

Roadx

GRADUATION PROJECT

PREPARED BY

Mohamed Osman Ebn Afan Mohamed	20-01317
Omar Ali Amer Mohamed	20-01300
Mohamed Mostafa Mohamed Ahmed	20-00061
Mohamed Khaled Mohamed Sayed	20-01013
Mostafa Gaber Ahmed Hussien	20-00540
Nermeen Khaled Mohamed Elkholisy	20-01330
Shaza Ayman Amin Ahmed	20-00457

SUPERVISED BY

Dr. Safi Ibrahim
Ta. Rofida abdelhafez

A Graduation Project Report Submitted for the Partial Fulfillment of the Requirements of the B.SC. Degree

Acknowledgments

I would like to thank God Almighty for giving me the confidence and ability to pursue our project. We are glad that our educational journey, has been an amazing experience with a lot of learning curves.

I would like to thank Dr. Safi ibrahim for being an amazing doctor and a strong teacher with a lot of experience. She helped us to choose the courses and we thank her for her continuous support and choose the courses and we thank her for her continuous support and follow-up during the implementation process of this project.

How much we extend our sincere thanks and appreciation to our supervisor, Eng Rofida abd-elhafez for her tremendous efforts and exceptional support that she gave us during our work on the project and for providing means of assistance at any time during the implementation of our graduation project.

Table of contents

1.	Chapter 1	6
1.1.	History.....	7
1.2.	Motivation.....	9
1.3.	Problem statement.....	11
1.4.	Problem solutions.....	13
1.5.	Project Phases.....	15
1.6.	Requirements.....	16
2.	Chapter 2	20
2.1	Accessibility.....	21
2.2.	Use Cases.....	22
2.3	Importance	25
2.4	How to study.....	27
2.5	Just idea.....	30
3.	Chapter 3	31
3.1	Project components.....	32
3.2	project prototype.....	34
3.3	Design.....	53
3.4	Tools.....	108
4.	Chapter 4	115
4.1	Related works.....	116
4.2	clarification of Road x.....	117
4.3	Traditional VS progressive education.....	119
5.	Chapter 5	122
5.1.	Main page	123
5.2.	Home page.....	143
5.3.	Categories page	149
5.4.	Chatbot page	156
5.5.	Documentation page	161
5.6.	Extra knowledge page	196
5.7.	Programming languages page	201
5.8.	Learning Kids page	205
5.9.	Quizzes page.....	214
5.10.	Machine Learning	220

6. Chapter 6.....	245
6.1 Important Packages in flutter.....	246
7. Chapter 7.....	250
7.1.conclusion.....	251
8. Chapter 8.....	253
8.1 Future Works.....	254
9. Chapter 9	256
9.1.References	257

Introduction

A roadmap is a strategic planning tool that provides a visual representation of the goals and to achieve a desired outcome.

It defines the path forward and guides the decision-making process.

Roadmaps are not static documents and are subject to change as circumstances evolve.

Regular reviews and updates are necessary to reflect new information, changing priorities, and emerging opportunities or challenges.

A road map is a tool that helps you plan your journey to success. It's a visual representation of your goals, strategies, and obstacles. A road map can make all the difference in achieving your desired outcome.

Without a road map, you may find yourself lost, unsure of which direction to take.

Chapter 1

1.1.History

The concept of a "learning roadmap" has evolved significantly over time, particularly in the context of education and professional development. Here's a brief history of how the idea of learning road-maps has developed.

Traditional Education Models:

Historically, formal education followed a relatively linear path, with students progressing from one grade level to the next, guided by a standardized curriculum. While this provided a structured learning experience, it often lacked customization to individual student needs and interests.

Integration of Learning Analytic: Recent developments in learning analytics have further enhanced the effectiveness of learning roadmaps, By tracking learner behavior, engagement, and performance data, educators and organizations can continuously refine and optimize learning pathways to maximize effectiveness and learner outcomes.

Tracking Long-term Progress: By tracking learner behavior and performance data over time, learning analytics enable educators and organizations to monitor long-term progress towards learning goals.

This longitudinal data can inform decisions about future learning pathways and interventions,

ensuring that learners continue to make progress towards mastery and proficiency

1.2 Motivation

The motivation behind creating learning roadmaps for programming tracks stems from several key factors:

Complexity of Programming: Programming can be a complex and multifaceted skill to learn. There are numerous programming languages, frameworks, libraries, and concepts to master, and the landscape is constantly evolving. A roadmap helps break down this complexity into manageable steps, guiding learners through the foundational concepts to more advanced topics.

Clear Learning Path: Learning roadmaps provide learners with a clear path to follow, outlining the sequence of topics and skills they need to master. This clarity helps learners understand where to start, what to focus on next, and how to progress towards their learning goals.

Community and Support: Learning roadmaps often come with resources, such as tutorials, courses, books, and online communities, to support learners along their journey. These resources provide guidance, encouragement, and assistance when learners encounter challenges or have questions about specific topics.

1.3 Problem statement

There are many problems that most information technology students face, such as :

They cannot choose the appropriate field for them, and some others could not complete the path they choose.

Some students may face difficulty in accessing the resources needed to study some of the fields they want to study, such as the front-end field, the back-end field and the Android field.

There are difficulties in dealing with programming languages and software development .

Complex concepts can be difficult to understand and put into practice. It requires good learning to program.

Complexity and Depth: Learning programming involves understanding abstract concepts, syntax,

logic, and problem-solving techniques. The sheer complexity and depth of programming languages and technologies can be overwhelming for beginners.

There is a lot of time wasting in searching about resources for programming tracks , searching in websites for good instructors.

Rapid Technological Changes: The field of programming is constantly evolving, with new languages, frameworks, and tools being introduced regularly, making it challenging to keep up with the latest trends.

Lack of Practice: Programming requires regular practice to become proficient, and finding meaningful projects or exercises to work on can be challenging.

Fear of Failure: Programming can be intimidating, and the fear of making mistakes or not understanding concepts can hinder learning progress.

1.4 Problem solution

Resource Accessibility: Provide students with access to a wide range of resources, including online tutorials, books, and educational websites, to help them learn front-end, back-end, and Android development.

Online Learning Platforms: Utilize online learning platforms such as Coursera, Udemy, and Khan Academy to offer courses on programming languages and software development, making them accessible to students.

Feedback and Support: Offer feedback and support to students as they navigate their studies, providing

guidance and encouragement to help them overcome challenges in programming languages and software development.

Feedback and Support: Offer feedback and support to students as they navigate their studies, providing guidance and encouragement to help them overcome challenges in programming languages and software development.

Library Resources: Ensure that students have access to a well-equipped library with books, journals, and other resources related to programming languages and software development.

Online Forums and Communities: Encourage students to participate in online forums and communities such as Stack Overflow, GitHub, and Reddit, where they can ask questions, share knowledge, and collaborate with others in the field.

1.5 project phases

A-Research

During this phase, researchers design the methodology and approach for the study. This includes selecting appropriate research methods, such as qualitative, quantitative, or mixed methods, designing data collection instruments, and determining sampling strategies.

B-Analysis

analysis phase in project management is essential for making informed decisions, evaluating performance, identifying problems, optimizing resources, monitoring progress, and improving future projects. It enables project managers and stakeholders to extract meaningful insights from data and information collected during the project, ultimately contributing to project success.

C -Prototyping

The prototype phase is a stage in the development process.

The prototype phase is a stage in the development process where a simplified, early version of a product or system is created to test its feasibility, functionality, and usability. This prototype serves as a tangible representation of the final product.

D-Implementation

In the project phase of a software development lifecycle, the focus is on implementing the design and turning it into a working software product

E- Testing

In the testing phase the software is tested to identify and fix defects and issues. This includes various types of testing such as : functional testing , integration testing, performance testing , and user acceptance testing , the goal of testing is to ensure that the software meets the specified requirements and functions correctly in different environments .

1.6 Requirements

The specific requirements of a programming project can vary depending on factors such as the nature of the project, its goals, the technologies being used, and the preferences of stakeholders. However, here are some common requirements that are typically considered in programming projects

Functional Requirements:

What specific features or functionalities should the software have?

What inputs should it accept, and what outputs should it produce?

How should it handle different scenarios or edge cases?

Performance Requirements:

What are the performance expectations in terms of speed, responsiveness, and scalability?

Are there any specific performance targets or benchmarks that need to be met?

Usability Requirements:

How should the user interface be designed to ensure ease of use and a positive user experience?

Are there any accessibility requirements to consider?

Security Requirements:

What security measures need to be implemented to protect the software and its data from unauthorized access or attacks?

Are there any regulatory compliance requirements related to data security or privacy?

Compatibility Requirements:

What platforms or environments should the software be compatible with (e.g., operating systems, web browsers, devices)?

Are there any specific integration points with other systems or services that need to be supported?

Maintainability and Extensibility Requirements:

How should the codebase be structured and documented to facilitate future maintenance and updates?

Are there any requirements related to the ability to extend or modify the software in the future?

Testing and Quality Assurance Requirements:

What testing methodologies and tools should be used to ensure the quality and reliability of the software?

Are there any specific quality metrics or standards that need to be adhered to?

Documentation Requirements:

What documentation needs to be created for the software (e.g., user manuals, developer guides, API

documentation)?

How should the documentation be organized and maintained?

Deployment and Installation Requirements:

How should the software be deployed and installed in different environments (e.g., on-premises, cloud)?

Are there any specific deployment or installation procedures that need to be followed?

Legal and Licensing Requirements:

Are there any legal or licensing considerations related to the use of third-party libraries, open-source software, or intellectual property rights?

Chapter 2

2.1 Accessibility

Who can use our Application and Website:

Anyone interested in the field of programming and would like to learn programming fields such as front-end, back-end, cross platform

He/she can be:

Adults

Children

Everyone wants to program

The administrator:

Can improve the application for new updates.
He can add verified captains to the system.

Accessibility of our project

Clarity of information: A roadmap should present its information clearly and concisely, ensuring that users can easily understand the goals, milestones, timelines, and dependencies.

Ease of interpretation: The roadmap should be structured in a way that makes it easy for users to interpret the information presented. This might involve using visual aids such as

color-coding, icons, or different shapes to represent different types of tasks or milestones.

Feedback Mechanisms: Providing mechanisms for users to provide feedback or ask questions about the roadmap can enhance its accessibility. This might involve holding regular meetings or Q&A sessions, or providing contact information for users to reach out with their inquiries.

Adaptability: A roadmap should be adaptable to changes in circumstances or priorities. This might involve building flexibility into the roadmap structure or regularly reviewing and updating the plan as needed.

Overall, the accessibility of a roadmap is crucial for ensuring that it serves its intended purpose of providing guidance and direction to the stakeholders involved in a project.

2.2 Use Cases

Career Planning: Programming roadmaps are often used by individuals to plan their career trajectory within the software development industry. They provide a clear path for acquiring the necessary skills and knowledge to progress from entry-level positions to more senior roles such as software engineer, technical lead, or architect.

Skill Development: Roadmaps serve as guides for learning new programming languages, frameworks, tools, and technologies. Programmers can use them to identify areas where they need to improve their skills and focus their learning efforts accordingly.

Project Planning: Roadmaps can be used by project managers and development teams to plan the development of software projects. They help identify the tasks, milestones, and dependencies involved in the project and allocate resources effectively to meet project deadlines.

Onboarding New Team Members: Roadmaps can be used by organizations to onboard new software developers and engineers. They provide new hires with a structured overview of the technologies and processes used within the organization and help them quickly get up to speed with the team's development practices.

Continuous Learning: Even experienced programmers can benefit from using roadmaps to stay updated with the latest trends and best practices in the industry. They provide a framework for continuous learning and professional development,

enabling programmers to stay competitive in a rapidly evolving field.

Setting Learning Goals: Roadmaps help programmers set realistic and achievable learning goals. By breaking down complex topics into manageable chunks, roadmaps make it easier for programmers to track their progress and measure their success as they work towards mastering new skills and technologies.

Community Engagement: Roadmaps are often shared within programming communities and forums as resources for aspiring developers. They foster community engagement by providing a platform for knowledge sharing, collaboration, and mentorship among programmers with varying levels of experience.

Overall, programming roadmaps are versatile tools that can be used by individuals, teams, and organizations to plan, execute, and manage software development projects effectively while promoting continuous learning and professional growth within the software .

2.3 Importance

The importance of a roadmap in programming cannot be overstated. Here are several key reasons why roadmaps are crucial in the field of programming:

Structured Learning Path: Programming roadmaps provide a structured and organized path for individuals who are learning to code. They outline the essential concepts, skills, and technologies needed to become proficient in programming, helping beginners avoid confusion and overwhelm.

Clear Goals and Objectives: Roadmaps set clear goals and objectives for programmers to achieve. They break down the learning process into manageable steps, making it easier for individuals to track their progress and stay motivated as they work towards mastering new skills.

Efficient Skill Development: By following a roadmap, programmers can focus their time and energy on learning the most relevant and in-demand skills. This targeted approach to skill development can help individuals progress more quickly and efficiently in their programming journey.

Stay Updated with Industry Trends: Programming roadmaps often include the latest technologies, tools, and best practices that are relevant to the current industry landscape. By following a roadmap, programmers can stay abreast of emerging trends

and ensure that their skills remain up-to-date and marketable.

Career Advancement: Roadmaps provide a roadmap for career advancement in the field of programming. They outline the skills and knowledge needed to progress from entry-level positions to more advanced roles, such as software engineer, technical lead, or architect.

Community and Peer Support: Many programming roadmaps are developed and shared by experienced programmers and industry experts. This community-driven approach to learning provides individuals with access to valuable resources, mentorship, and peer support as they navigate their programming journey.

Personalized Learning: While roadmaps provide a structured learning path, they also allow for flexibility and customization based on individual interests and goals. Programmers can adapt the road map to suit their specific needs and preferences, making the learning process more personalized and enjoyable.

Overall, programming road maps are invaluable for both beginners and experienced programmers, providing clarity, direction, and motivation. They help individuals achieve their programming goals more effectively by guiding learning and skill expansion.

2.4 How to study

Here's a comparison between the way of learning programming in the past and how it's done nowadays:

Learning Programming in the Past:

Limited Resources: In the past, resources for learning programming were limited. Most people relied on textbooks, manuals, and printed documentation, which could be expensive and difficult to obtain.

Classroom-Based Learning: Learning programming often involved attending traditional classroom-based courses or workshops. These courses were typically offered by educational institutions or training centers and followed a structured curriculum.

Less Interactive: Learning programming in the past was less interactive compared to today. Students had limited opportunities for hands-on practice and experimentation, as access to computers and software was not as widespread.

Emphasis on Theory: Programming education in the past often focused more on theoretical concepts and principles rather than practical application. Students spent a lot of time learning about algorithms, data structures, and programming paradigms.

Isolated Learning: Learning programming in the past was often an isolated activity. Students worked through textbooks and exercises on their own, with limited interaction with peers or mentors.

Learning Programming Nowadays:

Abundance of Online Resources: Nowadays, there is an abundance of online resources available for learning programming. These include tutorials, articles, videos, forums, and interactive coding platforms, many of which are freely accessible.

Self-Paced Learning: With the rise of online learning platforms and resources, individuals can learn programming at their own pace and on their own schedule. They have the flexibility to choose what, when, and how they learn. **Hands-On Learning:** Modern programming education emphasizes hands-on learning and practical application. Students have access to coding environments, IDEs (Integrated Development Environments), and online coding challenges that allow them to practice coding in real-time.

Project-Based Learning: Project-based learning has become increasingly popular in programming education. Students work on real-world projects and applications, which helps them develop practical skills and problem-solving abilities.

Interactive Learning Platforms: There are many interactive learning platforms and tools available nowadays that provide a more engaging and immersive learning experience. These platforms often include interactive coding exercises, quizzes, and gamified learning experiences.

Community and Collaboration: Programming education nowadays emphasizes collaboration and community. Students can connect with peers, mentors, and industry professionals through online forums, social media, and coding communities, where they can ask questions, share knowledge, and collaborate on projects.

Specialized Learning Paths: Modern programming education offers specialized learning paths and tracks tailored to different interests and career goals. Whether someone is interested in web development, mobile app development, data science, or cybersecurity, there are resources and courses available to meet their specific needs.

2.5 Just idea

In the beginning, we faced difficulty in learning programming, in determining our goals to continue in a specific field, and these problems appear in several forms, the most important of which is the lack of sources for learning a field and the lack of learning sources. All of these obstacles made us think about this project to help others.

The emergence of Roadx is based on the difficulties we face in our learning journey in the fields of programming

Chapter 3

3.1 project component

It sounds like you're referring to a project that involves developing both a mobile application and a website and it also contains AI (chat bot) the project would involve several key components:

User Interface (UI) Design: This involves designing the layout, appearance, and user experience of both the mobile app and the website. It includes creating wire frames and prototypes to visualize the design before development begins.

Frontend Development: Frontend focuses on building the user-facing aspects of both the mobile app and the website. This includes writing code in languages such as HTML, CSS, and JavaScript to create interactive and responsive interfaces.

Backend Development: Backend development involves building the server-side logic and database architecture that power both the mobile app and the website. This includes handling user authentication, managing data storage and retrieval, and implementing any business logic

Mobile Application (cross platform "flutter")

write code once and deploy it across multiple platforms, including iOS, Android, web, and desktop, thereby reducing development time and effort required to build separate applications for each platform.

Fast Development: Flutter offers a hot reload feature that enables developers to instantly see the changes they make to the code reflected in the app's UI, speeding up the development process and facilitating rapid iteration and experimentation.

Beautiful UIs: Flutter provides a rich set of customizable widgets and a flexible UI framework that allows developers to create stunning and visually appealing user interfaces that adhere to platform-specific design guidelines, such as Material Design for Android and Cupertino for iOS.

machine learning is essential for building chatbots that can understand natural language, learn from interactions, and provide personalized and contextually relevant.

chatbots are valuable tools in learning projects because they provide personalized learning experiences, instant assistance and support, interactive activities, feedback and assessment, accessibility and inclusivity, learning analytics, peer collaboration, adaptive learning paths, and scalability and cost efficiency. By leveraging chatbot technology, educators and learners can enhance learning outcomes, promote engagement, and create more effective and inclusive learning environments.

Chatbots can provide immediate help and guidance to learners whenever they encounter difficulties or have questions about the learning material. This real-time support enhances learning outcomes by addressing learner needs

3.2 project prototype:

The Vital Role of Prototypes in Product Development , The journey of a product begins with a spark of inspiration, a visionary idea that begs to be brought to life. However, ideas, no matter how brilliant, often remain nebulous until they are materialized. This is where prototypes come into play. By translating abstract concepts into tangible representations, prototypes provide stakeholders with a clear vision of the final product. Whether it's a physical prototype crafted from raw materials or a digital prototype simulated through sophisticated design software, the ability to see and interact with a product prototype bridges the gap between imagination and reality.

Feedback: The Compass of Iterative Improvement , Cost-Effective Experimentation: Minimizing Risk,Maximizing Value.

Innovation inherently involves risk, but it is the prudent management of risk that distinguishes successful ventures from their counterparts. Prototyping offers a cost-effective means of experimenting with different ideas, features, and functionalities while minimizing the associated risks. By creating prototypes early in the development process, teams can identify and address potential issues before substantial resources are invested in full-scale production. This proactive approach not only mitigates the financial risks associated with product development but also maximizes the value derived from each iteration, paving the way for a more refined and resilient final product.

In today's fast-paced and ever-changing market landscape, adaptability is paramount. Products must not only meet the needs of users today but also evolve to address emerging trends and technologies tomorrow. Prototyping facilitates agile adaptation by enabling teams to quickly test and iterate upon new ideas and features in response to evolving market dynamics. Whether it's incorporating user feedback, embracing emerging technologies, or pivoting in response to changing consumer preferences, prototypes provide teams with the flexibility to adapt and thrive in an increasingly competitive environment.

Product development is inherently collaborative, requiring cross-functional teams to work together towards a common goal. However, effective collaboration can be challenging, particularly when stakeholders have divergent perspectives and priorities. Prototypes serve as a common language that bridges the gap between disciplines, enabling designers, developers, marketers, and stakeholders to collaborate effectively. By providing a tangible representation of the product, prototypes facilitate communication, foster creativity, and foster consensus, thereby ensuring that everyone is aligned towards a shared vision of success.

Assumptions are the building blocks of innovation, but they are only as valuable as the insights they yield. Prototyping provides a means to validate assumptions and hypotheses through real-world testing and experimentation. By subjecting prototypes to user feedback, usability testing, and A/B testing, teams can gather actionable insights into user behavior, preferences, and pain points. This empirical validation not only validates the viability of the product but also informs subsequent iterations, ensuring that each iteration is grounded in data-driven decision-making.

Iterative Innovation: From Good to Great

products are not born overnight; they are forged through a process of iterative innovation. Prototyping catalyzes this process by fostering a culture of experimentation, iteration, and continuous improvement. By embracing an iterative approach to product development, teams can incrementally refine and enhance the product based on user feedback and testing results. This iterative cycle of innovation enables teams to push the boundaries of what's possible, continually raising the bar for excellence and differentiation in the market.

In conclusion, prototypes are more than just preliminary versions of products; they are indispensable tools that drive the success of product development initiatives. From visualization and feedback to cost-effective experimentation and agile adaptation, prototypes play a vital role in every stage of the product development lifecycle. By empowering collaboration, validating assumptions, and fostering iterative innovation, prototypes pave the way for triumph in the market, unlocking the full potential of visionary ideas and propelling products from concept to reality.

Prototypes come in various forms, each serving distinct purposes in the product development process. Broadly speaking, there are two main types of prototypes: low-fidelity prototypes and high-fidelity prototypes. Let's explore the importance of each type:

Low-Fidelity Prototypes:

Low-fidelity prototypes, often referred to as lo-fi prototypes, are foundational tools in the product development toolkit. They are simplistic representations of a product or system, designed to convey basic concepts and functionalities without focusing on detailed design elements or aesthetics.

Importance:

Early Exploration and Conceptualization: Low-fidelity prototypes are rough, basic representations of a product idea. They are typically created using simple materials such as paper, cardboard, or digital wireframing tools. These prototypes are invaluable in the early stages of product development for exploring multiple concepts and refining ideas.

Rapid Iteration: Because low-fidelity prototypes are quick and inexpensive to create, they facilitate rapid iteration. Teams can easily generate multiple versions of a prototype, test different design alternatives, and gather feedback from stakeholders without investing significant time or resources.

User Feedback: Low-fidelity prototypes are particularly effective for eliciting feedback from users in the early stages of development. Since these prototypes focus on core functionality and user flows rather than aesthetics, users can provide feedback on the fundamental aspects of the product without being distracted by visual details.

High-Fidelity Prototypes: High-fidelity prototypes represent a more polished and detailed iteration of a product or system compared to their low-fidelity counterparts. They aim to simulate the look, feel, and functionality of the final product as closely as possible, incorporating visual design elements, interactive features, and realistic content.

Importance:

Realistic Representation: High-fidelity prototypes are more polished and detailed representations of the final product. They often include interactive elements, visual design, and realistic content. These prototypes provide stakeholders with a more accurate depiction of the user experience and enable them to envision the final product more clearly.

User Testing: High-fidelity prototypes are ideal for conducting user testing and usability studies. By simulating the look and feel of the final product, these prototypes allow users to interact with the interface in a realistic environment. User feedback gathered from high-fidelity prototypes can help identify usability issues, uncover pain points, and refine the user experience.

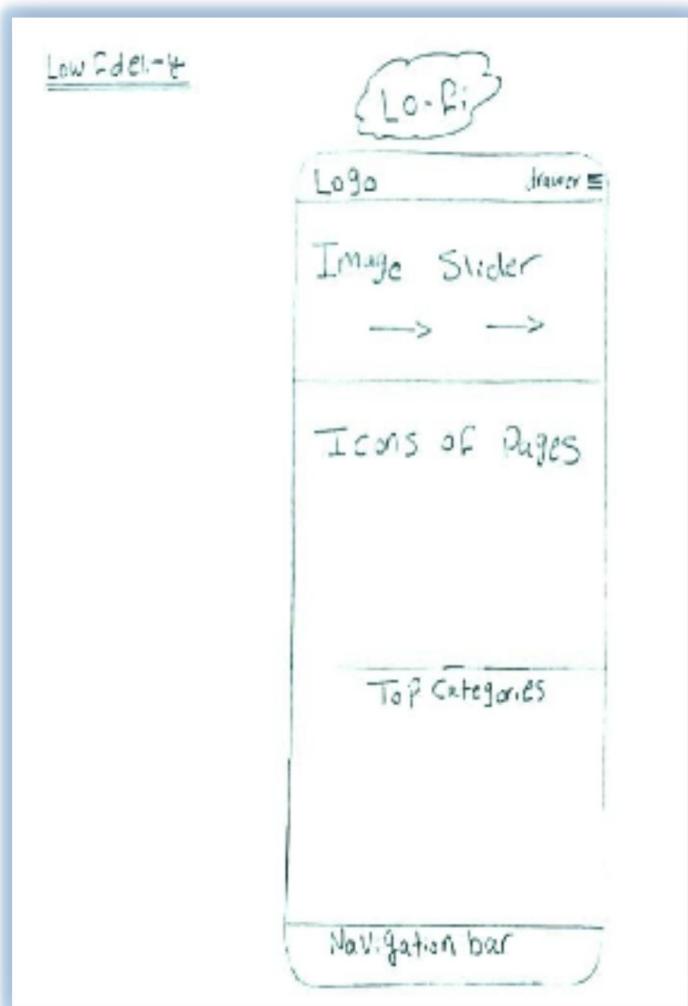
Stakeholder Buy-In: High-fidelity prototypes are powerful tools for gaining stakeholder buy-in and securing funding or approval for the project. The visual polish and interactivity of these prototypes make them more compelling and persuasive than their low-fidelity counterparts. Stakeholders can see firsthand how the product will look and function, which increases their confidence and support for the project.

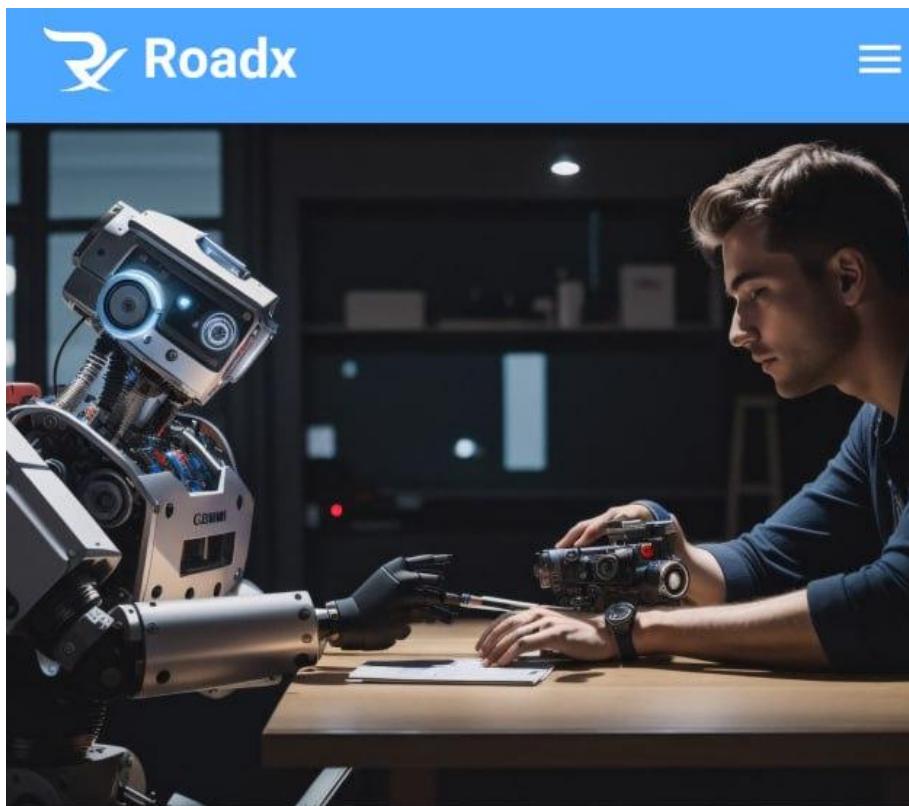
Market Validation: High-fidelity prototypes can also be used for market validation purposes. By presenting a realistic representation of the product to potential customers or investors, teams can gauge interest, gather feedback, and assess market demand before fully committing to development. This early validation can help mitigate the risk of building a product that fails to resonate with its target audience.

In summary, both low-fidelity and high-fidelity prototypes play essential roles in the product development process. Low-fidelity prototypes are valuable for early exploration, rapid iteration, and risk reduction, while high-fidelity prototypes excel at providing a realistic representation of the final product, conducting user testing, gaining stakeholder buy-in, and validating market demand. By leveraging both types of prototypes strategically throughout the development lifecycle, teams can increase their chances of success and deliver products that meet user needs and expectations.

In summary, high fidelity is essential across various domains as it enhances user experiences, supports decision-making, facilitates learning and skill development, improves design and development processes, fosters communication and collaboration, and strengthens marketing and sales efforts.

Mobile Application: Low Fidelity





Categories



Documentation



Extra Knowledge



Quizzes



Programming
Languages



Learning Kids

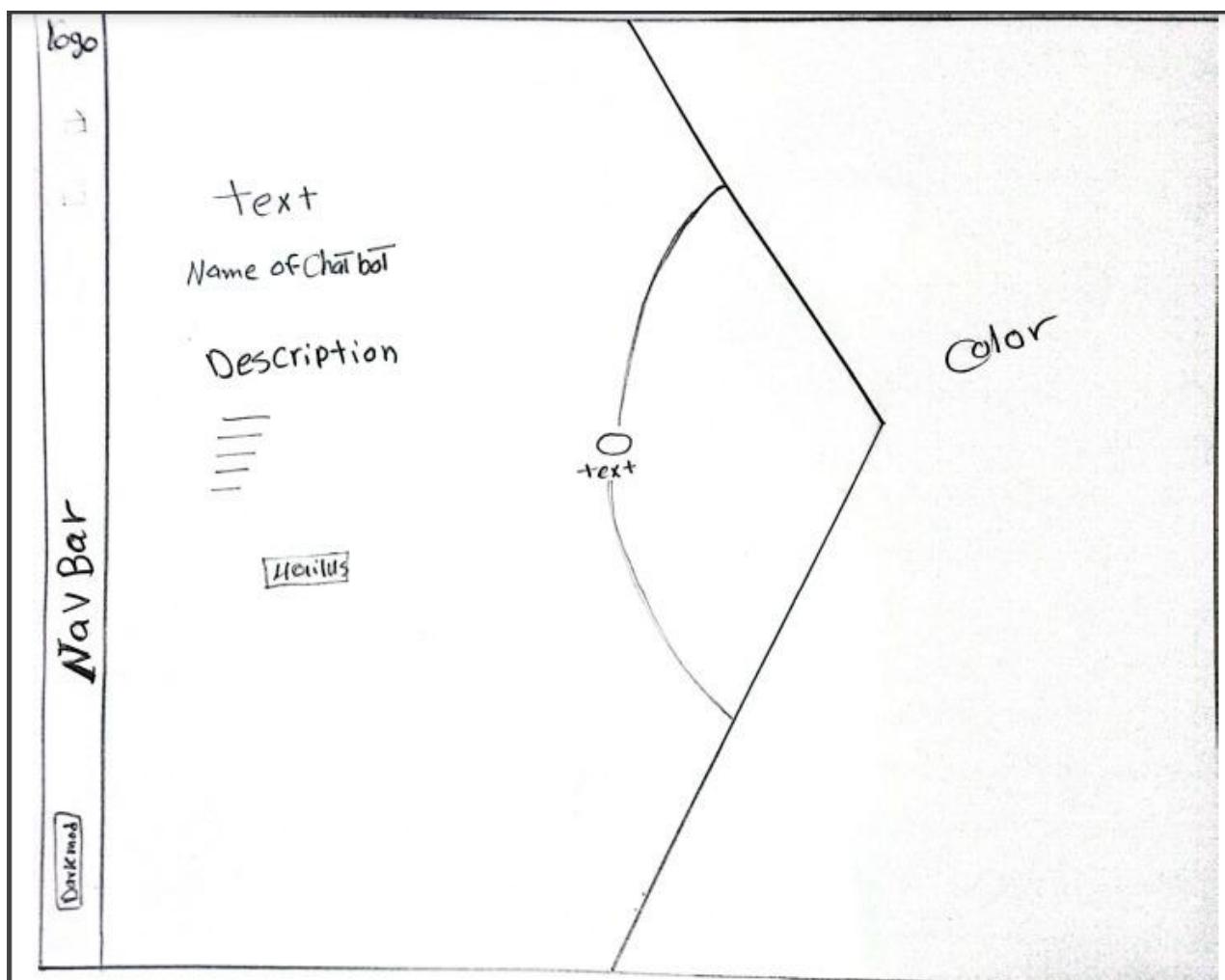
Top Categories

Home

Web Application:

Front_end

Low Fidelity



Logo

Nav Bar

Course

Stacks

and repetition

track name
Courses 1

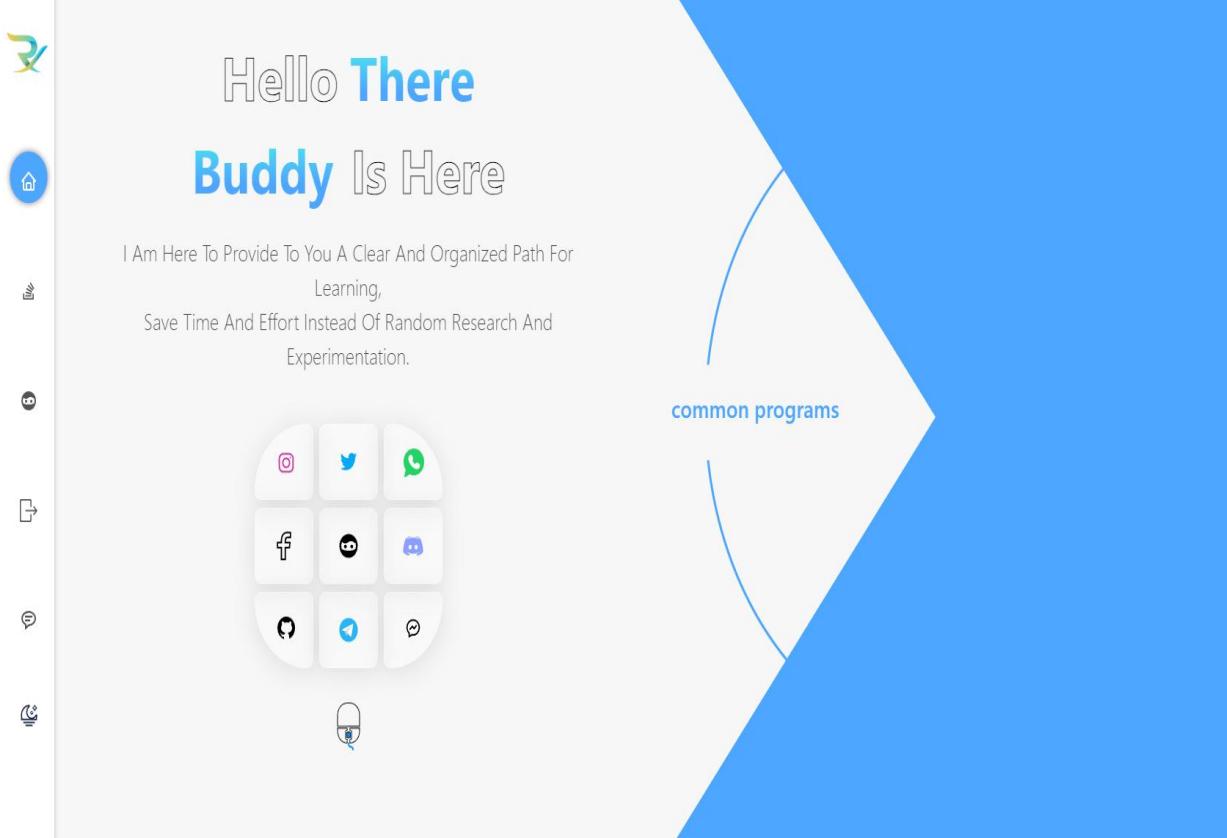
track name
Courses 2

track name
Courses 3

track name
Courses 4

2

High Fidelity

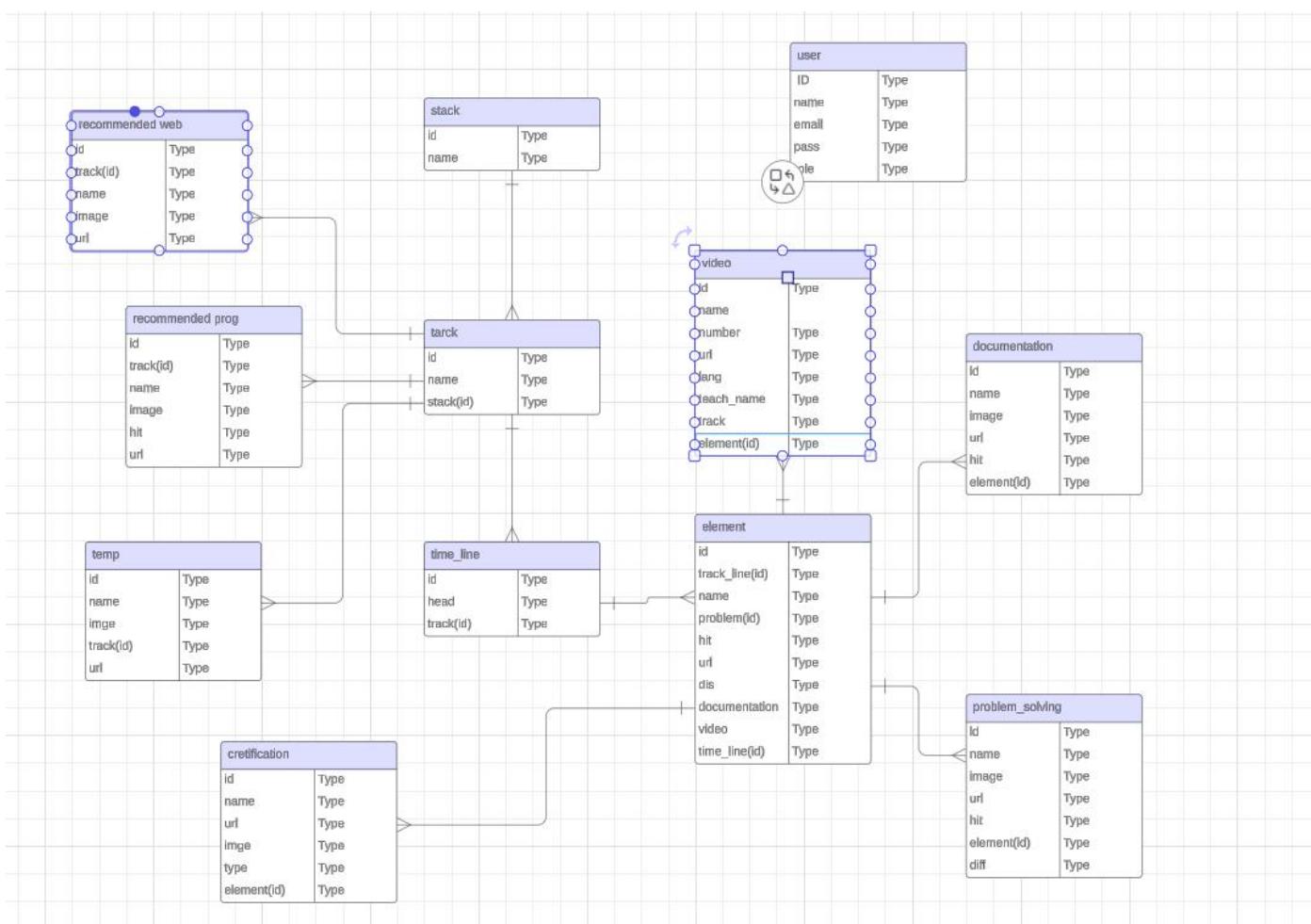


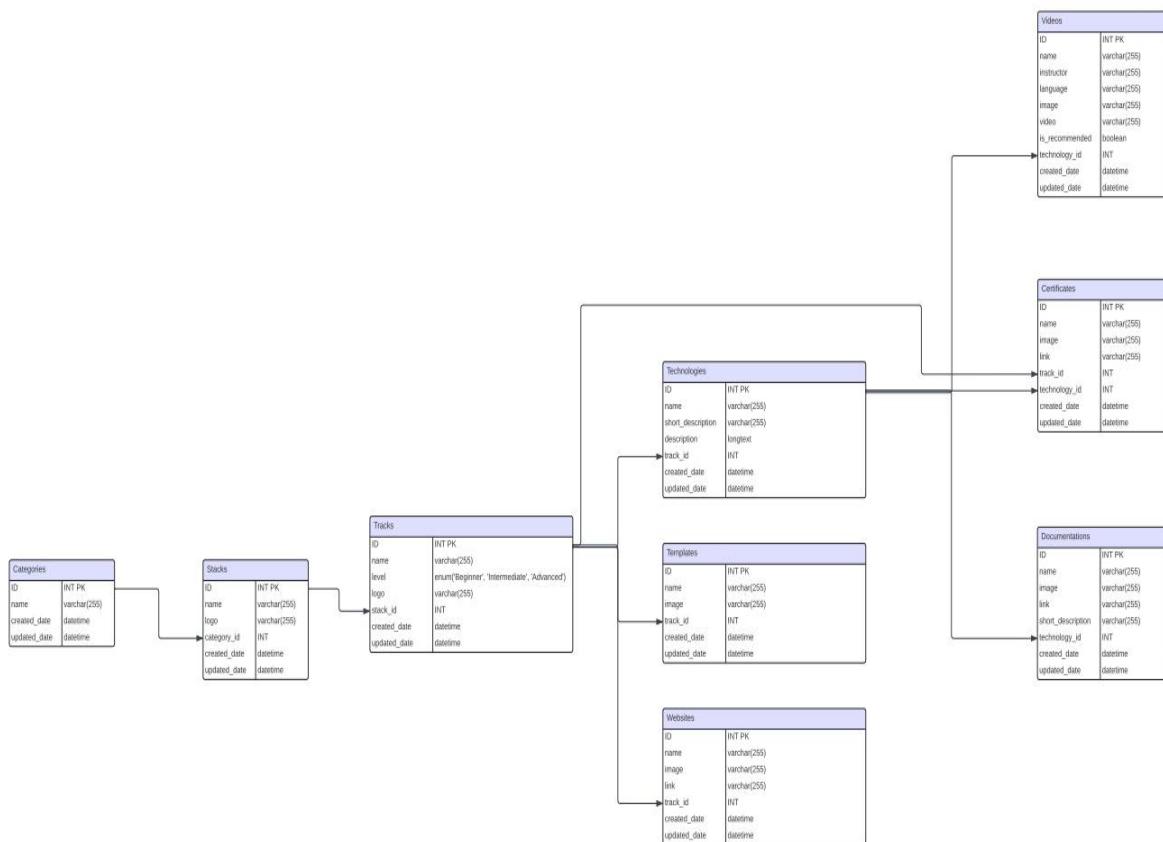
The image shows a user interface for choosing a specific track. On the left is a vertical sidebar with icons for Home, Front-End, Back-End, Network, and UI/UX. The main area is titled "Tracks" and contains four categories: "Web Development", "Application development", "Network", and "Artificial Intelligence and Design". Each category has associated icons and sub-labels.

Web Development		Application development	
Front-End	Back-End	WordPress	Android

Network		Artificial Intelligence and Design	
Network	Cyber security	AI	UI/UX

Back_end & Database





Design logo

First logo



Last logo



Importance of design:

Design plays a crucial role in project success across various fields, including technology, architecture, engineering, and art. Here are some of the key reasons why design is important in projects:

1. Problem Solving: Effective design helps in identifying and solving problems early in the project lifecycle. It ensures that solutions are not only technically feasible but also meet user needs and preferences.
2. User Experience: Design directly impacts how users interact with a product or service. Good design enhances user satisfaction by making products easy to use, accessible, and pleasant to interact with.
3. Functionality and Usability: A well-thought-out design improves the functionality and usability of a product, making it more intuitive and effective for its intended purpose.
4. Cost Efficiency: Thoughtful design can reduce costs by optimizing resources and materials used in a project. It can also minimize costly changes and revisions by addressing potential issues early in the project development.

5. Innovation Design: often drives innovation by challenging the status quo and encouraging creative solutions to conventional problems. This can lead to new products and services that open up new markets or improve competitive positioning.

In summary, design is integral to the success of projects as it encompasses functionality, aesthetics, user experience, and much more. It's not just about how things look but how they work, how they meet user needs, and how they contribute to business goals.

3.3 Design

Mobile Application:

Logo

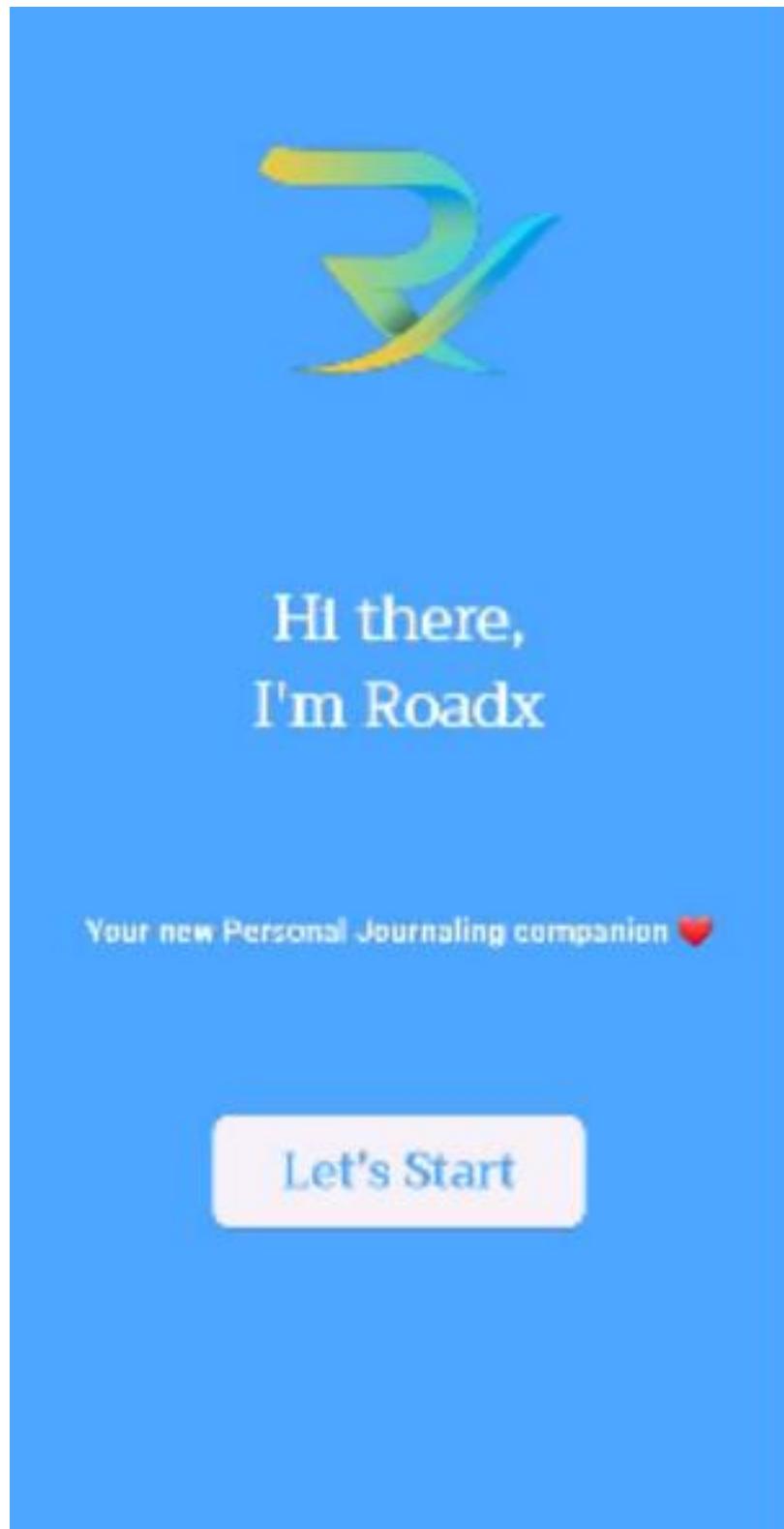


d
Ro^a x



Loading...

Welcome Screen



Home page:



The image shows a man in a dark room, sitting at a desk and interacting with a large, metallic humanoid robot. The robot's hand is reaching towards the man's hand, suggesting a handshake or a connection. The background is dimly lit, with some shelves and equipment visible.

Categories

Documentation

Extra Knowledge

Quizzes

Programming
Languages

Learning Kids

Top Categories

Home

Page 55

Roadx

☰

Categories Documentation Extra Knowledge

Quizzes Programming Languages Learning Kids

Top Categories

Home



Categories

It contains the most programming fields.



Documentation

Documentation plays a crucial role in nurturing users' intellectual growth throughout their learning journey.



Extra Knowledge

Familiarity with modern frameworks, including collaboration between development and operations teams, and continuous improvement essential in the world of programming.



Books

It contains books that facilitate structured learning and provide extensive coverage.



Programming Languages

It contains the most famous programming languages



Learning Kids

It contains educational resources have the potential to enhance children's comprehension of computer systems.



Categories

Android Application

Cross Platform



Cyber Security

Artificial Intelligence

Documentation page

← Documentation	
 mdn web docs	developer.mozilla Frontend
 DevDocs	devdocs.io Frontend
 schools	www.w3schools Frontend
 w3schools GATE PASS 2024	www.w3schools Backend
 MySQL	dev.mysql Backend
 nodejs	nodejs Backend
 codex.wordpress	codex.wordpress Wordpress
 codex.wordpress	



Dart

1 Dart is a modern, object-oriented programming language developed by Google, specifically designed for building user interfaces with its Flutter framework.

Fundamentals of Flutter

2 Flutter is an open-source UI toolkit from Google, designed for developing natively compiled applications for mobile, web, and desktop using a single codebase.

Flutter With Firebase

3 Flutter combined with Firebase offers a powerful duo for app development, streamlining the process from design to deployment.

State Management

4 Flutter with state management refers to the strategies and tools used to manage the state of a Flutter application.



State Management



Ripples Code

GetX - English



Vandad Nahavandipoor

English



The Tech Brothers

English



Wael abo hamza

Provider - Arabic



Wael abo hamza

GetX - Arabic



Code2Start - Mohamed Fathy Taha

Arabic



C++ Programming Language



freecodecamp

English - Single Video



Neso Academy

English



Naresh i Technologies

English



by The Cherno

English



Elzero Web School

Arabic



محمد الدسوقي

Arabic



Ali Shahin

Arabic



Learning Kids



NPStation

Programming Courses For Kids - English



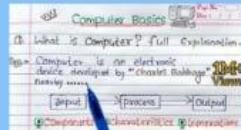
coding for kids

Programming Courses For Kids - Arabic



LearnFree

Computer Basics - English



Learn Coding

Computer Basics - English



Make It Easy Education

Computer Fundamentals - English



Bijoyan Das

Flowcharts - English

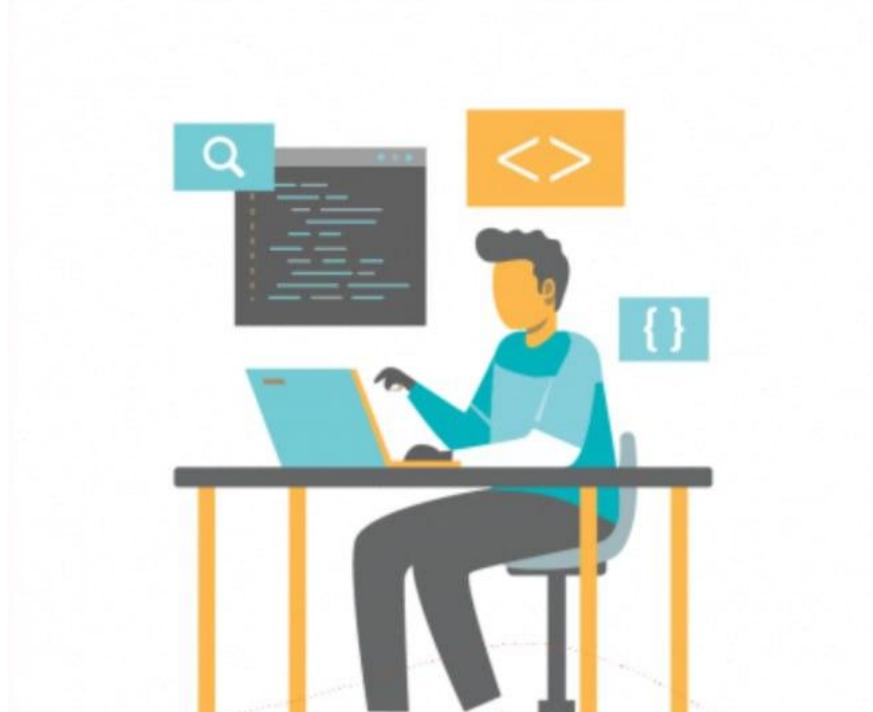
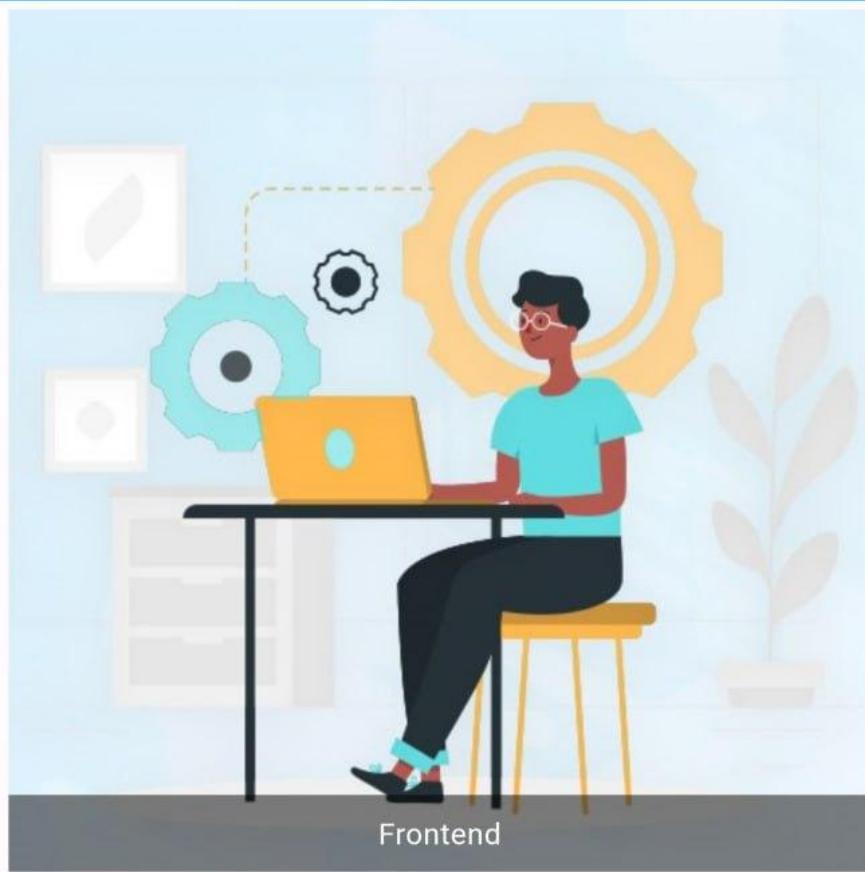


Wondershare Edraw

Flowcharts - English



Quizzes



2. Which property is used to change the background color of an element?

A. color

B. bgcolor

C. background-color

D. backgroundColor



6. Which of the following is NOT a JavaScript framework?

A. React

B. Angular

C. Vue

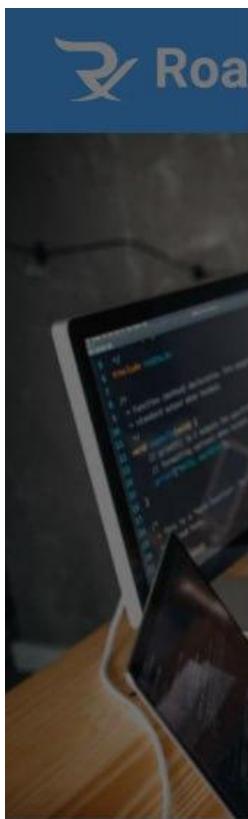
D.

Quiz Completed!

Your score is 6/6. You are at the Advanced level.

OK

Restart



i About

f Contact Us

↗ Visit Our Website

☆ Feedback

⚙ Setting

App Mode



Categorie



Quizzes

Top Cat

1



Settings



FAQ



Privacy Policy



Terms Of Use

What is Roadx?



Roadx is an innovative education app revolutionizing personalized learning with adaptive lessons, interactive content, and collaborative features for students of all levels.

What does Roadx do?



Roadx personalizes learning through adaptive lessons, interactive content, and collaboration features, revolutionizing education for students.

Why should i use Roadx?



Use Roadx for customized learning and engaging lessons, optimizing your educational progress.



Hello

what's your name?

my name is buddy ,i'm a chatbot

what is dart?

Dart is a programming language developed by Google. It is used for building web, mobile, and desktop applications. It's known for its fast performance and easy-to-learn syntax.

what is flutter?

Flutter is a UI toolkit developed by Google that allows you to build natively compiled applications for mobile, web, and desktop from a single codebase. It uses Dart as its programming language. Dart is the language you write your Flutter apps in, and Flutter provides a rich set of pre-designed widgets and tools for building beautiful and performant user interfaces.

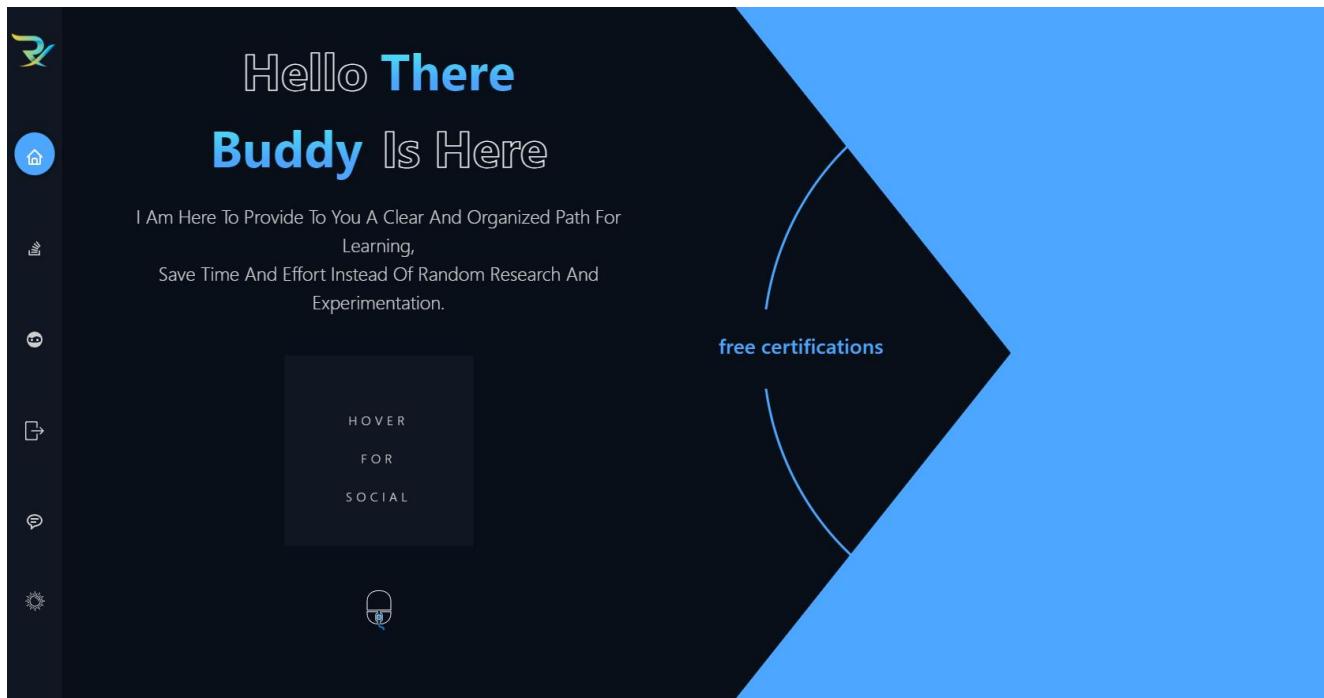
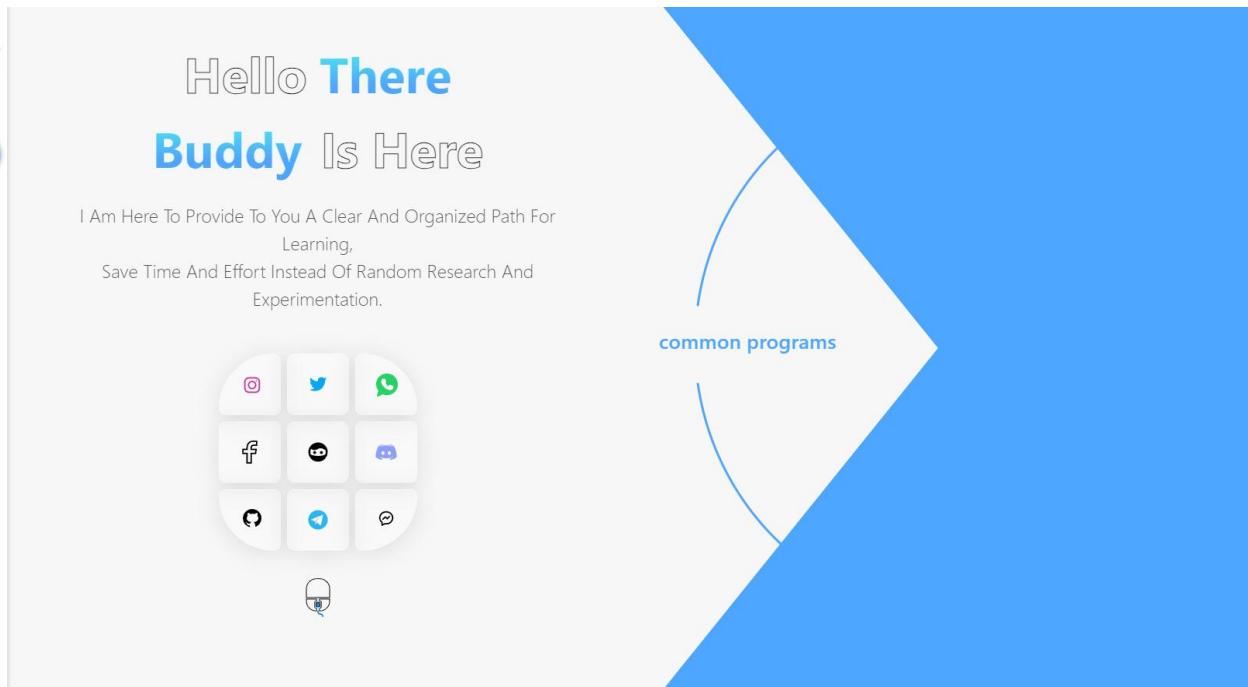
Type a message...



Chatbot

Web Application

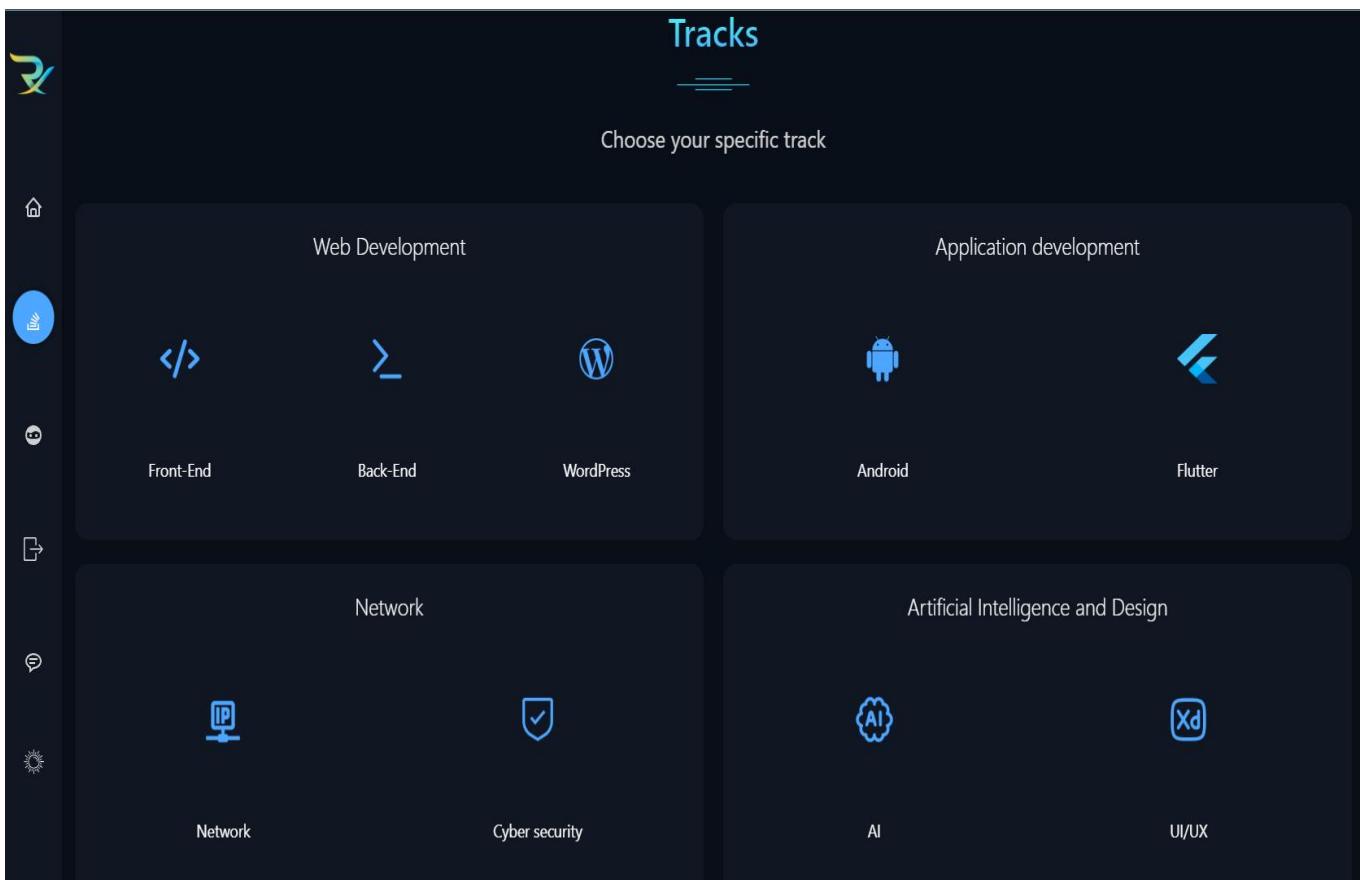
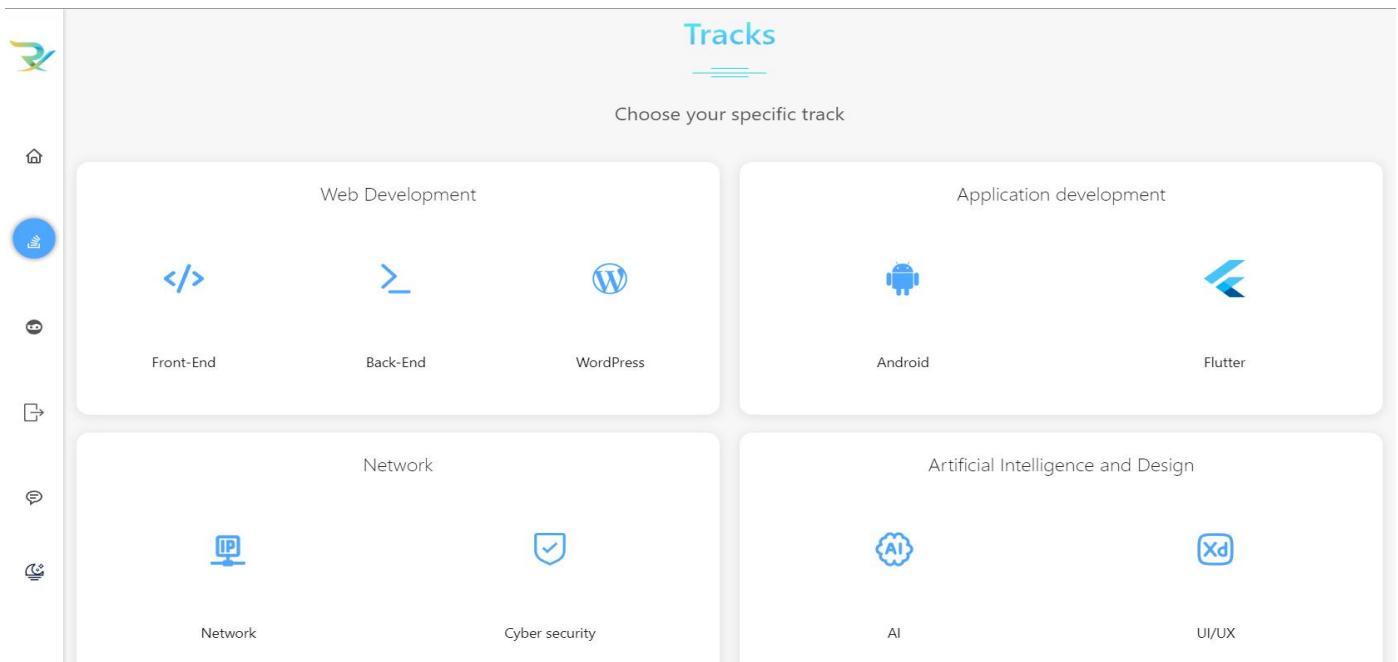
Frontend



In this page we focused that to introduces our services and to provide our social media

Important stacks in our website

Web development , Network, Android development , Artificial intelligences,etc .



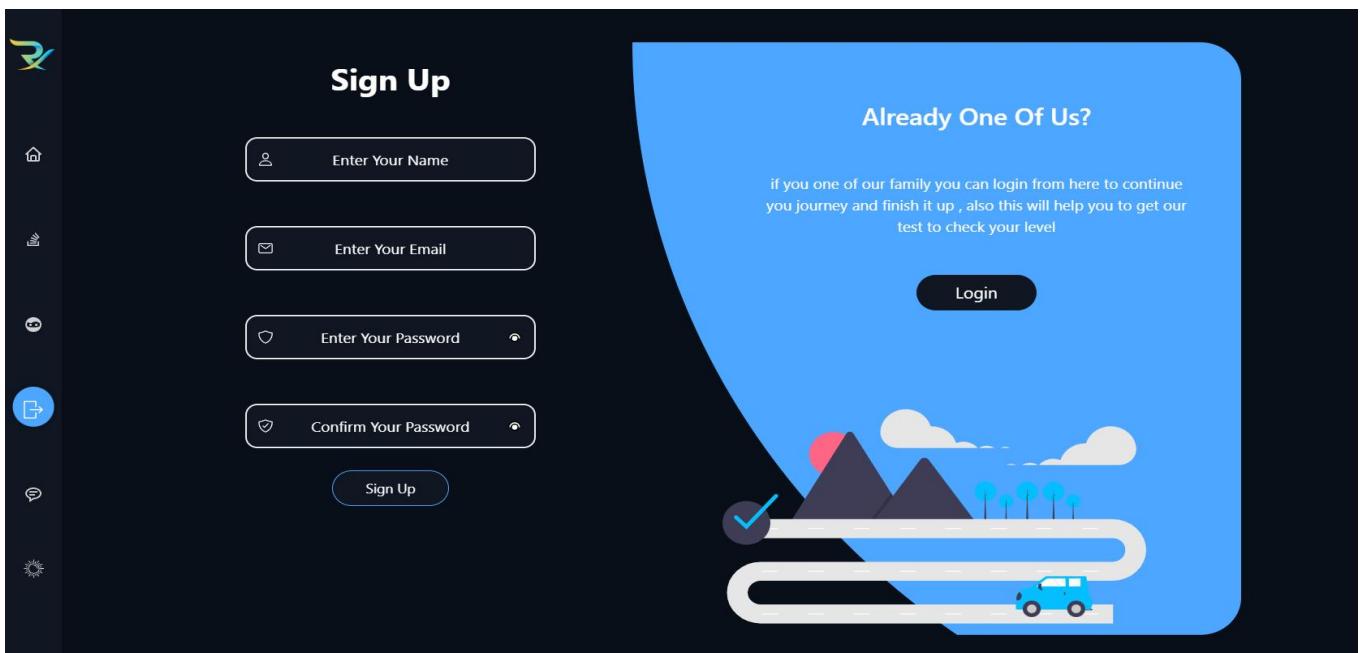
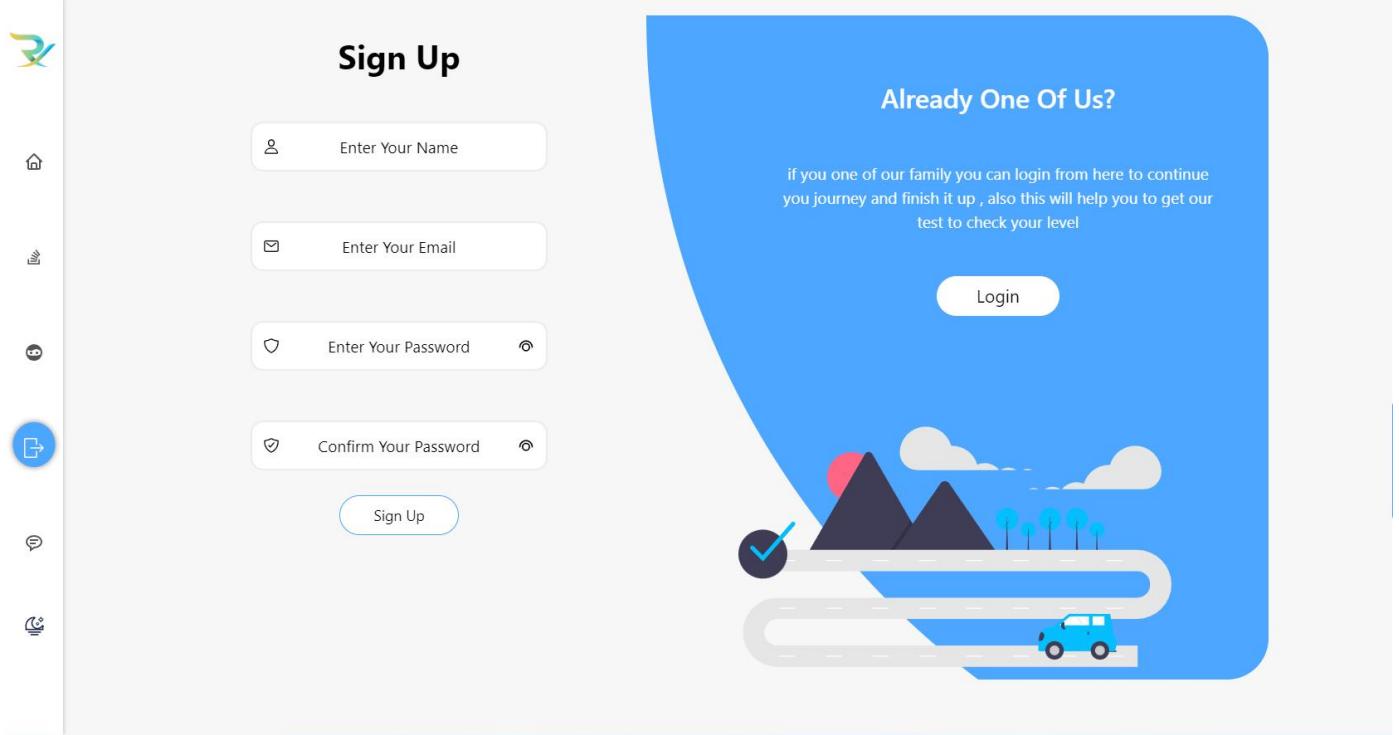
The screenshot shows a dark-themed chat application window. At the top right, there's a white circular icon with a black robot head inside, followed by the text "Hi, I am Buddy" and "Welcome! I am here to help you.". On the far left, there's a vertical sidebar with several icons: a blue "R" at the top, followed by a house, a gear, a document, a person, a blue circle with a white "e", a left arrow, a right arrow, and a sun-like icon. The main area contains a conversation between a user and the bot:

- User: hi
- Buddy: Good to see you again
- User: i have a question
- Buddy: go ahead
- User: what is dart
- Buddy: Dart is a programming language developed by Google. It is used for building web, mobile, and desktop applications. It's known for its fast performance and easy-to-learn syntax.
- User: what is sass
- Buddy: Sass, which stands for 'Syntactically Awesome Style Sheets,' is a powerful preprocessor scripting language that extends CSS. It introduces features such as variables, nesting, mixins, functions, and more, which enhance the capabilities of CSS. Sass provides a more efficient and maintainable way to write CSS code, making it easier to manage stylesheets for complex web projects.
- User: what is android

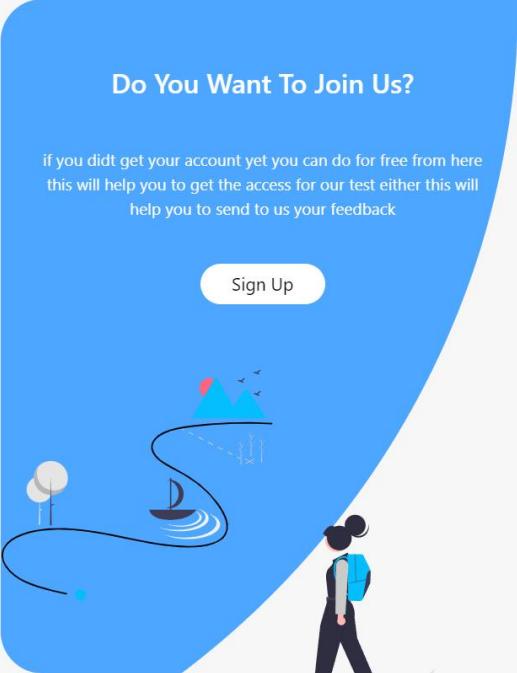
At the bottom, there's a text input field with the placeholder "Enter your question here..." and a blue send button with a white right-pointing arrow.

chatbots play a crucial role in enhancing customer experiences, streamlining business operations, and driving efficiency and cost savings.

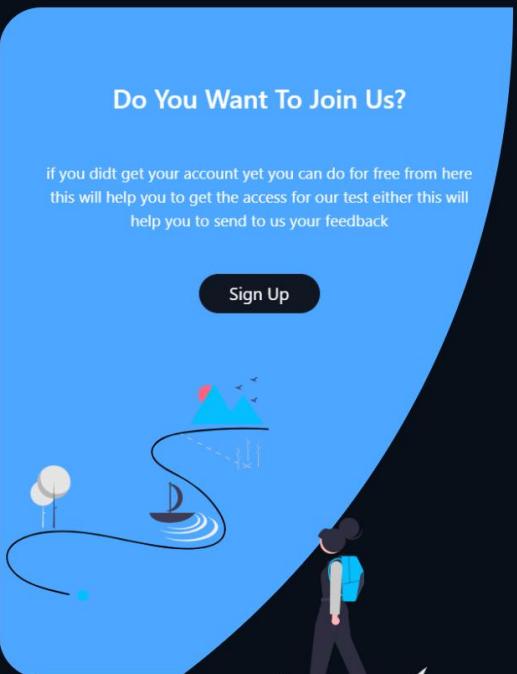
“Buddy” chatbot of Clear and Simple Languages should communicate in a clear and straightforward manner, or complex terminology. Using language that is easy to understand helps users feel more comfortable and engaged.



we provided register system in our page to make the user can access to our exam system .

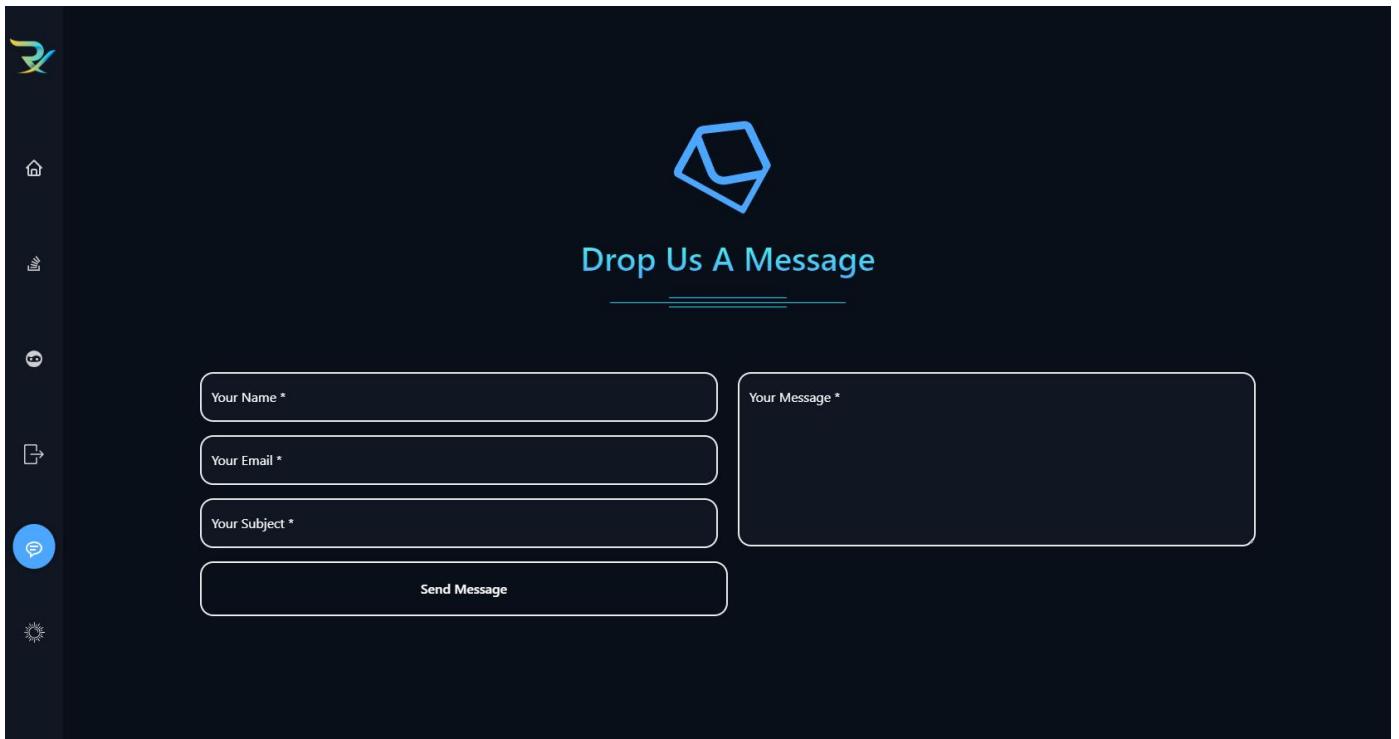


The image shows a light-colored mobile application interface. On the left, there is a vertical navigation bar with icons for home, search, notifications, and account. The main area features a large blue rounded rectangle containing a white sign-up form. The form has a title "Do You Want To Join Us?", a descriptive text about account creation, and a "Sign Up" button. Below the form is a stylized illustration of a sailboat on water with mountains in the background, and a person walking away from the viewer. To the right of the blue box is a "Log In" section with "Enter Your Email" and "Enter Your Password" fields, a "Log In" button, and a "Forgot your password? Reset your password" link.



The image shows a dark-themed mobile application interface. The layout is identical to the light mode version, featuring a vertical navigation bar on the left and a main area with a blue rounded rectangle for sign-up and a "Log In" section on the right. The "Log In" section includes fields for email and password, a "Log In" button, and a password reset link. The overall aesthetic is darker, with the primary colors being black, blue, and white.

we provided in our website sign in mod to our users to easily access to out exam



Feedback is valuable for improving products, services, and processes. Here's a general outline for collecting and utilizing feedback effectively.

User can send feedback to our website to improve our educational way .

Assets

A Front-End Developer is someone who creates websites and web applications. The basic languages for Front-End Development are HTML, CSS, and JavaScript.

HTML

HTML serves as the fundamental foundation of the Web

[learn now](#)

1

Java Script

JavaScript is the Programming Language for the Web.

[learn now](#)

2

CSS

CSS is the language we use to style a Web page.

[learn now](#)

3



Home Stacks Knowledge Buddy Contact

Assets

A Front-End Developer is someone who creates websites and web applications. The basic languages for Front-End Development are HTML, CSS, and JavaScript.

HTML

HTML serves as the fundamental foundation of the Web

[learn now](#)

1

Java Script

JavaScript is the Programming Language for the Web.

[learn now](#)

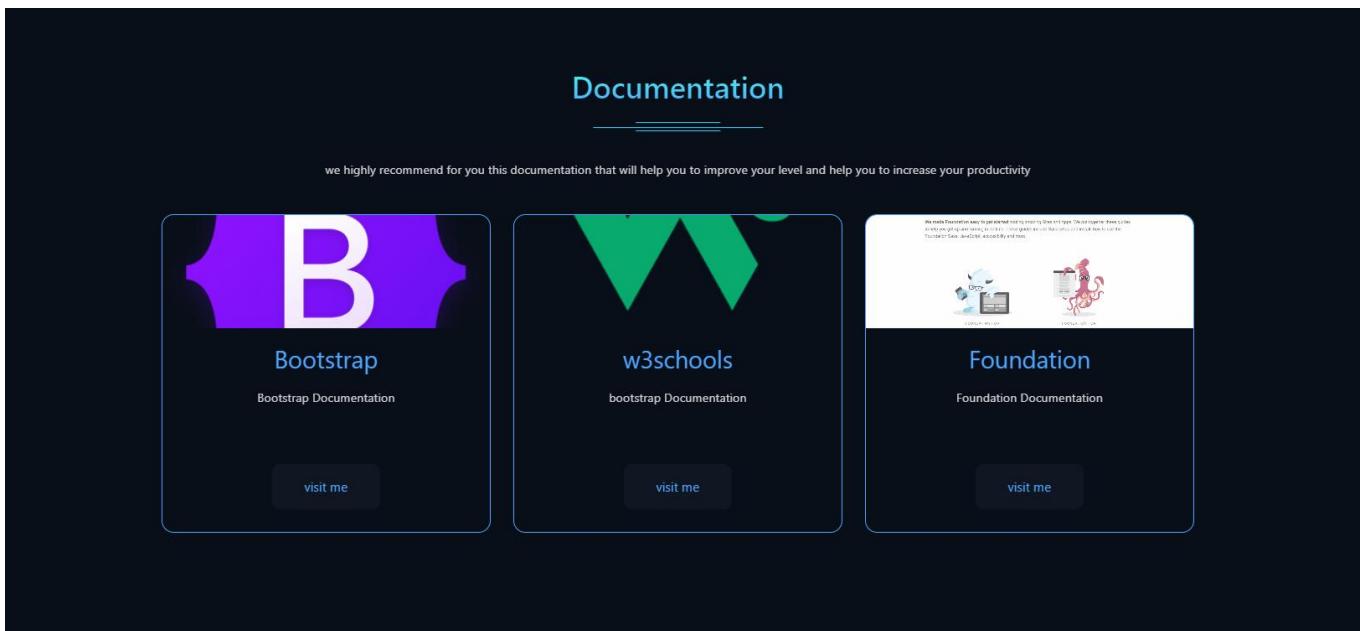
2

CSS

CSS is the language we use to style a Web page.

[learn now](#)

3



A screenshot of a website template library. It features four main preview cards: 'Template 1' (top left), 'Template 2' (top right), 'Get in touch' (bottom left), and 'Discover. Learn. Enjoy.' (bottom right). Each template has a distinct design and color scheme, demonstrating various layout options and features.

we provided templates to improve the user's level through building a real time project

Test Yourself

Level

Beginner

Intermediate

Advanced

Ready to Test Your Knowledge? Take Our Free Beginner Quiz!

Are you curious to see how much you already know about Topics and concepts in Front-End field? This fun and free quiz is designed for beginners, with 10 multiple-choice questions to challenge your understanding.

Here's what you can expect:

- Quick and easy: 10 multiple-choice questions make it a breeze to complete.
- Designed for beginners: Perfect for anyone starting their journey.
- Take it again! No pressure – you can retake the quiz as many times as you like.

Ready to get started? Click the button below to begin!

Ready

Front-End - Intermediate Quiz

Q1: Which property in CSS is used to set the spacing between lines of text?

text-spacing

line-break

line-height

Q2: In JavaScript, which method is used to stop further propagation of an event?

event.stop()

event.stopPropagation()

event.preventDefault()

Page 1 of 5 (Questions 1 to 2 of 10)

Next

Front-End - Intermediate Quiz

Your Degree: 4 correct out of 10

Q9: In HTML, what is the purpose of the <link> tag?

To define a relationship between the document and an external resource

To link JavaScript files

To create a hypertext link



Q10: What is the first element of an array?

Your Degree

4 correct out of 10

Elements with a single value

Done

Elements with multiple values

The first element of a type

Page 5 of 5 (Questions 9 to 10 of 10)

Previous

Submit All Answers

oadX

Home Stacks Knowledge Buddy Contact

What Does Tailwind CSS Mean?

Tailwind CSS scans all of your HTML files, JavaScript components, and any other templates for class names, generates the corresponding styles, and then writes them to a static CSS file.



Tailwind CSS
Full Course

Instructor Name : FreeCodeCamp.Org
Language : English
Duration : 4:12:17
We Recommend Hight This Channel To You

[visit now](#)



Tailwind CSS
Tutorial

Introduction

Instructor Name : Net Ninja
Language : English
Duration : 19 Videos

[visit now](#)

we were trying our best to provide best free sources on youtube and we recommended the pages that we learned from

we display the video and the instructor name ,duration , language , and direct link to page

```
import React, { useState } from "react";
import { HeroSection, FormSection, ContactSection } from "../components";
import { StacksSection } from "../components";
import Footer from "../components/Footer";
import { KnowledgeSection } from "../components";
import {BuddySection} from "../components"
import '../styles/home/style.css'
import { useDarkMode } from "../components/DarkModeContext";

const Home = () => {
  const [isClicked, setIsClicked] = useState(false)
  const { isDarkMode } = useDarkMode();
  return (
    <div className={`body ${isDarkMode ? 'dark-mode' : ''}`}>
      <div data-bs-spy="scroll" data-bs-target="#sideBar">

        <HeroSection/>
        <StacksSection/>
        <BuddySection/>
        {!localStorage.getItem('loggedin') && (
          <FormSection/>
        )}
        <ContactSection/>
        <div className="homeFooter">
          <Footer/>
        </div>
      </div>
    );
};

export default Home;
```

```
import React, { useEffect } from "react";
import Swiper from "swiper";
import "swiper/swiper-bundle.min.css";
import "../styles/stack/certifications.css"

const Certification = ({ data }) => {
  useEffect(() => {
    var CertificationSwiper = new Swiper(".CertificationSwiper", {
      slidesPerView: 1,
      spaceBetween: 0,
      freeMode: true,
      pagination: {
        el: ".swiper-pagination",
        clickable: true,
      },
      breakpoints: {
        576: {
          slidesPerView: 2,
        },
        900: {
          slidesPerView: 2,
        },
      },
    });
  });

  // Cleanup function to destroy the swiper instance on unmount
  return () => {
    if (CertificationSwiper) CertificationSwiper.destroy();
  };
}, []);

return (
  <div className={`certificationSection container-fluid mx-auto py-5`}>
    <div className="row">
      <div className="col-12">
        <div className={`certificationDescription flexCenter flex-column`}>
          <div className={`CertificationPayment coloredText sectionHead my-4 pt-4`}>
            certifications
          </div>
          <div className={`certificationDetails text-center`}>
            we recommend to you this certifications to get a high level and
            help to get Interesting informations
          </div>
        </div>
      </div>
      <div className="col-12">
        <div className={`certificationBox overflow-hidden mt-5`}>
          <div className="swiper-container CertificationSwiper">
            <div className="swiper-wrapper">
              {data &&
                data.certifications &&
                data.certifications.map((certification, index) => (
                  <div className="swiper-slide flexCenter" key={index}>
                    <a href={certification.link} className="">
                      <img
                        src={certification.image}
                        alt=""
                      />
                    </a>
                  </div>
                ))
              }
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>
)
```

```

const Front = () => {
  return (
    <div className={`body ${isDarkMode ? 'dark-mode' : ''}`}>
      {/* navBar section*/}
      <Navbar />
      {/* end navBar section*/}

      {/* Timelines */}
      <div className="bodyContainer container-fluid">
        <div className="trackName centerElement coloredText text-center">
          {data && data.name}
        </div>
        {data && data.tracks && data.tracks.map((track, index) => (
          <TimelineComponent
            key={index}
            trackName={track.name}
            trackDescription={track.description}
            trackLevel={track.level}
            technologies={track.technologies}
          />
        )));
      </div>

      {/* documentation section */}
      {data && data.documentation && data.documentation.length > 0 && (
        <Documents documentsData={data.documentation} />
      )}
      {/* end documentation section */}

      {/* certifications */}
      <Certification data={data}>
      {/* end certifications */}

      {/* Templates section */}
      {(data && data.templates && data.templates.length > 0) && (
        <Templates templatesData={data.templates} />
      )}
      {/* end Templates section */}

      {/* program section */}
      {(data && data.programs && data.programs.length > 0) && (
        <RecommendedPrograms programs={data.programs} />
      )}
      {/* end program section */}

      {/* websites Section */}
      {(data && data.websites && data.websites.length > 0) && (
        <RecommendedWebsite websites={data.websites} />
      )}
      {/* end websites Section */}

      {/* quize Section */}
      <QuizBtn data={data.name} />
      {/* end quize Section */}
    </div>
  );
}

```

```

const Templates = ({ templatesData }) => {
  return [
    <div className="container-fluid py-5">
      <div className="row">
        <div className="col-12">
          <div className="Templates">
            <div className={`head centerElement fs-1 fs-md-4 mb-md-3 mb-lg-2 coloredText fs-1 sectionHead`}>
              | Templates
            </div>
            <div className="TemplateContent container">
              <div className="templateHeader rounded-4">
                <div className="row p-3 my-5">
                  <ul className="nav nav-pills flexCenter" id="pills-tab" role="tablist">
                    {templatesData.map((template, index) => (
                      <li className="nav-item" role="presentation" key={template.id}>
                        <button
                          className={`nav-link px-4 my-2 text-capitalize ${index === 0 ? "active" : ""}`}
                          id={`pills-${template.id}-tab`}
                          data-bs-toggle="pill"
                          data-bs-target={`#pills-${template.id}`}
                          type="button"
                          role="tab"
                          aria-controls={`pills-${template.id}`}
                          aria-selected={index === 0 ? "true" : "false"}
                        >
                          {template.name}
                        </button>
                      </li>
                    )))
                  </ul>
                </div>
              </div>
              <div className="row">
                <div className="templateImgs d-flex flex-wrap p-0">
                  <div className="tab-content" id="pills-tabContent">
                    {templatesData.map((template, index) => (
                      <div
                        className={`tab-pane fade ${index === 0 ? "show active" : ""}`}
                        id={`pills-${template.id}`}
                        role="tabpanel"
                        aria-labelledby={`pills-${template.id}-tab`}
                        tabIndex="0"
                        key={template.id}
                      >
                        <div className="row">
                          {template.image.split("|").map((image, idx) => (
                            <div className="col-lg-6 col-sm-12 p-0" key={idx}>
                              <img
                                src={image}
                                className="mix img-fluid h-100"
                                alt="template img"
                              />
                            </div>
                          ))}
                        </div>
                      </div>
                    )));
                  </div>
                </div>
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  ];
}

```

```

import React, { useEffect, useState } from "react";
import { Link } from "react-router-dom";

import '../styles/home/stacks.css'

const apiUrl = process.env.REACT_APP_API_URL;

const StacksSection = ({ stacks }) => {
  const [firstLevelCategories, setFirstLevelCategories] = useState([]);

  // fetch data from api
  useEffect(() => {
    const fetchData = async () => {
      try {
        const response = await fetch(`${apiUrl}/stacks`);
        if (response.ok) {
          const data = await response.json();
          setFirstLevelCategories(data);
          console.log(data);
        } else {
          console.error("Failed to fetch data");
        }
      } catch (error) {
        console.error("Error:", error);
      }
    };
    fetchData();
  }, []);

  return (
    firstLevelCategories && firstLevelCategories.map((firstLevelCategory, index) => (
      <section id="stackSection" class="stackSection pt-5" >
        <div class="stacks" >

          {/* header */}
          <div class="heading pb-5" >
            <p class="sectionHead coloredText">{firstLevelCategory.name}</p>
            <div class="stacksDetails text-center">{firstLevelCategory.description}</div>
          </div>

          {/* body */}
          <div className="container-fluid-md" >
            <div className="row flexCenter px-0 px-md-3" >
              {firstLevelCategory &&
                firstLevelCategory.categories &&
                firstLevelCategory.categories.map((category, index) => (
                  <div className="col-11 col-lg-6 mb-4 ms-sm-0 ms-2" key={index} >
                    <div className="stacksContent" >
                      <div className="tracks rounded-4" >
                        <div className="stackName text-center py-4" >
                          <i className={category.logo}></i> {category.name}
                        </div>
                        <div className="trackContent flexCenter py-5" >
                          {category.stacks &&
                            category.stacks.map((stack, idx) => (
                              <div className="track text-center" key={idx} >
                                <div className="trackIcon" >
                                  <img src={stack.logo} alt="stackLogo" className="mb-4" style={{ width: '50px', height: '50px' }} />
                                </div>
                                <div className="trackName" >
                                  <Link to={`/stacks/${stack.id}`}><span> {stack.name} </span></Link>
                                </div>
                              </div>
                            )));
                          </div>
                        </div>
                      </div>
                    </div>
                  </div>
                )));
              </div>
            </div>
          </div>
        </div>
      </section>
    ))
  );
}

export default StacksSection;

```

```

return (
  <div className={` container-fluid documentationSection`}>
    <div className="row flexCenter">
      <div className="col-12 text-center">
        <h1 className={`coloredText my-5 sectionHead`}>Documentation</h1>
        <p className={`my-5 recommendtionText`}>
          we highly recommend for you this documentation that will help you to
          improve your level and help you to increase your productivity
        </p>
      </div>
    <div className="col-10 col-lg-9 mb-5">
      <div className="swiper-container mySwiper documentationSwiper overflow-hidden">
        <div className="swiper-wrapper">
          {documentsData &&
            documentsData.map((document, index) => (
              <div className="swiper-slide col-4" key={index}>
                <div className={`documentationBox rounded-4 flexCenter flex-column overflow-hidden`}>
                  <div className="img">
                    <img
                      className="w-100 h-100 object-fit-cover"
                      src={document.image}
                      alt={document.name}
                    />
                  </div>
                  <div className={`documentationName my-3 text-center`}>
                    {document.name}
                  </div>
                  <div className={`documentationDetails px-1`}>
                    {document.short_description}
                  </div>
                  <button
                    onClick={() => window.open(document.link, "_self")}
                    className="documentationButton"
                  >
                    visit me
                  </button>
                </div>
              </div>
            )));
        </div>
        <div className="swiper-pagination"></div>
      </div>
    </div>
  </div>
);
};

export default Documents;

```

```

const Front = () => {
  return (
    <div className={`body ${isDarkMode ? 'dark-mode' : ''}`}>

      {/* navBar section */}
      <Navbar />
      {/* end navBar section */}

      {/* Timelines */}
      <div className="bodyContainer container-fluid">
        <div className="trackName centerElement coloredText text-center">
          {data && data.name}
        </div>
        {data && data.tracks && data.tracks.map((track, index) => (
          <TimelineComponent
            key={index}
            trackName={track.name}
            trackDescription={track.description}
            trackLevel={track.level}
            technologies={track.technologies}
          />
        )));
      </div>

      {/* documentation section */}
      {data && data.documentation && data.documentation.length > 0 && (
        <Documents documentsData={data.documentation} />
      )};
      {/* end documentation section */}

      {/* certifications */}
      <Certification data={data}>
      {/* end certifications */}

      {/* Templates section */}
      {data && data.templates && data.templates.length > 0) && (
        <Templates templatesData={data.templates} />
      )};
      {/* end Templates section */}

      {/* program section */}
      {(data && data.programs && data.programs.length > 0) && (
        <RecommendedPrograms programs={data.programs} />
      )};
      {/* end program section */}
    </div>
  );
}

```

Why I Used React in My Project

1-Component-Based Architecture

Modularity: React's component-based architecture allows for the creation of reusable UI components, leading to a modular and maintainable codebase.

Reusability: Components can be reused across different parts of the application, enhancing development efficiency and consistency.

Performance Optimization

2-Virtual DOM: React uses a virtual DOM to minimize direct DOM manipulations, resulting in faster updates and improved performance.

Efficient Updates: The virtual DOM efficiently determines the minimal set of changes needed to update the actual DOM, reducing the computational load and enhancing performance.

Declarative Syntax

Readable Code: React's declarative approach makes the code more predictable and easier to debug.

Ease of Understanding: The declarative nature simplifies the understanding of the app's structure and behavior, which is especially beneficial for onboarding new developers.

Strong Community and Ecosystem

3-Extensive Libraries and Tools: React has a vast ecosystem of libraries, tools, and extensions that enhance functionality and streamline development processes.

Community Support: A large and active community provides ample resources, tutorials, and third-party components, making problem-solving more straightforward.

Flexibility and Integration

4-Interoperability: React can be easily integrated with other libraries and frameworks, offering flexibility to adapt to various project requirements.

Cross-Platform Development: With React Native, the same knowledge can be applied to develop mobile applications, promoting code reuse across web and mobile platforms.

Robust Development Experience

5-Developer Tools: React Developer Tools provide powerful debugging capabilities, making it easier to inspect and troubleshoot components.

JSX Syntax: JSX allows writing HTML-like code within JavaScript, providing a seamless and intuitive way to define UI components.

Future-Proofing

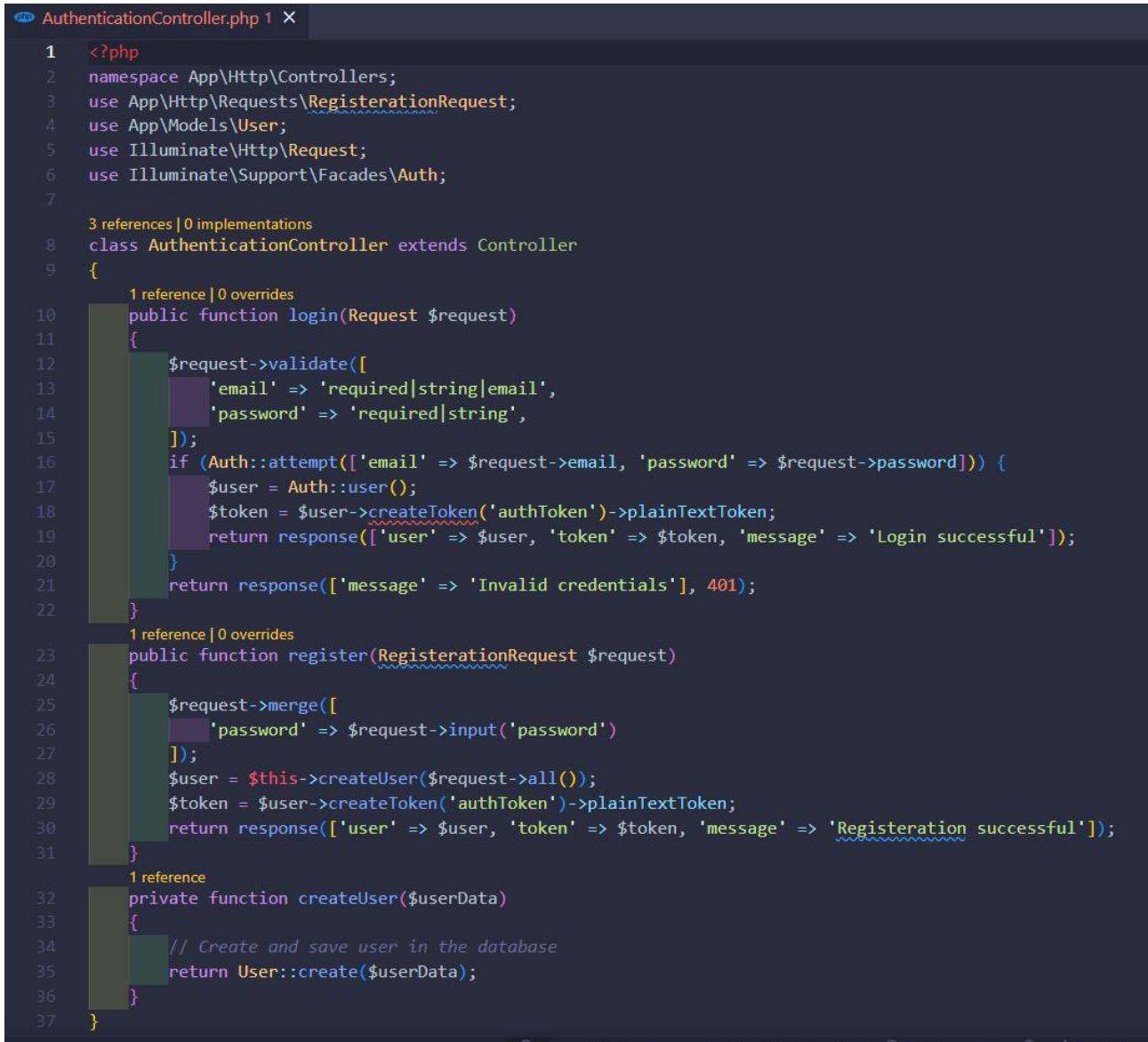
6-Continuous Improvement: React is backed by Facebook and a large community, ensuring continuous improvements, updates, and long-term support.

Scalability: React's architecture is well-suited for both small and large-scale applications, making it a versatile choice for projects that may grow in complexity over time.

State Management

7-Efficient State Handling: React's state management capabilities, especially when combined with libraries like Redux or Context API, provide efficient ways to handle complex state across the application..

Back_end



The screenshot shows a code editor with a dark theme displaying the `AuthenticationController.php` file. The code is written in PHP and defines a controller for user authentication. It includes methods for login and registration, both of which validate input and interact with the database through the `Auth` facade.

```
1 <?php
2 namespace App\Http\Controllers;
3 use App\Http\Requests\RegistrationRequest;
4 use App\Models\User;
5 use Illuminate\Http\Request;
6 use Illuminate\Support\Facades\Auth;
7
8 3 references | 0 implementations
9 class AuthenticationController extends Controller
{
10    1 reference | 0 overrides
11    public function login(Request $request)
12    {
13        $request->validate([
14            'email' => 'required|string|email',
15            'password' => 'required|string',
16        ]);
17        if (Auth::attempt(['email' => $request->email, 'password' => $request->password])) {
18            $user = Auth::user();
19            $token = $user->createToken('authToken')->plainTextToken;
20            return response(['user' => $user, 'token' => $token, 'message' => 'Login successful']);
21        }
22        return response(['message' => 'Invalid credentials'], 401);
23    }
24    1 reference | 0 overrides
25    public function register(RegistrationRequest $request)
26    {
27        $request->merge([
28            'password' => $request->input('password')
29        ]);
30        $user = $this->createUser($request->all());
31        $token = $user->createToken('authToken')->plainTextToken;
32        return response(['user' => $user, 'token' => $token, 'message' => 'Registration successful']);
33    }
34    1 reference
35    private function createUser($userData)
36    {
37        // Create and save user in the database
38        return User::create($userData);
39    }
}
```

The Authentication Controller class in this code is responsible for managing user authentication tasks, such as login and registration.

The login method validates the incoming request to ensure that both the email and password are provided and are of the correct types.

If the authentication fails, it returns a response with an "Invalid credentials" message and a 401 status code.

The register method validates the incoming registration request and calls the createUser method to create a new user using the provided data.

It also generates an authentication token for the newly created user.

The method then returns a response containing the new user, the token, and a success message.

```
1 <?php
2 namespace App\Http\Controllers;
3 use App\Models\Quiz;
4 use Illuminate\Http\Request;
5 class QuizController extends Controller
6 {
7     public function index()
8     {
9         $quiz = Quiz::with('stack')->get();
10        if (!$quiz) {
11            return response()->json(['message' => 'Quiz not found'], 404);
12        }
13        return response()->json($quiz);
14    }
15    public function store(Request $request)
16    {
17        $validated = $request->validate([
18            'question' => 'required|string',
19            'a' => 'required|string',
20            'b' => 'required|string',
21            'c' => 'required|string',
22            'correct_answer' => 'required|in:A,B,C',
23            'level' => 'required|string',
24            'stack_id' => 'required|exists:stacks,id'
25        ]);
26        $quiz = Quiz::create($validated);
27        return response()->json($quiz, 201);
28    }
29    public function show($id)
30    {
31        $quiz = Quiz::with('stack') // Eager Load the stack relationship
32        ->find($id);
33        if (!$quiz) {
34            return response()->json(['message' => 'Quiz not found'], 404);
35        }
36        return response()->json($quiz);
37    }
}
```

The QuizController within Laravel is responsible for handling quiz data:

index: Retrieves all quizzes along with their associated stacks.

store: Validates and generates a new quiz.

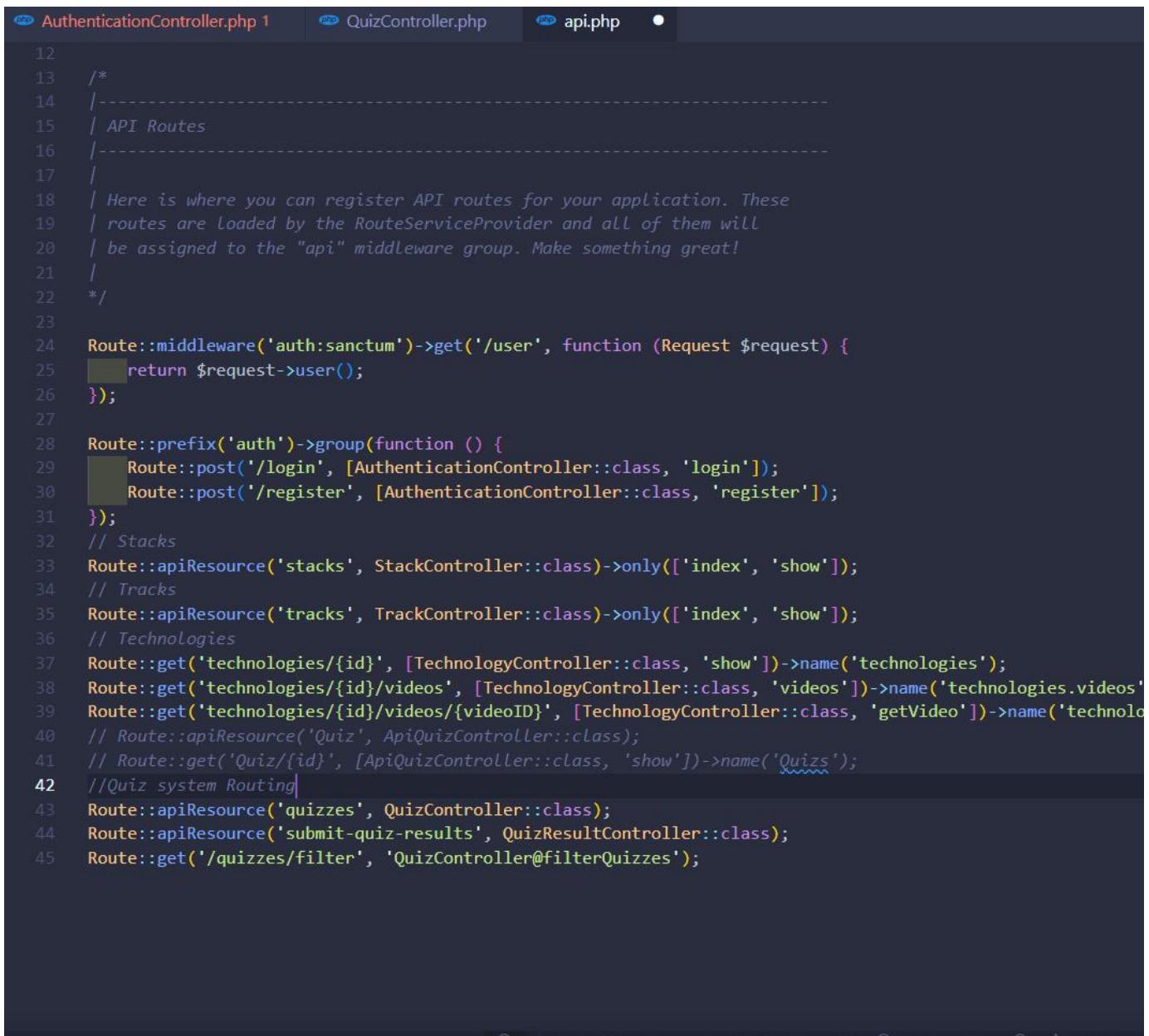
show: Retrieves a particular quiz by its ID along with its stack.

`update`: Modifies a specific quiz with validated data.

`destroy`: Removes a specific quiz by its ID.

`filterQuizzes`: Filters quizzes based on level and stack name.

Each function provides JSON responses.



The screenshot shows a code editor with three tabs at the top: 'AuthenticationController.php 1', 'QuizController.php', and 'api.php'. The 'api.php' tab is active, displaying PHP code for defining API routes. The code includes routes for user authentication, stacks, tracks, technologies, quizzes, and quiz results.

```
12
13  /*
14  |--------------------------------------------------------------------------
15  | API Routes
16  |--------------------------------------------------------------------------
17  |
18  | Here is where you can register API routes for your application. These
19  | routes are loaded by the RouteServiceProvider and all of them will
20  | be assigned to the "api" middleware group. Make something great!
21  |
22  */
23
24 Route::middleware('auth:sanctum')->get('/user', function (Request $request) {
25     return $request->user();
26 });
27
28 Route::prefix('auth')->group(function () {
29     Route::post('/login', [AuthenticationController::class, 'login']);
30     Route::post('/register', [AuthenticationController::class, 'register']);
31 });
32 // Stacks
33 Route::apiResource('stacks', StackController::class)->only(['index', 'show']);
34 // Tracks
35 Route::apiResource('tracks', TrackController::class)->only(['index', 'show']);
36 // Technologies
37 Route::get('technologies/{id}', [TechnologyController::class, 'show'])->name('technologies');
38 Route::get('technologies/{id}/videos', [TechnologyController::class, 'videos'])->name('technologies.videos');
39 Route::get('technologies/{id}/videos/{videoID}', [TechnologyController::class, 'getVideo'])->name('technolo');
40 // Route::apiResource('Quiz', ApiQuizController::class);
41 // Route::get('Quiz/{id}', [ApiQuizController::class, 'show'])->name('Quizzes');
42 //Quiz system Routing
43 Route::apiResource('quizzes', QuizController::class);
44 Route::apiResource('submit-quiz-results', QuizResultController::class);
45 Route::get('/quizzes/filter', 'QuizController@filterQuizzes');
```

The `api.php` file within Laravel is responsible for defining API routes for the application.

Authentication:

- Includes routes for user login and registration (/login, /register).

Stacks, Tracks, Technologies:

- Contains resource routes for stacks and tracks (specifically index and show).
- Includes routes for viewing technologies and their respective videos.

Quizzes:

- Consists of resource routes for quizzes.
 - Also features a specific route for filtering quizzes (/quizzes/filter).
- All routes are well-organized and utilize appropriate middleware and route grouping.

```
use OpenAdmin\Admin>Show;
use \App\Models\Quiz;
use App\Models\Stack;
use App\Models\Track;

0 references | 0 implementations
class QuizController extends AdminController
{
    /**
     * Title for current resource.
     *
     * @var string
     */
    0 references
    protected $title = 'Quiz';

    /**
     * Make a grid builder.
     *
     * @return Grid
     */
    0 references | 0 overrides
    protected function grid()
    {
        $grid = new Grid(new Quiz());
        $grid->column('id', __('Id'));
        $grid->column('Question', __('Question'));
        $grid->column('A', __('A'));
        $grid->column('B', __('B'));
        $grid->column('C', __('C'));
        $grid->column('Correct_Answer', __('Correct Answer'));
        $grid->column('level', __('Level'));
        $grid->column('stack_id', __('Stack id'))->display(function ($stack_id) {
            $stack = Stack::find($stack_id);
            return $stack ? $stack->name : '-';
        });
        $grid->column('created_at', __('Created at'));
        $grid->column('updated_at', __('Updated at'));

        return $grid;
    }
}
```

The QuizController class, which extends AdminController from the OpenAdmin package, is defined in this code. It is responsible for managing the admin interface for quizzes.

grid Method:

- This method creates a new grid for displaying quiz data.
- It defines columns for ID, question, answers (A, B, C), correct answer, level, and stack ID.
- The display of the stack ID is customized by fetching and showing the stack name.
- It also includes columns for creation and update timestamps.
- Overall, this method provides a tabular interface for managing quizzes in the admin panel.

Roadx

Administrator

Dashboard Description... Home

Resources

Github · Documentation · Demo

Environment

PHP version	PHP/8.2.19
Laravel version	10.48.4
CGI	fpm-fcgi
Uname	Linux friendly-pike.94-130-78-96.plesk.page 5.4
Server	Apache
Cache driver	file
Session driver	file
Queue driver	sync
Timezone	UTC
Locale	en
Env	local
URL	https://api.roadx.site

Available extensions

- [open-admin-ext/helpers](#) ✓
- [open-admin-ext/log-viewer](#)
- [open-admin-ext/backup](#)
- [open-admin-ext/config](#)
- [open-admin-ext/api-tester](#)
- [open-admin-ext/media-manager](#)
- [open-admin-ext/scheduling](#)
- [open-admin-ext/reporter](#)
- [open-admin-ext/redis-manager](#)

Dependencies

Dependency	Version
php	^8.1
guzzlehttp/guzzle	^7.2
laravel/framework	^10.10
laravel/sanctum	^3.3
laravel/tinker	^2.8
open-admin-ext/helpers	^1.0
open-admin-org/open-admin	^1.0

Roadx

Administrator

Home / Stacks

Stack List

+ New Filter ▾ Export ▾

<input type="checkbox"/>	ID	Name	Logo	Category ID	Created at	Updated at	Action
<input type="checkbox"/>	1	Front-End	https://k.top4top.io/p_308251d9c1.png	1	2024-06-09T10:25:11.000000Z	2024-06-09T10:25:11.000000Z	
<input type="checkbox"/>	2	Back-End	https://a.top4top.io/p_3082ydlq61.png	1	2024-04-26T09:43:30.000000Z	2024-06-09T10:27:55.000000Z	
<input type="checkbox"/>	3	WordPress	https://g.top4top.io/p_3082lz3wv1.png	1	2024-04-26T15:14:07.000000Z	2024-06-09T10:28:51.000000Z	
<input type="checkbox"/>	4	Android	https://a.top4top.io/p_3082rsuh55.png	2	2024-04-26T15:14:18.000000Z	2024-06-09T10:32:51.000000Z	
<input type="checkbox"/>	6	Flutter	https://g.top4top.io/p_3082xkmbr1.png	2	2024-04-26T19:03:14.000000Z	2024-06-09T10:33:48.000000Z	
<input type="checkbox"/>	7	Network	https://l.top4top.io/p_3082fjy04.png	4	2024-04-26T19:05:20.000000Z	2024-06-09T10:32:58.000000Z	
<input type="checkbox"/>	8	Cyber security	https://l.top4top.io/p_3082pp2x61.png	4	2024-04-26T19:09:55.000000Z	2024-06-09T10:33:22.000000Z	
<input type="checkbox"/>	11	AI	https://k.top4top.io/p_30825ucvr3.png	5	2024-04-26T19:13:38.000000Z	2024-06-09T10:33:05.000000Z	
<input type="checkbox"/>	12	UI/UX	https://j.top4top.io/p_3082951to2.png	5	2024-04-26T19:13:48.000000Z	2024-06-09T10:33:11.000000Z	

Showing 1 to 9 of 9 entries Show 20 entries

Roadx

Administrator

Stack Create

Home / Stacks / Create

Create

Name	<input type="text" value="Input Name"/>
Logo	<input type="text" value="Input Logo"/>
Category id	<input type="text"/>

Continue editing Continue creating View

Roadx

Administrator

Stack List

Home / Stacks

<input type="checkbox"/> + New	<input type="checkbox"/> 9 items selected	<input type="checkbox"/> Filter	Action
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> 1	Front-end	<input type="text" value="https://k.top4top.io/p_308251d9c1.png"/> <input type="text" value="1"/> <input type="text" value="2024-06-09T10:25:11.000000Z"/> <input type="button" value="Edit"/> <input type="button" value="View"/> <input type="button" value="Delete"/>
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> 2	Back-End	<input type="text" value="https://a.top4top.io/p_3082ydlq61.png"/> <input type="text" value="1"/> <input type="text" value="2024-06-09T10:27:55.000000Z"/> <input type="button" value="Edit"/> <input type="button" value="View"/> <input type="button" value="Delete"/>
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> 3	WordPress	<input type="text" value="https://g.top4top.io/p_3082lz3wv1.png"/> <input type="text" value="1"/> <input type="text" value="2024-06-09T10:28:51.000000Z"/> <input type="button" value="Edit"/> <input type="button" value="View"/> <input type="button" value="Delete"/>
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> 4	Android	<input type="text" value="https://a.top4top.io/p_3082rsuh55.png"/> <input type="text" value="2"/> <input type="text" value="2024-06-09T10:32:51.000000Z"/> <input type="button" value="Edit"/> <input type="button" value="View"/> <input type="button" value="Delete"/>
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> 6	Flutter	<input type="text" value="https://g.top4top.io/p_3082xkmbr1.png"/> <input type="text" value="2"/> <input type="text" value="2024-06-09T10:33:48.000000Z"/> <input type="button" value="Edit"/> <input type="button" value="View"/> <input type="button" value="Delete"/>
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> 7	Network	<input type="text" value="https://l.top4top.io/p_3082fpjv04.png"/> <input type="text" value="4"/> <input type="text" value="2024-06-09T10:32:58.000000Z"/> <input type="button" value="Edit"/> <input type="button" value="View"/> <input type="button" value="Delete"/>
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> 8	Cyber security	<input type="text" value="https://i.top4top.io/p_3082pp2x61.png"/> <input type="text" value="4"/> <input type="text" value="2024-06-09T10:33:22.000000Z"/> <input type="button" value="Edit"/> <input type="button" value="View"/> <input type="button" value="Delete"/>
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> 11	AI	<input type="text" value="https://k.top4top.io/p_30825ucvr3.png"/> <input type="text" value="5"/> <input type="text" value="2024-06-09T10:33:05.000000Z"/> <input type="button" value="Edit"/> <input type="button" value="View"/> <input type="button" value="Delete"/>
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> 12	UI/UX	<input type="text" value="https://l.top4top.io/p_3082951to2.png"/> <input type="text" value="5"/> <input type="text" value="2024-06-09T10:33:11.000000Z"/> <input type="button" value="Edit"/> <input type="button" value="View"/> <input type="button" value="Delete"/>

Show 20 entries

Roadx

Stack List

Administrator Home / Stacks

	ID	Name	Logo	Category ID	Created at	Updated at	Action
<input type="checkbox"/>	1	Front-End	https://k.top4top.io/p_308251d9c1.png	1	2024-04-09T10:25:11.000000Z	2024-06-09T10:25:11.000000Z	
<input type="checkbox"/>	2	Back-End	https://a.top4top.io/p_3082ydlq61.png	1	2024-04-26T09:43:30.000000Z	2024-06-09T10:27:55.000000Z	
<input type="checkbox"/>	3	WordPress	https://g.top4top.io/p_3082fjv04.png	1	2024-04-26T10:07.000000Z	2024-06-09T10:28:51.000000Z	
<input type="checkbox"/>	4	Android	https://a.top4top.io/p_3082pp2x61.png	1	2024-04-26T10:18.000000Z	2024-06-09T10:32:51.000000Z	
<input type="checkbox"/>	6	Flutter	https://g.top4top.io/p_3082fjv04.png	1	2024-04-26T10:14.000000Z	2024-06-09T10:33:48.000000Z	
<input type="checkbox"/>	7	Network	https://l.top4top.io/p_3082fjv04.png	4	2024-04-26T19:05.20.000000Z	2024-06-09T10:32:58.000000Z	
<input type="checkbox"/>	8	Cyber security	https://l.top4top.io/p_3082pp2x61.png	4	2024-04-26T19:09:55.000000Z	2024-06-09T10:33:22.000000Z	
<input type="checkbox"/>	11	AI	https://k.top4top.io/p_30825ucv3.png	5	2024-04-26T19:13:38.000000Z	2024-06-09T10:33:05.000000Z	
<input checked="" type="checkbox"/>	12	UI/UX	https://j.top4top.io/p_3082951to2.png	5	2024-04-26T19:13:48.000000Z	2024-06-09T10:33:11.000000Z	

Showing 1 to 9 of 9 entries

Show 20 entries

Roadx

Stack Edit

Administrator Home / Stacks / 12 / Edit

Edit	
Name	<input type="text"/> UI/UX
Logo	<input type="text"/> https://j.top4top.io/p_3082951to2.png
Category ID	<input type="text"/> Artificial Intelligence and Design

Continue editing Continue creating View

Roadx

Stack Show

Administrator

Detail

Id: 12
Name: UI/UX
Logo: https://top4top.io/p_3082951to2.png
Category id: 5
Created at: 2024-04-26 19:13:48
Updated at: 2024-06-09 10:33:11

Delete Edit List

Home / Stacks / 12

Dashboard Admin Roadmap quiz-results quizzes Users First Level Categories Categories Stacks Tracks Technologies Documentations Certificates Templates Videos Websites Programs

Roadx

Administrator List

Administrator

Home / Auth / Users

+ New Filter Export

ID	Username	Name	Roles	Created At	Updated At	Action
1	admin	Administrator	Administrator	2024-04-06 19:48:27	2024-04-06 19:48:27	
2	Abogabra	Abogabra	Administrator	2024-04-26 15:24:15	2024-04-26 15:24:15	
3	Abokhaled	Abokhaled	Administrator	2024-04-27 20:32:13	2024-04-27 20:32:13	
4	Osman	Osman	Administrator	2024-04-28 00:52:13	2024-04-28 00:52:13	
5	amer	amer	Administrator	2024-04-29 20:35:28	2024-04-29 20:35:28	

Show 20 entries < 1 >

https://api.roadx.site/admin/auth/users

<input type="checkbox"/>	ID	Name	Email	Email verified at	Password	Remember token	Created at	Updated at
<input type="checkbox"/>	1	Mohammed@yahoo.com	Mohammed@yahoo.com	2024-04-26T20:18:04.000000Z			2024-04-26T20:18:04.000000Z	2024-04-26T20:18:04.000000Z
<input type="checkbox"/>	2	test	test@test.com	2024-04-26T20:58:30.000000Z			2024-04-26T20:58:30.000000Z	2024-04-26T20:58:30.000000Z
<input type="checkbox"/>	3	Mohammed@yahoo.com	Mohammed@yahoo.com	2024-04-26T21:36:25.000000Z			2024-04-26T21:36:25.000000Z	2024-04-26T21:36:25.000000Z
<input type="checkbox"/>	4	Mohamed1@yahoo.com	Mohamed1@yahoo.com	2024-04-27T19:36:52.000000Z			2024-04-27T19:36:52.000000Z	2024-04-27T19:36:52.000000Z
<input type="checkbox"/>	5	abokhaled	aboasdf@gmail.com	2024-04-27T20:06:37.000000Z			2024-04-27T20:06:37.000000Z	2024-04-27T20:06:37.000000Z
<input type="checkbox"/>	6	Mohammed@123yahoo.com	Mohammed@123yahoo.com	2024-05-03T21:02:56.000000Z			2024-05-03T21:02:56.000000Z	2024-05-03T21:02:56.000000Z
<input type="checkbox"/>	7	Abogabra	abogabra1234560@gmail.com	2024-05-03T21:31:38.000000Z			2024-05-03T21:31:38.000000Z	2024-05-03T21:31:38.000000Z
<input type="checkbox"/>	8	Mody_Back@yahoo.com	Mody_Back@yahoo.com	2024-05-05T03:55:28.000000Z			2024-05-05T03:55:28.000000Z	2024-05-05T03:55:28.000000Z
<input type="checkbox"/>	9	LastTEst@yahoo.co	LastTEst@yahoo.co	2024-05-05T03:59:15.000000Z			2024-05-05T03:59:15.000000Z	2024-05-05T03:59:15.000000Z
<input type="checkbox"/>	10	123	mtoo@gdfcl.com	2024-05-06T03:05:08.000000Z			2024-05-06T03:05:08.000000Z	2024-05-06T03:05:08.000000Z
<input type="checkbox"/>	11	testmody@acwe.com	testmody@acwe.com	2024-05-06T14:25:16.000000Z			2024-05-06T14:25:16.000000Z	2024-05-06T14:25:16.000000Z
<input type="checkbox"/>	13	aboKklLs@gmail.com	aboKklLs@gmail.com	2024-05-27T21:11:25.000000Z			2024-05-27T21:11:25.000000Z	2024-05-27T21:11:25.000000Z

These images showcase the admin interface of a web application including various sections:

Stacks List: Displays a list of technology stacks with details, for example, ID, Name, Logo URL, Category ID, Created at, and Updated at.

Stack Create A form for creating a new technology stack, allowing to addition of new data.

Stack List with Selection: Shows multiple items selected for batch actions like editing or deleting.

Delete Confirmation: A confirmation dialog asking if the user is sure about deleting a selected stack.

Stack Edit: A form for editing an existing stack's details.

Stack Show: Detailed view of a stack, displaying its properties.

Administrator List: Lists administrators with their IDs, usernames, roles, and timestamps for creation and updates.

This admin interface allows for efficient management of various entities within the application.

3.4 TOOLS

Visual Studio Code



Visual Studio Code (VS Code) has become one of the most popular and widely used code editors among developers for several reasons. Here's a comprehensive overview of its importance and key features

Open Source and Free: VS Code is developed as an open-source project by Microsoft and is available for free. This makes it accessible to developers of all backgrounds and budgets.

Cross-Platform Compatibility: It runs on Windows, macOS, and Linux, ensuring a consistent development experience across different operating systems.



Google Colab, short for Google Colaboratory, is a free cloud-based platform provided by Google that offers a Jupyter notebook environment for running Python code. Here's an overview of its important features and tasks.

Jupyter Notebooks: Google Colab provides a Jupyter notebook interface, allowing users to create and share documents containing live code, equations, visualizations, and explanatory text. This makes it an ideal environment for data exploration, prototyping machine learning models, and collaborating on projects.

Free GPU and TPU Support: One of the key advantages of Google Colab is its provision of free access to Graphics Processing Units (GPUs) and Tensor Processing Units (TPUs). This enables users to accelerate their machine learning computations, particularly for training deep learning models, without the need for expensive hardware.

Integration with Google Drive: Colab notebooks are stored in Google Drive, making it easy to save, access, and share your work across devices. This integration also allows for seamless collaboration between team members who have access to the same Google Drive folders.

Pre-installed Libraries: Google Colab comes with many popular Python libraries pre-installed, including TensorFlow, PyTorch, scikit-learn, pandas, and matplotlib, among others. This eliminates the need for users to install these libraries manually and allows them to get started with their projects more quickly.

Access to BigQuery and Google Cloud Services: Colab provides integration with Google Cloud services, allowing users to access datasets stored in BigQuery, Google Cloud Storage, and other Google Cloud Platform services directly from their notebooks. This facilitates data analysis and machine learning tasks that involve large datasets stored in the cloud.

Interactive Visualizations: With support for libraries like matplotlib, seaborn, and Plotly, users can create interactive visualizations directly within their Colab notebooks. This helps in data exploration, model evaluation, and communicating results effectively.

Code Collaboration: Multiple users can collaborate on a Colab notebook in real-time, similar to Google Docs. This enables teams to work together on projects, share insights, and provide feedback without the need for setting up complex development environments.

Educational Use: Google Colab is widely used in educational settings for teaching and learning purposes. It provides a beginner-friendly environment for experimenting with code, learning Python, exploring data science concepts, and implementing machine learning algorithms.

In summary, Google Colab offers a powerful and convenient platform for running Python code, particularly for data analysis, machine learning, and educational purposes. Its combination of Jupyter notebook interface, free GPU/TPU support, integration with Google Drive and Cloud services, and collaborative features makes it a valuable tool for developers, data scientists, educators, and researchers alike.



Figma is a web-based collaborative design tool used primarily for interface design, prototyping, and collaboration among design teams. Here's a definition and overview of its functions.

Definition: Figma is a cloud-based design platform that allows designers to create, prototype, and collaborate on digital designs in real-time. It provides a comprehensive set of tools and features for UI/UX design, enabling teams to work together seamlessly regardless of their physical location.

Functions:

Interface Design: Figma offers a range of tools for creating user interfaces (UIs) and user experience (UX) designs for websites, web applications, mobile apps, and other digital products. Designers can create shapes, icons, buttons, text elements, and more using Figma's intuitive interface.

Prototyping: Designers can quickly create interactive prototypes of their designs within Figma. By linking different screens or components together with transitions and animations, designers can simulate user flows and interactions to visualize how their designs will behave in a real-world context.

Collaboration: One of Figma's key features is its real-time collaboration capabilities. Multiple team members can work on the same design file simultaneously, making edits, adding comments, and providing feedback in real-time. This fosters efficient collaboration and ensures that everyone stays on the same page throughout the design process.

Version History and Comments: Figma automatically maintains a version history of design files, allowing users to review past changes, revert to previous versions, and track the evolution of a design over time. Additionally, designers can leave comments directly on specific elements of a design, facilitating communication and feedback among team members.

Component-Based Design: Figma promotes a component-based approach to design, allowing designers to create reusable design elements (components) that can be easily duplicated and customized across multiple screens and projects. Changes made to a component are automatically reflected across all instances, ensuring consistency and efficiency in the design process.

Developer Handoff: Figma provides tools for seamless handoff of designs to developers. Designers can generate design specifications, export assets, and share design resources with developers directly within the platform, streamlining the transition from design to development.

Plugins and Integrations: Figma supports a wide range of plugins and integrations with third-party tools and services, allowing designers to extend the functionality of the platform to suit their specific needs. This includes integrations with design systems, prototyping tools, project management platforms, and more.

Overall, Figma offers a powerful and collaborative environment for designing digital interfaces, prototyping interactions, and streamlining the design-to-development workflow. Its cloud-based nature, real-time collaboration features, and comprehensive toolset make it a popular choice among design teams of all sizes.

Chapter 4

4.1 Related work

MaharaTech :

ITI online learning platform, serves as a vital training arm of the Ministry of Communication and Information Technology, providing scholarships and free access to IT online learning.



Itrax Academy :

Its programming academy to study any topic, anytime and explore thousands of courses.

4.2. clarification of Roadx

What is the roadx ?

Roadx is innovative education website and application revolutionizing personalized learning with adaptive lessons , interactive content , and collaborative features for students of all levels .

What does roadx do ?

Roadx personalizes learning through adaptive lessons , interactive content and collaborative features for students of all levels , and cater to students looking to enhance their skills or learn new ones, with courses designed for professional development and combines the best features of existing educational technologies to create a comprehensive, engaging, and effective learning environment that adapts to the needs and interests of its users.

Why should I use roadx?

Use roadx for customized learning and engaging lessons , optimizing your educational progress ,helps in deepening understanding and retention of information, could make learning more engaging and interactive, offer the flexibility to learn anytime and anywhere, AI-powered tutoring could provide instant help and corrections, guiding you through difficulties and enhancing your learning curve, This immediate support can be crucial for mastering complex subjects,Using RoadX would essentially leverage modern educational technology to offer a more customized, engaging, and flexible learning environment, which is increasingly important in today's fast-paced, technology-driven world.

4.3.Traditional VS progressive education

Traditional education

known as conventional education, typically emphasizes a structured curriculum and teacher-centered approaches. Key characteristics include:

Teacher-Centered: The teacher is the central figure in the classroom, delivering information to students who are expected to passively receive it.

Standardized Testing: Emphasis is placed on quantifiable results, often measured through standardized tests.

Fixed Curriculum: The curriculum is often set and follows a standardized scope and sequence that all students are expected to follow.

Discipline and Authority: There's a strong emphasis on discipline, and the teacher often acts as an authority figure.

Focus on Basics: Traditional methods stress the basics of education, such as reading, writing, and arithmetic, and often rely on rote learning and memorization.

Progressive Education

Progressive education, which emerged as a response to the perceived limitations of traditional methods, emphasizes student-centered learning and holistic development. Key features include:

Student-Centered: Learners are active participants in their education. Teaching is adaptive to the students' needs and interests.

Critical Thinking and Problem Solving: Education is not just about learning facts but developing skills like critical thinking, creativity, and problem-solving.

Holistic Development: Focuses on the emotional, social, and physical development of children, alongside their academic growth.

Flexible Curriculum: The curriculum can be flexible and integrated, designed to foster interconnections between different subjects.

Collaborative Learning: Encourages cooperation and collaboration among students through group work and projects.

We rely on the path of progressive education by providing various videos for various fields, providing various tests, educational resources for children, various documents, courses with certificates, frameworks, and recommended websites. We paid attention to the shape and colors to suit the user and comfort his eyes, and we relied on simplicity and ease of use.

Chapter 5

5.1.Main Page

```
import 'package:adaptive_theme/adaptive_theme.dart';

import 'package:device_preview/device_preview.dart';

import 'package:flutter/material.dart';

import 'package:roadx/tensorflow.dart';

import
'package:roadx/content_of_stacks/Android/basics_of_j
ava_kotlin.dart';

import
'package:roadx/content_of_stacks/Android/data_stora
ge.dart';

import
'package:roadx/content_of_stacks/Android/firebase_int
egration.dart';

import
'package:roadx/content_of_stacks/Android/fundament
als_of_android_development.dart';

import
'package:roadx/content_of_stacks/Android/networking
_and_api.dart';
```

```
import  
'package:roadx/content_of_stacks/Artificial_Intelligence/fundamentals_of_machine_learning_and_deep_learning.dart';
```

```
import  
'package:roadx/content_of_stacks/Artificial_Intelligence/learn_mathematics.dart';
```

```
import  
'package:roadx/content_of_stacks/Artificial_Intelligence/python_for_machine_learning_and_deep_learning.dart';
```

```
import  
'package:roadx/content_of_stacks/Back_End/csharp.dart';
```

```
import  
'package:roadx/content_of_stacks/Back_End/django.dart';
```

```
import  
'package:roadx/content_of_stacks/Back_End/dotnet.dart';
```

```
import  
'package:roadx/content_of_stacks/Back_End/dotnet_back_end_developer.dart';
```

```
import  
'package:roadx/content_of_stacks/Back_End/express.dart';
```

```
import  
'package:roadx/content_of_stacks/Back_End/flask.dart'  
;  
  
import  
'package:roadx/content_of_stacks/Back_End/flask_ba  
ckend_developer.dart';  
  
import  
'package:roadx/content_of_stacks/Back_End/laravel.d  
art';  
  
import  
'package:roadx/content_of_stacks/Back_End/laravel_b  
ackend_developer.dart';  
  
import  
'package:roadx/content_of_stacks/Back_End/mern_ba  
ckend_developer.dart';  
  
import  
'package:roadx/content_of_stacks/Back_End/mongod  
b.dart';  
  
import  
'package:roadx/content_of_stacks/Back_End/mysql.da  
rt';  
  
import  
'package:roadx/content_of_stacks/Back_End/nodejs.d  
art';  
  
import  
'package:roadx/content_of_stacks/Back_End/php.dart';
```

```
import  
'package:roadx/content_of_stacks/Back_End/python.d  
art';  
  
import  
'package:roadx/content_of_stacks/Back_End/sql.dart';  
  
import  
'package:roadx/content_of_stacks/Cross_Platform/dart.  
dart';  
  
import  
'package:roadx/content_of_stacks/Cross_Platform/flutter_with_firebase.dart';  
  
import  
'package:roadx/content_of_stacks/Cross_Platform/fun  
damentals_of_flutter.dart';  
  
import  
'package:roadx/content_of_stacks/Cross_Platform/rest  
_api.dart';  
  
import  
'package:roadx/content_of_stacks/Cross_Platform/sqlf  
ite.dart';  
  
import  
'package:roadx/content_of_stacks/Cross_Platform/sql.  
dart';  
  
import  
'package:roadx/content_of_stacks/Cross_Platform/stat  
e_management.dart';
```

```
import  
'package:roadx/content_of_stacks/Cyber_Security/ccn  
a_security.dart';  
  
import  
'package:roadx/content_of_stacks/Cyber_Security/fire  
walls.dart';  
  
import  
'package:roadx/content_of_stacks/Cyber_Security/lear  
n_the_basics_of_networking.dart';  
  
import  
'package:roadx/content_of_stacks/Cyber_Security/stu  
dying_linux.dart';  
  
import  
'package:roadx/content_of_stacks/Front_End/css.dart';  
  
import  
'package:roadx/content_of_stacks/Front_End/html.dart  
';  
  
import  
'package:roadx/content_of_stacks/Front_End/javascript.dart';  
  
import  
'package:roadx/content_of_stacks/Network/fundament  
als_of_network.dart';  
  
import  
'package:roadx/content_of_stacks/Network/network_pr  
otocols_and_ccna_fundamentals.dart';
```

```
import  
'package:roadx/content_of_stacks/Network/windows_s  
erver_certification_validates_skills.dart';  
  
import  
'package:roadx/content_of_stacks/UI_UX_Design/basi  
cs_of_ui_design.dart';  
  
import  
'package:roadx/content_of_stacks/UI_UX_Design/cho  
ose_the_tool.dart';  
  
import  
'package:roadx/content_of_stacks/UI_UX_Design/diffe  
rence_between_ui_ux.dart';  
  
import  
'package:roadx/content_of_stacks/UI_UX_Design/und  
erstand_the_platform_guidelines.dart';  
  
import  
'package:roadx/content_of_stacks/Wordpress/flywheel.  
dart';  
  
import  
'package:roadx/content_of_stacks/Wordpress/docker.  
dart';  
  
import  
'package:roadx/Content_Of_Stacks/Wordpress/introdu  
ction_to_wordPress.dart';  
  
import  
'package:roadx/content_of_stacks/Wordpress/mamp.d  
art';
```

```
import  
'package:roadx/content_of_stacks/Wordpress/xampp.  
dart';  
  
import 'package:roadx/api_demo.dart';  
  
import 'package:roadx/drawer/about.dart';  
  
import 'package:roadx/drawer/faq.dart';  
  
import 'package:roadx/drawer/privacypolicy.dart';  
  
import 'package:roadx/drawer/settings.dart';  
  
import 'package:roadx/drawer/termsofuse.dart';  
  
import 'package:roadx/pages/Quizzes.dart';  
  
import 'package:roadx/pages/categories.dart';  
  
import 'package:roadx/pages/chatbot.dart';  
  
import 'package:roadx/pages/documentation.dart';  
  
import 'package:roadx/pages/extra_knowledge.dart';  
  
import 'package:roadx/pages/homepage.dart';  
  
import 'package:roadx/pages/learning_kids.dart';  
  
import  
'package:roadx/pages/programming_language.dart';  
  
import  
'package:roadx/pages_of_extra_knowledge/framework  
s.dart';
```

```
import  
'package:roadx/pages_of_extra_knowledge/important_  
skills.dart';  
  
import  
'package:roadx/pages_of_extra_knowledge/knowledg  
e.dart';  
  
import  
'package:roadx/pages_of_extra_knowledge/templates.  
dart';  
  
import  
'package:roadx/pages_of_frameworks/css_advanced_  
technologies.dart';  
  
import  
'package:roadx/pages_of_frameworks/css_framework.  
dart';  
  
import  
'package:roadx/pages_of_frameworks/javascript_fram  
ework.dart';  
  
import  
'package:roadx/pages_of_important_skills/algorithms_  
and_datastructure.dart';  
  
import  
'package:roadx/pages_of_important_skills/api.dart';  
  
import  
'package:roadx/pages_of_important_skills/git.dart';
```

```
import  
'package:roadx/pages_of_important_skills/github.dart';  
  
import  
'package:roadx/pages_of_knowledge/domain.dart';  
  
import  
'package:roadx/pages_of_knowledge/hosting.dart';  
  
import 'package:roadx/pages_of_knowledge/http.dart';  
  
import  
'package:roadx/pages_of_knowledge/servers.dart';  
  
import  
'package:roadx/pages_of_knowledge/tcp_ip.dart';  
  
import 'package:roadx/pages_of_knowledge/www.dart';  
  
import  
'package:roadx/pages_of_programming_languages/cplusplus_programming_language.dart';  
  
import  
'package:roadx/pages_of_programming_languages/csharp_programming_language.dart';  
  
import  
'package:roadx/pages_of_programming_languages/java_programming_language.dart';  
  
import  
'package:roadx/pages_of_programming_languages/python_programming_language.dart';
```

```
import 'package:roadx/resources/android.dart';

import
'package:roadx/resources/artificialintelligence.dart';

import 'package:roadx/resources/backend.dart';

import 'package:roadx/resources/cross_platform.dart';

import 'package:roadx/resources/cyber_security.dart';

import 'package:roadx/resources/frontend.dart';

import 'package:roadx/resources/network.dart';

import 'package:roadx/resources/ui_ux_design.dart';

import 'package:roadx/resources/wordpress.dart';

import 'package:roadx/widgets/nav.dart';

import 'package:roadx/widgets/splashscreen.dart';

import 'package:roadx/widgets/welcomeScreen.dart';

void main() async {

  runApp(
    DevicePreview(
      enabled: false,
      builder: (context) => const MyApp(),
    )
  );
}
```

```
    ),  
 );  
}  
  
class MyApp extends StatefulWidget {  
  const MyApp({super.key});  
  
  @override  
  State<MyApp> createState() => _MyAppState();  
  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      locale: DevicePreview.locale(context),  
      builder: DevicePreview.appBuilder,  
      home: const SplashScreen(),  
    );  
  }  
}  
  
class _MyAppState extends State<MyApp> {  
  GlobalKey<ScaffoldState> scaffoldState =  
  GlobalKey();  
  
  @override
```

```
Widget build(BuildContext context) {  
    return AdaptiveTheme(  
        light: ThemeData.light(),  
        dark: ThemeData.dark(),  
        initial: AdaptiveThemeMode.light,  
        builder: (theme, darkTheme) => MaterialApp(  
            theme: theme,  
            darkTheme: darkTheme,  
            debugShowCheckedModeBanner: false,  
            routes: {  
                SplashScreen.id: (context) => const  
                    SplashScreen(),  
                WelcomeScreen.id: (context) => const  
                    WelcomeScreen(),  
                Nav.id: (context) => const Nav(),  
                HomePage.id: (context) => const HomePage(),  
                Chatbot.id: (context) => const Chatbot(),  
                Categories.id: (context) => const Categories(),  
                Documentation.id: (context) => const  
                    Documentation(),  
            },  
        ),  
    );  
}
```

ExtraKnowledge.id: (context) =>
ExtraKnowledge(),

Quizzes.id: (context) => const Quizzes(),

ProgrammingLanguages.id: (context) =>
ProgrammingLanguages(),

LearningKids.id: (context) => const
LearningKids(),

Frontend.id: (context) => const Frontend(),

Backend.id: (context) => const Backend(),

Wordpress.id: (context) => const Wordpress(),

AndroidApplication.id: (context) => const
AndroidApplication(),

CrossPlatform.id: (context) => const
CrossPlatform(),

Network.id: (context) => const Network(),

CyberSecurity.id: (context) => const
CyberSecurity(),

ArtificialIntelligence.id: (context) =>
const ArtificialIntelligence(),

UIUXDesign.id: (context) => const
UIUXDesign(),

About.id: (context) => const About(),

Settings.id: (context) => const Settings(),

FAQ.id: (context) => const FAQ(),

PrivacyPolicy.id: (context) => const PrivacyPolicy(),

TermsOfUse.id: (context) => const TermsOfUse(),

HTML.id: (context) => const HTML(),

CSS.id: (context) => const CSS(),

Javascript.id: (context) => const Javascript(),

Dart.id: (context) => const Dart(),

FundamentalsOfFlutter.id: (context) => const FundamentalsOfFlutter(),

FlutterWithFirebase.id: (context) => const FlutterWithFirebase(),

StateManagement.id: (context) => const StateManagement(),

SQLFlutter.id: (context) => const SQLFlutter(),

SQFLite.id: (context) => const SQFLite(),

RestAPI.id: (context) => const RestAPI(),

DifferenceBetweenUIUX.id: (context) => const DifferenceBetweenUIUX(),

```
    ChooseTheTool.id:      (context)      =>      const
ChooseTheTool(),

    UnderstandThePlatformGuidelines.id: (context)
=>

        const UnderstandThePlatformGuidelines(),

    BasicsOfUIDesign.id:      (context)      =>
BasicsOfUIDesign(),

    LearnBasicsOfJavaKotlin.id: (context) =>
        const LearnBasicsOfJavaKotlin(),

    FundamentalsOfAndroidDevelopment.id:
(context) =>
        const FundamentalsOfAndroidDevelopment(),

    DataStorage.id:      (context)      =>      const
DataStorage(),

    NetworkingAndAPIs.id:   (context)      =>      const
NetworkingAndAPIs(),

    FirebaseIntegration.id: (context)      =>      const
FirebaseIntegration(),

    MernBackendDeveloper.id: (context)      =>      const
MernBackendDeveloper(),

    NodeJS.id: (context) => const NodeJS(),

    Express.id: (context) => const Express(),
```

```
MongoDB.id: (context) => const MongoDB(),  
LaravelBackendDeveloper.id: (context) =>  
    const LaravelBackendDeveloper(),  
PHP.id: (context) => const PHP(),  
Laravel.id: (context) => const Laravel(),  
 MySql.id: (context) => const MySql(),  
FlaskBackendDeveloper.id: (context) => const  
FlaskBackendDeveloper(),  
Python.id: (context) => const Python(),  
Flask.id: (context) => const Flask(),  
Django.id: (context) => const Django(),  
DotNetBackendDeveloper.id: (context) =>  
    const DotNetBackendDeveloper(),  
Csharp.id: (context) => const Csharp(),  
DotNet.id: (context) => const DotNet(),  
Sql.id: (context) => const Sql(),  
FundamentalsOfNetwork.id: (context) => const  
FundamentalsOfNetwork(),  
NetworkProtocolsAndCCNAFundamentals.id:  
(context) =>
```

```
    const
NetworkProtocolsAndCCNAFundamentals(),

    WindowsServer.id: (context) => const
WindowsServer(),

    LearnBasicOfNetwork.id: (context) => const
LearnBasicOfNetwork(),

    CCNASecurity.id: (context) => const
CCNASecurity(),

    StudyingLinux.id: (context) => const
StudyingLinux(),

    Firewalls.id: (context) => const Firewalls(),

    Flywheel.id: (context) => const Flywheel(),

    Docker.id: (context) => const Docker(),

    XAMPP.id: (context) => const XAMPP(),

    MAMP.id: (context) => const MAMP(),

    IntroductionToWordPress.id: (context) =>
        const IntroductionToWordPress(),

    APIDemo.id: (context) => const APIDemo(),

    JavaProgrammingLanguage.id: (context) =>
        const JavaProgrammingLanguage(),
```

```
CPlusPlusProgrammingLanguage.id: (context)
=>

    const CPlusPlusProgrammingLanguage(),

CSharpProgrammingLanguage.id: (context) =>

    const CSharpProgrammingLanguage(),

PythonProgrammingLanguage.id: (context) =>

    const PythonProgrammingLanguage(),

Templates.id: (context) => Templates(),

LearnMathematics.id: (context) => const
LearnMathematics(),

PythonForMachineLearningAndDeepLearning.id:
(context) =>

    const
PythonForMachineLearningAndDeepLearning(),

FundamentalsOfMachineLearningAndDeepLearning.id:
(context) =>

    const
FundamentalsOfMachineLearningAndDeepLearning(),

ImportantSkills.id: (context) => const
ImportantSkills(),

Git.id: (context) => const Git(),
```

```
Github.id: (context) => const Github(),  
AlgorithmsAndDataStructure.id: (context) =>  
    const AlgorithmsAndDataStructure(),  
APISkills.id: (context) => const APISkills(),  
WWW.id: (context) => const WWW(),  
TCPIP.id: (context) => const TCPIP(),  
Hosting.id: (context) => const Hosting(),  
HTTP.id: (context) => const HTTP(),  
Domain.id: (context) => const Domain(),  
Servers.id: (context) => const Servers(),  
Knowledge.id: (context) => const Knowledge(),  
Frameworks.id: (context) => const  
Frameworks(),  
CSSFramework.id: (context) => const  
CSSFramework(),  
JavascriptFramework.id: (context) => const  
JavascriptFramework(),  
CSSAdvancedTechnologies.id: (context) =>  
    const CSSAdvancedTechnologies(),  
Tensorflow.id: (context) => const Tensorflow(),
```

```
        },  
        initialRoute: SplashScreen.id,  
    ),  
);  
}  
}  
  
double getResponsiveFontSize(BuildContext context,  
{required double fontSize}) {  
  
    double scaleFactor = getScaleFactor(context);  
  
    double responsiveFontSize = fontSize * scaleFactor;  
  
    double lowerLimit = fontSize * 0.8;  
  
    double upperLimit = fontSize * 1.2;  
  
    return responsiveFontSize.clamp(lowerLimit,  
upperLimit);  
}  
  
double getScaleFactor(BuildContext context) {  
  
    double width = MediaQuery.sizeOf(context).width;
```

```
if (width < 600) {  
    return width / 400;  
}  
else if (width < 900) {  
    return width / 700;  
}  
else {  
    return width / 1000;  
}  
}
```

5.2-Home Page

```
import 'package:carousel_slider/carousel_slider.dart';  
import 'package:flutter/material.dart';  
import 'package:roadx/resources/backend.dart';  
import 'package:roadx/resources/cross_platform.dart';  
import 'package:roadx/resources/frontend.dart';  
import 'package:roadx/resources/network.dart';  
import  
'package:roadx/widgets/custom_categories.dart';  
import 'package:roadx/widgets/pages.dart';
```

```
class HomePage extends StatefulWidget {  
  
    const HomePage({super.key});  
  
    static String id = 'HomePage';  
  
    @override  
  
        State<HomePage>      createState()      =>  
        _HomePageState();  
  
}  
  
class _HomePageState extends State<HomePage> {  
  
    GlobalKey<ScaffoldState>      scaffoldKey      =  
    GlobalKey();  
  
    @override  
  
    Widget build(BuildContext) {  
  
        return Scaffold(  
  
            body: ListView(  
  
                children: [  
  
                    CarouselSlider(  
  
                        items: [  
  
                            Image.asset(  

```

```
"assets/images/Slider1.png",  
 ),  
 Image.asset(  
 "assets/images/Slider2.png",  
 ),  
 Image.asset(  
 "assets/images/Slider3.png",  
 ),  
 ],  
 options: CarouselOptions(  
 scrollDirection: Axis.horizontal,  
 height: 300,  
 aspectRatio: 16 / 8,  
 viewportFraction: 1.5,  
 autoPlay: true,  
 autoPlayInterval: const Duration(seconds: 3),  
 autoPlayAnimationDuration: const  
 Duration(milliseconds: 800),  
 ),
```

```
        ),  
  
        const SizedBox(height: 15),  
  
        const Pages(),  
  
        Container(margin: const  
        EdgeInsets.symmetric(horizontal: 20, vertical: 20),  
  
        child: const Text(  
  
            "Top Categories",  
  
            style: TextStyle(fontWeight: FontWeight.bold,  
            fontSize: 25),  
  
        ),  
  
        ),  
  
        Row(  
  
            children: [  
  
            GestureDetector(  
  
                onTap: () {  
  
                    Navigator.pushNamed(context,  
Frontend.id);  
  
                },  
  
                child: const CustomCategories(  
  
                    title: 'Frontend',  

```

```
        img:  
      "assets/images/resources/frontend.jpg")),  
  
      GestureDetector(  
  
        onTap: () {  
  
          Navigator.pushNamed(context, Backend.id);  
  
        },  
  
        child: const CustomCategories(  
  
          title: 'Backend',  
  
          img:  
        "assets/images/resources/backend.png"),  
  
      ),  
  
    ],  
  
  ),  
  
  Row(  
  
    children: [  
  
      GestureDetector(  
  
        onTap: () {  
  
          Navigator.pushNamed(context,  
CrossPlatform.id);  
  
        },  
    ]  
  )  
)
```

```
        child: const CustomCategories(  
            title: 'Mobile Application',  
            img: "assets/images/resources/cross_platform.jpg"),  
        ),  
        GestureDetector(  
            onTap: () {  
                Navigator.pushNamed(context, Network.id);  
            },  
            child: const CustomCategories(  
                title: 'Network',  
                img:  
                "assets/images/resources/network.png"),  
            ),  
        ],  
    ),  
    const SizedBox(height: 15),  
],  
),  
); })
```

5.3-Cateogires page

```
import 'package:flutter/material.dart';
import 'package:roadx/constants.dart';
import 'package:roadx/resources/android.dart';
import
'package:roadx/resources/artificialintelligence.dart';
import 'package:roadx/resources/backend.dart';
import 'package:roadx/resources/cross_platform.dart';
import 'package:roadx/resources/cyber_security.dart';
import 'package:roadx/resources/frontend.dart';
import 'package:roadx/resources/network.dart';
import 'package:roadx/resources/ui_ux_design.dart';
import 'package:roadx/resources/wordpress.dart';
import 'package:roadx/widgets/custom_hero.dart';
import 'package:roadx/widgets/nav.dart';
import
'package:vertical_card_pager/vertical_card_pager.dart';

class Categories extends StatefulWidget {
```

```
const Categories({super.key});  
  
static String id = 'Categories';  
  
@override  
  
State<Categories> createState() =>  
_CategoriesState();  
}  
  
class _CategoriesState extends State<Categories> {  
  
@override  
  
Widget build(BuildContext context) {  
  
List<String> titles = [  
    "Frontend",  
    "Backend",  
    "Wordpress",  
    "Android Application",  
    "Cross Platform",  
    "Network",  
    "Cyber Security",  
    "Artificial Intelligence",  
    "UI/UX Design",  
];  
}
```

```
];
List<Widget> images = [
    const CustomHero(
        tag: 'Frontend', img:
        'assets/images/resources/frontend.jpg'),
    const CustomHero(
        tag: 'Backend', img:
        'assets/images/resources/backend.png'),
    const CustomHero(
        tag: 'Wordpress', img:
        'assets/images/resources/wordpress.png'),
    const CustomHero(
        tag: 'Android Application',
        img: 'assets/images/resources/android.jpg'),
    const CustomHero(
        tag: 'Cross Platform',
        img:
        'assets/images/resources/cross_platform.jpg'),
    const CustomHero(
        tag: 'Network', img:
        'assets/images/resources/network.png'),
```

```
const CustomHero(  
    tag: 'Cyber Security',  
    img:  
    'assets/images/resources/cyber_security.jpg'),  
  
const CustomHero(  
    tag: 'Artificial Intelligence',  
    img: 'assets/images/resources/artificial-  
intelligence.jpg'),  
  
const CustomHero(  
    tag: 'UI/UX Design',  
    img:  
    'assets/images/resources/uii_ux_design.png'),  
];  
  
final List<Widget> links = [  
    const Frontend(),  
    const Backend(),  
    const Wordpress(),  
    const AndroidApplication(),  
    const CrossPlatform(),  
    const Network(),
```

```
const CyberSecurity(),  
const ArtificialIntelligence(),  
const UIUXDesign(),  
];  
  
return Scaffold(  
    appBar: AppBar(  
        backgroundColor: kPrimaryColor,  
        leading: IconButton(  
            onPressed: () {  
                Navigator.pushNamed(context, Nav.id);  
            },  
            icon: const Icon(  
                Icons.arrow_back,  
                color: Colors.white,  
            )),  
        elevation: 0.0,  
        title: const Text(  
            "Categories",  
        ),  
    ),  
    body: Container(  
        padding: EdgeInsets.all(16),  
        child: Column(  
            children: [  
                Text("Category A"),  
                Text("Category B"),  
                Text("Category C"),  
            ],  
        ),  
    ),  
);
```

```
        style: TextStyle(  
            color: Colors.white,  
        ),  
        ),  
    ),  
  
body: SafeArea(  
    child: Column(  
        children: [  
            Expanded(  
                child: Container(  
                    margin:  
                        const EdgeInsets.symmetric(horizontal:  
                            10, vertical: 10),  
                    child: VerticalCardPager(  
                        titles: titles,  
                        images: images,  
                        textStyle: const TextStyle(  
                            color: Colors.black,  
                            fontWeight: FontWeight.bold,  
                        ),  
                    ),  
                ),  
            ),  
        ],  
    ),  
),
```

```
        ),  
  
        onSelectedItem: (index) {  
  
            Navigator.push(context,  
  
                MaterialPageRoute(builder: (_)  
                    =>  
                links[index]));  
  
            },  
  
        ),  
  
        ),  
  
    ],  
  
    ),  
  
    );  
}  
}  
}
```

5.4-Chatbot page

```
import 'package:flutter/material.dart';

class Chatbot extends StatefulWidget {

    const Chatbot({super.key});

    static String id = 'Chatbot';

    @override

    State<Chatbot> createState() => _ChatbotState();
}

class _ChatbotState extends State<Chatbot> {

    List<Map<String, String>> messages = [];

    TextEditingController controller = TextEditingController();

    void _sendMessage(String message) {

        setState(() {

            messages.add({'message': message, 'type': 'sent'});

        });

        controller.clear();
    }

    @override
}
```

```
Widget build(BuildContext context) {  
  return Column(  
    children: <Widget>[  
      Expanded(  
        child: ListView.builder(  
          itemCount: messages.length,  
          itemBuilder: (context, index) {  
            var message = messages[index];  
            return Padding(  
              padding:  
                const EdgeInsets.symmetric(horizontal:  
                  10, vertical: 15),  
              child: Row(  
                mainAxisAlignment: message['type'] ==  
                  'sent'  
                  ? MainAxisAlignment.start  
                  : MainAxisAlignment.end,  
                children: [  
                  Flexible(  

```

```
        child: Container(  
            decoration: BoxDecoration(  
                color: message['type'] == 'sent'  
                    ? const Color(0xFF86C3FD)  
                    : Colors.green[100],  
                borderRadius:  
                    BorderRadius.circular(8.0),  
            ),  
            padding: const EdgeInsets.all(8.0),  
            child: Text(  
                style: const TextStyle(color:  
                    Colors.black87),  
                message['message']!,  
                overflow: TextOverflow.visible,  
            ),  
        ),  
    ),  
],  
,
```

```
    );
  },
),
),
Container(
padding: const EdgeInsets.all(8.0),
child: Row(
children: <Widget>[
Expanded(
child: TextField(
controller: controller,
decoration: const InputDecoration(
hintText: 'Type a message...',  
border: OutlineInputBorder(),
),
),
),
const SizedBox(width: 8.0),
```

```
    IconButton(  
      icon: const Icon(Icons.send),  
      onPressed: () {  
        if (controller.text.isNotEmpty) {  
          _sendMessage(controller.text);  
        }  
      },  
    ),  
  ],  
,  
],  
);  
}  
}
```

5.5-documentation page

```
import 'package:flutter/material.dart';
import 'package:roadx/constants.dart';
import 'package:roadx/widgets/custom_list_tile.dart';
import 'package:roadx/widgets/nav.dart';
import 'package:url_launcher/url_launcher.dart';

// Frontend

final Uri _frontendfirst =
    Uri.parse("https://developer.mozilla.org/en-US/docs/Web/CSS");

final Uri _frontendsecond =
Uri.parse('https://devdocs.io/css-display/');

final Uri _frontendthird =
    Uri.parse('https://www.w3schools.com/cssref/index.php');

// Backend

final Uri _backendfirst =
Uri.parse('https://www.w3schools.com/php/');

final Uri _backendsecond =
Uri.parse('https://dev.mysql.com/doc/');
```

```
final Uri _backendthird =  
Uri.parse('https://nodejs.org/docs/latest/api/');  
  
// Wordpress  
  
final Uri _wordpressfirst =  
Uri.parse('https://codex.wordpress.org/');  
  
final Uri _wordpressecond =  
  
    Uri.parse(  
https://developer.wordpress.org/themes/basics/');  
  
final Uri _wordpressthird =  
Uri.parse('https://developer.wordpress.org/');  
  
// Android  
  
final Uri _androidfirst =  
Uri.parse('https://developer.android.com/guide');  
  
final Uri _androidsecond =  
  
Uri.parse('https://okaythis.com/developer/documentation/v1/android');  
  
final Uri _androidthird =  
  
    Uri.parse('https://kotlinlang.org/docs/android-overview.html');  
  
// Crossplatform  
  
final Uri _crossplatformfirst =  
Uri.parse('https://dart.dev/guides');
```

```
final Uri _crossplatformsecond =  
Uri.parse('https://docs.flutter.dev/');  
  
final Uri _crossplatformthird =  
  
Uri.parse('https://firebase.flutter.dev/docs/overview/');  
  
// Network  
  
final Uri _networkfirst = Uri.parse(  
  
'https://www.scribd.com/document/389397668/Cisco-  
CCNA-1-Introduction-to-Networks');  
  
final Uri _networksecond = Uri.parse(  
  
'https://learningnetwork.cisco.com/s/all-  
media?ccid=sem&dtid=mediabuy&oid=sem&gad_sour  
ce=1&gclid=CjwKCAjww_iwBhApEiwAuG6ccAa3iBv7e  
IkCSwZmk21p3JsvjWNQt1Hil3ttyngVHhZNMt5cNMq  
GRoCHEwQAvD_BwE&gclsrc=aw.ds');
```



```
final Uri _networkthird = Uri.parse(  
  
'https://www.wireshark.org/docs/wsug_html_chunked/  
ChapterIntroduction.html');
```



```
final Uri _networkfourth = Uri.parse(  
  
)
```

```
'https://www.wireshark.org/download/docs/Wireshark%  
20User%27s%20Guide.pdf');  
  
final Uri _networkfifth = Uri.parse(  
    'https://info.microsoft.com/rs/157-GQE-  
382/images/IntroducingWindowsServer2016_ebook.pd  
f');  
  
// Cyber  
  
final Uri _cybersecurityfirst = Uri.parse(  
    'https://www.simplilearn.com/tutorials/cyber-  
security-tutorial/cyber-security-for-beginners');  
  
final Uri _cybersecuritysecond =  
  
Uri.parse('https://www.w3schools.com/cybersecurity');  
  
final Uri _cybersecuritythird =  
  
Uri.parse('https://us.norton.com/blog/how-  
to/cybersecurity-basics');  
  
final Uri _cybersecurityfourth =  
  
Uri.parse('https://www.geeksforgeeks.org/cyber-  
security-tutorial/');  
  
final Uri _cybersecurityfifth = Uri.parse(
```

```
'https://www.eccouncil.org/cybersecurity-
exchange/cyber-novice/free-cybersecurity-courses-
beginners');

// AI

final Uri _aifirst =
Uri.parse('https://www.mdpi.com/2073-431X/12/5/91');

final Uri _aisecond = Uri.parse(
    'https://scikit-
learn.org/stable/modules/neural_networks_supervised.
html');

final Uri _aithird =
    Uri.parse('https://www.javatpoint.com/artificial-
neural-network');

final Uri _aifourth =
Uri.parse('https://cloud.google.com/document-ai/docs');

final Uri _aifth = Uri.parse(
    'https://docs.h2o.ai/h2o/latest-stable/h2o-docs/data-
science/deep-learning.html');

final Uri _aisixth = Uri.parse(
    'https://ecampusontario.pressbooks.pub/informationsy
stemscdn/chapter/13-5-machine-learning-and-deep-
learning');
```

```
// UI UX Design

final Uri _uiuxdesignfirst = Uri.parse(
    'https://www.pencilandpaper.io/articles/ux-design-
documentation-guide');

final Uri _uiuxdesignsecond = Uri.parse(
    'https://www.uxpin.com/studio/blog/3-step-guide-
lightweight-ux-documentation/');

final Uri _uiuxdesignthird = Uri.parse(
    'https://www.uiprep.com/blog/the-best-way-to-
document-ux-ui-design');

class Documentation extends StatefulWidget {

    const Documentation({super.key});

    static String id = 'Documentation';

    @override
        State<Documentation>      createState()      =>
    _DocumentationState();
}

class      _DocumentationState      extends
State<Documentation> {

// Frontend

    Future<void> __frontendfirst() async {
```

```
if (!await launchUrl(_frontendfirst)) {  
    throw Exception('Could not launch  
$_frontendfirst');}  
  
Future<void> __frontendsecond() async {  
    if (!await launchUrl(_frontendsecond)) {  
        throw Exception('Could not launch  
$_frontendsecond');  
    }  
}  
  
Future<void> __frontendthird() async {  
    if (!await launchUrl(_frontendthird)) {  
        throw Exception('Could not launch  
$_frontendthird');  
    }  
}  
  
// Backend  
  
Future<void> __backendfirst() async {  
    if (!await launchUrl(_backendfirst)) {  
        throw Exception('Could not launch $_backendfirst');  
    }  
}
```

```
Future<void> __backendsecond() async {
    if (!await launchUrl(_backendsecond)) {
        throw Exception('Could not launch
$_backendsecond');
    }
}

Future<void> __backendthird() async {
    if (!await launchUrl(_backendthird)) {
        throw Exception('Could not launch
$_backendthird');
    }
}

// Wordpress

Future<void> __wordpressfirst() async {
    if (!await launchUrl(_wordpressfirst)) {
        throw Exception('Could not launch
$_wordpressfirst');
    }
}

Future<void> __wordpressecond() async {
```

```
if (!await launchUrl(_wordpressecond)) {  
    throw Exception('Could not launch  
$_wordpressecond');  
}  
}  
  
Future<void> __wordpressthird() async {  
    if (!await launchUrl(_wordpressthird)) {  
        throw Exception('Could not launch  
$_wordpressthird');  
    }
}  
}  
  
// Android  
  
Future<void> __androidfirst() async {  
    if (!await launchUrl(_androidfirst)) {  
        throw Exception('Could not launch $_androidfirst');  
    }
}  
}  
  
Future<void> __androidsecond() async {  
    if (!await launchUrl(_androidsecond)) {
```

```
        throw      Exception('Could      not      launch
$_androidsecond');

    }

}

Future<void> __androidthird() async {

    if (!await launchUrl(_androidthird)) {

        throw Exception('Could not launch $_androidthird');

    }
}

// Crossplatform

Future<void> __crossplatformfirst() async {

    if (!await launchUrl(_crossplatformfirst)) {

        throw      Exception('Could      not      launch
$_crossplatformfirst');

    }
}

Future<void> __crossplatformsecond() async {

    if (!await launchUrl(_crossplatformsecond)) {

        throw      Exception('Could      not      launch
$_crossplatformsecond');

    }
}

Future<void> __crossplatformthird() async {
```

```
if (!await launchUrl(_crossplatformthird)) {  
    throw Exception('Could not launch  
$_crossplatformthird');  
}  
  
// network  
  
Future<void> __networkfirst() async {  
    if (!await launchUrl(_networkfirst)) {  
        throw Exception('Could not launch $_networkfirst');  
    }  
}  
  
Future<void> __networksecond() async {  
    if (!await launchUrl(_networksecond)) {  
        throw Exception('Could not launch  
$_networksecond');  
    }  
}  
  
Future<void> __networkthird() async {  
    if (!await launchUrl(_networkthird)) {  
        throw Exception('Could not launch $_networkthird');  
    }  
}
```

```
}

Future<void> __networkfourth() async {
    if (!await launchUrl(_networkfourth)) {
        throw Exception('Could not launch $_networkfourth');
    }
}

Future<void> __networkfifth() async {
    if (!await launchUrl(_networkfifth)) {
        throw Exception('Could not launch $_networkfifth');
    }
}

// cyber

Future<void> __cybersecurityfirst() async {
    if (!await launchUrl(_cybersecurityfirst)) {
        throw Exception('Could not launch $_cybersecurityfirst');
    }
}

Future<void> __cybersecuritysecond() async {
    if (!await launchUrl(_cybersecuritysecond)) {
```

```
        throw      Exception('Could      not      launch
$_cybersecuritysecond');

    }

Future<void> __cybersecuritythird() async {
    if (!await launchUrl(_cybersecuritythird)) {

        throw      Exception('Could      not      launch
$_cybersecuritythird');

    }

Future<void> __cybersecurityfourth() async {
    if (!await launchUrl(_cybersecurityfourth)) {

throw      Exception('Could      not      launch
$_cybersecurityfourth');

    }

Future<void> __cybersecurityfifth() async {
    if (!await launchUrl(_cybersecurityfifth)) {

        throw      Exception('Could      not      launch
$_cybersecurityfifth');

    }

// AI

Future<void> __aifirst() async {
```

```
if (!await launchUrl(_aifirst)) {  
    throw Exception('Could not launch ${_aifirst}');  
}  
  
Future<void> __aisecond() async {  
    if (!await launchUrl(_aisecond)) {  
        throw Exception('Could not launch ${_aisecond}');  
    }  
}  
  
Future<void> __aithird() async {  
    if (!await launchUrl(_aithird)) {  
        throw Exception('Could not launch ${_aithird}');  
    }  
}  
  
Future<void> __aifourth() async {  
    if (!await launchUrl(_aifourth)) {  
        throw Exception('Could not launch ${_aifourth}');  
    }  
}  
  
Future<void> __aifth() async {  
    if (!await launchUrl(_aifth)) {  
        throw Exception('Could not launch ${_aifth}');  
    }  
}
```

```
        throw Exception('Could not launch ${_aififth}');

    }

Future<void> __aisixth() async {
    if (!await launchUrl(_aisixth)) {

        throw Exception('Could not launch ${_aisixth}');

    }
}

// UI UX Design

Future<void> __uiuxdesignfirst() async {
    if (!await launchUrl(_uiuxdesignfirst)) {

        throw      Exception('Could      not      launch
${_uiuxdesignfirst}');

    }
}

Future<void> __uiuxdesignsecond() async {
    if (!await launchUrl(_uiuxdesignsecond)) {

        throw      Exception('Could      not      launch
${_uiuxdesignsecond}');

    }
}

Future<void> __uiuxdesignthird() async {
    if (!await launchUrl(_uiuxdesignthird)) {
```

```
        throw      Exception('Could      not      launch
$_uiuxdesignthird');}}
```

@override

```
Widget build(BuildContext context) {
```

return Scaffold(

```
        appBar: AppBar(
```

```
            backgroundColor: kPrimaryColor,
```

```
            leading: IconButton(
```

```
                onPressed: () {
```

```
                    Navigator.pushNamed(context, Nav.id);
```

```
                },
```

```
                icon: const Icon(
```

```
                    Icons.arrow_back,
```

```
                    color: Colors.white,
```

```
                )),
```

```
            elevation: 0.0,
```

```
            title: const Text(
```

```
                "Documentation",
```

```
                style: TextStyle(
```

```
        color: Colors.white,  
    ),),),  
  
body: ListView.custom(  
    childrenDelegate: SliverChildListDelegate(  
    [  
        // Front  
        const SizedBox(  
            height: 10,  
        ),  
        CustomListTile(  
            onTap: __frontendfirst,  
            img:  
                'assets/images/content_of_stacks/front  
documentation link 1.png',  
            title: 'developer.mozilla',  
            subtitle: 'Frontend',  
        ),  
        const SizedBox(  
            height: 10,
```

```
        ),  
        CustomListTile(  
            onTap: __frontendsecond,  
            img:  
                'assets/images/content_of_stacks/front  
documentation link 2.png',  
            title: 'devdocs.io',  
            subtitle: 'Frontend',  
        ),  
        const SizedBox(  
            height: 10,  
        ),  
        CustomListTile(  
            onTap: __frontendthird,  
            img:  
                'assets/images/content_of_stacks/front  
documentation link 3.png',  
            title: 'www.w3schools',  
            subtitle: 'Frontend',  
        ),
```

```
        ),  
        // Backend  
        const SizedBox(  
            height: 10,  
        ),  
        CustomListTile(  
            onTap: __backendfirst,  
            img:  
'assets/images/content_of_stacks/vector3.png',  
            title: 'www.w3schools',  
            subtitle: 'Backend',  
        ),  
        const SizedBox(  
            height: 10,  
        ),  
        CustomListTile(  
            onTap: __backendsecond,  
            img: 'assets/images/content_of_stacks/mysql-  
logo.png',
```

```
        title: 'dev.mysql',
        subtitle: 'Backend',
    ),
const SizedBox(
    height: 10,
),
CustomListTile(
    onTap: __backendthird,
    img:
'assets/images/content_of_stacks/svgexport-1.png',
    title: 'nodejs',
    subtitle: 'Backend',
),
// Wordpress
const SizedBox(
    height: 10,
),
CustomListTile(
    onTap: __wordpressfirst,
```

```
    img:'assets/images/content_of_stacks/wordpress
documentation link 1.png',
        title: 'codex.wordpress',
        subtitle: 'Wordpress',
    ),
const SizedBox(
    height: 10,
),
CustomListTile(
    onTap: __wordpressesecond,
    img:
'assets/images/content_of_stacks/wordpress
documentation link 2.png',
        title: 'codex.wordpress',
        subtitle: 'Wordpress',
    ),
const SizedBox(
    height: 10,),
CustomListTile(
```

```
onTap: __wordpressthird,  
img:  
'assets/images/content_of_stacks/wordpress  
documentation link 3.png',  
title: 'codex.wordpress',  
subtitle: 'Wordpress',  
),  
// Android  
const SizedBox(  
height: 10,  
),  
CustomListTile(  
onTap: __androidfirst,  
img:'assets/images/content_of_stacks/android  
documentation link 1.png',  
title: 'developer.android',  
subtitle: 'Android Application',  
),  
const SizedBox(  
height: 10,
```

```
        ),  
  
        CustomListTile(  
            onTap: __androidsecond,  
            img:  
'assets/images/content_of_stacks/android  
documentation link 2.png',  
            title: 'okaythis',  
            subtitle: 'kotlinlang',  
        ),  
  
        const SizedBox(  
            height: 10,  
        ),  
  
        CustomListTile(  
            onTap: __androidthird,  
            img:  
'assets/images/content_of_stacks/android  
documentation link 3.png',  
            title: 'kotlinlang',  
            subtitle: 'kotlinlang',),  
  
        // Crossplatform
```

```
const SizedBox(  
    height: 10,  
,  
CustomListTile(  
    onTap: __crossplatformfirst,  
    img:  
'assets/images/content_of_stacks/Dart.png',  
    title: 'Dart Documentation',  
    subtitle: 'CrossPlatform',  
,  
const SizedBox(  
    height: 10,  
,  
CustomListTile(  
    onTap: __crossplatformsecond,  
    img:  
'assets/images/content_of_stacks/Flutter.png',  
    title: 'Flutter Documentation',  
    subtitle: 'CrossPlatform',
```

```
        ),  
  
        const SizedBox(  
            height: 10,  
        ),  
  
        CustomListTile(  
            onTap: __crossplatformthird,  
            img:  
'assets/images/content_of_stacks/flutterfire.png',  
            title: 'FlutterFire Documentation',  
            subtitle: 'CrossPlatform',  
        ),  
  
        // Network  
  
        const SizedBox(  
            height: 10,  
        ),  
  
        CustomListTile(  
            onTap: __networkfirst,  
            img:  
'assets/images/content_of_stacks/download.png',
```

```
        title: 'scribd',  
        subtitle: 'Network',  
    ),  
    const SizedBox(  
        height: 10,  
    ),  
    CustomListTile(  
        onTap: __networksecond,  
        img:'assets/images/content_of_stacks/cisco-logo-transparent.png',  
        title: 'learningnetwork',  
        subtitle: 'Network',  
    ),  
    const SizedBox(  
        height: 10,  
    ),  
    CustomListTile(  
        onTap: __networkthird,
```

```
        img:  
        'assets/images/content_of_stacks/wireshark-logo.png',  
  
        title: 'wireshark',  
  
        subtitle: 'Network',  
  
  
        const SizedBox(  
  
            height: 10,  
  
        ),  
  
        CustomListTile(  
  
            onTap: __networkfourth,  
  
            img:  
            'assets/images/content_of_stacks/network.png',  
  
            title: 'wireshark',  
  
            subtitle: 'Network',  
  
        ),  
  
        const SizedBox(  
  
            height: 10,  
  
        ),  
  
        CustomListTile(  
  
    )
```

```
        onTap: __networkfifth,  
        img:  
        'assets/images/content_of_stacks/network1.jpg',  
        title: 'info.microsoft',  
        subtitle: 'Network',  
    ),  
    // Cyber  
    const SizedBox(  
        height: 10,  
    ),  
    CustomListTile(  
        onTap: __cybersecurityfirst,  
        img:    'assets/images/content_of_stacks/gfg-  
gg-logo.png',  
        title: 'simplilearn',  
        subtitle: 'Cyber Security',  
    ),  
    const SizedBox(  
        height: 10,
```

```
        ),  
  
        CustomListTile(  
            onTap: __cybersecuritysecond,  
            img:  
'assets/images/content_of_stacks/vector3.png',  
            title: 'w3schools',  
            subtitle: 'Cyber Security',),  
  
        const SizedBox(  
            height: 10,  
        ),  
  
        CustomListTile(  
            onTap: __cybersecuritythird,  
            img: 'assets/images/content_of_stacks/Norton  
Logo.png',  
            title: 'us.norton',  
            subtitle: 'Cyber Security',  
        ),  
  
        const SizedBox(  
            height: 10,
```

```
        ),  
  
        CustomListTile(  
            onTap: __cybersecurityfourth,  
            img:  
                'assets/images/content_of_stacks/EC-  
                Council_Cybersecurity-Exchange-tab.png',  
            title: 'geeksforgeeks',  
            subtitle: 'Cyber Security',  
        ),  
  
        const SizedBox(  
            height: 10,  
        ),  
  
        CustomListTile(  
            onTap: __cybersecurityfifth,  
            img:  
                'assets/images/content_of_stacks/logo.png',  
            title: 'eccouncil',  
            subtitle: 'Cyber Security',  
        ),
```

```
// AI

const SizedBox(
    height: 10,
),
CustomListTile(
    onTap: __aifirst,
    img:'assets/images/content_of_stacks/ai
documentation link 1.png',
    title: 'mdpi',
    subtitle: 'Artificial Intelligence',
),
const SizedBox(
    height: 10,
),
CustomListTile(
    onTap: __aisecond,
    img:
        'assets/images/content_of_stacks/ai
documentation link 2.png',
```

```
        title: 'scikit-learn',  
        subtitle: 'Artificial Intelligence',  
    ),  
    const SizedBox(  
        height: 10,  
    ),  
    CustomListTile(  
        onTap: __aithird,  
        img:  
            'assets/images/content_of_stacks/ai  
documentation link 3.png',  
        title: 'javatpoint',  
        subtitle: 'Artificial Intelligence',  
    ),  
    const SizedBox(  
        height: 10,  
    ),  
    CustomListTile(  
        onTap: __aifourth,
```

```
    img:  
        'assets/images/content_of_stacks/ai  
documentation link 4.png',  
  
        title: 'cloud.google',  
  
        subtitle: 'Artificial Intelligence',  
  
,  
  
const SizedBox(  
  
    height: 10,  
  
,  
  
CustomListTile(  
  
    onTap: __aififth,  
  
    img:  
        'assets/images/content_of_stacks/ai  
documentation link 5.png',  
  
        title: 'docs.h2o.ai',  
  
        subtitle: 'Artificial Intelligence',),  
  
const SizedBox(  
  
    height: 10,  
  
,
```

```
CustomListTile(  
    onTap: __aisixth,  
    img:  
        'assets/images/content_of_stacks/ai  
documentation link 6.png',  
    title: 'ecampusontario.pressbooks',  
    subtitle: 'Artificial Intelligence',  
,  
// UI UX Design  
const SizedBox(  
    height: 10,  
,  
CustomListTile(  
    onTap: __uiuxdesignfirst,  
    img: 'assets/images/content_of_stacks/ui ux  
design link 1.png',  
    title: 'pencilandpaper.io',  
    subtitle: 'UI UX Design',  
,
```

```
const SizedBox(  
    height: 10,  
,  
CustomListTile(  
    onTap: __uiuxdesignsecond,  
    img: 'assets/images/content_of_stacks/ui ux  
design link 2.png',  
    title: 'uxpin',  
    subtitle: 'UI UX Design',  
,  
const SizedBox(  
    height: 10,  
,  
CustomListTile(  
    onTap: __uiuxdesignthird,  
    img: 'assets/images/content_of_stacks/ui ux design  
link 3.png',  
    title: 'uiprep',  
    subtitle: 'UI UX Design', ),],),),);})}
```

5.6-extra knowledge page

```
import 'package:flutter/material.dart';

import 'package:roadx/constants.dart';

import
'package:roadx/pages_of_extra_knowledge/framework
s.dart';

import
'package:roadx/pages_of_extra_knowledge/certificate
s.dart';

import
'package:roadx/pages_of_extra_knowledge/important_
skills.dart';

import
'package:roadx/pages_of_extra_knowledge/knowledg
e.dart';

import
'package:roadx/pages_of_extra_knowledge/recommen
ded_websites.dart';

import
'package:roadx/pages_of_extra_knowledge/templates.
dart';
```

```
import  
'package:roadx/pages_of_extra_knowledge/tools.dart';  
  
import 'package:roadx/widgets/nav.dart';  
  
class ExtraKnowledge extends StatelessWidget {  
  
  ExtraKnowledge({super.key});  
  
  static String id = 'Extra Knowledge';  
  
  final List<String> gridTitles = [  
  
    "Certificates",  
  
    "Templates",  
  
    "Important Skills",  
  
    "Tools",  
  
    "Frameworks",  
  
    'Recommended Websites',  
  
    'Knowledge',];  
  
  final List<String> imagePaths = [  
  
    'assets/images/certificates.png',  
  
    'assets/images/templates.png',  
  
    'assets/images/important skills.jpg',  
  
    'assets/images/tools.jpg',
```

```
'assets/images/frameworks.jpg',
'assets/images/Recommended Websites.jpg',
'assets/images/knowledge.jpg'],);

final List<Widget> navigationTargets = [
    const Certificates(),
    Templates(),
    const ImportantSkills(),
    const Tools(),
    const Frameworks(),
    const RecommendedWebsites(),
    const Knowledge(),
];

@Override
Widget build(BuildContext context) {
    return Scaffold(
        appBar: AppBar(
            backgroundColor: kPrimaryColor,
            leading: IconButton(
```

```
 onPressed: () {
    Navigator.pushNamed(context, Nav.id);
},
icon: const Icon(
    Icons.arrow_back,
    color: Colors.white,
)),
elevation: 0.0,
title: const Text(
    "Extra Knowledge",
    style: TextStyle(
        color: Colors.white,
    ),),
),
body: GridView.count(
    crossAxisCount: 1,
    children: List.generate(gridTitles.length, (index) {
        return InkWell(
            onTap: () {
```

```
Navigator.push(  
    context,  
    MaterialPageRoute(  
        builder: (context) =>  
        navigationTargets[index]), );},  
  
    child: Container(  
  
        margin: const EdgeInsets.all(4.0),  
  
        child: GridTile(  
  
            footer: Container(  
  
                color: Colors.black.withOpacity(0.5),  
  
                child: Center(  
  
                    child: Text(  
  
                        gridTitles[index],  
  
                        style: const TextStyle(color:  
Colors.white),  
  
                        textAlign: TextAlign.center,  
                    ),  
                ),  
            ),  
        ),
```

```
        child: Image.asset(  
            imagePaths[index],  
            fit: BoxFit.cover,)),),);}), ),);}}
```

5.7.programming languages page

```
import 'package:flutter/material.dart';  
  
import 'package:roadx/constants.dart';  
  
import  
'package:roadx/pages_of_programming_languages/cp  
lusplus_programming_language.dart';  
  
import  
'package:roadx/pages_of_programming_languages/cs  
harp_programming_language.dart';  
  
import  
'package:roadx/pages_of_programming_languages/ja  
va_programming_language.dart';  
  
import  
'package:roadx/pages_of_programming_languages/ja  
vascript_programming_language.dart';  
  
import  
'package:roadx/pages_of_programming_languages/py  
thon_programming_language.dart';  
  
import 'package:roadx/widgets/nav.dart';
```

```
class ProgrammingLanguages extends StatelessWidget {  
    ProgrammingLanguages({super.key});  
    static String id = 'Programming Languages';  
    final List<String> gridTitles = ["Java", "C++", "C#",  
    "Python", "Javascript"];  
    final List<String> imagePaths = [  
        'assets/images/programming_languages/java.jpg',  
  
        'assets/images/programming_languages/911_c_logo.j  
        pg',  
  
        'assets/images/programming_languages/CSharp.png',  
        'assets/images/programming_languages/python.png',  
        'assets/images/programming_languages/javascript.bn  
        g',  
    ];  
    final List<Widget> navigationTargets = [  
        const JavaProgrammingLanguage(),  
        const CPlusPlusProgrammingLanguage(),  
        const CSharpProgrammingLanguage(),  
        const PythonProgrammingLanguage(),
```

```
const JavaScriptProgrammingLanguage(),  
];  
  
@override  
Widget build(BuildContext context) {  
  return Scaffold(  
    appBar: AppBar(  
      backgroundColor: kPrimaryColor,  
      leading: IconButton(  
        onPressed: () {  
          Navigator.pushNamed(context, Nav.id);  
        },  
        icon: const Icon(  
          Icons.arrow_back,  
          color: Colors.white,  
        )),  
      elevation: 0.0  
      title: const Text(  
        "Programming Languages",  
    ),  
  );  
}  
});
```

```
style: TextStyle(  
    color: Colors.white,),),),  
  
body: GridView.count(  
    crossAxisCount: 1,  
    children: List.generate(gridTitles.length, (index) {  
        return InkWell(  
            onTap: () {  
                Navigator.push(  
                    context,  
                    MaterialPageRoute(  
                        builder: (context) =>  
                        navigationTargets[index]),);},  
            child: Container(  
                margin: const EdgeInsets.all(4.0),  
                child: GridTile(  
                    footer: Container(  
                        color: Colors.black.withOpacity(0.5),  
                        child: Center(  
                            child: Text(  
                                style: TextStyle(  
                                    color: Colors.white,),),),  
                ),  
            ),  
        );  
    }),  
);
```

```
gridTitles[index],  
        style: const TextStyle(color:  
Colors.white),  
        textAlign: TextAlign.center,)),),  
        child: Image.asset(  
            imagePaths[index],  
            fit: BoxFit.cover,)),),);});}),);})}
```

5.8-learning kids page

```
import 'package:flutter/material.dart';  
  
import 'package:roadx/constants.dart';  
  
import 'package:roadx/widgets/nav.dart';  
  
import 'package:roadx/widgets/custom_list_tile.dart';  
  
import 'package:url_launcher/url_launcher.dart';  
  
final Uri _programmingkidsfirst = Uri.parse(  
'https://www.youtube.com/playlist?list=PLnb0FwCbM-  
50UuNjpelrdEnISbbMK891Q');
```

```
final Uri _programmingkidssecond = Uri.parse(  
'https://www.youtube.com/playlist?list=PLzTFSn-  
Bzi_wguTJLHUgrJgCy42mdT-Yn');
```

```
final Uri _computerbasicsfirst = Uri.parse(  
'https://www.youtube.com/playlist?list=PLzTFSn-  
Bzi_wguTJLHUgrJgCy42mdT-Yn');
```

```
'https://www.youtube.com/playlist?list=PL4316FC411AD077AA');

final Uri _computerbasicssecond = Uri.parse(
    'https://www.youtube.com/watch?v=W7-u_w-0zHc&list=PLqlleLpAMfxGAkXyW-QIwBPYDXpxAmb5La');

final Uri _computerfundamentalsfirst = Uri.parse(
    'https://www.youtube.com/playlist?list=PLSzumVgeINxSi7pWzRTPOSiqdXHLI0GP');

final Uri _flowchartsfirst = Uri.parse(
    'https://www.youtube.com/playlist?list=PLMQ4khUWGNI-_4tGH-2Gq-06yZbzl5az');

final Uri _flowchartssecond = Uri.parse(
    'https://www.youtube.com/playlist?list=PLQ8QGneqFjxNnPszJsQs2qS1m0uqfXCob');

class LearningKids extends StatefulWidget {
    const LearningKids({super.key});

    static String id = 'Learning Kids';

    @override
    State<LearningKids>      createState()      =>
        _LearningKidsState();
}
```

```
}

class _LearningKidsState extends State<LearningKids> {

    Future<void> __programmingkidsfirst() async {
        if (!await launchUrl(_programmingkidsfirst)) {
            throw Exception('Could not launch $_programmingkidsfirst');
        }
    }

    Future<void> __programmingkidssecond() async {
        if (!await launchUrl(_programmingkidssecond)) {
            throw Exception('Could not launch $_programmingkidssecond');
        }
    }

    Future<void> __computerbasicsfirst() async {
        if (!await launchUrl(_computerbasicsfirst)) {
            throw Exception('Could not launch $_computerbasicsfirst');
        }
    }

    Future<void> __computerbasicssecond() async {
        if (!await launchUrl(_computerbasicssecond)) {
```

```
        throw      Exception('Could      not      launch
$_computerbasicssecond');

    }

Future<void> __computerfundamentalsfirst() async {
    if (!await launchUrl(_computerfundamentalsfirst)) {

        throw      Exception('Could      not      launch
$_computerfundamentalsfirst');

    }

Future<void> __flowchartsfirst() async {
    if (!await launchUrl(_flowchartsfirst)) {

        throw      Exception('Could      not      launch
$_flowchartsfirst');

    }

Future<void> __flowchartssecond() async {
    if (!await launchUrl(_flowchartssecond)) {

        throw      Exception('Could      not      launch
$_flowchartssecond');}

    @override

Widget build(BuildContext context) {
```

```
return Scaffold(  
    appBar: AppBar(  
        backgroundColor: kPrimaryColor,  
        leading: IconButton(  
            onPressed: () {  
                Navigator.pushNamed(context, Nav.id);  
            },  
            icon: const Icon(  
                Icons.arrow_back,  
                color: Colors.white,),),  
        elevation: 0.0,  
        title: const Text(  
            "Learning Kids",  
            style: TextStyle(  
                color: Colors.white,),,),),  
    body: ListView.custom(  
        childrenDelegate: SliverChildListDelegate(  
            [
```

```
const SizedBox(  
    height: 10,  
,  
CustomListTile(  
    onTap: __programmingkidsfirst,  
    img:    'assets/images/content_of_stacks/kids  
link 1.png',  
    title: 'NPStation',  
    subtitle: 'Programming Courses For Kids -  
English',  
,  
const SizedBox(  
    height: 10,  
,  
CustomListTile(  
    onTap: __programmingkidssecond,  
    img:    'assets/images/content_of_stacks/kids  
link 2.png',  
    title: 'coding for kids',
```

```
        subtitle: 'Programming Courses For Kids - Arabic',  
    ),  
    const SizedBox(  
        height: 10,  
    ),  
    CustomListTile(  
        onTap: __computerbasicsfirst,  
        img:    'assets/images/content_of_stacks/kids link 3.png',  
        title: 'LearnFree',  
        subtitle: 'Computer Basics - English',  
    ),  
    const SizedBox(  
        height: 10,  
    ),  
    CustomListTile(  
        onTap: __computerbasicssecond,  
        img:    'assets/images/content_of_stacks/kids link 4.png',
```

```
        title: 'Learn Coding',  
        subtitle: 'Computer Basics - English',  
    ),  
    const SizedBox(  
        height: 10,  
    ),  
    CustomListTile(  
        onTap: __computerfundamentalsfirst,  
        img:    'assets/images/content_of_stacks/kids  
link 5.png',  
        title: 'Make It Easy Education',  
        subtitle: 'Computer Fundamentals - English',  
    ),  
    const SizedBox(  
        height: 10,  
    ),  
    CustomListTile(  
        onTap: __flowchartsfirst,
```

```
        img:    'assets/images/content_of_stacks/kids  
link 6.png',  
  
        title: 'Bijoyan Das',  
  
        subtitle: 'Flowcharts - English',  
  
,  
  
const SizedBox(  
  
        height: 10,  
  
,  
  
CustomListTile(  
  
        onTap: __flowchartssecond,  
  
        img:    'assets/images/content_of_stacks/kids  
link 7.png',  
  
        title: 'Wondershare Edraw',  
  
        subtitle: 'Flowcharts - English',  
  
,  
  
const SizedBox(height: 10,],],),,);}}
```

5.9.quizzes page

```
import 'package:flutter/material.dart';

import 'package:roadx/constants.dart';

import
'package:roadx/pages_of_quizzes/quiz_of_android_ap
plication.dart';

import
'package:roadx/pages_of_quizzes/quiz_of_artificial_int
elligence.dart';

import
'package:roadx/pages_of_quizzes/quiz_of_backend.d
art';

import
'package:roadx/pages_of_quizzes/quiz_of_cross_platf
orm.dart';

import
'package:roadx/pages_of_quizzes/quiz_of_cyber_secu
rity.dart';

import
'package:roadx/pages_of_quizzes/quiz_of_frontend.da
rt';

import
'package:roadx/pages_of_quizzes/quiz_of_network.da
rt';
```

```
import 'package:roadx/pages_of_quizzes/quiz_of_ui_ux_design.dart';

import 'package:roadx/pages_of_quizzes/quiz_of_wordpress.dart';

import 'package:roadx/widgets/nav.dart';

class Quizzes extends StatelessWidget {

    const Quizzes({super.key});

    static String id = 'Quizzes';

    @override

    Widget build(BuildContext context) {

        final List<String> titles = [

            "Frontend",

            "Backend",

            "Wordpress",

            "Android Application",

            "Cross Platform",

            "Network",
        ];
    }
}
```

```
"Cyber Security",  
"Artificial Intelligence",  
"UI/UX Design",  
];  
  
final List<Widget> navigationTargets = [  
    const QuizOfFrontend(),  
    const QuizOfBackend(),  
    const QuizOfWordpress(),  
    const QuizOfAndroidApplication(),  
    const QuizOfCrossPlatform(),  
    const QuizOfNetwork(),  
    const QuizOfCyberSecurity(),  
    const QuizOfArtificialIntelligence(),  
    const QuizOfUIUXDesign(),  
];  
  
final List<String> images = [  
    'assets/images/resources/frontend.jpg',  
    'assets/images/resources/backend.png',
```

```
'assets/images/resources/wordpress.png',  
'assets/images/resources/android.jpg',  
'assets/images/resources/cross_platform.jpg',  
'assets/images/resources/network.png',  
'assets/images/resources/cyber_security.jpg',  
'assets/images/resources/artificial-intellegence.jpg',  
'assets/images/resources/uii_ux_design.png'],  
  
return Scaffold(  
  
    appBar: AppBar(  
  
        backgroundColor: kPrimaryColor,  
        leading: IconButton(  
  
            onPressed: () {  
  
                Navigator.pushNamed(context, Nav.id);},  
            icon: const Icon(Icons.arrow_back,  
            color: Colors.white,),),  
            elevation: 0.0,  
            title: const Text(  
                "Quizzes",
```

```
        style: TextStyle(color: Colors.white,)),),  
  
body: GridView.count(  
    crossAxisCount: 1,  
  
    children: List.generate(titles.length, (index) {  
  
        return InkWell(  
  
            onTap: () {  
  
                Navigator.push(  
                    context,  
                    MaterialPageRoute(  
                        builder: (context) =>  
  
                            navigationTargets[index %  
navigationTargets.length],  
  
                    );  
  
            },  
  
            child: Container(  
                width: 500,  
  
                margin: const EdgeInsets.all(4.0),  
  
                child: GridTile(  
  
                    footer: Container(  
                       
```

```
padding: const EdgeInsets.all(8.0),  
color: Colors.black.withOpacity(0.5),  
child: Text(  
titles[index],  
style: const TextStyle(color: Colors.white),  
textAlign: TextAlign.center,  
),  
,  
child: Image.asset(  
images[index % images.length],  
fit: BoxFit.cover,  
),  
,  
,  
);  
},  
),  
); })
```

Machine learning

The following libraries are important at the beginning of the code :

- TensorFlow is an open-source framework specialized in developing artificial intelligence and machine learning models used for building and training deep learning models.
- nltk: A natural language processing library that is used for tokenization, stemming, and lemmatization of words.
- pickle: A module for serializing and deserializing Python objects.
- Numpy: A scientific computing library that is used for working with arrays and matrices.
- keras: An open-source software library for building and training the neural network.

```
import nltk
nltk.download('punkt')

✓ 2.0s
```

The code you provided shows how to use the Natural Language Toolkit (NLTK) library in Python for a specific task.

```
import nltk
```

This line imports the NLTK library. NLTK provides functionalities for various Natural Language Processing (NLP) tasks.

```
nltk.download('punkt')
```

This line downloads the "punkt" data package from NLTK. The "punkt" package is specifically used for sentence tokenization. Sentence tokenization is the process of splitting a text into individual sentences.

```
import nltk
from nltk.stem import WordNetLemmatizer
lemmatizer = WordNetLemmatizer()
import json
import pickle
```

✓ 0.0s

The code you provided seems to be setting up for some natural language processing (NLP) tasks. Here's a breakdown of what each line does:

`import nltk`: This line imports the Natural Language Toolkit (NLTK) library, which provides functionalities for various NLP tasks like tokenization, stemming, lemmatization, and more.

from nltk.stem import WordNetLemmatizer: This line imports the WordNetLemmatizer class specifically from the NLTK library. This class is used to convert words to their base forms (lemmas).

lemmatizer = WordNetLemmatizer(): This line creates an instance of the WordNetLemmatizer class and assigns it to the variable lemmatizer. This allows you to use the lemmatizer functions later in your code.

import json: This line imports the json library, which is used for working with JSON data format. This might be used for loading or saving data in JSON format.

import pickle: This line imports the pickle library, which is used for serializing and deserializing Python objects. This allows you to save and load complex data structures like models or pre-processed data.

Overall, this code snippet sets up the environment for working with NLP tasks. The specific functionalities used later will depend on the rest of your code.

```
words=[]
classes = []
documents = []
ignore_words = ['?', '!']
data_file = open('data.json').read()
intents = json.loads(data_file)
```

✓ 0.0s

The code you provided continues building the NLP environment, specifically focusing on preparing data for intent classification. Here's what each line does:

`words=[]`: This line initializes an empty list called `words`. This list will likely be used to store individual words extracted from the data.

`classes = []`: This line initializes another empty list called `classes`. This list will likely be used to store the different classes or categories (intents) present in your data.

`documents = []`: This line initializes yet another empty list called `documents`. This list will probably hold the actual data points, potentially consisting of text and corresponding intent labels.

`ignore_words = ['?', '!']`: This line defines a list called `ignore_words` containing punctuation marks like question mark and exclamation mark. These characters might not be useful for classification and will likely be filtered out during data processing.

`data_file = open('data.json').read()`: This line opens a file named "data.json" in read mode and assigns the entire content of the file to the variable `data_file`.

`intents = json.loads(data_file)`: This line assumes the content of "data.json" is in JSON format. It uses the `json.loads` function to convert the loaded data (stored in `data_file`) into a Python dictionary and assigns it to the variable `intents`. This dictionary likely holds information about your intents and the corresponding training data.

```
# Create our training data
train_x = [] # List to store patterns
train_y = [] # List to store intents

output_empty = [0] * len(classes)
```

train_x = This line creates an empty list named train_x. This list will eventually hold the features (inputs) for your machine learning model. In this case, each element in train_x will represent a text pattern (sentence or phrase) used for training.

train_y = Similarly, this line creates another empty list named train_y. This list will hold the target labels (outputs) for your model. Each element in train_y will correspond to an element in train_x and indicate the intended category (class) for that particular text pattern.

output_empty = [0] * len(classes): This line creates a list called output_empty with a length equal to the number of classes (intents) you have in your dataset. It fills the list with zeros, essentially creating a one-hot encoded vector template. This template will be used later to represent the intent for each training example in a one-hot encoded manner.

```
for doc in documents:
    bag = []
    pattern_words = doc[0]
    pattern_words = [lemmatizer.lemmatize(word.lower()) for word in pattern_words]

    for w in words:
        bag.append(1) if w in pattern_words else bag.append(0)

    output_row = list(output_empty)
    output_row[classes.index(doc[1])] = 1

    train_x.append(bag)
    train_y.append(output_row)
```

`train_x: [] train_y: [] output_empty: [0 ,..., 0 ,0] (length = number of classes)`

Process Each Document:

- Bag-of-Words Representation: Extract, lowercase & lemmatize text pattern. Loop vocabulary, add 1 if word exists, else .0

- One-Hot Encoded Target Label: Extract, find index, set corresponding one-hot encoded vector element to .1

Store Processed Data: Add bag-of-words to `train_x`, one-hot encoded label to `train_y`.

Outcome: `train_x` & `train_y`: Processed training data. Each `train_x` element: Text pattern, corresponding `train_y`: Intended class label.

```
# Convert train_x and train_y to np.array
train_x = np.array(train_x)
train_y = np.array(train_y)

print("Training data created")
0.0s
```

1. Convert Lists to NumPy Arrays:

- `train_x = np.array(train_x)`
- `train_y = np.array(train_y)`

2. Print Confirmation:

```
- print("Training data created")
```

Overall Summary:

- Converts processed training data to NumPy arrays.
- Enables efficient operations for machine learning algorithms.
- Confirmation message printed.

```
for intent in intents['intents']:
    for pattern in intent['patterns']:
        # tokenize each word
        w = nltk.word_tokenize(pattern)
        words.extend(w)
        # add documents in the corpus
        documents.append((w, intent['tag']))
        # add to our classes list
        if intent['tag'] not in classes:
            classes.append(intent['tag'])
```

The following code we use the Natural Language Toolkit (nltk) library to tokenize patterns and build a vocabulary of words. -The script reads a JSON file containing a list of intents and their corresponding patterns, then iterates through each pattern and tokenizes the words in the pattern using the `nltk.word_tokenize()` function. -The script then extends the list 'words' with the tokenized words in the pattern and appends a tuple of the tokenized words and the intent tag to the 'documents' list.

The 'documents' list is used to build the training data for a machine learning model that can classify new patterns.

-The 'classes' list is used to keep track of the unique intent tags in the 'intents' list. If a new intent tag is encountered while processing patterns, it is added to the

'classes' list.

-The 'ignore_words' list is a list of characters that should be ignored when

tokenizing patterns.

Overall, the script is used to prepare the data for training a machine learning

model to classify new patterns into one of the existing intent tags. -The resulting model can be used to build a chatbot that can understand and respond to user input.

```
# lemmatize and lower each word and remove duplicates
words = [lemmatizer.lemmatize(w.lower()) for w in words if w not in ignore_words]
words = sorted(list(set(words)))
# sort classes
classes = sorted(list(set(classes)))
print (len(documents), "documents")
# classes = intents
print (len(classes), "classes", classes)
# words = all words, vocabulary
print (len(words), "unique lemmatized words", words)
```

This code is a continuation of the previous code. It processes the extracted

patterns and intents from data.json and creates a list of unique words and classes. -First, it uses the WordNetLemmatizer from the nltk library to lemmatize each word in the words list.

Lemmatization is the process of reducing a word to its base

or root form, which helps in standardizing the words and reducing the total number of unique words.

It then removes any words in the ignore_words list. -Next, it sorts the words list and removes any duplicates by creating a set and converting it back to a list.

This creates a vocabulary of unique words. Then, it sorts the classes list and removes any duplicates to get the unique intents. -Finally, it creates a list of tuples called documents, which consists of tokenized

words in each pattern and the associated intent tag. -The length of documents, classes, and words are printed to the console.

```
pickle.dump(words, open('texts.pkl', 'wb'))
pickle.dump(classes, open('labels.pkl', 'wb'))
```

This code saves the processed words and classes variables using the pickle module to binary files named texts.pkl and labels.pkl respectively. -This allows for the faster loading of these variables in future executions of the program, rather than having to repeatedly process the raw data from scratch. - pickle.dump() method is used to serialize the words and classes Python objects to a byte stream, which can be written to disk. The wb parameter specifies that the file should be opened in write-binary mode.

```
training = []
output_empty = [0] * len(classes)

for doc in documents:
    bag = []
    pattern_words = doc[0]
    pattern_words = [lemmatizer.lemmatize(word.lower()) for word in pattern_words]
    for w in words:
        bag.append(1) if w in pattern_words else bag.append(0)

    output_row = list(output_empty)
    output_row[classes.index(doc[1])] = 1
    training.append([bag, output_row])
```

this code is responsible for creating the training data for the chatbot. -The training variable is initialized as an empty list, which will hold the training data. The output_empty variable is also initialized as a list of zeros with a length equal to the number of unique classes. -Then, the documents list is iterated over. Each doc contains a tuple of (pattern_words, tag). - The pattern_words are tokenized and lemmatized. A bag list is initialized as an empty list. Then, each word in the words list (the entire vocabulary of the chatbot)

it checked against the pattern_words. If the word is present in the pattern_words, a 1 is appended to the bag list, otherwise a 0 is appended. This process creates the bag-of-words representation of each pattern_words. -The output_row list is created by copying the output_empty list and replacing the element at the index of the tag with a 1. This is necessary to create the correct output for each training pattern. -Finally, the bag and output_row are appended to the training list as a single training example.

This process is repeated for every doc in documents. The resulting training list will be used to train the chatbot's neural network.

```
random.shuffle(training)
# Separate bag of words and output row into two separate lists
bags_of_words = [item[0] for item in training]
output_rows = [item[1] for item in training]
```

The code you provided shuffles your training data and separates it into two distinct lists: bags_of_words and output_rows. Here's a breakdown of what each line does:

Shuffling Training Data:

`random.shuffle(training)`: This line shuffles the elements within the training list using the `random.shuffle` function. Shuffling helps reduce biases that might be present in the order your data was collected and can improve the generalization of your model during training.

Separating Features and Labels:

List Comprehension: The following lines use list comprehension to create two new lists:

`bags_of_words = [item[0] for item in training]`: This line iterates through each item in the training list. Assuming each item is a tuple containing a bag-of-words representation (likely at index 0) and the corresponding output row (encoded label, likely at index 1), this line extracts only the first element (bag-of-words) and adds it to the `bags_of_words` list.

`output_rows = [item[1] for item in training]`: Similarly, this line extracts only the second element (output row/encoded label) from each item in the training list and adds it to the `output_rows` list.

Overall, this code snippet shuffles your training data and separates the features (bag-of-words representations) from the labels (encoded class labels) into distinct lists for potentially easier handling and usage during the training process of your intent classification model.

```
# shuffle the data
indices = np.arange(train_x.shape[0])
np.random.shuffle(indices)
train_x = train_x[indices]
train_y = train_y[indices]
```

This code block shuffles your training data (`train_x` and `train_y`) to ensure the model doesn't learn biases based on the order the data was presented. Here's a step-by-step explanation:

Creating Shuffling Indices:

`indices = np.arange(train_x.shape[0]):` This line creates a NumPy array named `indices` containing a sequence of numbers starting from 0 and ending at the length of `train_x` (number of training examples) minus 1. Essentially, it creates an index for each element in your training data.

Shuffling the Indices:

`np.random.shuffle(indices):` This line shuffles the elements within the `indices` array using the `np.random.shuffle` function from NumPy. Shuffling the indices ensures that the order in which the training data is accessed is randomized.

Reordering Training Data:

`train_x = train_x[indices]`: This line rearranges the elements in `train_x` (the bag-of-words features) based on the shuffled indices. It uses those shuffled indices to select elements from `train_x` and put them in a new order within `train_x` itself.

`train_y = train_y[indices]`: Similarly, this line rearranges the elements in `train_y` (the one-hot encoded labels) based on the same shuffled indices. This ensures that the shuffling applied to the features (`train_x`) is also applied to the corresponding labels (`train_y`) so they stay matched.

Benefits of Shuffling:

Reduced Bias: By shuffling the data, the model is less likely to learn patterns specific to the order the data was presented. This helps the model generalize better to unseen data.

Improved Performance: Shuffling can sometimes lead to a more robust training process and potentially improve the overall performance of your intent classification model.

Overall, this code snippet shuffles your training data using NumPy's functionalities. Shuffling helps mitigate biases and potentially improves the training process of your model.

```
# Create model
model = Sequential()
model.add(Dense(128, input_shape=(len(train_x[0]),), activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(64, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(len(classes), activation='softmax')) # Changed output units to len(classes)
# Compile model
model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])

# Fit the model
model.fit(train_x, train_y, epochs=200, batch_size=32, verbose=1, validation_split=0.2)

# Save the model
model.save('model.h5')

print("Model created and saved.")
```

This code block defines, compiles, trains, and saves your intent classification model. Here's a breakdown of each step:

1. Model Definition (Sequential Model):

model = Sequential(): This line creates a sequential model, which is a common type of neural network architecture where layers are added sequentially.

2. Adding Hidden Layers:

model.add(Dense(128,input_shape=(len(train_x[0]),), activation='relu')): This line adds the first hidden layer to the model.

Dense(128): This specifies a dense layer with 128 neurons. Dense layers are fully connected, meaning each neuron in the layer connects to all neurons in the previous layer.

`input_shape=(len(train_x[0]),):` This defines the input shape for the model, which is based on the length of the first element (likely the bag-of-words representation) in your `train_x` list. This ensures the model expects inputs of that specific size.

`activation='relu':` This defines the activation function for the layer. Here, 'relu' (Rectified Linear Unit) is used, which introduces non-linearity to the model and helps it learn complex patterns.

Similar lines are used to add two more hidden layers, each with 64 neurons and 'relu' activation.

3. Dropout Layers:

`model.add(Dropout(0.5)):` These lines (one after each hidden layer) add dropout layers to the model. Dropout randomly drops a certain percentage (here, 50%) of neurons during training. This helps prevent overfitting by reducing the co-dependency between neurons and encouraging them to learn more robust features.

4. Output Layer:

`model.add(Dense(len(classes), activation='softmax')):` This line adds the output layer of the model.

`Dense(len(classes)):` This specifies a dense layer with a number of neurons equal to the number of classes (intents) you have in your data. This is because the model needs to predict the probability of each class for a given input.

`activation='softmax'`: This defines the activation function for the output layer. The 'softmax' function is commonly used for multi-class classification problems. It squashes the outputs of the layer into a probability distribution between 0 and 1, where the sum of all outputs for a single input is 1. The highest output value indicates the predicted class.

5. Model Compilation:

`model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])`: This line compiles the model. Here's what each part does:

`loss='categorical_crossentropy'`: This defines the loss function used to measure the difference between the model's predictions and the true labels. Categorical crossentropy is commonly used for multi-class classification problems.

`optimizer='adam'`: This specifies the optimization algorithm used to train the model. Adam is a popular optimizer that efficiently updates the weights of the model during training.

`metrics=['accuracy']`: This defines the metrics used to track the model's performance during training and validation. Here, 'accuracy' is used to measure the percentage of correct predictions.

6. Model Training:

`model.fit(train_x, train_y, epochs=200, batch_size=32, verbose=1, validation_split=0.2)`: This line trains the model. Here's a breakdown:

`train_x`: This is the training data (bag-of-words features) you prepared earlier.

`train_y`: These are the corresponding labels (one-hot encoded) for the training data.

`epochs=200`: This specifies the number of times the entire training data will be passed through the model for training. 200 is a common starting point, but you might need to adjust it based on your data and performance.

`batch_size=32`: This defines the batch size, which is the number of training examples used to update the model's weights in each iteration. A batch size of 32 is a common choice.

`verbose=1`: This sets the verbosity level during training. Here, 1 indicates printing some information after each epoch.

`validation_split=0.2`: This splits 20% of your training data for validation. The model will use this split to monitor its performance on unseen data during training and help prevent overfitting.

7. Model Saving:

`model.save('model.h5')`

```
model = load_model('model.h5')
```

The model is stored in a file called model.h5. This file contains the architecture and weights of the trained model.

```
intents = json.loads(open('data.json').read())
words = pickle.load(open('texts.pkl','rb'))
classes = pickle.load(open('labels.pkl','rb'))
```

Loading the Intents and Responses

The following lines of code load the intents and responses from a JSON file:

The intents and responses are stored in a file called data.json. The file contains a list of intents, where each intent has a tag, a list of patterns, and a list of responses. Loading the Bag of Words

The following lines of code load the bag of words from a pickle file:

-The bag of words is stored in two files: texts.pkl and labels.pkl. -The texts.pkl file contains a list of all the words that are used in the patterns of the intents. -The labels.pkl file contains a list of the tags for each intent.

```
def clean_up_sentence(sentence):
    # Tokenize the pattern - split words into an array
    sentence_words = nltk.word_tokenize(sentence)
    # Lemmatize each word - create a short form for the word
    sentence_words = [lemmatizer.lemmatize(word.lower()) for word in sentence_words]
    return sentence_words
```

-The `clean_up_sentence()` function takes a sentence as input and performs tokenization and lemmatization to normalize the sentence.

-The output is a list of lemmatized words.

```
def bow(sentence, words, show_details=True):
    # Tokenize the pattern
    sentence_words = clean_up_sentence(sentence)
    # Bag of words - matrix of N words, vocabulary matrix
    bag = [0] * len(words)
    for s in sentence_words:
        for i, w in enumerate(words):
            if w == s:
                # Assign 1 if the current word is in the vocabulary position
                bag[i] = 1
                if show_details:
                    print(f"found in bag: {w}")
    return np.array(bag)
```

-The bow() function creates a bag of words vector for the given sentence. -The function takes the input sentence, a list of words, and a Boolean flag that determines whether to display details or not. -It creates a vector of the same length as the list of words and sets the values to 0. -For each word in the input sentence, it checks whether the word is in the list of words. If it is, it sets the corresponding value in the bag vector to 1.

```
def predict_class(sentence, model):
    # Filter out predictions below a threshold
    p = bow(sentence, words, show_details=False)
    res = model.predict(np.array([p]))[0]
    ERROR_THRESHOLD = 0.25
    results = [[i, r] for i, r in enumerate(res) if r > ERROR_THRESHOLD]
    # Sort by the strength of probability
    results.sort(key=lambda x: x[1], reverse=True)
    return_list = []
    for r in results:
        return_list.append({"intent": classes[r[0]], "probability": str(r[1])})
    return return_list
```

The predict_class() function takes a sentence and a model as input and returns a list of intents and their probabilities based on the input sentence. -It uses the bow() function to create a bag of words vector for the input sentence,

then uses the pre-trained model to predict the intent based on the vector. -The output is a list of intents with their corresponding probabilities

```
def getResponse(ints, intents_json):
    tag = ints[0]['intent']
    list_of_intents = intents_json['intents']
    for intent in list_of_intents:
        if intent['tag'] == tag:
            result = random.choice(intent['responses'])
            break
    return result
```

The `getResponse()` function takes a list of intents and the `intents.json` file as input and returns a response for the predicted intent. -It uses the tag field of the predicted intent to match it with an intent in the JSON file and selects a random response from the responses field.

```
def chatbot_response(msg):
    ints = predict_class(msg, model)
    res = getResponse(ints, intents)
    return res
```

The `chatbot_response()` function takes user input as input and returns a response

based on the predicted intent. It uses the `predict_class()` and `getResponse()`

functions to predict the intent and generate a response.

```
from flask import Flask, render_template, request

app = Flask(__name__)
app.static_folder = 'static'

@app.route("/")
def home():
    return render_template("index.html")

@app.route("/get")
def get_bot_response():
    userText = request.args.get('msg')
    return chatbot_response(userText)

if __name__ == "__main__":
    app.run(port=8081, host='0.0.0.0')
```

-The Flask import is used to create a web application. It provides methods for creating a web server and handling HTTP requests. -The home() function returns the home page of the web application. -The get_bot_response() function handles HTTP GET requests from the client and returns a response generated by the chatbot.

Data set

This code represents a sample implementation of a natural language processing (NLP) chatbot. It defines a list of intents, each with a set of patterns that the bot can recognize and respond to. The example intent in this code is the "greeting" intent.

Here is a breakdown of the different fields in the "greeting" intent:

"tag": This is a unique identifier for the intent. It is used to

match user input with the appropriate response.

"patterns": This is a list of phrases or keywords that the bot

can recognize as greetings. These patterns can be modified or

expanded to improve the bot's ability to recognize greetings

in different contexts.

"responses": This is a list of possible responses that the bot

can give when it recognizes a greeting. These responses can

also be modified or expanded to provide more variety in the

bot's output.

"context": This field is used to specify any context that may

be relevant to the intent. In this case, the "greeting" intent

does not require any context. To add more intents to the chatbot, you can follow the same format as

the "greeting" intent. Simply define a new object with a unique "tag"

identifier, a set of patterns, a list of possible responses, and any relevant context.

Overall, this code is a good starting point for building a simple chatbot with NLP capabilities. However, it is important to note that this code alone is not sufficient to create a fully functional chatbot. Additional

components such as language processing, entity recognition, and a response generation engine may be necessary to create a more robust and accurate chatbot.

Chapter 6

Important packages in flutter :

1-adaptive_theme: ^3.6.0

Description: This package provides an easy way to switch between light and dark themes dynamically in a Flutter app. It allows saving the user's theme preference and provides a simple API to toggle and manage themes.

Use case: Ideal for apps that need to support both light and dark themes based on user preferences.

2-animated_text_kit: ^4.2.2

Description: A package that offers a variety of animated text effects, such as rotating, scaling, and fading text animations. It helps to create engaging text animations with ease.

Use case: Perfect for apps that require eye-catching text animations for UI enhancements or storytelling.

3-carousel_slider: ^4.2.1

Description: A customizable carousel slider widget that allows creating image sliders, banners, or any kind of carousel display. It supports various customizations and configurations.

Use case: Suitable for displaying image galleries, promotional banners, or featured content in a swipeable format.

4-cupertino_icons: ^1.0.8

Description: This package contains the Cupertino Icons, the icon set used in Apple's iOS. It provides a wide range of icons following the iOS design guidelines.

Use case: Essential for apps targeting iOS, ensuring consistency with iOS native app icons.

5-dio: ^5.4.3+1

Description: Dio is a powerful HTTP client for Dart, which supports global configuration, interceptors, FormData, file uploading, and more. It simplifies making network requests and handling responses.

Use case: Ideal for apps that need to interact with REST APIs or perform network operations with advanced features like request/response interception.

6-google_nav_bar: ^5.0.6

Description: A modern and customizable navigation bar inspired by Google's Bottom Navigation Bar, with support for various styles and animations.

Use case: Useful for implementing intuitive and attractive bottom navigation bars in Flutter apps.

7-http: ^1.2.1

Description: A simpler HTTP client for making network requests. It supports basic features like GET, POST, PUT, DELETE methods and simplifies HTTP interactions.

Use case: Suitable for straightforward HTTP requests without the need for advanced features provided by Dio.

8-material_symbols_icons: ^4.2741.0

Description: This package provides access to Google's Material Symbols, an extensive set of icons designed under the Material Design guidelines.

Use case: Great for apps that follow Material Design principles and need a wide variety of icons.

9-timeline_tile: ^2.0.0

Description: A widget for creating beautiful and customizable timelines in Flutter apps. It helps in displaying chronological information, steps, or events in a vertical or horizontal layout.

Use case: Ideal for apps that need to present progress steps, history, or any sequential data visually.

10-typing_animation: ^0.0.1

Description: This package offers a typing animation effect for text, mimicking the appearance of typing one character at a time.

Use case: Perfect for chat interfaces, messaging apps, or any scenario where a typing indicator is needed.

11-url_launcher: ^6.2.6

Description: A Flutter plugin for launching URLs in the mobile platform. It supports opening web pages, emails, phone calls, and SMS.

Use case: Essential for apps that need to open external links, dial phone numbers, send SMS, or emails from within the app.

12-vertical_card_page: ^1.6.2

Description: A widget for creating a vertical card pager with swipeable cards. It provides an engaging way to navigate through content cards vertically.

Use case: Ideal for apps that display content in a card-based layout with vertical swiping, such as news apps or catalogs.

Chapter 7

Conclusion

Our website and application provide an importance of learning programming:

Learning programming is not just about writing code; it's about developing a skill set that empowers you to understand, interact with, and shape the digital world. Whether for career advancement, personal growth, or societal contribution, programming offers invaluable benefits that are increasingly essential in the modern world.

- Career Opportunities
- Problem-Solving Skills
- Digital Literacy
- Interdisciplinary Applications
- Future-Proofing
- Entrepreneurship
- Community and Collaboration
- Education and Empowerment

Chatbots play a crucial role in enhancing customer service, reducing operational costs, improving user engagement, and streamlining business processes. Their ability to provide instant, personalized, and consistent interactions makes them invaluable in today's digital landscape. As technology continues to evolve, the capabilities and applications of chatbots will only expand, further solidifying their importance in various sectors.

Chapter 8

Future Work

Application Development Steps

First Step: We will update the design to be more engaging and user-friendly based on user feedback.

Second Step: Enhance the video quality and streaming performance to ensure a smooth learning experience for all users.

Third Step: Expand the educational videos, covering a wider range of topics and subjects for both kids and adults.

Fourth Step: Integrate advanced chatbot functionalities to provide more personalized assistance and interactive learning experiences

Fifth Step: We provide templates, extra knowledge, and documentation for educated people.

Sixth Step: Introduce more interactive learning modules to make the educational process more interactive and enjoyable, particularly for kids.

Seventh Step: Implement AI-driven recommendations for videos and resources based on individual user progress and preferences.

Eighth Step: FAQs to support users better, and user's feedback.

Ninth Step: Allowing users to learn from educators for real-time learning, and support different sources for users in journey of learning.

Completed Work

The application was completed as initially planned, encompassing several key features aimed at improving the learning experience:

- A.** A well-designed logo and user interface were developed to reflect the educational nature of the program.
- B.** The main page includes all essential operations, offering a seamless user experience.
- C.** Key features implemented in the application:
 - **Difference sources of video:** A diverse collection of educational videos covering various subjects for different age groups.
 - **Interactive Chatbot:** Assists users with learning queries, provides recommendations, and offers support.
 - **Documentation:** Comprehensive guides, tutorials, and FAQs available for users to get the most out of the application.

Chapter 9

References

<https://www.pinterest.com/>

<https://www.pexels.com/>

<https://www.freepik.com/>

<https://color.adobe.com/create/color-wheel>

<https://www.sciencedirect.com/science/article/pii/S1110016820304907>

<https://www.quora.com/How-can-I-convert-a-research-paper-to-a-deep-learning-model>

<https://www.sciencedirect.com/science/article/pii/S1110016820304907>

Thanks