Table of Contents

[Instructions 2](#_Toc535802221)

[Scenario 1 3](#_Toc535802222)

[Scenario 2 3](#_Toc535802223)

[Architecture 4](#_Toc535802224)

[Database structure 5](#_Toc535802225)

[Docker building 6](#_Toc535802226)

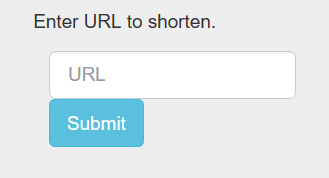
[Project files 8](#_Toc535802227)

[docker image 8](#_Toc535802228)

[docker run 8](#_Toc535802229)

## Instructions

Enter the URL to be shortened

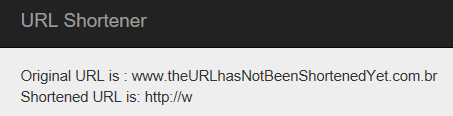


There are two possibilities:

Scenario 1 – The URL has NOT been shortened YET OR

Scenario 2-- The URL has ALREADY been shortened

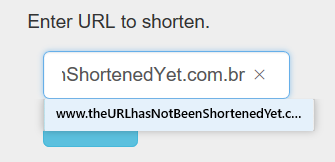
Scenario 1 – Lets insert a new URL into the API



The short and long URLs will be inserted on the database (sqlite)



Scenario 2 - Let s try to insert the same URL

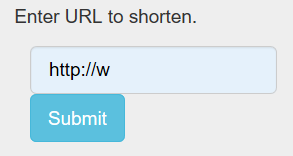


It shows the same short URL

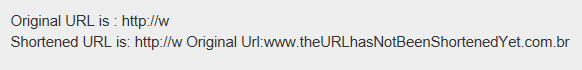
Notice that it did NOT include in the database – as it was already included!

Revering URL – getting the Original URL

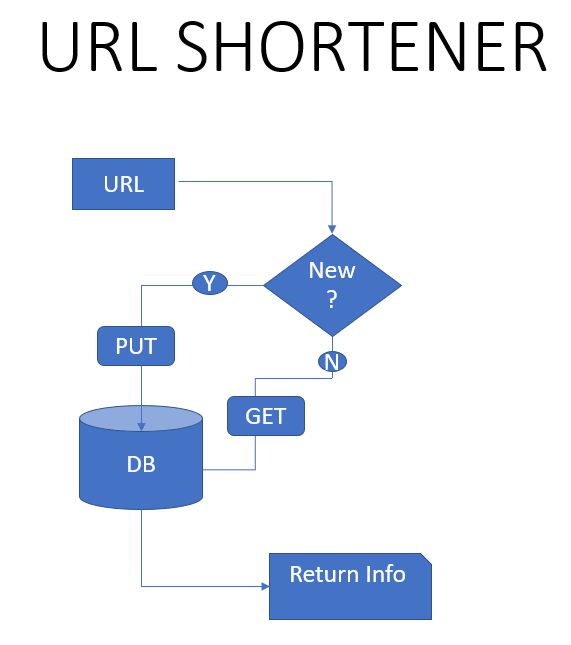
Suppose we have the following Url shortened: [**http://w**](http://w)used from the previous exercise



It returns the original URL



## Architecture



## Database structure

CREATE TABLE url (

id INTEGER PRIMARY KEY AUTOINCREMENT,

longUrl VARCHAR (300),

shortUrl VARCHAR (20)

);

## Docker building

rmsa@rmsa-dev:~/urlshort/van4/docker$ docker build --tag=urlshortener .

Sending build context to Docker daemon  27.65kB

Step 1/7 : FROM python:3.7-slim

3.7-slim: Pulling from library/python

177e7ef0df69: Pull complete

1dee839b70d8: Pull complete

ed87523633c1: Pull complete

62960803b9a0: Pull complete

52b8d619d6f0: Pull complete

Digest: sha256:07d851f56640f34bd6f6e5fad9b2dd116288d142839922f958c0c489cab02a4b

Status: Downloaded newer image for python:3.7-slim

 ---> f9c1866f07ea

Step 2/7 : WORKDIR /app

 ---> Running in 1d3793abc124

Removing intermediate container 1d3793abc124

 ---> 0d1d748ca440

Step 3/7 : COPY . /app

 ---> 62b7a5bc4c24

Step 4/7 : RUN pip install --trusted-host [pypi.python.org](http://pypi.python.org/) -r requirements.txt

 ---> Running in 6c73fc21df5d

Collecting Click==7.0 (from -r requirements.txt (line 1))

  Downloading <https://files.pythonhosted.org/packages/fa/37/45185cb5abbc30d7257104c434fe0b07e5a195a6847506c074527aa599ec/Click-7.0-py2.py3-none-any.whl> (81kB)

Collecting Flask==1.0.2 (from -r requirements.txt (line 2))

  Downloading <https://files.pythonhosted.org/packages/7f/e7/08578774ed4536d3242b14dacb4696386634607af824ea997202cd0edb4b/Flask-1.0.2-py2.py3-none-any.whl> (91kB)

Collecting itsdangerous==1.1.0 (from -r requirements.txt (line 3))

  Downloading <https://files.pythonhosted.org/packages/76/ae/44b03b253d6fade317f32c24d100b3b35c2239807046a4c953c7b89fa49e/itsdangerous-1.1.0-py2.py3-none-any.whl>

Collecting Jinja2==2.10 (from -r requirements.txt (line 4))

  Downloading <https://files.pythonhosted.org/packages/7f/ff/ae64bacdfc95f27a016a7bed8e8686763ba4d277a78ca76f32659220a731/Jinja2-2.10-py2.py3-none-any.whl> (126kB)

Collecting MarkupSafe==1.1.0 (from -r requirements.txt (line 5))

  Downloading <https://files.pythonhosted.org/packages/e4/c4/adcc2d6f2ac2146cc04e076f14f1006c1de8e1e747fa067668b6573000b8/MarkupSafe-1.1.0-cp37-cp37m-manylinux1_x86_64.whl>

Collecting Werkzeug==0.14.1 (from -r requirements.txt (line 6))

  Downloading <https://files.pythonhosted.org/packages/20/c4/12e3e56473e52375aa29c4764e70d1b8f3efa6682bef8d0aae04fe335243/Werkzeug-0.14.1-py2.py3-none-any.whl> (322kB)

Installing collected packages: Click, Werkzeug, MarkupSafe, Jinja2, itsdangerous, Flask

Successfully installed Click-7.0 Flask-1.0.2 Jinja2-2.10 MarkupSafe-1.1.0 Werkzeug-0.14.1 itsdangerous-1.1.0

Removing intermediate container 6c73fc21df5d

 ---> c43302bb6163

Step 5/7 : EXPOSE 80

 ---> Running in 5ab293bd8830

Removing intermediate container 5ab293bd8830

 ---> dcbe90eb315c

Step 6/7 : ENV NAME World

 ---> Running in 4b68ca9406ef

Removing intermediate container 4b68ca9406ef

 ---> bfa44b5403fa

Step 7/7 : CMD ["python", "api.py"]

 ---> Running in bb3ac6fd4fa1

Removing intermediate container bb3ac6fd4fa1

 ---> c7fd753fbce4

Successfully built c7fd753fbce4

Successfully tagged urlshortener:latest

## Project files

rmsa@rmsa-dev:~/urlshort/van4/docker$ ls -ltr

total 32

-rw-r--r-- 1 rmsa rmsa  3327 jan 21 00:01 api.py

-rw-r--r-- 1 rmsa rmsa 16384 jan 21 00:02 db.db

-rw-r--r-- 1 rmsa rmsa    98 jan 21 00:04 requirements.txt

drwxr-xr-x 2 rmsa rmsa  4096 jan 21 00:41 templates

-rw-r--r-- 1 rmsa rmsa   514 jan 21 00:52 Dockerfile

## docker image

rmsa@rmsa-dev:~/urlshort/van4/docker$ docker image ls

REPOSITORY                                            TAG                    IMAGE ID            CREATED             SIZE

urlshortener                                          latest                 c7fd753fbce4        26 seconds ago      152M

## docker run

rmsa@rmsa-dev:~/urlshort/van4/docker$ docker run -p 4000:80 urlshortener  
 \* Serving Flask app "api" (lazy loading)  
 \* Environment: production  
   WARNING: Do not use the development server in a production environment.  
   Use a production WSGI server instead.  
 \* Debug mode: on  
 \* Running on <http://127.0.0.1:5000/> (Press CTRL+C to quit)  
 \* Restarting with stat  
 \* Debugger is active!  
 \* Debugger PIN: 820-629-060