A

Major Project Report On

**ALUMNI MANAGEMENT SYSTEM**

*Submitted for partial fulfillment of the requirements for the award of the degree of*

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

**BY**

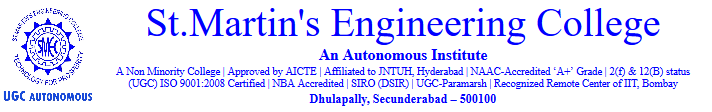
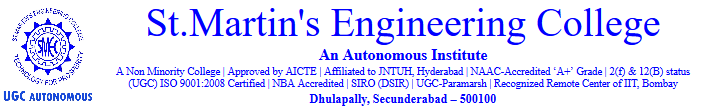
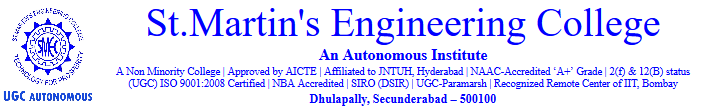
|  |  |
| --- | --- |
| **Mr. Siddharth Ramawat** | **16K81A05G8** |
| **Ms. Medam Sai Sirisha** | **16K81A05E7** |
| **Mr. Kamatala Ashish** | **16K81A05D5** |
|  |  |

UNDER THE GUIDANCE OF

**Mr. S. Raviteja**

**Associate Professor, Department of CSE**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**



**Certificate**

This is to certify that the project work entitled “**Alumni Management System**” that is submitted by **Mr. Siddharth Ramawat (16K81A05G8), Mr. Kamatala Ashish (16K81A05D5), Ms. Medam Sai Sirisha(16K81A05E7)** in partial fulfillment of the requirement for the requirement for the award of degree of **BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING** is a record of bonafide work carried out by them. The results embodied in this report have been verified and found satisfactory.

**Internal Guide Internal Examiner**

**Dr. S. A. Kalaiselvan**

**External Examiner Head of the Department**

**Dr. P. Udayakumar**

**Acknowledgement**

The satisfaction and euphoria that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose encouragement and guidance have crowded our efforts with success.

We extend our deep sense of gratitude to Principal, **Dr. P. SANTOSH KUMAR PATRA**, St. Martin’s Engineering College Dhulapally, for permitting us to undertake this project.

We are also thankful to **Dr. P. UDAY KUMAR**, Head of the Department, Computer Science and Engineering, St. Martin’s Engineering College Dhulapally, for his support and guidance throughout our project as well as Project Coordinator **Dr. P.K.A CHITRA**, Professor, Computer Science and Engineering department for his valuable support.

Finally, we express thanks to all those who have helped us successfully completing this project. Furthermore, we would like to thank our family and friends for their moral support and encouragement. We express thanks to all those who have helped us in successfully completing the project.

|  |  |
| --- | --- |
| Ms. Medam Sai Sirisha | 16K81A05E7 |
| Mr. Kamatala Ashish | 16K81A05D5 |
| Mr. Siddharth Ramawat | 16K81A05G8 |

i

**ABSTRACT**

Alumni management is one of the thrust areas considered focal to institutional development mostly in developing countries.

A strong alumni system plays an important role in reaping anonymous benefits for student-student networks as well as institution-student networks.

A major problem with some of the existing systems is the details of any student is exposed to anyone on the web. There is no Alumni Management System for  most of the colleges in “Telangana State”(Tier - 3).

Our system proposes an easy and interactive management portal for creating networks among students as well as institutes. The portal allows currently enrolled students as well to create networks with graduates of the organization. The system validates the students enrolled to the organization based on their Registration Number collected from the Institute/Organization.

Our system makes it easier for users to register into the system, connect with the alumni of organization, easy creation of events like postings on jobs, events happening around the city etc. The users will be notified about the new events and the users can register for a particular event from their feed.

ii

**CONTENTS**

Acknowledgement i

Abstract ii

List of Figures iii

List of Abbreviations iv

List of Tables v

1. Introduction

iii

**LIST OF FIGURES**

iv

**LIST OF ABBREVIATIONS**

v

**LIST OF TABLES**

vi

**1. INTRODUCTION**

**1.1 Domain Description**

**1.1.1 Web Programming**

Web programming, also known as web development, is the creation of dynamic web applications. Web development is the work involved in developing a website for the Internet (World Wide Web) or an intranet (a private network).Web development can range from developing a simple single static page of plain text to complex web-based internet applications (web apps), electronic businesses, and social network services. It includes aspects such as web design, web publishing, web programming, and database management.

Examples of web applications are social networking sites like Facebook or e-commerce sites like Amazon.

In fact, many argue it’s the best form of coding for beginners to learn. It’s easy to set up, you get instant results and there are a lot of online resources, videos and documentation of various technologies available.

People today are learning web development because they want to create a new startup or find a job in the industry. It’s super easy to get started hence the best choice to begin development with. No matter whether you’re looking for a career or just want to learn coding, learning how to develop for the web is for you. It’s one of the smartest decisions you will ever make.

The two broad categories into which web development is divided into is front-end development (sometimes called as client side development) and back-end development (sometimes called as server side development).

**1.1.1.1 Front end Development**

Front end development mainly refers to the construction of web pages that are loaded when the user starts a web application, basically the User Interface, and how the web page responds and handles the events which are fired by the user by interacting with components on the webpage.

For front-end development there are many booming technologies coming up which mainly are based on HTML, CSS, Javascript .

HTML expands itself as Hypertext Markup Language, is a special code for ‘marking up’ text in order to turn it into a web page. Every web page on the net is written in HTML, and it will form the backbone of any web application. CSS, short for Cascading Style Sheets, is a code for setting style rules for the appearance of web pages. CSS handles the cosmetic side of the web. Finally, JavaScript is a scripting language that’s widely used to add functionality and interactivity to web pages.

HTML is the standard markup language for Web pages.HTML elements are the building blocks of HTML pages.HTML elements are represented by <> tags.

An HTML element is a start tag and an end tag with content in between.

All HTML documents must start with a document type declaration: <!DOCTYPE html>.The HTML document itself begins with <html> and ends with </html>.The visible part of the HTML document is between <body> and </body>.

Javascript is used to define the behavior of the page. It is also used to make the page responsive. Javascript lets the users to enable dynamic loading of the content to be displayed on the webpage apart from static loading.

Responsive Web Design is used in all types of modern web development.

ECMAScript 5 (JavaScript 5) is supported in all modern browsers.

Popular Javascript libraries used these days are React.Js, React Native, Angular.Js, Angular 4,Vue.Js,W3.Js .

The frameworks used for creation of the responsive webpages are BootStrap, Material Design, W3.CSS.

**1.1.1.2 Back end Development**

Back-end development controls what goes on behind the scenes of a web application. A back-end often uses a database to generate the front-end.

Back-end scripts are written in many different coding languages and frameworks, such as:

1. PHP
2. Ruby on Rails
3. ASP.NET
4. Perl
5. Java
6. Node.js
7. Python

Back end developers are most focused on a site’s responsiveness and speed. These languages are used to create dynamic sites which are different from static sites in that these types of websites store database information. Content on the site is constantly changing and updating. Examples of dynamic sites include Facebook, Twitter, and Google Maps.

Backend development languages handle the ‘behind-the-scenes’ functionality of web applications. It’s code that connects the web to a database, manages user connections, and powers the web application itself. Backend development works in tandem with the front end to deliver the final product to the end user.

Backend programming can either be Object Oriented (OOP) or Functional.

The former is the technique that focuses on the creation of objects. With object-oriented programming, statements should be executed in a particular order. Popular OOP languages are Java, .NET, and Python,

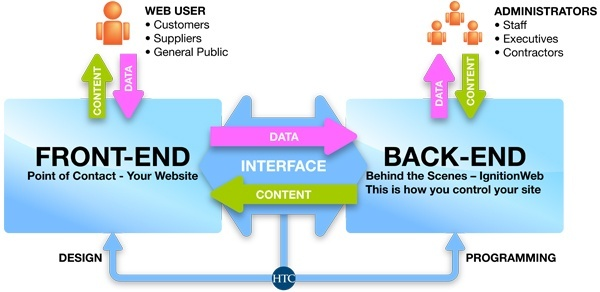
The latter is a technique that is more “action”-based. Functional programming uses declarative language, which means that statements can be executed in any order. It’s commonly used for data science, and popular languages are SQL, F#, and R.

Languages can either be statically typed or dynamically typed. The former is more rigid, but better at catching errors, whereas the latter is more flexible but allows for variables to change types (which could account for unexpected errors).

Back end web infrastructure consists of specific things like:

1. Databases—data storage applications that provide websites with dynamic content updates (e.g. when you check your bank account balance from a bank website, the site accesses your account information from a database, causing your balance to update on your screen).
2. Server scripts—scripts are sets of instructions written in code that tell computer programs to “do something.” Back end or “server side” scripts allow a website’s servers (web hosting hardware where a site’s images, videos, and other assets are stored) to respond to actions and commands from the front end (e.g. retrieving an image or a video from a server and displaying it on a user’s screen).
3. APIs—APIs (dev speak for Application Programming Interface) are sets of routines, protocols, and tools that allow applications to communicate with each other. When you share an article you just read to Facebook or Twitter with the click of a “share” button, it’s an API that allows that cross platform sharing to happen.

Back end developers build and maintain these server-side applications and tools, and in the process add a whole lot of function and utility to what users see on their computer, phone, or tablet screen.



**1.1.2 Android Development**

Android software development is the process by which new applications are created for devices running the Android operating system. Google states that "Android apps can be written using Kotlin, Java, and C++ languages" using the Android software development kit (SDK), while using other languages is also possible. All non-JVM languages, such as Go, JavaScript, C, C++ or assembly, need the help of JVM language code, that may be supplied by tools, likely with restricted API support.

Mobile app development is the creation of software intended to run on mobile devices and optimized to take advantage of those product's unique features and hardware. Mobile application development is the process of creating software applications that run on a mobile device, and a typical mobile application utilizes a network connection to work with remote computing resources. Hence, the mobile development process involves creating installable software bundles (code, binaries, assets, etc.) , implementing backend services such as data access with an API, and testing the application on target devices.

There are two dominant platforms in the modern smartphone market. One is the iOS platform from Apple Inc. The iOS platform is the operating system that powers Apple's popular line of iPhone smartphones. The second is Android from Google. The Android operating system is used not only by Google devices but also by many other OEMs to built their own smartphones and other smart devices.

The types of mobile apps that developers create include native apps, hybrid apps and HTML5 apps.

The creation of mobile applications draws much of its roots from traditional software development. The end result, however, is software intended to utilize the unique features and hardware of mobile devices. Modern smartphones are equipped with Bluetooth, NFC, gyroscopic sensors, GPS, cameras and so much more. They can be used for virtual or augmented reality, barcode scanning and more. Mobile apps should utilize the full range of smartphone features, which is easier said than done.

With desktop PC software development, programmers must create an application that can operate on a minimum set of hardware. The same goes for mobile applications, though the hardware variances in this instance are much more minimal. At the same time, the hardware in smartphones and tablets doesn't quite match that in laptops and desktop computers, which means mobile apps must be designed to show optimal performance.

For example, a gaming app would be limited in its graphical elements because of the less-powerful graphics processors in mobile devices. With that said, cloud computing is making it easier than ever to accomplish mobile gaming. Popular games such as Fortnite , Hearthstone, and PUBG connect players across computers, phones and even consoles.

**1.1.2.1 Mobile App Development Considerations:**

Solving the issue of performance on any given device is ultimately dependent on developing an app natively on that device. This means designing the code specifically for the hardware on a particular device. In the instance of iOS devices, this proves quite easy, as mobile developers only need versions of the app for the iPhone and iPad to achieve universal usability. For Android devices, however, each smartphone and tablet runs on different hardware and varying versions of the operating system.

Web-based apps, on the other hand, don't depend on the device; they run off of a web browser, making them cheaper to develop and easier to access. The problem with web apps, however, is that their performance doesn't compare to that of a native app. For example, with web apps, you cannot use the phone's full features or send proper notifications, and they look less professional.

**1.1.2.2 Types of Mobile Apps and Programming Languages:**

Like desktop software, mobile apps are designed using a wide range of programming languages and frameworks. While the most popular operating systems, iOS and Android, have done an excellent job of standardizing the types of mobile app development available for programmers, apps can still vary. Here are some mobile app types:

1. Native apps. These are apps created for a specific platform (iOS or Android) using the software development tools and languages supported by those operating systems. iOS uses Xcode and Objective-C, whereas Android uses Eclipse and Java. Developers often prefer native apps because of their ability to utilize a device's full potential. With smart home devices becoming more ubiquitous, developers are creating unique applications that integrate things like wearables, IoT sensors and smart screens for personalized experiences. Of course, development for each platform is easier said than done and is a costly and time-consuming process that doesn't work for all businesses.
2. HTML5 apps. Based on the near-universal standards of web technologies – namely, HTML5, JavaScript and CSS – this type of mobile app takes a write-once-run-anywhere approach to mobile development. Apps developed in this framework are compatible with many platforms and require only minimal changes to ensure complete functionality in each operating system. HTML5 apps can still send desktop notifications and trigger interactions through email and other avenues. Don't discount web apps' usability, but keep in mind that users are more likely to use a native app. A study from Oracle found that millennials spend 90% of their mobile time in apps, compared with 10% in web browsers.
3. Hybrid apps. These apps entail the creation of a container developed in the native system that makes it possible to embed an HTML5 app within it. This allows apps to make use of the diverse and unique elements of each native system. Before creating your own branded app, consider instead utilizing already existing apps for greater impact. For example, by using mobile-focused marketing on services such as Yelp, Facebook and Google Maps, you can drive traffic to both your website and brick-and-mortar location.
4. Cross-Platform Applications: Cross-platform native mobile applications can be written in variety of different programming languages and frameworks, but they are compiled into a native application running directly on the operating system of the device.
5. Hybrid-Web Applications: Hybrid mobile applications are built with standard web technologies - such as JavaScript, CSS, and HTML5 - and they are bundled as app installation packages. Contrary to the native apps, hybrid apps work on a 'web container' which provides a browser runtime and a bridge for native device APIs via Apache Cordova.
6. Progressive Web Applications: PWAs offer an alternative approach to traditional mobile app development by skipping app store delivery and app installations. PWAs are web applications that utilize a set of browser capabilities - such as working offline, running a background process, and adding a link to the device home screen -  to provide an 'app like' user experience.

**1.1.2.3 Software Development Kits:**

Mobile app development requires access to software development kits (SDKs) that provide an environment through which programmers can design and test code in a simulated mobile environment. However, creating an app does not require full use of these kits. For example, developers can create mobile games using Unity and then use the Android SDK to ensure its deliverability on mobile devices. Developing apps for iOS requires a paid iOS developer license, whereas the Android SDK is freely available to users.

iOS (47%) and Android (52%) have similar mobile market shares, but developing for Apple is somewhat easier in that you don't need to worry about a wide range of devices from different manufacturers. Regardless of which operating system you choose, however, there are barriers to entry.

**1.1.2.4 Mobile Application Development Services:**

Mobile application development is changing constantly. Typically, every six months or so, a new version of an operating system rolls out with unique features that mobile apps can utilize. Developing for a specific version of the operating system, or even for a native operating system, usually requires developers to try multiple solutions to find the one that suits their development needs.

**1.2 Objective:**

Our system proposes an easy and interactive management portal for creating networks among students as well as institutes. The portal allows currently enrolled students as well to create networks with graduates of the organization. The system validates the students enrolled to the organization based on their Registration Number collected from the Institute/Organization.

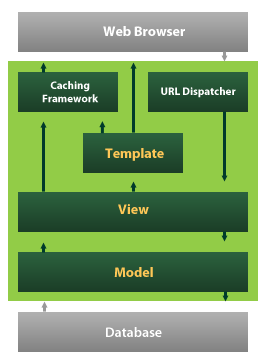
Our system makes it easier for users to register into the system, connect with the alumni of organization, easy creation of events like postings on jobs, events happening around the city etc. The users will be notified about the new events and the users can register for a particular event from their feed.

**2.Overview Of Technologies Used:**

**2.1 Model View Template Architecture:**

The MVT (Model View Template) is a software design pattern. It is a collection of three important components Model View and Template. The Model helps to handle database. It is a data access layer which handles the data.

The Template is a presentation layer which handles User Interface part completely. The View is used to execute the business logic and interact with a model to carry data and renders a template.



**2.2 Django Framework:**

Django is a Python’s framework that enables developers to develop web applications.

It allows the developers to develop the apps , while Django does the setup of the project (major boiler plate code).The major features of Django framework which makes it the best choice for the development is:

1. Ridiculously fast: Developers can design the application and complete writing the code in no time.
2. Reassuringly secure: It helps developers avoid making common mistakes.
3. Highly scalable : Django provides flexibility to scale the users efficiently.
4. Open source : Django is open source and its documentation can be referred from the link :- <https://docs.djangoproject.com/en/3.0/>

Although Django follows MVC pattern but maintains it’s own conventions. So, control is handled by the framework itself.

There is no separate controller and complete application is based on Model View and Template. That’s why it is called MVT application.

The following graph shows the MVT based control flow.



Here, a user requests for a resource to the Django, Django works as a controller and check to the available resource in URL.

If URL maps, a view is called that interact with model and template, it renders a template.

Django responds back to the user and sends a template as a response.

**2.2.1 Prerequisites to install Django:**

To install Django 2.0 versions or above make sure you have Python 3.0 or above installed into your system.

You can install Django using the command:

Python -m pip install Django

**2.2.2 Steps to create and run the project:**

1. To create a new project in Django use the command:

Django-admin startproject <project\_name>

1. Change the path to your project directory.
2. Then create a new app in your project use the command:

Python manage.py startapp <app\_name>

1. To run the project

Python manage.py runserver

1. Configure the settings.py file in your folder in order to add database credentials.
2. In order to create database tables, edit the models.py file of the app.
3. Make sure you add your app’s configuration into the INSTALLED\_APPS list of settings.py
4. To add the new changes to the database , use the command

Python manage.py makemigrations

1. To make those changes permanent to the database, use the command

Python manage.py migrate.

1. To register your models, use the following steps in admin.py of the app
   1. First import your models into admin.py
   2. Add this line : admin.site.register(<model\_name>)

**2.2.3 Overview of Django:**

In a traditional data-driven website, a web application waits for HTTP requests from the web browser (or other client). When a request is received the application works out what is needed based on the URL and possibly information in POST data or GET data. Depending on what is required it may then read or write information from a database or perform other tasks required to satisfy the request. The application will then return a response to the web browser, often

dynamically creating an HTML page for the browser to display by inserting the retrieved data into placeholders in an HTML template.

Django web applications typically group the code that handles each of these steps into separate files:



* URLs: While it is possible to process requests from every single URL via a single function, it is much more maintainable to write a separate view function to handle each resource. A URL mapper is used to redirect HTTP requests to the appropriate view based on the request URL. The URL mapper can also match particular patterns of strings or digits that appear in a URL and pass these to a view function as data.
* View: A view is a request handler function, which receives HTTP requests and returns HTTP responses. Views access the data needed to satisfy requests via models, and delegate the formatting of the response to templates.
* Models: Models are Python objects that define the structure of an application's data, and provide mechanisms to manage (add, modify, delete) and query records in the database.
* Templates: A template is a text file defining the structure or layout of a file (such as an HTML page), with placeholders used to represent actual content. A view can dynamically create an HTML page using an HTML template, populating it with data from a model. A template can be used to define the structure of any type of file; it doesn't have to be HTML!

Just a few of the other things provided by Django include:

* Forms: HTML Forms are used to collect user data for processing on the server. Django simplifies form creation, validation, and processing.
* User authentication and permissions: Django includes a robust user authentication and permission system that has been built with security in mind.
* Caching: Creating content dynamically is much more computationally intensive (and slow) than serving static content. Django provides flexible caching so that you can store all or part of a rendered page so that it doesn't get re-rendered except when necessary.
* Administration site: The Django administration site is included by default when you create an app using the basic skeleton. It makes it trivially easy to provide an admin page for site administrators to create, edit, and view any data models in your site.
* Serialising data: Django makes it easy to serialise and serve your data as XML or JSON. This can be useful when creating a web service (a website that purely serves data to be consumed by other applications or sites, and doesn't display anything itself), or when creating a website in which the client-side code handles all the rendering of data.

**2.2.3.1 Urls.py**

A URL mapper is typically stored in a file named urls.py. In the example below, the mapper  (urlpatterns) defines a list of mappings between routes (specific URL patterns) and corresponding view functions. If an HTTP Request is received that has a URL matching a specified pattern then the associated view function will be called and passed the request.

urlpatterns = [

    path('admin/', admin.site.urls),

]

The urlpatterns object is a list of path() and/or re\_path() functions.

The first argument to both methods is a route (pattern) that will be matched. The path() method uses angle brackets to define parts of a URL that will be captured and passed through to the view function as named arguments. The re\_path() function uses a flexible pattern matching approach known as a regular expression.

The second argument is another function that will be called when the pattern is matched.

**2.2.3.2 Models.py :**

Django web applications manage and query data through Python objects referred to as models. Models define the structure of stored data, including the field types and possibly also their maximum size, default values, selection list options, help text for documentation, label text for forms, etc. The definition of the model is independent of the underlying database — you can choose one of several as part of your project settings. Once you've chosen what database you want to use, you don't need to talk to it directly at all — you just write your model structure and other code, and Django handles all the dirty work of communicating with the database for you.

**2.2.3.3 Views.py:**

The Django model provides a simple query API for searching the database. This can match against a number of fields at a time using different criteria (e.g. exact, case-insensitive, greater than, etc.), and can support complex statements (for example, you can specify a search on U11 teams that have a team name that starts with "Fr" or ends with "al").

**2.2.3.4 HTML Templates:**

Template systems allow you to specify the structure of an output document, using placeholders for data that will be filled in when a page is generated. Templates are often used to create HTML, but can also create other types of document. Django supports both its native templating system and another popular Python library called Jinja2 out of the box (it can also be made to support other systems if needed).

**2.2.4 Django Working Process:**

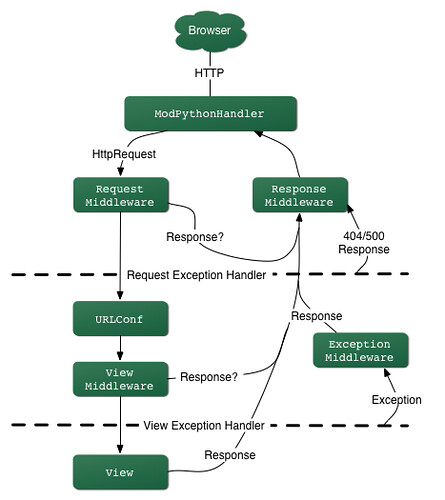
The simplest way to look at Django is to break it down into its component parts. First off, there’s a models.py file which defines your data model by extrapolating your single lines of code into full database tables and adding a pre-built (totally optional) administration section to manage content.

The next element is the urls.py file which uses regular expressions to capture URL patterns for processing.

The actual processing happens in your views which, if you haven’t seen the pattern yet, live in views.py. This is really the meat of Django, since views are where you grab the data you’re presenting to the visitor.

Here’s what happens when a visitor lands on your Django page:

1. First, Django consults the various URL patterns you’ve created and uses the information to retrieve a view.
2. The view then processes the request, querying your database if necessary.
3. The view passes the requested information on to your template.
4. The template then renders the data in a layout you’ve created and displays the page.



**3. LITERATURE SURVEY:**

Alumni are the living examples and testimonials of any organisation. It’s because of the strong alumni network that leads to the recognition and fame of the college.

A constant and active involvement of the alumni with the institute proves to be very useful for the college as well as students.

Alumni Management System is used to connect with the students who have passed out of the college(ex-students).

An Online Alumni Tracking System is an example of web application which is under the information systems. It helps an academic institution in tracking its alumni. Also, it helps the alumni to communicate with the institution through the use of the internet. It also helps the alumni to get updated with the latest news and upcoming events of the institution. This application can easily be accessed through the use of the internet which will be very useful to the alumni because they can keep in touch with the institution even if they do not visit the school. This application can be very useful especially to those alumni who are now living abroad because they can still get connected with their fellowmen and the institution. This application is also useful because it can make transactions and process paperless.

Nowadays, computers have infiltrated all the aspects of our society. The computer is most likely one of the great technological mechanism for future change. It can now simply make our works easier and lighter. With this great thing it won’t be more useful without the computer’s software. Software is a generic term for organized collections of computer data and instructions, often broken into two major categories: system software that provides the basic non-task-specific functions of the computer, and application software which is used by users to accomplish specific tasks.

The paper “Centralized Alumni System- A Prototype Proposal” proposes a system which is institution dependent.There are systems developed  for students studying in any educational institute who are above the age of 15.

“Online Alumni System “  proposes a system which allows anyone on the web to access and search information about any student.

“Alumni Network Analysis“ is a system that evaluates educational institutes and ranks them based on the data collected about the network connections of alumni in the entrepreneurial and technological domain.

Features of an Alumni Information System

According to Webaloo (2007) an alumni builders systems features includes the following: Alumni class page- a directory of alumni names by class year with links to individual profiles and email addresses; Alumni Profile – is controlled by individual users and displays only the information that he/she wants to display; Alumni Search – allows the users to search by name, class, occupation, address, etc.; Alumni Forum – offers alumni a way to stay in touch with classmates and friends form other graduation years; Alumni Notes – allows classmates to communicate by posting on a notes page; Secure Log-in – use to block sensitive alumni information from other school constituents; Profile Change Report – allows the school to keep track of the personal information that alumni can update; and Missing Alumni Page – helps the school reconnect with graduates whose personal information is outdated.

Alumni Portal

According to Goodwin College (2012) the use of Alumni Portal will keep the alumni administrators updated and to keep in the know! The Alumni Portal has been activated so that you can: Update your contact information in order to receive important communication and invitations to events and programs from the Alumni Association; Notify us of a new job or job change, family additions or other news; Access College Central Network to search for jobs, post your resume, access informative career related documents, videos, podcasts and much more; Browse the events calendar for upcoming happenings; Search for classmates, students and faculty and staff; View course and grade history; and access your 1098-T.

Web-based Alumni Information System of West Negros University

The web-based Alumni Information System of West Negros University (2010) is capable of gathering information of all the alumni using the web application form. The system helps the administrator, alumni personnel and even the public relation office in maintaining the data and can easily send and display information for all concerns.  The system is internet-based that can access all online portals and connect to all networks that enable to collect all the data as needed by the university particularly by the alumni office. The system is also a social network that provides space for chatting, forum blog and photo gallery functions.

On Student Information

In the study conducted by Agudera and Mendiola (2013) the implementation of Alumni Information System, the system will be secured, the process in having the student information will become faster, and accurate generating reports. The system will secure student files using the log-in log-out form for the unauthorized users. The developers design the interface so the users will understand the system easily and use the Microsoft Access for database to store important information. The system will help a school to be well-organized.

UMT Alumni Information System

The role of information system can’t be ignored doing things faster, doing things better, and doing thinks smarter these all traits are possible just because of two words, Information system. Alumni information system is one of the examples of information system.  To get contact with the old students and to provide the assistance to this old student for their future progress in all field of life and maintain the record of the students. Following core aims and objective can describe the real need of the AIS. The aims and objectives of UMT Alumni information system are to encourage alumni to maintain links with the University and with each other, in order: to promote more effectively the welfare and interests of the University and its alumni; to support the University’s aims and objectives and uphold its reputation as an ambassadors of the University; to establish mutually beneficial relationship between the University and its alumni and to bind the alumni more closely together; to assist in developing financial and other resources for the University and the Alumni Association; to develop linkages for mutual benefit (such as research) with other professional alumni bodies, and to remain the part of the university even after the study.

**4.Issues in Existing System**

1. The existing study stated shows that alumni information system contributes to a good relationship between them and the school.
2. The existing studies designs and implementations get involved the web as well as the Online Alumni Information System however our system informs only the alumni of the college and enable them to interact with each other. It also provides career service opportunities to all the graduates of the college.
3. Existing system is a manual one in which users are maintaining documents paper work to store the information like colleges details, student details. It is very difficult to maintain historical data.
4. It is difficult to maintain important information in documents paper work. More manual hours need to generate required reports.
5. It is tedious to manage historical data which needs much space to keep all the previous years, ledgers, documents paper work.

**5.Proposed System:**

* Our system is proposed for the alumni of various institutions across Telangana State, as there is no current existing system for the students of Tier - 3 colleges in Telangana.
* Through this system we hope to achieve student-alumni and alumni-institution interactions.
* The currently enrolled students of the institutions can contact their alumni working in various organizations or pursuing higher studies in other reputed institutions, this helps guide the students for their future, as they learn from the experiences of their seniors.
* The system also allows the institution to conduct events for the alumni such as the Alumni Meets for the alumni to get together and rendezvous with their batchmates providing a harmonious environment for them to interact with each other.
* This system allows students of the institution to interact with the alumni using the chatting system.
* The students can login using their google auth or linkedin which maintains the security of their data.
* We also allow the alumni or the institution to arrange events and publish on the platform.
* There is also a feedback system which helps the students/users to give feedback about the curriculum , college or generic feedback. The feedback can be viewed by the admin.
* There is a search tab on the dashboard which makes it easy to search for the alumni by company, by name or by department. We are making this system interactive

**6.Hardware and Software Requirements:**

**6.1 Hardware Requirements:**

RAM : 512 MB  
Operating System : Windows/MAC/Linux

**6.2 Software Development Tools:**

Frontend Framework : Django Templates

Backend Framework : Django

Database : PostGresSql

Implementation Language : Python

IDE’s: PyCharm Community , PgAdmin

Browsers : Chrome,Safari,Explorer

Android Versions : Above 8

Additional Requirements : Django version 2.5.4 or above ,postgresql 9.5 or above,Python 3.x.y

**7.System Design**

**8. Implementation**

**9.Output**

**10.Discussion on Results:**

**11.Challenges:**

**12.References:**

[1] N Rubens et al., "Alumni network analysis," in IEEE Global Engineering Education Conference (EDUCON), 2011, pp. 606- 611.

[2] Centralized Alumni Management System (CAMS) - A Prototype Proposal

IEEE, Dept. of CSE, Techno International Batanagar, Kolkata, India

[3] K Sivaprasad, "Making best use of alumni associations for holistic development of engineering institutes," in IEEE International Conference on Engineering Education: Innovative Practices and Future Trends (AICERA), 2012, pp. 1-11.

[4] Almabay. [Online]. HYPERLINK "https://www.almabay.com/" https://www.almabay.com/

[5] AlmaConnect. [Online]. HYPERLINK "https://www.almaconnect.com/" https://www.almaconnect.com/

[6] Alumnet. [Online]. HYPERLINK "https://www.alumnet.co.za/" https://www.alumnet.co.za/

[7] almabase. [Online]. HYPERLINK "https://www.almabase.com/" https://www.almabase.com/

[8] J E Sharp, "Work in progress: alumni mentoring of engineers in a technical communication course," in 34th Annual Frontiers in Education, 2004, pp. F2F-11.

[9] J L Huff, W C Qakes, and C B Zoltowski, "Work in progress: Understanding professional competency formation in a service learning context from an alumni perspective," in 2012 Frontiers in Education Conference Proceedings, 2012, pp. 1-3.

[10] J E Sharp, "Work in Progress: Using Mock Telephone Interviews with Alumni to Teach Job Search Communication," in Frontiers in Education. 36th Annual Conference, 2006, pp. 7- 8.

[11] VIT Alumni System : https://www.vitaa.org/