**A**

**PROJECT REPORT ON**

**“G-Campus”**

**Submitted To**

**UDHNA CITIZEN COMMERCE COLLEGE &**

**S.P.B. COLLEGE OF BUSINESS ADMINISTRATION &**

**SMT. DIWALIBEN HARJIBHAI GONDALIA COLLEGE OF BCA AND IT**

**Affiliated To**

**VEER NARMAD SOUTH GUJARAT UNIVERSITY, SURAT**

**As A Partial Fulfilment for The Degree Of**

**BACHELOR OF COMPUTER APPLICATION (B.C.A.)**

**T.Y.B.C.A. (SEM. - 6) ACADEMIC YEAR: 2023-24**

**DEVELOPED BY: GUIDED BY:**

1. **Student Name: - Ayush A. Varma (SEAT NO.: 3550) Prof.: - Dr. Ronak Shah**
2. **Student Name: - Ramashankar M. (SEAT NO.: 3415)**
3. **Student Name: - Kishore A. Sunchu (SEAT NO.: 3522)**

**:: PROJECT DEVELOPED AT::**

**UCCC & SPBCBA & SDHG COLLEGE OF BCA AND IT**

**[SURAT]**

**ACKNOWLEDGEMENT**

I extend my deepest gratitude to all those who have contributed to the completion of this project report.

First and foremost, I would like to express my sincere appreciation to Prof. Ronak Shah for their invaluable guidance, encouragement, and unwavering support throughout the duration of this project. Their expertise, constructive feedback, and patience have been instrumental in shaping the direction and quality of this work.

I am also indebted to UCCC & SPBCBA & SDHG COLLEGE OF BCA AND IT for providing the necessary resources and facilities essential for the successful execution of this project.

Furthermore, I wish to extend my heartfelt thanks to my family and friends for their understanding, encouragement, and moral support during the challenging phases of this endeavour. Their belief in my abilities has been a constant source of motivation.

Thank you all for your invaluable contributions.

**Ayush A. Varma**

**Ramashankar M. Pandey**

**Kishore A. Sunchu**

**INDEX**

|  |  |  |
| --- | --- | --- |
| **Chapter**  **No.** | **Chapter Name** | **Page**  **No.** |
| **1.** | **College Profile** |  |
| 1.1 Brief Overview / Highlights |  |
| 1.2 Institute Structure / Chart |  |
| **2.** | **Manual/Current / Existing System Study** |  |
| 2.1 Major Components / Flow |  |
| 2.2 Minimum Hardware / Software Configuration |  |
| 2.3 Drawbacks / Limitations |  |
| **3.** | **Propose Project Profile** |  |
| 3.1 Introduction |  |
| 3.2 Objective / Goal / Aim |  |
| 3.3 Scope |  |
| 3.4 Type of Project |  |
| 3.5 Technology / Environments E.g. tools |  |
| 3.6 Applicability of the system |  |
| **4.** | **Software Analysis** |  |
| 4.1 Preliminary Investigation |  |
| 4.2 Problem Identification |  |
| 4.3 Feasibility study / Risk Analysis |  |
| 4.3.1 Technical Feasibility |  |
| 4.3.2 Economical Feasibility |  |
| 4.3.3 Operational Feasibility |  |
| 4.3.4 Management Feasibility |  |
| 4.3.5 Time Feasibility |  |
| 4.4 Requirement Analysis |  |
| 4.4.1 Fact Finding Techniques |  |
| 4.4.2 Time Line Chart |  |
| 4.4.3 Model with Justification/ Agile Modeling |  |
| 4.4.4 Flow Chart |  |
| 4.4.5 DFD and/or UML |  |
| 4.4.6 Process / Control Specification |  |
| 4.4.7 Data Dictionary |  |

**INDEX**

|  |  |  |
| --- | --- | --- |
| **4.** | 4.4.8 E-R Diagram |  |
| 4.4.9 Data Object Description |  |
| **5.** | **Software Design** |  |
| 5.1 Project Design Process Hierarchy |  |
| 5.2 Database Design |  |
| 5.2.1 Justification of Normalization |  |
| 5.3 Architectural Design |  |
| 5.4 Algorithm Development / Pseudo-code |  |
| 5.5 User Interface Design |  |
| 5.6 Security Issues |  |
| 5.7 Quality / Reliability Measures |  |
| 5.8 System Map |  |
| **6.** | **Software Coding** |  |
| 6.1 Tools & Techniques |  |
| 6.2 Business Logic |  |
| 6.3 Result Snapshot |  |
| 6.4 System generated Reports |  |
| **7.** | **Software Testing** |  |
| 7.1 Test Cases & Test Data Design |  |
| 7.2 Output Comparison |  |
| 7.3 Testing Strategies |  |
| 7.4 Unit Testing |  |
| 7.5 Integration Testing |  |
| 7.6 System Testing |  |
| 7.7 Alpha Testing |  |
| **8.** | **Software Implementation** |  |
| 8.1 User Training |  |
| 8.2 User Manual / Help / SOP |  |
| **9.** | **Limitations / Constraints** |  |
| **10.** | **Future Enhancement / Path-A-Head** |  |
| **11.** | **Bibliography / Appendix / References** |  |
| **12.** | **Other Software Engineering Principles / Tools / Techniques / Models / Guidelines** |  |

**CHAPTER: 1**

**COLLEGE PROFILE**

* 1. **BRIEF OVERVIEW / HIGHLIGHTS**

**COLLAGE NAME:**

UDHNA CITIZEN COMMERCE COLLEGE AND

S.P.B. COLLEGE OF BUSINESS ADMINISTRATION AND

SMT. DIWALIBEN HARIJBAHI GONDLIA COLLEGE OF BCA AND IT

**TELE-PHONE NO. MOBILE NO.**

0261-2277739 8980277739

**ADDRESS:**

214, RANCHHOD NAGAR, UDHNA - NAVSARI MAIN RD, OPP. SWAMI NARAYAN TEMPLE, UDHNA, SURAT, GUJARAT 394210

**EMAIL ID:**

[uccbcc\_uaet@rediffmail.com](mailto:uccbcc_uaet@rediffmail.com)

[spbcba@gmail.com](mailto:spbcba@gmail.com)

[uaccait@gmail.com](mailto:uaccait@gmail.com)

**UDHNA COLLAGE EDUCATION TRUST:**

Udhna academy education trust was established in 1964 with the objective of catering to the educational needs of the citizen of the udhna area (i.e., south zone of Surat city) and South Gujarat. The trust has completed 52 years of brilliance since inception in 1996.it has spread the light of education in this region providing education ranging from pre-primary to higher secondary and graduation. Its pioneers started this institution with a very noble aim and far-reaching vision. As a result, today udhna academy education trust governs the following institutions, where about 6000 students seek high quality education.

**VISION:**

“To be an eminent vibrant institute for education, our credo always be excellence through innovations, empathy, ethics and team work and to cater to the ever-changing needs of community at large.”

**MISSION:**

“To impact quality education, nurture aspirations and facilitate continuous learning and to the society by developing outstanding individuals who would take up leadership challenges in various sectors of economy.”

**SILENT FEATURES:**

* Qualified and Experienced Faculty members
* Book Bank Facility
* Well-equipped Text and Reference Library
* Strong industry-institute Interaction through Seminar, Guest Lectures, Projects, Visits.
* Faculty Feedback System to Strengthen Teaching-Learning Process
* Indoor and Outdoor Co-curricular & Extra-curricular Activities
* Social welfare initiatives in plantation of Trees, Blood donation camp, NSS Camp and relief during natural calamities
* NSS, Sports as character building activities
* Tie-up with Health Centre for free Medical Service to all students and staff
* Well-equipped computer Laboratories with Broad band internet connection
* Canteen Facility for student and staff
* Scholarship to Topper in academies, extra-curricular activities and sports
* Anti-Ragging cell
* Women’s cell
* Placement cell
* Smart cell
* Seminar conducts by doctors

**WOMENCELL:**

Chairperson: Dr Daisy Sheby Thekkanal

Vice Chairperson: Ms. Tvisha J. Parmar

Members:

1. Ms. Perl kharas

2. Ms. Amina Nakhuda

3. Ms. Jinal Purohit

Udhna College has constituted a women cell in 2013-14 to provide harmony atmosphere at the

college for the female students. The objective of the cell is to promote intellectual & cultural

activities, to enhance self-esteem& to develop critical thinking ability of girl student. They can

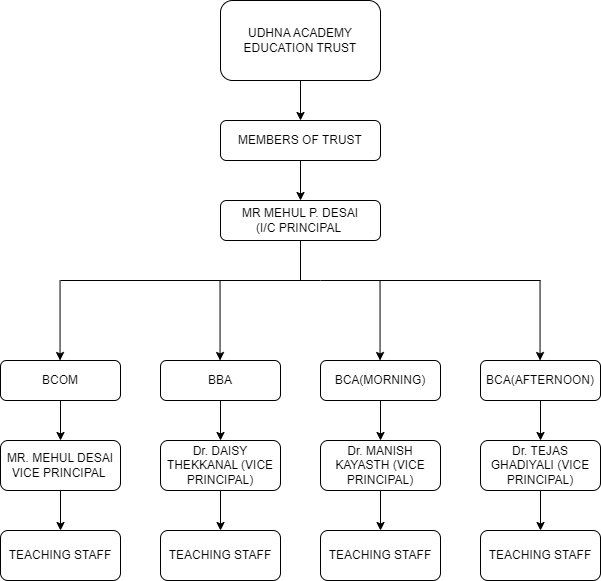
mail their suggestions, feedback creative contributions and can also drop box on the third floor

next to administrative office.

**PLACEMENT & CAREER COUNSELLING CELL:**

The College Placement & Career Counselling Cell invites some very reputed corporate to conduct interviews at the college every year. The firms like FBB Group, ICICI Bank, Kotak Mahindra Bank, Reliance Telecom, Transforms India, etc. visit our college and recruit students. In Addition, under the banner of Career Counselling Cell, the college invites experts and experienced professionals from the industry & corporate world to guide the students about their prospects. Thus, the placement & career counselling cell make shift in students future ready.

* 1. **INSTITUTE STRUCTURE / CHART**



**CHAPTER: 2**

**EXISTING SYSTEM STUDY**

**2.1** **MAJOR COMPONENTS / FLOW**

A college management system typically consists of several major components designed to

streamline various administrative, academic, and operational tasks within an educational

institution. These components may vary depending on the specific needs and functionalities

required by the college, but here are some common ones:

1.Student Information System (SIS):

- This component manages student data including personal information, academic records, enrolment status, grades, attendance, and disciplinary records.

- It allows for student registration, course enrolment, and tracking of academic progress.

2. Administrative Management Module:

- Handles administrative tasks such as admissions, student billing, fee management, financial

aid management, and scholarship/grant distribution.

- Manages staff information including payroll, attendance, and performance evaluation.

3.Course Management System (CMS):

- Facilitates course scheduling, assignment of faculty to courses, and management of course materials.

- Provides tools for curriculum planning, tracking of course progress, and assessment of student learning outcomes.

4. Library Management System:

- Manages library resources including books, journals, periodicals, and multimedia materials.

- Allows for cataloguing, circulation, reservation, and tracking of library items.

- Provides search capabilities and online access to library resources.

5. Examination Management System

- Handles the scheduling, administration, and grading of examinations.

- Manages exam timetables, seating arrangements, and allocation of exam venues.

- Provides tools for generating and analysing exam results.

6. Attendance Management System

- Tracks student and staff attendance.

- Generates reports on attendance patterns and trends.

- May include biometric or RFID-based attendance tracking mechanisms.

7. Human Resource Management System (HRMS)

- Manages employee information including recruitment, hiring, training, and performance evaluation.

- Handles payroll processing, benefits administration, and leave management.

8. Financial Management System:

- Manages budgeting, accounting, and financial transactions.

- Tracks revenue, expenses, and financial performance.

9. Communication and Collaboration Tools:

- Provides communication channels such as email, messaging, and notifications for students, faculty, and staff.

- Facilitates collaboration through discussion forums, online classrooms, and virtual meeting spaces.

10. Security and Access Control:

- Implements security measures to protect sensitive data and ensure compliance with privacy regulations.

- Manages user access rights and permissions to different system functionalities and data resources.

**2.2 MINIMUM HARDWARE / SOFTWARE CONFIGURATION**

**Hardware configuration (Sever side):**

* **Processor:** Intel Xeon or equivalent multicore processor
* **RAM:** Minimum 16 GB
* **Storage:** Minimum 1 TB HDD or SSD (SSD recommended for faster data access)
* **Operating System:** Linux (e.g., Ubuntu Server, CentOS) or Windows Server
* **Database Management System:** MySQL, PostgreSQL, or Oracle Database

**Hardware configuration (Client side):**

* **Processor:** Intel Core i3 or equivalent
* **RAM:** Minimum 8 GB
* **Storage:** Minimum 512 GB HDD or SSD
* **Operating System:** Windows 10 or newer, macOS, or Linux

**Software configuration:**

* **Web Server:** Apache HTTP Server or Nginx
* **Programming Languages**: PHP, Python, JavaScript
* **Frameworks**: Laravel (for PHP), Django (for Python)
* **Database Management System**: MySQL, PostgreSQL
* **Version Control:** Git for managing source code

**2.3 DRAWBACKS / LIMITATIONS**

College management systems offer numerous advantages in streamlining administrative

processes and enhancing communication within educational institutions, they also come with certain drawbacks and limitations:

1. **Initial Cost and Implementation Time**: Implementing a comprehensive college

management system can be costly and time-consuming. It requires significant

investment in software licenses, hardware infrastructure, customization, and staff training.

1. **Complexity and Customization**: College management systems often require

customization to meet the specific needs of each institution. Managing the

complexity of customization can be challenging and may require technical expertise.

1. **Integration Challenges**: Integrating different modules and components within the

system can be complex, especially when dealing with legacy systems or third-party

applications. Incompatibility issues may arise, leading to data inconsistencies and

workflow disruptions.

1. **Data Security Risks**: College management systems store vast amounts of sensitive

data including student records, financial information, and personnel data. Ensuring

data security and compliance with privacy regulations is crucial to prevent unauthorized

access, data breaches, and identity theft.

1. **Dependence on Technology**: Educational institutions become increasingly reliant on

technology for day-to-day operations. System outages, software bugs, or hardware failures

can disrupt critical processes and impact productivity.

1. **User Resistance and Training Needs**: Resistance to change among faculty, staff, and

students can pose challenges during system implementation. Adequate training and

support is essential to ensure smooth adoption and usage of the system.

1. **Scalability and Flexibility**: As educational institutions grow or evolve, scalability and

flexibility become important considerations. College management systems should be

able to accommodate changing needs and scale up to support larger user bases and

additional functionalities.

1. **Maintenance and Upkeep**: College management systems require ongoing maintenance,

updates, and technical support to keep them running smoothly and secure. This can place

additional demands on IT resources and budgets.

1. **User Experience and Usability**: Poor user experience and usability issues can hinder the

adoption and effectiveness of college management systems. Designing intuitive interfaces

and optimizing workflows are essential to enhance user satisfaction and productivity.

**CHAPTER: 3**

**PROPOSE PROJECT PROFILE**

**3.1 INTRODUCTION**

Welcome to G-Campus – Your Gateway to Seamless Academic Excellence!

G-Campus is a cutting-edge college management system designed to revolutionize the way students interact with their educational institution and administrators manage administrative tasks. With a seamless blend of innovative technology and user-centric design, G-Campus offers a comprehensive platform that caters to the diverse needs of students, faculty, and administrators alike.

At G-Campus, we understand the importance of simplifying the academic journey for students while empowering administrators with efficient tools to streamline administrative processes. Whether you're a prospective student exploring academic opportunities or an administrator managing the intricacies of campus life, G-Campus is your one-stop destination for all things academic.

For students, G-Campus offers a user-friendly interface where they can explore detailed information about our college, including program offerings, faculty profiles, admission procedures, and campus facilities. Through our secure portal, students can seamlessly apply for admission, track their application status, and access personalized information about their academic journey.

For administrators, G-Campus provides powerful administrative management tools that simplify complex tasks such as admissions management, student verification, course scheduling, and faculty management. With G-Campus, administrators can efficiently navigate the intricacies of campus administration, freeing up valuable time to focus on fostering academic excellence and student success.

Powered by the latest web technologies including ReactJS, Node.js, MySQL, and Tailwind CSS, G-Campus offers a responsive and dynamic platform that adapts to the evolving needs of our educational community. Our commitment to data security, accessibility, and continuous improvement ensures that G-Campus remains at the forefront of educational innovation, empowering students, and administrators to thrive in an ever-changing academic landscape.

At G-Campus, we believe in the power of education to transform lives. That's why we've created a platform that prioritizes accessibility, efficiency, and excellence in education for all.

**3.2 OBJECTIVE / GOAL / AIM**

The objectives for the G-Campus website could encompass various aspects aimed at benefiting both students and administrators. Here are some potential objectives for the website:

**1. Streamline Admission Process:** Simplify the admission process for prospective students by allowing them to apply online, track their application status, and receive timely notifications about admission decisions.

**2. Enhance Student Experience:** Provide students with a user-friendly platform to access essential information such as course schedules, grades, and announcements, thereby improving their overall academic experience.

**3. Improve Administrative Efficiency:** Empower administrators with tools to efficiently manage admissions, course assignments, faculty appointments, and other administrative tasks, reducing manual effort and streamlining processes.

**4. Drive Continuous Improvement:** Solicit feedback from users and stakeholders to identify areas for improvement and innovation, driving continuous enhancement of the website's functionality, usability, and effectiveness in meeting the needs of its users.

**5. Ensure Data Security and Privacy:** Implement robust security measures to safeguard sensitive student and administrative data, ensuring compliance with privacy regulations and protecting against unauthorized access or data breaches.

**6. Facilitate Academic Planning and Monitoring:** Enable students and administrators to plan, track, and monitor academic progress effectively, facilitating informed decision-making and proactive intervention when necessary.

**7. Promote Accessible Education:** Ensure that the website is accessible to users with disabilities, adhering to web accessibility standards and providing accommodations to ensure equitable access to educational resources and services.

**8. Support Institutional Growth and Innovation:** Lay the foundation for future growth and innovation within the institution by implementing scalable and adaptable technology solutions that can accommodate evolving needs and technological advancements.

**3.3 SCOPE**

The scope of the G-Campus website encompasses a wide range of functionalities and features aimed at improving the overall management and delivery of education within the college. Here are some key areas within the scope of the website:

**1. Admissions Management:** Streamlining the admissions process by providing an online platform for prospective students to apply, submit documents, track their application status, and receive admission decisions.

**2. Student Information System:** Managing student data including personal information, academic records, course schedules, grades, attendance, and extracurricular activities through a centralized student information system.

**3. Administrative Tools:** Providing administrators with tools to manage admissions, course assignments, faculty appointments, financial aid, student billing, and other administrative tasks efficiently.

**4. Course Management System:** Facilitating course planning, scheduling, assignment of faculty, distribution of course materials, tracking of course progress, and assessment of student learning outcomes.

**5. Scalability and Flexibility:** Designing the website architecture to accommodate growth, scalability, and flexibility, allowing for the addition of new features, adaptation to changing requirements, and support for future technological advancements.

**6. Security and Data Privacy:** Implementing robust security measures to protect sensitive student and administrative data, ensuring compliance with privacy regulations, and safeguarding against unauthorized access or data breaches.

**7. User Experience and Accessibility:** Designing a user-friendly interface that is accessible to users of all abilities, ensuring ease of navigation, intuitive interaction, and compatibility across different devices and screen sizes.

**8. Analytics and Reporting:** Providing analytics and reporting tools to track key performance indicators, monitor trends, and generate insights that inform decision-making and drive continuous improvement within the institution.

**3.4 TYPE OF PROJECT**

The G-Campus website, falls under the category of a **Web Application**.

A web application is a software application that runs on a web server and is accessed through a web browser over a network, typically the Internet. It provides users with interactive experiences and functionality like traditional desktop applications, but with the convenience of being accessible from any device with a web browser and an internet connection.

The G-Campus website serves as a centralized platform for managing various aspects of college operations, including admissions, student information, administrative tasks, course management, communication, and collaboration. It leverages modern web technologies such as ReactJS, Node.js, MySQL, and Tailwind CSS to deliver a dynamic, user-friendly, and responsive experience to its users.

**3.5 TECHNOLOGY / ENVIRONMENTS E.G. TOOLS**

Here are the technologies and environments commonly used in developing a web application like G-Campus:

**1. Frontend Development:**

**ReactJS:** A JavaScript library for building user interfaces, allowing for the creation of dynamic and interactive frontend components.

**HTML/CSS:** Standard markup and styling languages for structuring web pages and designing user interfaces.

**Tailwind CSS:** A utility-first CSS framework for quickly building custom designs with pre-built CSS classes.

**JavaScript (ES6+):** The programming language used to add interactivity and behavior to web pages.

**2. Backend Development:**

**Node.js:** A JavaScript runtime environment that allows for server-side scripting, enabling the development of scalable and high-performance backend applications.

**Express.js:** A minimalist web application framework for Node.js, providing a robust set of features for building web servers and APIs.

**3. Database:**

**MySQL:** A relational database management system used for storing and managing structured data related to students, faculty, courses, admissions, etc.

**4. Development Tools:**

**Visual Studio Code:** A popular code editor with a wide range of extensions and features that enhance productivity and facilitate collaborative development.

**Git and GitHub:** Version control system and hosting platform for managing code repositories, enabling collaboration, and tracking changes across development stages.

**Command Line Interface (CLI) tools:** For running scripts, managing dependencies, and other development tasks.

**3.6 APPLICABILITY OF THE SYSTEM**

The applicability of the G-Campus system extends to various stakeholders within the college community, including:

**1. Prospective Students:** Prospective students can use the system to explore academic programs, admission requirements, and campus facilities. They can apply for admission online, track their application status, and receive important updates and notifications.

**2. Current Students:** Current students benefit from the system by accessing their personal dashboard to view course schedules, grades, attendance records, and academic progress. They can also communicate with faculty, access course materials, and participate in online discussions and collaborative activities.

**3. Faculty Members:** Faculty members can use the system to manage course materials, assignments, and grades. They can communicate with students, post announcements, and provide feedback on assignments and assessments. The system also allows faculty to track student progress and identify students who may need additional support or intervention.

**4. Administrators:** Administrators have access to administrative tools that facilitate admissions management, course scheduling, faculty assignments, and financial aid distribution. They can generate reports, analyse data, and make informed decisions to improve institutional efficiency and effectiveness.

**5. Support Staff:** Support staff, such as advisors, counsellors, and registrars, can use the system to provide guidance and assistance to students. They can track student interactions, maintain records, and provide personalized support to help students navigate their academic journey.

**6. Alumni:** Alumni can stay connected with the college community through the system by accessing alumni directories, event calendars, and career resources. They can also contribute to fundraising efforts, mentorship programs, and other initiatives that support the college and its students.

**CHAPTER: 4**

**SOFTWARE ANALYSIS**

**4.1 PRELIMINARY INVESTIGATION**

Preliminary investigation for the G-Campus system involves gathering essential information and conducting initial analysis to determine the feasibility and requirements of the project. Here is an outline of the preliminary investigation process:

**1. Identify Stakeholders:** Identify key stakeholders who will be involved in the development and usage of the G-Campus system. This includes students, faculty, administrators, support staff, and possibly alumni and external partners.

**2. Define Objectives and Scope:** Clearly define the objectives and scope of the G-Campus system. Determine the primary goals the system aims to achieve and the functionalities it should encompass to meet the needs of stakeholders.

**3. Document Requirements:** Document the functional and non-functional requirements of the G-Campus system based on stakeholder input and analysis. Define user roles, system features, data management needs, security requirements, and integration points with existing systems.

**4. Assess Feasibility:** Assess the technical, operational, and financial feasibility of developing and implementing the G-Campus system. Consider factors such as available technology resources, infrastructure requirements, budget constraints, and potential risks.

**5. Explore Existing Solutions:** Research existing college management systems and software solutions to identify best practices, potential vendors, and off-the-shelf options that may meet the requirements of the G-Campus system.

**6. Risk Assessment:** Identify potential risks and challenges associated with the development and deployment of the G-Campus system. Evaluate mitigation strategies and contingency plans to address risks related to technology, resources, stakeholder resistance, and other factors.

**7. Cost-Benefit Analysis:** Perform a cost-benefit analysis to evaluate the potential return on investment (ROI) of implementing the G-Campus system. Compare the expected benefits in terms of improved efficiency, productivity, and user satisfaction against the costs of development, deployment, and maintenance.

**8. Present Findings and Recommendations:** Present the findings of the preliminary investigation, including requirements documentation, feasibility assessment, project plan, risk analysis, and cost-benefit analysis, to key stakeholders and decision-makers. Seek feedback and approval to proceed with the next phases of the project.

**4.2 PROBLEM IDENTIFICATION**

Identifying potential problems or challenges that your G-Campus website may encounter is crucial for ensuring its success and effectiveness. Here are some potential problem areas to consider:

* One challenge may be ensuring that students, faculty, and administrators actively engage with and adopt the G-Campus system. Resistance to change or unfamiliarity with the platform could hinder adoption rates and limit the system's effectiveness.
* Technical issues such as system downtime, slow performance, or bugs in the software could impact user experience and productivity. Ensuring system stability, scalability, and efficient performance under varying loads is essential.
* Protecting sensitive student and administrative data from unauthorized access, data breaches, or privacy violations is critical. Ensuring compliance with data protection regulations and implementing robust security measures are essential to mitigate these risks.
* Poor user experience, confusing navigation, or lack of accessibility features could frustrate users and hinder their ability to effectively use the G-Campus system. Designing an intuitive, user-friendly interface and ensuring accessibility for users of all abilities is essential.
* Integrating the G-Campus system with existing college systems, databases, and third-party applications may present technical challenges and compatibility issues. Ensuring smooth data exchange and interoperability is important for seamless system integration.
* Users may require training and ongoing support to effectively use the G-Campus system and leverage its full potential. Providing comprehensive training resources, user guides, and responsive support channels can help address this challenge.
* Resistance to change among faculty, staff, and students could impede the successful implementation and adoption of the G-Campus system. Addressing concerns, soliciting feedback, and fostering a culture of openness and collaboration can help mitigate resistance to change.
* Identifying and addressing potential gaps in functionality, missing features, or unmet user requirements early in the development process is essential to ensure that the G-Campus system effectively meets the needs of its users.
* Colleges and universities are dynamic environments with evolving needs and requirements. Ensuring that the G-Campus system remains adaptable and responsive to changing needs, technological advancements, and regulatory requirements is essential for its long-term success.

**4.3 FEASIBILITY STUDY / RISK ANALYSIS**

**4.3.1** **Technical Feasibility**

Conducting a feasibility study and risk analysis for the G-Campus website is crucial to assess the project's viability, identify potential challenges, and mitigate risks effectively. Here's how you can approach each aspect:

**- Technical Feasibility:**

- Evaluate the technical requirements for developing the G-Campus website, including the frontend (ReactJS), backend (Node.js), database (MySQL), and other technologies.

- Assess the availability of skilled developers, technology infrastructure, development tools, and third-party services required for the project.

**- Operational Feasibility:**

- Determine if the G-Campus website aligns with the college's operational goals and strategic objectives.

- Evaluate the readiness of stakeholders to adopt and support the new system, including administrators, faculty, students, and support staff.

**- Economic Feasibility:**

- Estimate the initial development costs, including software licenses, development tools, infrastructure setup, and personnel expenses.

- Conduct a cost-benefit analysis to determine the potential return on investment (ROI) and long-term financial viability of the G-Campus website.

**- Schedule Feasibility:**

- Develop a realistic project timeline and milestone schedule based on the scope, complexity, and resources available for the G-Campus website.

- Identify potential dependencies, critical path tasks, and risk factors that may impact the project schedule.

**4.3.2 Economical Feasibility**

Economic feasibility is a critical aspect of determining whether the G-Campus website project is financially viable and offers a positive return on investment (ROI). Here are key considerations for assessing the economic feasibility of the project:

**1. Cost Estimation:**

- Estimate the initial development costs required to build the G-Campus website, including software licenses, development tools, infrastructure setup, and personnel expenses.

- Consider ongoing operational costs such as maintenance, hosting fees, software updates, and support services over the project's lifecycle.

**2. Benefit Analysis:**

- Identify the potential benefits and value proposition offered by the G-Campus website, including improved efficiency, enhanced communication, streamlined processes, and better decision-making.

- Quantify the tangible and intangible benefits in terms of cost savings, productivity gains, revenue generation, and strategic advantages for the college.

**3. Cost-Benefit Analysis (CBA):**

- Conduct a comprehensive cost-benefit analysis to compare the expected benefits of the G-Campus website against the total costs incurred.

- Calculate the net present value (NPV), return on investment (ROI), payback period, and internal rate of return (IRR) to evaluate the project's financial viability and profitability over time.

**4. Risk Assessment:**

- Assess the potential risks and uncertainties that may impact the economic feasibility of the G-Campus website project, including technical risks, market risks, regulatory risks, and operational risks.

- Develop risk mitigation strategies and contingency plans to address potential threats and minimize adverse impacts on project costs and benefits.

**5. Alternative Solutions:**

- Explore alternative solutions and evaluate their economic feasibility compared to the G-Campus website project.

- Consider off-the-shelf software solutions, outsourcing options, or alternative technology platforms that may offer comparable functionality at a lower cost or reduced implementation time.

**6. Scalability and Flexibility:**

- Assess the scalability and flexibility of the G-Campus website infrastructure and technology stack to accommodate future growth, expansion, and evolving requirements.

- Consider the long-term economic implications of scalability upgrades, system enhancements, and technology investments required to sustain the project's success over time.

**7. Strategic Alignment:**

- Evaluate the strategic alignment of the G-Campus website project with the college's overall mission, goals, and priorities.

- Determine how the project contributes to strategic objectives such as student success, institutional excellence, competitive differentiation, and stakeholder satisfaction.

**4.3.3 Operation Feasibility**

Operational feasibility assesses whether the G-Campus website can be effectively integrated into the college's operations and whether stakeholders can adapt to and utilize the system efficiently. Here are key considerations for evaluating the operational feasibility of the website:

**- Stakeholder Readiness:**

- Assess the readiness of stakeholders, including students, faculty, administrators, and support staff, to adopt and use the G-Campus website.

- Identify potential barriers to adoption such as resistance to change, lack of training, or concerns about usability.

**- User Acceptance:**

- Conduct surveys, interviews, or usability tests to gauge user acceptance and satisfaction with the G-Campus website.

- Solicit feedback from stakeholders to identify usability issues, feature requests, and areas for improvement.

**- Training and Support:**

- Develop comprehensive training programs and user guides to help stakeholders learn how to use the G-Campus website effectively.

- Provide ongoing technical support and assistance to address user questions, troubleshooting issues, and system-related concerns.

**- Organizational Processes:**

- Evaluate existing organizational processes and workflows to identify opportunities for integration with the G-Campus website.

- Determine how the website can streamline operations, improve communication, and enhance collaboration among stakeholders.

**- Change Management:**

- Implement change management strategies to facilitate the transition to the G-Campus website and mitigate resistance to change.

- Communicate the benefits of the website, address concerns, and involve stakeholders in the decision-making process to foster buy-in and ownership.

**- Performance Metrics:**

- Define key performance indicators (KPIs) and metrics to measure the effectiveness and impact of the G-Campus website on college operations.

- Monitor system usage, user engagement, efficiency gains, and other relevant metrics to track progress and identify areas for optimization.

**- Compliance and Governance:**

- Ensure that the G-Campus website complies with relevant laws, regulations, and institutional policies governing data privacy, security, accessibility, and usage.

- Establish governance structures and procedures to oversee the operation, maintenance, and evolution of the website in alignment with institutional goals and priorities.

**4.3.4 Management Feasibility**

Management feasibility evaluates whether the G-Campus website can be effectively managed and sustained over the long term. Here are key considerations for assessing the management feasibility of the website:

**- Leadership Support:**

- Evaluate the level of support and commitment from college leadership, including administrators, department heads, and key decision-makers, for the development and implementation of the G-Campus website.

**- Project Governance:**

- Establish clear project governance structures, roles, and responsibilities to oversee the development, deployment, and ongoing management of the G-Campus website.

- Define decision-making processes, communication channels, and escalation procedures to ensure accountability and transparency throughout the project lifecycle.

**- Resource Allocation:**

- Assess the availability of resources, including personnel, funding, technology infrastructure, and external expertise, to support the development, maintenance, and operation of the G-Campus website.

- Allocate resources effectively to meet project milestones, address technical challenges, and sustain the website's functionality over time.

**- Risk Management:**

- Identify potential risks and uncertainties that may impact the successful management of the G-Campus website, including technical risks, operational risks, and stakeholder-related risks.

- Develop risk mitigation strategies, contingency plans, and monitoring mechanisms to address and mitigate potential threats to project success.

**- Change Management:**

- Implement change management processes to facilitate the adoption and integration of the G-Campus website into existing college operations and culture.

- Engage stakeholders, communicate the benefits of the website, and address concerns to minimize resistance to change and promote organizational buy-in.

**- Training and Capacity Building:**

- Provide training and capacity-building programs to equip staff and administrators with the knowledge, skills, and competencies required to effectively manage and use the G-Campus website.

- Offer continuous learning opportunities, technical support, and access to resources to empower stakeholders to leverage the full potential of the website.

**- Performance Monitoring and Evaluation:**

- Establish performance metrics, key performance indicators (KPIs), and benchmarks to monitor the effectiveness, usage, and impact of the G-Campus website on college operations and outcomes.

- Conduct regular performance reviews, user surveys, and stakeholder feedback sessions to assess the website's performance, identify areas for improvement, and make data-driven decisions.

**- Sustainability and Continuity:**

- Develop a sustainability plan to ensure the long-term viability and continuity of the G-Campus website beyond the initial implementation phase.

- Define maintenance schedules, upgrade cycles, and support arrangements to keep the website up-to-date, secure, and aligned with evolving user needs and technological trends.

**4.3.5 Time Feasibility**

Time feasibility assesses whether the development and implementation of the G-Campus website can be completed within a reasonable timeframe and meet project deadlines. Here are key considerations for evaluating time feasibility:

**- Project Scope and Complexity:**

- Evaluate the scope and complexity of the G-Campus website project, including its features, functionalities, and integration requirements.

- Break down the project into manageable tasks and estimate the time required for each phase of development, testing, deployment, and training.

**- Resource Availability:**

- Assess the availability of resources, including skilled developers, designers, project managers, and technology infrastructure, to support the G-Campus website project.

- Ensure that the necessary resources are allocated and accessible throughout the project lifecycle to meet project deadlines and milestones.

**- Technology Stack and Tools:**

- Select appropriate development tools, frameworks, and technologies for building the G-Campus website, considering factors such as development speed, scalability, and compatibility.

- Leverage existing libraries, templates, and platforms to expedite development and reduce time-to-market for the website.

**- Project Management and Planning:**

- Develop a detailed project plan with clear timelines, milestones, and deliverables for the G-Campus website project.

- Identify critical path tasks, dependencies, and potential bottlenecks that may impact project progress and adjust the schedule accordingly to mitigate risks.

**- Risk Management:**

- Anticipate potential risks and challenges that may affect the timeliness of the G-Campus website project, such as technical issues, resource constraints, or scope creep.

- Develop risk mitigation strategies, contingency plans, and alternative approaches to address potential delays and ensure project continuity.

**- Realistic Expectations:**

- Set realistic expectations for project timelines and deliverables based on available resources, technical constraints, and stakeholder requirements.

- Communicate transparently with stakeholders about project timelines, dependencies, and potential risks to manage expectations effectively and build trust.

**4.4 REQUIREMENT ANALYSIS**

**4.4.1** **Fact Finding Techniques:**

Fact-finding techniques are essential for gathering information, understanding requirements, and defining the scope of the G-Campus website project. Here are several techniques that can be used:

* **Interviews**

Conducting interviews with stakeholders, including students, faculty, administrators, and support staff, to gather insights, preferences, and requirements related to the G-Campus website.

* **Surveys**

Administering surveys to a broader audience of stakeholders to collect quantitative and qualitative data on their needs, expectations, and experiences with college management systems.

* **Questionnaires**

Distributing questionnaires to gather structured feedback on specific aspects of the G-Campus website, such as usability, functionality, and user satisfaction.

* **Observation**

Observing users interact with existing college management systems, websites, or related tools to identify usability issues, workflow inefficiencies, and areas for improvement.

* **Document analysis**

Reviewing existing documentation, reports, policies, and procedures related to college management, admissions, academic programs, and student services.

* **Expert Consultation**

Seeking input and guidance from subject matter experts, technology vendors, and industry professionals with experience in college management systems and website development.

**4.4.2 Time Line Chart:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Sr No** |  | **DEC** | | **JAN** | | | | **FEB** | | | | **MAR** | | | | **APR** |
| **WEEK ==============>** | | **1** | **2** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** |
|  | | | | | | | | | | | | | | | | |
| **1.** | **Study of Current System** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **2.** | **Analysis** | | | | | | | | | | | | | | | |
|  | **Identifying needs and Constraints** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Requirement Gathering** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Identify I/P and O/P of system** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **3.** | **Design** | | | | | | | | | | | | | | | |
|  | **Database Design** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Architecture** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **System Map** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Admin module** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Student Module** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Faculty Module** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **4** | **Coding** | | | | | | | | | | | | | | | |
|  | **Admin Module** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Login Module** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **User Module** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Faculty Module** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **5** | **Testing** | | | | | | | | | | | | | | | |
|  | **All Module** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Test Case and Test Data Design** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Output comparison** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Integration and Validation** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **6** | **Documentation** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**4.4.3 MODEL WITH JUSTIFICATION**

It was agreed that the web interface should be implemented in according with software engineering best practice. Software engineering requires the application of a disciplined approach to the software development process. These constraints impose stability and order in an environment that can otherwise become chaotic. Whenever possible, these activities are constrained further by the user of tools such as source code version control software.

For developing our college management system web application, we chosen **Waterfall Model.**

**Waterfall Model:**

The waterfall model is a traditional software development methodology that follows a linear and sequential approach. It is often described as a "one-way" process, resembling the cascading flow of water down a waterfall. Here is a breakdown of its key characteristics:

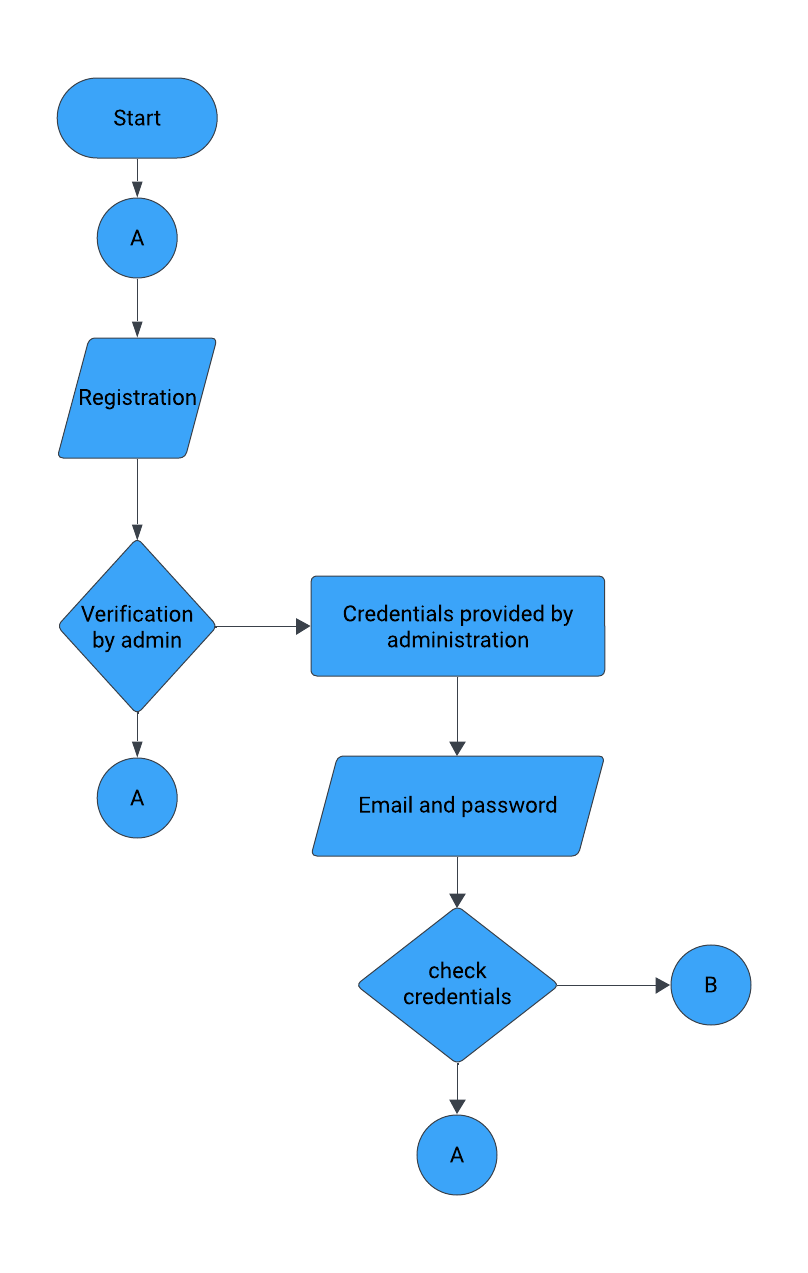
Phases:

The waterfall model is typically divided into distinct phases, each with well-defined deliverables and activities:

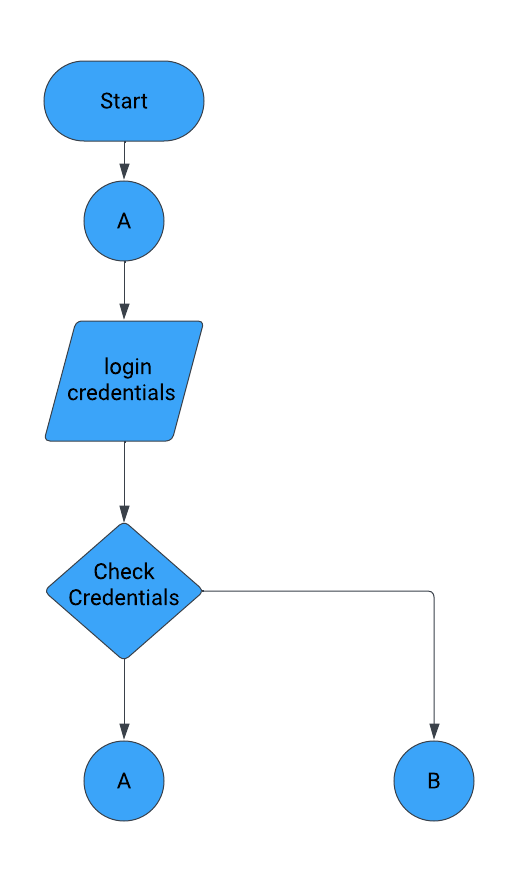
* **Requirement Gathering and Analysis:** This phase involves collecting, documenting, and analysing all the necessary requirements for the website.
* **System Design:** Based on the gathered requirements, this phase focuses on designing the website's architecture, user interface, and system components.
* **Implementation and Coding:** During this phase, developers translate the design into actual code, building the website's functionalities.
* **Testing and Integration:** The developed components are rigorously tested for functionality, usability, and performance.
* **Deployment and Maintenance:** Once testing is complete, the website is deployed to a live environment, and ongoing maintenance and support are provided.

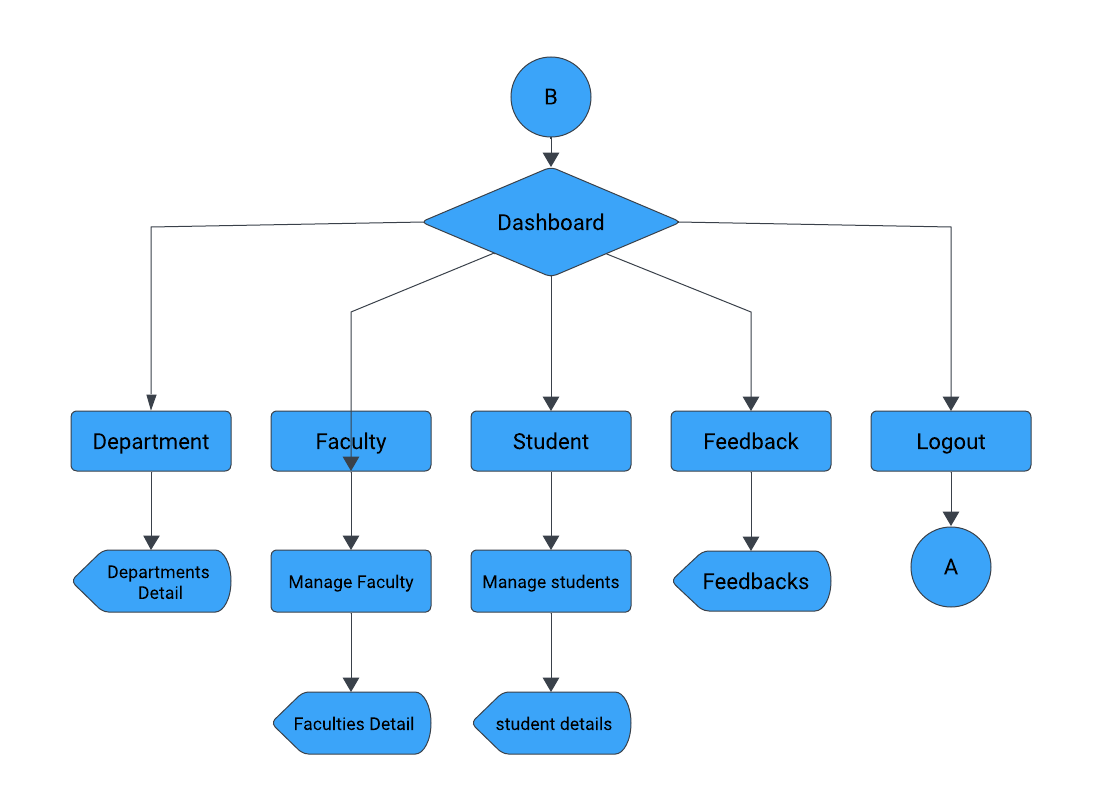
**4.4.4 FLOW CHART**

* **USER FLOW:**



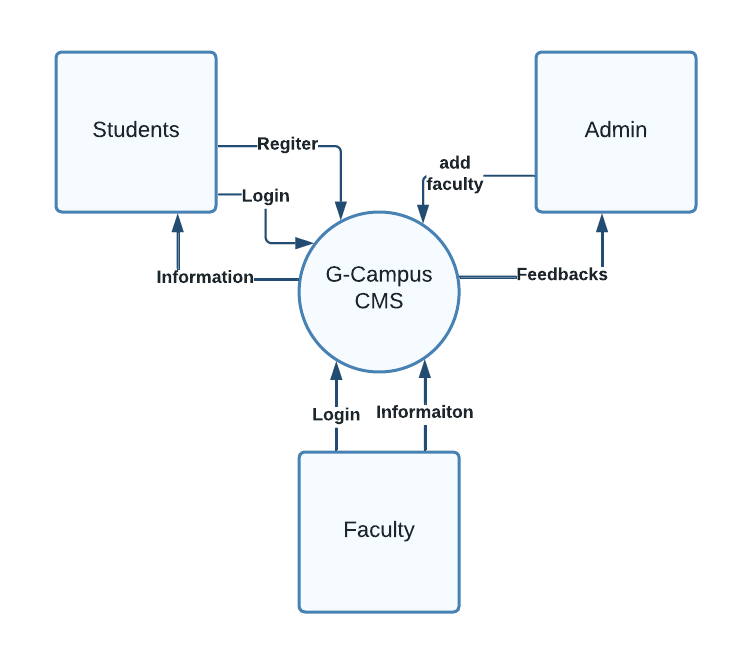
* **ADMIN FLOW:**



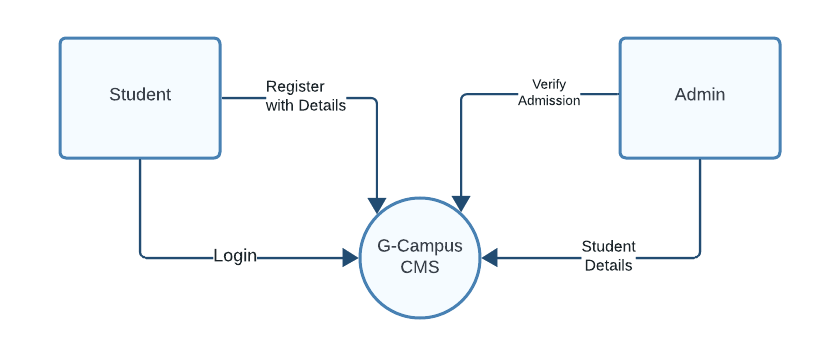


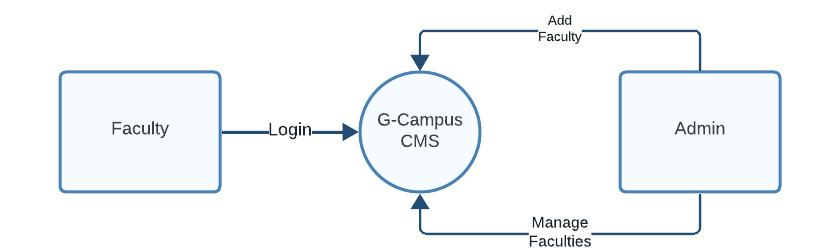
**4.4.5 DFD and/or UML**

* **0th LEVEL DFD**

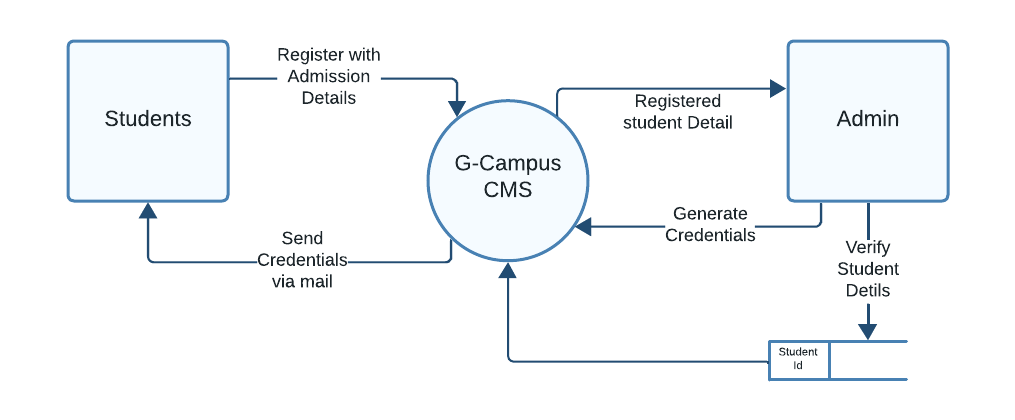


* **1ST LEVEL DFD:**





* **2nd LEVEL DFD**

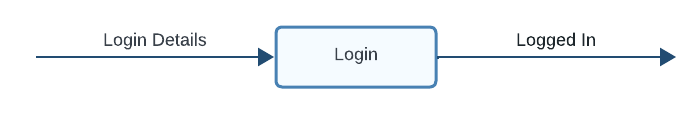


**4.4.6 PROCESS / CONTROL SPECIFICATION**

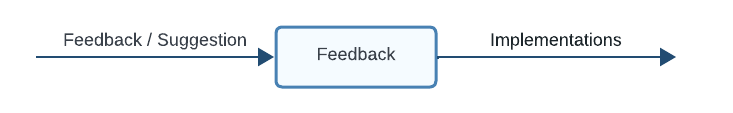
* **Registration process**



* Registration process is used to perform registration operation.
* It takes registration details from user
* User enter the registration detail and detail goes for verification process.
* **Login process**



* Login process is used to perform login operation
* It takes login detail like email address and password from user.
* User enter correct email address and password and login into the system.
* **Feedback Process**



* Feedback process is used to send feedbacks or suggestion to system.
* It takes email, phone, message from user.
* User enter his/her suggestion and send to system.

**4.4.7 DATA DICTIONARY**

* **DEPARATMENT TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| SR No. | Column Name | Datatype | Description |
| 1 | DeptId | Integer | Unique id for departments |
| 2 | DeptName | Varchar | Name of the department |
| 3 | Medium | Varchar | Medium of the department |
| 4 | HeadofDept | Integer | Head of department’s unique id |

* **CLASSES TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| SR No. | Column Name | Datatype | Description |
| 1. | Classid | Integer | Unique id for class |
| 2. | ClassName | Varchar | Name of the class |
| 3. | DeptId | Integer | Unique id of the department |
| 4. | FacultyId | Integer | Unique id of the faculty |
| 5. | Fees | Integer | Fees of the class |

* **FACULTY TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| SR No. | Column Name | Datatype | Description |
| 1. | FacultyId | Integer | Unique id of faculty |
| 2. | FName | Varchar | First name of the faculty |
| 3. | LName | Varchar | Last name of the faculty |
| 4. | Gender | Varchar | Gender of the faculty |
| 5. | Qualification | Varchar | Qualification of the faculty |
| 6. | ClassId | Integer | Unique id of the class |
| 7. | Experience | Varchar | Working experience of the faculty |
| 8. | Status | Varchar | Status of the faculty: active/inactive |

* **FEEDBACK TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| SR No. | Column Name | Datatype | Description |
| 1. | SR No. | Integer | Serial number of the feedback |
| 2. | Full Name | Varchar | Full name of user |
| 3. | Email | Varchar | Email address of user |
| 4. | Phone | Varchar | Phone number of the user |
| 5. | Message | Varchar | Actual message of the feedback |
| 6. | Time stamp | DateTime | Date and Time of the feedback |
| 7. | Status | Varchar | Status of the faculty: active/inactive |

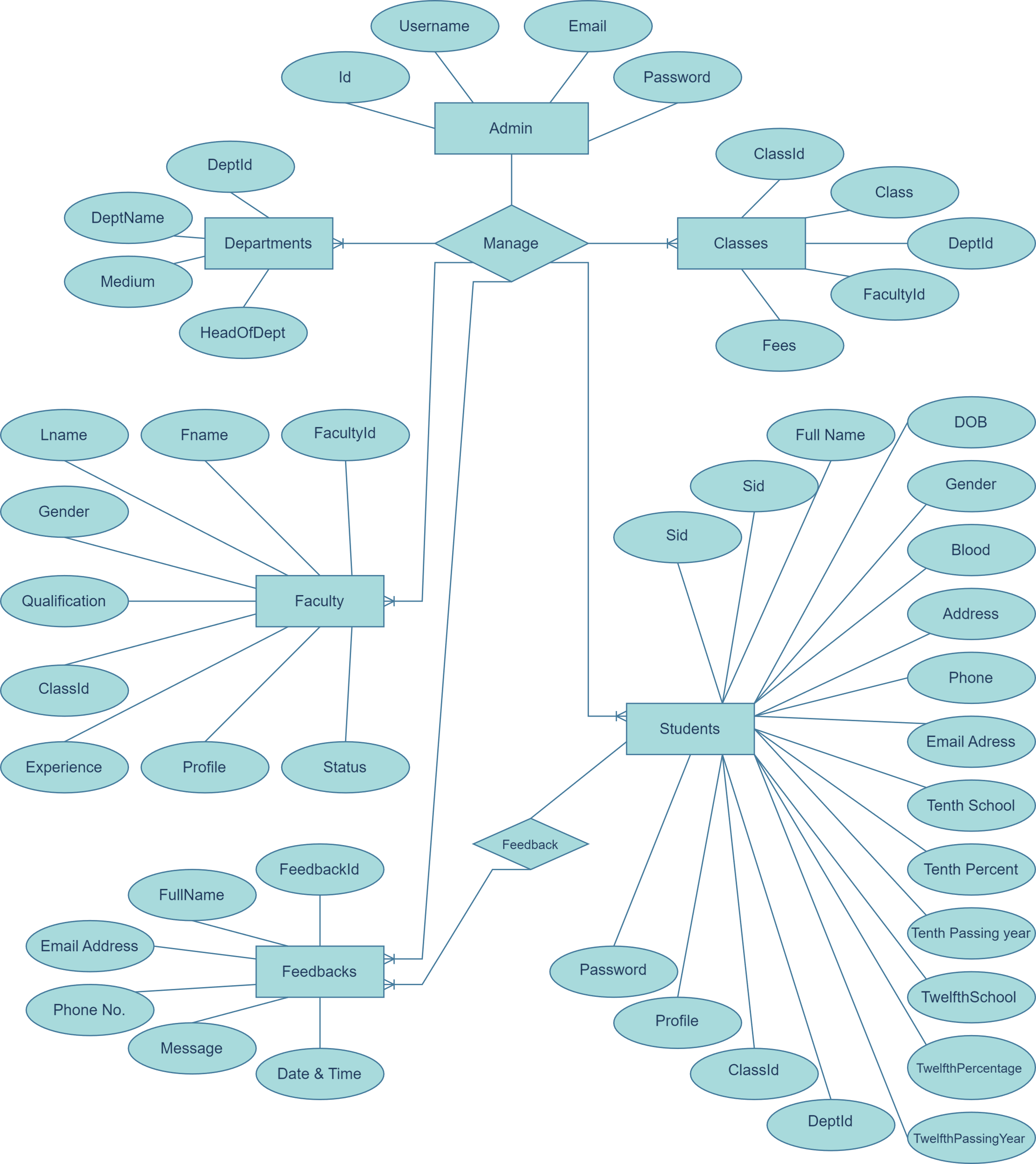
* **STUDENT TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| SR NO. | COLUMN NAME | DATATYPE | DESCRIPTION |
| 1. | Sid | Integer | Unique id of the student |
| 2. | Fname | Varchar | First name of the student |
| 3. | Mname | Varchar | Middle name of the student |
| 4. | Lname | Varchar | Last name of the student |
| 5. | DOB | Date | Date of birth of the student |
| 6. | Gender | Varchar | Gender of the student |
| 7. | Blood | Varchar | Blood group of the student |
| 8. | Address | Varchar | Address of the student |
| 9. | City | Varchar | City of the Student |
| 10. | State | Varchar | State of the student |
| 11. | Phone | Varchar | Phone number of the student |
| 12. | Email | Varchar | Email Address of the student |
| 13. | Tenth school | Varchar | Name of Tenth School |
| 14. | Tenth Passing Year | Varchar | Passing year of tenth school |
| 15. | Tenth Percentage | Varchar | Percentage of tenth school |

* **STUDENT TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| SR NO. | COLUMN NAME | DATATYPE | DESCRIPTION |
| 16. | Twelfthschool | Varchar | Name of Twelfth School |
| 17. | Twelfthpassingyear | Varchar | Passing year of Twelfth school |
| 18. | Twelfthpercentage | Varchar | Percentage of Twelfth school |
| 19. | Deptid | Integer | Unique id of the department- |
| 20. | Classid | Integer | Unique id of the department- |
| 21. | Password | Varchar | Student password for login purpose |
| 22. | Profile | Varchar | Name of the profile picture |
| 23. | Status | Varchar | Status of student |
| 24. | Verified | Varchar | Student verification status |

**4.4.8 ENTITY RELATIONSHIP DIAGRAM**

****

**4.4.8 DATA OBJECT DESCRIPTION**

For the G-Campus website, various data objects are involved in managing information related to students, faculty, courses, admissions, and other aspects of college management. Here is a description of some key data objects:

**- Student Profile:**

- Description: Contains information about individual students, including their name, email, contact details, program/course enrolment, academic records, and personal preferences.

- Attributes: Name, email, phone number, address, program/course enrolled, student ID, academic records (grades, GPA), extracurricular activities, etc.

**- Faculty Profile:**

- Description: Stores details about faculty members, including their name, email, contact information, department, courses taught, academic qualifications, and professional experience.

- Attributes: Name, email, phone number, department, courses taught, faculty ID, academic qualifications (degree, institution), professional experience, research interests, etc.

**- Course Information:**

- Description: Contains data related to courses offered by the college, including course code, title, description, prerequisites, schedule, instructor(s), and enrolment capacity.

- Attributes: Course code, title, description, prerequisites, schedule (days, times), instructor(s), enrolment capacity, classroom/location, semester/term, etc.

**- Admission Records:**

- Description: Stores information about student admissions, including application details, admission status, admission criteria, documents submitted, and enrolment decisions.

- Attributes: Application ID, applicant name, email, application date, admission status (accepted, pending, rejected), admission criteria (test scores, GPA), documents submitted (transcripts, essays), enrolment decision, etc.

**- User Authentication:**

- Description: Manages user authentication and access control for the G-Campus website, including user accounts, login credentials, permissions, and security settings.

- Attributes: Username, email, password (hashed), role/permissions (student, faculty, administrator), last login timestamp, authentication tokens, password reset requests, etc.

**- Administrative Records:**

- Description: Stores administrative data related to college operations, including organizational structure, policies, procedures, and system configurations.

- Attributes: Department/units, organizational hierarchy, administrative roles (administrator, department head), policies/procedures, system settings, configuration parameters, etc.

**- Feedback and Communication:**

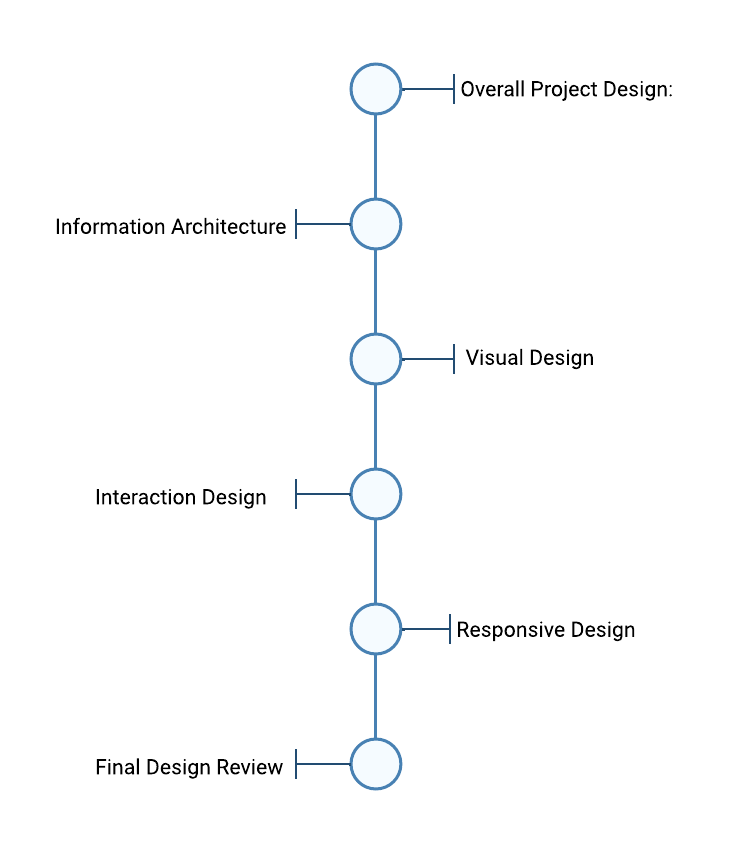
- Description: Manages feedback and communication channels within the G-Campus website, including user feedback, inquiries, announcements, and notifications.

- Attributes: Feedback ID, sender (user or system), message content, timestamp, recipient (individual or group), message type (announcement, notification), read status, response status, etc.

**CHAPTER: 5**

**SOFTWARE DESGIN**

**5.1 PROJECT DESIGN PROCESS HIERARCHY**



**- Overall Project Design:**

- Define the overarching design goals, objectives, and scope of the G-Campus website project.

- Establish design principles, branding guidelines, and user experience (UX) objectives to guide the design process.

**- Information Architecture:**

- Develop the information architecture (IA) of the G-Campus website, organizing content and features into a logical and intuitive structure.

- Create site maps, navigation flows, and content hierarchies to ensure easy access to information and seamless navigation for users.

**- Visual Design:**

- Develop the visual design elements of the G-Campus website, including color schemes, typography, imagery, and UI components.

- Design mock-ups or high-fidelity prototypes that reflect the branding guidelines, design principles, and user experience objectives established earlier.

**- Interaction Design:**

- Define interactive elements and behaviours within the G-Campus website, such as buttons, forms, menus, and animations.

- Ensure consistency, responsiveness, and accessibility across different devices and screen sizes to enhance user engagement and usability.

**- Responsive Design:**

- Implement responsive design techniques to optimize the G-Campus website for various devices and screen resolutions, ensuring a consistent and seamless user experience across desktops, tablets, and smartphones.

**- Final Design Review:**

- Conduct a final design review to evaluate the overall visual appeal, usability, and effectiveness of the G-Campus website design.

- Make any necessary refinements or adjustments based on feedback and design considerations before proceeding to development.

**5.2 DATABASE DESIGN**

* **DEPARATMENT TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| SR No. | Column Name | Datatype | Description |
| 1 | DeptId | Integer | Unique id for departments |
| 2 | DeptName | Varchar | Name of the department |
| 3 | Medium | Varchar | Medium of the department |
| 4 | HeadofDept | Integer | Head of department’s unique id |

* **CLASSES TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| SR No. | Column Name | Datatype | Description |
| 1. | Classid | Integer | Unique id for class |
| 2. | ClassName | Varchar | Name of the class |
| 3. | DeptId | Integer | Unique id of the department |
| 4. | FacultyId | Integer | Unique id of the faculty |
| 5. | Fees | Integer | Fees of the class |

* **FACULTY TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| SR No. | Column Name | Datatype | Description |
| 1. | FacultyId | Integer | Unique id of faculty |
| 2. | FName | Varchar | First name of the faculty |
| 3. | LName | Varchar | Last name of the faculty |
| 4. | Gender | Varchar | Gender of the faculty |
| 5. | Qualification | Varchar | Qualification of the faculty |
| 6. | ClassId | Integer | Unique id of the class |
| 7. | Experience | Varchar | Working experience of the faculty |
| 8. | Status | Varchar | Status of the faculty: active/inactive |

* **FEEDBACK TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| SR No. | Column Name | Datatype | Description |
| 1. | SR No. | Integer | Serial number of the feedback |
| 2. | Full Name | Varchar | Full name of user |
| 3. | Email | Varchar | Email address of user |
| 4. | Phone | Varchar | Phone number of the user |
| 5. | Message | Varchar | Actual message of the feedback |
| 6. | Time stamp | DateTime | Date and Time of the feedback |
| 7. | Status | Varchar | Status of the faculty: active/inactive |

* **STUDENT TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| SR NO. | COLUMN NAME | DATATYPE | DESCRIPTION |
| 1. | Sid | Integer | Unique id of the student |
| 2. | Fname | Varchar | First name of the student |
| 3. | Mname | Varchar | Middle name of the student |
| 4. | Lname | Varchar | Last name of the student |
| 5. | DOB | Date | Date of birth of the student |
| 6. | Gender | Varchar | Gender of the student |
| 7. | Blood | Varchar | Blood group of the student |
| 8. | Address | Varchar | Address of the student |
| 9. | City | Varchar | City of the Student |
| 10. | State | Varchar | State of the student |
| 11. | Phone | Varchar | Phone number of the student |
| 12. | Email | Varchar | Email Address of the student |
| 13. | Tenth school | Varchar | Name of Tenth School |
| 14. | Tenth Passing Year | Varchar | Passing year of tenth school |
| 15. | Tenth Percentage | Varchar | Percentage of tenth school |

* **STUDENT TABLE:**

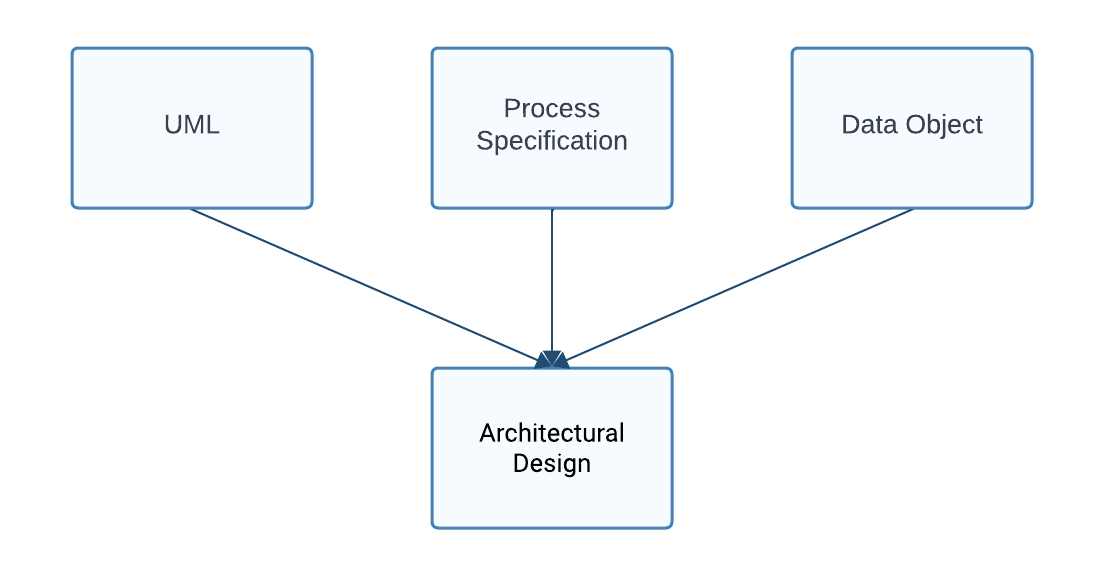
|  |  |  |  |
| --- | --- | --- | --- |
| SR NO. | COLUMN NAME | DATATYPE | DESCRIPTION |
| 16. | Twelfthschool | Varchar | Name of Twelfth School |
| 17. | Twelfthpassingyear | Varchar | Passing year of Twelfth school |
| 18. | Twelfthpercentage | Varchar | Percentage of Twelfth school |
| 19. | Deptid | Integer | Unique id of the department- |
| 20. | Classid | Integer | Unique id of the department- |
| 21. | Password | Varchar | Student password for login purpose |
| 22. | Profile | Varchar | Name of the profile picture |
| 23. | Status | Varchar | Status of student |
| 24. | Verified | Varchar | Student verification status |

**5.3 ARCHITECTURAL DESIGN**

When you making a design of system, it is more necessary to develop the architectural design of system which is useful for basic and the detailed.

The Architectural Design defines the relationship between the design pattern and the constraint of the System. It derived from system specification, the analysis model like UML and interaction of subsystem defined in the model.

When we develop the architectural design, we generally follow the architectural view with data design and design flow.



**ARCHITECTURE WITH DATA DESIGN:**

In this design, we describe all the aspects to relate to external data and entities for the software to be built. Here we consider the data connection of resources with our system and connection with the database.

**ARCHITECTURE WITH DATA FLOW:**

This architecture design considers the data flow of the system. For knowing the flow of the system, we consider some phases of the system analysis like DFD and process specification which gives the detail transaction and transform flow.

**5.4 ALGORITHM DEVELOPMENT / PSEUDO-CODE**

Pseudo-code is an informal way of programming description that does not require any strict programming language syntax or underlying technology considerations. It is used for creating an outline or a rough draft of a program. Pseudo-code summarizes a program's flow, but excludes underlying details. System designers write pseudo-code to ensure that programmers understand a software project's requirements and align code accordingly.

Pseudo-code is not an actual programming language. So, it cannot be compiled into an executable program. It uses short terms or simple English language syntax to write code for programs before it is converted into a specific programming language. This is done to identify top level flow errors, and understand the programming data flows that the final program is going to use. This helps save time during actual programming as conceptual errors have been already corrected.

Firstly, program description and functionality are gathered and then pseudo-code is used to create statements to achieve the required results for a program. Detailed pseudo-code is inspected and verified by the designer's team or programmers to match design specifications.

Catching errors or wrong program flow at the pseudo-code stage is beneficial for development as it is less costly than catching them later. Once the pseudo-code is accepted by the team, it is rewritten using the vocabulary and syntax of a programming language. The purpose of using pseudo-code is an efficient key principle of an algorithm. It is used in planning an algorithm with sketching out the structure of the program before the actual coding takes place.

**Algorithm:** User Registration and Email Verification

**Input:** User details (name, email, password)

**Output:** Success message or error message

1. Receive user details (name, email, password, and more details) from the registration form.

2. Validate user input:

- Check if all required fields are filled.

- Validate email format.

- Ensure password meets complexity requirements

3. If validation fails, display error message and prompt user to correct input.

4. If validation succeeds, proceed to the next step.

5. Check if the provided email address is already registered in the database:

- If email exists:

- Display error message: "Email address is already registered. Please sign in or use a different email."

- End the process.

- If email does not exist:

- Generate a unique verification code for the user (e.g., random alphanumeric string).

- Store user details and verification code in the database.

6. Send an email to the user's provided email address with a verification link containing the generated verification code.

- Email content: "Dear [User], thank you for registering with G-Campus."

7. Display a message to the user: "Registration successful. Please check your email for verification."

8. End the process.