MODERNIZATION OF SYSTEM FOR ELECTRONICS PAYMENTS

Vitalii Shyp

Master's thesis
2020

Tomas Bata University in Zlín
Faculty of Applied Informatics

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
Ústav informatiky a umělé inteligence

Akademický rok: 2019/2020

ZADÁNÍ DIPLOMOVÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení:	**Vitalii Shyp**
Osobní číslo:	**A16864**
Studijní program:	**N3902 Engineering Informatics**
Studijní obor:	**Information Technologies**
Forma studia:	**Prezenční**
Téma práce:	**Inovace systému elektronických plateb**
Téma práce anglicky:	**The Modernisation of an Electronic Payment System**

Zásady pro vypracování

1. Describe the state of the art in the area of electronics payments for merchants and their points of sale.
2. Analyze the code of the POS Merchant web portal for merchants.
3. Propose the possible improvements and functionalities for refactoring.
4. Implement the proposed innovations in the system.
5. Update the documentation where necessary.

Rozsah diplomové práce:
Rozsah příloh:
Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam doporučené literatury:

1. PARK, John, Steve MACKAY a Edwin WRIGHT. Practical Data Communications for Instrumentation and Control. Amsterdam ; London: Elsevier, 2003. ISBN 9780750657976.
2. MARTIN, Robert C. Čistý kód: [návrhové vzory, refaktorování, testování a další techniky agilního programování]. Brno: Computer Press, 2009. ISBN 978-80-251-2285-3.
3. RADU, C. Implementing Electronic Card Payment Systems. Artech House, 2002. ISBN: 978-1580533058
4. JOHNSON R. et al. Professional Java Development with the Spring Framework. John Wiley & Sons, 2007. ISBN: 9780764574832
5. WALLS, C. Spring in Action: Covers Spring 4. O'Reilly 2014. ISBN 978-1617291203

Vedoucí diplomové práce: **Ing. Tomáš Dulík, Ph.D.**
Ústav informatiky a umělé inteligence

Datum zadání diplomové práce: **28. listopadu 2019**
Termín odevzdání diplomové práce: **15. května 2020**

UNIVERZITA TOMÁŠE BATI VE ZLÍNĚ
Fakulta aplikované informatiky

L.S.

doc. Mgr. Milan Adámek, Ph.D.
děkan

prof. Mgr. Roman Jašek, Ph.D.
ředitel ústavu

Ve Zlíně dne 9. prosince 2019

I hereby declare that:

- I understand that by submitting my Diploma thesis, I agree to the publication of my work according to Law No. 111/1998, Coll., On Universities and on changes and amendments to other acts (e.g. the Universities Act), as amended by subsequent legislation, without regard to the results of the defence of the thesis.
- I understand that my Diploma Thesis will be stored electronically in the university information system and be made available for on-site inspection, and that a copy of the Diploma/Thesis will be stored in the Reference Library of the Faculty of Applied Informatics, Tomas Bata University in Zlin, and that a copy shall be deposited with my Supervisor.
- I am aware of the fact that my Diploma Thesis is fully covered by Act No. 121/2000 Coll. On Copyright, and Rights Related to Copyright, as amended by some other laws (e.g. the Copyright Act), as amended by subsequent legislation; and especially, by §35, Para. 3.
- I understand that, according to §60, Para. 1 of the Copyright Act, TBU in Zlin has the right to conclude licensing agreements relating to the use of scholastic work within the full extent of §12, Para. 4, of the Copyright Act.
- I understand that, according to §60, Para. 2, and Para. 3, of the Copyright Act, I may use my work - Diploma Thesis, or grant a license for its use, only if permitted by the licensing agreement concluded between myself and Tomas Bata University in Zlin with a view to the fact that Tomas Bata University in Zlín must be compensated for any reasonable contribution to covering such expenses/costs as invested by them in the creation of the thesis (up until the full actual amount) shall also be a subject of this licensing agreement.
- I understand that, should the elaboration of the Diploma Thesis include the use of software provided by Tomas Bata University in Zlin or other such entities strictly for study and research purposes (i.e. only for non-commercial use), the results of my Diploma Thesis cannot be used for commercial purposes.
- I understand that, if the output of my Diploma Thesis is any software product(s), this/these shall equally be considered as part of the thesis, as well as any source codes, or files from which the project is composed. Not submitting any part of this/these component(s) may be a reason for the non-defence of my thesis.

I herewith declare that:

- I have worked on my thesis alone and duly cited any literature I have used. In the case of the publication of the results of my thesis, I shall be listed as co-author.
- That the submitted version of the thesis and its electronic version uploaded to IS/STAG are both identical.

In Zlin; dated: ....

Student´s Signature

ABSTRACT

Various new technologies made our life way easier as for customers but to keep up with world progress, developing side should make service better. That is the aim of this project – deep explanation on real time example of improving merchant of terminals control.

With Java being one of the most popular languages among developers, hence the community and support for related technologies are outstanding. New people bring new ideas, therefore technologies. Concerning this fact, we can state that any type of development is progressing very quickly therefore modernization should be done in time.

In order to support for various new services such as Apple Pay, Google Pay etc., relatively new and convenient ways of wirelessly paying by card along with other features would be not possible unless modernization and updating took place.

ACKNOLEDGEMENTS

Acknowledgements, motto and a declaration of honor saying that the print version of the Bachelor's/Master's thesis and the electronic version of the thesis deposited in the IS/STAG system are identical, worded as follows:

I hereby declare that the print version of my Bachelor's/Master's thesis and the electronic version of my thesis deposited in the IS/STAG system are identical.

Foremost, I would like to express my sincere gratitude to my Supervisor Ing. Tomas Dulik, for the constant support of my Master study, for his knowledge, interest in the topic, patience and persistence. His advising helped me during all the time of research and writing of the thesis.

Beside my advisor, I would like to thank representatives of the company I am employed by “Monet+” company, my team leaders Tomas Jacko, Tomas Lascak, Jaroslav Gajdosik for provided data, support with explanations and education towards the topic.

I also would like to thank Head of Department of Informatics and Artificial Intelligence, FAI, TBU Prof. Dr. Roman Jasek, International Students Supervisor doc. Ing. Marek Kubalčík, Ph.D. and the rest of the thesis committee in particular Assoc. Prof. Ing. Jiří Vojtěšek, Ph.D, for their engagement, motivation, advises and guidance.

I thank my fellow classmates of Tomas Bata University: Denis Varaksin, Tatsuki Monji, Kompaneev Maxim, Walid Cheikh and Hasan Jon for cooperation, discussions and teamwork towards common goal.

My sincere thanks go to International Office of Tomas Bata University for continuous support during the whole period of studies in foreign country.

Last but not least, I would like to thank my family: My Grandmothers (Mrs. Nadia Shyp, Mrs. Svitlana Rybalko) and My Parents (Shyp Andrii and Shyp Liliya) and my uncle (Andrii Raykowski) for making my master’s education possible.

CONTENTS

INTRODUCTION

POS Merchant is a web portal from merchants who are customers of one of the Czech banks. On this terminal a customer could for example check the settling of terminals, payment getaways, request support and browse transactions done on those terminals.

It is a standard JAVA application mainly based on the libraries and technologies such as:

- Spring;
- Hibernate;
- Apache.

Dependencies are managed by Maven tool.

Main goal of this modernization is to replace the old generation of used libraries which should enable faster and more secure enhancement of this portal in the future. Part of this project is also to rewrite some of the critical parts of the portal by a new code style of code, which those new libraries enabled. For example:

Full rewrite the SQL database communication (MariaDB and Oracle):

- Modification of web components used in the portal.
- Improving communication between application modules.
- Rewrite communication with external modules to the REST interface.

By the end of the project, the 10 years progress in Java ecosystem and mentioned libraries and technologies can be observed.

I. THEORY

1 STATE OF ART

1.1 Introduction

Nowadays, purchasing goods and items is the necessary part of modern life. When people started using money as an exchange currency. That is where the trading becomes more secure, faster and more advanced. With introduction of receipts, the records of transactions became more useful for government and customers and tax authorities.

The point of sale (POS) in retail uses both software and hardware. This can include a manual or electronic cash register, scanners, scales, touch screen terminals, and a variety of other bits and beans [1][2][3].

An example of a point of sale in a conventional brick and mortar store is a scale used to weigh items in a traditional grocery or pastry shop.

Marketers emphasize POS for their products or services because consumers often make purchasing decisions at POS points of sale. Physical stores have traditionally placed their POS at exits to increase impulse purchases by shoppers leaving the store.

While Department Stores typically place POS in multiple locations through their stores, they have found that this provides an opportunity to focus on shoppers looking for specific types of products and influence their purchasing decisions early in the sales process.

The earliest POS systems were nothing more than written receipts, but retailers now prefer electronic POS solutions. They simplify the sales process as well as allow the merchant to collect valuable sales data.

A basic POS system can be no more than one cash register and software needed to collect sales data. They can be expanded and improved with additional hardware such as barcode scanners, memory card readers or other software modules.

Depending on the software, sellers can track not only their sales, but also inventory levels, gross income, profit margins, sales structure, and a host of other data points.

This data can then be analyzed to improve profitability and identify weaknesses in the sales or inventory process. Sophisticated software can even automate ordering and restocking and help entrepreneurs tailor their marketing campaigns based on consumer behavior.

1.2 History

The foundation of POS systems was years ago - technology was introduced to the world in the late 19th century when the first cash register systems started to be implemented. In the very beginning of its existing it was a machine that was able to keep track of the sales under a cashier and by the beginning of 20th century many businesses started to use the technology [2].

In the process of the development drawer and paper were added for an actual registration of the transactions and in the late 20th century LCD display screen, credit card, thermal printing additionally improved the POS technology [3]. Afterwards in the process of technology evolution computer software stated to be used in combination with an actual device which again extended the functionality of POS [1].

Ultimately POS were transformed and currently includes a wide combination of digital and IT technologies that is used by vast majority of businesses, detailed theoretical explanation of modern POS is given below.

1.3 Overview

In big businesses such as restaurants, grocery store, salons, shops there was always a need of secure and efficient system that can produce and print the receipts or sales overview. In modern world the purpose of the system is to ease and accurate the trade of goods and services. With new technologies the process has become automated providing more convenient and faster purchasing experience for customers. The system can hold the records of transactions in database which makes it way easier to analyze the business and keep track of sales. Point of Sale (POS) is a system or spot where customers can make a payment for provided services or goods that are offered by the salesman (by salesman it is implied company, corporations or any other organizational form). Most of the modern POS components contain [4]:

- scanner;
- card Reader;
- computer or tablet;
- keyboard or touch screen input;
- software on CPU for operations running;
- printer.

When a customer gets to the POS with desired goods, a cashier scans the items to be purchased, and POS system calculates the summary amount of the payment to be made [2]. A barcode scanner obtains the information about the items and gets the records with unique references to each item holding different values. While processing, upon cashier scanning the goods, a customer can observe the price per each item along with the summary of the scanned items. Customer has a choice to pay by Credit/Debit card, cash, or coupons. If the transactions are successful, the POS software updates the records in database, sometimes including number of items in stock. The customer receives a confirmation in a form of receipt for successful payment [5].

Most card terminals today transfer data over cellular networks and Wi-Fi. Obsolete terminals communicate over regular telephone lines or Ethernet connections [6]. Some also have the ability to cache transaction data, which will be sent to the gateway processor when a connection becomes available; the main disadvantage is that immediate authorization is not available at the time the card was processed, which can subsequently lead to failed payments. Wireless terminals transfer data from a card using Bluetooth, Wi-Fi, cellular, or even satellite networks in remote areas and on-board aircraft.

Prior to the development of payment terminals, merchants will capture card data manually (using a so-called "ZipZap" machine). The development of payment terminals has been driven by the efficiency benefits of reducing transaction processing times and immediately allowing payments. From a security point of view, terminals provide end-to-end data encryption and card end auditing functions. However, there are some cases of POS contact pad malware. There have also been instances of skimming at card terminals, and this has led to a move away from the use of magnetic stripe to collect information using EMV standards.

The payment terminal allows the merchant to grab the required credit and debit card information and transfer this data to the merchant's merchant service or to the bank for authorization and finally transfer the merchant's funds. The terminal allows a merchant or their customer to steal, insert, or hold a card near a device to collect information.[7] The terminal is often connected to the point of sale systems so that payment amounts, and payment confirmation can be transferred automatically to the retailer's management system. Terminals can also be used offline, in which the seller keys the quantity in the terminal before the customer present his card and personal identification number (PIN).

In order to use the system, some who has granted access (cashier, administrator, manager) for POS software must log be logged in. Some of the functionality. Further, the protected and confidential system management functions can only be accessed by administrators for security purposes. These operations are represented on figure 1.

1.4 Development and system requirements

For POS system development, developer can use any desired language that fits his needs the most, while POS terminal development has to be programmed only in POSXML.

POSXML (from POS and XML - markup language) - programming language, based on XML that uses to develop applications for POS-terminal.

The main purpose of POSXML is to simplify the development of applications for POS terminals [8]. It contains a set of commands and pre-programmed commands that allow direct interaction with the machine, resulting in a large reduction in application development time.

Windows Embedded Industry, formerly Windows Embedded POSReady and Windows Embedded for Point of Service (WEPOS), is an operating system subfamily developed by Microsoft as part of its Windows Embedded product family. Based on Windows NT, Windows Embedded Industry is intended for use in industrial devices such as cash registers, ATM machines, and self-service checkouts. Windows Embedded 8.1 Industry was the last release, with Windows 10 IoT Enterprise replacing both Windows Embedded Industry and Windows Embedded Standard.

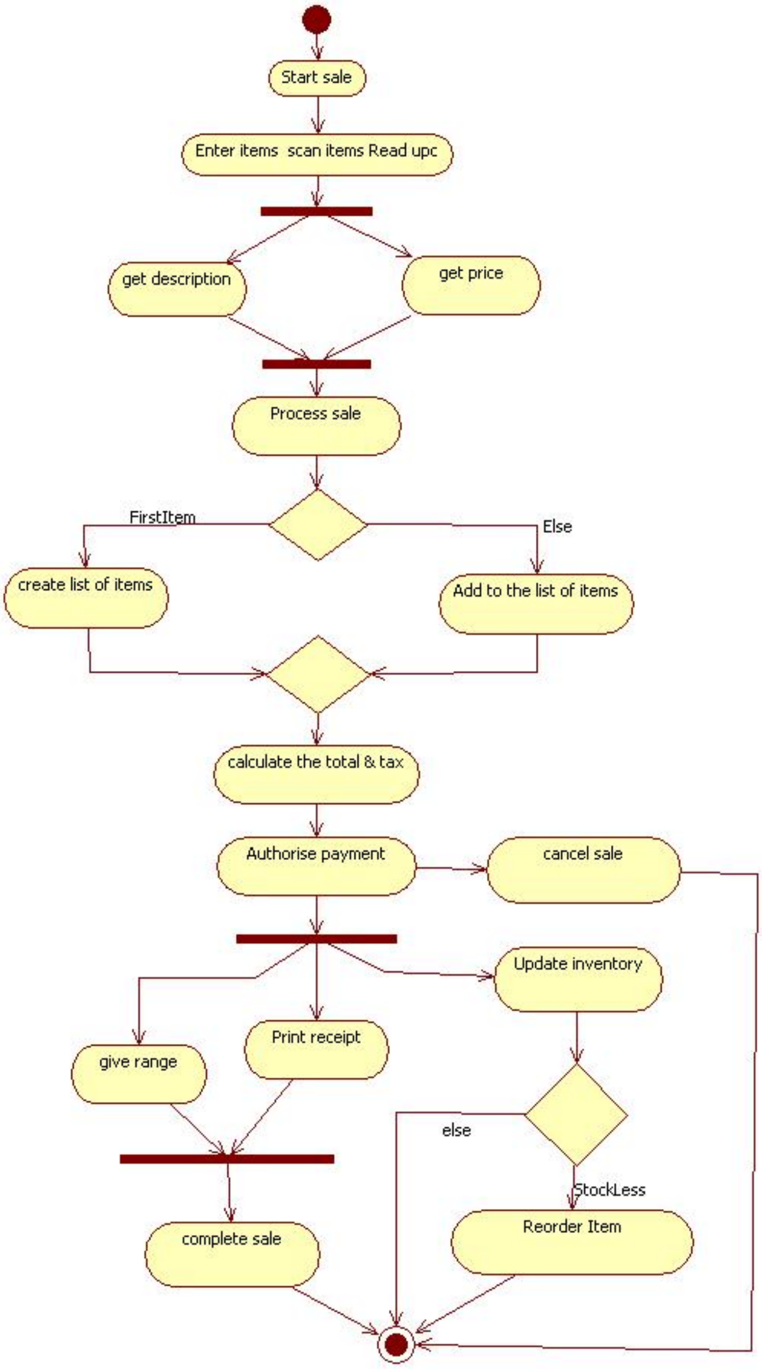


Figure 1: Activity Diagram, Sale Transaction

1.5 Security

POS system also have security critical security vulnerabilities:

1. Corrupted employee at retail can collude with swindler which perform another customer and get around scanning some items or enter lower number of products and get profit.
2. Another security vulnerability is possibility to void a closed sale receipt for refund purpose without responsible person password.
3. Possibility to issue a receipt with negative amount. I this case cashier can easily stole money from cash drawer.

There are some cases that help to prevent employee theft:

1. Set an admin window in POS system to inspect a daily list of sale receipts. Administrator must pay more attention to the frequency of cancelled receipts before sale, refunded and negative receipts. If the cashier steals money, there are going to be a lot of cancelled before completion.
2. Place closed-circuit camera pointed at the POS system to record cashier activities.
3. Set admin password for any changes in product prices and discounts and register all other changes in system for inspection purposes.
4. Limit access to generated by administrator reports about number of products, customers membership, customer preferences, most popular products, product providers and company income. This information is very important for any business and may affect future income and popularity. Access to that part must have only trusted personal.
5. Database must be also encrypted to prevent them from being copied or tampered with.

Despite all prevent measures, POS system cannot provide perfect security from entire attacks just like outer attacks. The most interesting data for hacker is customers payment cards and its reason why POS systems were attacked. Some of attacks were successful only because company ignored simple security measures as it was with Subway in 2011[9].

Also, POS system is vulnerable for entire malware attacks. In 2016 couple hounded of Wendy's restaurants were hacked by malware [10]. Payment cards were processed in POS system that allowing to malware to intercept and transmit card data.

After that cases, in 2017 security researchers found critical vulnerabilities in systems developed by SAP and Oracle. As researchers commented, system was hacked because vulnerabilities were under spotlight. After patching, Oracle confirmed that security vulnerability affects over 300,000 Oracle POS Systems [11].

To avoid payment card data, leak some countries process payments only through payment terminals. In that case payment card data less vulnerable for hackers, unlike when payment cards are processed through the POS system where security is contingent upon the actions taken by end-users and developers.

These days smartphones with NFC are widespread and its reason of introduction of payment systems as Google Pay and Apple Pay. These systems provide good security features and more comfortable for user.

2 THE BASIS OF POS MERCHANT

POS Management is a tool for remote management of payment terminals. The basis of this solution is the functionality of online communication with all terminals with the possibility of uploading individual applications of a perfect device. Other entities (salesman, store, administrator, helpdesk, service partner) can also enter the system via the web interface.

The "POS Merchant" module is a user interface designed for traders, which is built as a website, accessible from the Internet. This module allows traders to view individual trading venues, resp. individual terminals, and monitor realized card transactions within defined reports.

The POS Merchant application mainly offers the merchant:

- an overview of all installed payment terminals;
- an online overview of individual transactions made with payment cards;
- an overview of the number and volume of transactions at points of sale;
- export data on transactions in CSV format (data import into Excel, etc.);
- electronic archiving of terminal receipts;
- options of custom setting of application parameters (eg change of text on receipts);
- option of downloading statements on transactions made with payment cards.

2.1 Main Menu

Displays the merchant's navigation menu in the application, figure 2.

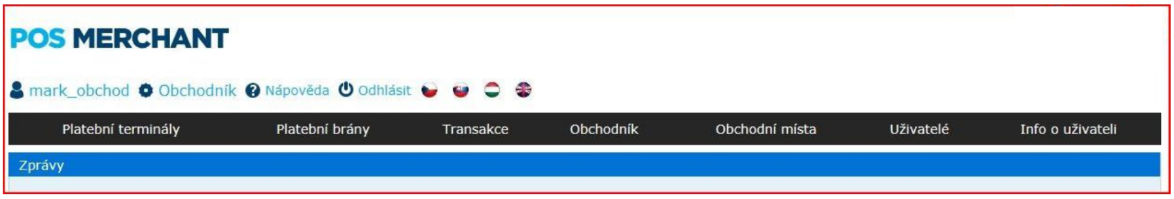


Figure 2: Menu

Payment terminals:

- Transaction sums.
- Transactions.
- Terminals.

Payment gateways:

- Payment gateways.

Transactions:

- Payment terminals.
- Payment gateways.
- Recharging.

Store:

- Store.
- Statements of merchants.
- Displays a list of all merchant terminals.

Terminals:

- A list of all payment terminals is displayed.

Users:

- A list of users is displayed set by the company administrator.
- Possibility of editing and creating new users (in the role of merchant).

User info:

- The contact e-mail of the merchant and the option to change the password will be displayed.

2.2 Users

The selected employee of the salesman (administrator) has the option to add / remove other users for the given company, who will have access to the application. The selected user has an overview only of the given establishment (establishments) / payment gateway (s). It is therefore possible to set accesses for certain persons for individual establishments / payment gateways. The administrator of the company is set by the bank (username IČO). The "Store" role is set for other users, where the administrator sets the name and password, the place (s) and sets the permissions.

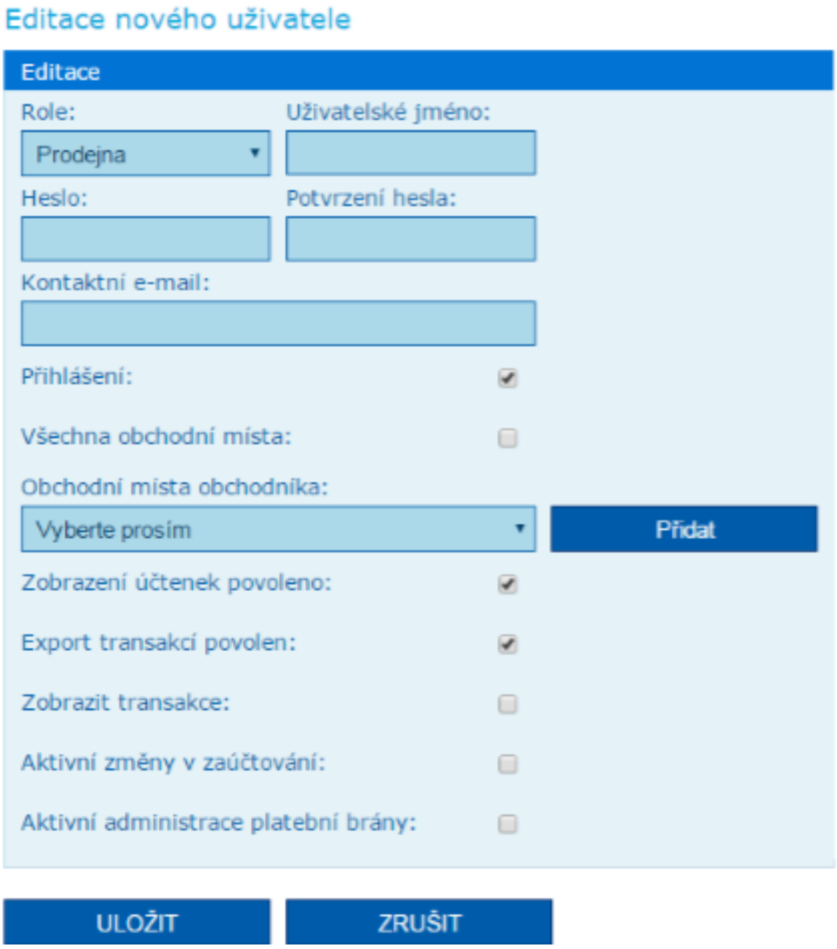
Merchant role:

- view of receipts;
- export of transactions;
- download of receipts in PDF;
- active changes of billing;
- active administration of payment gateway;
- establishment of users in a role of store.

The role of the store:

- display of transactions (default NO);
- export of transactions;
- downloading receipts in PDF;
- active billing changes (default NO);
- active payment gateway administration (default NO).

Editing a new user allows you to check which permissions will be allowed, figure 3.



Editace nového uživatele

Editace

Role: Prodejna

Uživatelské jméno:

Heslo:

Potvrzení hesla:

Kontaktní e-mail:

Přihlášení: ☑

Všechna obchodní místa: ☐

Obchodní místa obchodníka: Vyberte prosím

Zobrazení účtenek povoleno: ☑

Export transakcí povolen: ☑

Zobrazit transakce: ☐

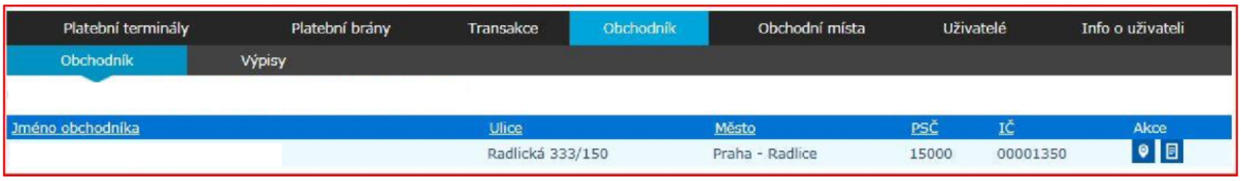
Aktivní změny v zaúčtování: ☐

Aktivní administrace platební brány: ☐

Figure 3: User editing

2.3 Trader

The Merchant option from the main menu displays detailed information of the registered company. The Business Locations item in the Action column opens a list of all the company's business locations, the Statements item displays an overview of statements, figure 4.

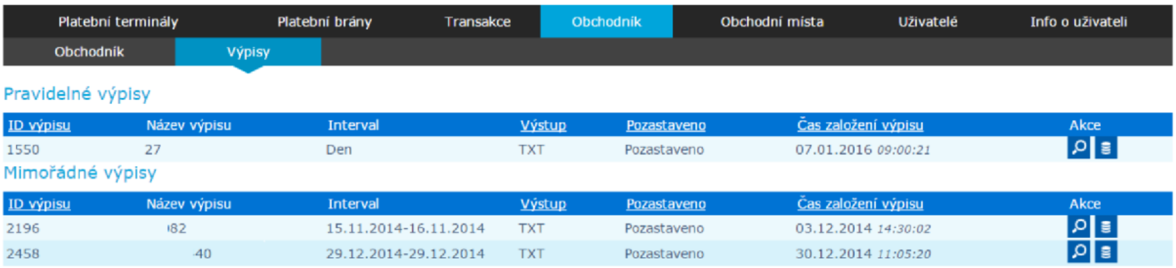


Jméno obchodníka	Ulice	Město	PSČ	IČ	Akce
	Radlická 333/150	Praha - Radlice	15000	00001350	

Figure 4: Record of a Trader

2.4 Statements

Statements option displays a list of all available statements. Statements can be found in the merchant section in the listing's submenu. The statements are divided into two categories - Regular and Extraordinary, figure 5.



Pravidelné výpisy

ID výpisu	Název výpisu	Interval	Výstup	Pozastaveno	Čas založení výpisu	Akce
1550	27	Den	TXT	Pozastaveno	07.01.2016 09:00:21	

Mimořádné výpisy

ID výpisu	Název výpisu	Interval	Výstup	Pozastaveno	Čas založení výpisu	Akce
2196	82	15.11.2014-16.11.2014	TXT	Pozastaveno	03.12.2014 14:30:02	
2458	40	29.12.2014-29.12.2014	TXT	Pozastaveno	30.12.2014 11:05:20	

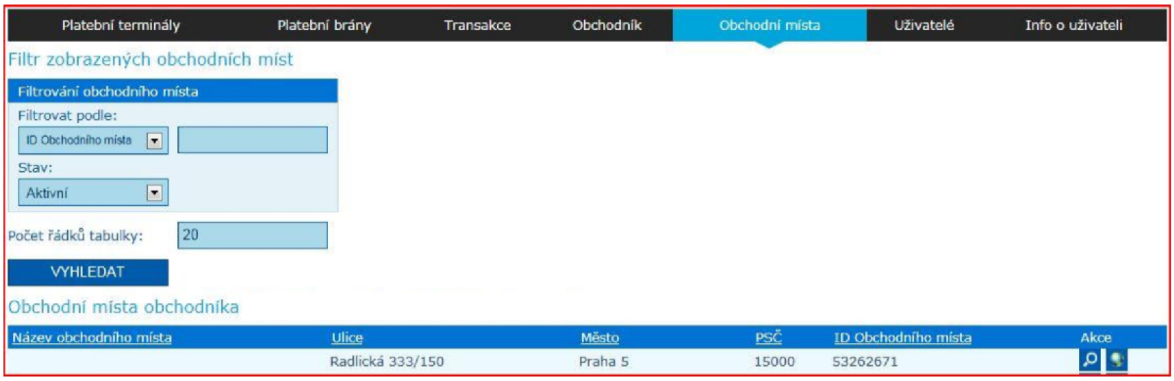
Figure 5: List of Statements

After pressing the second button (Show statements) in the Actions column, you will get to the overview of generated statements. You can filter their list by period and by file name.

Statements can be downloaded separately by clicking on the download button in the Actions column or collectively in one ZIP file. In the case of downloading to a ZIP file, it is necessary to mark all the statements you want to download on the left side and click on the Download files button.

2.5 Business Locations

The Business Locations option of the main menu displays a list of all business locations of the company. The filter provides the option of searching by store ID, store name or address. It is also possible to display active / inactive points of sale. The Terminals item in the Action column opens a list of all payment terminals of the given point of sale, the Payment gateways item displays an overview of all payment gateways, figure 6.



Filtr zobrazených obchodních míst

Filtrování obchodního místa

Filtrovat podle: ID Obchodního místa

Stav: Aktivní

Počet řádků tabulky: 20

Obchodní místa obchodníka

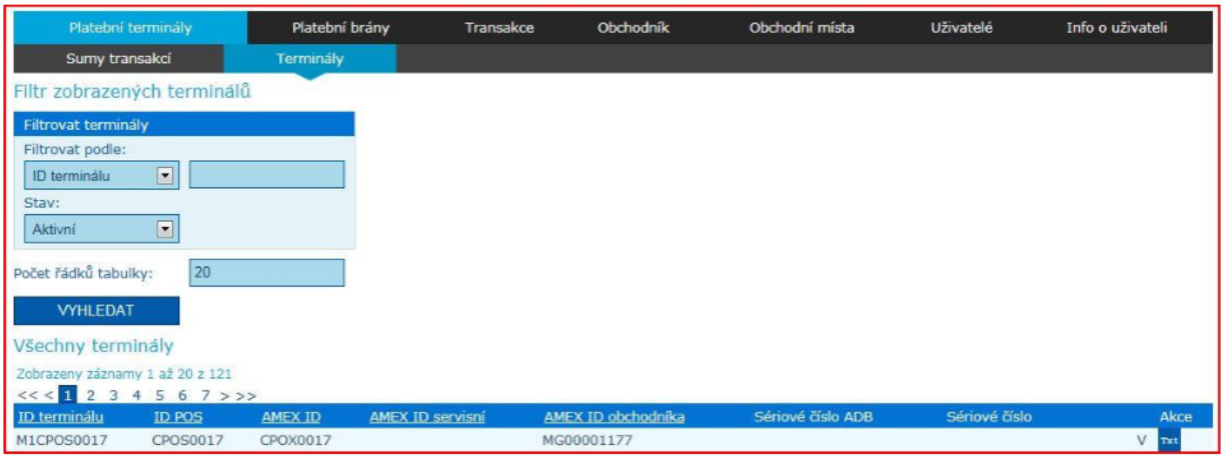
Název obchodního místa	Ulice	Město	PSČ	ID Obchodního místa	Akce
	Radlická 333/150	Praha 5	15000	53262671	

Figure 6: Listing of business locations

3 PAYMENT TERMINALS

3.1 Payment Terminals

The Terminals option of the main menu displays a list of all merchant payment terminals. The filter allows searching by POS ID, by terminal ID (for example PVTExxxx) or serial number of the terminal. It is also possible to display active / inactive terminals, figure 7.



Filtrovat podle: ID terminálu

Stav: Aktivní

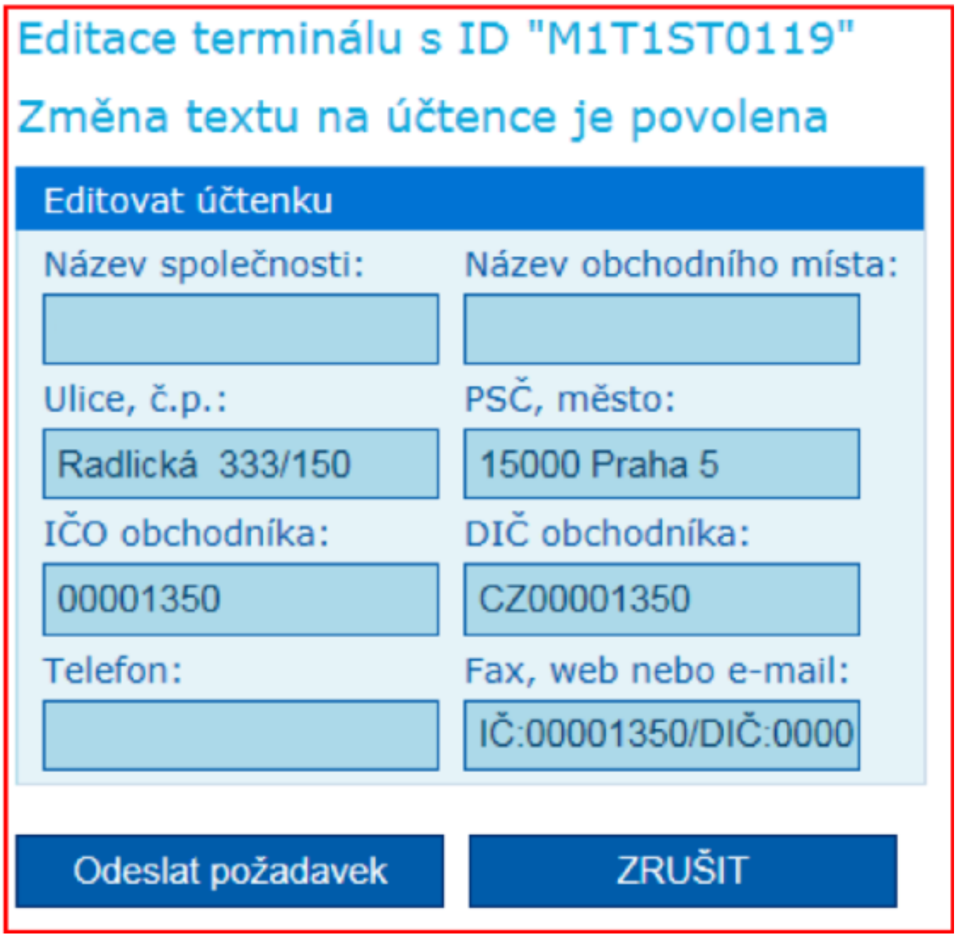
Počet řádků tabulky: 20

Zobrazeny záznamy 1 až 20 z 121

ID terminálu	ID POS	AMEX ID	AMEX ID servisní	AMEX ID obchodníka	Sériové číslo ADB	Sériové číslo	Akce
M1CPOS0017	CPOS0017	CPOX0017		MG00001177			V Txt

Figure 7: Terminal options

The Receipt text item in the Action column opens a window with a request to edit the receipt text on the given terminal. If the merchant is allowed to change the text on the receipt, he sends a change request to the bank, where this request is processed (approved / rejected), figure 8.



Editace terminálu s ID "M1T1ST0119"

Změna textu na účtence je povolena

Editovat účtenku

Název společnosti:

Název obchodního místa:

Ulice, č.p.: Radlická 333/150

PSČ, město: 15000 Praha 5

IČO obchodníka: 00001350

DIČ obchodníka: CZ00001350

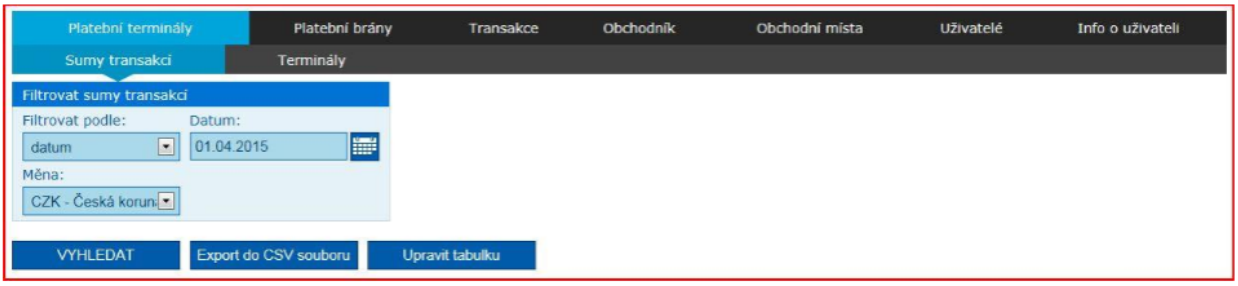
Telefon:

Fax, web nebo e-mail: IČ:00001350/DIČ:0000

Figure 8: Terminal Receipt Editing

3.2 Amounts of Transactions

An overview of the summary of card transactions (numbers and volumes) for the merchant, divided into individual points of sale, or terminals for a certain period. The summary of transactions is given according to the type of transaction (Sale / Reversal / Cashback/ Completion of pre-authorization / Pre-authorization / Authorized / Posted / Rejected / Closing), figure 9.



Filtrovat sumy transakcí

Filtrovat podle: datum

Datum: 01.04.2015

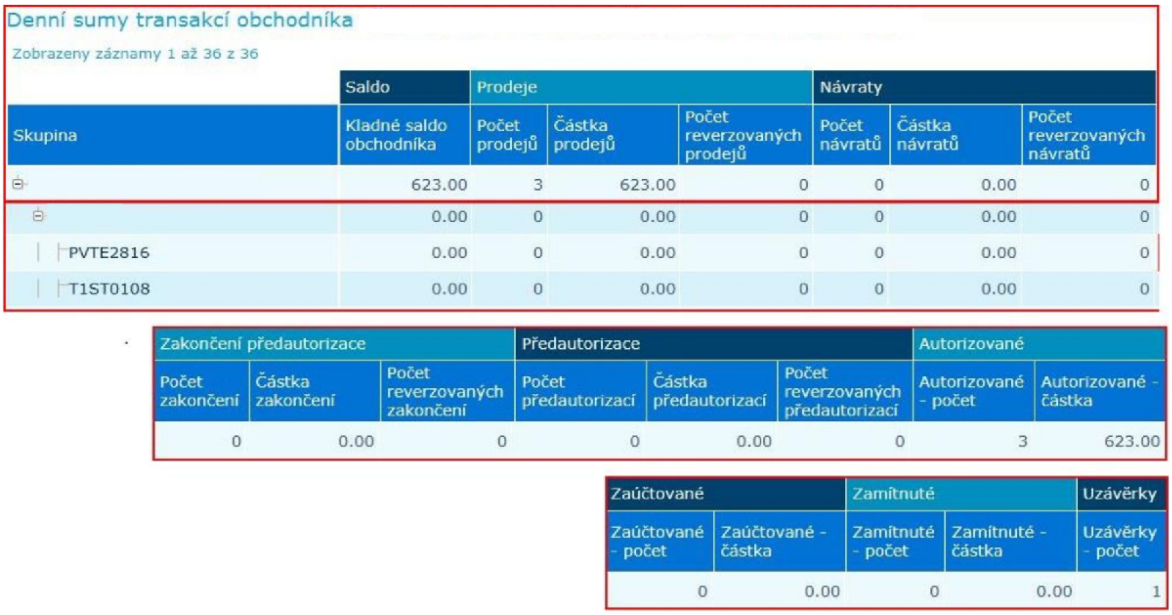
Měna: CZK - Česká korun

Figure 9: Filtering Transactions

The amounts of transactions can be filtered according to the time period:

- date / interval / month / quarter / year also by currency;
- CZK / EUR / GBP / USD.

The obtained values can then be converted to a CSV file (Export to CSV file), figure 10.



Denní sumy transakcí obchodníka

Zobrazeny záznamy 1 až 36 z 36

	Saldo	Prodeje			Návraty		
Skupina	Kladné saldo obchodníka	Počet prodejů	Částka prodejů	Počet reverzovaných prodejů	Počet návratů	Částka návratů	Počet reverzovaných návratů
⊟	623.00	3	623.00	0	0	0.00	0
⊟	0.00	0	0.00	0	0	0.00	0
PVTE2816	0.00	0	0.00	0	0	0.00	0
T1ST0108	0.00	0	0.00	0	0	0.00	0

Zakončení předautorizace			Předautorizace			Autorizované	
Počet zakončení	Částka zakončení	Počet reverzovaných zakončení	Počet předautorizací	Částka předautorizací	Počet reverzovaných předautorizací	Autorizované - počet	Autorizované - částka
0	0.00	0	0	0.00	0	3	623.00

Zaúčtované		Zamítnuté		Uzávěrky
Zaúčtované - počet	Zaúčtované - částka	Zamítnuté - počet	Zamítnuté - částka	Uzávěrky - počet
0	0.00	0	0.00	1

Figure 10: Example of Transaction Overview

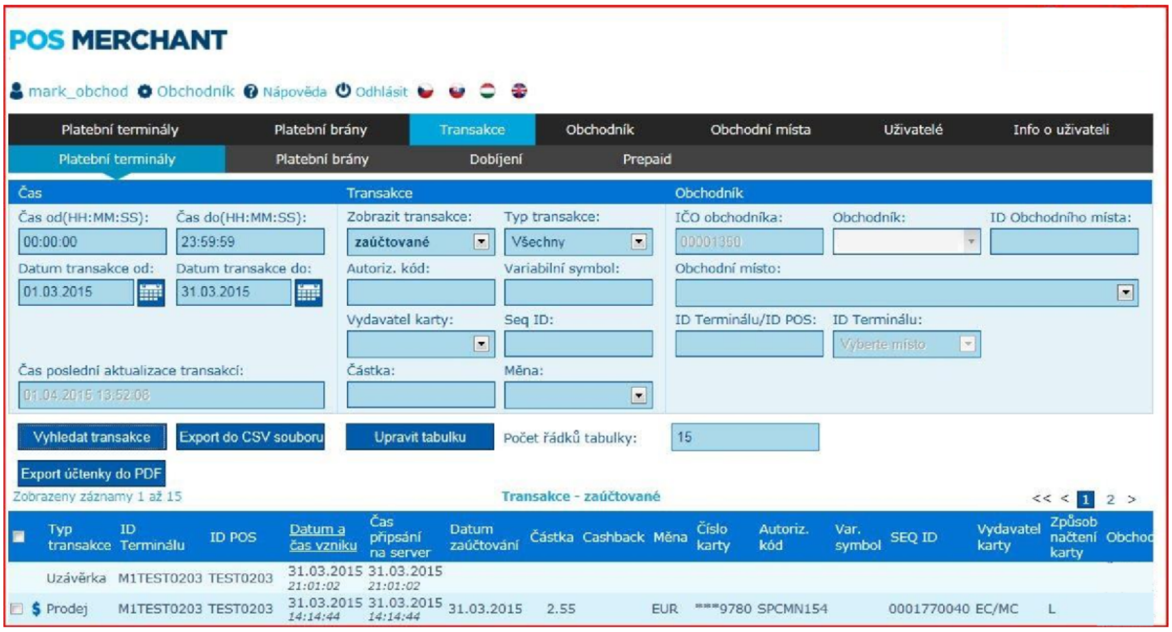
In the table, it is possible to set the display of individual parts using the check boxes, which are visible below (Balance / Sales / Cashback/ Returns / Completion of pre-authorization / Pre-authorization / Authorized / Posted / Rejected / Closing).

3.3 Transactions - Payment terminals

The overview of transactions is accessible online with a delay of about 10 minutes. The displayed transactions can be of the following type, depending on the settings of the terminal and the services provided:

- sales;
- sales – cashback;
- return of goods;
- subtotal Cash withdrawal;
- pre-authorization;
- pre-authorization – closing;
- description.

By exporting transactions to CSV format (MS Excel), you can independently create reports according to individual terminals, points of sale, etc. The Transactions option in the main menu allows you to search for transactions according to the criteria below, figure 11.



Čas

Čas od(HH:MM:SS): 00:00:00

Čas do(HH:MM:SS): 23:59:59

Datum transakce od: 01.03.2015

Datum transakce do: 31.03.2015

Čas poslední aktualizace transakcí: 01.04.2015 13:52:08

Transakce

Zobrazit transakce: zaúčtované

Typ transakce: Všechny

Autoriz. kód:

Variabilní symbol:

Vydavatel karty:

Seq ID:

Částka:

Měna:

Obchodník

IČO obchodníka: 00001350

Obchodník:

ID Obchodního místa:

Obchodní místo:

ID Terminálu/ID POS:

ID Terminálu: Vyberte místo

Počet řádků tabulky: 15

Zobrazeny záznamy 1 až 15

Transakce - zaúčtované

☐	Typ transakce	ID Terminálu	ID POS	Datum a čas vzniku	Čas připsání na server	Datum zaúčtování	Částka	Cashback	Měna	Číslo karty	Autoriz. kód	Var. symbol	SEQ ID	Vydavatel karty	Způsob načtení karty	Obchod
	Uzávěrka	M1TEST0203	TEST0203	31.03.2015 21:01:02	31.03.2015 21:01:02											
☐	$ Prodej	M1TEST0203	TEST0203	31.03.2015 14:14:44	31.03.2015 14:14:44	31.03.2015	2.55		EUR	***9780	SPCMN154		0001770040	EC/MC	L	

Figure 11: Overview of Payment Terminals

3.4 Requirement to Change the Status / Type of a Transaction (Reversal, Reprocessing)

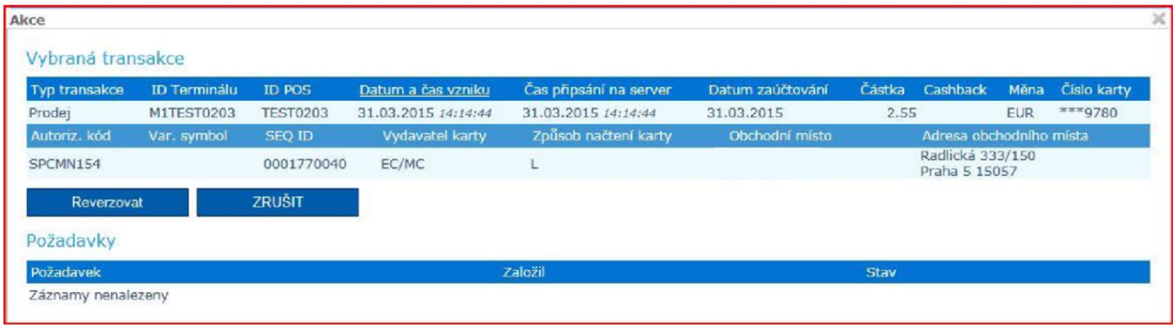
The POS Merchant application offers the following functions, within which the type / status of the transaction can be changed:

1. Reverse sales transactions, ie sending a request for the return of the relevant amount from the merchant's account to the account of the payment card holder

a) In the first phase, it is necessary to search for the relevant payment of posted transactions (upper menu Transactions - payment terminals, above) and click on the dollar symbol (Reverse) located in the Transaction type column.

b) After performing the step described above, a window called Action will open, where there is a detail of the selected payment and the following button, figure 12.

c)



Akce

Vybraná transakce

Typ transakce	ID Terminálu	ID POS	Datum a čas vzniku	Čas připsání na server	Datum zaúčtování	Částka	Cashback	Měna	Číslo karty
Prodej	M1TEST0203	TEST0203	31.03.2015 *14:14:44*	31.03.2015 *14:14:44*	31.03.2015	2.55		EUR	***9780

Autoriz. kód	Var. symbol	SEQ ID	Vydavatel karty	Způsob načtení karty	Obchodní místo	Adresa obchodního místa
SPCMN154		0001770040	EC/MC	L		Radlická 333/150 Praha 5 15057

Požadavky

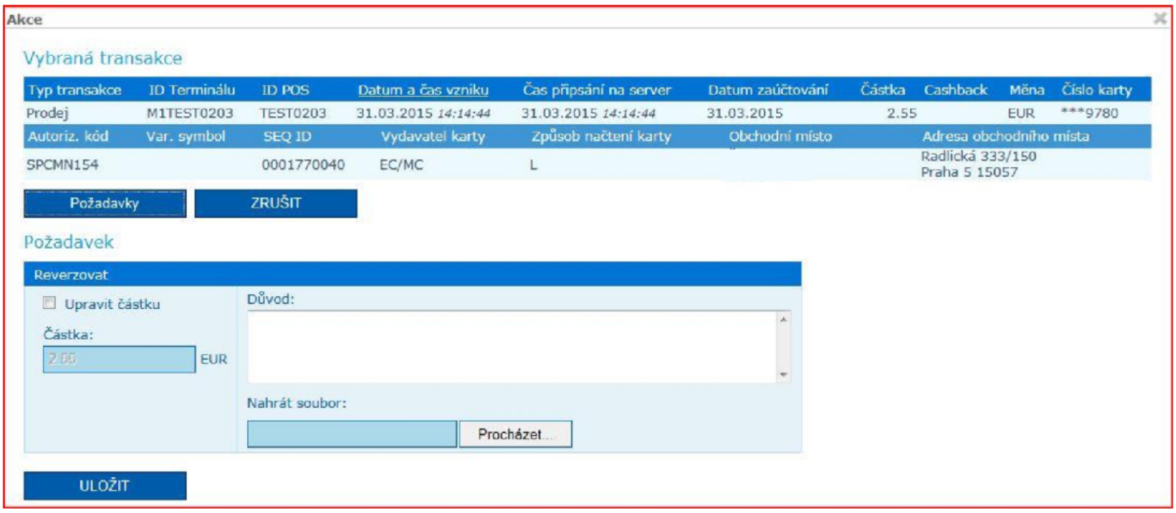
Požadavek	Založil	Stav

Záznamy nenalezeny

Figure 12: Action Window

2. Reverse - clicking on the button will expand the Actions window by a section called Request, where it is necessary to fill in the reason for reversing and click on the Save button. After saving, the line of the relevant transaction in the list of posted transactions is highlighted in red, which signals an active (not yet processed) request. In addition, the transaction to be reversed appears among unposted transactions as a Sales Reversal (Transaction Type column). As soon as the request is settled on the part of the bank, a new transaction is created in the posted transactions by the request processing, figure 13.

d)



Akce

Vybraná transakce

Typ transakce	ID Terminálu	ID POS	Datum a čas vzniku	Čas připsání na server	Datum zaúčtování	Částka	Cashback	Měna	Číslo karty
Prodej	M1TEST0203	TEST0203	31.03.2015 *14:14:44*	31.03.2015 *14:14:44*	31.03.2015	2.55		EUR	***9780

Autoriz. kód	Var. symbol	SEQ ID	Vydavatel karty	Způsob načtení karty	Obchodní místo	Adresa obchodního místa
SPCMN154		0001770040	EC/MC	L		Radlická 333/150 Praha 5 15057

Požadavek

Reverzovat

☐ Upravit částku

Částka: 2.55 EUR

Důvod:

Nahrát soubor:

Figure 13: Action Window Expended

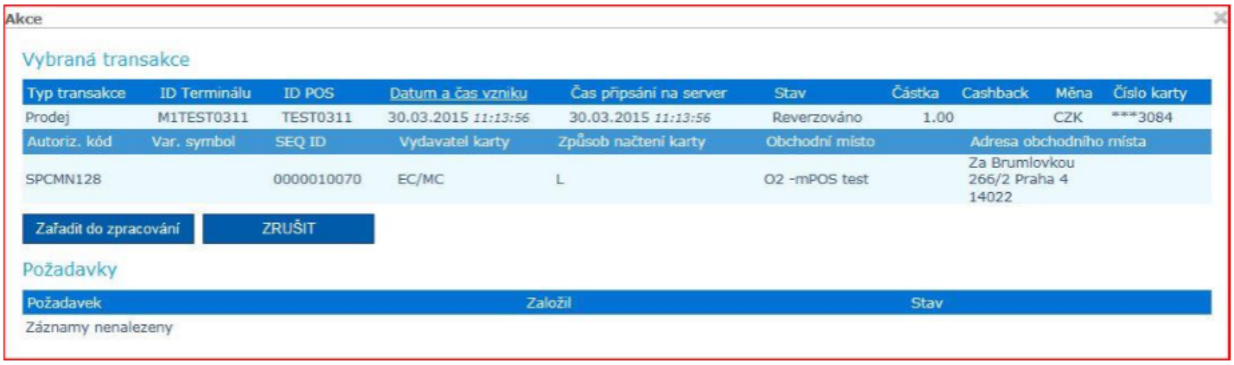
3. Partial Reversal — If only part of the transaction needs to be reversed, in the Request window, select the Adjust Amount option. This will make the field available for adjusting the amount - what amount (difference) of the transaction do you want to return to the payment card holder of the original transaction (sales transaction was made for 200.00 CZK, payment card holder should pay 150.00 CZK - partial reversal will be 50, 00 CZK) After entering the value of reversion and reason, it is necessary to save the request. After settling the partial reversal, in

the overview of transactions, it is possible to move the mouse cursor to a specific amount in the amount column and information about the amount of the reversed amount will be displayed.

4. Cancel - clicking on the button will return to the list of settled transactions.
5. Include the transaction in processing, include the reversed transaction in the list of transactions designated for settlement.

e) For unposted transactions for which it is Reversed (Status column), a reprocessing request can be sent. In the first phase, you need to find the appropriate payment in the unposted transactions and click on the dollar symbol located in the Transaction Type column.

After performing the step described above, a window called Actions will open, where there is a detail of the selected payment and the following buttons (figure 14):



Akce

Vybraná transakce

Typ transakce	ID Terminálu	ID POS	Datum a čas vzniku	Čas připsání na server	Stav	Částka	Cashback	Měna	Číslo karty
Prodej	M1TEST0311	TEST0311	30.03.2015 *11:13:56*	30.03.2015 *11:13:56*	Reverzováno	1.00		CZK	***3084

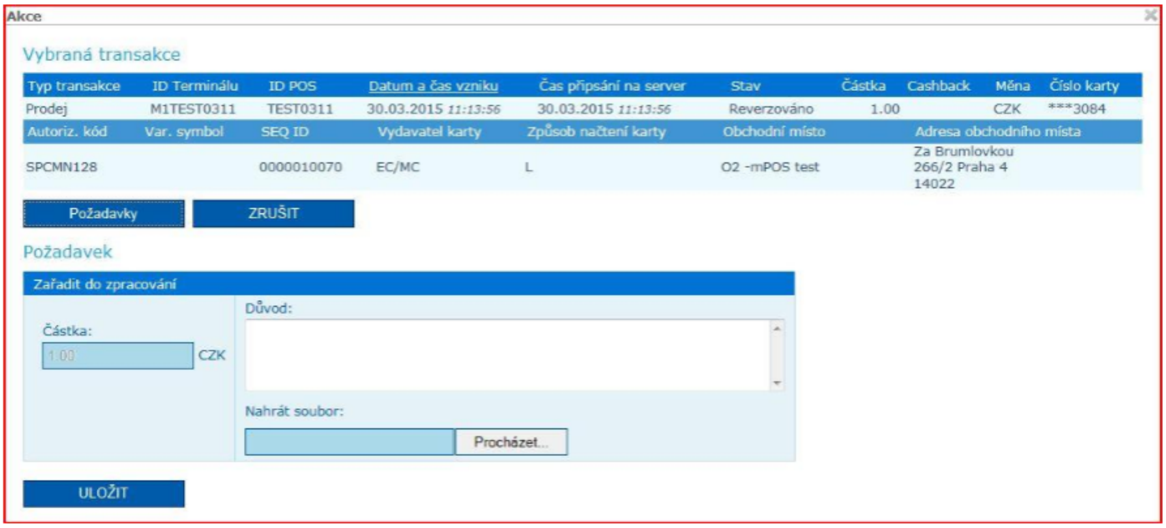
Autoriz. kód	Var. symbol	SEQ ID	Vydavatel karty	Způsob načtení karty	Obchodní místo	Adresa obchodního místa
SPCMN128		0000010070	EC/MC	L	O2 -mPOS test	Za Brumlovkou 266/2 Praha 4 14022

Požadavky

Požadavek	Založil	Stav
Záznamy nenalezeny		

Figure 14: Action Window

6. Include in processing - clicking on the button will expand the Actions window by the Request section, where it is necessary to fill in the reason for reprocessing and click on the Save button. As part of this action, the line of the relevant transaction in the overview of posted transactions will be highlighted in red, which signals an active (not yet processed) request. In addition, the transaction to be re-posted appears among the posted transactions as a Resale (Transaction Type column). As soon as the request is processed by the bank, a new transaction will be created in the posted transactions by the request processing, figure 15.



Akce

Vybraná transakce

Typ transakce	ID Terminálu	ID POS	Datum a čas vzniku	Čas připsání na server	Stav	Částka	Cashback	Měna	Číslo karty
Prodej	M1TEST0311	TEST0311	30.03.2015 11:13:56	30.03.2015 11:13:56	Reverzováno	1.00		CZK	***3084

Autoriz. kód	Var. symbol	SEQ ID	Vydavatel karty	Způsob načtení karty	Obchodní místo	Adresa obchodního místa
SPCMN128		0000010070	EC/MC	L	O2 -mPOS test	Za Brumlovkou 266/2 Praha 4 14022

Požadavek

Zařadit do zpracování

Částka: 1.00 CZK

Důvod:

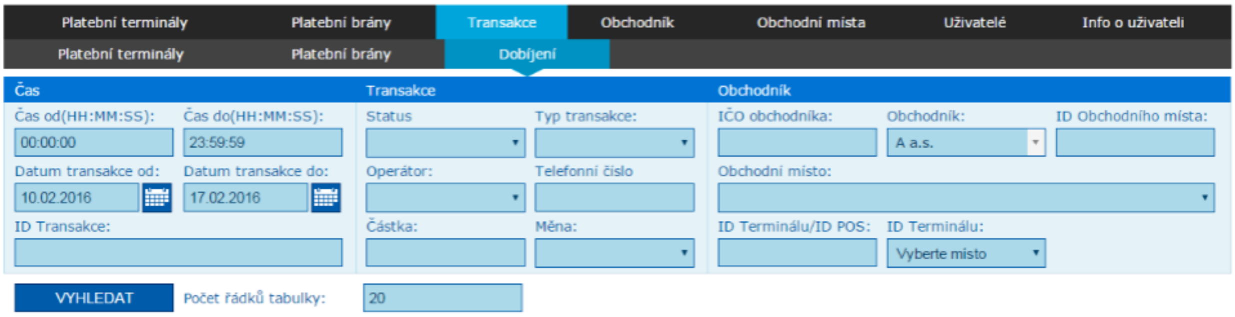
Nahrát soubor:

Figure 15: Action Window Expanded

Cancel - Click the button to return to the list of unposted transactions. For the above requirements, it is possible to attach a file to the application (a copy of the receipt, etc.)

3.5 Transactions – Recharging

The Recharge option is intended for merchants who use the mobile operator credit recharge service on the payment terminal, figure 16. The merchant can monitor completed transactions online for recharging.



Čas

Čas od(HH:MM:SS): 00:00:00

Čas do(HH:MM:SS): 23:59:59

Datum transakce od: 10.02.2016

Datum transakce do: 17.02.2016

ID Transakce:

Transakce

Status

Typ transakce:

Operátor:

Telefonní číslo

Částka:

Měna:

Obchodník

IČO obchodníka:

Obchodník: A a.s.

ID Obchodního místa:

Obchodní místo:

ID Terminálu/ID POS:

ID Terminálu: Vyberte místo

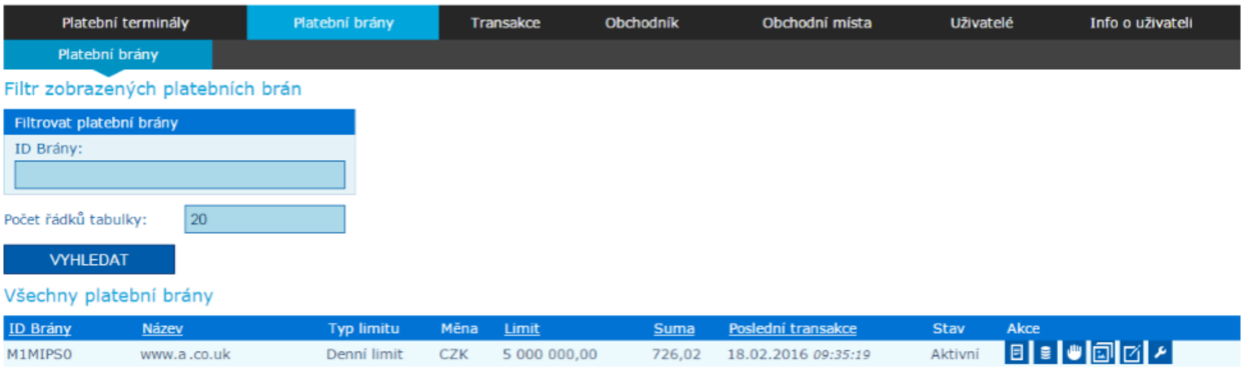
Počet řádků tabulky: 20

Figure 16: Recharge Option

4 PAYMENT GATEWAYS

4.1 Payment gateways

The Payment gateways option in the main menu displays a list of all merchant payment gateways. Filter Allows searching by POS ID, by payment gateway ID (for example MIPSxxxx), figure 17.



Filtr zobrazených platebních brán

Filtrovat platební brány

ID Brány:

Počet řádků tabulky: 20

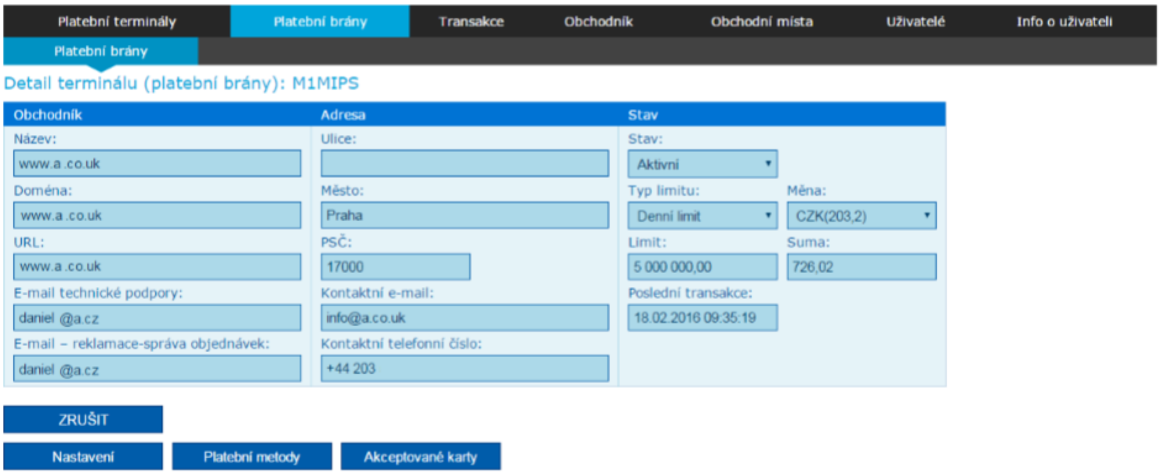
Všechny platební brány

ID Brány	Název	Typ limitu	Měna	Limit	Suma	Poslední transakce	Stav	Akce
M1MIPS0	www.a .co.uk	Denní limit	CZK	5 000 000,00	726,02	18.02.2016 *09:35:19*	Aktivní	

Figure 17: Payment Gateways

There are several options in the Action column:

1. Detail of a payment gateway. The Gateway settings can be seen in the Payment Gateway Detail. This is only an overview, and nothing can be changed and edited here, figure 18



Detail terminálu (platební brány): M1MIPS

Obchodník

Název: www.a .co.uk

Doména: www.a .co.uk

URL: www.a .co.uk

E-mail technické podpory: daniel @a.cz

E-mail – reklamace-správa objednávek: daniel @a.cz

Adresa

Ulice:

Město: Praha

PSČ: 17000

Kontaktní e-mail: info@a.co.uk

Kontaktní telefonní číslo: +44 203

Stav

Stav: Aktivní

Typ limitu: Denní limit

Měna: CZK(203,2)

Limit: 5 000 000,00

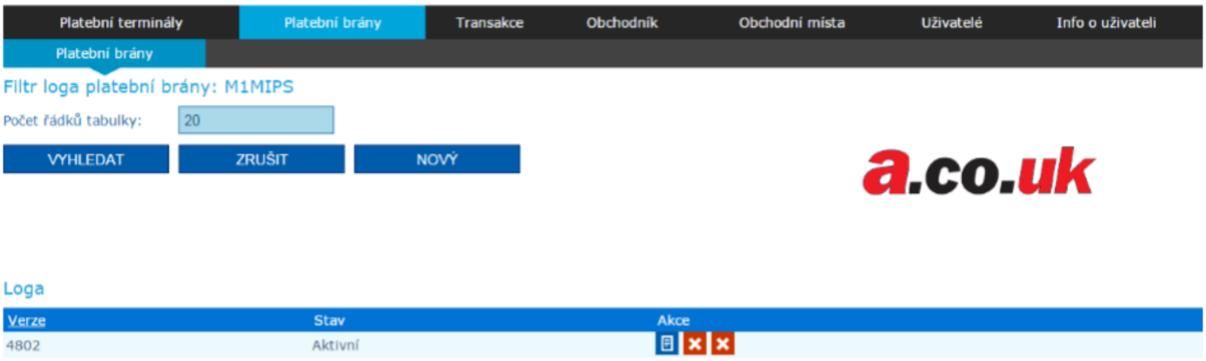
Suma: 726,02

Poslední transakce: 18.02.2016 09:35:19

2.

Figure 18: Overview of Payment Gateways Details

3. Transaction. This button takes you to an overview of the transactions of a specific payment gateway. Description of transactions below (3.3 Transactions - Payment Gateways).
4. Bin Blacklist. List of prohibited payment card number prefixes. It is not possible to set and add forbidden prefixes.
5. Logos. After clicking on the Logos button, the menu for inserting a new logo, setting the active logo and possibly deleting unused logos will be displayed. After inserting the logo via the NEW button and saving it, the logo is sent for approval. Until then, the status of the logo (in the status column) is pending approval and cannot be used. After its approval, the status will change to Approved and the logo can be set as displayed on the payment gateway, figure 19.



Filtr loga platební brány: M1MIPS

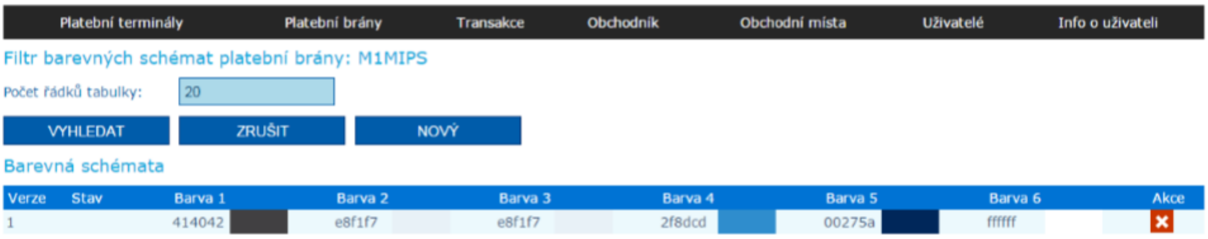
Počet řádků tabulky: 20

Loga

Verze	Stav	Akce
4802	Aktivní	▤ ✗ ✗

Figure 19: Inserted Logo

6. Color scheme. After clicking on the Color scheme button, an overview of color schemes and their current status will be displayed. After pressing the NEW button, it is possible to create a new color scheme and after saving it will be sent again for approval, just like the logo, figure 20.



Filtr barevných schémat platební brány: M1MIPS

Počet řádků tabulky: 20

Barevná schémata

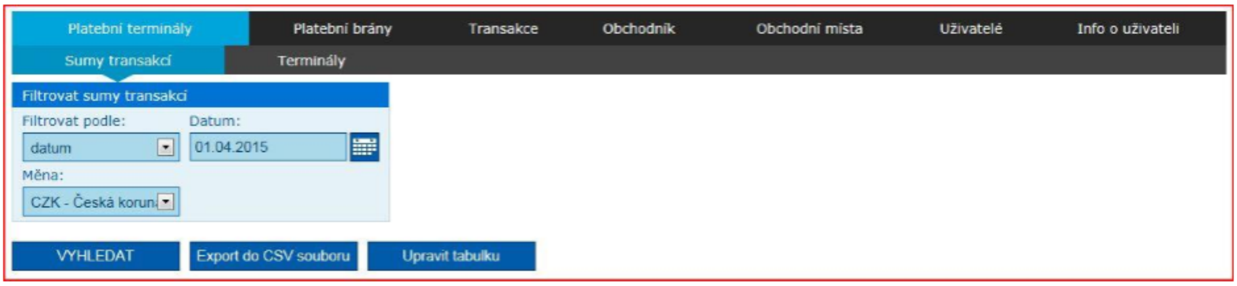
Verze	Stav	Barva 1	Barva 2	Barva 3	Barva 4	Barva 5	Barva 6	Akce
1		414042	e8f1f7	e8f1f7	2f8dcd	00275a	ffffff	✗

Figure 20: Color scheme

7. Keys. Requirement to enter payment gateway keys.

4.2 Amounts of Transactions

The overview of the summary of card transactions (numbers and volumes) for the payment gateway can be entered via the Payment terminals and then Transaction amounts tab. virtual terminals for a certain period. The summary of transactions is presented according to the type of transaction (Sale / Reverse / Authorized / Posted / Rejected), figure 21.



Filtrovat sumy transakcí

Filtrovat podle: datum

Datum: 01.04.2015

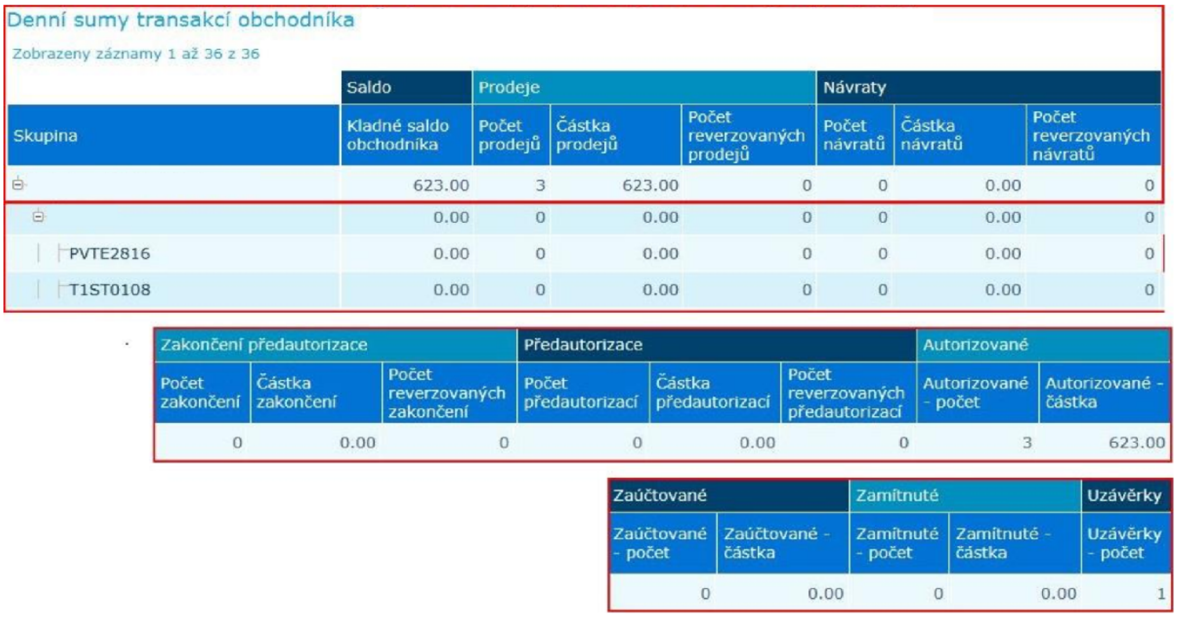
Měna: CZK - Česká korun

Figure 21: Filtering Summary of Card Transactions

Transaction amounts can be filtered by time period:

- date / interval / month / quarter / year and at the same time by currency;
- CZK / EUR / GBP / USD.

The obtained values can then be converted to a CSV file (Export to CSV file), figure 22.



Denní sumy transakcí obchodníka

Zobrazeny záznamy 1 až 36 z 36

	Saldo	Prodeje			Návraty		
Skupina	Kladné saldo obchodníka	Počet prodejů	Částka prodejů	Počet reverzovaných prodejů	Počet návratů	Částka návratů	Počet reverzovaných návratů
⊟	623.00	3	623.00	0	0	0.00	0
⊟	0.00	0	0.00	0	0	0.00	0
PVTE2816	0.00	0	0.00	0	0	0.00	0
T1ST0108	0.00	0	0.00	0	0	0.00	0

Zakončení předautorizace			Předautorizace			Autorizované	
Počet zakončení	Částka zakončení	Počet reverzovaných zakončení	Počet předautorizací	Částka předautorizací	Počet reverzovaných předautorizací	Autorizované - počet	Autorizované - částka
0	0.00	0	0	0.00	0	3	623.00

Zaúčtované		Zamítnuté		Uzávěrky
Zaúčtované - počet	Zaúčtované - částka	Zamítnuté - počet	Zamítnuté - částka	Uzávěrky - počet
0	0.00	0	0.00	1

Figure 22: Transaction Amounts

In the table, it is possible to set the display of individual parts using the check boxes, which are visible below (Balance / Sales / Returns / Authorized / Posted / Rejected).

4.3 Transactions – Payment Gateways

An overview of transactions is available on the Transactions - Payment Gateways tab. The overview is accessible online with a delay of approx. 10 minutes.
Explanation of transaction statuses:

- Authorization confirmed / Sent for posting it is not possible to change the transaction amount, only the transaction is excluded from settlement (Blue Dollar).
- Authorization confirmed / Not sent for posting: Waiting for final order confirmation, transaction amount can be changed, or transaction is excluded from settlement dollar).
- Posted: the transaction is settled.
- Direct transaction reversal request: transaction reversal request recorded.
- Reversed - Revoked: transaction was canceled (failed connection to server).
- Reversed –Returned: reversed transaction processed.
- Rejected.
- Cancelled order by user.
- Order processing.
- Order created.

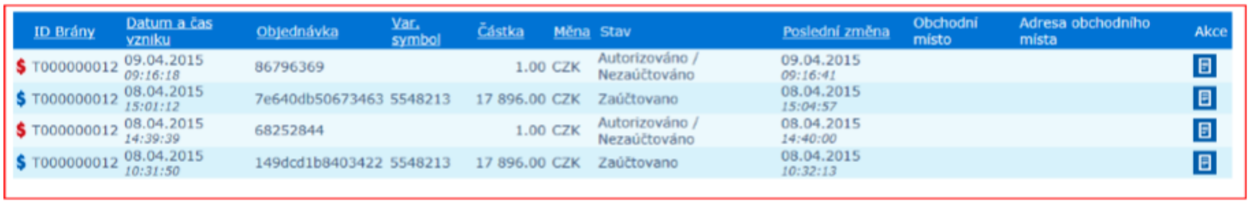
4.4 Requirement To change the status / type of a transaction (reversal, reprocessing)

The POS Merchant application offers the following functions, within which the type / status of the transaction can be changed.

If the transaction is in the Authorized / Unposted state (dollar symbol is red), it is possible to reverse it before posting, or to include it in the posting, if it is already posted (dollar symbol is blue), it is only possible to send a request for reversal.

Reverse / include for settlement, ie sending a request for the return of the relevant amount of the merchant's account and include the account of the payment card holder or include the transaction in the list of transactions designated for settlement.

In the First Phase, you need to find the appropriate payment in the Transactions tab and click on the dollar symbol located in the Gateway ID column, figure 23.

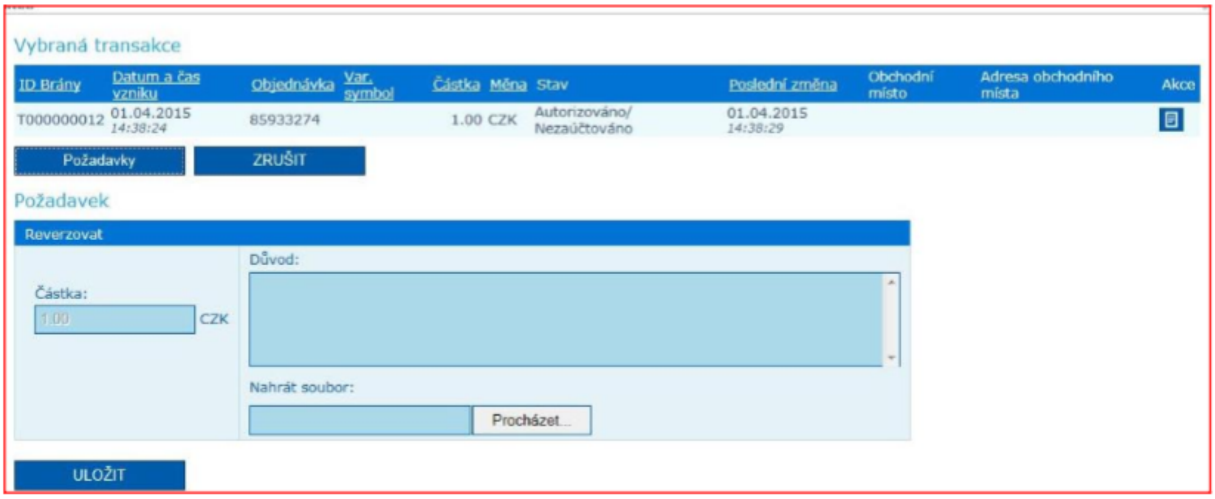


	ID Brány	Datum a čas vzniku	Objednávka	Var. symbol	Částka	Měna	Stav	Poslední změna	Obchodní místo	Adresa obchodního místa	Akce
$	T000000012	09.04.2015 09:16:18	86796369		1.00	CZK	Autorizováno / Nezaúčtováno	09.04.2015 09:16:41			▤
$	T000000012	08.04.2015 15:01:12	7e640db50673463	5548213	17 896.00	CZK	Zaúčtovano	08.04.2015 15:04:57			▤
$	T000000012	08.04.2015 14:39:39	68252844		1.00	CZK	Autorizováno / Nezaúčtováno	08.04.2015 14:40:00			▤
$	T000000012	08.04.2015 10:31:50	149dcd1b8403422	5548213	17 896.00	CZK	Zaúčtovano	08.04.2015 10:32:13			▤

Figure 23: Transaction Tab

After performing the step described above, a window called Action will open, where there is a detail of the selected payment and the following button (figure 24):

- Reverse - clicking on the blue dollar button will expand the Action window by a section called Request, where you must fill in the reason for the reversal and click on the Save button. After saving, the line of the respective transaction in the list of posted transactions is highlighted in red, which signals an active (not yet processed) request. In addition, the transaction to be reversed appears among the unposted transactions as Sales Reversal (Transaction Type column). As soon as the request is settled on the part of the bank, a new transaction is created in the posted transactions by the request processing date.



Vybraná transakce

ID Brány	Datum a čas vzniku	Objednávka	Var. symbol	Částka	Měna	Stav	Poslední změna	Obchodní místo	Adresa obchodního místa	Akce
T000000012	01.04.2015 14:38:24	85933274		1.00	CZK	Autorizováno/ Nezaúčtováno	01.04.2015 14:38:29			▤

Požadavek

Reverzovat

Částka: 1.00 CZK

Důvod:

Nahrát soubor:

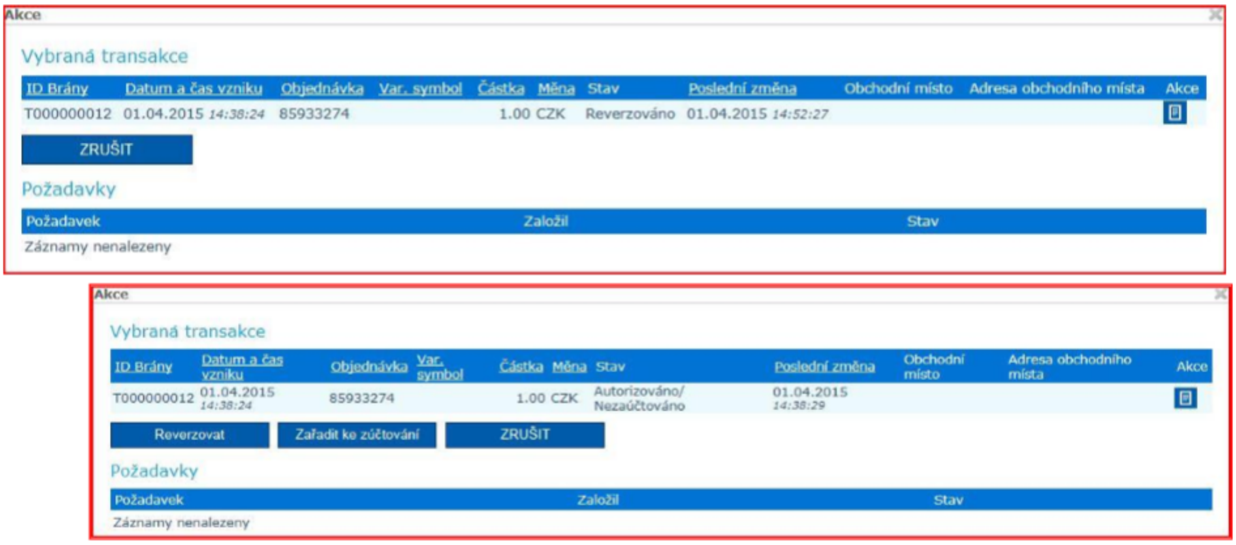
Figure 24: Action Window

- Partial Reverse - Clicking on the blue dollar sign expands the Actions window, where you click Reverse. The requirement to adjust the amount and to state the reason for the partial reversal will then be extended. Checking the Edit amount option will unlock the box where you specify the amount (difference) of the transaction you want to return the original transaction to the payment card holder (eg the sales transaction was made for CZK 200.00, the payment card holder should pay CZK 150.00 - partial reversal will be 50.00 CZK) After entering the value of the reversal and the reason, it is necessary to save the request. After settling the partial reversal, in the overview of transactions, it is possible to move the mouse cursor to a specific amount in the amount column and information about the reversed amount is displayed.

- Include the transaction in processing, include the transaction in the list of transactions specified for settlement. Clicking on the red dollar button opens the Actions window, where there is a detailed overview of the transaction. After pressing the Include for settlement button, the Actions window will be expanded by the Request section, where it is necessary to fill in the reason for processing and click on the Save button. Alternatively, the amount can be adjusted. As part of this action, the line of the relevant transaction in the list of unposted transactions is highlighted in red, which signals an active (as yet unprocessed) request. In addition, the transaction to be re-posted appears among the posted transactions as a Resale (Transaction Type column).
- Cancel - Click the button to return to the transaction overview.

For the above requirements, it is possible to attach a file to the application (a copy of the receipt, etc.).

If the transaction is in the Reverse state, it is possible to view the history of requests by clicking on the dollar symbol located in the Gateway ID column, figure 25.



Akce

Vybraná transakce

ID Brány	Datum a čas vzniku	Objednávka	Var. symbol	Částka	Měna	Stav	Poslední změna	Obchodní místo	Adresa obchodního místa	Akce
T000000012	01.04.2015 14:38:24	85933274		1.00	CZK	Reverzováno	01.04.2015 14:52:27			

Požadavky

Požadavek	Založil	Stav
Záznamy nenalezeny		

Akce

Vybraná transakce

ID Brány	Datum a čas vzniku	Objednávka	Var. symbol	Částka	Měna	Stav	Poslední změna	Obchodní místo	Adresa obchodního místa	Akce
T000000012	01.04.2015 14:38:24	85933274		1.00	CZK	Autorizováno/ Nezaúčtováno	01.04.2015 14:38:29			

Požadavky

Požadavek	Založil	Stav
Záznamy nenalezeny		

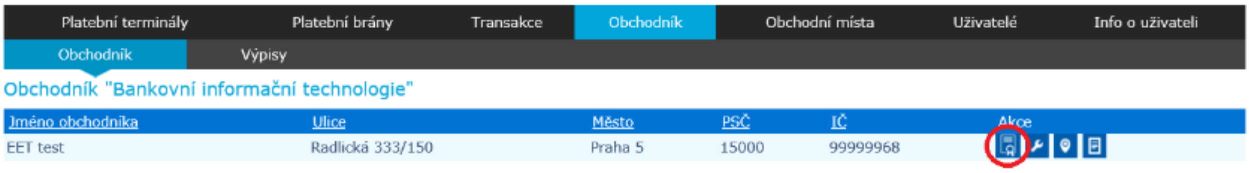
Figure 25: Overview of Chosen Transactions

5 ELECTRONIC RECORD OF SALES

Electronic records of sales (hereinafter referred to as ERS) paid not only by payment card, but also in cash (including meal vouchers) through a payment terminal, without the need to purchase a cash register.

5.1 Insertion of ERS certificate

To insert a certificate in the Merchant tab, click on the icon showed below on the picture 26:



Obchodník "Bankovní informační technologie"

Jméno obchodníka	Ulice	Město	PSČ	IČ	Akce
EET test	Radlická 333/150	Praha 5	15000	99999968	

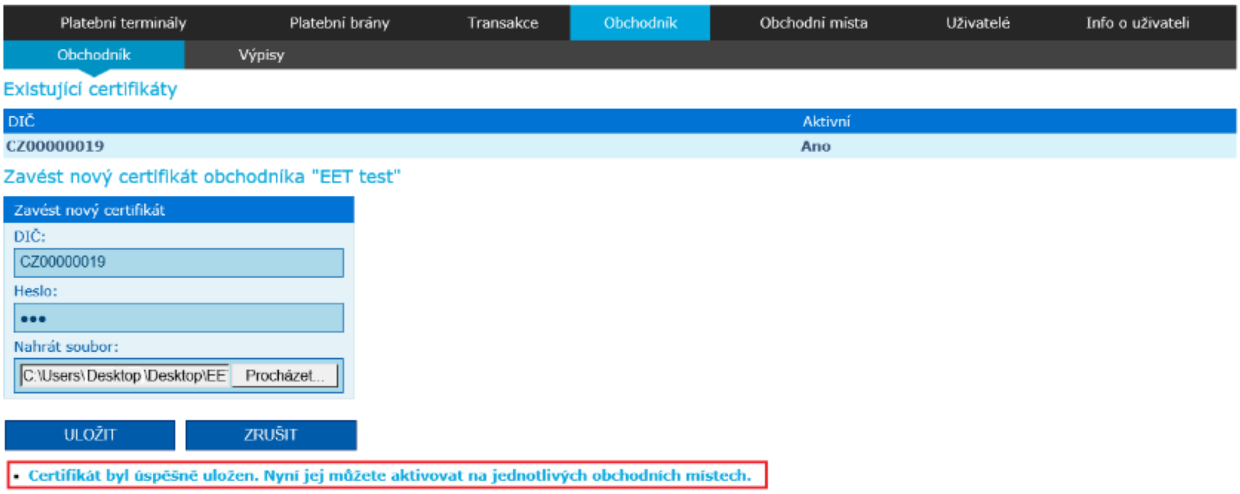
Figure 26: Merchant Tab

Click to display the Certificates page. Embedded certificates are displayed at the top of the application. The active certificate is marked in bold.

A new certificate is inserted at the bottom of the application. The VAT number is pre-filled by a bank, in case the VAT number field is displayed blank, it is necessary to contact by a bank by email

In the next field called Password it is necessary to fill in the obtained password of the certificate from the financial administration. Use the following field to select the certificate that you have already downloaded on your computer.

Click the Save button to upload the certificate to the POS Merchant system, figure 27.



Existující certifikáty

DIČ	Aktivní
CZ00000019	**Ano**

Zavést nový certifikát obchodníka "EET test"

Zavést nový certifikát

DIČ: CZ00000019

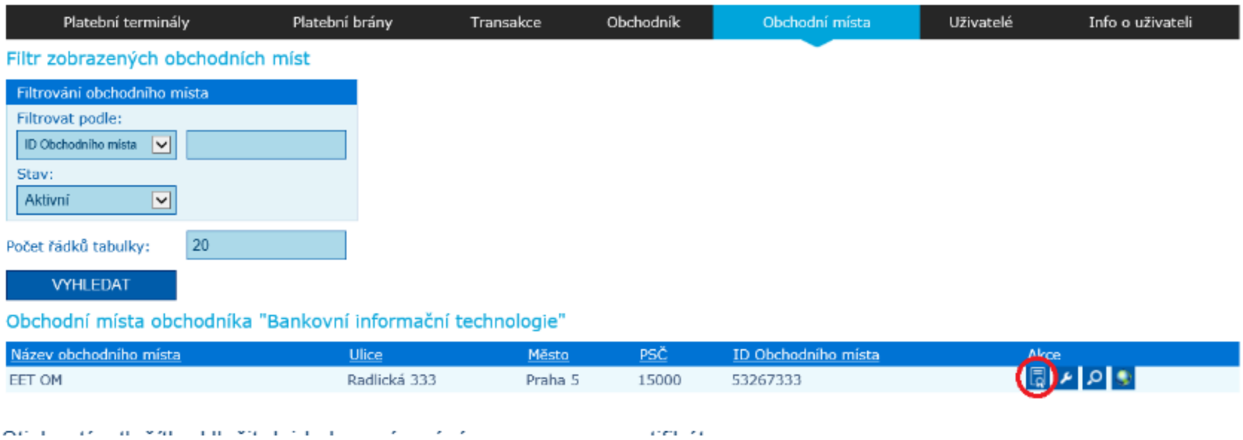
Heslo: •••

Nahrát soubor: C:\Users\Desktop\Desktop\EE

• Certifikát byl úspěšně uložen. Nyní jej můžete aktivovat na jednotlivých obchodních místech.

Figure 27: Certificate Page

After successful upload of the certificate, it is necessary to activate individual points of sale to receive ERS. In the POS Merchant application, go to the Business Locations tab, figure 28.



Filtr zobrazených obchodních míst

Filtrování obchodního místa

Filtrovat podle: ID Obchodního místa

Stav: Aktivní

Počet řádků tabulky: 20

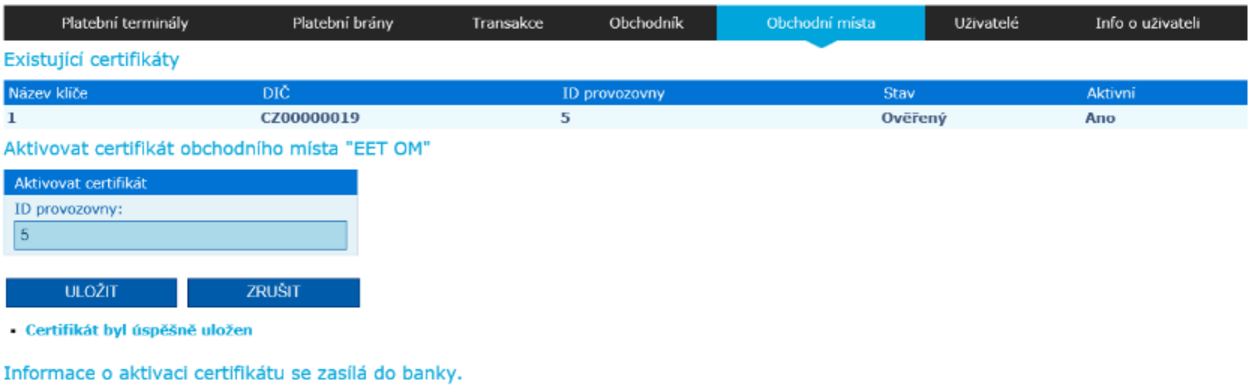
VYHLEDAT

Obchodní místa obchodníka "Bankovní informační technologie"

Název obchodního místa	Ulice	Město	PSČ	ID Obchodního místa	Akce
EET OM	Radlická 333	Praha 5	15000	53267333	

Figure 28: Business Location tab

Pressing the Save button will pair the establishment with the certificate, figure 29.



Existující certifikáty

Název klíče	DIČ	ID provozovny	Stav	Aktivní
1	CZ00000019	5	Ověřený	Ano

Aktivovat certifikát obchodního místa "EET OM"

Aktivovat certifikát

ID provozovny: 5

ULOŽIT ZRUŠIT

- Certifikát byl úspěšně uložen

Informace o aktivaci certifikátu se zasílá do banky.

Figure 29: Pairing Establishment with Certificate

For each individual Point of Sale, it is necessary to fill in the ID of the establishment, which you obtained by registering on the financial administration server.

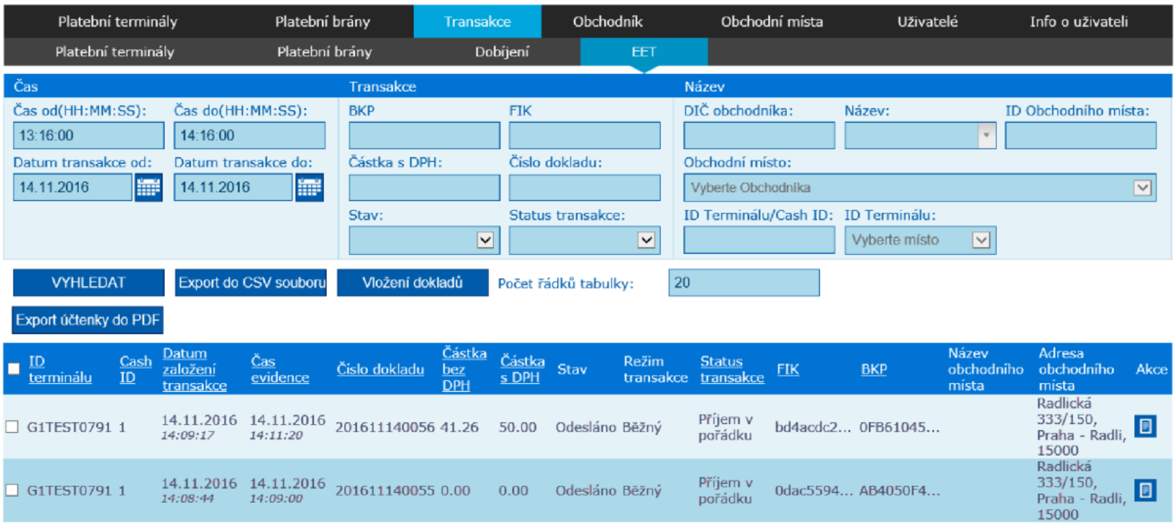
If the currently activated certificate contains multiple keys, you select which one to use for the given business location. If the certificate contains only one key, it will be selected automatically

During the introduction of the certificate, a test transaction is sent to the MFČR, which verifies the correctness of the entered data and the validity of the certificate. This transaction is not recorded among the merchant's sales.

When activating a certificate at a point of sale, the active certificate is used, which is currently loaded on the Store tab. After activation, the bank needs to reconfigure the terminal settings. Therefore, after successful verification and activation of the certificate at the point of sale, a notification is sent to the bank, after which the bank reconfigures the terminals.

5.2 Display of ERS transactions in the POS Merchant application

Filtering ERS transactions in the POS Merchant user interface is similar to other transactions. Filtering is performed by time from to one month, then by various statuses or states of transactions, amounts, document numbers, etc., figure 30.

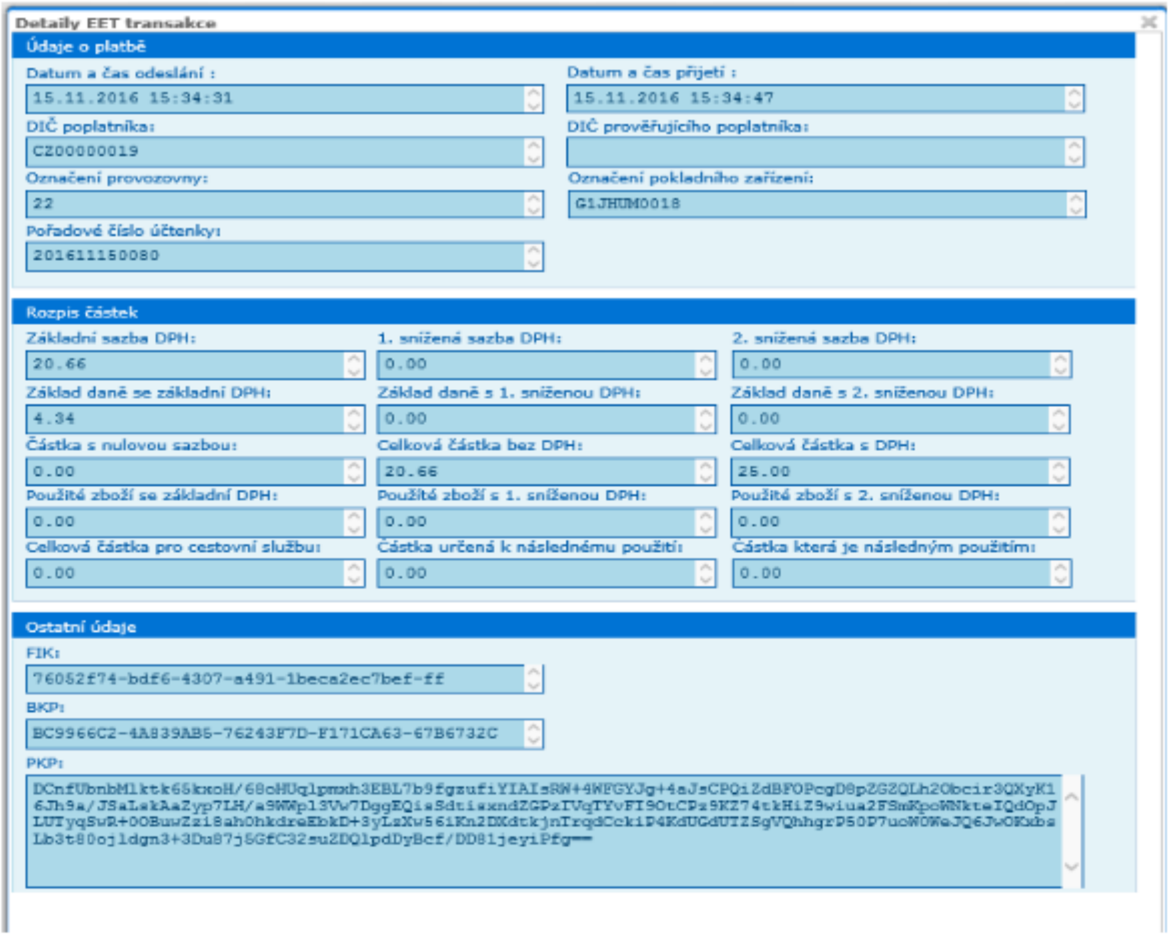


Čas od(HH:MM:SS): 13:16:00
Čas do(HH:MM:SS): 14:16:00
Datum transakce od: 14.11.2016
Datum transakce do: 14.11.2016
BKP:
FIK:
Částka s DPH:
Číslo dokladu:
Stav:
Status transakce:
DIČ obchodníka:
Název:
ID Obchodního místa:
Obchodní místo: Vyberte Obchodníka
ID Terminálu/Cash ID:
ID Terminálu: Vyberte místo
Počet řádků tabulky: 20

□	ID terminálu	Cash ID	Datum založení transakce	Čas evidence	Číslo dokladu	Částka bez DPH	Částka s DPH	Stav	Režim transakce	Status transakce	FIK	BKP	Název obchodního místa	Adresa obchodního místa	Akce
□	G1TEST0791	1	14.11.2016 *14:09:17*	14.11.2016 *14:11:20*	201611140056	41.26	50.00	Odesláno	Běžný	Příjem v pořádku	bd4acdc2...	0FB61045...		Radlická 333/150, Praha - Radli, 15000	🗎
□	G1TEST0791	1	14.11.2016 *14:08:44*	14.11.2016 *14:09:00*	201611140055	0.00	0.00	Odesláno	Běžný	Příjem v pořádku	0dac5594...	AB4050F4...		Radlická 333/150, Praha - Radli, 15000	🗎

Figure 30: ERS tab

After the search, the table is displayed to the user with the relevant transactions. Because transaction data is relatively large, the table allows you to view transaction details in the last column. After clicking on the detail icon for the given line, the modal window is displayed to the user, in which more detailed data about the transaction are displayed, such as amounts in all tax categories, FIC, BKP values, etc., figure 31.



Detaily EET transakce

Údaje o platbě

Datum a čas odeslání : 15.11.2016 15:34:31	Datum a čas přijetí : 15.11.2016 15:34:47
DIČ poplatníka: CZ00000019	DIČ prověřujícího poplatníka:
Označení provozovny: 22	Označení pokladního zařízení: G1JHUM0018
Pořadové číslo účtenky: 201611150080	

Rozpis částek

Základní sazba DPH: 20.66	1. snížená sazba DPH: 0.00	2. snížená sazba DPH: 0.00
Základ daně se základní DPH: 4.34	Základ daně s 1. sníženou DPH: 0.00	Základ daně s 2. sníženou DPH: 0.00
Částka s nulovou sazbou: 0.00	Celková částka bez DPH: 20.66	Celková částka s DPH: 25.00
Použité zboží se základní DPH: 0.00	Použité zboží s 1. sníženou DPH: 0.00	Použité zboží s 2. sníženou DPH: 0.00
Celková částka pro cestovní službu: 0.00	Částka určená k následnému použití: 0.00	Částka která je následným použitím: 0.00

Ostatní údaje

FIK:
76052f74-bdf6-4307-a491-1beca2ec7bef-ff

BKP:
BC9966C2-4A839AB5-76243F7D-F171CA63-67B6732C

PKP:

DCnfUbnbMlktk65kxoH/68oHUqlpmxh3EBL7b9fgzufiYIAIsRW+4WFGYJg+4aJsCPQi2dBFOPcgD8p2G2QLh2Obcir3QXyK1
6Jh9a/JSaLakAaZyp7LH/a9WWpl3Vw7DggEQiaSdtiaxndZGPzIVqTYvFI9OtCPz9KZ74tkHiZ9wiua2FSmKpoWNkteIQdOpJ
LUTyqSwR+0OBuwZzi8ah0hkdreEbkD+3yLzXw56iKn2DXdtkjnTrqdCckiP4KdUGdUTZSgVQhhgrP50P7uoW0WeJQ6JwOKxbs
Lb3t80ojldgn3+3Du87j5GfC32su2DQlpdDyBcf/DD81jeyiPfg==

Figure 31: Transaction Tab

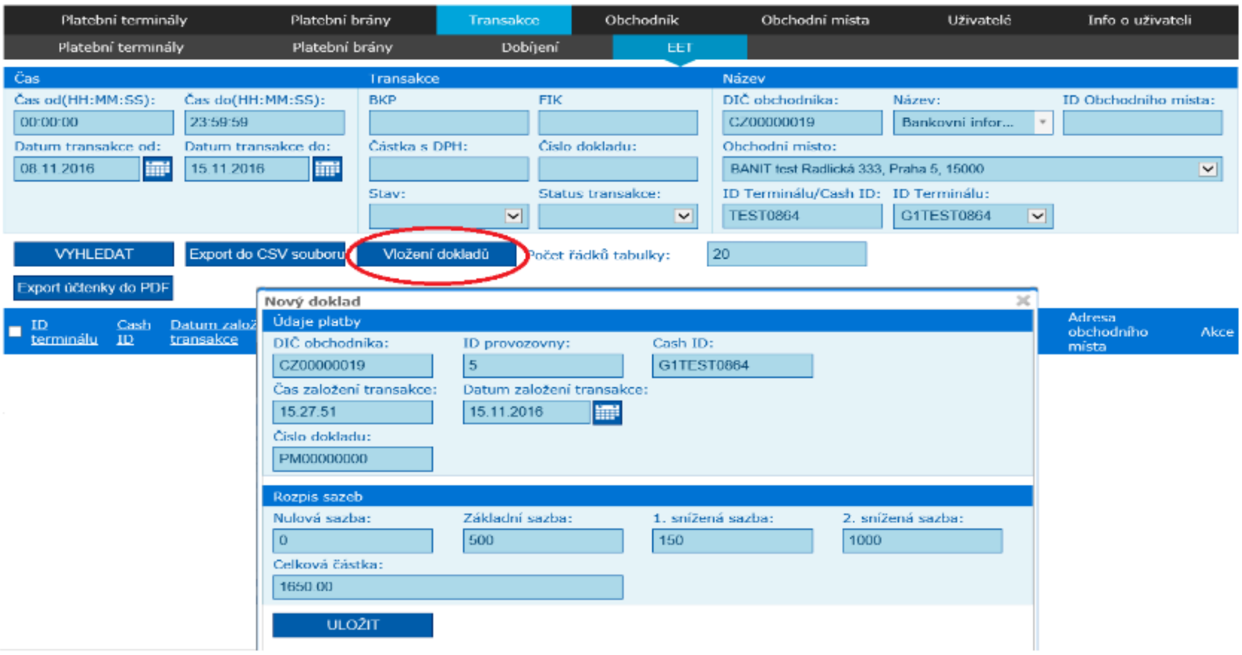
The meaning of individual statuses is as follows:

- Undefined - default state at the time of transaction creation.
- Receipt of order - transaction registered online in MFČR.
- Not accepted by terminal - terminal did not confirm receipt of response from server.
- Not sent to ERS – failed to establish connection with the MFČR server.
- Further sending – transaction sent to the terminal repeatedly.

The meaning of statuses other than "Receipt of order" is informative. Only the filled in value of the fiscal identification code (FIC) clearly indicates whether the transaction was successfully registered in the Ministry of Finance of the Czech Republic.

In contrast to other transactions, the " Insert documents” button has been added, which allows you to create a transaction in the event of a terminal failure. After clicking on this button, the trader is pre-filled with data about his certificate (VAT number, CashID data, etc.). In order to be able to pre-fill this data, it is necessary to select the merchant, the point of sale and the terminal in the filter. The receipt number is also generated.

It is then up to the user to enter the breakdown of sales in individual tax rates. For clarity, the sum of these amounts is displayed. After saving the document, the user is informed whether the transaction was saved successfully or not, figure 32.



Čas od(HH:MM:SS): 00:00:00
Čas do(HH:MM:SS): 23:59:59
Datum transakce od: 08.11.2016
Datum transakce do: 15.11.2016
BKP
FIK
Částka s DPH:
Číslo dokladu:
Stav:
Status transakce:
DIČ obchodníka: CZ00000019
Název: Bankovni infor...
ID Obchodního místa:
Obchodní místo: BANIT test Radlická 333, Praha 5, 15000
ID Terminálu/Cash ID: TEST0864
ID Terminálu: G1TEST0864
Počet řádků tabulky: 20

DIČ obchodníka: CZ00000019
ID provozovny: 5
Cash ID: G1TEST0864
Čas založení transakce: 15.27.51
Datum založení transakce: 15.11.2016
Číslo dokladu: PM00000000

Nulová sazba: 0
Základní sazba: 500
1. snížená sazba: 150
2. snížená sazba: 1000
Celková částka: 1650.00

Figure 32: Insert documents

II. TOOLS

6 TOOLS PREFERENCE

6.1 Why Java

So far, Java is the first choice among backend developers in the world. With Java 8 featured release in 2014 making it possible to write functional-oriented applications, making it even easier and robust. Over 3 billion devises use Java in some form making this one the most popular runtime environment.

Major advantages:

- Can be ran in any system with a JVM, hence offering high performance and making it very portable.
- Compile-time and runtime checking make it a very secured language.
- Garbage collection, memory management make it robust language.
- Scalability.
- Multi-threading.
- Ecosystem – great IDEs of choice, Maven or Gradle for dependency management, Spring framework/boot, etc.

6.2 Why Wicket

Wicket is a framework for building Java Web Applications. It was relatively popular among Java developers working with web applications. The aim of the framework is to simplify development of web interfaces using only Java code and XHTML similar. No need XML configurations and JavaScript.

Major advantages:

- Writing AJAX applications without a line of JavaScript code
- Web pages are objects therefore syntax would be familiar to Java Developers
- Testing becomes much easier

6.3 Why JPA

The Java Persistent API is considered to be a standard approach for ORM. It allows POJO to be persisted without a need for any methods nor interfaces.

Major advantages:

- Database independent.
- Ability to easily switch JPA implementations.
- Connection between domain model and databases must be defined just once.

- Simplicity compared to other approaches.

6.4 Why Maven

Maven is a project management tool which is used for dependency, documentation and project builds.

Major advantages:

- Maven adds dependencies set in pom file.
- Possibility to start projects on different machines without handling configuration (if one is set beforehand).

6.5 Why Spring

Spring is the most popular framework for Java. It eases the configuration process in order to avoid writing a lot of boilerplate code.

Major advantages:

- Productivity increase.
- Provide embedded HTTP server.
- Provides a lot of plugins using Maven or Gradle.
- Provides Command Line Interface to test applications from command prompt.

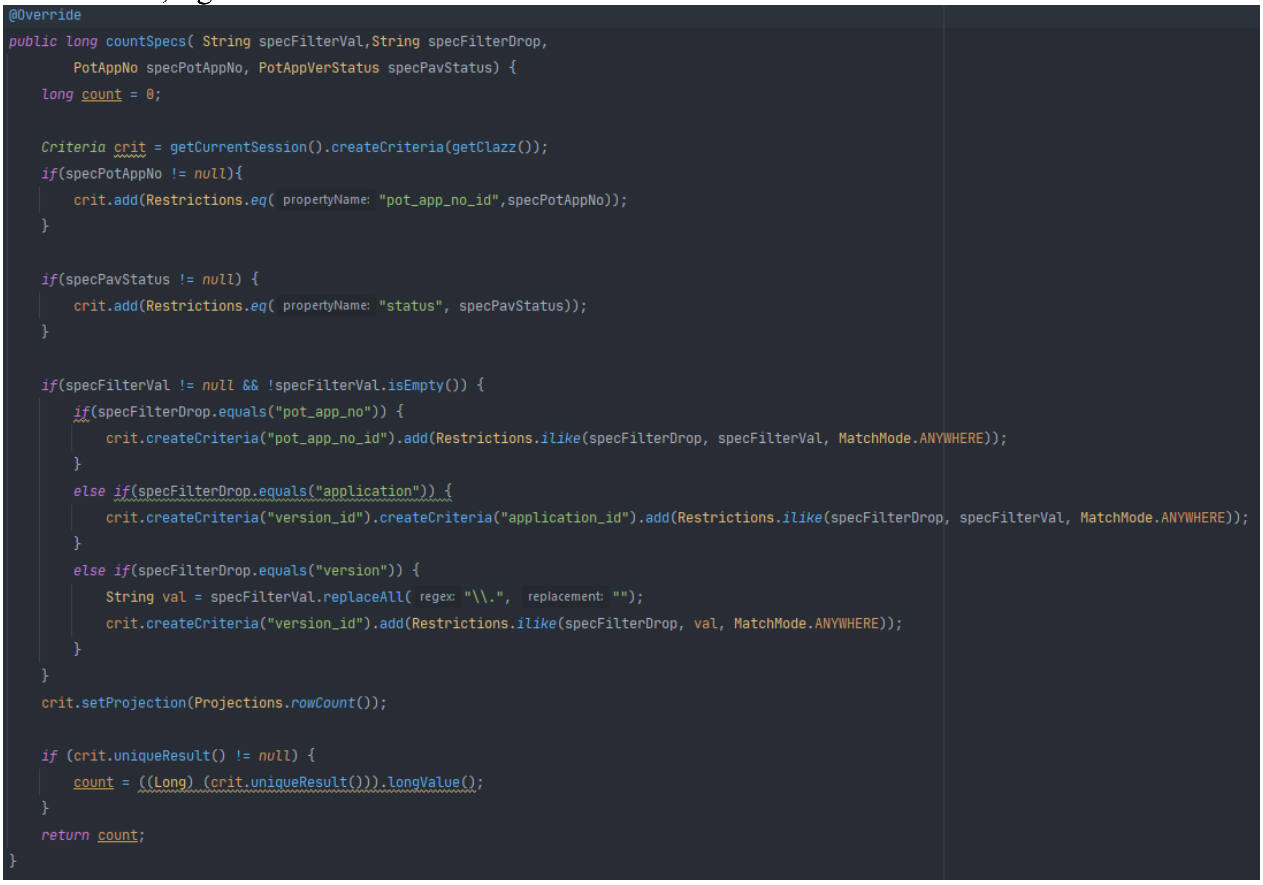
III. ANALYSIS

7 MERCHANT REPOSITORY

Terminal management system is the part of application which manages communication between Databases and POS Merchant. The main point of TMS is to separate DB from the backend

7.1 Dataflow with filters and predefined methods

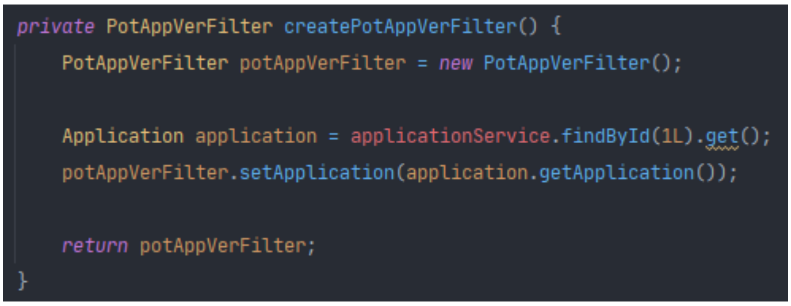
Before the update, Hibernate Restrictions class was used for filtering the output from the database, figure 33.



```
@Override
public long countSpecs( String specFilterVal,String specFilterDrop,
        PotAppNo specPotAppNo, PotAppVerStatus specPavStatus) {
    long count = 0;

    Criteria crit = getCurrentSession().createCriteria(getClazz());
    if(specPotAppNo != null){
        crit.add(Restrictions.eq( propertyName: "pot_app_no_id",specPotAppNo));
    }

    if(specPavStatus != null) {
        crit.add(Restrictions.eq( propertyName: "status", specPavStatus));
    }

    if(specFilterVal != null && !specFilterVal.isEmpty()) {
        if(specFilterDrop.equals("pot_app_no")) {
            crit.createCriteria("pot_app_no_id").add(Restrictions.ilike(specFilterDrop, specFilterVal, MatchMode.ANYWHERE));
        }
        else if(specFilterDrop.equals("application")) {
            crit.createCriteria("version_id").createCriteria("application_id").add(Restrictions.ilike(specFilterDrop, specFilterVal, MatchMode.ANYWHERE));
        }
        else if(specFilterDrop.equals("version")) {
            String val = specFilterVal.replaceAll( regex: "\\.",  replacement: "");
            crit.createCriteria("version_id").add(Restrictions.ilike(specFilterDrop, val, MatchMode.ANYWHERE));
        }
    }
    crit.setProjection(Projections.rowCount());

    if (crit.uniqueResult() != null) {
        count = ((Long) (crit.uniqueResult())).longValue();
    }
    return count;
}
```

Figure 33: Old Implementation of Filters

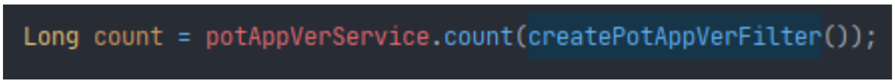
Now with JPA CriteriaBuilder we define JPA filters as explained below, figure 34.
This class creates filter, with instance of Application domain class where the records from databases can be filtered before rendering.



```
private PotAppVerFilter createPotAppVerFilter() {
    PotAppVerFilter potAppVerFilter = new PotAppVerFilter();

    Application application = applicationService.findById(1L).get();
    potAppVerFilter.setApplication(application.getApplication());

    return potAppVerFilter;
}
```

Figure 34: New Implementation of Filters

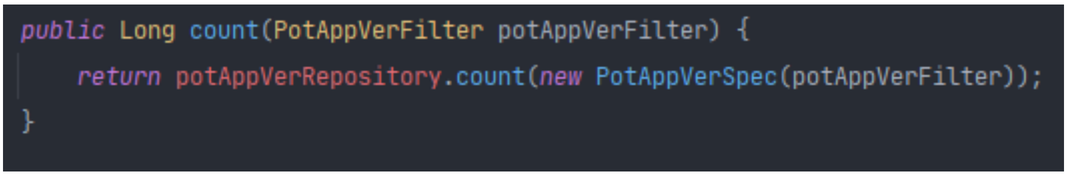
Calling the service count with the created Filter, figure 35.



```java
Long count = potAppVerService.count(createPotAppVerFilter());
```

Figure 35: New Implementation of Filters

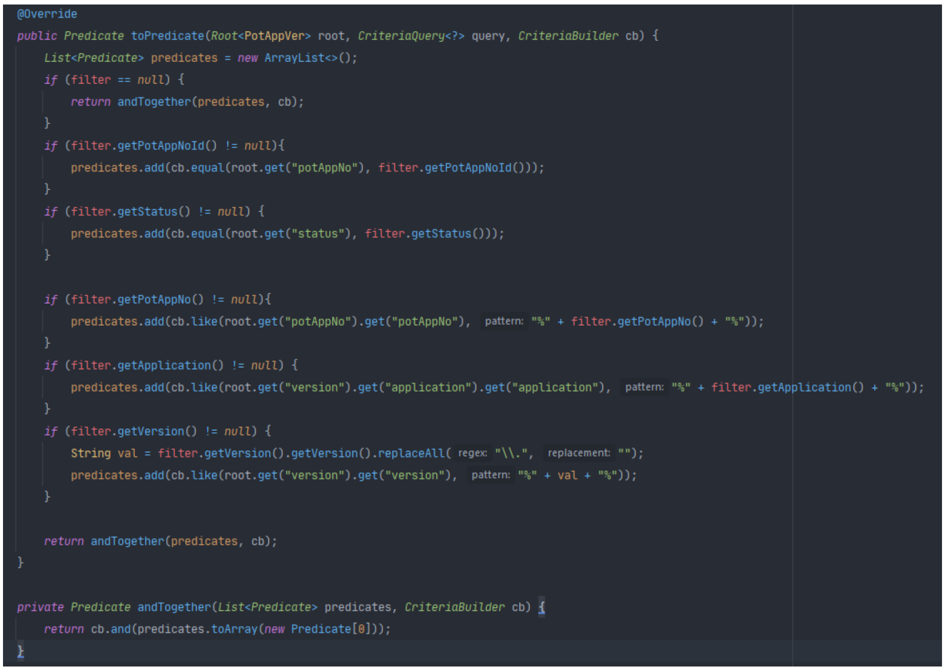
This is the service that is called and then the service is calling the Repository with the new instance from specifications with the filter we have created, figure 36.



```java
public Long count(PotAppVerFilter potAppVerFilter) {
    return potAppVerRepository.count(new PotAppVerSpec(potAppVerFilter));
}
```

Figure 36: New Implementation of Filters

It accepts the filter and runs it through the instance of functional interface Predicate, figure 37.

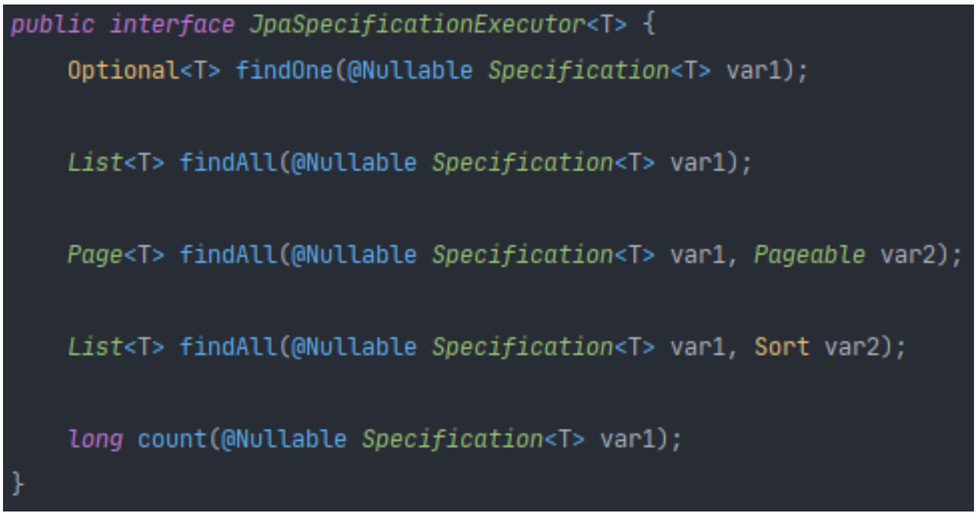


```java
@Override
public Predicate toPredicate(Root<PotAppVer> root, CriteriaQuery<?> query, CriteriaBuilder cb) {
    List<Predicate> predicates = new ArrayList<>();
    if (filter == null) {
        return andTogether(predicates, cb);
    }
    if (filter.getPotAppNoId() != null){
        predicates.add(cb.equal(root.get("potAppNo"), filter.getPotAppNoId()));
    }
    if (filter.getStatus() != null) {
        predicates.add(cb.equal(root.get("status"), filter.getStatus()));
    }

    if (filter.getPotAppNo() != null){
        predicates.add(cb.like(root.get("potAppNo").get("potAppNo"),  pattern: "%" + filter.getPotAppNo() + "%"));
    }
    if (filter.getApplication() != null) {
        predicates.add(cb.like(root.get("version").get("application").get("application"),  pattern: "%" + filter.getApplication() + "%"));
    }
    if (filter.getVersion() != null) {
        String val = filter.getVersion().getVersion().replaceAll( regex: "\\.",  replacement: "");
        predicates.add(cb.like(root.get("version").get("version"),  pattern: "%" + val + "%"));
    }

    return andTogether(predicates, cb);
}

private Predicate andTogether(List<Predicate> predicates, CriteriaBuilder cb) {
    return cb.and(predicates.toArray(new Predicate[0]));
}
```

Figure 37: New Implementation of Filters

Method „toPredicate“ accepts the filters, checking if there is a reason to do filterring and finally returning a list of filters to be applied.

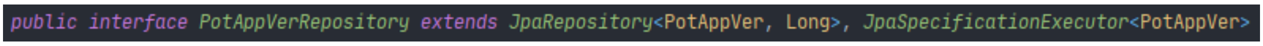
The count method is already predefined In the JpaSpecificationsExecutor class from Spring Data Jpa library, figure 38.



```
public interface JpaSpecificationExecutor<T> {
    Optional<T> findOne(@Nullable Specification<T> var1);

    List<T> findAll(@Nullable Specification<T> var1);

    Page<T> findAll(@Nullable Specification<T> var1, Pageable var2);

    List<T> findAll(@Nullable Specification<T> var1, Sort var2);

    long count(@Nullable Specification<T> var1);
}
```

Figure 38: JPA predify method

Also in order for JPA to work we need to extends the JPA repository specifying the domaim class and the type of primary key of the database table, along with JpaSpecificationsExecutor to call count method, figure 39.

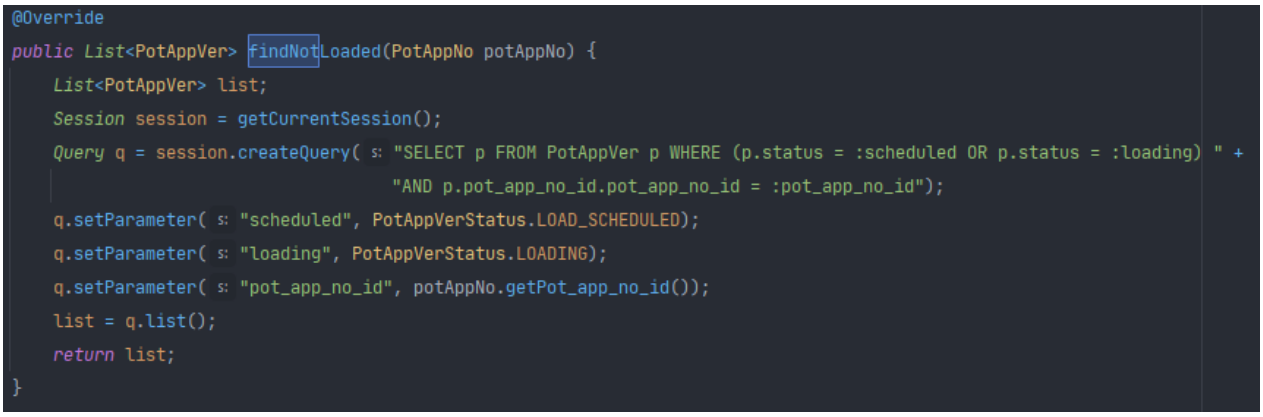


```
public interface PotAppVerRepository extends JpaRepository<PotAppVer, Long>, JpaSpecificationExecutor<PotAppVer>
```

Figure 39: New Implementation of Filters

7.2 Dataflow with custom queries

Before the update, Hibernate Sessions was used to get a physical connection with database, figure 40.

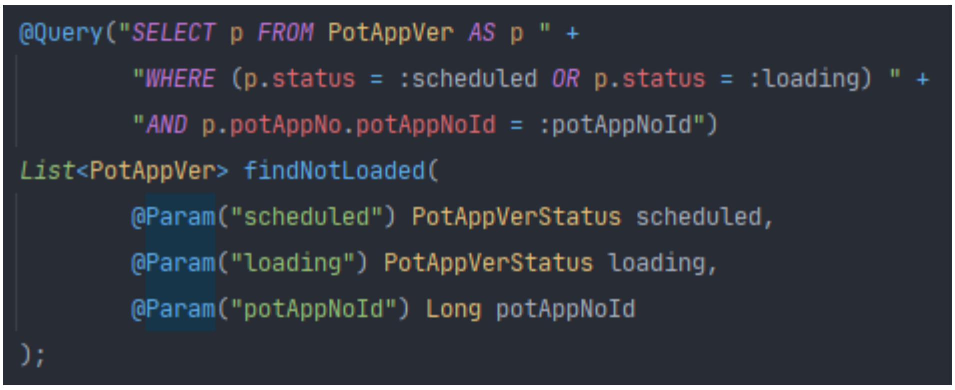


```
@Override
public List<PotAppVer> findNotLoaded(PotAppNo potAppNo) {
    List<PotAppVer> list;
    Session session = getCurrentSession();
    Query q = session.createQuery( s: "SELECT p FROM PotAppVer p WHERE (p.status = :scheduled OR p.status = :loading) " +
                                      "AND p.pot_app_no_id.pot_app_no_id = :pot_app_no_id");
    q.setParameter( s: "scheduled", PotAppVerStatus.LOAD_SCHEDULED);
    q.setParameter( s: "loading", PotAppVerStatus.LOADING);
    q.setParameter( s: "pot_app_no_id", potAppNo.getPot_app_no_id());
    list = q.list();
    return list;
}
```

Figure 40: Old Implementation of Custom Queries

Hibernate sessions are not as verbose, so the it was decided to switch to Spring Data JPA.

The dataflow is still the same and Repository but the custom query we have created that looks more readable and reusable, figure 41.



```
@Query("SELECT p FROM PotAppVer AS p " +
        "WHERE (p.status = :scheduled OR p.status = :loading) " +
        "AND p.potAppNo.potAppNoId = :potAppNoId")
List<PotAppVer> findNotLoaded(
        @Param("scheduled") PotAppVerStatus scheduled,
        @Param("loading") PotAppVerStatus loading,
        @Param("potAppNoId") Long potAppNoId
);
```

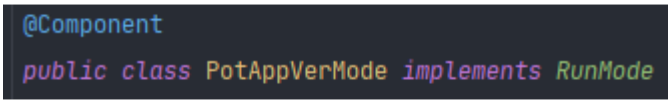
Figure 41: New Implementation of Custom Queries

Named Parameters will pass method parameters to the query matching in the method declaration.

For the queries above „status“ method parameter is assigned to named parameter to match the name, set in query, therefore „loading“ and „potAppNoId“ are assigned to query parameters for „loading“ and „potAppNoId“ to match the names in the query.

7.3 Testing

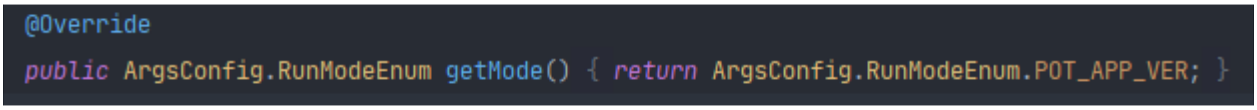
For testing purposed Mode class has to be created for each repository with implementation of RunMode, figure 42.



```
@Component
public class PotAppVerMode implements RunMode
```

Figure 42: Implementation for Testing

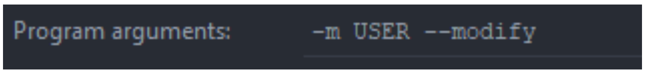
RunModeEnum is the enum which holds all the names of the Repositories. Here the mode is getting by returning the name of the repository so the RunMode knows which mode to start for testing purposes, figure 43.



```
@Override
public ArgsConfig.RunModeEnum getMode() { return ArgsConfig.RunModeEnum.POT_APP_VER; }
```

Figure 43: Implementation for Testing

In Promag Arguments we spicify which mode to run, figure 44.

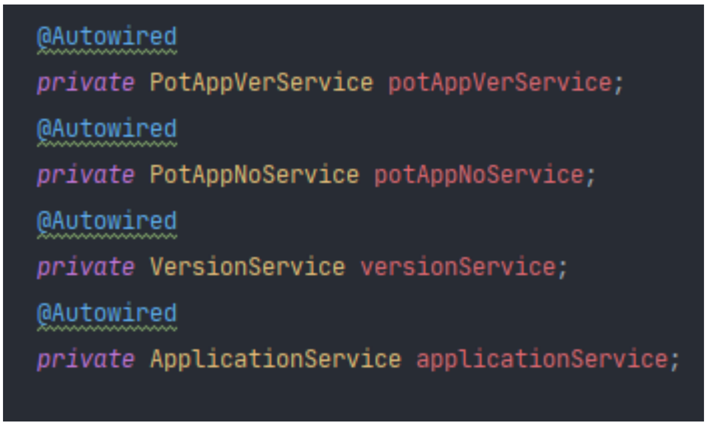


Program arguments: -m USER --modify

Figure 44: Configuration of Program Arguments for Testing

- -m stands for the name of the mode;
- --modify will allow running methods that will modify the data in database.

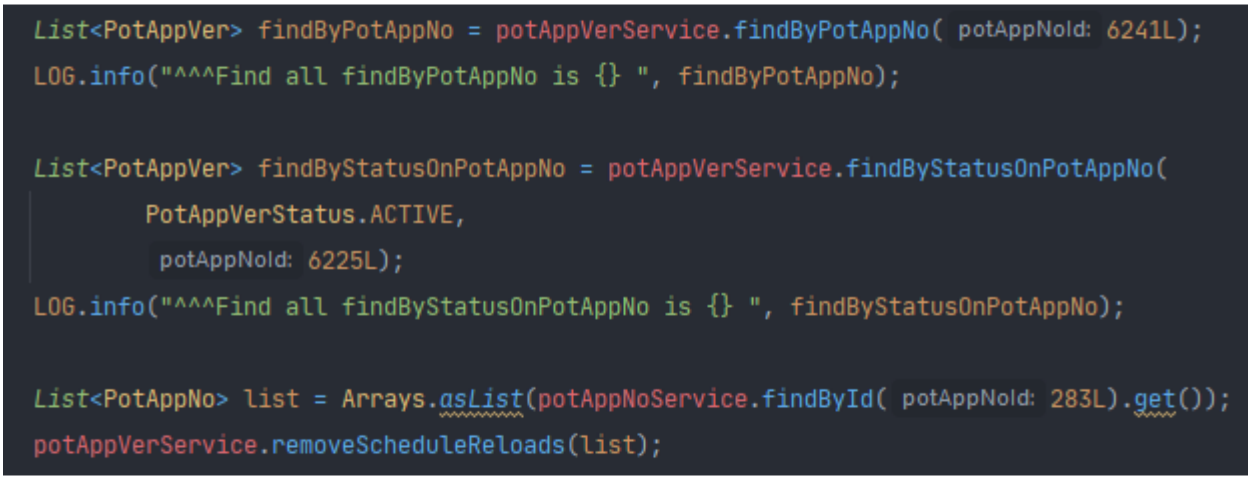
Using Spring Boot 2 the services were autowired for given repository. Figure 45.



```
@Autowired
private PotAppVerService potAppVerService;
@Autowired
private PotAppNoService potAppNoService;
@Autowired
private VersionService versionService;
@Autowired
private ApplicationService applicationService;
```

Figure 45: Implementation for Testing

Then the methods from the autowired services are called with given parameters and then the results are logged into the console so the output of the tests can be observed, figure 46.



```
List<PotAppVer> findByPotAppNo = potAppVerService.findByPotAppNo( potAppNoId: 6241L);
LOG.info("^^^Find all findByPotAppNo is {} ", findByPotAppNo);

List<PotAppVer> findByStatusOnPotAppNo = potAppVerService.findByStatusOnPotAppNo(
        PotAppVerStatus.ACTIVE,
         potAppNoId: 6225L);
LOG.info("^^^Find all findByStatusOnPotAppNo is {} ", findByStatusOnPotAppNo);

List<PotAppNo> list = Arrays.asList(potAppNoService.findById( potAppNoId: 283L).get());
potAppVerService.removeScheduleReloads(list);
```

Figure 46: Implementation for Testing

8 POS MERCHANT REPOSITORY

Regarding the POS Merchant Repository, the main goal was to migrate from Wicket 1.5 to Wicket 8.6.0. For smoother migration and for good behavior changes it was decided to, firstly, change from wicket 1.5 to wicket 6.30 and after successful migration from Wicket 6.30 migrate to Wicket 8.6.0.

For better understanding, the structure of the wicket framework is described below.

The main advantage of using Wicket is Wicket Components. Wicket Components consist of a Java class, the HTML markup, and a model. Models are facade that components use to access the data.

The wickets structure implements comfortable separation of concerns for component and the output on the web, increasing reusability of code. Java represents component logic and in order to add it to the web it is just needed to be annotated in HTML markup with required components written in Java, figure 47.

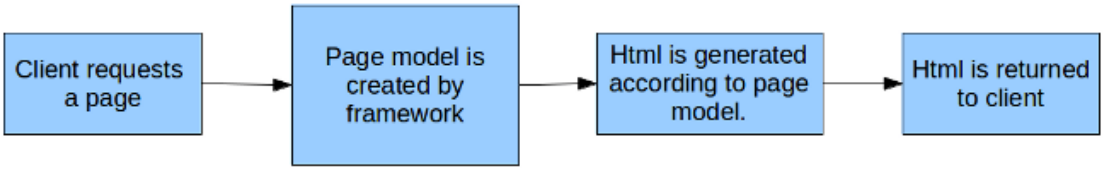

Client requests a page
Page model is created by framework
Html is generated according to page model.
Html is returned to client

Figure 47: Schema of page request handling for a component-oriented framework

Because Wicket is the component-oriented framework, compared to the frameworks such as Angular or React, the model of required pages is built on the server-side and generates HTML according to the model. Considering JavaScript DOM, Wicket works in the inverse way, meaning:

- Firstly, it is built on the server-side
- Secondly, it is built before HTML is sent to the client
- Lastly, in order to generate HTML code, it first must use a model

What makes Wicket great are the benefits which are:

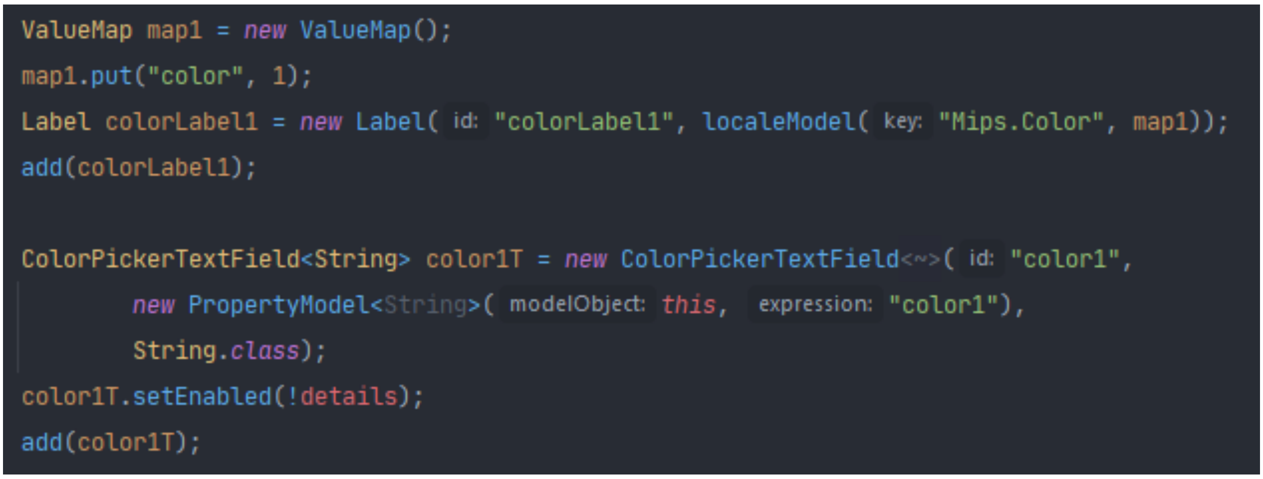
- Web pages are objects, meaning that web pages are objects instances rather than text fields and therefore we can take advantage of OOP to structure web pages and their components.
- There is no need to worry about application’s state. Java objects are considered to be stateful entities, because they keep the reference to other objects inside the objects;
- According to the previous benefits, testing web applications becomes easier.

As stated before, the main challenge was migration. There were many deprecated methods and classes which had to be rewritten for application to work faster and more secure.

Additional libraries which were working with Wicket 1.5 were not updated to newer style of coding, a lot of logic had to be rewritten.

As an example, Color Picker for styling Payment Web Application for owners of Merchant, could not been updated due to some of the changes in Wicket AJAX API. The task was to write a new Color Picker from scratch, which was a challenge. Because frameworks such as Angular and React and VueJs gained so much popularity, Wicket started losing its audience which led to big decrease of support in forums in past several years.

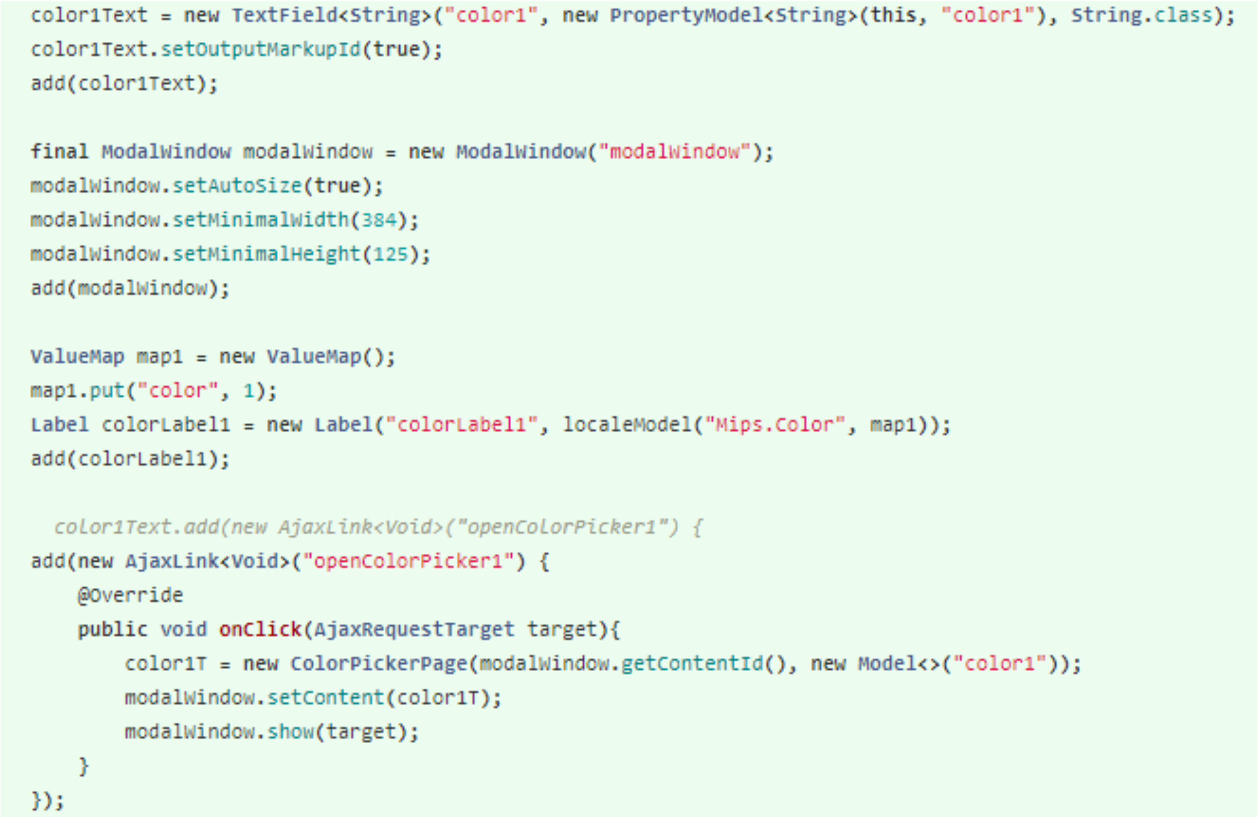
Before the update ColorPicker used a third-party library implementing much of the functionality for color picker, figure 48.



```
ValueMap map1 = new ValueMap();
map1.put("color", 1);
Label colorLabel1 = new Label( id: "colorLabel1", localeModel( key: "Mips.Color", map1));
add(colorLabel1);

ColorPickerTextField<String> color1T = new ColorPickerTextField<~>( id: "color1",
        new PropertyModel<String>( modelObject: this,  expression: "color1"),
        String.class);
color1T.setEnabled(!details);
add(color1T);
```

Figure 48: Implementation of Color Picker

After an update, the color picker third-party library was not supported due to some of the deprecated methods it used from Wicket 1.5. The challenge was to implement similar logic of third-party library, which was used before from scratch, figure 49.

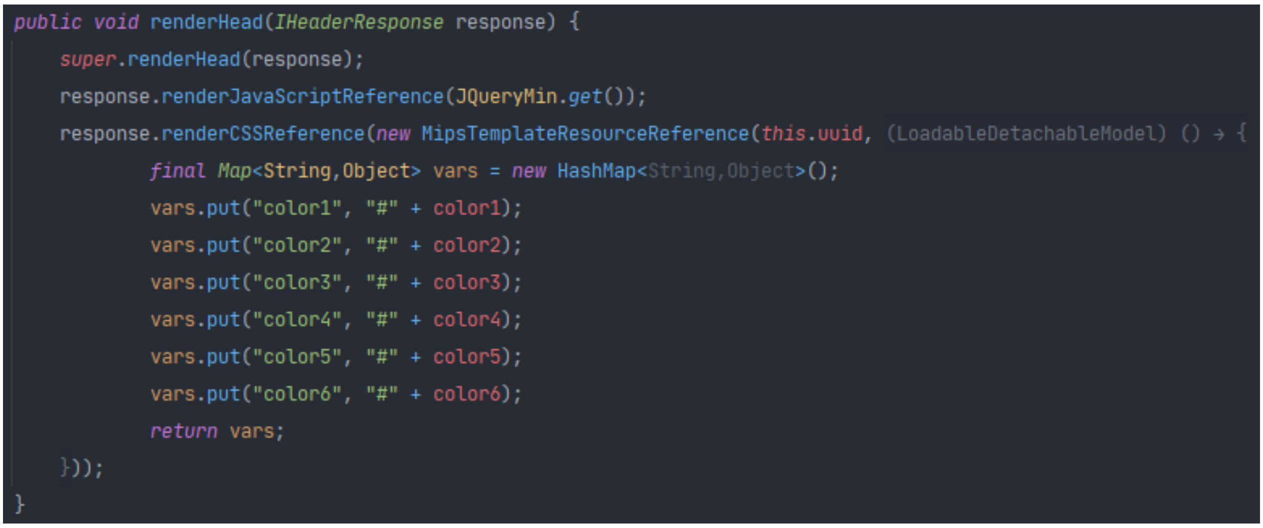


```
color1Text = new TextField<String>("color1", new PropertyModel<String>(this, "color1"), String.class);
color1Text.setOutputMarkupId(true);
add(color1Text);

final ModalWindow modalWindow = new ModalWindow("modalWindow");
modalWindow.setAutoSize(true);
modalWindow.setMinimalWidth(384);
modalWindow.setMinimalHeight(125);
add(modalWindow);

ValueMap map1 = new ValueMap();
map1.put("color", 1);
Label colorLabel1 = new Label("colorLabel1", localeModel("Mips.Color", map1));
add(colorLabel1);

  color1Text.add(new AjaxLink<Void>("openColorPicker1") {
add(new AjaxLink<Void>("openColorPicker1") {
    @Override
    public void onClick(AjaxRequestTarget target){
        color1T = new ColorPickerPage(modalWindow.getContentId(), new Model<>("color1"));
        modalWindow.setContent(color1T);
        modalWindow.show(target);
    }
});
```

Figure 49: New implementation on color picker

ModalWindow had to be implemented along with AjaxLink component. Before it was implemented by third-party library.

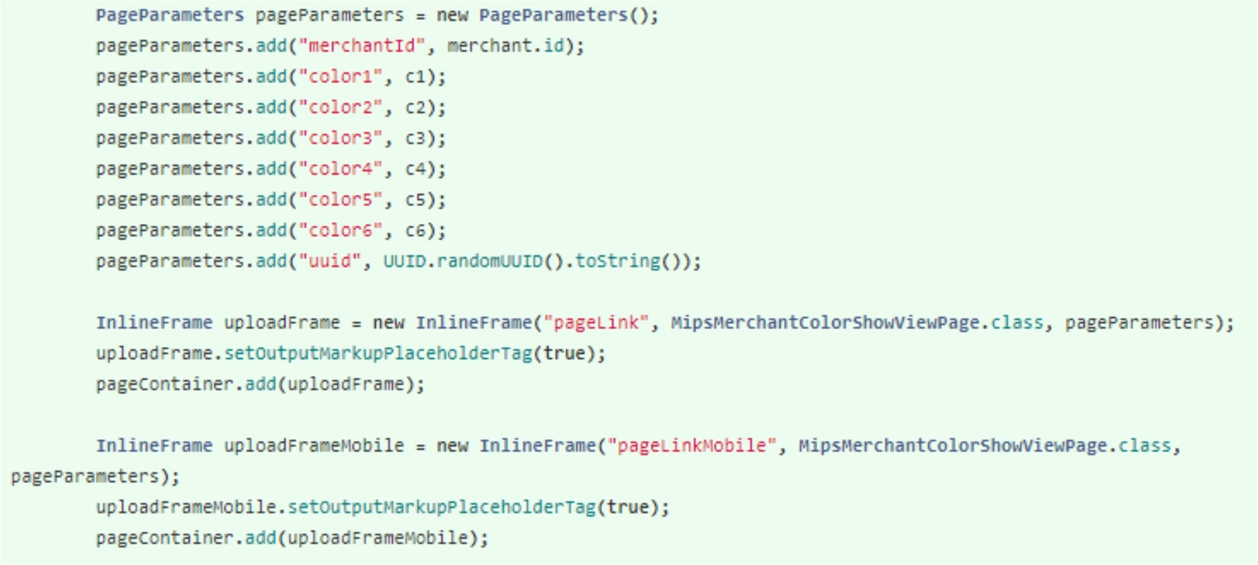
Before PageParameters API was improved, the hash map with references to newly set colors by user, were obtained from the object to change save the colors and apply them for rendering, figure 50.



```java
public void renderHead(IHeaderResponse response) {
    super.renderHead(response);
    response.renderJavaScriptReference(JQueryMin.get());
    response.renderCSSReference(new MipsTemplateResourceReference(this.uuid, (LoadableDetachableModel) () → {
            final Map<String,Object> vars = new HashMap<String,Object>();
            vars.put("color1", "#" + color1);
            vars.put("color2", "#" + color2);
            vars.put("color3", "#" + color3);
            vars.put("color4", "#" + color4);
            vars.put("color5", "#" + color5);
            vars.put("color6", "#" + color6);
            return vars;
    }));
}
```

Figure 50: HashMap of values

Improvement of PageParameters with new Wicket 8.60, motivated to slightly change the structure using PageParameters, figure 51.



```java
        PageParameters pageParameters = new PageParameters();
        pageParameters.add("merchantId", merchant.id);
        pageParameters.add("color1", c1);
        pageParameters.add("color2", c2);
        pageParameters.add("color3", c3);
        pageParameters.add("color4", c4);
        pageParameters.add("color5", c5);
        pageParameters.add("color6", c6);
        pageParameters.add("uuid", UUID.randomUUID().toString());

        InlineFrame uploadFrame = new InlineFrame("pageLink", MipsMerchantColorShowViewPage.class, pageParameters);
        uploadFrame.setOutputMarkupPlaceholderTag(true);
        pageContainer.add(uploadFrame);

        InlineFrame uploadFrameMobile = new InlineFrame("pageLinkMobile", MipsMerchantColorShowViewPage.class,
pageParameters);
        uploadFrameMobile.setOutputMarkupPlaceholderTag(true);
        pageContainer.add(uploadFrameMobile);
```

Figure 51: PageParameters implementation

CONCLUSION

The main goal was the modernization of currently used Payment Web Application replacing the old and some not supported libraries.

Migration of Wicket 1.5 which was based on Java 6 and Spring 4 to Wicket 8.6 which is using Java 8 and Spring 5, significantly increased security enhancement. Support of Lambda and Streams along with new Wicket features in some critical parts, gave application a new style of code, along with updated third-party libraries, which overall made application up to date. Along with that, Repository which separates database with backend was rewritten. Modules communicate faster and more secure; web components were modified and communication with external modules was rewritten using REST interface.

Overall, as the performance testing shows, the main operations done by applications increased in speed in about 25%.

BIBLIOGRAPHY

1. Dyson, J. (2015) Paperless Receipt Solution (PRS) System. ames Dyson Foundation. [online] Available at: http://www.jamesdysonaward.org/projects/paperless-receipt-solution-prs-system/
2. Electronic Transmission of Prescriptions. Business Services Authority (2015). [online] Available at: https://www.nhsbsa.nhs.uk/pharmacies-gp-practices-and-appliance-contractors/prescribing-and-dispensing/electronic-prescription-service-eps.
3. Tricks traders use to evade billions of francs in taxes. The New Times (2015). [online] Available at: http://www.newtimes.co.rw/section/article/2014-10-10/181782/.
4. Mehta, R. (2018) The Point of Sale Application System for Retail Industries all Over the World. [online] Available at: http://www.ibosocial.com/riteshmehta/pressrelease.aspx?prid=586672.
5. Planet Money Asks: What Small Thing Would You Do To Improve The World?. NPR (2017). Available at: https://www.npr.org/2015/07/22/425377064/planet-money-asks-what-small-thing-would-you-do-to-improve-the-world.
6. Find the Best mPOS System for Your Business. Market-Inspector (2020). [online] Available at: https://www.market-inspector.co.uk/epos-systems/pos/mpos.
7. Frazier, Bob (2013). Restaurant POS 3-Step Purchase Process. Newswire. [online] Available at: https://newswire.net/newsroom/pr/00077555-restaurant-pos.html.
8. Bert A., POSXML. – International Book Market Service Limited, 2012.
9. How hackers gave Subway a $3 million lesson in point-of-sale security. Ars Technica (2017). [online] Available at: https://arstechnica.com/information-technology/2011/12/how-hackers-gave-subway-a-30-million-lesson-in-point-of-sale-security/.
10. Seals, T. (2016). "Wendy's Point of Sale Hack Grows Bigger". Infosecurity. [online] Available at: https://www.infosecurity-magazine.com/news/wendys-point-of-sale-hack-grows/.

11. SAP point-of-sale systems were totally hackable with $25 kit. The Register (2017). [online] Available at: https://www.theregister.co.uk/2016/06/15/sap_patch_batch_fixes_3_yr_old_vuln/.
12. Bekmeshov, A., & Abyzov, B. (2018). Providing the stable functioning of payment terminals. Quality. Innovation. Education, (3), 60-62. [online] Available at: https://doi.org/10.31145/1999-513x-2018-3-60-62
13. Gutierrez, F. (2016). Pro spring boot. Apress.
14. Larsson, M. (2019). Hands-on Microservices with spring boot and spring cloud: Build and deploy Java microservices using spring cloud, Istio, and Kubernetes. – Packt Publishing.
15. Leonard, A. (2020). Spring boot persistence best practices: Optimize Java persistence performance in spring boot applications. Apress.
16. Point-of-sale system (POS). (n.d.). Encyclopedia of Production and Manufacturing Management, 524-524. [online] Available at: https://doi.org/10.1007/1-4020-0612-8_694
17. Rantos, K., & Markantonakis, K. (2013). Analysis of potential vulnerabilities in payment terminals. Secure Smart Embedded Devices, Platforms and Applications, 311-333. [online] Available at: https://doi.org/10.1007/978-1-4614-7915-4_13
18. Shieh, H., & Liao, Y. (2011). RFID restaurant POS system. 2011 International Conference on Machine Learning and Cybernetics. [online] Available at: https://doi.org/10.1109/icmlc.2011.6016799
19. PARK, John, Steve MACKAY a Edwin WRIGHT. Practical Data Communications for Instrumentation and Control. Amsterdam; London: Elsevier, 2003. ISBN 9780750657976.
20. MARTIN, Robert C. Čistý kód: [návrhové vzory, refaktorování, testování a další techniky agilního programování]. Brno: Computer Press, 2009. ISBN 978-80-251-2285-3.
21. RADU, C. Implementing Electronic Card Payment Systems. Artech House, 2002. ISBN: 978-1580533058
22. JOHNSON R. et al. Professional Java Development with the Spring Framework. John Wiley & Sons, 2007. ISBN: 9780764574832

23. WALLS, C. Spring in Action: Covers Spring 4. O'Reilly 2014. ISBN 978-1617291203

LIST OF ABBREVIATIONS

POS	Point of Sale
WEPOS	Windows Embedded for Point of Service
JPA	Java Persistent API
JVM	Java Virtual Machine
AJAX	Asynchronous JavaScript and XML
IDE	Integrated Development Environment
FIC	Fiscal Identification Code
API	Application Programming Interface
CPU	Central Processing Unit
PIN	Personal Identification Number
MFČR	Ministry of Finance of the Czech Republic

LIST OF FIGURES