ETAPA II

RAPORT STIINTIFIC SI TEHNIC

ELABORAREA SPECIFICAŢIILLOR SOFTWARE CORESPUNZATOARE SMART CARDURILOR, PROIECTAREA BAZELOR DE DATE, A APLICATIILOR SOFTWARE SI INTEGRAREA COMPONENTELOR SRSPIRIM

CUPRINS

1 ARHITECTURA GENERALA A SISTEMULUI SRSPIRIM

Proiectul SRSPRIM are ca obiectiv central implementarea unui sistem pentru monitorizarea investigatiilor radiologice in cadrul unui mediu medical. Aplicatia este formata din mai multe module care prin relatia dintre ele ofera functionalitatile necesare. Aceste module sunt:

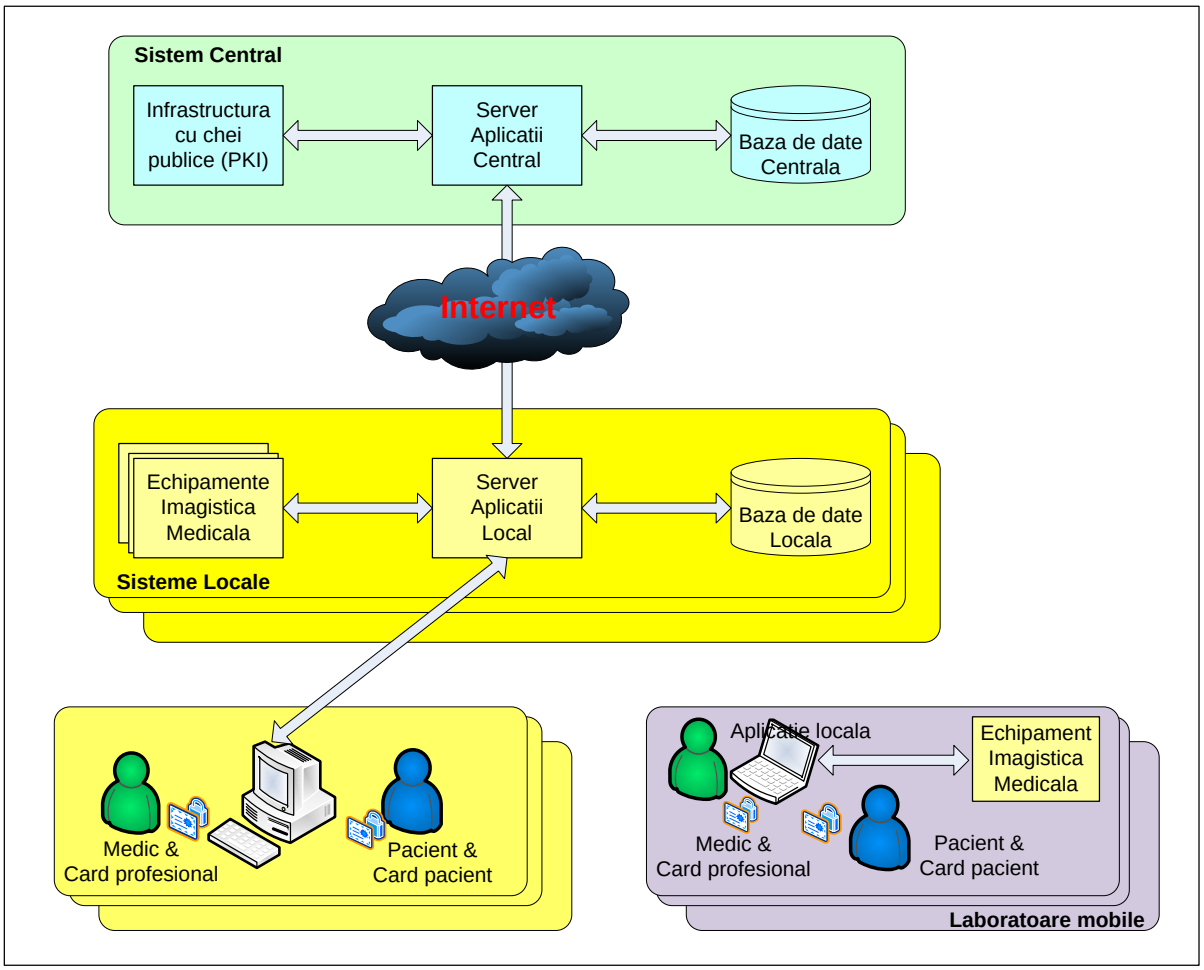
a) aplicatiile care se afla pe smart carduri, numite si aplicatii on-card. Acestea sunt incarcate pe carduri in cadrul procesului de personalizare.

b) aplicatii care vor rula pe statiile de lucru ale personalului medical (medic de familie, medic radiolog), numite si aplicatii off-card.

c) bazele de date:

- medicale folosite pentru a stoca volumul de date medicale ale pacientilor care sunt retinute in cadrul sistemului.
- administrative (pentru managementul cardurilor si a certificatelor din cadrul infratsrucurii PKI)

d) aplicatiile de gestiune a cardurilor, serverele folosite in cardul infrastructurii PKI.



FIGURĂ 1-1 ARHITECTURA GENERALA A SISTEMULUI SRSPIRIM

Sistemul este fondat in jurul celor doua tipuri de carduri:

- cardul medical radiologic destinat pacienţilor : are rolul de a pastra datele medicale ale unui pacient precum si dozele la care a fost expus pe durata investigatiilor radiologice.
- cardul medical profesionist destinat medicilor : are rolul de autentificare la aplicatiile sistemului precum si de semnare a datelor medicale (pe card si in baza de date).

Principalele aplicaţii off-card care au fost proiectate si implementate sunt:

- aplicaţia de personalizare a cardurilor electronice radiologice.
- aplicaţia de semnare a documentelor medicale.
- aplicaţia de personalizare si vizualizare a fisei radiologice.

La aplicaţiile enumerate anterior se adăuga setul de servicii necesare implementării unei infrastructuri PKI precum si setul de aplicaţii off-card web la nivelul componentei E-Health CMS pentru administrarea cardurilor medicale si a celor profesionale.

Roluri in cadrul sistemului:

- OPERATOR CARD
- MEDIC DE FAMILIE
- MEDIC RADIOLOG
- PACIENT

In cadrul sistemului SRSPIRIM au fost dezvoltate doua clase de aplicatii: cele on-card prezente pe smart carduri numite si appleti si cele off-card, aplicatii web ce permit comunicarea atat cu appletii de pe card cat si cu bazele de date in mod securizat.

De asemenea, in cadrul proiectului au fost utilizate si o serie de servere si aplicatii web care permit implementarea si functionarea unei infrastructuri PKI complete. Arhitectura SRSPIRIM fiind bazata pe o serie de roluri pe care actorii le joaca in cadrul sistemului, pentru fiecare dintre aceste roluri se vor preda cate o mostra de card medical personalizat cu aplicatiile dezvoltate.

In cadrul activitatii de experimentare a sistemului informatic SRSPIRIM au fost efectuate o serie de teste pentru fiecare dintre categoriile de aplicatii dezvoltate in cadrul proiectului. A fost intocmit un raport de experimentare care detaliaza modelul experimental, procedurile de testare, planul de testare, detalierea testelor efectuate si rezultatele obtinute.

2 OBIECTIVELE ETAPEI DE EXECUTIE

Etapa a II-a a proiectului SRSPIRIM şi-a propus îndeplinirea următoarelor obiective:

1) Elaborarea specificatiilor pentru aplicaţiile care rezida pe smartcarduri, asa numitii appleti JavaCard. S-au avut in vedere urmatoarele probleme:
    a) descrierea functionalitatilor de baza ce trebuiesc asigurate de appleti
    b) entitatile principale din applet si atributele acestora

2) Elaborarea specificaţiilor privind infrastructura de chei publice folosita pentru asigurarea securitatii informatiilor medicale in cadrul proiectului. Pentru atingerea acestui obiectiv, au fost identificate urmatoarele probleme care trebuiesc abordate:
    a) Arhitectura generala a infrastructurii cu chei publice din SRSPIRIM
    b) Specificatiile pentru autoritatea rădăcina si autoritatile subordonate
    c) Specificatiile pentru autoritatea de înregistrare a persoanelor
    d) Fluxuri de date si operationale specifice administrarii cardurilor
    e) Specificatiile pentru autoritatea de validare on-line a certificatelor digitale

3) Proiectarea bazelor de date centrale şi locale. S-au avut in vedere urmatoarele aspecte care au fost abordate si detaliate in cadrul documentului:
    a) Arhitectura generala a bazelor de date din cadrul sistemului SRSPIRIM
    b) Structura bazei de date centrale si relatiile dintre tabelele componente
    c) Detalierea campurilor specifice fiecarui tabel din baza de date

4) Proiectarea aplicaţiilor software de pe smart carduri. In cadrul acestui obiectiv, au fost vizate si descrise urmatoarele aspecte:
    a) Standardele specifice smart cardurilor
    b) Principiile si mecanismele de comunicatie cu appletii JavaCard
    c) Autentificarea intre carduri, rolurile si drepuri de acces la informatii
    d) Structurile de date din clasele existente si interfetele appletilor JavaCard

5) Dezvoltarea aplicaţiilor software off-card. Acest obiectiv a fost atins prin rezolvarea urmatoarelor probleme:
    a) Descrierea functionalitatilor specifice aplicatiilor software off-card
    b) Descrierea componentelor software ale aplicatiilor off-card
    c) Structura logica a aplicatiei
    d) Modul de utilizare al mecanismelor de securitate din infrastructura PKI

6) Dezvoltarea aplicaţiilor software on-card. Acest obiectiv a fost atins prin rezolvarea urmatoarelor probleme:
    a) Prezentarea tehnologiei utilizate : JavaCard
    b) Descrierea claselor din componenta appletului pentru pacienti
    c) Conventiile folosite pentru codarea datelor in interiorul claselor
    d) Aspecte privind serializarea si deserializarea obiectelor

7) Integrarea şi testarea componentelor de bază ale sistemului SRSPIRIM. Acest obiectiv a fost atins prin rezolvarea urmatoarelor probleme:
    a) Configurarea si instalarea aplicatiilor dezvoltate in proiectul SRSPIRIM
    b) Testarea appletilor on-card folosind unelte din kitul de dezvoltare Oberthur
    c) Testarea aplicatiilor off-card in cele mai relevante scenarii de test

3 ELABORAREA SPECIFICATIILOR PENTRU APLICAŢIILE DE PE SMARTCARDURI

3.1 FUNCTIONALITATI DE BAZA ALE APLICATIILOR ON-CARD

Prima activitate din cadrul etapei a II-a a proiectului a constat in elaborarea specificatiilor pentru aplicaţiile de pe smartcarduri şi a design-ului din punct de vedere software a acestora in care au fost implicati partenerii P1 si P2.

In urma studiilor si analizelor impreuna cu personalul medical specializat, s-a ajuns la concluzia ca functionalitatile pe care trebuie sa le ofere aplicatiile On-Card prezente pe cardul radiologic al pacientului trebuie sa fie urmatoarele :

- autentificarea pacientului folosind un cod PIN ; posibilitatea de schimbare a codului PIN.
- autentificarea cardului profesional pentru a ne asigura de cel care citeste sau trimite informatiile ce trebuiesc stocate in interiorul cardului
- determinarea rolului celui care opereaza cu cardul (MEDIC DE FAMILIE /RADIOLOG)
- stocarea si citirea informatiilor administrative : nume, prenume, adresa, etc.
- stocarea si citirea informatiilor medicale de baza : grupa sangvina, RH, etc.
- stocarea si citirea altor informatii medicale : afectiuni, trimiteri, expuneri.
- metoda de calcul a dozei cumulative care sa integreze informatiile referitoare la dozele de radiatii folosite in expunerile precedente ale pacientului.

3.2 STRUCTURILE DE DATE ASOCIATE INFORMATIILOR DE PE CARD

Pentru o consultatie au fost desemnate urmatoarele informatii care se vor retine pe card:

- data si ora la care a avut loc consultatia
- medicul care a efectuat aceasta consultatie
- locatia in care s-a desfasurat consultatia
- descrierea sumara a problemelor medicale constatate
- diagnosticul prezumtiv

Pentru o anumita boala sunt stocate pe card urmatoarele informatii :

- data la care a fost diagnosticat pacientul
- denumirea exacta a diagnosticului final
- o scurta descriere specifica pacientului

Pentru un anumit pacient, s-au retinut pe card urmatoarele informatii :

- nume, prenume si initiala tatalui
- codul numeric personal (CNP)
- adresa postala pacientului
- adresa electronica a pacientului

- persoana de contact in caz de urgenta
- numarul de telefon in caz de urgenta
- grupa sangvina si RH-ul
- alergii cunoscute pana in prezent

Pentru o trimitere medicala, urmatoarele informatii sunt stocate pe card:

- data si ora la care s-a facut trimiterea
- diagnosticul prezumtiv avut in vedere
- tipul de investigatie radiologica
- o descriere sumara a motivelor investigatiei
- medicul care a facut trimiterea

Pentru o investigatie radiologica, pe card sunt retinute urmatoarele informatii:

- medicul practician (nume si prenume) - informatie care va fi preluata automat de pe cardul profesionistului medical sau din baza de date.
- unitatea sanitară – - informatie care va fi preluata automat de pe cardul profesionistului medical sau din baza de date locala sau centrala.
- secţia unde are loc investigatia medicala - informatie care va fi preluata automat de pe cardul profesionistului medical sau din baza de date!
- data la care s-a efectuat investigatia respectiva
- medicul ordonator (nume si prenume) - informatie care va fi preluata automat de pe cardul pacientului sau din baza de date, in baza trimiterii in vederea examinarii.
- unitatea sanitară a medicului ordonator – - informatie care va fi preluata automat de pe cardul pacientului sau din baza de date, in baza trimiterii in vederea examinarii.
- secţia medicului ordonator - - informatie care va fi preluata automat de pe cardul pacientului sau din baza de date, in baza trimiterii in vederea examinarii.

Tot in cadrul fisei de radiatii urmeaza apoi o serie de informatii referitoare la expunere:

- Procedura specifică si Localizarea

Se memoreaza apoi, după caz, una dintre variantele urmatoare:

1) radiodiagnostic sau radiologie intervenţională

In acest caz, expunerea este completata de valoarea masurată a dozei, astfel:

a) Doza la piele (mGy) b) Valoare DAP (Gy x cm. 2) c) CDTI (mGy)

2) medicină nucleară (diagnostic sau terapie)

In acest caz, expunerea este completata de urmatoarele informatii:

a) Activitate administrată (MBq) b) Tip radionuclid

c) Forma chimică d) Doza efectivă estimată (mSv)

3.1) teleradioterapie (de tipul X, cobalt sau linac) sau

3.2) brahiterapie (de tipul manual, HDR, LDR sau radionuclid)

In acest caz, expunerea este completata de urmatoarele informatii:

a) Volum ţintă b) Doza totală cumulată în volumul ţintă (Gy)

4 ELABORAREA SPECIFICAŢIILOR PENTRU INFRASTRUCTURA DE CHEI PUBLICE FOLOSITE PENTRU ASIGURAREA SECURITATII SISTEMULUI SRSPIRIM

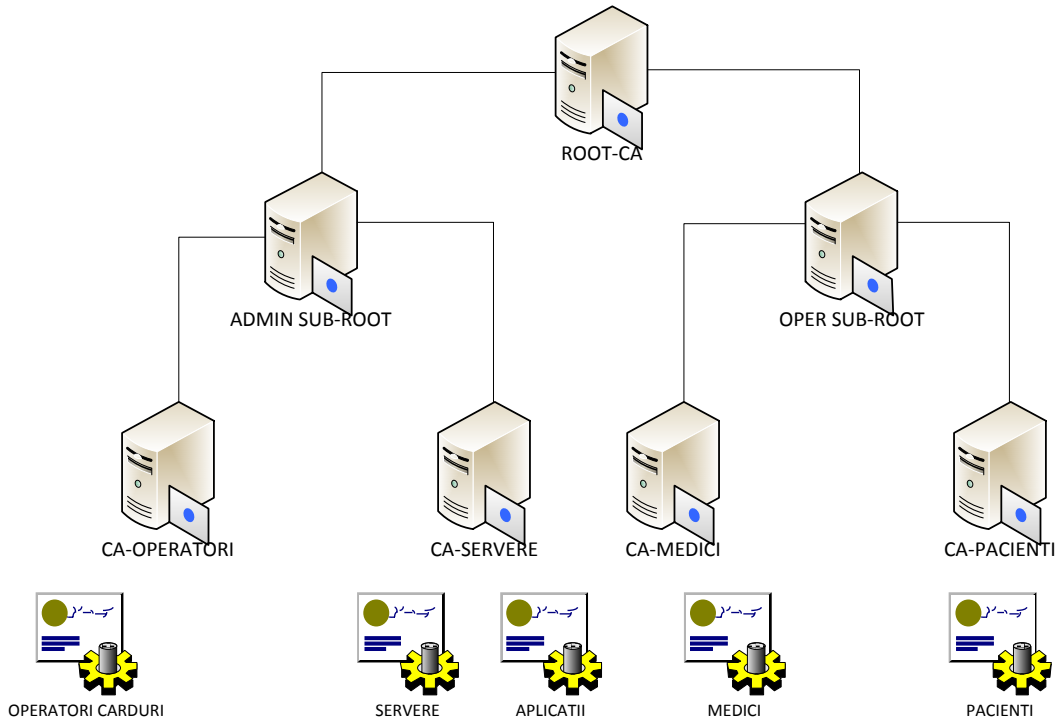
4.1 ARHITECTURA GENERALA A INFRASTRUCTURII CU CHEI PUBLICE

In cadrul celei de-a doua activitati din etapa II a proiectului SRSPIRIM, care da titlul acestui capitol al raportului au fost elaborate specificatiile privind infrastructura de chei publice. O astfel de infrastructura naţionala pentru emiterea, gestiunea si utilizarea cardurilor electronice radiologice integrează cel puţin doua infrastructuri PKI independente, si anume:

- infrastructura PKI destinata emiterii certificatelor digitale necesare zonei de administrare a sistemului. Aceasta infrastructura PKI va include cel puţin următoarele 2 autoritati de certificare:
  - ✓ autoritatea de certificare care emite certificate pentru operatorii si administratorii întregii infrastructuri informatice a sistemului
  - ✓ autoritatea de certificare care emite certificate SSL pentru serverele de aplicaţii si aplicaţiile client care necesita autentificare mutuala.

- infrastructura PKI destinata emiterii certificatelor digitale pentru cardurile radiologice. Aceasta infrastructura va conţine 2 autoritati de certificare:
  - ✓ autoritatea de certificare care emite certificate pentru cardurile de pacient
  - ✓ autoritatea de certificare care emite certificate pentru cardurile profesionale.

In cadrul proiectului SRSPIRIM a fost proiectata in acesta etapa o singura infrastructura PKI care va îngloba cele 2 infrastructuri PKI amintite anterior, si anume PKI-ul administrativ (ADMIN SUB-ROOT) si PKI-ul operaţional (OPER SUB-ROOT).



FIGURĂ 4-1 INFRASTRUCTURA CU CHEI PUBLICE (PKI) A SISTEMULUI INFORMATIC SRSPIRIM

SRSPIRIM ROOT CA, este autoritatea de certificare rădăcina a infrastructurii PKI SRSPIRIM. ROOT CA va dispune de o pereche de chei RSA de lungime 2048 biţi, iar certificatul corespunzător va fi auto-semnat; acest certificat digital auto-semnat este utilizat pentru semnarea certificatelor digitale ale celor doua autorităţi de certificare subordonate.

ADMIN SUB-ROOT simulează autoritatea de certificare rădăcina a infrastructurii PKI administrative. ADMIN SUB-ROOT poseda o pereche de chei RSA pe 2048 biţi si semnează certificatele digitale ale autorităţilor de certificare subordonate necesare administrării cardurilor utilizatorilor (SRSPIRIM CMS), a serverelor si aplicatiilor din sistem.

OPER SUB-ROOT simulează autoritatea de certificare rădăcina a infrastructurii PKI operaţionale. OPER SUB-ROOT poseda o pereche de chei RSA pe 2048 biţi si semnează certificatele digitale ale autorităţilor de certificare subordonate care se ocupa de:

- certificate de autentificare si de semnare pentru cardurile medicilor
- certificate de autentificare si de semnare pentru cardurile pacientilor

Specificatiile pentru infrastructura cu chei publice sunt împărţite după cum urmează:

- specificatii pentru autoritatea de certificare rădăcina (ROOT CA)
- specificatii pentru autoritatile de certificare intermediare denumite Subroot/SubCA
- specificatii pentru autoritatea de înregistrare denumita RA (Registration Authority)
- specificatii pentru autoritatea de validare on-line a stării certificatelor
- specificatii pentru autoritatea de marcare temporala

4.2 AUTORITATEA DE CERTIFICARE RĂDĂCINA - ROOT CA

Autoritatea de certificare rădăcina este punctul de încredere al unei PKI. O autoritate de certificare rădăcina este compusă din elemente hardware, software şi din personalul care le utilizează. O autoritate de certificare rădăcina are un nume şi o pereche de chei. O autoritate de certificare rădăcina trebuie sa îndeplinească următoarele cerinţe:

1) Interfaţa aplicaţiei trebuie sa permită o autentificare sigura, utilizând certificate digitale si dispozitive criptografice hardware pentru stocarea si utilizarea cheilor criptografice.

2) Logica aplicaţiei trebuie sa se regăsească in totalitate la nivel de server. Se utilizeaza minimul de funcţionalităţi necesare pentru a rula pe client.

3) Operatorii aplicaţiei utilizeaza un browser web pentru operarea aplicaţiei.

4) Sistemul utilizeaza biblioteci criptografice certificate FIPS 140-2 sau CC EAL4. (RYCOMBE, 2013)

5) Certificatele operatorilor sunt emise pe o ierarhie care sa nu fie subordonata ierarhiei Autorităţii de Certificare Rădăcina.

6) Autoritatea funcţioneaza obligatoriu cu chei minim RSA de lungime minima de 1024 de biţi.

7) Dispozitivele criptografice hardware trebuie sa ofere suport pentru standardele PKCS#11, PKCS#15 si Microsoft Crypto API. (CRYPTSOFT, 2013)

8) Dispozitivele criptografice hardware trebuie sa ofere posibilitatea de recuperare a contextului criptografic pe un alt dispozitiv in caz de defectare sau dezastru natural.

9) Toate operaţiunile criptografice se vor realiza cu procesoarele criptografice ale dispozitivelor hardware iar cheia privata nu trebuie sa părăsească niciodată dispozitivul hardware.

10) Dispozitivele vor fi prevăzute cu mecanisme de protecţie la atac fizic. Fie accesul nu se va putea efectua fără distrugerea dispozitivului, fie conţinutul se va reseta automat in cazul accesului neautorizat.

11) Autoritatea de certificare rădăcina va respecta o politica de certificare conform căreia va accepta cereri si va emite certificate digitale.

12) Politica de certificare poate fi actualizata periodic. Daca actualizarea politicii introduce incompatibilităţi cu versiunea precedenta, autoritatea de certificare rădăcina trebuie sa permită revocarea certificatelor emise sub respectiva politica si re-emiterea acestora.

13) In cazul in care actualizarea politicii de certificare face imposibila re-emiterea certificatelor autorităţilor subordonate, atunci acestea vor fi emise abia in momentul in care cererile de certificat vor fi conforme cu politica de certificare a autorităţii rădăcina.

14) Autoritatea de certificare ofera funcţionalităţi de revocare a certificatelor emise utilizând mecanisme de tip CRL (Certificates Revocation List).

15) Autoritatea de certificare rădăcina poate oferi funcţionalităţi de tip delta CRL. Aceste funcţionalităţi nu sunt obligatorii, mecanismul neffind foarte des utilizat.

16) Autoritatea de certificare rădăcina ofera funcţionalităţi de roll-over. La un interval definit in politica de certificare, certificatul auto-semnat va fi re-emis pentru continuarea operaţiunilor fără întreruperi.

17) Pentru roll-over, cross-certificarea intre autoritatea veche si autoritatea noua va fi utilizata.

18) Autoritatea de certificare rădăcina ofera funcţionalităţi de import a cererilor de emitere a unui certificat digital.

19) Autoritatea de certificare rădăcina ofera funcţionalităţi de export a certificatului si CRL-ului pentru import in autorităţile de certificare subordonate.

20) Cererile de certificare sunt importate doar însoţite de datele responsabililor sistemului autorităţii subordonate si de politica de certificare respectata de autoritatea respectiva.

21) Toate cererile de certificare, certificatele emise si CRL-urile emise vor fi stocate intr-o baza de date proprie autorităţii de certificare rădăcina.

22) Baza de date a autorităţii de certificare rădăcina va fi obligatoriu o baza de date relaţionala.

23) Sistemul autorităţii de certificare rădăcina trebuie sa permită configurarea astfel încât accesul sa fie permis doar pe portul aferent HTTPS.

24) Autoritatea de certificare rădăcina trebuie sa menţină arhive cu informaţii despre toate certificatele emise.

4.3 AUTORITATILE DE CERTIFICARE SUBORDONATE

Autoritatile de certificare subordonate din infrastructura PKI aferenta sistemului SRSPIRIM sunt ADMIN SUB-ROOT si OPER SUB-ROOT, prima dintre ele ocupandu-se de certificatele pentru administrarea cardurilor, aplicatiilor si serverelor iar cea de-a doua de certificatele digitale emise pentru functionalitati de autentificare si semnare, catre medici si pacienti.

Specificatiile pentru aceste autoritati de certificare sunt cele deja descrise pentru autoritatea de de certificare radacina (cu exceptia specificatiei nr. 5) la care se adauga urmatoarele:

1) Autoritatea de certificare subordonata va respecta o politica de certificare conform căreia va accepta emite certificate pentru clase de certificare.

2) Autoritatea de certificare subordonata va respecta politica de certificare a autorităţii de certificare rădăcina care ii va semna certificatul digital.

3) Autoritatea de certificare subordonata va accepta certificate emise de o autoritate de certificare rădăcina pe care o considera de încredere.

4) Autoritatea de certificare subordonata va oferi posibilitatea de import periodic a CRL-ului emis de autoritatea de certificare rădăcina.

5) Autoritatea de certificare subordonata va oferi funcţionalităţi de roll-over. La un interval definit in politica de certificare, cererea autorităţii va fi transmisa pentru emitere la autoritatea ce certificare rădăcina.

6) După recepţionarea certificatului emis, pentru roll-over, cross-certificarea intre autoritatea veche si autoritatea noua va fi utilizata.

7) Certificatele claselor de certificare vor fi de asemenea incluse in procesul de rollover.

8) Autoritatea de certificare subordonata va fi responsabila in totalitate de funcţionalităţile de roll-over ale claselor de certificare.

9) Autoritatea de certificare subordonata va oferi funcţionalităţi de import a cererilor de certificare pentru cross-certificare.

10) Autoritatea de certificare subordonata va oferi funcţionalităţi de emitere a certificatelor de cross-certificare cat si de salvare a cererii de certificat pentru cross-certificare.

11) Certificatele de cross-certificare vor fi acceptate doar pe baza cererii de certificare si a verificării certificatului emitent.

12) Cererile de cross-certificare vor fi importate doar însoţite de datele responsabililor sistemului autorităţii respective si de politica de certificare respectata de autoritatea respectiva.

13) Autoritatea de certificare subordonata va fi responsabila de administrarea politicilor de certificare a claselor de certificate.

14) Autoritatea de certificare subordonata va oferi funcţionalităţi de export a certificatului si CRL-ului pentru importul in autorităţi cross-certificate.

15) Toate cererile de cross-certificare, certificatele si CRL-urile emise vor fi stocate intr-o baza de date relationala, proprie autorităţii de certificare subordonate.

4.4 AUTORITATEA DE ÎNREGISTRARE

Autoritatea de certificare reprezintă punctul de interacţiune cu utilizatorii al unei PKI. O autoritate de înregistrare este compusă din elemente hardware, software şi din personalul care le utilizează. O autoritate de înregistrare trebuie sa îndeplinească următoarele cerinţe:

1) Interfaţa aplicaţiei trebuie sa permită o autentificare sigura, utilizând dispozitive criptografice hardware si certificate digitale.

2) Logica aplicaţiei trebuie sa se regăsească in totalitate la nivel de server. Se va utiliza minimul de funcţionalităţi necesare pentru a rula pe client.

3) Operatorii aplicaţiei vor utiliza un browser web pentru operarea aplicaţiei.

4) Sistemul va utiliza biblioteci criptografice certificate FIPS 140-2 sau CC EAL4. (RYCOMBE, 2013)

5) Certificatele operatorilor trebuie sa fie emise pe o ierarhie care sa nu fie subordonata ierarhiei Autorităţii de Certificare.

6) Autoritate va funcţiona obligatoriu cu chei minim RSA si de lungime minima de 1024 de biţi.

7) Autoritatea de înregistrare va oferi funcţionalităţi de înregistrare a utilizatorilor in sistem.

8) Autoritatea de înregistrare va oferi funcţionalităţi de avertizare a utilizatorilor asupra certificatelor care urmează sa expire.

9) Avertismentele se vor emite obligatoriu de minim doua ori înainte de expirarea certificatelor. Intervalele de timp pentru avertismente vor fi configurabile.

10) Autoritatea de înregistrare va emite avertismente privind expirarea certificatelor si CRL-urilor autorităţilor către administratori, la intervale stabilite.

11) Autoritatea de înregistrare va oferi funcţionalităţi de actualizare automata a repository-ului autorităţii la intervale prestabilite.

12) Autoritatea de înregistrare va oferi funcţionalităţi de replicare cu o autoritate de certificare utilizând mijloace offline.

13) Autoritatea de înregistrare va oferi funcţionalităţi de monitorizare si avertizare asupra proceselor specifice care se deruleaza in cadrul autorităţii de înregistrare.

14) Autoritatea de înregistrare va permite înregistrarea utilizatorilor utilizând orice fel de dispozitiv criptografic hardware care respecta standardul PKCS#11. (CRYPTSOFT, 2013)

15) Autoritatea de înregistrare va implementa un mecanism prin care un utilizator sa nu-si poată crea singur o cerere de certificat care sa ajungă la aprobare.

16) Toate cererile vor fi stocate intr-o baza de date relationala proprie autorităţii de înregistrare.

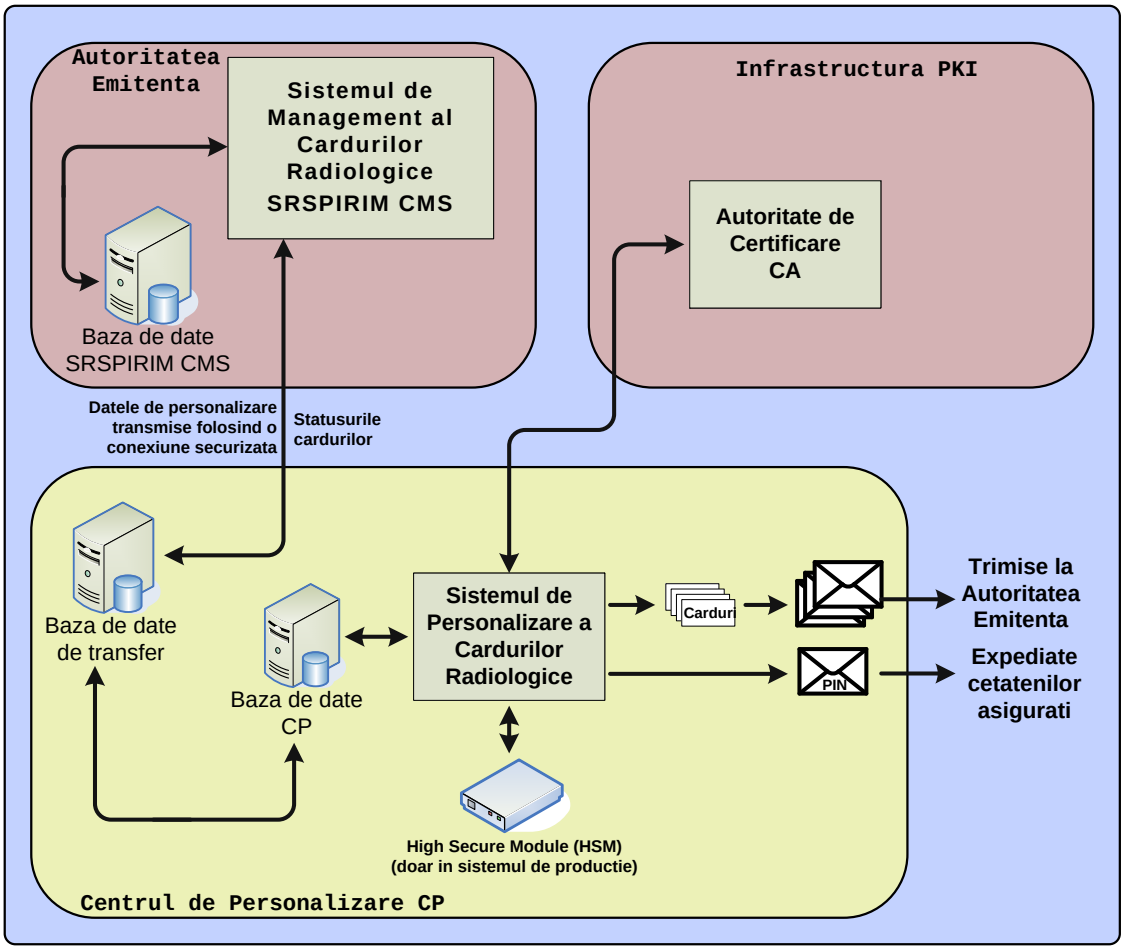
17) Sistemul autorităţii de înregistrare trebuie sa permită configurarea astfel încât accesul sa fie permis doar pe portul aferent HTTPS.

18) Autoritatea de înregistrare trebuie sa ofere posibilitatea de înrolare pentru cereri cu chei atât in containere software (PCKS#12) cat si pe dispozitive hardware.

4.5 FLUXURI DE DATE SI OPERATIONALE SPECIFICE ADMINISTRARII CARDURILOR

Fluxul de personalizare al smart cardurilor conţine o serie de operaţiuni astfel: primirea cererilor se face de la SRSPIRIM CMS (Card Management System), urmând apoi ca aceste cereri sa fie preluate in mod bulk si stocate intr-o baza de date intermediara, de transfer intre CMS si CP (Centru de Personalizare). A doua etapa o constituie pregătirea datelor pentru personalizarea cardurilor radiologice: datele de personalizare sunt stocate intr-o baza de date centrala a sistemului de personalizare. Se efectuează apoi verificarea cererilor primite de la SRSPIRIM CMS din punct de vedere al corectitudinii datelor si formatelor.

In ultima etapa are loc generarea cheilor criptografice si personalizarea vizuala a cardurilor radiologice astfel: se generează perechile de chei RSA direct in cipul cardurilor radiologice si se personalizează optic suprafaţa cardului. Urmează apoi obţinerea certificatelor digitale de autentificare, de semnătura digitala calificata de la autorităţile de certificare dedicate.



FIGURĂ 4-2 ARHITECTURA SISTEMULUI DE PERSONALIZARE A SMART CARDURILOR RADIOLOGICE

In cadrul sistemului informatic SRSPIRIM, vor fi implementate mai multe fluxuri si proceduri, cele mai importante fiind următoarele:

- fluxuri de management al cardurilor electronice, care afectează starea acestora:
  - ✓ emiterea cardurilor electronice radiologice
  - ✓ reînnoirea cardurilor electronice radiologice
  - ✓ pierderea, furtul si defectarea cardurilor electronice

- fluxuri de utilizare (operaţionale) a cardurilor electronice radiologice:
  - ✓ autentificarea la un serviciu cu ajutorul cardului electronic
  - ✓ semnătura calificata aplicata unui document medical de către medic

4.6 AUTORITATEA DE VALIDARE ON-LINE A CERTIFICATELOR DIGITALE

Autoritatea de validare on-line a stării certificatelor digitale emise de către infrastructura PKI a sistemului informatic SRSPIRIM va implementa un server OCSP (Online Certificate Status Protocol) care va răspunde următoarelor cerinţe:

1) Va asigura on-line, prin protocolul OCSP, validarea stării certificatelor.

2) Datele privind starea certificatelor trebuie sa poată fi luate din baza de date a autorităţii de certificare.

3) Datele privind starea de certificare trebuie sa poată fi luate de pe repository de tip LDAP sau HTTP.

4) Importul manual de date de validare (certificate, CRL) trebuie sa fie acceptat de serverul de validare.

5) Serverul de validare trebuie sa permită funcţionalităţi de tip proxy, atât explicit cat si implicit.

6) Pentru sistemul de producţie, cheia de semnare a autorităţii de validare on-line a stării certificatelor trebuie sa poată fi stocată pe un dispozitiv hardware compatibil cu standardul FIPS 140 – 2 level 3.

7) Pentru platforma informatica SRSPIRIM, cheia de semnare a autorităţii de validare on-line a stării certificatelor va fi stocata intr-un container software PKCS#12.

4.7 AUTORITATEA DE MARCARE TEMPORALA

Autoritatea de marcare temporala SRSPIRIM trebuie sa îndeplinească următoarele cerinţe:

1) Autoritatea va utiliza dispozitive criptografice hardware pentru stocarea si utilizarea cheilor criptografice pentru semnătura digitala.

2) Autoritatea de marcare va emite mărci temporale conform cu RFC 3161.

3) Trebuie sa suporte obligatoriu protocolul HTTP pentru transport a mărcilor temporale.

4) Cheia de semnare trebuie sa poată fi stocată pe un dispozitiv hardware compatibil cu standardul FIPS 140 – 2 level 3 care sa respecte standardul PKCS#11. Pentru platforma informatica SRSPIRIM, această cheie va fi stocata intr-un container software PKCS#12.

5) Sistemul va oferi funcţionalităţi de inalta disponibilitate (high-availability). Sistemul va oferi suport pentru sisteme de tip cluster.

6) Sistemul va permite citirea timpului atât de la sistemul local cat si de la o sursa externa.

7) Politici de marcare temporale vor putea fi definite in funcţie de necesitaţi. Sistemul va permite schimbarea acestor politici cat si reflectarea acestora in mărcile emise.

8) Sistemul va preciza precizia in mărcile temporale.

9) Sistemul va permite blocarea emiterii mărcilor in cazul in care precizia ceasului nu mai este de încredere.

10) Mărcile temporale vor putea fi salvate pe disc sau intr-o baza de date relaţionala.

5 PROIECTAREA BAZELOR DE DATE CENTRALE ŞI LOCALE

5.1 ARHITECTURA BAZELOR DE DATE

In cadrul proiectului SRSPIRIM a fost dezvoltat un sistem de baze de date complex, care permite stocarea şi gestionarea istoricului de expunere la radiatii a pacientului, prin implementarea unei arhitecturi cu trei nivele distincte de stocare a informatiilor critice pentru pacienti, ceea ce elimină practic posibilitatea de a pierde informaţii şi oferă acces în timp real la informaţii pentru personalul de specialitate. Cele trei niveluri de stocare sunt:

- la nivel central a fost implementata o baza de date pentru stocarea datelelor corespunzatoare investigatiilor prin metode imagistice radiologice la nivel naţional;

- la nivel local sunt create baze de date în fiecare laborator; aceste baze de date isi sincronizeaza continutul corespunzator pacientilor inregistrati cu baza centrală de date;

- la nivelul cardurilor radiologice ale pacientilor, sunt stocate şi gestionate datele corespunzătoare istoriei investigatiilor radiologice efectuate, si anume dozele de radiatii acumulate cu ocazia fiecarei investigatii, tipul investigatiei, momentul acesteia, laboratorul care a efectuat investigatia, cat si doza cumulativa totala calculata la data ultimei investigatii, corespunzatoare pacientului respectiv.

Sistemul SRSPIRIM asigura replicarea informaţiilor stocate în baza de date centrala cu bazele de date locale şi cardurile de pacient pentru a acoperi toate situaţiile posibile, din punct de vedere al disponibilitatii si conectivitatii la bazele de date din cadrul arhitecturii SRSPIRIM.

Practic, in cadrul sistemului SRSPIRIM exista doua tipuri baze de date, si anume :

- baze de date pentru administrarea cardurilor profesionale si medicale aflate la nivelul centrului de personalizare si a SRSPIRIM CMS (Sistem de Management al Cardurilor);

- baze de date pentru gestiunea informaţiilor medicale specifice pacienţilor unde sunt vor fi înregistrate consultaţii, trimiteri, rezultatul investigaţiilor medicale, etc.

Baza de date pentru gestiunea informaţiilor medicale este constituita din tabele ce conţin informaţii despre utilizatorii sistemului, atât pacienţi cat si profesionişti medicali. Pentru a asigura confidenţialitatea informaţiilor, o parte a acestor date pot fi criptate; practic, pentru fiecare pacient se poate cunoaşte întreagul sau istoric medical, nu însa si datele personale ale acestuia (nume, prenume, adresa, CNP, etc.). Informaţiile medicale din baza de date sunt disponibile pentru studii de caz sau aplicaţii din domeniul statisticilor medicale de exemplu, fără a fi compromisa confidenţialitatea pacienţilor.

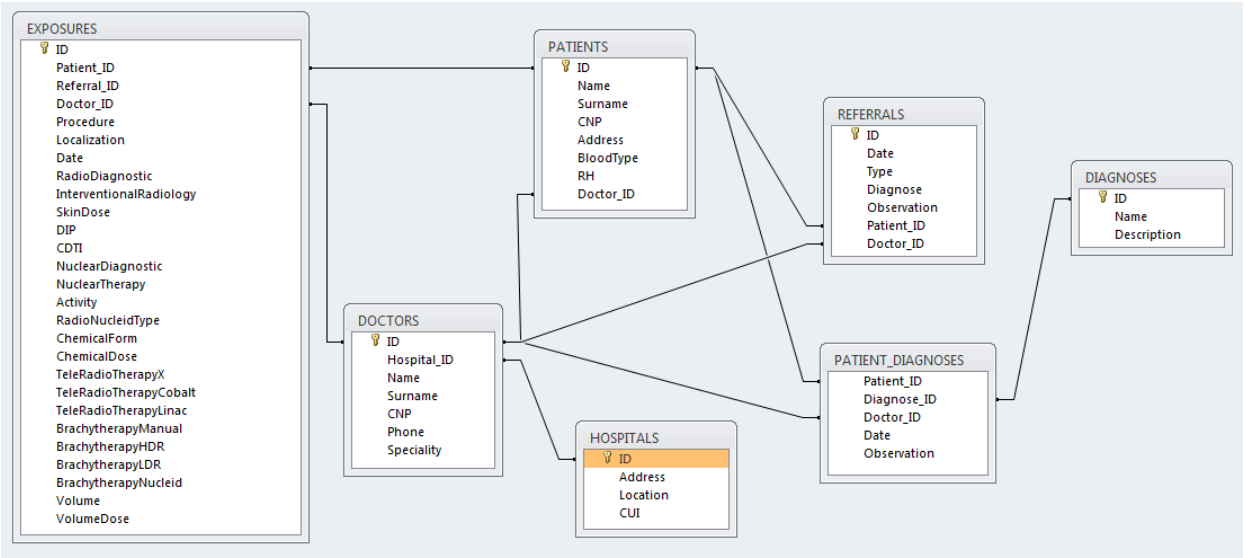
5.2 STRUCTURA BAZEI DE DATE CENTRALE

Cele doua tipuri de baze de date din cadrul sistemului ce contin informatii medicale sunt similare ca structura, ceea ce le diferentiaza fiind cantitatea de date stocata, la nivel local si respectiv central. Baza de date centralizata contine toate informatiile medicale din sistem, la nivel national, putand deveni astfel considerabil mai extinsa decat bazele de date locale.

Din acest motiv, pe termen lung, am prevazut, la anumite intervale de timp, ca o parte din date sa fie arhivate si stocate intr-un mediu securizat. Pe de alta parte, bazele de date locale contin numai o cantitate de date limitata, datorita faptului ca dupa o sincronizare cu baza de date centrala, informatiile locale pot fi sterse sau substituite de alte informatii recente.

Existenta in cadrul sistemului a celor doua tipuri de baze de date a fost conceputa in principal din motive de toleranta la defectari. Daca sistemul ar contine numai o baza de date centralizata, in cazul unei probleme de conectivitate, functiile sistemului nu ar putea fi indeplinite pana cand problema conexiunii nu este rezolvata. Un astfel de scenariu este desigur neacceptabil in contextul unei aplicatii de productie la scara nationala.

Structura bazei de date centrale ce contine informatiile medicale, impreuna cu tabelele din care aceasta este compusa si relatiile existente intre aceste tabele este prezentata in continuare:



FIGURĂ 5-1 4 STRUCTURA BAZEI DE DATE CENTRALE

5.3 STRUCTURA TABELELOR BAZEI DE DATE

Informatiile din cadrul bazei de date pot fi clasificate in 3 parti diferite, astfel : informatii despre pacienti, informatii despre doctori si institutii medicale si informatii referitoare la trimiteri si expuneri radiologice.

Informatiile cu privire la pacienti sunt organizate intr-un numar de tabele, fiecare continand anumite date specifice unui pacient. Tabela PATIENTS contine informatii generale despre pacienti, dintre care cele mai importante sunt prezente in urmatoarele campuri:

- Identificator Unic : ID INTEGER AUTO_INCREMENT,
- Numele : Name VARCHAR(30),
- Prenumele : Surname VARCHAR(30),
- Cod Numeric Personal : CNP VARCHAR(13),
- Adresa : Address VARCHAR(255),
- Grupa sangvina : BloodType INTEGER,
- RH RH INTEGER,
- Medic de Familie Doctor_ID INTEGER.

Identificatorul unic este folosit pentru a regasi pacientul plecand de la aceasta informatie prezenta in celelalte tabele ale bazei de date in care acest identificator apare. Campul ID este prin urmare cheie primara in cadrul acestei tabele si cheia externa in cadrul celorlalte tabele.

Celelalte tabele contin informatii despre bolile sau alergiile pacientilor. In toate aceste tabele, pe langa campurile cu informatii, va exista si semnatura digitala a medicului asupra inregistrarii curente, ca masura de securitate pentru a asigura autenticitatea, integritatea si non-repudierea informatiilor prezente in baza de date.

Tabela DIAGNOSES contine informatiile despre boli, organizate in urmatoarele campuri:

- Identificator Boala : ID INTEGER AUTO_INCREMENT,
- Denumirea bolii : Name VARCHAR(20),
- Descrierea bolii : Description VARCHAR(255)

Tabela PATIENT_DIAGNOSES contine informatii despre bolile pacientilor si este legata prin intermediul celor doua chei secundare de tabelele precedente, PATIENTS si DIAGNOSES:

- Identificator Pacient : Patient_ID INTEGER,
- Identificator Boala : Diagnose_ID INTEGER,
- Identificator Medic : Doctor_ID INTEGER,
- Data Diagnosticarii : Date DATETIME,
- Descriere Sumara : Observation VARCHAR(255)

Informatiile cu privire la medici si institutii medicale sunt stocate in tabelele HOSPITALS si respectiv DOCTORS, ele fiind reprezentate in urmatoarele campuri:

- Identificator Spital : ID INTEGER AUTO_INCREMENT,
- Adresa Spitalului : Address VARCHAR(255),
- Localitatea : Location VARCHAR(20),
- Cod Unic : CUI VARCHAR(30).

- Identificator Medic : ID INTEGER AUTO_INCREMENT,
- Spitalul Aferent : Hospital_ID INTEGER,
- Nume : Name VARCHAR(20),
- Prenumele : Surname VARCHAR(20),
- Cod Numeric Personal : CNP VARCHAR(13),
- Telefon : Phone VARCHAR(13),
- Specialitate : Speciality VARCHAR(20).

Informatiile privitoare la trimiterile si expunerile radiologice sunt continute in tabelele REFERRALS si EXPOSURES. In aceste tabele sunt prezente ca si chei externe, campurile de identificare a pacientului si a medicului care face trimiterea sau cel care efectueaza examinarea radiologica. Informatiile cu privire la trimiteri sunt urmatoarele:

- Identificator Trimitere : ID INTEGER AUTO_INCREMENT,
- Data Trimiterii : Date DATETIME,
- Tipul : Type INTEGER,
- Diagnostic Prezumtiv : Diagnose_ID INTEGER,
- Observatii : Observation VARCHAR(255),
- Identificator Medic : Doctor_ID INTEGER,
- Identificator Pacient : Patient_ID INTEGER.

Informatiile cu privire la examinari sunt continute in urmatoarele campuri:

- Identificator Examinare : ID INTEGER AUTO_INCREMENT,
- Identificator Pacient : Patient_ID INTEGER,
- Identificator Trimitere : Referral_ID INTEGER,
- Identificator Radiolog : Doctor_ID INTEGER,
- Procedura : Procedure VARCHAR(20),
- Localizare : Localization VARCHAR(30),
- Data Examinarii : Date DATETIME,

Datele specifice unei examinari sunt corespunzatoare campurilor din fisa radiologica, astfel :

- RadioDiagnostic TINYINT(1) DEFAULT 0,
- InterventionalRadiology TINYINT(1) DEFAULT 0,
- SkinDose INTEGER,
- DIP INTEGER,
- CDTI INTEGER,
- NuclearDiagnostic TINYINT(1) DEFAULT 0,
- NuclearTherapy TINYINT(1) DEFAULT 0,
- Activity INTEGER,
- RadioNucleidType VARCHAR(20),
- ChemicalForm VARCHAR(20),
- ChemicalDose INTEGER,
- TeleRadioTherapyX TINYINT(1) DEFAULT 0,
- TeleRadioTherapyCobalt TINYINT(1) DEFAULT 0,
- TeleRadioTherapyLinac TINYINT(1) DEFAULT 0,
- BrachytherapyManual TINYINT(1) DEFAULT 0,
- BrachytherapyHDR TINYINT(1) DEFAULT 0,
- BrachytherapyLDR TINYINT(1) DEFAULT 0,
- BrachytherapyNucleid VARCHAR(20),
- Volume INTEGER,
- VolumeDose INTEGER

La aceste campuri se adauga un camp de calcul a dozei de radiatii absorbite, exprimate in mSV, corespunzatoare unei examinari. De asemenea, exista o procedura stocata la nivelul bazei de date centralizate care permite calculul dozei de radiatii cumulate de catre un pacient care poate fi calculata in orice moment si tine cont de precedentele expuneri.

Obiectivul general al activitatii 2.4 legata de proiectarea bazelor de date la care au participat coordonatorul si partenerul P1 este astfel realizat.

6 PROIECTAREA APLICAŢIILOR SOFTWARE ON-CARD

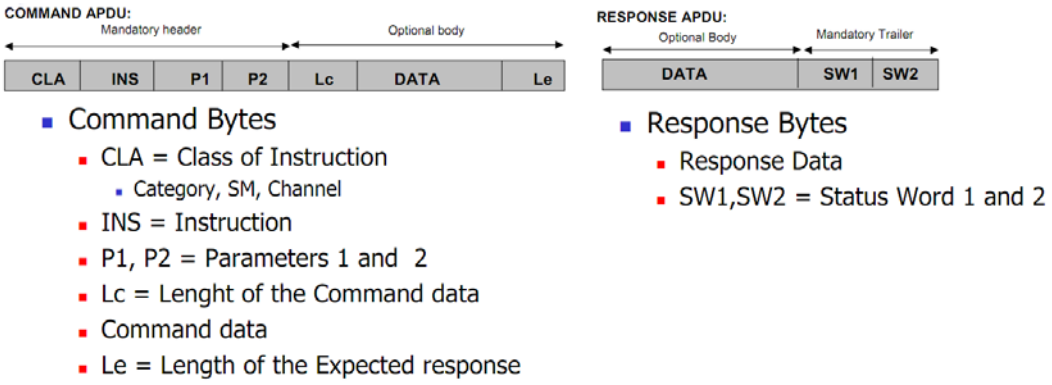
6.1 PRINCIPIUL DE COMUNICATIE CU APPLETII JAVACARD

Appletii prezenti pe un smart card nu efectueaza nici o operatie pana cand nu se trimite o comanda de la o aplicatie off-card pentru executia unei sarcini. Altfel spus, un applet nu ruleaza independent, ci el trebuie sa fie comandat de catre o aplicatie off-card. Aceste doua tipuri de aplicatii (on-card si off-card) sunt conectate printr-un model de comunicatie de tip Master-Slave.

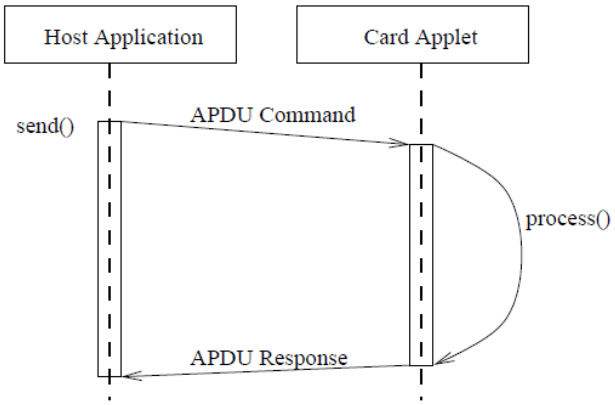
Practic, aplicatiile off-card reprezinta un grup de programe care ofera o interfata de comunicatie cu smart cardul folosind un dispozitiv intermediar sub forma unui cititor de carduri. Cele doua tipuri de aplicatii sunt combinate in cadrul unui sistem care gestioneaza comunicatia cu smart cardul, modulele off-card furnizand si o interfata grafica pentru vizualizarea si gestiunea datelor.

Vom prezenta in continuare modalitatea in care doua aplicatii, prima on-card si a doua off-card pot comunica intre ele. Protocolul care face posibila aceasta comunicatie intre aplicatia de nivel inalt (off-card) si appletul JavaCard poarta numele de APDU (Application Protocol Data Unit).

Protocolul APDU este descris in standardul ISO 7816-4 standard. (WIKIPEDIA, 2013) In acest protocol este folosit modelul de comunicatie comanda-raspuns: aplicatia de nivel inalt trimite o comanda APDU catre smart card iar smart cardul trimite inapoi un raspuns de tip APDU.



FIGURĂ 6-1 STRUCTURA UNEI COMENZI SI A UNUI RASPUNS APDU



FIGURĂ 6-2 SINTEZA PROTOCOLULUI APDU

Ultimii doi octeti ai unui raspuns APDU semnaleaza daca respectiva comanda trimisa catre smart card a fost executata cu succes sau a intervenit o eroare in executia acesteia. Prin urmare, daca ultimii doi octeti ai raspunsului APDU formeaza valoarea 0x9000, executia s-a derulat normal. Orice alta valoare pentru acesti doi acteti semnifica un cod de avertizare sau de eroare.

In modelul de programare master-slave, aplicatia de nivel inalt joaca rolul activ al masterului, trimitand in mod repetat comenzi APDU catre smart card si asteptand apoi raspunsurile acestuia, ca si cum ar apela direct o serie de metode puse la dispozitie de smart card.

Aplicatiile care ruleaza pe smart cardurile JavaCard se numesc appleti. Odata ce un applet este instalat si selectat de pe smart card, el asteapta comenzi de la aplicatia master, jucand astfel rolul pasiv . Dupa primirea unei comenzi APDU, masina virtuala JavaCard livreaza comanda respectiva catre applet invocand metoda de procesare specifica appletului.

Astfel, appletul incepe apoi executia unei metode ca si cum aceasta ar fi apelata direct de catre aplicatia de nivel inalt. In urma apelului metodei din applet, se proceseaza comanda primita si se creeaza un raspuns APDU. In momentul in care appletul preda controlul masinii virtuale JavaCard, raspunsul APDU este trimis catre aplicatia de pe computer.

Datele de pe card pot fi citite si scrise prin intermediul appletilor care rezida pe smart card. Acesti appleti sunt scrisi intr-un limbaj de programare derivat din Java, adaptat pentru memoria si resursele de calcul specifice unui smart card.

Fiecare applet are propriul sau identificator numit AID(Applet ID) si un identificator al pachetului din care face parte. Aceste doua numere identifica in mod unic appletul pe un smart card. Pentru a comunica cu orice applet de pe smart card, acesta trebuie mai intai sa fie selectat. Selectia unui applet se realizeaza printr-o comanda specifica ai carei parametri sunt AID si PID.

6.2 STANDARDE SPECIFICE SMART CARDURILOR

In acest subcapitol, sunt prezentate smart cardurile utilizate in cadrul proiectului SRSPIRIM impreuna cu caracteristicile acestora din punct de vedere al memoriei si resurselor de calcul. De asemenea, sunt descrise etapele care tin de proiectarea appletilor, metoda ce tine de autentificarea la carduri precum si rolurile din cadrul sistemului si drepturile aferente acestora.

Smart cardurile folosite in cadrul acestui proiect sunt de tipul ID ONE Cosmo V7.0.1-N 80K DUAL OBNO30, fabricate de catre firma Oberthur Technologies. Smart cardurile prezinta o interfata duala, din punct de vedere al contactului, insa pentru scopul proiectului a fost retinuta numai interfata de contact.

Cardul ID-One Cosmo V7.0.1-N 80K DUAL OBNO30 prezinta urmatoarele caracteristici:

- compatibilitate cu limbajul si masina virtuala Java Card 2.2.2 (ORACLE, 2007)
- compatibilitate cu standardul GlobalPlatform 2.1.1 (GLOBAL PLATFORM, 2013)
- posibilitatea de autentificare biometrica
- chei RSA de criptare asimetrica de lungime de la 512 pana la 2048 biti
- chei AES de criptare simetrica de lungime de la 128 pana la 256 biti
- functii de hashing SHA-1

Detaliile legate de memoria smart cardului utilizat in cadrul acestui proiect sunt prezentate in tabelul urmator. Smart cardul are 80 KBytes de memorie persistenta EEPROM care poata fi utilizata pentru stocarea datelor. Aproximativ 40 KBytes din aceasta memorie sunt folositi de catre appletul criptografic care este implicit incarcat pe card de la productia acestuia. Spatiul ocupat de appletul criptografic poate fi utilizat pentru a memora chei RSA si certificate X.509.

Tip	Locatie	Dimensiune (octeti)
Persistent Heap	EEPROM	80K
Transient Heap	RAM	1536
Stack	RAM	254
APDU Buffer	RAM	360
Transaction Buffer	EEPROM	576

FIGURE 1 MEMORIA SMART CARDULUI

ISO/IEC este una dintre organizatiile foarte cunoscute in domeniul standardizarii tehnologiei, inclusiv in ceea ce priveste cardurile de plastic. Principalele standarde folosite pentru smart carduri sunt ISO/IEC 7816, ISO/IEC 14443, ISO/IEC 15693 si ISO/IEC 7501. (WEBSTORE, 2013)

ISO/IEC 7816 este un standard international format din mai 14 parti. Partile 1,2 si 3 din standardul ISO/IEC 7816 sunt specifice smart cardurilor cu contact, si prevad numeroase aspecte ale cardului si interfetelor sale, printre care dimensiunile sale fizice, interfata electrica si protocoalele de comunicatie.

Partile 4, 5, 6, 8, 9, 11, 13 si 15 din standardul ISO/IEC 7816 sunt relevante pentru toate tipurile de smart carduri, cu sau fara contact. In aceste parti se defineste structura logica a cardului (fisierele si structurile de date), diverse comenzi folosite de interfata de programare pentru o utilizare simpla, managementul aplicatiilor de pe card, verificarea biometrica, serviciile criptografice si chestiunile legate de numele aplicatiilor.

Partea 10 din standardul ISO/IEC 7816 este specifica cardurilor cu memorie folosite in aplicatii precum cartelele de telefon preplatite. Partea 7 a standardului ISO/IEC 7816 defineste o abordare bazata pe o baza de date relationala securizata pentru smart carduri care foloseste interfetele SQL (SCQL).

ISO/IEC 14443 este un standard international care defineste interfetele pentru un card de proximitate fara contact, incluzand care opereaza pe frecventa radio (RF), interfata electrica, si partea legata de comunicatii si de protocoalele anti-coliziune. Cardurile compatibile cu standardul ISO/IEC 14443 opereaza la o frecventa de 13.56 MHz si au o raza operationala de pana la 10 centimetri (3.94 inches). ISO/IEC 14443 este cel mai important standard pentru smart cardurile fara contact ce sunt folosite pentru aplicatii financiare sau de controlul accesului. Acest standard este de asemenea folosit in domeniul pasapoartelor electronice.

Standardul ISO/IEC 15693 descrie problematica cardurile de vecinatate. Mai precis, in acest standard se stabilesc caracteristicile fizice, puterea pe frecventa radio si interfata de semnal, precum si protocoalele de transmisie si de protectie impotriva coliziunilor pentru cardurile de vecinatate care opereaza la o distanta de maxim 1 metru.

6.3 MECANISME DE COMUNICATIE CU SMART CARDURILE

Singura modalitate de comunicare cu smart cardurile o constituie folosirea comenzilor APDU. Un APDU este o multime de octeti care poate fi trimisa catre card pentru executia unei anumite comenzi, sau poate fi primita de la smart card pentru a indica modul de procesare al comenzii si rezultatul returnat. Exista doua tipuri de APDU: comenzi APDU si rspunsuri APDU.

Command APDU						
Header (required)				Body (optional)		
CLA	INS	P1	P2	Lc	Data Field	Le

FIGURĂ 6-3 STRUCTURA UNEI COMENZI APDU

O comanda APDU este un APDU care este trimis de catre cititorul de carduri catre smart card pentru executia unei comenzi in interiorul cardului. Structura unei comenzi APDU contine obligatoriu 4 octeti in antet (CLA INS P1 si P2) si doi octeti optionali in corpul APDU (LC si LE), pe langa campul Data.

CLA indica clasa comenzii; datorita specificatiilor din standardul ISO/IEC 7816-3, valoarea 'FF' este invalida. Bitul 8 din CLA permite distingerea intre clase predefinite (valoarea 0) si clase proprietare (valoarea 1). Valorile 000x xxxx si 01xx xxxx sunt specificate in continuare. Valorile 001x xxxx sunt rezervate pentru utilizari viitoare. Restul bitilor depind de contextul aplicatiei.

INS indica smart cardului comanda care trebuie procesata. Datorita specificatiilor din standardul ISO/IEC 7816-3, valorile '6X' si '9X' sunt considerate invalide. In cadrul claselor predefinite, orice cod INS valid care nu este definit de catre ISO/IEC 7816 este rezervat pentru utilizari viitoare de catre ISO/IEC.

P1 si P2 sunt parametri ai comenzii ce trebuie executate. Daca respectiva comanda nu necesita parametri, acestia sunt trimisi ambii cu valoarea 0x00. Campul LC este optional si codeaza numarul de octeti care urmeaza dupa antet. In cazul in care comanda nu contine nici un fel de date,nu e nevoie sa se specifice acest camp.

Campul Data reprezinta sirul de octeti trimisi catre smart card in vederea procesarii (ex. scrierii acestora pe card). Lungimea maxima permisa pentru acest camp este de 255 octeti. Octetul LE este de asemenea optional si codeaza numarul maxim de octeti pentru raspunsul comenzii. In cazul in care comanda nu necesita date ca raspuns, campul LE poate fi omis.

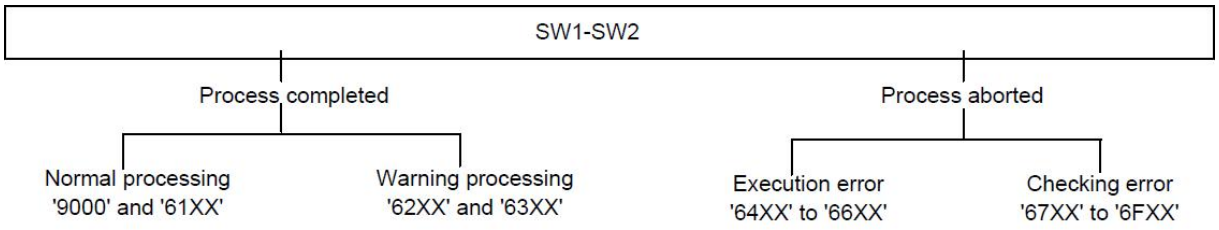
Raspunsul APDU este un APDU care este trimis de catre smart card catre cititorul de card si apoi catre aplicatia de pe computer pentru o informare asupra modului de executie al comenzii. Structura unui raspuns APDU contine obligatoriu doi octeti ce formeaza asa numitul response trailer, precum si un camp optional pentru date.

Response APDU		
Body (optional)	Trailer (required)	
Data Field	SW1	SW2

FIGURĂ 6-4 STRUCTURA UNUI RASPUNS APDU

Campul de date este folosit pentru a returna informatii catre aplicatia de pe computer. In cazul in care comanda anterioara nu necesita date ca raspuns, campul de date nu va fi prezent iar raspunsul APDU va contine numai cei doi octeti obligatorii, SW1-SW2 care indica starea de procesare a comenzii.

Datorita specificatiilor ISO/IEC 7816-3, orice valoare diferita de '6XXX' si '9XXX' este invalida; orice valoare de genul '60XX' este de asemenea invalida. Valorile '61XX', '62XX', '63XX', '64XX', '65XX', '66XX', '68XX', '69XX', '6AXX' si '6CXX' sunt predefinite. Conform standardului ISO/IEC 7816-3, valorile '67XX', '6BXX', '6DXX', '6EXX', '6FXX' si '9XXX' sunt proprietare, cu exceptia valorilor '6700', '6B00', '6D00', '6E00', '6F00' si '9000' care sunt predefinite.



FIGURĂ 6-5 SCHEMA STRUCTURALA CU VALORILE SW1-SW2

Figura urmatoare prezinta toate valorile predefinite pentru cuplul SW1-SW2, impreuna cu semnificatia acestora. Orice valoare predefinita pentru SW1-SW2 care nu este definita in ISO/IEC 7816 este rezervata pentru utilizari viitoare.

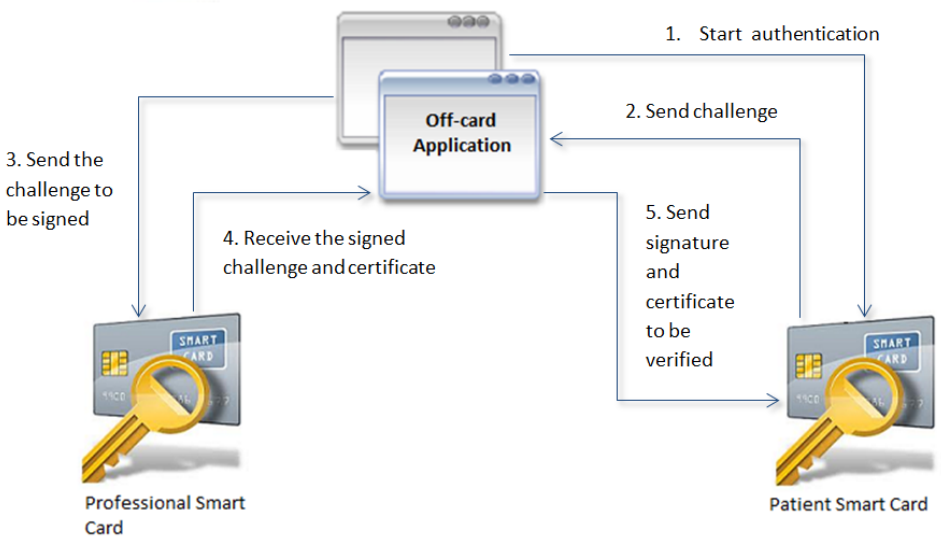
Daca procesarea este abandonata cu o valore pentru SW1 intre '64' si '6F', atunci campul de date din raspunsul APDU este absent. Daca SW1 este setat la valoarea '63' sau '65', atunci starea memoriei non-volatile a fost schimbata. Daca SW1 este setat la valorea '6X', cu exceptia valorilor '63' si '65', atunci starea memoriei non-volatile nu a fost schimbata. Mai multe informatii legate de structura comenzilor si a raspunsurilor APDU pot fi gasite in standardul ISO/IEC 7816-4.

	SW1-SW2	Meaning
Normal processing	'9000' '61XX'	No further qualification SW2 encodes the number of data bytes still available (see text below)
Warning processing	'62XX' '63XX'	State of non-volatile memory is unchanged (further qualification in SW2) State of non-volatile memory has changed (further qualification in SW2)
Execution error	'64XX' '65XX' '66XX'	State of non-volatile memory is unchanged (further qualification in SW2) State of non-volatile memory has changed (further qualification in SW2) Security-related issues
Checking error	'6700' '68XX' '69XX' '6AXX' '6B00' '6CXX' '6D00' '6E00' '6F00'	Wrong length; no further indication Functions in CLA not supported (further qualification in SW2) Command not allowed (further qualification in SW2) Wrong parameters P1-P2 (further qualification in SW2) Wrong parameters P1-P2 Wrong L_e field; SW2 encodes the exact number of available data bytes (see text below) Instruction code not supported or invalid Class not supported No precise diagnosis

FIGURĂ 6-6 SEMNIFICATIA GENERALA A VALORILOR PREDEFINITE PENTRU SW1-SW2

6.4 AUTENTIFICAREA CARDULUI PROFESIONIST SI DREPURI DE ACCES

Aplicatia off-card dialogheaza cu cele doua tipuri de carduri simultan, astfel incat primul pas il constituie autentificarea mutuala intre acestea. Aceasta autentificare este realizata pe baza unui protocol de tip challenge-response intre cele doua carduri.



FIGURĂ 6-7 SCHEMA DE AUTENTIFICARE INTRE CARDURI

Figura anterioara prezinta modalitatea in care medicul profesionist se autentifica catre pacient. Autentificarea incepe cu trimiterea unui challenge de pe cardul pacientului. Acest challenge va fi trimis prin intermediul aplicatiei off-card (appletului Java) catre appletul criptografic instalat pe cardul profesional.

Provocarea va fi semnata pe cardul profesionistului cu cheia privata de autentificare a acestuia. Provocarea astfel semnata, impreuna cu certificatul digital corespuznator sunt trimise inapoi catre cardul pacientului; daca certificatul este valid si se verifica semnatura, autentificarea in sensul pacient-profesionist este asigurata. Nici o forma de acces la datele cardului nu este posibila pana cand nu se termina procesul de autentificare. Dupa aceasta, applicatia ofera accesul profesionistului la continutul cardului.

De asemenea, aplicatia off-card trebuie sa se asigure de autenticitatea cardului pacientului sau a medicului profesionist. Acest lucru se va face prin intermediul unui protocol similar de challenge-response, dinspre aplicatie catre cardul pacientului sau al medicului.

Pentru siguranta datelor medicale stocate pe cardul pacientului, in cadrul proiectului au fost definite o serie de reguli privind accesul la datele cardului, in functie de rolul celui care dialogheaza cu smart cardul pacientului. Aceste drepuri de acces sunt redate in tabelul urmator.

Astfel, operatorul de carduri are dreptul doar de a citi si de a scrie datele administrative ale pacientului (nume, prenume, adresa, CNP, etc.). Aceste date trebuiesc insotite obligatoriu de semnatura operatorului de carduri, pentru a preveni utilizarea frauduloasa a unui card.

Medicul de familie are dreptul de a citi si scrie datele medicale generale ale pacientului. El are de asemenea dreptul de a adauga un diagnostic sau o trimitere in vederea unei investigatii radiologice. Medicul are dreptul de a citi toate bolile sau trimiterile scrise anterior, precum si rezultatele investigatiilor radiologice efectuate de pacient.

Medicul radiolog are dreptul de a citi toate datele medicale ale pacientului fara insa a avea dreptul de a le modifica. El are doar dreptul de a inscrie o fisa radiologica pe cardul pacientului.

DATE/ROL	OPERATOR CARDURI	MEDIC DE FAMILIE	MEDIC RADIOLOG
DATE ADMINISTRATIVE	READ/ WRITE	READ	READ
FISA MEDICALA	---	READ/ WRITE	READ
BOLI	---	READ/ WRITE	READ
TRIMITERI	---	READ/ WRITE	READ
INVESTIGATII RADIOLOGICE	---	READ	READ/ WRITE

TABEL 1 ROLURI SI DREPURI DE ACCES LA INFORMATIILE CARDULUI

Din punct de vedere al programarii, datele de pe cardul pacientului sunt organizate in cateva clase, fiecare dintre acestea regrupand o serie de informatii corespunzatoare. Principalele clase din cadrul pachetului software dezvoltat sunt : AdministrativeData, MedicalData, Diagnosis, Referral si Exposure.

Clasa AdministrativeData contine informatiile generale ale pacientului si cuprinde atributele:

- name : byte[20]
- surname : byte[20]
- CNP : byte[13]
- postalAddress : byte[50]
- emailAddress : byte[30]
- emergencyContact : byte[30]
- emergencyPhone : byte[20]
- signature : byte[128]

Clasa MedicalData contine informatiile medicale generale si cuprinde urmatoarele atribute:

- medicalState : byte[20]
- bloodType : short
- RH : short
- allergies : byte[50]
- signature : byte[128]

Clasa Diagnosis contine informatiile referitoare la un anumit diagnostic si cuprinde atributele:

- date : byte[4]
- time : byte[4]
- doctor : byte[50]
- location : byte[20]
- description : byte[255]
- diagnosis : byte[20]
- signature : byte[128]

Clasa Referral contine informatiile referitoare la o trimitere medicala si cuprinde atributele:

- date : byte[4]
- time : byte[4]
- doctor : byte[50]
- description : byte[255]
- presumptiveDiagnosis : byte[20]
- exposureType : byte[20]
- signature : byte[128]

Clasa Exposure contine informatiile referitoare la o investigatie radiologica si cuprinde:

- radiologyDoctor : byte[50]
- radiologicUnity : byte[30]
- date : byte[4]
- time : byte[4]
- familyDoctor : byte[50]
- medicalUnity : byte[30]
- procedure : short
- localization : short
- radioDiagnostic : bool
- interventionalRadiology : bool
- skinDose : integer
- DIP integer
- CDTI integer
- nuclearDiagnostic boolr
- nuclearTherapy bool
- activity integer
- radioNucleidType byte[20]
- chemicalForm byte[20]
- chemicalDose integer
- teleRadioTherapyX bool
- teleRadioTherapyCobalt bool
- teleRadioTherapyLinac bool

- brachytherapyManual bool
- brachytherapyHDR bool
- brachytherapyLDR bool
- brachytherapyNucleid bool
- volume integer
- volumeDose integer

Pentru fiecare dintre campurile claselor descrise anterior exista metode de tip Get si Set pentru a putea citi sau scrie campul respectiv. Bineinteles, in cadrul clasei exista si un atribut role care stocheaza rolul celui care efectueaza operatii cu smart cardul; acest rol este decodificat in urma parsarii certificatului digital de pe cardul profesionistului. Operatiile Set si Get sunt permise numai daca rolul este autorizat sa citeasca sau sa scrie acea informatie.

Alte interfete puse la dispozitie de appletul JavaCard sunt :

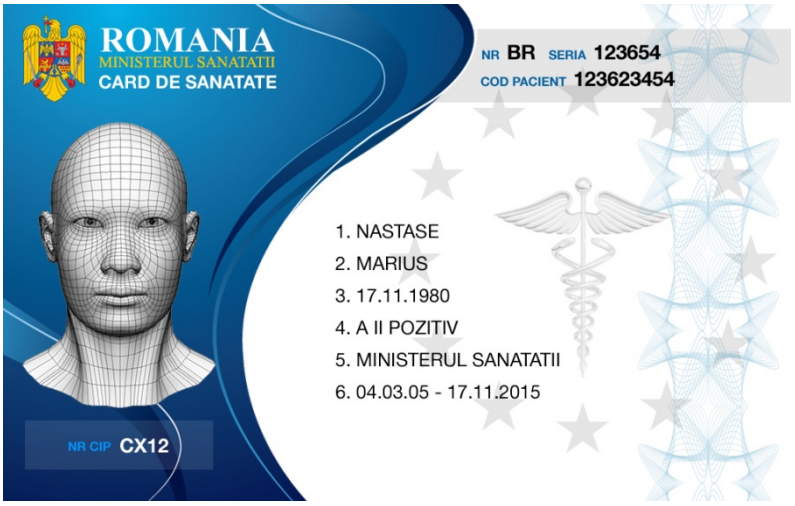
- byte[] getAllDiagnosises() : intoarce toate bolile cu care a fost diagnosticat pacientul
- byte[] getDiagnosises(date t1, date t2) : intoarce toate bolile cu care a fost diagnosticat pacientul intre momentele de timp (date calendaristice) t1 si t2
- byte[] getAllReferral() : intoarce toate trimiterile specifice pacientului
- byte[] getReferrals(date t1, date t2) : intoarce toate trimiterile specifice pacientului prescrise intre momentele de timp (date calendaristice) t1 si t2
- byte[] getAllExposures() : intoarce toate expunerile radiologice efectuate de pacient
- byte[] getExposures(date t1, date t2) : intoarce toate expunerile radiologice efectuate de pacient intre momentele de timp (date calendaristice) t1 si t2
- bool checkSignature(byte[] buffer, byte[] signature, byte[] publicKey) : verifica semnatura asupra unui buffer de date folosind cheia publica a semnatarului.
- bool checkSignature(byte[] buffer, byte[] signature, byte[] certificate) : verifica semnatura asupra unui buffer de date folosind certificatul digital al semnatarului.
- byte[] getRandomChallenge() : returneaza un challenge generat in maniera aleatoare pentru proceduri de autentificare; ultimul challenge generat va fi stocat si pe card.

6.5 PERSONALIZAREA EXTERIOARA A CARDURILOR

Prima etapă prin care un smart card trebuie să treacă este faza de personalizare. Această personalizare se realizează in incinta unui Centru de Personalizare care preia smart cardul de la producatori, şi personalizează atât suprafata exterioara a cardului, precum şi datele din interiorul cardului. In aceasta sectiune vom prezenta personalizarea exterioara a cardurilor.

Personalizarea exterioara a cardului este procesul care transformă plasticul alb al smart cardului într-o carte care are imprimate pe fata informaţii cu caracter personal privitoare la cel care va deveni proprietarul cardului. Suprafata numita fata cardului este aceea care contine contactele electrice. Informatiile personale prezente pe suprafata exterioara a cardului sunt:

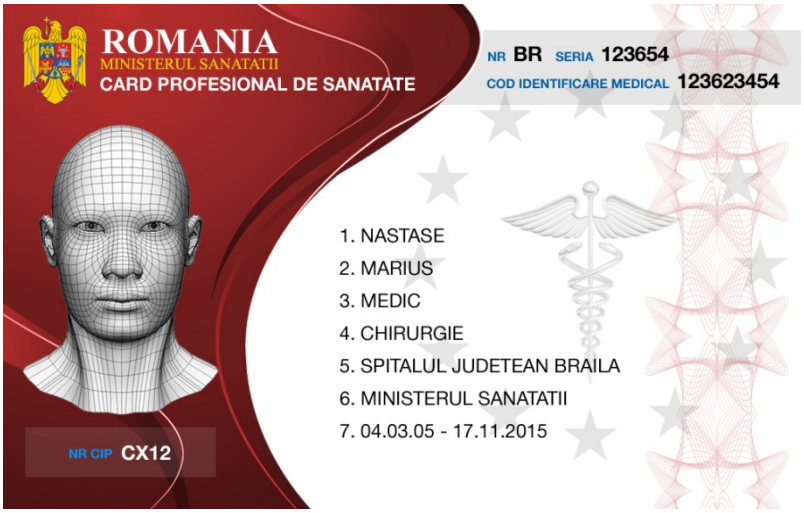
- Numele intreg
- O fotografie recenta
- Codul numeric personale (CNP)
- Grupa sangvina
- Data de expirare a cardului
- Numarul de identificare al cardului



FIGURĂ 6-8 FATA EXTERIOARA A UNUI CARD DE PACIENT

FIGURĂ 6-9 LEGENDA INSCRISA PE SPATELE SMARD CARDULUI DE PACIENT

FIGURĂ 6-10 LEGENDA INSCRISA PE SPATELE SMARD CARDULUI DE MEDIC DE FAMILIE



FIGURĂ 6-11 FATA EXTERIOARA A UNUI CARD DE MEDIC DE FAMILIE

In acest capitol a fost prezentata proiectarea claselor si interfetelor pe care le ofera aplicatiile de pe smart card catre aplicatiile off-card, precum si designul cardurilor indeplinindu-se astfel obiectivele activitatii 2.5, activitate la care au participat coordonatorul si partenerul P1.

7 DEZVOLTAREA APLICAŢIILOR SOFTWARE OFF-CARD

7.1 DESCRIEREA APLICATIILOR SOFTWARE OFF-CARD

Acest capitol prezinta detaliile de dezvoltare proiectare specifice aplicatiilor off-card. Aplicatiile principale off-card dezvoltate in cadrul sistemului SRSPIRIM sunt urmatoarele:

1) Aplicatia de Inregistrare a Persoanelor (Pacienti si Medici)
2) Aplicatia de Personalizare a Cardurilor
3) Aplicatia Medicala SRSPIRIM care cuprinde urmatoarele module:
   3.1) Modul de Vizualizare a Informatiilor Personale Generale
   3.2) Modul de Completare a unei Trimiteri Medicale
   3.3) Modul de Completare a unei Fise de Radiatii
   3.4) Modul de Raportare catre Ministerul Sanatatii

La setul de aplicatii enumerate mai sus se adauga desigur toate serverele si aplicatiile specifice infrastructurii PKI folosite in cadrul proiectului, precum si aplicatiile de management pentru cardurile si certificatele digitale emise.

APLICATIA DE INREGISTRARE A PERSOANELOR

Aplicaţia permite inregistrarea in baza de date a persoanelor pentru care se vor emite carduri. In cadrul sistemului exista urmatoarele tipuri/roluri de persoane: MEDIC DE FAMILIE, MEDIC RADIOLOG, PACIENT si OPERATOR CARDURI.

Rolul OPERATOR CARDURI are urmatoarele functii:

- introduce datele persoanelor in aplicatia de inregistrare
- incarca appletii criptografici pe cardurile persoanelor
- incarca appletii de lucru pe cardurile pacientilor
- personalizeaza datele din interiorul appletului de lucru al cardului de pacient

Aplicatia de inregistrare permite accesul numai operatorilor de carduri, pe baza unui certificat digital semnat de o autoritate special destinata acestor operatori.

Datele specifice unui medic sunt urmatoarele : Nume, Prenume, Specialitate (Medic de Familie / Medic Radiolog), Spital/Policlinica, CNP, Telefon, Rol De Raportare (numai anumiti medici vor putea avea posibilitatea de a crea rapoarte catre Ministerul Sanatatii din modulul de raportare).

Datele personale specifice unui pacient sunt urmatoarele : Nume, Prenume, CNP, Adresa, Persoana de Contact (in caz de urgenta), Telefon de Contact (in caz de urgenta), Medic De Familie (Cod Unic al Medicului sau CNP). La aceste informatii se vor adauga informatiile medicale generale : Grupa Sangvina, RH, Alergii precum si alte date medicale care vor fi completate ulterior de catre medicul de familie.

Aplicatia de inregistrare permite urmatoarele :

- Introducerea manuala a datelor pentru fiecare persoana.
- Preluarea in mod bulk a datelor dintr-un fisier in format CSV.
- Exportarea datelor intr-un format XML pentru imprimarea cardurilor.
- Informatiile care vor fi afisate pe suprafata cardului sunt :
  - ✓ PACIENT : Nume, Prenume, CNP, Grupa Sangvina, RH si Telefon de Contact.
  - ✓ MEDIC : Nume, Prenume, Spital, Specialitate, Telefon.

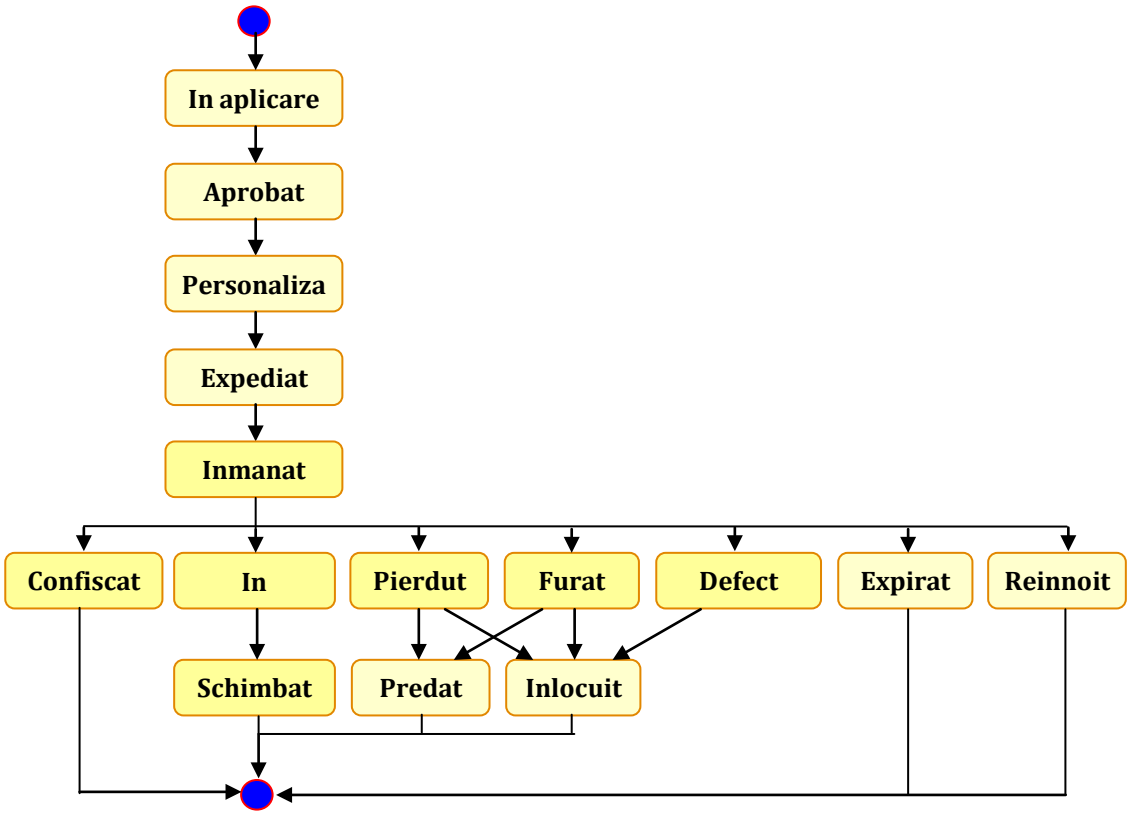
APLICAŢIA DE PERSONALIZARE A CARDURILOR

Aplicaţia permite personalizarea unui smart-card pentru a putea fi folosit ca şi card radiologic de sănătate sau card radiologic profesionist. Astfel, această aplicaţie va fi responsabilă cu următoarele:

- Pentru cardul radiologic profesionist (destinat medicilor):
    - ✓ personalizeaza smart-cardul cu cheile private RSA; exista doua perechi de chei, folosite pentru autentificare şi semnătură digitală. Cheile vor fi generate în interiorul cardului.
    - ✓ obtine certificatele digitale X.509 corespunzătoare celor doua chei de la Autoritatea de Certificare si le stocheaza pe cardul medicului, in appletul criptografic disponibil pe card.
    - ✓ accesul la aceste chei se face prin intermediul unor coduri PIN. Astfel, fiecare din cele două chei este protejată de un cod PIN stabilit la personalizarea cardului. Acest cod poate fi apoi schimbat de către posesorul cardului folosind serviciile sistemului de management al cardurilor;
- Pentru cardul radiologic al pacientului :
    - ✓ personalizeaza smart-cardul cu cheia privata RSA folosita pentru autentificare. Cheia este generata în interiorul cardului;
    - ✓ obtine certificatul digital X.509 corespunzător cheii publice de la Autoritatea de Certificare si il stocheaza in appletul criptografic disponibil pe card;
    - ✓ accesul la aceasta cheie se face prin intermediul unui cod PIN. Acest cod poate fi apoi schimbat de către posesorul cardului folosind serviciile puse la dispoziţie de sistemul de management al cardurilor;
    - ✓ face instalarea appletului de lucru pe smart-card;
    - ✓ personalizeaza datele din interiorul cardului cu datele specifice de identificare ale pacientului preluate din baza de date;

Toate informaţiile din interiorul cardului (cele prezente in appletul de lucru) sunt semnate digital de către Administratorul de Carduri pentru a se asigura autenticitatea şi integritatea. De asemenea, in cadrul certificatului digital al unei persoane se precizeaza rolul acesteia : MEDIC DE FAMILIE / RADIOLOG / PACIENT. De asemenea, in cadrul certificatului unui medic exista si o o modalitate de specificare a dreptului de raportare pentru medicii radiologi.

In privinta cardurilor, exista un flag in baza de date specifica prin care se cunoaste in permanenta statusul cardului unei persoane. Stărilor cardurilor de protectie radiologica (Citizen Radiation Safety Card) şi diagrama tranziţiilor acestor stări este prezentata in figura urmatoare.



FIGURĂ 7-1 DIAGRAMA STARILOR DE TRANZITIE ALE UNUI CARD

Stările cardurilor de protectie radiologica sunt următoarele:

1) În aplicare : s-a primit o cerere de emitere a unui card şi s-au înregistrat datele în sistemul CMS folosind modulul de enrollment.

2) Aprobat : datele cererii de emitere a unui card au fost validate şi s-a decis emiterea cardului respectiv. Datele de personalizare şi nota de comandă au fost trimise la CP.

3) Personalizat : sistemul CP (Centru de Personalizare) a validat datele de personalizare, a trimis automat mesajul corespunzător sistemului CMS. Mesajul este transmis folosind un canal de comunicaţie SSL, şi notifică sistemul CMS asupra corectitudinii datelor de personalizare. Sistemul CMS actualizează automat starea cardului in baza de date pe baza informaţiilor din acest mesaj. Dacă datele de personalizare sunt valide, cardul este personalizat la CP.

4) Expediat : sistemul CP a expediat cardul la unitatea medicala solicitanta. Sistemul CP trimite un mesaj automat sistemului CMS folosind acelaşi canal de comunicaţie securizat SSL, prin care informează CMS asupra stării fiecărui card din comanda respectivă. CMS actualizează automat starea cardului pe baza informaţiilor din acest mesaj.

5) Inmânat : unitatea medicala a înmânat cardul pacientului care l-a solicitat. Inmanarea se inregistreaza in sistemul CMS.

6) Confiscat : cardul a fost confiscat de catre autoritatile medicale in cazul in care acesta este nu este folosit conform reglementarilor (de ex este folosit de catre alta persoana decat posesorul cardului). Confiscarea se inregistreaza in sistemul CMS.

7) Predat : un card pierdut sau furat dacă este găsit ulterior, este predat unitatii medicale emitente. Cardul este trecut în starea predat.

8) Pierdut : pacientul/medicul a declarat cardul pierdut. Doctorul de familie va actualiza starea cardului în sistemul CMS.

9) Furat : pacientul/medicul a declarat cardul furat. Doctorul de familie va actualiza starea cardului în sistemul CMS.

10) Expirat : perioada de valabilitate a cardului a expirat. Cardul este trecut automat în această stare de către sistemul CMS.

11) Înlocuit : cardul declarat pierdut, furat sau defect a fost înlocuit cu unul nou, cu aceleaşi date, cu excepţia numărului de card.

12) Înnoit : cardul a fost înnoit ca urmare a apropierii datei de expirare.

13) În schimbare : procedura de schimbare a cardului este în curs, ca urmare a schimbării datelor pacientului/medicului. Noul card încă nu a fost înmânat posesorului.

14) Schimbat : cardul a fost schimbat ca urmare a schimbării datelor pacientului/medicului. Noul card a fost înmânat pacientului/medicului.

APLICATIA MEDICALA SRSPIRIM

Aplicatia contine 4 module si anume :

1) Modul de Vizualizare si Editare a Informatiilor Medicale : vizualizarea si editarea unor informatii medicale generale si anume : diagnostice si alergii ale pacientului.

2) Modul de Completare a unei Trimiteri Medicale : vizualizarea trimiterilor anterioare, completarea unei trimiteri noi.

3) Modul de Completare a unei Fise de Radiatii : vizualizarea expunerilor anterioare, completarea unei noi fise radiologice.

4) Modul de Raportare catre Ministerul Sanatatii

Dintre aceste module, primele 3 necesita prezenta cardului pacientului. La aceasta aplicatie vor avea acces urmatoarele roluri:

1) MEDIC DE FAMILIE – pentru Modulul de Vizualizare si Editare a Informatiilor Medicale, Modul de Completare a unei Trimiteri Medicale
2) MEDIC RADIOLOG – Modulul de Vizualizare a Informatiilor Medicale si Modulul de Completare a unei Fise de Radiatii

Daca in plus, MEDICUL RADIOLOG are si dreptul de RAPORTARE, atunci el va putea folosi si Modulul de Raportare. Autentificarea in aplicatie se face pe baza certificatului de autentificare prezent pe cardul profesional al medicului.

Datele medicale pot fi vizualizate de catre orice medic insa numai MEDICUL DE FAMILIE are posibilitatea de a adauga noi diagnostice sau noi alergii. Datele sunt salvate atat in baza de date cat si pe cardul pacientului, alaturi de semnatura medicului. Datele specifice unui diagnostic, a unei alergii, a unei trimiteri sau a fisei radiologice rezulta din schema bazei de date.

Expunerile radiologice medicale pot fi vizualizate de catre orice medic insa numai MEDICUL RADIOLOG are posibilitatea de a adauga o noua expunere (fisa de radiatii). Datele unei expuneri sunt salvate in baza de date cat si pe cardul pacientului, alaturi de semnatura medicului.

Modul de Completare a unei Fise de Radiatii cuprinde o fereastra in care se introduc urmatoarele informatii:

- Medicul practician (nume si prenume) - informatie preluata automat de pe cardul profesionistului medical sau din baza de date!
- Unitatea sanitară si sectia - informatii preluate automat de pe cardul profesionistului medical sau din baza de date!
- Data
- Medic ordonator (nume si prenume) - informatie preluata automat de pe cardul pacientului (folosind trimiterea medicala) sau din baza de date!
- Unitatea sanitară a medicului ordonator - informatie preluata automat de pe cardul pacientului (folosind trimiterea medicala) sau din baza de date!
- Secţia medicului ordonator - informatie preluata automat de pe cardul pacientului (folosind trimiterea medicala) sau din baza de date!

Urmeaza apoi o serie de informatii referitoare la expunerea pacientului si anume : procedura specifică si localizarea. Se completează apoi, după caz, una dintre variantele urmatoare:

1) radiodiagnostic sau radiologie intervenţională

In acest caz, expunerea este completata de valoarea masurată a dozei, astfel:

b) Doza la piele (mGy) b) Valoare DAP (Gy x cm. 2) c) CDTI (mGy)

2) medicină nucleară (diagnostic sau terapie)

In acest caz, expunerea este completata de urmatoarele informatii:

a) Activitate administrată (MBq) b) Tip radionuclid

c) Forma chimică d) Doza efectivă estimată (mSv)

3.1) teleradioterapie (de tipul X, cobalt sau linac) sau

3.2) brahiterapie (de tipul manual, HDR, LDR sau radionuclid)

In acest caz, expunerea este completata de urmatoarele informatii:

a) Volum ţintă b) Doza totală cumulată în volumul ţintă (Gy)

Modulul genereaza pe baza informatiilor introduse un document (ex. pdf) dupa modelul fisei de expunere la radiatii. Documentul poate fi generat si pentru a fi tiparit, asa cum se intampla in momentul actual.

De asemenea, oricare ar fi tehnica de investigatie folosita se memoreaza intr-un camp si doza de radiatii specifica acelei expuneri. Informatiile specifice unei asemenea inregistrari/expuneri sunt in totalitate semnate de catre medicul radiolog. Un element important este si transformarea dozelor in mSv, in fct de explorare, pe baza unor formule matematice, astfel incat sa existe aceeasi unitate de masura, independent de natura investigatiei.

MODUL DE ANALIZA SI RAPORTARE PRIVIND EXPUNERILE MEDICALE

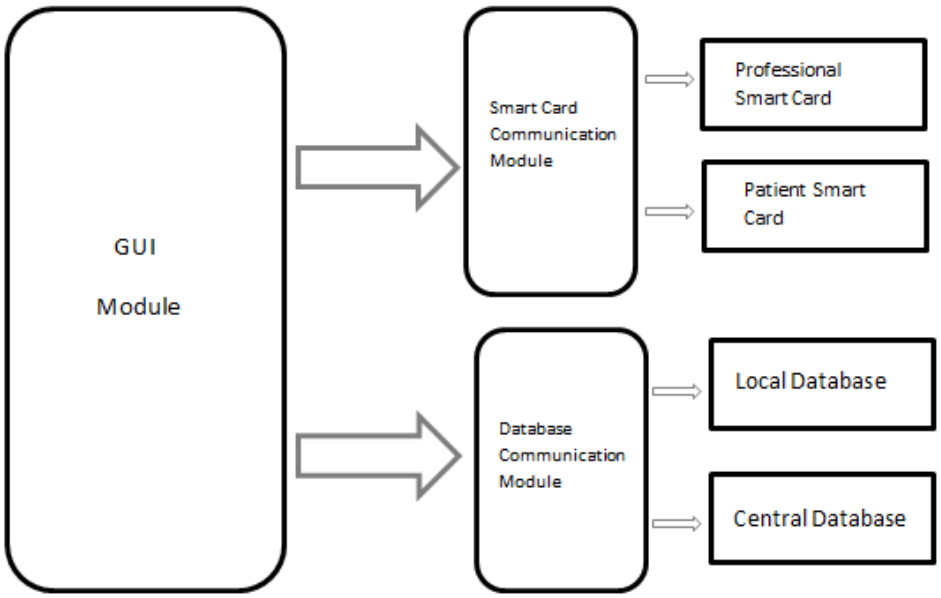
Modulul de analiza si interpretare va include anumite statistici pornind de la datele specifice expunerilor radiologice stocate in baza de date sau pe smart cardul pacientului. De exemplu, pentru un anumit pacient, se poate obtine:

- adunarea dozelor acumulate pe un anumit interval de timp
- reprezentarea grafica in functie de timp si cantitatea de radiatii
- care este cantitatea de radiatii a unui pacient pana la doza tinta

Pentru toti pacientii sau pentru o categorie de pacienti, folosind anumite criterii bazate de exemplu pe varsta sau pe sex, se vor genera o serie de rapoarte dupa modelul celor prevazute de in Ordinul Ministrului Sanatatii privitor la rapoartele care trebuiesc trimise catre minister.

MODULELE SOFTWARE ALE APLICATIEI

Aplicatiile off-card sunt impartite in general in trei module care implementeaza partile principale ale specificatiilor tehnice: comunicarea cu smart cardul, interfata grafica pentru utilizator si comunicarea cu bazele de date. Urmatoarele sectiuni se axeaza pe prezentarea acestor module si modalitatea lor de implementare.



FIGURĂ 7-2COMPONENTELE UNEI APLICATII OFF-CARD

7.2 COMUNICATIA APLICATIILOR OFF-CARD CU SMART CARDURILE

Aplicatiile off-card lucreaza adesea cu doua tipuri de carduri simultan; pentru o functionare corecta, cele doua carduri, cardul pacientului si cardul profesional trebuie sa fie fizic conectate la calculatorul gazda prin intermediul a doua cititoare distincte.

Toate aplicatiile off-card dezvoltate sunt de tip web, folosind limbajul de programare PHP pentru generarea paginilor dinamice si pentru interactiunea cu bazele de date si limbajul JavaScript pentru partea rulata de client, in scopul de a realiza corectarea imediata a datelor eronate introduse in formularele web. Aplicatiile web contin si o serie de appleti dezvoltati in limbajul de programare Java, decizia privind limbajul de programare utilizat fiind puternic legata de suportul oferit pentru comunicatiile cu smart cardul si de independenta de platforma:

- Suportul smart card - Java oferă o bibliotecă de comunicare cu smart cardurile şi anume javax.smartcardio. Aceasta defineşte un API Java pentru comunicarea cu cartele inteligente folosind standardul ISO/IEC 7816 (WIKIPEDIA, 2013). Acest lucru permite aplicaţiilor Java de a interacţiona cu aplicaţiile care rulează pe carduri inteligente, pentru a stoca şi regăsi datele de pe card.

- Independenta de platforma - este foarte important ca aplicatia să fie capabila să ruleze pe mai multe platforme având în vedere scopul proiectului iar limbajul de programare Java ofera această caracteristică. Biblioteca pentru smart carduri din Java ofera suport pentru operaţiunile cu carduri multiple, cum ar fi conectarea la un card inteligent, trimiterea şi primirea de APDU-uri si inchiderea conexiunii existente. Aceasta prevede, de asemenea, sprijin si pentru detectarea cititoarelor de carduri inteligente conectate la calculator şi ofera posibilitatea de a alege între mai multe cititoare de carduri, lucru absolut necesar având în vedere ca aplicatia se ocupă de două smart carduri simultan.

Java ofera, de asemenea, suport pentru accesarea appletului criptografic, care este în mod implicit încărcat pe card. Acest applet criptografic este folosit pentru crearea de semnături digitale şi verificarea certificatelor digitale prin intermediul API-ului Java disponibil. Platforma Java defineste un set de interfeţe de programare pentru efectuarea operaţiunilor criptografice; aceste interfeţe sunt cunoscute sub numele de Java Cryptographic Architecture (JCA) şi Java Cryptographic Extension (JCE).

Standardul pentru tokenuri criptografice, PKCS # 11 (RSA LABORATORIES, 2013), este un produs de securitate RSA şi defineşte interfeţele de programare pentru tokenuri criptografice, cum ar fi cardurile inteligente. PKCS # 11 face parte din din familia de standarde destinate criptografiei bazată pe chei publice (PKCS), publicat de RSA Laboratories, si defineşte un API independent de platforma si de tokenuri criptografice, cum ar fi modulele de hardware securizate (HSM) şi cardurile inteligente.

Pentru a facilita integrarea tokenurilor native PKCS#11 in platforma Java, un nou furnizor de servicii criptografice, Sun PKCS#11, a fost introdus in versiunea J2SE 5.0. Acest furnizor permite aplicatiilor existente scrise in API-urile JCA si JCE sa aiba acces la tokenurile compatibile PKCS#11. Singura cerinta este configurarea corecta a furnizorului in platforma Java Runtime.

Configurarea furnizorului necesita incarcarea unei librarii care este o implementare compatibila cu PKCS #11 pentru accesul appletului criptografic de pe smart card. Aceasta librarie este furnizata de catre fabricantul cardului sub forma unui fisier .dll.

Rolul principal al modului de comunicatie cu smart cardul este gestiunea conexiunilor si a comunicatiilor cu smart cardurile prezente. Functionalitatile sale includ:

- Detectarea numarului de cititoare de carduri prezente
- Stabilirea unei conexiuni cu cele doua tipuri de carduri
- Selectarea appletului corepunzator de pe fiecare card
- Codarea si decodarea datelor pentru a corespunde formatului impus de applet
- Trimiterea comenzilor APDU si receptia raspunsurilor APDU
- Comunicarea cu appletul criptografic
- Inchiderea conexiunii cu cele doua carduri
- Gestiunea erorilor de comunicatie cu cardurile

Acest modul foloseste API-ul Java de comunicare cu smart carduri javax.smartcardio pentru a oferi functionalitatile enumerate anterior. Acesta interactioneaza direct cu smart cardurile. Comunicatia dintre cele doua carduri, simultan conectate la aplicatie, trebuie sa fie implementata prin intermediul aplicatiei gazda, datorita faptului ca nu exista nici o metoda de comunicare directa intre cele doua smart carduri.

PACHETUL APDU

Acest pachet contine clasele Java care se ocupa de gestiunea conexiunii si a comunicatiilor cu cardul. Ea include trei clase si anume : APDUControls, MyCardChannel si Crypto.

APDUControls – aceasta clasa gestioneaza operatiile legate de card. Ea include metode de conectare la smart card, de selectare a appletului corespunzator, de trimitere si receptie APDU, de verificare a erorilor precum si de codarea si decodarea datelor pentru ca acestea sa fie trimise intr-un format acceptat de smart card.

MyCardChannel – aceasta clasa este o incapsulare a clasei javax.smartcardio.CardChannel. Ea extinde clasa de baza mentionata astfel incat trimiterea si receptia pachetelor APDUs sa poata fi logate si verificate in vederea detectarii eventualelor erori. Clasa este folosita pentru operarii de logare si debug.

Crypto – aceasta clasa gestioneaza comunicarea cu appletul criptografic care este incarcat implicit pe card. Ea furnizeaza functii pentru trimiterea de date in vederea semnarii lor digitale, receptiei semnaturii si a certificatului digital corespondent. Clasa nu comunica cu appletul de pe smart card prin APDU-uri, ci foloseste o biblioteca wrapper PKCS#11 (RSA LABORATORIES, 2013) oferita de Oberthur. Kitul de dezvoltare de la Oberthur ofera o interfata de nivel inalt pentru schimbarea mesajelor cu appletul criptografic.

7.3 INTERFATA APLICATIEI OFF-CARD CU BAZELE DE DATE

Modulul de acces la baze de date ofera aplicatiei posibilitatea de conectare la bazele de date locale sau la baza de date centrala a sistemului. Acest modul foloseste interfata oferita de limbajul PHP pentru lucrul cu bazele de date, fie ca este vorba despre MySQL, dBase, Oracle, DB2, PostgreSQL, Sybase, InterBase, SQLServer sau ODBC.

In cadrul proiectului a fost folosit API-ul din PHP pentru conectarea la bazele de date MySQL, principalele functii folosite din cadrul API-ului fiind urmatoarele:

- **mysql_connect** : această funcţie realizează conectarea la un server MySQL.
- **mysql_close** : această funcţie primeşte ca parametru un identificator de acces la o conexiune spre un server MySQL şi realizează închiderea acesteia.
- **mysql_ping** : funcţia *mysql_ping* verifică dacă serverul MySQL a închis conexiunea.
- **mysql_create_db** : funcţie utilizată pentru a crea o bază de date pe serverul MySQL.
- **mysql_drop_db** : funcţie folosită pentru a şterge o bază de date.
- **mysql_select_db :** aceasta funcţie setează baza de date pentru o conexiune către un server MySQL pentru interogările care vor urma.
- **mysql_query** : această funcţie se foloseşte pentru a interoga o anumită bază de date.
- **mysql_real_escape_string** : funcţie utilă în momentul în care se doreşte introducerea în cadrul unei interogări a unui şir de caractere care nu poate fi interpretat corect de către server-ul MySQL şi realizează transformarea şirului într-unul care poate fi interpretat.
- **mysql_free_result** : această funcţie se foloseşte pentru a elibera memoria alocată stocării unui rezultat primit în urma unei interogări de la un server MySQL.
- **mysql_num_rows** : această funcţie returnează numărul de înregistrări conţinute de către un rezultat primit de la serverul MySQL.
- **mysql_info :** această funcţie returnează un şir de caractere care conţine informaţii referitoare la ultima interogare a unei baze de date în urma căreia nu s-a primit nici un rezultat, cum este cazul funcţiilor *INSERT* sau *UPDATE*.
- **mysql_affected_rows :** in cazul interogărilor în urma cărora nu se obţine nici un rezultat, se poate folosi funcţia *mysql_affected_rows* pentru a verifica numărul de înregistrări care au fost actualizate la ultima interogare a bazei de date, pentru celelalte tipuri de interogări putându-se folosi funcţia *mysql_num_rows*.
- **mysql_fetch_array**: această funcţie transformă o înregistrare dintr-un rezultat primit de la server-ul MySQL într-o listă.
- **mysql_stat** : această funcţie returnează un şir de caractere care reprezintă statusul server-ului MySQL şi are un singur parametru care reprezintă identificatorul de acces la o conexiune către un server MySQL.

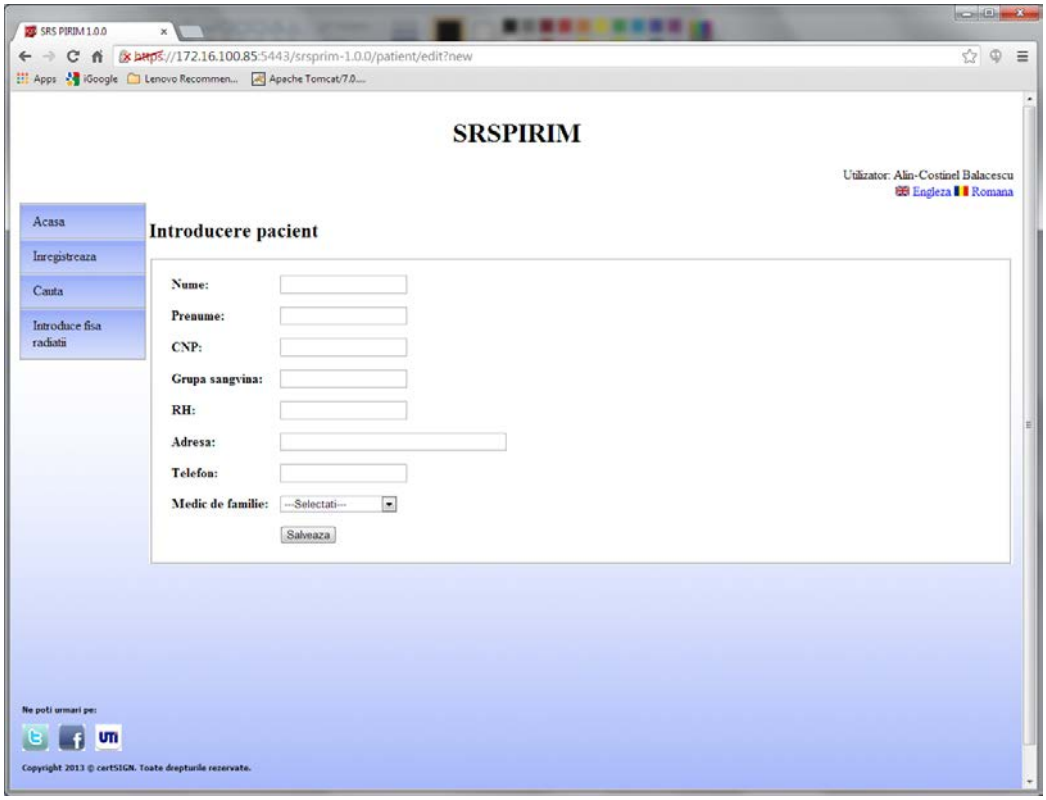
API-ul descris anterior este folosit pentru a stabili conexiuni si a face interogari si actualizari in bazele de date. Aplicatia foloseste bazele de date locale si realizeaza periodic actualizari si sincronziari cu baza de date centrala pentru cele mai recente inregistrari.

7.4 INTERFATA GRAFICA CU UTILIZATORUL FINAL

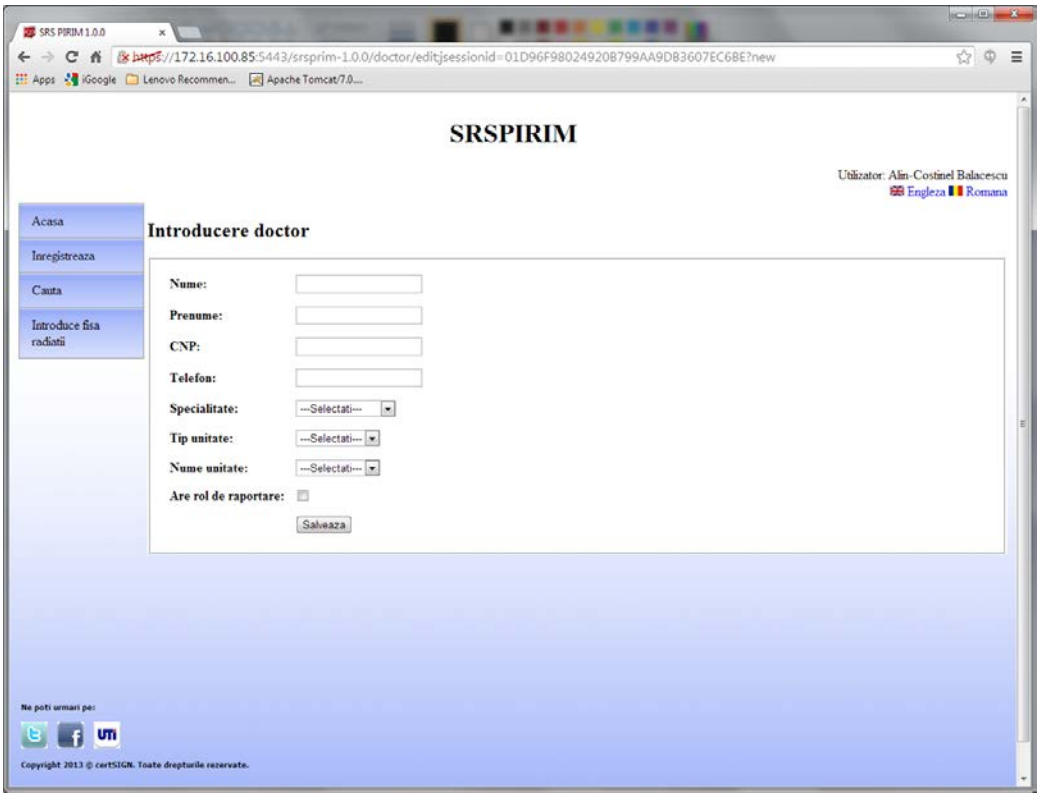
Modulul GUI este cel care permite interactiunea cu utilizatorul final. Acest modul este compus din cateva pagini web dinamice, fiecare dintre acestea oferind informatii diferite utilizatorului. Acest modul foloseste modulul de comunicatie cu smart cardul pentru a obtine informatiile de pe card si pentru a stoca date si utilizeaza de asemenea si modulul de comunicatie cu bazele de date pentru diverse interogari in tabelele bazelor de date.

Dezvoltarea interfetei grafice este foarte simpla si intuitiva si ofera multiple indicatii despre modul in care trebuie utilizata. Optiunile oferite de interfata utilizatorului sunt diferite depinzand de rolul acestuia in cadrul sistemului. In functie de certificatul X509 aflat pe cardul profesional, rolul poate fi determinat in cadrul primei faze de autentificare dintre carduri.

Considerand faptul ca aplicatia isi propune sa inlocuiasca formularele existente in prezent pentru diverse documente medicale disponibile acum in format hartie, interfata cu utilizatorul ofera formate de prezentare similare cu acestea astfel incat ele sa fie foarte usor de completat. In continuare sunt prezentate o serie de capturi ecran ale aplicatiei de inregistrare a medicilor si a pacientilor, impreuna cu formularul specific de completare a unei fise radiologice.

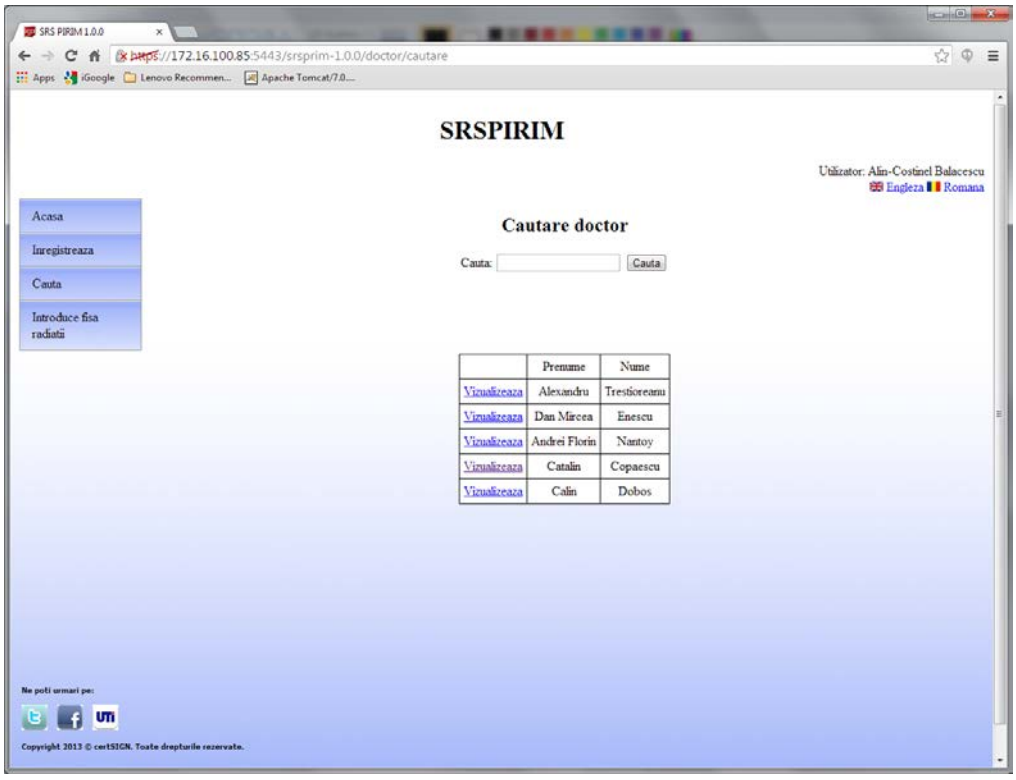


FIGURĂ 7-3 FORMULAR DE INREGISTRARE IN SISTEM A UNUI PACIENT

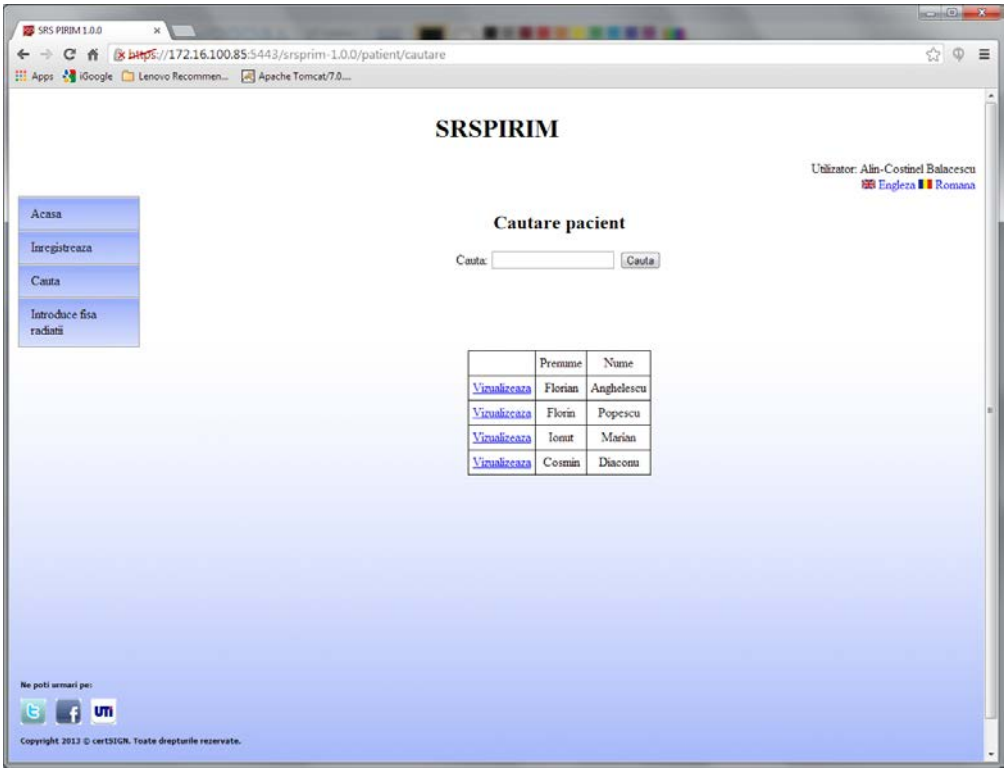


FIGURĂ 7-4 FORMULAR DE INREGISTRARE IN SISTEM A UNUI DOCTOR

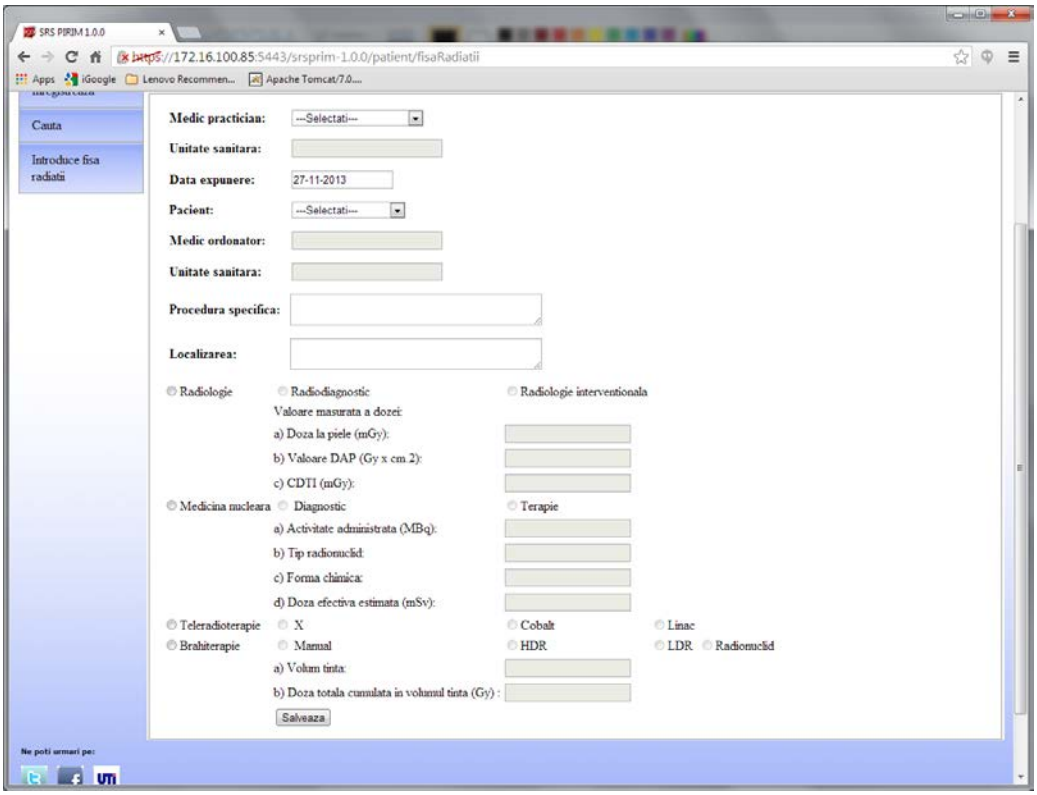
Aceste documente includ inregistrarea medicala, trimiterea si fisa de radiatii electronica. In orice caz, exista anumite campuri din formularele actuale care au fost inlocuite sau redenumite in formularele electronice oferite de modulul GUI cum ar fi de exemplu semnatura olografa a medicului si stampila unitatii medicale. Aceste campuri nu mai sunt necesare, ele fiind inlocuite de mecanismele de autentificare si integritate obtinute prin folosirea semnaturilor digitale.



FIGURĂ 7-5 FORMULAR DE CAUTARE A UNUI DOCTOR IN BAZA DE DATE



FIGURĂ 7-6 FORMULAR DE CAUTARE A UNUI PACIENT IN BAZA DE DATE



FIGURĂ 7-7 FORMULAR DE COMPLETARE A UNEI FISE RADIOLOGICE

Modulul de interfata grafica trebuie sa trateze toate erorile, atat cele aparute in comunicatia cu cardurile cat si cele legate de operatiunile cu bazele de date. Aceste erori includ exceptii legate de retragerea smart cardului din cititorul de carduri propagate de appletul de pe card, comenzi incorecte date spre executie cardului sau exceptii legate de interogarile efectuate in baza de date. Aceste erori trebuiesc tratate corespunzator astfel incat utilziatorul sa cunoasca faptul ca o comanda nu a putut fi realizata cu succes si care au fost cauzele erorii.

PACHETUL UTILITIES

Acest pachet contine clasele PHP utilizate pentru prezentarea informatiilor utilizatorului si cele folosite pentru crearea formularelor. Pachetul contine o serie de constante printre care sunt toate codurile posibile INS, referinta statica catre smart card si conectorul catre baza de date, valorile pentru AID si PID pentru ambele carduri si appleti precum si alte variabile statice.

Pachetul mai include si formularele pentru vizualizarea si completarea trimiterilor, a informatiilor personale, a fisei radiologice si a diagnosticelor. Acelasi formular este folosit indiferent daca datele sunt citite de pe smart card sau din baza de date locala sau centrala.

Paginile aferente acestui modul ofera o interfata intuitiva utilizatorului pentru accesul la smart card respectand intocmai cerintele de sistem. Interfata are rolul de a prezenta informatiile intr-o forma usor de interpretat; astfel, toti identificatorii care sunt cititi de pe card trebuiesc schimbati cu denumirile lor reale ce sunt citite din baza de date.

O capabilitate importanta a interfetei grafice este precompletarea a anumitor campuri non-variabile in formularele destinate trimiterii medicale sau fisei radiologice. Acestea includ data curenta, informatii despre pacient preluate de pe smart card, informatii despre doctor, denumirea si alte detalii administrative legate de institutia medicala.

Interfata mai ofera de asemenea si ajutor permanent; in dreptul fiecarui camp, autocompletat sau nu, exista posibilitatea de a afisa o descriere a semnificatiei acelui camp. Aceste campuri reprezinta termeni medicali care sunt organizati logic in nomenclatoare, si sunt afisati sub forma de obiecte combo-box, pentru a alege una dintre valorile posibile.

GESTIUNEA ERORILOR

Tipul de erori care pot aparea in timpul rularii aplicatiei sunt legate in principal de conexiunea si comunicarea cu smart cardul precum si cu baza de date. Utilizatorul este anuntat in momentul aparitiei acestor erori. Cauza erorii o constituie adesea o utilizare incorecta a cardului cum ar fi retragerea lui fara efectuarea operatiei de deconectare sau in timpul unei anumite operatii. Interfata afiseaza utilizatorului un mesaj corespunzator in cazul unei operatii care nu a putut fi executata cu succes. In majoritatea cazurilor, comanda poate fi reluata prin trimiterea parametrilor corecti care sunt sugerati in fereastra de dialog a mesajului de eroare.

7.5 MECANISME DE SECURITATE

Mecanismele de securitate joaca un rol foarte important in cadrul aplicatiei, datorita faptului ca folosesc informatii private extrem de importante. Pentru implementarea acestor mecanisme aplicatia foloseste atat appletul dezvoltat in cadrul proiectului cat si appletul criptografic cu care este dotat smart cardul. Acesti doi appleti nu pot comunica intre ei in mod independent iar aplicatia offcard trebuie sa intermedieze aceasta comunicare, actionand ca o punte intre ei.

CHEI SI CERTIFICATE

Din moment ce aplicatia offcard intermediaza comunicarea intre appletul criptografic si appletul SRSPIRIM, mecanismele de securitate trebuiesc implementate folosind criptografia cu chei publice. Aceste chei publice sunt prezente in cadrul certificatelor digitale X509. Ierarhia autoritatilor de certificare care emit aceste certificate a fost prezentata anterior.

Fiecare dintre cele doua smart carduri are doua seturi de certificate digitale si chei dintre care unul este folosit in procesul de autentificare mutuala iar cel de-al doilea pentru generarea semnaturilor digitale. Aceste certificate apartin detinatorului de card si sunt stocate in appletul criptografic. In orice caz, smart cardul stocheaza si alte certificate ale autoritatilor de certificare, pastrate de aceasta data in appletul de pe cardul pacientului.

Considerand ca validarea certificatelor trebuie sa poata fi facuta si offline, certificatele autoritatii de certificare de la nivelul Centrului de Personalizare impreuna cu certificatul auto-semnat al autoritatii radacina (Root CA) sunt pastrate in applet. Aceste doua certificate reprezinta intreg lantul de incredere. Atunci cand un certificat profesional emis de Centrul de Personalizare este trimis catre appletul de pe cardul pacientului, autenticitatea sa poate fi verificata fiindca atat certificatul autoritatii de personalizare cat si cel al autoritatii radacina sunt prezente pe card.

Atunci cand o noua inregistrare este adaugata pe card, semnatura sa digitala impreuna cu certificatul prezent pe celalalt card sunt trimise alaturi de date in vederea validarii semnaturii. Atat semnatura cat si certificatul vor fi stocate pe card alaturi de datele semnate.

APPLETUL CRIPTOGRAFIC

Appletul criptografic numit si appletul crypto este incarcat pe card implicit la momentul achizitiei cardului. El nu este accesat in acelasi mod ca si appletul dezvoltat de echipa proiectului, prin intermediul comenzilor APDU; dimpotriva, fiecare fabricant de smart carduri implementeaza o biblioteca care comunica cu appletul de pe smart card si care ofera o interfata pentru accesul functiilor criptografice.

Aceasta biblioteca vine sub forma unui fisier .dll ce trebuie incarcat de fiecare data cand aplicatia doreste sa utilizeze appletul crypto. Ea contine o implementare proprietara a standardului PKCS#11 pentru acele smart carduri. Functiile oferite se regasesc si in pachetul Java din versiunea JDK6 numit sun.security.pkcs11.wrapper.

Implementarea standardului PKCS#11 a fost furnizata prin kitul de dezvoltare oferit de Oberthur si se numeste OCSCryptoki.dll. Appletul criptografic este prin urmare accesat printr-o biblioteca de nivel mai inalt in comparatie cu appletul dezvoltat in cadrul proiectului.

Interfata PKCS#11 permite functionalitati complexe cum ar fi generarea perechilor de chei, stocarea certificatelor digitale si a cheilor corespondente, alegerea unei anumite chei pentru semnare digitala, returnarea unui certificat digital si a cheii publice corespondente stocate anterior pe appletul cardului. Biblioteca suporta de asemenea diferite tipuri de chei si algoritmi de semnare. Appletul crypto este folosit in toate mecanismele de securitate implementate in cadrul sistemului iar modalitatea in care este utilizat va fi descrisa in continuare.

SEMNATURI DIGITALE

La scrierea oricarei informatii pe card, continutul acesteia este semnat de catre profesionistul medical si trimis catre cardul pacientului impreuna cu certificatul digital corespondent. Acest mecanism permite si verificarea ulterioara a datelor inscrise pe card, oferind astfel si proprietatea de non-repudiere. Semnaturile digitale sunt calculate efectiv de catre appletul criptografic dar functiile de semnare sunt apelate dintr-un wrapper PKCS#11 oferit de Sun.

Pasii necesari pentru a efectua o semnatura digitala asupra unui sir de octeti cuprind deschiderea unei sesiuni, logarea in appletul crypto cu ajutorul unui PIN, cautarea cheii private necesare, initierea si terminarea operatiei de semnare.

```
/**** functie pentru semnarea unor date de catre appletul cryptografic

     * handle
     * @param handle - which of the two cards to use
     * @param data - data to sign
     * @param pin - crypto applet pin
     * @return  - signature
     */
```

```
public static byte[] signPatient(byte[] data, char[] pin, String label){
        long[] keyHandles;
        byte[] signature = null;
        Class pkcs11Class;
        long p11_session = 0;
        PKCS11 pkcs11 = null;
        int slot = 0;
        String aliasCrt;

        try {
               p11_session =
               pkcs11.C_OpenSession(slots[slot],PKCS11Constants.CKF_SERIAL_SESSION,
               null,null);
               Constants.pkcs11Patient.C_Login(Constants.patientSession,PKCS11Constants.CKU_US
               ER, new char[]{'9', '9', '9', '9'});

               CK_ATTRIBUTE[] attributes = new CK_ATTRIBUTE[2];
               attributes[0] = new CK_ATTRIBUTE();
               attributes[0].type = PKCS11Constants.CKA_CLASS;
               attributes[0].pValue = PKCS11Constants.CKO_PRIVATE_KEY;
               attributes[1] = new CK_ATTRIBUTE();
               attributes[1].type = PKCS11Constants.CKA_LABEL;
               attributes[1].pValue = label;
               Constants.pkcs11Patient.C_FindObjectsInit(Constants.patientSession,attributes);
               keyHandles = Constants.pkcs11Patient.C_FindObjects(Constants.patientSession,1);
               if (keyHandles.length == 0) {
                      log.fatal("Nu s-a gasit cheia cu label-ul " + label + " !");
                      return null;
               }
               long signatureKey = keyHandles[0];
               Constants.pkcs11Patient.C_FindObjectsFinal(Constants.patientSession);
               //Initialize the signature
               CK_MECHANISM mechanism = new CK_MECHANISM();
               mechanism.mechanism = PKCS11Constants.CKM_SHA1_RSA_PKCS;
               mechanism.pParameter = null;
               Constants.pkcs11Patient.C_SignInit(Constants.patientSession, mechanism,
               signatureKey);

               //Sign the data
               signature = Constants.pkcs11Patient.C_Sign(Constants.patientSession, data);
               Constants.pkcs11Patient.C_Logout(Constants.patientSession);
               log.info("Signing successful " + keyHandles.length);
        }
       catch (Exception ex) {
               Logger.getLogger(Crypto.class.getName()).log(Level.SEVERE, null, ex);
        }/* finally {
        try {
               pkcs11.C_CloseSession(p11_session);
        } catch (PKCS11Exception ex) {
               Logger.getLogger(Crypto.class.getName()).log(Level.SEVERE, null, ex);
        }*/
        return signature;
}
```

In procesul de cautare al cheii private, sunt folosite anumite atribute care specifica criteriul de cautare al cheii. Atributele sunt un sir de elemente CK_ATTRIBUTE, fiecare element avand doua campuri: tip si valoare. Aceste campuri sunt folosite astfel incat cautarea sa gaseasca cheia potrivita. De exemplu, pentru a gasi cheia privata asociata unei anumite etichete, sirul de atribute se completeaza astfel :

```
CK_ATTRIBUTE[] attributes = new CK_ATTRIBUTE[2];
attributes[0] = new CK_ATTRIBUTE();
attributes[0].type = PKCS11Constants.CKA_CLASS;
attributes[0].pValue = PKCS11Constants.CKO_PRIVATE_KEY;
attributes[1] = new CK_ATTRIBUTE();
attributes[1].type = PKCS11Constants.CKA_LABEL;
attributes[1].pValue = label;
```

Semnaturile digitale sunt stocate sunt stocate atat pe card cat si in baza de date astfel incat daca datele de pe card sunt rescrise, ele sa poata totusi fi regasite si verificate pe baza semnaturii digitale folosind o interogare adecvata in baza de date a sistemului.

Exista cateva cazuri in care semnaturile digitale realizate de catre appletul crypto sunt verificate. In primul rand, are loc o verificare a semnaturii in timpul procesului de autentificare mutuala. Ambele carduri trebuie sa verifice semnaturile realizate asupra provocarilor. Daca semnatura nu se verifica, orice acces la card este interzis. Un alt caz de verificare a semnaturii este adaugarea unor noi date pe card.

Procedura de adaugare a unor noi inregistrari pe card include semnarea acestei informatii si trimiterea catre card a semnaturii si a certificatului corespondent. Atunci cand appletul de pe card primeste aceste informatii, mai intai verifica validitatea certificatului si extrage apoi cheia publica din certificat. Folosind cheia publica extrasa, semnatura este verificata si daca toti pasii se deruleaza cu succes, atunci datele, semnatura asupra lor si certificatul sunt stocate pe card.

Verificarea semnaturilor digitale calculate de appletul criptografic este realizata in appletul dezvoltat de echipa proiectului, fie ca este vorba de cel professional sau de cel pentru pacient. Acest ultim pas ar fi putut fi facut si in appletul criptografic dar datorita faptului ca cei doi appleti nu pot comunica independent, datele ar fi trebuit sa fie trimise prin intermediul aplicatiei offcard care nu poate fi considerata de incredere. O astfel de abordare ar fi constituit o prima bresa in securitatea sistemului.

8 DEZVOLTAREA APLICAŢIILOR SOFTWARE ON-CARD

8.1 TEHNOLOGII UTILIZATE : JAVACARD

Tehnologia Java Card (ORACLE, 2007) permite cardurilor inteligente şi altor dispozitive cu memorie foarte limitata de a rula aplicatii mici, numite appleti, care folosesc tehnologia Java. JavaCard oferă producătorilor de carduri inteligente o platforma de execuţie sigură şi interoperabila, care poate stoca şi actualiza aplicaţii multiple pe un singur dispozitiv.

Tehnologia Java Card este compatibila cu standardele existente pentru smart card. Tehnologia permite dezvoltatorilor să construiască, să implementeze si sa testeze aplicaţii şi servicii rapide si sigure. Acest proces accelerat reduce costurile de dezvoltare si creşte valoarea adaugata a produselor destinate clienţilor. Într-un mod complementar fata de Standard Enterprise şi ediţia Mobile Java 2 Platform, tehnologia Java Card permite usor integrarea token-urilor securizate într-o soluţie completă Java. Principalele obiective de proiectare ale tehnologiei Java Card sunt portabilitatea şi securitatea.

Java Card vizează definirea unui mediu standard de procesare inteligenta care să permită aceluiasi applet de a rula pe diferite carduri inteligente, similar modului in care un applet Java ruleaza pe computere diferite. Ca şi în Java, acest lucru este realizat folosind o combinaţie intre o maşină virtuală (Java Card Virtual Machine) şi o bibliotecă bine definita de rulare, care în mare măsură abstractizeaza conceptia appletului de diferentele între diverse carduri inteligente. Din păcate, portabilitatea rămâne limitata din cauza dimensiunii memoriei, a performanţelor şi a suportul de rulare a diferitelor modele de smart carduri.

Unul dintre obiectivele prioritare de proiectare Java Card este de a consolida securitatea unui smart card. Profitând de caracteristicile de securitate generale în platforma Java, platforma Java Card a integrat trei accesorii speciale de securitate şi anume : atomicitatea tranzacţiilor, clase criptografice şi un applet cu rol de firewall.

Atomicitatea tranzactiilor rezolva problema unei tranzacţii întrerupte şi a posibilelor modificări în memoria non-volatila. Dacă o tranzacţie este finalizată în mod normal, memoria va fi actualizata, în caz contrar, cardul nu va efectua nici o actualizare şi va reveni la starea anterioară.

Firewall-ul este folosit pentru a oferi o partitionare separată a memoriei pentru fiecare applet încărcat pe smart card. Acest lucru înseamnă că fiecare applet este stocat izolat de alte appleturi de pe card. Astfel, este imposibil pentru un applet ce nu functioneaza corect sa afecteze functionalitatea altor appleti.

Clasele criptografice oferă algoritmi de criptare/decriptare simetrică şi asimetrică, crearea semnăturii digitale şi verificarea acesteia, gestionarea codului PIN şi multe alte caracteristici. Criptografia şi clasele de securitate pot fi folosite pentru a semna şi autentifica fişierele de tip CAP şi pot să furnizeze un mecanism de instalare securizat.

Aşa cum am menţionat înainte, Java Card este unul dintre cele mai populare sisteme de operare pentru carduri inteligente. Tehnologia Java Card permite cardurilor inteligente şi altor dispozitive cu resurse foarte limitate de a rula aplicatii mici, numite appleti (NETBEANS, 2010). Aceasta ofera, de asemenea, producătorilor de carduri inteligente o platforma de execuţie sigură şi interoperabila, care poate stoca şi actualiza aplicaţii multiple pe un singur dispozitiv.

Tehnologia Java Card vizează definirea unui standard în mediul de procesare smart card care să permită aceluiasi applet Java Card de a rula, fără nici o modificare, pe diferite carduri inteligente. Acest lucru este un important avantaj deoarece appleturile pot rula pe orice alt smart card, atâta timp cât acesta acceptă tehnologia Java Card.

În plus, Java Card oferă mai multe mecanisme de securizare care asigura un grad inalt de securitate pentru applet-uri. Java Card oferă încapsularea datelor, ceea ce înseamnă că datele stocate pe card şi appleturile încărcate sunt executate într-un mediu izolat Java Card VM, separate de sistemul de operare şi hardware-ul.

Spre deosebire de alte maşini virtuale Java, un Java Card VM administrează, de obicei, mai multe aplicaţii, fiecare dintre ele manipuland date sensibile. Diferite aplicaţii sunt, prin urmare, separate una de alta printr-un applet firewall care controaleaza şi restricţionează accesul datelor intre appleti.

În plus, applet-ul în sine este un mecanism de securitate. De fapt, este o maşină de stare care procesează solicitările primite numai de la comanda offcard şi răspunde prin date sau cuvinte de stare ca răspuns înapoi catre dispozitivul de interfaţă. Pe lângă acest mediu sigur pentru appleturi şi date, Java Card ofera de asemenea servicii criptografice cum ar fi semnarea şi manipularea certificatelor digitale şi suportă algoritmi de criptare utilizati în mod obişnuit precum Encryption Standard Datelor (DES), triple Data Encryption Standard (DES triple), Encryption Standard avansat (AES) , Rivest, Shamir şi Adleman (RSA).

Având în vedere toate caracteristicile oferite de tehnologia Java Card şi cerintele proiectului de cercetare, s-a decis că aceasta tehnologie sa fie pusă în aplicare cu astfel de carduri inteligente, astfel încât appleturi sa poata rula pe orice smart card, independent de producător. Prin urmare, appleturile care urmează să fie dezvoltate nu se limitează la un anumit tip smart card şi sunt portabile pe orice cartela inteligenta care are suport Java Card.

8.2 APPLETUL PENTRU PACIENTI

Appletul pentru smart cardul pacientului se gaseste in pachetul srspirim; appletul este de altfel singura componenta a pachetului mentionat. Clasa principala a appletului este numita SRSPIRIM; pachetul mai contine alte sase clase definite astfel: BloodTest, Diagnosis, MedicalRecord, PersonalInformation, MedicalReference si RadiologicExposure.

Clasa principala a appletului contine un vector de obiecte de tip Diagnosis, o colectie de obiecte MedicalVisit, un obiect de tip PersonalInformation si unul de tip MedicalRecord. Colectiile mentionate anterior sunt buffere circulare, ceea ce inseamna faptul ca un numar predefinit de elemente pot fi continute in interiorul vectorului iar daca numarul acesta este depasit, atunci cel mai vechi (altfel spus primul) este inlocuit cu cel mai nou element.

Aceasta este o abordare frecvent utilizata in contextul programarii in Java Card datorita nevoii constante de a consuma cat mai putina memorie. Dimensiunea bufferelor circulare este configurata momentan in cadrul proiectului la 10 elemente dar poate fi redefinita inainte de compilarea appletului. Obiectele MedicalRecord si PersonalInformation nu sunt retinute in nici un buffer fiindca un pacient are numai o singura inregistrare medicala si un singur set de informatii personale (nume, prenume, adresa, etc.).

Clasa PersonalInformation este utilizata pentru a retine informatiile personale ale pacientului in interiorul smart cardului. Identificatorul pentru aceasta clasa este 0x01. Informatiile personale ale pacientului pot fi citite de catre toti membrii din sistem (orice rol) dar nu pot fi scrise decat de operatorul de carduri. Identificatorul clasei asigura aceasta politica de securitate asa cum a fost descris anterior.

Numele si prenumele titularului de card, initiala tatalui si codul numeric personal sunt stocate ca variabile de clasa impreuna cu un camp care codifica statutul special al pacientului (ex. veteran de razboi, student, persoana cu dizabilitati). O alta informatie importanta stocata in clasa PersonalInformation este informatia de contact in caz de urgenta. Sunt memorate numele, prenumele si un numar de telefon al persoanei care trebuie contactata in cazul unui accident.

STATUS	CODE
Copil (<18 ani)	0
Student (18-26 ani)	1
Gravida	2
Pensionar	3
Veteran de Razboi	4
Handicapat	5
Somer	6
Ajutorat Social	7

FIGURĂ 8-1 CODIFICAREA STATUTULUI SOCIAL AL PACIENTULUI

Setarea informatiilor personale ale pacientului si a informatiilor in caz de urgenta este realizata de catre o autoritate nationala autorizata inainte de predarea cardului catre pacient. Prezenta informatiilor personale ale pacientului in cadrul appletului de pe smart card nu este numai o masura suplimentara de securitate pentru posesorul cardului, dar si o caracteristica foarte utila in cazul unei urgente medicale.

Clasa MedicalRecord simbolizeaza o inregistrare medicala apartinand unui pacient. Prezenta inregistrarii medicale a pacientului pe un smart card pe care pacientul il are asupra lui tot timpul este foarte importanta mai ales in cazul unei urgente. De exemplu, un paramedic poate sa cunoasca grupa sanguina a pacientului chiar daca pacientul este in stare de inconstienta doar prin citirea informatiei de pe smart card. Codul de identificare al clasei MedicalRecord este 0x02.

Clasa MedicalRecord contine un camp ce reprezinta codificarea grupei sagvine a pacientului. Acest camp are o lungime fixa de un octet si este codificat in conformitate cu tabela de mai jos.

Byte	0x00	0x01	0x02	0x03	0x04	0x05	0x06	0x07
Grupa Sangvina	AB+	AB-	A+	A-	B+	B-	0+	0-

FIGURĂ 8-2 CODIFICAREA GRUPEI SANGVINE

Dupa setarea acestui camp in cadrul appletului de pe smart cardul pacientului, acesta nu mai poate fi ulterior modificat. Un alt camp important al clasei MedicalRecord este campul Donor. Acesta specifica daca pacientul care este posesorul cardului este eligibil pentru a dona sange. Spre deosebire de campul BloodType, valoarea acestui camp poate fi modificata dupa faza de initializare.

Motivul pentru acest lucru este evident, datorita faptului ca starea de donator depinde direct de starea de sanatate a pacientului. Clasa MedicalRecord contine de asemenea si alergiile pacientului; acesta este un camp important datorita faptului ca anumite tratamente sau medicamente prescrise trebuie sa fie corelate si cu alergiile sale cunoscute pana in prezent.

Codul de identificare al clasei MedicalReference este 0x03 si el este folosit pentru respectarea politicii de securitate descrise anterior. Clasa Medical Reference este folosita de catre medicul de familie atunci cand acesta decide sa-i faca o trimitere pacientului catre un medic radiolog. Medicul generalist emite astfel o trimitere medicala pe cardul pacientului catre o unitate medicala precizand si investigatia avuta in vedere pentru pacient. Trimiterile medicale sunt stocate pe smart card ca un buffer circular de obiecte MedicalReference. Trimiterea medicala contine un numar serial compus din doua parti: prima formata din litere si a doua din cifre.

Urmatoarele campuri din inregistrarea medicala sunt identificatorul medicului care a facut trimiterea si identificatorul unitatii sale medicale. Urmeaza apoi identificatorii pentru specializarea si unitatea medicala catre care este indrumat pacientul. Mai sunt de asemenea retinute data trimiterii si identificatorul diagnosticului prezumptiv pus de medicul de familie. Medicul de familie poate sa scrie si un motiv al trimiterii emise care va aparea pe ecranul medicului radiolog.

Trimiterile medicale au doua tipuri de prioritati ce pot servi doua scopuri si anume : campul prioritate din clasa MedicalReference indica daca trimiterea are un caracter ordinar sau urgent. Daca acest camp este setat la valoarea 0x00 atunci trimiterea este una normala, iar daca are valoarea 0x01, este vorba despre o trimitere in urgenta. Campul type din interiorul clasei indica daca pacientul trebuie vazut de un medic specialist si are nevoie de un set de teste medicale (valoarea 0x00) sau trebuie sa fie spitalizat (valoarea 0x01).

Clasa medicala este folosita pentru a stoca ultimele diagnostice ale unui pacient pe smart card. Diagnosticele sunt pastrate sub forma unui buffer circular de obiecte de tip Diagnosis. Un obiect Diagnosis contine un identificator pentru denumirea acestuia si data la care pacientul a fost diagnosticat. Codul de identificare al clasei Diagnosis este 0x04.

8.3 CONVENTII DE CODARE A DATELOR

Exista anumite aspecte ce trebuiesc luate in considerare atunci cand se stabilesc stilul de codare a datelor si formatul mesajelor. Mai intai, intreaga memorie disponibila pe smart carduri trebuie sa fie atent utilizata acordand o atentie deosebita reducerii spatiului de memorie alocat ce risca sa nu fie utilizat eficient. Prin urmare, au fost introduse o serie de conventii de codare ce vor fi discutate in continuare.

In al doilea rand, schimbul de informatii cu cardul este „punctul slab” al aplicatiei datorita implicarii operatiilor de citire/scriere foarte lente. Aplicatia trebuie sa astepte ca smart cardul sa primeasca, sa proceseze si sa returneze un raspuns. Astfel, schimbul de mesaje cu smart cardul trebuie sa depaseasca acest inconvenient si trebuie sa incapsuleze cat mai multa informatie fara posibilitatea de a depasi cei 255 octeti de date. Abordarea aleasa de catre aplicatia offcard in privinta acestor mesaje trebuie tratata cu deosebita atentie si rezolvata eficient.

Dat fiind faptul ca anumite informatii care trebuiesc stocate pe smart card au o lungime variabila (ex. nu toate numele au aceeasi lungime), exista doua alternative pentru stocarea acestor date pe card. O abordare naiva ar fi sa se aloce buffere pe card de dimensiune maxima pentru fiecare camp. Acest lucru este inaccceptabil dat fiind faptul ca o mare cantitate de memorie este risipita. Prin urmare, o abordare diferita trebuie luata in considerare: toate campurile sunt precedate de lungimea bufferului care trebuie alocat pe card pentru a stoca acea informatie.

Datorita faptului ca sunt multe campuri de lungime variabila, conventia este aplicata si variabilelor de lungime constanta, pentru pastrarea consistentei. Fata de cantitatea de memorie economisita folosind aceasta abordare, memoria suplimentara necesara gestiunii acestor campuri de lungime este neglijabila. Pentru o mai buna intelegere a acestui stil de codare a informatiei, urmatoarele paragrafe prezinta modul in care de codare a sirurilor de caractere, a numerelor si a datelor calendaristice.

Sirurile de caractere ASCII – sunt precedate de un byte care indica lungimea sirului. Exemplu de codare a unui sir de 7 caractere : 0x07 0x65 0x78 0x61 0x6D 0x70 0x6C 0x65

Numere – octetii care compun numarul intreg sunt de asemenea precedati de numarul lor : Exemplu de codare pentru 2011 : 0x02 0x07 0xDB.

Date calendaristice – data este precedata de un octet cu valoarea 4 , urmeaza apoi octetul care indica ziua, cel care indica luna si inca doi octeti pentru an. Exemplu de codare a datei de 30.08.2011 : 0x04 0x1E 0x08 0x07 0xDB.

Data calendaristica este un exemplu de camp de lungime fixa care este codat in aceeasi maniera in ideea pastrarii aceluasi format. Acest mecanism de codare se aplica tuturor campurilor descrise in sectiunile care urmeaza. Singurele informatii care nu este codate in acest fel sunt campurile care ocupau un singur octet si care puteau fi trimise folosind unul dintre parametrii APDU. In orice caz, majoritatea campurilor pastreaza formatul discutat in aceasta sectiune.

8.4 SERIALIZAREA SI DESERIALIZAREA OBIECTELOR

In scopul de a reduce numarul de comenzi APDU pentru transferul informatiilor cu smart cardul, a fost propusa urmatoarea abordare: toate informatiile specifice unei clase sunt transferate complet, intr-o singura comanda, fie ca este vorba despre o operatie de citire sau una de scriere.

De exemplu, in loc sa existe pentru fiecare camp o metoda Get si o metoda Set, vor exista doua metode Set si Get la nivelul clasei PersonalInformation. Pe card informatia este salvata ca un simplu buffer de octeti, insotit de semnatura operatorului de card. In momentul citirii informatiei prin metoda GetPersonalInformation(), bufferul de octeti este parsat de catre aplicatia off-card si apoi se poate extrage informatia legata de nume.

Aceasta abordare presupune practic serializarea tuturor campurilor specifice unei clase si deserializarea lor dupa un format bine stabilit intre aplicatiile off-card si on-card. Aceasta abordare a fost ulterior aplicata tuturor claselor dezvoltate in cadrul appletului JavaCard.

9 INTEGRAREA ŞI TESTAREA COMPONENTELOR DE BAZĂ ALE SISTEMULUI SRSPIRIM

9.1 CONFIGURAREA SI INSTALAREA APLICATIILOR SRSPIRIM

Aplicatiile on-card si off-card din cadrul proiectului SRSPIRIM vor fi disponibile in forma finala sub forma unor aplicatii web care sunt operationale odata ce serverul web si serverele de baze de date sunt pornite si configurate corespunzator. Statia de lucru pe care se pot testa aplicatiile SRSPIRIM poate fi orice tip de PC, care are urmatoarea configuratie minimala :

- HDD 10 Gb
- CD-Rom/ DVD-Rom
- Minimum 1Gb Mb RAM
- Procesor Intel minimum 1 GHz
- Tastatura normala (104 taste)
- Mouse normal
- Cel putin 2 porturi USB
- Conexiune Ethernet sau wireless configurat pentru accesul la Internet

In privinta echipamentelor aditionale sunt necesare doua cititoare de smart carduri cu conexiune USB cum sunt de exemplu cele oferit de Omnikey. Cerintele software sunt:

- Sistem de operare Windows XP sau Windows 7
- Suport Java: JDK 6+.

Etapele de instalare ale aplicatiei client care trebuiesc parcurse sunt urmatoarele:

- Instalare automata (Windows)
- Instalare Java JDK6
- Instalare manuala a driverului pentru cititorul de card
- Instalare automata a driverului pentru smart card

Etapele de instalare ale bazelor de date care intra in componenta sistemului SRSPIRIM sunt:

- Instalarea servelor de baze de date MySql pentru stocarea datelor medicale
- Incarcarea unei bazei de date existente sau crearea uneia noi folosind scripuri SQL

Pentru instalarea si configurarea infrastructurii PKI specifica sistemului SRSPIRIM, se va folosi aplicatia certSAFE; instalarea acesteia cuprinde 3 faze principale si anume:

1) Instalarea propriu-zisa a aplicatiei, care consta in executia programelor de setup continute in kit-ul de instalare.

2) Configurarea aplicatiei, care consta in definirea infrastructurii de administrare specific cu profilul si cerintele arhitecturii pentru organizatia in cadrul careia se realizeaza implementarea.

3) Emiterea de certificate si managementul acestora.

9.2 TESTAREA APPLETILOR ON-CARD

Testarea aplicatiilor on-card a fost realizata folosind setul de aplicatii Oberthur Technologies PCOM32. Acest set curpinde o serie de aplicatii ce furnizeaza o modalitate de a trimite multiple comenzi APDU catre smart card si de a verifica usor rezultatul obtinut. Aceasta aplicatie furnizeaza de asemenea o modalitate de a crea scripturi APDU pentru a fi rulate in mod automat pe smart card. In procesul de testare, toate clasele implementate si interactiunile dintre ele au fost evaluate in concordanta cu specificatiile tehnice ale proiectului.

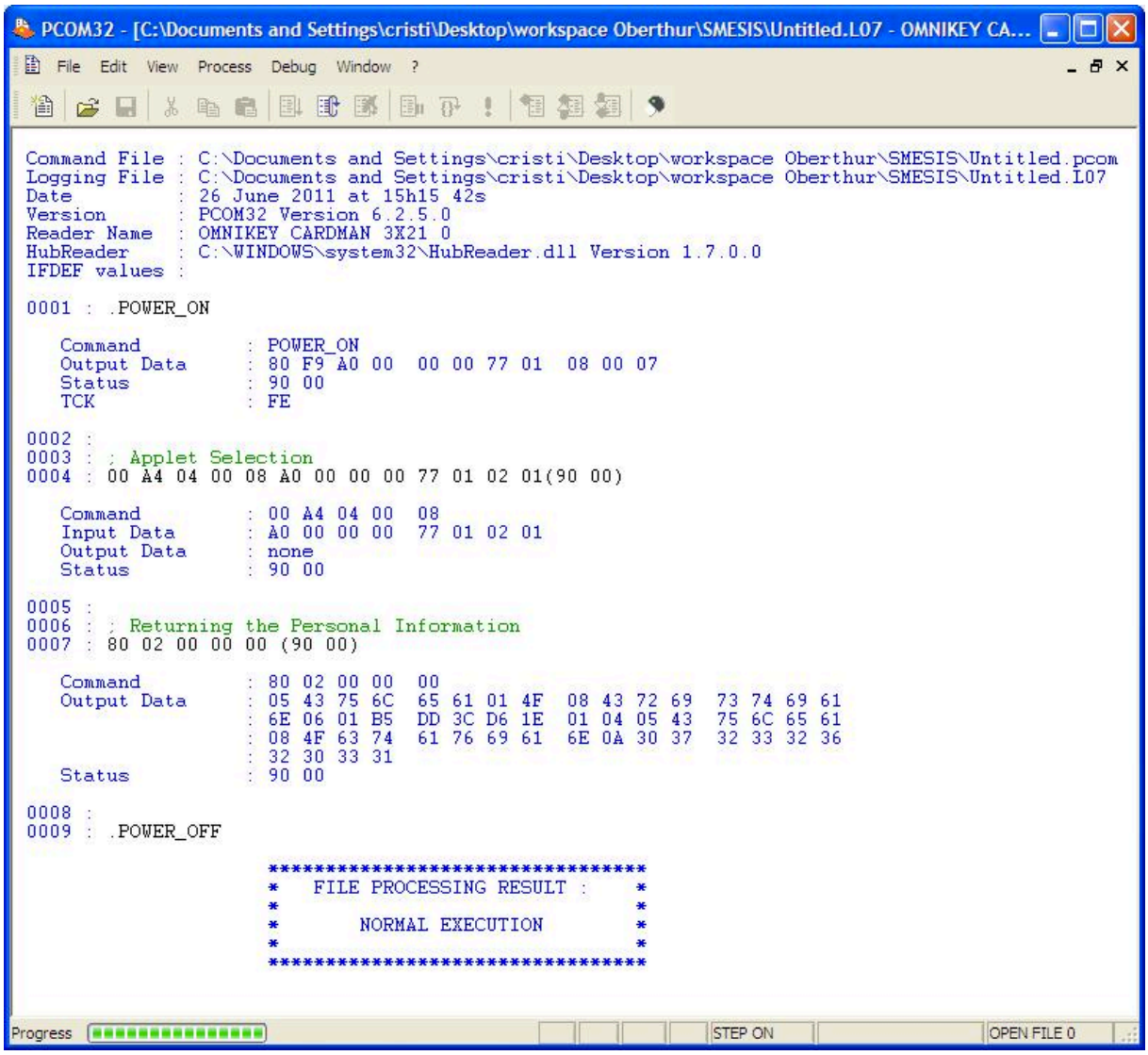
```
.POWER_ON

; Applet Selection
00 A4 04 00 08 A0 00 00 00 77 01 02 01(90 00)

; Returning the Personal Information
80 02 00 00 00 (90 00)

.POWER_OFF
```

FIGURĂ 9-1 COMENZI APDU TRIMISE CATRE CARD

Exemplul de script prezentat mai sus arata doua comenzi APDU. Prima este folosita pentru a selecta appletul dorit sa ruleze iar a doua este utilizata pentru a-i trasmite appletului sa returneze informatiile personale apartinand titularului de smart card. Codul dintre paranteze este codul de raspuns asteptat la comenzile APDU. Rezultatul rularii scriptului este prezentat in continuare. In figura de mai jos este prezentat rezultatul rularii comenzii GET_INFO din cadrul scriptului.

```
Command          : 80 02 00 00  00
Output Data      : 05 43 75 6C  65 61 01 4F  08 43 72 69  73 74 69 61
                 : 6E 06 01 B5  DD 3C D6 1E  01 04 05 43  75 6C 65 61
                 : 08 4F 63 74  61 76 69 61  6E 0A 30 37  32 33 32 36
                 : 32 30 33 31
Status           : 90 00
```

FIGURĂ 9-2 APDU PENTRU OBTINEREA INFORMATIILOR GENERALE



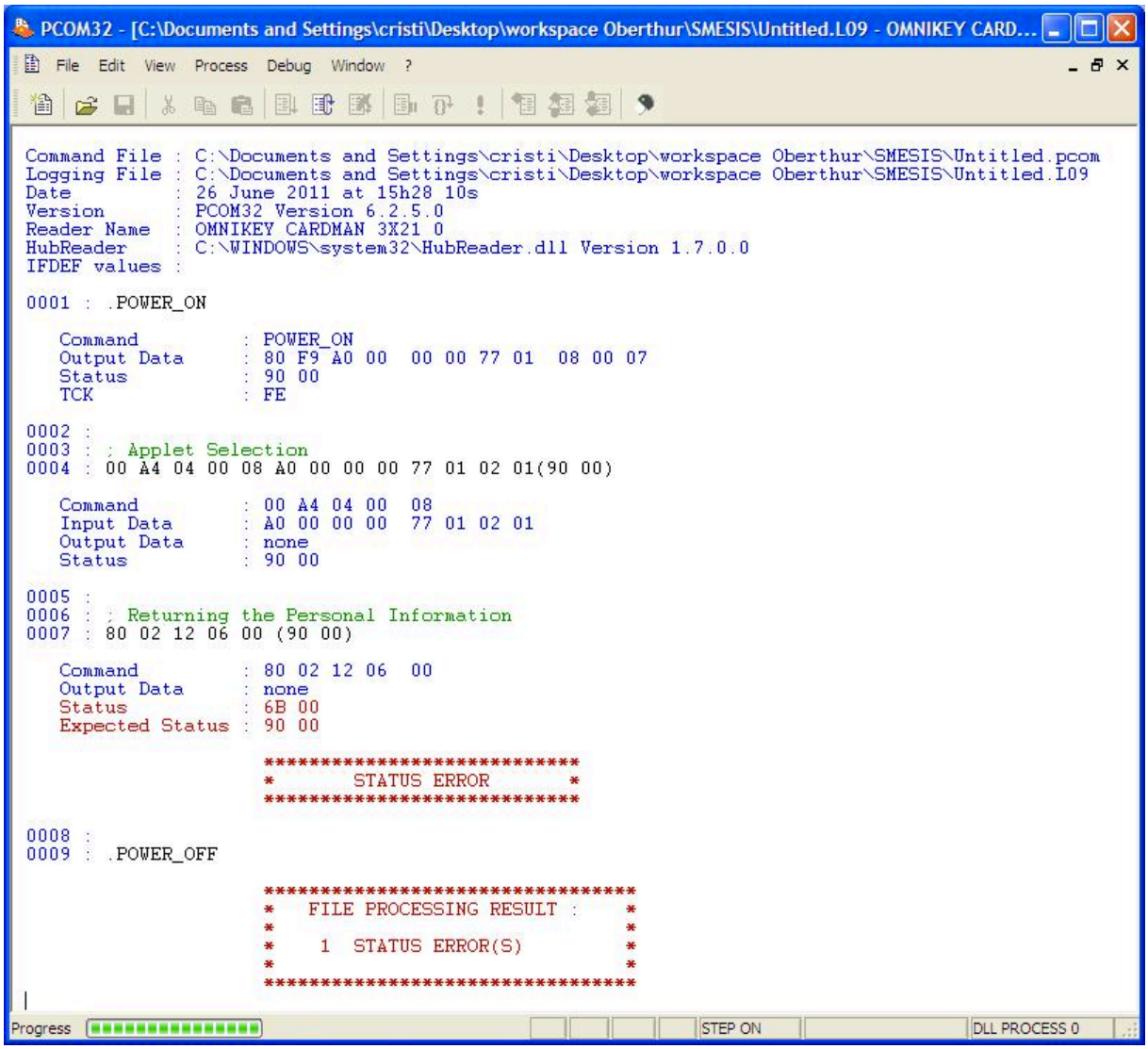
FIGURĂ 9-3 REZULTATUL UNEI COMENZI EXECUTATE CORECT

Primul lucru care merita observat este faptul ca nu exista nici o data de intrare si ca starea raspunsului este 90 00, ceea ce semnifica succesul executiei. Iesirea comenzii reprezinta informatiile personale ale titularului de card codate asa cum a fost deja descris in cadrul documentului. Dupa ce toate instructiunile din script au fost rulate, aplicatia semnalizeaza daca au aparut erori pe durata executiei. In exemplul prezentat in figura, nu au existat erori iar mesajul aplicatiei este NORMAL EXECUTION.

Figura urmatoare arata iesirea aplicatiei pentru o comanda incorecta din cadrul scriptului; comanda a fost modificata prin schimbarea valorilor parametrilor.

```
Command          : 80 02 12 06  00
Output Data      : none
Status           : 6B 00
Expected Status  : 90 00
```

FIGURĂ 9-4 TRIMITEREA UNUI APDU CU PARAMETRI INCORECTI



FIGURĂ 9-5 IESIREA UNEI COMENZI INCORECTE

Iesirea acestei comenzi este foarte diferita de cea a primei comenzi. Partea de date nu mai exista iar starea raspunsului nu este 90 00 ci 6B 00, ceea ce indica exceptia SW_WRONG_P1P2. Aplicatia PCOM32 afiseaza mesajul STATUS ERROR dupa executia scriptului anuntand esecul executiei. Aplicatia PCOM32 oferita de Oberthur Technologies s-a dovedit foarte utila in procesul de testare al claselor implementate, ajutand la detectarea erorilor si aratandu-le intr-un mod usor de interpretat.

9.3 TESTAREA APLICATIILOR OFF-CARD

Toate functionalitatile specificate in cadrul acestui document sunt implementate si complet functionale. Intregul sistem ce include aplicatiile off-card , appletii on-card si bazele de date au fost testate folosind diverse scenarii. Scenariile au fost concepute pentru a simula situatii reale care pot aparea intr-un astfel de sistem. Detalii suplimentare despre aceste scenarii sunt prezentate in sectiunile urmatoare.

DATE DESPRE TESTARE

In comntinuare este prezentat un raport de testare folosit de catre echipa insarcinata cu testarea suitei de aplicatii SRSPIRIM in cateva scenarii reprezentative pentru sistem. Ca documente de referinta au fost utilizate planul de testare software si descrierea testării software. Aplicaţiile SRSPIRIM au fost testate atât pe sistemul de operare iniţial Windows XP Prof, versiunea SP3, pe care au fost dezvoltate, dar şi pe versiuni ulterioare ca VISTA, sau Windows 7 Prof, SP 1.

Crearea testelor a fost realizata de catre unul dintre membrii echipei de cercetare ai CertSIGN si anume lect. dr. ing. Cezar Plesca – acronim CezarP. Execuţia si obtinerea logurilor de testare a fost realizata de dr. ing. Armand Ropot – acronim ArmandR. Perioada de testare a fost 01.11.2013 - 22.11.2013.

9.4 TESTE LEGATE DE PERSONALIZAREA CARDURILOR

1. Autentificarea în aplicaţia de personalizare a cardului de pacient

ID: 1	**Data creare: 01.11.2013**		
Creator:	CezarP	**Platforma:**	IBM ThinkCenter P4 3Ghz/ 1GB
		OS:	Windows 7 Prof SP1
		Versiune:	2012
Starea ultimului log:	valid	**Versiune de start:**	Windows XP Prof SP3
Sumar:	Autentificarea operatorului de card (organ abilitat pentru personalizarea cardurilor de pacient) este parte componenta a aplicaţiei de personalizare a unui card radiologic electronic		
Precondiţii:	SRSPIRIM instalat si functional, aplicaţia de personalizare card radiologic instalată şi functională, operator de card valid		
Specificaţii de intrare:	1. Pentru a utiliza această aplicaţie este necesară prezenţa unui operator de card. Acesta trebuie să deţină un card de operator şi să se autentifice pe baza unui cod pin; 2. Se introduce cardul/token-ul operatorului de carduri care urmează să emită cardul de pacient; 3. Odată detectat cardul de către staţia de lucru va apărea o fereastră, în care i se va cere operatorului de card introducerea codului pin; 4. După introducerea şi validarea codului pin se va deschide un ecran care permite introducerea de la tastatura a unui CNP pentru pacientul căruia urmează a i se edita sau emite cardul.		

Specificaţii de ieşire (Rezultate aşteptate):	1. După introducerea cardului profesional de operator apare un ecran „Introduceţi codul pin”; 2. În câmpul respectiv de date trebuie introdus un cod pin corect; 3. Dacă s-a introdus un cod pin incorect se atenţionează corespunzător operatorul de card;. 4. Dacă s-a introdus un cod pin corect se trece la ecranul de introducere de la tastatură a unui CNP de pacient căruia urmează a i se personaliza cardul;

2. Căutarea în baza de date de personalizare a unui pacient

ID: 2	**Data creare: 02.11.2013**		
Creator:	CezarP	**Platforma:**	IBM ThinkCenter P4 3Ghz/ 1GB
		OS:	Windows 7 Prof SP1
		Versiune:	2012
Starea ultimului log:	valid	**Versiune de start:**	Windows XP Prof SP3
Sumar:	Căutarea unui pacient în baza de date a aplicaţiei pentru personalizarea unui card radiologic		
Precondiţii:	SRSPIRIM instalat si functional, aplicaţia de personalizare card radiologic instalată şi functională, operator de card valid		
Specificaţii de intrare:	1. Se efectuează întocmai paşii de la 1 la 3 din testul ID 1; 2. Se introduc, în câmpul CNP, cele 13 cifre ale pacientului care urmează a fi selectat; 3. Se da click pe butonul "Cautare"		
Specificaţii de ieşire (Rezultate aşteptate):	1. Dacă persoana cu CNP-ul specificat se găseşte în baza de date se trece la următoarea etapă, adică pe ecran apar datele personale ale acesteia; 2. Dacă persoana cu CNP-ul specificat nu se găseşte în baza de date, atunci trebuie actualizată baza de date numai de autoritatea de personalizare a cardurilor; 3. Dacă CNP-ul este introdus greşit – conţine, de exemplu, şi litere pe lângă cifre – se atenţionează operatorul printr-un mesaj corespunzator; 4. Dacă CNP-ul este introdus greşit – de exemplu s-a introdus incorect data naşterii – se atenţionează operatorul printr-un mesaj corespunzator; 5. Dacă CNP-ul este introdus greşit –de exemplu s-au introdus mai multe sau mai puţine cifre – se atenţionează operatorul printr-un mesaj corespunzator;		

3. Scrierea datelor în aplicaţia de personalizare a cardului de pacient

ID: 3	**Data creare: 03.11.2013 11:56**		
Creator:	CezarP	**Platforma:**	IBM ThinkCenter P4 3Ghz/ 1GB
		OS:	Windows 7 Prof SP1
		Versiune:	2012
Starea ultimului log:	valid	**Versiune de start:**	Windows XP Prof SP3
Sumar:	Scrierea unui card de pacient în aplicaţia de personalizare		
Precondiţii:	SRSPIRIM instalat si functional, aplicaţia de personalizare card radiologic instalată şi functională, operator de card valid		
Specificaţii de intrare:	1. Această operaţie este condiţionată de găsirea pacientului în baza de date conform paşilor 1-3 din cadrul testului ID 2; 2. Găsirea pacientului căruia urmează să i se creeze un card radiologic este însoţită de datele sale personale; 3. Se da click pe butonul „Scriere pe card”.		
Specificaţii de ieşire (Rezultate aşteptate):	1 Datele extrase din baza de date a aplicaţiei SRSPIRIM sunt numai de tip read-only; 2. Dacă nu este introdus – în cititorul de carduri aferent – un smart card blank sau cardul pacientului ce urmează a fi actualizat, butonul de scriere pe card nu este activ; 3. La introducerea unui card blank sau a unui card deja scris în cititorul aferent scrierii se activează opţiunea de scriere pe card; 4. Încheierea activităţii de scriere a unui card nou de pacient este anunţată pe ecran printr-un mesaj corespunzător („Scrierea cardului s-a efectuat cu succes”).		

9.5 TESTE DE GESTIUNE A FISEI MEDICALE

4. Autentificarea in aplicatia de gestiune a fisei medicale

ID: 4	**Data creare: : 05.11.2013 11:51**		
Creator:	CezarP	**Platforma:**	IBM ThinkCenter P4 3Ghz/ 1GB
		OS:	Windows 7 Prof SP1
		Versiune:	2012
Starea ultimului log:	valid	**Versiune de start:**	Windows XP Prof SP3
Sumar:	Autentificarea în aplicaţia de gestiune a fisei medicale		
Precondiţii:	SRSPIRIM instalat si functional, aplicaţia de gestiune a fisei medicale - instalată şi functională, card de medic de familie valid		
Specificaţii de intrare:	1. Aplicaţia de gestiune a fisei medicale este utilizată numai de către medicul de familie şi are ca scop vizualizarea si/sau modificarea datelor medicale ale unui pacient 2. Utilizarea aceastei aplicaţii este restricţionată doar pentru medici de familie şi aceştia trebuie să aibă carduri profesionale cu care să se autentifice 3. Se introduce cardul profesional al medicului, care urmează sa realizeze gestiunea 4. Odată detectat cardul de către staţia de lucru va apărea fereastra de autentificare care cere medicului introducerea codului pin; 5. După introducerea şi validarea codului pin, se va deschide un ecran, care permite medicului să vizualizeze fisa medicala electronică a pacientului.		
Specificaţii de ieşire (Rezultate aşteptate):	1. Dacă se introduce cardul profesional al medicului de familie apare un ecran de introducere a codului pin; 2. Dacă se introduce un alt card (de radiolog) fisa medicala va fi disponibila numai pentru vizualizare dar nu si pentru modificare;		

5. Vizualizarea informatiilor din fisa medicala specifica unui pacient

ID:5	**Data creare: 05.11.2013 12:08**		
Creator:	CezarP	**Platforma:**	IBM ThinkCenter P4 3Ghz/ 1GB
		OS:	Windows 7 Prof SP1
		Versiune:	2012
Starea ultimului log:	valid	**Versiune de start:**	Windows XP Prof SP3
Sumar:	Vizualizarea informatiilor din fisa medicala a unui pacient		
Precondiţii:	SRSPIRIM instalat si functional, aplicaţia de gestiune a fisei medicale - instalată şi functională, card de profesionist medical valid		
Specificaţii de intrare:	1. Această operaţie este condiţionată de autentificarea profesionistului paşii 1-5 din cadrul testului ID 4; 2. La introducerea în cititorul de card a cardului de pacient apare ecranul ce permite vizualizarea fisei sale medicale;		
Specificaţii de ieşire (Rezultate aşteptate):	1. Precizările făcute la specificaţii de intrare sunt corecte numai dacă ambele carduri (de profesionist şi pacient) sunt prezente în cititoarele de card alocate aplicaţiei; 2. Dacă se scoate cardul profesionistului, aplicaţia se va închide, întrucât se consideră că nu există nivelul de acces necesar pentru vizualizarea acestor date confidenţiale ale pacientului; 3. Dacă se va retrage cardul cetăţeanului, datele acestuia vor dispărea putând fi revizualizate numai dacă se reintroduce cardul sau; 4. Aplicaţia permite medicului de familie sa verifice datele pacientului cu cele din baza de date, dacă exista conexiune cu aceasta baza de date prin apăsarea pe opţiunea “Verifica online”.		

6. Editarea datelor în aplicaţia destinată gestiunii fişei medicale

ID: 6	**Data creare: 06.11.2013 13:53**		
Creator:	CezarP	**Platforma:**	IBM ThinkCenter P4 3Ghz/ 1GB
		OS:	Windows 7 Prof SP1
		Versiune:	2012
Starea ultimului log:	valid	**Versiune de start:**	Windows XP Prof SP3
Sumar:	Editarea/scrierea datelor medicale în aplicaţia destinată scrierii pe card a fişei medicale a pacientului		
Precondiţii:	SRSPIRIM instalat si functional, aplicaţia pentru scrierea pe card a fisei medicale a pacientului lansată în execuţie, card de medic de familie si card de pacient valide.		
Specificaţii de intrare:	1. Această operaţie este condiţionată de autentificarea medicului de familie în conformitate cu paşii din cadrul testului ID 4; 2. Dacă CNP-ul este validat şi este găsit în baza de date pe ecran vor apărea datele pacientului al cărui card urmează a fi personalizat; 3. Câmpurile „Alergii", „Alerte speciale" pot fi editate suplimentar faţă de ultima scriere din baza de date; 4. Se acţionează asupra butonului „Scriere pe card" – activ dacă în cititorul de card aferent pacientului este introdus cardul acestuia – noile date fiind transferate pe card si in baza de date.		
Specificaţii de ieşire (Rezultate aşteptate):	1. Precizările făcute la specificaţii de intrare sunt corecte numai dacă ambele carduri (de medic de familie şi pacient) sunt prezente în cititoarele de card alocate aplicaţiei; 2. Dacă se scoate cardul medicului de familie, aplicaţia se va închide, întrucât se consideră că nu există nivelul de acces necesar pentru vizualizarea acestor date confidenţiale ale pacientului; 3. Dacă se va retrage cardul pacientului, datele acestuia vor dispărea putând fi revizualizate numai dacă se reintroduce cardul;		

9.6 TESTE DE VALIDARE A SEMNATURII SI A CERTIFICATULUI

7. Verificarea semnaturii medicului asupra informatiilor de baza

ID: 7	**Data creare: 08.11.2013 20:53**		
Creator:	CezarP	**Platforma:**	IBM ThinkCenter P4 3Ghz/ 1GB
		OS:	Windows 7 Prof SP1
		Versiune:	2012
Starea ultimului log:	valid	**Versiune de start:**	Windows XP Prof SP3

Sumar: Verificarea semnaturii medicului asupra informatiilor de baza din fisa medicala

Precondiţii: SRSPIRIM instalat si functional, aplicaţia pentru gestiunea fisei medicale lansată în execuţie, card de medic de familie si card de pacient valide

Specificaţii de intrare:

1. Aplicatia permite verificarea informatiilor semnate de medicul de familie prin intermediul unui buton numit „Verificare date card”.
2. Verificarea informatiilor are loc prin intermediul appletului de pe cardul pacientului.
3. Certificatul medicului este incarcat din containerul de certificate stocate pe cardul pacientului.
4. Se extrage cheia publica din certificatul medicului si se verifica semnatura aplicata datelor medicale de baza specifice pacientului.

Specificaţii de ieşire (Rezultate aşteptate):

1. Precizările făcute la specificaţii de intrare sunt corecte numai dacă cardul de pacient este prezent în cititorul de card alocate aplicaţiei;
2. Dacă se va retrage cardul pacientului, datele acestuia vor dispărea putând fi revizualizate numai dacă se reintroduce cardul;
3. Daca semnatura este validata se genereaza un mesaj corespunzator iar in caz contrar se genereaza un mesaj de eroare prin care utilizatorul este anuntat.

8. Validarea autenticitatii certificatelor profesionistilor medicali

ID: 8	**Data creare: 10.11.2013 21:53**		
Creator:	CezarP	**Platforma:**	IBM ThinkCenter P4 3Ghz/ 1GB
		OS:	Windows 7 Prof SP1
		Versiune:	2012
Starea ultimului log:	valid	**Versiune de start:**	Windows XP Prof SP3
Sumar:	Validarea autenticitatii certificatelor profesionistilor (medici, radiologi)		
Precondiţii:	SRSPIRIM instalat si functional, aplicaţia pentru gestiunea fisei medicale lansată în execuţie, card de pacient valid		
Specificaţii de intrare:	1. Aplicatia permite verificarea informatiilor certificatelor stocate pe cardul pacientului prin intermediul unui buton numit „Verificare certificate”. 2. Verificarea informatiilor are loc prin intermediul appletului de pe cardul pacientului. 3. Se alege unul dintre certificatele din containerul de certificate stocate pe cardul pacientului. 4. Se extrage semnatura din certificatul ales si aceasta este verificata folosind certificatul autoritatii de personalizare (PERSONALIZATION CA). 5. Se verifica pe card certificatul autoritatii de personalizare folosind certificatul autosemnat al autoritatii centrale de certificare (ROOT CA).		
Specificaţii de ieşire (Rezultate aşteptate):	1. Precizările făcute la specificaţii de intrare sunt corecte numai dacă cardul de pacient este prezent în cititorul de card alocat aplicaţiei; 2. Dacă se va retrage cardul pacientului, datele acestuia vor dispărea putând fi revizualizate numai dacă se reintroduce cardul; 3. Daca verificarile lantului de certificate sunt validate se genereaza un mesaj corespunzator iar in caz contrar se genereaza un mesaj de eroare prin care utilizatorul este anuntat care dintre certificatele din lant nu este valid.		

9.7 TESTE LEGATE DE INREGISTRARI, TRIMITERI SI FISE RADIOLOGICE

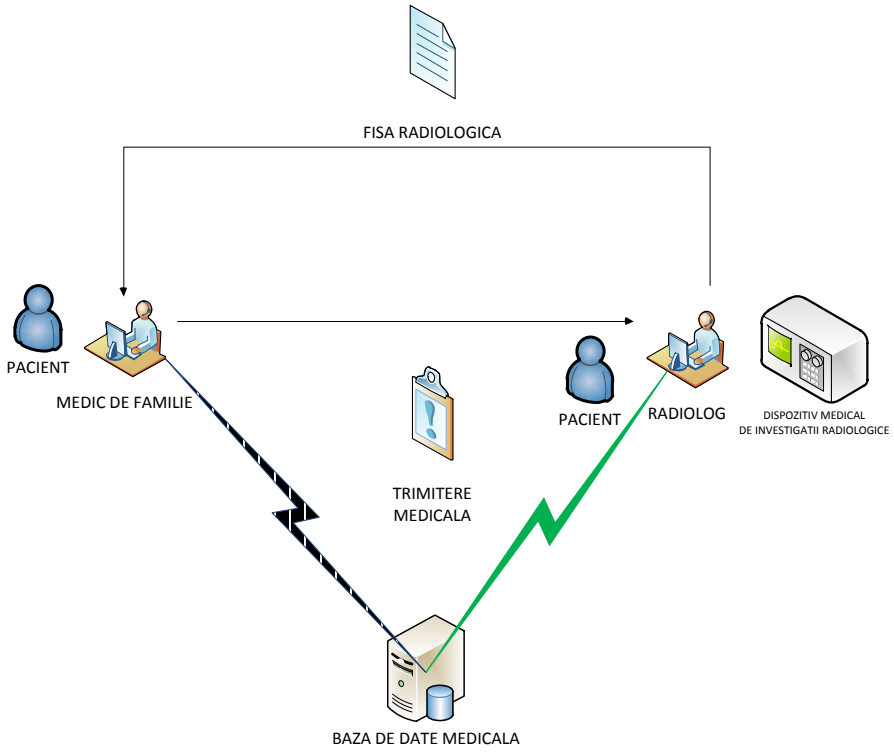
TESTAREA INREGISTRARILOR MEDICALE

Scenariile de test referitoare la inregistrarile medicale au constat in completarea unuia sau a tuturor campurilor necesare, ori in modificarea acestor informatii si afisarea datelor anterior completate. In cazul completarii unei fise pentru prima data, s-a realizat un test pentru a se vedea daca grupa de sange este corect completata.

Au mai fost efectuate de asemenea teste si pentru cazul modificarii ulterioare a acestor informatii. S-a testat faptul ca grupa sangvina nu poate fi resetata pe cand celelalte campuri (ex. alergiile) pot suferi modificari. Mecanismul care impiedica folosirea incorecta, este inainte de toate, implementat in aplicatia off-card, pentru a permite utilizatorului completarea corecta a campurilor necesare, dar a fost implementat si testat si la nivelul appletului de pe smart card.

TRIMITERI MEDICALE SI FISE RADIOLOGICE ELECTRONICE

Un scenariu mai complex include urmatorul ciclu:



FIGURĂ 9-6 SCENARIU DE TESTARE

Acest test include cateva etape. Pentru inceput, pacientul se adreseaza cu cateva probleme medicale doctorului de familie care, neputand preciza un diagnostic precis, face o trimitere medicala catre un radiolog. In acel moment, legatura cu baza de date locala si baza de date centralizata nu este posibila si prin urmare trimiterea medicala electronica va fi salvata pe cardul pacientului.

In momentul prezentarii pacientului la medicul radiolog, acesta citeste de pe smart card trimiterea doctorului de familie, o verifica folosind certificatul acestuia si o incarca apoi in sistemul bazelor de date. Radiologul executa apoi o examinare radiologica pentru a descoperi exact conditia medicala a pacientului si pentru a intelege problema acestuia. Fisa radiologica emisa va fi salvata atat in baza de date locala si centralizata, cat si pe cardul pacientului. In momentul intoarcerii pacientului la medicul de familie, acesta consulta fisa radiologica si observatiile facute de medicul radiolog si determina diagnosticul precis al pacientului.

Se poate observa ca acest scenariu de test cuprinde aproape orice aspect al aplicatiei. Mai intai, pentru ca trimiterea sa fie completata, trebuiesc citite datele pacientului de pe card. Apoi, este testata scrierea si citirea corecta a trimiterii medicale pe smart card. Apoi este testata sincronizarea care trebuie sa existe intre continutul cardului si cel al bazelor de date; in acest moment, trimiterea neexistand in baza de date, ea va fi salvata folosind inregistrarea existenta pe cardul pacientului. In final, se testeaza si eliberarea unei fise radiologice pentru o examinare efectuata la medicul radiolog si pentru determinarea unui diagnostic precis.

10 CONCLUZII SI PERSPECTIVE

Smart cardurile disponibile la ora actuala sunt din ce in ce mai complexe iar tehnologia lor este in continua dezvoltare, lucru care deschide din ce in ce mai multe oportunitati de aplicatii in domeniu. Dincolo de capabilitatea lor de purtatoare sigure de informatii, smart cardurile au capacitati de procesare importante care impreuna cu suportul criptografic, le transforma in unelte extrem de utile in domeniul medical in care protectia datelor este esentiala.

Implementarea modelului experimental dezvoltat in cadrul acestui proiect destinat sistemului medical pune in evidenta caracteristicile de baza ale smart cardurilor mentionate anterior. Folosirea lor permite stocarea unui istoric medical radiologic ce se poate dovedi extrem de util in luarea unor decizii privind examinarile viitoare, datorita faptului ca se poate evalua in orice moment cantitatea de radiatii prezenta in organismul pacientului. In acest context, smart cardurile ofera un mediu de transport sigur si eficient al documentelor medicale precum trimiteri, diagnostice si fise radiologice in format electronic.

Pentru implementarea aplicaţiilor si serviciilor din sistemul informatic SRSPIRIM şi pentru a asigura o performanţa deosebita a întregului ansamblu s-au avut în vedere, următoarele caracteristici tehnice:

- arhitectură incipientă pe două niveluri, cu extensie la trei niveluri (three-tier), deschisă, compusă din baze de date şi servere de gestiune a bazelor de date, servere si aplicatii web si appleti oncard.

- baze de date relaţionale, gestionate de un server flexibil : MySql Server.

- asigurarea portabilităţii sistemului informatic, din punct al tehnologiilor si limbajelor (JavaCard/JavaScript/Java/HTML/PHP), cât şi a sistemului de operare utilizat (Windows/Linux)

- securitate, modularitate, scalabilitate, fiabilitate si interactivitate; se vor implementa mecanisme on-line şi off-line pentru accesul la date.

Documentul de fata prezinta detaliile legate de specificatiile, proiectarea, implementarea si testarea aplicatiilor off-card si on-card dezvoltate in cadrul unei plaforme experimentale bazata pe smart carduri si infrastructuri PKI destinata sistemului medical. Documentul contine o descriere a specificaţiilor software ale platformei informatice SRSPIRIM precum si detalierile acestora pentru fiecare categorie de aplicaţii, astfel:

- Infrastructura generala a sistemului cu definirea celor mai importante componente si ai principalilor sai actori. A fost apoi specificat, pentru fiecare actor in parte un rol bine precizat impreuna cu drepturile si sarcinile sale. In cadrul sistemului au fost implementate următoarele roluri: PACIENT, MEDIC DE FAMILIE, MEDIC RADIOLOG, OPERATOR CARDURI. Rolurile din cadrul sistemului sunt informaţii prezente in certificatele actorilor fiind semnate de autorităţile de certificare si integrate in logica aplicaţiilor off-card si on-card.

- Detalierea specificaţiilor software pentru fiecare componenta a infrastructurii de chei publice PKI folosita in cadrul proiectului SRSPIRIM: autoritatea de certificare (principala si subordonata), autoritatea de înregistrare, autoritatea de validare online a certificatelor si autoritatea de marca temporala.

- Elaborarea specificatiilor si proiectarea aplicaţiilor software care rezida pe smart carduri. Mai intai au fost descrise specificatiile functionale, datele care trebuiesc stocate pe smart card si dreptul de acces al fiecarui rol la aceste date. Pornind de la aceste specificatii si de la principiile si mecanismele de comunicatie cu appletii JavaCard, au fost concepute principalele clase din cadrul appletului. A fost detaliat apoi pentru fiecare clasa care sunt campurile stocate si modul acestora de reprezentare din punct de vedere al memoriei. Urmeaza apoi detalierea mecanismului de autentificare intre carduri folosind un protocol de tip challenge-response si certificatele prezente pe cele doua carduri. In final, au fost prezentate interfetele catre aplicatiile off-card.

- Proiectarea bazelor de date centrale şi locale. A fost prezentata mai intai arhitectura generala a bazelor de date din cadrul sistemului SRSPIRIM, urmand apoi a fi detaliata structura bazei de date centrale si relatiile dintre tabelele componente. Pentru fiecare tabel in parte, a fost apoi detaliata structura acestuia prin precizarea fiecarui camp, a tipului de date ales, si a cheilor si indecsilor folositi in acel tabel.

- Dezvoltarea celorlalte aplicaţii off-card din cadrul sistemului si anume: aplicatia de inregistrare a pacientilor si a medicilor in sistem, aplicaţia de personalizare a cardurilor radiologice si profesionale, aplicaţia de vizualizare si personalizare a fisei medicale a pacientului, aplicaţia de completare a unei trimiteri medicale, aplicatia de completare a unei fise radiologice, aplicatia de creare a rapoartelor catre Ministerul Sanatatii. Au mai fost de asemenea detaliate si modalitatile de comunicare cu smart cardurile, de asigurare a securitatii datelor folosind biblioteci specifice pentru aceste operatii.

- Integrarea şi testarea componentelor de bază ale sistemului SRSPIRIM. In aceasta activitate a avut loc configurarea si instalarea aplicatiilor dezvoltate in proiectul SRSPIRIM, plecand de la bazele de date locale si centrale, realizand personalizarea si incarcarea appletilor pe carduri, trecand apoi la instalarea servelelor web si a serviciilor necesare functionarii infrastructurii PKI. Au fost efectuate apoi o serie de teste asupra functionarii corecte a appletilor on-card folosind unelte din kitul de dezvoltare Oberthur. La final, au fost proiectate o serie de scenarii relevante pentru sistemul informatic in ansamblu si fiecare dintre acestea a fost testat.

Din punct de vedere al utilzatorului final, implementarea descrisa in cadrul acestui proiect ofera o interfata grafica intuitiva, prin intermediul careia acesta poate sa scrie sau sa citeasca date de pe smart card in confromitate cu o politica de securitate bazata de roluri.

Implementarea realizata a pus in mod pregnant accentul pe minimizarea memoriei utilizate pe smart card, lucru realizat prin stocarea in multe situatii a unor identificatori din bazele de date in locul unor informatii textuale extinse. Proiectul a mai pus de asemenea in evidenta si completarea eficienta a modurilor de lucru online si offline prin intermediul bazelor de date locale oferind astfel mai multa flexibilitate si toleranta la defectari intregului sistem.

Implementarea descrisa in cadrul acestui document nu a fost realizata in scopul de a acoperi nevoile unui sistem medical din lumea reala. Ea constituie numai un pilot capabil sa demonstreze faptul ca un astfel de sistem SRSPIRIM poate fi implementat la scara nationala si poate sa aduca o serie de avantaje clare in domeniul medical. O masura mai clara asupra eficientei proiectului va fi stabilita in urma unor teste si discutii aprofundate care vor avea loc in ultima etapa de testare intensiva a sistemului informatic. Trecerea catre o implementare reala presupune desigur si discutii aprofundate cu actorii principali din domeniul sanatatii, evidentei populatiei si nu in ultimul rand cu factorii de decizie economici.

11 BIBLIOGRAFIE

CRYPTSOFT. (2013). *PKCS#11: Cryptographic Token Interface Standard*. http://www.cryptsoft.com/pkcs11doc/

GLOBAL PLATFORM. (2013). *Card specification 2.2.1*. http://www.globalplatform.org/specificationscard.asp

NETBEANS. (2010). *Java Card Development Quick Start Guide*. https://netbeans.org/kb/docs/javame/java-card.html

ORACLE. (2007). *About Java Card Technology*. www.oracle.com/technetwork/java/javame/javacard/overview/about/index.html

PAN AMERICAN HEALTH ORGANIZATION. (2003). *Integrated circuit health data cards (smart cards)*. http://www.scribd.com/doc/12390280/Health-cards

RSA LABORATORIES. (2013). *Pkcs 11 base functionality v2.30: Cryptoki*. http://www.cryptsoft.com/pkcs11doc/STANDARD/pkcs-11v2-30b-d5.pdf

RYCOMBE. (2013). *FIPS 140-2 Overview*. http://www.rycombe.com/short140.htm

WEBSTORE. (2013). *International Organization for Standardization and International Electrotechnical Commission. Identification cards*. http://webstore.iec.ch/

WIKIPEDIA. (2013). *ISO/IEC 7816*. http://en.wikipedia.org/wiki/ISO/IEC_7816