Login Page View Controller File:

**import** UIKit

**var** userName = "Vishnu"

**var** updatepassword = "Vishnu"

**class** loginPage: UIViewController

{

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

**super**.viewDidLoad()

}

**@IBOutlet** **weak** **var** label1: UILabel!

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

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

**func** RestartPage()

{

uname.text

pwd.text

}

**@IBAction** **func** forgotPass(\_ sender: **Any**)

{

**let** vc = storyboard?.instantiateViewController(withIdentifier: "cp2") **as**? changePasswordPage

**self**.navigationController?.pushViewController(vc!, animated: **true**)

}

**@IBAction** **func** senddata(\_ sender: UIButton)

{

**let** vc = storyboard?.instantiateViewController(withIdentifier: "cp3") **as**? changePassword1

**if** uname.text == userName && pwd.text == updatepassword

{

**let** vc1 = storyboard?.instantiateViewController(withIdentifier: "wp") **as**! welcomePage

**self**.navigationController?.pushViewController(vc1, animated: **true**)

vc1.dataString = "Welcome \(uname.text!)"

}

**else** **if** uname.text != userName && pwd.text != updatepassword

{

**let** alert = UIAlertController(title: "User credentials are invalid", message: "Please Try Again", preferredStyle: .alert)

**let** restart = UIAlertAction(title: "Press Here To Go Back To Login Page", style: .default, handler: {action **in** **self**.RestartPage()})

alert.addAction(restart)

present(alert , animated: **true**,completion: **nil**)

}

**else**

{

**if** uname.text != userName

{

**let** alert = UIAlertController(title: "userName is Invalid", message: "Please Try Again", preferredStyle: .alert)

**let** restart = UIAlertAction(title: "Press Here To Go Back To Login Page", style: .default, handler: {action **in** **self**.RestartPage()})

alert.addAction(restart)

present(alert , animated: **true**,completion: **nil**)

}

**else** **if** pwd.text != updatepassword

{

**let** alert = UIAlertController(title: "Password is Invalid", message: "Please Try Again", preferredStyle: .alert)

**let** restart = UIAlertAction(title: "Press Here To Go Back To Login Page", style: .default, handler: {action **in** **self**.RestartPage()})

alert.addAction(restart)

present(alert , animated: **true**,completion: **nil**)

}

}

}

}

Welcome Page View Controller File:

**import** UIKit

**class** welcomePage: UIViewController

{

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

**var** dataString: String = ""

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

**super**.viewDidLoad()

print(dataString)

textlabel.text = dataString

}

**@IBAction** **func** calcbutton(\_ sender: UIButton)

{

**let** vc1 = storyboard?.instantiateViewController(withIdentifier: "cp1") **as**! CalculatorViewController

**self**.navigationController?.pushViewController(vc1, animated: **true**)

}

**@IBAction** **func** changepasswordbutton(\_ sender: UIButton)

{

**let** vc2 = storyboard?.instantiateViewController(withIdentifier: "cp3") **as**! changePassword1

**self**.navigationController?.pushViewController(vc2, animated: **true**)

}

}

Calculator View Controller File:

**import** UIKit

**class** CalculatorViewController: UIViewController

{

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

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

**var** workings:String = ""

**var** symbol : Character = "c"

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

{

**super**.viewDidLoad()

clearAll()

// Do any additional setup after loading the view.

}

**@IBAction** **func** equalsTap(\_ sender: **Any**)

{

**if** (validInput())

{

**if** symbol == ">"

{

print(leftexp)

**var** leftcheckedWorkingsForPercent = leftexp.replacingOccurrences(of: "%", with: "\*0.01")

**var** leftexpression = NSExpression(format:leftcheckedWorkingsForPercent)

**var** leftresult = leftexpression.expressionValue(with: **nil**, context: **nil**) **as**! Double

//var leftresultString = formatResult(result:result)

print(rightexp)

**let** start = rightexp.index(rightexp.startIndex, offsetBy: 1)

**let** end = rightexp.index(rightexp.startIndex, offsetBy: rightexp.count-1)

**let** range = start...end

**let** newRightExp = String(rightexp[range])

**var** rightcheckedWorkingsForPercent = newRightExp.replacingOccurrences(of: "%", with: "\*0.01")

**var** rightexpression = NSExpression(format:rightcheckedWorkingsForPercent)

**var** rightresult = rightexpression.expressionValue(with: **nil**, context: **nil**) **as**! Double

// var rightresultString = formatResult(result:result)

**var** result = great(lexp: leftresult, rexp: rightresult)

calculatorResults.text = result

}

**else**

{

**let** checkedWorkingsForPercent = workings.replacingOccurrences(of: "%", with: "\*0.01")

**let** expression = NSExpression(format:checkedWorkingsForPercent)

**let** result = expression.expressionValue(with: **nil**, context: **nil**) **as**! Double

**let** resultString = formatResult(result:result)

calculatorResults.text = resultString

}

}

**else**

{

**let** alert = UIAlertController(title: "Invalid Input", message: "its not working based on your input", preferredStyle: .alert)

alert.addAction(UIAlertAction(title: "okay", style: .default))

**self**.present(alert, animated: **true**, completion: **nil**)

}

}

**func** validInput() -> Bool

{

**var** count = 0

**var** funcInIndexes = [Int]()

**for** char **in** workings

{

**if**(specialCharacters(char: char))

{

funcInIndexes.append(count)

}

count+=1

}

**var** previous:Int = -1

**for** index **in** funcInIndexes

{

**if**(index == 0){

**return** **false**

}

**if**(index == workings.count - 1)

{

**return** **false**

}

**if**(previous != -1){

**if** (index - previous == 1)

{

**return** **false**

}

}

previous = index

}

**return** **true**

}

**func** specialCharacters(char :Character) -> Bool

{

//symbol = char

**if** (char == "\*")

{

**return** **true**

}

**if** (char == "/")

{

**return** **true**

}

**if** (char == "%")

{

**return** **true**

}

**if** (char == "-")

{

**return** **true**

}

**if** (char == "+")

{

**return** **true**

}

**return** **false**

}

**func** formatResult (result:Double) -> String

{

**if** (result.truncatingRemainder(dividingBy: 1) == 0)

{

**return** String(format: "%.0f", result)

}

**else**

{

**return** String(format: "%.2f", result)

}

}

**func** clearAll()

{

workings = ""

calculatorWorkings.text = ""

calculatorResults.text = ""

}

**@IBAction** **func** allClearTap(\_ sender: **Any**)

{

clearAll()

}

**@IBAction** **func** backTap(\_ sender: **Any**)

{

**if** (!workings.isEmpty){

workings.removeLast()

calculatorWorkings.text = workings

}

}

**var** leftexp = ""

**var** rightexp = ""

**func** addToWorkings(value:String)

{

**if** value == ">"

{

leftexp = workings

workings = ""

calculatorWorkings.text = workings

}

//rightexp = rightexp + value

workings = workings + value

print(workings)

calculatorWorkings.text = workings

rightexp = workings

}

**@IBAction** **func** percentTap(\_ sender: **Any**) {

addToWorkings(value: "%")

}

**@IBAction** **func** divideTap(\_ sender: **Any**) {

addToWorkings(value: "/")

}

**@IBAction** **func** timesTap(\_ sender: **Any**) {

addToWorkings(value: "\*")

}

**@IBAction** **func** minusTap(\_ sender: **Any**) {

addToWorkings(value: "-")

}

**@IBAction** **func** plusTap(\_ sender: **Any**) {

addToWorkings(value: "+")

}

**@IBAction** **func** decimalTap(\_ sender: **Any**) {

addToWorkings(value: ".")

}

**@IBAction** **func** zeroTap(\_ sender: **Any**) {

addToWorkings(value: "0")

}

**func** great(lexp : Double , rexp: Double)->String

{

**if** lexp > rexp

{

**return** "true"

}

**if** lexp == rexp

{

**return** "equal"

}

**else**

{

**return** "false"

}

}

**@IBAction** **func** greaterTap(\_ sender: **Any**)

{

addToWorkings(value: ">")

symbol = ">"

}

**@IBAction** **func** oneTap(\_ sender: **Any**) {

addToWorkings(value: "1")

}

**@IBAction** **func** twoTap(\_ sender: **Any**) {

addToWorkings(value: "2")

}

**@IBAction** **func** threeTap(\_ sender: **Any**) {

addToWorkings(value: "3")

}

**@IBAction** **func** fourTap(\_ sender: **Any**) {

addToWorkings(value: "4")

}

**@IBAction** **func** fiveTap(\_ sender: **Any**) {

addToWorkings(value: "5")

}

**@IBAction** **func** sixTap(\_ sender: **Any**) {

addToWorkings(value: "6")

}

**@IBAction** **func** sevenTap(\_ sender: **Any**) {

addToWorkings(value: "7")

}

**@IBAction** **func** eightTap(\_ sender: **Any**) {

addToWorkings(value: "8")

}

**@IBAction** **func** nineTap(\_ sender: **Any**) {

addToWorkings(value: "9")

}

}

Forgot Password View Controller File:

**import** UIKit

**class** changePasswordPage: UIViewController

{

**func** reset(){

userNameError.isHidden = **true**

}

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

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

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

**@IBAction** **func** updatePassword(\_ sender: **Any**)

{

print(newpass.text)

**if** oldusername.text == userName {

updatepassword = newpass.text!

**let** alert = UIAlertController(title: "Password Updation is successfull", message: "You can now login using your new password", preferredStyle: .alert)

**let** restartAction = UIAlertAction(title: "Login Again", style: .default, handler: {action **in** **self**.loginAgain()})

alert.addAction(restartAction)

present(alert , animated:**true** , completion: **nil**)

}

**else** **if** oldusername.text != userName{

userNameError.text = "Incorrect UserName or No UserName"

userNameError.isHidden = **false**

}

}

**func** loginAgain()

{

navigationController?.popToRootViewController(animated: **true**)

}

**var** dataString: String = ""

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

**super**.viewDidLoad()

print(dataString)

dataString = newpass.text!

reset()

}

}

Change Password View Controller File:

**import** UIKit

**class** changePassword1: UIViewController {

**func** reset(){

oldPasswordError.isHidden = **true**

}

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

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

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

**@IBAction** **func** changePassword(\_ sender: **Any**) {

print(newPassword.text)

**if** oldPass.text == updatepassword{

updatepassword = newPassword.text!

**let** alert = UIAlertController(title: "Password Updation is successfull", message: "You can now login using your new password", preferredStyle: .alert)

**let** restartAction = UIAlertAction(title: "Login Again", style: .default, handler: {action **in** **self**.loginAgain()})

alert.addAction(restartAction)

present(alert , animated:**true** , completion: **nil**)

}

**else** **if** oldPass.text != updatepassword{

oldPasswordError.text = "Incorrect Password or No Password"

oldPasswordError.isHidden = **false**

}

}

**func** loginAgain()

{

navigationController?.popToRootViewController(animated: **true**)

}

**var** dataString: String = ""

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

**super**.viewDidLoad()

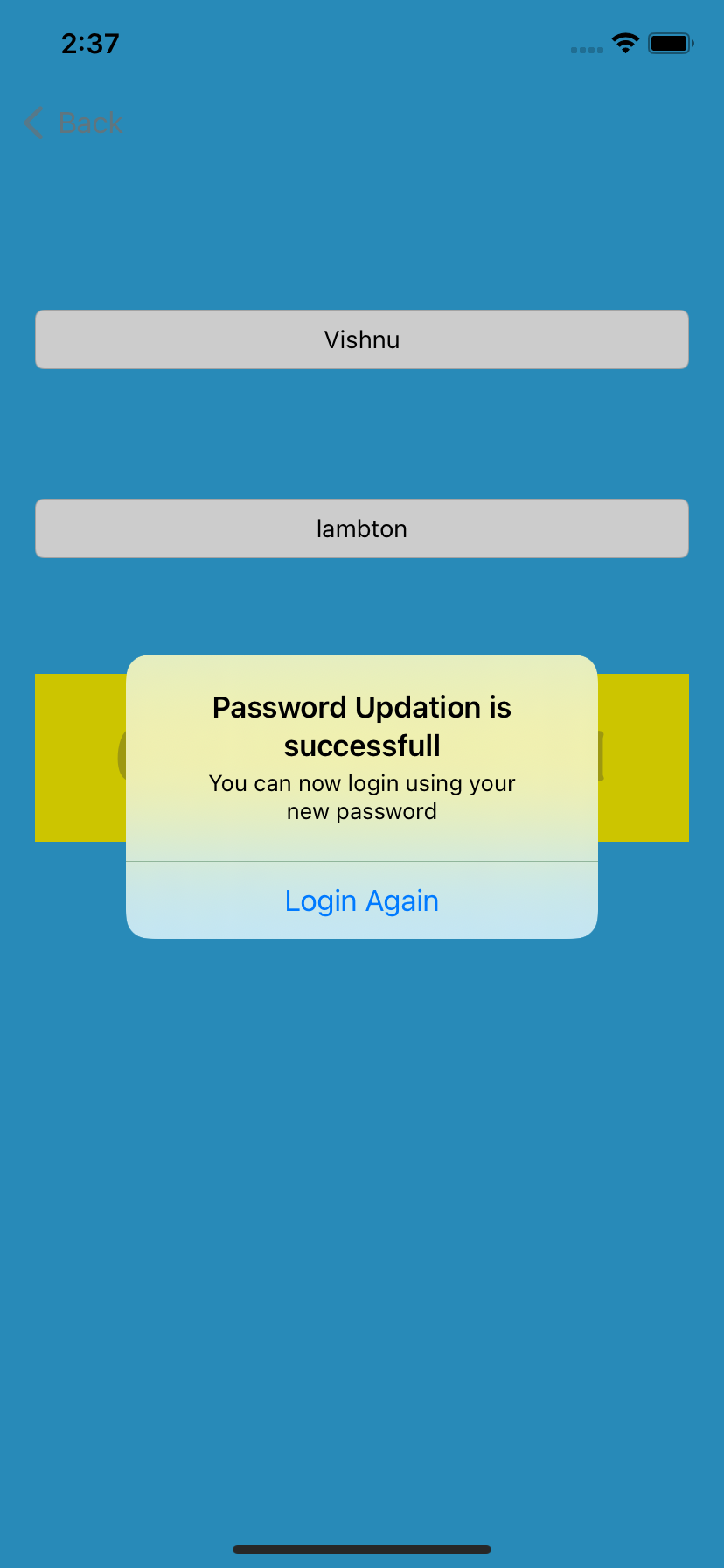
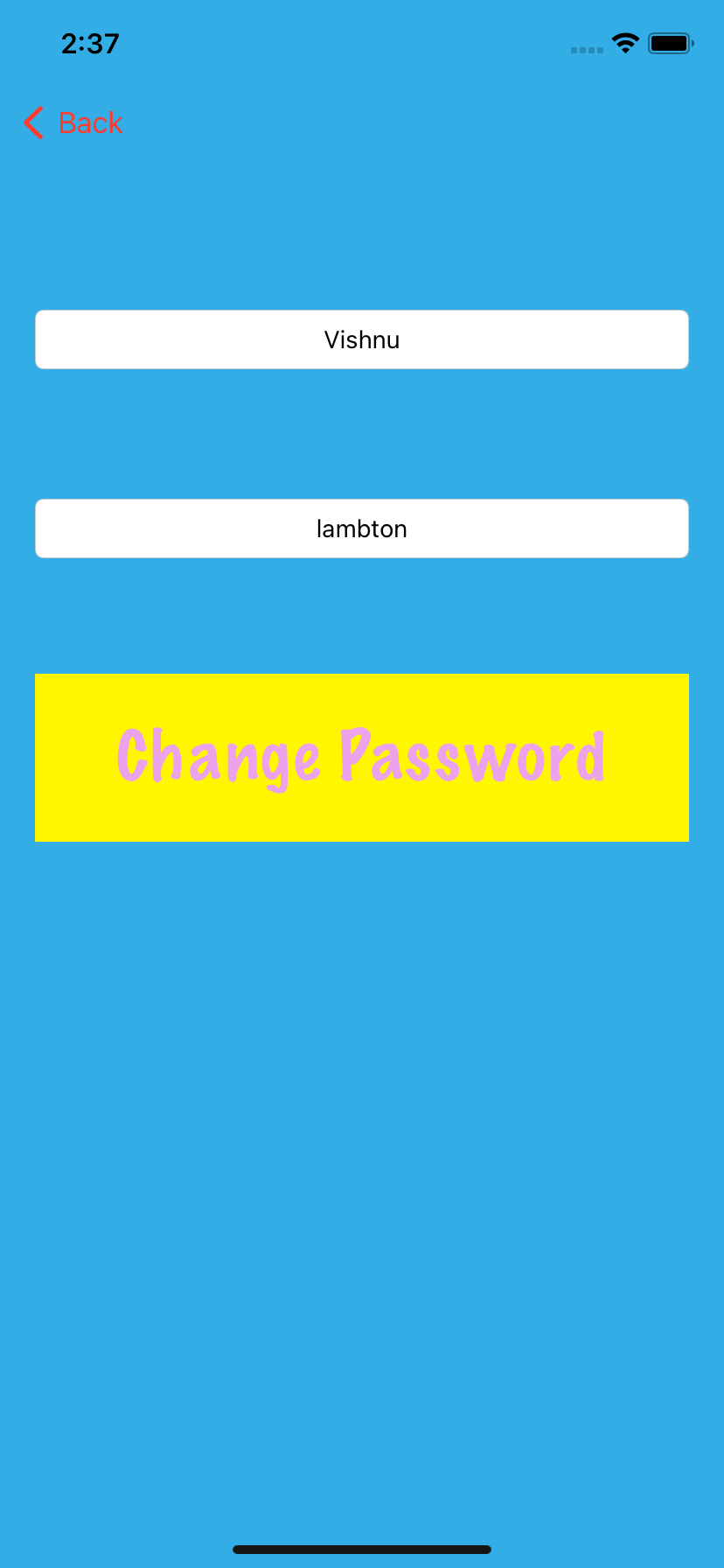
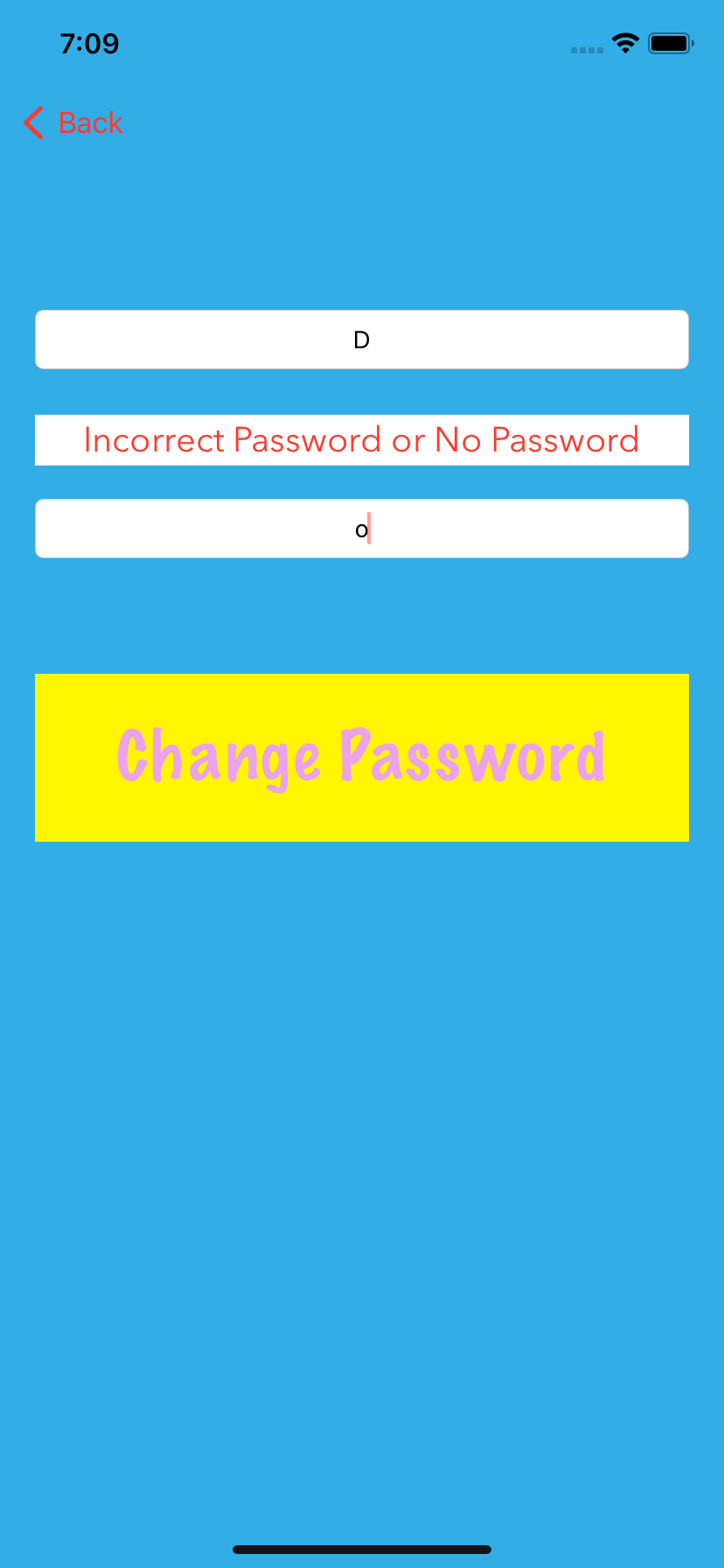
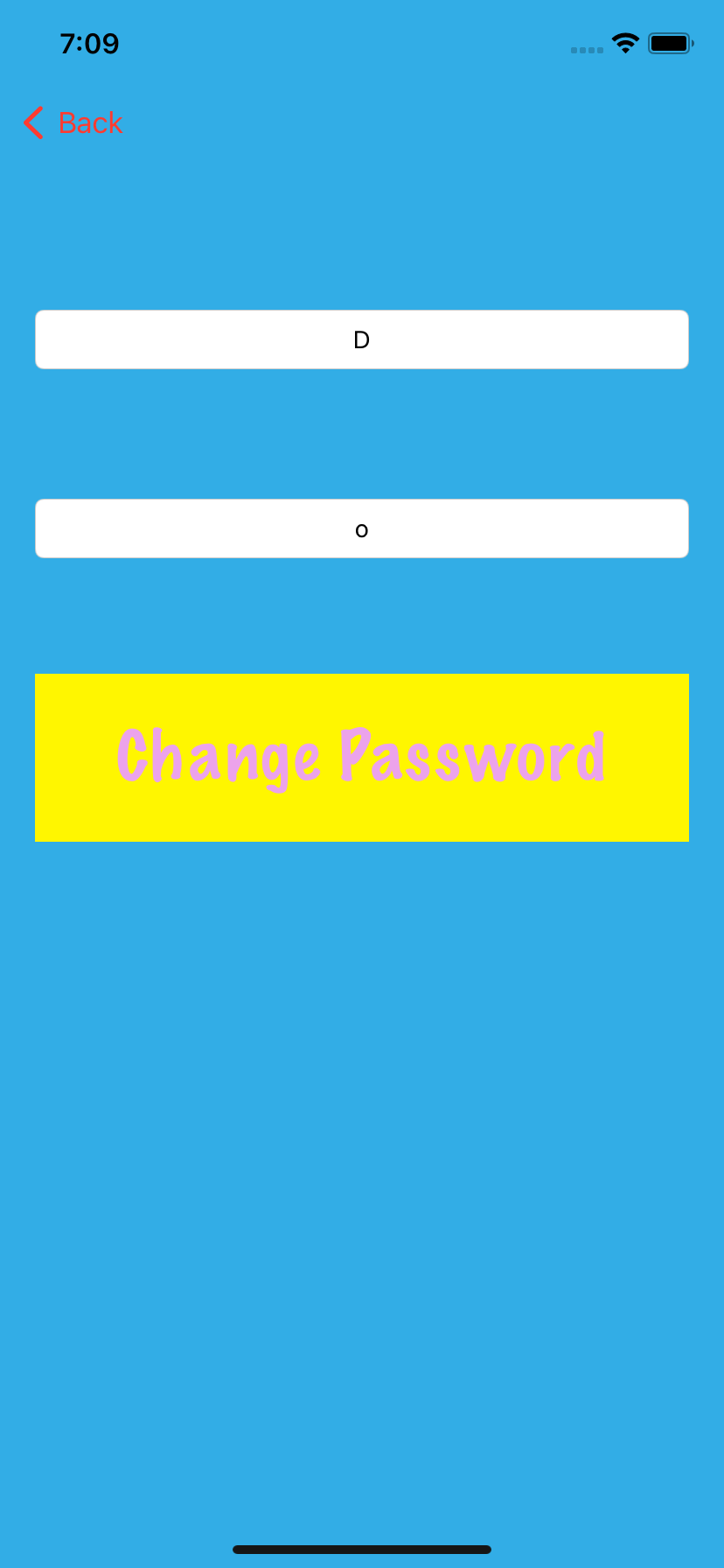
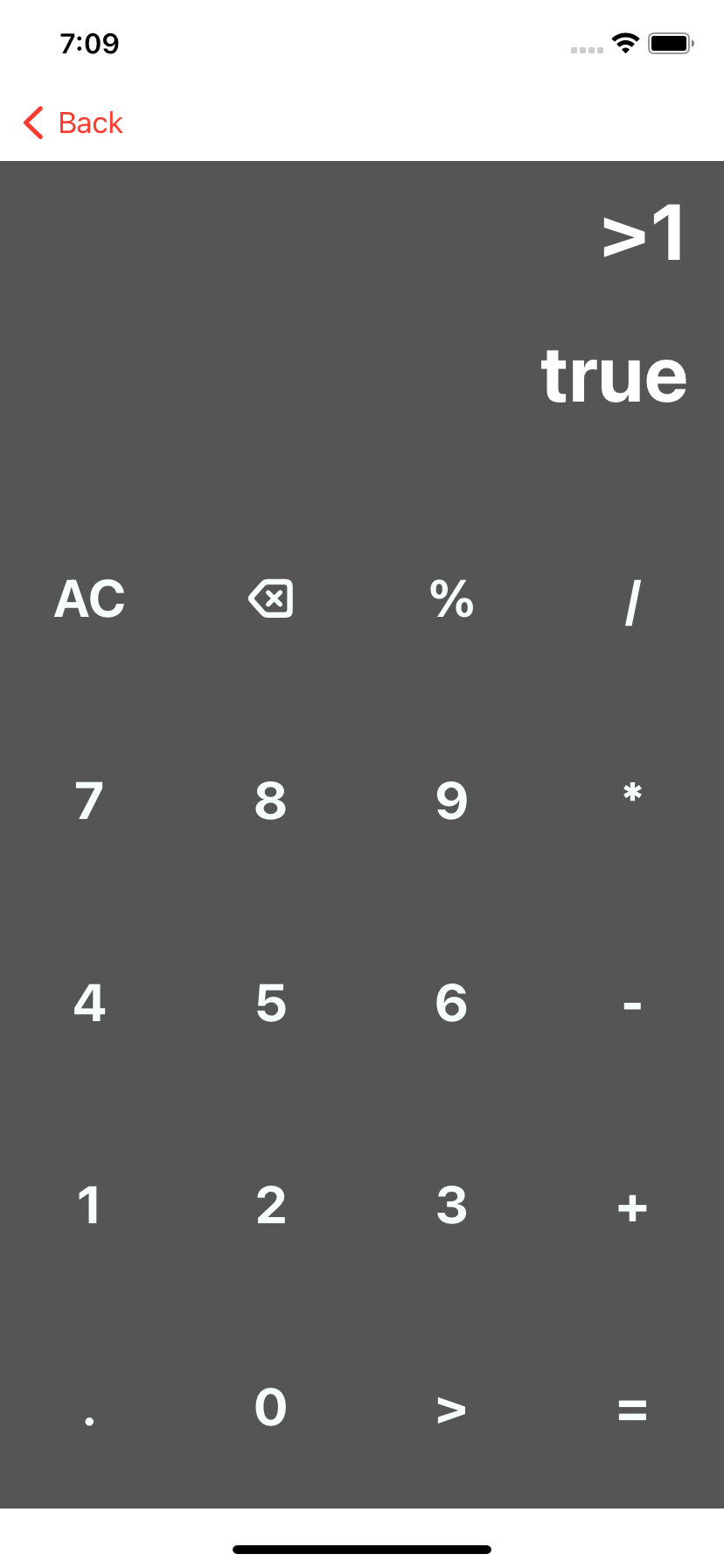
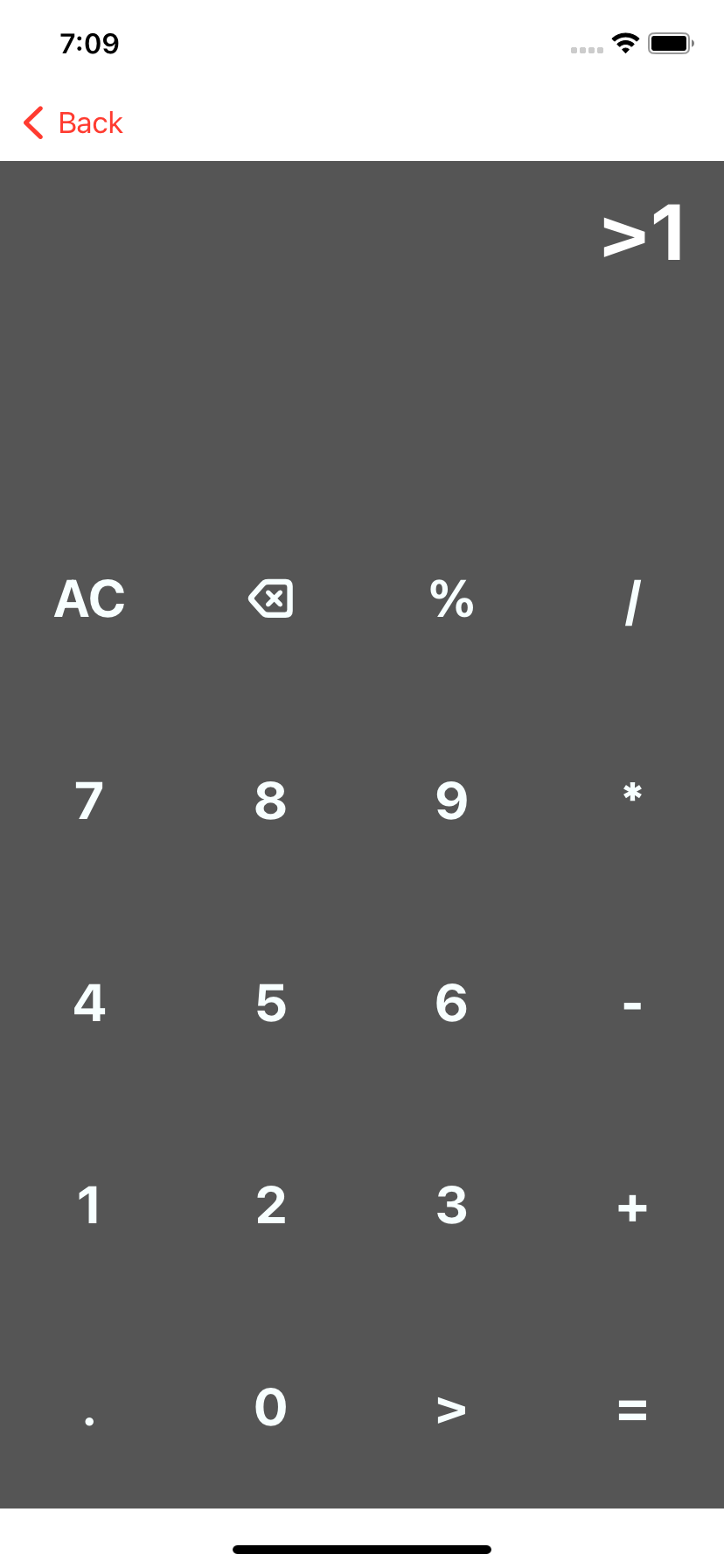
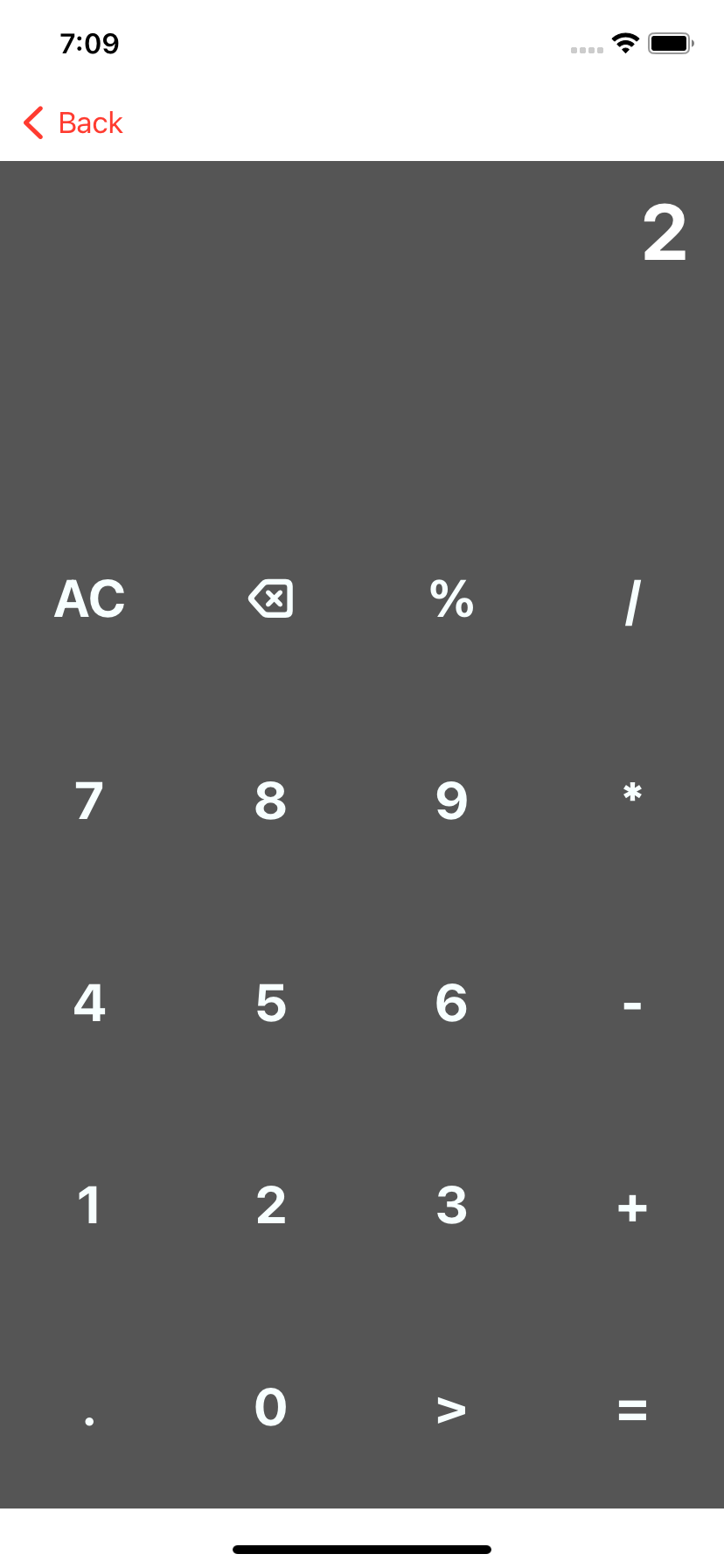
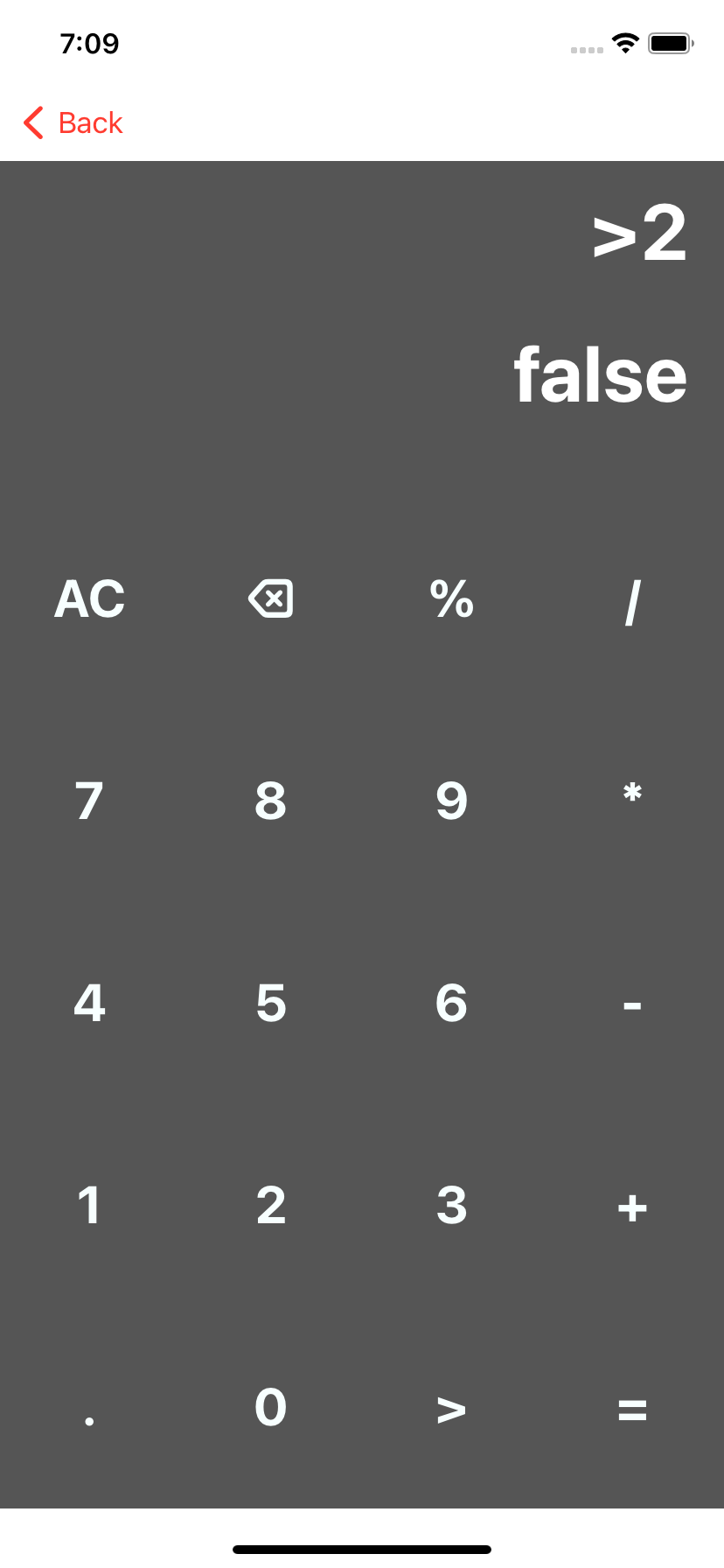
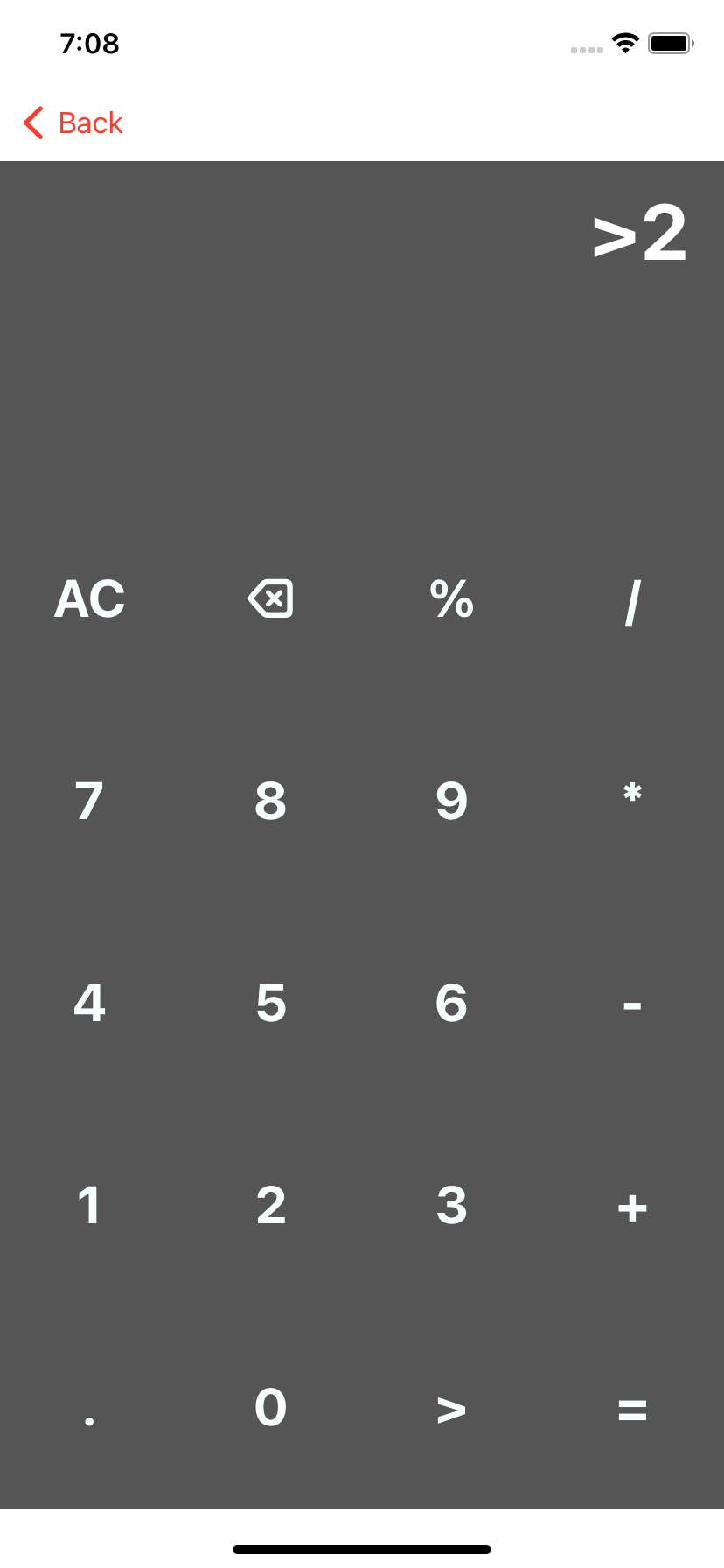
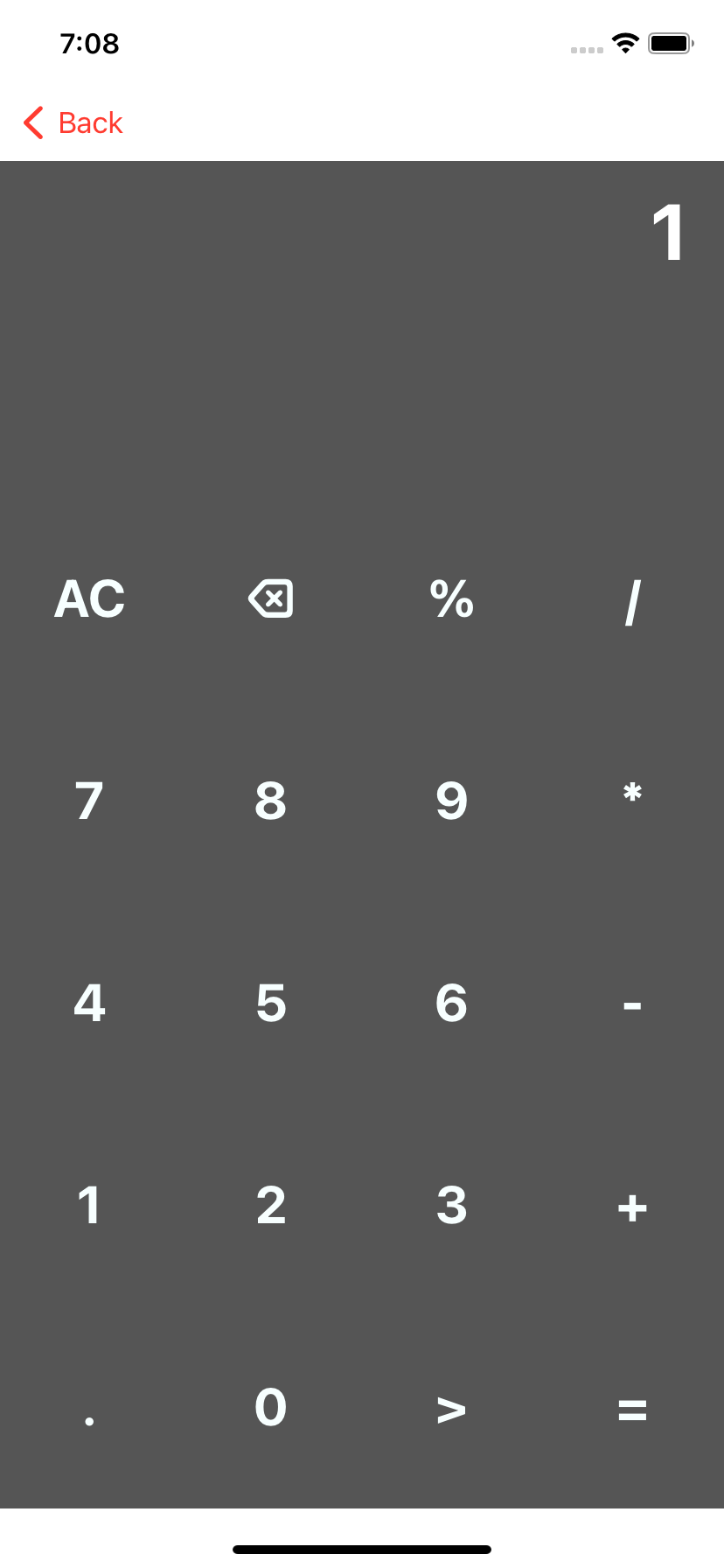
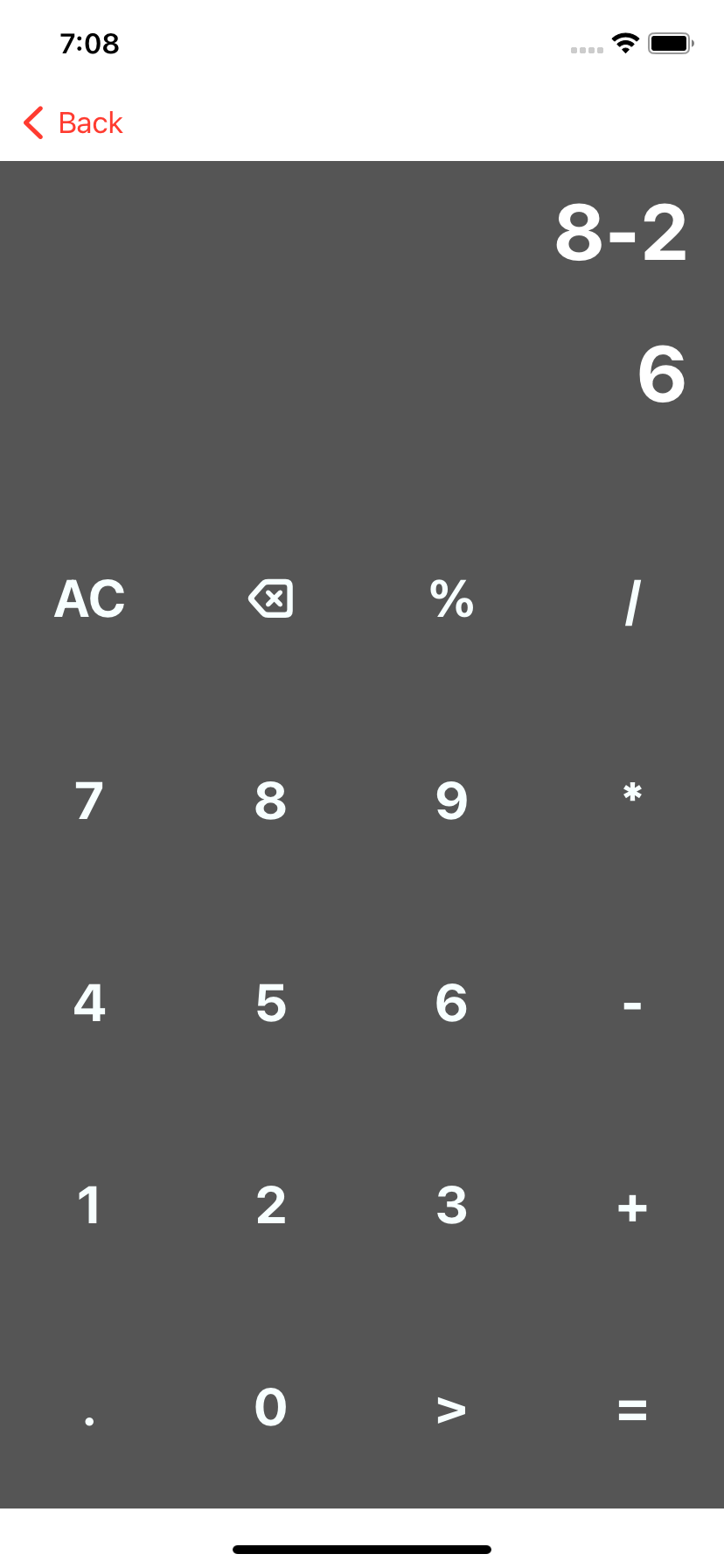
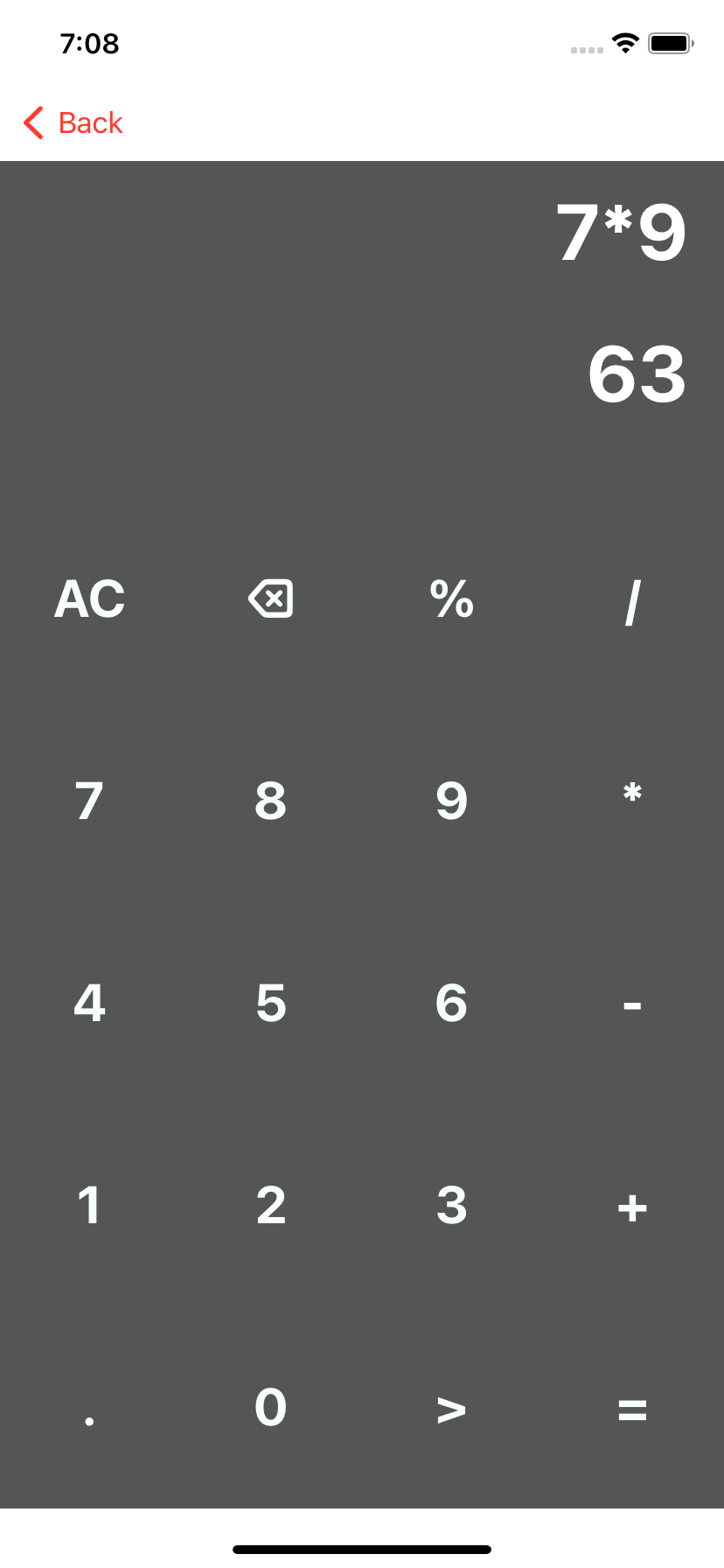
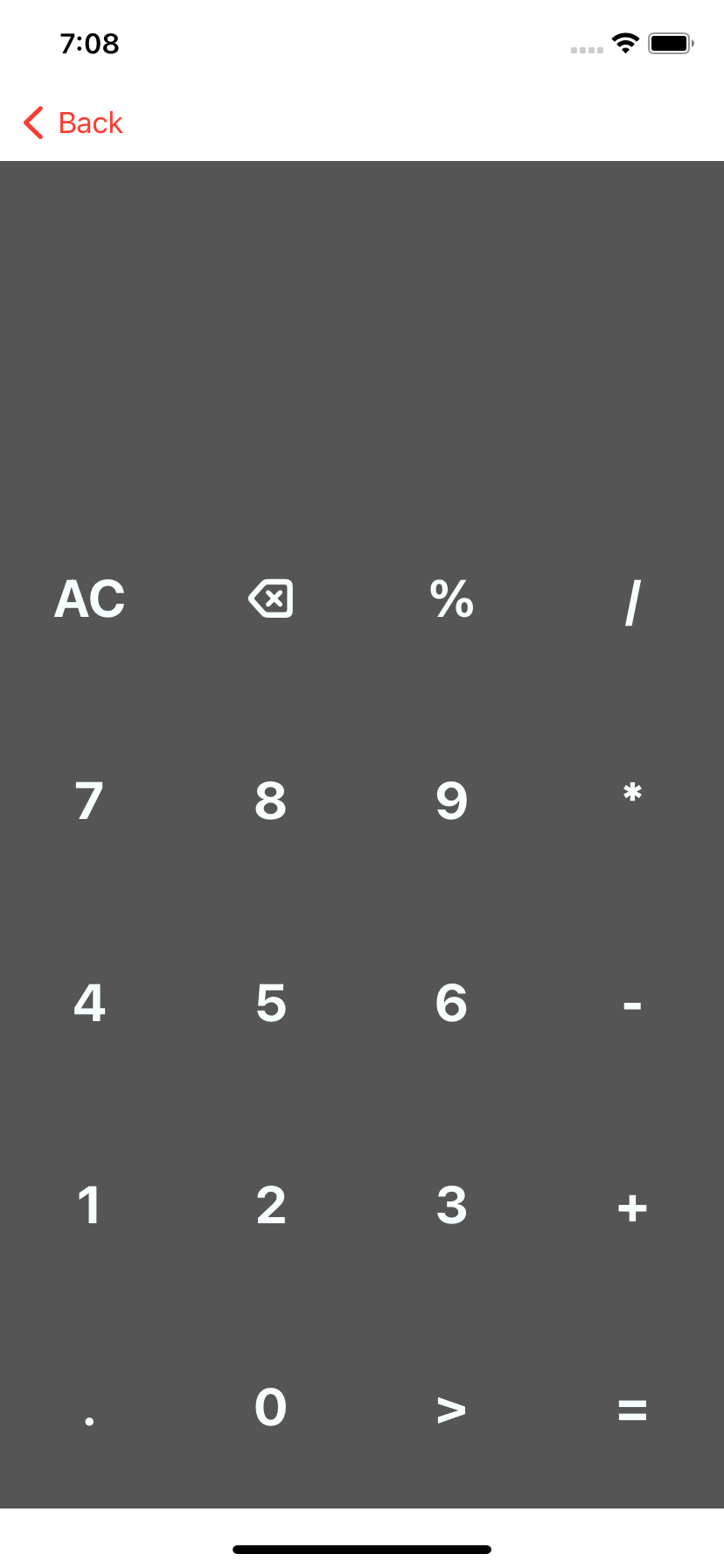
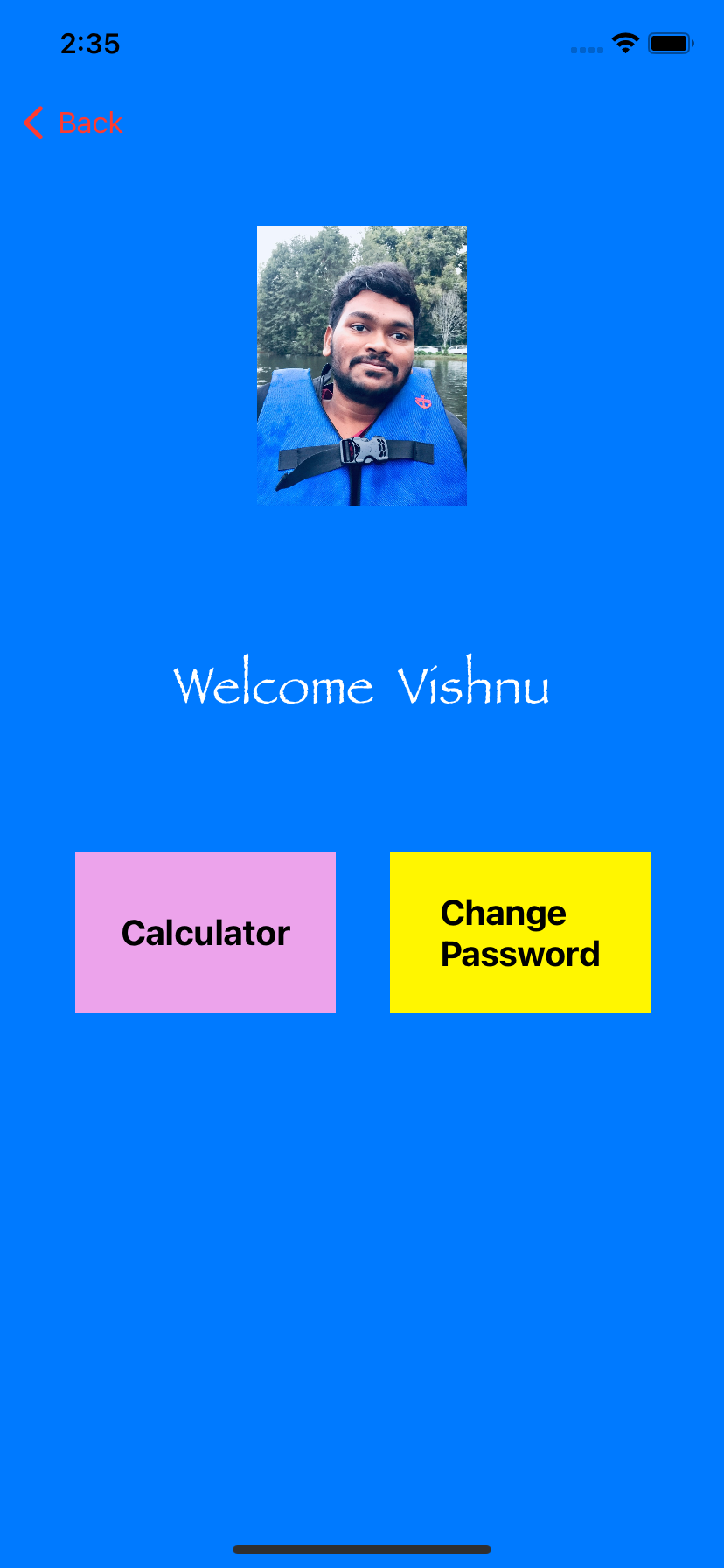
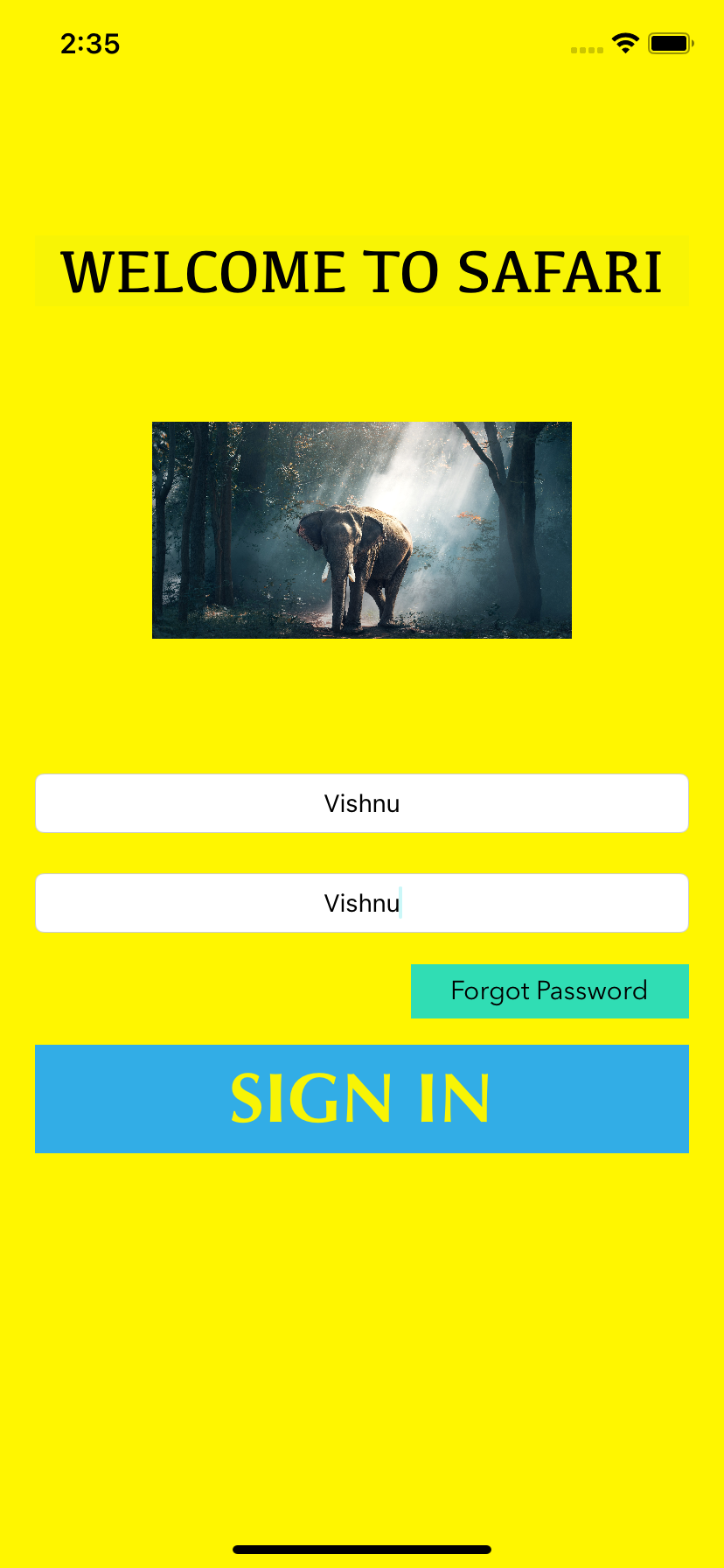
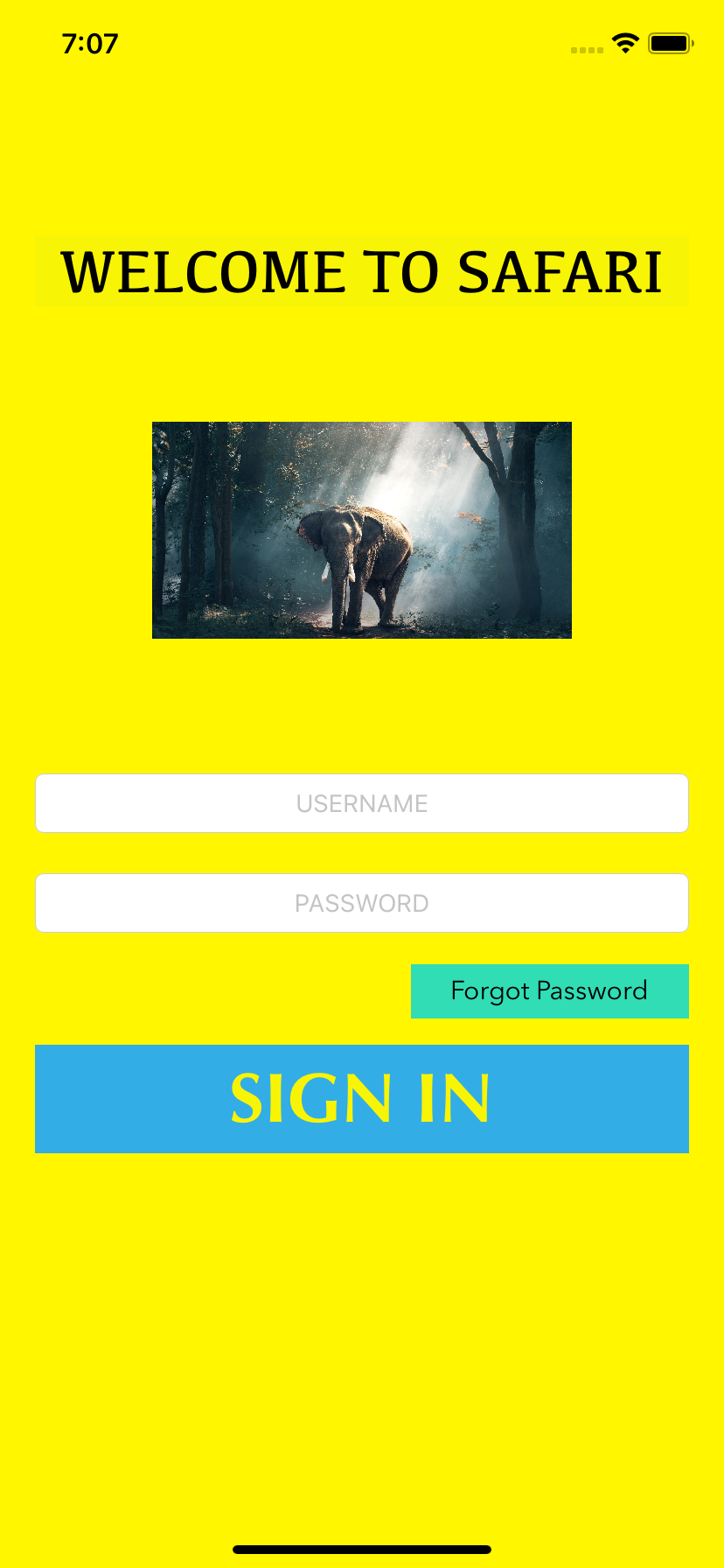
print(dataString)

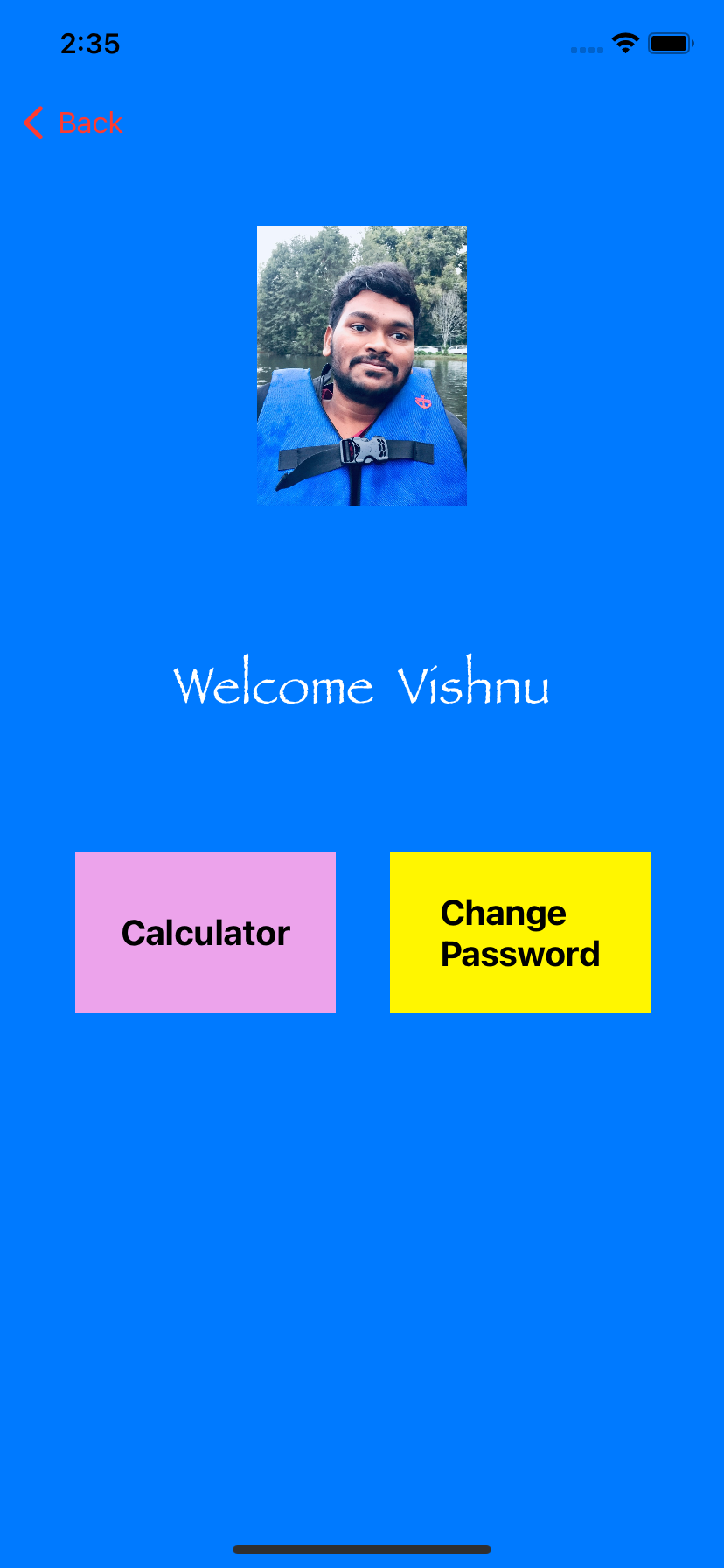
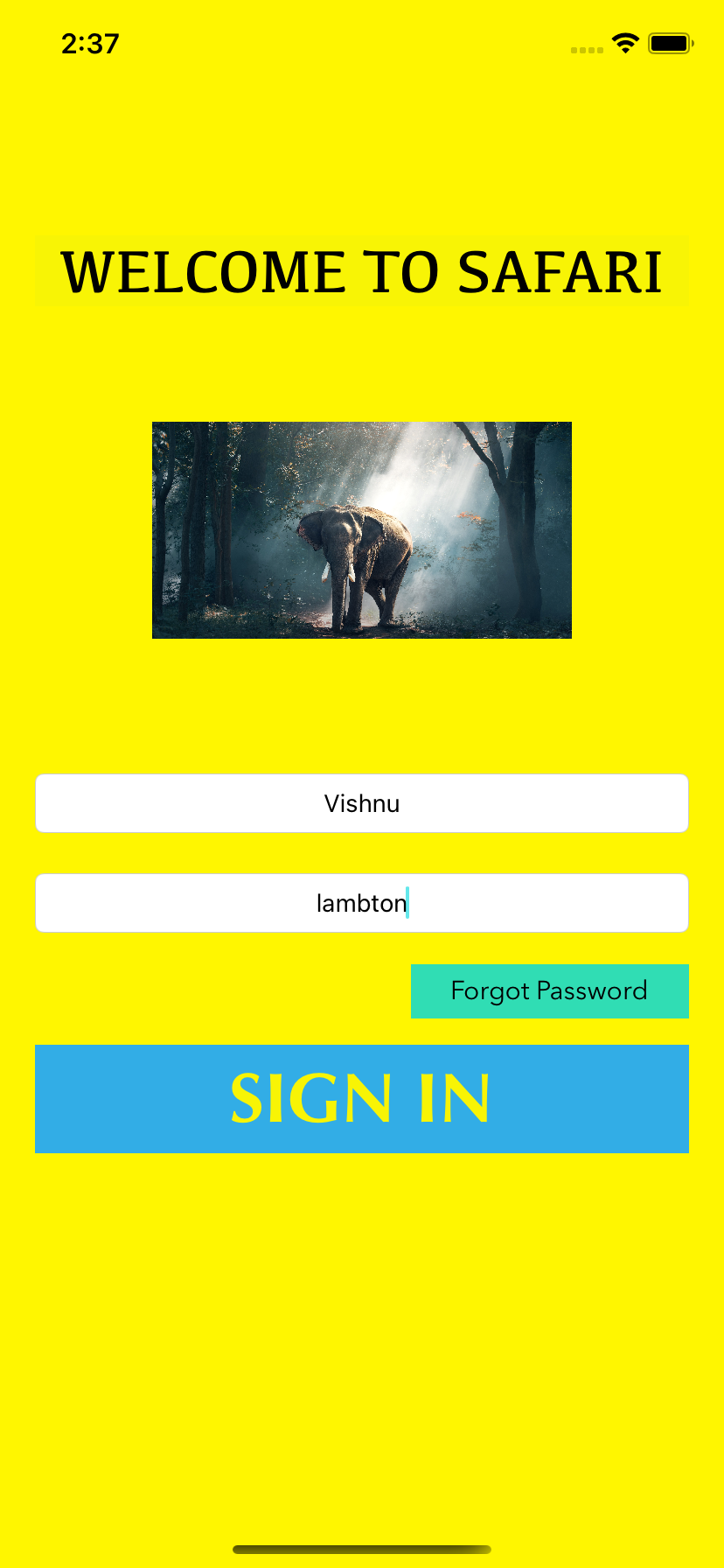
dataString = newPassword.text!

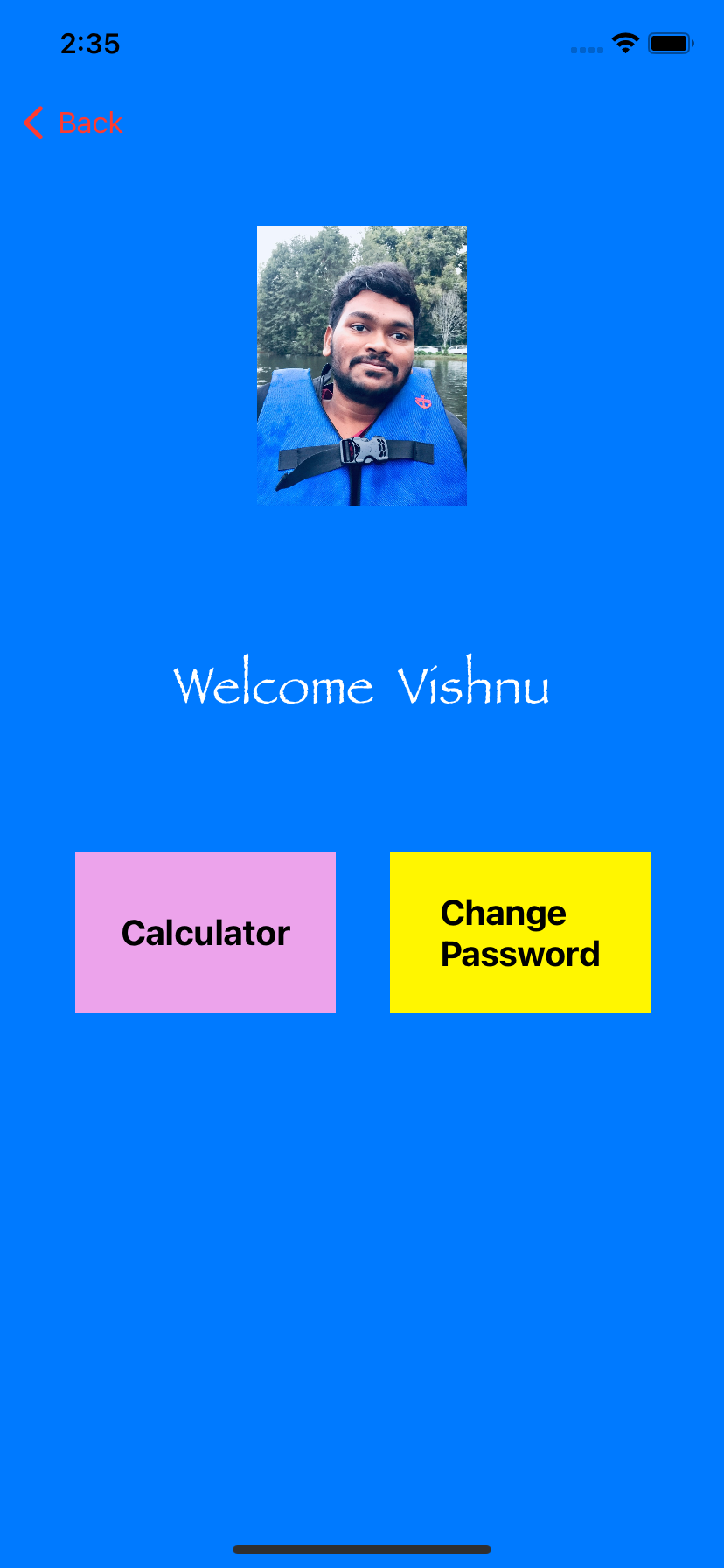
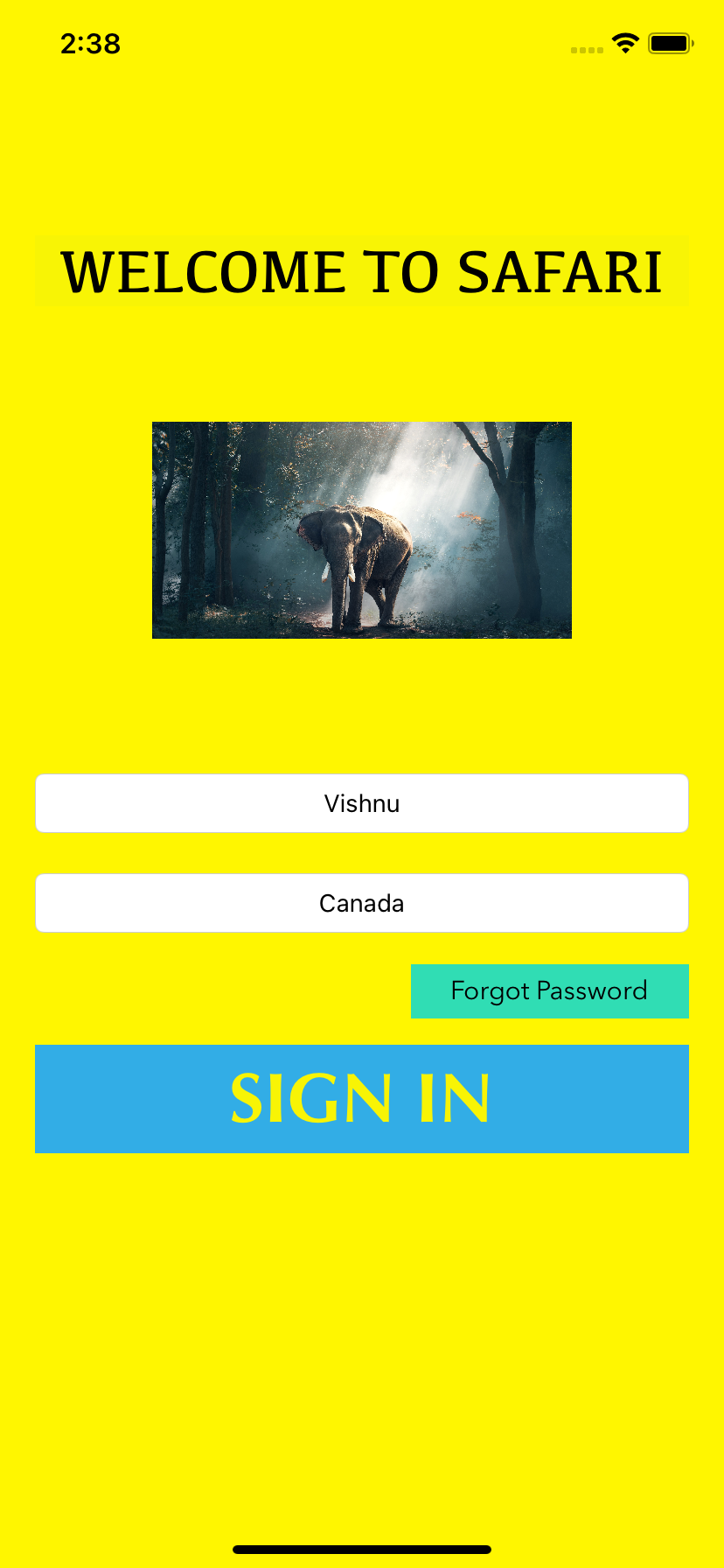
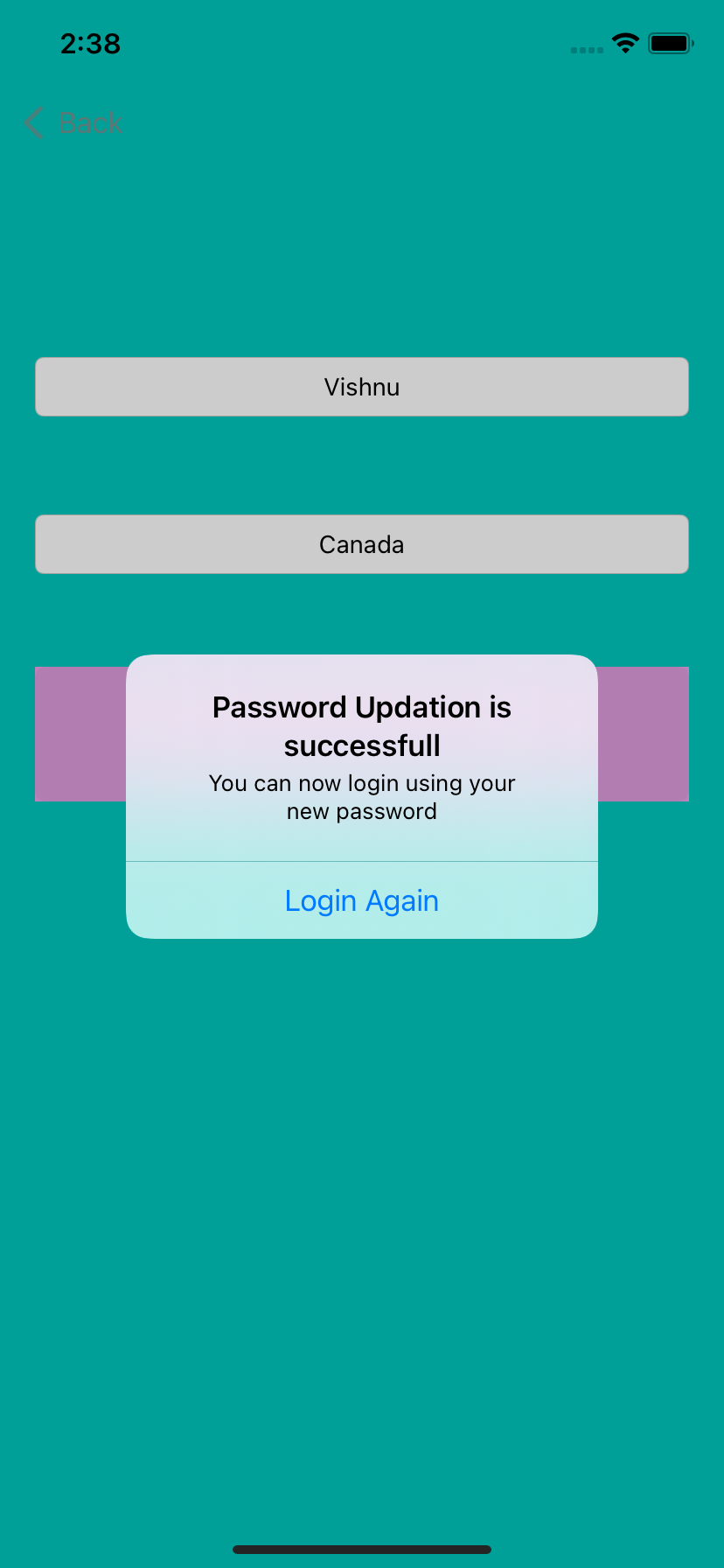
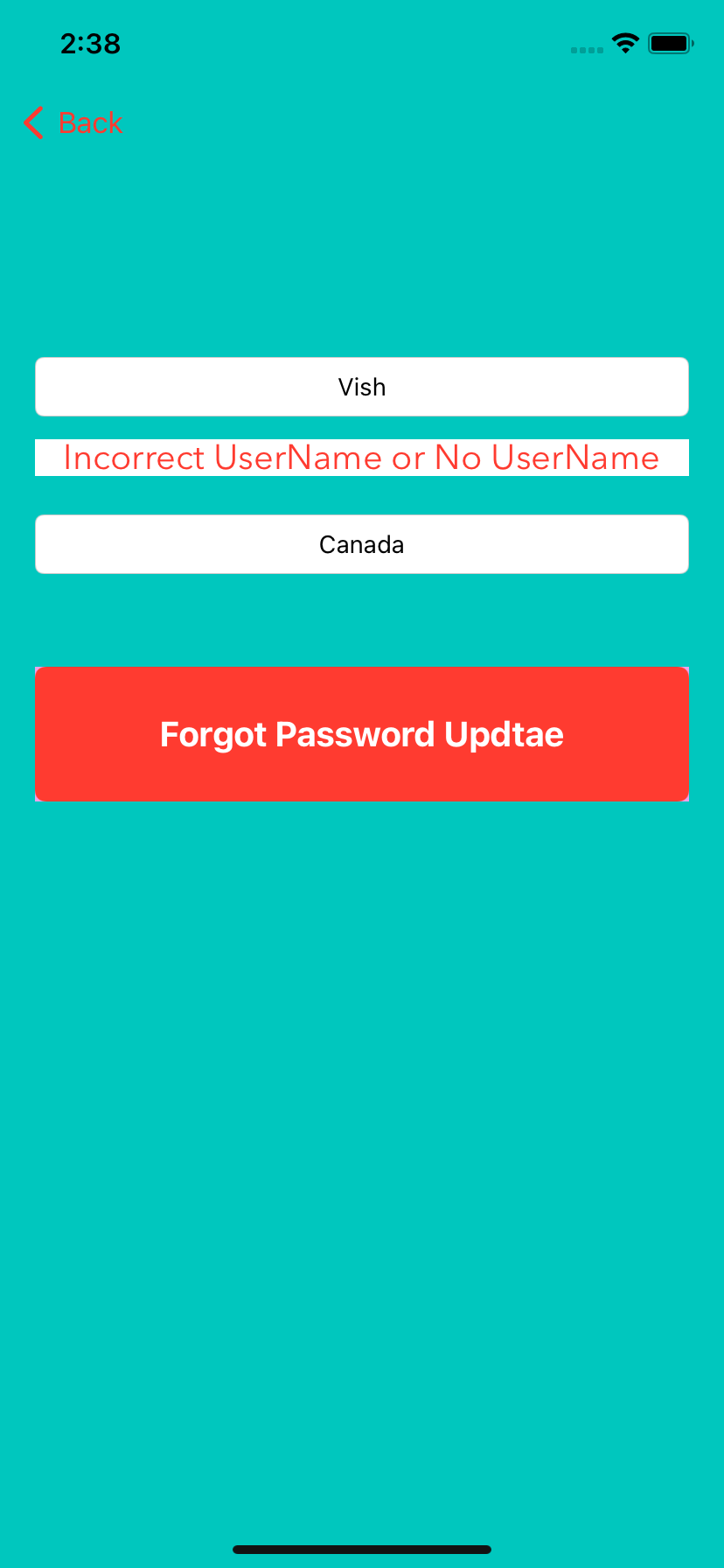
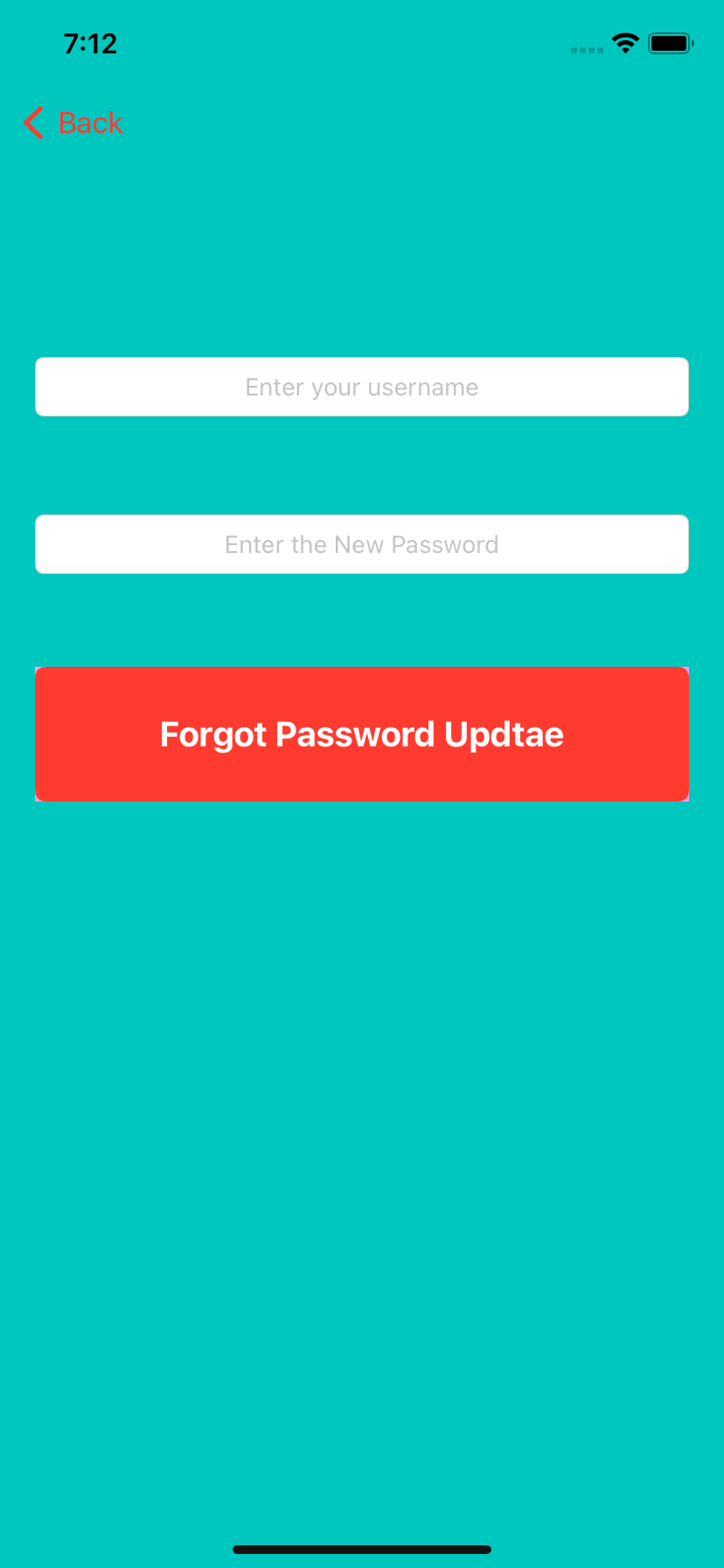
reset()

}

}

****

****

****