

KAUNO TECHNOLOGIJOS UNIVERSITETAS

INFORMATIKOS FAKULTETAS

PROGRAMŲ INŽINERIJOS KATEDRA

Paulius Tamašauskas

**Įeinančių prekių valdymo modulis  
sandėlio valdymo sistemoje**

Magistro darbas

Darbo vadovas

dr. K. Kavaliauskas

Kaunas, 2008

KAUNO TECHNOLOGIJOS UNIVERSITETAS  
INFORMATIKOS FAKULTETAS  
PROGRAMŲ INŽINERIJOS KATEDRA

Paulius Tamašauskas

**Įeinančių prekių valdymo modulis  
sandėlio valdymo sistemoje**

Magistro darbas

Recenzentas  
dr. Sigitas Drasutis  
2008-05-26

Darbo vadovas  
dr. Kazys Kavaliauskas  
2008-05-26

Atliko  
IFM-2/2 gr. stud.  
Paulius Tamašauskas  
2008-05-26

Kaunas, 2008

## Turinys

Santrauka .....	6
Summary .....	7
1. Įvadas .....	8
1.1 Dokumento paskirtis .....	8
1.2 Tyrimo sritis, objektas ir problema .....	9
1.3 Tyrimo tikslas ir uždavinys .....	9
1.4 Tyrimo planas .....	9
1.5 Tyrimo metodai .....	10
2. Įeinančių prekių valdymo modelio analizė .....	10
2.1 Siekiamo sprendimo aprašymas .....	10
2.1.1 Efektyvi komunikacija su tiekėju .....	10
2.1.2 Tiekėjų katalogų importas .....	11
2.1.3 Užsakymų prognozavimas .....	12
2.1.4 Papildomos tiekėjų funkcijos .....	12
2.1.5 Prekių priėmimas .....	13
2.1.6 Prekių sandėliavimas .....	14
2.1.7 Įmonės sandėliai ir filialai .....	15
2.2 Įeinančių prekių modulio apribojimo reikalavimai .....	15
2.2.1 Vartotojo sąsaja .....	15
2.2.2 Skirtingų kalbų palaikymas įeinančių prekių valdymo modulyje .....	16
2.2.3 Integracija su kitais moduliais .....	16
2.2.4 Įeinančių prekių valdymo modulio saugumas .....	17
2.3 Esami sprendimai ir jų palyginimas .....	17
2.3.1 Radio Beacon sistema .....	18
2.3.2 Aiva 9001 sistema .....	18
2.3.3 Microsoft Navision sistema .....	19

2.3.4	Vision sistema.....	19
2.3.5	Sandėlio valdymo sistemų palyginimas.....	20
2.4	Aplinkos analizė .....	22
2.5	Analizės išvados .....	23
3.	Įeinančių prekių valdymo modulio projektavimas .....	23
3.1	Funkciniai reikalavimai.....	23
3.1.1	Aktoriai .....	24
3.1.2	Įeinančių prekių valdymo modulio funkcijos.....	24
3.2	Architektūros sprendimai .....	27
3.2.1	Architektūriniai tikslai ir apribojimai .....	28
3.2.2	Įeinančių prekių valdymo modulio architektūrinė realizacija.....	29
3.2.3	Vartotojų sąsajos klasių diagrama .....	30
3.2.4	Veiklos logikos klasių diagramos.....	32
3.2.5	Duomenų klasių ryšių diagrama .....	33
3.3	Projektavimo išvados .....	34
4.	Tyrimo dalis.....	34
4.1	Komunikacijos su tiekėjais pasirinkimas .....	34
4.1.1	Išskaidyta komunikacija su tiekėjais.....	35
4.1.2	Projektavimo šablono „Factory method“ panaudojimas.....	37
4.1.3	Konfigūruojamas ryšio su tiekėjais komponentas .....	38
4.1.4	Papildomų tiekėjų funkcijų panaudojimas .....	38
4.1.5	Apibendrintas išorinių sistemų duomenų valdymo modelis .....	40
4.2	Prekių priėmimo proceso galimybių padidinimas .....	41
4.3	Įeinančių prekių valdymo modulio užsakymo prognozavimo mokymosi schema... 42	
5.	Pasirinkto sprendimo spręsti komunikaciją su tiekėjais įvertinimas .....	43
5.1	Aplinkos įvertinimas.....	43
5.2	Darbo laiko ir kodo eilučių įvertinimas .....	44

5.3	Konfigūruojamo komponento efektyvumo įvertinimas .....	46
5.4	Grafinės vartotojo sąsajos efektyvumo įvertinimas .....	46
6.	Išvados .....	47
7.	Terminų ir santrumpų žodynas .....	49
8.	Literatūros šaltinių sąrašas .....	50
9.	Priedai .....	52

## **SANTRAUKA**

Rinkos analizė rodo, kad įmonėms reikalinga įeinančių prekių valdymo aplinka, kuri leistų valdyti sudėtingus prekių srautus. Esamos sistemos leidžia vykdyti tik standartines funkcijas pritaikytas visoms įmonėms, neatsižvelgiant į konkretaus užsakovo įmonės verslo taisykles. Todėl šiame darbe bus nagrinėjamas į sandėlių įeinančių prekių valdymo sprendimas. Tai yra bus aprašyti ir palyginti rinkoje egzistuojantys sprendimai, analizuojamos problemos, kurias reikia išspręsti norint sukurti efektyvų įeinančių prekių srautų valdymo modulį. Taip pat bus aprašomas kaip buvo projektuotas ir realizuotas pasirinktas sprendimas bei įvertinamos darbo metu panaudotos metodikos pademonstruojant jų pasirinkimą eksperimentiniame tyrime.

# **Incoming Goods Management Module in Warehouse Management System**

## **SUMMARY**

Market analysis shows that companies need good incoming goods management systems which would help to efficiently control vast amounts of goods flow. Common solutions existing in market only allow doing basic goods management without considering special customer needs and custom business processes. Thus this paper is researching incoming goods management module in warehouse management system. Paper describes and compares market alternatives in this field, analyzes problems which have to be solved to create an efficient incoming goods management module. Also it describes how architecture of module was formed and how it was implemented, what decisions were made and what experiments were these decisions based on.

## **1. Įvadas**

Pasaulyje egzistuoja nemažai verslo valdymo sistemų. Jos visos skiriasi daugeliu aspektu, vienos yra mažiau specializuotos konkrečiai sričiai, kitos labiau. Dauguma įmonių, užsiimančių prekių gamyba ar perpardavimu, jau dabar turi sukaupę dideles duomenų bazes su savo produktais, užsakovais ir kitais duomenimis, bei vienaip ar kitaip šias duomenų bazes panaudoja. Tokios informacijos kaupimas skaitmeniniu būdu labai paspartina įmonių atliekamą darbą. Nereikia ieškoti ar prekė tikrai yra sandėlyje, nereikia ieškoti nei kurioje vietoje ji randasi, nei kaupti dideles krūvas su klientų užsakymais susijusių dokumentų. Viskas randama akimirksniu, o informacijos panaudojimas tampa dar platesnis ir patogesnis.

Svarbią vietą tokiose sistemose užima sandėlio valdymas. Sandėlio valdymo posistemes dažniausiai sudaro daug modulių, tokių kaip sandėlio būklės stebėjimo, klientų duomenų bazė, užsakymų priėmimo, pakuotės formavimo modulis bei kiti.

Šiame darbe nagrinėjamas įeinančių prekių valdymo modulis. Modulis turi užtikrinti prekių užsakymų iš tiekėjų formavimą, užsakymo vykdymo stebėjimą, atvykusių prekių priėmimą, važtaraščių apdorojimą, bei visos informacijos, susijusios su tiekėjais ir jų tiekiamomis prekėmis, valdymą. Modulio pagrindu yra tiekėjų informacinė bazė ir jų prekių katalogai. Toks modulis turi integruotis į sandėlio valdymo sistemą bei atitikti šiai sistemai teikiamus reikalavimus. Modulis turėtų sugebėti optimaliai parinkti prekių saugojimo vietą, atsižvelgiant į prekės paėmimo iš sandėlio dažnumą, prekės transportavimo sudėtingumą ir kitus faktorius. Taip pat tokie moduliai sugeba numatyti reikiamą turėti prekių kiekį sandėlyje, kad greičiau patenkinti pirkėjų poreikius.

### **1.1 Dokumento paskirtis**

Dokumento paskirtis – užtikrinti vienodą terminologijos bei reikalavimų suvokimą naudotojų ir sistemos kūrėjų tarpe. Šiame dokumente bus pateikiama „Įeinančių prekių valdymo modulio sandėlio valdymo sistemoje“ projekto terminologija, apimanti sistemos veiklos bei techninius terminus. Taip pat bus aprašomas tiriamasis realizacijos įvertinimas ir eksperimentinis pademonstravimas.



## 1.2 Tyrimo sritis, objektas ir problema

*Tyrimo sritis:* Sandėlio valdymo sistemos.

*Tyrimo objektas:* Įeinančių prekių modulis sandėlio valdymo sistemoje.

*Tyrimo problema:* Tyrimo problema yra ta, kad įmonėms reikalinga įeinančių prekių vykdymo aplinka, kuri leistų valdyti sudėtingus prekių srautus. Esamos sistemos leidžia vykdyti tik standartines funkcijas pritaikytas visoms įmonėms, neatsižvelgiant į konkretaus užsakovo įmonės verslo taisykles.

## 1.3 Tyrimo tikslas ir uždavinys

Pagal kliento reikalavimus sukurti į sandėlį įeinančių prekių valdymo modulį, kuris padidintų komunikavimo su tiekėjais galimybes.

## 1.4 Tyrimo planas

Nr.	Užduotis
1.	Išanalizuoti literatūros šaltiniuose aprašytus egzistuojančius sandėlio valdymo sprendimus ir aprašyti panašiausias į kuriamą sistemą.
2.	Surinkti ir aprašyti kuriamos sistemos verslo taisykles bei funkcijas.
3.	Padaryti palyginamąją analizę kuriamos sistemos funkcijų su aprašytais standartiniais sprendimais.
4.	Kuriamai sistemai pasiruošti darbo aplinką.
5.	Suprojektuoti ir realizuoti įeinančių prekių modulį sandėlio valdymo sistemoje.
6.	Įvertinti naudojamą darbe metodiką ir ją pademonstruoti eksperimentiniame tyrime.
7.	Įvertinti siūlomų pakeitimų efektyvumą ir sprendimų pagrįstumą.

## **1.5 Tyrimo metodai**

Dokumentų analizė (analogų specifikacijos, analogų lyginamieji tyrimai), praktinis tyrimas.

## **2. Įeinančių prekių valdymo modelio analizė**

Šiame skyriuje panagrinėsime bendras į sandėlių įeinančių prekių valdymo sistemų savybes, sprendžiamas problemas, įgyvendinimo problemas, taip pat apžvelgsime kokius sprendimus siūlo rinka bei padarysime išvadas.

### **2.1 Siekiamo sprendimo aprašymas**

Šiame skyriuje bus aprašoma pagrindinės užsakovo verslo taisyklės, kuriomis remiantis bus įgyvendintas įeinančių prekių modulis sandėlio valdymo sistemoje.

#### **2.1.1 Efektyvi komunikacija su tiekėju**

Visos kasdieninės sandėlio įmonės operacijos reikalauja tam tikros komunikacijos su tiekėjais. Pradedant esamų prekių informacijos surinkimu, prekių kainų derinimu, užsakymų pateikimu, jų vykdymo stebėjimu prekių gamybos, komplektacijos, transporto stadijose, baigiant įvykusių klaidų apdorojimu, sugadintų prekių grąžinimu bei važtaraščių sudarymu. Neturint jokios informacinės bei tokių procesų valdymo sistemos toks kasdieninis darbas gali pareikalauti labai daug žmogiškų resursų, tuo labiau kai tiekėjų būna daug ir vien žmogiškaisiais resursais visko aprėpti nebeįmanoma. Tada sakome, kad toks darbas tampa nebe efektyvus. Daugelis įmonių, kurios užsiima prekių tiekimu, pardavimu bei sandėliavimu, naudoja vienokius ar kitokius programinius paketus. Egzistuoja dvi tiekėjų rūšys: tiekėjai, kurie yra suinteresuoti tiekti prekes konkrečiai įmonei (taip gali būti dėl įvairių priežasčių, viena jų – įmonė pateikia didžiausius užsakymus iš visų užsakovų); kita - tiekėjai, kurių prekes gauti suinteresuota įmonė, tačiau šie įmonės užsakymai, minėtiems tiekėjams, nėra reikšmingi [4]. Tai rodo, kad vieni tiekėjai bus suinteresuoti parduoti prekes ir taikytis prie siūlomų komunikacijos priemonių, o kiti tiekėjai norės, kad įmonė prisitaikytų prie jų siūlomų komunikacijos taisyklių.

Vienas populiariausių ir seniausių elektroninis komunikavimo tarp tiekėjų standartas – EDI [22]. EDI standartas nuo pat pradžių buvo kuriamas taip, kad nepriklausytų nuo žemesnio lygio technologijų ir galėtų būti perduodamas Internetu ar lokaliais tinklais. Daugelyje įmonių dirbančių su šiuo standartu dar turi įrenginių naudojamų tik ryšiui su konkrečiu užsakovu užtikrinti. Dabar vis dažniau sutinkamas EDI duomenų perdavimo Internetu

standartizavimas šiuolaikiniais MIME protokolais [18]. Naujieji standartai daug dėmesio kreipia į duomenų perdavimo saugumą, tam užtikrinti yra naudojama PGP[19] sistema, S/MIME [20]. Nepaisant to, jog egzistuoja standartai, tarp tiekėjų ir užsakovų EDI formatu keliauja daug informacijos, kuri standartuose nėra aprašyta, kas sukelia įvairių realizavimo bei korektiško duomenų interpretavimo problemų [24].

Kitas taip pat populiarus komunikavimo su tiekėjais būdas yra Web Servisai. Bėda tame, kad Web Servisai yra labai plati technologija ir nesukuria specifinės srities bendravimo standartų. Dėl šios priežasties skirtingi tiekėjai teikia skirtingas bendravimo sąsajas, kas labai apriboja bendrą tokių paslaugų panaudojimą. Šioje situacijoje išeitis yra. Tai yra išskirti kiek įmanoma plačiausią reikalingų funkcijų sritį ir apibendrinti turimus tiekėjus bei sukurti vieningą vartotojo sąsają leidžiančią sistemai bendrauti su visais tiekėjais vienodai. Nepaisant skirtingų naudojamų paslaugų, dauguma tiekėjų leidžia importuoti savo katalogus bei kurti užsakymus. Papildomos funkcijos, kaip elektroninių važtaraščių importavimas, prekių judėjimo stebėjimas, užsakymų atšaukimas bei kitos, gali būti realizuotos per įskiepus.

Visgi, dar pasitaiko smulkesnių tiekėjų, kurie neturi savo informacinės sistemos su turimų prekių katalogais. Todėl šie tiekėjai negali pateikti prekių katalogų bei priimti užsakymų elektroniniu formatu. Pasitaiko, kad užsakymai yra vykdomi ir popierinėse formose, kurias dažnai dar papildomai suvedinėjame į turimas sistemas, kad galėtume atlikti prekių judėjimo apskaitas ir analizę įvairiais pjūviais turimoje informacinėje sistemoje.

### **2.1.2 Tiekėjų katalogų importas**

Tiekėjai gali ne tik skirtingai pateikti informaciją apie prekes, tačiau gali pateikti ir skirtingą informacijos kiekį, sugrupuotą pagal skirtingus laukus. Tai gali apspręsti lokaliai saugojamų duomenų struktūrizaciją, taigi negalima rištis prie konkrečių laukų ir informacijos skirtumo problemas reikia spręsti dinamiškais sprendimais. Didelis skirtingų laukų kiekis nebūtinai reiškia skirtingą informaciją. Daugelis tokių pačių prekių pas vieną tiekėją gali vadintis vienaip, tuo tarpu pas kitą tiekėją gali vadintis kiek kitaip. Tam, kad išskirti bei suvienodinti tokias prekes reikia atlikti semantinę prekes apibūdinančių laukų analizę, išgauti esminius prekę nusakančius žodžius [1]. Sukūrus vieną bendrą įrašą lokaliaje sistemoje skirtingų tiekėjų analogiškomis prekėms galima lengviau operuoti kainų skirtumais bei sudarinėti optimalius užsakymus.

### **2.1.3 Užsakymų prognozavimas**

Įeinančių prekių valdymo modulis turi numatyti, kokių prekių ir kada reikia užsakyti iš kokių tiekėjų. Tokią informaciją galima gauti remiantis daugeliu parametru [15]:

- Ankstesnių sezonų pardavimai
- Likučiai lokaliuose sandėliuose
- Vykdomos akcijos ir ankstesnė analogiškų vykdomų akcijų patirtis
- Bendras prekių kiekis įmonės filialų tinkle

Egzistuoja trečiųjų šalių komponentai, programinė įranga, kuri sugeba prognozuoti pardavimus, taigi ir reikiamus užsakyti prekių kiekius, jiems padavus tam tikrus parametrus (ankstesnę pardavimų statistiką, kitą reikalingą informaciją).

### **2.1.4 Papildomos tiekėjų funkcijos**

Priklausomai nuo tiekėjų galimybių (programinės įrangos galimybių bei verslo modelio lankstumo galimybių), į užsakymų numatymą bei užsakymų transportavimą gali būti įtraukti papildomi parametrai. Galime išskirti kelis tokius parametrus:

- Prekių kainų priklausomybė nuo užsakyto prekių kiekio. Informaciją apie kainos priklausomybę nuo kiekio dalis tiekėjų pateikia elektroniniais kanalais (pvz., Web Servisais), su kita dalimi tiekėjų tenka bendrauti telefonu ir atitinkamai pasižymėti naują prekių kainą.
- Per tam tikrą laikotarpį, už fiksuotą kainą, iš anksto sutartas užsakymų kiekis. Tai panašus parametras į aukščiau paminėtąjį, tačiau čia užsakovo įmonė iš anksto individualiai su tiekėju sutaria dėl parametru taip dažniausiai gaudama didesnę privalumą: dar žemesnę kainą bei užtikrintą prekių kiekį reikalingu jas pristatyti laiku. Tokiomis galimybėmis dažniausiai naudojasi stambiausios įmonės užsakančios iš tiekėjų didžiąją jų pagaminamą produkcijos dalį bei galinčios įtakoti tiekėją.
- Užsakymo pristatymo galimybės. Užsakymas gali būti pristatytas į konkretų užsakovo sandėlį ar filialą arba tiesiogiai klientui. Pristatymas gali būti vykdomas su atidėjimu. Taip pat užsakovas gali atsiimti prekes pats. Visos šios galimybės įtakoja prekių pristatymo terminus, būdą bei kainą. Pavyzdžiui, prekių pervežimas bei laikymas savo sandėliuose kainuoja, tai įtakoja galutinę prekės kainą bei užsakymo naudingumą.

### 2.1.5 Prekių priėmimas

Prekių priėmimo į sandėlį procedūra gali būti atliekama su išankstiniu pasiruošimu arba be jo.

Labiau struktūrizuotose įmonėse, kur iš vieno tiekėjo užsakomų prekių kiekiai yra ganėtinai dideli (pvz.: skaičiuojama sunkvežimiais, konteineriais), darbuotojai dažniausiai yra paskirstomi operacijoms atlikti tik su konkrečiu tiekėju ar tiekėjų grupe. Tokiu atveju prekių priėmimui galima iš anksto pasiruošti – susidaryti atvykstančių prekių sąrašus, atsispausdinti važtaraščius, numatyti bei atlaisvinti vietą sandėlyje.

Įmonėse, kurios užsakymus daro dažniau ir mažais kiekiais, pristatytas prekes dažniausiai tenka tikrinti tik pristatymo metu. Tokiu atveju įeinančių prekių valdymo moduliui tenka didesnis krūvis, kadangi visus skaičiavimus susijusius su prekės saugojimo vieta bei važtaraščiu į kurį prekė turi būti įrašyta, reikia vykdyti realiu laiku. Todėl šiuo atveju turi būti kreipiamas didelis dėmesys į skaičiavimų spartą.

Įmonės, kurios būna pasiruošę priimti prekes, patiria mažiau prekių priėmimo klaidų nei įmonės, kurios nėra tam pasiruošę. Įeinančių prekių valdymo modulyje svarbiausia yra greitai atpažinti einamąją prekę. Todėl dažniausios klaidos atsiranda būtent šioje vietoje, nes jei nepavyksta atpažinti gautų prekių - sustoja judėjimas, dėl ko gali atsirasti prastovos, kurios dažniausiai gan daug kainuoja. Taigi, reikia įvertinti prekių priėmimą įvairiais aspektais, norint nepatirti prastovų. Tai yra, naudojamus įrankius prekių priėmimui, gamybos projektų valdymo metodikas (pvz.: „Kritinės grandinės“) ir t.t.

Dažniausiai sprendžiamos priėmimo problemos:

- **Ką daryti su pažeista preke.** Dažniausiai tokios prekės yra ta pačia siunta (jei įmanoma) gražinamos tiekėjui – tiesiog nepriimamos. Tokiu atveju reikia pakeisti bei perskaičiuoti važtaraščius, nuspręsti ką daryti su prekių trūkumu (aprašyta žemiau).
- **Kaip identifikuoti prekę neturint jos bar-kodo ar kitos identifikuojančios informacijos.** Šiai problemai spręsti prekių priėmimo vartotojo sąsajoje turi būti galimybė nesunkiai susirasti prekę pagal bet kokį jos atributą (pvz.: pavadinimą, apibūdinimą, tiekėją, tiekėjo prekės identifikatorių, prekės kodą, gamintoją ir t.t.). Taip pat turi būti galimybė priskirti prekei naują sugeneruotą bar-kodą. Jei bar-kodas yra pažeistas, jį gali reikėti perspausdinti (kadangi prekės gali keliauti konvejeriu, kuris jų pristatymą į reikiamą vietą kontroliuoja sprendamas nuskanavus prekės bar-kodą).

- **Kaip spręsti prekių trūkumą.** Esant prekių trūkumui (kai buvo numatyta jog siuntoje bus didesnis prekių kiekis nei iš tiesų buvo) tenka registruoti pristatymo klaidą, bendrauti su tiekėju. Vartotojas turi nuspręsti kokio veiksmo imtis toliau: laukti kol atvyks prekės, neuždarant (nepabaigiant) užsakymo arba koreguoti patį užsakymą, kad jis būtų pabaigtas ir galima būtų daryti atvykusių prekių apskaitą, pajamavimą.
- **Kaip spręsti prekių perteklių.** Tai rečiausiai pasitaikanti situacija, kurios sprendimas priklauso nuo susitarimo su tiekėju. Jei prekė yra užsakytų prekių sąrašė, o skiriasi tik prekių kiekis, tai dažnas atvejis, kad ji bus priimta į sandėlį. Jei prekė priimama, gali tekti pakeisti užsakymą, kad suskaičiuoti važtaraščius bei sąskaitas.

Iš aukščiau aprašytų problemų matosi, kad sėkmingam prekių priėmimui į sandėlį vykdyti reikia pakankamai daug prekių paieškos funkcijų bei užsakymų, sąskaitų modifikavimo funkcijų.

### 2.1.6 Prekių sandėliavimas

Gan svarbu prekių priėmimo metu, naujoms atvykstančioms prekėms atlaisvinti vietą sandėlyje, kitaip sakant, paruošti vietą, kur bus prekės sudėtos ir sandėliuojamos iki kol jos išvyks. Prekėms parinkti sandėliavimo vietą galima remiantis skirtingais optimizacijos tikslais:

- **Optimizuoti pagal numatomą prekės saugojimo laiką sandėlyje.** Toks optimizacijos būdas būtų naudojamas, kai orientuojamasi į greitą sandėlio atlaisvinimą - prekės sudedamos pagal tai kada jos išvyksta iš sandėlio. Pavyzdžiui, anksčiausiai išvykstančios prekės bus sandėliuojamos pasiekiamiausioje vietoje. Todėl ilgiausiai laikomas prekes pasiekti yra sunkiausia (kai kurios įmonės turi atskiras ilgojo saugojimo patalpas, kuriose dažniausiai saugomi didžiausi neišpakuoti prekių kiekiai paletėse).
- **Optimizuoti pagal prekių komplektaciją galutiniam vartotojui.** Kai pirkėjai užsisakinėja po daugiau nei vieną prekę, prekes prieš išsiunčiant tenka surinkti į vieną pakuotę. Kad tai efektyviau atlikti surinkimo metu, galima prekes jau joms atvykus grupuoti pagal jų išvykimo komplektaciją.
- **Optimizuoti dėl greičiausio prekių priėmimo.** Dažnai prekes atvežusius sunkvežimius reikia labai greitai atlaisvinti, todėl prekes tenka dėlioti kiek įmanoma greičiau, kiek įmanoma artesniais atstumais. Tokiu būdu dėliojant prekes dažnai

turima omenyje, kad prekių padėtis vėliau bus vėl optimizuojama vienu iš aukščiau paminėtų būdų.

Didesnius sandėlius turinčiose įmonėse prekių paėmimą ir sandėliavimą dažnai realizuoja tam tikslui skirtos mašinos, be jokio žmogaus įsikišimo. Įeinančių prekių modulio bendravimas su tokiomis mašinomis vyksta per tam skirtas vartotojo sąsajas. Dalis tokių mašinų turi tik sandėlio lentynų duomenų bazę, kurioje saugomas lentynų užimtumo statusas, lentynų dydis bei prekės identifikavimo numeris. Kitos sistemos pačios atlieka optimizavimą, leisdamos atlikti tik prekės atidavimo (su tam tikrais prekės saugojimo parametrais) bei paėmimo operacijas.

### **2.1.7 Įmonės sandėliai ir filialai**

Įmonei, kuri turi daugiau nei vieną sandėlį ar filialą, reikia spręsti didesnę uždavinį – į kurią sandėlį ar filialą užsakyti prekes iš tiekėjo, kad jų pristatymas kainuotų mažiausiai. Tai nereiškia, kad pristatinėti prekes į artimiausią sandėlį yra ekonomiškiausia. Reikia įvertinti galimybę transportuoti prekes tarp sandėlių arba pristatinėti prekes tiesiai užsakovui (pirkėjui). Transportavimą tarp sandėlių galima būtų apibendrinti kaip užsakymą pas atitinkamą tiekėją, pas kurį prekės vertė lygi transportavimo išlaidoms. Žinoma tarp sandėlių transportuojamos turėtų būti tik nerezervuotos arba pirmojo būtinumo prekės.

## **2.2 Įeinančių prekių modulio apribojimo reikalavimai**

### **2.2.1 Vartotojo sąsaja**

Bene svarbiausias programinės įrangos elementas – paprasta ir nesudėtinga bei tuo pačiu funkcionali vartotojo sąsaja. Vartotojas turi matyti visas reikalingas funkcijas ir pasikeitimus realiu laiku. Dažniausiai sandėlio valdymo sistemose informacijos kiekis yra labai didelis, tai lemia įvairūs formuojami dokumentai bei skirtingais pjūviais atliekama statistika. Informacijos išdėstymas ekrane vaidina svarbią rolę [4]. Taip pat reikia atsižvelgti į vartotojo sąsajos elementų pakartotinį panaudojimą [17]. Įvairūs statistiniai elementai gali būti naudojami keliuose skirtinguose programinės įrangos moduluose, tačiau vaizdžiai turi išikomponuoti į konkretų vartotojo sąsajos elementą taip, kad bereikalingai nenukreiptų vartotojo dėmesio. Testuojant sistemą, būtina įsitikinti jog vartotojo sąsaja yra nesunkiai valdoma bei joje pateikiama visa vartotojui reikalinga informacija [5].

### **2.2.2 Skirtingų kalbų palaikymas įeinančių prekių valdymo modulyje**

Dirbant tarptautinėje rinkoje, turint sandėlius bei darbuotojus įvairiose šalyse, yra svarbu, kad sistema galėtų nesudėtingai veikti keliomis skirtingomis kalbomis. Būtina sukurti tokią sistemos architektūrą, kad ji nesudėtingai leistų įvesti naują kalbą vartotojo sąsajai. Gali iškilti specifinių rašmenų atvaizdavimo bei saugojimo problema, kurią spręsti galima pasinaudojant Unicode simbolių kodavimo sistema. Analogiškoms lokalizacijos problemoms spręsti kelios įmonės siūlo pasinaudoti jų produktais-įrankiais, kurie integruoja savo kodą į kuriamos sistemos platformą ar pasiūlo patogius būdus realizuoti vartotojo sąsajos vertimą bei skirtingų kalbų vartotojų sąsajos palaikymą. Reikia paminėti, kad vartotojo sąsajos lokalizavimas tai tik viena problemos pusė. Kita pusė – sistemos išeigos dokumentų lokalizavimas. Dokumentus siunčiamus į užsienio šalis gali tekti versti į užsienio kalbą. Vartotojas ekrane dokumentą nori matyti savo kalba, tačiau atspausdintas jis turi išeiti tos šalies kalbos, į kurią dokumentas bus siunčiamas. Taip pat svarbu nepamiršti jog skirtingose šalyse skiriasi matavimo vienetai bei valiuta, todėl kuriant sistemą gali tekti atsižvelgti ir į nutolusių sandėlių bendravimo tarpusavyje vienetų ar valiutos konvertavimą.

### **2.2.3 Integracija su kitais moduliais**

Įeinančių prekių valdymo modulis turi egzistuoti ir be kitų sistemos modulių arba bendrauti su kitais moduliais tik per prekių likučio sandėlyje duomenų struktūras. Integracija su kitais moduliais, leidžia vartotojui naudotis platesniu funkcijų spektru [14]. Įeinančių prekių valdymo modulis integruojasi su kitais moduliais dėl šių priežasčių:

- užsakymo iš tiekėjo pristatymas tiesiogiai klientui (integruojant su CRM moduliu),
- automatinis elektroninių tiekėjų sąskaitų apmokėjimas integruojant su apskaitos moduliu,
- kainų maržos kainodaros modulyje koregavimas atsižvelgiant į užsakomus iš tiekėjų kiekius bei tiekėjų siūlomas mažesnes kainas šiems kiekiams,
- prekių alternatyvų siūlymas atsižvelgiant į garantinio aptarnavimo modulio pateiktą prekių gražinimo statistiką,
- prekės padėjimo į sandėlį vietai nustatyti taip pat gali reikėti integracijos su CRM bei klientų užsakymų moduliu, jei pasirinkta prekių vietą sandėlyje optimizuoti pagal užsakytų prekių komplektaciją bei pristatymo vietą.

Visas papildomas funkcionalumas pasiekiamas tam tikra kaina: reikia įdėti daugiau darbo sudarant reikalingus ryšius, juos palaikant.



## 2.2.4 Įeinančių prekių valdymo modulio saugumas

Įeinančių prekių modulio saugumą galima nagrinėti iš trijų aspektų: vartotojų prisijungusių prie modulio, modulio funkcijomis besinaudojančių kitų sistemų ar modulių, ir modulio ryšio su kitomis (tiekėjų) sistemomis aspektu.

Vartotojo sąsaja sąlygoja tai kaip sistema bus apsaugota nuo nesankcionuoto ar nekorektiško jos panaudojimo. Kadangi įmonės viduje naudojami Windows operacinės sistemos serveriai bei darbo stotys, tai natūralu būtų pasinaudoti Windows sistemos teikiamomis autentifikacijos priemonėmis ir taip suskaidant vartotojus, leisti jiems prieiti prie tik jiems leistinų sistemos modulių.

Web Servisų panaudojimas leidžia nutolusiems kompiuteriams pasinaudoti sistemos funkcijomis per Internetą, tačiau taip pat sukelia saugumo problemų. Kad saugiai perduoti duomenis, galima pasinaudoti duomenų šifravimu (pvz.: VPN/IPSec, 3DES [18]). Tai reikalauja papildomos tinklo konfigūracijos bei techninės įrangos resursų, tačiau yra būtina norint saugiai komunikuoti tarp nutolusių įmonės padalinių ar sandėlių.

Ryšys su tiekėjų bei kitomis sistemomis taip pat atliekamas interneto kanalais. Kadangi čia naudojami kitokie protokolai, reikia atsižvelgti į šių protokolų saugumą. MIME protokolais atliekamo ryšio saugumą galima užtikrinti pasinaudojus PGP[19] bei S/MIME [20].

## 2.3 Esami sprendimai ir jų palyginimas

Šiame skyriuje bus aprašomi jau egzistuojantys rinkoje produktai, kurie atlieka panašias į kuriamo įeinančių prekių valdymo modulio funkcijas. Daugelis nagrinėjamų pavyzdžių yra didelių įmonių produktai, todėl kai kurie bus žinomi. Verta pastebėti, kad esami sprendimai yra standartiniai, dar kitaip vadinami „universalūs“ sandėlio valdymo sistemų programiniai paketai. Todėl jie atlieka ne vien įeinančių prekių į sandėlį valdymą, bet ir kitas prekių judėjimo operacijas.

Žemiau pateiktuose skyriuose bus analizuojamos ir lyginamos šios sistemos:

- Radio Beacon
- Aiva 9001
- Microsoft Navision
- Vision
- Kuriamas įeinančių prekių valdymo modulis sandėlio valdymo sistemoje

### **2.3.1 Radio Beacon sistema**

Radio Beacon [6] 1992 metais išleido pirmąją savo sandėlio valdymo sistemos versiją, skirtą vidutiniam verslui. Šiuo metu sistema yra pilnai pritaikyta surinkimo, pakavimo ir gabenimo sprendimams taikyti didmeninėje prekyboje.

Pagrindinės sistemos funkcijos: prekių priėmimas, atpažinimas, prekių apdirbimas, prekių saugojimo kontrolė, prekių surinkimo iš sandėlio organizavimas, užsakymų valdymas, pirkėjų aptarnavimas.

Radio Beacon sistemos pagrindas yra automatinis prekių atpažinimas, kuris remiasi brūkšniniu kodavimu. Tai leidžia sistemoje užtikrinti tikslumą, patikimumą bei greitą darbą. Kad informacija patektų į sistemą, yra naudojami mobilūs radijo terminalai su integruotais brūkšninio kodo skaitytuvais. Sistema stengiasi optimizuoti prekių išvykimą iš sandėlio net iki tokio lygio, kad prekės iš tiekėjų atvykusio transporto yra pakraunamos tiesiai į paskirstantį prekes klientams transportą, taip net neužimant vietos sandėlyje. Radio Beacon sistema ne tik greitai tvarkosi su dideliais prekių kiekiais, bet ir leidžia priimti į sandėlį neužsakytas ar tam tikrų dokumentų neturinčias prekes (jos yra padedamos į karantino zoną). Sistema optimizuoja gautų prekių paskirstymą po sandėlį.

### **2.3.2 Aiva 9001 sistema**

Tai lietuvių sukurta verslo valdymui ir elektroninei komercijai skirta sistema [7]. Ją sudaro keletas modulių, tokių kaip parduotuvė, marketingas, CRM, komercija, katalogas bei prekių apskaita. Prekių apskaitos modulį sistemos kūrėjai apibūdina kaip sandėlio valdymo sistemą.

Sistemos atliekamos funkcijos: įvesti turimas sandėlyje prekes, užsakyti prekes sandėlio papildymui, užpajamuoti prekes pagal padarytus užsakymus tiekėjams, išrašyti sąskaitas faktūras, gauti įvairias reikalingas buhalterijai ataskaitas apie prekių judėjimą, likučius.

Sistema naudojasi virš 20 Lietuvos bei užsienio šalių įmonių. Sistema paremta tiesioginio užsakymo principu, t.y. padeda organizuoti užsakymus tiekėjams taip, kad prekių pristovos sandėlyje būtų kuo mažesnė. Aiva 9001 kūrėjai tvirtina jog jų sistema atitinka visus ISO 9000 serijos kokybės standartus.

### **2.3.3 Microsoft Navision sistema**

Ši sistema yra sukurta Danijoje. Ji išversta į daugelį kalbų, tuo tarpu ir lietuvių bei pritaikyta atitinkamų šalių įstatymų reikalavimams. Microsoft Business Solutions - Navision – tai ne tik gerai žinoma, bet ir labiausiai paplitusi verslo valdymo sistema pasaulyje. Navision – tai integruota, modulinė atviro tipo sistema. Ryšys tarp atskirų modulių funkcijų atliekamas naudojantis vieninga duomenų baze. Duomenys į sistemą įvedami vieną kartą, vėliau gali būti apdorojami ir interpretuojami kitų modulių programiniais instrumentais [8].

Tai sistema, kuri yra nuolatos papildoma naujomis galimybėmis ir ištaisais moduliais, atsižvelgiant į rinkoje sukauptą patirtį ir naujausias tendencijas verslo informacinių sistemų srityje.

Pagrindinę programą sudaro tokios integruotos funkcinės taikymo sritys (moduliai): didžioji knyga - skirtas finansų apskaitai, ilgalaikis turtas, pardavimai, verslo ryšių valdymas (CRM), aptarnavimo valdymas, pinigų valdymas, pirkimai, Atsargos, sandėlio valdymas, gamyba, ištekliai, personalas, darbai, bendra programinė sritis.

### **2.3.4 Vision sistema**

VISION [9] – tai sandėlio valdymo sistema sukurta Lietuvoje, skirta didelėms ir vidutinėms įmonėms. Savo paskirtimi ir veikimo principais panaši į kitas sandėlio valdymo sistemas. Informacija suvedama naudojant brūkšninio kodo skanavimą. Vision architektūra leidžia įrangą pateikti kaip atskirą universalų produktą, arba priderinti kiekvienam klientui pagal jo specifinius poreikius.

Taip pat, ši sistema leidžia suderinti operacijas iškart keliuose sandėliuose, darbo zonose, daugeliu apdirbimo būdų. Kiekvienos prekės judėjimas sandėlyje gali būti lengvai patikrinamas realiu laiku, naudojant radijo bangomis valdomus brūkšninių kodų skanerius. Sistemoje yra integruota transporto valdymo sistema leidžianti suoptimizuoti išvykstančio bei atvykstančio transporto srautus. Sistema gali apskaičiuoti optimaliausią maršrutą su minimaliu kelionės laiku ir nukreipia operatorių į reikiamą sekciją. Tiekėjų kontrolės modulis padeda efektyviai priimti prekes. Prekių informacija su tiekėjais sinchronizuojama EDI protokolais.

### 2.3.5 Sandėlio valdymo sistemų palyginimas

Visas verslo valdymo bei analogiškas operacijas atliekančias sistemas galima skirstyti į keletą tipų [10]:

- Specifinės srities verslo valdymo sistemos. Tai tik konkrečiai sričiai pritaikytos, pavyzdžiui, sistemos transporto tvarkaraščiams sudaryti, restorano patiekalams užsakyti, prekėms sandėlyje valdyti ir panašiai.
- Universalios, viskas viename tipo sistemos. Tai tokios sistemos, kurios bando aprėpti visą verslo sritį. Jos dažniausiai susideda iš daugelio atskirų modulių bei jų architektūra būna atviro tipo, taip leidžiant trečioms šalims kurti papildomas sistemos įskiepius bei pilnus modulius.
- Trečios šalies įskiepiai, galintys funkcionuoti atskirai. Tai dažniausiai atliekančios konkrečias valdymo funkcijas bei įvairius įrenginius valdančios sistemos. Jos nesudėtingai integruojasi į vieną ar kelias antrojo tipo sistemas.

Pagal ankstesnį apibūdinimą čia nagrinėtas sistemas galima suskirstyti taip:

- Radio Beacon – trečiojo tipo, kadangi atlieka konkrečias sandėlio valdymo operacijas, iš kurių labiausiai pabrėžiamos bei daugiausia dėmesio skiriama konvejerinių įrenginių valdymui, priiminėjant prekes į sandėlį bei formuojant pirkėjų užsakymus, o taip pat automatiniam prekių apdirbimui. Ši sistema yra puikiai pritaikyta integravimui į Microsoft Navision verslo valdymo sistemą, tačiau gali funkcionuoti ir be jos. Šią sistemą galima būtų priskirti ir pirmam tipui, nes jos atliekamos funkcijos yra kur kas platesnės nei paprasto įskiepio.
- Microsoft Navision – antrojo tipo sistema, kadangi bandoma aprėpti visas įmanomas verslo valdymo funkcijas. Turi modulinę struktūrą todėl perkant sistemą galima įsigyti tik atitinkamai reikalingus modulius. Sandėlio valdymo bei įeinančių prekių posistemės su standartine sistemos versija negali kontroliuoti sandėlyje esančių įrenginių.
- Aiva 9001 – taip pat antrojo tipo sistema, kurioje bandoma aprėpti daug pardavimo proceso automatizavimo operacijų, tačiau jos nėra tokios pilnos kaip Microsoft Navision sistemoje [6].

- Vision – pirmojo tipo sistema, skirta daugiausia valdyti prekių judėjimą sandėlyje, į jį bei iš jo, atlikti analogiškas funkcijas kaip ir Radio Beacon sistema.
- Kuriamas įeinančių prekių valdymo modulis (IPVM) – pirmojo tipo sistema, skirta įeinančių prekių valdymui.

Visų nagrinėtų sistemų (išskyrus kuriamos sistemos) vartotojo sąsajos yra sukurtos hipertekstinių sistemų pagrindu, todėl jomis galima naudotis įvairiose operacinėse sistemose, įvairiomis naršyklėmis. Dauguma funkcijų bei operacijų, atliekamų šiose sistemose, realizuotos Web servisų pagrindu, kas leidžia tam tikrą funkcionalumą perkelti į kitą analogiškais principais veikiančią sistemą [7].

Galima teigti jog iš nagrinėtų sistemų, Microsoft Navision bei Aiva 9001 įeinančių prekių priėmimo funkcionalumas nusileidžia Vision bei Radio Beacon sistemų galimybėms. Taip yra todėl, kad MS Navision ir Aiva 9001 atlieka tik prekių apskaitos funkcijas, bei tiesiogiai nesiintegruoja su sandėlyje esančiais įrenginiais [8]. Tiesioginio komunikavimo su tiekėjais sistemą EDI siūlo tik Radio Beacon ir Vision sistemos.

Aiva 9001 bei Vision rinkoje atsirado pakankamai neseniai, todėl šių sistemų atliekamos funkcijos nėra iki galo išstobulintos bei parodo mažesnę funkcionalumą nei rinkos lyderių sistemos. Tuo tarpu Microsoft Navision ir Radio Beacon sukurtos gana seniai ir yra pastoviai atnaujinamos pagal rinkos poreikius. Šios technologijos turi savo senas ir tvirtas tradicijas, todėl kūrėjų komanda turi daug patirties ir gali pasiūlyti kokybišką produktą.

Radio Beacon vienintelė iš nagrinėtų sistemų integruojasi su RFID technologijomis – gali automatiškai atpažinti prekes pasinaudojant mini radijo siųstuvais integruotais į pakuotes. Tai vis labiau plintanti ateities technologija [9] ypatingai paspartinanti atvykusių prekių priėmimą bei sumažinanti galimas klaidas inventorizuojant, padidinanti sandėlio saugumą.

Žemiau lentelėje pateiktas aprašytų sistemų funkcinis palyginimas.

1. lentelė. Sandėlio valdymo sistemų palyginimas

Sistemos pavadinimas	Tiekėjų bei jų prekių DB	Užsakymų pasiūlymas	Prekių išdėstymo optimizavimas	Komunikacija su tiekėju EDI protokolu	Skanavimo įrangos palaikymas	Sprendimas pritaikytas užsakovo verslo taisyklėms
Radio Beacon	+	-	+	+	+	-
Aiva 9001	+	-	-	-	-	-
Microsoft Navision	+	+	+/- (per įskiepius)	+/- (per įskiepius)	+/- (per įskiepius)	-
Vision	+	+/-	+/-	+	+	-
IPVM	+	+	+	+	+	+

Sandėlio valdymo sistemų analizė parodė, kad įeinančių prekių valdymo modulis turi visas pagrindines į sandėlį įeinančių prekių valdymo funkcijas. Tuo labiau, kai palyginimas buvo daromas su Lietuvos rinkoje etaloninėmis sandėlio valdymo sistemomis. Todėl kurti tokią sistemą yra geras pasirinkimas, nes tokia sistema savo funkcionalumai gali drąsiai konkuruoti su jau seniai rinkoje egzistuojančiomis sistemomis. Taip pat, kuriamas įeinančių prekių modulis turi didelį privalumą, kurio neturi analizuotos sandėlio valdymo sistemos – įeinančių prekių valdymo modulis buvo kuriamas atsižvelgiant į visas užsakovo verslo taisykles, o ne pritaikytas standartas daugeliui įmonių.

## 2.4 Aplinkos analizė

Pagal statistiko departamento atlikto tyrimo duomenis [2], mažmenine bei didmenine prekyba užsiimančių įmonių Lietuvoje yra apie 25 tūkstančiai, apie 3 tūkstančiai iš jų turi daugiau nei po 20 darbuotojų. Taip pat daugėja įmonių užsiimančių prekių sandėliavimu. Tokios įmonės gali būti potencialūs mūsų kuriamos sistemos klientai. Taip pat iš statistikos duomenų [3] galime matyti jog įmonių susidomėjimas verslo automatizavimu bei kompiuterizavimu auga. Iš to galime spręsti, kad analogiškų sistemų paklausa Lietuvoje tik didės.

Lietuvoje egzistuoja ir panašių sistemų analogų, atsiranda įmonių siūlančių užsienyje kurtas analogiškų sistemų versijas. Dauguma sistemų, kurios atkeliauja iš užsienio siūlo daug technologinių naujovių, kurios dar Lietuvos rinką nepasiekę. Todėl pirkti produktą su

papildomu funkcionalumu už kurį reikia sumokėti ir kurio galbūt neteks niekad panaudoti – nėra geriausias variantas. Įmonės vertina sistemas kokybės, funkcionalumo, investuotų pinigų ir naudingumo aspektais. Todėl dažnai sprendimai būna susikurti individualią sistemą, kuri kompiuterizuotų visus įmonės verslo procesus.

## **2.5 Analizės išvados**

1. Rinkos analizė parodė, kad įmonėms nepakanka standartinių įeinančių prekių valdymo sprendimų
2. Literatūros šaltinių analizės metu nustatyta, kad kuriamas įeinančių prekių valdymo modulis turi visas pagrindines standartines įeinančių prekių valdymo funkcijas lyginant su etaloninėmis sandėlio valdymo sistemomis. Todėl kurti tokią sistemą yra geras pasirinkimas, nes tokia sistema savo funkcionalumu gali drąsiai konkuruoti su jau seniai rinkoje egzistuojančiomis sistemomis. Taigi pasirinktas sprendimas – sukurti įeinančių prekių valdymo modulį.
3. Projektuojant modulį reikia išskirti 3 pagrindinius komponentus:
  - Tiekėjų katalogų duomenų bazė
  - Užsakymų sudarymas
  - Prekių priėmimas

## **3. Įeinančių prekių valdymo modulio projektavimas**

Projekto metu buvo sukurtas ne atskiras produktas, o kito produkto modulis, posistemė. Pagrindinis produktas yra sandėlio valdymo sistema susidedanti iš daugelio besirišančių modulių. Sukurto modulio funkcija – valdyti įeinančių į sandėlį prekių srautus. Modulio funkcijomis naudojasi ne tik realūs vartotojai, tačiau ir keli kiti visos sistemos moduliai.

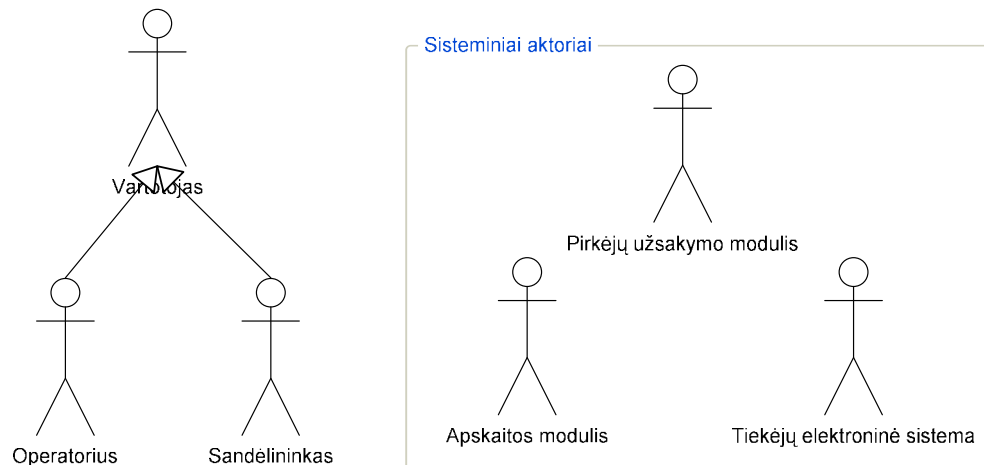
Šiame skyriuje bus aprašomas kuriamo įeinančių prekių valdymo modulio projektavimas. Žemiau esantys skyriai bus padalinti į dvi pagrindines grupes: funkcinių reikalavimų ir architektūros aprašymus.

### **3.1 Funkciniai reikalavimai**

Šiame skyriuje bus aprašomi sistemai keliami funkciniai reikalavimai.

### 3.1.1 Aktoriai

Įeinančių prekių valdymo modulyje egzistuoja tokie aktoriai:



*1 pav. Aktorių diagrama*

Operatorius – įeinančių prekių valdymo modulario aktorius, kuris gali sudarinėti užsakymus, užsakymus išsiųsti tiekėjui, peržiūrėti pirkimus, valdyti pristatymo informaciją.

Sandėlininkas - įeinančių prekių valdymo modulario aktorius, kuris gali priimti prekes.

Pirkėjų užsakymo modulis - įeinančių prekių valdymo modulario sisteminis aktorius. Šis aktorius surenka pirkėjų užsakomas prekes ir automatiškai formuoja užsakymus tiekėjams.

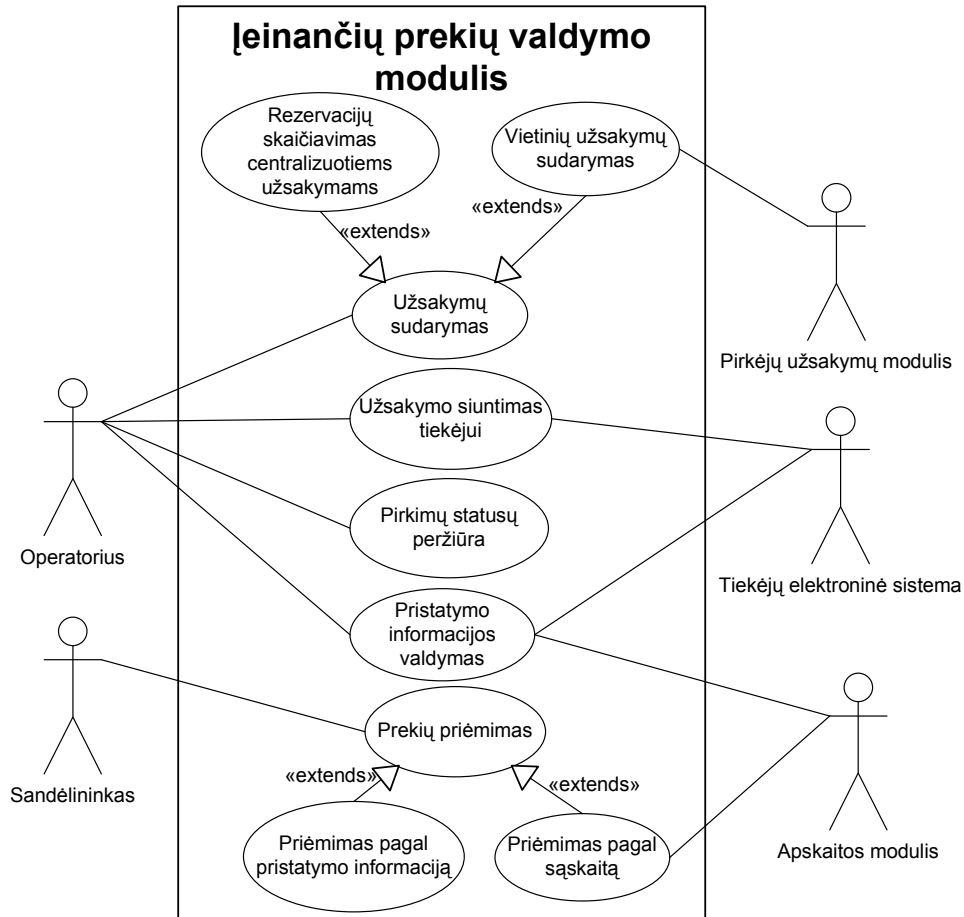
Tiekėjų elektroninė sistema - įeinančių prekių valdymo modulario sisteminis aktorius. Šis aktorius atstovauja tiekėjo sistemą. Jis naudoja jam skirtą informaciją apie pristatymo vietą ir užsakymus.

Apskaitos modulis - įeinančių prekių valdymo modulario sisteminis aktorius. Šis aktorius atsakingas už sąskaitą atvykusioms prekėms paruošimą.

### 3.1.2 Įeinančių prekių valdymo modulario funkcijos

Funkciniai reikalavimai bus aprašomi naudojant panaudos atvejus. Visi įeinančių prekių valdymo modulario panaudos atvejai yra nubraižyti žemiau esančiame paveikslėlyje (žr. 2 pav.).



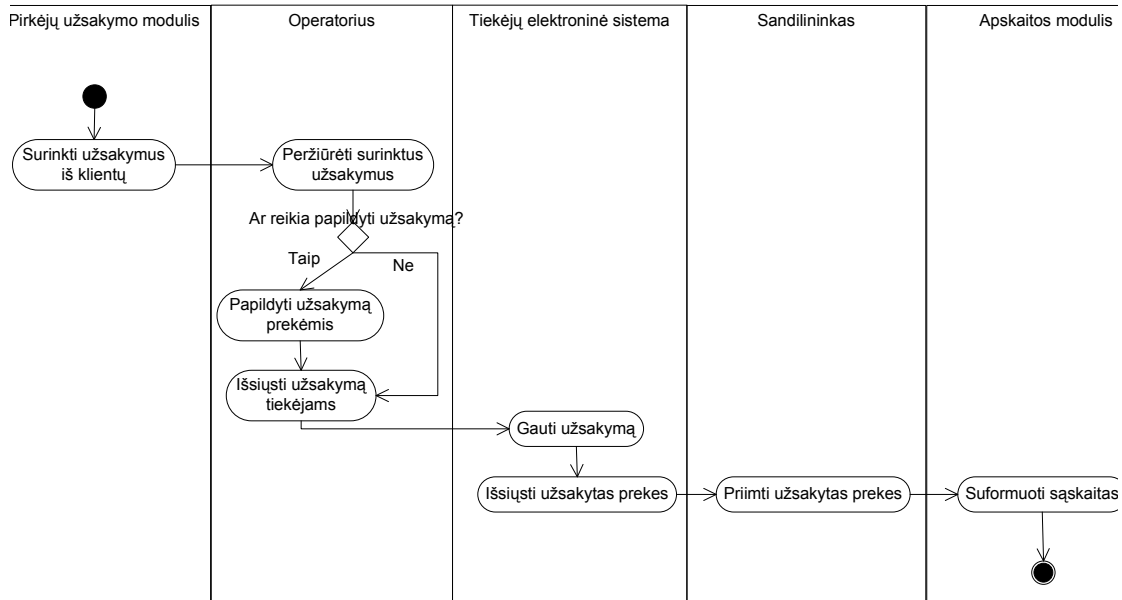


2 pav. Sukurto modulio panaudojimo atvejų diagrama

Modulyje buvo realizuotas toks funkcionalumas:

1. Užsakymų sudarymas bei sudarytų užsakymų peržiūrėjimas. Operatorius sukuria naują užsakymą, į jį sudeda reikiamas prekes, pasirenka norimus tiekėjus, pristatymo datą. Taip pat operatorius gali peržiūrėti jau egzistuojančius užsakymus.
2. Vietinių užsakymų sudarymas, paremtas pirmu panaudojimo atveju. Jį dažniausiai naudos pirkėjų užsakymo modulis. Operatorius sudaro vietinius užsakymus, kurie bus pristatyti į operatoriaus filialą.
3. Centralizuotų užsakymų sudarymas bei rezervuotų prekių skaičiavimas. Rezervuotų prekių kiekiai turi būti skaičiuojami įvertinant visas vietiniuose sandėliuose trūkstamas prekes.

4. Užsakymų, sudarytų aukščiau paminėtuose panaudojimo atvejuose, siuntimas tiekėjui. Operatorius turi galėti išsiųsti suformuotą užsakymą tiekėjui. Nepavykus išsiųsti iš pirmo karto, operatorius turi galėti pakartoti siuntimą.
5. Pirkimų vykdymo statuso stebėjimas, pirkimą sudarančių prekių bei jų atvykimo statusų peržiūra. Turi būti galimybė stebėti vykdomo pirkimo statusą bei jį sudarančių prekių atvykimo grafiką, prekių atvykimo/neatvykimo statusą.
6. Informacijos apie pristatomas prekes suvedimas, t.y. numatomos atvykimo datos, atvykstančių prekių kiekių bei kainų valdymas. Operatorius turi galėti pažymėti kada numatomas prekės atvykimas, koks prekės kiekis turi atvykti tą datą ir kokiomis kainomis. Ši informacija gali būti importuojama iš tiekėjo, jei tiekėjas tokią operaciją palaiko.
7. Atvykusių prekių priėmimas, skenuojant prekes. Skenuojama prekė pažymėta kaip priimta į sandėlį, jos saugojimo vieta galutinai parinkta ir ji sistemos valdomu konvejeriu važiuoja link saugojimo vietos.
8. Priėmimas pasinaudojant ankstesniame panaudojimo atvejuje suvesta informacija apie pristatymą. Prekes turi būti galima priimti remiantis pristatymo informacija suvesta prie pirkimo. Atvykus prekėms sutikrinamas koks jų kiekis turėjo atvykti, jos nuskanuojamos ir pažymimos atvykusiomis.
9. Priėmimas pagal sąskaitą gautą iš sąskaitų modulio. Turi būti galimybė iš sąskaitų modulio gauti informaciją apie šiuo metu skenuojamų prekių sąskaitą ir pagal ją atlikti skanavimą, sutikrinti atvykusių prekių teisingumą.

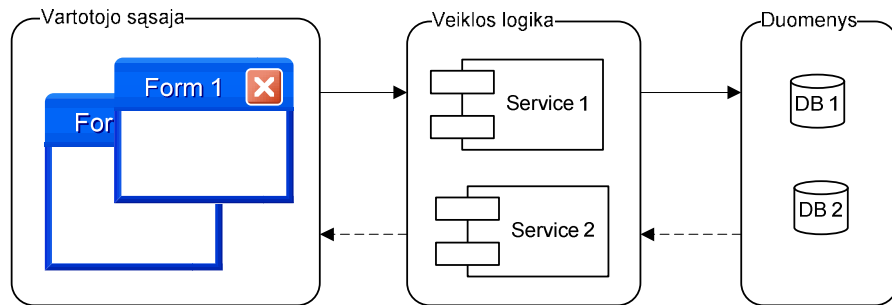


3 pav. Prekių užsakymo ir gavimo judėjimo schema

### 3.2 Architektūros sprendimai

Sistema yra realizuota trijų lygių architektūra, kurią sudaro:

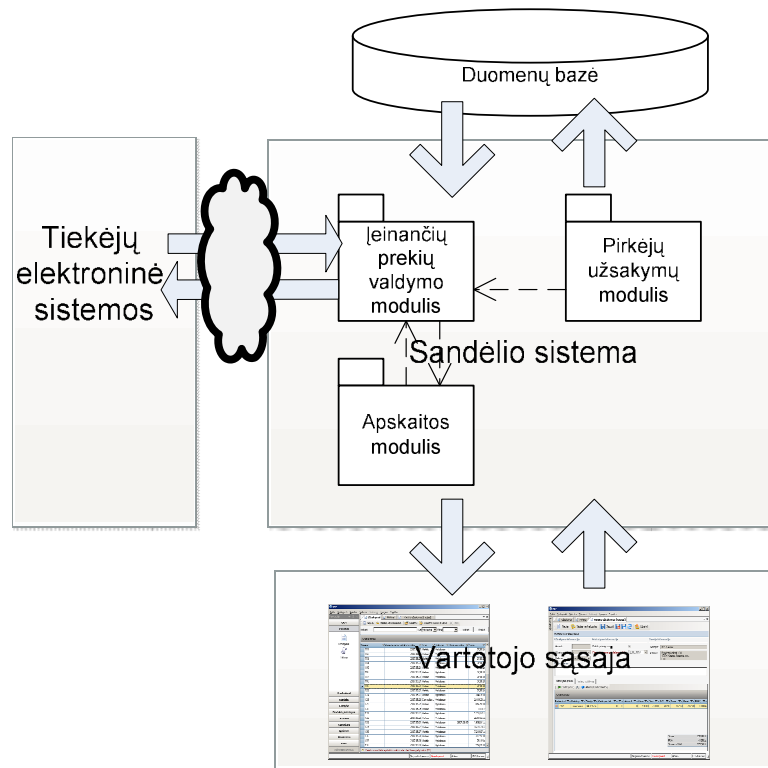
- Vartotojo sąsaja,
- Veiklos logika,
- Duomenys.



4 pav. Trijų lygių architektūra

Tokia architektūra leidžia atskirti vartotojo sąsają nuo veiklos logikos, kas užtikrina sistemos pakartotinį panaudojamumą.

Žemiau pateiktame paveikslėlyje (žr. 5 pav.) yra pateikiama įeinančių prekių valdymo modulio ir sandėlio valdymo sistemos architektūrinė schema.



5 pav. Įeinančių prekių valdymo modulio ir sandėlio valdymo sistemos architektūrinė schema

### 3.2.1 Architektūriniai tikslai ir apribojimai

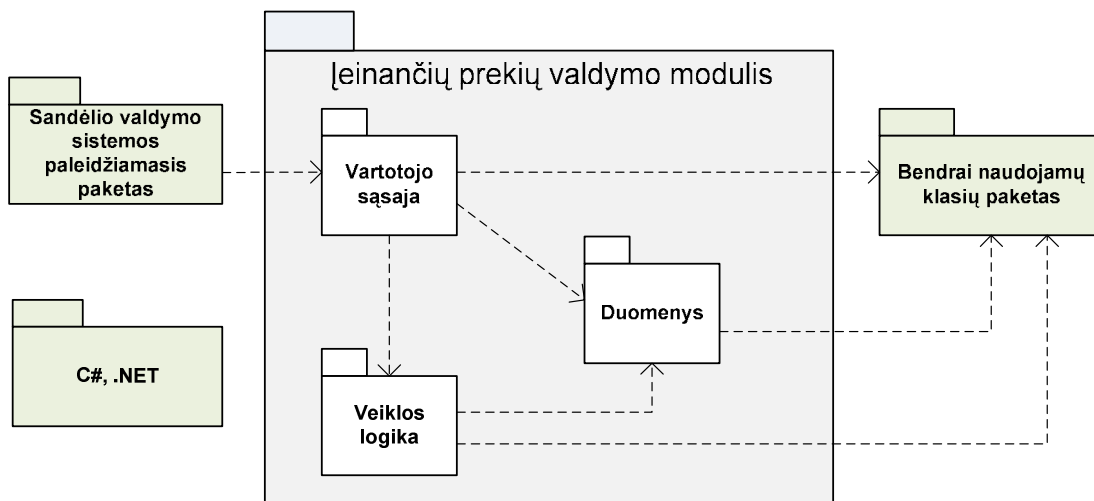
Veiklos analizės ir projektavimo metu buvo išanalizuoti ir aprašyti įeinančių prekių valdymo modulio apribojimai ir funkciniai reikalavimai, kurie įtakojo sistemos architektūrą ir jai keliamus reikalavimus:

1. Įeinančių prekių valdymo modulis bus kuriamas naudojant vieningą informacijos objektų ir jų savybių identifikavimo sistemą, kuri sudaro pagrindą duomenų bazių integravimui ir duomenų apsikeitimui. Todėl naudojant deklaratyvius duomenų bazės apribojimus bei mechanizmus, užtikrinančius sudėtingą vientisumo taisyklių realizavimą (pvz.: duomenų bazės procedūras ir t.t.), būtina tikrinti:
  - nuorodų tenkinimą (ang. *foreign key*);
  - įrašų identifikuojančių laukų unikalumą (ang. *primary/unique key*);

- leistinas įrašo laukų reikšmes;
  - atributų būtinumą.
2. Atliekant veiksmus su duomenų baze, turi būti kuo mažiau apkraunamas tinklas, kad kuo mažiau stabdytų tinklu keliaujančius kitus duomenis.
  3. Integracija su kitais sistemos moduliais turi vykti per atskiruose komponentuose aprašytas vartotojo sąsajas.
  4. Įeinančių prekių valdymo modulio vartotojo sąsaja turi būti paprasta. Joje turi būti nesunku atlikti visas reikiamas įeinančių prekių valdymo sandėlyje operacijas. Todėl vartotojo sąsajos langų klasės turi paveldėti atitinkamas bendras langų klases.

### 3.2.2 Įeinančių prekių valdymo modulio architektūrinė realizacija

Žemiau pateiktoje įeinančių prekių valdymo modulio paketų komunikacijos schemoje (žr. 6 pav.) matosi pagrindinių paketų sąveika.



6 pav. Įeinančių prekių valdymo modulio paketų komunikacijos schema

*Vartotojo sąsaja* - grafinės vartotojo sąsajos paketas turi visas ribines klases, kurios atvaizduoja programinės įrangos vaizdus, kuriuos mato vartotojas, bei jų pagalba komunikuoja su programine įranga. Šis paketas yra priklausomas nuo veiklos logikos, duomenų (duomenų bazės) bei bendrai naudojamų klasių paketo.

*Veiklos logika* – veiklos logikos pakete yra pagrindinės klasės, kurios vykdo įmonės verslo taisykles aprašytas panaudojimo atvejais. Šios klasės komunikuoja su duomenų baze ir

atlieka nuskaitymo, įrašymo, modifikavimo bei trynimo operacijas. Šis paketas yra priklausomas nuo duomenų bei bendrai naudojamo klasių paketo.

*Duomenys* - duomenų pakete yra klasės kurių pagalba vyksta informacijos mainai tarp kliento ir duomenų bazių serverio. Duomenų paketas yra priklausomas nuo bendrai naudojamo klasių paketo.

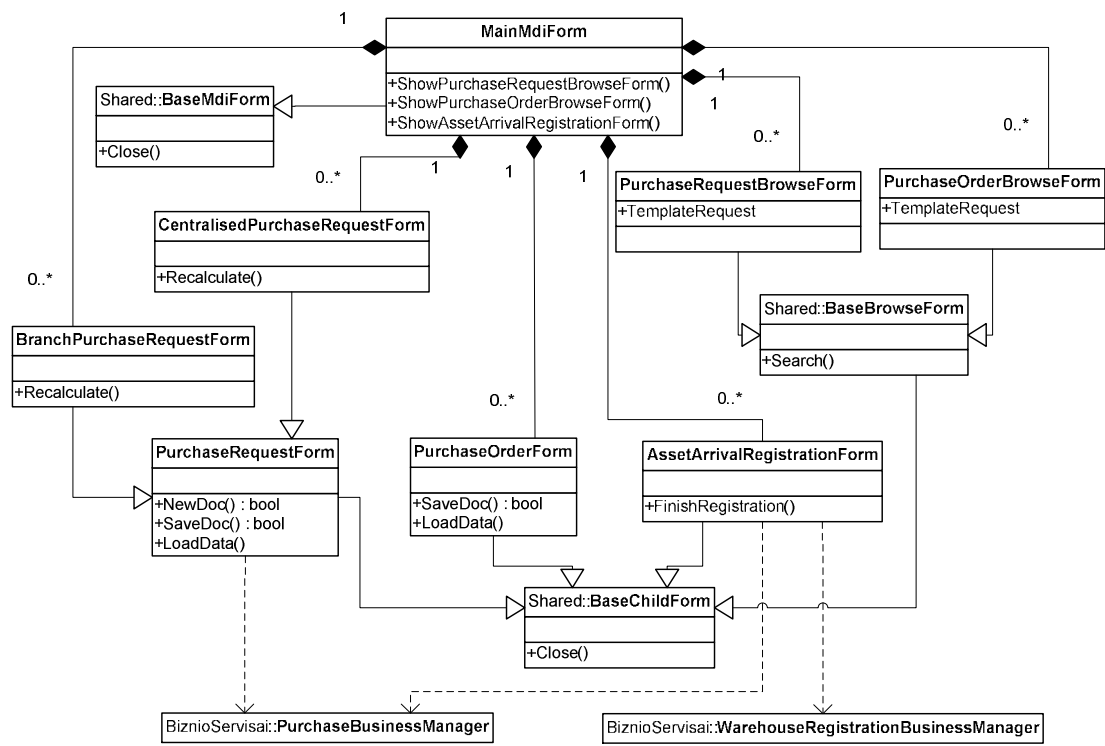
*Sandėlio valdymo sistemos paleidžiamasis paketas* - šiame pakete yra viena klasė, kuri nustato pradinis duomenis, bei inicijuoja pagrindinio sistemos lango paleidimą.

*Bendrai naudojamų klasių paketas* - šiame pakete yra klasės, kurios yra bendros visiems sistemos moduliams. Tai bazinės klasės, kurios vykdo pagrindinius veiksmus, jų pagalba visų modulių analogiški veiksmai yra vykdomi vienodai bei tie patys grafiniai elementai atrodo vienodai.

*C#, .NET paketas* - pakete yra klasės, kurių pagalba realizuota programinė įranga.

### 3.2.3 Vartotojų sąsajos klasių diagrama

Žemiau pateiktame paveikslėlyje yra pavaizduota vartotojų sąsajos klasių diagrama (žr. 7 pav.).



7 pav. Vartotojų sąsajos klasių diagrama

*Shared::BaseMdiForm* - ši klasė atspindinti bazinę pagrindinę sistemos formą, kurią kituose projektuose reikia paveldėti tam, kad būtų išlaikoma visų projektų vienoda vartotojo sąsaja bei vienodas funkcionalumas. Tokio tipo langas gali turėti daug kitų formų (bet ne MDI tipo).

*Shared::BaseChildForm* - ši klasė atspindinti bazinę paprastą formą, kurią kituose projektuose reikia paveldėti tam, kad būtų išlaikoma visų projektų vienoda vartotojo sąsaja bei vienodas funkcionalumas.

*Shared::BaseBrowseForm* - tai bazinė klasė objektų paieškos formoms, kurią reikia paveldėti tam, kad būtų išlaikoma visų projektų vienoda vartotojo sąsaja bei vienodas funkcionalumas.

*GUI::MainMdiForm* - pagrindinis sistemos langas. Šis langas yra MDI tipo, tad visi kiti sistemos langai bus atidaromi šio, pagrindinio, lango viduje. Šis langas turi pagrindinį meniu bei įrankių juostą. Ji yra paveldėta iš *Shared::BaseMdiForm* klasės.

*GUI::PurchaseRequestBrowseForm* - klasė skirta užsakymų paieškai atlikti bei paieškos rezultatams atvaizduoti. Ji paveldėta iš *Shared::BaseBrowseForm* klasės.

*GUI::PurchaseOrderBrowseForms* - klasė skirta pirkimų paieškai atlikti bei paieškos rezultatams atvaizduoti. Ji paveldėta iš *Shared::BaseBrowseForm* klasės.

*GUI::PurchaseRequestForm* - ši klasė yra skirta užsakymo objekto (*PurchaseRequest*) duomenims sukurti bei modifikuoti. Ji yra paveldėta iš *Shared::BaseChildForm* klasės.

*GUI::BranchPurchaseRequestForm* - ši klasė yra skirta vietinio (branch) tipo užsakymo duomenims sukurti bei modifikuoti. Ji yra paveldėta iš *GUI::PurchaseRequestForm* klasės.

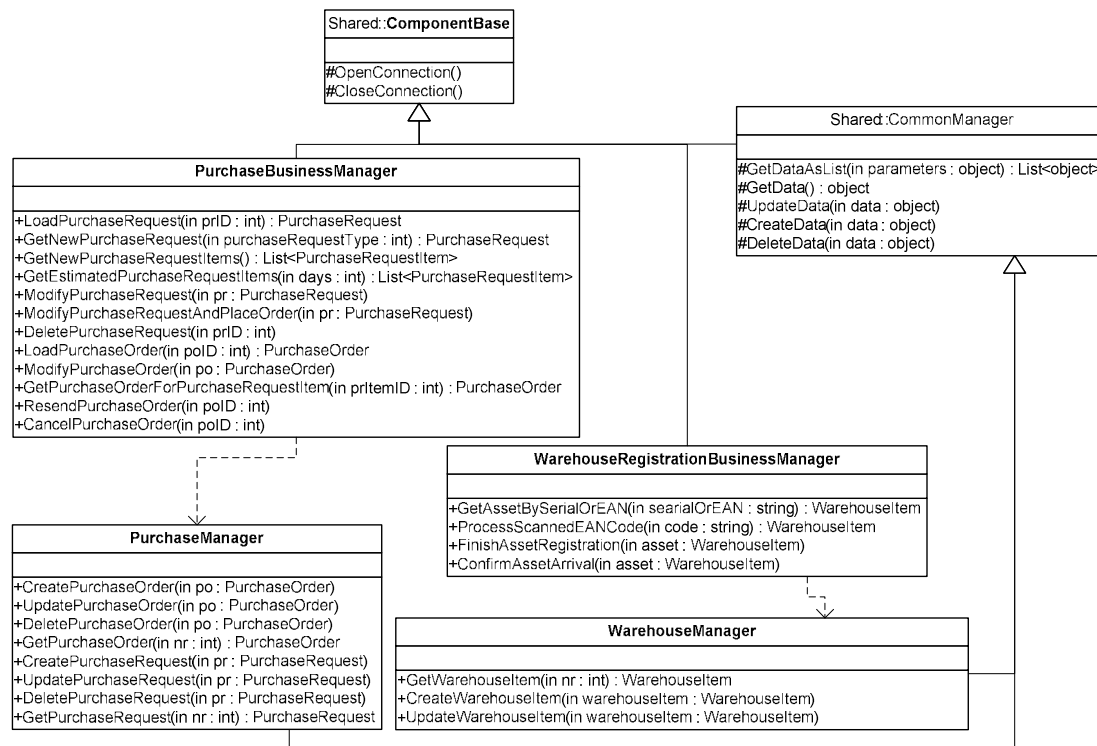
*GUI::CentralisedPurchaseRequestForm* - ši klasė yra skirta centralizuoto (centralised) tipo užsakymo duomenims sukurti bei modifikuoti. Ji yra paveldėta iš *GUI::PurchaseRequestForm* klasės.

*GUI::PurchaseOrderForm* - klasė yra skirta pirkimo objekto (*PurchaseOrder*) duomenims modifikuoti. Ji yra paveldėta iš *Shared::BaseChildForm* klasės.

*GUI::AssetArrivalRegistrationForm* - klasė skirta atvykusioms prekėms registruoti. Ji paveldėta iš *Shared::BaseChildForm* klasės.

### 3.2.4 Veiklos logikos klasių diagramos

Žemiau pateiktame paveikslėlyje matosi kaip veiklos logika yra aprašyta klasių diagramomis (žr. 8 pav.).



8 pav. Veiklos logikos klasių diagrama

*Shared::ComponentBase* – tai bazinė klasė skirta veiklos logikai aprašyti. Ją paveldi visos kitos veiklos logiką aprašančios klasės.

*Shared::CommonManager* - tai bazinė klasė skirta įvairiems duomenų bazėje saugomiems duomenims valdyti. Ją turėtų paveldėti visos klasės, kurios tiesiogiai skaito, rašo, keičia ar trina duomenis.

*PurchaseManager* - tai veiklos logikos klasė, kuri yra skirta užsakymų bei pirkimų duomenims sukurti, išsaugoti, modifikuoti bei trinti iš duomenų bazės. Ji paveldėta iš *Shared::CommonManager* klasės.

*WarehouseManager* - tai veiklos logikos klasė, kuri yra skirta sandėlio prekių įrašams duomenų bazėje kurti, saugoti bei modifikuoti. Ji paveldėta iš *Shared::CommonManager* klasės.

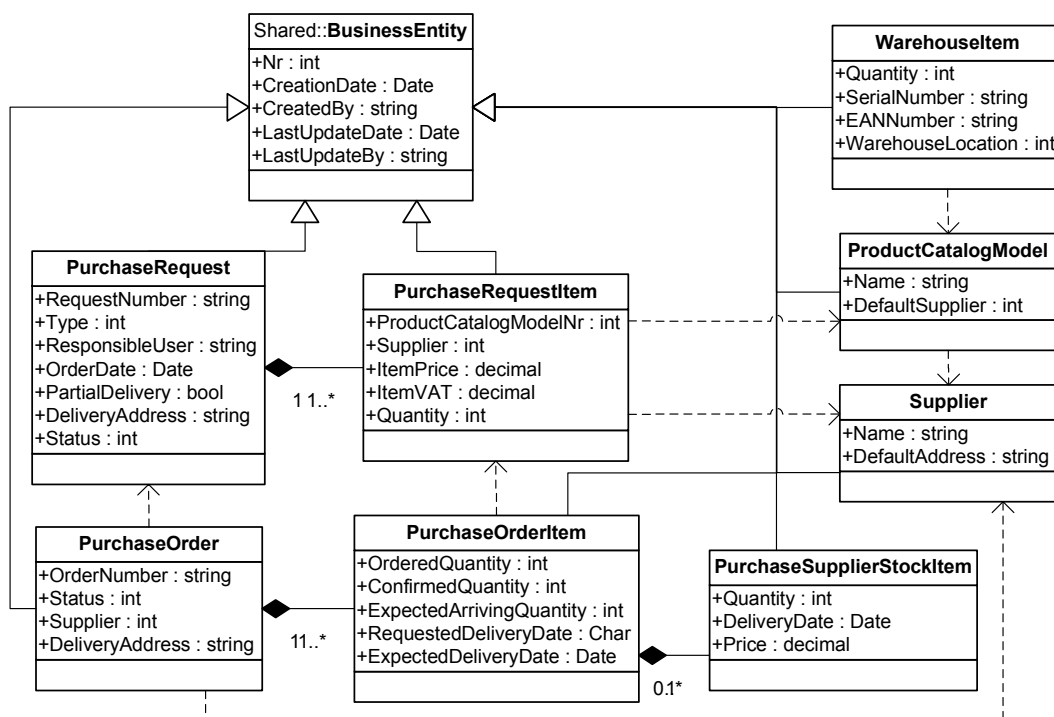


*PurchaseBusinessManager* - tai veiklos logikos klasė, kuri yra skirta operacijoms susijusioms su užsakymais bei pirkimais valdyti. Ji paveldėta iš *Shared::ComponentBase* klasės.

*WarehouseRegistrationBusinessManager* - tai veiklos logikos klasė, kuri yra skirta operacijoms susijusioms su atvykusių prekių registravimu valdyti. Ji paveldėta iš *Shared::ComponentBase* klasės.

### 3.2.5 Duomenų klasių ryšių diagrama

Žemiau pateiktame paveikslėlyje matosi duomenų bazėje esančių lentelių tarpusavio ryšiai (žr. 9 pav.).



9 pav. Duomenų klasių ryšių schema

*Shared::BusinessEntity* - klasė atspindi duomenų bazės įrašo bazinę klasę. Ši klasė skirta užtikrinti duomenų mainus tarp vartotojo sąsajos ir veiklos logikos paketo.

*ProductCatalogModel* - klasė atspindi prekės įrašą prekių duomenų bazės lentelėje. Ji yra paveldėta iš *Shared::BusinessEntity* klasės.

*PurchaseRequest* - klasė atspindi užsakymo įrašą užsakymų duomenų bazės lentelėje. Ji yra paveldėta iš *Shared::BusinessEntity* klasės.

*PurchaseRequestItem* - klasė atspindi užsakymo eilutės (užsakomos prekės) įrašą užsakomų prekių duomenų bazės lentelėje. Ji yra paveldėta iš *Shared::BusinessEntity* klasės.

*PurchaseOrder* - klasė atspindi pirkimo įrašą pirkimų duomenų bazės lentelėje. Ji yra paveldėta iš *Shared::BusinessEntity* klasės.

*PurchaseOrderItem* - klasė atspindi pirkimo eilutės įrašą pirkimo eilučių duomenų bazės lentelėje. Ji yra paveldėta iš *Shared::BusinessEntity* klasės.

*PurchaseSupplierStockItem* - klasė atspindi tiekėjo prekės pristatymo eilutės įrašą tiekėjo pristatomų prekių duomenų bazės lentelėje. Ji yra paveldėta iš *Shared::BusinessEntity* klasės.

*WarehouseItem* - klasė atspindi prekės vieneto sandėlyje įrašą prekių sandėlyje duomenų bazės lentelėje. Ji yra paveldėta iš *Shared::BusinessEntity* klasės

*Supplier* - klasė atspindi tiekėjo įrašą tiekėjų duomenų bazės lentelėje. Ji yra paveldėta iš *Shared::BusinessEntity* klasės.

### **3.3 Projektavimo išvados**

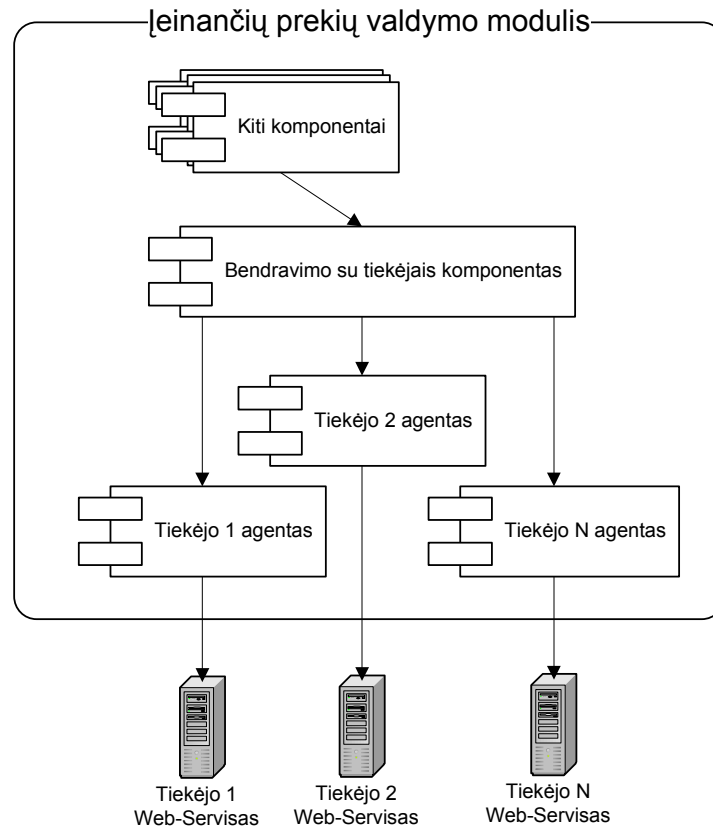
1. Surinktos ir aprašytos užsakovo verslo taisyklės funkciniais įeinančių prekių valdymo modulio reikalavimais.
2. Suprojektuota sistemos architektūra pagal funkcinis ir nefunkcinis reikalavimus.
3. Pagal kliento reikalavimus sukurtas įeinančių prekių valdymo modulis sandėlio valdymo sistemoje, kuris pagerino komunikavimą su tiekėjais, leisdamas automatiškai importuoti tiekėjų katalogus, formuoti bei siųsti užsakymus tiekėjams, stebėti užsakymų statusą bei priimti atvykusias prekes į sandėlį.

## **4. Tyrimo dalis**

### **4.1 Komunikacijos su tiekėjais pasirinkimas**

Išanalizavus sukurtą įeinančių prekių valdymo modulio architektūrą, buvo pastebėta, kad jos plečiamumas yra sudėtingas, jei atsirastų naujas tiekėjas, turintis savo išskirtinį informacinės sistemos modelį. Modulio palaikymas ir praplėtimas būtų sudėtingas dėl to, kad reikia modifikuoti programos kodą, kurti specifinį komponentą, kuris galėtų bendrauti su nauja tiekėjo informacine sistema, priimant bei siunčiant specialiai šiai sistemai suformuotus

duomenis. Žemiau pateiktoje schemoje matosi įeinančių prekių valdymo modulio ir tiekėjų sąsaja (žr. 10 pav.).



*10 pav. Įeinančių prekių valdymo modulio ir tiekėjų sąsajos schema*

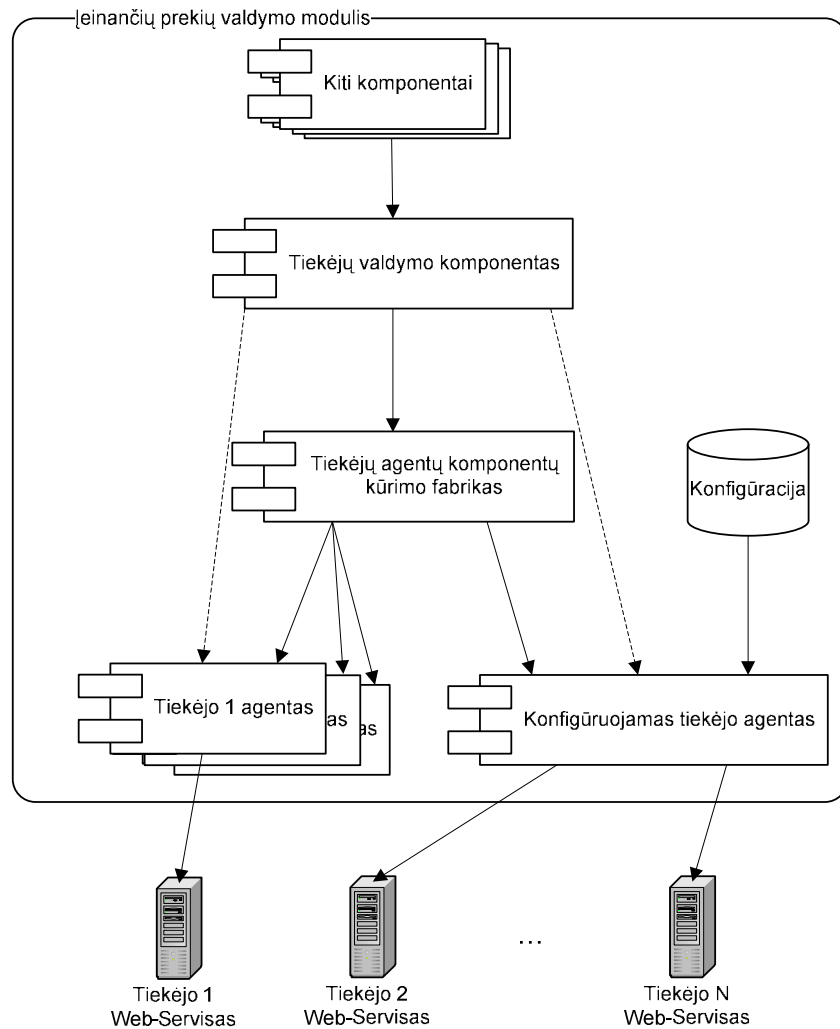
Iš schemos matyti, kad norint pridėti naują komunikavimo su papildomu tiekėju funkcionalumą, reikia papildyti sandėlio valdymo modulį, o po to jį iš naujo įdiegti į kliento kompiuterius. Taip pat modulis visada turi baigtinį ir maksimalų palaikomų tiekėjų skaičių.

Žemiau esančiuose skyreliuose yra pateikiama galimo sprendimo įgyvendinimas ir pasirinkto sprendimo apibendrinimas.

#### **4.1.1 Išskaidyta komunikacija su tiekėjais**

Kad sistemos palaikomus tiekėjus būtų galima lengviau kurti bei komplektuoti, reikia konkrečius tiekėjų komponentus atsieti nuo pačio įeinančių prekių valdymo modulio. Todėl reikia išskirti bendrą sąsają, kurią turi palaikyti kiekvienas įeinančių prekių valdymo modulyje egzistuojantis tiekėjo agento komponentas bei turės palaikyti visi būsimi tiekėjų agentų komponentai.

Žemiau esančioje schemoje yra pavaizduota įeinančių prekių valdymo modulio ir tiekėjų siūloma sąsaja (žr. 11 pav.).



**11 pav.** Įeinančių prekių valdymo modulio ir tiekėjų siūloma(1) sąsajos schema

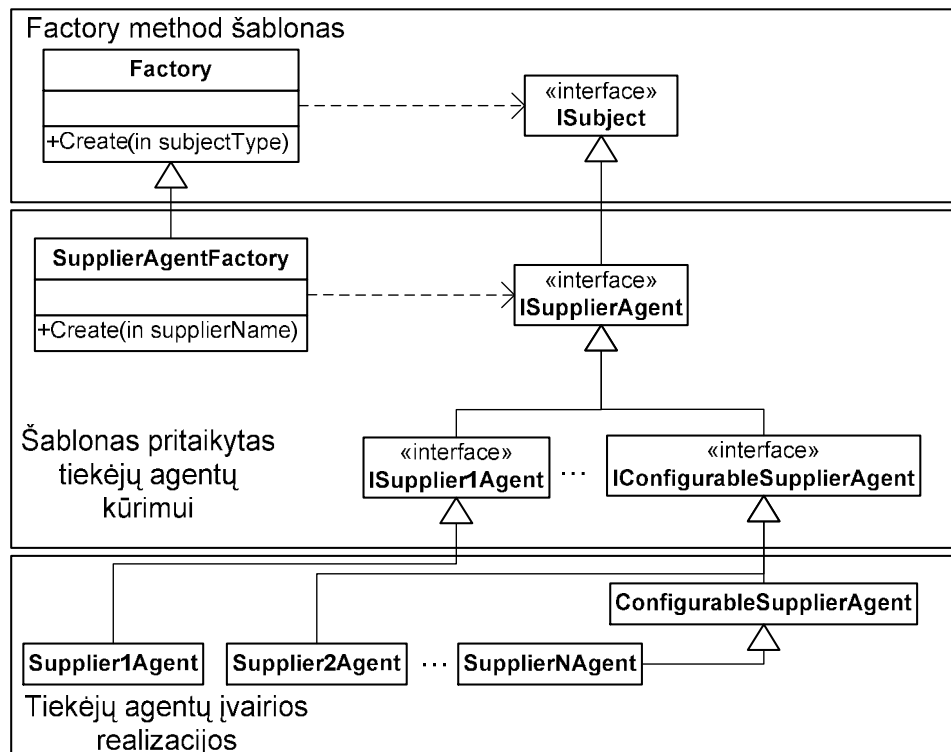
Patobulintos architektūros privalumai ir trūkumai:

- + Sumažėjęs ryšys tarp tiekėjų valdymo komponento ir tiekėjų agentų komponentų;
- + Kuriant naują projektą komponentus galima lengviau komplektuoti;
- + Aiški sąsaja tarp komponentų ir jų vartotojų;
- + Yra centralizuota tiekėjų konfigūracija saugoma duomenų bazėje;
- - Dėl suvienodintos tiekėjų informacinių sistemų sąsajos, dalies tiekėjų papildomas funkcionalumas (pvz., užsakymų atšaukimas) tapo nebe prieinamas;

Žemiau esančiuose skyriuose padetalizuosime įtrauktus papildomus architektūrinius elementus, tokius kaip šablono *Factory method* pritaikymą tiekėjų agentų komponentams bei konfigūruojamą tiekėjo agentą. Taip pat pademonstruosime paskutiniame punkte aprašyto patobulintos architektūros trūkumo pašalinimą, leisdami tiekėjų agentų komponentams turėti papildomą sąsają ar jų aibę.

#### 4.1.2 Projektavimo šablono „Factory method“ panaudojimas

Įeinančių prekių valdymo modulyje, bendravimo su tiekėjais posistemėje, panaudosime projektavimo šablono *Factory method* [17]. Naudojant šį šabloną galima atskirti bendrą modulyje funkcionalumą nuo konkretaus tiekėjo informacijos srautų valdymo. Žemiau pavaizduotame paveikslėlyje yra pademonstruotas *Factory method* projektavimo šablono panaudojimas (žr. 12 pav.).



12 pav. Factory method panaudojimas

*Factory method* projektavimo šablonas susideda iš dviejų klasių. Pirmoji klasė yra *Factory* klasė, turinti tik vieną metodą (*Create* – sukurti), kuris sukuria antrosios klasės tipo objektą. *Create* metodui yra perduodamas objektas, kuris turi būti sukurtas. Pats objektas gali būti bet kokio tipo, paveldėto iš *ISubject* sąsajos (ang. *interface*).

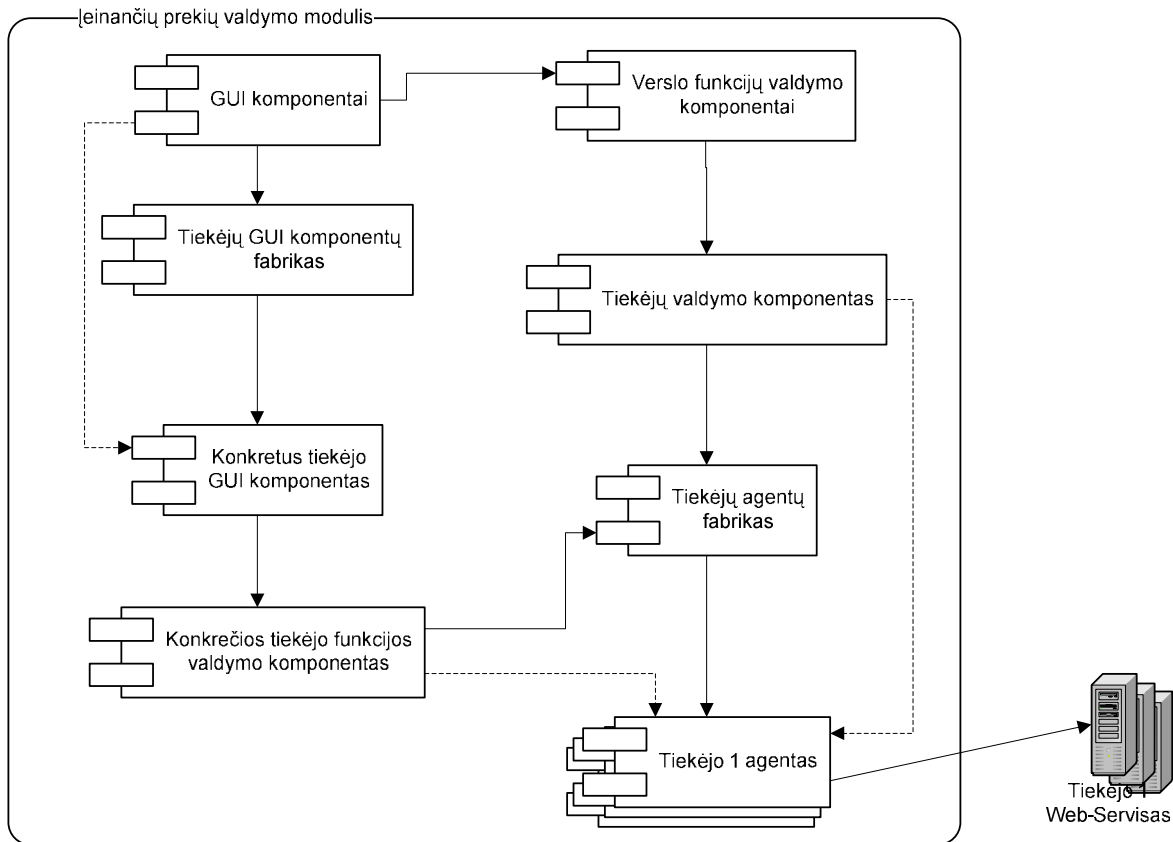
Panaudotas projektavimo šablonas pagal tiekėjo pavadinimą sukuria konkrečią tiekėjo agento komponento realizaciją. Su tiekėjo agento komponentu bendraujama per tam skirtos sąsajos (*ISupplierAgent*) metodus. Naudojant *Factory method* projektavimo šabloną yra paslepiamas konkrečių tiekėjo agentų komponentų sukūrimas. Mums nėra įdomus pats kūrimo procesas, svarbiausia – rezultatas. Tai yra viena iš objektinio programavimo savybių.

### **4.1.3 Konfigūruojamas ryšio su tiekėjais komponentas**

Sąsaja su visais modulyje palaikomais tiekėjais realizuota XML Web Servisų principu. Ši technologija internete yra labai paplitusi, todėl tikėtina, kad nauji tiekėjai irgi leis prieiti prie savo informacinių sistemų per XML Web Servisus. Apibendrinant ryšio su tiekėju komponento atliekamą darbą, galima teigti, kad jis atlieka tokią funkciją: keičia duomenų struktūrą iš tiekėjo duomenų struktūros į sistemos vidinę duomenų struktūrą. Aprašius sistemos vidinę duomenų struktūrą XML pagrindu, galima daryti keitimą iš vieno tipo XML į kito tipo XML struktūrą. Tokiam keitimui atlikti galima pasinaudoti egzistuojančia technologija – XSLT transformacijomis. Transformacijų aprašus galima lengvai saugoti duomenų bazėje ir keisti kada tik prireikus. Tokiu būdu, kad į sistemą įvesti papildomą tiekėją užtenka parašyti ir į duomenų bazę suvesti naują XSLT transformaciją.

### **4.1.4 Papildomų tiekėjų funkcijų panaudojimas**

4.1.1 skyrelyje aprašyta architektūra remiasi tuo, kad visi tiekėjai palaikys tik standartines įeinančių prekių moduliui reikalingas funkcijas. Tačiau yra tokių tiekėjų, kurie per Web Servisus leidžia atlikti ir papildomas funkcijas (pvz.: leidžia išsiūtą užsakymą atšaukti; importuoti užsakymų statusus ir t.t.). Tokios funkcijos dažniausiai būna prieinamos tik tiesiogiai bendraujant su tiekėju (pvz.: telefonu, susitarimo principu ir t.t.). Žemiau esančiame paveikslėlyje pavaizduota patobulinta architektūra leidžianti išnaudoti tokias papildomas tiekėjų funkcijas (žr. 13 pav.).



**13 pav.** Įeinančių prekių valdymo modulio ir tiekėjų siūloma(2) sąsajos schema

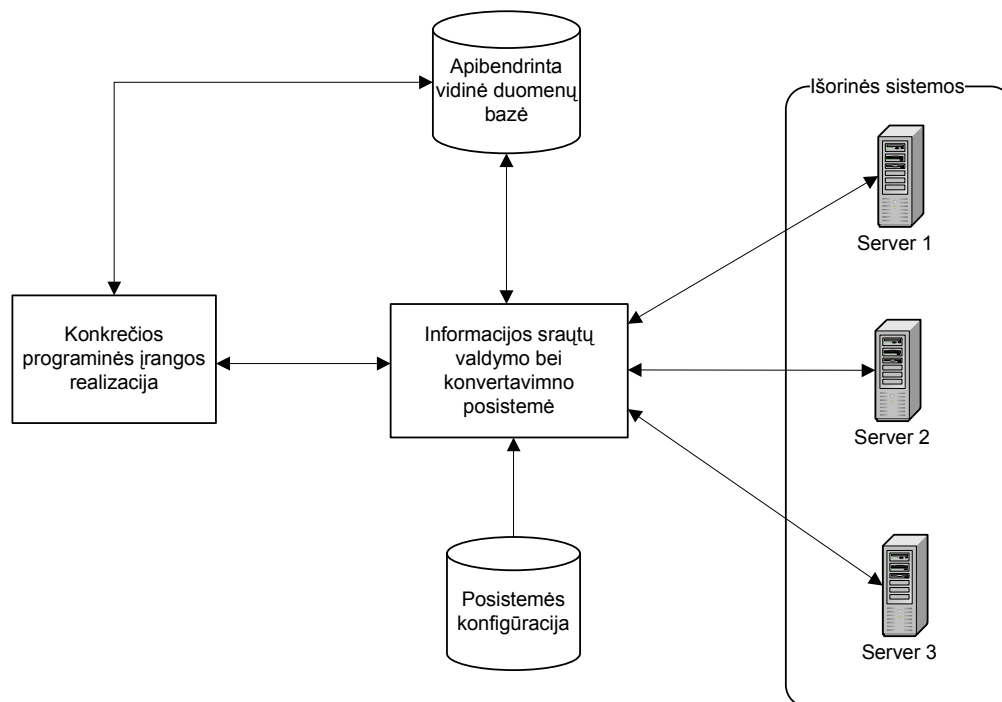
Kai tiekėjas palaiko daugiau papildomų funkcijų, kurios nėra aprašytos standartinėje sąsajoje, šios funkcijos turi būti apibendrintos bei joms sukurta papildoma sąsaja, kurią tiekėjo agento komponentas taip pat turės palaikyti.

Tokia architektūra leidžia turėti daugeliui tinkamus duomenų apsikeitimo būdus su išoriniais sistemos komponentais, kurie gali atlikti ne tik standartinės funkcijas, bet ir papildomas. Tai yra realizuotas tik vienoje konkrečioje ar keliose išorinėse sistemose. Šios architektūros pagalba galima naudotis papildomu funkcionalumu, kuris yra susijęs su vienu ar kitu tiekėju. Pavyzdžiui, Tiekėjas1 leidžia atšaukti užsakymus, todėl vartotojo sąsajoje atsiras papildomas funkcionalumas, leidžiantis atšaukti užsakymą (pvz.: mygtukas „Atšaukti“).

#### 4.1.5 Apibendrintas išorinių sistemų duomenų valdymo modelis

Apibendrinant visą uždavinį galima teigti, kad tiekėjų informacinės sistemos yra paprasčiausias informacijos šaltinis, o įeinančių prekių valdymo modulio duomenų bazė yra tik vienas įmanomas modelis iš tiekėjų (informacijos šaltinių) gaunamiems duomenims saugoti. bei siūsti atgal.

Žemiau esančiame paveikslėlyje yra pateikiama informacijos šaltinių ir srautų schema(žr. 14 pav.).



14 pav. Informacijos šaltinių ir srautų schema

Taigi šį sprendimą galima taikyti ne vien ryšiui su tiekėjais palaikyti tačiau ir įvairiuose uždaviniuose, kaip pavyzdžiui:

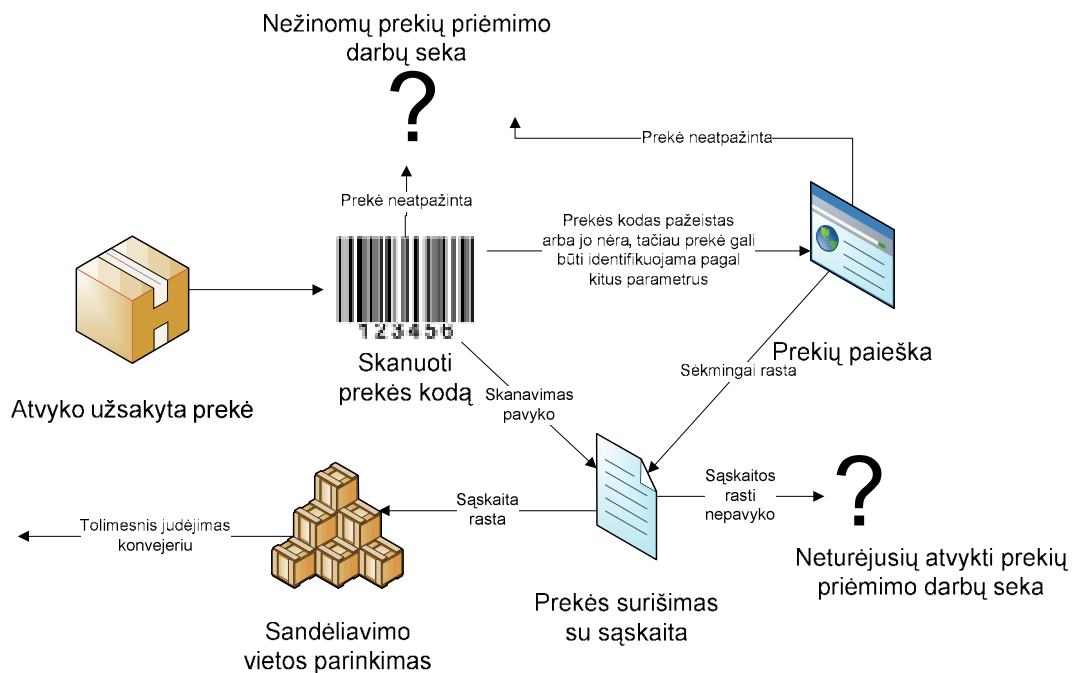
- adresų duomenų bazė, imanti adresus iš įvairių šalių viešai prieinamų šaltinių;
- skirtingų šalių bankų kodų informacija;
- įvairių oro uostų informacija;
- orų prognozių statistikos pateikimas iš skirtingų šaltinių;
- kelių internetinių parduotuvių aptarnavimas per vieningą sąsają.



## 4.2 Prekių priėmimo proceso galimybių padidinimas

Sandėlio valdymo sistema kompiuterizuoja daug įmonės vidaus procesų įvairioms užduotims vykdyti, tokioms kaip užsakymų sudarymas ir pateikimas, kainų maržos koregavimas atsižvelgiant į užsakomus iš tiekėjų kiekius bei tiekėjų siūlomas mažesnes kainas šiems kiekiams ir t.t. Visi šie procesai, įeinančių prekių valdymo modulyje, buvo gerai specifikuoti ir realizuoti pagal užsakovo pateiktas verslo taisykles. Tačiau pasitaiko, kad procesai keičiasi ir įmonė persitvarko savo gamybos procesą (siekdama pagerinti produkto kokybę, padidinti pelną ir t.t.) ir taip pasikeičia senosios verslo taisyklės kitomis - naujesnėmis. Tokiu atveju sukurta ir pritaikyta pagal senas verslo taisykles sistema jau tampa nebetinkama arba nebe tokia naudinga kaip seniau.

Daugelį įmonės vidinių procesų galima aprašyti darbų sekomis (ang. *workflow*). Darbų sekos perteikia šiuos nepriklausomus procesus dinamiškame ir tvirtame pavidale. Kai įmonės verslo taisyklės dažnai keičiasi ir tuo pačiu dažnai keičiasi funkciniai reikalavimai – tikslinga naudoti darbų sekas. Pavyzdžiui, įeinančių prekių valdymo modulio prekių priėmimo posistemės realizacija gali būti nesudėtinga, jei yra pateikti aiškūs ir nesikeičiantys reikalavimai. Bet jei reikia realizuoti daugiau pasirenkamų galimybių (pvz.: prekės vietos sandėlyje parinkimo nustatymus, prekes identifikuojančių lipdukų spausdinimo parametrus), procesą galima būtų patobulinti realizuojant jį darbų sekomis (žr. 15 pav.).



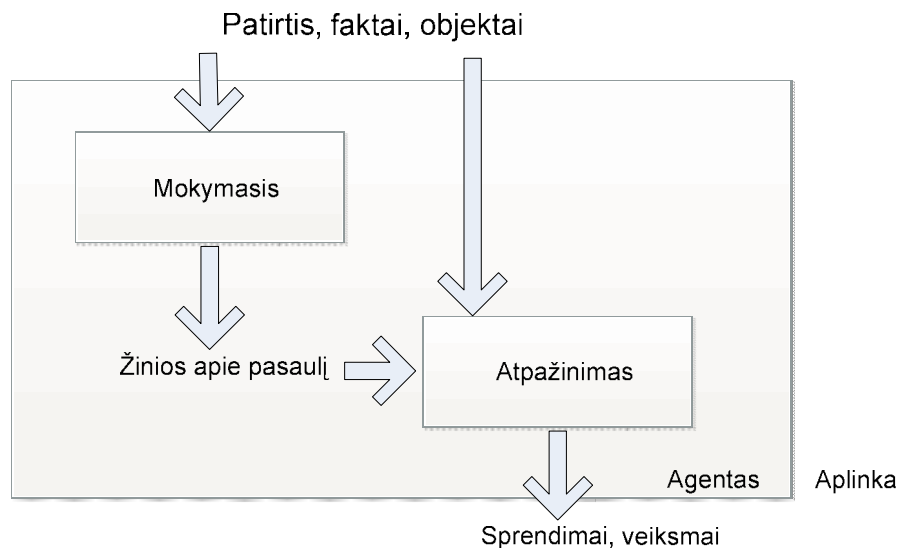
15 pav. Darbų sekų pritaikymas įeinančių prekių valdymo modulyje

Kai įmonės verslo taisyklės nežymiai ir retai keičiasi, tai nėra tikslinga naudoti darbų sekas. Visų pirma, darbų sekų projektavimo priemonės yra brangi investicija, nes neužtenka įsigyti darbų sekų projektavimo ir realizavimo įrankius (pvz.: MS Workflow Foundation), dažnai tenka papildomai pirkti darbų sekų veiklos įvykių bibliotekas, kurios yra pritaikytos vienoms ar kitoms technologijoms (pvz. MS Office biblioteka, MS SharePoint biblioteka ir t.t.). Antra, trūksta specialistų, kurie gerai išmanytų darbų sekų pritaikymą ir panaudojimą realiuose projektuose.

Pasvėrus visus plusus ir minusus įeinančių prekių valdymo modulyje nebuvo naudinga naudoti darbų sekas, nes užsakovo suformuluoti reikalavimai nesikeitė, tinkami įrankiai darbų sekoms kurti nebuvo nupirkti ir nebuvo laiko ieškoti specialistų ar apmokyti darbuotojus dirbti su darbų sekomis. Todėl įeinančių prekių valdymo modulis buvo sukurtas nenaudojant darbų sekų.

### 4.3 Įeinančių prekių valdymo modulis užsakymo prognozavimo mokymosi schema

Apie sistemos intelektualumą galima kalbėti tik tada, jei ji yra adaptyvi. Kad sistema būtų adaptyvi, ji turi sugebėti mokytis iš savo patirties. Taigi, reikia sukurti pakankamai realistišką kompiuterinį mokymosi modelį (žr.16 pav.).



16 pav. Įeinančių prekių valdymo modulis užsakymo mokymosi schema

Mokymosi metu sukurtos žinios bus saugomos skaitmeniniame pavidale: skaičių, svorių, koeficientų rinkinių ir pan.

Įeinančių prekių valdymo modulio užsakymo prognozavimas turi būti realizuotas pagal aukščiau pavaizduotą mokymosi schemą. Modulis (schemoje jis vadinamas agentu, nes kalbame apie intelektualią sistemą) surinks iš aplinkos patirtį ir faktus, kurios naudos mokymuisi. Taigi, svarbu, kad duomenys: apie ankstesnių sezonų pardavimus (ketvirčius), likučius lokaliuose sandėliuose, vykdomas akcijas ir ankstesnes analogiškų vykdomų akcijų patirtis bei bendrus prekių kiekius įmonės filialų tinkle, - būtų kaupiami kaip patirtis iš kurios sistema mokosi. Pavyzdžiui, kad atskirti, kurios prekės yra akcijnės, bus pridėtas papildomas laukas, kuris tai ir nurodys. Veiksmai, kurios modulis turi atlikti turėdamas žinias apie aplinką bus valdomi per tam tikras aprašytas sąlygas. Pavyzdžiui, IV ketvirtį buvo parduota 900 X prekių. Sąlyga: jei lokaliuose sandėliuose X prekės mažiau nei 100 vienetų ir pardavimų skaičius išaugo 40% per paskutinį ketvirtį lyginant su prieš tai esančiu ketvirčiu, tai X prekių užsakyti:  $900+900*40\%=1260$ .

Prekių prognozavimas modulyje nėra realizuotas, kadangi su užsakovu nėra suderintos tikslios prognozavimo sąlygos. Tačiau prekių prognozavimui yra sudaryta kūrimo metodika.

## **5. Pasirinkto sprendimo spręsti komunikaciją su tiekėjais įvertinimas**

Šiame skyriuje bus įvertinamas įeinančių prekių valdymo modelio tiekėjų komunikacijos pridėjimo architektūros pasirinkimas.

### **5.1 Aplinkos įvertinimas**

Projektuojant sandėlio valdymo programinę įrangą, o kartu su ja ir įeinančių prekių modulį naujam klientui, sunku prognozuoti ar reikės naujų tiekėjų pridėjimo ateityje. Tuo labiau nėra aišku, ar priimtas sprendimas numatyti nesunkų tiekėjų pridėjimą, nepakenks projekto vykdymui dar labiau. Šiuo atveju, įeinančių prekių valdymo modulis buvo realizuotas pagal specifikacijoje nurodytus reikalavimus. Todėl jei būtų nesilaikoma specifikacijoje nurodytų reikalavimų ir realizuotas papildomos galimybės, gali būti tokios neigiamos pasekmės:

- Dėl *sudėtingesnio architektūrinio sprendimo* realizacijos laikas pailgėja.
- Dėl *pailgėjusio sprendimo realizacijos laiko* didėja rizika laiku nepabaigti projektą.
- Dėl *laiku nepabaigto projekto* leidžiami pinigai iš modulio kūrėjo kišenės.

- Dėl specifikacijos *nesilaikymo* gali tekti įeinančių prekių valdymo modulį perdaryti du kartus (vieną kartą tobulinant ir ieškant geresnio varianto, antrą – suteiktos garantijos metu klientas gali pareikalauti, kad sistemą sutvarkytų pagal specifikaciją).
- Dėl specifikacijos *pasikeitimų* kūrimo metu gali tekti įeinančių prekių valdymo modulį perdaryti du kartus (vieną kartą tobulinant ir ieškant geresnio varianto, antrą – pagal naujus reikalavimus).
- Dėl kodo pakeitimo didėja klaidų atsiradimo galimybė.

Pasvėrus visas aukščiau išvardintas rizikas, galima teigti, kad įeinančių prekių valdymo modulio „pagerinimas“ palengvinant tiekėjų pridėjimą, nebūtų teisingas pasirinkimas. Todėl pasirinktas sprendimas buvo tinkamiausias.

## 5.2 Darbo laiko ir kodo eilučių įvertinimas

Paprasto naujų tiekėjų pridėjimo sprendimas gali būti panaudojamas tik tada, kai tiksliai žinotume (specifikacijoje būtų parašyta), kad ateityje reikės dažnai pridėti naujus tiekėjus arba nurodytas didelis tiekėjų pridėjimo skaičius.

Žemiau pateiktoje lentelėje yra pateikta kiek reikia kodo eilučių ir laiko įgyvendinant tiek esamą sprendimą, tiek paprastą tiekėjų pridėjimą (žr. 2 lentelė).

**2. lentelė. Kodo eilučių ir laiko įvertinimas esamam sprendimui ir paprastam tiekėjo pridėjimo sprendimui**

Sprendimas	Pradinės kodo bazės dydis	Kodo eilutės/1 tiekėjas	Laikas/Pradinės kodo bazės sukūrimas	Laikas/1 tiekėjas	Pastabos
Esamas sprendimas	500	1500	2 dienos	7 dienos	-
Parastas naujų tiekėjų pridėjimo sprendimas	10 000	500	2 mėnesiai	2 dienų	Gali tekti naudoti papildomas programas

Lentelėje laikas yra įvertinamas, kai viskas sekasi ir nėra trukdžių.

Iš lentelės matome, kad paprastam naujo tiekėjo pridėjimo sprendimui pasiruošti pradinę kodo bazę užtrunka 2 mėnesius. Tai reiškia, kad 40 dienų bus kuriamas tik pradinis kodas, bet

tai yra gan ilgas laikotarpis. Todėl tokiam sprendimui priimti turi būti iš anksto planuojami terminai, derinama su užsakovu, keičiama specifikacija ir t.t. Pridėjus visus kitus darbus, kuriuos įtakoja toks sprendimas gaunasi, kad dar labiau prailgėja projekto terminai.

Panagrinėkime, kada mums apsimoka keisti esamą sprendimą į paprastą naujų tiekėjų pridėjimo sprendimą. Paskaičiavus kodo eilučių ir laiko sąnaudas 6 tiekėjams (įeinančių prekių valdymo modulis šiuo metu palaiko sąsają su 6 tiekėjais), žemiau pateiktoje lentelėje gauname rezultatus (žr. 3 lentelė).

**3. lentelė. Kodo eilučių ir laiko įvertinimas esamam sprendimui ir paprastam tiekėjo pridėjimo sprendimui, kai tiekėjų skaičius lygus 6**

<b>Sprendimas</b>	<b>Viso kodo eilutės</b>	<b>Laikas</b>
<b>Esamas sprendimas</b>	9500	44
<b>Paprastas naujų tiekėjų pridėjimo sprendimas</b>	10300	52

Iš aukščiau esančios lentelės duomenų, matome, kad paprasto naujų tiekėjų pridėjimo sprendimas savo laiko sąnaudomis gan žymiai skiriasi nuo esamo sprendimo. Todėl kai įeinančių prekių valdymo modulyje reikalinga palaikyti 6 tiekėjus, tai galim drąsiai rinktis esamą sprendimą.

Toliau panagrinėkime kada paprasto naujų tiekėjų sprendimo realizavimas būtų tinkamas pasirinkimas (įvertinant tik kodo eilučių kiekį ir laiko sąnaudas). Žemiau pateiktoje lentelėje yra pateikti duomenys, kaip keičiasi kodo eilučių bendras kiekis didinant tiekėjų skaičių.

**4. lentelė. Kodo eilučių kaita didinant tiekėjų skaičių**

<b>Kodo eilutės, kai X tiekėjų skaičius</b>	<b>Esamas sprendimas</b>	<b>Paprastas naujų tiekėjų pridėjimo sprendimas</b>
6	9500	13000
7	11000	13500
8	12500	14000

9	14000	14500
10	15500	15000
11	17000	15500
12	18500	16000

Iš lentelės matome, kad jei reikėtų pridėti 9 tiekėjus, o ne 6, galima būtų galvoti apie kitą sprendimo pasirinkimą. O jau nuo 10 tiekėjų greičiausiai rinktumėmės paprastą tiekėjų pridėjimą. Grafinis kodo eilučių pasiskirstymas pavaizduotas A Priede.

### 5.3 Konfigūruojamo komponento efektyvumo įvertinimas

Naujos architektūros komponento efektyvumui patikrinti buvo sugeneruota užsakymų aibė. Kad skaičiavimai būtų lengvesni, užsakymai buvo pakartotinai siunčiami apdoroti dviem skirtingom programom. Pirmoji programa buvo paremta pirmine sistemos architektūra, o antroji realizuota su konfigūruojamu tiekėjo agentu. Tiekėjo agentas buvo sukonfigūruotas pagal vieno iš tiekėjų duomenų modelį taip, kad abiejų programų galutiniai suformuoti XML duomenys būtų vienodi.

Buvo įvertintas abiejų programų duomenų pakeitimui skirtas laikas (žr. C Priedas). Eksperimentas parodė, kad konfigūruojamas komponentas duomenis importuoja lėčiau nei statiška parašytas konkrečiam tiekėjui skirtas duomenų valdymo komponentas. Duomenų apdorojimo operaciją atliekant po 1000 kartų, pasirinktos architektūros komponentas darbą atlieka beveik 1,5 sekundėmis greičiau nei konfigūruojamas tiekėjo agento komponentas. Todėl tai tik dar kartą įrodo, kad pasirinktas sprendimas yra ne tik geras išanalizavus aplinką, bet ir technologiškai greitesnis.

Nors ir konfigūruojamas komponentas yra lėtesnis, tačiau jo galimybės yra didesnės – pridėti papildomą tiekėją į sistemą yra paprasčiau – nereikia keisti egzistuojančios kodo bazės, o be to nebūtina pakartotinai įdiegti sistemą pas jos vartotojus.

### 5.4 Grafinės vartotojo sąsajos efektyvumo įvertinimas

Grafiinei vartotojo sąsajai kurti buvo panaudotas *Factory method* projektavimo šablonas, kuris GUI komponentą gražina pagal tiekėjo informaciją. Todėl architektūra tampa kur kas

patogesnė ir labiau išskaidyta – daugiau programinės įrangos kūrėjų gali dirbti vienu metu su skirtingų tiekėjų komponentų realizacijomis.

Kad įvertinti naujos grafinės vartotojo sąsajos architektūros efektyvumą, buvo imituojamas darbas su skirtingas funkcijas palaikančiais tiekėjais ir įvertinamas laikas per kurį grafinė vartotojo sąsaja buvo pilnai „įkrauta“ bei atvaizduota.

Iš eksperimento rezultato (žr. D Priedas) matome jog patobulinta grafinė vartotojo sąsaja yra kiek lėtesnė nei esamoji. Tačiau vidutinis patobulintos grafinės vartotojo sąsajos efektyvumo sumažėjimas nėra didelis – 40 milisekundžių. Didesnis sulėtėjimas gali būti pastebėtas tik patį pirmą kartą įkraunant konkretaus tiekėjo valdymo komponentą. Taip yra todėl, kad pirmą kartą atvaizduojant šį komponentą, jis iš programinio kodo bylos įkraunamas į atmintį. Nuskaitymas iš bylos visada lėtesnis nei skaitymas iš operatyviosios atminties.

## 6. Išvados

1. Rinkos analizė parodė, kad įmonėms nepakanka standartinių įeinančių prekių valdymo sprendimų. Taigi kurti vartotojo poreikiams pritaikytą sprendimą verslo požiūriu yra racionalu.
2. Literatūros šaltinių analizės metu nustatyta, kad kuriamas įeinančių prekių valdymo modulis turi visas pagrindines standartines įeinančių prekių valdymo funkcijas lyginant su etaloninėmis sandėlio valdymo sistemomis. Todėl kurti tokią sistemą yra geras pasirinkimas, nes tokia sistema savo funkcionalumu gali drąsiai konkuruoti su jau seniai rinkoje egzistuojančiomis sistemomis. Dėl šių priežasčių pasirinktas sprendimas – sukurti įeinančių prekių valdymo modulį.
3. Pagal kliento reikalavimus sukurtas įeinančių prekių modulis sandėlio valdymo sistemoje, kuris pagerino komunikavimą su tiekėjais, leisdamas automatiškai importuoti tiekėjų katalogus, formuoti bei siųsti užsakymus tiekėjams, stebėti užsakymų statusą bei priimti atvykusias prekes į sandėlį.
4. Tyrimo metu nustatyta, kad įeinančių prekių valdymo moduliui nėra tikslinga naudoti darbų sekas, nes darbų sekų projektavimo priemonės yra brangi investicija, trūksta specialistų, kurie gerai išmanytų darbų sekų pritaikymą ir panaudojimą realiuose projektuose. Todėl kuriant įeinančių prekių valdymo modulį buvo pasirinkta nenaudoti darbų sekų.

5. Pasvėrus visas rizikas (1. Dėl *sudėtingesnio architektūrinio sprendimo* realizacijos laikas pailgėja. 2. Dėl *pailgėjusio sprendimo realizacijos laiko* didėja rizika laiku nepabaigti projektą. 3. Dėl *laiku nepabaigto projekto* leidžiami pinigai iš modulio kūrėjo kišenės. 4. Dėl specifikacijos *nesilaikymo* gali tekti įeinančių prekių valdymo modulį perdaryti du kartus. 5. Dėl specifikacijos *pasikeitimų* kūrimo metu gali tekti įeinančių prekių valdymo modulį perdaryti du kartus. 6. Dėl kodo pakeitimo didėja klaidų atsiradimo galimybė.) buvo priimtas sprendimas – nepraplėsti įeinančių prekių valdymo modulio galimybių realizuojant paprastą naujų tiekėjų pridėjimo architektūrą.
6. Eksperimentinio tyrimo metu buvo nustatyta, kad paprasto naujų tiekėjų pridėjimo sprendimas gali būti panaudojamas tik tada, kai tiekėjų skaičius yra 10. Dabartiniam sprendimui, kai tiekėjų skaičius yra 6, netikslinga realizuoti paprasto naujų tiekėjų pridėjimo sprendimo, nes laiko atžvilgiu esamas sprendimas įgyvendinamas mažiausiai 8 dienomis greičiau.
7. Sukurtas įeinančių prekių valdymo modulis sėkmingai įdiegtas ir vartojamas užsakovo įmonėje (perdavimo aktas pateiktas prieduose).



## **7. Terminų ir santrumpų žodynas**

**EDI** – Elektroninio duomenų apsikeitimo standartas (Electronic Data Interchange)

**UML** – grafinė programų sistemų modeliavimo kalba (Unified Modeling Language)

**SQL** – struktūrizuota užklausų kalba naudojama duomenų bazėse (Structured Query Language)

**CRM** – Verslo ryšių valdymas (Customer Relationship Management)

**RFID** – Radijo signalų atpažinimo technologija (Radio Frequency Identification)

**VPN** – Virtualus privatus tinklas (Virtual Private Network)

**IPSec** – IP apsaugos standartas (IP Security)

**3DES** – duomenų šifravimo algoritmas (Triple DES)

**PGP** – Interneto saugumo protokolas (Pretty Good Privacy)

**S/MIME** – Elektroninių žinučių siunčiamų Internetu kodavimo standartas (Secure / Multipurpose Internet Mail Extensions)

**XML** – bendros paskirties duomenų struktūrų bei jų turinio aprašomoji kalba (eXtensible Markup Language)

**XSLT** – kalba, aprašanti XML dokumento transformaciją į kitokios struktūros XML dokumentą (eXtensible Stylesheet Language Transformations)

**GUI** – Grafinė vartotojo sąsaja (Graphical User Interface)

**IPVM** – Įeinančių Prekių Valdymo Modulis

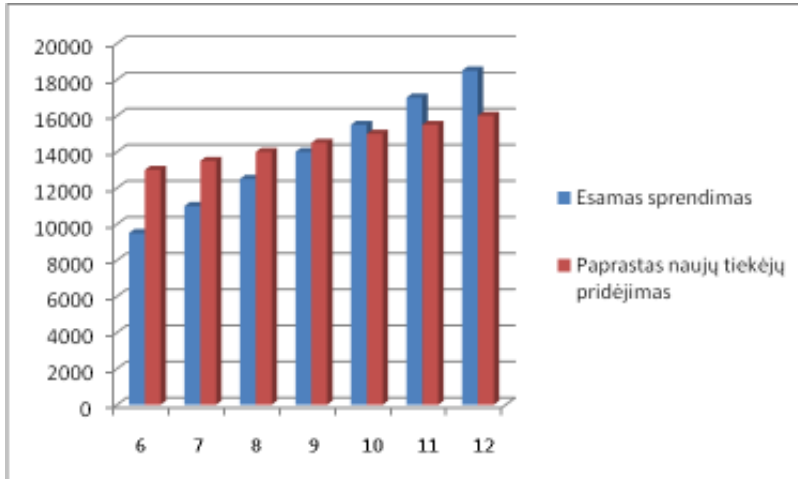
## 8. Literatūros šaltinių sąrašas

1. Wincel J. P. *Lean Supply Chain Management: A Handbook for Strategic Procurement*, Productivity Press, 2004
2. Statistikos departamentas, *Įmonių skaičius pagal dydį ir ekonominę veiklą*, 2007 [žiūrėta 2008-01-27]. Prieiga per internetą: <<http://www.std.lt/lt/pages/view/?id=1247>>
3. Statistikos departamentas, *Informacinių technologijų panaudojimas įmonėse*, 2007 [žiūrėta 2008-01-27]. Prieiga per internetą:  
<<http://www.std.lt/uploads/docs/3.Informacini%C5%B3%20technologiju%20panaudojimas%20%C4%AFmon%C4%97se.doc>>
4. Pfleeger S. L. *Software Engineering theory and practice. Second Edition*. Prentice-Hall, 2001.
5. Hughes B.; ir Cotterell M. *Software Project Management. Second Edition*, McGraw-Hill, 2000.
6. *RADIO BEACON sistema* [žiūrėta 2008-01-23]. Prieiga per internetą:  
<<http://www.radiobeacon.com>>
7. *AIVA 9001 verslo valdymo sistema* [žiūrėta 2008-01-24]. Prieiga per internetą:  
<[www.9001.aiva.lt](http://www.9001.aiva.lt)>
8. *Product Information for Microsoft Navision* [žiūrėta 2008-01-23]. Prieiga per internetą:  
<<http://www.microsoft.com/BusinessSolutions/Navision/product/default.msp>>
9. *Equinox Europe – VISION*, [žiūrėta 2008-01-23].  
Prieiga per internetą :<[http://www.equinoxlt.com/prod\\_vision.php](http://www.equinoxlt.com/prod_vision.php)>
10. Carlton J. C. *It's probably best to avoid vertical accounting software*, 2003  
[žiūrėta 2008-01-22]. Prieiga per internetą:  
<<http://www.asaresearch.com/articles/verticalsolutions.htm>>
11. Sertvytis R. *Pardavimų automatizavimas ir kontaktų su klientais valdymas*, 2004

- [žiūrėta 2008-03-05]. Prieiga per internetą <<http://www.ebiz.lt/article.php3/22/6808/6>>
12. Piasecki D. *Warehouse management systems*, 2006 [žiūrėta 2008-03-05]. Prieiga per internetą: <[http://www.inventoryops.com/warehouse\\_management\\_systems.htm](http://www.inventoryops.com/warehouse_management_systems.htm)>
  13. Barnes C. *No breakout new technology in 2006, but plenty of activity in existing apps*, 2006.
  14. Chorafas D. N. *Integrating ERP, CRM, Supply Chain Management, and Smart Materials*, 2000
  15. Calvin B. L., Ph.D. *Demand Chain Optimization: Pitfalls and Key Principles*, 2004
  16. Amelia S. C. *The impact of purchasing and supplier involvement on strategic purchasing and its impact on firm's performance*, 2002
  17. Gamma E., Helm R., Johnson R., Vlissides J. *Design Patterns: Elements of Reusable Object Oriented Software*. Addison-Wesley 1994.
  18. Behrouz A. F., Fegan S. C., *TCP/IP Protocol Suite*. McGrawHill, 2003.
  19. *An Open Specification for Pretty Good Privacy* [žiūrėta 2008-01-17]. Prieiga per internetą: <<http://www.ietf.org/html.charters/openpgp-charter.html>>
  20. *S/MIME Mail Security* [žiūrėta 2008-01-17]. Prieiga per internetą: <<http://www.ietf.org/html.charters/smime-charter.html>>
  21. Gopal K. K., Wong A. *Business Excellence model for supply chain management*, 1999
  22. *Electronic Data Interchange*, [žiūrėta 2008-04-12] Prieiga per internetą: <[http://en.wikipedia.org/wiki/Electronic\\_Data\\_Interchange](http://en.wikipedia.org/wiki/Electronic_Data_Interchange)>

## 9. Priedai

A. Priedas. Kodo eilučių kaita didinant tiekėjų skaičių.



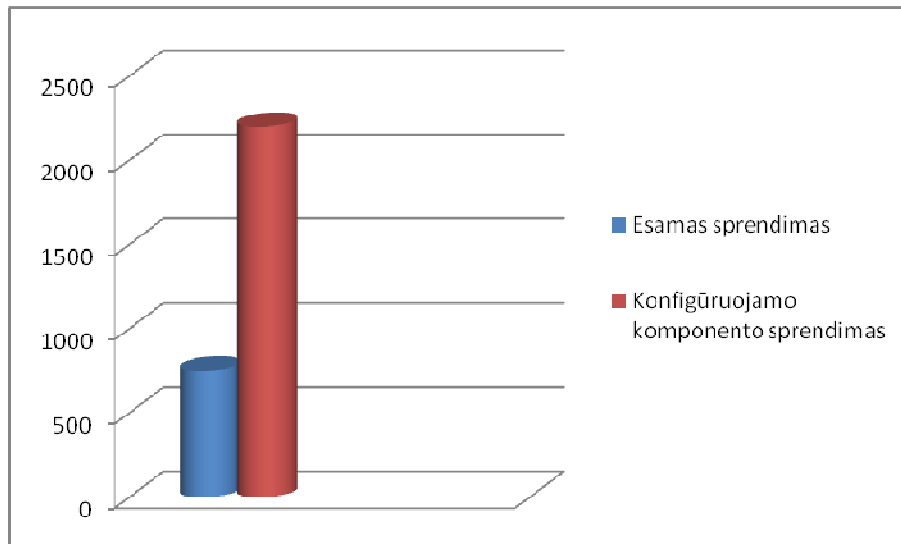
Vertikali ašis žymi kodo eilučių skaičių; Horizontali ašis žymi tiekėjų skaičių

B. Priedas. Paprasto naujų tiekėjų pridėjimo sprendimo pasirinkimo augimas augant tiekėjų skaičiui



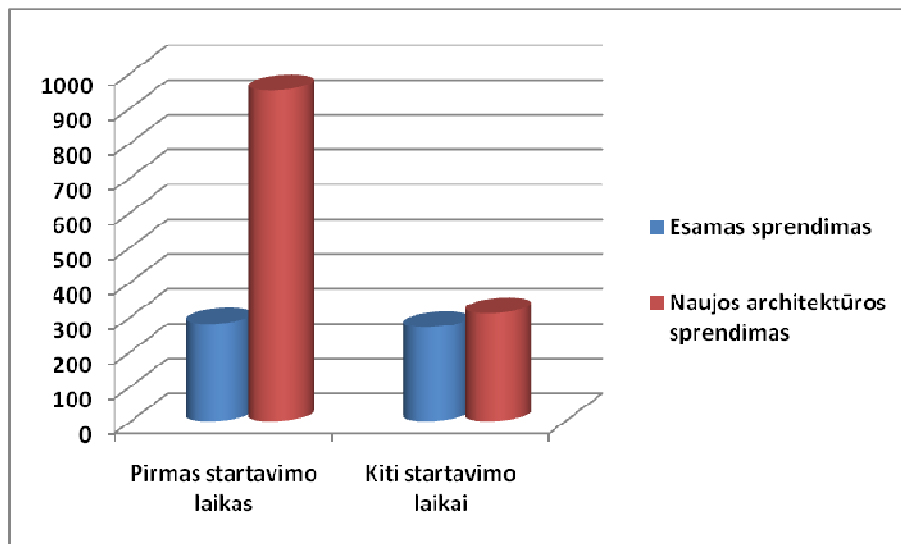
Vertikali ašis rodo procentinę dalį lyginant su esamu sprendimu; Horizontali ašis rodo tiekėjų skaičiaus augimą

C. Priedas. Konfigūrojamo komponento sprendimo ir esamo sprendimo efektyvumo palyginimas



Vertikali ašis – laikas mili sekundėmis

D. Priedas. Grafinės vartotojo sąsajos efektyvumas



Vertikali ašis – laikas mili sekundėmis