QR CODE GENERATOR

A Course Based Project Submitted in Partial Fulfilment of the Requirement for the Award of the degree of

BACHELOR OF TECHNOLOGY

COMPUTER SCIENCE AND ENGINEERING

Submitted by

SARAVANA SAI GARAPATI   21071A6254

DEPARTMENT OF CSE-CYS, DS & (AI &DS)

VALLURUPALLI NAGESWARARAO VIGNANA JYOTHI

INSTITUTE OF ENGINEERING AND TECHNOLOGY

(An Autonomous Institute, NAAC Accredited With 'A++' Grade, NBA Accredited, Approved by AICTE, New Delhi, Affiliated to JNTUH)

VALLURUPALLI NAGESWARARAO VIGNANA JYOTHI

INSTITUTE OF ENGINEERING AND TECHNOLOGY

(An Autonomous Institute)

CERTIFICATE

This is to Certify that Saravana Sai Garapati (21071A6254) has successfully completed their project work at CSE CYS, DS & (AI & DS) Department of VNRVJIET, Hyderabad entitled "QR CODE GENERATOR" in partial fulfilment of the requirements forthe award of the Bachelor of Technology degree during the Academic year 2022-2023

Project Guide   Head of Department

Mrs. E. Lalitha   Dr. M. Rajasekhar
Assistant Prof. and Internal Guide   Prof. and Head
Dept. of CSE-CYS, DS and AI&DS   Dept. of CSE-CYS, DS and AI&DS
VNRVJIET   VNRVJIET

DECLARATION

This is to certify that the project work entitled "QR CODE GENERATOR" submitted in VNR Vignana Jyothi Institute of Engineering & Technology in partial fulfilment of requirement for the award of Bachelor of Technology in Computer Science and Engineering. It is a Bonafide report of the work   carried out by us under the guidance and supervision of Mrs.E.Lalitha (Assistant Professor), Department of CSE-CYS,DS,AI&DS, VNRVJIET. To the best of our knowledge, this report has not been submitted in any form to any university or institution for the award of any degree or diploma.

Saravana Sai Garapati

21071A6254

CSE-CYS

ACKNOWLEDGEMENT

Behind every achievement lies the heartfelt gratitude to those who activated in completing the project. To them we lay the words of gratitude within us.

We are indebted to our venerable principal Dr. C.D. NAIDU for this inflicting devotion, which led us to complete this project. The support, encouragement given by him and his motivation led us to complete the project.

We express our sincere thanks to internal guide Mrs. E. Lalitha and also Head of the Department Dr. M. RAJA SHEKHAR for having provided us a lot of facilities to undertake the project work and guide us to complete the project.

We take the opportunity to express thanks to our faculty of the Dept. of COMPUTER SCIENCE AND ENGINEERING-CYBER SECURITY and remaining members of our college VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY who extended their valuable support in helping us to complete the project in time.

Saravana Sai Garapati (21071A6254)

ABSTRACT

The QR Code Generator Python project is an application that generates Quick Response (QR) codes. A quick response (QR) code is a barcode that a digital device can easily scan. In contrast to barcodes, which need a beam of light to bounce off the parallel lines, QR codes may be scanned digitally by devices like smartphones. QR codes are used in cryptocurrency systems to enable digital payments, such as when displaying a Bitcoin address. QR codes are also often used to communicate site URLs to mobile devices. The project uses the Pythonprogramming language and the pyqrcode library to generate QR codes thatcan be saved as images or printed. The user inputs the information they wantto encode into the QR code, and the program generates the code in real-time. The project is designed to be easy to use and customizable, allowing users toadjust the size and color of the generated QR code. Overall, this project provides a useful tool for creating QR codes quickly and easily.

CONTENTS

INTRODUCTION

The QR Code Generator Python project is a useful tool for generating QR codes. QR codes are two-dimensional barcodes that can be scanned by a smartphone camera to quickly access information such as website URLs, contact information, and more. QR codes have become increasingly popular in recent years due to their convenience and ease of use.

The project uses the Python programming language and the PyQRCode library to generate QR codes. Python is a popular programming language that is widely used for data analysis, machine learning, and web development. PyQRCode is a library that provides an easy way to generate QR codes in Python. It is a pure Python library, which means that it does not require any external dependencies.

The QR Code Generator Python project is designed to be easy to use and customizable. The user inputs the information they want to encode into the QR code, and the program generates the code in real-time. The generated QR code can be saved as an image or printed. The project allows users to adjust the size and color of the QR code to their preferences.

One of the key benefits of the QR Code Generator Python project is that it can save time and effort. Creating QR codes manually can be a time-consuming and error-prone process, especially for larger amounts of information. With this project, users can generate QR codes quickly and easily, without the need for specialized software or technical skills.

Another benefit of the QR Code Generator Python project is that it can be used in a variety of contexts. QR codes can be used for marketing, educational, and informational purposes. For example, businesses can use QR codes to direct customers to their website or social media pages, while teachers can use QR codes to provide students with additional resources or assignments.

LIBRARIES

QRcode: It is an external python library used to generate QR codes. It can be installed using the below command:

```
pip install qrcode
```

Usage:
To generate a QR code using QRcode library, you first need to import the qrcode module. Here's an example:

```
import qrcode
```

Once you have imported the module, you can use the make() method to create a QR code object. The make() method takes the data to be encoded as an argument. Here's an example:

```
qr = qrcode.make('Hello, World!')
```

This will create a QR code object that encodes the text "Hello, World!". You can then save the QR code image to a file using the save() method. Here's an example:

```
qr.save('hello_world.png')
```

This will save the QR code image to a file named hello_world.png.

Methods:

Here are some of the methods provided by the QRcode library:

make(data): Creates a QR code object that encodes the given data.

add_data(data): Adds more data to an existing QR code object.

make_image(fill_color='black', back_color='white'): Creates a PIL image object of the QR code.

Pillow: It is a lightweight image processing tool that aid in editing, creating, and saving images. Pillow supports many image file formats including BMP, PNG, JPEG, and TIFF. It can be installed using the below command:

pip install Pillow

Some of the most commonly used methods of the Image module in Pillow include:

open(): This method is used to open an image file and returns an Image object. It takes the path of the image file as an argument.

save(): This method is used to save an image to a file. It takes the path of the output file as an argument and an optional format parameter.

CODE FOR QR GENERATOR

```python
1    import qrcode
2    from PIL import Image
3
4    Logo_link = "C:\\Users\\Mohan\\Downloads\\vnr.png"
5    logo = Image.open(Logo_link)
6
7    # taking base width
8    basewidth = 100
9
10   # adjust image size
11   wpercent = (basewidth/float(logo.size[0]))
12   hsize = int((float(logo.size[1])*float(wpercent)))
13   logo = logo.resize((basewidth, hsize), Image.ANTIALIAS)
14   QRcode = qrcode.QRCode(error_correction=qrcode.constants.ERROR_CORRECT_H)
15
16   # taking url or text
17   url=input('Enter the text or paste the link to convert it into a QR Code : \n')
18
19   # adding URL or text to QRcode
20   QRcode.add_data(url)
21
22   # generating QR code
23   QRcode.make()
24
25   # taking color name from user
26   QRcolor = 'Brown'
27
28   # adding color to QR code
29   QRimg = QRcode.make_image(fill_color=QRcolor, back_color="white").convert('RGB')
30
31   # set size of QR code
32   pos = ((QRimg.size[0] - logo.size[0]) // 2,(QRimg.size[1] - logo.size[1]) // 2)
33   QRimg.paste(logo, pos)
34
35   # save the QR code generated
36   QRimg.save("QR.png")
37
38   img = Image.open(r"QR.png")
39   img.show()
```

Description of Code:

This is a Python script that generates a QR code with a logo added to it. Here's how it works:

The script imports two modules: qrcode and Image from PIL.

The script prompts the user to provide a path to an image file (in this case, the file is named "python.png") that will be used as the logo to be added to the QR code.
The script sets a base width of 150 pixels for the logo and resizes the logo to fit within that width while maintaining its aspect ratio.

The script creates a qrcode.QRCode object and adds data to the QR code using a URL provided by the user.The script prompts the user to provide a color name to be used for the QR code (in this case, the color name is "Brown").

The script creates the QR code image using the qrcode.QRCode.make_image() method, specifying the fill color for the QR code as the user-provided color name and the background color as white. The resulting image is then converted to RGB format.

The script calculates the position at which to paste the logo onto the QR code by centering the logo within the QR code image.

The script pastes the resized logo onto the QR code image at the calculated position using the Image.paste() method.

Finally, the script prints a displays the generated QR code.
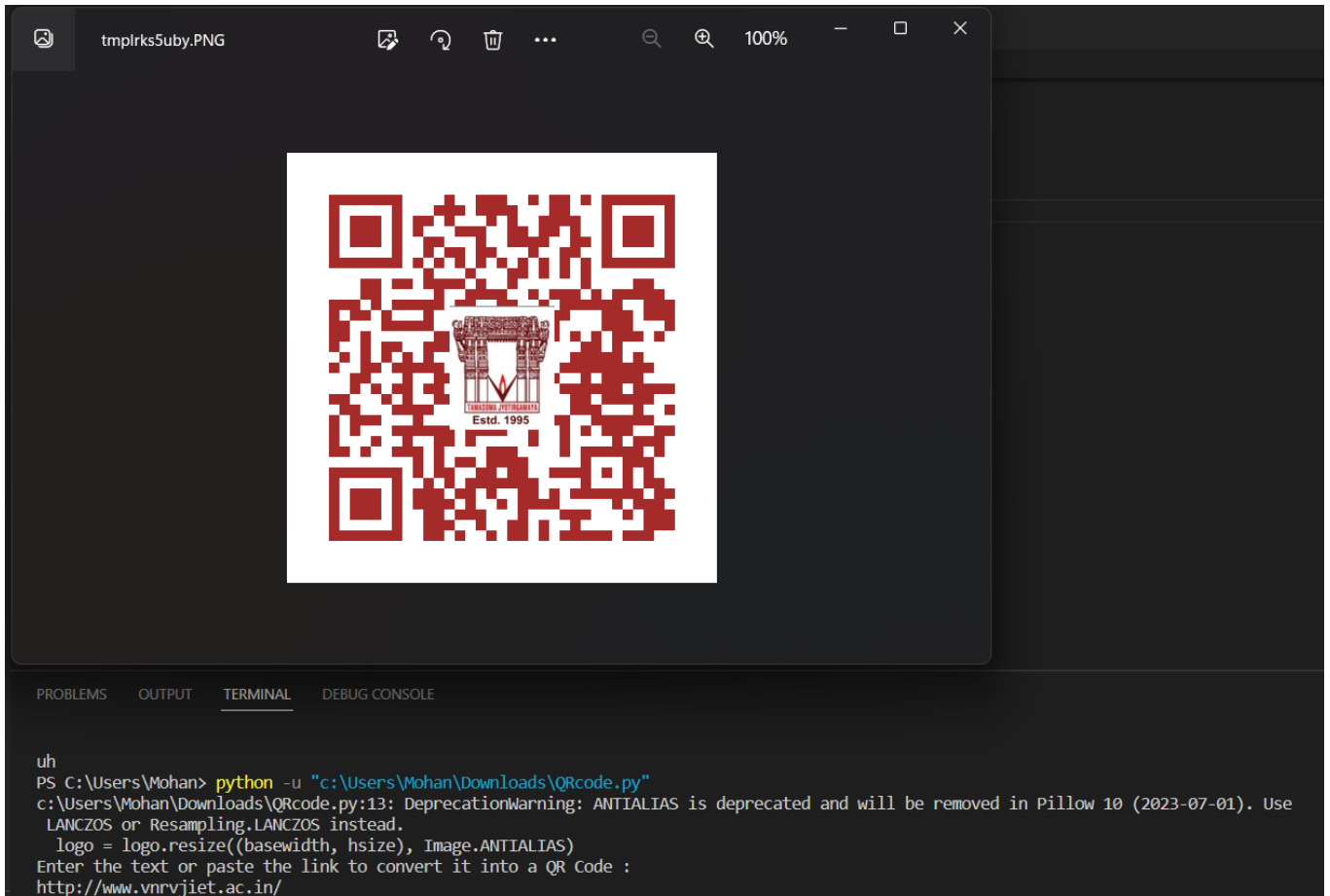
CONCLUSION

Overall, this script demonstrates how to use the qrcode and Image modules from Pillow to generate a customized QR code image with a logo added to it.

The project is successful in generating a customized QR code image with a logo added to it, which can be saved to a file or included in a webpage. Further work could include enhancing the user interface or adding additional features.

In addition to being saved to a file, the generated QR code image could also be included in a webpage using HTML. This would allow the QR code to be displayed on a website or shared online. To include the image in a webpage, you would need to specify the file path to the generated QR code image in the source attribute of an <img> tag.

The generated image could be useful in a variety of applications, such as marketing or branding campaigns, or for sharing website links. However, potential areas for improvement could include enhancing the user interface, adding additional customization options, or optimizing the code for performance or scalability.

Overall, the project is a useful demonstration of how Python can be used to generate customized QR code images with logos, and could serve as a starting point for further development and customization.