

GNG 2501

Livable J - Manuel d'utilisation

Soumis par:

[GROUPE FA3.2]

[FÉLIX LAROCHE, 300144353]

[MAHDI CHIBOUB, 300094626]

[MAYSSA TEBOURBI,300147817]

3 décembre 2020

Université d'Ottawa

Résumé

Ce rapport présente le manuel d'utilisation pour le travail que nous avons fait sur le clavier sur écran Onboard qui a été créé par les développeurs de l'hôpital Saint-Vincent à Ottawa. Il présentera d'abord les objectifs et principes derrière notre travail, ensuite toutes les informations reliées au manuel d'utilisation tel que les attributs importants du produit, ses fonctions et capacités, des instructions sur comment est-ce que le prototype final fut créé ainsi que des directives sur son installation et son utilisation. Finalement, une brève conclusion discutera des leçons apprises et des options pour le futur de notre travail.

Table des matières

Introduction	6
Pourquoi est-ce que c'est important?	6
Pour qui est-ce que c'est important ?	6
Pourquoi nous?	7
Manuel d'utilisation	8
Attributs du produit	8
Création du prototype final	9
Installer un environnement Ubuntu	9
Utiliser une VM Box	11
Utiliser WSL2	11
Modifier l'aspect graphique du clavier	12
Debugging	14
Liens importants!	14
Conclusions et recommandations pour travaux futurs	15
Bibliographie	16

Liste de figures

Figure 1 - Prototype final du clavier sur écran Onboard	7
Figure 2 - Disposition alphabétique du clavier sur écran Onboard	7
Figure 3 - Menu ajouté au clavier sur écran Onboard	8
Figure 4 - Exemple d'une partie du code XML pour la disposition alphabétique du clavier Onboard.	12

Liste des acronymes

Acronyme	Définition
WSL2	Windows Sub-system for Linux 2
VM Box	Virtual Machine Box

1 Introduction

1.1 Pourquoi est-ce que c'est important?

Notre travail est important car nous apportons une grande dose de simplicité au produit déjà en place. Après avoir passé beaucoup de temps à travailler et à faire de la recherche sur celui-ci, nous nous sommes rendu compte que le problème majeur de ce clavier sur écran est sa difficulté au niveau de l'utilisation, mais aussi au niveau de son installation. Pour un client régulier, donc un patient de l'hôpital Saint-Vincent, travailler avec ce type de produit pourrait certainement être un défi. Nous espérons que notre travail est le premier pas dans la direction de rectifier ce problème.

1.2 Pour qui est-ce que c'est important ?

Ce travail est principalement important pour les clients qui utilisent le clavier sur écran, donc principalement les patients de l'hôpital Saint-Vincent. Ces personnes ont besoin d'outils simples pour améliorer leur qualité de vie et, même si le clavier déjà mis en place par les développeurs de l'hôpital Saint-Vincent est de très bonne qualité, nous voulions contribuer à l'amélioration de celui-ci non seulement en changeant l'interface d'utilisation pour les clients, mais aussi en améliorant la facilité d'utilisation et de modification du clavier.

1.3 Pourquoi nous?

Car nous nous sommes concentrés sur un aspect du clavier qui, à notre avis, diffère beaucoup de ce que les autres équipes ont fait. De ce que nous avons entendu, certaines équipes qui travaillent sur le même projet que nous ont décidé, après avoir jeté un coup d'œil au clavier, de simplement en recréer un de zéro. Pour nous, il était important de travailler avec ce que notre client nous avait donné même si le code semblait effrayant à première vue. Nous avons vu ce projet comme étant une bonne opportunité pour apprendre à utiliser XML, SVG et avec des outils tels que GitHub et Bazaar, et ce fût certainement le cas. Les améliorations que nous avons faites au clavier sont simples, efficaces et vont grandement aider les travaux futurs sur ce produit, ce pourquoi nous considérons que notre travail se distingue des autres.

2 Manuel d'utilisation

Attributs du produit

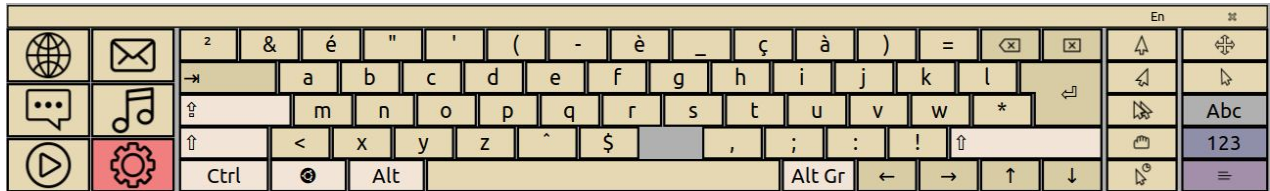


Figure 1 - Prototype final du clavier sur écran Onboard

Le clavier sur écran pouvait initialement permettre à une personne ayant des capacités motrices réduites d'utiliser un ordinateur à son plein potentiel en ayant accès à un clavier qui supporte non seulement le toucher tactile, mais aussi le "Eye Tracking". Ceci veut dire qu'une personne qui ne peut pas toucher à un clavier pourrait tout de même écrire des mots ou faire des recherches simplement en regardant certaines lettres du clavier. Il offre une grande variété de thèmes pour pouvoir personnaliser le clavier aux goûts de l'utilisateur et une fonctionnalité en cours de développement qui permet à l'utilisateur d'ajouter des raccourcis à certains sites webs ou applications pour faciliter la navigation de l'ordinateur.

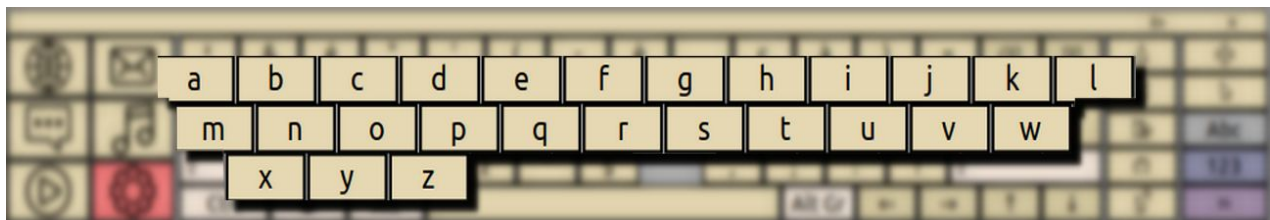


Figure 2 - Disposition alphabétique du clavier sur écran Onboard

Après notre travail, le clavier possède maintenant un mode alphabétique qui change l'apparence du clavier de QWERTY en ordre alphabétique. Ceci fut la première demande de notre client

puisque la plupart des utilisateurs ont beaucoup plus de facilité à travailler avec l'alphabet qu'avec une disposition QWERTY.



Figure 3 - Menu ajouté au clavier sur écran Onboard

Ensuite, le clavier comporte maintenant des boutons additionnels qui améliore théoriquement la facilité et la rapidité d'utilisation du clavier. Ces boutons sont composés d'un bouton Google qui permet à l'utilisateur d'accéder à une page Google instantanément, un bouton qui permet d'accéder à la messagerie, un bouton qui permet de clavarder en ligne, un bouton qui permet de faire jouer de la musique, un bouton qui permet de regarder des vidéos et un bouton pour accéder aux paramètres du clavier, Toutes ces nouvelles fonctions sont situées au même endroit, soit à la gauche du clavier dans une section que nous appelons 'menu'. Aucun de ces boutons n'est actuellement fonctionnel puisque notre tâche principale était de faire de la recherche pour savoir s'il était possible d'implémenter ce genre de fonctionnalité dans le code source existant, mais tout est en place pour que des étudiants futurs puissent les implémenter,

Création du prototype final

Installer un environnement Ubuntu

Le clavier sur écran Onboard se trouve initialement sur un repo Bazaar¹. On peut y retrouver l'entièreté du code source pour le clavier sur écran. Pour commencer à faire nos modifications, la

¹ <https://launchpad.net/onboard>

première étape fut donc d'avoir accès à celui-ci sur nos propres ordinateurs. Le clavier sur écran sur lequel nous avons travaillé fonctionne seulement sur un système Linux Ubuntu. De base, celui-ci réside sur Raspberry Pi et l'utilisateur n'a qu'à connecter le Pi dans son ordinateur et lancer l'application pour pouvoir l'utiliser, mais dans le cas où quelqu'un veut utiliser le clavier ou travailler sur celui-ci mais n'a pas de Raspberry Pi ou de système Linux, deux options s'offrent à lui/elle : Option 1, télécharger une VM Box² (Virtual Machine Box) sur son ordinateur, installer Ubuntu, télécharger l'application et l'exécuter sur cet environnement. Cette option est relativement simple car télécharger une VM Box n'est pas super compliqué, mais le désavantage vient du fait qu'on n'utilise pas le clavier sur l'environnement Windows habituel. Cependant, si on veut travailler avec le code source du clavier, c'est une alternative plutôt simple. Pour installer la VM Box, simplement suivre le tutoriel qui est sur le lien en bas de page. La deuxième option est d'installer WSL³ (Windows Sub-system for Linux) sur son ordinateur Windows pour avoir accès à Ubuntu. Cette fonctionnalité est plus difficile à installer que l'option 1 (surtout si vous êtes relativement débutant/avez peu d'expérience à modifier les paramètres d'un ordinateur), mais donne un plus grand degré de liberté pour modifier le code source et autre. De plus, elle permet d'avoir accès au clavier en restant sur un environnement Windows, ce qui est un léger avantage comparé à l'option 1. Cependant, ces deux options ont en commun qu'elles sont gratuites, ce pourquoi nous avons décidé de les utiliser au lieu d'acheter un Raspberry Pi.

Après avoir trouvé un moyen d'avoir accès à un système Linux, la prochaine chose à faire était d'avoir accès au clavier Onboard. Selon quelle option on choisit, les étapes pour bien installer le

² <https://itsfoss.com/install-linux-in-virtualbox/>

³ <https://docs.microsoft.com/en-us/windows/wsl/install-win10>

clavier et avoir accès au code source et au clavier lui-même sont différentes. Nous avons utilisé les deux méthodes pour travailler sur notre projet,

Utiliser une VM Box

Après que la VM Box soit installée et qu'on ait accès à un système d'exploitation Linux Ubuntu, la prochaine étape est simplement de soit télécharger le code source depuis leur site original (<https://launchpad.net/onboard>) en cliquant sur le bouton "download" pour télécharger la dernière version du software qui fonctionne. Il ne reste plus qu'à lancer le fichier .exe qui vient avec le téléchargement tout en étant dans l'environnement virtuel pour lancer l'application. Cependant, en faisant cela, on télécharge une version qui ne contient pas les nouvelles fonctionnalités que nous avons ajoutées au clavier. Maintenant qu'une version mise à jour du code source est disponible sur GitHub⁴ (grâce à nous), une autre option est de simplement fork de ce repo GitHub et clone le code sur son ordinateur personnel pour y avoir accès et le faire fonctionner. Il y aura deux branches, une master qui est le code de base avec seulement le fichier README de modifier et une branche dev qui possède les fichiers XML et SVG que nous avons modifiés/ajoutés. Ensuite, simplement build le code en suivant les directives situées dans le fichier README.md pour générer un fichier exécutable et ainsi pouvoir lancer le clavier dans un environnement Ubuntu.

⁴ <https://github.com/flarouche3129/OnboardKeyboardUOttawa>

Utiliser WSL2

Pour WSL2, à moins d'avoir de l'expérience avec Linux, c'est un peu plus compliqué d'installer le clavier. Lorsqu'on a WSL2 fonctionnel et qu'on a un terminal Ubuntu sous la main, la prochaine étape est d'aller chercher le package Onboard. Onboard est une application que l'on peut retrouver sur Launchpad. Launchpad est une plateforme de services en ligne permettant le travail collaboratif entre différentes personnes et groupes sur des projets open-sources (un peu comme GitHub) qui fonctionne sur Ubuntu. Cela veut dire qu'on peut trouver les fichiers de code source simplement en tapant **sudo apt install onboard** dans notre terminal ubuntu. Cela ira chercher tous les fichiers requis pour faire fonctionner le clavier sur écran et les installera dans un dossier appelé "onboard". Ensuite, on doit suivre les mêmes étapes que pour la VM Box et suivre le fichier README.md pour s'assurer que toutes les dépendances soient bien installées, mais, pour que le clavier puisse vraiment apparaître, on doit aussi installer Xming et PuTTY⁵ (simplement suivre le tutoriel mis en bas de page) pour pouvoir lancer des applications à partir de WSL2. Finalement, simplement ouvrir un terminal dans le nouveau dossier onboard (vous devriez pouvoir voir tous les fichiers requis pour faire fonctionner l'application dans ce dossier) et écrire **./onboard** pour lancer l'application. Ces étapes sont aussi valides si on veut simplement fork tu repo GitHub. Au lieu d' exécuter la commande pour aller chercher les fichiers sources sur launchpad, on a qu'à clone le repo GitHub sur un ordinateur personnel

⁵ http://www.geo.mtu.edu/geoschem/docs/putty_install.html

Modifier l'aspect graphique du clavier

Après avoir installé le clavier et d'avoir accès au code source, notre prochaine étape fut de modifier l'aspect graphique de ce dernier. Pour ce faire, nous avons utilisé deux logiciels gratuits que nous avons trouvés en ligne appelés Glade⁶ et Inkscape⁷ (principalement Inkscape) pour pouvoir faire des changements à l'aspect graphique du clavier et générer des fichiers XML et SVG pour pouvoir les incorporer au code source du clavier.

⁶ <https://inkscape.org/fr/>

⁷ <https://glade.gnome.org/>

```

<keyboard id="Alphabetical" format="3.2" section="system" summary="Medium size desktop keyboard">
  <include file="key_defs.xml"/>
  <box border="0.5" spacing="1.5" orientation="vertical">
    <!-- word suggestions -->
    <panel filename="Alphabetical-Alpha.svg" scan_priority="1">
      <include file="word_suggestions.xml"/></panel>
    <box spacing="1.5">
      <box spacing="2">
        <!-- click helpers -->
        <panel id="menu" filename="Alphabetical-Alpha.svg">
          <key group="menu" id="internet"/>
          <key group="menu" id="music"/>
          <key group="menu" id="video"/>
          <key group="menu" id="email"/>
          <key group="menu" id="chat"/>
          <key group="menu" id="setting"/>
        </panel>
        <!-- keyboard, multiple layers -->
        <panel>
          <panel layer="alpha" filename="Alphabetical-Alpha.svg">
            <key group="alphanumeric" id="AB01"/>
            <key group="alphanumeric" id="AE02"/>
            <key group="alphanumeric" id="AE03"/>
            <key group="alphanumeric" id="AD09"/>
            <key group="alphanumeric" id="AE01"/>
            <key group="alphanumeric" id="AE06"/>
            <key group="alphanumeric" id="AE07"/>
            <key group="alphanumeric" id="AE04"/>
            <key group="alphanumeric" id="AE05"/>
            <key group="alphanumeric" id="AD03"/>
            <key group="alphanumeric" id="AD02"/>
            <key group="alphanumeric" id="AD01"/>
            <key group="alphanumeric" id="AE09"/>
            <key group="alphanumeric" id="AD07"/>
            <key group="alphanumeric" id="AD06"/>
            <key group="alphanumeric" id="AD05"/>
            <key group="alphanumeric" id="AD04"/>
            <key group="alphanumeric" id="AB10"/>
            <key group="alphanumeric" id="AC11"/>
            <key group="alphanumeric" id="AC10"/>
            <key group="alphanumeric" id="TLDE"/>

```

Figure 4 - Exemple d'une partie du code XML pour la disposition alphabétique du clavier Onboard. 12

Ceci n'est qu'un aperçu de ce à quoi ressemble le fichier XML généré par Inkscape lorsque nous avons modifié l'apparence QWERTY du clavier à alphabétique. Cela nous a été très utile puisque nous avons pu modifier en temps réel l'apparence du clavier tout en voyant au fur et à mesure à quoi ressemblait notre travail. Puis, lorsque nous étions satisfait avec l'apparence du clavier, on

n'avait qu'à ajouter la disposition dans le dossier "layouts" du Onboard pour y avoir accès lorsqu'on lançait l'application une nouvelle fois. Nous devons aussi inclure les fichiers SVG générés par Glade. Tous ces fichiers sont situés dans notre GitHub repo sous le dossier "layouts".

Debugging

Lors de l'installation du clavier Onboard, nous avons eu beaucoup de difficultés au niveau des dépendances de celui-ci. Lorsque nous avons suivi le fichier README tel quel (surtout en utilisant la méthode WSL2), les commandes à exécuter pour aller chercher toutes les dépendances nécessaires pour le clavier si on est sur Ubuntu n'ont pas fonctionné pour nous. Nous ne sommes pas à 100% certain pourquoi (ce pourquoi nous conseillons tout de même de suivre le fichier), mais nous soupçonnons que ce soit à cause du fait que la plupart des liens soient rendus trop vieux et que les développeurs ne les ont simplement pas mis à jour. Nous avons inclus une note dans le fichier README de la commande que nous avons exécutée pour compenser des vieilles commandes de dépendances. Ceci fut le seul vrai problème que nous avons eu lors de nos tests et lors de l'installation du clavier.

Liens importants!

Tous les fichiers sur lesquels nous avons travaillé se trouvent sur notre repo GitHub (<https://github.com/flarouche3129/OnboardKeyboardUOttawa>) et tous les travaux que nous avons fait pour notre cours GNG2501 se trouve sur notre MakerRepo (<https://makerepo.com/mayssatbr/clavier-sur-cran-fa32?fbclid=IwAR2lbeKh1e0wmspAtghVqijhEnPU83Cx11nIdkiizgYQfoMeWfUqW7BVC10>).

3 Conclusions et recommandations pour travaux futurs

Lors de notre session, nous avons rencontré plusieurs problèmes qui ont rendu cette tâche très difficile. De membre qui quitte le cours à membres qui tombent malade pendant plusieurs semaines, nous avons dû travailler très fort pour essayer de produire quelque chose dont nous étions fiers. Nous avons appris à travailler avec XML et SVG à travers des outils de conception graphique, comment travailler sur une application déjà existante de grande envergure tout en essayant de garder le même degré de qualité, comment utiliser une interface Linux, comment manipuler Bazaar et comment travailler en groupe dans une situation de pandémie. Notre seul regret est que nous n'avons pas eu le temps d'implémenter les boutons, mais toute la recherche que nous avons fait se trouve sur notre repo GitHub. Donc, pour les futurs étudiants, tout ce qu'il reste à faire est d'implémenter les boutons que nous avons créés et les faire fonctionner de manière efficace.