e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:05/May-2023 Impact Factor- 7.868 www.irjmets.com

REACT NATIVE FOR NAVIGATION APPLICATION DEVELOPMENT

Mohammed sarfaraz*1, Naveen Naik*2

*1,2Student, Department of MCA, NMAM Institute of technology, Nitte , Udupi, Karnataka, India

DOI : https://www.doi.org/10.56726/IRJMETS38937

ABSTRACT

React Native is a powerful framework to build navigation apps that offers various advantages over traditional native app development. React Native allows developers to create cross-platform applications with a single codebase, saving time and resources during the development process. React Native also offers quick and smooth performance by rendering components using native APIs, and it has a huge and active developer community that contributes to its development and shares their knowledge and expertise. Furthermore, React Native has several pre-built navigation libraries that can assist developers in rapidly and effectively developing navigation applications. Overall, React Native is an excellent choice for building navigation applications, and it offers an extensive collection of tools and frameworks to aid developers in their efforts.

Keywords: cross-platform, performance, navigation libraries, single codebase

I. INTRODUCTION

In recent years, mobile navigation apps have grown in popularity, and due to the rise in cross-platform software development, many developers have turned to frameworks such React Native to construct these apps. React Native is a robust framework that enables developers to create mobile applications with JavaScript and React that can be delivered to Android as well as iOS. This method has various advantages to traditional native app creation, including one code base, shorter periods of development, and the potential to tap into a big and active developer community. In addition to these advantages, React Native provides various already constructed navigational libraries that can assist developers in quickly and efficiently creating navigation applications. These libraries include already-constructed elements and navigational patterns that developers may utilize to construct high-quality, user-friendly navigation experiences. In general, React Native provides a robust and versatile basis for constructing navigation apps, and its advantages make it a good choice for developers trying to create cross-platform programs which are efficient, adaptable, and simple to maintain.

II. METHODOLOGY

The methodology used in this research involves defining the requirements of the navigation application, designing the user interface of the application, developing the application using React Native, testing the application, and deploying the application to the appropriate app stores and marketplaces. The research also involves the use of debugging tools, code editors, and testing frameworks to ensure that the application is functional and meets the needs of the users.

Define the requirements.

The first step in building a navigation application using React Native is to define the requirements and functionality of the application. This includes identifying the key features, navigation patterns, and user flows required to meet the needs of the application.

Design the UI

After defining the requirements, developers can design the user interface of the application. This includes selecting an appropriate navigation library, designing the navigation structure, and creating visual components.

Develop the application.

Once the UI has been designed, developers can begin building the application using React Native. This involves writing code in JavaScript and leveraging the React Native framework to build UI components, navigation structures, and other application functionality.

Test the application. After developing the application, developers can test it to ensure that it functions correctly and meets the requirements. This includes testing the navigation flows, user interface components, and other key functionality to identify and resolve any issues.

III. MODELLING AND ANALYSIS

Frontend Design patterns

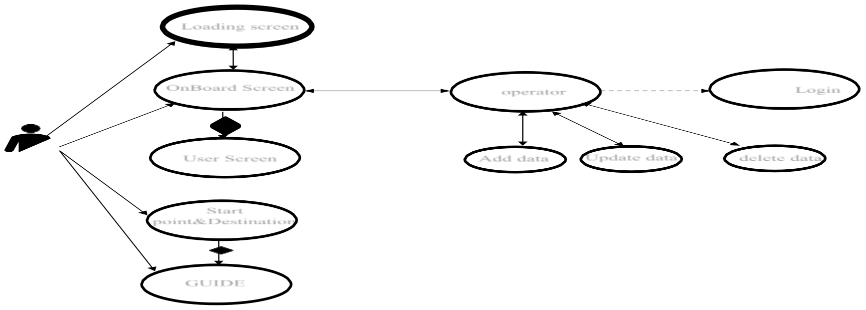
- Higher Order Component (HOC) Pattern - for adding data and functionality to components.
- Presentational and Container Component Pattern - for easier management and components.
- React Hooks Design Patterns - for building functional components with access to React features.

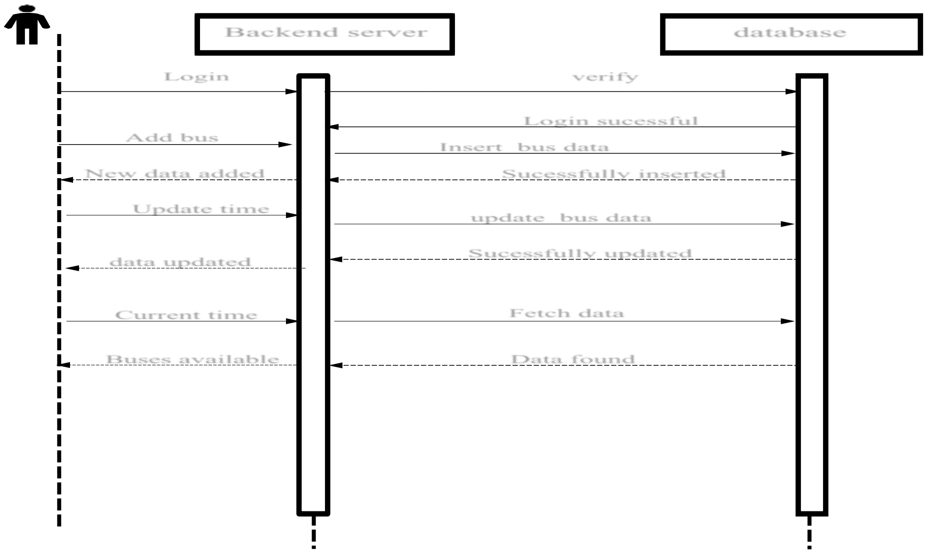
Backend Design Patterns

- Middleware Pattern - for performing tasks before and after a request/response cycle in express.
- Router Pattern - for handling different routes in Express applications.

Application

- Frontend was built using React Native Expo for building native mobile apps.
- Backend built using Node.js Express for fast and flexible web application development.
- Database powered by MongoDB for storing and retrieving expenditure data.
- User interface designed to be user-friendly and intuitive for farmers to manage their finances.





IV. TECH STACK

Table 1: Technologies available for this project

FRONT END	React Native, Redux
BACK-END Language	MongoDB, Node JS

React Native- React Native is a JavaScript framework that may be used to construct, design, and deploy complex user interfaces for mobile and online apps. It is a popular open-source programme used for constructing various types of Web, Android, and iOS applications. One of the most intriguing aspects of React Native is its ability to delivers contemporary web strategies to mobile without sacrificing much in terms of features or performance[1]. This framework is utilised in the project's front end. To use React Native in the development of various types of apps, one needs first have a basic understanding of HTML, CSS, JavaScript, NodeJS, and ReactJS. Organisations all over the world utilise React Native to develop multi-platform phone/tablet apps with a single code base [2].

Redux- Redux serves as a standalone library that is mostly used for application state management and can be used in conjunction with any other library or framework, such as React Native, React, and so on. Redux makes application debugging simple. Redux is also utilised on the computer side to render data. Redux is a state container with predictable behavior [3].

NodeJS- NodeJS is an environment used for running JavaScript code outside of the browser. It is an open-source cross-platform used for server-side application development and networking. It is fast and ideal for real-time applications. It is now a popular tool in nearly any type of project. It is a development framework.

high-performance, concurrent programmes that combine synchronous input/output with an event-driven programming architecture instead of the standard multithreading approach [4].

MongoDB- MongoDB is a free and open-source document database with high performance, high reliability, and automatic scaling [5]. MongoDB is currently the most used NoSQL database management system. When managing extremely large amounts of data, also known as big information, it becomes quite tough. Consider MongoDB before MySQL databases when it comes to tables. MongoDB is known for its excellent performance, high availability, and simple scalability.

V. RESULT

According to the findings of this study, React Native offers a powerful and flexible framework for developing navigation applications. The framework provides various advantages, such a single codebase, shorter phases of development, and the opportunity to harness a large and active developer community. React Native also includes a number of already constructed navigation modules that can assist developers in quickly and effectively creating navigation applications. These libraries include already-constructed elements and navigation principles that developers may utilise to construct high-quality, user-friendly navigation experiences.

Login

Test Case ID	User Event	User Input	Expected Output	Result
1	Screen (onLoad())	Loads theLoading Screen	Animated logo , Next button should appear after 2 seconds	Success
2	Email (onKeyPress())	User entersemail	Values are accepted	Success
3	Password (onKeyPress())	Operator enters password	Values are accepted	Success
4	Login (onClick())	Operator has entered invalid values	"Invalid email orpassword!"	Success
5	Login (onClick())	Operator has entered validvalues	Operator is navigatedto inventory screen of the app	Success

Home Screen

Test Case ID	User Event	User Input	Expected Output	Result
1	Page (onLoad())	Load the Home screen	Loads User Home Screen	Success
2	Refresh (onPullToRefresh())	Pull the screen to refresh	Updated Bus data	Success

3	Scroll (onScroll())	User scrolls the screen	Data cards arescrolled	Success

Detail Screen

Test Case ID	User Event	User Input	Expected Output	Result
1	Page (onLoad())	Load the detailsscreen	Bus timings shouldappear	Success
2	map image (onKeyPress())	On click on mapimage	Redirected to gogglemaps	Success

Admin screen:

Test Case ID	User Event	User Input	Expected Output	Result
1	Page (onLoad())	Load the new data listscreen	New data List must appear	Success
2	edit button (onKeyPress())	Operator changes the details of bus	Changes are accepted	Success
3	edit button (onKeyPress())	Delete card details	Selected card datamust disappear	Success

VI. CONCLUSION

- In conclusion, the development of a local bus guide app has the potential to significantly improve the public transportation experience for the local community.By providing real-time information, increasing accessibility, and fostering a sense of community, the app can help to make public transportation a more reliable andaccessible option for everyone.
- However, there are also several challenges associated with the development and implementation of the app, including data accuracy, user adoption, and privacy concerns. To overcome these challenges, it is important to carefully consider the needs and preferences of the local community, and to develop a user-friendly interface that is easy to navigate and provides all the information users need in one place.
- In terms of future enhancements, there are several areas where the local bus guide appcould be improved or expanded. For example, the app could be integrated with other transportation options, such as bike-share programs or ride-hailing services, to provide users with a more comprehensive range of transportation options. Additionally, the app could incorporate features such as trip planning and real-time traffic updates, or allow users to provide feedback on the bus service directly through the app.
- By continually evolving and improving the app, the local bus guide app can become an even more valuable resource for the local community, helping to make public transportation a more convenient, accessible, and reliable form of transportation.

VII. REFERENCES

[1] Smart Public Transportation: A Review of the State of the Art and Future Trends." Journal ofIntelligent Transportation Systems, vol. 23, no. 3, 2019, pp. 210-221.

[2] "Transportation App Usage: An Exploratory Study." Transportation Research Part C: Emerging Technologies, vol. 97, 2019, pp. 215-226.

[3] "The Impact of Real-Time Information on Public Transport Usage: A Systematic Review."Transport Reviews, vol. 38, no. 4, 2018, pp. 463-486.

[4] "Assessing the Effectiveness of Public Transportation Mobile Apps." Journal of PublicTransportation, vol. 19, no. 2, 2016, pp. 121-139.

[5] "The Use of Mobile Apps for Public Transportation: A Case Study in the City of Milan." Transportation Research Part A: Policy and Practice, vol. 106, 2018, pp. 249-260.