

Name :Khushi Ganesh Suvarna

Intern ID :121

URL SHORTNER POC

Objective

- Accept long URLs from the user through a simple interface.
- Generate a unique, short alphanumeric code for each URL.
- Store the mapping of short code and original URL in a local SQLite database.
- Redirect visitors from the short link to the original destination.
- Present a custom fake domain name in the displayed short URL for demonstration purposes, while the actual link uses the local Flask server for functionality.

Technology Stack

- Backend: Python with Flask framework
- Database: SQLite
- Frontend: HTML with Flask templating

Features

- User Input Form – Users can paste any long URL into a simple interface.
- Unique Short Code Generation – Generates a random 6-character alphanumeric short code.
- Database Storage – Stores short code and corresponding long URL in SQLite.
- Redirection – Visiting /shortcode redirects to the saved original URL.
- Fake Domain Display – Shows a fake domain in the UI for demo purposes; the clickable link points to the actual local server.

How It Works

- The user opens the application in a browser via <http://127.0.0.1:5000>.
- A long URL is entered into the form and submitted.
- The backend normalizes the URL, generates a unique 6-character short code, and saves it to SQLite with the original URL.
- The result page shows:
 - Display Link: Fake domain + short code (e.g., <https://demo.com/xyz123>).
 - Clickable Link: Local server + short code (e.g., <http://127.0.0.1:5000/xyz123>).
- When the clickable short link is visited, the application fetches the original URL from the database and redirects the user.

Limitations

- The fake domain is only for display purposes.
- Clicking the fake domain URL will not work unless hosted with proper DNS setup.
- Current implementation stores data only locally (no persistence across servers).

Example Output

Input URL: <https://www.example.com/very/long/link/path>

Displayed Link (Fake): <https://demo.com/aB9xY2>

Clickable Local Link: <http://127.0.0.1:5000/aB9xY2>

Instructions to Run

Prerequisites:

- Python 3.x installed.
- Flask installed (pip install flask).

Steps:

1. Clone or copy the project folder to your local system.

2. Open a terminal in the project folder.
 3. Run the application: `python app.py`
- Open a browser and go to `http://127.0.0.1:5000`.
 - Paste any long URL, generate a short link, and click it to test the redirection.

What This PoC Intends to Achieve

This PoC serves as a functional demonstration of a basic URL shortening service. It verifies that long URLs can be effectively shortened, stored, and redirected using a unique short code, all within a minimal local environment. Additionally, the inclusion of a fake domain in the display link enhances its presentation value without affecting the functional redirection process.