

TP1

MYPIZZA

INTRODUCTION

- ▶ Reprendre sous iOS le TP api REST MyPizza vu avec Mr Meunier en programmation Android.

TRAVAIL A FAIRE

- ▶ Refaire les visuels présents dans le sujet de Mr Meunier
- ▶ <https://www.lamarmotte.info/wp-content/uploads/2023/01/Android-TP-2-API-Rest.pdf>
- ▶ Récupérer la photo de fond sur Teams. (pizza.jpg)

TRAVAIL A FAIRE

- ▶ Mettre en place les navigations entre les vues.
- ▶ Utilisez les navigationstack et le modifier .navigationDestination
- ▶ Pour quitter une vue et retourner à la vue précédente on peut utiliser:
 - La variable d'environnement : `@Environment(\.dismiss)` **private var dismiss**
 - Et la fonction `dismiss()` dans votre code

TRAVAIL A FAIRE

- ▶ API REST
 - Reprendre les fonctionnalités décrites dans le document de Mr Meunier.
 - Utiliser les fichiers de l'archive présente sur Teams : API.zip
 - ▶ Décompresser cette archive dans le répertoire de votre projet.

TRAVAIL A FAIRE

► API REST

- Utilisation de la méthode getRequest
- TheData correspond au type de donnée retourné par la requête GET
- Token est facultatif

```
private final let urlString = "https://yoururl.com/api/getsomething"

func getData() {
    if let url = URL(string: self.urlString) {
        ApiService.shared.getRequest(url: url, type: TheData.self, [token: Token,] completionHandler: { (response) in
            if response.success ?? false {
                #do something with TheData
            } { (error) in
                print("Could not fetch images! »)
            }
        }
    }
}
```

TRAVAIL A FAIRE

► API REST

- Utilisation de la méthode postRequest
- TheData correspond au type de donnée retourné par la requête POST
- Token est facultatif
- Params est un dictionnaire correspondant aux paramètres soumis.

```
private final let postURLString = "https://jsonplaceholder.typicode.com/posts"
```

```
func postData() {  
    let params: [String: Any] = ["title": "foo", "body": "bar", "userId": 1, "id": 101]  
    if let url = URL(string: self.postURLString) {  
        APIService.shared.postRequest(url: url, params: params, type: User.self, [token: Token,] completionHandler: { (response) in  
            #do something with response  
        }) { (error) in  
            print(error)  
        }  
    }  
}
```