

URL Shortener Report

Introduction

URL shortening is a technique where a long URL is shortened to a shorter version that redirects to the original URL. This technique is useful in situations where there is a limit on the number of characters that can be used, such as in social media posts or messaging apps. URL shortening can also be used to track clicks on the links and to hide the original URL.

This URL Shortener web application is a Flask-based application that allows users to shorten URLs and view their history of shortened URLs. This web application is built using HTML, CSS for the front-end and Flask, SQLAlchemy, and pyshorteners for the back-end.

Features

The URL Shortener web application has the following features:

1. Shorten URLs: Users can enter a long URL in the input field on the home page and click on the "Short URL Now" button to generate a shortened URL. The application uses the pyshorteners library to generate a unique short URL for the given long URL.
2. Copy Short URL: After a short URL is generated, users can click on the "Copy Short URL" button to copy the shortened URL to the clipboard. The application uses the `navigator.clipboard.writeText()` method to copy the short URL to the clipboard.
3. View History: Users can click on the "Show History" button on the home page to view a list of all the shortened URLs that they have generated. The history page displays the long URL and the corresponding shortened URL.

Design

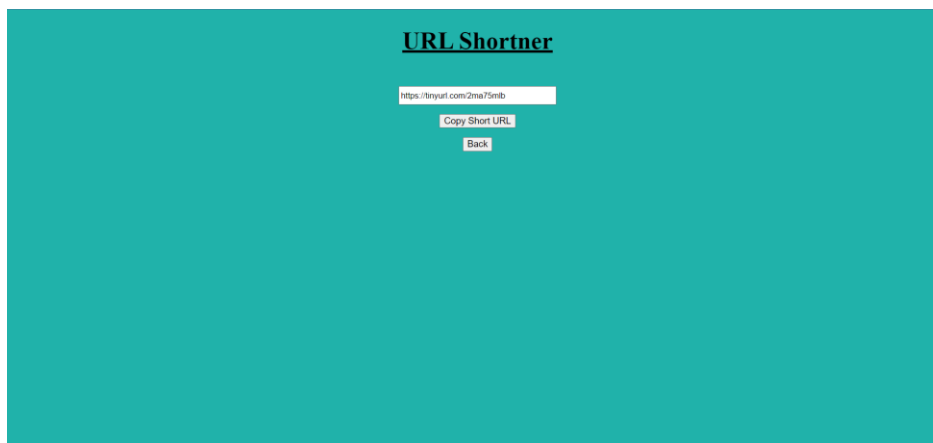
The URL Shortener web application has a simple and intuitive design. The home page has an input field where users can enter the long URL and a button to generate the shortened URL. The history page displays a list of all the shortened URLs that the user has generated.



[home.html](#)

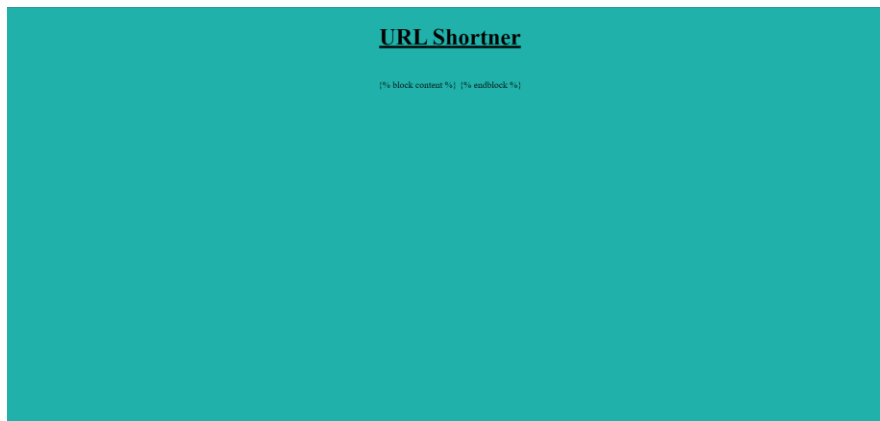


[history.html](#)



[display.html](#)

The application uses a `layout.html` file that defines the basic structure of the web pages. The `layout.html` file has a header that contains the title of the web page and a center tag that wraps around the content of the web page. The `home.html`, `display.html`, and `history.html` files extend the `layout.html` file and define the specific content of the web pages.



[layout.html](#)

The CSS file (style.css) is used to style the web pages. The CSS file defines the font, colour, and layout of the web pages.

Implementation

The URL Shortener web application is implemented using Flask, SQLAlchemy, and pyshorteners.

Flask

Flask is a micro web framework written in Python. Flask is used to define the routes of the web application and to render the HTML templates. The app.py file defines the routes of the web application and contains the logic for generating the shortened URLs.

SQLAlchemy

SQLAlchemy is a SQL toolkit and ORM that provides a set of high-level APIs for working with relational databases in Python. SQLAlchemy is used to create the database schema and to interact with the database. The app.py file defines the URL model and creates the database schema using SQLAlchemy.

pyshorteners

pyshorteners is a Python library for URL shortening. pyshorteners is used to generate a unique shortened URL for a given long URL. The app.py file uses pyshorteners to generate the shortened URL.

Deployment

The URL Shortener web application can be deployed on a web server such as Apache or Nginx. The application can also be deployed on a platform-as-a-service (PaaS) provider such as Heroku or AWS Elastic Beanstalk.

Conclusion

The URL Shortener web application is a simple and useful web application that allows users to generate shortened URLs and view their history of shortened URLs. It can be very useful in various real-life cases where sharing the long URL may be inconvenient or the URL owner may not want to share the whole URL or there may be some character limit for sharing some text or any other case.