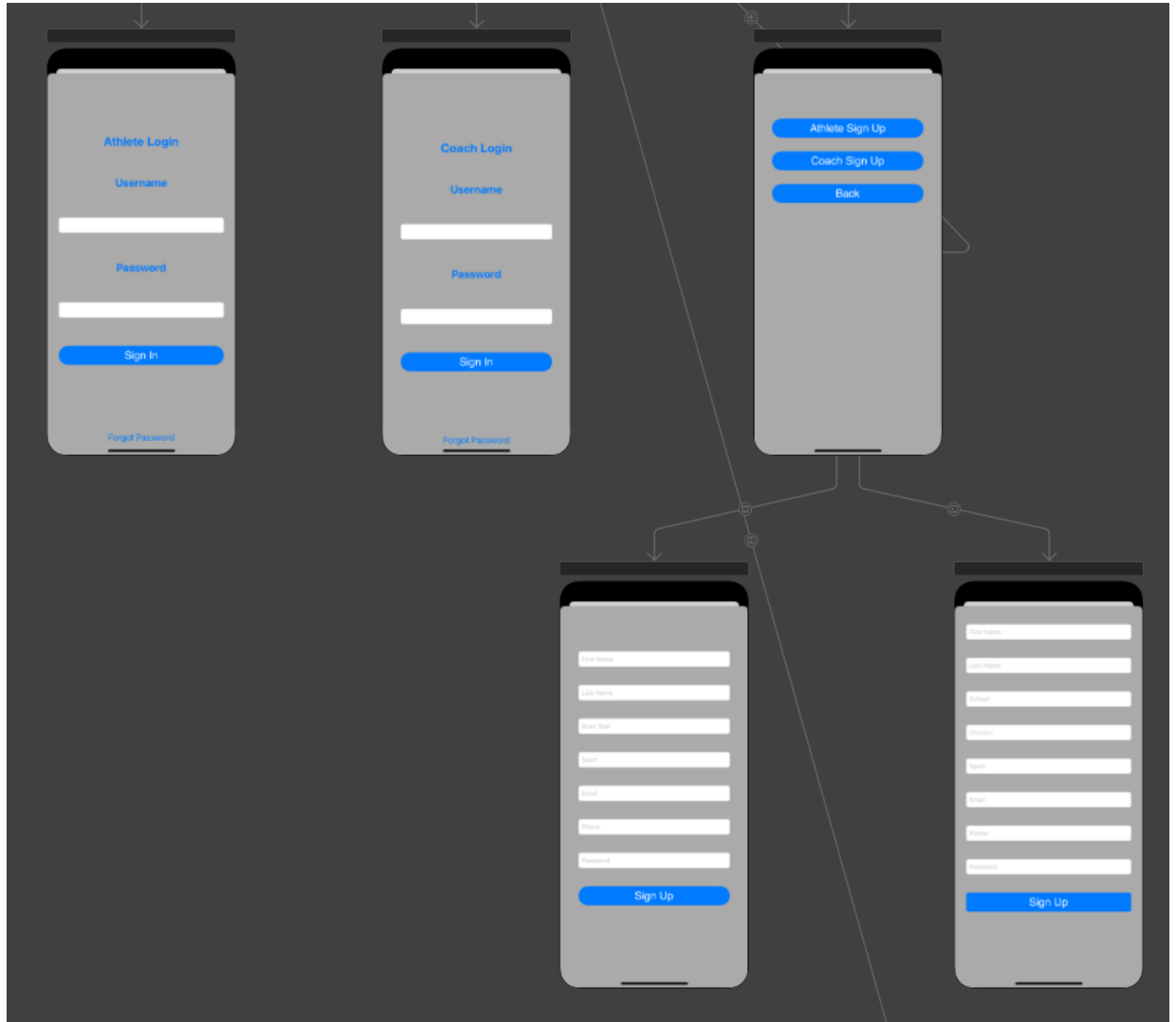
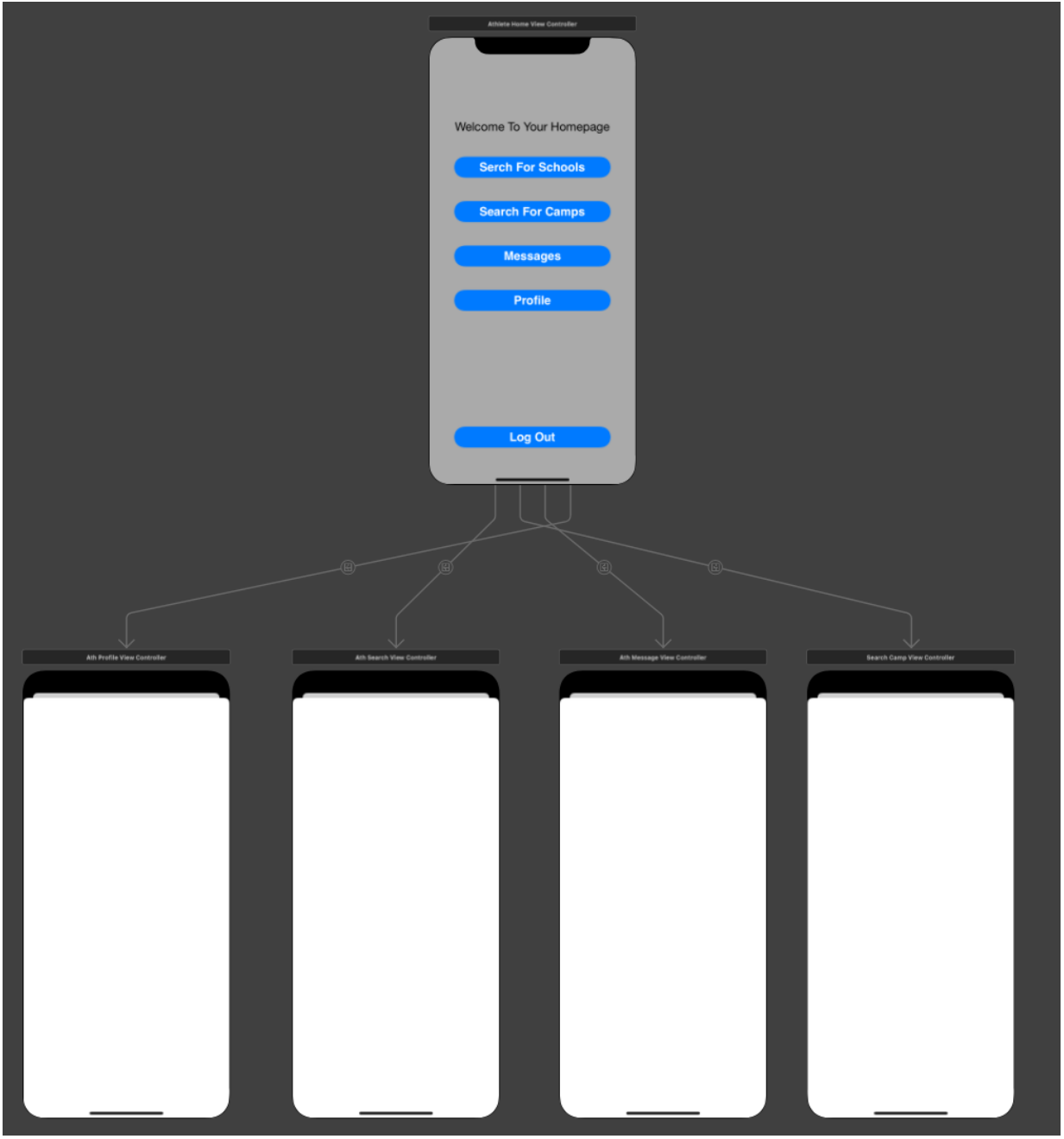


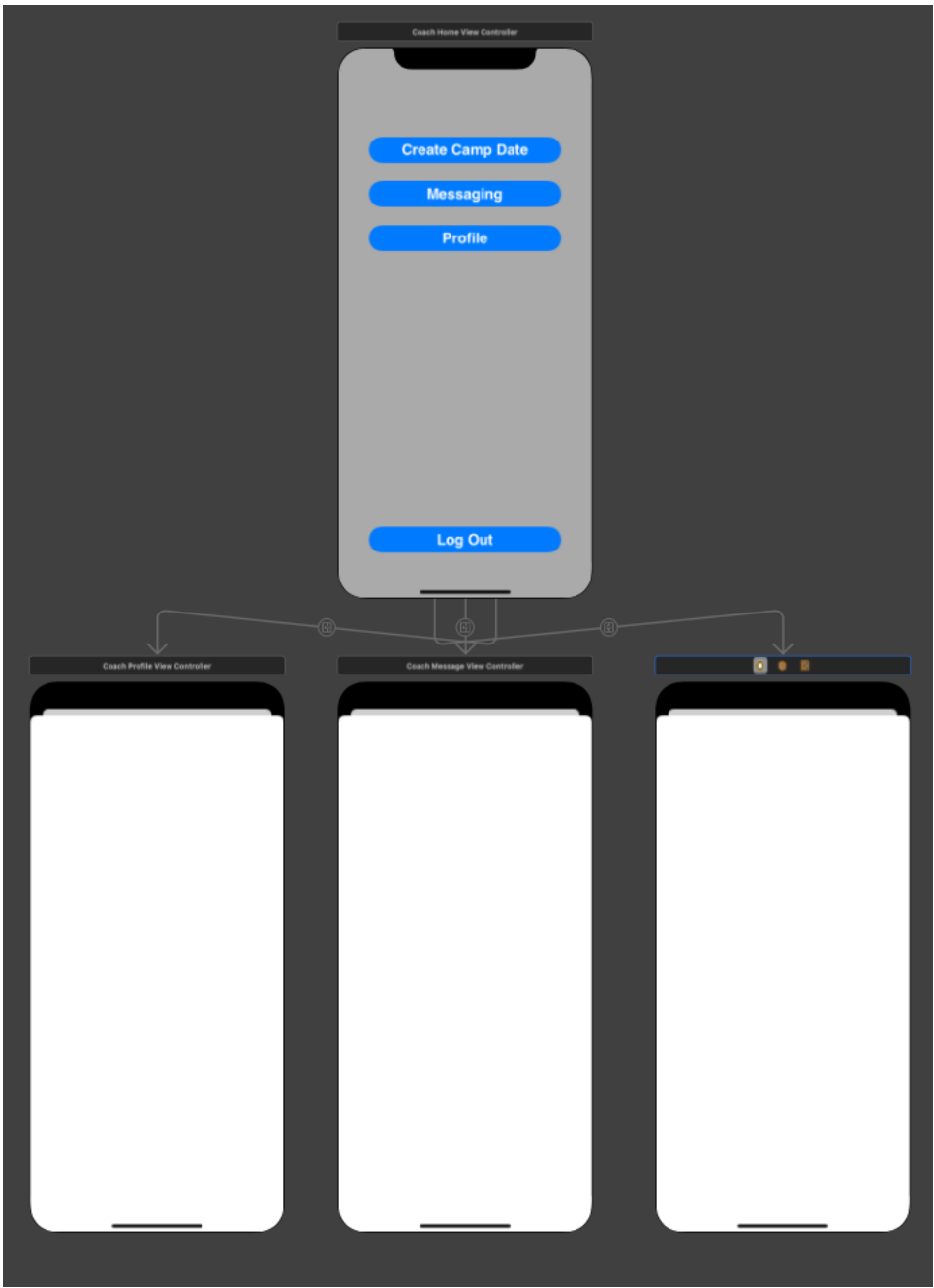
Term 3 Assignment

College Baseball Recruiting App for IOS

- Completed the sign-in and log-in view controllers with the firebase implantation. This allows users to create accounts and have their data stored within a database. The process also shows any errors that may occur with the sign-up and login process
- Started the process of creating the actual app with the functions that are intended for the project. I want to implement a school search that searches based on state, division, and the sport of the user's desire. I also want to implement a messaging service that is provided through firebase to allow an athlete and coach to have a conversation about anything they made need to talk about. The next implementation is for a coach to create events that athletes can view as needed and filter by date and location such as the state.







```

// returns the number of columns to display.

func numberOfComponents(in pickerView: UIPickerView) -> Int {
    return 1
}

// returns the # of rows in each component..

func pickerView(_ pickerView: UIPickerView, numberOfRowsInComponent component: Int) -> Int {
    return divisions.count
}

func pickerView(_ pickerView: UIPickerView, titleForRow row: Int, forComponent component: Int) ->
String? {
    return divisions[row]
}

func pickerView(_ pickerView: UIPickerView, didSelectRow row: Int, inComponent component: Int) {

    division.text = divisions[row]
    division.resignFirstResponder()
}

@IBAction func signUpTap(_ sender: Any) {

    //validate fields
    let error = validateFields()

    if error != nil
    {
        // There was an error
        showError(error!)
    }
    else
    {
        // Create Cleaned versions of data
        let firstname = fName.text!.trimmingCharacters(in: .whitespacesAndNewlines)
        let lastname = LName.text!.trimmingCharacters(in: .whitespacesAndNewlines)
        let school = school.text!.trimmingCharacters(in: .whitespacesAndNewlines)
        let division = division.text!.trimmingCharacters(in: .whitespacesAndNewlines)
        let sport = sport.text!.trimmingCharacters(in: .whitespacesAndNewlines)
        let email = email.text!.trimmingCharacters(in: .whitespacesAndNewlines)
        let phone = phone.text!.trimmingCharacters(in: .whitespacesAndNewlines)
        let password = password.text!.trimmingCharacters(in: .whitespacesAndNewlines)

        Auth.auth().createUser(withEmail: email, password: password) { (result, err) in
            // Check for errors
            if err != nil {
                // There was an error
                self.showError("Error Creating User")
            }
            else
            {
                Auth.auth().currentUser?.sendEmailVerification { (error) in
                    if let error = error
                    {print("Error when sending Email verification is
                    \ \(error)")}
                }
            }
        }
    }
}

```

```

// User was created successfully, stores data
let dbCoach = Firestore.firestore()

dbCoach.collection("users").document("Users").collection("Coach").addDocument(data: [
    "firstname":firstname,
    "lastname":lastname,
    "school":school,
    "division":division,
    "sport":sport,
    "email":email,
    "phone":phone,
    "password":password,
    "UID": result!.user.uid]) { (error) in

    if error != nil {
        self.showError("error saving user data")
    }

}

self.transitionToHome()
}
}
}
}

func showError(_ message:String) {
    errorLabel.text = message
    errorLabel.alpha = 1
}

func transitionToHome(){

let homeViewController = storyboard!.instantiateViewController(identifier:
    Constants.Storyboard.VerifyView) as? VerifyViewController

view.window?.rootViewController = homeViewController
view.window?.makeKeyAndVisible()
}

```