Methodology for Dairify App

1. Tools and Technologies Used

- Frontend: React Native (Expo)

- Backend: Node.js with Express

- Database: MongoDB

- Development Environment: Visual Studio Code (VS Code)

- Libraries and Packages:

- axios (for API requests)

- express (for server creation)

- mongodb (for database interaction)

- cors (for enabling cross-origin requests)

- dotenv (for environment variables)

- mongoose (for MongoDB ORM)

2. Frontend Development (Mobile App)

- Initialized a new React Native (Expo) project using npx create-expo-app.

- Designed and structured screens such as:

- Login Screen

- Signup Screen

- Dashboard

- Settings

- Implemented axios to communicate with backend APIs.

- Managed user authentication, data fetching, and navigation between screens.

3. Backend Development (Server Side)

- Set up a Node.js project using npm init.

- Installed and configured necessary packages: express, cors, mongodb, mongoose, dotenv.

- Created RESTful APIs for user authentication, data management, and communication with the mobile app.

- Used Express for routing and middleware setup.

- Implemented security practices like environment variables for sensitive data (MongoDB URI, JWT secrets).

4. Database Management (MongoDB)

- Used MongoDB Atlas for cloud-based database hosting.

- Created database schemas and models using Mongoose:

- User Schema

- Session or Data Schema

- Performed CRUD operations (Create, Read, Update, Delete) for managing app data.

5. Integration and Testing

- Connected the frontend (React Native app) with backend APIs using axios.

- Performed testing of endpoints using Postman and ensured correct responses.

- Conducted testing on Android and iOS simulators to verify frontend-backend communication.

6. Deployment (Optional if applicable)

- (If applicable) Deployed backend server on services like Render, Vercel, or Heroku.

- Set up the app for production build using Expo's build services if needed.

Final Notes:

- Full project development and coding were done using Visual Studio Code.

- Proper Git version control was maintained during development.

- Focused on code reusability, modularity, and maintainability.