

AVTOMATIZIRANA PRETVORBA ZNAKOV ARABSKE ABECEDDE IZ KODNEGA SISTEMA 0600-06FF V KODNI SISTEM PREZENTACIJSKE OBLIKE FE70-FEFF

Saša Klampfer, Peter Šamperl, Zdenko Mezgec, Amor Chowdhury

Margento R&D d.o.o., Maribor, Slovenija

Ključne besede: mPOS, kodni sistemi, arabska abeceda, arabski znaki, pretvorbe, translacijska matrika, pravila, plačilni terminal, zaslon, hexadecimalni zapis, inverzna pretvorba, terminalsko orodje, translacijsko orodje, enostopenjsko pravilo, večstopenjsko pravilo.

Izleček: Razširjenost brezgotovinskega plačevanja z mobilnimi telefoni preko mPOS plačilnih terminalov postaja vse bolj aktualno in priročno. Razvite rešitve se lansirajo tako na domače, kot tuje trge. Z lansiranjem rešitev na tržišča bližnjega vzhoda, pa so se pojavile specifične potrebe po podpori arabskih jezikov v menijih mPOS plačilnih terminalov. Članek opisuje enega izmed možnih načinov pretvarjanja arabskih znakov iz kodnega sistema 0600-06FF v prezentacijske forme arabskih znakov, ki se nahajajo v kodnem sistemu FE70-FEFF. V prvem delu smo se omejili predvsem na problematiko prikazovanja arabskih znakov na zaslonu terminala, ki se nahajajo v kodnem zapisu 0600-06FF. Iz tega razloga smo predstavili rešitev, ki se navezuje na pretvorbo arabskih znakov kodnega sistema 0600-06FF v kodni sistem arabskih prezentacijskih form FE70-FEFF. Na osnovi tega smo v nadaljevanju na praktičnih primerih predstavili rešitev pretvorbe v dveh korakih. Najprej se pretvorba izvrši preko translacijske matrike, v drugem koraku pa se pretvorjen zapis preko translacijske matrike primerja z ekvivalentnim originalnim arabskim zapisom. Na osnovi primerjave in ugotovljenih odstopanj se zasnujejo pravila po katerih se izbirajo oblike črk, glede na predhodno črko v besedi. Vsak predstavljen korak smo obrazložili na številnih praktičnih primerih. V članku predstavljamo tudi koncept izbire in aktivacije pravila, ki je povsem identičen ekspertnim sistemom, ki temeljijo na osnovi produkcijskih pravil. V drugem delu smo se osredotočili na jasno predstavitev posameznih rešitev, ki jih uporabljamo za konfiguracijo mPOS plačilnega terminala, kamor smo umestili tudi razvit produkt in njegove funkcionalnosti. V zadnjem delu smo se omejili predvsem na prikaz pretvorjenih besed v kodni zapis FE70-FEFF na zaslonu mPOS terminala.

Arabic characters automatic translation from 0600-06FF code system to Arabic presentation forms code system FE70-FEFF

Key words: mPOS, code systems, Arabic alphabet, Arabic signs, conversions, conversions matrix, rules, payment terminal, screen, hexadecimal record, inverse conversion, terminal tool, conversion tool, single rule, multiple rule.

Abstract: Prevalence of non-cash payment with mobile phones via mPOS payment terminal is becoming increasingly important and convenient. The developed solutions are ready to launch on the domestic and foreign markets. With the launch of a different solutions to the Middle Eastern markets, have emerged to support the specific needs upon the Arab language in the menus of mPOS payment terminals. This paper describes one possible way to convert the Arab characters from code page 0600-06FF into the Arabic presentation forms which are specified in FE70-FEFF code page. In the first part of the article, we are limited primarily to the problems which concerns correct displaying Arabic words (written in 0600-06FF code system) on the terminal screen. For this reason, we presented a solution which is tightly connected with the transformation of the Arab characters from 0600-06FF code system into the FE70-FEFF Arabic presentation form code page. On this basis, we present a solution on practical examples which is constructed with two steps. First step presets conversion over conversion matrix, meanwhile in second step we compare converted word (FE70-FEFF) with the original one (0600-06FF). Based on a comparison of the anomalies the rules are created and written into conversion setup file. The rules are created on the following way: for temporary char (FE70-FEFF), system observe char which is before it and then temporary char compares with original (0600-06FF) one shape. If there are differences in shape, then system produces rule which contains proper shape of temporary char correspond to char before it. All described is explain on practical examples. During this paper we present the concept of rules creation, selection and activation, which is completely identical to the expert system based on production rules. In the second part, we focused on a clear presentation of individual solutions that are used to configure mPOS payment terminal, where we place our well-developed product and its functionality. In the last part, we have limited primarily to see how the converted words (FE70-FEFF) looks like on the screen of the mPOS terminal.

1. Uvod

Brezgotovinsko plačevanje z mobilnim telefonom je zaradi razširjenosti mobilne telefonije postalo tržno zanimivo. Razvile so se različne tehnologije, ki omogočajo mobilno plačevanje. MPay je eden od sistemov mobilnega plačevanja, ki ga je razvilo podjetje Ultra d.o.o.. Sistem temelji na osnovi prenosa podatkov po govornem kanalu različnih mobilnih omrežij, kot so GSM, CDMA in UMTS. Glede na različne plačilne storitve, ki jih omogoča MPay sistem, se na trgu uporabljata dva tipa terminalov, in sicer aPOS ter

mPOS. Osnova obeh terminalov je enaka, razlika je v tem, da ima mPOS terminal vgrajen uporabniški vmesnik (tipkovnica, zaslon, tiskalnik). aPOS terminal se najpogosteje uporablja v restavracijah, trgovinah, taksiji, ..., in ne vsebuje grafičnega uporabniškega vmesnika, zato za našo razvito aplikacijo ni primeren. aPOS terminal ima vgrajene različne komunikacijske vmesnike, ki omogočajo priklop na razne prodajne avtomate, parkirne ure, igralne avtomate, ... Na novo razvita aplikacija se seveda nanaša na mPOS plačilne terminale, ki razpolagajo z uporabniškim vmesnikom s sposobnostjo prikazovanja različnih jezikov.

Ideja za strokovni prispevek je nastala na podlagi vse večje večje potrebe podjetja Margento R&D d.o.o., ki razvija sisteme brezgotovinskega plačevanja z mobilnimi telefoni preko mPOS terminalov za celotno svetovno tržišče. V večini primerov se na mPOS terminalih podpre večjezična podpora. S pridobitvijo razvojnih projektov za področje Združenih Arabskih Emiratov se je v podjetju pojavila potreba implementacije arabskega jezika na plačilne terminale. Problematika s katero smo se srečali se nanaša na sposobnost prikazovanja arabskih znakov na plačilnem terminalu T4000.



Slika 1: Prikaz aPOS in mPOS terminalov

Postopek priprave izpisov, ki se prikazujejo na zaslonu plačilnega terminala se prične z osnovno excel datoteko. Npr. V koloni 'A' se nahajajo npr. angleški izrazi, v koloni 'G' pa arabski izrazi, ki jih preko arabske tipkovnice zapiše prevajalec. Privzeto se arabski znaki, ki se vnašajo preko arabske tipkovnice nahajajo v kodnem zapisu 0600-06FF /1/. Z združevanjem znakov iz omenjenega kodnega sistema, črke spreminjajo obliko, glede na lokacijo v besedi in glede na tip predhodne in zaporedne črke. To pomeni, da črka spremeni obliko, vendar v hexadecimalnem zapisu, le ta še vedno ohranja enako 'hexa' vrednost, kljub temu, da se je črki spremenila oblika. To je tudi glavni problem na katerega naletimo pri implementaciji arabskega jezika na plačilni terminal (mPOS). Le ta ne pozna pretvorbe črk, glede na predhodno in naslednjo črko, zato jo prikaže nepovezano v obliki samostojne črke (beseda kot končni produkt ne izgleda povezano, temveč vsaka črka stoji sama zase). Tekom študije smo prišli do zaključka, da je za pravilno prikazovanje arabskega jezika na plačilnem terminalu, potrebna pretvorba znakov arabske abecede iz kodnega sistema 0600-06FF v prezentacijsko obliko arabskega kodnega sistema FE70-FEFF /5/. Le ta vsebuje kodne znake, ki upoštevajo lokacijo črke v besedi. To pomeni, da za vsak znak arabske abecede v kodnem sistemu 0600-06FF obstajajo štiri različice zapisa istega znaka v prezentacijskem /2/ arabskem kodnem sistemu FE70-FEFF. Takšne štiri različice se navezujejo na lokacijo arabskega znaka v besedi, in sicer: začetna črka besede, vmesna črka besede, končna črka besede in osamljena (izolirana/samostojna) črka abecede (vsaka ima svojo obliko).

Ker arabsko tržišče predstavlja enega izmed najpomembnejših področij uvajanja sodobnih načinov brezgotovinskega plačevanja, je bilo za nas izrednega pomena razviti avtomatizirano rešitev za pretvorbo znakov arabske abecede v prezentacijsko formo, saj podobne programske rešitve, ki bi bile sposobne izvesti opisano pretvorbo, ne obstajajo.

Pri razvoju tako imenovanega 'translacijskega' orodja smo šli še korak dlje v smeri avtomatične priprave t.i. 'printout' datotek, preko katerih v interni pomnilnik plačilnega terminala naložimo izraze različnih jezikov (npr. angleški in arabski jezik). Povzetek pregleda operabilnosti in funkcionalnosti sistema bomo na kratko predstavili v nadaljevanju.

V drugem poglavju predstavljamo definicijo problema in osnovno problematiko pretvorbe arabskih znakov iz enega kodnega sistema v drug kodni sistem. V okviru tretjega poglavja bomo predstavili metodologijo pretvorbe in t.i. 'translacijsko' matriko z upoštevanjem pozicij posameznih črk v besedi. Četrto poglavje je namenjeno predstavitvi drugega koraka, ki upošteva pravila predhodne in zaporedne črke glede na trenutno črko, in s tem pripomore k pravilni pretvorbi arabskih izrazov v kodni sistem FE70-FEFF. V petem poglavju predstavljamo pomembne korake od pretvorbe arabskih znakov do postopka naložitve le teh v flash pomnilnik mPOS terminala. Šesto poglavje je namenjeno predstavitvi funkcionalnosti razvite rešitve in njenih režimov delovanja. V sedmem poglavju so prikazani rezultati razvitega orodja, medtem ko z osmim sklepnim poglavjem zaključujemo članek.

2. Definicija problema in osnovna problematika

Kot smo že v uvodu nakazali, se osnovna problematika navezuje na razširitev trženja razvitih tehnologij na tuja tržišča, predvsem pa na tržišče Združenih Arabskih Emiratov. S prodorom na tuje trge se pojavlja potreba po več jezični podpori. Zaradi specifičnosti arabskega jezika, in omejitev programske opreme mPOS terminala smo morali razviti univerzalno rešitev, ki nam bo v veliko pomoč pri lansiranju nadaljnjih produktov na omenjeno tržišče.

Omejitev programske opreme plačilnega terminala mPOS se navezuje na nesposobnost prikazovanja znakov arabske abecede, ki se nahajajo v izvornem kodnem sistemu 0600-06FF (slika 2).

0600		Arabic																06FF	
	060	061	062	063	064	065	066	067	068	069	06A	06B	06C	06D	06E	06F			
0	0600	0601	0602	0603	0604	0605	0606	0607	0608	0609	060A	060B	060C	060D	060E	060F			
1	0610	0611	0612	0613	0614	0615	0616	0617	0618	0619	061A	061B	061C	061D	061E	061F			
2	0620	0621	0622	0623	0624	0625	0626	0627	0628	0629	062A	062B	062C	062D	062E	062F			
3	0630	0631	0632	0633	0634	0635	0636	0637	0638	0639	063A	063B	063C	063D	063E	063F			
4	0640	0641	0642	0643	0644	0645	0646	0647	0648	0649	064A	064B	064C	064D	064E	064F			
5	0650	0651	0652	0653	0654	0655	0656	0657	0658	0659	065A	065B	065C	065D	065E	065F			

Slika 2: Izsek kodne tabele 0600-06FF z arabskimi znaki /7/

Uvodoma smo omenili, da osnovni element med nami in prevajalcem v arabski jezik predstavlja excel datoteka. Vsak preveden izraz v excel datoteki je zapisan v 0600-06FF kodnem zapisu (gledano hexadecimalno). Lastnost takšnega zapisa je avtomatsko prilagajanje in spreminjanje oblike črk /4/, glede na predhodno in zaporedno zapisano črko. Naj spomnimo, da je lastnost arabske pisave tudi zapis od desne proti levi, za razliko od našega jezika, v katerem pišemo od leve proti desni. Da bo opisan princip čimbolj nazoren si pogledjmo naslednji primer:

Tabela 1: Spreminjanje oblike arabskih črk pri proceduri združevanja

Samostojna črka (0600-06FF)	Samostojna črka (0600-06FF)	Združeni črki (0600-06FF)
ف	ي	في

Na zgornjem primeru je lepo razviden status in oblika samostojnih (skrajno levi in arabski znak na sredini – glej tabelo 1) arabskih črk iz kodnega sistema (0600-06FF), ter spremenjena oblika znakov v primeru zapisa obeh, enega za drugim. S pregledom hexadecimalnega zapisa /6/ združenih znakov v besedo, pa le ta še vedno ohranjata izvirni hexadecimalni zapis, kljub spremenjeni obliki črk.

Tabela 2: Hexadecimalna vrednost posameznih ter združenih arabskih znakov

Samostojna črka (0600-06FF)	Samostojna črka (0600-06FF)	Združeni črki (0600-06FF)
ف	ي	في
Hexadecimalni zapis arabskih znakov		
0641	064A	0641 064A

Iz zgornjega primera je že lepo razvidna problematika, s katero smo se srečali pri prikazovanju arabskih izrazov na zaslonu terminala. Kljub združenim črkam in spremenjeni obliki le te še vedno ohranjajo kodni hexadecimalni zapis v takšni formi, kot da črka stoji samostojno in ni vezana na druge črke v zapisani besedi (kljub avtomatski spremembi oblike ob zapisu dveh zaporednih arabskih znakov). Iz tega razloga je tudi prikaz na zaslonu terminala napačen.

Rešitev za takšen problem najdemo v arabskih prezentacijskih znakih, ki so zapisani v kodnem zapisu FE70-FEFF. Takšen kodni zapis vsebuje za vsak znak iz 0600-06FF vse oblike tovrstnega znaka glede na pozicijo v besedi (samostojni znak, na začetku besede, na koncu v sredini). Da bo razlaga še bolj jasna, si oglejmo primer za arabski znak FEH (0641):

Marsikomu se na tem mestu porodi ideja, da je potrebno spremljati pozicije črk v besedi iz kodnega sistema 0600-06FF, in na osnovi teh pozicij nato izbiramo črke iz kodnega sistema FE70-FEFF. To je deloma pravilno, in tudi to metodologijo, ki jo bomo razložili v nadaljevanju uporabljaj-

Tabela 3: Primerjava hexadecimalnih zapisov oblike arabske črke FEH v obeh kodnih tabelah

Kodni sistem 0600-06FF				
Črka	Osamljena	Začetek	Sredina	Konec
FEH	ف			
0641	0641	0641	0641	0641
Kodni sistem FE70-FEFF				
FEH	ف	ف	ف	ف
0641	FED1	FED3	FED4	FED2

mo tudi mi. Vendar pa končna rešitev ni tako trivialna in enostavna, kot smo jo ravnokar navedli. Problematiko dodatno otežijo še dejstva, da je v FE70-FEFF kodnem sistemu potrebno upoštevati še dodatna pravila, katera črka je pred in za pretvorjeno.

Tipičen primer upoštevajoč samo pozicije znakov brez upoštevanja dodatnih pravil pretvorbe daje napačen rezultat (glej tabelo 4). Vendarle pa nam je bil takšen način v veliko pomoč pri postavljanju pravil za drugi način pretvorbe, ki upošteva tako pozicijo kot tudi pravila. Iz tega razloga je potrebno za pravilne pretvorbe zasnovati še dodatna pravila, ki pripomorejo k točnemu končnemu rezultatu. V sredinski koloni tabele 4 sta z rdečo barvo prikazani arabski črki zapisani v kodnem zapisu FE70-FEFF, ki sta bili vpeljeni z upoštevanjem pozicije brez dodatnih pravil, le ti pa sta napačni.

Tabela 4: Pretvorba v FE70-FEFF kodni sistem z upoštevanjem samo pozicije (sredinska kolona) ter z upoštevanjem pozicije in dodatnih pravil (desna kolona)

Originalen arabski izraz 0600-06FF	Pretvorba z upoštevanjem pozicije FE70-FE77	Upoštevana dodatna pravila Pretvorbe FE70-FE77
افخ	افخ	افخ
	Napačno	Pravilno

3. Potek pretvorbe z upoštevanjem pozicije

Logiko pretvarjanja izrazov iz 0600-06FF v FE70-FEFF smo zasnovali v programskem jeziku C# /3/.

Da bo predstava lažja, bomo potek tovrstne pretvorbe opisali na primeru arabske besede, ki jo sestavljajo arabski znaki iz 0600-06FF. Logika programa je v prvi fazi zasnovana tako, da besedo najprej razdeli na posamezne arabske znake, ki le to sestavljajo in jih zapiše v tako imenovane tipizirane liste, pri čemer ohranja zaporedje črk (od desne proti levi). V kolikor gre za stavek, le tega najprej razdruži na posamezne besede, nato pa vsako besedo v posamične črke, ki se ponovno zapišejo v pripravljene podatkovne strukture. Temelj pretvarjanja je torej razdruženje stavka na

besede in nadalje besede na posamične črke. Na tem mestu se marskomu pojavi vprašanje zakaj besedo razdeliti na posamezne črke? Odgovor je sila preprost. Za pretvorbo črke iz 0600-06FF v FE70-FEFF uporabljamo translacijsko matriko v naslednji obliki:

Tabela 5: Izsek iz sestavljene translacijske matrike

Original (hex)	Končna črka (hex)	Začetna črka (hex)	Vmesna črka (hex)	Izolirana črka (hex)
0641	FED2	FED3	FED4	FED1
0642	FED6	FED7	FED8	FED5
062B	FE9A	FE9B	FE9C	FE99
0635	FEBA	FEBB	FEBC	FEB9
0636	FEBE	FEBF	FEC0	FEBD
0643	FEDA	FEDB	FEDC	FED9
0645	FEE2	FEE3	FEE4	FEE1
0646	FEE6	FEE7	FEE8	FEE5
062A	FE96	FE97	FE98	FE95
...	...	...	...	...

Posamezno črko iz besede (0600-06FF), ki smo jo v predhodnem koraku razdelili pretvorimo v hexadecimalni zapis, in upoštevamo njeno pozicijo. Pozicije ovrednotimo s števili, kjer število 0 predstavlja začetno črko besede, število x pa končno črko besede. Vsa vmesna števila med 0 in x predstavljajo pozicije vmesnih črk. Po takšnem markiranju lahko pri nadaljnji obdelavi ugotovimo, lokacijo posamezne črke.

V naslednji fazi implementirana logika v translacijski matriki (Tabela 5) v koloni *Original (hex)* poišče istoimensko hexadecimalno vrednost, kot jo ima npr. začetna črka besede. Ko logika najde ujemanje, preveri indeks pozicije, na osnovi katerega s podpravili oblike *if in then* poišče ustrezni hexadecimalni zapis črke v isti vrstici translacijske matrike. Če se ozremo na tabelo 5 in predpostavimo, da imamo opravka z začetno črko besede, ki ima hexadecimalno vrednost 0641 bo logika v isti vrstici glede na poznan indeks črke izbrala hexadecimalno vrednost iz kolone *Začetna črka (hex)* FED3. S podobno logiko operiramo tudi v primeru stavkov itd. V primeru osamljene črke pa je logika nekoliko drugačna, saj preveri ali osamljena črka vsebuje še kakšno črko, ki se nahaja za njo. Če takšna črka ne obstaja jo markira kot izolirano, in po enakem principu poišče ekvivalent FE70-FEFF preko translacijske matrike. Procedura se po takšnem principu ponavlja, dokler niso pretvorjeni vsi arabski izrazi, besede, stavki...

Rezultat takšne pretvorbe je prikazan v tabeli 4 – sredinska kolona.

4. Princip zasnove pravil s pomočjo rezultatov predhodne metode

Na osnovi primerjave rezultatov predhodne metode z arabskimi izrazi, ki se nahajajo v excel datoteki (originalni izra-

zi), lahko zasujemo tabelo pravil za vsako posamezno črko, glede na njeno zaporedno in predhodno črko. Logika snovanja matrike pravil je sledeča. Z orodjem za primerjavo datotek, primerjamo ustvarjeno 'printout' datoteko po prvi metodi z originalno excel datoteko. Na besedah, ki se razlikujejo izvršimo analizo in hkrati snujemo matriko pravil za posamezno črko. Naprimer, primerjajmo spodnja izraza, kjer je levi pravilen, desni pa po predhodni metodi brez pravil napačno pretvorjen.

افخ افخ

Desni izraz se tako že nahaja v prezentacijski obliki arabskih znakov. Vidimo pa, da je zadnja črka (od desne proti levi) desnega izraza napačna. Iz tega razloga lahko za rdečo kombinacijo črk zasujemo naslednjo matriko pravila. Logika preverjanja deluje tako, da vzame hexadecimalno vrednost trenutne črke, in pregleda naslednjo črko. Če v pravilu trenutne črke najde še hexadecimalno vrednost naslednje črke, potem aktivira pravilo, ki vrne rezultat pravilne naslednje črke. Na enak princip deluje tudi iskanje in aktivacija drugih pravil, za druge kombinacije črk.

Poenostavljeno enostopenjsko pravilo za kombinacijo črk:

فخ

Trenutna črka	Naslednja črka	Rezultat pravila	Izraz po upoštevanju pravila
ف	خ	خ	افخ

V naslednjem koraku si oglejmo primer kompleksnejšega pravila z več internimi stanji. Prejšnje pravilo je tako imenovano enkratno pravilo, saj vsebuje samo eno obliko naslednje črke.

Razširjeno sedem-stopenjsko pravilo za kombinacijo črk:

\ in vseh iz družine غ in družine غ

Trenutna črka	Naslednja črka (p1)	Naslednja črka (p2)	Naslednja črka (p3)
FE8D (١)	FEC9 (ع)	FECA (ع)	FECB (ع)
	Naslednja črka (p4)	Naslednja črka (p5)	Naslednja črka (p6)
	FECC (ع)	FECD (ع)	FECE (ع)
	Naslednja črka (p7)	Rezultat pravila družine FEC9 (ع)	Rezultat pravila družine FECD (ع)
	FECF (ع)	FEC9 (ع)	FECD (ع)

Podobna pravila ustvarimo za vse ostale kombinacije in družine črk. Princip prepoznavne in aktivacije pravila je identičen tistemu, ki smo ga spoznali pri poenostavljenem enostopenjskem pravilu. Stopnja je odvisna od števila podpravil (p1), (p2), (p3), ..., (pn) za posamezno družino črke, ki se nahaja za trenutno. Družina podpravil npr. za črko FECA (ع) je: FEC9 (ع), FECB (ع) in FECC (ع). Ne glede na to, katera črka iz te družine se bo pojavila za črko FE8D (١), vedno bo pravilo vrnilo rezultat FEC9 (ع). Enako velja za

črke iz druge družine, ki so zajete v zgornjem sedem-stopenjskem pravilu.

Aplikacija, ki smo jo zasnovali v programskem jeziku C#, je zasnovana modularno in fleksibilno, saj omogoča dodajanje novih pravil po potrebi. Dodajanje pravil je izvedeno preko tako imenovane namestitvene (*ang. setup*) datoteke. Gledano dolgoročno se s tem izognemo nepotrebnim posegom v programsko kodo, ter hkrati uporabniku omogočimo ob vsakem trenutku kreacijo in dodajanje novih pravil. Namestitvena datoteka je običajna tekstovna datoteka z urejenimi tabelarnimi zapisi pravil. To pomeni, da se mora uporabnik pri vnosu novih pravil držati samo tabelarčne ureditve v datoteki. Ob zagonu okenske aplikacije za pretvorbo arabskih izrazov se pravila prenesejo iz namestitvene datoteke v interno dinamično strukturo programa, kjer jih le ta koristi po že opisanem postopku.

Iz tega vidika je razvita aplikacija na las podobna ekspertnemu sistemu, saj vsebuje večino ključnih gradnikov, kot so: uporabniški vmesnik, baza, pravila ipd. Da bi lahko aplikacijo uvrstili med ekspertne sisteme bi morali še vključiti mehanizme sklepanja, ki bi se navezovali na avtomatsko primerjavo originalnih arabskih izrazov iz excel datoteke z izrazi, ki jih dobimo pri pretvorbi, katera upošteva samo pozicijo. V naslednji fazi bomo sistem nadgradili na stopnjo ekspertnega sistema, ki bo v navezi z mehanizmi sklepanja in avtomatske primerjave opravil avtomatsko generiranje pravil, katere smo v našem primeru definirali sami (empirično). V smislu obravnavanja pravil po principu *if* in *then*, kombinaciji le teh, ter aktivaciji, pa je delovanje razvitega '*translacijskega*' sistema povsem identično delovanju ekspertnih sistemov. Korelacijo med obema sistemoma je moč potrditi tudi pri načinu učenja preko dodajanja pravil.

5. Shematski prikaz postopka od pretvorbe do naložitve pretvorjenih zapisov v flash pomnilnik plačilnega terminala mPOS

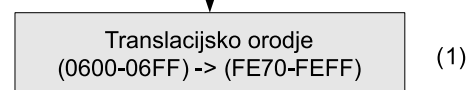
Proces pretvorbe in posamezni koraki le tega, so prikazani na spodnji sliki. Ker smo proces pretvorbe že spoznali, si podrobneje oglejmo strukturo izhodne '*printout*' datoteke, ter nadaljnje postopke, kako vsebino '*printout*' datoteke prenesti v flash pomnilnik mPOS terminala.

'Printout' datoteka: Slednja služi kot vmesni člen med *Translacijskim orodjem* (1) in *Terminal Tool* orodjem (2). Omenjena datoteka predstavlja končni produkt pretvorbe, v njej pa se nahajajo tako angleški, kakor tudi arabski izrazi, ki so že v prezentacijski formi (FE70-FEFF). Vsaka kombinacija izrazov (angleški – arabski) se nahaja v svoji vrstici, pri čemer sta oba izraza medsebojno ločena z znakom tilda (~). Konec enega para je označen z znakom (⌘). V naslednji vrstici je po enaki sintaksi zapisan drugi par itd. Označevanje in ločevanje zapisov z znakoma (~, ⌘) je pomembno zaradi kasnejšega poenostavljenega branja vsebine '*printout*' hex datoteke v *Terminalskem orodju*.

Izvorna excel datoteka z angleškimi izrazi v koloni A in ekvivalentnimi Arabskimi izrazi v koloni C

	A	B	C
1	Printer Test		فحص الطابعة
2	Transaction Info		معلومات المعاملة
3	OK		جيد
4	Battery		البطارية

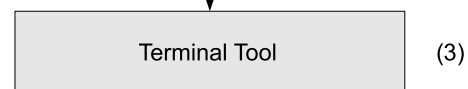
Vhod v translacijski sistem



Izhod iz translacijskega sistema

1 Printer Test~فحص الطابعة⌘
2 Transaction Info~معلومات المعاملة⌘
3 OK~جيد⌘
4 Battery~البطارية⌘ (2)

'Printout' datoteka je vhod v terminalsko orodje



mPOS Terminal



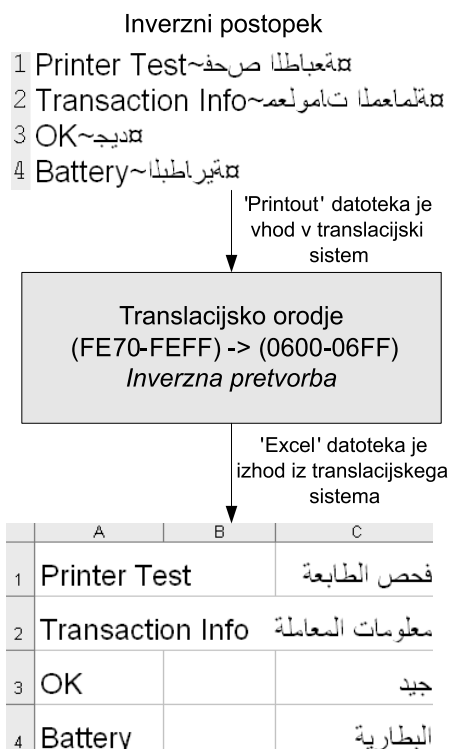
Slika 3: Procedura od pretvorbe do zapisa na flash pomnilnik mPOS terminala

Terminalsko orodje (Terminal Tool):

S pomočjo terminalskega orodja urejamo tako imenovane *M-Pay* konfiguracijske datoteke. Primarni namen orodja je nastavitve parametrov, ki se nato zapišejo v flash pomnilnik mPOS terminala. Dodajamo lahko tudi druge parametre, katerih vrednost lahko spremenimo preko uporabniškega vmesnika, brez posega na kodni nivo. Z enakim orodjem lahko naložimo tudi pripravljeno '*printout*' datoteko, ki predstavlja izhod in *translacijskega* orodja. Vse nastavljene in naložene vsebine v *M-Pay* konfiguracijski datoteki se nato preko Terminalskega orodja prenesejo in zapišejo na interni pomnilnik mPOS terminala. Terminalsko orodje se uporablja tudi za zapis konfiguracijskih parametrov na aPOS terminale (glej uvodno poglavje), vendar se v tem primeru

'*printout*' datoteka ne uporabi, saj aPOS terminal ne razpolaga z uporabniškim zaslonom, torej ne izpisuje in ne prikazuje nikakršnih vsebin. Če povzamemo funkcionalnosti terminalskega orodja; omogoča spreminjanje 'firmware-a', terminalskih nastavitev, logotipov, izpisov, fontov, zvočnih zapisov itd.

Inverzna pretvorba: Proces pretvorbe, ki smo ga spoznali v predhodnih poglavjih smo razširili tudi na pretvorbo v obratni (inverzni) smeri.



Slika 4: Inverzna pretvorba

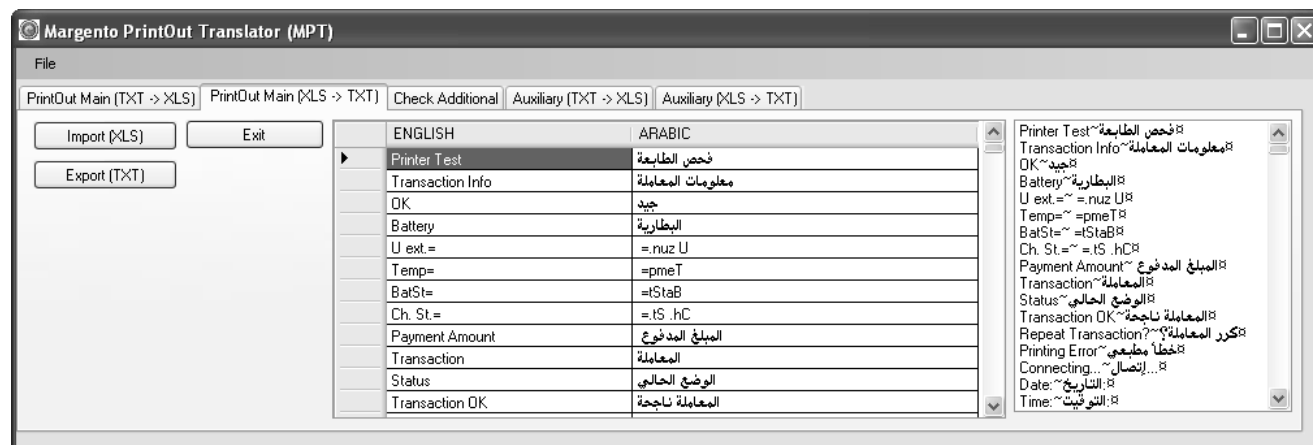
Inverzno pretvorbo smo uvedli iz dveh razlogov: v kolikor potrebujemo kakršne koli spremembe arabskih izrazov s strani prevajalca, je najenostavnejša rešitev izvoz v novo excel datoteko, ki jo lahko prevajalec enostavno spremeni.

Na osnovi spremenjene datoteke lahko s translacijskim orodjem ponovno ustvarimo novo '*printout*' datoteko. Inverzna pretvorba, je enostavnejša v primerjavi z normalno pretvorbo, zaradi samo ene oblike arabskih črk v kodnem zapisu 0600-06FF. Naj spomnimo, da pri inverznem poteku pretvarjamo arabske črke iz kodnega sistema FE70-FE7F v kodni sistem 0600-06FF. Pretvorba poteka na sledeč način. Vsako arabsko besedo v prezentacijski formi (FE70-FE7F) razdelimo na posamezne črke, le te pa pretvorimo v *hexadecimalni* zapis. V naslednjem koraku za vsako črko besede s pomočjo preproste programske logike preiščemo translacijsko matriko. Ko algoritem najde ekvivalentno hexadecimalno vrednost črke besede kodnega sistema FE70-FE7F v translacijski matriki (Tabela 5), poišče istoležno originalno hexadecimalno vrednost (zapisano v hexadecimalni obliki sistema 0600-06FF), ki jo nato pretvori nazaj v črko iz kodnega sistema 0600-06FF. Črke posamičnih besed algoritem sestavi, in zapiše na ustrezno mesto ustrezne kolone excel datoteke. Enako storimo za vse ostale črke besed.

Drugi razlog, zakaj inverzna pretvorba, pa se nanaša na morebitno izgubo oziroma poškodovanje izvorne .xls datoteke s prevodi. Ker se le ti nahajajo v flashu terminala, jih lahko najprej s pomočjo *terminalskega orodja* izvozimo v '*printout*'.hex datoteko, nato pa s pomočjo *translacijskega orodja* v izvorno .xls datoteko.

6. Uporabniški vmesnik in funkcionalnosti

Uporabniški vmesnik temelji na dveh temeljnih korakih, ki hkrati poenostavljata uporabo razvite rešitve. Uporabniški korak tvori ukaz *naloži*, s katerim uporabnik naloži izvorno datoteko (.xls, .hex) in ukaz *pretvori* s katerim sistem pretvori arabske zapise iz enega kodnega sistema v drugega. Z minimalnim številom korakov smo izpolnili pogoj uporabniške prijaznosti, z upoštevanjem minimalističnega oblikovanja pa poskrbeli za pregleden uporabniški vmesnik. Prebrane in pretvorjene izraze ispisujemo v dve ločeni tekstovni listi, predvsem iz razloga preverjanja ustreznosti



Slika 5: Uporabniški vmesnik in osnovne funkcije razvite rešitve

0,Unauthorized Terminal. Please Call Mobily Support.	0, الجهاز غير مرخص الرجاء الاتصال بخدمة العملاء
0,Transaction Denied. Please Call Mobily Support.	0, المعاملة مرفوضة الرجاء الاتصال بخدمة العملاء
0,Transaction Type Denied. Please Call Mobily Support.	0, نوع المعاملة مرفوض الرجاء الاتصال بخدمة العملاء
0,Transaction Delivery Denied. Please Call Mobily Support.	0, تسليم المعاملة مرفوض الرجاء الاتصال بخدمة العملاء
1,Transaction Delivery Failed. Please Repeat.	1, فشل في تسليم المعاملة الرجاء اعد المحاولة
0,Transaction Authorization Denied. Please Call Mobily Support.	0, ترخيص المعاملة مرفوض. الرجاء الاتصال بخدمة العملاء
1,Transaction Authorization Failed. Please Repeat.	1, ترخيص المعاملة مرفوض. الرجاء الاتصال بخدمة العملاء
1,M-Pay System Is Currently Unavailable. Please Repeat.	1, نظام M-Pay غير متوفر حاليا. الرجاء اعد المحاولة

Slika 6: Originalen zapis (0600-06FF) v Auxiliary.xls datoteki

pretvorbe. Funkcij in gumbob, ki jih uporabnik v določenem trenutku ne potrebuje ne prikazujemo. V primeru pretvorbe .xls datoteke v .hex datoteko se izvrši normalna pretvorba, torej iz kodnega sistema 0600-06FF v kodni zapis arabskih prezentacijskih fontov FE70-FEFF. V primeru inverzne pretvorbe pa sistem pretvori .hex 'printout' datoteko in njeno arabsko vsebino iz kodnega sistema FE70-FEFF v kodni sistem 0600-06FF, in arabsko vsebino zapiše v novo .xls datoteko. Na ta način zagotovimo podporo pretvorbe v obe smeri. Zraven pretvorbe t.i. 'printout' datoteke smo vključili še funkcionalnost pretvorbe t.i. 'auxiliary' datoteke, kjer so zapisani razni dodatni napotki in errorji, ki jih terminal prikazuje ob različnih dogodkih. Struktura zписа v 'auxiliary' .hex in .xls datoteki je povsem identična strukturi zписа v 'printout' .hex in .xls datoteki. Iz tega razloga je enaka logika uporabljena tako za normalno kot inverzno pretvorbo auxiliary datoteke.

V razvito programsko rešitev smo dodatno implementirali še rešitev dopolnjevanja dodatne (ang. additional) .hex datoteke. Additional.hex datoteka privzeto vsebuje vse alfanumerične znake (evropsko tržišče). V primeru implementacije drugih jezikov, kot je npr. Arabski, pa implementirana logika za vsak znak vsake besede preveri, ali ga že ima zapisanega v t.i. additional.hex datoteki. Če le tega ne vsebuje, ga avtomatično doda. Postopek ponavlja vse do zadnje črke zadnje besede, ki je zapisana v .xls datoteki. Na ta način napolnimo additional datoteko z vsemi znaki, ki so zastopani v .xls datoteki, vendar en znak lahko v additional datoteki nastopa samo enkrat. Pri proceduri zписа v flash pomnilnik mPOS terminala, terminalsko orodje preverja vsako črko besede z črkami, ki so enkratno zastopane v additional datoteki. Če slednjega v datoteki ne bi našel, ga tudi terminal na zaslonu ne bi prikazal (znak bi enostavno izpustil). Iz tega razloga je še toliko pomembnejše, da avtomatiziran postopek napolni vsebino additional datoteke z novimi znaki, ki jih do tedaj še ne vsebuje. Da bi bila uporaba razvite rešitve še toliko lažja, smo vsakemu gumbu, funkciji, listi ipd. dodali namige (ang. tool tips), s katerimi ob enem uporabnika usmerja, in seznanja čemu je določen gumb, meni ipd. namenjen. Pri načrtovanju uporabniškega vmesnika smo prav tako upoštevali Millerjev zakon kratkotrajnega spomina, ter priporočene smer-nice.

7. Rezultati

Primer pretvorbe izrazov iz Auxiliary.xls datoteke v prezentacijsko obliko ter zapis v Auxiliary.hex datoteko:

Na sliki 6, je prikazan izsek zписа Auxiliary.xls datoteke. Na skrajni desni se nahajajo izrazi zapisani v arabskem jeziku kodnega sistema 0600-06FF. Na sliki 7 je prikazan izsek izpisov (kodni sistem FE70-FEFF) iz ustvarjene Auxiliary.hex datoteke, ki predstavlja enega izmed izhodov translacijskega orodja.

- 0,Unauthorized Terminal.
- Please Call Mobily Support.~0, الجهاز غير مرخص
- الرجاء الاتصال بخدمة العملاء
- 0,Transaction Denied.
- Please Call Mobily Support.~0, المعاملة مرفوضة
- الرجاء الاتصال بخدمة العملاء
- 0,Transaction Type Denied.
- Please Call Mobily Support.~0, نوع المعاملة مرفوض
- الرجاء الاتصال بخدمة العملاء

Slika 7: Pretvorjen arabski zapis v kodni sistem FE70-FEFF

Potrditev, da se pretvorjen zapis nahaja v kodnem sistemu FE70-FEFF prikazuje slika 8.

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
00000000h:	FF	FE	30	00	2C	00	55	00	6E	00	61	00	75	00	74	00
000000010h:	68	00	6F	00	72	00	69	00	7A	00	65	00	64	00	20	00
000000020h:	54	00	65	00	72	00	6D	00	69	00	6E	00	61	00	6C	00
000000030h:	2E	00	0D	00	0A	00	50	00	6C	00	65	00	61	00	73	00
000000040h:	65	00	20	00	43	00	61	00	6C	00	6C	00	20	00	4D	00
000000050h:	6F	00	62	00	69	00	6C	00	79	00	20	00	53	00	75	00
000000060h:	70	00	70	00	6F	00	72	00	74	00	2E	00	7E	00	30	00
000000070h:	2C	00	BA	FE	A8	FE	AE	FE	E3	FE	20	00	AE	FE	F4	FE
000000080h:	CF	FE	20	00	B0	FE	8E	FE	EC	FE	AD	FE	EG	FE	8D	FE
000000090h:	0D	FE	0A	00	80	FE	FC	FE	E4	FE	CC	FE	EG	FE	8D	FE
0000000a0h:	20	00	94	FE	E4	FE	AA	FE	A8	FE	91	FE	20	00	DE	FE
0000000b0h:	8E	FE	BC	FE	98	FE	FB	FE	8D	FE	20	00	80	FE	8E	FE
0000000c0h:	AD	FE	AE	FE	E0	FE	8D	FE	A4	00	0D	00	0A	00	30	00

Slika 8: Hexadecimalni zapis vsebine v kodnem sistemu FE70-FEFF

Pretvorjeni izpisi v kodni sistem FE70-FEFF, ki jih nato zapišemo v flash pomnilnik mPOS terminala in prikažemo v menijih na zaslonu terminala so desno poravnani. Branje tovrstnih arabskih izpisov v menijih prav tako poteka od desne proti levi strani (zakonitosti pretvorbe iz enega v drug kodni sistem in obratno se ohranjajo).

Primer arabskih izpisov na zaslonu mPOS terminala je prikazan na sliki 9.



Slika 9: Izpis arabskih znakov kodnega sistema FE70-FEFF na zaslonu mPOS terminala

8. Sklep

Tekom članka smo predstavili princip in implementirane rešitve, ki pripomorejo k pravilni pretvorbi arabskih znakov iz kodnega sistema 0600-06FF v kodni sistem FE70-FEFF. Pristop z uporabo translacijskih matrik in aktivacijskih pravil se je v našem primeru izkazal za pravilnega. Ker so translacijske matrike zapisane v tako imenovani namestitveni datoteki, lahko uporabnik poljubno dodaja nove kombinaci-

je. Enak pristop velja za pravila. To pomeni, da je sistem možno dograjevati in izpopolnjevati s strani drugih uporabnikov, saj je le ta javno dostopen vsakomur. Uspešna pretvorba je bila prav tako potrjena s strani naročnika iz Združenih Arabskih Emiratov. Rezultati, ki smo jih prejeli zadovoljujejo tako naše, kakor tudi naročnikove potrebe.

Literatura

- /1/ <http://unicode.org/charts/PDF/U0600.pdf>
- /2/ Zina Saadi, Arabic, Farsi and Urdu Text Normalization for Natural Language Processing, <http://www.basistech.com/knowledge-center/Arabic/arabic-text-normalization.pdf>
- /3/ Kari Laitinen, A Natural Introduction to Computer Programming with C#
- /4/ Sarmad Hussain, Nadir Durrani, Sana Gul, Survey of Language Computing in Asia 2005, <http://www.pan10n.net/english/outputs/Survey/Arabic.pdf>
- /5/ <http://www.unicode.org/charts/PDF/Unicode-3.2/U32-FE70.pdf>
- /6/ <http://www.uconv.com/translit.htm>
- /7/ <http://en.wikipedia.org/wiki/Glyph>

Saša Klampfer, Peter Šamperl,
Zdenko Mezgec, Amor Chowdhury

Margento R&D d.o.o.
Gospodsvetska cesta 84, Maribor, 2000, Slovenija
Epošta: sasa.klampfer@margento.com

Prispelo (Arrived): 21.09.2009 Sprejeto (Accepted): 09.03.2010