



## **URL SHORTNER WEB APPLICATION (FLASK)**

**AUTHOR**

**VINAY NAMANI**

**MENTOR & GUIDE**

**\*\* KANAV BANSAL \*\***

# URL SHORTNER with FLASK FRAMEWORK

## **Objective:**

The name URL Shortener itself speaks for the Shortening the lengthy URLs. A URL is a short form of Uniform Resource Locator, which will request a particular website or any IP address. But sometimes it gets very lengthy and clumsy due to tremendous number of Websites or the capabilities they are having or the number of pages a website holds. So, a Shortened URL is very useful when a person wants to share it with his colleagues or with friends or to remember yourself.

## **Advantages:**

1. Not like very lengthy and clumsy URLs, Shortened URLs looks clean and aesthetic.
2. When you want to circulate or share a particular URL with the team or Developers it will get hard to memorize or type out the entire URL.
3. It will be easy to handle while mentioning in posts or blogs, gives clean look.
4. Easy to memorize than the long URLs.

## **What will our Web app do?**

The Application will get the Long URL as an input from the User who wants a shorten URL and it has to be a valid, or else there will be no use in making a short URL for an original URL which is not exist. The user will get the Shortened URL at the End of the line and he can copy it directly to the Clipboard with a single click.

## Technical Stack:

1. Python
2. Flask
3. HTML
4. CSS
5. MongoDB
6. Visual Studio Code
7. MongoDB Compass

## Application Flow:

1. It will Starts when the User Enters an URL which he wants in short form and it will get passed to a method.
2. The method will take the Input URL and create an alternate & UNIQUE ID to keep them as a key-valued pair.
3. Then the alternate URL will get added with the localhost. To make sure that it is working.
4. Then both short URL and Original URL will get inserted into Database for the Record purpose and avoid making an existing UNIQUE id.
5. Then when you call the Shortened URL after copying it from the Application.
6. Whenever you requested the shortened URL, it will check the Value for that Key and look for the Value which is assigned with it.
7. Finally, it will get Redirected to the URL which is in the Dictionary Value.