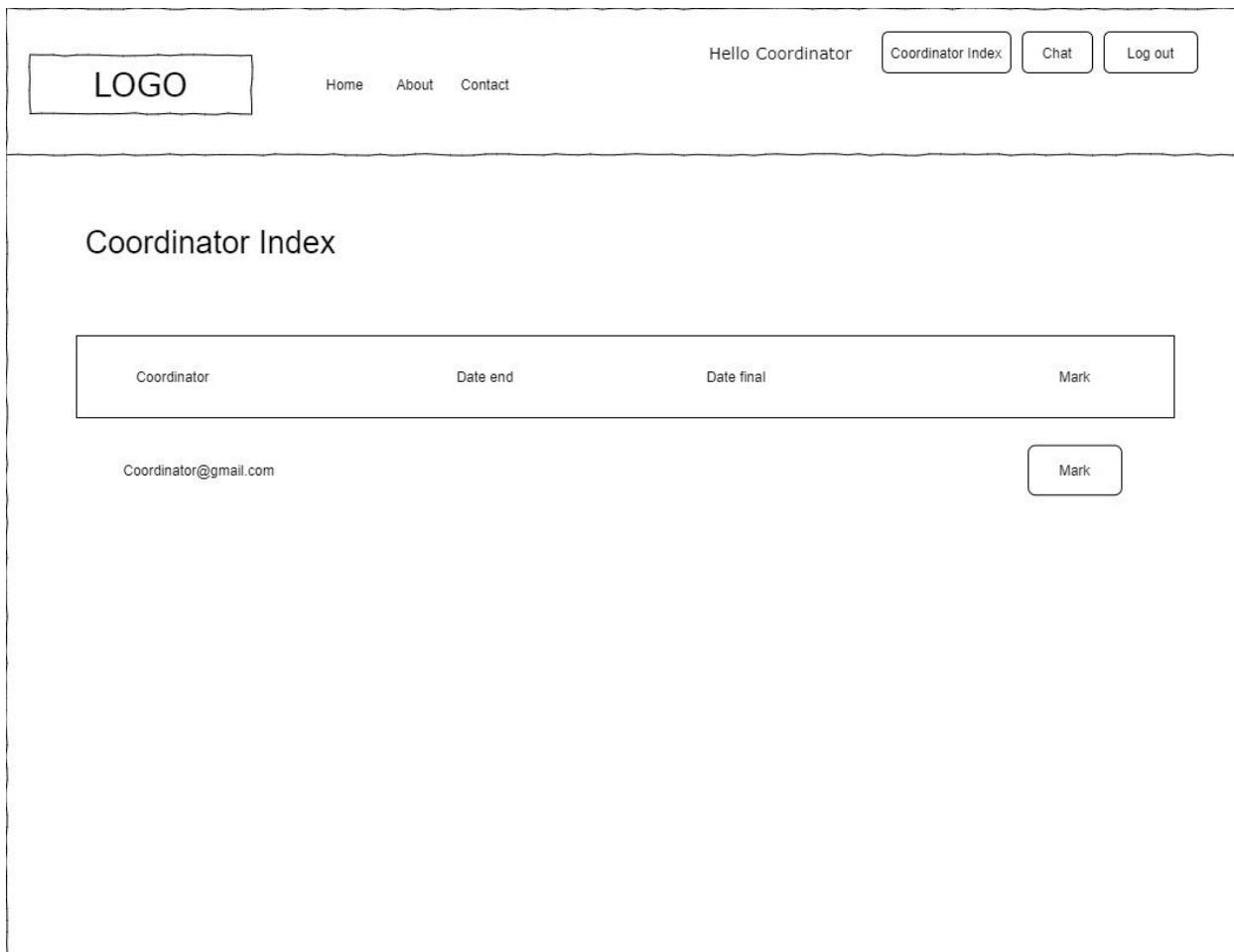


Figure 43: Wireframes Marketing Coordinator Index

Marketing Coordinator Mark

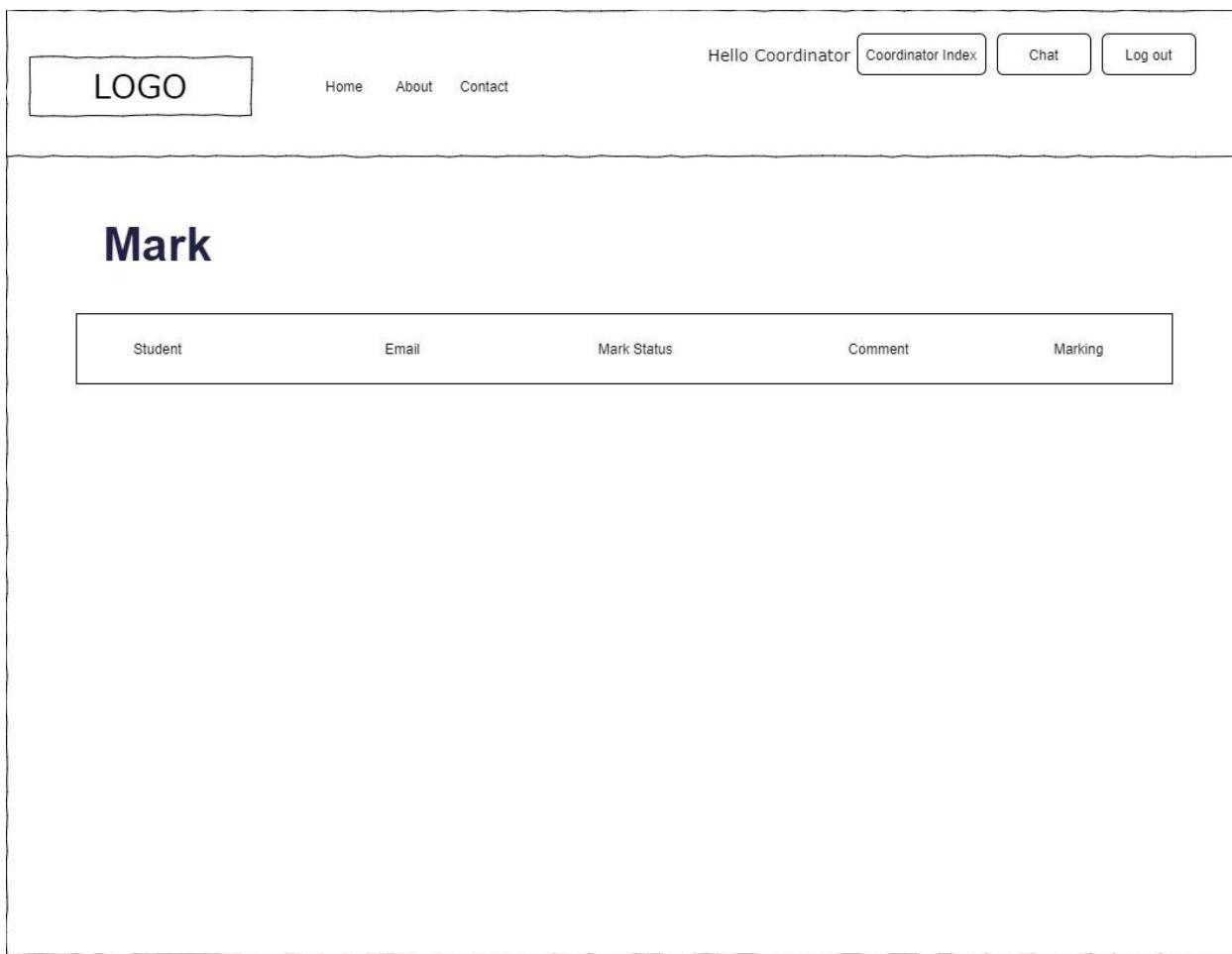
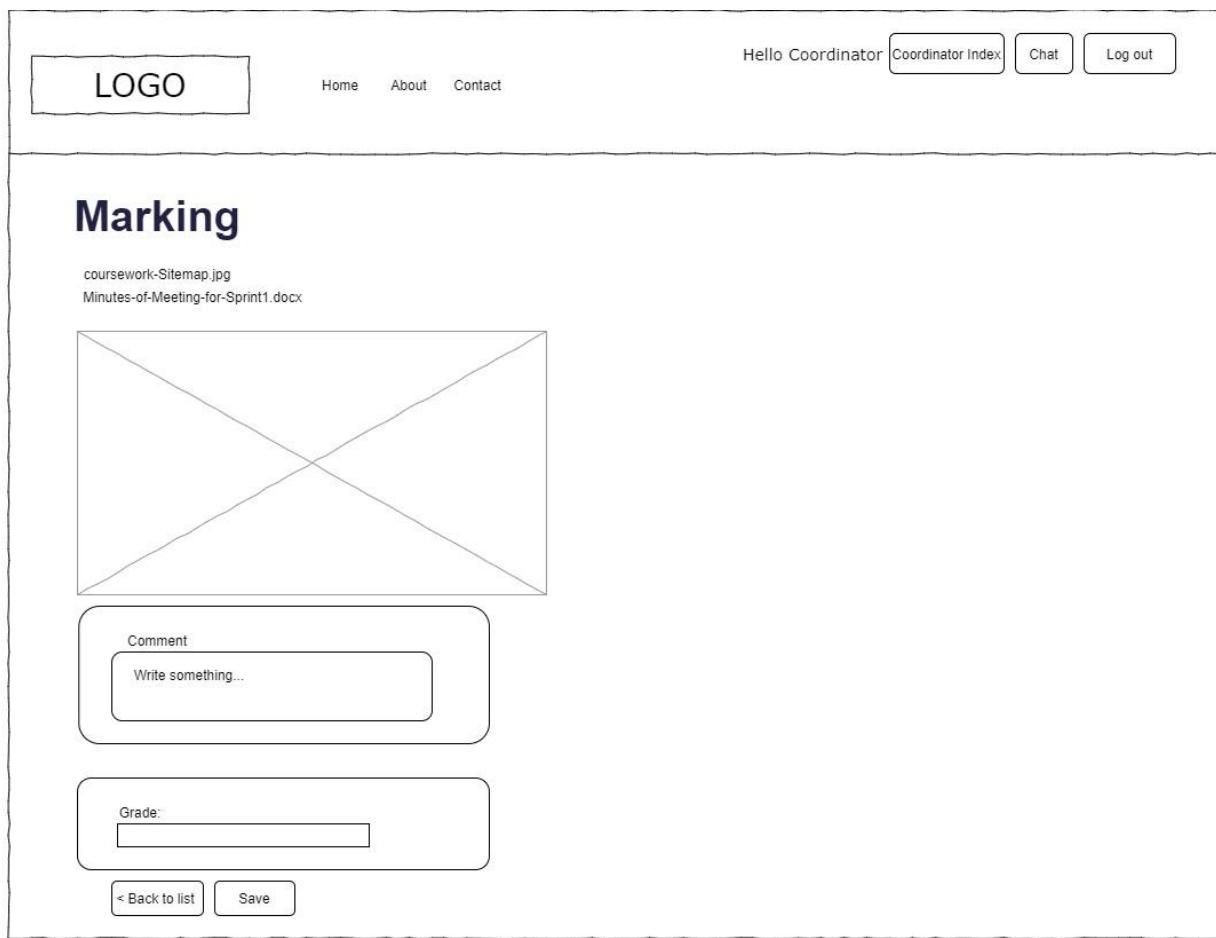


Figure 44: Wireframes Marketing Coordinator Mark

Marketing Coordinator Marking student



The wireframe illustrates the user interface for marking student work. At the top, there is a header bar with a logo placeholder, navigation links for Home, About, and Contact, and a user menu with options like Hello Coordinator, Coordinator Index, Chat, and Log out.

The main content area is titled "Marking". It displays two document attachments: "coursework-Sitemap.jpg" and "Minutes-of-Meeting-for-Sprint1.docx". Below the attachments is a large rectangular area with a diagonal cross, likely representing a redacted or marked document.

Underneath the redacted area is a comment input field labeled "Comment" with the placeholder "Write something...".

Further down is a grade input field labeled "Grade:" with a text input box.

At the bottom of the page are two buttons: "< Back to list" and "Save".

Figure 45: Wireframes Marketing Coordinator Marking student

Marketing Coordinator Marking Index chat with student

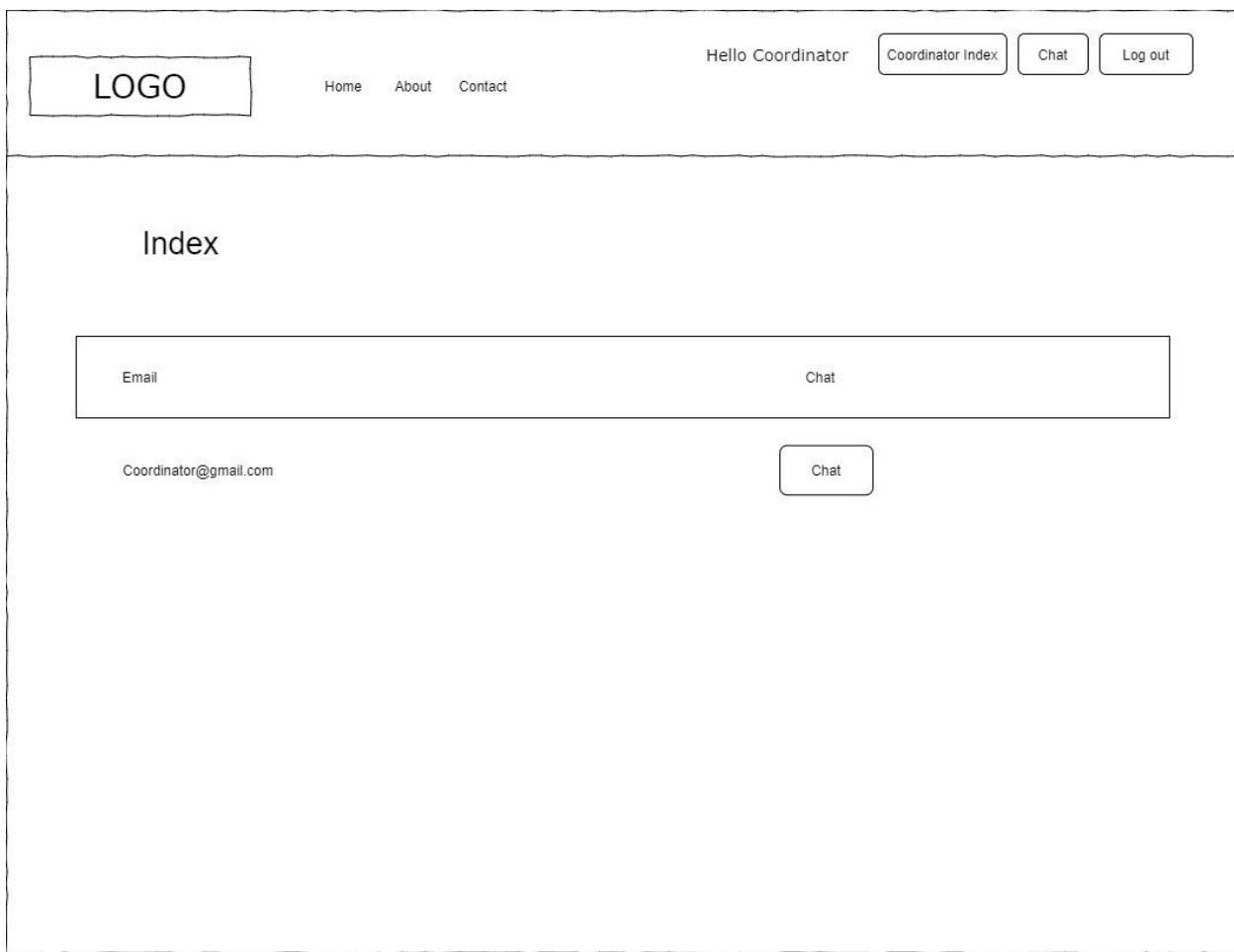


Figure 46: Wireframes Marketing Coordinator Marking Index chat with student

Marketing Coordinator Marking chat with student

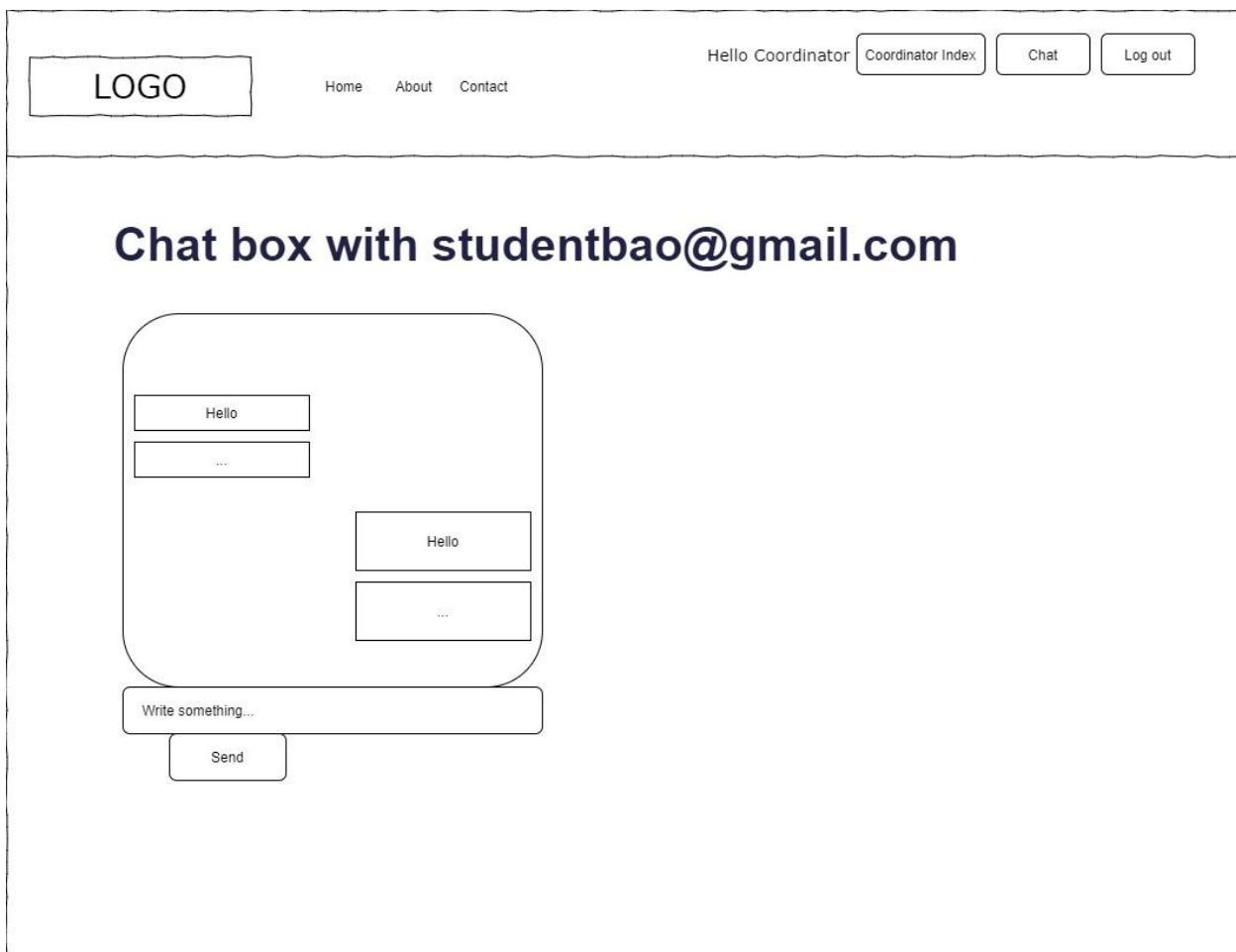


Figure 47: Wireframes Marketing Coordinator Marking chat with student

f. Student

Student Index

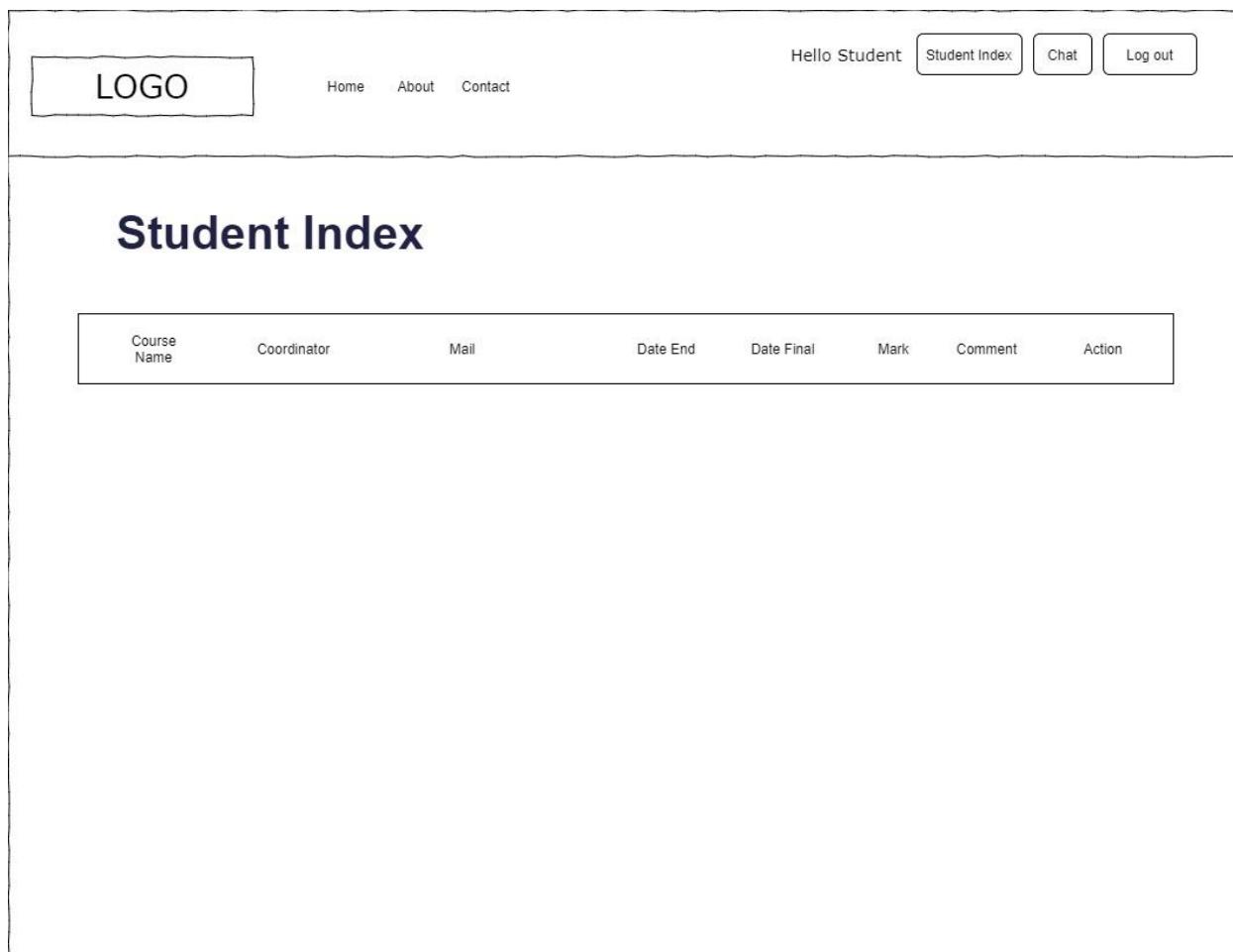


Figure 48: Wireframes Student Index

Student Submit file

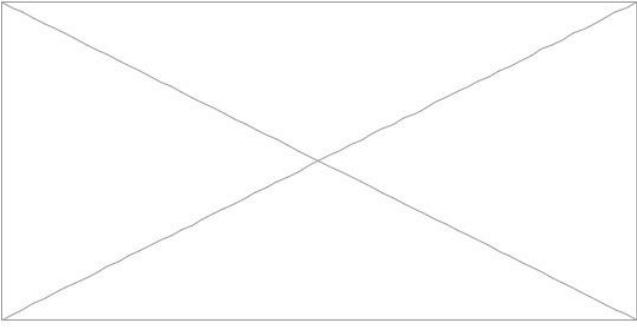
LOGO

Home About Contact

Hello Student [Student Index](#) [Chat](#) [Log out](#)

Submit File

Coordinator:
Mail:
End:
Coursework.jpg
Delete
Test.docx
Delete



[Choose](#)

[Create](#)

[< Back to list](#)

Figure 49: Wireframes Student Submit file

Student Index chat Marketing Coordinator

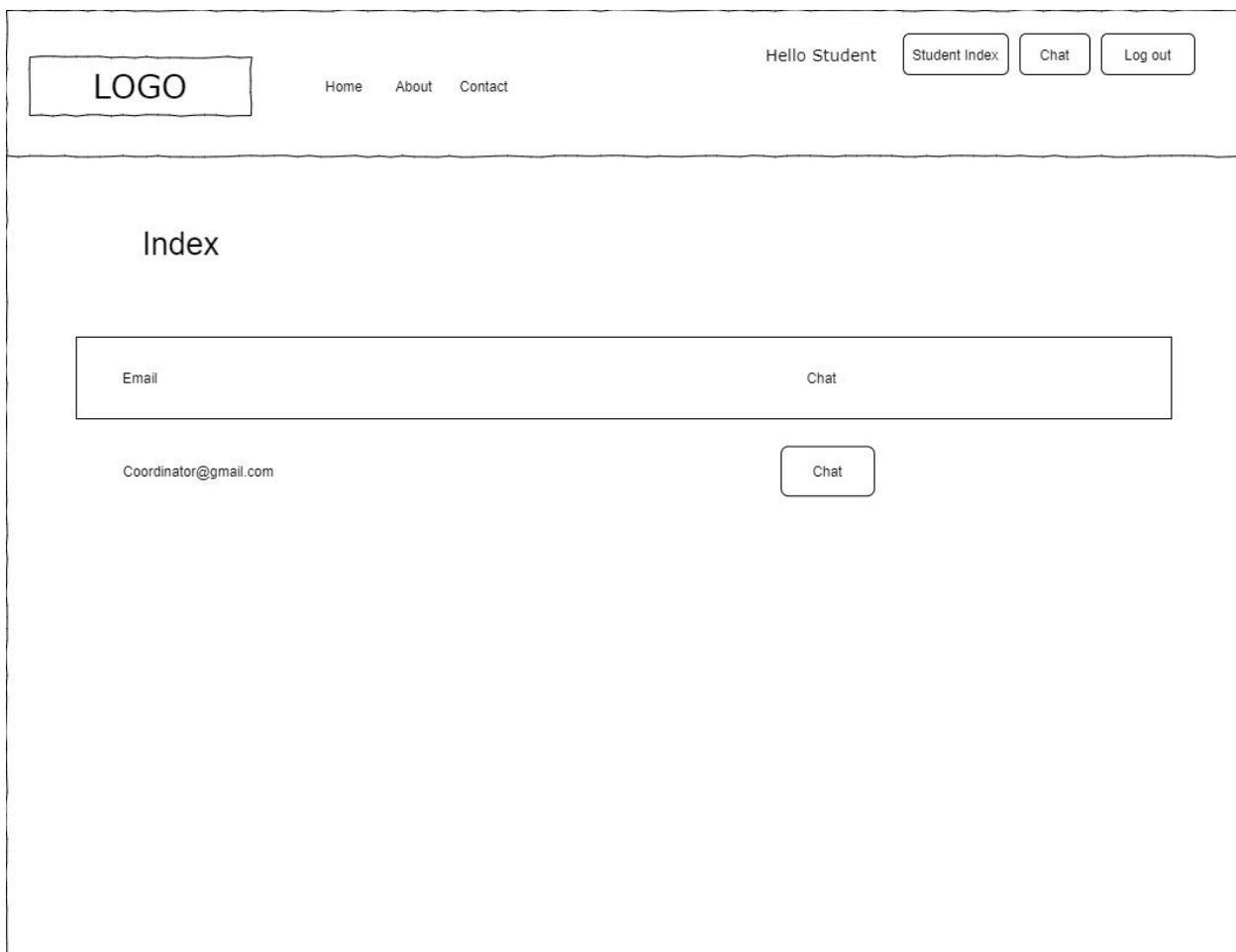


Figure 50: Wireframes Student Index chat Marketing Coordinator

Student chat Marketing Coordinator

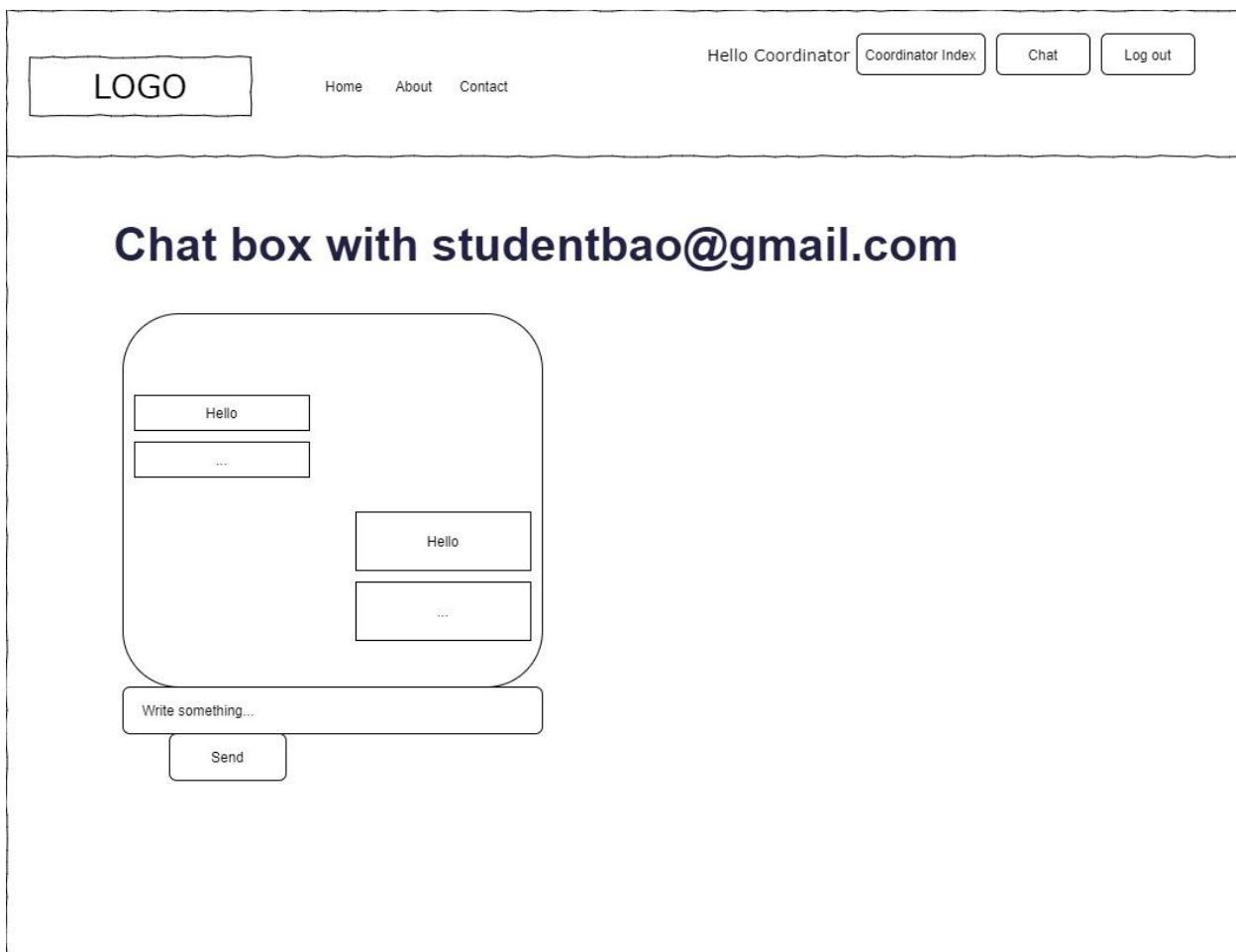


Figure 51: Wireframes Student chat Marketing Coordinator

g. Guest

Guest Index

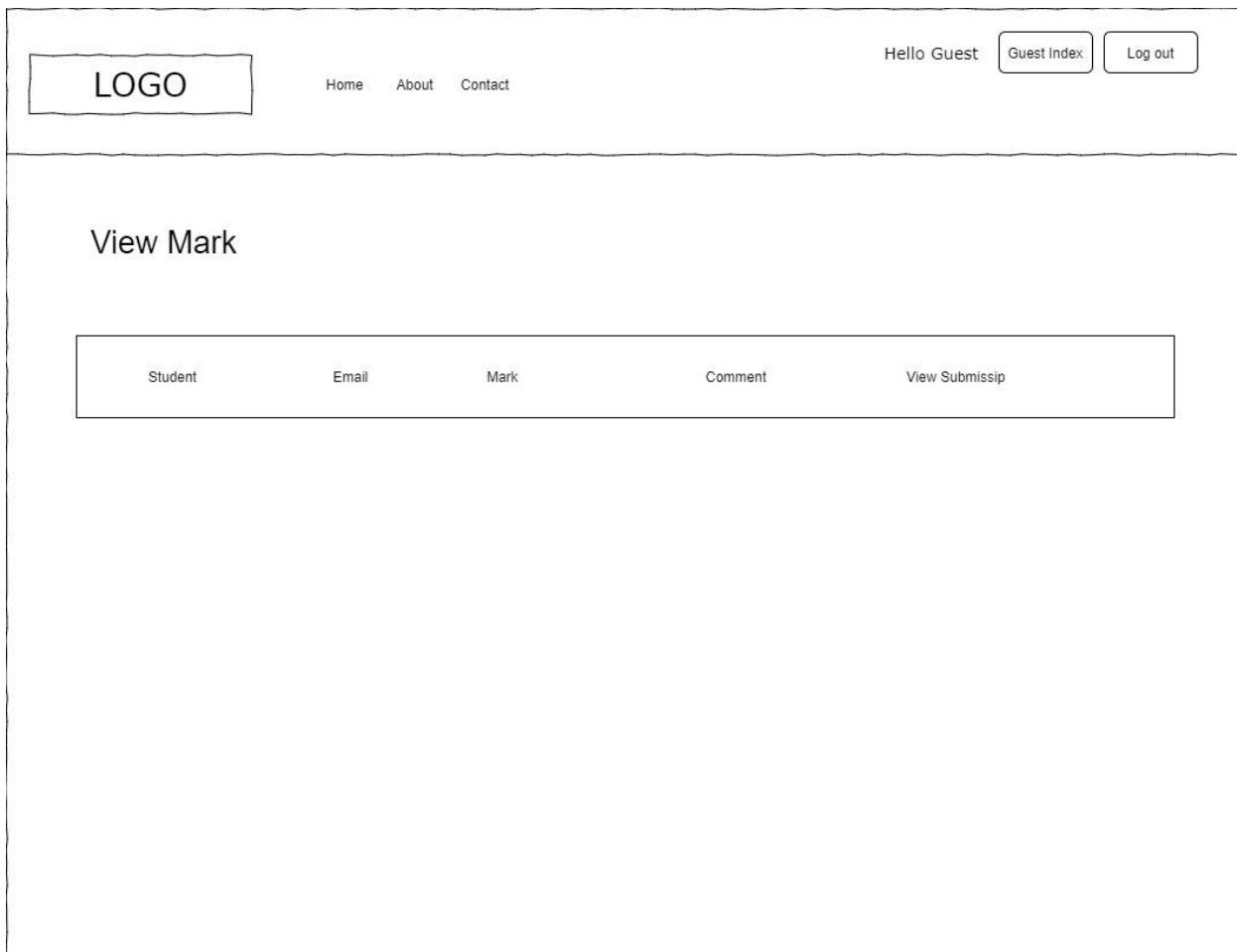


Figure 52: Wireframes Guest Index

Guest view submission



View Submissions

Coordinator:

Student id:

TestAgain4.docx

test.jpg

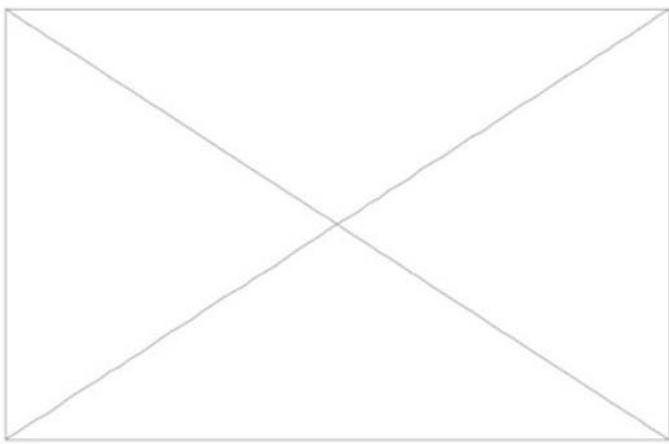


Figure 53: Wireframes Guest view submission

3. Interface

a. Home page

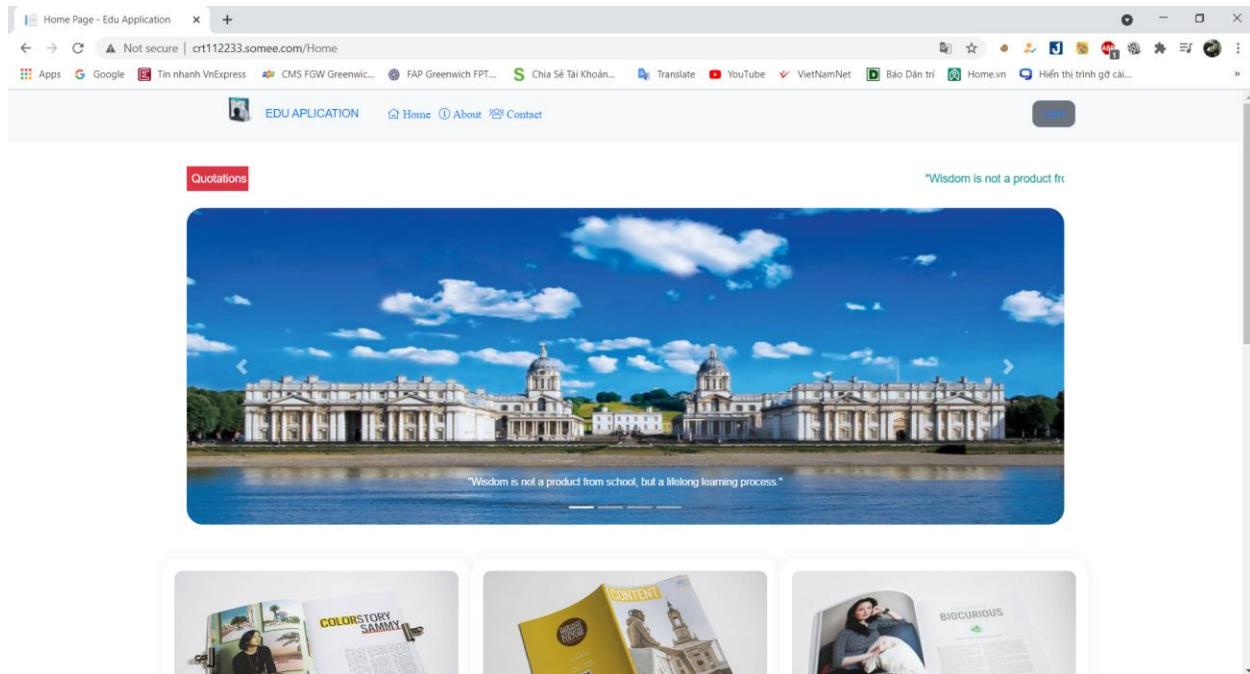


Figure 54: Interface Home page

b. Login page

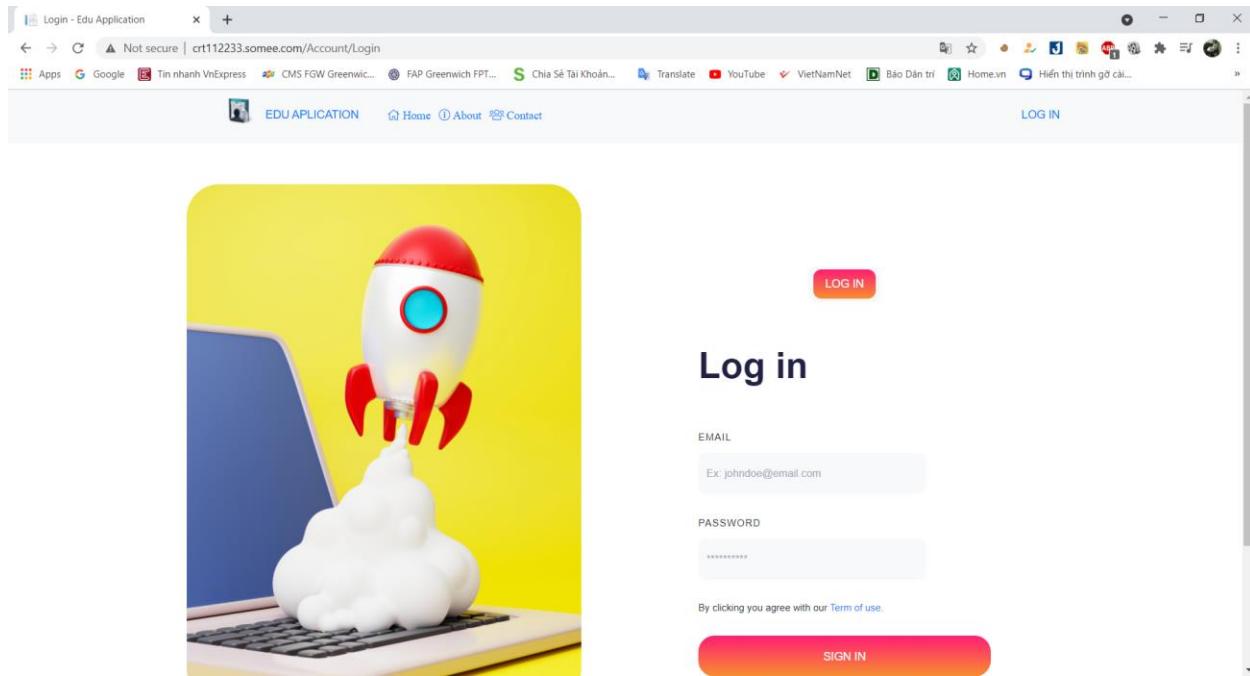
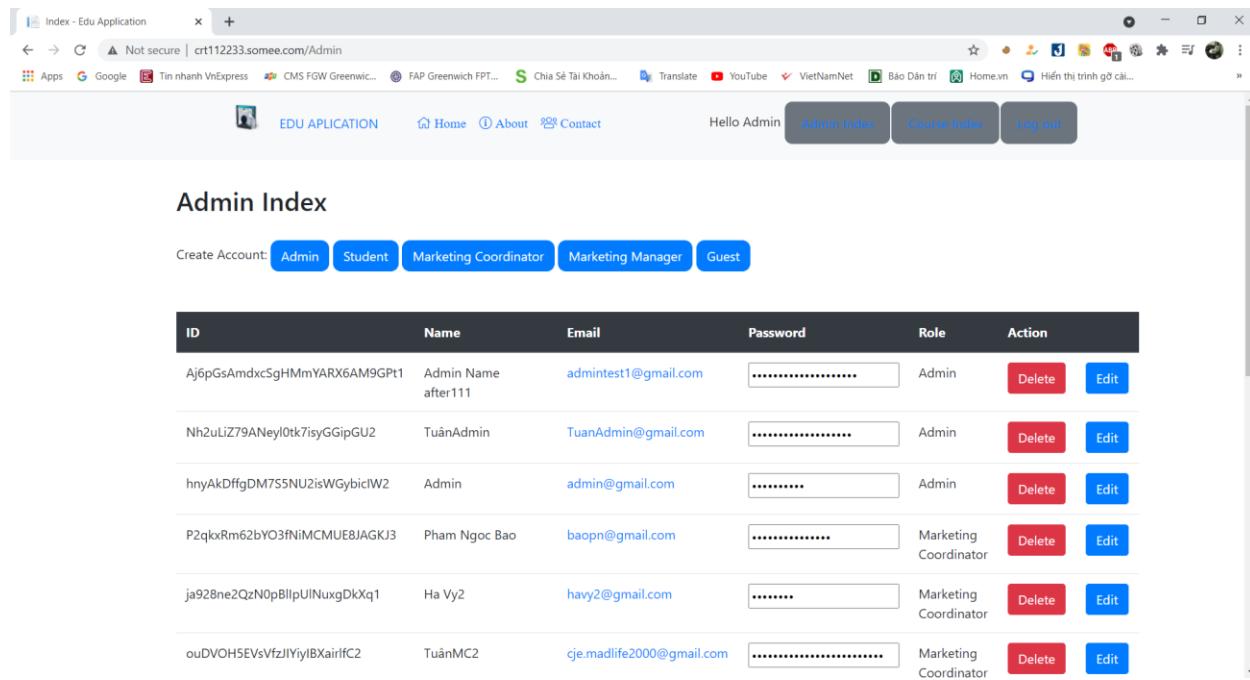


Figure 55: Interface Login page

c. Admin

Admin Index

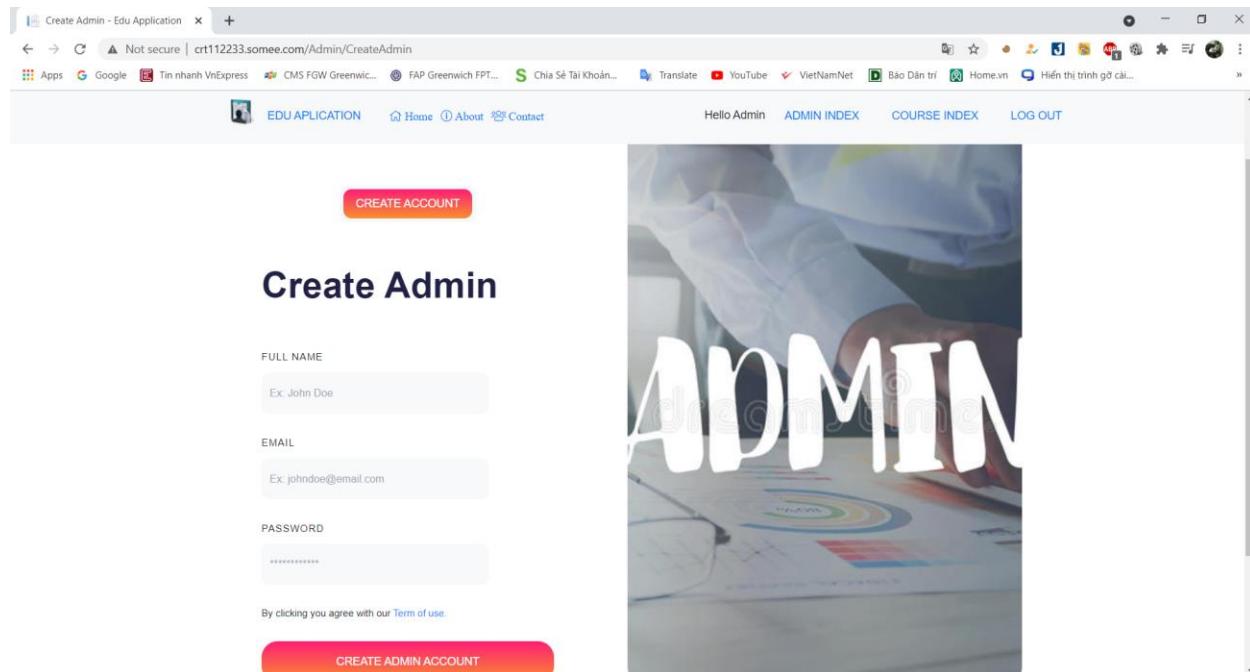


The screenshot shows the 'Admin Index' page of the Edu Application. At the top, there is a navigation bar with links for 'Home', 'About', 'Contact', 'Hello Admin', 'Admin Index' (which is highlighted in blue), 'Course Index', and 'Log out'. Below the navigation bar is a section titled 'Admin Index' with a sub-section 'Create Account' containing buttons for 'Admin', 'Student', 'Marketing Coordinator', 'Marketing Manager', and 'Guest'. The main content area displays a table of user data:

ID	Name	Email	Password	Role	Action
Aj6pGsAmdx5gHMmYARX6AM9Gpt1	Admin Name after111	adminintest1@gmail.com	*****	Admin	<button>Delete</button> <button>Edit</button>
Nh2uLiZ79ANeyl0tk7isyGGipGU2	TuanAdmin	TuanAdmin@gmail.com	*****	Admin	<button>Delete</button> <button>Edit</button>
hnyAkDffgDM755NU2isWGYbicIW2	Admin	admin@gmail.com	*****	Admin	<button>Delete</button> <button>Edit</button>
P2qkxRm62bYO3fNiMCMUE8JAGKJ3	Pham Ngoc Bao	baopn@gmail.com	*****	Marketing Coordinator	<button>Delete</button> <button>Edit</button>
ja928ne2QzNOpBllpUiNuxgDkXq1	Ha Vy2	havy2@gmail.com	*****	Marketing Coordinator	<button>Delete</button> <button>Edit</button>
ouDV0H5EVsVfzJIYiyIBXairlfC2	TuanMC2	cje.madlife2000@gmail.com	*****	Marketing Coordinator	<button>Delete</button> <button>Edit</button>

Figure 56: Interface Admin Index

Admin create Admin account



The screenshot shows the 'Create Admin' page of the Edu Application. At the top, there is a navigation bar with links for 'Home', 'About', 'Contact', 'Hello Admin', 'ADMIN INDEX' (which is highlighted in blue), 'COURSE INDEX', and 'LOG OUT'. The main content area has a 'CREATE ACCOUNT' button at the top right. Below it is a large watermark image of a person's hands holding a pen over a document with the word 'ADMIN' written in large letters. To the left of the watermark, there are three input fields: 'FULL NAME' (with placeholder 'Ex: John Doe'), 'EMAIL' (with placeholder 'Ex: john.doe@email.com'), and 'PASSWORD' (with placeholder '*****'). At the bottom left, there is a link 'By clicking you agree with our [Term of use](#)'. At the very bottom is another 'CREATE ADMIN ACCOUNT' button.

Figure 57:Interface Admin create Admin account

Admin create student account

The screenshot shows a web browser window titled "Create Student - Edu Application". The URL is "crt112233.somee.com/Admin/CreateStudent". The page header includes links for "Home", "About", "Contact", "Hello Admin", "Admin Index" (which is highlighted in blue), "Course Index", and "Log out". The main content area is titled "Create Student Account" and contains a form for creating a student account. The form fields are: "Full name" (Ex: John Doe), "Email" (Ex: johndoe@email.com), "Password" (*****), and "Marketing Coordinator" (Pham Ngoc Bao (P2qkxRm62bYO3fr)). Below the form, a note states "By creating an account student need agree to [Terms & Privacy](#)". There are two buttons at the bottom: "Create" and "< Back to List".

Figure 58: Interface Admin create student account

Admin create Marketing Coordinator account

The screenshot shows a web browser window titled "Create Marketing Coordinator - Edu Application". The URL is "crt112233.somee.com/Admin/CreateMarketingCoordinator". The page header includes links for "Home", "About", "Contact", "Hello Admin", "Admin Index" (highlighted in blue), "Course Index", and "Log out". The main content area is titled "Create Marketing Coordinator" and contains a form for creating a Marketing Coordinator account. The form fields are: "Full name" (Ex: John Doe), "Course name" (Ex: Course name), "Email" (Ex: johndoe@email.com), and "Password" (*****). Below the form, a note states "By creating an account, need agree to [Terms & Privacy](#)". There are two buttons at the bottom: "Create" and "< Back to List".

© 2021 - Edu Application by ASP.NET

Figure 59: Interface Admin create Marketing Coordinator account

Admin create Guest account

The screenshot shows a web browser window titled "Create Guest - Edu Application". The URL is "crt112233.somee.com/Admin/CreateGuest". The page header includes links for "Home", "About", "Contact", "Hello Admin", "Admin Index" (which is highlighted in blue), "Course Index", and "Log out". Below the header, the main content area has a title "Create Guest" and a sub-instruction "Please fill in this form to create an account.". There are three input fields: "Full name" (Ex: John Doe), "Email" (Ex: johndoe@email.com), and "Password" (*****). A dropdown menu for "Marketing Coordinator" shows "Ha Vy2 (ja928ne2QzN0pBlpUlNxgI)". Below the form, a note says "By creating an account student need agree to [Terms & Privacy](#)". At the bottom are two buttons: "Create" and "< Back to List".

Figure 60: Interface Admin create Guest account

Admin create Marketing Manager account

The screenshot shows a web browser window titled "Create Marketing Manager - Edu Application". The URL is "crt112233.somee.com/Admin/CreateMarketingManager". The page header includes links for "Home", "About", "Contact", "Hello Admin", "ADMIN INDEX" (highlighted in orange), "COURSE INDEX", and "LOG OUT". Below the header, there is a large graphic on the right side featuring various marketing icons like a lightbulb, charts, money, and a person's head. On the left, there is a title "Create Marketing Manager" and a "CREATE ACCOUNT" button. The form fields are: "FULL NAME" (Ex: John Doe), "EMAIL" (Ex: johndoe@email.com), and "PASSWORD" (*****). A note at the bottom says "By clicking you agree with our [Term of use](#)".

Figure 61: Interface Admin create Marketing Manager account

Admin Edit account

The screenshot shows a web browser window titled "Edit Account - Edu Application". The URL is "crt112233.somee.com/Admin/Edit/Aj6pGsAmdxcSgHMmYARX6AM9GPt1?role=Admin". The page header includes "Hello Admin" and navigation links for "Admin Index", "Course Index", and "Log out". Below the header is a table with the following data:

ID	Name	Email	Password	Role	Active Status
Aj6pGsAmdxcSgHMmYARX6AM9GPt1	Admin N	admintes	*****	Adi	<input checked="" type="checkbox"/>

At the bottom of the form are two buttons: "Save" and "< Back to List".

© 2021 - Edu Application by ASP.NET
Web hosting by Somee.com

Figure 62: Interface Admin Edit account

Admin Index course

The screenshot shows a web browser window titled "Index Course - Edu Application". The URL is "crt112233.somee.com/Admin/IndexCourse". The page header includes "Hello Admin" and navigation links for "Admin Index", "Courses Index", and "Logout". Below the header is a table titled "Index Course" with columns: Course Name, Coordinator, Mail, Date end, Date Final, Guest, View Mark, and Edit date end. Three rows of course data are listed:

Course Name	Coordinator	Mail	Date end	Date Final	Guest	View Mark	Edit date end
Bao pn	P2qkxRm62bYO3fNiMCMUE8JAGKJ3	baopn@gmail.com	13/04/2021 09:20:00 AM	15/04/2021 09:20:00 AM	5f7tnNKnH5dpTa8ujRfaCxdAQh1	View	Edit
HaVY1	ja928ne2QzN0pBllpUINuxgDkXq1	havy2@gmail.com	01/04/2021 04:18:00 PM	01/04/2021 04:18:00 PM		View	Edit
QTKD	ouDV0H5EvSvfzJlYiyIBXairfc2	cje.madlife2000@gmail.com	03/04/2021 06:03:00 PM	10/04/2021 06:03:00 PM	1TDlx1cUo5hYPKiDTjCb4D2tjWY2	View	Edit

At the bottom of the page, there is a copyright notice "© 2021 - Edu Application by ASP.NET" and a link "Web hosting by Somee.com".

Figure 63: Interface Admin Index course

Admin Edit course

The screenshot shows a web browser window titled "Edit Course - Edu Application". The URL is "crt112233.somee.com/Admin/EditCourse?coordinator=P2qkxRm62bYO3fNiMCMUE8JAGKJ3". The page header includes "Hello Admin" and navigation links for "Admin Index", "Courses Index", and "Logout". Below the header is a form titled "Edit Course" with fields for Coordinator, Old date end, Old date Final, nameCourse, New date End, and New date Final. The "Coordinator" field contains "P2qkxRm62bYO3fNiMCMUE8JAGKJ3". The "Old date end" field shows "13/04/2021 09:20:00 AM". The "Old date Final" field shows "15/04/2021 09:20:00 AM". The "nameCourse" field is empty. The "New date End" field has a dropdown menu showing "Bac" and a date picker set to "mm/dd/yyyy --:-- --". The "New date Final" field has a date picker set to "mm/dd/yyyy --:-- --". A "Save" button is at the bottom left.

At the bottom of the page, there is a copyright notice "© 2021 - Edu Application by ASP.NET" and a link "Web hosting by Somee.com".

Figure 64: Interface Admin Edit course

d. Marketing Manager

Marketing Manager Index

The screenshot shows a web browser window titled "Marketing Manager Index - Edu". The address bar indicates the URL is "crt112233.somee.com/MarketingManager". The page header includes the "EDU APPLICATION" logo, "Home", "About", "Contact", and user information "Hello TuanMM@gmail.com". The main content area is titled "Marketing Manager Index" and displays a table of courses:

Course Name	Coordinator	Coordinator mail	Date End	Date Final	View Mark	Exceptional	Contributions
Bao pn	P2qkxRm62bYO3fNiMCMUE8JAGKJ3	baopn@gmail.com	13/04/2021 09:20:00 AM	15/04/2021 09:20:00 AM	View	View	View
HaVy1	ja928ne2QzNOpBlpUlNxugDkXq1	havy2@gmail.com	01/04/2021 04:18:00 PM	01/04/2021 04:18:00 PM	View	View	View
QTKD	ouDVOH5EVsVfzJlYiyIBXairffC2	cje.madlife2000@gmail.com	03/04/2021 06:03:00 PM	10/04/2021 06:03:00 PM	View	View	View

At the bottom of the page, there is a copyright notice "© 2021 - Edu Application by ASP.NET" and a link "Web hosting by Somee.com".

Figure 65: Interface Marketing Manager Index

Marketing Manager View Course

The screenshot shows a web browser window titled "Mark - Edu Application". The address bar indicates the URL is "crt112233.somee.com/MarketingManager/viewMark?coordinator=ouDVOH5EVsVfzJlYiyIBXairffC2". The page header includes the "EDU APPLICATION" logo, "Home", "About", "Contact", and user information "Hello TuanMM@gmail.com". The main content area is titled "View Mark" and displays a table of student marks:

Student	Email	Mark	Comment	View Submissions
2y47LajwVVhKDQfH5czQG1OyEBh2	studentbao@gmail.com	Accept		View submissions
g0JGKjlmoDS21Bb8djO3ndSbfYp2	TuanSTUDENT2@gmail.com	Accept		View submissions
sHZ1uXsrZc3gKaqf8czjLOvs13	pnb@gmail.com	Accept		View submissions
uyKjyKQWaYgjs06izMU99ftm9HB2	TuanSTUDENT1@gmail.com	Accept	GOOD JOB	View submissions

At the bottom of the page, there is a copyright notice "© 2021 - Edu Application by ASP.NET" and a link "Web hosting by Somee.com".

Figure 66: Interface Marketing Manager View Courses

Marketing Manager View Submissions

The screenshot shows a web browser window for 'View Submissions - Edu Application'. The URL is crt112233.somee.com/MarketingManager/viewSubmissions?coordinator=ouDVOH5EVsVfzJlYiyIBXairfC2&Student=2y47LajwVvhKDQfH5cz.... The page title is 'View Submissions'. The top navigation bar includes links for Home, About, Contact, and a user account (Hello TuanMM@gmail.com). A 'Logout' button is also present.

The main content area displays a PDF document titled 'Minutes-of-Meeting-for-Sprint1.docx'. The PDF header includes the University of Greenwich logo and the meeting details: 'Sprint 1 review meeting' and 'Nguyen Van Tuan'. The PDF content is organized into sections: '1. MEETING OBJECTIVE', '2. ATTENDEES PRESENT', and '3. AGENDA & NOTES, DECISIONS, ISSUES'. The '2. ATTENDEES PRESENT' section lists attendees with their roles, names, emails, and phone numbers:

NAME	DEPARTMENT / DIVISION	EMAIL	PHONE
Nguyen Hoang Thuyen	Programmer, Product owner	thuyennghg1808@fpt.edu.vn	+84 833059889
Pham Ngoc Bao	Programmer, Web designer	bao.ngoingh1813@fpt.edu.vn	+84 98326420
Do The Vien	Database designer	vienhptg1723@fpt.edu.vn	+84 964063455
Nguyen Van Tuan	Tester, Scrum master	tuanvngn1829@fpt.edu.vn	+84 880329115
Do Ho Nguyen	Tester	Nguyenhdg1825@fpt.edu.vn	

At the bottom of the PDF, there is a 'Download Zip' link.

Figure 67: Interface Marketing Manager View Submissions

Marketing Manager View Contributions

The screenshot shows a web browser window for 'Contributions - Edu Application'. The URL is crt112233.somee.com/MarketingManager/Contributions?coordinator=ouDVOH5EVsVfzJlYiyIBXairfC2. The page title is 'Contributions'. The top navigation bar includes links for Home, About, Contact, and a user account (Hello TuanMM@gmail.com). A 'Logout' button is also present.

The main content area displays a summary of contributions. It includes the following text:

Number of contributions within each Faculty for each academic year: 6
Percentage of contributions by each Faculty for any academic year: 100.00%
Number of contributors within each Faculty for each academic year: 3
The percentage of contributions vs other contributions
ouDVOH5EVsVfzJlYiyIBXairfC2

At the bottom of the page, there is a 'Back to index' link and a copyright notice: '© 2021 - Edu Application by ASP.NET' and 'Web hosting by Somee.com'.

Figure 68: Interface Marketing Manager View Contributions

Marketing Manager View Exceptional



Figure 69: Interface Marketing Manager View Exceptional

e. Marketing Coordinator

Marketing Coordinator Index

The screenshot shows a web browser window titled "Index - Edu Application". The URL is "crt112233.somee.com/Course". The page header includes "EDU APPLICATION", "Home", "About", "Contact", "Hello TuânMC2", "Coordinator Index" (which is highlighted), "Chat", and "Log out". Below the header is a section titled "Coordinator Index" containing a table:

Coordinator	Date end	Date final	Marking
ouDVOH5EVsVfzJlYiyLBXairlfC2	03/04/2021 06:03:00 PM	10/04/2021 06:03:00 PM	<button>Mark</button>

At the bottom of the page, there is a copyright notice "© 2021 - Edu Application by ASP.NET" and a link "Web hosting by Somee.com". A status bar at the bottom of the browser window says "Waiting for crt112233.somee.com...".

Figure 70: Interface Marketing Coordinator Index

Marketing Coordinator Mark

The screenshot shows a web browser window titled "Mark - Edu Application". The URL is "crt112233.somee.com/Course/Mark". The page header includes "EDU APPLICATION", "Home", "About", "Contact", "Hello TuânMC2", "Coordinator Index" (which is highlighted), "Chat", and "Log out". Below the header is a section titled "Mark" containing a table:

Student	Email	Mark Status	Comment	Marking
2y47LajwVVhKDQfH5czQG1OyEBh2	studentbao@gmail.com	Accept		<button>Mark</button>
g0JGKjLmoDS21Bb8djO3ndSbfYp2	TuanSTUDENT2@gmail.com	Accept		<button>Mark</button>
sHZ1uXsrZe3gKaqf8czjlOvs13	pnb@gmail.com	Accept		<button>Mark</button>
uyKyjKQWaYgjs06izMU99fTm9HB2	TuanSTUDENT1@gmail.com	Accept	GOOD JOB	<button>Mark</button>

At the bottom of the page, there is a copyright notice "© 2021 - Edu Application by ASP.NET" and a link "Web hosting by Somee.com".

Figure 71: Interface Marketing Coordinator Mark

Marketing Coordinator Marking student

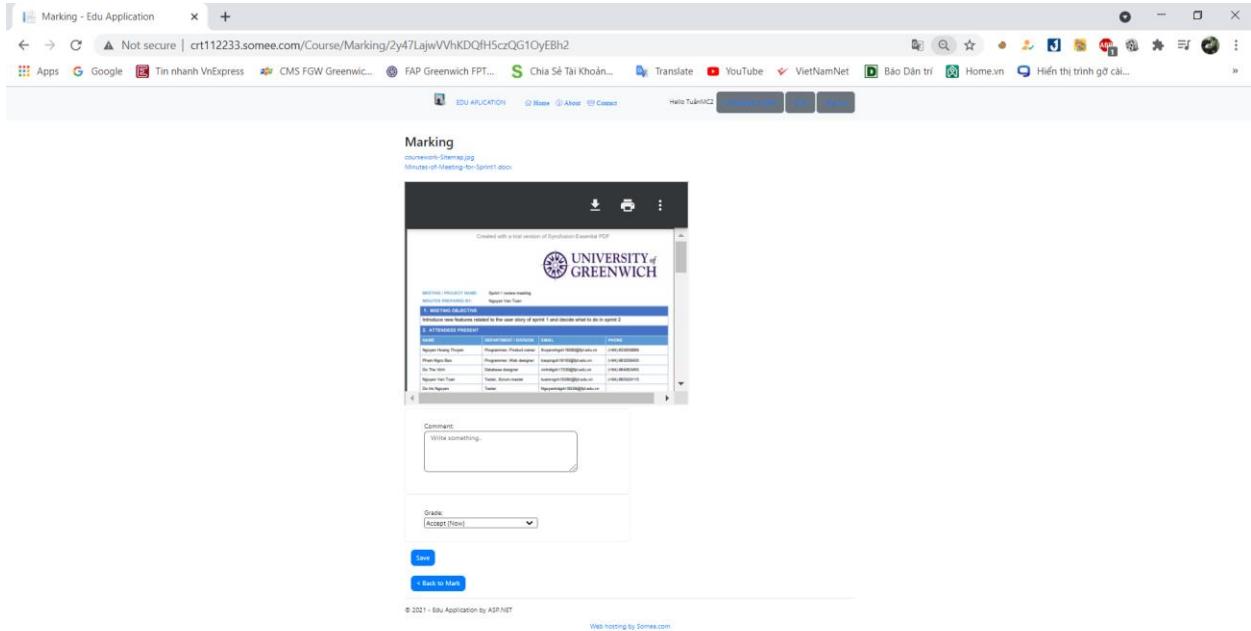


Figure 72: Interface Marketing Coordinator Marking student

Marketing Coordinator Marking Index chat with student

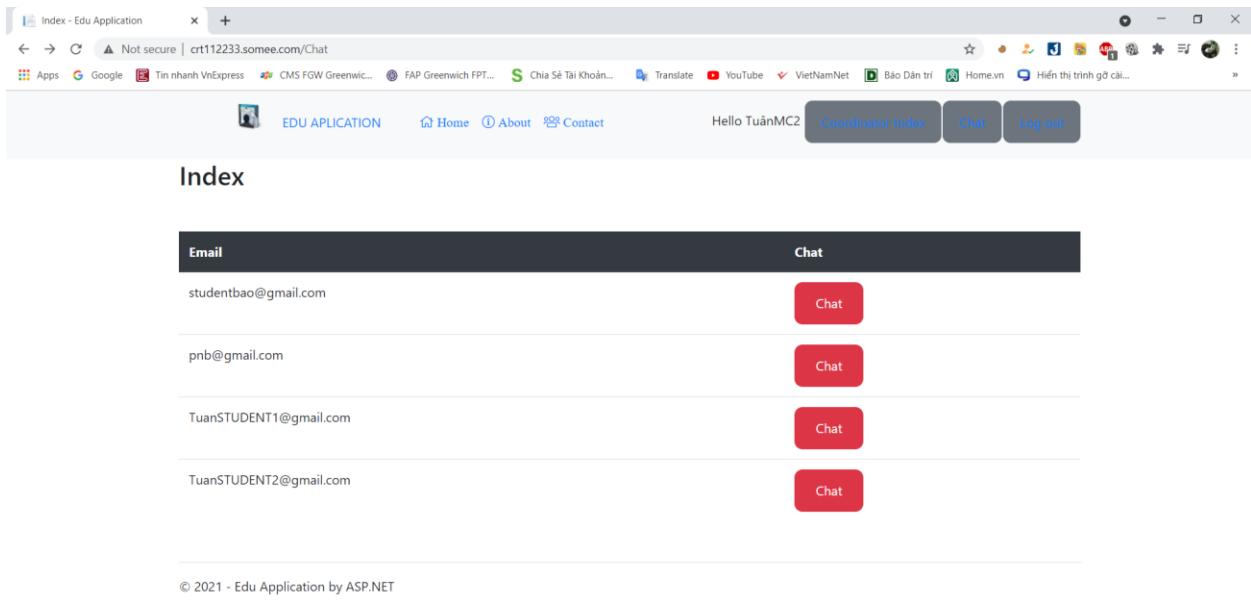


Figure 73: Interface Marketing Coordinator Marking Index chat with student

Marketing Coordinator Marking chat with student

The screenshot shows a web browser window titled "Chat - Edu Application". The URL is crt112233.somee.com/Chat/Chat/shZ1uXsxZe3gKaql8czjlLOvs13. The page header includes links for Home, About, Contact, and a user profile "Hello TuânMC2". Below the header are three buttons: "Student Index" (highlighted in blue), "Chat", and "Log out". The main content area is a chat box with two messages: "hello student." and "good luck.". Below the chat box is a text input field with placeholder text "Write something.." and a "Send" button.

Figure 74: Interface Marketing Coordinator Marking chat with student

f. Student

Student Index

The screenshot shows a web browser window titled "Index - Edu Application". The URL is crt112233.somee.com/Student. The page header includes links for Home, About, Contact, and a user profile "Hello Pham Ngoc Bao". Below the header are three buttons: "Student Index" (highlighted in blue), "Chat", and "Log out". The main content area is titled "Student Index" and displays a table with one row of data:

Course Name	Coordinator	Mail	Date End	Date Final	Mark	Comment	Action
QTKD	ouDVOH5EvsvfzJlYiyIBXairfc2	cje.madlife2000@gmail.com	03/04/2021 06:00:00 PM	10/04/2021 06:00:00 PM	Accept		<button>Submit</button>

At the bottom of the page are copyright and hosting information: "© 2021 - Edu Application by ASP.NET" and "Web hosting by Somee.com".

Figure 75: Interface Student Index

Student Submit file

The screenshot shows a web browser window titled "Submit File - Edu Application". The URL is crt112233.somee.com/Student/Submit?coordinator=ouDV0H5EVsVfzjIYiyIBXairfC2. The page displays a "Submit File" form with the following details:

- Coordinator: ouDV0H5EVsVfzjIYiyIBXairfC2
- Mail: cje.madlife2000@gmail.com
- End: 4/3/2021 6:00:00 PM
- cooperative-Sitemap.jpg
- Delete
- Minutes-of-Meeting-for-Sprint1.docx
- Delete

The main content area shows a PDF document titled "UNIVERSITY of GREENWICH" with a table of attendees. The table includes columns for Name, Department / Division, Email, and Phone.

Name	Department / Division	Email	Phone
Nguyen Hoang Thuyen	Programmer, Product owner	huynguyengh1800@std.edu.vn	+84-833059888
Pham Ngoc Bao	Programmer, Web designer	bao.nguyengh1716@std.edu.vn	+84-86296402
Do The Vinh	Database designer	vinhnguyengh1720@std.edu.vn	+84-964054055
Nguyen Van Tuan	Tester, System manager	tuannguyengh1826@std.edu.vn	+84-860329115
Do Ho Nguyen	Tester	ho.nguyengh1825@std.edu.vn	+84-860329115

Below the PDF, there is a "Choose file" button, a checkbox for "Agree to Terms and Conditions", and a "Create" button. A link "[< Back to index](#)" is also present.

Figure 76: Interface Student Submit file

Student Index chat Marketing Coordinator

The screenshot shows a web browser window titled "Index - Edu Application". The URL is crt112233.somee.com/Chat. The page displays an "Index" section with the following elements:

- Email input field: cje.madlife2000@gmail.com
- Chat button: Chat (highlighted in red)

At the bottom of the page, there is copyright information: "© 2021 - Edu Application by ASP.NET" and "Web hosting by Somee.com".

Figure 77: Interface Student Index chat Marketing Coordinator

Student chat Marketing Coordinator

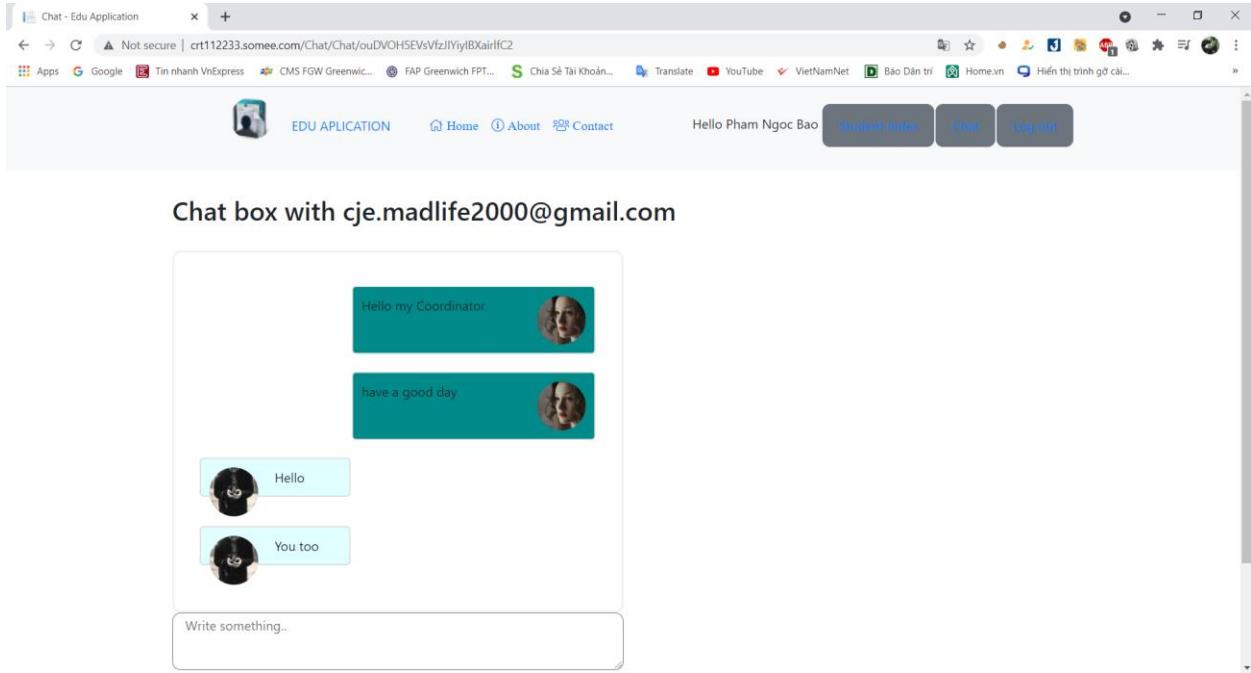


Figure 78: Interface Student chat Marketing Coordinator

g. Guest

Guest Index

The screenshot shows a web browser window titled "Mark - Edu Application". The URL is crt112233.somee.com/Guest. The page header includes links for Home, About, Contact, and a user profile for "Hello TuanGuest1". Below the header are three buttons: "Guest Index", "Chat", and "Log out". The main content area displays a table titled "View Mark". The table has columns for Student, Email, Mark, Comment, and View Submissions. There are four rows of data:

Student	Email	Mark	Comment	View Submissions
2y47LajwVVhKDQfH5czQG1OyEBh2	studentbao@gmail.com	Accept		View
g0JGKjLmoDS21Bb8djO3ndSbfyp2	TuanSTUDENT2@gmail.com	Accept		View
sHZ1uXsrZe3gKaqf8czjlOvs13	pnb@gmail.com	Accept		View
uyKyjKQWaYgjs06izMU99fTm9HB2	TuanSTUDENT1@gmail.com	Accept	GOOD JOB	View

© 2021 - Edu Application by ASP.NET

Web hosting by Somee.com

Figure 79: Interface Guest Index

Guest view submission

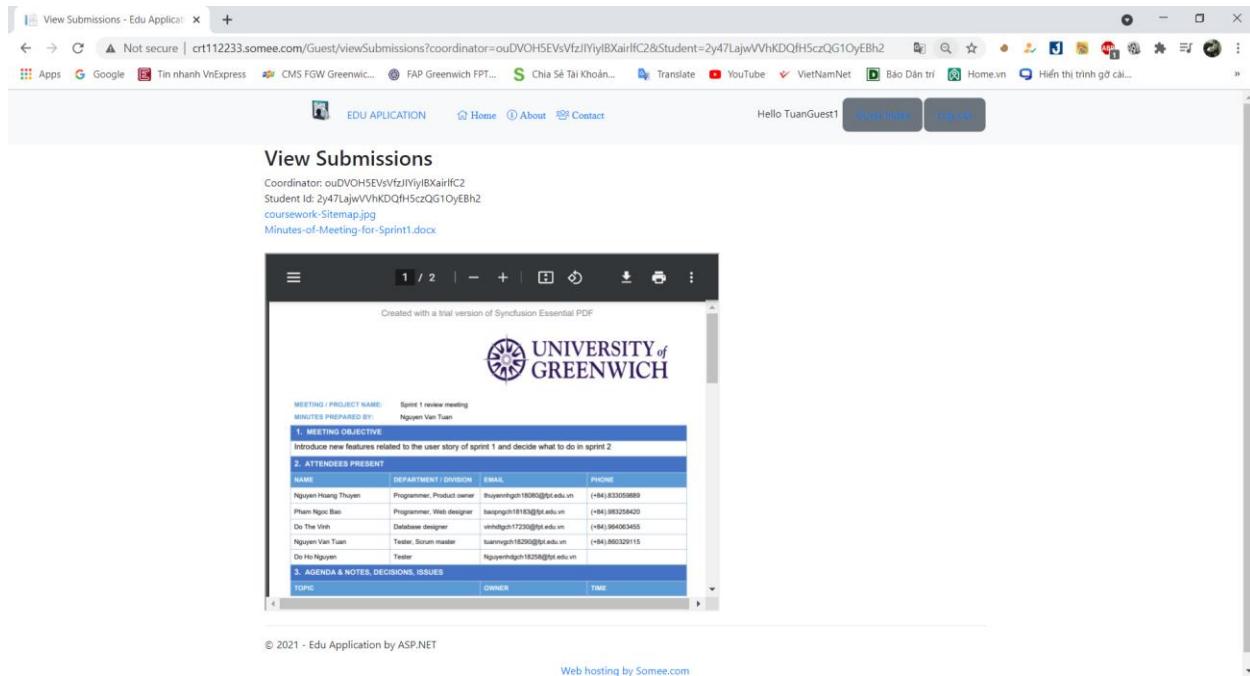


Figure 80: Interface Guest view submission

V. Implementation

1. Techniques

a. Back-end

Our project is developed based on the ASP.NET MVC 4.8 framework. This is a great framework for building and developing web applications. We see a lot of benefits that this framework brings and think it is very suitable for our project development.

When applying ASP.NET we are provided with a powerful set of Controls based on JavaScript, Html, Bootstrap, C#. Very superior when compared to the Controls provided by some traditional forms on the web.

ASP.NET MVC is very testable. We can run testing while building the functions. This is very suitable for working under the Scrum model with Agile method.

ASP.NET MVC is divided into 3 separate parts: Model, View, Control, so it shows great benefits in programming and object analysis. Because it is divided into separate parts, for faster web application development, easy to fix if there is a problem without affecting other components of the system. This also makes it easier to maintain

b. Front-end

We use Bootstrap, combined with Html and CSS to build interfaces for our website. Bootstrap is increasingly used in website design thanks to its advantages. We can quickly create the interface using Bootstrap's built-in components, which can help me eliminate repetitive time when creating Class of CSS, or lines code Html.

The bootstrap platform has been well optimized. In bootstrap has created a library to store that designers can use and customize according to personal purpose. This makes the development of our website much faster because I can choose a suitable template and customize it to have a beautiful and elegant interface right away. What's more, bootstrap is made and used so much that the compatibility with browsers and devices like phones and tablets has been tested many times so I absolutely cannot worry about website creation. Output will not be displayed as desired on different device types.

c. Database

To deploy our projects, we use the Firebase platform and Firebase available services such as Firebase Authentication, Firebase Realtime Database, and Firebase Storage. Firebase enables the development of fast apps without wasting time, manpower to manage systems, and back-end infrastructure: Firebase includes features such as analytics, databases, activity reporting, and reporting. bug reports so that we can develop and navigate our applications more easily.

Realtime database: This service allows you to store and synchronize data in real time. Applications that use this feature can store and retrieve data, as well as read and write data from the server, in a very short period of time, resulting in high efficiency when in use.

Firebase Authentication: Easily and securely manage users. Firebase Authenticator supports a variety of authentication methods, including email and passwords, as well as third-party providers such as Google or Facebook.

Cloud Storage: Use powerful, simple, and cost-effective object storage to store and share user-generated content such as images, audio, and video.

d. Hosting

Because we use the ASP.NET MVC framework, selecting a hosting provider that supports this framework to deploy the web application is difficult. Somee is one of the hosts which provided us with the necessities.

2. Flowcharts and features

a. Marketing Manager

View exceptional reports: It first collects all reports, then determines whether or not they are exceptional, and then prints all exceptional reports.

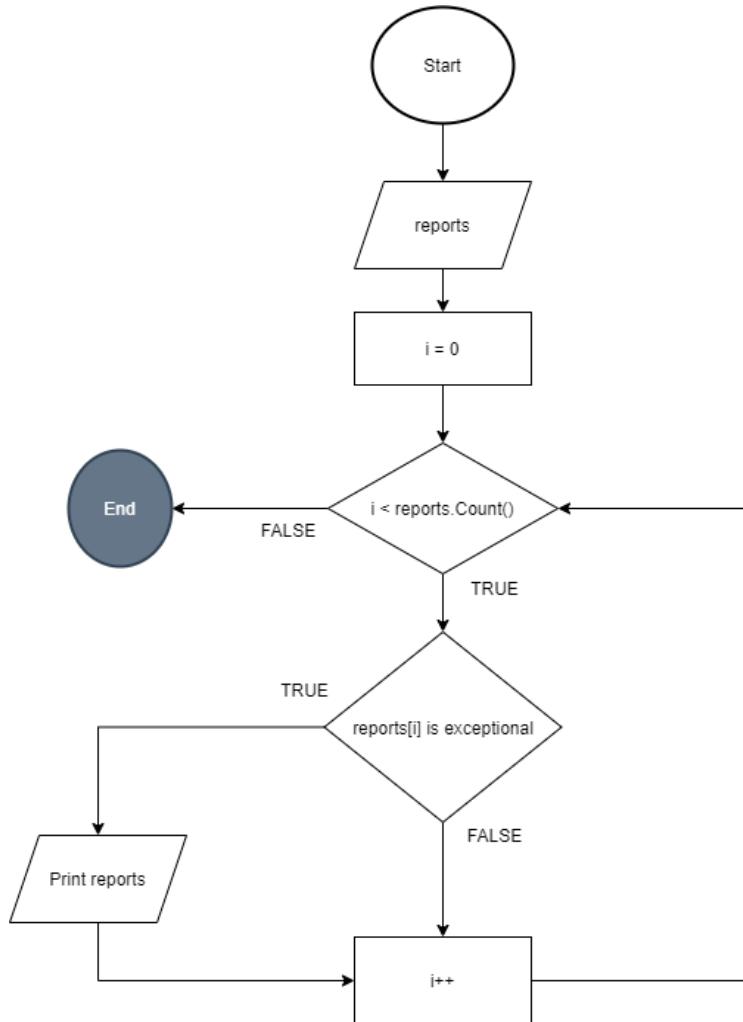


Figure 81: Flowcharts View exceptional reports

```
var response = client.Get("Mark/" + coordinator);
Dictionary<string, string> mark = JsonConvert.DeserializeObject<Dictionary<string, string>>(response.Body);
var markSort = new Dictionary<string, string>();
foreach(var item in mark)
{
    if(item.Value == "Accept")
    {
        markSort.Add(item.Key, item.Value);
    }
}

var responseComment = client.Get("Comment/" + coordinator);
var Comment = JsonConvert.DeserializeObject<Dictionary<string, string>>(responseComment.Body);

var b = new Dictionary<string, List<string>>();
foreach (var a in markSort)
{
    var c = new List<string>();
    c.Add(JsonConvert.DeserializeObject<string>(client.Get("Account/Student/" + a.Key + "/Email").Body));
    c.Add(a.Value);
    if (Comment != null)
    {
        if (Comment.ContainsKey(a.Key))
        {
            c.Add(Comment[a.Key]);
        }
    }
    b.Add(a.Key, c);
}
```

Figure 82: Features View exceptional reports

- View all selected submissions from every faculty (cannot edit): It first collects all reports, then prints all exceptional reports

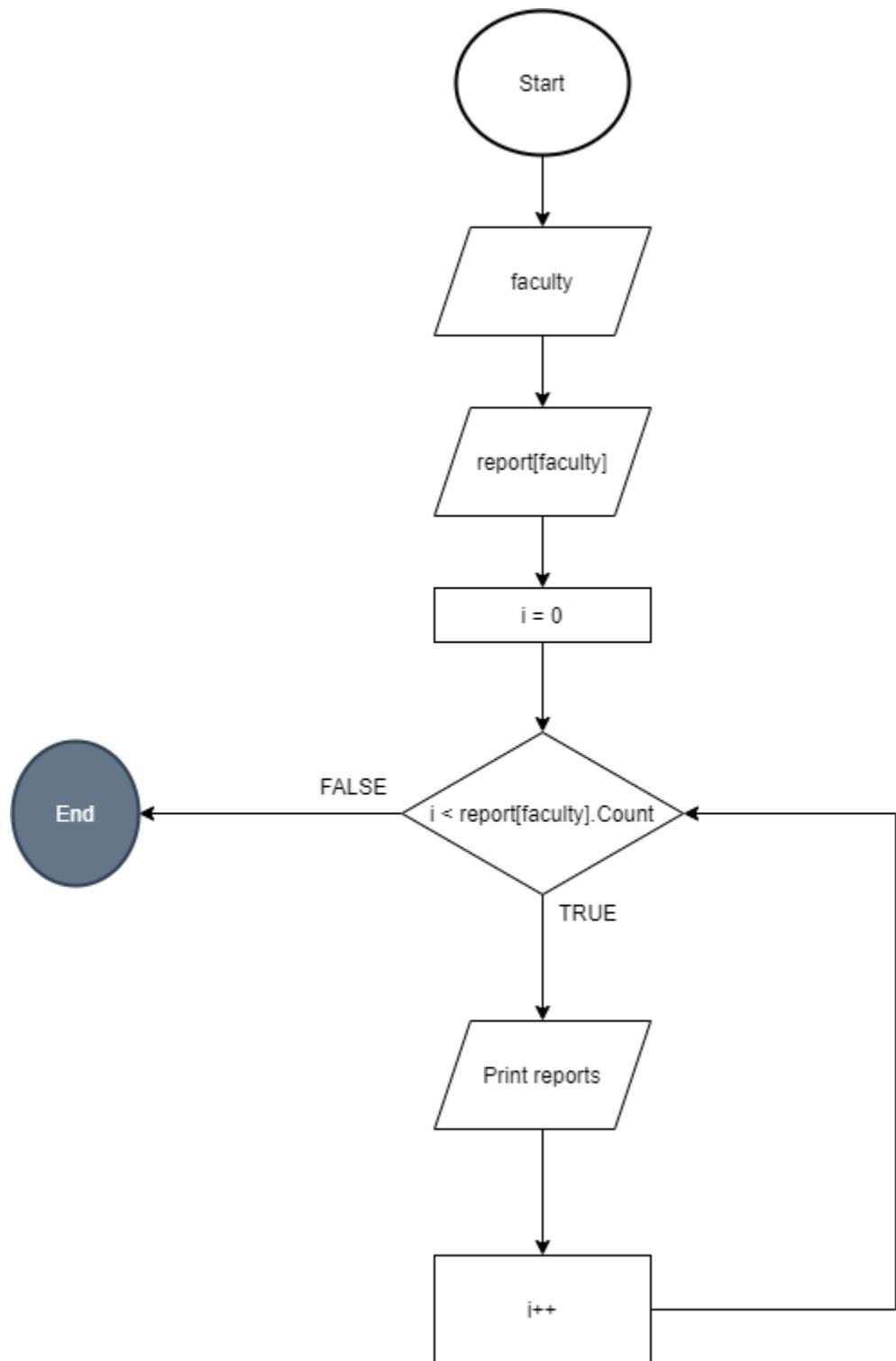


Figure 83: Flowcharts View all selected submissions

```
FirebaseResponse response = client.Get("Link/" + coordinator + "/" + Student);
var prinicpal = (ClaimsPrincipal)Thread.CurrentPrincipal;
var token = prinicpal.Claims.Where(c => c.Type == "Token").Select(c => c.Value).SingleOrDefault();
List<string> nameFile = new List<string>();
List<string> link = new List<string>();
if (response.Body != "null")
{
    List<string> a = JsonConvert.DeserializeObject<List<string>>(response.Body);
    foreach (var item in a)
    {
        var task = await new FirebaseStorage(Bucket, new FirebaseStorageOptions
        {
            AuthTokenAsyncFactory = () => Task.FromResult(token),
            ThrowOnCancel = true
        }).Child("Student submit").Child(coordinator).Child(Student).Child(item).GetDownloadUrlAsync();
        link.Add(task);
        nameFile.Add(item);
    }
}
```

Figure 84: Features View all selected submissions

- Download submissions by ZIP format: First, it gathers all of this student's documents, zips them, and then prints a connection to download them.

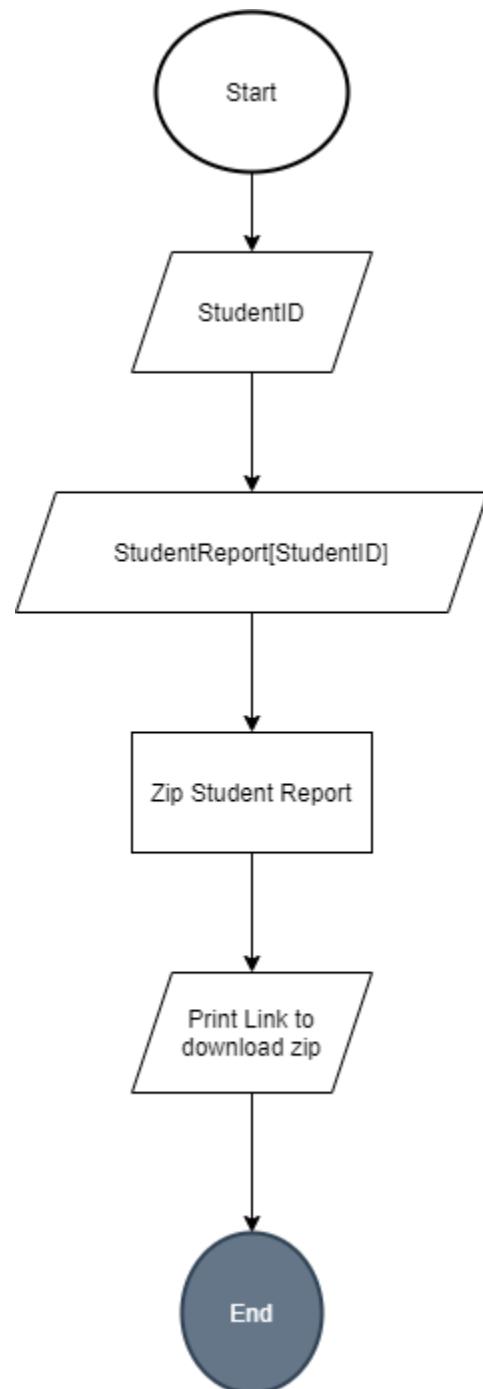


Figure 85: Flowcharts Download submissions by ZIP format

```
string[] filePaths = Directory.GetFiles(Server.MapPath("~/Content\\images\\" + sid));

List<FileModel> files = new List<FileModel>();
foreach (string filePath in filePaths)
{
    files.Add(new FileModel())
    {
        FileName = Path.GetFileName(filePath),
        FilePath = filePath
    });
}
using (ZipFile zip = new ZipFile())
{
    zip.AlternateEncodingUsage = ZipOption.AsNecessary;
    zip.AddDirectoryByName("Files");
    foreach (FileModel file in files)
    {
        zip.AddFile(file.FilePath, "Files");
    }
    string zipName = String.Format("Zip_{0}.zip", DateTime.Now.ToString("yyyy-MMM-dd-HHmss"));
    using (MemoryStream memoryStream = new MemoryStream())
    {
        zip.Save(memoryStream);
        return File(memoryStream.ToArray(), "application/zip", zipName);
    }
}
```

Figure 86: Features Download submissions by ZIP format

- View statistics: It will retrieve all reports then calculate the submission of this course vs. all courses.

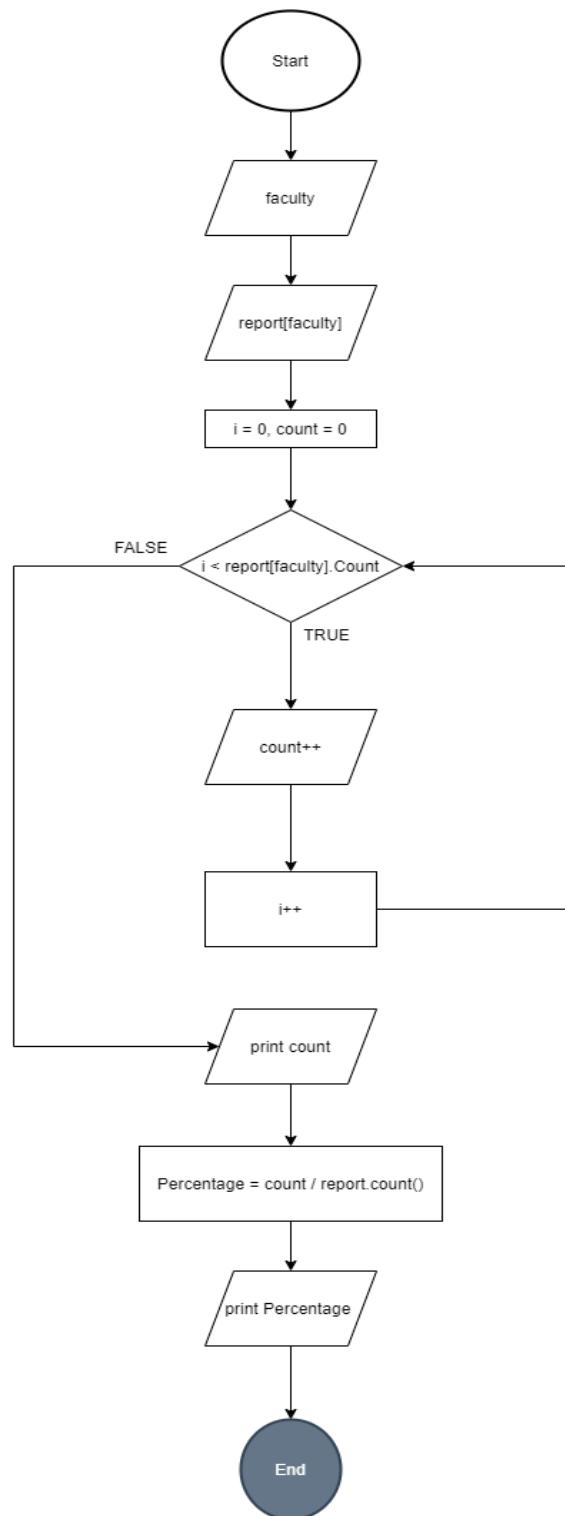


Figure 87: Flowcharts View statistics

b. Marketing Coordinator

View submissions from students in specific faculty: The Marketing Coordinator can only view the submission and cannot edit it. The system searches through all students in the course that the Marketing Coordinator teaches to view and then prints it all.

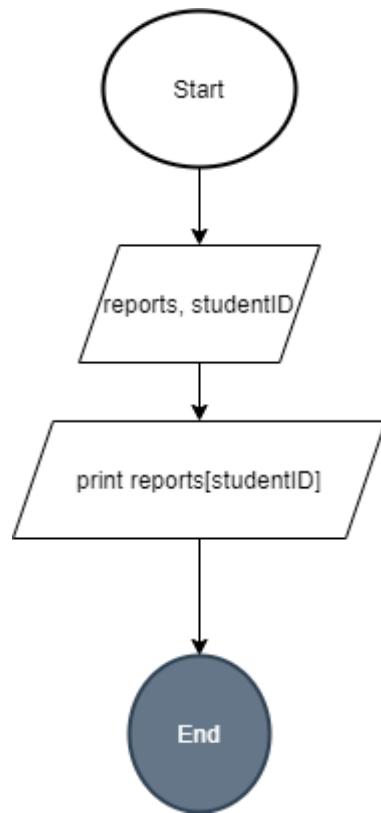


Figure 88: Flowcharts View submissions from students in specific faculty

```
List<string> nameFile = new List<string>();
List<string> link = new List<string>();
if (response.Body != "null")
{
    List<string> a = JsonConvert.DeserializeObject<List<string>>(response.Body);
    foreach (var item in a)
    {
        var task = await new FirebaseStorage(Bucket, new FirebaseStorageOptions
        {
            AuthTokenAsyncFactory = () => Task.FromResult(token),
            ThrowOnCancel = true
        }).Child("Student submit").Child(sid).Child(id).Child(item).GetDownloadUrlAsync();
        link.Add(task);
        nameFile.Add(item);
    }
}
```

Figure 89: Features View submissions from students in specific faculty

Receive notifications (emails) for new submissions: After the student submits, the Marketing Coordinator will receive the mail and ask the Marketing Coordinator to comment and mark the submission.

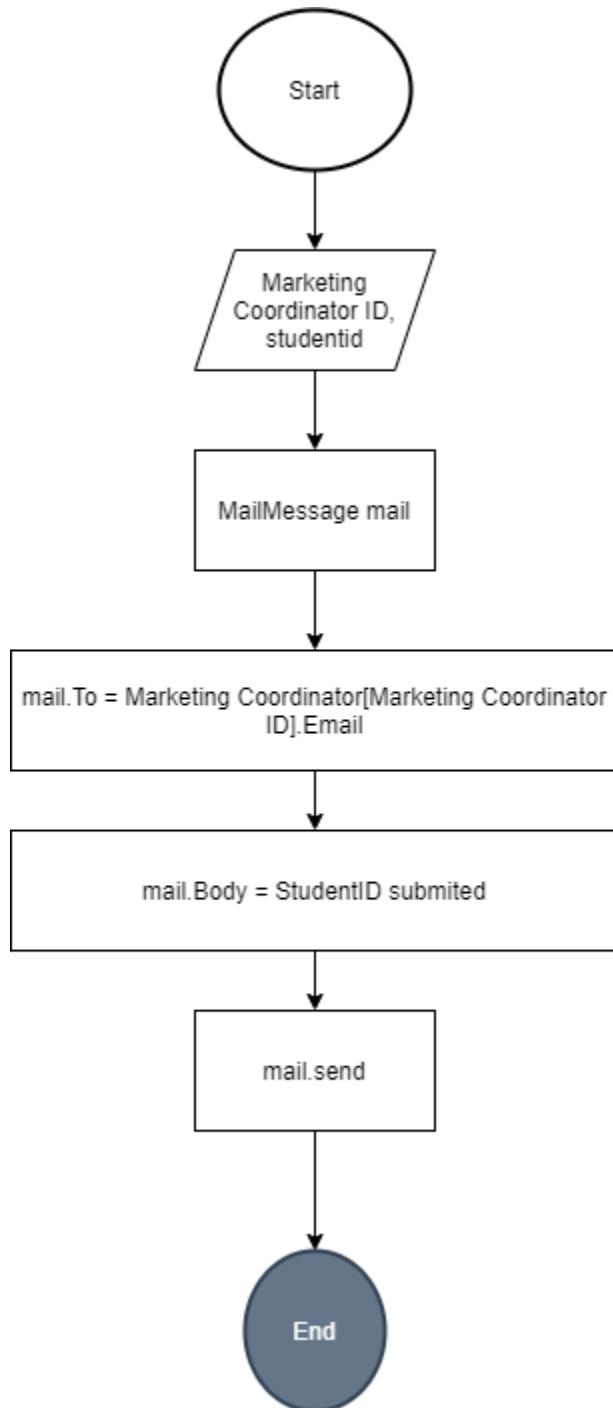


Figure 90: Flowcharts Receive notifications (emails) for new submissions

```
var body = "<p>Email From: {0} ({1})</p><p>Message:</p><p>{2}</p>";
var message = new MailMessage();
message.To.Add(new MailAddress(JsonConvert.DeserializeObject<string>(client.Get("Account/Marketing Coordinator/" + coordinator + "/Email/").Body)));
message.From = new MailAddress("thuyenprovjp@outlook.com.vn"); // replace with valid value
message.Subject = JsonConvert.DeserializeObject<string>(client.Get("Account/Marketing Coordinator/" + coordinator + "/Email/").Body) + " has given submissions to get feedback";
message.Body = string.Format(body, "Donotreply", "thuyenprovjp@outlook.com.vn", "Student " + sid + " submitted");
message.IsBodyHtml = true;

using (var smtp = new SmtpClient())
{
    var credential = new NetworkCredential
    {
        UserName = "thuyen@outlook.com.vn",
        Password = "provjp112233"
    };
    smtp.Credentials = credential;
    smtp.Host = "smtp-mail.outlook.com";
    smtp.Port = 587;
    smtp.EnableSsl = true;
    await smtp.SendMailAsync(message);
}
```

Figure 91: Features Receive notifications (emails) for new submissions

- Give comments for each submission: Marketing Coordinator can comment submission of student in 14 days after end date. System check date now less tan date end. If not, the Marketing Coordinator may comment; otherwise, the Marketing Coordinator may not comment.

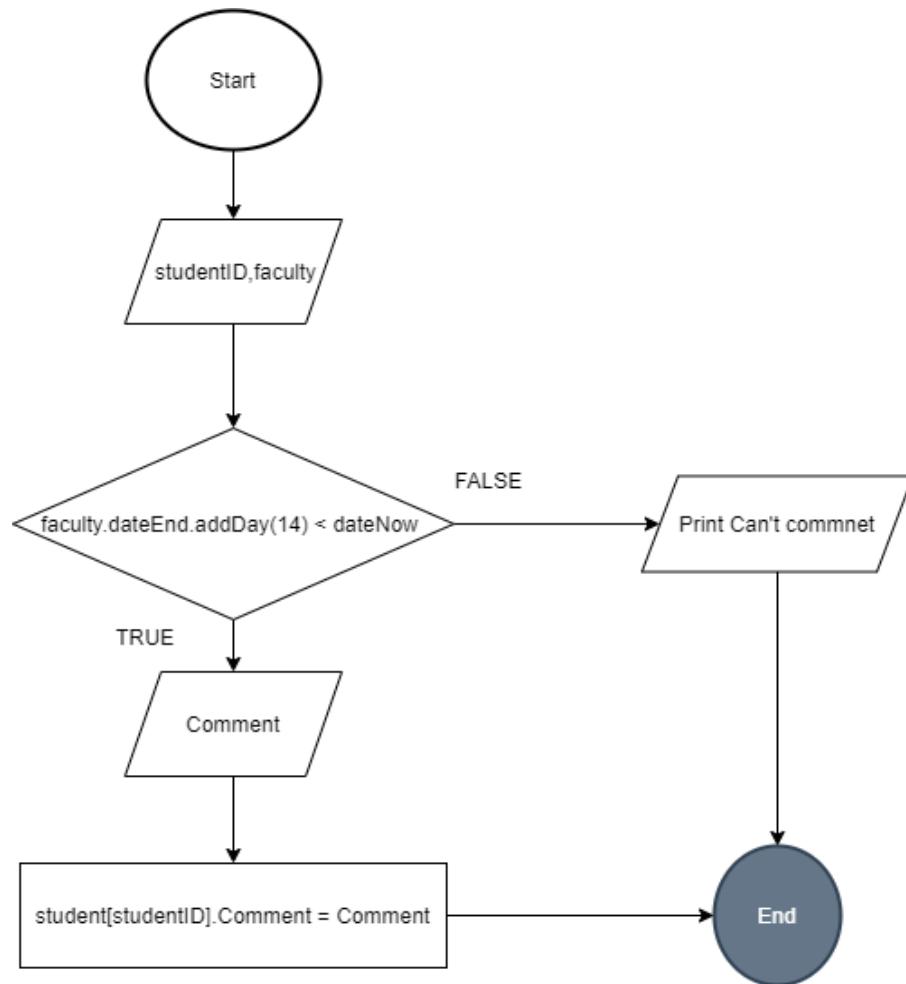


Figure 92: Flowcharts Give comments for each submission

```

var studentGrade = new Dictionary<string, string>();
studentGrade.Add(id, grade);

var studentComment = new Dictionary<string, string>();

studentComment.Add(id, comment);
await client.UpdateAsync("Mark/" + sid, studentGrade);
await client.UpdateAsync("Comment/" + sid, studentComment);
  
```

Figure 93: Feature Give comments for each submission

- Determine to accept or reject submissions: Marketing Coordinator mark student by Accept and Reject. Marketing Coordinator input Accept or Reject, then system save to database.

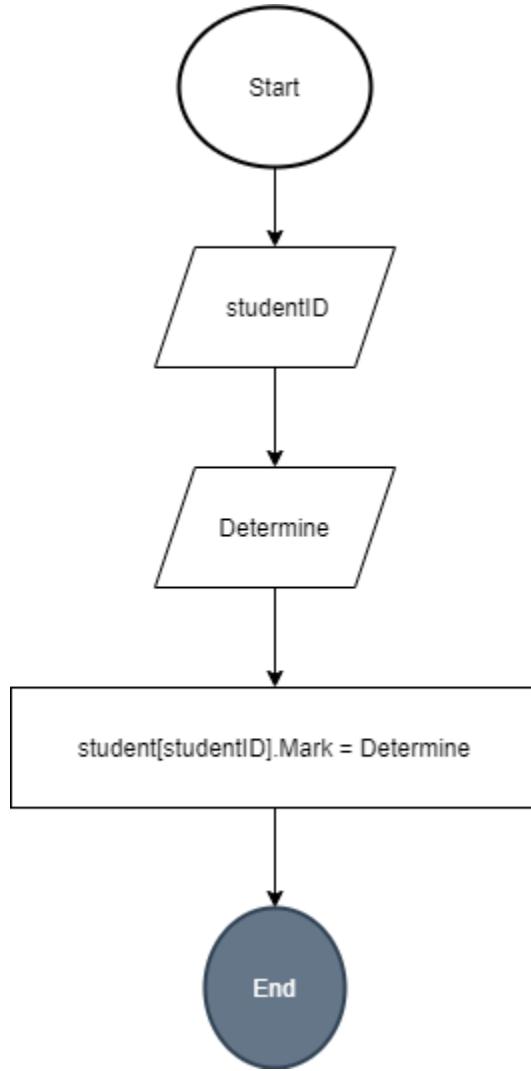


Figure 94: Flowcharts Determine to accept or reject submissions

```

var studentGrade = new Dictionary<string, string>();
studentGrade.Add(id, grade);

var studentComment = ...Dictionary<string, string>();

studentComment.Add(id, comment);
await client.UpdateAsync("Mark/" + sid, studentGrade);
await client.UpdateAsync("Comment/" + sid, studentComment);
  
```

Figure 95: Feature Determine to accept or reject submissions

c. Student

Submit article in Word format (1 or more) before first closure date, can edit before final closure date (after first closure date) and edit submissions. System check date now is less than date end, if not student can't submit, else system checks this student Submit before date end and check date now less than date final. If student had submitted before date end and date now more than date end, system not allow student edit submission. If student had submitted before date end and date now less than date end, system allow student edit submission, system allow student edit submission. If student had not submitted before date end, system not allow student edit submission

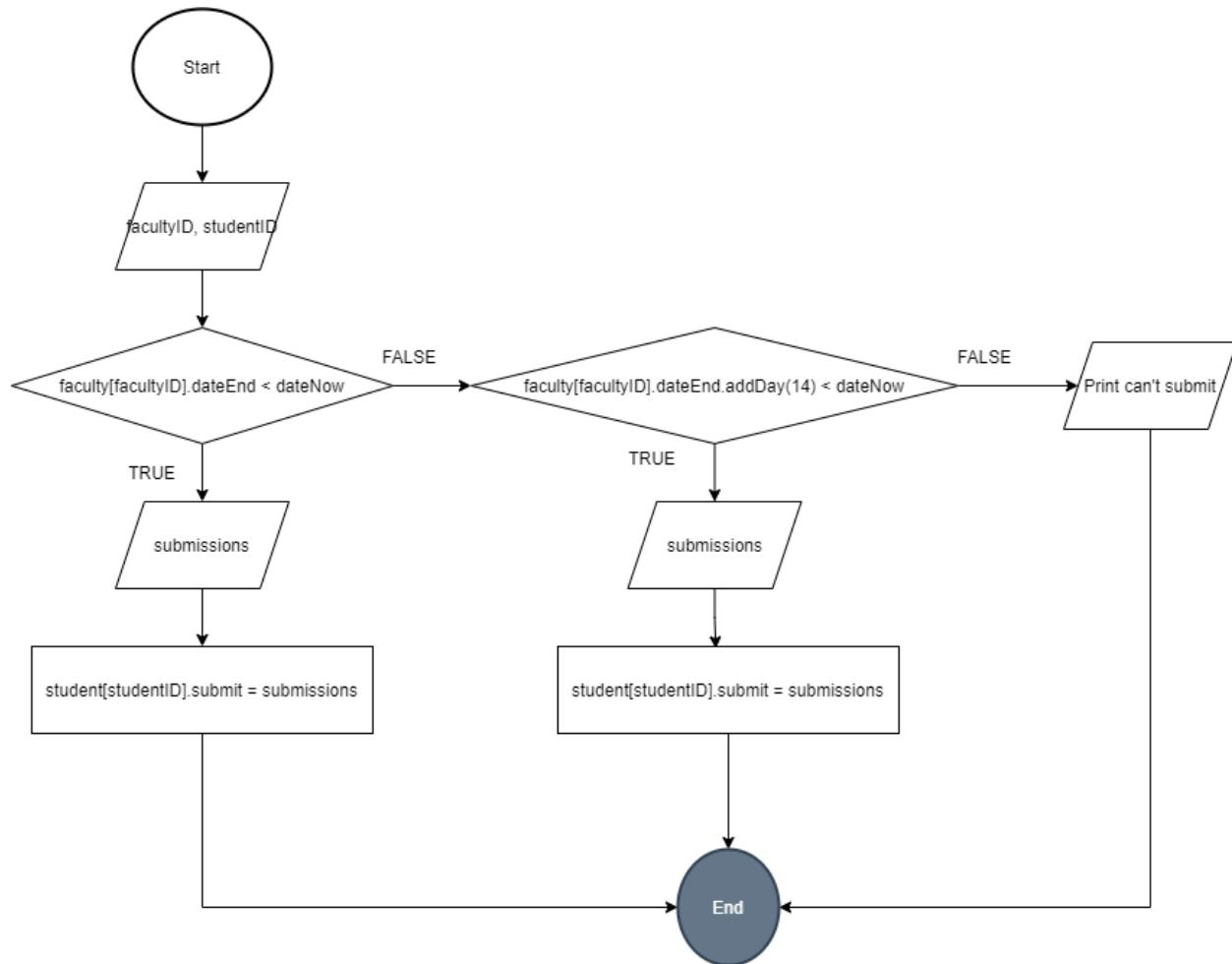


Figure 96: Flowchart Submit article in Word format (1 or more) before first closure date, can edit before final closure date (after first closure date)

```

var dateEnd = JsonConvert.DeserializeObject<Course>(response.Body).dateEnd;
var dateNow = DateTime.Now;
if (dateEnd > dateNow)
{
    ViewData["canSubmit"] = true;
}
else
{
    ViewData["canSubmit"] = false;
}
response = client.Get("Link/" + coordinator + "/student");
List<string> student = JsonConvert.DeserializeObject<List<string>>(response.Body);

if (student != null)
{
    if (dateEnd.AddDays(14) > dateNow && student.IndexOf(sid) != -1)
    {
        ViewData["canSubmit"] = true;
    }
}

```

Figure 97: Feature Submit article in Word format (1 or more) before first closure date, can edit before final closure date (after first closure date)

- Upload images: Students can submit high-quality images. When a student uploads an image, the system retrieves it and saves it to the database.

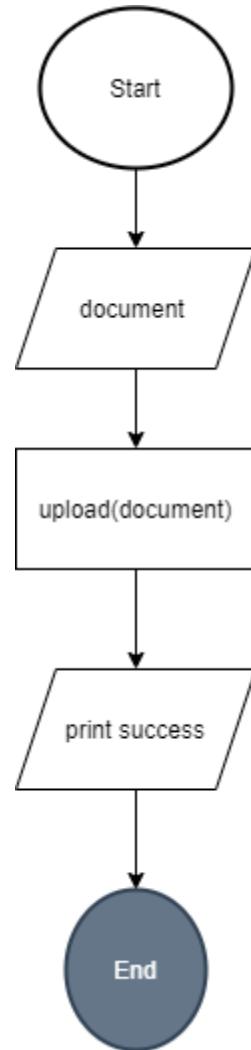


Figure 98: Flowchart Upload images

```

await new FirebaseStorage(Bucket, new FirebaseStorageOptions
{
    AuthTokenAsyncFactory = () => Task.FromResult(document.Token),
    ThrowOnCancel = true
}).Child("Student submit").Child(coordinator).Child(sid).Child(name).PutAsync(stream);
  
```

Figure 99: Feature Upload images

- Read comments for own submissions: Students can read the comments that the Marketing Coordinator has written about them. If the student selects read comment, the system will print the comment.

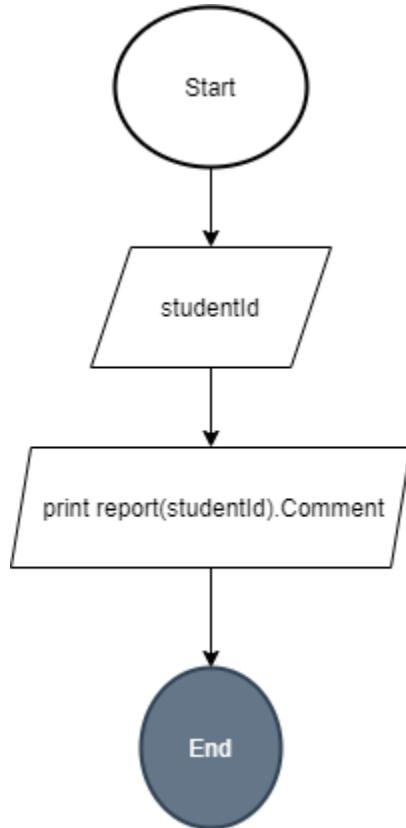


Figure 100: Flowchart Read comments for own submissions

```

var response = client.Get("Mark/" + item.Coordinator + "/" + sid);
var mark = JsonConvert.DeserializeObject<string>(response.Body);

var responseComment = client.Get("Comment/" + item.Coordinator + "/" + sid);
var Comment = JsonConvert.DeserializeObject<string>(responseComment.Body);
  
```

Figure 101: Feature Read comments for own submissions

d. Admin

Create new accounts: Admin has the ability to create accounts for all roles, including students, guests, marketing coordinators, and marketing managers. Admin enters an account, and the system determines whether it exists or not; if not, the system creates the account; otherwise, the system prints that it exists.

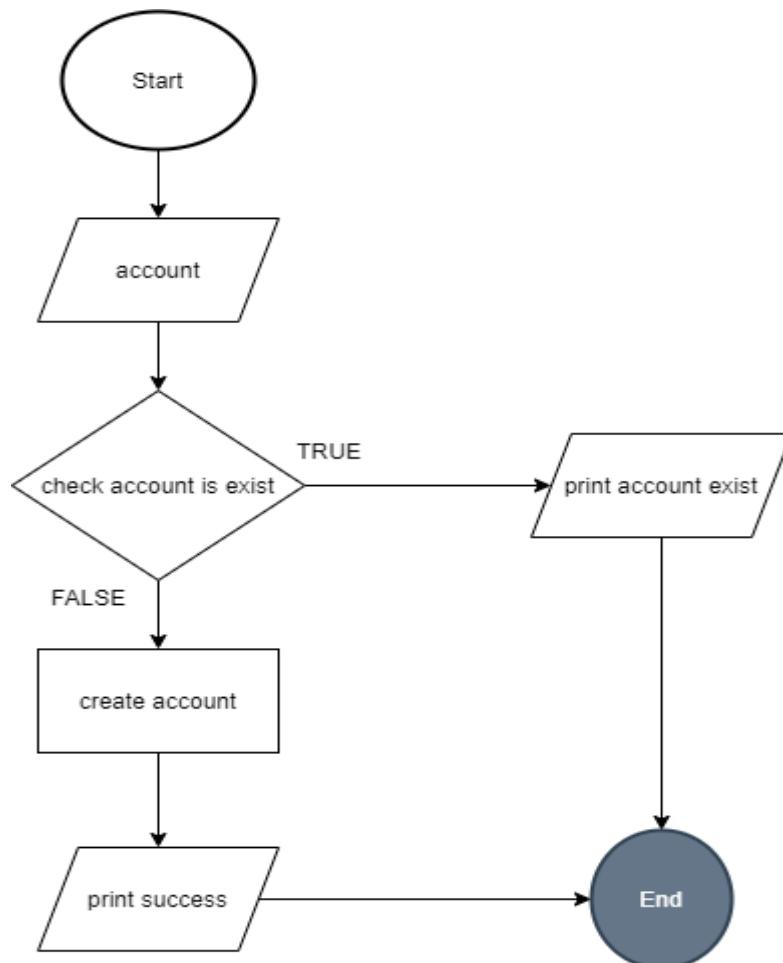


Figure 102: Flowchart Create new accounts

```
var auth = new FirebaseAuthProvider(new FirebaseConfig(ApiKey));  
  
var a = await auth.CreateUserWithEmailAndPasswordAsync(model.Email, model.Password, model.Name, true);  
  
client = new FireSharp.FirebaseClient(config);  
  
model.id = a.User.LocalId;  
SetResponse setResponse = client.Set("Account/" + model.role + "/" + model.id, model);
```

Figure 103: Feature Create new accounts

- Set and edit closure date: All courses' end dates and final dates can be set/edited by the administrator. Admin input date end date final, If the date final is less than the date end, the date final will be the date end.

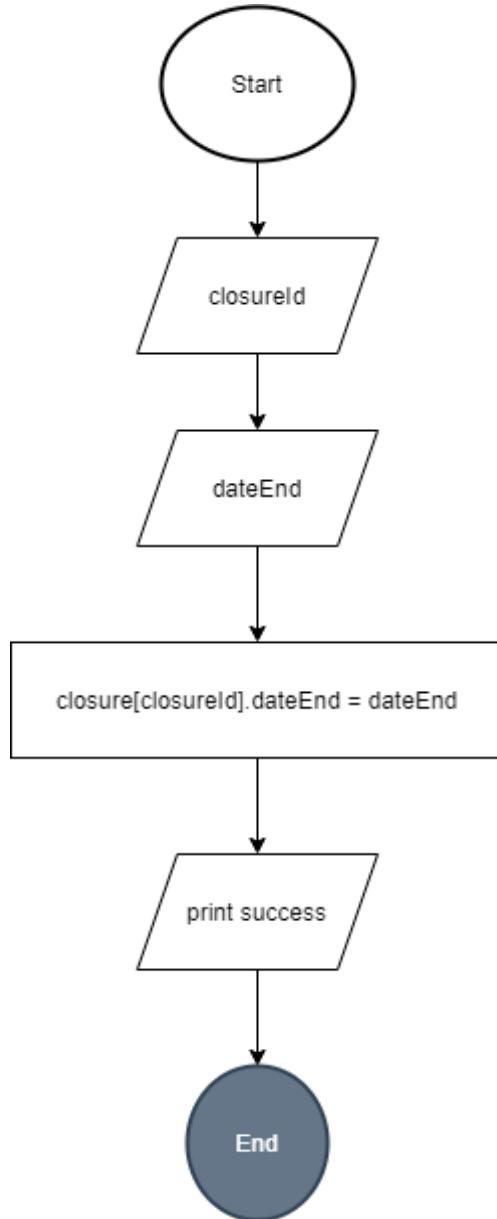


Figure 104: Flowchart Set and edit closure date

```
if (collection.dateEnd == DateTime.Parse("1/1/0001 12:00:00 AM"))
{
    FirebaseResponse response = await client.GetAsync("Editorial/" + collection.Coordinator);
    Course data = JsonConvert.DeserializeObject<Course>(response.Body);
    collection.dateEnd = data.dateEnd;
}

await client.SetAsync("Course/" + collection.Coordinator, collection);
```

Figure 105: Feature Set and edit closure date

e. Guest

View selected reports for each faculty: Guest can view report have accepted of course that Admin join guest in. System get course that this guest in then print all the report have accepted

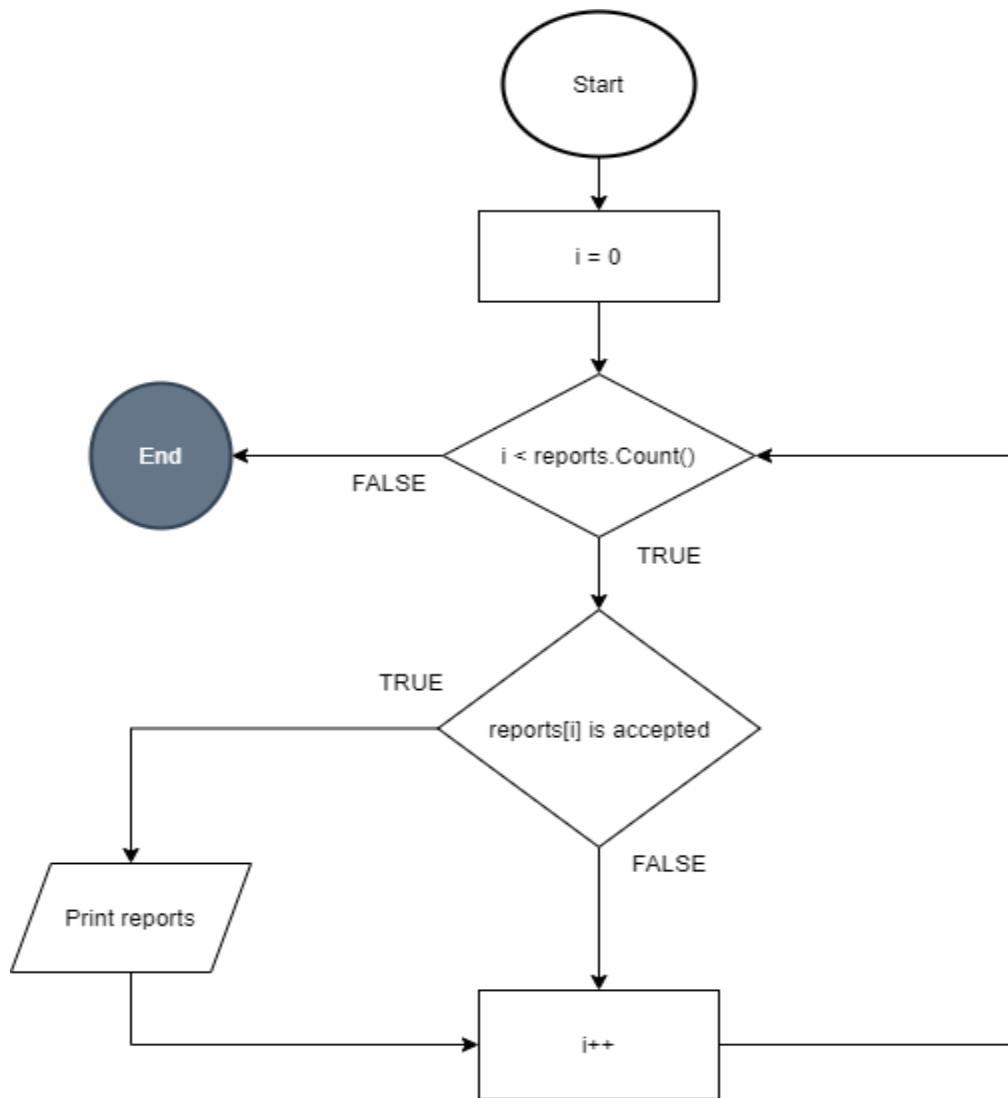


Figure 106: Flowchart View selected reports for each faculty

```
List<string> nameFile = new List<string>();
List<string> link = new List<string>();
if (response.Body != "null")
{
    List<string> a = JsonConvert.DeserializeObject<List<string>>(response.Body);

    foreach (var item in a)
    {
        var task = await new FirebaseStorage(Bucket, new FirebaseStorageOptions
        {
            AuthTokenAsyncFactory = () => Task.FromResult(token),
            ThrowOnCancel = true
        }).Child("Student submit").Child(coordinator).Child(Student).Child(item).GetDownloadUrlAsync();
        link.Add(task);
        nameFile.Add(item);
    }
}
```

Figure 107: Feature View selected reports for each facult

VI. Testing

1.1. Test plan

1.1.1. Team members

Name	Role
Nguyen Hoang Thuyen	Programmer, Information architect, Product owner
Pham Ngoc Bao	Web designer, Programmer
Do The Vinh	Database designer, Web designer
Ha Ngoc Vy	Web designer
Nguyen Van Tuan	Scrum master, Tester
Do Ho Nguyen	Tester

Table 3: Team members and Roles

1.1.2. Scope

The functions tested in this project are:

General functions:

- Log in.
- Log out.

Specific functions:

- **For Admin account:**
 - Create and edit update time, closing time.
 - Create, edit and delete account.
 - Storing and preserving all system data.
- **For Student account:**
 - Send articles as word documents.
 - Upload high-quality images.
 - Read comments for own submissions.
 - Update contributions until a final closure date.
 - Accept the terms and conditions that require approval before submitting articles.
- **For Marketing Coordinator account:**

- Receive email notifications from the system whenever the students in faculty publish their articles.
 - Make comments for student's articles within 14 days after the system stops accepting new articles.
 - View contributions from students in faculty.
 - Interact with the students in faculty.
 - Select contributions to publish.
- **For Marketing Manager account:**
- View all the selected contributions.
 - View the exceptional reports.
 - Download all selected contributions as zip file.
 - Keep track of the number of contributions, contributors and the percentage of contributions by multiple faculties.
- **For Guest account:**
- View faculty selected reports.

1.1.3. Test Approach

Our project uses an agile approach, with iterations daily and after each sprint. Whenever functions or requirements are made for that sprint, they will be delivered and immediately checked by testers.

Our team also chose manual testing as a method for web application testing and development. More specifically, our manual testing is performed in a sequence of steps: running the web application on the server, testing each function with as many assumptions as possible, writing error reports and send to the coder, check the functions after being fixed, test all the functions passed previously again; if an error occurs, the tester will continue to write the error report, send it to the coder and continue to test until all the functions have been passed. Finally, write test cases, test logs for all cases tested in the previous process.

1.1.4. Test Environment

The web application, the database and the server to publish web application are required.

➤ Web application:

 - Back-end: ASP.NET MVC 4.8, JavaScript, C#.
 - Front-end: bootstrap 4, css, html.

➤ Database: Firebase Authentication, Firebase Realtime Database, Firebase Storage.

➤ Server: Somee.

1.1.5. Assumptions/Risks

1.1.5.1. Assumptions

1. Delays in the Sprints may cause the team to increase the working time of the day.
2. Lack of experience, knowledge in developing and testing the system can make the team spend more time doing and researching.
3. The lack of manpower may cause some team members to take on more tasks.
4. Misinterpreting the requirements of the system can cause the team to spend more time and effort on editing, testing, etc.
5. Having more members or having any member leave the team during project implementation may cause the team to rearrange many things from the beginning.

1.1.5.2. Risks

Below are some of the risks that can be encountered during our team's project implementation.

	Risk	Impact	Trigger	Mitigation Plan
1	Modifications, changes to the system's functionality can make the test cases false and may have to be revised a lot	Medium	The test case deviated too much	Gather human resources, time to discuss system requirements before starting a sprint, and constantly cross-monitor among members during system development
2	Functions become unstable or not function properly as originally achieved, after the successive stages of system development	High	Too much faulty functionality is neglected during development and testing	Perform a rigorous testing process multiple times after each sprint and retest all functionality included in the old sprint when testing new functionality implemented in the new sprint
3	Functions in sprint have execution time deviations that prevent the system from completing on schedule	High	Too many stages are delayed in project development	Continuously monitor the system development process, identify difficulties and obstacles that members face and find ways to overcome them as effectively as possible

Table 4:Risk

1.1.6. Test schedule

Task Name	Start	Finish	Effort
Test Planning	17/2	19/2	2 day
Functional testing - Sprint 1	15/2	26/2	12 day
Functional testing - Sprint 2	1/3	12/3	12 day
Functional testing - Sprint 3	15/3	26/3	12 day
System test	27/3	27/3	1 day
Regression testing	28/3	28/3	1 day
UAT (User Acceptance Testing)			
Solve final defects and check the entire system one last time	29/3	29/3	1 day
Performance testing			

Table 5: Test schedule

1.1.7. Deliverables

The following are the test deliverables:

1. Test Plan
2. Test Case and Test Log
3. Summary Report

1.2. Sprint 1

1.2.1. Test Cases and Test Logs of Sprint 1

Below is a table showing the test cases and test logs of Sprint 1 which are made to test user account creation and control functionality, exclusively for the Admin account.

Test Case ID	Role	Test Scenario	Test Case	Pre-Conditions	Test Steps	Test Data	Post Conditions	Expected Result	Actual Result	Status
TC_SIGN_UP_001	Admin	Check sign up functionality exclusively for the admin account	Check response on entering Full Name, Email and Password	Requires created and enabled admin account (Active Status = True)	1. Select Log in 2. Login with the enabled admin account 3. Select Admin Index 4. Select Create Admin/Create Student/ Create Marketing Manager/Create Guest 5. Enter Full Name 6. Enter Email 7. Enter Password 8. Click Create button	Full Name: TuanAdmin Email: TuanAdmin@gmail.com Password: TuanAdmin@gmail.com Full Name: TuanStudent Email: TuanStudent@gmail.com Password: TuanStudent@gmail.com Full Name: TuanMM Email: TuanMM@gmail.com Password: TuanMM@gmail.com Full Name: TuanGuest Email: TuanGuest@gmail.com Password: TuanGuest@gmail.com	The newly created account appears in the account data table on the Admin Index page	- Successfully created a new account with the entered data - For the Student and Guest accounts, in addition to the entered data, they will be created and assigned with the name of the default marketing coordinator account	Sign up successful	Pass
TC_SIGN_UP_002	Admin	Check sign up functionality exclusively for the admin account	Check response on entering Full Name, Email, Password and selecting one name from Marketing Coordinator accounts	Requires created and enabled admin account (Active Status = True)	1. Select Log in 2. Login with the enabled admin account 3. Select Admin Index 4. Select Create Student/Create Guest 5. Enter Full Name 6. Enter Email 7. Enter Password	Full Name: TuanStudent2 Email: TuanStudent2@gmail.com Password: TuanStudent2@gmail.com Marketing Coordinator: TuanMC1 Full Name: TuanGuest2 Email: TuanGuest2@gmail.com Password: TuanGuest2@gmail.com Marketing Coordinator: TuanMC1	The newly created account appears in the account data table on the Admin Index page	Successfully created a new account with the entered data, the name selected from Marketing Coordinator accounts	Sign up successful	Pass

					8. Select one name from the Marketing Coordinator accounts 9. Click Create button					
TC_SIGN_UP_003	Admin	Check sign up functionality exclusively for the admin account	Check response on entering Full Name, Course Name, Email and Password	Requires created and enabled admin account (Active Status = True)	1. Select Log in 2. Login with the enabled admin account 3. Select Admin Index 4. Select Create Marketing Coordinator 5. Enter Full Name 6. Enter Course Name 7. Enter Email 8. Enter Password 9. Click Create button	Full Name: TuanMC1 Course Name: IT Email: TuanMC1@gmail.com Password: TuanMC1@gmail.com	The newly created account appears in the account data table on the Admin Index page	Successfully created a new account with the entered data	Sign up successful	Pass
TC_SIGN_UP_004	Admin	Check sign up functionality exclusively for the admin account	Check response when not entering Full Name, Email and Password	Requires created and enabled admin account (Active Status = True)	1. Select Log in 2. Login with the enabled admin account 3. Select Admin Index 4. Select Create Admin/Create Student/ Create Marketing Coordinator/Create Marketing Manager/Create Guest 5. Click Create button	Name: (Course Name: Filled for Create Marketing Coordinator) Email: Password: (Marketing Coordinator: The names of the Marketing Coordinator account available in the system for Create Student and Create Guest)	The program sends a message asking users to enter data for Full Name, Email and Password	Can't create an account without filling out all 3 of these boxes	There are no new accounts with missing data even in 1 of 3 data fields	Pass

TC_SIGN_UP_005	Admin	Check sign up functionality exclusively for the admin account	Check response when not entering Course Name	Requires created and enabled admin account (Active Status = True)	<ol style="list-style-type: none"> 1. Select Log in 2. Login with the enabled admin account 3. Select Admin Index 4. Select Create Marketing Coordinator 5. Enter Full Name 6. Enter Email 7. Enter Password 8. Click Create button 	Full Name: TuanMC2 Course Name: Email: TuanMC2@gmail.com Password: TuanMC2@gmail.com	When the Create button is clicked, the mouse cursor is returned to the Course Name text box	Can't create marketing coordinator account when the Course Name text box is left blank while the Course Name text box is still blank	The create button cannot be used to create a new marketing coordinator account	Pass
TC_SIGN_UP_006	Admin	Check sign up functionality exclusively for the admin account	Check response when don't have any name from Marketing Coordinator accounts to select	Requires created and enabled admin account (Active Status = True)	<ol style="list-style-type: none"> 1. Select Log in 2. Login with the enabled admin account 3. Select Admin Index 4. Select Create Student/Create Guest 5. Enter Full Name 6. Enter Email 7. Enter Password 8. Click Create button 	Full Name: TuanStudent2 Email: TuanStudent2@gmail.com Password: TuanStudent2@gmail.com Marketing Coordinator:	When the Create button is clicked, the mouse cursor is returned to the marketing coordinator account name selection box	Student and guest accounts cannot be created without any marketing coordinator account name in the selection box	The create button cannot be used to create a new student account and guest account while the marketing coordinator account name selection box has no name	Pass

									to choose from	
TC_LOGIN_01	Admin	Check login functionality	Check response when logging in with email and password of the created and enabled admin account	Requires created and enabled admin account (Active Status = True)	1. Select Log in 2. Login with the enabled admin account	Email: TuanAdmin@gmail.com Password: TuanAdmin@gmail.com	The home page of admin accounts appears	Log in successful	Account is logged in successfully	Pass
TC_LOGIN_02	Student	Check login functionality	Check response when logging in with email and password of the created and enabled student account	Requires created and enabled student account (Active Status = True)	1. Select Log in 2. Login with the enabled student account	Email: TuanStudent@gmail.com Password: TuanStudent@gmail.com	The home page of student accounts appears	Log in successful	Account is logged in successfully	Pass

TC_LOGIN_03	Marketing Coordinator	Check login functionality	Check response when logging in with email and password of the created and enabled marketing coordinator account	Requires created and enabled marketing coordinator account (Active Status = True)	1. Select Log in 2. Login with the enabled marketing coordinator account	Email: TuanMC1@gmail.com Password: TuanMC1@gmail.com	The home page of marketing coordinator accounts appears	Log in successful	Account is logged in successfully	Pass
TC_LOGIN_04	Marketing Manager	Check login functionality	Check response when logging in with email and password of the created and enabled marketing manager account	Requires created and enabled marketing manager account (Active Status = True)	1. Select Log in 2. Login with the enabled marketing manager account	Email: TuanMM@gmail.com Password: TuanMM@gmail.com	The home page of marketing manager accounts appears	Log in successful	Account is logged in successfully	Pass
TC_LOGIN_05	Guest	Check login functionality	Check response when logging in with email and password of the created and enabled guest account	Requires created and enabled guest account (Active Status = True)	1. Select Log in 2. Login with the enabled guest account	Email: TuanGuest@gmail.com Password: TuanGuest@gmail.com	The home page of guest accounts appears	Log in successful	Account is logged in successfully	Pass

TC_LOGIN_06	Enabled account	Check login functionality	Check response when logging in with the email entered incorrectly and the password entered correctly	Requires created and enabled account (Active Status = True)	1. Select Log in 2. Login with any enabled account	Email: Tuan@gmail.com Password: TuanMC1@gmail.com	The program sends a message that the user has entered an incorrect username or password	Login failed	Fail to login	Pass
TC_LOGIN_07	Enabled account	Check login functionality	Check response when logging in with the email entered correctly and the password entered incorrectly	Requires created and enabled account (Active Status = True)	1. Select Log in 2. Login with any enabled account	Email: TuanMC1@gmail.com Password: 123456	The program sends a message that the user has entered an incorrect username or password	Login failed	Fail to login	Pass
TC_LOGIN_08	None	Check login functionality	Check response when logging in without entering Email and Password	None	1. Select Log in 2. Click Sign In button	None	The program sends a message asking users to enter data for Email and Password	Login failed	Fail to login	Pass
TC_LOGIN_09	Unenabled account	Check login functionality	Check response when logging in with an unenabled account	Requires created and unenabled account (Active Status = False)	1. Select Log in 2. Login with any unenabled account	Email: AAA@gmail.com Password: AAA@gmail.com	The program sends a message saying that this account has been unenabled	Login failed	Fail to login	Pass

TC_LOG_OUT T_001	Enabled account	Check logout functionality	Check response when logging out account	Requires account that successfully logged in	1. Select Log in 2. Login with any enabled account 3. Select Log out	None	The Login page for account login is displayed	Log out successful	Account is logged out successfully	Pass
TC_EDIT_001	Admin	Check edit functionality exclusively for the admin account	Check response when editing the name of the account	- Requires admin account that successfully logged in - Requires account which needs to be edited	1. Select Log in 2. Login with the enabled admin account 3. Select Admin Index 4. Select Edit 5. Edit Name 6. Click Save button	Name: TuanMC1 (New Name: TuanMC123)	- The account management page of the admin account that has just made modifications to another account is displayed - The new name of the newly edited account appears in the account data table containing that account	Successfully edited the name of the account	The name of the account was successfully edited	Pass
TC_EDIT_002	Admin	Check edit functionality exclusively for the admin account	Check response when editing the email of the account	- Requires admin account that successfully logged in - Requires account which	1. Select Log in 2. Login with the enabled admin account 3. Select Admin Index 4. Select Edit 5. Edit Email 6. Click Save button	Email: TuanMC1@gmail.com (New Email: TuanMC123@gmail.com)	- The account management page of the admin account that has just made modifications to	Successfully edited email of the account	The email of the account was successfully edited	Pass

				needs to be edited			another account is displayed - The new email of the newly edited account appears in the account data table containing that account		
TC_EDIT_003	Admin	Check edit functionality exclusively for the admin account	Check response when editing the password of the account	- Requires admin account that successfully logged in - Requires account which needs to be edited	1. Select Log in 2. Login with the enabled admin account 3. Select Admin Index 4. Select Edit 5. Edit Password 6. Click Save button	Password: TuanMC1@gmail.com (New Password: TuanMC123)	- The account management page of the admin account that has just made modifications to another account is displayed - The new password of the newly edited account appears in the account data table containing that account	Successfully edited password of the account	The password of the account was successfully edited Pass

TC_EDIT_004	Admin	Check edit functionality exclusively for the admin account	Check response when editing the active status box of the account	- Requires admin account that successfully logged in - Require account which needs to be edited	1. Select Log in 2. Login with the enabled admin account 3. Select Admin Index 4. Select Edit 5. Edit active status box 6. Click Save button	Active Status: ✓ (True) (New Active Status: □ (False))	- The account management page of the admin account that has just made modifications to another account is displayed - The new value of the active status box of the newly edited account appears in the account data table containing that account	Successfully edited the value for the active status box of the account	The value of the active status box of the account was successfully edited
TC_EDIT_005	Admin	Check edit functionality exclusively for the admin account	Check response when editing account without re-filling for deleted values and leave blank at Full Name, Email and Password	- Requires admin account that successfully logged in - Require account which needs to be edited	1. Select Log in 2. Login with the enabled admin account 3. Select Admin Index 4. Select Edit 5. Delete Name 6. Delete Email 7. Delete Password 6. Click Save button	Name: Email: Password: Active Status: (selected both True and False)	The program sends a message asking users to enter data for Name, Email and Password	Edit failed	The account has not changed

TC_EDIT_006	Admin	Check edit functionality exclusively for the admin account	Check response when editing account with an email already in use by another account in the system	<ul style="list-style-type: none"> - Requires admin account that successfully logged in - Require account which needs to be edited 	<ol style="list-style-type: none"> 1. Select Log in 2. Login with the enabled admin account 3. Select Admin Index 4. Select Edit 5. Edit Email 6. Click Save button 	Email: TuanMC1@gmail.com (New Email: TuanStudent@gmail.com)	The account management page of the admin account is displayed with the account's email unchanged	Edit failed	The account has not changed	Pass
TC_DELETE_001	Admin	Check delete functionality exclusively for the admin account	Check response when deleting admin, student, guest and marketing manager accounts	<ul style="list-style-type: none"> - Requires admin account that successfully logged in - Require account which needs to be deleted 	<ol style="list-style-type: none"> 1. Select Log in 2. Login with the enabled admin account 3. Select Admin Index 4. Select Delete Student/Admin/Guest/Marketing Manager account 5. Click Delete button 	Name: TuânMM Email: TuanMM@gmail.com Password: TuanMM@gmail.com Role: Marketing Manager Active Status: True Name: TuanAdmin Email: TuanAdmin@gmail.com Password: TuanAdmin@gmail.com Role: Admin Active Status: True Name: TuanStudent Email: TuanStudent@gmail.com Password: TuanStudent@gmail.com Role: Student Active Status: True Name: TuanGuest Email: TuanGuest@gmail.com Password: TuanGuest@gmail.com	- The account management page of the admin account that has just deleted another account is displayed - The deleted account disappears from the account data table that contains it	Delete successful	The account has been deleted	Pass

						Role: Guest Active Status: True				
TC_DELETE_002	Admin	Check delete functionality exclusively for the admin account	Check response when deleting the marketing coordinator account without deleting the student accounts and the guest accounts it is in charge of in the faculty	- Requires admin account that successfully logged in - Require account which needs to be deleted	1. Select Log in 2. Login with the enabled admin account 3. Select Admin Index 4. Select Delete Marketing Coordinator account 5. Click Delete button	Name: TuanMC1 Email: TuanMC1@gmail.com Password: TuanMC1@gmail.com Role: Marketing Coordinator Active Status: True	The program sends a notice asking the user to delete the student and guest accounts in that faculty first	Delete failed	The account was not deleted	Pass

TC_DELETE_003	Admin	Check delete functionality exclusively for the admin account	Check response when deleting the marketing coordinator account once the student and guest accounts it is in charge of in the faculty have been deleted	<ul style="list-style-type: none"> - Requires admin account that successfully logged in - Require account which needs to be deleted 	<ol style="list-style-type: none"> 1. Select Log in 2. Login with the enabled admin account 3. Select Admin Index 4. Select Delete all Student and Guest accounts that the Marketing Coordinator account in charge of 5. Select Delete Marketing Coordinator account 5. Click Delete button 	Name: TuanMC1 Email: TuanMC1@gmail.com Password: TuanMC1@gmail.com Role: Marketing Coordinator Active Status: True	<ul style="list-style-type: none"> - The account management page of the admin account that has just deleted other accounts is displayed - The deleted accounts disappear from the account data table that contains them 	Delete successful All related accounts have been deleted	Pass
---------------	-------	--	--	---	---	--	---	---	------

Table 6: Sprint 1

Below is a table showing the test case and test log created to test student account functions such as submitting articles as doc files and images, reading comments, updating articles, accepting terms and conditions. condition.

Test	Role	Pre-Conditions	What is being test	Test data used	Expected Results	Actual result	Status
1	Student	Requires created and enabled student account (Active Status = True)	View Student Index	Click "Student Index "button	Show view Student Index successfully	Show view Student Index successfully	Pass
2	Student	Requires created and enabled student account (Active Status = True)	Chat box	Click" Chat "button	Show view chat index successfully	Show view chat index successfully	Pass
3	Student	Requires created and enabled student account (Active	View Submit File	Click "Submit" button	Show view Submit File	Show view Submit File	Pass

		Status = True)					
4	Student	Requires created and enabled student account (Active Status = True)	Add file	Click "Choose file" button	Add successfully	Add successfully	Pass
5	Student	Requires created and enabled student account (Active Status = True)	Submitting articles	Click "Create" button without Choose "Agree to Terms and Conditions "	Submission file fail	Submission file fail	Pass
6	Student	Requires created and enabled student account (Active Status = True)	Accept the terms and conditions that require approval before submitting articles	Choose "Agree to Terms and Conditions and Click "Create" button	Submission the file successfully	Submission the file successfully	Pass

7	Student	Requires created and enabled student account (Active Status = True)	Back Student Index	Click "Back to index" button	Show view Student Index successfully	Show view Student Index successfully	Pass
8	Student	Requires created and enabled student account (Active Status = True)	Read comments for my own submissions	Display comments of submissions in Student Index page	Comment is displayed successfully	Comment is displayed successfully	Pass
9	Student	Requires created and enabled student account (Active Status = True)	Update my contributions until a final closure date	Articles may be updated before the submission deadline	Update successfully	Update successfully	Pass

Table 7: Test case and Test log for student account

1.2.2. Sprint 1 backlog

Below is a Sprint 1 backlog with user story related to all the functions of the respective roles mentioned in test cases and test logs of Sprint 1.

Product backlog item	Tasks	Volunteer	Status	Original Estimate	Monday (15/2)	Tuesday (16/2)	Wednesday (17/2)	Thursday (18/2)	Friday (19/2)	Monday (22/2)	Tuesday (23/2)	Wednesday (24/2)	Thursday (25/2)	Friday (26/2)	Sprint review
As a student, I want to be able to post one or more photos and articles so that I can contribute to the school magazine the best products	Create database	Vinh	Done	0	0	0	0	0	0	0	0	0	0	0	0
	Code front-end for submit articles and photos	Bao, Vy	Done	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1	0	0	0	0
	Code back-end for submit articles and photos	Thuyen	Done	1.5	1.5	1.5	0	0	0	0	0	0	0	0	0
	Write test case, test log	Nguyen	Done	1	1	1	1	0.5	0.5	0.5	0.5	0.5	0	0	0
As an administrator, I want to be able to create new account and control login permissions so that I can grant permissions to suit the user's role	Create database	Vinh	Done	6	4	3	3	1	1	0	0	0	0	0	0
	Code front-end for create user accounts and authorize them	Bao, Vy	Done	2	2	2	2	2	2	2	1	0	0	0	0
	Code back-end for create user accounts and authorize them	Thuyen	Done	3	3	3	3	2	1	0	0	0	0	0	0
	Write test case, test log	Tuan	Done	4	4	4	4	4	4	4	4	4	2	0	0
Total				19	17	16	14.5	11	10	8	8	6.5	2	0	0

Table 8: Sprint 1 backlog

1.2.3. Sprint 1 burndown chart

The burndown chart below depicts our team's working process during Sprint 1. It is clear that there are significant differences between actual and estimated effort. Most of these differences can be attributed to many subjective factors, such as coding, which takes more time to figure out how to do before actually implementing a function, or the testing process, which also takes a

long time to complete before receiving the final results. Because of this, the project's Sprint 1 implementation did not go as smoothly as planned. The graph clearly shows these things, as can be seen in the early stages, because our team spent a lot of time figuring out how to do the functions, testing incomplete functions, and so on, so there isn't much progress on the workflow in the early stages. However, after gaining more knowledge to perform those tasks in the second half of Sprint1, we were able to pick up the pace and meet the Sprint1 deadline.

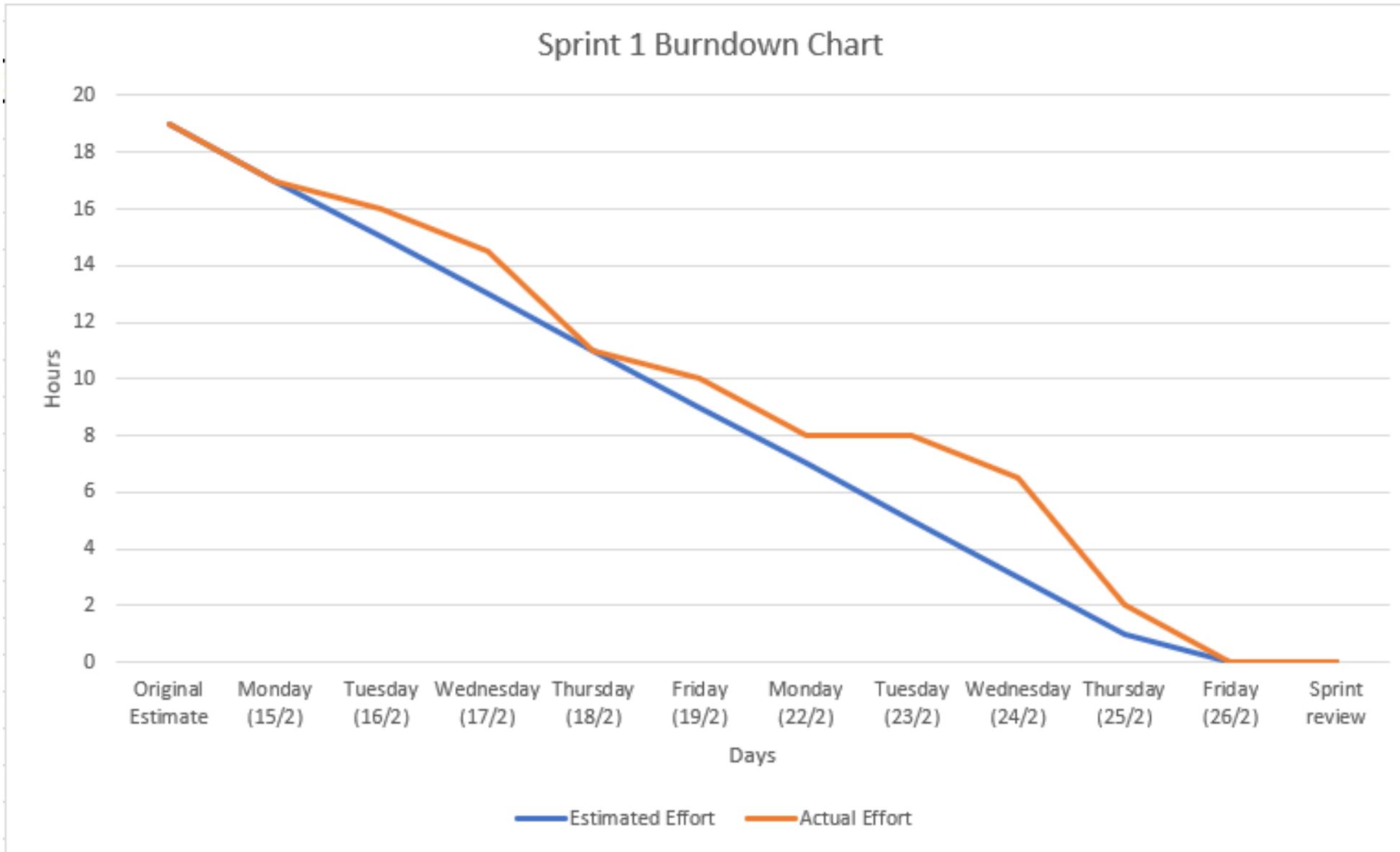


Figure 108: Sprint 1 Burndown Chart

1.3. Sprint 2

1.3.1. Test Cases and Test Logs of Sprint 2

Below is the table showing the test case and test log of Sprint 2 which are made to test some functions specific to the marketing coordinator account and the admin account. More specifically, the functions such as receiving emails, creating comments, scoring, viewing articles of the marketing coordinator account and the function of controlling the deadline of the faculties of the admin account.

Test Case ID	Role	Test Scenario	Test Case	Pre-Conditions	Test Steps	Test Data	Post Conditions	Expected Result	Actual Result	Status
TC_GET_NOTIFICATION_001	Marketing Coordinator	Check the email notification function, exclusively for the marketing coordinator account	Check response when the student accounts submit articles	- Requires email of enabled marketing coordinator account (Active Status = True) - Requires enabled student account (Active Status = True)	1. Select Log in 2. Login with the enabled student account 3. Select Student Index 4. Select Submit 5. Post images and DOC files 6. Click Create button 7. Check the email of the marketing coordinator account in charge of the student just posted	Images and DOC files	- The successfully submitted images or doc files are displayed on the Submit File page of the student account - The notification appears in the email inbox of marketing coordinator account	The notification is successfully sent by the system to the email of the marketing coordinator account	The system sent notifications to the email of the marketing coordinator's account with each student submission	Pass
TC_MAKE_COMMENTS_01	Marketing Coordinator	Check the comments creation function, exclusively for the marketing coordinator account	Check response when using text box to create comments within 14 days after the system stops accepting new articles and during the time before the system stops accepting new articles	- Requires student article - Requires enabled marketing coordinator account (Active Status = True)	1. Select Log in 2. Login with the enabled marketing coordinator account 3. Select Coordinator Index 4. Select Mark (on the Coordinator Index page) 5. Select Mark (on the Mark page) 6. Create comments 7. Click Save button	The lines of words, numbers, and special characters	- The Mark page of the marketing coordinator account appears with new comments corresponding to the student's article - The comments appear in the Comment section of the Student Index page (of student accounts)	The marketing coordinator account can create comments for student articles within 14 days after the system stops accepting new articles and during the time before the system stops accepting new articles	The comments corresponding to the student articles have been successfully created	Pass

TC_MAKE_COMMENTS_02	Marketing Coordinator	Check the comments creation function, exclusively for the marketing coordinator account	Check response when trying to use text box to create comments after 14 days since the system stopped accepting new articles from students	- Requires student article - Requires enabled marketing coordinator account (Active Status = True)	<ol style="list-style-type: none"> 1. Select Log in 2. Login with the enabled marketing coordinator account 3. Select Coordinator Index 4. Select Mark (on the Coordinator Index page) 5. Select Mark (on the Mark page) 	None	The Marking page appears with submitted photos, DOC files and the text box used to create the comments is gone	After 14 days since the system stopped accepting new articles from students, the Marketing Coordinator account in charge of these articles cannot create comments anymore	After 14 days since the system stopped accepting new articles from students, Marketing Coordinator account was unable to create comments	Pass
TC_SELECT_ARTICLES_001	Marketing Coordinator	Check the grading function to choose the student articles that can public, exclusively for the marketing coordinator account	Check response when changing the grade value	- Requires student article - Requires enabled marketing coordinator account (Active Status = True)	<ol style="list-style-type: none"> 1. Select Log in 2. Login with the enabled marketing coordinator account 3. Select Coordinator Index 4. Select Mark (on the Coordinator Index page) 5. Select Mark (on the Mark page) 6. Select the value of Grade 7. Click Save button 	Grade: (selected all values)	<p>- The Mark page of the marketing coordinator account appears with a new grade corresponding to the student's article</p> <p>- The grade appears in the Mark section of the Student Index page (of student accounts)</p>	The marketing coordinator account can choose which student articles will be published, are in need of editing and will not be published	The marketing coordinator account was able to choose which articles were selected for publication through grading	Pass

TC_CONTROL_TIME_001	Admin	Check the function of setting an end date for the new submission, exclusively for the admin account	Check response when using the Edit Date End function in the Edit Course page to change the New date End for the new submission	- Requires enabled admin account (Active Status = True) - Requires created faculty includes a number of student account that is held by one marketing coordinator account	1. Select Log in 2. Login with the enabled admin account 3. Select Course Index 4. Select Edit (on the Index Course page) 5. Adjust the New date End 6. Click Save button	New date End: 12/04/2021 09:26 AM (The data is displayed in order: day/month /year hours:minutes session)	- The Index Course page of the admin account appears with the edited Date end data corresponding to the selected faculty determined by the ID of the marketing coordinator in charge of that faculty - The Date end for the new submission of some student accounts and the Date end for comment creation of the marketing coordinator account related to the newly adjusted Date end will correspondingly change	The Date end for the new submission of faculties and marketing coordinator comment creation can be customized	Successfully adjusted the Date end for the faculties	Pass

TC_CONTROL_TIME_002	Admin	Check the function of setting an end date to stop accepting edited articles, exclusively for the admin account	Check response when using the Edit Date End function in the Edit Course page to change the New date Final for the edited submission	- Requires enabled admin account (Active Status = True) - Requires created faculty includes a number of student account that is held by one marketing coordinator account	1. Select Log in 2. Login with the enabled admin account 3. Select Course Index 4. Select Edit (on the Index Course page) 5. Adjust the New date Final 6. Click Save button	New date Final: 1/05/2021 11:29 AM (The data is displayed in order: day/month /year hours:minutes session)	- The Index Course page of the admin account appears with the edited Date Final data corresponding to the selected faculty determined by the ID of the marketing coordinator in charge of that faculty - The Date Final for the edited submission of some student accounts related to the newly adjusted Date Final will correspondingly change	The Date Final for the edited submission of faculties can be customized	Successfully adjusted the Date Final for the faculties	Pass
TC_VIEW_ARTICLES_001	Marketing Coordinator	Check the ability to view student articles online of marketing coordinator account	Check response when the marketing coordinator account views the student's submitted articles online	- Requires student article - Requires enabled marketing coordinator account (Active Status = True)	1. Select Log in 2. Login with the enabled marketing coordinator account 3. Select Coordinator Index 4. Select Mark (on the Coordinator Index page) 5. Select Mark (on the Mark page)	Images and DOC files submitted	Images and DOC files corresponding to the student's account are displayed online (if they exist)	The submitted DOC files and images can be viewed online	Images and DOC files displayed online	Pass

Table 9: Test Cases and Test Logs of Sprint 2 for admin account and marketing coordinator account

1.3.2. Sprint 2 backlog

Below is a Sprint 2 backlog with user story related to all the functions of the respective roles mentioned in test cases and test logs of Sprint 2.

Product backlog item	Tasks	Volunteer	Status	Original Estimate	Monday (1/3)	Tuesday (2/3)	Wednesday (3/3)	Thursday (4/3)	Friday (5/3)	Monday (8/3)	Tuesday (9/3)	Wednesday (10/3)	Thursday (11/3)	Friday (12/3)	Sprint review
As a marketing coordinator, I want to be able to receive email notifications from the system whenever the students in my faculty publish their articles so that I can work with those contributions more conveniently	Edit database and code front-end	Vinh	Done	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0	0
	Code front-end for display all submitted articles	Bao, Vy	Done	0	0	0	0	0	0	0	0	0	0	0	0
	Code back-end for display all submitted articles	Thuyen	Done	1	1	1	1	0	0	0	0	0	0	0	0
	Write test case, test log	Tuan	Done	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0	0
As an administrator, I want to be able to control the update time, the closing time of some data in the system such as how long new articles are allowed to be posted, the	Edit database and code front-end	Vinh	Done	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0	0	0
	Code front-end for control the update time, the closing time of some data in the system	Bao, Vy	Done	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	0.5	0	0	0

posted articles are allowed to be edited and the end time of the editing of posted articles so that the system works properly as required	Code back-end for control the update time, the closing time of some data in the system	Thuyen	Done	3	2	1	0	0	0	0	0	0	0	0	0
	Write test case, test log	Tuan	Done	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	0.5	0	0
As a student, I want to be able to read comments for my own submissions so that I can easily edit the articles accordingly	Edit database and code front-end	Vinh	Done	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0	0	0
	Code front-end for reading comments for own submissions	Bao, Vy	Done	1	1	1	1	1	1	1	0	0	0	0	0
	Code back-end for reading comments for own submissions	Thuyen	Done	1.5	1.5	1.5	1.5	0	0	0	0	0	0	0	0
	Write test case, test log	Nguyen	Done	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0
As a marketing coordinator, I want to be able to select contributions to publish so that my	Edit database and code front-end	Vinh	Done	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0	0	0
	Code front-end for accept or	Bao, Vy	Done	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0	0	0

faculty can contribute the best quality articles to the magazine	reject student articles														
	Code back-end for accept or reject student articles	Thuyen	Done	1	1	1	1	1	0	0	0	0	0	0	0
	Write test case, test log	Tuan	Done	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0	0
As a student, I want to be able to update my contributions until a final closure date so that I can directly make the necessary changes	Edit database and code front-end	Vinh	Done	0.5	0.5	0.5	0.5	0.5	0	0	0	0	0	0	0
	Code front-end for update contributions until the final closure date	Bao, Vy	Done	0	0	0	0	0	0	0	0	0	0	0	0
	Code back-end for update contributions until the final closure date	Thuyen	Done	1	1	1	1	1	0	0	0	0	0	0	0
	Write test case, test log	Nguyen	Done	1	1	1	1	1	1	1	0	0	0	0	0
As a marketing manager, I want to be able to view all the selected contributions so	Edit database and code front-end	Vinh	Done	1	1	1	1	1	1	1	1	0	0	0	0
	Code front-end for view all	Bao, Vy	Done	1.5	1.5	1.5	1.5	1.5	1.5	1.5	0	0	0	0	0

that I can give the necessary feedback for contributions	selected contributions														
	Code back-end for view all selected contributions	Thuyen	Done	1.5	1.5	1.5	1.5	0.5	0.5	0	0	0	0	0	0
	Write test case, test log	Nguyen	Done	1	1	1	1	1	1	1	1	1	0.5	0	0
As a marketing coordinator, I want to be able to view contributions from students in my faculty so that I can review and evaluate their contributions	Edit database and code front-end	Vinh	Done	1	1	1	1	1	1	1	0	0	0	0	0
	Code front-end for view contributions from students	Bao, Vy	Done	1	1	1	1	1	1	1	1	0	0	0	0
	Code back-end for view contributions from students	Thuyen	Done	1.5	1.5	1.5	1.5	1.5	1.5	0	0	0	0	0	0
	Write test case, test log	Tuan	Done	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0
As a marketing coordinator, I want to be able to make comments for my student's articles so that I can help my students make the necessary updates	Edit database and code front-end	Vinh	Done	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0	0	0	0
	Code front-end for making comments on student articles	Bao, Vy	Done	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0	0	0	0

	Code back-end for making comments on student articles	Thuyen	Done	1	1	1	0	0	0	0	0	0	0	0	0	0
	Write test case, test log	Tuan	Done	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0	0
Total				28.5	27.5	26.5	24.5	21	18.5	14.5	9	4.5	1	0	0	0

Table 10: Sprint backlog 2

1.3.3. Sprint 2 burndown chart

Below is a burndown chart showing the working process of our team during Sprint 2. In Sprint 2, because the workload has increased significantly, so in order to meet the project deadline, the working time of the team members also increases. Because the way our team working has not changed too much since Sprint 1, so in the early stages of Sprint 2, we have been progressing quite slowly compared to our initial estimate. However, when the same problems and difficulties as mentioned for Sprint 1 have been resolved, we have accelerated the progress and caught up with the Sprint 2 deadline.

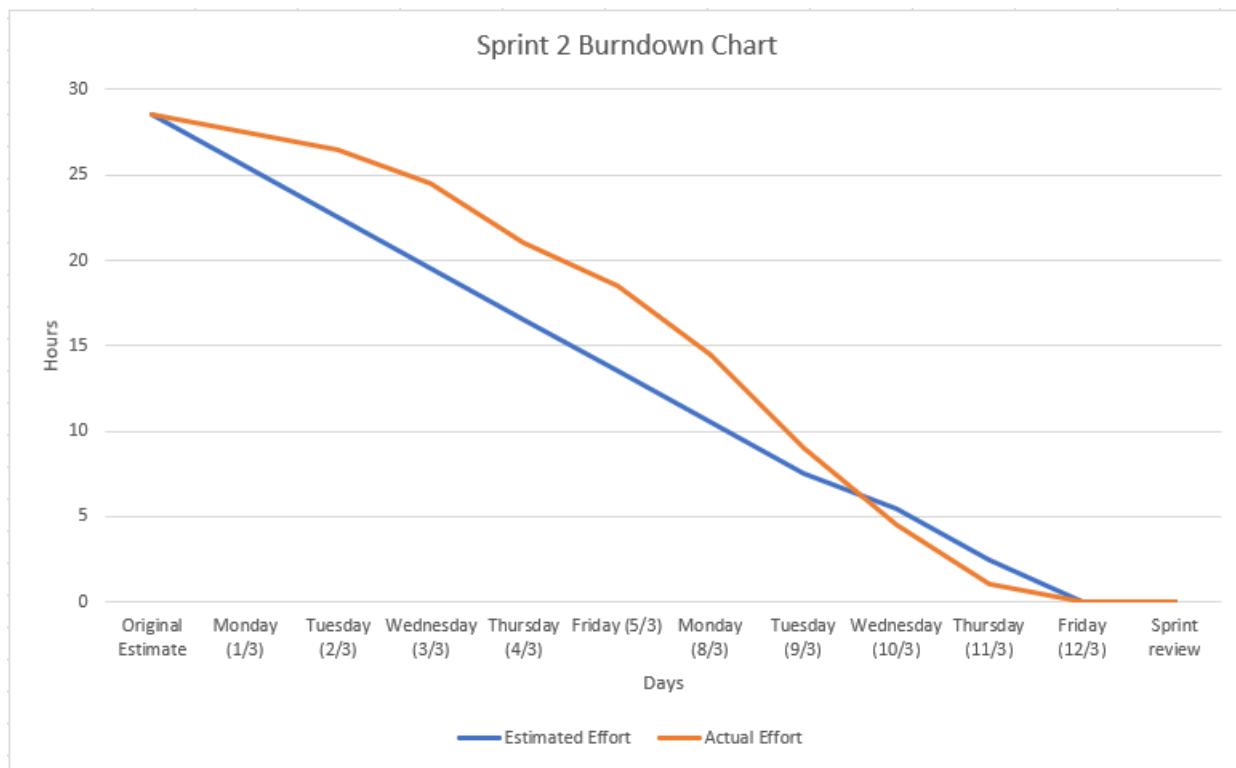


Figure 109: Sprint 2 Burndown Chart

1.4. Sprint 3

1.4.1. Test Cases and Test Logs of Sprint 3

Test Case ID	Role	Test Scenario	Test Case	Pre-Conditions	Test Steps	Test Data	Post Conditions	Expected Result	Actual Result	Status
TC_SEND_MESSAGE_001	Marketing Coordinator	Check messaging functionality, exclusively for student accounts and marketing coordinator accounts	Check response when the marketing coordinator account sends a message to the student account	Requires enabled marketing coordinator account (Active Status = True)	1. Select Log in 2. Login with the enabled marketing coordinator account 3. Select Chat 4. Click Chat corresponding to the email of student account needs to message 5. Enter messages 6. Click Send button	The lines of words, numbers, and special characters	Recently sent messages appear on the chat frame for 2 accounts	The marketing coordinator account can interact with student accounts belonging to the faculty that account is in charge	The messages have been successfully transmitted to the student account	Pass
TC_MAIN_TAIN_DATA_001	Admin	Check the ability to maintain data in the system through the View function to view information, the status of student accounts and related articles of all faculties, exclusively for admin account	Check the response when clicking the View button in the Index Course page for the faculties in the system and when clicking the View button to see the contributions of student accounts	Requires enabled admin account (Active Status = True)	1. Select Log in 2. Login with the enabled admin account 3. Select Course Index 4. Select View (on the Index Course page) 5. Select View (on the View Mark page)	None	- The View Mark page of the admin account appears with information related to the student accounts of the selected faculty - The View Submissions page	Admin has the ability to view and preserve all articles submitted by the student accounts in the system	Student's submitted articles can be viewed and preserved through the Admin account	Pass

		belonging to those faculties			of the admin account appears with the student's submitted articles (if they exist)		
--	--	------------------------------	--	--	--	--	--

Below is the table showing the test case and test log of Sprint 3 which are made to test the messaging functionality of the marketing coordinator account and the data maintenance function of the admin account.

Table 11: Test Cases and Test Logs of Sprint 3 for admin account and marketing coordinator account

Test	Role	Pre-Conditions	What is being test	Test data used	Expected Results	Actual result	Status
1	Marketing Manager	Requires created and enabled marketing manager account (Active Status = True)	Manager Index	Click "Manager Index" button to view Marketing Manager Index	Show View Marketing Manager Index successfully	Show View Marketing Manager Index successfully	Pass
2	Marketing Manager	Requires created and enabled marketing manager account (Active Status = True)	View Mark page	Click" View" button in View Mark column	Show View Mark successfully	Show View Mark successfully	Pass
3	Marketing Manager	Requires created and enabled marketing manager account (Active Status = True)	View Submissions	Click " View Submissions" button to View Submissions page	Show View Submissions page successfully	Show View Submissions page successfully	Pass
4	Marketing Manager	Requires created and enabled marketing manager account (Active Status = True)	Download zip	Click “Download zip” button in View Submissions page	Download file zip successfully	Download file zip successfully	Pass
5	Marketing Manager	Requires created and enabled marketing manager account (Active Status = True)	View the exceptional reports	Click " View" in Exceptional column	Show view Exceptional page successfully	Show view Exceptional page successfully	Pass
6	Marketing Manager	Requires created and enabled marketing manager account (Active Status = True)	Keep track of the number of contributions, contributors and the percentage of contributions by multiple faculties	Click "View" in Contributions column	Show view Contribution page successfully	Show view Contribution page successfully	Pass
7	Marketing Manager	Requires created and enabled marketing manager account (Active Status = True)	Back to index	Click" Back to index" button to back Marketing Manager Index	Show View Marketing Manager Index successfully	Show View Marketing Manager Index successfully	Pass
8	Guest	Requires created and enabled guest account (Active Status = True)	View Mark page	Click “Guest Index” button	Show View Mark page successfully	Show View Mark page successfully	Pass

9	Guest	Requires created and enabled guest account (Active Status = True)	View Submissions page	Click "View" button	Show View Submissions page successfully	Show View Submissions page successfully	Pass
---	-------	---	-----------------------	---------------------	---	---	------

Below is a table showing the test case and test log created to test the marketing manager account functions such as viewing all selected articles, viewing exception articles, downloading articles as zip file, monitoring data of all faculties and the guest account function is viewing all the student articles in the faculty.

Table 12: Test Cases and Test Logs for marketing manager account and guest account

1.4.2. Sprint 3 backlog

Below is a Sprint 3 backlog with user story related to all the functions of the respective roles mentioned in test cases and test logs of Sprint 3.

Product backlog item	Tasks	Volunteer	Status	Original Estimate	Monday (15/3)	Tuesday (16/3)	Wednesday (17/3)	Thursday (18/3)	Friday (19/3)	Monday (22/3)	Tuesday (23/3)	Wednesday (24/3)	Thursday (25/3)	Friday (26/3)	Sprint review
As a marketing coordinator, I want to be able to interact with the students in my faculty to easily give suggestions, discuss necessary adjustments with my students	Edit database and code front-end	Vinh	Done	1	1	1	1	1	1	1	1	0	0	0	0
	Code front-end for interact with the students	Bao, Vy	Done	1.5	0	0	0	0	0	0	0	1	0.5	0	0
	Code back-end for interact with the students	Thuyen	Done	3	3	3	3	3	3	3	3	2	1	0	0
	Write test case, test log	Tuan	Done	1	1	1	1	1	1	1	1	1	1	0	0
As a marketing manager, I want to be able to view the exceptional reports so that I can choose the best articles	Edit database and code front-end	Vinh	Done	1	1	1	1	1	1	1	0	0	0	0	0
	Code front-end for view the exceptional reports	Bao, Vy	Done	1	1	1	1	1	1	0.5	0	0	0	0	0
	Code back-end for view the exceptional reports	Thuyen	Done	2	2	2	2	1	0	0	0	0	0	0	0

	Write test case, test log	Nguyen	Done	1	1	1	1	1	1	1	1	0	0	0	0
As a marketing manager, I want to be able to download all selected contributions as zip file so that I can transfer them out of the system	Edit database and code front-end	Vinh	Done	0	0	0	0	0	0	0	0	0	0	0	0
	Code front-end for download all selected contributions as zip file	Bao, Vy	Done	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0	0	0	0
	Code back-end for download all selected contributions as zip file	Thuyen	Done	2.5	2.5	1	0	0	0	0	0	0	0	0	0
	Write test case, test log	Nguyen	Done	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0	0	0	0	0
	Edit database and code front-end	Vinh	Done	1	1	1	1	1	1	1	0	0	0	0	0
As a marketing manager, I want to be able to keep track of the number of contributions, contributors and the percentage of contributions by multiple faculties so that I can ensure the	Code front-end for keep track of the number of contributions, contributors and the percentage of contributions	Bao, Vy	Done	1	1	1	1	1	1	1	1	0	0	0	0

	Write test case, test log	Tuan	Done	1	1	1	1	1	1	0.5	0	0	0	0	0
As a student, I want to be able to accept the terms and conditions that require approval before submitting articles so that I can avoid the issues related to them when submitting my submission	Edit database and code front-end	Vinh	Done	0	0	0	0	0	0	0	0	0	0	0	0
	Code front-end for accept the terms and conditions	Bao, Vy	Done	0	0	0	0	0	0	0	0	0	0	0	0
	Code back-end for accept the terms and conditions	Thuyen	Done	0.5	0	0	0	0	0	0	0	0	0	0	0
	Write test case, test log	Nguyen	Done	0.5	0	0	0	0	0	0	0	0	0	0	0
As a guest, I want to be able to view faculty selected reports so that I can keep track of magazine contents	Edit database and code front-end	Vinh	Done	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1	0	0	0
	Code front-end for view selected reports for each faculty	Bao, Vy	Done	1	1	1	1	1	1	1	0	0	0	0	0
	Code back-end for view selected reports for each faculty	Thuyen	Done	2	2	2	2	2	2	1	0	0	0	0	0

	Write test case, test log	Nguyen	Done	1	1	1	1	1	1	1	1	1	1	0	0
Total				30	26	24.5	22.5	21	19	16.5	10	6.5	3.5	0	0

Table 13: Sprint 3 backlog

1.4.3. Sprint 3 burndown chart

Below is a burndown chart showing the working process of our team during Sprint 3. In Sprint 3, because we gained more experience and knowledge through the process of researching and learning from previous Sprints, so our team went through the work more quickly, sticking to the original estimate. As a result, the first stage of Sprint 3 has progressed faster than the early stages of the previous Sprints. However, there are still certain difficulties when performing some functions such as interaction between marketing coordinator account and student account through messages, monitoring data of faculties, etc. So the mid-stage implementation of Sprint 3 has become slower than originally estimated due to the fact that coding and testing for some functions have to take place many times. However, our team has found a way to improve and overcome the limitations and bugs of the system as much as possible and has caught up with the Sprint 3 deadline.

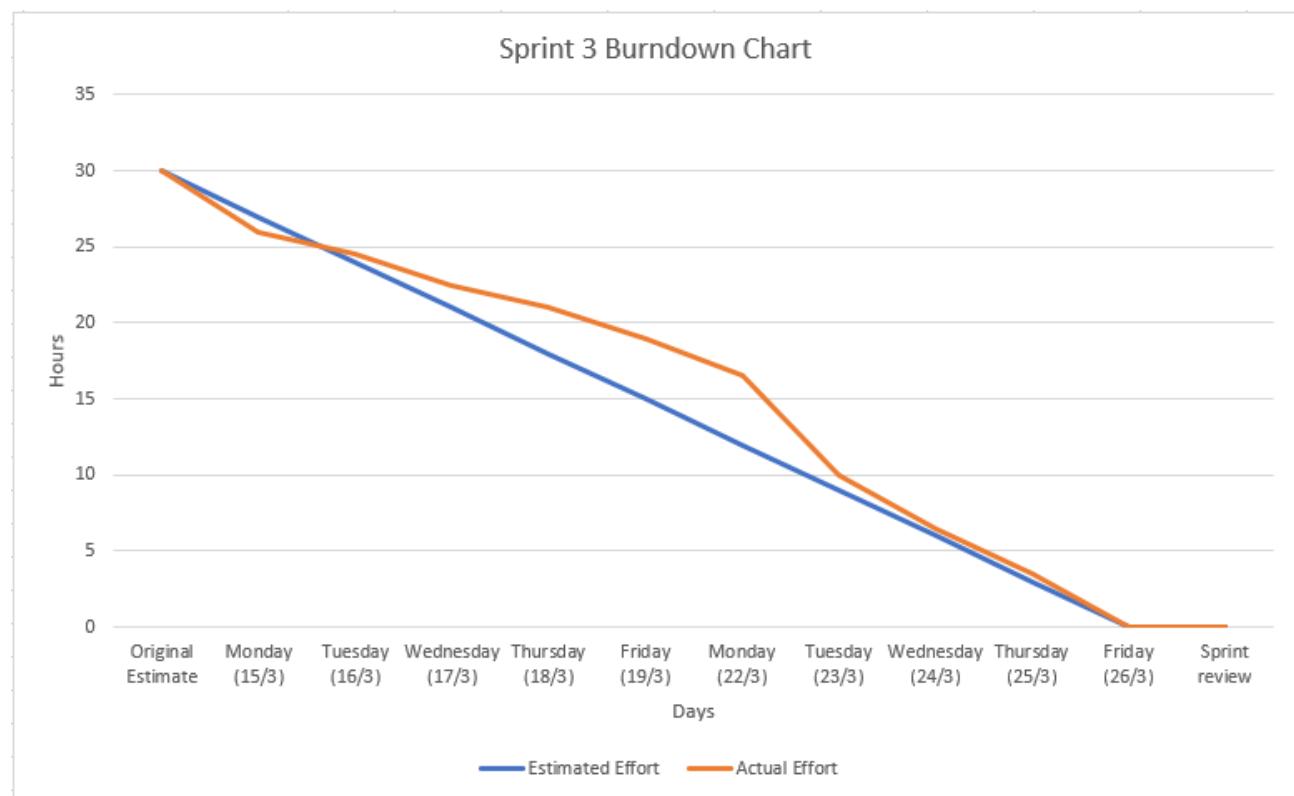


Figure 110: Sprint 3 Burndown Chart

1.5. Test evidence

Below is the test evidence used for the functionality contained in our system.

Test evidence for general functions:

- **Log in**

Below is an image of the page containing the login function, which can be used by entering the email, password of the user account and clicking the Sign In button.

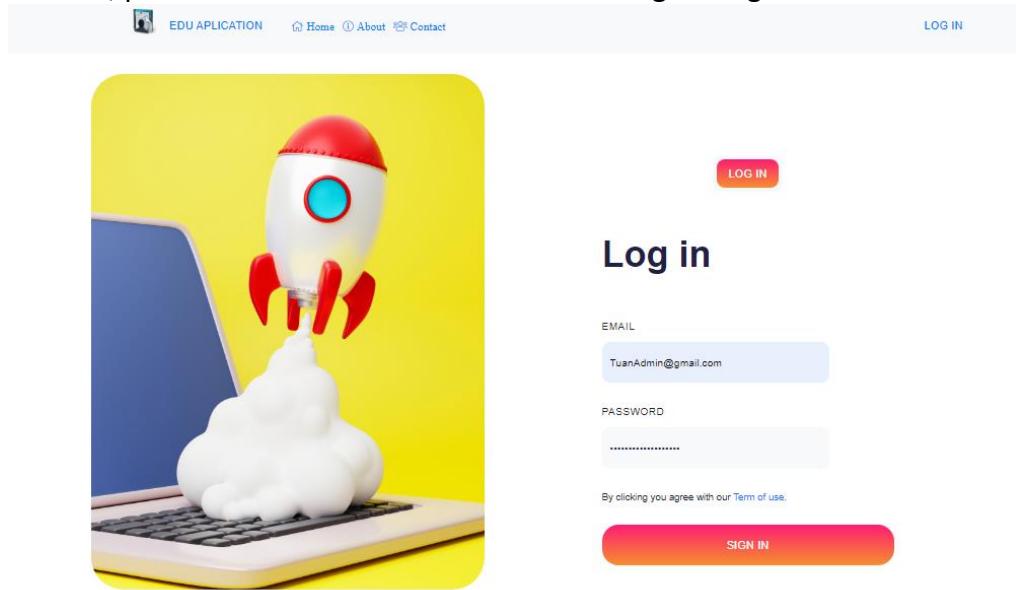


Figure 111: Test evidence of the login function

- **Log out**

In the upper right corner of the image below is the Log out button that is used to log out of the user account when clicking on it.



Figure 112: Test evidence of the logout function

Test evidence for specific functions:

➤ For Admin account:

- **Create and edit update time, closing time**

Below is an image of the Edit Course page with 2 adjustment bars New date End and New date Final. More specifically, the New date End is used to adjust the deadline for submitting new articles of the student account and the comment creation deadline of the marketing coordinator account in a faculty. The New date Final is used to adjust the deadline for submitting edited articles of the student account. After adjusting these bars to the necessary deadlines, the administrator clicks the Save button to be able to successfully create and edit them.

Coordinator	Old date end	Old date Final	nameCourse	New date End	New date Final
ouDVOHSEvVfzJlYiyBXairfc2	15/04/2021 04:09:00 PM	22/04/2021 04:09:00 PM	QT	mm/dd/yyyy ...::: ..	mm/dd/yyyy ...::: ..

Figure 113: Test evidence of the function to create and edit update time, closing time

- **Create account**

Below is an image of a page used to create user accounts. To successfully perform the function of creating user accounts, the administrator needs to fill in the Full Name, Email and Password for these accounts. In addition, for the student account and guest account, the administrator needs to choose a marketing coordinator account for them to be in charge of their faculty. For marketing coordinator account, when creating it, the administrator needs to fill in the Course name to specify the name of the faculty that the marketing coordinator is in charge of. After filling in the necessary information to create a user account, the administrator clicks the Create button to complete the account creation.

The screenshot shows a web application interface for creating a student account. At the top, there is a navigation bar with links for Home, About, Contact, Admin Index (selected), Course Index, and Log out. The main title is 'Create Student Account'. A message below the title says 'Please fill in this form to create an account.' There are four input fields: 'Full name' (Ex: John Doe), 'Email' (Ex: john doe@email.com), 'Password' (*****), and 'Marketing Coordinator' (Pham Ngoc Bao (P2qkxRm62bYO3fh)). Below the form, a note states 'By creating an account student need agree to Terms & Privacy.' with a 'Create' button. At the bottom left is a link '< Back to List'.

Figure 114: Test evidence of the function to create account

- **Storing and preserving all system data**

Below is an image of the View Mark page, used by the admin to keep track of student articles from all faculties. In addition, the administrator can view the student's articles by clicking the View button and from there, the administrator can also download the articles for storing.

The screenshot shows the Admin Index page of the Edu Application. At the top, there is a navigation bar with links for Home, About, Contact, Admin Index (which is highlighted in blue), Course Index, and Log out. The main content area is titled "View Mark" and contains a table with three rows of student information. The columns are Student, Mail, Mark, Comment, and View Submissions. Each row has a "View" button in the last column.

Student	Mail	Mark	Comment	View Submissions
2y47LajwVVhKDQfH5czQG1OyEBh2	studentbao@gmail.com	Reject		View
sHZ1uXsxrZe3gKaqf8czjLOvs13	pnb@gmail.com	Not Grade		View
uyKjKQWaYgjs06lzMU99fTm9HB2	TuanSTUDENT1@gmail.com	Accept		View

© 2021 - Edu Application by ASP.NET

[Web hosting by Somee.com](#)

Figure 115: Test evidence of the function to store and preserve all system data

➤ For Student account:

- Submit articles as word documents and high-quality images

Below is an image of the Submit File page, used by students to submit their articles as doc files and high-quality images. To submit their articles, students click on Choose File, select the doc file to send, choose to Agree to Terms and Conditions and click on the Create button.

The screenshot shows the Student Index page of the Edu Application. At the top, there is a navigation bar with links for Home, About, Contact, Student Index (which is highlighted in blue), Chat, and Log out. The main content area is titled "Submit File" and contains a form for uploading a file. It includes fields for Coordinator, Mail, End date, a file input field labeled "Choose File" with "Test.docx" selected, a checkbox for "Agree to Terms and Conditions", and a "Create" button.

Submit File

Coordinator: ouDVOH5EVsVfzJIYiyIBXairlfC2

Mail: cje.madlife2000@gmail.com

End: 4/15/2021 4:09:00 PM

 Test.docx

Agree to Terms and Conditions

[Create](#)

[< Back to index](#)

© 2021 - Edu Application by ASP.NET

[Web hosting by Somee.com](#)

Figure 116: Test evidence of the function to submit articles

- **Read comments for own submissions**

Below is an image of the Student Index page, used by students to read the comments that the marketing coordinator writes for their articles. These comments can be seen immediately in the Comment section.

The screenshot shows a web application interface titled "EDU APPLICATION". At the top, there is a navigation bar with links for "Home", "About", "Contact", "Hello Tuan", "Student Index" (which is highlighted in blue), "Chat", and "Log out". Below the navigation bar, the title "Student Index" is displayed. A table lists course submissions with columns for Course Name, Coordinator, Mail, Date End, Date Final, Mark, Comment, and Action. One row in the table is shown, with a "Comment" field containing "GOOD JOB" and a "Submit" button. At the bottom of the page, there is a copyright notice "© 2021 - Edu Application by ASP.NET" and a link "Web hosting by Somee.com".

Course Name	Coordinator	Mail	Date End	Date Final	Mark	Comment	Action
QTKD	ouDVOH5EVsVfzJlYiyIBXairfc2	cje.madlife2000@gmail.com	15/04/2021 04:09:00 PM	22/04/2021 04:09:00 PM	Accept	GOOD JOB	Submit

Figure 117: Test evidence of the function to read comments for own submissions

- **Update contributions until a final closure date**

Below is an image of the Submit File page with the same functions as mentioned above. Therefore, it can be seen that this is also the page used by students to update their previously submitted articles. As can be seen in the photo below, if the deadline for submitting edited articles remains, students can still delete and submit new articles by clicking Delete and Choose File.

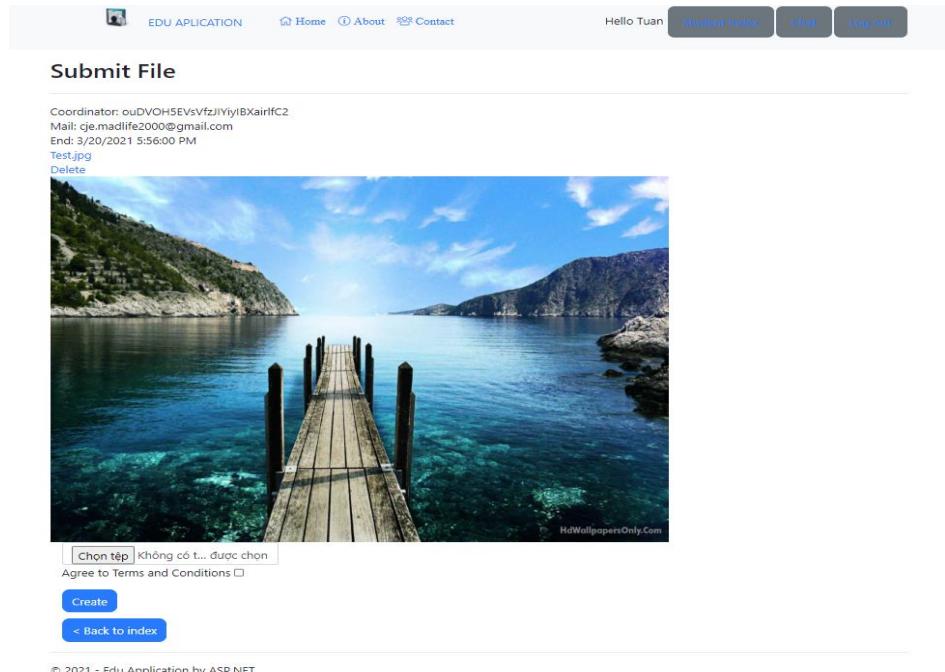


Figure 118: Test evidence of the function to update contributions until a final closure date

- **Accept the terms and conditions that require approval before submitting articles**

Similar to the Submit File pages mentioned above, however in order for students to actually submit their articles, they need to select Agree to Terms and Conditions before clicking the Click button.

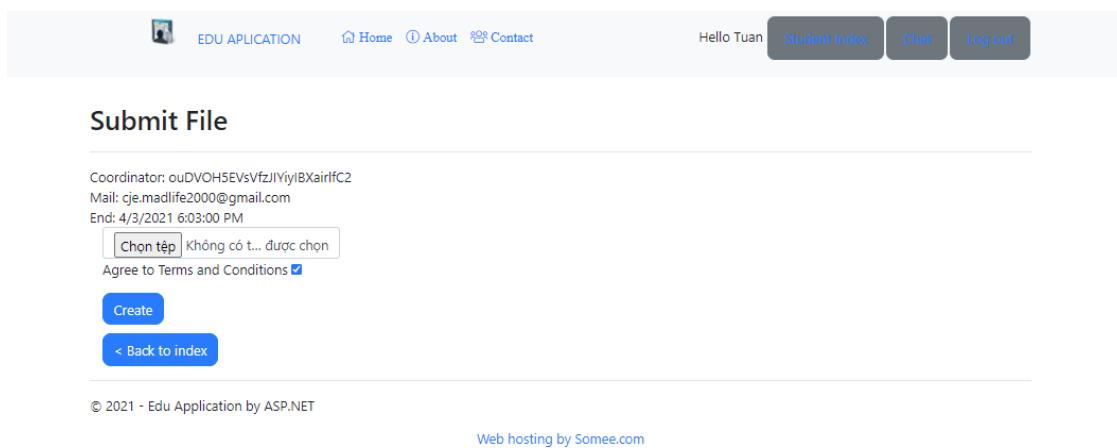


Figure 119: Test evidence of the function to accept the terms and conditions

➤ **For Marketing Coordinator account:**

- **Receive email notifications from the system whenever the students in faculty publish their articles**

Below is an image showing the notifications that the system will send to the email of the marketing coordinator account in charge of the students who have just submitted their articles.

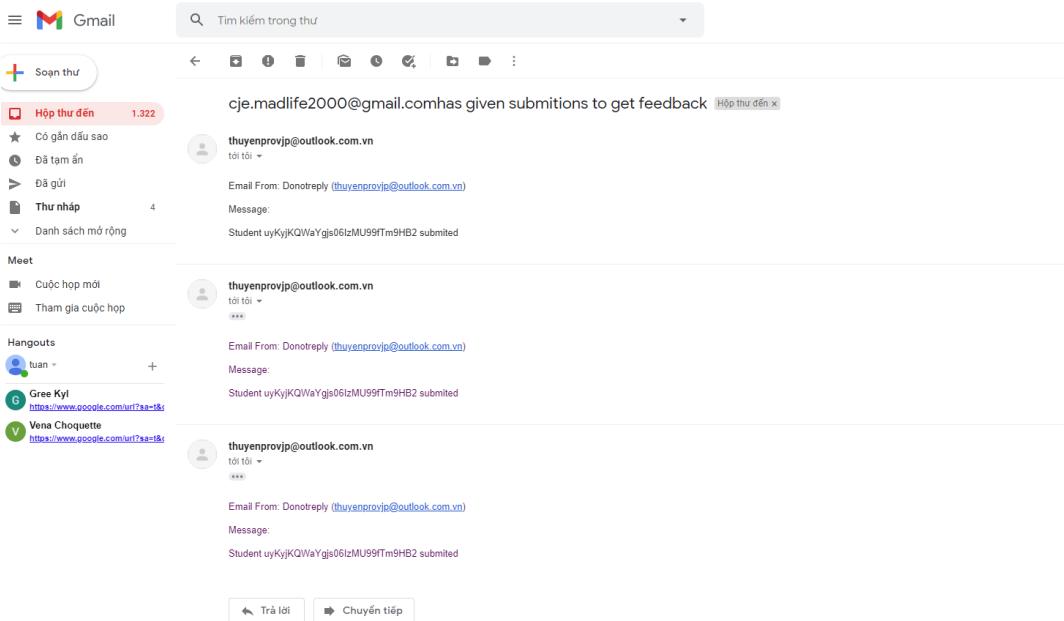


Figure 120: Test evidence of the function to receive email notifications

- **Make comments for student's articles within 14 days after the system stops accepting new articles**

Below is an image of the Marking page, used by the marketing coordinator to create comments for student articles. To send these comments to students, the marketing coordinator needs to create a comment in the Comment section and click on the Save button.

Comment:
You need to edit part 1 of the article

Grade:
Accept (Now)

Save

< Back to Mark

© 2021 - Edu Application by ASP.NET
Web hosting by Somee.com

Figure 121: Test evidence of the function to make comments for student's articles

- **View contributions from students in faculty**

As can be seen in the image below, on the Marking page, the marketing coordinator can also view the articles submitted by students in their faculty.

Discuss.docx

Test/ing

1 / 1

In my opinion, the agile model is more suitable for the company Lizzo's Leather Belts because the progress of this project is very urgent. The company wanted to have a website to distribute its products to consumers within 3 months, but developing a product in a short time was not the strength of the waterfall model. In contrast, the waterfall model requires a detailed plan, which it does not require a detailed, meticulous plan to implement the product. With the agile model, programmers can immediately implement the most necessary functions and interfaces for the website. This will help the company to meet the deadline. However, the waterfall model has many steps and needs to be planned in detail and meticulous in advance (Aron, 2013).

If using waterfall model Lizzo's Leather Belts Company will be very difficult to make the necessary changes and upgrades to meet the actual needs of customers or to adjust the plan to suit new requirements. The waterfall model is not flexible enough to handle changes in requirements, while it does not require a detailed, meticulous plan to implement the product. With the agile model, programmers can immediately implement the most necessary functions and interfaces for the website. This will help the company to meet the deadline. However, the waterfall model has many steps and needs to be planned in detail and meticulous in advance (Aron, 2013).

The most complicated details, within 3 months, is very difficult to design a complete plan of the most complicated details and so when using the waterfall model, it is very likely that stages will be delayed longer than expected, causing the project to be delayed due to the lack of flexibility in the waterfall model (Aron, 2013).

However, with the agile model, being able to perform multiple stages at the same time will help minimize the waste of resources such as human, time, money to implement the website and that is also helps to meet the deadline (Aron, 2013).

Figure 122: Test evidence of the function to view contributions from students in faculty

- **Interact with the students in faculty**

Below is an image showing the chat box used by the marketing coordinator to interact with students in the faculty. To send a message to the student, the marketing coordinator needs to write the message in the text box and click the Send button.

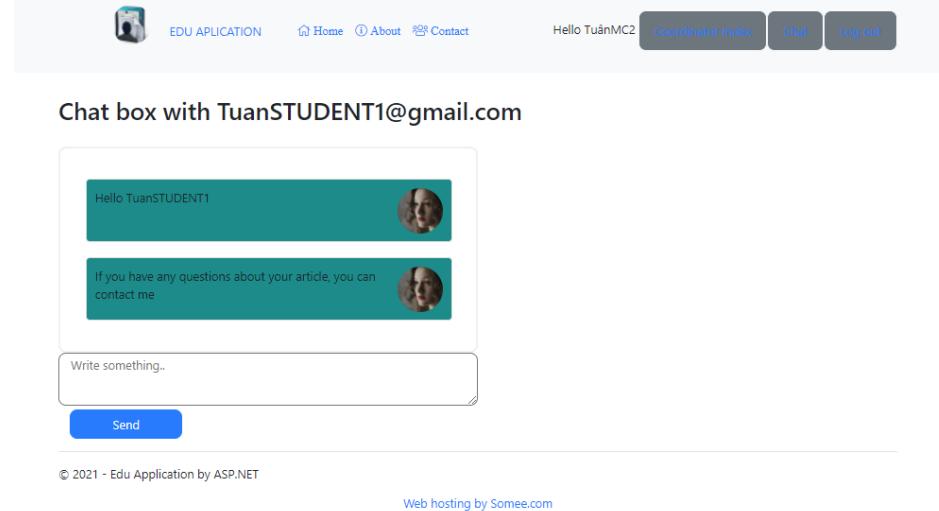


Figure 123: Test evidence of the function to interact with the students in faculty

- **Select contributions to publish**

Similar to the Marking page mentioned above, the marketing coordinator can also select articles that can be published through grading. More specifically, the Not Grade is used for articles that can be published after being edited, the Accept is used for articles that are published immediately after submission without editing and Reject is for articles that are rejected as soon as they are submitted. So, to select articles that can be published, the marketing coordinator needs to grade those articles Accept and click the Save button.

The screenshot shows a 'Marking' page. At the top, there is a header bar with the logo 'EDU APPLICATION' and navigation links for 'Home', 'About', and 'Contact'. On the right side of the header, it says 'Hello TuanMC2' and has buttons for 'Coordinator Index', 'Chat', and 'Log out'. Below the header is the page title 'Marking'. The main form has two sections: 'Comment:' with a text input field containing 'Write something..' and 'Grade:' with a dropdown menu set to 'Reject'. At the bottom of the form are two buttons: a blue 'Save' button and a link '[< Back to Mark](#)'. Copyright information at the very bottom: '© 2021 - Edu Application by ASP.NET' and 'Web hosting by Somee.com'.

Figure 124: Test evidence of the function to select contributions to publish

➤ **For Marketing Manager account:**

○ **View all the selected contributions**

Below is an image of the View Mark page, used by the marketing manager to keep track of all selected articles. To view the selected articles of a particular student, the marketing manager needs to click on the View submissions button.

The screenshot shows a web application interface for a marketing manager. At the top, there is a navigation bar with links for 'EDU APPLICATION', 'Home', 'About', 'Contact', 'Hello TuanMM@gmail.com', 'Manager Index', and 'Log out'. Below the navigation bar, the title 'View Mark' is displayed. A blue button labeled '< Back to index' is visible. The main content area is a table titled 'View Mark' with the following columns: Student, Email, Mark, Comment, and View Submissions. The table contains three rows of data:

Student	Email	Mark	Comment	View Submissions
2y47LajwVVhKDQfH5czQG1OyEBh2	studentbao@gmail.com	Accept		View submissions
sHZ1uXsrxZe3gKaqf8czjLOvs13	pnb@gmail.com	Accept		View submissions
uyKykQWaYgjs06lzMU99fTm9HB2	TuanSTUDENT1@gmail.com	Accept	GOOD JOB	View submissions

© 2021 - Edu Application by ASP.NET

Web hosting by Somee.com

Figure 125: Test evidence of the function to view all the selected contributions

○ **View the exceptional reports**

Below is an image of the View Exceptional page, which is used by the marketing manager to keep track of all articles that are better than other selected articles. These exceptional articles are selected by grading the Accept score right on the first submission without any comment. To view these exceptional articles, the marketing coordinator needs to click the View button.

View Exceptional

Student	Mail	Submissions
g0JGKjLmoDS21Bb8djO3ndSbfYp2	TuanSTUDENT2@gmail.com	View

© 2021 - Edu Application by ASP.NET

[Web hosting by Somee.com](#)

Figure 126: Test evidence of the function to view the exceptional reports

- **Download all selected contributions as zip file**

Below is an image of the View Submissions page, used by the marketing manager to view selected articles of students from all faculties. In addition, the marketing manager can also download these articles as zip files by clicking Download Zip located below the articles.

Coordinator: ouVOH5EVsVfzJIYiyIBairfc2
Student Id: uyKjyKQWaYgjs06izMU99ffm9HB2
Test.jpg

[Download Zip](#)

© 2021 - Edu Application by ASP.NET

[Web hosting by Somee.com](#)

Figure 127: Test evidence of the function to download all selected contributions as zip file

- **Keep track of the number of contributions, contributors and the percentage of contributions by multiple faculties**

Below is an image of the Contributions page, used by the marketing manager to monitor the data of a particular faculty. The page displays data such as the total number of articles submitted by the faculty, the total number of students participating in the faculty submission and the percentage of articles submitted between that faculty and the other faculties.



Contributions

[Back to index](#)

Number of contributions within each Faculty for each academic year: 6
 Percentage of contributions by each Faculty for any academic year: 100.00%
 Number of contributors within each Faculty for each academic year: 3
 The percentage of contributions vs other contributions
 ouDVOH5EVsVfzJlYiyIBXairfcC2 ▾

© 2021 - Edu Application by ASP.NET

[Web hosting by Somee.com](#)

Figure 128: Test evidence of the function to keep track of faculty data

➤ For Guest account:

- **View faculty selected reports**

Below is the View Mark page, which is used by guest to keep track of all the selected articles of students in the faculty. To view these articles, guest need to click on the View button.



View Mark

Student	Email	Mark	Comment	View Submissions
2y47LajwVVhKDQfH5czQG1OyEBh2	studentbao@gmail.com	Accept		View
g0JGKjLmoDS21Bb8djO3ndSbfYp2	TuanSTUDENT2@gmail.com	Accept		View
sHZ1uXsrxZe3gKaqf8czjlOVs13	pnb@gmail.com	Accept		View
uyKyjKQWaYgjs06izMU99fTm9HB2	TuanSTUDENT1@gmail.com	Accept	GOOD JOB	View

© 2021 - Edu Application by ASP.NET

[Web hosting by Somee.com](#)

Figure 129: Test evidence of the function to view faculty selected reports

VII. Agile

2.1. Product backlog

As a/an	I want to be able to...	So that...	Priority level	Added in sprint	Story points	Estimate (time)
Student	Send articles as word documents	I can contribute to the school magazine	Must have	1	50	2
	Upload high-quality images	I can increase the persuasion, interactivity and aesthetics of the article	Must have	1	50	2
	Read comments for own submissions	I can easily edit the articles accordingly	Must have	2	50	3.5
	Update contributions until a final closure date	I can directly make the necessary changes	Must have	2	50	2.5
	Accept the terms and conditions that require approval before submitting articles	I can avoid the issues related to the terms and conditions when submitting my submission	Should have	3	30	1.5
Marketing manager	View all the selected contributions	I can give the necessary feedback for contributions	Must have	2	50	5

	View the exceptional reports	I can choose the best articles	Should have	3	40	5
	Download all selected contributions as zip file	I can transfer them out of the system	Should have	3	40	3.5
	Keep track of the number of contributions, contributors and the percentage of contributions by multiple faculties	I can ensure the quality of the school magazine	Must have	3	40	5
Marketing coordinator	Receive email notifications from the system	I can be more convenient in working with contributions	Should have	2	30	2
	Make comments for my student's articles within 14 days after the system stops accepting new articles	I can help my students make the necessary updates	Must have	2	50	2.5
	View contributions from students in my faculty	I can review and evaluate the contributions collected	Must have	2	40	4
	Interact with the students in my faculty	I can make it easy to give suggestions, discuss necessary	Should have	3	40	4

		adjustments with my students				
	Select contributions to publish	I can contribute the best quality articles to the magazine	Must have	2	50	2.5
Administrator	Maintain any system data	I can store and preserve all system data	Must have	3	40	4
	Control update time, closing time	I can control the operation of the system	Must have	2	50	6.5
	Create new account and control login permissions	I can control different user accounts	Must have	1	50	15
Guest	View faculty selected reports	I can keep track of magazine contents	Must have	3	50	5.5

Table 14: Product backlog

2.2. Product burndown chart

Our product burndown chart, shown below, depicts our team's working process throughout each Sprint. It's clear that there's a big difference between fact and estimate. Because our team's Sprint 1 does not include a large number of tasks, the remaining story points at the end of Sprint 1 are quite large. However, in order to meet the project deadline, our team has decided to significantly increase the number of tasks that must be completed in Sprints 2 and 3 compared to Sprint 1. As a result, the remaining story points dropped more sharply from Sprint 1 onwards and remained consistent with the estimate until the end of the project.

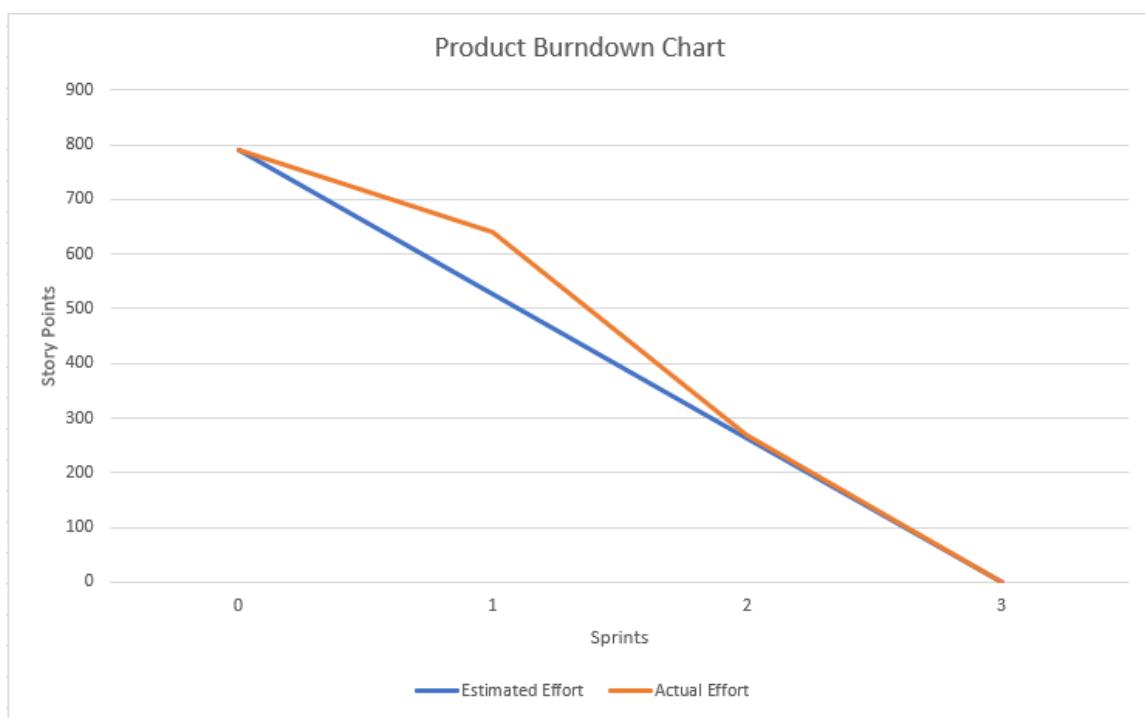


Figure 130: Product Burndown Chart

2.3. Team challenges

During the project's implementation, our team faced a variety of challenges, including:

Difficulty communicating among team members in the early stages of the project, specifically before the start of Sprint 1, when no one knew each other and had never worked together before. It took a long time for our team to get to know each other and feel comfortable enough to work together effectively, so the time spent on the project is significantly reduced

before it even begins. However, we have overcome this quite effectively by communicating with each other more actively through Facebook, working together with a strong sense of responsibility, and meeting and chatting on a regular basis when we come to class.

Furthermore, it can be said that the deadline of this project is also a big challenge for our team members, because when the project needs to be started, it is also the time when my country's biggest holiday of the year, the lunar new year, begins, so it was really difficult for everyone to immediately implement this project early. Furthermore, it was during this time that the Covid-19 epidemic in my country became more dangerous, making it difficult to meet and work face to face with each other. However, we have overcome this by accelerating all tasks, getting started on the project as soon as possible, and devoting more time to completing this project on time.

One of the challenges is a lack of experience, knowledge, and skills required to complete the project professionally, quickly, and smoothly. More specifically, having to work while learning and supplementing knowledge to meet the needs of the project not only takes more time for the team, but also leaves the project lacking and hazy. However, our team also attempted to learn the necessary knowledge and skills in coding, testing, and so on from a variety of sources. As a result, the project was carried out more effectively, and we gained new and valuable knowledge and experience.

Other difficulties arose for our team, such as the loss of a team member due to personal reasons, forcing him to leave this project. This has resulted in a significant loss of manpower for our team, as well as a significant increase in the workload that team members must undertake. However, we have overcome this difficulty by sharing the tasks that this member left behind and spending more time assisting each other in completing the various project tasks.

Other challenges arose for our team, such as the loss of a team member due to personal reasons, which forced him to leave this project. This has resulted in a significant loss of manpower for our team, as well as an increase in the workload that team members must perform. We overcame this difficulty, however, by sharing the tasks that this member left behind and spending more time assisting each other in completing the various project tasks.

Finally, it is worth noting that misinterpreting project requirements can be a challenge. This necessitates a significant amount of time and effort to correct and supplement the deficiencies, and it can sometimes have an impact on other functions and factors that have been successfully implemented in the project. However, we overcame this obstacle by consulting with our instructor and thoroughly discussing these requirements before implementing them.

2.4. Minutes of meetings

Meeting date	Activities take place during the meeting	Participants
Thursday, February 25, 2021	<ul style="list-style-type: none"> - Describe new functions implemented in Sprint1. - Describe the interfaces and interactions related to the functions implemented in Sprint1. - Describes the database implementation and the required data has been successfully created. - Describe the testing process for the different functions implemented in Sprint1. - Describe the elements that need to be fixed or improved in Sprint1. - Discuss the difficulties team members have encountered during Sprint1 and find solutions to overcome them. - Discuss the project implementation progress and the amount of time allocated to the remaining Sprints. - Identify the functions that need to be implemented in Sprint2. 	<ul style="list-style-type: none"> - Nguyen Hoang Thuyen - Pham Ngoc Bao - Do The Vinh - Nguyen Van Tuan - Do Ho Nguyen - Ha Ngoc Vy
Saturday, March 13, 2021	<ul style="list-style-type: none"> - Describe new functions implemented in Sprint2 and the old features that have been modified or improved. - Describe the interfaces and interactions related to the functions implemented in Sprint2 and the modifications or improvements to the old elements. 	<ul style="list-style-type: none"> - Nguyen Hoang Thuyen - Pham Ngoc Bao - Do The Vinh - Nguyen Van Tuan - Do Ho Nguyen - Ha Ngoc Vy

	<ul style="list-style-type: none"> - Describe the developments and edits made to the database. - Describe the testing process for the different functions implemented in Sprint2. - Discuss the adjustments made to the time and progress of the functions, elements in Sprint2. - Describe the elements that need to be fixed or improved in Sprint2. - Identify the functions that need to be implemented in Sprint3. 	
Saturday, March 27, 2021	<ul style="list-style-type: none"> - Describe the last remaining new functions, implemented in Sprint3, and old features that have been modified or improved. - Describe the interfaces and interactions related to the functions implemented in Sprint3 and the modifications or improvements to the old elements. - Describe the developments and edits made to the database. - Describe the testing process for the different functions implemented in Sprint3. - Describe the elements that need to be added, fixed or improved in Sprint3. - Review and identify all functions, elements that have worked well in the system and those that need to be improved or added to complete the project. 	<ul style="list-style-type: none"> - Nguyen Hoang Thuyen - Pham Ngoc Bao - Do The Vinh - Nguyen Van Tuan - Do Ho Nguyen - Ha Ngoc Vy

	<ul style="list-style-type: none"> - Discuss all the difficulties experienced during the project implementation to gain more experience when implementing future projects. - Determine what to do next to announce the results of the team's project implementation. 	
--	--	--

Table 15: Minutes of meetings

VIII. Conclusion

The process of building our website was completed on time and with all of the required criteria met. The website is stable, loads quickly, and is error-free. The interface is user-friendly and simple to use. We took it very seriously and were disciplined during the implementation process. Make a clear plan for each part, and divide the work in accordance with each individual's capacity. There are short meetings to learn about the situation and the progress of the work. Control the situation and intervene if problems arise. I always write down the content of each meeting and plan for the next phase. The member in charge of designing the database should be well-versed in this area and use appropriate technology. Designers do not work alone or independently; they constantly exchange and receive support from team members, as well as suggestions and upgrades to improve website functionality and design. The examiner will cross-check each other's answers. Examine each function as well as the overall performance of the website. After each test, comprehensive statistical reports are generated. When the implementation and development process is completed. Sit down and go over the website one

last time to ensure that it was implemented as planned, completed as required, the system was stable, and no errors occurred. Is it necessary to repair or improve it? That's how we're going to build and finish the website. Working together, dividing work according to each individual's abilities. Hold regular meetings to check on the status of the project. Always be there for one another, share and help one another, and grow together.