1*. Introduction*

Give a brief introduction to your project and the list of features, with a special focus on data management (add/change/delete).

* This project is a full-featured Recipe Sharing Website built with React, Bootstrap, and SCSS. The goal is to provide users with access to public and member-exclusive recipes, while also enabling user engagement through features like ratings, reviews, and personal submissions.
* Quick key features and data management: Sign-up / Login system, member-only access to advanced recipes, filtering by cultural origin, rating system (0–5 stars) and text reviews, recipes are stored in localStorage, users can add recipes (pending admin approval), admins can approve/delete pending submissions, and users can delete their own recipes.

*2. User Interface Design and Prototype*

Give some sample user interface layouts for your application. You can use either wireframe diagrams or prototyping tools to construct the mock representations of your UI design.

**Sample Screens (Wireframes/Mockups):**

You can represent the following as wireframes (use tools like Figma, Balsamiq, or draw them):

* Home Page (Public Recipes Grid)
* Sign Up / Login Form
* Member Dashboard (Advanced Recipes + Filters)
* Recipe Detail Page (Image, ingredients, steps, reviews, rating)
* Submit Recipe Page (Form with required fields + image upload)
* Admin Dashboard (List of pending recipes with approve/delete)
* Spinner Page (“What should I cook?”)
* Profile Page (user info + posted/liked recipes)

Briefly explain the rationale behind designing your UI and how it is geared towards supporting the list of features for your application.

* Bootstrap ensures responsive, mobile-friendly layout.
* Filters and search make recipe browsing efficient.
* Large images + cultural badges increase visual appeal.
* Step-by-step cooking guide improves UX during cooking.
* Visual spinner and top-rated sections boost engagement.
* User profile pics and badges make community feel more personal.
* Admin tools remain hidden from regular users.

*3. Navigation Structure*

Give a brief overview of the navigation structure for your application.

* / → Home (Public Recipes)
* /login → Login Page
* /signup → Signup Page
* /members → Member Page (Advanced Recipes)
* /submit → Submit Recipe
* /admin → Admin Dashboard (Admin only)
* /recipe/:id → Recipe Detail
* /spinner → Random Spinner
* /profile → User Profile (Logged-in users)
* /bookmarks → Bookmarked Recipes (Members)

Briefly indicate a typical flow of your application in terms of user experience. You can use any way of representing the flow. You can also construct a prototype using one of the prototyping tools to illustrate this.

* Visit homepage → see basic recipes
* Try to access member page → see login/signup prompt
* Sign up → redirected to member page
* Browse advanced recipes → click on one → view recipe details
* Bookmark or rate a recipe (if logged in)

Indicate how your React components are connected to one another (routing).

Main Routes defined in App.jsx:

* <Route path="/" element={<PublicRecipes />} />
* <Route path="/login" element={<Login />} />
* <Route path="/signup" element={<Signup />} />
* <Route path="/members" element={<MemberRecipes />} />
* <Route path="/submit" element={<SubmitRecipe />} />
* <Route path="/admin" element={<AdminDashboard />} />
* <Route path="/recipe/:id" element={<RecipeDetail />} />
* <Route path="/spinner" element={<RecipeSpinner />} />
* <Route path="/profile" element={<Profile />} />
* <Route path="/bookmarks" element={<Bookmarks />} />

*4. References*

Provide any references relevant to the report.

React Official Docs: https://react.dev

• React Router Documentation: https://reactrouter.com

• Wireframing tool: https://wireframe.cc/

 https://protoio.medium.com/why-wireframes-are-important-in-the-design-process-de4e773e611

- https://wireframe.cc

- https://www.webdesign-inspiration.com