Keyman Developer

Nous allons faire un petit tour de Keyman.

Claviers, Packages, Métadonnées

Comment ouvrir des fichiers Keyman.
Chaque clavier Keyman se compose de plusieurs fichiers.
https://help.keyman.com/developer/13.0/reference/file-types/[1]
Si quelqu'un vous à envoyé des fichiers Keyman, cherchez à ouvrir le fichier projet `.prj` s'il y en a, et le projet affichera comme il faut. Si vous ouvrez d'autres fichiers directement, vous risquez de confondre ou combinez les projets.

On ouvrant (ou après la création d')un projet en Keyman, il y aura plusieurs onglets disponibles.
La plupart du travail de développement des claviers passera derrière l'onglet Clavier (Keyboard), le développement d'un dictionnaire d'autocomplétion se passera sur l'onglet Modèle (Models), et on travaille avec le paquetage (packaging) quand on veut publier ou partager les claviers.

Survol des onglets de bases pour les claviers "desktop"

J’ai déjà un clavier ouvert.
Une fois que le clavier est ouvert, il y a plusieurs onglets disponibles.

Il y a l’onglet Détails, qui donne qui donne tous les détails du clavier à configurer.

Bien sûr, il y a le nom du clavier.
Après, il y a les cibles, et normalement, ce devrait être “tout”, et cela va essayer de produire une copie pour Windows, Android, macOS, Linux, etc. On peut le laisser su “any”.

Il faut remplir les droits d’auteur, et donner un tout-petit résumé d’un mode d’emploi.

Il ne faut cacher cette case que si le texte sorti du clavier est gauche à droite.

Plus bas, on voit le numéro de version de ce clavier.

Ici, on peut voir les fonctionnalités supplémentaires ajoutées au clavier.

Cette boîte commence vide, mais on peut voir que mon clavier a les styles CSS customisée, une icône personnalisée, le Clavier visuel de desktop, et finalement un clavier tactile.

Dans ces jours, il ne faut pas créer un clavier sans le clavier tactile.

Si on regarde le prochain onglet, nous avons “Layout”. Ceci commence en mode design au début, mais une fois qu’on commence à ajouter des touches mortes, des combinaisons, et d’autres choses complexes qui nous poussent à utiliser Keyman, on sera condamné à utiliser “Layout” en mode code pour l’avenir.

On commence avec les métadonnées du clavier.

Normalement, il y a un group "main" (primaire) qui contient des combinaisons de bas.

Quand on regarde le code, normalement le “input” est à gauche, ce qu’on tape, et la sortie est à droite. Si on tape majuscule + e, par exemple, mon clavier sort le “e” en majuscule.

Si les codes Unicode sont utilisés, on peut les cliquer, et l’identificateur des caractères nous montre le “vrai” symbole, ou “glyphe”. Cette présentation ne cible pas de vous explique comment écrire et débugger le clavier.

Si on utilise des touches mortes et des modifications supplémentaires, on verra d’autres groupes plus bas.

Sur l’onglet icône, on peut dessiner ou importer une icône.

L’onglet “on-screen” ou clavier visuel peut montrer à l’utilisateur la disposition des touches, ou lui laisser saisir avec la souris. Le clavier visuel n’est pas trop utile en tant que tel avec des touches mortes. Quand même, si on a utilisé plusieurs couches (comme normal, majuscule, Alt (ou) Ralt, et RALT + Maj) le clavier “On-Screen” peut être vite rempli en important les touches à partir du code.

Survol de l'interface pour les claviers tactiles

Quand on commence le clavier tactile (on screen), il y a des modèles disponibles (latin, latin plus et une autre), mais si on a déjà travaillé la Disposition (Layout) et clavier visuel, on peut tout de suite remplir le clavier tactile à partir du clavier visuel. Sauf si on n’avait aucune intention de sortir des claviers web ou desktop, c’est à cause de cet avantage d’importation qu’il faut que l’onglet layout et l’onglet visuel soient déjà assez stable avant de faire le clavier tactile.

Quand on importe le clavier, on verra quelque chose d'une forme très “desktop”. Ce n’est pas forcement mauvais pour les tablettes, mais il pourrait être très difficile à utiliser les petites touches avec de gros doigts.

Quand on parle de **Platforme** (platform) ici, ce n'est pas Android, Windows, etc. comme on aurait imaginé. Ce n'est que la taille de l'écran, téléphone ou tablette. Avec une tablette qui est plus grande qu'un téléphone, on a plus d'espace pour les touches. C’est recommandé par l’équipe d’avoir 11 touches ou moins sur chaque ligne pour les téléphones, même si j’en ai 12.

Note: Si on n'a pas deux plates-formes, ou si on en supprime en, Keyman adaptera la plateforme existante aux deux plateformes. Si vous voulez customiser l'expérience pour des grands et petits appareils, je vous suggère de créer et tester la plateforme tablette, faire une copie dans le code pour le téléphone, et essayer de le simplifier. Ce processus évitera la nécessité de faire le même travail deux fois.

Quand on parle des **layers** (couches), ce clavier a des couches spéciales, RALT et SHIFT+RALT…

…à continuer.

Soumettre un clavier:

Pour apprenre à soumettre un clavier, allez a Soumettre un clavier Keyman[2].

Keyman Developer Avancé:

Keyman Developer n'est pas en français, et la plupart des informations sont en anglais. Si vous comprenez l'anglais, toute les details sont sur ((https://help.keyman.com/developer/[3])) .

En plus, Il y a une serie des videos de 2020 qui passent de A à Z ici: ((https://keyman.com/training/[4]))

---

[1] https://help.keyman.com/developer/13.0/reference/file-types/

[2] https://outilingua.net/Soumettre+un+clavier+Keyman

[3] https://help.keyman.com/developer/

[4] https://keyman.com/training/