Learning Roadmap

Backend Development (Node.js)

1. **JavaScript Fundamentals:**

   - Learn the basics of JavaScript including variables, data types, functions, and control flow.

2. **Node.js Fundamentals:**

   - Understand the basics of Node.js including modules, file system, and event loop.

3. **Express.js Framework:**

   - Learn Express.js, a popular Node.js web application framework for building APIs and web applications.
   - Understand routing, middleware, and request handling in Express.js.

4. **RESTful API Development:**

   - Learn principles of RESTful API design.
   - Implement CRUD operations using Express.js and MongoDB/Mongoose (or any other database).

5. **Middleware:**

   - Understand middleware functions and their role in Express.js applications.
   - Implement custom middleware for authentication, error handling, etc.

6. **Database Interaction:**

   - Learn MongoDB, a NoSQL database commonly used with Node.js applications.
   - Understand how to perform CRUD operations and schema design in MongoDB.
   - Alternatively, learn SQL and relational databases if desired.

7. **Authentication and Authorization:**

   - Implement user authentication and authorization using middleware like Passport.js.
   - Learn about JSON Web Tokens (JWT) for token-based authentication.

8. **File Uploads:**

   - Learn how to handle file uploads in Node.js applications using libraries like Multer.
   - Implement file upload functionality in your project.

9. **Testing and Debugging:**

   - Learn about unit testing and integration testing for Node.js applications.
   - Explore debugging techniques for Node.js applications.

10. **Deployment:**

- Deploy your Node.js application to a hosting platform like Heroku or AWS.
- Configure environment variables and manage production environments.

Frontend Development (React.js)

1. **HTML and CSS:**

   - Refresh your knowledge of HTML and CSS.
   - Learn modern CSS techniques like Flexbox and Grid layout.

2. **JavaScript ES6+:**

   - Learn advanced JavaScript concepts like arrow functions, classes, and modules.

3. **React.js Fundamentals:**

   - Understand the core concepts of React.js including components, props, state, and JSX.
   - Learn about React hooks for managing state and side effects.

4. **Routing and State Management:**

   - Implement client-side routing using React Router.
   - Learn about state management libraries like Redux or React Context API.

5. **REST API Integration:**

   - Learn how to make HTTP requests to backend APIs using Fetch API or Axios.
   - Implement CRUD operations in your React application.

6. **Styling in React:**

   - Explore styling options in React including inline styles, CSS modules, and styled-components.

7. **Webpack and Babel:**

   - Understand the basics of Webpack and Babel for bundling and transpiling React applications.

8. **Testing React Applications:**

   - Learn about testing frameworks like Jest and testing utilities like React Testing Library.
   - Write unit tests and integration tests for React components.

9. **Optimization and Performance:**

   - Optimize your React application for performance by code splitting and lazy loading.
   - Implement caching strategies for optimizing data fetching.

10. **Deployment:**

    - Deploy your React application to a hosting platform like Netlify or Vercel.
    - Configure environment variables and manage production builds.

Additional Resources:

- Node.js Documentation: https://nodejs.org/en/docs/
- Express.js Documentation: https://expressjs.com/
- MongoDB Documentation: https://docs.mongodb.com/
- React.js Documentation: https://reactjs.org/docs/getting-started.html

```
.
├── project-server/
│   ├── README.md
│   ├── app.js
│   ├── bin/
│   │   └── www
│   ├── config/
│   │   ├── keywords.js
│   │   └── multerConfig.js
│   ├── controllers/
│   │   └── userController.js
│   ├── middleware/
│   │   └── authMiddleware.js
│   ├── models/
│   │   └── User.js
│   ├── node_modules/
│   ├── package-lock.json
│   ├── package.json
│   ├── public/
│   │   ├── imgss/
│   │   └── stylesheets/
│   ├── routes/
│   │   ├── auth.js
│   │   └── users.js
│   └── views/
│       ├── error.jade
│       ├── forbidden.html
│       ├── forbidden.jade
│       ├── index.jade
│       └── layout.jade
├── project-web/
│   ├── README.md
│   ├── build/
│   │   ├── asset-manifest.json
│   │   ├── favicon.ico
│   │   ├── favicon.png
│   │   ├── index.html
│   │   ├── manifest.json
│   │   ├── robots.txt
│   │   └── static/
│   ├── node_modules/
│   ├── package-lock.json
│   ├── package.json
│   ├── public/
│   │   ├── favicon.ico
│   │   ├── favicon.png
│   │   ├── index.html
```

```
│   │       ├── logo192.png
│   │       ├── logo512.png
│   │       ├── manifest.json
│   │       └── robots.txt
│   ├── src/
│   │   ├── AdminComponents/
│   │   ├── App.css
│   │   ├── App.js
│   │   ├── App.test.js
│   │   ├── ContributorComponents/
│   │   ├── assets/
│   │   ├── components/
│   │   ├── configs/
│   │   ├── contexts/
│   │   ├── index.css
│   │   ├── index.js
│   │   ├── logo.svg
│   │   ├── reportWebVitals.js
│   │   ├── setupTests.js
│   │   └── utils/
│   └── webpack.config.js
├── README.md
├── documents/
│   ├── cb-fonts.md
│   └── request.pdf
└── logos/
└── upload-karo/
```