Cloud Counselage

GIFT A CAREER FOUNDATION

PROJECT REPORT

Web Development

Campus Search

Created By:	Atharv Musale	**Approved By:**	
Created On:	10-09-2024	**Approved On:**	23-07-2024

INDEX


Cloud Counselage

1 PROJECT DETAILS

Project Name	Campus Search		
Project Sponsor	Tushar Topale		
Project Manager	Harshada Topale		
Start Date	23-07-2024	**Completion Date**	14-09-2024

2 SUMMARY

The project aimed to deliver a web-based application called Campus Search, designed to help students in India find and discover IT and management colleges efficiently. The project responded to the need for a centralized, user-friendly platform that provides accurate and up-to-date college information without the distractions of advertisements. The long-term benefit of this project will be improved access to reliable college data, helping students make more informed decisions about their education.

3 INTRODUCTION

3.1 Background

Currently, prospective students in India face challenges when searching for colleges. Existing resources are scattered and often contain outdated information. Many websites are cluttered with advertisements that make it difficult for users to focus on relevant content. Campus Search solves this by providing a single platform where students can easily find colleges that match their preferences, especially focusing on IT and management courses.

3.2 Stakeholders

The Campus Search project primarily serves students in India seeking reliable information about IT and management colleges. Secondary stakeholders include colleges, which will gain visibility through the platform, and Cloud Counselage, which may explore partnerships with institutions to enhance student professional development.

3.3 Objectives

The main objective of the Campus Search project is to develop a centralized web application that allows students to easily search for IT and management colleges in India. The platform will feature a responsive, user-friendly design, providing accurate, up-to-date information without distractions, enabling students to make informed decisions about their education.

4 METHODOLOGY

These conventions are all about the positions of line breaks, how many characters should go on a line, and everything in between.

4.1 Considerations & Assumption

- Accurate data collection is essential to ensure reliable search results.
- The project required a flexible design to accommodate future features such as user authentication and possible integrations.
- Assumed that the majority of users will access the platform from mobile devices, necessitating responsive design.

4.2 Approach

A component-based development approach was adopted, where each section of the platform (e.g., search functionality, college details page) was developed and tested independently. This ensured modularity and allowed for easier debugging and updates.

4.3 Activities

- **Requirement Gathering:** Understanding the specific needs for college search and filtering.
- **Database Setup:** Compiling a list of IT and management colleges, ensuring data was complete and up-to-date.
- **Front-End Development:** Creating a responsive and user-friendly interface using HTML, CSS, and JavaScript.
- **Back-End Development:** Setting up a SQL-based database and API endpoints for accessing college data.

5 TARGETTED V/S ACHIEVED OUTPUT

- **Targeted Output:** Develop a fully functional, user-friendly web application with a searchable database of IT and management colleges.
- **Achieved Output:** The project successfully met the primary objectives. A responsive and intuitive web application was built, and it includes a robust search feature and a comprehensive database.

6 CONCLUSION

The Campus Search project has successfully addressed the need for a centralized college search platform, providing accurate information and ease of use. The application will serve prospective students by simplifying the process of finding suitable colleges for IT and management studies in India. Future enhancements could include expanding the database to cover more fields of study and incorporating a user authentication system for personalized search experiences.

7 APPENDICES

7.1 Appendix A – Detailed Description of Components

Identification	Home Page
Type	HTML Page
Purpose	The Home Page allows users to search for colleges by entering search terms such as college name, course, and location.
Subordinates	- Results Page (result.html) - Contact Page (contact.html)
Dependencies	Requires backend services to handle search queries and fetch results.
Interfaces	The page contains a search bar and submit button. Navigates to Results Page upon submission.
Resources	Requires a connection to the database for fetching college information and a frontend JavaScript handler for form submission.
Processing	On submission, JavaScript captures the search input, validates the field, and submits the request to the backend to retrieve the results.
Data	The input consists of search keywords entered by the user, which is then passed to the backend for query execution.

Identification	Results Page
Type	HTML Page
Purpose	Displays search results based on user queries entered on the Home Page. Users can browse through a list of colleges relevant to their search terms.
Subordinates	Contact Page (contact.html)
Dependencies	Requires backend service for fetching search results.
Interfaces	Search result cards, pagination, and filters for additional search refinement.
Resources	Backend service for retrieving filtered data, JavaScript for dynamically rendering the results.
Processing	The backend processes the search query and returns a list of colleges that match the user's criteria, which is displayed dynamically via JavaScript.
Data	College data such as name, location, and available courses are retrieved from the backend database based on the user's search.

Identification	Contact Page
Type	HTML Page
Purpose	Allows users to submit their inquiries or feedback through a contact form.
Subordinates	Home Page
Dependencies	Requires backend service to handle form submissions and send the data to appropriate storage (e.g., database or email).
Interfaces	Contact form with fields like name, email, and message. A submit button triggers form submission.
Resources	JavaScript for form validation and backend service for handling submitted data.
Processing	JavaScript validates form fields (name, email, message), then sends the data to the backend for processing. A confirmation message is shown upon successful submission.
Data	User input such as name, email, subject, and message are validated and sent to the backend.