

Bezpečný systém pro mikroplatby s RFID kartami

Secure system for micropayments using RFID cards

Roman Došek

Bakalářská práce
2011



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
akademický rok: 2010/2011

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Roman DOŠEK**
Osobní číslo: **A08033**
Studijní program: **B 3902 Inženýrská informatika**
Studijní obor: **Informační a řídicí technologie**

Téma práce: **Bezpečný systém pro mikroplatby s RFID kartami**

Zásady pro vypracování:

1. Implementujte zabezpečený systém pro mikroplatby, založený na technologii RFID.
2. Realizujte autentizaci karty a čtečky, která bude využívat šifrování 3DES.
3. Ukládejte na kartě čítadlo přístupů k jednotlivým položkám.
4. Podepisujte data na kartě pomocí šifry RSA s využitím privátního klíče, který zná jen čtečka a odmítněte komunikaci s kartou, jejíž RSA podpis je chybný.
5. Vytvořte SQL databázi, do které budou zaznamenávány veškeré transakce.

Rozsah bakalářské práce:

Rozsah příloh:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

1. GLOVER, Bill. RFID essentials. Beijing ; Sebastopol, CA : OReilly, 2006. xiii, 260 s. : ISBN 0-596-00944-5 (brož.). ISBN 978-0-596-00944-1.
2. THORNTON, Frank. RFID Security. Rockland, MA : Syngress Publishing, 2006. 242 s. : ISBN 1-59749-047-4. ISBN 978-1-59749-047-4.
3. BLANCHETTE, Jasmin; SUMMERFIELD, Mark. C++ GUI Programming with Qt 4 (2nd Edition). [s.l.] : Prentice Hall, 2008. 752 s. ISBN 978-0132354165.
4. PRATA, Stephen. Mistrovství v C++. 2, aktualiz. vyd. Brno : Computer Press, 2004. 1006 s. ISBN 8025100987

Vedoucí bakalářské práce:

Ing. Tomáš Dulík

Ústav informatiky a umělé inteligence

Datum zadání bakalářské práce:

25. února 2011

Termín odevzdání bakalářské práce:

7. června 2011

Ve Zlíně dne 25. února 2011

prof. Ing. Vladimír Vašek, CSc.
děkan



prof. Ing. Vladimír Vašek, CSc.
ředitel ústavu

ABSTRAKT

Cílem této práce bylo vytvořit zabezpečený systém pro mikroplatby. V teoretické části jsou popsány dostupné možnosti a vlastnosti technologie RFID a bezkontaktních karet. V praktické části je popsán návrh a implementace programu včetně použitých knihoven a vývojových nástrojů.

Klíčová slova: RFID, GUI, mikroplatby, Qt, bezpečnost

ABSTRACT

The goal of this thesis is to design and implement secured system for micropayments. In theoretic part are described available options and properties of RFID technology and contactless smart cards. The practical part describes the design and implementation of the program, including used libraries and development tools.

Keywords: RFID, GUI, micropayments, Qt, security

Chtěl bych poděkovat vedoucímu mé práce Ing. Tomáši Dulíkovi za rady a připomínky a za možnost pracovat na takovém zajímavém tématu. Mé díky patří i mé rodině a přátelům za podporu během studia.

“Theory is when you know something, but it doesn't work. Practice is when something works, but you don't know why. Programmers combine theory and practice: Nothing works and they don't know why. “

- Unknown

Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně

.....
podpis diplomanta

OBSAH

ÚVOD	9
I TEORETICKÁ ČÁST	10
1 RFID	11
1.1 Co je to RFID	11
1.2 Výhody	11
1.3 Fyzické charakteristiky tagů.....	12
1.4 Napájení	13
1.5 Komunikace.....	14
1.5.1 Operační frekvence.....	14
1.5.2 Komunikační mód	15
1.5.3 Modulace.....	15
1.5.4 Kódování	15
1.5.5 Párování.....	15
2 BEZKONTAKTNÍ CHYTRÉ KARTY	18
2.1 ISO/IEC 14443A - MIFARE	18
II PRAKTICKÁ ČÁST	20
3 NÁVRH APLIKACE	21
3.1 Zjednodušené schéma programu.....	21
3.2 Databáze	22
3.2.1 Tabulky.....	22
3.2.2 Komunikace s databází.....	23
3.3 Rozhraní programu.....	23
4 POUŽITÉ KNIHOVNY	25
4.1 Požadavky na použité knihovny	25
4.2 Qt framework.....	25
4.3 libnfc	26
4.4 libfreefare	27
5 POUŽITÉ VÝVOJOVÉ NÁSTROJE	28
6 IMPLEMENTACE	29
6.1 Datová část	29
6.1.1 Třída Customer	29
6.1.2 Třída Item.....	29
6.1.3 Třída Basket.....	29
6.2 Komunikace se čtečkou a kartou.....	29
6.2.1 Třída CardReader	30
6.2.2 Třída CardDetector	30
6.2.3 Třída SmartCard	30
6.3 Rozhraní programu.....	30
6.3.1 Hlavní okno.....	31
6.3.2 Další dialogy	32
ZÁVĚR	35

ZÁVĚR V ANGLIČTINĚ.....	36
SEZNAM POUŽITÉ LITERATURY.....	37
SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	38
SEZNAM OBRÁZKŮ	39
SEZNAM TABULEK.....	40
SEZNAM PŘÍLOH.....	41

ÚVOD

V současné době se stále více prosazují platební systémy s využitím bezkontaktních karet, které zvyšují uživatelský komfort a rychlost plateb. Velká část těchto systémů ale neobsahuje žádné, nebo jen malé zabezpečení proti krádeži, zkopírování karty nebo manipulaci s údaji na kartě. Cílem této práce je navrhnout a naprogramovat systém pro mikroplatby (tedy pro platbu malých částek – např. za občerstvení nebo vstupenky), který bude možné snadno nasadit a který bude obsahovat zabezpečení, které zamezí zneužití karet.

K tomuto účelu budou využity bezkontaktní karty MIFARE DESFire, které jsou zabezpečené symetrickým TripleDES šifrováním a obsahují 4kB paměti. V systému nebude využíváno výrobcem vložené ID karty, ale na kartě bude pro identifikaci a ověření uloženo 32b ID zákazníka spolu s dalšími informacemi, které zaručí autenticitu karty.

I. TEORETICKÁ ČÁST

1 RFID

1.1 Co je to RFID

RFID (Radio Frequency Identifier) je termín popisující jakékoliv identifikační zařízení, které využívá ke komunikaci rádiové frekvence nebo změny magnetického pole.

Hlavní dvě komponenty RFID jsou tag, což je identifikační zařízení, které uchovává informace a čtečka (Reader), pomocí které můžeme tyto informace přečíst.

RFID má velmi široké možnosti využití a zasahuje i do oblastí, ve kterých donedávna převažovaly jiné technologie. Lze je využít jak pro značení věcí, tak i lidí a zvířat.

Kde lze nalézt RFID:

- *v obchodech* – využití pro označení zboží a ochranu proti krádeži
- *v zásobování* – sledování zásilek a automatické sledování stavu skladu
- *značkování zvířat* – pro výzkumné účely nebo při chovu
- *přístupové systémy* – parkovací systémy, řízení vstupu do budov
- *platební systémy*

RFID tagy v některých oblastech začaly nahrazovat univerzálně používané čárové kódy, protože mají proti nim několik výhod, které mohou značně šetřit čas při identifikaci velkého množství zboží. Pro přečtení tagu není nutný přímý dohled, je možné číst i přes překážky a nezáleží na natočení předmětu. Zároveň jsou v některých oblastech vhodnější než jiné formy identifikace, jako jsou magnetické karty, které je nutné pro přečtení správně zastrčit do čtecího zařízení, zatímco u RFID stačí s kartou (která může být i v peněžence) přejet nad čtečkou nebo dokonce jen projít skrz čtecí rám, který sám přečte obsah karty. RFID čtečky navíc zpravidla zvládnou číst více tagů současně. RFID tagy dále mohou využívat šifrování, díky kterému není možné obsah jednoduše přečíst nebo zkopírovat neoprávněnou osobou a nabízejí se jak přepisovatelné, tak nepřepisovatelné varianty.

1.2 Výhody

- *Není důležité umístění* – čtení tagu nevyžaduje přímou viditelnost. Díky tomu lze ušetřit čas, který by bylo nutné vynaložit pro natáčení věcí jako u čárových kódů.
- *Čtení více tagů zároveň* – čtečky zvládají číst více tagů zároveň. Díky tomu čas nutný k přečtení velkého množství tagů výrazně klesne.

- *Velké množství typů* – RFID tagů existuje veliké množství – od výbuchu odolných tagů o velikosti krabice od bot po pasivní tagy o velikosti zrnka rýže. Díky tomu je lze použít ve velkém množství prostředí.
- *Přepisovatelnost* – některé typy tagů mohou být mnohokrát přepsány, což může být výhodné například u označování přepravních kontejnerů, jejich obsah se mění, nebo u některých platebních systémů, kde je nutné obsah karty měnit. Zároveň jsou ale k dispozici tagy, které je možné zapsat pouze jednou, které najdou použití tam, kde by možnost přepisu pouze zhoršovala bezpečnost.

Díky tomu, že jsou RFID zařízení hojně využívána, investuje se do jejich vývoje velké množství prostředků a na trh přicházejí další a další možnosti jak je využít, a zároveň se snižuje i jejich cena.

1.3 Fyzické charakteristiky tagů

Základním cílem RFID je připojit data k nějakému objektu data o něm. Každý tag má nějaký interní mechanismus k ukládání dat a možnost tato data vysílat. Ne každý druh RFID tagu má v sobě mikročip nebo zabudované napájení ale každý tag má cívku nebo anténu k vysílání a přijímání signálů. [1]

Protože RFID tagy musí fyzicky připojit data k objektům různých tvarů a velikostí a v různých prostředích, vyrábějí se ve velkém množství tvarů a velikostí. Navíc mohou být zabaleny v množství různých materiálů. [1]

Příklady různých tagů:

- PVC nebo plastové knoflíky a disky, zpravidla s dírou uprostřed pro připínání. Tyto tagy jsou odolné a znovupoužitelné.
- RFID tagy vypadající jako platební karty, které jsou nazývané „bezkontaktní chytré karty“
- Tagy vložené do vrstev papíru v nálepkách, nazývané „smart labels“. Tyto mohou být aplikované s automatickým aplikátorem podobným těm, jaké se používají pro nalepování čárových kódů.
- Malé tagy vložené v běžných předmětech, jako je oblečení, hodinky, náramky. Tyto malé tagy mohou být také ve formě klíčů nebo přívěsků na klíče.
- Tagy ve skleněných kapslích, které vydrží i v korozních prostředích nebo tekutinách.



Obrázek 1 - příklady RFID tagů [1]

1.4 Napájení

Další běžná kategorizace tagů je podle jejich zdroje napájení. To je také jeden z hlavních faktorů, který ovlivňuje cenu a životnost tagu. Pasivní tagy získávají všechnu svou energii nějakou metodou přenosu od čtečky. Aktivní tagy mají vlastní baterii, které napájí komunikaci, procesor a paměti. Tagy, které mají vlastní napájení integrovaného obvodu, ale pro komunikaci využívají energii od čtečky byly dříve také označovány jako aktivní, ale pro větší odlišení se vžilo označení semi-pasivní. Poslední typ tagu je schopný napájen nejen sebe, ale je schopný i komunikace s dalšími tagy svého druhu bez pomoci čtečky – tyto tagy se nazývají two-way tagy. [1]

Jak by se dalo očekávat, doplnění baterie zvyšuje cenu tagu, ale semi-pasivní a aktivní tagy mají několik výhod oproti pasivním tagům. V případě semi-pasivních může být čtecí vzdálenost delší, neboť se pro komunikaci může využít veškerá energie dodaná čtečkou a není nutné se o ni dělit s čipem. Aktivní tagy pak mohou mít extrémně dlouhou čtecí

vzdálenost a mohou vykonávat některé funkce bez čtečky, například zaznamenávat hodnoty z připojených senzorů.

1.5 Komunikace

Způsob, jakým tag komunikuje se čtečkou, nám umožňuje zjistit, na jakou vzdálenost lze tag přečíst a s kterými čtečkami lze komunikovat. Hlavní vlastnosti ovlivňující komunikaci zahrnují napájení, operační frekvenci, komunikační mód, modulace, kódování a párování.

1.5.1 Operační frekvence

Operační frekvence je elektromagnetická frekvence, kterou tag využívá ke komunikaci nebo k získání energie. Elektromagnetické spektrum, ve které obvykle RFID operuje je většinou rozděleno na nízké frekvence (LF), vysoké frekvence (HF), ultra vysoké frekvence (UHF) a mikrovlny. Protože RFID systémy vysílají elektromagnetické vlny, jsou regulovány jako rádiová zařízení. RFID systémy proto nemohou využívat frekvence, které již využívají jiné rádiové systémy, jako jsou rádia a televize.

Jméno	Frekvenční rozsah	ISM frekvence
LF	30 – 300 kHz	< 135 kHz
HF	3 – 30 MHz	6,78 MHz; 13,56 MHz; 27,125 MHz; 40,680 MHz
UHF	300MHz – 3 GHz	433,920 MHz; 869 MHz; 915 MHz
Mikrovlny	> 3 GHz	2,45 GHz; 5,8 GHz; 24,125 GHz

Tabulka 1 – frekvenční rozsahy využívané technologií RFID [1]

V praxi to znamená, že frekvence použitelné pro RFID jsou limitované na ty frekvence, které jsou definované jako Industrial Scientific Medical (ISM). Frekvence menší než 135 kHz nejsou ISM frekvence, ale v tomto rozsahu RFID systémy většinou využívají silná magnetická pole a komunikují na krátkou vzdálenost, takže rušení není takový problém.

Různé frekvence mají různé vlastnosti. Nízké frekvence lépe procházejí skrz vodu, zatímco vysoké frekvence mohou nést více informací. Vysoko frekvenční signály jsou také zpravidla snáze čitelné na větší vzdálenost.

Používané rozsahy pro UHF se v různých zemích světa liší. V Evropě, Jižní Americe a ve většině Asie operují UHF RFID tagy ve frekvencích od 865 MHz do 868 MHz. V Severní

Americe operují od 902 MHz do 928 MHz a v Indii se používá rozsah od 865 MHz do 867 MHz.

1.5.2 Komunikační mód

Další způsob odlišení tagů je podle toho, jestli mohou vysílat a přijímat zároveň. Stejně jako v kabelových přenosech, rádiová komunikace může být plně duplexní (FDX) nebo polo duplexní (HDX).

1.5.3 Modulace

Modulace popisuje, který atribut analogového nosiče - elektromagnetická vlna nebo pole – bude modulován pro reprezentaci nul a jedniček digitální zprávy. Existují tři hlavní typy modulací:

- *Amplitudová modulace* – Amplitude-shift keying (ASK) - typ modulace, při které se vysílají digitální data přes analogový nosič tak, že se mění amplituda vysílané vlny dle toho, zda vysíláme jedničku nebo nulu.
- *Frekvenční modulace* – Frequency-shift keying (FSK) - typ modulace, která zasílá data změnou frekvence vysílané vlny (nebo jak často nastává vrchol vlny).
- *Fázová modulace* – Phase-shift keying (PSK) - modulace, která zasílá data změnou fáze vysílané vlny.

1.5.4 Kódování

Kódování udává způsob, jakým budou čtečka a tag interpretovat změny v analogovém nosiči k reprezentaci digitálních dat. Kódování je tedy dohoda mezi vysílačem a přijímačem, co která změna znamená. Jako příklad kódování lze uvést Morseovu abecedu. Ta využívá pro reprezentaci čárky dlouhý tón a pro reprezentaci tečky krátký tón.

- *Biphase Manchester Encoding*
- *Pulse interval encoding*
- *Biphase space encoding*
- *Pulsed RZ encoding*
- *EPC Miller encoding*
- „1 of 256“ a „1 of 4“
- *FSK subcarrier encoding*

1.5.5 Párování

Párovací mechanismus tagu určuje způsob, jakým se obvod na tagu a obvod na čtečce vzájemně ovlivňují k zasílání a přijímání dat nebo energie. Typ použitého párování přímo

ovlivňuje čtecí vzdálenost mezi tagem a čtečkou. Různé čtecí vzdálenosti lze přibližně rozdělit jako blízké (do jednoho centimetru), vzdálené (jeden centimetr až jeden metr) a dálkové (více jak jeden metr). Spolu se vzdáleností výběr párovacího mechanismu silně ovlivňuje, které frekvence může tag využít.

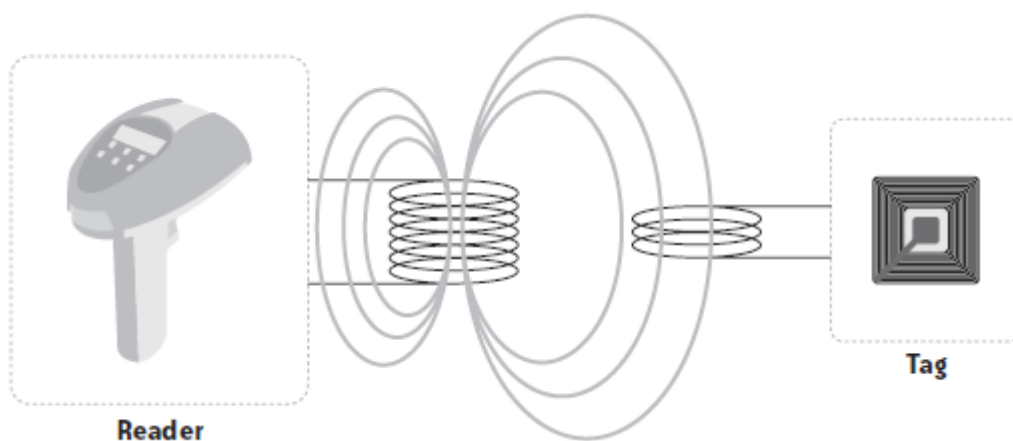
Párování Backscatter

Tento typ párování nabízí elegantní řešení, jak vyrobit RFID tag bez napájení. Princip tohoto párování je ten, že elektromagnetické vlny přijaté od čtečky jsou odraženy zpět a zároveň je do nich vložena požadovaná informace. Tagy přitom využívají stejnou frekvenci, ale mění jiné vlastnosti odráženého signálu pro zakomponování informace. Některé tagy tak činí na základě svých fyzických vlastností, jiné mají na anténu připojenou zátěž a jejím zapínáním a vypínáním mění odrazivost antény.

Protože čtečka a tag používají pro komunikaci stejnou frekvenci, musí se ve vysílání střídát. Jak bylo zmíněno dříve, tento komunikační mód se nazývá poloduplexní (half-duplex (HDX)). Čtečka napájí tag, i když právě nevysílá.

Induktivní párování

Induktivní párování je běžný příklad vzdáleného párování. Některé z nejběžnějších typů tagů v posledních letech jsou různé druhy induktivně párovaných tagů, včetně tagů, které následují standard ISO 15693 pro bezdotykové chytré karty. Čtečka napájí induktivně párované tagy využitím cívkové antény pro generování magnetického pole. Pole indukuje proud v cívce na tagu stejným způsobem, jako transformátor převádí energii mezi dvěma cívkami.



Obrázek 2 – ukázka induktivního párování RFID tagů [1]

Z tohoto důvodu je také tento typ párování často nazýván transformátorové párování. Pole dodává dostatek energie k napájení mikročipu, který poté může komunikovat se čtečkou. Při použití tohoto typu párování tag nepoužívá k odpovědi stejnou frekvenci.

Magnetické párování

Magnetické párování je párování na blízkou vzdálenost, které je podobné induktivnímu párování v tom, že cívka čtečky má kovové jádro, které je kulaté nebo ve tvaru U. Tag pak musí být ve vzdálenosti do 1 cm od čtečky a v místě, kde je v jádru mezera.

Tento typ párování je schopný napájet komplexní čipy. U tohoto typu párování je většinou nutné vložení tagu do čtečky, je proto vhodný pro chytré karty.

Kapacitní párování

Kapacitní párování je další forma párování, které funguje dobře na krátké vzdálenosti s tagem vloženým do čtečky. Je také stejně jako magnetické párování používána v chytrých kartách. Tagy s kapacitním párováním mají namísto antény elektrody. Čtečka i tag mají vodivá místa, která vytváří dohromady kondenzátor, pokud jsou bez dotyku přesně vedle sebe. Stejně jako magnetické párování i kapacitní párování je schopné napájet komplexní čipy.

2 BEZKONTAKTNÍ CHYTRÉ KARTY

Bezkontaktní chytré karty jsou karty s vestavěnými integrovanými obvody, které jsou schopné zpracovávat a ukládat data a komunikovat s čtečkou prostřednictvím radiových vln. Karty s čtečkou komunikuje prostřednictvím induktivního párování (s rychlostí přenosu od 106 po 848 kbps). Tyto karty vyžadují pouze blízkou vzdálenost ke čtečce k dokončení transakce. Jejich nejčastější využití je k platbám (například za veřejnou dopravu) nebo k identifikaci a řízení vstupu. V některé literatuře se tyto karty nepočítají pod technologii RFID, neboť jsou komplexnější než běžné RFID čipy a používají mírně odlišnou technologii přenosu a napájení (která je však na stejných základech). Pro účely této práce proto budu tyto karty brát jako podkategorii RFID zařízení.

Moderní bezkontaktní karty s čtečí vzdáleností do 10 cm jsou pokryté standardem ISO/IEC 14443. Tento standard se dále dělí na několik podtypů – A, B, apod. Všechny typy komunikují rádiově na frekvenci 13,56 MHz. Hlavní rozdíly mezi typy jsou v modulaci a kódování (popsáno v 2. části standardu) a inicializaci protokolu (popsáno v 3. části standardu). Přenosový protokol (popsaný v 4. části standardu) je pro všechny typy společný. Přenosový protokol definuje blokovou výměnu dat a související mechanismy. Dle tohoto standardu jsou také definovány následující názvy pro komponenty:

PCD (proximity coupling device) – je název pro čtečku

PICC (proximity integrated circuit card) – je název pro tag/kartu

Dále existuje standard ISO/IEC 15693 pro vzdálené karty, které umožňují čtečí vzdálenost až 1 - 1,5m.

2.1 ISO/IEC 14443A - MIFARE

MIFARE Classic

Jedná se o jeden z nejstarších a zároveň světově nejvyužívanějších MIFARE čipů. Výrobce uvádí, že byla prodána více než 1 miliarda těchto čipů a více než 200 milionů je nyní v provozu, což znamená přibližně 80% všech bezkontaktních karet. Vyrábí se o kapacitě 1 - 4 kB. Obsahuje hardwarově implementovaný neveřejný šifrovací algoritmus CRYPTO1 a generátor náhodných hodnot. Použitá šifra byla již před několika lety prolomena a karty je možné přečíst i bez znalosti klíče. K tomu přispěl i fakt, že použitý generátor náhodných hodnot se při zapnutí napájení inicializuje vždy na stejnou hodnotu, díky čemuž byl

CRYPTO1 výrazně oslaben. Tento typ karet je využíván v hromadné dopravě, řízení vstupu, bezkontaktních platebních systémech a dalších. Systémy, které tuto kartu využívají, jsou zranitelné – kartu lze zkopírovat, nebo z ní přečíst citlivé údaje. [4][5]

MIFARE Plus

Tato karta byla vytvořena jako nástupce MIFARE Classic. Je s ní zpětně kompatibilní, což umožňuje využít stávající čtečky a infrastrukturu, ale navíc nabízí kromě CRYPTO1 možnost využití AES šifrování, což výrazně zvyšuje její bezpečnost. Obsahuje opravený generátor náhodných čísel, který již není náchylný na časované útoky. [4][6]

MIFARE DESFire

Další typ karty podobný MIFARE Classic s většími hardwarovými a softwarovými možnostmi zabezpečení. Je již od výrobce naprogramována s vlastním DESFire operačním systémem, který nabízí jednoduchou adresářovou strukturu. K prodeji jsou 4 varianty této karty – jedna s 3DES šifrou a kapacitou 4 kB a novější varianta MIFARE DESFire EV1, která dává na výběr z algoritmů DES/2K3DES/3K3DES/AES a kapacitami 2, 4 a 8 kB. [4][7]

MIFARE Ultralight

Nejlevnější varianta MIFARE čipů, nepodporuje žádné šifrování, má velmi omezenou kapacitu (384b) a nízkou přenosovou rychlost (106 kbps). Použitelné jako tikety ve veřejné dopravě například jako jednosměrné jízdenky, jízdenky na omezený počet jízd, víkendové jízdenky, apod., dále jako vstupenky na stadiony, výstavy, a další. Lze ji velmi snadno zkopírovat a přečíst, je proto nevhodná k ukládání jakýchkoliv dlouhodobějších nebo citlivých dat. Zároveň však tomu odpovídá i její cena, která začíná na 0,25\$. [4][8]

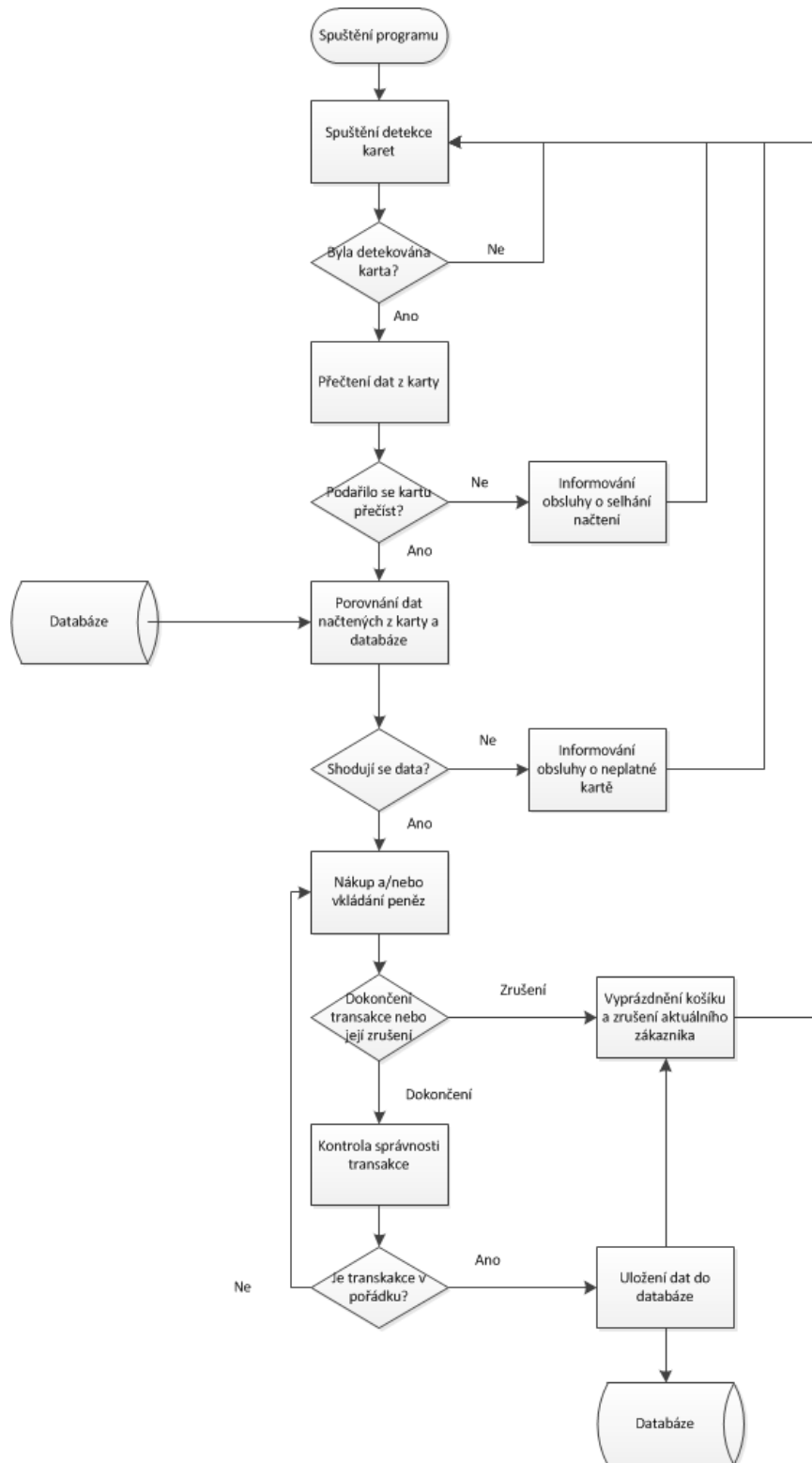
MIFARE Ultralight C

Zabezpečená varianta tagu MIFARE Ultralight, která navíc nabízí ochranu dat 3DES šifrováním a ochranu proti kopírování. [9]

II. PRAKTICKÁ ČÁST

3 NÁVRH APLIKACE

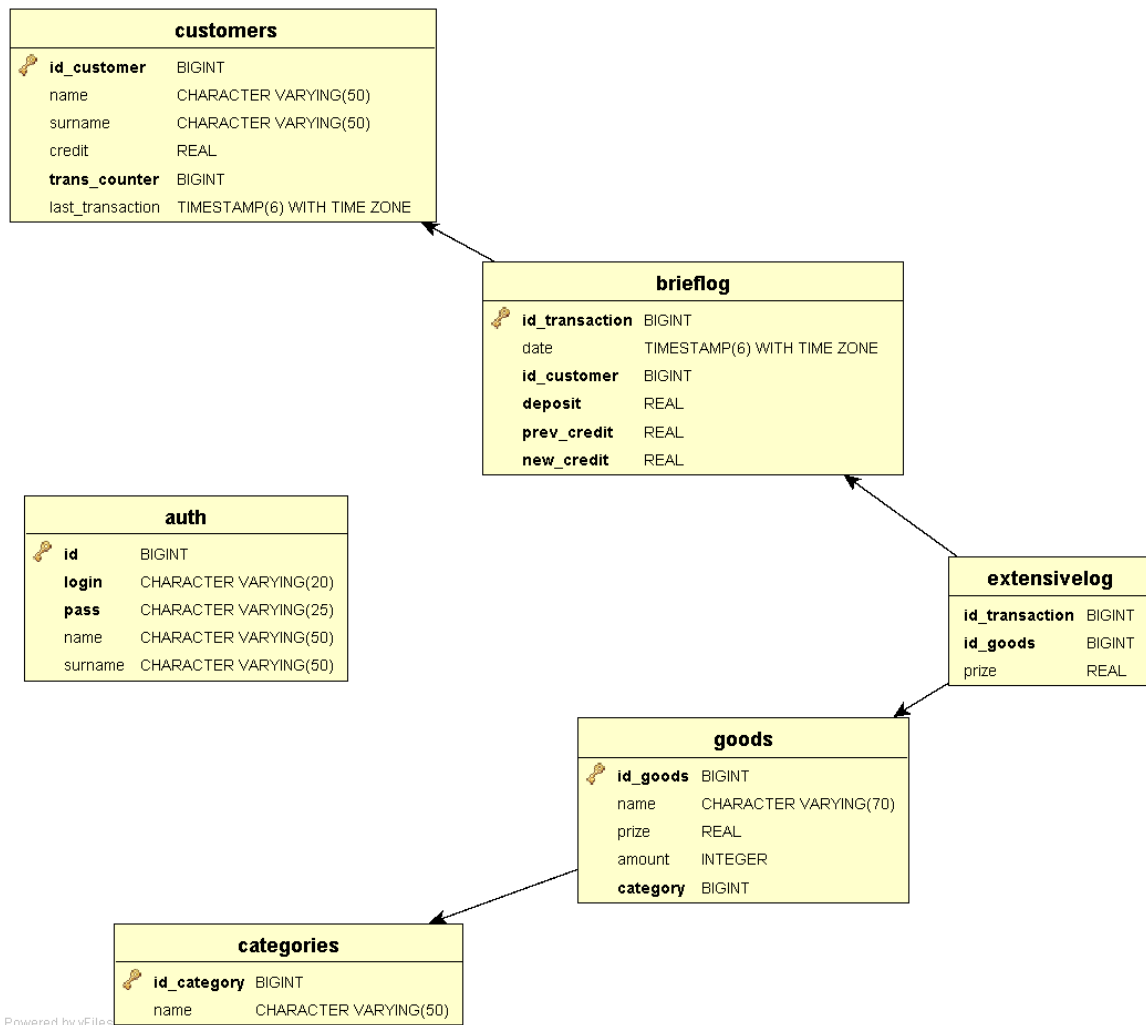
3.1 Zjednodušené schéma programu



Obrázek 3 – zjednodušený vývojový diagram programu

3.2 Databáze

Pro ukládání dat aplikace byla zvolena databáze. V současnosti se využívá 5 provázaných tabulek, plus jedna určená k autentifikaci obsluhy aplikace. Je možné použít libovolnou databázi, která podporuje transakce a je podporována v Qt – např. MySQL, PostgreSQL, IBM DB2 a další.



Obrázek 4 – schéma databázových tabulek

3.2.1 Tabulky

- *Customers* – tato tabulka obsahuje unikátní ID zákazníka, jeho jméno, příjmení, výši aktuálního kreditu, čítač který obsahuje počet transakcí, který zákazník provedl a datum a čas jeho poslední transakce. Transakcí se zde myslí přiložení karty ke čtecímu terminálu – není nutné nic koupit. Čítač transakcí a její datum a čas jsou do databáze ukládány z bezpečnostních důvodů a stejné informace jsou ukládány i na zákaznickou kartu. Díky tomu není možné jednoduše kartu buď

zkopírovat, nebo zazálohovat její obsah a obnovit, neboť poté již nebudou souhlasit tyto informace a systém kartu odmítne jako neplatnou.

- *categories* – pomocí této tabulky se zboží organizuje do skupin podle typu zboží – jídlo, nápoje, atp. Tabulka obsahuje pouze ID skupiny a její název.
- *goods* – v této tabulce je ukládáno ID zboží, jméno, cena a kategorie, do které zboží spadá.
- *auth* – s pomocí této tabulky je ověřována identita obsluhy programu. Vždy při zapnutí programu se obsluha musí přihlásit, a během práce je opět možné aplikaci uzamknout. Díky tomu nemůže s programem pracovat nepovolaná osoba. Tabulka obsahuje ID, login, heslo a jméno a příjmení obsluhy.
- *brieflog* – do této tabulky je ukládán každý nákup, případně vložení peněz. V tabulce je ID transakce, datum jejího provedení, ID zákazníka, kredit zákazníka před nákupem, množství vložených peněz a kredit po dokončení nákupu.
- *extensivelog* – v této tabulce se využívá ID transakce z tabulky *brieflog*, ID zboží a cena. Zároveň tato tabulka nemá vlastní primární klíč, slouží pouze k vytvoření M:N relace mezi transakcí a zbožím (V jedné transakci může být zakoupeno více zboží a zároveň to samé zboží může být zakoupeno i ve více transakcích).

3.2.2 Komunikace s databází

S databází se komunikuje prostřednictvím SQL dotazů s využitím jedné z částí knihovny Qt – QSql modulu. Ten nabízí připojení k databázi, zasílání SQL dotazů, získávání výsledků a mnoho dalšího. K tomu poskytuje jednotné rozhraní bez ohledu na použitou databázi, pouze vyžaduje ovladač databáze.

3.3 Rozhraní programu

Rozhraní programu je navrženo tak, aby umožňovalo snadnou orientaci a bylo intuitivní, a zároveň obsahuje i standardní prvky – stavový řádek a menu, které skrývá další funkce. Při návrhu byl hojně využíván prostředek Qt toolkitu pro rozvrhování rozhraní – Qt Layouts. Díky tomu se aplikace automaticky přizpůsobí na libovolné rozlišení obrazovky. Program je navíc kompletně stylizovatelný pomocí QSS (Qt Style Sheets), což je obdoba technologie CSS, která je hojně využívána na webu. Program lze tak snadno a rychle přizpůsobit prostředí, ve kterém bude nasazen – například zvětšení tlačítek a odstupů mezi nimi v případě použití dotykového displeje, doplnění loga zařízení, zmenšení/zvětšení písma, aniž by bylo jakkoliv nutné zasahovat do zdrojového kódu aplikace.

Přestože Qt nabízí možnost vytvářet dialogy prostřednictvím návrhového nástroje Qt Designer, byly všechny dialogy naprogramovány ručně pro vyšší přehlednost kódu a rychlost (generované dialogy mají mírně větší spotřebu paměti).

4 POUŽITÉ KNIHOVNY

Mezi nezbytné knihovny v této aplikaci patří knihovna pro tvorbu grafického uživatelského rozhraní (GUI) a knihovna pro manipulaci s čtečkou a kartami. Pro implementaci GUI bylo rozhodnuto pro knihovnu Qt, protože splňovala požadavky a má výbornou dokumentaci. V případě manipulace s čtečkou a kartou prakticky nebylo na výběr, protože v začátku práce existovala pouze jedna kombinace knihoven, která zvládla požadované věci. Těmito knihovnami je nízkoúrovňová knihovna libnfc a vysokoúrovňová libfreefare.

4.1 Požadavky na použité knihovny

- *Multiplatformnost* – z důvodu vyšší flexibility je nutné, aby program fungoval jak na počítačích s OS Windows, tak i s OS Linux nebo Mac OS X.
- *Vhodná licence* – je nutné použít knihovny s takovými licencemi, které nebudou omezovat využití programu v budoucnosti. Vhodné jsou licence GNU GPL, GNU LGPL a kompatibilní, nebo volnější – seznam kompatibilních licencí lze nalézt na webu GNU (<http://www.gnu.org/licenses/license-list.html>).
- *Jazyk C/C++* – celý program bude psán v C++, proto je vhodné, aby použité knihovny byly psány ve stejném jazyku.

4.2 Qt framework

Qt je jedna ze dvou nejpopulárnějších multiplatformních knihoven pro vytváření programů s grafickým uživatelským rozhráním.[2]

Qt toolkit byl vytvořen v roce 1999 společností Trolltech, která jej v roce 2008 prodala firmě Nokia. V březnu roku 2011 Nokia ohlásila prodej práv na provoz podpůrných služeb a prodej licencí pro komerční projekty vytvořené pomocí Qt společností Digia. Zároveň však Nokia ujišťuje, že po transakci zůstane hlavním vývojářem tohoto toolkitu. [2]

Od roku 1999 se Qt toolkit vyvinul v multiplatformní nástroj, ve kterém lze vyvíjet konzolové nebo GUI aplikace v odlišných programovacích jazycích pro různé platformy. Aplikace napsané s pomocí toolkitu je možno distribuovat pod licencí GPL, LGPL, nebo po splnění určitých podmínek i komerčně. [2]

Qt je knihovna programovacího jazyka C++, ale existuje i pro jazyky Python (PyQt), Ruby (QtRuby), C, Perl, Pascal, C#, Java (Jambi) a Haskell. Podporuje lokalizaci aplikací a také SQL, zpracování XML, správu vláken, přístup k souborům, práci s grafikou a multimédií.

Velkou výhodou je Qt je velmi přehledně zpracovaná dokumentace a také vývojové programy Qt Creator nebo Qt Designer. Aplikace vytvořené pro grafické uživatelské prostředí používají nativní vzhled operačního systému, takže vyvinuté aplikace se vždy přizpůsobí do používaného prostředí. [2]

Qt společně s GTK+ nahradila starší Motif. Důkazem kvality a rozšířenosti toolkitu budiž použití například pro projekty Skype, Google Earth, prostředí KDE, webový prohlížeč Opera, VirtualBox a jiné. [2]

4.3 libnfc

Libnfc je první volné NFC SDK a programátorské API uvolněné pod licencí GNU LGPL. Poskytuje kompletní průhlednost a použití bez licenčních poplatků pro každého jako opak ke stávajícímu RFID trhu, který je zaplněn proprietárním hardwarem a softwarem. [3]

Jedná se o nízkoúrovňovou knihovnu napsanou v C++, která umožňuje ovládat kompatibilní čtečky (jejich seznam lze nalézt na webu knihovny - <http://www.libnfc.org/documentation/hardware/compatibility>). Jejím prostřednictvím lze s kartami manipulovat na nízké úrovni. Knihovna takové obsahuje řadu ukázkových programů, které napomáhají rychlému seznámení s knihovnou.

4.4 libfreefare

Libfreefare je knihovna pro vysokoúrovňovou manipulaci s MIFARE kartami. Ke své práci využívá knihovnu libnfc. V současné době podporuje většinu MIFARE karet (s výjimkou Mini a Plus). Rovněž obsahuje ukázkové programy.

Karta	Stav podpory
MIFARE Classic 1k	Podporována
MIFARE Classic 4k	Podporována
MIFARE DESFire 2k	Podporována
MIFARE DESFire 4k	Podporována
MIFARE DESFire 8k	Podporována
MIFARE DESFire EV1	Podporována ve vývojové větvi
MIFARE Mini	Nepodporována (nedostupný hardware)
MIFARE Plus S 2k	Nepodporována (nedostupný hardware)
MIFARE Plus S 4k	Nepodporována (nedostupný hardware)
MIFARE Plus X 2k	Nepodporována (nedostupný hardware)
MIFARE Plus X 4k	Nepodporována (nedostupný hardware)
MIFARE Ultralight	Podporována
MIFARE UltralightC	Podporována ve vývojové větvi

Tabulka 2 – Typy karet podporované knihovnou libfreefare [10]

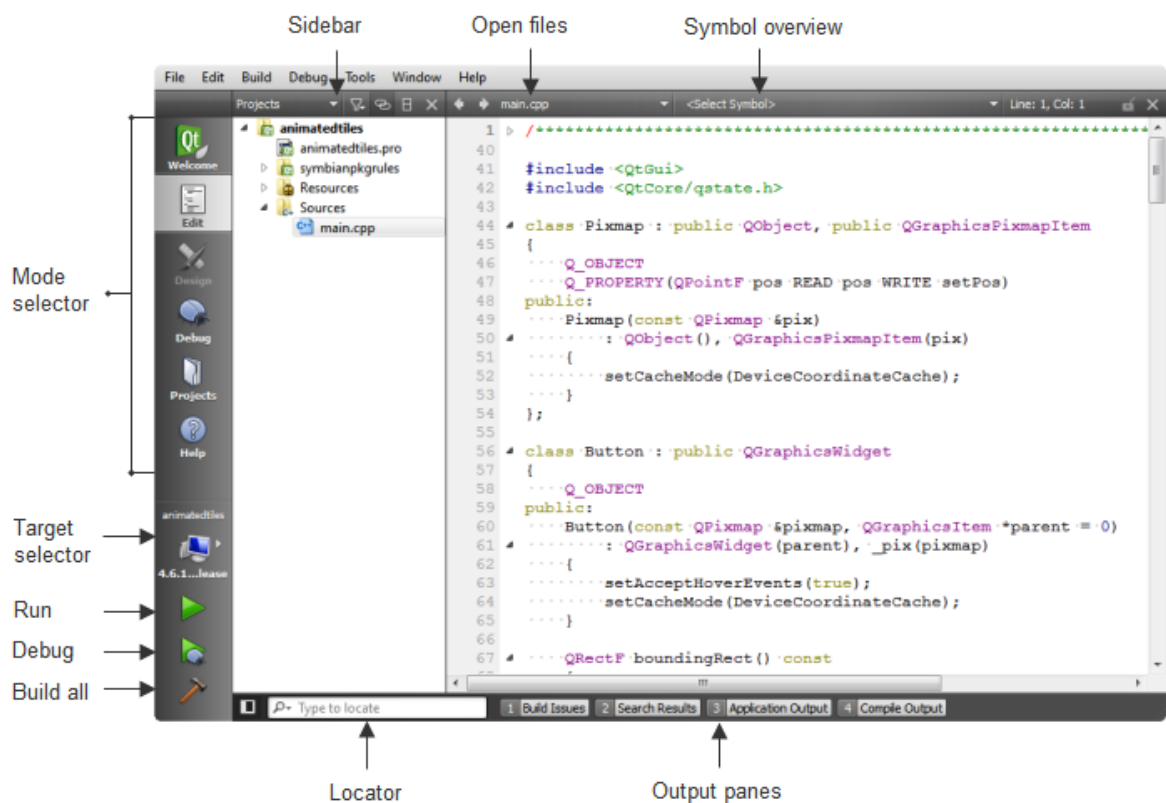
5 POUŽITÉ VÝVOJOVÉ NÁSTROJE

K vývoji celé aplikace byl používán nástroj Qt Creator, což je multiplatformní IDE dodávané jako součást Qt SDK.

Poskytuje:

- *Editor kódu v C++ a JavaScriptu*
- *Integrovaný návrhář rozhraní*
- *Nástroje pro správu projektu a sestavení*
- *Gdb a CDB debuggery*
- *Podpora systému pro správu verzí (SVN, Git, Mercurial,...)*
- *Simulátor mobilních rozhraní*
- *Podpora pro desktopové i mobilní platformy*

[11]



Obrázek 5 – Qt Creator 2.1 [11]

6 IMPLEMENTACE

Aplikaci lze logicky rozčlenit na několik částí, které by na sebe ideálně neměly mít vliv, pokud budou splňovat požadované rozhraní. Bylo by například možné vyměnit čtečku a karty za jiné s tím, že by se změnila pouze část programu, která se tím zabývá a do ostatních částí by se nemuselo vůbec zasahovat.

6.1 Datová část

V této části programu se uchovávají data z databáze a od uživatelů. Slouží k tomu několik tříd, které uchovávají data a zároveň umožňují pouze přesně omezenou manipulaci s nimi. Jmenovitě se jedná o třídy: Customer, Item a Basket.

6.1.1 Třída Customer

Tato třída uchovává údaje o zákazníkovi, obsahuje stejné údaje jako tabulka *customer* v databázi, jen má navíc proměnnou *deposit*, která dočasně uchovává množství vložených peněz – pro potřeby logování.

6.1.2 Třída Item

Opět se jedná o třídu kopírující databázovou tabulku, tentokrát tabulku *goods*. Oproti ní ale neobsahuje ID kategorie, do které položka patří, neboť se jedná o nadbytečnou informaci.

6.1.3 Třída Basket

Tato třída reprezentuje nákupní košík a má pouze jeden atribut a tím je *QList<Item> polozky*, což je lineární seznam instancí typu Item. Třída poskytuje několik metod pro manipulaci a zjišťování údajů o košíku: přidávání položek, mazání položek, vrácení položky na určité pozici v košíku, zjištění velikosti celého košíku, cenu všech položek v košíku a smazání všech položek.

6.2 Komunikace se čtečkou a kartou

Ke komunikaci se čtečkou a kartou se využívají již výše zmíněné knihovny *libnfc* a *libfreefare*. Přesto by jejich přímé použití v kódu snižovalo přehlednost a zvyšovalo obtížnost budoucích úprav. Proto jsou vytvořeny další třídy – třída čtečky, třída karty a třída pro detekci karet.

6.2.1 Třída CardReader

Tato třída v programu reprezentuje čtečku karet. Obsahuje metody pro nalezení čtečky a připojení k ní, zapnutí a vypnutí detekce karet, identifikaci karet přiložených ke čtečce a odpojení od čtečky. Ke své činnosti využívá knihovnu libnfc a libfreefare.

6.2.2 Třída CardDetector

Jedná se o speciální třídu, která se spouští v samostatném vlákně a má za úkol přepnout čtečku do tzv. poolovacího režimu a vyslat signál, pokud dojde k přiložení karty ke čtečce. Musí být spouštěna v samostatném vlákně, protože metoda knihovny libnfc, která způsobuje přepnutí čtečky do poolovacího režimu je blokující – to znamená, že k návratu z funkce dojde pouze při vyčerpání určitého počtu pokusů (které jsou předány jako parametr funkci) nebo při přiložení karty. Při přímém použití této funkce v hlavním vlákně programu by nebylo možné s ním jak programem interagovat. Tato třída je využívána ve třídě CardReader.

6.2.3 Třída SmartCard

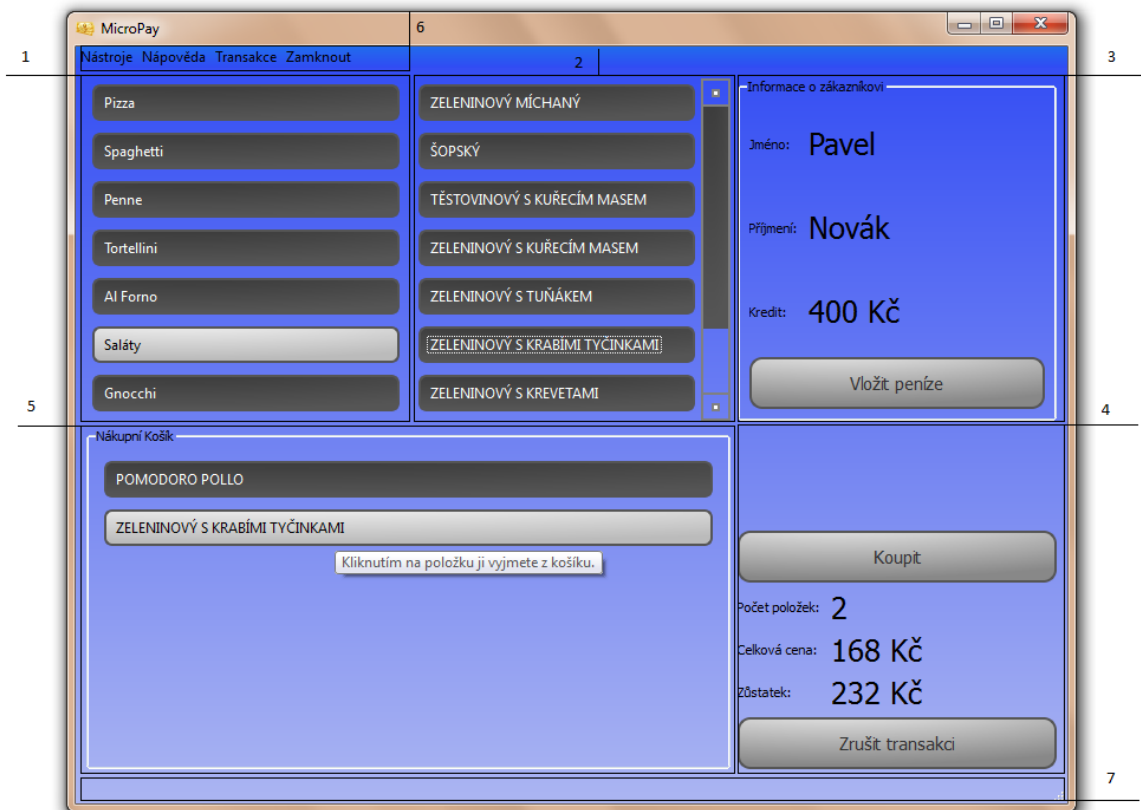
Třída uchovávající informace o kartě a umožňující z karty číst, zapisovat, provádět autentifikaci atd.

6.3 Rozhraní programu

Rozhraní programu je tvořeno třídou MainWindow, která dědí vlastnosti ze třídy QMainWindow. Tato třída se stará o vykreslování hlavního okna programu a zároveň spojuje všechny části programu do funkčního celku.

Kromě hlavního okna ještě aplikace obsahuje velké množství dialogů, které obstarávají další funkčnost.

6.3.1 Hlavní okno



Obrázek 6 – hlavní okno aplikace

- 1 – výběr kategorie zboží
- 2 – vkládání zboží do košíku
- 3 – informace o současném zákazníkovi
- 4 – informace o nákupním košíku
- 5 – nákupní košík
- 6 – menu aplikace
- 7 – stavový řádek

6.3.2 Další dialogy

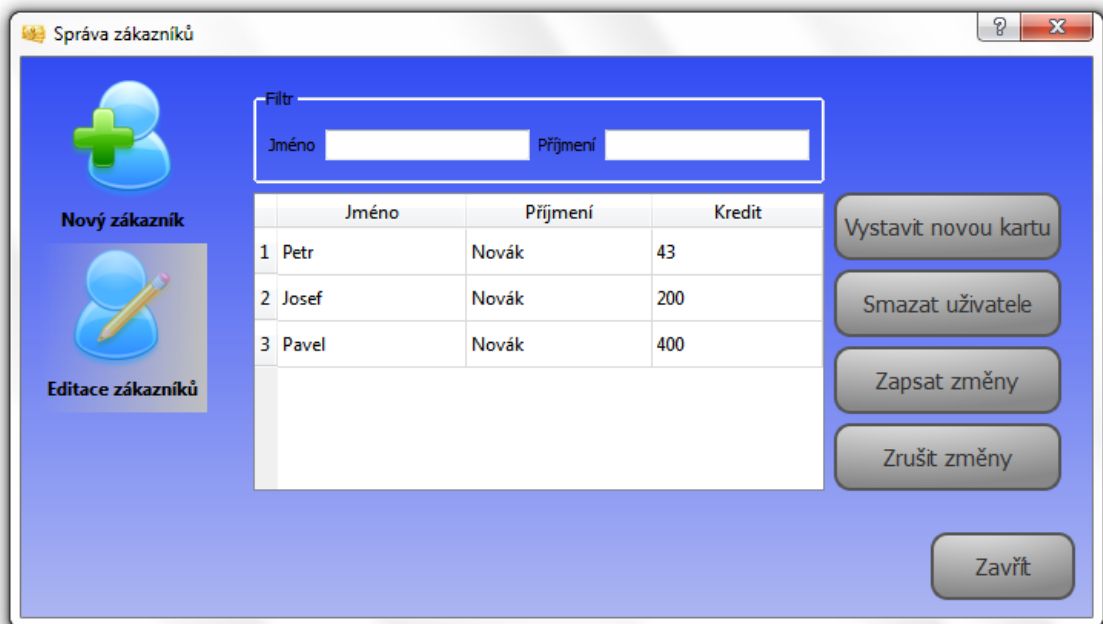
Nastavení

Pomocí tohoto dialogu lze ovlivňovat různé nastavení programu, v současné době je možné ovlivnit připojení k databázi, doby zobrazování některých upozornění a změnu vzhledu programu prostřednictvím QSS stylů.



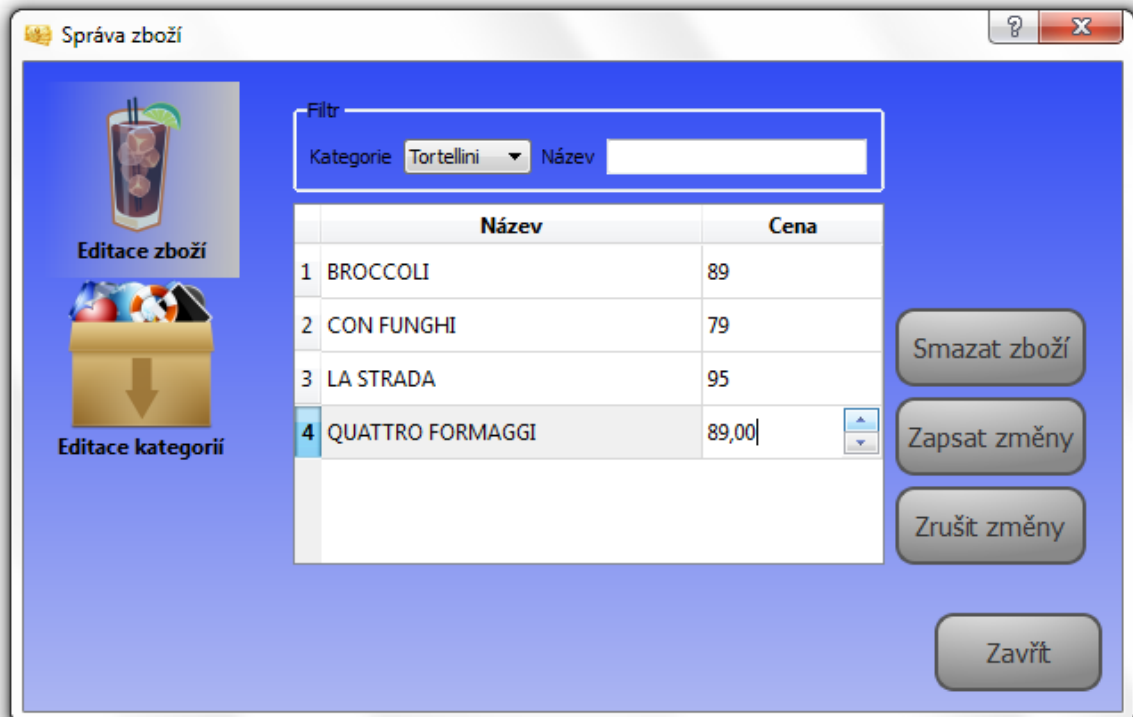
Obrázek 7 – nastavení programu

Správa zákazníků



Obrázek 8 – dialog pro správu zákazníků

Správa zboží a kategorií



Obrázek 9 – dialog pro správu zboží a kategorií

Prohlížení logu

Tento dialog kombinuje data ze všech tabulek a umožňuje zobrazit kdy, kde, co, za jakou cenu zákazník nakoupil a kolik přitom vložil peněz na účet. Tyto údaje lze zobrazit jak přímo pro aktuálního zákazníka, tak pro všechny a vyhledávat. Snadno lze tak sledovat pohyby financí na účtech nebo dokonce určit o jaké zboží je velký zájem a které nikdo nekupuje. Cena položky se nezískává z tabulky *goods* ale z tabulky *extensivelog*, neboť cena položek se může změnit a poté by již nesouhlasily záznamy.

	Datum a čas	Jméno a příjmení	Vklad	Předchozí kredit	Nový kredit
1	12.5.2011 19:23:09	Petr Novák	0	300	132
2	12.5.2011 19:23:16	Petr Novák	0	132	43
3	16.5.2011 17:54:11	Petr Novák	100	43	54

	Název	Cena
1	ZELENINOVÝ S KRABÍMI TYČINKAMI	79
2	POMODORO POLLO	89

Obrázek 10 – Log nákupů v MicroPay

ZÁVĚR

Jako součást této práce byla vytvořena funkční aplikace pro mikroplatby, která obsahuje řadu zabezpečujících prvků. Přes její funkčnost však zbývá vyzkoušet její nasazení v reálném prostředí a analyzovat potenciální slabiny. Díky použití karet MIFARE DESFire je celý obsah paměti šifrovaný a vzhledem k návrhu karty se klíč nikdy neposílá vzduchem, takže ho není možné odposlechnout. Zároveň karta i aplikační databáze obsahují pro každého zákazníka autentizační data (timestamp poslední transakce a počet transakcí provedených s kartou), takže ani při případném zkopírování karty by ji nebylo možné použít. Největší nebezpečí pro zákazníka tak spočívá v odcizení karty – v takovém případě je nutné si nechat vystavit novou kartu a/nebo zablokovat starou kartu. Teoreticky by bylo možné zabránit nedovolenému použití karty přidáním další ochrany v podobě PINu jako je tomu u platebních karet, ale tím by se ztratily výhody použití těchto karet. Vzhledem ke způsobu řešení programu nakonec nebylo využito podepisování karet pomocí RSA, neboť při každém čtení se data ověřují v databázi a použití RSA podpisu by více jak zdvojnásobilo dobu nutnou k přečtení dat z karty, to by pak mělo za následek větší množství neúspěšných čtení. RSA podpis by byl vhodný při přímém využívání dat na kartě, například při použití čítače kreditu bez ověření s databází. Ale i pak by bylo možné data na kartě zálohovat a obnovit.

Na aplikaci mám v úmyslu dále pracovat, neboť se jedná o velmi zajímavou oblast. Nejprve by bylo vhodné pročištění a zřehlednění zdrojových kódů – některé části programu by mohly být lépe navrženy. Další logický krok by byl přidat podporu pro další karty: MIFARE DESFire EV1 a MIFARE Plus, které využívají šifrování AES.

ZÁVĚR V ANGLIČTINĚ

As a part of this thesis was created application for micropayments, that contains a number of safeguards. Apart from its function, there still remains a need of field-testing this application and evaluation of potential threats. Thanks to use of MIFARE DESFire contactless smart cards, the entire card memory is encrypted and with regard to card design, encryption key is never transmitted between card and reader, so it can't be eavesdropped. Both card and application database contains authentication data from each customer (timestamp of last transaction and transaction counter), which ensures, that even if entire card could be copied, the copy would still be unusable. This way, the greatest danger for customer lays in dispossession of card – in which case, the customer must request new card or block the old one. Theoretically, this could be prevented by applying another layer of security, such as PIN used with classical credit cards, but that would mean losing almost all of contactless smart cards advantages. Because of program design, RSA signing of cards wasn't used at all. During every card reading all data are verified with respect to database and using RSA signing on card would just more than double read time, which would cause higher percentage of read failures. RSA signing would be prosperous if some of data on cards would be used directly, for example with credit counter without database verification. Even then, the data on card could be backed up and restored.

I intend to continue working on this application because this area of programming is very interesting. Initially a cleanup and refactoring of source codes would be advisable. The next logical step would be to add support for newer cards, such as MIFARE DESFIRE EV1 and MIFARE Plus, which utilizes AES encryption.

SEZNAM POUŽITÉ LITERATURY

- [1] GLOVER, Bill; BHATT, Himamsu. RFID essentials [online]. Beijing : O'Reilly, [cit. 2011-05-15]. 260 s. ISBN 0-596-00944-5.
- [2] Qt (knihovna). In Wikipedia : the free encyclopedia [online]. St. Petersburg (Florida) : Wikipedia Foundation, 24. 3. 2006, last modified on 10. 5. 2011 [cit. 2011-05-16]. Dostupné z WWW: <[http://cs.wikipedia.org/wiki/Qt_\(knihovna\)](http://cs.wikipedia.org/wiki/Qt_(knihovna))>.
- [3] Libnfc.org [online]. 2009-11-13 [cit. 2011-05-11]. Introduction. Dostupné z WWW: <<http://www.libnfc.org/documentation/introduction>>.
- [4] Mifare. In Wikipedia : the free encyclopedia [online]. St. Petersburg (Florida) : Wikipedia Foundation, 10 November 2005, last modified on 7 December 2007 [cit. 2011-05-12]. Dostupné z WWW: <<http://en.wikipedia.org/wiki/Mifare>>.
- [5] MIFARE.net [online]. c2011 [cit. 2011-05-12]. MIFARE Classic 1k. Dostupné z WWW: <<http://www.mifare.net/products/mifare-smartcard-ic-s/mifare-1k/>>.
- [6] MIFARE.net [online]. c2011 [cit. 2011-05-12]. MIFARE Plus. Dostupné z WWW: <<http://www.mifare.net/products/mifare-smartcard-ic-s/mifare-plus/>>.
- [7] MIFARE.net [online]. c2011 [cit. 2011-05-12]. Mifare DESFire EV1. Dostupné z WWW: <<http://www.mifare.net/products/mifare-smartcard-ic-s/mifare-desfire-ev1/>>.
- [8] MIFARE.net [online]. c2011 [cit. 2011-05-12]. MIFARE Ultralight. Dostupné z WWW: <http://www.mifare.net/products/mifare-smartcard-ic-s/mifare_ultralight/>.
- [9] MIFARE.net [online]. c2011 [cit. 2011-05-12]. MIFARE Ultralight C. Dostupné z WWW: <<http://www.mifare.net/products/mifare-smartcard-ic-s/mifare-ultralight-c/>>.
- [10] Nfc-tools [online]. 2010-08-12 [cit. 2011-05-12]. Libfreefare. Dostupné z WWW: <<http://code.google.com/p/nfc-tools/wiki/libfreefare>>.
- [11] Nokia Qt [online]. c2011 [cit. 2011-05-17]. Qt Creator IDE and tools. Dostupné z WWW: <<http://qt.nokia.com/products/developer-tools/>>.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

AES	Advanced Encryption Standard
API	Application Programming Interface
CSS	Cascading Style Sheets
DES	Data Encryption Standard
FDX	Full Duplex
GUI	Graphical User Interface
HDX	Half Duplex
HF	High Frequency
IDE	Integrated Development Environment
ISM	Industrial Scientific Medical
LF	Low Frequency
PCD	Proximity Coupling Device
PICC	Proximity Integrated Circuit Card
QSS	Qt Style Sheets
RFID	Radio Frequency Identifier
RSA	Rivest, Shamir, Adleman
SDK	Software Development Kit
SQL	Structured Query Language
UHF	Ultra High Frequency

SEZNAM OBRÁZKŮ

<i>Obrázek 1 - příklady RFID tagů [1]</i>	<i>13</i>
<i>Obrázek 2 – ukázka induktivního párování RFID tagů [1]</i>	<i>16</i>
<i>Obrázek 3 – zjednodušený vývojový diagram programu</i>	<i>21</i>
<i>Obrázek 4 – schéma databázových tabulek</i>	<i>22</i>
<i>Obrázek 5 – Qt Creator 2.1 [11]</i>	<i>28</i>
<i>Obrázek 6 – hlavní okno aplikace</i>	<i>31</i>
<i>Obrázek 7 – nastavení programu</i>	<i>32</i>
<i>Obrázek 8 – dialog pro správu zákazníků</i>	<i>33</i>
<i>Obrázek 9 – dialog pro správu zboží a kategorií</i>	<i>33</i>
<i>Obrázek 10 – Log nákupů v MicroPay</i>	<i>34</i>

SEZNAM TABULEK

<i>Tabulka 1 – frekvenční rozsahy využívané technologií RFID [1]</i>	<i>14</i>
<i>Tabulka 2 – Typy karet podporované knihovnou libfreefare [10]</i>	<i>27</i>

SEZNAM PŘÍLOH

- P I Zdrojové kódy aplikace
- P II Spustitelný program pro OS Windows
- P III Dokumentace programu
- P IV Ukázková databáze
- P V Použitý software

PŘÍLOHA P V: POUŽITÝ SOFTWARE

cmake	vývoj aplikace
DB Visualizer	tvorba schématu databáze
doxygen	tvorba dokumentace
GCC	vývoj aplikace
Memory Validator	odladování aplikace
Microsoft Visio 2010	tvorba diagramů
Microsoft Word 2007	sazba práce
pgAdmin III	správa databáze
PostgreSQL	databáze aplikace
qmake	vývoj aplikace
Qt Assistant	dokumentace ke Qt frameworku
Qt Creator	vývoj aplikace