

TEHNIČKA SPECIFIKACIJA ZAŠTIĆENOGA
PROGRAMA ZA OVLAŠTENE MJENJAČE

1. MODEL PODATAKA

1.1. PROPISANI MODEL PODATAKA

Ovom Tehničkom specifikacijom propisan je samo minimalni skup podataka koje mora sadržavati svaki zaštićeni program za ovlaštene mjenjače. Dopušteno je ovaj model proširivati prema vlastitim potrebama radi drugih željenih podataka o radu mjenjačkog mjesta kao i radi optimizacije izrade drugih izvještaja, koji nisu propisani Odlukom o uvjetima i načinu na koji ovlaštene mjenjači obavljaju mjenjačke poslove (u nastavku teksta: Odluka).

Nije dopušteno reducirati ovaj model ni mijenjati nazive pojedinih stupaca, već ih treba nazvati točno onako kako je napisano u prikazu i opisu modela. Programska rješenja koja to ne budu poštivala, neće biti certificirana.

1.2. PRIKAZ PROPISANOG MODELA PODATAKA

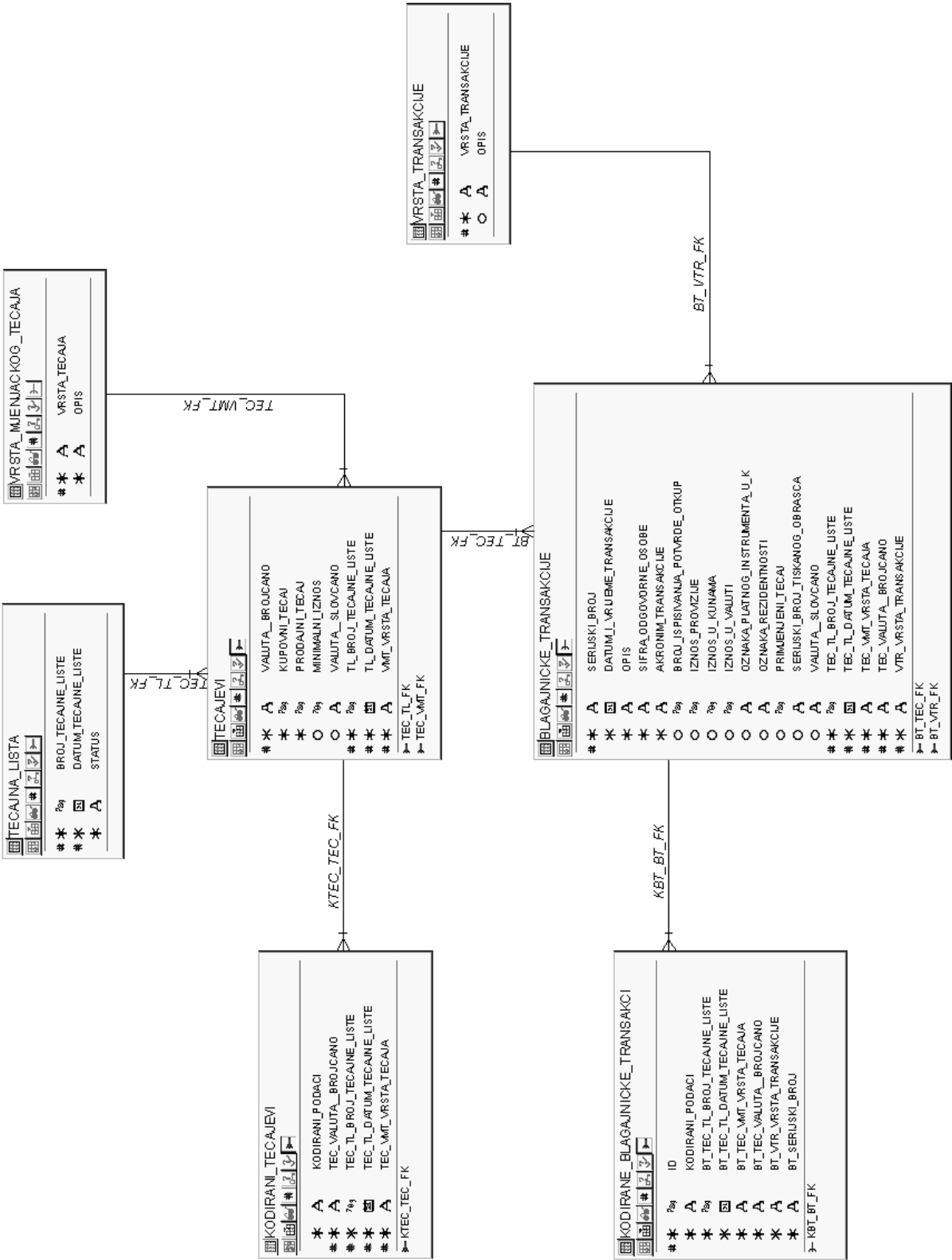
1.2.1. Simboli

U prikazu propisanog modela podataka upotrijebljeni su sljedeći simboli:

- # – označuje stupac primarnoga ključa
- * – označuje podatke koji se obvezno unose
- ° – označuje podatke koji se mogu, ali ne moraju unositi
- – veza između pojedinih tablica u modelu podataka; strana ‘–’ označuje stranu nadređene tablice, a strana ‘➤’ označuje stranu podređene tablice. Primarni ključ iz nadređene tablice čini ili dio primarnoga ključa podređene tablice ili vanjski ključ na podređenoj tablici.

Uz svaku tablicu u prikazu modela, osim popisa stupaca koji čine pojedinu tablicu, nalaze se i vanjski ključevi koji pripadaju toj tablici. Isto tako, može se vidjeti kategorija podataka (primarni ključ, obvezni podatak, neobvezni podatak) i tip podataka (numerički, znakovni, datumski).

1.2.2. Prikaz modela



1.3. OPIS MODELA PODATAKA – TABLICA

1.3.1. Tečajna lista

Pohranjuje se broj važeće tečajne liste i datum za koji vrijedi te oznaka stanja tečajne liste (važeća, stornirana). Na početku svake kalendarske godine broj tečajne liste počinje od broja jedan (1).

Stupci u tablici:

BROJ_TECAJNE_LISTE;	primarni ključ, numerički stupac
DATUM_TECAJNE_LISTE;	primarni ključ, datumski stupac
STATUS;	znakovni stupac

1.3.2. Vrsta mjenjačkog tečaja

Pohranjuje se vrsta mjenjačkog tečaja. Trenutačno su dopuštene dvije vrste tečaja: temeljni i povoljni. U ovoj tablici mogu se i fiksno odrediti te dvije vrijednosti (npr. upotrebom padajućih lista s fiksnim vrijednostima), ali u tom slučaju svaka naknadna promjena tih vrijednosti podliježe postupku ponovnog certificiranja.

Stupci u tablici:

VRSTA_TECAJA;	primarni ključ, znakovni stupac
OPIS;	znakovni stupac

1.3.3. Tečajevi

U ovoj tablici pohranjuju se valutni tečajevi koji će zatim biti primijenjeni u transakcijama tijekom dana za koji vrijedi tečajna lista. Primarni ključ ove tablice formira se iz primarnoga ključa tablice *Tecajna lista* i stupca *Valuta* u ovoj tablici. Valuta može biti unesena brojčanom ili slovnom oznakom prema propisanim standardima (brojčana ili slovna oznaka od tri znaka). Moguć je i unos obiju oznaka, tako da budu međusobno spojene (šest znakova) ili razdvojene jednim znakom razmaka (sedam znakova).

Stupci u tablici:

VALUTA_BROJCANO;	primarni ključ, znakovni stupac
KUPOVNI_TECAJ;	numerički stupac
PRODAJNI_TECAJ;	numerički stupac
MINIMALNI_IZNOS;	numerički stupac
VALUTA_SLOVCANO;	znakovni stupac
TL_BROJ_TECAJNE_LISTE;	primarni ključ, numerički stupac
TL_DATUM_TECAJNE_LISTE;	primarni ključ, datumski stupac
VMT_VRSTA_TECAJA;	primarni ključ, znakovni stupac

1.3.4. Kodirani tečajevi

Pohranjuju se kodirani podaci, s pomoću dane procedure (opisane u točkama 3.1. i 3.2.), a kako bi poslije bili na raspolaganju nadzornim tijelima. Primarni ključ ove tablice identičan je primarnom ključu tablice *Tecajevi*.

Stupci u tablici:

KODIRANI_PODACI;	znakovni stupac
TEC_VALUTA_BROJCANO;	primarni ključ, znakovni stupac
TEC_TL_BROJ_TECAJNE_LISTE;	primarni ključ, numerički stupac
TEC_TL_DATUM_TECAJNE_LISTE;	primarni ključ, datumski stupac
TEC_VMT_VRSTA_TECAJA;	primarni ključ, znakovni stupac

1.3.5. Vrsta transakcije

Pohranjuje se vrsta blagajničkih transakcija. Kao i kod tablice *Vrsta mjenjačkog tečaja*, u ovoj tablici mogu se i fiksno odrediti vrijednosti za vrstu transakcije, ali u tom slučaju svaka naknadna promjena tih vrijednosti podliježe postupku ponovnog certificiranja.

Stupci u tablici:

VRSTA_TRANSAKCIJE;	primarni ključ, znakovni stupac
OPIS;	znakovni stupac

1.3.6. Blagajničke transakcije

U ovoj tablici pohranjuju se sve transakcije koje oblikuju blagajnički dnevnik. S obzirom na različite vrste transakcija, postoji veći broj stupaca čiji unos nije obavezan. Primarni ključ ove tablice oblikovan je iz primarnih ključeva tablica *Tecajevi* i *Vrsta transakcije* te stupca *Serijski broj transakcije*. Na početku svake kalendarske godine brojevi transakcija počinju iznova od broja jedan (1).

Budući da u skladu s člankom 4. stavcima 6. i 7. Odluke oznaka za rezidentnost stranke nije obavezan element potvrde o otkupu odnosno potvrde o prodaji, programskim rješenjem ne predviđa se unos oznake rezidentnosti stranke, te se vrijednost oznake rezidentnosti stranke postavlja na »R«.

Stupci u tablici:

SERIJSKI_BROJ;	primarni ključ, numerički stupac
DATUM_I_VRIJEME_TRANSAKCIJE;	datumski stupac
OPIS;	znakovni stupac
SIFRA_ODGOVORNE_OSOBE;	znakovni stupac
AKRONIM_TRANSAKCIJE;	znakovni stupac
BROJ_ISPISIVANJA_POTVRDE_OTKUP;	numerički stupac
IZNOS_PROVIZIJE;	numerički stupac
IZNOS_U_KUNAMA;	numerički stupac
IZNOS_U_VALUTI;	numerički stupac
OZNAKA_PLATNOG_INSTRUMENTA_U_K;	znakovni stupac
OZNAKA_REZIDENTNOSTI;	znakovni stupac
PRIMJENJENI_TECAJ;	numerički stupac
SERIJSKI_BROJ_TISKANOG_OBRASCA;	znakovni stupac
VALUTA_SLOVCANO;	znakovni stupac
TEC_VALUTA_BROJCANO;	primarni ključ, znakovni stupac
TEC_TL_BROJ_TECAJNE_LISTE;	primarni ključ, numerički stupac
TEC_TL_DATUM_TECAJNE_LISTE;	primarni ključ, datumski stupac
TEC_VMT_VRSTA_TECAJA;	primarni ključ, znakovni stupac
VTR_VRSTA_TRANSAKCIJE;	primarni ključ, znakovni stupac

1.3.7. Kodirane blagajničke transakcije

Pohranjuju se kodirani podaci blagajničkih transakcija, s pomoću dane procedure (opisane u točkama 3.1. i 3.2.), a kako bi poslije bili na raspolaganju nadzornim tijelima. Primarni je ključ ove tablice stupac *ID* koji čini generički primarni ključ (sekvencijalni brojevi, koji se povećavaju u koracima za jedan i upisuju u taj kolonu), a koji generira zaštićeni program. Vanjski je ključ ove tablice primarni

ključ tablice *Blagajnicke transakcije*. Za svaki stupac iz tablice *Blagajnicke transakcije* dopušteno je više kodiranih redaka (načelno jedan ili dva retka) jer u slučaju storniranja već provedene transakcije treba sačuvati u kodiranom obliku i originalnu transakciju i stornirane podatke o transakciji.

Stupci u tablici:

ID;	primarni ključ, numerički stupac
KODIRANI_PODACI;	znakovni stupac
BT_SERIJSKI_BROJ;	numerički stupac
BT_TEC_VALUTA_BROJCANO;	znakovni stupac
BT_TEC_TL_BROJ_TECAJNE_LISTE;	numerički stupac
BT_TEC_TL_DATUM_TECAJNE_LISTE;	datumski stupac
BT_TEC_VMT_VRSTA_TECAJA;	znakovni stupac
BT_VTR_VRSTA_TRANSAKCIJE;	znakovni stupac

1.3.8. Nazivi stupaca u tablicama

S obzirom na to da je naknadnim provjerama ustanovljeno da pojedine baze podataka ne dopuštaju unos propisanih naziva stupaca u cijelosti, već su nazivi stupaca ograničeni na manji broj znakova, dopušteno ih je promijeniti. Promjena se izvodi tako da se odbaci višak znakova (više od najvećeg mogućeg broja znakova za naziv stupca koji dopušta odabrana baza). Svi ostali stupci, čiji su nazivi kraći od najduljega mogućeg za tu bazu, moraju zadržati propisani naziv i nije dopušteno njihovo skraćivanje.

Primjer:

Baza podržava naziv od najviše 25 znakova. Tada naziv:

123456789012345678901234567890

SERIJSKI_BROJ_TISKANOG_OBRASCA

treba napisati kao:

1234567890123456789012345

SERIJSKI_BROJ_TISKANOG_OB

Ako baza na kojoj je zasnovano programsko rješenje traži ovakve izmjene u propisanom nazivlju, pri predaji programskog rješenja za certificiranje potrebno je navesti koliko znakova najviše u nazivu podržava baza na kojoj je zasnovano programsko rješenje. Isto tako potrebno je napisati nazive stupaca koji su morali biti skraćeni te u kojim se tablicama ti stupci nalaze.

2. PLATFORME ZA IZRADU ZAŠTIĆENIH PROGRAMA

Zaštićeni program treba se izvoditi na osobnom računalu (IBM-PC kompatibilno računalo). Rješenja koja se ne budu zasnivala na točkama od 2.1. do 2.3., neće se razmatrati u postupku certificiranja.

2.1. OPERATIVNI SUSTAVI

Operativni sustavi koji se mogu primjenjivati za izradu i upotrebu zaštićenog programa:

- Microsoft Windows
- Linux.

2.2. PROGRAMSKI JEZICI

Programski jezici (alati) koji se mogu primjenjivati za izradu zaštićenog programa:

- MS Visual Basic
- MS Visual C/C++

- MS Access
- Borland C/C++ Builder
- Borland Delphi
- C/C++
- Pascal
- MS Visual Fox Pro
- JAVA.

2.3. BAZE PODATAKA

Zaštićeni program može se za svoj rad služiti svim dostupnim bazama podataka, ali je nužno priložiti bazu podataka na kojoj se zasniva programsko rješenje. Prema željama autora programskih rješenja dopušteno je napisati i vlastitu bazu podataka, ali tada se treba opisati njezina struktura i priložiti tako kreirana baza.

3. PROGRAMSKI ZAHTJEVI

Zaštićeni program treba kodirati skup podataka definiran ovom Uputom, a za potrebe nadzornih tijela. **Ako se postojeći i kodirani podaci pokušaju promijeniti, bit će izmijenjeni tako da će ih moći pročitati (dekodirati) samo nadzorno tijelo.**

Slijedi opis funkcija kodiranja i dekodiranja u okružjima MS Windows i Linux. Obavezno je koristiti se funkcijama kodiranja podataka radi potreba nadzornih tijela.

3.1. MS WINDOWS

U okružju MS Windows zaštićeni program upotrebljava biblioteku procedura MJENJACI.DLL koja u sebi sadržava sljedeće procedure:

- KoderTL
- DekoderTL
- KoderTransakcija
- DekoderTransakcija.

3.1.1. KoderTL

Deklaracija procedure (napisana u C sintaksi):

```
void KoderTL (char brojtl[4], char datumtl[9], char valutan[4], char
valutac[4], char vrstat[4], char kupovnit[11], char prodajnit[11],
char miniznos[11], char kodirani_podaci[70]);
```

Ulazni parametri:

brojtl	– broj tečajne liste
datumtl	– datum primjene tečajne liste
valutan	– brojčana oznaka valute
valutac	– slovna oznaka valute
vrstat	– vrsta tečaja (prema odluci o mogućim vrstama mjenjačkih tečajeva)
kupovnit	– kupovni tečaj strane valute
prodajnit	– prodajni tečaj strane valute
miniznos	– minimalni iznos strane valute za koji vrijedi objavljeni tečaj

Izlazni parametar:

kodirani_podaci	– kodirani ulazni podaci za kontrolu od strane nadzornih tijela
-----------------	---

Svi parametri na ulazu u proceduru moraju biti u obliku znakovnog niza (string). Ulazni parametri koji su inače numeričkog ili datumskog tipa trebaju prije poziva procedure biti pretvoreni u znakovni tip podataka. Datum za koji vrijedi tečajna lista treba biti pretvoren

u oblik DDMMGGGG. Kod numeričkih podataka, ako ne sadržavaju decimalni zarez, potrebno je dodati vodeće nule, a ako sadržavaju decimalni zarez, do punog formata moguće je dodati nule na početak ili kraj broja. Kodirani podaci za upis u tablicu *Kodirani tecajevi* vraćaju se u parametru *kodirani_podaci*. Taj se podatak zajedno s odgovarajućim podacima iz tablice *Tecajevi* (primarni ključ te tablice) koji su nekodirani, smješta u jedan redak tablice *Kodirani tecajevi*.

3.1.2. DekoderTL

Deklaracija procedure (napisana u C sintaksi):

```
void DekoderTL (char brojtl[4], char datumtl[9], char valutan[4],
char valutac[4], char vrstat[4], char kupovnit[11], char prodajnit[11],
char miniznos[11], char kodirani_podaci[70]);
```

Ulazni parametar:

kodirani_podaci – kodirani ulazni podaci za kontrolu od strane nadzornih tijela

Izlazni parametri:

brojtl – broj tečajne liste
datumtl – datum primjene tečajne liste
valutan – brojčana oznaka valute
valutac – slovna oznaka valute
vrstat – vrsta tečaja (prema odluci o mogućim vrstama mjenjačkih tečajeva)
kupovnit – kupovni tečaj strane valute
prodajnit – prodajni tečaj strane valute
miniznos – minimalni iznos strane valute za koji vrijedi objavljeni tečaj

Ulazni/izlazni parametri samo su zamijenili mjesta. Po obliku su ostali isti, tako da za uredno kodirane podatke iz ulaznog niza dobivamo i uredno dekodirane podatke na izlazu. Dekodirani podaci upotrebljavaju se za potrebe nadzornog tijela, a mogu se upotrebljavati i za vlastite potrebe (izrada potrebnih izvještaja).

3.1.3. KoderTransakcija

Deklaracija procedure (napisana u C sintaksi):

```
void KoderTransakcija (char brojtl[4], char datumtl[9], char valutan[4],
char valutac[4], char vrstatran[9], char sbtransakcije[11], char datvritran[13],
char sifraoo[9], char akronimt[9], char iznuval[12], char iznukn[12], char primtec[11],
char iznprov[12], char platinstrkn[5], char oznrez[2], char sbtiskobr[13], char tiptran[3],
char kodirani_podaci[145]);
```

Ulazni parametri:

brojtl – broj tečajne liste
datumtl – datum primjene tečajne liste
valutan – brojčana oznaka valute
valutac – slovna oznaka valute
vrstatran – vrsta transakcije
sbtransakcije – serijski broj transakcije
datvritran – datum i vrijeme obavljene transakcije
sifraoo – šifra odgovorne osobe
akronimt – akronim transakcije propisan ovom Uputom
iznuval – iznos transakcije u valuti

iznukn – iznos transakcije u kunama
primtec – tečaj primijenjen u obavljanju transakcije
iznprov – iznos mjenjačke provizije u kunama
platinstrkn – oznaka platnog instrumenta u kunama propisana Uputom
oznrez – oznaka rezidentnosti
sbtiskobr – serijski broj tiskanog obrasca koji se upotrebljava u slučajevima *više sile*
tiptran – unosi se vrijednost za tip transakcije (RD – redovna transakcija, ST – storno transakcija)

Izlazni parametar:

kodirani_podaci – kodirani ulazni podaci za kontrolu od strane nadzornih tijela

Svi parametri na ulazu u proceduru moraju biti u obliku znakovnog niza (string). Ulazni parametri koji su inače numeričkog ili datumskog tipa trebaju prije poziva procedure biti pretvoreni u znakovni tip podataka. Datum za koji vrijedi tečajna lista treba biti pretvoren u oblik DDMMGGGG. Datum i vrijeme transakcije trebaju biti pretvoreni u oblik DDMMGGGGHHMI, gdje su HH sati od 00 do 23, a MI minute od 00 do 59. Kod numeričkih podataka, ako ne sadržavaju decimalni zarez, potrebno je dodati vodeće nule, a ako sadržavaju decimalni zarez, do punog formata moguće je dodati nule na početak ili kraj broja. Kodirani podaci za upis u tablicu *Kodirane blagajničke transakcije* vraćaju se u parametru *kodirani_podaci*. Taj se podatak, zajedno s generičkim primarnim ključem i odgovarajućim podacima iz tablice *Blagajničke transakcije* (primarni ključ te tablice) koji su nekodirani, smješta u jedan redak tablice *Kodirane blagajničke transakcije*.

3.1.4. DekoderTransakcija

Deklaracija procedure (napisana u C sintaksi):

```
void DekoderTransakcija (char brojtl[4], char datumtl[9], char valutan[4], char valutac[4], char vrstatran[9],
char sbtransakcije[11], char datvritran[13], char sifraoo[9], char akronimt[9], char iznuval[12],
char iznukn[12], char primtec[11], char iznprov[12], char platinstrkn[5], char oznrez[2], char sbtiskobr[13],
char tiptran[3], char kodirani_podaci[145]);
```

Ulazni parametar:

kodirani_podaci – kodirani ulazni podaci za kontrolu od strane nadzornih tijela

Izlazni parametri:

brojtl – broj tečajne liste
datumtl – datum primjene tečajne liste
valutan – brojčana oznaka valute
valutac – slovna oznaka valute
vrstatran – vrsta transakcije
sbtransakcije – serijski broj transakcije
datvritran – datum i vrijeme obavljene transakcije
sifraoo – šifra odgovorne osobe
akronimt – akronim transakcije propisan ovom Uputom
iznuval – iznos transakcije u valuti

iznukn	– iznos transakcije u kunama
primtec	– tečaj primijenjen u obavljanju transakcije
iznprov	– iznos mjenjačke provizije u kunama
platinstrkn	– oznaka platnog instrumenta u kunama propisana Uputom
oznrez	– oznaka rezidentnosti
sbtiskobr	– serijski broj tiskanog obrasca koji se upotrebljava u slučajevima <i>više sile</i>
tiptran	– unosi se vrijednost za tip transakcije (RD – redovna transakcija, ST – storno transakcija)

Ulazni/izlazni parametri samo su zamijenili mjesta. Po obliku su ostali isti, tako da za uredno kodirane podatke iz ulaznog niza dobivamo i uredno dekodirane podatke na izlazu. Dekodirani podaci upotrebljavaju se za potrebe nadzornog tijela, a mogu se upotrebljavati i za vlastite potrebe (izrada potrebnih izvještaja).

3.2. LINUX

U okruženju Linux zaštićeni program upotrebljava četiri izvršna programa:

- KoderTL
- DekoderTL
- KoderTr
- DekoderTr.

Razmjena vrijednosti između tih programa i zaštićenoga programa odvija se preko ASCII tekstnih datoteka. To se radi tako da se formira odgovarajuća ulazna datoteka za te programe, a nakon što su pozvani ti programi, oni upisuju izlazne vrijednosti u izlaznu datoteku. Sve datoteke imaju isto ime, a razlikuju se prema sufiks, tako da ulazne datoteke imaju sufiks *.In, a izlazne datoteke imaju sufiks *.Out.

3.2.1. KoderTL

Deklaracija programa (napisana u C sintaksi):

KoderTL;

Ulazni parametri:

brojtl	– broj tečajne liste
datumtl	– datum primjene tečajne liste
valutan	– brojčana oznaka valute
valutac	– slovna oznaka valute
vrstat	– vrsta tečaja (prema odluci o mogućim vrstama mjenjačkih tečajeva)
kupovnit	– kupovni tečaj strane valute
prodajnit	– prodajni tečaj strane valute
miniznos	– minimalni iznos strane valute za koji vrijedi objavljeni tečaj

Izlazni parametar:

kodirani_podaci	– kodirani ulazni podaci za kontrolu od strane nadzornih tijela
-----------------	---

Budući da je ulazna datoteka *KoderTL.In* ASCII tekstna datoteka, svi parametri moraju biti u obliku znakovnog niza (string). Ulazni parametri koji su inače numeričkog ili datumskog tipa trebaju prije poziva procedure biti pretvoreni u znakovni tip podataka. Datum za koji vrijedi tečajna lista treba biti pretvoren u oblik DDMMGGGG. Kod numeričkih podataka, ako ne sadržavaju decimalni zarez, po-

trebno je dodati vodeće nule, a ako sadržavaju decimalni zarez, do punog formata moguće je dodati nule na početak ili kraj broja. Kodirani podaci za upis u tablicu *Kodirani tečajevi* vraćaju se u ASCII tekstnoj datoteci *KoderTL.Out*. Taj se podatak, zajedno s odgovarajućim podacima iz tablice *Tecajevi* (primarni ključ te tablice) koji su nekodirani, smješta u jedan redak tablice *Kodirani tečajevi*.

Primjer izgleda ulazne datoteke:

```
098
10052003
978
EUR
RED
7.52300000
7.58700000
0000000000
```

Primjer izgleda izlazne datoteke:

```
a` ,f%oo"•d"±f dŽš,ÍÓŽřĚ,ŮŮ™et a"aq̇ .Ůí €Łó@tZ „ŮeŮ-Ů"Čeas-iaUnř ĀiZ.ÓĚĚ
```

3.2.2. DekoderTL

Deklaracija programa (napisana u C sintaksi):

DekoderTL;

Ulazni parametar:

kodirani_podaci	– kodirani ulazni podaci za kontrolu od strane nadzornih tijela
-----------------	---

Izlazni parametri:

brojtl	– broj tečajne liste
datumtl	– datum primjene tečajne liste
valutan	– brojčana oznaka valute
valutac	– slovna oznaka valute
vrstat	– vrsta tečaja (prema odluci o mogućim vrstama mjenjačkih tečajeva)
kupovnit	– kupovni tečaj strane valute
prodajnit	– prodajni tečaj strane valute
miniznos	– minimalni iznos strane valute za koji vrijedi objavljeni tečaj

Uredno kodirani podaci iz ulaznog niza daju i uredno dekodirane podatke na izlazu. Kodirani podaci ulaze u proces dekodiranja zapisani u ASCII tekstnoj datoteci *DekoderTL.In*. Dekodirani podaci za daljnju upotrebu u okviru zaštićenoga programa vraćaju se u ASCII tekstnoj datoteci *DekoderTL.Out*. Podaci se vraćaju u istom obliku u kojem su ušli u program KoderTL i u istom poretku jedan ispod drugoga. Dekodirani podaci upotrebljavaju se za potrebe nadzornog tijela, a mogu se upotrebljavati i za vlastite potrebe (izrada potrebnih izvještaja).

Ulazna i izlazna datoteka izgledaju kao zamijenjene datoteke programa KoderTL.

3.2.3. KoderTr

Deklaracija programa (napisana u C sintaksi):

KoderTr;

Ulazni parametri:

brojtl	– broj tečajne liste
datumtl	– datum primjene tečajne liste
valutan	– brojčana oznaka valute

valutac	– slovna oznaka valute
vrstatran	– vrsta transakcije
sbtransakcije	– serijski broj transakcije
datvritran	– datum i vrijeme obavljene transakcije
sifraoo	– šifra odgovorne osobe
akronimt	– akronim transakcije propisan ovom Uputom
iznuval	– iznos transakcije u valuti
iznukn	– iznos transakcije u kunama
primtec	– tečaj primijenjen u obavljanju transakcije
iznprov	– iznos mjenjačke provizije u kunama
platinstrkn	– oznaka platnog instrumenta u kunama propisana Uputom
oznrez	– oznaka rezidentnosti
sbtiskobr	– serijski broj tiskanog obrasca koji se upotrebljava u slučajevima <i>više sile</i>
tiptran	– unosi se vrijednost za tip transakcije (RD – redovna transakcija, ST – storno transakcija)
Izlazni parametar:	
kodirani_podaci	– kodirani ulazni podaci za kontrolu od strane nadzornih tijela

Budući da je ulazna datoteka *KoderTr.In* ASCII tekstna datoteka, svi parametri moraju biti u obliku znakovnog niza (string). Ulazni parametri koji su inače numeričkog ili datumskog tipa trebaju prije poziva procedure biti pretvoreni u znakovni tip podataka. Datum za koji vrijedi tečajna lista treba biti pretvoren u oblik DDMMGGGG. Datum i vrijeme transakcije trebaju biti pretvoreni u oblik DDM-MGGGGHHMI, gdje su HH sati od 00 do 23, a MI minute od 00 do 59. Kod numeričkih podataka, ako ne sadržavaju decimalni zarez, potrebno je dodati vodeće nule, a ako sadržavaju decimalni zarez, do punog formata moguće je dodati nule na početak ili kraj broja. Kodirani podaci za upis u tablicu *Kodirane blagajničke transakcije* vraćaju se u ASCII tekstnoj datoteci *KoderTr.Out*. Taj se podatak, zajedno s generičkim primarnim ključem i odgovarajućim podacima iz tablice *Blagajničke transakcije* (primarni ključ te tablice) koji su nekodirani, smješta u jedan redak tablice *Kodirane blagajničke transakcije*.

Primjer izgleda ulazne datoteke:

```
098
10052003
978
EUR
Otkup
0000000710
100520031756
23236565
OTKGOT
100.0000000
752.3000000
7.52300000
2.700000000
GOT
N
RD
```

Primjer izgleda izlazne datoteke:

```
±áŠo©Üü·û·ŁšİŮr TDLÝ·R·Dš×°Dđšb%Ü·äSđ%ööü·|EEŸS°NēĀ"ŪēUāĀđ'6
"š@T°™öS·Šr ē·šá^YBēsñZaŮCİĖ□™·Nēē°T·Z°ö°"ēİZf LŠ·zĖē·ā!±@š·
NšŸs°·üđŁŪā"zİĖ
```

3.2.4. DekoderTr

Deklaracija programa (napisana u C sintaksi):

DekoderTr;

Ulazni parametar:

kodirani_podaci – kodirani ulazni podaci za kontrolu od strane nadzornih tijela

Izlazni parametri:

brojtl – broj tečajne liste
 datumtl – datum primjene tečajne liste
 valutn – brojčana oznaka valute
 valutac – slovna oznaka valute
 vrstatran – vrsta transakcije
 sbtransakcije – serijski broj transakcije
 datvritran – datum i vrijeme obavljene transakcije
 sifraoo – šifra odgovorne osobe
 akronimt – akronim transakcije propisan ovom Uputom
 iznuval – iznos transakcije u valuti
 iznukn – iznos transakcije u kunama
 primtec – tečaj primijenjen u obavljanju transakcije
 iznprov – iznos mjenjačke provizije u kunama
 platinstrkn – oznaka platnog instrumenta u kunama propisana Uputom
 oznrez – oznaka rezidentnosti
 sbtiskobr – serijski broj tiskanog obrasca koji se rabi u slučajevima *više sile*
 tiptran – unosi se vrijednost za tip transakcije (RD – redovna transakcija, ST – storno transakcija)

Uredno kodirani podaci iz ulaznog niza daju i uredno dekodirane podatke na izlazu. Kodirani podaci ulaze u proces dekodiranja zapisani u ASCII tekstnoj datoteci *DekoderTr.In*. Dekodirani podaci za daljnju upotrebu u okviru zaštićenoga programa vraćaju se u ASCII tekstnoj datoteci *DekoderTr.Out*. Podaci se vraćaju u istom obliku u kojem su ušli u program *KoderTr* i u istom poretaku jedan ispod drugoga. Dekodirani podaci upotrebljavaju se za potrebe nadzornog tijela, a mogu se upotrebljavati i za vlastite potrebe (izrada potrebnih izvještaja).

Ulazna i izlazna datoteka izgledaju kao zamijenjene datoteke programa *KoderTr*.

4. IZVOZ PODATAKA ZA POTREBE NADZORNIH TIJELA

4.1 IZVOZ SAMO KODIRANIH PODATAKA

4.1.1. Izvoz podataka tečajne liste

Potrebno je omogućiti izvoz samo kodiranih podataka tečajne liste za zadano razdoblje. Podaci se izvoze u obliku ASCII tekstne datoteke.

4.1.2. Izvoz podataka blagajničkih transakcija

Potrebno je omogućiti izvoz samo kodiranih podataka blagajničkih transakcija za zadano razdoblje. Podaci se izvoze u obliku ASCII tekstne datoteke.

4.1.3. Izvoz podataka tečajne liste i blagajničkih transakcija

Potrebno je omogućiti izvoz kodiranih podataka tečajne liste i blagajničkih transakcija za zadano razdoblje. Podaci se izvoze tako da idu redom za određeni datum najprije podaci o tečajnoj listi, a zatim podaci o blagajničkim transakcijama. Podaci se izvoze u obliku ASCII tekstne datoteke.

4.2. IZVOZ KODIRANIH I NEKODIRANIH PODATAKA

4.2.1 Izvoz podataka tečajne liste

Potrebno je omogućiti izvoz kodiranih i nekodiranih podataka tečajne liste za zadano razdoblje. Podaci se izvoze u obliku ASCII tekstne datoteke. Podaci se smještaju jedni pokraj drugih u istom retku izvozne datoteke.

4.2.2. Izvoz podataka

Potrebno je omogućiti izvoz kodiranih i nekodiranih podataka blagajničkih transakcija za zadano razdoblje. Podaci se izvoze u obliku ASCII tekstne datoteke. Podaci se smještaju jedni pokraj drugih u istom retku izvozne datoteke.

4.2.3. Izvoz podataka

Potrebno je omogućiti izvoz kodiranih i nekodiranih podataka tečajne liste i blagajničkih transakcija za zadano razdoblje. Podaci se izvoze tako da idu redom za određeni datum najprije podaci o tečajnoj listi, a zatim podaci o blagajničkim transakcijama. Podaci se izvoze u obliku ASCII tekstne datoteke. Podaci se smještaju jedni pokraj drugih u istom retku izvozne datoteke.

5. IZMJENE U ZAŠTIĆENOM PROGRAMU

Sve izmjene načinjene u zaštićenom programu nakon što je certifikat već izdan, podliježu novom postupku izdavanja certifikata.

6. PRILOZI ZA CERTIFICIRANJE

Da bi zaštićeni program ušao u postupak certificiranja, pri predaji programskog rješenja potrebno je priložiti:

- instalacijsku verziju programa
- opis instalacijske procedure
- korisničku Uputu
- bazu podataka na kojoj je zasnovano programsko rješenje
- izvorni kod programskog rješenja u digitalnom obliku.

7. DOSTAVA IZVJEŠĆA U ELEKTRONIČKOM OBLIKU

7.1. Izvješće »Izvješće o kupnji odnosno prodaji strane gotovine i otkupu čekova koji glase na stranu valutu za razdoblje _____«

Ovlašteni mjenjač može dostavljati, ako tako ugovori s ugovornom bankom, izvješće »Izvješće o kupnji odnosno prodaji strane gotovine i otkupu čekova koji glase na stranu valutu za razdoblje _____« u elektroničkom obliku. U tom slučaju izvještaj se dostavlja u obliku XML datoteke, prema shemi (MJEN_mjenjaci.xsd):

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:element name="MJEN" type="sveType">
    <xs:annotation>
      <xs:documentation>Promet ovlaštenog mjenjača u
razdoblju</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:complexType name="sveType">
    <xs:sequence>
      <xs:element name="Zaglavlje" type="ZaglavljeType"/>
      <xs:element name="GotovinaR" type="GotovinaRType"
minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="GotovinaN" type="GotovinaNType"
minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="CekoviR" type="CekoviRType" minOccurs="0"
maxOccurs="unbounded"/>
      <xs:element name="CekoviN" type="CekoviNType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="ZaglavljeType">
    <xs:sequence>
      <xs:element name="OIB_Mjenjaca">
```

```

        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:length value="11"/>
                <xs:pattern value="([0-9]{11})"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:element>
    <xs:element name="TvrtkaMjenjaca">
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:minLength value="1"/>
                <xs:maxLength value="2000"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:element>
    <xs:element name="Oznaka_Mjenjackog_mjesta" minOccurs="0">
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:minLength value="1"/>
                <xs:maxLength value="100"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:element>
    <xs:element name="Adresa_Mjenjackog_mjesta" minOccurs="0">
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:minLength value="1"/>
                <xs:maxLength value="200"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:element>
    <xs:element name="OIB_banke">
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:length value="11"/>
                <xs:pattern value="([0-9]{11})"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:element>
    <xs:element name="IzvjestajnoRazdoblje">
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:length value="6"/>
                <xs:pattern value="((2)[0-9]{3})(0[1-
9]|[1012])"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:element>
    <xs:element name="RedniBrojSlanja" type="xs:integer"/>
    <xs:element name="Osoba" type="OsobaType" minOccurs="0"
maxOccurs="3"/>
</xs:sequence>
</xs:complexType>

```

```

<xs:complexType name="OsobaType">
  <xs:sequence>
    <xs:element name="Ime">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:minLength value="1"/>
          <xs:maxLength value="50"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="Prezime">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:minLength value="1"/>
          <xs:maxLength value="50"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="EMail" minOccurs="0">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:minLength value="1"/>
          <xs:maxLength value="50"/>
          <xs:pattern value="[A-Za-z0-9\._%+\-
]+@[A-Za-z0-9\.\-]+\.[A-Za-z]{2,4}"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="Telefon" minOccurs="0">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:minLength value="6"/>
          <xs:maxLength value="14"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="GotovinaRType">
  <xs:sequence>
    <xs:element name="SifraValute">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:length value="3"/>
          <xs:pattern value="([0-9]{3})"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="Rezidentnost">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:length value="1"/>
          <xs:pattern value="(R)"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
  </xs:sequence>
</xs:complexType>

```

```

        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element      name="IznosKupnjaR"      type="IznosType"
minOccurs="0"/>
    <xs:element      name="IznosProdajaR"     type="IznosType"
minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="GotovinaNType">
  <xs:sequence>
    <xs:element name="SifraValute">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:length value="3"/>
          <xs:pattern value="([0-9]{3})"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="Rezidentnost">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:length value="1"/>
          <xs:pattern value="(N)"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element      name="IznosKupnjaN"      type="IznosType"
minOccurs="0"/>
    <xs:element      name="IznosProdajaN"     type="IznosType"
minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="CekoviRType">
  <xs:sequence>
    <xs:element name="SifraValute">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:length value="3"/>
          <xs:pattern value="([0-9]{3})"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="Rezidentnost">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:length value="1"/>
          <xs:pattern value="(R)"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="IznosR" type="IznosType" minOccurs="1"/>
  </xs:sequence>

```

```

</xs:complexType>
<xs:complexType name="CekoviNType">
  <xs:sequence>
    <xs:element name="SifraValute">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:length value="3"/>
          <xs:pattern value="([0-9]{3})"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="Rezidentnost">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:length value="1"/>
          <xs:pattern value="(N)"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="IznosN" type="IznosType" minOccurs="1"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="IznosType">
  <xs:sequence>
    <xs:element name="IznosUValuti">
      <xs:simpleType>
        <xs:restriction base="xs:long">
          <xs:totalDigits value="38"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="IznosUKunama">
      <xs:simpleType>
        <xs:restriction base="xs:long">
          <xs:totalDigits value="38"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
  </xs:sequence>
</xs:complexType>
</xs:schema>

```

7.2. Izvješće "Promet ovlaštenih mjenjača – zbirno izvješće o kupnji i prodaji strane gotovine i otkupljenim čekovima koji glase na stranu valutu"

Ugovorne banke dostavljaju Hrvatskoj narodnoj banci izvješće "Promet ovlaštenih mjenjača – zbirno izvješće o kupnji odnosno prodaji strane gotovine i otkupljenim čekovima koji glase na stranu valutu" – Prilog br. 3A u elektroničkom obliku. Prilog 3A sastavni je dio ove Upute. Izvještaj se dostavlja u obliku XML datoteke, prema shemi (MJEN_banke.xsd):

```

<?xml version="1.0" encoding="ISO-8859-2"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:element name="MJEN" type="sveType">
    <xs:annotation>
      <xs:documentation>Zbirni promet ovlaštenih
mjenjača</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:complexType name="sveType">
    <xs:sequence>
      <xs:element name="Zaglavlje" type="ZaglavljeType"/>
      <xs:element name="GotovinaR" type="GotovinaRType"
minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="GotovinaN" type="GotovinaNType"
minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="CekoviR" type="CekoviRType" minOccurs="0"
maxOccurs="unbounded"/>
      <xs:element name="CekoviN" type="CekoviNType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="ZaglavljeType">
    <xs:sequence>
      <xs:element name="OIB">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:length value="11"/>
            <xs:pattern value="[0-9]{11}"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="IzvjestajnoRazdoblje">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:length value="6"/>
            <xs:pattern value="((2)[0-9]{3})(0[1-
9]|1[012])"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="RedniBrojSlanja" type="xs:integer"/>
      <xs:element name="Osoba" type="OsobaType" minOccurs="1"
maxOccurs="3"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="OsobaType">
    <xs:sequence>
      <xs:element name="Ime">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:minLength value="1"/>
            <xs:maxLength value="50"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:schema>

```

```

        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="Prezime">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:minLength value="1"/>
          <xs:maxLength value="50"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="EMail">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:minLength value="1"/>
          <xs:maxLength value="50"/>
          <xs:pattern value="[A-Za-z0-9\._%\\+\\-
]+@[A-Za-z0-9\\._\\-]+\\.[A-Za-z]{2,4}"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="Telefon">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:minLength value="6"/>
          <xs:maxLength value="14"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="GotovinaRType">
  <xs:sequence>
    <xs:element name="SifraValute">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:length value="3"/>
          <xs:pattern value="([0-9]{3})"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="Rezidentnost">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:length value="1"/>
          <xs:pattern value="(R)"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="IznosKupnjaR" type="IznosType"
minOccurs="0"/>
    <xs:element name="IznosProdajaR" type="IznosType"
minOccurs="0"/>

```

```

        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="GotovinaNType">
        <xs:sequence>
            <xs:element name="SifraValute">
                <xs:simpleType>
                    <xs:restriction base="xs:string">
                        <xs:length value="3"/>
                        <xs:pattern value="([0-9]{3})"/>
                    </xs:restriction>
                </xs:simpleType>
            </xs:element>
            <xs:element name="Rezidentnost">
                <xs:simpleType>
                    <xs:restriction base="xs:string">
                        <xs:length value="1"/>
                        <xs:pattern value="(N)"/>
                    </xs:restriction>
                </xs:simpleType>
            </xs:element>
            <xs:element name="IznosKupnjaN" type="IznosType"
minOccurs="0"/>
            <xs:element name="IznosProdajaN" type="IznosType"
minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="CekoviRType">
        <xs:sequence>
            <xs:element name="SifraValute">
                <xs:simpleType>
                    <xs:restriction base="xs:string">
                        <xs:length value="3"/>
                        <xs:pattern value="([0-9]{3})"/>
                    </xs:restriction>
                </xs:simpleType>
            </xs:element>
            <xs:element name="Rezidentnost">
                <xs:simpleType>
                    <xs:restriction base="xs:string">
                        <xs:length value="1"/>
                        <xs:pattern value="(R)"/>
                    </xs:restriction>
                </xs:simpleType>
            </xs:element>
            <xs:element name="IznosR" type="IznosType" minOccurs="1"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="CekoviNType">
        <xs:sequence>
            <xs:element name="SifraValute">
                <xs:simpleType>
                    <xs:restriction base="xs:string">
                        <xs:length value="3"/>

```

```

        <xs:pattern value="([0-9]{3})"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <xs:element name="Rezidentnost">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:length value="1"/>
        <xs:pattern value="(N)"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <xs:element name="IznosN" type="IznosType" minOccurs="1"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="IznosType">
  <xs:sequence>
    <xs:element name="IznosUValuti">
      <xs:simpleType>
        <xs:restriction base="xs:long">
          <xs:totalDigits value="38"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="IznosUKunama">
      <xs:simpleType>
        <xs:restriction base="xs:long">
          <xs:totalDigits value="38"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
  </xs:sequence>
</xs:complexType>
</xs:schema>
```

PRILOG BR. 1

(tvrtka i OIB ovlaštenog mjenjača)

(tvrtka banke)

Blagajnički dnevnik za evidentiranje transakcija

Datum: dd.mm.gggg

Sifra odgovorne osobe

Red. br. transakcije	Vrijeme transakcije	Opis pojedine transakcije	Serijski broj	Valuta	Tečaj	Platni instrument u stranoj gotovini	Primici u stranoj gotovini	Izdaci u stranoj gotovini	Platni instrument u kunama	Primici u kunama	Izdaci u kunama	Provizija u kunama
1	2	3	4	5	6	7	8	9	10	11	12	13

PRILOG BR. 2

(tvrtka i OIB ovlaštenog mjenjača)

(tvrtka banke)

Obračun dnevne blagajne – stanje blagajničkog dnevnika
na temelju tečajne liste redni broj: _____
primijenjene dana _____ u vremenu od _____ do _____

Sifra odgovorne osobe

Valuta	Paritet	Tečaj	Početno stanje blagajne	Obračunska tečajna razlika po valutama	Primatka od mjeni. poslova	Izdatka mjeni. poslova	Prilod od finansijske dobiti	Prilod od provizije	Uplata u kunama	Iplata u kunama	Uplata u stranoj gotovini	Iplata u stranoj gotovini	Obrač. tečajna razlika	Konačno stanje blagajne	Stvarno stanje blagajne	Razlika	Višak	Manjak
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17 (16 – 15)	18	19
					Primici u kunsku blagajnu na osnovi prodaje strane valute GOT:													
Ukupno u kn:																		

Ukupna obračunska tečajna razlika iz stupca 5: kn
Ukupna obračunska tečajna razlika iz stupca 14: kn
Ukupna tečajna razlika od kupoprodaje: kn
(prilod od finansijske dobiti – stupac 8) kn
Ukupna tečajna razlika od prodaje banči na otkup: kn
Ukupno naplaćena provizija (stupac 9): kn
Ukupna provizija banke na predanu stranu gotovinu i čekove: kn
Ukupno kn

Otkup je izvršio:
Prodaju je izvršio:

Potpis odgovorne osobe

PRILOG BR. 3

(tvrtka i OIB ovlaštenog mjenjača)

(tvrtka banke)

Izvešće o kupnji odnosno prodaji strane gotovine i otkupu čekova koji glase na stranu valutu
za razdoblje _____u apsolutnim iznosima strane gotovine
u apsolutnim iznosima kn

Redni broj	Šifra valute	Oznaka valute	KUPNJA STRANE GOTOVINE		PRODAJA STRANE GOTOVINE	
			Iznos (valute)	Iznos(kn)	Iznos (valute)	Iznos (kn)
1	2	3	4	5	6	7
REZIDENTI						
NEREZIDENTI						

Redni broj	Šifra valute	Oznaka valute	Otkup čekova koji glase na stranu valutu	
			Iznos (valute)	Iznos (kn)
1	2	3	4	5
REZIDENTI				
NEREZIDENTI				

potpis odgovorne osobe_____
pečat ovlaštenog mjenjača

PRILOG BR. 3A

(tvrtka banke)

Promet ovlaštenih mjenjača

Zbirno izvješće o kupnji odnosno prodaji strane gotovine i otkupljenim čekovima koji glase na stranu valu
za razdoblje _____

u apsolutnim iznosima strane gotovine
u apsolutnim iznosima kn

Redni broj	Šifra valute	Oznaka valute	KUPNJA STRANE GOTOVINE		PRODAJA STRANE GOTOVINE	
			Iznos (valute)	Iznos (kn)	Iznos (valute)	Iznos (kn)
1	2	3	4	5	6	7
REZIDENTI						
NEREZIDENTI						

Redni broj	Šifra valute	Oznaka valute	Otkup čekova koji glase na stranu valutu	
			Iznos (valute)	Iznos (kn)
1	2	3	4	5
REZIDENTI				
NEREZIDENTI				

(izradio/izradila)

(ovjera ovlaštene osobe)

OTKUPLJENA STRANA GOTOVINA						
Serijski broj potvrde o otkupu	Brojčana i slovna oznaka valute	Iznos otkupljene valute		Ukupno	Kupovni tečaj	Protuvrijednost u kunama
		Rezidenti	Nerezidenti			
1	2	3	4	5	6	7
					Ukupno u kunama:	
OTKUPLJENI ČEKOVI KOJI GLASE NA STRANU VALUTU						
Serijski broj potvrde o otkupu	Brojčana i slovna oznaka valute	Iznos otkupljene valute		Ukupno	Kupovni tečaj	Protuvrijednost u kunama
		Rezidenti	Nerezidenti			
1	2	3	4	5	6	7
					Ukupno u kunama:	
					Sveukupno u kunama:	

Potpis ovlaštene osobe u banci
Pečat banke