



HAL
open science

Conception d'une méthode de saisie phonologique pour le cantonais en utilisant l'Alphabet Phonétique International (API) Parcours : Traitement Automatique de Langue

Lian Chen

► **To cite this version:**

Lian Chen. Conception d'une méthode de saisie phonologique pour le cantonais en utilisant l'Alphabet Phonétique International (API) Parcours : Traitement Automatique de Langue. INALCO. 2024, pp.1-14. hal-04819858v1

HAL Id: hal-04819858

<https://hal.science/hal-04819858v1>

Submitted on 4 Dec 2024 (v1), last revised 12 Dec 2024 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Conception d'une méthode de saisie phonologique pour le cantonais en utilisant l'Alphabet Phonétique International (API)

Parcours : Traitement Automatique de Langue

CHEN Lian 陈恋

30/11/2024

Sommaire

1. Introduction

2. Analyse des IME existantes

Comparaison des avantages et inconvénients des outils Keymanweb, Lexilogos et Google Input Tools.

3. Conception de la méthode de saisie pour le cantonais

3.1. Inventaire phonologique du cantonais : consonnes, voyelles et tons.

3.2. Caractères API non accessibles via un clavier AZERTY.

3.3. Codage Unicode des caractères non accessibles.

3.4. Méthode de saisie conçue pour taper efficacement les phonèmes.

4. Implémentation de la méthode de saisie

4.1. Présentation de l'outil Keyman Developer.

4.2. Fichier de configuration pour le cantonais en API.

4.3. Expériences et résultats.

5. Conclusion

Conception d'une méthode de saisie phonologique pour le cantonais en utilisant l'Alphabet Phonétique International (API)

CHEN Lian

30 Novembre 2024

Contents

1	Introduction	3
2	Analyse des IME existantes	3
3	Conception de la méthode de saisie pour le cantonais	7
3.1	Inventaire phonologique du cantonais : consonnes, voyelles et tons.	7
3.2	Codage Unicode des signes non accessibles	9
3.3	Méthode de saisie conçue pour taper efficacement les phonèmes.	9
4	Implémentation de la méthode de saisie	10
5	Conclusion	13

1 Introduction

Dans ce rapport, nous explorons la conception et l'implémentation d'une méthode de saisie adaptée à une langue spécifique, en utilisant l'Alphabet Phonétique International (API). Ce projet poursuit un double objectif : analyser les forces et les limites des solutions de saisie existantes, et concevoir une méthode innovante adaptée à un système phonologique particulier.

Nous avons choisi de travailler sur le cantonais, une langue caractérisée par sa richesse tonale et ses particularités phonétiques (neuf tons répartis en six registres distincts). Le cantonais présente des défis spécifiques en termes de saisie, notamment en raison de son inventaire complexe de phonèmes, comprenant des consonnes et voyelles absentes du mandarin, de ses tons multiples, et de son système d'écriture basé principalement sur les caractères chinois traditionnels. L'objectif de ce projet est de développer une méthode de saisie intuitive et efficace pour transcrire les phonèmes cantonais en API, en exploitant des outils comme Keyman Developer.

Ce rapport est structuré en trois parties :

Une première section examine le fonctionnement et les principes d'implémentation des outils de saisie existants.

Nous y détaillons ensuite une approche méthodique basée sur l'analyse phonologique et les contraintes techniques, afin de concevoir une méthode de saisie en API adaptée au cantonais.

Enfin, une troisième partie décrit la création et le test d'un fichier de configuration pour Keyman Developer, traduisant notre conception en une application pratique.

2 Analyse des IME existantes

Nous analysons tout d'abord les méthodes de saisie informatiques existantes, appelées IME (Input Method Editors), afin de mieux comprendre leurs fonctionnalités, leurs forces et leurs limites. Ces outils jouent un rôle crucial dans la saisie de langues utilisant différents systèmes d'écriture, en particulier lorsque ces systèmes nécessitent un grand nombre de caractères ou des combinaisons complexes de symboles.

Cette étude s'appuie sur trois solutions de saisie largement utilisées : Keymanweb, Lexilogos et Google Input Tools, et se concentre sur leur application à quatre langues aux systèmes d'écriture variés : le vietnamien (alphabet latin), l'ukrainien (alphabet cyrillique), le hindi (devanāgarī) et le birman (alphasyllabaire birman). Ces langues ont été choisies pour illustrer la diversité des besoins en matière de saisie linguistique et les défis spécifiques associés à chaque système d'écriture.

Cette analyse servira de base à la conception ultérieure d'une méthode de saisie personnalisée dans le cadre de ce projet. Pour ce faire, nous effectuons une comparaison globale des trois outils de saisie, en identifiant leurs différences, leurs avantages et leurs inconvénients, après avoir testé ces solutions sur les quatre langues sélectionnées.

Voici les résultats que nous avons obtenus sur ces trois plateformes pour les quatre langues :

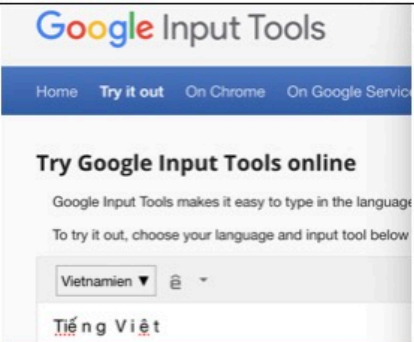








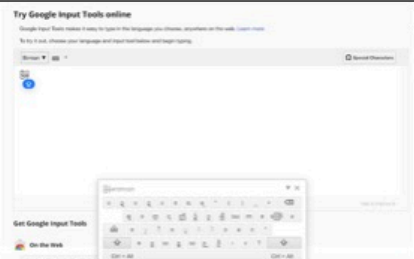
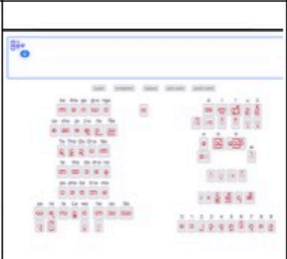

Langues	Google input tools	Lexilogos	KeymanWeb.com
vietnamien			
ukrainien			
hindi			
birman			<p>Cette langue n'est pas disponible.</p>
Exemple d'écriture sur écran tactile.			

Figure 1: Capture d'écran des résultats des trois plateformes pour les quatre langues testées.

Comme illustré par les captures d'écran ci-dessus, nous avons testé ces quatre langues à l'aide des trois outils de saisie en ligne. Chacun des outils permet de taper directement

avec un clavier physique standard après avoir sélectionné une langue spécifique. De plus, les caractères spéciaux sont facilement accessibles via le clavier virtuel en ligne, ce qui rend ces outils compatibles avec un large éventail de langues.

Personnellement, je trouve que les trois outils offrent une interface confortable. Keyman et Google Input Tools utilisent un clavier standard, tandis que Lexilogos propose une disposition qui sépare les consonnes et les voyelles. Leurs interfaces sont claires et bien organisées, avec un clavier de taille idéale, ce qui facilite son utilisation même pour les personnes qui ne connaissent pas la langue ou qui sont des débutants en apprentissage.

Ces outils se distinguent également par leur personnalisation étendue des claviers, permettant d'adapter la saisie à de nombreux systèmes d'écriture. Ils soutiennent une large communauté linguistique et bénéficient de mises à jour régulières. Leur interface simple convient particulièrement aux utilisateurs novices.

Il convient également de noter que les trois outils nécessitent une connexion Internet, ce qui peut poser problème en cas d'absence de réseau. Par ailleurs, ces outils peuvent nécessiter un certain temps d'adaptation pour les utilisateurs peu familiers, notamment en raison des dispositions spécifiques des claviers.

Après avoir examiné les quatre langues, nous avons constaté que Keymanweb se distingue par sa classification des langues par continent, suivie d'un classement alphabétique. Son interface est claire et facile à utiliser, ce qui rend la recherche plus intuitive et permet une saisie rapide de mots simples. Cependant, cette classification peut compliquer la navigation : par exemple, le vietnamien est classé sous les langues asiatiques, puis dans la section alphabétique sous la lettre V. En revanche, les deux autres outils classent directement les langues par ordre alphabétique.

Par ailleurs, Keymanweb n'est pas entièrement exhaustif. Certains caractères spécifiques au vietnamien, comme *ê* ou *ê*, ne sont pas disponibles, tandis que Lexilogos et Google Input Tools prennent en charge ces caractères. De plus, la langue birmane n'est pas prise en charge par Keymanweb.

Google Input Tools propose un système intelligent qui fournit des suggestions basées sur la fréquence des mots, ce qui accélère la saisie. De plus, il offre une fonctionnalité de saisie manuscrite, particulièrement utile pour les caractères complexes. Intégré de manière transparente aux autres services Google, cet outil est facile à utiliser et inclut des fonctionnalités de saisie prédictive. Son interface intuitive facilite la reconnaissance rapide des mots et rend leur saisie encore plus efficace.

Bien que pratique, Google Input Tools propose moins d'options de personnalisation par rapport à Keymanweb. Il ajoute automatiquement des espaces après chaque lettre saisie, ce qui peut rendre la saisie laborieuse, car il nécessite souvent des corrections manuelles. Les suggestions automatiques peuvent entraîner des erreurs, surtout si l'utilisateur ne vérifie pas attentivement les caractères. Lexilogos prend en charge un grand nombre de caractères spéciaux, ce qui le rend idéal pour des besoins variés. Il offre des options détaillées pour les consonnes et les voyelles, ainsi que des alphabets latins, ce qui en fait un bon support pour l'enseignement. De plus, Lexilogos est polyvalent et inclut

des outils linguistiques supplémentaires, tels que des dictionnaires et des traducteurs, ce qui le rend attrayant pour une utilisation occasionnelle.

Néanmoins, la navigation sur Lexilogos peut être complexe en raison d'une surcharge visuelle. Pour accéder à un clavier spécifique, il faut d'abord parcourir de nombreuses options de langues, ce qui diminue la fluidité de l'expérience utilisateur.

La saisie peut être plus lente, car l'abondance d'options peut provoquer des hésitations. Personnellement, j'ai constaté que l'utilisation de Lexilogos me prend plus de temps par rapport aux deux autres outils que j'utilise, car ce n'est pas le clavier avec lequel je suis habituellement familiarisé. En résumé, bien que Lexilogos offre de nombreux avantages, son interface moins conviviale et le temps nécessaire pour naviguer à travers ses options peuvent nuire à l'expérience utilisateur.

En résumé, Keymanweb est idéal pour une saisie rapide et intuitive, bien qu'il soit moins complet pour certaines langues. Google Input Tools se distingue par son intelligence et ses fonctionnalités avancées, mais l'ajout automatique d'espaces peut nuire à la fluidité de la saisie. Enfin, Lexilogos est le plus exhaustif en termes de support linguistique, mais son interface peu ergonomique peut ralentir le processus de saisie.

Cette comparaison des systèmes d'écriture illustre que chaque langue présente des défis spécifiques en matière de saisie, nécessitant des outils adaptés pour optimiser à la fois la productivité et le confort de l'utilisateur.

3 Conception de la méthode de saisie pour le cantonais

3.1 Inventaire phonologique du cantonais : consonnes, voyelles et tons.

Le tableau ci-dessous illustre comment l'alphabet phonétique international (API) représente la prononciation cantonaise <https://fr.wikipedia.org/wiki/Wikip%C3%A9dia>. La case en rouge indique les phonèmes qui ne peuvent pas être saisis avec un clavier AZERTY classique lorsqu'on utilise l'API.

Consonnes

		Bilabiales	Dentales		Palatales	Vélaires		Glottales
			Non-affriquées	Affriquées		Consonnes non-arrondies	Consonnes arrondies	
Nasales		m	n			ŋ		
Occusives	Non-aspirées	p	t	ts		k	k ^w (w)	ʔ
	Aspirées	p ^h	t ^h	ts ^h		k ^h	k ^{wh} (w)	
Fricatives		f	s					h
Spirantes		w			j			
Fricatives latérales			ʃ ^N					
Spirantes latérales			l					

Figure 2: Illustration de l'API pour la prononciation des consonnes en cantonnais.

Voyelles et combinaisons

Noyaux / Codos ¹	i	e	ɛ	y	œ	ø	ɐ	a	ɔ	o	u
∅	i		ɛ	y	œ			a	ɔ		u
i/y ²		ei				øy	ɛi	ai	ɔy		uy
u	iu		ɛu ^N				ɛu	au		ou	
m	im		ɛm				ɛm	am			
n	in			yn		ɛn	ɛn	an	ɔn		un
ŋ		ɛŋ	ɛŋ		œŋ		ɛŋ	aŋ	ɔŋ	oŋ	
p	ip		ɛp				ɛp	ap			
t	it		ɛt	yt	œt	øt	ɛt	at	ɔt		ut
k		ek	ɛk		œk		ɛk	ak	ɔk	ok	

Tons

Tons	55	51	15	14	33	11	³¹ Note 1	13	22
API	˥	˨	˧	˦	˩	˨˩	˨˥	˧˥	˨˨

Figure 3: Illustration de l'API pour la prononciation des voyelles et des tons en cantonnais.

Les cases rouges représentent des consonnes, des voyelles et des tons spécifiques au cantonnais qui ne peuvent pas être directement saisies à l'aide d'un clavier AZERTY.

3.2 Codage Unicode des signes non accessibles

Selon le site Unicode Table et le site Cantonese Tools (<https://www.cantonese-tools.org/fr/convert-cantonese-chinese-into-unicode>), nous avons obtenu les codes Unicode pour les signes mentionnés :

Consonnes :

ŋ: U+014B

[tʃ]: U+0074 + U+032A

[tʃ̂]: U+0074 + U+032C + U+0073

[tʃʰ]: U+0074 + U+032A + U+02B0

[tʃʰ̂]: U+0074 + U+032C + U+0073 + U+02B8

[ʃ]: U+026C

Voyelles :

[œ]: U+0153

[ø]: U+0275

[ɘ]: U+0250

Tons

[˥]: U+02E5

[˥˥]: U+02E5 + U+02E1

[˥˥˥]: U+02E1 + U+02E5

[˥˥˥˥]: U+02E4

[˥˥˥˥˥]: U+02E1

3.3 Méthode de saisie conçue pour taper efficacement les phonèmes.

Lors de la conception de mes claviers cantonnais dans Keyman Developer, je commence par planifier les caractères nécessaires, notamment les consonnes qui ne peuvent pas être saisies via l'API, en utilisant des outils comme la Character Map pour identifier leurs codes Unicode. Par exemple, j'inclus des sons spécifiques comme ŋ (U+014B), ainsi que des groupes de consonnes telles que [tʃ], [tʃ̂], etc. qui nécessitent des combinaisons de frappes pour être correctement représentés. De même, j'intègre les voyelles particulières du cantonnais, telles que [œ] (U+0153) et [ɘ] (U+0250), en veillant à ce que le clavier puisse gérer les tons, avec des caractères Unicode pour chaque tonalité. Après avoir défini les caractères à inclure, j'écris l'en-tête du fichier du clavier, avec des commentaires pour expliquer son fonctionnement, ainsi que des stores pour spécifier le nom, la version et l'icône du clavier. Ensuite, je me concentre sur le corps du clavier, structuré en groupes de règles qui définissent les réponses aux frappes. Certaines règles sont simples, reliant directement une touche à un caractère, tandis que d'autres sont plus complexes et tiennent compte du contexte des frappes précédentes. Pour alléger le nombre de règles nécessaires, j'utilise des stores pour regrouper des caractères similaires, ce qui rend les règles plus concises. Une fois les règles finalisées et le clavier compilé, je le teste minutieusement pour garantir que tous les caractères s'affichent correctement dans divers contextes. Finalement, il

est primordial de résoudre tout problème, comme une conversion incorrecte de voyelles dans des phrases, afin d'assurer un fonctionnement fluide et efficace du clavier pour les utilisateurs quotidiens.

Voici le code que j'ai conçu pour mon clavier cantonais :

```

Cantonese.kmn
1 store(&Version) "9.0"
2 c Ce clavier est pour une utilisation avec Keyman 9.0
3 store(&Name) "Cantonese Phoneme"
4 store(&Bitmap) "cantonese.ico"
5 store(&MnemonicLayout) "1"
6
7 group(main) using keys
8 c Ce clavier utilise une disposition mnémotechnique.
9
10 begin Unicode > use(main)
11
12 c Règles pour les phonèmes
13 "n" + "g" > "ŋ" c Entrée "ng" produit [ŋ]
14 "t" + "_" > "t̚" c Entrée "t_" produit [t̚]
15 "t" + "s" > "t̚s" c Entrée "ts" produit [t̚s]
16 "t" + "h" > "t̚h" c Entrée "t_h" produit [t̚h]
17 "s" + "h" > "t̚s̚h" c Entrée "s_h" produit [t̚s̚h]
18 "l" + "_" > "l̚" c Entrée "l_" produit [l̚]
19
20 c Règles pour les nouveaux phonèmes
21 "o" + "e" > "œ" c Entrée "oe" produit [œ]
22 "o" + "_" > "ø" c Entrée "o_" produit [ø]
23 "a" + "_" > "ɛ" c Entrée "a_" produit [ɛ]
24
25 c Règles pour les symboles tonals
26 "t" + "1" > "ᵿ" c Entrée "t1" produit [ᵿ] (ton élevé)
27 "t" + "2" > "ᵿᵿ" c Entrée "t2" produit [ᵿᵿ] (ton élevé-descendant)
28 "t" + "3" > "ᵿᵿ" c Entrée "t3" produit [ᵿᵿ] (ton descendant-élevé)
29 "t" + "4" > "ᵿ" c Entrée "t4" produit [ᵿ] (ton moyen)
30 "t" + "5" > "ᵿ" c Entrée "t5" produit [ᵿ] (ton bas)
31 "t" + "6" > "ᵿᵿ" c Entrée "t6" produit [ᵿᵿ] (ton bas-moyen)

```

Figure 4: Illustration de l'API pour la prononciation cantonaise.

4 Implémentation de la méthode de saisie

Keyman Developer est un logiciel puissant et flexible développé par Tanaka Keyboard pour la création et la gestion de claviers multilingues. Il permet aux utilisateurs de concevoir des dispositions de clavier personnalisées pour différentes langues et systèmes d'écriture, facilitant ainsi la saisie de caractères spéciaux et accentués qui ne sont pas disponibles sur les claviers standard. L'interface de Keyman Developer est conviviale

et intuitive, offrant une variété d'outils pour planifier, créer et tester des claviers. Les utilisateurs peuvent dessiner la disposition du clavier, définir des règles de saisie et intégrer des caractères Unicode, tout en ayant la possibilité de visualiser en temps réel les modifications apportées. De plus, Keyman Developer prend en charge l'importation de fichiers de configuration, ce qui permet de simplifier le processus de création. L'outil propose également un environnement de test intégré qui permet aux développeurs de vérifier rapidement le fonctionnement de leurs claviers avant de les déployer. En outre, Keyman Developer est compatible avec plusieurs systèmes d'exploitation, notamment Windows (voir plus tard avec macOS,), et il offre une large gamme de fonctionnalités permettant une personnalisation avancée des claviers. Keyman Developer est un outil indispensable pour les linguistes, les développeurs et tous ceux qui souhaitent créer des solutions de saisie adaptés à des besoins linguistiques spécifiques, rendant la communication écrite accessible dans de nombreuses langues à travers le monde.

Nous avons ensuite testé notre clavier sur notre ordinateur Windows (Mac ne fonctionne pas très bien). Voici les expériences et résultats :

4 IMPLÉMENTATION DE LA MÉTHODE DE SAISIE

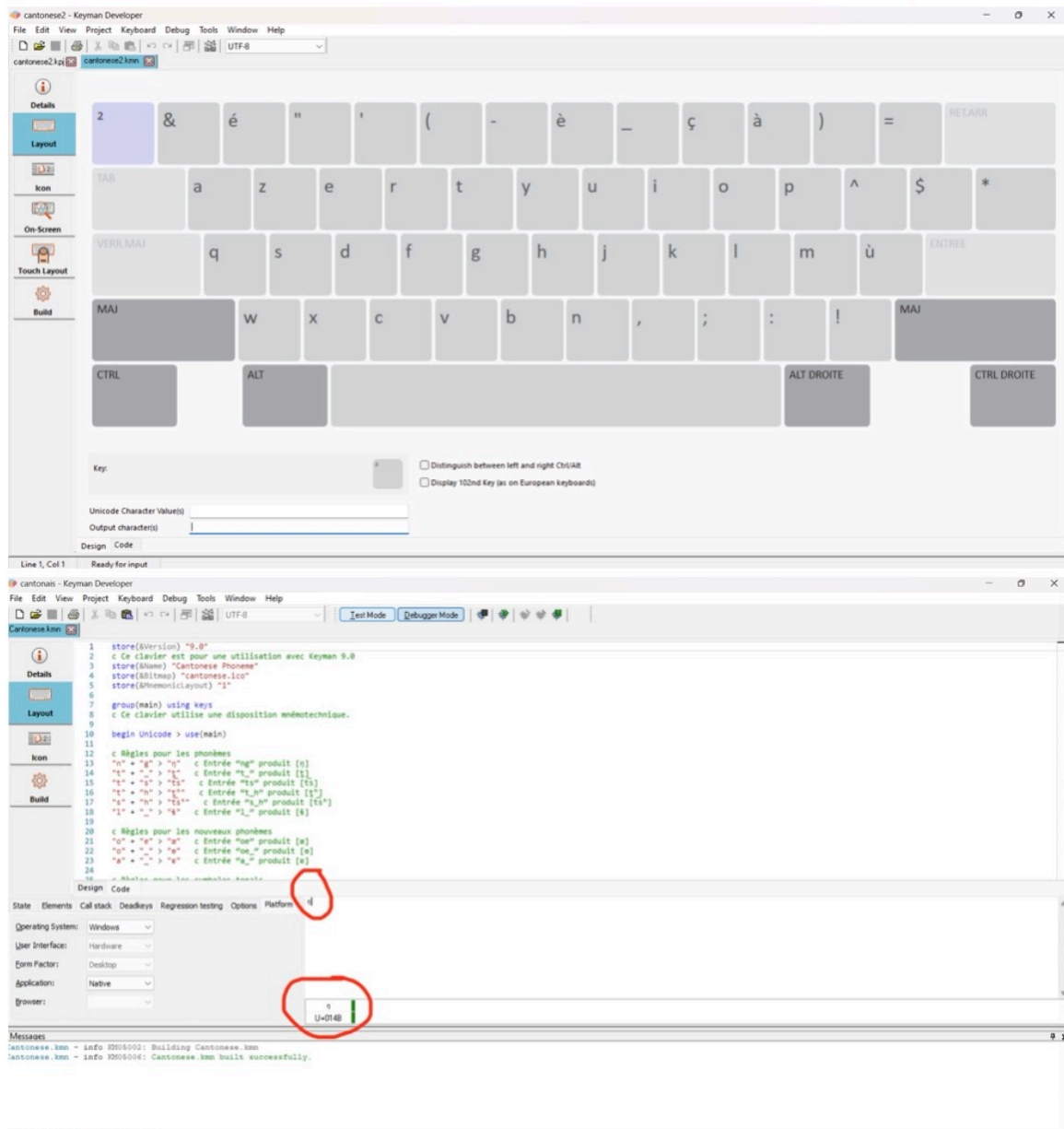


Figure 5: Illustration du code pour la saisie au clavier cantonais

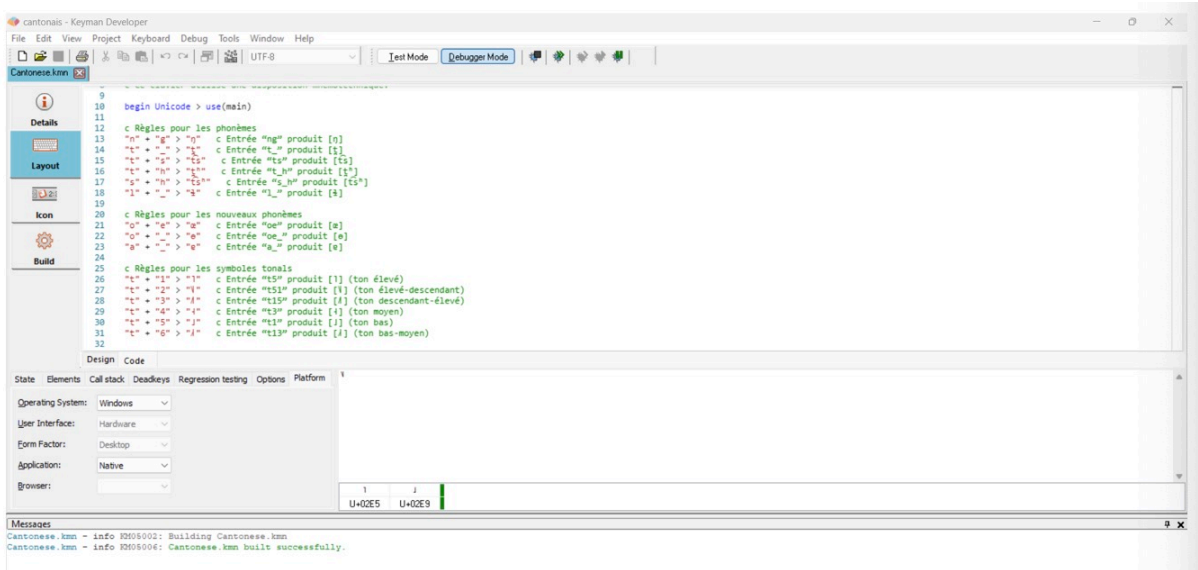


Figure 6: Illustration des résultats des tests concernant le code de saisie pour le clavier cantonais

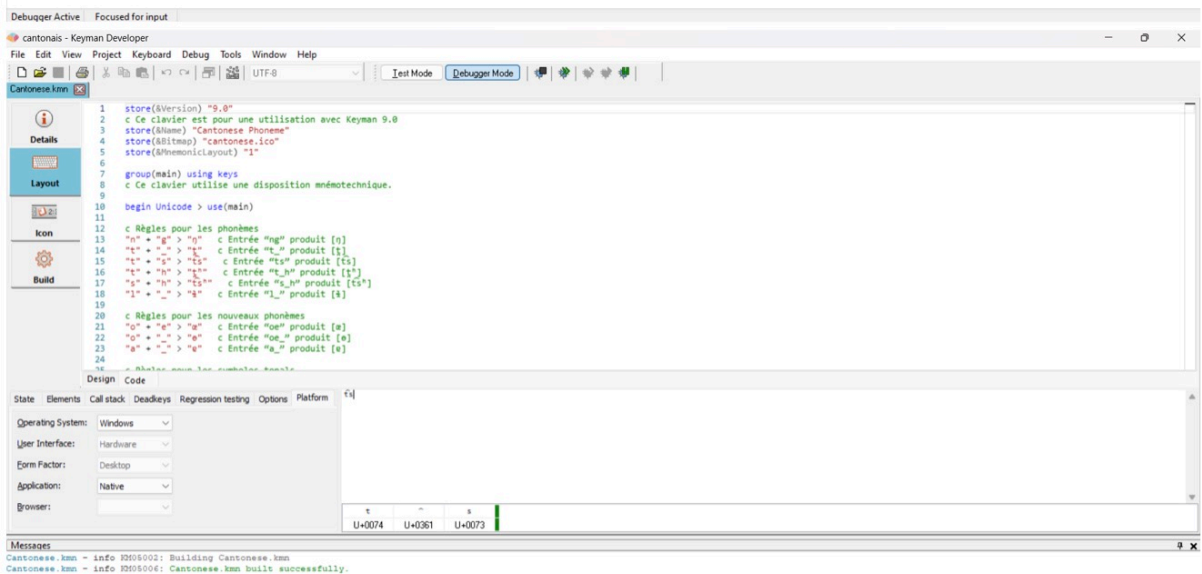


Figure 7: Illustration des résultats des tests concernant le code de saisie pour le clavier cantonais

5 Conclusion

Dans le cadre de ce projet, nous avons conçu une méthode de saisie phonologique pour le cantonais en utilisant l'Alphabet Phonétique International (API). Après une analyse approfondie des méthodes de saisie existantes et des défis spécifiques posés par la langue cantonaise, nous avons développé un clavier adapté qui facilite la saisie des consonnes, voyelles et tons caractéristiques de cette langue. Grâce à l'utilisation de Keyman Developer, nous avons pu intégrer un codage Unicode et des règles de saisie intuitives qui rendent l'expérience utilisateur fluide et efficace. Les tests réalisés ont montré que

notre clavier répond aux attentes, permettant une transcription précise des phonèmes cantonais. Ce projet représente une avancée significative dans l'accessibilité de l'écriture cantonaise, ouvrant la voie à une meilleure communication écrite pour les utilisateurs de cette langue.

References

- [1] Culture.gouv.fr. *Portail Langues en France: Cantonais*. <https://www.culture.gouv.fr/Thematiques/portail-langues-en-france/langues/cantonais>.
- [2] Wikipedia. *Aide: Alphabet phonétique cantonal*. https://fr.wikipedia.org/wiki/Aide:Alphabet_phon%C3%A9tique_cantonais.
- [3] Unicode Table. *Unicode Table*. <https://unicode-table.com/>.
- [4] Keyman. *Command Line Guide*. <https://help.keyman.com/developer/14.0/guides/command-line>.
- [5] Keyman. *Development Tutorial*. <https://help.keyman.com/developer/14.0/guides/develop/tutorial/>.
- [6] GitHub. *Keyman Web*. <https://github.com/keymanapp/keymanweb.com>.