**🧩System Design Document – URL Shortener Web App**

**Candidate: Ishika Bansal**

**Evaluation Type: Frontend (React)**

**Time Limit: 2 Hours**

**Hosting Environment: http://localhost:3000**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**1. 🎯Objective**

**Create a responsive and scalable client-side URL Shortener Web App with React where the user can shorten URLs, optionally create custom shortcodes and expiry dates, and monitor usage analytics such as click history and source. The application should make use of logging middleware and store data locally.**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**2. Architectural Overview**

* **Frontend Architecture:Component-based design with React Functional Components and Hooks**
* **Modular directory structure (components/, pages/, utils/, middleware/)**
* **Routing:React Router v6 for SPA navigation**
* **Routes: / (Home - shorten URLs), /stats (Analytics), /:shortCode (Redirector)**
* **Persistence:All data in localStorage (client-side persistence)**
* **Redirection Logic: Client-side route /:shortCode initiates redirection if valid and not expired**
* **Traces and updates analytics on every click using updateClicks**
* **Theming & Styling: Material UI with dynamic light/dark mode switch**
* **Responsive layouts with Grid and Box components**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**3. Key Components**

**Component Responsibility**

**URLForm.jsx Form to shorten up to 5 URLs with validation**

**Stats.jsx Displays click stats, location (via Geolocation API), etc.**

**Redirector.jsx Handles URL redirection, logging, and click tracking**

**Navbar.jsx Navigation bar with theme toggle**

**URLCard.jsx Displays shortened URLs with copy option**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**4. Data Modelling**

**Shortened URL Object:**

**{**

**longUrl: string,**

**shortCode: string,**

**createdAt: string (ISO),**

**expiry: string (ISO),**

**clicks: [**

**{**

**timestamp: string,**

**source: string,**

**location: string (or coordinates)**

**}**

**]}**

**Stored as array in localStorage under key: "shortenedURLs"**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**5. ⚙️ Technology Choices & Justifications**

**Technology Purpose & Justification**

**React.js Single-page app with dynamic state, rapid rendering, ease of maintenance**

**Material UI Modernity, accessibility, responsiveness in UI components**

**localStoragePersistent data between sessions on client**

**React Router v6 Client-side navigation without the page reload**

**Geolocation API Captures location of user click data**

**Custom Logging Middleware**

**Replaces console.log for systematic audit-style logging**

**UUID / Random Generator**

**Generates unique shortcodes programmatically**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**6. Client-Side Logging Strategy**

**• Custom middleware logs each user action (Shorten URL, Click, Redirect, etc.)**

**• Saved in localStorage as "logs" for session-spanning traceability**

**• Has timestamp and metadata**

**• Console logging is disabled according to instructions**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**7. ✅ Validation & Error Handling**

**Input Field Validation Applied**

**Long URL Must be valid HTTP/HTTPS URI**

**Validity (min) Must be an integer (default = 30 min if unspecified)**

**Custom Shortcode**

**Must be unique and alphanumeric**

**• Inline errors indicated using TextField helperText and error props**

**• Invalid/expired links redirect to homepage with logging**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**8. Assumptions Made**

* **Users are pre-authenticated; no authentication required**
* **All redirection and data retrieval done client-side only**
* **no backend/database; storage and analytics handled through localStorage**
* **uniqueness of Shortcode enforced through loop and in-memory check**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**9. Scalability Considerations (Future Scope)**

* **Migrate storage from localStorage to a backend with a DB (e.g., MongoDB/PostgreSQL)**
* **•Implement user authentication for personalized URL management**
* **•Implement rate-limiting & bot detection**
* **•Implement IP-based Geo API for precise location tracking**
* **•Implement QR code generation for mobile sharing**
* **•Host on cloud with real hosting (Vercel, Firebase, Vultr, etc.)**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**10. Testing & Dev Notes**

* **Component unit testing can be implemented through Jest**
* **Manual testing was performed for:**
* **URL shortening**
* **Redirection**
* **Expiry handling**
* **Statistics tracking**
* **Responsive layout on desktop & mobile**
* **Light/Dark theme toggle**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**✅ Conclusion**

**This front-end application is user-centered, maintainable, and modular. All of the essential features — analytics, logging, routing, validation, and data handling — are handled neatly on the client side according to the constraints of the problem.**

**Please let me know if you prefer it in PDF or in .md format for submission.**