Prerequisite and Tools

Prerequisite	Tools	Remorks
.NET Core 3.1	Visual Studio 2019	VS Code
C#	Docker Desktop	Windows/Linux, Cmd
MongoDB	Postman	

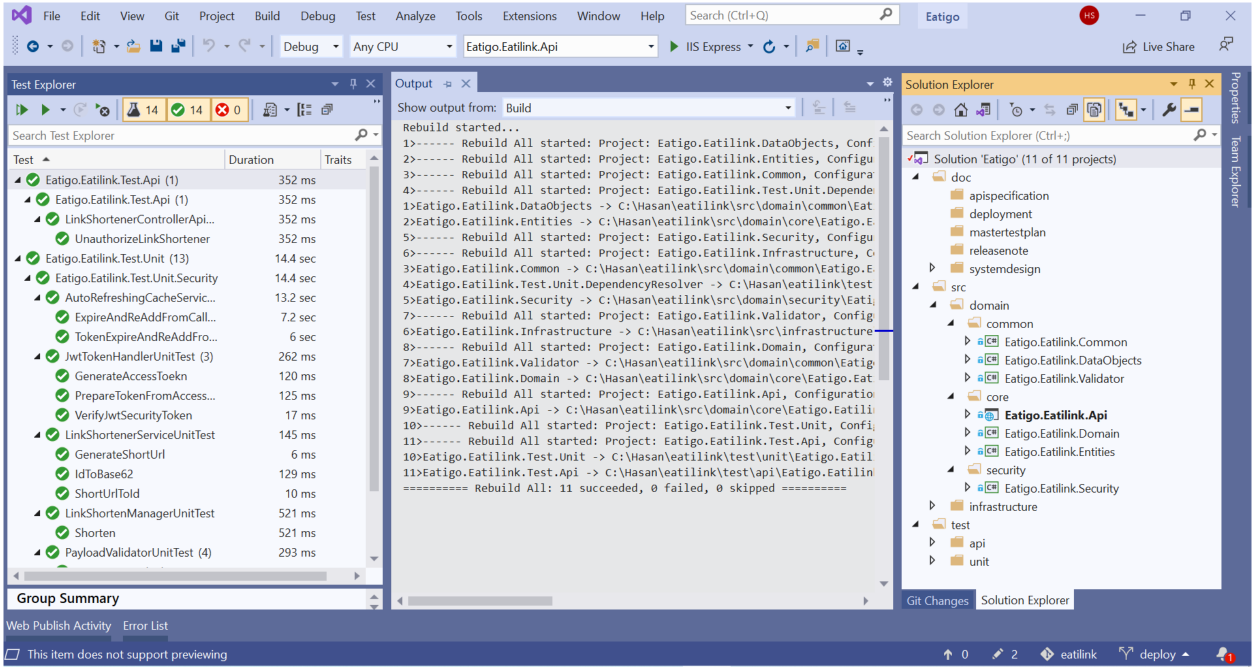
Github Repository

https://github.com/HasanShahjahan/eatilink/tree/master

Project Structure and Configuration

Project Structure

Project has basically three parts doc, src and test. All the documents are available under the doc folder, src includes main project and test section includes api and unit tests.



```
Test                                      Duration   Traits
◢ Eatigo.Eatilink.Test.Api (1)              352 ms
  ◢ Eatigo.Eatilink.Test.Api (1)            352 ms
    ◢ LinkShortenerControllerApi...         352 ms
        UnauthorizeLinkShortener            352 ms
◢ Eatigo.Eatilink.Test.Unit (13)           14.4 sec
  ◢ Eatigo.Eatilink.Test.Unit.Security     14.4 sec
    ◢ AutoRefreshingCacheServic...         13.2 sec
        ExpireAndReAddFromCall...           7.2 sec
        TokenExpireAndReAddFro...             6 sec
    ◢ JwtTokenHandlerUnitTest (3)           262 ms
        GenerateAccessToekn                 120 ms
        PrepareTokenFromAccess...           125 ms
        VerifyJwtSecurityToken               17 ms
    ◢ LinkShortenerServiceUnitTest          145 ms
        GenerateShortUrl                      6 ms
        IdToBase62                          129 ms
        ShortUrlToId                         10 ms
    ◢ LinkShortenManagerUnitTest            521 ms
        Shorten                             521 ms
    ◢ PayloadValidatorUnitTest (4)          293 ms
```

Show output from: Build

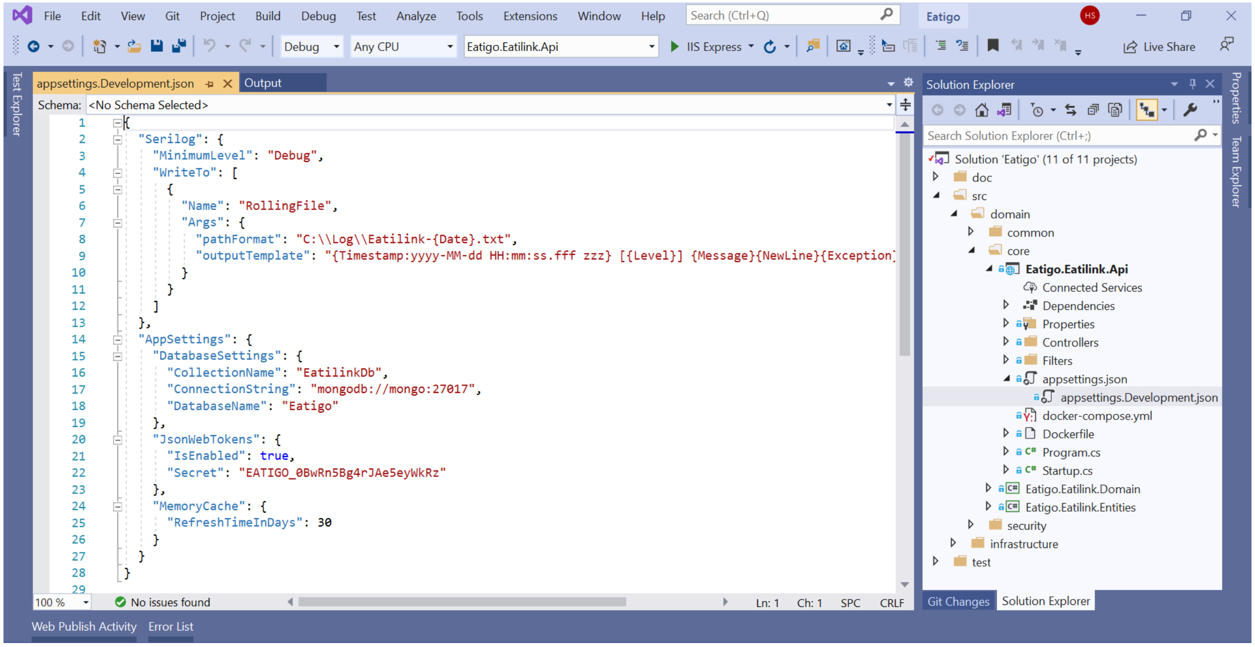
```
Rebuild started...
1>------ Rebuild All started: Project: Eatigo.Eatilink.DataObjects, Conf
2>------ Rebuild All started: Project: Eatigo.Eatilink.Entities, Configu
3>------ Rebuild All started: Project: Eatigo.Eatilink.Common, Configura
4>------ Rebuild All started: Project: Eatigo.Eatilink.Test.Unit.Depende
1>Eatigo.Eatilink.DataObjects -> C:\Hasan\eatilink\src\domain\common\Eat
2>Eatigo.Eatilink.Entities -> C:\Hasan\eatilink\src\domain\core\Eatigo.E
5>------ Rebuild All started: Project: Eatigo.Eatilink.Security, Configu
6>------ Rebuild All started: Project: Eatigo.Eatilink.Infrastructure, C
3>Eatigo.Eatilink.Common -> C:\Hasan\eatilink\src\domain\common\Eatigo.E
4>Eatigo.Eatilink.Test.Unit.DependencyResolver -> C:\Hasan\eatilink\test
5>Eatigo.Eatilink.Security -> C:\Hasan\eatilink\src\domain\security\Eati
7>------ Rebuild All started: Project: Eatigo.Eatilink.Validator, Config
6>Eatigo.Eatilink.Infrastructure -> C:\Hasan\eatilink\src\infrastructure
8>------ Rebuild All started: Project: Eatigo.Eatilink.Domain, Configura
7>Eatigo.Eatilink.Validator -> C:\Hasan\eatilink\src\domain\common\Eatig
8>Eatigo.Eatilink.Domain -> C:\Hasan\eatilink\src\domain\core\Eatigo.Eat
9>------ Rebuild All started: Project: Eatigo.Eatilink.Api, Configuratio
9>Eatigo.Eatilink.Api -> C:\Hasan\eatilink\src\domain\core\Eatigo.Eatili
10>------ Rebuild All started: Project: Eatigo.Eatilink.Test.Unit, Confi
11>------ Rebuild All started: Project: Eatigo.Eatilink.Test.Api, Config
10>Eatigo.Eatilink.Test.Unit -> C:\Hasan\eatilink\test\unit\Eatigo.Eatil
11>Eatigo.Eatilink.Test.Api -> C:\Hasan\eatilink\test\api\Eatigo.Eatilin
========== Rebuild All: 11 succeeded, 0 failed, 0 skipped ==========
```

```
Solution 'Eatigo' (11 of 11 projects)
◢ doc
    apispecification
    deployment
    mastertestplan
    releasenote
  ▷ systemdesign
◢ src
  ◢ domain
    ◢ common
      ▷ Eatigo.Eatilink.Common
      ▷ Eatigo.Eatilink.DataObjects
      ▷ Eatigo.Eatilink.Validator
    ◢ core
      ▷ Eatigo.Eatilink.Api
      ▷ Eatigo.Eatilink.Domain
      ▷ Eatigo.Eatilink.Entities
    ◢ security
      ▷ Eatigo.Eatilink.Security
  ▷ infrastructure
◢ test
  ▷ api
  ▷ unit
```

This item does not support previewing

Project Configuration

Web Configuration



Schema: <No Schema Selected>

```json
{
  "Serilog": {
    "MinimumLevel": "Debug",
    "WriteTo": [
      {
        "Name": "RollingFile",
        "Args": {
          "pathFormat": "C:\\Log\\Eatilink-{Date}.txt",
          "outputTemplate": "{Timestamp:yyyy-MM-dd HH:mm:ss.fff zzz} [{Level}] {Message}{NewLine}{Exception}
        }
      }
    ]
  },
  "AppSettings": {
    "DatabaseSettings": {
      "CollectionName": "EatilinkDb",
      "ConnectionString": "mongodb://mongo:27017",
      "DatabaseName": "Eatigo"
    },
    "JsonWebTokens": {
      "IsEnabled": true,
      "Secret": "EATIGO_0BwRn5Bg4rJAe5eyWkRz"
    },
    "MemoryCache": {
      "RefreshTimeInDays": 30
    }
  }
}
```

Serilog

Please specify the ***log path*** where application log will be saved. It's a ***rolling file*** and we can set the ***log level*** too.

Database Settings

Contains MongoDB database credentials including ***connection string, collection name and database name***.

JWT Token

Requests will be authorized by JsonWebToken ***secret***. Eatilink api is also configurable with token ***enable or disable***. *If we don't want to authorize, please specify false it's isEnabled parameter.*

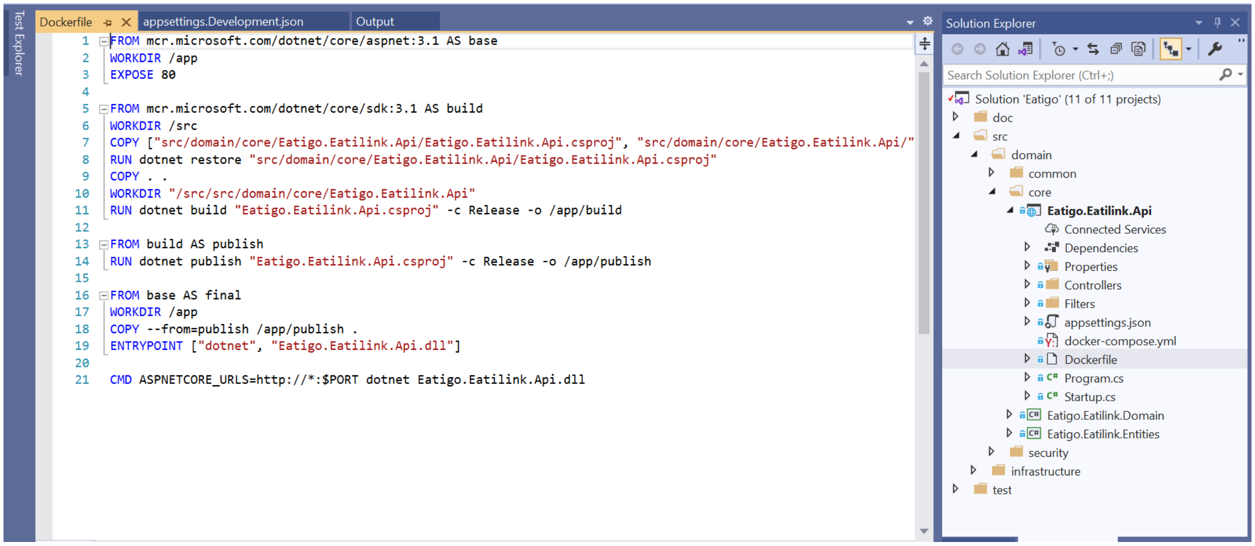
Memory Cache

Auto refreshing memory cache is in place and the ***time limit is also configurable***.

Docker Container Configuration

Docker file

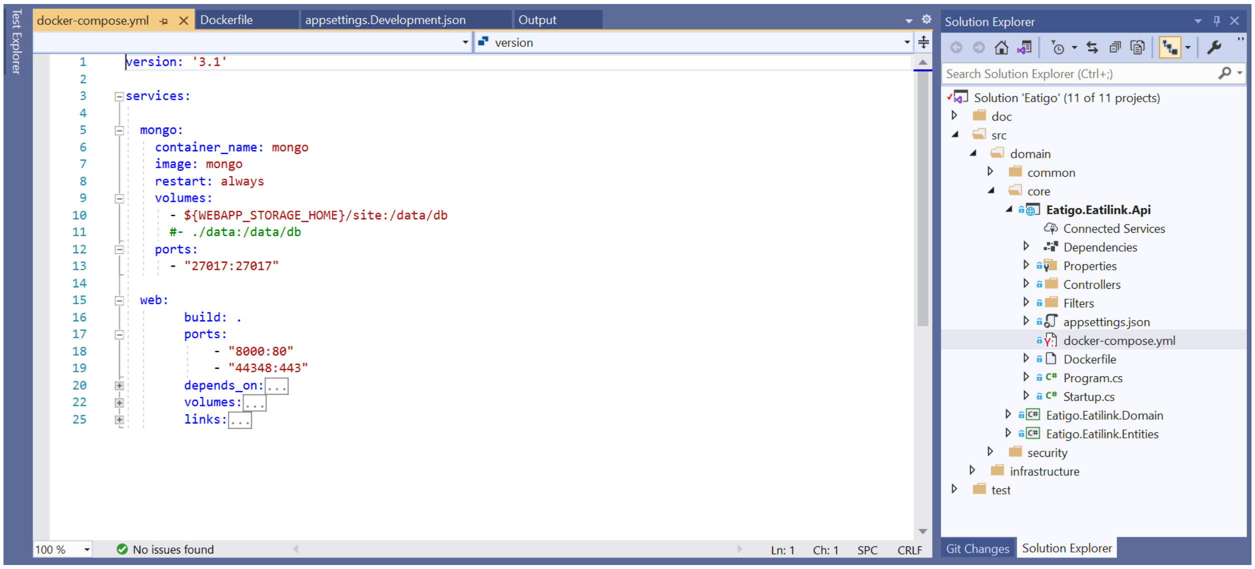
For docker images and Heroku deployment configuration.



```
FROM mcr.microsoft.com/dotnet/core/aspnet:3.1 AS base
WORKDIR /app
EXPOSE 80

FROM mcr.microsoft.com/dotnet/core/sdk:3.1 AS build
WORKDIR /src
COPY ["src/domain/core/Eatigo.Eatilink.Api/Eatigo.Eatilink.Api.csproj", "src/domain/core/Eatigo.Eatilink.Api/"
RUN dotnet restore "src/domain/core/Eatigo.Eatilink.Api/Eatigo.Eatilink.Api.csproj"
COPY . .
WORKDIR "/src/src/domain/core/Eatigo.Eatilink.Api"
RUN dotnet build "Eatigo.Eatilink.Api.csproj" -c Release -o /app/build

FROM build AS publish
RUN dotnet publish "Eatigo.Eatilink.Api.csproj" -c Release -o /app/publish

FROM base AS final
WORKDIR /app
COPY --from=publish /app/publish .
ENTRYPOINT ["dotnet", "Eatigo.Eatilink.Api.dll"]

CMD ASPNETCORE_URLS=http://*:$PORT dotnet Eatigo.Eatilink.Api.dll
```

Docker Compose

For mongo database and browser localhost port configuration.

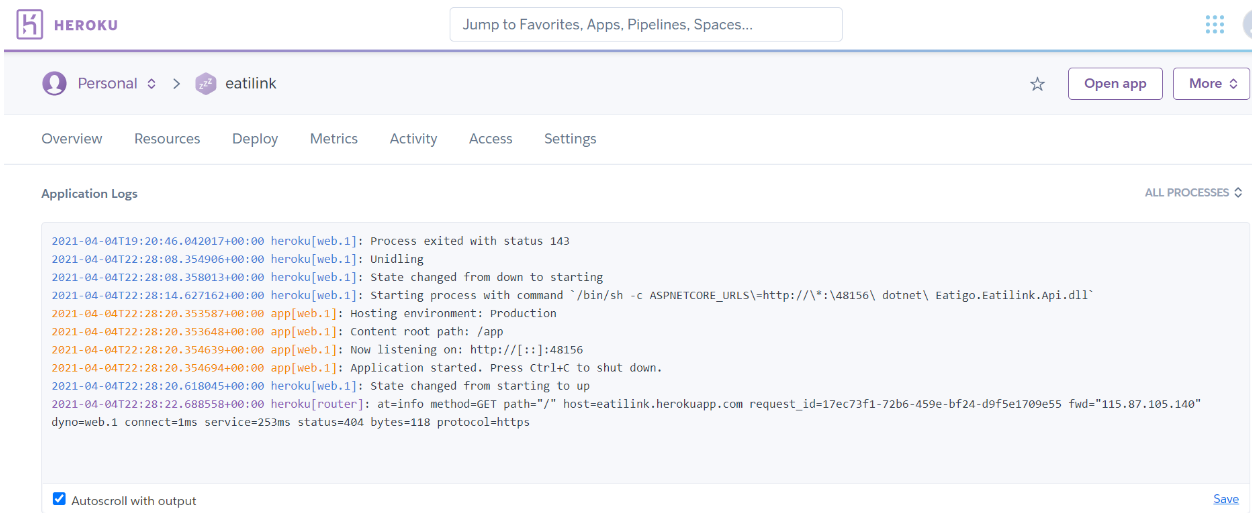


```
version: '3.1'

services:

  mongo:
    container_name: mongo
    image: mongo
    restart: always
    volumes:
      - ${WEBAPP_STORAGE_HOME}/site:/data/db
      #- ./data:/data/db
    ports:
      - "27017:27017"

  web:
        build: .
        ports:
            - "8000:80"
            - "44348:443"
        depends_on:...
        volumes:...
        links:...
```

Deployment

Published Version

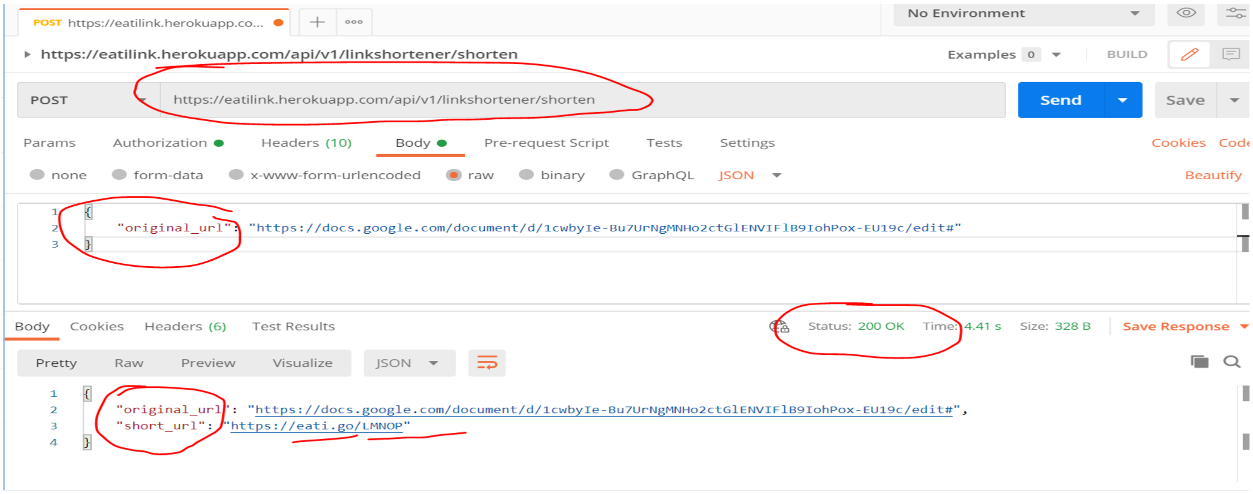
Platform	HEROKU
Publish Type	Container Registry
Published URL	https://eatilink.herokuapp.com/api/v1/linkshortener/shorten
Deployment Status	Running
Api Authorization Token	*eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyX2lkZW50aXR5IjoiSGFzYW4iLCJuYmYiOjE2MTcyNzk1MDAsImV4cCI6MTYxOTg3MTUwMCwiaWF0IjoxNjE3Mjc5NTAwfQ.I0W_jbhcUUCb3kQBjg3GB0TiJn7WEwFRnUiN9PCI4Lg*



```
2021-04-04T19:20:46.042017+00:00 heroku[web.1]: Process exited with status 143
2021-04-04T22:28:08.354906+00:00 heroku[web.1]: Unidling
2021-04-04T22:28:08.358013+00:00 heroku[web.1]: State changed from down to starting
2021-04-04T22:28:14.627162+00:00 heroku[web.1]: Starting process with command `/bin/sh -c ASPNETCORE_URLS\=http://\*:\48156\ dotnet\ Eatigo.Eatilink.Api.dll`
2021-04-04T22:28:20.353587+00:00 app[web.1]: Hosting environment: Production
2021-04-04T22:28:20.353648+00:00 app[web.1]: Content root path: /app
2021-04-04T22:28:20.354639+00:00 app[web.1]: Now listening on: http://[::]:48156
2021-04-04T22:28:20.354694+00:00 app[web.1]: Application started. Press Ctrl+C to shut down.
2021-04-04T22:28:20.618045+00:00 heroku[web.1]: State changed from starting to up
2021-04-04T22:28:22.688558+00:00 heroku[router]: at=info method=GET path="/" host=eatilink.herokuapp.com request_id=17ec73f1-72b6-459e-bf24-d9f5e1709e55 fwd="115.87.105.140"
dyno=web.1 connect=1ms service=253ms status=404 bytes=118 protocol=https
```

☑ Autoscroll with output

Let's try -



https://eatilink.herokuapp.com/api/v1/linkshortener/shorten

POST https://eatilink.herokuapp.com/api/v1/linkshortener/shorten

```
{
    "original_url": "https://docs.google.com/document/d/1cwbyIe-Bu7UrNgMNHo2ctGlENVIFlB9IohPox-EU19c/edit#"
}
```

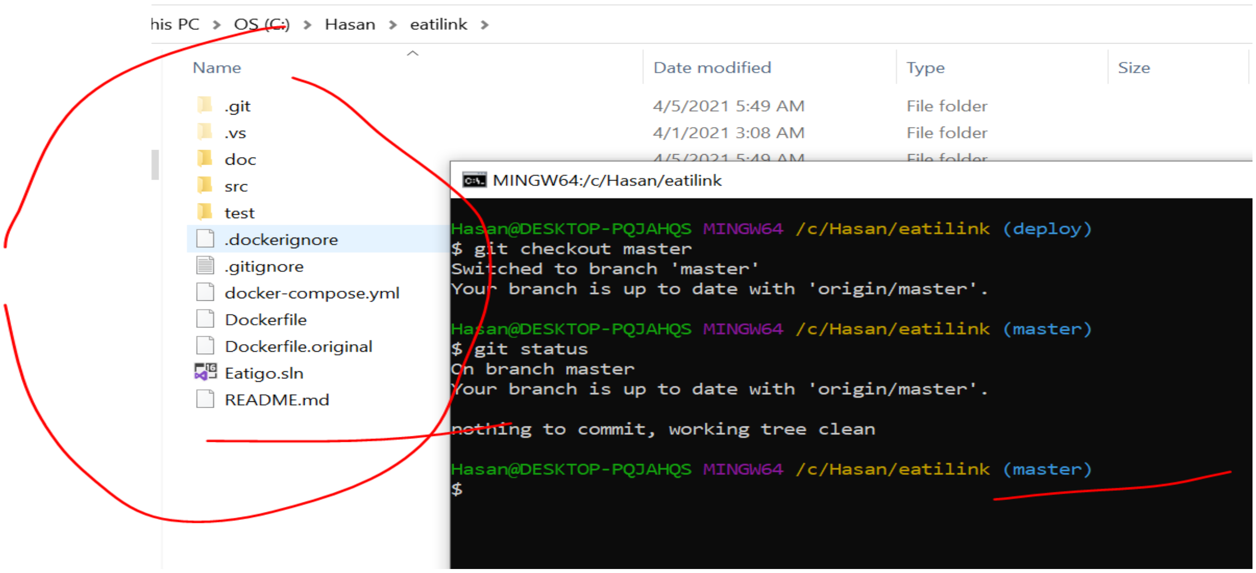
Status: 200 OK Time: 4.41 s Size: 328 B

```
{
    "original_url": "https://docs.google.com/document/d/1cwbyIe-Bu7UrNgMNHo2ctGlENVIFlB9IohPox-EU19c/edit#",
    "short_url": "https://eati.go/LMNOP"
}
```

Local Deployable Docker container

Step 1

Download or clone project from https://github.com/HasanShahjahan/eatilink/tree/master . Please make sure it is a master branch and we can also download it’s tag which is v1.0.

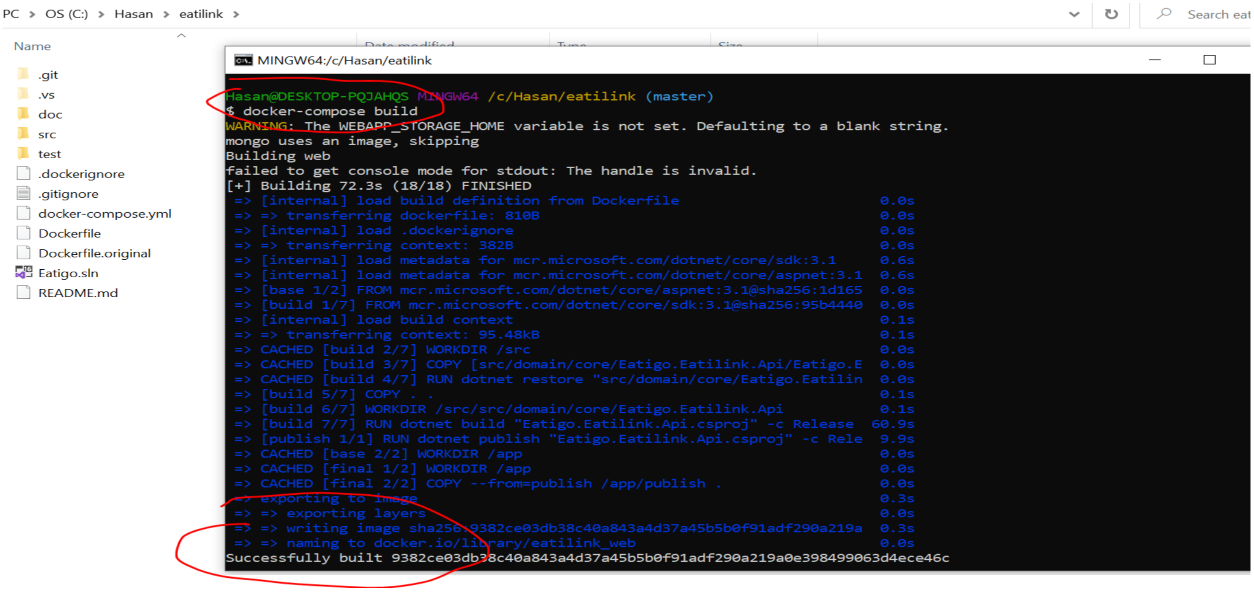

his PC > OS (C:) > Hasan > eatilink >
Name
Date modified
Type
Size
.git 4/5/2021 5:49 AM File folder
.vs 4/1/2021 3:08 AM File folder
doc 4/5/2021 5:49 AM File folder
src
test
.dockerignore
.gitignore
docker-compose.yml
Dockerfile
Dockerfile.original
Eatigo.sln
README.md
MINGW64:/c/Hasan/eatilink

```
Hasan@DESKTOP-PQJAHQS MINGW64 /c/Hasan/eatilink (deploy)
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.

Hasan@DESKTOP-PQJAHQS MINGW64 /c/Hasan/eatilink (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean

Hasan@DESKTOP-PQJAHQS MINGW64 /c/Hasan/eatilink (master)
$
```

Step 2

Write below command and enter -

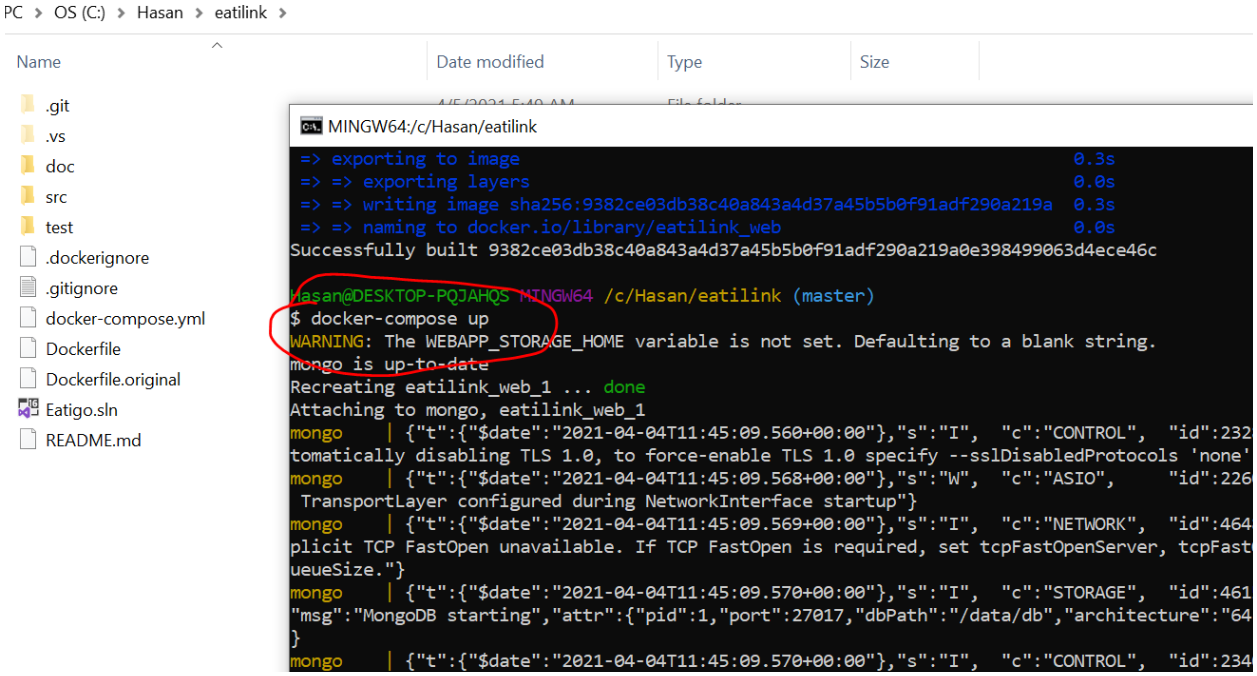
docker-compose build



```
Hasan@DESKTOP-PQJAHQS MINGW64 /c/Hasan/eatilink (master)
$ docker-compose build
WARNING: The WEBAPP_STORAGE_HOME variable is not set. Defaulting to a blank string.
mongo uses an image, skipping
Building web
failed to get console mode for stdout: The handle is invalid.
[+] Building 72.3s (18/18) FINISHED
 => [internal] load build definition from Dockerfile                        0.0s
 => => transferring dockerfile: 810B                                        0.0s
 => [internal] load .dockerignore                                           0.0s
 => => transferring context: 382B                                           0.0s
 => [internal] load metadata for mcr.microsoft.com/dotnet/core/sdk:3.1      0.6s
 => [internal] load metadata for mcr.microsoft.com/dotnet/core/aspnet:3.1   0.6s
 => [base 1/2] FROM mcr.microsoft.com/dotnet/core/aspnet:3.1@sha256:1d165  0.0s
 => [build 1/7] FROM mcr.microsoft.com/dotnet/core/sdk:3.1@sha256:95b4440  0.0s
 => [internal] load build context                                           0.1s
 => => transferring context: 95.48kB                                        0.1s
 => CACHED [build 2/7] WORKDIR /src                                         0.0s
 => CACHED [build 3/7] COPY [src/domain/core/Eatigo.Eatilink.Api/Eatigo.E  0.0s
 => CACHED [build 4/7] RUN dotnet restore "src/domain/core/Eatigo.Eatilin  0.0s
 => [build 5/7] COPY . .                                                    0.1s
 => [build 6/7] WORKDIR /src/src/domain/core/Eatigo.Eatilink.Api            0.1s
 => [build 7/7] RUN dotnet build "Eatigo.Eatilink.Api.csproj" -c Release  60.9s
 => [publish 1/1] RUN dotnet publish "Eatigo.Eatilink.Api.csproj" -c Rele  9.9s
 => CACHED [base 2/2] WORKDIR /app                                          0.0s
 => CACHED [final 1/2] WORKDIR /app                                         0.0s
 => CACHED [final 2/2] COPY --from=publish /app/publish .                   0.0s
 => exporting to image                                                      0.3s
 => => exporting layers                                                     0.0s
 => => writing image sha256:9382ce03db38c40a843a4d37a45b5b0f91adf290a219a  0.3s
 => => naming to docker.io/library/eatilink_web                             0.0s
Successfully built 9382ce03db38c40a843a4d37a45b5b0f91adf290a219a0e398499063d4ece46c
```

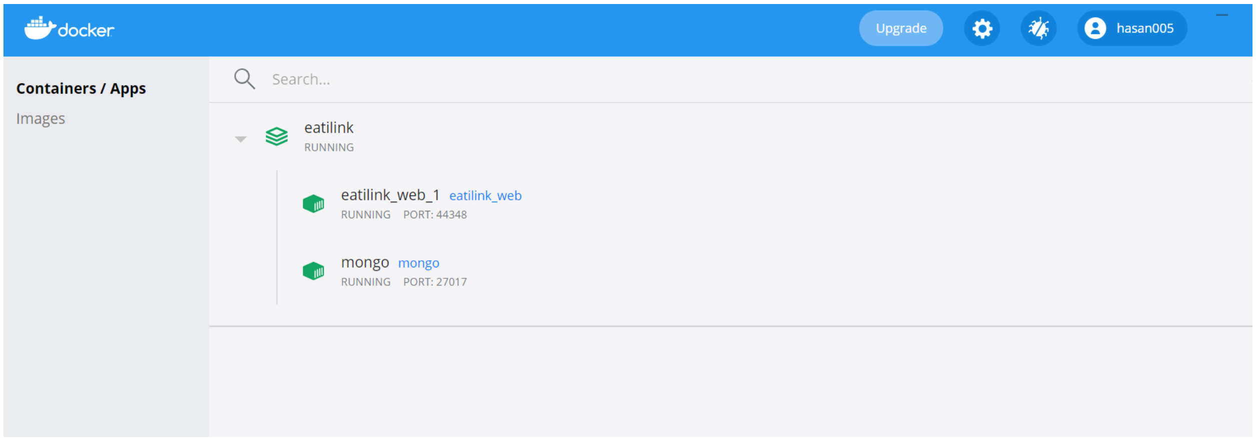
Step 3

Write below command and enter -
docker-compose up



```
 => exporting to image                                                0.3s
 => => exporting layers                                               0.0s
 => => writing image sha256:9382ce03db38c40a843a4d37a45b5b0f91adf290a219a  0.3s
 => => naming to docker.io/library/eatilink_web                       0.0s
Successfully built 9382ce03db38c40a843a4d37a45b5b0f91adf290a219a0e398499063d4ece46c

Hasan@DESKTOP-PQJAHQS MINGW64 /c/Hasan/eatilink (master)
$ docker-compose up
WARNING: The WEBAPP_STORAGE_HOME variable is not set. Defaulting to a blank string.
mongo is up-to-date
Recreating eatilink_web_1 ... done
Attaching to mongo, eatilink_web_1
mongo     | {"t":{"$date":"2021-04-04T11:45:09.560+00:00"},"s":"I",  "c":"CONTROL",  "id":232
tomatically disabling TLS 1.0, to force-enable TLS 1.0 specify --sslDisabledProtocols 'none'
mongo     | {"t":{"$date":"2021-04-04T11:45:09.568+00:00"},"s":"W",  "c":"ASIO",     "id":226
 TransportLayer configured during NetworkInterface startup"}
mongo     | {"t":{"$date":"2021-04-04T11:45:09.569+00:00"},"s":"I",  "c":"NETWORK",  "id":464
plicit TCP FastOpen unavailable. If TCP FastOpen is required, set tcpFastOpenServer, tcpFast
ueueSize."}
mongo     | {"t":{"$date":"2021-04-04T11:45:09.570+00:00"},"s":"I",  "c":"STORAGE",  "id":461
"msg":"MongoDB starting","attr":{"pid":1,"port":27017,"dbPath":"/data/db","architecture":"64
}
mongo     | {"t":{"$date":"2021-04-04T11:45:09.570+00:00"},"s":"I",  "c":"CONTROL",  "id":234
```

We are done. Our application is running inside docker containers.



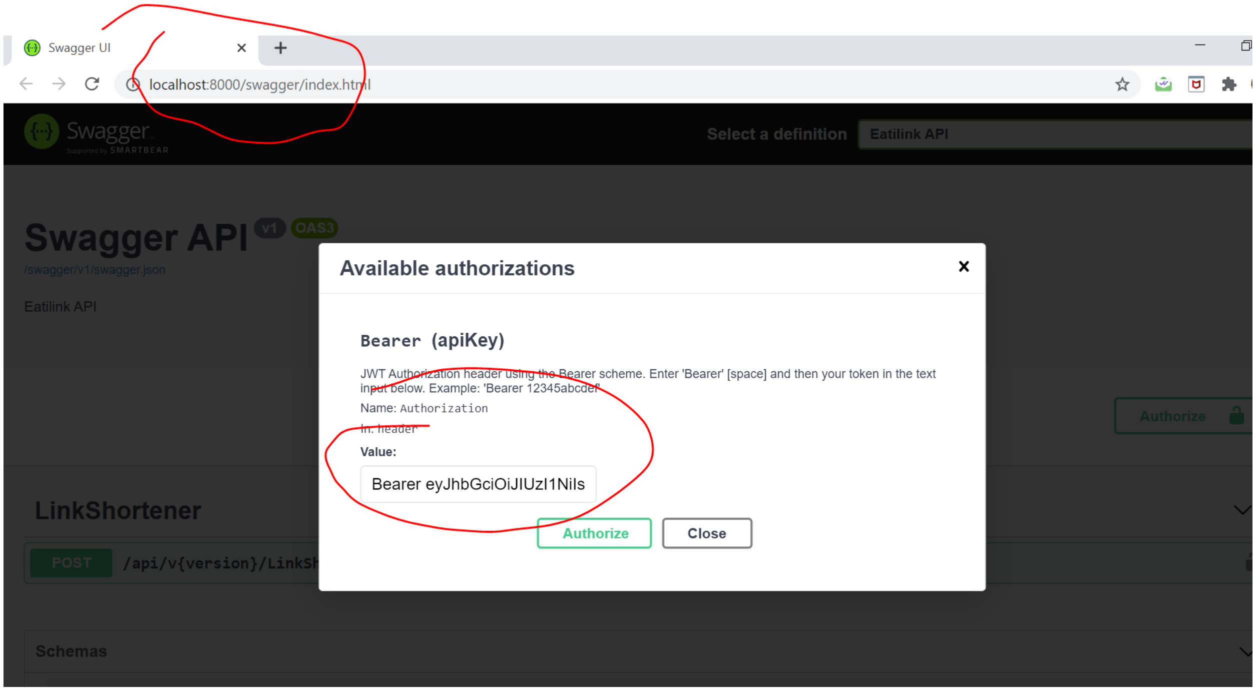
Step 4

As we have seen our port configuration inside ***docker-compose.yml, let's try through the browser.*** Eatilink service also has implemented swagger, let's try through. We can also try by postman.

Platform	localhost
Publish Type	Docker Container
Local URL	http://localhost:8000/swagger/index.html
Api Authorization Token	***Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyX2lkZW50aXR5IjoiSGFzYW4iLCJuYmYiOjE2MTcyNzk1MDAsImV4cCI6MTYxOTg3MTUwMCwiaWF0IjoxNjE3Mjc5NTAwfQ.I0W_jbhcUUCb3kQBjg3GB0TiJn7WEwFRnUiN9PCI4Lg***

Let's try -

Let's set authorization token



Available authorizations

Bearer (apiKey)

JWT Authorization header using the Bearer scheme. Enter 'Bearer' [space] and then your token in the text input below. Example: 'Bearer 12345abcdef'

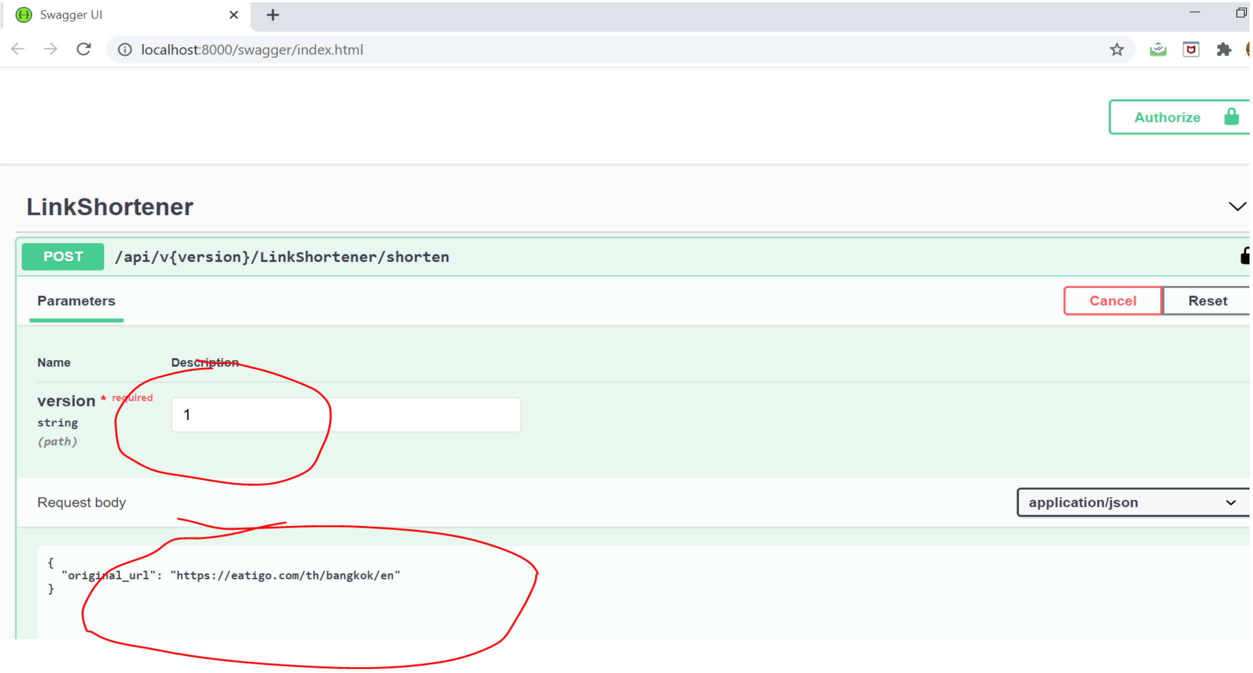
Name: Authorization

In: header

Value:

Bearer eyJhbGciOiJIUzI1NiIs

Let’s set version and request body



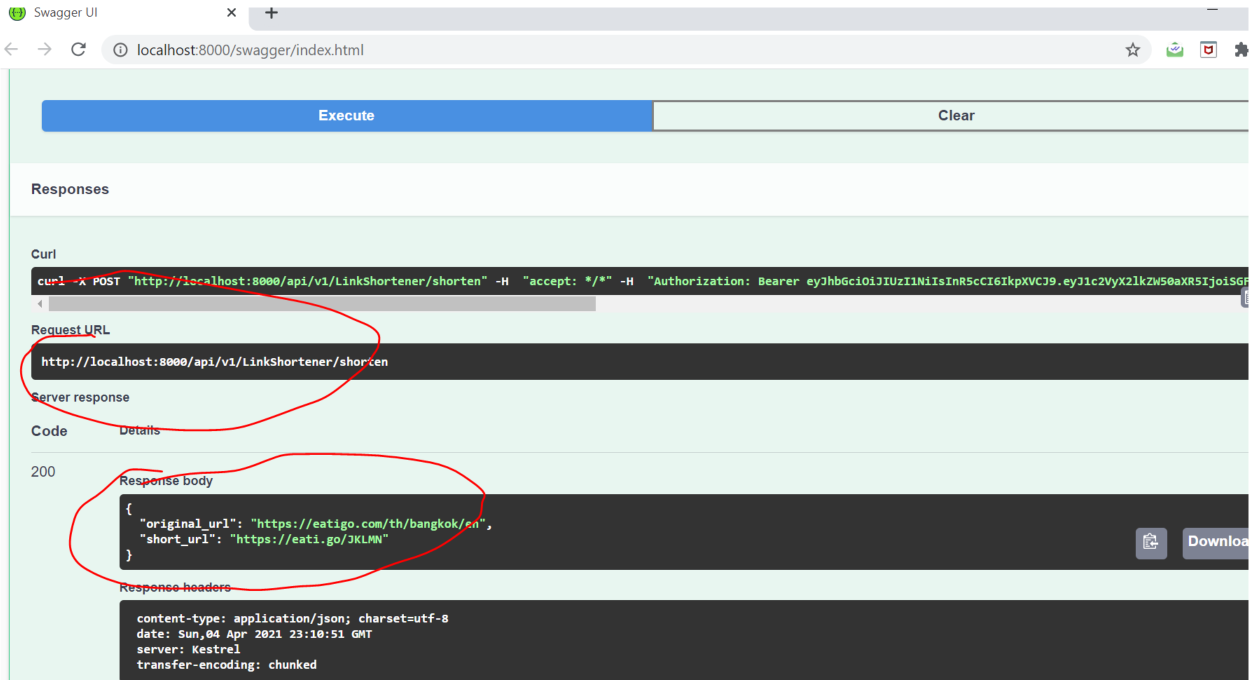
```
POST /api/v{version}/LinkShortener/shorten
```

version * required
string
(path)
1

Request body

```
{
  "original_url": "https://eatigo.com/th/bangkok/en"
}
```

Response Body



Curl

```
curl -X POST "http://localhost:8000/api/v1/LinkShortener/shorten" -H  "accept: */*" -H  "Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyX2lkZW50aXR5IjoiSGF
```

Request URL

```
http://localhost:8000/api/v1/LinkShortener/shorten
```

Server response

Code | Details
200

Response body

```
{
  "original_url": "https://eatigo.com/th/bangkok/en",
  "short_url": "https://eati.go/JKLMN"
}
```

Response headers

```
content-type: application/json; charset=utf-8
date: Sun,04 Apr 2021 23:10:51 GMT
server: Kestrel
transfer-encoding: chunked
```

Cheers !