Learning Path for Building the React Frontend

Based on your project requirements and current knowledge level, here's a detailed phased learning plan to build the React frontend:

Phase 1: JavaScript Fundamentals (3-5 days)

Concepts to learn:

- ES6+ syntax (arrow functions, destructuring, spread/rest operators)
- Array methods (map, filter, reduce)
- Promises and async/await
- Modules (import/export)
- Fetch API or Axios for HTTP requests

Practice exercises:

- Create a simple app that fetches data from a public API
- Build a form with validation using vanilla JavaScript

Phase 2: React Basics (5-7 days)

Concepts to learn:

- JSX syntax
- Functional components vs class components (focus on functional)
- Props and component composition
- State management with useState hook
- Event handling
- Conditional rendering
- Lists and keys

Practice project:

- Build a simple todo application
- Create a user profile card component

Phase 3: Intermediate React (5-7 days)

Concepts to learn:

- useEffect hook for side effects
- Custom hooks
- Forms handling (controlled components)
- React Router for navigation

- Context API for state management
- Component lifecycle (focus on functional component equivalent)

Practice project:

- Build a multi-page application with routing
- Create a form with validation that sends data to an API

Phase 4: Advanced React for Your Project (7-10 days)

Concepts to learn:

- Authentication flow with JWT tokens
- Protected routes implementation
- Local storage for token persistence
- Error boundaries
- API integration patterns
- Loading states and error handling

Practice project:

- Implement a login/logout flow with mock authentication
- Create a dashboard that fetches and displays data from an API

Phase 5: UI Libraries and Styling (3-5 days)

Concepts to learn:

- Material-UI or Tailwind CSS (choose one based on preference)
- Component libraries and theming
- Responsive design principles
- CSS-in-JS (if using Material-UI)

Practice project:

- Style your previous projects with the chosen UI library
- Create a responsive layout

Phase 6: Project-Specific Implementation (7-10 days)

Concepts to learn:

- Chart integration (Chart.js or Recharts)
- Data visualization techniques
- Complex state management patterns
- Code organization and folder structure

Implementation steps for your project:

1. Set up React application with Create React App or Vite
2. Implement authentication components (Login/Register)
3. Create protected route component
4. Build dashboard layout and components
5. Implement API service for backend communication
6. Create review detail page with category breakdown
7. Add chart visualization for scores
8. Implement responsive design

Phase 7: Testing and Optimization (3-5 days)

Concepts to learn:

- React testing basics (Jest and React Testing Library)
- Performance optimization (useMemo, useCallback)
- Code splitting
- Deployment preparation

Recommended Resources:

1. Official React documentation (reactjs.org)
2. React Router documentation
3. Material-UI or Tailwind CSS documentation
4. Chart.js or Recharts documentation
5. JavaScript.info for modern JavaScript concepts

Timeline Estimate:

- Total: 30-45 days (learning while building)
- Accelerated pace: 3-4 weeks with full-time focus

Remember to build small practice projects at each phase to reinforce your learning before tackling the complete application. Start with simple components and gradually add complexity as you become more comfortable with React concepts.