

iOS Programmierung

- QR-Code App
 - QR-Code Generator

QR-Code App

QR-Code Generator

Hier ist ein Beispiel für eine iOS-App, die als QR-Code-Generator dient:

```
import UIKit
import CoreImage

class ViewController: UIViewController {

    @IBOutlet weak var textField: UITextField!
    @IBOutlet weak var imageView: UIImageView!

    @IBAction func generateButtonTapped(_ sender: Any) {
        guard let text = textField.text, !text.isEmpty else { return }

        let data = text.data(using: .ascii, allowLossyConversion: false)
        let filter = CIFilter(name: "CIQRCodeGenerator")
        filter?.setValue(data, forKey: "inputMessage")

        guard let outputImage = filter?.outputImage else { return }

        let scaleX = imageView.frame.size.width / outputImage.extent.size.width
        let scaleY = imageView.frame.size.height / outputImage.extent.size.height

        let transformedImage = outputImage.transformed(by: CGAffineTransform(scaleX: scaleX, y:
scaleY))

        imageView.image = UIImage(ciImage: transformedImage)
    }
}
```

Diese App enthält ein Textfeld und ein Bild, in dem der QR-Code angezeigt wird. Der Benutzer gibt den Text ein, den er in den QR-Code codieren möchte, und drückt auf den "Generate"-Button. Die App verwendet dann den Core Image-Filter "CIQRCodeGenerator", um aus dem eingegebenen Text

einen QR-Code zu erstellen, und zeigt den QR-Code im Bild an.

- Sie können das Textfeld und das Bild über den Interface Builder oder programmgesteuert hinzufügen.
- Sie können die CIQRCodeGenerator-Klasse verwenden, um aus jedem beliebigen Text einen QR-Code zu erstellen. Sie können auch andere Core Image-Filter verwenden, um verschiedene Arten von Barcodes zu erstellen.
- Sie können das Aussehen des QR-Codes anpassen, indem Sie verschiedene Einstellungen für den Filter festlegen, z.B. die Korrektorebene oder die Eingabekorrektur.
- Sie können den QR-Code auch in anderen Formaten speichern, z.B. als PNG- oder JPEG-Bild, indem Sie das Bild in einen UIImage- oder UIImage-Datentyp konvertieren und dann die entsprechenden APIs verwenden, um das Bild zu speichern. Sie könnten zum Beispiel einen Button hinzufügen, der dem Benutzer ermöglicht, den QR-Code zu speichern oder zu teilen. Hier ist ein Beispiel für eine Funktion, die den QR-Code als PNG-Bild speichern könnte:

```
@IBAction func saveButtonTapped(_ sender: Any) {
    guard let image = imageView.image else { return }

    let activityViewController = UIActivityViewController(activityItems: [image],
applicationActivities: nil)
    present(activityViewController, animated: true)
}
```

Diese Funktion zeigt einen "Aktivitätsansicht"-Controller an, der dem Benutzer verschiedene Optionen bietet, um das Bild zu speichern oder zu teilen. Der Benutzer kann zum Beispiel das Bild in seine Fotos-App speichern oder es über eine Nachrichten-App oder eine andere App teilen.