QR Code Generator Web Application

Introduction

This project is a web application built in the Go programming language that allows users to generate, store, and retrieve QR codes based on user-provided data.The primary features of the project include generating QR codes, storing them in a SQLite database, viewing saved QR codes, and tracking their generated QR.

Tech Stack Used

- **Backend:** Golang
- **Database:** SQLite
- **Frontend:** HTML, CSS
- **Development Environment:** Visual Studio Code

Project Features

1. QR Code Generation

- Users can enter any text or URL in the application, and a corresponding QR code is generated.
- The application utilizes the Boombuler Barcode package to create high-quality QR codes.

2. QR Code Storage

- Generated QR codes data are stored in a SQLite database along with the associated text data.

3. Retrieving Stored QR Codes

- Users can view a list of all previously generated QR codes.
- Clicking on a QR code link retrieves and displays the corresponding QR code.

4. Scan Count

- Tracks the count of QR code scan and increments it by one whenever it gets scaned.

Project Flow

1. **Home Page:**
    - Displays a form for users to input text data to generate a QR code.
2. **QR Code Generation:**
    - The application generates the QR code and stores it in the database.
    - The generated QR code is displayed to the user.
3. **Retrieve Page:**
    - Shows a list of stored QR codes.
4. **View QR Code Page:**
    - Displays the QR code of any previously generated QR data.

Third-Party Package Usage

Boombuler Barcode Package

- Used for generating QR codes and scaling them to appropriate sizes.
- Provides efficient QR code encoding and customization.

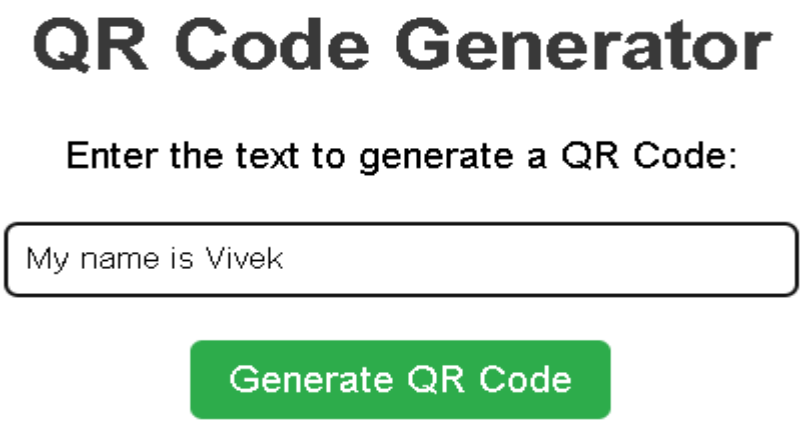
SQLite Integration

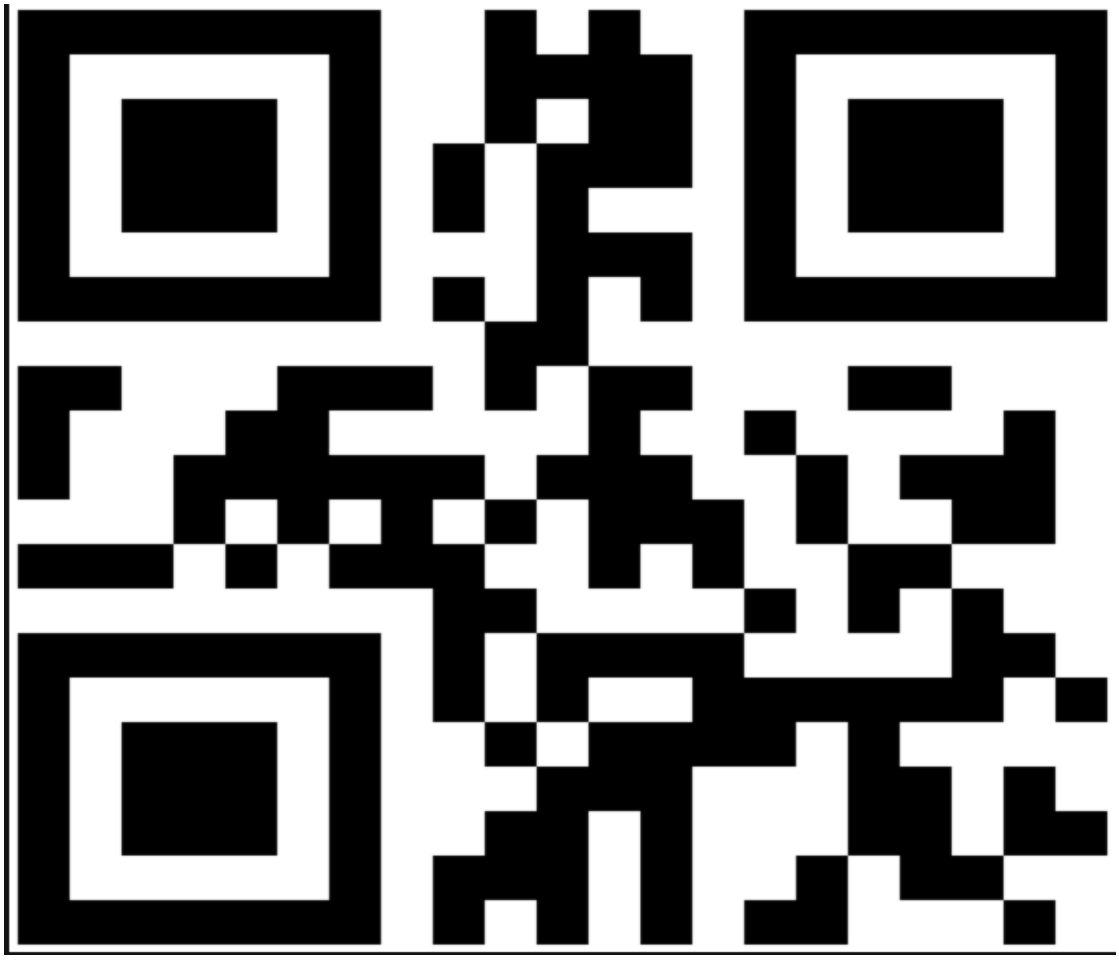
- The `go-sqlite3` driver is used to interact with the SQLite database.
- It provides a lightweight and embedded database solution for storing QR code data.

Future Enhancements

1. **User Authentication:**
   - Add user login and authentication to personalize QR code storage.
2. **Download QR Codes:**
   - Allow users to download generated QR codes as image files.
3. **Data Analytics:**
   - Provide analytics on QR code scan trends.
4. **API Integration:**
   - Expose RESTful APIs for external applications to generate and retrieve QR codes.

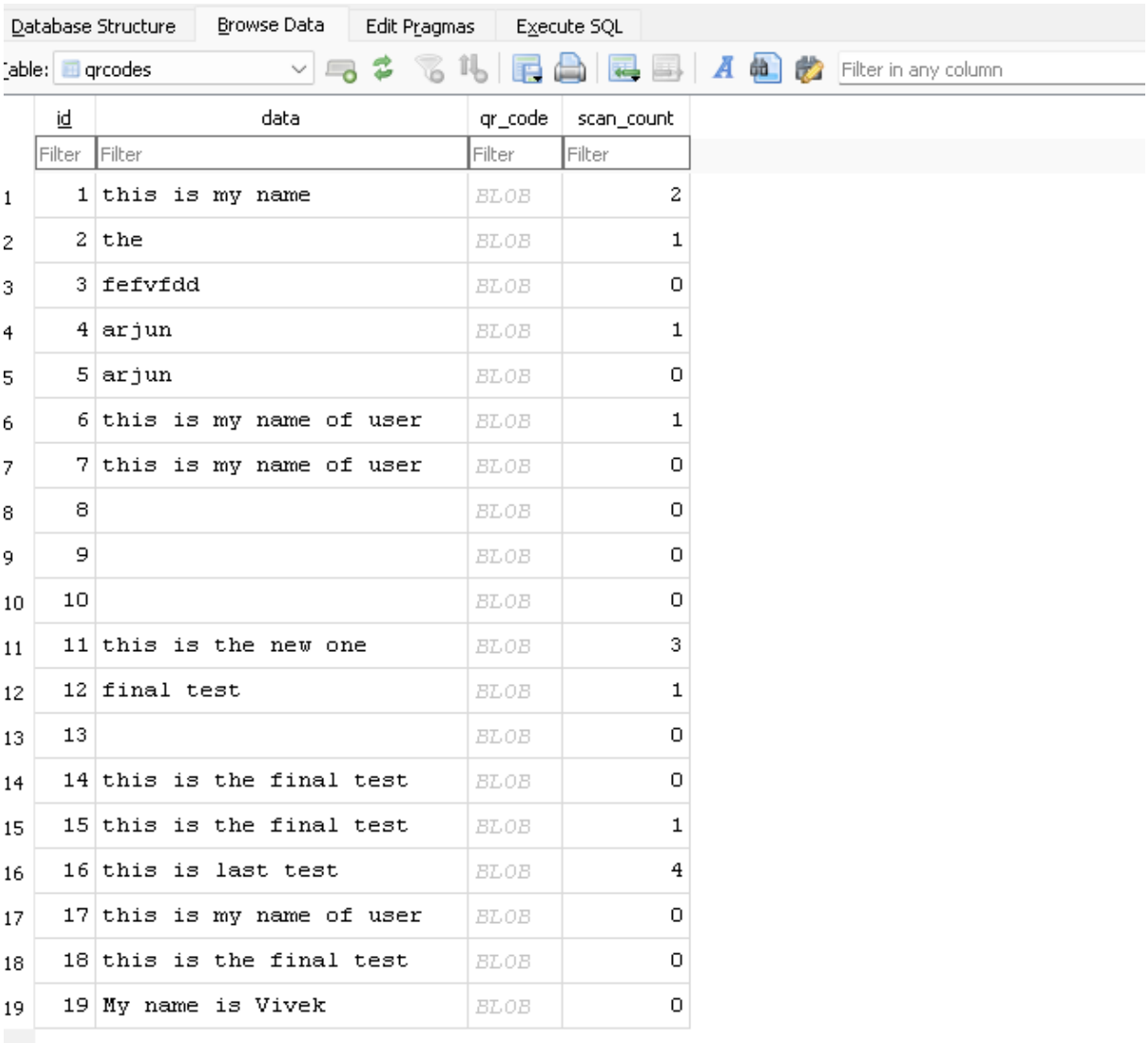
Snapshots





Stored QR Codes

- this is my name
- the
- fefvfdd
- arjun
- arjun
- this is my name of user
- this is my name of user
-
-
-
- this is the new one
- final test
-
- this is the final test
- this is the final test
- this is last test
- this is my name of user
- this is the final test
- My name is Vivek



	id	data	qr_code	scan_count
	Filter	Filter	Filter	Filter
1	1	this is my name	BLOB	2
2	2	the	BLOB	1
3	3	fefvfdd	BLOB	0
4	4	arjun	BLOB	1
5	5	arjun	BLOB	0
6	6	this is my name of user	BLOB	1
7	7	this is my name of user	BLOB	0
8	8		BLOB	0
9	9		BLOB	0
10	10		BLOB	0
11	11	this is the new one	BLOB	3
12	12	final test	BLOB	1
13	13		BLOB	0
14	14	this is the final test	BLOB	0
15	15	this is the final test	BLOB	1
16	16	this is last test	BLOB	4
17	17	this is my name of user	BLOB	0
18	18	this is the final test	BLOB	0
19	19	My name is Vivek	BLOB	0

Conclusion

This project successfully demonstrates the development of a functional QR Code Generator web application using the Go programming language. It efficiently handles QR code generation, storage, and retrieval while maintaining a clean and intuitive interface.The use of a lightweight SQLite database and the Boombuler Barcode package makes it efficient and scalable for small to medium-scale projects.