

TALLINNA TEHNIKAÜLIKOOL  
Infotehnoloogia teaduskond  
Tarkvarateaduse instituut

Siim Rebane 050620IAPB

# PANGA E-ARVE SÜSTEEMI ARENDUS

Bakalaureusetöö

Juhendaja: Tanel Tammet, Ph.D.

Tallinn 2017

## **Autorideklaratsioon**

Kinnitan, et olen koostanud antud lõputöö iseseisvalt ning seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on töös viidatud.

Autor: Siim Rebane

22.05.2017

## **Annotatsioon**

Käesolevas bakalaureusetöös antakse ülevaade pankadele mõeldud e-arve süsteemi loomisest, tutvustatakse projekti tagamaid ja seadusandlikust poolest tulenevaid nõudeid ning selgitatakse tehnilisi kitsaskohti ja lahendusi. Projekt keskendub Eesti Pangaliidu poolt defineeritud e-arve standardile. Programmeerimiskeelena on kasutusel Java.

Töö tulemusena valmis toode, mis loob e-arvete saatmiseks vajaliku ühenduse müüja, panga ja kliendi vahel, vahendab e-arveid ning võimaldab sõlmida püsimakselepinguid. Antud toodet on võimalik, minimaalseid muudatusi tehes, integreerida erinevate pankade allsüsteemidesse nii Eestis kui ka väljaspool Eestit ning seda on praeguseks hetkeks juba kolme panga poolt tehtud.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 35 leheküljel, 5 peatükki, 4 joonist, 2 tabelit.

## **Abstract**

### **The Development of E-invoice System for Banks**

As of February 1<sup>st</sup>, 2014, all the countries residing in eurozone are subject to and must be accordance with Single Euro Payments Area (SEPA) rules for direct debit. Due to impractical (non-technical) demands on direct debit, banks in several countries, including Estonia, have opted for e-invoice and e-invoice standing order agreement systems.

The thesis at hand provides an overview of the creation of the e-invoice system for banks, briefly touches on reasons for creating the project, the legal requirements for the system, and explains the technical bottlenecks and applied solutions.

In chapter 2 the author describes the essence of e-invoice and e-invoice standing order agreement, how the service works and brings out the specifications for the e-invoice, provided by the Estonian Banking Association.

The next chapter contains the analysis of the application with in depth description of the process and system logic. Including its functional requirements, provided by the e-invoice standard and the host bank, and non-functional requirements, which are dictated solely by the host bank.

Chapter 4 is concentrated on the technical side of the project with describing the e-invoice system architecture, explaining decisions concerning software and hardware, and testing the final product.

The final product creates a necessary link between a seller, a bank and a client, mediates e-invoices and enables to sign e-invoice standing order agreements. With just few minor changes, this product can be integrated into different banks in Estonia and abroad.

The thesis is in Estonian and contains 35 pages of text, 5 chapters, 4 figures, 2 tables.

## Lühendite ja mõistete sõnastik

DI	<i>Dependency Injection</i> - sõltuvuste sisestuse raamistik.
EAA	Eesti Pangaliidu poolt defineeritud e-arve standardile vastav XML sõnum kliendilt pangale e-arve tellimiseks.
EAAV	Pangapoolne vastussõnum EAA sõnumile.
EARVE	Eesti Pangaliidu poolt defineeritud e-arve standardile vastav XML sõnum müüjalt pangale e-arve saatmine
E-arve	Elektrooniline arve, mis luuakse, edastatakse, kirjendatakse ja säilitatakse elektroonilises keskkonnas, st mille käitlemine toimub elektrooniliselt .
EJB	<i>Enterprise JavaBeans</i>
EL	Euroopa Liit
EMP	Euroopa Majanduspiirkond
SCNew	Eesti Pangaliidu poolt defineeritud e-arve standardile vastav XML sõnum müüjalt pangale müüjalepingu sõlmimiseks.
FTP	<i>File Transfer Protocol</i> - failiedastusprotokoll.
IBAN	<i>International Bank Account Number</i> – rahvusvaheline pangakonto number.
JMS	<i>The Java Message Service</i>
MVC	<i>Model–view–controller</i>
ORM	<i>Object-Relational Mapping</i>
PDF	<i>Portable Document Format</i>
SCAcc	Pangapoolne vastussõnum SCNew sõnumile
ServiceId	Teenuse tunnus - ostja teenuselepingu unikaalne tunnus müüja juures, mis identifitseerib maksjale esitatavate perioodiliste arvete voo.
SEPA	Euroopa Liidu ühtne euromaksete piirkond
SOAP	<i>Simple Object Access Protocol</i>
VEA	Pangapoolne vastussõnum EARVE sõnumile
XML	<i>Extensible Markup Language</i> - üldotstarbeline märgistuskeel.

## Sisukord

1 Sissejuhatus .....	10
2 E-arved.....	11
2.1 E-arve ja maksete lühikirjeldus .....	11
2.2 Euroopa ühtne euromaksete piirkond .....	12
2.3 E-arve teenuse kirjeldus.....	12
2.4 E-arve spetsifikatsioon .....	13
2.4.1 Müüjalepingu sõlmimine.....	13
2.4.2 E-arve tellimine .....	14
2.4.3 E-arve saatmine .....	14
2.5 E-arve kasutus ja alternatiivid .....	15
3 Rakenduse analüüs .....	16
3.1 Funktsionaalsed nõuded .....	16
3.2 E-arved.....	17
3.3 Püsimakselepingud .....	19
3.4 Mittefunktsionaalsed nõuded.....	19
3.5 Integratsioonid .....	20
3.5.1 Makse sooritamine.....	21
3.5.2 Kliendi andmete päring .....	21
3.5.3 Konto andmete päring .....	21
3.5.4 Kliendi id küsimine IBAN-i järgi.....	21
3.5.5 Panga info küsimine IBAN-i järgi.....	21
3.6 Põhiprotsessid.....	22
3.6.1 E-arve sõnumite vastuvõtmine .....	22
3.6.2 E-arve tasumine püsimakselepingute alusel .....	23
4 Süsteemi realisatsioon .....	26
4.1 Arhitektuur.....	26
4.1.1 E-arve integratsioonimoodul .....	26
4.1.2 Põhimoodul.....	27
4.1.3 XML töötlemise ja kontrollimoodul.....	27

4.1.4	Teenuste moodul .....	27
4.1.5	Kasutajaliidese moodul .....	27
4.2	Valitud tehnoloogiad .....	28
4.3	Funktsionaalsuse testimine .....	28
4.4	Koormustestimine.....	30
4.4.1	E-arve töötlemine .....	31
4.4.2	E-arve maksmine .....	32
4.4.3	Koormustestide järelused .....	33
5	Kokkuvõte .....	34
	Kasutatud kirjandus .....	35

## Jooniste loetelu

Joonis 1. E-arve staatused.....	18
Joonis 2. Sõnumite andmebaasi salvestamine .....	23
Joonis 3. E-arve maksmise protsess .....	24
Joonis 4. Makse vastuse töötlemine .....	25



## **Tabelite loetelu**

Tabel 1. E-arve töötlemise koormustesti tulemused.....	32
Tabel 2. E-arve maksmise koormustesti tulemused .....	32

## 1 Sissejuhatus

Paljudes Euroopa riikides on pangad võtnud kasutusele e-arved ja püsimakselepingud. Selle käigus on nad pidanud asendama oma olemasolevad otsekorraldussüsteemid e-arve ja püsimakselepingu süsteemidega.

Lõputöö eesmärgiks on arendada pangale allsüsteem, mis pakuks e-arve ja püsimakseteenuseid. Süsteem peab vastama üldisele e-arve spetsifikatsioonile. Samas, lisanõudena, on vaja loodavat lahendust kasutada eri riikides ja eri pankades, mis omakorda tähendab, et arenduses tuleb silmas pidada võimalust sama süsteemi erinevates pankades kasutusele võtta, tehes koodis võimalikult vähe muudatusi.

Näitamaks, et süsteem töötab, tuleb rakendus katta nii automaatsetestidega, mis kontrollivad e-arve standardile vastavust, kui ka koormustestidega, et olla kindel süsteemi võimekuses tegeleda korraga suure hulga e-arvetega.

Lõputöös kirjeldatud rakendus on loodud ettevõttes AS Helmes, kus sellega tegeles (erinevatel ajahetkedel) kokku kolm inimest. Autor osales projektis kui juhtivarendaja/arhitekt ning tema vastutusala oli arhitektuuri koostamine, tarkvara arendus ja projekti tehniline juhtimine. Kirjeldatud projektis on üle poole lähtekoodist ja suur enamus põhimoodulist autori poolt kirjutatud.

## 2 E-arved

### 2.1 E-arve ja maksete lühikirjeldus

Pankade kaudu saadetakse e-arved on spetsifitseeritud pangaliidu poolt. Tegu on *Extensible Markup Language* (XML) formaadis dokumentidega, mis võimaldavad neid infosüsteemide poolt töödelda, erinevalt näiteks *Portable Document Format* (PDF) dokumentidest, kus arvuti poolt täpse info välja lugemine on raskendatud.

E-arve sisaldab kõiki klassikalise arve kohustuslikke välju ja selle baasilt on pankadel võimalik genereerida automaatselt maksekorraldusi. Tänu sellele ongi pankades kasutusel e-arve püsivaks teenus, mis võimaldab kliendil sõlmida lepingu oma arvet automaatselt tasumiseks.

Lisaks e-arve XML formaadi kirjeldusele sätestab e-arve standard ka kõiki tehnilisi ja ärioloogilisi kontrole, mida tuleb rakendada enne e-arve vastu võtmist. Näiteks tuleb kontrollida, et sisestatud viitenumber on korrektne, arve summa on positiivne, kliendi konto number on korrektne ja arve maksekuupäev on tulevikus.

E-arve standardis on defineeritud ka kaks lisasõnumit, müüjalepingu loomise ja e-arve tellimise jaoks. Mõlemad sõnumitüübid on vajalikud selleks, et kõigi osapoolteni jõuaks tehniline info, mis muudab e-arve saatmise teostatavaks.

E-arved on tavakasutajale väga sarnased varem eksisteerinud otsekorraldustega, kuid tegelikkuses pakuvad need kliendile palju rohkem võimalusi. Selline süsteem mitte ainult ei võimalda teha automaatseid makseid, vaid vabastab ka müüja vajadusest saata välja paberarveid.

Kuna e-arveid on IT-süsteemides mugav kasutada, on mitmete Euroopa riikide riiklike asutuste jaoks tehtud e-arvete kasutamine suuremal või vähemal määral kohustuslikuks. Eestis peavad alates 1. märtsist 2017 kõik avaliku sektori asutused olema võimelised vastu võtma e-arveid [1].

## 2.2 Euroopa ühtne euromaksete piirkond

*Single Euro Payments Area* (SEPA) eesmärgiks on ühtlustada üleeuroopalised maksed, et inimesed saaksid sooritada ka teistesse Euroopa riikidesse (eurodes teostatavaid) makseid sama lihtsalt kui oma kodumaal. Seda regulatsiooni peavad täitma kõik Euroopa liidu (EL) ja Euroopa Majanduspiirkonna (EMP) liikmed, mitte ainult eurosooni liikmed [2].

Eestis andis suure tõuke e-arvete kasutuselevõtuks SEPA regulatsiooni otsekorralduste osa. Enne seda oli e-arve vähe kasutatud instrument, kuid kuna SEPA otsekorralduste osa ei olnud ühilduv seni Eestis toimunud regulatsiooniga, siis Eesti pangad lõpetasidki otsekorraldus teenuse pakkumise 1 veebruarist 2014 [3].

Üle Euroopa oli eurot kasutavatele riikidele viimaseks otsekorralduste SEPA standarditele vastavusse viimise tähtajaks 1. veebruar 2014. Riigid, kes ei olnud sel ajal eurosooni liikmed, pidid oma otsekorraldused viima vastavusse hiljemalt 31. oktoobriks 2016, kuid mitte hiljem, kui üks aasta pärast euro kasutuselevõttu. Näiteks Leedus oli selleks kuupäevaks 1. jaanuar 2016 [4].

## 2.3 E-arve teenuse kirjeldus

E-arve teenuse pakkumine algab müüja ja panga vahel loodud arvete edastamise lepingust ja lõpeb kasutaja poolt e-arve saamisest ja püsimakselepingu sõlmimisest. Tavaliselt näeb kogu protsess välja järgnev:

- pangas või operaatori juures sõlmitakse müüjaga leping;
- kui leping sõlmitakse läbi operaatori saadab viimane müüja poolsed lepingud edasi kõikidesse pankadesse, mille pangad peavad aktsepteerima;
- klient saab tellida pangast endale vastava müüja e-arve, mille tulemusena saadetakse e-arve avaldus edasi müüjale;
- müüja saab e-arve avalduse, mis võimaldab tal saata e-arveid vastavale kontole;
- kui klient on saanud e-arve saab ta sõlmida püsimakselepingu antud arvete automaatseks tasumiseks.

E-arve avaldust saates ja e-arveid vastu võttes on siduvaks väljaks *ServiceId*, mis tähendab kliendi tunnust müüja juures.

## **2.4 E-arve spetsifikatsioon**

E-arved on spetsifitseeritud vastava riigi pangaliidu poolt. Eesti täpset e-arve standardit saab lugeda Eesti Pangaliidu lehelt [5]. Lisaks e-arvetele pakuvad pangad püsimakse teenust, mis lubab kliendil määrata arved, mis lähevad automaatselt maksmisele.

Üldiselt on Euroopas e-arve sõnumite formaadid väga sarnased. Selles töös keskendutakse peamiselt Eesti e-arve standardile, mis defineerib andmevahetuseks kolm XML sõnumit: 1) müüjalepingu sõlmimine (SCNew), 2) e-arve tellimine (EAA) ja 3) e-arve saatmine (EARVE), koos vastussõnumitega (SCAcc, EAAV, VEA).

### **2.4.1 Müüjalepingu sõlmimine**

Operaator teavitab panka uue müüja lisandumisest kasutades SCNew sõnumit, millega edastatakse järgnev oluline informatsioon:

- ettevõtte nimi;
- ettevõtte registrikood;
- müüja lepingu id;
- kas piiratud kuvaga arvete esitamine on lubatud;
- kas pangast on saab tellida e-arveid;
- ServiceId nimetus ja valideerimise reeglid;
- makseperiood;
- operaatori tunnus;
- kasutatud kujundus.

Pärast seda peab sõnumi saanud pank vastama sõnumile aktsepteeriva sõnumiga (SCAcc). Vastussõnumi saatmise järgselt on müüjal võimalik läbi operaatori saata e-arveid ja e-arve avaldusi (kui ülalkirjeldatud lepingus on see lubatud).

#### **2.4.2 E-arve tellimine**

Juhul kui vastav müüja on pangas registreeritud ja lepingu raames on lubatud kliendil tellida e-arveid läbi panga, saab interneti pangast tellida vastava müüja e-arveid. Selleks kasutatakse EAA XML sõnumit, mis edastatakse müüjale.

E-arve avaldus on kliendilt müüjale saadetud sooviavaldus antud e-arve saamiseks kliendi poolt määratud pangakontole. Oluline on tähele panna, et läbi panga saadetav e-arve avaldus on vabatahtlik ja sellise avalduse võib klient teha ka otse müüja juures. See tähendab, et infosüsteemi arendades puuduvad meil otsesed seosed e-arve avalduse ja e-arve vahel.

E-arve avaldusega saadetakse müüjale:

- kliendi andmed;
- *serviceId*;
- avalduse tüüp (kas avaldusega soovitakse tellida või tellimust lõpetada);
- *International Bank Account Number* (IBAN) kuhu e-arve on tellitud;
- e-arve esitus tüüp (täis või piiratud);
- avalduse loomise aeg;
- müüja andmed.

Müüja peab vastama antud sõnumile EAAV XML sõnumiga, kus ta teavitab pank, et ta on e-arve avalduse kätte saanud.

#### **2.4.3 E-arve saatmine**

EARVE sõnumiga on võimalik müüjal saata pank, kuvamiseks vastav e-arve. Selle õnnestumiseks on vaja, et pank oleks aktsepteerinud vastava müüja oma süsteemis.

Kuna e-arvega antakse edasi kogu arve sisu, siis on see suhteliselt mahukas dokument. Nendest kirjeldatava infosüsteemi jaoks olulisemad väljad on:

- päises olev info (kellelt/kellele);
- viide müüjalepingule;
- saaja konto;
- arve tähtaeg;
- arve summa;
- *ServiceId*.

## **2.5 E-arve kasutus ja alternatiivid**

E-arve on Euroopas palju kasutust leidnud lahendus, eriti pärast SEPA direktiivi kehtima hakkamist. Mingil kujul e-arvet kasutavad näiteks kõik Baltimaad, Soome, Rootsi, Saksamaa ja paljud teised.

Peamise alternatiivina e-arvetele eksisteerivad otsekorralduslepingud (mida kasutas ka Eesti enne e-arvetele üleminekut). Need on kolmepoolsed lepingud panga, ostja ja müüja vahel, kus müüjal lubatakse ise algatada ostja kontolt raha kandmine mingisuguse teenuse eest müüja kontole.

### 3 Rakenduse analüüs

Vastavalt varem kirjeldatud e-arve tööpõhimõtetele, peab panga allsüsteem olema võimeline vastu võtma ja välja saatma kõiki e-arve standardis defineeritud sõnumeid. Samuti peab eksisteerima võimalus püsimakseteenuse pakkumiseks.

#### 3.1 Funktsionaalsed nõuded

E-arve moodulile esitatavad funktsionaalsed nõuded tulenevad peamiselt e-arve standardist.

E-arve süsteem peab olema võimeline saatma ja vastu võtma kõiki e-arve standardis defineeritud sõnumeid ja tegema sealt tulnud andmestiku teistele süsteemidele kättesaadavaks.

Süsteemisiseselt peab olema võimalik luua püsimakselepinguid ja eksisteerima taustaprotsess, mis nende alusel makseid automaatselt sooritab. Selle nõude puhul on olulisim küsimus, kuidas siduda omavahel e-arved ja püsimakselepingud, et oleks tagatud võimalikult kiire õigete e-arve ja püsimakselepingu paaride leidmine.

Enamus ülejäänud nõuetest tulenevad e-arve standardi detailidest, kus on kirjeldatud kõik e-arvete valideerimise reeglid ja esitamise nõuded. Mõned olulisemad punktid millega arvestada:

- kliendil peab olema võimalik takistada ühe spetsiifilise arve automaatselt maksmisele minekut, katkestamata kogu püsimakse lepingut;
- eksisteerivad kreditarved, mille alusel makset ei genereerita;
- arveid, mis ei ole edukalt läbinud ärioloogilisi kontrole, ei tohi kliendile kuvada.

Standardis mitte kirjeldatud, kuid pangalt tulnud olulisemad soovid olid:

- Osalise makse võimalus - kui kliendil ei ole kontol piisavalt raha, siis võimaldaks see arve maksta vähemalt osaliselt, eeldusel, et kliendi kontol on vähemalt 50% nõutavast summast.



- Maksete mitmekordne proovimine – juhul kui mingil põhjusel arve maksmine ebaõnnestub (näiteks kontol pole piisavalt raha). Sellisel juhul tuleks arvet proovida korduvalt maksta enne, kui arve maksmine loetakse lõplikult ebaõnnestunuks.
- Kõigest eelnevast järeldub, et väga oluline on e-arve staatuste haldamise meetod, kindlustamaks, et süsteem teab millised arved on mõeldud maksmiseks ja milliste reeglite alusel.

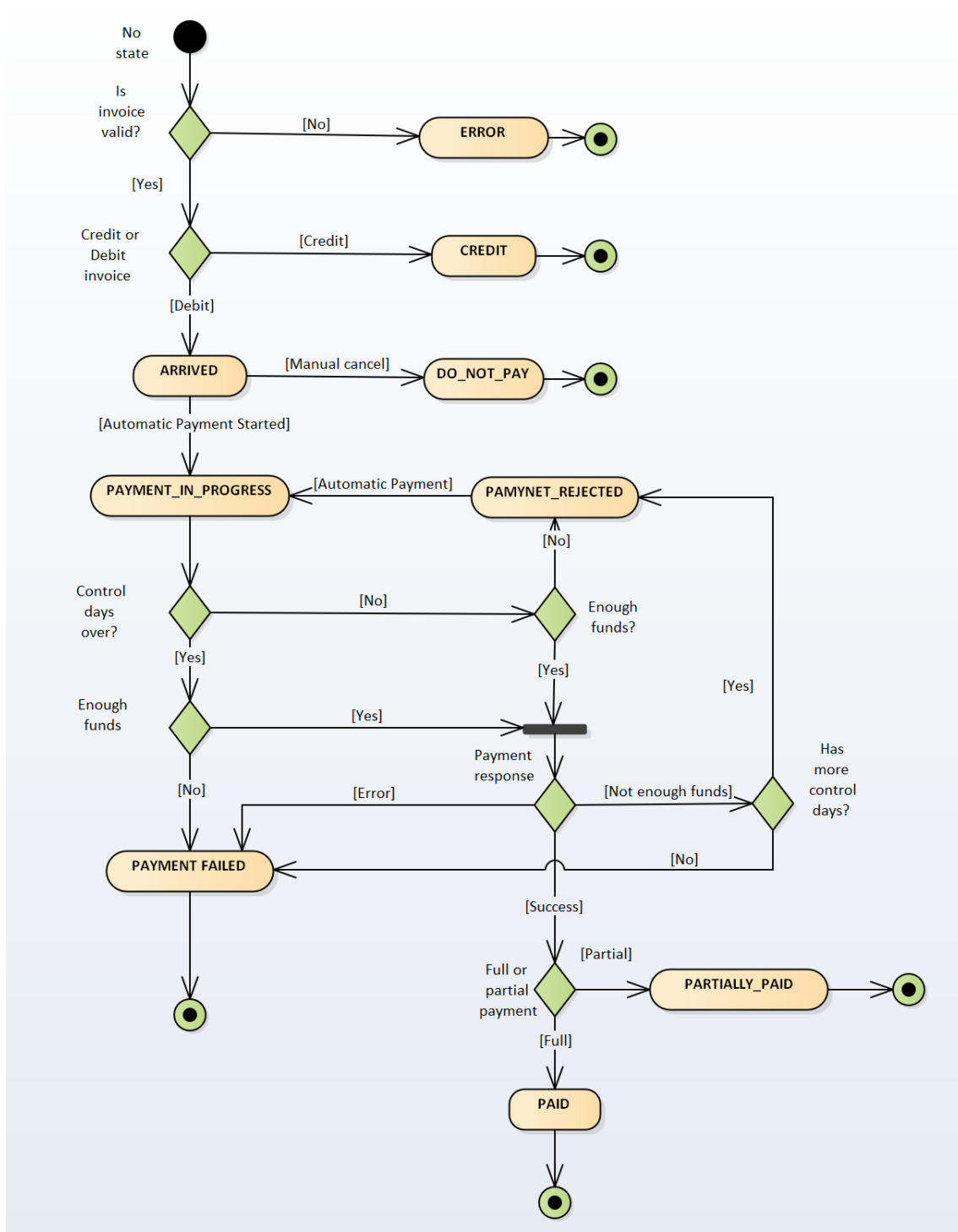
### 3.2 E-arved

E-arved sisaldavad kogu infot, mis on vastu võetud e-arve XML sõnumist. Väga hästi tuleb läbi mõelda süsteemis olevad erinevad e-arve staatused.

E-arve staatused:

- **ERROR** – e-arve on vigane ja seda kliendile välja ei kuvata;
- **CREDIT** – tegemist on krediid e-arvega, mida ei ole võimalik maksta;
- **ARRIVED** – uus sisse tulnud e-arve;
- **DO\_NOT\_PAY** – e-arve, mis on märgitud automaatselt maksmisele mitte kuuluvaks. Ükski püsimakse leping ei tohi selles staatuses e-arvet automaatselt maksta;
- **PAYMENT\_IN\_PROGRESS** – e-arve vahestaatus, mis näitab, et makse sooritamine on töös;
- **PAYMENT\_REJECTED** – selle e-arve kohta on üritatud teha makset, kuid viimane ebaõnnestus. Samas on ette nähtud, et seda e-arvet peaks proovima automaatselt hiljem uuesti maksta;
- **PAYMENT\_FAILED** – kõik automaatsed e-arve maksed on ebaõnnestunud ja rohkem ei proovita seda e-arvet automaatselt maksta;
- **PARTIALLY\_PAID** – e-arve on osaliselt makstud;

- PAID – e-arve on täielikult makstud.



Joonis 1. E-arve staatused

### 3.3 Püsimakselepingud

E-arve moodul peab suutma automaatselt maksta e-arveid, mis vastavad lepingus toodud tingimustele. Selleks on vajalik, et infosüsteem leiab vastavad e-arved, itereerib üle nende ja üritab sooritada igat makset.

Püsimakselepingus peab olema võimalik defineerida, millal automaatne makse toimub. Selle tarbeks on hetkel rakendatud kolm erinevat lahendust:

- 1) x päeva pärast arve saamist, kus x tuleneb üldiselt riiklikust seadusandlusest;
- 2) arve tähtajal;
- 3) kasutaja poolt määratud kuupäev.

Hoolimata valitud meetodist, eeldame edaspidistes arvestustest, et eksisteerib mingisugune püsimakselepingu maksekuupäev, mis on esimene kuupäev, millal vastavat makset üritatakse sooritada.

Makse sooritamine võib teatud juhtudel ebaõnnestuda. Üheks peamiseks põhjuseks on ebapiisavad rahalised vahendid kliendi kontol. Seetõttu on võimaldatud igale püsimakselepingule eraldi määrata kontrollpäevad. See tähendab, mitu päeva pärast makse kuupäeva üritatakse teha makset, enne kui e-arve maksmise protsess loetakse ebaõnnestunuks. Üldiselt ei ole see kliendi poolt vabalt valitud suurus vaid mingisugune vaikeväärtus, mis on määratud internetipangas.

Lisaks on makset sooritades oluline, et klient ei üritaks e-arvet maksta samal ajal, kui automaatne e-arvete maksmise protsess seda teha üritab. Selle tarbeks on e-arvel staatus *PAYMENT\_INPROGRESS*. Osapool, kes üritab teha makset, peab alustuseks määrama vastava arve sellesse staatusesse. Juhul, kui e-arve staatuse muutmine ebaõnnestub, ei tohi makset sooritada.

### 3.4 Mittefunktsionaalsed nõuded

Lisaks e-arvete süsteemi funktsionaalsetele nõuetele on väga olulised ka mittefunktsionaalsed nõuded nagu jõudlus ja käideldavus. Mittefunktsionaalsed nõuded on pangaspetsiifilised, mitte ühtsed ja neid tuleb arenduse käigus arvesse võtta, et panga vajadused oleks kaetud.

Selle projekti raames arvesse võetud kõige karmimad mittefunktsionaalsed nõuded on järgnevad:

- kuni 500 000 arvet kuus;
- kuni 500 000 arve baasil tehtud makset kuus;
- kuni 250 000 arvet päevas;
- kuni 250 000 arve baasil tehtud makset päevas;
- kõik arved tuleb sisse lugeda ühe päeva jooksul;
- kõik maksed tuleb sooritada ettenähtud kuupäeval;
- probleemide tekkimisel on lubatud kuni neljatunnine süsteemi maas olek.

Jõudluses tuleb arvestada sellega, et süsteem peab toime tulema 500 000 sisse tulnud arvega kuus, milledest enamuse pealt tuleb genereerida maksekorraldus. Samas, kuni pool sellest koormusest võib langeda ühele päevale. Makset ei tohi sooritada samal päeval, kui e-arve on pankas jõudnud. Sellest tulenevalt saame me vaadata nende kahe protsessi, e-arve töötlemine ja püsimakse sooritamine, koormust eraldi.

Kuna me eeldame, et ühele päevale langeb kuni pool kogu kuu koormusest, siis arvutusi tehes on oluline maksimaalne päevane maht ja kuu koormus ei ole reaalsuses oluline. Arvestada tuleb ka lubatud neljatunnise süsteemi tõrkega, millest hoolimata peavad päevased protsessid saama oma töö lõpetada.

### **3.5 Integratsioonid**

Arendatud rakenduse puhul on tegemist osaga pangasisesest integratsiooniprojektist, mis vajab mitmeid teenuseid panga allsüsteemidest.

Järgnevalt on loetletud kõik teenused, mida antud projektis vaja on. See nimekiri on oluline, kuna igas pangas tuleb need teenused uuesti arendada, sest pankade allsüsteemid on erinevad.

Vajalike teenuste nimekirja on üritatud hoida võimalikult minimaalsena:

### **3.5.1 Makse sooritamine**

- Sisend: ISO 20022 [6] formaadile vastav XML sõnum, millega algatatakse maksekorraldus
- Väljund: ISO 20022 formaadile vastav vastussõnum, mis uuendab makse staatust.

Tegu on asünkroonse teenusega, mis võib saada vastuseid palju hiljem, kui need on teele saadetud.

### **3.5.2 Kliendi andmete päring**

- Sisend: Kasutaja id
- Väljund: Kasutaja andmeid

### **3.5.3 Konto andmete päring**

- Sisend: IBAN konto number
- Väljund: Konto staatus (aktiivne/blokeeritud) ja konto jääk

### **3.5.4 Kliendi id küsimine IBAN-i järgi**

- Sisend: IBAN
- Väljund: Konto number

### **3.5.5 Panga info küsimine IBAN-i järgi**

- Sisend: IBAN
- Väljund: Panga info

## 3.6 Põhiprotsessid

Rakendusel eksisteerib kaks põhiprotsessi, milledeks on e-arve vastuvõtmine ja e-arve maksmine.

### 3.6.1 E-arve sõnumite vastuvõtmine

Arved jõuavad pankka erinevate kanalite kaudu. Seni kasutatud kanalid on:

- *File Transfer Protocol (FTP)*;
- *Simple Object Access Protocol (SOAP)* veebiteenus;
- *Java Message Service (JMS)* teenus.

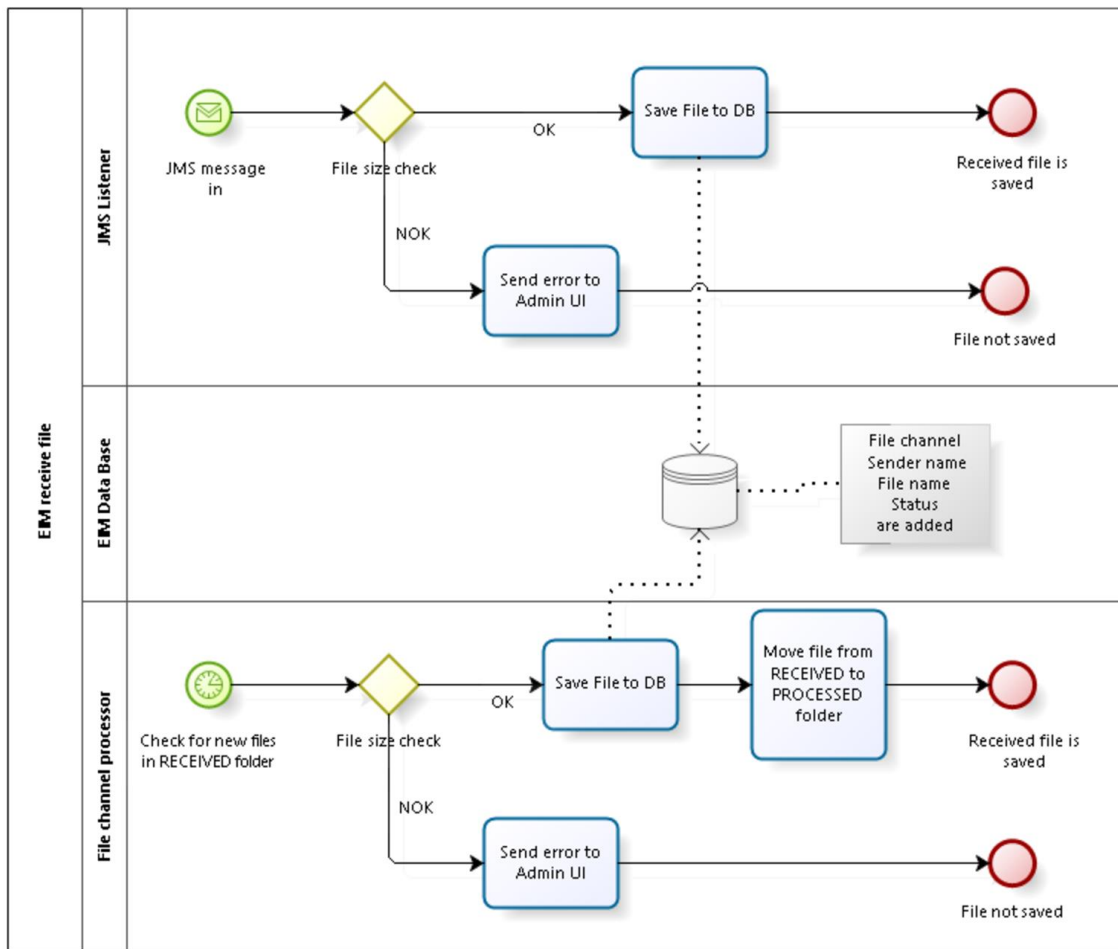
Nii suure hulga failide kiireks, kuid võimalikult veakindlaks töötlemiseks, on antud failide laadimine jaotatud kaheks eraldi sammuks:

- 1) Faili andmebaasi salvestamine (Joonis 2).

Igale sisestuskanalile on kirjutatud oma komponent, mis peab olema võimeline saatma ja vastu võtma faile vastavas kanalis. Vastuvõetud failidele tehakse ainult suuruse kontroll, et see ei ületaks ettenähtud mahupiirangut (näiteks < 100 MB). Juhul kui fail on ettenähtud suuruses, kirjutatakse see andmebaasi, tegemata ühtegi kontrolli faili sisu üle. Kui sissetulev fail ületab lubatud mahtu, kuvatakse viga administraatori kasutajaliideses.

- 2) Faili töötlemine andmebaasist.

Selles faasis ei ole enam määrav, millist kanalit pidi fail süsteemi sisenes ja edaspidi töötleb kõiki faili üks protsess. Edaspidi on antud protsessi ülesandeks saada aru, mis tüüpi failiga on tegu, teha vastavad kontrollid ja salvestada andmed andmebaasi. Vajadusel tuleb saata ka vastusfail.



Joonis 2. Sõnumite andmebaasi salvestamine

### 3.6.2 E-arve tasumine püsimakselepingute alusel

E-arve maksmise protsess peab iga päev maksma kõik arved, millede püsimakselepingus määratud maksetähtaeg on kätte jõudnud.

Selleks, et arve leidmine oleks kiire, hoitakse rakenduses kogu aeg viidet e-arvelt püsimakselepingule, mille alusel peaks toimuma makse. Makseprotsess ei pea tegelema õige püsimakselepingu leidmisega.

Makse protsess on jaotatud kaheks osaks:

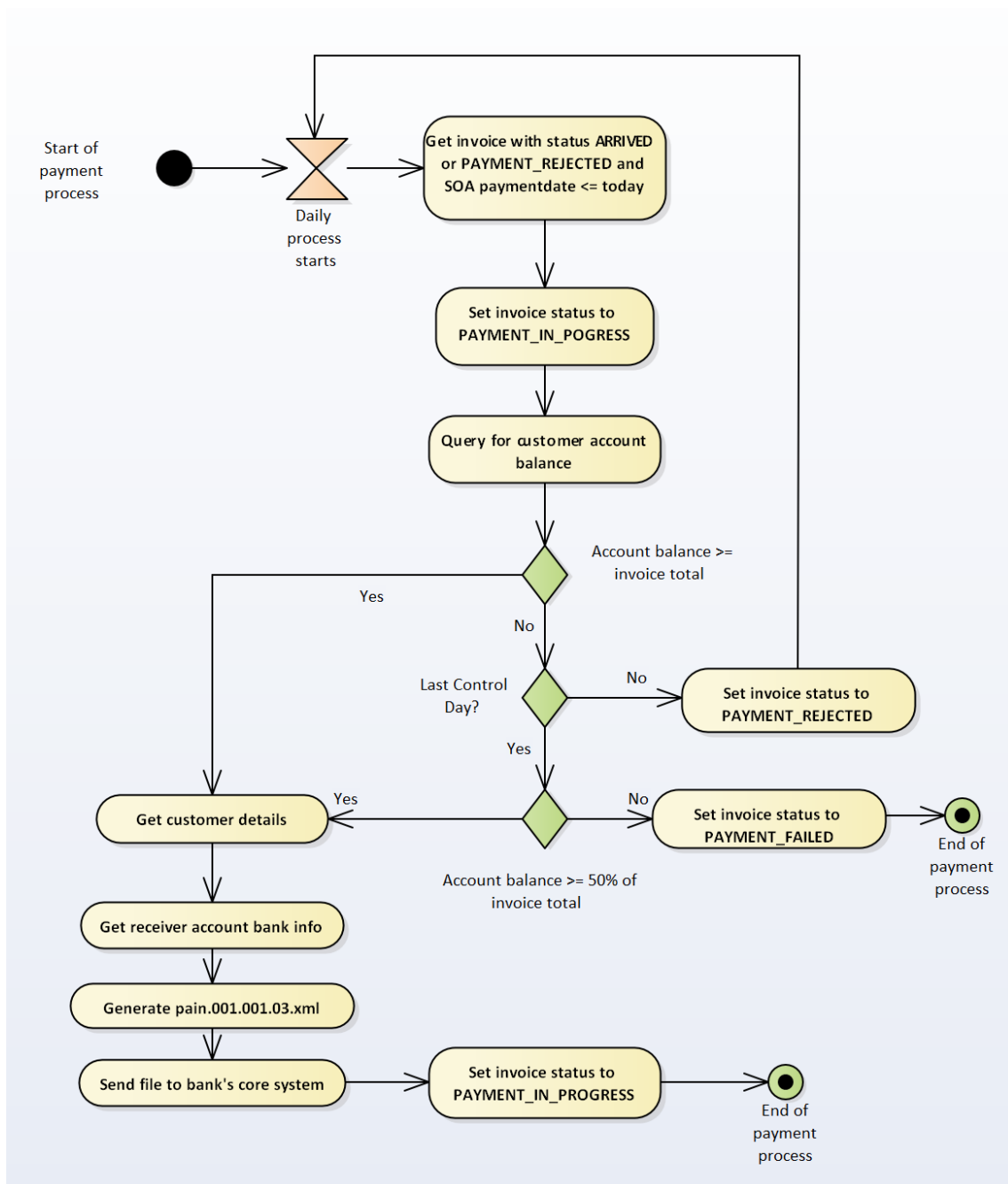
1) Põhiprotsessiks (Joonis 3), mis:

- leiab maksmist vajavad arved;
- kontrollib, kas kasutajal on piisavalt vabu vahendeid;

- genereerib ISO formaadis maksekorralduse;
- saadab maksekorralduse panga allüsteemile täitmiseks.

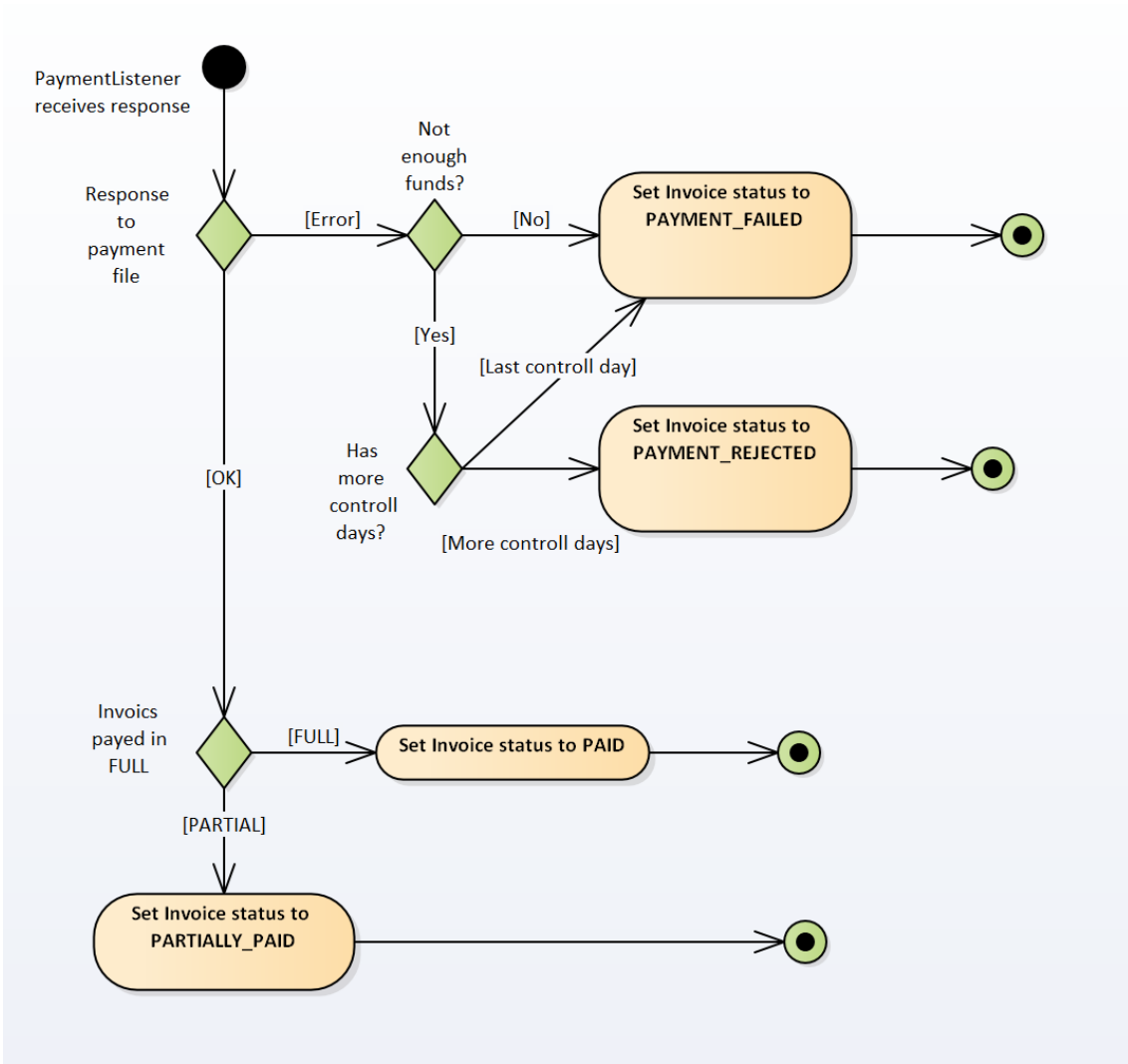
1) Makse vastussõnumi vastu võtmiseks (Joonis 4), mis :

- kontrollib, kas makse oli edukas;
- määrab e-arve õigesse staatusesse.



Joonis 3. E-arve maksmise protsess





Joonis 4. Makse vastuse töötlemine

## 4 Süsteemi realisatsioon

Antud infosüsteemis on väga palju erinevaid integratsiooni punkte. Süsteem peab olema võimeline:

- vahetama e-arve standardi alusel sõnumeid teiste pankade ja müüjatega;
- kasutama teenuseid teistest panga allsüsteemidest;
- pakkuma teenuseid internetipangale.

Kõik pangasisesed ühendused kasutavad JMS protokollid, kuid pangast välja võivad protokollid erineda.

### 4.1 Arhitektuur

Tegu on infosüsteemiga, millele ei ole võimalik internetist selle poole teha ühtegi päringut. Soovitavalt ei lubata ka süsteemil endal teha päringuid avalikku interneti.

Rakendus on kokku kompileeritud ühe tükiks, kuid sisemiselt on see jaotatud erinevateks mooduliteks, et lihtsustada muudatuste teostamist erinevate pankade juures.

#### 4.1.1 E-arve integratsioonimoodul

Selle mooduli eesmärk on luua ühendus mõne teise süsteemiga e-arve standardis defineeritud failide vahetamiseks. Praeguseks on realiseeritud kolm erinevat lahendust (Fail, JMS, SOAP). See moodul on üldiselt pangaspetsiifiline, kuna faili vahetuseks kasutatakse erinevaid protokolle.

1) Fail – Rakendusele on failisüsteemis defineeritud sisend- ja väljundkaustad. Infosüsteem kontrollib pidevalt, kas sisendkataloogi on tulnud uusi faile. Uue faili leidmisel, kirjutatakse see andmebaasi. Juhul, kui on vaja fail välja saata, siis see lihtsalt kirjutatakse vastavasse kataloogi.

Kirjeldatud moodul ei ole vastutav selle eest, kuidas faile reaalselt erinevate osapoolte vahel liigutatakse.

2) JMS – Tegu on tavalise JMS teenusega, millega saab infosüsteemile saata faili nime ja sisu. Faili töötlemise järgselt saadetakse üle JMS-i vastusfail.

3) SOAP veebiteenused – See realisatsioon võimaldab algatada SOAP päringu teise infosüsteemi, millega käiakse regulaarset küsimas uusi faile ja saadetakse vastusfailid.

#### **4.1.2 Põhimoodul**

Põhimoodulis on realiseeritud rakenduse põhiprotsessid (e-arve töötlemine ja püsimate sooritamine). Põhimoodul kasutab oma töös teisi mooduleid. Näiteks XML töötlemise moodulit (mis on erinev sõltuvalt riigist).

Põhimoodulis on realiseeritud ka ühenduse kiht kogu andmebaasiga. Juhul kui mõnel teisel moodulil on vaja sooritada andmebaasi päringuid, siis kasutab see selleks siin realiseeritud funktsionaalsust.

#### **4.1.3 XML töötlemise ja kontrollimoodul**

XML töötlemise mooduli eesmärk on sisse lugeda e-arve standardile vastav XML ja luua sellest kindlas struktuuris Java objekt, mida põhimoodul saaks kasutada. See moodul on eraldiseisev, sest antud rakendust kasutatakse eri riikides ja XML standard, defineeritud vastava riigi pangaliidu poolt, on erinev.

Lisaks XML-i töötlemisele peab see moodul tegema ka kõik ärioloogilised kontrollid ja vajadusel tagastama ka e-arve standardi järgse veakoodi.

#### **4.1.4 Teenuste moodul**

Teenuste moodul pakub panga teistele allsüsteemidele JMS teenuseid, millede kaudu need saavad küsida e-arvetega seotud informatsiooni. Näiteks, interneti pank peab tegema päringu e-arvete infosüsteemi selleks, et kliendile kuvada tema püsimatelepinguid.

#### **4.1.5 Kasutajaliidese moodul**

Tegu on ainult panga sisevõrgus kättesaadava veebipõhise kasutajaliidese. Selles saab vaadata rakenduse esialgset logi, hallata ühendusi teiste pankadega ja seadistada rakenduse parameetreid. Antud moodul aitab lihtsustada infosüsteemi haldamist, kuid põhiprotsessid töötavad korrektselt ka ilma selle moodulita.

## 4.2 Valitud tehnoloogiad

Rakendus on arendatud Java 7 baasil. Suhtluskihina on kasutatud JMS protokoll. JMS on kasutuses paljudes pankades, kuna see tagab hea töökindluse ja koormustaluvuse.

Andmebaasi valik langes Oracle kasuks, tänu viimase töökindlusele ning pankade suurele kogemusele antud andmebaasi kasutamisel. Andmebaasi päringute sooritamiseks kasutatakse Spring JdbcTemplate ja mitte ORM raamistike, seda päringute optimeerimise eesmärgil.

Sõltuvuste sisestuse (DI - Dependency Injection) raamistikuna on kasutatud Spring-i, mis on kõige laiemalt levinud DI raamistik. Ainsaks reaalseks alternatiiviks on Enterprise JavaBeans (EJB) raamistik, mis on aga sõltuv rakendusserverist. Kuna enamuses servereid mis toetava EJB-d on tasulised, loobuti sellest variandist. Lisaks lukustaks EJB kasutamine rakenduse kindlasse rakendusserverisse, mida autor üritas arenduse käigud vältida. Rakendusserverina on kasutusel avatud lähtekoodiga Apache Tomcat Server, kuid tehniliselt ei ole rakendusserveri valik piiratud.

E-arve infosüsteemi kasutajaliidese moodul on mõeldud ainult administreerimiseks ja seda kasutab vähe inimesi, seetõttu ei ole arendades sellele suurt rõhku pandud. Tegemist on andmebaasi tabelitele lisatud lihtsa graafilise liidesega, mille kiireks arendamiseks on valitud Spring Model-View-Controller (MVC) raamistik ning välditud on keerukamaid javascripti põhiseid raamistikke nagu Angular ja React JS.

## 4.3 Funktsionaalsuse testimine

E-arve süsteemide funktsionaalsuse testimine on jagatud kaheks osaks, ühiktestideks ja vastuvõtutestideks.

Esiteks on arvete ja maksmiste protsess kaetud ühiktestidega, mis peaks tagama, et kõik rakenduse komponendid oleksid töökorras. Tegu on regressiooni testidega, mis tagavad, et tehes koodi muudatusi ei tehtaks kogemata katki juba varem kirjutatud koodi.

Kõik ühiktestid on kirjutatud JUnit raamistikus ja need käivitatakse iga kord Apache Maven programmiga rakendust kokku kompileerides.

E-arve süsteem on rakendus, mille puhul on oluline (teenuste omavaheline) integratsioon. Nii e-arve töötlemine kui makse sooritamine kasutavad mitmeid panga allsüsteeme ja seetõttu on vajalik testida kogu ahelat, e-arve sisestamisest kuni reaalse makse sooritamiseni. Kogu rakenduse töötamist täielikult automatiseeritud testidega kontrollida ei saa, vaid põhimõtteliselt "mängitakse" kogu protsess läbi.

Testida tuleb kahte protsessi:

- 1) e-arve töötlemine;
- 2) püsimakse genereerimine.

Eelloetletud testide sooritamine on ajamahukas ja võtab minimaalselt ühe päeva, kuna sisse tulnud e-arve maksetähtaeg ei tohi olla tänane kuupäev (ehk testi alustamise kuupäev).

Sellise vastuvõtutesti sooritamiseks on vajalik, et testi andmestik oleks kooskõlas allsüsteemides olevate andmetega. Oluline on, et:

- vastava kliendi info on allsüsteemis olemas;
- panga allsüsteemis eksisteerib konto millele esitatakse arve;
- makse teenus on võimeline vastu võtma makset e-arves ettenähtud pangakontole;
- vastaval kontol on piisavalt vabu vahendeid;
- konto ei ole blokeeritud.

Eelnevast tulenevalt ei ole kogu tsükli võimalik täisautomaatselt testida, kuna arendajal puudub juurdepääs panga allsüsteemidele. Kontrollide sooritamise lihtsustamiseks on loodud programm, mis genereerib erinevaid e-arveid pangaga eelnevalt kokku lepitud testimiseks mõeldud konto jaoks.

E-arvete testimise juures on oluline kontrollida:

- e-arve töötlemise positiivset stsenaariumit;

- e-arve töötlemise kõiki defineeritud valideerimise reegleid (näiteks esitatakse arve, mille makse tähtaeg on eilne);
- kas makse sooritamine õnnestus.

Kaheastmeline kontrollsüsteem võimaldab tagada, et iga element iseseisvalt kui ka kogu süsteem on töökorras. Ühiktestidega saab vältida lisaarenduste juures võimalike vigade teket. Vastuvõtutestid tagavad nii e-arve süsteemi ja kõikide allsüsteemide kui ka nendevaheliste integratsioonide korrektse töötamise. Viimane on oluline veendumaks, et kogu ahel on töökorras.

#### **4.4 Koormustestimine**

Kuna rakendus kasutab suhtluseks asünkroonseid JMS-i päringuid, millel ei ole staatust, siis on võimalik e-arve süsteem panna tööle paralleelselt mitmele masinale. Koormustestide põhieesmärgiks on hinnata:

- ligikaudselt, kui suurt koormust süsteem kannatab;
- süsteemis esineda võivaid pudelikaelu;
- planeeritava riistvara sobivust.

Süsteemi töö kiirust mõjutavad kolm olulist faktorit: 1) andmebaasi päringud, 2) rakendusserveri töö, 3) integratsioonid allsüsteemidega.

Kuna kirjeldatud infosüsteemi on kasutatud mitmes pangas, tuleb aru saada, kas koormustaluvus võib kuidagi erineda. Kõikides pankades on andmebaasina kasutusel Oracle ja päringud on samasugused. Rakendusserverina on kasutusel Apache Tomcat ja rakenduse põhimoodul on sama. See tähendab, et nende tegurite puhul ainuke erinevus saab olla riistvaras.

Pangati on erinevaks faktoriks integratsioonid allsüsteemidega. Koormusteste kirjutades tuleb arvestada, et ei ole teada milline on kasutatud teenuste koormustaluvus, mis isegi pangasiseselt võib, sõltuvalt allsüsteemide koormusest, märkimisväärselt muutuda.

Koormustestides, simuleerides ise panga allsüsteeme, eeldame, et kõik teenused annavad vastuse konstantse aja jooksul. Selleks, et saada parem ülevaade süsteemi tööst, proovime seda konstanti muuta.

Koormustestidega on kaetud kaks erinevat protsessi: 1) e-arvete töötlemine, 2) püsimumaksete sooritamine.

Äriloogiliselt ei saa nende protsesside maksimum koormuse ajad kunagi kattuda (arved peavad olema enne infosüsteemi jõudnud, kui nende pealt makseid teha saab), mis tähendab, et võime vaadelda neid protsesse eraldiseisvalt.

Kõik testid jooksevad ühes masinas. Mõlemale protsessile antakse ette 1000 arvet, mille järgi saab välja arvutada teoreetilise maksimaalse koormuse. Kuna meil ei ole reaalseid integratsioonipunkte, siis testides „mängitakse“ panga allsüsteemide teenuste kiirused erinevate parameetritega läbi. Hetkel on mõlemat protsessi jälgitud nelja erineva parameetriga, kus kõik teenuste väljakutsed võtavad:

- 1) 0.0 sekundit (võimaldab selgeks teha, mis on teoreetiline maksimum);
- 2) 0.1 sekundit;
- 3) 0.3 sekundit;
- 4) 1.0 sekundit.

#### **4.4.1 E-arve töötlemine**

E-arvete töötlemise protsess kasutab ainult ühte allsüsteemi teenust (Kliendi id küsimine IBAN-i järgi), millest on selgelt näha, et peaaegu kogu e-arve töötlemiseks kulunud aeg läheb allsüsteemi teenuse välja kutsumiseks. Tabelis 1 on näha, et süsteemis genereeritud 1000 arve töötlemiseks kulub keskmiselt 5 sekundit, millele lisandub kogu tuhande alampäringu soorituseks kulunud aeg.

Järgnevalt on toodud testi tulemused. Tabeli 1 vasakus tulbas on valitud alampäringu kestuse aeg ja paremal tulbas kogu e-arve töötlemisele kulunud aeg.

Tabel 1. E-arve töötlemise koormustesti tulemused

<b>Ühe alamsüsteemi teenuse päringu kestus</b>	<b>1000 arve töötlemise aeg</b>
0.0 sekundit	4.3 sekundit
0.1 sekundit	106 sekundit
0.3 sekundit	306 sekundit
1.0 sekund	1034 sekundit

#### 4.4.2 E-arve maksmine

Testi tulemustest on näha, et e-arve maksmise protsess on umbes kaks korda pikem, kui e-arve töötlemise protsess. Üldnimetatud protsess teeb kokku neli päringut: 1) makse sooritamine, 2) kliendi andmete päring, 3) konto andmete päring, 4) panga info küsimine IBAN-i järgi.

Makse saatmine on asünkroonne ja ei oota vastust, mis tähendab, et see ei lisa erilist ajakulu. Panga info küsimise vastuse võib süsteem soovi korral meelde jätta. Kuna üldiselt on ühel müüjal üks kindel maksekonto, siis seda teenust peab välja kutsuma vaid korra müüja kohta, mitte korra iga e-arve kohta eraldi. Tabelis 2 on näha, et süsteemis genereeritud 1000 arve maksmiseks kulub keskmiselt 7 sekundit, millele lisandub kogu tuhande e-arve kohta tehtud kahe erineva (kliendiandmete päring ja konto andmete päring) alampäringu soorituseks kulunud aeg.

Järgnevalt on toodud testi tulemused. Tabeli 2 vasakus tulbas on valitud alampäringu kestuse aeg ja paremal tulbas kogu kulunud aeg.

Tabel 2. E-arve maksmise koormustesti tulemused

<b>Ühe alamsüsteemi teenuse päringu kestus</b>	<b>1000 arve maksmise aeg</b>
0.0 sekundit	6.2 sekundit
0.1 sekundit	207 sekundit
0.3 sekundit	608 sekundit
1.0 sekundit	2007 sekundit



### **4.4.3 Koormustestide järelused**

Mittefunktsionaalsetes nõuetes oli defineeritud soovitud jõudluseks 250 000 makset päevas, arvestades süsteemile lubatud võimalikku kuni nelja tunnist tõrget. Sellest järeldeb, et meil on vaja sooritada 250 000 makset 20 tunni jooksul. Lihtne arvutus näitab, et 1000 e-arve töötlemiseks või maksmiseks tohib maksimaalselt kuluda 288 sekundit.

Selle teadmise valguses saame iga kliendi juures hinnata, et kas on vaja mõelda lisalahenduste peale. Võimalikeks lahendusteks saavad olla näiteks kliendi andmete päringu salvestamine või protsesside paralleelselt töös hoidmine.

## 5 Kokkuvõte

Antud töö põhieesmärgiks oli arendada infosüsteem, mida saaks pangasüsteemis kasutada e-arve funktsionaalsuse pakkumiseks. Arhitektuuri koostades on oluline silmas pidada nõutud jõudlust ja lihtsust integreerimisel teiste süsteemidega.

Rakendust arendades ilmnes, et üks kõige raskemaid ja riskantsemaid kohti on erinevad integratsioonid panga teiste allsüsteemidega. Antud ülesande keerukust on selgesti näha koormustestide pealt, kui aeglaselt töötav päring mõjutab drastiliselt süsteemi jõudlust. Jõudluse suhtes kriitiliste parameetrite kaardistamine võimaldab hinnata süsteemi võimekust sõltuvalt teistest allsüsteemidest. Nii saab tuvastada potentsiaalseid probleeme enne, kui need lõppkliendi poolt kasutatavas süsteemis avalduvad.

Antud lõputöös realiseeritud moodul on hetkel kasutuses mitmes eri pangas, nii Eestis kui ka väljaspool. Kõik projektid on edukalt ellu viidud ja hetkel toimuvad, kirjeldatud lahendust kasutades, sajad tuhanded e-arvete saatmised ja püsimaksete sooritamised iga kuu.

Autor saavutas lõputöös püstitatud eesmärgid. Põhinõuded, mis tulenevad e-arve standardist, on kaetud automaattestidega, näitavad, et rakenduse funktsionaalsed nõuded töötavad vastavalt Eesti Pangaliidu poolt kehtestatud e-arve standardile. Probleemivaba e-arve ja püsimakseteenuste toimimine tõestab, et antud arhitektuuriline lahendus on sobiv nõutud jõudlusparameetrite jaoks.

## Kasutatud kirjandus

- [1] E-arved avalikus sektoris - Rahandusministeerium [WWW]  
<http://www.fin.ee/e-arved>
- [2] SEPA ulatus – Pangaliit [WWW]  
<http://www.pangaliit.ee/et/arveldused/sepa>
- [3] E-arve – Pangaliit [WWW]  
<http://www.pangaliit.ee/et/arveldused/e-arve>
- [4] Ruth Wanhöfer, Transaction Banking and the Impact of Regulatory Change: Basel III and Other Challenges for the Global Economy, UK: Palgrave Macmillan, 2014
- [5] Eesti e-arve kirjelduse juhend. E-arve pankas saatmine ja e-arve presenteerimine pangas (Ver 1.05) - Eesti Pangaliit [WWW]  
[http://www.pangaliit.ee/images/files/E-arve/E\\_arve\\_saatmine%20ja%20presenteerimine%20pangas\\_ver\\_1\\_0\\_5\\_EST.pdf](http://www.pangaliit.ee/images/files/E-arve/E_arve_saatmine%20ja%20presenteerimine%20pangas_ver_1_0_5_EST.pdf)
- [6] ISO 20022 sõnumid – Pangaliit [WWW]  
<http://www.pangaliit.ee/et/arveldused/xml-b2c-a-c2b-suhtlussonumid>