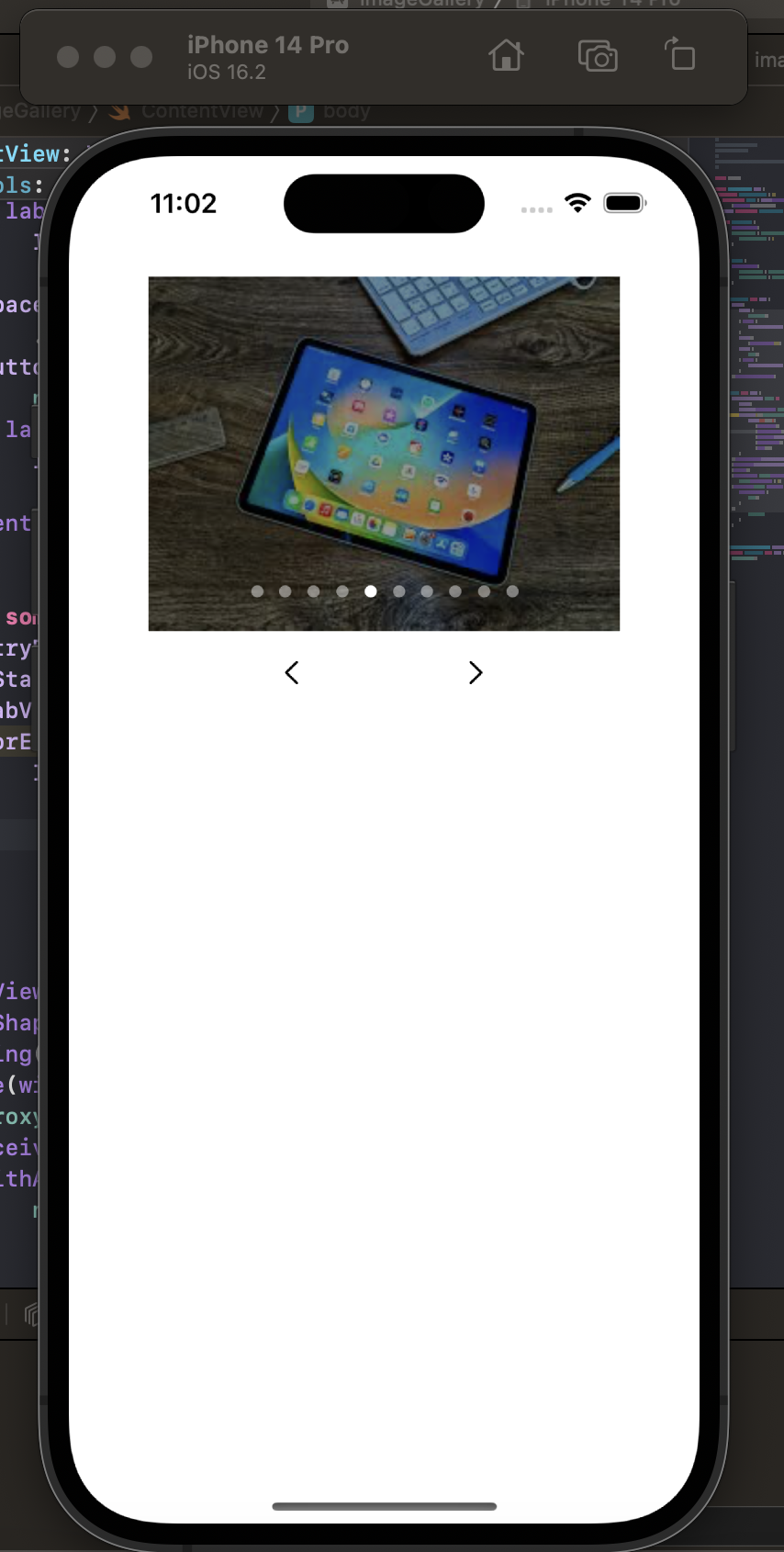
Nandaram Prashanth Kumar

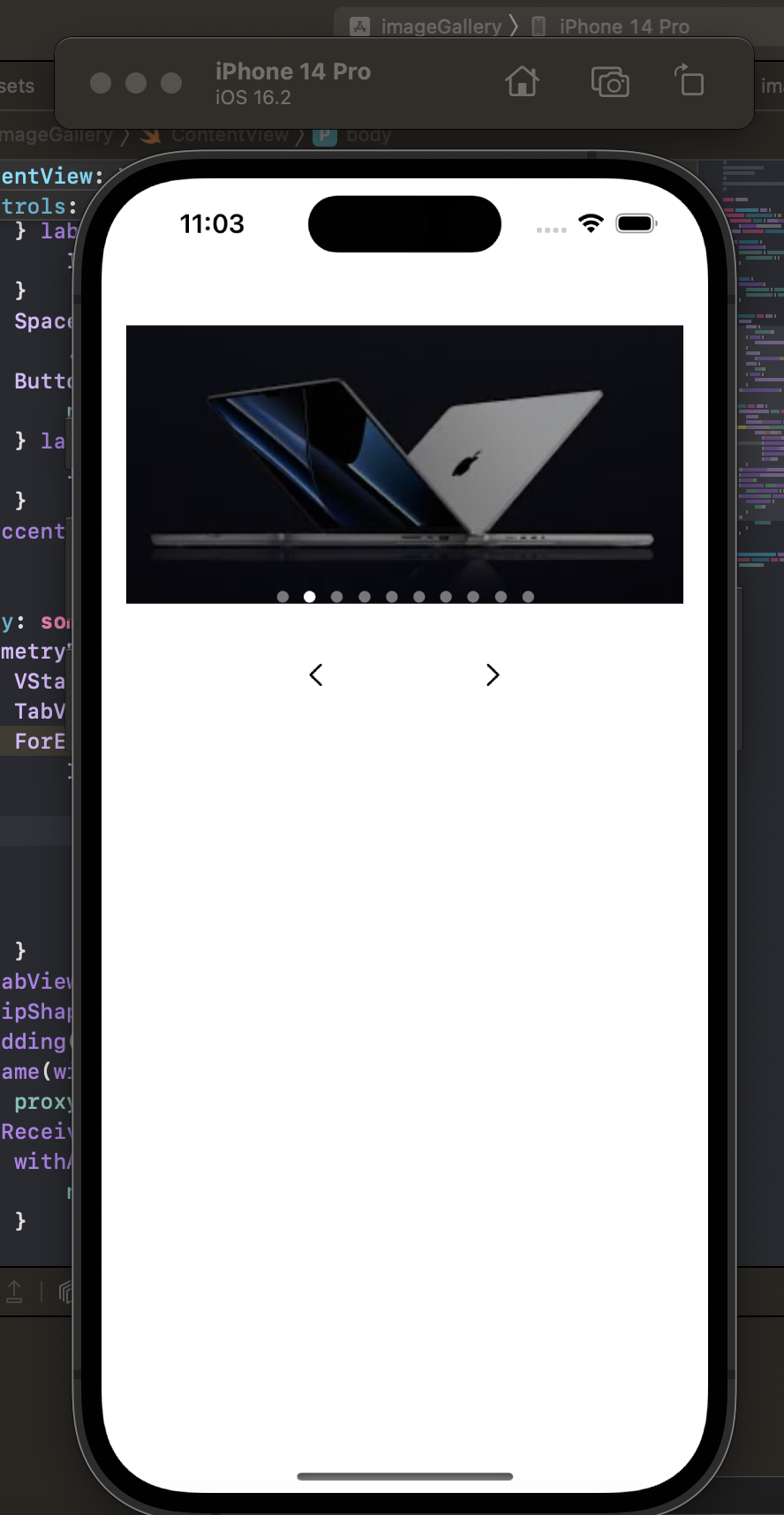
C0871509

**Option I - 30%**

Load 10 different images in your iOS application and create a simple user interface so when application loads it shows the first image. Proper navigation should be provided to navigate to the list of images and when you reach the last image, the first one is displayed. In similar fashion, if you go back on the list and you are already in the first image, the last one is shown (swiping left or right).

If the user does not swipe at all then the application automatically moves to the next image every 3 seconds (slide show). If the user taps in an image the timer stops, and the user can enjoy viewing that image for as long as they want. Swiping still works the same way as option two. To restart the slide show, user should double-tap on the image.





**import** SwiftUI

**@main**

**struct** GalleryApp: App {

**var** body: **some** Scene {

WindowGroup {

ContentView()

}

//

// ContentView.swift

// imageGallery

//

// Created by Prashanth on 2023-01-16.

//

**import** SwiftUI

**struct** ContentView: View {

**private** **var** numberOfImages = 10

**private** **let** timer = Timer.publish(every: 3, on: .main, in: .common).autoconnect()

@State **private** **var** currentIndex = 0

**func** previous() {

withAnimation{

currentIndex = currentIndex > 0 ? currentIndex - 1 : numberOfImages - 1

}

}

**func** next() {

withAnimation{

currentIndex = currentIndex <

numberOfImages ? currentIndex + 1 : 0

}

}

**var** controls: **some** View {

HStack{

Button {

previous()

} label: {

Image(systemName: "chevron.left")

}

Spacer()

.frame(width:100)

Button {

next()

} label: {

Image(systemName: "chevron.right")

}

}.accentColor(.primary)

}

**var** body: **some** View {

GeometryReader{ proxy **in**

VStack{

TabView(selection: $currentIndex) {

ForEach(0..<numberOfImages) { num **in**

Image("\(num)")

.resizable()

.scaledToFill()

.overlay(Color.black.opacity(0.4))

.scaledToFit()

.tag(num)

}

}.tabViewStyle(PageTabViewStyle())

.clipShape(RoundedRectangle(cornerRadius: 5))

.padding()

.frame(width: proxy.size.width, height: proxy.size.height / 3)

.onReceive(timer, perform: { \_ **in**

withAnimation {

next()

}

})

controls

}

}

}

}

**struct** ContentView\_Previews: PreviewProvider {

**static** **var** previews: **some** View {

ContentView()

}

}

**Part II - 70%**

Create a UI which user can add:

* Name
* Tel number
* Image

All data should be added to the db (core data/sqlite db)

User should be able to Edit (update/delete) any data

A screenshot of a phone

Description automatically generated with low confidenceA screenshot of a cell phone

Description automatically generated with medium confidence

}

} //

// ViewController.swift

//

// Created by Prashanth Nandaram

**import** UIKit

**class** ViewController: UIViewController, UINavigationControllerDelegate, UIImagePickerControllerDelegate {

**@IBOutlet** **weak** **var** saveImageView: UIImageView!

**@IBOutlet** **weak** **var** name: UITextField!

**@IBOutlet** **weak** **var** mobileNumber: UITextField!

**@IBOutlet** **weak** **var** status: UILabel!

**override** **func** viewDidLoad() {

**super**.viewDidLoad()

navigationItem.rightBarButtonItem = UIBarButtonItem(barButtonSystemItem: .camera, target: **self**, action: **#selector**(cameraButtonPressed))

saveImageView.layer.cornerRadius = 12

}

**@objc** **func** cameraButtonPressed() {

**let** picker = UIImagePickerController()

picker.delegate = **self**

picker.allowsEditing = **true**

picker.sourceType = .photoLibrary

present(picker, animated: **true**)

}

**func** imagePickerController(\_ picker: UIImagePickerController, didFinishPickingMediaWithInfo info: [UIImagePickerController.InfoKey : **Any**]) {

**guard** **let** userPickedImage = info[.editedImage] **as**? UIImage **else** { **return** }

saveImageView.image = userPickedImage

picker.dismiss(animated: **true**)

}

**@IBAction** **func** saveImageButtonPressed(\_ sender: UIButton) {

**if** **let** imageData = saveImageView.image?.pngData() {

DataBaseHelper.shareInstance.saveImage(data: imageData)

status.text = "Data saved"

}

}

}

//

// DataBaseHelper.swift

// Created by Prashanth Nandaram on 2023-01-16

//

**import** UIKit

**import** CoreData

**class** DataBaseHelper {

**static** **let** shareInstance = DataBaseHelper()

**let** context = (UIApplication.shared.delegate **as**! AppDelegate).persistentContainer.viewContext

**func** saveImage(data: Data) {

**let** imageInstance = Image(context: context)

imageInstance.img = data

**do** {

**try** context.save()

print("Image is saved")

} **catch** {

print(error.localizedDescription)

}

}

}