

LATVIJAS UNIVERSITĀTE
DATORIKAS FAKULTĀTE

Drupal modulis „Laika ziņas”

Kvalifikācijas darbs

Autors: **Krista Puķe**

Stud.apl. kp09044

Darba vadītājs: B.dat Georgijs Samoilenko

Rīga 2011

Anotācija

DRUPAL modulis „Laika ziņas” ir sistēma, kura ļauj lietotājiem ērti un vienkārši aplūkot laika apstākļu prognozi un attiecīgi arī izvēlēties pilsētu, no datubāzes, par kuru iegūt informāciju. Sistēmu ērti lietot un konfigurēt pēc saviem ieskatiem un vajadzībām ir iespēja administratoram, kurš var mainīt „Laika ziņu” bloka izskatu un izvadāmo informāciju, kā arī pievienot un dzēst pilsētas pēc saviem ieskatiem.

Abstract

DRUPAL module „Weather forecast” is a system, which enables user to view weather forecast in an easy and convenient way by choosing a particular city from the database. System can be easily configured by the administrator according to necessities, it is possible to change appearance and information showed in the „Weather forecast” block, as well as adding and removing different cities function is available.

SATURS

Apzīmējumu saraksts.....	1
1. Ievads.....	2
2. Programmatūras prasību specifikācija.....	3
2.1 Ievads.....	3
2.1.1 Nolūks.....	3
2.1.2 Darbības sfēra.....	3
2.1.3. Saistība ar citiem dokumentiem.....	3
2.2. Vispārējais apraksts.....	3
2.2.1. Produkta lietojamība.....	3
2.2.2. Produkta funkcijas.....	3
2.3 Funkcionālās prasības.....	5
2.3.1 Pilsētas nosaukuma pārbaude.....	5
2.3.2 WOEID pārbaude.....	5
2.3.3 Valsts nosaukuma pārbaude.....	5
2.3.4 Informācijas iegūšana no xml.....	6
2.3.5 Iegūtās informācijas ielikšana datubāzē.....	6
2.3.6 Datuma ģenerēšana.....	7
2.3.7 Dzēst pilsētu.....	7
2.3.8 Pašreizējās laika ziņas.....	7
2.3.9 Šodienas laika ziņas.....	8
2.3.10 Rītdienas laika ziņas.....	8
2.3.11 Atjaunošanās.....	9
2.3.12 Apskatīt pilsētu.....	9
2.4 Nefunkcionālās prasības.....	10
2.4.1 Datu drošība.....	10
2.4.2 Pieejamība.....	10
3. Programmatūras projektējuma apraksts.....	11

3.1 Ievads.....	11
3.1.1 Nolūks.....	11
3.1.2 Darbības sfēra	11
3.1.3 Saistība ar citiem dokumentiem	11
3.2 Dekompozīcijas apraksts	11
3.2.1 Moduļu dekompozīcija	11
3.2.2 Datu dekompozīcija.....	12
3.3 Atkarību apraksts.....	12
3.3.1 Starpprocesu atkarības.....	12
3.3.2 Datu atkarības	15
3.4 Datu detalizēts projektējums	15
3.4.1 weather_city_weather	15
3.4.2 weather_city_codes	17
3.5 Ārējā saskarne.....	17
3.5.1 Lietotāju saskarne	17
3.5.2 Skati	17
4. Testēšanas dokumentācija	22
4.1 Testēšanas plāns	22
4.2 Testēšanas žurnāls	22
4.2.1 Pilsētas pievienošanas modulis.....	22
4.2.2 Pilsētas izdzēšanas modulis	23
4.2.3 Laika ziņu konfigurācijas modulis	23
4.2.4 Pilsētas apskatīšanas modulis	25
4.3 Testēšanas rezultātu kopsavilkums.....	25
5. Programmatūras projekta organizācija	26
6. Kvalitātes nodrošināšana	27
7. Konfigurāciju pārvaldība.....	28
8. Darbietilpības novērtējums.....	29

9. Izmantotā literatūra un avoti.....	31
Pielikumi.....	32
1.pielikums. Koda fragments no weather.module faila	32
2.pielikums. Koda fragments no weather-content.tpl faila.....	41

APZĪMĒJUMU SARAKSTS

Melnās kastes princips	Funkcionālā testēšana ir melnās kastes testēšana. Tiek testētas funkcijas ar ieejas datiem un pārbaudīts izejas rezultāts
DRUPAL	Drupal ir modulārs bezmaksas atvērta koda ietvars un saturs vadības sistēma (SVS), izveidota PHP programmēšanas valodā
modulis	Atsevišķa identificējama programmas daļa, kuru var autonomi izveidot un izmantot, lai atvieglotu programmu sastādīšanu
WOEID	Unikāla vietas identifikācija, kura nemainās un neatjaunojās
xml	Valoda XML nodrošina globālā tīmekļa izstrādātājiem iespēju radīt savas personiskās birkas (kodus), lai nodrošinātu tādu funkciju izpildi, ko nevar nodrošināt ar valodas HTML starpniecību
.tpl.php	Šī sagatave nosaka moduļa bloka karkasu
css	Cascading Style Sheets – valoda, ko izmanto esošās informācijas vizuālajai noformēšanai, definējot konkrēta elementa vai noteikta tipa elementu izskatu
MySQL	Datu bāzu vadības sistēma
ER diagramma	ER diagrammas ir datu bāzu modelēšanas metode, kura tiek izmantota lai izveidotu datu bāzes shēmu vai datu modeli
SLOCCount	Darbietilpības noteikšanas, mērīšanas metode
COCOMO	Darbietilpības noteikšanas, mērīšanas metode
dd	Mēneša dienas skaitlis, piemēram, 1 jūnijs - 01
mm	Mēnesis, piemēram, jūnijs - 06
HTTP	Hyper Text Transfer Protocol, galvenais informācijas pārraides protokols globālajā tīmeklī
HTML	Hyper Text Markup Language – iezīmēšanas valoda, ko izmanto teksta un citu datu attēlošanai Interneta pārlūkprogrammā
PPA	Programmatūras projektējuma apraksts
PPS	Programmatūras prasību specifikācija

1. IEVADS

Mūsdienās, cilvēku dzīves temps kļūst arvien raitāks, tādēļ rodas nepieciešamība arvien ātrāk, ērtāk un efektīvāk iegūt nepieciešamo informāciju.

Lai atvieglotu cilvēku ikdienišķo darbu, meklējot laika ziņas caur dažnedažādiem ziņu portāliem, kuri veidoti gan Latvijā, gan ārvalstīs, izveidoju Drupal moduli „Laika ziņas”. Izveidoju laika ziņu moduli, kuru katrs pēc ieskatiem varēs pievienot nepieciešamās pilsētas un novietot savā vietnē ērtākajā apskates vietā. Apmeklētāji, lietotāji un klienti var brīvi apskatīt savas pilsētas laika prognozi, kā arī dodoties braucienos pie klientiem vai savās vajadzībās zinātu gaidāmo laiku šodienai un rītdienai.

Manis izveidotā moduļa pamatprasības: lai apmeklētājam, lietotājam un klientam būtu ērti, ātri un vienkārši apskatīt laika ziņas un iegūt sev nepieciešamo informāciju, pilnvērtīgai dienai, pārrunām, izbraukumiem apmeklētajā vietnē, kā arī lai administratoram būtu iespēja izkārtot laika ziņu moduli pēc savām nepieciešamībām un ieskatiem.

2. PROGRAMMATŪRAS PRASĪBU SPECIFIKĀCIJA

2.1 Ievads

2.1.1 Nolūks

Dokuments satur DRUPAL moduļa „Laika ziņas” funkcionālo un nefunkcionālo prasību apkopojumu. Dokumenta mērķis ir aprakstīt programmatūras prasības, kas kalpos kā pamats programmaprodukta projektējumam.

Izstādāmajā sistēmā realizētas tiks prasības, kuras aprakstītas šajā dokumentā.

2.1.2 Darbības sfēra

Laika ziņu modulis paredzēts, lai ikviens, kurš apmeklēs attiecīgo vietni uz, kuras šis modulis ir uzstādīts, varētu ērti, ātri un vienkārši apskatīt sev nepieciešamo informāciju par laika ziņām.

Lai apskatītu laika ziņas nebūs jāreģistrējas vietnē, jo šīs ziņas ir publiski pieejamas jebkura veida lietotājam.

2.1.3. Saistība ar citiem dokumentiem

Dokuments „Programmatūras prasību specifikācija” tiks ņemts kā pamats izstrādājot tādus dokumentus kā: „Programmatūras projektējuma apraksts” un „Testēšanas dokumentācija”

2.2. Vispārējais apraksts

2.2.1. Produkta lietojamība

Kā neregistrētiem, tā reģistrētiem lietotājiem ir nodrošināta iespēja vietnē, kurā pievienots „Laika ziņu” modulis, apskatīt laika ziņas un izvēlēties pilsētu, no saraksta, ko aplūkot.

2.2.2. Produkta funkcijas

Lai manis veidotā sistēma varētu strādāt pilnvērtīgi, tā sastāv no vairākiem loģiskiem moduļiem.

Sistēmas moduļi:

- Pilsētas pievienošanas modulis,
- Pilsētas izdzēšanas modulis,
- Laika ziņu konfigurācijas modulis,

- Pilsētas apskatīšanas modulis.

2.2.2.1 Pilsētas pievienošanas modulis

- Pilsētas nosaukuma pārbaude – jābūt pārbaudei, kura nosaka vai pilsētas nosaukums norādīts korekts;
- WOEID pārbaude - jāpārbauda vai norādītais kods jau neeksistē datubāzē un vai WOEID kods no xml datnes sniedz informācija;
- Valsts nosaukuma pārbaude – pārbauda vai valsts nosaukums ir norādīts korekti;
- Informācijas iegūšana no xml – jāiegūst informācija par jaunizveidoto pilsētu no xml datnes;
- Iegūtās informācijas ielikšana datubāzē – iegūtā informācija no xml datnes un lietotāja ievadītā informācija jāievada datubāzē.
- Datuma ģenerēšana – funkcija izveido šodienas un rītdienas datumu, kā arī atjaunošanās laiku.

2.2.2.2 Pilsētas izdzēšanas modulis

- Izdzēst pilsētu – jāpārbauda vai rūtiņa ir ieķeksēta un jābūt iespējai dzēst kādu no pilsētām, kas atrodas datubāzē.

2.2.2.3 Laika ziņu konfigurācijas modulis

- Pašreizējie laika apstākļi – jāpārbauda vai konfigurācijās norādīts rādīt pašreizējās laika ziņas un jāapskatās kādas tieši ziņas ir jāuzrāda blokā;
- Šodienas laika ziņas - jāpārbauda vai konfigurācijās norādīts rādīt šodienas laika ziņas un jāapskatās kādas tieši ziņas ir jāuzrāda blokā;
- Rītdienas laika ziņas - jāpārbauda vai konfigurācijās norādīts rādīt rītdienas laika ziņas un jāapskatās kādas tieši ziņas ir jāuzrāda blokā;
- Atjaunošanās – jāpārbauda vai konfigurācijās ir norādīts rādīt atjaunošanās laiku un jāizvada blokā.

2.2.2.4 Pilsētas apskatīšanas modulis

- Apskatīt pilsētu – jābūt iespējai, jebkuram lietotājam, izvēlēties pilsētu no datubāzes, kuras laika prognozi var aplūkot.

2.3 Funkcionālās prasības

2.3.1 Pilsētas nosaukuma pārbaude

Mērķis:

Funkcijas mērķis ir pārbaudīt vai pilsētas nosaukums ir ievadīts korekti

Ievaddati:

- Pilsētas nosaukums

Apstrāde:

- Pārbauda vai pilsēta ar šādu nosaukumu jau neeksistē datubāzē;
- Vai pilsētas nosaukums ierakstīts korekti, ar burtiem;
- Vai pilsētas nosaukums nav pārāk īss, nesastāv no trim vai mazāk burtiem;

Izvaddati:

Nav.

Kļūdu paziņojumi:

1. Pilsēta ar šādu nosaukumu jau eksistē.
2. Pilsētas vārdam jāsaturs burti.
3. Pilsētas nosaukumam jābūt garākam.

2.3.2 WOEID pārbaude

Mērķis:

Mērķis ir pārbaudīt vai WOEID kods ir derīgs, lai iegūtu informāciju no xml un vai nedublējās ar kādu jau esošu datubāzē

Ievaddati:

- WOEID kods

Apstrāde:

- Ievieto kodu saitē atbilstošā vietā un pārbauda vai šī saite atbild ar pilnvērtīgu informāciju
- Pārbauda vai šāds kods jau nepastāv datubāzē
- Pārbauda vai WOEID kods sastāv no cipariem

Izvaddati:

Nav.

Kļūdu paziņojumi:

1. Šis WOEID neatbild ar prognozi.
2. Pilsēta ar šādu WOEID jau eksistē.
3. WOEID jāsaturs skaitļi.

2.3.3 Valsts nosaukuma pārbaude

Mērķis:

Šīs funkcijas mērķis ir pārbaudīt vai valsts nosaukums ir norādīts korekts un satur vairāk kā trīs burtus

Ievaddati:

- Valsts nosaukums

Apstrāde:

- Pārbauda vai ievadītais valsts nosaukums sastāv no burtiem
- Pārbauda burtu skaitu, jābūt lielākam par trim.

Izvaddati:

Nav.

Kļūdu paziņojumi:

1. Valsts nosaukumam jāsaturs burti.
2. Valsts nosaukumam jābūt garākam.

2.3.4 Informācijas iegūšana no xml

Mērķis:

Funkcijas mērķis ir iegūt jaunāko informāciju, no xml datnes izmantojot WOEID (city_id)

Ievaddati:

- WOEID (city_id)

Apstrāde:

- Iegūto WOEID ieliek norādītajā saitē
- No iegūtā WOEID iegūst nepieciešamos(norādītos) datus: vēja virziens, vēja ātrums, saulriets, saullēkts, pašreizējais kods, temperatūra, šodienas augstākā, zemākā temperatūra un kods, rītdienas augstākā, zemākā temperatūra un kods, atjaunošanās datums un laiks

Izvaddati:

Nav.

Kļūdu paziņojumi:

Nav.

2.3.5 Iegūtās informācijas ielikšana datubāzē

Mērķis:

Mērķis ir jauniegtā informāciju ievietot datubāzē

Ievaddati:

Nav.

Apstrāde:

- Datus, kuri tika iegūti no xml un papildus vēl šodienas un rītdienas datumu ieliek datubāzē attiecīgajā tabulā.

Izvaddati:

Ja pilsēta ir pievienota izvada – „Pilsēta sekmīgi pievienota.”

Kļūdu paziņojumi:

Nav.

2.3.6 Datuma ģenerēšana

Mērķis:

Funkcijas mērķis ir izveidot šodienas, rītdienas datumu un informācijas atjaunošanās laiku

Ievaddati:

Nav.

Apstrāde:

- Izveido šodienas datumu
- Izveido rītdienas datumu
- Izveido atjaunošanās laiku

Izvaddati:

Nav.

Kļūdu paziņojumi:

Nav.

2.3.7 Dzēst pilsētu

Mērķis:

Funkcijas mērķis ir pārbaudīt vai izvēles rūtiņa ir ieķeksēta un izdzēst pilsētu

Ievaddati:

- Ieķeksēta izvēles rūtiņa

Apstrāde:

- Pārbauda vai izvēles rūtiņa ir ieķeksēta
- Lietotājs norādot pilsētas nosaukumu, sistēma tam piesaistīto WOEID(city_id) izmanto lai no divām datubāzes tabulām izdzēstu informāciju, kura attiecas uz šo pilsētu.

Izvaddati:

1. „Pilsēta tika izdzēsta”

Kļūdu paziņojumi:

Ja vēlaties izdzēst pilsētu, tad jums ir nepieciešams uzklikšķināt uz izvēles rūtiņas.

2.3.8 Pašreizējās laika ziņas

Mērķis:

Funkcijas mērķis ir konfigurēt un izvadīt blokā pašreizējās laika ziņas

Ievaddati:

- Ķeksīši izvēles rūtiņās

Apstrāde:

- Pārbauda vai ieķeksēta ir izvēles rūtiņa ar nosaukumu „Rādīt pašreizējos laika apstākļus”, ja tā nav ieķeksēta tad blokā neparādās informācija par pašreizējo laiku;
- Pārbauda, kuras izvēles rūtiņas ir ieķeksētas un pielāgojoties izvēlētajai informācijai, padod mainīgos .tpl.php datnei un tālāk css datnei

Izvaddati:

1. „Izmaiņas tika saglabātas”

Izvada bloku ar attiecīgi pielāgoto skatu, kurš satur informāciju par pašreizējajiem laika apstākļiem.

Kļūdu paziņojumi:

Nav.

2.3.9 Šodienas laika ziņas

Mērķis:

Šīs funkcijas mērķis ir konfigurēt un izvadīt blokā šodienas laika ziņas

Ievaddati:

- Ķeksīši izvēles rūtiņās

Apstrāde:

- Pārbauda vai ieķeksēta ir izvēles rūtiņa ar nosaukumu „Rādīt šodienas laika prognozi”, ja tā nav ieķeksēta tad blokā neparādās informācija par šodienas laika apstākļiem;
- Pārbauda, kuras izvēles rūtiņas ir ieķeksētas un pielāgojoties izvēlētajai informācijai, padod mainīgos .tpl.php datnei un tālāk css datnei.

Izvaddati:

1. „Izmaiņas tika saglabātas”

Izvada bloku ar attiecīgi pielāgoto skatu, kurš satur informāciju par šodienas laika prognozi.

Kļūdu paziņojumi:

Nav.

2.3.10 Rītdienas laika ziņas

Mērķis:

Mērķis ir konfigurēt un izvadīt blokā rītdienas laika ziņas

Ievaddati:

- Ķeksīši izvēles rūtiņās

Apstrāde:

- Pārbauda vai ieķeksēta ir izvēles rūtiņa ar nosaukumu „Rādīt rītdienas laika prognozi”, ja tā nav ieķeksēta tad blokā neparādās informācija par rītdienas laika apstākļiem;
- Pārbauda, kuras izvēles rūtiņas ir ieķeksētas un pielāgojoties izvēlētajai informācijai, padod mainīgos .tpl.php datnei un tālāk css datnei.

Izvaddati:

1. „Izmaiņas tika saglabātas”

Izvada bloku ar attiecīgi pielāgoto skatu, kurš satur informāciju par rītdienas laika prognozi.

Kļūdu paziņojumi:

Nav.

2.3.11 Atjaunošanās

Mērķis:

Funkcijas mērķis ir rādīt atjaunošanās datumu un laiku, ja tas norādīts konfigurācijā

Ievaddati:

- Ķeksītis izvēles rūtiņā

Apstrāde:

- Ja izvēlnes rūtiņā ir ķeksītis tad funkcija padod nepieciešamos mainīgos uz tpl.php datni un css datni.

Izvaddati:

1. „Izmaiņas tika saglabātas”

Izvada bloku ar attiecīgi pielāgoto skatu, kurš satur informāciju par atjaunošanās laiku un datumu.

Kļūdu paziņojumi:

Nav.

2.3.12 Apskatīt pilsētu

Mērķis:

Funkcijas mērķis ir lai jebkurš lietotājs varētu izvēlēties pilsētu, no datubāzē esošajām, kuras laika prognozi viņš vēlas aplūkot.

Ievaddati:

- Pilsētas nosaukums

Apstrāde:

- Pēc norādītās pilsētas nosaukuma piesaistītā WOEID(city_id), tiek paņemta informācija no datubāzes tabulas, kura visu nepieciešamo informāciju par pilsētu

Izvaddati:

Izvada bloku ar attiecīgajiem skatiem, par norādīto pilsētu.

Kļūdu paziņojumi:

Nav.

2.4 Nefunkcionālās prasības

2.4.1 Datu drošība

Jebkurš lietotājs var aplūkot to informāciju, kura izvadīta laika ziņu blokā un var mainīt pilsētas, iegūstot sev nepieciešamo informāciju par laika prognozi attiecīgajā vietā.

Vietnes administratoram ir tiesības mainīt laika ziņu bloka izskatu un var konfigurēt laika ziņu izvadāmo informāciju pēc saviem ieskatiem un prasībām.

2.4.2 Pieejamība

Laika ziņu modulim vietnē jābūt pieejamam 24 stundas dienā, 7 dienas nedēļā. Modulis pieejams jebkāda veida lietotājam.

3.PROGRAMMATŪRAS PROJEKTĒJUMA APRAKSTS

3.1 Ievads

3.1.1 Nolūks

Programmatūras projektējuma apraksta nolūks, manis veidotajai sistēmai DRUPAL modulis „Laika ziņas” , ir aprakstīt kā prasības, kuras definētas programmatūras projektējuma specifikācijā, tiks realizētas un atvieglot sistēmas izstrādi.

3.1.2 Darbības sfēra

Manis veidotais laika ziņas ir paredzētas, lai ikviens interneta lietotājs aplūkojot vietni, kurai pievienots modulis, varētu ērti un brīvi apskatīt laika prognozi pievienotajām pilsētām.

3.1.3 Saistība ar citiem dokumentiem

Programmatūras projektējuma apraksts tika izstrādāts balstoties uz programmatūras prasību specifikāciju.

3.2 Dekompozīcijas apraksts

3.2.1 Moduļu dekompozīcija

3.2.1.1 Pilsētas pievienošanas modulis

Nolūks: Funkcija paredzēta lai varētu pievienot pilsētas, pārbaudītu vai informācija ir norādīta korekti un pievienotu informāciju datubāzei.

Funkcijas:

1. Pilsētas nosaukuma pārbaude
2. WOEID pārbaude
3. Valsts nosaukuma pārbaude
4. Informācijas iegūšana no xml
5. Iegūtās informācijas ielikšana datubāzē
6. Datuma ģenerēšana

3.2.1.2 Pilsētas izdzēšanas modulis

Nolūks: Izdzēst pilsētu no datubāzes.

Funkcijas:

1. Izdzēst pilsētu

3.2.1.3 Laika ziņu konfigurācijas modulis

Nolūks: Nodrošina administratoram, iespējas konfigurēt pēc savām vajadzībām, ieskatiem laika ziņu moduļa bloku.

Funkcijas:

1. Pašreizējie laika apstākļi
2. Šodienas laika ziņas
3. Rītdienas laika ziņas
4. Atjaunošanās

3.2.1.3 Pilsētas apskatīšanas modulis

Nolūks: Nodrošina iespēju aplūkot laika ziņas kādai konkrētai pilsētai no datubāzes, mainīt pilsētas.

Funkcijas:

1. Apskatīt pilsētu

3.2.2 Datu dekompozīcija

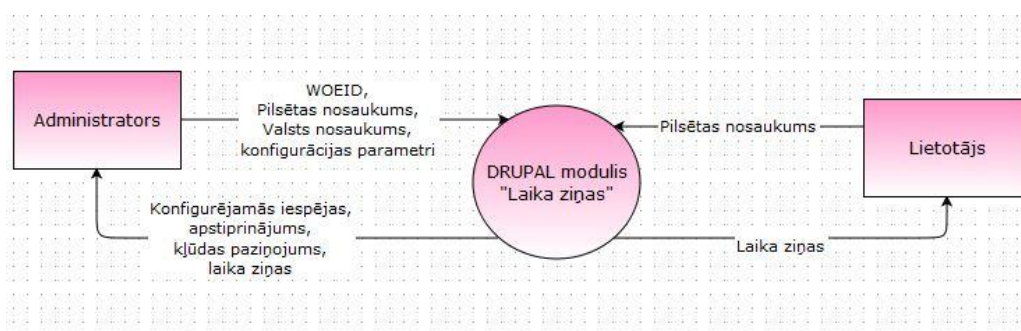
Sistēma datus glabā vienā MySQL datubāzē. Datubāze sastāv no 2 tabulām:

1. Tabula „weather_city_weather” - glabājas detalizēta informācija par pilsētām un tām piesaistītajām laika ziņām.
2. Tabula „weather_city_codes” – tabulā glabājās pilsētu nosaukumi, WOEID kodi un valsts nosaukumi.

3.3 Atkarību apraksts

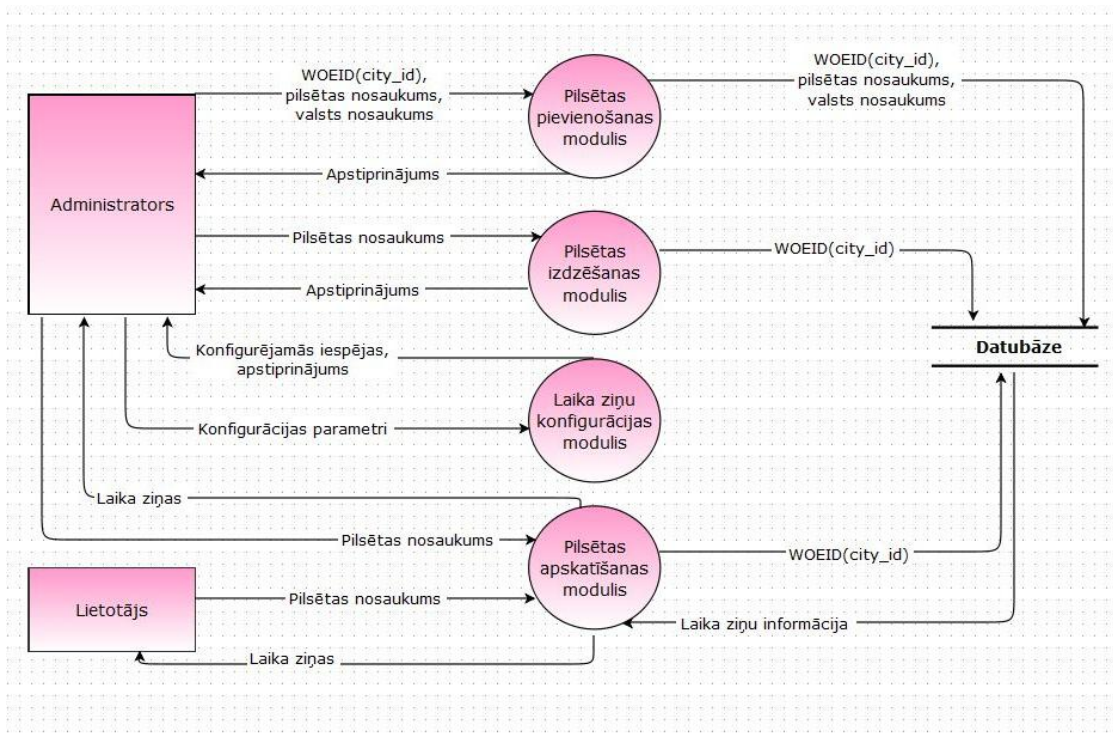
3.3.1 Starpprocetu atkarības

3.3.1.1 Nultā līmeņa datu plūsmu diagramma



Attēls nr.3.1: „Nultā līmeņa datu plūsmu diagramma”

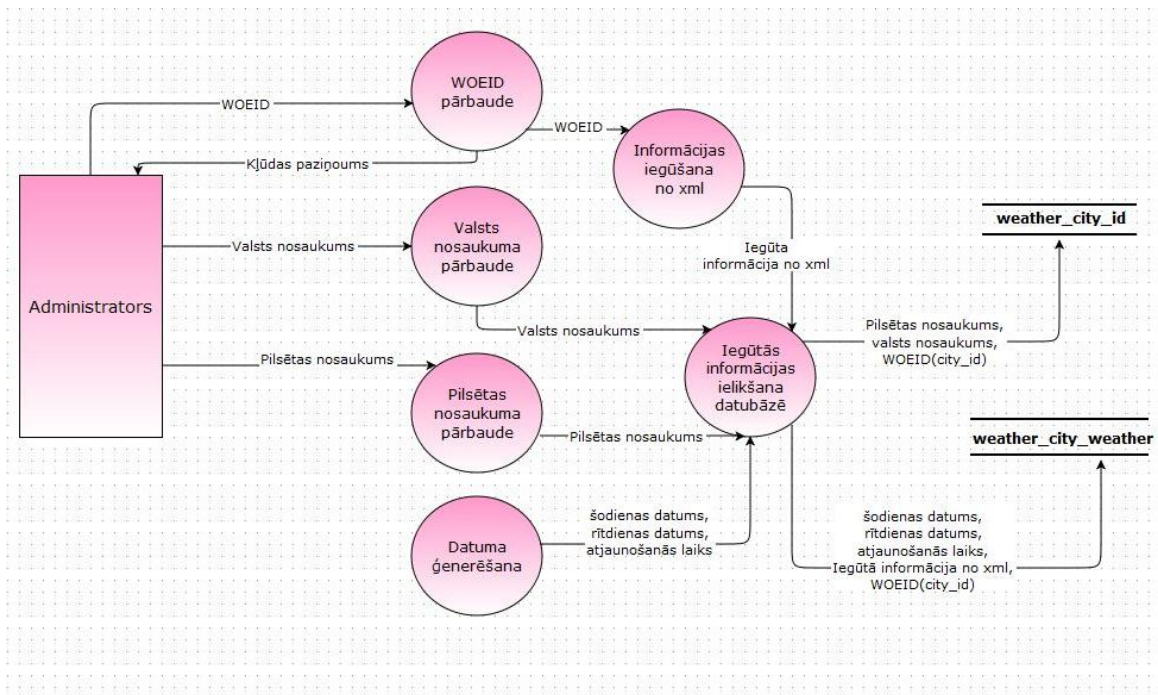
3.3.1.2 Pirmā līmeņa datu plūsmu diagramma



Attēls nr.3.2: „Pirmā līmeņa datu plūsmu diagramma”

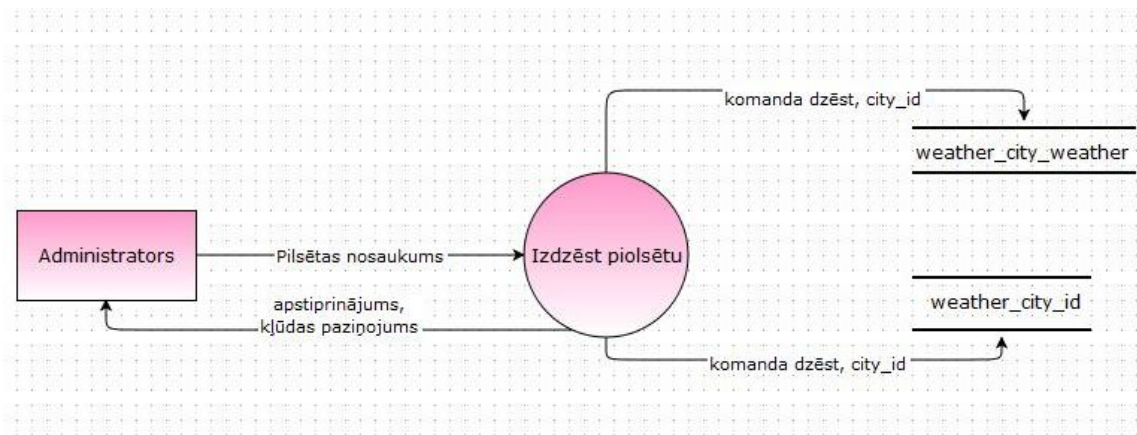
3.3.1.3 Otrā līmeņa datu plūsmu diagrammas

3.3.1.3.1 Pilsētas pievienošanas modulis



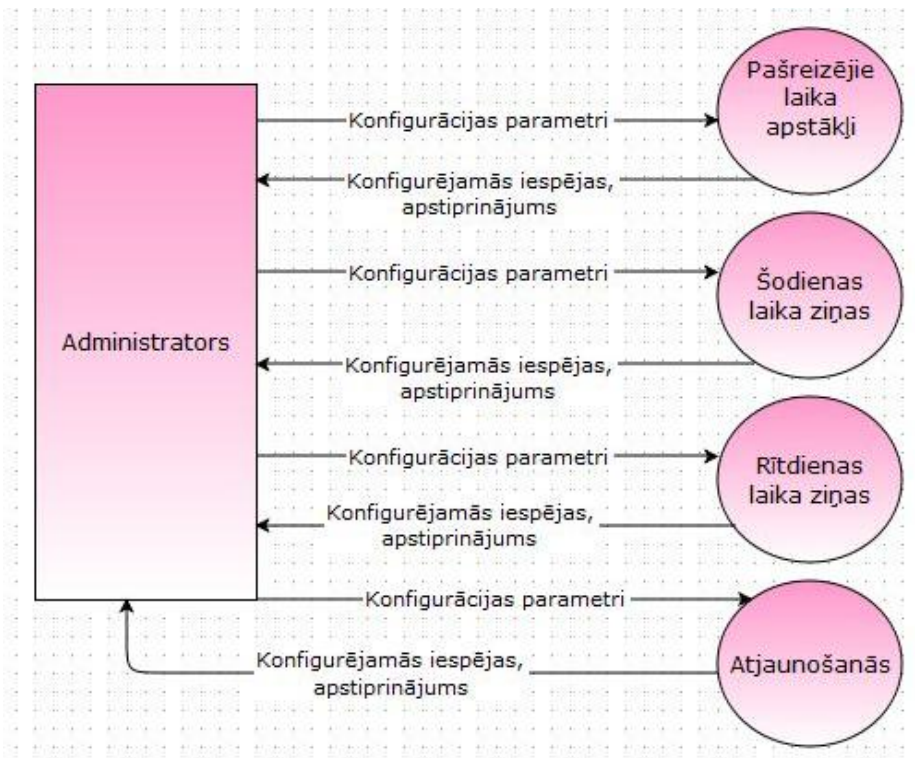
Attēls nr.3.3: „Pilsētas pievienošanas modulis”

3.3.1.3.2 Pilsētas izdzēšanas modulis



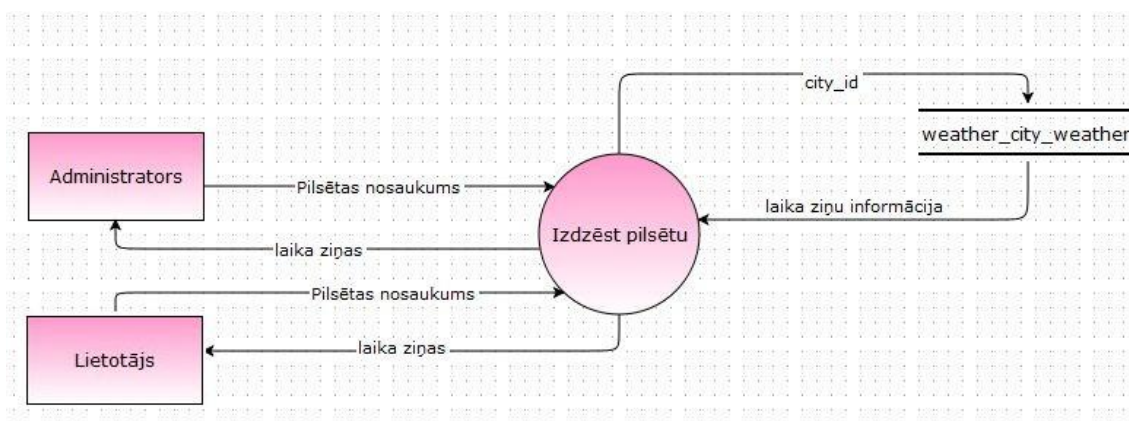
Attēls nr.3.4: „Pilsētas izdzēšanas modulis”

3.3.1.3.3 Laika ziņu konfigurācijas modulis



Attēls nr.3.5: „Laika ziņu konfigurācijas modulis”

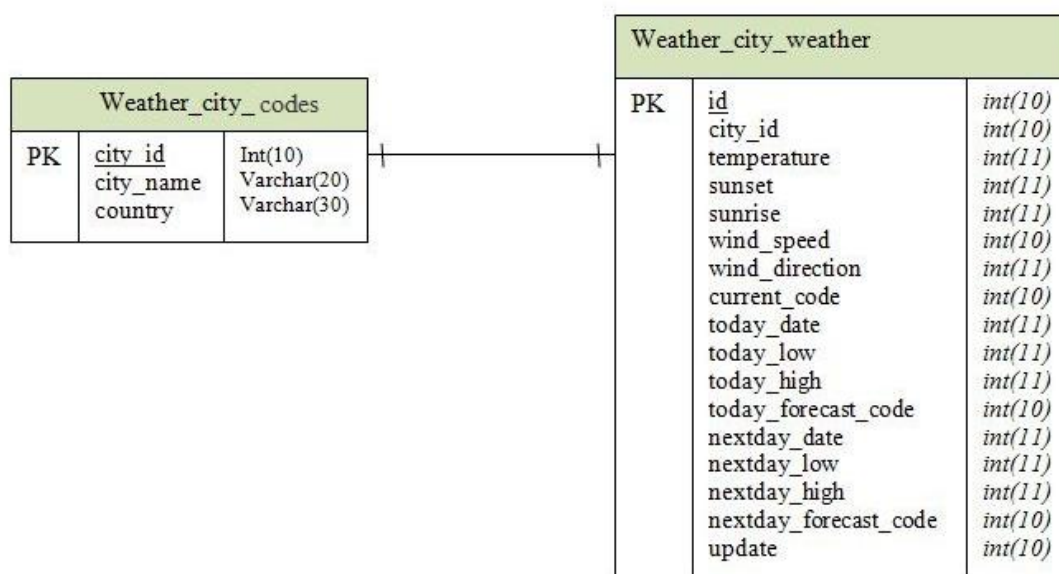
3.3.1.3.4 Pilsētas apskatīšanas modulis



Attēls nr.3.6: „Pilsētas apskatīšanas modulis”

3.3.2 Datu atkarības

3.3.2.1 Fiziskais ER modelis



Attēls nr.3.7: „Fiziskais ER modelis”

3.4 Datu detalizēts projektējums

3.4.1 weather_city_weather

Šajā tabulā glabājas informācija par pilsētas pašreizējajiem un prognozētajiem laika apstākļiem un arī laiki un datumi: šodienas, rītdienas, atjaunošanās.

Tabula – „weather_city_weather”

Lauks	Tips	Atslēga	Obligāti jāaizpilda	Komentārs
city_id	int(10)	UNSIGNED	Jā	Pilsētas unikāls WOEID kods
temperature	int(11)		Jā	Pašreizējā temperatūra
id	int(10)	UNSIGNED, primary key	Jā	Unikāls id, kurš ģenerējas automātiski
sunset	int(11)		Jā	Saulriets
sunrise	int(11)		Jā	Saullēkts
wind_speed	int(10)	UNSIGNED	Jā	Vēja ātrums
wind_direction	int(11)		Jā	Vēja virziens
current_code	int(10)		Jā	Kods, ko izmanto pašreizējās prognozes bildītei un aprakstam
today_date	int(11)		Jā	Šodienas datums
today_low	int(11)		Jā	Šodienas prognozētā zemākā temperatūra
today_high	int(11)		Jā	Šodienas prognozētā augstākā temperatūra
today_forecast_code	int(10)	UNSIGNED	Jā	Kods, ko izmanto šodienas bildītei un prognozes aprakstam
nextday_date	int(11)		Jā	Rītdienas datums
nextday_low	int(11)		Jā	Rītdienas prognozētā zemākā temperatūra
nextday_high	int(11)		Jā	Rītdienas prognozētā augstākā temperatūra
nextday_forecast_code	int(10)	UNSIGNED	Jā	Kods, ko izmanto rītdienas bildītei un prognozes aprakstam
update	int(10)		Jā	Atjaunošanās datums un laiks

3.4.2 weather_city_codes

Tabulā glabājas informācija, ko administrators ievadījis un sistēma apstiprinājusi, kā derīgu, tie ir WOEID(city_id), pilsētas nosaukums un valsts nosaukums.

3.2. tabula

Tabula - „weather_city_codes”

Lauks	Tips	Atslēga	Obligāti jāaizpilda	Komentārs
city_id	int(10)	UNSIGNED, primary key	Jā	Pilsētas unikāls WOEID kods
city_name	varchar(20)		Jā	Pilsētas nosaukums
country	varchar(30)		Jā	Valsts, kurā pilsēta atrodas

3.5 Ārējā saskarne

DRUPAL modulis „Laika ziņas” , kā sistēma ir veidota vienkārša, lai lietotājiem un administratoriem būtu viegli apgūt sistēmu, kā arī orientēties tajā.

3.5.1 Lietotāju saskarne

- Mainot pilsētas, laika ziņu blokā, informācija nomainās nepārlādējoties lapai.
- Mainot valodas visām pogām un iespējām, jābūt attiecīgajā valodā.
- Visām iespējām jābūt skaidri saprotamām.

3.5.2 Skati

3.5.2.1 Pilsētas pievienošana

Iespēja pievienot pilsētu (skat.attēlu:3.8). Lai pilsēta tiktu veiksmīgi pievienota, klikšķinot uz pogas „Saglabāt pilsētu”, ir nepieciešams aizpildīt visus laukus.

Pievienot pilsētu

Ievadiet WOEID: *

Lai iegūtu WOEID kodu jāiet uz [Weather](#) un meklējiet pilsētu.

Pilsētas nosaukums: *

Pilsētas nosaukums

Valsts nosaukums: *

Valsts nosaukums

Attēls nr.3.8: „Pilsētas pievienošana”

3.5.2.2 Pilsētas izdzēšana

Lai izdzēstu izvēlēto pilsētu, izkrītošajā izvēlnē tā jāizvēlas un lai apstiprinātu dzēšanu jāieklikšķina izvēles rūtiņā, ja tā būs atstāta tukša pilsēta neizdzēsīsies.

Izdzēst pilsētu

Izvēlaties pilsētas vārdu, kuru vēlaties izdzēst:

Rīga ▾

Ja jūs tiešām vēlaties izdzēst pilsētu noklikšķiniet uz izvēles rūtiņas.
Ja jūs tiešām vēlaties izdzēst pilsētu noklikšķiniet uz izvēles rūtiņas.

Attēls nr.3.9: „Pilsētas izdzēšana”

Izdzēst pilsētu

Izvēlaties pilsētas vārdu, kuru vēlaties izdzēst:

Rīga ▾
Jelgava
Rīga
Iecava
Parīze

Ja jūs tiešām vēlaties izdzēst pilsētu noklikšķiniet uz izvēles rūtiņas.
 Ja jūs tiešām vēlaties izdzēst pilsētu noklikšķiniet uz izvēles rūtiņas.

Attēls nr.3.10: „Pilsētas izdzēšana: pilsētas no datubāzes”

3.5.2.3 Laika ziņu aplūkošana

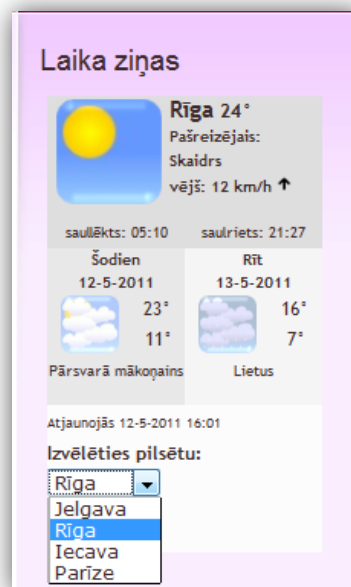
Jebkuram lietotājam ir iespēja aplūkot vienādas laika ziņas (skat.attēlu.3.11), kur šajā gadījumā ir iespēja aplūkot visu pieejamo informāciju.



Attēls nr.3.11: „Laika ziņu aplūkošana”

3.5.2.4 Pilsētas nomainīšana

Jebkuram lietotājam aplūkojot laika ziņas ir iespēja nomainīt apskatāmo pilsētu (skat.attēlu:3.12), pilsētas izkrītošajā izvēlnē tiek ievietotas no datubāzē pieejamajām.



Attēls nr.3.12: „Pilsētas nomainīšana”

3.5.2.5 Laika ziņu konfigurācija(iespējas)

Iespējas sistēmas administratoram konfigurēt un mainīt pēc saviem ieskatiem un vēlmēm laika ziņu bloka izskatu un informāciju tajā. Laika ziņu pieejamās konfigurācijas iespējas (skat.attēlu:3.13)

Laika ziņu iespējas

Pašreizējie laika apstākļi:

- Rādīt pašreizējos laika apstākļus
- Rādīt pašreizējo temperatūru
- Rādīt pašreizējo laika apstākļu skaidrojumu
- Rādīt vēja ātrumu
- Rādīt vēja virzienu
- Rādīt saulrietu
- Rādīt saullēktu

Šodienas laika prognoze:

- Rādīt šodienas laika prognozi
- Rādīt šodienas datumu
- Rādīt šodienas laika apstākļu skaidrojumu
- Rādīt šodienas laika prognozes attēlu

Rītdienas laika prognoze:

- Rādīt rītdienas laika prognozi
- Rādīt rītdienas datumu
- Rādīt rītdienas laika prognozes attēlu
- Rādīt rītdienas laika apstākļu skaidrojumu

Atjaunošanās:

- Rādīt atjaunošanās laiku

Saglabāt iestatījumus

Attēls nr.3.13: „Laika ziņu konfigurācija(iespējas)”

Klikšķinot uz norādītajām izvēles rūtiņām ar to darbību skaidrojumiem un saglabājot iestatījumus, mainās laika ziņu bloka izskats un informācija tajā (skat.attēlu:3.14 un 3.15)

Laika ziņu iespējas

Pašreizējie laika apstākļi:

- Rādīt pašreizējos laika apstākļus
- Rādīt pašreizējo temperatūru
- Rādīt pašreizējo laika apstākļu skaidrojumu
- Rādīt vēja ātrumu
- Rādīt vēja virzienu
- Rādīt saulrietu
- Rādīt saullēktu

Šodienas laika prognoze:

- Rādīt šodienas laika prognozi
- Rādīt šodienas datumu
- Rādīt šodienas laika apstākļu skaidrojumu
- Rādīt šodienas laika prognozes attēlu

Rītdienas laika prognoze:

- Rādīt rītdienas laika prognozi
- Rādīt rītdienas datumu
- Rādīt rītdienas laika prognozes attēlu
- Rādīt rītdienas laika apstākļu skaidrojumu

Atjaunošanās:

- Rādīt atjaunošanās laiku

Laika ziņas

Rīga 9°
Pašreizējais:
Mākoņains
vējš: 4 km/h ↑

saulārkts: 05:04

Rī: 17-5-2011

Atjaunojās: 16-5-2011 09:17

Izvēlēties pilsētu:
Rīga

Languages

- English
- Latviešu
- Русский

Attēls nr.3.14: „Laika ziņu iespējas, noņemta šodienas laika apstākļu prognoze”

Laika ziņu iespējas

Izmaiņas tiks saglabātas

Pašreizējie laika apstākļi:

- Rādīt pašreizējos laika apstākļus
- Rādīt pašreizējo temperatūru
- Rādīt pašreizējo laika apstākļu skaidrojumu
- Rādīt vēja ātrumu
- Rādīt vēja virzienu
- Rādīt saulrietu
- Rādīt saullēktu

Šodienas laika prognoze:

- Rādīt šodienas laika prognozi
- Rādīt šodienas datumu
- Rādīt šodienas laika apstākļu skaidrojumu
- Rādīt šodienas laika prognozes attēlu

Rītdienas laika prognoze:

- Rādīt rītdienas laika prognozi
- Rādīt rītdienas datumu
- Rādīt rītdienas laika prognozes attēlu
- Rādīt rītdienas laika apstākļu skaidrojumu

Atjaunošanās:

- Rādīt atjaunošanās laiku

Laika ziņas

Rīga 9°
Pašreizējais:
Mākoņains
vējš: 4 km/h ↑

saulārkts: 05:04 saulriets: 21:36

Izvēlēties pilsētu:
Rīga

Languages

- English
- Latviešu
- Русский

Attēls nr.3.15: „Laika ziņu iespējas, rāda tikai pašreizējos laika apstākļus”

4. TESTĒŠANAS DOKUMENTĀCIJA

Šis dokuments ir DRUPAL moduļa „Laika ziņas” testēšanas dokumentācija.

4.1 Testēšanas plāns

Testēšanā jāveic sistēmas funkcionalitātes un vadības pārbaudes. Procesā jāpārbauda, jānotestē visas prasību specifikācijā norādītās prasības.

Procesā testētie pārvaldības moduļi:

- Pilsētas pievienošanas modulis
- Pilsētas izdzēšanas modulis
- Laika ziņu konfigurācijas modulis
- Pilsētas apskatīšanas modulis

Testēšana tik veikta izmantojot melnās kastes principu.

4.2 Testēšanas žurnāls

4.2.1 Pilsētas pievienošanas modulis

4.1. tabula

Pilsētas pievienošanas moduļa testēšanas rezultāti

Testa nr.	Testa apraksts	Rezultāts	Komentārs
1	Vai visi nepieciešamie ievades lauki ir attēloti	Izpildās	
2	Vai lauks „valsts nosaukums”, ir obligāti aizpildāms	Izpildās	
3	Vai lauks „pilsētas nosaukums” ir obligāti aizpildāms	Izpildās	
4	Vai lauks „ievadiet WOEID” ir obligāti aizpildāms	Izpildās	
5	Veic WOEID pārbaudi, ja nederīgs kods, paziņo	Izpildās	
6	Pilsētas nosaukums nedrīkst sakrist ar kādu jau esošu datubāzē	Izpildās	

7	Lauks „Valsts nosaukums ir korekti ievadīts”	Izpildās	
8	Lauks „WOEID” ir korekti ievadīts	Izpildās	
9	Lauks „Pilsētas nosaukums” ir korekti ievadīts	Izpildās	
10	Pievienojot pilsētu paziņojums	Izpildās	
11	Uzbīdot kursoru uz saite „WEATHER” tas pasvītrojās	Izpildās	
12	Klikšķinot uz norādītās saite, tā korekti atveras jaunā cilnē	Izpildās	

4.2.2 Pilsētas izdzēšanas modulis

4.2. tabula

Pilsētas izdzēšanas moduļa testēšanas rezultāti

Testa nr.	Testa apraksts	Rezultāts	Komentārs
13	Pilsētas, kuras atrodas datubāzē ir iespējams izdzēst	Izpildās	
14	Neieklikšķinot aptirināšanas izvēles rūtiņā un dzēšot pilsētu, parādās paziņojums	Izpildās	
15	Dzēšanas apstiprinājuma paziņojums	Izpildās	

4.2.3 Laika ziņu konfigurācijas modulis

4.3. tabula

Laika ziņu konfigurācijas moduļa testēšanas rezultāti

Testa nr.	Testa apraksts	Rezultāts	Komentārs
16	Iespējas tiek sakārtotas pa tēmām un izkārtojas viena zem otras	Izpildās	
17	Zem „Pašreizējie laika apstākļi:” ir septiņas iespējas	Izpildās	
18.1	Izķeksējot izvēles rūtiņu „Rādīt pašreizējos laika apstākļus” no bloka pazūd informācija par pašreizējajiem laika	Izpildās	

	apstākļiem		
18.2	Izķeksējot izvēles rūtiņu „Rādīt pašreizējo temperatūru” no bloka pazūd informācija par pašreizējo temperatūru	Izpildās	
18.3	Izķeksējot izvēles rūtiņu „Rādīt pašreizējo laika apstākļu skaidrojumu” no bloka pazūd informācija, kura skaidro pašreizējos laika apstākļus	Izpildās	
18.4	Izķeksējot izvēles rūtiņu „Rādīt vēja ātrumu” no bloka pazūd informācija par vēja ātrumu	Izpildās	
18.5	Izķeksējot izvēles rūtiņu „Rādīt vēja virzienu” no bloka pazūd informācija par vēja virzienu	Izpildās	
18.6	Izķeksējot izvēles rūtiņu „Rādīt saulrietu” no bloka pazūd informācija par saulrietu	Izpildās	
18.7	Izķeksējot izvēles rūtiņu „Rādīt saullēktu” no bloka pazūd informācija par saullēktu	Izpildās	
19	Izķeksējot izvēles rūtiņu „Rādīt šodienas laika prognozi” no bloka pazūd informācija par šodienas prognozētajiem laika apstākļiem	Izpildās	
19.1	Izķeksējot izvēles rūtiņu „Rādīt šodienas datumu” no bloka pazūd informācija par šodienas datumu	Izpildās	
19.2	Izķeksējot izvēles rūtiņu „Rādīt šodienas laika apstākļu skaidrojumu” no bloka pazūd informācija par laika apstākļu skaidrojumu	Izpildās	
19.3	Izķeksējot izvēles rūtiņu „Rādīt šodienas laika prognozes attēlu” no bloka pazūd šodienas laika apstākļu attēls	Izpildās	
20	Izķeksējot izvēles rūtiņu „Rādīt rītdienas laika prognozi” no bloka pazūd	Izpildās	

	informācija par rītdienas prognozētajiem laika apstākļiem		
20.1	Izķeksējot izvēles rūtiņu „Rādīt rītdienas datumu” no bloka pazūd informācija par rītdienas datumu	Izpildās	
20.2	Izķeksējot izvēles rūtiņu „Rādīt rītdienas laika apstākļu skaidrojumu” no bloka pazūd informācija par laika apstākļu skaidrojumu	Izpildās	
20.3	Izķeksējot izvēles rūtiņu „Rādīt rītdienas laika prognozes attēlu” no bloka pazūd rītdienas laika apstākļu attēls	Izpildās	
21	Izķeksējot izvēles rūtiņu „Rādīt atjaunošanās laiku” no bloka pazūd informācija par atjaunošanās laiku un datumu	Izpildās	
22	Saglabājot izmaiņas parādās paziņojums	Izpildās	

4.2.4 Pilsētas apskatīšanas modulis

4.4. tabula

Pilsētas apskatīšanas moduļa testēšanas rezultāti

Testa nr.	Testa apraksts	Rezultāts	Komentārs
23	Iespēja izvēlēties pilsētas no datubāzes	Izpildās	
24	Izvēloties kādu citu pilsētu, informācija nomainās	Izpildās	

4.3 Testēšanas rezultātu kopsavilkums

Testēšana noritēja veiksmīgi, netika atrastas nevienas kritiskas vai maznozīmīgas kļūdas. Sistēma darbojās korekti.

5. PROGRAMMATŪRAS PROJEKTA ORGANIZĀCIJA

Pirms projekta izstrādes tik izvēlēts programmatūras izstrādes modelis pēc, kura vadoties tika izstrādāts programmaprodukts. Izvēlēts tika tieši ūdenskrituma modelis, jo tas ļauj katras fāzes beigās veikt pārbaudi tādējādi panākot augstāku kvalitāti.

Lai varētu skaidri un saprotami nodefinēt prasības un funkcijas, kuras nepieciešams aprakstīt programmatūras prasību specifikācijā (PPS), sākumā sāku ar to, ka izpētīju, kur man nepieciešamo informāciju ir iespējams iegūt un kādā apjomā tā ir pieejama. Pirms programmatūras prasību specifikācijas (PPS) rakstīšanas nolēmu, ka sistēmu veidošu maksimāli vienkāršu, pārskatāmu un lai tā būtu viegli lietojama kā lietotājiem tā arī administratoriem.

Nepieciešamā datu ieguves vieta tika atrasta un prasības nodefinētas es veicu aptuveno darbietilpības novērtējumu, lai pārliecinātos, ka projekta izstrāde iekļausies noteiktajos termiņos. Balstoties uz programmatūras prasību specifikāciju izstrādāju programmatūras projektējuma aprakstu (PPA), procesā ietilpa arī datubāzes modeļa un tabulu struktūru veidošana.

Laika ietilpīgākā projekta fāze bija programmaprodukta koda izstrāde. Testēšana un atklāšana notika arī koda izstrādes procesā, bet tomēr galvenā testēšana risinājās pēc programmaprodukta koda pabeigšanas.

6. KVALITĀTES NODROŠINĀŠANA

Tika veikti vairāki pasākumi, lai nodrošinātu maksimāli lielāku programmaprodukta kvalitāti.

Projekta izstrādes sākumposmā tika izveidota programmatūras prasību specifikācija. Balstoties uz PPS aprakstītājām prasībām un rūpīgas tā izpētes tika izstrādāts programmatūras projektējuma apraksts. Programmas kods tika veidots ņemot vērā programmatūras projektējuma apraksta dokumentu. Lai nodrošinātu precīzu programmas darbību, koda izstrādes procesā vairākkārt un regulāri tika veikta testēšana, pārbaudot katru jaunizveidoto funkciju. Kods tika rakstīts ievērojot vienotu stilu, tādējādi uzlabojot lasāmību un veicinot vieglāku uztveršanu, lai nepieciešamības gadījumā varētu veikt izmaiņas tajā.

Rūpīgi ievērojot W3C standartus tika izstrādāts CSS un HTML kods, kā arī sistēma tika pārbaudīta uz jaunāko versiju pārlūkprogrammām, lai pārlicinātos vai kods darbojas korekti.

7. KONFIGURĀCIJU PĀRVALDĪBA

Šo projektu izstrādāju viena, tādējādi konfigurācijas pārvaldība bija samērā vienkārša un netika izmantotas versiju vadības sistēmas.

Lai nodrošinātu samērā vienkāršo versiju vadību, visu projekta datu glabāšanai tika piemērota arhivēšanas metode, mapēs ar nosaukumu (dd.mm), attiecīgi dd- datums un mm-mēnesis, kurā versija tika arhivēta.

Rezerves kopijas tika glabātas arī zibatmiņā, lai problēmu gadījumā varētu ātri un sekmīgi atjaunot informāciju.

8. DARBIETILPĪBAS NOVĒRTĒJUMS

Projekta izstrādes sākumā tika izveidota Programmatūras prasību specifikācija balstoties uz atvēlēto laika posmu projekta realizēšanai. Specifikācijā tika nodefinētas galvenās funkcijas, kuras nepieciešamas sekmīgai sistēmas darbībai, protams, tika ņemts vērā, ka izvēlētajā izstrādes vidē nav iepriekšējas pieredzes.

Veicot darbietilpības aprēķinus tika izmantotas divas metodes: Basic COCOMO, *SLOCCount*.

Pēc Basic COCOMO metodes

8.1. tabula

COCOMO metode

Aptuvenais rindiņu skaits	1200
Aptuvenā darbietilpība – $2,4 * (\text{rindiņu skaits tūkstošos})^{1,05}$	3,04 personmēneši

Pēc *SLOCCount* metodes, var precīzāk novērtēt darbietilpību, ievērojot 15 faktoros.

Katru faktoru novērtē ar skalu no 1 līdz 6, jeb “Very Low” (ļoti zems), “Low” (zems), “Nominal” (nominālvērtība), “High”(augsts), “VeryHigh” (ļoti augsts), “Extra High” (īpaši augsts).

8.2. tabula

Manis dotā novērtējumu tabula

Faktors	Novērtējums	Reizinātājs
Product attributes		
Required software reliability (nepieciešamais drošums)	Nominal	1
Size of application database (datubāzes apjoms)	Low	0.94
Complexity of the product (produkta sarežģītība)	Nominal	1
Hardware attributes		
Run-time performance constraints (Izpildes laika veiktspējas ierobežojumi)	Nominal	1
Memory constraints (datu glabātuves ierobežojums)	Nominal	1

Volatility of the virtual machine environment (virtuālās mašīnas (operētājsistēmas un aparatūras) nepastāvīgums)	Nominal	1
Required turnabout time (laiks cik jāgaida līdz programma izdod kādu rezultātu)	High	1.07
Personnel attributes		
Analyst capability (analītiķa spējas)	Nominal	1
Applications experience (pieredze izstrādājot programmatūru)	Very Low	1.29
Software engineer capability (programmētāja spējas)	Low	1.17
Virtual machine experience (pieredze darbā ar virtuālo mašīnu)	Nominal	1
Programming language experience (pieredze programmēšanas valodā)	Nominal	1
Project attributes		
Use of "modern" programming practices (modernās programmēšanas pielietojums)	Nominal	1
Use of software tools (palīgprogrammu pielietojums)	Nominal	1
Required development schedule (izstrādes grafika ierobežojumi)	Nominal	1

Izmantojot tos pašus koeficientus : reizinātāju 2,4 un eksponentu 1,05

Koeficients: $1 * 0,94 * 1 * 1 * 1 * 1 * 1,07 * 1 * 1,29 * 1,17 * 1 * 1 * 1 * 1 * 1 = 1,52$

8.3. tabula

SLOCCount metode

Koeficients	1,52
Aptuvenā darbietilpība – 2,4 * (koeficients) ^{1,05}	3,89 personmēneši

Izvērtējot abas metodes un novērtējot darbietilpību, secinu, ka abas metodes diezgan precīzi noteica, mani veidotās sistēmas darbietilpību.

9. IZMANTOTĀ LITERATŪRA UN AVOTI

Grāmatas:

1. APress_Pro_Drupal_Development_2nd_Edition
2. Front End Drupal Designing, Theming, Scripting (Developer's Library)
3. Learning Drupal 6 Module Development [aVKs]

Elektroniskie informācijas avoti:

4. <http://stuff.mit.edu/iap/debian/solutions/sloccount-2.26/table.html>
5. <http://en.wikipedia.org/wiki/COCOMO>
6. W3Schools Online Web Tutorials. [tiešsaiste].

Pieejams: <http://www.w3schools.com/>

Video:

7. Lullabot-DrupalModuleDevelopment-Part1
8. Lullabot-DrupalModuleDevelopment-Part2

PIELIKUMI

1.pielikums. Koda fragments no weather.module faila

```
function weather_get_weather_from_xml($city_id) {
    $url = "http://weather.yahooapis.com/forecastrss?w=$city_id&u=c";
    $file = file_get_contents($url);
    $xml = simplexml_load_string($file);
    $ns = $xml->getNamespaces(TRUE);
    $xml->registerXPathNamespace('c', $ns['yweather']);

    $element_wind = $xml->xpath('//c:wind');
    $element_astronomy = $xml->xpath('//c:astronomy');
    $element_condition = $xml->xpath('//c:condition');
    $element_forecast = $xml->xpath('//c:forecast');

    //wind
    $wind = array();
    foreach ($element_wind[0]->attributes() as $a => $b) {
        $wind[$a] = (string)$b;
    }

    //astronomy
    $astronomy = array();
    foreach ($element_astronomy[0]->attributes() as $a => $b) {
        $astronomy[$a] = (string)$b;
    }

    //condition
    $condition = array();
    foreach ($element_condition[0]->attributes() as $a => $b) {
        $condition[$a] = (string)$b;
    }

    //forecast
    $forecast = array();
    foreach ($element_forecast as $key => $value) {
        $aux = array();
        foreach ($element_forecast[$key]->attributes() as $a => $b) {
            $aux[$a] = (string)$b;
        }
        $forecast[] = $aux;
    }
    $weather = array (
        'wind'=>$wind,
        'astronomy'=>$astronomy,
        'condition'=>$condition,
        'forecast'=>$forecast,
    );
    return $weather;
}

function weather_update_weather() {
    $result = db_query("SELECT city_id
        FROM {weather_city_codes}");
    while ($item = db_fetch_object($result)) {
        $city_id = $item->city_id;
        $weather = weather_get_weather_from_xml($city_id);
    }
}
```

```

        weather_update_weather_table($city_id,                $weather['wind'],
$weather['astronomy'], $weather['condition'], $weather['forecast']);
    }
}

//Update weather table, ar iegūtajiem datiem
function weather_update_weather_table($city_id,    $wind,    $astronomy,
$condition, $forecast) {

    // izdzēš infotmāciju, kura piesaistīta norādītajam city_id, njem un
pielīdzina mainiigos, tad saliek attieciigi db
    db_query("DELETE FROM {weather_city_weather}
        WHERE city_id=%d", $city_id);

    $wind_direction = $wind['direction'];
    $wind_speed = $wind['speed'];

    $sunset = $astronomy['sunset'];
    $sunset = strtotime($sunset);

    $sunrise = $astronomy['sunrise'];
    $sunrise = strtotime($sunrise);

    $current_code = $condition['code'];
    $temperature = $condition['temp'];

    $today_date = $forecast[0]['date'];
    $today_date = strtotime($today_date);

    $today_low = $forecast[0]['low'];
    $today_high = $forecast[0]['high'];
    $today_forecast_code = $forecast[0]['code'];

    $nextday_date = $forecast[1]['date'];
    $nextday_date = strtotime($nextday_date);

    $nextday_low = $forecast[1]['low'];
    $nextday_high = $forecast[1]['high'];
    $nextday_forecast_code = $forecast[1]['code'];

    $update = time(); //kad liek datus db tad kaa update time pieliekas
pashreizejais laiks

    db_query("INSERT INTO {weather_city_weather} (city_id, wind_direction,
wind_speed, sunset, sunrise, current_code, temperature, today_date,
today_low, today_high, today_forecast_code, nextday_date, nextday_low,
nextday_high, nextday_forecast_code, `update`)
        VALUES ('%d', '%d', '%d', '%d', '%d', '%d', '%d', '%d', '%d', '%d',
'%d', '%d', '%d', '%d', '%d', '%d')", $city_id, $wind_direction,
$wind_speed, $sunset, $sunrise, $current_code, $temperature, $today_date,
$today_low, $today_high, $today_forecast_code, $nextday_date, $nextday_low,
$nextday_high, $nextday_forecast_code, $update);
}

function weather_block($op = 'list', $delta = 0, $edit = array()) {
    switch ($op) {
        case 'list':
            $blocks[0]['info'] = t('Krista weather block');
            $blocks[0]['cache'] = BLOCK_NO_CACHE;
            return $blocks;
        case 'view':
            $block['subject'] = t('Weather news');
            // We theme our array of links as an unordered list.

```

```

    $block['content'] = theme('weather_block',
$_SESSION['weather_city_id']);
    return $block;
}
}

/*
Implementation of hook_theme().
*/
function weather_theme() {
    return array(
        'weather_block' => array(
            'arguments' => array('form' => NULL, 'city_id' => NULL),
            'template' => 'weather-content',
        ),
    );
}

function template_preprocess_weather_block(&$vars) {

    $city_id = $_SESSION['weather_city_id'];
    $weather = weather_get_weather($city_id);
    $weather['current_description'] = weather_get_description($city_id);
    $weather['current_img'] = '';
    $weather['wind_direction'] = '';
    $weather['today_date'] = weather_today_format_date($city_id);
    $weather['today_picture'] = '';
    $weather['today_description'] =
weather_get_description_today($city_id);
    $weather['sunrise'] = weather_sunrise_format_date($city_id);
    $weather['sunset'] = weather_sunset_format_date($city_id);
    $weather['nextday_date'] = weather_nextday_format_date($city_id);
    $weather['nextday_picture'] = '';
    $weather['nextday_description'] =
weather_get_description_nextday($city_id);
    $vars['city_id'] = $city_id;
    $weather['chose_city'] = weather_city_codes();
    $weather['update'] = weather_update_format_date($city_id);

    drupal_add_css(drupal_get_path('module', 'weather') . '/stile.css');

    $settings1 = variable_get('show_options1', array(0, 0, 0, 0, 0, 0, 0));
    $settings2 = variable_get('show_options2', array(0, 0, 0, 0));
    $settings3 = variable_get('show_options3', array(0, 0, 0, 0));
    $settings4 = variable_get('show_options4', array(0));
    //CURRENT
    if ($settings1['current']) {
        $vars['current_img'] = $weather['current_img'];
        $vars['city_name'] = $weather['city_name'];

        if ($settings1['temperature']) {
            $vars['temperature'] = $weather['temperature'];
        }
        if ($settings1['current_description']) {
            $vars['current_description'] = $weather['current_description'];
        }
    }
}

```

```

    }
    if ($settings1['wind_speed']) {
        $vars['wind_speed'] = $weather['wind_speed'];
    }
    if ($settings1['wind_direction']) {
        $vars['wind_direction'] = $weather['wind_direction'];
    }
}

    if ($settings1['sunrise']) {
        $vars['sunrise'] = $weather['sunrise'];
    }
    if ($settings1['sunset']) {
        $vars['sunset'] = $weather['sunset'];
    }

//TODAY
if ($settings2['today_forecast']) {
    if ($settings2['today_date']){
        $vars['today_date'] = $weather['today_date'];
    }
    if ($settings2['today_picture']){
        $vars['today_picture'] = $weather['today_picture'];
    }

    $vars['today_high'] = $weather['today_high'];
    $vars['today_low'] = $weather['today_low'];

    if ($settings2['today_description']){
        $vars['today_description'] = $weather['today_description'];
    }
}

//TOMORROW
if($settings3['nextday_forecast']) {
    if ($settings3['nextday_date']){
        $vars['nextday_date'] = $weather['nextday_date'];
    }
    if ($settings3['nextday_picture']) {
        $vars['nextday_picture'] = $weather['nextday_picture'];
    }
    $vars['nextday_high'] = $weather['nextday_high'];
    $vars['nextday_low'] = $weather['nextday_low'];

    if($settings3['nextday_description']){
        $vars['nextday_description'] = $weather['nextday_description'];
    }
}

    if($settings4['update']) {
        $vars['update'] = $weather['update'];
    }
    $vars['chose_city'] = $weather['chose_city'];
}

//get info from database, lai liktu block
function weather_get_weather($city_id) {
    $result = db_query("
        SELECT *
        FROM {weather_city_weather} AS A, {weather_city_codes} AS B
        WHERE A.city_id=B.city_id AND A.city_id=%d
        ", $city_id);
}

```

```

$weather = array();
while ($item = db_fetch_object($result)) {
    $weather['temperature'] = $item->temperature;
    $weather['city_name'] = $item->city_name;
    $weather['today_date'] = $item->today_date;
    $weather['today_low'] = $item->today_low;
    $weather['today_high'] = $item->today_high;
    $weather['nextday_date'] = $item->nextday_date;
    $weather['nextday_low'] = $item->nextday_low;
    $weather['nextday_high'] = $item->nextday_high;
    $weather['wind_speed'] = $item->wind_speed;
    $weather['wind_direction'] = $item->wind_direction;
    $weather['sunset'] = $item->sunset;
    $weather['sunrise'] = $item->sunrise;
    $weather['update'] = $item->update;
}
return $weather;
}

//izkrītošā izvēlne blokā, kur izvēlas pilsētas
function weather_city_codes() {
    return drupal_get_form('weather_city_codes_form');
}

function weather_city_codes_form() {
    $form = array();

    $cities = array(); //uztaisa array no datubāzē atrastajām pilsētām
    [city_name un city-id]
    $result = db_query("SELECT * FROM {weather_city_codes}");
    while ($item = db_fetch_array($result)){
        $cities[$item['city_id']] = $item['city_name'];
    }

    $form['city_codes'] = array(//izkritosha izvelne
        '#type' => 'select',
        '#title' => t('Choose city'),
        '#default_value' => $_SESSION['weather_city_id'],
        '#options' => $cities,
    );

    $form['submit'] = array(//poga submit
        '#type' => 'submit',
        '#value' => t('Change'),
    );

    return $form;
}

function weather_city_codes_form_submit(&$form) {
    $_SESSION['weather_city_id']=$form['city_codes']['#value'];
}

/*
Implementation of hook_menu().
*/
function weather_menu() {
    $items = array();

    $items['admin/settings/weather'] = array(
        'title' => 'Weather settings',
        'description' => 'Administer your&#39s forecasts',
    );
}

```

```

    'type' => MENU_NORMAL_ITEM,
    'page callback' => 'system_admin_menu_block_page',
    'access arguments' => array('administer weather'),
    'file' => 'system.admin.inc',
    'file path' => drupal_get_path('module', 'system'),
);

$items['admin/settings/weather/forecast_options'] = array(
  'title' => 'Forecast options',
  'description' => 'Configure forecast',
  'page callback' => 'weather_forecast_options',
  'access arguments' => array('administer weather'),
);

$items['admin/settings/weather/add'] = array(
  'title' => 'Add city',
  'description' => 'Add new city',
  'page callback' => 'weather_add_city',
  'access arguments' => array('administer weather'),
);

$items['admin/settings/weather/delete'] = array(
  'title' => 'Delete city',
  'description' => 'Delete city from list',
  'page callback' => 'weather_delete_city',
  'access arguments' => array('administer weather'),
);
return $items;
}

function weather_forecast_options() {
  return drupal_get_form('weather_settings_form');
}

//izsauc fonkciju formu pievienot pilsetu
function weather_add_city() {
  return drupal_get_form('weather_add_city_form') ;
}

//izsauc fonkciju formu izdzēst pilsetu
function weather_delete_city() {
  return drupal_get_form('weather_delete_city_form');
}

// WEATHER OPTIONS -admin options, izvelies ko radis tavas laika zinjaas
forma
function weather_settings_form() {
  $form = array();

  $show_options1 = array(
    'current' => t('Show current forecasts'),
    'temperature' => t('Show current temperature'),
    'current_description' => t('Show current weather description'),
    'wind_speed' => t('Show wind speed'),
    'wind_direction' => t('Show wind direction'),
    'sunset' => t('Show sunset'),
    'sunrise' => t('Show sunrise'),
  );

  $show_options2 = array(
    'today_forecast' => t('Show today&#39s forecast'),
    'today_date' => t('Show today&#39s date'),
    'today_description' => t('Show today&#39s weather description'),

```

```

    'today_picture' => t('Show today&#39s weather picture'),
);

$show_options3 = array(
    'nextday_forecast' => t('Show Tomorrow&#39s forecast'),
    'nextday_date' => t('Show Tomorrow&#39s date'),
    'nextday_picture' => t('Show nextday&#39s weather picture'),
    'nextday_description' => t('Show nextday&#39s weather description'),
);

$show_options4 = array(
    'update' => t('Show update time'),
);

$form['show_options1'] = array(
    '#type' => 'checkboxes',
    '#title' => t('Current weather'),
    '#options' => $show_options1,
    '#default_value' => variable_get('show_options1', array(0, 0, 0, 0, 0,
0, 0)),
);

$form['show_options2'] = array(
    '#type' => 'checkboxes',
    '#title' => t('Today forecast'),
    '#options' => $show_options2,
    '#default_value' => variable_get('show_options2', array(0, 0, 0, 0)),
);

$form['show_options3'] = array(
    '#type' => 'checkboxes',
    '#title' => t('Tomorrow forecast'),
    '#options' => $show_options3,
    '#default_value' => variable_get('show_options3', array(0, 0, 0, 0)),
);

$form['show_options4'] = array(
    '#type' => 'checkboxes',
    '#title' => t('Update'),
    '#options' => $show_options4,
    '#default_value' => variable_get('show_options4', array(0)),
);

$form['submit'] = array( //poga submit
    '#type' => 'submit',
    '#value' => t('Save settings'),
);

return $form;
}

function weather_settings_form_submit(&$form, &$form_state){
    variable_set('show_options1', $form_state['values']['show_options1']);
    $dabujam1 = variable_get('show_options1', array(0, 0, 0, 0, 0, 0, 0, 0));

    variable_set('show_options2', $form_state['values']['show_options2']);
    $dabujam2 = variable_get('show_options2', array(0, 0, 0, 0));

    variable_set('show_options3', $form_state['values']['show_options3']);
    $dabujam3 = variable_get('show_options3', array(0, 0, 0, 0));

    variable_set('show_options4', $form_state['values']['show_options4']);

```

```

$dabujam4 = variable_get('show_options4', array(0));

dsm(t('Changes have been saved'));
}

//ADD CITY -pilsētas pievienošanas forma
function weather_add_city_form() {
    $form = array();

    $form['city_id'] = array(
        '#type' => 'textfield',
        '#title' => t('Input WOEID'),
        '#description' => t('To get this WOEID code go to !link and look up a
city.', array('!link' => l('Weather',
'http://sigizmund.info/woeidinfo/?woeid=', array('attributes' =>
array('target' => '_new'))))),
        '#default_value' => '',
        '#required' => TRUE,
    );

    $form['city_name'] = array(
        '#type' => 'textfield',
        '#title' => t('City name'),
        '#description' => t('City name'),
        '#default_value' => '',
        '#required' => TRUE,
    );

    $form['country'] = array(
        '#type' => 'textfield',
        '#title' => t('Country name'),
        '#description' => t('Country name'),
        '#default_value' => '',
        '#required' => TRUE,
    );

    $form['submit'] = array( //poga submit
        '#type' => 'submit',
        '#value' => t('Save settings'),
    );

    return $form;
}

//pārbauda vai ievadītie lauki(ADD CITY) ir korekti aizpildīti
function weather_add_city_form_validate(&$form, &$form_state){
    $city_id=$form_state['values']['city_id'];
    $city_name=$form_state['values']['city_name'];

    $result=db_query("SELECT * FROM {weather_city_codes}");
    while($data = db_fetch_object($result)){
        $id = $data->city_id;
        $name = $data->city_name;

        if($id == $city_id) {
            form_set_error('city_id', t('City with this WOEID already
exist.'));
        }
        elseif($name == $city_name) {
            form_set_error('city_name', t('City with that name already
exists.'));
        }
    }
}

```

```

if (!is_numeric($form_state['values']['city_id'])) {
    form_set_error('city_id', t('WOEID must contain numbers.'));
}

elseif (weather_check_woeid($form_state['values']['city_id'])==FALSE) {
    form_set_error('city_id', t('This WOEID does not respond with
forecasts.'));
}

elseif($name == $city_name) {
    form_set_error('city_name', t('City with that name already
exists.'));
}

elseif (is_numeric($form_state['values']['city_name'])) {
    form_set_error('city_name', t('City name must contain letters.'));
}

elseif (strlen(trim($form_state['values']['city_name']))<3) {
    form_set_error('city_name', t('City name is too short.'));
}

elseif (is_numeric($form_state['values']['country'])) {
    form_set_error('country', t('Country name must contain the
letters.'));
}

elseif (strlen(trim($form_state['values']['country']))<3) {
    form_set_error('country', t('Country name is too short.'));
}
}

//funkcija pārbauda vai ievadītais WOEID ir derīgs (dod atbildi no ziņu
lapas)
function weather_check_woeid($city_id) {
    $url = "http://weather.yahooapis.com/forecastrss?w=$city_id&u=c";
    $file = file_get_contents($url);
    $xml = simplexml_load_string($file);
    $ns = $xml->getNamespaces(TRUE);
    $xml->registerXPathNamespace('c', $ns['yweather']);
    $element_location = $xml->xpath('//c:location');
    return count($element_location);
}

//pievieno datubāzē jauno city_id, city_name un country
function weather_add_city_form_submit(&$form, &$form_state){
    $values = $form_state['values'];

    db_query("INSERT INTO {weather_city_codes} (city_id, city_name, country)
VALUES('%d', '%s', '%s')", $values['city_id'],
$values['city_name'], $values['country']);
    $city_id = $values['city_id'];
    $weather = weather_get_weather_from_xml($city_id);
    weather_update_weather_table($city_id, $weather['wind'],
$weather['astronomy'], $weather['condition'], $weather['forecast']);

    dsm(t('City successfully added'));
}

```

2.pielikums. Koda fragments no weather-content.tpl faila

```
<div id = "weather">
  <div id = "weather_block">

    <?php if (isset($city_name)) : ?>
    <div id = "container1">
      <div id = "current_img" >
        <?php print $current_img ?>
      </div>
      <div id = "current" >
        <big><?php print $city_name ?></big>
        <?php if (isset($temperature)) : ?>
          <?php print $temperature ?><?php print t('&#176')?>
        <?php endif; ?>

        <?php if (isset($current_description)) : ?>
          <br/><small><?php print t('Current: ')?><br/>
          <?php print $current_description ?></small>
        <?php endif; ?>
        <?php if (isset($wind_speed)) : ?>
          <small><br/><?php print t('wind: ') ?><?php print $wind_speed ?>
km/h </small>
        <?php endif; ?>
        <?php if (isset($wind_direction)) : ?>
          <?php print $wind_direction ?>
        <?php endif; ?>
      </div>
    </div>

    <?php if (isset($sunrise) && isset($sunset)) : ?>
    <div id="sun">
      <?php if (isset($sunrise)) : ?>
      <div id= "sunrise">
        <?php print t('sunrise: ') ?><?php print $sunrise ?>
      </div>
      <?php endif; ?>
      <?php if (isset($sunset)) : ?>
      <div id= "sunset">
        <?php print t('sunset: ') ?><?php print $sunset ?>
      </div>
      <?php endif; ?>
    </div>
    <?php endif; ?>
  <?php endif; ?>

  <?php if (isset($sunrise) xor isset($sunset)) : ?>
  <div id="sun1">
    <?php if (isset($sunrise)) : ?>
    <div id= "sunrise1">
      <?php print t('sunrise: ') ?><?php print $sunrise ?>
    </div>
    <?php endif; ?>
    <?php if (isset($sunset)) : ?>
    <div id= "sunset1">
      <?php print t('sunset: ') ?><?php print $sunset ?>
    </div>
    <?php endif; ?>
  </div>
  <?php endif; ?>
```

```

<?php if (isset($today_high) && isset($nextday_high)) : ?>

    <?php if (isset($today_date) || isset($nextday_date)) : ?>
        <div id = "container2">

            <div id ="today_date" >
                <?php if (isset($today_date)) : ?>
                    <?php print t('Today')?>
                    <br/><?php print $today_date ?>
                <?php endif; ?>
                <?php if (!isset($today_date)) : ?>
                    <?php print t('Today')?>
                <?php endif; ?>
            </div>

            <div id ="nextday_date" >
                <?php if (isset($nextday_date)) : ?>
                    <?php print t('Tomorrow')?>
                    <br/><?php print $nextday_date ?>
                <?php endif; ?>
                <?php if (!isset($nextday_date)) : ?>
                    <?php print t('Tomorrow')?>
                <?php endif; ?>
            </div>

        </div>
    <?php endif; ?>

<?php if (!isset($today_date) && !isset($nextday_date)) : ?>
    <div id = "dayday">
        <div id = "todaytoday">
            <?php if (!isset($today_date)) : ?>
                <?php print t('Today')?>
            <?php endif; ?>
        </div>
        <div id = "tomorrowtomorrow">
            <?php if (!isset($nextday_date)) : ?>
                <?php print t('Tomorrow')?>
            <?php endif; ?>
        </div>
    </div>
<?php endif; ?>

<div id = "container3">
<div id ="today_picture" >
    <?php if (isset($today_picture)) : ?>
        <?php print $today_picture ?>
    <?php endif; ?>
</div>
<div id ="today" >
    <?php if (isset($today_high)) : ?>
        <?php print $today_high ?><?php print t('&#176')?><br/><?php print
$today_low ?><?php print t('&#176')?>
    <?php endif; ?>
</div>
<div id ="nextday_picture" >
    <?php if (isset($nextday_picture)) : ?>
        <?php print $nextday_picture ?>
    <?php endif; ?>
</div>
<div id ="nextday" >
    <?php if (isset($nextday_high)) : ?>

```

```

        <?php print $nextday_high ?><?php print t('&#176')?><br/><?php
print $nextday_low ?><?php print t('&#176')?>
        <?php endif; ?>
    </div>
</div>

<?php if (isset($today_description) || isset($nextday_description)) : ?>
    <div id = "container4">
        <div id = "today_description" >
            <?php if (isset($today_description)) : ?>
                <?php print $today_description ?>
            <?php endif; ?>
        </div>
        <div id = "nextday_description" >
            <?php if (isset($nextday_description)) : ?>
                <?php print $nextday_description ?>
            <?php endif; ?>
        </div>
    </div>
<?php endif; ?>

<?php endif; ?>

<?php if (isset($today_high) && !isset($nextday_high)) : ?>
    <div id = "container2_one">
        <div id = "today_date_one" >
            <?php if (isset($today_date)) : ?>
                <?php print t('Today')?>
                <?php print $today_date ?>
            <?php endif; ?>
            <?php if (!isset($today_date)) : ?>
                <?php print t('Today')?>
            <?php endif; ?>
        </div>
    </div>

    <div id = "container3_one">
        <div id = "today_picture_one" >
            <?php if (isset($today_picture)) : ?>
                <?php print $today_picture ?>
            <?php endif; ?>
        </div>

        <div id = "today_one" >
            <?php if (isset($today_high)) : ?>
                <?php print $today_high ?><?php print t('&#176')?><br/><?php
print $today_low ?><?php print t('&#176')?>
            <?php endif; ?>
        </div>

        <div id = "container4_one">
            <div id = "today_description_one" >
                <?php if (isset($today_description)) : ?>
                    <?php print $today_description ?>
                <?php endif; ?>
            </div>
        </div>
    </div>

<?php endif; ?>

```