

VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
PROGRAMŲ SISTEMŲ KATEDRA

Komandinio programų kūrimo proceso tyrimas

Analysis of the Team Software Process (TSP)

Magistro baigiamasis darbas

Atliko: Andrej Gavrilov

Darbo vadovas: Doc. Saulius Ragaišis

Recenzentas: Andrius Dienys

Vilnius – 2008

Santrauka

Siekiant pagerinti kuriamų programinių produktų kokybę, IT įmonės taiko skirtingus proceso modelius. Populiariausi iš jų gebėjimo brandos modelis (CMMI) ir ISO 15504. Šie modeliai nusako, *kas* (kokie procesai) turi būti vykdoma brandžioje organizacijoje, tačiau neatsako į klausimą: *kaip* tai turi būti daroma. Komandinis programų kūrimo procesas (TSP) pateikia strategiją, kartu su procedūrų rinkiniu, skirtą disciplinuotų programų kūrimo metodų naudojimui komandos lygyje.

Pagrindinė su TSP susijusi problema – oficialios dokumentacijos trūkumas. Pirmas darbo tikslas yra rekonstruoti TSP metodiką. Antras tikslas yra sukurti TSP diegimo metodiką. Pirmoje darbo dalyje yra aprašyta bendra TSP struktūra, TSP ryšys su asmeniniu programų kūrimo procesu (PSP) ir brandos modeliu CMMI. Antroje dalyje yra pateiktas rekonstruotas TSP modelis ir TSP diegimo metodika apibrėžtam projekto komandos tipui.

Raktiniai žodžiai: komandinis programų kūrimo procesas, TSP diegimas.

Summary

IT organizations use different models to guide their software process improvement efforts. The most popular are the Capability Maturity Model Integration for Software (CMMI-SW) and ISO 15504. Still these organizations often struggle with implementation details. Both the CMMI-SW and ISO 15504 describe *what* an organization at a high level of process maturity should be doing, but do not says *how* it should be implemented. The Team Software Process (TSP) provides a framework as well as a set of processes, procedures, guidelines, and tools for project teams to use in the mature software development process.

The main problem of the Team Software Process is the lack of official documentation. So the first goal of this master thesis is to reconstruct TSP methodology. The second goal is to produce TSP implementation guide. This master thesis contains the main structure of the TSP, it's relation with the Personal Software Process (PSP) ant the Capability Maturity Model (CMMI), reconstructed TSP model and TSP implementation guide for defined project team type.

Keywords: Team Software Process (TSP), TSP implementation.

Turinys

Įvadas.....	6
1. TSP metodikos apžvalga	8
1.1. Pagrindiniai principai	8
1.2. TSP ir asmeninis programų kūrimo procesas (PSP).....	9
1.3. TSP ir gebėjimų brandos modelis (CMMI).....	10
1.4. TSP instruktoriaus vaidmuo	11
1.5. TSP projekto komanda	12
1.6. TSP struktūra	14
2. TSP metodikos rekonstrukcija.....	16
2.1. Rekonstrukcijos metodika	17
2.2. Bendra TSP ciklo struktūra (DEV scenarijus).....	18
2.3. TSP startavimo (LAU) scenarijus	20
2.3.1. Pasiruošimas startavimui	26
2.3.2. Scenarijus LAU1	27
2.3.3. Scenarijus LAU2	28
2.3.4. Scenarijus LAU3	29
2.3.5. Scenarijus LAU4	32
2.3.6. Scenarijus LAU5	34
2.3.7. Scenarijus LAU6	35
2.3.8. Scenarijus LAU7	37
2.3.9. Scenarijus LAU8	39
2.3.10. Scenarijus LAU9	40
2.3.11. Scenarijus LAUPM	41
2.3.12. Scenarijus REL1	42
2.4. Inspektavimo (INS) scenarijus	44
2.5. Reikalavimų sudarymo (REQ) scenarijus	47
2.6. Aukšto lygio projektavimo (HLD) scenarijus	51
2.7. Realizavimo (IMP) scenarijus	55
2.8. Modulių testavimo (UT) scenarijus.....	57
2.9. Testavimo (TEST) scenarijus	60

2.9.1.	Sistemos surinkimo (TEST1) scenarijus	62
2.9.2.	Integravimo testavimo (TEST2) scenarijus	64
2.9.3.	Sistemos testavimo (TSET3) scenarijus	65
2.9.4.	Scenarijus TSETD	67
2.10.	Projekto aptarimo (PM) scenarijus	68
2.11.	Kassavaitinio susitikimo (WEEK) scenarijus.....	70
3.	TSP diegimo metodika	74
3.1.	Projekto komanda	75
3.2.	Apmokymai	76
3.2.1.	Asmeninis programų kūrimo procesas (PSP).....	77
3.2.2.	Komandinis programų kūrimo procesas (TSP)	79
3.3.	TSP proceso diegimo apimtis	80
3.4.	TSP praktinio taikymo rekomendacijos	81
3.4.1.	Startavimo proceso adaptavimas	81
3.4.2.	Bendro plano sudarymo (LAU4) scenarijaus taikymas.....	84
3.4.3.	Kokybės plano sudarymo (LAU5) scenarijaus taikymas	85
	Rezultatai ir išvados	87
	Šaltinių sąrašas	88
	Terminų žodynas	90
	1 priedas. Pagrindinių TSP elementų sąrašas	91
	2 priedas. TSP kokybės planas	93
	3 priedas. TSP formos	95

[vadas

Daugybė įmonių visame pasaulyje užsiima programinės įrangos kūrimu. Konkurencija šioje rinkoje yra pakankamai didelė, ir bene geriausias būdas įgyti pranašumą – yra gaminti kokybiškus, produktus, kurie turi didesnę paklausą. Siekti kokybės galima skirtingais būdais. Didelį indėlį į galutinio produkto kokybę įneša programų sistemų kūrimo proceso kokybė. Kitais žodžiais, kuo tvarkingiau organizacija planuoja, vykdo ir valdo kūrimo ir kitus darbus, tuo didesnė tikimybė, kad ši organizacija sukurs kokybišką produktą, t.y. produktas bus sukurtas laiku, už numatytą kainą ir atitiks numatytus reikalavimus.

Šiai dienai egzistuoja nemažai programų kūrimo proceso modelių ir standartų, populiariausi iš jų CMMI ir ISO 15504 standartas. Šie modeliai nusako, kokios veiklos turi būti įtrauktos į programinės įrangos kūrimo procesą, kitaip tariant, kokios proceso sritys turi būti realizuotos (t.y. *kas* turi būti daroma) brandžioje organizacijoje. Šių modelių aprašymai užima šimtus puslapių, tačiau, dažnai net to nepakanka. Nei CMMI, nei ISO 15504 neatsako į klausimą: *kaip* tai turi būti daroma? Todėl dažnai organizacijoms tenka taikyti vieną iš egzistuojančių procesų (XP, RUP, TSP ir t.t.), kurie papildo minėtus modelius ir padeda pasiekti šių modelių išskeltus tikslus.

Komandinis programų kūrimo procesas (TSP) - smarkiai populiarėjanti Jungtinėse Amerikos Valstijose metodika, kurią sėkmingai įsodiegė tokios žymios organizacijos kaip Boeing, NAVAIR ir Teradyne Inc. Visos šios organizacijos pastebėjo ženklus programinės įrangos kūrimo proceso pagerėjimus: sumažėjo daromų defektų skaičius, padidėjo laiko ir projekto apimties vertinimo tikslumas, darbo produktyvumas ir kt. [DM03, Kim04, WMP05]. Pagrindinė su TSP susijusi problema – literatūros ir informacijos trūkumas. Nėra viešai prieinamo pilno TSP aprašo, ir, kaip bebūtų keista, netgi internete, kur galima rasti beveik viską, pasitaiko tik atskiri TSP aprašo fragmentai. Pirmas šio magistro darbo tikslas yra rekonstruoti TSP metodika.

Deja, programų sistemų kūrimo srityje Lietuva vis dar atsilieka nuo Jungtinių Amerikos Valstijų ir Vakarų Europos. Mūsų šalies įmonės ne taip seniai suprato proceso kokybės ir brandos reikšmę ir dauguma jų tik dabar pradeda diegtis kurią nors iš egzistuojančių PS kūrimo metodikų (RUP, XP ir pan.) arba, remiantis sukaupta patirtimi, kurti savo. Ryšium su šia tendencija, iškyla savalaikis ir aktualus klausimas: ar komandinio programų kūrimo (TSP) metodika yra tinkamas ir Lietuvos įmonėms? Kokios yra TSP taikymo galimybės Lietuvoje, atsižvelgiant į mūsų šalies ypatumus? Nežiūrint TSP populiarumo ir sėkmingo taikymo Jungtinėse Amerikos Valstijose, iki šiol nėra nei vieno straipsnio, nagrinėjančio TSP taikymo galimybes Lietuvoje ar kaimyninėse šalyse.

Bet kurio programinės įrangos kūrimo modelio diegimas ir taikymas priklauso nuo procesą taikančios komandos ir vykdomų projektų tipo. Todėl, prieš kuriant TSP diegimo metodiką, reikia apibrėžti komandos tipą, į kuri bus orientuota TSP diegimo metodika. Antras šio darbo tikslas yra pateikti TSP diegimo metodiką apibrėžtam komandos tipui.

Šiuo darbu yra siekiama:

- rekonstruoti TSP metodiką;
- sukurti TSP diegimo metodiką apibrėžtam komandos tipui.

Pirmame darbo skyriuje yra pateikta bendra komandinio programų kūrimo proceso (TSP) struktūra. Yra aprašyti pagrindiniai TSP principai, aptarti sąryšiai su asmeniniu programų kūrimo procesu (PSP) ir gebėjimo brandos modeliu (CMMI), apibrėžta projekto komandos sudėtis.

Antras magistro darbo skyrius yra skirtas TSP modelio rekonstrukcijai. Šiame skyriuje yra pateikta rekonstrukcijos metodika ir pats rekonstruotas TSP modelis. Rekonstruotas modelis apima pagrindinius TSP scenarijus ir jų aprašymus.

Trečiame skyriuje yra apibrėžtas projekto komandos tipas ir aprašyta šiam komandos tipui skirta TSP diegimo metodika.

1. TSP metodikos apžvalga

Pradinė komandinio programų kūrimo proceso (TSP) versija buvo sukurta 1996 metais Jungtinėse Amerikos Valstijose, šios metodikos kūrėjas - Watts S. Humphrey. Nuo 1996 metų TSP metodika buvo tobulinama ir pirmoji knyga apie TSP [Hum99] pasirodė tik 1999 metais, o oficiali SEI techninė ataskaita apie TSP ([Hum00]) - 2000 metais. Tiesa, metodika yra tobulinama ir plėtojama iki šiol. Iš pat pradžių TSP buvo skirtas nedidelėms (iki 10 žmonių) komandoms, šiandien TSP yra taikomas komandoms iki 20 žmonių, yra sukurtas išplėtimas TSPm, skirtas kelioms komandoms (bendras dalyvaujančių žmonių skaičius nuo 100 iki 200).

Komandinis programų kūrimo procesas - metodika, pateikianti strategiją, kartu su procedūrų rinkiniu, skirtą disciplinuotų programų kūrimo metodų naudojimui komandos lygyje. Ypatingas dėmesys TSP yra skiriamas:

- efektyviam komandos darbui;
- projekto planavimui;
- kokybės valdymui.

1.1. Pagrindiniai principai

TSP remiasi šiais pagrindiniais principais [DM03]:

- inžinieriai daugiausiai žino apie darbą ir gali sudaryti geriausius planus;
- kai inžinieriai patys planuoja savo darbą, jie jaučiasi įsipareigoję;
- tikslus projekto sekimas reikalauja detalių planų ir tikslų duomenų;
- tikslūs duomenis gali surinkti tik tie žmonės, kurie ir atlieka darbus;
- tam, kad minimizuoti ciklo trukmę, inžinieriai turi subalansuoti visų komandos narių darbo apkrovas;
- tam, kad maksimaliai padidinti produktyvumą, pirmiausia reikia skirti dėmesį kokybei.

Didelis dėmesys TSP yra skiriamas šioms veikloms:

- suburti efektyvią komandą ir paskirstyti roles;
- apibrėžti komandos strategiją (kaip planuoti ir vykdyti projektą);
- planuoti darbą ir laikytis šio plano;
- numatyti galimas rizikas ir jų šalinimo būdus;
- komandos viduje tolygiai paskirstyti darbus;
- efektyviai valdyti projektą;

- planuoti ir valdyti galutinio produkto kokybę;
- valdyti defektus.

Taipogi, TSP apibrėžia:

- kaip komandoje yra priimami sprendimai;
- kokios veiklos turi būti vykdomos programų kūrimo proceso metu;
- kaip vertinti projekto būseną (angl. project status).

1.2. TSP ir asmeninis programų kūrimo procesas (PSP)

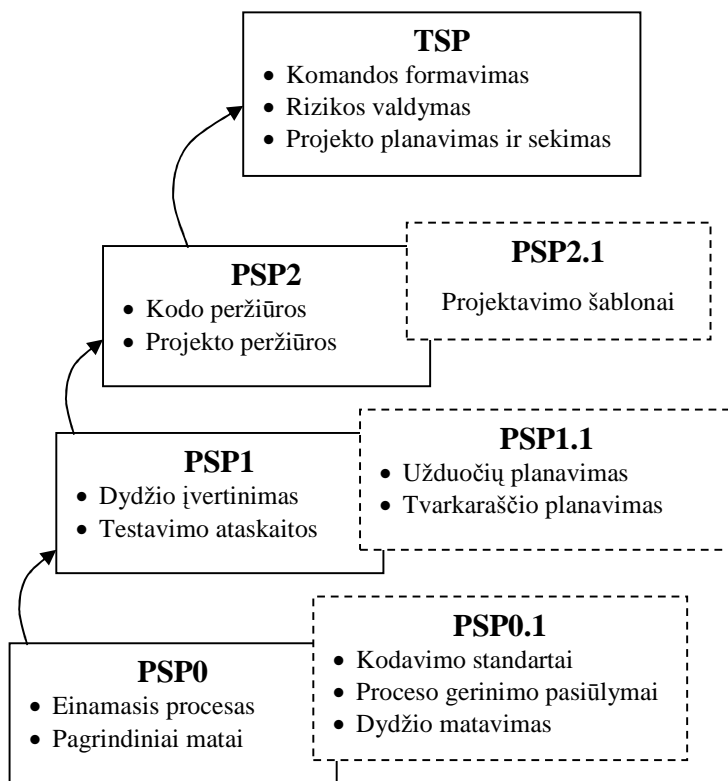
TSP komandos efektyvumas priklauso nuo ją sudarančių narių pasiruošimo lygio, todėl visi jos nariai turi būti tinkamai apmokyti. Tam kad komanda galėtų sėkmingai naudoti TSP, visi jos nariai turi būti apmokyti PSP, kuris suteikia komandos nariams individualaus darbo žinias ir įgūdžius, kuriais remiasi TSP matavimai, projekto planavimas ir kokybės vertinimas.

PSP remiasi šiais pagrindiniais principais [DM03]:

- visi inžinieriai yra skirtingi; tam, kad pasiektų maksimalų efektyvumą, inžinieriai turi planuoti savo darbą, ir jų planai turi remtis jų pačių sukauptais individualiais duomenimis;
- norint pastoviai tobulinti savo darbą, inžinieriai turi matuoti savo darbą ir gautus rezultatus naudoti tobulinimui;
- tam, kad pagamintų kokybišką produktą, inžinieriai turi jausti asmeninę atsakomybę už savo pagamintos produkcijos kokybę; aukštos kokybės produktas nėra atsitiktinumas, inžinieriai turi stengtis dirbti kokybiškai;
- pigiau surasti ir ištaisyti defektą anksčiau negu vėliau;
- defektų prevencija yra efektyvesnė už defektų radimą ir taisymą;
- geriausias būdas atlikti darbą - visada greičiausias ir pigiausias būdas.

PSP nagrinėjimo sritis – individualaus komandos nario darbas. PSP apima detalių planų sudarymą, informacijos apie kūrimo proceso rinkimą ir panaudojimą, planų kūrimą, suplanuotos ir uždirbtos vertės metodo naudojimą projekto eigai sekti, produktų kokybės vertinimą ir valdymą ir kt.

TSP savo ruožtu nagrinėja programų kūrimo procesą komandiniame lygyje. Ryšys tarp asmeninio (PSP) ir komandinio (TSP) programų kūrimo procesų yra parodytas 1 paveikslėlyje.

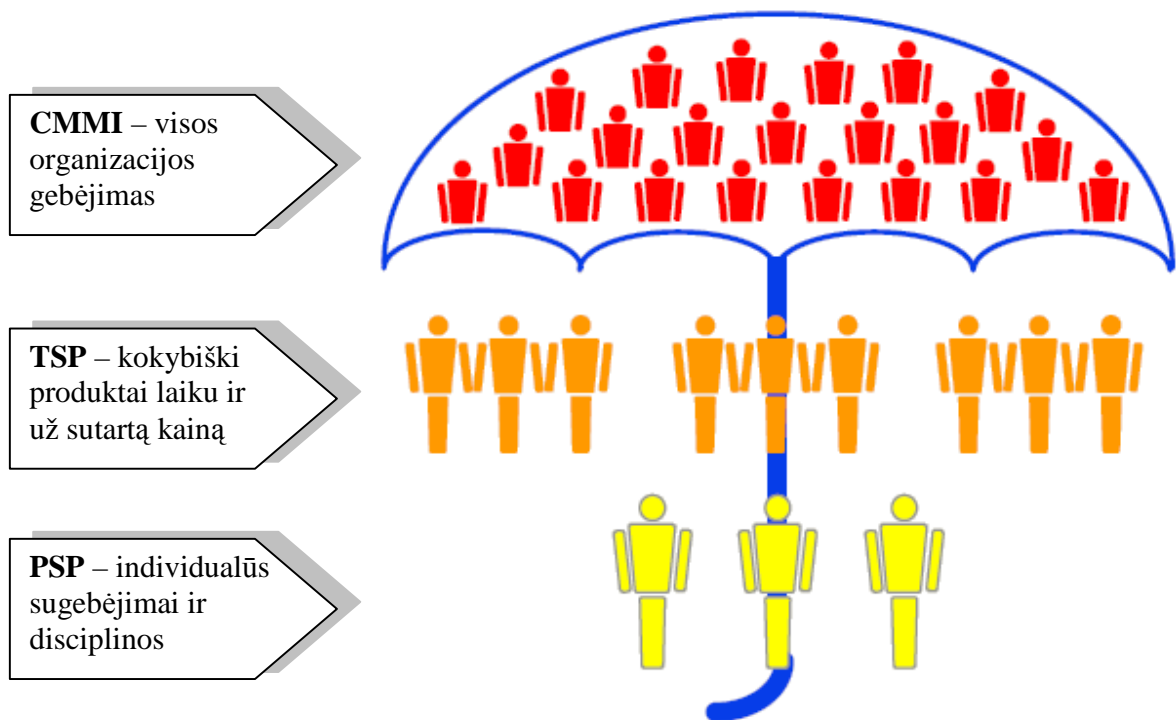


1 pav. PSP schema [Hum06B]

1.3. TSP ir gebėjimų brandos modelis (CMMI)

Sutinkama nemažai publikacijų, kuriose yra pateikiamas TSP ir CMM/CMMI modelių sugretinimas [DMcH03, WMP05, MW05]. Šiose publikacijose yra nagrinėjamos TSP ir CMM/CMMI suderinimo galimybės, parodoma, kaip TSP gali papildyti minėtus modelius, padėti pasiekti CMM/CMMI keliamus tikslus. Tyrimai rodo, kad TSP sėkmingai palaiko CMMI veiklas nuo 1 iki 5 brandos lygio.

TSP užima vietą tarp PSP (individualus lygis) ir CMMI (organizacijos lygis). Komandinis programų kūrimo procesas apjungia savyje PSP ir CMMI bruožus ir užtikrina sėkmingą programų kūrimo procesą projekto (komandos) lygyje. TSP nėra skirtas pakeisti gebėjimų brandos modelį, o yra sukurtas tam, kad paremtų CMMI visuose brandos lygiuose. Grafiškai PSP, TSP ir CMMI sąryšis yra parodytas 2 paveikslėlyje.



2 pav. Ryšys tarp PSP, TSP ir CMMI [McH03]

Praktiškai TSP derinimo su CMM nauda buvo parodyta NAVAIR organizacijoje, kai perėjimas nuo 1 pakopinio CMM-SW modelio lygio prie 4 lygio šioje organizacijoje užtruko vos 30 mėnesių, kai paprastai toks perėjimas reikalauja dvigubai daugiau laiko [DMcH03].

1.4. TSP instruktoriaus vaidmuo

TSP kūrėjai pabrėžia instruktoriaus dalyvavimo projekto eigoje svarbą. Ypač svarbus yra instruktoriaus vadovavimas mažai patirties TSP turinčioms komandoms. Pagrindinės TSP instruktoriaus užduotys:

1. TSP instruktoriaus yra atsakingas už pasiruošimą projekto startavimui. Jis turi užtikrinti, kad visi žingsniai, išvardinti pasiruošimo startavimui kontroliniame sąraše (žr., 2.3.1 skyrių), yra atlikti. Jeigu kažkuris iš šių žingsnių nėra atliktas, TSP instruktoriaus turi imtis veiksmų, skirtų užtikrinti pasiruošimo žingsnio atlikimą (pvz., sudaryti startavimo proceso tvarkaraštį ir pasirūpinti kad šiem susitikimams būti išskirtos patalpos ir visa reikalinga infrastruktūra). Jeigu instruktoriaus negali arba neturi įgaliojimų imtis tokių veiksmų, jis turi susisiekti su organizacijos vadovybę ir inicijuoti šių darbų vykdymą. Pavyzdžiui, TSP

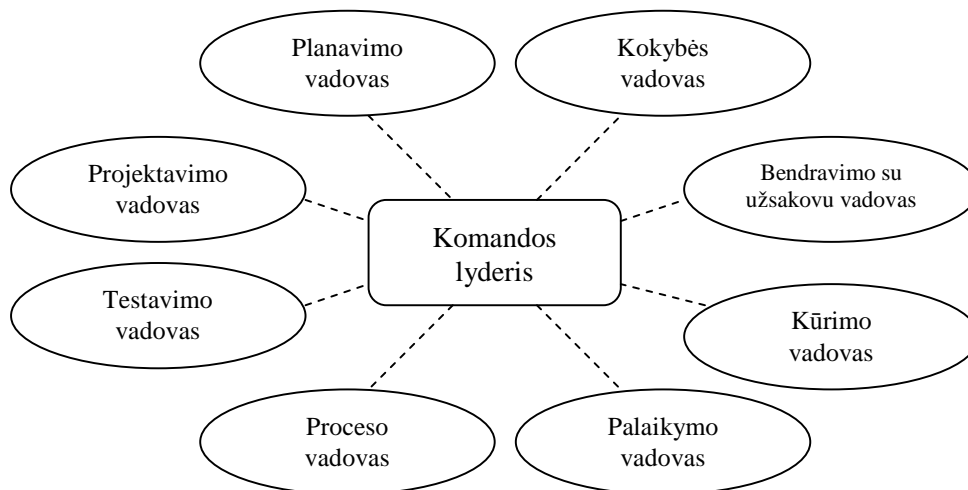
instruktorius gali nuturėti įgaliojimų pačiam apmokyti komandos narius PSP arba TSP, šiuo atveju jis privalo išaiškinti vadovybei šių mokymų svarbą ir įtikinti pasiūsti komandos narius į reikalingus mokymus.

2. Viso projekto metu TSP instruktorius turi prižiūrėti, kad komandos darbas vyktų nenukrypstant nuo TSP aprašyto proceso, ir visi komandos nariai atlieka tvarkingai renka reikalingus duomenis ir pildo TSP formas. Jeigu yra pastebimi nukrypimai nuo TSP proceso, arba kažkuris komandos narys neprisilaiko TSP procedūrų, TSP instruktorius privalo išaiškinti ir pašalinti nukrypimų priežastis. Jeigu komandos narys atsisako dirbti pagal TSP, instruktorius turi įtikinti komandos narį laikytis TSP procedūrų arba bandyti paveikti komandos narį kitais būdais (pvz., per jo tiesioginį vadovą). Tuo atveju, kai komandos narys vis tiek atsisako dirbti pagal TSP, jis turi būti pašalintas iš projekto ir pakeistas kitu darbuotoju.
3. Jeigu projekto metu iškila rimtesnės problemos, TSP instruktorius turi padėti komandai jas išspręsti. Instruktorius turi nemažai patirties TSP projektuose ir dažnai gali pasiūlyti problemos sprendimą, kuris nėra akivaizdus mažiau patyrusiems komandos nariams.
4. TSP instruktorius turi užtikrinti komandos priimamų sprendimų objektyvumą. Sprendžiant iškilusias problemas, komandos nariai gali turėti visiškai skirtingas nuomones, kaip šios problemos turi būti spendžiamos. Dažnai, ieškant vieningo sprendimo, lemiamą įtaka ieškant konsensuso turi ne objektyvus faktai, o viršesnių kolegų užsispyrimas arba kurio nors vieno komandos nario asmeninė įtaka kolegoms (profesionalo savo srityje įvaizdis, gebėjimas įtikinti kitus ir pan.). Taip priimti sprendimai dažnai nėra optimalūs ir gali sulėtinti projekto darbus arba turėti kitų neigiamų pasekmių. Instruktoriaus užduotis yra neleisti komandai priimti tokių sprendimų, išsakyti savo nešališką nuomonę ir įtikinti visus komandos narius priimant sprendimus remtis objektyviais faktais o ne subjektyvia nuomone.

Dirbant su patyrusiomis komandomis, instruktorius atlieka konsultanto vaidmenį, vienas instruktorius gali vadovauti kelioms komandoms. Tuo tarpu dirbant su mažiau patyrusia komanda, instruktoriui yra patariama numatyti pilną etatą [Hum06A].

1.5. TSP projekto komanda

TSP numato komandos lyderio ir 8 vadovų roles [MW05], grafiškai TSP komandos sudėtis yra pavaizduota 3 paveikslėlyje.



3 pav. TSP rolės [MW05]

Kiekvienas iš vadovų yra atsakingas už tam tikrų veiklų vykdymą ir artefaktų sukūrimą. T.y. jis turi koordinuoti ir prižiūrėti jo valdomos srities darbus:

- bendravimo su užsakovu vadovas yra atsakingas už reikalavimų sudarymą, reikalavimų pakeitimų valdymą, reikalavimų standartų sukūrimą ir palaikymą;
- projektavimo vadovas – vadovavimas komandai projektavimo metu, projektavimo pakeitimų valdymas, projektavimo standartų sukūrimas ir palaikymas;
- kūrimo vadovas – vadovavimas komandai kūrimo fazėje, projekto realizavimo pakeitimų valdymas, standartų, susijusių su realizavimu, sukūrimas ir palaikymas;
- testavimo vadovas – testavimo planavimas, palaikymas, testavimo rezultatų analizė;
- planavimo vadovas – vadovavimas komandai planavimo metu, projekto būsenos sekimas;
- proceso vadovas – proceso palaikymas, sekimas, analizė, proceso problemų ir gerinimo pasiūlymų valdymas;
- palaikymo vadovas – įrankių palaikymas, konfigūracijos valdymas, pakeitimų kontrolė, pakartotinis panaudojimas;
- kokybės vadovas – kokybės palaikymas, sekimas, analizė.
- komandos lyderis - yra atsakingas už žmonių valdymą, vadovavimą komandai, kokybės valdymą, projekto valdymą, komandos priimtas atsakomybes.

Visi vadovai turi paruošti ir kas savaitinių susitikimų metu pristatyti ataskaitas apie savo darbų eigą.

Dažniausiai, komandos lyderis neprisiima jokių kitų rolių, nes jo tiesioginės pareigos atima praktiškai visą projektui skirtą laiką. Visi kiti komandos nariai atlieka pasirinkto vadovo rolę ir tuo pat metu užsiima produkto kūrimo darbais [Hum05]. Jeigu komandos narių yra mažiau negu yra

numatyta rolių, vienam ar daugiau komandos narių gali būti priskirtos dvi rolės. Tuo atveju, jei komandos narių yra daugiau negu rolių, tie komandos nariai, kuriems rolių neužteko gali būti arba vieno iš vadovų asistentais, arba tiesiog inžinieriais, t.y. užsiimti tik programinio produkto kūrimo darbais.

1.6. TSP struktūra

TSP yra orientuotas į evoliucinį kūrimo modelį, todėl TSP numato kelis kūrimo ciklus. Kiekvienas ciklas prasideda nuo startavimo, kurio metu komanda sudaro/peržvelgia bendrą projekto planą ir sudaro detalų ciklo planą. Vykdamas projektą, komanda periodiškai grįžta prie pakartotinio startavimo (angl. relaunch) tam, kad atnaujintų ir detalizuotų sekančio ciklo planą.

TSP modelis numato 6 pagrindines ciklo fazes:

- startavimas;
- reikalavimai;
- projektavimas;
- realizavimas (kodavimas);
- testavimas;
- aptarimas.

TSP pateikia rinkinį formų, skirtų planavimui, projekto eigos sekimui, defektų ir rizikų fiksavimui ir istorinių duomenų rinkimui, bei visą eilę scenarijų (skriptų), detaliam aprašančių fazių veiklas. Pilnas TSP elementų sąrašas yra pateiktas 1 priede.

Viso yra žinoma 31 komandinio programų kūrimo proceso scenarijų. Penkiolika iš jų yra skirti bendresnių scenarijų detalizavimui. Oficialiai paskelbtas ir pilnai žinomas iš visų 31 yra vienintelis startavimo (LAU) scenarijus (žr. 3 priedą). Apie kitus scenarijus informacija yra nepilna – yra žinoma scenarijaus paskirtis ir dalinis tekstinis aprašymas. O apie kai kuriuos scenarijus, pavyzdžiui MAINT, apskritai yra žinomas tik pavadinimas.

1 lentelė. TSP scenarijai [DMcH03]

Scenarijaus trumpinys	Scenarijaus pavadinimas
DEV	Bendras kūrimo procesas
MAINT	Bendras priežiūros procesas (angl. overall maintenance process)
ANA	Analizės procesas
HLD	Aukšto lygio projektavimas

Scenarijaus trumpinys	Scenarijaus pavadinimas
IMP	Realizavimas
IMP6 (UT)	Modulių testavimas ir testų kūrimas (IMP scenarijaus šeštas žingsnis) ¹
INS	Inspektavimo procesas
LAU	Startavimo procesas
LAU1	Projekto apžvalga (LAU scenarijaus pirmas susitikimas)
LAU2	Tikslai ir rolės (LAU antras susitikimas)
LAU3	Strategija, procesas, palaikymas (LAU trečias susitikimas)
LAU4	Bendras komandos planas (LAU ketvirtas susitikimas)
LAU5	Kokybės planas (LAU penktas susitikimas)
LAU6	Detalus einamojo ciklo planas (LAU šeštas susitikimas)
LAU7	Rizikų įvertinimas (LAU septintas susitikimas)
LAU8	Pasiruošimas susitikimui su vadovybe (LAU aštuntas susitikimas)
LAU9	Susitikimas su vadovybe (LAU devintas susitikimas)
LAUPM	Startavimo aptarimas
MTG	Bendras susitikimo procesas
PM	Projekto aptarimas
REL	Pakartotinis startavimas
REL1	Būsenos ir valdymo pasikeitimai (REL scenarijaus pirmas susitikimas) ²
REQ	Reikalavimai
STATUS	Projekto būsenos pristatymo vadovybei ir užsakovui susitikimas
TEST	Testavimas
TEST1	Sistemos surinkimas (angl. build) (TEST scenarijaus pirmas žingsnis)
TEST2	Integravimas (TEST antras žingsnis)
TEST3	Sistemos testavimas (TEST trečias žingsnis)
TESTD	Testavimo metu aptiktų defektų šalinimas
WEEK	Kassavaitinis komandos susitikimas
MTG	Bendras susitikimo procesas

DEV ir MAINT scenarijai yra sudėtiniai. DEV scenarijus aprašo naujo produkto kūrimo procesą, o MAINT – produkto priežiūros procesą. Šie scenarijai susideda iš kitų scenarijų, aprašančių pagrindinius procesus, sudarančius produkto kūrimą/priežiūrą.

¹ - atskirą scenarijų turi tik šeštas realizavimo scenarijaus (IMP) žingsnis, visi kiti žingsniai yra pakankamai aiškūs arba per daug specifiniai, kad būtų galima juos apibendrinti visiems įmanomiems projektams.

² - visi kiti pakartotinio startavimo žingsniai sutampa su LAU scenarijaus žingsniais, detaliau apie tai yra parašyta skyrelyje 2.1 (bendra TSP struktūra)

Scenarijus LAU susideda iš 10 mažesnių scenarijų (LAU1, ..., LAU9 ir LAUPM), aprašančių kiekvieno iš startavimo susitikimų veiklas. Testavimo scenarijus TEST apjungia savyje 4 atskirus scenarijus, kurie aprašo testavimo proceso etapus:

- sistemos surinkimas – skirtas įsitikinti, kad atskiros produkto dalys yra suderinamos tarpusavyje (TEST1 scenarijus);
- integravimo – skirtas atskirų produkto dalių sąveikos testavimui (TEST2 scenarijus);
- sistemos – skirtas pilnai surinktos sistemos testavimui (TEST3 scenarijus);
- TSP numato ir testavimo metų rastų defektų šalinimo veiklas (TESTD scenarijus).

Scenarijai REL ir IMP turi savyje po vieną žemesnio lygio scenarijų. Visi kiti scenarijai yra paprasti, t.y. į jų sudėtį neįeina jokie kiti scenarijai.

2. TSP metodikos rekonstrukcija

TSP metodiką sudaro visa eilė elementų: scenarijai, rolių ir jų tikslų aprašymai, veiklų (pvz., konfigūracijos valdymas) aprašymai, formos ir t.t. Pilnos metodikos rekonstravimas yra labai didelis ir daug laiko užimantis darbas. Šiame darbe yra apsiribojama TSP scenarijų rekonstrukcija, nes jie sudaro TSP pagrindą ir struktūrą. Be TSP scenarijų yra praktiškai neįmanomas gilesnis TSP proceso tyrimas ir taikymas.

Rekonstruojant TSP metodiką buvo siekiama atkurti TSP aprašą jo originalioje formoje. Rekonstrukcijos tikslas - atkurti visus TSP scenarijus, išvardintus 1 lentelėje. Tačiau, apie kai kuriuos scenarijus nepavyko rasti praktiškai jokios informacijos, todėl šie scenarijai liko neaprašyti: MAINT, ANA, STATUS, MTG.

Bendro priežiūros proceso (MAINT) scenarijus yra analogiškas DEV scenarijui. Į jį įeina pagrindiniai produkto priežiūros procesai (jų scenarijai). Tačiau jokios konkrečios informacijos apie šį scenarijų nėra ir lieka neaišku, kokie tiksliai procesai sudaro MAINT scenarijų. Yra galimos dvi šios informacijos neatskleidimo priežastys. Pirma, kad jis yra labai panašus į DEV scenarijų ir yra sudarytas iš tų pačių procesų (su tam tikrais pakeitimais). Antra, kad TSP kūrėjai, vykdydami informacijos neatskleidimo politiką, dėl kurios nėra paviešinti praktiškai visi TSP scenarijai, sąmoningai slepia šią informaciją. TSP pristatymams ir reklamai yra naudojamas bendras kūrimo procesas DEV, o informacija apie palaikymo scenarijų MAINT yra suteikiama tik už pinigus.

Analizės scenarijus yra minimas tik tuose darbuose, kuriuose yra sugretinami TSP ir CMMI modeliai. Kaip bebūtų keista, analizės scenarijus nėra minimas nei TSPi [Hum99], nei TSP [HO99] aprašymuose. Galima to priežastys yra ta, kad ANA scenarijus buvo panaikintas, o pats analizės procesas tapo reikalavimų sudarymo proceso dalimi. Pirmi reikalavimų (REQ) scenarijaus žingsniai

yra rinkos analizė ir reikalavimų išaiškinimas. Šios veiklos yra labai artimos analizės procesui. Dviejų scenarijų apjungimą patvirtina ir tas faktas, kad ANA scenarijus visur, kur yra minimas, visada stovi greta REQ scenarijaus.

Priežastis, dėl kurios nėra informacijos apie STATUS ir MTG scenarijus, matomai, yra ta, kad šie scenarijai nėra esminiai, norint suprasti TSP principus, o tik aprašo kelių susitikimų eigą. Be to, scenarijaus WEEK, aprašančio kassavaitinių susitikimų eigą, versija yra pateikta TSPi [Hum99] aprašyme. Visų susitikimų struktūra yra panaši, tai parodo ir startavimų susitikimų aprašymai, todėl STATUS ir MTG scenarijai nebuvo paviešinti.

2.1. Rekonstrukcijos metodika

Komandinio programų kūrimo proceso rekonstrukcija remiasi visa eile šaltinių. Informacija yra surenkama iš skirtingų knygų, straipsnių ir techninių ataskaitų ir yra sudedama kaip mozaika. Rekonstruotas TSP modelis susidės iš scenarijų, lentelės forma, ir jų tekstinio aprašymo. Tokia rekonstruoto modelio forma buvo pasirinkta dėl kelių priežasčių:

- tokia forma yra arčiausia originaliam TSP modeliui;
- lentelės leidžia sugrupuoti informaciją ir glaustai pateikti pagrindinius TSP scenarijų žingsnius ir veiklas, kas palengvina scenarijų naudojimą;
- tekstinis aprašymas padaro scenarijus suprantamesnius, paaiškina scenarijaus žingsnių seką, sąryšius, nurodo, kokios veiklos ir kaip yra vykdomos kiekvieno scenarijaus žingsnio metu.

Tokia kombinuota lentelių ir tekstinio aprašymo struktūra yra suprantamesnė ir patogesnė naudojimui negu vien lentelės arba vien tekstas.

TSP scenarijų ir formų sąrašas yra paimtas iš Noopur Davis ir Jim McHale techninės ataskaitos „Relating the Team Software ProcessSM (TSPSM) to the Capability Maturity Model for Software (SW-CMM)“ [DMcH03]. Šioje ataskaitoje yra pateiktas labiausiai pilnas TSP elementų (scenarijų, formų, rolių, specifikacijų ir standartų) sąrašas. Tačiau, šioje ataskaitoje TSP elementai yra tik išvardijami ir sugrupuojami pagal elementų pobūdį (pvz., scenarijai, formos). Autoriai nepateikia jokio aprašymo arba paaiškinimų, kaip šie elementai yra susiję ir koks yra jų turinys.

Oficialiai paskelbtas TSP startavimo (LAU) scenarijus [Hum00] ir įvado į TSP (angl., TSPi - TSP introduction) scenarijai [Hum99] yra naudojami kaip rekonstruojamų TSP scenarijų pagrindas. LAU scenarijus leidžia įsivaizduoti, kaip turėtų atrodyti originalūs TSP scenarijai, koks yra jų detalumo lygis. Savo ruožtu, TSPi scenarijai, nors ir yra supaprastintos originalių TSP scenarijų versijos ir jų struktūra bei detalumo lygis ne visada pilnai atitinka originalų TSP modelį, yra labai

vertingi nustatant pagrindinius rekonstruojamų scenarijų žingsnius ir jų turinį, t.y. TSPi scenarijai yra pagrindinis paaiškinamosios (panaudotos scenarijų aprašymuose) informacijos apie rekonstruojamų TSP scenarijų paskirtį ir turinį šaltinis.

Nors TSPi scenarijuose pateikta informacija ir užduoda pagrindines gaires rekonstrukcijai, jos neužtenka tam, kad sukurti pakankamai detalų aprašymą. Rekonstrukcijai atlikti reikalinga informacija yra renkama iš visų literatūros sąraše nurodytų šaltinių. Nemažai vertingos informacijos apie TSP scenarijų turinį (žingsnius, veiklas) detalesniame, negu TSPi scenarijuose, lygyje yra paimta iš TSP aprašymo (angl., tutorial) [HO99]. Startavimo scenarijų rekonstrukcijos metu labai pravertė W. Humphrey ataskaitos ir straipsniai apie TSP [Hum00, Hum02].

TSP ir CMM brandos modelio palyginimo ataskaitos [DMcH03, McH03, MW05, WMP05] praverčia ieškant detalios informacijos apie TSP scenarijų rėmuose atliekamas veiklas ir kuriamus artefaktus. Tačiau, tam kad atlikti pakankamai detalią ir tikslią rekonstrukciją, visa galima ir naudinga informacija yra renkama iš visų prieinamų šaltinių.

Komandos lyderio ir TSP instruktoriaus vaidmens detalės yra paimtos iš W. Humphrey knygu „TSPSM-Leading a Development Team” [Hum05] ir „TSPSM: Coaching Development Teams” [Hum06A]. Informacija apie asmeninį programų kūrimo procesą (PSP) ir jo naudojimą TSP procese yra išdėstyta W. Humphrey straipsniuose „PSPSM: A Self-Improvement Process for Software Engineers” [Hum06B], „How PSP and TSP Can Impact an Organization” [Hum02] ir Donald R. McAndrews ataskaitos „The Team Software Process (TSPSM): An Overview and Preliminary Results of Using Disciplined Practices” [McA00]. Pravertė ir TSP praktinio taikymo aprašymai [Kim04, NR99], juose galima rasti konkrečios informacijos apie tam tikrų TSP elementų turinį ir taikymą.

2.2. Bendra TSP ciklo struktūra (DEV scenarijus)

Komandinio programų kūrimo procesas yra iteracinis. Projekto iteracijų (ciklų) skaičius yra apibrėžiamas pirmo startavimo metu. Pirmo ciklo struktūra yra parodyta 2 lentelėje.

2 lentelė. TSP pirmo ciklo struktūra [McA00, Hum99, HO99]

Scenarijaus tikslas	Apibrėžti PĮ kūrimo proceso veiklas.
Pradžios kriterijai	<ul style="list-style-type: none"> • Projekto sutartis yra pasirašyta; • Projekto komanda yra suformuota; • Visi komandos nariai yra apmokyti PSP; • TSP instruktorius yra pasiekiamas ir pasiruošęs dirbti su komanda. Jeigu komandos nariai yra patyrę TSP, instruktoriaus dalyvavimas nėra būtinas.

Bendra informacija	<ul style="list-style-type: none"> • Paprastai kūrimo procesas susideda iš kelių ciklų; • Labai mažo programinio produkto kūrimui gali užtekti vieno ciklo.
Scenarijus	Pavadinimas
LAU	Startavimo procesas
LAU1	Projekto apžvalga
LAU2	Tikslai ir rolės
LAU3	Strategija, procesas, palaikymas
LAU4	Bendras projekto planas
LAU5	Kokybės planas
LAU6	Detalus einamojo ciklo planas
LAU7	Rizikų įvertinimas
LAU8	Pasiruošimas susitikimui su vadovybe
LAU9	Susitikimas su vadovybe
LAUPM	Startavimo aptarimas
REQ	Reikalavimai
HLD	Aukšto lygio projektavimas
IMP	Realizavimas
IMP6	Modulių testavimas ir testų kūrimas
TEST	Testavimas
TEST1	Produkto surinkimas (angl. build)
TEST2	Integravimo testavimas
TEST3	Sistemos testavimas
TESTD	Testavimo metu aptiktų defektų šalinimas
PM	Projekto (ciklo) aptarimas
Pabaigos kriterijai	<ul style="list-style-type: none"> • Sukurtas produktas ar produkto dalis bei vartotojo dokumentacija; • Sukurta ir atnaujinta projekto dokumentacija; • Sukurtos ciklo ataskaitos.

Visų kitų projekto ciklų struktūra yra analogiška pirmam ciklui, bet vietoje LAU yra taikomas REL (pakartotinio startavimo) scenarijus. REL scenarijus yra identiškas LAU scenarijui, išskyrus pirmą startavimo susitikimą. Pakartotinio startavimo pirmo susitikimo metu yra aptariami projekto būsenos ir valdymo pasikeitimai, o šio susitikimo veiklos yra aprašytos REL1 scenarijuje.

Nors standartinis TSP ciklas apima 6 pagrindinius scenarijus, realybėje projektas gali neapimti, pavyzdžiui, reikalavimų stadijos (tuo atveju kai reikalavimų specifikacija jau yra sudaryta) arba, jeigu projekto tikslas yra sukurti detalų projektą ir neapima jokių realizavimo darbų, gali būti sudarytas tik iš startavimo, projektavimo ir aptarimo scenarijų. Taipogi, yra galimos ir kitokios scenarijų kombinacijos. Konkreti pasirinkta scenarijų kombinacija turi būti apibrėžta projekto

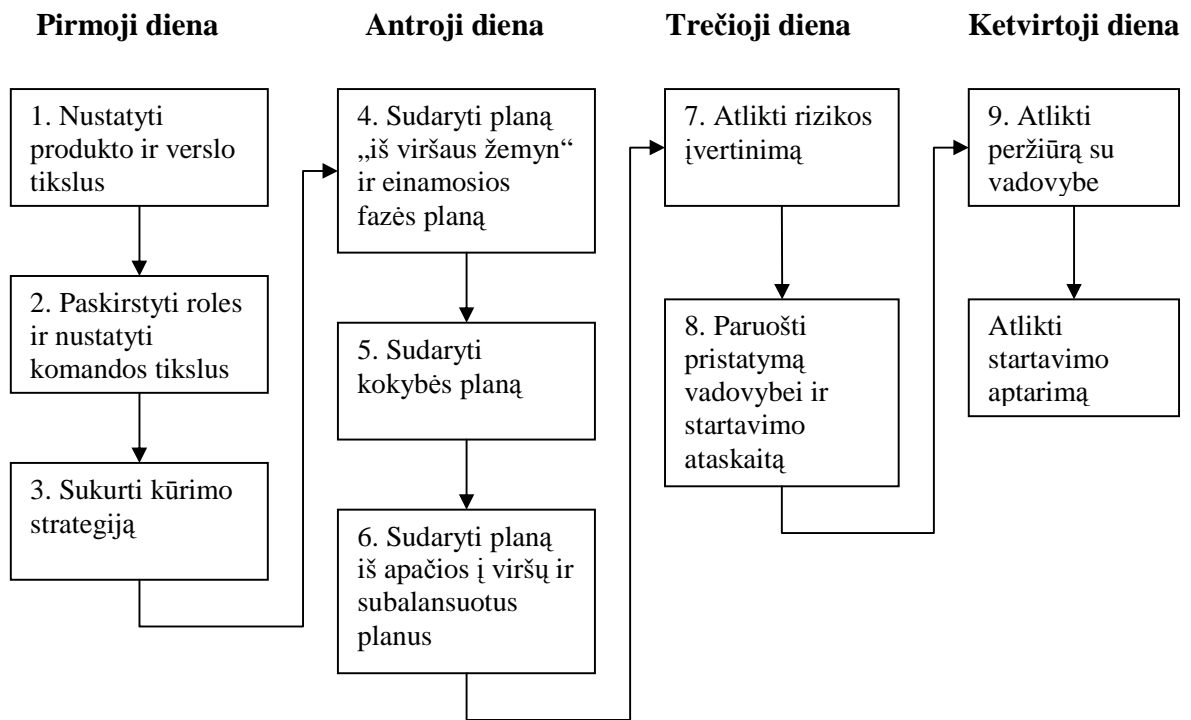
startavimo metu. Projektą gali sudaryti nuo 3 iki 6 scenarijų: LAU ir PM yra būtini, be šių dviejų scenarijų turi būti dar bent vienas scenarijus (REQ, HLD, IMP arba TEST).

TSP modelio rekonstrukcijos tikslas – atkurti visus TSP scenarijus. Pirmiausia, yra siekiama atkurti scenarijus, kurie įeina į DEV scenarijų (žr., 2 lentelę). Šie scenarijai sudaro TSP proceso pagrindą. Neaprašius jų, visi kiti scenarijai negalės sudaryti darnios visumos, t.y. bus gautas tik atskirų, nenuoseklių scenarijų rinkinys. Šio darbo tikslas yra rekonstruoti visą modelį, kurį būtų galima naudoti praktikoje.

2.3. TSP startavimo (LAU) scenarijus

Startavimo procesas apima strategijos sudarymą, planavimą bei rizikos valdymą. Prieš pradėdant startavimą projekto komanda jau turi būti suformuota. Startavimo procese dalyvauja visi komandos nariai. Kaip ir bet kuri kita projekto dalis startavimas yra planuojamas ir sekamas. Jo metu komanda susipažįsta su projekto tikslais (vadovybės, užsakovo), darbais, kuriuos reikės atlikti, nustato komandos ir komandos narių tikslus, sudaro bendrą projekto planą ir detalius ciklo planus: bendrą visai komandai ir asmeninius planus kiekvienam iš komandos narių.

Startavimas yra atliekamas pirmo projekto ciklo pradžioje, pakartotinis startavimas yra atliekamas kiekvieno sekančio ciklo pradžioje. Pakartotinis startavimas gali būti atliktas ir tuo atveju, jei projekto metu įvyksta reikšmingi pasikeitimai. Jeigu projektas yra trumpalaikis, tai jis gali būti vykdomas vienu ciklu. TSP yra siūloma 3 arba 4 dienų startavimo procedūra. 9 startavimo susitikimas ir startavimo aptarimas gali vykti trečios dienos vakare arba ketvirtos dienos ryte. Keturių dienų startavimo procedūros planas yra parodytas 4 paveikslėlyje [DM03]:



4 pav. TSP startavimo procedūra [DM03]

Projektavimo strategijos gali būti skirstomos į:

- 1) iš viršaus žemyn – kai einama nuo bendriausio abstrakcijos lygio (esminių produkto užduočių) ir leidžiamasi prie konkrečių modulių (kurios reikės įgyvendinti);
- 2) iš apačios į viršų – atvirkščiai, pradedama nuo modulių ir einama link visos sistemos vaizdo (projekto).

Komandos sudaromi planai yra skirstomi į 2 grupes pagal tai, kokia projektavimo strategija komanda rėmėsi, sudarant tą planą. T.y. planai, kaip ir jų pagrinde esančios projektavimo strategijos, yra skirstomi į planus „iš viršaus žemyn“ ir „iš apačios į viršų“. Pirmas metodas yra skirtas daugiau sudaryti bendrą vaizdą, ką gi reikės sukurti, ir bendrą sistemos struktūrą (4 startavimo susitikimas), antras metodas yra daug detalesnis, leidžia sudaryti tikslesnius planus bei darbų sąrašą (6 startavimo susitikimas).

Startavimo scenarijus (LAU) yra vienintelis viešai paskelbtas TSP scenarijus, jis yra pateiktas 3 lentelėje. Deja, detalūs startavimo susitikimų scenarijai (LAU1,..., LAU9) nėra paviešinti, šiuo metu yra žinomas tik bendras šių scenarijų apibūdinimas.

3 lentelė. TSP startavimo (LAU) scenarijus [Hum00]

Scenarijaus paskirtis		Apibrėžti startavimo proceso veiklas.
Pradžios kriterijai		<ul style="list-style-type: none"> • Pasiruošimo startavimui darbai yra atlikti (PREPL, PREPT kontroliniai sąrašai, žr. 2.3.1 skyrių); • Susitarta dėl vadovybės ir užsakovo atstovų dalyvavimo susitikimuose (1 ir 9). Jeigu projektas neturi konkretaus užsakovo, vietoje jo susitikimuose dalyvauja pardavimų skyriaus atstovai; • Visi komandos nariai ir komandos lyderis yra įsipareigoję dalyvauti visuose startavimo susitikimuose ir startavimo aptarime; • Yra įgaliotas TSP instruktorius, kuris vadovaus komandai startavimo procese.
Bendra informacija		<ul style="list-style-type: none"> • 1, 2 ir 3 susitikimai vyksta pirmą startavimo dieną; • 4, 5 ir 6 susitikimai vyksta antrą startavimo dieną; • 7 ir 8 susitikimai vyksta trečią startavimo dieną; • 9 susitikimas ir startavimo aptarimas vyksta trečios dienos vakare arba ketvirtos dienos ryte; • Visi startavimo susitikimai yra dokumentuojami (TSP formos, projekto dokumentacija).
Nr.	Veikla	Aprašymas
1	Nustatyti produkto ir verslo tikslus	Pravesti pirmą startavimo susitikimą (naudoti LAU1 scenarijų). <ul style="list-style-type: none"> • Pristatyti vadovybei ir komandai visus projekto komandos narius; • Peržvelgti startavimo procesą; • Aptarti projekto tikslus su vadovybe ir užduoti klausimus.
2	Komandos tikslai ir rolės	Pravesti antrą startavimo susitikimą (naudoti LAU2 scenarijų). <ul style="list-style-type: none"> • Pasiskirstyti roles; • Nustatyti ir dokumentuoti komandos tikslus.
3	Projekto strategija ir palaikymas	Pravesti 3 startavimo susitikimą (naudoti LAU3 scenarijų). <ul style="list-style-type: none"> • Sukurti koncepcinį sistemos modelį; • Nustatyti kūrimo strategiją ir produktus, kurie turi būti sukurti; • Apibrėžti kūrimo procesą, kuris bus naudojamas; • Sukurti proceso ir palaikymo planus.
4	Bendras planas	Pravesti 4 startavimo susitikimą (naudoti LAU4 scenarijų). <ul style="list-style-type: none"> • Atlikti dydžio įvertinimus ir sudaryti projekto planą.
5	Kokybės planas	Pravesti 5 startavimo susitikimą (naudoti LAU5 scenarijų). <ul style="list-style-type: none"> • Sudaryti kokybės planą.
6	Subalansuotas planas	Pravesti 6 startavimo susitikimą (naudoti LAU6 scenarijų). <ul style="list-style-type: none"> • Priskirti darbus komandos nariams; • Kiekvienam komandos nariui sudaryti individualų einamojo ciklo planą „iš apačios į viršų“;

		<ul style="list-style-type: none"> • Subalansuoti komandos ir komandos narių planus.
7	Projekto rizikos analizė	Pravesti 7 startavimo susitikimą (naudoti LAU7 scenarijų). <ul style="list-style-type: none"> • Identifikuoti ir įvertinti projekto rizikas; • Nustatyti rizikų vertinimo kontrolinius taškus ir atsakomybes; • Pasiūlyti neatidėliotinų didelės svarbos (poveikio) rizikų sumažinimo veiksmus.
8	Startavimo ataskaitos paruošimas	Pravesti 8 startavimo susitikimą (naudoti LAU8 scenarijų). <ul style="list-style-type: none"> • Paruošti startavimo ataskaitą vadovybei.
9	Vadovybės peržiūra	Pravesti 9 startavimo susitikimą (naudoti LAU9 scenarijų). <ul style="list-style-type: none"> • Peržiūrėti startavimo veiklas ir projekto planus su vadovybe; • Aptarti projekto rizikas, atsakomybes ir numatytus veiksmus.
PM	Startavimo aptarimas	Atlikti startavimo aptarimą (naudoti LAUPM scenarijų). <ul style="list-style-type: none"> • Aptarti pasiruošimą kas savaitiniams susitikimams; • Surinkti startavimo duomenis ir sudaryti ataskaitą; • Papildyti projekto dokumentaciją; • Įvertinti startavimo procesą ir užpildyti proceso gerinimo pasiūlymų (PIP) formas.
Pabaigos kriterijai		<ul style="list-style-type: none"> • Sudaryti komandos ir komandos narių planai; • Paskirstytos rolės, nustatyti tikslai, procesai ir atsakomybės; • Komandos planas yra suderintas su vadovybe; • Startavimo proceso duomenys yra įtraukti į projekto dokumentaciją.

Pirmame susitikime komanda, komandos lyderis ir startavimo instruktorius susitinka su vadovybe ir užsakovo arba pardavimų skyriaus atstovais. Šio susitikimo tikslas yra supažindinti komandą su produkto ir vadovybės tikslais, susijusiais su vykdomu projektu. Susitikimo metu yra aptariami keli verslo tikslai, kuriame produkto poreikis (kodėl reikia jį kurti?), pagrindines savybės bei kitą susijusią informaciją. Šis aptarimas apima projekto pateikimo terminus, kainą ir kokybės tikslus. Produkto ir vadovybės tikslų aptarimas padeda komandai nustatyti, kaip projektas prisidės prie įmonės verslo tikslų pasiekimo, minimalius projekto reikalavimus.

Susitikimuose nuo 2 iki 8 dalyvauja komanda ir startavimo instruktorius. Šių susitikimų tikslas yra sukurti sąlygas efektyviam komandos darbui.

Antro susitikimo metu projekto komanda peržiūri verslo ir produkto tikslus, pristatytus pirmame susitikime, ir jų pagrindu formuluoja rinkinį pamatuojamų komandos ir komandos narių tikslų. Po to komandos nariai pasiskirsto rolėmis, tam yra skirtos specialios formos ROLE. Jeigu yra

prasminga, gali būti sukurtos ir priskirtos papildomos rolės (pvz., saugumo vadovas ir pan.). Kiekvienas komandos narys pasirenka bent po vieną rolę.

Trečio susitikimo metu komanda apibrėžia bendrą projekto strategiją [HO99]:

1. Konceptinį kuriamos sistemos modelį, kuris apibrėžia, kas turi būti padaryta (sukurta) projekto metu. Konceptinis modelis apima minimalaus/idealaus rezultato, kuris tenkintų vadovybės reikalavimus, pagrindinių produkto dalių aprašymą ir pirminius šių dalių dydžio įvertinimus.
2. Kūrimo strategiją, kuri apibrėžia projekto ciklų skaičių, pagrindinius kontrolinius taškus, pagrindinių versijų turinį, prototipus, kuriuos reikės sukurti, integravimo ir testavimo strategijas. Taipogi kūrimo strategija apima produkto dalių dydžio ir pastangų, reikalingų jiems sukurti, įvertinimų patikslinimą.
3. Patikslina (adaptuoja) TSP procesą, kuris bus naudojamas projekto metu. Komanda apibrėžia kaip bus vykdomas projektas, kokie scenarijai, formos ir standartai bus naudojami. Tai gali būti TSP elementai arba organizacijos proceso elementai.
4. Palaikymo planą, kuriame yra apibrėžiama, kokia įranga ir įrankiai bus reikalingi komandai vykdant projektą, už šio plano sekimą ir vykdymo užtikrinimą yra atsakingas palaikymo vadovas.

Ketvirto susitikimo tikslas – bendro kūrimo plano sudarymas. Šis planas leidžia komandai išlaikyti dėmesį ties galutiniu projekto tikslu ir yra pagrindas individualiems planams sudaryti. TSP numato plano „iš viršaus žemyn“ sudarymą: komanda nustato pagrindines užduotis, jas detalizuoja į smulkesnes, palygina su panašiais, anksčiau vykdytais, projektais, nustato istoriniais duomenimis pagrįstus užduočių vykdymo laikus. Tada komanda sudaro visos komandos užduočių ir uždirbtos vertės planus. Planai yra derinami su bendrais projekto poreikiais (apribojimais): laikas, resursai ir t.t.. Taipogi yra nagrinėjamos alternatyvios kūrimo strategijos. Sudarant bendrą planą, komanda nusprendžia, ar produktas gali būti kuriamas etapais (ciklais), arba „prieaugiais“, jei taip, tai kiekvienai fazei yra nustatomi reikalavimai ir „kūrimo užšaldymo“ datos – datos, po kurių reikalavimai konkrečiam sistemos prieaugiui nebegali kisti, projektavimo specifikacijų paruošimo galutiniai terminai ir datos, po kurių negali būti keičiamas kodas. Toliau, vadovaujant planavimo vadovui, komanda sudaro bendrą projekto tvarkaraštį. Kiekvienam ciklui yra nustatoma jo trukmė ir pagrindinės užduotys.

Pirmas bendro projekto tvarkaraščio sudarymo žingsnis – sukurti užduočių sąrašą, užduotys turi būti išskiriamos atsižvelgiant į TSP kūrimo procesą, startavimo procesą ir vadovų roles. Toliau

komanda atlieka numatytų užduočių dydžio (kodo eilutėmis, teksto eilutėmis ir pan.) ir laiko sąnaudų įvertinimus. Kitas žingsnis – sukurti tvarkaraštį, kuris remiasi bendru valandų skaičiumi, kuris yra reikalingas atlikti visas numatytas užduotis. Remiantis užduočių sąrašu ir tvarkaraščiu, komanda sudaro uždirbtos vertės planą, kuris apima planuojama kiekvienos užduoties planuojamą vertę (priklausomai nuo jai atlikti skiriamų valandų skaičiaus), planuojamas datas kiekvienai užduočiai ir planuojamą vertę, kuri turi būti „uždirbta“ kiekvieną savaitę, remiantis atliktomis užduotimis [HO99].

Penkto susitikimo metu komanda sudaro kokybės planą, kuris turi atitikti komandos kokybės tikslus. Paprastai kokybės tikslai yra išreiškiami numatant padarytų ir ištaisytų kiekvienoje fazėje defektų skaičių ir defektų tankį galutiniam produkte. Komanda turi užtikrinti, kad į projekto planą būtų įtrauktos atitinkamos veiklos, skirtos pasiekti užsibrėžtus kokybės tikslus. Kokybės planas suteikia galimybę sekti darbų ir produkto kokybę. TSP kokybės plano šablonas yra pateiktas 2 priede.

Šeštas susitikimas yra skirtas detalių einamojo ciklo planų sudarymui ir darbų perskirstymui taip, kad kiekvienam komandos nariui tektų įveikiamas darbų krūvis. Pirmiausia, yra pasidalinami einamojo ciklo darbai, kiekvienas inžinierius sudaro savo asmeninį planą, remiantis savo asmeniniais istoriniais duomenimis. Kai asmeniniai planai yra sudaryti, kiekvienas komandos narys trumpai pristato savo planą. Komanda svarsto, ar planas yra pilnas, ar pakankamai detalus, ar užduotys yra suderinamos su bendru komandos planu, ar planas yra pagrįstas ir realus (ar nėra užsibrėžti neįveikiami tikslai). Toliau, jeigu komandos narių darbai nėra subalansuoti, seka darbų perskirstymas ir iš individualių planų yra sudaromas visos komandos detalus ciklo planas.

Septinto susitikimo metu komanda atlieka rizikų įvertinimą. Yra identifikuojamos pagrindinės projekto rizikos, įvertinama jų atsiradimo tikimybė ir įtaka projektui. Kiekvienai projektui rizikai, su didele atsiradimo galimybe ir/arba įtaka, yra paskiriamas žmogus, kuris yra atsakingas už tos rizikos stebėjimą, nurodomas laikas, kada ji tikriausiai įvyks, triggerį – sąlygą, žyminčią, kada reikia pradėti koregavimo veiksmus, ir pačius koregavimo veiksmus.

Aštuntas susitikimas yra skirtas paruošti sudaryto plano pristatymą vadovybei ir užsakovo arba pardavimų skyriaus atstovams. Pristatymas turi apimti komandos narių ir rolių sąrašą, startavimo proceso apžvalgą, komandos tikslus, resursų ir užduočių tvarkaraščius, alternatyvias kūrimo strategijas, kokybės planą, identifikuotų rizikų sąrašą ir numatomus jų poveikio sumažinimo veiksmus.

Devintas susitikimas – susitikimas su vadovybės ir užsakovo arba pardavimų skyriaus atstovais. Šio susitikimo metu komanda pristato projekto planą ir paaiškina, kaip jis buvo sudarytas. Jeigu sukurtas planas netenkina vadovybės arba užsakovo tikslų, komanda turi pristatyti alternatyvius planus, parodydama, ką galima padaryti pridėjus papildomų resursų ar pakeitus reikalavimus. TSP startavimo pabaigoje komanda ir vadovybė turi sutarti dėl projekto plano, t.y. turi būti gautas vadovybės pritarimas.

Aptarimas - paskutinis startavimo susitikimas, kuriam vadovauja TSP instruktorius. Aptarimo metu yra aptariami pasiruošimo kassavaitiniams susitikimams darbai, pildomos proceso gerinimo pasiūlymų formos, atnaujinama ir peržiūrima projekto dokumentacija.

2.3.1. Pasiruošimas startavimui

Prieš pradėdant startavimą, TSP instruktorius turi atlikti eilę darbų, kurie yra surašyti pasiruošimo startavimui kontroliniame sąraše (PREPL). Šių darbų paskirtis yra užtikrinti, kad projekto komanda yra suformuota, jos nariai yra tinkamai apmokyti, yra sudarytas startavimo tvarkaraštis, susitarta dėl vadovybės ir užsakovo (arba pardavimų skyriaus) atstovų dalyvavimo ir pan.

4 lentelė. PREPL kontrolinis sąrašas [Hum05]

Savaitė iki startavimo	Žingsnis	Aprašymas	Atlikta
6	Projekto identifikavimas	Buvo nuspręsta vykdyti TSP projektą	
6	Atsakingas asmuo	Projektui yra paskiriamas TSP instruktorius	
4	Komanda suformuota	Visi komandos nariai buvo identifikuoti (parinkti)	
2	Apmokymas	Visi komandos nariai yra apmokyti PSP ir TSP.	
2	Dalyvavimas	<ul style="list-style-type: none"> • Visi komandos nariai gali ir sutiko dalyvauti visuose startavimo susitikimuose; • Vadovybės ir užsakovo (arba pardavimų skyriaus) atstovai sutiko dalyvauti 1 ir 9 startavimo susitikimuose 	
2	Tvarkaraštis	Yra sudaromas visų startavimo susitikimų tvarkaraštis	
2	Pasiruošimas 1 ir 9 susitikimams	<ul style="list-style-type: none"> • Yra užrezervuotos patalpos susitikimams; • Projektorius ir ekranas; 	
2	Pasiruošimas susitikimams nuo 2 iki 8	<ul style="list-style-type: none"> • Yra užrezervuotos patalpos susitikimams; • Vienas ar daugiau personalinių kompiuterių su tinkama programine įranga (pvz., MS Word, Excel) • Projektorius ir ekranas; • Lenta ir kreida. 	

2	Proceso apžvalga	Visi komandos nariai gali ir sutiko dalyvauti TSP proceso apžvalgos susitikime, kuris vyksta po startavimo aptarimo.	
2	Vadovybės informavimas	Vadovybės atstovai yra supažindinti su startavimo procesu, jo svarba ir yra informuoti apie pasiruošimo darbus, kuriuos jie turės atlikti (verslo tikslų formulavimas, kitos reikalingos informacijos paruošimas).	
2	Užsakovo informavimas	Užsakovo (pardavimų skyriaus) atstovai yra supažindinti su startavimo procesu, jo svarba ir yra informuoti apie pasiruošimo darbus, kuriuos jie turės atlikti (projekto pristatymas, projekto tikslai, pagrindiniai reikalavimai ir t.t.).	
2	Maitinimas	Visas dienas, kol truks startavimas, komandos nariai turi būti maitinami, jiems turi būti skirtos poilsio patalpos ir kiti reikalingi patogumai.	

2.3.2. Scenarijus LAU1

Pirmame susitikime dalyvauja komanda, startavimo instruktorius, vadovybės ir užsakovo arba pardavimų skyriaus atstovai. Šio susitikimo tikslas yra supažindinti komandą su produkto, užsakovo ir vadovybės tikslais, susijusiais su vykdomu projektu. Kartu tai yra gera proga komandos motyvavimui.

5 lentelė. TSP LAU1 scenarijus

Scenarijaus paskirtis	Apibrėžti susipažinimo su reikalavimais produktui ir vadovybės keliamais tikslais veiklas.
Pradžios kriterijai	<ul style="list-style-type: none"> • Pasiruošimo startavimui darbai yra atlikti (PREPL, PREPR kontroliniai sąrašai); • Susitarta dėl vadovybės ir užsakovo atstovų dalyvavimo pirmame startavimo susitikime;
Bendra informacija	<ul style="list-style-type: none"> • Susitikime dalyvauja visi komandos nariai ir vadovybės bei užsakovo (arba pardavimų skyriaus) atstovai; • Susitikimui vadovauja TSP instruktorius.
Veikla	Aprašymas
Komandos pristatymas	Komandos lyderis trumpai pristato savo komandos narius.
Reikalavimų produktui pristatymas	<ul style="list-style-type: none"> • Užsakovo ar pardavimų skyriaus atstovas pristato: <ul style="list-style-type: none"> ○ projekto tikslus; ○ pagrindines produkto savybes; ○ svarbius reikalavimus ir /ar standartus, kuriuos turi atitikti kuriamas produktas; ○ užsakovo organizaciją; ○ kitą aktualią informaciją;

Verslo tikslų pristatymas	<ul style="list-style-type: none"> • Vadovybė pristato verslo tikslus, susijusius su projektu: <ul style="list-style-type: none"> ○ vadovybės lūkesčiais; ○ idealiu rezultatu; ○ minimaliu priimtinu rezultatu; ○ pristatymas apima kainą, laiką ir kokybės tikslus;
Reikalavimų ir tikslų aptarimas	<ul style="list-style-type: none"> • Vadovybės ir užsakovo atstovai bei komandos nariai aptaria verslo tikslus ir reikalavimus produktui: <ul style="list-style-type: none"> ○ aptariama, kaip projektas prisidės prie verslo tikslų realizavimo; ○ nustatomi minimalūs reikalavimai; ○ jeigu reikia, randami kompromisai ir t.t.;
Pabaigos kriterijai	<ul style="list-style-type: none"> • Identifikuoti ir dokumentuoti pagrindiniai projekto ir produkto reikalavimai, verslo tikslai.

Susitikimo pradžioje komandos lyderis trumpai pristato savo komandos narius. Šis pristatymas yra skirtas tiek vadovybės ir užsakovo atstovams, tiek patiems komandos nariams. Užsakovo atstovai turi suprasti, kad jų projektą vykdys kompetentingi specialistai. Komandos nariams šis pristatymas suteikia informacijos apie tai, kas iš jų kolegų ką žino/sugeba ir į ką galima kreiptis pagalbos ir konsultacijų.

Toliau užsakovo atstovai trumpai pristato užsakovo įmonę, projekto tikslus, laukiamas kuriamo produkto savybes, reikalavimus ir kitą aktualią informaciją.

Po užsakovo atstovų pristatymo savo lūkesčius išsako vadovybės atstovai. Šis pristatymas apima verslo tikslus, ko yra tikimasi iš projekto, idealų ir minimalų priimtina rezultatą, numatoma projekto kainą, trukmę ir kokybės tikslus.

Galiausiai komandos nariai gali užduoti savo klausimus vadovybės ir užsakovo atstovams, aptarti su jais visus iškilusius neaiškumus.

2.3.3. Scenarijus LAU2

Antro susitikimo metu projekto komanda pasiskirsto rolėmis, kurios yra aprašytos 1.5 skyriuje „TSP projekto komanda“, peržiūri verslo ir produkto tikslus, pristatytus pirmame susitikime, ir jų pagrindu formuluoja rinkinį pamatuojamų komandos ir komandos narių tikslų.

6 lentelė. TSP LAU2 scenarijus

Scenarijaus paskirtis	Apibrėžti komandos veiklas paskirstant roles ir formuluojant komandos tikslus.
Pradžios kriterijai	<ul style="list-style-type: none"> • Identifikuoti ir dokumentuoti pagrindiniai produkto reikalavimai ir verslo tikslai.

Bendra informacija	<ul style="list-style-type: none"> • Susitikime dalyvauja visi komandos nariai; • Šio susitikimo metu komandai vadovauja komandos lyderis arba TSP instruktorius.
Veikla	Aprašymas
Paskirstyti roles	<ul style="list-style-type: none"> • Kiekvienas komandos narys pasirenka sau rolę. Yra pasirenkamos pagrindinė ir alternatyvi rolės; • Rolės yra paskirstomos tarp komandos narių. Jeigu yra keli pretendentai į vieną rolę, tai rolę yra paskiriama tam, kuris pasirinko šią rolę kaip pagrindinę, arba komandos nariai tiesiog susitaria tarpusavyje.
Suformuluoti komandos tikslus	<ul style="list-style-type: none"> • Komandos ir rolių tikslai yra formuluojami remiantis dokumentuotais verslo tikslais ir reikalavimais produktui; • Tikslai turi būti kiekybiniai ir išmatuojami; • Formuluojami komandos narių tikslai.
Pabaigos kriterijai	<ul style="list-style-type: none"> • Rolės yra paskirstytos tarp komandos narių; • Komandos ir komandos narių tikslai yra suformuluoti ir dokumentuoti.

Susitikimo pradžioje komandos nariai pasiskirsto rolėmis, tam yra naudojamos ROLE formos. Kiekvienas komandos narys pasirenka sau pagrindinę (labiausiai norimą) ir alternatyvią roles. Komandos lyderis surenka visas ROLE formas, ir paskirsto roles tokiu būdu: jeigu tik vienas komandos narys nurodė tam tikrą rolę kaip pagrindinę, tai jam ir yra priskiriama ši rolė. Jeigu keli komandos nariai nurodė tą pačią rolę, pirmumo teisę turi tas narys, kuris nurodė šią rolę kaip pagrindinę. Visais kitais atvejais komandos nariai turi tiesiog susitarti tarpusavyje, kam kokia rolę atiteks. Kiekvienas komandos narys turi pasirinkti bent po vieną rolę. Jei komandoje yra daugiau negu 8 žmonės, viena rolė gali būti padalinta tarp kelių žmonių, t.y. du žmonės atlieka vieną rolę, arba vienas žmogus tampa vadovu, o kitas jo asistentu. Mažesnėse komandose nariai pasirenka po kelias roles. Komandos lyderis paprastai neprisiima jokių papildomų rolių.

Pasiskirstę rolėmis komandos nariai peržiūri iškeltus verslo tikslus ir produkto reikalavimus. Jų pagrindu yra formuluojami komandos tikslai. Šie tikslai turi būti kiekybiniai ir išmatuojami. Galiausiai yra formuluojami komandos narių tikslai. TSP kiekvienai rolei numato tam tikrus tikslus, tačiau komanda gali šiuos tikslus pakoreguoti arba papildyti.

2.3.4. Scenarijus LAU3

Trečio susitikimo metu komanda apibrėžia bendrą projekto strategiją. Komandos nariai sukuria koncepcinį modelį, apibrėžia kūrimo strategiją, procesą, kuris bus naudojamas projekto metu, bei sudaro palaikymo planą.

7 lentelė. TSP LAU3 scenarijus

Scenarijaus paskirtis	Apibrėžti komandos veiklas sudarant bendrą produkto kūrimo strategiją.
Pradžios kriterijai	<ul style="list-style-type: none"> • Rolės yra paskirstytos tarp komandos narių; • Komandos ir komandos narių tikslai yra suformuluoti ir dokumentuoti;
Bendra informacija	<ul style="list-style-type: none"> • Sudarant bendrą produkto kūrimo strategiją dalyvauja visi komandos nariai; • Šio susitikimo metu komandai vadovauja komandos lyderis; • Kuriant koncepcinį modelį ir kūrimo strategiją komandai vadovauja kūrimo vadovas.
Veikla	Aprašymas
Sukurti koncepcinį modelį	<ul style="list-style-type: none"> • Koncepcinis modelis turi apimti: <ul style="list-style-type: none"> ○ kas turi būti realizuota, norint įgyvendinti minimalius/planuojamus produkto reikalavimus; ○ pagrindinės produkto dalys (moduliai);
Sudaryti kūrimo strategiją	<ul style="list-style-type: none"> • Projekto ciklą skaičiaus ir jų trukmės nustatymas; • Pagrindinių kontrolinių taškų identifikavimas; • Pagrindinių versijų turinio aprašymas; • Prototipų, kuriuos reikės sukurti, identifikavimas; • Integravimo ir testavimo strategijų aprašymas; • Kūrimo strategijos dokumentavimas apima STRAT formos pildymą.
Preliminarus laiko ir dydžio vertinimas	<ul style="list-style-type: none"> • Šiame etape yra atliekamas grubus dydžio ir pastangų vertinimas: <ul style="list-style-type: none"> ○ kiekvienam projekto ciklui; ○ kiekvienos identifikuotos koncepcinio modelio dalies įgyvendinimui; • Vertinimas turėtų remtis istoriniais duomenimis ir/ar geriausiu spėjimu (optimistinis/labiausiai tikėtinas/pesimistinis vertinimai).
Apibrėžti kūrimo procesą	<ul style="list-style-type: none"> • Kūrimo procesas nusako, <i>kaip</i> bus kuriamas produktas; • Ši veikla yra skirta aprašyti visiems proceso elementams, kurie bus naudojami projekto metu; • Naudojamų proceso elementų rinkinys susideda iš: <ul style="list-style-type: none"> ○ TSP procesų elementų (pvz., scenarijai, formos); ○ organizacijos proceso elementų (pvz., standartai); • Komanda įvertina trūkstamų proceso elementų dydį ir laiką, kurio reikės šiems elementams sukurti, paskiria atsakingus už šių elementų sukūrimą asmenis. Šios veiklos turi būti įtrauktos į projekto planą;
Sudaryti palaikymo planą	<ul style="list-style-type: none"> • Identifikuoti, kokie įrankiai yra reikalingi: <ul style="list-style-type: none"> ○ reikalavimų sudarymo ir projektavimo pagalbinės priemonės; ○ redaktoriai, kompiliatoriai, programos derinimo priemonės; ○ testavimo įrankiai;

	<ul style="list-style-type: none"> • Komanda nustato savo poreikius ir sudaro reikalingų priemonių įsigijimo/parūpinimo planą. Už šio plano vykdymą yra atsakingas palaikymo vadovas.
Pabaigos kriterijai	<ul style="list-style-type: none"> • Sukurtas koncepcinis modelis; • Sudaryta ir dokumentuota kūrimo strategija; • Atliktas preliminarus dydžio ir laiko vertinimas; • Užpildyta STRAT forma; • Apibrėžtas kūrimo procesas; • Sudarytas palaikymo planas; • Papildyta projekto dokumentacija.

Pirmasis scenarijaus žingsnis – sukurti koncepcinį modelį. Koncepcinis modelis apibrėžia, kas turi būti padaryta (sukurta), kad būtų patenkinti minimalūs vadovybės reikalavimai, bei apima pagrindinių produkto dalių identifikavimą ir tų dalių dydžio įvertinimus.

Antrasis žingsnis – apibrėžti kūrimo strategija. Komanda turi nuspręsti, į kiek ciklų bus padalintas projektas, bei kokios bus šių ciklų trukmės, apibrėžti pagrindinius kontrolinius taškus, pagrindinių versijų turinį, identifikuoti prototipus, kuriuos reikės sukurti, apibrėžti integravimo ir testavimo strategijas. Tuo pačiu, komanda užpildo STRAT formą.

Turint pagrindinių užduočių sąrašą, gautą kuriant koncepcinį modelį, ir apibrėžtą kūrimo strategiją, komanda turi atlikti preliminarų laiko ir pastangų, reikalingų kiekvienam iš projektų ciklų ir kiekvienos iš identifikuotų koncepcinio modelio dalių įgyvendinimui, įvertinimą. Vertinimas turi remtis sukauptais istoriniais duomenimis. Jeigu tokie duomenys dar nėra surinkti, vertinimas turi remtis geriausiu spėjimu. T.y. komanda turi atlikti optimistinį (kiek mažiausiai gali prireikti pastangų), pesimistinį (kiek daugiausiai) ir labiausiai tikėtiną vertinimus. Geriausiu spėjimu yra laikomas arba labiausiai tikėtinas variantas. Šis vertinimas gali būti daromas tiesiogiai (pvz., ekspertinis vertinimas) arba išskaičiuojamas pagal tam tikrą formulę. Pavyzdžiui, $(0.4X + 0.6Y)/2$, kur X – yra optimistinis, Y – pesimistinis laiko vertinimas.

Sukūrus koncepcinį modelį ir atlikus preliminarų laiko ir pastangų vertinimą, komanda apibrėžė, ką reikia sukurti. Kūrimo procesas nusako, kaip turi būti kuriamas programinis produktas. Komanda turi apibrėžti visus kūrimo proceso elementus, kurie bus naudojami projekto metu. Šie elementai susideda iš TSP procesų elementų (pvz., formos, scenarijai) ir organizacijos ir/ar individualiai komandos naudojamus proceso elementus. Prie organizacijos elementų yra priskiriami:

- organizacijos arba komandos naudojami standartai;
- vidinės darbo taisyklės;
- vidaus organizacijos procedūros;

- ir pan.

Jeigu komanda nusprendžia naudoti proceso elementus, kurie dar nėra sukurti, tai į projekto planą turi būti įtraukti šių elementų sukūrimo darbai. Komanda turi paskirti asmenis, atsakingus už šių elementų sukūrimą, bei įvertinti sukūrimo darbų laiką ir pastangas.

Paskutinis trečio susitikimo žingsnis – palaikymo plano sudarymas. Komanda turi numatyti, kokia įranga ir kokie įrankiai bus reikalingi projekto metu. Paprastai, projekto metu gali būti naudojami redaktoriai, kompiliatoriai, programos derinimo priemonės, reikalavimų sudarymo ir projektavimo, testavimo ir kiti įrankiai. Komanda nustato savo poreikius ir sudaro reikalingų priemonių įsigijimo/parūpinimo planą. Už šio plano vykdymą yra atsakingas palaikymo vadovas.

2.3.5. Scenarijus LAU4

Bendras projekto planas yra sudaromas ketvirto TSP startavimo susitikimo metu. Šis procesas yra aprašytas LAU4 scenarijuje. Prieš pradėdant šį scenarijų, komandos nariai jau turi apibrėžti projekto ir komandos tikslus bei pasiskirstyti rolėmis. Pagrindinį vaidmenį šio susitikimo metu atlieka planavimo vadovas. Bendras projekto planas leidžia komandai išlaikyti dėmesį ties galutiniu projekto tikslu ir yra pagrindas individualiems planams sudaryti.

8 lentelė. TSP LAU4 scenarijus

Scenarijaus paskirtis	Apibrėžti komandos veiklas sudarant bendrą projekto planą.
Pradžios kriterijai	<ul style="list-style-type: none"> • Sudaryta projekto strategija, apibrėžtas kūrimo procesas; • Rolės yra paskirstytos, projekto ir komandos tikslai apibrėžti; • Yra sudaryta kūrimo strategija, apibrėžti kūrimo ir palaikymo procesai; • TSP instruktorius dalyvauja susitikime.
Bendra informacija	<ul style="list-style-type: none"> • Sudarant bendrą projekto planą dalyvauja visi komandos nariai; • Šio susitikimo metu komandai vadovauja planavimo vadovas.
Veikla	Aprašymas
Užduočių sąrašo sudarymas	<ul style="list-style-type: none"> • Sudaryti visų užduočių (darbų) sąrašą; • Nustatyti planuojamą (pagrįstą) darbų trukmę; • Priskirti planuojamus projekto darbus pagal ciklus; • Užpildyti TASK formą;
Užduočių sąrašo peržiūra	<ul style="list-style-type: none"> • Visa komanda peržiūri sudarytą darbų sąrašą; • Jeigu randami netikslumai, jie pašalinami;
Projekto darbų tvarkaraščio sudarymas	<ul style="list-style-type: none"> • Nustatyti, kiek laiko projekto vykdymo metu kiekvienas iš komandos narių galės skirti projekto darbams; • Remiantis gautais duomenimis, sudaryti projekto tvarkaraštį;

	<ul style="list-style-type: none"> • Sukurti uždirbtos vertės planą; • Užpildyti TASK ir SCHEDULE formas;
Projekto darbų tvarkaraščio peržiūra	<ul style="list-style-type: none"> • Visa komanda peržiūri sudarytą projekto tvarkaraštį; • Jeigu randami netikslumai, jie pašalinami.
Pabaigos kriterijai	<ul style="list-style-type: none"> • Sukurtas bendras projekto planas, užpildytos TASK ir SCHEDULE formos; • Papildyta projekto dokumentacija; • Visi komandos nariai pritaria sukurtam projekto planui.

Jeigu komandai tai yra pirmas TSP projektas, tai susitikimo pradžioje planavimo vadovas ar TSP instruktorius turi išaiškinti visiems komandos nariams TSP planavimo tikslus ir bendrą struktūrą [Hum99, Hum06A].

Pirmas bendro projekto tvarkaraščio sudarymo žingsnis – sukurti užduočių sąrašą, kuris atitinka TSP kūrimo procesą (kūrimo stadijas, vadovų roles ir t.t.). Šis užduočių sąrašas yra sudaromas „iš viršaus žemyn“: komanda nustato pagrindines projekto užduotis, toliau detalizuoja jas į smulkesnes [HO99].

Kai jau yra gautas visų projekto užduočių sąrašas, komanda įvertina jų dydį ir vykdymo laiką. Nustatant pastarąjį, užduotys yra palyginamos su panašiomis anksčiau vykdytų projektų užduotimis. Istorinių duomenų pagrindu yra nustatomas pagrįstas planuojamų užduočių vykdymo laikas.

Turint visų užduočių sąrašą bei planuojamą jų vykdymo laiką ir apimtį (dydį), komanda nusprendžia, kaip šios užduotys bus padalintos tarp projekto ciklų, patikslina kiekvieno ciklo tikslus, reikalavimus, nustato planuojamą trukmę ir numatomus rezultatus. Sudarant projekto darbų sąrašą yra pildoma TASK forma.

Sudarius darbų sąrašą ir jų vykdymo laiko įvertinimą, komanda kartu dar kartą peržiūri visas užduotis ir patikrina įvertinimų teisingumą.

Sekantis susitikimo žingsnis – projekto tvarkaraščio, kuris remiasi bendru valandų skaičiumi, kuris yra reikalingas atlikti visas numatytas užduotis, sudarymas. Komanda nustato, kiek valandų per savaitę kiekvienas komandos narys galės/turės skirti projektui, kiek dienų ir kada jie turės atostogas ir pan. Remiantis šiais duomenimis yra sudaromas projekto tvarkaraštis, kuriame yra nurodytos visos svarbiausios projekto datos, tuo tarpu ir galutiniai ciklų ir viso projekto terminai.

Atlikus visus minėtus darbus ir naudojant užduočių sąrašą ir tvarkaraštį, komanda sudaro uždirbtos vertės planą, kuris apima planuojama kiekvienos užduoties planuojamą vertę (priklausomai nuo jai atlikti skiriamų valandų skaičiaus), planuojamas datas kiekvienai užduočiai ir planuojamą vertę, kuri turi būti „uždirbta“ kiekvieną savaitę, remiantis atliktomis užduotimis

[HO99]. Sudarius šį planą, komanda užpildo TASK ir SCHEDULE formas bei atnaujina projekto dokumentaciją.

Toliau yra atliekama projekto tvarkaraščio peržiūra. Šioje peržiūroje dalyvauja visi komandos nariai. Visos svarstybinos vietos yra aptariamoms, netikslumai pašalinami. Susitikimo pabaigoje visi komandos nariai turi pritarti sukurtam projekto planui.

2.3.6. Scenarijus LAU5

Kokybės plano sudarymas yra projekto startavimo sudėtinė dalis. Kokybės plano sudarymo veikloms aprašyti yra skirtas atskiras scenarijus LAU5. Prieš pradėdant šį scenarijų, komandos nariai jau nustatė projekto ir komandos tikslus, pasiskirstė rolėmis ir sudarė bendrą projekto planą.

Kone pagrindinį vaidmenį kokybės planavimo procese turi kokybės vadovas. Jis turi gerai išmanyti kiek asmeninio programų kūrimo procesą (PSP), tiek TSP. TSP pateikia kokybės plano šabloną, kuriame yra nurodoma, kokie kokybės rodikliai bus matuojami projekto metu ir planuojamos šių rodyklių reikšmės. Pilnas TSP kokybės planas yra pateiktas 2 priede.

9 lentelė. TSP LAU5 scenarijus

Scenarijaus paskirtis	Apibrėžti komandos veiklas sudarant kokybės planą.
Pradžios kriterijai	<ul style="list-style-type: none"> • Sudarytas bendras projekto planas; • TSP instruktorius dalyvauja susitikime. Jeigu komandos nariai yra patyrę TSP, instruktoriaus dalyvavimas nėra būtinas.
Bendra informacija	<ul style="list-style-type: none"> • Kokybės planas remiasi užsibrėžtais projekto/komandos tikslais; • Sudarant kokybės planą dalyvauja visi komandos nariai; • Šio susitikimo metu komandai vadovauja kokybės vadovas.
Veikla	Aprašymas
Peržiūrėti komandos kokybės tikslus	<ul style="list-style-type: none"> • Peržiūrėti 2 susitikimo metu (LAU2 scenarijus) užsibrėžtus tikslus; • Šie tikslai yra pagrindas nustatant kiekybinius kokybės plano rodiklius; • Jeigu yra būtina (pvz., užsibrėžti tikslai pasirodo yra nerealiūs), pakeisti tuos tikslus.
Kokybės plano sudarymas	<ul style="list-style-type: none"> • Nustatyti leistinus nukrypimus nuo planuojamų kokybės rodiklių reikšmių (pvz., padarytų defektų skaičių ir kiek jų turi būti pašalinta); • Sukurti kokybės planą (SUMQ forma) ir papildyti projekto planavimo suvestinę (SUMP forma).
Kokybės plano peržiūra	<ul style="list-style-type: none"> • Visi komandos nariai peržiūri sukurtą kokybės planą; • Visos svarstybinos vietos yra aptariamoms; • Kokybės planas turi būti realus ir aktualus, t.y. jame esantys kokybės rodikliai turi būti tikrai svarbūs komandai ir projektui.

Pabaigos kriterijai	<ul style="list-style-type: none"> • Sukurtas kokybės planas; • Papildyta projekto dokumentacija; • Visi komandos nariai sutinka su sukurtu kokybės planu.
----------------------------	---

Jeigu komandai tai yra pirmas TSP projektas, tai susitikimo pradžioje kokybės vadovas ar TSP instruktorius turi išaiškinti visiems komandos nariams kokybės plano svarbą ir paskirtį [Hum99, Hum06A].

Tuomet komanda aptaria startavimo scenarijaus antro susitikimo metu iškeltus tikslus. Šio aptarimo tikslas yra išskirti svarbiausius kokybės rodiklius, kuriuos reikės sekti ir matuoti projekto eigoje. Jeigu vieno ar daugiau tikslų neįmanoma pasiekti arba išmatuoti, ar jie yra pasiekti, tai šie tikslai turi būti reformuluojami, arba jų turi būti atsisakyta.

Komanda turi nuspręsti, kokie kokybės rodikliai bus sekami ir parinkti jų matavimus. Kiekvienam pasirinktam rodikliui turi būti dokumentuojamos leistinos reikšmių ribos. Šie duomenys yra fiksuojami kokybės plane (SUMQ forma) ir planavimo suvestinėje (SUMP forma) [Hum00].

Sudarius kokybės planą, visa komanda dar kartą peržiūri gautą planą, visos abejotinos plano vietos turi būti aptartos komandoje, visi netikslumai pašalinti. Galutinis kokybės planas turi būti pasiekiamas ir aktualus, t.y. renkami duomenis ir atliekami matavimai turi padėti siekti iškeltų tikslų ir/ar sekti šių tikslų pasiekimo lygį.

Susitikimo pabaigoje visi komandos nariai turi pritarti sukurtam kokybės planui. Taip pat turi būti atnaujinta projekto dokumentacija.

2.3.7. Scenarijus LAU6

Šeštas susitikimas yra skirtas detalių einamojo ciklo planų sudarymui ir darbų perskirstymui taip, kad kiekvienam komandos nariui tektų įveikiamas darbų krūvis.

10 lentelė. TSP LAU6 scenarijus

Scenarijaus paskirtis	Apibrėžti komandos veiklas sudarant detalų subalansuotą einamojo ciklo planą.
Pradžios kriterijai	<ul style="list-style-type: none"> • Sukurtas bendras projekto planas, užpildytos TASK ir SCHEDULE formos; • Sukurtas kokybės planas.
Bendra informacija	<ul style="list-style-type: none"> • Sudarant subalansuotą planą dalyvauja visi komandos nariai; • Subalansuotas planas yra sudaromas remiantis jau sukurtu bendru projekto planu; • Šio susitikimo metu komandai vadovauja planavimo vadovas.

Veikla	Aprašymas
Sudaryti komandos detalų planą „iš viršaus žemyn“	<ul style="list-style-type: none"> • Einamojo ciklo darbai yra skaidomi į mažesnes užduotis, užduotys yra paskirstomos tarp komandos narių; • Sudaromas pirminis projekto darbų tvarkaraštis, tenkinantis iškeltus laiko apribojimus;
Sudaryti individualius detalius planus	<ul style="list-style-type: none"> • Kiekvienas komandos narys sudaro savo asmeninį darbų tvarkaraštį, apimantį produkto kūrimo darbus ir priskirtos rolės veiklas; • Kiekvienas komandos narys pristato savo planą komandai. Komanda nusprendžia: <ul style="list-style-type: none"> ○ Ar planas pilnas? ○ Ar planas pakankamai detalus? ○ Ar užduotys suderinamos su bendru komandos planu? ○ Ar planas pagrįstas ir pasiekiamas? • Atliekami reikalingi asmeninių planų pakeitimai.
Sudaryti komandos planą „iš apačios į viršų“	<ul style="list-style-type: none"> • Planavimo vadovas vadovauja komandai sudarant komandos planą „iš apačios į viršų“³. • Plano sudarymas remiasi sukurtais individualiais planais.
Peržiūrėti projekto detalų planą ir, jeigu reikia, perskirstyti darbus	<ul style="list-style-type: none"> • Gautas projekto planas „iš apačios į viršų“ yra peržiūrimas ir palyginamas su planu „iš viršaus žemyn“⁴. Jeigu yra randami nesuderinamumai, jie pašalinami vienu iš šių būdų: <ul style="list-style-type: none"> ○ perskirstyti darbus arba pridėti papildomus resursus; ○ atidėti projekto pabaigos terminą; ○ sumažinti produkto/proceso reikalavimus. • Bet kokie projekto galutinių terminų arba produkto reikalavimų pasikeitimai turi būti derinami su vadovybe ir užsakovo atstovais. Už tai atsakingas komandos lyderis. • Pildomos TASK ir SCHEDULE formos.
Pabaigos kriterijai	<ul style="list-style-type: none"> • Sukurtas detalus einamojo ciklo planas; • Sukurti detalūs ir subalansuoti individualūs planai; • Užpildytos bendros ciklo ir individualios TASK ir SCHEDULE formos; • Visi komandos nariai sutinka su sukurtais planais.

Einamojo ciklo darbai yra skaidomi į mažesnes užduotis, Šios savo ruožtu gali būti skaidomos toliau. TSP rekomenduoja skaidyti užduotis tol, kol gausis maždaug 10 valandų trukmės užduotis. Skaidant didesnes užduotis į mažesnes, yra gaunamas ciklo planas „iš viršaus žemyn“. Visos identifikuotos užduotys yra paskirstomos tarp komandos narių. Kai komanda pasidalija visas

³ Plano „iš apačios į viršų“ apibrėžimas pateiktas 2.3 skyriuje.

⁴ Plano „iš viršaus žemyn“ apibrėžimas pateiktas 2.3 skyriuje.

užduotis, ji sudaro pirminį ciklo planą, tenkinantį iškeltus laiko apribojimus, kartu yra užpildomos bendros ciklo TASK ir SCHEDULE formos.

Kiekvienas komandos narys sudaro savo asmeninį planą, į kurį įtraukia jam paskirtus produkto kūrimo darbus ir savo rolės veiklas, užpildo asmenines TASK ir SCHEDULE formas. Kai visi asmeniniai planai yra paruošti, kiekvienas komandos narys pristato savo planą komandai. Komanda aptaria pristatyta planą ir nusprendžia: ar planas yra pilnas? ar jis yra pakankamai detalus? ar suderinamas su bendru planu? ar planas yra pagrįstas ir realus (ar nėra užsibrėžti neįveikiami tikslai)? Po aptarimo kiekvienas komandos narys ištaiso visus pastebėtus savo plano trūkumus, pataiso asmenines TASK ir SCHEDULE formas.

Vadovaujami planavimo vadovo, komandos nariai sudaro einamojo ciklo planą „iš apačios į viršų“. T.y. remiantis sukurtais asmeniniais planais, visos identifikuotos užduotys yra surenkamos į bendrą ciklo planą ir surikiuojamos laike. Sudarant šį planą komanda pradeda nuo mažiausių užduočių ir eina link pačių stambiausių. Toks planavimo būdas paprastai yra tikslesnis negu planavimas „iš viršaus žemyn“.

Gautas planas yra peržiūrimas ir palyginamas su planu „iš viršaus žemyn“. Jeigu yra randami nesuderinamumai, jie yra pašalinami. Šie nesuderinamumai gali būti šalinami tokiais būdais:

- perskirstyti darbus arba pridėti papildomus resursus;
- atidėti projekto pabaigos terminą;
- sumažinti produkto/proceso reikalavimus.

Komandos lyderis yra atsakingas už tai, kad bet kokie projekto galutinių terminų arba produkto reikalavimų pasikeitimai būtų suderinti su vadovybe ir užsakovo atstovais. Visi plano pasikeitimai yra įtraukiami į ciklo TASK ir SCHEDULE formas. Susitikimo pabaigoje visi komandos nariai turi pritarti sudarytam ciklo planui.

2.3.8. Scenarijus LAU7

7 susitikimas yra skirtas rizikų įvertinimui. Komanda identifikuoja projekto rizikas, surikiuoja jas pagal svarbumą ir sudaro pagrindinių rizikų poveikio sumažinimo planą.

11 lentelė. TSP LAU7 scenarijus

Scenarijaus paskirtis	Apibrėžti komandos veiklas atliekant projekto rizikų įvertinimą.
Pradžios kriterijai	<ul style="list-style-type: none"> • Sukurtas detalus einamojo ciklo planas; • Sukurti detalūs ir subalansuoti individualūs planai;

Bendra informacija	<ul style="list-style-type: none"> • Rizikų įvertinime dalyvauja visi komandos nariai; • Susitikimo metu komandai vadovauja komandos lyderis;
Veikla	Aprašymas
Rizikų identifikavimas	<ul style="list-style-type: none"> • Kiekvienas komandos narys išsako savo nuomonę, kokios yra galimos projekto rizikos; • Komanda kolektyviai aptaria visas reikšmingas rizikas;
Rizikų vertinimas	<ul style="list-style-type: none"> • Kiekvienai identifikuotai rizikai yra nustatoma: <ul style="list-style-type: none"> ○ tikimybė, kad ji įvyks; ○ įtaka projektui (pvz., piniginiiais vienetais); • Rizikos yra surikiuojamos pagal numanomą poveikį projektui (tikimybė padauginta iš įtakos); • Visos reikšmingos (turinčios didelį poveikį projektui) rizikos yra dar kartą aptariamoms, nustatomoms mažiausios ir didžiausios tikimybės ir įtakos reikšmės, komandos nariai kartu dar kartą surikiuoja rizikas pagal jų svarbą.
Rizikų poveikio sumažinimo planavimas	<ul style="list-style-type: none"> • Pačioms svarbiausioms rizikos yra nustatoma: <ul style="list-style-type: none"> ○ asmuo, atsakingas už rizikos stebėjimą; ○ laikas, kada galėtų pasireikšti rizikos poveikis; ○ sąlygos, kurioms esant reikia imtis prevencinių/rizikos poveikio mažinimo veiksmų; • Svarbiausios rizikos yra įtraukiamos į ITL formą; • Yra nustatomi rizikų, kurios turėtų pasireikšti artimiausiu metu, poveikio sumažinimo veiksmai; • Paruošiama rizikų ataskaita, kuri bus pateikta vadovybei 9 susitikimo metu;
Pabaigos kriterijai	<ul style="list-style-type: none"> • Visos pagrindinės projekto rizikos yra identifikuotos ir įvertintos; • Užpildyta ITL forma.

Bendros diskusijos metu komandos nariai išsako jų nuomonę svarbias projekto rizikas. Visos rizikos yra aptariamoms, svarbiausios iš jų yra dokumentuojamos.

Kiekvienai identifikuotai rizikai yra nustatoma tikimybė, kad rizika įvyks, ir rizikos įtaka projektui. Rizikos yra surikiuojamos pagal numanomą poveikį projektui, paprastai šį poveikį išreiškia rizikos tikimybės ir įtakos sandauga. Visos rizikos, turinčios didelį poveikį projektui, yra dar karta aptariamoms, nustatomoms jų mažiausios ir didžiausios tikimybės ir įtakos reikšmės. Šios rizikos yra dar kartą surikiuojamos pagal jų svarbą ir įtraukiamos į ITL formą.

Kiekvienai svarbiai projektui rizikai, nurodytai ITL formoje, yra paskiriamas žmogus, kuris yra atsakingas už tos rizikos stebėjimą, nurodomas laikas, kada ji tikriausiai įvyks, triggeris – sąlyga,

žyminti, kada reikia pradėti koregavimo veiksmus, ir pačius koregavimo veiksmus. Toms rizikos, kurios turi įvykti artimiausiu metu, yra sudaromas rizikų poveikio sumažinimo veikslių planas.

Susitikimo pabaigoje, komanda paruošia rizikų ataskaitą, kuri bus įtraukta į bendrą startavimo ataskaitą 8 susitikimo metu.

2.3.9. Scenarijus LAU8

8 susitikimas yra skirtas paruošti sudaryto plano pristatymą vadovybei ir užsakovo atstovams.

12 lentelė. TSP LAU8 scenarijus

Scenarijaus paskirtis	Apibrėžti komandos veiklas ruošiant startavimo ataskaitą.
Pradžios kriterijai	<ul style="list-style-type: none"> • Sukurti detalūs ir subalansuoti individualūs planai; • Visos pagrindinės projekto rizikos yra identifikuotos ir įvertintos;
Bendra informacija	<ul style="list-style-type: none"> • Susitikime dalyvauja visi komandos nariai; • Susitikimo metu komandai vadovauja komandos lyderis;
Veikla	Aprašymas
Paruošti startavimo ataskaitą	Ataskaita apima: <ul style="list-style-type: none"> • Komandos narių ir jų rolių sąrašą; • Startavimo proceso veiklų apžvalgą; • Komandos tikslus; • Bendrus projekto resursų naudojimo ir užduočių atlikimo tvarkaraščius; • Alternatyvius planus (jeigu tokie buvo sukurti); • Kokybės plano santrauką; • Identifikuotas rizikas ir jų poveikio sumažinimo planus.
Peržiūrėti startavimo ataskaitą	<ul style="list-style-type: none"> • Visi komandos nariai peržiūri sukurta ataskaitą; • Visi rasti netikslumai yra aptariami ir ištaisomi.
Pabaigos kriterijai	<ul style="list-style-type: none"> • Paruošta startavimo ataskaita.

Komanda, vadovaujama komandos lyderio, sudaro startavimo ataskaitą, į kurią įtraukia:

- visų komandos narių sąrašą su priskirtomis rolėmis;
- trumpą startavimo veiklų apžvalgą – bendra startavimo proceso struktūra, kas buvo daroma, kokių rezultatų pasiekta;
- komandos tikslus ir tai, kaip šie tikslai padės pasiekti išskeltus verslo ir projekto tikslus;
- projekto planą: užduotys, jų trukmės, išsidėstymas laike, resursų naudojimo tvarkaraštis;

- jeigu komanda numatė alternatyvius planus, į ataskaitą turi būti įtraukta šių planų santrauka ir skirtumai nuo pagrindinio plano;
- kokybės plano santrauką;
- rizikų vertinimo santrauką (rizikų ataskaita, sukurta 7 susitikimo metu).

Paruošta startavimo ataskaita yra peržiūrima, visi rasti netikslumai yra aptariami ir pašalinami.

2.3.10. Scenarijus LAU9

9 susitikimas – susitikimas su vadovybės ir užsakovo atstovais. Šio susitikimo metu komanda pristato projekto planą, paaiškina, kaip jis buvo sudarytas, ir parodo, kad visi komandos nariai sutinka su šiuo planu ir yra įsipareigoję.

13 lentelė. TSP LAU9 scenarijus

Scenarijaus paskirtis	Apibrėžti komandos veiklas pristatant vadovybei startavimo ataskaitą.
Pradžios kriterijai	<ul style="list-style-type: none"> • Sukurti detalūs ir subalansuoti individualūs planai; • Visos pagrindinės projekto rizikos yra identifikuotos ir įvertintos; • Paruošta startavimo ataskaita;
Bendra informacija	<ul style="list-style-type: none"> • Susitikime dalyvauja visi komandos nariai; • Susitikime dalyvauja vadovybės ir užsakovo atstovai;
Veikla	Aprašymas
Pristatyti startavimo ataskaita	<p>Komanda pristato:</p> <ul style="list-style-type: none"> • Sukurtus planus; • Kaip šie planai buvo sudaryti (procesas, prielaidos, argumentai); • Komanda turi parodyti, kad visi komandos nariai pritaria šiems planams ir prisiima įsipareigojimą juos vykdyti.
Aptarti ataskaita su vadovybe ir užsakovo atstovais	<ul style="list-style-type: none"> • Komanda aptaria startavimo ataskaita kartu su vadovybės ir užsakovo atstovais, atsako į visus iškilusius klausimus; • Jeigu sukurti planai netenkina vadovybės ir/ar užsakovo, komanda pristato alternatyvius planus; • Jeigu komandai yra pavedama sukurti naują alternatyvų planą, komandos nariai turi suprasti (išsiaiškinti): <ul style="list-style-type: none"> ○ kokie plano pakeitimai turi būti padaryti; ○ kokie yra planuojamų resursų projektui pasikeitimai; ○ numatomus papildomus darbus arba darbus, kurių yra atsisakyta.
Pabaigos kriterijai	<ul style="list-style-type: none"> • Vadovybės ir užsakovo atstovai pritaria sukurtam projekto planui.

Pirmoje susitikimo pusėje komandos lyderis pristato startavimo metu sukurtą planą, kūrimo strategiją, prielaidas, kuriais buvo remiamasi, argumentuoja pagrindinius sprendimus. Pristatant startavimo ataskaitą svarbu parodyti, kad visi komandos nariai sutinka su planu ir yra išipareigoję jį vykdyti.

Antroje susitikimo pusėje vadovybės ir užsakovo atstovai užduoda komandos nariams klausimus, išsako savo pastabas. Bendroje diskusijoje yra aptariami visi iškilę klausimai. Jeigu vadovybės ar užsakovo netenkina komandos sukurtas projekto planas, komandos nariai pristato alternatyvius planus (jeigu tokie buvo sukurti), jų privalumus ir trūkumus.

Jeigu komanda nėra paruošusi alternatyvaus plano, arba jis irgi netenkina vadovybės arba užsakovo, komanda turi išsiaiškinti:

- kurios plano ar strategijos vietos yra taisytinios, ir kodėl jos turi būti pakeistos;
- kokie yra planuojami/leistini resursų, skirtų projektui, pasikeitimai;
- kokie papildomi darbai turi būti įtraukti į projekto planą ir/arba kokių darbų yra atsisakoma.

TSP startavimo pabaigoje komanda ir vadovybė turi sutarti dėl projekto plano, t.y. turi būti gautas vadovybės pritarimas. Arba komanda turi pakartoti startavimo procesą ir paruošti naują projekto planą.

2.3.11. Scenarijus LAUPM

Aptarimas - paskutinis startavimo susitikimas. Aptarimo metu yra aptariami pasiruošimo kassavaitiniams susitikimams darbai, pildomos proceso gerinimo pasiūlymų formos, atnaujinama ir peržiūrima projekto dokumentacija.

14 lentelė. TSP LAUPM scenarijus

Scenarijaus paskirtis	Apibrėžti komandos veiklas startavimo aptarimo metu.
Pradžios kriterijai	<ul style="list-style-type: none"> • Sukurti detalūs ir subalansuoti individualūs planai; • Visos pagrindinės projekto rizikos yra identifikuotos ir įvertintos; • Vadovybės ir užsakovo atstovai pritaria sukurtam projekto planui;
Bendra informacija	<ul style="list-style-type: none"> • Susitikime dalyvauja visi komandos nariai; • Komandai vadovauja TSP instruktorius arba komandos lyderis;
Veikla	Aprašymas
Apžvelgti kassavaitinių susitikimų tvarką	<ul style="list-style-type: none"> • Susitarti dėl kassavaitinių susitikimų laiko; • Aptarti susitikimų tvarką ir pasiruošimo susitikimams darbus;

Aptarti startavimo proceso eigą	<ul style="list-style-type: none"> • Aptarti startavimo proceso eigą, kilsius neaiškumus ir kitus susijusius klausimus; • Užpildyti proceso gerinimo pasiūlymų formas (PIP);
Papildyti ir peržiūrėti projekto dokumentaciją	<ul style="list-style-type: none"> • Papildyti projekto dokumentaciją startavimo aptarimo metu gautais duomenimis ir dokumentais; • Peržiūrėti projekto dokumentaciją. Jeigu kažkokių dokumentų ar formų trūksta, papildyti projekto dokumentaciją. • Sutvarkyti projekto dokumentaciją taip, kad ją būtų galima pasinaudoti būsimuose projektuose.
Pabaigos kriterijai	<ul style="list-style-type: none"> • Užpildytos PIP formos; • Papildyta projekto dokumentacija.

Pirmoje aptarimo dalyje komanda apsisprendžia dėl kasavaitinių susitikimų laiko, kuris tenkintų visus komandos narius, aptaria susitikimų tvarką, paskirtį ir turinį. Taipogi yra aptariami pasiruošimo kasavaitiniams susitikimams darbai, kiekvienas komandos narys yra informuojamas, kokius pasiruošimo darbus jis turės atlikti.

Antroje susitikimo dalyje komanda aptaria startavimo proceso eigą, startavimo metu kilusias problemas ir kitus susijusius klausimus. Komandos nariai užpildo proceso gerinimo pasiūlymų formas (PIP).

Paskutinėje aptarimo dalyje komanda peržiūri projekto dokumentaciją, papildo ją PIP formomis. Jeigu komanda aptinka, kad kažkokių duomenų trūksta, projekto dokumentacija yra papildoma. Galiausiai, komandos nariai sutvarko projekto dokumentaciją taip, kad joje esančiais duomenimis būtų galima kuo paprasčiau pasinaudoti vėliau projekto metu ir būsimuose projektuose.

2.3.12. Scenarijus REL1

REL1 yra naudojamas pakartotinių startavimų metu vietoje LAU1 scenarijaus. Visi kiti pakartotinio startavimo (REL) scenarijai sutampa su atitinkamais LAU scenarijais. Pirmame pakartotinio startavimo susitikime dalyvauja visi komandos nariai, komandos lyderis, startavimo instruktorius, vadovybės ir užsakovo arba pardavimų skyriaus atstovai. Šio susitikimo tikslas yra peržvelgti produkto, užsakovo ir vadovybės tikslus, susijusiais su vykdomu projektu, aptarti numatytus ciklo darbus bei bet kokius tikslų, darbų apimties, laiko apribojimų pasikeitimus ir kitus reikšmingus faktorius, įtakančius projekto eigą.

15 lentelė. TSP REL1 scenarijus

Scenarijaus paskirtis	Apibrėžti pirmo pakartotinio startavimo susitikimo veiklas.
Pradžios kriterijai	<ul style="list-style-type: none"> • Ankstesnis ciklas yra baigtas ir yra atliktas jo aptarimas; • Arba projekto metu įvyko pasikeitimai, kurie privertė iš esmės koreguoti projekto planus; • Susitarta dėl vadovybės ir užsakovo atstovų dalyvavimo pirmame pakartotinio startavimo susitikime; • Susitikimui vadovauja TSP instruktorius.
Bendra informacija	<ul style="list-style-type: none"> • Susitikime dalyvauja visi komandos nariai ir vadovybės bei užsakovo (arba pardavimų skyriaus) atstovai;
Veikla	Aprašymas
Išmoktos pamokos	TSP instruktorius trumpai apžvelgia praėjusio ciklo rezultatus, aptaria pagrindines problemas ir jų priežastis, veiksmus, kurie turi užtikrinti sklandesnę komandos darbą einamajame cikle.
Komandos pristatymas	Jeigu komandoje įvyko kokie nors pasikeitimai (pvz., personalo pasikeitimai), komandos lyderis trumpai pristato šiuos pasikeitimus.
Reikalavimų produktui pristatymas	<ul style="list-style-type: none"> • Užsakovo ar pardavimų skyriaus atstovas pristato: <ul style="list-style-type: none"> ○ projekto tikslus einamajam ciklui; ○ pagrindines produkto savybes, kurios yra planuojamos įgyvendinti einamojo ciklo metu; ○ svarbius reikalavimus ir specifikacijas ir /ar standartus, kuriems turi atitikti kuriamas produktas; ○ kitą aktualią informaciją;
Verslo tikslų pristatymas	<ul style="list-style-type: none"> • Vadovybė pristato verslo tikslus, susijusius su projektu: <ul style="list-style-type: none"> ○ vadovybės lūkesčiai; ○ idealus rezultatas; ○ minimalus priimtinas rezultatas; ○ pristatymas apima kainą, laiką ir kokybės tikslus;
Reikalavimų ir tikslų aptarimas	<ul style="list-style-type: none"> • Vadovybės ir užsakovo atstovai bei komandos nariai aptaria verslo tikslus ir reikalavimus produktui: <ul style="list-style-type: none"> ○ aptariama, kaip projektas prisidės prie verslo tikslų realizavimo; ○ nustatomi minimalūs reikalavimai; ○ jeigu reikia, randami kompromisai ir t.t.;
Pabaigos kriterijai	<ul style="list-style-type: none"> • Identifikuoti ir dokumentuoti pagrindiniai produkto reikalavimai ir verslo tikslai einamajam projekto ciklui.

Susitikimo pradžioje TSP instruktorius trumpai pristato praėjusio ciklo rezultatus: kokie darbai yra atlikti, kokia yra šių darbų kokybė, kokie darbai persikelia į naują ciklą (neatlikti ir papildomi darbai). Taip pat instruktorius kartu su komanda ir vadovybės bei užsakovo atstovais aptaria

pagrindines praėjusio ciklo problemas ir šių problemų priežastis. Instruktorius pasiūlo galimus šių problemų sprendimo būdus.

Komandos lyderis pristato visus pasikeitimus, kurie įvyko komandoje palyginus su praeitū ciklu. Dažniausiai tai būna personalo pasikeitimai: komandos nariai, kurie paliko projektą, naujai atėję žmonės ir pan. Šio pristatymo tikslas parodyti, kad įvykę pasikeitimai neturės neigiamų pasekmių projektui, bei pristatyti visiems naujus komandos narius.

Toliau užsakovo atstovai trumpai pristato projekto tikslus einamajam ciklui, laukiamas kuriamo produkto savybes, reikalavimus ir kitą aktualią informaciją.

Po užsakovo atstovų pristatymo savo lūkesčius išsako vadovybės atstovai. Šis pristatymas apima verslo tikslus, ko yra tikimasi iš einamojo projekto ciklo, idealų ir minimalų priimtina rezultata, numatomą projekto ciklo kainą, trukmę ir kokybės tikslus.

Galiausiai komandos nariai gali užduoti savo klausimus vadovybės ir užsakovo atstovams, aptarti su jais visus iškilusius neaiškumus.

2.4. Inspektavimo (INS) scenarijus

Inspektavimas yra skirtas specifinių klaidų paieškai. Inspektavimas turi būti atliekamas mažiausiai trijų žmonių: produkto kūrėjas ir dar du inžinieriai. Prieš pradėdant inspektavimą, produktas (ar inspektuojama jo dalis) turi būti peržiūrėtas. Elementarios klaidos blaško inspektavimo komandos narių dėmesį ir dėl to, jie gali praleisti svarbias ir ne tokias akivaizdžias klaidas. Tokiu būdu inspektavimo produktyvumas gali smarkiai sumažėti.

16 lentelė. TSP INS scenarijus

Scenarijaus paskirtis	Apibrėžti inspektavimo proceso veiklas.
Pradžios kriterijai	<ul style="list-style-type: none"> • Inspektuojamas produktas yra sukurtas ir peržiūrėtas.
Bendra informacija	<ul style="list-style-type: none"> • Inspektavimas yra skirtas tam tikrų specifinių problemų analizei, o ne paprastų defektų paieškai ir taisymui.
Veikla	Aprašymas
Atlikti pre-inspekciją	<ul style="list-style-type: none"> • Yra paskiriamas inspektavimo moderatorius (paprastai kokybės arba proceso vadovas); • Inspektavimo moderatorius atlieka produkto pre-inspekciją, kurios tikslas yra įsitikinti, kad produktas yra paruoštas inspektavimui. Jeigu ne, produktas yra grąžinamas kūrėjui; • Inspektavimo moderatorius parenka kitus inspektavimo dalyvius.
Pravesti trumpa inspektavimo pasitarimą	<ul style="list-style-type: none"> • Pasitarime dalyvauja inspektavimo moderatorius, produkto kūrėjas ir inspektavimo dalyviai;

	<ul style="list-style-type: none"> • Inspektavimo moderatorius trumpai pristato inspektavimo procesą ir tikslus; • Inspektuojamo produkto kūrėjas supažindina inspektavimo dalyvius su savo produktu; • Inspektavimo dalyviai nusprendžia, kokiems produkto savybėms (aspektams) bus skirtas ypatingas dėmesys; • Inspektavimo moderatorius paskiria inspektavimo laiką.
Produkto peržiūra	Kiekvienas inspektavimo dalyvis individualiai peržiūri produktą, pažymi rastus defektus.
Produkto inspektavimas	<ul style="list-style-type: none"> • Inspektavimo dalyviai, vadovaujami moderatoriaus, aptaria kiekvieną produkto dalį: <ul style="list-style-type: none"> ○ kiekvienas inspektavimo dalyvis pristato savo rastus defektus; ○ esminiai defektai yra fiksuojami INS formoje; ○ pažymi, kurie inspektavimo dalyviai kokius esminius defektus rado; ○ produkto kūrėjas visus rastus esminius defektus fiksuoja LOGD formoje. • Inspektavimo moderatorius pagal specialią formulę išskaičiuoja neaptiktų (ir paliktų produkte) defektų skaičių; • Inspektavimo moderatorius apskaičiuoja kiekvieno inspektavimo dalyvio efektyvumą; • Inspektavimo dalyviai papildo savo kontrolinius sąrašus, skirtus peržiūrai; • Inspektavimo dalyviai ir moderatorius nusprendžia, ar yra reikalingas pakartotinis produkto inspektavimas, kas jame dalyvaus, kada jis vyks, kaip verifikuoti pataisymus, t.y. kaip patikrinti pataisimų teisingumą; • Inspektavimo moderatorius užpildo LOGD ir INS formas.
Defektų taisymas ir pataisymų verifikavimas	<ul style="list-style-type: none"> • Produkto kūrėjas pataiso produktą ir atnaujina dokumentaciją; • Yra atliekamos reikalingos peržiūros ir/ar inspektavimas; • Yra atliekamas pataisymų verifikavimas.
Aptarimas	<ul style="list-style-type: none"> • Aptarti inspektavimo eigą; • Užpildyti proceso gerinimo pasiūlymų formas; • Atnaujinti projekto dokumentaciją.
Pabaigos kriterijai	<ul style="list-style-type: none"> • Užpildytos LOGD ir INS formas; • Visi rasti defektai yra ištaisyti; • Atnaujinta projekto dokumentacija.

Inspektavimas yra inicijuojamas produkto kūrėjo. Kūrėjas susisieikia su inspektavimo moderatoriumi, dažniausiai juo būna proceso arba kokybės vadovas, bet gali būti ir kitas inžinierius, kuris žino, kaip vadovauti inspektavimui, ir susitaria dėl produkto inspektavimo. Inspektavimo

moderatorius atlieka produkto pre-inspekciją, kurios tikslas yra atmesti akivaizdžiai netinkamus inspektavimui produktus. Moderatorius įsitikina, kad produktas yra baigtas kurti ir visos jo dalys (pvz., visas kodas, dokumentacija) yra prieinamos visiems inspektavimo dalyviams, kad produktas yra peržiūrėtas ir visos akivaizdžios klaidos yra ištaisytos. Moderatorius, nusprendžia, kiek žmonių turi dalyvauti inspektavime, parenka tinkamus inžinierius ir susitaria dėl jų dalyvavimo inspektavime.

Suderintu laiku įvyksta pirmas inspektavimo susitikimas, kuriame dalyvauja inspektavimo moderatorius, produkto kūrėjas ir pasirinkti inspektavimo dalyviai. Moderatorius trumpai pristato inspektavimo procesą, jo tikslus ir atlikimo terminus. Produkto kūrėjas trumpai pristato savo produktą, pagrindines jo funkcijas ir svarbiausias savybes. Susitikimo dalyviai susitaria, kokios produkto savybės yra svarbiausios/kritinės ir kokiems aspektams inspektavimo metu bus skiriama daugiausiai dėmesio. Tokiais aspektais gali būti saugumas, atsparumas trykiams ir pan. Toliau moderatorius suderina su inspektavimo dalyviais inspektavimo laiką ir paskiria kito susitikimo datą.

Iki nurodytos datos kiekvienas inspektavimo dalyvis individualiai peržiūri produktą ir pažymi visus rastus defektus. Peržiūros metu patariama naudoti išlanksto sudarytais kontroliniais sąrašais.

Antro susitikimo metu inspektavimo dalyviai ir produkto kūrėjas, vadovaujami moderatoriaus, aptaria iš eilės visas produkto dalis. Kiekvienas inspektavimo dalyvis pristato savo rastus defektus. Visi rasti defektai yra aptariami. Inspektavimo (INS) formoje yra fiksuojami visi esminiai defektai ir juos aptikusių inžinierių vardai. Produkto kūrėjas visus rastus defektus fiksuoja LOGD formoje.

Susitikimo pabaigoje inspektavimo moderatorius apskaičiuoja kiekvieno inspektavimo dalyvio našumą (rastų defektų skaičių) ir neaptiktų (ir paliktų produkte) defektų skaičių. Paskutinis skaičius yra apskaičiuojamas pagal statistikos metodais paremtą formulę:

$$\frac{A}{B * C} - (A + B - C)$$

Kur, tuo atveju, kai inspektavime dalyvauja du inžinieriai, A – pirmo inžinieriaus rastų defektų skaičius, B- antro inžinieriaus rastų defektų skaičius, C - abiejų inžinierių rastų vienuodų defektų skaičius, A*B/C – bendras produkto defektų skaičius, A+B-C – rastų defektų skaičius. Tuo atveju, kai inspektavime dalyvauja daugiau negu du inžinieriai, skaičius A yra lygus inžinieriaus, kuris rado daugiausiai unikalių (kurių nerado kiti inžinieriai) defektų, rastų defektų skaičiui, B – bendras skirtingų defektų, rastų visų kitų inžinierių, skaičius.

Toliau yra apskaičiuojamas bendras inspekcijos efektyvumas:

$$\frac{100 * (A + B - C) * C}{A * B}$$

Ir kiekvieno iš inspektavimo dalyvių efektyvumas:

$$\frac{100 * N * C}{A * B}$$

Kur N – konkretaus inspekcijos dalyvio rastų defektų skaičius, A*B/C – bendras produkto defektų skaičius.

Apskaičiavus efektyvumo reikšmes, inspektavimo dalyviai nusprendžia, ar produktas yra pakankamai geros kokybės, ar po defektų ištaisymo reikės atlikti papildomą inspektavimą. Antru atveju taip pat reikia nuspręsti, kada inspektavimas vyks, kas jame dalyvaus ir kaip patikrinti pataisymų korektiškumą. Susitikimo pabaigoje inspektavimo dalyviai papildo savo kontrolinius sąrašus, naudojamus peržiūros ir inspektavimo metu. Moderatorius užpildo LOGD ir INS formas.

Jeigu produkto inspektavimo metu buvo rasti defektai, tai produkto kūrėjas jas ištaiso, atnaujina dokumentaciją. Kai visi defektai yra ištaisyti, yra atliekamas pakartotinis produkto inspektavimas ir pataisymų verifikavimas.

Inspektavimas pasibaigia aptarimu. Jo metu inspektavimo dalyviai, moderatorius ir produkto kūrėjas aptaria inspektavimo eigą, iškilusias problemas, užpildo proceso gerinimo pasiūlymo formas (PIP). Inspektavimo moderatorius atnaujina projekto dokumentaciją.

2.5. Reikalavimų sudarymo (REQ) scenarijus

Pagrindinis reikalavimų sudarymo proceso rezultatas – reikalavimų specifikacija, kuri aprašo visas būtinas kuriamos programų sistemos savybes ir funkcijas. Reikalavimų sudarymo procesas apima rinkos tyrimo, reikalavimų išaiškinimo, dokumentavimo, derinimo veiklas ir sistemos testavimo plano sudarymą.

17 lentelė. TSP REQ scenarijus

Scenarijaus paskirtis	Apibrėžti reikalavimų sudarymo proceso veiklas.
Pradžios kriterijai	<ul style="list-style-type: none"> • Kūrimo strategija yra sudaryta; • Projekto planas yra suderintas su vadovybe; • Startavimo proceso duomenys yra įtraukti į projekto dokumentaciją.
Bendra informacija	<ul style="list-style-type: none"> • Reikalavimų sudarymo proceso rezultatas – programinės įrangos reikalavimų specifikacija. Ši specifikacija apima programinio produkto funkcijų aprašymą ir šių funkcijų naudojimo diagramas (angl. use-case). • Ypatingas dėmesys turi būti skirtas reikalavimų detalizavimui: <ul style="list-style-type: none"> ○ neturint pakankamai patirties su konkrečių taikomųjų programų tipu, reikalavimai, kurie atrodo paprasti, gali pareikalauti daug daugiau laiko, negu buvo planuota;

	○ gera praktika – pridėdinti funkcijas nedideliais prieaugiais.
Veikla	Aprašymas
Atlikti rinkos tyrimą	<p>Prieš pradėdant sudarinėti reikalavimus komanda turi išsiaiškinti:</p> <ul style="list-style-type: none"> • kokie produktai, panašūs į kuriamą, jau yra išleisti; • kokios yra šių produktų galimybės ir kokie yra trūkumai; • ar verta kurti produktą nuo nulio? Gal yra prasminga nusipirkti jau sukurtą produktą ir pritaikyti jį užsakovo poreikiams?
Reikalavimų išaiškinimas	<p>Reikalavimų išaiškinimas (angl. elicitation) apima:</p> <ul style="list-style-type: none"> • sistemos įgyvendinamumo įvertinimą; • užsakovo organizacijos poreikių supratimą; • visų suinteresuotų asmenų identifikavimą; • reikalavimų šaltinių identifikavimą; • sistemos operacinės aplinkos apibrėžimą; • reikalavimų ir jų pagrindimo dokumentavimą.
Reikalavimų prototipavimas	<p>Tam kad geriau suprasti svarbius arba ne iki galo aiškius reikalavimus, yra atliekamas reikalavimų prototipavimas. Lengvasvorių prototipų sukūrimas neužima daug laiko, o jų naudojimas padeda lengviau suprasti reikalavimų esmę ir užtikrinti vienodą užsakovo ir vykdytojų reikalavimų supratimą.</p>
Reikalavimų specifikacijos sukūrimas	<ul style="list-style-type: none"> • Komanda pasidalina darbais, paprastai kiekvienas komandos narys yra atsakingas už tam tikros reikalavimų specifikacijos dalies sudarymą; • Reikalavimų specifikacija aprašo, ką turi daryti kuriamas produktas ir kaip jis turi elgtis tam tikrose situacijose. • Reikalavimų specifikacijoje neturi būti informacijos apie tai, kaip produktas bus realizuojamas. Produkto realizavimo klausimai yra sprendžiami projektavimo stadijoje; • Už skirtingų specifikacijos dalių surinkimą į vieną visumą yra atsakingas kūrimo vadovas; • Sujungus visas dalis į vieną dokumentą, komanda atlieka jo peržiūrą.
Reikalavimų specifikacijos inspektavimas	<ul style="list-style-type: none"> • Inspektavimo tikslas – surasti klaidas ir nesuderinamumus reikalavimų specifikacijoje prieš pradėdant produkto kūrimo darbus; • Inspektavimas vyksta pagal INS scenarijų.
Sistemos testavimo plano sudarymas	<p>Planuojami testai, skirti patikrinti:</p> <ul style="list-style-type: none"> • ar sistema tikrai atlieka visas numatytas funkcijas; • ar sistema tenkina iškeltus kokybės reikalavimus; • ar sistemos elgesys normaliomis/stresinėmis sąlygomis atitinka specifikaciją. <p>Sudarytas sistemos testavimo planas yra peržiūrimas.</p>
Sistemos testavimo plano inspektavimas	<ul style="list-style-type: none"> • Inspektavimo tikslas – surasti klaidas ir nesuderinamumus sistemos testavimo plane prieš pradėdant testų kūrimo darbus; • Inspektavimas vyksta pagal INS scenarijų.

Reikalavimų specifikacijos peržiūra su vartotojais	<ul style="list-style-type: none"> • Reikalavimų specifikacijos kopija yra pateikiama busimiems sistemos vartotojams ir/ar užsakovo atstovams; • Kūrimo vadovas išaiškina vartotojams ir užsakovo atstovams visas jiems neaiškias vietas; • Visi vartotojų rasti netikslumai yra pataisomi; • Pataisyta reikalavimų specifikaciją turi būti patvirtinta užsakovo atstovų.
Aptarimas	<ul style="list-style-type: none"> • Aptarti reikalavimo formulavimo proceso eigą; • Užpildyti proceso gerinimo pasiūlymų formas; • Atnaujinti projekto dokumentaciją.
Pabaigos kriterijai	<ul style="list-style-type: none"> • Sukurti reikalavimų specifikacija ir sistemos testavimo planas; • Užpildyta reikalavimų specifikacijos inspektavimo forma (INS); • Užsakovas patvirtino reikalavimų specifikaciją; • Duomenys apie atliktus darbus, sugaištą laiką, padarytus ir ištaisytus defektus yra užpildyti į atitinkamas TSP formas; • Atnaujinta projekto dokumentacija.

Prieš pradėdant sudarinėti reikalavimus komanda turi atlikti esamos rinkos tyrimą, išsiaiškinti, kokie analogiški produktai egzistuoja, kokios yra jų galimybės, kokie trūkumai. Šis tyrimas padeda komandai susipažinti su dalykine sritimi, kuriai bus kuriamas programinis produktas, suprasti, kokie yra analogiškų produktų bendriniai reikalavimai. Tuo pačiu komanda turi nuspręsti, ar verta kurti produktą nuo nulio, o gal yra prasminga nusipirkti jau egzistuojantį produktą ir pritaikyti jį užsakovo poreikiams.

Atlikus rinkos tyrimą, komanda pradeda reikalavimų išaiškinimą. Jo metu komanda:

- nustato užsakovo organizacijos poreikius;
- identifikuoja visus suinteresuotus asmenis, t.y. visus, kas tiesiogiai ir netiesiogiai naudosis sistema arba ją prižiūrės, ir jų poreikius;
- identifikuoja visus galimus reikalavimų šaltinius;
- apibrėžia sistemos operacinę aplinką;
- įvertina sistemos įgyvendinamumą;
- dokumentuoja reikalavimus ir jų pagrindimą ir/ar šaltinį.

Tam, kad geriau suprasti svarbius reikalavimus komandos nariams gali tekti sukurti vieną arba keletą prototipų. Paprastai reikalavimų sudarymo metu yra kuriami lengvasvoriniai prototipai (pvz., nupieštas popieriuje vartotojo interfeiso eskizas). Tokių prototipų kūrimas palengvina reikalavimų supratimą ir užtikrina vienodą reikalavimų supratimą tiek iš vartotojų, tiek iš sistemos kūrėjų pusės.

Išsiaiškinus užsakovo organizacijos poreikius ir pagrindinius sistemos reikalavimus, komanda sudaro reikalavimų specifikaciją, kuri aprašo, ką ir kaip turį daryti kuriama sistema (produktas), kaip ji turi elgtis tam tikrose situacijose. Reikalavimų specifikacijoje nėra aprašoma, kaip sistema bus realizuota. Sistemos realizavimo aspektai yra aprašomi projektinėje specifikacijoje. Paprastai kiekvienas komandos narys yra atsakingas už tam tikros reikalavimų specifikacijos dalies sukūrimą. Už skirtingų dalių surinkimą į vieną dokumentą yra atsakingas kūrimo vadovas. Visa komanda peržiūri gautą reikalavimų specifikaciją. Po to yra atliekamas reikalavimų specifikacijos inspektavimas. Inspektavimas vyksta pagal INS scenarijų (žr. 2.4 skyrių).

Egzistuoja keletas reikalavimų specifikacijos standartų, konkrečios specifikacijos turinys ir apipavidalinimas priklauso nuo projekto ypatybių, užsakovo norų ir kitų aspektų. Pagrindinės reikalavimų specifikacijos dalys [Hum99]:

1. Turinys.
2. Įvadas (reikalavimų specifikacijos paskirtis, taikymo srities aprašymas, projekto informacija).
3. Funkciniai reikalavimai (sistemos funkcinų reikalavimų aprašymas, pirmo ciklo reikalavimai, antro ciklo reikalavimai, ...).
4. Nefunkciniai reikalavimai (sistemos nefunkcinų reikalavimų aprašymas, pirmo ciklo reikalavimai, antro ciklo reikalavimai, ...).
5. Išorinių interfeisų reikalavimai (vartotojo grafinei sąsaja, grafinių elementų išsidėstymas ekrane).
6. Projektavimo/realizavimo apribojimai (standartai, kuriuos turi atitikti sistema, kiti kūrimo apribojimai).
7. Specialūs sistemos reikalavimai (dokumentacija, suderinamumas).
8. Nuorodos ir informacijos šaltiniai.

Turėdama dokumentuotus sistemos reikalavimus, komanda nusprendžia, kaip bus tikrinamas šių reikalavimų įgyvendinimas ir sudaro sistemos testavimo planą, kuris turi apimti ir režijų bei stresinį testavimą, sistemos naudojamumo ir atstatomumo įvertinimą. Sistemos testavimo plane yra nurodoma, kokie sistemos aspektai bus testuojami, o kokie ne, ir kodėl. Sudarytas planas yra peržiūrimas ir inspektuojamas. Inspektavimas vyksta pagal INS scenarijų (žr. 2.4 skyrių).

Toliau reikalavimų specifikacija yra pateikiama užsakovui ir būsimiems sistemos naudotojams. Pirmą, užsakovas ir būsimi vartotojai susipažįsta su reikalavimų specifikacija. Tada yra surengiamas susitikimas, kurio metu komandos nariai (paprastai už tai yra atsakingas kūrimo vadovas arba komandos lyderis) aptaria kartu su užsakovu ir būsimais vartotojais visas neaiškias

vietas. Užsakovas ir vartotojai pateikia savo pastabas, kurios irgi yra aptariamoms, ir jeigu reikia, yra reikalavimų specifikacija yra pataisoma. Galiausiai, reikalavimų specifikacija turi būti patvirtinta abiejų šalių (užsakovo ir kūrėjų).

Aptarimo metu komandos nariai aptaria reikalavimų sudarymo proceso eigą, iškilusias problemas, užpildo proceso gerinimo pasiūlymo formas (PIP), atnaujina projekto dokumentaciją.

2.6. Aukšto lygio projektavimo (HLD) scenarijus

Šis scenarijus yra skirtas aukšto lygio projektavimo veikloms apibrėžti. Projektavimo proceso tikslas – sukurti sistemos (programinės įrangos) projektinę specifikaciją. Ši specifikacija apima einamojo ciklo darbus, t.y. aprašo, kas ir kaip turi būti sukurta einamojo ciklo metu.

18 lentelė. TSP HLD scenarijus

Scenarijaus paskirtis	Apibrėžti projektavimo proceso veiklas.
Pradžios kriterijai	<ul style="list-style-type: none"> • Kūrimo strategija yra sudaryta; • Projekto planas yra suderintas su vadovybe; • Atliktas reikalavimų specifikacijos inspektavimas, specifikacija yra patvirtinta užsakovo.
Bendra informacija	<p>Projektavimo proceso rezultatas – sistemos (programinės įrangos) projektinė specifikacija, apibrėžianti bendrą produkto struktūrą einamajam ciklui, t.y. kas ir kaip turi būti sukurta einamojo ciklo metu. Projektinė specifikacija apima:</p> <ul style="list-style-type: none"> • Pagrindinius sistemos komponentus ir jų interfeisų specifikacijas; • Komponentų naudojimo diagramas (angl. use-case); • Projektavimo standartus (failų ir pranešimų standartus, vardų sudarymo susitarimus, defektų standartus ir t.t.).
Veikla	Aprašymas
Struktūrinis projektavimas	<ul style="list-style-type: none"> • Šios veiklos metu komandai vadovauja kūrimo vadovas; • Struktūrinis projektavimas apima: <ul style="list-style-type: none"> ○ produkto struktūros, kurią planuojama įgyvendinti einamojo ciklo metu, aprašymą; ○ vardų suteikimą produkto komponentams; ○ produkto komponentų naudojimo diagramų sukūrimą; ○ projektavimo užduočių (darbų) identifikavimą; ○ užduočių paskirstymą tarp komandos narių (šiai veiklai vadovauja komandos lyderis).
Projektavimo standartų apibrėžimas	<p>Komanda, vadovaujama proceso vadovu, apibrėžia, kokius standartus ji naudos projektavimo metu. Projektavimo standartų pavyzdžiai:</p> <ul style="list-style-type: none"> • vardų sudarymo susitarimai; • interfeisų formatas (apipavidalinimas, parametrų skaičius, kurie

	<p>parametrai yra skirti kintamiesiems, klaidų kodams, vidinėms sąlygoms ir pan.);</p> <ul style="list-style-type: none"> • sistemos ir klaidų pranešimų standartai; • defektų standartai; • kodo eilučių (LOC) skaičiavimo standartai; • projektavimo dokumentacijos standartizavimas.
Interfeisų ir duomenų specifikacijų sukūrimas	<p>Komanda, vadovaujama kūrimo vadovų sukuria:</p> <ul style="list-style-type: none"> • visų išorinių sistemos interfeisų specifikacijas; • vidinių sistemos interfeisų specifikacijas (t.y. interfeisiai, skirti duomenų apsikeitimui tarp sistemos komponentų); • sistemos naudojamų duomenų ir duomenų struktūrų specifikacijas.
Komponentų specifikacijų sukūrimas	<ul style="list-style-type: none"> • Komandos nariai, atliekami jiems paskirtas užduotis, sukuria atskirų komponentų specifikacijas; • Komponentų specifikacijos apima tik bendrųjų komponento struktūros elementų apibrėžimą. Detalusis komponentų projektavimas yra atliekamas vėliau (žr. realizavimo (IMP) scenarijų).
Integracijos testavimo plano sudarymas	<ul style="list-style-type: none"> • Integracijos testavimas yra skirtas įsitikinti, kad visi vidiniai sistemos interfeisai yra suderinami ir jų kūrimo metu nebuvo padaryta klaidų; • Planuojant integracijos testavimą, komanda nusprendžia: <ul style="list-style-type: none"> ○ kokia bus testavimo strategija (testavimo metodikos ir būdai); ○ kokie testai yra reikalingi tam, kad patikrinti kiekvieno iš sistemos vidinių interfeisų korektiškumą; • Sukurtas integracijos testavimo planas yra peržiūrimas (peržiūroje dalyvauja visa komanda).
Sistemos projektinės specifikacijos sudarymas	<ul style="list-style-type: none"> • Kūrimo vadovas yra atsakingas už galutinio projektinės specifikacijos dokumento sudarymą; • Sudaryta projektinė dokumentacija yra peržiūrima (peržiūroje dalyvauja visa komanda);
Sistemos projektinės specifikacijos ir integracijos testavimo plano inspektavimas	<ul style="list-style-type: none"> • Inspektavimo tikslas – surasti klaidas ir nesuderinamumus sistemos testavimo plane prieš pradėdant testų kūrimo darbus; • Inspektavimas vyksta pagal INS scenarijų.
Aptarimas	<ul style="list-style-type: none"> • Aptarti projektavimo proceso eigą; • Užpildyti proceso gerinimo pasiūlymų formas; • Atnaujinti projekto dokumentaciją.
Pabaigos kriterijai	<ul style="list-style-type: none"> • Sukurta sistemos projektinė specifikacija ir integracijos testavimo planas; • Užpildyta projektinės specifikacijos inspektavimo forma (INS); • Duomenys apie atliktus darbus, sugaištą laiką, padarytus ir ištaisytus defektus yra užpildyti į atitinkamas TSP formas;

- | | |
|--|--|
| | <ul style="list-style-type: none">• Atnaujinta projekto dokumentacija. |
|--|--|

Sistemos projektavimas prasideda nuo kuriamos sistemos struktūros apibrėžimo, kurio metu komandai vadovauja kūrimo vadovas. Pirmo ciklo struktūrinis projektavimas prasideda nuo koncepcinio modelio, kuris yra startavimo metu sudaryto modelio detalizavimas, sukūrimo. Struktūrinio projektavimo metu komanda identifikuoja, iš kokių komponentų bus sudaryta sistema, priskiria šiems komponentams unikalius vardus, apibrėžia komponentų tarpusavio sąveiką. Komanda turi nuspręsti, kaip sistemos funkcionalumas bus padalintas tarp identifikuotų komponentų. Kiekvienam komponentui yra sukuriamos naudojimo diagramos. Komanda dar kartą patikslina, kurie sistemos komponentai bus realizuoti konkrečiame cikle ir sudaro einamojo plano projektavimų darbų sąrašą. Vadovaujami komandos lyderio, komandos nariai pasiskirsto projektavimo darbais.

Kitas projektavimo žingsnis yra apibrėžti, kokiais standartais bus vadovaujama projektavimo ir realizavimo metu. Šiai veiklai vadovauja proceso vadovas. Projektavimo metu komanda gali naudoti įvairiausias standartus:

- Vardų sudarymo susitarimai. Vienas iš pirmųjų standartų, kurį komanda turi sudaryti/apibrėžti yra vardų suteikimo standartas. Visi naudojami vardai turi būti sudaryti pagal vienodą schemą. Tai žymiai palengvins inžinierių darbą inspektavimo metu ir pagerins bendrą sistemos suprantamumą. Vardų sudarymo susitarimai turi būti uždokumentuoti ir naudojami visų komandos narių. Vardų sudarymo susitarimai apima:
 - hierarchinę programų pavadinimų schemą (pvz., sistema, produktas, komponentas, modulis, objektas);
 - kintamųjų, parametrų, metodų ir failų pavadinimų sudarymo taisykles;
 - vardų suteikimo ir keitimo procedūrą.
- Interfeisų formatas apibrėžia komponentų formatą ir tūrinį. Šis standartas apima interfeisų apipavidalinimą, didžiausią leistiną parametrų skaičių, nurodymus, kurie parametrai yra skirti kintamiesiems, klaidų kodams, vidinėms sąlygoms ir pan.;
- Sistemos ir klaidų pranešimų standartai. Visi sistemos klaidų pranešimai turi būti tinkamai apipavidalinti ir suprantami sistemos vartotojams;
- Defektų standartai, TSP siūlo naudoti PSP defektų standartą (18 lentelė). Šiame standarte pagrindiniai klaidų tipai turi kodus 10, 20, ..., 100. Toks numeravimas leidžia kiekvienai komandai susikurti naujus defektų potipius pagal savo poreikius. Pavyzdžiui, jeigu

peržiūros metu dažnai pasitaiko klaida (trūksta kabliataško sakinio gale), ją galima išskirti į atskirą potipį ir priskirti jai unikalų kodą (pvz., 21);

19 lentelė. PSP defektų standartas

Tipo kodas	Tipo pavadinimas	Aprašymas
10	Dokumentacija	Komentarai, pranešimai
20	Sintaksė	Gramatinės, spausdinimo klaidos, punktuacija, instrukcijų formatas
30	Surinkimas, versijavimas	Pakeitimų valdymas, bibliotekų ir versijų kontrolė
40	Priskyrimas	Kintamųjų/metodų apibrėžimai, pasikartojantys vardai, galiojimo sritys, režiai
50	Interfeisas	Metodų kvietimai ir nuorodos, įvedimas/išvedimas
60	Tikrinimai	Klaidų pranešimai, neteisingos sąlygos
70	Duomenys	Struktūra, turinys
80	Funkcija	Logika, rodyklės, ciklai, rekursija, skaičiavimai, funkcijų defektai
90	Sistema	Konfigūracija, laiko ribojimai, atminties naudojimas
100	Aplinka	Projektavimo, kompiliavimo, testavimo ir kitos palaikymo problemos

- Kodo eilučių (LOC) skaičiavimo standartai nusako programų dydžio skaičiavimo susitarimus (vienodi programiniai įrankiai, ar skaičiuoti tuščias eilutes, komentarų eilutes ir pan.);
- Projektavimo dokumentacijos standartizavimas apima projektinių specifikacijų struktūrą ir apipavidalinimo taisykles. Standartas užtikrina, kad visų sistemos dalių projektinės dokumentacijos bus sudarytos pagal vienodas taisykles ir joms bus taikomos vienodos apipavidalinimo taisyklės, kas palengvins inžinierių darbą ir leis lengviau suprasti kiekvienos specifikacijos turinį. Kartu šie standartai padeda užtikrinti specifikacijų pilnumą (kad visos reikalingos dalys bus įtrauktos į specifikaciją).

Naudodama pasirinktus ir apibrėžtus standartus ir vadovaujama kūrimo vadovo, komanda pradeda projektavimo darbus. Pirmiausiai komanda projektuoja vidinius ir išorinius sistemos interfeisus, duomenis ir duomenų struktūras, naudojamas informacijos tarp sistemos komponentų apsikeitimui, ir vidines sistemos duomenų struktūras.

Toliau komandos nariai projektuoja atskirus sistemos komponentus. Projektinė specifikacija apima tik bendros komponentų struktūros aprašymą, detalesnis projektavimas yra atliekamas jau realizavimo fazėje (žr. 2.7 skyrių).

Kai yra paruoštos sistemos komponentų ir jų interfeisų specifikacijos, komanda pradeda sudarinėti sistemos integravimo testavimo planą. Integracijos testavimas yra skirtas įsitikinti, kad visi vidiniai sistemos interfeisai yra suderinami tarpusavyje ir yra realizuoti be klaidų. Integracijos testavimo planas apima testavimo metodikų ir būdų parinkimą, testavimo apimties nustatymą, testų, reikalingų kiekvieno interfeiso korektiškumui patikrinti, parinkimą ir planavimą. Sukurtas integracijos testavimo planas yra peržiūrimas.

Kūrimo vadovas yra atsakingas už galutinio projektinės specifikacijos dokumento sudarymą. Visos atskiros projektinės specifikacijos dalys, kurias sukūrė skirtingi inžinieriai, yra surenkamos, suderinamos tarpusavyje, papildomos ir apiforminamos pagal pasirinktus standartus. Galutinis projektinės specifikacijos variantas yra peržiūrimas, peržiūroje dalyvauja visi komandos nariai. Yra atliekamas projektinės specifikacijos ir integravimo testavimo plano inspektavimas. Inspektavimas vyksta pagal INS scenarijų (žr. 2.4 skyrių).

Aptarimo metu komandos nariai aptaria projektavimo proceso eigą, iškilusias problemas, užpildo proceso gerinimo pasiūlymo formas (PIP), atnaujina projekto dokumentaciją.

2.7. Realizavimo (IMP) scenarijus

Reagavimo proceso tikslas yra sukurti kokybišką produktą, atitinkantį visus reikalavimus, už nustatytą laiką. Realizavimo scenarijus apima detalų projektavimą, programavimą, inspektavimą ir modulių testavimą.

20 lentelė. TSP IMP scenarijus

Scenarijaus paskirtis	Apibrėžti sistemos realizavimo veiklas.
Pradžios kriterijai	<ul style="list-style-type: none"> • Kūrimo strategija yra sudaryta; • Projekto planas yra suderintas su vadovybe; • Atliktas reikalavimų specifikacijos inspektavimas, specifikacija yra patvirtinta užsakovo; • Sistemos projektinė dokumentacija yra sukurta ir praėjo inspektavimą.
Bendra informacija	Realizavimo proceso tikslas – sukurti produktą, kuris: <ul style="list-style-type: none"> • Pilnai atitinka reikalavimų ir projektinę specifikacijas; • Atitinka visus nustatytus standartus; • Yra peržiūrėtas, inspektuotas ir ištestuotas (modulių testavimas).
Veikla	Aprašymas
Realizavimo darbų planavimas	Komanda, vadovaujama kūrimo vadovo, sudaro sistemos realizavimo darbų sąrašą. Darbai yra paskirstomi tarp komandos narių, sudaromas realizavimo darbų planas. Pildomos SUMP ir SUMQ formos.

Detalus projektavimas	<ul style="list-style-type: none"> • Komandos nariai atlieka sistemos komponentų (modulių) detalu projektavimą; • Planuojamas modulių testavimas; • Detalus projektas yra peržiūrimas (peržiūroje dalyvauja visi komandos nariai); • Pildomos defektų (LOGD) ir laiko (LOGT) fiksavimo formos.
Detalaus projekto inspektavimas	<ul style="list-style-type: none"> • Inspektavimo tikslas – surasti klaidas ir nesuderinamumus sistemos testavimo plane prieš pradėdant testų kūrimo darbus; • Inspektavimas vyksta pagal INS scenarijų.
Programavimas	<ul style="list-style-type: none"> • Komandos nariai rašo kodą, kuris realizuoja detalu projektą; • Kiekvienas komandos narys individualiai peržiūri savo parašytą kodą; • Pildomos defektų (LOGD) ir laiko (LOGT) fiksavimo formos.
Kodo inspektavimas	<ul style="list-style-type: none"> • Inspektavimo tikslas – surasti klaidas ir nesuderinamumus sistemos testavimo plane prieš pradėdant testų kūrimo darbus; • Inspektavimas vyksta pagal INS scenarijų.
Modulių testavimas	Naudodamasi testavimo planu, komanda sukuria testus ir atlieka sistemos modulių testavimą. Testavimas vyksta pagal UT scenarijų (žr., 2.8 skyrių).
Komponento išleidimas ir aptarimas	<ul style="list-style-type: none"> • Kiekvienas sistemos komponentas, kuris yra tinkamai realizuotas ir ištestuotas, yra perduodamas konfigūracijos valdymui; • Palaikymo vadovas prideda komponentą prie bazinio komplekto. Nuo šiol visi komponento pakeitimai vyksta tik per pakeitimų prašymo formas (CCR).
Realizavimo proceso aptarimas	<ul style="list-style-type: none"> • Aptarti realizavimo proceso eigą; • Užpildyti proceso gerinimo pasiūlymų formas; • Atnaujinti projekto dokumentaciją.
Pabaigos kriterijai	<ul style="list-style-type: none"> • Sukurti ir ištestuoti visi sistemos komponentai; • Užpildytos detalaus projekto ir kodo inspektavimo formos (INS); • Duomenys apie atliktus darbus, sugaištą laiką, padarytus ir ištaisytus defektus yra užpildyti į atitinkamas TSP formas; • Atnaujinta projekto dokumentacija.

Pirmas realizavimo scenarijaus žingsnis – detalus darbų planavimas. Komanda, vadovaujama kūrimo vadovo, sudaro realizavimo darbų sąrašą, patikslina ir papildo startavimo metu sudarytą planą. Kartu yra pildomos SUMP (plano santraukos) ir SUMQ (kokybės plano) formos. Visi identifikuoti darbai yra paskirstomi tarp komandos narių.

Toliau komandos nariai atlieka detalu atskirų sistemos komponentų projektavimą. Šio projektavimo metu yra dokumentuojama, kaip bus realizuotas kiekvienas iš sistemos komponentų. Užbaigus komponento (modulio) projektavimą yra atliekamas šio modulio testavimo planavimas.

Modulio testavimas yra skirtas patikrinti modulio funkcionalumo ir realizavimo korektiškumą. T.y. testavimo metu yra patikrinama, ar modulis atlieka visas jam priskirtas funkcijas, ir ar tos funkcijos yra atliekamos korektiškai. Sudarytas detalus projektas yra peržiūrimas ir inspektuojamas. Inspektavimas vyksta pagal INS scenarijų (žr. 2.4 skyrių).

Kai jau yra atliktas detalaus projekto inspektavimas ir visos rastos klaidos yra ištaisytos, komanda pradeda programavimo darbus. Kiekvienas komandos narys atlieka jam paskirtas realizavimo užduotis, peržiūri savo parašytą kodą, pildo laiko fiksavimo (LOGT) ir defektų (LOGD) formas. TSP siūlo skirti kodo peržiūrai nuo 50 iki 75% nuo laiko, skirto kodavimui. Kiekvienos peržiūros metu patariama paskirti skirtingų kodo aspektų tikrinimui (kintamųjų inicializavimas, punktuacija, rodyklės, metodų iškvietimo eiliškumas ir t.t.). Tam, kad kodo peržiūros būtų maksimaliai efektyvios, inžinierius turi atidžiai peržiūrėti visą kodą, per vieną valandą peržiūrėdamas ne daugiau 200 kodo eilučių. Po peržiūros kodas yra kompiliuojamas. Visos rastos klaidos yra ištaisomos. Kai komponentas yra baigtas kurti ir yra peržiūrėtas, yra atliekamas to modulio inspektavimas. Inspektavimas vyksta pagal INS scenarijų (žr. 2.4 skyrių).

Kiekvienas sukurtas sistemos komponentas (modulis) yra testuojamas. Modulio testavimas vyksta pagal UT scenarijų (žr., 2.8 skyrių). Kai modulio testavimas yra atliktas ir visos aptiktos klaidos yra ištaisytos, modulis yra perduodamas kokybės vadovui, kuris dar karta peržiūri modulį ir patikrina, ar jis atitinka visus jam nustatytus kriterijus. Jeigu sukurto modulio kokybė yra pakankamai aukšta, modulis yra perduodamas palaikymo vadovui, kuris įtraukia šį modulį į konfigūracijos valdymo procesą ir į bazinį komplektą. Nuo šiol visi modulio pakeitimai vyksta tik per pakeitimų prašymų (CCR) formas. Jeigu modulio kokybė yra prasta, kokybės vadovas grąžina jį pakartotinai peržiūrai arba tiesiai kūrėjui, kad šis perdarytų modulį iš naujo.

Aptarimo metu komandos nariai aptaria realizavimo proceso eigą, iškilusias problemas, užpildo proceso gerinimo pasiūlymo formas (PIP), atnaujina projekto dokumentaciją.

2.8. Modulių testavimo (UT) scenarijus

Nors didžioji dalis klaidų buvo rasta ir ištaisyta peržiūrų ir inspektavimo metu, modulių testavimas yra skirtas surasti visas (arba beveik visas) klaidas, kurios dar liko sukurtame modulyje. Jeigu modulių testavimas aptinka mažiau negu 5 klaidas tūkstančiui kodo eilučių, tai reiškia, kad vėlesnio testavimo ir modulio naudojimo metu nebus rasta praktiškai ne vieno defekto.

21 lentelė. TSP UT scenarijus

Scenarijaus paskirtis	Apibrėžti modulių testavimo veiklas.
Pradžios kriterijai	<ul style="list-style-type: none"> • Sistemos komponentas (modulis) yra sukurtas, peržiūrėtas, sukompiliuotas ir inspektuotas.
Bendra informacija	<ul style="list-style-type: none"> • Modulių testavimas turi būti atliekamas kruopščiai, nes tam, kad surasti praleistus defektus vėlesnėse stadijose, prireikia žymiai daugiau laiko (nuo 5 iki 40 val. vienam defektui); • Kruopščiai atliekamas modulių testavimas gali rasti visas klaidas tinkamai sukurtame modulyje; • Labai svarbu atidžiai peržiūrėti ir dar kartą ištestuoti visus pataisymus.
Veikla	Aprašymas
Modulių testų sukūrimas	<ul style="list-style-type: none"> • Sukurti visus reikalingus testus ir paruošti testavimo duomenis; • Peržiūrėti testus ir testavimo duomenis, įsitikinti, kad juose nėra klaidų.
Modulių testavimas	<ul style="list-style-type: none"> • Atlikti visus modulių testus ir ištaisyti rastus defektus.
Naudojimo scenarijų testavimas	<ul style="list-style-type: none"> • Ištestuoti visus modulių naudojimo scenarijus.
Logikos testavimas	<ul style="list-style-type: none"> • Ištestuoti kiekvieną loginę šaką; • Ištestuoti visas sąlyginio sakinio šakas (ištestuoti kiekvieną įmanomą sąlygą); • Patikrinti kiekvieno ciklo įėjimo ir išėjimo taškus ir sąlygas; • Ištestuoti nuorodų/rodyklių priskyrimo korektiškumą, tuščios reikšmės (null) problemas.
Interfeisų testavimas	<p>Kiekvienam interfeisui patikrinti:</p> <ul style="list-style-type: none"> • gražinamų reikšmių teisingumą visiems galimiems įėjimo duomenų variantams; • duomenų tipų atitikimą; • klaidingų situacijų apdorojimo atitikimą specifikacijai.
Klaidų testavimas	<p>Ištestuoti, kaip modulis apdoroja įvairias klaidas:</p> <ul style="list-style-type: none"> • neteisingos reikšmės ir duomenų tipai; • perpildymo (angl. overflow) ir reikšmės išnykimo (angl. underflow) problemos; • duomenų tipų konvertavimas.
Kintamųjų testavimas	<p>Ištestuoti kiekvieno kintamojo ir parametro reikšmes:</p> <ul style="list-style-type: none"> • nominalias, maksimalias ir minimalias reikšmes; • reikšmės, kurios yra didesnės už galimas reikšmes, nurodytas specifikacijoje; • reikšmės, kurios yra mažesnės už galimas reikšmes, nurodytas specifikacijoje;

	<ul style="list-style-type: none"> • korektiškas modulio veikimas su nuline reikšme, be įeigos duomenų, su klaidingomis įeigos duomenimis ir t.t.
Įrangos testavimas	Ištestuoti įrenginių veikimo korektiškumą: <ul style="list-style-type: none"> • spausdintuvų, monitorių, sensorių ir pan.; • normalų ir klaidingą veikimą (spausdintuve nėra popieriaus, spausdintuvas išjungtas ir pan.).
Kiti testai	Ištestuoti kitas svarbias modulio vietas: <ul style="list-style-type: none"> • buferių dydį, persipildymą; • bendras užduočių atlikimo ir atsako laikas; • datos ir datų išskaičiavimas; • saugumas, suderinamumas, instaliuojamumas, atstatomumas .
Pabaigos kriterijai	<ul style="list-style-type: none"> • Modulis yra ištestuotas, visi rasti defektai ištaisyti; • Modulio testavimo duomenys yra įrašyti į LOGD, LOGT ir LOGTEST formas.

Modulių testų sukūrimas gali vykti arba po realizavimo, arba lygiagrečiai modulio realizavimo darbams. Šiame žingsnyje yra kuriami visi planuoti testai, paruošiami testavimo duomenys (pvz., testų pradiniai/įeigos duomenys).

Sekantis žingsnis yra pats modulių testavimas. Testavimas turi vykti pagal sukurtą modulių testavimo planą ir susidaryti iš šių dalių:

1. *Naudojimo scenarijų testavimas.* Kiekvienas iš aprašytų sistemos naudojimo scenarijų yra išbandomas, atliktų scenarijų rezultatas yra palyginamas su laukiamais rezultatais.
2. *Logikos testavimo metu* yra apeinamos visos loginės modulių šakos, patikrinami visi sąlyginiai sakiniai, visi ciklų žingsniai ir išėjimo iš ciklų korektiškumas, rodyklių priskyrimas, atlaisvinimas, tuščios reikšmės (null) naudojimo korektiškumas.
3. *Interfeisų testavimas* apima gražinamų rezultatų korektiškumo visoms įmanomoms įeigos duomenų kombinacijoms, duomenų tipų atitikimo ir suderinamo, klaidingų interfeisų įeigos duomenų apdorojimo atitikimo specifikacijai tikrinimą.
4. *Klaidų testavimas* apimą įvairių klaidų apdorojimo teisingumą:
 - neteisingi duomenų tipai ir reikšmės;
 - persipildymas (angl. overflow) ir duomenų trūkumas (angl. underflow);
 - tipų konvertavimas. Pavyzdžiui, kaip modulis susidoroja su situacijomis, kai jam yra paduodami skirtingi skaičių duomenų tipai: integer, long integer, unsigned integer, real ir pan.

5. *Kintamųjų testavimas* susideda iš:

- nominalių, maksimalių ir minimalių reikšmių testavimo;
- reikšmių, kurios yra didesnės/mažesnės už galimas reikšmes, nurodytas specifikacijoje, testavimas;
- modulio veikimo testavimas su nuline reikšme, be įeigos duomenų, su klaidingomis įeigos duomenimis ir t.t.

6. *Įrangos testavimas* apima modulio darbo su išoriniais įrenginiais korektiškumo tikrinimą. Šis testavimas apima normalų darbą su išorinę įrangą (pvz., ataskaitų spausdinimas) ir klaidingą veikimą (pvz., spausdintuve pasibaigė popierius arba spausdintuvas yra išjungtas).

7. *Kiti testai*. Taip pat yra testuojamos kitos svarbios modulio savybės, pavyzdžiui:

- buferių dydis, dydžio apribojimai ir persipildymas;
- bendras užduočių atlikimo ir atsako laikas;
- duomenų kiekis ir jo ribojimai;
- datos ir datų skaičiavimas;
- saugumas, suderinamumas, , instaliuojamumas, atstatomumas ir t.t.

Testavimą atliekantys komandos nariai testavimo darbų trukmę fiksuoja LOGT ir LOGTEST formose, rastus defektus įrašo į LOGD formą.

2.9. Testavimo (TEST) scenarijus

Pagrindinis TSP testavimo tikslas yra įvertinti produkto kokybę, o ne ištaisyti produkto defektus, nes beveik visi defektai turėjo būti ištaisyti iki testavimo pradžios. Nors, žinoma, visi rasti testavimo metu defektai turi būti ištaisyti. Žemos kokybės produktų testavimas užsitęsia, ir jo metu, gana didelis skaičius paliktų defektų, tikriausiai, taip ir liks nerastas. TSP testavimą sudaro 4 pagrindinės dalys: sistemos surinkimas, integravimo testavimas, sistemos testavimas ir defektų taisymas. Kiekvienai iš šių dalių yra skirtas atskiras scenarijus.

22 lentelė. TSP TEST scenarijus

Scenarijaus paskirtis	Apibrėžti testavimo proceso veiklas.
Pradžios kriterijai	<ul style="list-style-type: none">• Yra sudarytos ir inspektuotos reikalavimų projektinė specifikacijos;• Visi kuriamos sistemos komponentai yra sukurti, inspektuoti ir sėkmingai praėjo modulių testavimą.

Bendra informacija	<ul style="list-style-type: none"> • Kai defektai yra randami sistemos surinkimo, integravimo ar sistemos testavimo metu, kokybės vadovas turi nuspręsti, ar yra tikslinga tęsti testavimo darbus, ar sistema turi būti grąžinta į kūrimo/klaidų taisymo fazę; • Visi rasti defektai yra fiksuojami LOGD formoje, komanda aptaria visus rastus defektus ir nustato, kur sistemoje galėjo likti panašių defektų, kaip šiuos defektus rasti ir ištaisyti, kaip išvengti panašių defektų ateityje.
Veikla	Aprašymas
Sistemos surinkimas	<p>Šiame žingsnyje yra naudojamas TEST1 scenarijus.</p> <ul style="list-style-type: none"> • Komanda patikrina, ar visi reikalingi sistemos komponentai yra sukurti ir atitinka specifiкуotus interfeisus; • Identifikuoja visus trūkstamus komponentus arba tuos komponentus, kuriuose liko neištaisytų klaidų. Šio žingsnio tikslas įvertinti, ar sistema gali būti surinkta iš esamų komponentų; • Surenka sistemą. Mažiems produktams šiame žingsnyje užtenka kompiliavimo ir surišimo (angl. linking); • Visi rasti defektai fiksuojami LOGD formoje.
Integravimo testavimas	<ul style="list-style-type: none"> • Šiame žingsnyje yra naudojamas TEST2 scenarijus; • Komanda patikrina surinktos sistemos pilnumą, įsitikina, kad visos reikalingos dalys yra įtrauktos į surinktą sistemą; • Atliekamas integravimo testavimas; • Visi rasti defektai fiksuojami LOGD formoje.
Sistemos testavimas	<ul style="list-style-type: none"> • Šiame žingsnyje yra naudojamas TEST3 scenarijus; • Sistemos testavimo metu komandai vadovauja kūrimo vadovas; • Atliekamas sistemos testavimas normaliomis ir streso sąlygomis; • Testuojamas sistemos instaliavimas, atsparumas trykiams, atstatomumas; • Pildomos LOGTEST ir LOGD formos;
Testavimo metu aptiktų defektų šalinimas	<ul style="list-style-type: none"> • Šiame žingsnyje yra naudojamas TESTD scenarijus; • Komandos nariai šalina rastus defektus arba grąžina sistemą į kūrimo fazę; • Pildomos LOGT formos.
Testavimo proceso aptarimas	<ul style="list-style-type: none"> • Aptarti testavimo proceso eigą; • Užpildyti proceso gerinimo pasiūlymų formas; • Atnaujinti projekto dokumentaciją.
Pabaigos kriterijai	<ul style="list-style-type: none"> • Sistema yra surinkta, integruota ir ištestuota; • Kiekvienam testui yra užpildytos LOGD ir LOGTEST formos; • Duomenys apie atliktus darbus, sugaištą laiką, padarytus ir ištaisytus defektus yra užpildyti į atitinkamas TSP formas; • Atnaujinta projekto dokumentacija.

Po kiekvieno scenarijaus žingsnio kokybės vadovas, remiantis gautais duomenimis (pvz., rastų defektų skaičius ir rimtumas), nusprendžia, ar verta tęsti testavimą, ar produktas turi būti gražintas kūrėjams ir perdarytas/pataisytas.

Pirmas testavimo žingsnis – sistemos surinkimas (naudoti TEST1 scenarijų, žr. 2.9.1 skyrių). Komanda patikrina, ar visi reikalingi sistemos surinkimui komponentai yra sukurti ir ar jie yra prieinami, ar visų komponentų interfeisai atitinka specifikacijas. Toliau komandos nariai surenka visus komponentus į vieną sistemą ir sukuria visus reikalingus integravimo ir sistemos testus.

Antras žingsnis – integravimo testavimas (naudoti TEST2 scenarijų, žr. 2.9.2 skyrių). Šiame žingsnyje komanda dar karta patikrina surinktos sistemos pilnumą ir atlieka integravimo testus.

Trečias testavimo scenarijaus žingsnis – sistemos testavimas (naudoti TEST3 scenarijų, žr. 2.9.3 skyrių). Šiame žingsnyje komandai vadovauja kūrimo vadovas. Komanda atlieka sistemos testavimą normaliomis ir stresinėmis sąlygomis, bei sistemos patikimumo ir našumo testus pagal reikalavimų sudarymo metu sudarytą planą. Taip pat atliekamas regresinis testavimas.

Atlikus visus reikalingus testus komanda ištaiso visus rastus defektus, atlieką pataisytų vietų peržiūrą ir pakartotinį testavimą. Šiam žingsniui yra naudojamas TESTD scenarijus (žr., 2.9.4 skyrių).

Aptarimo metu komandos nariai aptaria testavimo proceso eigą, iškilusias problemas, užpildo proceso gerinimo pasiūlymo formas (PIP), atnaujina projekto dokumentaciją.

2.9.1. Sistemos surinkimo (TEST1) scenarijus

Sistemos surinkimas skirtas atskirų komponentų surinkimui į bendrą visumą. Šis scenarijus apima komponentų patikrinimą, peržiūrą, sistemos surinkimą ir integravimo ir sistemos testų kūrimą.

23 lentelė. TSP TEST1 scenarijus

Scenarijaus paskirtis	Apibrėžti sistemos surinkimo veiklas.
Pradžios kriterijai	<ul style="list-style-type: none"> • Yra sudarytos ir inspektuotos reikalavimų projektinė specifikacijos; • Visi kuriamos sistemos komponentai yra sukurti, inspektuoti ir sėkmingai praėjo modulių testavimą.
Bendra informacija	<ul style="list-style-type: none"> • Kai defektai yra randami sistemos surinkimo metu, kokybės vadovas turi nuspręsti, ar yra tikslinga tęsti testavimo darbus, ar sistema turi būti gražinta į kūrimo/klaidų taisymo fazę; • Visi rasti defektai yra fiksuojami LOGD formoje.

Veikla	Aprašymas
Sistemos komponentų patikrinimas	Komanda patikrina, ar visi reikalingi komponentai yra sukurti ir pasiekiami. T.y. visi sistemos surinkimui reikalingi komponentai turi būti pasiekiami komandos nariams ir visi elementai (duomenų bazės, įrenginiai ir pan.), reikalingi šių komponentų veikimui yra prieinami komandos nariams.
Sistemos komponentų peržiūra	Komanda peržiūri visus sistemos komponentus, identifikuoja trūkstamus arba netinkamus komponentus, patikrina komponentų interfeisų atitikimą specifikacijoms. Šio peržiūros tikslas yra užtikrinti, kad sistema gali būti surinkta iš esamų komponentų.
Sistemos surinkimas	<ul style="list-style-type: none"> • Komanda aptaria projektavimo metu pasirinktą sistemos surinkimo būdą, dar karta įsitikina, kad jis yra tinkamas kuriamai sistemai ir apima visus sistemos komponentus; • Komanda surenka sistemą iš atskirų komponentų; • Mažoms sistemoms surinkimas gali susidaryti vien iš kompiliavimo ir surišimo (angl. linking); • Visi rasti defektai yra fiksuojami LOGD formoje; • Pildomos laiko fiksavimo formos (LOGT).
Testų kūrimas	<ul style="list-style-type: none"> • Kartu su sistemos surinkimu yra kuriami integravimo ir sistemos testai. Šie testai yra kuriami pagal reikalavimų sudarymo ir sistemos projektavimo sudarytus testavimo planus.
Pabaigos kriterijai	<ul style="list-style-type: none"> • Sistema yra surinkta; • Sukurti integravimo ir sistemos testai; • Duomenys apie atliktus darbus, sugaištą laiką, rastus defektus yra užpildyti į atitinkamas TSP formas.

Pirmas scenarijaus žingsnis yra sistemos komponentų patikrinimas. Šio patikrinimo metu komanda įsitikina, kad visi reikalingi komponentai yra sukurti, pasiekiami ir atitinka specifikuotus interfeisus. Taip pat komanda patikrina, ar visi elementai, reikalingi normaliam šių komponentų veikimui yra sukurti ir pasiekiami komandos nariams. Tokių elementų pavyzdžiai: duomenų bazės, įrenginiai, tvarkyklės (angl. driver) ir pan. Dalis šių elementų gali būti dar nesukurta (jie bus sukurti vėlesnių ciklų metu), tokiu atveju yra naudojami kamščiai (angl. stub) – programos, imituojančios įrenginių ar sistemos komponentų veikimą.

Antras žingsnis – sistemos komponentų peržiūra. Šiame žingsnyje komandos nariai patikrina visų komponentų interfeisus ir įsitikina, kad jie yra suderinami tarpusavyje, identifikuoja netinkamus surinkimui komponentus. Šio žingsnio tikslas yra įsitikinti, kad sistema (arbo jos dalis) gali surinkta iš esamų komponentų.

Trečio scenarijaus žingsnio metu komanda dar kartą aptaria pasirinktą sistemos surinkimo būdą, įsitikina, kad jis yra tinkamas kuriamai sistemai. Sistema gali būti surenkama keliais būdais:

1. Didelio sprogimo būdas – visi komponentai yra sujungiami vienu žingsniu. Šis būdas yra trumpiausias laiko prasmė, bet klaidų paieška ir šalinimas atima labai daug laiko.
2. Komponentų sujungimas po vieną – yra ilgiausias sistemos surinkimo būdas. Tačiau šiuo atveju klaidų diagnozavimas ir taisymas yra daug paprastesnis.
3. Klasterių prijungimas. Prie jau surinktos sistemos dalies yra prijungiama tam tikro tipo komponentų grupė. Pavyzdžiui:
 - visi komponentai, dirbantys su failais/duomenų bazėmis;
 - visi komponentai, skirti tiesioginiam bendravimui su vartotojais.

Komanda surenka visus sukurtus komponentus į vieną darnią sistemą. Mažų produktų surinkimas gali susidaryti vien iš kompiliavimo ir surišimo (angl. linking). Komandos nariai užpildo laiko fiksavimo formas, visi rasti defektai yra fiksuojami LOGD formoje.

Paskutinis sistemos surinkimo scenarijaus žingsnis yra testų kūrimas. Komandos nariai, remiantis reikalavimų sudarymo ir sistemos projektavimo sudarytais testavimo planais, sukuria visus reikalingus integravimo ir sistemos testus bei paruošia testavimui reikalingus duomenis (testų pradiniai duomenys ir laukiami rezultatai).

2.9.2. Integravimo testavimo (TEST2) scenarijus

Integravimo scenarijus yra skirtas patikrinti, ar visi reikalingi komponentai buvo sujungti sistemos surinkimo metu ir ar šie komponentai sugeba bendrauti tarpusavyje taip, kaip yra nurodyta specifikacijoje.

24 lentelė. TSP TEST2 scenarijus

Scenarijaus paskirtis	Apibrėžti sistemos integravimo testavimo veiklas.
Pradžios kriterijai	<ul style="list-style-type: none"> • Yra sudarytos ir inspektuotos reikalavimų projekcinė specifikacijos; • Visi kuriamos sistemos komponentai yra sukurti, inspektuoti ir sėkmingai praėjo modulių testavimą; • Sistema yra surinkta.
Bendra informacija	<ul style="list-style-type: none"> • Kai defektai yra randami integravimo testavimo metu, kokybės vadovas turi nuspręsti, ar yra tikslinga tęsti testavimo darbus, ar sistema turi būti gražinta į kūrimo/klaidų taisymo fazę; • Visi rasti defektai yra fiksuojami LOGD formoje.
Veikla	Aprašymas
Sistemos peržiūra	<ul style="list-style-type: none"> • Komanda peržiūri surinktą sistemą; • Atliekami testai, skirti patikrinti surinkimo teisingumą ir pilnumą;

Integravimo testavimas	<ul style="list-style-type: none"> • Atliekamas integravimo testavimas; • Visos testavimo veiklos yra fiksuojamos LOGTEST formoje; • Visi rasti defektai yra fiksuojami LOGD formoje.
Regresinis testavimas	Regresinis testavimas nėra atliekamas pirmo ciklo metu.
Pabaigos kriterijai	<ul style="list-style-type: none"> • Sistema praėjo integravimo testavimą; • Kiekvienam testui yra užpildytos LOGD ir LOGTEST formos; • Duomenys apie atliktus darbus, sugaištą laiką, padarytus ir ištaisytus defektus yra užpildyti į atitinkamas TSP formas.

Pirmiausiai, komandos nariai peržiūri surinktą sistemą ir patikrina jos pilnumą bei surinkimo teisingumą. Šiame žingsnyje dažniausiai yra naudojami kontroliniai sąrašai, kuriuose yra surašyti visi komponentai, kurie turi būti sistemoje.

Antru žingsniu yra atliekami integravimo testai. Visi testavimo darbai yra fiksuojami LOGTEST formoje, aptikti defektai – LOGD formoje.

Taip pat yra atliekamas regresinis testavimas, skirtas įsitikinti, kad nauji sistemos papildymai nesugriovė anksčiau surinktų komponentų darbo. Pirmo ciklo metu, kai sistema yra surenkama pirmą kartą, regresinis testavimas nėra atliekamas.

2.9.3. Sistemos testavimo (TSET3) scenarijus

Sistemos testavimas yra skirtas patikrinti, ar sukurta sistema tenkina jai iškeltus reikalavimus. Scenarijus apima sistemos testavimą normaliomis ir stresinėmis sąlygomis, o taip pat patikimumo ir našumo testavimą.

25 lentelė. TSP TEST3 scenarijus

Scenarijaus paskirtis	Apibrėžti sistemos testavimo veiklas.
Pradžios kriterijai	<ul style="list-style-type: none"> • Yra sudarytos ir inspektuotos reikalavimų projektinė specifikacijos; • Visi kuriamos sistemos komponentai yra sukurti, inspektuoti ir sėkmingai praėjo modulių testavimą; • Sistema praėjo integravimo testavimą.
Bendra informacija	<ul style="list-style-type: none"> • Kai defektai yra randami sistemos testavimo metu, kokybės vadovas turi nuspręsti, ar yra tikslinga tęsti testavimo darbus, ar sistema turi būti grąžinta į kūrimo/klaidų taisymo fazę; • Visi rasti defektai yra fiksuojami LOGD formoje.
Veikla	Aprašymas
Sistemos testavimas normaliomis sąlygomis	<ul style="list-style-type: none"> • Atliekami sistemos testai, skirti įsitikinti, kad sistemos veikimas normaliomis darbo sąlygomis atitinka sistemos specifikaciją; • Visi rasti defektai fiksuojami LOGD formoje.

Stresinis testavimas	<ul style="list-style-type: none"> • Atliekami sistemos testai, skirti įsitikinti, kad sistemos veikimas perkrovimo metu ir esant sąlygoms, skirtingoms nuo normalių, atitinka specifikaciją; • Visi rasti defektai fiksuojami LOGD formoje.
Sistemos patikimumo testavimas	<p>Testuojama:</p> <ul style="list-style-type: none"> • sistemos instaliavimas (ar pakankamai lengva instaliuoti sistemą ir pan.); • atsparumas trikiams; • atstatomumas (per kiek laiko, įvykus trikiui, sistema gali atstatyti į darbinę būseną); • Pildomos LOGTEST ir LOGD formos;
Sistemos našumo testavimas	Atliekami testai, skirti patikrinti, ar sistemos našumas atitinka specifikaciją.
Regresinis testavimas	Regresinis testavimas nėra atliekamas pirmo ciklo metu.
Pabaigos kriterijai	<ul style="list-style-type: none"> • Sistema yra pilnai ištestuota; • Kiekvienam testui yra užpildytos LOGD ir LOGTEST formos; • Duomenys apie atliktus darbus, sugaištą laiką, padarytus ir ištaisytus defektus yra užpildyti į atitinkamas TSP formas.

Sistemos testavimo metu komanda siekia atsakyti į šiuos klausimus:

- ar sistema tinkamai atlieka visas numatytas funkcijas?
- ar yra pasiekti sistemos kokybės tikslai?
- ar sistema veikia korektiškai esant normalioms sąlygoms?
- ar sistema veikia korektiškai esant nepalankioms sąlygoms?

Pirmas sistemos testavimo žingsnis yra sistemos veikimo testavimas esant normalioms sąlygoms. Yra atliekami testai, skirti įsitikinti, kad sistema gali atlikti jai paskirtas užduotis nustatytomis specifikacijoje sąlygomis (teisingi įeities duomenys, stabilus maitinimas, visa reikalinga įranga veikia nepriekaištingai).

Sekantis žingsnis – sistemos testavimas stresinėmis sąlygomis: dideliai duomenų kiekiai, didelis vartotojų skaičius, elektros maitinimo nutrūkimai, sugedę išoriniai įrenginiai ir t.t.

Sistemos patikimumo testavimo metu yra atliekami testai skirti patikrinti:

1. Ar sistema gali būti įdiegta/suinstaliuota į nurodyta aplinką, ar tai atlikti yra pakankamai paprasta (pvz., ar užsakovo įmonės administratorius sugebės pats suinstaliuoti sistemą)?
2. Kiek sistema yra atspari trikiams? Ar įvykus klaidai sistema nenustos veikti, ar nebus prarasti duomenis ir pan.?

3. Ar po sistema lengvai atsistato po gedimo? Kiek laiko trunka klaidos priežasties diagnozavimas, ištaisymas, normalaus sistemos veikimo atstatymas?

Sistemos našumo testavimas yra skirtas patikrinti kitus nefunkcinius sistemos reikalavimus. Pavyzdžiui, atsako laikas, operacijų atlikimo laikas, duomenų atvaizdavimo tikslumas, maksimalus vienu metu aptarnaujamų vartotojų skaičius ir pan.

Galiausiai yra atliekamas regresinis testavimas, skirtas įsitikinti, kad nauji sistemos papildymai nesugriovė anksčiau surinktų komponentų darbo. Pirmo ciklo metu, kai sistema yra surenkama pirmą kartą, regresinis testavimas nėra atliekamas.

2.9.4. Scenarijus TSETD

Šis scenarijus yra skirtas defektų, rastų testavimo metu, šalinimui. Scenarijus apima defektų šalinimo darbų planavimą, paskirstymą tarp komandos narių, pačius defektų šalinimo darbus ir pakartotinį testavimą.

26 lentelė. TSP TESTD scenarijus

Scenarijaus paskirtis	Apibrėžti testavimo metu aptiktų defektų šalinimo veiklas.
Pradžios kriterijai	<ul style="list-style-type: none"> Atliktas integravimo ir sistemos testavimas; Visi rasti defektai yra įtraukti į LOGD formą.
Bendra informacija	<ul style="list-style-type: none"> Šis scenarijus yra skirtas klaidų, aptiktų testavimo metu šalinimui; Jeigu testavimo metu paaiškėja, kad klaidų yra per daug arba aptiktų klaidų taisymas užims per daug laiko, sistema yra gražinama į kūrimo fazę (komanda atlieka pakartotinį startavimą, klaidų taisymo darbai yra įtraukiami į naujo ciklo planą).
Veikla	Aprašymas
Darbų paskirstymas	<ul style="list-style-type: none"> Komanda peržiūrį rastų defektų sąrašą, paskirsto defektų šalinimo darbus, identifikuoja, kokie testai turi būti įvykdyti tam, kad patikrinti, ar gerai buvo ištaisyti defektai.
Defektų pašalinimas	<ul style="list-style-type: none"> Komandos nariai atlieka jiems paskirtus darbus; Pildomos laiko fiksavimo formos (LOGT).
Testavimas	Atliekami testai, skirti patikrinti atliktų pataisymų korektiškumą ir ar neatsirado naujų klaidų.
Regresinis testavimas	Jeigu reikia, yra atliekamas regresinis testavimas.
Pabaigos kriterijai	<ul style="list-style-type: none"> Visi rasti testavimo metu defektai yra pašalinti; Visos sistemos vietos, kurios buvo modifikuotos, yra ištestuotos; Duomenys apie atliktus darbus, sugaištą laiką, padarytus ir ištaisytus defektus yra užpildyti į atitinkamas TSP formas.

Atlikus visus suplanuotus testavimo darbus ir užpildžius LOGD formas, komanda susirenka ir peržiūri visų rastų defektų sąrašą. Kiekvienas esminis defektas yra aptariamas, identifikuojami defektų veiksmi, paskirstomi darbai. Kiekvienas komandos narys atlieka jam paskirtus darbus ir užpildo laiko fiksavimo formą (LOGT).

Kai visi defektai yra ištaisyti, komanda atlieka pakartotinį testavimą, skiriant ypatingą dėmesį toms sistemos vietoms, kuriose buvo atlikti pakeitimai.

Galiausiai yra atliekamas regresinis testavimas, skirtas įsitikinti, kad nauji sistemos papildymai nesugriovė anksčiau surinktų komponentų darbo.

2.10. Projekto aptarimo (PM) scenarijus

Aptarimas yra paskutinis kiekvieno TSP ciklo etapas. Jo metu komanda aptaria viso ciklo eigą, iškilusias problemas, nusprendžia, ką būtų galima daryti geriau kitame cikle/projekte. Komandos nariai užpildo proceso gerinimo pasiūlymų formas (PIP) ir komandos ir jos narių įvertinimo formas (PEER). Taipogi aptarimo metu komanda įvertina savo efektyvumą ir paruošia ciklo/projekto ataskaitą.

27 lentelė. TSP PM scenarijus

Scenarijaus paskirtis	Apibrėžti projekto aptarimo veiklas.
Pradžios kriterijai	<ul style="list-style-type: none"> • Sistema yra sukurta ir ištestuota; • Visos reikalingos TSP formos yra užpildytos ir surinktos.
Bendra informacija	<p>Aptarimo metu komanda:</p> <ul style="list-style-type: none"> • įvertina komandos ir komandos narių efektyvumą; • nustato, kokius pakeitimus komanda turi padaryti, tam kad pagerinti savo darbą kitame cikle/projekte; • sudaro ciklo/projekto ataskaitą.
Veikla	Aprašymas
Peržiūrėti kūrimo proceso metu surinktus duomenis	<ul style="list-style-type: none"> • Komanda, vadovaujama proceso vadovo, analizuoja ciklo/projekto metu surinktą informaciją, identifikuoja problemines vietas ir proceso gerinimo galimybes; • Komandos nariai užpildo proceso gerinimo pasiūlymų formas (PIP).
Įvertinti komandos ir komandos narių efektyvumą	Komandos nariai, vadovaujami komandos lyderio, įvertina bendrą komandos ir atskirų narių (ypatingai, tų kuriems buvo priskirtos vadovų rolės) efektyvumą, TSP instruktoriaus veiksmus, palaikymo infrastruktūrą. Ar jie buvo efektyvus, ką būtų galima pagerinti?
Paruošti ataskaitą	<ul style="list-style-type: none"> • Komanda, vadovaujama komandos lyderio, sudaro ciklo/projekto ataskaitą. • Komandos lyderis paskirsto darbus, peržiūri, pataiso ir apjungia visas

	<p>ataskaitos dalis į vieną dokumentą.</p> <ul style="list-style-type: none"> • Ataskaita apima: <ul style="list-style-type: none"> ○ komandos darbo efektyvumo pristatymą; ○ projekto plano ir realybės skirtumų analizę; ○ bendrą produkto projekto ir standartų apžvalgą; ○ konfigūracijos valdymo ir palaikymo apžvalgą; ○ proceso ir produkto kokybės pristatymą. <p>Ataskaita turi būti trumpa, orientuota į rezultato pristatymą, pagrįsta projekto duomenimis.</p>
Rolių įvertinimas	<p>Kiekvienas komandos narys užpildo PEER formą, kurioje įvertina savo ir kolegų darbo apimtį (procentais nuo bendro komandos darbų apimties), sudėtingumą ir efektyvumą (balais nuo 1 iki 5).</p> <p>Komandos lyderis atnaujina projekto dokumentaciją.</p>
Pabaigos kriterijai	<ul style="list-style-type: none"> • Kūrimo ciklo rezultate buvo sukurtas kokybiškas produktas, atitinkantis visus iškeltus reikalavimus; • Sukurto produkto konfigūracija yra valdoma (produktas įtrauktas į konfigūracijos valdymą); • Užpildytos PEER ir PIP formos; • Sukurta ciklo/projekto ataskaita; • Plano santraukos (SUMP) ir kokybės plano (SUMQ) formos yra užpildytos sistemai ir visiems jos komponentams; • Atnaujinta projekto dokumentacija.

Aptarimo pradžioje komanda, vadovaujama proceso vadovo, peržiūri ir analizuoja ciklo/projekto metu surinktą informaciją, identifikuoja tas vietas, kur kilo daugiausiai problemų, ir aptaria proceso gerinimo galimybes. Komandos nariai užpildo proceso gerinimo pasiūlymų formas (PIP) ir pateikia jas proceso vadovui.

Toliau, komandos nariai, vadovaujami komandos lyderio, įvertina bendrą komandos ir atskirų jos narių efektyvumą. Komandos narių efektyvumo įvertinimas prasideda nuo komandos lyderio darbo aptarimo, toliau iš eilės yra aptariamoms visos kitos TSP rolės, TSP instruktoriaus darbas ir palaikymo infrastruktūra (įranga, įrankiai ir pan.). Šio aptarimo metu komandos nariai atsako į klausimus:

- Kas konkrečiam komandos nariui sekėsi?
- Kokie buvo pagrindiniai jo darbo trūkumai ir kokios iškilo problemos?
- Ką ir kaip būti galima pagerinti konkrečios rolės vykdytojui?
- Kokie pakeitimai turi būti padaryti jau kito ciklo/projekto metu?

Kitas scenarijaus žingsnis – ciklo/projekto ataskaitos sudarymas. Komandos nariai, vadovaujami komandos lyderio, pasiskirsto darbais, paruošia savo vadovo ir inžinieriaus rolių ataskaitas, peržiūri, pataiso ir apjungia visas ataskaitos dalis į vieną dokumentą. Ataskaita turi būti pagrįsta realiais duomenimis, surinktais ciklo/projekto metu. Ciklo/projekto ataskaita apima šiuos klausimus:

- komandos darbo efektyvumo pristatymą;
- projekto plano ir realybės skirtumų analizę;
- bendrą produkto projekto ir standartų apžvalgą;
- konfigūracijos valdymo ir palaikymo apžvalgą;
- proceso ir produkto kokybės pristatymą.

Paskutinis aptarimo scenarijaus žingsnis – rolių įvertinimas. Komandos nariai užpildo PEER formas (žr. 3 priedą), kuriose įvertina bendrą komandos darbą, savo ir kolegų darbų apimtį, sudėtingumą ir kokybę balais nuo 1 iki 5. Aptarimo pabaigoje komandos nariai atnaujina plano santraukos (SUMP) ir kokybės plano (SUMQ) formas visai sistemai ir kiekvienam jos komponentui, atnaujina projekto dokumentaciją.

2.11. Kassavaitinio susitikimo (WEEK) scenarijus

Kassavaitiniu susitikimų metu komandos nariai atsiskaitinėja, kokie darbai jau yra atlikti, pristato ateinančios savaitės darbus, aptaria iškilusiais problemas.

28 lentelė. TSP WEEK scenarijus

Scenarijaus paskirtis	Apibrėžti kassavaitinio susitikimo veiklas.
Pradžios kriterijai	<ul style="list-style-type: none"> • Susitikime dalyvauja visi komandos nariai; • Visi komandos nariai perdavė planavimo vadovui atnaujintas TASK, SCHEDULE ir WEEK formas; • Planavimo vadovas visus duomenis įvedė į bendrą komandos WEEK formą; • Komandos lyderis paskelbė susitikimo darbotvarkę.
Bendra informacija	<ul style="list-style-type: none"> • Prieš prasidedant susitikimui komandos lyderis surenka iš komandos narių jų pristatymų temas ir paskelbia visiems komandos narams susitikimo darbotvarkę; • Susitikimo metu komandai vadovauja komandos lyderis; • Proceso vadovas dokumentuota susitikimo metu aptartas temas ir trumpas aptarimų santraukas; • Kiekvienas komandos narys pristato jo atliktus rolės ir produkto

	<p>kūrimo darbus;</p> <ul style="list-style-type: none"> Po susitikimo komandos lyderis sudaro ir išplatina komandos nariams susitikimo ataskaitą, prideda susitikimo ataskaitos kopiją prie projekto dokumentacijos.
Veikla	Aprašymas
Darbotvarkės apžvalga	<p>Komandos lyderis pradeda susitikimą ir:</p> <ul style="list-style-type: none"> Paskelbia susitikimo darbotvarkę, klausia komandos narių dėl darbotvarkės papildymų/pakeitimų; Patikrina, ar visi komandos nariai yra pasiruošę susitikimui, jeigu ne, nusprendžia, tęsti ar atidėti susitikimą.
Vadovų ataskaitos	<ul style="list-style-type: none"> Pirmas savo ataskaitą pristato kūrimo vadovas: <ul style="list-style-type: none"> bendros rolės problemos ir/ar neaiškumai; kūrimo vadovo rolės darbų būseną; komandos sprendžiamų problemų ataskaita; produkto suprojektuotos, peržiūrėtos, inspektuotos, realizuotos ir ištestuotos dalys (komponentai). Planavimo vadovas pristato komandos išnaudotas valandas ir „uždirbtą vertę“, palygina jas su planuotais; Proceso vadovas pristato, kaip sėkmingai komandos nariai dirbą pagal apibrėžtą procesą; Kokybės vadovas apžvelgia: <ul style="list-style-type: none"> inspekcijų ir testavimo metų aptiktus defektus; esamas problemas, susijusias su produkto kokybe. Palaikymo vadovas pateikia ataskaita apie konfigūracijos valdymą (CSR forma) ir problemų sekimą (ITL forma). Trumpai apžvelgia, kokie elementai buvo įtraukti į konfigūracijos valdymą, kokie pakeitimai buvo atlikti, kokie elementai šiuo metu įeina į bazinį komplektą.
Inžinierių ataskaitos	<p>Kiekvienas inžinierius pristato savo darbų būseną:</p> <ul style="list-style-type: none"> atidirbtos valandos šią savaitę per visą ciklą palyginus su planu; uždirbta vertė šią savaitę ir per visą ciklą palyginus su planu; laikas, sugaištas konkrečioms užduotims palyginus su planu; užduotis, kurias planuojama atlikti kitą savaitę; darbo valandų skaičius, planuojamas kitai savaitei; iškilusios problemos.
Susitikimo užbaigimas	<p>Komandos lyderis praveda diskusiją likusiomis temomis ir:</p> <ul style="list-style-type: none"> patikrina, ar visos planuotos temos yra aptartos; patikrina, ar visos rizikos ir problemos buvo aptartos; užtikrina, kad visos kitos savaitės užduotys yra identifikuotos ir jų atlikimas yra priskirtas vienam (ar keliems) iš komandos narių; aptaria temas, kurios bus įtrauktos į susitikimo ataskaitą.

Pabaigos kriterijai	<ul style="list-style-type: none"> • Susitikimo ataskaita yra sukurta ir pridėta prie projekto dokumentacijos; • Projekto dokumentacija yra papildyta atnaujintomis TASK, SCHEDULE, WEEK ir konfigūracijos būsenos ataskaitos (CSR) formomis; • Atnaujinta problemų sekimo formos (ITL) kopija yra pridėta prie projekto dokumentacijos.
----------------------------	---

Prieš prasidedant susitikimui komandos lyderis surenka iš komandos narių jų pristatymų temas ir paskelbia visiems komandos narams susitikimo darbotvarkę. Vienas iš komandos narių dokumentuoja susitikimo metu aptartas temas ir priimtus sprendimus.

Susitikimo pradžioje komandos lyderis paskelbia temas, kurios bus aptariamos susitikimo metu, ir klausia komandos narių dėl darbotvarkės pasikeitimų ir/ar papildymų. Taip pat komandos lyderis patikrina, ar visi pranešėjai susirinko į susitikimą ir ar jie yra pasiruošę pristatyti savo ataskaitas.

Toliau seka vadovų (komandos narių, atliekančių TSP vadovų roles) ataskaitos. Pirmas savo ataskaitą pristato kūrimo vadovas. Toliau po vieną atsiskaitinėja visi kiti vadovai:

1. Kūrimo vadovas pristato bendros rolės problemas ir/ar neaiškumus, kūrimo vadovo rolės darbų būseną, pristato šiuo metu komandos sprendžiamų problemų sąrašą, išvardina ir trumpai aptaria per einamąją savaitę suprojektuotas, peržiūrėtas, inspektuotas, realizuotas ir ištestuotas produkto dalis (komponentus).
2. Planavimo vadovas pristato komandos išnaudotas valandas ir „uždirbtą vertę“, palygina jas su planuotais.
3. Proceso vadovas pristato, kaip sėkmingai komandos nariai dirbą pagal apibrėžtą procesą;
4. Kokybės vadovas apžvelgia inspekcijų ir testavimo metų aptiktus defektus ir esamas problemas, susijusias su produkto kokybe.
5. Palaikymo vadovas pateikia ataskaita apie konfigūracijos valdymą (CSR forma) ir problemų sekimą (ITL forma). Trumpai apžvelgia, kokie elementai buvo įtraukti į konfigūracijos valdymą, kokie pakeitimai buvo atlikti, kokie elementai šiuo metu įeina į bazinį komplektą.

Po vadovų savo ataskaitas pristato inžinieriai. Jeigu vienas žmogus kartu atlieka kurio nors vadovo ir inžinieriaus vaidmenis, tai pirmoje dalyje jis pristato savo rolės ataskaitą, o šioje susitikimo dalyje – inžinierinių darbų ataskaitą. Komandos nariai pristato, kiek valandų ir kokiems darbams buvo skirta šią savaitę, uždirbtą vertę per šią savaitę ir bendrą uždirbtą vertę, bei palygina šias reikšmes su planuotomis. Taip pat kiekvienas komandos narys pristato, kokius darbus jis darys

kitą savaitę, kiek laiko planuoja jiems skirti ir aptaria iškilusias problemas su kitais komandos nariais.

Susitikimo pabaigoje komandos lyderis pristato visas likusias darbotvarkėje temas ir, jeigu reikia, praveda šiomis temomis diskusiją. Po diskusijos komandos lyderis patikrina, ar visos darbotvarkėje esančios temos, svarbios rizikos ir problemos buvo aptartos, ar visi komandos nariai žino, kokius darbus jie turės atlikti kitą savaitę.

Po susitikimo komandos lyderis sudaro susitikimo ataskaitą, prideda ją prie projekto dokumentacijos ir išplatina ataskaitos kopijas. Kiekvienas komandos narys turi gauti po vieną susitikimo ataskaitos kopiją.

3. TSP diegimo metodika

Ruošiant TSP diegimo metodiką yra labai svarbu apsispręsti, kokioms komandoms ši metodika bus skirta. Skirtingo dydžio komandoms turi būti taikomos skirtingos diegimo taisyklės. Pavyzdžiui, jeigu komandoje yra daugiau nei 9 žmonės, t.y. daugiau negu yra skirtingų TSP rolių, keli žmonės galėtų pasidalinti vienos rolės atsakomybes. Tačiau, jeigu komandoje yra 4-5 žmonės, rolių paskirstymas turi vykti visiškai kitaip: dabar jau vienas žmogus turi atlikti kelias roles. Jeigu komanda yra geografiškai išskirstyta, jai turi būti taikomas visiškai kitoks bendravimo tarp komandos narių būdas. Pavyzdžiui, dėl to, kad komandos nariai gyvena ir dirba skirtinguose miestuose ar net šalyse, kassavaitiniai TSP susitikimai, tokioje formoje, kokioje yra aprašyti TSP, praranda savo prasmę, nes didelė komandos dalis tiesiog negali juose dalyvauti. Todėl pirmiausia buvo apibrėžtas projekto komandos tipas, kuriam bus skirta TSP diegimo metodika.

Apmokymai yra neatsiejama praktiškai bet kokio diegimo dalis. Norint įdiegti TSP, yra svarbios dvi apmokymų sritys:

- 1) asmeninis programų kūrimo procesas (PSP);
- 2) komandinis programų kūrimo procesas (TSP).

TSP procesas yra grindžiamas asmeninio programų kūrimo proceso (PSP) veiklomis ir atliekamais matavimais. Todėl, didelę įtaką TSP diegimui ir taikymui daro tai, kiek patyrę yra projekto komandos nariai taikydami PSP ir kokia apimtimi yra realu taikyti PSP projekto metu. Pavyzdžiui, jeigu nėra atliekamas laiko matavimas ir nėra pildomos LOGT formos, tampa neįmanoma vykdyti projekto sekimo veiklų, kurias numato TSP.

Sudarant diegimo metodiką buvo remiamasi W.Humphrey knygomis „PSPSM: A Self-Improvement Process for Software Engineers” [Hum06B] ir „How PSP and TSP Can Impact an Organization” [Hum02].

Ruošiant TSP diegimo metodiką, labai daug naudingos informacijos buvo paimta iš W. Humphrey knygos „Introduction to the Team Software Process(SM)“ [Hum99]. Šioje knygoje yra pateikta išanginė TSP versija, skirta TSP pagrindų įsisavinimui ir taikymui universitetiniuose projektuose. TSP aprašyme (angl., tutorial) [HO99] yra pateikti keli TSP praktinio taikymo pavyzdžiai, kuriuose yra trumpai aptariama, kokios komandos ir kokiuose projektuose taikė TSP, kokios problemos iškilo, kokios buvo šių problemų priežastys, kokių rezultatų pavyko pasiekti šiuose projektuose.

R. Narawane ir V. Ranganathan straipsnyje „Applying TSP to very small teams” [NR99] yra pateikta dalis TSP praktinio pritaikymo 3 žmonių komandai. Šiame straipsnyje pateiktas adaptuotas

konkrečiam projektui projektavimo scenarijus (HLD) tapo geru praktinio TSP diegimo ir taikymo pavyzdžiu ir visos metodikos kūrimo atramos tašku. Nors pats scenarijus yra pernelyg supaprastintas ir pateiktoje formoje netinka apibrėžtam komandos tipui, tačiau pats HLD scenarijaus taikymo būdas suteikė nemažai vertingos informacijos.

Taipogi, TSP diegimo metodikos sudarymo metu pravertė Noopur Davis ir Julia Mullaney ataskaita „The Team Software ProcessSM (TSPSM) in Practice: A Summary of Recent Results” [DM03] ir Donald R. McAndrews straipsnis „The Team Software Process (TSPSM): An Overview and Preliminary Results of Using Disciplined Practices” [McA00]. Šiuose šaltiniuose esanti informacija papildė kuriamą metodiką svarbiomis detalėmis.

3.1. Projekto komanda

TSP diegimas ir praktinis taikymas skirtingo dydžio komandoms skiriasi. Be to, proceso diegimui ir taikymui turi įtakos ir tai, ar komanda dirba vienoje patalpoje, ar yra geografiškai išsiskirsčiusi, ar visi komandos nariai kalba viena kalba ir t.t.. Todėl, prieš sudarant diegimo metodiką, reikia apibrėžti tikslinį projekto komandos tipą, t.y. į kokias projekto komandas bus orientuota kuriama TSP diegimo metodika.

Lietuvoje yra vos keletas įmonių, kurias būtų galima priskirti prie didelių ir vidutinių, pasaulio mastu, programinę įrangą gaminančių įmonių. Didžioji dalis Lietuvos įmonių, vertinant pasaulio mastu, yra mažos (iki 100 darbuotojų) ir jų biudžetas paprastai yra gana ribotas. Todėl darbuotojų siuntimas į SEI organizuojamus kursus, kurie vyksta Jungtinėse Amerikos Valstijose, arba SEI konsultantų samdymas nėra tinkamas sprendimas, norint įdiegti TSP šiose Lietuvos įmonėse.

Dauguma Lietuvos projektų komandų yra tradicinės, t.y. visi komandos nariai dirba viename biure, ir kalba viena bendra kalba – lietuvių. Kuriama TSP diegimo metodika bus skirta būtent tokioms komandoms. Lietuvoje vykdomi projektai paprastai nėra dideli, ir dažnai juose vienu metu dirba 5-6 žmonės, o projekto vykdymo laikas yra apie 3-6 mėnesius. Būtent tokiam projekto komandos tipui yra skirta TSP diegimo metodika.

Nėra patartina diegti TSP projekte, kurio metu komanda kuria visiškai naujo tipo produktą. Bet kokios naujos metodikos diegimas savaime jau yra rizika. Bandydas įdiegti TSP visiškai naujo tipo projekte tik dar labiau apsunkins komandos darbą ir ženkliai padidins projekto žlugimo tikimybę.

Taigi, kuriant TSP diegimo metodiką šiame darbe buvo orientuojamasi į tokią projekto komandą:

- komandoje yra 5-6 nariai;

- projekto trukmė yra nuo 3 iki 6 mėnesių;
- visi komandos nariai dirba viename biure;
- visi komandos nariai kalba viena kalba ir tarp jų nėra didelių kultūrinių skirtumų;
- komanda nėra patyrusi TSP ir PSP, t.y. komanda tik pradeda naudoti abu šiuos procesus;
- įmonė nenumato siųsti savo darbuotojų į TSP mokymus arba samdyti TSP konsultantą.

3.2. Apmokymai

Organizacijoje, pradedančioje arba planuojančioje diegti TSP, vienaip ar kitaip yra žmogus, atsakingas už diegimą. Dažniausiai tas žmogus yra entuziastas, kuris ir pasiūlė vadovybei diegti TSP organizacijoje. Šis žmogus geriausiai tinka TSP instruktoriaus vaidmeniui. Tam, kad organizacija galėtų sėkmingai įsidiesti TSP, turi būti išpildytos dvi sąlygos:

- TSP instruktorius turi turėti autoritetą tarp savo kolegų ir oficialius įgaliojimus surinkti TSP projekto komandą ir vadovauti jos nariams. Geriausia būtų, jeigu šis žmogus eitų projekto vadovo arba vyriausiojo specialisto pareigas.
- Instruktoriaus darbotvarkėje TSP diegimo darbams turi būti numatytos oficialios darbo valandos. TSP diegimas yra didelis ir atsakingas darbas, reikalaujantis daug laiko. Sėkmingas TSP diegimas laisvu nuo darbo metu yra praktiškai neįmanomas. Paprastai panašaus pobūdžio diegimai yra apiforminami kaip atskiras projektas. Visai nebūtina kad TSP instruktorius skirtų šiam projektui visą savo darbo laiką, bet reikia turėti omenyje, kad TSP diegimo darbai reikalauja nemažai laiko.

Praktikoje žmogus, atsakingas už TSP diegimą gali kartu atlikti TSP instruktoriaus ir komandos lyderio vaidmenis, bet idealiu atveju tai turėtų būti du skirtingi žmonės. Tai yra svarbu dėl to, kad patys komandos nariai dažnai nepastebi dalykų, kurie yra gerai matomi iš šono.

TSP instruktorius pirmas pradeda studijuoti PSP ir TSP procesus, pats pradeda taikyti PSP. Jis yra atsakingas už komandos narių apmokymą. Komandos narių apmokymai turi prasidėti anksčiau, negu pirmas TSP projektas. Norint sėkmingai įdiegti ir taikyti TSP, visi komandos nariai turi būti apmokyti dviejose susijusiose srityse:

- asmeninis programų kūrimo procesas (PSP), apibrėžiantis individualų komandos narių darbą;
- komandinis programų kūrimo procesas (TSP), apibrėžiantis visos komandos darbą projekto metu.

3.2.1. Asmeninis programų kūrimo procesas (PSP)

Asmeninio programų kūrimo proceso mokymai turi prasidėti mažiausiai du mėnesius prieš pirmo TSP startavimo datą. Du mėnesiai yra reikalingi komandos nariams tam, kad išmokti PSP pagrindus ir priprasti prie kasdienio formų pildymo. Pradžioje komandos nariai skaito knygas apie PSP, lanko seminarus, kuriuose TSP instruktorius supažindina juos su PSP procesu. Vėliau būsimi TSP projekto dalyviai pradeda taikyti PSP savo asmeniniame darbe. Pageidautina kad jau pradedant taikyti PSP visi komandos nariai dirbtų tame pačiame projekte, tačiau ši sąlyga nėra būtina. Pirmų dviejų PSP taikymo mėnesių metu komandos nariai reguliariai susitinka su TSP instruktoriumi, aptaria ir sprendžia išskylančias problemas, instruktorius patikrina, ar komandos narių darbas atitinka PSP procesą (ar vykdomos peržiūros, ar pildomos formos ir pan.). Išankstinis PSP taikymas (dar prieš pradedant taikyti TSP) turi tokius privalumus:

- Sumažinamas naujovių, kurias vienu metu turi įsisavinti komandos nariai, kiekis. Tai leidžia sumažinti komandos narių fizinį ir psichologinį krūvį. Tuo atveju, kai PSP ir TSP procesai pradedami taikyti vienu metu, komanda susiduria su dideliu problemų skaičiumi, kas gali palaužti komandos narių motyvaciją ir tikėjimą taikomų metodų naudingumu.
- Tuo metu, kai komanda pradės taikyti TSP, komandos nariams kils mažiau klausimų ir neaiškumų. Tai leis greičiau spręsti išskylančias problemas ir palengvins TSP diegimo ir taikymo procesą.
- PSP proceso įsisavinimas padidina komandos motyvaciją. Išankstinis PSP taikymas ne tik palengvina proceso įsisavinimą, bet ir įtvirtina komandos narių tikėjimą tuo, kad jiems keliami uždaviniai (PSP ir TSP diegimas ir taikymas) yra įvykdomi, o PSP teigiami taikymo rezultatai (pvz., PSP peržiūrų metu yra randama daugiau defektų, todėl galutiniam produkte jų lieka mažiau), kuriuos turi pajusti būsimi komandos nariai, įtvirtina komandos narių tikėjimą, kad diegiami procesai atneša realią naudą.
- Po dviejų PSP taikymo mėnesių komandos nariai galės įvertinti PSP proceso taikymo sėkmingumą ir trūkumus. Komandos nariai turi galimybę identifikuoti visas gerintinas vietas ir padaryti atitinkamus PSP proceso taikymo pakeitimus (pvz., atsisakyti tam tikrų produkto charakteristikų matavimo, arba atvirškščiai, paaiškės, kad kažkurių matavimų trūksta). Sklandesnis PSP taikymas leidžia sumažinti kylančių problemų, susijusių su PSP proceso taikymu, skaičių, kas palengvins TSP diegimo procesą.

Žinoma, išankstinis PSP taikymas turi ir savo kainą. Apmokymai, duomenų rinkimas ir formų pildymas atima nemažai būsimų komandos narių laiko. Todėl yra reikalingi papildomi resursai ir laikas.

TSP kūrėjai teigia, kad visi komandos nariai turi būti apmokyti PSP ir turi taikyti šį procesą pilna apimtimi [Hum02, Hum06B]. Nors PSP mokymų būtinumas yra akivaizdus, tačiau TSP diegimo proceso pradžioje, PSP taikymas pilna apimtimi gali turėti neigiamų pasekmių. Detalus visų PSP formų pildymas pasiteisina tik tuomet, kai komanda vykdo projektą pagal gerai žinomą metodiką arba modelį. Tik pradėdant taikyti TSP (arba bet kurią kitą metodiką), didelė PSP darbų apimtis ir didelis renkamos informacijos kiekis gali užgožti pagrindines projekto veiklas. Tai gi, reikia nepamiršti, kad svarbiausia yra produkto kūrimas. Net tiksliausias PSP ir TSP nurodymų laikymasis bus nieko vertas, jei projekto pabaigoje nebus sukurtas produktas, atitinkantis visus reikalavimus.

Pirmų TSP projektų metu siūloma taikyti supaprastintą ir adaptuotą PSP procesą: visos pagrindinės PSP veiklos (matavimai, užduočių planavimas, dydžio vertinimas, peržiūros ir t.t.) yra neatsiejama PSP proceso dalis ir todėl privalo likti adaptuotame PSP procese, supaprastinamas yra tik šių veiklų vykdymas. Visos pagrindinės PSP veiklos yra tęstinės (vykdomos viso projekto metu) ir tampriai susijusios tarpusavyje. Pavyzdžiui, atsisakius PSP matavimo veiklos, tampa neįmanomos užduočių planavimo ir jų dydžio vertinimo veiklos, nes pagal PSP šios veiklos naudoja matavimų, kurių buvo atsisakyta, metu surinktus duomenis.

Tačiau, norint išsivinti PSP procesą, pirmųjų projektų metu visai nebūtina akiai sekti proceso aprašymo. Atskirų PSP veiklų vykdymas gali būti supaprastintas ir adaptuotas konkrečiau projekto reikmėms [Hum06B].

Remiantis iškeltais verslo ir komandos tikslais, kurių pirmo TSP projekto metu neturėtų būti daugiau negu du, TSP instruktorius kartu su komandos nariais turi parinkti svarbiausius parametrus ir rinkti tik šių parametrų matavimus. Būtinai yra laiko ir defektų matavimai, visi kiti matavimai turi būti parenkami atsižvelgiant į komandos tikslus. Jeigu komandos nariai turi pakankamai laiko, jie gali parinkti didesnę kiekį projekto/produkto parametrų, kuriuos matuos projekto metu, tai tikrai nepamaišys, nes šie duomenis pravers ateityje (kai komanda pradės kitą TSP projektą, ji galės planuoti savo darbus ir produkto kokybę, remiantis šiais istoriniais duomenimis). Svarbiausia yra ne renkamu duomenų kiekis, o šių duomenų naudingumas.

Vykdam pirmą TSP projektą, komanda turės pildyti laiko formas (LOGT forma) [Hum99]. Tai yra vienas pagrindinių projekto sekimo ir kontrolės duomenų šaltinis. Tačiau, PSP aprašyme yra

reikalaujamas 5 minučių tikslumas įvedamam laikui, tai akivaizdžiai yra pernelyg griežtas reikalavimas, apsunkinantis projekto vykdytojų darbą. Taip pat reikia fiksuoti visas pertraukas, net jei tai yra kelių minučių pokalbis telefonu arba kavos pertraukėlė. Visų panašių nurodymų laikymasis kainuoja labai brangiai laiko ir pastangų prasme. Pirmų TSP projektų metu tokie reikalavimai yra pertekliniai. Komandos nariai gali pasijusti spaudžiami, o laikas sugaištas formų pildymui gali atrodyti jiems nepagrįstai didelis. Dėl šių priežasčių komanda gali apskritai nustoti pildyti PSP formas, o PSP ir TSP procesai gali pasirodyti jiems visiškai nenaudingi.

Todėl, pradedantiesiems naudoti TSP galima pasiūlyti supaprastinti PSP proceso naudojimą. Be to, kad galima rinkti tik dalį visų PSP apibrėžtų duomenų apie pačias svarbiausias projekto ir produkto charakteristikas, kurios yra nustatomos atsižvelgiant į komandos tikslus, papildomai galima sumažinti ir reikalavimus renkamiems duomenims. Svarbiausia, kad, sumažinus reikalavimus, renkami duomenis išliktų tinkami naudoti pagal paskirtį.

Taip, pavyzdžiui, laiko formos pildymo reikalavimai, PSP mokymosi ir pirmo TSP projekto vykdymo metu, galėtų atrodyti taip:

- Vieno įrašo (~8 val.) į LOGT formą leistina paklaida – 0,5 valandos;
- Pildomas bendras laikas, kuris buvo sunaudotas užduoties atlikimui. Pertraukėlės ir jų laikas yra išskaičiuojami iš bendro laiko tik tais atvejais, jeigu bendras pertraukėlių laikas yra didesnis negu 0,5 valandos;
- Forma turi būti pildoma bent kartą per dieną. Tai leidžia padidinti įvedamo laiko tikslumą, bei patiems komandos nariams yra lengviau prisiminti, kokius darbus jie atliko per dieną ir kiek laiko šie darbai truko. Kuo dažniau yra pildoma LOGT forma, tuo tikslesnis yra fiksuojamas laikas. Svarbu, kad laiko forma būtų pildoma bent kartą per dieną.

Tokiu būdu laiko formų pildymo procedūra yra supaprastinama, o renkamų duomenų tikslumas išlieka priimtinas. Leistina vienos dienos darbo laiko įrašų paklaida yra $(0,5/8)*100\%=6,25\%$, turint omenyje, kad paprastai paklaida yra mažesnė už apskaičiuotą, tai gautas tikslumas yra pakankamai geras (net naudojant pačius tiksliausius metodus, retai kada pavyksta planuoti laiką su mažesne paklaida). Tokie duomenys gali būti sėkmingai naudojami projekto darbų ir būsenos sekimui.

3.2.2. Komandinis programų kūrimo procesas (TSP)

Prieš pradedant pirmą TSP projektą, visi komandos nariai turi gerai susipažinti su TSP procesu. Tam TSP instruktorius turi organizuoti papildomus seminarus/kiek jų turėtų būti? Kokio

detalumo ir pan.? – „krūva“ klausimų/, užtikrinti, kad kiekvienam komandos nariui būtų prieinamas TSP modelio aprašymas:

- Scenarijai ir jų aprašai (naudoti rekonstruotą TSP modelį, žr. 2 skyrių);
- TSP formos (žr. 3 priedą). Pirmame TSP projekte rekomenduojama naudoti šias formas (minimalus rinkinys):
 - LOGT – laiko fiksavimo forma;
 - LOGD – defektų fiksavimo forma;
 - TASK – užduočių planavimo forma;
 - SCHEDULE – užduočių tvarkaraštis;
 - SUMQ – kokybės planas (žr., 2 priedą);
 - ROLE – rolių pasirinkimo forma;
 - INS – inspekcijos ataskaita;
 - ITL – problemų sekimo forma;
 - PEER – komandos ir komandos narių įvertinimo forma;
 - PIP – proceso gerinimo pasiūlymų forma;
 - CCR – konfigūracijos pakeitimo prašymas;
 - SUMP – planavimo suvestinė;
 - SUMS – dydžio suvestinė.
- TSP rolių ir veiklų (pvz., konfigūracijos valdymas) aprašymas (galima naudoti Watts S. Humphrey. Knygas „TSPSM-Leading a Development Team“ [Hum05], „Introduction to the Team Software Process(SM)“[Hum99]).

3.3. TSP proceso diegimo apimtis

Kaip ir asmeninio programų kūrimo proceso atveju, TSP veiklos yra stipriai susijusios tarpusavyje. Atskirų TSP veiklų taikymas neatneša teigiamų rezultatų.

Kaip pavyzdį panagrinėkime atvejį, kai pirmame TSP projekte yra taikomi tik startavimo, realizavimo ir aptarimo scenarijai. Reikalavimų sudarymas, projektavimas, programavimas ir testavimas yra atliekami taip, kaip ir iki TSP diegimo. Toks TSP taikymo pavyzdys yra netinkamas nes:

- Neatliekant TSP planavimo, projekto sekimo veiklų ir uždirbtos vertės metodo, yra labai sunku pasakyti, ar projekto eiga atitinka planą ir kokia darbų dalis jau yra atlikta. Komanda neišnaudoja pagrindinių TSP teikiamų pranašumų. Dėl tos pačios priežasties projekto rizikos yra prastai sekamos (arba visai nėra sekamos), laiku nėra imamasi rizikų

poveikio sumažinimo veiksmų. Atsirandančių problemų sprendimas užsitęsia, dėl ko komanda praranda daug brangaus laiko.

- Testavimo metu nėra renkami TSP reikalingi duomenys. Dėl to tampa sunku vertinti iškeltų produktui kokybės tikslų pasiekiamumą.
- Nors realizavimo procesas ir vyko pagal TSP ir jo metu buvo renkami visi reikalingi duomenys, šių duomenų analizė yra labai komplikauta. Pavyzdžiui, neįmanoma atskirti, kurie iš aptiktų defektų buvo padaryti reikalavimų, projektavimo fazėse, o kurie yra realizavimo defektai.
- Dėl reikalingų TSP duomenų apie kūrimo procesą nebuvimo, aptarimų metu yra daug sunkiau identifikuoti problemines vietas ir pateikti proceso gerinimo pasiūlymus. Neefektyvus tampa proceso gerinimas, kuris yra svarbi TSP dalis.

Taigi, TSP diegimas turi apimti visas TSP proceso aprašytas veiklas. Tik taip yra įmanoma pasiekti teigiamų projekto rezultatų. Tačiau, kaip ir PSP atveju, komanda gali pritaikyti TSP procesą projekto reikmėms, adaptuodama TSP veiklų aprašymą. Bet kokie TSP veiklų aprašymo pakeitimai neturi iš esmės pakeisti nei TSP struktūros, nei TSP proceso veiklų esmės. Pavyzdžiai, kaip būtų galima taikyti TSP scenarijus, yra pateikti 3.4 skyriuje.

3.4. TSP praktinio taikymo rekomendacijos

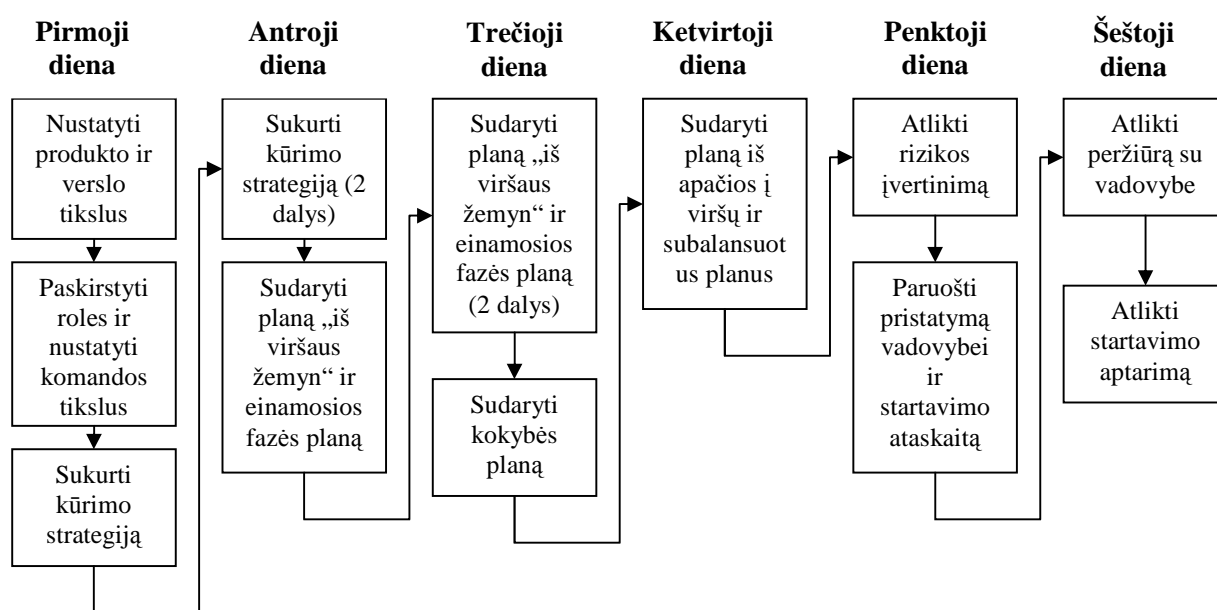
Šiame skyriuje yra pateikti TSP scenarijų adaptavimo, apibrėžtam komandos tipui, ir trijų TSP scenarijų taikymo rekomendacijos. Pirmas pavyzdys parodo, kaip startavimo procesas gali būti adaptuotas konkretaus projekto reikmėms. Antras pavyzdys aprašo bendro plano sudarymo scenarijaus, o trečias kokybės plano sudarymo scenarijaus taikymo rekomendacijas apibrėžtam komandos tipui.

3.4.1. Startavimo proceso adaptavimas

Viena pirmųjų užduočių, kurias turi atlikti TSP instruktorius prieš prasidedant projektui, yra startavimo tvarkaraščio sudarymas. TSP numato standartinę 3 arba 4 dienų startavimo procedūrą. Tačiau produkto kūrimo strategijos ir projekto plano sudarymas dažnai yra nelengvos ir daug laiko atimančios užduotys. TSP aprašyme yra minima, kad komandos nariai turi būti pasiruošę startavimo metu dirbti iki vėlumos, bet kyla klausimas, ar sugebės komanda per trumpą laiką ir esant dideliame darbo krūviui sudaryti kokybiškus strategiją ir planą.

TSP instruktorius, matydamas, kad startavimo scenarijaus veiklos gali užsitęsti (o pirmųjų TSP projektų metu taip tikriausiai ir bus), gali iš anksto numatyti kitokią startavimo tvarkaraštį.

Pagrindinis principas, kuris turi būti išlaikytas – startavimas neturi užsitęsti. Nepažeidžiant šio principo startavimas gali trukti 5-6 dienas. Šešių dienų startavimo procedūra yra parodyta 5 pav. Šioje procedūroje, palyginus su standartine, yra numatyta daugiau laiko strategijos sukūrimui ir planavimo darbams. Strategijos sukūrimui yra skirtas ne tik pirmos dienos vakaras, bet ir antros dienos pirma pusė, projekto plano „iš viršaus žemyn“ yra sudaromas antros dienos antroje pusėje ir trečios dienos pirmoje pusėje. Plano „iš apačios į viršų“ sudarymui ir planų subalansavimui yra skirta visa diena, kai standartinėje procedūroje šiam susitikimui yra numatytas nepilnas pusdienis. Toks startavimo susitikimų išsidėstymas leidžia komandos nariams skirti daugiau laiko planavimui ir sukurti detalų ir kokybišką projekto planą.



5 pav. Šešių dienų TSP startavimo procedūra

Kitas labai svarbus startavimo proceso momentas – pasiskirstymas rolėmis. TSP numato 8 skirtingas roles, neskaitant komandos lyderio. 3.1 skyriuje apibrėžtoje projekto komandoje yra 5 arba 6 žmonės. Nors originaliame TSP komandos lyderis neprisiima jokių papildomų rolių, 5-6 žmonių komandoje jam dažniausiai tenka prisiimti bendravimo su užsakovu vaidmenį. Lieka natūralus klausimas, kaip paskirstyti likusias roles? 29 lentelėje yra pateiktas siūlomas TSP rolių sąrašas 5 ir 6 žmonių komandoms.

29 lentelė. TSP rolių paskirstymas 5 ir 6 žmonių komandose.

5 žmonių komanda	6 žmonių komanda
<ul style="list-style-type: none"> • Komandos lyderis (bendravimo su užsakovu vadovas) • Kūrimo vadovas (projektavimo, testavimo vadovas) • Planavimo vadovas • Kokybės/proceso vadovas • Palaikymo vadovas 	<ul style="list-style-type: none"> • Komandos lyderis (bendravimo su užsakovu vadovas) • Kūrimo vadovas (projektavimo vadovas) • Testavimo vadovas • Planavimo vadovas • Kokybės/proceso vadovas • Palaikymo vadovas

Rolių paskirstymas 5 žmonių komandoje iš esmės atitinka TSPi siūlomą paskirstymą, išskyrus tai, kad komandos lyderis dar prisiima bendravimo su užsakovu rolę. Šis pataisymas buvo įvestas todėl, kad TSPi procese nėra nenumatytas realus užsakovas, todėl nėra ir bendravimo su užsakovu rolės.

6 žmonių komandoje yra išskirta papildoma testavimo vadovo rolė. Ši rolė buvo atskirta nuo kūrimo vadovo rolės dėl šių priežasčių:

- didelė darbų apimtis.
- testavimo darbų tęstinumas.
- geriausias programinės įrangos kūrimo praktikos.

Testavimas yra pradedamas planuoti jau sudarant kuriamos sistemos reikalavimus, integravimo testavimo planas yra sudaromas projektavimo metu, o paskutiniai testavimo darbai baigiasi prieš pat projekto aptarimą. 5 žmonių komandoje kūrimo vadovas atlieka dar projektavimo ir testavimo vadovų rolę. Testavimas yra vykdomas lygiagrečiai projektavimui, o vėliau ir kūrimui, todėl vienam žmogui yra sunku daryti du darbus iš karto (vadovauti produkto projektavimui/kūrimui ir testavimo planavimui/testų kūrimui ir produkto testavimui). Todėl atskira testavimo vadovo rolė leis vadovams daugiau laiko skirti tiek testavimui, tiek projektavimui ir kūrimui.

Kaip jau buvo minėta, testavimo darbai tęsiasi praktiškai nuo projekto pradžios iki pat pabaigos, todėl testavimo vadovas turės darbo viso projekto metu. T.y. viso projekto metu testavimo vadovas neturės „langų“ – projekto dalis, kurioje vadovas neturi jokių su savo role susijusių darbų. Pavyzdžiui, jeigu būtų išskirta atskira projektavimo (o ne testavimo) rolė, tai pasibaigus visiems projektavimo darbams, projektavimo vadovas visai neturėtų darbo, o kūrimo ir testavimo vadovas būtų užsivertęs darbais. T.y. vadovų darbų apimtis būtų nesubalansuota.

Pagal geriausias programinės įrangos praktikas testavimas turi būti planuojamas ir vykdomas atskiros testavimo grupės. T.y. produktą kuria vieni žmonės, o testuoja kiti. Toks atskyrimas turi dvi pagrindines priežastis. Pirma, žmogus, „iš šalies“ testavimo metu paprastai randa daugiau klaidų, negu produkto kūrėjas. Antra, produkto kūrėjui kartais būna sunku pripažinti, kad jo sukurtas produktas turi trūkumų, ir kūrėjas gali paslėpti tam tikrus defektus arba tiesiog neįtraukti jų į testavimo ataskaitą (galbūt, manydamas, kad dar spės juos ištaisyti iki produkto išleidimo). Todėl kito žmogaus atliekamas testavimas užtikrina testavimo rezultatų objektyvumą.

Dėl ribotų projekto resursų (palyginus mažas komandos narių skaičius), TSP procese praktiškai neįmanoma visiškai atskirti kūrimo ir testavimo grupių. TSP testavime dalyvauja visi komandos nariai. Bet kūrimo ir testavimo vadovų rolių atskyrimas priartina procesą prie geriausių praktikų ir įneša savo indėlį į testavimo proceso efektyvumo gerinimą.

3.4.2. Bendro plano sudarymo (LAU4) scenarijaus taikymas

Pirmas bendrojo projekto plano sudarymo žingsnis – sudaryti projekto darbų sąrašą. Šiai užduočiai gerai tinka TSP siūlomas „iš viršaus žemyn“ metodas. Pirmiausia yra nustatomos pagrindinės projekto užduotys. Paprasčiausiu atveju gali būti viena pagrindinė užduotis, pvz., sukurti programinį produktą. Toliau užduotys yra skaidomas į mažesnes, pvz., į atskirų modulių sukūrimo užduotis. Procesas kartojamas tol, kol nebus gautas pakankamas detalumo lygis. Žodis „pakankamas“ šiuo atveju gali būti suprantamas skirtingai, priklausomai nuo projekto tipo ir komandos patirties. Svarbu prisiminti, kad atliekama užduočių dekompozicija neturi leisti iki detalių darbų. Detalus darbų planas yra sudaromas vėliau, ir tam yra skirtas atskiras scenarijus. Todėl, šiame etape, nereikalingas gilinimasis į detales tik atims brangų komandos laiką. Bendru atveju, projekto darbų sąrašą turi sudaryti užduotys, kurių trukmė yra maždaug 5-10 dienų.

Sudarius darbų sąrašą, komanda turi patikslinti šių darbų trukmę. Jeigu komanda tik pradėjo taikyti TSP, ji gali neturėti sukauptų istorinių duomenų, kurie padėtų tiksliau planuoti darbų laiką. Tačiau retai pasitaiko, kad TSP yra pradamas taikyti projekte, kurio metu būtų kuriamas visiškai naujo tipo produktas, tuo labiau kad taip daryti nėra patartina. Komandoje turi būti vienas ar keli patyrę žmonės, kurie jau anksčiau yra dalyvavę panašiuose projektuose. Ir jeigu komanda neturi sukauptų istorinių duomenų, planuojama darbų trukmė gali būti nustatoma remiantis daugiau patyrusių komandos narių ekspertine nuomone.

Turėdama projekto darbų sąrašą ir planuojamą tų darbų trukmę, komanda turi nuspręsti, kaip šios užduotys bus padalintos tarp projekto ciklų. Ciklai neturi būti nei per trumpi, nei per ilgi. Jeigu

ciklas trunka vos vieną savaitę, tai vargu, ar per tą laiką galima suspėti padaryti kažką reikšmingo. Be to, trumpi ir dažni ciklai prideda daug organizacinio pobūdžio darbo (startavimai, planavimai, aptarimai), padaugėja ir konfigūracijos valdymo darbų (kiekvieno ciklo produktai turi būti saugomi ir valdomi, praeitų ciklų produktų keitimas yra sudėtingesnis). Jeigu ciklas trunka per ilgai, tampa sunku sekti ir kontroliuoti projekto eigą, o reikalavimų pasikeitimai gali sugriauti visą ilgai planuotą darbų grafiką.

Jeigu projektas trunka 6 mėnesius, tai optimali vieno ciklo trukmė yra 2-3 mėnesiai. T.y. projektas galėtų būti sudarytas iš 2-3 ciklų. Ciklų trukmės gali skirtis. Pavyzdžiui, tam kad sukurti sistemos prototipą gali reikėti vos kelių savaičių, bet pirma produkto (sistemos) kūrimo fazė gali užtrukti 3 mėnesius ar net daugiau. Bet kokiu atveju, kiekvienam ciklui turi būti nustatomi tikslai, reikalavimai, planuojama trukmė ir numatomi rezultatai (pvz., kurios sistemos funkcijos turi būti realizuotos).

Sudarant projekto darbų grafiką, reikia atsižvelgti į komandos narių užimtumą kituose projektuose, jų planuojamas atostogas, išėigines dienas ir panašius aspektus. Geriausia, kad visų komandos narių darbo savaitė neviršytų 40 darbo valandų. Nes pastovus padidintas krūvis gali neigiamai atsiliiepti žmonių produktyvumui ir susikaupimui, ypatingai tai gali pasijusti arčiau projekto pabaigos. O to pasekmėje gali ne tik vėluoti projekto darbai, bet ir sumažėti kuriamo produkto kokybė.

Darbų sąrašo ir projekto tvarkaraščio peržiūrose turi dalyvauti visi komandos nariai. Svarbu, kad visi iškilę klausimai būtų aptarti komandoje, jeigu kažkurio komandos nario netenkina sudarytas tvarkaraštis (pvz., darbuotojas negalės skirti projektui tiek laiko, kiek yra suplanuota), tai projekto tvarkaraštis turi būti pataisytas. Galiausiai visi komandos nariai turi pritarti sudarytam projekto planui.

3.4.3. Kokybės plano sudarymo (LAU5) scenarijaus taikymas

Kokybės planas turi atspindėti iškeltus projekto, produkto ir komandos tikslus. Šie tikslai yra nustatomi startavimo scenarijaus antro susitikimo (LAU2 scenarijus) metu. Prieš sudarant kokybės planą, komanda turi dar kartą peržiūrėti kokybės tikslus, jeigu tikslas yra nepasiekiamas arba jo pasiekimo neįmanoma pamatuoti, tai tikslas turi būti reformuluojamas arba šio tikslo turi būti atsisakyta [HO99].

Gražūs šūikiai, traktuojami kaip projekto tikslai, neneša jokios naudos, o tik gali įvesti sumaištį, nes užsakovas, vadovybė ir projekto vykdymo grupė gali skirtingai juos suprasti. Dėl to,

produkto pridavimo užsakovui metu, gali kilti nesusipratimų, o gal netgi paaiškės, kad sukurtą produktą teks perdaryti [Hum99].

TSP pateikiamas kokybės planas (žr. 2 priedą) yra gana didelis. Pirmiems TSP projektams jis yra perteklinis. Pradedantiems taikyti TSP yra patariama remiantis kokybės tikslais pasirinkti 1-2 pagrindinius kokybės rodiklius (pvz., patikimumas). Pirmam TSP projektui to pilnai užteks. Kiekvienas iš pasirinktų rodiklių gali turėti kelis matavimus, pavyzdžiui, sisteminių lūžių skaičius per laiko vienetą, vidutinis laikas tarp lūžių, sistemos atstatymo po lūžimo laikas ir pan. Sudarytas kokybės planas gali būti apiforminamas kaip SUMQ forma, tik turėti mažesnę apimtį.

Kiekvienam iš pasirinktų kokybės rodiklių matavimų turi būti nustatomos leistinos reikšmių ribos. Pavyzdžiui, yra nustatyta, kad pirmais programinio produkto eksploatavimo metais aptiktų defektų skaičius negali viršyti 2. Jeigu per nustatytą laiką buvo aptikti 3 defektai, tai kokybės rodiklio vertinimas yra neigiamas, kas tikriausiai reikš, kad vienas iš kokybės tikslų nėra pasiektas.

Kokybės plano peržiūroje turi dalyvauti visi komandos nariai. Peržiūros tikslas yra užtikrinti, kad pasirinkti kokybės rodikliai yra tinkami naudoti pagal paskirtį ir atspindi kokybės tikslų pasiekimo lygį. Susitikimo pabaigoje visi komandos nariai turi pritarti sudarytam kokybės planui.

Rezultatai ir išvados

Šio magistro darbo metu buvo surinkta visa prieinama medžiaga apie komandinį programų kūrimo procesą (TSP). Remiantis šia medžiaga buvo atlikta TSP modelio rekonstrukcija, kuri apėmė TSP scenarijus ir jų aprašymus. Antrame magistro darbo etape buvo aprašytas projekto komandos tipas ir sukurta rekonstruoto TSP modelio diegimo metodika apibrėžtam komandos tipui.

Pagrindiniai darbo rezultatai:

- rekonstruota komandinio programų kūrimo proceso (TSP) metodika;
- sukurta TSP diegimo metodika, skirta apibrėžtam komandos tipui.

Nagrinėjant komandinio programų kūrimo procesą (TSP) ir jo taikymo galimybes Lietuvoje, buvo padarytos tokios išvados:

- viešai prieinamos informacijos apie komandinį programų kūrimo procesą neužtenka, kad būtų galima atlikti pilną TSP metodikos rekonstravimą;
- trūkstama informacija neturi esminės įtakos TSP metodikai ir jos diegimui;
- adaptuota TSP metodika gali būti taikoma Lietuvos įmonėse.

Šaltinių sąrašas

- [DM03] Noopur Davis, Julia Mullaney. The Team Software ProcessSM (TSPSM) in Practice: A Summary of Recent Results. 2003. 105 pages.
<http://www.sei.cmu.edu/pub/documents/03.reports/pdf/03tr014.pdf>
- [DMcH03] Noopur Davis, Jim McHale. Relating the Team Software ProcessSM. (TSPSM) to the Capability Maturity Model for Software (SW-CMM). 2003 178 pages.
<http://www.sei.cmu.edu/pub/documents/02.reports/pdf/02tr008.pdf>
- [HO99] Watts S. Humphrey, Jim Over. The Team Software processSM (TSPSM) Tutorial. Software Engineering Institute Carnegie Mellon University, Pittsburgh. 1999. 203 pages. <https://seir.sei.cmu.edu/seir/>
- [Hum99] Watts S. Humphrey. Introduction to the Team Software Process(SM). Addison-Wesley Professional. 1999. 496 pages.
- [Hum00] Watts S. Humphrey. The Team Software ProcessSM (TSPSM). 2000. 51 pages.
www.sei.cmu.edu/pub/documents/00.reports/pdf/00tr023.pdf
- [Hum02] Watts S. Humphrey. How PSP and TSP Can Impact an Organization. 2002. pp. 40-72. <http://www.sei.cmu.edu/tsp/perspectives.pdf>
- [Hum05] Watts S. Humphrey. TSPSM-Leading a Development Team. [Addison Wesley Professional](http://www.addison-wesley.com). 2005. 336 pages.
- [Hum06A] Watts S. Humphrey. TSPSM: Coaching Development Teams. [Addison Wesley Professional](http://www.addison-wesley.com). 2006. 448 pages.
- [Hum06B] Watts S. Humphrey. PSPSM: A Self-Improvement Process for Software Engineers. [Addison Wesley Professional](http://www.addison-wesley.com). 2006. 346 pages.
- [Kim04] K. Kimberland. Microsoft's Pilot of TSP Yields Dramatic Results.
<http://www.sei.cmu.edu/publications/news-at-sei/features/2004/2/feature-1-2004-2.htm>. 76KB. 2004.
- [McA00] Donald R. McAndrews, The Team Software Process (TSPSM): An Overview and Preliminary Results of Using Disciplined Practices. 2000. 52 pages.
<http://www.sei.cmu.edu/pub/documents/00.reports/pdf/00tr015.pdf>
- [McH03] Jim McHale. The Great Myths About TSPSM and CMMI[®]. Software Engineering Institute Carnegie Mellon University, Pittsburgh. 2003. 15 pages.
<https://seir.sei.cmu.edu/seir/>

- [MW05] James McHale, Daniel S. Wall. Mapping TSP to CMMI. 2005. 209 pages.
<http://www.sei.cmu.edu/pub/documents/04.reports/pdf/04tr014.pdf>
- [NR99] R. Narawane, V. Ranganathan. Applying TSP to very small teams. 1999. 15 pages.
<https://seir.sei.cmu.edu/seir/>
- [WMP05] Daniel S. Wall, James McHale, Marsha Pomeroy-Huff. Case Study: Accelerating Process Improvement by Integrating the TSP and CMMI. September 2005. 41 pages.
<http://www.sei.cmu.edu/pub/documents/05.reports/pdf/05sr012.pdf>

Terminų žodynas

Gebėjimo brandos modelis	angl. Capability Maturity Model (CMMI)
Projekto dokumentacija	angl. Project notebook
Programos derinimas	angl. debugging
TSP instruktorius	angl. TSP coach
Uždirbtos vertės metodas	angl. Earned Value Method

1 priedas. Pagrindinių TSP elementų sąrašas

Pagrindiniai TSP elementai [MW05]:

Grupė/pavadinimas	Aprašymas	Pastabos
Startavimo scenarijai		
LAU	Komandos vadovas, aprašantis startavimo procesą	
LAU1	Pirmas startavimo susitikimas – apžvalga ir startavimo pradžia	1 žingsnis LAU scenarijuje
LAU2	Antras startavimo susitikimas – rolės ir tikslai	2 žingsnis LAU scenarijuje
LAU3	Trečias startavimo susitikimas – strategija, procesas, palaikymas	3 žingsnis LAU scenarijuje
LAU4	Ketvirtas startavimo susitikimas – bendras komandos planas	4 žingsnis LAU scenarijuje
LAU5	Penktas startavimo susitikimas – kokybės planas	5 žingsnis LAU scenarijuje
LAU6	Šeštas startavimo susitikimas – detalūs einamojo ciklo planai	6 žingsnis LAU scenarijuje
LAU7	Septintas startavimo susitikimas – rizikos įvertinimas	7 žingsnis LAU scenarijuje
LAU8	Aštuntas startavimo susitikimas – pasiruošimas susitikimui su vadovybe	8 žingsnis LAU scenarijuje
LAU9	Devintas startavimo susitikimas – susitikimas su vadovybe	9 žingsnis LAU scenarijuje
LAUPM	Startavimo aptarimas -	PM žingsnis LAU scenarijuje
REL	Pakartotinis startavimas	
REL1	Pirmas pakartotinio startavimo susitikimas – projekto būseną ir tikslai	
Kūrimo scenarijai		
DEV	Kūrimo procesas	
MAINT	Palaikymo procesas	
ANA	Analizės procesas	
HLD	Aukšto lygio projektavimo procesas	
IMP	Realizavimo procesas	
IMP6	Modulių testavimas ir testų kūrimo procesas	6 žingsnis IMP scenarijuje
INS	Inspektavimo procesas	
PM	Projekto aptarimas	
REQ	Reikalavimų procesas	
TEST	Testavimo procesas	
TEST1	Sistemos surinkimo procesas	1 žingsnis TEST scenarijuje
TEST2	Integravimo testavimas	2 žingsnis TEST scenarijuje
TEST3	Sistemos testavimo procesas	3 žingsnis TEST scenarijuje
TESTD	Testavimo metu rastų defektų valdymo procesas	
Kiti scenarijai		
MTG	Bendras susitikimo procesas	Naudojamas kaip daugumos susitikimų scenarijų pagrindas
STATUS	Susitikimas su vadovybe ir/ar užsakovu, projekto būsenos ataskaita	
WEEK	Kas savaitinis komandos susitikimas	

Grupė/pavadinimas	Aprašymas	Pastabos
Startavimo formos		
GOAL	Komandos tikslai	
INV	Proceso inventorių	
ITL	Problemu/riziku sekimo forma	
MTG	Susitikimo ataskaitos forma	
PIP	Proceso gerinimo pasiūlymų forma	
ROLE	Komandos rolių priskyrimas	
ROLEMX	Rolių priskyrimo matrica	
SCHED	Tvarkaraščio planavimo šablonas	
STRAT	Strateginio planavimo forma	
SUMDI	Padarytų defektų suvestinė	
SUMDR	Pašalintų defektų suvestinė	
SUMP	Planavimo suvestinės forma	
SUMQ	Kokybės suvestinės forma	
SUMS	Programos dydžio suvestinė	
SUMT	Kūrimo laiko suvestinės forma	
SUMTASK	Užduočių planavimo suvestinė	
TASK	Užduočių planavimo šablonas	
Kūrimo formos		
DEFECT	Pranešimų apie defektus forma	
INS	Inspektavimo ataskaita	
TESTLOG	Testavimo ataskaita	
LOGD	Defektų fiksavimo forma	
LOGT	Laiko fiksavimo forma	
WEEK	Kasavartinė projekto būsenos ataskaita	Modifikuota startavimo formos WEEK versija
Kontroliniai pasiruošimo sąrašai		
PREPL	Pasiruošimas startavimui	
PREPR	Pasiruošimas pakartotiniam startavimui	
Startavimo nurodymai		
Startavimo instruktorius	Nurodymai startavimo instruktoriui	
Marketingas	Nurodymai prezentacijos, kurioje dalyvauja marketingo skyriaus (arba užsakovo) atstovai, paruošimui.	
Kiti dalyviai	Nurodymai startavimo instruktoriui	
Vadovybė	Nurodymai prezentacijos, kurioje dalyvauja vadovybės atstovai, paruošimui.	
Komandos lyderis	Nurodymai komandos lyderiui	
Komandos nariai	Nurodymai komandos nariams	
Kiti dokumentai		
NOTEBOOK	Projekto dokumentacija	
STATUS	Ataskaita vadovybei apie projekto būseną	
SUMMARY	Projekto analizės ataskaita	

2 priedas. TSP kokybės planas

TSP kokybės plano šablonas [Hum00]:

Name	_____	Date	_____
Project	_____	Launch/Phase	_____
Part/Assembly	_____	Assembly Level	_____
Percent Defect Free		Plan	Actual
In compile		_____	_____
In unit test		_____	_____
In integration test		_____	_____
In system test		_____	_____
In acceptance test		_____	_____
In one year of use		_____	_____
In product life		_____	_____
Defect/page			
Requirements inspection		_____	_____
HLD review		_____	_____
HLD inspection		_____	_____
Defects/KLOC			
DLD review		_____	_____
DLD inspection		_____	_____
Code review		_____	_____
Compile		_____	_____
Code inspection		_____	_____
Unit test		_____	_____
Build and integration		_____	_____
System test		_____	_____
Total development		_____	_____
Acceptance test		_____	_____
Product life		_____	_____
Total		_____	_____
Defect Ratios			
Code review/Compile		_____	_____
DLD review/Unit test		_____	_____
Development time ratios (%)			
Requirements inspection/Req. time		_____	_____
HLD inspection/HLD time		_____	_____
DLD/code		_____	_____
DLD review/design		_____	_____
Code review/code		_____	_____
A/FR		_____	_____
Personal review rates			
DLD lines/hour		_____	_____
Code LOC/hour		_____	_____

Inspection rates		
Requirement pages/hour		
HLD pages/hour		
DLD lines/hour		
Code LOC/hour		
Part/Assembly	Date	
Defect-injection Rates	Plan	Actual
Requirements		
HLD		
DLD		
Coding		
Compile		
Unit test		
Build and integration		
System test		
Defect-removal Rates	Plan	Actual
Requirements		
System test planning		
Requirements inspection		
HLD		
Integration test planning		
HLD inspection		
DLD review		
Test development		
DLD inspection		
Code		
Code review		
Compile		
Code inspection		
Unit test		
Build and integration		
System test		
Phase Yields	Plan	Actual
Requirements inspection		
HLD inspection		
DLD review		
DLD inspection		
Code review		
Compile		
Code inspection		
Unit test		
Build and integration		
System test		
Process Yields	Plan	Actual
% before compile		
% before unit test		
% before build and integration		
% before system test		
% before system delivery		

3 priedas. TSP formos

TSP komandos ir komandos narių įvertinimo forma PEER [Hum99]:

For each role, evaluate the work required and the relative difficulty in % during this cycle.		
Role	Work Required	Role Difficulty
Team Leader		
Development Manager		
Planning Manager		
Quality/Process Manager		
Support Manager		
Total Contribution (100%)		

Rate the overall team against each criteria - circle one number from 1 (least) to 5 (most).					
Team spirit	1	2	3	4	5
Overall effectiveness	1	2	3	4	5
Rewarding experience	1	2	3	4	5
Team productivity	1	2	3	4	5
Process quality	1	2	3	4	5
Product quality	1	2	3	4	5

Rate role for overall contribution - circle one number from 1 (least) to 5 (most).					
Team Leader	1	2	3	4	5
Development Manager	1	2	3	4	5
Planning Manager	1	2	3	4	5
Quality/Process Manager	1	2	3	4	5
Support Manager	1	2	3	4	5

Rate each role for helpfulness and support - circle one number from 1 (least) to 5 (most).					
Team Leader	1	2	3	4	5
Development Manager	1	2	3	4	5
Planning Manager	1	2	3	4	5
Quality/Process Manager	1	2	3	4	5
Support Manager	1	2	3	4	5

Rate each role for how well it was performed - circle one number from 1 (least) to 5 (most).					
Team Leader	1	2	3	4	5
Development Manager	1	2	3	4	5
Planning Manager	1	2	3	4	5
Quality/Process Manager	1	2	3	4	5
Support Manager	1	2	3	4	5

TSP konfigurācijas keitimo forma CCR [Hum99]:

Name _____ Date _____
Part/Level _____ Cycle _____

Product Information

Product Name _____ Product Owner _____
Product/Change Size _____ Size Measure _____
Recent Inspection _____ Moderator _____
Backup Address: _____

Change Information

Reason for Change: _____

Change Benefits: _____

Change Impact: _____

Change Description: (For source code, attach listing, for code changes, include defect number (if any) and listing of changed and unchanged program segment)

Status

Approved: Additional Information: Disapproved:
Information needed: _____

Approvals

Product Owner _____ Date: _____
Quality/Process Manager _____ Date: _____
CCB _____ Date: _____

TSP defektų fiksavimo forma LOGD [Hum99]:

Name _____					Date _____	
Part/Level _____					Cycle _____	
Date	Number	Type	Inject	Remove	Fix Time	Fix Defect
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Description: _____						

Date	Number	Type	Inject	Remove	Fix Time	Fix Defect
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Description: _____						

Date	Number	Type	Inject	Remove	Fix Time	Fix Defect
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Description: _____						

Date	Number	Type	Inject	Remove	Fix Time	Fix Defect
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Description: _____						

Date	Number	Type	Inject	Remove	Fix Time	Fix Defect
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Description: _____						

TSP inspektavimo ataskaita INS [Hum99]:

Name _____ Date _____
 Part/Level _____ Cycle _____
 Moderator _____ Owner _____

Engineer Data

Name	Defects		Preparation Data			Est.
	Major	Minor	Size	Time	Rate	Yield
Totals:						

Defect Data

No	Defect Description	Defects		Engineers (finding major defects)						
		Maj	Min					A	B	
Totals										
Unique Defects										

Inspection Summary

Total defects for A: _____ **Product Size:** _____ **Size Measure:** _____
Total Defects (AB/C): _____ **Total defects for B:** _____ **C (# common):** _____
Meeting Time: _____ **Number Found (A+B-C):** _____ **Number Left:** _____
Total Inspection Hours: _____ **Overall Rate:** _____

TSP problemų sekimo forma ITL [Hum99]:

Name _____ Date _____
 Part/Level _____ Cycle _____

Date	Risk/Issue	Number	Priority	Owner	FU Date	Resolved
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Description: _____						

Date	Risk/Issue	Number	Priority	Owner	FU Date	Resolved
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Description: _____						

Date	Risk/Issue	Number	Priority	Owner	FU Date	Resolved
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Description: _____						

Date	Risk/Issue	Number	Priority	Owner	FU Date	Resolved
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Description: _____						

Date	Risk/Issue	Number	Priority	Owner	FU Date	Resolved
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Description: _____						

TSP proceso gerinimo pasiūlymų forma PIP [Hum99]:

PIP Number _____

Priority _____

Problem Description:

Briefly describe the problem encountered and its impact.

Proposal Description

Describe suggested changes as completely as possible, including affected forms, scripts, etc.

When completed, submit PIP to the Quality/Process Manager and keep a copy.

Do not write below this line

PIP Control #	_____	Organization	_____
Received	_____	Acknowledged	_____
Updated	_____	Closed	_____
Changes	_____		_____
	_____		_____
	_____		_____

TSP planavimo suvestinė SUMP [Hum99]:

Name	Date		
Part/Level	Cycle		
Product Size	Plan	Actual	
Requirements pages (SRS)			
Other text pages			
High-level design pages (SDS)			
Detailed design lines			
Base LOC (B) (measured)			
Deleted LOC (D)			
	(Estimated)		(Counted)
Modified LOC (M)			
	(Estimated)		(Counted)
Added LOC (A)			
	(N-M)		(T-B+D-R)
Reused LOC (R)			
	(Estimated)		(Counted)
Total New & Changed LOC (N)			
	(Estimated)		(A+M)
Total LOC (T)			
	(N+B-M-D+R)		(Measured)
Total New Reuse LOC			
Estimated Object LOC (E)			
Upper Prediction Interval (70%)			
Lower Prediction Interval (70%)			
Time in Phase (hours)	Plan	Actual	Actual %
Management and miscellaneous			
Launch and strategy			
Planning			
Requirements			
System test plan			
Requirements inspection			
High-level design			
Integration test plan			
High-level design inspection			
Implementation planning			
Detailed design			
Detailed design review			
Test development			
Detailed design inspection			
Code			
Code review			
Compile			
Code inspection			
Unit test			
Build and integration			
System test			
Documentation			
Postmortem			
Total			
Total Time UPI (70%)			
Total Time LPI (70%)			

TSP rolių pasirinkimo forma ROLE [Hum99]:

Name _____ Date _____

Briefly describe your relevant experience and interests:

Briefly describe your work on other team projects:

Briefly describe any leadership or management positions you have held (at work or in clubs/organizations):

State your team preferences, if any:

List your class schedule and other times when you have scheduled activities such as work, ROTC, clubs, sports teams, etc.							
Time	Mon.	Tue.	Wed.	Thu.	Fri.	Sat.	Sun.
800-900							
915-1015							
1030-1130							
1145-1245							
1300-1400							
1415-1515							
1530-1630							
1645-1745							

Rank from 1 (least) to 5 (most) your preferences for serving in the following team roles:					
Team Leader	1	2	3	4	5
Development Manager	1	2	3	4	5
Planning Manager	1	2	3	4	5
Quality/Process Manager	1	2	3	4	5
Support Manager	1	2	3	4	5