

## BONUS

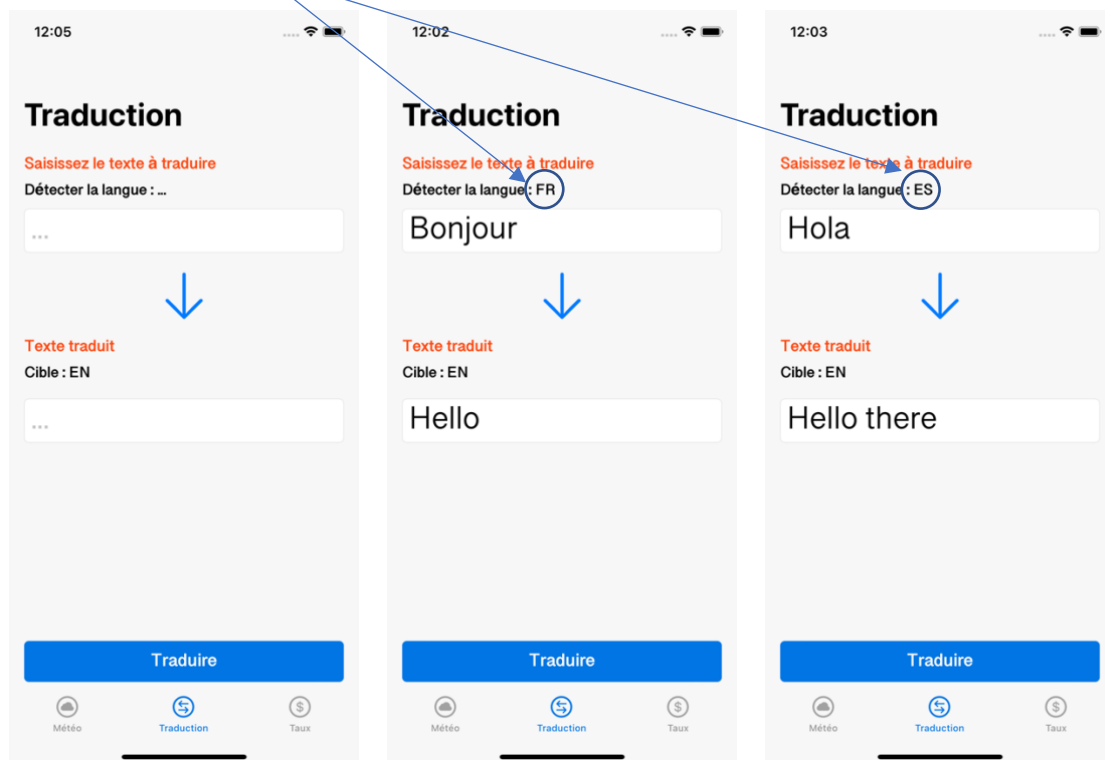
Afin de conclure le projet, un bonus est obligatoire. J'ai choisi d'implémenter l'auto-détection de la langue d'origine dans la page de traduction avec l'API Google Translation.

```
50 // for automatic detection using the network call
51 func getDetection(text: String) {
52     detectionService.getDetection(text: text) { [weak self] result in
53         guard let self = self else {
54             return
55         }
56         DispatchQueue.main.async {
57             switch result {
58                 case .success(let detection):
59                     guard let languageDetected = detection.data.detections.first?.first?.language else { return }
60                     self.autoDetectedLanguageLabel.text = "Détecter la langue : \(languageDetected.uppercased())"
61                 case .failure(let apiError):
62                     self.alertPopUp()
63                     print(apiError)
64             }
65         }
66     }
67 }
```

La fonction getDetection permet d'obtenir et charger les données d'auto-détection.

- weak, empêche que chaque objet est une référence vers l'autre, des fuites mémoires (retain cycle). Sans, la RAM de l'appareil arriverait à saturation et l'application crash.
- DispatchQueue.main.async, permet d'exécuter le bloc de code sans passer par la file d'attente, et donc limiter les blocages de l'application.
- switch result va gérer deux cas :
  - succès qui prend pour paramètre d'entrée une constante détection de type DetectionLanguage (modèle de données de la détection automatique) et où il sera passé toutes les données que l'on veut charger dans l'application. Dans un premier temps, je déballe mon optionnel à l'aide guard let qui sont les données de l'auto-détection. Enfin j'affiche la langue détecter dans un UILabel, en indiquant c'elle-ci à l'aide d'un String interpolation.
  - échec, est le cas où les données sont inexistantes ou erronées, alors une alertPopUp sera affiché afin que l'utilisateur en soit informé.

Résultat du String interpolation ainsi que l'auto-détection :



## MODEL DE DONNÉES

```
10 struct DetectionLanguage: Decodable {
11     let data: DataClassDetection
12 }
```

```
10 struct DataClassDetection: Decodable {
11     let detections: [[Detections]]
12 }
```

```
10 struct Detections: Decodable {
11     let isReliable: Bool
12     let confidence: Double
13     let language: String
14 }
```