INNOMATİCS
RESEARCH LABS

INTERNSHIP PROJECT REPORT

ON

FLASK - APPLICATION DEVELOPMENT

WEB BASED URL SHORTENER

SUBMITTED BY

LOKA AKASH REDDY

BATCH

OCTOBER 2021

GUIDED BY

CHIEF DATA SCIENTIST KANAV BANSAL

WEB BASED URL SHORTENER

INTRODUCTION

URL shortening is a technique on the World Wide Web in which a URL (Uniform Resource Locator) may be made substantially shorter and still direct a user to a required webpage. This is achieved by using a redirect which links to the web page that has a long URL. For example, the URL "https://example.com/assets/category_B/subcategory_C/Foo/" can be shortened to "https://example.com/Foo", can be shortened to "https://w.wiki/U. Often the redirect domain name is shorter than the original one. A friendly URL may be desired for messaging technologies that limit the number of characters in a message (for example SMS), for reducing the amount of typing required if the reader is copying a URL from a print source, also for making it easier for a person to remember, or for the intention of a permalink. In November 2009, the shortened links of the URL shortening service Bitly were accessed 2.1 billion times.

PURPOSE

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 hierarchies, command structures, transition paths or session information. This can result in URLs that are hundreds of characters long and that contain complex character patterns. Such URLs are difficult to memorize, type out or distribute. As a result a Long URLs must be copied and pasted from reliability. Thus, short URLs may be more convenient for websites or hard copy publications, the latter often requiring that very long strings be broken into multiple lines or truncated.

URLs encoded in two dimensional barcodes such as QR code are often shortened

by a URL shortener in order to reduce the printed area of the code, or allow printing at lower density in order to improve scanning reliability.

PROJECT SCOPE

With the advancement in a web based systems, security is one of the major aspect coming in through any form. The spamming of URL by making a bot attack a website can cause the website to crash. So there should be a check on which URLs are allowed to be shortened, for example. there should not be a URL which is invalid but is in our DB as a shortened URL since the URL is not valid it should not be shortened. The websites returning status code like 400, 401, 402 etc should not be shortened.

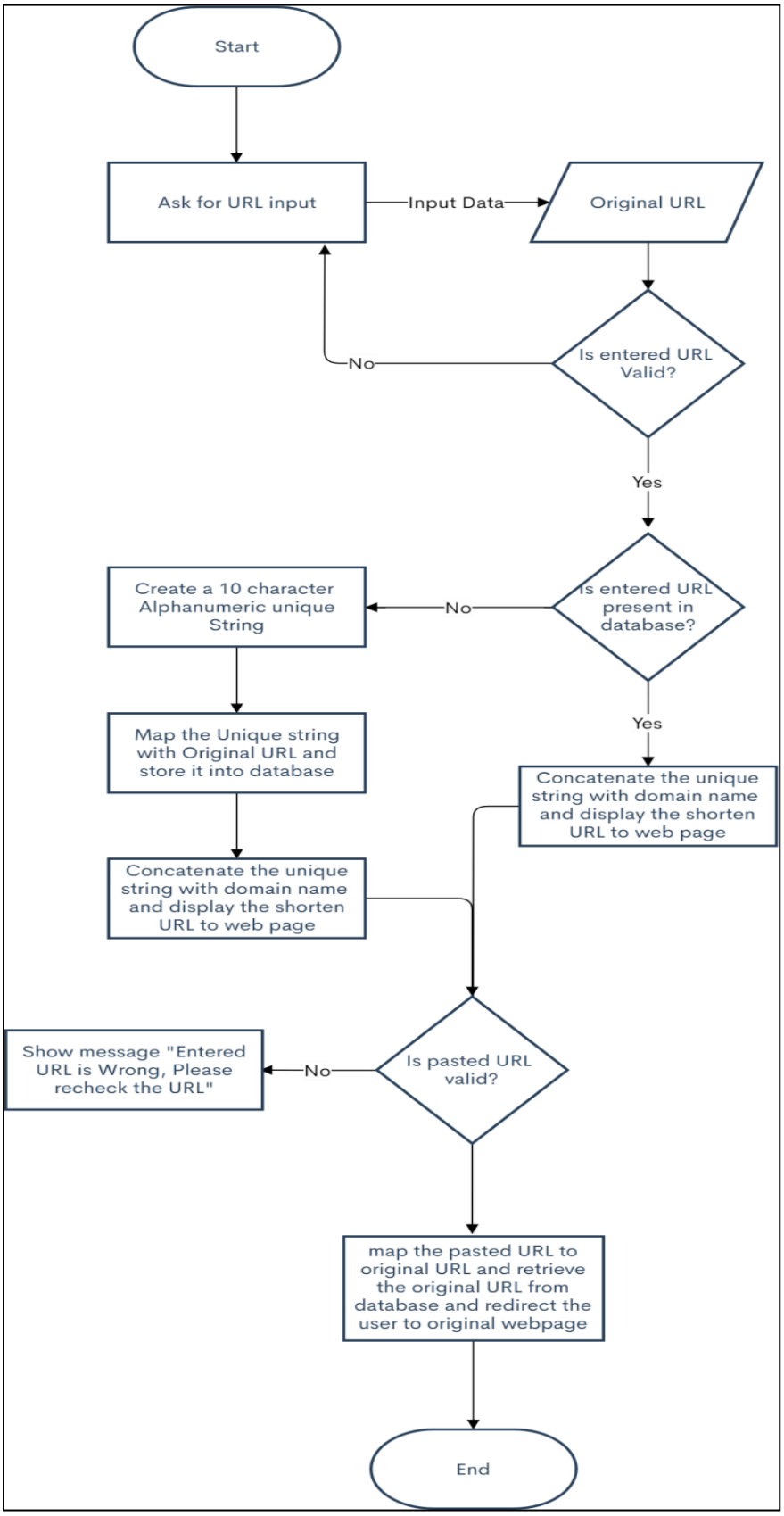
PROJECT GOAL

The goal of the project is to build the system which allows user to enter a URL. The URL should be a valid URL and the request to shorten the URL should not be possible unless and until the entered URL is valid based on semantics. If the URL is valid, the shortened URL is given to the user.

Technology Stack

1. Python
2. Flask
3. HTML 5
4. CSS
5. JavaScript
6. MySQL

FLOWCHART


Start
Ask for URL input
Input Data
Original URL
Is entered URL Valid?
No
Yes
Is entered URL present in database?
No
Create a 10 character Alphanumeric unique String
Map the Unique string with Original URL and store it into database
Concatenate the unique string with domain name and display the shorten URL to web page
Yes
Concatenate the unique string with domain name and display the shorten URL to web page
Is pasted URL valid?
No
Show message "Entered URL is Wrong, Please recheck the URL"
map the pasted URL to original URL and retrieve the original URL from database and redirect the user to original webpage
End