

Développement mobile - TP6

October 9, 2018

Victor COLEAU
victor.coleau@etu.univ-tours.fr
Thomas COUCHOUD
thomas.couchoud@etu.univ-tours.fr



Contents

1	Questions	2
1.1	Question 2	2
1.2	Question 3	2
1.3	Question 4	2
1.4	Question 5	2
1.4.1	Requête	4

Chapter 1

Questions

1.1 Question 2

En exécutant la requête dans le thread UI, il ne se passe rien du tout même en ayant précisé que notre application a besoin des permissions internet.

1.2 Question 3

Avec le bypass cela fonctionne cependant cette pratique n'est pas conseillée.

1.3 Question 4

Le AsyncTask permet aussi de réaliser ce que l'on veut tout en ayant un code propre.

1.4 Question 5



Figure 1.1 – Ecran d'accueil

Au démarrage de l'application nous pouvons voir les éléments suivant:

- Un champ de texte afin de renseigner l'adresse IP à chercher (si vide l'adresse IP de la requête sera utilisée).
- Un bouton « Random » permettant d'afficher une adresse IP aléatoire.

- Un bouton « Start » permettant de lancer la requête.
- Ainsi qu'un champ de texte, pour l'instant vide, où les résultats s'afficheront.

Il est à noter que le champ de saisie est configuré en tant que nombre afin d'afficher un clavier adapté (chiffres et point) (inputType=phone). De plus un filtre est présent afin de valider ou non le format de l'adresse entrée. Si ce dernier est invalide la saisie n'est pas prise en compte. Cela est réalisé grâce à un InputFilter.

Lors du clic sur le bouton start, un tache asynchrone est démarrée afin d'effectuer la requête.

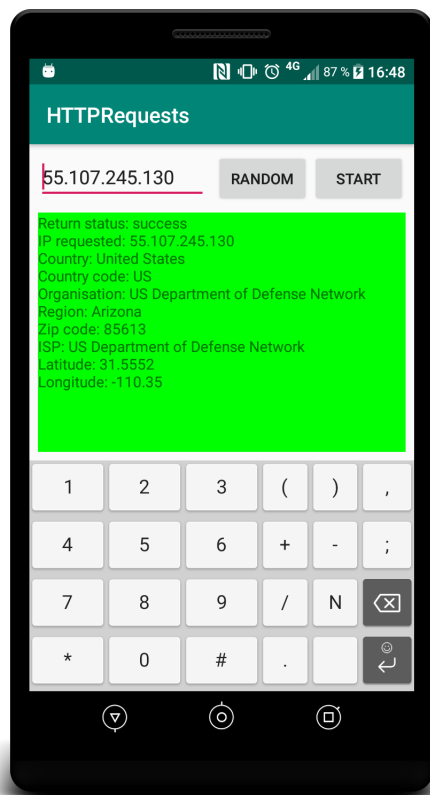


Figure 1.2 – Requête valide

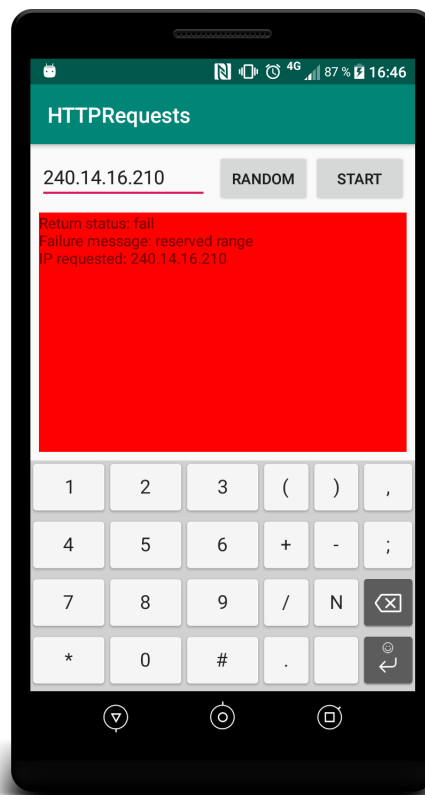


Figure 1.3 – Requête valide

Si les données reçues correspondent à une IP valide, on affiche ces dernières sur un fond vert.

Si les données reçues correspondent à une IP indisponible, on affiche un message d'erreur sur fond rouge.

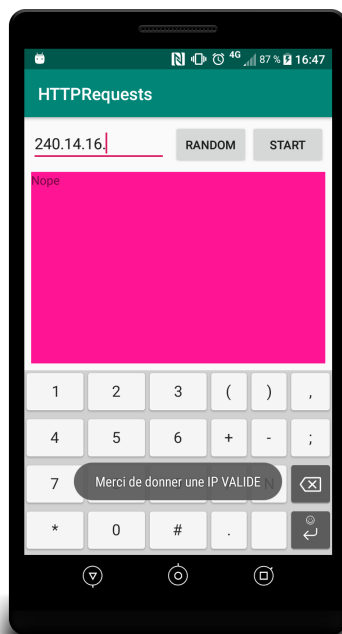


Figure 1.4 – Requête valide

Si l'adresse de requêtage est mal formatée (partie manquante), on affiche un message d'erreur sur fond deep pink ainsi qu'un Toast.

1.4.1 Requête

Afin d'effectuer les requêtes vers l'api, nous avons créé une classe héritant de `AsyncTask`. Pour pouvoir écrire nos résultats dans notre activity de base, le constructeur prend en paramètre une `TextView`.

Lors d'un appel à cette tache, la méthode `doInBackground` est appelée avec en paramètre l'URL à requêtes. Une fois la page récupérée, `doInBackground` retourne un `String` la représentant ce qui va permettre de la traiter dans `onPostExecute`. Dans cette dernière on parse le JSON obtenu et construisons un objet `GeoIp` qui est ensuite écrit dans le `TextView` passé dans le constructeur.