



**KAUNO TECHNOLOGIJOS UNIVERSITETAS**  
**INFORMATIKOS FAKULTETAS**

**Lina Piatikonė**

**PROGRAMINĖS ĮRANGOS DOKUMENTACIJOS RENGIMO**  
**LANKŠČIU FORMATU TYRIMAS**

Baigiamasis magistro projektas

**Vadovas**

Dr. Ramūnas Kubiliūnas

**KAUNAS, 2016**

**KAUNO TECHNOLOGIJOS UNIVERSITETAS**  
**INFORMATIKOS FAKULTETAS**

**PROGRAMINĖS ĮRANGOS DOKUMENTACIJOS RENGIMO  
LANKŠČIU FORMATU TYRIMAS**

Baigiamasis magistro projektas

Nuotolinio mokymosi informacinės technologijos (kodas 621E14002)

**Vadovas**

Dr. Ramūnas Kubiliūnas

**Recenzentas**

Doc. dr. Danguolė Rutkauskienė

**Projektą atliko**

Lina Piatikonė

**KAUNAS, 2016**



**KAUNO TECHNOLOGIJOS UNIVERSITETAS**

Informatikos fakultetas

(Fakultetas)

Lina Piatikonė

(Studento vardas, pavardė)

Nuotolinio mokymosi informacinės technologijos, 621E14002

(Studijų programos pavadinimas, kodas)

„Programinės įrangos dokumentacijos rengimo lanksčiu formatu tyrimas“

**AKADEMINIO SAŽINGUMO DEKLARACIJA**

20 \_\_\_\_ m. \_\_\_\_\_ d.  
Kaunas

Patvirtinu, kad mano, **Linos Piatikonės**, baigiamasis projektas tema „Programinės įrangos dokumentacijos rengimo lanksčiu formatu tyrimas“ yra parašytas visiškai savarankiškai ir visi pateikti duomenys ar tyrimų rezultatai yra teisingi ir gauti sąžiningai. Šiame darbe nei viena dalis nėra plagijuota nuo jokių spausdintinių ar internetinių šaltinių, visos kitų šaltinių tiesioginės ir netiesioginės citatos nurodytos literatūros nuorodose. Įstatymų nenumatytų piniginių sumų už šį darbą niekam nesu mokėjęs.

Aš suprantu, kad išaiškėjus nesąžiningumo faktui, man bus taikomos nuobaudos, remiantis Kauno technologijos universitete galiojančia tvarka.

\_\_\_\_\_  
(vardą ir pavardę įrašyti ranka)

\_\_\_\_\_  
(parašas)

Piatikonė, Lina. PROGRAMINĖS ĮRANGOS DOKUMENTACIJOS RENGIMO LANKSČIU FORMATU TYRIMAS. Magistro baigiamasis projektas / vadovas dr. Ramūnas Kubiliūnas; Kauno technologijos universitetas, Informatikos fakultetas.

Mokslo kryptis ir sritis: Technologijos mokslai, informatikos inžinerija – 07T

Reikšminiai žodžiai: programinės įrangos dokumentacija, dita, dita-ot

Kaunas, 2016. 68 p.

## **SANTRAUKA**

Šiame darbe yra tiriamas dabartinis įmonės „IT Sprendimai“ dokumentacijos rengimo bei teikimo procesas, įvardinami egzistuojantys trūkumai ir tobulinimo galimybės. Kuriant programinės įrangos dokumentaciją pagal įprastinį jos rengimo procesą, kyla problemų, nes turinys yra neatskiriamas nuo atvaizdavimo, nėra galimybės konvertuoti turinio į skirtingus formatus, sudėtingai valdomi turinio pakeitimai, informacija dubliuojasi. Darbe sukurtas programinės įrangos dokumentacijos metodas remiasi lankstaus formato technologija DITA. Dokumentų konvertavimas į skirtingus formatus, pritaikius sukurtą metodą, vyksta vienos komandos paleidimu. Tam tikri dokumentacijos turinio duomenys yra automatiškai įdedami į lentelę ir taip supaprastina dokumentacijos rengimo procesą.

Atlikus eksperimentinį tyrimą, nustatyta, kad sukurtas dokumentacijos rengimo bei teikimo metodas veikia greičiau ir patogiau. Įmonei „IT Sprendimai“ pateikiama rekomendacija pereiti prie sukurto dokumentacijos metodo. Sudarytas diegimo planas ir naujos technologijos integravimo į egzistuojančią sistemą sprendimas.

Piatikonė, Lina. RESEARCH ON SOFTWARE DOCUMENTATION DEVELOPMENT USING FLEXIBLE FORMAT: Master's thesis / supervisor dr. Ramūnas Kubiliūnas. The Faculty of Informatics, Kaunas University of Technology.

Research area and field: Technological sciences, informatics engineering – 07T

Key words: software documentation, dita, dita-ot

Kaunas, 2016. 68 p.

## **SUMMARY**

The research object of this Master thesis is the process of software documentation preparation and delivery in company called "IT Sprendimai". During research some gaps and opportunities for improvements are identified: not possible to separate documentation's content from display, no opportunity to easily convert document to different formats, difficulties in managing documentation's content and duplication of information in document. The developed software documentation method is based on a flexible format technology DITA, which allows to convert documents into different formats using one command. Also, the new approach allows to place certain documentation content data automatically into the table and simplifies the documentation preparation process.

The experimental study showed that the developed documentation preparation and delivery method is faster and more convenient. The company "IT Sprendimai" has been given a recommendation to move to a newly developed method of documentation. Moreover, deployment and integration of new technologies into existing system solution plan has been prepared.

## TURINYS

ĮVADAS .....	10
1. PROGRAMINĖS ĮRANGOS DOKUMENTACIJOS RENGIMO BEI TEIKIMO PROCESO ANALIZĖ .....	13
1.1. Programinės įrangos dokumentacijos tipai .....	13
1.2. Reikalavimai programinės įrangos dokumentacijos kokybei .....	15
1.3. Programinės įrangos dokumentacijos specifiška.....	16
1.4. Programinės įrangos dokumentacijos rengimo procesas .....	17
1.5. Įmonės „IT Sprendimai“ programinės įrangos kūrimo architektūra .....	19
1.6. Dokumentacijos rengimo proceso problemos.....	20
1.7. XML technologijos .....	21
1.7.1. DITA architektūra .....	22
1.7.2. „DocBook“ .....	24
1.7.3. DITA ir „DocBook“ technologijų palyginimas .....	25
1.8. DITA architektūros analizė.....	26
1.8.1. OASIS organizacija.....	26
1.8.2. DITA OASIS standartų apžvalga.....	26
1.8.3. DITA architektūros pagrindiniai principai.....	27
1.8.4. DITA žemėlapiai .....	29
1.8.5. DITA publikavimo mechanizmai.....	30
1.9. Išvados .....	32
2. PROGRAMINĖS ĮRANGOS DOKUMENTACIJOS RENGIMO METODAS .....	33
2.1. Dokumentacijos proceso dalyviai .....	33
2.2. DITA integracija į sistemą .....	34
2.3. DITA sistemos diegimo žingsniai.....	35
2.4. Metodas programinės įrangos dokumentacijai rengti .....	37
2.5. Išplėstinis duomenų bazės lentelės dokumentacijos algoritmas .....	39
2.6. Išvados .....	42

3. DOKUMENTACIJOS PROCESO EKSPERIMENTINIS TYRIMAS .....	43
3.1. Konvertavimas į skirtingus formatus .....	43
3.2. Pakeitimų valdymas .....	43
3.3. Skirtingų dokumentų versijų kūrimas panaudojant tą patį turinį.....	46
3.4. Failų kiekio ir dydžio palyginimas .....	47
3.5. Dokumento stiliaus palyginimas.....	48
3.5.1. PDF formato dokumentų palyginimas .....	48
3.5.2. HTML formato dokumentų palyginimas .....	50
3.5.3. CHM formato dokumentų palyginimas .....	51
3.6. Išvados .....	53
IŠVADOS .....	54
LITERATŪRA .....	55
PRIEDAI.....	57
1 priedas: Tyrime naudotas dokumentacijos pavyzdys .....	57
2 priedas: Diegimo aktas.....	68

## PAVEIKSLŲ SĄRAŠAS

1 pav.	Dokumentacijos tipai .....	14
2 pav.	Dokumentacijos rengimo procesas .....	18
3 pav.	Vikio dokumentų kūrimas.....	19
4 pav.	Sistemos architektūra .....	20
5 pav.	Dokumentavimo technologijų panaudojimas.....	23
6 pav.	DITA koncepcija.....	28
7 pav.	DITA žemėlapiai .....	29
8 pav.	DITA failų sąryšis .....	30
9 pav.	Publikavimo mechanizmų panaudojimo tyrimas, 2009m.....	31
10 pav.	Dokumentacijos rengimo sistemos panaudojimo atvejų modelis .....	34
11 pav.	Sistemos architektūra ir pritaikyta DITA technologija .....	35
12 pav.	Programinės įrangos dokumentacijos rengimo metodą realizuojantis algoritmas .	38
13 pav.	Išplėstinis metodo algoritmas .....	40
14 pav.	Procedūros sugeneruoto XML dokumento pavyzdys.....	41
15 pav.	Word teksto rengykle konvertuotas PDF dokumentas .....	49
16 pav.	PDF dokumentas sukurtas naudojant DITA technologiją.....	50
17 pav.	Word teksto rengykle konvertuotas HTML dokumentas .....	51
18 pav.	HTML dokumentas sugeneruotas panaudojant DITA technologiją.....	51
19 pav.	chmProcessor programa konvertuotas CHM dokumentas .....	52
20 pav.	CHM dokumentas konvertuotas panaudojant DITA technologiją.....	52



## LENTELIŲ SĄRAŠAS

1 lentelė.	Vikio puslapių ir XML formato dokumentų panaudojimo palyginimas .....	21
2 lentelė.	DITA tinkamumas įmonei „IT sprendimai“ .....	23
3 lentelė.	DITA dokumentavimo struktūra.....	28
4 lentelė.	DITA diegimo žingsniai .....	36
5 lentelė.	Duomenų bazės lentelės dokumentacijos pavyzdys .....	39
6 lentelė.	Konvertavimo į skirtingus formatus palyginimas.....	43
7 lentelė.	Skirtingų stilių pritaikymas skirtingiems formatams.....	44
8 lentelė.	Teksto pakeitimai skirtinguose formatuose .....	44
9 lentelė.	Skyrių sukeitimas vietomis .....	45
10 lentelė.	Skirtingų versijų dokumentų kūrimas.....	46
11 lentelė.	Failų kiekis ir dydis diske .....	47

# IVADAS

## Darbo aktualumas

Programinės įrangos kūrimo procesas susideda iš daugelio veiklų, kurios gali trukti ilgą laiko tarpą. Viena iš veiklų, kuri trunka beveik nuo pat pirmųjų programinės įrangos kodo eilučių parašymo iki sistemos atidavimo galutiniam naudotojui ir net vėliau tobulinant sistemą, yra sistemos dokumentacijos rengimas bei teikimas. Programinės įrangos dokumentacija yra labai svarbi visiems naudotojams, nes dėl kompleksiškos verslo srities aspektų ar dėl kitų programinės įrangos naudojimo niuansų, nėra visuomet aišku, kaip tiksliai ir efektyviai naudotis sistema. Sistemos naudotojų dokumentacija taip pat užtikrina, kad visi jos naudotojai, kūrėjai, platintojai ir kiti asmenys turi vienodą supratimą apie sistemą, jos diegimą, pritaikymo galimybes ir funkcijas. Todėl tokia sistemos dokumentacija turi būti išsami, nuolatos atnaujinama ir platinama.

Sistemos dokumentacija gali būti rengiama panaudojant įvairias programines priemones, tačiau daugiausiai yra naudojamos lokalių teksto rengyklės ar žiniatinklio priemonės, skirtos dirbti su tekstu ir jį redaguoti. Tokią dokumentaciją dažniausiai rengia sistemos analitikas, kuris betarpiškai bendrauja tiek su programinės įrangos užsakovu ir žino visus sistemos reikalavimus, tiek su tiesiogiai prie programinės įrangos kūrimo prisidedančiais programuotojais ir testuotojais. Jeigu kuriama sistema yra didelė, tuomet ir ją kurianti komanda gali susidėti iš kelių analitikų, sistemos architektų, daugybės programuotojų bei testuotojų. Taigi sistemos dokumentacija gali būti nuolatos atnaujinama to paties ar skirtingų asmenų – papildoma, taisoma ir kitaip redaguojama. Jei egzistuoja tik vienas dokumentacijos dokumentas, tuomet jį redaguoti visiems asmenims vienu metu yra sudėtinga. Kyla ir kita problema, kai skirtingi dokumentacijos rengėjai pritaiko skirtingą dokumentų stilių, tuomet reikia įdėti nemažai pastangų norint suvienodinti viso dokumento stilių. Todėl reikia užtikrinti, kad būtų naudojamas efektyvus programinės įrangos dokumentacijos rengimo procesas ir viena technologija, kuri leistų lengvai valdyti turinio ir stiliaus pakeitimus.

Be to, programinės įrangos dokumentacija privalo būti pateikiama skirtingais formatais. Kiekvienas formatas turi tam tikras ypatybes, jei pvz. virtualioje mokymosi aplinkoje pateikta mokymosi medžiaga yra patogi skaitymui ar mokymuisi, tai tikėtina, kad toks elektroninės mokymosi medžiagos formatas nebus patogus spausdinimui. Parengta sistemos dokumentacija vienu formatu gali būti konvertuojama į kitus formatus, tačiau tam reikia laiko, o kartais ir papildomos programinės įrangos. Dokumentacijos, parengtos įprastu formatu (kai turinys nėra atskirtas nuo atvaizdavimo), konvertavimas nėra efektyvus būdas organizuoti dokumentacijos

rengimo ir teikimo procesą. Sistemos naudotojų dokumentacija, kintant ar atnaujinant sistemos funkcionalumą, turi būti nuolat atnaujinama, o esant daugeliui skirtingų failų su skirtingais formatais tai gali užimti daug laiko ir pastangų. Dėl šios priežasties, reikia rasti efektyvų programinės įrangos dokumentacijos rengimo būdą tokiu formatu, kurį būtų patogu konvertuoti. Naujasis formatas galėtų būti atvaizduojamas įvairiais formatais be papildomų naudotojo pastangų bei sumažinant laiko sąnaudas.

### **Darbo tikslas**

Darbo tikslas yra sukurti ir iširti programinės įrangos dokumentacijos rengimo ir pateikimo naudotojams metodą bei priemones, užtikrinančias efektyvų dokumentacijos rengimą ir vienodą dokumentų stilių bei leidžiančias konvertuoti dokumentaciją į skirtingus formatus automatiškai.

### **Darbo uždaviniai**

1. Išanalizuoti programinės įrangos dokumentacijos rengimo procesą, siekiant įvertinti kylančias problemas.
2. Išbandyti ir įvertinti technologijų, skirtų rengti turinį lanksčiu formatu, panaudojimo galimybes programinės įrangos dokumentacijai rengti.
3. Sukurti metodą ir priemones programinės įrangos dokumentacijai rengti lanksčiu formatu.
4. Iširti ir įvertinti programinės įrangos dokumentacijos rengimo lanksčiu formatu metodą ir priemones.

### **Darbo objektas**

Technologijos, skirtos sistemos naudotojų dokumentaciją rengti lanksčiu formatu.

### **Darbo produktas**

Metodas ir priemonės programinės įrangos dokumentacijai rengti lanksčiu formatu.

### **Darbo metodai**

Analizės būdu nustatomi programinės įrangos dokumentacijai rengti ir teikti reikalingi formatai. Lankstaus formato aprašymas ir dokumentacijos rengimo modeliavimas skirtingiems formatams atvaizduoti.

## **Darbo struktūra**

Darbą sudaro: įvadas, trys pagrindinės dalys, išvados, naudotos literatūros sąrašas ir priedai. Pirmoje dalyje yra išanalizuotas programinės įrangos dokumentacijos rengimo procesas, identifikuotos tobulinimo galimybės, pasirinkta ir aprašyta DITA technologija. Antroje dalyje yra pristatomas sukurtas metodas, kuris atskiria turinį nuo atvaizdavimo, bei pateikiamas jį realizuojantis algoritmas. Naujo metodo algoritmas detalizuoja failų konvertavimą į skirtingus formatus vienos komandos paleidimu. Šioje dalyje taip pat pateikiama sukurta procedūra, palengvinanti dokumentacijos turinio kūrimą. Trečioje dalyje yra palyginamas dviejų metodų (senojo ir naujojo) veikimas ir identifikuojamas efektyvesnis programinės įrangos dokumentacijos rengimo būdas.

# 1. PROGRAMINĖS ĮRANGOS DOKUMENTACIJOS RENGIMO BEI TEIKIMO PROCESO ANALIZĖ

Šiame skyriuje apžvelgiami galimi dokumentacijos tipai ir standartai bei analizuojamas konkretus pavyzdys – informacinių technologijų įmonės „IT sprendimai“ (įmonės pavadinimas pakeistas) dokumentacijos rengimo ir teikimo procesas.

## 1.1. Programinės įrangos dokumentacijos tipai

Programinės įrangos dokumentacija aprašo būsimą, einamuoju metu kuriamą arba jau funkcionuojantį produktą. Dažnai programinės įrangos rengimo procesas trunka ilgą laiką, nes produkto reikalavimai kinta ir atsiranda poreikis atnaujinti dokumentaciją.

Sistemos naudotojai gali būti skirtingi: sistemos administratoriai, diegėjai, pradedantieji naudotojai, patyrę naudotojai, sistemos vertintojai ir pan. Kiekvienas programinės įrangos produktas gali turėti skirtingus naudotojus. Sistemos naudotojų dokumentacija aprašo tam tikrą sistemą, kurios funkcijos gali būti pritaikomos tik tam tikram naudotojui, keliems naudotojams arba visiems. Pavyzdžiui, prisijungimo prie sistemos funkcija gali būti atliekama vienodai, nesvarbu koks yra sistemos naudotojas, jis vis vien turės tam tikrame sistemos laukelyje įvesti naudotojo vardą ir slaptažodį. Toks sistemos funkcionalumo žingsnis gali būti įtrauktas tiek į pradedančiųjų dokumentacijos dokumentą, tiek į administratoriaus ar patyrusio naudotojo dokumentaciją. Todėl kyla problema, kai toks universalus visiems naudotojams žingsnis tekste yra dubliuojamas daug kartų, nes jei sistema pasikeičia, tuomet ir tą patį žingsnio aprašymą tenka keisti visose dokumentų vietose.

Yra svarbu užtikrinti, kad kiekvienas sistemos naudotojas turėtų pagal jo patirtį ir reikalavimus pritaikytą dokumentaciją, kurią galėtų naudoti. Pavyzdžiui, sistemos administratorių ir galutinių naudotojų dokumentacija turėtų ypatingai skirtis, nes šiems naudotojams yra aktualios skirtingos užduotys ir problemų sprendimo būdai. Dažnai sistemos administratoriams yra aktualios sistemos techninės galimybės bei sistemos valdymas ar pritaikymas konkrečiam tikslui, o galutiniam naudotojui yra svarbu kaip atlikti tam tikras užduotis bei išbandyti sistemos funkcionalumą.

Siekiant pritaikyti dokumentaciją skirtingiems naudotojų tipams, reikia skirtingų dokumentų. Galimi įvairūs dokumentai, pritaikyti skirtingiems naudotojams (žr. 1 pav.):

- funkcinis sistemos aprašymas pateikia sistemos reikalavimus ir pagrindinį funkcionalumą. Tai dokumentas, kuris aprašo sistemą, detalai nesigilinant į esamų

funkcijų atlikimą, todėl naudotojas, perskaitęs šį dokumentą, gali susidaryti nuomonę ar tai yra jam tinkamas produktas, kurį jis ir toliau norės naudoti;

- sistemos diegimo instrukcija aprašo sistemos diegimo žingsnius, reikalingą papildomą programinę įrangą, sistemos reikalavimus. Ši instrukcija taip pat apibrėžia kaip paleisti sistemą bei ištaisyti dažniausiai diegimo metu pasitaikančias klaidas;
- įvadinė instrukcija yra skirta susipažinti su sistema. Tai dokumentas, kuris yra pritaikytas pradedančiajam naudotojui, todėl turėtų būti detalus ir apibrėžti pagrindines sistemos funkcijas bei paaiškinti kaip paleisti sistemą;
- detali sistemos instrukcija aprašo pilną sistemos funkcionalumą, klaidų pranešimus, klaidų sprendimo būdus. Šiame dokumente nėra būtina aprašyti primityvius naudotojo žingsnius (pvz. kaip prisijungti prie sistemos), nes ši dokumentacija yra skirta patyrusiam naudotojui;
- techninė sistemos specifikacija aprašo sistemos realizaciją techniniu lygmeniu. Tokia dokumentacija yra skirta sistemos administratoriams ir turi aprašyti klaidų pranešimus, sistemos klaidų sprendimo būdus, kaip prijungti sistemą prie kitų įrenginių (jei tokių yra) ir pan. [18.]



1 pav. Dokumentacijos tipai

Skirtingi dokumentacijos dokumentai leidžia naudotojui nepasimesti tarp skirtingų dokumentacijos tipų (žr. 1 pav.). Sistemos naudotojas turėtų rinktis sistemos dokumentacijos dokumentą pagal savo žinias, jei jis yra pradedantysis naudotojas, tuomet jis turėtų pasirinkti įvadinės instrukcijos dokumentą, kuriame bus aprašomi sistemos naudojimosi pradmenys (pvz.

internetinės parduotuvės atveju, kaip įsidėti prekę į krepšelį). Tokia pati dokumentų pasirinkimo strategija galioja ir kitiems sistemos naudotojams.

## **1.2. Reikalavimai programinės įrangos dokumentacijos kokybei**

Programinės įrangos dokumentacijos kokybė yra tokia pati svarbi, kaip ir produkto, kurį ji aprašo. Kurti ir palaikyti kokybišką dokumentaciją nėra lengva bei gali būti brangu. Kiekviena dokumentacija turi atitikti tam tikrą struktūrą, kad galutinis naudotojas galėtų rasti reikiamą informaciją. Naudotojo dokumentacijos standarte, patvirtiname „The Institute of Electrical and Electronics Engineers“ (IEEE) yra nurodomi tokie rekomenduojami dokumento komponentai: [13]

- identifikatorius – dokumento pavadinimas ir unikalus identifikatorius pagal kurį galima būtų identifikuoti dokumentą;
- turinys – skyrių ir poskyrių puslapių numeriai;
- paveikslų sąrašas – paveikslų pavadinimai ir puslapių numeriai;
- įžanga – nurodamas dokumento tikslas ir santrauka;
- dokumento panaudojimo instrukcija – aprašoma kaip skirtingiems naudotojams skaityti bei rasti informaciją dokumente;
- sistemos operacijos – konceptuali informacija apie galimas sistemos operacijas;
- procedūros – sistemos panaudojimo instrukcija;
- sistemos komandų informacija – sistemos komandų panaudojimo instrukcija;
- klaidų pranešimai – klaidų pranešimų aprašymas ir galimi klaidų sprendimo būdai;
- žodynas – sąvokų apibrėžimai;
- susijusi informacija – nuorodos į kitus susijusius informacijos šaltinius;
- dokumento navigacija – nuorodos į kitas dokumento vietas, padedančios lengviau naudotis dokumentu;
- rodyklė – sąvokos ir puslapiai, kuriuose šios sąvokos yra panaudotos;
- paieška – elektroninio dokumento atveju yra pridama paieška dokumente.

Tam tikri išvardinti komponentai (pvz. turinys) turėtų būti atskirame puslapyje, dokumento pradžioje, tačiau kiti komponentai (pvz. dokumento navigacija) gali būti aprašyti visame dokumente, kiekviename skyriuje pridedant nuorodas į kitus susijusius skyrius. [18]

Programinės įrangos dokumentacijos kokybės reikalavimai yra labai svarbūs rengiant dokumentacijos dokumentus, nes pagrindinis dokumentacijos tikslas yra pagelbėti naudotojui naudotis sistema. Tinkamai aprašytas sistemos funkcionalumas atsako į galimai kylančius naudotojo klausimus, todėl naudotojas yra linkęs skaityti dokumentaciją. Jei naudotojas nebus suinteresuotas naudotis dokumentacijos dokumentais, tuomet nebus ir tikslo juos rengti. O jei atsisakysime naudotojų dokumentacijos, tai nauji programinės įrangos dokumentacijos rengimo metodai taps taip pat nereikalingi.

### **1.3. Programinės įrangos dokumentacijos specifi­ka**

Priklausomai nuo programinės įrangos srities, jos dokumentacijos specifi­ka, t.y. struktūra, naudojama dokumentacijos kūrimo technologija ir pritaikomumas skirtingoms naudotojų grupėms gali skirtis. Pavyzdžiui, įmonė „IT sprendimai“ kuria verslo informacinės sistemas. Vienas iš daugelio kuriamų produktų yra Norvegijos rinkai pritaikyta viešųjų pirkimų sistema, o jos naudojimui yra reikalinga tik interneto naršyklė ir interneto ryšys. Šioje viešųjų pirkimų sistemoje galima skelbti konkursus tam tikrai prekei įsigyti arba parduoti. Ši sistema yra kuriama iš pat pradžių, todėl sistemos naudotojų dokumentacija yra rašoma taip pat nuo pradžių, o tai leidžia lengvai pakeisti dokumentacijos rengimo metodą, nes nereikia atlikti dokumentacijos migravimo darbų. Kaip ir kiekvienas IT produktas, ši viešųjų pirkimų sistema privalo būti gerai dokumentuota, kad visiems naudotojams būtų aiškus jos panaudojimas bei esamas funkcionalumas.

Programinės įrangos dokumentacijai rengti yra naudojami vikio (angl. *wiki*) puslapiai, kuriuose yra nuolatos atnaujinama sistemos naudotojų dokumentacija. Sistemos naudotojų dokumentacija yra išskaidyta į skirtingus vikio puslapius, kuriuose yra kaupiama tikslinė dokumentacija:

- įvadinė sistemos instrukcija;
- detali instrukcija;
- techninė specifikacija.

Programinės įrangos naudotojų dokumentaciją rengia sistemos analitikas, o priėjimą prie šios medžiagos turi visi įmonėje dirbantys asmenys. Kol sistema nėra pabaigta, galutinis



naudotojas (užsakovas) sistemos dokumentacijos matyti negali, ji bus pristatyta tuomet, kai visa sistema bus pabaigta. Visgi sistemos naudotojas galėtų būti įtraukiamas į dokumentacijos rengimo procesą anksčiau, galbūt jis turėtų tam tikrų pastabų, o ištaisant visus netikslumus pagerėtų dokumentacijos kokybė. Tačiau naudojant dabartinį įmonės „IT Sprendimai“ dokumentacijos rengimo procesą, konvertuoti ir pateikti dokumentaciją naudotojui skirtingais formatais yra sudėtinga ir kaskart tai atlikti užimtų daug laiko, todėl to yra atsisakoma.

#### **1.4. Programinės įrangos dokumentacijos rengimo procesas**

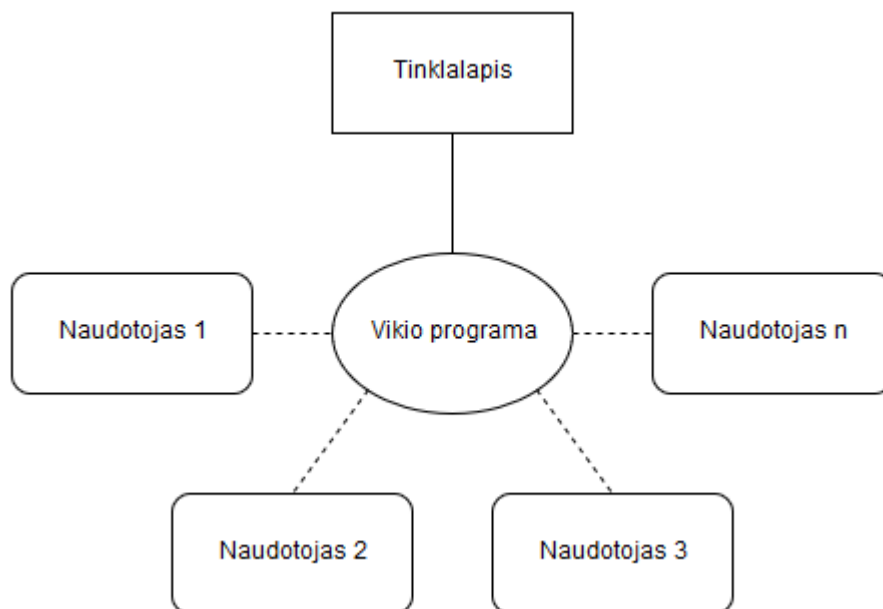
Programinės įrangos dokumentacija, programinės įrangos kūrimo metu, yra nuolatos keičiama: atnaujinamas tekstas, lentelės ir paveikslėliai. Po dokumentacijos pristatymo galutiniam naudotojui, sistemos palaikymo metu dokumentacija taip pat gali kisti: atnaujinami sistemos panaudojimo žingsniai, taisomos klaidos, pridedami nauji žingsniai. Todėl sistemos dokumentacija nėra rengiama vieną kartą, o yra kaskart atnaujinama ir papildoma.

Programinės įrangos dokumentacija yra pateikiama galutiniam naudotojui PDF formatu, kad būtų paprasta tokią medžiagą atspausdinti ir skaityti įvairiuose įrenginiuose. Siekiant, kad užsakovas gautų dokumentaciją PDF formatu, programinės įrangos kūrėjai gali tai padaryti dviem būdais:

- konvertuoti kiekvieną vikio puslapį į PDF formatą, o vėliau panaudojant specialią programinę įrangą, sujungti visus PDF failus į vieną dokumentą;
- nukopijuoti kiekvieną puslapį į „Microsoft Word“ teksto rengyklę, tuomet konvertuoti visą dokumentą į PDF formatą.

Nesvarbu kurį variantą programinės įrangos kūrėjai naudoja, kiekvieną kartą dokumentų kopijavimas užima labai daug laiko ir pastangų, o dokumentacijos keliais formatais palaikymas yra nepatogus. Dokumentacija yra nuolat atnaujinama, todėl dokumentų konvertavimą reikia atlikti dažnai.

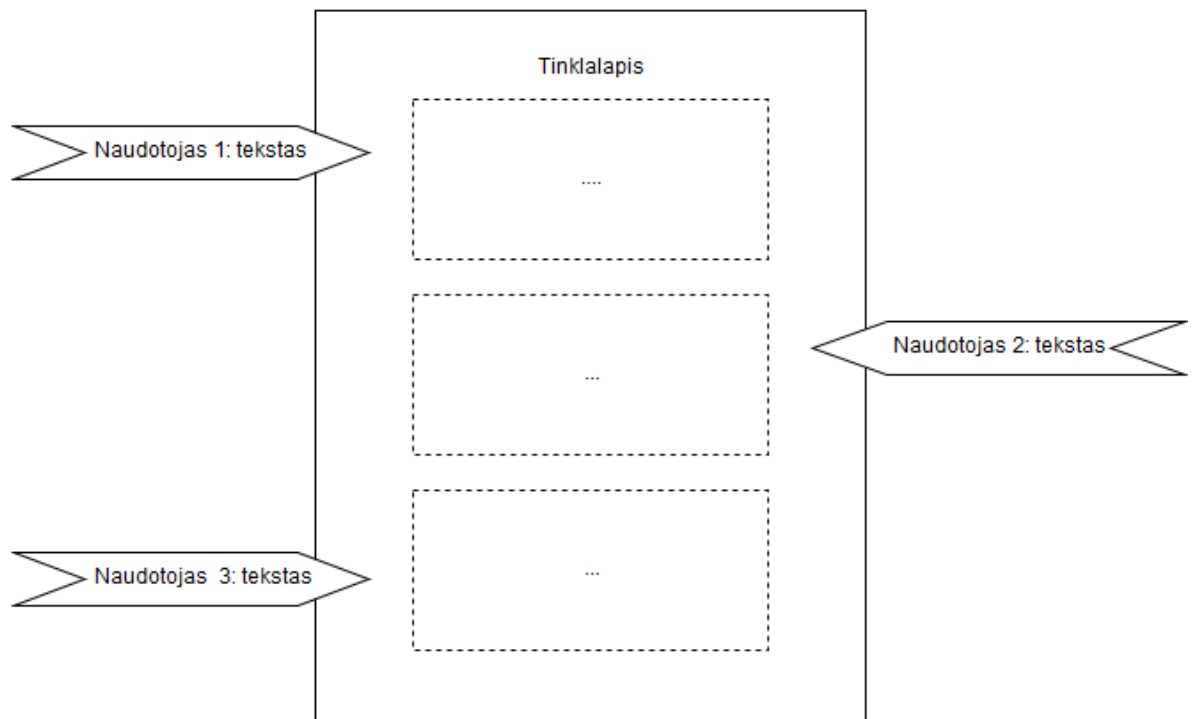
Rengiant dokumentaciją vikio sistemą, vieną dokumentą gali redaguoti keli naudotojai (žr. 2 pav.).



2 pav. Dokumentacijos rengimo procesas

Dokumentacijos dokumentams kurti ir redaguoti dokumentacijos rengėjai naudoja interneto naršyklę, o vikio programa, kuri yra instaliuota tam tikrame internetiniame serveryje, atlieka šių dokumentų atvaizdavimą. Programinės įrangos dokumentacijos rengėjas neturi jokio sąryšio su vikio programos kodu bei atvaizdavimo techninėmis detalėmis, nes sistemos dokumentaciją atvaizduoja vikio programa iškart po teksto įvedimo ir mygtuko „Užsaugoti“ paspaudimo.

Dokumentacijos rengėjai turi galimybę kurti kiekvieną dokumentą kartu (žr. 3 pav.), todėl visas dokumentacijos turinys gali susidėti iš kelių dokumentacijos kūrėjų indelio. Vikio sistema palaiko skirtingas duomenų versijas, todėl dokumentacijos kūrėjai gali peržiūrėti senesnes dokumentų versijas bei jas panaudoti iš naujo [14]. Gali pasitaikyti atvejų, kai tą patį sistemos funkcionalumo žingsnį keli dokumentacijos rengėjai aprašo skirtingai (pvz. žingsnis „Kaip prisijungti prie sistemos“ gali būti aprašytas įdedant sistemos paveikslėlį su prisijungimo langu arba ne, aprašant kokio ilgio gali būti įvedami laukai arba ne). Atsiradus sistemos pakeitimui ir tokio žingsnio dokumentacijoje atnaujinimo poreikiui, bus sunku surasti keliose vietose jis buvo aprašytas. O suradus užtruks laiko jį pakeisti. Viena iš tokios problemos sprendimų būdų galėtų būti nuorodos įterpimas su paaiškinančiu tekstu, pvz. „kaip prisijungti prie sistemos žr. 2 skyrių“. Tačiau šiuo atveju skyriai gali keistis ir tuomet reikės visose dokumento vietose atnaujinti nuorodą.

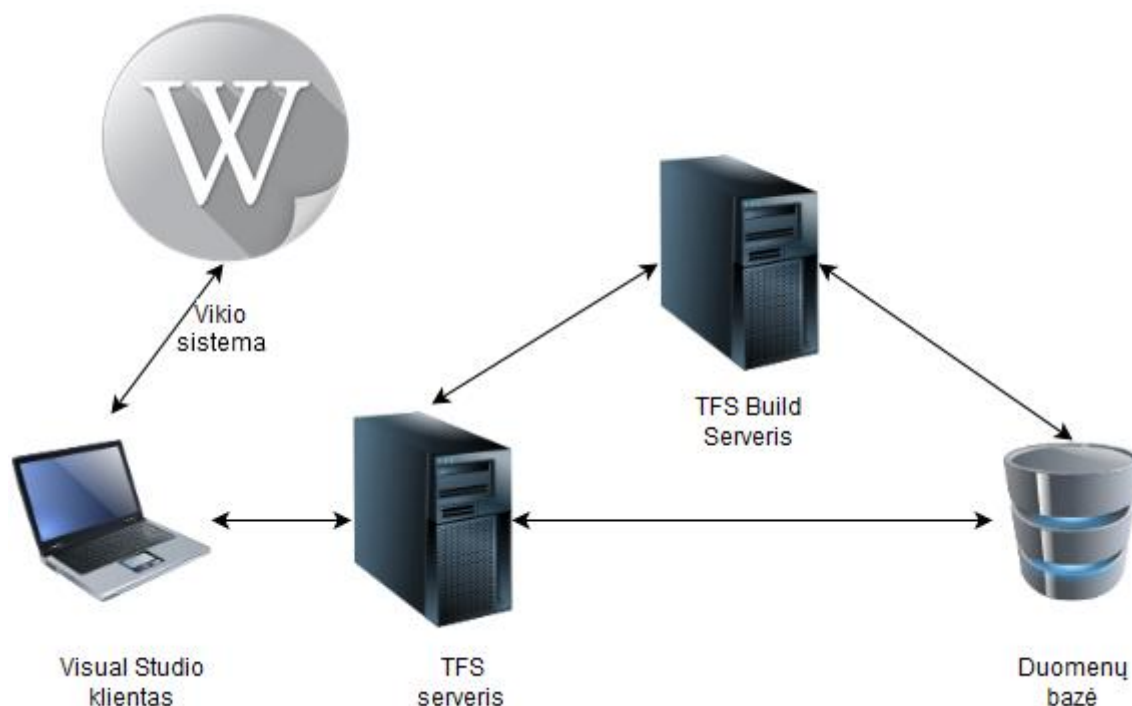


3 pav. Vikio dokumentų kūrimas

Vikio puslapiuose yra galimybė užsisakyti išsamų puslapio aprašymą (angl. *Rich Site Summary*) (RSS). Norėdamas gauti pranešimus apie turinio naujienas, naudotojas turi įsidiesti tam tikrą RSS programą, parinkti tinkamas naršyklės parinktis arba užsisakyti RSS siuntimą el. pašto dėžutėje nurodytu adresu [3. ].

### 1.5. Įmonės „IT Sprendimai“ programinės įrangos kūrimo architektūra

Įmonės „IT Sprendimai“ sistemos architektūra, kurioje yra suinstaliuota testinė viešųjų pirkimų sistema ir vikio puslapių sistema, susidaro iš naudotojo kompiuterio, TFS serverio, „TFS Build“ serverio, duomenų bazės ir vikio sistemos (žr. 4 pav.). Naudotojo kompiuteryje yra įdiegtas „Visual Studio“ klientas. Visas programinės įrangos kodas yra laikomas „Team Foundation serveryje“ (TFS). Šis serveris turi ryšį su „Build“ serveriu, kuris kompiluoja kodą. Duomenų bazė taip pat turi tiesioginį ryšį su TFS serveriais. „Team Foundation“ serveris įgalina versijų kontrolę.



4 pav. Sistemos architektūra

Įmonės „IT Sprendimai“ naudojamos sistemos architektūroje (žr. 4 pav.) vikio sistema yra susijusi su naudotojo kompiuteriu tiesioginiu ryšiu, o TFS serveris šios sistemos nemato. Sistemos naudotojų dokumentacija nėra saugoma skirtingomis versijomis tame pačiame serveryje kaip ir programinės įrangos kodas, todėl yra naudojama papildoma vikio sistema. Vikio sistema neatskiria dokumentacijos turinio nuo atvaizdavimo, nėra galimybės turėti skirtingo turinio ir skirtingo dizaino versijas tuo pačiu metu. Taip pat egzistuojanti sistema nesaugo skirtingų formatų dokumentacijos, todėl konvertuoti dokumentai gali pasimesti.

## 1.6. Dokumentacijos rengimo proceso problemos

Pagrindiniai esamo dokumentacijos rengimo proceso trūkumai yra šie:

- nėra galimybės visiškai atskirti dokumentacijos turinį nuo jo atvaizdavimo – viskas yra sujungta į vieną dokumentą. Tai apsunkina dokumentacijos valdymą ir atnaujinimą, nes kiekvieną kartą pakeitus ar pridėjus naują dokumentacijos elementą (tekstą, paveikslą, lentelę ar pan.) reikia pritaikyti atitinkamą dizainą. Jei dokumentacija yra saugoma keliais formatais, tai kiekvienam iš jų po atlikto turinio pakeitimo, reikia pritaikyti dizainą;
- dokumentacijos turinio konvertavimas į kitus formatus yra ribotas. Naudojant vikio puslapius turinį galima konvertuoti dažniausiai tik į PDF, tačiau toks konvertavimas

nėra patogus, nes reikia konvertuoti kiekvieną vikio puslapį atskirai. Konvertavimas į kitus formatus galimas nebent panaudojant papildomą programinę įrangą;

- sudėtingai valdomi pakeitimai skirtinguose dokumentacijos formatuose. Turint daugybę failų su skirtingais formatais reikia kiekvienoje iš jų pritaikyti turinio pakeitimą, o tai užima daug laiko bei pastangų;
- nėra galimybės patogiai valdyti dokumentacijos turinį, temos gali dubliuotis keliose dokumento vietose.

## 1.7. XML technologijos

Kadangi dabartinis įmonės „IT sprendimai“ dokumentacijos rengimo bei teikimo procesas nėra patogus dėl ankstesniame skyrelyje minėtų priežasčių, šiame skyrelyje bus panagrinėtos keletas kitų sistemų, kurios gali būti naudojamos dokumentacijai rengti bei teikti ir galėtų būti naudojamos sprendžiant kylančias problemas. Toliau pateiktas vikio puslapių palyginimas su XML formato dokumentais (žr. 1 lentelė). Lentelėje nurodyti kriterijai yra esminiai, kurie leidžia palyginti dvi technologijas pagal išnagrinėtą dokumentacijos rengimo procesą.

1 lentelė. Vikio puslapių ir XML formato dokumentų panaudojimo palyginimas

Kriterijus	Vikio puslapiai	XML architektūra
Galimybė transformuoti turinį į įvairius formatus.	Ganėtinai sudėtinga, nes konvertavimui į tam tikrus formatus reikia specialios programinės įrangos.	Turinys yra konvertuojamas į įvairius formatus, tačiau tam reikia turėti parengtą šabloną.
Galimybė panaudoti tą patį turinį kelis kartus dokumente.	Realizuojama nukopijuojant ir įklijuojant reikiamą turinį.	Gali būti realizuota įterpiančiais nuorodas į skirtingus xml dokumentus ar jų vietas.
Galimybė pritaikyti skirtingą dizainą skirtingiems dokumentams.	Nėra, arba reikalauja daug pastangų. Vikio puslapių dokumentai gali atrodyti tvarkingai, tačiau konvertavus į HTML formatą, dizainą reikia tvarkyti.	Labai lengvai įgyvendinama, tereikia kiekvienam formatui sukurti šabloną.

Galimybė nuolatos atnaujinti turinį ir atlikti transformavimą į įvairius formatus	Sudėtinga, nes kiekvieną kartą tai užimtų daug laiko. Reikia papildomos programinės įrangos ir kaskart atnaujinti dokumento dizainą.	Lengva ir patogiu transformuoti į įvairius formatus nuolatos. Tereikia turėti parengtą šabloną.
Technologijos kaina	Mokamos arba nemokamos viki puslapių sistemos.	Mokamos arba nemokamos XML pagrindu grįstos sistemos.
Kitų įrankių / technologijų panaudojimas, pvz. sistemos leidžiančios konvertuoti turinį į skirtingus formatus.	Dažniausiai reikalingi kiti įrankiai.	Dažniausiai nereikalingi kiti įrankiai.

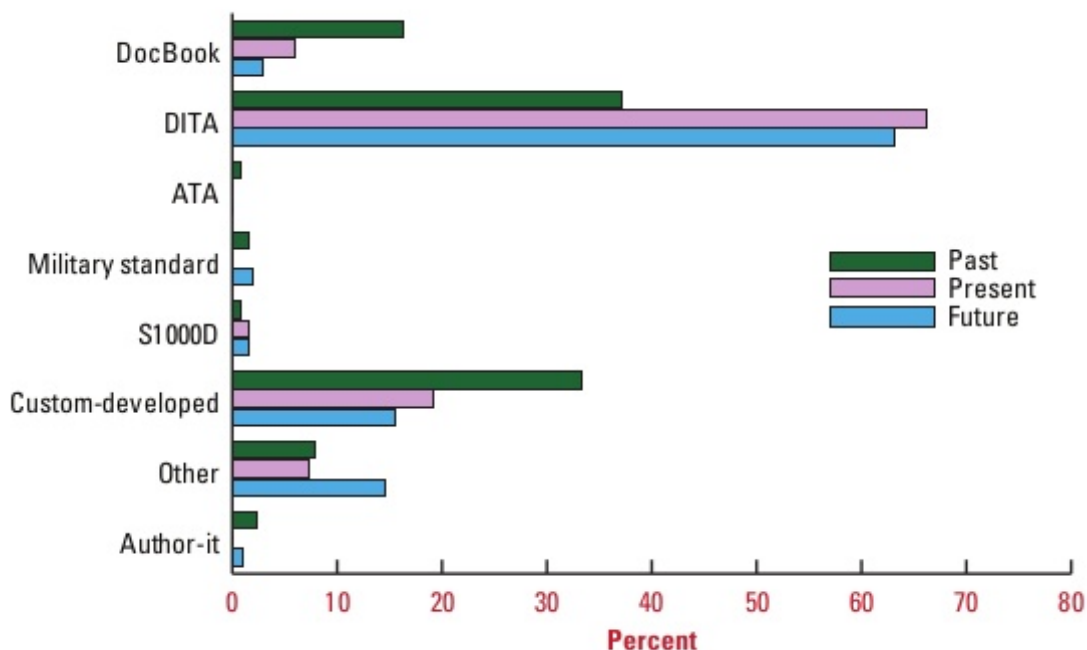
Dokumentų rengimas bei palaikymas, naudojant vikio sistemą, yra ganėtinai sudėtingas. XML technologija leidžia lengviau atnaujinti bei konvertuoti turinį. Reikalavimas, kurio reikia laikytis tam, kad XML dokumentai būtų paprastai konvertuojami į kitus formatus ir nereikėtų atlikti papildomo dizaino atnaujinimo kiekvieną kartą, yra šablonai. XML šablonai turi nusakyti elementų atvaizdavimo taisykles, eiliškumą ir vietą dokumente. Jeigu kiekvienam formatui turėsime sukurtą šabloną, tuomet pakeitus dokumento turinį, atvaizdavimas į visus reikiamus formatus vyks greitai ir sklandžiai, o tai leis dažnai keisti ir atnaujinti sistemos naudotojų dokumentaciją.

### 1.7.1. DITA architektūra

Viena iš plačiausiai paplitusių XML technologijos taikymo dokumentacijoms rengti priemonių yra Darvino informacijos pateikimo architektūra (DITA). Tai yra XML pagrindu grįsta architektūra, skirta kurti bei publikuoti dokumentaciją. Šios architektūros pagrindinis principas yra kurti turinį, kuris yra suskirstytas į tam tikrus modulius, o visa dokumentacija gali būti publikuojama įvairiais formatais. Kadangi ši architektūra remiasi XML technologija, dokumento dizainas yra atskirtas nuo jo turinio [12].

DITA technologija yra plačiai naudojama dokumentacijai ir kitiems dokumentams rengti įvairiuose sektoriuose. 2009 m. įmonė „Scriptorium“ atliko tyrimą, kuriuo bandė išsiaiškinti kokia dokumentacijos rengimo technologija yra labiausiai paplitusi. Anketinėje apklausoje dalyvavo daugiau nei 600 žmonių. Grafike (žr. 5 pav.) matyti, kad DockBook

technologija bus naudojama mažiau, o DITA technologija bus pritaikoma plačiau. Tyrimas parodė, kad DITA technologija yra populiariausia, palyginti su kitomis panašiomis elektroninio turinio rengimo sistemomis [7].



5 pav. Dokumentavimo technologijų panaudojimas

Taip pat, remiantis „easyDita“ paskelbtais teiginiais, padedančiais apsispręsti dėl DITA naudojimo, galima teigti, kad DITA architektūra gali būti pritaikyta rengti įvairių naudotojų dokumentaciją (žr. 2 lentelė) [11], nes tenkina daugelį programinės įrangos dokumentacijos procesui keliamų reikalavimų.

2 lentelė. DITA tinkamumas įmonei „IT sprendimai“

Teiginys	Ar tinka programinės įrangos dokumentacijai rengti?
Turinio pakartotinis panaudojimas (pvz. pritaikyti tam tikrą turinio elementą keliuose dokumentuose, bet nedubliuojant paties elemento)	TAIP
Turinio pateikimas keliomis kalbomis	TAIP (iš dalies)
Turinio pateikimas skirtingais formatais	TAIP

Nėra sudėtingo dizaino reikalavimų (kiekvienas puslapis neprivalo būti skirtingo dizaino)	TAIP
-------------------------------------------------------------------------------------------	------

DITA technologiją naudoja dauguma didžiųjų organizacijų visame pasaulyje [4]: „Adobe“, „Amazon“, „eBay“, „Ericsson“, „HTC“, „IBM“, „Intel“, „Lenovo“, „Microsoft“, „Oracle“, „Tieto“, „Xerox“ ir t.t. Tačiau DITA technologija negali būti pritaikyta visoms įmonėms vienodai, nes skiriasi įmonių verslo sritis, konvertuojamų failų formatų poreikiai ir stiliai. DITA technologijai gali būti pritaikyti skirtingi konvertavimo varikliai, kuriuos kiekviena įmonė prisitaiko pagal savo poreikius.

### 1.7.2. „DocBook“

Kita gana plačiai naudojama technologija dokumentacijai rengit XML formatu yra „DocBook“. Ji buvo sukurta 1991 m. susijungus dviem projektams: „HaL Computer Systems“ ir „O’Reilly & Associates“ (prieš tai vadinosi „O’Reilly Media, Inc“). Su laiku šios technologijos populiarumas didėjo, todėl 1998 m. OASIS organizacija (DocBook Technical Committee) pradėjo vystyti „DocBook“.

Pačioje „DocBook“ taikymo pradžioje ši technologija buvo pagrįsta SGML technologija, apibrėžta kaip Dokumento tipo apibrėžtis (*angl. Document Type Definition*) (DTD). Nuo 4.1 versijos „DocBook“ buvo leidžiama tiek SGML tiek ir XML žodynu (XML žymėjimo kalba). [10]

Buvo sukurta keletas „DocBook“ versijų [16] :

- 2001 m. sausio mėn. „DocBook“ 2.2 patvirtinta OASIS standartu
- 2006 m. spalio mėn. „DocBook“ 4.5 patvirtinta OASIS standartu
- 2009 m. lapkričio mėn. „DocBook“ 5.0 patvirtinta OASIS standartu.

„DocBook“ technologija yra patogi, nes dokumentai gali būti transformuojami į įvairius formatus: HTML, XHTML, EPU, PDF, pagalbos puslapius ir pan. Turinys yra ženklinamas XML kalba, jis gali būti kuriamas bei redaguojamas panaudojant bet kurią tekstinę rengyklę. Daugiausia „DocBook“ technologija yra naudojama rašyti knygoms, straipsniams, programinės įrangos dokumentacijai [15]. Kadangi „DocBook“ turinys yra ženklinamas XML kalba, todėl jis yra atskiriamas nuo grafinio formatavimo taisyklių. Panašiai daroma taikant anksčiau aprašytą DITA technologiją.



### 1.7.3. DITA ir „DocBook“ technologijų palyginimas

DITA ir „DocBook“ technologijos turi bendrą panašumą – tai XML pagrindu grįstos dokumentacijos kūrimo bei rengimo technologijos. Tačiau kaip ir buvo anksčiau minėta, šios technologijos suteikia tik pagrindinį dokumentų konvertavimo variklį, kurį kiekviena įmonė turi prisitaikyti pagal savo poreikius. Šiame skyrelyje bus pateiktas šių technologijų palyginimas, aptariant šiuos kriterijus: turinys, turinio konvertavimas į kitus formatus, pritaikomumas tam tikriems poreikiams, panaudojimo skalė (naudotojų skaičius) [1], [15].

Jeigu turinys yra nuoseklus, kaip ir yra rašomos knygos ar straipsniai – geriau rinktis „DocBook“. Ši technologija neakcentuoja turinio išskaidymo į skirtingas temas, todėl nėra poreikio turėti daug failų su skirtingomis temomis. Jei turinys yra išskaidytas į kitas smulkesnes temas, kurios gali kartotis, tuomet geriau tinka DITA architektūra, nes ji palaiko žemėlapių funkciją. Panaudojant DITA žemėlapius galime lengvai įterpti nuorodas į reikiamas temas ir taip turėti skirtingus žemėlapius, kurie vėliau bus konvertuojami į skirtingų tipų dokumentaciją.

Turinio konvertavimo į kitus formatus aspektu, abi technologijos palaiko XSL, todėl konvertavimas į kitus formatus galimas abiem atvejais. „DocBook“ stiliai, naudojami konvertuojant dokumentus, yra gerai dokumentuoti, o DITA yra naujesni, todėl dokumentacijos yra šiek tiek mažiau. Turint nedaug programavimo patirties, „DocBook“ siūlo jau parengtus stiliaus parametrus, kurie gali būti lengvai pritaikomi. Tiek „DocBook“, tiek ir DITA stiliai yra nuolatos palaikomi ir atnaujinami.

Jei poreikis pritaikyti įvairius nustatymus ir gaires pagal save yra didelis, tuomet geriau naudoti DITA, nes ši technologija palaiko naujų gairių (XML kalbos žodis) pridėjimą. „DocBook“ technologija jau turi daug įvairių gairių, kurias galima panaudoti, todėl jei nėra didelio poreikio pritaikyti technologiją konkrečiai sričiai, tuomet yra paprasčiau rinktis „DocBook“.

Jei projektas, kuriame dokumentacijai rengti bus taikoma XML pagrindu grįsta architektūra, yra mažas, „DocBook“ yra geresnis pasirinkimas, nes tokia architektūra reikalauja mažiau programavimo, konfigūravimo žinių. Tačiau jei projektas yra didelis, tuomet ir jo specifiška gali būti platesnio konteksto (gali būti reikalingi skirtingi dokumentacijos šablonai kiekvienam failų formatui, kurie turi būti skirtingai atvaizduojami bei praplečiami pagal savo poreikius), DITA technologija yra labai pravarti, nes ji leidžia pritaikyti dokumentacijos kūrimo procesą savo poreikiams ir kurti reikiamas gaires.

Kadangi įmonė „IT Sprendimai“ kuria programinę įrangą ir jos dokumentaciją, kurios turinys gali kartotis, todėl DITA žemėlapių funkcija turėtų išspręsti informacijos dubliavimo

problema. Taip pat DITA technologija yra patogesnė taikant skirtingų formatų failų dizainą, todėl ši technologija ir bus pritaikyta įmonėje „IT Sprendimai“.

## **1.8. DITA architektūros analizė**

DITA architektūra yra lengvai konfigūruojama pagal individualius įmonės poreikius, palaiko žemėlapių funkciją ir leidžia atskirti programinės įrangos dokumentacijos turinį nuo atvaizdavimo. Kadangi ši technologija turėtų padėti išspręsti įmonės „IT Sprendimai“ problemas, kituose skyreliuose paanalizuokime plačiau šią technologiją.

### **1.8.1. OASIS organizacija**

DITA technologiją standartizuoja „Organization for the Advancement of Structured Information Standards“ (OASIS). Tai yra ne pelno siekiantis konsorciumas, kuris standartizuoja interneto išteklius tokiose srityse kaip saugumas, daiktų internetas, debesų inžinerija, turinio valdymo technologijos ir kt. Šį konsorciumą sudaro daugiau kaip 5000 dalyvių, kurie atstovauja daugiau kaip 600 organizacijų bei kiti nepriklausomi asmenys visame pasaulyje (daugiau kaip 65-iose valstybėse).

OASIS pradžia galima laikyti 1993 m., nors tuomet egzistavo SGML Open (konsorciumas, kuris buvo atsakingas už techninių gairių rengimą, kurios buvo skirtos „Standard Generalized Markup Language“ (SGML) kalbai). 1998m. šis konsorciumas pakeitė pavadinimą bei praplėtė sritį ir tapo OASIS. [16]

Šiuo metu OASIS yra patvirtinęs daugiau kaip 80 standartų. Plačiausiai paplitę yra šie standartai: „AMQP“, „CAP“, „DITA“, „DocBook“, „ebXML“ (ISO 15000), „EDXL“, „MQTT“, „OpenDocument Format“ (ISO 26300), „OSLC“, „SAML“, „WS-Security“ ir „XACML“. [16]

DITA technologija yra viena iš plačiausiai paplitusių standartų, kuriuos patvirtino OASIS. Tai leidžia neabejoti jos taip pat plačiu panaudojimu įvairiose organizacijose, o tai suteikia galimybę nuolat atnaujinti bei palaikyti DITA technologiją.

### **1.8.2. DITA OASIS standartų apžvalga**

DITA technologija yra nuolat atnaujinama. Šio darbo rengimo ir metodo kūrimo metu yra naudojama naujausia DITA 1.3 versija. Iš DITA standartų evoliucijos matyti, kurie standartai, aktualūs ketinant panaudoti programinės įrangos dokumentaciją kaip mokomąją medžiagą, buvo pritaikyti plėtojant DITA technologiją.

2005 m. DITA 1.0 versija buvo patvirtinta OASIS standartu.

2007 m. rugpjūčio mėn. DITA 1.1 buvo patvirtinta OASIS standartu, kartu su „Bookmap“ specializacija.

2008 m. DITA 1.2 buvo patvirtinta OASIS standartu. DITA 1.2 turėjo galimybę kurti mokymosi objektus, atitinkančius kitus e.mokymosi standartus, pvz. „SCORM“.

2015m. gruodžio mėn. DITA 1.3 patvirtinta OASIS standartu. Šioje versijoje pridėta problemų sprendimo tema, atnaujinimų valdymas bei kiti pakeitimai. [6]

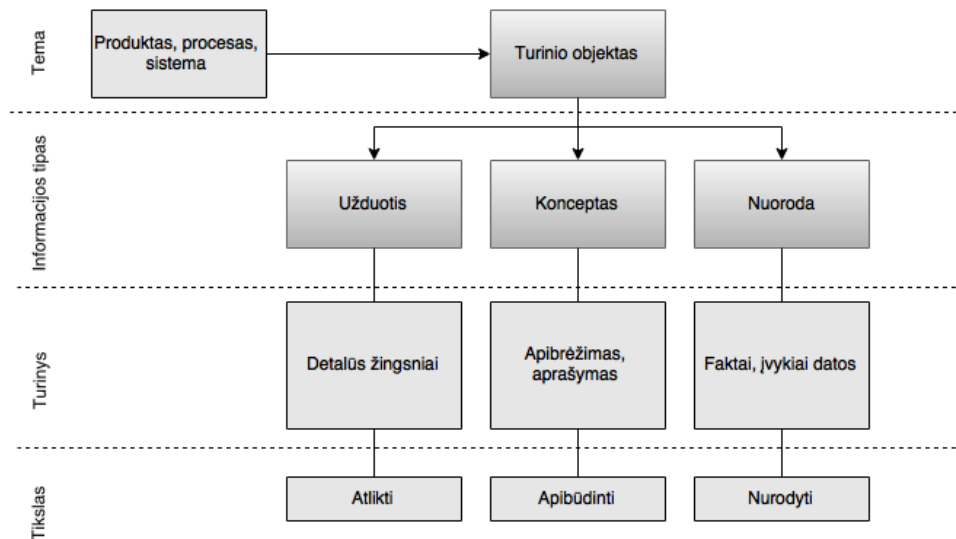
### 1.8.3. DITA architektūros pagrindiniai principai

DITA architektūra pasižymi trimis ypatybėmis [2]:

- **turinio pakartotinis panaudojimas.** Kadangi DITA yra į temas (modulius) orientuota architektūra, pakartotinis temų panaudojimas turinyje yra svarbus DITA privalumas. Pakartotinis panaudojimas atliekamas panaudojant DITA žemėlapius, turinio nuorodas bei sąlyginį apdorojimą;
- **paveldimumas.** DITA turinys gali būti lengvai paveldimas ir tam nereikės atskirai perprojektuoti turinio publikavimo proceso. Pavyzdžiui, tam tikras poskyris gali automatiškai paveldėti tėvinės temos stilių;
- **ryšys tarp turinio elementų.** DITA turinys automatiškai gali turėti nuorodas į kitas temas, kurios turi susijusių informaciją.

DITA remiasi turinio skaidymu į mažesnes dalis arba temas, kur viena tema turi aprašyti tam tikrą konkretų dalyką. Aprašant tam tikrą temą, yra rekomenduojama kiek įmanoma minimizuoti informaciją bei pateikti ją aiškiai ir suprantamai skaitytojui. Yra išskiriami trys informacijos tipai, pagal kuriuos dokumentacija gali būtų klasifikuojama (žr. 6 pav.) [21]:

- Užduotis – paaiškinimas kaip atlikti tam tikrus žingsnius sistemoje (pvz. kaip užsaugoti failą kompiuteryje);
- Konceptas – konceptuali informacija apie tam tikrą temą, pavyzdžiui pavadinimas, aprašymas, paveikslėlis ar grafinė diagrama;
- Nuoroda – nuoroda (-os) į kitas panašias temas.



6 pav. DITA koncepcija

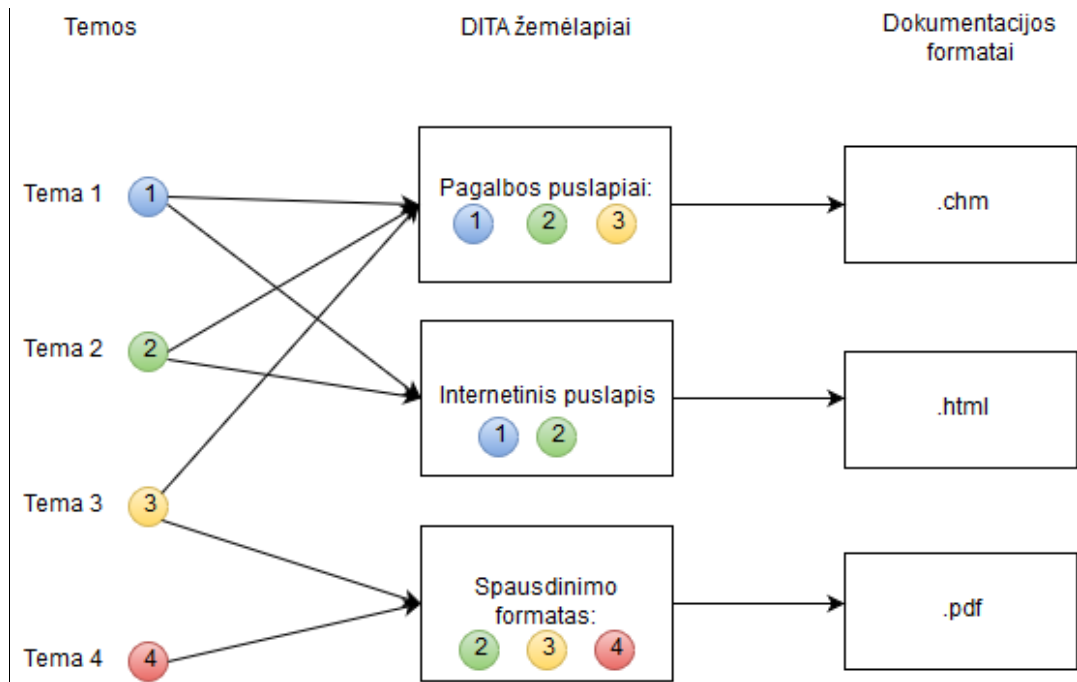
Kiekvienas informacijos tipas turi šiuos elementus: pavadinimas, aprašymas, įvadas, dėstymas (žr. 3 lentelė). DITA dokumentas privalo turėti bent vieną temą.

3 lentelė. DITA dokumentavimo struktūra

	<b>Užduotis</b>	<b>Konceptas</b>	<b>Nuoroda</b>
<b>Struktūra</b>	<b>Antraštė:</b> Pavadinimas Aprašymas Įvadas <b>Turinys:</b> Turinio objektas <b>Susijusios nuorodos</b>	<b>Antraštė:</b> Pavadinimas Aprašymas Įvadas <b>Turinys:</b> Turinio objektas <b>Susijusios nuorodos</b>	<b>Antraštė:</b> Pavadinimas Aprašymas Įvadas <b>Turinys:</b> Turinio objektas <b>Susijusios nuorodos</b>
<b>Kitų neprivalomų elementų pavyzdžiai (turinio elemente)</b>	Kontekstas Sąlygos Žingsniai Žingsnių elementai Rezultatas Pavyzdžiai Kiti reikalavimai	Sekcija Paragrafas Sąrašas Lentelės Paveikslėliai Pavyzdžiai	Sekcija Paragrafas Savybių sąrašai Lentelės Sintaksė Pavyzdžiai

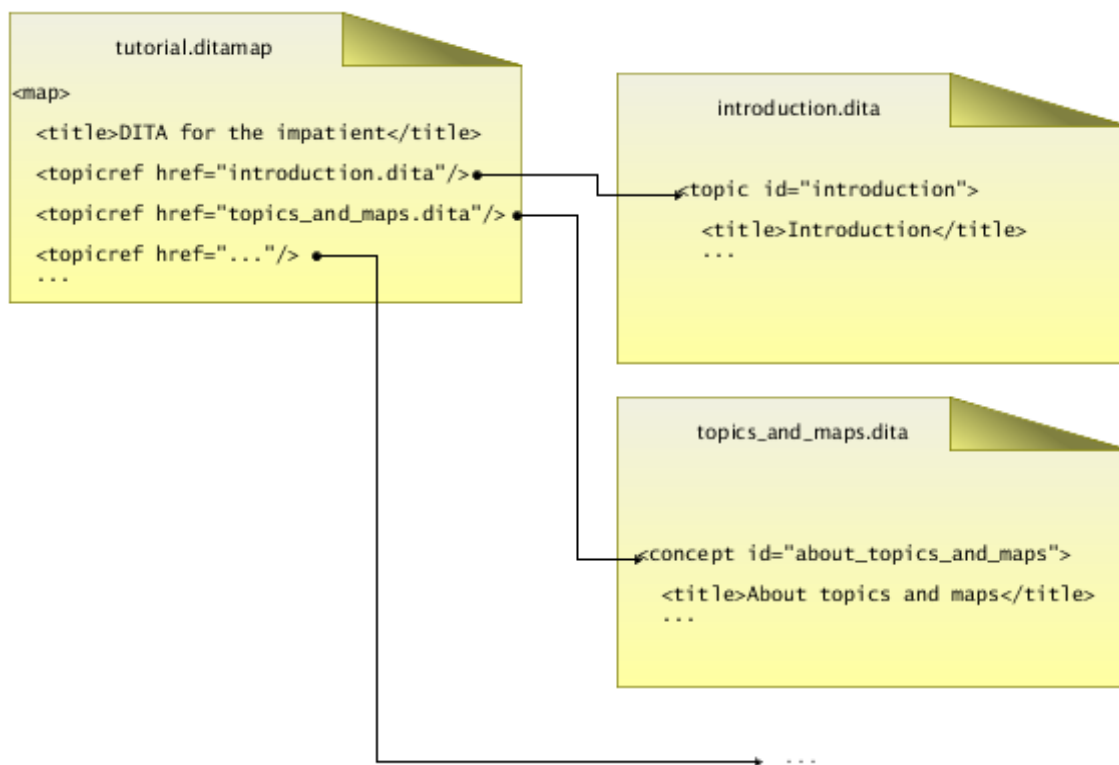
#### 1.8.4. DITA žemėlapiai

DITA architektūra patogi tuo, kad leidžia informacija skaidyti į temas, o vėliau viską apjungti bei pakartotinai panaudoti žemėlapiuose (žr. 7 pav.) [2]. Vietoj to, kad rašytume du atskirus dokumentus skirtingoms auditorijoms, galime parašyti vieną ir konvertuoti į skirtingus formatus, panaudojant DITA technologiją.



7 pav. DITA žemėlapiai

DITA žemėlapiai yra sudaromi įtraukiant tam tikro dokumento nuorodą į žemėlapių failą. Kaip buvo minėta anksčiau, DITA yra XML kalba pagrįsta architektūra, todėl dokumentai yra rašomi XML kalba (žr. 8 pav.).



8 pav. DITA failų sąryšis

Paveikslėlis paimtas iš: <http://www.xmlmind.com/tutorials/DITA/>

### 1.8.5. DITA publikavimo mechanizmai

Failai, sukurti panaudojant DITA yra konvertuojami į skirtingus formatus. Be konvertavimo variklio (t.y. programinės įrangos, atsakingos už konvertavimo dalį į skirtingus formatus), DITA failų plėtiniai yra tokie [18]:

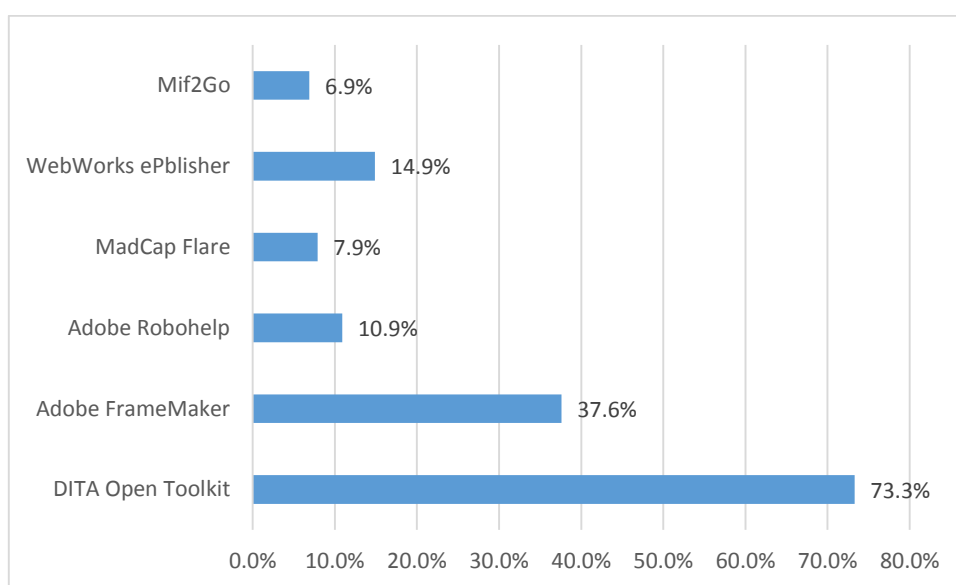
- .xml – dažniausiai rekomenduojamas formatas aprašyti tam tikrą poskyrį;
- .dita – dažniausiai rekomenduojamas formatas aprašyti tam tikrą didesnę apibendrinantį skyrį;
- .ditamap – dažniausiai rekomenduojamas formatas aprašyti DITA žemėlapi.

Norint pateikti dokumentaciją skaitymui ar naršymui patogiu formatu (pvz. PDF, HTML, CHM, ODT ir pan.), DITA failai turi būti konvertuoti. Yra ganėtinai nemažai publikavimo mechanizmų, kurie siūlo publikavimą į skirtingus, tačiau dažniausiai labai panašius, failų formatus. Toliau bus aptarti keli mechanizmai.

## DITA Open Toolkit

DITA „Open Toolkit“ (DITA-OT) yra nemokama („Apache License“ 2.0. licencija [8]) DITA dokumentų konvertavimo sistema. Ši sistema buvo sukurta kompanijos „IBM“ 2005m. [20] Dabartinė DITA-OT versija 2.1.2 palaiko 1.0, 1.1 ir 1.2 DITA versijas, o DITA 1.3 (šiuo metu naujausia versija) palaiko tik peržiūrą. DITA-OT taip pat palaiko ir prieš OASIS standartus atsiradusią DITA versiją „dita132“.

DITA „Open Toolkit“ yra Java kalba parašyta sistema, kuri transformuoja DITA žemėlapių bei tam tikrų skyrių failus į šiuos formatus: PDF, HTML, TocJS, HTML Help, Eclipse Help, Eclipse Content, Java Help, ODT, Word RTF, DocBook ir troff. [8]



9 pav. Publikavimo mechanizmų panaudojimo tyrimas, 2009m.

2009 m. „Write point“ kompanijos atliktas tyrimas parodė, kad net 73,3% respondentų naudoja DITA „Open Toolkit“ sistemą (žr. 9 pav.). Tyrime dalyvavo 137 respondentai. [23] Tai leidžia padaryti išvadą, kad DITA-OT yra plačiai naudojamas mechanizmas, kuris yra nuolatos atnaujinamas ir palaikomas.

## „DITA2Go“

„DITA2Go“ yra nemokama Windows programa, kuri konvertuoja DITA dokumentus į kitus formatus. Sistema yra sukurta remiantis XSLT bei Java kalbomis. „DITA2Go“ palaiko tokias DITA versijas: 1.1 ir beveik visas 1.2 versijos funkcijas. Ši sistema veikia Windows 2000/XP/Vista/7/8 aplinkose.

„DITA2Go“ dokumentacijoje yra nurodomas publikavimo greitis, kuris gali būti ~2 MB per minutę (tai yra maždaug 250 puslapių). Pagrindiniai publikavimo formatai yra šie: HTML, XHTML, Flat XML, DocBook, RTF, MS On-line Help ir OmniHelp. [9]

## **1.9. Išvados**

1. Programinės įrangos dokumentacija gali būti skirtingų tipų. Tam tikros dokumentacijos dalys, pvz. kaip prisijungti prie sistemos, gali būti panaudotos keletą kartų rengiant skirtingų tipų dokumentacijos dokumentus. Todėl kyla dokumentacijos turinio palaikymo ir atnaujinimo problema, kuomet ta pati informacija yra dubliuojama skirtingose vieno ar kelių dokumentų vietose.
2. Įmonės „IT Sprendimai“ esamas dokumentacijos procesas nėra efektyvus ir todėl kyla įvairios problemos: nėra galimybės atskirti programinės įrangos dokumentacijos turinį nuo atvaizdavimo, turinio konvertavimas į skirtingus formatus yra ribotas, sudėtingai valdomi pakeitimai.
3. Iš išanalizuotų technologijų, DITA gali būti panaudota kaip tinkama dokumentacijai rengti, nes turi patogią žemėlapių funkciją, kuri leidžia panaudoti tą patį turinį daugybę kartų. DITA-OT yra vienas iš mechanizmų, kuris yra nuolatos atnaujinamas bei palaiko failų konvertavimą į skirtingus formatus.



## **2. PROGRAMINĖS ĮRANGOS DOKUMENTACIJOS RENGIMO METODAS**

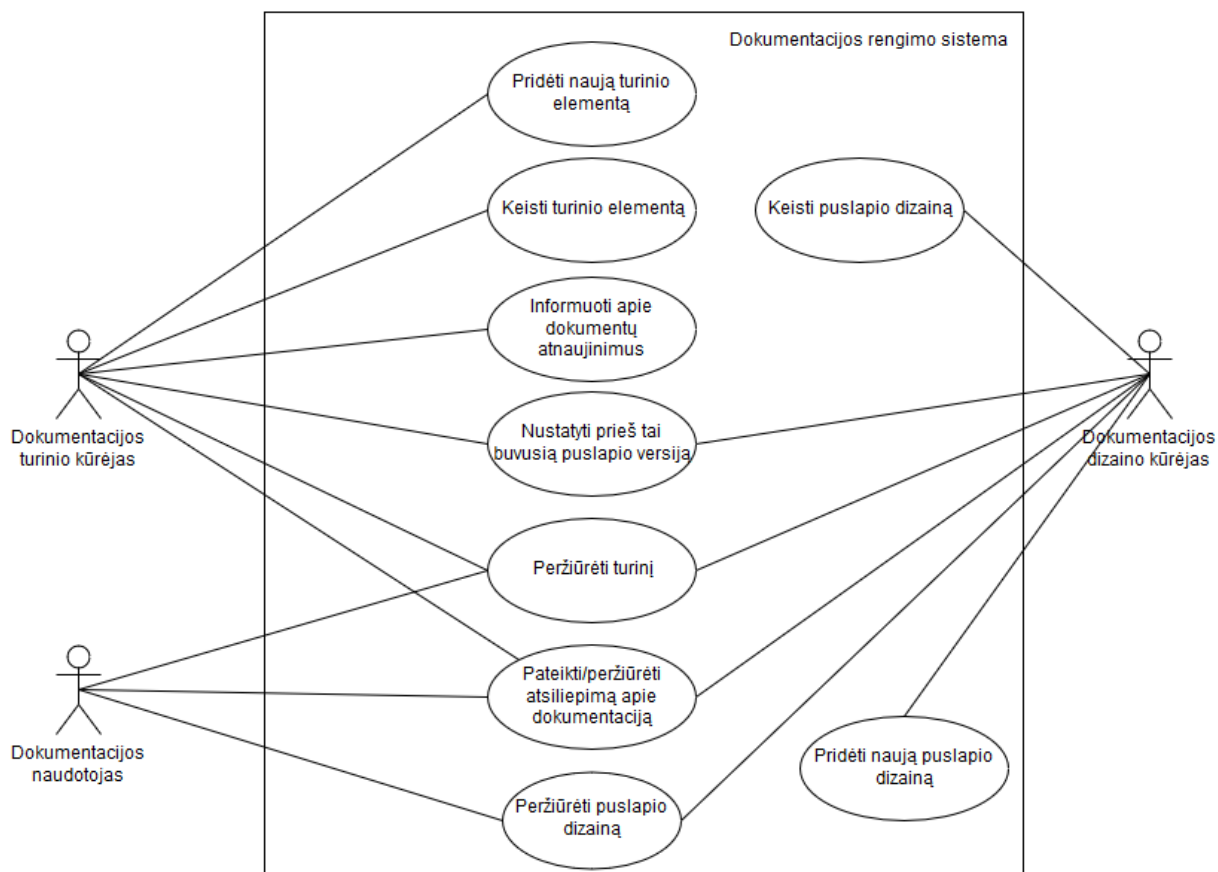
Programinės įrangos dokumentacijos kūrimas, panaudojant vikio puslapius turi tam tikrų trūkumų, kurie yra aptarti ankstesniame skyriuje. Šiame skyriuje aprašyti su dokumentacija susiję dalyviai, kurie tiesiogiai dalyvauja dokumentacijos rengimo procese, sukurtas programinės įrangos dokumentacijos kūrimo metodas kartu su siūlomomis naudoti technologijomis bei apžvelgtas sistemos infrastruktūros projektas, kuris paaiškins kaip atrodys sistema po technologijų įdiegimo ir pritaikymo.

### **2.1. Dokumentacijos proceso dalyviai**

Dokumentacijos rengimo procese dalyvauja kelios dokumentacijos dalyvių grupės:

- dokumentacijos turinio kūrėjai (sistemos analitikai, testuotojai, programuotojai, įmonės atstovai ir pan.);
- dokumentacijos dizaino kūrėjai;
- dokumentacijos naudotojai;

Dažnai dokumentaciją rengia daugiau nei vienas asmuo ir atlieka tai tuo pačiu metu. Kiekvienai dalyvių grupei, dalyvaujančiai programinės įrangos dokumentacijos rengimo procese, kuris realizuojamas tam tikra sistema, būdingos tam tikros veiklos. Kai kurias jų gali atlikti visi dalyviai, kai kurias tik tam tikrai grupei priklausantys dalyviai (žr. 10 pav.).

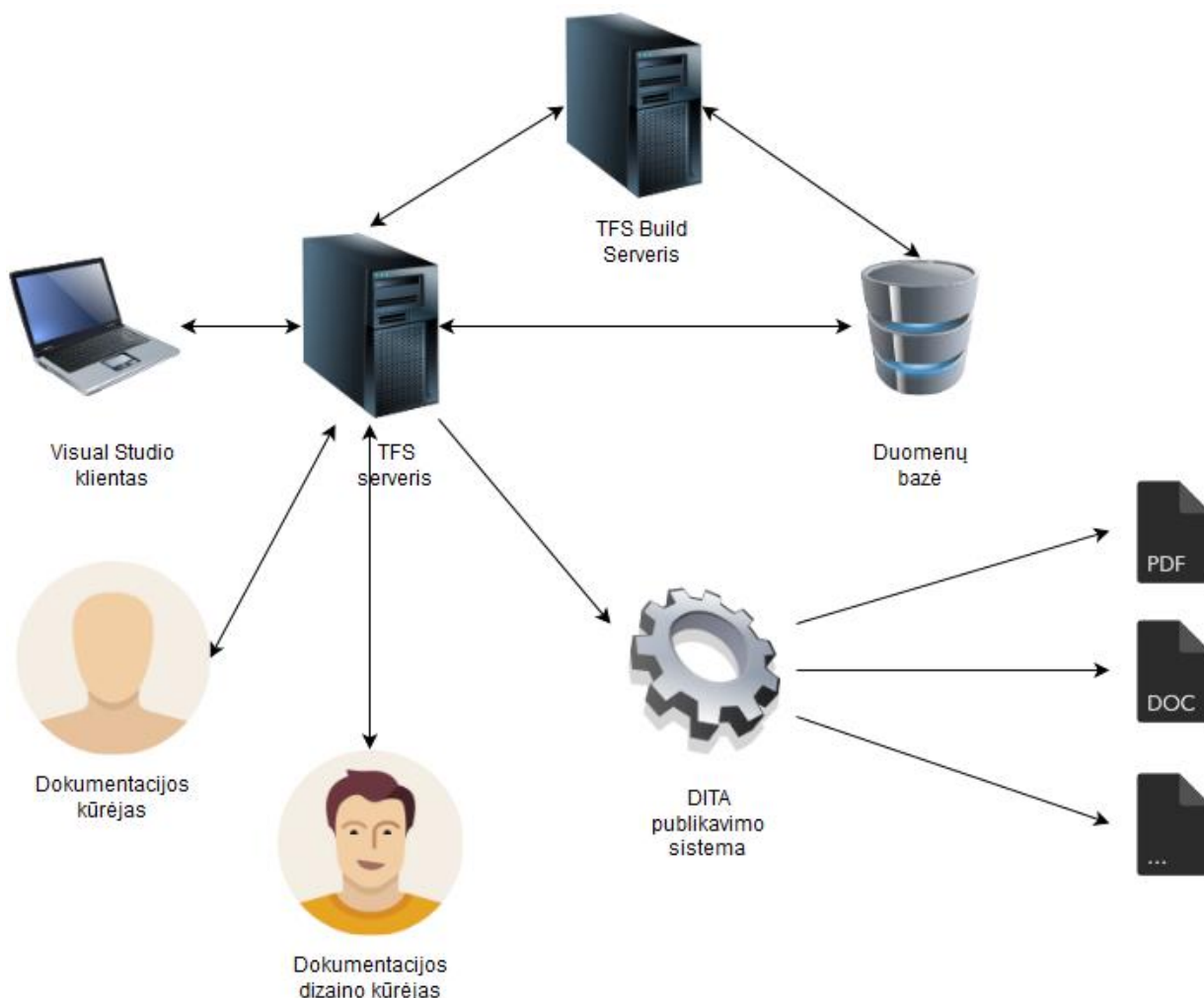


10 pav. Dokumentacijos rengimo sistemos panaudojimo atvejų modelis

Pagal dokumentacijos rengimo sistemos panaudojimo atvejų modelį (žr. 10 pav.), už dokumento turinį atsakingas dokumentacijos kūrėjas bendradarbiauja tiek su dizaino kūrėju, tiek su tiesioginiu dokumentacijos naudotoju. Dokumentacijos dizaino kūrėjas taip pat gali bendradarbiauti su turinio kūrėju ir dokumentacijos naudotoju, tačiau šis dalyvis, skirtingai nuo dokumentacijos turinio kūrėjo, yra atsakingas vien tik už dokumentacijos dizaino sukūrimą ir keitimą. Tiesioginis dokumentacijos naudotojas turi galimybę bendrauti su dokumentacijos turinio kūrėju ir dokumentacijos dizaino kūrėju ir jei turi tam tikrų pastabų apie dokumentacijos turinio ar dizaino kokybę, gali pranešti savo pastabas dokumentacijos dizaino ar turinio kūrėjui.

## 2.2. DITA integracija į sistemą

DITA gali būti integruota į esamą įmonės „IT Sprendimai“ sistemos architektūrą bei DITA kodas gali būti saugomas jau egzistuojančiame „Team Foundation“ serveryje, kuriame ir laikomas visas kitas programinės įrangos kodas. Pagrindinis skirtumas tarp buvusios įmonės „IT Sprendimai“ sistemos architektūros ir naujos yra tas, kad vietoje vikio sistemos yra naudojama DITA publikavimo sistema, kuri turi integraciją su TFS serveriu bei gali eksportuoti failus skirtingais formatais (žr. 11 pav.).



11 pav. Sistemos architektūra ir pritaikyta DITA technologija

DITA sistemai funkcionuoti reikalinga tokia programinė įranga [8]:

- „Java Runtime Environment“ (JRE) ar JDK 7 arba vėlesnė versija;
- „Ant“, versija 1.7.1 arba vėlesnė;
- „Saxon“ XSLT procesorius, versija 9.1.0.8;
- Operacinė sistema – Windows, Linux arba Mac OS X.

Įmonė „IT“ Sprendimai naudoja Windows operacinę sistemą (kuri yra mokama), o visa kita DITA technologijai sėkmingai funkcionuoti reikalinga programinė įranga nemokama.

### 2.3. DITA sistemos diegimo žingsniai

DITA sistemai įdiegti, reikia atlikti žingsnius, kurie yra nurodyti lentelėje (žr. 4 lentelė). Visi veiksmai yra išvardinti eilės tvarka. Šiame darbe panaudota naujausia DITA publikavimo sistemos versija – DITA „Open Toolkit“ 2.2.

Nr.	Žingsnio aprašymas	Atsakingas asmuo (vaidmuo)
1	Atsisiųsti naujausią DITA publikavimo sistemą iš: <a href="https://github.com/dita-ot/dita-ot/releases">https://github.com/dita-ot/dita-ot/releases</a> . Jei naudojama operacinė sistema yra Windows ir kiti komponentai veikia Windows aplinkoje, DITA Open Toolkit konvertavimo sistemą taip pat reiktų rinktis skirtą Windows operacinei sistemai.	Sistemos architektas
2	Išpakuoti parsisiųstą dita-ot failą į aplanką: D:\dita	Sistemos architektas
3	Sukurti .ditamap failus (DITA žemėlapius) bei patalpinti į: D:\dita\dita-ot-2.2.0\samples aplanką.	Dokumentacijos turinio kūrėjas (sistemos analitikas)
4	Sukurti .xml failus (DITA turinį) bei patalpinti į: D:\dita\dita-ot-2.2.0\samples\tasks aplanką.	Dokumentacijos turinio kūrėjas (sistemos analitikas)
5	Sukurti .xsl failus bei patalpinti atitinkamai į tam tikro failo konvertavimo mechanizmo aplanką, pvz. D:\dita\dita-ot-2.2.0\plugins\org.dita.pdf2\Customization	Dokumentacijos dizaino kūrėjas
6	Jei DITA turinys jau paruoštas, tuomet konvertuoti pateiktą dokumentacijos turinį į skirtingus formatus (kurie yra nurodyti scenarijuje, t.y. PDF, HTML ir CHM), paleidžiant šią komandą komandinėje eilutėje: D:\dita\dita-ot-2.2.0\bin ant -f D:\dita\dita-ot-2.2.0\samples\ant_sample\sample_all.xml	Sistemos architektas, Dokumentacijos turinio kūrėjas (sistemos analitikas)
7	Peržiūrėti D:\dita\dita-ot-2.2.0\out katalogą, kuriame turi būti konvertuoti visi formatai.	Sistemos architektas, Dokumentacijos turinio kūrėjas (sistemos analitikas)
8	Pridėti D:\dita\ katalogą į „Team Foundation“ serverį.	Sistemos architektas

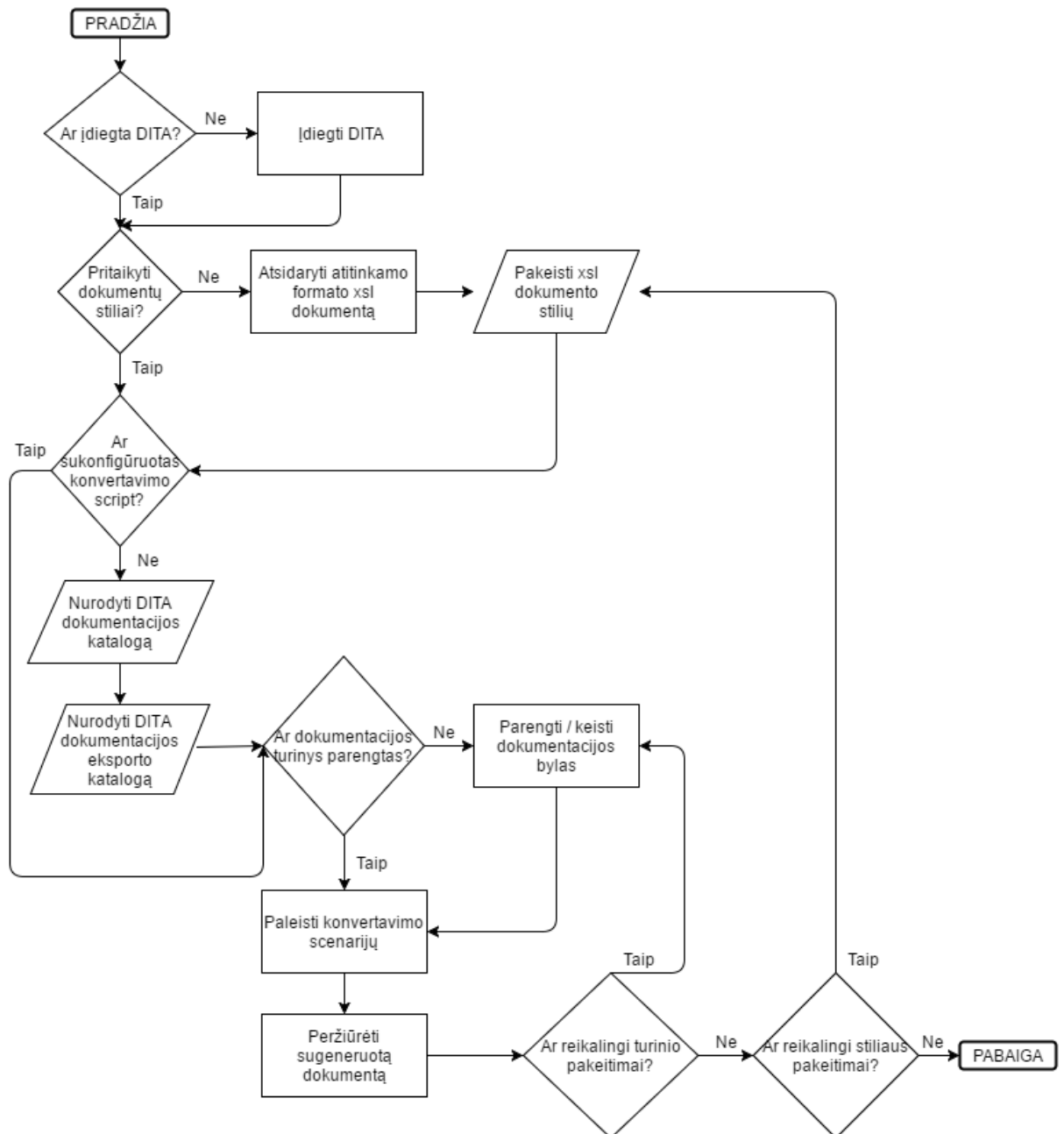
## 2.4. Metodas programinės įrangos dokumentacijai rengti

DITA technologijai kartu su sukurtomis priemonėmis, scenarijumi (angl. *script*) – programa, sudaryta iš interpretavimui skirtų komandų ir duomenų bazės procedūra, sudaromas naujas programinės įrangos dokumentacijos rengimo metodas. Scenarijus leidžia vienos komandos paleidimu konvertuoti dokumentacijos failus į skirtingus formatus. Remiantis įmonės „IT Sprendimai“ dokumentacijos formatų poreikiu, scenarijus konvertuoja programinės įrangos dokumentaciją į tris formatus: HTML (patogų skaitymui internete), PDF (patogų spausdinimui) ir CHM (patogų skaitymui kai nėra interneto ryšio, nes gali būti pridėtas prie sistemos ir atsidaryti paspaudus pagalbos mygtuką). Scenarijus taip pat konvertuoja visą dokumentacijos turinį į tris skirtingus dokumentacijos tipus: sistemos administratoriaus dokumentacija, pradedančiojo sistemos naudotojo dokumentacija ir pažengusio naudotojo dokumentacija. Duomenų bazės procedūra palengvina duomenų bazės lentelių dokumentacijos rengimą. Ši procedūra sugeneruoja jau dokumentuoti paruoštą failą, kuriame reikia pridėti tik laukų aprašymus, o visi nurodytos lentelės laukai ir jų tipai yra dokumentuoti automatiškai.

Metodas apibrėžia tiek sistemos paruošiamuosius darbus (kai DITA ir pagalbinės priemonės nėra suinstaliuotos), tiek ir dokumentacijos palaikymo ir atnaujinimo galimybes (žr. 12 pav.). Pirmiausia reikia įdiegti DITA sistemą. Tada reikia patikrinti, ar kiekvienas formatas turi apibrėžtus stilius (paraščių lygiavimas, paveikslėlių centravimas, fono spalva, šrifto dydis ir pan.), nes dokumentacijos konvertavimo į skirtingus formatus metu, bus panaudojami dokumento stiliai. Sukurto scenarijaus parametruose turime nurodyti esamą dokumentacijos katalogą (kuriame dokumentacija bus saugoma XML formatu) ir aplanką, į kurį bus eksportuojama dokumentacija trimis skirtingais formatais (iš viso 9 produktai)::

- CHM formatu užsaugota sistemos administratoriaus dokumentacija;
- CHM formatu užsaugota pradedančiojo naudotojo dokumentacija;
- CHM formatu užsaugota pažengusio naudotojo dokumentacija;
- PDF formatu užsaugota sistemos administratoriaus dokumentacija;
- PDF formatu užsaugota pradedančiojo naudotojo dokumentacija;
- PDF formatu užsaugota pažengusio naudotojo dokumentacija;
- HTML formatu užsaugota sistemos administratoriaus dokumentacija;

- HTML formatu užsaugota pradedančiojo naudotojo dokumentacija;
- HTML formatu užsaugota pažengusio naudotojo dokumentacija;



12 pav. Programinės įrangos dokumentacijos rengimo metodą realizuojantis algoritmas

Jeigu dokumentacijos turinio dar nėra, panaudojant DITA technologiją reikia sukurti XML failus, kuriuose naudojant DITA gaires suženklintas dokumentacijos turinys. Sukūrus dokumentacijos turinį ir norimus formatų stilius, galima paleisti konvertavimo scenarijų, komandinėje eilutėje suvedus šią komandą: `ant -f publish.xml`. Jeigu peržiūrėjus konvertuotus dokumentus reikia atlikti pakeitimus, tuomet yra galimybė tai padaryti, pakeičiant turinį ir stilių. Jeigu daugiau pakeitimų nereikia ir konvertuota dokumentacija tenkina, tuomet darbas užbaigiamas.

Pagrindinis metodo principas yra paimti XML formatu išsaugotą dokumentaciją ir konvertuoti ją, panaudojant sukurtą scenarijų, į tris formatus. Tai užtikrina greitą ir patogų dokumentacijos pateikimą skirtingais formatais ir skirtingiems naudotojams pritaikytus dokumentacijos produktus.

## 2.5. Išplėstinis duomenų bazės lentelės dokumentacijos algoritmas

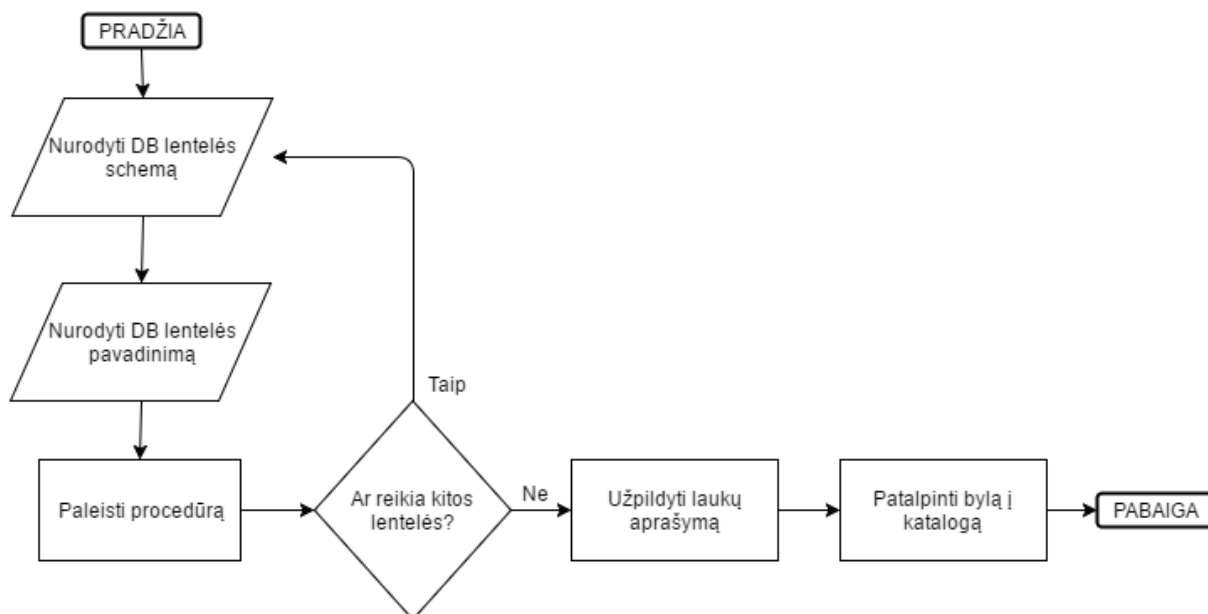
Kartais rengiant administratoriui skirtą dokumentaciją yra reikalinga aprašyti duomenų bazės lenteles. Dokumentuojant duomenų bazių lenteles yra aprašomos pateikiant lentelės aprašymą ir sudedant visus laukus į lentelę. Lentelėje yra nurodoma duomenų bazės lauko pavadinimas, tipas, sąryšis ir aprašymas. Pavyzdinėje duomenų bazės lentelės dokumentacijoje yra aprašoma lentelė „Studentai“ (žr. 5 lentelė).

5 lentelė. Duomenų bazės lentelės dokumentacijos pavyzdys

Lauko pavadinimas	Lauko tipas	Sąryšis	Aprašymas
StudentoId	Int	PK	Unikalus studento numeris, generuojamas automatiškai
Vardas	Varchar(256)		Pilnas studento vardas. Jei studentas turi daugiau nei vieną vardą, visi čia ir surašomi.
Pavarde	Varchar(256)		Pilna studento pavarde. Jei studentas turi daugiau nei vieną pavardę, visos čia ir surašomos.
VadovoId	Int	FK	Išorinis raktas rodantis į „Vadovai“ lentelę.
StudijuProgramosKodas	Int	FK	Išorinis raktas rodantis į „StudijuProgramos“ lentelę.
ArStudijuoja	Bit		1 – studijuoja, 0 – nestudijuoja

Išplėstinis metodo algoritmas (žr. 13 pav.) naudoja sukurtą duomenų bazės procedūrą, kuri paima per parametrus nurodytą lentelės pavadinimą ir randa jos laukus duomenų bazės sisteminėje lentelėje. Duomenų bazės lauko pavadinimas ir tipas yra automatiškai įrašomi į lentelę, kuri yra paruošta naudojant DITA lentelės gaires, o failas yra išsaugotas XML formatu.

Duomenų bazės procedūra gali sugeneruoti bet kurios duomenų bazės lentelės dokumentacijos failą XML formatu, o dokumentacijos rengėjui telieka užpildyti laukų aprašymus.



13 pav. Išplėstinis metodo algoritmas

Išplėstinio metodo privalumas yra tas, kad dokumentacijos rengėjui nereikia kiekvieną kartą kopijuoti ar perrašyti duomenų (duomenų bazės lentelės pavadinimas ir tipas) ir nereikia kurti lentelės, panaudojant DITA gaires XML formatu. Visa tai padaro suprogramuota duomenų bazės procedūra, kurią reikia tik paleisti ir gautą XML failą (žr. 14 pav.) patalpinti į norimą dokumentacijos katalogą (dažniausiai tą, kuris yra nurodytas konvertavimo scenarijaus parametruose), kur yra saugomi kiti dokumentacijos turinio failai.

XML faile aprašoma lentelė. Pagrindinė gairė <simpletable> reiškia, kad kitose viduje esančiose gairėse bus aprašomi lentelės laukai. Gairė <thead> aprašo pačią pirmąją eilutę, kurioje yra sukuriami šie stulpeliai: lauko pavadinimas, lauko tipas, sąryšis ir aprašymas. Gairės <strow> aprašo kitas eilutes, o <stentry> gairėse yra nurodoma ta informacija, kuri bus įrašyta konkrečioje lentelės eilutėje. Šis XML dokumentas sugeneruos lentelę iš 7 eilučių ir 4 stulpelių.



```

<simpletable>
  <thead>
    <stentry>Lauko pavadinimas</stentry>
    <stentry>Lauko tipas</stentry>
    <stentry>Sąryšis</stentry>
    <stentry>Aprašymas</stentry>
  </thead>
  <strow>
    <stentry>StudentoId</stentry>
    <stentry>Int</stentry>
    <stentry />
    <stentry />
  </strow>
  <strow>
    <stentry>Vardas</stentry>
    <stentry>Varchar(256)</stentry>
    <stentry />
    <stentry />
  </strow>
  <strow>
    <stentry>Pavarde</stentry>
    <stentry>Varchar(256)</stentry>
    <stentry />
    <stentry />
  </strow>
  <strow>
    <stentry>VadovoId</stentry>
    <stentry>Int</stentry>
    <stentry />
    <stentry />
  </strow>
  <strow>
    <stentry>StudijuProgramosKodas</stentry>
    <stentry>Int</stentry>
    <stentry />
    <stentry />
  </strow>
  <strow>
    <stentry>ArStudijuoja</stentry>
    <stentry>Bit</stentry>
    <stentry />
    <stentry />
  </strow>
</simpletable>

```

14 pav. Procedūros sugeneruoto XML dokumento pavyzdys

Panaudojant metodą ir sukurtą konvertavimo scenarijų, XML dokumentas galės būti konvertuojamas į įvairius formatus (įmonei „IT Sprendimai“ pritaikytus CHM, HTML ir PDF formatus) automatiškai.

## 2.6. Išvados

1. Patobulintame dokumentacijos rengimo metode yra įtraukiamas dokumentacijos naudotojo vaidmuo, kuris yra skirtas teikti pasiūlymus dėl dokumentacijos kokybės gerinimo.
2. Suprojektuota DITA sistemos integracija į egzistuojančią sistemą. Tai leis dokumentacijos turinio ir dizaino kūrėjams matyti vieni kitų pakeitimus bei užtikrins, kad visas su programine įranga susijęs kodas (tiek programinė įranga, tiek ir dokumentacija) yra saugomi toje pačioje vietoje. DITA sistema išspręs dokumentacijos turinio atskyrimo nuo dizaino problemą.
3. Sukurtas programinės įrangos dokumentacijos rengimo metodas, kuris remiasi sukurtu scenarijumi (angl. *script*), leidžiančiu konvertuoti pasirinktus dokumentacijos failus į skirtingus formatus vienos komandos paleidimu. Tai išspręs dokumentacijos turinio konvertavimo į skirtingus formatus apribojimą bei sudėtingo valdymo problemą.
4. Išplėstinis metodo algoritmas supaprastina dokumentacijos rengimą, kai yra dokumentuojami duomenų bazės lentelės laukai. Tai leis lengviau rengti ir valdyti dokumentacijos turinį.

### 3. DOKUMENTACIJOS PROCESO EKSPERIMENTINIS TYRIMAS

Šiame skyriuje aprašytas atliktas dokumentacijos proceso tyrimas. Pagrindinis šio tyrimo tikslas yra išsiaiškinti ar dokumentacijos rengimo procesas pagal sukurtą metodą yra greitesnis, patogesnis bei lengviau palaikomas. Atliekant tyrimą lygintos dvi dokumentacijos rengimo technologijos: vikio puslapiai ir DITA.

#### 3.1. Konvertavimas į skirtingus formatus

**Scenarijus 1:** Konvertuoti turinį į: PDF, HTML ir CHM

6 lentelė. Konvertavimo į skirtingus formatus palyginimas

Technologija	Žingsnių aprašymas	Žingsnių kiekis
Vikio puslapiai	<ol style="list-style-type: none"><li>1. Nusikopijuoti dokumentacijos tekstą iš Viki puslapio, atsidaryti „MS Word“ arba „LibreOffice“ teksto rengyklę ir įklijuoti tekstą.</li><li>2. Padaryti atitinkamą reikalingą formatavimą (pvz. sulygiuoti paraštes, centruoti paveikslėlius ir pan.), įrašyti .pdf formatu.</li><li>3. Įrašyti tą patį dokumentą pasirinkus „Web Page“ tipą.</li><li>4. Konvertavimui į CHM reikės papildomos programinės įrangos, pvz. „chmProcessor“.</li></ol>	4
DITA	<ol style="list-style-type: none"><li>1. Paleisti scenarijų, kuris konvertuos į visus formatus iš karto: PDF, HTML, CHM.</li></ol>	1

DITA technologijai reikia tik vieno žingsnio norint konvertuoti turinį į tris formatus. Naudojant Vikio puslapius reikalingi 4 žingsniai, todėl DITA technologija optimizuoja konvertavimo procesą ir tai galima atlikti daug greičiau ir patogiau.

#### 3.2. Pakeitimų valdymas

**Scenarijus 2:** Pritaikyti skirtingą stilių (skyrių/poskyrių išdėstymą, lentelių ir paveikslėlių lygiavimą, teksto šriftą, spalvą ir pan.) skirtingiems dokumentams PDF, HTML ir CHM formatams.

7 lentelė. Skirtingų stilių pritaikymas skirtingiems formatams

Technologija	Žingsnių aprašymas	Žingsnių kiekis
Vikio puslapiai	<ol style="list-style-type: none"> <li>1. Atsidaryti „MS Word“ arba „LibreOffice“ teksto rengyklę ir įklijuoti tekstą norimą redaguoti tekstą. Pritaikyti reikiamą dokumento stilių.</li> <li>2. Konvertuoti dokumentą į PDF.</li> <li>3. Atsidaryti jau konvertuotą HTML dokumentą, pritaikyti stilius.</li> <li>4. Atsidaryti jau konvertuotą CHM failą, padaryti stilių pakeitimus.</li> </ol>	4
DITA	<ol style="list-style-type: none"> <li>1. Pakeisti PDF publikavimo mechanizmą ir pritaikyti skirtingą stilių.</li> <li>2. Pakeisti HTML publikavimo mechanizmą ir pritaikyti skirtingą stilių.</li> <li>3. Pakeisti CHM publikavimo mechanizmą ir pritaikyti skirtingą stilių.</li> </ol>	3

Panaudojant DITA technologiją, pakeitimams valdyti reikia atlikti 3 žingsnius, o vikio puslapiams vienu žingsniu daugiau. Nors žingsnių skaičius yra panašus, tačiau DITA technologija leidžia apibrėžti stilių visiems formatams vieną kartą. DITA technologijos privalumas yra tas, kad apibrėžus formatų stilių, galima konvertuoti daug skirtingų failų ir visiems jiems bus pritaikytas nurodytas stilius. Naudojant Vikio puslapius kiekvienam dokumentui reikės kiekvieną kartą atskirai nurodyti stilius, kas ilgalaikėje perspektyvoje užims daugiau laiko ir pastangų.

### Scenarijus 3: Pakeisti sakinių/žodžių skirtinguose dokumentuose PDF, HTML ir CHM

8 lentelė. Teksto pakeitimai skirtinguose formatuose

Technologija	Žingsnių aprašymas	Žingsnių kiekis
Vikio puslapiai	<ol style="list-style-type: none"> <li>1. Nusikopijuoti tekstą, atsidaryti „MS Word“ arba „LibreOffice“ teksto rengyklę ir įklijuoti tekstą.</li> <li>2. Pritaikyti reikiamą stilių.</li> <li>3. Pakeisti sakinių/žodžių.</li> </ol>	8

	<p>4. Konvertuoti į PDF.</p> <p>5. Atsidaryti HTML dokumentą ir pakeisti sakinių/žodį, užsaugoti.</p> <p>6. Nusikopijuoti tekstą, atsidaryti „MS Word“ arba „LibreOffice teksto rengyklę“ ir įklijuoti tekstą.</p> <p>7. Pakeisti sakinių/žodį</p> <p>8. Konvertuoti į CHM, panaudojant išorinę programinę įrangą, pvz. „chmProcessor“.</p>	
DITA	<p>1. Atsidaryti atitinkamą XML failą ir pakeisti sakinių/žodį.</p> <p>2. Paleisti scenarijų, kuris konvertuos į visus formatus iš karto: PDF, HTML, CHM.</p>	2

Jeigu dokumentacija užbaigta ir pateikta trimis formatais, tačiau reikia joje pakeisti tam tikrą žodį ar sakinį, tuomet Vikio puslapių atveju reikia keisti kiekvieno formato failą. Kiekvienam formatui jau yra pritaikytas stilius ir jei konvertuotume viską iš naujo tekstą iš naujo pritaikyti dokumentų stilius, o tai užimtų dar daugiau laiko nei pakeisti tam tikrą žodį ar sakinį visuose dokumentuose. Vikio puslapių atveju yra reikalingi 8 žingsniai, o DITA technologija leidžia vienu žingsniu pakeisti XML dokumentą, pakeitimas bus automatiškai pritaikytas visiems formatams. Taip pat dar vieno žingsnio metu (paleidžiant sukurtą scenarijų) dokumentacija bus konvertuojama į visus formatus. Viso DITA technologija reikalauja 2 žingsnių, lyginant su Viki puslapiais tai yra 4 kartus mažiau.

#### **Scenarijus 4:** Sukeisti skyrius vietomis PDF, HTML ir CHM

*9 lentelė. Skyrių sukeitimas vietomis*

<b>Technologija</b>	<b>Žingsnių aprašymas</b>	<b>Žingsnių kiekis</b>
Vikio puslapiai	<p>1. Nusikopijuoti tekstą, atsidaryti „MS Word“ arba „LibreOffice“ teksto rengyklę ir įklijuoti tekstą.</p> <p>2. Pritaikyti reikiamą stilių.</p> <p>3. Sukeisti skyrius vietomis, nukopijuojant skyriaus turinį ir po to jį įklijuojant reikiamoje vietoje.</p> <p>4. Konvertuoti į PDF.</p> <p>5. Atsidaryti HTML dokumentą ir sukeisti skyrius vietomis, užsaugoti.</p>	8

	6. Nusikopijuoti tekstą, atsidaryti „MS Word“ arba „LibreOffice“ teksto rengyklę ir įklijuoti tekstą. 7. Sukeisti skyrius vietomis, nukopijuojant skyriaus turinį ir po to jį įklijuojant reikiamoje vietoje. 8. Konvertuoti į CHM, panaudojant išorinę programinę įrangą, pvz. „chmProcessor“.	
DITA	1. Atsidaryti atitinkamą XML failą ir sukeisti skyrius vietomis. 2. Paleisti scenarijų, kuris konvertuos į visus formatus iš karto: PDF, HTML, CHM.	2

Panašiai kaip ir žodžio ar sakinio pakeitimo scenarijaus atveju, kartais pabaigtoje rašyti dokumentacijoje prireikia sukeisti skyrius vietomis. Panaudojant Vikio puslapius yra reikalingi 8 žingsniai, kurių metu yra keičiami visi trijų formatų failai. Atliekant šį darbą yra reikalinga daug pastangų bei atidumo ir žinoma laiko. DITA technologija leidžia optimizuoti procesą ir sukeisti skyrius vietomis viename XML dokumente, o vėliau konvertuoti jį į skirtingus formatus vieno scenarijaus paleidimu.

### 3.3. Skirtingų dokumentų versijų kūrimas panaudojant tą patį turinį

**Scenarijus 5:** Sukurti skirtingų versijų (skirtingo turinio) PDF, HTML ir CHM dokumentus.

10 lentelė. Skirtingų versijų dokumentų kūrimas

Technologija	Žingsnių aprašymas	Žingsnių kiekis
Vikio puslapiai	1. Nusikopijuoti tekstą, atsidaryti „MS Word“ arba „LibreOffice“ teksto rengyklę ir įklijuoti tekstą. 2. Pritaikyti reikiamą stilių. 3. Atlikti turinio pakeitimus. 4. Konvertuoti į PDF, suteikti pavadinimą „pdf_v1“. 5. Įrašyti tą patį dokumentą pasirinkus „Web Page“ tipą, suteikti pavadinimą „html_v1“. 6. HTML dokumentui pritaikyti stilių. 7. Konvertuoti į CHM, panaudojant išorinę programinę įrangą, pvz. „chmProcessor“, suteikti pavadinimą „chm_v1“.	12

	8. Atlikti turinio pakeitimus „MS Word“ ar „LibreOffice“ teksto rengyklėje dar kartą. 9. Konvertuoti į PDF, suteikti pavadinimą „pdf_v2“. 10. Įrašyti tą patį dokumentą pasirinkus „Web Page“ tipą, suteikti pavadinimą „html_v2“. 11. HTML dokumentui pritaikyti stilių. 12. Konvertuoti į CHM, panaudojant išorinę programinę įrangą, pvz. „chmProcessor“, suteikti pavadinimą „chm_v1“.	
DITA	1. Nusikopijuoti reikiamus xml turinio failus ir naujuose failuose padaryti pakeitimus. 2. Sukurti naują DITA žemėlapi ir jame nurodyti kelias iki naujai sukurtų failų. 3. Paleisti scenarijų, kuris konvertuos abu DITA žemėlapius (abi turinio versijas) į visus formatus iš karto: PDF, HTML, CHM.	3

Kai yra poreikis kurti skirtingų versijų dokumentus, panaudojant Vikio puslapius tai užima ganėtinai daug laiko ir pastangų. Tą patį tekstą reikia kopijuoti ir įrašyti skirtingais formatais, tai yra atliekama per 12 žingsnių. Panaudojant DITA technologiją yra sukuriami nauji turinio failai bei suteikiamos jų nuorodos į taip pat naujai sukurtą žemėlapi. Vėliau yra paleidžiamas sukurtas scenarijus, kuris konvertuos visus DITA žemėlapiuose nurodytus failus į skirtingus formatus, tai atliekama per 3 žingsnius.

### 3.4. Failų kiekio ir dydžio palyginimas

**Scenarijus 6:** Palyginti eksportuotų failų kiekį ir užimamą vietą diske.

11 lentelė. Failų kiekis ir dydis diske

Technologija	Žingsnių aprašymas	Failų kiekis	Failų dydis
Vikio puslapiai	1. PDF – 1 dokumentas, 0.97 MB 2. HTML – 36 dokumentai, 1.25 MB 3. CHM – 1 dokumentas, 0.21 MB	38	2.43 MB
DITA	1. PDF – 1 dokumentas, 0.50 MB 2. HTML – 41 dokumentas, 0.51 MB 3. CHM – 44 dokumentai, 0.97 MB	86	1.98 MB

Iš lentelėje pateiktų duomenų matyti, kad nors DITA technologija sugeneruoja beveik dvigubai daugiau failų, tačiau jie užima mažiau vietos diske t.y. 450 KB mažiau. Nors dabartiniai diskai yra didelės talpos, tačiau vis tiek kuo mažiau vietos panaudosime – tuo daugiau laisvos vietos liks diske, kurią bus galima išnaudoti kitiems poreikiams.

### **3.5. Dokumento stiliaus palyginimas**

Toliau palyginami dokumentų stiliai, kurie buvo konvertuoti panaudojant „MS Word“ teksto rengyklę (arba „chmProcessor“ programą) ir DITA technologijas. Palyginimui atlikti panaudojama „Atutor“ sistemos dokumentacija. (žr. 1 priedą). Įmonėje kuriamos programinės įrangos dokumentacijai pateikti naudojami trys formatai: PDF, HTML ir CHM.

#### **3.5.1. PDF formato dokumentų palyginimas**

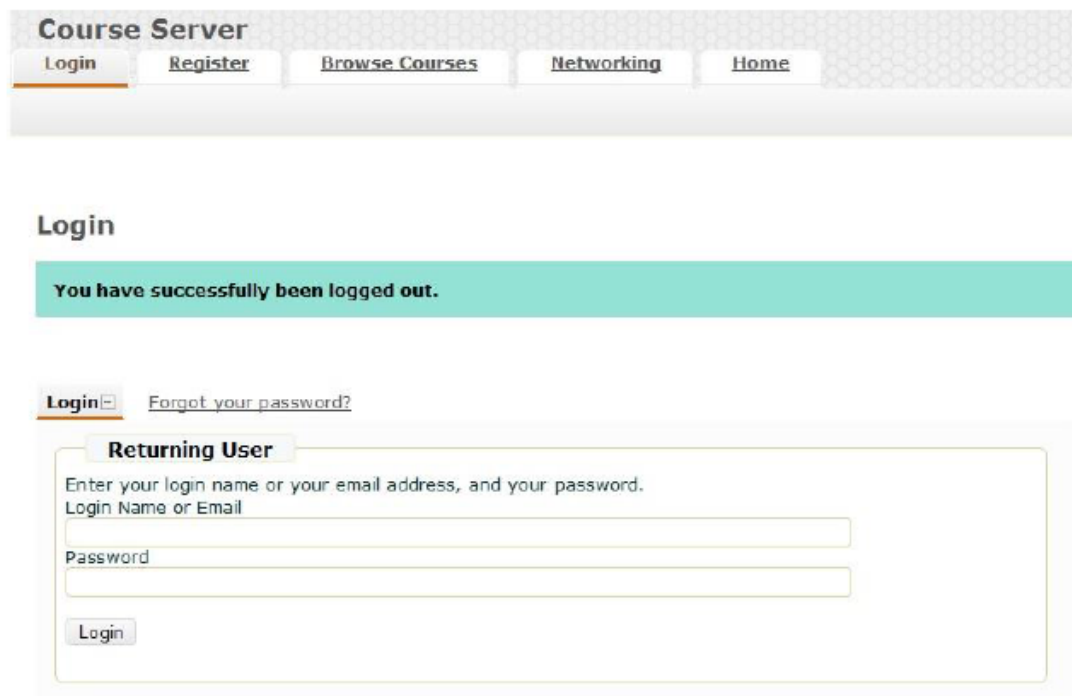
PDF formato dokumentas, kuris buvo konvertuotas panaudojant „MS Word“ teksto rengyklę (žr. 15 pav.) turi identišką stilių „MS Word“ dokumentui. Paveikslėlių ir žingsnių numeracija liko nepakitus, viso puslapio stilius gali būti lengvai keičiamas MS Word programoje.



## 1. Prisijungimo prie sistemos instrukcija

Kiekvienas ATutor sistemos naudotas privalo prisijungti prie sistemos. Ši instrukcija yra rengiama dėstytojui, todėl tolimesniuose žingsniuose matysime dėstytojo sistemos galimybes. Prisijungimo prie ATutor sistemos žingsniai:

1. Užėjus į ATutor sistemos pradinį langą yra rodomi prisijungimo prie sistemos laukeliai (žr. 1 pav.). Įveskite naudotojo prisijungimo vardą ir slaptažodį bei paspauskite *Login* mygtuką.



1 pav. ATutor prisijungimo langas

2. Po sėkmingo prisijungimo, sistema nukreips į pagrindinį puslapį (žr. 2 pav.).



15 pav. Word teksto rengykle konvertuotas PDF dokumentas

DITA technologijos sugeneruotas PDF dokumentas (žr. 16 pav.) turi savitą stilių, kuris buvo nurodytas XSL dokumente. Puslapio antraštė, skyrių bei paveikslėlio pavadinimai, tekstas yra stilizuojamas pagal iš anksto apibrėžtas taisykles. Dokumentacijos žingsniai yra automatiškai numeruojami. Abu dokumentai atrodo gražiai stiliaus prasme, nėra elementų, kurie būtų netinkamoje vietoje, tvarkingas teksto lygiavimas, paveikslėlių lygiavimas centruotas.

## Pilna ATutor sistemos dokumentacija

---

Čia pateikiama pilna ATutor sistemos naudotojo dokumentacija

### Prisijungimo prie sistemos instrukcija

---

Kiekvienas ATutor sistemos naudotas privalo prisijungti prie sistemos.

Prisijungimo prie ATutor sistemos žingsniai:

1. Užėjus į ATutor sistemos pradinį langą yra rodomi prisijungimo prie sistemos laukeliai (žr. ATutor prisijungimo langas). Įveskite naudotojo prisijungimo vardą ir slaptažodį bei paspauskite Login mygtuką.

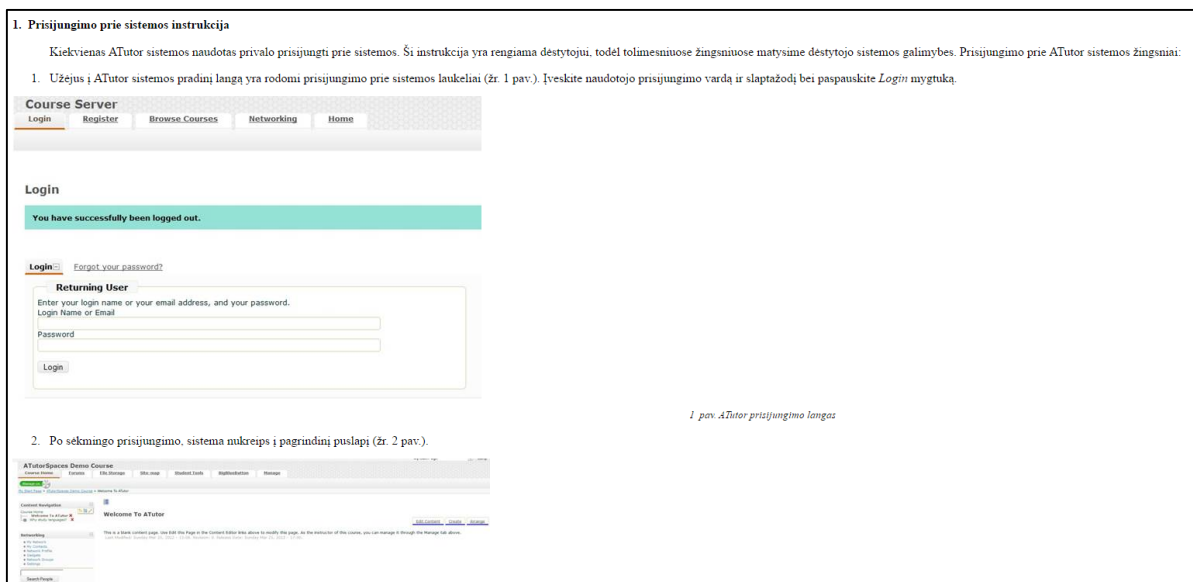
Piešinys 1: ATutor prisijungimo langas

2. Po sėkmingo prisijungimo, sistema nukreips į pagrindinį puslapį (žr. ATutor pagrindinis dėstytojo langas).

16 pav. PDF dokumentas sukurtas naudojant DITA technologiją

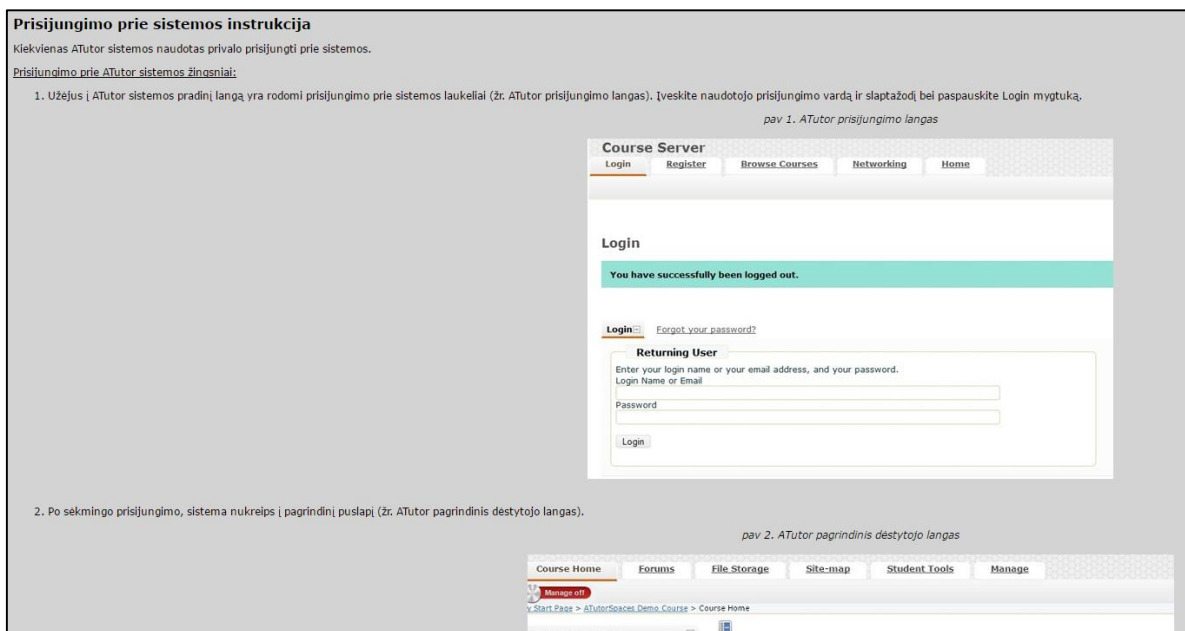
### 3.5.2. HTML formato dokumentų palyginimas

HTML formato dokumentas, konvertuotas panaudojant „MS Word“ teksto rengyklę (žr. 17 pav.) yra standartinio balto fono, paveikslėliai yra lygiuojami kairėje lapo pusėje, o jų pavadinimas yra centruotas. Nors visas tekstas ir antraštės yra išlikusios, sugeneruotame dokumente nėra temų turinio, o tai gali būti nepatogu, kai dokumentacijos dokumentas turi daug skyrių.



17 pav. Word teksto rengykle konvertuotas HTML dokumentas

DITA technologijos sugeneruotas dokumentas (žr. 18 pav.) turi išryškintus skyrių pavadinimus, centruotus paveikslėlius bei jų pavadinimus. Taip pat yra pritaikyti kiti stiliai, pavyzdžiui fono spalva ar teksto pabraukimas esminėse vietose. DITA technologijos sugeneruotas dokumentas atrodo tvarkingiau bei turi išskirtinį stilių.

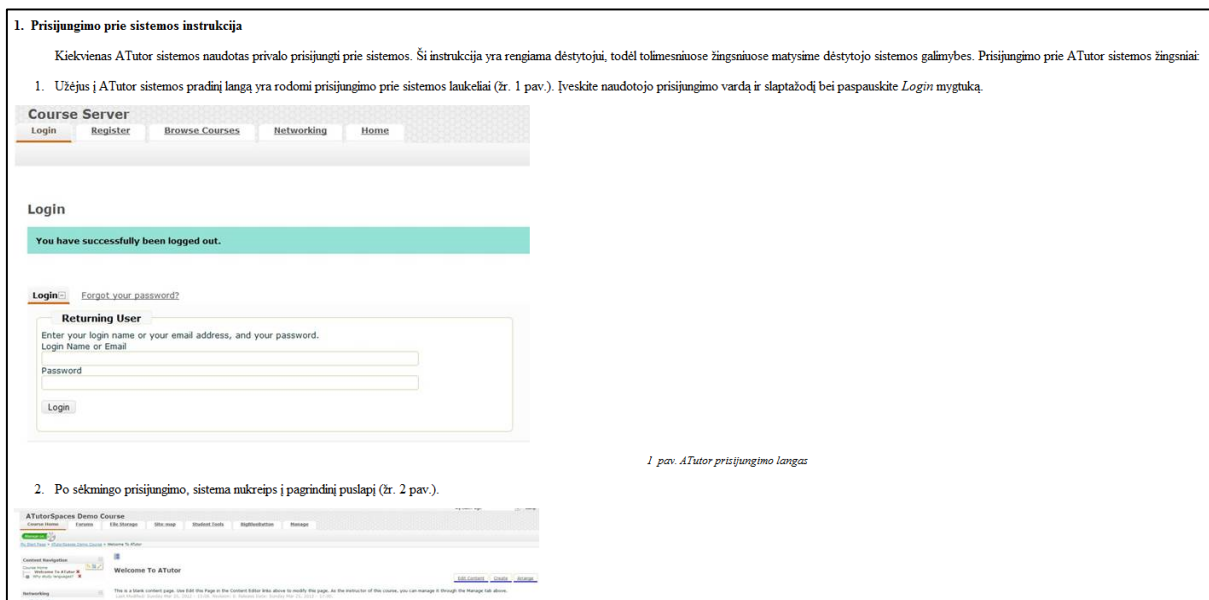


18 pav. HTML dokumentas sugeneruotas panaudojant DITA technologiją

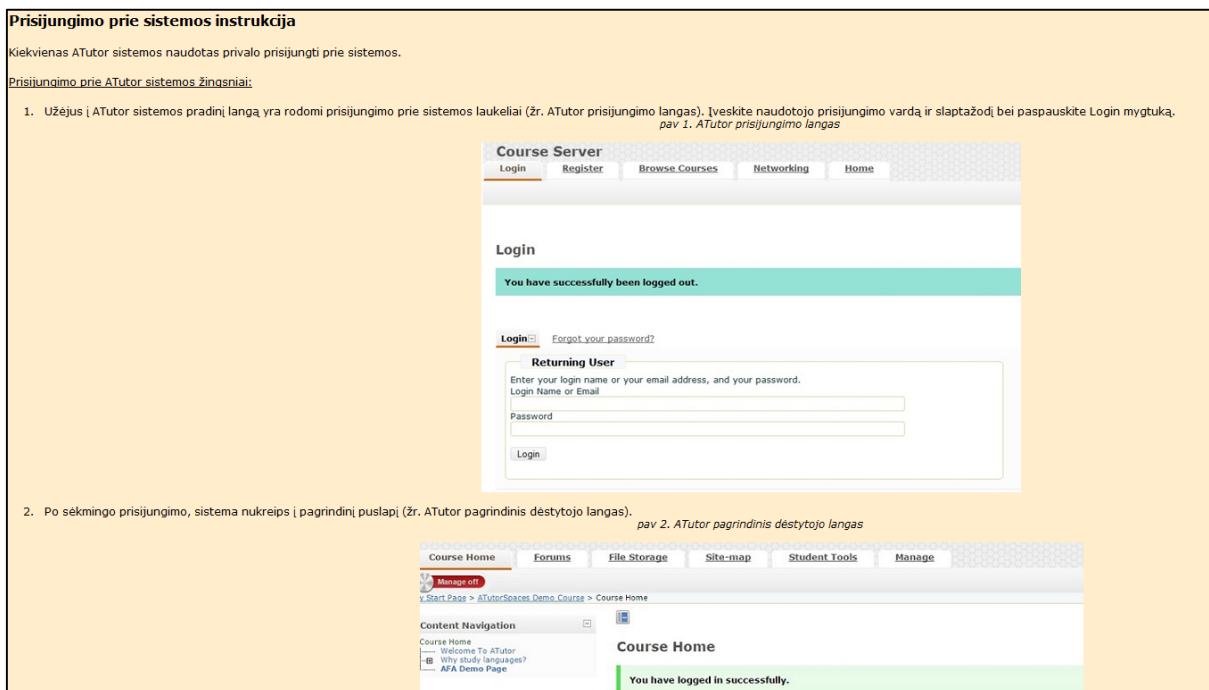
### 3.5.3. CHM formato dokumentų palyginimas

Lyginant du CHM dokumentus galima nesunkiai pastebėti skirtumus. Dokumentas, kuris buvo sugeneruotas panaudojant programinę įrangą „chmProcessor“ (žr. 19 pav.), atrodo panašiai kaip ir HTML dokumentas, paveikslėliai lygiuojami kairėje, kai jų pavadinimai yra

centruojami. DITA technologijos sugeneruotas dokumentas (žr. 20 pav.) turi savitą stilių, paveikslėliai atrodo tvarkingai bei yra išryškintos pagrindinės teksto vietos.



19 pav. *chmProcessor* programa konvertuotas CHM dokumentas



20 pav. CHM dokumentas konvertuotas panaudojant DITA technologiją

Dokumentai, kurie buvo generuojami panaudojant tiek „MS Word“, arba „chmProcessor“ programinę įrangą, tiek ir DITA technologiją, atrodo neblogai. Tačiau DITA technologijos dokumentai yra labiau pritaikyti naudotojo poreikiams bei atrodo tvarkingiau.

### **3.6. Išvados**

1. Atlikus dokumentacijos rengimo eksperimentinį tyrimą, paaiškėjo, kad naudojant DITA technologiją reikia atlikti mažiau žingsnių, norint pasiekti tą patį ar net geresnį rezultatą. Mažesnis žingsnių kiekis reikalauja mažiau laiko bei pastangų atlikti rezultatą, tai reiškia, kad šiuo atveju DITA yra pranašesnė technologija.
2. Failų skaičius visais atvejais, išskyrus CHM, buvo gautas panašus. DITA technologija sugeneravo daug daugiau failų, kurie užėmė daugiau vietos diske. Tačiau jei palygintume visus konvertuotus formatus (PDF, HTML ir CHM) ir jų užimamą vietą diske, DITA technologijos failai užima mažiau vietos diske, konkrečiu atveju tai 450KB mažiau. Šis talpos sumažėjimas gali būti aktualus parsisiunčiant ir peržiūrint dokumentaciją mobiliuosiuose įrenginiuose.
3. DITA technologijos stiliai buvo koreguoti pagal individualius naudotojo poreikius, todėl šios technologijos sugeneruoti failai atrodo tvarkingiau. Kiekvieno konvertavimo metu yra naudojami tie patys stiliai, todėl visi tam tikro formato dokumentai turi vienodą dizainą.

## IŠVADOS

1. Dokumentacijos rengimo ir teikimo procesą reikia tobulinti, nes esamas procesas yra neefektyvus: dokumentacijos turinys nėra atskirtas nuo atvaizdavimo, todėl yra sudėtinga pakeisti dokumentacijos dokumento stilių, sudėtinga konvertuoti turinį į kitus formatus bei valdyti turinio pakeitimus.
2. Palyginus galimas technologijas, pasirinkta DITA, nes ji turi žemėlapių funkciją, kuri leidžia lengvai valdyti dokumento turinį ir skirtingas dokumento versijas. Pasirinktas publikavimo mechanizmas DITA-OT, kurį panaudojant bus galima efektyviai konvertuoti failus į skirtingus formatus.
3. Sukurtas programinės įrangos dokumentacijos rengimo metodas, panaudojant DITA technologiją, kuriame dokumentai yra saugomi XML formatu bei vėliau gali būti konvertuojami į skirtingus formatus. Metodas palaiko automatinį turinio konvertavimą į reikiamus formatus vienos komandos paleidimu. Sukurta procedūra, kuri leidžia sutaupyti laiko rengiant turinį, kuris turi būti dokumentuotas lentelėje. Visos priemonės, reikalingos metodui įgyvendinti, turi atvirą licenciją.
4. Atlikus tyrimą paaiškėjo, kad sukurtas metodas reikalauja mažiau žingsnių norint pasiekti tą patį ar geresnį rezultatą. Mažesnis žingsnių kiekis reiškia greičiau ir kokybiškiau atliktą dokumentaciją. Nustatyta, kad DITA technologija yra pranašesnė (lengviau valdomi turinio pakeitimai, galimybė atskirti turinį nuo atvaizdavimo, konvertuoti į įvairius formatus) už dokumentacijos rengimą panaudojant vikio puslapius, todėl įmonėje „IT Sprendimai“ įdiegta DITA technologija ir pradėtas taikyti sukurtas dokumentacijos rengimo metodas.

## LITERATŪRA

1. Abel S., Choosing an XML Schema: DocBook or DITA?, 2008 [žiūrėta 2015-11-03].  
Prieiga per internetą:  
[http://thecontentwrangler.com/2008/04/11/choosing\\_an\\_xml\\_schema\\_docbook\\_or\\_dita/](http://thecontentwrangler.com/2008/04/11/choosing_an_xml_schema_docbook_or_dita/)
2. Adobe Help [interaktyvus] [žiūrėta 2015-10-30]. Prieiga per internetą:  
[http://help.adobe.com/en\\_US/FrameMaker/8.0/help.html?content=Chap22-Structured-Authoring-using-DITA\\_06.html](http://help.adobe.com/en_US/FrameMaker/8.0/help.html?content=Chap22-Structured-Authoring-using-DITA_06.html)
3. Atlassian Documentation [interaktyvus] [žiūrėta 2015-10-05]. Prieiga per internetą:  
<https://confluence.atlassian.com/doc/subscribe-to-rss-feeds-within-confluence-139453.html>
4. Companies Using DITA, ditawriter [interaktyvus] [žiūrėta 2016-01-04]. Prieiga per internetą: <http://www.ditawriter.com/companies-using-dita/>
5. Dick K. XML A Managers Guide Addison-Wesley, 2000 m.
6. DITA [interaktyvus] [žiūrėta 2015-10-30]. Prieiga per internetą:  
<http://dita.xml.org/book/export/html/1047>
7. DITA dominates structure adoption.Docbook. [interaktyvus] [žiūrėta 2016-01-04]. Prieiga per internetą: <http://www.slideshare.net/Scriptorium/the-state-of-structure/9-DITA-dominates-structure-adoption-DocBook>
8. DITA Open Toolkit [interaktyvus] [žiūrėta 2015-11-07]. Prieiga per internetą:  
<http://www.dita-ot.org/>
9. DITA2Go [interaktyvus] [žiūrėta 2015-11-07]. Prieiga per internetą: <http://dita2go.com/>
10. E. Rusty Harold, W. Scott Means, XML in a Nutshell 3rd Edition, 2004 m.
11. How to decide if DITA is right for you – and where to learn more, easyDITA [interaktyvus] [žiūrėta 2016-01-04]. Prieiga per internetą: <http://easydita.com/how-to-decide-if-dita-is-right-for-you-and-where-to-learn-more/>  
<https://ifs.host.cs.st-andrews.ac.uk/Books/SE9/Web/ExtraChaps/Documentation.pdf>
12. IBM developerWorks [interaktyvus] [žiūrėta 2015-10-30]. Prieiga per internetą:  
<http://www.ibm.com/developerworks/library/x-dita1/>
13. IEEE Standard for Software User Documentation, 2001

14. Internet Safety primary education [interaktyvus] [žiūrėta 2015-10-05] Prieiga per internetą: <http://internet-safety-primary-education.wikispaces.com/What-is-a-Wiki>
15. Mulvihill T., DocBook versus DITA, 2007 m. [žiūrėta 2015-12-11]. Prieiga per internetą: <http://www.slideshare.net/day/doc-book-vs-dita-teresa>
16. OASIS [interaktyvus] [žiūrėta 2015-10-30]. Prieiga per internetą: <https://www.oasis-open.org/org>
17. SGML international standard - iso8879, 1986 m.
18. Shafie H. DITA for the Impatient, 2015 [žiūrėta 2015-11-07]. Prieiga per internetą: <http://www.xmlmind.com/tutorials/DITA/>
19. Sommerville, I. Software Engineering, 9th Edition, 2010 [žiūrėta 2016-03-20]. Prieiga per internetą:
20. Sourceforge [interaktyvus] [žiūrėta 2015-11-07] ] Prieiga per internetą: <http://sourceforge.net/projects/dita-ot/files/DITA-OT%20Stable%20Release/dita-ot%201.0.0/>
21. Verbert K., Klerkx J., Meire M., Najjar J., Leuven E. Towards a Global Component Architecture for Learning Objects: An Ontology Based Approach 717 p. [žiūrėta 2015-11-05]. Prieiga per internetą: <https://lirias.kuleuven.be/bitstream/123456789/125459/1/>
22. W3C, Extensible Markup Language (XML) [interaktyvus] [žiūrėta 2015-11-07]. Prieiga per internetą: <https://www.w3.org/XML/>
23. WritePoint, DITAChallenges Survey, 2009 [žiūrėta 2015-11-07]. Prieiga per internetą: [http://www.writepoint.com/site/downloads/WritePoint\\_DITASurveyChallenges\\_Results.pdf](http://www.writepoint.com/site/downloads/WritePoint_DITASurveyChallenges_Results.pdf)



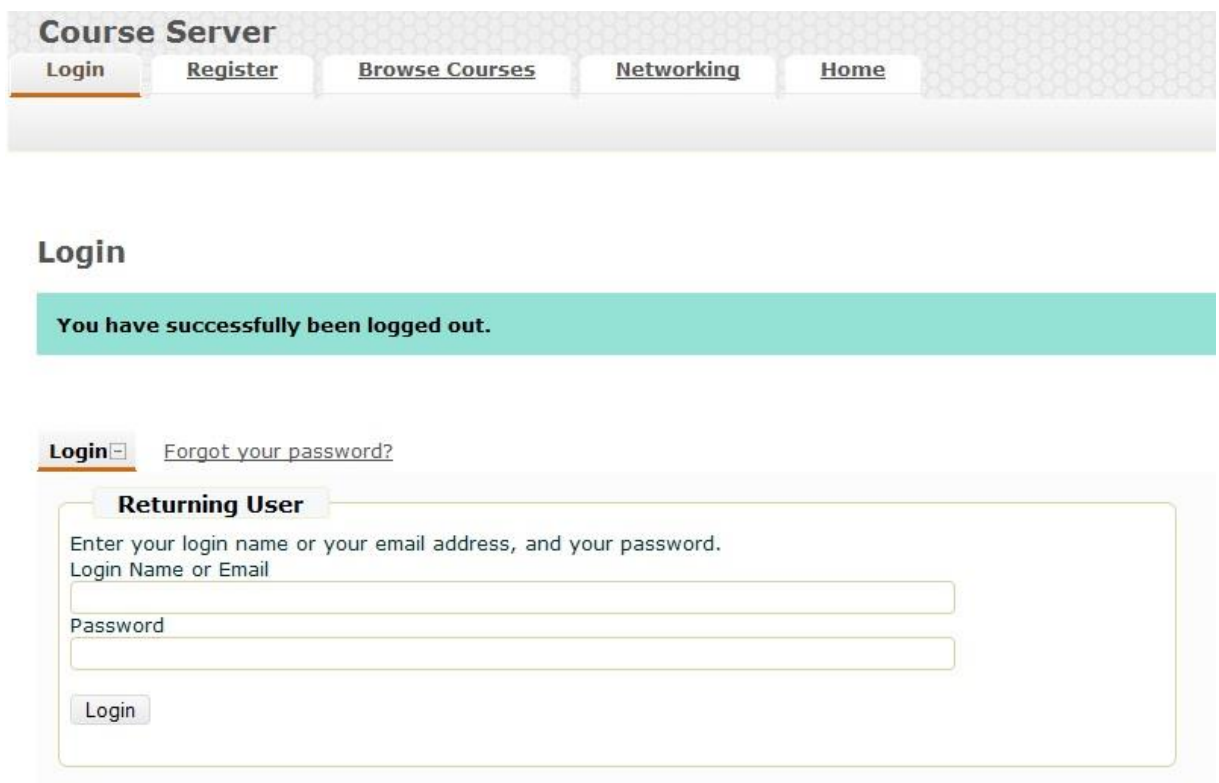
## PRIEDAI

### 1 priedas: Tyrime naudotas dokumentacijos pavyzdys

#### 1. Prisijungimo prie sistemos instrukcija

Kiekvienas ATutor sistemos naudotas privalo prisijungti prie sistemos. Ši instrukcija yra rengiama dėstytojui, todėl tolimesniuose žingsniuose matysime dėstytojo sistemos galimybes. Prisijungimo prie ATutor sistemos žingsniai:

1. Užėjus į ATutor sistemos pradinį langą yra rodomi prisijungimo prie sistemos laukeliai (žr. 1 pav.). Įveskite naudotojo prisijungimo vardą ir slaptažodį bei paspauskite *Login* mygtuką.



**Course Server**

[Login](#) [Register](#) [Browse Courses](#) [Networking](#) [Home](#)

### Login

You have successfully been logged out.

[Login](#) [Forgot your password?](#)

**Returning User**

Enter your login name or your email address, and your password.

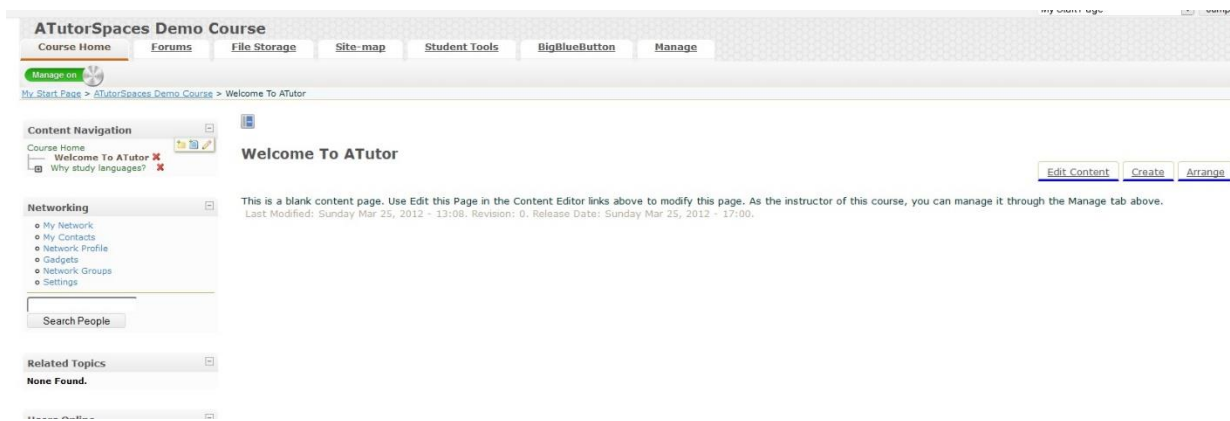
Login Name or Email

Password

Login

1 pav. ATutor prisijungimo langas

2. Po sėkmingo prisijungimo, sistema nukreips į pagrindinį puslapį (žr. 2 pav.).



2 pav. ATutor pagrindinis dėstytojo langas

Prisijungimas prie sistemos yra pabaigtas.

## 2. Kurso kūrimo instrukcija

Kurso kūrimo funkcija yra viena iš pagrindinių funkcijų, šią funkciją galima įvykdyti prisijungus prie ATutor sistemos ir atliekant šiuos žingsnius:

1. Viršutinėje sistemos ATutor užduočių juostoje paspausti ant *My Courses* nuorodos (žr. 3 pav.)



3 pav. ATutor užduočių juosta

2. Atsidariusiame lange sistema automatiškai pasirenka *My Courses* atvaizdavimą. Šiame lange bus atvaizduoti visi jau anksčiau šio dėstytojo sukurti kursai. Jei tokių nėra, bus matoma tuščia erdvė (žr. 4 pav.). Tam, kad sukurtume naują kursą, reikia spausti *Create Course* (žr. 4 pav.).

manoDemo

## My Start Page

My Courses Browse Courses Profile Preferences Networking Calendar Payments

### My Courses

? You can [Create a Course](#) by opening the My Courses tab above then opening the Create Course in the sub-menu that appears.

- Your personal preferences have not been setup yet. You can click on the icon next to you name above to open the personal preferences wizard, or ignore this message and click on the [Preferences](#) tab on My Start Page.
- No courses found. [Browse](#) existing courses or [create](#) a new one.

My Courses:  Create Course

Course	Instructor	Status	Shortcuts	Current Activity
				None Found.

4 pav. Esamų kursų peržiūra bei naujo kurso kūrimo nuoroda

3. Atsidariusiame naujame lange reikia nurodyti šiuos naujo kurso parametrus (žr. 5 pav. ir 6 pav.):

- Įvesti kurso pavadinimą (*Title*)
- Pasirinkti kurso kalbą (*Original Language*)
- Pasirinkti kurso kategoriją (*Category*). Jei kursų kategorijos nėra iš anksto sukurtos, galima palikti šį lauką tuščią. Esant poreikiui vėliau galima bus sukurti naują kategoriją ir priskirti kursą į šią kategoriją.
- Pasirinkti turinio eksportavimo lygį (*Export Content*). Galima pasirinkti vieną iš trijų variantų:
  - Nesuteikti galimybės eksportuoti puslapių (*Not available on any of the pages*). Šis pasirinkimas neleis kitiems naudotojams eksportuoti kurso turinio.
  - Eksportuoti kurso pagrindinius puslapius (*Available only for top level pages*). Šis pasirinkimas reiškia, kad kurso naudotojai galės eksportuoti visą kursą arba tam tikrą kurso skyrių kaip *IMS Content package* arba *IMS Common Cartridge* paketus. Šiuos paketus naudotojas galės atsidaryti panaudojant *IMS ar SCORM 1.2 viewer* savo lokaliame kompiuteryje.
  - Eksportuoti visus kurso puslapius (*Available on every page*). Šis pasirinkimas reiškia, kad kurso naudotojai galės eksportuoti bet kurį kurso puslapį, nepriklausomai ar tai visas kursas, kurso skyrius ar kurso medžiagos puslapis. Eksportuota medžiaga bus atidaryta panaudojant *IMS ar SCORM 1.2 viewer*.

- Rinkti kurso atnaujinimus (*Syndicate Announcements*). Galima pasirinkti du variantus:
  - Įjungti *RSS* kurso atnaujinimus (*Enable Announcements syndication via RSS*). Šis pasirinkimas reiškia, kad kituose puslapiuose (ar svetainėse) bus galimybė atvaizduoti šio kurso atnaujinimus su nuoroda į patį kursą.
  - Išjungti *RSS* atnaujinimus (*Disable Announcements syndication*)
- Priėjimo prie kurso nustatymai (*Access*). Galimi trys variantai:
  - Viešas (*Public*). Pasirinkus šį variantą naudotojas galės matyti kursą net ir neprisijungęs prie ATutor sistemos.
  - Apsaugotas (*Protected*). Pasirinkus šį variantą naudotojai galės prisijungti prie kurso tik tuo atveju, jei yra prisijungę prie ATutor sistemos.
  - Privatus (*Private*). Šiuo atveju naudotojai turi būti prisijungę prie ATutor sistemos, bet tam, kad galėtų peržvelgti kurso medžiagą, jie turės sulaukti patvirtinimo iš dėstytojo. Pasirinkus šį variantą, yra galimi dar du pasirinkimai:
    - Atsiųsti elektroninį laišką, kai atsiranda naudotojas, norintis patekti į kursą (*Email me when new enrollments require approval*)
    - Paslėpti šį kursą iš kursų pasirinkimo sąrašo (*Hide this course from the Browse Courses list*)
- Kurso pradžios data (*Release Date*). Kurso pradžios datos nustatymai gali būti du:
  - Kurso pradžios data yra dabartinė (*Release immediately*). Šis pasirinkimas reiškia, kad kursas bus pasiekiamas naudotojams tuoj pat po jo sukūrimo.
  - Nustatyti kurso pradžios datą (*Release on*). Galima nustatyti tikslią kurso pradžios datą, pvz. 2015 m. sausio 1 d. 00:00 val.
- Kurso pabaigos data (*End Date*). Kurso pabaigos nustatymai yra tokie:
  - Kurso pabaigos data nėra nustatyta (*No end date*)
  - Nustatyti kurso pabaigos datą (*End on*). Galima nustatyti tikslią kurso pabaigos datą, pvz. 2016 m. sausio 1 d. 00:00 val.
- Paleisti vizualinį teksto redagavimo režimą (*Switch to visual editor*). Paspaudus šį mygtuką bus atidarytas vizualinis teksto redagavimo režimas, kuriame galima bus

sukurti kurso reklamą (*Banner*). Jei šis mygtukas nėra paspaustas, tuomet kurso reklamos skiltyje galima įvesti tik nesuformatuotą tekstą.

- Pirminis kurso turinys (*Initial Content*). Galimi du variantai:
  - Tuščias (*empty*). Šis pasirinkimas reiškia, kad sukurtas kursas bus tuščias.
  - Sukurti skelbimą, turinį ir forumą (*Create basic announcement, content and forum*). Šis pasirinkimas reiškia, kad sukūrus kursą, skelbimas, kurso turinys bei forumas bus taip pat sukurti.
- Kurso licencijos aprašas (*Optional Copyright Notice*). Teksto įvedimo lauke, reikia nurodyti licencijos aprašymą, šis laukas nėra privalomas, tačiau jeigu yra tam tikros kurso medžiagos kopijavimo sąlygos, jas reikia nurodyti.
- Kurso paveikslėlis (*Icon*). Galima įkelti kurso paveikslėlį panaudojant *Browse komandą* arba palikti kurso paveikslėlį tuščią.

The screenshot shows the 'Create Course' form in the ATutor system. The 'Properties' section is active, displaying various configuration options for a new course. The form includes a 'Title' field, an 'Original Language' dropdown menu set to 'English', and a large 'Description' text area. Below these are a 'Category' dropdown, 'Export Content' options (with 'Available on every page' selected), 'Syndicate Announcements' options (with 'Disable Announcements syndication' selected), and 'Access' options (with 'Private' selected). There are also checkboxes for 'Email me when new enrollments require approval' and 'Hide this course from the Browse Courses list'. The 'Release Date' section has 'Release immediately' selected, and the 'End Date' section has 'No end date' selected.

5 pav. Naujo kurso kūrimo langas (1 dalis)

Switch to visual editor

Banner

Initial Content  
empty  
Create basic announcement, content, and forum.

Optional Copyright Notice

Icon  
No Icon Or

Upload Custom Course Icon  
Browse... No file selected. (gif, jpg, png)

Save Cancel

6 pav. Naujo kurso kūrimo langas (2 dalis)

- Kai visi kurso duomenys yra įvesti, reikia paspausti *Save* mygtuką tam, kad kursas būtų sukurtas.
- Po kurso išsaugojimo, sistema nukreipia į kurso puslapį ir parodo pranešimą apie sėkmingą kurso sukūrimą (žr. 7 pav.)

The screenshot shows the ATutor Course Home interface. On the left is a sidebar with navigation menus: Networking (My Network, My Contacts, Network Profile, Gadgets, Network Groups, Settings), Content Navigation (Course Home), Related Topics (None Found), Users Online (Lina K, Guests are not listed), and Glossary (N/A). The main content area is titled 'Course Home' and features a green banner with the message 'Action completed successfully.' Below this is a help message: 'Modify the initial Course Properties that were set when the course was created. Open the Manage Tab above, then open the Properties.' There are four icons for 'File Storage', 'My Tests and Surveys', 'My Tracker', and 'Directory'. At the bottom, there is an 'Announcements' section with the text 'Welcome To ATutor' and a timestamp 'Monday, April 13, 2015 - 05:30 by Lina K'.

7 pav. Pranešimas apie sėkmingą kurso sukūrimą

Kurso kūrimas yra pabaigtas.

### 3. Kurso medžiagos valdymo instrukcija

Kurso medžiaga gali būti kuriama bei valdoma tuomet, kai yra sukurtas kursas. Ankstesniame poskyryje pateikėme instrukciją kaip susikurti naują kursą, todėl šiame poskyryje pateiksime žingsnius, reikalingus kurti bei valdyti (t.y. rūšiuoti kurso puslapius) elektroninę mokymosi medžiagą.

## Kurso medžiagos kūrimo žingsniai, panaudojant HTML:

1. Po sėkmingo kurso sukūrimo (7 pav.) kurso valdymo režimas pagal nutylėjimą yra išjungtas. Kurso turinys negali būti kuriamas ir redaguojamas, kol nėra įjungtas kurso valdymo režimas. Jis įjungiamas paspaudus *Manage* mygtuką (žr. 8 pav.).



8 pav. Kurso valdymo režimo įjungimo mygtukas

2. Norint pridėti kurso puslapį, reikia paspausti mėlyną naujo failo mygtuką (žr. 9 pav.). Jis atsiranda kurso turinyje kairėje puslapio pusėje po to, kai yra įjungiamas kurso valdymo režimas.



9 pav. Naujo failo pridėjimas į kurso turinį

3. Atsidariusiame lange (žr. 10 pav.) įvedame puslapio pavadinimą, pasirenkame *HTML* bei *HTML* kalba aprašome elektroninio mokymosi medžiagą. Šiuo atveju aprašysime

kurso tikslus. Spaudžiame mygtuką *Save* ir užsaugome kurso tikslus. Sistema pateiks pranešimą apie sėkmingą medžiagos sukūrimą. Jeigu sistemos naudotojas neturi HTML žinių, tuomet jis gali pasirinkti *Plain Text* ir rašyti tekstą panaudojant teksto redaktorių.

**Networking**

- My Network
- My Contacts
- Network Profile
- Gadgets
- Network Groups
- Settings

Search People

**Content Navigation**

Course Home

Kurso tikslai ✖

**Related Topics**

None Found.

**Users Online**

Lina K

Guests are not listed

**Glossary**

There are no glossary terms being used in this content page.

**Search**

**Edit Content**

Action completed successfully.

Modify the initial [Course Properties](#) that were set when the course was created.

Content Create Arrange Import Export AContent Usage

**Content** Properties Glossary Terms Adapted Content Tests and Surveys Account

Save Close  Close after saving

\* Title Kurso tikslai

\* Formatting  Plain Text  HTML  HTML - Visual Editor  Web Link

Preview Accessibility Scripts/CSS Paste Files Forums External Tool

**Body**

```
<h1>Kurso tikslai</h1>

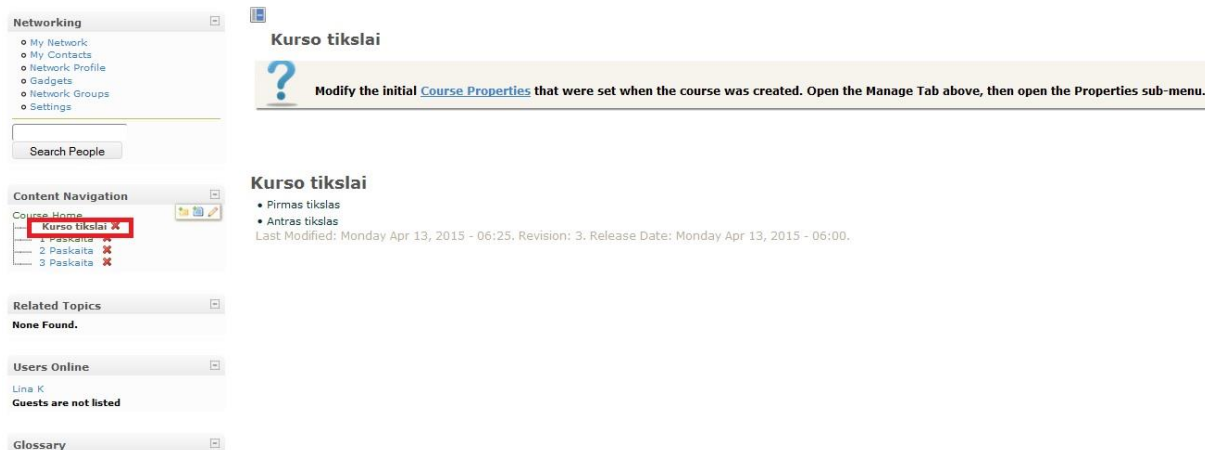
<ul>
<li> Pirmas tikslas </li>
<li> Antras tikslas </li>
</ul>
```

10 pav. Naujo failo kūrimas

### Kurso medžiagos valdymo žingsniai:

1. Kuomet kursas turi daug puslapių, jų atvaizdavimo tvarką galima redaguoti paspaudus ant bet kurio kurso puslapio nuorodos, esančios kurso turinyje, puslapio kairėje (žr. 11 pav.).





11 pav. Kurso turinio elementas

- Atsidariusiame lange yra rodomos kurso parinktys (puslapio dešiniame viršutiniame kampe) (žr. 12 pav.). Tam, kad galėtume keisti puslapių tvarką, reikia paspausti *Arrange*.



12 pav. Kurso puslapių eiliškumo keitimo nuoroda

- Kitame lange matome kurso puslapių išdėstymą. Jeigu norime, kad pvz. paskutinis puslapis būtų rodomas prieš pirmąjį, tuomet pažymime paskutinį puslapį ir spaudžiame mygtuką, kuris yra prie pirmojo puslapio (žr. 13 pav.)



13 pav. Kurso puslapių eiliškumo keitimas

Kurso medžiaga yra sukurta ir sutvarkyta.

#### 4. Naujos užduoties pridėjimo instrukcija

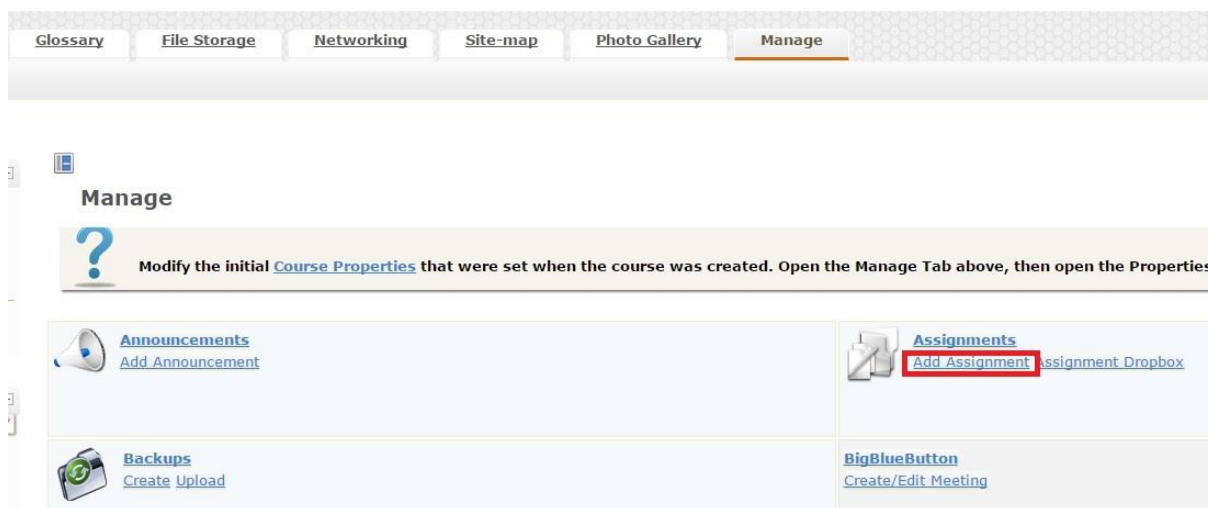
Naujos užduoties pridėjimo funkcija yra įvykdoma pagal šiuos žingsnius:

1. Viršutinėje užduočių juostoje (ji matoma esant bet kuriame sistemos ATutor puslapyje) paspausti *Manage* mygtuką (žr. 14 pav.). Šio mygtuko paspaudimu bus parodomi visi galimi kurso įrankiai.



14 pav. Kurso užduočių juosta

2. Atsidariusiame lange paspausti *Add assignment* nuorodą (žr. 15 pav.).



15 pav. Naujos užduoties pridėjimo nuoroda

3. Atidaromas naujas puslapis, kuriame galime nurodyti šiuos parametrus (žr. 16 pav.):
  - Užduoties pavadinimą (*Title*)
  - Užduoties priskyrimą grupei (*Assign To*). Galima pasirinkti priskirti šią užduotį visiems kurso studentams arba tik tam tikrai studentų grupei.
  - Užduoties pristatymo laiką (*Due Date*). Po šiuo parametru yra galimi du pasirinkimai:
    - Nėra užduoties pristatymo datos (*None*). Pasirinkus šį variantą, studentai galės įkelti užduotį visą laiką.
    - Iki tam tikros datos (*Date*). Pasirinkus šį variantą, galima įvesti tikslų užduoties pristatymo terminą, pvz. 2016 m. sausio 1 d. 00:00 val.

- Priimti vėluojančias pristatyti užduotis (*Accept Late Submissions*). Pasirinkus šį parametą, galimi trys pasirinkimai:
  - Visuomet priimti (*Always*). Pasirinkus šį variantą, sistema visuomet leis įkelti darbus, nors jie ir vėluos.
  - Niekada nepriimti (*Never*). Pasirinkus šį variantą, sistema nebeleis įkelti darbų, jei bus vėluojama.
  - Priimti darbus iki (*Until*). Pasirinkus šį variantą, sistema leis įkelti vėluojančius darbus iki tam tikros nustatytos datos, pvz. 2016 m. sausio 1 d. 00:00 val.

16 pav. Naujos užduoties kūrimas

4. Kuomet visi nustatymai yra įvesti, spausti *Save* mygtuką. Sistema praneš apie sėkmingą naujos užduoties užsaugojimą (žr. 17 pav.). Sistema nukreipia į kurso įrankių puslapį.

17 pav. Naujos užduoties sėkmingo užsaugojimo patvirtinimas


Nauja užduotis yra pridėta.

## 2 priedas: Diegimo aktas

KAUNO TECHNOLOGIJOS UNIVERSITETUI

### DIEGIMO AKTAS

Šiuo patvirtinu, jog 2016 m. balandžio 20 d. įmonėje UAB „Visma Lietuva“ viename iš programinės įrangos kūrimo projektų buvo įdiegta ir pritaikyta Linos Piatikonės pasiūlyta dokumentavimo technologija DITA.

  
TVIRTINU  
direktorius  
Mantas URBONAS

