****

**Kauno technologijos universitetas**

Informatikos fakultetas

**Biometrinės saugos priemonės**

Informacinių technologijų saugos metodai (T120M151)

Atliko:

IFM-1/3 gr. studentas

Eligijus Kiudys

2021 m. spalio 27 d.

Priėmė:

Asist .Šatkauskas Nerijus

**Kaunas, 2021**

Turinys

[Paveikslų sąrašas 4](#_Toc85978655)

[1. Biometrinių duomenų skenavimo laboratorinis darbas 5](#_Toc85978656)

[1.1. Įvadas 5](#_Toc85978657)

[1.2. Naudojamų priemonių analizė 5](#_Toc85978658)

[1.3. Realizacijos aprašymas 6](#_Toc85978659)

[1.4. Dalinės išvados 9](#_Toc85978660)

[2. Elektroninės kortelės 10](#_Toc85978661)

[2.1. Įvadas 10](#_Toc85978662)

[2.2. Naudojamų priemonių analizė 10](#_Toc85978663)

[2.3. Realizacijos aprašymas 10](#_Toc85978664)

[2.4. Dalinės išvados 10](#_Toc85978665)

Paveikslų sąrašas

[1 pav. Eikon piršto antspaudo skaitytuvas 5](#_Toc85978511)

[2 pav. BSAPI bibliotekos inicializavimas ir sesijos sukūrimas 6](#_Toc85978512)

[3 pav. Dviejų pirštų antspaudų nuskaitymas 6](#_Toc85978513)

[4 pav. Laikino masyvo sukūrimas 6](#_Toc85978514)

[5 pav. Piršto anspaudo patikrinimas 7](#_Toc85978515)

[6 pav. Dviejų piršto antspaudų nuskaitymas ir išsaugoto piršto antspaudo patikrinimas 7](#_Toc85978516)

[7 pav. Dviejų piršto antspaudų nuskaitymas ir neišsaugoto piršto antspaudo patikrinimas 8](#_Toc85978517)

# Biometrinių duomenų skenavimo laboratorinis darbas

## Įvadas

Užduotis: 1. Sukurti keletą antspaudų šablonų, surašyti į masyvą, nuskaityti naują antspaudą ir patikrinti, ar yra masyve.

Saugaus autentifikavimo problema iškyla vis dažniau ir dažniau, kadangi naudotojai nesaugiai naudoja autentifikavimo įrankius, kaip slaptažodžius ar kitokius panašaus veikimo autentifikacijos būdus. Biometrinis autentifikavimo būdas yra vienas iš saugesnių ir patogesnių kadangi naudotojui nereikia nieko prisiminti ir naudotojo veidas ar ranka yra autentifikavimo raktas. Laboratorinio darbo metu buvo išanalizuotas pirštų antspaudo biometrinis autentifikavimo būdas.

Darbo tikslas – sukurti programą kuri naudoja biometrinį pirštų antspaudo autentifikavimo būdą, išsaugo nuskenuotus piršto antspaudus ir patikrina ar nuskenuotas piršto antspaudas yra tarp išsaugotų.

Iškelti laboratorinio darbo uždaviniai:

* Nuskaityti kelis piršto antspaudus ir juos išsaugoti.
* Nuskaityti autentifikuojamo piršto antspaudą ir patikrinti ar jis yra išsaugotų antspaudų sąraše.

## Naudojamų priemonių analizė

Laboratorinio darbo metu naudojamas Eikon piršto antspaudų skaitytuvas.

1 pav. Eikon piršto antspaudo skaitytuvas

Norint nuskaityti piršto antspaudą naudojant šitą skaitytuvą reikia braukti pirštą iš viršaus į apačia kelis kartus. Piršto braukimas per skaitytuvą turi būti lėtas ir vienodas, kitaip skaitytuvui nepavyks nuskaityti piršto antspaudo. Nuskaičius piršto anspaudą jis yra išsaugomas masyve arba tikrinamas ar jau toks piršto antspaudas egzistuoja. Biometrinio įrankio naudojimui yra sukurta programa naudojant C++ programavimo kalbą.

## Realizacijos aprašymas

Pirmiausia inicializuojame BSAPI biblioteką bei sukuriame nauja sesiją kuri naudoja „Eikon“ piršto antspaudo skaitytuvą.

Text

Description automatically generated

2 pav. BSAPI bibliotekos inicializavimas ir sesijos sukūrimas

Sukūrus sesija su „Eikon“ įrankiu yra nuskaitomi du piršto antspaudai kurie yra saugomi masyve. Saugomu pirštų antspaudas yra sukuriamas programos pradžioje.

A screenshot of a computer

Description automatically generated with medium confidence

3 pav. Dviejų pirštų antspaudų nuskaitymas

Nuskaičius tris piršto antspaudus yra sukuriamas laikinas pirštų antspaudų masyvas kurio pagalba yra tikrinamas nuskaitytas piršto antspaudas.

Text

Description automatically generated

4 pav. Laikino masyvo sukūrimas

Išsaugojus piršto antspaudus laikiname masyve yra nuskaitomas piršto antspaudas ir tikrinamas ar nuskaitytas piršto antspaudas sutampa su vienu iš laikino masyvi pirštų antspaudų.

A screenshot of a computer screen

Description automatically generated with medium confidence

5 pav. Piršto anspaudo patikrinimas

Paleidus sukurtą programą yra nuskaitomi dveji pirštų antspaudai. Nuskaičius ir įrašius antspaudus į masyvą programa nuskaito ir tikrina kitus nuskaitomus pirštų antspaudus. Programa skaito pirštų antspaudus ir juos tikrina kol ji yra išjungiama.

Paleidžiame programą ir nuskaitome du pirštų antspaudus.

Text

Description automatically generated

6 pav. Dviejų piršto antspaudų nuskaitymas ir išsaugoto piršto antspaudo patikrinimas

Nuskaičius antspaudus patikriname vieną piršto antspaudą ar jis yra išsaugotas. Nuskaičius piršto antspaudą programa gražina kad tai yra pirmasis išsaugotas piršto anspaudas.

Paleidžiame programą iš naujo ir vėl nuskaitome du pirštų antspaudus. Pirmame tikrinime buvo galima tikrinti daugiau nei vieną kartą, bet buvo norma pasirinkti kitus pirštus testavimui.

Text

Description automatically generated

7 pav. Dviejų piršto antspaudų nuskaitymas ir neišsaugoto piršto antspaudo patikrinimas

Nuskaičius piršto antspaudą kuris nebuvo išsaugotas išspausdino rezultatą, kad antspaudo nėra sąraše.

## Dalinės išvados

Laboratorinio darbo metu susipažinau su biometrinio identifikavimo metodu, naudojant piršto skenavimą. Biometrinių duomenų nuskaitymui buvo naudojama Eikon biometrinių duomenų skaitytuvas. Sukurta programa kuri naudoja minėtą skaitytuvą buvo naudojama išsaugoti du piršto anspaudus. Išsaugojus antspaudus buvo nuskenuoti pasirinkti pirštai, kurie yra tikrinami ar jie yra tarp išsaugotų piršto antspaudų. Programa buvo realizuota sklandžiai, prijungtas įrenginys buvo iškarto aptiktas. Programa veikė be nesklandumų. Realizuojant programą buvo naudojama BSAPI biblioteka kuri suteikė galimybę naudotis Eikon biometrinių duomenų skaitytuvu.

# Elektroninės kortelės

## Įvadas

Užduotis: Sukurti kortelės ir kliento programas, kurios imituotų keletą viešojo transporto kortelės operacijų (pvz. mėnesinio bilieto pirkimas ir tikrinimas, vienkartinių bilietų pirkimas, naudojimas ir tikrinimas).

Saugaus autentifikavimo problema iškyla ne viena naudojant internetinėmis svetainėmis ar išmaniais telefonais, bet ir kitų paslaugų modernizavime ar pastatų modernizavime. Žmonės statant naujus pastatus nori juos modernizuoti, kad būtų lengviau ir patogiau pastatu naudoti. Vienas iš tokių pavyzdžių būtų viešbučio kambario atrakinimas. Kitas pavyzdys yra iš paslaugų sektoriaus modernizavimo. Bankinių kortelių modernizavimas, autobuso bilietų modernizavimas ir t.t. Tokie minėti pavyzdžiai reikalauja saugios ir paprastos technologijos kuri padėtų naudoti minėtuose pavyzdžiuose. Vienas iš sprendimo būdų yra naudoti apsaugotas RFID korteles kurios atlieka specifinius veiksmus. Laboratorinio darbo metu yra analizuojamas RFID kortelės panaudojimas.

Darbo tikslas – sukurti programą kuri naudoja RFID kortelę, ši kortelė bus naudojama atlikti viešojo transporto operacijoms kurios yra: mėnesinio bilieto pirkimas, valandinio bilieto pirkimas, pasirinktuose miestuose. Programėlė taip pat galės patikrinti ar pasirinktame mieste naudotojas turi galiojantį bilietą. Bilietus galima pirkti su programėlėje esamais kreditais, kurie taip pat gali būti pridedami.

Iškelti laboratorinio darbo uždaviniai:

* Įrašyti nupirktų bilietų informaciją į kortelę ir redaguoti kortelėje esančių kreditų kiekį.
* Įrašyti kreditų informaciją į kortelę.
* Nuskaityti kortelės duomenis ir parodyti naudotojui
* Patikrinti ar naudotojas turi galiojantį bilietą.

## Naudojamų priemonių analizė

Laboratorinio darbo metu naudojama išmanųjį telefoną kuris palaiko NFC funkcionalumą ir RFID kortelę į kurią galima įrašyti ir nuskaityti norimus duomenis.



Pav. 8 Išmanusis telefonas su NFC funkcionalumu

`

pav. 9 RFID „Mifare“ 1k kortelė

Norint naudoti RFID kortelę reikia pridėti šią kortelę prie telefoną norint atlikti pasirinktą veiksmą arba reikia pridėti prie pasirikto kortelės skaitytuvo veiksmo atlikimui. Pasirinkus veiksmą ir pridėjus kortelę prie skaitytuvo yra modifikuojami duomenys pagal parašytą programą. RFID kortelės naudojimui yra sukurta programa naudojant Kotlin programavimo kalbą ir Android Studio programos kūrimo aplinką.

## Realizacijos aprašymas

Sukonfigūruojamas projekto failas kuriame yra nurodoma, reikiami leidimai naudoti NFC technologiją bei yra nurodyti pradiniai veiksmai kuriuos aplikacija turės pritaikyti. Taip pat yra nurodoma į NFC technologijų aprašymo failą.

Text

Description automatically generated

pav. 10

Kitoje nuotraukoje yra aprašytos technologijos kurios gali būti nuskaitomos naudojant NFC technologiją (pav. 11). Sukurtoje programoje yra naudojama tik MifareClassic kortelės tipas.

Text

Description automatically generated

pav. 11

Sukurtos aplikacijos dizaino kodas. Pirmiausia buvo sukurtas pasirenkamasis sąrašas kuriame galima pasirinkti miestą. Tada buvo sukurtas mygtukas kurio pagalba galima tikrinti bilietus.

Text

Description automatically generated

pav. 12

Sukūrus pirmąją juostą buvo sukurta ir antroji juosta skirta valdymui. Šioje kodo dalyje yra aprašytas teksto langas ir mygtukas, teksto langas yra naudojamas mėnesinio bilieto kainai nurodyti, o mygtukas yra skirtas mėnesinio bilieto pirkimui.

Text

Description automatically generated

pav. 13

Sekančioje kodo dalyje yra aprašytas teksto langas ir mygtukas, teksto langas yra naudojamas valandinio bilieto kainai nurodyti, o mygtukas yra skirtas valandinio bilieto pirkimui.

Text

Description automatically generated

pav. 14

Likusioje dalyje yra sukurtas teksto įvesties langas kur galima nurodyti kiek norima pridėti kreditų. Šalia jo yra sukurtas mygtukas kurio pagalba galima pridėti kreditus. Tada yra teksto langas kuris parodo turimų kreditų kiekį. Paskutinis langas yra skirtas parodyti informacija kaip, kada reikia pridėti kortelę prie telefono arba nupirktus bilietus.

Text

Description automatically generated

pav. 15

Nuotraukoje matome naudojamas bibliotekas ir aprašytus naudojamus kintamuosius (pav. 16).

Text

Description automatically generated

pav. 16

Paleidimo funkcijos pirmoji dalis. Šioje dalyje yra parodytas elementų gavimas ir jų priskyrimas kintamiesiems (pav. 17).

A screenshot of a computer

Description automatically generated with medium confidence

pav. 17

Mygtukų funkcionalumo dalis. Yra aprašomi visi mygtukai kurie yra sukurtame ui, jų veiksmai, ką jie turi daryti (pav. 18).

Text

Description automatically generated

pav. 18

OnCreate funkcijos paskutinėje dalyje yra inicializuojamas NFC bibliotekos.

Text

Description automatically generated

pav. 19

Gaunamas NFC valdiklis kuris naudojamas NFC inicializavimui.

Text

Description automatically generated

pav. 20

Sukūrus „OnCreate“ funkcija kuri yra skitą NFC inicializavimui, mygtukų aprašymui, ui elementų gavimui ir kintamųjų priskyrimui, galima pradėti rašyti likusį programos kodą. Jis yra skirtas pačios programos veikimui, kaip duomenų nuskaitymui ir įrašymui į ir iš kortelės.

Aprašyta funkcija nuotraukoje (pav. 21) yra paleidžiama, kai RFID kortelė yra pridedama prie telefono galo. Ši funkcija iškviečia ir atlieka funkcijas pagal naudotojo pasirinktą funkciją, tai gali būti nuo bilieto pirkimo ir kreditų pridėjimo.

Text

Description automatically generated

pav. 21

Bilieto pirkimo kodas, kurio pagalba galima nusipirkti bilietą ir jį įrašyti į naudojama RFID kortelę. Bilieto funkcijai reikia kortelės skaitytuvo valdikio, kurio pagalba galime įrašyti duomenis ir juos nuskaityti. Reikia nurodyti kurioje vietoje yra saugomi bilietų duomenys, tada reikia nurodyti kokio tipo bilietą naudotojas pirks ir galiausiai bilieto kainą. Pirmiausia naudojamas kontroleris autentifikuoja kortelės sektorių duomenų skaitymui ir rašymui tada kortelės duomenys yra nuskaitomi ir pagal tai yra nusprendžiama ar galima pirkti bilietą ar ne. Jei galima pirkti bilietą, nupirkto bilieto duomenys yra įrašomi į kortelę ir yra nuskaičiuojama bilieto kaina, kuri yra atnaujinama kortelėje (pav. 22).

Text

Description automatically generated

pav. 22

Atsidarius aplikaciją ir nuskaičius kortelę turi būti parodomi kortelės duomenis, ką būtent ši funkcija ir daro (pav. 23). Į šią funkciją yra paduodamas kortelės kontroleris kurio pagalba bus nuskaitomi duomenys. Šitoje funkcijos dalyje yra patikrinama data nustatant rėžius, kurie padeda nuspręsti ar galima spausdinti bilietą. Toliau matome, kad bus spausdinami dviejų tipų bilietai. Prieš pradedant spausdinti bilietus reikia autentifikuoti sektorių pagal spausdinamus bilietus. Autentifikavus matoma, kad šioje dalyje yra Kauno bilieto spausdinimas.

Text

Description automatically generated

pav. 23

Antroje kodo dalyje (pav. 24) yra spausdinami likusių miestų bilietai.

Text

Description automatically generated

pav. 24

Kreditų skaitymui ir jų pridėjimui yra naudojamos dvi atskiros funkcijos „readCredits“ ir „addCredits“ (pav. 25). Sukurtų funkcijų pagalba galima nuskaityti esamą kreditų kiekį ir jį redaguoti naudojant NFC kontrolerį. Funkcijos veikia paprastai. Pirmoji funkcija autentifikuoja sektorių kuris saugo kreditų duomenis ir iš jo nuskaito kreditų kiekį. Antroji funkcija veikia panašiai tik ji autentifikavus sektorių paverčia pasirinktą kreditų kiekį į baitus ir juos įrašo į sektorių kuris saugo kreditus.

Text

Description automatically generated

pav. 25

Kreditų mažinimui yra naudojama „decreaseCredits“ funkcija kuri leidžia sumažinti pasirinktą kreditų kiekį (pav. 26). Kontrolerio pagalba yra autentifikuojamas trečias sektorius, kuriame yra saugomas kreditų kiekis. Autentifikavus sektorių yra nuskaitomas kreditų kiekis ir iš jo atimamas pasirinktas kiekis kreditu. Sumažinus kreditų kiekį jis yra išsaugomas į tą pačią vietą kortelėje ir yra atnaujinamas naujas kreditų kiekis naudotojo sąsajoje.

Text

Description automatically generated

pav. 26

Bilietų tikrinimui yra panaudojama „checkSelectedCity“ funkcija (pav. 27). Ši funkcija tikrina ar pasirinktame mieste naudotojas turi galiojantį bilietą. Funkcijoje yra autentifikuojami du sektoriai, kadangi viename yra mėnesiniai bilietai, kitame yra valandiniai bilietai. Sektorių duomenys yra nuskaitomi ir jų pagalba yra patikrinama, ar bilieto galiojimo laikas nėra pasibaigęs, arba galiojimo laikas neperžengia metų ribos.

Text

Description automatically generated

pav. 27

Nuotraukoje parodytos funkcijos (pav. 28) yra skirtos konvertuoti baitus į long tipo kintamuosius ir int tipo kintamuosius. Kitos dvi funkcijos yra skirtos konvertuoti int ir long tipo kintamuosius į baitus.

Text

Description automatically generated

pav. 28

Rašymo ir skaitymo funkcijos (pav. 29) yra svarbiausios tokio tipo programoje, kadangi skaitymas ir rašymas yra pagrindiniai elementai darbui su RFID kortelėmis. Primoji funkcija yra skirta baitų rašymui į kortelę. Pirmiausia yra autentifikuojamas pasiriktas sektorius ir tada pasirikti duomenys yra įrašomi į pasirinkto sektoriaus pasiriktą bloką. Antroji funkcija yra skirta kortelės nuskaitymui. Pasirenkamas blokas ir sektorius kuris bus nuskaitomas. Autentifikavus sektorių pasiriktas sektoriaus blokas yra atspausdinamas ir gražinamas nuskaitytas rezultatas.

Text

Description automatically generated

pav. 29

Yra dar keletas funkcijų, kurias reikia aprašyti (pav. 30). Šios funkcijos yra pagrindinės funkcijos, kurių pagalba veikia NFC skenavimas pridėjus kortelę prie telefono galo. Pirmoji funkcija „onNewIntent“ yra funkcija kuri aptinka kortelę pridėjus prie telefono galo ir iškviečia pagrindinę funkciją kurios pagalba visos kitos funkcijos susijungia. Kitos funkcijos, „onResume“ ir „onPause“, atlieka paprastesnius veiksmus, kaip programos tęsimas po pauzės arba programos pauzė, tęsimas po pauzės leidžia pratęsti RFID kortelių skenavimą ir programos naudojimą. Pauzės funkcija išjungia RFID kortelių nuskaitymą.

Text

Description automatically generated

pav. 30

Paskutinėje nuotraukoje yra matomas, kaip atrodo sukurta aplikacija (pav. 31). Šioje aplikacijoje matome, kad galima pasirikti norimą miestą ir patikrinti ar naudotojas turi galiojantį bilietą, pasirinktame mieste. Galima nusipirkti tiek valandinį, tiek mėnesinį bilietą, pasirinktame mieste. Yra kreditai, kurių pagalba galima nusipirkti pasiriktą bilietą. Apačioje, kur yra daug laisvos vietos, kur yra atspausdinamas informacinis tekstas.

Graphical user interface, text, application

Description automatically generated

pav. 31

## Dalinės išvados