

Coding in Swift

Good tutorials

TableViews & how to populate them -> <https://www.youtube.com/watch?v=fFpMiSsynXM>
Dealing with dates -> <https://stackoverflow.com/questions/24089999/how-do-you-create-a-swift-date-object> and <https://stackoverflow.com/questions/42524651/convert-nsdate-to-string-in-ios-swift>

```
let formatter = DateFormatter()
formatter.dateFormat = "yyyy/MM/dd HH:mm"
let someDateTime = formatter.date(from: "2016/10/08 22:31")

let formatter = DateFormatter()
// initially set the format based on your datepicker date
formatter.dateFormat = "yyyy-MM-dd HH:mm:ss"

let myString = formatter.string(from: Date())
// convert your string to date
let yourDate = formatter.date(from: myString)
//then again set the date format whhich type of output you need
formatter.dateFormat = "dd-MMM-yyyy"
// again convert your date to string
let myStringafd = formatter.string(from: yourDate!)

print(myStringafd)
```

flef

Imagepicker

UIImagePickerControllerDelegate, UINavigationControllerDelegate -> add to the class declaration

let imagePicker = UIImagePickerController() -> before viewDidLoad

imagePicker.delegate = self -> in viewDidLoad

```
@IBAction func loadImage(_ sender: UIButton) {
    imagePicker.sourceType = .photoLibrary

    present(imagePicker, animated: true, completion: nil)
}

func imagePickerController(_ picker: UIImagePickerController, didFinishPickingMediaWithInfo info: [String : Any]) {
    if let pickedImage = info[UIImagePickerControllerOriginalImage] as?
        UIImage {
        profilePhotoImageView.contentMode = .scaleAspectFit
    }
}
```

```

        profilePhotoImageView.image = pickedImage
    }

    dismiss(animated: true, completion: nil)
}

func imagePickerControllerDidCancel(_ picker: UIImagePickerController) {
    dismiss(animated: true, completion: nil)
}

```

Upload picture into FireStorage

```

// Saves the profile pictures in Firebase storage, in the folder users_profile_pictures, with the user UID as the name
let storageRef = FIRStorage.storage().reference().child("users_profile_pictures/" + (user?.uid)! + ".png")
if let uploadData = UIImagePNGRepresentation(self.profilePhotoImageView.image!) {
    storageRef.put(uploadData, metadata: nil) { (metadata, error) in
        if error != nil {
            print("error")

            // Show an alert message
            let alertController = UIAlertController(title: "Alert", message: "Could not upload the profile photo.",
preferredStyle: .alert)

            let OK_button = UIAlertAction(title: "OK", style: .default) { (action: UIAlertAction) in
                print("You've pressed OK");
            }
            alertController.addAction(OK_button)
            self.present(alertController, animated: true, completion: nil)
        } else {

        }
    }
}
}

```

Pass object through navigation controller

```

guard let navVC = segue.destination as? UINavigationController else {
    fatalError("Unexpected destination view controller \(segue.destination)")
}

```

```
guard let addActivityViewController1 = navVC.viewControllers.first as? AddActivityViewController else {
    fatalError("Unexpected destination: \(navVC.viewControllers.first)")
}
```

```
addActivityViewController1.recent_activities = group?.recent_activities
```

//MARK: Navigation, prepareForSegue

```
super.prepare(for: segue, sender: sender)
```

```
switch(segue.identifier ?? "") {
```

```
case "view_editMembers":
```

```
    os_log("Viewing or editing the members of the activity.", log: OSLog.default, type: .debug)
```

```
case "showDetail":
```

```
    default:
```

```
        fatalError("Unexpected Segue Identifier; \(segue.identifier)")
```

Add member into Firebase database

```
// Add the owner to the members list of the group
```

```
self.refGroupMembers = FIRDatabase.database().reference().child("groups/(group_key)/members")
```

```
let member_key = refGroupMembers.childByAutoid().key
```

```
var member = ["id": userID]
```

```
self.refGroupMembers.child(member_key).setValue(member)
```

```
// Add the group to the groups list of the user
```

```
self.refUserGroups = FIRDatabase.database().reference().child("users/(userID!)/groups")
```

```
let group_key2 = refUserGroups.childByAutoid().key
```

```
var group2 = ["id": group_key]
```

```
self.refUserGroups.child(group_key2).setValue(group2)
```