URL Shortener – Design Document

Author: Campus Hiring Evaluation
Date: September 2025

1. Overview

This document describes the architectural and design choices for the client-side URL Shortener application built using React and Material UI. The app supports URL shortening, shortcode uniqueness, optional validity, client-side redirection, analytics display, and custom logging middleware. The application runs exclusively at http://localhost:3000.

2. Architecture

2.1 Application Structure

- React SPA (Single Page Application) using React Router v6.
- Pages:
  • Shortener Page – Create short URLs (up to 5 at a time).
  • Statistics Page – List all shortened URLs with expiry, clicks, and analytics.
  • Redirector Route – Handles /:shortcode and redirects to the original URL.

2.2 Core Services

- LoggerMiddleware
  • Custom logger (no console.log).
  • Stores structured logs (timestamp, action, details) in localStorage.
  • Used for recording all operations: shortening, redirecting, errors.

- ShortcodeService
  • Generates random alphanumeric codes (default length 6–8).
  • Ensures uniqueness across the app session.
  • Validates user-provided shortcodes (alphanumeric, not already taken).

- StorageService
  • Persists shortened URLs in localStorage.
  • Model:
    shortcode: abcd12
    longUrl: https://example.com
    expiry: 2025-09-04T12:30:00Z
    createdAt: ...
    clicks: 0
  • Enables session persistence and statistics display.

3. Data Modeling

URL Entry Object

Field	Type	Description
shortcode	string	Unique shortcode for redirect
longUrl	string	Original URL

expiry	datetime	Expiry time (default = now + 30 mins)
createdAt	datetime	When the short link was created
clicks	integer	Incremented on each redirect

4. Routing Strategy

- `/` → ShortenerPage (default entry)
- `/stats` → StatisticsPage
- `/:shortcode` → Redirector
  - Checks StorageService for matching shortcode.
  - Validates expiry.
  - Redirects to original URL or shows error.