

ŠIAULIŲ UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
INFORMATIKOS KATEDRA

Aurimas Strikulis
Informatikos (gamybos) magistro specialybės II kurso studentas

Darbo organizavimo valdymo sistema

BAIGIAMASIS MAGISTRO DARBAS

Darbo vadovas:
Asist. G. Felinskas

Darbo recenzentas:
Lekt. L. Kaklauskas

Šiauliai, 2006-2007 m.m.

Turinys

Darbo organizavimo valdymo sistemos.....	4
DOV sistemų uždaviniai	5
Planavimas	5
Informacijos teikimas.....	6
DOV sistemų tipai	8
Standartinės darbalaukio klientinės aplikacijos	8
Tinklinės aplikacijos	8
Personalinės aplikacijos	9
Vieno vartotojo sistemos.....	9
Bendradarbiavimo sistemos	9
Integruotos sistemos.....	9
Nespecializuotos priemonės.....	10
DOV sistemų analizė.....	11
dotProject	12
GForge	13
Trac	14
Bugzilla	15
Sistemų galimybių ir funkcionalumų palyginimas	16
Galutinis sistemos pasirinkimas	18
Bugzilla diegimo ir pritaikymo projektas	19
Programinės įrangos pasirinkimas.....	19
Sistemos pakeitimai ir nauji funkcionalumai.....	20
Papildomi laiko apskaitos laukai.....	20
Datos įvedimo supaprastinimas	20
Kalendoriaus funkcionalumas	21
Darbų planų ataskaitos	22
„Deadline“ automatinis apskaičiavimas.....	22
Atliktų darbų ataskaitos.....	23
Komentarų įvedimas „assignee“ vardu	23

Komentarų redagavimas.....	23
Komponentų funkcionalumo papildymas	24
Veiksmų planas	25
Darbų eigos aprašymas.....	26
Darbų eigos grafas	26
Problemos ir jų sprendimai.....	27
„Bugzilla“ ir lietuviybė	27
Prisegamų failų limitas.....	27
Ruošinių sistema	27
„Bugzilla“ klaidos	28
Galutinis projekto stovis.....	29
Papildomi laiko apskaitos laukai.....	29
Datos įvedimo supaprastinimas	30
Kalendoriaus funkcionalumas	30
Darbų planų ataskaitos	31
„Deadline“ automatinis apskaičiavimas.....	31
Atliktų darbų ataskaitos.....	31
Komentarų įvedimas „assignee“ vardu	32
Komentarų redagavimas.....	33
Komponentų funkcionalumo papildymas	33
Darbo rezultatų analizė	35
„Bugzilla“ funkcinė schema.....	36
„Bugzilla“ duomenų bazės pakeitimai	38
Skriptų ir ruošinių modifikacijos	40
Išvados	42
Rodyklė	43
Priedas	44
Anotacija (summary)	44
Lietuviškai.....	44
In English	44
Pažyma iš UAB “Baltic Data Center“	45

Darbo organizavimo valdymo sistemos

Darbo organizavimo valdymo (DOV) sistemomis vadinamos sistemos, naudojamos įmonės ar organizacijos darbuotojų darbo valdymui, apskaitai, planavimui organizuoti. Iš principo tai labai plati sfera, todėl šios sistemos turi ne vieną terminą, skirtą joms apibūdinti. Vienas iš dažniausių terminų - projektų valdymo sistema. Tokią terminologijos gausos situaciją lemia tai, kad šias sistemas labai sunku grupuoti, klasifikuoti, nes dažnai jos viena nuo kitos skiriasi kokiais nors specifiniais funkcionalumais, darbo principu ar galimybėmis.

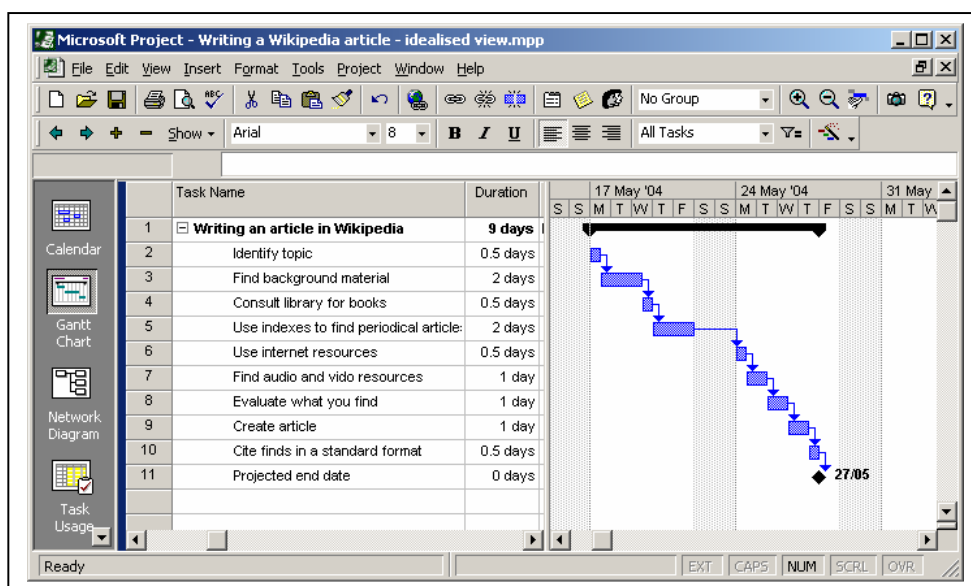
Iš esmės šios sistemos naudojamos didesnėse įmonėse arba kolektyvuose, užsiimančiuose dideliais projektais, kuomet labai svarbus yra projekto darbų planavimas, kuris gali nulemti viso projekto sėkmę.

DOV sistemų uždaviniai

Galima išskirti dvi pagrindines užduotis, kurias atlieka DOV sistemos: planavimas, informacijos teikimas.

Planavimas

Viena pagrindinių užduočių vykdant vieną ar kitą projektą yra įvykių (darbų, užduočių ir pan.) paskirstymas jų pačių ir laiko atžvilgiu. Ir ši užduotis, priklausomai nuo to kaip ir kokie įrankiai tam yra naudojami, gali būti gan sudėtinga. Planavimo (ang. Scheduling) procese dažnai yra naudojamos Ganto [1] ar kitokios schemas, grafiškai atvaizduojančios įvykius laiko atžvilgiu.



Microsoft Project 2000, rodantis ganto schema

Planavimo metu gali būti naudojami įvairūs metodai ir algoritmai, tokie kaip Kritinis kelias [2] arba Kritinė grandinė [3].

Planavimas apima:

- Projekto įvykių išskyrimas, jų surikiavimas.

Pradedant dirbti prie naujo projekto, visų pirma išskiriami įvykiai. Juos išskyrus reikia nustatyti jų tarpusavio ryšius bei įtaka pačiam projektui. Tarpusavio ryšių nustatymas yra svarbus žingsnis, kuris palengvina užduočių surikiavimą, paskirstymą ir kurio pagalba galima išskirti svarbiausius projekto darbus.

- Užduočių trukmių (imlumų) nustatymas.

Tam, kad nustatyti, kiek projekto įgyvendinimas užtruks, būtina nustatyti atskirų to projekto darbų imlumus. Imlumų nustatymas taip pat gali padėti resursų paskirstymui,

nusprendžiant kiek tų resursų reikėtų skirti vienai ar kitai užduočiai atlikti (priklausomai nuo to kokios svarbos ir įtakos visam projektui tas darbas yra).

- Užduočių kritinių datų nustatymas.

Norint užtikrinti projekto įvykdymą laiku, reikia nustatyti svarbiausių darbų kritines datas (ang. Deadline). Šios datos padeda paskirtiems darbuotojams orientuotis ir pasiskirstyti prioritetus.

- Resursų paskirstymas.

Dažniausiai resursai, kuriuos reikia paskirstyti – darbuotojai. Šis žingsnis taip pat yra gan reikšmingas, nes būtent nuo to, kaip sėkmingai bus paskirstytos užduotys, priklauso tų užduočių imlumų ir kritinių datų neviršijimas. Resursų paskirstymo procese dalyvaujantys asmenys turi būti gerai susipažinę su projekto užduotimis ir gan gerai pažinoti darbuotojus (kad sugebėtų racionaliai paskirstyti jų darbo laiką projekto užduotims atsižvelgiant į kiekvieno darbuotojo kompetenciją).

- Kelių (ar keliolikos) projektų užduočių prioritizavimas (siekiant patenkinti įvairius tų projektų reikalavimus).

Šis procesas aktualus įmonėms/organizacijoms, vienu metu vykdančioms kelis projektus. Atsižvelgiant į pačių projektų prioritetus, juos sudarančius darbus bei resursų išteklius, yra vykdomas darbų prioritizavimas jų tarpusavio atžvilgiu.

Informacijos teikimas

DOV sistemos turi teikti nemažai informacijos daugeliui žmonių (t.y. būti informatyvios), kad pateisintų laiko, skiriamo joms naudotis, kiekį. Dažniausi keliama reikalavimai:

- Užduočių sąrašai žmonėms ir panaudos planai kitiems (ne žmogiškiesiems) resursams.

Sistema turi turėti funkcionalumus, leidžiančius žmonėms, besinaudojantiems sistema, gauti sąrašą užduočių, kurias jie turi atlikti. Pageidautina kad tos užduotys būtų pateiktos kaip planas (grafinis), sudarytas pagal žmogaus darbų parametrus, jo asmeninį kalendorių ir pan. Analogiškai turėtų būti pasiekiami ir nežmogiškųjų resursų panaudojimo planai.

- Bendra informacija apie užduotį ir kiek laiko jai įvykdyti skirta.

Žmogus turi vienu ar kitu būdu matyti visą reikiamą informaciją apie užduotį bei įvairius tos užduoties laiko parametrus (kiek valandų skirtą, kritinė data ir pan.)

- Išankstiniai perspėjimai apie pavojus projektui.

Įvedinėjant užduotis/projektus, koreguojant jų laiko parametrus vartotojas turi būti informuojamas apie šių veiksmų poveikį visam projektui, jo planui. Taip pat pageidautina

kad sistema galėtų siųsti pranešimus projektų vadovams apie potencialiai vėluojančius darbus ir pan.

- Realių veiksmų įrodančioji informacija.

Sistemoje turi būti įvedinėjama informacija, kuri rodytu darbuotojų veiksmus atliekant darbus (valandų/veiksmų/komentarų pavidalu), kurią galėtų pasiekti projektų vadovai darbų analizei.

- Istorinė informacija apie projekto eigą.

Visa su (baigtu) projektu susijusi informacija turi būti prieinama ir tam projektui pasibaigus. Tokia informacija gali padėti įgyvendinant naujus projektus, nes dalis informacijos turi išliekamąją vertę (pavyzdžiui konkrečių problemų sprendimas ar pan.). Tokia informacija taip pat labai naudinga, kai įmonėje/organizacijoje nebėra su ankstesniais projektais dirbusių žmonių, t.y. išlieka jų patirtis.

Neretai DOV sistemos atlieka ir bendravimo tarp vartotojų terpės vaidmenį, t.y. leidžia sistemos vartotojams bendrauti tarpusavyje nuotoliniu būdu (pvz. siuntinėjamų laiškų ar forumų forma, taip pat sistemoje gali būti integruotos ir kitokios bendravimą palaikančios technologijos kaip realaus laiko susirašinėjimas [\[4\]](#) ar pan.).

DOV sistemų tipai

Pagal tai, kaip tos DOV sistemos yra suprojektuotos ir realizuotos, jas galima išskirstyti į kelias grupes.

Standartinės darbalaukio klientinės aplikacijos

DOV sistemos, kurios realizuotos kaip darbalaukinės aplikacijos (ang. Desktop application) – tai tiesiog sistemos, einančios įprastinių klientinių programų (ang. Executables) pavidalu. Tokios sistemos suteikia vartotojui daugiau įprastą ir suprantamą aplinką. Dažniausiai tokios aplikacijos saugo visą informaciją vietiniuose (to vartotojo sistemoje laikomuose) failuose, nors kai kurios suteikia galimybę keisti duomenimis su kitais vartotojais. Net ir failais paremtos aplikacijos gali leisti vartotojams naudotis tais pačiais resursais (failais), jei šie failai yra saugomi tinkliniame diske ir į juos vienu metu kreipiasi tik vienas vartotojas. Taip pat pasitaiko aplikacijų, kurios visus duomenis saugo vienoje centralizuotoje duomenų bazėje.

Darbalaukinės aplikacijos gali būti kuriamos daugiaterpinėms operacinėms sistemoms arba jos gali būti rašomos kaip multiplatforminės aplikacijos [\[5\]](#).

Tinklinės aplikacijos

Dažnai DOV sistemos būna suprojektuotos kaip tinklinės (ang. Web application) aplikacijos. Tokios aplikacijos, skirtingai nuo darbalaukinių aplikacijų, neturi sukompilijuotos paleidžiamosios programos (exe), jos yra atidaromos ir jomis naudojamosi per internetines naršyklės (ang. Browsers), pavyzdžiui „Microsoft Internet Explorer“ ar „FireFox“. Kaip ir visos tinklinės aplikacijos, tinklinės DOV sistemos turi joms būdingus privalumus ir trūkumus:

- Jos pasiekiamos iš bet kurio kompiuterio nediegiant jokios papildomos programinės įrangos.
- Lengva reguliuoti priėjimą (pagal IP adresus, tinklinius identifikatorius ir pan.).
- Iš prigimties multi-vartotojiškos (vienu metu gali naudotis keli vartotojai).
- Vienintelė programos instaliacija ir versija prižiūrėti.
- Dažniausiai lėčiau reaguoja į vartotojų veiksmus nei darbalaukinės aplikacijos.
- Grafinė vartotojo sąsaja daugiau apribota (galimybėmis) nei darbalaukinės aplikacijos.
- Informacija nepasiekama jei serveris (ar vienas iš serverių), ant kurio aplikacija veikia fiziškai, neveikia.
- Kai kurių sistemų paketai gali būti naudojami netinkliniu (ang. Offline) režimu.

Personalinės aplikacijos

Tokios DOV sistemos, kurios yra realizuotos kaip personalinės (arba asmeninės) aplikacijos, dažniausiai yra naudojamos buities ar namų ruošos darbų valdymui. Iš principo tokios sistemos (pagal paskirtį) persidengia su vieno vartotojo aplikacijomis, tik personalinės aplikacijos dažnai turi paprastesnę vartotojo sąsają.

Vieno vartotojo sistemos

DOV sistemos, realizuotos kaip vieno vartotojo (ang. Single user) aplikacijos, yra kuriamos su prielaida, kad vienu metu darbų/projektų planą redaguos tik vienas vartotojas. Tokios sistemos naudojamos mažose įmonėse/organizacijose ar įmonėse/organizacijose, kuriose darbų/projektų organizavimu ir planavimu užsiima tik keli žmonės. Darbalaukio klientinės DOV sistemos dažnai patenka į šią kategoriją.

Bendradarbiavimo sistemos

Šios DOV sistemos yra suprojektuotos taip kad vienu metu keli ar keliasdešimt vartotojų gali redaguoti skirtingas to paties darbo/projekto plano sekcijas, pavyzdžiui, redaguoti sekcijas, už kurias jie yra asmeniškai atsakingi, ir šių sekcijų redagavimas keičia bendrą darbo/projekto planą. Tinklinės, įskaitant ir ekstranetines [6], aplikacijos dažniausiai papuola į šią grupę, tačiau jos turi apribojimą, nes jomis naudotis gali tik tie vartotojai, kurie turi prieigą prie „gyvo“ interneto ryšio. Šiai problemai apeiti yra kuriamos klientas-serveris tipo sistemos, kurios turi darbalaukio klientines dalis, įdiegiamas sistemos vartotojams. Šios klientinės dalys periodiškai replikuoja (susikuria vietines kopijas) reikalingus duomenis vartotojo kompiuteryje iš centrinio serverio bei nusiunčia vartotojo naujai įvestus duomenis į šį serverį.

Integruotos sistemos

Integruotos sistemos apjungia darbų/projektų valdymą ar planavimą su daugeliu kitų įmonės/organizacijos darbo aspektų. Pavyzdžiui [PHPPProject](#) projektai turi klaidų (ang. Bugs) valdymo sekciją, priskirtą prie kiekvieno projekto, projekto klientų sąrašas iš esmės atlieka klientų aptarnavimo valdymo (ang. CRM – Client relationship management) modulio vaidmenį, o kiekvienas sistemos vartotojas turi savo asmeninį užduočių sąrašą, kalendorių ir bendravimo funkcionalumą pririštą prie konkrečių projektų.

Panašiai specializuotos priemonės, tokios kaip [SourceForge](#), integruoja sistemas su versijų (šaltinių) kontrolės (ang. Source control arba CVS) ir klaidų valdymo sistemomis, tokiu būdu visa su projektais dirbančių vartotojų informacija integruojama į bendrą produktą.

Nespecializuotos priemonės

Nors specializuotos priemonės yra gan populiaros ir aktyviai propaguojamos jų gamintojų, tačiau yra gan daug ir įvairių nespecializuotų programinės (ir ne tik programinės) įrangos priemonių, naudojamų darbų/projektų planavimui ir vykdymui. Kaip vienas iš pavyzdžių – kalendorinės sistemos, jos dažnai gali pasiūlyti paprastą ir patogų darbų/projektų planavimo procesą.

DOV sistemų analizė

Vienas iš išskeltų reikalavimų yra nemokama atviro kodo tinklinė aplikacija (žiūrėti [reikalavimų](#) sekciją, 20 pusl.), tad analizuojamos bus tik nemokamos atviro kodo tinklinės DOV aplikacijos.

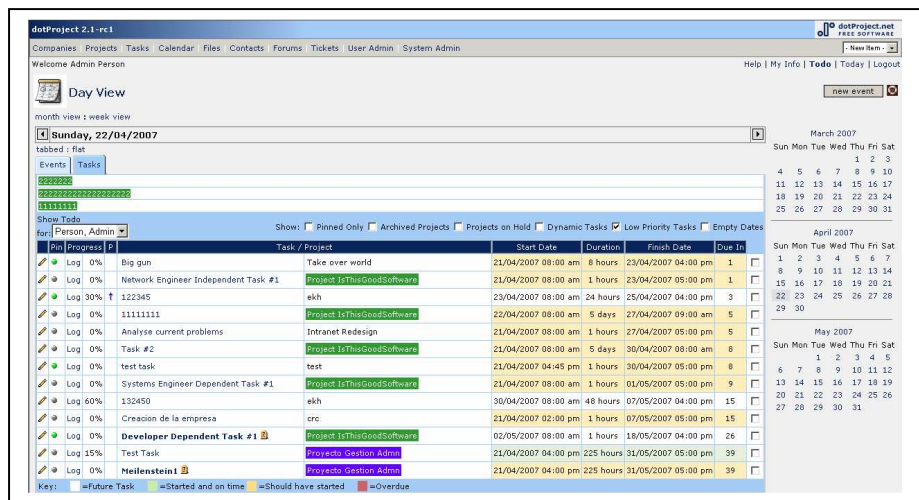
Pirminei analizei buvo atrinktos šios sistemos:

1. dotProject.
2. GForge.
3. GNU Savannah
4. LibreSource
5. Trac
6. Wrike
7. Bugzilla

Sąrašas sudarytas pagal sistemų populiarumą ir vartotojų atsiliepimus apie šias sistemas. Tačiau po pirminės analizės buvo atmestos šios sistemos: GNU Savannah – nėra galimybės turėti savo sistemos kopijos, naudojama viena visiems vartotojams sistema, Wrike – taip pat naudojama viena sistema, LibreSource – dėl Q Public licenzijos [\[7\]](#).

dotProject

Ši sistema pradėta kurti 2000 metais, o jos pirmoji versija oficialiai išleista 2005 metais. dotProject pagrindinė idėja (kurios vis dar laikomasi ir dabar) – atviro kodo alternatyva Microsoft ir kitų kompanijų brangiems komerciniams produktams.



Sistema parašyta PHP ir JavaScript kalbomis, naudoja ADOdb arba MySQL duomenų bazes. Ši sistema unikali tuo, kad ją kuria ir palaiko savanoriai, neatstovaujantys jokiai kompanijai (nors yra kompanijų, remiančių šį projektą).

Pagrindinė dotProject sistemos paskirtis – projektų valdymas. Sistemoje įgyvendinta:

- Vartotojų administravimas.
- E-paštu paremta problemų kortelių (ang. Trouble ticket) sistema.
- Klientų-kompanijų valdymas.
- Projektų sąrašas.
- Hierarchinė užduočių sistema.
- Ganto schemos.
- Kalendorius.
- Diskusijų forumas.
- Failų saugojimo sistema.

GForge

[GForge](#) iš pradžių buvo kurta kaip tinklinė projektų valdymo – bendradarbiavimo

sistema skirta

[SourceForge](#) projektui.

Tačiau kompanija VA

Linux (dabar [VA](#)

[Software](#)), susidūrusi su

finansiniais sunkumais,

nusprendė sustabdyti šios

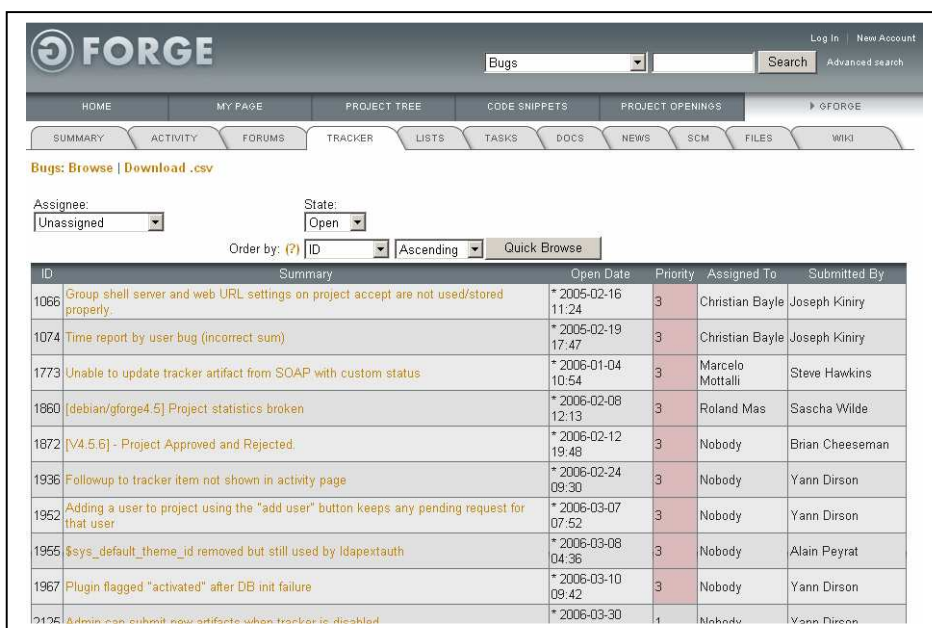
sistemos vystymą. Po

kelerių metų, vienas iš sistemos kūrėjų – Tim Perdue, nusprendė atgaivinti sistemą. Pasinaudojęs paskutiniąja išleista sistemos versija (2.5) jis perleido ją, pavadinęs GForge.

Sistemoje realizuota:

- CVS.
- Forumas.
- Klaidų valdymo modulis.
- Vaidmenimis paremtas sistemos vartotojų valdymo modulis.
- Užduočių valdymo modulis.
- Ganto schemų modulis (integruotas su užduočių valdymo moduliu).
- Dokumentavimo valdymo modulis.
- WiKi [\[8\]](#).
- Integruota e-pašto modulis (jo pagalba galima siuntinėti kodo pakeitimus, dirbti su užduotimis, atsakinėti į forumo komentarus).

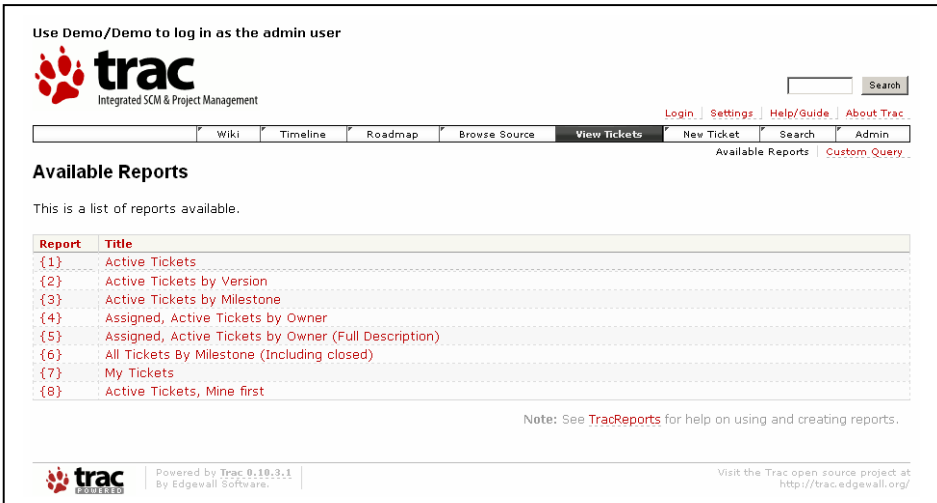
Sistemai taip pat yra prikurta nemažai įskiepių (ang. Plugins), papildančių ją įvairiais funkcionalumais.



ID	Summary	Open Date	Priority	Assigned To	Submitted By
1066	Group shell server and web URL settings on project accept are not used/stored properly.	* 2005-02-16 11:24	3	Christian Bayle	Joseph Kiniry
1074	Time report by user bug (incorrect sum)	* 2005-02-19 17:47	3	Christian Bayle	Joseph Kiniry
1773	Unable to update tracker artifact from SOAP with custom status	* 2006-01-04 10:54	3	Marcelo Mottali	Steve Hawkins
1860	[debian/gforge4.5] Project statistics broken	* 2006-02-08 12:13	3	Roland Mas	Sascha Wilde
1872	[V4.5.6] - Project Approved and Rejected.	* 2006-02-12 19:48	3	Nobody	Brian Cheeseman
1936	Followup to tracker item not shown in activity page	* 2006-02-24 09:30	3	Nobody	Yann Dirson
1952	Adding a user to project using the "add user" button keeps any pending request for that user	* 2006-03-07 07:52	3	Nobody	Yann Dirson
1955	\$sys_default_theme_id removed but still used by ldapextauth	* 2006-03-08 04:36	3	Nobody	Alain Peyrat
1967	Plugin flagged "activated" after DB init failure	* 2006-03-10 09:42	3	Nobody	Yann Dirson
2125	Admin can submit new artifacts when tracker is disabled	* 2006-03-30	1	Nobody	Yann Dirson

Trac

[Trac](#) pirmą kartą buvo išleista 2004 metais. Ją kuria ir palaiko kompanija [Edgewall Software](#). Sistema parašyta Python programavimo kalba o duomenų saugojimui naudojamos SQLite, PostgreSQL arba MySQL



The screenshot shows the Trac web interface. At the top, it says "Use Demo/Demo to log in as the admin user". Below that is the Trac logo and "Integrated SCM & Project Management". There is a search bar and navigation links for "Login", "Settings", "Help/Guide", and "About Trac". A menu bar includes "Wiki", "Timeline", "Roadmap", "Browse Source", "View Tickets" (which is highlighted), "New Ticket", "Search", and "Admin". Below the menu is the "Available Reports" section, which states "This is a list of reports available." and contains a table of reports:

Report	Title
{1}	Active Tickets
{2}	Active Tickets by Version
{3}	Active Tickets by Milestone
{4}	Assigned, Active Tickets by Owner
{5}	Assigned, Active Tickets by Owner (Full Description)
{6}	All Tickets By Milestone (Including closed)
{7}	My Tickets
{8}	Active Tickets, Mine first

At the bottom of the screenshot, it says "Powered by Trac 0.10.3.1 By Edgewall Software" and "Visit the Trac open source project at http://trac.edgewall.org/".

duomenų bazės. Trac nėra unikali idėja, joje sudėta daug funkcionalumų, „pasiskolintų“ iš kitų sistemų, o pagrindinis šių idėjų šaltinis buvo sistema [CVSTrac](#).

Pagrindiniai sistemos komponentai:

- Incidentų valdymo modulis. Tai pagrindinis sistemos modulis. Visi incidentai, klaidos, pakeitimų prašymai yra registruojami per vadinamąsias korteles.
- WiKi [\[8\]](#) modulis. WiKi mechanizmas yra pilnai integruotas visoje sistemoje. Trac palaiko standartines WiKi komandas, taip pat turi savo papildomų komandų, leidžiančių daryti nuorodas į korteles, ataskaitas, sistemoje patalpintus failus ir pan.
- Subversion modulis. Šis modulis realizuoja sistemos SCM (Source code management – kodo kontrolės) funkcionalumą. Trac gali veikti ne tik su Subversion bet ir su kitomis SCM priemonėmis.

Bugzilla

„Bugzilla“ – tai klaidų/defektų šalinimo sistema. Kaip galima nesunkiai atspėti iš sistemos pavadinimo, ji priklauso „Mozilla“ projektui. „Bugzilla“ buvo sukurta 1998 metais, kuri pakeitė vietinę sistemą „mozilla.org“ projekte. Sistemos autorė Terry



Weissman parašė „Bugzilla“ TCL kalba [9], tačiau vėliau perkėlė ją į Perl kalbą, tikėdamasi kad taip daugiau žmonių galės prisidėti prie sistemos kūrimo. Nuo to laiko nemažai projektų, tiek nemokamų tiek ir komercinių, pradėjo naudoti „Bugzilla“ savo reikmėms.

Nors „Bugzilla“ naudojama kaip klaidų/defektų šalinimo sistema, tačiau jos kodas parašytas taip, kad ją nesunku paversti į techninio palaikymo ar kitokią sistemą. Pagrindiniai „Bugzilla“ privalumai:

- Galimybė paleisti sistemą ant nemokamos programinės įrangos – „Bugzilla“ gali naudoti nemokamas duomenų bazes (MySQL, PostgreSQL ir kitos), [Apache HTTP serverį](#) ar bet kurią kitą HTTP serverį, palaikantį CGI skriptus. Sistemą galima realizuoti tiek Unix, Linux tiek ir Windows terpėse.
- Greičio ir efektyvumo užtikrinimas bet kokia kaina. „Bugzilla“ kuriama taip, kad būtų naudojama kuo mažiau užklausų į duomenų bazę, o generuojamas HTML kodas būtų kiek įmanoma mažesnis.
- Duomenų bazių užklausose naudojama ANSI SQL notacija, stengiantis nenaudoti specifinių duomenų bazių užklausų.

Sistemų galimybių ir funkcionalumų palyginimas

Šioje lentelėje pateikiamos anksčiau išrinktų programų palyginimas.

	dotProject	GForge	Trac	Bugzilla
Kalbos, kuriomis sistemos parašytos	PHP/Java Script	PHP	Python	Perl/Java Script
Palaikomos duomenų bazės	ADOdb, MySQL	PostgreSQL	SQLite, PostgreSQL, MySQL	MySQL, PostgreSQL (Nuo 3.0 versijos palaikoma ir Sybase)
SCM	Nėra	CVS	Subversion (Taip pat galima integruoti su kitomis priemonėmis)	CVS
Bendravimo priemonės	Forumai, komentarai prie užduočių/kortelių	Forumai, WiKi, komentarai prie užduočių	WiKi, komentarai prie užduočių	Komentarai prie užduočių
E-pašto siuntimas	Taip	Taip	Ne	Taip
E-pašto priėmimas	Taip	Taip	Ne	Per papildomus sistemos priedus (Nuo 3.0 versijos palaikoma be priedu)
Galimybė naudoti LDAP protokolą prisijungimui prie sistemos	Taip	Taip	Galima pajungti per	Taip

			papildomus sistemos priedus	
Unikodo palaikymas	Taip	Taip	Taip	Taip
Dinaminio dokumentavimo/generavimo integracija	Grafinis kalendorius, ganto schemos	WiKi, Ganto schemos	WiKi	Integruotos ataskaitos ir grafikai, automatizuotas ataskaitų siuntinėjimas e- paštu
Galimybė keisti darbo eigos procesą	Nėra	Nėra	Nėra	Per balsavimo sistemą, statusų eigos/reikšmių keisti negalima (prioritetai ir įtakos konfigūruojami)
Vartotojo kalendorius	Yra	Nėra	Nėra	Nėra
Galimybė pridėti naujus laukus formose	Taip	Ne	Ne	Ne (Nuo 3.0 versijos jau palaikoma)
RSS	Yra	Yra	Yra	Yra
Vartotojų ir jų grupių administravimas	Taip	Taip	Ne	Taip
Galimybė talpinti failus sistemoje	Taip	Taip	Taip	Taip

Galutinis sistemos pasirinkimas

Po atliktos sistemų analizės ir palyginimo, „Bugzilla“ buvo pasirinkta kaip sistema, labiausiai atitinkančia iškeltus reikalavimus (žiūrėti [reikalavimų](#) sekciją, 20 pusl.). Didžiausią įtaką tokiam pasirinkimui lėmė aktyvus „Bugzilla“ vystymas (tiek naujų funkcionalumų tiek ir klaidų taisymo atžvilgiu), platus papildomų priedų pasirinkimas (kas rodo didelį sistemos populiarumą), ataskaitų ir grafikų funkcionalumas, pašto siuntinėjimo galimybė.

Bugzilla diegimo ir pritaikymo projektas

Išsirinkus „Bugzilla“, kaip būsimą įmonės DOV sistemą, buvo pradėtas projektas šios sistemos diegimui ir pritaikymui įgyvendinti.

Viena iš pirmųjų projekto stadijų buvo vadinamasis „pilotas“, t.y. sistemos diegimas testavimo aplinkose ir jos įvertinimas. Atlikus jau pirmuosius sistemos įvertinimus paaiškėjo, kad dabartinio jos funkcionalumo įmonei neužteks, tad kitas projekto žingsnis buvo reikalingų funkcionalumų bei modifikacijų išskėlimas, analizė bei planavimas.

Programinės įrangos pasirinkimas

Kadangi įmonėje praktiškai naudojama vien tik Windows operacinė aplinka, tai ir „Bugzilla“ buvo nuspręsta įdiegti Windows serveryje (Windows 2003 Profesional x86). Tam kad paleisti „Bugzilla“ ant Windows terpės, reikia šios programinės įrangos:

1. MySql duomenų bazės serveris (pasirinkta 4.1.20 versija, kuri yra rekomenduojama 2.20 „Bugzilla“ versijai).
2. Apache HTTP serveris (versija 2.0).
3. Perl kalbos interpretatorius (ActivePerl 5.8).

Sistemos pakeitimai ir nauji funkcionalumai

Įvertinus visus pasiūlytus pakeitimus ir naujus funkcionalumus buvo sudarytas modifikacijų/naujų funkcionalumų sąrašas. Reikia pažymėti kad žemiau bus pateiktas galutinis sąrašas, kuris gerokai pasikeitė palyginus su pradiniu jo variantu projekto pradžioje.

Papildomi laiko apskaitos laukai

Pasirinkta „Bugzilla“ versija (2.20) standartiškai turi šiuos laiko apskaitos laukus:

- 1) „Estimated Hours“ – šiame lauke (registruojant naują užduotį) įvedamos užduočiai atlikti skirtos valandos.
- 2) „Hours Worked“ – šiame lauke rodoma suma visų valandų, kurios buvo įvestos prie užduoties komentarų ir per šį lauką, rašant naują komentarą, įvedamos naujai „nudirbtos“ valandos (tos valandos yra pririšamos prie įvedinėjamo komentaro).
- 3) „Hours Left“ – šis laukas rodo kiek liko valandų, skirtų užduočiai atlikti (šio lauko reikšmė automatiškai pakoreguojama įvedant naujai „nudirbtas“ valandas, tačiau norint reikšmę galima keisti ir rankiniu būdu).
- 4) „Deadline“ – šiame lauke įrašoma data, nurodanti kada užduotis turėtų būti atlikta.

Tačiau įmonės poreikiams patenkinti, buvo reikalingi dar du papildomi laukai:

- 1) „Start date“ – šiame lauke turėtų būti saugoma data, nurodanti kada reikėtų pradėti dirbti su pasirinkta užduotimi.
- 2) „Fixed hours per day“ – šis laukas turėtų saugoti valandas, nurodančias kiek laiko reikėtų skirti darbui per dieną su pasirinkta užduotimi.

Šie laukai reikalingi darbo planavimui supaprastinti.

Kadangi „Bugzilla“ neleidžia įvedinėti komentarų atgaline data, buvo nuspręsta pridėti dar vieną papildomą lauką – „Accountable date“. Šiame lauke, įvedinėjant naują komentarą, būtų galima nurodyti komentaro apskaitinę datą (t.y. datą, kurią reikėtų imti darbų apskaitai vietoj komentaro įvedimo datos). Šis laukas gerokai palengvins darbų apskaitos realizavimą naudojantis „Bugzilla“.

Datos įvedimo supaprastinimas

Po „pilotinio“ sistemos išbandymo paaiškėjo vienas sistemos nepatogumas – visus datos laukus tenka įrašinėti rankomis. Šis trūkumas gan nemažai pailgina vartotojų laiką, reikalingą užduotims suvedinėti bei redaguoti. Šios problemos sprendimui buvo iškeltas naujas funkcionalumas – interaktyvus kalendorius, skirtas datų laukų užpildymui. Šis kalendorius turi leisti vartotojui įvesti datas nerašinėjant jų rankomis.

Kalendoriaus funkcionalumas

Tam kad būtų galima atlikti korektišką darbų planavimą ir apskaitą, sistema turi turėti kalendorių. Deja „Bugzilla“ tokio funkcionalumo neturi, tad buvo nuspręsta šį funkcionalumą įdiegti į sistemą patiems.

Įvertinus visus likusius funkcionalumus ir apsvarsčius jų įgyvendinimo galimybę, kalendoriaus funkcionalumą buvo nuspręsta suskirstyti į dvi dalis: bendras visos sistemos kalendorius ir vartotojų asmeninis kalendorius. Sudarinėjant planus ir darant apskaitas, vartotojų asmeninis kalendorius turėtų turėti pirmenybę bendro sistemos kalendoriaus atžvilgiu. Bendro kalendoriaus paskirtis – saugoti bendrą įmonės darbo kalendorių, galiosiantį visiems vartotojams (jei konkrečios reikšmės nesikirs su vartotojų asmeniniu kalendoriumi). Taip žymiai paprasčiau ir greičiau bus suvedinėti tas pačias reikšmes, nei vesti vieną ir ta pačią reikšmę (sudarinėjant kiekvieno vartotojo kalendorių) kiekvienam vartotojui atskirai.

Bendram sistemos kalendoriui buvo išskelti tokie reikalavimai:

- 1) Turi būti galimybė nurodyti darbo ir savaitgalio dienas. Kadangi įmonė turi partnerių iš užsienio, kalendorių buvo nuspręsta daryti universalesnį, su galimybe keisti savaitės dienų paskirtį (darbo/nedarbo), taip pritaikant sistemą pavyzdžiui prie partnerių iš Izraelio (kur „savaitgaliai“ laikomi penktadienis ir šeštadienis). Šis universalumas taip pat tiktų ir įmonėm, kurios didžioji dauguma darbuotojų dirba šešias ar net septynias dienas per savaitę.
- 2) Turi būti galimybė nurodyti darbo dienų trukmę valandomis. Įmonėje darbo valandų skaičius nėra vienodas visomis darbo dienomis: penktadieniais yra dirbama valanda trumpiau. Iš čia ir kilo šis poreikis.
- 3) Turi būti galimybė įvesti papildomas nedarbo ir darbo dienas. Dažnai pasitaiko, kad kuri nors savaitės darbo diena yra šventinė, tad tai dienai netūrėtų būti planuojami jokie darbai ir vedama apskaita už ją. Lygiai taip pat ir su savaitgaliais, kurių dienos yra paskelbiamos darbo dienomis.
- 4) Kadangi jau numatytas darbo dienų trukmės ir papildomų darbo/nedarbo dienų funkcionalumas, turi būti ir galimybė nurodyti nestandartinį darbo valandų skaičių darbo dienoms. Pavyzdžiui įmonė penktadieniais ir dienomis prieš šventadienius dirba valanda trumpiau. Tad tokiu atveju darbo planai turi būti sudarinėjami atitinkamai pagal tokių dienų darbo valandų trukmę.

Vartotojų asmeniniam kalendoriui buvo išskelti šie reikalavimai:

- 1) Turi būti galimybė nurodyti darbo dienų trukmę valandomis.

- 2) Turi būti galimybė įvesti papildomas nedarbo ir darbo dienas.
- 3) Turi būti ir galimybė nurodyti nestandartinį darbo valandų skaičių darbo dienoms.

Darbų planų ataskaitos

Vienas pagrindinių iškeltų funkcionalumų, kurio „Bugzilla“ neturi – vartotojų darbų planai. Šiuos planus buvo nuspręsta daryti grafinės ataskaitos pavidalu.

Ataskaitai keliami tokie reikalavimai:

- 1) Planai turi būti sudarinėjami atsižvelgiant į laiko apskaitos laukus:
 - a) „Start date“ – jei užduotis turi užpildyta šio lauko reikšmę, tai plane ši užduotis netūrėtų būti pažymėta anksčiau nurodytos datos.
 - b) „Fixed hours per day“ – jei užduotis turi užpildyta šio lauko reikšmę, tai plane šiai užduočiai skiriamų valandų kiekis per dieną neturi viršyti nurodytos reikšmės.
- 2) Sudarinėjant planus, vartotojų užduotys turi būti rūšiuojamos pagal:
 - a) „Severity“ – plane prioritetas pirmiausia turi būti teikiamas užduotims, turinčioms didžiausią įtaką.
 - b) „Priority“ – jei užduočių įtaka vienoda, tuomet jos turi būti rikiuojamos pagal prioritetą, pirmiausiai imant aukščiausio prioriteto užduotis.
 - c) „ID“ – jei užduočių įtaka ir prioritetas vienoda, tuomet jos turi būti rikiuojamos pagal tų užduočių ID numerį, pirmiausiai imant mažiausią ID turinčią (seniausią) užduotį.
- 3) Jei pagal sudarytą planą užduoties planuojama atlikimo data viršija lauko „Deadline“ datą, tuomet plane tai turėtų matytis.

Taip pat iškeltas reikalavimas sukurti ir ataskaitą, kuri rodytų planą užduočių, pagal jas užregistravusį vartotoją. Šios ataskaitos logika turi būti identiška vartotojų darbų planų ataskaitai. Ši ataskaita yra reikalinga vadovams, registruojantiems užduotis vartotojams. Pagal šią ataskaitą jie galės matyti jų registruotų užduočių planus ir imtis vienokių ar kitokių veiksmų, jei pateikti planai jų netenkina.

„Deadline“ automatinis apskaičiavimas

Šis pakeitimas glaudžiai susijęs su vartotojų darbų planų funkcionalumu. Pakeitimo esmė – suteikti vartotojui galimybę automatiškai užpildyti lauko „Deadline“ reikšmę. Ši reikšmė turi būti apskaičiuojama pasinaudojant vartotojo darbų plano algoritmu. Šis funkcionalumas nuimtų dalį darbo nuo užduotį registruojančio vartotojo, jei jis neturi konkrečios datos užduoties atlikimui tačiau nenorėtų, kad jos atlikimas užsitęstų pernelyg ilgai.

Atliktų darbų ataskaitos

Dar vienas esminių reikalaujamų funkcionalumų, kurių „Bugzilla“ neturi (tiksliau turi tačiau ne tokių kokių reikia). Šių ataskaitų esmė – realiai atliktų darbų grafinis atvaizdavimas. Ataskaitos reikalingos dviejų tipų: vartotojų atliktų darbų ataskaita ir komponentų būsenų ir atliktų darbų ataskaita.

Vartotojų atliktų darbų ataskaita turės atrodyti identišškai vartotojų darbų planų ataskaitai, tik vietoj planuojamo darbo turėtų būti rodoma realiai atliktų darbų istorija. Šios ataskaitos reikalavimai:

- 1) Ataskaita turi būti sudarinėjama iš užduočių (nepriklausomai nuo to ar užduotis priskirta tam vartotojui ar ne) komentaruose įvestų to vartotojo valandų.
- 2) Įvestas valandas skaičiuoti datoms, nurodytos lauke „Accountable date“.
- 3) Ataskaita turi būti sudarinėjama už pasirinktą laikotarpį.

Komponentų būsenų ir atliktų darbų ataskaita būtų naudinga vertinant ne konkrečių vartotojų darbą bet stebint konkrečių produktų (projektų) stovį. Ši ataskaita taip pat labai palengvintų produktų (projektų) įvertinimą valandine (ir pinigine) išraiška. Pati ataskaita turi turėti dvi dalis:

- 1) Komponento būseną – šioje ataskaitos dalyje turėtų matytis tokia komponentų statistinė informacija kaip užduočių skaičius, tų užduočių pasiskirstymas pagal jų statusus, paskirtų, išnaudotų ir likusių valandų skaičius.
- 2) Atliktų darbų istorija – šioje dalyje turėtų matytis visos per pasirinktą laikotarpį įvestos į to komponento užduotis valandos.

Komentarų įvedimas „assignee“ vardu

Šis funkcionalumas dabar jau nebenaudojamas, tačiau jis buvo pilnai įgyvendintas projekto eigoje ir kurį laiką naudojamas, todėl ir liko neišbrauktas iš sąrašo.

Šio pakeitimo esmė – leisti vartotojams įvedinėti komentarus ne tik savo, bet ir asmens, kuriam yra paskirta pasirinkta užduotis („Assignee“), vardu. Iš pradžių buvo galvojama naudoti sistemą šiek tiek apeinant jos logiką (t.y. darbus už kai kuriuos vartotojus turėjo suvedinėti jų vadovai), dėl ko ir buvo iškeltas šis pakeitimas.

Komentarų redagavimas

Šios modifikacijos esmė – suteikti vartotojui galimybę pakoreguoti jau įvestą komentarą. Dažnai pasitaiko jog vartotojai įveda komentarus su klaidom (įvesdami ne tą „Accountable date“ reikšmę ar įvesdami komentarą ne prie tos užduoties), tad šis funkcionalumas leistų jiems ištaisyti savo klaidas. Komentarų redagavimas turi leisti pakeisti šių laukų reikšmes:

- 1) Komentaro autorius.

- 2) „Worked hours“.
- 3) Komentaro tekstas.
- 4) „Accountable date“.

Komponentų funkcionalumo papildymas

Šios modifikacijos buvo iškeltos darbo su sistema palengvinimui.

Kuriant naujas užduotis pasirinkus reikiamą produktą, reikia išsirinkti norimą komponentą iš sąrašo. Jei produktas yra aktyvus ilgą laiką, komponentų sąrašas pildomas vis naujomis reikšmėmis. Praktiškai komponentai yra naudojami tik kurį laiką ir vėliau tampa nebeaktualūs, tačiau sąraše jie yra pateikiami visi. Todėl buvo pasiūlyta įvesti komponento savybę, nurodančia ar naudoti komponentą naujoms užduotims („Closed for bug entry“, analogiškas laukui produktuose). Jei šis laukas yra pažymimas, tuomet, sudarinėjant komponentų sąrašą, atitinkamas komponentas netūrėtų būti įtraukiamas į jį.

Analogiškai ir su vartotojų sąrašu (jei sistema sukonfigūruota atitinkamai). Tam, kad kuriant naują užduotį, nereikėtų naršyti po visų sistemos vartotojų sąrašą, norint rasti tinkamą vartotoją, kuriam paskirti užduotį, buvo pasiūlyta įvesti komponentų resursų sąrašą, iš kurio ir būtų formuojamas vartotojų sąrašas.

Veiksmų planas

Kaip jau buvo minėta anksčiau, visi patvirtinti pakeitimai nebuvo iškelti vienu, jų sąrašas buvo nuolat koreguojamas beveik visoje projekto eigoje.

Čia pateikiamas planas, sudarytas iš visų realiai patvirtintų pakeitimų. Buvo siūlyta ir daugiau pakeitimų, tačiau jie nebuvo patvirtinti, tad jie liko neįtraukti į planus.

1. Darbų planų ataskaitos.
2. Papildomi laiko apskaitos laukai: „Start date“.
3. Kalendoriaus funkcionalumas.
4. „Deadline“ automatinis apskaičiavimas.
5. Atliktų darbų ataskaitos: vartotojų atliktų darbų ataskaita.
6. Datos įvedimo supaprastinimas.
7. Papildomi laiko apskaitos laukai: „Fixed hours per day“.
8. Atliktų darbų ataskaitos: komponentų būsenų ir atliktų darbų ataskaita.
9. Komentarų įvedimas „assignee“ vardu.
10. Komentarų redagavimas.
11. Papildomi laiko apskaitos laukai: „Accountable date“.
12. Komponentų funkcionalumo papildymas.

Darbų eigos aprašymas

Darbų eigos grafas

1. 2005.07.04 – 2005.08.08; Sistemų kandidačių paieška, analizė ir pasirinkimas.
2. 2005.08.22 – 2005.08.29; „Bugzilla“ pilotinis diegimas, analizė.
3. 2005.09.19 – 2005.09.23; „Bugzilla“ diegimas ir paruošimas gamybinėje aplinkoje.
4. 2005.09.26 – 2005.09.27; Termino „bug“ pakeitimas terminu „task“.
5. 2005.09.28 – 2008.10.14; Darbų planų ataskaitos.
6. 2005.10.17 – 2007.10.26; „Start date“ lauko įvedimas, ataskaitų koregavimas.
7. 2005.10.31 – 2005.11.11; Kalendoriaus funkcionalumas.
8. 2005.11.14 – 2005.11.17; „Deadline“ automatinis apskaičiavimas.
9. 2005.11.18 – 2005.11.30; Vartotojų atliktų darbų ataskaita.
10. 2005.12.01 – 2005.12.05; Kalendoriaus funkcionalumo pakoregavimas.
11. 2005.12.06 – 2005.12.09; Datos įvedimo supaprastinimas.
12. 2005.12.12 – 2005.12.30; „Fixed hours per day“ lauko įvedimas, ataskaitų koregavimas.
13. 2006.01.09 – 2006.01.20; Komponentų būsenų ir atliktų darbų ataskaita.
14. 2006.02.06 – 2006.02.16; Komentarų įvedimas „assignee“ vardu.
15. 2006.03.06 – 2006.03.22; Komentarų redagavimas.
16. 2006.04.10 – 2006.04.28; „Accountable date“ lauko įvedimas, ataskaitų koregavimas.
17. 2006.06.05 – 2006.06.07; Kalendoriaus funkcionalumo koregavimas.
18. 2007.03.05 – 2007.03.21; Komponentų funkcionalumo papildymas.

Problemos ir jų sprendimai

Įgyvendinant „Bugzilla“ diegimo ir pritaikymo projektą, buvo susidurta su keletą problemų, kurias beveik visas vienokiu ar kitokiu būdu pavyko išspręsti.

„Bugzilla“ ir lietuviškumas

Įdiegus sistemos „pilotinę“ versiją iš karto buvo susidurta su lietuviškumo problema. Analizuojant šią problemą buvo iškeltos šios galimos problemos priežastys:

- 1) Netinkami serverio, ant kurio paleista „Bugzilla“, regioniniai nustatymai.
- 2) MySQL duomenų bazės netinkamas sukonfigūravimas.
- 3) HTTP serverio netinkama konfigūracija.
- 4) Kliento naršyklės (Internet Explorer ar kit.) konfigūracijos problemos.

Galiausiai, patikrinus visas aukščiau minėtas galimas priežastis, lietuviškumo problema buvo išspręsta sekančiais žingsniais:

- Nurodyta HTTP serveriui tinkama puslapių koduotė (windows-1257).
- Pakoreguotas „Bugzilla“ antraštinis ruošinys, reikiamoje HTML sekcijoje nurodant tinkamą puslapių koduotę.

Prisegamų failų limitas

Jau perdėjus naudoti „Bugzilla“ iškilo problema su prisegamais failais – bandant prisegti didesnius nei 850 kilobaitai failus sistema parodė klaidos pranešimą. Išanalizavus problemą paaiškėjo, jog tai MySQL problema – MySQL duomenų bazė (bent jau 4.0 versija) negali saugoti didesnių nei 850 kilobaitai BLOB [\[10\]](#) objektų.

Išanalizavus MySQL dokumentacijas ir pasiūlymus internete, buvo nuspręsta pasinaudoti „Bugzilla“ galimybe saugoti prisegamus failus serverio failų sistemoje.

Ruošinių sistema

„Bugzilla“ kodas parašytas taip, kad HTML kodo generavimas vyksta per vidinę ruošinių (ang. Template) sistemą. Tad norint atlikti bet kokias sistemos modifikacijas, kodo pakeitimus reikia atlikti ne tik sistemos CGI skriptuose bet ir tuose ruošiniuose. Problema iškilo tame, kad tų ruošinių yra labai daug ir jie tarpusavyje tiesiogiai susiję. Pavyzdžiui norint įvesti vienoje ar kitoje formoje naują lauką, reikia susirasti tos formos pagrindinį ruošinį, atlikti ten pakeitimus, ir aiškintis ar tas ruošinys nekviečia kito ruošinio, kuris savo ruoštu gali kviesti dar vieną. Tuomet dar reikia redaguoti klaidų apdorojimo ir

pranešimų ruošinį, įrašant leistinas lauko reikšmes ir formatus ir pridėdant naują klaidos įrašą (nurodant klaidos kodą ar pranešimo tekstą).

Didžiausią tokių sistemos modifikacijų, kaip esamų formų redagavimas, darbo laiko dalį užėmė būtent šių ruošinių paieška ir analizė.

„Bugzilla“ klaidos

Pradėjus naudotis sistema, buvo atrastos dvi pačios „Bugzilla“ klaidos:

- 1) Jei vartotojai produktų ar komponentų pavadinimuose įveda taškus arba kablelius, tuomet sistema pradeda veikti nebe korektiškai (pradedami rodyti klaidos pranešimai redaguojant užduotis, užduočių/komponentų sąrašai užpildomi ne iki galo ir t.t.). Ši problema buvo išspręsta pataisius atitinkamų CGI skriptų kodus, automatiškai pašalinant taškus ir kablelius iš produktų ir komponentų pavadinimų.
- 2) Bandant rūšiuoti paieškos rezultatus pagal lauką „Deadline“, sistema gražina neapdorotą klaidos pranešimą. Išanalizavimus problemos esmę ir įvertinus laiko sąnaudas, reikalingas problemai pašalinta, buvo nuspręsta šios klaidos netaisyti o tiesiog informuoti sistemos vartotojus apie ją.

Galutinis projekto statusas

Projektui pasiekus pabaigą, buvo įgyvendinti visi projekto metu patvirtinti pakeitimai ir nauji funkcionalumai. Naujosios modifikacijos praėjo testavimo stadijas ir, su nedideliais pakeitimais (atsiradusiais dėl pastabų/klaidų testavimo metu), įdiegtos į darbinę aplinką.

Papildomi laiko apskaitos laukai

Visi reikalaujami laukai buvo įvesti į „Bugzilla“ sistemą. Tačiau neužteko šių laukų įvesti tik vienoje (užduoties redagavimo) formoje. Formos, kuriose teko įvesti naujus laukus:

1) Užduoties redagavimas:

Orig. Est.	Current Est.	Hours Worked	Hours Left	%Complete	Gain	Start date	Deadline	Fixed hours per day
80.0	80.0	3.0 + 0	77.0	3	0.0	2007-05-11	<input type="text"/> <small>Offered deadline date</small>	0.0

[Summarize time \(including time for tasks blocking this task\)](#)

Additional Comments:

Add comment in the name of assignee **Accountable date:**

----- Comment #1 From [Aurimas Strikulis](#) 2007-04-28 00:50 [reply] -----
Additional hours worked: 1.0 Accountable date: 2007-04-28

Testines valandos.

----- Comment #2 From [Aurimas Strikulis](#) 2007-04-28 00:53 [reply] -----
Additional hours worked: 2.0 Accountable date: 2007-04-29

Testines valandos (2).

2) Naujos užduoties kūrimas:

Estimated Hours:	<input type="text" value="0.0"/>
Start date:	<input type="text"/> (YYYY-MM-DD)
Deadline:	<input type="text"/> (YYYY-MM-DD)
URL:	<input type="text" value="http://"/>
Summary:	<input type="text"/>

3) Kelių užduočių redagavimas vienu metu:

Estimated Hours: <input type="text" value="-do_not"/>	Remaining Hours: <input type="text" value="-do_not"/>	Worked Hours: <input type="text" value="-do_not"/>
QA Contact: <input type="text" value="-do_not_change-"/>		
CC List: <input type="text"/>	Add these to the CC List <input type="button" value="v"/>	
Additional Comments:		
<div style="border: 1px solid gray; height: 40px;"></div>		
<input type="checkbox"/> Add comment in the name of assignee	Accountable date: <input type="text"/> <input type="button" value="calendar"/>	

Datos įvedimo supaprastinimas

Šio pakeitimo įgyvendinimui buvo pasinaudota kompanijos „Zapatec.com“ nemokamu produktu „Calendar“ [<http://www.zapatec.com/>]. Tam kad įterpti šį interaktyvų kalendorių į reikiamas „Bugzilla“ formas, buvo modifikuoti atitinkamų formų šablonai bei antraštinis šablonas, ištraukiant reikiamus Java skriptus.

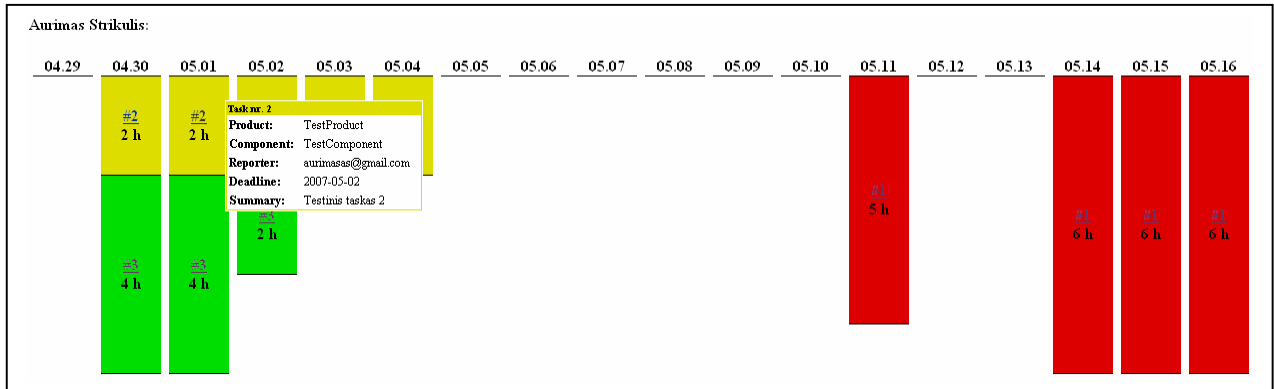
Kalendoriaus funkcionalumas

Kaip ir buvo numatyta projekto reikalavimuose, „Bugzilla“ sistemoje įgyvendintas kalendoriaus funkcionalumas tiek vartotojams tiek ir visai sistemai. Papildomai buvo įvestas dar vienas laukas „Unexpected days“, kuriame saugomas valandų skaičius nenumatytoms datoms:

	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday	Unexpected days
Weekends:	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Default workload per day (h):	<input type="text" value="8.0"/>	<input type="text" value="8.0"/>	<input type="text" value="8.0"/>	<input type="text" value="8.0"/>	<input type="text" value="8.0"/>	<input type="text" value="0.0"/>	<input type="text" value="0.0"/>	<input type="text" value="8.00"/>
	<input type="button" value="Update"/>							
Workload exceptions:	<div style="border: 1px solid gray; padding: 5px;"> <div style="display: flex; justify-content: space-between; align-items: center;"> <div style="border: 1px solid gray; width: 40px; height: 40px;"></div> <div><input type="button" value="Remove dates"/></div> </div> <div style="display: flex; justify-content: space-between; align-items: center; margin-top: 5px;"> <input type="text"/> <div><input type="button" value="Add new dates"/></div> </div> </div>							
Additional free days:	<div style="border: 1px solid gray; padding: 5px;"> <div style="display: flex; justify-content: space-between; align-items: center;"> <div style="border: 1px solid gray; width: 40px; height: 40px;"></div> <div><input type="button" value="Remove dates"/></div> </div> <div style="display: flex; justify-content: space-between; align-items: center; margin-top: 5px;"> <input type="text"/> <div><input type="button" value="Add new dates"/></div> </div> </div>							
Additional working days:	<div style="border: 1px solid gray; padding: 5px;"> <div style="display: flex; justify-content: space-between; align-items: center;"> <div style="border: 1px solid gray; width: 40px; height: 40px;"></div> <div><input type="button" value="Remove dates"/></div> </div> <div style="display: flex; justify-content: space-between; align-items: center; margin-top: 5px;"> <input type="text"/> <div><input type="button" value="Add new dates"/></div> </div> </div>							

Darbu planų ataskaitos

Remiantis šiuo reikalavimu buvo sukurtos dvi ataskaitos: „Work plan“ ir „Reporters task plan“. Abiejose ataskaitose užduotys suskirstytos spalvų grupėmis pagal jų įtaką, be to „Work plan“ ataskaitoje užduotys atvaizduojamos skirtingo aukščio pagal joms skiriamas valandas atitinkamoms dienoms. Taip pat papildomai realizuota galimybė pamatyti pagrindinę užduoties informaciją užvažiavus ant tos užduoties pele:



„Deadline“ automatinis apskaičiavimas

Šis funkcionalumas pasiekiamas per užduočių redagavimo formą. Po „Deadline“ lauku pridėta nuoroda, kurią paspaudus vartotojas nukreipiamas į šio lauko automatinio parinkimo formą. Šioje formoje pridėtas dar vienas papildomas funkcionalumas – perspėjimas apie užduotis, kurių planuojama darbo pabaiga nusikėlė po tų užduočių „Deadline“ datos dėl pasirinktos užduoties įterpimo:

Offered deadline date for task 3

Current deadline date:

Suggested deadline date:

The insertion of task #3 moved following tasks scheduled end date beyond deadline:

#2 **Deadline:** **Scheduled end date:**

Atliktų darbų ataskaitos

Taip kaip ir buvo nurodyta projekto reikalavimuose, buvo sukurtos dvi atliktų darbų ataskaitos: „Users actual work“ ir „Components state and actual work“.

Pirmoji ataskaita grafiškai niekuo nesiskiria nuo vartotojų darbų plano ataskaitos („Work plan“).

Antroji ataskaita buvo suformuota visiškai kitokia logika (to reikalavo ir iškeltas funkcionalumas). Ši ataskaita susideda iš dviejų dalių: „Component statistics“ ir „Actual work on component“:

TestProduct: TestComponent

Component statistics

Component task count:		Orig. Est.	Hours worked (per period)	Hours worked (total)	Hours left
Open: <input type="text" value="3"/>	Unconfirmed: <input type="text" value="0"/>	0	0	0	0
	New: <input type="text" value="0"/>	0	0	0	0
	Assigned: <input type="text" value="3"/>	100	14	14	89
	Reopened: <input type="text" value="0"/>	0	0	0	0
Resolved: <input type="text" value="0"/>	Fixed: <input type="text" value="0"/>	0	0	0	
	Invalid: <input type="text" value="0"/>	0	0	0	
	Wontfix: <input type="text" value="0"/>	0	0	0	
	Later: <input type="text" value="0"/>	0	0	0	
	Remind: <input type="text" value="0"/>	0	0	0	
	Duplicate: <input type="text" value="0"/>	0	0	0	
	Workforme: <input type="text" value="0"/>	0	0	0	
	Moved: <input type="text" value="0"/>	0	0	0	
Totals:		100	14	14	89

Actual work on component

	04.15	04.16	04.17	04.18	04.19	04.20	04.21	04.22	04.23	04.24	04.25	04.26	04.27	04.28	04.29	
Aurimas Strikulis														1	2	3 h
#1 Test Administrator													3			6 h
#3 Aurimas Strikulis									8							3 h
																8 h

Abi ataskaitos, taip kaip ir darbų planų ataskaitos, turi realizuotą funkcionalumą, leidžiantį pamatyti pagrindinę užduočių informacija užvažiavus ant jų pele.

Komentarų įvedimas „assignee“ vardu

Kaip jau buvo minėta, šis funkcionalumas dabar nebenaudojamas, tačiau visiškai įgyvendintas ir funkcionuojantis. Užduoties redagavimo ir kelių užduočių redagavimo formose buvo įdėta papildoma žymė „Add comment in the name of assignee“ (žiūrėti iliustracijas prie [papildomu laiko apskaitos lauku](#)), kurią pažymėjus komentaras įvedamas „assignee“ vardu. Taip pat dėl šio funkcionalumo buvo šiek tiek pakoreguotos atliktų darbų ataskaitos, kuriose buvo pridėta galimybė, skaičiuojant užduočių valandas, priskirti jas atitinkamų užduočių „assignee“ vartotojams, nepriklausomai nuo to kas tas valandas įvedinėjo.

Komentarų redagavimas

Kadangi šis funkcionalumas yra gan pavojingas vartotojo sąžiningumo atžvilgiu (jiems suteikiama galimybė redaguoti praktiškai bet kokią komentaro informaciją), jo (funkcionalumo) pasiekimas buvo apribotas specialiai tam sukurta grupe „editcomments“ (nustatoma per „Bugzilla“ parametrus). Jei vartotojas priklauso nurodytai grupei, tuomet jis gali pasiekti komentaro redagavimo formą:

Edit comment of task #1

Commentar owner: test_admin@test.com
Worked hours: 3.00
Comment text: Testines valandos (3).

Accountable date of event: 2007-04-27
Update changes: Update

Last updated by Aurimas Strikulis on 2007-04-29 19:59:36

Komponentų funkcionalumo papildymas

Komponentų redagavimo forma buvo papildyta nauju lauku, kuris leidžia pažymėti komponentą kaip uždaryta naujų užduočių kūrimui, bei laukais skirtais valdyti resursų sąrašą:

Edit Component 'TestComponent'

Component: TestComponent
Component Description: This is a test component in the test product database. This ought to be blown away and replaced with real stuff in a finished installation of Bugzilla.

 Closed for bug entry.
Default Assignee: Aurimas Strikulis <aurimasas@gmail.com>
Default QA contact:

Resources:
Delete selected resources:
Add new resources:
Aurimas Strikulis <aurimasas@gmail.com>
Test Administrator <test_admin@test.com>
Add selected resources:

Tasks: 3
Update or Delete this component.

Taip pat buvo pakoreguota naujos užduoties forma, pridedant šalia jau buvusio visų vartotojų sąrašo naują sąrašą, sudaromą iš pasirinkto komponento resursų:

Initial State: NEW

Assign To: (List of component resources)
 (List of all resources)

QA Contact:

Cc:

Darbo rezultatų analizė

Šio darbo eigoje buvo pasirinktos ir išanalizuotos keturios skirtingos sistemos, palygintos šių sistemų galimybės, privalumai ir trūkumai. Smulkiau išnagrinėta „Bugzilla“, jos struktūra, kodas.

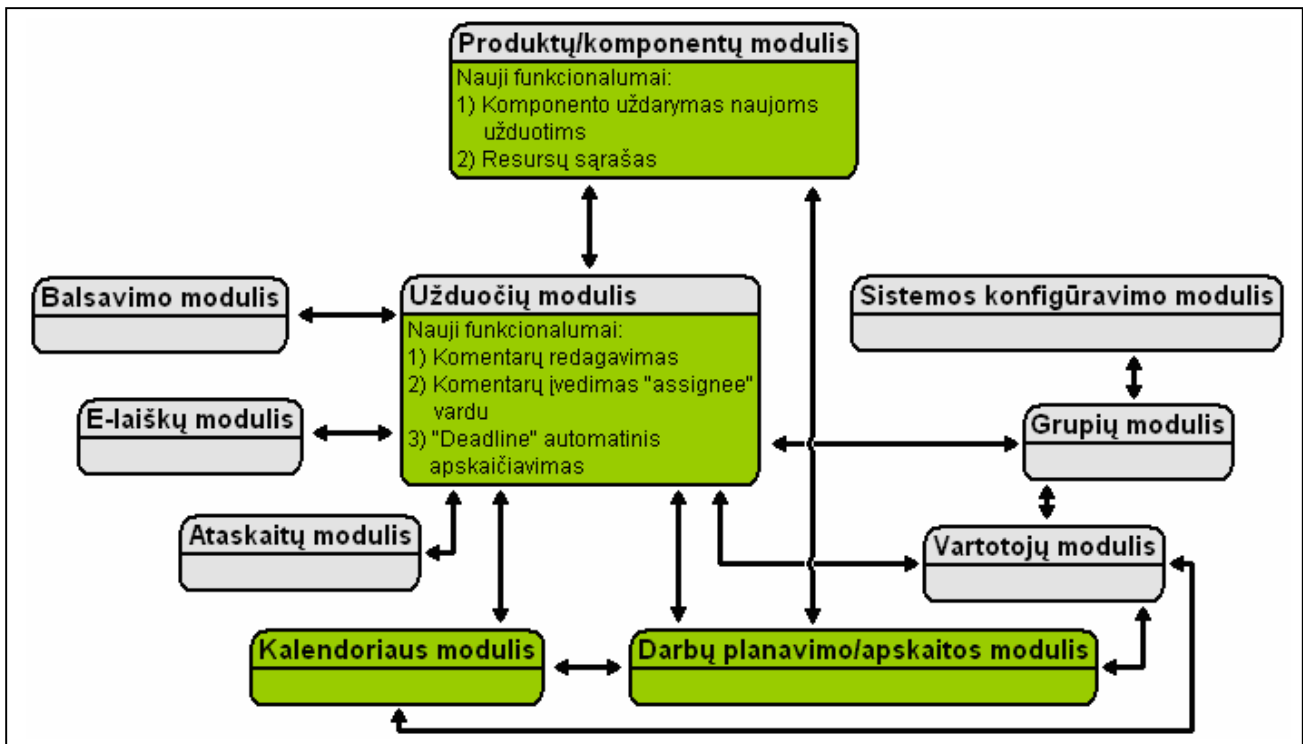
Sistemos modifikacijos eigoje teko įsisavinti Perl programavimo kalbos sintaksę, jos įvairius panaudojimo niuansus CGI skriptuose. Taip pat teko panaudoti turimas SQL kalbos žinias modifikuojant „Bugzilla“ duomenų bazę, pritaikant ją naujiems funkcionalumams. Dar teko įsisavinti HTML šablonų struktūrą, perprasti jų pagrindinius sintaksės principus.

Buvo įgyvendinti visi „Bugzilla“ diegimo ir pritaikymo projekto eigoje patvirtinti pakeitimai ir nauji funkcionalumai. Pakeitimų eigoje buvo sukurti 7 nauji ir modifikuoti 15 skriptai, bei modifikuoti 26 šablonai.

Pasibaigus projektui, „Bugzilla“ įgijo visiškai naujus ir jai unikalius funkcionalumus, kuriuos būtų galima atskirti į kelis atskirus modulius: kalendorius ir planuojamų/atliktų darbų ataskaitos. Šie moduliai, papildydami „Bugzilla“ unikaliomis savybėmis, buvo pilnai integruoti su likusiu sistemos funkcionalumu. Kartu su kitais smulkesniais pakeitimas, visos šios sistemos modifikacijos pakeitė „Bugzilla“ iš klaidų registravimo ir šalinimo sistemos į pilnavertę DOV sistemą, tenkinančią jai keliamus reikalavimus.

„Bugzilla“ funkcinė schema

Galutinė sistemos funkcinė schema:



Po sistemos modifikacijų, „Bugzilla“ įgijo du naujus funkcinis modulius:

1. Kalendorius.

Šis modulis realizuoja sistemos ir vartotojų kalendoriaus funkcionalumą ir jis yra tiesiogiai susijęs su užduočių, darbų planavimo/apskaitos ir vartotojų moduliais

2. Darbų planavimas/apskaita.

Šis modulis įgalino darbų planavimo ir apskaitos funkcionalumus. Jis yra tiesiogiai surištas su užduočių, kalendoriaus ir vartotojų moduliais.

Taip pat modifikacijas patyrė ir šie „Bugzilla“ moduliai:

1. Produktai/komponentai.

Šis modulis įgavo naujų funkcionalumų: komponento uždarymas naujoms užduotims ir resursų sąrašas.

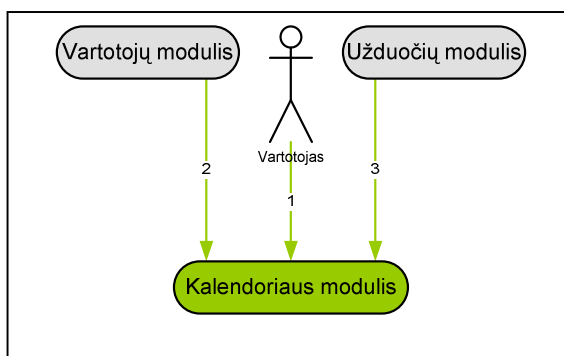
2. Užduotys.

Šis modulis pasipildė naujais funkcionalumais: komentarų redagavimu, komentarų įvedimu „assignee“ vardu ir „Deadline“ automatinio paskaičiavimo.

Naujieji sistemos moduliai yra glaudžiai susiję su kitais sistemos moduliais.

Kalendoriaus modulis yra priklausomas nuo šių modulių ir vartotojo teikiamos informacijos:

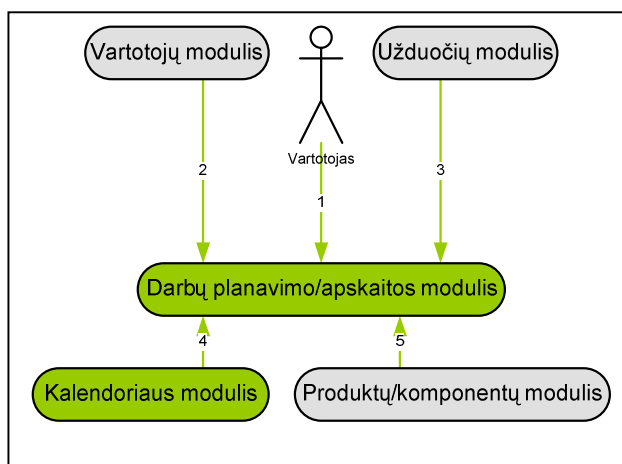
1. Vartotojas – vartotojas įveda datas, valandas ir savaitės dienų nustatymus, kuriuos modulis turi apdoroti ir išsaugoti sistemoje.
2. Vartotojų modulis – iš šio modulyje yra traukiamas sistemos vartotojų sąrašas. Vartotojų redagavimo atveju abu moduliai



- veikia paraleliai ir vartotojui abiejų modulių informacija/funkcionalumas pasiekiamas per tą pačią formą (editusers.cgi).
3. Užduočių modulis – šis modulis naudojamas apskaičiuojant „Deadline“. Informacija, kuri naudojama iš šio modulyje – komentarų datos ir suvestos valandos.

Darbų planavimo/apskaitos moduliui reikalinga informacija iš:

1. Vartotojas – jis nurodo žmones, kurių planus/ataskaitas rodyti bei atliktų darbų ataskaitos atveju nurodo ataskaitos datos intervalą.
2. Vartotojų modulis – iš šio modulyje traukiamas sistemos vartotojų sąrašas.
3. Užduočių modulis – šiame modulyje patalpinta visa su užduotimis susijusi



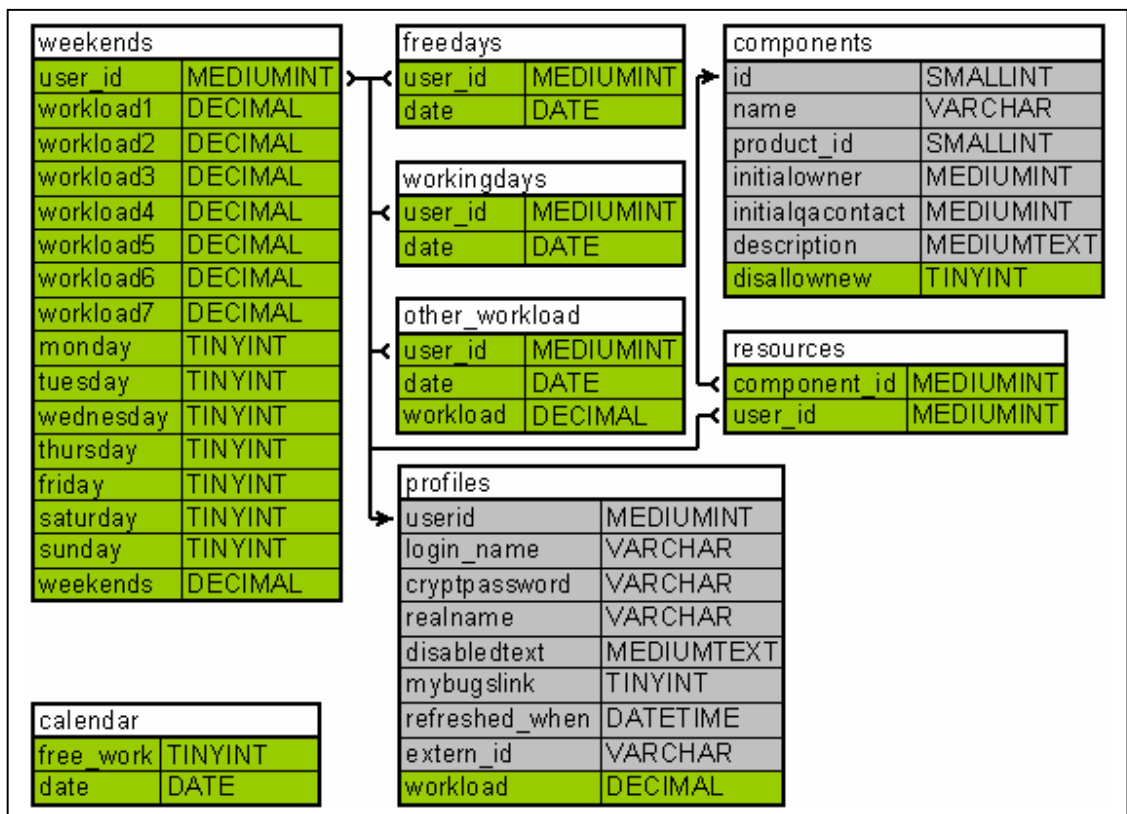
- reikalinga informacija – komentarų datos su suvestomis valandomis, užduočių pradžios ir pabaigos datos, užduotims skirtų bei išnaudotų valandų skaičiumi.
4. Kalendoriaus modulis – šis modulis (kaip ir užduočių modulis) yra vienas pagrindinių informacijos šaltinių sudarinėjant vartotojų užduočių planus. Planams ir atlikto darbo ataskaitoms generuoti yra naudojama visa kalendoriaus modulyje informacija: tiek bendro sistemos tiek ir vartotojų individualių darbo savaitės nustatymai bei papildomos darbo/nedarbo datos.
 5. Produktų/komponentų modulis – iš šio modulyje formuojamas komponentų sąrašas komponentų ataskaitoms generuoti.

„Bugzilla“ duomenų bazės pakeitimai

Tam kad būtų galima įgyvendinti visus iškeltus funkcionalumus teko pakoreguoti sistemos duomenų bazę pridėdant naujus laukus ir lenteles.

Naujai sukurtos lentelės:

1. weekends – šioje lentelėje talpinami savaitės dienų nustatymai.
2. freedays – šioje lentelėje saugomos vartotojų papildomos nedarbo dienos.
3. workingdays – ši lentelė skirta saugoti vartotojų papildomoms darbo dienoms.
4. other_workloads – šioje lentelėje saugomos nestandartinės darbo dienų trukmės.
5. calendar – lentelė skirta sistemos kalendoriaus papildomoms darbo ir nedarbo dienoms saugoti.
6. resources – šioje lentelėje saugomi komponentų resursų sąrašai.



Lentelės *weekends*, *freedays*, *workingdays*, *other_workload* ir *resources* yra surištos su lentele *profiles* per lauką „user_id“. Lentelė *resources* taip pat yra surišta su lentele *components* per lauką „component_id“. Šie surišimai nėra realizuoti duomenų bazės lygyje (nėra naudojami išoriniai raktai ar RI taisyklės [11]), jie realizuoti tik sistemos kodo logikoje.

Modifikuotos lentelės:

1. profiles – šioje lentelėje saugomi sistemos vartotojai. Šiai lentelei buvo pridėtas naujas laukas „workload“, saugantis vartotojo standartinę darbo dienos trukmę.
2. components – šioje lentelėje talpinama visa komponentų informacija. Šioje lentelėje buvo pridėtas naujas laukas „disssallownew“, nurodantis ar komponentas nėra uždarytas naujoms užduotims.
3. bugs – užduočių duomenų lentelė. Ši

bugs		longdescs	
bug_id	MEDIUMINT	bug_id	MEDIUMINT
assigned_to	MEDIUMINT	who	MEDIUMINT
bug_file_loc	TEXT	bug_when	DATETIME
bug_severity	VARCHAR	work_time	DECIMAL
bug_status	VARCHAR	the_text	MEDIUMTEXT
creation_ts	DATETIME	isprivate	TINYINT
delta_ts	DATETIME	already_wrapped	DECIMAL
short_desc	MEDIUMTEXT	accountable_date	DATE
op_sys	VARCHAR	last_who	MEDIUMINT
priority	VARCHAR	last_when	DATETIME
product_id	SMALLINT		
rep_platform	VARCHAR		
reporter	MEDIUMINT		
version	VARCHAR		
component_id	SMALLINT		
resolution	VARCHAR		
target_milestone	VARCHAR		
qa_contact	MEDIUMINT		
status_whiteboard	MEDIUMTEXT		
votes	MEDIUMINT		
keywords	MEDIUMTEXT		
lastdiffed	DATETIME		
everconfirmed	TINYINT		
reporter_accessible	TINYINT		
cclist_accessible	TINYINT		
estimated_time	DECIMAL		
remaining_time	DECIMAL		
start_date	DATETIME		
deadline	DATETIME		
fixed_unit	DECIMAL		
alias	VARCHAR		

lentelė papildyta dviem naujais laukais: „start_date“, saugančiu užduoties formos lauko „Start date“ reikšmę, ir „fixed_unit“, saugančiu užduoties formos lauko „Fixed hours per day“ reikšmę.

4. longdescs – lentelė, sauganti užduočių komentarus. Ši lentelė papildyta trim naujais laukais: „accountable_date“, saugantis užduoties formos lauko „Accountable date“ reikšmę, „last_who“, saugantis paskutinio asmens, redagavusio komentarą, ID ir „last_when“, nurodantis paskutinę komentaro redagavimo datą.

Skriptų ir ruošinių modifikacijos

Baigus visas sistemos modifikacijas buvo sukurti šie nauji CGI skriptai:

1. actualwork.cgi – šiame skripte realizuota vartotojų atlikto darbo ataskaita.
2. component_report.cgi – šiame skripte realizuota komponentų būsenų ir atliktų darbų ataskaita.
3. deadline.cgi – šis skriptas atsakingas už ataskaitos „Deadline“ lauko automatinį apskaičiavimą.
4. editcalendar.cgi – šio skripto pagalba realizuota bendro sistemos kalendoriaus forma.
5. editcomments.cgi – skriptas, kuriame realizuotas komentarų redagavimo funkcionalumas.
6. reportersplan.cgi – skriptas, kuriuo formuojama registruojančių vartotojų (darbo grupių vadovų) ataskaita.
7. workplan.cgi – šiame skripte realizuotas vartotojų darbo plano funkcionalumas.

Modifikacijas patyrė šie skriptai:

1. bug_list.cgi – skriptas, atsakingas už užduočių sąrašų formavimą. Pridėtas papildomų laiko apskaitos laukų atvaizdavimas.
2. colchange.cgi – skriptas, kuriuo keičiamas atvaizduojamų stulpelių sąrašas užduočių sąrašuose. Pridėtas papildomų laiko apskaitos laukų palaikymas.
3. createaccount.cgi – skriptas, kuriantis naujus sistemos vartotojus. Papildyta duomenų bazės lentelių duomenų apsauga (įtraukiant kalendoriaus lentelę).
4. editcomponents.cgi – komponentų redagavimo skriptas. Papildytas naujai iškeltais funkcionalumais (užrakinimo naujoms užduotims žyme ir resursų sąrašu) bei problemos su taškais ir kableliais pavadinime sprendimu.
5. editproducts.cgi – produktų redagavimo skriptas. Papildytas problemos su taškais ir kableliais pavadinime sprendimu.
6. editusers.cgi – vartotojų redagavimo skriptas. Papildytas kalendoriaus funkcionalumu.
7. editversions.cgi – produktų versijų redagavimo skriptas. Papildytas problemos su taškais ir kableliais pavadinime sprendimu.
8. enter_bug.cgi – naujų užduočių kūrimo skriptas. Papildytas resursų sąrašo funkcionalumu ir laiko apskaitos laukais.
9. post_bug.cgi – komentarų įvedimo skriptas. Papildytas komentarų įvedimo „assignee“ vardu funkcionalumu ir laiko apskaitos laukais.

10. process_bug.cgi – užduočių redagavimo skriptas. Papildytas naujais funkcionalumais (papildomi laiko apskaitos laukai, komentarų redagavimas).
11. Schema.pm – duomenų bazės schemos skriptas. Papildytas naujais laukais.
12. Bug.pm – skriptas, kuriame aprašyti pagalbiniai metodai skriptam su užduotimis. Pataisytas pritaikant naujiems funkcionalumams.
13. Config.pm – skriptas, kuriame aprašyti įvairūs sistemos parametrai. Pataisytas sistemos versijos parametras.
14. DB.pm – skriptas, kuriame aprašyti pagalbiniai metodai darbui su duomenų baze. Pataisytas laukų sąrašas (pritaikant naujiems apskaitos laukams).
15. User.pm – skriptas, kuriame aprašyti pagalbiniai metodai reikalingi vartotojų redagavimui. Pritaikytas vartotojų kalendoriaus palaikymui.

Taip pat modifikacijas patyrė šie ruošiniai:

- Komponentų redagavimo (`\template\en\default\admin\components\`):
edit.html.tmpl, list.html.tmpl, updated.html.tmpl.
- Vartotojų redagavimo (`\template\en\default\admin\users\`):
edit.html.tmpl, userdata.html.tmpl.
- Užduočių redagavimo (`\template\en\default\bug\`):
create.html.tmpl, comments.html.tmpl, edit.html.tmpl, show.html.tmpl.
- Globalūs (`\template\en\default\global\`):
field-descs.none.tmpl, header.none.tmpl, messages.none.tmpl, useful-links.none.tmpl, userd-error.none.tmpl, userselect.none.tmpl, variables.none.tmpl.
- Sąrašų formavimo (`\template\en\default\list\`):
change-columns.html.tmpl, edit-multiple.html.tmpl, list.html.tmpl, table.html.tmpl.
- Ataskaitų sąrašo (`\template\en\default\reports\`):
menu.html.tmpl.
- Paieškos (`\template\en\default\search\`):
form.html.tmpl, search-advanced.html.tmpl, search-report-graph.html.tmpl, report-table.html.tmpl.
- Laiškų siuntimo (`\template\en\default\whine\`):
mail.html.tmpl.

Išvados

Kaip parodė praktika, didesniuose kolektyvuose dirbti be darbo organizavimo valdymo (DOV) sistemų yra labai nepatogu ir neefektyvu. Be šių sistemų labai sudėtinga darbo grupių vadovams skirstyti darbus, nes jie negali tiksliai numatyti kiekvieno žmogaus užimtumo, jų planų, taip pat jie (vadovai) negali tiksliai atsakyti, koks vieno ar kito darbo (projekto) stovis, kokie darbai užsitęsė ir viršijo numatytus planus. DOV sistemos visas šias problemas pašalina, padėdamos automatizuoti darbų planavimą bei apskaitą.

Šio darbo eigoje buvo apžvelgta kas tai yra DOV sistemos, kam jos naudojamos, kokie jų uždaviniai.

Praktinis atlikto darbo rezultatus – sistemos „Bugzilla“ įdiegimas ir modifikavimas pritaikant ją įmonės UAB „Baltic Data Center“ reikmėms. Šis projektas buvo baigtas be didesnių problemų ir šiuo metu modifikuota sistema sėkmingai naudojasi daugiau nei 40 įmonės darbuotojų.

Rodyklė

[1] Henry Laurence Gantt – mechanikos inžinierius ir vadybos konsultantas (1861m. – 1919m.), labiausiai išgarsėjęs savo sukurta Ganto schema.

[2] Kritinis kelias (ang. Critical path) – tai matematinis algoritmas skirtas planuoti projekto veiklas. Šis algoritmas buvo sukurtas 1950 metais bendromis korporacijų DuPont ir Remington Rand pajėgomis elektrinių priežiūroms projektams.

[3] Kritinė grandinė (ang. Critical chain) – metodas (pirmą kartą publikuotas 1997 metais knygoje „Kritinė grandinė“), pagrįstas matematiniais algoritmais, skirtas projektų valdymo optimizavimui. Metodo autorius – Eliyahu M. Goldratt, fizikas, gimęs Izraelyje 1948 metais.

[4] Realus laiko susirašinėjimas (ang. Instant Messaging) – bendravimo forma, kuomet vartotojai, kompiuterinių tinklų pagalba, bendrauja „gyvu“ laiku siuntinėdami vienas kitam tekstines žinutes.

[5] Multiplatforminės aplikacijos – aplikacijos (programos), kurios gali veikti skirtingose operacinėse sistemose (pavyzdžiui Windows ir Linux).

[6] Ekstranetinės aplikacijos – aplikacijos (programos), kurios yra įdiegtos vidiniame įmonės/organizacijos tinkle, tačiau yra pasiekiamos ir iš išorės tinklų, dažnai naudojantis viešomis komunikacijos sistemos.

[7] Q Public licenzija – šią licenziją užregistravo kompanija [Trolltech](#). Šios licenzijos esmė – produktas, naudojantis Q Public licenziją yra nemokamas, tačiau programinės įrangos kūrėjai privalo viešai pateikti bet kokį kodą, kuris nors kaip yra susijęs su Q Public licencijuotu produktu. Visas pateiktas kodas automatiškai patenka po pirminio produkto kūrėjų licenzija (t.y. jam galioja visos pirminio produkto licenzijos nuostatos).

[8] [WiKi](#) – tinklinė svetainė, kurios turinį gali redaguoti bet kuris vartotojas. Tai lyg vieša visų nuolatos papildoma enciklopedija.

[9] TCL – „[Tool Command Language](#)“ (liet. Įrankių komandinė kalba). Tai skriptų programavimo kalba, kurią 1988 metais sukūrė John Ousterhout. Kartu su Tk GVS (grafinės vartotojo sąsajos) įrankiu, dažnai vadinama **Tcl/Tk**.

[10] BLOB – Binary large object (liet. Dvejetainiai dideli objektai). Taip vadinami binariniai duomenys, saugomi vientisame pavidale duomenų bazėse. Dažniausiai BLOB būna įvairūs multimedijos failai (audio, video, piešiniai ir t.t.). Deja standarto, aprašančio BLOB tarp skirtingų duomenų bazių, nėra.

[11] RI (Relation integrity, liet. Ryšių vientisumo) taisyklės – taisyklės, nusakančios veiksmus su duomenimis, susietais keliose lentelėse.

Priedas

Anotacija (summary)

Lietuviškai

Darbo organizavimo valdymo sistema

Darbų planavimas, paskirstymas ir apskaita, šiais laikais vaidina didelę reikšmę bet kurios įmonės ar organizacijos gyvenime. Ar tai būtų nedidelė įmonė, kurioje dirbama su vienu ar keliais projektais, ar stambi įmonė, kurioje dirba daug žmonių ir jie užsiima keliasdešimčia projektų vienu metu, darbo planavimo ir apskaitos problema vis tiek išlieka aktuali. Darbo organizavimo valdymo (DOV) sistemos kaip tik ir skirtoms tokioms užduotims spręsti.

Šio darbo eigoje buvo atrinktos ir palygintos kelios sistemos, kaip galimos kandidatės į UAB „Baltic Data Center“ DOV sistemą. Po jų įvertinimo buvo pasirinkta „Bugzilla“, kurios laukė nemažos modifikacijos, kurias, norint patenkinti iškeltus įmonės reikalavimus DOV sistemai, reikėjo įgyvendinti. Galiausiai, sėkmingai atlikus visas numatytas modifikacijas, „Bugzilla“ įgavo visiškai naujus funkcionalumus, su kuriais ji pilnai patenkino įmonės iškeltus reikalavimus ir buvo pradėta naudoti kaip pagrindinė priemonė kelių įmonės skyrių darbų planavimui, paskirstymui ir apskaitai.

In English

Work organization management system

Work planning, distribution and accounting plays big role in the life of companies or organizations nowadays. If it's a small company, which develops one or two projects, or if it's a large enterprise with a lot of workers and many ongoing projects at the same time, the issue of work planning and accounting is still actual. Work organization management (WOM) systems are designed for solving problems just like that.

In a process of this work, there were analyzed several systems, considering them as possible candidates for UAB "Baltic Data Center" WOM system. After inspection of each of this system, "Bugzilla" was chosen to be the one. But it had to be modified a lot, to meet the expectations of the company. Finally, after all the modifications it has undergone, "Bugzilla" got completely new functionality, with which it met all the requirements of an organization and it was started to use as a main tool to plan, distribute and account work of several departments of an organization.

Pažyma iš UAB “Baltic Data Center”