🧪 Full Stack Developer Internship - Assignment

📌 Title: Advanced URL Shortener with Analytics Dashboard

⏱️ Duration: 1 Day

🚀 Introduction

This assignment assesses your ability to design and implement a robust backend API and a lightweight frontend for a **URL Shortener** system with advanced **analytics and tagging capabilities**.

🎯 Objective

✅ Backend Objectives:

Design an API with the following key features:

1. **Shorten URLs**
   - POST request with the original URL
   - Generate a **unique short code**
   - Support for **custom short codes** (with uniqueness check)

2. **Redirection with Tracking**
   - GET `/short/:code` should:
     - Log visit with:
       - Timestamp
       - IP address
       - Device type (mobile/desktop/tablet detection)
     - Increment counters:
       - Total visits
       - Unique visitors (via hashed IP/User-Agent or fingerprint)
       - Visit source (referrer)
     - Redirect with HTTP 302

3. **Analytics Endpoint**
   - GET `/analytics/:code` should return:
     - Original URL
     - Total number of visits

        - Unique visitors
        - Breakdown by:
            - Device type
            - Referrers (Top 5 sources)
        - Time series data (visits per day/hour)
        - Tags assigned to the shortened URL (see Tagging below)

4. **Tagging Mechanism**
    - Add ability to associate **custom tags** (like "social", "marketing", "internal") with each shortened URL during creation.
    - GET `/tags/:tag` should return a list of short URLs with that tag and basic analytics.

5. **Short URL Expiry**
    - Add an optional expiry time (in hours or date) when creating the short URL
    - After expiry, short URL returns 410 (Gone)

🖥️ Frontend Objectives (Mini Dashboard):

Create a simple and responsive **React** dashboard to:

- Allow user to shorten a new URL with:
    - Expiry date/time
    - Optional custom code
    - Optional tags
- View list of their shortened URLs with:
    - Analytics summary (visits, unique, etc.)
    - Tags
- Click to expand/view:
    - Pie chart for device types (desktop, mobile, etc.)
    - Line/bar graph for time series data
    - Table of top referrers
- Filter links by tag (optional)

🧱 Technical Stack (Recommended)

- **Backend**: Node.js (Express.js preferred)
- **Database**: PostgreSQL or MongoDB
- **Frontend**: React.js (Vite or CRA)

📦 Deliverables

1. **Source Code** in GitHub or zipped format with structured folders:
    - `/backend`, `/frontend`
2. **README.md** including:
    - API documentation (with request/response examples)
    - Postman collection (exported JSON)
    - How to run backend + frontend locally
    - Optional: deployed link (if possible)
3. **Database schema/ERD** diagram (image or markdown)
4. (Optional) Script to generate test visit data