

Utilisation du clavier sous Android



Tutos-android



Benbourahla

par Feanorin

Date de publication : 29 avril 2011

Dernière mise à jour :

Ce tutoriel va aborder l'utilisation du **clavier sur Android**. C'est un sujet très important pour tout développeur d'applications car cela amène un véritable plus à votre application. Je vous conseille donc de ne pas négliger cette partie, qui peut paraître simple et inintéressante. Cet article regroupera donc des astuces pour améliorer l'expérience utilisateur de votre application.

I - Comment adapter le clavier à vos besoins.....	3
II - Où est le champ suivant ?.....	3
III - Mon clavier cache ma vue.....	3
IV - Conclusion.....	4
V - Remerciements.....	4
VI - Liens.....	4

I - Comment adapter le clavier à vos besoins

Dans votre application, l'utilisateur aura peut être besoin de saisir quelques données de types différents. Dans ce cas, il faut faciliter le travail à l'utilisateur en faisant apparaître un clavier en fonction du type de champ à remplir (adresse mail, numéro de téléphone etc.). Pour cela dans votre **"EditText"**, il faut simplement spécifier l'option **android:inputType**. Voici les différentes valeurs disponibles :

- **text** (par défaut) : clavier normal ;
- **textCapCharacters** : clavier tout en majuscule ;
- **textCapWords** : première lettre automatiquement en majuscule ;
- **textCapSentences** : phrase en majuscule ;
- **textAutoCorrect** : activer la correction automatique ;
- **textAutoComplete** : activer la correction automatique forcée ;
- **textMultiLine** : texte sur plusieurs lignes ;
- **textNoSuggestions** : pas de suggestion de correction ;
- **textUri** : saisie d'une URL Web ;
- **textEmailAddress** : adresse mail ;
- **textEmailSubject** : sujet de mail ;
- **textShortMessage** : active le raccourci smiley sur le clavier ;
- **textPersonName** : saisie du nom d'une personne (affichage de speech to text en bas à gauche du clavier) ;
- **textPostalAddress** : saisie d'une adresse postale (affichage de speech to text en bas à gauche du clavier) ;
- **textPassword** : saisie de mot de passe ;
- **textVisiblePassword** : saisie d'un mot de passe visible ;
- **textWebEditText** : texte Web (activation de raccourci tabulation et speech to text) ;
- **number / numberSigned / numberDecimal / phone / datetime / date / time** : clavier numérique.

Vous pouvez mettre plusieurs valeurs en les séparant simplement par des "|".

II - Où est le champ suivant ?

Quand vous avez un clavier qui s'affiche, vous pouvez avoir l'envie qu'il soit lié au champ d'après, par exemple qu'un bouton **"suivant"** s'affiche au lieu du bouton **"retour à la ligne"**. Par défaut la valeur est **"Suivant"**, sinon **"Ok"** si vous êtes sur le dernier champ.

Mais vous pouvez choisir l'option que vous souhaitez voir affichée à l'aide de l'option **android:imeOptions**. Voici les différentes options disponibles :

- **actionUnspecified** : aucune spécification particulière, l'éditeur affichera ce qui lui semble le plus logique ;
- **actionNone** : aucune spécification particulière ;
- **actionGo** : affiche la valeur "Go" ;
- **actionSearch** : affiche la valeur "recherche" ;
- **actionSend** : affiche la valeur "envoyer" ;
- **actionNext** : affiche la valeur "suivant" ;
- **actionDone** : affiche la valeur "ok".

Vous pouvez récupérer l'action qui a été sélectionnée afin d'exécuter une action spécifique à l'aide de **setOnEditorActionListener()**. Vous allez recevoir un événement du type **IME_ACTION_SEND** quand le bouton **"envoyer"** sera cliqué

III - Mon clavier cache ma vue

Vous pouvez rencontrer un problème quand l'utilisateur clique sur un **EditText**, celui-là est caché par le clavier ou alors ce sont les champs en dessous qui ne sont plus visibles. Pour résoudre ce problème il vous faut jouer avec deux options :

- dans la partie correspondante à votre vue dans l'**AndroidManifest.xml**, vous pouvez ajouter **android:windowSoftInputMode="adjustResize"** qui vous permet de redimensionner votre vue quand le clavier apparaît. Pour plus de détails vous pouvez regarder sur le **site officiel d'Android** ;
- vous pouvez combiner cette option avec le poids des différents éléments pour votre vue : **android:weight**.

IV - Conclusion

Ce tutoriel se finit ici en espérant qu'il vous a aidé à comprendre le fonctionnement du clavier sous Android et comment faciliter l'interaction entre l'utilisateur et votre application.

V - Remerciements

Je tiens à remercier tout particulièrement **Feanorin** qui a mis ce tutoriel au format Developpez.com. Merci également à **_Max_** d'avoir pris le temps de le relire et de le corriger.

VI - Liens

Tutoriel origine sur Tutos-Android