

Gauthier Akim
Parcours Développeur IOS
Openclassrooms

Projet 5 : Instagrid

Présentation :

Dans le cadre de la formation Openclassrooms dans le parcours Développeur d'applications IOS. Je présente la feature créer en bonus pour le Projet 5 Instagrid.

Explication de ma feature :

Appareil Photo :

Je l'ai implémenter lorsque l'on va appuyer sur le petit + pour ajouter une photo à notre vue, la possibilité de choisir entre la librairie ou l'appareil photo.

Premièrement je créer une fonction « **displayImageSourceMenu** » qui va contenir un popup (**ici la constante alerte**), qui nous permettra de choisir.

Si l'on ajoute une image de notre librairie photo (**la constante library**) ou en utilisant l'appareil photo (**constante appareil**) à notre vue.

```

        button.isSelected = false
    }
}

@IBAction func buttonAddPhoto(_ sender: UIButton) {
    tag = sender.tag
    displayImageSourceMenu()
}

func displayImageSourceMenu() {
    let alerte = UIAlertController(title: "Take Photo?", message: "Choice Média", preferredStyle: .actionSheet)
    let appareil = UIAlertAction(title: "Camera Device", style: .default) { (act) in
        self.cameraMode()
    }
    let library = UIAlertAction(title: "Library Photo", style: .default) { (act) in
        self.libraryMode()
    }
    let cancel = UIAlertAction(title: "Cancel", style: .cancel, handler: nil)
    alerte.addAction(appareil)
    alerte.addAction(library)
    alerte.addAction(cancel)
    present(alerte, animated: true, completion: nil)
}
}

//MARK: - UIImagePickerController
extension ViewController: UIImagePickerControllerDelegate, UINavigationControllerDelegate {

    func imagePickerController(_ picker: UIImagePickerController, didFinishPickingMediaWithInfo info: [String : Any])
    guard let tag = tag else { return }
    guard let selectedImage = info[UIImagePickerControllerOriginalImage] as? UIImage else { return }
    guard let imageView = self.imageView else { return }
    imageView.image = selectedImage
}

```

Dans la **constante appareil** après avoir défini son alerte j'appelle dans la fermeture la fonction cameraMode()

La fonction **cameraMode** est créer et aura 2 conditions :

Premièrement si l'appareil photo est disponible vérifier que la source soit bien la camera alors :

- Alors l'image prise avec l'appareil photo est ajoutée à notre vue.
- Sinon on va afficher un message d'erreur sur l'écran et avertir à l'utilisateur que l'appareil photo n'est pas disponible.

```

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

///Access on library photo mode
func libraryMode() {
    self.imagePicker.sourceType = .photoLibrary
    self.present(imagePicker, animated: true, completion: nil)
}

///Access on Camera Device mode
func cameraMode() {
    if UIImagePickerController.isSourceTypeAvailable(.camera) {
        self.imagePicker.sourceType = .camera
        self.present(imagePicker, animated: true, completion: nil)
    } else {
        print("No camera available")
        let error = UIAlertController(title: "Error", message: "No camera Device", preferredStyle: .alert)
        let cancel = UIAlertAction(title: "Cancel", style: .cancel, handler: nil)
        error.addAction(cancel)
        self.present(error, animated: true, completion: nil)
    }
}

/// When a add image on view hide a button and add gesture to tap on it
func checkPhoto() {
    guard let tag = tag else {return}
    gridView.addPhotoButton[tag].isHidden = true
    gridView.photoImageViews[tag].isUserInteractionEnabled = true
}

```