



Daffodil
International
University

TITLE OF THE PROJECT

Alumni Management System

Submitted By

MD.Anisuzzaman

(Id: 191-16-434)

Supervised By

Abdullah Bin Kasem Bhuiyan

Lecturer (DIU)

Summer 2021

Course: Internship

Course code: CIS499

Department of Computing and Information System (CIS)

DAFFODIL INTERNATIONAL UNIVERSITY

Submission Date: 16- November-2021

APPROVAL

This Project titled “**Alumni Management System**”, Submitted by **MD.ANISUZZAMAN**, ID No: **191-16-434** to the Department of Computing & Information Systems, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computing & Information Systems and approved as to its style and contents. The presentation has been held on-09-11-2021.

BOARD OF EXAMINERS



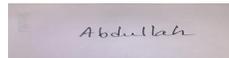
Mr. Md Sarwar Hossain Mollah
Associate Professor and Head
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University

Chairman



Ms. Nayeema Rahman
Sr. Lecturer
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Mr. Abdullah Bin Kasem Bhuiyan
Lecturer
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



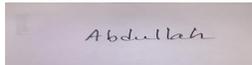
Dr. Saifuddin Md. Tareeq
Professor
Department of Computer Science and Engineering
Dhaka University, Dhaka

External Examiner

Declaration

I hereby declare that; this project has been done by me under supervision of Mr. Abdullah Bin Kasem Bhuiyan, Lecturer, department of Computing and Information System (CIS) of Daffodil International University. I am also declaring that this project or any part of there has never been submitted anywhere else for the award of any educational degree like, B.Sc., M.Sc., Diploma or other qualifications.

Supervised By

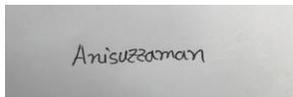


Mr. Abdullah Bin Kasem Bhuiyan

Lecturer

Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University

Submitted By



MD. Anisuzzaman

ID: 191-16-434

Department of CIS

Daffodil International University

Acknowledgement

For this project, we would like to thank all our instructors for teaching us little by little and making us ready to do this project since the first day in our institute. Thanks to almighty Allah, who forgives us physical and mental strength, and we are deeply thankful to our honorable teachers' Md. Mazharul Islam, and Abdullah Bin Kasem sir for inspiring us and guiding us for this massive project. Without their inspiration, we faced very difficulties finishing the project.

Executive Summary

This project is aimed toward developing a Repository and every Engine for alumni of the faculty that is of importance to a university. The Alumni info could be a web-based application that will be accessed throughout the globe (intelschool, 2014). Anyone will access the program to understand concerning any Alumni of that college. This system may be utilized as an alumni information application.

Abstract

Alumni Management System is associating online-based an application that helps to trace school graduates. The project aims to boost the present following procedure of faculty graduates and supply alumni information to varsity schools. It aims to develop an online portal that will be helpful for the school to watch the alumni and for the alumni to update their current standing concerning the school activities. The plan of this method is to produce an associate optimized resolution for aggregation and managing alumni information. The system is an associate all-in-one place to handle alumni information, to alter communication between alumni and therefore the establishment and provides nice versatile features to stay track of the info. The user will update their information by merely filling the forms and uploading their file. A new advantage of the planned system is that, generally, associate alumni forums are managed by a bunch of alumni which would be associate overhead to them, this method is machine-controlled hence doesn't need maintenance effort from the alumni side.

Table of Contents

Acknowledgement	i
Executive Summary	ii
Abstract	iii
Chapter 1 – Introduction	1
1.1 Introduction:.....	1
Chapter 2 – Initial Study	1
2.1 Background of the project:.....	1
2.2 Problem Area	2
2.3 Possible solution	3
Chapter 3 – Literature Review	3
Best Features:.....	6
Chapter 4 – Methodology	7
4.1 What to use	7
4.2 Why to use	10
4.3 Sections of methodology.....	10
4.4 Implementation plans.....	11
Chapter 5 – Planning	11
5.1 Project Plan	11
5.1.1 Management Plan / Work Breakdown Structure (WBS)	11
5.1.2 Gantt Chart.....	12
Chapter 6 – Feasibility	13
6.1 All Possible Types of Feasibility	13
6.2 Cost benefit Analysis	16
6.3 DSDM – good or not for this project	17
Chapter 7 – Foundation	18
7.1 Overall Requirement List.....	18
7.2 Technology to be implemented:.....	19
Chapter 8 – Exploration	21
8.1 Full System Use Case	21
8.2 Full System Activity Diagram	22
Chapter 9 – Engineering	24

9.1 Case Diagram of the Alumni Management System	24
9.2 ERD Diagram of the Alumni Management System.....	25
9.3 Sequence Diagram of the Alumni Management System	26
Chapter 10 – Deployment / Development	27
10.1 Core Module Coding Samples:	27
10.2 Possible problem break down	36
Chapter 11 – Testing	37
11.1 Test Case.....	37
11.2 Unit test – test case:	38
11.3 Module Test – test case:.....	41
11.4 Integration Testing – test case:.....	44
Chapter 12 – Implementation	44
12.1 Training.....	44
12.2 Big Bang	45
Chapter 13 – Critical Appraisal and Evaluation	45
13.1 Success rate against each objective.....	45
13.2 How much better could have been done	45
13.3 How better is the features of the solution.....	46
13.4 Which features could not be touched	46
Chapter 14 – Conclusion	46
14.1 Summary of the project.....	46
14.2 Goal of the project	46
14.3 My experience.....	47
Works Cited	48
Appendices	50
User Guide:	50
System Code-	62

List of Figures

Figure 1: Brac University, Bangladesh.....	5
Figure 2: East West University, Bangladesh	6
Figure 3: SDLC - Waterfall Model.....	7
Figure 4: Dynamic System Development Method (DSDM).....	9
Figure 5: Work Breakdown Structure (WBS).....	12
Figure 6: Gantt chart	12
Figure 7: Client Server application	20
Figure 8: Admin Activity Diagram.....	21
Figure 9: System Activity Diagram Admin	22
Figure 10: System Activity Diagram Alumni	23
Figure 11: Case Diagram of the Alumni Management System	24
Figure 12: ERD Diagram of the Alumni Management System	25
Figure 13: Sequence Diagram of the Alumni Management System.....	26
Figure 14: alumni registration.....	27
Figure 15: Alumni registration.....	28
Figure 16: Front in Alumni registration.....	31
Figure 17: Admin Dashboard code	32
Figure 18: Admin Dashboard py file	33
Figure 19: Event creates code	34
Figure 20: Event creates.....	35
Figure 21: Access of this user.....	35
Figure 22: Alumni Register	39
Figure 23: Alumni Register successfully	40
Figure 24: Admin panel	41
Figure 25: User registration	42
Figure 26: Admin registration.....	43

List of Tables

Table 1: Cost of a web-based application	15
Table 2: Cost of a Mobile App.....	15
Table 3: Cost of Desktop Application.....	15
Table 4: Estimation of the project's total cost	16
Table 5: Estimating the project's earnings	17
Table 6: Estimated Income on a Four-year scale	17
Table 7: Unit Test 1	38
Table 8: Unit Test 2	39
Table 9: Unit Test 3	40
Table 10: Module Test 1	42
Table 11: Module Test 2	43
Table 12: Integration Testing – test case	44

Chapter 1 – Introduction

1.1 Introduction:

A communication establishment between graduates and current students may be very useful in several things. As an example, the establishment is also fascinated by extending a request for a guest lecture to one of its alumni who could be appallingly sure-fire bourgeois currently. However while not having correct contact the establishment can pay time tracing the contact. Thus, maintaining consistent information goes hand-in-hand with establishing correct communication (studentprojects, 2021). The project of this method is to assist in accomplishing this, wherever alumni will communicate among themselves, with their juniors, and with the organization. In this fashion, they will build a robust network of execs. This method proves to be helpful to this student of the establishment since they get the contacts of their seniors from whom they can rummage around for opportunities and steerage (technodg, 2019).

The admin will communicate with the alumni furthermore and be notified of any update created by the users to their profile through weekly reports. Except for communication and admin reports, the system conjointly permits users to form events (studentprojectguide, 2015). The approaching events would be displayed on the home page; therefore, the user would be reminded of the events to be happening in order that they might not miss out on any vital event.

Chapter 2 – Initial Study

2.1 Background of the project:

Each year numerous students graduated from their individual universities or schools. It's is incredibly arduous for the colleges and universities to take care of their comprehensive knowledge and end in lack of communication between alumni and their school. Institutes can also facilitate current students by arranging regular events with alumnus. These events can easily be maintained by a software system where alumnus and students can virtually plan and organize the events (academia, 2020). Some graduates can start their business right after graduation, but

for schools to know that this alum is a good fit for their faculty, they must have a connection with them.

2.2 Problem Area

The majority of grad student wasn't connection Alumni association member, rather a number of them extremely don't fathom this method. This development occurred in our University. Though it's its own administration workplace, it's long-faced with plenty of issues and doesn't work properly. This was of association system is insufficient and doesn't run within the right method. We can usually establish the issues that are chiefly as:

- Service related problem and
- Human power related problem

Service related problem: Because most graduates travel to various parts of the country and speak different languages, there is a challenge with promoting scholarships and other educational opportunities to students who live far away from the institution. Interaction with an alumna in their local region necessitates a lot of services as well (ijert, 2018). As a result of entering into a diverse culture, this alumni office may find it challenging to provide support to such graduating students.

Human capacity related problem: It is time-consuming to retrieve student data, such as a graduate student's personality, address, and other students' information from the institution. Retrieving the status of graduates who have entered the workforce requires a significant amount of strength. After they leave, the alumnus is not initiated to be enrolled as a member. I.e. the system is physically operated (www.researchgate.net, 2017). Even if one is a member, it is difficult to keep track of the organization, and there is no effective service that enables daily activation of both memberships, as most significant information is put on the notice board. It's also tough to figure out who is actively involved in the organization.

2.3 Possible solution

The alumni association design and implementation will be accessible to the general public via a web interface. A non-registered visitor can browse the list of graduates by graduation year or subject of study. He can also glance at the profiles of grads. The amount of information on a person's profile that is visible to the general public is limited. A public visitor can only view a graduate's name and surname, year of graduation, and field of study by default. The faculty works hard to spread the word about its alumni. As a result, during their studies, graduates can submit information about themselves to the system, such as job experience and knowledge. Graduates can choose to make this information visible to the public on their profiles.

Keyword: Alumni, System, Student, Management, Dashboard

Chapter 3 – Literature Review

The literature review provides the general notion to comprehend a specific issue for the development purpose of the system. A project literature study aids in the identification of problem areas and their solutions (inettutor, 2011). This component facilitates a prospective conversation about the study as well as a comparison within the same work area, paving the path for a successful project.

3.1 Discussion on problem domain based on published articles.

The problem domain is based on the distance based of ex-students communication. We know, every university have an alumni association where every student gets together every years and communicate each other's they need their purposes and collected bloods, and universities gives job opportunities for their brilliant students (link, 2010). If any university have not an alumni association they will not proper communicate with students and those universities cannot successful and their reputation will be fallen.

3.2 Discussion on problem solutions based on published articles.

There are several types of problem solutions. Those are given below:

In my personal opinion, I think every university needs to celebration alumni days for good communication each other's students. Here, ex-students advices how to develop proper skill for good jobs opportunities and how to study in universities for their future career. Ex-students create well bonding in all studying students by the alumni association (sciencedirect, 2026).

3.3 Comparison of leading solutions-

Here I discuss 2(two) leading solution. Those discussed in below:

1. Brac University, Bangladesh.

This University is a well-known university in our country. This is an outstanding achievement. However, in the global higher education landscape, we go overlooked. We lag behind the best in the world in terms of procedures and standards (www.bracu.ac.bd, 1995).

<https://www.bracu.ac.bd/>



Figure 1: Brac University, Bangladesh.

Best features

- ✓ Online Apply
- ✓ Finding program
- ✓ News and notices

Limitations

- ❖ There isn't a lot of functionality in the system interface.
- ❖ Privacy and security of data are both lacking.
- ❖ There is no infrastructure in place to handle requests in real-time or to provide notifications.

2. East West University, Bangladesh

<https://alumni.ewubd.edu/>

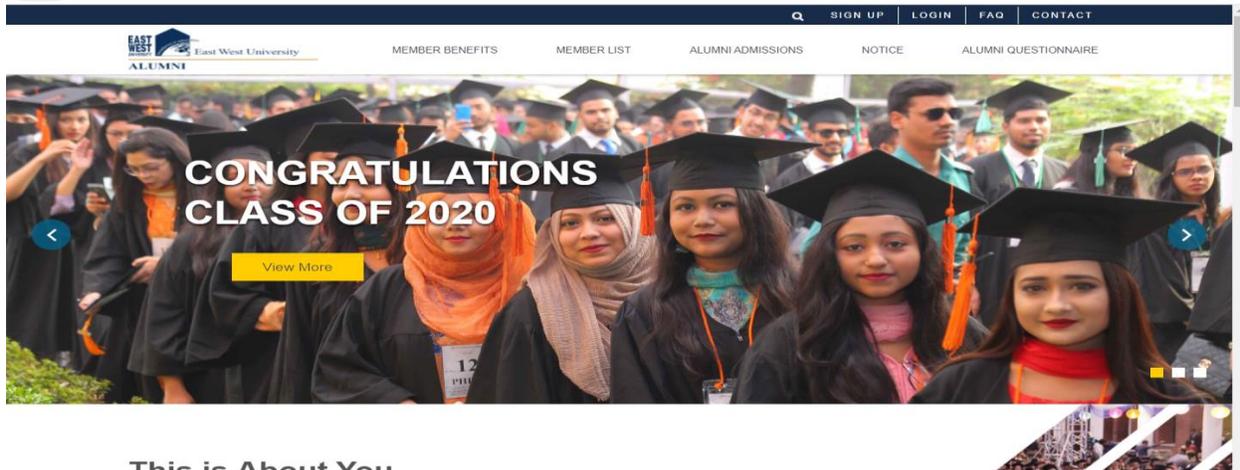


Figure 2: East West University, Bangladesh

Best Features:

- ✓ Registration and Log in processes.
- ✓ Location access
- ✓ Notices

Limitations

- ❖ The system's function isn't particularly dynamic
- ❖ The interface isn't standard.

3.4 Recommended approach

By observing stages, I discovered a few highlights that should be remembered for my system. Those are given below:

- i. The user interface and navigation bar should be appealing.
- ii. Only registered alumni students are permitted to attend.
- iii. Any event can only be created by alumni students.
- iv. Alumni students can start a fund for poor students.

Chapter 4 – Methodology

The software team that works for the best of the project chooses software development techniques for a successful project and effective project management. Each methodology has its own set of strengths and weaknesses, and each one ends for a different reason. The success rate of development is frequently determined by the approach used for development aims. With sufficient study and knowledge (sparklyn, 2021), this chapter will comprehensively examine the chosen methodology and the right explanation for adopting the technique.

4.1 What to use

For product development, a variety of methodologies are employed. Because the project is IT-related, the agile technique would be beneficial. In many prior multinational initiatives, the agile approach was important in the project's success. The DSDM framework is one of the agile methodologies that may be applied. I'll outline two methodologies, together with their benefits and drawbacks.

SDLC - Waterfall Model

A waterfall model is exemplified by a sequential model. The software development activity is organized into distinct phases in this paradigm, with each phase consisting of a sequence of activities and achieving different goals (tutorialspoint, 2015).

The several phases of the Waterfall Model are depicted in the following diagram:

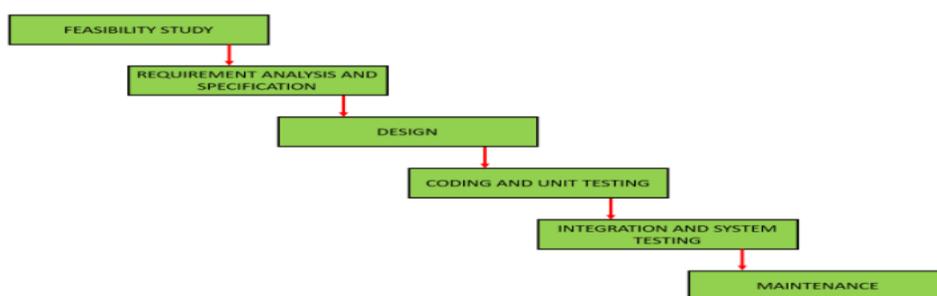


Figure 3: SDLC - Waterfall Model

Waterfall Model – Advantages

- It's easy to grasp and implement.
- Thanks to its stiffness, the model is straightforward to handle. There are precise deliverables and a review mechanism for every step.
- Every step is restricted and accomplished one by one.
- This methodology is best suited to smaller come with well-defined specifications.
- Stages that is distinct and well-defined.
- Milestones that are clear and well-defined (studentprojects, 2021).

Waterfall Model – Disadvantages:

- Until the tip of the life cycle, no functioning software package is developed.
- There may be a heap of danger and uncertainty.
- For refined and object-oriented applications, this is often not associate degree acceptable paradigm.
- For long-run comes, this paradigm is insufficient.
- Not acceptable for comes with a moderate to high risk of modification in needs. As a result, this method model contains a high level of risk and uncertainty. Changes in requirements cannot be accommodated.
- Changes in needs can't be accommodated (studentprojects, 2021).
- Changing the scope of a project throughout its life cycle would possibly cause its termination.

Dynamic System Development Method (DSDM)

The dynamic system development technique is a way for quickly developing applications. Throughout the iterative and incremental process, this method places a premium on simultaneous user involvement (digite, 2016). With continual customer feedback, DSDM always takes the right approach to generating a fantastic project.

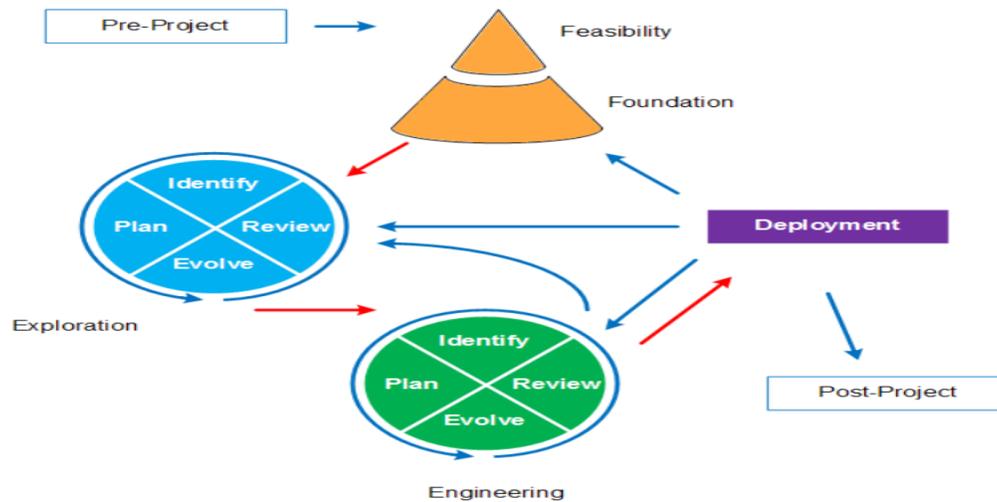


Figure 4: Dynamic System Development Method (DSDM)

Advantages of the DSDM:

- There is a high level of user interaction.
- Basic functions are delivered more swiftly and on a more consistent basis.
- Projects are finished on time and on budget.
- Enables developers to communicate with end-users (ukessays, 2001).

Disadvantages of the DSDM:

- One-time projects and small enterprises are not eligible.
- It is not as well recognized or understood as earlier classic versions such as the waterfall because it is a newer model.
- When compared to other agile development software approaches, DSDM's strictness and eight principles may make it challenging to work with. (ukessays, 2001).

4.2 Why to use

There are several frameworks for using IT-based projects, the most popular of which is Agile Methodology. It's because the agile technique is designed to integrate into all aspects of the project and ensure that all jobs are completed flawlessly. The Dynamic System Development Method (DSDM) will be the appropriate framework to apply for this project since it focuses on stakeholder requirements, deadlines, and quality control, among other things. These will assist the stakeholder in completing the project on schedule and putting the product to use in the real world.

4.3 Sections of methodology

The DSDM framework is divided into numerous components. They'll be mentioned farther down.

Pre-project Phase

The first project concept, as well as the timeframe, budget, and basic needs, are developed in this part.

Feasibility Study Phase

This part assesses the economic, technical, and operational feasibility of the proposed project.

Requirement Gathering Phase

Various strategies are used to identify both functional and non-functional project needs in this area.

Requirement Analysis and Prioritization Phase

Using a prioritizing approach such as MoSCoW, the acquired needs are analyzed and prioritized in this step.

Exploration and Engineering Phase

The methodology's major iterative portion was used to iteratively examine the requirements and design the solution.

Review Phase

The created deliverables are tested with users in this step, and if adjustments are required, they are returned to the previous phase (libguides, 2014).

4.4 Implementation plans

At the conclusion of the project, the produced application is made available for usage. Before the new system can be utilized, any issues with it must be recognized and remedied. The release criteria, settings, and techniques are all determined in this section. The new system will be implemented if all goes well.

Chapter 5 – Planning

5.1 Project Plan

A plan is required for the completion of any project. A project plan outlines the project's objectives and goals, as well as the activities and methods for achieving them. It also considers the resources that will be necessary, as well as the financial and timeframes for completion. A project plan outlines all of the work that will be done on a project and distinguishes who will be responsible for it. In general, project plans, including work breakdown structure, time boxing, and a Gantt chart, will be discussed in this section. The following are some of them:

5.1.1 Management Plan / Work Breakdown Structure (WBS)

WBS (Work Breakdown Structure) is essential work for each project. The Work Breakdown Structure has broken down the project's tasks into all of its components within a set time frame. The Work Breakdown Structure will be the most effective technique for completing the complete project by completing each assignment.

Title	T/M	Start	End	Duration	%
Introduction	T	06/16/2021	06/24/2021	8 days	100
Initial Study	T	06/25/2021	07/03/2021	7 days	40
Literature Review	T	07/04/2021	07/09/2021	5 days	20
Methodology	T	07/10/2021	07/21/2021	11 days	30
Planning	T	07/22/2021	08/04/2021	12 days	40
Feasibility Study	T	08/05/2021	08/12/2021	7 days	10
Foundation	T	08/13/2021	08/19/2021	6 days	50
Exploration	T	08/20/2021	08/31/2021	10 days	40
Engineering	T	09/01/2021	09/13/2021	11 days	25
Deployment	T	09/14/2021	10/03/2021	17 days	22
Testing	T	10/04/2021	10/10/2021	6 days	30
Implementation	T	10/11/2021	10/16/2021	5 days	30
Critical Appraisal and Evaluation	T	10/17/2021	10/19/2021	3 days	20
Conclusion	T	10/20/2021	10/20/2021	1 day	80

Figure 5: Work Breakdown Structure (WBS)

5.1.2 Gantt Chart

One of the most useful charts in project development is the Gantt chart. A detailed chronology for the project is presented in this graphic, as well as durations.

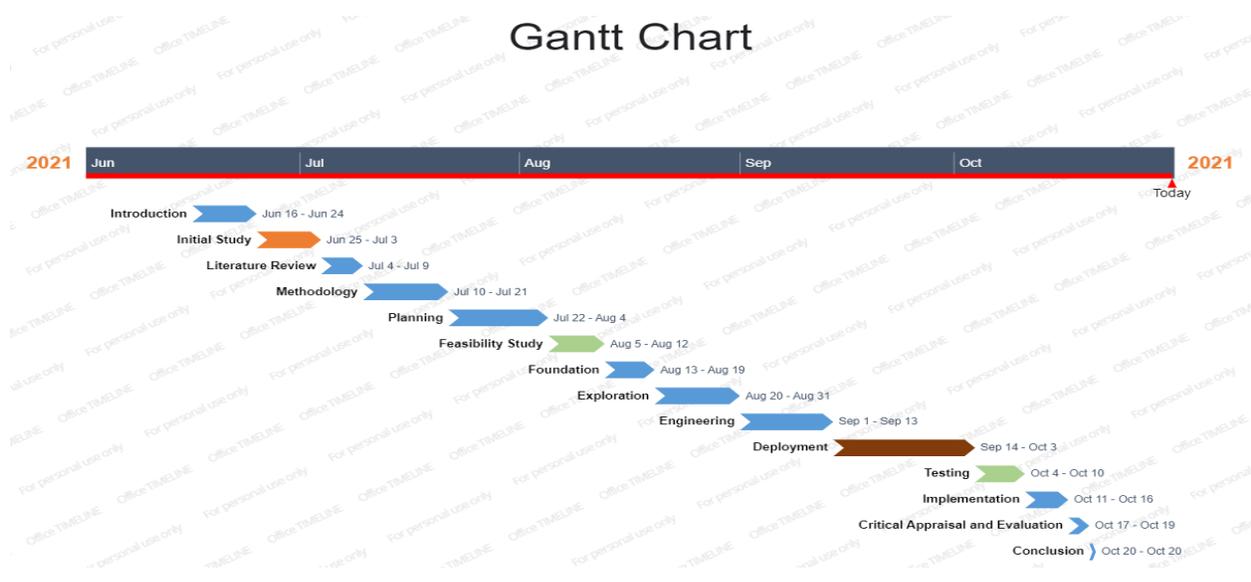


Figure 6: Gantt chart

Chapter 6 – Feasibility

6.1 All Possible Types of Feasibility

Operational feasibility, technological feasibility, and economic feasibility are examples of conceivable feasibilities.

Operational Feasibility:

This feasibility is well-designed user interface and functionality, which displays only the items that are required; this system is less expensive to operate than others. It also took less time to run it from a server and do maintenance on it. Ex-students can access the system from any operating system without the need to install any add-ons or other programs.

Technical Feasibility:

Users of the Alumni Management System may simply manage their activities, according to a technical feasibility test, as opposed to the prior manual method. The system was built and developed utilizing the most up-to-date and popular web and mobile technologies, ensuring that it can work in any web browser with a low-bandwidth Internet connection. To ensure safe access to system data, the system contains an access control hierarchy (simplilearn, 2013). The web-based application is platform-independent, and the development platform and resources used are both reasonable and cost-effective, ensuring that it is technically possible.

As a result, the technical features of this project are explicitly listed below:

Hardware:

- ❖ ASUS laptop (Configuration)
- ❖ Wi-Fi Router

Software:

- ❖ pgAdmin
- ❖ Google chrome browser

- ❖ Windows 10(operating system)
- ❖ PyCharm Community Edition

Database:

- ❖ PostgreSQL

Technology:

- ❖ Html
- ❖ XML
- ❖ CSS
- ❖ Java Script
- ❖ Bootstrap
- ❖ Odoo
- ❖ Python
- ❖ Django

Economic feasibility

Web-based application, mobile application, desktop application, and other options are available for developing the suggested system.

- **Cost of a web-based application:**

The whole program and data are stored on a remote server, which can be accessed through the Internet from anywhere at any time.

Equipment	Cost per unit	Price
Laptop (core i5, 3.6 GHz processor, 8GB DDR4 RAM, 1 TB HDD)	₹65,000	₹65,000
File, Web and Email servers	₹13,000 per month	₹13,000
Internet and VPN	₹4000 per month	₹4000

Total Price	₹82,000
-------------	---------

Table 1: Cost of a web-based application

- **Cost of a Mobile App:**

The system's app must be installed on every device, and alumni association software must be developed. Which may be more expensive than other options .

Equipment	Cost per unit	Price
Laptop (core i5, 3.6 GHz processor, 8GB DDR4 RAM, 1 TB HDD)	₹65,000	₹65,000
File, Web and Email servers	₹13,000 per month	₹7,000
Total Price		₹72,000

Table 2: Cost of a Mobile App

- **Cost of Desktop Application:**

Equipment	Cost per unit	Price
Laptop (core i5, 3.6 GHz processor, 8GB DDR4 RAM, 1 TB HDD)	₹65,000	₹65,000
File, Web and Email servers	₹13,000 per month	₹7,000
Total Price		₹72,000

Table 3: Cost of Desktop Application

6.2 Cost benefit Analysis

The primary goal of a cost-benefit analysis is to determine the earnings and spending assumptions. The entire cost and earnings are compared to determine the benefit. The following is a cost-benefit analysis for this project:

Total budget:

SL no.	Equipment	Year 1	Year 2	Year 3	Year 4	Total
1	Web-based application Expenses	₹82,000	-	-	-	₹82,000
2	Mobile App Expenses	₹72,000	-	-	-	₹72,000
3	Desktop Application Expenses	₹72,000	-	-	-	₹72,000
4	Domain and hosting Expenses	₹25,000	₹25,000	₹25,000	₹25,000	₹1,00,000
5	Employee Expenses	₹30,000	₹30,000	₹30,000	₹30,000	₹1,20,000
6	Other Expenses	₹20,000	₹20,000	₹20,000	₹20,000	₹80,000
	Total Cost	₹3,00,000	₹75,000	₹75,000	₹75,000	₹5,26,000

Table 4: Estimation of the project's total cost

Total Earn:

SL no.	Earnings Sector	Year 1	Year 2	Year 3	Year 4	Total
1	Government Tax 1	₳1,40,000	₳1,70,000	₳2,00,000	₳2,30,000	₳7,40,000
2	Government Tax 2	₳1,90,000	₳2,10,000	₳2,50,000	₳2,80,000	₳9,30,000
	Total Cost	₳3,30,000	₳3,80,000	₳4,50,000	₳5,10,000	₳16,70,000

Table 5: Estimating the project's earnings**Total Income:**

SL no.	Equipment	Year 1	Year 2	Year 3	Year 4	Total
1	Total Earn	₳3,30,000	₳3,80,000	₳4,50,000	₳5,10,000	₳16,70,000
2	Total Equipment Cost	₳3,00,000	₳75,000	₳75,000	₳75,000	₳5,26,000
	Total Income	₳6,30,000	₳4,55,000	₳5,25,000	₳5,85,000	₳21,96,000

Table 6: Estimated Income on a Four-year scale

As a result, it is apparent that by promoting this method, the government would be able to collect a large number of taka in income each year. Each year, the revenue will be raised. As a result, this initiative will benefit both the government and the general public.

6.3 DSDM – good or not for this project

The DSDM will suffice for this project since, as previously said, this framework is extensively utilized for project development and is successfully employed across the world. Other frameworks, such as Scrum and Extreme Programming (XP), are also utilized to build projects, but because this will be centered on client demand, DSDM will be chosen by everybody.

Chapter 7 – Foundation

7.1 Overall Requirement List

Two types of requirements:

- i) Functional requirements.
- ii) Non-functional requirements.

Functional requirements: The main thing of functional requirements is the type which is must need for the proposed system. The main functionality is to check data entry into the system is going the proper way or not. Functional requirements of this alumni management system are:

- Student Registration System
- Users have the ability to login, logout, and change their password.
- Contact information may be seen and modified by users.
- Student Registration this alumni page
- User accounts can be created, modified, deleted, viewed, and searched by administrators.
- Administrators have the ability to give, revoke, view, and search permissions.
- Areas can be created, modified, deleted, viewed, and searched by administrators.
- Administrators post events, accept registration requests, not approved request, manage alumni list, can approved event, can reject event, Post an event
- Administrators manage alumni list

Non-functional requirements:

- Data security and privacy are essential.
- Make sure you have a backup of your data and resources.
- Make the UI easy to use.
- The system's security
- To restrict access, implement authentication and authorization.
- Maintainability
- The system's Validation and verification
- To restrict access, implement authentication and authorization.

7.2 Technology to be implemented:

The technology that will be used is web technology, which is a server-based system that anyone will be able to access over the internet. Web technology is extremely beneficial nowadays since it can be accessed from anywhere in the globe without the need to install any software. The web system is divided into two sections, one for the administrator and another for the customer. Both parts are sufficiently informational, and the interface design is reasonable.

Client Server application:

A client-server application is a component of a program that splits responsibilities between servers and clients. The client-Server program sends a request to the distant server for permission to utilize the data stored on the client computer (omnisci, 2015). The following are some basic notions concerning client-server applications:

- 1) A client-server architecture that works with both explicit and cross-stage programming languages.
- 2) It's impossible to reach out to a large number of clients at once.
- 3) Requires client-side setup.
- 4) Client-server is expensive and risky to use since requests cannot be completed if the server goes down (omnisci, 2015).

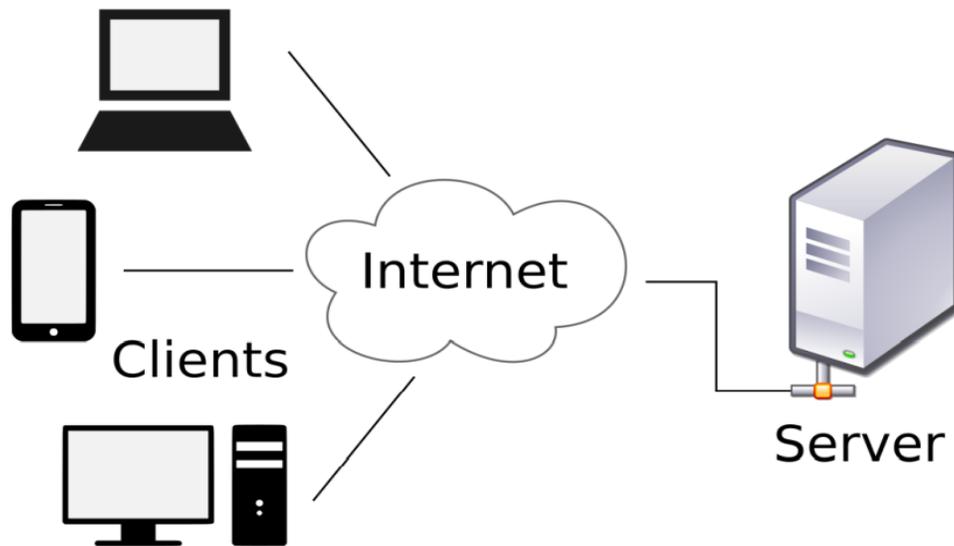


Figure 7: Client Server application

Web application:

A web application is a program that runs on a web browser and has a unique domain name. It collects orders from clients using the site's domain name. It doesn't have any stages. The following are some high-quality web applications:

- i. It makes use of a multi-level architecture.
- ii. There is no need for any kind of establishment.
- iii. Accessible from a variety of browsers (askanydifference, 2015).

Chapter 8 – Exploration

8.1 Full System Use Case

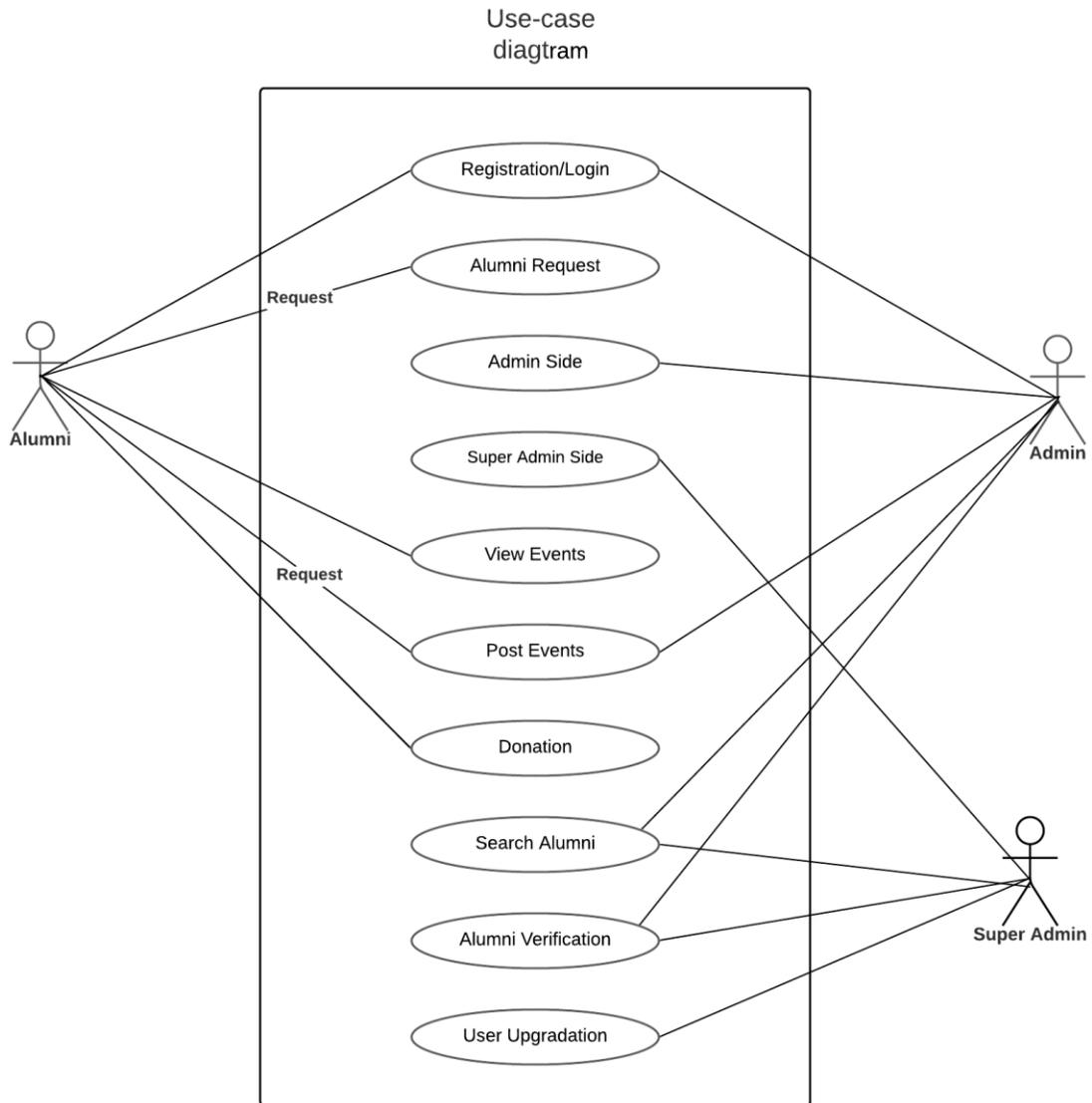


Figure 8: Admin Activity Diagram

8.2 Full System Activity Diagram

There are several sorts of users in the proposed alumni management system, each with its own workflow. Below are the activity diagrams for various work processes.

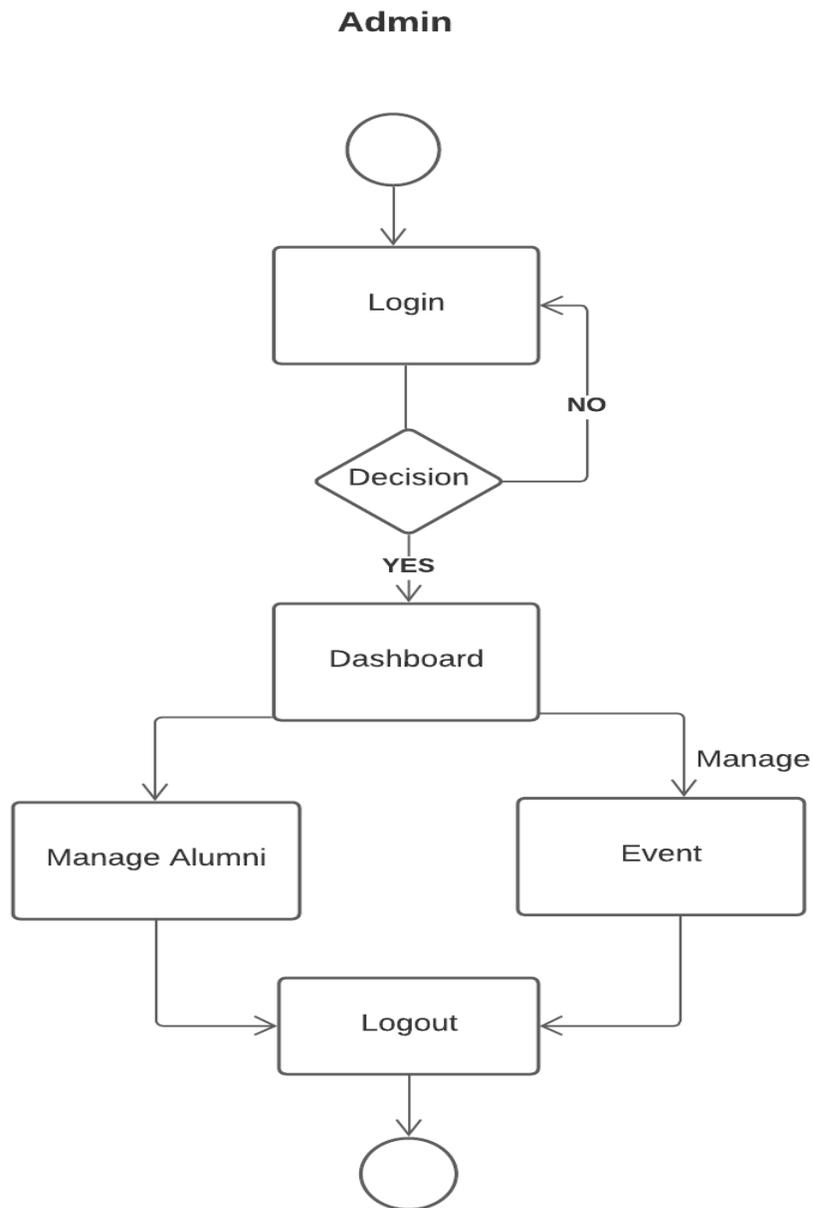


Figure 9: System Activity Diagram Admin

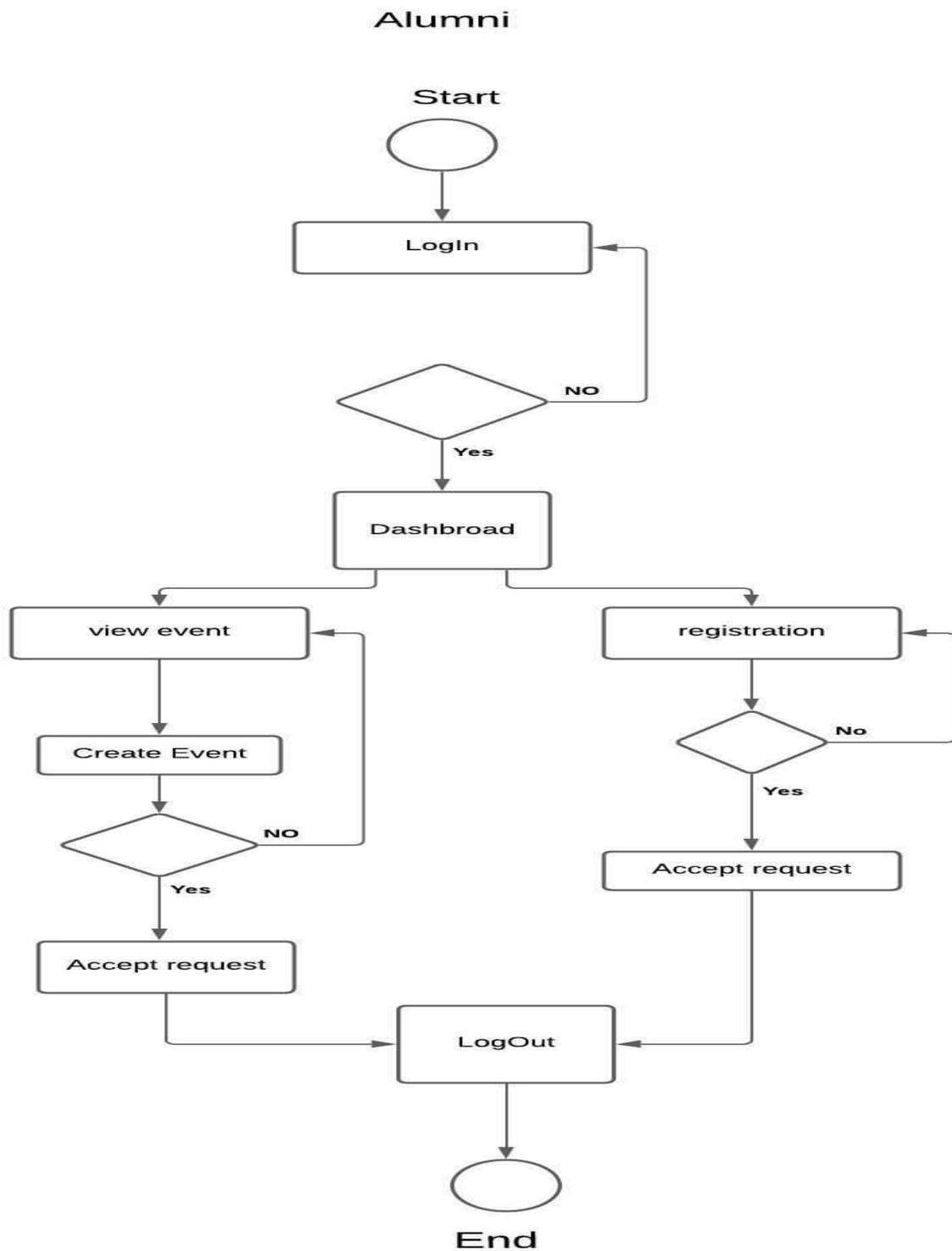


Figure 10: System Activity Diagram Alumni

Chapter 9 – Engineering

9.1 Case Diagram of the Alumni Management System

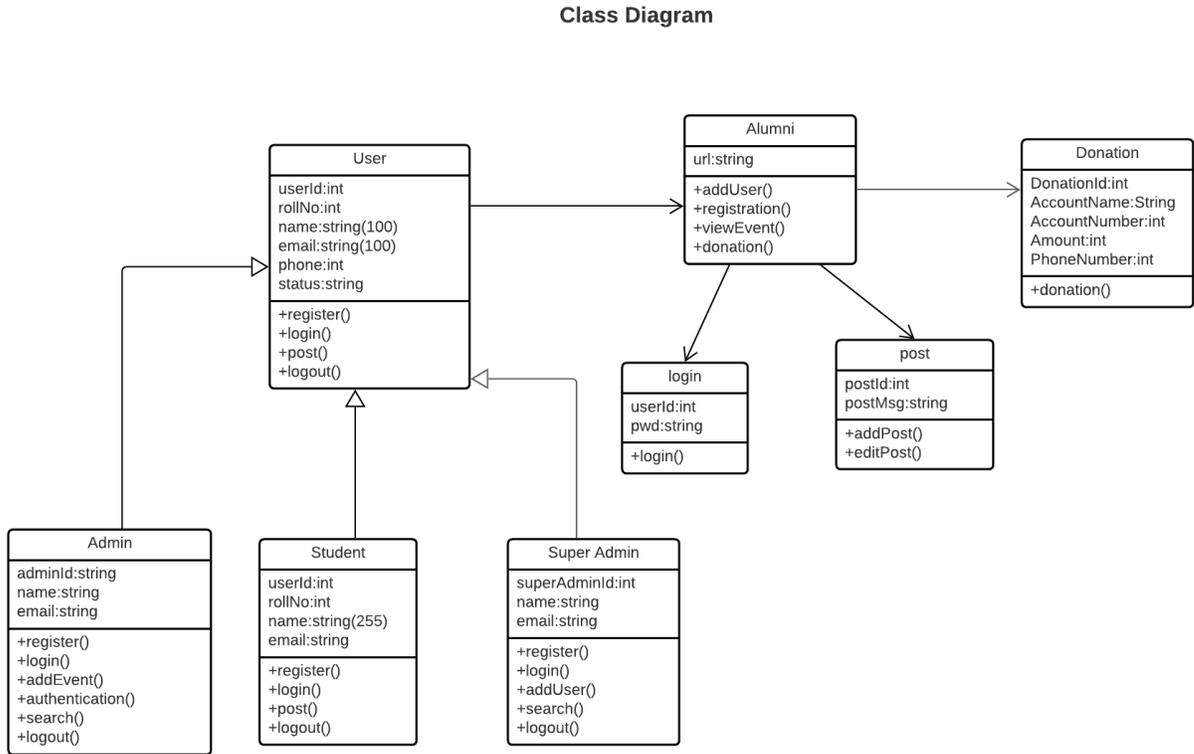


Figure 11: Case Diagram of the Alumni Management System

9.2 ERD Diagram of the Alumni Management System

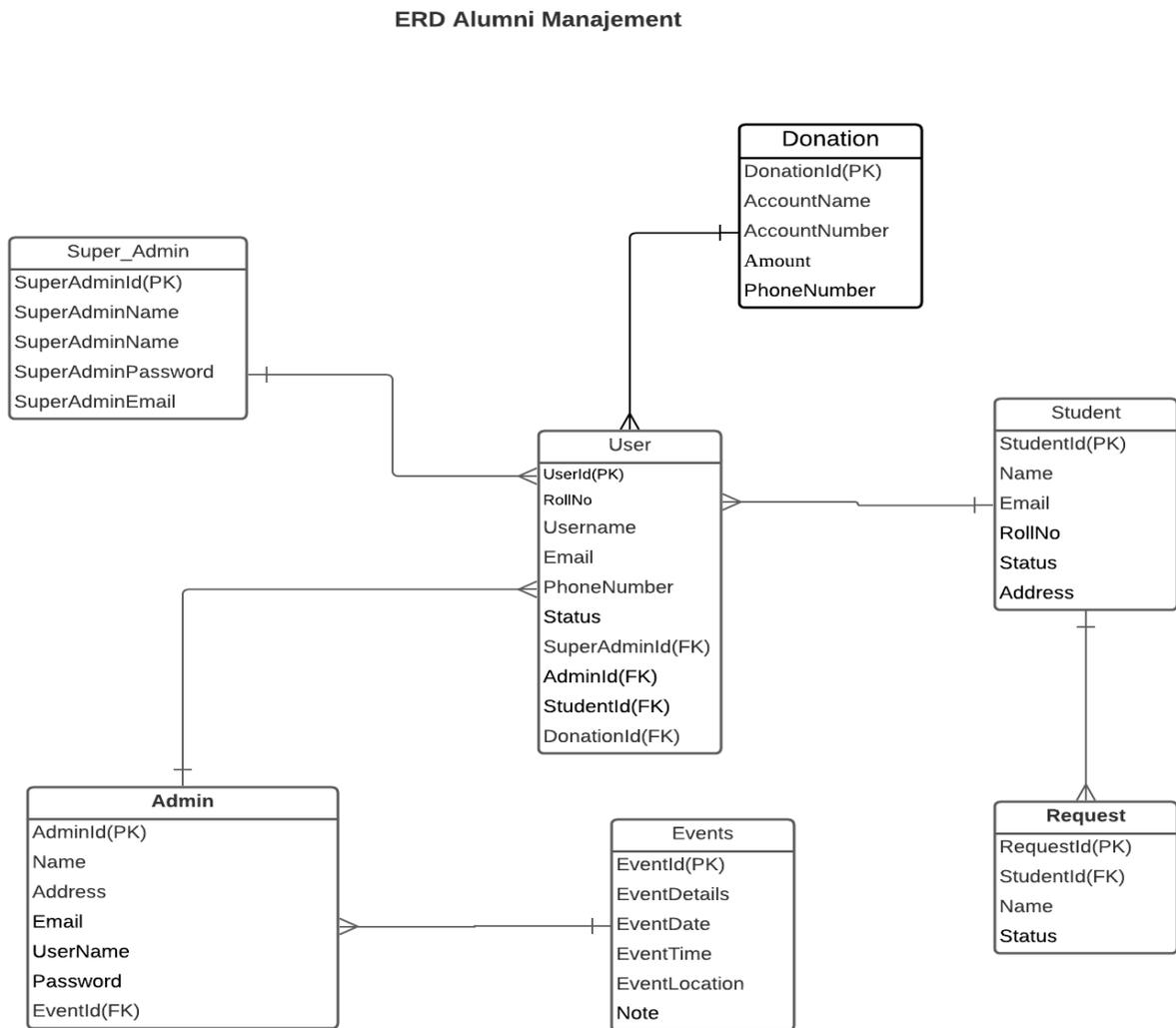


Figure 12: ERD Diagram of the Alumni Management System

9.3 Sequence Diagram of the Alumni Management System

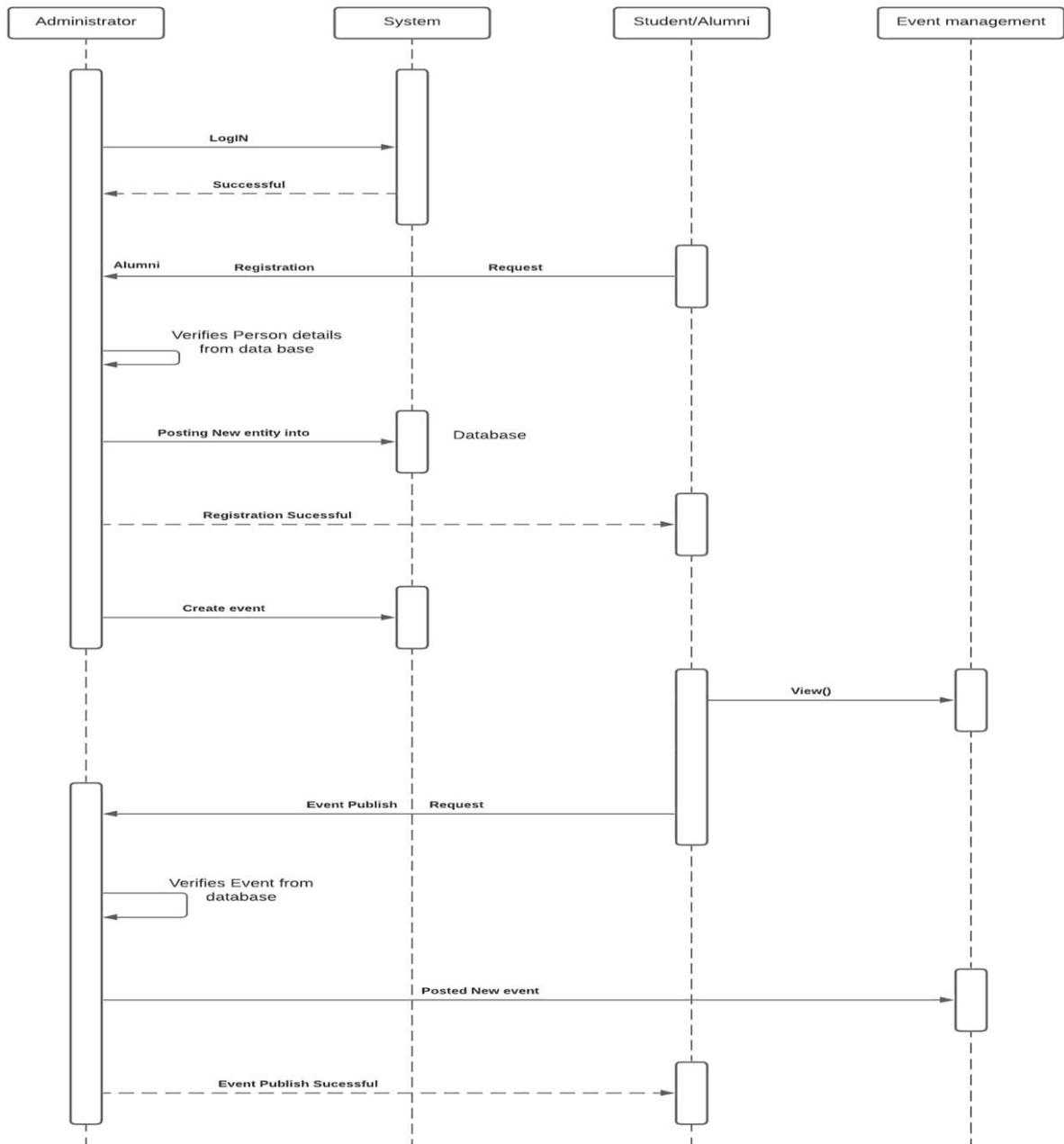


Figure 13: Sequence Diagram of the Alumni Management System

Chapter 10 – Deployment / Development

10.1 Core Module Coding Samples:

In this section, I'll teach you how to code numerous page core modules.

Registration sample for the users-

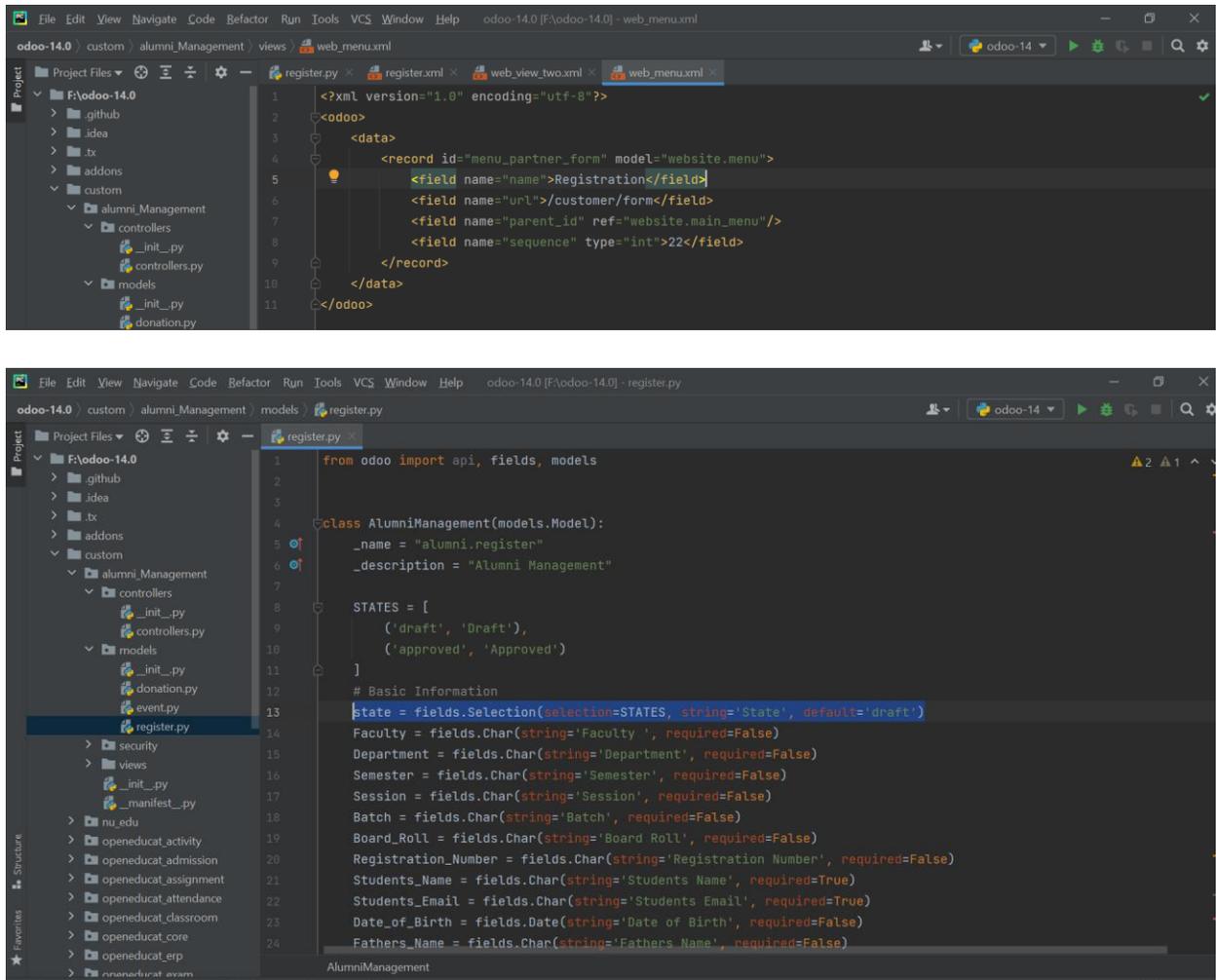


Figure 14: alumni registration

```

25 Mothers_name = fields.Char(string='Mothers name', required=False)
26 Student_Phone_Number = fields.Char(string='Student Phone Number', required=False)
27 Fathers_Phone_Number = fields.Char(string='Fathers Phone Number', required=False)
28 Mothers_Phone_Number = fields.Char(string='Mothers Phone Number', required=False)
29 Local_Guardian_Phone_Number = fields.Char(string='Local Guardian Phone Number ', required=False)
30 Result = fields.Char(string='Result', required=False)
31 Present_Address = fields.Char(string='Present Address', required=False)
32 Permanent_Address = fields.Char(string='Permanent Address', required=False)
33 Interested_Areas = fields.Char(string='Interested Areas', required=False)
34 # Other information (Social Media)
35 Facebook = fields.Char(string='Facebook ', required=False)
36 LinkedIn = fields.Char(string='LinkedIn', required=False)
37 Youtube_Channel = fields.Char(string='Youtube_Channel', required=False)
38 Personal_Website = fields.Char(string='Personal_Website', required=False)
39 # Internship data
40 Company_Name = fields.Char(string='Company Name ', required=False)
41 Company_Address = fields.Char(string='Company Address', required=False)
42 Company_Responsible_person_contract_number = fields.Char(string='Company Responsible person contract number',
43 required=False)
44 Duration = fields.Char(string='Duration', required=False)
45 Company_Phone = fields.Char(string='Company Phone', required=False)
46 # Higher Education
47 University_Name = fields.Char(string='University Name ', required=False)
48 Subject_Name = fields.Char(string='Subject_Name', required=False)

```

```

46 # Higher Education
47 University_Name = fields.Char(string='University Name ', required=False)
48 Subject_Name = fields.Char(string='Subject_Name', required=False)
49 # Industry category
50 Industry_category = fields.Selection([
51
52 ('Education/Training', 'Education/Training'),
53 ('Electrical/Electronics', 'Electrical/Electronics'),
54 ('Garments/Textile', 'Garments/Textile'),
55 ('Design/Creative', 'Design/Creative'),
56 ('Production/Operation', 'Production/Operation'),
57 ('Hospitality/ Travel/ Tourism', 'Hospitality/ Travel/ Tourism'),
58 ('IT/Telecommunication', 'IT/Telecommunication'),
59 ('Medical/Pharma', 'Medical/Pharma'),
60 ('Real Estate/ Development', 'Real Estate/ Development'),
61 ('Architecture', 'Architecture'),
62 ('Accounting/Finance', 'Accounting/Finance'),
63 ('Web Developer', 'Web Developer'),
64 ('Freelancer', 'Freelancer'),
65 ], required=False, default='Education/Training')
66 # Previous Employment
67 Company_Industry_Name = fields.Char(string='Company/Industry Name', required=False)
68 Designation = fields.Char(string='Designation', required=False)

```

Figure 15: Alumni registration

```

65 ], required=False, default='Education/Training')
66
67 # Previous Employment
68 Company_Industry_Name = fields.Char(string='Company/Industry Name', required=False)
69 Designation = fields.Char(string='Designation', required=False)
70 Company_Address_A = fields.Char(string='Company Address', required=False)
71 Company_Phone_TNT_Number = fields.Char(string='Company Phone/TNT Number', required=False)
72 Duration_Joining_Date_Resign_date = fields.Date(string='Duration: Joining Date & Resign date ', required=False)
73 Duration_Joining_Date_Resign_date_C = fields.Date(string='Duration: Joining Date & Resign date C ', required=False)
74 CV = fields.Binary(string='CV', required=False)
75 Video_Resume = fields.Binary(string='Video Resume', required=False)
76
77 # Current Employment
78 Company_Industry_Name_A = fields.Char(string='Company/Industry Name ', required=False)
79 Designation_A = fields.Char(string='Designation', required=False)
80 Company_Address_A = fields.Char(string='Company Address', required=False)
81 Company_Phone_TNT_Number_A = fields.Char(string='Company Phone/TNT Number', required=False)
82 Duration_Joining_Date_Resign_date_A = fields.Date(string='Duration: Joining Date & Resign date', required=False)
83 Duration_Joining_Date_Resign_date_B = fields.Date(string='Duration: Joining Date & Resign date B', required=False)
84 CV_A = fields.Binary(string='CV', required=False)
85 Video_Resume_A = fields.Binary(string='Video Resume', required=False)
86
87
88 def action_send_email(self):
89     mail_template = self.env.ref('alumni_Management.Student_email_template')

```

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <odoo>
3     <record id="Register_tree" model="ir.ui.view">
4         <field name="name">alumni.register.tree</field>
5         <field name="model">alumni.register</field>
6         <field name="arch" type="xml">
7             <tree>
8                 <field name="Faculty"/>
9                 <field name="Department"/>
10                <field name="Semester"/>
11                <field name="Session"/>
12                <field name="Batch"/>
13                <field name="Board_Roll"/>
14                <field name="Registration_Number"/>
15                <field name="Students_Name"/>
16                <field name="Students_Email"/>
17                <field name="Date_of_Birth"/>
18                <field name="Fathers_Name"/>
19                <field name="Mothers_name"/>
20                <field name="Student_Phone_Number"/>
21                <field name="Fathers_Phone_Number"/>
22                <field name="Mothers_Phone_Number"/>
23                <field name="Local_Guardian_Phone_Number"/>
24                <field name="Result"/>

```

```

25 <field name="Present_Address"/>
26 <field name="Permanent_Address"/>
27 <field name="Interested_Areas"/>
28 <field name="Facebook"/>
29 <field name="LinkedIn"/>
30 <field name="Youtube_Channel"/>
31 <field name="Personal_Website"/>
32 <field name="Company_Name"/>
33 <field name="Company_Address"/>
34 <field name="Company_Responsible_person_contract_number"/>
35 <field name="Duration"/>
36 <field name="Company_Phone"/>
37 <field name="University_Name"/>
38 <field name="Subject_Name"/>
39 <field name="Industry_category"/>
40 <field name="Company_Industry_Name"/>
41 <field name="Designation"/>
42 <field name="Company_Address"/>
43 <field name="Company_Phone_TNT_Number"/>
44 <field name="Duration_Joining_Date_Resign_date"/>
45 <field name="Duration_Joining_Date_Resign_date_C"/>
46 <field name="CV"/>
47 <field name="Video_Resume"/>
48 <field name="Company_Industry_Name_A"/>

```

```

41 <field name="Designation"/>
42 <field name="Company_Address"/>
43 <field name="Company_Phone_TNT_Number"/>
44 <field name="Duration_Joining_Date_Resign_date"/>
45 <field name="Duration_Joining_Date_Resign_date_C"/>
46 <field name="CV"/>
47 <field name="Video_Resume"/>
48 <field name="Company_Industry_Name_A"/>
49 <field name="Designation_A"/>
50 <field name="Company_Address_A"/>
51 <field name="Company_Phone_TNT_Number_A"/>
52 <field name="Duration_Joining_Date_Resign_date_A"/>
53 <field name="Duration_Joining_Date_Resign_date_B"/>
54 <field name="CV_A"/>
55 <field name="Video_Resume_A"/>
56
57 </tree>
58 </field>
59 </record>
60
61 <record id="Register_form" model="ir.ui.view">
62 <field name="name">alumni.register.form</field>
63 <field name="model">alumni.register</field>
64

```

```

61
62 <record id="Register_form" model="ir.ui.view">
63   <field name="name">alumni.register.form</field>
64   <field name="model">alumni.register</field>
65   <field name="arch" type="xml">
66     <form>
67       <header>
68         <button name="action_send_email" string="Send By Email" type="object" class="oe_highlight"/>
69         <button name="approve_alumni" states="draft" string="Approve" type="object"
70           class="oe_highlight"/>
71       </header>
72       <field name="state" widget="statusbar" statusbar_visible="draft,approved"/>
73     </form>
74     <sheet>
75       <group>
76         <group>
77           <h2>Basic Information</h2>
78           <field name="Faculty"/>
79           <field name="Department"/>
80           <field name="Semester"/>
81           <field name="Session"/>
82           <field name="Batch"/>
83           <field name="Board_Roll"/>
84           <field name="Registration_Number"/>
85           <field name="Students_Name"/>

```

```

81   <field name="Batch"/>
82   <field name="Board_Roll"/>
83   <field name="Registration_Number"/>
84   <field name="Students_Name"/>
85   <field name="Students_Email"/>
86   <field name="Date_of_Birth"/>
87   <field name="Fathers_Name"/>
88   <field name="Mothers_name"/>
89 </group>
90 <group>
91   <br></br>
92   <br></br>
93   <field name="Student_Phone_Number"/>
94   <field name="Fathers_Phone_Number"/>
95   <field name="Mothers_Phone_Number"/>
96   <field name="Local_Guardian_Phone_Number"/>
97   <field name="Result"/>
98   <field name="Present_Address"/>
99   <field name="Permanent_Address"/>
100  <field name="Interested_Areas"/>
101 </group>
102 <group>
103   <h2>Other information (Social Media)</h2>

```

Figure 16: Front in Alumni registration

```

174 <record id="register" model="ir.actions.act_window">
175 <field name="name">Register</field>
176 <field name="type">ir.actions.act_window</field>
177 <field name="res_model">alumni.register</field>
178 <field name="view_mode">tree,kanban,form</field>
179 <field name="help" type="html">
180 <p class="o_view_content_smalling_face">
181 Alumni Students!
182 </p>
183 </field>
184 </record>
185 <menuitem id="Alumni_root"
186 name="Alumni"
187 sequence="10"/>
188 <menuitem id="Alumni_Register_root"
189 name="Register"
190 parent="Alumni_root"
191 sequence="10"/>
192 <menuitem id="Alumni_Register"
193 name="Register"
194 parent="Alumni_Register_root"
195 action="Partner.action"
196 </menuitem>
197 </odoo>

```

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <odoo>
3 <template id="tmp_customer_form" name="Partner Form">
4 <t t-name="website.registration-1">
5 <t t-call="website.layout">
6 <div id="wrap" class="oe_empty oe_structure">
7 <section class="o_colored_level s_title pb0 pt8" data-vcss="001" data-snippet="s_title"
8 data-name="Title" style="background-color: rgb(57, 132, 198) !important;">
9 <div class="s_allow_columns container">
10 <div class="row">
11 <div class="o_colored_level col-lq-12">
12 <h4 style="text-align: center;">
13 <font style="" class="text-180">Registration</font>
14 </h4>
15 </div>
16 </div>
17 </section>
18 <ul>
19 <section class="o_colored_level pb16 pt16 s_website_form undefined bg-288 o_half_screen_height"
20 data-vcss="001" data-snippet="s_website_form" data-name="Form" style="">
21 <div class="container">
22 <form action="/website_form/" method="post" enctype="multipart/form-data"
23 class="o_mark_required" data-mark="*" data-success-mode="nothing"
24 </form>

```

Figure 17: Admin Dashboard code

```

1  # -*- coding: utf-8 -*-
2  import ...
3
4
5  class PartnerForm(http.Controller):
6      @http.route(['/customer/form'], type='http', auth="public", website=True)
7      def partner_form(self, **post):
8          return request.render("alumni_Management.tmp_customer_form", {})
9
10     @http.route(['/customer/form/submit'], type='http', auth="public", website=True)
11     def customer_form_submit(self, **post):
12         partner = request.env['res.partner'].create({
13             'name': post.get('name'),
14             'email': post.get('email'),
15             'phone': post.get('phone')
16         })
17
18         vals = {
19             'partner': partner,
20         }
21         return request.render("alumni_Management.tmp_customer_form_success", vals)

```

```

4  class AlumniManagement(models.Model):
5      _name = "alumni.register"
6      _description = "Alumni Management"
7
8      STATES = [
9          ('draft', 'Draft'),
10         ('approved', 'Approved')
11     ]
12
13     # Basic Information
14     state = fields.Selection(selection=STATES, string='State', default='draft')
15     Faculty = fields.Char(string='Faculty ', required=False)
16     Department = fields.Char(string='Department', required=False)
17     Semester = fields.Char(string='Semester', required=False)
18     Session = fields.Char(string='Session', required=False)
19     Batch = fields.Char(string='Batch', required=False)
20     Board_Roll = fields.Char(string='Board Roll', required=False)
21     Registration_Number = fields.Char(string='Registration Number', required=False)
22     Students_Name = fields.Char(string='Students Name', required=True)
23     Students_Email = fields.Char(string='Students Email', required=True)
24     Date_of_Birth = fields.Date(string='Date of Birth', required=False)
25     Fathers_Name = fields.Char(string='Fathers Name', required=False)
26     Mothers_name = fields.Char(string='Mothers name', required=False)
27     Student_Phone_Number = fields.Char(string='Student Phone Number', required=False)
28     Fathers_Phone_Number = fields.Char(string='Fathers Phone Number', required=False)

```

Figure 18: Admin Dashboard py file

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <odoo>
3 <record id="Register_tree" model="ir.ui.view">
4 <field name="name">alumni.register.tree</field>
5 <field name="model">alumni.register</field>
6 <field name="arch" type="xml">
7 <tree>
8 <field name="Faculty"/>
9 <field name="Department"/>
10 <field name="Semester"/>
11 <field name="Session"/>
12 <field name="Batch"/>
13 <field name="Board_Roll"/>
14 <field name="Registration_Number"/>
15 <field name="Students_Name"/>
16 <field name="Students_Email"/>
17 <field name="Date_of_Birth"/>
18 <field name="Fathers_Name"/>
19 <field name="Mothers_name"/>
20 <field name="Student_Phone_Number"/>
21 <field name="Fathers_Phone_Number"/>
22 <field name="Mothers_Phone_Number"/>
23 <field name="Local_Guardian_Phone_Number"/>
24 <field name="Result"/>

```

Figure 19: Event creates code

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <odoo>
3 <record id="event_tree" model="ir.ui.view">
4 <field name="name">alumni.event.tree</field>
5 <field name="model">alumni.event</field>
6 <field name="arch" type="xml">
7 <tree>
8 <field name="Event_Title"/>
9 <field name="Location"/>
10 <field name="Phone_Number"/>
11 <field name="start_date"/>
12 <field name="end_date"/>
13 <field name="description"/>
14 </tree>
15 </field>
16 </record>
17
18 <record id="event_form" model="ir.ui.view">
19 <field name="name">alumni.event.form</field>
20 <field name="model">alumni.event</field>
21 <field name="arch" type="xml">
22 <form>
23 <header>
24 <button name="approve_alumni" states="draft" string="Approve" type="object"

```

```

1 from odoo import api, fields, models
2 from odoo.exceptions import ValidationError
3
4
5 class AlumniManagement(models.Model):
6     _name = "alumni.event"
7     _description = "Alumni Management"
8
9     STATES = [
10         ('draft', 'Draft'),
11         ('approved', 'Approved')
12     ]
13
14 state = fields.Selection(selection=STATES, string='State', default='draft')
15 Event_Title = fields.Char(string='Event Title ', required=False)
16 Location = fields.Char(string='Location', required=False)
17 Phone_Number = fields.Char(string='Phone Number', required=False)
18 start_date = fields.Datetime(string='Start Date', required=False)
19 end_date = fields.Datetime(string='End Date', required=False)
20 description = fields.Text(string='Description', required=True)
21
22 def approve_alumni(self):
23     self.state = 'approved'
24

```

Figure 20: Event creates

Access of this user:

```

1 id,name,model_id:id,group_id:id,perm_read,perm_write,perm_create,perm_unlink
2 access_alumni_register,access.alumni.register,model_alumni_register,,1,1,1,1
3 access_alumni_event,access.alumni.event,model_alumni_event,,1,1,1,1
4 access_alumni_donation,access.alumni.donation,model_alumni_donation,,1,1,1,1

```

Figure 21: Access of this user

10.2 Possible problem break down

The entire project must be divided down into discrete jobs to make system development easier and more effective. The following are some possible breakdowns for the proposed project:

- Database design and analysis
- Development of an admin panel
- Select and organize the System template.
- Create a system supplier panel.
- Create a customer panel for the System.

Design database:

- Determine the system's needs.
- Finish the normalizing procedure.
- Create a diagram of entity relationships.
- Design the database schema for the system.

Develop admin panel:

- Create the necessary pages.
- Develop a method for admin manipulation.
- Create all resource management pages necessary for the vehicle Alumni Management system.

Alumni panel development:

- Front-end design.
- Login and Registration system.
- Manipulation is requested by alumni.

Chapter 11 – Testing

11.1 Test Case

A test case is a collection of activities performed to verify the functionality of a programming application's specific element. A test case is a collection of test procedures, test information, preconditions, and post conditions developed for a specified test circumstance in order to confirm any requirement. Following the completion of the test acceptance plan, test cases must be developed (searchsoftwarequality, 2011). The following are the test cases for the Alumni Management System.

Unit test – test case:

Test Case Name	Unit Test		
Unit Test			
Test Description			
Data Source	Test Steps	Expected Result	Actual Result

Module Test – test case:

Test Case Name	Module Test		
Unit Test			
Test Description			
Data Source	Test Steps	Expected Result	Actual Result

Integration Testing – test case:

Test Case Name	Integration Test		
Unit Test			
Test Description			
Data Source	Test Steps	Expected Result	Actual Result

11.2 Unit test – test case:**Unit Test 1**

Test Case Name	Unit Test		
Unit Test	Alumni Register Controller		
Test Description	Alumni name validation for alumni registration		
Data Source	Test Steps	Expected Result	Actual Result
User Entry	<ol style="list-style-type: none"> Except for the alumni faculty field, all essential fields have been completed. Submit the form 	The alumni faculty must not be empty, according to an error notice.	A notification appears stating that the donor's name is necessary in order to register.

Table 7: Unit Test 1

The screenshot shows a web application interface for an alumni registration form. At the top, there is a navigation menu with links for Home, Registration, Events, Donation, Contact us, and Application. The user is logged in as 'Mitchell Admin'. The main heading is 'Registration'. Below this is a 'Basic Information' section with the following fields:

- Faculty * (empty, with a red 'x' error icon)
- Semester * (12)
- Batch * (21)
- Registration Number * (23452)
- Department * (CIS)
- Session * (summer2020)
- Board Roll * (751147)
- Students Name * (Anisuzzaman)

Below the basic information, there are two sections for file uploads:

- CV * (Choose File Capture.PNG)
- Video Resume * (Choose File Capture.PNG)

At the bottom, there is a green 'Submit' button and a red error message: 'Please fill in the form correctly.'

Figure 22: Alumni Register

Unit Test 2

Test Case Name	Unit Test		
Unit Test	Alumni Register Controller		
Test Description	Test user registration		
Data Source	Test Steps	Expected Result	Actual Result
User Entry	<ol style="list-style-type: none"> 1. Student signup testing 2. Submit the form 	If the customer enters his or her information and then presses the "Signup" button, the message "Success!" appears. Registration was completed successfully.	Expected Output

Table 8: Unit Test 2

Duration: Joining Date & Resign date *

CV * No file chosen No file chosen

Video Resume * No file chosen No file chosen

✔ The form has been sent successfully.

Figure 23: Alumni Register successfully

Unit Test 3

Test Case Name	Unit Test		
Unit Test	Admin		
Test Description	Validation of the admin list is still pending.		
Data Source	Test Steps	Expected Result	Actual Result
User Entry	<ol style="list-style-type: none"> 1. Log in as an administrator. 2. Go to the Alumni Requests page to see what's new. 	Alumni requests that are still pending should be shown.	Expected Output

Table 9: Unit Test 3

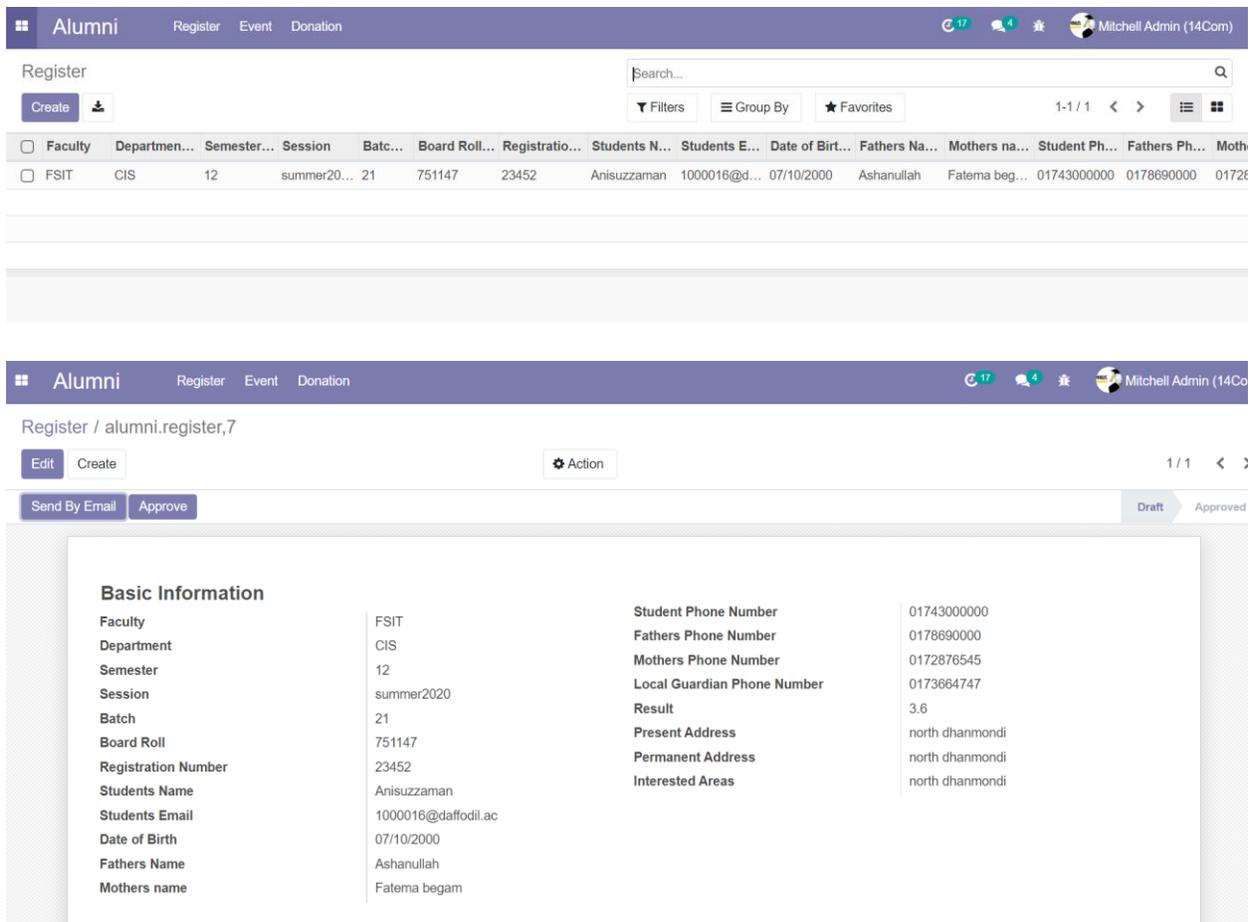


Figure 24: Admin panel

11.3 Module Test – test case:

Module Test 1

Test Case Name	Module Test		
Unit Test	User registration		
Test Description	Customer signup validation checking		
Data Source	Test Steps	Expected Result	Actual Result

Student registration for alumni	Validation checks, such as pressing the registration button without entering the customer's information.	If a consumer clicks the registration button without filling out the required information, the following notice appears.	Expected Output
---------------------------------	--	--	-----------------

Table 10: Module Test 1

The screenshot shows a web application interface for user registration. At the top, there is a navigation bar with links for Home, Registration, Events, Donation, Contact us, and Application. The user is logged in as 'Mitchell Admin'. The main content area is titled 'Registration' and contains a 'Basic Information' section. This section has a grid of input fields for various details. Each field is currently empty and has a red 'x' icon next to it, signifying that the form is not valid due to missing or incorrect information. The fields are: Faculty, Semester, Batch, Registration Number, Date of Birth (with a calendar icon), Fathers Name, Student Phone Number, Department, Session, Board Roll, Students Name, Students Email, Mothers name, and Fathers Phone Number.

Figure 25: User registration

Module Test 2

Test Case Name	Module Test		
Unit Test	Admin registration		
Test Description	Admin signup validation checking		
Data Source	Test Steps	Expected Result	Actual Result
Student registration for alumni	Validation checks, such as pressing the registration button without entering the customer's information.	If a consumer clicks the registration button without filling out the required information, the following notice appears.	Expected Output

Table 11: Module Test 2

The screenshot shows the 'Alumni Register / New' form. The form is divided into two columns of input fields. The left column includes fields for Faculty, Department, Semester, Session, Batch, Board Roll, Registration Number, Students Name, Students Email, Date of Birth, and Fathers Name. The right column includes fields for Student Phone Number, Fathers Phone Number, Mothers Phone Number, Local Guardian Phone Number, Result, Present Address, Permanent Address, and Interested Areas. A red error message box in the top right corner states 'Invalid fields:' and lists 'Students Name' and 'Students Email' as the problematic fields. The 'Students Name' and 'Students Email' fields in the form are highlighted in red, indicating they are the source of the error.

Figure 26: Admin registration

11.4 Integration Testing – test case:

Test Case Name	Integration Test		
Unit Test	Data source		
Test Description	Alumni create Event		
Data Source	Test Steps	Expected Result	Actual Result
Create event to the system	Checking event create and Manage event to the admin page	If an admin clicks the login button and then the manage event button, the website displays events to the admin.	Expected Output

Table 12: Integration Testing – test case

Chapter 12 – Implementation

12.1 Training

This is a brand-new system for employees to use in the real world, and certain functionalities may be unfamiliar to them. In this instance, training will be required on the premises of the company so that employees may feel at ease while learning from the teacher. In addition, an instruction manual will be given with all relevant information.

12.2 Big Bang

Big bang works by turning off the old system and immediately turning on the new one. This technique is substantially faster than the others since it employs the new system immediately upon the completion of the test. As a result of the data transfer, data from the old system may be damaged or deleted, and the new system may crash. It's based on a single website and employs a single group of human resources.

Chapter 13 – Critical Appraisal and Evaluation

13.1 Success rate against each objective

The previously stated objectives have been realized in this web platform. First and foremost, the online system will have a user-friendly interface to allow for optimum efficiency. The odoo14 design framework was used to implement this concept. The design's second feature is the navigation system. This is required for thorough scanning of the sites, as users might become trapped when exploring many pages. The next step is to develop a comprehensive product viewing, event, and payment system. This is required, and the success rate is quite high.

13.2 How much better could have been done

In comparison to other forms of online Alumni registration and oil events, the web system that has been established is more superior. However, there was room for improvement. Some features were explored during development, but because they were put into the "could have" category, they were transferred into subsequent development.

13.3 How better is the features of the solution

- i. **Alumni Manage:** System will be doing atomically graduate every student as alumni.
- ii. **Payment system:** Every undergraduate student must give payment for be alumni. Even alumni have to register with money if they want to attend an event.
- iii. **Manage Poor Fund and blood donation:** University have managed a poor fund for poor brilliant students where alumni help poor brilliant students. Alumni will be collect blood for peoples because this work is great work as a human being.

13.4 Which features could not be touched

The voice system, computer vision system features could not be touched. These will be introduced in the future.

Chapter 14 – Conclusion

14.1 Summary of the project

The project's summary is that this web system will assist both stakeholders and end-users in online alumni administration since it incorporates all contemporary capabilities and features. Stakeholders may have faith in the project since the functionalities produced are equivalent to a worldwide standard online system, and consumers can easily utilize the payment system.

14.2 Goal of the project

The project's purpose was to create an efficient and dependable system that could be accessed from anywhere. The project's aim has been achieved.

14.3 My experience

My project development experience was positive since the web system presented additional obstacles. I'll need to seek assistance from a variety of sources and maintain constant contact with stakeholders. Furthermore, the framework proved beneficial, and new information was gained as a result of it.

Works Cited

- www.bracu.ac.bd*. (1995, 3 25). Retrieved from bracu: <https://www.bracu.ac.bd/>
- ukessays*. (2001, 11 26). Retrieved from ukessays: <https://www.ukessays.com/essays/information-systems/dynamic-systems-development-methodology.php>
- link*. (2010, 3 26). Retrieved from link.springer.com: <https://link.springer.com/article/10.1007/s13132-020-00643-3>
- inettutor*. (2011, 8 24). Retrieved from www.inettutor.com: <https://www.inettutor.com/source-code/alumni-information-system-review-of-related-literature/>
- searchsoftwarequality*. (2011, 7 21). Retrieved from searchsoftwarequality: <https://searchsoftwarequality.techtarget.com/definition/test-case>
- simplilearn*. (2013, 2 12). Retrieved from simplilearn: <https://www.simplilearn.com/feasibility-study-article>
- intelschool*. (2014, 12 25). Retrieved from www.intelschool.in: <https://www.intelschool.in/alumni-management-system/#:~:text=An%20alumni%20management%20system%20is,institution%20as%20future%20brand%20ambassadors.>
- libguides*. (2014, 5 14). Retrieved from libguides: <https://libguides.usc.edu/writingguide/methodology>
- askanydifference*. (2015, 3 18). Retrieved from askanydifference: <https://askanydifference.com/difference-between-client-server-application-and-web-application/>
- omnisci*. (2015, 6 23). Retrieved from omnisci: <https://www.omnisci.com/technical-glossary/client-server>
- studentprojectguide*. (2015, 4 16). Retrieved from <https://studentprojectguide.com>: <https://studentprojectguide.com/php/alumni-management-system/>
- tutorialspoint*. (2015, 1 18). Retrieved from tutorialspoint: https://www.tutorialspoint.com/sdlc/sdlc_waterfall_model.htm
- digite*. (2016, 4 23). Retrieved from digite: <https://www.digite.com/agile/dynamic-system-development-method-dsdm/>
- www.researchgate.net*. (2017, 3 27). Retrieved from researchgate: https://www.researchgate.net/publication/348431044_A_Model_and_Framework_for_Alumni_Tracking_and_Information_System
- ijert*. (2018, 9 30). Retrieved from www.ijert.org: <https://www.ijert.org/alumni-database-management-system>

technodg. (2019, 6 11). Retrieved from www.technodg.com: <https://www.technodg.com/alumni-management-system.html>

academia. (2020, 3 23). Retrieved from www.academia.edu :
https://www.academia.edu/36219135/Chapter_1_THE_PROBLEM_AND_ITS_BACKGROUND

sparklyn. (2021, 2 31). Retrieved from sparklyn.com.ng: <https://sparklyn.com.ng/design-and-implementation-of-a-webbased-alumni-database-management-system-5624-d>

studentprojects. (2021, 1 21). Retrieved from www.studentprojects.live:
<https://www.studentprojects.live/phpscript/alumni-management-system/>

sciencedirect. (2026, 4 16). Retrieved from <https://www.sciencedirect.com>:
<https://www.sciencedirect.com/science/article/pii/S1877042815030761/pdf?md5=d5bb2a49cf24b38d107778f69a415814&pid=1-s2.0-S1877042815030761-main.pdf>

Appendices

User Guide:

Using for Students

The image shows two screenshots of the ALUMNIS website. The top screenshot displays the registration form with the following fields:

- Your Email:** An empty text input field.
- Your Name:** A text input field with the placeholder text "e.g. John Doe".
- Password:** An empty text input field.
- Confirm Password:** An empty text input field.
- Sign up:** A teal button.
- Already have an account?:** A link below the sign up button.

The bottom screenshot shows the user dashboard after a successful login. The navigation menu includes: Home, Registration, Events, Fund Collection, and Contact us. The main content area is divided into three sections:

- Registration:** A link to "Registration".
- Event:** A link to "Create Event".
- Upcoming Events:** A table listing events:

Event Name	Date
Fresher's Orientation - Fall 2021	Nov 24-26
International Mother Language Day	Feb 20-21

A "Save password?" dialog box is overlaid on the right side of the dashboard, showing the username "anisuzzaman1@gmail.com" and a masked password field. The dialog has "Save" and "Never" buttons.

Figure: Student signup and login successfully to the system

ALUMNIS  Home Registration Events Fund Collection Contact us Anis1 Contact Us

Registration

Basic Information

Faculty *	<input type="text"/>	Department *	<input type="text"/>
Semester *	<input type="text"/>	Session *	<input type="text"/>
Batch *	<input type="text"/>	Board Roll *	<input type="text"/>
Registration Number *	<input type="text"/>	Students Name *	<input type="text"/>
Date of Birth *	<input type="text"/> 	Students Email *	<input type="text"/>
Fathers Name *	<input type="text"/>	Mothers name *	<input type="text"/>
Student Phone Number *	<input type="text"/>	Fathers Phone Number *	<input type="text"/>

Duration: Joining Date & Resign date *	<input type="text"/> 	<input type="text"/> 	Duration: Joining Date & Resign date *	<input type="text"/> 	<input type="text"/> 
CV *	<input type="button" value="Choose File"/> No file chosen		CV *	<input type="button" value="Choose File"/> No file chosen	
Video Resume *	<input type="button" value="Choose File"/> No file chosen		Video Resume *	<input type="button" value="Choose File"/> No file chosen	

✓ The form has been sent successfully.

Figure: Alumni Registration successfully

[Home](#)
[Registration](#)
[Events](#)
[Fund Collection](#)
[Contact us](#)
Anis1
[Contact Us](#)

Create Event

Event Title *

Location * **Phone Number ***

Start Date * **End Date ***

Description

[Home](#)
[Registration](#)
[Events](#)
[Fund Collection](#)
[Contact us](#)
Anis1
[Contact Us](#)



Thank You!

 Your message has been sent **successfully**
 We will get back to you shortly.

OpenEduCat
 Dhanmondi 32
 Dhaka CA 1205
 Bangladesh
 01772875121
 1000016@daffodil.ac
[Google Maps](#)

Figure: Event Create successfully

Events

Upcoming Events All countries Search an event...

NOV 24

Fresher's Orientation - Fall 2021
November 24, 2021 - 11:00 AM
Dhaka, Bangladesh
Music

FEB 21

International Mother Language ...
February 21, 2022 - 12:00 AM
Dhaka, Bangladesh
Music
Sold Out

ABOUT US

Use this paragrah to write a short text about your events or company.

FOLLOW US

f t in y r i

PHOTOS

Figure: View this events

Home Registration Events Fund Collection Contact us

Fund Collection

Select Operator * Bkash Rocket Nagad

Amount *

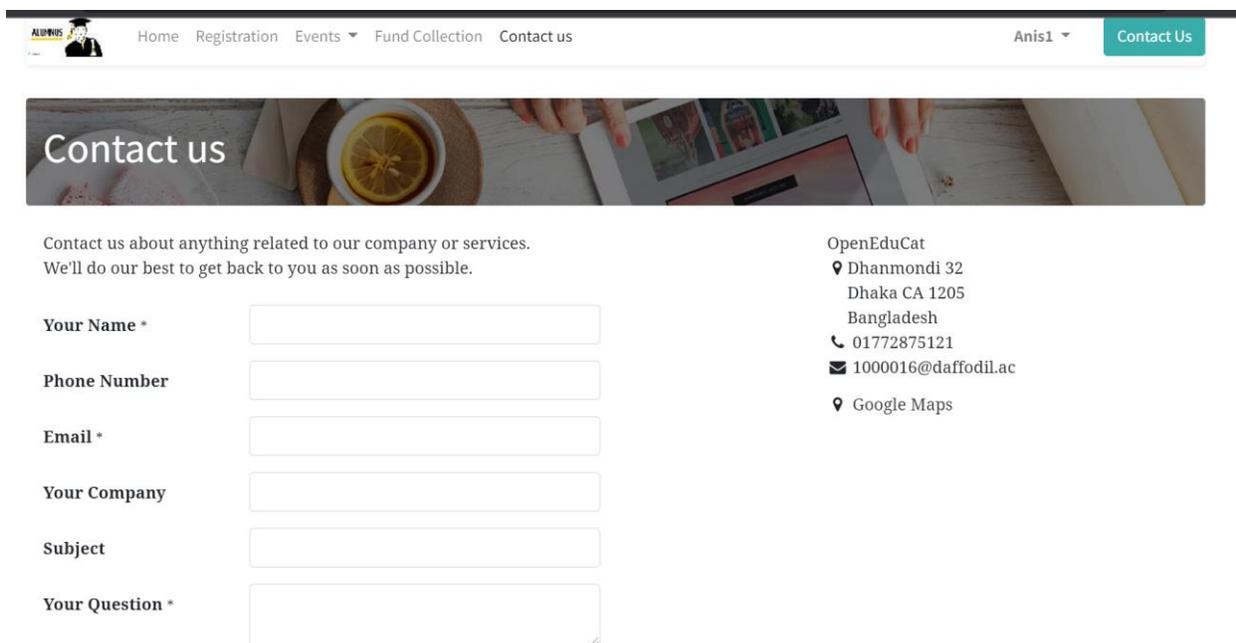
Account Number *

Account Name *

Enter Your Pin *

Submit

Figure: Fund collection



ALUMNI [Home](#) [Registration](#) [Events](#) [Fund Collection](#) [Contact us](#) Anis1 [Contact Us](#)

Contact us

Contact us about anything related to our company or services. We'll do our best to get back to you as soon as possible.

Your Name *

Phone Number

Email *

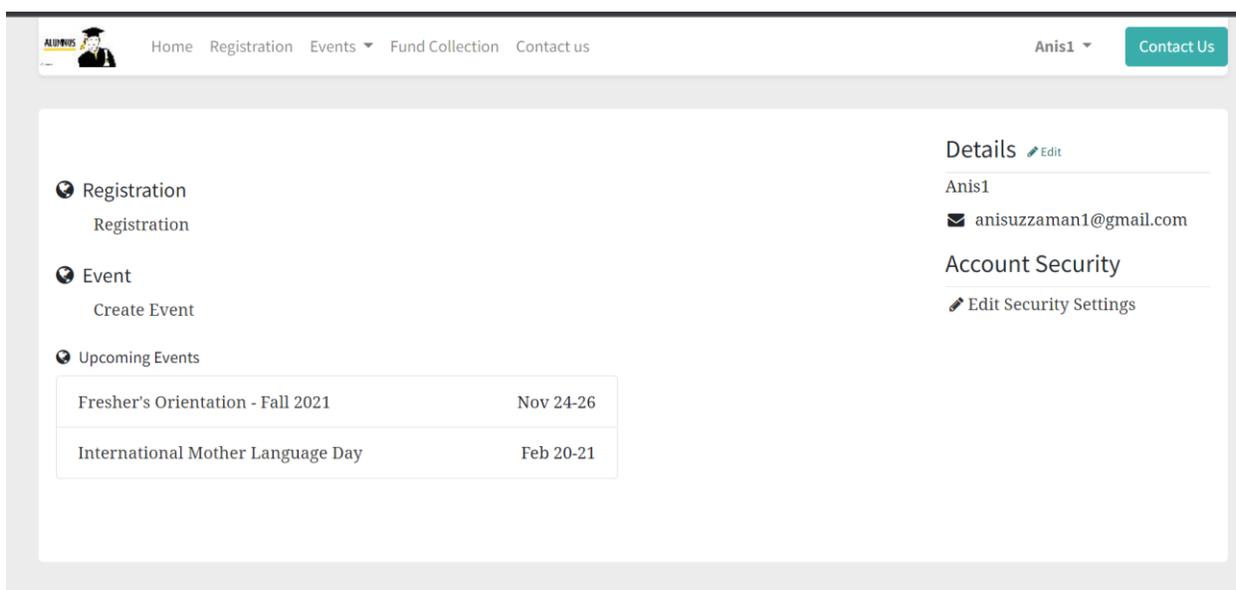
Your Company

Subject

Your Question *

OpenEduCat
 Dhanmondi 32
 Dhaka CA 1205
 Bangladesh
 01772875121
 1000016@daffodil.ac
 Google Maps

Figure: Alumni Contacts Page



ALUMNI [Home](#) [Registration](#) [Events](#) [Fund Collection](#) [Contact us](#) Anis1 [Contact Us](#)

Registration
Registration

Event
Create Event

Upcoming Events

Fresher's Orientation - Fall 2021	Nov 24-26
International Mother Language Day	Feb 20-21

Details [Edit](#)

Anis1
 ✉ anisuzzaman1@gmail.com

Account Security
 ✎ Edit Security Settings

Figure: Alumni Profile

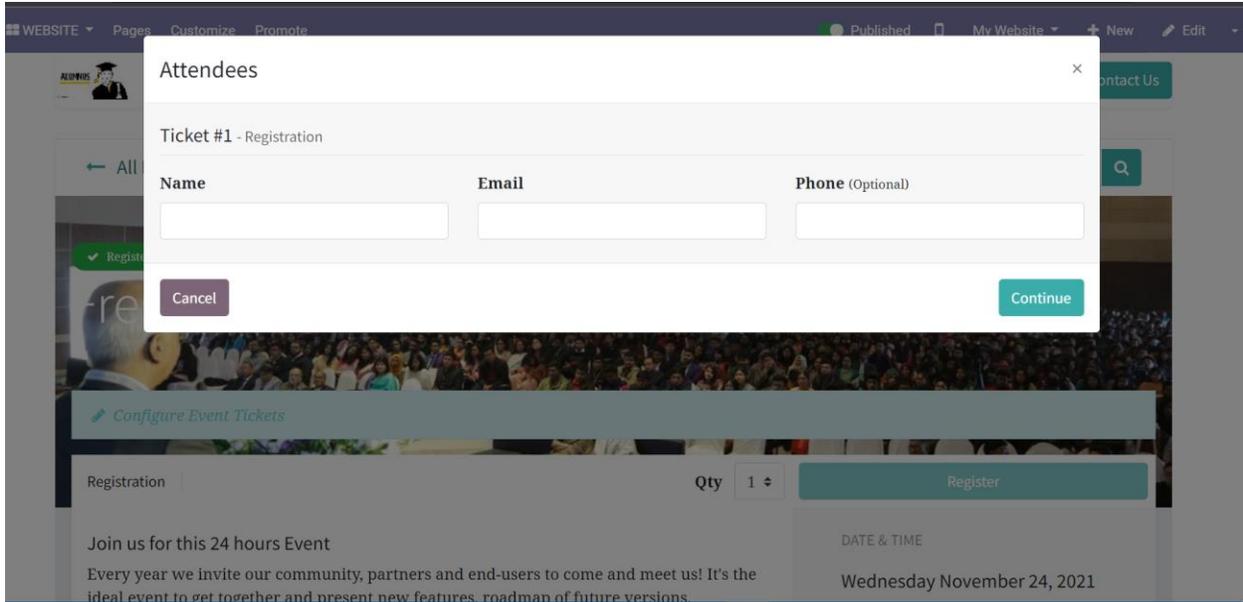


Figure: Event Registration from

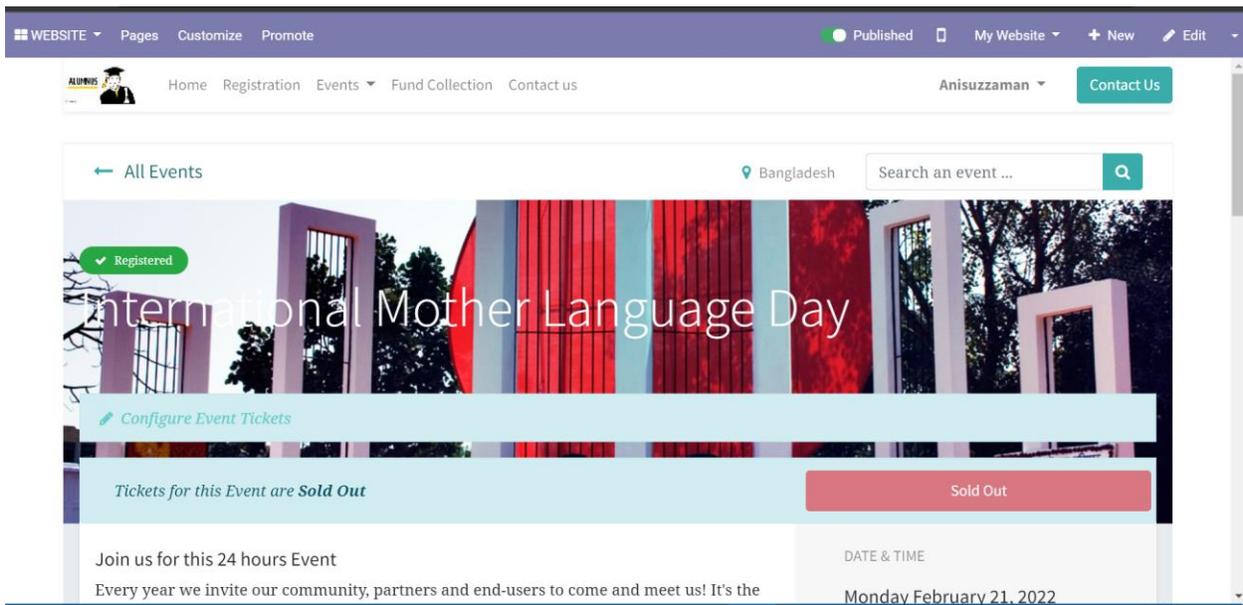


Figure: Event Sold Out

The screenshot shows a website interface for 'Events'. At the top, there are navigation links: WEBSITE, Pages, Customize, Promote, My Website, + New, and Edit. Below this is a search bar with 'Search an event...' and a magnifying glass icon. There are also filters for 'Past Events' and 'All countries'. The main content area displays two event cards:

- Peace Day 2021**: November 12, 2021 - 6:00 AM, Dhaka, Bangladesh. Category: Culture. Status: Registrations Closed.
- Bangoli New Year - 1428**: April 13, 2021 - 7:00 PM, Dhaka, Bangladesh. Category: Music. Status: Registrations Closed.

On the right side, there is an 'ABOUT US' section with a placeholder text: 'Use this paragrah to write a short text about your events or company.' Below it is a 'FOLLOW US' section with social media icons for Facebook, Twitter, LinkedIn, YouTube, and Instagram. At the bottom right, there is a 'PHOTOS' section with a placeholder image.

Figure: Pasts Events

Using for Admin

The screenshot shows an admin login page. At the top, there is a navigation bar with links: Home, Registration, Events, Fund Collection, Contact us, Sign in, and Contact Us. The main content area contains a login form:

- Email**: Input field containing 'admin'.
- Password**: Input field containing '.....'.
- Log in**: A teal button.
- Below the button are links for 'Don't have an account?' and 'Reset Password'.

At the bottom, there is a dark footer section with three columns:

- Useful Links**: Home
- About us**: Manage and update your university alumni lists every
- Connect with us**: Contact us

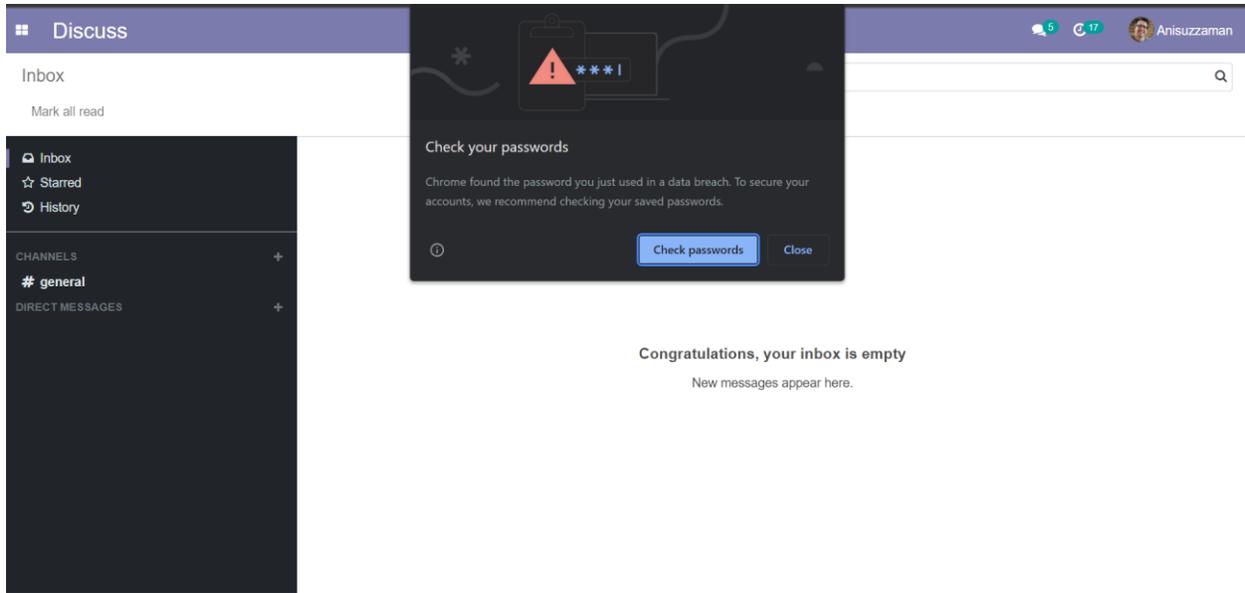


Figure: Admin Login successfully

The screenshot shows an 'Alumni' dashboard with a top navigation bar containing 'Register', 'Event', and 'Fund Collection'. Below the navigation is a 'Register' section with a search bar and a table. The table has columns for Faculty, Department, Semester, Session, Batch, Board Roll, Registration, Students N, Students E, Date of Birth, Fathers Name, Mothers Name, Student Phone, Fathers Phone, and Mother. The table contains two rows of data.

<input type="checkbox"/>	Faculty	Departmen...	Semester...	Sessio...	Batch	Board Roll...	Registratio...	Students N...	Students E...	Date of Birt...	Fathers Na...	Mothers na...	Student Ph...	Fathers Ph...	Mother
<input type="checkbox"/>	FSIT	CIS	Fall 2021	2018-2022	CIS1	102100	213421	MD. Anisuz...	1000016@d...	06/08/1994	MD.Ashan ...	Mst. Fatem...	01772875121	01738474746	019383
<input type="checkbox"/>	FSIT	CIS	Fall 2021	2018-2021	cis4	12321	23432	Gasul hok	1000016@d...	08/11/1992	Rohim uddin	Khadija Beg...	01772875121		

Figure: Admin Dashboard

Figure: Admin add new Alumni

Faculty	Departmen...	Semester...	Sessio...	Batch	Board Roll...	Registratio...	Students N...	Students E...	Date of Birt...	Fathers Na...	Mothers na...	Student Ph...	Fathers Ph...	Mother	
<input type="checkbox"/>	FSIT	CIS	Fall 2021	2018-2022	CIS1	102100	213421	MD. Anisuz...	1000016@d...	06/08/1994	MD.Ashan ...	Mst. Fatem...	01772875121	01738474746	019383
<input type="checkbox"/>	FSIT	CIS	Fall 2021	2018-2021	dis4	12321	23432	Gasul hok	1000016@d...	08/11/1992	Rohim uddin	Khadija Beg...	01772875121		

Figure: Admin view Alumni members

Event Title	Location	Phone Number...	Start Date	End Date	Description	
<input type="checkbox"/>	Victory day	Mirpur	0172373633	11/29/2021 12:56:00	11/25/2021 12:56:00	Victory day is national holiday in Bangladesh celebrated on 16 December to commemorate the victory of the Bangladesh forces over the Pakistani forces in the
<input type="checkbox"/>	Independence Day of Bangladesh 20...	Dhaka	0187736364	11/23/2021 13:35:59	11/22/2021 13:35:59	Independence Day The independence of Bangladesh was declared on 26 March 1971 following the crackdown by the Pakistani army on the night of 25 ...
<input type="checkbox"/>	sdd	sda	sadsa	10/06/2021 10:00:00	10/12/2021 12:40:22	sasa
<input type="checkbox"/>	asd	dss	sds	11/19/2021 11:07:48	12/04/2021 11:07:50	dsd

Figure: Admin view Events

Select Operator	Amount	Account Name	Account Number	Enter Your Pin
<input type="checkbox"/> Bkash	230	Anisuzzaman	01772875121	12345
<input type="checkbox"/> Rocket	234	0177366354	Anis	2134

Figure: Admin view Fund Collection

New	Booked	Announced	Ended
<p>24 Nov 2021 11:00 AM → 26 Nov</p> <p>Fresher's Orientation - Fall 2021 OpenEduCat 3 Expected attendees ↳ 1 Confirmed</p>	<p>21 Feb 2022 12:00 AM</p> <p>International Mother Language ... OpenEduCat 1 Expected attendees ↳ 1 Confirmed</p>	<p>14 Apr 2021 8:00 AM</p> <p>Bangoli New Year - 1428 OpenEduCat 0 Expected attendees</p>	
<p>12 Nov 2021 11:00 AM → 13 Nov</p> <p>Peace Day 2021 OpenEduCat 0 Expected attendees</p>			

Figure: Admin view All Events

Events / New

Save Discard

Invite Preview Badges

New Booked Announced Ended Cancelled

Attendees: 0 Go to Website

Event Name: e.g. Conference for Architects

Date: [] → []

Timezone: Asia/Dhaka

Template: []

Tags: []

Organizer: OpenEduCat

Responsible: Anisuzzaman

Website: []

Venue: OpenEduCat

Limit Registrations:

Autoconfirmation:

Tickets Communication Notes

Name	Product	Price	Sales Start	Sales End	Maximum	Confirmed	Unconfirmed
------	---------	-------	-------------	-----------	---------	-----------	-------------

Figure: Admin Create Events

Events Analysis

Search...

Measures [] [] [] []

Filters [] Group By [] Favorites []

	+ Total
	Reserved Seats
- Total	0
+ Bangoli New Year - 1428	0
+ Fresher's Orientation - Fall 2021	0
+ International Mother Language Day	0
+ Peace Day 2021	0

Figure: Admin view total events

Event Templates

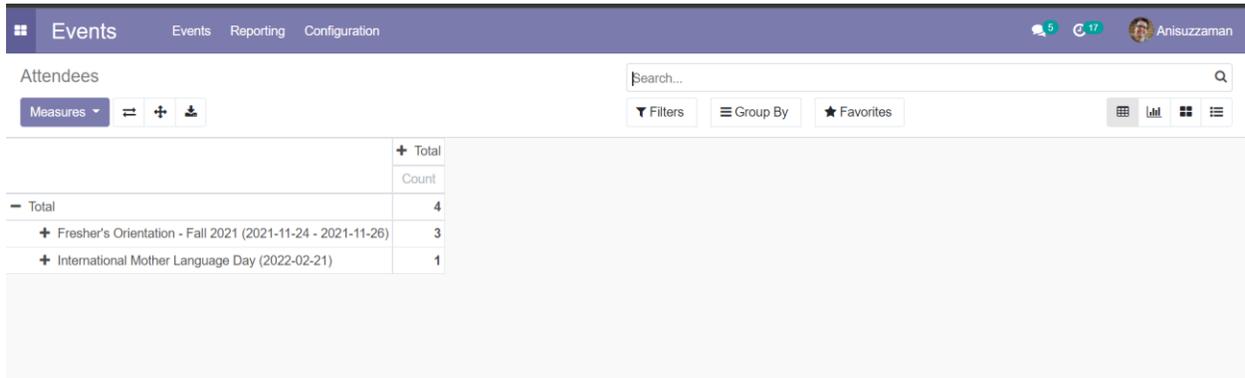
Search...

Create []

Filters [] Group By [] Favorites [] 1-5 / 5 < >

Event Template
<input type="checkbox"/> + Ticketing
<input type="checkbox"/> + Conference
<input type="checkbox"/> + Exhibition
<input type="checkbox"/> + Training
<input type="checkbox"/> + Sport

Figure: Event registration category



Attendees

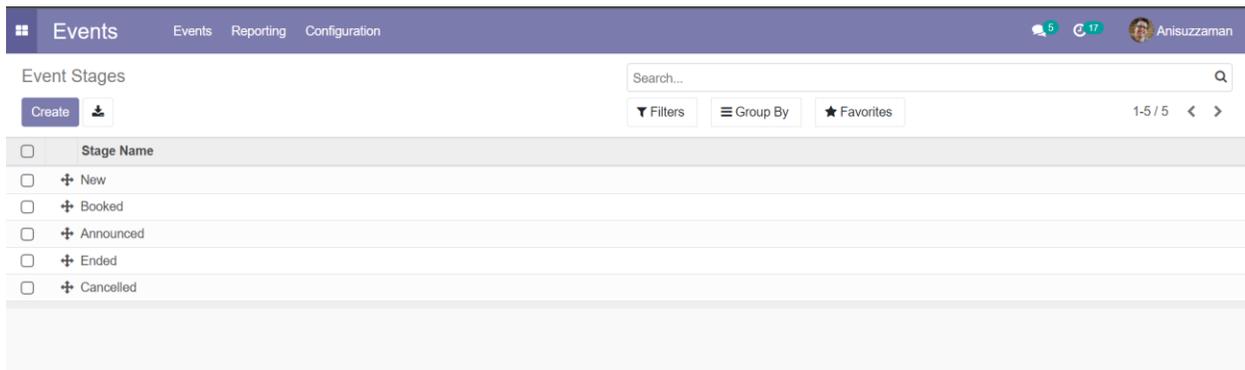
Search...

Measures ▾

Filters Group By Favorites

	+ Total
	Count
- Total	4
+ Fresher's Orientation - Fall 2021 (2021-11-24 - 2021-11-26)	3
+ International Mother Language Day (2022-02-21)	1

Figure: Admin view all event registration person



Event Stages

Search...

Create

Filters Group By Favorites

1-5 / 5 < >

<input type="checkbox"/>	Stage Name
<input type="checkbox"/>	+ New
<input type="checkbox"/>	+ Booked
<input type="checkbox"/>	+ Announced
<input type="checkbox"/>	+ Ended
<input type="checkbox"/>	+ Cancelled

Figure: Event Stages

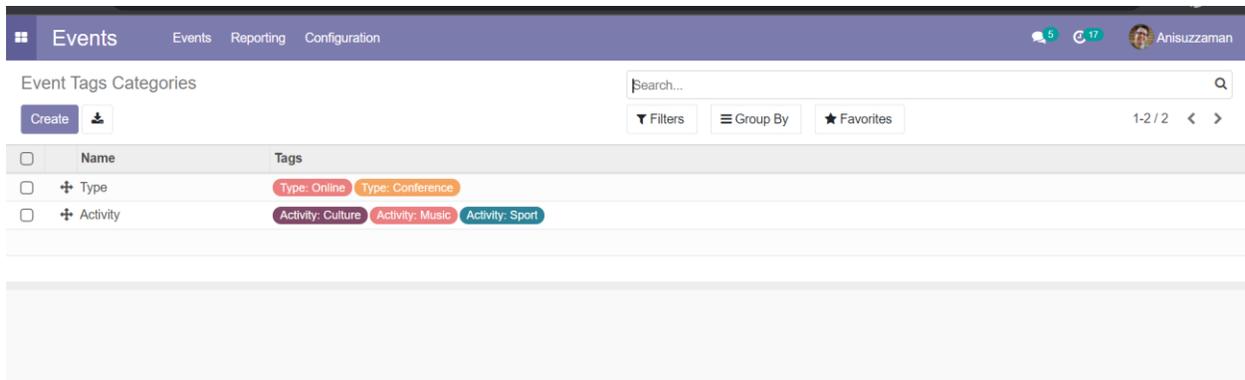


Figure: Event Tags Categories.

System Code-

Registration sample for the users-

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <odoo>
3   <data>
4     <record id="menu_partner_form" model="website.menu">
5       <field name="name">Registration</field>
6       <field name="url">/customer/form</field>
7       <field name="parent_id" ref="website.main_menu"/>
8       <field name="sequence" type="int">22</field>
9     </record>
10  </data>
11 </odoo>

```

```

1 from odoo import api, fields, models
2
3
4 class AlumniManagement(models.Model):
5     _name = "alumni.register"
6     _description = "Alumni Management"
7
8     STATES = [
9         ('draft', 'Draft'),
10        ('approved', 'Approved')
11    ]
12    # Basic Information
13    state = fields.Selection(selection=STATES, string='State', default='draft')
14    Faculty = fields.Char(string='Faculty ', required=False)
15    Department = fields.Char(string='Department', required=False)
16    Semester = fields.Char(string='Semester', required=False)
17    Session = fields.Char(string='Session', required=False)
18    Batch = fields.Char(string='Batch', required=False)
19    Board_Roll = fields.Char(string='Board Roll', required=False)
20    Registration_Number = fields.Char(string='Registration Number', required=False)
21    Students_Name = fields.Char(string='Students Name', required=True)
22    Students_Email = fields.Char(string='Students Email', required=True)
23    Date_of_Birth = fields.Date(string='Date of Birth', required=False)
24    Fathers_Name = fields.Char(string='Fathers Name', required=False)

```

Figure: alumni registration

```

25 Mothers_name = fields.Char(string='Mothers name', required=False)
26 Student_Phone_Number = fields.Char(string='Student Phone Number', required=False)
27 Fathers_Phone_Number = fields.Char(string='Fathers Phone Number', required=False)
28 Mothers_Phone_Number = fields.Char(string='Mothers Phone Number', required=False)
29 Local_Guardian_Phone_Number = fields.Char(string='Local Guardian Phone Number ', required=False)
30 Result = fields.Char(string='Result', required=False)
31 Present_Address = fields.Char(string='Present Address', required=False)
32 Permanent_Address = fields.Char(string='Permanent Address', required=False)
33 Interested_Areas = fields.Char(string='Interested Areas', required=False)
34 # Other information (Social Media)
35 Facebook = fields.Char(string='Facebook ', required=False)
36 LinkedIn = fields.Char(string='LinkedIn', required=False)
37 Youtube_Channel = fields.Char(string='Youtube_Channel', required=False)
38 Personal_Website = fields.Char(string='Personal_Website', required=False)
39 # Internship data
40 Company_Name = fields.Char(string='Company Name ', required=False)
41 Company_Address = fields.Char(string='Company Address', required=False)
42 Company_Responsable_person_contract_number = fields.Char(string='Company Responsible person contract number',
43 required=False)
44 Duration = fields.Char(string='Duration', required=False)
45 Company_Phone = fields.Char(string='Company Phone', required=False)
46 # Higher Education
47 University_Name = fields.Char(string='University Name ', required=False)
48 Subject_Name = fields.Char(string='Subject_Name', required=False)

```

```

46 # Higher Education
47 University_Name = fields.Char(string='University Name ', required=False)
48 Subject_Name = fields.Char(string='Subject_Name', required=False)
49 # Industry category
50 Industry_category = fields.Selection([
51
52 ('Education/Training', 'Education/Training'),
53 ('Electrical/Electronics', 'Electrical/Electronics'),
54 ('Garments/Textile', 'Garments/Textile'),
55 ('Design/Creative', 'Design/Creative'),
56 ('Production/Operation', 'Production/Operation'),
57 ('Hospitality/ Travel/ Tourism', 'Hospitality/ Travel/ Tourism'),
58 ('IT/Telecommunication', 'IT/Telecommunication'),
59 ('Medical/Pharma', 'Medical/Pharma'),
60 ('Real Estate/ Development', 'Real Estate/ Development'),
61 ('Architecture', 'Architecture'),
62 ('Accounting/Finance', 'Accounting/Finance'),
63 ('Web Developer', 'Web Developer'),
64 ('Freelancer', 'Freelancer'),
65 ], required=False, default='Education/Training')
66 # Previous Employment
67 Company_Industry_Name = fields.Char(string='Company/Industry Name', required=False)
68 Designation = fields.Char(string='Designation', required=False)

```

Figure: Alumni registration

```

65 ], required=False, default='Education/Training')
66
67 # Previous Employment
68 Company_Industry_Name = fields.Char(string='Company/Industry Name', required=False)
69 Designation = fields.Char(string='Designation', required=False)
70 Company_Address_A = fields.Char(string='Company Address', required=False)
71 Company_Phone_TNT_Number = fields.Char(string='Company Phone/TNT Number', required=False)
72 Duration_Joining_Date_Resign_date = fields.Date(string='Duration: Joining Date & Resign date ', required=False)
73 Duration_Joining_Date_Resign_date_C = fields.Date(string='Duration: Joining Date & Resign date C ', required=False)
74 CV = fields.Binary(string='CV', required=False)
75 Video_Resume = fields.Binary(string='Video Resume', required=False)
76
77 # Current Employment
78 Company_Industry_Name_A = fields.Char(string='Company/Industry Name ', required=False)
79 Designation_A = fields.Char(string='Designation', required=False)
80 Company_Address_A = fields.Char(string='Company Address', required=False)
81 Company_Phone_TNT_Number_A = fields.Char(string='Company Phone/TNT Number', required=False)
82 Duration_Joining_Date_Resign_date_A = fields.Date(string='Duration: Joining Date & Resign date', required=False)
83 Duration_Joining_Date_Resign_date_B = fields.Date(string='Duration: Joining Date & Resign date B', required=False)
84 CV_A = fields.Binary(string='CV', required=False)
85 Video_Resume_A = fields.Binary(string='Video Resume', required=False)
86
87
88 def action_send_email(self):
89     mail_template = self.env.ref('alumni_Management.Student_email_template')

```

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <odoo>
3     <record id="Register_tree" model="ir.ui.view">
4         <field name="name">alumni.register.tree</field>
5         <field name="model">alumni.register</field>
6         <field name="arch" type="xml">
7             <tree>
8                 <field name="Faculty"/>
9                 <field name="Department"/>
10                <field name="Semester"/>
11                <field name="Session"/>
12                <field name="Batch"/>
13                <field name="Board_Roll"/>
14                <field name="Registration_Number"/>
15                <field name="Students_Name"/>
16                <field name="Students_Email"/>
17                <field name="Date_of_Birth"/>
18                <field name="Fathers_Name"/>
19                <field name="Mothers_name"/>
20                <field name="Student_Phone_Number"/>
21                <field name="Fathers_Phone_Number"/>
22                <field name="Mothers_Phone_Number"/>
23                <field name="Local_Guardian_Phone_Number"/>
24                <field name="Result"/>

```

```

25 <field name="Present_Address"/>
26 <field name="Permanent_Address"/>
27 <field name="Interested_Areas"/>
28 <field name="Facebook"/>
29 <field name="LinkedIn"/>
30 <field name="Youtube_Channel"/>
31 <field name="Personal_Website"/>
32 <field name="Company_Name"/>
33 <field name="Company_Address"/>
34 <field name="Company_Responsible_person_contract_number"/>
35 <field name="Duration"/>
36 <field name="Company_Phone"/>
37 <field name="University_Name"/>
38 <field name="Subject_Name"/>
39 <field name="Industry_category"/>
40 <field name="Company_Industry_Name"/>
41 <field name="Designation"/>
42 <field name="Company_Address"/>
43 <field name="Company_Phone_TNT_Number"/>
44 <field name="Duration_Joining_Date_Resign_date"/>
45 <field name="Duration_Joining_Date_Resign_date_C"/>
46 <field name="CV"/>
47 <field name="Video_Resume"/>
48 <field name="Company_Industry_Name_A"/>

```

```

41 <field name="Designation"/>
42 <field name="Company_Address"/>
43 <field name="Company_Phone_TNT_Number"/>
44 <field name="Duration_Joining_Date_Resign_date"/>
45 <field name="Duration_Joining_Date_Resign_date_C"/>
46 <field name="CV"/>
47 <field name="Video_Resume"/>
48 <field name="Company_Industry_Name_A"/>
49 <field name="Designation_A"/>
50 <field name="Company_Address_A"/>
51 <field name="Company_Phone_TNT_Number_A"/>
52 <field name="Duration_Joining_Date_Resign_date_A"/>
53 <field name="Duration_Joining_Date_Resign_date_B"/>
54 <field name="CV_A"/>
55 <field name="Video_Resume_A"/>
56
57 </tree>
58 </field>
59 </record>
60
61 <record id="Register_form" model="ir.ui.view">
62 <field name="name">alumni.register.form</field>
63 <field name="model">alumni.register</field>
64

```

```

61
62 <record id="Register_form" model="ir.ui.view">
63   <field name="name">alumni.register.form</field>
64   <field name="model">alumni.register</field>
65   <field name="arch" type="xml">
66     <form>
67       <header>
68         <button name="action_send_email" string="Send By Email" type="object" class="oe_highlight"/>
69         <button name="approve_alumni" states="draft" string="Approve" type="object"
70           class="oe_highlight"/>
71         <field name="state" widget="statusbar" statusbar_visible="draft,approved"/>
72       </header>
73       <sheet>
74         <group>
75           <group>
76             <h2>Basic Information</h2>
77             <field name="Faculty"/>
78             <field name="Department"/>
79             <field name="Semester"/>
80             <field name="Session"/>
81             <field name="Batch"/>
82             <field name="Board_Roll"/>
83             <field name="Registration_Number"/>
84             <field name="Students_Name"/>

```

```

81   <field name="Batch"/>
82   <field name="Board_Roll"/>
83   <field name="Registration_Number"/>
84   <field name="Students_Name"/>
85   <field name="Students_Email"/>
86   <field name="Date_of_Birth"/>
87   <field name="Fathers_Name"/>
88   <field name="Mothers_name"/>
89
90 </group>
91 <group>
92   <br></br>
93   <br></br>
94   <field name="Student_Phone_Number"/>
95   <field name="Fathers_Phone_Number"/>
96   <field name="Mothers_Phone_Number"/>
97   <field name="Local_Guardian_Phone_Number"/>
98   <field name="Result"/>
99   <field name="Present_Address"/>
100  <field name="Permanent_Address"/>
101  <field name="Interested_Areas"/>
102
103 </group>
104 <group>
105   <h2>Other information (Social Media)</h2>

```

Figure: Front in Alumni registration

```

174 <record id="register_action" model="ir.actions.act_window">
175 <field name="name">Register</field>
176 <field name="type">ir.actions.act_window</field>
177 <field name="res_model">alumni.register</field>
178 <field name="view_mode">tree,kanban,form</field>
179 <field name="help" type="html">
180 <p class="o_view_content_smalling_face">
181 Alumni Students!
182 </p>
183 </field>
184 </record>
185
186 <menuitem id="Alumni_root"
187 name="Alumni"
188 sequence="10"/>
189
190 <menuitem id="Alumni_Register_root"
191 name="Register"
192 parent="Alumni_root"
193 sequence="10"/>
194
195 <menuitem id="Alumni_Register"
196 name="Register"
197 parent="Alumni_Register_root"
198 action="Partner.action"
199 </menuitem>

```

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <odoo>
3 <template id="tmp_customer_form" name="Partner Form">
4 <t t-name="website.registration-1">
5 <t t-call="website.layout">
6 <div id="wrap" class="oe_empty oe_structure">
7 <section class="o_colored_level_s_title pb0 pt8" data-vcss="001" data-snippet="s_title"
8 data-name="Title" style="background-color: rgb(57, 132, 198) !important;">
9 <div class="s_allow_columns container">
10 <div class="row">
11 <div class="o_colored_level_col-lq-12">
12 <h4 style="text-align: center;">
13 <font style="" class="text-180">Registration</font>
14 </h4>
15 </div>
16 </div>
17 </section>
18 </t>
19 </t>
20 <section class="o_colored_level_pb16 pt16 s_website_form undefined bg-288 o_half_screen_height"
21 data-vcss="001" data-snippet="s_website_form" data-name="Form" style="">
22 <div class="container">
23 <form action="/website_form/" method="post" enctype="multipart/form-data"
24 class="o_mark_required" data-mark="*" data-success-mode="nothing"

```

Figure: Admin Dashboard code

```

1 # -*- coding: utf-8 -*-
2 import ...
3
4
5 class PartnerForm(http.Controller):
6     @http.route(['/customer/form'], type='http', auth="public", website=True)
7     def partner_form(self, **post):
8         return request.render("alumni_Management.tmp_customer_form", {})
9
10    @http.route(['/customer/form/submit'], type='http', auth="public", website=True)
11    def customer_form_submit(self, **post):
12        partner = request.env['res.partner'].create({
13            'name': post.get('name'),
14            'email': post.get('email'),
15            'phone': post.get('phone')
16        })
17
18        vals = {
19            'partner': partner,
20        }
21        return request.render("alumni_Management.tmp_customer_form_success", vals)

```

```

4 class AlumniManagement(models.Model):
5     _name = "alumni.register"
6     _description = "Alumni Management"
7
8     STATES = [
9         ('draft', 'Draft'),
10        ('approved', 'Approved')
11    ]
12
13    # Basic Information
14    state = fields.Selection(selection=STATES, string='State', default='draft')
15    Faculty = fields.Char(string='Faculty ', required=False)
16    Department = fields.Char(string='Department', required=False)
17    Semester = fields.Char(string='Semester', required=False)
18    Session = fields.Char(string='Session', required=False)
19    Batch = fields.Char(string='Batch', required=False)
20    Board_Roll = fields.Char(string='Board Roll', required=False)
21    Registration_Number = fields.Char(string='Registration Number', required=False)
22    Students_Name = fields.Char(string='Students Name', required=True)
23    Students_Email = fields.Char(string='Students Email', required=True)
24    Date_of_Birth = fields.Date(string='Date of Birth', required=False)
25    Fathers_Name = fields.Char(string='Fathers Name', required=False)
26    Mothers_name = fields.Char(string='Mothers name', required=False)
27    Student_Phone_Number = fields.Char(string='Student Phone Number', required=False)
28    Fathers_Phone_Number = fields.Char(string='Fathers Phone Number', required=False)

```

Figure: Admin Dashboard py file

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <odoo>
3 <record id="Register_tree" model="ir.ui.view">
4 <field name="name">alumni.register.tree</field>
5 <field name="model">alumni.register</field>
6 <field name="arch" type="xml">
7 <tree>
8 <field name="Faculty"/>
9 <field name="Department"/>
10 <field name="Semester"/>
11 <field name="Session"/>
12 <field name="Batch"/>
13 <field name="Board_Roll"/>
14 <field name="Registration_Number"/>
15 <field name="Students_Name"/>
16 <field name="Students_Email"/>
17 <field name="Date_of_Birth"/>
18 <field name="Fathers_Name"/>
19 <field name="Mothers_name"/>
20 <field name="Student_Phone_Number"/>
21 <field name="Fathers_Phone_Number"/>
22 <field name="Mothers_Phone_Number"/>
23 <field name="Local_Guardian_Phone_Number"/>
24 <field name="Result"/>

```

Figure: Event creates code

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <odoo>
3 <record id="event_tree" model="ir.ui.view">
4 <field name="name">alumni.event.tree</field>
5 <field name="model">alumni.event</field>
6 <field name="arch" type="xml">
7 <tree>
8 <field name="Event_Title"/>
9 <field name="Location"/>
10 <field name="Phone_Number"/>
11 <field name="start_date"/>
12 <field name="end_date"/>
13 <field name="description"/>
14 </tree>
15 </field>
16 </record>
17
18 <record id="event_form" model="ir.ui.view">
19 <field name="name">alumni.event.form</field>
20 <field name="model">alumni.event</field>
21 <field name="arch" type="xml">
22 <form>
23 <header>
24 <button name="approve_alumni" states="draft" string="Approve" type="object"

```

```

1 from odoo import api, fields, models
2 from odoo.exceptions import ValidationError
3
4
5 class AlumniManagement(models.Model):
6     _name = "alumni.event"
7     _description = "Alumni Management"
8
9     STATES = [
10         ('draft', 'Draft'),
11         ('approved', 'Approved')
12     ]
13
14     state = fields.Selection(selection=STATES, string='State', default='draft')
15     Event_Title = fields.Char(string='Event Title ', required=False)
16     Location = fields.Char(string='Location', required=False)
17     Phone_Number = fields.Char(string='Phone Number', required=False)
18     start_date = fields.Datetime(string='Start Date', required=False)
19     end_date = fields.Datetime(string='End Date', required=False)
20     description = fields.Text(string='Description', required=True)
21
22     def approve_alumni(self):
23         self.state = 'approved'
24

```

Figure: Event creates

Access of this user:

```

1 id,name,model_id:id,group_id:id,perm_read,perm_write,perm_create,perm_unlink
2 access_alumni_register,access.alumni.register,model_alumni_register,,1,1,1,1
3 access_alumni_event,access.alumni.event,model_alumni_event,,1,1,1,1
4 access_alumni_donation,access.alumni.donation,model_alumni_donation,,1,1,1,1

```

Figure: Access of this user

11/17/21, 12:43 PM

Turnitin

Turnitin Originality Report

Processed on: 17-Nov-2021 12:32 +06
 ID: 1705369374
 Word Count: 4831
 Submitted: 1

191-16-434 By Md. Anisuzzaman

Similarity Index

13%

Similarity by Source

Internet Sources: 4%
 Publications: 1%
 Student Papers: 12%

7% match (student papers from 29-Dec-2019)

[Submitted to Daffodil International University on 2019-12-29](#)

1% match (student papers from 30-Jul-2020)

[Submitted to Daffodil International University on 2020-07-30](#)

1% match (student papers from 04-Jan-2016)

[Submitted to University of Greenwich on 2016-01-04](#)

1% match (Internet from 01-Oct-2021)

<http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/5401/181-16-274%20%2816%25%29.pdf?isAllowed=y&sequence=1>

1% match (student papers from 16-Jun-2021)

[Submitted to Middle East College of Information Technology on 2021-06-16](#)

< 1% match (student papers from 15-Sep-2019)

[Submitted to Daffodil International University on 2019-09-15](#)

< 1% match (student papers from 25-Jun-2020)

[Submitted to Daffodil International University on 2020-06-25](#)

< 1% match (student papers from 28-Apr-2016)

[Submitted to University of Greenwich on 2016-04-28](#)

< 1% match (student papers from 18-Dec-2017)

[Submitted to University of Greenwich on 2017-12-18](#)

< 1% match (student papers from 03-Jun-2019)

[Submitted to University of Greenwich on 2019-06-03](#)

< 1% match (student papers from 18-Dec-2017)

[Submitted to University of Greenwich on 2017-00-00](#)

< 1% match (student papers from 18-Dec-2017)

[Submitted to University of Greenwich on 2017-12-18](#)

< 1% match (student papers from 27-Jan-2015)

[Submitted to University of Greenwich on 2015-01-27](#)

< 1% match (student papers from 22-May-2018)

[Submitted to University of Greenwich on 2018-05-22](#)

< 1% match (Internet from 07-Apr-2021)

http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/5130/P14906%20%2820_%29CSE.pdf?isAllowed=y&sequence=1

< 1% match (student papers from 06-Nov-2018)

[Submitted to NCC Education on 2018-11-06](#)

< 1% match (student papers from 19-Aug-2020)

[Submitted to NCC Education on 2020-08-19](#)https://www.turnitin.com/newreport_printview.asp?eq=1&eb=1&esm=10&oid=1705369374&sid=0&n=0&m=2&svr=29&r=27.455654182532953&lang=e... 1/7

11/17/21, 12:43 PM

Turnitin

< 1% match (Internet from 08-Feb-2019)

https://repository.up.ac.za/bitstream/handle/2263/62776/Kiptoo_Ontology_2017.pdf?isAllowed=y&sequence=1

< 1% match (Internet from 07-Apr-2021)

<https://www.slideshare.net/mahbubdiit/employee-leave-management-system>

TITLE OF THE PROJECT Alumni Management System Submitted By MD.Anisuzzaman (Id: 191-16-434) Supervised By Abdullah Bin Kasem Bhuiyan Lecturer (DIU) Summer 2021 Course: Internship Course code: CIS499 Department of Computing and Information System (CIS) DAFFODIL INTERNATIONAL UNIVERSITY Submission Date: 16- November-2021 Chapter 1 – Introduction 1.1 Introduction: A communication establishment between graduates and current students may be very useful in several things. As an example, the establishment is also fascinated by extending a request for a guest lecture to one of its alumni who could be appallingly sure-fire bourgeois currently. However while not having correct contact the establishment can pay time tracing the contact. Thus, maintaining consistent information goes hand-in-hand with establishing correct communication. The project of this method is to assist in accomplishing this, wherever alumni will communicate among themselves, with their juniors, and with the organization. In this fashion, they will build a robust network of execs. This method proves to be helpful to this student of the establishment since they get the contacts of their seniors from whom they can rummage around for opportunities and storage. The admin will communicate with the alumni