

Project Documentation



A Project Document on

Educational Website

Submitted to **Laxmi Narain Dubey College, Motihari**

(Affiliated to B.R.A.B.U)

In Partial fulfilment for the award of degree of

Bachelor of Computer Application

Submitted by

Adarsh, Ritika, Vishal, Sakshi & Aliya

Guided by

Prof. Prabhat Kumar

Laxmi Narain Dubey College, Motihari

Jan-2024

Team Member's Name and Roll No.

Name	Roll No.	Registration No.
Adarsh Kumar	214121	20CCVGCS001
Ritika Rani	214124	20CCVGCS005
Vishal Kumar	214131	20CCVGCS012
Sakshi	214148	20CCVGCS032
Aliya Naz	214156	20CCVGCS039

Project Documentation

L.N.D College Motihari of BCA

(Affiliated to B.R.A. Bihar University)

Date:-

TO WHOMSOEVER IT MY CONCERN

This is to certify that the Student **Adarsh, Ritika, Vishal, Sakshi & Aliya** of Laxmi Narain Dubey College, Bachelor of Computer Application, Motihari satisfactorily his Project **EDUCATIONAL WEBSITE** during the period November 2023 to December 2023 in the partial fulfilment of

Name & Signature of Project Guide:

(Prof. Prabhat Kumar)

Signature and seal of Principal

Address:

Project Documentation

ACKNOWLEDGEMENT

I take this occasion to thank God, almighty for blessings with his grace and taking our endeavor to a successful culminative.

I extend my BCA and heartfelt thanks to our esteemed guide, **Prof. Prabhat Kumar**, for providing me with the right guidance and advice at the crucial junctures and for showing me the right way.

I extend my BCA thanks to our respected Head of the College **Prof. Dr. Arun Kumar** for allowing us to use the facilities available.

I extend my BCA thanks to our respected Head of the Department, **Dr. Pinaki Laha**, for allowing us to use the facilities available.

I would like to thank the other faculty members also, at thus occasion. Last but not the least, I Would like to thank my friends and family for the support and encouragement they have given.

Project Documentation

ABSTRACT OF THE PROJECT EDUCATION WEBSITE:

An abstract for an educational website might look like this:

"Welcome to our online education platform, a dynamic and comprehensive hub designed to cater to diverse learning needs. Our website is a digital ecosystem offering a wide array of educational resources, tools, and interactive content, fostering an inclusive learning environment for students, educators, and lifelong learners. From K-12 curriculum support to higher education materials, professional development resources, and specialized courses, our platform aims to inspire, engage, and empower learners of all ages and backgrounds. With a user-friendly interface, personalized learning experiences, and innovative teaching methodologies, we strive to revolutionize the way knowledge is accessed and shared in the digital age. Join us on a journey of exploration, discovery, and growth as we embrace the limitless possibilities of education in the 21st century."

Project Documentation

TABLE OF CONTENTS:

S.No.	Contents	Page No.
1.	Introduction: 1.1 Background 1.2 Objective 1.3 Purpose 1.4 Scope 1.5 Working	6
2.	Requirement and Analysis: 2.1 Problem definition 2.2 Requirement 2.3 Hardware Requirement 2.4 Software Requirement 2.5 Planning	20
3.	System Desing: 3.1 System Design 3.2 Data Dictionary	50
4.	Testing	58
5.	Source Code & Snapshot of web pages	65 & 145
6.	Conclusion: 5.1 Conclusion 5.2 Bibliography	156

Project Documentation

CHAPTER 1: INTRODUCTION

The Educational Website is a collections of web page. The main purpose of “Educational website” is to provide convenient way for a student to gather information about college.

Project Title	Furniture Management System
Project Description	Educational Web Application, It is a website
Project Duration	2 Months
Project Guide	Prof. Prabhat Kumar
Platform	Window 10
Technologies	HTML, CSS, JavaScript & PHP
Tools Used	Microsoft Visual Studio, XAMPP

Project Documentation

INTRODUCTION OF THE PROJECT EDUCATIONAL WEBSITE:

Introduction to an Educational Website:

Welcome to “Educational Website”, our premier destination for all things related to education. Whether you’re a student, teacher, parent, or anyone passionate about learning, our website is designed to be your go-to resource for educational information, resources, and inspiration.

At “Educational Website”, we believe that education is the key to personal and societal growth. We are dedicated to providing a platform where learners of all ages and backgrounds can access a wealth of knowledge and tools to help them achieve their educational goals. Our mission is to empower individuals to explore, discover, and excel in their educational journeys.

Welcome to Learniverse, your gateway to a world of comprehensive education designed to elevate students in their academic journey. Learniverse is not just an educational website; it's a dynamic platform committed to nurturing young minds, offering tailored courses for both academic excellence and competitive success.

For students in the 9th and 10th grades, Learniverse provides a meticulously crafted academic curriculum that aligns with school syllabus. Our courses are designed to enhance understanding, foster critical thinking, and lay a strong foundation for future academic pursuits. We believe that learning should be engaging, and our platform integrates innovative teaching methodologies to make education a captivating experience.

But Learniverse goes beyond traditional academics. We recognize the importance of competitive exams in shaping a student's trajectory. That's why we offer specialized courses for exams like NTSE and Navodaya entrance exams. Our exam preparation modules are meticulously curated to provide students with the knowledge, skills, and confidence needed to excel in these competitive arenas.

At Learniverse, we understand that each student is unique, and their educational journey should reflect that uniqueness. Our courses are designed to cater to diverse learning styles, ensuring that every student can thrive in their academic and competitive pursuits. Join

Project Documentation

Learniverse, where education meets innovation, and let's embark on a journey of knowledge, exploration, and success together. Your future, our commitment!

Welcome to Code India, a dynamic realm within the vast Learniverse website dedicated to empowering students from class 6 to 12 with the art and science of programming. At Code India, we believe in unlocking the potential of young minds, fostering a passion for coding, and preparing students for the challenges and opportunities of the digital age.

About Code India:

Code India is not just a platform; it's an educational journey tailored to cater to the unique needs of students exploring the world of programming languages. Our courses are meticulously designed to make coding accessible, enjoyable, and intellectually rewarding for students at various academic levels.

Key Features of Code India:

1. Comprehensive Programming Courses
2. Targeted Audience
3. Interactive Learning
4. Skill Development

Integration with Learniverse:

Code India is seamlessly integrated into the Learniverse website, a comprehensive educational platform offering a wide array of academic and competition exam preparation courses. This integration allows students to explore additional learning opportunities beyond coding, enhancing their overall educational journey.

Learniverse Overview:

Learniverse, the parent platform, is a hub for academic excellence and holistic learning. It provides courses to prepare students not only for their academic curriculum but also for competitive exams, ensuring a well-rounded educational experience.

At Code India, we are committed to nurturing the next generation of programmers, problem solvers, and innovators. Join us on this exciting learning adventure, where coding becomes a gateway to endless possibilities in the ever-evolving landscape of technology. Welcome to Code India, where the language of the future is yours to learn!

Project Documentation

Advantages of Code India

1. Early Exposure to Coding
2. Tailored for School Students
3. Integrated with Learniverse
4. Holistic Learning Environment
5. Skill Development
6. Preparation for Future Challenges

Future Goals of Code India:

1. Expansion of Course Offerings
2. Collaborations and Partnerships
3. Continuous Curriculum Enhancement
4. Engagement and Community Building
5. Integration with Industry Standards
6. Global Reach

In summary, Code India, as part of Learniverse, has the advantage of being integrated into a broader educational platform. Its future goals likely involve continuous improvement, expansion, and ensuring its offerings remain aligned with the evolving landscape of programming education.

What you can expect from our educational website:

1. **Comprehensive Information:** We offer a wide range of educational content, including articles, guides, and tutorials that cover a variety of subjects, from mathematics and science to language arts and history. Our goal is to provide you with reliable and up-to-date information that enhances your understanding of different topics.
2. **Learning Resources:** We provide an extensive collection of learning resources, such as study guides, worksheets, and interactive quizzes, to support students at various levels. These resources are designed to help you grasp complex concepts and develop your skills.

Project Documentation

3. **Teaching Support:** Teachers play a crucial role in shaping the future, and we're here to support them. Our website offers lesson plans, teaching strategies, and classroom materials to assist educators in delivering effective and engaging lessons.
4. **Parental Guidance:** We understand that parents are integral to a child's educational success. Our site offers advice, tips, and resources to help parents actively participate in their children's education and provide a supportive learning environment at home.
5. **Inspiration and Motivation:** Education is a journey, and we want to inspire and motivate you to keep learning and pushing your boundaries. You'll find success stories, educational quotes, and articles to keep you enthusiastic about your learning adventure.
6. **Community and Interaction:** We encourage interaction and community-building among our users. Connect with like-minded individuals, ask questions, and share your knowledge through our forums, discussion boards, and social media platforms.
7. **News and Updates:** Stay informed about the latest developments in the education world. We provide news and updates about educational policies, innovations, and trends to keep you in the loop.

Join us at "Educational Website" and embark on a fulfilling educational journey. Together, we can foster a love for learning, develop critical skills, and build a brighter future. Education is the cornerstone of progress, and we're here to help you unlock your full potential. Start exploring, start learning, and start succeeding with "Education Website" today!

OBJECTIVE OF PROJECT EDUCATIONAL WEBSITE:

The objective of an educational website can vary depending on the specific goals and target audience.

Here are some common objectives that educational websites often aim to achieve:

1. **Dissemination of Information:** Provide a platform for the efficient and organized delivery of educational content, resources, and information to a wide audience.

Project Documentation

2. **Learning Facilitation:** Facilitate learning by offering engaging and interactive materials, such as tutorials, videos, quizzes, and other educational tools.
3. **Accessibility:** Ensure that educational resources are accessible to a diverse audience, including individuals with different learning styles, abilities, and backgrounds.
4. **Skill Development:** Support the development of skills and competencies by offering courses, workshops, and training materials that align with the needs of the target audience.
5. **Community Building:** Foster a sense of community among learners, educators, and other stakeholders by providing forums, discussion boards, or social media integration to facilitate collaboration and knowledge-sharing.
6. **User Engagement:** Create an engaging and user-friendly interface to keep learners motivated and interested in the educational materials. This may include the use of multimedia, gamification, and other interactive elements.
7. **Assessment and Feedback:** Implement assessment tools, quizzes, and feedback mechanisms to evaluate learner progress and provide constructive feedback for improvement.
8. **Continuous Improvement:** Collect and analyze data on user interactions and learning outcomes to continually improve and update the content and features of the educational website.
9. **Resource Centralization:** Act as a central hub for educational resources, bringing together materials from various sources to make learning more convenient for users.
10. **Adaptability:** Design the website to be adaptable to different learning environments, including formal education, self-paced learning, and professional development.
11. **Promotion of Lifelong Learning:** Encourage a culture of lifelong learning by providing resources that cater to learners of all ages and backgrounds.

Project Documentation

12. **Inclusivity:** Ensure that the website is inclusive, considering the needs of diverse learners, including those with disabilities, and providing accommodations as necessary.
13. **Alignment with Educational Standards:** Align content and courses with relevant educational standards to ensure credibility and relevance.
14. **Promotion of Critical Thinking:** Foster critical thinking skills by presenting information in a way that encourages analysis, evaluation, and synthesis of knowledge.
15. **Career Development:** Provide resources and information that support career development, such as job market trends, industry insights, and skill-building opportunities.

It's essential to tailor the objectives based on the specific mission and vision of the educational website, as well as the needs of the target audience.

PURPOSE OF EDUCATIONAL WEBSITE:

The purpose of an educational website is multifaceted, encompassing various goals and objectives aimed at facilitating learning, disseminating information, and fostering a positive educational experience.

Here are some key purposes of an educational website:

1. **Accessible Learning Resources:** Provide a platform for easy access to educational content, resources, and materials, making learning more convenient for a diverse audience.
2. **Global Reach:** Extend the reach of education beyond geographical boundaries, allowing individuals from different locations to access quality educational materials and opportunities.

Project Documentation

3. **Flexible Learning Opportunities:** Offer flexibility in learning by providing self-paced courses, on-demand resources, and materials that cater to different learning styles and schedules.
4. **Skill Development:** Support the development of various skills and competencies, including academic, professional, and personal skills, through targeted courses and resources.
5. **Supplementary Learning:** Serve as a supplementary resource to formal education, providing additional materials, explanations, and examples to reinforce classroom learning.
6. **Lifelong Learning:** Encourage a culture of lifelong learning by offering resources that cater to learners of all ages and backgrounds, promoting continuous personal and professional development.
7. **Interactive Learning:** Foster engagement and interactivity through the use of multimedia, simulations, quizzes, and other interactive tools to enhance the learning experience.
8. **Collaboration and Community:** Facilitate collaboration and knowledge-sharing among learners, educators, and experts through discussion forums, social media integration, and collaborative projects.
9. **Professional Development:** Provide resources and courses that contribute to the professional development of individuals, helping them stay current with industry trends and advancements.
10. **Career Guidance:** Offer information and resources related to career paths, job opportunities, and industry insights to assist learners in making informed decisions about their future.
11. **Teacher Support:** Support educators by offering teaching resources, lesson plans, and tools that enhance their effectiveness in the classroom and help keep their teaching methods up-to-date.

Project Documentation

12. Assessment and Feedback: Implement tools for assessing learner progress, providing feedback, and monitoring performance to support continuous improvement and understanding.

13. Inclusivity: Create an inclusive learning environment that accommodates diverse learners, including those with different abilities, learning styles, and backgrounds.

14. Adaptive Learning: Utilize technology to personalize learning experiences, adapting content and pace based on individual learner needs and preferences.

15. Research and Innovation: Serve as a platform for educational research and innovation, allowing educators and researchers to share findings, methodologies, and best practices.

16. Promotion of Critical Thinking: Encourage critical thinking and problem-solving skills by presenting information in a way that prompts analysis, evaluation, and application of knowledge.

17. Cultural Exchange: Facilitate cultural exchange and understanding by offering diverse perspectives and resources that reflect different cultures and viewpoints.

18. Preparation for Standardized Tests: Provide resources and practice materials to help learners prepare for standardized tests and assessments.

The overall purpose of an educational website is to contribute to the development and enrichment of individuals through accessible, engaging, and effective learning experiences. Whether used as a supplement to formal education or as a standalone learning platform, educational websites play a crucial role in democratizing education and promoting knowledge dissemination on a global scale.

SCOPE OF THE PROJECT EDUCATIONAL WEBSITE:

The scope of an educational website project encompasses all the features, functionalities, and content that the website will include to achieve its objectives. The scope defines the boundaries and parameters of the project, outlining what will and will not be included.

Project Documentation

Here are some elements to consider when defining the scope of an educational website:

1. **Content Development:**

- Define the types of educational content to be included (e.g., text-based articles, videos, interactive simulations, quizzes).
- Determine the subjects or topics that the content will cover.
- Specify whether the content will be created in-house or sourced from external providers.

2. **User Features:**

- Identify user roles (students, teachers, administrators) and determine their respective privileges and access levels.
- Specify user authentication and authorization mechanisms.
- Define user engagement features, such as discussion forums, chat functionality, and collaborative tools.

3. **Learning Management System (LMS):**

- Decide whether the website will have a Learning Management System for course creation, enrollment, tracking progress, and managing assessments.
- Outline the features of the LMS, such as course organization, grading, and reporting.

4. **Interactive Tools:**

- Determine if the website will include interactive tools, such as virtual labs, simulations, or educational games.
- Specify the technology required to implement these interactive elements.

5. **Assessment and Feedback:**

- Define the assessment methods (quizzes, exams, projects) and feedback mechanisms.
- Outline how progress will be tracked and communicated to learners.

Project Documentation

6. Community and Collaboration:

- Specify features that support community building, such as discussion forums, social media integration, or collaborative projects.
- Determine moderation and administration tools for community interactions.

7. Mobile Responsiveness:

- Decide if the website will be accessible and usable on various devices, including mobile phones and tablets.
- Specify the design and functionality considerations for different screen sizes.

8. Accessibility and Inclusivity:

- Establish guidelines and features to ensure the website is accessible to individuals with disabilities.
- Consider features such as text-to-speech, alternative text for images, and adaptable fonts and color schemes.

9. Technology Stack:

- Determine the technology stack required for development, including the programming languages, databases, and frameworks.
- Specify hosting and infrastructure requirements.

10. Integration with External Systems:

- Identify any external systems or APIs that need to be integrated, such as payment gateways, analytics tools, or content delivery networks.

11. Security Measures:

- Outline security measures to protect user data, prevent unauthorized access, and secure payment transactions (if applicable).

12. Marketing and Promotion:

- Include features for marketing and promoting the website, such as SEO optimization, email campaigns, and social media sharing.

Project Documentation

13. Scalability and Future Expansion:

- Consider the scalability of the website to accommodate growth in user base and content.
- Plan for future expansion, additional features, and updates.

14. Regulatory Compliance:

- Ensure compliance with relevant educational standards, data protection regulations, and accessibility guidelines.

15. Budget and Timeline:

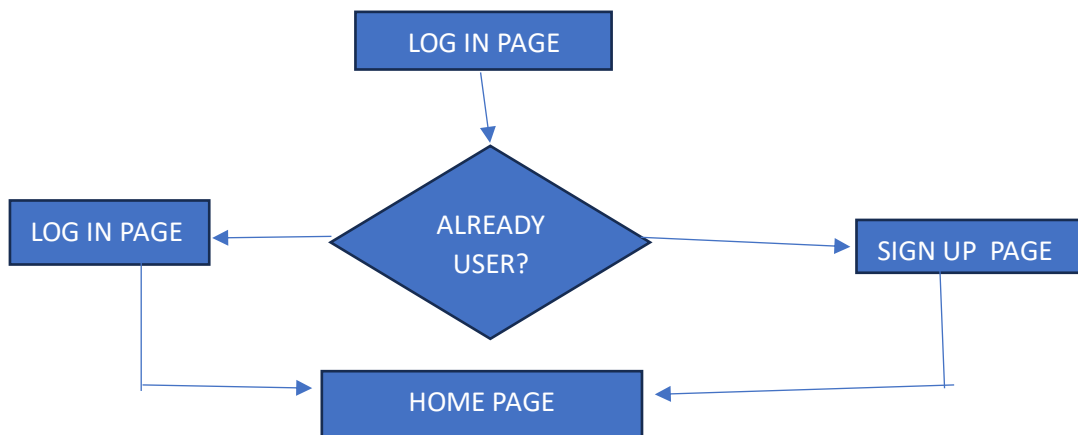
- Define the budget and timeline for the project, including milestones and deliverables.

Clearly defining the scope of the educational website project is crucial for successful planning, execution, and delivery. Regular communication with stakeholders and a well-documented scope will help manage expectations and ensure that the project stays on track.

WORKING OF PROJECT:

1. SIGN IN/SIGN UP: -

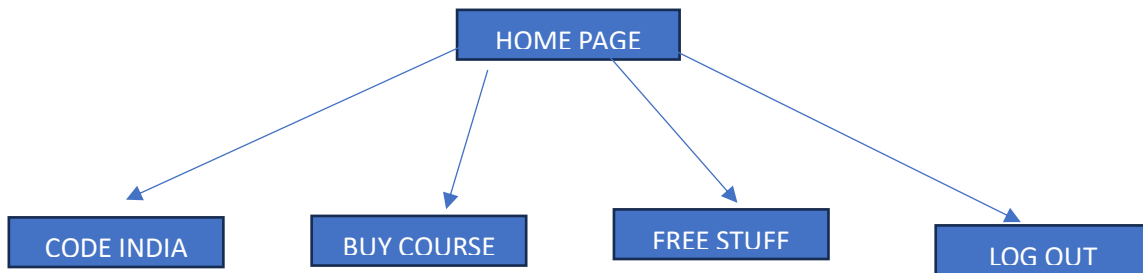
User first interact with login page where two options are required to fill correctly for log in, Username and password. If user is new to the website and does not have a login account then user can click on link for sign up. Then user will get three options which are required to fill for sign up, Username, password and confirm password. After filling the correct inputs, the user can login successfully.



Project Documentation

2.HOME PAGE: -

After login user will get home page on screen. On the top left side user have a hello msg with user name on navigation bar including other options like about, notes, lecture, quiz, profile, code India and log out. Home page have all courses detail and their join link which lead them to the application form. Home page have an introduction summary about Learniverse and also code India link which lead user to code India page where students from class 6th to 10th can learn coding. Home page is accessible by both paid or free user. Home has these features.



3. APPLICATION FORM: -

When a user wants to purchase a certain course then the user click on join button and application form appears on screen. after filling all required fields user will pay the required amount and input the transaction Id then submit the form. The form details will get submitted on database from where the profile page get updated.

4. PAID COURSE: -

Those users who have paid classes will get the mail for zoom/google meet online classes. And those students also get the other features which are not available for other unpaid users.

5. FREE COURSE: -

For those who just login on page but didn't apply for paid course will get summary notes and other free stuff which are actually our marketing strategy plan to get known by everyone.

6. CODE INDIA: -

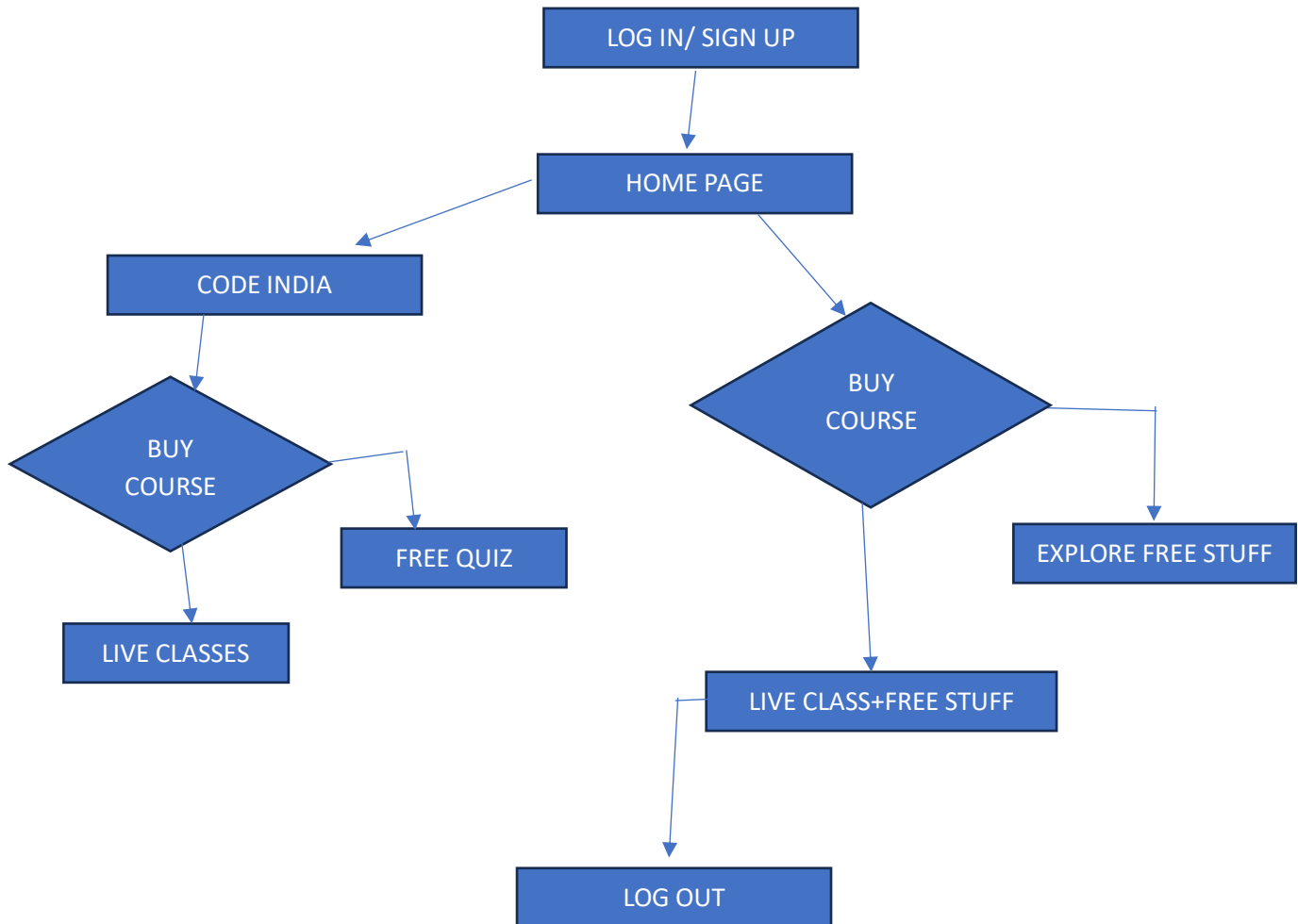
On page there is link for code India from where user can learn coding it also has its quiz and paid courses. It does not have any free stuff other than quiz. It's all courses and notes are paid which are only available to paid user.

Project Documentation

7.LOG OUT: -

On Home page user click on log out then it again transfers to the login page and then if it clicks back then user go to their browser home page. If user will log out then he cannot go to the home page of project.

A rough diagram of our project is:-



Project Documentation

CHAPTER 2: REQUIREMENT AND ANALYSIS

Requirement of project on Educational Website:

- **User Authentication:** Implement secure user registration and login functionalities for students, teachers, and administrators.
- **User Profiles:** Allow users to create and customize profiles, tracking their progress and achievements.
- **Content Management System (CMS):** Create a robust CMS for easy management and update of educational content, including lessons, quizzes, and resources.
- **Course Management:** Organize content into courses, with clear structures, outlines, and objectives.
- **Interactive Learning Materials:** Include multimedia elements like videos, quizzes, and interactive simulations to enhance engagement.
- **Discussion Forums:** Add discussion forums or chat features for students to interact, ask questions, and collaborate on projects.
- **Assessment and Grading:** Implement a system for quizzes, assignments, and exams with automated grading where possible.
- **Progress Tracking:** Provide tools for students to monitor their progress, view grades, and receive feedback.
- **Responsive Design:** Ensure the website is accessible on various devices, including desktops, tablets, and smartphones.
- **Search Functionality:** Include a robust search feature for users to quickly find specific courses, topics, or resources.

Project Documentation

- **Feedback Mechanism:** Allow students to provide feedback on courses, materials, and overall user experience.
- **Accessibility:** Ensure the website complies with accessibility standards to accommodate users with disabilities.
- **Notifications:** Implement a notification system to alert users about new courses, updates, or important announcements.
- **Security Measures:** Prioritize security to protect user data, especially considering the sensitive nature of educational information.
- **Integration with Learning Management Systems (LMS):** If applicable, integrate with existing LMS platforms for a seamless experience.
- **Analytics:** Incorporate analytics tools to track user engagement, popular courses, and other relevant metrics.
- **Responsive Support:** Provide a support system for users to get assistance, whether through FAQs, live chat, or email.
- **Social Integration:** Allow users to share achievements or interesting content on social media platforms.
- **Mobile App Integration:** Consider developing a mobile app to complement the website and provide on-the-go access.
- **Regular Updates:** Plan for regular updates to keep content fresh, fix bugs, and introduce new features based on user feedback.
- **Multi-language Support:** Offer content in multiple languages to cater to a diverse user base.
- **Gamification Elements:** Integrate gamification features like badges, leader boards, and rewards to enhance user motivation.

Project Documentation

- **Virtual Classroom or Webinar Integration:** Provide tools for live classes, webinars, or virtual classrooms for real-time interaction between instructors and students.
- **Personalized Learning Paths:** Enable users to set personalized learning goals and recommend courses based on their preferences and performance.
- **Peer Review System:** Include a feature for peer reviews on assignments or projects to encourage collaboration and constructive feedback.
- **Offline Access:** Allow users to download course materials for offline access, beneficial for those with limited internet connectivity.
- **API Integration:** Support integration with external tools and services, such as payment gateways, analytics tools, or third-party applications.
- **Adaptive Learning:** Implement adaptive learning technologies that adjust content difficulty based on individual user performance.
- **Content Rating and Reviews:** Enable users to rate and review courses, providing valuable insights for prospective learners.
- **Progress Certificates and Badges:** Issue certificates or badges upon course completion, providing a tangible recognition of achievements.
- **Parental Controls:** If the target audience includes minors, implement parental controls to monitor and manage their educational activities.
- **Collaborative Projects:** Facilitate collaborative projects or group assignments to enhance teamwork and communication skills.
- **Real-world Applications:** Integrate real-world examples and case studies into educational materials for practical understanding.

Project Documentation

- **Integration with Social Media Platforms:** Allow users to share their accomplishments directly on social media platforms, extending the reach of the educational content.
- **Customizable Dashboards:** Provide users with the ability to customize their dashboards based on personal preferences and priorities.
- **Career Guidance and Resources:** Include resources for career guidance, job placement, or internship opportunities related to the courses offered.
- **Regular Backups:** Implement a robust backup system to prevent data loss and ensure the availability of content even in unexpected situations.
- **Scalability:** Design the website architecture to handle growth and increased user traffic over time.
- **Legal Compliance:** Ensure compliance with data protection regulations, copyright laws, and other legal requirements relevant to educational content.
- **User Surveys and Feedback Forms:** Periodically gather feedback through surveys and forms to continually improve the website based on user input.

❖ WE WILL USE THE FOLLOWING SQA STRATEGY:

The Software Quality Assurance (SQA) strategy for an educational website typically involves thorough testing and validation to ensure the platform's functionality, performance, and security.

This may include :-

- **Functional Testing:** Ensuring all features work as intended, such as registration, login, course enrollment, and assessment functionalities.
- **Usability Testing:** Assessing the user interface and experience to ensure it's intuitive and user-friendly for both students and educators.

Project Documentation

- **Performance Testing:** Checking the website's responsiveness, loading times, and handling of concurrent users to guarantee optimal performance under various conditions.
- **Security Testing:** Identifying and fixing potential vulnerabilities to protect sensitive data, such as student information and payment details.
- **Compatibility Testing:** Verifying that the website functions correctly on various browsers, devices, and operating systems to accommodate a diverse user base.
- **Accessibility Testing:** Ensuring the website is accessible to users with disabilities, conforming to relevant accessibility standards.
- **Database Testing:** Validating the integrity and reliability of the database, especially regarding data storage, retrieval, and updates.
- **Scalability Testing:** Assessing the website's ability to handle growth in terms of users, courses, and content without compromising performance.
- **Cross-Browser Testing:** Verifying that the website functions consistently across different web browsers.
- **Regression Testing:** Ensuring that new updates or features don't negatively impact existing functionalities.

❖ FEATURES OF THE PROJECT EDUCATIONAL WEBSITE:

- ✓ **User-friendly Interface:** Ensure the website is easy to navigate with clear menus and organized content.
- ✓ **Responsive Design:** Optimize the website for various devices, ensuring a seamless experience on desktops, tablets, and smartphones.

Project Documentation

- ✓ **Content Variety:** Include diverse educational content such as articles, videos, interactive quizzes, and downloadable resources to cater to different learning styles.
- ✓ **Search Functionality:** Implement a robust search feature to help users quickly find specific topics or resources.
- ✓ **Personalized User Accounts:** Allow users to create accounts for personalized experiences, tracking progress, and accessing exclusive content.
- ✓ **Interactive Learning Tools:** Incorporate tools like forums, discussion boards, and live chat to encourage collaboration and interaction among users.
- ✓ **Clear Learning Objectives:** Clearly define learning goals and outcomes for each section or course to guide users through their educational journey.
- ✓ **Progress Tracking:** Provide users with tools to monitor their progress, including completed lessons, quiz scores, and achievements.
- ✓ **Multimedia Integration:** Support various multimedia formats, such as images, videos, and audio, to enhance the learning experience.
- ✓ **Regular Updates:** Keep content up-to-date and relevant to ensure continued engagement and provide the latest information.
- ✓ **Accessibility Features:** Ensure the website is accessible to users with disabilities by following accessibility standards and providing alternative formats for content.
- ✓ **Feedback Mechanism:** Include a feedback system for users to provide comments, suggestions, and report issues, fostering continuous improvement.
- ✓ **Security Measures:** Implement security protocols to protect user data and ensure a safe learning environment.
- ✓ **Social Media Integration:** Allow users to share content easily on social media platforms, increasing the reach and engagement of the educational material.

Project Documentation

Resource Library: Create a comprehensive library of educational resources, categorized and easily accessible for users to explore.

- ✓ **Mobile App Integration:** Consider developing a mobile app for an even more convenient learning experience on smartphones.
- ✓ **Community Building:** Foster a sense of community by incorporating discussion forums, groups, or events where users can connect and collaborate.
- ✓ **Gamification Elements:** Integrate gamification features such as badges, points, or rewards to enhance motivation and engagement.
- ✓ **Responsive Support:** Provide responsive customer support through various channels to assist users with inquiries or technical issues.
- ✓ **Analytics and Reporting:** Implement analytics tools to track user behavior, assess website performance, and gather insights for continuous improvement.

❖ SOFTWARE REQUIREMENT SPECIFICATION:

The Software Requirement Specification is the produced at the culmination of the analysis task. The function and performance allocated to software as part of system as part of system engineering are refined by establishing a complete information description , a detailed functional and behavioural description, an indication of performance requirements and design constraints , appropriate validation criteria , and other data pertinent to requirements.

It is a comprehensive document that outlines the functional and non-functional requirements of a software system. The SRS serves as a blueprint for software developers, detailing what the software should accomplish and how it should perform.

Key components of an SRS include:

- **Introduction:** Overview of the software, its purpose, scope, and objectives.

Project Documentation

- **Functional Requirements:** Detailed description of the system's functionality, including input and output behavior, data processing, and system interactions.
- **Non-Functional Requirements:** Specification of criteria such as performance, security, reliability, and usability that the software must meet.
- **System Architecture:** High-level architecture and design specifications, providing an understanding of the system's structure.
- **User Interfaces:** Description of how users will interact with the system, including screen layouts, navigation, and input methods.
- **System Features:** Breakdown of specific features and functionalities the software will include.
- **System Features:** Breakdown of specific features and functionalities the software will include.
- **External Interfaces:** Integration points with other systems or external components.
- **Performance Requirements:** Criteria related to system response time, throughput, and scalability.
- **Security Requirements:** Measures to ensure the confidentiality, integrity, and availability of data.
- **Quality Attributes:** Criteria related to maintainability, portability, and other aspects affecting the software's overall quality.
- **Constraints :** Limitation or restrictions on the development process , such as budget , time , or technology constraint.

Project Documentation

❖ IDENTIFICATION OF NEED:

The identification of needs for an educational website involves determining the goals, target audience, and desired features. Consider factors like the type of education offered, interactive elements, user engagement, and content delivery methods to meet the specific requirements of learners.

When identifying needs for an educational website, it's crucial to :-

- ✓ **Define Goals:** Clearly outline the primary objectives of the website, whether it's providing courses, resources, or facilitating collaboration.
- ✓
- ✓ **Understand Target Audience:** Identify the demographics, interests, and preferences of the intended users to tailor content and features accordingly.
- ✓ **User-Friendly Design:** Ensure an intuitive and accessible design to enhance user experience, making navigation and interaction seamless.
- ✓ **Content Relevance:** Offer high-quality, relevant educational content that aligns with the target audience's needs and expectations.
- ✓ **Interactivity:** Include interactive elements such as quizzes, discussions, or virtual labs to engage users and reinforce learning.
- ✓ **Accessibility:** Make the website accessible to users with different abilities, ensuring inclusivity in education.
- ✓ **Responsive Design:** Optimize the website for various devices to accommodate users accessing content from different platforms.
- ✓ **Feedback Mechanism:** Implement a feedback system for users to provide comments, suggestions, and report issues, fostering continuous improvement.
- ✓ **Security and Privacy:** Prioritize the security of user data and ensure compliance with privacy regulations to build trust among users.

Project Documentation

- ✓ **Scalability:** Design the website infrastructure to handle growth in users and content over time.

❖ FEASIBILITY STUDY:

A feasibility study for an educational website should assess technical, financial, and operational aspects. Consider factors like market demand, development costs, potential revenue streams, and competitors. Additionally, analyze technical requirements, user experience, and scalability. It's crucial to gauge the viability and sustainability of the project before proceeding.

Key elements in a feasibility study for an educational website:

1) Market Analysis:

- Identify your target audience and assess their needs.
- Analyze competitors and determine your unique selling points.
- Evaluate market trends and potential growth.

2) Technical Feasibility:

- Assess the technical requirements for website development.
- Evaluate the availability of necessary technologies and resources.
- Consider scalability and potential future enhancements.

3) Financial Feasibility:

- Estimate initial development costs, including design, coding, and testing.
- Project ongoing operational costs such as hosting, maintenance, and updates.
- Develop revenue projections and determine the payback period.

4) Operational Feasibility:

- Assess how the website will integrate into existing educational systems.
- Identify potential challenges in terms of content management and user engagement.
- Evaluate the availability of skilled personnel for ongoing operations.

Project Documentation

5) Legal and Regulatory Considerations:

- Ensure compliance with data protection and privacy regulations.
- Identify any legal requirements related to educational content.
- Consider copyright and intellectual property issues.

6) Risk Assessment:

- Identify potential risks and uncertainties associated with the project.
- Develop risk mitigation strategies.
- Evaluate the impact of external factors, such as changes in technology or market conditions.

7) User Experience (UX) and Design:

- Consider the usability and accessibility of the website.
- Plan for a user-friendly interface and navigation.
- Test prototypes with potential users to gather feedback.

8) Marketing and Promotion:

- Develop a marketing strategy to attract users.
- Consider partnerships with educational institutions or influencers.
- Assess the effectiveness of potential marketing channels.

9) Feedback and Iteration:

- Plan for continuous improvement based on user feedback.
- Establish mechanisms for collecting and analyzing user input.
- Iterate on features and content based on user preferences and needs.

A comprehensive feasibility study is a crucial step in ensuring the success of your educational website. It provides a roadmap for development and helps in making informed decisions.

Project Documentation

❖ SYSTEM DESIGN OF EDUCATIONAL WEBSITE:

Designing an educational website involves several key components :-

1) User Authentication:

- Implement secure user registration and login functionalities.
- Include password recovery mechanisms.

2) User Profiles:

- Allow users (students, teachers, administrators) to create and manage their profiles.
- Include features like profile pictures, bio, and contact information.

3) Course Management:

- Enable course creation, editing, and deletion for instructors.
- Include features for adding lectures, quizzes, and assignments.

4) Content Management:

- Support various content types (text, images, videos, documents).
- Implement version control for course materials.

5) Learning Material Delivery:

- Provide a structured and intuitive interface for accessing course content.
- Consider multimedia integration for a more engaging experience.

6) Discussion Forums:

- Include forums for students and instructors to discuss course-related topics.
- Implement moderation features to manage discussions.

Project Documentation

7) Assessment and Grading:

- Build a system for quizzes, exams, and assignments.
- Implement automated grading where applicable.

8) Progress Tracking:

- Include features for students to track their progress.
- Provide analytics for instructors to monitor student performance.

9) Notification System:

- Implement email or in-app notifications for important updates.
- Include configurable notification preferences for users.

10) Search and Navigation:

- Develop a robust search functionality for courses and content.
- Implement clear navigation menus and breadcrumbs.

11) Responsive Design:

- Ensure the website is accessible and user-friendly on various devices.
- Optimize for both desktop and mobile experiences.

12) Security Measures:

- Employ secure coding practices to protect user data.
- Implement HTTPS, data encryption, and secure authentication.

13) Scalability:

- Design the system to handle growth in user base and content.
- Use scalable infrastructure and consider future expansion.

14) Feedback Mechanism:

- Include ways for students to provide feedback on courses.

Project Documentation

- Implement surveys or rating systems.

15) Accessibility:

- Ensure the website complies with accessibility standards.
- Provide features for users with disabilities.

16) Admin Panel:

- Develop a robust admin dashboard for managing users, courses, and system settings.
- Include logs and monitoring for system health.

❖ USER INTERFACE DESIGN:

User Interface Design is concerned with the dialogue between a user and the computer. It is concerned with everything from starting the system or logging into the system to the eventually presentation of desired inputs and outputs. The overall flow of screens and messages is called a dialogue.

Designing a user interface involves creating a visually appealing and intuitive layout for software or websites to enhance user experience. Consider user needs, prioritize simplicity, and maintain consistency for effective UI design.

The following steps are various guidelines for User Interface Design :-

1. **Understand User Needs:** Identify the target audience, their goals, and expectations to tailor the interface to their requirements.
2. **Simple Navigation:** Create an intuitive menu structure and navigation system, ensuring easy access to key sections and content.
3. **Clear Hierarchy:** Organize information hierarchically, with important elements prominently displayed and lesser-used features appropriately nested.

Project Documentation

4. **Consistent Design:** Maintain a consistent design language, including colors, fonts, and layout, to provide a cohesive and professional appearance.
5. **Responsive Design:** Ensure the website is accessible on various devices, adapting to different screen sizes and orientations for a seamless user experience.
6. **Readable Typography:** Use legible fonts, appropriate font sizes, and maintain good contrast between text and background to enhance readability.
7. **Engaging Visuals:** Incorporate relevant images, graphics, and multimedia content to make the learning experience visually appealing and effective.
8. **Interactive Elements:** Implement user-friendly forms, buttons, and interactive features to engage users and facilitate smooth interaction with the site.
9. **Feedback Mechanism:** Provide clear feedback on user actions, such as successful form submissions or error messages, to enhance user understanding and confidence.
10. **Performance Optimization:** Optimize the website's performance to ensure quick loading times, minimizing user frustration and improving overall satisfaction.
11. **Accessibility:** Design with accessibility in mind, adhering to web content accessibility guidelines (WCAG) to accommodate users with diverse needs.
12. **User Testing:** Conduct usability testing with the target audience to identify any pain points, gather feedback, and refine the interface accordingly.
13. **Scalability:** Design a scalable architecture that accommodates future growth in terms of content, users, and features.
14. **Security Measures:** Implement robust security measures to protect user data and ensure a safe learning environment.

Project Documentation

15. **Intuitive Search:** Include a powerful and intuitive search functionality, enabling users to quickly find specific information or resources.
16. **Personalization:** Incorporate features that allow users to customize their experience, such as bookmarking, preferences, or personalized dashboards.
17. **Mobile-Friendly Design:** Prioritize a mobile-friendly design to accommodate the growing number of users accessing educational content on smartphones and tablets.
18. **Compliance:** Ensure compliance with legal and regulatory standards, including data protection laws and educational content requirements.

❖ SOFTWARE REQUIREMENT:

Software requirements define the functionalities, constraints, and qualities that a software system must exhibit. They serve as a foundation for the development process, guiding the design, implementation, and testing phases. These requirements can include features, performance expectations, user interactions, and system constraints, providing a clear roadmap for the development team to follow.

The software requirements for an educational website typically encompass a range of features and functionalities to support effective learning and user interaction. Here are some common software requirements for an educational website:

1. User Authentication:

- Secure user login and registration processes.
- Password recovery mechanisms.

2. User Roles:

- Differentiate between roles such as students, teachers, and administrators.
- Role-specific permissions and access levels.

3. Course Management:

- Ability to create, edit, and manage courses.

Project Documentation

- Support for multimedia content, assignments, and assessments.

4. User Profiles:

- Personalized profiles for students and teachers.
- Tracking progress and achievements.

5. Content Management:

- Upload and organize educational content such as lectures, documents, and multimedia.
- Support for different file formats.

6. Communication:

- Discussion forums, messaging, or chat features.
- Announcements and notifications for updates.

7. Assessment and Grading:

- Online quizzes, assignments, and exams.
- Automated grading and feedback mechanisms.

8. Responsive Design:

- Compatibility with various devices and screen sizes.

9. Search and Navigation:

- Efficient search functionality for courses and content.
- Intuitive navigation to easily find information.

10. Analytics and Reporting:

- Track user engagement, progress, and performance.
- Generate reports for administrators and instructors.

Project Documentation

11. Accessibility:

- Compliance with accessibility standards to ensure inclusivity.

12. Security:

- Measures to protect user data and ensure a secure learning environment.

13. Integration:

- Compatibility with third-party tools or services.
- APIs for potential integrations with other systems.

14. Feedback Mechanism:

- User feedback forms to continuously improve the platform.

15. Compliance:

- Adherence to relevant educational standards and regulations.

These requirements collectively aim to create a robust, user-friendly, and feature-rich educational website that meets the needs of both students and educators.

SOFTWARE REQUIREMENTS

Name of component	Specification
Operating System	Windows 10
Language	HTML,CSS,JS,PHP
Browser	Chrome
Scripting language enable	JavaScript, PHP

Project Documentation

WINDOWS 10:

Windows 10 is a widely used operating system developed by Microsoft. It is the successor to Windows 8.1 and was released in July 2015. Windows 10 is designed to work across a variety of devices, including PCs, tablets, smartphones, and embedded systems. It introduced several new features compared to its predecessors, such as the Cortana virtual assistant, the Microsoft Edge web browser, a revamped Start menu, and the ability to run Universal Windows Platform apps that can adapt to different device types. Windows 10 also includes regular updates and security patches to improve performance and address potential vulnerabilities.

Windows 10 has been well-received for its user interface improvements, performance enhancements, and focus on modern computing trends. It is widely used by consumers, businesses, and organizations around the world.

HTML:

HTML, which stands for HyperText Markup Language, is the standard markup language used to create web pages. It is the basic building block of web development and is used to structure content on the web, such as text, images, videos, and links. HTML uses a system of tags to define the structure and layout of a web page, with each tag representing different elements like headings, paragraphs, lists, links, and more.

HTML documents are interpreted by web browsers to render the content and display it to users. Along with CSS (Cascading Style Sheets) for styling and JavaScript for interactivity, HTML forms the core technologies used to create websites and web applications.

It provides a set of tags that structure the content and layout of a web page, allowing developers to define headings, paragraphs, lists, links, images, and other elements. HTML documents are interpreted by web browsers to display the content to users.

CSS:

CSS, which stands for Cascading Style Sheets, is a style sheet language used to describe the presentation of a document written in HTML or XML (including XML dialects such as SVG or XHTML). It defines how elements are displayed on a screen, in print, or in other media. CSS separates the content of a web page (defined in HTML) from its presentation, allowing developers to control the layout, colors, fonts, and other visual aspects of a website.

CSS works by selecting HTML elements and applying style properties to them. For example, you can use CSS to set the color and size of text, define the layout of a page, create animations, and make the website responsive to different screen sizes.

CSS is an essential part of web development and is used alongside HTML and JavaScript to create visually appealing and interactive web pages.

Project Documentation

JAVASCRIPT:

JavaScript is a high-level programming language that is primarily used to create interactive effects within web browsers. It is one of the core technologies of the World Wide Web, along with HTML and CSS. JavaScript allows developers to add dynamic behavior to their websites, such as responding to user actions, manipulating the DOM (Document Object Model), and making asynchronous requests to web servers.

Key features of JavaScript include:

1. **Client-Side Scripting:** JavaScript code is executed on the client's browser, which allows for dynamic content generation and manipulation without requiring server interaction.
2. **Object-Oriented:** JavaScript is an object-oriented language, which means it uses objects and their properties and methods to organize code.
3. **Event-Driven:** JavaScript is often used to create event-driven applications, where code is executed in response to certain events, such as a user clicking a button or submitting a form.
4. **Asynchronous Programming:** JavaScript supports asynchronous programming through features like callbacks, promises, and `async/await`, allowing developers to write non-blocking code that can handle multiple operations simultaneously.
5. **Cross-Platform:** JavaScript is supported by all major web browsers, making it a cross-platform language for web development.

JavaScript is a versatile language that is not only used for web development but also for server-side development (Node.js), mobile app development (React Native, Ionic), game development, and more.

PHP:

PHP is a server-side scripting language that is widely used for web development. It is especially popular for creating dynamic web pages and web applications. PHP code is executed on the server, generating HTML which is then sent to the client's web browser, allowing for the creation of dynamic and interactive web pages.

Key features of PHP include:

1. **Server-Side Scripting:** PHP is designed for server-side scripting, meaning that it is processed on the web server before the result is sent to the client's browser. This allows for dynamic content generation based on various factors such as user input, database queries, and system variables.
2. **Embedded in HTML:** PHP code can be embedded directly into HTML, making it easy to mix PHP and HTML to create dynamic web pages. PHP code is enclosed in `<?php ?>` tags within an HTML document.

Project Documentation

3. **Database Integration:** PHP has strong support for interacting with databases, allowing developers to easily connect to and manipulate data in databases such as MySQL, PostgreSQL, and others.
4. **Support for Various Protocols:** PHP supports various network protocols, such as HTTP, SMTP, IMAP, and more, which enables it to interact with other services and systems on the web.
5. **Open Source:** PHP is an open-source language, which means it is free to use and has a large community of developers contributing to its growth and improvement. There are also many third-party libraries and frameworks available for PHP that can help streamline development.

PHP is a versatile language that is widely used in web development for tasks ranging from simple website scripting to complex web application development.

CHROME:

Chrome, short for Google Chrome, is a popular web browser developed by Google. It was first released in 2008 and has since become one of the most widely used web browsers, competing with other major browsers like Mozilla Firefox, Microsoft Edge, and Apple Safari.

Key features of Google Chrome include:

1. **Speed and Performance:** Chrome is known for its fast performance, quick startup times, and efficient use of system resources.
2. **Synchronization:** Chrome allows users to sign in with a Google account, which enables synchronization of bookmarks, history, and settings across multiple devices.
3. **Security:** Chrome includes built-in features such as Safe Browsing, which helps protect users from phishing and malware, as well as automatic updates to ensure users have the latest security patches.
4. **Customization:** Chrome supports a wide range of extensions and themes, allowing users to customize their browsing experience with additional features and visual changes.
5. **Developer Tools:** Chrome includes powerful developer tools that help web developers debug and profile their web applications, inspect HTML and CSS, and analyze network performance.
6. **Cross-Platform:** Chrome is available on multiple platforms, including Windows, macOS, Linux, Android, and iOS, providing a consistent browsing experience across devices.

Google Chrome has a large market share in the web browser market due to its speed, simplicity, and integration with other Google services. It is also known for its support of modern web standards and its focus on providing a secure and stable browsing experience.

Project Documentation

XAMPP:

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends. The name "XAMPP" is an acronym, and it stands for:

X: Cross-platform (available on multiple operating systems)

A: Apache (web server software)

M: MySQL (database management system)

P: PHP (scripting language)

P: Perl (programming language)

XAMPP provides a convenient way to set up a local server environment on your computer, allowing you to develop and test web applications offline. It includes the Apache web server, MySQL database, PHP, and Perl. This software package is commonly used by developers for web development and testing before deploying applications to a live server.

By installing XAMPP, you can create a local server on your computer, which enables you to run and test dynamic websites and web applications without the need for internet connectivity. It's popular among developers working with PHP and MySQL-based applications.

❖ **HARDWARE REQUIREMENT:**

The hardware requirements for an educational website depend on factors such as the anticipated traffic, content complexity, and functionality. Generally, you'll need:

- **Web Server:** A powerful server to host and serve web pages efficiently. Consider factors like processor speed, RAM, and storage capacity based on expected traffic.
- **Database Server:** If your website involves user accounts, courses, or other dynamic content, a robust database server is essential. MySQL, PostgreSQL, or similar databases are common choices.
- **Storage:** Ensure sufficient storage for website files, multimedia content, and database storage. SSDs can improve data retrieval speed.

Project Documentation

- **Processor:** A multi-core processor can handle concurrent requests and computations more effectively.
- **Memory (RAM):** Sufficient RAM is crucial for smooth performance, especially if your website deals with concurrent users or resource-intensive operations.
- **Network Infrastructure:** A reliable and high-speed internet connection to handle user requests and data transfers efficiently.
- **Load Balancer:** For scalability and distribution of incoming traffic across multiple servers, especially if your website grows in popularity.
- **Backup Systems:** Implement regular backups to prevent data loss in case of server failures or other issues.
- **Security Measures:** Firewalls, SSL certificates, and other security protocols to safeguard user data and protect against cyber threats.
- **Content Delivery Network (CDN):** To optimize content delivery, especially if your audience is distributed globally.

It's essential to scale your hardware based on your website's growth and user demands. Regular monitoring and performance optimization are key to maintaining a reliable educational website.

Project Documentation

HARDWARE REQUIREMENT

Name of component	Specification
Processor	Intel Core i5 10 th Gen
RAM	128MB
Hard disk	20MB
monitor	15'' color monitor
Keyboard	122 keys

❖ SYSTEM ANALYSIS:

System analysis is the process of studying, understanding, and documenting the requirements and functionalities of a system in order to design a solution that meets the specified needs. It is a crucial phase in the development of information systems, software, or business processes.

System analysis is an iterative process, and the results from this phase guide the subsequent stages of system design, implementation, and testing. It plays a crucial role in ensuring that the developed system effectively addresses the identified needs and delivers value to the organization or users.

The primary objectives of system analysis include:

- **Understanding Requirements:** Identify and comprehend the needs and expectations of users, stakeholders, and the organization. This involves gathering information about the existing system (if any) and understanding the problems or opportunities that necessitate a new solution.
- **Defining Goals and Objectives:** Clearly articulate the goals and objectives that the new system or solution is expected to achieve. This helps in setting the direction for the development process.

Project Documentation

- **Stakeholder Involvement:** Engage with various stakeholders, including end-users, managers, and IT professionals, to ensure that their perspectives and requirements are considered throughout the analysis process.
- **Data Collection:** Collect relevant data and information about the processes, data flow, and functionalities of the existing system or the system to be developed. This involves interviews, surveys, observations, and document reviews.
- **Modeling:** Create models or visual representations that illustrate the system's components, processes, data flow, and interactions. Common modeling tools include flowcharts, data flow diagrams, entity-relationship diagrams, and use case diagrams.
- **Feasibility Study:** Evaluate the feasibility of implementing the proposed system. This includes assessing technical feasibility, operational feasibility, economic feasibility, and legal or regulatory feasibility.
- **Risk Analysis:** Identify potential risks and challenges that may arise during the development and implementation of the system. Develop strategies to mitigate these risks.
- **Requirements Specification:** Document detailed system requirements, including functional and non-functional requirements. This serves as the foundation for the design and development phases.
- **Prototyping:** In some cases, create prototypes or mock-ups of the proposed system to give stakeholders a tangible representation of how the final system will look and function.

System Designing an educational website involves several components:

- **User Authentication:** Implement secure user registration and login systems to manage user access.
- **User Profiles:** Allow users to create and manage their profiles, including personal information and preferences.

Project Documentation

- **Course Management:** Develop a system to add, organize, and manage courses. Include features like course descriptions, syllabus, and multimedia content.
- **Content Management System (CMS):** Create an admin interface for easy content updates, including text, images, and videos.
- **Search Functionality:** Implement a robust search feature to help users find courses, materials, and other relevant content easily.
- **Enrollment System:** Design a mechanism for users to enroll in courses, with notifications and progress tracking.
- **Discussion Forums:** Integrate discussion forums or chat features for students and instructors to interact, ask questions, and share insights.
- **Assessment and Grading:** Build tools for quizzes, assignments, and exams. Include features for automatic grading and result tracking.
- **Notifications:** Implement a notification system to update users on course changes, upcoming deadlines, and other relevant information.
- **Responsive Design:** Ensure the website is accessible and user-friendly across various devices, including desktops, tablets, and smartphones.
- **Payment Integration:** If applicable, integrate secure payment gateways for course enrollment and other premium features.
- **Analytics:** Include analytics tools to track user engagement, course popularity, and other relevant metrics.
- **Accessibility:** Design the website to be accessible to users with disabilities, adhering to accessibility standards.
- **Security Measures:** Implement robust security measures to protect user data, prevent unauthorized access, and ensure the integrity of the platform.

Project Documentation

- **Feedback Mechanism:** Provide a way for users to provide feedback on courses and the overall platform.

❖ PROJECT PROFILE:

Project profile of educational website :

1. Project Overview:

- **Objective:** Develop an interactive and user-friendly educational website to facilitate online learning.
- **Target Audience:** Students, educators, and anyone seeking quality educational content.
- **Key Features:** User registration, course navigation, multimedia resources, discussion forums, and assessment tools.

2. Scope:

- **Content:** Varied subjects, courses, and learning materials.
- **Interactivity:** Engaging quizzes, assignments, and interactive discussions.
- **Platforms:** Responsive design for desktop and mobile accessibility.

3. Technical Specifications:

- **Technology Stack:** HTML, CSS, JavaScript for frontend; Php or Node.js for backend; Database (e.g., MySQL, MongoDB).
- **Hosting:** Cloud-based hosting for scalability.
- **Security:** SSL encryption, user authentication protocols.

4. Key Modules:

- **User Registration:** Secure sign-up and login functionality.
- **Course Management:** Admin panel for adding, updating, and organizing courses.
- **Multimedia Integration:** Videos, images, and documents to enhance learning.
- **Discussion Forums:** Platform for students and educators to interact.

Project Documentation

5. User Experience (UX):

- **Intuitive Design:** User-friendly interface for easy navigation.
- **Responsive:** Ensure seamless experience across devices.
- **Feedback Mechanism:** Surveys, feedback forms for continuous improvement.

6. Assessment and Progress Tracking:

- **Quizzes and Tests:** Integrated assessments for course evaluation.
- **Progress Tracker:** Visual representation of user progress within courses.

7. Community Building:

- **Discussion Forums:** Foster a sense of community and collaborative learning.
- **User Profiles:** Customizable profiles to showcase achievements.

8. Marketing and Outreach:

- **SEO Optimization:** Ensure high visibility on search engines.
- **Social Media Integration:** Shareable content and announcements.

9. Maintenance and Updates:

- **Regular Backups:** Ensure data integrity.
- **Software Updates:** Keep the platform up-to-date with the latest technologies.

10. Timeline:

- **Milestones:** Design, Development, Testing, Deployment, Maintenance.
- **Estimated Completion:** [Specify timeline].

11. Budget:

- **Development Costs:** Including software, hosting, and licensing.
- **Marketing and Outreach:** Budget for promotions.

Project Documentation

12. Risks and Mitigation:

- **Technical Challenges:** Regular testing and debugging.
- **User Engagement:** Continuous improvement based on user feedback.

13. Conclusion:

- **Impact Assessment:** Measure the success based on user engagement, feedback, and learning outcomes.

PLANNING:

DETAILS ABOUT PROJECT:

- Online tuition provider
- More interaction with tuition via live classes
- Meeting with parents and giving them feedback report of student
- Live classes through sharing screen, ppt, diagrams and etc.
- Special classes for doubt
- No extra fees for doubt session
- Paid classes
- 9th & 10th
- Academic syllabus, quiz, notes and doubt discussion
- Home, about, login, join-form, quiz and notes
- Payment done via UPI, we know gateway concept but currently using transaction id for verification of transaction.
- Link provided on email
- Connect through social media
- Add faculty details
- Learniverse, universe of learning
- Logo
- Pic for subjects, about page(pic of team member), social media logo icon, bg-img for notes page

Project Documentation

- Color is mainly white and rgb(4,4,88)
- backgroundColor for navigation, Fontcolor for text , footer, faculty-box, subject-box, button
- Code learning platform for students of class 6 to 12.
- Have course of programming languages.
- Free quiz on code India platform.
- Learniverse have 3 major parts, academics, coding and competitive exams.
- Free lecture available for user
- All user who login can use free stuff and buy paid course.
- Coding course is for class 6th to 10th and academics/ competition are for class 9th and 10th.

Project Documentation

CHAPTER 3: SYSTEM DESIGN

Designing an educational website involves considering various aspects such as user experience, functionality, scalability, and security.

Below is a high-level overview of the system design for an educational website:

1. Requirements Analysis:

- Identify the target audience (students, teachers, administrators).
- Define the features and functionalities (courses, quizzes, forums, user profiles).
- Determine the types of content (text, video, quizzes, assignments).

2. Architecture:

- Choose a suitable architecture (monolithic, microservices) based on scalability requirements.
- Use a layered architecture (presentation layer, business logic layer, data access layer).
- Implement a RESTful API for communication between different components.

3. User Authentication and Authorization:

- Implement secure user authentication (username/password, social login).
- Use role-based access control to manage user permissions (student, teacher, admin).

4. Database Design:

- Choose an appropriate database system (SQL or NoSQL) based on data requirements.
- Design a database schema to store user data, course content, user progress, etc.
- Optimize database queries for performance.

Project Documentation

5. Content Management:

- Implement a content management system for courses, lectures, and multimedia content.
- Allow users to upload, organize, and manage course materials.

6. User Interface (UI) Design:

- Design an intuitive and user-friendly interface.
- Use responsive design for mobile compatibility.
- Incorporate accessibility features for users with disabilities.

7. Search Functionality:

- Implement a robust search engine to help users find courses and content.
- Include filters and sorting options for search results.

8. Communication and Collaboration:

- Integrate discussion forums, chat, or messaging features for communication.
- Implement collaborative tools for group projects or discussions.

9. Progress Tracking:

- Provide features to track user progress, such as completed courses, grades, and achievements.
- Generate reports and analytics for administrators and teachers.

10. Payment Integration (if applicable):

- If the website offers paid courses, integrate a secure payment gateway.
- Implement subscription models or one-time payments.

11. Security:

- Implement SSL/TLS for secure data transfer.
- Regularly update and patch software to address security vulnerabilities.
- Use encryption for sensitive data.

Project Documentation

12. Scalability:

- Design the system to handle a growing number of users and content.
- Consider load balancing and caching strategies.

13. Testing:

- Perform thorough testing, including unit testing, integration testing, and user acceptance testing.
- Test for security vulnerabilities, such as SQL injection and cross-site scripting.

14. Deployment:

- Choose a reliable hosting platform.
- Set up continuous integration and deployment pipelines.
- Monitor the system for performance and security.

15. Maintenance and Updates:

- Regularly update software components to patch security vulnerabilities.
- Provide user support and address feedback.
- Plan for future enhancements and features.

Remember that this is a high-level overview, and the specific details will depend on the unique requirements of your educational website. It's also crucial to involve stakeholders, including educators and students, in the design process to ensure the platform meets their needs.

DATA DICTIONARY:

DEFINITION OF DATA DICTIONARY:

A data dictionary is a comprehensive document that defines and describes the data elements used in a system. For an educational website, the data dictionary might include information about entities such as users, courses, quizzes, and other relevant components.

Project Documentation

A data dictionary is a document that provides detailed information about the data elements used in a database or system. For an educational website, the data dictionary would include definitions and descriptions of various entities and attributes.

A data dictionary is a centralized repository of information about data elements used in a database or information system. It serves as a reference for both technical and non-technical users to understand the structure, meaning, and usage of data within a particular context. The data dictionary provides a comprehensive and organized description of data, including metadata such as data types, relationships, constraints, and other attributes.

Key components of a data dictionary typically include:

1. Data Element Names:

- The names or labels of individual data elements used in a database or system.

2. Data Types:

- The type of data that each data element can hold (e.g., integer, varchar, date).

3. Field Sizes:

- The maximum length or size of each data element, particularly relevant for character or string data types.

4. Descriptions:

- Detailed explanations or definitions of the purpose and meaning of each data element.

5. Valid Values:

- The allowable or expected values for each data element, including any constraints or limitations.

6. Relationships:

- Information about how different data elements or tables are related to each other, including foreign key relationships.

Project Documentation

7. Constraints:

- Any rules or restrictions imposed on the data, such as uniqueness constraints, primary key constraints, or check constraints.

8. Usage Notes:

- Additional information or notes about the usage, business rules, or specific considerations for each data element.

9. Metadata:

- Information about the data, including creation date, modification date, and ownership details.

10. Data Source:

- If applicable, information about the source of the data, such as the system or process that generates or updates it.

The primary purposes of a data dictionary include promoting consistency, ensuring data quality, facilitating collaboration among team members, aiding in system development, and providing documentation for database administrators, developers, and other stakeholders. It serves as a valuable reference for understanding the structure and meaning of data elements within a given system or database.

IMPORTANCE OF DATA DICTIONARY IN EDUCATIONAL WEBSITE:

In the context of an educational website, a data dictionary plays a crucial role in ensuring effective management, understanding, and utilization of the data within the system. Here are several reasons highlighting the importance of a data dictionary in the context of an educational website:

1. Clarity and Understanding:

- Provides a clear and concise explanation of data elements, helping both technical and non-technical stakeholders understand the purpose and meaning of each piece of data.

Project Documentation

2. Consistency and Standardization:

Promotes consistency in data usage by establishing standardized definitions, data types, and constraints. This ensures uniformity across the entire system.

3. Communication and Collaboration:

- Facilitates communication and collaboration among team members, including developers, database administrators, educators, and administrators, by providing a common reference point for discussing data-related concepts.

4. System Development:

- Aids in the development and maintenance of the educational website by serving as a guide for database design, helping developers understand the structure of the database, relationships between tables, and constraints on data.

5. Data Quality and Integrity:

- Helps maintain data quality by documenting validation rules, constraints, and acceptable values. This ensures that data entered into the system is accurate, consistent, and adheres to predefined standards.

6. Efficient Data Management:

- Streamlines data management tasks by offering a centralized repository of information about the data. This is particularly important in complex educational systems with multiple databases, tables, and data sources.

7. Training and Onboarding:

- Assists in the training and onboarding of new team members, providing them with a comprehensive guide to understand the data structures and relationships within the educational website.

8. Data Governance:

- Supports data governance efforts by documenting ownership, responsibilities, and stewardship of data elements. This helps in establishing accountability for data quality and usage.

Project Documentation

9. Facilitates Reporting and Analytics:

- Enhances reporting and analytics capabilities by providing insights into the structure and meaning of data. This is essential for generating accurate and meaningful reports for educational administrators and stakeholders.

10. Documentation for Audits and Compliance:

- Acts as documentation for audits and compliance purposes, helping educational institutions adhere to regulatory requirements and standards related to data management.

11. Adaptability and Scalability:

- Facilitates the adaptability and scalability of the educational website by providing a roadmap for future changes or enhancements to the data model. This is particularly important as the website grows and evolves.

In summary, a data dictionary is instrumental in fostering a well-organized, efficient, and understandable data environment within an educational website, contributing to the overall success and effectiveness of the system.

ADVANTAGE OF DATA DICTIONARY:

A data dictionary provides numerous advantages for organizations in managing and understanding their data. Here are some key benefits:

1. **Data Clarity and Understanding:** Enhances clarity and understanding of data elements.
2. **Consistency and Standardization:** Promotes consistency and standardization in data usage.
3. **Communication and Collaboration:** Facilitates communication and collaboration among team members.
4. **Efficient Data Management:** Streamlines data management tasks.

Project Documentation

5. **System Development and Maintenance:** Aids in system development and maintenance.
6. **Data Quality and Integrity:** Supports data quality and integrity.
7. **Training and Onboarding:** Assists in training and onboarding processes.
8. **Data Governance:** Supports data governance efforts.
9. **Facilitates Reporting and Analytics:** Enhances reporting and analytics capabilities.
10. **Documentation for Audits and Compliance:** Serves as documentation for audits and compliance.
11. **Adaptability and Scalability:** Facilitates adaptability and scalability.
12. **Time and Cost Savings:** Leads to time and cost savings.

A data dictionary is a valuable tool that brings transparency, consistency, and efficiency to data management processes, ultimately contributing to better decision-making and the overall success of an organization.

Project Documentation

CHAPTER 4: TESTING

THE STEPS IN THE SOFTWARE TESTING:

1. **Requirements Analysis:** Understand the specifications and requirements of the educational website to determine the scope of testing.
2. **Test Planning:** Develop a comprehensive test plan outlining testing objectives, scope, resources, schedule, and testing methodologies.
3. **Test Case Design:** Create test cases based on functional and non-functional requirements, covering different scenarios, user interactions, and edge cases.
4. **Test Environment Setup:** Configure a testing environment that mirrors the production environment to ensure realistic testing conditions.
5. **Unit Testing:** Verify individual components/modules of the educational website to ensure they function as intended.
6. **Integration Testing:** Test the interaction between different components/modules to identify and address any issues arising from their integration.
7. **System Testing:** Conduct end-to-end testing of the entire system to evaluate its overall functionality, performance, and security.
8. **User Acceptance Testing (UAT):** Allow end-users or stakeholders to validate that the educational website meets their expectations and requirements.
9. **Performance Testing:** Assess the system's responsiveness, scalability, and stability under varying loads to ensure it can handle concurrent users.
10. **Security Testing:** Evaluate the website's security measures to identify and address potential vulnerabilities, protecting against unauthorized access and data breaches.

Project Documentation

11. **Compatibility Testing:** Ensure the website is compatible with different browsers, devices, and operating systems to provide a consistent user experience.
12. **Usability Testing:** Evaluate the website's user interface and overall user experience to ensure it is intuitive and user-friendly.
13. **Accessibility Testing:** Verify that the educational website complies with accessibility standards, ensuring it can be used by individuals with disabilities.
14. **Regression Testing:** After each modification or update, re-run relevant tests to ensure that existing functionalities have not been negatively impacted.
15. **Defect Reporting And Tracking:** Document and prioritize any issues or defects found during testing and track their resolution.
16. **Documentation:** Maintain comprehensive documentation, including test plans, test cases, and test results, to facilitate future testing efforts and system maintenance.
17. **Training:** Provide training to end-users or support teams on the proper usage of the educational website and address any common issues that may arise.
18. **Continuous Improvement:** Gather feedback from testing phases and use it to improve the development and testing processes for future releases.

❖ THE UNIT TESTING DONE INCLUDE THE TESTING OF THE FOLLOWING ITEMS :-

When unit testing an educational website, specific considerations and test cases can help ensure the functionality, usability, and reliability of the platform.

Here are some items to include in unit testing for an educational website:

Project Documentation

1. User Authentication:

- Test user registration and login processes.
- Verify password recovery/reset functionality.

2. User Profiles:

- Test profile creation and editing.
- Validate the display of user information.

3. Course Management:

- Test course creation, modification, and deletion.
- Ensure proper enrollment and unenrollment of students in courses.

4. Content Delivery:

- Test the display of educational content (text, images, videos, etc.).
- Verify proper functioning of multimedia elements.

5. Assessments:

- Test quizzes, exams, and other assessment tools.
- Check scoring and feedback mechanisms.

6. Discussion Forums:

- Test the creation and management of discussion forums.
- Ensure proper posting, editing, and deletion of forum threads and replies.

7. Progress Tracking:

- Validate the tracking of student progress within courses.
- Test the accuracy of completion status.

8. Notifications:

- Verify that users receive relevant notifications (e.g., announcements, reminders).

Project Documentation

- Test the unsubscribe/opt-out functionality for notifications.

9. Search Functionality:

- Test the search feature for courses, content, and users.
- Ensure accurate and relevant search results.

10. Responsive Design:

- Test the website's responsiveness on various devices and screen sizes.
- Ensure usability on both desktop and mobile platforms.

11. Security:

- Verify that user data is securely stored and transmitted.
- Test for vulnerabilities, such as SQL injection or cross-site scripting.

12. Accessibility:

- Check for compliance with accessibility standards (WCAG) to ensure the website is usable for all users, including those with disabilities.

13. Browser Compatibility:

- Test the website on different browsers (Chrome, Firefox, Safari, Edge) to ensure consistent performance.

14. Performance:

- Assess the website's loading times, especially for content-heavy pages.
- Test under various network conditions to ensure usability.

15. Feedback Mechanisms:

- Validate the effectiveness of feedback forms.
- Test error messages for clarity and helpfulness.

By Incorporating these elements into unit testing, you can identify and address issues early in the development process, ensuring a reliable and user-friendly educational website.

Project Documentation

❖ THE SYSTEM TESTING DONE INCLUDE THE TESTING OF THE FOLLOWING ITEMS :-

System testing for an educational website typically involves checking various components and functionalities to ensure they work as intended. Some items to include in system testing for an educational website are:

1. User Registration And Authentication:

- Verify that users can register successfully.
- Ensure secure authentication processes.

2. User Interface (UI) Testing:

- Check the overall design and layout.
- Test navigation and user interaction.

3. Content Display:

- Confirm that educational content is displayed correctly.
- Ensure multimedia elements (videos, images) work properly.

4. Course Enrollment And Management:

- Test the process of enrolling in courses.
- Verify that users can access and manage enrolled courses.

5. Assessment And Grading:

- Test quizzes, exams, and assignments.
- Ensure accurate grading and feedback.

6. Communication Features:

- Verify messaging systems and discussion forums.
- Check notifications for announcements or updates.

Project Documentation

7. Search Functionality:

- Confirm that the search feature works effectively.
- Ensure relevant results are displayed.

8. Performance And Load Testing:

- Test the website's performance under various loads.
- Check response times and server stability.

9. Browser Compatibility:

- Ensure the website functions correctly on different browsers.
- Verify compatibility with various devices.

10. Security Testing:

- Check for vulnerabilities and ensure data privacy.
- Test against common security threats.

11. Accessibility Testing:

- Verify that the website is accessible to users with disabilities.
- Ensure compliance with accessibility standards.

12. Payment And Transaction Processing:

- If applicable, test payment processes for paid courses.
- Confirm transaction security.

13. Feedback And Reporting:

- Test user feedback forms.
- Ensure reporting features are functional.

14. System Integration:

Project Documentation

- Verify integration with other systems if applicable.
- Check for seamless data flow between components.

15. Backup And Recovery:

- Test backup and recovery mechanisms.
- Ensure data integrity and restoration processes.

Project Documentation

Chapter 5: Source Code

ABOUT.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <!-- css link -->
  <link rel="stylesheet" href="/style3.css">
  <!-- font stle link -->
  <link rel="preconnect" href="https://fonts.googleapis.com">
  <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
  <link href="https://fonts.googleapis.com/css2?family=Poppins&display=swap"
rel="stylesheet">
  <title>About_Zoom_Tuition</title>
</head>
<body>
  <!-- heading -->
  <div class="our_about">
    <div class="circle">
      <p>About us</p>
    </div>
  </div>
  <!-- title and logo -->
  <div class="title_about">
    
    <div class="body_about">
      <span>Who Are We?</span>
      <br>
      <p> We are your one-stop destination for online tuition classes that fit your needs
and schedule. Our platform is designed to make it easy for you to connect with expert tutors
via Zoom, so you can get the help you need to excel in your studies. <br> Whether you're
```

Project Documentation

preparing for an exam, working on a project, or just need extra support, we're here to help you achieve your goals.</p>

<p>

We believe that every student deserves access to the best possible education, no matter where they are located. That's why we've created a cutting-edge online platform that connects you with top-rated tutors via Zoom. Whether you need help with math, science, or writing, our team of experienced educators is here to help.

</p>

</div>

</div>

<!-- values -->

<div class="circle">

<p> Our Values</p>

</div>

<div class="values">

<div class="val">

Learn

<p>"Learning opens door with endless growth."</p>

<p>Learning in a fun and engaging way can make the process more enjoyable and effective. Learning is the foundation of progress, personal development</p>

</div>

<div class="val">

Practice

<p>"More you practice, the better you become."</p>

<p>Practice is essential for exam preparation. It involves working through practice tests and sample questions to become familiar with the format and content of exams. </p>

</div>

<div class="val">

Achieve

<p>"Set your goal in mind to achieve that."</p>

Project Documentation

<p>Achievements signify personal growth and progress. Achievements serve as powerful motivators. When students set and achieve academic or personal goals, it fuels their motivation to continue learning and striving for success.</p>

</div>

</div>

<!-- teams -->

<!-- Team heading starts -->

<div >

<div class="circle">

<p>Our Team</p>

</div>

<!-- Team heading ends -->

<!-- Teachers details starts -->

<div class="team_det1">

<div class="teacher">

<p> </p>

<p class="name">Ritika Rani</p>

<p>Faculty for Mathematics </p>

<p>BCA From L.N.D. College</p>

<p>12th From C.B.S.E.</p>

<p>10th From C.B.S.E.</p>

</div>

<div class="teacher">

<p> </p>

<p class="name">Vishal Kumar</p>

<p>Faculty for Science </p>

<p>BCA From L.N.D. College</p>

<p>12th From B.S.E.B</p>

<p>10th From C.B.S.E.</p>

</div>

Project Documentation

```
<div class="teacher">
  <p> </p>
  <p class="name">Aliya Naz</p>
  <p>Faculty for <b> Social Science </b></p>
  <p>BCA From L.N.D. College</p>
  <p>12th From B.S.E.B.</p>
  <p>10th From B.S.E.B. </p>
</div>
</div>
<div class="team_det1">
  <div class="teacher">
    <p> </p>
    <p class="name">Sakshi </p>
    <p>Faculty for <b> English </b></p>
    <p>BCA From L.N.D. College</p>
    <p>12th From C.B.S.E.</p>
    <p>10th From C.B.S.E. </p>
  </div>
  <div class="teacher">
    <p> </p>
    <p class="name">Adarsh Kumar</p>
    <p>Faculty for <b> Computer Science </b></p>
    <p>BCA From L.N.D. College </p>
    <p>12th From B.S.E.B.</p>
    <p>10th From C.B.S.E. </p>
  </div>
</div>
</div>
<!-- Team details ends -->
<!-- our gallery starts -->
<div >
```

Project Documentation

```
<div class="circle">
  <p>Gallery</p>
</div>
<div>
  <!-- <div class="gallery"> -->
    
    
    
    
    
    
  <!-- </div> -->
</div>
<!-- Footer starts -->
<footer>
  <div class="last_part">
    <p>
      Address: <br>
      Head office: Learniverse, Hospital Chowk,<br>
      Motihari, East Champaran, Bihar <br> <br>
    </p>
    <p>
      Pincode: 845401 <br> <br>
    </p>
    <p>
      Email:
      info@LVerse.com <br>
      Phone Number: 06252 235351
    </p>
  </div>
</div>
```

Project Documentation

```
<p style="font-size: 15px; padding-top: 300px;"> &copy 2023 Learniverse private limited</p>
```

```
<div class="footer_contact">
```

```
  
we_r_learniverse <br>
```

```
  
we_r_lv <br>
```

```
  
info@lverse.com <br>
```

```
  
learniverse
```

```
</div>
```

```
</footer>
```

```
</body>
```

```
</html>
```

APPFORM.PHP

```
<?php
```

```
// Replace with your database credentials
```

```
$servername = "localhost";
```

```
$username = "root";
```

```
$password = "";
```

```
$dbname = "lvdata";
```

```
// Create a connection
```

```
$conn = new mysqli($servername, $username, $password, $dbname);
```

```
// Check the connection
```

```
if ($conn->connect_error) {
```

```
    die("Connection failed: " . $conn->connect_error);
```

```
}
```

```
// Check if the form is submitted
```

```
if ($_SERVER["REQUEST_METHOD"] == "POST") {
```

Project Documentation

```
// Retrieve form data
$username = $_POST["username"];
$fname = $_POST["fname"];
$mname = $_POST["mname"];
$email = $_POST["email"];
$adhar = $_POST["adhar"];
$mob = $_POST["mob"];
$dob = $_POST["dob"];
$userclass = $_POST["userclass"];
$usersub = $_POST["usersub"];
$street = $_POST["street"];
$city = $_POST["city"];
$pincode = $_POST["pincode"];
$district = $_POST["district"];
$state = $_POST["state"];
$transaction = $_POST["transaction"];

// Insert data into the database

$sql = "INSERT INTO appformtable (username,fname,mname, email, adhar, mob, dob,
userclass, usersub, street, city, pincode, district, state) VALUES
('$username','$fname','$mname','$email','$adhar','$mob','$dob','$userclass','$usersub','$street','$city','$pincode','$district','$state)";

if ($conn->query($sql) === TRUE) {
    echo "Record added successfully. We will verify all details. After few hours you will get
invoice through mail. We will connect to you via mail so make sure you visit your mailbox
daily.";
} else {
    echo "Error: " . $sql . "<br>" . $conn->error;
}
}
```

Project Documentation

```
// Close the connection
```

```
$conn->close();
```

```
?>
```

APPL.HTML

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
  <title>Form_batch</title>
```

```
  <link rel="stylesheet" href="./forml.css">
```

```
</head>
```

```
<body>
```

```
  <div class="heading">
```

```
    Application Form
```

```
  </div>
```

```
  <div class="form-container">
```

```
    <form action="appform.php" method="post">
```

```
      <fieldset><table>
```

```
        <legend >Personal Detail</legend>
```

```
        <!-- <table> -->
```

```
          <tr>
```

```
            <td> <label for="username">Name: </label></td>
```

```
            <td><input type="text" name="username" id="username" placeholder="UserName" required></td>
```

```
          </tr>
```

```
          <tr>
```

```
            <td><label for="fname">Father's Name: </label></td>
```


Project Documentation

```
<td><input type="text" name="fname" id="username" placeholder="Father Name"
required></td>
</tr>
<tr>
<td><label for="mname">Mother's Name: </label></td>
<td><input type="text" name="mname" id="username" placeholder="Mother Name"
required></td>
</tr>
<tr>
<td><label for="email">E-mail: </label></td>
<td><input type="email" name="email" id="mail" placeholder="eg: abc@def.com"
required></td>
</tr>
<!-- </table> -->
<!-- <table> -->
<tr>
<td><label for="adhar">Adhar Number: </label></td>
<td><input type="number" name="adhar" id="adhar" placeholder="Adhar No"
required></td>
</tr>
<tr>
<td> <label for="mob">Mobile No.: </label></td>
<td><input type="number" name="mob" id="mob" placeholder="eg: 9988776655"
required></td>
</tr>
<tr>
<td><label for="dob">Date of birth:</label></td>
<td><input type="date" name="dob" id="dob" placeholder="DOB" required></td>
</tr>
</table>
</fieldset>
<br>
```

Project Documentation

```
<fieldset>
  <legend>Course Detail</legend>
  <table>
    <tr>
      <td><label for="userclass">Class: </label></td>
      <td><select name="userclass" id="userclass">
        <option value="9">9th</option>
        <option value="10">10th</option>
      </select></td>
    </tr>
    <tr>
      <td><label for="usersub">Subject: </label> </td>
      <td><select name="usersub" id="usersub">
        <option value="science">Science</option>
        <option value="socialsci">Social Science</option>
        <option value="computersci">Computer Science</option>
        <option value="maths">Maths</option>
        <option value="english">English</option>
      </select></td>
    </tr>
  </table>
</fieldset>
<br>
<fieldset>
  <legend>Residential Detail</legend>
  <table>
    <tr>
      <td><label for="street">Street: </label></td>
      <td><input type="text" name="street" id="street" placeholder="Street"
required></td>
    </tr>
```

Project Documentation

```
<tr>
  <td><label for="city">City: </label></td>
  <td><input type="text" name="city" id="city" placeholder="City" required></td>
</tr>
<tr>
  <td><label for="pincode">Pincode: </label></td>
  <td><input type="number" name="pincode" id="pincode" placeholder="123456"
required></td>
</tr>
<tr>
  <td><label for="district">District: </label></td>
  <td><input type="text" name="district" id="district" placeholder="District"
required></td>
</tr>
<tr>
  <td><label for="state">State: </label></td>
  <td><input type="text" name="state" id="state" placeholder="State" required></td>
</tr>
</table>
</fieldset>
<br>
<fieldset>
  <legend>Payment</legend>
  <br>
  <strong> "Please scan the UPI QR code, transfer an amount of Rs.500, and kindly provide
the transaction ID in the text box below. <br>If any error occurs then your registrastion will
get cancelled and amount will get refund to your account. Thank you!"</strong> <br>
   <br>
  <label for="transaction">Transaction Id: </label>
  <input type="text" name="transaction" id="transaction" placeholder="Transaction Id."
required>
  <br>
```

Project Documentation

```
</fieldset>
<fieldset>
  <legend>Declaration</legend>
  <br>
  <input type="checkbox" name="dec" id="decl">
  <strong> I, declare that both my parents, are aware of and consent to the submission of
my application form. <br> We have thoroughly verified all the details in the application,
including payment information, before submission.</strong> <br><br>
  <button type="reset">Reset</button> &nbsp; &nbsp;
  <button type="submit">Submit</button>
</fieldset>
</form>
</body>
</html>
```

APPLCODE.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Form_batch</title>
  <link rel="stylesheet" href="./forml.css">
</head>
<body>
  <div class="heading">
    Application Form
  </div>
  <div class="form-container">
    <form action="appform.php" method="post">
```

Project Documentation

```
<fieldset><table>
  <legend >Personal Detail</legend>
  <tr>
    <td <label for="username">Name: </label></td>
    <td><input type="text" name="username" id="username" placeholder="Username"
required></td>
  </tr>
  <tr>
    <td><label for="fname">Father's Name: </label></td>
    <td><input type="text" name="fname" id="username" placeholder="Father Name"
required></td>
  </tr>
  <tr>
    <td><label for="mname">Mother's Name: </label></td>
    <td><input type="text" name="mname" id="username" placeholder="Mother Name"
required></td>
  </tr>
  <tr>
    <td><label for="email">E-mail: </label></td>
    <td><input type="email" name="email" id="mail" placeholder="Email Address"
required></td>
  </tr>
  <tr>
    <td><label for="adhar">Adhar Number: </label></td>
    <td><input type="number" name="adhar" id="adhar" placeholder="Adhar No."
required></td>
  </tr>
  <tr>
    <td <label for="mob">Mobile No.: </label></td>
    <td><input type="number" name="mob" id="mob" placeholder="Mobile No."
required></td>
  </tr>
</tr>
```

Project Documentation

```
<td><label for="dob">Date of birth:</label></td>
<td><input type="date" name="dob" id="dob" placeholder="DOB" required></td>
</tr>
</table>
</fieldset>
<br>
<fieldset>
<legend>Course Detail</legend>
<table>
<tr>
<td><label for="userclass">Class: </label></td>
<td><select name="userclass" id="userclass">

<option value="6">6th</option>
<option value="7">7th</option>
<option value="8">8th</option>
<option value="9">9th</option>
<option value="10">10th</option>
<option value="11">11th</option>
<option value="12">12th</option>
</select></td>
</tr>
<tr>
<td><label for="usersub">Subject: </label> </td>
<td> <select name="usersub" id="usersub">
<option value="C language">C language</option>
<option value="C++">C++</option>
<option value="Java">Java</option>
<option value="Python">Python</option>
</select></td>
</tr>
```

Project Documentation

```
</table>
</fieldset>
<br>
<fieldset>
  <legend>Residential Detail</legend>
  <table>
    <tr>
      <td><label for="street">Street: </label></td>
      <td><input type="text" name="street" id="street" placeholder="Street"
required></td>
    </tr>
    <tr>
      <td><label for="city">City: </label></td>
      <td><input type="text" name="city" id="city" placeholder="City" required></td>
    </tr>
    <tr>
      <td><label for="pincode">Pincode: </label></td>
      <td><input type="number" name="pincode" id="pincode" placeholder="PinCode"
required></td>
    </tr>
    <tr>
      <td><label for="district">District: </label></td>
      <td><input type="text" name="district" id="district" placeholder="District"
required></td>
    </tr>
    <tr>
      <td><label for="state">State: </label></td>
      <td><input type="text" name="state" id="state" placeholder="State" required></td>
    </tr>
  </table>
</fieldset>
<br>
```

Project Documentation

```
<fieldset>
  <legend>Payment</legend>
  <br>
  <strong> "Please scan the UPI QR code, transfer an amount of Rs.500, and kindly provide
the transaction ID in the text box below. <br>If any error occurs then your registrastion will
get cancelled and amount will get refund to your account. Thank you!"</strong> <br>
   <br>
  <label for="transaction">Transaction Id: </label>
  <input type="text" name="transaction" id="transaction" placeholder="Transaction Id"
required>
  <br>
</fieldset>
<fieldset>
  <legend>Declaration</legend>
  <br>
  <input type="checkbox" name="dec" id="decl" required>
  <strong> I, declare that both my parents, are aware of and consent to the submission of
my application form. <br> We have thoroughly verified all the details in the application,
including payment information, before submission.</strong> <br><br>
  <button type="reset">Reset</button> &nbsp; &nbsp;
  <button type="submit">Submit</button>
</fieldset>
</form>
</body>
</html>
```

CHECK_LOGIN.PHP

```
<?php
session_start();
// Replace with your database credentials
$servername = "localhost";
$username = "root";
```


Project Documentation

```
$password = "";
$dbname = "lvdata";

// Create a connection
$conn = new mysqli($servername, $username, $password, $dbname);

// Check the connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

// Check if the form is submitted
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Retrieve form data
    $username = $_POST["username"];
    $pass = $_POST["pass"];

    // Perform a basic query to check credentials
    $sql = "SELECT * FROM login WHERE username = '$username' AND pass = '$pass'";
    $result = $conn->query($sql);

    if ($result->num_rows > 0) {
        $_SESSION['username']=$username;
        header("location: index.php");
        exit();
        echo "Login successful!";
        // Redirect or perform other actions after successful login
    } else {
        echo "Invalid username or password";
        // Handle failed login (e.g., redirect to login page with an error message)
    }
}
```

Project Documentation

```
}  
// Close the connection  
$conn->close();  
?>
```

CODE.HTML

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <meta charset="UTF-8">  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  <!-- font -->  
  <link rel="preconnect" href="https://fonts.googleapis.com">  
  <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>  
  <link href="https://fonts.googleapis.com/css2?family=Poppins&display=swap" rel="stylesheet">  
  <title>Code India</title>  
  <style>  
    *{  
      background-color: rgb(65, 63, 63);  
      color: rgb(102, 242, 102);  
      font-family: 'Poppins', sans-serif;  
    }  
    h2{  
      color: white;  
    }  
    .content{  
      display: flex;  
      justify-content: space-around;  
      margin: 30px;  
    }  
  </style>
```

Project Documentation

```
.para{
  width: 600px;
  margin: 30px;
}
.box{
  background-color: rgb(87, 87, 87);
  border-radius: 5px;
  width: 230px;
  color: rgb(11, 245, 128);
  padding: 8px;
  margin: 15px;
}
.box1{
  background-color: rgb(87, 87, 87);
}
h1,p{
  text-align: center;
  color: white;
}
.quiz{
  width: 500px;
  margin: auto;
}
button{
  padding: 8px;
  margin: 3px;
  border-radius: 4px;
  background-color: rgb(16, 16, 16);
}
a{ padding: 5px;
  margin: 100px;
```

Project Documentation

```
border-radius: 4px;
background-color: rgb(16, 16, 16);
text-decoration: none;
}
</style>
```

```
</head>
```

```
<body>
```

```
<h1>Code India</h1>
```

```
<p>"Code India Grow India"</p>
```

```
<div class="content">
```

```
<div class="para">
```

Welcome to Code India, a dynamic realm within the vast Learniverse dedicated to empowering students from class 6 to 12 with the art of programming. At Code India, we believe in unlocking the potential of young minds, fostering a passion for coding, and preparing students for the challenges and opportunities of the digital age.

```
<h2>About Code India</h2>
```

Code India is not just a platform; it's an educational journey tailored to cater to the unique needs of students exploring the world of programming languages. Our courses are meticulously designed to make coding accessible, enjoyable, and intellectually rewarding for students at various academics levels.

```
<h2>QUIZ </h2>
```

Welcome to programming Quiz-an exciting journey through the world of code! This quiz is designed to test your knowledge and passion for programming languages.

Whether you're a novice or a seasoned coder, these questions will challenge your understanding of the fundamental concepts.

```
<br> <br>
```

```
<a href="/codequiz.html">Attempt Quiz</a>
```

```
</div>
```

```
<div>
```

```
<div class="box">
```

```
<h2 class="box1"> PYTHON </h2>Theory, notes, live class, coding practice
```

```
<a href="/applcode.html">Join</a></div>
```

```
<div class="box">
```

Project Documentation

```
<h2 class="box1">JAVA</h2>Theory, notes, live class, coding practice
<a href="/.appliance.html">Join</a></div>
<div class="box"><h2 class="box1">C++</h2>Theory, notes, live class, coding practice
<a href="/.appliance.html">Join</a></div>
<div class="box"><h2 class="box1">C</h2>Theory, notes, live class, coding practice
<a href="/.appliance.html">Join</a></div>
</div>
</div>
</body>
</html>
```

CODEQUIZ.HTML

```
<!-- quiz.html -->
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Quiz</title>
  <style>
    body{
      background-color: black;
      margin: 0;
    }
    .box{
      height: 50px;
      width: 100%;
      background-color: rgb(4, 143, 41);
    }
    .circle{
      height: 100px;
```

Project Documentation

```
width: 100px;
border-radius: 50%;
background-color: rgb(5, 131, 49);
text-align: center;
/* margin: auto; */
display: flex;
align-items: center;
}
.circle p{
margin: auto;
font-size: 20px;
color: white;
}
#quiz-container{
/* text-align: center; */
margin: auto;
font-size: 25px;
color: rgb(241, 241, 243);
width: 500px;
}
#result-container{

/* text-align: center; */
margin: auto;
font-size: 25px;
color: white;
width: 500px;
}
.btn{
display: flex;
justify-content: center;
```

Project Documentation

```
}
button{
  margin: auto;
  height: 50px;
  width: 150px;
  border-radius: 7px;
  font-size: 15px;
  background-color: rgb(4, 143, 41);
  color: white;
}
ul li{
  list-style: none;
}
</style>
</head>
<body>
  <div class="box">
    <div class="circle">
      <p>Quiz</p>
    </div>
  </div>
  <div id="quiz-container"></div>
  <div class="btn"><button onclick="submitQuiz()">Submit Answers</button></div>
  <div id="result-container"></div>
  <script>
    let questions = [];

    // Use AJAX or Fetch API to get questions from the server
    fetch('codequiz.php')
      .then(response => response.json())
```

Project Documentation

```
.then(data => {
  questions = data;
  // Use the questions to dynamically generate the quiz UI
  displayQuiz();
})
.catch(error => console.error('Error fetching questions:', error));

function displayQuiz() {
  const quizContainer = document.getElementById('quiz-container');

  questions.forEach((question, index) => {
    const questionElement = document.createElement('div');
    questionElement.innerHTML = `
      <p>${index + 1}. ${question.question}</p>
      <ul>
        ${question.options.map((option, optionIndex) => `
          <li>
            <input type="radio" name="q${index}" value="${optionIndex}">
            ${option}
          </li>`).join("")}
      </ul>
    `;
    quizContainer.appendChild(questionElement);
  });
}

function submitQuiz() {
  const resultContainer = document.getElementById('result-container');
  let score = 0;
  questions.forEach((question, index) => {
    const selectedOptionIndex =
document.querySelector(`input[name="q${index}"]:checked`);
```


Project Documentation

```
if (selectedOptionIndex) {
    const selectedOption = parseInt(selectedOptionIndex.value);
    if (question.options[selectedOption] === question.correctAnswer) {
        score++;
    }
}
});

const resultMessage = `You scored ${score} out of ${questions.length}.`;
resultContainer.innerHTML = `

${resultMessage}</p>`;
}
</script>
</body>
</html>


```

CODEQUIZ.PHP

```
<?php
// Mock data for demonstration purposes
$allQuestions = [
    ["question" => "Which programming language is known for simplicity and readability?",
    "options" =>["Ruby","C++","Java","Python"], "correctAnswer" => "Python"],
    ["question" => "Which language is known as a low-level programming language often used
for system programming?", "options" =>["Python","Java","Assembly
language","C"],"correctAnswer"=>"Assembly language"],
    ["question"=> "Which is of the following is not a valid data type in most programming
language?", "options"=>["String","Float","Character","Loop"], "correctAnswer"=> "Loop"],
    ["question"=> "Who invented C language?", "options"=>["Dennis Ritchie","Robert
Downey","Jim sheridin","Tim berner"], "correctAnswer"=> "Dennis Ritchie"],
    ["question"=> "How many types of iteration are in C language?",
"options"=>["1","2","3","4"], "correctAnswer"=> "3"],
    ["question"=> "++ What is this symbol stands for?",
"options"=>["decrement","multiply","increment","modulus"], "correctAnswer"=>
"increment"],
    ["question" => "What is the first index of array?", "options" => ["0", "1", "01", "10"],
"correctAnswer" => "0"],
```

Project Documentation

```
// ... add more questions ...  
];  
shuffle($allQuestions); // Shuffle the array to get random questions  
$selectedQuestions = array_slice($allQuestions, 0, 5); // Select the first 10 questions  
echo json_encode($selectedQuestions);  
?>
```

DO_LOGIN.PHP

```
<?php  
session_start();  
  
// Replace with your database credentials  
$servername = "localhost";  
$username = "root";  
$password = "";  
$dbname = "lvdata";  
  
// Create a connection  
$conn = new mysqli($servername, $username, $password, $dbname);  
  
// Check the connection  
if ($conn->connect_error) {  
    die("Connection failed: " . $conn->connect_error);  
}  
  
// Check if the form is submitted  
if ($_SERVER["REQUEST_METHOD"] == "POST") {  
    // Retrieve form data  
    $username = $_POST["username"];  
    $pass = $_POST["pass"];  
  
    // Add other form fields as needed
```

Project Documentation

```
// Insert data into the database
$sql = "INSERT INTO login (username, pass) VALUES ('$username','$pass)";

if ($conn->query($sql) === TRUE) {
    $_SESSION['username']=$username;
    header("location: index.php");
    exit();
    echo "Sign up Successfully. New Account has been added.";
} else {
    echo "Error: " . $sql . "<br>" . $conn->error;
}
}

// Close the connection
$conn->close();
?>
```

FORML.CSS

```
{
    margin: 0;
    background-color: #ffffff;
    font-family: 'Poppins', sans-serif;
}

.heading{
    font-size: 30px;
    padding: 20px;
    color: rgb(4, 4, 88);
    text-align: center;
}

fieldset{
    width: 600px;
```

Project Documentation

```
    margin: auto;
}
label{
    font-size: 20px;
    font-weight: 600;
    color: rgb(4, 4, 88);
}
label,input{
    /* margin: 10px; */
    height: 25px;
    font-size: 18px;
}
select{
    height: 30px;
}
table{
    margin: auto;
}
button{
    width: 70px;
    height: 35px;
    border-radius: 4px;
    background-color: rgb(171, 219, 250);;
}
legend{
    color:rgb(4, 4, 88); ;
    text-align: center;
    font-size: 22px;
}
input{
    border-radius: 5px;
```

Project Documentation

```
    color: rgb(5, 5, 183);  
}
```

HOMESTYLE.CSS

```
 *{ margin: 0px;  
    font-family: 'Poppins', sans-serif;  
 }  
 /* nav bar */  
 header{  
    padding-left: 10px;  
    height: 60px;  
    width: 100%;  
    background-color: rgb(4,4,88);  
 }  
 .nav{  
    display: flex;  
    justify-content: left;  
 }  
 /* Style the custom dropdown container */  
 .dropdown, .dp {  
    position: relative;  
    display: inline-block;  
 }  
 /* Style the dropdown button */  
 .dropdown button ,.dp {  
    font-size: 16px;  
    border: none;  
    outline: none;  
    color: white;  
    background-color: rgb(4,4,88);  
    padding: 14px 16px;
```

Project Documentation

```
margin: 0;
}
a{
text-decoration: none;
font-size: 16px;
color: white;
}

/* Style the dropdown content (hidden by default) */
.dropdown-content {
display: none;
position: absolute;
background-color: #f9f9f9;
min-width: 160px;
box-shadow: 0 8px 16px 0 rgba(0,0,0,0.2);
z-index: 1;
}

/* Style the links inside the dropdown */
.dropdown-content a {
color: black;
padding: 12px 16px;
text-decoration: none;
display: block;
text-align: left;
}

/* Change color on hover */
.dropdown-content a:hover {
background-color: #ddd;
}
```

Project Documentation

```
/* Show the dropdown menu on hover */
.dropdown:hover .dropdown-content {
  display: block;
}

/* slogan */
.mid_class{
  position: relative;
  width: 100%;
  height: 500px;
  background: url(/pexels-julia-m-cameron-4143800.jpg)center/cover no-repeat;
  opacity: 0.7;
}
.overlay{
  position: absolute;
  top: 50%;
  left: 50%;
  transform: translate(-50%,-50%);
  color: rgb(3, 3, 52);
  font-size: 22px;
}
h1{
  margin: 40px;
  font-size: 40px;
  text-align: center;
  color: rgb(4, 4, 88);
  padding-left: 25px;
}

/* content section */
```

Project Documentation

```
.content_section{
  font-size: 20px;
  margin-top: 40px;
  margin-bottom: 40px;
  padding-left: 25px;
  color: black;
  /* font-family: 'Quicksand', sans-serif; */
  text-align: center;
}

.step{
  display: flex;
  justify-content: space-around;
}

#section3 p{
  text-align: center;
  color: black;
  font-size: 20px;
  margin-top: 10px;
}

.adv{
  display: flex;
  justify-content: space-around;
}

.adv a{
  color: rgb(4,4,88);
}

#sub9,#sub10{
  margin-top: 50px;
  margin-left: 50px;
  margin-right: 50px;
  margin-bottom: 20px;
}
```


Project Documentation

```
border-radius: 8px;
color: white;
background-color: rgb(3, 3, 130) ;
width: 150px;
height: 60px;
font-size: 20px;
}
/* copy botton css */
.batch_head{
font-size: 30px;
display: flex;
justify-content: space-around;
margin: 25px;
color: rgb(4, 4, 88);
padding-top: 80px;
}
.btn{
border-radius: 8px;
color: white;
background-color: rgb(4, 4, 88) ;
width: 70px;
}
.btn a{
font-size: 15px;
color:white ;
}
.batch{
font-size: 15px;
font-weight: 600;
margin: 25px;
padding: 25px;
```

Project Documentation

```
color: rgb(46, 126, 216);
border-radius: 15px;
width: 400px;
font-family: 'Exo', sans-serif;
font-family: 'Quicksand', sans-serif;
box-shadow: 2px 2px 8px 2px #a1a1a1;
}
.batchpic{
width: 200px;
height: 200px;
padding-top : 15px;
border-radius: 50%;
}
.phy_batch, .chem_batch, .maths_batch, .bio_batch, .eng_batch{
display: flex;
align-items: center;
justify-content: space-evenly;
}
.phy_batch :hover{
transform: scale(1.1);
}
.chem_batch :hover{
transform: scale(1.1);
}
.bio_batch :hover{
transform: scale(1.1);
}
.maths_batch :hover{
transform: scale(1.1);
}
.eng_batch :hover{
```

Project Documentation

```
        transform: scale(1.1);
    }
div p span{
    color: rgb(11, 112, 167);
    font-weight: bolder;
    font-size: 22px;
}
.last_part{
    background-color: rgb(4, 4, 88);
    font-size: 15px;
    padding: 30px;
    margin-top: 20px;
    color: azure;
    border-radius: 1px;
}
.footer_contact{
    color: azure;
    font-size: 15px;
    padding: 30px;
    margin-top: 20px;
    color: azure;
    border-radius: 1px;
}
footer{
    display: flex;
    background-color: rgb(4, 4, 88);
    font-size: 20px;
    padding: 30px;
    margin-top: 20px;
    color: azure;
    border-radius: 1px;
```

Project Documentation

```
justify-content: space-around;
}
```

INDEX.PHP

```
<?php
session_start();

// Check if the user is not logged in, redirect to login page
if (!isset($_SESSION['username'])) {
    header("Location: login.php");
    exit();
}

// User is logged in, fetch user profile data
$username = $_SESSION['username'];

// Replace with your database credentials
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "lvdata";

$conn = new mysqli($servername, $username, $password, $dbname);
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

// Fetch user profile data from the 'user_profile' table
$sqlProfile = "SELECT * FROM login WHERE username='$username'";
$resultProfile = $conn->query($sqlProfile);

if ($resultProfile->num_rows > 0) {
    $userProfile = $resultProfile->fetch_assoc();
}
```

Project Documentation

```
} else {  
    // Handle the case where user profile data is not found  
    $userProfile = array();  
    $username=$userProfile['username'];  
}  
$conn->close();  
?>  
<!DOCTYPE html>  
<html lang="en">  
<head>  
    <meta charset="UTF-8">  
    <meta name="viewport" content="width=device-width, initial-scale=1.0">  
    <title>Home page</title>  
    <!-- CSS link -->  
    <link rel="stylesheet" href="/homestyle.css">  
    <!-- Font link -->  
    <link rel="preconnect" href="https://fonts.googleapis.com">  
    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>  
    <link href="https://fonts.googleapis.com/css2?family=Poppins&display=swap" rel="stylesheet">  
    <script>  
        // function to show class 9 subject button  
        function show_sub(){  
            const d=document.getElementById('detail_1');  
            const newbtn=document.createElement('button');  
            newbtn.textContent = "Science";  
            newbtn.id="scroll1";  
            newbtn.onclick=function scl(){  
                const sclbtn=document.getElementById("scroll1");  
                const tgbtn=document.getElementById("target1");  
                tgbtn.scrollIntoView({behavior:"smooth"});  
            }  
        }  
    </script>  
</head>  
<body>  
<div id="detail_1" style="border: 1px solid black; padding: 5px; width: fit-content; margin: auto; text-align: center;">  
<h3 style="margin: 0;">Class 9 Subjects  
<div style="display: flex; justify-content: space-around; margin-top: 10px;">  
<div style="border: 1px solid black; padding: 5px; width: 40%; text-align: center;">  
<h4 style="margin: 0;">Science  
<div style="margin-top: 5px;">  
<button style="border: none; background: none; padding: 0 10px; cursor: pointer; color: #000080; font-weight: bold;">Show Science Button  
</div>  
</div>  
</div>  
</div>  
</body>  
</html>
```

Project Documentation

```
}  
d.appendChild(newbtn);  
const newbtn2=document.createElement('button');  
newbtn2.textContent = "Social Science";  
newbtn2.id="scroll2";  
newbtn2.onclick=function scl(){  
    const sclbtn=document.getElementById("scroll2");  
    const tgbtn=document.getElementById("target2");  
    tgbtn.scrollIntoView({behavior:"smooth"});  
}  
d.appendChild(newbtn2);  
const newbtn3=document.createElement('button');  
newbtn3.textContent = "Computer Science";  
newbtn3.id="scroll3";  
newbtn3.onclick=function scl(){  
    const sclbtn=document.getElementById("scroll3");  
    const tgbtn=document.getElementById("target3");  
    tgbtn.scrollIntoView({behavior:"smooth"});  
}  
d.appendChild(newbtn3);  
const newbtn4=document.createElement('button');  
newbtn4.textContent = "Maths";  
newbtn4.id="scroll4";  
newbtn4.onclick=function scl(){  
    const sclbtn=document.getElementById("scroll4");  
    const tgbtn=document.getElementById("target4");  
    tgbtn.scrollIntoView({behavior:"smooth"});  
}  
d.appendChild(newbtn4);  
const newbtn5=document.createElement('button');  
newbtn5.textContent = "English";
```

Project Documentation

```
newbtn5.id="scroll5";
newbtn5.onclick=function scl(){
    const sclbtn=document.getElementById("scroll5");
    const tgbtn=document.getElementById("target5");
    tgbtn.scrollIntoView({behavior:"smooth"});
}
d.appendChild(newbtn5);
// adding style on button of subject
newbtn.style.backgroundColor="white ";
newbtn.style.fontSize="17px";
newbtn.style.margin="25px";
newbtn.style.borderRadius="5px"
newbtn.style.width="100px";
newbtn.style.height="45px";
newbtn.style.borderColor="rgb(4, 4, 88)";
newbtn.style.color="rgb(4, 4, 88)";

newbtn2.style.backgroundColor="white";
newbtn2.style.fontSize="17px";
newbtn2.style.margin="25px";
newbtn2.style.borderRadius="5px"
newbtn2.style.width="150px";
newbtn2.style.height="45px";
newbtn2.style.borderColor="rgb(4, 4, 88)";
newbtn2.style.color="rgb(4, 4, 88)";

newbtn3.style.backgroundColor="white ";
newbtn3.style.fontSize="17px";
newbtn3.style.margin="25px";
newbtn3.style.borderRadius="5px"
newbtn3.style.width="180px";
```

Project Documentation

```
newbtn3.style.height="45px";
newbtn3.style.borderColor="rgb(4, 4, 88)";
newbtn3.style.color="rgb(4, 4, 88)";

newbtn4.style.backgroundColor="white ";
newbtn4.style.fontSize="17px";
newbtn4.style.margin="25px";
newbtn4.style.borderRadius="5px"
newbtn4.style.width="100px";
newbtn4.style.height="45px";
newbtn4.style.borderColor="rgb(4, 4, 88)";
newbtn4.style.color="rgb(4, 4, 88)";

newbtn5.style.backgroundColor="white ";
newbtn5.style.fontSize="17px";
newbtn5.style.margin="25px";
newbtn5.style.borderRadius="5px"
newbtn5.style.width="100px";
newbtn5.style.height="45px";
newbtn5.style.borderColor="rgb(4, 4, 88)";
newbtn5.style.color="rgb(4, 4, 88)";
// disabling button for more click
disableButton();
function disableButton(){
    document.getElementById("sub9").disabled=true;
}
}
// function for class 10 subject detail button
function show_sub2(){
    const d=document.getElementById('detail_2');
    const newbtn=document.createElement('button');
```


Project Documentation

```
newbtn.textContent = "Science";
newbtn.id="scroll6";
newbtn.onclick=function scl(){
    const sclbtn=document.getElementById("scroll6");
    const tgbtn=document.getElementById("target6");
    tgbtn.scrollIntoView({behavior:"smooth"});
}
d.appendChild(newbtn);
const newbtn2=document.createElement('button');
newbtn2.textContent = "Social Science";
newbtn2.id="scroll7";
newbtn2.onclick=function scl(){
    const sclbtn=document.getElementById("scroll7");
    const tgbtn=document.getElementById("target7");
    tgbtn.scrollIntoView({behavior:"smooth"});
}
d.appendChild(newbtn2);
const newbtn3=document.createElement('button');
newbtn3.textContent = "Computer Science";
newbtn3.id="scroll8";
newbtn3.onclick=function scl(){
    const sclbtn=document.getElementById("scroll8");
    const tgbtn=document.getElementById("target8");
    tgbtn.scrollIntoView({behavior:"smooth"});
}
d.appendChild(newbtn3);
const newbtn4=document.createElement('button');
newbtn4.textContent = "Maths";
newbtn4.id="scroll9";
newbtn4.onclick=function scl(){
    const sclbtn=document.getElementById("scroll9");
```

Project Documentation

```
const tgbtn=document.getElementById("target9");
tgbtn.scrollIntoView({behavior:"smooth"});
}
d.appendChild(newbtn4);
const newbtn5=document.createElement('button');
newbtn5.textContent = "English";
newbtn5.id="scroll10";
newbtn5.onclick=function scl(){
    const sclbtn=document.getElementById("scroll10");
    const tgbtn=document.getElementById("target10");
    tgbtn.scrollIntoView({behavior:"smooth"});
}
d.appendChild(newbtn5);
// adding style to button
newbtn.style.backgroundColor="white ";
newbtn.style.fontSize="17px";
newbtn.style.margin="25px";
newbtn.style.borderRadius="5px"
newbtn.style.width="100px";
newbtn.style.height="45px";
newbtn.style.borderColor="rgb(4, 4, 88)";
newbtn.style.color="rgb(4, 4, 88)";

newbtn2.style.backgroundColor="white ";
newbtn2.style.fontSize="17px";
newbtn2.style.margin="25px";
newbtn2.style.borderRadius="5px"
newbtn2.style.width="150px";
newbtn2.style.height="45px";
newbtn2.style.borderColor="rgb(4, 4, 88)";
newbtn2.style.color="rgb(4, 4, 88)";
```

Project Documentation

```
newbtn3.style.backgroundColor="white";
newbtn3.style.fontSize="17px";
newbtn3.style.margin="25px";
newbtn3.style.borderRadius="5px"
newbtn3.style.width="180px";
newbtn3.style.height="45px";
newbtn3.style.borderColor="rgb(4, 4, 88)";
newbtn3.style.color="rgb(4, 4, 88)";

newbtn4.style.backgroundColor="white ";
newbtn4.style.fontSize="17px";
newbtn4.style.margin="25px";
newbtn4.style.borderRadius="5px"
newbtn4.style.width="100px";
newbtn4.style.height="45px";
newbtn4.style.borderColor="rgb(4, 4, 88)";
newbtn4.style.color="rgb(4, 4, 88)";

newbtn5.style.backgroundColor="white ";
newbtn5.style.fontSize="17px";
newbtn5.style.margin="25px";
newbtn5.style.borderRadius="5px"
newbtn5.style.width="100px";
newbtn5.style.height="45px";
newbtn5.style.borderColor="rgb(4, 4, 88)";
newbtn5.style.color="rgb(4, 4, 88)";

// disabling button for more click
disableButton();

function disableButton(){
    document.getElementById("sub10").disabled=true;
```

Project Documentation

```
    }
  }
</script>
</head>
<body>
  <header>
<!-- Navigation bar starts-->
<form action="./logout.php" method="post">
<span class="dp" id="hello">Hello,<?php echo $userProfile['username']; ?></span>
<button class="dp"><a href="./index.php">Home</a></button>
<button class="dp"> <a href="./about.html">About</a></button>
<button class="dp"><a href="./notes.html">Notes</a></button>

<div class="dropdown">
  <button>Quiz</button>
  <div class="dropdown-content">
    <a href="./quiz.html" target="_blank">9th</a>
    <a href="quiz10.html" target="_blank">10th</a>
  </div>
</div>
<button class="dp"><a href="./lecture.html">Lecture</a></button>
<button class="dp"><a href="./profile.php">Profile</a></button>
<button type="submit" class="dp">Logout</button></form>
<!-- navigation bar ends -->
  </header>
  <main>
    <!-- Slogan part start -->
    <section id="section1">
      <div class="mid_class">

        <!-- aside code india check -->
```

Project Documentation

```
<aside style="width:200px; height:480px ; background-color:#000000; color: green; padding: 10px;">
```

```
  <h2 style="padding:10px; font-size:30px"> <u>Code India</u></h2>
```

```
  <h4 style="padding:15px 5px">"Code India Grow India" </h4>
```

```
  <p style="font-size:12px; padding:5px;">The unbreakable part of Learniverse. This is a code learning platform. </p>
```

```
  <p style="font-size:12px; padding:5px;">We are providing various language courses to enhance your coding skills from beginner to advance level for students.</p>
```

```
  <p style="font-size:12px; padding:5px;">Learn programming language in a fun way. For junior class students to higher secondary students.</p>
```

```
  <button style="width:170px; height:30px; background-color:green;margin:25px 15px"><a href="/code.html" style="padding:10px;">visit code india</a></button>
```

```
</aside>
```

```
  <div class="overlay">
```

```
    <h1 class="slogan">Learn anything from anywhere <br> Learn online </h1>
```

```
  </div>
```

```
</div>
```

```
</section>
```

```
<!-- slogan part ends -->
```

```
<!-- aside starts for code india -->
```

```
<!-- <aside style="width:1400px; height:200px;border-radius:50px ; background-color:#dedede; color: #9c09e0; padding: 10px; margin:30px;">
```

```
  <h2 style="padding:10px; font-size:30px; text-align:center;"> <u>Code India</u></h2>
```

```
  <h4 style="padding:15px 5px">"Code India Grow India" </h4>
```

```
  <p style="font-size:12px; padding:5px;">The unbreakable part of Learniverse. This is a code learning platform. We are providing various language courses to enhance your coding skills from beginner to advance level for students.Learn programming language in a fun way. For junior class students to higher secondary students.</p>
```

```
  <button style="width:170px; height:30px; background-color:green;margin:25px 15px"><a href="#" style="padding:10px;">visit code india</a></button>
```

```
</aside> -->
```

```
<!-- description part -->
```

```
<section id="section2">
```

Project Documentation

```
<h1> <strong>We are Learniverse</strong> </h1>
```

```
<p class="content_section"> We believe that every student deserves access to the best  
<br> possible education, no matter where they are located. That's why <br> we've created a  
cutting-edge online platform that connects you <br> with top-rated tutors via Zoom.  
Whether you need help with math, <br> science or other subjects, our team of experienced  
educators is here to <br> help. These are steps to get enroll in course. <br>
```

```

```

```
</p>
```

```
<p class="content_section" style="text-align: left; padding-left: 40px; font-weight: 600;  
font-size: 28px; color: rgb(4,4,88);">
```

```
Clicking the button below. Uncover all the subjects. <br>
```

```
</p>
```

```
</section>
```

```
<!-- end of description part -->
```

```
<!-- sub detail part starts -->
```

```
<div class="sub_detail_1">
```

```
<button onclick="show_sub()" id="sub9">Class 9th</button>
```

```
<div id="detail_1">
```

```
</div>
```

```
<!-- </div>
```

```
<div class="sub_detail_2"> -->
```

```
<button onclick="show_sub2()" id="sub10">Class 10th</button>
```

```
<br>
```

```
<div id="detail_2">
```

```
</div>
```

```
</div>
```

```
<!-- subject detail part ends -->
```

```
<!-- steps to enroll -->
```

```
<section id="section3">
```

```
<p>
```

Project Documentation

" Step into an engaging learning experience with our courses! Each subject brings you a daily dose of live classes on weekdays, making learning interactive and fun.
 Dive into the world of knowledge with our weekend doubt classes, where your questions find clear answers, ensuring you're on the right track.

Picture this: every day, you get a live class bringing the subject to life, and on weekends, a dedicated doubt session. The
 learning resources, from notes to quizzes, are regularly refreshed, keeping you engaged and up-to-date. Join us in this vibrant learning community where every day is a step toward mastering the subject and achieving your academic goals. "

</p>

</section>

<div class="adv">

<p>

International Maths Olympiad

</p>

<p>

International Science Olympiad

</p>

<p>

NTSE Exam

</p>

<p>

Navodya Entrance

</p>

</div>

<!-- batch heading -->

Project Documentation

```
<p class="batch_head"> Batch class 9th </p>
```

```
<!-- batch detail part start-->
```

```
<div class="batch_section">
```

```
  <div class="phy_batch" id="target1">
```

```
    <p class="batch"> <span>SCIENCE</span> <br> <br>
```

```
      Batch Timing: 4 PM - 5PM <br> Monday - Tuesday - Wednesday <br>
```

```
      Fees: Rs.500 <br> <br>
```

```
      <button class="btn"><a href="/.appl.html">JOIN</a></button>
```

```
    </p>
```

```
    
```

```
  </div>
```

```
  <div class="chem_batch" id="target2">
```

```
    
```

```
    <p class="batch"> <span>SOCIAL SCIENCE</span> <br> <br>
```

```
      Batch Timing: 4 PM - 5PM <br> Thursday - Friday - Saturday <br>
```

```
      Fees: Rs.500 <br> <br>
```

```
      <button class="btn"><a href="/.appl.html">JOIN</a></button>
```

```
    </p>
```

```
  </div>
```

```
  <div class="bio_batch" id="target3">
```

```
    <p class="batch"> <span>COMPUTER SCIENCE</span> <br> <br>Batch Timing: 4 PM -  
5PM <br> Monday - Tuesday - Wednesday <br>
```

```
      Fees: Rs.500 <br> <br>
```

```
      <button class="btn"><a href="/.appl.html">JOIN</a></button>
```

```
    </p>
```

```
    
```

```
  </div>
```

```
  <div class="maths_batch" id="target4">
```

```
    
```

```
    <p class="batch"><span>MATHS</span> <br> <br>
```


Project Documentation

Batch Timing: 4 PM - 5PM
 Thursday - Friday - Saturday

Fees: Rs.500

<button class="btn">JOIN</button>

</p>

</div>

<div class="eng_batch" id="target5">

<p class="batch"> ENGLISH

Batch Timing: 4 PM - 5PM
 Monday - Tuesday - Wednesday

Fees: Rs.500

<button class="btn">JOIN</button>

</p>

</div>

</div>

<!-- batch1 part ends -->

<!-- batch2 starts -->

<p class="batch_head"> Batch class 10th</p>

<div class="batch_section">

<div class="phy_batch" id="target6">

<p class="batch"> SCIENCE

Batch Timing: 4 PM - 5PM
 Monday - Tuesday - Wednesday

Fees: Rs.500

<button class="btn">JOIN</button>

</p>

</div>

<div class="chem_batch" id="target7">

<p class="batch"> SOCIAL SCIENCE

Batch Timing: 4 PM - 5PM
 Thursday - Friday - Saturday

Project Documentation

```
Fees: Rs.500 <br> <br>
<button class="btn"><a href="/.appl.html">JOIN</a></button>
</p>
</div>
<div class="bio_batch" id="target8">
  <p class="batch"> <span>COMPUTER SCIENCE</span> <br> <br>Batch Timing: 4 PM -
5PM <br> Monday - Tuesday - Wednesday <br>
  Fees: Rs.500 <br> <br>
  <button class="btn"><a href="/.appl.html">JOIN</a></button>
  </p>
  
</div>
<div class="maths_batch" id="target9">
  
  <p class="batch"><span> MATHS</span> <br> <br>
  Batch Timing: 4 PM - 5PM <br> Thursday - Friday - Saturday <br>
  Fees: Rs.500 <br> <br>
  <button class="btn"><a href="/.appl.html">JOIN</a></button>
  </p>
</div>
<div class="eng_batch" id="target10">
  <p class="batch"> <span>ENGLISH</span><br> <br>
  Batch Timing: 4 PM - 5PM <br> Monday - Tuesday - Wednesday <br>
  Fees: Rs.500 <br> <br>
  <button class="btn"><a href="/.appl.html">JOIN</a></button>
  </p>
  
</div>
</div>
</main>
<!-- batch2 ends -->
```

Project Documentation

```
<!-- Footer starts -->
<footer>
<div class="last_part">
  <p>
    Address: <br>
    Head office: LearniVerse, Hospital Chowk,<br>
    Motihari, East Champaran, Bihar <br> <br>
  </p>
  <p>
    Pincode: 845401 <br> <br>
  </p>
  <p>
    Email:
    info@LVerse.com <br>
    Phone Number: 06252 235351
  </p>
</div>
<p style="font-size: 15px; padding-top: 300px;"> &copy 2023 Learniverse private
limited</p>
<div class="footer_contact">
  
we_r_learniverse <br>
  
we_r_lv <br>
  
info@lverse.com <br>
   learniverse
</div>
</footer>
</body>
</html>
```

Project Documentation

LECTURE.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Lecture</title>
  <style>
    *{
      margin: 0;
    }
    .our_about{
      background-color: rgb(4, 4, 88);
      height: 70px;
      width: 100%;
      margin-bottom: 70px;
    }
    .circle{
      background-color: rgb(3, 3, 131);
      color: white;
      width: 160px;
      height: 160px;
      border-radius: 50%;
      margin-bottom: 50px;
    }
    .circle p{
      text-align: center;
      font-size: 30px;
      padding-top: 50px;
      font-weight: 700;
    }
  </style>
</head>
<body>
  <div class="our_about">
    <div class="circle">
      <p></p>
    </div>
  </div>
</body>
</html>
```

Project Documentation

```
h1{
  text-align: center;
  color: rgb(4, 4, 88);
}
video{
  padding: 5px;
}
.box{
  margin-bottom: 150px;
}
</style>
</head>
<body>
  <div class="our_about">
    <div class="circle">
      <p>Lecture</p>
    </div>
  </div>
  <main>
    <div class="box">
      <h1>Class 9th</h1>
      <video src="#" width="360px" height="400px" controls
poster="./upload.jpg"></video>
      <video src="#" width="360px" height="400px" controls
poster="./upload.jpg"></video>
      <video src="#" width="360px" height="400px" controls
poster="./upload.jpg"></video>
      <video src="#" width="370px" height="400px" controls
poster="./upload.jpg"></video>
    </div>
    <div class="box">
      <h1>Class 10th</h1>
```

Project Documentation

```
<video src="#" width="360px" height="400px" controls
poster="/upload.jpg"></video>
```

```
<video src="#" width="360px" height="400px" controls
poster="/upload.jpg"></video>
```

```
<video src="#" width="360px" height="400px" controls
poster="/upload.jpg"></video>
```

```
<video src="#" width="370px" height="400px" controls
poster="/upload.jpg"></video>
```

```
</div>
```

```
<div class="box"> <h1>NTSE/OLYMPIAD</h1>
```

```
<video src="#" width="360px" height="400px" controls
poster="/upload.jpg"></video>
```

```
<video src="#" width="360px" height="400px" controls
poster="/upload.jpg"></video>
```

```
<video src="#" width="360px" height="400px" controls
poster="/upload.jpg"></video>
```

```
<video src="#" width="370px" height="400px" controls
poster="/upload.jpg"></video>
```

```
</div>
```

```
</main>
```

```
</body>
```

```
</html>
```

LOG.HTML

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<title>Login_page</title>
```

```
<!-- css -->
```

```
<style>
```

Project Documentation

```
.box{
    margin: auto;
    /* margin-top: 150px;
    margin-left: 500px; */
    color: white;
    padding: 50px;
    width: 300px;
    height: 250px;
    background-color: rgb(4,4,80);
    font-size: 18px;
    border-radius: 6px;
}
button{
    color: rgb(4,4,80);
    background-color: rgb(191, 191, 191);
    width: 70px;
    height: 40px;
    margin-left: 50px;
    font-size: 16px;
}
input{
    height: 25px;
    border-radius: 6px;
}
a{
    color: white;
    font-size: 15px;
}
</style>
</head>
<body>
```

Project Documentation

```

<form action="check_login.php" method="post">
<div class="box">
  <table>
    <tr>
      <td><label for="username">User Name:</label></td>
      <td><input type="text" id="username" name="username"></td>
    </tr>
    <tr>
      <td><label for="pass">Password:</label></td>
      <td><input type="password" name="pass" id="pass"></td>
    </tr>
    <tr><td><button type="submit">Log in</button></td></tr>
    <br>
    <!-- <a href="./lognew.html" target="_blank">Create New Account</a> -->
  </table>
  I am new user. <a href="./lognew.html">Create New Account</a>
</div>
</form>
</body>
</html>
```

LOGNEW.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>New user Login_page</title>
  <!-- css -->
```


Project Documentation

```
<style>
  .box{
    margin: auto;      /* margin-top: 110px;
    margin-left: 500px; */
    color: white;
    padding: 50px;
    width: 300px;
    height: 300px;
    background-color: rgb(4,4,80);
    font-size: 18px;
    border-radius: 6px;
  }
  button{
    color: rgb(4,4,80);
    background-color: rgb(206, 206, 206);
    width: 120px;
    height: 40px;
    margin-left: 50px;
    font-size: 16px;
  }
  input{
    height: 25px;
    border-radius: 6px;
  }
  a{
    color: white;
    font-size: 15px;
  }
</style>
</head>
<body>
```


Project Documentation

```
body{
  background: url(./pexels-tirachard-kumtanom-733857.jpg)center/cover no-repeat;
  backdrop-filter: blur(90px);
  opacity: 0.9;
}
.box{
  display: flex;
  justify-content: space-around;
}
header{
  text-align: center;
  padding: 50px;
  color: rgb(4, 4, 119);
}
h1{
  font-size: 35px;
}
</style>
<!-- font link -->
<link rel="preconnect" href="https://fonts.googleapis.com">
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
<link href="https://fonts.googleapis.com/css2?family=Poppins&display=swap"
rel="stylesheet">
</head>
<body>
  <header>
    <h1>NOTES</h1>
  </header>
  <main>
    <!-- <section id="section1"> -->
    <div class="box">
```

Project Documentation

```
<div class="notes">
  <h2>Class 9th</h2>
  <ul>
    <li><a href="/.9th science.pdf">Science</a> </li>
    <li><a href="/.9th social science.pdf">Social Science</a></li>
    <li><a href="/.9th computer science.pdf">Computer</a></li>
    <li> <a href="/.9th math.pdf">Maths</a></li>
    <li> <a href="/.9th english.pdf">English</a></li>
  </ul>
</div>
```

```
<div class="notes">
  <h2>Class 10th</h2>
  <ul>
    <li><a href="/.10th science-1.pdf">Science</a> </li>
    <li><a href="/.10th social science.pdf">Social Science</a></li>
    <li><a href="/.10th computer science.pdf">Computer</a></li>
    <li><a href="/.10th math.pdf">Maths</a></li>
    <li><a href="/.10th english.pdf">English</a></li>
  </ul>
</div>
```

```
</div>
```

```
<header>
```

```
<h1>SAMPLE PAPER QUESTION</h1>
```

```
</header>
```

```
<div class="box">
```

```
<div class="pyq">
```

```
<h2>Class 9th</h2>
```

```
<ul>
```

```
<li><a href="">Science</a></li>
```

```
<li><a href="">Social Science</a></li>
```

```
<li><a href="">Computer</a></li>
```

Project Documentation

```
<li><a href="">Maths</a></li>
<li><a href="">English</a></li>
</ul>
</div>
<div class="pyq">
  <h2>Class 10th</h2>
  <ul>
    <li><a href="">Science</a></li>
    <li><a href="">Social Science</a></li>
    <li><a href="">Computer</a></li>
    <li><a href="">Maths</a></li>
    <li><a href="">English</a></li>
  </ul>
</div>
</div>
</main>
</body>
</html>
```

PROFILE.CSS

```
*{
  margin: 0;
  font-family: 'Poppins', sans-serif;
}
body{
  background-image: url(./dp.png)center/cover no-repeat;
  backdrop-filter: blur(90px);
  opacity: 0.9;
}
.box{
  height: 60px;
```

Project Documentation

```
width: 100%;
background-color: rgb(4,4,88);
}
.circle{
height: 120px;
width: 120px;
border-radius: 50%;
background-color: rgb(3, 3, 146);
color: white;
font-size: 25px;
text-align: center;
display: grid;
justify-content: center;
}
.circle p{
margin: auto;
}
.data{
display: flex;
justify-content: space-around;
margin-top: 100px;
}
.info, .sub{
font-size: 24px;
font-weight: 700;
color: rgb(7, 7, 185);
background-color: rgb(244, 244, 244);
padding: 60px;
border-radius: 6px;
}
.infoc{
```

Project Documentation

```
    color: red;
}
```

PROFILE.PHP

```
<?php
session_start();

if (!isset($_SESSION["username"])) {
    header("Location: log.html");
}
$username = $_SESSION["username"];
$conn = new mysqli("localhost", "root", "", "lvdata");
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
$sql = "SELECT * FROM appformtable WHERE username='$username'";
$result = $conn->query($sql);
?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Profile Page</title>
    <link rel="preconnect" href="https://fonts.googleapis.com">
    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
    <link href="https://fonts.googleapis.com/css2?family=Poppins&display=swap"
rel="stylesheet">
    <link rel="stylesheet" href="./profile.css">
</head>
<body>
```

Project Documentation

```
<div class="container mt-5">
  <div class="box">
    <div class="circle"><p>Profile</p> </div>
  </div>
  <div class="data">
    <?php if ($result->num_rows > 0): ?>
      <!-- Paragraph Display -->
      <?php $row = $result->fetch_assoc(); ?>
    <div class="info">
      <table>
        <tr>
          <td>Name: </td>
          <td></td>
          <td class="infoc"><?= $row['username'] ?></td>
        </tr>
        <tr>
          <td>Father's Name:</td>
          <td></td>
          <td class="infoc"><?= $row['fname'] ?></td>
        </tr>
        <tr>
          <td>Mother's Name:</td>
          <td></td>
          <td class="infoc"><?= $row['mname'] ?></td>
        </tr>
        <tr>
          <td>Date of Birth:</td>
          <td></td>
          <td class="infoc"><?= $row['dob'] ?></td>
        </tr>
      </table>
    </div>
  </div>
</div>
```


Project Documentation

```
<tr>
  <td>E-mail:</td>
  <td></td>
  <td class="infoc"><?=$row['email'] ?></td>
</tr>
<tr>
  <td>Adhaar Number:</td>
  <td></td>
  <td class="infoc"><?=$row['adhar'] ?></td>
</tr>
<tr>
  <td>Mobile Number:</td>
  <td></td>
  <td class="infoc"><?=$row['mob'] ?></td>
</tr>
<tr></tr>
<tr></tr>
<tr>
  <td>Street:</td>
  <td></td>
  <td class="infoc"><?=$row['street'] ?></td>
</tr>
<tr>
  <td>City:</td>
  <td></td>
  <td class="infoc"><?=$row['city'] ?></td>
</tr>
<tr>
  <td>District:</td>
  <td></td>
  <td class="infoc"><?=$row['district'] ?></td>
```

Project Documentation

```
</tr>
</table>
</div>
<div class="sub">
<table class="table mt-4">
  <thead>
    <tr>
      <th scope="col">Class</th>
      <th></th>
      <th></th>
      <th scope="col">Subject</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td class="infoc"><?= $row['userclass'] ?></td>
      <td></td>
      <td></td>
      <td class="infoc"><?= $row['usersub'] ?></td>
    </tr>
  </tbody>
</table>
<?php else: ?>
  <p>No profile data found.</p>
<?php endif; ?>
</div>
</div>
</div>
</body>
</html>
```

Project Documentation

```
<?php  
$conn->close();  
?>
```

QUIZ.HTML

```
<!-- quiz.html -->  
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <meta charset="UTF-8">  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  <title>Quiz</title>  
  <style>  
    .box{  
      height: 50px;  
      width: 100%;  
      background-color: rgb(4,4,88);  
    }  
    .circle{  
      height: 100px;  
      width: 100px;  
      border-radius: 50%;  
      background-color: rgb(5, 5, 131);  
      text-align: center;  
      /* margin: auto; */  
      display: flex;  
      align-items: center;  
    }  
    .circle p{  
      margin: auto;  
      font-size: 20px;
```

Project Documentation

```
    color: white;
}
#quiz-container{
    /* text-align: center; */
    margin: auto;
    font-size: 25px;
    color: rgb(4,4,88);
    width: 500px;
}
#result-container{

    /* text-align: center; */
    margin: auto;
    font-size: 25px;
    color: rgb(4,4,88);
    width: 500px;
}
.btn{
    display: flex;
    justify-content: center;
}
button{
    margin: auto;
    height: 50px;
    width: 150px;
    border-radius: 7px;
    font-size: 15px;
    background-color: rgb(4,4,88);
    color: white;
}
ul li{
```

Project Documentation

```
        list-style: none;
    }
</style>
</head>
<body>
    <div class="box">
        <div class="circle">
            <p>9th Quiz</p>
        </div>
    </div>
    <div id="quiz-container"></div>
    <div class="btn"><button onclick="submitQuiz()">Submit Answers</button></div>
    <div id="result-container"></div>
    <script>
        let questions = [];

        // Use AJAX or Fetch API to get questions from the server
        fetch('quiz.php')
            .then(response => response.json())
            .then(data => {
                questions = data;
                // Use the questions to dynamically generate the quiz UI
                displayQuiz();
            })
            .catch(error => console.error('Error fetching questions:', error));

        function displayQuiz() {
            const quizContainer = document.getElementById('quiz-container');
            questions.forEach((question, index) => {
                const questionElement = document.createElement('div');
                questionElement.innerHTML = `
                    <p>${index + 1}. ${question.question}</p>
                `
            })
        }
    </script>
</body>
</html>
```

Project Documentation

```
<ul>
  ${question.options.map((option, optionIndex) => `
    <li>
      <input type="radio" name="q${index}" value="${optionIndex}">
      ${option}
    </li>`)}
</ul>
`;
quizContainer.appendChild(questionElement);
});
}
function submitQuiz() {
  const resultContainer = document.getElementById('result-container');
  let score = 0;
  questions.forEach((question, index) => {
    const selectedOptionIndex =
document.querySelector(`input[name="q${index}"]:checked`);
    if (selectedOptionIndex) {
      const selectedOption = parseInt(selectedOptionIndex.value);
      if (question.options[selectedOption] === question.correctAnswer) {
        score++;
      }
    }
  });
  const resultMessage = `You scored ${score} out of ${questions.length}.`;
  resultContainer.innerHTML = `<p>${resultMessage}</p>`;
}
</script>
</body>
</html>
```

Project Documentation

QUIZ.PHP

```
<?php
// Mock data for demonstration purposes
$allQuestions = [
    ["question" => "What is the atomic number of carbon?", "options" =>["5","12","6","14"],
    "correctAnswer" => "6"],
    ["question" => "Which process is responsible for the formation of clouds in the
    atmosphere?", "options"
    =>["Evaporation","Condensation","Sublimation","Precipitation"],"correctAnswer"=>"Condens
    ation"],
    ["question"=> "What is the SI unit of force?",
    "options"=>["Joule","Newton","Watt","Pascal"], "correctAnswer"=> "Newton"],
    ["question"=> "What is the chemical formula of water is?",
    "options"=>["CO2","O2","H2O","CH4"], "correctAnswer"=> "H2O"],
    ["question"=> "In human body which organ produces insulin",
    "options"=>["Heart","Lungs","Liver","Pancreas"], "correctAnswer"=> "Pancreas"],
    ["question"=> "What is the unit of Electric Current?",
    "options"=>["Volt","Ohm","Ampere","Pascal"], "correctAnswer"=> "Ampere"],
    ["question" => "What is the atomic number of oxygen?", "options" => ["6", "8", "10",
    "12"], "correctAnswer" => "8"],
    ["question" => "Which planet is known as the Red Planet?", "options" => ["Earth", "Mars",
    "Venus", "Jupiter"], "correctAnswer" => "Mars"],
    ["question" => "Which planet have life?", "options" => ["Earth", "Mars", "Venus",
    "Jupiter"], "correctAnswer" => "Earth"],
    ["question" => "What is the capital of India?", "options" => ["Paris", "New Delhi", "Rome",
    "Madrid"], "correctAnswer" => "Paris"],
    // ... add more questions ...
];
shuffle($allQuestions); // Shuffle the array to get random questions
$selectedQuestions = array_slice($allQuestions, 0, 5); // Select the first 10 questions
echo json_encode($selectedQuestions);
?>
```

QUIZ10.HTML

```
<!-- quiz.html -->
```

Project Documentation

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Quiz</title>
  <style>
    .box{
      height: 50px;
      width: 100%;
      background-color: rgb(4,4,88);
    }
    .circle{
      height: 100px;
      width: 100px;
      border-radius: 50%;
      background-color: rgb(4, 4, 127);
      text-align: center;
      /* margin: auto; */
      display: flex;
      align-items: center;
    }
    .circle p{
      margin: auto;
      font-size: 20px;
      color: white;
    }
    #quiz-container{
      /* text-align: center; */
      margin: auto;
      font-size: 25px;
```


Project Documentation

```
    color: rgb(4,4,88);
    width: 500px;
}
#result-container{
    /* text-align: center; */
    margin: auto;
    font-size: 25px;
    color: rgb(4,4,88);
    width: 500px;
}
.btn{
    display: flex;
    justify-content: center;
}
button{
    margin: auto;
    height: 50px;
    width: 150px;
    border-radius: 7px;
    font-size: 15px;
    background-color: rgb(4,4,88);
    color: white;
}
ul li{
    list-style: none;
}
</style>
</head>
<body>
    <div class="box">
    <div class="circle">
```

Project Documentation

```
<p>10th Quiz</p>
</div>
</div>
<div id="quiz-container"></div>
<div class="btn"><button onclick="submitQuiz()">Submit Answers</button></div>
<div id="result-container"></div>
<script>
  let questions = [];

  // Use AJAX or Fetch API to get questions from the server
  fetch('quiz10.php')
    .then(response => response.json())
    .then(data => {
      questions = data;
      // Use the questions to dynamically generate the quiz UI
      displayQuiz();
    })
    .catch(error => console.error('Error fetching questions:', error));
  function displayQuiz() {
    const quizContainer = document.getElementById('quiz-container');
    questions.forEach((question, index) => {
      const questionElement = document.createElement('div');
      questionElement.innerHTML = `
        <p>${index + 1}. ${question.question}</p>
        <ul>
          ${question.options.map((option, optionIndex) => `
            <li>
              <input type="radio" name="q${index}" value="${optionIndex}">
                ${option}
            </li>`
            ).join("")}
        </ul>
      `;
    });
  }
</script>
```

Project Documentation

```
`;  
    quizContainer.appendChild(questionElement);  
});  
}  
function submitQuiz() {  
    const resultContainer = document.getElementById('result-container');  
    let score = 0;  
    questions.forEach((question, index) => {  
        const selectedIndex =  
document.querySelector(`input[name="q${index}"]:checked`);  
        if (selectedIndex) {  
            const selectedOption = parseInt(selectedOptionIndex.value);  
            if (question.options[selectedOption] === question.correctAnswer) {  
                score++;  
            }  
        }  
    });  
  
    const resultMessage = `You scored ${score} out of ${questions.length}.`;   
    resultContainer.innerHTML = `

${resultMessage}</p>`;  
}  
</script>  
</body>  
</html>


```

QUIZ10.PHP

```
<?php  
// Mock data for demonstration purposes  
$allQuestions = [  
    ["question" => "What is the atomic number of carbon?", "options" =>["5","12","6","14"],  
    "correctAnswer" => "6"],
```

Project Documentation

```
["question"=> "What is the chemical formula of methane is?",  
"options"=>["CO2","O2","H2O","CH4"], "correctAnswer"=> "CH4"],  
  
["question"=> "How many laws of motion?", "options"=>["1","2","3","4"],  
"correctAnswer"=> "3"],  
  
["question"=> "Which scientist proposed the laws of motion?", "options"=>["Isaac  
Netwon","Albert Einstein","Galileo Galilei","Nikola Tesla"], "correctAnswer"=> "Isaac  
Newton"],  
  
["question" => "Which process is responsible for the formation of clouds in the  
atmosphere?", "options"  
=>["Evaporation","Condensation","Sublimation","Precipitation"],"correctAnswer"=>"Condens  
ation"],  
  
["question"=> "What is the SI unit of force?",  
"options"=>["Joule","Newton","Watt","Pascal"], "correctAnswer"=> "Newton"],  
  
["question"=> "What is the chemical formula of water is?",  
"options"=>["CO2","O2","H2O","CH4"], "correctAnswer"=> "H2O"],  
  
["question"=> "In human body which organ produces insulin",  
"options"=>["Heart","Lungs","Liver","Pancreas"], "correctAnswer"=> "Pancreas"],  
  
["question"=> "What is the unit of Electric Current?",  
"options"=>["Volt","Ohm","Ampere","Pascal"], "correctAnswer"=> "Ampere"],  
  
["question" => "What is the atomic number of oxygen?", "options" => ["6", "8", "10",  
"12"], "correctAnswer" => "8"],  
  
["question" => "Which planet is known as the Red Planet?", "options" => ["Earth", "Mars",  
"Venus", "Jupiter"], "correctAnswer" => "Mars"],  
  
["question" => "Which planet have life?", "options" => ["Earth", "Mars", "Venus",  
"Jupiter"], "correctAnswer" => "Earth"],  
  
["question" => "What is the capital of India?", "options" => ["Paris", "New Delhi", "Rome",  
"Madrid"], "correctAnswer" => "Paris"],  
  
// ... add more questions ...  
];  
shuffle($allQuestions); // Shuffle the array to get random questions  
$selectedQuestions = array_slice($allQuestions, 0, 5); // Select the first 10 questions  
echo json_encode($selectedQuestions);  
?>
```

STYLE3.CSS

```
*{
```

Project Documentation

```
margin: 0px;
font-family: 'Poppins', sans-serif;
}
/* navigation */
/* heading */
.our_about{
background-color: rgb(4, 4, 88);
height: 70px;
width: 100%;
}
.circle{
background-color: rgb(3, 3, 131);
color: white;
width: 160px;
height: 160px;
border-radius: 50%;
}
.circle p{
text-align: center;
font-size: 30px;
padding-top: 50px;
font-weight: 700;
}
/* body of about */
.title_about{
display: flex;
justify-content: space-around;
color: rgb(4, 4, 88);
}
.title_about img{
background-color:#ffffff;
```

Project Documentation

```
    height: 350px;
    width: 350px;
    padding: 20px;
}
.body_about{
    width: 400px;
    padding: 40px;
}
.body_about span{
    font-size: 30px;
    font-weight: 700;
    padding: 20px;
}
.body_about p{
    font-size: 17px;
    padding: 20px;
}
/* values */
.values{
    display: flex;
    justify-content: space-evenly;
    padding: 50px;
}
.val{
    padding: 30px;
    font-size: 18px;
    margin: 20px;
}
.val span{
    font-size: 20px;
    font-weight: 700;
```

Project Documentation

```
    color: rgb(4, 4, 88) ;
}
/* teacher */
/* each teacher */
.teacher{
    /* width: 800px; */
    border: solid 2px rgb(3, 108, 169);
    padding: 50px;
    border-top-left-radius: 18px;
    border-bottom-right-radius: 18px;
}
.teacher img{
    border-radius: 50%;
    width: 100px;
    height: 100px;
}
/* all teacher */
.team_det1{
    display: flex;
    justify-content: space-around;
    padding: 50px;
}
/* name of teacher */
div .name{
    font-size: 20px;
    font-weight: 800;
    padding: 10px;
}
.values p, .teacher p{
    color: rgb(5, 46, 136);
    font-size: 18px; }
```

Project Documentation

```
.last_part{
  background-color: rgb(4, 4, 88);
  font-size: 15px;
  padding: 30px;
  margin-top: 20px;
  color: azure;
  border-radius: 1px;
}

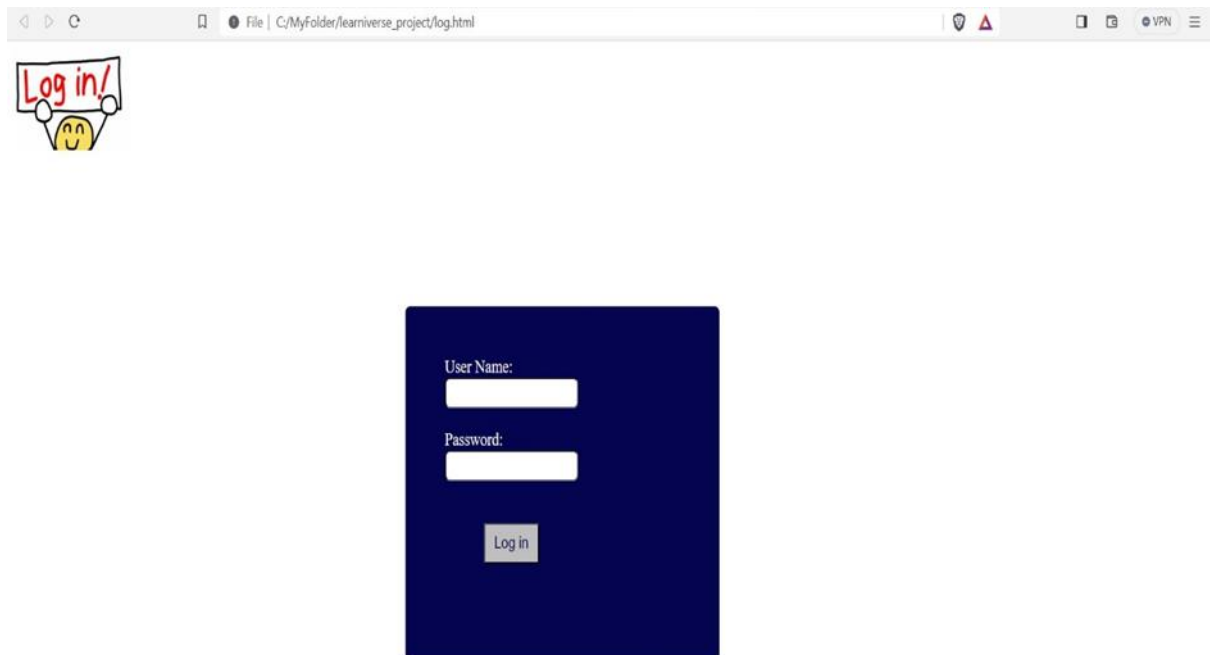
.footer_contact{
  color: azure;
  font-size: 15px;
  padding: 30px;
  margin-top: 20px;
  color: azure;
  border-radius: 1px;
}

footer{
  display: flex;
  background-color: rgb(4, 4, 88);
  font-size: 20px;
  padding: 30px;
  margin-top: 20px;
  color: azure;
  border-radius: 1px;
  justify-content: space-around;
}
```

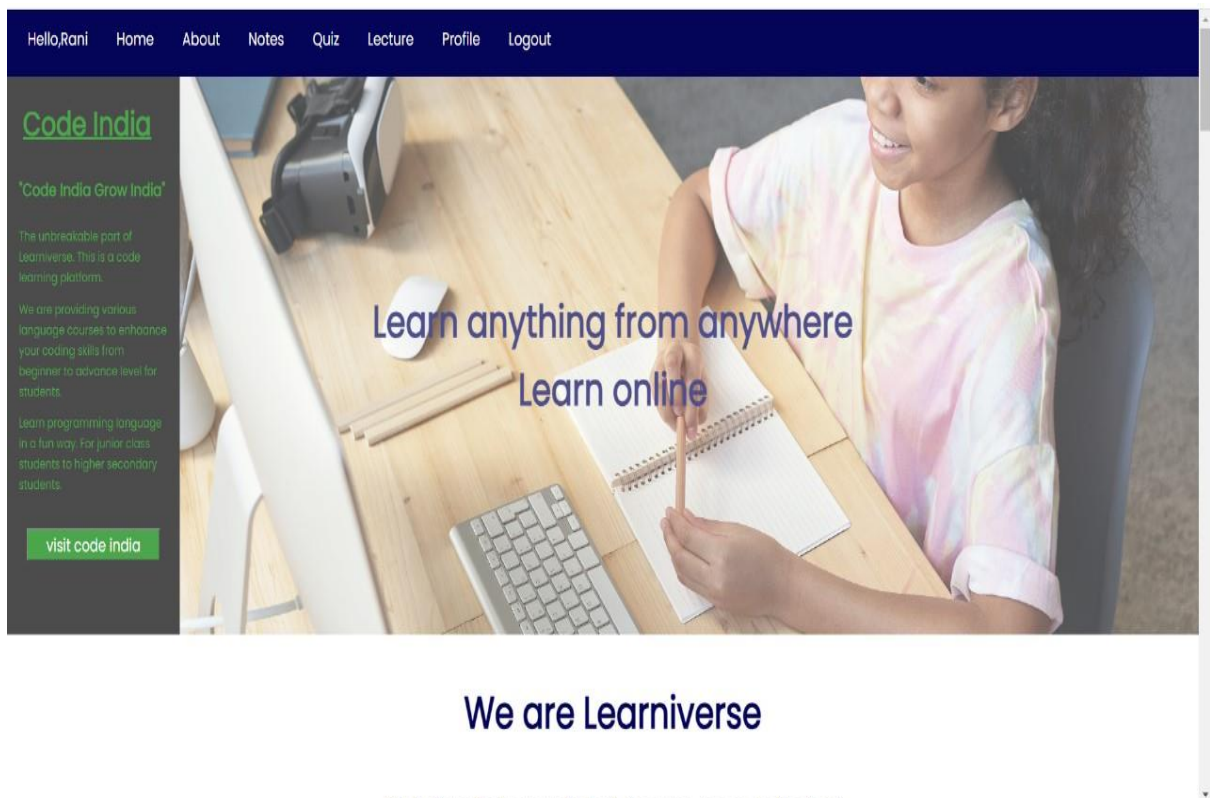

Project Documentation

Project Snapshot

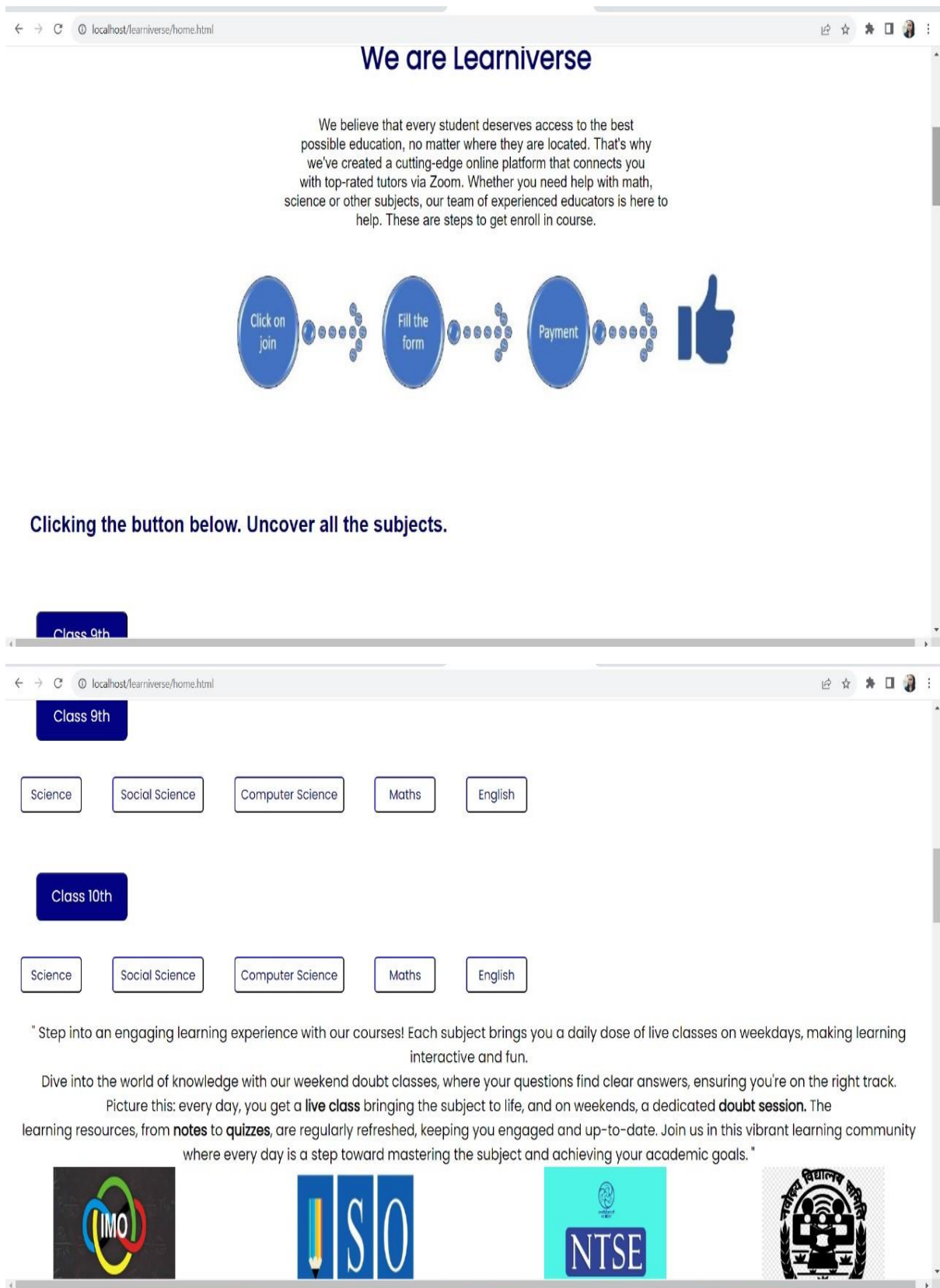
Login Page Snapshot:



Home Page Snapshot:



Project Documentation



Project Documentation

Our Batch Snapshot:

Batch class 9th

SCIENCE
Batch Timing: 4 PM - 5PM
Monday - Tuesday - Wednesday
Fees: Rs.500
[JOIN](#)

SOCIAL SCIENCE
Batch Timing: 4 PM - 5PM
Thursday - Friday - Saturday
Fees: Rs.500
[JOIN](#)

COMPUTER SCIENCE
Batch Timing: 4 PM - 5PM
Monday - Tuesday - Wednesday

Concepts of

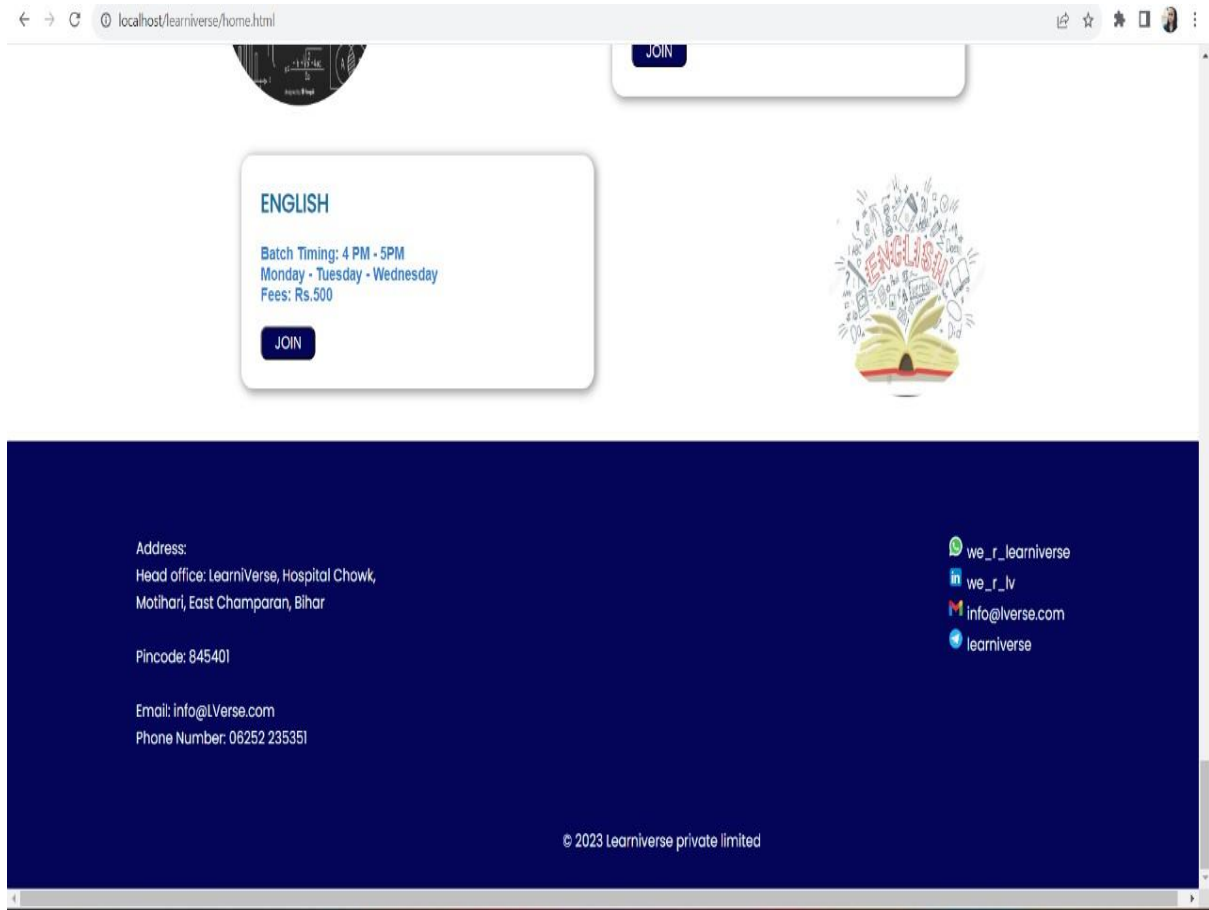
Batch class 10th

SCIENCE
Batch Timing: 4 PM - 5PM
Monday - Tuesday - Wednesday
Fees: Rs.500
[JOIN](#)

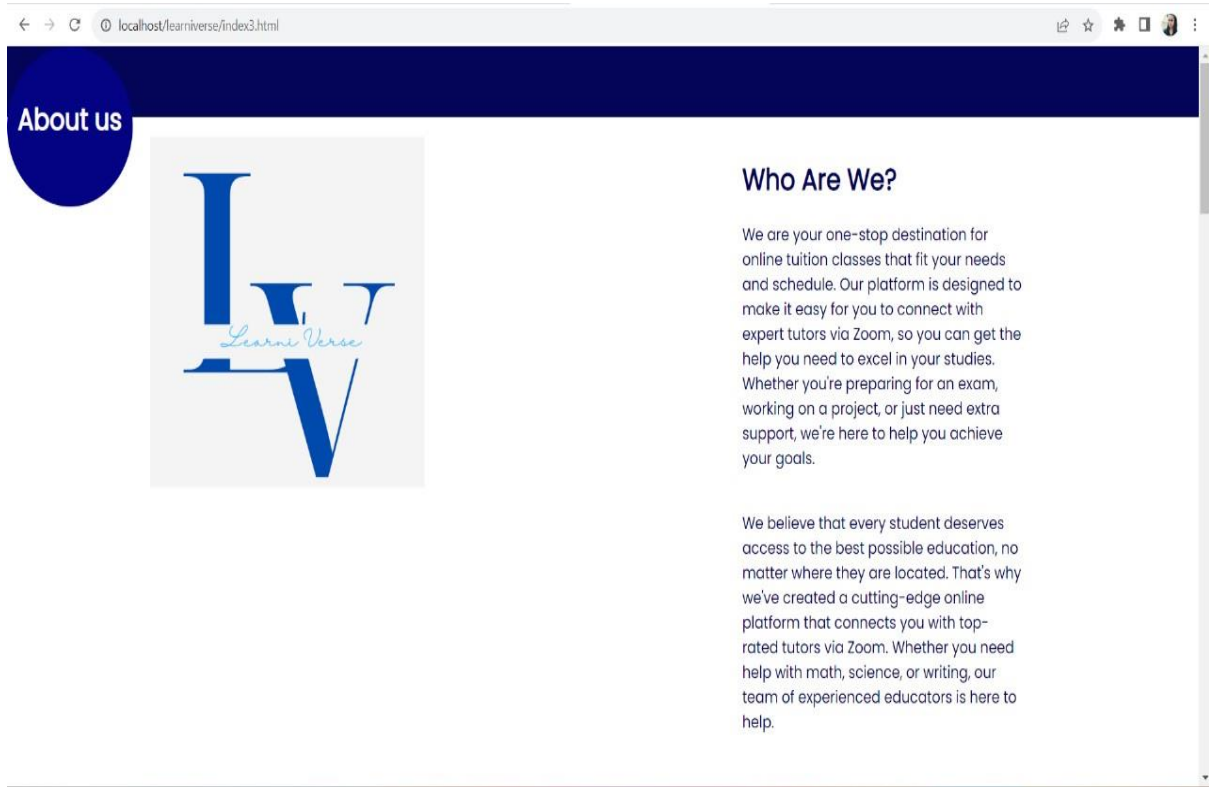
SOCIAL SCIENCE
Batch Timing: 4 PM - 5PM
Thursday - Friday - Saturday
Fees: Rs.500
[JOIN](#)

COMPUTER SCIENCE
Batch Timing: 4 PM - 5PM

Project Documentation

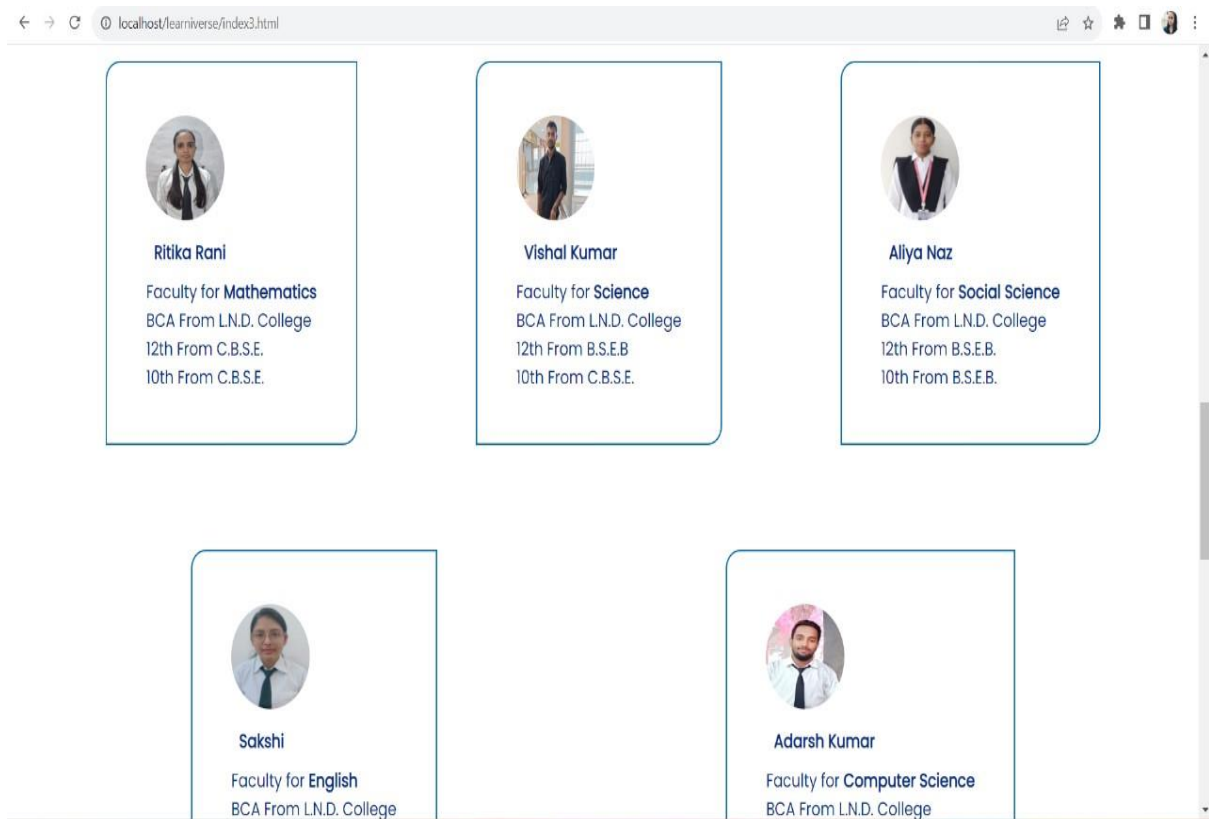


About Page:

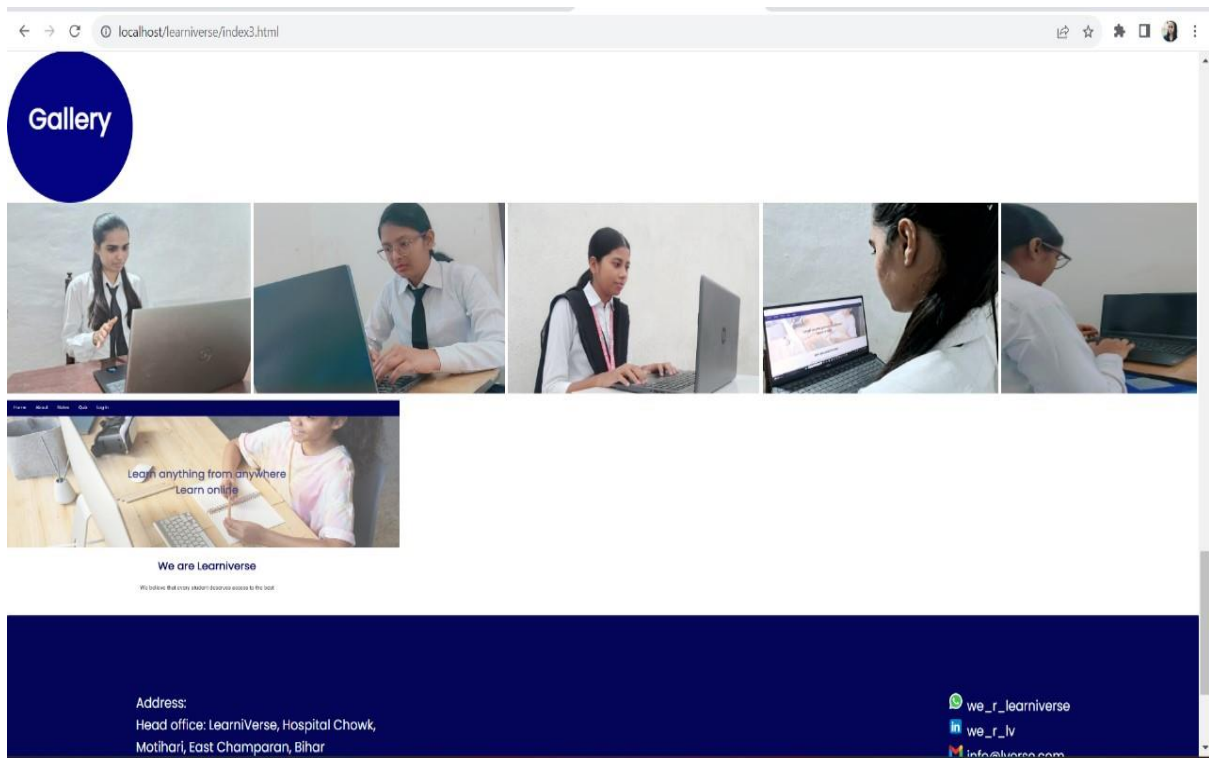


Project Documentation

Our Teams Snapshot:

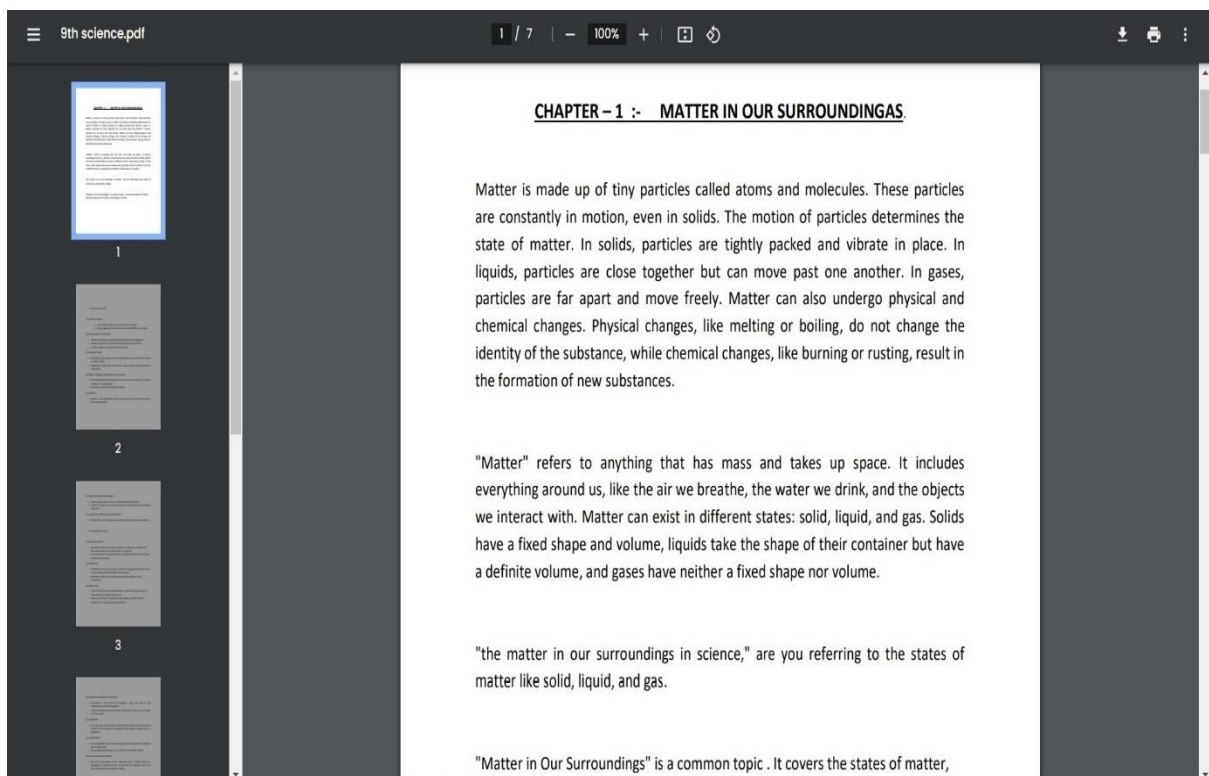
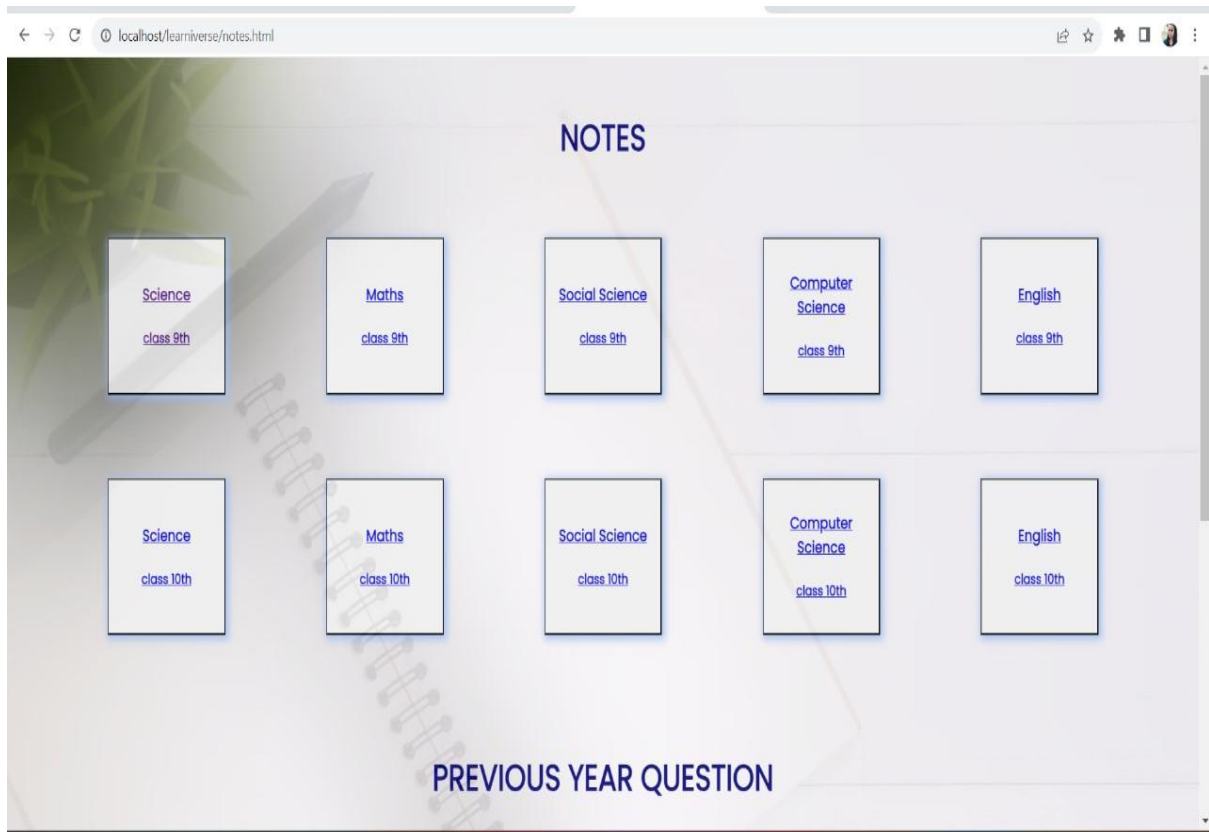


Gallery Snapshot:



Project Documentation

Notes Page Snapshot:



Project Documentation

Quiz Page Snapshot:

9th Quiz

1. What is the atomic number of oxygen?
 - 6
 - 8
 - 10
 - 12
2. What is the capital of India?
 - Paris
 - New Delhi
 - Rome
 - Madrid
3. Which process is responsible for the formation of clouds in the atmosphere?
 - Evaporation
 - Condensation
 - Sublimation
 - Precipitation
4. In human body which organ produces insulin

Lecture page snapshot:

Lecture

Class 9th

Content will be uploaded Soon!

Content will be uploaded Soon!

Content will be uploaded Soon!

Content will be uploaded Soon!

Project Documentation

Profile Snapshot:

Profile

Name: Rani
Father's Name: asdff
Mother's Name: cjhvhv
Date of Birth: 2023-12-13
E-mail: ri@gma.com
Adhaar Number: 23445566
Mobile Number: 12334455
Street: main road
City: dzsfdzfg
District: axskcdsljf

Class Subject
10 maths

Application form Snapshot:

Application Form

Personal Detail

Name:
Father's Name:
Mother's Name:
E-mail:
Adhar Number:
Mobile No.:
Date of birth:

Course Detail

Class:
Subject:

Residential Detail

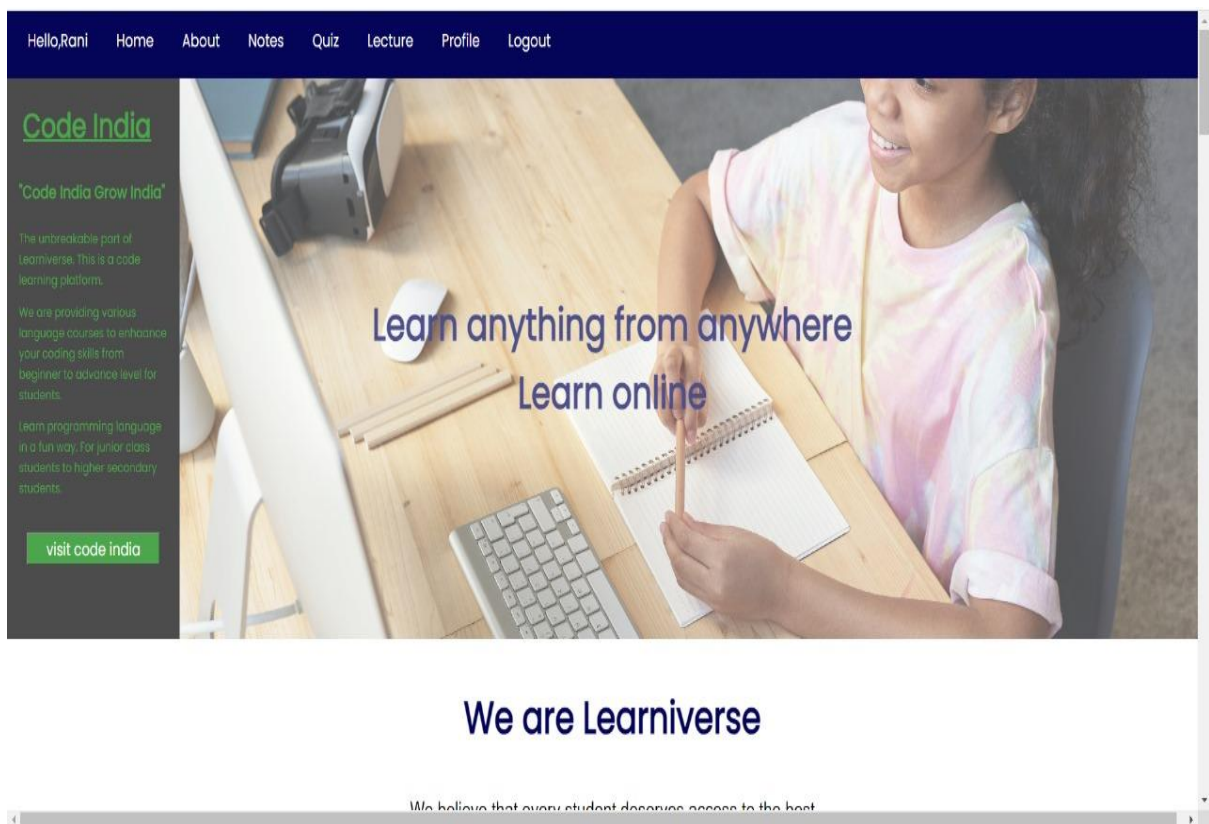
Street:
City:
Pincode:
District:
State:

Project Documentation

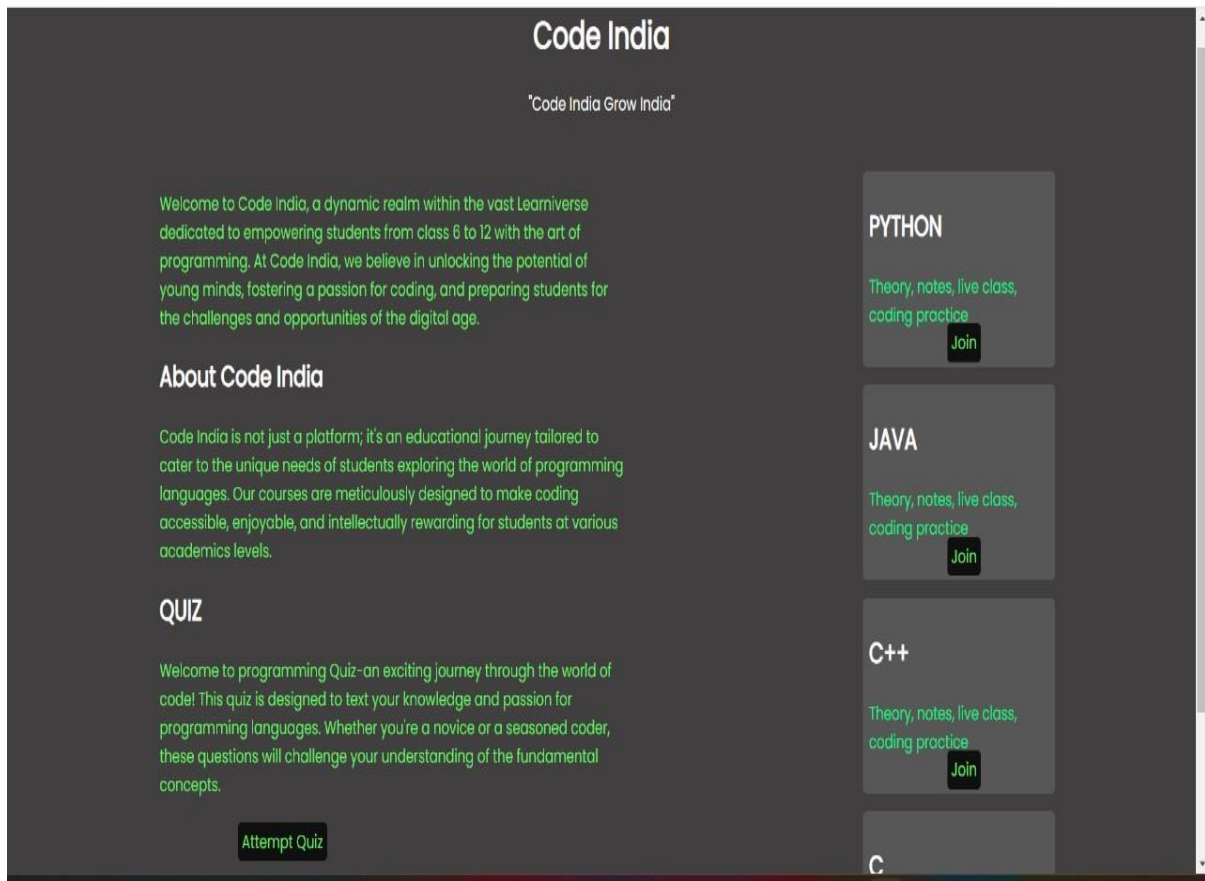
The screenshot shows a web browser window with the address bar displaying 'localhost/learniverse/appl.html'. The page contains a registration form with the following sections:

- State:** A text input field with the word 'State' inside.
- Payment:** A section with the heading 'Payment' and the text: "Please scan the UPI QR code, transfer an amount of Rs.500, and kindly provide the transaction ID in the text box below. If any error occurs then your registration will get cancelled and amount will get refund to your account. Thank you!". Below this text is a QR code for a UPI payment, with the name 'Rishika Rani' and the amount '₹ 500.00' visible. Below the QR code is a text input field labeled 'Transaction Id:'.
- Declaration:** A section with the heading 'Declaration' and a checkbox. The text reads: "I, declare that both my parents, are aware of and consent to the submission of my application form. We have thoroughly verified all the details in the application, including payment information, before submission." Below this text are two buttons: 'Reset' and 'Submit'.

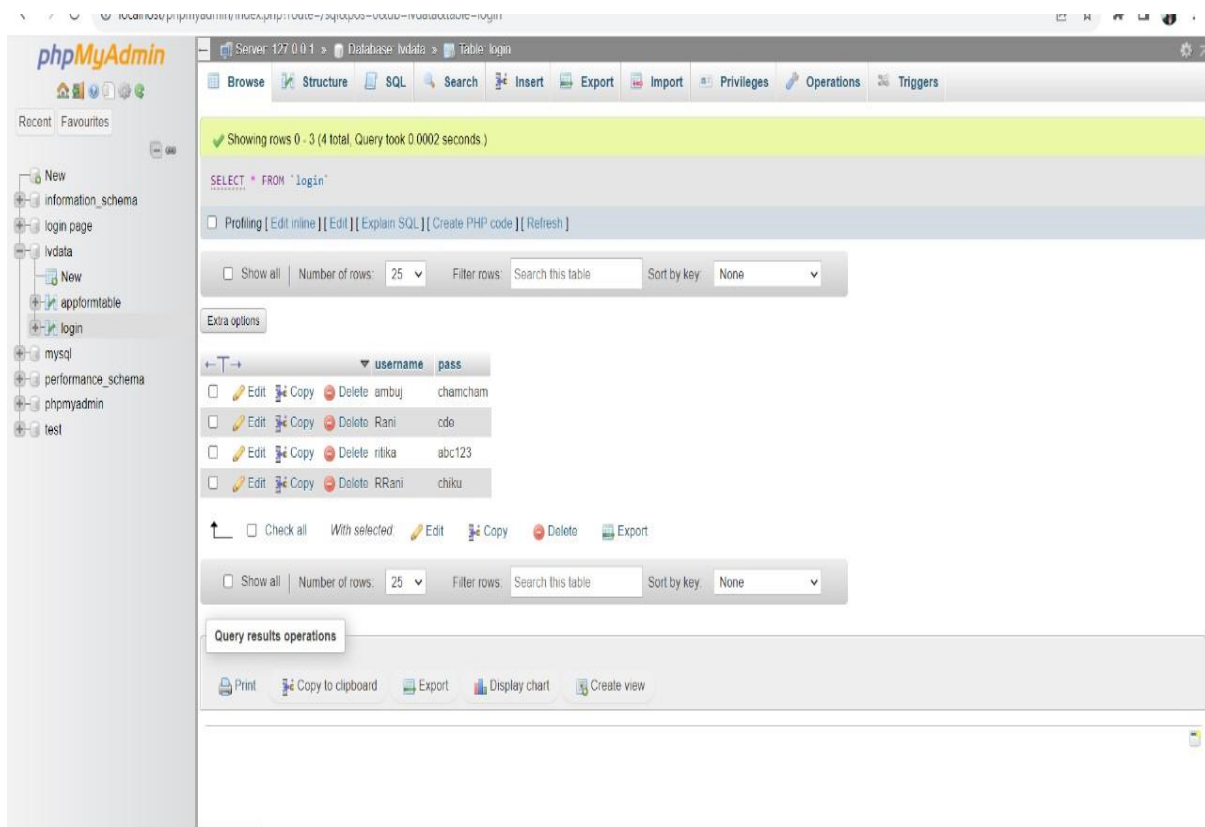
Code India Page Snapshot:



Project Documentation



Database page Snapshot:



Project Documentation

The screenshot displays the phpMyAdmin interface for a table named 'appformtable' in the 'lvdata' database. The interface includes a navigation sidebar on the left, a top menu with options like 'Browse', 'Structure', 'SQL', 'Search', 'Insert', 'Export', 'Import', 'Privileges', 'Operations', and 'Triggers', and a main content area. A warning message states: 'Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.' Below this, a green bar indicates 'Showing rows 0 - 6 (7 total. Query took 0.0003 seconds)'. The SQL query shown is 'SELECT * FROM `appformtable`'. There are controls for 'Show all', 'Number of rows' (set to 25), and a 'Filter rows' search box. A section for 'Extra options' is visible above the table. The table itself has 14 columns: 'username', 'fname', 'mname', 'email', 'adhar', 'mob', 'dob', 'userclass', 'usersub', 'street', 'city', 'pincode', 'district', and 'state'. It contains 7 rows of user data. Below the table, there are more 'Show all', 'Number of rows', and 'Filter rows' controls, and a 'Query results operations' section with buttons for 'Print', 'Copy to clipboard', 'Export', 'Display chart', and 'Create view'.

username	fname	mname	email	adhar	mob	dob	userclass	usersub	street	city	pincode	district	state
RRani	sunil	anamika	rit4933@gmail.com	2147483647	2147483647	2023-06-09	on	on	main road	motihari	84		
sima	sunil	anamika	rit4933@gmail.com	123456784321	7070050529	2023-06-09	on	on	main road	motihari	84		
ambuj	sunil	kr	ambuj@gmail.com	876512341234	9931873253	2023-11-18	on	on	main road	dzsfldzlg	845401	axskcdsijf	bihar
RRani	papa	maa	rit4933@gmail.com	123412341234	7070050529	2023-11-11	10	maths	main road	motihari	845401	axskcdsijf	bihar
chiku	sunil	kr	Vis928@gmail.com	456745674567	7070050529	2023-11-06	9	computersci	main road	dzsfldzlg	845477	axskcdsijf	up
Rani	asdf	cjrhv	ri@gmail.com	23445566	12334455	2023-12-13	10	maths	main road	dzsfldzlg	845401	axskcdsijf	yukh
adarsh	fuhgvkug	fugfvhj	fyugfn@gfhg	12344555	1223445	2023-12-29	10	english	cfh	motihari	65879	axskcdsijf	bihar

Project Documentation

CHAPTER 6: CONCLUSION

In the culmination of this educational website project, we arrive at a juncture that marks the realization of our initial vision and objectives. Over the course of its development, the website has evolved into a robust platform designed to address the unique needs of our target audience – be they students, educators, or administrators. This section serves as a comprehensive summary, encompassing the achievements, challenges, and impact of our endeavour. It is an opportunity to underscore the significance of the educational website in the broader context of modern education, acknowledging the strides made and laying the foundation for future enhancements. Join us in revisiting the key milestones and insights gained during this transformative venture.

SUMMARY OF ACHIEVEMENTS:

- Students get benefit of live classes with their mentors, with provide them a live interaction without hesitation. Every doubt of students gets easily solved.
- Provides many important links of exams and also the contents which is helps them to compete the exams.
- Provides them notes and sample papers for all the subjects. Students can download notes in their system without any problem.
- Daily quizzes are arranged for students. These quizzes daily get updated and they are built on google forms.
- Student can purchase course through completing the application form and payment then after verification from our side they get the invoice and daily email for live class link.

CHALLENGES AND SOLUTIONS:

We get challenge when we are trying to add the payment in application form. The payment Gateway is the mode which is used to transfer money online. Since it is the paid feature and we are students who are making a college project it is difficult for us to purchase that feature.

To resolve that payment gateway problem, we get an idea of adding the UPI QR code of on which can get payment by clients. Then after we take their transaction id as input and in our database all their data get store including that transaction id. The transaction id we check and if it correctly matches with our transaction id then that client gets an email from us of invoice. If the transaction is wrong then we send them an email which says that again apply the form with the correct old transaction id if they had performed the transaction correctly if not then send the money once again.

Project Documentation

USER FEEDBACK AND TESTING:

A user gives feedback that this website perfectly runs as it can get updated on time to time. A user can open the website, can apply the form, can login, can gets notes, can solve quiz without any problem.

LEARNING AND DEVELOPMENT:

This project teaches us many things but few things are mention below

- Team Management is the key role which lead us to get complete the project without any problem.
- Frontend and Backend work together to get a full running Web Application.
- Knowledge of many exams and topics which is related to educational field.

ACKNOWLEDGMENTS:

Our mentor, advisor and guide throughout the journey is Prabhat Kumar. He is our faculty in BCA department who guide us at each step where we feel demotivated and we were confuse. Some guidance of senior was also remarkable.

Conclusion Statement

All the achievements are perfectly added to our web application. It is a get success since this website gives a start for our small startup and lead us to become like big education platform.

Making students more leaning toward study by ppt, diagrams and etc. Prepare their foundation for big exams at school level. Help them to gain confidence by giving competition exams.

Project Documentation

REFERENCES

- **Mozilla Developer Network (MDN)**

Website: Mozilla Developer Network

- **HTML5, CSS3, JAVASCRIPT, PHP BLACK BOOK**

- **PW WEB DEVELOPMENT COURSE**

- **W3Schools**

Website: W3Schools

- **PEXELS (IMAGE SOURCE)**

- **GOOGLE FONTS(FONT LINKS)**

- **YOUTUBE (THAPA TECHNICAL FOR XAMPP)**