

**KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
INFORMACIJOS SISTEMŲ KATEDRA**

**Birutė Kudirkaitė
Vaidas Žilionis**

**SMULKAUS IR VIDUTINIO DYDŽIO ĮMONIŲ
VIDAUS VALDYMO SISTEMA**

Magistro darbas

**Vadovė
doc. L. Nemuraitė**

KAUNAS, 2006

**KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
INFORMACIJOS SISTEMŲ KATEDRA**

**SMULKAUS IR VIDUTINIO DYDŽIO ĮMONIŲ
VIDAUS VALDYMO SISTEMA**

Magistro darbas

Vadovas
doc. L. Nemuraitė

Recenzentas
doc. E. Mačikėnas

Atliko
IFM 0/4 gr. stud.
B. Kudirkaitė
V. Žilionis

KAUNAS, 2006

SUMMARY

„INTERNAL MANAGEMENT SYSTEM FOR SMALL & MEDIUM ENTERPRICES “

The main problem analyzed in this work was the need to design and create an internal management system for small and medium enterprises, that would meet their particular requests on usability, extensibility and adaptability at available costs. As solution to this problem was deep analysis of available systems of that kind, and creation of our own system, that had met necessary requirements and even more – had some technological improvements, such as reduced number of users actions, necessary to make things done. This was implemented with the help of AJAX programming technique and a special framework, created for this system. This system can be used to manage projects in a small or medium enterprise, collect statistics about projects, staff work, distribute project tasks to staff and monitor the implementation. There also is an opportunity to generate various reports, such as summary of accomplished tasks for every employee, lists of project tasks and their status and so on. This system can be easily installed in any enterprise because of its client – server architecture, where client connects to the server with the help of a web browser, that can be found in every computer, so there is no need to install system in all computers in enterprise. In that way maintenance costs are reduced, because all you have to look after is server with a system on it.

Turinys

1.	Įvadas	8
2.	Įmonių vidaus valdymo sistemų kompiuterizavimo galimybių analizė.....	9
2.1.	Analizės tikslas.....	9
2.2.	Tyrimo sritis, objektas ir problema	9
2.3.	Kompiuterizuojamos organizacijos vidaus valdymo veiklos procesų analizė	11
2.4.	Vartotojų analizė	13
2.4.1.	Vartotojų aibė, tipai ir savybės.....	13
2.4.2.	Vartotojų tikslai ir problemos	14
2.5.	Problemos sprendimo metodų literatūros šaltiniuose analizė	15
2.6.	Panašių sistemų (Lietuvos ir tarptautiniu mastu) analizė.....	17
2.6.1.	SAP sukurta įmonės resursų valdymo sistema mySAP ERP	17
2.6.2.	mySAP ERP Žmogiškųjų resursų valdymo (HCM) savybės ir funkcijos	18
2.6.3.	mySAP ERP Corporate Services projektų ir darbų valdymo modulis.....	19
2.6.4.	Microsoft Navision UBP – vidaus valdymo sistema nedidelėms įmonėms.....	19
2.6.5.	Epicor Scala sistema.....	21
2.6.6.	Scala projektų valdymo modulis	23
2.6.7.	Gatavų produktų analizės apibendrinimas	24
2.7.	Architektūros ir galimų įgyvendinimo priemonių variantų analizė	26
2.7.1.	Panaudotos realizavimo priemonės	27
2.7.2.	Programavimo kalba, duomenų bazė	28
2.8.	Sukurtos sistemos apibrėžimas.....	31
2.9.	Darbo tikslas ir siejami privalumai	32
2.10.	Kompiuterizuojamos sistemos funkcijos	33
2.11.	Reikalavimai duomenims.....	34
2.12.	Nefunkciniai reikalavimai ir apribojimai	36
2.13.	Rizikos faktorių analizė.....	36
2.14.	Rezultato kokybės kriterijai	36
2.15.	Analizės išvados.....	37
3.	Įmonių vidaus valdymo sistemos reikalavimai	38
3.1.	Reikalavimų specifikacija	38
3.2.	Dalykinės srities modelis	45
4.	Įmonės vidaus valdymo sistemos projektas	46
4.1.	Sistemos modelio pagrindimas ir esmės išdėstymas.....	46
4.2.	Sistemos architektūra - statinės struktūros modelis	46
4.2.1.	Loginė visos sistemos architektūra	46
4.2.2.	Vartotojo paslaugos.....	47
4.2.3.	Veiklos paslaugos.....	49
4.2.4.	Duomenų paslaugos	50
4.3.	Duomenų bazės schema	51
4.4.	Sistemos failų struktūra.....	52
4.5.	Realizuotas sistemos karkasas.....	53
4.6.	Įdiegimo modelis.....	58
5.	Ekspirimentinis sistemos tyrimas	59
5.1.	Ekspirimentinės sistemos kūrime panaudotų pagalbinių priemonių aprašymas	59
5.2.	Ekspirimentinės sistemos veikimas	60
5.3.	Ekspirimentinio diegimo aprašymas	64
5.4.	Sukurtos sistemos charakteristikų bei atitikimo reikalavimams tyrimas	65
5.5.	Kas jau padaryta, ir ką dar būtų galima padaryti ateityje.....	66
6.	Išvados.....	68
7.	Literatūra.....	69

8	Terminų ir santrumpų žodynas.....	71
9	PRIEDAI	72
9.1	IT konferencija – seminaras.....	72
9.2	PHP konferencija.....	79

Lentelių sąrašas

Lentelė 1. Vartotojų problemos bei planuojamas jų sprendimas	14
Lentelė 2 . Microsoft Navision UBP teikiamos funkcijos	20
Lentelė 3 . Planuojamos panaudoti realizavimo priemonės	27
Lentelė 4. Prisijungimo funkcijos specifikacija	38
Lentelė 5 . Dokumentų šablonų naudojimas	38
Lentelė 6 . Klientų informacijos registravimo funkcijos specifikacija	39
Lentelė 7. Atliktų darbų registravimo funkcijos specifikacija	39
Lentelė 8 . Projekto registravimo funkcijos specifikacija	39
Lentelė 9 . Darbų sąrašo funkcijos specifikacija	40
Lentelė 10. Darbų paskirstymo funkcijos specifikacija	40
Lentelė 11. Darbų tipų registravimo funkcijos specifikacija	40
Lentelė 12. Vartotojų teisių stebėjimo funkcijos specifikacija	41
Lentelė 13. Naujo vartotojo registravimo funkcijos specifikacija	41
Lentelė 14. Darbų būklės stebėjimo funkcijos specifikacija	41
Lentelė 15. Vartotojų teisių stebėjimas	42

Paveikslėlių sąrašas

Pav. 1 Įmonės padalinio veiklos diagrama	12
Pav. 2 Sistemos savybių diagrama	15
Pav. 3 Komponentinė sistemos architektūra, kai vartotojo sąsaja yra interneto naršyklė	26
Pav. 4 Tradicinio web aplikacijų modelio palyginimas su Ajax aplikacijos modeliu	28
Pav. 5 PHP naudojimo statistika	29
Pav. 6 Kuriamos sistemos funkcinų savybių medis	31
Pav. 7 Kuriamos sistemos nefunkcinų savybių medis	32
Pav. 8 Sistemos panaudojimo atvejų diagrama	33
Pav. 9 Projekto registravimo veiklos diagrama	42
Pav. 10 darbų (užduočių) sąrašo kūrimo veiklos diagrama	43
Pav. 11 Atliktų darbų sąrašo pildymo veiklos diagrama	44
Pav. 12 Konceptuali UML klasių diagrama, parodanti sistemos dalykinę sritį	45
Pav. 13 Sistemos loginė schema vaizdžiai	46
Pav. 14 Loginė sistemos architektūra	47
Pav. 15 Vartotojo paslaugų klasių diagrama	48
Pav. 16 Vartotojo sąsajos navigacijos planas	49
Pav. 17 Detali projekto valdymo veiklos paslaugų klasių diagrama	50
Pav. 18 klasė darbui su duomenų baze	50
Pav. 19 Duomenų bazės schema	51
Pav. 20 Sistemos failų struktūros pavyzdys	52
pav. 21 Sistemos veikimas kliento ir serverio pusėse	53
Pav. 22 Užduoties atlikimo procento fiksavimo langas	54
Pav. 23 Vartotojo veiksmų kiekis standartinėje realizacijoje ir panaudojus AJAX metodiką	55
pav. 24 JSON metodu aprašyti duomenys	56
Pav. 25 Sistemos įdiegimo modelis	58
Pav. 26 Pranešimo atvaizdavimas sistemoje	59
Pav. 27 Kalendorius	59
Pav. 28 Spalvos pasirinkimas	59
Pav. 29 Sistemos debugeris, atvaizduojantis objektų bei duomenų informaciją kliento dalyje	60
Pav. 30 Prisijungimo langas	60
Pav. 31 Pradinis langas, vartotojo duomenys	61
Pav. 32 TimeSheet vedimas	62

Pav. 33 Užduoties informacijos redagavimo langas	63
Pav. 34 Bendras užduočių sąrašo langas	63
Pav. 35 Darbuotojų darbų ataskaitos langas.....	64
Pav. 36 Vartotojo informacijos redagavimo langas	65

1. Įvadas

Šiandieninės ekonomikos sąlygomis, tam, kad išgyventų ir uždirbtų pelną, įmonės susiduria su augančiais reikalavimais tiek verslo procesų kokybei, tiek sprendimų priėmimo greičiui. Reikia ne tik sugebėti greitai ir efektyviai priimti valdymo sprendimus, bet taip pat analizuoti esamą situaciją ir planuoti būsimus darbus. Čia į pagalbą ateina informacinės technologijos. Tam tikros programinės įrangos pagalba galime dalinai arba visiškai automatizuoti įmonės verslo procesus, ir tuo pačiu pašalinti trikdžius, atsirandančius dėl žmogiškojo faktoriaus veikimo. Darbų organizavimas, savalaikis informacijos pateikimas, darbų paskirstymas ir resursų užimtumo planavimas, atlyginimų skaičiavimas – tai tik keletas iš įmonės veiklos aspektų, kuriuos galima pagreitinti automatizuojant ir tuo pačiu labai palengvinti už šiuos procesus atsakingų darbuotojų darbą.

Prieš kuriant sistemą buvo išanalizuoti jau esamų pasaulinėje ir Lietuvos rinkoje produktų aprašymai, jų galimybės, taip pat atlikta detali įmonės padalinio veiklos analizė ir išskirti reikalavimai būsimai sistemai. Naudojantis informacija, rasta internete, pasirinkta labiausiai tinkama tokio tipo sistemos realizavimo metodologija.

Pasinaudojant savybių modeliais (Feature modelling) buvo sudaryti sistemos būtinų ir neprivalomų savybių medžiai, iš kurių po to buvo suformuluoti reikalavimai kuriamos sistemos funkcionalumams. Taip pat prieš kuriant tokio tipo sistemą reikėjo apžvelgti galimus realizavimo būdus ir pasirinkti labiausiai mūsų poreikius atitinkantį sprendimą.

Remiantis asmenine patirtimi WEB programavime, kuriamą sistemą nuspręsta daryti kliento-serverio tipo, veikiančią internete, ir jos realizavime pritaikyti pakankamai naują AJAX programavimo metodiką. Ši metodika pasirinkta dėl jos privalomų - dinamiškumo, galimybės atliktus tam tikrus veiksmus kliento dalyje gauti reikalingą duomenų porciją iš serverio, neperkraunant viso puslapio.

Kuriant sistemą panaudotas mūsų sukurtas programinis „varikliukas“, pagreitinantis komunikaciją su serveriu ir padedantis lengviau kurti paprastus programinius elementus.

Sukurta sistema šiuo metu pritaikyta konkrečiai įmonei, tačiau tai yra adaptyvi sistema, lengvai pritaikoma kitų panašaus pobūdžio įmonių vidaus valdymui.

Taip pat reiktų paminėti, kad apie šiame darbe panaudotus sprendimo būdus ir technologijas buvo skaitomi pranešimai 2005m. gegužės 4 dieną vykusioje KTU INFOSA organizuotoje konferencijoje „Remote programming“, kuri vyko KTU festivalio metu ir 2005m. lapkričio 26 dieną vykusioje PHP bendruomenės organizuotame renginyje „PHP konferencija 2005“.

Darbo pasidalinimas buvo toks:

Vaidas Žilionis: 4 skyriaus 4.2 – 4.6 poskyriai, 5 skyriaus 5.1 – 5.3 poskyriai, 8, 9 skyriai.

Birutė Kudirkaitė: 1, 2, 3 skyriai, 4 skyriaus 4.1 poskyris, 5 skyriaus 5.4 ir 5.5 poskyriai, 6, 7 skyriai.

2. Įmonių vidaus valdymo sistemų kompiuterizavimo galimybių analizė

2.1. Analizės tikslas

Atliekamos analizės tikslas – išanalizuoti turimą informaciją apie įmonės vidaus valdymo sistemas ir suformuoti reikalavimus mūsų kuriamai sistemai.

Tam tikslui pirmiausia reikia apžvelgti informaciją, pateikiamą apie tokias sistemas užsienyje ir Lietuvoje, išsiaiškinti kokios yra populiariausios įmonių vidaus valdymo sistemos, kokiais bruožais pasižymi jų pateikiami moduliai ir suformuluoti reikalavimus savo sistemai.

2.2. Tyrimo sritis, objektas ir problema

Įmonės vidaus valdymo sistema – tai sistema, diegiama įmonėje siekiant efektyviai automatizuoti įmonėje veikiančius verslo procesus, padidinti darbų našumą, pagerinti informacijos pasiekiamumą ir bendrą įmonės darbo kokybę bei pelningumą. Tokio tipo sistemos būna modulinio tipo – t.y. atskiri moduliai apima tokias įmonės veiklos sritis kaip projektų, resursų valdymą, darbų organizavimą, žmogiškųjų resursų valdymą, finansus, logistiką ir pan. Dėka integracijos tarp įmonėje įdiegtų modulių, valdymas tampa geriau organizuotas, o reikalinga informacija pasiekama visada laiku ir tokia, kokios reikia.

Įmonės vidaus valdymo sistemos taikymo **privalumai**:

- Efektyvesnis planavimas;
- Dėl geresnės koordinacijos sumažėja daromų klaidų kiekis;
- Mažesni užsakymų priėmimo, gamybos, inventorizacijos, apskaitos ir transportavimo kaštai;
- Sumažėjus kaštams padidėja pelningumas;
- Atsiranda galimybė sutrumpinti užsakymo realizavimo laiką;
- Padidėja vykstančių procesų “skaidrumas”, t.y. galima aiškiau matyti procesų eigą, matomas aiškus įmonės vaizdas konkrečiu laiko momentu.
- Tikslī informacija gaunama laiku, kas yra būtina siekiant priimti teisingus sprendimus;
- Leidžiamas didesnis gamybos pritaikymas konkretiems kliento poreikiams
- Kliento pasitenkinimas, kylantis iš aukščiau paminėto faktoriaus, leidžia padidinti pardavimų apimtį, gaunamas iš pardavimų pajamas, padeda stipriau įsitvirtinti rinkoje ir padidinti pelningumą.

Tačiau turi šios sistemos ir apribojimų bei **trūkumų**:

- Šių sistemų įdiegimas ir palaikymas gali būti labai brangus;
- Kai kurios sistemos yra sudėtingos naudoti;
- Ši sistema visuomet yra silpniausia grandis grandinėje – klaida vienoje srityje gali labai paveikti kitų sričių darbą.

Populiariausių įmonių vidaus valdymo sistemų **tiekJėjai** yra:

Microsoft <http://www.microsoft.com/businesssolutions/navision/default.msp>

Epicor <http://www.epicor.com/www/>

SAP <http://www.sap.com/index.aspx>

Toliau nagrinėsime šiuos paminėtų gamintojų produktus – Microsoft Navision UBP, Epicor Scala ir mySAP ERP.

Pasaulinėje rinkoje siūlomas platus asortimentas programinės įrangos, skirtos įmonių vidaus valdymo problemoms spręsti. Tarp jų yra vadinamos ERP (Enterprise Resource Planning) sistemos. Tačiau seniai žinoma, kad universalus, visiems be išimties atvejams tinkamo sprendimo nebūna, tad renkantis programinę įrangą konkrečiai įmonei, reikia labai gerai žinoti jos poreikius ir galimybes. Pasaulinėje rinkoje pripažinimą išsikovojusių produktų gamintojai teigia, kad gali pasiūlyti kiekvienam klientui tinkamą sprendimą, tačiau yra keletas aspektų, į ką Lietuvos įmonių vadovai turėtų atkreipti dėmesį prieš rikdamiesi tokio tipo programinę įrangą:

- Kokios apimties paslaugų paketą pasirinkti diegti savo įmonėje. Kartais būna atvejų, kad pasirenkamas galinga sistema, tačiau vėliau paaiškėja, jog didelė dalis jos teikiamų funkcionalumų lieka nepanaudoti arba nepanaudojami pilnai. Todėl labai svarbu sudaryti konkrečius reikalavimus sistemai.
- **Programinės įrangos kaina.** Čia yra pats pagrindinis kriterijus, nes pasaulyje pripažinti produktai, tokie kaip MySAP ERP, Microsoft Navision, Epicor Scala ir kiti yra brangūs. Microsoft teigia, kad į įmonėje diegiamo sprendimo kainą įeina tokie kriterijai kaip:
 - Reikalinga verslui ir kita susijusi programinė įranga;
 - Vartotojų skaičius ir konkrečios jų priėjimo teisės;
 - Paslaugų, reikalingų kompanijai, lygis.

Pasak Microsoft, konkretaus programinio sprendimo kaina gali svyruoti nuo 5000\$ iki 500000\$, bet konkrečiai pasakyti kiek kainuoja sistema labai sunku, nes tai labai priklauso nuo įmonės individualių poreikių.

Be to, reikia nepamiršti, jog naujos sistemos įmonėje diegimas ir pritaikymas naudoti neapsieina be papildomų išlaidų techninei įrangai, darbuotojų mokymams, specialistų, prižiūrinčių sistemą, atlyginimams ir pan.

Taigi, įvertinus kainą, o taip pat ir riziką, kad brangaus gatavo produkto įdiegimas gali nepasiteisinti, nori nenori kyla klausimas, kokios yra alternatyvios galimybės šiai problemai spręsti. Vienas tokių sprendimų – darytis tokio tipo sistemą patiems.

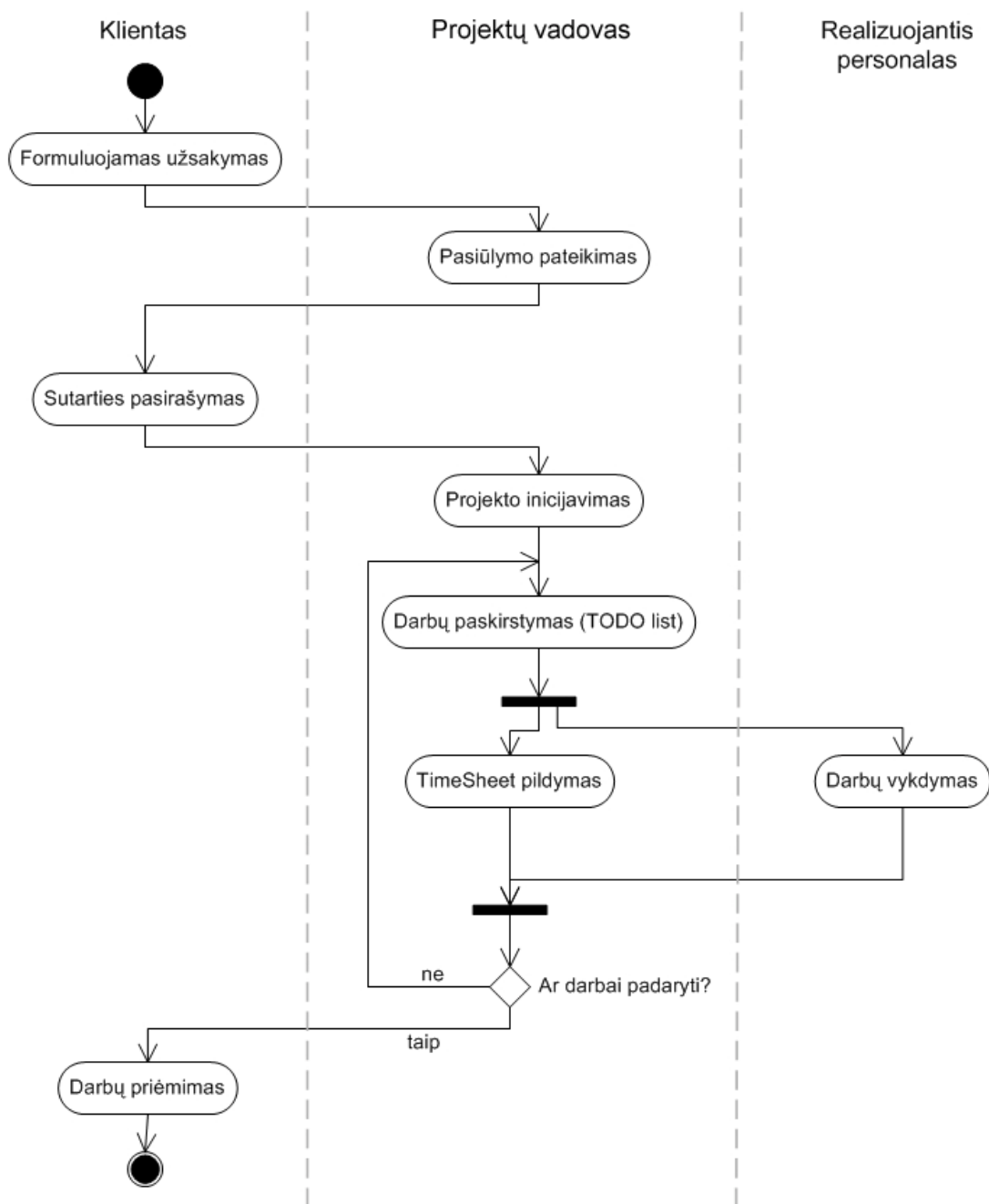
Vienas iš šio darbo atlikėjų dirba reklamos paslaugas teikiančioje įmonėje ir yra susipažinęs su ten vykstančiais procesais, ir jam jau yra tekę padaryti panašaus pobūdžio sistemą. Tačiau laikui bėgant įmonės veikla plėtėsi ir, siekiant didesnio efektyvumo, atsirado poreikis tą jau veikiančią sistemą perprojektuoti. O kadangi panašia veikla užsiimančių įmonių Lietuvoje nemažai, galima tikėtis, jog tokia sistema, kurios reikalingumas ir naudingumas jau pasitvirtino vienoje įmonėje, galėtų atnešti naudos ir kitai. Iš to ir kilo idėja padaryti įmonės vidaus valdymo sistemą, kurią vėliau būtų galima pardavinėti kitoms panašaus pobūdžio įmonėms.

Remiantis visais šiais samprotavimais buvo iškeltas tikslas sukurti efektyvią, patogią vartoti sistemą, kuri atitiktų konkrečius įmonės poreikius ir kainos atžvilgiu būtų labiau prieinama nei jau minėti gatavi rinkoje siūlomi sprendimai.

2.3. Kompiuterizuojamos organizacijos vidaus valdymo veiklos procesų analizė

Čia pateikiame įmonės padalinio, kuriame pirmiausia bus taikoma sukurtoji sistema, veiklos diagramą (pav.1) Įmonės veikla – reklamos paslaugos, o konkrečiai analizuojamas padalinys dirba su Internetiniais projektais. Ši diagrama parodo, kokias stadijas praeina užsakymas, kol jis būna galutinai realizuotas. Diagramoje matote tris veikėjų grupes – klientą, realizuojantį personalą ir tarpininką tarp jų – projektų vadovą. Klientas pirmiausia suformuluoja savo pageidavimą, kokios paslaugos jis nori. Tada projektų vadovas jam pateikia pasiūlymą. Suderinus visas pasiūlymo sąlygas, pasirašoma sutartis su klientu ir inicijuojamas projektas. Kliento projektas išskaidomas į atskirus darbus (suformuojamas TODO sąrašas), kuriuos vėliau paskirsto tarp realizuojančio personalo (programuotojų, dizainerių, projektuotojų ir panašiai..). Darbuotojams dirbant prie projekto, atlikinėjami darbai iš TODO sąrašo, tuo pačiu pildoma ir atliktų darbų ataskaita

(timesheet). Pasinaudojus ataskaitose surinkta informacija galima atlikti darbų analizę keletu pjūvių. Kai visi darbai atlikti, atliekamos darbų perdavimo klientui procedūros ir procesas pasibaigia. Ši diagrama yra apibendrintas vieno įmonės veiklos padalinio modelis, ir kiekvieną šios diagramos darbų dar galima skaidyti į smulkesnius darbus, kol gautume schemą, sudarytą elementarių darbų.



Pav. 1 Įmonės padalinio veiklos diagrama

2.4. Vartotojų analizė

2.4.1. Vartotojų aibė, tipai ir savybės

Šios sistemos vartotojų ratas yra gan platus. Tiesiogiai su ja dirba įmonės darbuotojai bei savininkai, netiesiogiai įtraukiami klientai bei tiekėjai. Tokią sistemą, kurią sukūrėme, naudinga ir efektyvu taikyti tik tada, kai vyksta komandinis darbas ir reikalinga plataus masto organizacija.

Kadangi įmonėje vykstantys procesai yra glaudžiai vieni su kitais susiję, prie kiekvienos užduoties dirba daug darbuotojų, organizuojantiems asmenims būtina turėti galimybę stebėti darbų vykdymo eigą ir rezultatus. Tad grubiai vartotojus galima būtų suskirstyti į tokias grupes:

Stebėtojai (čia būtų klientai, įmonės savininkai, strateginiai planuotojai ir kt.) – tai tokie vartotojai, kuriems reikalinga statistinė informacija apie įmonės veiklą.

Tokia informacija būtų vidutiniai projektų atlikimo laikai, resursų užimtumo stebėjimas, darbo efektyvumo stebėjimas ir panašiai.. Visa ta informacija būtų panaudojama darant tolesnius strateginius sprendimus, priimant naujus nuostatus ar sudarant naujas sutartis su klientais.

Organizatoriai (padalinių vadovai, projektų vadovai ir kt.) – asmenys, kurie atsakingi už darbų organizavimą. Jiems būtina ne tik statistika, bet ir konkretūs duomenys apie kiekvieno projekto būklę (statusą). Jie planuoja darbus, jie atsakingi už kiekvieno projekto įvykdymą laiku, todėl jiems turi būti prieinama informacija apie tai, kokioj stadijoje esamu momentu yra projektas, kas prie jo dirba ir kiek laiko, ar tas projekto vykdymas „telpa“ į numatytą laiką, ar tas projektas bus padarytas iki numatyto termino, kokie resursai užimami dirbant su tuo projektu ir panašiai.

Vykdytojai (realizacinės dalies atstovai, tokie kaip projektuotojai, programuotojai, dizaineriai, maketuotojai ir pan.) – šie žmonės taip pat turi žinoti jiems paskirtas užduotis, jų įgyvendinimo laiką . Vykdytojai turi fiksuoti sistemoje savo atliktus darbus bei sugaištą laiką prie jų, nes nuo to priklauso kiekvieno iš jų atlyginimas (šita funkcija padeda motyvuoti darbuotojus dirbti kokybiškai ir greitai, siekiant didesnio motyvacinio atlyginimo). Kai vykdytojai sistemoje registruoja savo darbus ir laiką, organizuojantys ir statistiką renkantys asmenys mato, kiek prognozuojamos darbų trukmės atitinka realias ir pagal tai gali koreguoti planus.

Išoriniai vartotojai (čia būtų buhalteriai ir kiti asmenys) naudotų sistemoje surinktus duomenis darbuotojų atlyginimams skaičiuoti bei bendrai įmonės apskaitai vesti.

Reikia pažymėti, jog šitoks vartotojų suskaidymas į grupes yra sąlyginis, gautas tiesiog apibendrinus jiems reikalingos informacijos paskirtį.

2.4.2. Vartotojų tikslai ir problemos

Kaip jau vardinta 2.4.1 paragrafe, šią sistemą vartojančių žmonių poreikiai yra labai įvairūs. Pagrindinius jų galima būtų išvardinti taip:

- statistinės informacijos rinkimas
- darbų eigos stebėjimas
- resursų užimtumo stebėjimas
- darbų paskirstymas
- realaus darbo trukmės fiksavimas
- koordinuotas informacijos tarp įmonės darbuotojų platinimas
- kt..

Toliau pateiksime keletą problemų, išskylančių įmonės veikloje ir parodančių, jog reikalingas sprendimas joms pašalinti.

Lentelė 1. Vartotojų problemos bei planuojamas jų sprendimas

Problema	Kaip yra dabar	Kaip planuojama padaryti
Sunku nustatyti, kiek konkrečiai darbo sugaišta prie kiekvieno projekto	Sugaišto laiko darbams skaičiavimas yra tik apytikslis.	Bus realizuota galimybė fiksuoti 15min. tikslumu sugaištą laiką prie konkretaus darbo. Kiekvienas darbuotojas pats asmeniškai rūpinsis savo tvarkaraščio pildymu.
Sistema užima daug tinklo pralaidumo.	Dabar sistema veikia kaip web puslapis, vartotojas gali duoti tikrai užklausas, o visas apdorojimo procesas vyksta serveryje. Po kiekvienos užklausos apdorojimo vartotojo sąsajos puslapis turi persikrauti (t.y. iš naujo siunčiami visi duomenys).	Planuojama realizuoti kitokią architektūrą, kurios veikimo principas leistų vartotojui padavus užklausą, siųsti tik užklausos prašomus duomenis, o likusiai sąsajai persikrauti nereikėtų. Taip būtų taupomas laikas ir tinklo pralaidumas.
Prieinamumas per internetą	Sistema veikia tik įmonės viduje ir priejimo iš išorės kol kas nėra. Tai nepatogu tiems darbuotojams, kuriems sistemos teikiamos galimybės ir informacija reikalingi būnant už įmonės ribų	Planuojama realizuoti galimybę prisijungti prie šios sistemos iš išorės, aišku, saugumo sumetimais pateikti tik keletą pačių reikalingiausių sistemos funkcionalumų.
Klientų informavimas	Klientas, norėdamas sužinoti, kokioje stadijoje yra jo projektas, turi skambinti į įmonę ir teirautis.	Esant reikalui, bus galima suteikti klientui galimybę stebėti savo projekto būsenas per internetą.

2.5. Problemos sprendimo metodų literatūros šaltiniuose analizė

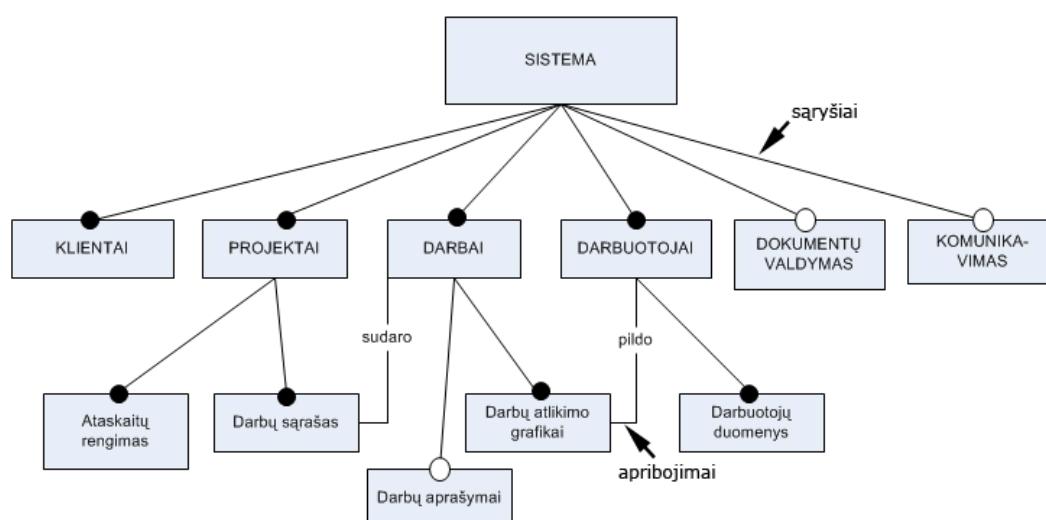
Iš patirties žinoma, kad Informacinių sistemų kūrimas neapsieina be rizikos. Siekiant išvengti didesnių finansinių nuostolių, atsirandančių dėl informacinės sistemos neatitikimo reikalavimams, jos ne visiškai tinkamo veikimo ir panašiai, sistemų projektuotojai labai linkę naudoti jau patikrintus ir išbandytus sistemos objektus (pakartotinis komponentų panaudojimas).

Labai retai būna, kad naujos sistemos būtų iš pagrindų kuriamos visiškai nepriklausomai, nieko nepanaudojant iš senosios sistemos. Senosios sistemos turi būti perprojektuojamos, tam kad jų architektūra labiau atitiktų naujus reikalavimus. Ir naujų sistemų architektūros kūrimas dažniausiai yra senosios sistemos restruktūrizuota versija, kurioje pašalinti senosios sistemos trūkumai.

Gera, lanksti, atitinkanti reikalavimus architektūra yra pagrindinė reinžinerijos proceso užduotis.

Naujosios sistemos architektūros projektavimui panaudojama įvairi informacija – senosios sistemos dokumentacija, išvesties kodas, naudojami dokumentai. Tačiau kartais viso to nepakanka, ir lieka spraga tarp dokumentacijos (kadangi ji gali būti pasenusi, nepakankamai detali ir neatitikti šiandieninės veiklos reikalavimų), išvesties kodo (galimas atvejis, kad sistema per laika buvo nuolat keičiama ir architektūra tapo nebe tokia kokia buvo pačioje pradžioje) ir reikalavimų. Kad užpildyti šią spragą ir surinkti dar daugiau naudingos informacijos, siūloma pasinaudoti sistemos savybių modeliavimu, kuris skirtas struktūrizuoti dalykinės srities savybes iš kliento pozicijų. Tai yra aiškus, paprastas, bet kartu informatyvus metodas, suprantamas tiek sistemos vartotojams, tiek naujosios sistemos projektuotojams.

Savybių modelyje savybėmis laikomos klientui vertingi sistemos teikiami funkcionalumai. Modelis pateikia hierarchinę tokių savybių struktūrą, kartu parodydamas ryšius tarp jų bei esamus apribojimus. Savybių modelyje kiekviena savybė gali būti pažymima kaip privaloma arba neprivaloma. Kaip pavyzdį pateiksime mūs nagrinėjamos sistemos savybių modelį:



Pav. 2 Sistemos savybių diagrama

Čia juodi apskritimai savybės bloko viršuje reiškia kad savybė privaloma, balti - neprivaloma. Apribojimai rodo, kokia savybė su kokia ir kaip siejasi.

Literatūros šaltiniuose teigiama, kad šis modelis buvo išvystytas tiek, kad teikė labai naudingą informaciją sistemos architektūros projektavimui. Pasinaudojant juo ir sekamumo ryšiais (traceability links), kurie rodo kuriame sistemos komponente šios savybės realizuotos, gaunama informacija, padedanti eliminuoti sistemos architektūros problemas.

Naudojantis šiuo modeliu, galima nustatyti architektūros disproporcijas ir komponentų dublikatus.

Siūlomas statistinis metodas, kuris analizuoja sistemos savybes ir susijusius architektūrinius komponentus ir aptinka disproporcijas tarp komponentų. Trumpai, šis metodas:

- Apskaičiuoja funkcionalumų skaičių, susijusių su architektūriniu komponentu;
- Apskaičiuoja vidutinį skaičių architektūriniuose komponentuose realizuotų savybių;
- Surikiuoja architektūrinius komponentus atsižvelgiant į santykį susijusių funkcionalumų su vidutiniu skaičiumi realizuotų funkcionalumų.

Naudojantis atliktų skaičiavimų rezultatais, galima nustatyti, kurie sistemos architektūros komponentai reikalauja perprojektavimo, o kurie išliko gerai subalansuoti.

Taigi, atliekant sistemos architektūros projektavimą bus atkreipiamas dėmesys ir į savybių modelį.

Toliau, naudojantis informacija, rasta internete, reikėjo pasirinkti labiausiai tinkamą tokio tipo sistemos realizavimo metodologiją. Pasirinkimo variantai buvo 3:

- **„Big Bang“ metodologija**, kai vidaus valdymo sprendimas įdiegiamas visoje įmonėje iš karto. Ši metodologija pasiteisina tik tada, kai vykdoma kruopščiai ir atidžiai. Kadangi įdiegiami visi sistemos moduliai, tai reiškia kad turi būti perorganizuojami visi verslo procesai iš karto, kas yra susiję su didele rizika, jog sistema nepasiteisins. Šiuo metu labai nedaug kad drįsta rinktis tokią metodologiją.
- **Modulinės realizacijos metodologija** remiasi tuo, kad vienu metu įmonėje diegiamas vienas įmonės vidaus valdymo sistemos modulis. Šis sprendimas yra pats populiariausias, kadangi rizika, jog realizavimas nepasiteisins- labai sumažėja, ir kainos atžvilgiu tai nėra taip brangu.
- **Į procesus orientuota metodologija** – tai variantas kuri mes pasirinkome. Ši metodologija sukoncentruota į keletą kritinių verslo procesų, esančių viename padalinyje ir apimančių palyginti nedaug verslo dalyvių, automatizavimą. Pagal šią metodologiją realizuotas sprendimas yra savotiška informacinė sistema. Ši informacinė sistema pritaikyta konkrečios įmonės verslo procesams, ir palaiapsniui gali būti išvystyta iki galingos vidaus valdymo sistemos lygio sprendimų

2.6. Panašių sistemų (Lietuvos ir tarptautiniu mastu) analizė

Lietuvoje vienos iš labiau paplitusių įmonės valdyme taikomų sistemų yra Microsoft Navision UBP, Scala, Labis ir kitos, tačiau smulkiau panagrinėsime tik Navision UBP bei Epicor Scala produktus.

Nors mySAP yra kur kas daugiau nei įmonės vidaus valdymo sistema, apžvelgsime ir mySAP ERP sistemos modulius, skirtus projektų bei žmogiškųjų resursų valdymui, nes norime susidaryti bendrą vaizdą, kokiais funkcionalumais pasižymi aukščiausio lygio sprendimai.

2.6.1. SAP sukurta įmonės resursų valdymo sistema mySAP ERP

Tai SAP korporacijos produktas, kuris, anot gamintojų „apjungia labiausiai pasaulyje išbaigtą, išplečiamą ir efektyvią programinę įrangą, skirtą įmonės resursų planavimui (ERP) su lanksčia, atvira technologijų platforma (NetWeaver), kuri gali įtakoti ir apjungti SAP ir ne SAP sistemas.“

mySAP sudaro keturi pagrindiniai moduliai, palaikantys pagrindinius verslo procesus. Tai būtų:

- mySAP ERP Financials (skirta finansų valdymui)
- mySAP ERP Human Capital Management (skirta žmogiškųjų resursų valdymui)
- mySAP ERP Operations (įmonės operacijų valdymas)
- mySAP ERP Corporate services (bendrujų paslaugų valdymas)

Praleidžiant visus reklaminius pagyrimus susikoncentruokime ties antruoju šios sistemos modulių, skirtu žmogiškųjų resursų valdymui.

Tam, kad sėkmingai konkuruoti rinkoje, reikia visus savo įmonės resursus, taip pat ir darbuotojus, suderinti siekiant įmonės verslo tikslų. Išskyla poreikis tradicines darbuotojų darbo organizavimo priemones ir būdus pakeisti visapusiška programa, skirta žmogiškųjų resursų valdymui. Su **mySAP ERP Human Capital Management** galima maksimizuoti kiekvieno darbuotojo vertę ir palyginti darbuotojo įgūdžius, užsiemimus ir motyvaciją siekiant verslo tikslų ir laikantis strategijos. **mySAP ERP HCM** taip pat pateikia įrankius, skirtus valdyti, matuoti ir įvertinti individualius ir komandinius pasiekimus.

Naudojantis šiuo įrankiu daromas gilesnis poveikis darbuotojų produktyvumui, kadangi darbuotojai labiau susikaupia ties vertingesniais darbais.

mySAP ERP HCM palaiko vertingų darbuotojų samdos, paskirstymo, kvalifikacijos kėlimo, motyvavimo ir išlaikymo procesus, pagerina šiuos procesus nuo pradžios iki galo. Šis modulis teikia integruotą, visos įmonės mastu teikiama funkcionalumą, kuris:

- modernizuoja darbuotojų valdymo procesus ir vientisai juos integruoja į globalius procesus
- teikia realaus laiko priėjimą prie informacijos, kas pagreitina sprendimų, susijusių su darbo jėga, priėmimą.
- Leidžia paskirti tinkamiausiems žmonėms tinkamiausius projektus tinkamiausiu laiku.
- Palaiko tiek darbuotojus, tiek vadovus visą jų buvimo įmonėje laiką.
- Įgalina darbuotojus valdyti procesus bendroje aplinkoje.

Suprojektuotas globaliems verslams, **mySAP ERP HCM** palaiko algalapių funkcijas, kontrolės reikalavimus ir geriausias praktikas iš daugiau nei 50 šalių. Jis integruojasi su verslo sistemomis ir gali būti pritaikytas jūsų poreikiams. Daugiau nei 9 000 kompanijų visame pasaulyje patiki **mySAP ERP HCM** valdyti daugiau nei 54 milijonus savo darbuotojų.

2.6.2. mySAP ERP Žmogiškųjų resursų valdymo (HCM) savybės ir funkcijos

Šis modulis pateikia tam tikras funkcijas, skirtas realizuoti tokioms verslo veikloms:

- **Darbuotojo gyvavimo ciklo valdymas** – pateikiamos funkcijos, kurios automatizuoja ir optimizuoja visas darbuotojo gyvavimo ciklo fazes. To pasekoje, galima rasti geriausius darbuotojus, vystyti ir įtakoti jų talentus, palyginti jų pastangas su bendrais rodikliais, maksimizuoti mokymų poveikį ir išlaikyti pačius geriausius darbuotojus.
- **Darbuotojų darbų valdymas** – šis programinis paketas padeda modernizuoti ir apjungti esminius darbuotojų darbus ir procesus, atsižvelgiant į kontrolės reikalavimų paisymą.
- **Darbo jėgos paskirstymas.** Siekiant efektyvaus darbo jėgos valdymo reikia žinoti, kaip paskirti tam tikriems projektams tinkamiausius žmones tinkamiausiu metu. **mySAP HCM** padidina darbo jėgos efektyvumą ir pelningumą leisdamas sukurti projektų komandas, pagrįstas darbuotojų įgūdžiais ir užimtumu. Taip pat leidžiama stebėti ir analizuoti progresą ir rezultatus, kad įsitikinti, jog resursai nešvaistomi ant neefektyvių projektų.

2.6.3. mySAP ERP Corporate Services projektų ir darbų valdymo modulis

MySAP ERP Corporate Services teikia galingą projektų ir atliktų darbų valdymo funkcionalumą, kuris leidžia projektų vadovams geriau identifikuoti, parinkti, nustatyti prioritetus ir valdyti organizacijos projektus apimant įmonės IT projektus, paslaugų projektus bei tyrimų ir tolesnio vystymo projektus.

Palaikomi šie projektų ir atliktų darbų valdymo procesai:

- **Strateginis darbų valdymas** – leidžia matyti visą verslą kaip turto, resursų ir projektų investicijų darbus, ir tuo būdu projektų vadovai įgalinami valdyti šiuos darbus siekiant gauti maksimalią naudą.
- **Projektų planavimas** – leidžia nustatyti projektų struktūrą naudojantis šablonais, apibrėžti projekto komandą, planuoti biudžetą ir nustatyti projekto užduotis.
- **Resursų ir laiko valdymas** – leidžia suplanuoti reikalingus kiekybinius ir kokybinius reikalavimus projektui, ir paskirstyti projekto atlikimą tarp tinkamiausių tam darbuotojų įmonėje. Projekto komandos nariai gali fiksuoti savo įdėtą darbą į projekto realizavimą.
- **Projektų vykdymas** – Leidžia stebėti tiek didelių, tiek mažų projektų vykdymą, užtikrinant kad projektai vykdomi laiku, efektyviai ir biudžeto ribose.
- **Projekto apskaita** – leidžia planuoti, skirti finansavimą ir stebėti projekto pajamas ir kaštus. Pagrindinis veiklos rezultatas matuojamas biudžetu, plano laikymusi ir personalo darbu, ir teikia centralizuotą projekto vykdymo vaizdą ir rizikos elementus.

2.6.4. Microsoft Navision UBP – vidaus valdymo sistema nedidelėms įmonėms

„Microsoft Business Solutions“ - 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. Microsoft Business Solutions - Navision programinė įranga teikia įrankius tiek eilinių darbuotojų kasdieninio darbo automatizavimui, tiek ir priemonės vadovams reikalingos agreguotos, realaus laiko analitinės informacijos formavimui. Programinės įrangos architektūra leidžia vienoje sistemoje nepriklausomai vesti kelių įmonių apskaitą, kartu konsoliduotai matyti jų veiklos rezultatus. Didelės konfigūravimo ir funkcinės galimybės kartu su lanksčia papildomo funkcionalumo technologija leidžia šią programinę įrangą sėkmingai diegti įvairaus profilio įmonėse bei užtikrina efektyvų investicijų atsipirkimą.

Programą pasižymi plačiu funkcijų diapazonu, greita įdiegimo procedūra, visišku suderinamumu su Microsoft programomis, bei pilno pritaikymo įmonės specifiniams poreikiams priemonėmis. Grafinė

programos vartotojo sąsaja sukurta griežtai laikantis visų šiuolaikinių darbo vietos reikalavimų bei Microsoft standartų.

Pateiksime lentelę su šio produkto teikiamais funkcionalumais ir jų aprašymais:

Lentelė 2 . Microsoft Navision UBP teikiamos funkcijos

Didžioji knyga	Modulis skirtas finansų apskaitai. Apsibrėžiamas įmonės sąskaitų planas, kuriame registruojama visa įmonės finansinių sandorių informacija. Galima registruoti operacijas bendrajame žurnale, sudaryti finansų ataskaitas, planuoti biudžetus, konsoliduoti kelių įmonių duomenis, paskirstyti kaštus. Modulis Atsakingieji centrai skirta pajamų, kaštų ar investicijų centrums (pavyzdžiui, įmonės filialams) valdyti, taip palengvinant verslo administravimą.
Ilgalaikis turtas	Visos galimybės apskaityti ilgalaikį turtą, registruoti su turto susijusias operacijas (pirkimo, perleidimo ir pan.), įvairiais būdais skaičiuoti nusidėvėjimą. Papildomai - galimybės apskaityti turto draudimą ir remontus.
Pardavimai	Šis modulis įgalina valdyti visą pardavimų veiklą, pradedant nuo pardavimo pasiūlymų ir baigiant pardavimo grąžinimais. Pardavimo kainodaros ir nuolaidų sistema leidžia automatizuoti pardavimo vadybininkų veiklą.
Verslo ryšių valdymas	Verslo ryšių valdymas apima visus kontaktus su kitomis įmonėmis ar asmenimis - tiek tradicinius verslo santykius (pirkimas/pardavimas), tiek elektroninę ar pašto korespondenciją, skambučius ir pan. Atsiranda papildomos galimybės vertinti marketingo kaštus įmonės viduje. Vartotojo patogumui sukurti verslo ryšių vedliai padeda pagal išankstinius šablonus sukurti laiškus ir kitus dokumentus.
Aptarnavimo valdymas	Microsoft Business Solutions- Navision modulis Serviso valdymas kartu su Verslo ryšių valdymo moduliu realizuoja ryšių su klientais valdymo funkcionalumą (vadinamąjį CRM). Jis skirtas toms įmonėms, kurios po prekės pardavimo dar ilgą laiką atlieka parduotų prekių aptarnavimo, garantinio remonto ir panašius darbus.
Pinigų valdymas	Modulis įgalina tvarkyti neribotą banko sąskaitų skaičių tiek litais, tiek užsienio valiuta. Tai vartotojui suteikia galimybę sekti visas savo banko sąskaitas viename ar keliuose bankuose.
Pirkimai	Modulis apima tiekimo veiklą nuo pirkimo užsakymų iki prekių gavimo. Galimi įprastiniai ir pasikartojantys pirkimo užsakymai, prekių gavimas į sandėlį atskirai nuo vertės pajamavimo. Tiekimo procesą galima planuoti, remiantis pardavimo užsakymais ir maksimaliais/minimaliais kiekiais sandėlyje.

Atsargos	Atsargų modulis apima prekių sąrašus, metodus savikainai skaičiuoti, prekių kainodarą, prekių judėjimą tarp skirtingų vietų. Yra galimybės komplektuoti prekes į vadinamąsias komplektavimo specifikacijas.
Sandėlio valdymas	Naujai sukurtas modulis papildo ankstesnį Microsoft Business Solutions- Navision Atsargų modulį naujomis atsargų tvarkymo įmonės viduje galimybėmis. Tai sandėlio vadybininko darbo įrankis. Prekės dabar gali ne tik "atvykti" į įmonę ir "išvykti" iš jos. Su prekėmis dabar galima atlikti tokias operacijas: gauti prekes į sandėlį; ruošti komplektacijai; komplektuoti; perkelti į "pardavimo" vietą, išsiųsti. Tokiu būdu galima tiksliai sekti pardavimo užsakymo vykdymą tarp užsakymo įvedimo ir išvežimo pirkėjui.
Gamyba	Microsoft Business Solutions- Navision Gamybos modulis šiandien jau sėkmingai diegiamas gamybinėse įmonėse Lietuvoje. Atnaujintas ir papildytas naujomis galimybėmis Microsoft Business Solutions- Navision Gamybos modulis dabar tapo trimis integruotais programos moduliais: gamybos branduolys, tiekimo planavimo branduolys, pajėgumų planavimo branduolys.
Ištekliai	Modulis skirtas įmonės vidiniams ištekliams (žmonėms arba įrengimams) apskaityti ir planuoti.
Personalas	Galimybė saugoti įmonės darbuotojų kartoteką. Modulis neapima darbo užmokesčio apskaitos, tam skirtas specialus papildomai sukurtas Lietuvai Darbo Užmokesčio modulis.
Darbai	Modulis įgalina valdyti vykdomus ilgalaikius projektus (pvz. statybinius, naujo produkto kūrimo ir pan.), sudaryti sąmatas atskiriems projekto etapams, planuoti įmonės išteklius, skirtus darbams, sekti projekto vykdymą, lyginti faktinius rezultatus su sąmatiniais.

2.6.5 Epicor Scala sistema

Scala - tai išvystyta ir funkcionali sistema, sudaryta iš integruotų modulių ir galinti pilnai kontroliuoti visas finansų, logistikos ir įmonės valdymo sritis. Tai daugiakalbė sistema, leidžianti vartotojams vienu metu dirbti bei spausdinti ataskaitas keliomis kalbomis. Scala palengvina įmonės finansinį administravimą ir operatyvų planavimą. Visi moduliai yra suprojektuoti pagal vienodą modelį, ir tuo labai palengvinamas bei supaprastinamas sistemos instaliavimas bei vartotojų apmokymas. Finansai, logistika, gamybos bei projektų ir paslaugų (serviso) valdymas, darbo užmokestis - visos šios funkcijos yra įtrauktos į Scala kaip moduliai. Modulus galima gana laisvai pasirinkti. Sistemos finansinę dalį sudaro Didžioji, Pardavimo ir

Pirkimo knygos, kurias galima papildyti Ilgalaikio turto, Konsolidacijos ir Piniginių srautų planavimo funkcijomis. Scala logistika apima Atsargų valdymo, Pardavimo ir Pirkimo užsakymų apdorojimo funkcijas, kurias išplečia Statistikos modulis. Papildomai galima įsigyti Serijinių numerių ir Gaminų struktūrų kontrolės funkcijas. Darbo užmokesčio modulis yra pritaikytas Rytų ir Vidurio Europos šalių poreikiams.

Scala standartinių modulių sąrašas:

- Finansai
 - Didžioji knyga
 - Pardavimo knyga
 - Pirkimo knyga
- Logistika
 - Pardavimo valdymas
 - Pirkimo valdymas
 - Atsargų valdymas
 - Statistika
- Gamybos valdymas
 - Pagrindiniai gamybos duomenys
 - Gaminio savikainos skaičiavimas
 - Ilgalaikis planavimas
 - Medžiagų poreikio planavimas
 - Užsakymų planavimas
 - Cecho valdymas
- Kiti
 - Projektų valdymas
 - Paslaugų valdymas
 - Darbo užmokestis
 - Žmonių resursai
 - Tiesioginis sąskaitų išrašymas
 - Rinkos duomenų bazė

2.6.6 Scala projektų valdymo modulis

Scala sistemoje projektais laikomi įmonės vidaus darbai, pvz – sandėlių rekonstravimas. Kai veikla organizuojama pagal projektą, reikia kontroliuoti biudžetą ir sekti jo vykdymą. **Scala** Projektų valdymas kaip tik ir yra integruota priemonė, leidžianti sudaryti biudžetą, jį įvertinti, planuoti ir valdyti išteklius. Lanksti **Scala** projekto struktūra, apimanti iki keturių lygių, padeda valdyti projektą ir atskleisti jo vykdymo problemas.

- **Projektų apibrėžimas keturiais lygiais.** Scala sistemoje projektus galima suskirstyti į keturis atskirus lygius: Pagrindinį projektą, Projektą, Tarpinį projektą ir Darbą. Taip galima valdyti kiekvieną projektą, netgi jeigu atskiros jo dalys skiriasi savo pobūdžiu arba laikotarpiu. Pavyzdžiui, lengvai atskiriama projekto medžiagų naudojimo dalis nuo darbo laiko ir darbuotojų dalies. Galima sudaryti didesnio projekto biudžetą ir kontroliuoti tarpinius projektus. Tai labai palengvina darbą. Tarpiniai projektai atitinka tam tikrus bendro projekto struktūros sandus. Taip pat galima kontroliuoti pelną, pasirinkus kurią nors projekte naudojamą valiutą - vartotojo arba pirkėjo.
- **Pagrindinis projekto duomenų failas.** Pradiniai projekto duomenys - tai visų projekto sąlygų ir vykdymo terminų aprašai (pavyzdžiui, sąskaitų-faktūrų plano ir projekto kainų sudarymo), tarpinių projektų ir darbų, kurie įtraukiami į projektą, planai. Darbus galima grupuoti pagal jų terminus, medžiagų poreikius ir išlaidas. Planuojant taip pat galima naudoti numatomus įvykius. Planuojant numatoma, kas ką darys.
- **Projekto maketai.** Projektų valdymo modulyje yra įvairių projekto maketų, kad būtų galima lengviau įvesti naujų projektų duomenis. Vartotojas gali naudoti įvairių tipų maketus. Kiekvienas iš jų dar turi iki 100 alternatyvų. Kiekviename makete, pavyzdžiui, tokiaime, kaip Tarpinis projektas arba Darbas, įrašomi atitinkami duomenys ir tam tikros sąlygos, pavyzdžiui, pridėtinės vertės mokesčio normos. Projekto maketai taip pat apima informaciją apie nebaigtos gamybos ir savikainos skaičiavimo metodus ir duomenis apie įvairių dokumentų tipus ir jų spausdinimo eiliškumą. Tokia maketo informacija, kurią galima lengvai modifikuoti, naudojama įvedant naują projektą.
- **Sąskaitų planas.** Šis planas susideda iš kelių kainodaros modelių: sąskaitų-faktūrų pirminio ir pakartotinio išrašymo, fiksuotų ir aukščiausių kainų nustatymo ir kt. Vartotojas nurodo, kaip turi būti išrašomos sąskaitos-faktūros: jas galima išrašyti įvykus tam tikram įvykiui, galima nurodyti, ar projektas pradedamas įgyvendinti, ar jis sustabdomas, ar turėję būti atlikti darbai susiję su sąskaitų išrašymu. Sąskaita-faktūra gali būti išrašyta praėjus tam tikram laikui arba

nurodytą dieną. Vartotojas nustato sąskaitos-faktūros išrašymo režimą - esant reikalui arba periodiškai (kas savaitę, mėnesį ar pan.).

- **Biudžetai ir prognozės.** Scala Projektų valdymo modulis pasižymi savo funkcionalumu. Galima planuoti, sudaryti, įvertinti projekto biudžetą, įtraukti į apskaitą ir valdyti projekto išlaidas, modeliuoti įvairius jų įvertinimo scenarijus, keičiant kainų ir laiko parametrus. Suskirsčius projektą į dalis, galima stebėti ir analizuoti kelerių finansinių metų situaciją ir tendencijas, įvestas biudžeto alternatyvas palyginti su ankstesnėmis prognozėmis. Projektų valdymo modulis taip pat laiku suteikia informaciją apie išteklių naudojimą ir projekto darbų įvykdymo lygį.
- **Išteklių paskirstymas ir darbo grafikas.** Modulis numato išteklių paskirstymą, pavyzdžiui, vidaus darbams ir tarpinėms sutartims. Darbo ištekliai įvertinami pagal vartotojo duomenis. Vykdamas projektą galima kontroliuoti planines ir faktines laiko sąnaudas. Tai galima atlikti kompiuterio ekrane arba spausdinant reikiamas ataskaitas.
- **Vertybės.** Šio modulio dalis, susijusi su vertybių logistika, sąveikauja su Scala Pardavimo ir Atsargų valdymo moduliais. Jų funkcijas nustato pats vartotojas, rengdamas reikalingų vertybių planą. Siekiant aukštesnio kontrolės lygio, sąskaitas-faktūras galima išrašyti naudojant Projektų valdymo modulį.
- **Darbų sąnaudos.** Darbų sąnaudoms priklauso, pavyzdžiui, kelionių, apsirūpinimo, pridėtinės ir kitos panašios išlaidos. Galima nustatyti kiekvieno projekto išlaidų dydį.
- **Kitos sąnaudos.** Gali būti tokių atvejų, kai kurio nors projekto išlaidos įvedamos kituose Scala moduluose, pavyzdžiui, sąskaita už pietus su klientu įvedama į Didžiąją knygą. Projektų valdymo modulyje galima surasti visas tokias išlaidas ir perkelti jas į atitinkamą projektą.
- **Sąveika su kitais Scala moduliais.** Scala Projektų valdymo moduliui reikia Pardavimo knygos modulio. Norint pasiekti geresnių rezultatų, jis turi būti taip pat sujungtas su Didžiosios knygos ir Atsargų valdymo moduliais. Šie moduliai naudoja bendrų išteklių duomenų rinkmenas kartu su Aptarnavimo valdymo ir Darbo apmokėjimo moduliais. Projektų valdymo modulis leidžia sudaryti iki dešimties įvairių vartotojo duomenų bazių, apimančių įvairius projekto lygius. Duomenų bazėse taip pat galima saugoti įvairius papildomus projekto duomenis.

2.6.7 Gatavų produktų analizės apibendrinimas

Peržiūrėjus šių produktų siūlomus funkcionalumus matome, kiek daug ir įvairių funkcijų sukurta tam, kad automatizuoti ir pagerinti verslo procesus, vykstančius įmonėje. Tiek jau apžvelgtos, tiek kitos įmonės valdymui taikomos sistemos yra naudingos, kai tiksliai žinoma jų taikymo sritis, įmonės

poreikiai ir realios galimybės. Įmonės verslo proceso žinojimas - tai pirmas žingsnis į teisingą valdymo sistemos pasirinkimą.

Atlikus gatavų produktų apžvalgą matome, kad populiariosios sistemos (Navision, Scala) turi labai plačias galimybes, tačiau jų siūlomi funkcionalumai yra daugiau orientuoti į gamyba bei pirkimais arba pardavimais užsiimančių įmonių finansinę apskaitą. Nagrinėtos sistemos turi modulius, skirtus projektų valdymui, taip pat yra moduliai skirti žmoniškųjų resursų valdymui, bet Scala ir Navision naudojama projekto sąvoka kiek skiriasi nuo projekto sąvokos įmonėje, teikiančioje WEB paslaugas; skiriasi ir darbuotojų darbo laiko apskaitos vedimas, taip pat skiriasi kai kurie sistemos veikimo aspektai.

Dauguma atvejų sistemos veikia kliento – serverio principu, turi modulinę struktūrą, kas įgalina vėliau, esant reikalui, šias sistemas plėsti bei palengvina jų priežiūrą.

Taigi, remiantis anksčiau nagrinėtų sistemų pavyzdžiais, suformuluosime pradinis reikalavimus funkcionalumams, kuriuos turi teikti kuriama įmonės vidaus valdymo sistema.

Tai būtų:

- projektų valdymas su galimybe sekti projekto būseną ir atliktų darbų statusą,
- aktyvus informacijos apie projektus, darbuotojus ir klientus rinkimas ir palaikymas,
- darbuotojų veiklos valdymas įskaitant darbų grafikų (timesheet) pildymą,
- darbuotojų priskyrimas konkrečioms darbams,
- darbo laiko prie konkrečių darbų fiksavimas ir panašiai.

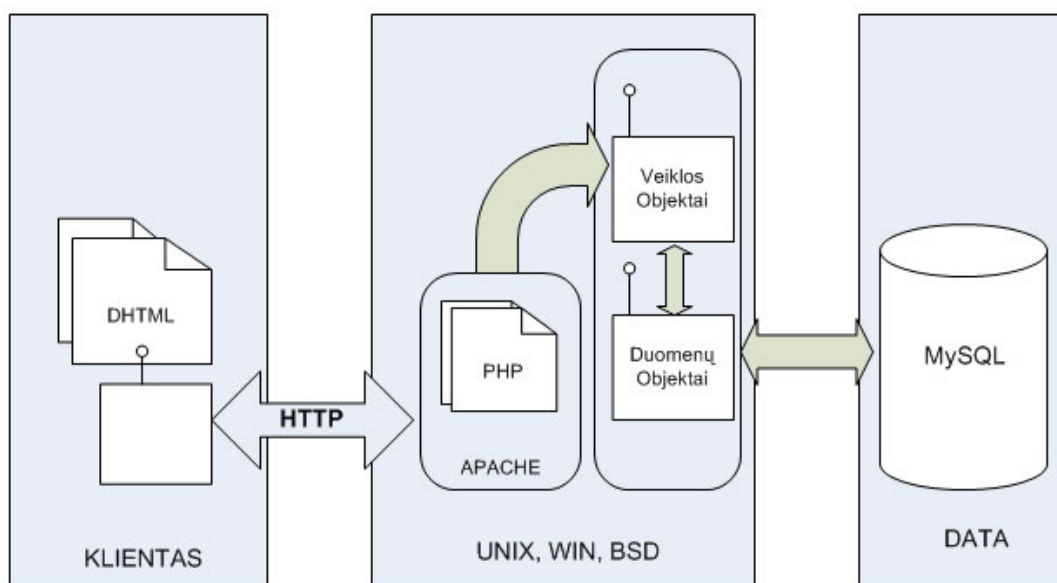
Smulkiau funkcionalumus panagrinėsime reikalavimų sistemai apibrėžimo srityje.

2.7 Architektūros ir galimų įgyvendinimo priemonių variantų analizė

Kaip ir daugelis produktų paremtų su WEB sąsaja veikimas yra panašus, Programai kreipiantis adresu per HTTP protokolą sugeneruojamas jam HTML dokumentas kurį supranta naršyklė. Generuojant dokumentą kliento pusėje duomenys apjomami PHP klasių metodais, funkcijomis. Esant reikalui per sistemoje esančią klasę galimi duomenų mainai su duomenų baze.

Norint pagerinti vartotojo sąsają, kuri gaunama kliento dalyje neapseinama be programų kliento dalyje. Tai visų pirma paprasčiausias pirminis klaidų apdorojimas, įvairūs automatiniai apskaičiavimai (darbo laiko sumavimas...)

Tai, kuo ši sistema skirsis nuo daugelio kitų, yra galimybė duomenis gauti iš serverio dalies net tada, kai puslapis yra pilnai užkrautas, ir tai atliekama be puslapio persikrovimo. Šitai padaryti įgalina realizacijoje panaudota AJAX programavimo metodika. Taip netik suteikiama daugiau galimybių vartotojo sąsajai, bet ir sumažinamas duomenų srautas. (Nereikalingas papildomas puslapio perkrovimas).



Pav. 3 Komponentinė sistemos architektūra, kai vartotojo sąsaja yra interneto naršyklė

2.7.1 Panaudotos realizavimo priemonės

Vienas iš pagrindinių aspektų, kodėl apsisprendėm daryti savo sistemą – kaina. Taigi atsižvelgiant į šį kriterijų, buvo stengiamasi, kad sistemos kūrimo ir palaikymo kaštai būtų kuo mažesni. Tuo tikslu buvo naudojamos nemokamos (atviro kodo) programinės realizacijos priemonės.

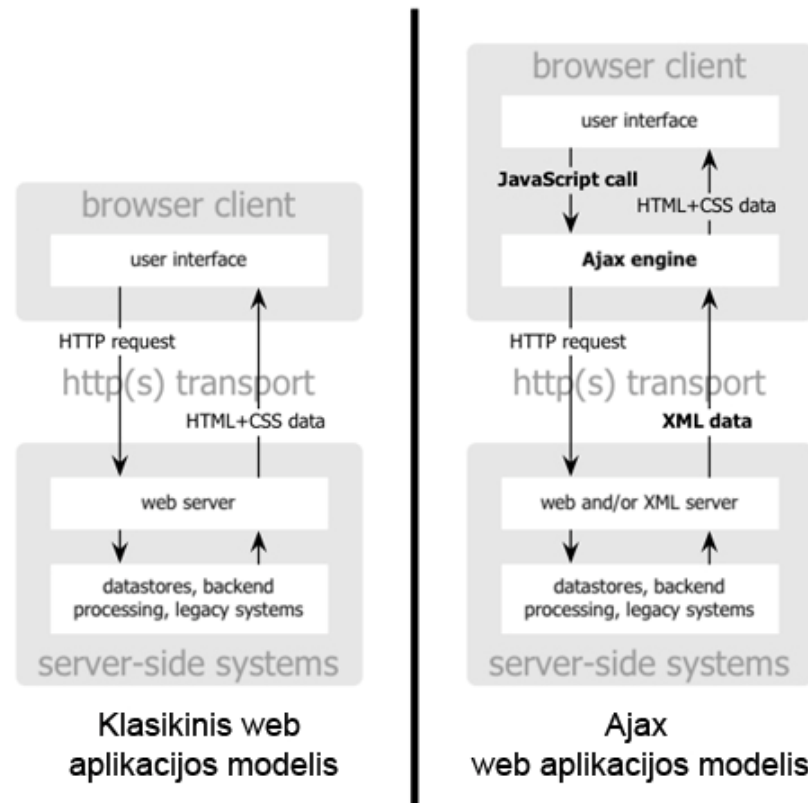
Lentelė 3 . Planuojamos panaudoti realizavimo priemonės

Sritis	Realizavimas
Serverio dalis	PHP
Kliento dalis	Java script
Duomenų bazė	MySQL
Serverio platforma	UNIX, arba BSD

Realizavime bus taikomas AJAX (Asynchronous JavaScript and XML) programavimas – tai WEB programavimo metodika, jungianti :

- XHTML (arba HTML) ir CSS informacijos pateikimui ir stilizavimui;
- Document Object Model, valdomą per JavaScript, skirtą dinamiškai pateikti ir sąveikauti su pateikta informacija.
- XMLHttpRequest objektus, skirtus keisti duomenimis asinchroniškai su WEB serveriu.(Paprastai naudojamas XML, nors veiktų bet kuris formatas, įskaitant performatuotą HTML, Plain text ir JSON).

Klasikinė web aplikacija veikia šitaip: dauguma vartotojo veiksmų klientinėje dalyje iššaukia http užklausą web serveriui. Serveris atlieka veiksmus – paima duomenis, apdoroja skaičius, kartais „pabendrauja“ su įvairiomis senomis sistemomis ir po to gražina html puslapį klientui. Šis realizacijos metodas techniškai geras, bet vartotojui nepatinka tai, kad kol serveris atlieka savo darbą, jam tenka laukti. Pav.4 galime pamatyti skirtumus tarp tradicinės ir Ajax web aplikacijos. Kuo Ajax web aplikacija yra kitokia? Ajax aplikacija pašalina iš sąveikos su serveriu poreikį laukti, kol serveris atliks savo darbą. Tai atliekama per tarpininką – Ajax varikliuką – esantį tarp vartotojo ir serverio. Nors ir atrodo, kad papildomo sluoksnio pridėjimas prie aplikacijos sulėtins jos darbą, bet taip nėra. Vietoj to, kad užkrautų internetinį puslapį, sesijos pradžioje naršyklė užkrauna Ajax varikliuką, kuris yra parašytas JavaScript kalba ir paprastai paslėptas nematomame freime. Šis varikliukas yra atsakingas tiek už vartotojo matomas sąsajos atvaizdavimą, tiek už bendravimą su serveriu iš vartotojo pusės. Ajax varikliukas leidžia vartotojo bendravimui su serveriu vykti asinchroniškai – nepriklausomai nuo sąveikos su serveriu. Tuo būdu vartotojas niekada nespokso į tuščią naršyklės langą ir laikrodžio ikoną, laukdamas kol serveris kažką padarys.



Pav. 4 Tradicinio web aplikacijų modelio palyginimas su Ajax aplikacijos modeliu

Kiekvienas vartotojo veiksmas, kuris paprastai sugeneruotų http užklausa, įgyja JavaScript užklauso Ajax varikliui formą. Bet kurią reakciją į vartotojo veiksmus, kuri nereikalauja kelionės atgal į serverį – tokius kaip paprastas duomenų patikrinimas, duomenų, esančių atmintyje redagavimas, ir netgi kai kuri navigacija – Ajax varikliukas įvykdo pats. Jeigu varikliukui reikia iš serverio kažko, kad galėtų tęsti darbą – jei tai duomenų apdorojimui pateikimas, papildomo sąsajos kodo užkrovimas arba naujų duomenų gavimas – varikliukas šias užklauso atlieka asinchroniškai, paprastai naudodamas XML, nevilkindamas vartotojo sąveikos su aplikacija.

2.7.2 Programavimo kalba, duomenų bazė

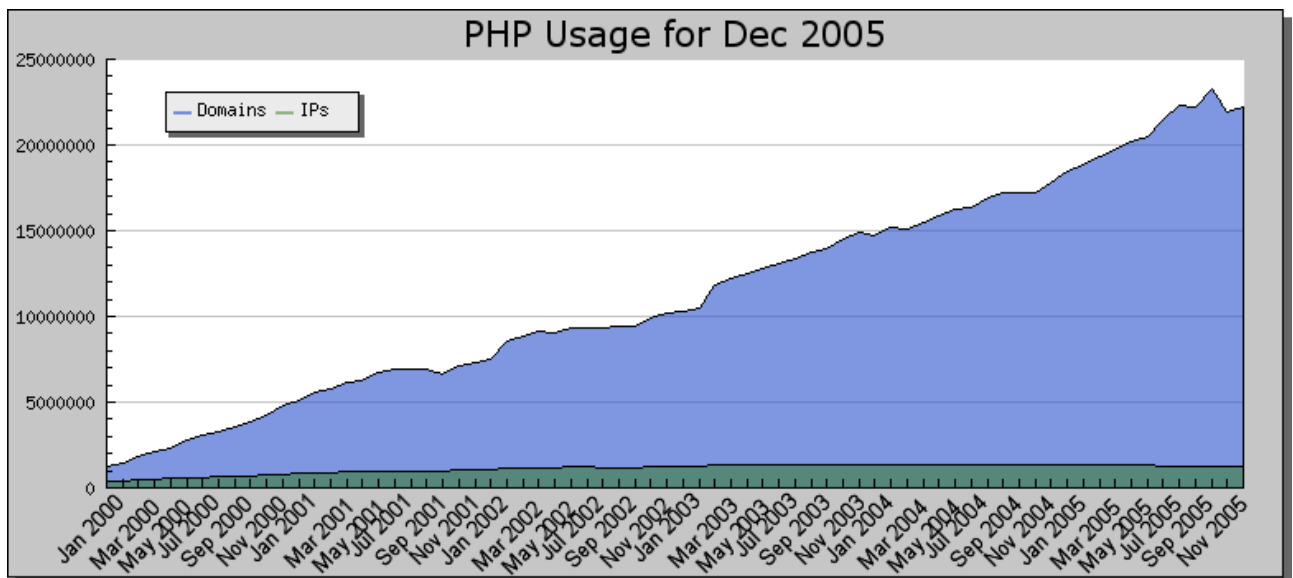
Kodėl pasirinkta PHP?

Pagrindine kalba buvo pasirinkta PHP, įvertinus galimybes ir daugelį faktorių, išanalizavus palyginimus. Toliau pateikiame esminius privalumus, nulėmusius pasirinkimą:

- PHP yra plačiai paplitusi ir yra sukurta daug nemokamų priedų.
- PHP yra atviro kodo projektas, todėl jį kuria didelė grupė žmonių. Tai leidžia pastebėtas klaidas ištaisyti labai greitai, bei plėtoti pačią kalbą.
- PHP veikia įvairiose operacinėse sistemose: Windows, *nix, MacOS, Solaris, HP-UX ir t.t.

- Turi galimybę naudoti „pagreitinimo“ sistemą, kai dalis vykdomo kodo paliekama „bitų kodo“ pavidalu.
- Leidžia užkoduoti programinį kodą.
- PHP gali veikti su daugeliu WEB serverių: Apache, IIS, PWS, OmniHTTP, BadBlue ir t.t.
- Išmokti PHP programavimo pagrindus yra labai lengva, o kadangi paruošta plati dokumentacija, atsakymus ras ir profesionalas.
- Pasižymi dideliu greičiu serverio pusėje, bei dideliu greičiu dirbant su duomenų bazėmis.
- Sukurta daug papildomų paketų PHP kalbos interpretatoriui.

PHP pasirinkimą taip pat lemia didelis šios programavimo kalbos populiarumas: 22 172 983 svetainių, 1 277 375 unikalių IP adresų (2005 metų gruodis). Kalbos panaudojimo atvejų progresas pavaizduotas grafiškai **Error! Reference source not found.** paveikslėlyje.



Pav. 5 PHP naudojimo statistika

MySQL duomenų bazių sistema

MySQL yra viena iš populiariausių atviro kodo duomenų bazių sistemų. Kuriama ir prižiūrima MySQL AB kompanijos, kuri MySQL sistemą platina pagal dvigubą licencijavimo sistemą. Pati licencijavimo sistema apibūdinama sakiniu: „Jei platinate MySQL nemokamai, tai ir ją gaunate nemokamai, tačiau, jei ją platinate ir už ją imate pinigus, tai jūs taip pat turėtumėt sumokėti mums už licenziją“. Kadangi informacinė sistema bus platinama atskirai be MySQL sistemos programinio kodo, tai už MySQL naudojimą nereikia mokėti. Taip gaunama patikima, greita bei nemokama duomenų bazių valdymo sistema.

MySQL privalumai:

- Platinama kaip atviro kodo sistema
- Bandomosios versijos yra laisvai platinamos:
 - randamos pagrindinės klaidos, kurias pataisomos iki paskelbiant versiją stabilia
 - vartotojų bendruomenė gali susipažinti su naujomis savybėmis, dar prieš išleidžiant naują versiją

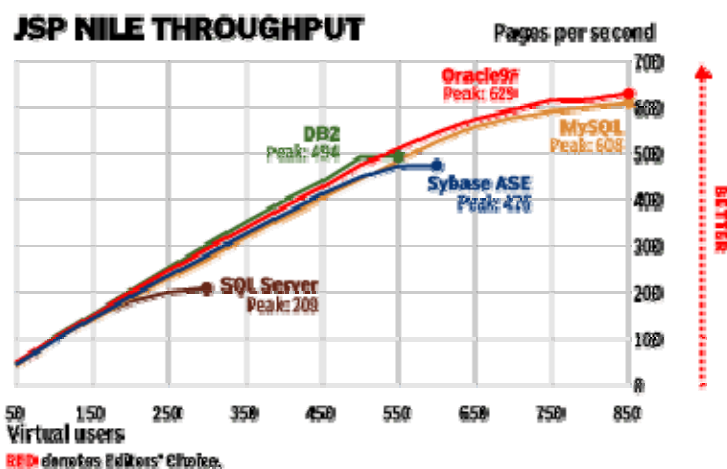
MySQL privalumai verslo vartotojams:

- Sumažėja išlaidos už naudojimosi licenziją (apie 90%)
- Padidėja sistemos nepertraukiamo darbo laikas (apie 60%)
- Mažesnės kompiuterio sistemos sąnaudos (apie 70%)
- Sumažėja išlaidos administravimui, programavimui bei priežiūrai (apie 50%)
- Sistema nėra tokia sudėtinga, kaip kad konkurentų siūlomos sistemos Oracle, IBM DB2, ir Microsoft SQL Server

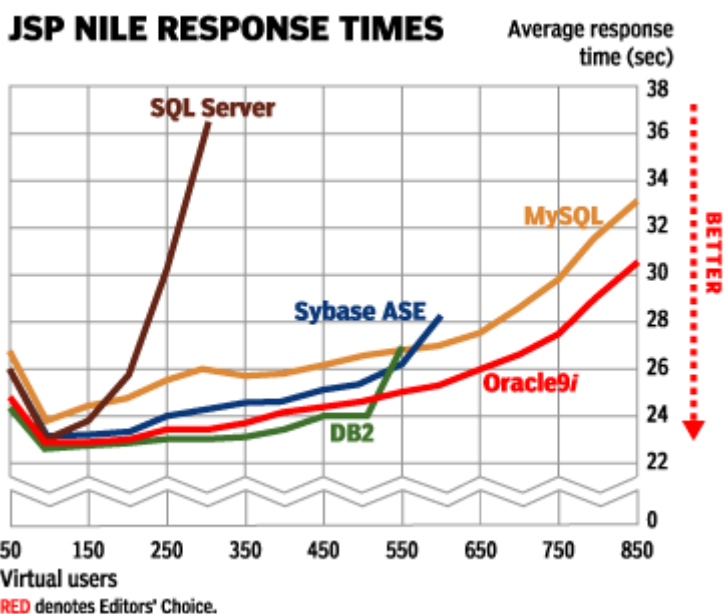
MySQL duomenų bazių sistema ne ką tenusileidžia gerai žinomoms sistemoms kaip Oracle, DB2 ir SQL server.

Buvo atliktas testas, kur testuotos sistemos: IBM's DB2 7.2 su FixPack 5, Microsoft Corp.'s SQL Server 2000 Enterprise Edition su Service Pack 2, MySQL AB's MySQL 4.0.1 Max, Oracle Corp.'s Oracle9i Enterprise Edition 9.0.1.1.1 and Sybase Inc.'s ASE (Adaptive Server Enterprise) 12.5.0.1. Testuota buvo emaliojant nuo 50 iki 2000 vartotojų, dirbančių online režimu su pateikta internetine sistema, kurią turėjo aptarnauti duomenų bazių valdymo sistemos.

Sistemų įvertinimai grafiškai parodytas 6 ir 7 paveikslėliuose:



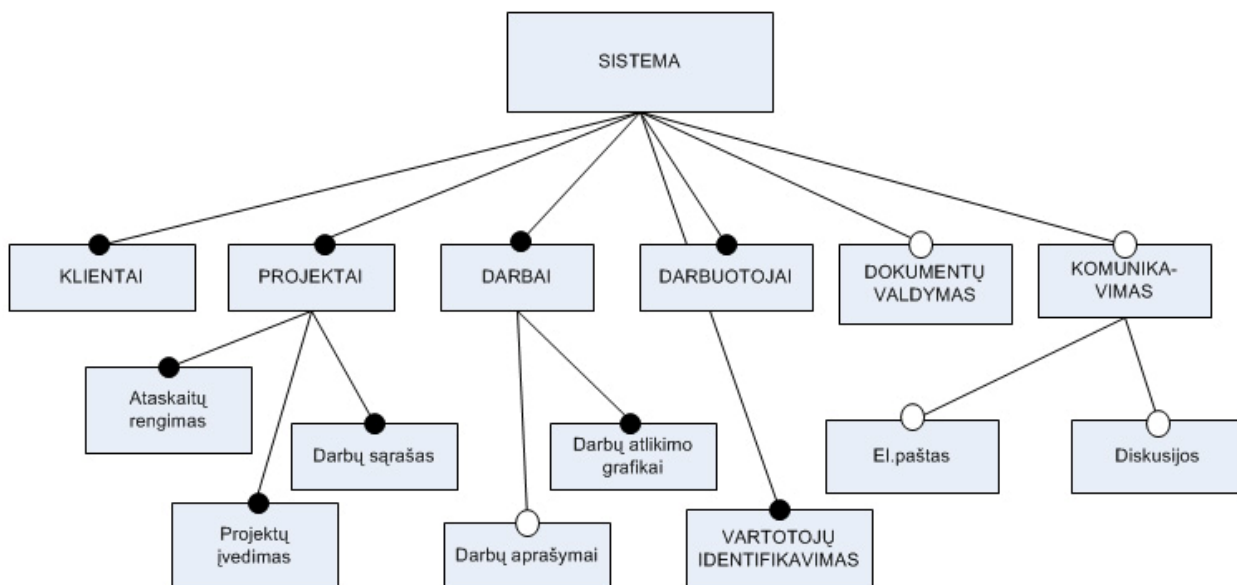
1 pav. Duomenų bazių palyginimas pagal apkrautumą



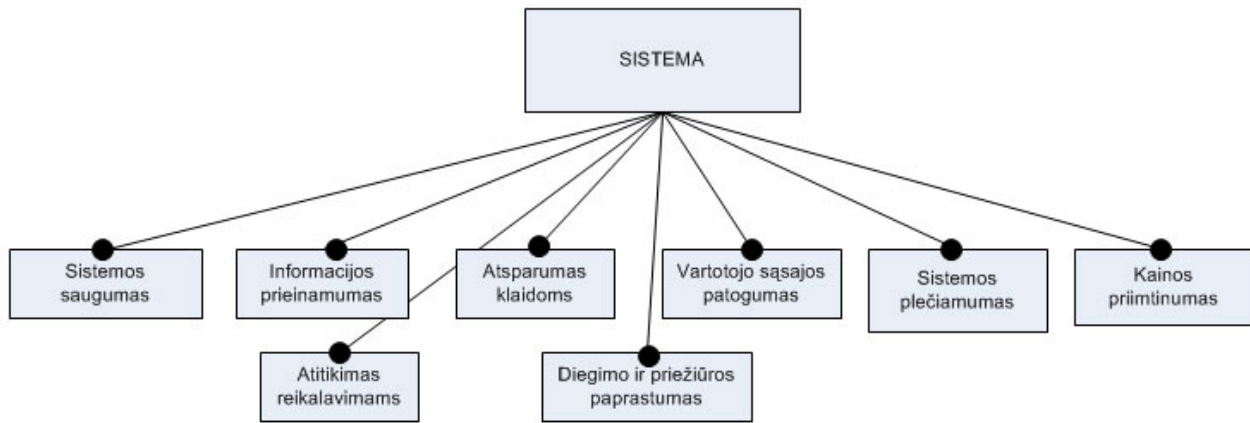
2 pav. Duomenų bazių palyginimas pagal atsakymo laiką

2.8 Sukurtos sistemos apibrėžimas

Tam, kad galėtumėm išskirti konkrečius reikalavimus sistemai, pirmiausia reikėjo išsiaiškinti, kokiomis savybėmis turėtų pasižymėti mūsų projektuojama sistema. Šias savybes suskirstėme į būtinas ir nebūtinias ir pavaizdavome jas kaip funkcinį ir nefunkcinį savybių medžius. Būtinias savybes pažymėjome juodu apskritimu, nebūtinias - baltu.



Pav. 6 Kuriamos sistemos funkcinį savybių medis



Pav. 7 Kuriamos sistemos nefunkcinių savybių medis

Kodėl konkrečios savybės išskirtos kaip būtinos? Valdymo sistema buvo projektuojama atsižvelgiant į įmonėje vykstančius verslo procesus. Verslo procesas susideda iš tam tikrų darbų, kurie privalo būti realizuoti, tam kad gaminamas produktas ar teikiama paslauga būtų įgyvendinta. Taigi, tie darbai, be kurių gaminamas produktas ar teikiama paslauga negali būti realizuoti, ir yra būtinos sistemos savybės.

2.9 Darbo tikslas ir siejami privalumai

Šio darbo tikslas buvo:

Suprojektuoti ir sukurti sistemą, kuri tenkintų konkrečius įmonės vidaus valdymo poreikius, būtų patogi naudoti, lengvai plečiama ir pritaikoma;

Nors šiame darbe kuriamos darbų valdymo sistemos pirmtakas jau egzistuoja toje įmonėje, tačiau laikui bėgant atsirado naujų funkcionalumų poreikis. Naujoji sistema sukurta **nuo pat pradžių**, pasiremiant senosios sistemos idėjomis, tačiau architektūros, vartotojo sąsajos ir kai kuriais kitais atžvilgiais padarytos kiek kitaip.

Buvo siekiama:

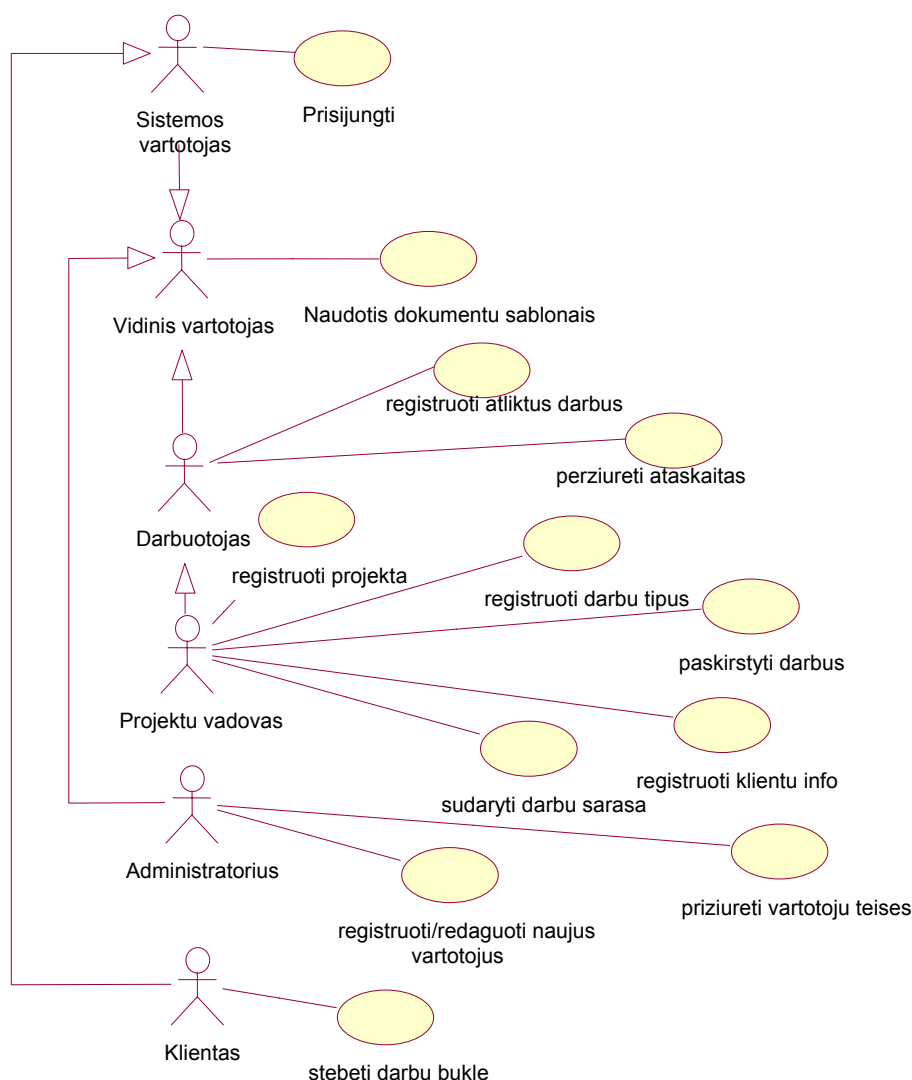
- padaryti vartotojo sąsają kiek įmanoma panašią į Windows operacinės aplinkos sąsają (langai, meniu , išdėstymas ir pan.), kad vartotojui ši sistema kuo mažiau skirtųsi nuo jam įprastų Windows programų.
- Architektūra perprojektuoti taip, kad kai kurias operacijas iš serverio pusės perkelti į kliento pusę, ir su serveriu komunikuojant gauti tik reikalingus duomenis, tuo būdu išvengiant bendros sąsajos perkrovimo tuo metu, kai atnaujinami ar pateikiami kiti duomenys.
- Realizuoti efektyvų darbų paskirstymą (ToDo list), kad kiekvienas projektas būtų išskaidomas į optimalų darbų skaičių ir realizacinis personalas bei kiti įmonės resursai būtų teisingai užimti
- Patobulinti atliktų darbų ataskaitos (Timesheet) pildymą, kad darbuotojams būtų patogiau fiksuoti išdirbtą prie konkretaus projekto laiką.

- Numatoma realizuoti apribotą priėjimą prie sistemos iš išorinių tinklų, nes pastebėta, jog kai kuriems įmonės darbuotojams, nesantiems fiziškai įmonės ribose, šios sistemos teikiami duomenys ir galimybės vis tiek yra labai reikalingi.
- Kt..

Ši sistema (beje, jos pirmtakas taip pat) patogi tuo, kad jos nereikia instaliuoti kiekviename įmonės kompiuteryje. Ji sukonfigūruojama serveryje, o vartotojams reikia tiesiog naršyklės pagalba prisijungti prie jos. Tai labai palengvina jos priežiūrą.

2.10 Kompiuterizuojamos sistemos funkcijos

Kompiuterizuojamos sistemos funkcijas pateiksime grafiškai, pasinaudodami panaudojimo atvejų diagrama:



Pav. 8 Sistemos panaudojimo atvejų diagrama.

Trumpas diagramos paaiškinimas:

Bet kuris sistemos vartotojas gali prisijungti prie sistemos, naudotis elektroniniu paštu bei diskusijomis. Vienas iš tokių vartotojų tipų yra klientas, kuriam suteikiama papildoma teisė stebėti darbų būklę. Vidiniai sistemos vartotojai gali naudotis esamais dokumentų šablonais, o administratorius (specifinis vidinis vartotojas) gali prižiūrėti vartotojų teises, taip pat registruoti naujus vartotojus, redaguoti jų informaciją. Darbuotojai turi galimybę registruoti atliktus darbus. Labiausiai privilegijuotas darbuotojų tipas – projekto vadovas. Jis registruoja projektus, darbų tipus, sudaro darbų sąrašus ir atlieka darbų paskirstymą, redaguoja kliento informaciją.

2.11 Reikalavimai duomenims

Siekiant užtikrinti, kad sistema funkcionuos efektyviai ir visuomet pateiks tikslią informaciją, reikalinga rinkti duomenis apie klientus, darbuotojus, projektus, užduotis bei darbus. Toliau pateiksime sąrašą duomenų, kurie privalo būti renkami ir komentarus, kodėl tas turi būti daroma:

- apie klientus turi būti surenkama tokia informacija:
 - kliento firmos pavadinimas
 - atstovaujančio asmens vardas ir pavardė
 - kliento adresas, telefonas bei el. pašto adresas
 - įmonės kodas
 - banko rekvizitai

Šie duomenys surenkama tam, kad būtų žinoma visa reikalinga kontaktinė informacija apie klientą, kuri vėliau panaudojama sudarinėjant sutartis, informuojant apie darbų eigą ir pan.

- apie darbuotojus renkami tokie duomenys:
 - vardas, pavardė
 - telefono nr., el. pašto adresas
 - asmens kodas
 - pareigos
 - vartotojo grupė

Sistemoje privalo būti tokie duomenys apie visus dirbančius darbuotojus, nes atsižvelgiant į pareigas darbuotojui skiriami darbai; atsižvelgiant į jam priskirtą vartotojo grupę – suteikiamos priėjimo prie sistemos resursų teisės, o asmeninė informacija panaudojama algalapio formavime.

- apie projektą turi būti renkama tokia informacija:
 - projekto pavadinimas
 - projekto tipas
 - kodas
 - apibūdinimas
 - projektą sudarančių užduočių sąrašas
 - užduotis vykdantys darbuotojai
 - užduoties atlikimo rodiklis (procentais)
 - užduotį sudarančių darbų sąrašas
 - darbų pradžios rodiklis
 - darbų trukmės rodiklis
 - komentarai (tiek apie užduotis, tiek apie atskirus darbus)

Projektai yra visos sistemos darbo ašis, todėl siekiant didesnio efektyvumo ir skaidrumo, turi būti nuolat kaupiama aukščiau išvardinta informacija apie projektus. Tuo būdu visada matome, kokia yra projekto būklė (bendras darbų atlikimo procentas), kokie darbai projektą sudaro (užduočių ir darbų sąrašas), kurie darbuotojai prie jų dirba ir kiek laiko sugaištama tiems darbams įgyvendinti. Komentarų pagalba galima išsaugoti papildomą informaciją apie darbus bei užduotis.

Sistema realizuota taip, kad informaciją apie projekto darbų atlikimą pildyti turi kiekvienas darbuotojas individualiai, tam jam pateikiama speciali aplinka, kurioje darbuotojas pildo savo atliktų darbų grafiką pasirinkdamas jam priskirtus darbus bei fiksuodamas prie tų darbų sugaištą laiką. Vėliau, remiantis sukaupta informacija, skaičiuojami darbuotojų atlyginimai, projekto įvykdymo statistika, bendra darbų trukmės bei pasiskirstymo statistika ir pan.

- vartotojų grupės aprašanti informacija turi būti tokia;
 - vartotojų grupės pavadinimas
 - galimos teisės
 - pagal galimas teises prieinami resursai

Ši informacija turėtų būti suvedama iš karto po įdiegimo, kadangi aprašant darbuotojus jie jau turi turėti jiems priskirtas konkrečias teises. Tą funkciją privalo atlikti sistemos administratorius, nes jis turi daugiausia galimybių teikiančias vartotojo teises.

2.12 Nefunkciniai reikalavimai ir apribojimai

- Saugumas- turi būti garantuojama duomenų apsauga. Taip pat turi būti įgyvendintas patikimas vartotojų identifikavimas;
- Prieinamumas kainos atžvilgiu;
- Išmokstamumas – turi būti lengvai perprantama ir vartotojas turi greitai išmokti ją naudotis;
- Atnaujinamumas – turi būti suprojektuota taip, kad nauji priedai ir naujos funkcijos būtų lengvai pridedamos;
- Įdiegimo savybės – turi būti kuo paprasčiau;
- Vartotojo sąsajos patogumas – vartotojo sąsaja turi būti intuityvi ir patogi, kad vartotojui reiktų kuo mažiau laiko prie jos priprasti;
- Atsparumas klaidoms – esant gedimui kurioje nors dalyje, turi būti sudaromos galimybės kuo greičiau surasti ir pašalinti nesklandumus;
- Diegimo ir palaikymo paprastumas – sistema neturėtų reikalauti ypatingai daug dėmesio ir resursų ją diegiant, turi būti paprasta ją prižiūrėti;

2.13 Rizikos faktorių analizė

1. Programų sistema gali tapti nevaldoma. Gali smarkiai išaugti sistemos apimtis ir pasidaryti sudėtinga jos priežiūra. Sistemos plėtimas gali būti sudėtingas.

Sprendimo būdas: projektuoti modulinę architektūrą, taikyti architektūros valdymui produktų savybių modelį, dokumentuoti modulius, susieti su reikalavimais, išsaugoti meta duomenis apie kiekvieną programinį modulį.

2. Sistema gali būti nepriimtina vartotojams.

Sprendimo būdas: kurti sistemą pagal naudojamumo kriterijų.

3. Priklausomybė nuo interneto paslaugų tiekėjo. Jeigu įmonės serveris ne įmonės ribose, dingus ryšiui sistema gali būti nefunkcionala.

Sprendimo būdas: sistemą aptarnaujančius serverius statyti įmonės patalpose, pasirūpinti atsarginių duomenų saugojimu (backup) jei su pagrindine duomenų baze kas nors atsitiktų.

2.14 Rezultato kokybės kriterijai

Pirma – sistemos naudojamumas (usability) - sistema turi gerai tikti visų tipų vartotojams, neapkrauti jų papildomu darbu. Vartotojo sąsaja turi būti intuityvi, optimali.

Antra – tolesnis sistemos plečiamumas. Šitai užtikrinama suprojektuojant modulinę architektūrą, kuri įgalina sistemos kūrėjus ją tobulinti papildant naujais funkciniais moduliais ir galimybėmis.

2.15 Analizės išvados

1. Atlikus įmonės vidaus valdymo kompiuterizavimo poreikių ir esamų įmonių resursų planavimo sistemų analizę, nustatyta, kad vienas iš svarbiausių įmonės vidaus valdymo arba sistemos pasirinkimo aspektų yra detalus vykstančio verslo proceso išmanymas ir konkrečių poreikių žinojimas. Nagrinėti įmonių vidaus valdymo programų paketai mažai atitinka šiuos poreikius ir turi daug perteklinių savybių.
2. Kadangi dirbama konkrečioje įmonėje, su ten vykstančiais procesais susiduriama kiekvieną dieną, nuspręsta, kad geriausia tokią sistemą susikurti patiems. Tai būtų naudinga ir kainos atžvilgiu, ir programinė įranga būtų pritaikyta poreikiams, ir programuotojas įgytų vertingos patirties.
3. Buvo apžvelgtos Lietuvoje žinomų panašaus pobūdžio sistemos, jų teikiami funkcionalumai, jose panaudoti principai valdymo sprendimams įgyvendinti. Pasisemta idėjų, kaip kokius verslo procesus reikėtų automatizuoti.
4. Remiantis atlikta vartotojų poreikių ir panašių programinių produktų analize, buvo sudarytas sistemos funkcijų sąrašas, kuriame pagrindinės funkcijos – projektų darbų paskirstymas ir jų vykdymo sekimas, darbuotojų bei klientų informavimas.
5. Nagrinėta literatūra, siūlanti sprendimo metodus siekiant efektyvaus architektūros projektavimo, konkrečiai – savybių modelis, kurio pagalba pavaizduotos būtinos ir nebūtinos sistemos savybės.
6. Reikalavimai šiai sistemai buvo suformuoti nagrinėjant jau egzistuojančią sistemą, siekiant jos patobulinimo.
7. Sudaryta projektavimo užduotis sukurti sistemą, kurioje būtų kompiuterizuojamos projektų valdymo, projekto užduočių sąrašo sudarymo, užduočių darbuotojams paskirstymo, darbuotojų atlikto darbo fiksavimo bei statistikos apie projekto užduočių atlikimą rinkimo funkcijos. Taip pat šioje sistemoje turi būti realizuotas vartotojų teisių valdymas, įvairių ataskaitų generavimas.

3 Įmonių vidaus valdymo sistemos reikalavimai

3.1 Reikalavimų specifikacija

Apibrėždami sistemos reikalavimus pateikėme panaudojimo atvejų diagramą (Pav.7), o dabar pateiksime detalias specifikacijas kiekvienam panaudojimo atvejui. Taip pat pateiksime keletą veiklos diagramų, iliustruojančių sudėtingesnių funkcijų veikimą.

Lentelė 4. Prisijungimo funkcijos specifikacija

Prisijungimas	
Tikslas	Identifikuoti sistemos vartotojus
Aktorius (-iai)	Klientai, darbuotojai, administratoriai, įmonės valdžia
Ryšiai su kitais panaudojimo atvejais	-
Nefunkciniai reikalavimai	Saugumas
Prieš sąlygos	-
Sužadinimo sąlyga	Kreipiamasi į sistemą
Po sąlyga	Vartotojas prijungiamas prie sistemos
Pagrindinis scenarijus	Vartotojas įveda savo prisijungimo vardą ir slaptažodį
Alternatyvūs scenarijai	Bandantis prisijungti asmuo nėra įregistruotas kaip sistemos vartotojas, prisijungimas neleidžiamas

Lentelė 5. Dokumentų šablonų naudojimas

Dokumentų šablonų naudojimas	
Tikslas	Kaupiami ir prireikus greitai pateikiami reikiamų dokumentų pavyzdžiai
Aktorius (-iai)	Vidiniai sistemos vartotojai (darbuotojai, projektų vadovai)
Ryšiai su kitais panaudojimo atvejais	-
Nefunkciniai reikalavimai	Aiškumas, naudojimo paprastumas, prieinamumas
Prieš sąlygos	Dokumento pavyzdys turi būti įvestas į šią bazę
Sužadinimo sąlyga	Vartotojas užklausia kokio nors dokumento
Po sąlyga	Dokumento pavyzdys pateikiamas vartotojui
Pagrindinis scenarijus	Vartotojas užklausia sistemos reikiamo pavyzdžio, sistema jam pateikia prašomo dokumento šabloną.
Alternatyvūs scenarijai	Sistemoje dokumento šablono nėra, vartotojui šablonas nepateikiamas.

Lentelė 6 . Klientų informacijos registravimo funkcijos specifikacija

Klientų informacijos registravimas	
Tikslas	Rinkti, kausti ir pateikti informaciją apie klientus
Aktorius (-iai)	darbuotojai
Ryšiai su kitais panaudojimo atvejais	-
Nefunkciniai reikalavimai	Saugumas, priėjimas bet kuriuo metu
Prieš sąlygos	Ieškantis informacijos vartotojas turi turėti tą leidžiančias teises
Sužadinimo sąlyga	Kreipiamasi į sistemą
Po sąlyga	Pateikiama reikalinga informacija
Pagrindinis scenarijus	Pirmiausia identifikuojamas klausiantysis asmuo, po to surandamas klientas pagal projektą, pagal pavadinimą arba pagal ID, pasirenkamas veiksmas ką su ta informacija daryti (ataskaita, panaudojimas kitur, keitimas), atliekami pasirinkti veiksmi
Alternatyvūs scenarijai	Klausiantysis vartotojas neturi teisės peržiūrėti/keisti informaciją, ieškomas klientas nerastas. Informacija nepateikiama

Lentelė 7. Atliktų darbų registravimo funkcijos specifikacija

Atliktų darbų registravimas	
Tikslas	Fiksuoti kiekvieno darbuotojo sugaištą laiką prie konkretaus darbo, rinkti statistiką apie atliekamų darbų trukmę.
Aktorius (-iai)	darbuotojai
Ryšiai su kitais panaudojimo atvejais	Informacijos apie darbų tipus teikimas
Nefunkciniai reikalavimai	Naudojimo patogumas, patikimumas, apsauga nuo sukčiavimų
Prieš sąlygos	Darbuotojas turi būti priskirtas konkrečiam darbui
Sužadinimo sąlyga	Darbuotojas kreipiasi į sistemą
Po sąlyga	Įvedamas konkretus darbas bei išdirbtas prie jo laikas
Pagrindinis scenarijus	Darbuotojas prisijungia prie sistemos, įveda rankomis arba pasirenka iš pateikto sąrašo darbo tipą ir laiką, kiek prie jo pradirbo. Sistema pateikia darbo dienos laiką, suskaidytą po vieną valandą. Esant reikalui, vieną valandą galima skelti į 4 dalis po 15 minučių.
Alternatyvūs scenarijai	-

Lentelė 8 . Projekto registravimo funkcijos specifikacija

Registruoti projektą	
Tikslas	Registruoti projektą ir stebėti būklę, rinkti statistinę informaciją apie atlikimo terminus, resursų poreikius ir pan.
Aktorius (-iai)	Projektų vadovas, kitos sistemos
Ryšiai su kitais panaudojimo atvejais	Time sheet pildymas, todo sąrašo pildymas, klientų ir darbuotojų informacijos rinkimas
Nefunkciniai reikalavimai	Naudojimo paprastumas, greitas informacijos atnaujinimas, patikimumas
Prieš sąlygos	-
Sužadinimo sąlyga	Atliekamas konkretus darbas, susijęs su projektu, projekte atsiranda naujų darbų, prie projekto priskiriami nauji darbuotojai ir pan..
Po sąlyga	Informacija apie konkretų įvykį išsaugota.
Pagrindinis scenarijus	Automatiškai turi būti fiksuojama, į kiek darbų projektas padalintas, kokie darbuotojai jam priskirti, kiek laiko sugaišta konkrečių darbų atlikimui, kokie pokyčiai padaryti ir pan..
Alternatyvūs scenarijai	Esant kurios nors susijusios sistemos dalies gedimui tuoj pat formuojami pranešimai apie klaidas.

Lentelė 9 . Darbų sąrašo funkcijos specifikacija

Sudaryti darbų sąrašą	
Tikslas	Išskaidyti projektą į atskirus darbus, priskirti tuos darbus konkrečioms darbuotojams.
Aktorius (-iai)	Projektų vadovas
Ryšiai su kitais panaudojimo atvejais	-
Nefunkciniai reikalavimai	Patikimumas, greitas informacijos atnaujinimas
Prieš sąlygos	Turi būti inicijuotas naujas projektas
Sužadinimo sąlyga	Projektas patvirtinamas ir perduodamas realizavimui
Po sąlyga	Projektas išskaidytas į atskirus darbus, prie kurių priskirti konkretūs darbuotojai
Pagrindinis scenarijus	Inicijuotas projektas perduodamas realizavimui, projektų vadovas išskaido projektą į atskirus darbus, atsižvelgdamas į darbuotojų kvalifikaciją ir užimtumą, priskiria juos darbams.
Alternatyvūs scenarijai	Darbuotojai, esant mažesniai užimtumui, gali ir patys užsiimti koki nors darbu. Esant veikimo klaidoms, turi būti formuojami pranešimai ir stengiamasi kuo greičiau pašalinti trikdžius.

Lentelė 10. Darbų paskirstymo funkcijos specifikacija

Paskirstyti darbus	
Tikslas	Paskirstyti kiekvieno projekto darbus konkrečioms darbuotojams
Aktorius (-iai)	Projekto vadovas
Ryšiai su kitais panaudojimo atvejais	Sudaryti darbų sąrašą
Nefunkciniai reikalavimai	Naudojimo patogumas, patikimumas, apsauga nuo sukčiavimų
Prieš sąlygos	Turi būti sudarytas konkretaus projekto darbų sąrašas
Sužadinimo sąlyga	Projektų vadovas kreipiasi į sistemą
Po sąlyga	Darbas priskirtas darbuotojui
Pagrindinis scenarijus	Projektų vadovas pagal užimtumą ir kvalifikaciją priskiria darbą darbuotojui.
Alternatyvūs scenarijai	-

Lentelė 11. Darbų tipų registravimo funkcijos specifikacija

Registruoti darbų tipus	
Tikslas	Kaupti informaciją apie dažnai pasitaikančius specifinius darbus, palengvinti time sheet pildymą.
Aktorius (-iai)	Darbuotojai
Ryšiai su kitais panaudojimo atvejais	Time sheet pildymas
Nefunkciniai reikalavimai	Aiškumas, lengvas prieinamumas
Prieš sąlygos	-
Sužadinimo sąlyga	Pildant time sheet, tam tikros rūšies darbas kartojasi keletą kartų. Darbai klasifikuojami pagal savo specifiką.
Po sąlyga	Besikartojantis darbas užsaugomas tam skirtame sąraše.
Pagrindinis scenarijus	Pildydamas time sheet, darbuotojas koki nors dažnai pasitaikantį darbą užsaugo tam skirtame sąraše. Prie to darbo taip pat saugomas jo tarifas. Vėliau pildant time sheet tą darbą galima pasirinkti iš pateikiamo sąrašo.
Alternatyvūs scenarijai	-

Lentelė 12. Vartotojų teisių stebėjimo funkcijos specifikacija

Vartotojų teisių stebėjimas	
Tikslas	Valdyti sistemos vartotojų teises
Aktorius (-iai)	Administratorius
Ryšiai su kitais panaudojimo atvejais	-
Nefunkciniai reikalavimai	Saugumas, naudojimo patogumas, greitas informacijos atnaujinimas
Prieš sąlygos	-
Sužadinimo sąlyga	Administratorius kreipiasi į sistemą
Po sąlyga	Pakeičiamos vartotojų teises, suteikiama reikiama informacija
Pagrindinis scenarijus	Administratorius prisijungia prie sistemos, suranda reikiamą vartotoją pagal jo ID, pavardę ar pan.. ir atlieka reikiamus veiksmus
Alternatyvūs scenarijai	-

Lentelė 13. Naujo vartotojo registravimo funkcijos specifikacija

Naujo vartotojo registravimas	
Tikslas	Užregistruoti naują sistemos vartotoją ir suteikti jam tam tikras teises
Aktorius (-iai)	Administratorius
Ryšiai su kitais panaudojimo atvejais	Darbuotojų informacijos registravimas
Nefunkciniai reikalavimai	saugumas
Prieš sąlygos	Naujas vartotojas turi būti arba įmonės darbuotojas, arba klientas, kuriam turi būti suteikta teisė stebėti projekto vykdymo eigą
Sužadinimo sąlyga	Administratorius kreipiasi į sistemą
Po sąlyga	Naujas vartotojas užregistruotas
Pagrindinis scenarijus	Registracijos formoje fiksuojama visa reikalinga informacija apie vartotoją, suteikiamas jam prisijungimo vardas, slaptažodis ir tam tikros teisės.
Alternatyvūs scenarijai	-

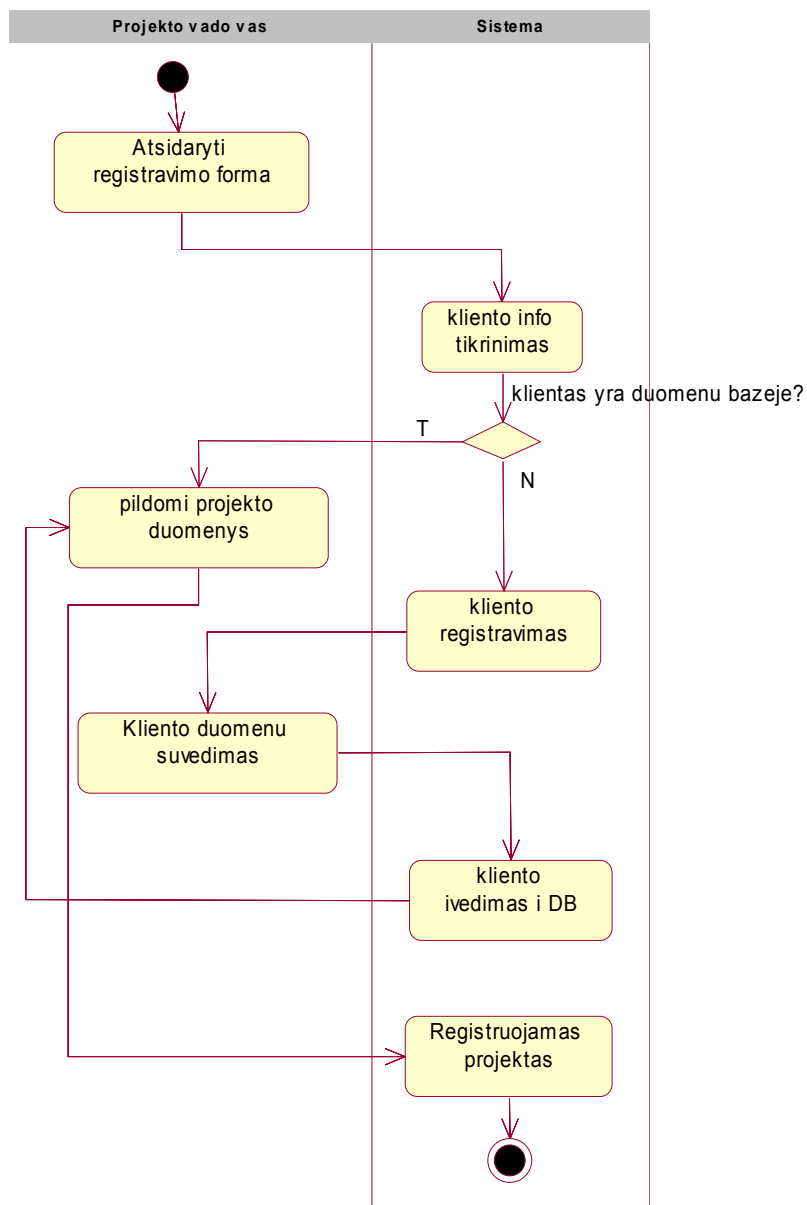
Lentelė 14. Darbų būklės stebėjimo funkcijos specifikacija

Darbų būklės stebėjimas	
Tikslas	Teikti operatyvią ir išsamią informaciją apie atliekamus darbus
Aktorius (-iai)	klientai
Ryšiai su kitais panaudojimo atvejais	-
Nefunkciniai reikalavimai	Greitas informacijos atnaujinimas, patikimumas, saugumas, prieinamumas
Prieš sąlygos	Vartotojas turi turėti teisę stebėti tam tikro projekto būklę
Sužadinimo sąlyga	Pateikiama užklausa sistemai
Po sąlyga	Pateikiama prašoma informacija esamam momentui.
Pagrindinis scenarijus	Vartotojas įveda projekto ID į sistemos paieškos lauką, jam parodomas konkretus projektas, kokie darbai jį sudaro, kas prie jų dirba ir kiek procentų darbo jau atlikta.
Alternatyvūs scenarijai	Vartotojas neturi teisės peržiūrėti visos arba dalies informacijos, informacija nepateikiama.

Lentelė 15. Vartotojų teisių stebėjimas

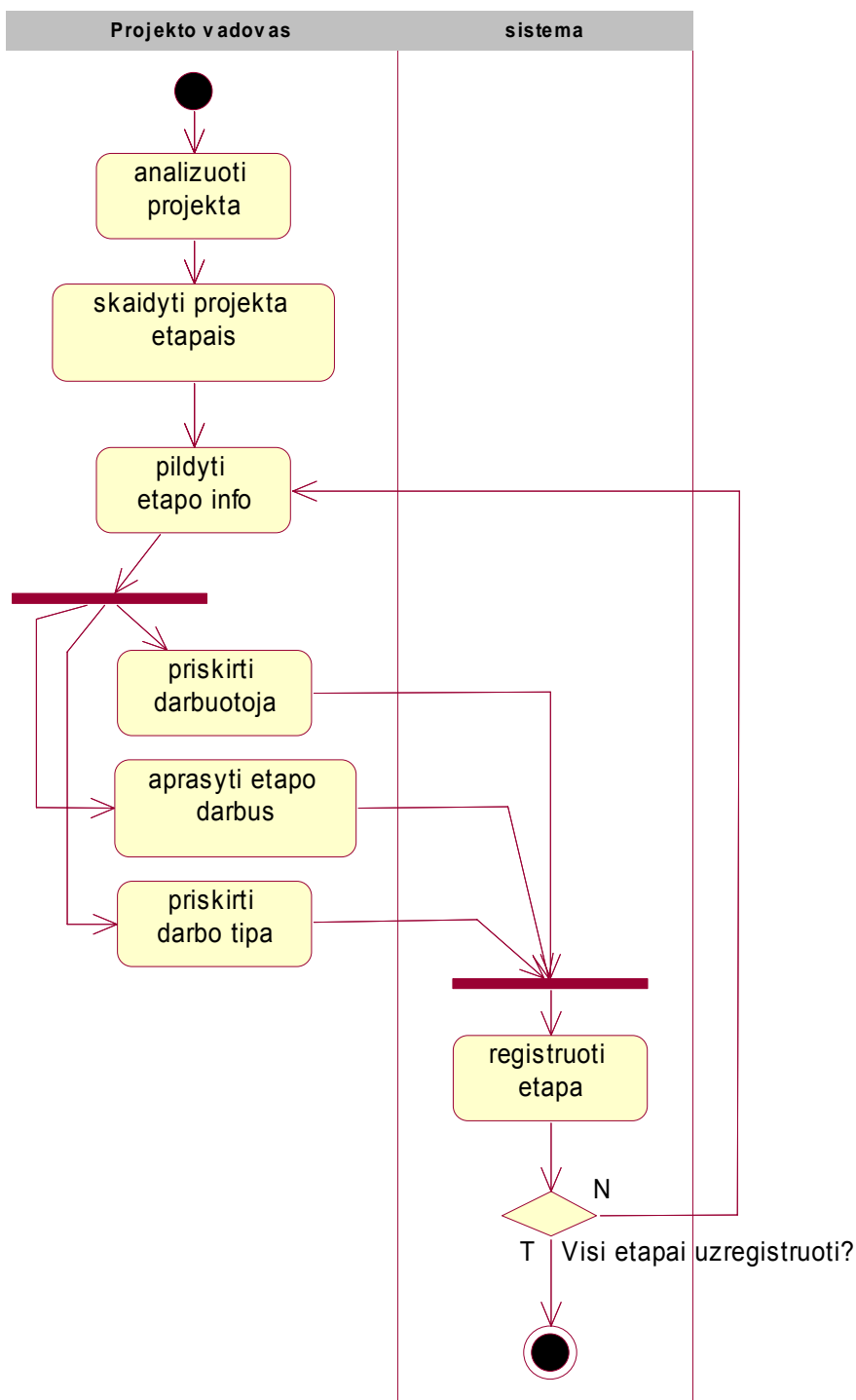
Vartotojų teisių stebėjimas	
Tikslas	Gauti sistemos vartotojų darbų ataskaitas
Aktorius (-iai)	Vartotojas, Projektų vadovas
Ryšiai su kitais panaudojimo atvejais	-
Nefunkciniai reikalavimai	Naudojimo patogumas, Patogus informacijos pateikimas
Prieš sąlygos	-
Sužadinimo sąlyga	Vartotojas kreipiasi į sistemą
Po sąlyga	Vartotojas gauna reikiamas ataskaitas, priklausomai nuo teisių
Pagrindinis scenarijus	Vartotojas priklausomai nuo jam priskirtų teisių gauna jam prieinamas ataskaitas, kurias esant reikalui galima atsispausdinti
Alternatyvūs scenarijai	-

Toliau pateiksime keletą pagrindinių veiklos procesų veiklos diagramas:



Pav. 9 Projekto registravimo veiklos diagrama.

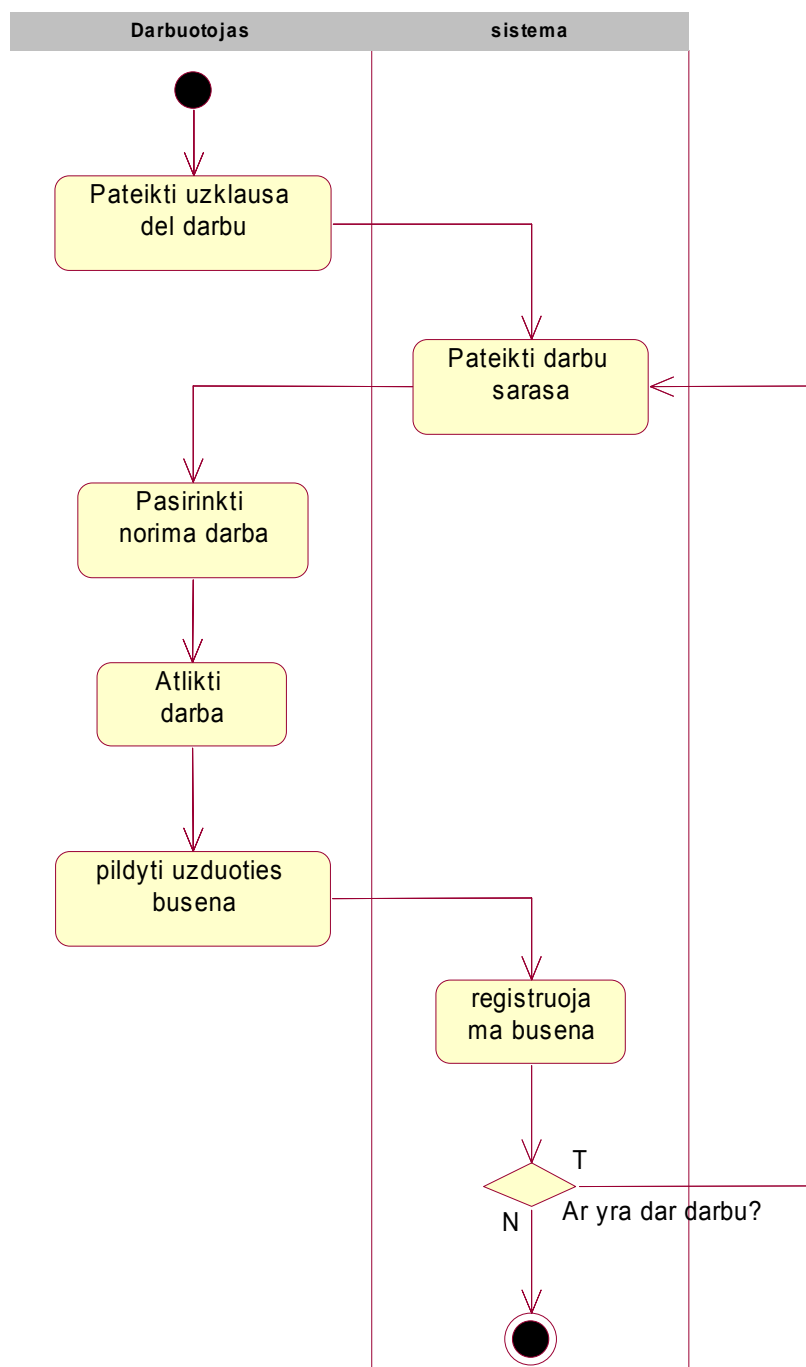
Registruojant projektą svarbu, kad klientas būtų užregistruotas. Jei kliento duomenų bazėje nėra, tada jis užregistruojamas, ir tik tada pildomi jam priklausančio projekto duomenys.



Pav. 10 darbų (užduočių) sąrašo kūrimo veiklos diagrama

Prieš formuodamas užduočių sąrašą, projekto vadovas turi gerai išanalizuoti projektą ir suskaidyti jį dalimis. Tuomet kiekviena dalis registruojama kaip atskira konkreti užduotis, priskiriant jai darbuotoją, aprašant jos darbus ir priskiriant darbo tipą. Užduotis registruojama, kai jai priskirtas darbo tipas, darbuotojas, ir ji yra aprašyta. Registruojant naują užduotį projektą vadovas vėl turi ją aprašyti,

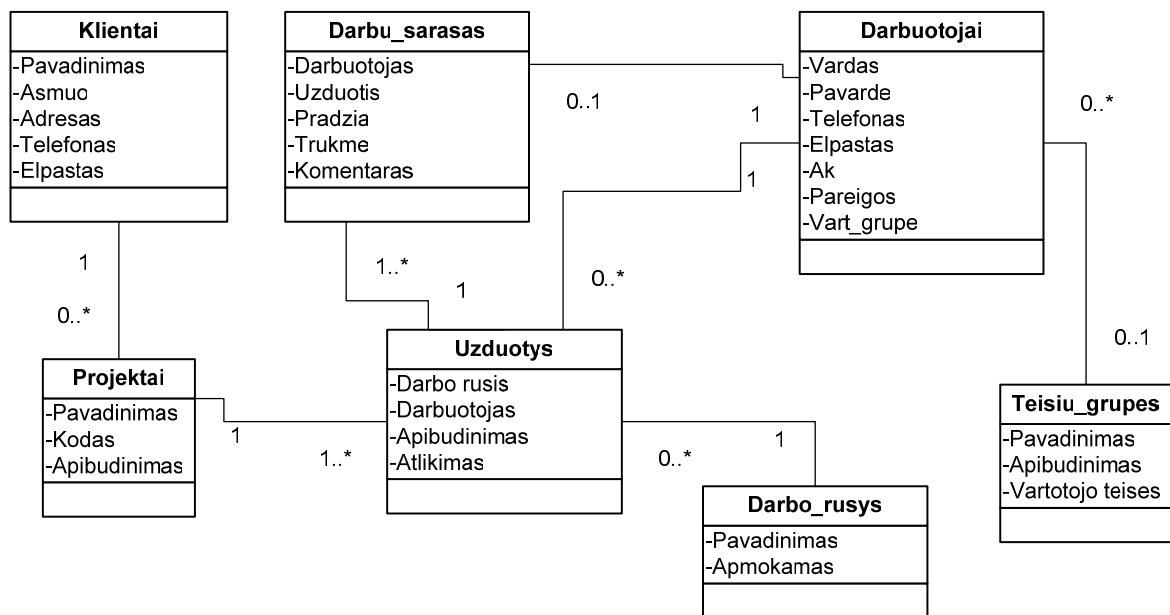
priskirti darbo tipą ir darbuotoją. Taigi, formuodamas užduočių sąrašą projektų vadovas kartu atlieka ir darbų paskirstymą.



Pav. 11 Atliktų darbų sąrašo pildymo veiklos diagrama

Darbuotojas, norėdamas registruoti atliktus darbus pirmiausia turi pateikti užklausą sistemai, kad ši jam parodytų sąrašą darbų, kurie jam paskirti. Darbuotojas pasirenka norimą užduotį, ją atlieka ir registruoja užduoties būseną (išdirbtą prie jos laiką). Sistema registruoja užduoties būseną ir tikrina ar yra neatliktų užduočių. Jeigu yra, darbuotojui vėl pateikiamas darbų sąrašas ir šis vėl renka darbą.

3.2 Dalykinės srities modelis



Pav. 12 Konceptuali UML klasių diagrama, parodanti sistemos dalykinę sritį.

Pav.12 vaizduoja dalykinės srities konceptualią UML klasių diagramą, kurioje pateikiami pagrindiniai sistemos konceptai ir ryšiai tarp jų. Iš paveikslėlio matome, kad tie konceptai yra klientai, projektai, užduotys, jas sudarančių darbų sąrašas bei susiję darbuotojai, darbų rūšys bei darbuotojams priskiriamos teisių grupės. Klientai susiję su projektais, sąryšio tipas rodo, kad klientas gali turėti daug projektų. Projektas sudarytas iš bent vienos užduoties, kuri, savaime aišku, turi būti sudaryta iš bent vieno darbo. Darbuotojai gali būti priskiriami daugeliui užduočių ir darbų, bet užduotis arba darbas gali būti priskirti tik vienam darbuotojui. Darbo rūšis neša informaciją apie tai, koks darbas kaip apmokamas, na o teisių grupės rodo, kokias teises turi konkretus darbuotojas. Darbuotojas vienu metu gali būti priskirtas ne daugiau kaip vienai teisių grupei.

4 Įmonės vidaus valdymo sistemos projektas

4.1 Sistemos modelio pagrindimas ir esmės išdėstymas

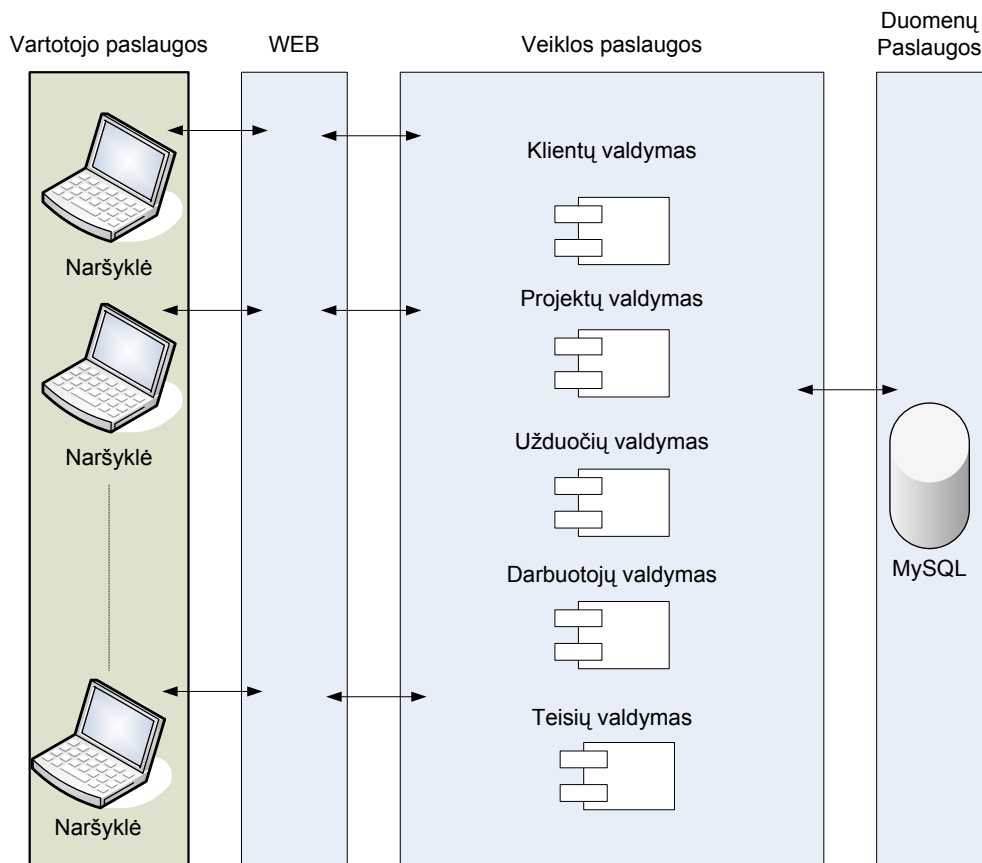
Pagrindinis kuriamos sistemos tikslas - apjungti įmonėje veikiančius verslo procesus į vientisą, centralizuotai veikiančią sistemą ir tuo pagerinti įmonės darbo organizavimą, pasiekti efektyvesnio resursų paskirstymo bei panaudojimo.

Trumpai priminsime, kaip vyksta verslo procesas įmonėje, kuriai taikysime šią sistemą:

Klientas užsako sprendimus internetinių projektų kūrimo paslaugas teikiančioje įmonėje. Tai galėtų būti internetinė svetainė, Multimedia, internetinė reklama ir t.t. Jeigu tai yra naujas klientas firmai jis yra registruojamas. Kiekvienam naujam sprendimui yra registruojamas naujas projektas, Projekto darbus išskaidomas į etapus arba užduotis, kuriuos atlieka firmoje esantys darbuotojai. Atlikti darbai registruojami sistemoje. Iš šių duomenų galima generuoti statistiką įvairiais pjūviais ir panaudoti ją apskaitoje, iš jos galima spręsti apie atliekamų darbų efektyvumą ir pan.

4.2 Sistemos architektūra - statinės struktūros modelis

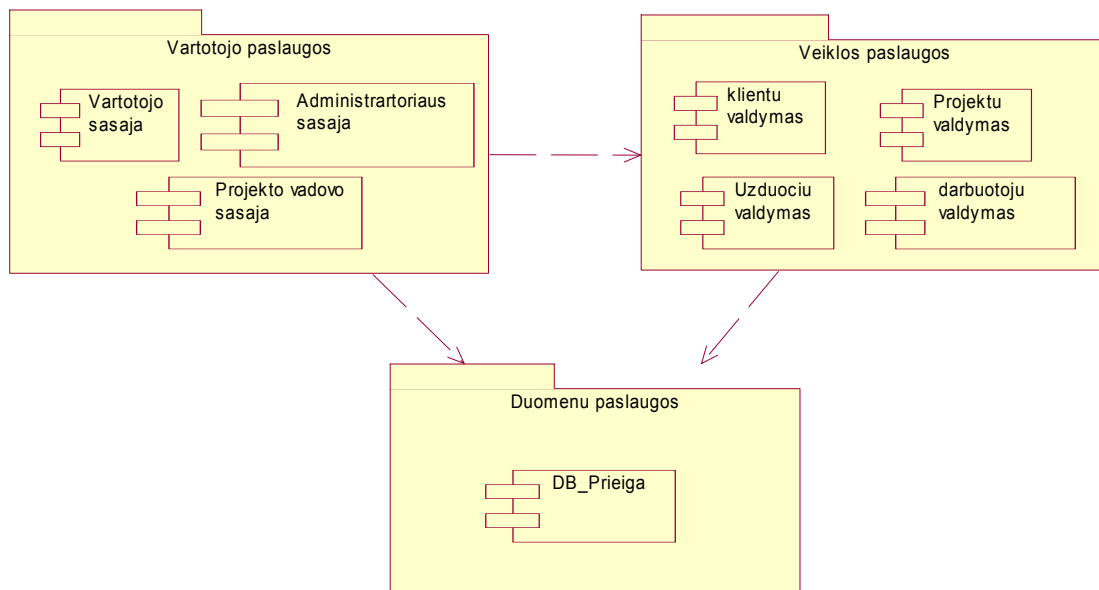
4.2.1 Loginė visos sistemos architektūra



Pav. 13 Sistemos loginė schema vaizdžiai

13 Pav. parodome, kad sistemą galime suskaidyti į keletą loginių dalių, kurių kiekviena teikia specifines paslaugas. Vartotojo paslaugų lygis apima tą sistemos dalį, kurioje vyksta bendravimas su vartotoju – tai būtų visos galimos vartotojo sąsajos. Per naršyklę šis lygis bendrauja su veiklos paslaugų lygiu, kuris fiziškai yra serverio pusėje ir ten atliekamos visos reikalingos operacijos. Veiklos paslaugų lygyje vykstant operacijoms, kreipiamasi į duomenų paslaugas teikiančią duomenų bazę.

Pagal UML standartus loginė architektūra atrodytų taip:



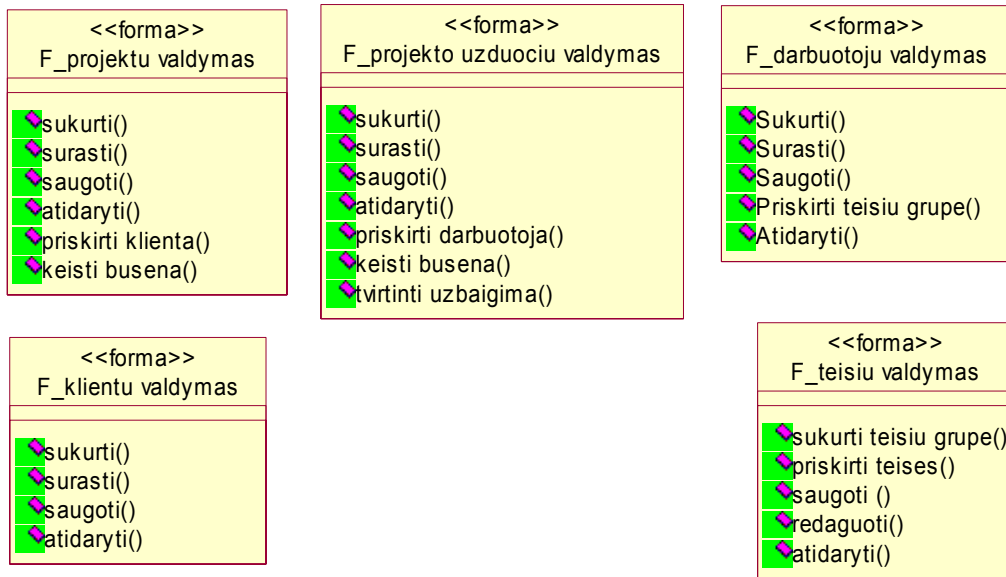
Pav. 14 Loginė sistemos architektūra

4.2.2 Vartotojo paslaugos

Vartotojo paslaugos atspindi vartotojo sąsajas. Sistema su vartotoju bendrauja naudodama šias sąsajas:

- Autorizacijos sąsaja - skirta sistemos vartotojui prisijungti prie sistemos, įvedus teisingus duomenis gauti atitinkamas teises.
- Darbuotojo informacijos modifikavimo sąsaja.
- Kliento informacijos modifikavimo sąsaja.
- Projekto informacijos modifikavimo sąsaja.
- Užduočių valdymo sąsaja.
- Teisių valdymo sąsaja.
- ...

Vartotojo paslaugos – tai formos, teikiančios prisijungusiam vartotojui galimybę atlikti darbus, susijusius su projektu, užduočių, teisių valdymu, darbuotojų ir klientų informacijos valdymu. Pagrindines vartotojo paslaugas pateiksime kaip dideles klases, kuriose yra joms būdingi metodai (funkcijos):



Pav. 15 Vartotojo paslaugų klasių diagrama

Projektų valdymo forma teikia galimybę sukurti, surasti, saugoti, atidaryti esamą projektą, priskirti projektui klientą, keisti projekto būseną.

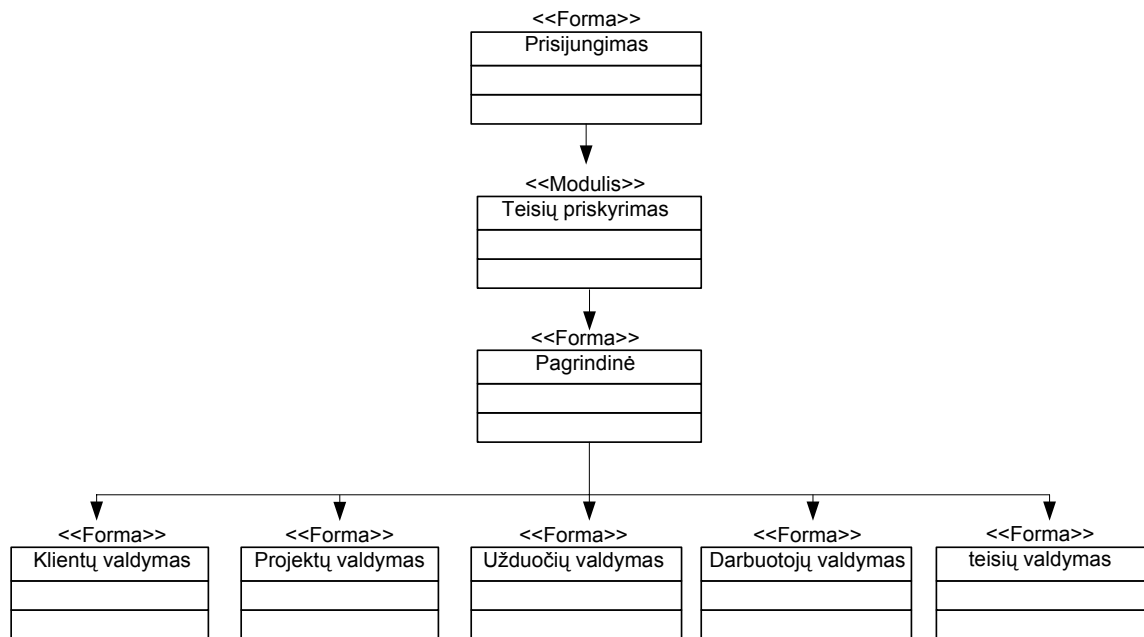
Projekto užduočių valdymo forma leidžia vartotojui atlikti visas reikiamas operacijas su užduotimis (sukurti, surasti....), taip pat priskirti užduočiai darbuotoją, keisti užduoties būseną, tvirtinti užduoties užbaigimą.

Darbuotojų valdymo forma leidžia sukurti, surasti darbuotoją, išsaugoti informacijos pakeitimus, priskirti darbuotoją teisių grupei.

Klientų valdymo forma leidžia sukurti naują klientą, surasti bei saugoti informaciją.

Teisių valdymo forma leidžia sukurti naują teisių grupę, priskirti teises, saugoti pakeitimus, atlikti redagavimą.

Toliau pateiksime vartotojo sąsajos navigacijos planą, kuriame parodyta kas vyksta, kai vartotojas jungiasi prie sistemos ir kokios sąsajos jam pateikiamos.

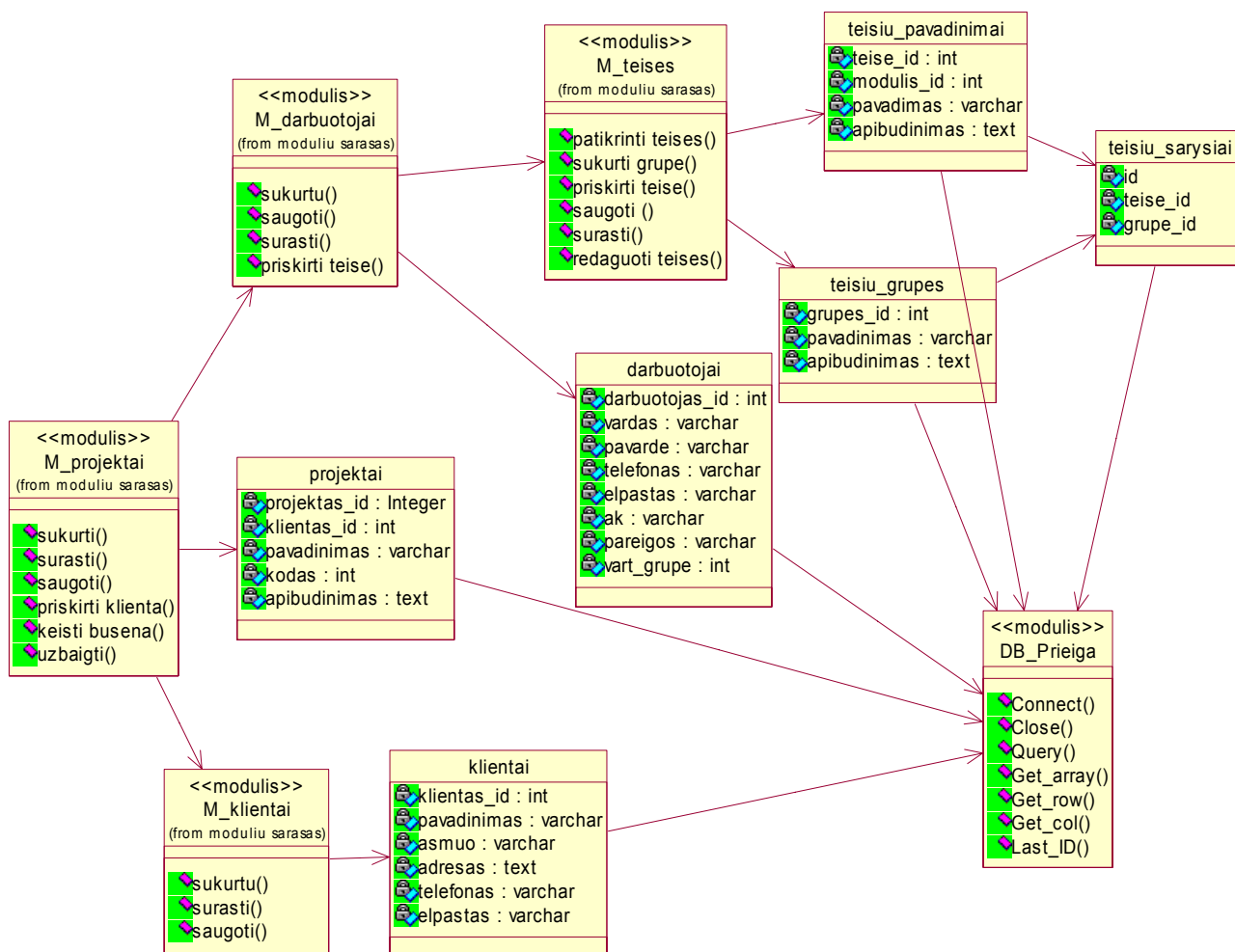


Pav. 16 Vartotojo sąsajos navigacijos planas

15 pav. rodo, kad vartotojas pirmiausia patenka į prisijungimo formą, kuri kreipiasi į teisių priskyrimo modulį, kuris pagal vartotojo įvestus duomenis priskiria jam atitinkamas teises, nuo kurių priklauso kokias sąsajas vartotojas matys ir kokias funkcijas galės atlikti.

4.2.3 Veiklos paslaugos

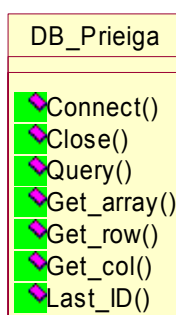
Pagrindinės veiklos paslaugos mūsų kuriamoje sistemoje yra klientų, projektų, užduočių, darbuotojų ir teisių valdymas. Šias paslaugas realizuojančios klasės yra tarpusavyje glaudžiai susijusios. Detali projekto valdymo veiklos paslaugų klasių diagrama pateikiame detalią projekto valdymo veiklos paslaugų klasių diagramą, kurioje galima matyti, kokios klasės ir kaip komunikuoja tarpusavyje.



Pav. 17 Detali projekto valdymo veiklos paslaugų klasių diagrama

4.2.4 Duomenų paslaugos

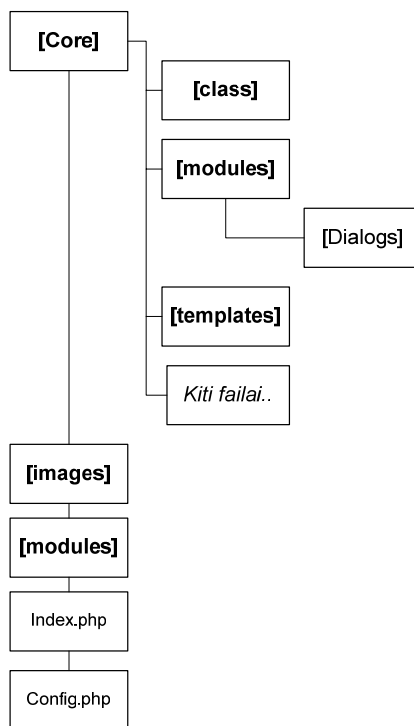
Duomenų paslaugos – tai sąveikos su duomenų baze operacijos, kurias realizuoja viena klasė, turinti visus reikiamus metodus darbui su duomenimis.



Pav. 18 klasė darbui su duomenų baze

4.4 Sistemos failų struktūra

Kiekvienoje sistemoje, turi būti ne tik tvarkingai suprojektuojamos duomenų bazės, veikimo diagramos, bet ir failų struktūra, kad kiekvienas sistemos kūrėjas galėtų lengvai surasti kiekvieną failą. Failų struktūros pavyzdys pateikiamas žemiau:



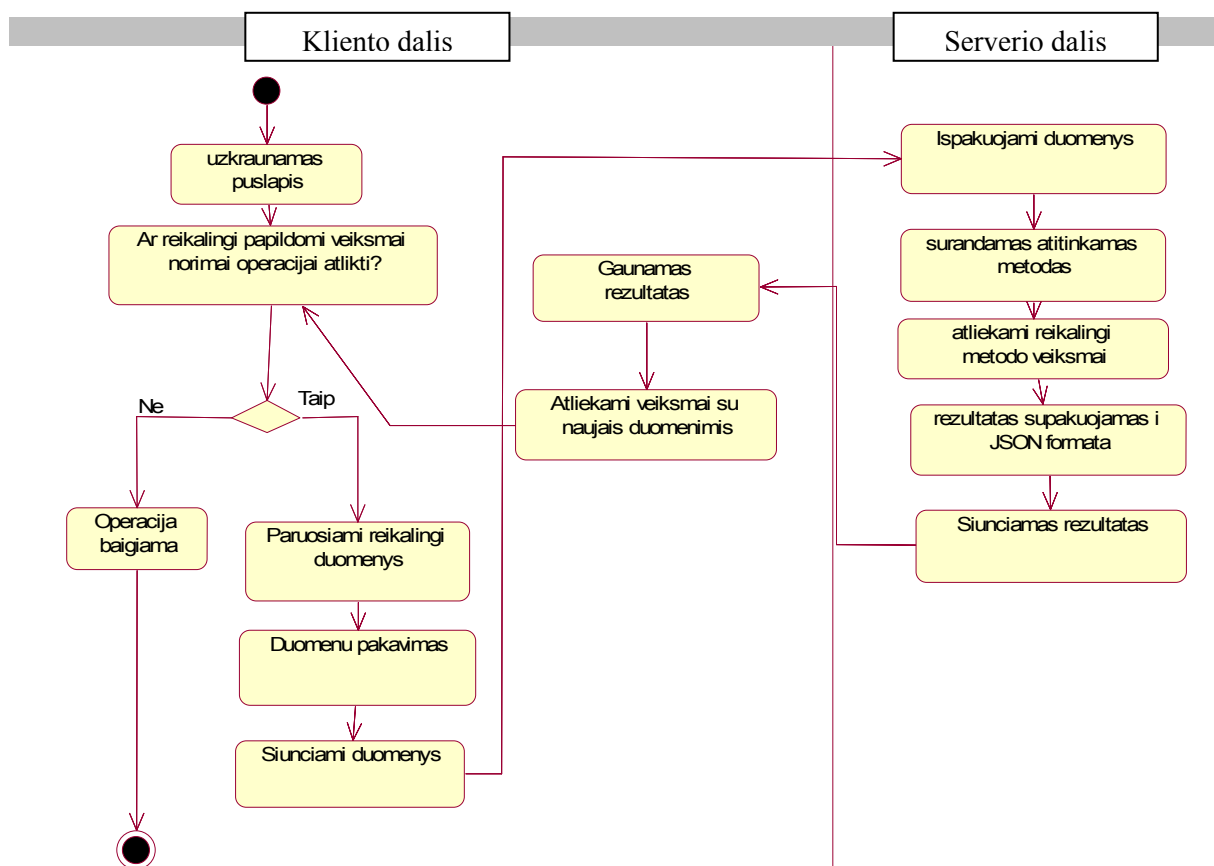
Pav. 20 Sistemos failų struktūros pavyzdys

Pateiktame paveiksle parodyta, kad norint turėti tvarkingą failų struktūrą, reikia paisyti hierarchijos ir failų vardų sudarymo taisyklių (aiškumo, informatyvumo bei atitikimo turiniui). Tokie reikalavimai failų struktūrai keliami todėl, kad tikėtina, jog prie sistemos plėtimo ir tobulinimo prisidės kiti programuotojai, ir aiškumas bei informatyvumas reikalingas tam, kad jie žinotų, kur kokie failai padėti ir kas juose yra .

4.5 Realizuotas sistemos karkasas

Pagrindinis mūsų sukurtos sistemos naujumas – AJAX metodikos pagrindu veikiantis varikliukas, kurio pagalba mūsų sistema įgyja daugiau dinamiškumo ir duomenų siuntimo/keitimo/gavimo procesai vyksta greičiau. Detalesnis AJAX metodikos aprašymas pateiktas 2.7.1 paragrafe.

Žemiau pateikiame veiklos diagramą, iliustruojančią mūsų sukurto varikliuko darbą:



pav. 21 Sistemos veikimas kliento ir serverio pusėse

Varikliukas veikia taip: užkrovus puslapį, tikrinama, ar norimam veiksmui atlikti reikalingi papildomi veiksmai iš serverio pusės. Jeigu ne, varikliukas atlieką tą operaciją pats ir darbas baigiamas. Jeigu reikalingi papildomi veiksmai, varikliukas išrenka reikalingą apdoroti duomenų porciją, juos supakuoja į reikiamą formatą ir išsiunčia serveriui. Serveris, gavęs duomenis, juos išpakuoja, suranda reikiamą metodą jiems apdoroti, apdoroja ir po to rezultatus supakuoja į JSON formatą ir siunčia atgal klientui. Klientas, gavęs naujus duomenis atlieka reikiamus veiksmus, ir toliau vėl tikrinama, ar dar reikia kokių nors veiksmų. Jei ne, operacijos baigiamos vykdyti ir darbas baigiasi.

Sukurtasis varikliukas naudojamas visame sistemos karkase.

Sistemos karkasas yra sukurtas taip, kad esant reikalui modifikuoti tam tikrus modulius (pavyzdžiui, pridėti naujus laukelius), tai būtų galima padaryti kuo greičiau ir paprasčiau.

Pasinaudojus AJAX metodika, galima pasiekti tokių rezultatų, kad sumažinti vartotojo veiksmų, reikalingų atlikti operacijai, kiekį iki minimumo.

Mūsų sistemoje vienas iš tokių pavyzdžių – darbo atlikimo procento fiksavimas užduočių sąraše. Jeigu sistema būtų realizuota tradicinėmis WEB programavimo priemonėmis, tokiam veiksmui atlikti reiktų 3 žingsnių:

1. Standartinis scenarijus:
 - a. Pasirinkus įrašą, kurį norima redaguoti, reiktų paspausti nuorodą, atidarančią atlikimo procento keitimo langą.
 - b. Užsikrovus naujam langui pakeisti darbo atlikimo procentą
 - c. Paspaudus, saugojimo mygtuką, grįžtama į pagrindinį langą

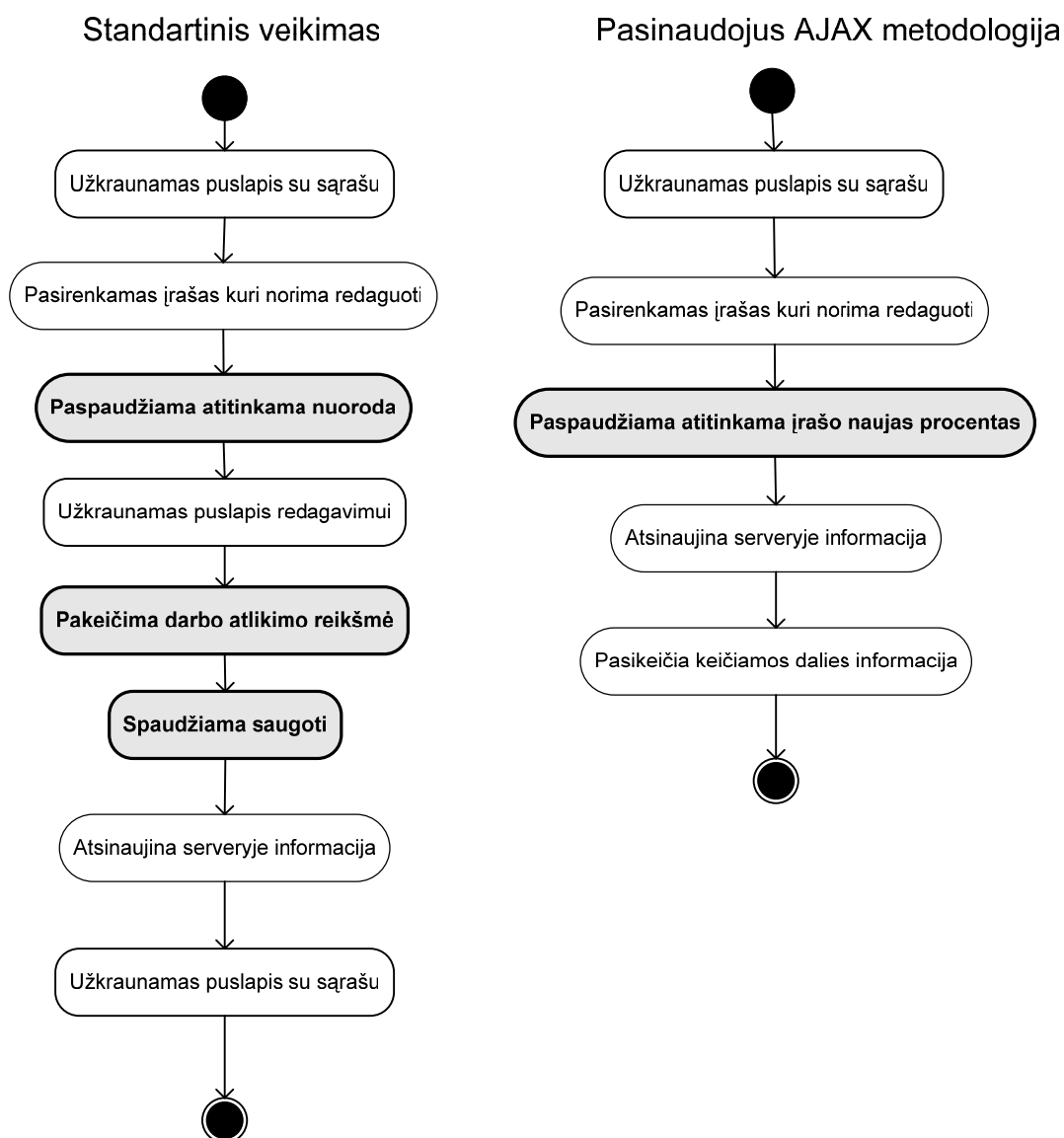
2. Pasinaudojus AJAX metodologija
 - a. Pasirinkus norimos užduoties, paspausti ant atitinkamo kvadrato. Automatiškai duomenys išsisaugo serveryje, bei vizualiai pasikeičia atvaizdavimas tos pakeistos dalies. Vizualiai tai atrodo taip:

Projektas	Užduotis	Atlikta	Terminas
1 Topspot: i-lažybos	Duomenų sinchronizacija	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	2005-12-08
3 Topspot: i-lažybos	Pasiulos modulis	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	2005-12-09
1 Jonas: Jonas.lt svetaine		<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	2005-12-15
4 Jonas: Jonas.lt svetaine		<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	2005-12-23
3 Vidiniai darbai		<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	2007-12-30

Atlikimas suskaidytas dalimis po 20%. Kad fiksuoti atlikimą, tereikia spustelti ant tuščio kvadratėlio, ir pakeitimas fiksuojamas be viso puslapio perkrovimo.

Pav. 22 Užduoties atlikimo procento fiksavimo langas

Schematiškai šie du scenarijai atrodytų taip, kaip pateikta žemiau esančioje veiklos diagramoje. Ten pilkiau pažymėti blokeliai reiškia konkretų vartotojo veiksmą, kuris formuoja vienokią ar kitokią užklausą sistemai.



Pav. 23 Vartotojo veiksmų kiekis standartinėje realizacijoje ir panaudojus AJAX metodiką

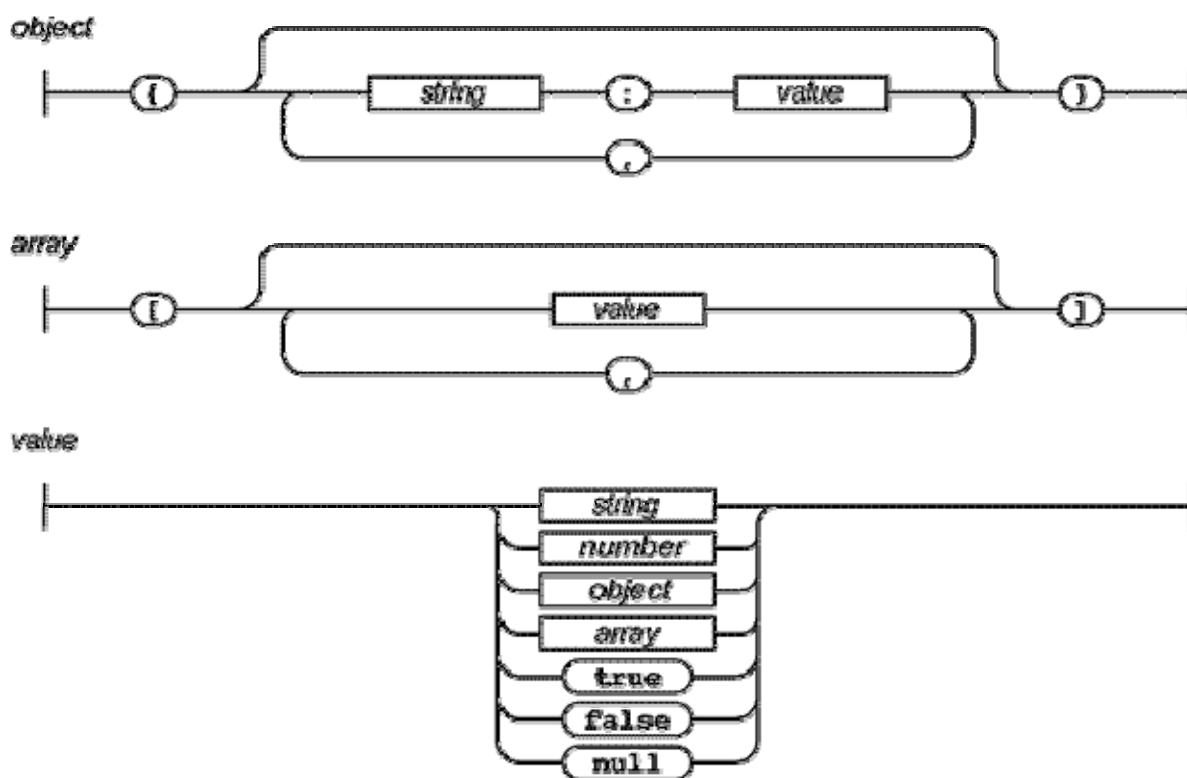
Mūsų sukurto varikliuko veikimo principas labai paprastas: kliento daliai prireikus tam tikrų duomenų arba atsiradus poreikiui serverio dalyje atlikti tam tikrus duomenų pakeitimus, kliento dalis pasiunčia reikiamą metodo pavadinimą bei turimus duomenis (kintamuosius). Duomenys gali būti tiek skaičiai, tiek tekstas arba ištisas masyvas. Pagal nutylėjimą sistema kreipiasi į tą patį puslapį, kuriame dabar vartotojas yra. Tame pačiame faile aprašytas ne tik paprastas HTML atvaizdavimas vartotojui, bet ir apdorojami šio AJAX varikliuko metodo duomenys.

AJAX – tai asinchroninis JavaScript ir XML. Tačiau realizuojant konkrečią sistemą, pagal esamus poreikius galima naudoti tiek sinchroninį, tiek asinchroninį duomenų gavimo būdą, o gauti duomenys iš karto gali būti suformuoti taip, kaip tam atvejui labiausiai reikia: paprasto teksto, XML ar jau suformuoto JavaScript masyvo pavidalu.

Šiame procese dalyvauja tiek kliento, tiek serverio pusės. Kliento dalyje paruošiami pradiniai duomenys ir išsiunčiami su atitinkamo metodo pavadinimu. Šie duomenys yra supakuojami, nes, kaip minėjome anksčiau, duomenys gali būti ne tik tekstas, bet ir visas masyvas. Serverio pusėje suradus atitinkamą metodą, atliekami reikalingi veiksmai su pradiniais duomenimis. Gautas rezultatas išsiunčiamas atgal į kliento pusę. Rezultato išvedimas, priklausomai nuo situacijos, gali būti paruošiami keliais būdais:

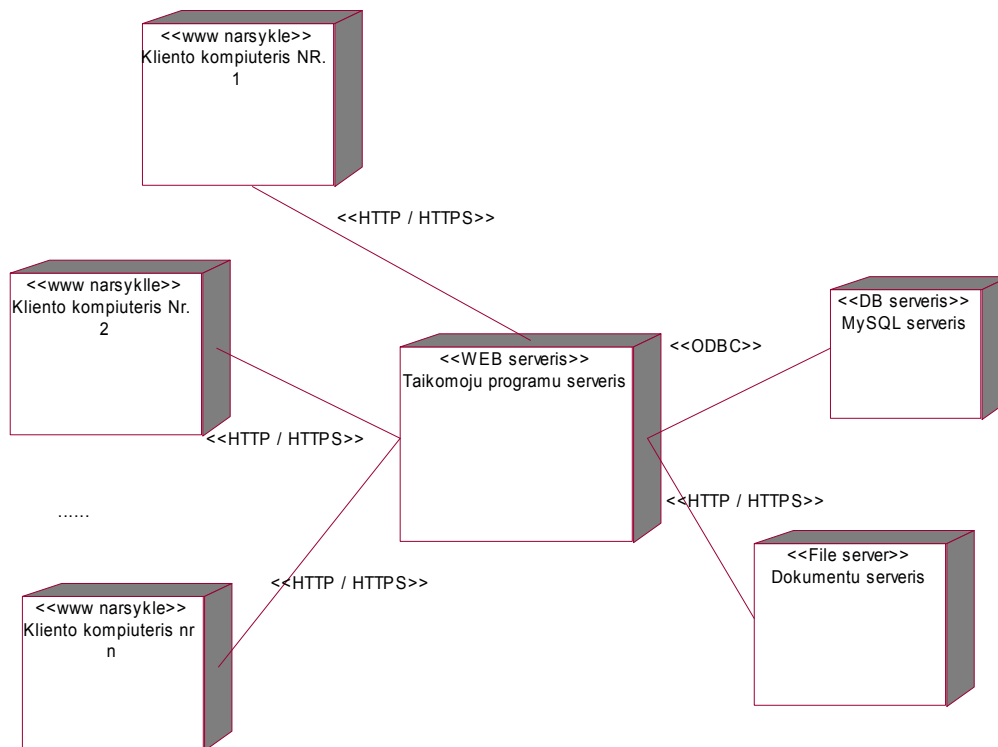
- Paprastas tekstas
- XML
- JSON

Geriausia pasirinkti tokį metodą, kurį sistema apdorotų greičiausiai. Padavus JavaScript'ui duomenis tiesiogiai XML formatu, JavaScript'as, prieš juos apdorodamas, turi atlikti jų analizę (*eng.* parse) ir paversti į sau priimtina formatą. Duomenis paruošus JSON metodu, JavaScript'ui nereikia papildomos analizės, nes šį formatą Javascript'as supranta tiesiogiai, tad iš karto galima jais naudotis. JSON metodu galima aprašyti tiek paprastą tekstą, tiek skaičių ar ištisą masyvą arba netgi objektą .



pav. 24 JSON metodu aprašyti duomenys

4.6 Įdiegimo modelis



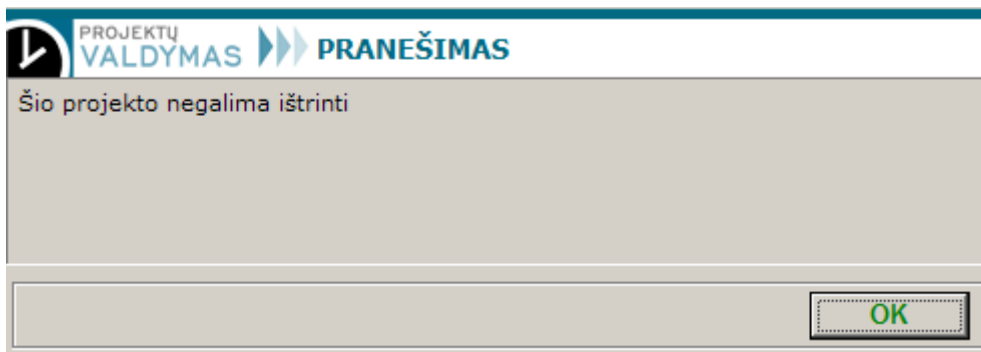
Pav. 25 Sistemos įdiegimo modelis

Šis modelis vaizduoja fizinę sistemos komponentų išsidėstymą. Web serveris, Failų serveris ir DB serveris gali būti realizuoti tiek viename serveryje, tiek kiekvienas atskirai. Šiuo atveju parodyta, kad šie serveriai yra atskirai, ir tai būtų geresnis variantas, nes sistema tokiu atveju veikia greičiau.

5 Eksperimentinis sistemos tyrimas

5.1 Eksperimentinės sistemos kūrime panaudotų pagalbinių priemonių aprašymas

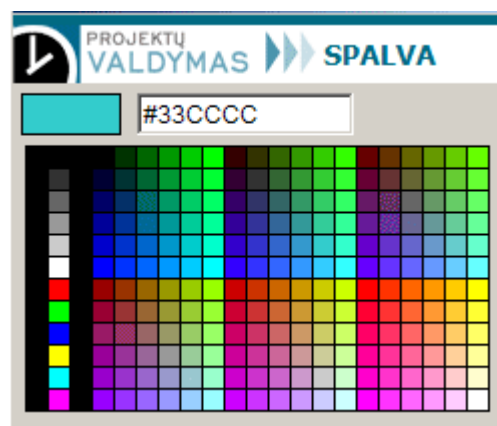
Siekiant palengvinti realizavimo procesą, buvo sukurta keletas pagalbinių priemonių, skirtų pagreitinti kai kurių programinių elementų kūrimą. Tai būtų specialūs dialogai, tokie kaip patvirtinimas, reikšmės įvedimas, kalendorius, datos įvedimas, duomenų dialogai ir priemonė klaidoms surasti bei informacijai apie objektus pateikti (debuggeris). Jų vaizdus pateikiame žemiau:



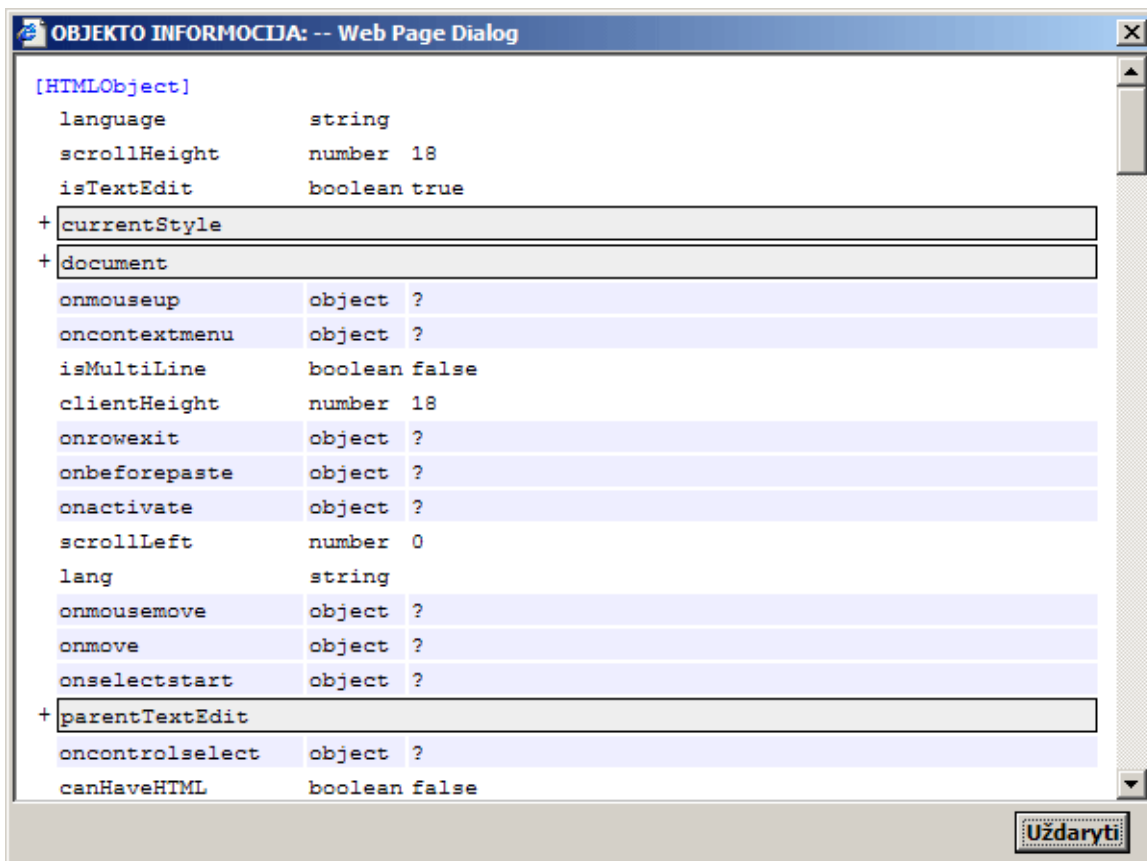
Pav. 26 Pranešimo atvaizdavimas sistemoje



Pav. 27 Kalendorius



Pav. 28 Spalvos pasirinkimas



Pav. 29 Sistemos debugeris, atvaizduojantis objektų bei duomenų informaciją kliento dalyje

5.2 Eksperimentinės sistemos veikimas

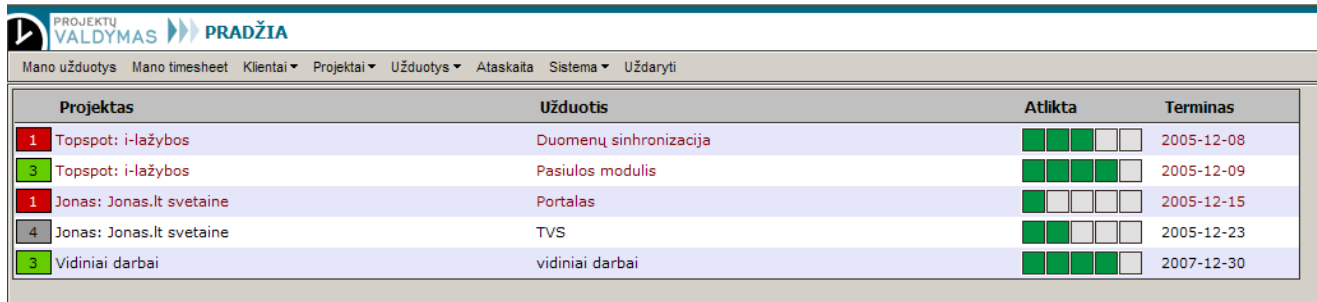
Šiame skyriuje pateiksime glaustą sistemos veikimo vartotojo požiūriu aprašymą.

Įeinant į įmonės vidinę sistemą vartotoją pasitinka prisijungimo langas, kuriame reikia įvesti prisijungimo duomenis. Mūsų atveju tai el. pašto adresas. Tokio tipo vartotojo identifikatorius pasirinktas todėl, kad vėliau plečiant sistemą numatoma įdiegti el. pašto modulį.

Pav. 30 Prisijungimo langas

Pirmas langas, pasitinkantis vartotoją yra jam priskirtų užduočių sąrašas. Užduotys rikiuojamos pagal užduoties terminą ir svarbumą, o jei abi reikšmės vienodos - tai pagal pradžios datą. Žinoma, darbuotojas gali neatsižvelgti į šio sąrašo eiliškumą, jei firmos viduje nusprendžiama kažkurį darbą

padaryti greičiau. Taip pat pateikiama informacija apie kiekvieno darbo atlikimą (procentais). 32 pav. parodys, kad užduoties atlikimas suskirstytas į 5 dalis po 20% ir pateikiama kvadratėlių pavidalu. Užpildytas žaliai kvadratis reiškia kad ši užduoties atkarpa jau įvykdyta. Vartotojas, paspaudęs ant kvadratis, gali labai greitai pakeisti užduoties atlikimo procentą.



Projektas	Užduotis	Atlikta	Terminas
1 Topspot: i-lažybos	Duomenų sinchronizacija	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	2005-12-08
3 Topspot: i-lažybos	Pasiulos modulis	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	2005-12-09
1 Jonas: Jonas.lt svetaine	Portalas	<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	2005-12-15
4 Jonas: Jonas.lt svetaine	TVS	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	2005-12-23
3 Vidiniai darbai	vidiniai darbai	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	2007-12-30

Pav. 31 Pradinis langas, vartotojo duomenys

Prisijungus prie sistemos matomas pagrindinis vartotojo meniu, su gilesniais lygiais, kurie išsiskleidžiami pasirinkus konkretų meniu punktą. Priklausomai nuo vartotojo teisių grupės, kiekvieno vartotojo meniu gali būti matomas skirtingai.

Menu struktūra:

- Mano užduotys - *užduotis kurias reikia atlikti*
- Mano timesheet – *darbų registravimo sistema*
- Klientai – *įrankiai skirti klientų valdymui*
 - Klientų sąrašas
 - Sukurti naują klientą
- Projektai - *įrankiai skirti projektų valdymui*
 - Projekto sąrašas
 - Naujas projektas
- Užduotys - *įrankiai skirti užduočių valdymui*
 - Užduočių sąrašas
 - Naujas užduotis
- Ataskaitos – *įvairių ataskaitų spausdinimas*
 - Darbuotojo ataskaitos
 - Projekto ataskaitos
 - Statistika
 - ...
- Sistema – *Sisteminiai nustatymai (vartotojai, teisės, įvairūs tipai)*
- Vartotojai
- Vartotojų grupės

- Projektų tipai
- Darbų tipai
- ...

Dirbant kiekvieną darbą būtina vesti atliktų darbų suvestinę (timesheet). Siekiant neklaidinti vartotojo, jam pateikiami tik tie projektai bei užduotys, kurios yra jam priskirtos. Rekomenduotina neapleisti šio darbų registravimo, nes priešingu atveju galimos įvairios renkamos statistikos paklaidos.

Pasirinkus atitinkamus projektus bei užduotis, atliktų darbų suvestinės lange matomas tvarkaraštis su laukais, kuriuose galima kas valandą, arba kas 15 minučių įvesti atliekamo darbo pavadinimą bei komentarus.

Remiantis darbo patirtimi įmonėje, nuspręsta, kad į smulkesnius laiko intervalus užduočių atlikimo skaidyti neapsimoka.

The screenshot displays a TimeSheet application interface. The main window is divided into a task list on the left and a sidebar on the right. The task list shows entries for 16:00, 17:00, 18:00, and 19:00. The 18:00 entry is highlighted in yellow. A callout box points to the 17:00 entry with the text "Atliktų darbų suvestinės langas". The sidebar shows a calendar view for December 21, 2005, with a callout box pointing to the "Projektų langas" and another pointing to the "Užduočių langas". The bottom of the sidebar has a "SAUGOTI VISUS" button.

Pav. 32 TimeSheet vedimas

Turint duomenis apie kiekvieno vartotojo atliktus darbus, galima gauti skirtingas ataskaitas, kurios gali būti naudojamos skirtingoms reikmėms. Tarkim viena tokių reikmių - vartotojo algalapio dalis, rodanti jo pradirbtas valandas prie kiekvieno projekto užduočių.

Projektą į užduotis skaido projektų vadovas. Jo darbas yra registruoti projektą, suskaidyti jį į sudedamąsias užduotis ir paskirstyti jas tarp įmonėje esančių darbuotojų. Žemiau pateiksime užduoties registravimo/informacijos redagavimo lango vaizdą:

PROJEKTŲ VALDYMAS UŽDUOTIES REDAGAVIMAS

Užduoties id: 2

Pavadinimas: Pasiulos modulis

Aprašymas: Iš XML pateiktų duomenų suformuoti pasiūla, kiekvienai sporto šakai po atskirą

Darbo tipas: Programavimas

Projektas: Topspot: i-lažybos

Priskirta darbuotojui: Vaidas Žilionis (Programuotojas)

Svarbumas: Normalus

Numatoma darbo pradžia: 2005-11-01

Galutinis terminas: 2005-12-09

Atlikta: 80%

Saugoti Uždaryti

Pav. 33 Užduoties informacijos redagavimo langas

Toliau pateikiame bendrą užduočių lango vaizdą:

PROJEKTŲ VALDYMAS UŽDUOČIŲ SĄRAŠAS

Mano užduotys Mano timesheet Klientai Projektai Užduotys Ataskaita Sistema Uždaryti

Sukurti Redaguoti Ištrinti Filtruoti Valyti

#	Projekto pavadinimas	Darbo pavadinimas	Darbo tipas	Darbuotojas	Pradžia	Pabaiga	Svarbumas
1	Topspot: i-lažybos	Suprojektuoti	Projektavimas	Julius Janonis	2005-11-10	2005-11-25	Normalus
2	Topspot: i-lažybos	Pasiulos modulis	Programavimas	Vaidas Žilionis	2005-11-01	2005-12-09	Normalus
3	Topspot: i-lažybos	Duomenų sinchronizacija	Programavimas	Vaidas Žilionis	2005-12-02	2005-12-08	L. svarbus
4	Topspot: i-lažybos	Svetainės dizainas	Grafininis dizainas	Petras Jankauskas	2005-11-28	2005-12-06	Normalus
5	Topspot: i-lažybos	TVS integravimas	Programavimas	Vaidas Žilionis	2005-11-25	2005-12-22	Svarbus
6	Topspot: i-lažybos	Naujienu modulis	Programavimas	Vaidas Žilionis	2005-12-20	2005-12-23	Normalus
7	Topspot: i-lažybos	Reklaminiis Baneris	Programavimas	Petras Jankauskas	2005-12-20	2005-12-22	L. svarbus
8	Topspot: i-lažybos	Su klientu aptariamais klausimais	Susitikimas su klientais	Birute Kudrikaitė	2005-12-01	2005-12-30	Normalus
9	Omnitel: wap OmniSurf svetainė	Sukurti muzikini klipa	Grafininis dizainas	Petras Jankauskas	2005-12-22	2007-12-28	L. svarbus
10	Omnitel: wap OmniSurf svetainė	XML 2 WAP	Programavimas	Vaidas Žilionis	2005-12-22	2005-12-27	Svarbus
11	Jonas: Jonas.lt svetaine	TVS	Programavimas	Vaidas Žilionis	2005-12-21	2005-12-23	Nesvarbus

< > Puslapiai: 1 - 13 Viso: 13 Eiluciu puslapyje: 25 Rodyti: Visus

Pav. 34 Bendras užduočių sąrašo langas

Įvedant informaciją apie užduotį, svarbu fiksuoti kokiam projektui užduotis priklauso, koks darbo pavadinimas, koks darbo tipas, koks darbuotojas už tą užduotį atsakingas, užduoties pradžia, pabaiga bei svarbumo rodiklis. Yra galimybės filtruoti užduočių sąrašą pagal projektą bei pagal darbo tipą.

Kaip jau minėta, darbuotojai savo atliktų darbų suvestinę privalo pildyti reguliariai, o surinkta informacija naudojama sumuojant darbuotojo išdirbtą darbo laiką bei renkant įvairią statistiką apie užduočių atlikimo terminus, apie darbuotojo darbo laiko paskirstymą ir pan.

Viena iš galimų ataskaitų – darbuotojo darbų ataskaita:

PROJEKTŲ VALDYMAS » UŽDUOČIŲ SĄRAŠAS							
Mano užduotys Mano timesheet Klientai Projektai Užduotys Ataskaita Sistema Uždaryti							
		14:00-16:00	2:00	Topspot: i-lažybos	Pasiulos modulis	analize	53:15 / 53:15
12-21	10:00	08:00-16:00	8:00	Omnitel: wap OmniSurf svetainė	XML 2 WAP		33:45 / 33:45
		16:00-17:00	1:00	Omnitel: wap OmniSurf svetainė	XML 2 WAP	darbas	33:45 / 33:45
		17:00-18:00	1:00	Vidiniai darbai	vidiniai darbai		18:00 / 18:00
12-22	09:00	08:00-13:00	5:00	Omnitel: wap OmniSurf svetainė	XML 2 WAP		33:45 / 33:45
		13:00-14:00	1:00				
		14:00-18:00	4:00	Omnitel: wap OmniSurf svetainė	XML 2 WAP		33:45 / 33:45
12-23							
12-24							
12-25							
12-26							
12-27							
12-28							
12-29							
12-30							
12-31							

Projektas	%	Dirbta
Jonas: Jonas.lt svetaine		41:15
Portalas		29:15
TVS		12:00
Topspot: i-lažybos		53:15
TVS integravimas		07:00
Naujienu modulis		18:15
Duomenų sinchronizacija		05:00
Pasiulos modulis		23:00

Pav. 35 Darbuotojų darbų ataskaitos langas.

5.3 Eksperimentinio diegimo aprašymas

Kaip minėta anksčiau, šios sistemos reikavimai yra:

- PHP (nuo 4.20 versijos)
- MySQL Serveris nuo 3.23 versijos
- Apache Serveris (nuo 3.23 versijos)

Sistemos paleidimui būtini keli žinginiai:

- Failų nukopijavimas – reikia nukopijuoti visus instaliacinius failus į Apache serverio sukonfigūruotą virtualhostą. Rekomenduotina naudoti atskirą domeną
- Duomenų bazės sukūrimas, - įterpti į duomenų bazę paruoštą SQL failą
- Konfigūracinio failo redagavimas - įvesti tinkamus MySQL serverio adresu, duomenų bazės, vartotojo vardo bei slaptažodį.

Standartiniame įdiegime yra jau sukurti šabloniniai bendrieji nustatymai, kurios reikalui esant sistemos vartotojas galės pasikeisti. Tokius kaip projekto tipų pavadinimai, darbų tipai ir pan.

5.4 Sukurtos sistemos charakteristikų bei atitikimo reikalavimams tyrimas

Vienas iš svarbiausių dalykų sukūrus naują sistemą – įvertinti kiek ji atitinka užduotus funkcinius ir nefunkcinius reikalavimus bei kokybės kriterijus.

Pirmas kokybės kriterijus skambėjo taip: **sistemos naudojamumas** (usability) - sistema turi gerai tikti visų tipų vartotojams, neapkrauti jų papildomu darbu. Vartotojo sąsaja turi būti intuityvi, optimali. Sistemos vartotojo sąsajai pasirinkta taikyti stilių, artimą Windows operacinės sistemos sąsajos stiliui, kad vartotojui pateikiama sistemos aplinka kuo labiau primintų jam įprastą aplinką ir būtų kuo lengviau įsisąmoninama.

Sistemos realizavime panaudota AJAX programavimo metodika sumažino vartotojui reikalingų atlikti veiksmų kiekį iki minimumo, kas taip pat pagerino sistemos naudojamumą.

Vartotojo sąsajos langai suprojektuoti taip, kad būtų aišku kur kokia informacija reikalinga ir ką reikia daryti. Pavyzdžiui, vartotojo informacijos redagavimo langas atrodo taip:

Pav. 36 Vartotojo informacijos redagavimo langas

Antras – plečiamumo kriterijus įgyvendintas suprojektavus modulinę sistemos architektūrą, ko pasekoje tolesnis sistemos plėtimas bus ganėtinai lengvas.

Toliau – sistemos **funkciniai reikalavimai** :

- klientų valdymas – realizuotas informacijos apie klientus rinkimo forma;
- projektų valdymas, su galimybėmis rengti ataskaitas, įvedinėti informaciją apie projektus bei sudarinėti užduočių bei darbų sąrašus realizuotas projektų informacijos rinkimo bei užduočių sąrašų formavimo moduliuose.

- Darbuotojų bei jų atliekamų darbų informacijos rinkimas realizuotas darbuotojo informacijos rinkimo, atliktų darbų suvestinės pildymo moduluose.
- Vartotojų identifikavimas realizuotas vartotojo autorizacijos forma.

Nefunkciniai sistemos **reikalavimai** buvo tokie :

- Saugumas – sukurta vartotojų teisių sistema, prisijungiančių vartotojų autorizavimas, o siekiant bendro sistemos darbo saugumo užtikrinimo, rekomenduojama sistemą paleisti internete naudojantis saugiu https protokolu.
- Informacijos prieinamumas – visi vartotojai, kam suteiktos atitinkamos teisės, gali dominančią informaciją gauti ataskaitų bei sąrašų forma.
- Atsparumas klaidoms – didžiąja dalimi tai vartotojo klaidų vengimas, realizavus tokius apribojimus kaip išdirbto laiko ribojimas (pvz negalima įvesti 25 darbo valandų per parą), užduočių filtravimas kiekvienam darbuotojui (rodomi tik jam priskirti darbai), draudimas keisti projekto atlikimo duomenis, kai sueina projekto pabaigos terminas ir projektas yra realiai pabaigtas (siekiant išvengti sukčiavimo).
- Diegimo ir priežiūros paprastumas - šis reikalavimas įgyvendintas pasirinkus kliento – serverio architektūrą, kai vartotojai prie pagrindinės sistemos jungiasi paprastos interneto naršyklės pagalba. Tokiu būdu sistema prieinama visiems, nereikia jos diegti atskirose darbo vietose, o prižiūrėti reikia tiktai serverius.
- Vartotojo sąsajos patogumas jau aptartas anksčiau.
- Kainos prieinamumas – sistemą pasirinkta realizuoti atviro kodo priemonėmis, kas labai sutaupė piniginių lėšų. Realizavimas kainavo pačių atlikėjų darbo ir laiko.

Taigi, apibendrintai galime pasakyti, kad visi užsibrėžti reikalavimai ir tikslai buvo pasiekti.

5.5 Kas jau padaryta, ir ką dar būtų galima padaryti ateityje

Sukurta sistema leidžianti mainytis duomenimis tarp kliento ir serverio pusės neperkraunant naršyklės. Tai reiškia, kad kai kuriais atvejais galima sumažinti vartotojo veiksmų skaičių iki minimalaus kiekio (pavyzdys nagrinėtas 4.5 paragrafe)

Kas dar galėtų būti padaryta naudojantis sistemos varikliu?

- **Paieška realiu laiku** – Norint pasirinkti atitinkamus duomenis, nebūtina atlikti visos pilnos paieškos. Galima vedant paieškos lauke tekstą gyvai siūlyti galimus paieškos rezultatus, kuri vartotojas gali pasirinkti.

- **Duomenų teisingumo tikrinimas puslapiui nepersikrovus** – ne visuomet išeina tikrinti duomenų teisingumą tik kliento pusėje, kartais būtinas duomenų tikrinimas ir serverio pusėje. Pagal standartinį scenarijų, persikrovus puslapiui turėtų būti parodomas pranešimas apie tai, kad tam tikri duomenys įvesti nekorektiškai, arba jei yra tam tikra sąlyga – jie jos netenkina. Pasinaudojus nauja metodologija, šis pranešimas būtų parodomas dar prieš išsaugant duomenis.
- **Rašybos tikrinimas** – tam tikruose moduluose, yra būtinas rašybos tikrinimas (el. pašto modulis, ataskaitų formavimas). Galima pasiekti tokio rezultato, kad sistema tikrintų rašybą gyvai, pažymėtų netinkamus žodžius, pasiūlytų žodžio keitimo variantus.

6 Išvados

1. Sprendžiant problemą išanalizuota informacija apie kitas įmonių vidaus valdymo sistemas, ir pastebėta, kad daugumoje jos yra orientuotos į įmonės finansų bei apskaitos tvarkymą, ir yra skirtos daugiau prekybinėms arba gamybinėms įmonėms. Konkrečios įmonės, kuriai buvo kuriama mūsų sistema, poreikiai kiek kitokie, todėl, atsižvelgus į turimą patirtį web programavime, o taip pat į galimas išlaidas, buvo priimtas sprendimas daryti sistemą patiems.
2. Sudarant reikalavimus sistemai buvo susipažinta ir pasinaudota sistemos savybių modeliu, skirtu dalykinės srities modeliavimui ir dalykinės srities analizės metodais (įvairiomis UML diagramomis, modeliuojančiomis sistemos statines ir dinamines savybes).
3. Realizuojant sistemą, buvo pritaikyta AJAX programavimo metodika, kuri pasirinkta dėl savo galimybės padaryti sistemą kur kas dinamiškesne nei naudojant tradicines web programavimo priemones.
4. Pagrindinis darbo rezultatas: sukurta patogi vartotojui, lengvai plečiama ir lengvai diegiama bei prižiūrima sistema, atliekanti web paslaugas teikiančios įmonės projektų valdymo funkcijas. Vartotojo sąsajos patogumas, diegimo ir priežiūros lengvumas bei maži kūrimo kaštai buvo vieni svarbiausių reikalavimų šiai sistemai.
5. Sukurtą įmonės vidaus valdymo sistemą planuojama diegti konkrečioje įmonėje.
6. Realizuojant sistemą sukurtas programinis „varikliukas“, pagreitinantis projektuotojo darbą ir , naudojamas kartu su AJAX programavimo metodika, paspartinantis duomenų siuntimo/gavimo procesą tarp kliento ir serverio.
7. Sukurtos sistemos savybės atitinka iškeltus sistemos naudojamumo, sąsajos patogumo, plečiamumo ir kitus kriterijus, kurie plačiau aptarti 5.4 a-me paragrafe.

7 Literatūra

1. Epicor Scala gamintojų puslapis. Žiūrėta - 2006-01-02

< <http://www.epicor.com> >

2. Epicor Scala produkto aprašymas. Žiūrėta – 2006-01-02

< <http://www.scala.lt/> >

3. Microsoft navision produkto aprašymas. Žiūrėta – 2006-01-02

< http://www.sonex.lt/sprendimai/navision/navision_imonei.asp >

4. Microsoft informacija apie įmonėje diegiamų verslo valdymo sistemų kainas.

Žiūrėta – 2006-01-02 < <http://www.microsoft.com/BusinessSolutions/pricing.msp> >

5. Informacinis bukletas apie tinkamos ERP sistemos savo įmonei parinkimą.

[žiūrėta – 2004-12-20]. Prieiga per internetą:

<http://www.cognos.com/products/epp/ep_selecting_booklet.pdf>

6. mySAP ERP produkto aprašymas

[žiūrėta – 2005-01-9]. Prieiga per internetą:

<<http://www.sap.com/index.aspx>>

7. Doc. S.Gudas Objektinės Case Technologijos

UML modeliai ir papildomos diagramos [2004.10.24]

8. I. Pashow, M.Riebisch, I.Philippow Supporting architectural Restructuring by Analyzing Feature Models, *Proceedings of the Eighth European Conference on Software Maintenance and Reengineering*, 2004 m.

9. M.Harsu, A survey on domain engineering, Institute of software systems, Tampere University of Technology. [Žiūrėta: 2004 m. Gruodžio 28d.]

10. C.Koch ,The ABC's of ERP, Dated: March 07, 2002,

Žiūrėta 2005 m. 12 20 d.prieiga per internetą

< <http://www.cio.com/research/erp/edit/erpbasics.html> >

11. **Doc. Lina Nemuraitė**, IS projektavimas ir CASE priemonės. Laboratoriniai darbai 2005. mokomoji medžiaga

12. **ERP implementation methodologies** [žiūrėta 2005 12 20] [atnaujinta 2005 03 11]. Prieiga per internetą: < http://www.sysoptima.com/erp/implementation_methodologies.php >

13. **Jesse James Garrett** Straipsnis "Ajax: A New Approach to Web Applications"
[žiūrėta – 2005-03-19]. Prieiga per internetą:
<http://www.adaptivepath.com/publications/essays/archives/000385.php>

14. **Įvairūs eksperimentai panaudojus XMLHttpRequest objektą.**
[žiūrėta – 2005-05-9]. Prieiga per internetą:
< <http://www.fiftyfoureleven.com/resources/programming/xmlhttprequest> >

15. **JSON aprašymas „Introducing JSON“**
[žiūrėta – 2005-03-9]. Prieiga per internetą:
< <http://www.crockford.com/JSON/> >

16. **Internetinės enciklopedijos straipsnis apie AJAX.** [žiūrėta 2005–10-03]
Prieiga per internetą: < <http://en.wikipedia.org/wiki/AJAX> >

17. **Navision UBP produkto aprašymas lietuvių kalba.** [žiūrėta 2005-10-03]
Prieiga per internetą: < <http://sistemas.sonex.lt/index.php/lt/valdymo-sprendimai/verslo-valdymas/NavisionUBP/> >

18. **Performance Tests: The Big SQL Shootout.** [žiūrėta 2005-12-15] prieiga per internetą:
<<http://www.pcmag.com/article2/0,1895,1172043,00.asp>>

19. **Informacija apie PHP programavimo kalbą.** [žiūrėta 2006-01-05] prieiga per internetą
< <http://www.php.net> >

20. **Informacija apie SQL.** [žiūrėta 2005-10-20] prieiga per internetą: <<http://www.mysql.com> >

8 Terminų ir santrumpų žodynas

- **WEB** – (santrumpa iš World Wide Web.) *Žiniatinklis.*
- **AJAX** – (Asynchronous JavaScript and XML). *terminas, apibrėžiantis svetainių programavimo metodologiją, naudojančią šias priemones maksimaliam interaktyvumui pasiekti*
- **PHP** – (PHP: Hypertext Preprocessor) *plačiai paplitusi dinaminė interpretuojama programavimo kalba, sukurta 1997 m. ir specialiai pritaikyta svetainių kūrimui.*
- **ERP** (Enterprise Resource Planning) – *įmonės resursų, planavimo ir valdymo sistema. Galingas įrankis apjungiantis visus verslo procesus vykstančius įmonėje.*
- **TODO** - *Vartotojui paskirtų darbų sąrašas*
- **TimeSheet** - *atliktų darbų suvestinė, naudojama darbuotojo algalapiui formuoti.*
- **Reinžinerijos procesas** – *tai procesas analizuojant seną sistemą suformuoti reikalavimus naujai sistemai.*
- **JSON** (JavaScript Object Notation) – *JavaScript objekto notacija, Lengva svoris kompiuterinių duomenų apsikeitimo formatas*
- **XML** (Extensible Markup Language) - *yra W3C rekomenduojama bendros paskirties duomenų struktūrų bei jų turinio aprašomoji kalba*
- **Feature modelling** (savybių modelis)
- **HTTP** (HyperText Transfer Protocol) - *pagrindinis metodas pasiekti informaciją pasauliniame tinkle (WWW). Pradinė protokolo paskirtis - pateikti standartinį būdą HTML puslapių skelbimui ir skaitymui.*
- **HTML** (Hypertext Markup Language) *"Hiperteksto žymėjimo kalba" tai kompiuterinė žymėjimo kalba, naudojama pateikti turinį internete. Kalbą standartizuoja W3 konsorciumas.*
- **CSS** - (Cascading Style Sheets) *naudojamų stilių kalba aprašanti dokumento pateikimą*
- **XMLHttpRequest** - *XMLHTTP yra dalis APIs kuri gali būti naudojama JavaScript, JScript, VBScript ir kitų web programavimo kalbų parsisiųsti ir manipuluoti XML duomenimis. Informacija siunčiama iš klientinės dalies į serverio dalį ir atgal.*
- **UML** (Unified Modeling Language) - *Vieninga modeliavimo ir specifikacijų kūrimo kalba, skirta specifiuoti, atvaizduoti ir konstruoti objektiškai orientuotų programų dokumentus.*

9 PRIEDAI

9.1 IT konferencija – seminaras.

2005 metais gegužės 4 dieną dalyvauta KTU festivalyje 2005 rengtame renginyje IT konferencijoje – seminare. Skaityta tema apie dinaminių internetinių aplikacijų kūrimas, pasinaudojus tokiom naujom metodologijomis, kaip AJAX, Microsoft data binding.

Už dalyvavimą konferencijoje gauta padėka (skanuotas vaizdas):



Konferencijoje skaityto pranešimo metu naudotos skaidrės pateikiamos žemiau:

Remote programming

Dinaminių web aplikacijų kūrimas

2005 Gegužės 4 diena,
KTU Festivalis 2005
IT konferencija seminaras

Vaidas Žilionis

Įžangai...

- Apibendrintai tai reikalingos papildomos informacijos gavimas neperkraunant internetinės naršyklės
 - 2005m. Vasario mėnesį tokiu principu veikiančias sistemas [Jesse James Garrett](#) pavadino AJAX
-

Kas tai AJAX

- ❑ Tai nėra technologinis variklis
 - ❑ AJAX – technologinis rinkinys leidžiantis įgyvendinti naujus sprendimus
 - ❑ AJAX šifruojasi Asynchronous JavaScript and XML
-

AJAX susideda

- ❑ XHTML ar HTML su CSS
 - ❑ Javascript ir DOM'as
 - ❑ XML, XSL ir [XMLHttpRequest](#)
-

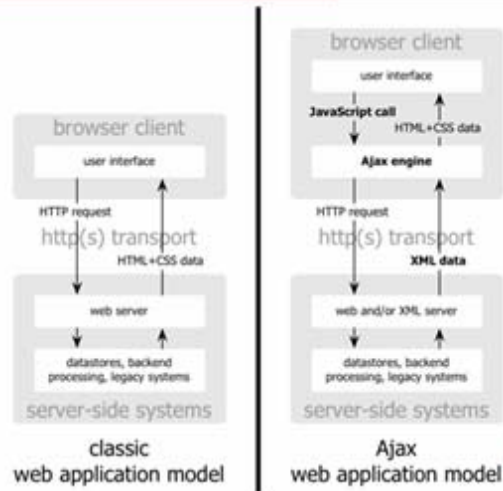
Žiūrint plačiau...

- ❑ AJAX supratimą galima išplėsti
 - ❑ Duomenys gali būti gaunami ne vien asinhroniškai, bet ir sinchroniškai
 - ❑ Nebūtinai gali būti panaudotas XmlHttpRequest, bet ir XML-RPC, Data binding...
-

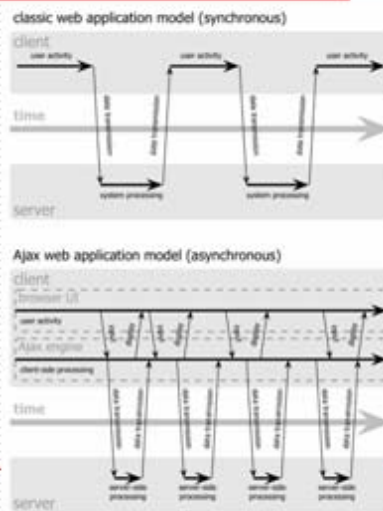
Trūkumai

- ❑ Nesaikingai naudojančioje web aplikacijoje, naršytojui tenka atsisakyti mygtukų „Back“, „Forward“, „Refresh“ įprastinio naudojimo — AJAX puslapis nėra valdomas standartinėmis priemonėmis.
 - ❑ AJAX neveikia senesnėse naršyklėse, alternatyviose (grynojo teksto, balsu skaitančiose ar telefoninėse) naršyklėse.
 - ❑ Įgyvendinimas nėra elementarus — reikia pakankamai žinių iš visų paminėtų technologijų skirtingose naršyklėse.
 - ❑ paieškos sistemos neindeksuoja (a)sinchroniškai užkrautos informacijos
-

Veikimo principai



Veikimo principai



Panaudojimas



Web Images Groups News Froogle Local^{Next} more »

ajax	3,940,000 results	Advanced Search Preferences Language Tools
aja	2,410,000 results	
ajay devgan	72,000 results	
ajay	916,000 results	
ajax amsterdam	502,000 results	results. Learn more
ajax fc	710,000 results	
ajanta	190,000 results	
ajanta caves	46,500 results	
ajax ontario	275,000 results	
ajax grips	6,060 results	

Panaudojimai (JSPAN)

□ PHP pusė

□ JavaScript pusė

```

<?php
//...

class Math {
    function add ($x, $y) {
        return $x + $y;
    }
}
//...
?>

```

```

var m = new math();
alert( m.add(2,2) );

```

Panaudojimas



Panaudojimas

The image shows a software window with a menu bar (Nauges, Redaguoti, Išbrauti, Grupės, Nauginimas, Eksportuoti, Statistika) and a toolbar (Filtrai, Visos grupės, Visi amžiai, Visi mėsėstai, Registracija, Išskirti, Vairai). The main area is a table with the following columns: #, Amžius, El. paštas, Grupė, Pildymas, Patais, Registracija, and Atlygis. The table contains 20 rows of data, with the 10th row highlighted in blue.

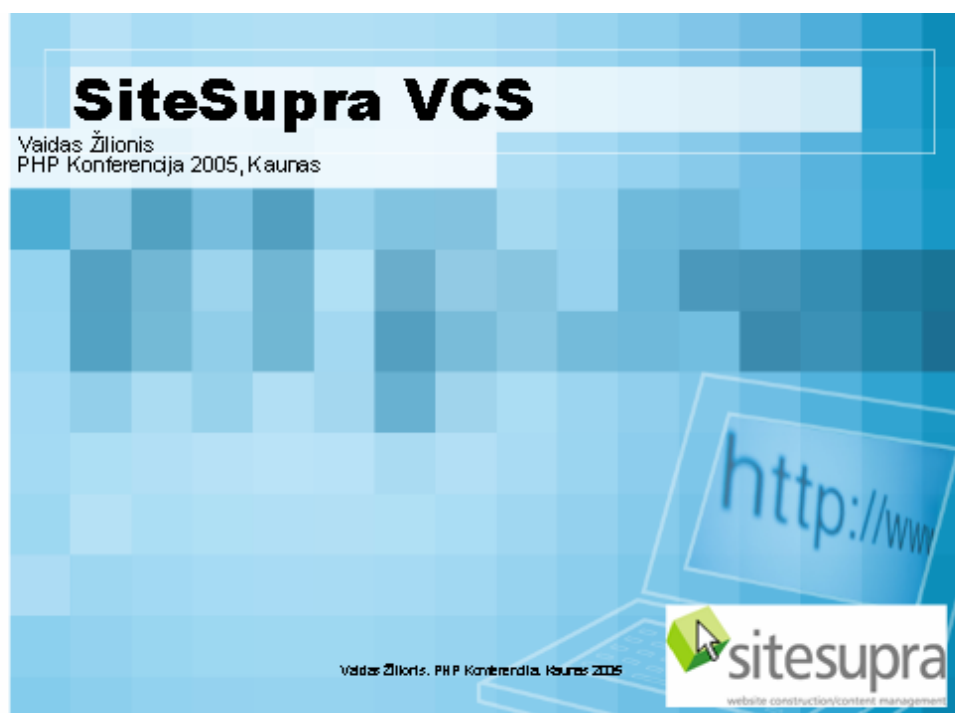
#	Amžius	El. paštas	Grupė	Pildymas	Patais	Registracija	Atlygis
3076	vaikėlyviškas gentis		Nauginimas	Kadūčiai	1991-04-05 (14)	2005-05-03 11:22:25	Tag
3078	Dobruškas Egidijus		Nauginimas	Mažiečiai	1972-08-08 (33)	2005-05-03 11:08:54	Tag
3079	Januševičius Vytautas		Nauginimas	Kauno r.	1980-04-13 (25)	2005-05-02 21:58:53	Tag
3079	Mušėnaitis Indrė		Nauginimas	Mažiečių r.	1987-04-09 (18)	2005-05-02 21:54:10	Tag
3079	Uchutytė Rūta		Nauginimas	Kaunas	1993-08-05 (12)	2005-05-02 21:05:26	Tag
3079	Kaimo vietos TOP 20		Nauginimas	Leidžia r.	1992-01-18 (13)	2005-05-02 19:27:37	Tag
3080	Kaimo vietos TOP 20		Nauginimas	Leidžia r.	1992-01-18 (13)	2005-05-02 19:27:37	Tag
3080	Čerėnaitis Mantas		Nauginimas	Telšiai	1992-02-19 (13)	2005-05-02 18:05:53	Tag
3080	Udošius Vincentas		Nauginimas	Mažiečių r.	1989-01-05 (16)	2005-05-02 18:05:48	Tag
3084	A. Šėva		Nauginimas	klajėdai	1990-09-28 (15)	2005-05-02 11:35:15	Tag
3085	Kalnaitis Laura		Nauginimas	Vilnius	1988-09-24 (17)	2005-05-03 22:32:19	Tag
3084	BLAZNITIS RYTIS		Nauginimas	Kaunas	1968-07-17 (37)	2005-05-03 21:06:57	Tag
3083	Konkenavičius Martynas		Nauginimas	Jurbarkas	1993-05-29 (12)	2005-05-03 20:03:10	Tag
3082	Čerėnaitis Rūta		Nauginimas	Jonava	1991-04-21 (14)	2005-05-03 20:49:23	Tag
3082	Kalnaitis Audrius		Nauginimas	Kaunas	1970-02-20 (34)	2005-05-03 20:08:52	Tag
3082	Konkenavičius Mantas		Nauginimas	Vilniaus r.	1989-07-19 (16)	2005-05-03 19:05:48	Tag
3083	Dugnaitis Karolina		Nauginimas	Elektrėnai	1993-04-04 (12)	2005-05-03 19:14:25	Tag
3082	Korvėlis Mantas		Nauginimas	Vilnius	1994-07-14 (11)	2005-05-03 18:20:18	Tag
3087	Konkenavičius Algirdas		Nauginimas	Kaunas	1988-12-07 (16)	2005-05-03 13:32:23	Tag
3086	Čerėnaitis Deividas		Nauginimas	Vilnius	1994-07-28 (11)	2005-05-03 11:32:33	Tag
3085	ankstyta Karolyne		Nauginimas	Vilnius	1991-08-05 (14)	2005-04-30 23:02:19	Tag
3084	Kačanauskis Beatričė		Nauginimas	Tauragė	1988-02-12 (17)	2005-04-30 22:05:07	Tag
3083	Konkenavičius Viktorija		Nauginimas	Šiauliai	1989-04-05 (17)	2005-04-30 22:22:40	Tag

Nuorodos

- ❑ <http://jibbering.com/2002/4/httprequest.html>
 - ❑ <http://www.adaptivepath.com/publications/essays/archives/000385.php>
 - ❑ <http://www.modernmethod.com/sajax/>
 - ❑ <http://www.fiftyfoureleven.com/resources/programming/xmlhttprequest>
 - ❑ <http://jpspan.sourceforge.net/wiki/doku.php>
 - ❑ http://msdn.microsoft.com/workshop/author/databind/data_binding.asp
 - ❑ <http://www.google.com>
-

9.2 PHP konferencija

2005 metų lapkričio 26 dieną dalyvauta PHP antrojoje konferencijoje. Pristatyta turinio valdymo sistema „SiteSupra“. Jos veikimas pagrįstas, kaip ir mūsų sistemos, AJAX metodologija. Konferencijoje buvo pasakota ne tik apie šios turinio valdymo sistemos privalumus, trūkumus, bet ir buvo apibūdintas veikimas, pristatytos galimybės kaip galima realizuoti savo modulius ir kaip galima pritaikyti naujus komponentus.



Vietoj Intro

- SiteSupra autoriai Latvijos internetinių sprendimų kompanija “VidInfra”
- 2000m. SiteManager
- 2001m. SiteSupra 1.4
- 2002m. SiteSupra 2.0
- 2004m. SiteSupra 3.0

Vidas Ziloris, PHP Konferencija, Kaunas 2005

2

Reikalavimai

- Linux, Windows operacinė sistema
- Apache HTTPD nuo 1.3.XX,
(mod_rewrite on)
- PHP 4.3.x
 - Zlib
 - MySQL
 - Zend Optimizer
- MySQL 4.0.1x
- Internet Explorer nuo 5.5 versijos

Vidas Ziloris, PHP Konferencija, Kaunas 2005

3

SiteSupra savybės (1)

- Tai turbūt vienintelė vizualinė sistema išnaudojanti WYSIWYG principą ne vien turinio formatavimui, bet ir svetainės dizaino kūrimo, bei funkcionalumo
- SiteSupra pagrįsta XMLHttpRequest technologija. Vienintelė forma administravime – failų įkėlimo.
- Human Links palaikymas
- Galingas debuggeris
- SiteSupra puslapis – lyg mozaika sudaryta iš daugia kartinio panaudojimo elementų (blocks)

Vaidas Žiloris, PHP Konferencija, Kaunas 2005

Sitesupra savybės (2)

- Atskiros administravimo bei viešos svetainės dalies duomenų bazės lentelės.
- Patvirtinimu pagrįsta sistema (approval workflow)
- Versijavimas
- Teisių sistema
- Šablonai
- Daugelio kalbų palaikymas
- Planuotas publikavimas
- Įvairiausi moduliai (naujienos, balsavimas, vartotojų sistema ir t.t.)

Vaidas Žiloris, PHP Konferencija, Kaunas 2005

Vizualinė aplinka

- Svetainės administravimas labai suprantamai redaguojamas, pasirenki bloką kurią nori redaguoti, paspaudi klavišą ir keisti tekstus ar kitus parametrus. Visi pakeitimai atvaizduojami administravime.
- Galima ne tik keisti lango rezoliucija, bet ir priartinti.



Vaidas Žilioris, PHP Konferencija, Kaunas 2005

6

Svetainės kūrimas

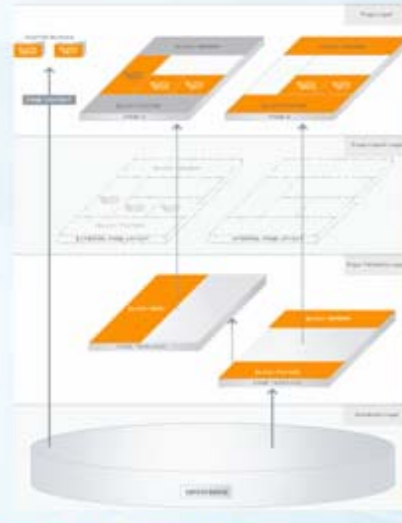
- Svetainę galima kurti vizualiai. Išskiriami trys kūrimo rūšys
 - Be šablonų bet su pilnu Drag && drop režimu. Blokai dedami į vieną lentelę.
 - Su daliniais šablonais, bet išlaikant drag && drop galimybę. (Naudoja ta pati SiteSupra table)
 - Su šablonais iš failų, bet prarandama drag && drop galimybė.

Vaidas Žilioris, PHP Konferencija, Kaunas 2005

7

Puslapių šablonai (2)

- Šablonai gali būti kuriam, kaip puslapiai
- Šablonas gali paveldėti "tėvo" šablono kintamuosius, o reikiamus parašyti
- Suderinus su puslapių teisėmis galimas rezultatas teisė į bloko redagavimą



Vaidas Žilioris, PHP Konferencija, Kaunas 2005

9

Blokai (1)

- Pilnai parametrizuoti
- Galimybė išsirinkti daugelį tipų parametrų (tekstas, nuoroda, ilgas tekstas, XHTML redaktorius, sąrašas...)
- Kiekvienas elementas gali būti validuojamas

Vaidas Žilioris, PHP Konferencija, Kaunas 2005

10

Blokai (2)

- Blokas gali priklausomai nuo nustatymų kešuojamas arba ne.
- Jei visi puslapyje blokai yra kešuojami, tai automatiškai ir visas puslapis bus kešuojamas.

Vaidas Žiloris, PHP Konferencija, Kaunas 2005

11

Debuggeris

- Įdiegtas tai public tiek edit pusėje
- Su edit pusėje debuggeriu galima magrinėti ne tik užklausos greičius ar bloko vygdymo laiką, bet ir nagrinėti JavaScript masyvus ar net visą objektą

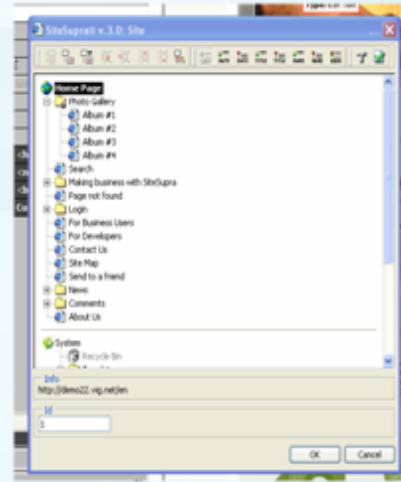


Vaidas Žiloris, PHP Konferencija, Kaunas 2005

12

Svetainės medis

- Lengva navigacija svetainės puslapių
- Kuriami puslapiai paveldi tėvo šabloną ir turinį
- Galimybė rikiuoti
- Perkelti į kitą ašaką

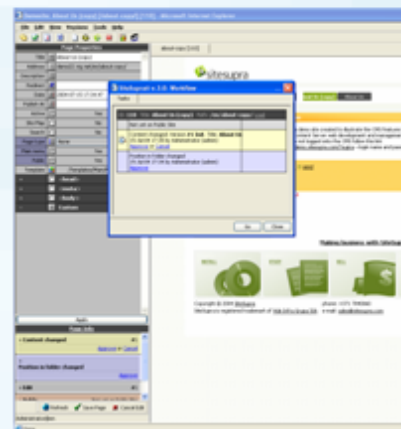


Vaidas Žilionis. PHP Konferencija. Kaunas 2005

13

Workflow

- Patvirtinti keitimus reikia:
 - Publikuojant naują puslapį
 - Keičiant puslapio struktūrą
 - Keičiant poziciją medyje



Vaidas Žilionis. PHP Konferencija. Kaunas 2005

14

Versijavimas

- Galima sukurti ne vieną to pačio puslapio versiją, ir pasirinkti kada kiekvienas būtų publikuotas
- Arba tai įvykdyti rankiniu būdu



Vaidas Žilioris, PHP Konferencija, Kaunas 2005

15

Teisių sistema

- Kiekvienas vartotojas gali turėti teisę redaguoti puslapį, bei jį publikuoti
- Galima išskirti į skirtingus vartotojo tipus
 - pagrindinis administratorius
 - atsakingas už tam tikras puslapio zonas vartotojus
 - tik naudotis tam skirtais moduliais

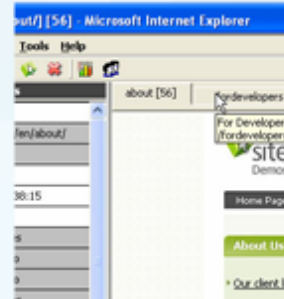


Vaidas Žilioris, PHP Konferencija, Kaunas 2005

16

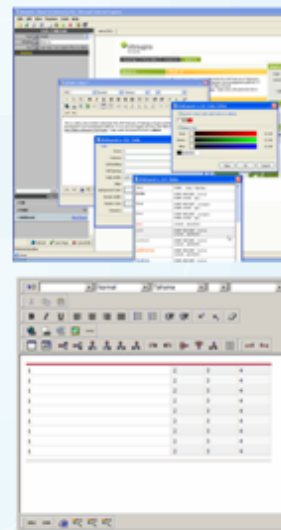
Galima redaguoti daug puslapių

- Turbūt unikali savybė redaguoti vienu metu daugiau nei vieną puslapį.



WYSIWYG redaktorius

- Pilnas teksto formatavimas
- Galimybė naudoti macro komandomis `{link(332)}`
- Lengvas darbas su anhorais
- Galimybė kurti popusus
- Galimybė taikyti savo komandas, pvz filtus lentelei



Savų modulių gamyba

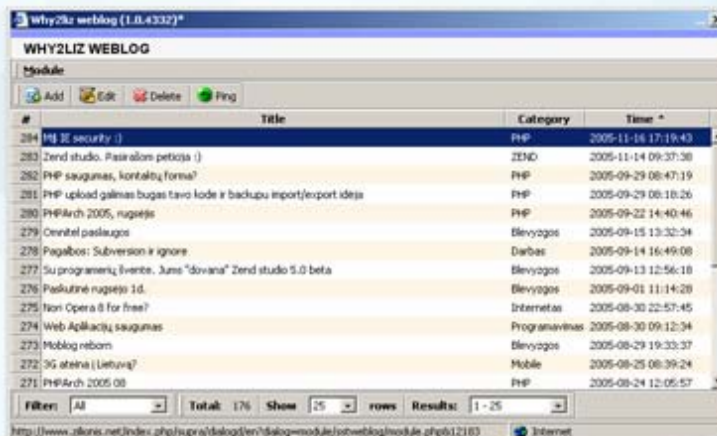
- Galima lengvai kurti savus modulius kurie administruoja tam tikrus duomenis
- Pasinaudoti galima visais elementais tiek vizualiniais (mygtukai, tabai, dhtml dropdown menu), tiek programiniais (Debuggeris, "SendData" ir pan)



Vaidas Žilioris. PHP Konferencija. Kursas 2005

19

Savų komponentų gamyba (1)



Vaidas Žilioris. PHP Konferencija. Kursas 2005

20

Savų komponentų gamyba (2)

Vote Manager

Create Edit Delete Filter: ALL

Question	Vote type	Date From	Date To	Active
Where are you from?	normal	2005-09-12 00:22:50	Unlimited	✗
What programming languages do you know?	multi	2005-09-12 00:00:00	2005-09-17 00:00:00	✓
What module do you need?	normal	2005-09-04 00:21:12	Unlimited	✓

Close

Vote Manager : Edit Vote

ID: 1 Vote active: Active

Vote question: What programming languages do you know?

Vote type: User can select only one answer (normal) User can select many answers (multi)

Start date: 2005-09-12 00:00:00 End date: 2005-09-17 00:00:00 Clear

Vote control: Code: One week Save

Answers	Votes
PHP	2
JavaScript	2
C++	1
SQL	4
Other	2
Perl	2
Not listed...	1

Close

Václav Zlámal, PHP Konferencia, káure 2005

21

SiteSupra planai...

- PHP5
- Naujas sistemos variklis (framework)
- Populiariausių naršyklių palaikymas
- Naujas Grafinis vartotojo interfeisas (GUI)
- Pilna Drag & Drop aplinka
- Pagerintas šablonų variklis
- ...

Václav Zlámal, PHP Konferencia, káure 2005

22

SiteSupra Lietuvoje



Vaidas Žilionis, PHP Konferencija, Kaunas 2005

24

Nuorodos

- www.sitesupra.com – pagrindinė svetainė
- Manuals.sitesupra.com
- Forum.sitesupra.com
- Plugins.sitesupra.com
- Ssdn.sitesupra.com

Vaidas Žilionis, PHP Konferencija, Kaunas 2005

25