

LATVIJAS UNIVERSITĀTES  
DATORIKAS FAKULTĀTES

**TĪMEKĻA VIETNE  
„LABO DARBU DARBNĪCA”  
SADAĻA – VIENS OTRAM**

**KVALIFIKĀCIJAS DARBS**

Autors: **Katrīna Ladutjko**

Studenta apliecības Nr.: k110038

Darba vadītājs: M.Dat. Mārtužs Jānis

RĪGA 2012

## ANOTĀCIJA

Sociāla tīmekļa vietne “Labo Darbu Darbnīca”, ar kuras palīdzību cilvēki varēs darīt un attīstīt labus un noderīgus darbus. Vietnes saturs sastāv no cilvēku darbiem, vēlmēm, interesēm, vajadzībām un pieredzes.

Modulis „Viens otram” ir tīmekļa vietnes pamatdoma, jo tieši šajā sadaļā lietotāji varēs ievietot ierakstus ar piedāvājumiem vai vēlmēm, ka arī apskatīt citu lietotāju ievietotus ierakstus, kā arī sazināties savā starpā.

Sistēma ir realizēta ar PHP programmēšanas valodas palīdzību, izmantojot bibliotēku „CodeIgniter” ar Model-View-Controller strukturēšanas principiem. Sistēma ir izstrādāta pēc spējās (Agile) pieejas metodes Scrum, ievērojot visus kontrolsaraksta punktus.

Atslēgvārdi: CodeIgniter, spējā (Agile) pieeja, COCOMO .

## ABSTRACT

### Website "LDD". Subsystem "Each other"

A social website "LDD" through which people can develop good and useful work. Site content is composed of human desires, interests, needs and experiences.

Module "Each other" is the basic idea of the website, there users will be able to insert offers and desires, as well as view other entries for users, as well as communicate with each other.

The system is realized with the help of PHP programming language, using the library "CodeIgniter" with the Model-View-Controller structuring principles. The system is designed by Agile methodology Scrum method close to Agile checklists.

Tags: CodeIgniter, capability (agile) approach, COCOMO.

## SATURS

1. Ievads .....	6
2. Lietotājstāsti .....	7
2.1. Ievads .....	7
2.1.1. Nodaļas nolūks .....	7
2.1.2. Darbības sfēra .....	7
2.1.3. Saistība ar citām nodaļām .....	7
2.2. Moduļa „Viens otram ” lietotājstāsti .....	7
2.2.1. Apskatīt visus piedāvājumus .....	7
2.2.2. Apskatīt visas vēlmes .....	7
2.2.3. Apskatīt ievietotas bildes .....	7
2.2.4. Sazināties ar ieraksta autoru .....	8
2.2.5. Apskatīt kopīgas intereses .....	8
2.2.6. Atzīmēt, ka ieraksts patīk .....	8
2.2.7. Šķirot visu ierakstu attēlošanu .....	8
2.2.8. Pievienot piedāvājumu .....	8
2.2.9. Pievienot vēlmi .....	8
2.2.10. Skatīt savus ierakstus – piedāvājumus .....	8
2.2.11. Skatīt savus ierakstus – vēlmes .....	9
2.2.12. Labot savu ierakstu – piedāvājumu .....	9
2.2.13. Labot savu ierakstu – vēlmi .....	9
2.2.14. Dzēst savu ierakstu – piedāvājumu vai vēlmi .....	9
2.2.15. Šķirot manu ierakstu attēlošanu – piedāvājumus un vēlmes .....	9
2.2.16. Atzīmēt ierakstu, kā pabeigtu .....	9
2.3. Vispārējās prasības .....	9
3. Akceptēšanas scenāriji .....	10
3.1. Ievads .....	10
3.1.1. Nodaļas nolūks .....	10
3.1.2. Darbības sfēra .....	10
3.1.3. Saistība ar citām nodaļām .....	10
3.2. Moduļa „Viens otram ” akceptēšanas scenāriji .....	11
3.2.1. Apskatīt visus piedāvājumus .....	11
3.2.3. Apskatīt ievietotas bildes .....	11
3.2.4. Sazināties ar ieraksta autoru .....	11
3.2.5. Apskatīt kopīgas intereses .....	11
3.2.6. Atzīmēt, ka ieraksts patīk .....	11
3.2.7. Šķirot visu ierakstu attēlošanu .....	12
3.2.8. Pievienot piedāvājumu .....	13
3.2.9. Pievienot vēlmi .....	13
3.2.10. Skatīt savus ierakstus – piedāvājumus .....	13

3.2.11. Skatīt savus ierakstus – vēlmes.....	14
3.2.12. Labot savu ierakstu – piedāvājumu .....	14
3.2.13. Labot savu ierakstu – vēlmi.....	14
3.2.14. Dzēst savu ierakstu – piedāvājumu vai vēlmi.....	15
3.2.15. Šķirot manu ierakstu attēlošanu – piedāvājumus un vēlmes .....	15
3.2.16. Atzīmēt ierakstu, kā pabeigtu .....	15
4. Programmatūras projektējuma apraksts.....	16
4.1. Ievads.....	16
4.1.1. Dokumenta nolūks.....	16
4.1.2. Darbības sfēra.....	16
4.2. Programmatūras moduļu projektējums.....	16
4.2.1. Vispārējais apraksts.....	16
4.2.2. Moduļa „Viens otram” savstarpējās atkarības .....	18
4.2.3. Sistēmas moduļa apraksts.....	18
4.3. Datu modelis .....	20
4.3.1. ER realizācijas modelis .....	20
4.3.2. Moduļa „Viens otram” datu tabulu projektējums .....	22
5. Testēšanas dokumentācija.....	25
5.1. Ievads .....	25
5.2. Testēšanas rezultāti .....	25
5.2.1. Pamatfunkcionalitāte .....	25
5.2.2. Funkcionalitāte .....	26
5.2.3. Papildfunkcionalitāte.....	31
5.3. Secinājumi .....	33
6. Projekta organizācija.....	34
7. Kvalitātes nodrošināšana .....	35
8. Konfigurācijas pārvaldība.....	36
9. Darbietilpības novērtējums .....	37
9.1. Sākotnēji aprēķināta darbietilpība .....	37
9.2. COCOMO kalkulators .....	37
9.3. Secinājumi .....	38
10. Programmatūras kods .....	39
10.1. offers.php(Kontrolieris) .....	39
10.2. offer.php(Modelis).....	42
11. Nobeigums.....	44
12. Izmantotā literatūra un avoti.....	45
13. Dokumentārā lapa.....	46

## 1. IEVADS

Daudzi sociālos tīklus izmanto kā laika kavēkli un izklaidi ikkatrā brīvā brīdī, kas nav īsti lietderīgs laika pavadīšanas veids. Šī projekta mērķis ir apvienot cilvēciskās vērtības ar mūsdienu tehnoloģijām, izveidojot tādu sociālo vietni, kas kopā ar socializēšanās iespējām vienlaikus sniegs arī nozīmīgus ieguvumus visiem tās lietotājiem, jo cilvēki šajā vietnē laiku pavadīs lietderīgi, palīdzot cits citam, sniedzot un saņemot atbalstu (lietas, pakalpojumus, konsultācijas utt.), kā arī nopelnot atzinības un rekomendācijas par labiem darbiem. Šī nav arī kārtējā labdarības organizācija, kuras mērķis ir lūgt cilvēkiem palīdzību. Mērķis ir palīdzēt cilvēkiem pašiem atrast veidu kā uzlabot dzīvi gan sev, gan citiem.

Manis izstrādātajā modulī „Viens otram” ir funkcionalitāte, ar kuras palīdzību cilvēki var dāvēt patīkamas emocijas gan sev, gan arī citiem, darot labus darbus, piepildīt sapņus, kā arī iegūt jaunus draugus un paziņas. Tīmekļa vietnes pamats ir savest cilvēkus kopā reālajā dzīvē, nevis virtuālajā, padarīt pasauli draudzīgāku un labāku. Kāpēc gan neiepriecināt citu, ja tas ir Tavos spēkos?

Modulis „Viens otram” ietver sevī piedāvājumu un vēlmju pievienošanu, savu ierakstu apskatīšanu, kā arī iespēju apskatīt visus tīmekļa vietnē ievietotos piedāvājumus un vēlmes. Tāpat no konkrēta ieraksta ir iespējams sazināties ar ieraksta autoru, uzrakstot privātu iekšējo ziņojumu jeb vēstuli.

Visas augstākminētas funkcijas ir pieejamas tikai pierakstītam lietotājam (talāk apzīmēts kā lietotājs).

## 2. LIETOTĀJSTĀSTI

### 2.1. Ievads

#### 2.1.1. *Nodaļas nolūks*

Lietotājstāstu nolūks ir aprakstīt Tīmekļa vietnes „Labo darbu darbnīca” moduļa „Viens otram” prasības. Ar lietotājstāstu palīdzību tiek definēta izstrādājamā moduļa funkcionalitāte, kā arī sagaidāmie rezultāti.

#### 2.1.2. *Darbības sfēra*

Lietotājstāstu galvenā mērķauditorija ir testētāji un programmatūras izstrādātāji.

#### 2.1.3. *Saistība ar citām nodaļām*

Balstoties uz šo nodaļu – lietotājstāsti, tiks izstrādāta 3. nodaļa „Akceptēšanas scenāriji”.

### 2.2. Moduļa „Viens otram ” lietotājstāsti

#### 2.2.1. *Apskatīt visus piedāvājumus*

Lietotājs piekļūst visām tīmekļa vietnes funkcijām, šajā situācijā – apskatīt visu lietotāju ievietotos sadaļā „Piedāvājumi” ierakstus. Katrā ieraksta ir redzams autora vārds, uzvārds, profila bilde, izveidošanas datums, piedāvājuma apraksts, joma pie kuras pieder šis konkrētais piedāvājums, kā arī atrašanas vieta.

#### 2.2.2. *Apskatīt visas vēlmes*

Lietotājs piekļūst visām tīmekļa vietnes funkcijām, šajā situācijā – apskatīt visu lietotāju ievietotos sadaļā „Piedāvājumi” ierakstus. Katrā ieraksta ir redzams autora vārds, uzvārds, profila bilde, izveidošanas datums, vēlmes apraksts, joma pie kuras pieder šis konkrētais piedāvājums, kā arī atrašanas vieta.

#### 2.2.3. *Apskatīt ievietotas bildes*

Ja ieraksta autors ir pievienojis bildi, lietotājam ir iespējams to apskatīt.

#### **2.2.4. Sazināties ar ieraksta autoru**

Ar ieraksta autoru ir iespējams sazināties, aizsūtot privāto iekšējo ziņojumu jeb vēstuli, ierakstot tēmu un paša ziņojuma tekstu.

#### **2.2.5. Apskatīt kopīgas intereses**

Lietotājam ir iespēja apskatīt kopīgas intereses ar ieraksta autoru, taču šo iespēju programmēja cits cilvēks.

#### **2.2.6. Atzīmēt, ka ieraksts patīk**

Lietotājam ir iespēja atzīmēt, ka konkrētais ieraksts patīk, nospiežot pogu „Patīk”. Pēc pogas nospiešanas attēlojas kopējais skaits, cik cilvēkiem šis ieraksts patika.

#### **2.2.7. Šķirot visu ierakstu attēlošanu**

Lietotājam ir iespēja šķirot visu piedāvājumu attēlošanu pēc izveidošanas datuma – jaunākie vai vecākie ieraksti, kā arī piedāvājumus ir iespējams attēlot pēc lietotāja profilā pievienotam interesēm, taču intereses taisīja cits cilvēks.

Savukārt vēlmes ir iespējams šķirot pēc pievienošanas datuma – jaunākie un vecākie ieraksti.

#### **2.2.8. Pievienot piedāvājumu**

Lietotājam ir iespējams pievienot savu piedāvājumu, aizpildot ievadformas laukus – nosaukums, apraksts, tips – bārteris vai dāvinājums, joma un atrašanas vieta – obligāti lauki, kurus neaizpildot nav iespējams pievienot ierakstu, savukārt bildes pievienošana nav obligāta, to var pievienot divos veidos – ievietojot saiti uz atrašanas vietu citos resursos, kā arī augšuplādējot no sava datora. Aprakstam ir noteikts maksimālu simbolu skaits – 1000.

#### **2.2.9. Pievienot vēlmi**

Lietotājam ir iespēja pievienot savu vēlmi, aizpildot ievadformas laukus – nosaukums, apraksts, joma ir obligāti lauki, uz kurus neaizpildot nav iespējams pievienot ierakstu. Aprakstam ir noteikts maksimālu simbolu skaits – 1000.

#### **2.2.10. Skatīt savus ierakstus – piedāvājumus**

Lietotājam ir iespēja apskatīt visus savus piedāvājumus sadaļā „Mani ieraksti”. Tāpat katrā piedāvājumā ir pieejamas pogas „Labot” un „Dzēst”, kā arī poga „Pabeigts”

### ***2.2.11. Skatīt savus ierakstus – vēlmes***

Lietotājam ir iespēja apskatīt visas savas vēlmes sadaļā „Mani ieraksti”. Tāpat katrā piedāvājumā ir pieejamas pogas „Labot” un „Dzēst” ”,kā arī poga „Pabeigts”.

### ***2.2.12. Labot savu ierakstu – piedāvājumu***

Lietotājam ir iespēja labot savu piedāvājumu, ar iespēju labot visus ievadformas laukus. Obligātie ievadformas lauki nedrīkst būt tukši.

### ***2.2.13. Labot savu ierakstu – vēlmi***

Lietotājam ir iespēja labot savu vēlmi, ar iespēju labot visus ievadformas laukus. Obligātie ievadformas lauki nedrīkst būt tukši.

### ***2.2.14. Dzēst savu ierakstu – piedāvājumu vai vēlmi***

Lietotājam ir iespēja dzēst savu piedāvājumu vai vēlmi, nospiežot pogu „Dzēst”. Lietotājam paradās iznirstošais logs ar brīdinājumu par dzēšanu, ar iespēju to apstiprināt vai atcelt.

### ***2.2.15. Šķirot manu ierakstu attēlošanu – piedāvājumus un vēlmes***

Lietotājam ir iespēja šķirot savu ierakstu attēlošanu pēc izveidošanas datuma – jaunākie vai vecākie ieraksti.

### ***2.2.16. Atzīmēt ierakstu, kā pabeigtu***

Lietotājam ir iespēja atzīmēt ierakstu – piedāvājumu vai vēlmi, kā pabeigtu, līdz ar ko ta paliek neaktīva un ir redzama profila sadaļā „Paveiktie labie darbi”.

## **2.3. Vispārējās prasības**

- Ievadformām ir jābūt validācijai uz obligātajiem laukiem, to simbolu maksimālo skaitu (ja nepieciešams, tad rādīt lietotājam, cik simbolu ir atlicis);
- Dzēšot, vietās, kur tas nepieciešams, paziņojums “Vai tiešām dzēst?”;
- Failiem jābūt UTF-8 bez BOM formātā;

Microsoft Internet Explorer 8

Microsoft Internet Explorer 9

Mozilla Firefox 9.0.

Opera Software 11

## 3. AKCEPTĒŠANAS SCENĀRIJI

### 3.1. Ievads

#### 3.1.1. Nodaļas nolūks

Lietotāju stāsti ir paredzēti, lai definētu sistēmas funkcionalitāti, savukārt akceptēšanas scenāriju nolūks ir aprakstīt situācijas, kad sistēma ir akceptējama kā pareizi funkcionējoša.

#### 3.1.2. Darbības sfēra

Akceptēšanas scenāriju galvenā mērķauditorija ir testētāji un programmatūras izstrādātāji.

#### 3.1.3. Saistība ar citām nodaļām

„Akceptēšanas scenāriji” ir izstrādāti balstoties uz 2. nodaļu „Lietotājstāsti”.

## **3.2. Moduļa „Viens otram ” akceptēšanas scenāriji**

### ***3.2.1. Apskatīt visus piedāvājumus***

Atbilst **2.2.1.** nodaļā aprakstītajam lietotājistāstam.

- 3.2.1.1 Nospiežot „Viens otram” lietotājam tiek atteloti visu portāla lietotāju aktīvie piedāvājumi.

### ***3.2.2. Apskatīt visas vēlmes***

Atbilst **2.2.2.** nodaļā aprakstītajam lietotājistāstam.

- 3.2.2.1 Nospiežot „Vēlmes” lietotājam tiek atteloti visu portāla lietotāju aktīvie piedāvājumi.

### ***3.2.3. Apskatīt ievietotas bildes***

Atbilst **2.2.3.** nodaļā aprakstītajam lietotājistāstam.

- 3.2.3.1 Apskatot konkrētu piedāvājumu, nospiežot „Bilde” lietotājam tiek attēlota piedāvājumā ievietota bilde.

### ***3.2.4. Sazināties ar ieraksta autoru***

Atbilst **2.2.4.** nodaļā aprakstītajam lietotājistāstam.

- 3.2.4.1 Apskatot konkrētu ierakstu – piedāvājumu vai vēlmi, nospiežot „Sazinaties” lietotājam tiek attēlota vēstules ievadforma, kurā ir iespējams ierakstīt tēmu un ziņojuma tekstu,
- 3.2.4.2 Tiek attēlota poga „Sūtīt vēstuli”- pēc pogas nospiešanas vēstule tiek aizsūtīta ieraksta autoram. Aizsūtītu vēstuli ir iespējams apskatīt profila aizsūtītajās vēstulēs.

### ***3.2.5. Apskatīt kopīgas intereses***

Atbilst **2.2.5.** nodaļā aprakstītajam lietotājistāstam.

- 3.2.5.1 Apskatot konkrētu ierakstu – piedāvājumu vai vēlmi, nospiežot „Kopīgas intereses”, lietotājam tiek atteloti piedāvājumi, kas sakrīt pēc lietotāja un ieraksta autora interesēm, preteja gadījuma parādās paziņojums – „Nav piedāvājumu”,
- 3.2.5.2 Lietotājam tiek attēlotas tiešās sakritības intereses, preteja gadījuma parādās paziņojums – „Nav kopīgo interešu”.

### ***3.2.6. Atzīmēt, ka ieraksts patīk***

Atbilst **2.2.6.** nodaļā aprakstītajam lietotājistāstam.

3.2.6.1 Apskatot konkrētu ierakstu – piedāvājumu vai vēlmi, nospiežot pogu „Patīk”, tiek saglabāta lietotāja izvēle, ka ieraksts patīk, ka arī lietotājam attelojas skaits, cik lietotāji kopumā ir atzīējuši šo ierakstu.

### **3.2.7. Škirot visu ierakstu attēlošanu**

Atbilst **2.2.7.** nodaļā aprakstītajam lietotājstāstam.

3.2.7.1 Apskatot visus piedāvājumus, lietotājam ir iespēja škirot tos:

- Pēc izveidošanas datuma – jaunākie sākumā, vecākie sākumā;
- Pēc interesēm – atbilstoši pievienotām profilā interesēm un to prioritātem.

3.2.7.2 Apskatot vēlmes, lietotājam ir iespēja škirot tās pēc izveidošanas datuma:

- Pēc izveidošanas datuma – jaunākie sākumā, vecākie sākumā;

### **3.2.8. Pievienot piedāvājumu**

Atbilst **2.2.8.** nodaļā aprakstītajam lietotājistāstam.

3.2.8.1 Atvērot piedāvājuma pievienošanas formu, lietotājam jāaizpilda ievadformas obligātie lauki:

- Nosaukums – obligāti aizpildams lauks, to atstājot tukšu, parādīsies brīdinājuma paziņojums „Piedāvājuma nosaukums nevar būt tukšs”;
- Apraksts – obligāti aizpildams lauks, to atstājot tukšu, parādīsies brīdinājuma paziņojums „Apraksts nevar būt tukšs”. Apraksta simbolu skaits nedrīkst pārsniegt 1000 zīmes;
- Joma– obligāti aizpildams lauks, to atstājot tukšu, parādīsies brīdinājuma paziņojums „Joma nevar būt tukšs”;
- Atrašanas vieta – obligāti aizpildams lauks, to atstājot tukšu, parādīsies brīdinājuma paziņojums „Atrašanas vieta nevar būt tukšs”;

3.2.8.2 Jānorāda piedāvājuma tips – bārteris vai dāvinājums, pēc noklusējuma ir atzīmēts bārteris;

3.2.8.3 Bilde pievienošana nav obligāta, taču ja to pievieno, bildei jāattēlojas piedāvājuma pēc pievienošanas.

### **3.2.9. Pievienot vēlmi**

Atbilst **2.2.9.** nodaļā aprakstītajam lietotājistāstam.

3.2.9.1 Atvērot vēlmes pievienošanas formu, lietotājam jāaizpilda ievadformas obligātie lauki:

- Nosaukums – obligāti aizpildams lauks, to atstājot tukšu, parādīsies brīdinājuma paziņojums „Vēlmes nosaukums nevar būt tukšs”;
- Apraksts – obligāti aizpildams lauks, to atstājot tukšu, parādīsies brīdinājuma paziņojums „Apraksts nevar būt tukšs”. Apraksta simbolu skaits nedrīkst pārsniegt 1000 zīmes;
- Joma– obligāti aizpildams lauks, to atstājot tukšu, parādīsies brīdinājuma paziņojums „Joma nevar būt tukšs”;
- Atrašanas vieta – obligāti aizpildams lauks, to atstājot tukšu, parādīsies brīdinājuma paziņojums „Atrašanas vieta nevar būt tukšs”.

### **3.2.10. Skatīt savus ierakstus – piedāvājumus**

Atbilst **2.2.10.** nodaļā aprakstītajam lietotājistāstam.

3.2.10.1 Nospiežot „Mani ieraksti” lietotājam tiek attēloti visi savi aktīvie piedāvājumi.

### **3.2.11. Skatīt savus ierakstus – vēlmes**

Atbilst **2.2.11.** nodaļā aprakstītajam lietotājstāstam.

3.2.11.1 Nospiežot „Mani ieraksti” lietotājam tiek attēlotas visas savas aktīvās vēlmes.

### **3.2.12. Labot savu ierakstu – piedāvājumu**

Atbilst **2.2.12.** nodaļā aprakstītajam lietotājstāstam.

3.2.12.1 Atvērot piedāvājuma labošanas formu, lietotājam ir iespēja labot ievadformas laukus:

- Nosaukums – obligāti aizpildams lauks, to atstājot tukšu, parādīsies brīdinājuma paziņojums „Piedāvājuma nosaukums nevar būt tukšs”;
- Apraksts – obligāti aizpildams lauks, to atstājot tukšu, parādīsies brīdinājuma paziņojums „Apraksts nevar būt tukšs”. Apraksta simbolu skaits nedrīkst pārsniegt 1000 zīmes;
- Joma– obligāti aizpildams lauks, to atstājot tukšu, parādīsies brīdinājuma paziņojums „Joma nevar būt tukšs”;
- Atrašanas vieta – obligāti aizpildams lauks, to atstājot tukšu, parādīsies brīdinājuma paziņojums „Atrašanas vieta nevar būt tukšs”;

3.2.12.2 Jānorāda piedāvājuma tips – bārteris vai dāvinājums, pēc noklusējuma ir atzīmēts bārteris;

3.2.12.3 Bilde pievienošana nav obligāta, labojot, iepriekšējā bilde netiek saglabāta, jāvada no jauna, taču ja to pievieno, bildei jāattēlojas piedāvājuma pēc pievienošanas.

### **3.2.13. Labot savu ierakstu – vēlmi**

Atbilst **2.2.13.** nodaļā aprakstītajam lietotājstāstam.

3.2.13.1 Atvērot vēlmes labošanas formu, lietotājam ir iespēja labot ievadformas laukus:

- Nosaukums – obligāti aizpildams lauks, to atstājot tukšu, parādīsies brīdinājuma paziņojums „Vēlmes nosaukums nevar būt tukšs”;
- Apraksts – obligāti aizpildams lauks, to atstājot tukšu, parādīsies brīdinājuma paziņojums „Apraksts nevar būt tukšs”. Apraksta simbolu skaits nedrīkst pārsniegt 1000 zīmes;
- Joma– obligāti aizpildams lauks, to atstājot tukšu, parādīsies brīdinājuma paziņojums „Joma nevar būt tukšs”;
- Atrašanas vieta – obligāti aizpildams lauks, to atstājot tukšu, parādīsies brīdinājuma paziņojums „Atrašanas vieta nevar būt tukšs”.

### ***3.2.14. Dzēst savu ierakstu – piedāvājumu vai vēlmi***

Atbilst **2.2.14.** nodaļā aprakstītajam lietotājstāstam.

- 3.2.14.1 Lietotājam nospiežot pogu „Dzēst” un apstiprinot savu izvēli, ieraksts – piedāvājums vai vēlme, tiek dzēsti gan no aktīvajiem ierakstiem – iedāvājumiem vai vēlmēm, gan arī no lietotāja profila sadaļas „Paveiktie labie darbi”

### ***3.2.15. Šķirot manu ierakstu attēlošanu – piedāvājumus un vēlmes***

Atbilst **2.2.15.** nodaļā aprakstītajam lietotājstāstam.

- 3.2.15.1 Lietotājam izvēloties šķirošanu pēc jaunākajiem – sākumā attēlojas jaunākie pievienotie ieraksti
- 3.2.15.2 Lietotājam izvēloties šķirošanu pēc vecākajiem – sākumā attēlojas vecākie pievienotie ieraksti

### ***3.2.16. Atzīmēt ierakstu, kā pabeigtu***

Atbilst **2.2.16.** nodaļā aprakstītajam lietotājstāstam.

- 3.2.16.1. Lietotājam atzīmējot ierakstu – piedāvājumu vai vēlmi, kā pabeigtu, piedāvājums tiek pārcelts no aktīvajiem piedāvājumiem, respektīvi no sadaļas „Viens otram” uz lietotāja profila sadaļu „Paveiktie labie darbi”.

## 4. PROGRAMMATŪRAS PROJEKTĒJUMA APRAKSTS

### 4.1. Ievads

#### 4.1.1. Dokumenta nolūks

Programmatūras projektējuma apraksta (PPA) nolūks ir izstrādājamās Tīmekļa vietnes „Labo darbu darbnīca” moduļa „Viens otram” sistēmas projektējuma aprakstīšana. PPA tiek ņemts vērā 2. nodaļā aprakstītie lietotārstāsti un 3. nodaļā aprakstītie lietotārstāstu akceptēšanas scenāriji.

#### 4.1.2. Darbības sfēra

Programmatūras projektējuma apraksts tika veidots saskaņā ar PHP (versija: 5.3.11-1~dotdeb.0), MySQL (versija: mysqlnd 5.0.8-dev) un CodeIgniter (versija: 2.1.0) vidēm. PPA mērķauditorija ir programmatūras izstrādātāji un testētāji.

### 4.2. Programmatūras moduļu projektējums

#### 4.2.1. Vispārējais apraksts

Sistēma tika būvēta izmantojot uz MVC (Model –View –Controller) strukturēšanas principiem balstīto „CodeIgniter” palīgfunkciju bibliotēku - sistēmas moduļi ir sadalīti trīs daļās:

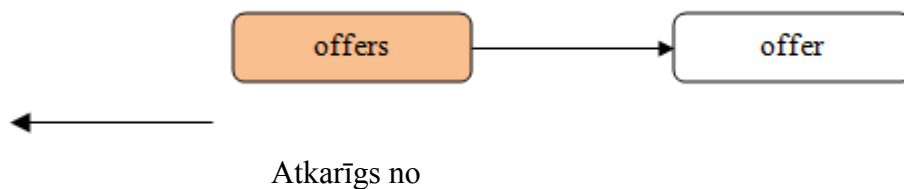
- Model (modelis) atbild par pieslēgšanos datubāzei, datu atlasīšanu, iegūšanu, saglabāšanu tajā (php valodā),
- View (skats) - satur klases un saskarnes, kas nodrošina modeļa vizuālo attēlojumu. Skatu var nosaukt par Modeļa stāvokļa reprezentāciju, kas tikai lasa datus no Modeļa izmantojot Modeļa klases metodes (HTML, css, javascript valodās),
- Controller (kontrolieris) - saņem no lietotāja signālus un translē tos pieprasījumos pie modeļa vai skata, kuri liek modelim vai skatam mainīties un atbildēt par veiktam izmaiņām controllerim, atbild par modeļu izsaukšanu, skatu ielādi, no skatiem izsaukto funkciju apstrādāšanu (php valodā) .

Galveno serveru funkcionalitāti veido „CodeIgniter” modeļi un kontrolieri, bet skati gan ir tikai vizuālais noformējums.

Piemērs - visas lietotāja funkcijas, kuras tiek izmantotas modulā „Viens otram” izstrādē, tiks projektētas „offers” (kontrolieris) un „offer”(modelis) failos, ja funkcijai ir nepieciešama pieslēgšanās datubāzei, tad „offers” modulis izsauc „offer” modulī atbilstošo funkciju, kas atgriež atpakaļ datus no datubāzes. Sistēmas skatos tapat tiks plaši izmantotas CodeIgniter bibliotēku palīgi. Sistēma ir vairāk tendēta uz funkcionalitāti, nevis uz vizuālo izskatu, jo ta ir izstrādes pirmā versija.

### 4.2.2. Moduļa "Viens otram" savstarpējās atkarības

Moduļu savstarpējās atkarības parādītas 4.1. attēlā.. Oranžā krāsā iekrāsots „CodeIgniter” kontrolieris, taču baltā - modelis.



4.1. att. Moduļa savstarpējās atkarības

### 4.2.3. Sistēmas moduļa apraksts

Moduļa funkcijas atribūts apraksta, kādas funkcijas tas nodrošina. Iekavās moduļa funkcijām ir minēts lietotārstāta numurs, kurā ir aprakstīts šīs funkcijas uzdevums.

#### 4.2.3.1 Modulis „Viens otram”

<b>Tips</b>	„CodeIgniter” kontrolieris
<b>Nolūks</b>	Nodrošināt lietotājam pieejamas funkcijas. Ja funkcijās ir nepieciešams iegūt, saglabāt, dzēst datus no datubāzes, tad pāradresē uz abilstošām funkcijām modelī .
<b>Galvenā datne</b>	offers.php
<b>Funkcijas</b>	Modulis nodrošina šādas funkcijas: <ul style="list-style-type: none"><li>• Visu portāla lietotāju aktīvo piedāvājumu skata attēlošana (2.2.1.);</li><li>• Visu portāla lietotāju aktīvo vēlmju skata attēlošana (2.2.2.);</li><li>• Piedāvājumu bilžu attēlošana (2.2.3.);</li><li>• Vēstuļu sūtīšanas formas izsaukšana piedāvājumos, vēlmēs (2.2.4.);</li><li>• Kopīgo interešu atlasīšana un attēlošana (2.2.5.);</li><li>• Pogas „Patīk’ attēlošana un funkcionēšana (2.2.6.);</li><li>• Visu portāla lietotāju aktīvo piedāvājumu vai vēlmju šķirošana(2.2.7.);</li><li>• Piedāvājuma pievienošana un ievadformas attēlošana (2.2.8.);</li><li>• Vēlmes pievienošana un ievadformas attēlošana (2.2.9.);</li><li>• Visu savu aktīvo piedāvājumu skata attēlošana (2.2.10.);</li><li>• Visu portāla lietotāju aktīvo vēlmju skata attēlošana(2.2.11);</li></ul>

	<ul style="list-style-type: none"> <li>• Piedāvājuma labošana un saglabāšana (2.2.12.);</li> <li>• Vēlmes labošana un saglabāšana (2.2.13.);</li> <li>• Sava ieraksta – piedāvājuma, vēlmes, dzēšana no datubāzes (2.2.14);</li> <li>• Savu ierakstu – piedāvājumu vai vēlmju šķirošana (2.2.15);</li> <li>• Savu ieraksta – piedāvājumu vai vēlmes atzīmēšana par pabeigtu (2.2.16.)</li> </ul>
<b>Atkarība</b>	Šis kontrolieris ir saistīts ar modeli „offer”
<b>Pakļautība</b>	Nav

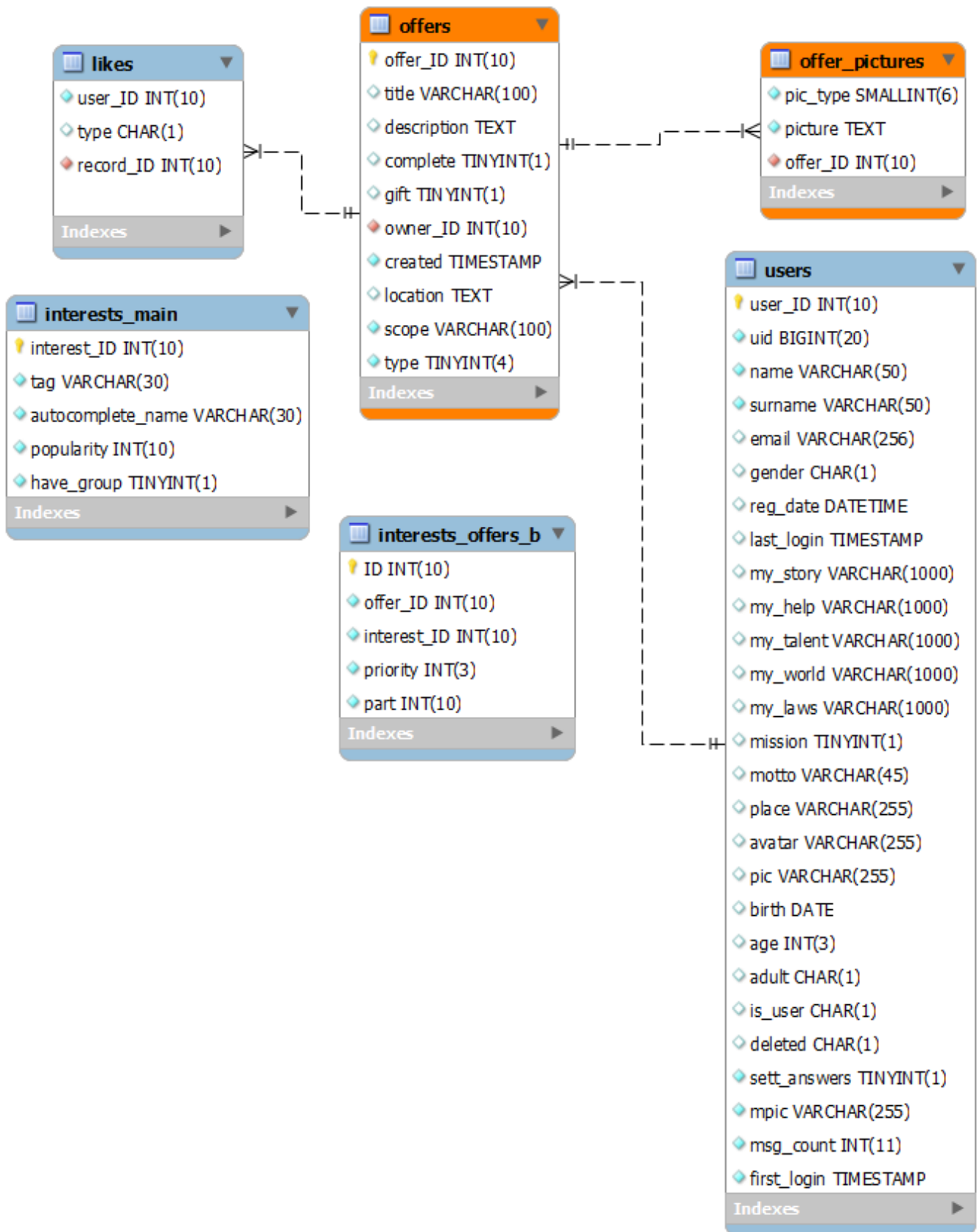
<b>Tips</b>	„CodeIgniter” modelis
<b>Nolūks</b>	Nodrošināt funkcijām no „offers” kontroliera piekļuvi datubāzei.
<b>Galvenā datne</b>	offer.php
<b>Funkcijas</b>	<p>Modulis nodrošina šādas funkcijas:</p> <ul style="list-style-type: none"> <li>• Visu portāla lietotāju aktīvo piedāvājumu atlasīšana (2.2.1.);</li> <li>• Visu portāla lietotāju aktīvo vēlmju atlasīšana (2.2.2.);</li> <li>• Piedāvājumu bilžu atlasīšana (2.2.3.);</li> <li>• Vēstuļu sūtīšanas formas izsaukšana piedāvājumos, vēlmēs (2.2.4.);</li> <li>• Kopīgo interešu atlasīšana (2.2.5.);</li> <li>• Atlasa skaitu, cik daudz lietotājiem patika konkrētais ieraksts (2.2.6.);</li> <li>• Visu portāla lietotāju aktīvo piedāvājumu vai vēlmju izveidošanas datuma atlasīšana (2.2.7.);</li> <li>• Piedāvājuma datu ievietošana datubāzē (2.2.8.);</li> <li>• Vēlmes datu ievietošana datubāzē (2.2.9.);</li> <li>• Visu savu aktīvo piedāvājumu atlasīšana (2.2.10.);</li> <li>• Visu portāla lietotāju aktīvo vēlmju atlasīšana (2.2.11);</li> <li>• Piedāvājuma datu ievietošana datubāzē labošanas gadījumā (2.2.12.);</li> <li>• Vēlmes datu ievietošana datubāzē labošanas gadījumā (2.2.13.);</li> <li>• Sava ieraksta – piedāvājuma, vēlmes, datu dzēšana no datubāzes</li> </ul>

	(2.2.14); <ul style="list-style-type: none"> <li>• Savu ierakstu – piedāvājumu vai vēlmju izveidošanas datuma atlasīšana (2.2.15);</li> <li>• Savu ieraksta – piedāvājumu vai vēlmes datu ievietošana datubāzē (2.2.16.)</li> </ul>
<b>Atkarība</b>	Šis modelis ir saistīts ar kontrolieri „offers”
<b>Pakļautība</b>	Nav

## 4.3. Datu modelis

### 4.3.1. ER realizācijas modelis

Tīmekļa vietnes „Labo darbu darbnīca” moduļa „Viens otram” ER realizācijas modelis ir attēlots 4.3. attēlā. Moduļa „Viens otram” tiešā izstrādē tika izmantotas tabulas „offers” un „offer\_picture” (iekrāsotas oranžā krāsā). Savukārt no tabulām „likes”, „interests\_main”, „interests\_offers\_b” un „users” tika izmantota informācija.



4.3. att. ER realizācijas modelis

### ***4.3.2. Moduļa „Viens otram” datu tabulu projektējums***

Datu tabulu aprakstā, kolonnā „Datu tips” atrodas lauka „MySQL 5.0.8-dev.” datu tips. Kolonnā „Lauka tips” atrodamo vērtību saīsinājumi nozīmē :

- PK - primary key,
- FK - foreign key,
- U - unique ,
- A – autoincrement.

#### 4.3.2.1 Tabula „offers”

Nr. p.k.	Lauka nosaukums	Datu tips	Obligāts	Lauka tips	Apraksts
1.	offer_ID	INT(10)	Jā	PK/U/A	Primārā atslēga.
2.	title	VARCHAR(100)	Jā		Nosaukums
3.	description	TEXT	Jā		Apraksts
4.	complete	TINYINT(1)	Jā		Vai ieraksts(piedāvājums, vēlme) ir aktīvs(1)
5.	gift	TINYINT(1)	Jā		Bārteris(0) vai dāvinājums(1)
6.	owner_ID	INT(10)	Jā	FK	Ieraksta (piedāvājuma, vēlmes) autora identifikātors
7.	created	TIMESTAMP	Jā		Izveidošanas datums
8.	location	TEXT	Jā		Atrašanas vieta
9.	scope	VARCHAR(100)	Jā		Joma
10.	type	TINYINT(4)	Jā		Ieraksta tips – piedāvājums(0) vai vēlme(1)

#### 4.3.2.1 Tabula „offer\_pictures”

Nr. p.k.	Lauka nosaukums	Datu tips	Obligāts	Lauka tips	Apraksts
1.	pic_type	SMALLINT(6)	Nē		Bildes tips – augšuplādēta (1) vai saite no citiem resursiem (0)
2.	picture	TEXT	Nē		Bilde
3.	offer_ID	INT(10)	Jā	FK	Ieraksta – piedāvājuma vai vēlmes identifikātors

## 5. TESTĒŠANAS DOKUMENTĀCIJA

### 5.1. Ievads

Šajā nodaļā ir aprakstīti Tīmekļa vietnes „Labo darbu darbnīca” modeļa „Viens otram” akcepttestēšanas rezultāti.

Tā kā projekts tika izstrādāts pēc spējās (Agile) pieejas un izstrāde bija sadalīta iterācijās, tad testēšana tika veikta pēc katras no 3 iterācijām. Iterāciju testos tika testētas visas izstrādātās iterācijas funkcionalitātes.

Testējot, tiek pieņemts, ka sistēmas kods bija pieejams. Testi tika izveidoti tā, lai pārbaudītu visus ievadlaukus, to validāciju un programmatūras saskarni. Iterācijas tika sadalītas pēc funkcionalitātes prioritātes.

### 5.2. Testēšanas rezultāti

#### 5.2.1. Pamatfunkcionalitāte

Šajā iterācijā tika izstrādāta funkcionalitāte, kas atbilst 2.2.1., 2.2.2., 2.2.7., 2.2.10., 2.2.11. , 2.2.15. nodaļās aprakstītajiem lietotājstāstiem, kuriem atbilst akceptēšanas scenāriji, kas aprakstīti atbilst 3.2.1., 3.2.2., 3.2.7., 3.2.10., 3.2.11. , 3.2.15. nodaļās. Šie lietotājstāsti ir realizēti 4.2.3.1. un 4.2.3.2. nodaļās aprakstītajos moduļos – „offers” un „offer”. Šī moduļa parējā funkconalitāte – funkcijas, kas vēl tika papildus pievienotas izstrādei tika testētas otrajā un trešajā iterācijās.

Nr.	Akcept. scen.	Testa apraksts	Paredzētais rezultāts	Rezultāts
1.	3.2.1.	Lietotājs ieiet sadaļā „Viens otram”	Lietotājam tiek attēloti visu portāla lietotāju aktīvie piedāvājumi, sašķiroti pēc interesēm	+
2.	3.2.2.	Atrodies sadaļā „Viens otram”, lietotājs nospiež saiti „Vēlas”	Lietotājam tiek attēlotas visu portāla lietotāju aktīvas vēlmes, sašķirotas pēc jaunākajiem.	+
3.	3.2.7.	Atrodies sadaļā „Viens otram” un apskatot	Lietotājam tiek attēloti visu portāla lietotāju aktīvie	+

		piedāvājumu, lietotājs izvēlās sašķirot to attēlošanu pēc jaunākajiem	piedāvājumi, sašķiroti pēc jaunākajiem	
4.	3.2.7.	Atrodies sadaļā „Viens otram” un apskatot piedāvājumus, lietotājs izvēlās sašķirot to attēlošanu pēc vecākajiem	Lietotājam tiek attēloti visu portāla lietotāju aktīvie piedāvājumi, sašķiroti pēc vecākajiem	+
5.	3.2.7.	Atrodies sadaļā „Viens otram” – „Vēlmes”, lietotājs izvēlās sašķirot vēlmju attēlošanu pēc vecākajām	Lietotājam tiek attēlotas visu portāla lietotāju aktīvas vēlmes, sašķirotas pēc vecākajām	+
5.	3.2.10	Atrodies sadaļā „Viens otram”, lietotājs apskata saiti „Mani ieraksti” – „Piedāvājumi”	Lietotājam tiek attēloti visi savi aktīvie piedāvājumi, sašķiroti pēc jaunākajiem	+
6.	3.2.11.	Atrodies sadaļā „Viens otram”, lietotājs apskata saiti „Mani ieraksti” – „Vēlmes”	Lietotājam tiek attēlotas visas savas aktīvās vēlmes, sašķirotas pēc jaunākajām	+
7.	3.2.15.	Atrodies sadaļā „Viens otram”, un apskatot saiti „Mani ieraksti” – „Piedāvājumi”, lietotājs izvēlas šķirot tos pēc vecākajiem	Lietotājam tiek attēloti visi savi aktīvie piedāvājumi, sašķiroti pēc vecākajiem	+
8.	3.2.15.	Atrodies sadaļā „Viens otram”, un apskatot saiti „Mani ieraksti” – „Vēlmes”, lietotājs izvēlas šķirot tās pēc vecākajām	Lietotājam tiek attēlotas visas savas aktīvās vēlmes, sašķirotas pēc vecākajām	+

### 5.2.2. Funkcionalitāte

Šajā iterācijā tika izstrādāta funkcionalitāte, kas atbilst 2.2.8., 2.2.9., 2.2.12., 2.2.13. , 2.2.14. nodaļās aprakstītajiem lietotājtāstiem, kuriem atbilst akceptēšanas scenāriji, kas aprakstīti

atbilst 3.2.8., 3.2.9., 3.2.12., 3.2.13. , 3.2.14. nodaļās. Šie lietotājstāsti ir realizēti 4.2.3.1. un 4.2.3.2. nodaļās aprakstītajos moduļos – „offers” un „offer”. Šī moduļa parējā funkconalitāte – funkcijas, kas vēl tika papildus pievienotas izstrādei tika testētas trešajā iterācijā.

Nr.	Akcept. scen.	Testa apraksts	Paredzētais rezultāts	Rezultāts
1.	3.2.8.	Atrodies sadaļā „Viens otram” – „Pievienot ierakstu”, lietotājs izvēlas „Ievietot piedāvājumu”	Lietotājam tiek attēloti ievadformas lauki: <ul style="list-style-type: none"> <li>• piedāvājuma</li> <li>• nosaukums,</li> <li>• apraksts,</li> <li>• joma,</li> <li>• atrašanas vieta,</li> <li>• bildes url,</li> <li>• bilde,</li> <li>• piedāvājuma tipa izvēle</li> <li>• poga „Pievienot”</li> </ul>	+
2.	3.2.8.	Atrodies sadaļā „Viens otram” – „Pievienot ierakstu” - „Ievietot piedāvājumu”, lietotājs aizpilda ievadformas laukus	<ul style="list-style-type: none"> <li>• Nosaukums – obligāti aizpildāms lauks, to atstājot tukšu, tiks attēlots brīdinājuma paziņojums „Piedāvājuma nosaukums nevar būt tukšs”;</li> <li>• Apraksts – obligāti aizpildāms lauks, to atstājot tukšu, tiks attēlots brīdinājuma paziņojums „Apraksts nevar būt tukšs”. Apraksta simbolu skaits nedrīkst pārsniegt 1000 zīmes;</li> <li>• Joma– obligāti</li> </ul>	+

			<p>aizpildāms lauks, to atstājot tukšu, tiks attēlots brīdinājuma paziņojums „Joma nevar būt tukšs”;</p> <ul style="list-style-type: none"> <li>• Atrašanas vieta – obligāti aizpildāms lauks, to atstājot tukšu, tiks attēlots brīdinājuma paziņojums „Atrašanas vieta nevar būt tukšs”;</li> <li>• Piedāvājuma tips – bārteris vai dāvinājums, pēc noklusējuma ir atzīmēts bārteris;</li> <li>• Lauks bildes saites ierakstīšanai;</li> <li>• Bildes augšupielādes modālais dialoglogs ;</li> </ul>	
3.	3.2.8.	<p>Atrodies sadaļā „Viens otram” – „Pievienot ierakstu” - „Ievietot piedāvājumu”, lietotājs, veiksmīgi aizpildījis ievadformas laukus, nospiež pogu „Pievienot”</p>	<p>Piedāvājums tiek pievienots visu aktīvu piedāvājumu sarakstam</p>	+
4.	3.2.9.	<p>Atrodies sadaļā „Viens otram” – „Pievienot ierakstu”, lietotājs izvēlas „Ievietot vēlmi”</p>	<p>Lietotājam tiek attēloti ievadformas lauki:</p> <ul style="list-style-type: none"> <li>• vēlmes nosaukums,</li> <li>• apraksts,</li> <li>• joma,</li> <li>• atrašanas vieta.</li> </ul>	+

5.	3.2.9.	<p>Atrodies sadaļā „Viens otram” – „Pievienot ierakstu” - „Ievietot vēlmi”, lietotājs aizpilda ievadformas laukus</p>	<ul style="list-style-type: none"> <li>• Nosaukums – obligāti aizpildāms lauks, to atstājot tukšu, tiks attēlots brīdinājuma paziņojums „Vēlmes nosaukums nevar būt tukšs”;</li> <li>• Apraksts – obligāti aizpildāms lauks, to atstājot tukšu, tiks attēlots brīdinājuma paziņojums „Apraksts nevar būt tukšs”. Apraksta simbolu skaits nedrīkst pārsniegt 1000 zīmes;</li> <li>• Joma– obligāti aizpildāms lauks, to atstājot tukšu, tiks attēlots brīdinājuma paziņojums „Joma nevar būt tukšs”;</li> <li>• Atrašanas vieta – obligāti aizpildāms lauks, to atstājot tukšu, tiks attēlots brīdinājuma paziņojums „Atrašanas vieta nevar būt tukšs”;</li> </ul>	+
----	--------	---	---	---

6.	3.2.9.	Atrodies sadaļā „Viens otram” – „Pievienot ierakstu” - „Ievietot vēlmi”, lietotājs, veiksmīgi aizpildījis ievadformas laukus, nospiež pogu „Pievienot”	Vēlme tiek pievienota visu aktīvu vēlmju sarakstam	+
7.	3.2.12.	Atrodies sadaļā "Viens otram", lietotājs kopējo piedāvājumu sarakstā, apskatot savu ierakstu, nospiež pogu "Labet"	Lietotājam atveras piedāvājuma labošanas forma ar iepriekš ievadītajiem datiem	+
8.	3.2.12.	Lietotājs labo piedāvājuma iepriekš ievadītos datus	Lauki: nosaukums, apraksts, joma, atrašanas vieta nedrīkst būt tukši. Lauki: bildes url, bilde ir tukši	+
9.	3.2.12.	Lietotājs, izlabojis piedāvājuma iepriekš ievadītos datus, nospiež pogu „Saglabāt”	Piedāvājuma informācija tiek saglabāta	+
10.	3.2.13.	Atrodies sadaļā "Viens otram", lietotājs kopējo vēlmju sarakstā, apskatot savu ierakstu, nospiež pogu "Labet".	Lietotājam atveras vēlmes labošanas forma ar iepriekš ievadītajiem datiem	+
11.	3.2.13.	Lietotājs labo vēlmes iepriekš ievadītos datus	Lauki: nosaukums, apraksts, joma, atrašanas vieta nedrīkst būt tukši	+
12.	3.2.13.	Lietotājs, izlabojis vēlmes iepriekš ievadītos datus, nospiež pogu „Saglabāt”	Vēlmes informācija tiek saglabāta	+
13.	3.2.14.	Atrodies sadaļā "Viens otram", lietotājs kopējo	Lietotājam tiek attēlots iznirstošais logs ar	+

		piedāvājumu vai vēlmju sarakstā, apskatot savu ierakstu, nospiež pogu „Dzēst”	vaicājumu „Vai tiešām dzēst šo ierakstu?”. Tāpat tiek attēlota iespēja apstiprināt dzēšanu vai atcelt.	
14.	3.2.14.	Lietotājs apstiprina savu izvēli iznirstošā loga vaicājumam „Vai tiešām dzēst šo ierakstu?” (nospiež „OK”)	Piedāvājums tiek dzēsts no datubāzes	+
15.	3.2.14.	Lietotājs atceļ savu izvēli iznirstošā loga vaicājumam vaicājumam „Vai tiešām dzēst šo ierakstu?” (nospiež „Atcelt”)	Piedāvājums netiek dzēsts, tiek attēlots visu aktīvo piedāvājumu sarakstā.	+

### 5.2.3. Papildfunkcionalitāte

Šajā iterācijā tika izstrādāta funkcionalitāte, kas atbilst 2.2.3., 2.2.4., 2.2.5., 2.2.6. , 2.2.16. nodaļās aprakstītajiem lietotājstāstiem, kuriem atbilst akceptēšanas scenāriji, kas aprakstīti atbilst 3.2.3., 3.2.4., 3.2.5., 3.2.6. , 3.2.16. nodaļās. Šie lietotājstāsti ir realizēti 4.2.3.1. un 4.2.3.2. nodaļās aprakstītajos moduļos – „offers” un „offer”.

Nr.	Akcept. scen.	Testa apraksts	Paredzētais rezultāts	Rezultāts
1.	3.2.3.	Atrodies sadaļā "Viens otram", lietotājs kopējo piedāvājumu sarakstā, apskatot konkrētu ierakstu, nospiež saiti „Bilde”	Tiek attēlota piedāvājuma ievietotā bilde	+
2.	3.2.4.	Atrodies sadaļā "Viens otram", lietotājs kopējo piedāvājumu vai vēlmju sarakstā, apskatot konkrētu ierakstu, nospiež saiti „Sazināties”	Bez lapas pārlādes iznirst iepriekš sagatavota vēstuļu sūtīšanas forma	+

3.	3.2.4.	Lietotājs aizpild vēstuļu sūtīšanas formu un nospiež pogu „Sūtīt”	Vēstules veiksmīgas nosūtīšanas gadījumā iznirstošais logs aizverās un lietotājam tiek attēlots visu aktīvo piedāvājumu vai, attiecīgi, vēlmju saraksts	+
4.	3.2.4.	Lietotājs aizver vēstuļu sūtīšanas formu, nospiežot „X”	Lietotājam tiek attēlots iepriekš atvērtais piedāvājums vai vēlme	+
5.	3.2.5.	Atrodies sadaļā "Viens otram", lietotājs kopējo piedāvājumu vai vēlmju sarakstā, apskatot konkrētu ierakstu, nospiež saiti „Kopīgas intereses”	Ja kopīgas intereses ir, lietotājam tiek attēlots saraksts ar piedāvājumiem, kas sakrīt pēc lietotāja un ieraksta autora interesēm, ka arī interešu tiešās sakritībās, attiecīgi ja kopīgas intewrese nav – parādās paziņojumi „Nav piedāvājumu”, vai arī „Nav kopīgu interešu”	+
6.	3.2.6.	Atrodies sadaļā "Viens otram", lietotājs kopējo piedāvājumu vai vēlmju sarakstā, apskatot konkrētu ierakstu, nospiež pogu „Patīk”	Bez lapas pārlades lietotājam attēlojas kopējas skaits, cik citiem cilvēkiem patīk šis ieraksts	+
7.	3.2.16.	Atrodies sadaļā "Viens otram" – „Mani ieraksti”, lietotājs savu piedāvājumu vai vēlmju sarakstā, apskatot	Ieraksts tiek izdzēsts no sadaļas "Viens otram" – „Mani ieraksti”, lietotāja piedāvājumu vai vēlmju	+

		konkrētu ierakstu, nospiež pogu „Pabeigts”	saraksta un tiek attelots profila sadaļā „Paveiktie labie darbi”	
--	--	---	--	--

### 5.3. Secinājumi

Kopumā testēšana noritēja veiksmīgi. Testēšanas laikā tika atrastas kļūdas, taču tās tika izlabotas tuvākajā laikā pēc pārbaudēm. Tās bija neuzmanības, validācijas kļūdas, paziņojumu neprecizitātes.

## 6. PROJEKTA ORGANIZĀCIJA

Pirms izstrādes uzsākšanas notika apmācības par izstrādē nepieciešamajām valodām, spējās (Agile) metodoloģiju, izstrādes parvaldības rīku Assembla, kā arī versiju pārvaldības rīku GIT.

Tīmekļa vietne tika izstrādāta, izmantojot spējās (Agile) metodoloģijas Scrum metodi, ievērojot kontrolsarakstu punktus, ka piemēram:

- komanda ir krosfunkcionāla - komandas biedriem nav nostiprināto atsevišķo lomu,
- Scrum meistars ir komandas biedrs,
- iterāciju plānošanā produkta īpašnieks un komandas biedri piedalās personīgi,
- katra iterācija ilga vienu mēnesi.

## 7. KVALITĀTES NODROŠINĀŠANA

Tīmekļa vietnes „Labo darbu darbnīca” moduļa „Viens otram” izstrādē tika ievēroti vairāki nosacījumi, kas paredzēti, lai nodrošinātu kvalitāti. Programmas kods tika rakstīts pēc viena stila. Katras metodes sākumā tika ievietoti komentāri par funkcijas pielietojumu. Iterāciju beigās paveiktais darbs tika atrādīts pasūtītājam. Moduļa „Viens otram” lietotārstāsti, akceptēšanas scenāriji un akcepttesti tika taisīti pirms darba uzsākšanas, izņemot tās funkcijas, kuras tika izplānotas pēc sistēmas izstrādes sākuma.

Jāpiemin arī programmatūras pārvaldības rīks Assembla, kuru izmantojot tika izplānotas trīs iterācijas – pamatfunkcionalitāte, funkcionalitāte un papildfunkcionalitāte. Iterācijām bija izveidoti lietotārstāsti, kurus bija iespējams pārvietot pa izstrādes posmiem. Katras nedēļas beigās notika testēšanas apskate, kad tika parbaudīti jau realizētie lietotārstāsti – to funkcionalitāte un programmas kods.

## 8. KONFIGURĀCIJAS PĀRVALDĪBA

Izstrādē tika pielietots versiju pārvaldības rīks GIT – tas neglabā neiedomājamu skaitu modificēto failu versijas, bet gan tikai izmaiņas tajos. Ir lieliski piemērots, lai sekotu līdzi izmaiņām failos un iegūtu pilnīgu vēsturi un kontroli par to - kad kas ko ir labojis, dzēsis, pārvietojis un kāpēc, ar iespēju atgriezties laikā, meklēt, apvienot izmaiņas dažādos laika posmos.

Tapat, veicot izmaiņas datubāzē, tika saglabāta rezerves kopija. Kā arī katrs izstrādātājs strādāja ar programmkodu uz sava lokālā datora, taču gala izmaiņas regulāri tika ievietotas uz FTP-serveri.

## 9. DARBIETILPĪBAS NOVĒRTĒJUMS

### 9.1. Sākotnēji aprēķināta darbietilpība

Pirms darba uzsākšanas, ņemot vērā savu pieredzi, tika aprēķināta paredzama darbietilpība moduļa „Viens otram” izstrādei.

Lietotārstāsta Nr	Dienu skaits
2.2.1.	3
2.2.2.	3
2.2.3.	10
2.2.4.	3
2.2.5.	1
2.2.6.	3
2.2.7.	3
2.2.8.	6
2.2.9.	6
2.2.10.	3
2.2.11.	3
2.2.12.	7
2.2.13.	7
2.2.14.	4
2.2.15.	4
2.2.16.	3

Aprēķinot paredzamo darbietilpību, izstrādē pavadīto kalendāro dienu skaits – 65, personmēnešu skaits – 3.45.

### 9.2. COCOMO kalkulators

Koda rindiņu skaits - 1701

Programmatūras izstrādes metode - Semi-Detached

Nepieciešamā uzticamība - Normal

Datu bāzes izmērs - Low

Produkta sarežģītība - Normal

Izpildes laika ierobežojumi - Normal  
Informācijas glabāšanas ierobežojumi - Normal  
Virtuālās mašīnas nepastāvība - Low  
Datu apgrozījuma laiks - Normal  
Analītiskās spējas - Normal  
Pieredze ar līdzīgu programmatūras izstrādi – Very low  
Programmētāju spējas - Low  
Programmētāja pieredze programmas darbināšanas vidē - Normal  
Pieredze konkrētajā programmēšanas valodā - Low  
Mūsdienīgas programmēšanas prakses izmantošana - Normal  
Programmatūras izstrādes rīku izmantojums - Normal  
Izstrādes grafika ierobežojumi – Normal

Aizpildot šos kritērijus, COCOMO kalkulators aprēķināja šādu rezultātu - 3.96 personmēneši.

### **9.3. Secinājumi**

Salīdzinot darbietilpības aprēķinus izstrādes sākuma – 3.45 personmēneši, un izstrādes beigās – 3.96 personmēneši, rezultāts ir apmieram vienāds, kas liecina par pareizu zināšanu un pieredzes novērtējumu.

## 10. PROGRAMMATŪRAS KODS

### 10.1. offers.php(Kontrolieris)

```
<?php

if (!defined('BASEPATH'))
    exit('No direct script access allowed');

class Offers extends MY_Controller {

    public function __construct() {
        parent::__construct();
        $this->load->view('static/header', array('title' =>
'Piedāvājumi'));
        $this->load->view('static/menu');
        $this->load->model('two/Offer');
        $this->load->model('user/Usr_mdl');
        $this->load->model('user/Usr_msg');
        $this->load->model('dynamic_mdl');
        $this->load->library('interest');
        $this->load->library('upload');
        $this->load->library('images');

//Atlasa visus aktīvus bārterus
    public function index() {
        $this->load->view('two/two_left_sidebar', array('active_page' =>
'offers'));
        $userdata = $this->session->userdata('user_sess');
        $sort_by = $this->input->post('sorting');
        if ($sort_by == 'new') {
            $all_gifts = $this->Offer->get_all_offers();
        } else if ($sort_by == 'old') {
            $all_gifts = $this->Offer->get_offers_asc();
        } else if ($sort_by == 'interest') {
            $all_gifts = $this->Offer->get_all_offers();
            $all_gifts = $this->interest->show_by_interest($all_gifts,
$userdata['id']);
        } else {
            $all_gifts = $this->Offer->get_all_offers();
            $all_gifts = $this->interest->show by interest($all_gifts,
$userdata['id']);
        }
        //vēstules sūtīšanas no piedāvājuma
        $msg = $this->input->post('msg');
        if ($msg == 'sent') {
            # Savāc datus, ko sūtīs
            $data = array(
                'title' => $this->input->post('title'),
                'msg' => $this->input->post('text'),
                'sender_ID' => $userdata['id']
            );
            $receiver = array($this->input->post('ID'));
            # Nosūta ziņu
            $this->Usr_msg->send_msg($receiver, $data, $this->input-
>post('reply_ID'));
        }

        if (is_array($all_gifts))
            foreach ($all_gifts as $gift) {
```

```

        $user = $this->Usr_mdl->get_user_data($gift->owner_ID);
        $gift->mpic = $user->mpic;
        $gift->avatar = $user->avatar;
        $gift->user_ID = $gift->owner_ID;
        $gift->likes = $this->dynamic_mdl-
>checkLike($userdata['id'], 'O', $gift->offer_ID);
        $gift->picture = Images::GetImage($gift->picture, $gift-
>pic_type);
    }
    $this->load->view('two/gifts', array('all_gifts' => $all_gifts,
'viewer_ID' => $userdata['id'], 'sort_by' => $sort_by));
    $this->load->view('user/mission_menu');
    $this->load->view('two/two_right_sidebar');
    $this->load->view('static/footer');
}

//Atlasa visus manus aktīvus piedāvājumus - gan bārterus, gan dāvinājumus
public function getmyoffers() {
    $this->load->view('two/two_left_sidebar', array('active_page' =>
'getmyoffers', 'big_menu' => 'myoffers'));
    $all_gifts = $this->Offer->get_my_offers();
    $userdata = $this->get_userdata('id, mpic');
    $user_pic = $userdata['mpic'];
    //Šķirošana
    $sort_by = $this->input->post('sorting');
    if ($sort_by == 'new') {
        $all_gifts = $this->Offer->get_my_offers();
    } else if ($sort_by == 'old') {
        $all_gifts = $this->Offer->get_my_offers_asc();
    } else {
        $all_gifts = $this->Offer->get_my_offers();
    }
    // Like-poga
    if (is_array($all_gifts))
        foreach ($all_gifts as $gifts) {
            $gifts->likes = $this->dynamic_mdl-
>checkLike($userdata['id'], 'O', $gifts->offer_ID);
        }
    $this->load->view('two/getmyoffers', array('getmyoffers' =>
$all_gifts, 'user_pic' => $user_pic, 'sort_by' => $sort_by));
    $this->load->view('user/mission_menu');
    $this->load->view('two/two_right_sidebar');
    $this->load->view('static/footer');
}

//Dzēst savu piedāvājumu
public function deleteoffer() {
    $this->load->model('two/offer');
    $offer_ID = $this->input->post('offer_ID');
    $_user_data = $this->session->userdata('user_sess');
    $user_id = $_user_data['id'];
    $this->offer->delete_offer($offer_ID);
    $this->load->model('user/Usr_wall');
    $this->Usr_wall->delete_new('ba', $offer_ID);
    $this->interest->delete_all_offer_interests($offer_ID);
    redirect("/two/offers/getmyoffers");
}

//Pievienot piedāvājumu
public function addoffer() {
    $this->load->model('two/offer');
    $_user_data = $this->session->userdata('user_sess');

```

```

        $owner_ID = $_user_data['id'];
        //Ievadlauku pārbaude
        $this->load->library('form_validation');
        $this->form_validation->set_rules('title', "Piedāvājuma nosaukums",
'required');
        $this->form_validation->set_rules('text', "Apraksts", 'required');
        $this->form_validation->set_rules('scope', "Joma", 'required');
        $this->form_validation->set_rules('location', "Atrašanas vieta",
'required');
        $this->form_validation->set_message('required', '%s nevar būt
tukšs');
        $this->form_validation->set_error_delimiters('<div
class="er_mes">', '</div>');
        $title = $this->input->post('title');
        $description = $this->input->post('text');
        $scope = $this->input->post('scope');
        $location = $this->input->post('location');
        $pic_url = $this->input->post('pic_url');
        $gift = $this->input->post('offertype');
        $data['offers']->title = $title;
        $data['offers']->description = $description;
        $data['offers']->location = $location;
        $data['offers']->pic_url = $pic_url;
        $data['offers']->scope = $this->input->post("scope");
        $data['offers']->title = '';

        $ImageResult = $this->images->LoadImageOrURL('pic',
trim($pic_url));
        if ($this->form_validation->run() == FALSE || !$this->input-
>post('addappeal') || ($ImageResult->is_upload == TRUE && $ImageResult-
>success == FALSE)) {

            //iepriekš ievadīta informācija
            $data['title'] = $title;
            $data['text'] = $description;
            $data['offertype'] = $gift;
            $data['scope'] = $scope;
            $data['location'] = $location;
            $data['pic_url'] = $pic_url;
            $data['upload_error'] = '';
            if (!empty($ImageResult->error)) {
                $data['upload_error'] = '<div class="er_mes">' .
$ImageResult->error . '</div>';
            }
            $this->images->CancelUpload($ImageResult);
            //Kreisā sāna izvēle
            $this->load->view('two/two_left_sidebar');
            //Skati
            $this->load->view('two/offerinput', $data);
            $this->load->view('user/mission_menu');
            $this->load->view('two/two_right_sidebar', array('active_page'
=> '', 'big_menu' => 'choise'));
            $this->load->view('static/footer');
        } else {
            $title = trim($title);
            $description = trim($description);
            $location = trim($location);
            $pic_url = trim($pic_url);
            $scope = trim($scope);
            if ($ImageResult->is_upload == TRUE && $ImageResult->success ==
TRUE) {
                $this->images->DoImageResize($ImageResult->image);
            }

```

```

        $this->offer->save_offer($title, $owner_ID, $description,
        $location, $ImageResult->image, $ImageResult->type, $scope, $gift);
        //izsauc saglabāšanu
        $this->interest->add_offer_interests($scope, $this->interest-
        >get_last_id('offers_table', true));
        redirect("/two/offers"); ///Uz sadalu kurā tika pievienots
    }
}

//Atzīmēt ka pabeigtu piedāvājumu
public function completed_offer() {
    $this->load->model('two/offer');
    $offer_ID = $_POST["offer_ID"];
    $user_data = $this->session->userdata('user_sess');
    $user_id = $user_data['id'];
    $getmyoffers = $this->offer->get_my_offers($offer_ID);
    $this->load->model('user/Usr_wall');
    $this->Offer->completed_offer($offer_ID); //izsauc saglabāšanu
    redirect("/two/offers/getmyoffers"); ///Uz sadalu kuru tika
    pievienota
}

//Attēlo bildes
public function offerpictures($offerID) {
    $all_pics = $this->Offer->get_offer_pic($offerID);
    if (is_array($all_pics))
        foreach ($all_pics as $pics) {
            $pics->picture = Images::GetImage($pics->picture, $pics-
            >pic_type);
        }
    $this->load->view('two/two_left_sidebar');
    $this->load->view('two/picture', array('all_pics' => $all_pics));
    $this->load->view('user/mission_menu');
    $this->load->view('two/two_right_sidebar');
    $this->load->view('static/footer');
}
}
}

```

## 10.2.offer.php(Modelis)

```

<?php
class Offer extends MY_Model {
    private $offer_pictures_table = 'offer_pictures';
    function __construct() {
        parent::__construct();
    }

    //Attēlo visus piedāvājumus - vecākie sākumā
    public function get_offers_asc() {
        return $this->db->select(self::offers_table . '.offer_ID, title,
        description, owner_ID, gift, location, scope, type, ' . $this-
        >offer_pictures_table . '.picture, pic_type')->from(self::offers_table)-
        >join($this->offer_pictures_table, $this->offer_pictures_table . '.offer_ID
        = ' . self::offers_table . '.offer_ID', 'left')->where('complete', 1)-
        >where('type', 0)->order_by('offer_ID', 'asc')->get()->result();
    }

    //Attēlo manus piedāvājumus
    public function get_my_offers() {

```

```

        $_user_data = $this->session->userdata('user_sess');
        $_id = $_user_data['id'];
        return $this->db->select(self::offers_table . '.offer_ID, title,
description, location, scope, type')->from(self::offers_table)-
>where('complete', 1)->where('type', 0)->where('owner_ID', $_id)-
>order_by('offer_ID', 'desc')->get()->result();
    }

//Saglabāt vēlmi
    public function save_offerinterest($title, $owner_ID, $description,
$location, $scope) {
        $data = array(
            'title' => $title,
            'description' => $description,
            'owner_ID' => $owner_ID,
            'location' => $location,
            'scope' => $scope,
            'complete' => 1,
            'type' => 1
        );
        $this->db->insert(self::offers_table, $data);
        $this->id = $this->db->insert_id();
        return $this;
    }

//Atzīmēt piedāvājumu kā pabeigtu
    public function completed_offer($offer_ID) {
        $data = array(
            'complete' => 0
        );
        $this->db->where('offer_ID', $offer_ID);
        $this->db->update('offers', $data);

        return $this;
    }

//Piedāvājuma labošana(atjaunina datus)
    public function update_offer($offer_ID, $title, $owner_ID,
$description, $location, $pic_url, $scope, $gift) {
        $data = array(
            'title' => $title,
            'description' => $description,
            'location' => $location,
            'pic_url' => $pic_url,
            'scope' => $scope,
            'gift' => $gift
        );
        $this->db->where('offer_ID', $offer_ID)->where('owner_ID',
$owner_ID);
        $this->db->update('offers', $data);
        return $this;
    }
}

```

## 11. NOBEIGUMS

Izstrādājot tīmekļa vietnes „Labo darbu darbnīca” moduli „Viens otram” tika realizēta iecerētā funkcionalitāte, taču darbu pie projekta var turpināt un attīstīt to arī turpmāk, uzlabojot vizuālo izskatu un pieliekot klāt vēl kādas nepieciešamas funkcijas, kas arī tiks darīts.

Strādājot pie moduļa „Viens otram” izstrādes, tika apgūtas un uzlabotas PHP, CSS valodas zināšanas, kā arī iegūta lielaka pieredze darbā ar MVC (Model –View –Controller) strukturēšanas principiem balstīto „CodeIgniter” palīgfunkciju bibliotēku.

## 12. IZMANTOTĀ LITERATŪRA UN AVOTI

1. <http://www.assembla.com/>
2. <http://www.agile-latvia.org/>
3. <http://laacz.lv/2011/08/12/laiks-versiju-kontrolei-pirmas-asis/>
4. <http://lv.wikipedia.org/wiki/MVC>
5. [scum.org.ua/scum-i-xp-zametki-s-peredovoj/](http://scum.org.ua/scum-i-xp-zametki-s-peredovoj/)
6. <http://stackoverflow.com/questions/56546/activerecord-as-model-is-this-a-good-idea>
7. [www.tinyurl.com/agilerussiafb](http://www.tinyurl.com/agilerussiafb)

## DOKUMENTĀRĀ LAPA

Kvalifikācijas darbs „Tīmekļa vietne „**Labo darbu darbnīca**”. Modulis – „**Viens otram**” izstrādāts Latvijas Universitātes Datorikas fakultātē.

Ar savu parakstu apliecinu, ka darbs izstrādāts patstāvīgi, izmantoti tikai tajā norādītie informācijas avoti un iesniegtā darba elektroniskā kopija atbilst izdrukai.

Autors: *Katrīna Ladutjko* \_\_\_\_\_ .05.2012.

Rekomendēju darbu aizstāvēšanai

Darba vadītājs: *M.Dat. Jānis Mārtužs* \_\_\_\_\_ .05.2012.

Recenzents: *M.dat. Gobleja Kārlis*

Darbs iesniegts 24.05.2012.

Kvalifikācijas darbu pārbaudījumu komisijas sekretārs: *Dainis Dosbergs* \_\_\_\_\_

Darbs aizstāvēts kvalifikācijas darbu pārbaudījuma komisijas sēdē

\_\_\_\_. 06.2012. prot. Nr. \_\_\_\_\_, vērtējums \_\_\_\_\_ (\_\_\_\_\_)

Komisijas sekretārs(-e): \_\_\_\_\_