## CERTIFICATE OF APPROVAL

This is to certify that the Project work entitled “**Uni Alumni web portal**” is carried out by **Harish Gupta**, a student of MSc IT-IV semester at **A.P.S.G.M.N.S. Govt. P.G. College, Kawardha (C.G.)** is hereby approved as a credible work in the discipline of Information Technology for the award of degree of  **Master of Science in Information Technology** during the year 2024-25 from **Hemchand Yadav Vishwavidyalaya, Durg** **(C.G.)**.

**Dr. Anil Kumar Sharma**

**(HOD)**

**Department of Information Technology**

**CERTIFICATE**

This is to certify that the project work entitled **“UNI ALUMNI WEB PORTAL”** submitted to **A.P.S.G.M.N.S. Govt. P.G. College, Kawardha (C.G.)** by **Harish Gupta**, Roll No: 2440165006, in partial fulfilment for the requirements relating to the nature for the award of degree of  **Master of Science in Information Technology** during the year 2024-25 from **Hemchand Yadav Vishwavidyalaya, Durg** **(C.G.)**.

This project work has been carried out under my co guidance.

**Dr. Anil Kumar Sharma (H.O.D)**

**Department of Information Technology**

**CERTIFICATE OF THE COMPANY**

**Indira Gandhi Krishi Vishwavidyalaya(IGKV)**

**Raipur, jora(C.G.)**

This is to certify that the project entitled **“UNI ALUMNI WEB PORTAL”** made by **Harish Gupta**, a student of **M.Sc.(I.T) Fourth Semester** at **A.P.S.G.M.N.S. Govt. P.G. College, Kawardha (C.G.)**, affiliated to **Hemchand Yadav Vishwavidyalaya, Durg** **(C.G.)**. will be used by us as our live website. I hereby affirm that project is according to the standards and requirements of firm. We are about to register a domain and publish this website and it is found working well and satisfying all our needs. He signed the agreement also to not to sell or give this website to anywhere else or anybody else either in part or in full. We wish him all the best for all his future assignments.

**(Signature & Seal)**

**Proprietor**

**(IGKV)**

**Raipur, jora(C.G.)**

**CERTIFICATE OF EVALUATION**

This is to certify that the project work entitled **“UNI ALUMNI WEB PORTAL”** is carried out by **Mr. Harish Gupta**, a student of **M.Sc.(I.T) Fourth Semester** at **A.P.S.G.M.N.S. Govt. P.G. College, Kawardha (C.G.)**, after proper evaluation and examination, is hereby approved as a credible work in the discipline of Information Technology and is done in a satisfactory manner for its acceptance as a requisitefor the award of degree of  **Master of Science in Information Technology** during the year 2024-25 from **Hemchand Yadav Vishwavidyalaya, Durg** **(C.G.)**.

**Internal Examiner Extern Examiner**

**DECLARATION**

This to certify that the project report entitled **“UNI ALUMNI WEB PORTAL”**,which is submitted by me in the partial fulfillment for the award of degree of  **Master of Science in Information Technology, A.P.S.G.M.N.S. Govt. P.G. College, Kawardha (C.G.),** comprises the original work carried out by me.

I further declare that the work reported in this project has been submitted and will not be submitted, either in part or in full for the award of any other degree or diploma in this Institute or any other Institute or University.

**Name:** Harish Gupta

M.Sc. I.T Fourth Semester

**Enroll No:** HU/401/20007127

**Roll No.**2440165006

**Place:** Kawardha

**Date:**

**ACKNOWLEDGEMENT**

I extend my sincere thanks and gratitude to our Head of the Department **Dr. Anil Kumar Sharma Department of Information Technology at A.P.S.G.M.N.S. Govt. P.G. College, Kawardha (C.G.),** who provided me the lab facilities as well as who steered all of us in a right direction for completing this project. I am significantly indebted and thankful to my parents for their love, sacrifice, inspiration and the valuable help that enabled us in the completion of this project.

I express my gratitude to **Mr. Abhijeet Kaushik, Dy. Director(IT),NIC** for their guidance and encouragement throughout the duration of my project development. With a deep sense of gratitude, I express my humble thanks to all the members of **Knowledge & Technology Resource Centre**, given me help & guidance to make this project.

I also thanks to Miss. Chanchal Sahu, Mr.Salman khan, Ashish pandey ,virendra sharma and all the members of **Information Technology Department at A.P.S.G.M.N.S. Govt. P.G. College, Kawardha (C.G.),** who have given me a lot of help in these two years.

At last, I also express my thanks to my friends, colleagues, and all, who constantly interacted with me during the study period.

**Sincerely,**

**Harish Gupta**

**ABSTRACT**

In today’s digitally connected world, maintaining strong relationships with alumni has become a strategic priority for academic institutions. The **UNI Alumni Web Portal** is a comprehensive web-based platform developed to bridge the gap between the university, its alumni, and current students. This portal serves as a digital hub that fosters communication, engagement, and collaboration among stakeholders while offering a centralized system to manage alumni-related data and activities efficiently.

The **primary objective** of this portal is to centralize and manage alumni data while promoting interaction and collaboration through a structured, user-friendly online environment. Designed to enhance connectivity and engagement, the platform empowers alumni to remain involved with their alma mater, share their experiences, mentor current students, and contribute to institutional development.

One of the core features of the portal is a **Blog Post Section**. This feature allows alumni to create and publish blog posts on a variety of topics, including their personal and professional experiences, achievements, career advice, and reflections on university life. These blog submissions are moderated by an admin to ensure quality and relevance. This not only gives alumni a voice but also creates a rich repository of content that benefits both students and fellow alumni.

From an administrative perspective, the portal includes robust **data management features**. The admin panel provides comprehensive tools to manage alumni records, verify user identities, and monitor content shared across the platform. The admin can also approve or reject blog posts, manage user access, and generate reports on alumni activities. These capabilities support the university’s efforts to maintain an up-to-date and dynamic alumni database.

Alumni users are given the ability to **update their personal profiles** with relevant information, such as current employment status, achievements, areas of expertise, and location. This self-service functionality ensures that the alumni records remain accurate and reflects the evolving nature of their careers. These updates are beneficial for both institutional recordkeeping and networking purposes.

A significant advantage of the portal is its **value to current students**. The platform acts as a valuable resource for them to explore alumni profiles, read blog posts, and even initiate contact with former students for advice, mentorship, or internship opportunities. This feature opens up new doors for students to learn from real-world experiences and gain insights that go beyond the classroom.

The portal also supports **internship and career opportunities**. By enabling alumni to post internship openings and share career guidance, it strengthens the connection between academia and industry. This function not only benefits current students but also fosters professional collaboration among alumni themselves.

Furthermore, the portal promotes a strong **sense of community and belonging** among alumni. Many graduates lose contact with their university after finishing their education. This platform ensures that alumni remain an active part of the university ecosystem by participating in discussions, sharing knowledge, and staying informed about university events and announcements.

Technologically, the portal is built using modern web development frameworks that ensure scalability, security, and ease of use. The user interface is designed to be responsive, allowing seamless access across desktops, tablets, and mobile devices. Security is also a key consideration, with proper authentication and authorization mechanisms in place to protect sensitive user information.

The platform provides the university with a powerful **tracking system** that allows the administration to analyze the career progression of their alumni. Such data is invaluable for accreditation purposes, marketing, and improving academic programs based on the professional outcomes of graduates. It also allows the university to recognize notable alumni and foster goodwill by highlighting their achievements.

From a social standpoint, this platform enhances the **university’s reputation** by showcasing a vibrant and successful alumni community. Engaged alumni are more likely to support their alma mater through donations, partnerships, or volunteering, contributing to the institution's long-term success.

To summarize, the **University Alumni Web Portal** serves multiple purposes:

* **For alumni**: It offers a platform to reconnect with the university, share professional updates, engage in mentoring, and contribute to the community.
* **For students**: It provides a source of guidance, role models, internship opportunities, and professional networking.
* **For the university**: It enables systematic alumni management, enhances engagement, collects valuable data, and supports strategic decision-making.

This multi-functional platform goes beyond traditional alumni directories or social media groups by offering structured engagement, verified content, and institutional oversight. It is a step forward in strengthening the bond between the university and its graduates, ensuring that this relationship continues to flourish long after graduation.

In conclusion, the **University Alumni Web Portal** is not just a technological solution; it is a community-building initiative that empowers alumni, supports students, and enhances the institution's ability to remain connected with those it has educated. It is a testament to the power of digital innovation in education and alumni relations. As universities continue to evolve in the digital age, such portals will become essential tools for creating value-driven, connected, and collaborative academic communities.

|  |  |
| --- | --- |
| **INDEX** |  |
| **CONTENTS** | **PAGENO.** |
| **CHAPTER1: INTRODUCTION** | **01 -04** |
| 1.1) About the Organization | 02 -03 |
| 1.2) About the Project | 04 |
| **CHAPTER2:SYSTEM ANALYSIS** | **05 -08** |
| **CHAPTER3:SYSTEM PLANNING** | **09 -15** |
| 3.1) Introduction | 10 |
| 3.2) PERT Chart | 10 -12 |
| 3.3) Cost & Benefit Analysis | 13 -15 |
| **CHAPTER4:FEASI BILITY STUDY** | **16 -19** |
| 4.1) Technical feasibility | 17 |
| 4.2) Economical feasibility | 17 -18 |
| 4.3) Operational feasibility | 19 |
| **CHAPTER5:SYSTEMRE QUIREMENT SPECIFICATION(SRS)** | **20 -30** |
| 5.1) SRS(Software Requirement Specifications) | 21 -22 |
| 5.2) Module description | 23 -24 |
| 5.3) Hardware & Software requirement | 25 |
| 5.4) Technology description | 26 -30 |
| **CHAPTER6:SYSTEMDESIGN** | **31 -49** |
| 6.1) Prototype model | 32 -33 |
| 6.2) Database design | 34 -39 |
| 6.3) E-R Diagram | 40 -41 |
| 6.4) Flow Chart | 43 |
| 6.5) Data Flow Diagram (DFD) | 43 -49 |
| **CHAPTER7:SYSTEM TESTING** | **50 -53** |
| **CHAPTER8:INPUT & OUTPUT SCREEN** | **54 -67** |
| **CHAPTER9:SYSTEM SECURITY** | **68 -69** |
| **CHAPTER10:LIMITATION & FUTURE ENHANCEMENT** | **70 -71** |
| **CHAPTER11: CONCLUSION** | **72 -73** |
| **CHAPTER12:REFERENCES & BIBLIOGRAPHY** | **74 -75** |

**Chapter1**

**Introduction**

**1. INTRODUCTION**

* 1. **About the Company**

**NATIONAL INFORMATION CENTRE (NIC)**

National Informatics Centre (NIC) under the Ministry of Electronics and Information Technology (MeitY) is the technology partner of the Government of India. It was established in 1976 with an objective to provide technology-driven solutions to Central and State Governments in various aspects of development. NIC has been instrumental in adopting and providing Information and Communication Technology (ICT) and eGovernance support to Central Government.

NIC has also developed several digital platforms for the socio-economic development of the country with ‘One-Nation One-Platform’ initiative to empower citizens digitally. Its services have created a perfect interaction of the Government with citizens, Government employees and businesses. With an objective of focused study of new technology, explore and experiment their use in governance, NIC has set-up Centre of Excellence (CoE) in Data Analytics, Artificial Intelligence, Blockchain and Application Security.

National Information Centre (NIC) Mantralaya, New Raipur Chhattisgarh. NIC has played an important role of an active catalyst and facilitator in informatics development programmed in Governments at the national, state and district levels, during the last 27 years.

The efforts of NIC Chhattisgarh for providing start-of-the art e-Governance solutions to state government, district government and citizen of Chhattisgarh have been recognized with various prestigious awards at National, State and District level. Chhattisgarh state center (CGSC), of NIC was setup at Raipur, in the year 2001 to provide the ICT (Information & Communication Technologies) services to government departments and organizations.

Page 3

NIC Chhattisgarh, Department of Electronics And Information Technology, Government of India is providing network backbone and e-Governance support to Chhattisgarh State Government and Districts .National Informatics Center is a premier organization in the field of Information Technology (IT) in India. NIC implementing IT.

Projects for both Central Government and State Government in the areas of:

(a) Central sector

(b) State sector

(c) District

District centers are operational in the districts with state of the art VC studios, high speed NICNET connectivity and DIO/DIA to manage the district centers activities and support various e- Governance initiatives for achieving targeted goal of delivering efficient citizen services to the common public in the State.

NIC has been instrumental in adopting Information and Communication Technology to reach out to India i.e. by implementing IT applications in Social and Public Administrations.

NIC supporting several services & projects such as bio-metric attendance, cloud, e-office, messaging, network, cyber security, VC, Webcast data center.  
  
**1.2About the Project:**

**Project Title: UNI ALUMNI WEB PORTAL**

An Online Alumni Management System helps institutions to strategically build and maintain their alumni network, by facilitating engagement, community-building, networking, communications amongst many more functionalities. And thus a secondary array of positive marketing takes place. With an Online Alumni Management System, the alumni data can be centralized Background to the Problem

The greatest asset any Institution can have is the Alumni System. Alumni are the people who represent the Institution in the real world. University Alumni Systems exist to promote active and ongoing relationships between graduates and their alma mater. However, it is a big challenge on how to enhance mentoring between alumni and current students. Interaction between the University and students starts from the time student enrolls into the University as a freshman over the period in which the student lives on campus. Cooperation between University and students has occurred since admission until graduation. When students graduate and then begin their work, the only thing that can make an engagement to alumni is the Alumni Association. Alumni Association will serve as a bridge between the University and Alumni.

The University means a lot to the alumni. In many ways, a University is a place that alumni

can relate to with nostalgic stories. It is where they were introduced to the “adult world”

with knowledge, guidance, and help. Many alumni are thankful for this stepping stone that

the University was for them. This relationship with the alma mater begins with each person’s “sense of belonging”, hence individuals want to be a member of a community. A community, like a University and an Alumni Association, is a network of individuals that share values, norms and codes of conduct.

**Chapter 2**

**System Analysis**

**SYSTEM ANALYSIS**

System analysis is the first stage according to the System Development Life Cycle model. This system analysis is a process that starts with the analyst. Analysis is a detailed study of the various operations performed by a system and their relationships within and outside of the system. One aspect of analysis is defining the boundaries of the system and determining whether a candidate system should consider other related systems. During analysis, data are collected on the available files, decision points, and transaction handled by present system. Logical system models and tools that are used in analysis. Training, experience and common sense are required for collection of the information needed to do the analysis.

### ****Proposed System****

The proposed **University Alumni Web Portal** is a dynamic, secure, and user-centric online platform designed to strengthen the relationship between a university and its alumni. This system aims to streamline alumni engagement, facilitate student-alumni interaction, and provide a structured environment for communication, collaboration, and career development opportunities.

At its core, the proposed system provides a **centralized platform** where alumni can register, create personalized profiles, and stay connected with their alma mater. The system offers various interactive features, such as a blog post section, messaging tools, and internship postings, enabling alumni to actively participate in university life even after graduation.

One of the key features of the system is the **blog post submission module**, where alumni can share their professional experiences, offer guidance to current students, or write about any topic of interest. To maintain content quality and relevance, all posts are reviewed and approved by an administrator before being published. This not only encourages knowledge sharing but also creates an informative digital library for both alumni and students.

The system also includes an **admin dashboard**, which gives the university full control over alumni data management. Admins can view, edit, and verify alumni profiles, manage submitted blog content, and oversee overall user activity. This ensures that the platform remains secure, reliable, and free from spam or inappropriate content.

Alumni are empowered to **update their own profiles** with professional achievements, current job positions, contact information, and areas of expertise. This feature ensures data accuracy and allows other users to view meaningful, up-to-date information for networking or collaboration purposes.

The proposed system provides **real-time connectivity** between current students and alumni. Students can view alumni profiles, send messages, or request mentorship or internship opportunities. This direct link to experienced professionals enhances student career development and helps them gain valuable industry insights.

In addition to personal and professional networking, the portal supports the **posting of internships and job opportunities** by alumni who are in a position to hire or refer students. This feature creates a career-supportive ecosystem within the university community.

To ensure accessibility and usability, the portal is built with a **responsive design**, making it fully functional on desktops, tablets, and smartphones. Modern technologies such as Angular (frontend), Node.js (backend), and MySQL (database) are used to build a secure, scalable, and efficient system.

Security is a top priority in the proposed system. Proper **authentication and authorization mechanisms** are in place to protect user data and ensure that only verified alumni and students gain access to the portal. Admin approval processes add another layer of control over content and user activity.

The proposed system not only promotes communication between alumni and the university but also serves as a **valuable tracking tool**. University administrators can generate reports, analyze alumni career growth, and maintain long-term relationships with graduates. These insights can be used for academic improvement, fundraising, accreditation processes, and public relations.

### ****Advantages of the System****

1. **Strengthens Alumni-University Relationships**  
   The portal helps maintain a continuous connection between alumni and the university, fostering a sense of belonging even after graduation.
2. **Centralized Alumni Data Management**  
   The university can manage all alumni data in a structured and centralized database, making it easier to track alumni achievements, employment status, and contact details.
3. **Efficient Communication**  
   Both alumni and administrators can communicate effectively through the portal. Admins can send announcements, updates, or event invitations, while alumni can reach out to peers or students.
4. **Blog Post Sharing**  
   Alumni can contribute valuable content through blog posts, sharing knowledge, experiences, and advice that benefit current students and the larger community.
5. **Career Support and Opportunities**  
   Alumni can post internship or job openings, offering direct employment opportunities to students and recent graduates. This creates a career-focused ecosystem within the university.
6. **Mentorship and Networking**  
   Students can connect with experienced alumni for mentorship, career guidance, or professional networking, enhancing their readiness for the job market.
7. **Profile Customization**  
   Alumni can update their own profiles with career progress, achievements, and skills. This not only ensures data accuracy but also helps build professional credibility among peers and students.
8. **Admin Control and Content Moderation**  
   Admins can approve blog posts, verify alumni registrations, and manage portal content, ensuring quality control and data integrity.
9. **Responsive and User-Friendly Interface**  
   The portal is designed to work seamlessly across devices—desktops, tablets, and smart phones—providing a smooth and consistent user experience.
10. **Secure and Role-Based Access**  
    The system includes secure login features and role-based access control (admin and alumni), protecting sensitive data and preventing unauthorized access.
11. **Event and Campaign Support (Future Scope)**  
    The system can support the organization of alumni reunions, webinars, and fundraising campaigns, helping the university stay connected with graduates at scale.
12. **Institutional Benefits**  
    The university can use alumni data for accreditation, rankings, PR, and funding opportunities by showcasing alumni success stories and building a strong community network.
13. **Cost and Time Efficient**  
    Automating alumni engagement and data management reduces the manual effort and administrative costs associated with traditional communication methods (e.g., phone calls, emails, postal letters).
14. **Long-Term Engagement**  
    The portal helps keep alumni engaged over the long term, encouraging them to give back to the university in the form of time, knowledge, or resources.

**Chapter 3**

**System Planning**

**SYSTEM PLANNING**

System planning is a critical phase in the development of the University Alumni Web Portal. It involves identifying the project goals, determining system requirements, defining the scope, and preparing for implementation. Proper planning ensures that the system meets the needs of the university, alumni, and students while remaining scalable and secure.

#### ****3.1. Objectives of the System****

* To create a centralized platform for managing alumni data.
* To facilitate communication between alumni, students, and university administrators.
* To allow alumni to share professional experiences and job opportunities.
* To provide a space for blogging, mentorship, and community engagement.
* To enable secure and efficient data management and access control.

#### ****2. Scope of the System****

The portal will provide functionalities for:

* Alumni registration, login, and profile management.
* Admin-controlled data and content moderation.
* Alumni blog post submissions and approval workflow.
* Alumni-student interaction for mentorship and career support.
* Internship and job postings by alumni.
* Responsive design for cross-device compatibility.
* Future extensions like event management, donation modules, and real-time messaging.

#### ****3. Requirement Analysis****

**a) Functional Requirements:**

* User authentication (alumni and admin roles).
* Alumni registration and profile update.
* Admin dashboard for user and content management.
* Blog post creation, submission, and approval process.
* Alumni directory for student browsing.
* Internship/job posting by alumni.

**b) Non-Functional Requirements:**

* System security and data privacy.
* High performance and fast load times.
* Scalability to accommodate growing users.
* Responsive and user-friendly interface.
* Backup and data recovery capabilities.

#### ****4. System Architecture****

* **Frontend**: Angular with Material UI for dynamic user interfaces.
* **Backend**: Node.js with Express.js for RESTful APIs and business logic.
* **Database**: MySQL using Sequelize ORM for structured data storage.
* **Authentication**: JWT (JSON Web Tokens) for secure user sessions and role-based access.
* **File Storage**: Server-side handling of image or document uploads (e.g., alumni profile pictures or blog images).

#### ****5. Module Planning****

| **Module Name** | **Description** |
| --- | --- |
| User Registration & Login | Allows alumni to register and log in securely. |
| Alumni Profile | Enables users to add and update personal and professional details. |
| Blog Post Management | Alumni can write posts; admins can approve or reject them. |
| Admin Dashboard | Admin can manage users, approve content, and monitor system activities. |
| Alumni Directory | A searchable listing of alumni, useful for networking and mentorship. |
| Internship/Job Posting | Alumni can share career opportunities for students and recent graduates. |
| Notification System | Sends alerts for post approvals, messages, or event announcements. |

#### ****6. Development Plan****

* **Phase 1: Requirement Gathering & UI Design**  
  Research user needs, define core features, and create mockups.
* **Phase 2: Backend and Database Development**  
  Set up server, API routes, database schemas, and user authentication.
* **Phase 3: Frontend Development**  
  Build responsive user interfaces for alumni and admin panels.
* **Phase 4: Integration & Testing**  
  Connect frontend and backend, test all features, fix bugs, and ensure performance.
* **Phase 5: Deployment**  
  Launch the portal on a secure server with live database and domain integration.
* **Phase 6: Maintenance & Future Enhancements**  
  Monitor system performance, fix issues, and roll out new features like real-time chat and event management.

#### ****7. Security and Privacy Considerations****

* Passwords are hashed using secure algorithms (e.g., bcrypt).
* Role-based access ensures users can only access appropriate features.
* Data is validated at both client and server ends to prevent injection attacks.
* Secure file upload mechanisms prevent malicious content.

#### ****8. Resource Planning****

* **Team**: Full-stack developer(s), UI/UX designer, tester, and project manager.
* **Tools**: VS Code, GitHub, MySQL Workbench, Postman, Figma (for design).
* **Hosting**: Cloud platform (e.g., Vercel for frontend, Heroku or AWS for backend).
* **Timeframe**: Estimated development time is 6–8 weeks based on features.

#### ****9. Risk Management****

* **Data loss or breach**: Use regular backups and data encryption.
* **User inactivity**: Send email notifications or alerts to increase engagement.
* **Scalability issues**: Use modular architecture to add features easily in the future.

**3.2 PERT Chart** – University Alumni Web Portal

| **Task ID** | **Task Description** | **Predecessor(s)** | **Estimated Duration (Days)** |
| --- | --- | --- | --- |
| A | Requirement Gathering & Planning | – | 3 |
| B | UI/UX Design (Wireframes, Mockups) | A | 4 |
| C | Database Design (Schema & Relationships) | A | 3 |
| D | Backend Setup (Node.js, API Routing) | C | 5 |
| E | Frontend Setup (Angular Project) | B | 3 |
| F | Authentication System (JWT, bcrypt) | D | 3 |
| G | Alumni Profile Module | D, E | 4 |
| H | Admin Dashboard | D, E | 4 |
| I | Blog Post Module | D, E | 3 |
| J | Internship/Job Posting Module | G | 3 |
| K | Integration & Testing | F, G, H, I, J | 5 |
| L | Deployment | K | 2 |
| M | User Training & Documentation | K | 2 |
| N | Final Review & Launch | L, M | 1 |

### ****Critical Path****

A → C → D → F → G → J → K → L → N  
**Total Duration:** 29 Days (approx.)

**3.3 Cost and Benefit Analysis:** For “**UNI ALUMNI WEB PORTAL**”, Cost/benefit analysis gives a picture of the various costs, benefits, and rules associated with each alternative system. In developing cost estimates for a system, we need to consider several cost elements. Among them are hardware, personal, facilities, operating. Hardware cost relates to the actual purchase or lease of the computer and peripherals. Costs incurred during the development of a system are one-time costs and are labelled developmental costs. Once the system is installed the cost of operating and maintaining the system become a recurring cost.

Facility costs are expenses incurred in the preparation of the physical site where the application or the computer will be in operation.

Operating costs include costs associated with the day-to-day operation of the system. One approach is to treat operating costs like overhead. Another approach is to charge each authorized user for the amount of processing the request from the system.

A system is also expected to provide benefits. The first task is to identify each benefit and turns to assign a monetary value to it for cost/benefit analysis. Benefits may be tangible and intangible, direct or indirect.

**Terms:**

**1. Tangible or Intangible Costs**

Tangibility refers to the ease with which costs or benefits can be measured. An outlay of cash for a specific item or activity is referred to as a tangible cost. The purchase of hardware or software personnel training and employee salaries are examples of tangible costs. They are readily identified and measured. Costs that are known to exist but whose financial value cannot be accurately measured are referred to as intangible costs. For example, employee movable problems caused by a new system or lowered company image are an intangible cost.

**Benefits**

Tangible benefits such as completing jobs in fewer hours or producing reports with no errors are quantifiable. Intangible benefits such as more satisfied customers or an improved corporate image are not easily qualified. Both tangible and intangible costs and benefits, however, should be considered in the evaluation process.

**2. Direct or indirect costs**

Direct costs are those with which a rupee’s figure can be directly associated in a project. They are applied directly to the operation. For example, the purchase of a box of diskettes for Rs.2800 is a direct cost because the diskettes can be associated with the rupees expanded. Indirect costs are the results of operations that are not directly associated with a given system or activity. They are often referred to as overhead. A system that reduces overhead realizes savings.

**Benefits**

Direct benefits also can be specifically attributable to a given project. For example, a new system than can handle 25 percent no transaction per day is a direct benefit. Indirect benefits are realized as a by-product of another activity or system. For example, a proposed safe deposit billing system that provides profits showing vacant boxes by sizes, location, and price, will help management decide on how much advertising to do for box rental.

**3.Fixed or Variable Costs**

Fixed costs are sunk costs. They are straight-line depreciation of hardware, exempt employee salaries and insurance. In contrast, variable costs are incurred on a regular basis. They are usually proportional to work volume and continue as long as the system is in operation. For example, the costs of computer forms vary in proportion to the amount of processing or the length of the reports required.

**Benefits**

Fixed benefits are also constant and do not change. For example, decrease in the number of personnel by 20 percent resulting from the use of a new computer. In this project, cost is incurred in terms of time consumed, electricity used etc.

**Costs and benefits are as follows:**

**Cost**

**Hardware Cost:** Single computer system is used in the development of an application. All hardware parts are working well and its quality is perfect**.**

**Personnel Cost:** Staff is not required. An administrator will be responsible for maintaining the system and its records. So, the developer is responsible for an extra cost.

**Facility Cost:** Electricity is being consumed in developing this application.

**Benefits**

➢ This application leads to less time-consuming.

➢ Our database can store a large amount of data can be stored serially and accessed frequently.

➢ Staff reduction-only single person is required which will act as an administrator to maintain the system

**Chapter 4**

**Feasibility Study**

**Feasibility Study**

Feasibility study is an important phase in the software development process. It enables the developer to have an assessment of the product being developed. It refers to the feasibility study of the product in terms of outcomes of the product, operational use and technical support required for implementing it.

Once the need of system requirement has been defined. The user may authorize the system analyst to create a more detailed for better understanding the opportunities and limitation associated with proposed project. Feasibility study guides the user to determine the process of the project. Feasibility study also guide user to determine the risk associated with project. The feasibility study includes Technical, Economic and Operational, the result of these study is combined known as Feasibility Study.

**4.1 Technical Feasibility:** We can strongly say that it is technically feasible, since there will not be much difficulty in getting required resources for the development and maintaining the system as well. All the resources needed for the development of the software as well as the maintenance of the same is available in the organization here we are utilizing the resources which are available already.

**The technical issues raised during the feasibility stage of the investigation are:**

**1.** Does the necessary technology exist (can it be acquired) to do what is suggested?

**2.** Does the proposed equipment have the technical capacity to hold the data required touse the new system?

**3.** Will the proposed system and components provide adequate responses to queries, regardless of the number or location of users?

**4.** Can the system be expanded, if developed?

**5**. Are there technical guarantees of accuracy, reliability, ease of access and data security?

**4.2 Economic Feasibility:** Economic feasibility is the most frequently used methods for evaluating the effectiveness of a candidate system. More commonly known as cost/benefit analysis, the procedure is to determine the benefits and savings that are expected from a candidate system and compare them with cost. The result of comparison is found and changed

if needed. If benefits outweigh costs, then the decision is made to design and implement the system. Otherwise, further justification or alternation in the proposed system will have to be made if it is to have a chance of being approved. As we are developing a completely new system the cost is on the higher side. The implementation costs involve the installation of a new hardware and software as well as the cost of hosting the website on the internet. Maintenance of the system is much costly. Training for the operating personnel is also expected to be by the people who have never been initialized to operating a computerized system.

In performing cost and benefit analysis it is important to identify cost and benefits factors. Cost and benefits can be categorized into the following categories:

1) Development cost: A Development cost is the costs that are incurred during the development of the system. It is one time investment.

2) Operating cost: Operating cost are the expenses required for the day to-day running of the system. As, operating cost are wages, supplies and overheads.

3) Hardware/Software cost: It includes the cost of purchasing or leasing of computers and its peripherals. Software costs involve required software cost.

4) Supply cost: These are variable costs that are very proportionately with the amount of use of paper, ribbons, disks, and others.

**Benefits:**

1. Fast and easy access to all procedures and functions.

2. No need for large storage spaces sized of rooms for storing the cabinets because all the information about the members and other details is saved in the computer’s hard disks.

3. High level of security and authentication of each and every user.

4. Reliability is increased, as backups of files, and records can be made and saved.

5. Different locations and information will be highly secure, unlike in file cabinets where entries can easily be ripped or tampered with by users.

**4.3 Operational Feasibility:** The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system. It is mainly related to human organizational and political aspects. The points to be considered are:

1. What changes will be brought with the system?

2. What new skills will be required?

3. Do the existing staff members have these skills?

People are inherently resistant to change and computers have been known to facilitate change. Now most people support computerized system. An estimate should be made of how strong a reaction the user staff is likely to have toward the development of a new system. Therefore, it is understandable that the introduction of the new system required special effort to educate and train the staff on way of operating system. Also required to give awareness to the customers. The staffs were not against the system; however the user would accept the concept.

**Chapter 5**

**System Requirement Specification**

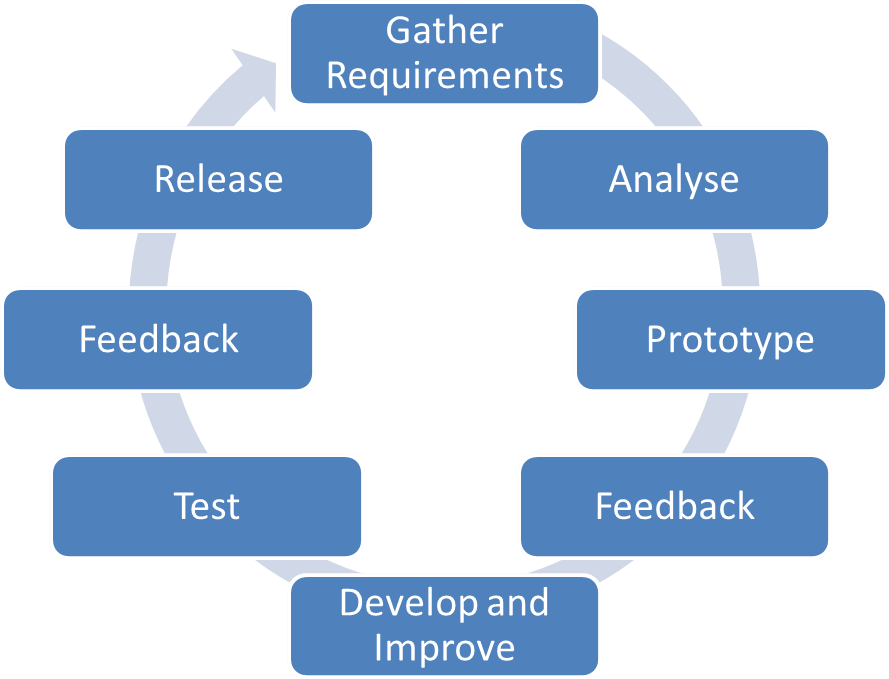
**System Requirement Specifications**

The requirement phase basically consists of three activities:

• Requirement Analysis

• Requirement Specification

• Requirement Validation



**SRSpresentedin**

**thisphase.**

## 5.1.1 User Requirements:User Requirements define the needs and expectations of the different types of users who will interact with the system. These requirements help guide the design and development of a user-centric platform that serves its intended purpose effectively.

* In the University Alumni Web Portal, there are three primary types of users:
* **Alumni (Registered Users)**
* **Students (View-only Access)**
* **Admin (University Staff or Portal Manager)**

### ****1. Alumni User Requirements****

* Alumni are the main contributors and participants in the system. Their requirements include:
* Ability to **register** and create a personal profile.
* Secure **login and logout** functionality.
* Option to **update their profile** details such as name, email, phone number, current employment, graduation year, etc.
* Ability to **upload a profile picture**.
* Feature to **submit blog posts**, including titles, images, and content.
* Option to **view the status** of submitted blog posts (approved/rejected/pending).
* Ability to **browse other alumni profiles** for networking purposes.
* Function to **post internship or job opportunities**.
* Option to **search or filter alumni profiles** based on name, batch, profession, etc.
* View and respond to **messages or contact requests** (future enhancement).
* Receive **notifications or updates** regarding blog post approval, messages, or events.

### ****2. Student User Requirements****

* Students can access selected parts of the system to connect with alumni and benefit from shared resources.
* Ability to **view alumni directory** and profiles (read-only).
* Option to **search alumni** by graduation year, company, or job title.
* Access to **view approved blog posts** written by alumni.
* Ability to **see internship/job opportunities** posted by alumni.
* Contact alumni through a **message or request feature** (if allowed).

X No access to create content or update profiles.

### ****3. Admin User Requirements****

* Admin users manage the entire system, including user access, content moderation, and platform maintenance.
* Secure **admin login panel** with role-based access control.
* Ability to **view and manage all alumni profiles** (edit, deactivate, delete).
* Option to **approve, reject, or edit submitted blog posts** before publishing.
* View **dashboard analytics**, such as active alumni, number of blog posts, and active job/internship listings.
* Ability to **manage job and internship postings**.
* Option to **send announcements** or notifications to alumni.
* Control over **content visibility**, such as featured posts or highlighted alumni.
* Manage system settings such as logo, contact details, and site-wide configuration.
* Ability to **export data** or generate reports for university use.
* Monitor and **respond to support queries** from alumni or students.

### ****4. System-Wide Requirements (Common to All Users)****

* Mobile-friendly and responsive design for access on all devices.
* Fast-loading pages and user-friendly navigation.
* Secure authentication using encrypted passwords (e.g., bcrypt).
* Form validations to prevent incorrect or incomplete submissions.
* Role-based permissions to ensure users access only what is intended for them.
* Reliable error handling and feedback messages for actions taken.
* Accessibility for users with disabilities (future enhancement).

**5.1.2 Functional Requirements:** In software engineering and systems engineering, a functional requirement defines a function of a system or its component, where a function is described as a specification of behaviour between outputs and inputs.

Functional requirements may involve calculations, technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish. Behavioural requirements describing all the cases where the system uses the functional requirements are captured in use cases. Functional requirements are supported by non- functional requirements (also known as "quality requirements"), which impose constraints on the design or implementation (such as performance requirements, security, or reliability). Generally, functional requirements are expressed in the form "system must do <requirement>," while non- functional requirements take the form "system shall be <requirement>."The plan for implementing functional requirements is detailed in the system design, whereas nonfunctional requirements are detailed in the system architecture.

As defined in requirements engineering, functional requirements specify particular results of a system. This should be contrasted with non-functional requirements, which specify overall characteristics such as cost and reliability. Functional requirements drive the application architecture of a system, while non-functional requirements drive the technical architecture of a system.

**5.1.3 Non-Functional Requirements:** In systems engineering and requirements engineering, a non-functional requirement (NFR) is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviours. They are contrasted with functional requirements that define specific behaviour or functions. The plan for implementing functional requirements is detailed in the system design. The plan for implementing non-functional requirements is detailed in the system architecture, because they are usually architecturally significant requirements.

Non-functional requirements are often called "quality attributes" of a system. Other terms for non-functional requirements are "qualities", "quality goals", "quality of service requirements", "constraints", "non-behavioural requirements", or "technical requirements". Informally these are sometimes called the "ileitis", from attributes like stability and portability. Qualities—that is non- functional requirements—can be divided into two main categories:

**Execution Qualities:** such as safety, security and usability, which are observable during operation (at run time).

**Evolution Qualities:** such as testability, maintainability, extensibility and scalability, which are embodied in the static structure of the system.

**5.2 Module Description:** The University Alumni Web Portal consists of multiple interconnected modules designed to manage alumni data, enable communication, and provide valuable resources to alumni, students, and university administrators. Below is a detailed description of each core module in the system.

### ****1. User Authentication Module****

* **Purpose**:  
  To manage secure login, registration, and access control for different user roles (Alumni and Admin).
* **Features**:
* User registration (for alumni only)
* Secure login/logout using JWT tokens
* Password encryption (e.g., bcrypt)
* Role-based access (Alumni, Admin)
* Forgot password and reset functionality

### ****2. Alumni Profile Management Module****

* **Purpose**:  
  To allow alumni users to create, view, and update their personal and professional profiles.
* **Features**:
* View and edit profile information (name, graduation year, profession, etc.)
* Upload and update profile photo
* Display alumni profiles to other users
* Filter alumni directory by batch, company, or industry

### ****3. Admin Dashboard Module****

* **Purpose**:  
  To give administrators control over the entire portal, including user and content management.
* **Features**:
* View list of all alumni users
* Approve, edit, or deactivate user accounts
* Review submitted blog posts and approve/reject them
* Monitor platform activity (dashboard with key metrics)
* Manage internship/job postings
* Send announcements to users

### ****4. Blog Post Module****

* **Purpose**:  
  To enable alumni to contribute blog posts and share professional stories, tips, or updates.
* **Features**:
* Create new blog post (title, content, image)
* Submit for admin approval
* View list of submitted and published posts
* Admin approval and editing interface
* Display approved blog posts to students and alumni

### ****5. Internship and Job Posting Module****

* **Purpose**:  
  To allow alumni to post internship and job opportunities for students and fellow alumni.
* **Features**:
* Add new opportunity (title, description, company, location)
* Manage posted opportunities (edit/delete)
* View all available internships/jobs
* Admin moderation and approval (optional)

### ****6. Alumni Directory Module****

* **Purpose**:  
  To provide a searchable list of all registered alumni for networking and collaboration.
* **Features**:
* Display alumni profiles publicly (except sensitive data)
* Search and filter by name, batch, department, job title, etc.
* Optional messaging feature for students to contact alumni (future enhancement)

### ****7. Notification Module****

* **Purpose**:  
  To keep users informed about important actions and updates on the platform.
* **Features**:
* Email or on-site notifications for blog post status, new job opportunities, etc.
* Admin announcements and alerts
* Notification center/dashboard for users

### ****8. Content Management Module****

* **Purpose**:  
  To allow admin users to manage site content including homepage text, banners, contact info, etc.
* **Features**:
* Edit static content (About Us, Contact Us pages)
* Manage featured blog posts or alumni profiles
* Update logos, headers, and social links

### ****9. System Configuration Module****

* **Purpose**:  
  To control portal settings, user permissions, and general configurations.
* **Features**:
* Set default roles and permissions
* Configure email services and security settings
* Manage backup schedules and data recovery options

**5.3 Minimum Hardware & Software Requirements:**

**5.3.1 Hardware Requirements: -**

**❖ Processor:** Inteli3/i5 or higher (U series or higher).

**❖ RAM:** Minimum 4GB or above.

❖ **HDD:** 40 to 100 GB (Recommended) or above.

**5.3.2 Software Requirements: -**

❖ **Operating System:** Windows 7 or above

**❖ For front-end:** Languages: HTML, CSS, JavaScript, TypeScript.

**Source-code editor:** Visual Studio Code.

**Framework:** Angular Version 18

**❖ For Back-end: Languages:** JavaScript. Run time

**Environment:** Node.js.

**Source-code editor:** Visual Studio Code.

❖ **For Database:**

**Database:** MariaDB.

**Management Tool:** Heidi SQL

**❖ Browser:** Google Chrome / Firefox / Microsoft Edge.

**5.3.3 Runtime Requirement: -**

❖ **Device:** Laptop/Desktop/Mobile phones.

❖ **Operating System:** Windows/Linux/Android

❖ **Processor:** 512 MB

**5.4 Technology Description**

**5.4.1 Angular Framework:** Angular is a popular open-source front-end web application framework developed and maintained by Google. It is written in TypeScript and allows developers to build dynamic, single-page web applications (SPAs) and progressive web apps (PWAs). Angular provides a structured and powerful framework for building client-side applications, and it follows the Model-View-Controller (MVC) architectural pattern**.**

**Key features and characteristics of the Angular framework include:**

**1. TypeScript:** Angular is built using TypeScript, which is a superset of JavaScript that adds static typing and other advanced features to JavaScript. TypeScript helps catch errors during development and provides better tooling support.

**2. Component-Based Architecture:** Angular applications are built using components. Components are self-contained and reusable building blocks that encapsulate the template, logic, and styles associated with a specific part of the user interface.

**3. Data Binding:** Angular provides powerful data binding capabilities that allow automatic synchronization of data between the component and the template, making it easier to manage the application state.

**4. Directives**: Angular uses directives to extend the functionality of HTML elements. Directives can be custom or built-in, and they allow developers to add behavior or manipulate the DOM elements.

**5. Dependency Injection:** Angular uses a robust dependency injection system, which promotes modularity and makes it easier to manage the application's dependencies and services.

**6. Routing:** Angular provides a powerful routing system that enables navigation within the application without requiring full page reloads.

**7. Forms:** Angular offers extensive support for creating forms, including templatedriven forms and reactive forms, making it easier to handle form inputs and validation

**8. Testing:** Angular has built-in support for unit testing and end-to-end testing, making it easier to ensure the quality and reliability of applications.

**5.4.2 Node.js:** Node.js is an open-source, cross-platform, server-side JavaScript runtime environment. It allows developers to execute JavaScript code on the server-side outside of a web browser. Node.js is built on the V8 JavaScript engine, which is developed by Google and used in their Chrome web browser. This engine compiles JavaScript code into machine code, providing high-performance execution.

Node.js was created by Ryan Dahl and was first released in 2009. Since then, it has gained significant popularity and has become a fundamental technology for building scalable and realtime applications

**Key features and characteristics of Node.js include:**

**1. Asynchronous and Non-Blocking I/O:** Node.js is designed with an eventdriven architecture, enabling asynchronous processing and non-blocking I/O operations. This makes it well-suited for handling concurrent connections and I/Ointensive tasks without getting blocked.

**2. Single-Threaded Event Loop:** Node.js uses a single-threaded event loop to handle multiple concurrent connections efficiently. It allows handling a large number of connections without the need for spawning new threads for each request, making it lightweight and memory-efficient.

**3. NPM (Node Package Manager):** Node.js comes with NPM, a powerful package manager that provides access to a vast ecosystem of open-source modules and libraries. NPM makes it easy for developers to install, manage, and share reusable code.

**4. Cross-Platform:** Node.js is compatible with various operating systems, including Windows, macOS, and Linux, making it highly portable and flexible.

**5. Server-Side Development:** Node.js is commonly used for server-side development to build web applications, APIs (Application Programming Interfaces), and backend services.

**6. Full-Stack Development:** With Node.js, developers can use the same language (JavaScript) for both client-side and server-side development, promoting code reuse and enhancing developer productivity

**5.4.3 MariaDB:** MariaDB is an open-source, relational database management system (RDBMS) and a popular alternative to MySQL. It was developed by the original creators of MySQL after MySQL was acquired by Oracle Corporation. MariaDB is designed to be a dropin replacement for MySQL, meaning it is mostly compatible with MySQL and can be used as a seamless replacement without requiring significant changes to existing applications or databases.

**Key features and characteristics of MariaDB include:**

**1. Open-Source:** MariaDB is released under the GNU General Public License (GPL), making it freely available for use, distribution, and modification.

**2. Performance:** MariaDB is known for its excellent performance, scalability, and optimization features. It has a robust query optimizer and supports various storage engines, including InnoDB, My Rocks, and Aria, among others.

**3. High Availability and Replication**: MariaDB supports features like MasterSlave replication and Master-Master replication, enabling data redundancy and high availability for critical applications.

**4. Security:** MariaDB emphasizes security and provides features like data encryption, secure connections using SSL/TLS, and user authentication mechanisms.

**5. Compatibility:** As mentioned earlier, MariaDB is designed to be compatible with MySQL. It supports the same SQL syntax, APIs, and connectors, making it easy to migrate from MySQL to MariaDB.

**6. Active Community:** MariaDB has a large and active community of developers and contributors, ensuring continuous development and improvement of the database system.

**7. Storage Engines**: MariaDB supports multiple storage engines, allowing users to choose the most suitable one for their specific use case. The default storage engine, InnoDB, is a transactional storage engine widely used for its ACID-compliance (Atomicity, Consistency, Isolation, and Durability).

**8. Tools and Ecosystem:** MariaDB comes with various tools and utilities for database management, administration, and performance tuning.

**5.4.4 JavaScript:** JavaScript is a lightweight, interpreted programming language. It is designed for creating network-centric applications. It is complimentary to and integrated with Java. JavaScript is very easy to implement because it is integrated with HTML. It is open and cross-platform.

**Key advantages of learning JavaScript:**

**1.** JavaScript is the most popular programming language in the world and that makes it a programmer’s great choice. Once you learnt JavaScript, it helps you developing great front-end as well as back-end software’s using different JavaScript based frameworks like jQuery, Node.JS etc.

**2.** JavaScript is everywhere, it comes installed on every modern web browser and so to learn JavaScript you really do not need any special environment setup. For example, Chrome, Mozilla Firefox, Safari and every browser you know as of today, supports JavaScript.

**3.** JavaScript helps you create really beautiful and crazy fast websites. You can develop your website with a console like look and feel and give your users the best Graphical User Experience.

**4.** JavaScript usage has now extended to mobile app development, desktop app development, and game development. This opens many opportunities for you as JavaScript Programmer.

**5.** Due to high demand, there is tons of job growth and high pay for those who know JavaScript. You can navigate over to different job sites to see what having JavaScript skills looks like in the job market.

**6.** Great thing about JavaScript is that you will find tons of frameworks and Library.

**5.4.5 jQuery:** jQuery is a cross-platform JavaScript library designed to simplify the clientside scripting of HTML. jQuery is the most popular JavaScript library in use today, with installation on 65% of the top 10 million highest-trafficked sites on the Web. jQuery is free, open-source software licensed under the MIT License.

**5.4.6 SCSS**

SCSS stands for "Sassy CSS," and it is a popular extension of CSS (Cascading Style Sheets). SCSS is a superset of CSS, which means that all valid CSS code is also valid SCSS code. It introduces additional features and syntax that make writing and organizing CSS styles more powerful and maintainable.

**5.4.7 Bootstrap**

Bootstrap is a free and open-source front end development framework for the creation of websites and web apps. The Bootstrap framework is built on HTML, CSS, and JavaScript (JS) to facilitate the development of responsive, mobile-first sites and apps. Responsive design makes it possible for a web page or app to detect the visitor’s screen size and orientation and automatically adapt the display accordingly; the mobile first approach assumes that smartphones, tablets and task- specific Mobile apps are employees' primary tools for getting work done and addresses the requirements of those technologies in design.

**5.4.8 HTML**

HTML is a markup language used for structuring and presenting content on the World Wide Web. It is the fifth and current version of the HTML standard.HTML5 includes detailed processing models to encourage more interoperable implementations; it extends, improves and rationalizes the markup available for documents, and introduces markup and application programming interfaces (APIs) for complex web applications. For the same reasons, HTML5 is also a candidate for cross-platform mobile applications, because it includes features designed with low-powered devices in mind.

**Chapter 6**

**System Design**

**System Design**

System design is the phase that bridges the gap between problem domain and the existing system in a manageable way. This phase focuses on the solution domain, i.e. “how to implement?”

It is the phase where the SRS document is converted into a format that can be implemented and decides how the system will operate.

In this phase, the complex activity of system development is divided into several smaller sub- activities, which coordinate with each other to achieve the main objective of system development

**6.1 PROTOTYPE MODEL:** The concepts of software engineering have been implemented successfully and uniformly throughout the system. The performance of the integrated system will be uniform. For building this project, we followed Prototype Model as the requirements of this project are completed analyzed at the beginning of the project itself. The prototyping model is a systems development method (SDM) in which a prototype (an early approximation of a final system or product) is built, tested, and then reworked as necessary until an acceptable prototype is finally achieved.

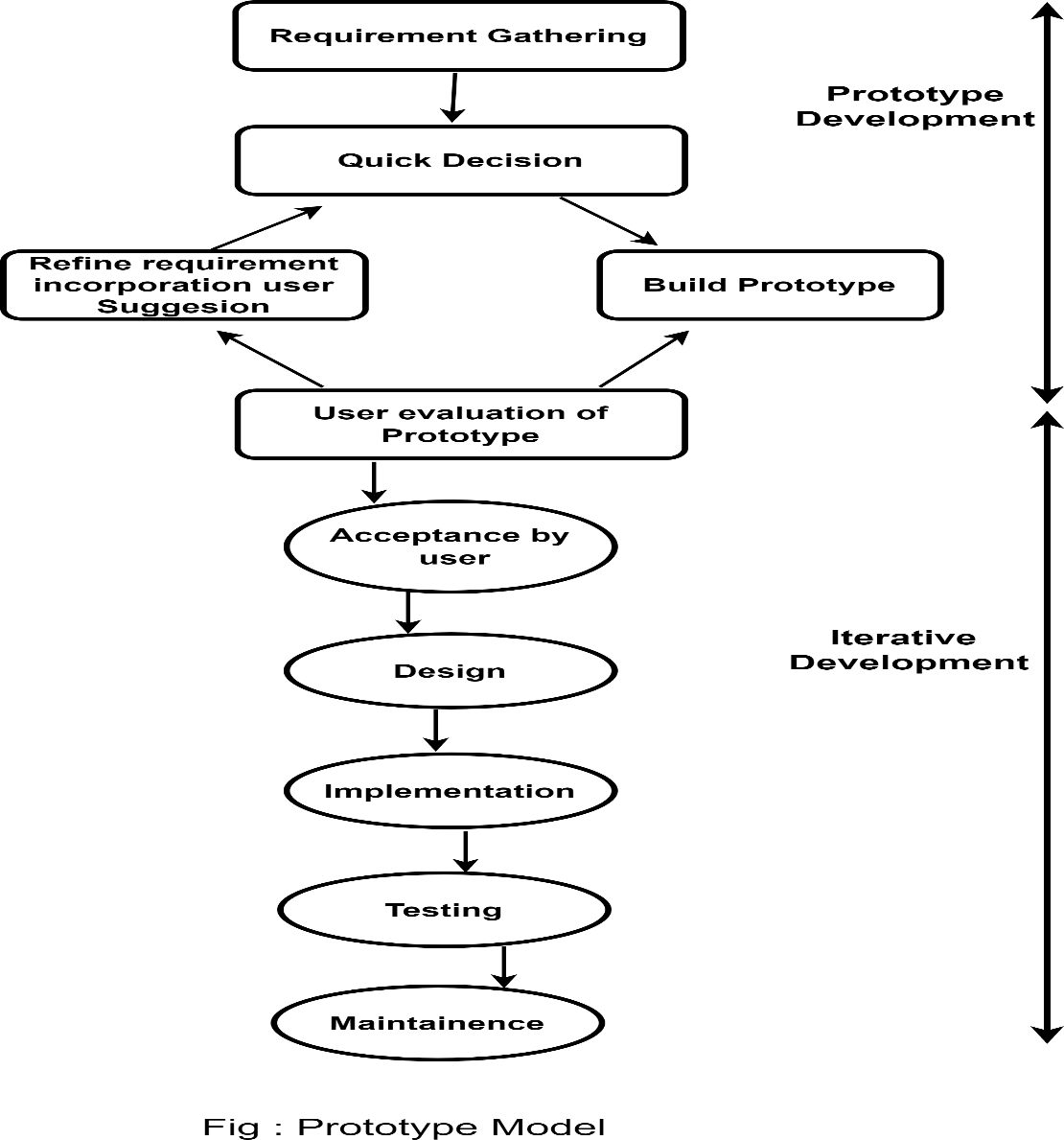
**There are several steps in the Prototyping Model:**

**1.** The new system requirements are defined in as much detail as possible. This usually involves interviewing a number of users representing all the departments or aspects of the existing system.

**2.** A preliminary design is created for the new system.

**3.** A first prototype of the new system is constructed from the preliminary design. This is usually a scaled-down system, and represents an approximation of the characteristics of the final product, change approval process. For developing my project” Project Name”, firstly I gather some requirement of this project, then with this initial requirement I have start the project designing and a preliminary design of project is construct.

A first prototype of the new system is constructed from the preliminary design. This is usually a scaled-down system, and represents an approximation of the characteristics of the final product. After making this prototype model and I present this model to the Project Name has.

Evaluate this first prototype, noting its strengths and weaknesses, what needs to be added, and what should to be removed. I collects and analyzes the remarks from the users. Then I modified the first prototype based on the comments supplied by the Project Name, and a second prototype of the new system is constructed. The second prototype is evaluated in the same manner as was the first prototype by Project Name.

**Advantages:**

1. Estimates (i.e. Budget, schedule, etc.) become more realistic work progresses, because important issues are discovered.

2. It is more able to cope with the software development generally entails.

Software engineers (who can get restless with protected design processes) can get their hands in and start working on a project earlier.

**6.2 Database Design:** A database is an organized mechanism that has the capability of storing information through which a user can retrieve stored information in an effective and efficient manner. The data is the purpose of any database and must be protected. The database design is a two-level process. In the first step, user requirements are gathered together and a database is designed which will meet these requirements as clearly as possible. This step is called Information Level Design and it is taken independent of any individual DBMS. In the second step, this Information level design is transferred into a design for the specific DBMS that will be used to implement the system in question. This step is called Physical Level Design, concerned with the characteristics of the specific DBMS that will be used. A database design runs parallel with the system design. The organization of the data in the database is aimed to achieve the following two major objectives.

• **Data Integrity**

**• Data independence**

**NORMALIZATION**

It is a process of converting a relation to a standard form. The process is used to handle the problems that can arise due to data redundancy i.e. repetition of data in the database, maintain data integrity as well as handling problems that can arise due to insertion, updating, deletion anomalies. Decomposing is the process of splitting relations into multiple relations to eliminate anomalies and maintain anomalies and maintain data integrity. To do this we use normal forms or rules for structuring relation. Insertion anomaly: Inability to add data to the database due to absence of other data. Deletion anomaly: Unintended loss of data due to deletion of other data. Update anomaly: Data inconsistency resulting from data redundancy and partial update Normal Forms: These are the rules for structuring relations that eliminate anomalies.

**FIRST NORMAL FORM:**

A relation is said to be in first normal form if the values in the relation are atomic for every attribute in the relation. By this we mean simply that no attribute value can be a set of values or, as it is sometimes expressed, a repeating group.

**SECOND NORMAL FORM:**

A relation is said to be in second Normal form is it is in first normal form and it should satisfy any one of the following rules. • Primary key is a not a composite primary key. • No non key attributes are present. • Every non key attribute is fully functionally dependent on full set of primary key.

**THIRD NORMAL FORM:**

A relation is said to be in third normal form if their exits no transitive dependencies. Transitive Dependency: If two non-key attributes depend on each other as well as on the primary key then they are said to be transitively dependent. The above normalization principles were applied to decompose the data in multiple tables thereby making the data to be maintained in a consistent state.

**DATA DICTIONARY:** A data dictionary contains:

• The definitions of all schema objects in the database (tables, views, indexes, clusters, synonyms, sequences, procedures, functions, packages, triggers, and so on)

• How much space has been allocated for, and is currently used by, the schema objects

• Default values for columns.

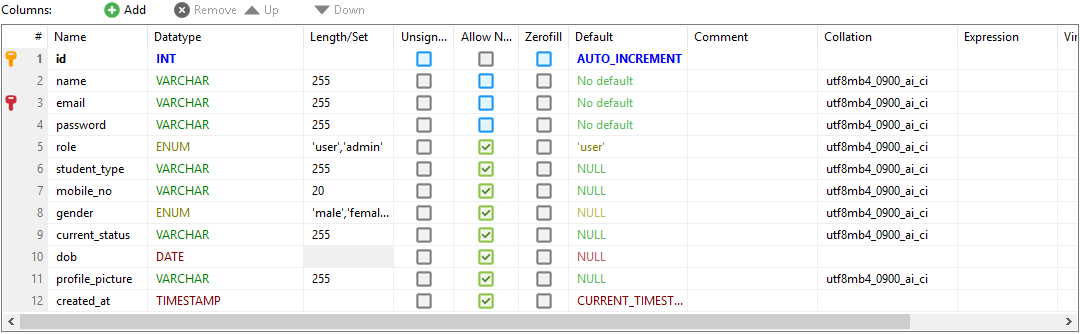
• Integrity constraint information.

• Privileges and roles each user has been granted.

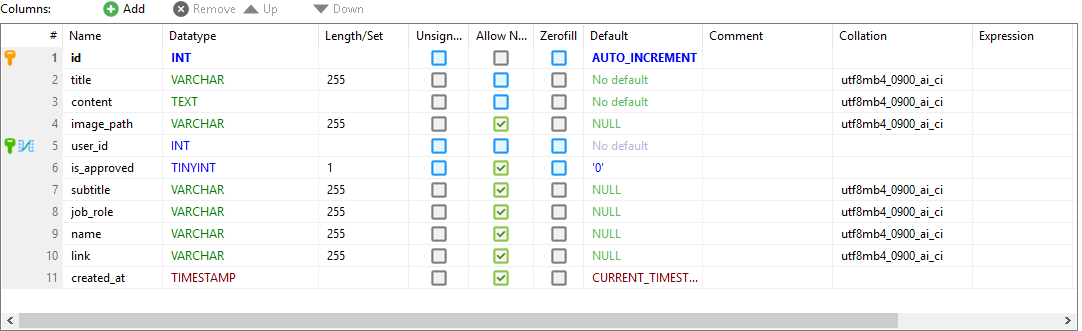
• Auditing information, such as who has accessed or updated various schema objects.

**Tables Structure:**

**Login and Registration Table**

****

**Blog Post Table**

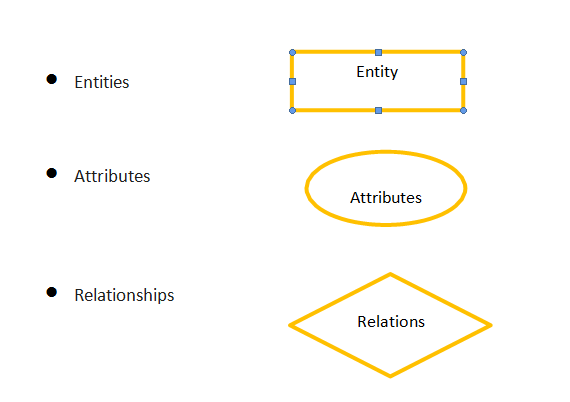
****

**6.3 E-R Diagram:** In software engineering, an entity-relationship model (ERM) is an abstract and conceptual representation of data. Entity-relationship modeling is a database modelling method, used to produce a type of conceptual schema or semantic data model of a system, often a relational database, and its requirements in a top-down fashion. Diagrams created by this process are called entity-relationship diagrams, ER diagrams, or ERDs.

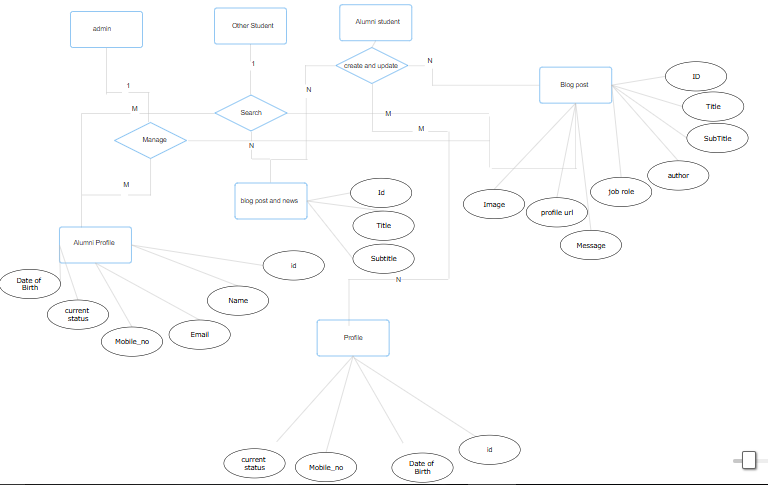
The first stage of information system design uses these models during the requirements analysis to describe information needs or the type of information that is to be stored in a database. The data modeling technique can be used to describe any ontology (i.e. an overview and classifications of used terms and their relationships) for a certain area of interest. In the case of the design of an information system that is based on a database, the conceptual data model is, at a later stage (usually called logical design), mapped to a logical data model, such as the relational model; this in turn is mapped to a physical model during physical design.

**Components of the ER Diagram**

This model is based on three basic concepts:



**ER Diagram**



**6.5 Data Flow Diagrams:** A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It can be manual, automated, or a combination of both. It shows how data enters and leaves the system, what changes the information, and where data is stored. The objective of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communication tool between a system analyst and any person who plays a part in the order that acts as a starting point for redesigning a system. The DFD is also called as a data flow graph or bubble chart.

Standard symbols for DFDs are derived from the electric circuit diagram analysis and are shown

|  |  |
| --- | --- |
| Symbol | Description |
|  | DataFlow–Dataflowarepipelinesthroughthepacketsofinformation flow. |
|  | Process:AProcessortaskperformedbythesystem. |
|  | Entity:Entityareobjectofthesystem.Asourceordestinationdataof a system. |
|  | DataStore:Aplacewheredatatobestored. |

**Circle:** A circle (bubble) shows a process that transforms data inputs into data outputs. **Data Flow:** A curved line shows the flow of data into or out of a process or data store. **Data Store:** A set of parallel lines shows a place for the collection of data items. A data store indicates that the data is stored which can be used at a later stage or by the other processes in a different order. The data store can have an element or group of elements. **Source or Sink:** Source or Sink is an external entity and acts as a source of system inputs or sink of system outputs.

**6.5.1 Context Level DFD (Level 0):** This diagram shows that UNI ALUMNI WEB PORTAL consists of two actors are ALUMNI and ADMIN

Each actor requests login action to enter the system and does their tasks as shown

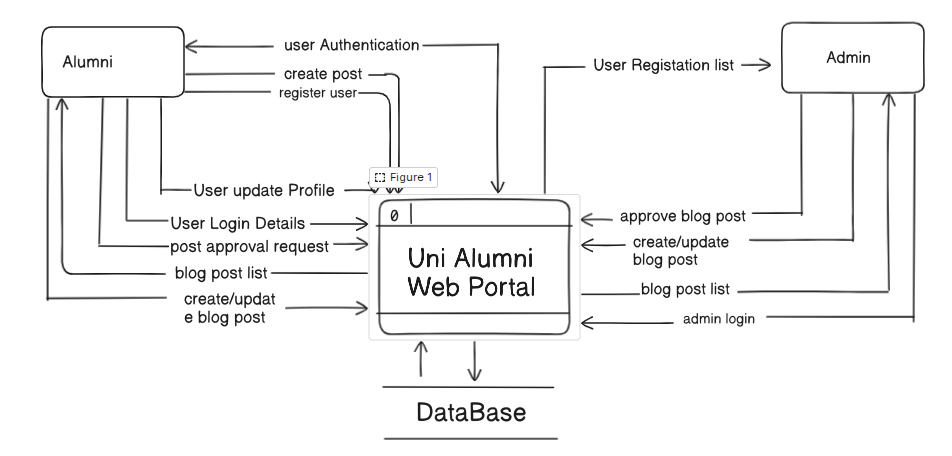


Figure (6.5.1) zero level DFD Uni Alumni web Portal

**6.5.2 Data Flow Chart**

Manage user

Create Event

Approve blog post

start

End

Admin

View blog posts & news

Get user

Alumni

Have

Account

No yes

yes

Update Profile

Create blog post

**Chapter 7**

**System Testing**

**System Testing**

System Testing is the testing of a complete and fully integrated software product. Usually, the software is only one element of a larger computer-based system. Ultimately, the software is interfaced with other software/hardware systems. System Testing is actually a series of different tests whose sole purpose is to exercise the full computer-based system. System testing of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements.

Testing is the process of detecting errors. Testing performs a very critical role for quality assurance and for ensuring the reliability of software and application.

**White Box Testing:** To follow the concept of white box testing we have tested each form we have created independently to verify that Data flow is correct. All conditions are exercised to check their validity. All loops are executed on their boundaries**.**

**Black Box Testing:** The Testing Method focuses on the functional Requirements of the software. Here each module will be treated as a black box that will take some input and generate output. Output for a given set of Input combinations is forwarded to other modules**.**

**Unit Testing:** Each module has been tested by giving different sets of inputs. The inputs are validated when accepting from the user.

**Validation Testing:** In the present system, validations are been written for Reference, Pay Type etc., entries and proper error messages are displayed when any validation error occurs. Validations such as a Text field should accept only Character data but no any other Characters and only Number data but no any other Characters.

**Testing Strategies:**

Testing is a set of activities that can be planned in advanced and conducted systematically. A strategy for software testing must accommodation low-level tests that are necessary to verify that a small source code segment has been correctly implemented as well as highlevel tests that validate major system functions against customer requirements. Software testing is one element of verification and validation. Verification refers to the set of activities that ensure that software correctly implements as specific function. Validation refers to a different set of activities that ensure that the software that has been built is traceable to customer requirements.

The main objective of software testing is to uncover errors. To fulfil this objective, a series of test steps unit, integration, and system tests are planned and executed. Each test step is accomplished through a series of systematic test technique that assist in the design of test cases. With each testing step, the level of abstraction with which software is considered is broadened.

**UNIT TESTING:**

Unit testing is usually performed by the developer who writes different code units that could be related or unrelated to achieve a particular functionality. Here individual units/components of a software/system are tested. The purpose is to validate that each unit of the software performs as designed.

The unit-testing we have is white box oriented and some modules steps are conducted in parallel.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test No.** | **Test Cases** | **Input Value** | **Expected Output** | **Pass/Fail** |
| 1 | Login with wrong User name and Password | **Username:**User1  **Password:**User1@123 | Incorrect User name or Password | Pass |
| 2 | Login with Right User name and Password | User name: Right User name Password: Right Password | Login success and User redirected to the User  Dashboard. | Pass |
| 3 | Adding member with some wrong input | **Name:** harishgupta  **Email Id:** harsh  **MobileNo.:**89597944811  **Password:** user | Show error message for all fields that are incorrect or invalid. | Pass |
| 4 | Adding member with right input | **Name:** Harish Gupta **Mobile No.:** 8959794481 **Email:**[deepak@gmail.com](mailto:deepak@gmail.com)  **Password:**Admin@123 | Saved at and show success message. | Pass |
| 5 | Select files | Select files and submit | Files uploaded | Pass |

Above test cases proved that all the functions, loops, conditions are working fine in this project

**System Testing:** System testing can be considered as a black-box test technique. Black box Testing technique does not require internal knowledge of the code.

System testing is actually a series of different tests (i.e., performance, security, recovery) whose primary purpose is to fully exercise the computer-based system. Although each test has a different purpose, all work to verify that system elements have been properly integrated and perform allocated functions.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test No.** | **Test Case Objective** | **Actual Output** | **Expected Output** |
| 1 | Correct Workflow? | Each module is correctly connected with each other. | Yes |
| 2 | Behaviours? | Behaviours of the system is user friendly. | User Friendly |
| 3 | Bug Free? | Changes can be made easily. | Yes |

**Performance Testing:**

➢ Better response time because website take less time to load because using of lazy loading.

➢ No refreshing the website while doing any action.

➢ Process activities like redirecting other pages and file uploading etc. on the website are tested.

➢ Interoperability verified i.e.; an application should be able to inter-operate with the other computer and mobile devices.

**Security testing:**

➢ Authentication: Only the authenticated user should be able to Login.

➢ Authorized: User should be able to log into those modules only for which he is authorized or for which the user has been provided access to.

➢ Password: Password requirements are verified i.e., password should be as per how the requirement defines i.e., length, special characters, numbers etc.

➢ Internal links to the web application are not accessible if placed directly in the browser.

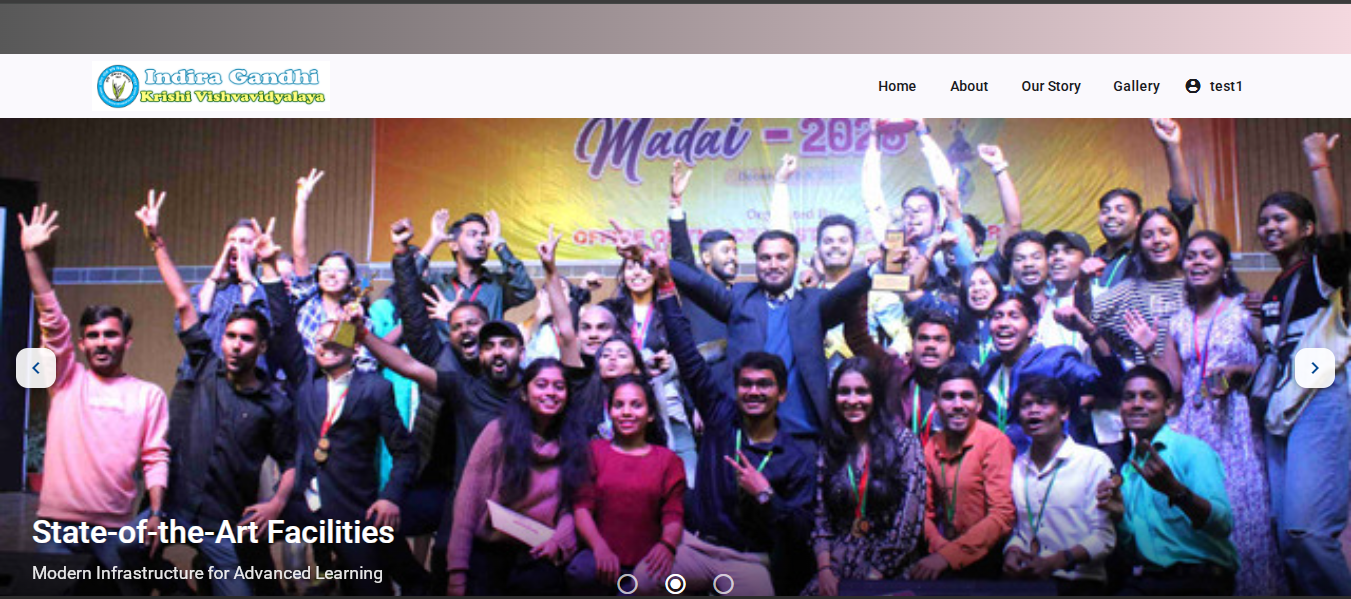
➢ All the communication are encrypted.

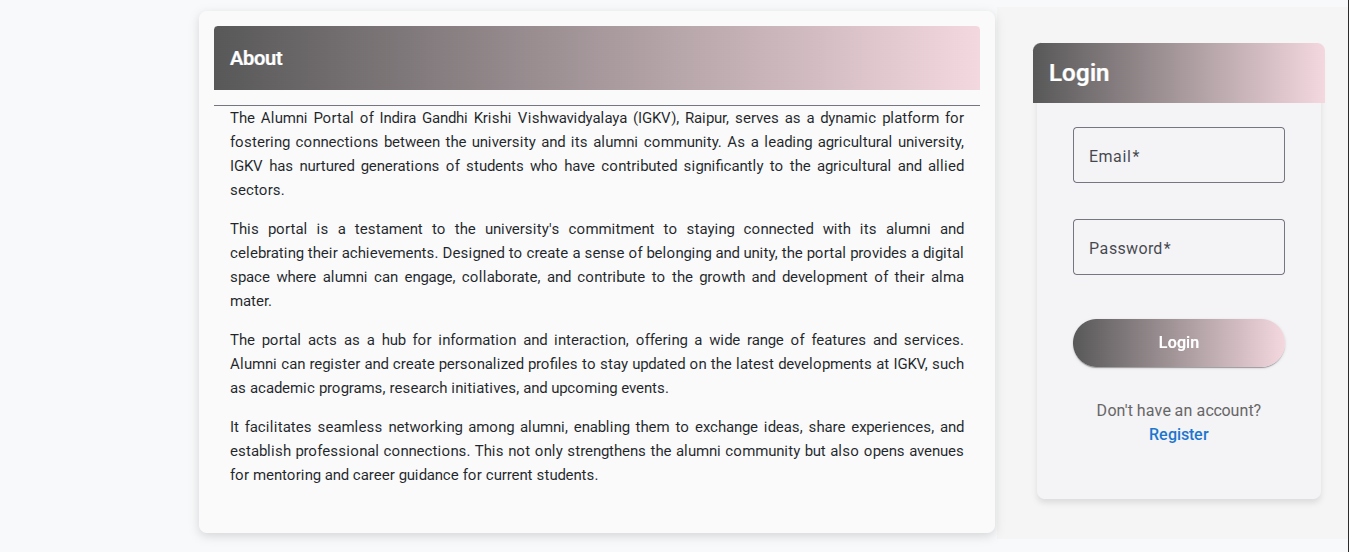
**Chapter 8**

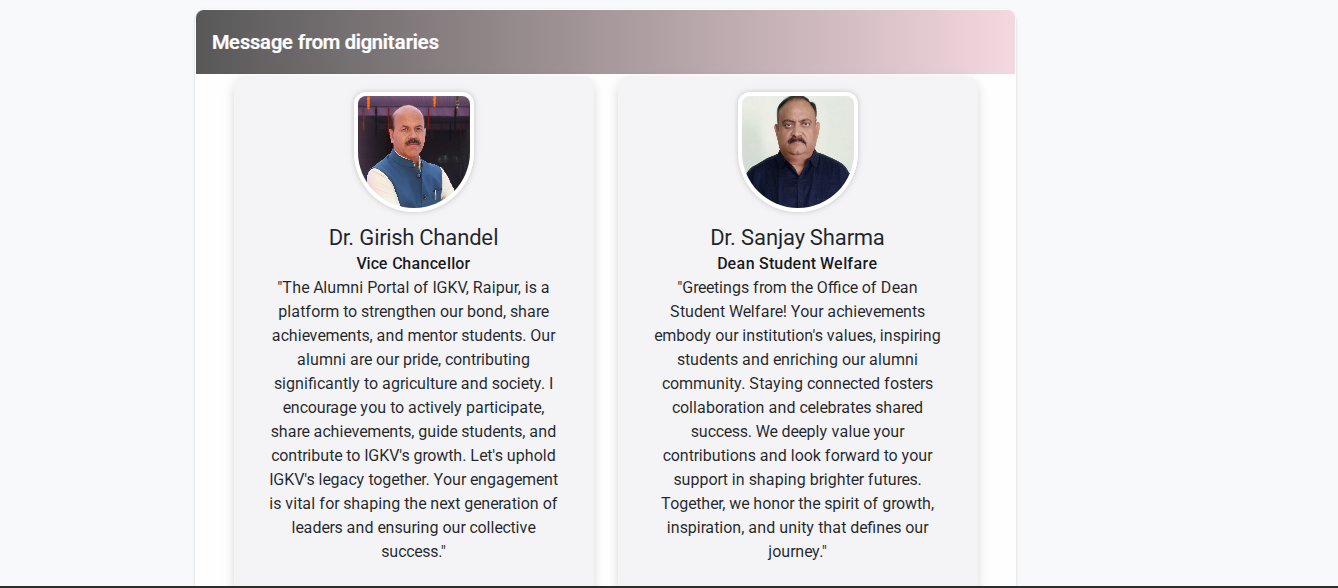
**INPUT – OUTPUT FORMS**

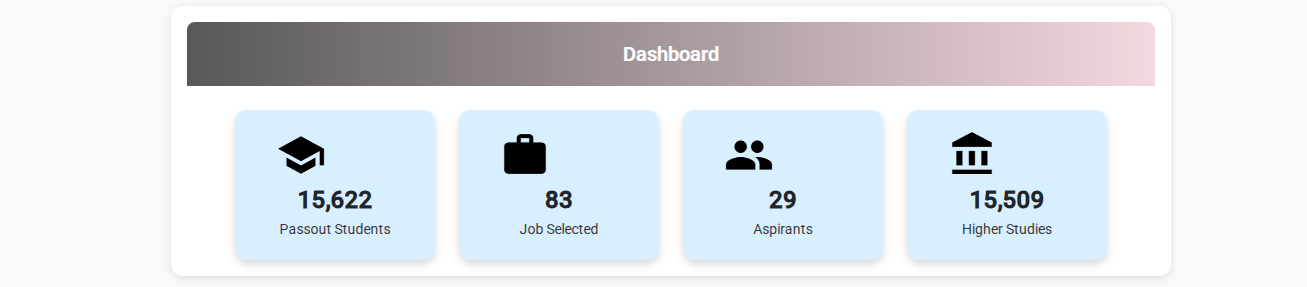
#### Input and Output Screens:

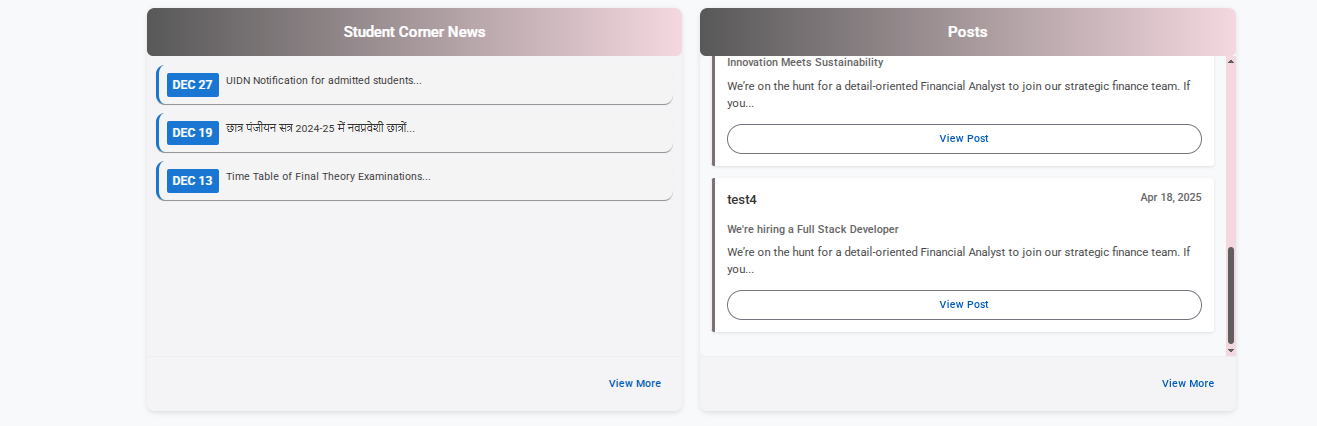
**1. Home Page:** This is the home page or starting page of Uni Alumni Web Portal project.

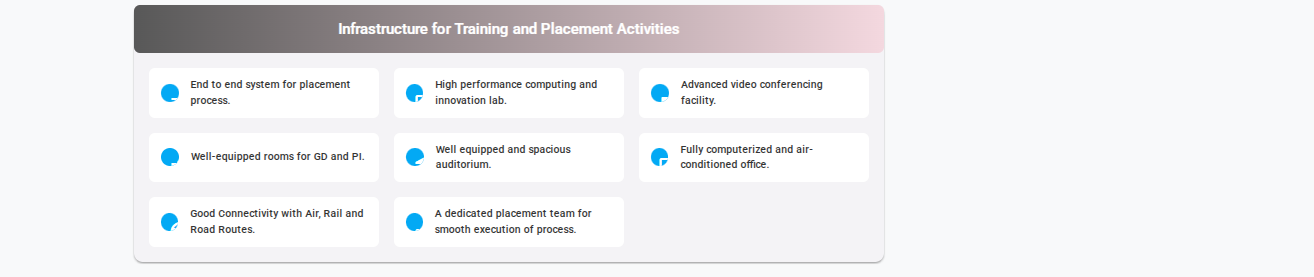
****

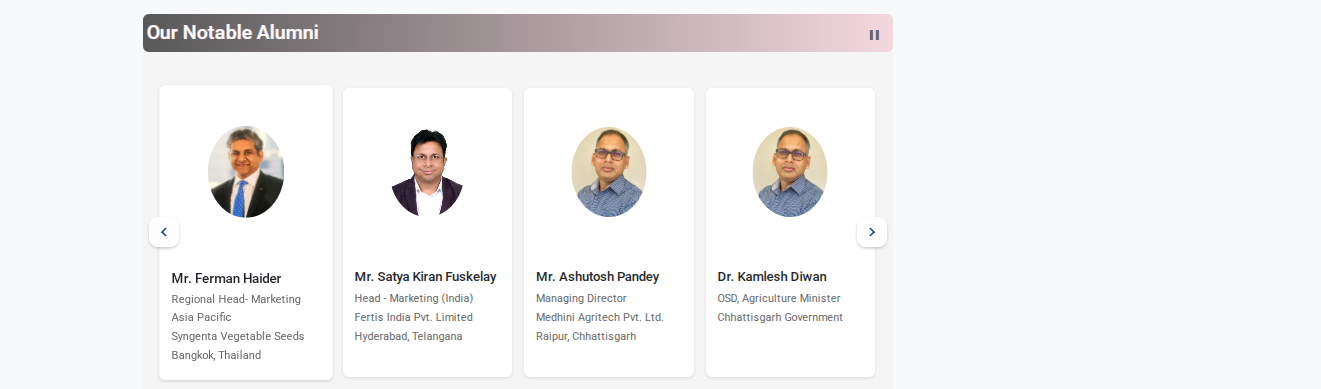
****

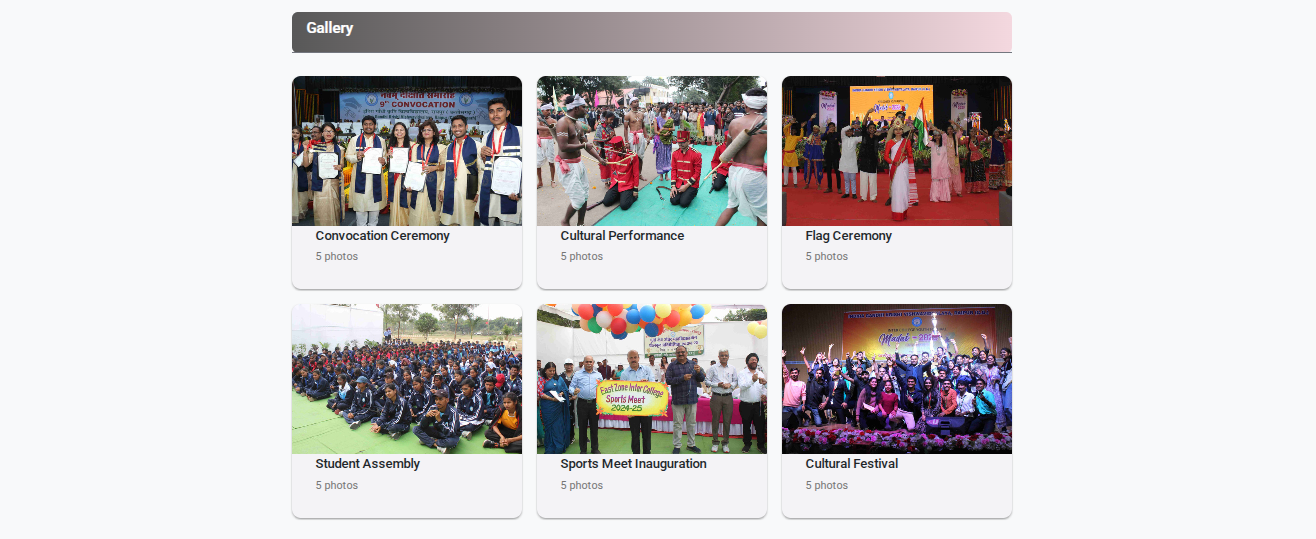
****

****

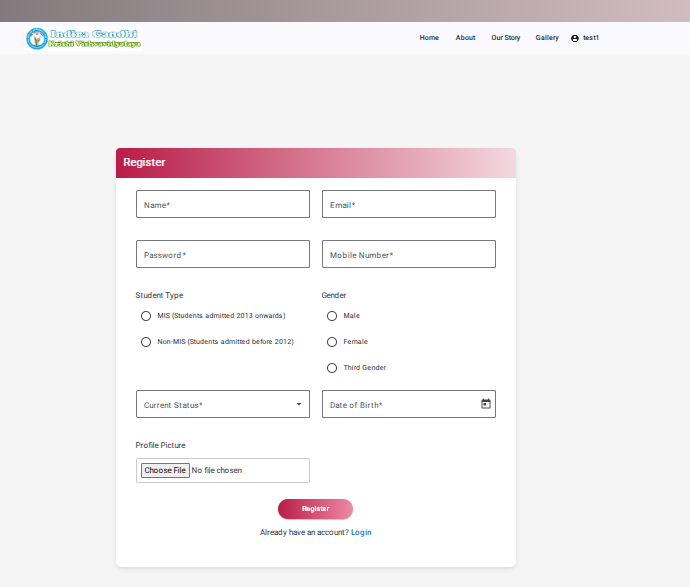
****

****

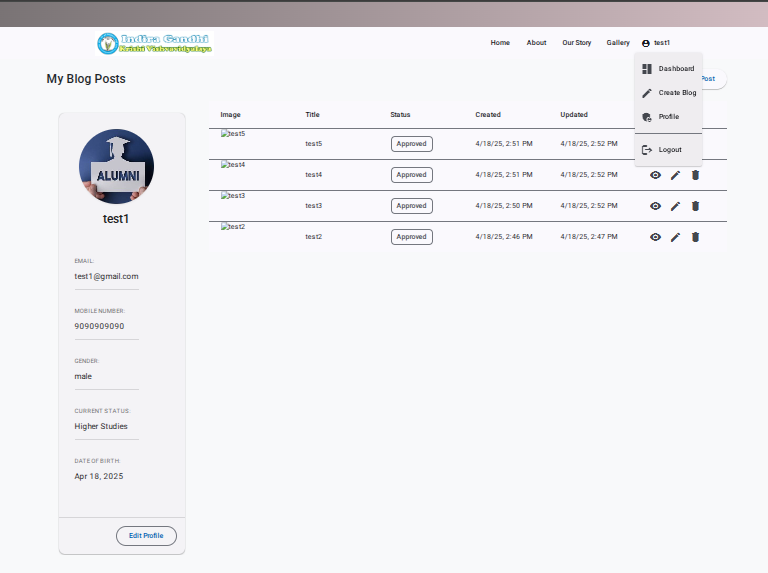
****

****

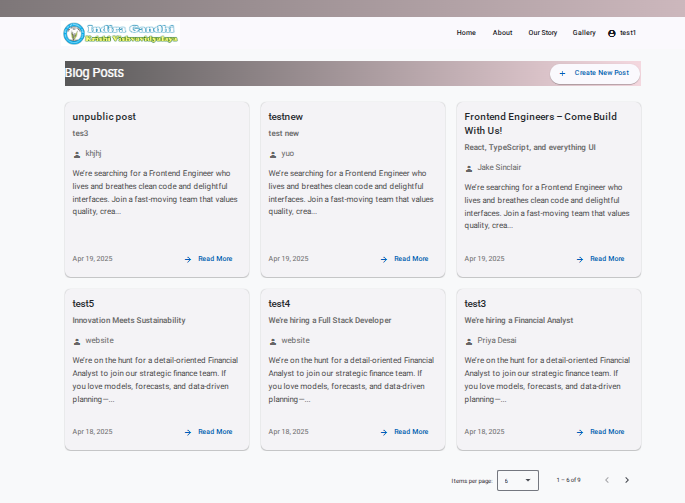
**Alumni Registration page :** In this page user can create our account

****

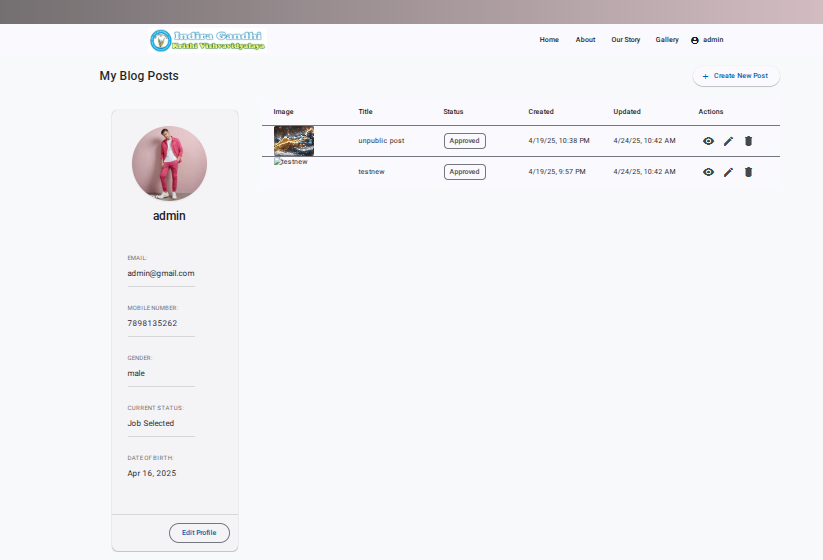
**Alumni Dashboard Page :** In this page alumni can create blog post, update profile information, and get number of post they create . show in dashboard.



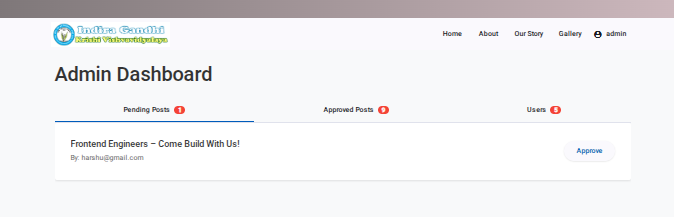
**Our Story Page :** In this page get all alumni created post.



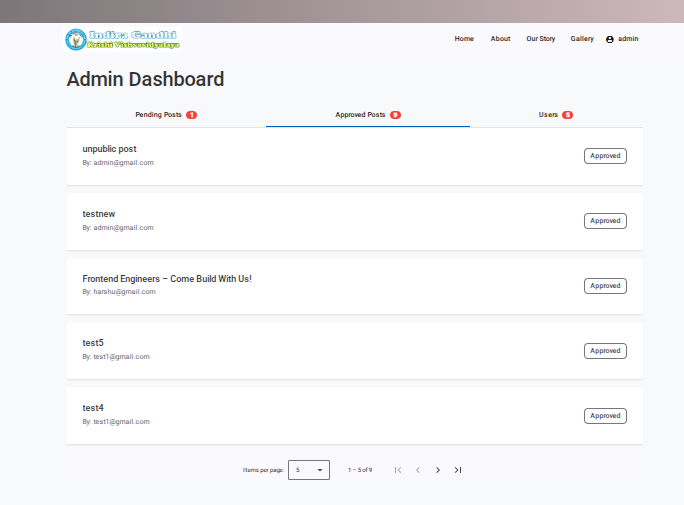
**Admin Dashboard Page :** In this page admin can create blog post, update profile information, and get number of post they create . show in dashboard.



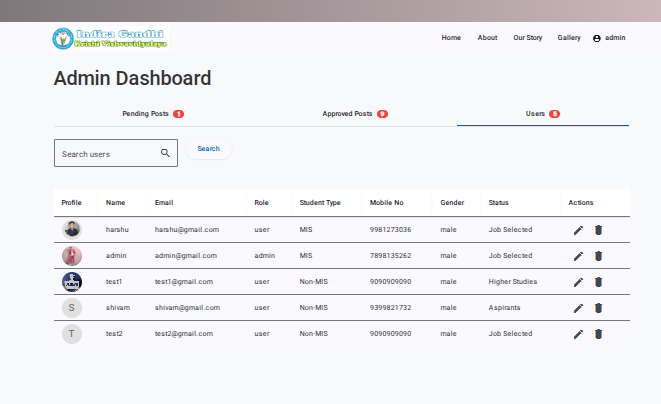
**Admin Dashboard Page :** here admin can check the alumni created post .if post is relevant and use full then approve the post and this approved post display in our story section. And all the approved post list show in approved section and also admin can view and manage alumni details in users section.



img(1)



img(2)



img(3)

**Chapter 9**

**SYSTEM SECURITY**

**Introduction:** The protection of applications that include hardware, software, data, procedures and people against unauthorized use or natural disaster is known as Application Security.

Application security refers to various validations on data in the form of checks and controls to avoid the application from failing. It is always important.

**9.1 Security:**

➢ A secure login and logout facility is provided. Only Registered User can login into the application.

➢ Parameterized queries are used to prevent SQL injection.

➢ Internal pages are only accessible when the user is authenticated.

➢ Here I am using two types of validation i.e., client-side validation and server-side validation to ensure security and to ensure that only valid data is entered and only valid operations are performed on the system.

**9.1.1 Client-Side Validation:** Client-side validation saves server time and loads to handle invalid data. Some checks imposed are:

➢ TypeScript is used to ensure that all required fields are filled with valid data only. Maximum lengths of the fields of the forms are appropriately defined.

➢ Forms cannot be submitted without filling up the mandatory data so that manual mistakes of submitting empty fields that are mandatory can be sorted out at the clientside to save the server time and load. 9.1.2 Server-Side Validation: Server-side validation is also used to protect against malicious users, who can easily bypass our Client-Side scripting language and submit dangerous input to the server. Some of the server-side checks imposed is:

➢ Server-side constraint has been imposed to check for the validity of primary key and foreign key. If any attempt does not satisfy this key, then a proper error message are showing to the user.

➢ Forms cannot be submitted without filling up the mandatory data so that manual mistakes of submitting empty fields that are mandatory can be sorted out at the serverside.

➢ User is intimating through appropriate messages about the successful operations or exceptions occurring at server side.

**Chapter 10**

**Limitation/Future Enhancement**

**Limitation / Future enhancement**

**10.1 Limitation of system**

• This is a web-based application, so this need internet facility.

• This application needs high performance processor and RAM.

• Data processing can be slow if internet will be slow.

**10.2 Future enhancement:**

It is not possible to develop a system that meets all the requirements of the user. User requirements keep changing as the system is being used. Some of the future enhancements that can be done to this system are:

* Remote access: Database access from anywhere cloud services provide.
* OTP/SMS Notification: New system provides new functionality.
* Master Form: In future we will add some form to get additional details
* Chat featurs: In future we will add chat features thus user can chat with each other.

The above-mentioned points are the enhancements which can be done to increase the ability and usage of this project.

**Chapter 11**

**Conclusion**

**Conclusion:** The **University Alumni Web Portal** is a dynamic and strategic digital platform that bridges the gap between graduates and their alma mater. It provides a centralized, interactive space where alumni can stay connected, share their experiences, and contribute to the growth of the university community. The system is designed to serve not only the alumni but also current students and university administrators, offering benefits to all stakeholders.

Through well-integrated modules like **user authentication**, **profile management**, **blog post creation**, **internship/job posting**, and **admin moderation**, the portal ensures seamless communication, professional networking, and career support. With secure login, role-based access, and user-friendly interfaces, the portal encourages active participation and engagement among alumni.

From the university’s perspective, the platform is an essential tool for tracking alumni progress, improving institutional relationships, and enhancing the overall image of the university. Students benefit greatly from real-life insights, internship opportunities, and mentorship from experienced graduates. Administrators can efficiently manage alumni data, approve content, and maintain an active, relevant, and clean environment for interaction.

The **cost-benefit analysis** reveals that while the initial development cost may be significant, the long-term value and return on investment are strong. With minimal ongoing maintenance and infrastructure costs, the portal becomes highly sustainable and beneficial over time.

In conclusion, the University Alumni Web Portal is a forward-thinking solution that enhances university-alumni engagement, fosters student success, and supports a culture of collaboration and lifelong connection. It strengthens the university’s community and reputation while empowering alumni to give back meaningfully and remain an integral part of their educational journey.

**References and Bibliography**

**BIBLIOGRAPHY**

While developing this project, I have used some websites which helped me in our development process.

**WEB REFERENCES:**

**1.** [**https://angular.io/docs**](https://angular.io/docs)

**2.** [**https://stackoverflow.com**](https://stackoverflow.com)

**3.** [**https://nodejs.org/en/docs**](https://nodejs.org/en/docs)

**4.** [**https://www.youtube.com/@TechnicalBabaji1**](https://www.youtube.com/@TechnicalBabaji1)

**5.** [**https://github.com/topics/**](https://github.com/topics/)

**6.** [**https://mariadb.org/documentation/**](https://mariadb.org/documentation/)