



**INNOMATICS**  
**RESEARCH LABS**

**INTERNSHIP PROJECT REPORT**

**ON**

**WEB BASED URL SHORTENER**  
**(BASIC)**

**SUBMITTED BY**

**DURGESH BHARGAVA**

**BATCH**

**OCT – 21**

**GUIDED BY**

**CHIEF DATA SCIENTIST “KANAV BANSAL”**

# Flask Project (Basic Implementation)

## URL Shortener Web Application (Basic)

### Abstract

URL is an acronym for Uniform Resource Locator and is a reference (an address) to a resource on the Internet. A URL has two main components: Protocol identifier: For the URL `http://example.com`, the protocol identifier is `http`. Resource name: For the URL `http://example.com`, the resource name is `example.com`.

URL shortening is a technique on the World Wide Web in which a URL (Uniform Resource Locator) is substantially shorter. Using python, flask module, sqlite and html to make a web application where the long or heavy url will be shorter and the shorter link can be copied directly using the button, and all data history will be stored in the history page.

### Motivation

There are several reasons to use URL shortening. Often regular unshortened links may be aesthetically unpleasing. Many web developers pass descriptive attributes in the URL to represent data hierarchy, command structure, transition path or session information. This can result in URLs that are hundreds of character long and contains complex character patterns. Such URLs are difficult to memorize, type out or distribute. As a result, Long URLs must be copied and pasted for reliability. Thus, short URLs may be more convenient for a websites or hard copy publications

### Scope

This web application is very useful when people need to give the reference in paper, journals, books and many more places. URL shorteners spit out links that are easier to share. More importantly, they let small businesses track marketing campaigns and other incoming traffic

## Methodology

Set the environment using the import module of python and set the path of html and .py file.

## Technology Stack

1. Python
2. Flask
3. HTML
4. SQLAlchemy

## Tool Stack

1. Visual Studio Code IDE

In the .py file we make three routes and one of the “/home” routes consist of a get and post request to give the proper sorted URL to the user.

Instead of this, two main html files for history and shortening purposes with some basic **CSS** code for color and wording style.

The main logic which is used for giving the sorted URL is that, first it consists of all flask IP ADDRESS then use combinations of alphabets A to Z in small and capital and numerical values. So, we will provide a unique shortened URL with the factorial of this combination.

## Working Phases

The system working is divided into 4 phases -

1. URL input: - The user pastes the URL into the input.
2. URL shortening: - The goal of this phase is to map the given URL with a max of 10 unique alphanumeric characters.
3. URL Mapping:- The goal of this phase is to map the short and long URL in the database so that the URL redirection phase can use this information for future use.
4. URL redirection:- The goal of this phase is to concatenate the unique alphanumeric character with the systems domain, to make the shorten URL work.