

LATVIJAS UNIVERSITĀTE
DATORIKAS FAKULTĀTE

**UZDEVUMU PĀRRAUDZĪBAS
TĪMEKĻA VIETNES IZSTRĀDE**

KVALIFIKĀCIJAS DARBS

Autors: **Pēteris Rudzusīks**

Studenta apliecības Nr.: pr11001

Darba vadītājs: Dr.dat. Dainis Dosbergs

RĪGA 2013

ANOTĀCIJA

Kvalifikācijas darbā aprakstīta uzdevumu pārraudzības tīmekļa vietnes izstrāde. Tīmekļa vietne tās lietotājiem ļauj izveidot uzdevumus, nodot tos izpildīt citiem vietnes lietotājiem un veidot dialogu par uzdevuma izpildes gaitu. Lai vietni lietotu, tajā ir jāpiesakās vai nu izmantojot sistēmas piedāvātās reģistrēšanās un pieteikšanās funkcijas, vai arī izmantojot vienu no *Facebook*, *Google*, *LinkedIn* vai *Twitter* kontiem.

Kvalifikācijas darbā izstrādāts programmprodukts pēc spējas izstrādes metodes, veikta prasību apkopošana lietotājstāstos, aprakstīts sistēmas projektējums, programmēts valodās PHP un JavaScript, veikta sistēmas testēšana.

Atslēgas vārdi: **tīmekļa vietne, uzdevumu pārraudzība, spēja izstrādes metode, PHP.**

ABSTRACT

Paper “task management website development” describes the development of a web-based task management application. The application allows its users to create tasks, assign them to other registered users and to enable a dialog between the two parties. To use this application, a user first must sign up using the provided native sign up form or using one of the provided third party authentication methods – *Facebook*, *Google LinkedIn* or *Twitter*.

The author develops the application using agile software development methodology, defines the requirements as user stories, describes the system's architecture, writes code in PHP and JavaScript and tests the application.

Keywords: **website, task management, agile software development, PHP.**

Satura rādītājs

APZĪMĒJUMU SARAKSTS.....	6
IEVADS.....	7
1. PROGRAMMATŪRAS PRASĪBU APRAKSTS.....	9
1.1. Ievads.....	9
1.1.1. Nolūks.....	9
1.1.2. Darbības sfēra.....	9
1.1.3. Saistība ar citiem dokumentiem.....	9
1.2. Lietotārstāsti.....	9
1.2.1. Moduļa “Pieteikšanās, atteikšanās” lietotārstāsti.....	10
1.2.2. Moduļa “Profils” lietotārstāsti.....	13
1.2.3. Moduļa “Kontakti” lietotārstāsti.....	14
1.2.4. Moduļa “Administrācija” lietotārstāsti.....	15
1.2.5. Moduļa “Uzdevumi” lietotārstāsti.....	16
1.2.6. Vispārīgas prasības sistēmai.....	18
1.3. Nefunkcionālās prasības.....	19
1.3.1. Drošības prasības.....	19
1.3.2. Lietojamības prasības.....	19
2. PROGRAMMATŪRAS PROJEKTĒJUMA APRAKSTS.....	20
2.1. Ievads.....	20
2.1.1. Nolūks.....	20
2.1.2. Saistība ar citiem dokumentiem.....	20
2.2. Dekompozīcijas apraksts.....	20
2.2.1. Datu dekompozīcija.....	20
2.2.1.1. Tabula “user”.....	21
2.2.1.2. Tabula “user_authentication”.....	22
2.2.1.3. Tabula “contact_message”.....	23
2.2.1.4. Tabula “task”.....	24
2.2.1.5. Tabula “task_comment”.....	25
2.2.1.6. Tabula “task_tag”.....	25
2.2.1.7. Tabula “migration”.....	25

2.3. Atkarības apraksts.....	26
2.3.1. Datu atkarības – fiziskais datubāzes modelis.....	26
3. TESTĒŠANAS DOKUMENTĀCIJA.....	27
3.1. Ievads.....	27
3.2. Programmkoda vienībtestēšana.....	27
3.3. Sistēmas akcepttestēšana.....	27
3.3.1. Ievads.....	27
3.3.2. Testējamās raksturiezīmes.....	27
3.3.3. Testēšanas žurnāls.....	28
3.3.3.1. Modulis “Pieteikšanās, atteikšanās”	28
3.3.3.2. Modulis “Profils”.....	32
3.3.3.3. Modulis “Kontakti”	33
3.3.3.4. Modulis “Administrācija”.....	34
3.3.3.5. Modulis “Uzdevumi”	34
4. PROJEKTA ORGANIZĀCIJA.....	38
5. KVALITĀTES NODROŠINĀŠANA.....	39
6. KONFIGURĀCIJU PĀRVALDĪBA.....	40
7. DARBIETILPĪBAS NOVĒRTĒJUMS.....	41
7.1. Darbietilpības novērtējums pirms katras iterācijas.....	41
7.2. Darbietilpības novērtējums pēc izstrādātā koda daudzuma ar COCOMO metodi.....	42
NOBEIGUMS UN SECINĀJUMI.....	44
PATEICĪBAS.....	45
IZMANTOTĀ LITERATŪRA UN AVOTI.....	46
PIELIKUMI.....	47
Kontrolieris “Profile” (classes/controller/profile.php).....	47
Modeļa serviss “ProfileForm” (classes/model/service/profileform.php).....	50
Modeļa servisa “ProfileForm” vienībtesti (tests/model/service/test_profileform.php).....	52

APZĪMĒJUMU SARAKSTS

FuelPHP – PHP valodas atvērtā pirmkoda, kopienas izstrādāts ietvars tīmekļa vietņu izstrādei.

Third party identity provider (TPIP) – ārējs identitātes sniedzējs / apstiprinātājs (šajā darbā, piemēram, *Facebook*, *Twitter*).

Vietējais (*native*) sistēmas lietotājs – lietotājs, kas ir reģistrējies sistēmā ar savu e-pasta adresi un paroli, nevis pieteicies, izmantojot kādu no TPIP.

SQL injekcija – uzbrukuma veids tīmekļa vietnēm, kurā uzbrucējs, izmantojot nepareizu ievaddatu apstrādi, cenšas izpildīt SQL teikumus ar mainītu jēgu.

XSS (Cross-Site Scripting) – ievainojamība tīmekļa vietnē, kas ļauj uzbrucējam izpildīt JavaScript kodu vietnes apmeklētāju pārlūkprogrammās.

CSRF (Cross-Site Request Forgery) – uzbrukuma veids tīmekļa vietnēm, kura pamatā ir likt lietotājam atvērt saites viņam nezinot, tādā veidā izdarot nesankcionētas darbības vietnē.

IEVADS

Kvalifikācijas darba ietvaros tiek izveidota sistēma, kura palīdz tās lietotājiem izveidot uzdevumus, kurus ir jāveic citām personām. Uzdevuma izveidotājs norāda lietotāju, kas būs uzdevuma izpildītājs, norāda uzdevuma tēmu, sniedz aprakstu un nosaka izpildes termiņu. Uzdevuma saņēmējs redz saņemtos uzdevumus, var pievienot komentārus uzdevumam, var uzdot jautājumus par uzdevumu, kā arī ziņot par izpildes gaitu. Sistēmai jābūt pieejamai internetā, tai jābūt ērti lietojamai uz ierīcēm ar dažādām ekrāna izšķirtspējām (stacionāriem datoriem, viedtālruniem).

Tā kā sistēmai jābūt pieejamai internetā, jāpielāgojas dažādām ekrāna izšķirtspējām, tad sistēmu tika nolemts veidot kā tīmekļa vietni, kuras saskarne pielāgojas dažādām izšķirtspējām. Līdz ar šo izvēli, automātiski ir noteiktas dažas tehnoloģijas, kas būs jāizmanto klientpusē – HTML, CSS, JavaScript. Savukārt serverpusē tika lietots PHP tādēļ, ka tā ir izplatīta, atvērta pirmkoda valoda ar plašu pieejamo bibliotēku daudzumu. Pie tam autoram ir pieredze šīs tehnoloģijas izmantošanā, bet nav pietiekamu zināšanu par alternatīvām (piemēram, Java, Python, NodeJs).

Sistēmas pasūtītājs ir persona no uzņēmuma, kurā strādāja kvalifikācijas darba autors. Pasūtītājam ir labas tehniskas zināšanas un pieredze sistēmu izstrādē, izmantojot tās pašas tehnoloģijas. Ar pasūtīju vienojāmies, ka lietotājistāstus (tai skaitā akceptēšanas kritērijus), akcepttestu scenārijus, programmkoda komentārus autors rakstīs angļu valodā.

Papildus mērķim izveidot minēto sistēmu, kvalifikācijas darba mērķis ir tā autoram gūt praktisku pieredzi dažādos programmatūras izstrādes posmos – prasību apkopošanā, projektēšanā, programmēšanā, testēšanā. Konkrētāk – autors vēlas izmēģināt spējo izstrādes pieeju, prasību apkopošanu lietotājistāstos, gūt pieredzi tīmekļa vietnes veidošanā, kas spēj pielāgoties dažādām izšķirtspējām, automātisku vienībtestu rakstīšanā.

Kvalifikācijas darbs sastāv no vairākām nodaļām.

- Nodaļā “**PROGRAMMATŪRAS PRASĪBU APRAKSTS**” aprakstītas izstrādājamajai sistēmai izvirzītās prasības lietotājistāstu formā.
- Nodaļā “**PROGRAMMATŪRAS PROJEKTĒJUMA APRAKSTS**” aprakstīta izstrādātās sistēmas uzbūve.
- Nodaļā “**TESTĒŠANAS DOKUMENTĀCIJA**” aprakstīta sistēmas testēšana, akcepttesti un

dokumentēta veiktā testēšana.

- Nodaļā “**PROJEKTA ORGANIZĀCIJA**” aprakstīta programmatūras izstrādes pieeja.
- Nodaļā “**KVALITĀTES NODROŠINĀŠANA**” aprakstīti procesi izstrādātā programmaprodukta kvalitātes nodrošināšanai.
- Nodaļā “**KONFIGURĀCIJU PĀRVALDĪBA**” aprakstīta pirmkoda un datu konfigurācijas saglabāšana.
- Nodaļā “**DARBIETILPĪBAS NOVĒRTĒJUMS**” novērtēta izstrādātā programmaprodukta darbietilpība.

1. PROGRAMMATŪRAS PRASĪBU APRAKSTS

1.1. Ievads

1.1.1. Nolūks

Nodaļas nolūks ir apkopot funkcionālās, drošības, un lietojamības prasības sistēmai lietotājistāstu formā. Katram lietotājistāstam uzskaitīti arī atbilstošie akceptēšanas kritēriji, kuriem ir jāizpildās, lai varētu uzskatīt, ka lietotājistāsts ir pabeigts. Šajā dokumentā aprakstītie lietotājistāsti atbilst sistēmai pēc pēdējās izstrādes iterācijas.

1.1.2. Darbības sfēra

Sistēmas, kuras prasības tiek apkopotas, galvenais uzdevums ir ļaut tās lietotājiem veidot uzdevumus, nodot tos citiem sistēmas lietotājiem, izmantojot komentāru pievienošanu uzdevumiem, veidot sarunu par uzdevuma izpildes gaitu.

1.1.3. Saistība ar citiem dokumentiem

Uz šajā nodaļā aprakstīto prasību pamata, veidotas nodaļas “2. Programmatūras prasību apraksts” un “3. Testēšanas dokumentācija”.

1.2. Lietotājistāsti

Lietotājistāsti (un akceptēšanas kritēriji) ir rakstīti pirmajā personā no sistēmas lietotāja skatpunkta, norādot kāds lietotāju tips(-i) ar šo lietotājistāstu tiek aprakstīts(-i). Cieši saistīta funkcionalitāte tiek sagrupēta moduļos.

Sistēmā tiek izdalīti trīs lietotāju tipi:

- **“guest”** (“viesis”) – lietotājs, kas sistēmā nav pieteicies;
- **“user”** (“parastais lietotājs”) – parastais lietotāju tips, kas ir pieteicies sistēmā;
- **“admin”** (“administrators”) – sistēmas administrators, kas ir pieteicies sistēmā.

1.2.1. Moduļa “Pieteikšanās, atteikšanās” lietotājstāsti

Modulī “Pieteikšanās, atteikšanās” apkopotas prasības sistēmai, kas saistītas ar lietotāju pilnvarošanu. Galvenās funkcijas ir:

- pieteikšanās sistēmā, izmantojot *Facebook*, *Google*, *LinkedIn* vai *Twitter* kontus;
- lietotāja vietējā (*native*) sistēmas konta reģistrēšana;
- lietotāja pieteikšanās sistēmā ar vietējo (*native*) kontu;
- vienreiz lietojama pieteikšanās saites nosūtīšana uz lietotāja e-pastu, ja viņš ir aizmirsis savu paroli;
- atteikšanās no darba sistēmā.

Identifikators	Lietotājstāsts	Akceptēšanas kritēriji
auth_providers _sign_in_first	As a guest who doesn't have a user account , I want to sign in using my <i>Facebook</i> , <i>Google</i> or <i>LinkedIn</i> accounts.	<ul style="list-style-type: none"> Clicking a link takes me to third party identity provider (TPIP) site, where I can accept or decline handing over my information to the system. <ul style="list-style-type: none"> If I don't allow TPIP to hand over my data to the system, an error message is displayed. If I allow to hand over my data, a verified user account (type is user) is created for me, using my email address and name from TPIP and I am signed in to the system.
auth_providers _sign_in_again	As a guest who has signed an account , I want to sign in using my <i>Facebook</i> , <i>Google</i> or <i>LinkedIn</i> accounts.	<ul style="list-style-type: none"> Clicking a link for one of the TPIP signs me in the system. If I have accounts at multiple TPIP (e.g. <i>Facebook</i> and <i>Google</i>) with the same email address, they all are linked to the same account in the system (i.e. I can sign in with <i>Facebook</i> or <i>Google</i>).
auth_twitter_fir st	As a guest , I want to sign in using my <i>Twitter</i> account.	<ul style="list-style-type: none"> Clicking a sign in link takes me to <i>Twitter</i>. <ul style="list-style-type: none"> If I don't allow <i>Twitter</i> to hand over my data to the system, an error message is displayed. If I allow <i>Twitter</i> to hand over my data, a form is displayed asking me to enter my email (because <i>Twitter</i> does not hand my email address over to the system automatically). If I enter an invalid email address or an email address that is already in use, an error message is displayed and I can enter my email address again. After I enter a correct email address, an unverified account (type is user) is created and

Identifikators	Lietotājstāsts	Akceptēšanas kritēriji
		<p>an email is sent to me with a verification link.</p> <ul style="list-style-type: none"> After clicking the link, my account is verified and I am signed in the system.
auth_twitter_again	As a guest that has already signed in with Twitter before , I want to sign in using my <i>Twitter</i> account.	<ul style="list-style-type: none"> Clicking the <i>Twitter</i> sign in link signs me in the system.
auth_native_sign_up_step1	As a guest , I want to sign up.	<ul style="list-style-type: none"> I can enter my email address. If I enter an invalid email address or an email address that is already in use, an error message is displayed and I can enter my email address again. After entering a valid email address, an email with sign up link is sent to my email address.
auth_native_sign_up_step2	As a guest who has completed native sign up step 1 , I want to complete my sign up.	<ul style="list-style-type: none"> I can enter my name (optionally) and I am required to enter my password twice. My password must match the password strength requirements (see general_password_strength). In case invalid data was provided, appropriate errors are displayed and I can reenter my data. After entering valid data, a verified user account (type is user) is created, and I am signed in.
auth_native_sign_in	As a guest , I want to sign in using my native user account.	<ul style="list-style-type: none"> I can enter my email address and my password. In case invalid data was provided, appropriate error messages are displayed and I can reenter my data. After entering valid information, I am signed in.
auth_recovery	As a guest , I want to recover my account in case I have forgotten my	<ul style="list-style-type: none"> I can enter my email address. If provided email address is invalid or there are no users with given email address, an error

Identifikators	Lietotājstāsts	Akceptēšanas kritēriji
	password.	<p>message is displayed.</p> <ul style="list-style-type: none"> • If valid email address was provided, an email is sent to me with an instant sign in link. The • After clicking the link, I am signed in.
auth_native_auth_recovery	As a guest , I want to automatically receive recovery email after 3 failed sign in attempts.	<ul style="list-style-type: none"> • If user enters a correct email address but an incorrect password 3 times in a row (without any successful sign ins in between), a recovery email is sent to the user.
auth_welcome_email	As a guest who signs in for the first time , I want to receive an email with basic information about the system.	<ul style="list-style-type: none"> • The first time I sign in with <i>Facebook</i>, <i>Linkedin</i>, or <i>Google</i>, I should receive this email. • After I am signed in with <i>Twitter</i> for the first time, I should receive this email. • After I complete the native sign up step 2, I should receive this email. • The email should give me direct links for <ul style="list-style-type: none"> ◦ creating a new task, ◦ editing my profile, ◦ sending contact messages.
auth_sign_out	As a user or admin I can sign out.	<ul style="list-style-type: none"> • Clicking a “sign out” link, signs me out of the system.

1.2.2. Moduļa “Profils” lietotājstāsti

Modulī “Profils” apkopotas funkcijas, kas ļauj lietotājam, kas ir pieteicies, mainīt savus datus.

Identifikators	Lietotājstāsts	Akceptēšanas kritēriji
profile_change_name	As a user or admin , I want to change my name.	<ul style="list-style-type: none"> I can change the name associated with my user account (0 to 255 characters long). I can delete (leave the name field empty).
profile_change_password	As a user or admin , I want to change my password.	<ul style="list-style-type: none"> I can enter my new password twice (for verification). <ul style="list-style-type: none"> If the entered passwords don't match or the passwords don't pass the password strength requirements (see general_password_strength), error messages are displayed. Otherwise, my password is changed.

1.2.3. Moduļa “Kontakti” lietotājstāsti

Modulī “kontakti” apkopotas funkcijas, kas ļauj lietotājiem nosūtīt ziņojumus sistēmas administratoriem.

Identifikators	Lietotājstāsts	Akceptēšanas kritēriji
contact_send_message	As a guest, user or admin , I want to send a message to the administrators.	<ul style="list-style-type: none"> I can enter <ul style="list-style-type: none"> my email address (optional, 0 to 255 characters long), my name (optional, 0 to 255 characters long), the message (mandatory, 5 to 4000 characters long). If I am signed in (user or admin), the email address and name should be set automatically.
contact_forward_to_email	As an admin , I want to receive all contact messages to an email address (contact@taskgps.com).	<ul style="list-style-type: none"> If a new, valid message is sent via the contact form, an email should be sent to contact@taskgps.com. The email must contain: <ul style="list-style-type: none"> sender's email address, sender's name,

Identifikators	Lietotājstāsts	Akceptēšanas kritēriji
		<ul style="list-style-type: none"> the sent message.

1.2.4. Moduļa “Administrācija” lietotājstāsti

Modulī “Administrācija” apkopotas funkcijas, kas pieejamas tikai sistēmas administratoriem.

Identifikators	Lietotājstāsts	Akceptēšanas kritēriji
admin_users	As an admin , I want to see a list of all users.	<ul style="list-style-type: none"> I want to be able to see information about all users. For each user, I want to see: <ul style="list-style-type: none"> when his account was created; his email address; his name; user type (admin or user); all TPIP he has authorized with. I want to be able to search through the list in user name and email fields. I want to be able to change the sorting order of the list by date created, email and name. The list must be split into pages, 50 users per page.
admin_contact_messages	As an admin , I want to see a list of all contact form messages.	<ul style="list-style-type: none"> I want to see all received contact messages. For each message, I want to see: <ul style="list-style-type: none"> when the message was sent; sender's email address; sender's name address; sender's IP address; the message. I want to be able to search through the list in the message, sender's email, name and IP fields. I want to be able to change the sorting order of

Identifikators	Lietotājstāsts	Akceptēšanas kritēriji
		<p>the list by date sent, email, name and IP address.</p> <ul style="list-style-type: none"> The list must be split into pages, 50 messages per page.

1.2.5. Moduļa “Uzdevumi” lietotājstāsti

Modulī “Uzdevumi” apkopotas funkcijas, kas saistītas ar uzdevumiem. Galvenās no tām:

- jaunu uzdevumu izveidošana;
- eksistējošu uzdevumu labošana, dzēšana;
- izveidoto un saņemto uzdevumu apskatīšana;
- komentāru pievienošana uzdevumiem.

Identifikators	Lietotājstāsts	Akceptēšanas kritēriji
task_new	As a user or admin , I want to create new tasks.	<ul style="list-style-type: none"> I must enter email address of the person I want to assign the task to (the doer). The user must be registered in the system. I must enter the title of the task (between 3 and 255 characters long). I can enter a description (0 to 4000 characters). I can enter up to 15 tags, each tag between 1 and 255 characters long. I must enter valid deadline date in format YYYY-MM-DD (YYYY – year, MM – month, DD – day) and time in format HH:MM (HH – hours, MM – minutes). If entered data is not valid, error messages are displayed. If all entered data is valid, a new task is created. It is visible to me and the person I assigned the task to.

Identifikators	Lietotājistāsts	Akceptēšanas kritēriji
task_edit_creat or	As a user or admin who is the creator of a task , I want to edit the task.	<ul style="list-style-type: none"> • I can edit the tasks I have created. • I can edit all task data just like when creating a task. • I can mark the task as done. • I can delete the task.
task_edit_doer	As a user or admin who is the doer of a task , I want to edit the task.	<ul style="list-style-type: none"> • I can edit the tasks I am the doer of. • I can edit all task data, except the “doer” field. • I can mark the task as done. • I can delete the task.
task_list	As a user or admin , I want to see a list of all tasks that I have created or that where assigned to me.	<ul style="list-style-type: none"> • I want to see all tasks that I have created or that I am the doer of. • For each task, I want to see: <ul style="list-style-type: none"> ◦ when the tas was created; ◦ the title; ◦ the creator; ◦ the doer; ◦ the deadline. • I want to be able to search through the list in the title, description, name and email address of the creator, name and email address of the doer, comments and tasks. • I want to be able to change the sorting order of the list by date created, title, owner email, doer email and deadline. • The list must be split into pages, 25 tasks per page.
task_view	As a user or admin , I want to view a task with it's full details and	<ul style="list-style-type: none"> • I can view only those tasks, that I am the owner of or the doer of. • I can see the task title, description, date created,

Identifikators	Lietotājistāsts	Akceptēšanas kritēriji
	comments.	deadline date, owner, doer, tags and all comments.
task_add_comment	As a user or admin , I want to add a comment to a task.	<ul style="list-style-type: none"> I can add comments only to tasks, that I am the owner of or the doer of. I can enter the text of the comment (between 1 and 4000 characters).

1.2.6. Vispārīgas prasības sistēmai

Šajā apakšnodaļā apkopotas prasības, kas attiecas uz visu sistēmu, ne tikai konkrētu moduli.

Identifikators	Lietotājistāsts	Akceptēšanas kritēriji
general_password_strength	As a user or admin , I want my account to be protected by a reasonably strong password.	<ul style="list-style-type: none"> The minimum length of password is 8 characters.
general_emails	As a guest and user , I want receive all emails in HTML and plain text.	<ul style="list-style-type: none"> All emails sent by the system to guests and users are available in HTML and plain text.
general_access_control	As a user or admin , I want resources that <i>should be</i> only available to me, are available only to me.	<ul style="list-style-type: none"> Functions that are accessible only by admins should be tested by opening them with a user account to verify. Objects that are accessible only by some users should be tested by opening them with other users to verify.
general_mobile_usability	As a guest or user , I want to use the system with my mobile device.	<ul style="list-style-type: none"> Modules available to guests and admins should be easily usable on screens with a small resolution – 320×480px and slightly larger. It is acceptable, that some content might be hidden on these devices to gain better usability.

1.3. Nefunkcionālās prasības

1.3.1. Drošības prasības

Sistēmai jānodrošina aizsardzība pret tipiskām tīmekļa vietņu ievainojamībām – *SQL* injekcijām, *XSS* un *CSRF* ievainojamībām.

1.3.2. Lietojamības prasības

Sistēmas lietotāja saskarnēm jābūt ērti lietojamām galvenokārt klēpj datoru un galddatoru ekrāna izšķirtspējām (1024×786px un augstāka izšķirtspēja). Saskarnēm, kurām lietotāju stāstos norādīts, ka tām jāpielāgojas arī mazākām (viedtelefonu) izšķirtspējām, jābūt ērti lietojamām (robežās no 480×320px līdz 480×800px).

2. PROGRAMMATŪRAS PROJEKTĒJUMA APRAKSTS

2.1. Ievads

2.1.1. Nolūks

Nodaļas mērķis ir sniegt sistēmas projektējuma aprakstu, kas atbilst izstrādātajai sistēmai pēc pēdējās iterācijas.

2.1.2. Saistība ar citiem dokumentiem

Projektējums atbilsts nodaļā “1. Programmatūras prasību apraksts” aprakstītajām prasībām.

2.2. Dekompozīcijas apraksts

Sistēma izstrādāta, balstoties uz *FuelPHP* ietvaru, kas ļauj veidot sistēmas pēc *Model-View-Controller* arhitektūras.

- *View* (skati) – datu reprezentācijas slānis, kas saņemtos datus no kontroliera attēlo par HTML, ko nosūtīt lietotājam;
- *Controller* (kontrolieris) – atbild par pareizo modeļa servisu izsaukšanu un saņemto datu nodošanu skatam;
- *Model* (modelis) – veidots no divu tipu objektiem:
 - *Entity* (datu objekts) – saistītu datu kopa;
 - *Service* (serviss) – objekts, kas atbildīgs par modeļa datu objektu manipulēšanu vajadzīgajā veidā.

2.2.1. Datu dekompozīcija

Šajā nodaļā sīkāk tiks aprakstīta katra tabula, dots katra lauka nosaukums, datu tips, apraksts. Izmantoto apzīmējumu skaidrojums:

- PK – *primary key* – primāra atslēga;

- FK – *foreign key* – ārējā atslēga;
- AI – *auto increment* – INT tipa lauks, kura vērtība tiek automātiski piešķirta jauna ieraksta izveidošanas brīdī, lauka vērtības netiek izmantotas vairāk kā vienu reizi;
- INT – datu tips – vesels skaitlis;
- VARCHAR(X) – datu tips, kas var saturēt līdz X simboliem garu simbolu virkni;
- ENUM(X,Y,Z) – datu tips, kas pieļauj tikai vienu no vērtībām X, Y vai Z;
- TEXT – datu tips – teksts, bez garuma ierobežojuma;
- TIMESTAMP – *UNIX timestamp* – vesels skaitlis, kas apzīmē cik daudz sekunžu ir pagājis kopš 1970-01-01 00:00:00 pēc UTC laika zonas.

2.2.1.1. Tabula “user”

Tabula satur reģistrētu lietotāju datus (tipi *parastais lietotājs* un *administrators*).

Lauka nosaukums	Datu tips	Īpašības	Apraksts
id	INT	PK, AI	Lietotāja ID.
email	VARCHAR(255)		Lietotāja e-pasta adrese.
name	VARCHAR(255)		Lietotāja vārds.
password	VARCHAR(255)		Lietotāja parole šifrētā veidā. Lauks var būt NULL gadījumā, ja lietotājs ir reģistrēts sistēmā caur TPIP.
group	INT		Grupas ID, kurai pieder lietotājs. <ul style="list-style-type: none"> • 10 – parastais lietotājs; • 100 – administrators.
login_hash	VARCHAR(255)		Vērtība, kas ir unikāla katrai lietotāja pieteikšanās reizei.
sign_up_code	VARCHAR(40)		Kods, kas lietotājam tiek nosūtīts e-pastā, pēc pirmā vietējā (<i>native</i>) konta reģistrēšanas soļa.
signed_up_at	INT	TIMESTAMP	Laiks, kad lietotājs pabeidza sava vietējā (<i>native</i>) konta reģistrēšanu.

Lauka nosaukums	Datu tips	Īpašības	Apraksts
			Ja reģistrēšana nav pabeigta, tad lauka vērtība ir 0.
instant_sign_in_code	VARCHAR(40)		Kods, kas lietotājam tiek nosūtīts e-pastā saites veidā. Uzspiežot uz saites, lietotājs tiek automātiski pieteikts sistēmā.
instant_sign_in_at	INT	TIMESTAMP	Laiks, kad pieteikšanās kods tika uzģenerēts. Šis laiks ir nepieciešams, jo kodam ir noteikts derīguma laiks.
verify_email_code	VARCHAR(40)		Kods, kas lietotājam tiek nosūtīts e-pastā saites veidā, lai apstiprinātu lietotāja e-pasta adresi (attiecas tikai uz TPIP <i>Twitter</i>).
verified_email_at	INT	TIMESTAMP	Laiks, kad lietotājs apstiprināja savu e-pasta adresi. Ja konts nav apstiprināts, tad šajā laukā ir 0.
created_at	INT	TIMESTAMP	Laiks, kad lietotāja konts tika izveidots.
updated_at	INT	TIMESTAMP	Laiks, kad pēdējoreiz tika mainīti lietotāja dati.
enabled_at	INT	TIMESTAMP	Laiks, kopš kura lietotājs drīkst pieteikties sistēmā. Ja lauka vērtība ir 0, tad lietotājs sistēmā nevar pieteikties.
failed_sign_in_attempts	INT		Neveiksmīgu vietējās pieteikšanās mēģinājumu skaits ar lietotāja e-pasta adresi, kopš pēdējā veiksmīgā pieteikšanās mēģinājuma.

2.2.1.2. Tabula “user_authentication”

Tabula satur datus par lietotāja TPIP – trešo pušu identitātēm. Šī tabula ir saistīta ar tabulu “user”,

jo katram lietotājam var būt no 0 līdz 4 dažādām TPIP identitātēm.

Lauka nosaukums	Datu tips	Īpašības	Apraksts
id	INT	PK, AI	Lietotāja identitātes ID.
user_id	INT	FK	Lietotāja ID, kuram ir piesaistīta šī identitāte.
provider	VARCHAR(50)		TPIP nosaukums (piem. <i>Facebook</i> , <i>LinkedIn</i>).
uid	VARCHAR(255)		Lietotāja identitātes unikāls identifikators, ko identitātei piešķir TPIP.
access_token	VARCHAR(255)		<i>OAuth</i> protokolā noteika pilnvara, kas ļauj sistēmai izpildīt darbības TPIP sistēmā lietotāja vārdā (piem. rakstīt ziņu <i>Facebook</i> lietotāja “sienā”).
expires	INT	TIMESTAMP	Laiks, līdz kuram piešķirtā pilnvara ir derīga.
refresh_token	VARCHAR(255)		<i>OAuth</i> protokolā noteika pilnvara, kas sistēmai ļauj iegūt jaunu <i>access_token</i> .
secret	VARCHAR(255)		Paraksts, ko <i>OAuth</i> protokolā izmanto, “ <i>access_token</i> ” pārsūtīšanas brīdī.
created_at	INT	TIMESTAMP	Laiks, kad lietotājs ir pirmo reizi pieteicies ar šo TPIP.
updated_at	INT	TIMESTAMP	Laiks, kad lietotājs pēdējo reizi ir pieteicies, izmantojot šo TPIP.

2.2.1.3. Tabula “contact_message”

Tabula satur datus par saņemtajiem ziņojumiem no kontaktu moduļa. Šī tabula ir saistīta ar tabulu “*user*” – katram ziņojumam var būt viens vai neviens reģistrēts lietotājs, kurš šo ziņu ir nosūtījis.

Lauka nosaukums	Datu tips	Īpašības	Apraksts
id	INT	PK, AI	Ziņojuma ID.
user_id	INT	FK	Lietotāja ID, kurš ziņojumu nosūtīja. Gadījumā, ja ziņojumu nosūta

Lauka nosaukums	Datu tips	Īpašības	Apraksts
			neregistrēts lietotājs, šis lauks pieņem vērtību <i>NULL</i> .
email	VARCHAR(255)		Ziņojuma nosūtītāja e-pasta adrese.
name	VARCHAR(255)		Ziņojuma nosūtītāja vārds.
ip	VARCHAR(15)		Ziņojuma nosūtītāja IP adrese.
message	TEXT		Ziņojuma teksts.
created_at	INT	TIMESTAMP	Ziņojuma izveidošanas laiks.

2.2.1.4. Tabula “task”

Tabula satur datus par uzdevumiem. Šī tabula ir saistīta ar tabulu:

- “user” (*owner_id*) – katram uzdevumam ir tieši viens izveidotājs, bet katrs lietotājs var izveidot vairākus uzdevumus;
- “user” (*doer_id*) – katram uzdevumam ir tieši viens izpildītājs, bet katram lietotājam var būt piešķirti vairāki uzdevumi.

Lauka nosaukums	Datu tips	Īpašības	Apraksts
id	INT	PK, AI	Uzdevuma ID.
owner_id	INT	FK	Lietotāja ID, kurš ir uzdevuma izveidotājs.
doer_id	INT	FK	Lietotāja ID, kurš ir uzdevuma izpildītājs.
title	VARCHAR(255)		Uzdevuma virsraksts.
status	ENUM(in_progress, done, deleted)		Uzdevuma pašreizējais statuss. <ul style="list-style-type: none"> • “in progress” – uzdevums ir izveidots, bet vēl nav izpildīts; • “done” – uzdevums ir izpildīts; • “deleted” – uzdevums ir izdzēsts
description	TEXT		Uzdevuma apraksts.
created_at	INT	TIMESTAMP	Uzdevuma izveidošanas laiks.
deadline_at	INT	TIMESTAMP	Uzdevuma izpildes termiņš.
done_at	INT	TIMESTAMP	Laiks, kad uzdevums tika izpildīts. Ja

Lauka nosaukums	Datu tips	Īpašības	Apraksts
			uzdevums nav izpildīts, tad šis lauks ir 0.

2.2.1.5. Tabula “task_comment”

Tabula satur datus par komentāriem, kuri pievienoti uzdevumam. Šī tabula ir saistīta ar:

- tabulu “user” – katram komentāram ir tieši viens tā rakstītājs, bet lietotājs var rakstīt vairākus komentārus;
- tabulu “task” – katrs komentārs pieder tieši vienam uzdevumam, bet katram uzdevumam var būt vairāki komentāri.

Lauka nosaukums	Datu tips	Īpašības	Apraksts
id	INT	PK, AI	Komentāra ID.
user_id	INT	FK	Lietotāja ID, kurš šo komentāru rakstīja.
task_id	INT	FK	Uzdevuma ID, kuram šis komentārs ir pievienots.
comment	TEXT		Komentāra teksts.
created_at	INT	TIMESTAMP	Komentāra izveidošanas laiks.

2.2.1.6. Tabula “task_tag”

Tabula satur datus par birkām, kas pievienotas uzdevumam. Šī tabula ir saistīta ar tabulu “task” – katra birka pieder tieši vienam uzdevumam, bet katram uzdevumam var būt vairākas birkas.

Lauka nosaukums	Datu tips	Īpašības	Apraksts
id	INT	PK, AI	Birkas ID.
task_id	INT	FK	Uzdevuma ID, kuram šī birka ir pielikta.
tag	VARCHAR(255)		Birkas teksts.

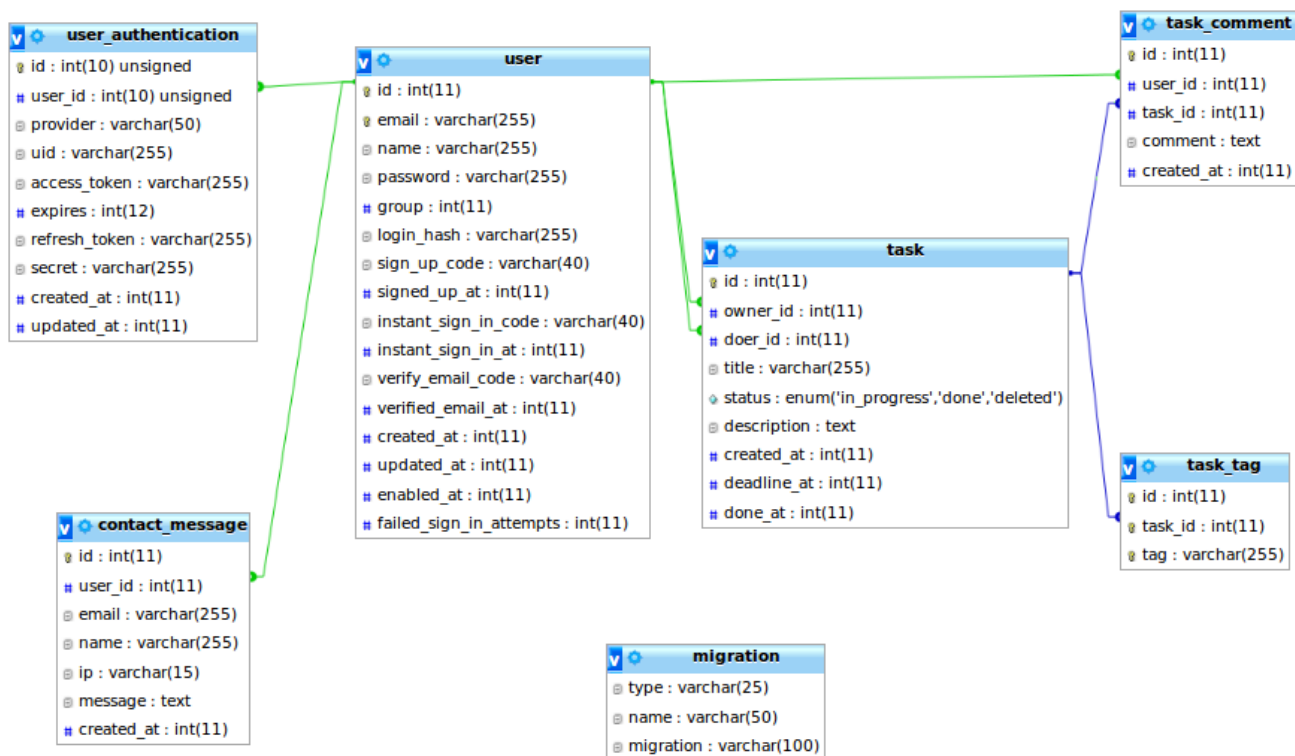
2.2.1.7. Tabula “migration”

Tabula satur datus par datubāzes stāvokli. Tā satur izpildīto migrāciju skriptu sarakstu (skriptus, kas maina datubāzes struktūru). Šī tabula ir nepieciešama ietvara *FuelPHP* migrāciju rīkam un šis rīks arī pats šo tabulu izveido.

Lauka nosaukums	Datu tips	Īpašības	Apraksts
type	VARCHAR(25)		Migrācijas tips (<i>package</i> – <i>FuelPHP</i> papildinājuma, <i>app</i> – sistēmas migrācija).
name	VARCHAR(50)		Migrācijas nosaukums.
migration	VARCHAR(100)		Migrācijas faila nosaukums, kas reizē kalpo kā migrācijas identifikators.

2.3. Atkarības apraksts

2.3.1. Datu atkarības – fiziskais datubāzes modelis



Attēls 2.1: Datubāzes realizācijas modelis

Gan zilās, gan zaļās bultiņas norāda viens pret daudz attiecības starp tabulām ar lielāko pusluku *daudz* galā.

Tabula “*migration*” nav saistīt ar citām, jo tā glabā informāciju par pašreizējo datubāzes stāvokli

(Šī tabula ir nepieciešama tikai ietvara *FuelPHP* rīkam, kas atbild par datubāzes struktūras izmainīšanu).

3. *TESTĒŠANAS DOKUMENTĀCIJA*

3.1. *Ievads*

Sistēmas testēšana notika ar automātiskiem programmkoda vienībtestiem, kā arī ar sistēmas akcepttestiem. Akcepttestēšana notika katras iterācijas beigās.

3.2. *Programmkoda vienībtestēšana*

PHP programmkoda rakstīšanas gaitā tika rakstīti arī atbilstoši automātiski vienībtesti, izmantojot testēšanas ietvaru *PHPUnit*. Šo vienībtestu uzdevums bija pārliecināties, ka uzrakstītās programmatūras vienības (kas pārsvarā bija klašu metodes) strādā tā, kā paredzēts (dokumentēts programmkoda komentāros) gan normālos gadījumos, gan robežgadījumos, gan kļūdu gadījumos. Šie testi tika veidoti pēc *baltās kastes* testēšanas principiem. Šie testi tika aktīvi izpildīti programmkoda rakstīšanas gaitā.

Ar šiem vienībtestiem galvenokārt tika testēta MVC modeļa daļa, kā arī paplašinājumi *FuelPHP* ietvara funkcionalitātei, bet netika testēti MVC kontrolieri un skati. Autors izmēģināja arī testu vadītās izstrādes pieeju (*TDD*), taču aktīvi to nepielietoja, jo tas apgrūtināja izstrādes gaitu.

Nodaļā “*PIELIKUMI*” var apskatīt vienu testpiemēru.

3.3. *Sistēmas akcepttestēšana*

3.3.1. *Ievads*

Akcepttesti tika definēti katras iterācijas gaitā. Katras iterācijas beigās bija pilnīgs akcepttestu komplekts, pie tam iterācijas beigās sistēmai bija jāiztur visi akcepttestu piemēri. Akcepttesti tika veidoti pēc *melnās kastes* testēšanas principiem un tika izpildīti gan iterācijas gaitā (pabeidzot kaut kādu funkcionalitātes daļu), gan vēlreiz iterācijas beigās.

3.3.2. *Testējamās raksturiezīmes*

- Tiek testēti visi lietotārstāsti un akceptēšanas kritēriji ar korektiem ievaddatiem.
- Tiek pieņemts, ka visi dažādie nekorekto ievaddatu gadījumi tiek notestēti jau programmkoda

vienībtestos, tāpēc akcepttestos katrā ievadlaukā tiek ievadīts tikai viena veida nekorektu ievaddatu piemērs (ja tāds vispār ir), lai pārliecinātos, ka lietotāja saskarnē korekti parādās kļūdu paziņojumi.

- Tiek testētas drošības prasības sistēmai.
- Tiek testētas lietojamības prasības sistēmai.

3.3.3. Testēšanas žurnāls

Šajā nodaļā ir uzskaitīti visi akcepttestu piemēri, sagaidāmie rezultāti un reālie rezultāti pēc pēdējās iterācijas beigām.

3.3.3.1. Modulis “Pieteikšanās, atteikšanās”

Saistītais(-ie) lietotājstāsts(-i)	Priekšnosacījumi	Testpiemērs	Sagaidāmie rezultāti	Statuss
auth_providers _sign_in_first auth_welcome_email, general_emails	I am a guest located on the index page.	Click on “ <i>Facebook</i> ” icon and accept handing over my data.	I am signed in the system and index page is displayed. I received a “welcome” email. Email should be in HTML and plain text.	Correct.
auth_providers _sign_in_first	I am a guest located on the index page.	Click on “ <i>Facebook</i> ” icon and decline handing over my data.	I am not signed in to the system and an error message is displayed.	Correct.
auth_twitter_first, general_email, auth_welcome_email	I am a guest located on the index page.	Click on “ <i>Google</i> ” icon and accept handing over my data.	I am signed in the system and index page is displayed. I received a “welcome” email. Email should be in HTML and plain text.	Correct.

Saistītais(-ie) lietotājs(-i)	Priekšnosacījumi	Testpiemērs	Sagaidāmie rezultāti	Statuss
auth_twitter_fir st, general_email, auth_welcome_email	I am a guest located on the index page.	Click on “ <i>LinkedIn</i> ” icon and accept handing over my data.	I am signed in the system and index page is displayed. I received a “welcome” email. Email should be in HTML and plain text.	Correct.
auth_providers_si gn_in_again	I am a guest located on index page. I have previously signed in with <i>Facebook</i> . I also have a <i>Google</i> account with same email but different name.	I click on the “ <i>Google</i> ” icon and accept to hand over data.	I am signed in the system. My name from <i>Facebook</i> account is displayed, therefore I have signed in to the same account.	Correct.
auth_twitter_fir st	I am a guest located on the index page.	Click on “ <i>Twitter</i> ” icon and accept handing over my data.	A form requesting my email address is displayed.	Correct.
auth_twitter_fir st	Form asking for my email is displayed.	I enter an invalid email address “abcd@example”.	An error message is displayed.	Correct.
auth_twitter_fir st	Form asking for my email is displayed.	I enter a valid email address “example@taskgps.com”.	A message that a verification email was sent. I received an email with the verification link.	Correct.
auth_twitter_fir st, general_email,	A verification email was sent to me.	I click the verification link.	My account is verified. I am signed in. Index page is displayed. I received a “welcome” email. Email	Correct.

Saistītais(-ie) lietotājs(-i)	Priekšnosacījumi	Testpiemērs	Sagaidāmie rezultāti	Statuss
auth_welcome_email			should be in HTML and plain text.	
auth_twitter_again	I am a guest who has signed in with Twitter before.	I click on the “ <i>Twitter</i> ” icon.	I am signed in. The index page is displayed.	Correct.
auth_native_sign_up_step1	I am a guest located on the index page.	I enter an invalid email “abcd@example”.	An error message is displayed.	Correct.
auth_native_sign_up_step1	I am a guest located on the index page.	I enter a valid email address “example@taskgps.com”.	A success message is displayed. A “sign up” email is sent to me.	Correct.
auth_native_sign_up_step2	I have received a “sign up” email.	I click the sign up link.	I am taken to a form, where I can enter my name and password.	Correct.
auth_native_sign_up_step2	Form asking me for my email and password is displayed.	I leave the name field empty, and enter an invalid password “1234” (twice).	An error message is displayed.	Correct.
auth_native_sign_up_step2, general_password_strength, general_email, auth_welcome_email	Form asking me for my email and password is displayed.	I leave the name field empty, and enter a valid password “\$p@w0rd#” (twice).	I am signed in to the system. The index page is displayed. I received a “welcome” email. Email should be in HTML and plain text.	Correct.
auth_native_sign	I am a guest	I enter my email address	An error message is	Correct.

Saistītais(-ie) lietotājs(-i)	Priekšnosacījumi	Testpiemērs	Sagaidāmie rezultāti	Statuss
_in	located on the index page.	“example_user@taskgps.com” and an incorrect password “12345678”.	displayed.	
auth_native_sign_in	I am a guest located on the index page.	I enter my email address “example_user@taskgps.com” and a correct password “321p@wrld123”.	I am signed in to the system. The index page is displayed.	Correct.
auth_recovery	I am a guest located on the index page.	I enter an invalid email address “abcd@example”.	An error message is displayed.	Correct.
auth_recovery, general_email	I am a guest located on the index page.	I enter a valid email address “example_user@taskgps.com”.	A success message is displayed. An email containing an “instant sign in” link is sent to me. Email should be in HTML and plain text.	Correct.
auth_recovery	I am a guest who has received an instant “sign in” email.	I click the “instant sign in” link in the email.	I am signed in to the system.	Correct.
auth_native_auto_recovery	I am guest located on the index page.	I enter my email address in the sign in form, and an incorrect password “12345678”. I repeat this 3 times.	I should automatically receive an “instant sign in” email.	Correct.
auth_sign_out	I am a signed in user.	I click the “sign out” link in top navigation.	I am signed out of the system. Index page is displayed.	Correct.
general_mobile_u	I am a guest		The top navigation is	Correct.

Saistītais(-ie) lietotājs(-i)	Priekšnosacījumi	Testpiemērs	Sagaidāmie rezultāti	Statuss
sability	located on the index page. I am using a smart phone.		collapsed and can be expanded by clicking on it. The left side column of the page is hidden. The input fields are big enough to click them.	

3.3.3.2. Modulis “Profils”

Saistītais(-ie) lietotājs(-i)	Priekšnosacījumi	Testpiemērs	Sagaidāmie rezultāti	Statuss
profile_change_name	I am a user , located on the profile page.	I type “John Doe” in the name field and leave other fields blank and submit.	A success message is displayed. The name field now contains “John Doe”.	Correct.
profile_change_name	I am a user , located on the profile page.	I delete the contents of the name field and submit.	A success message is displayed. The name field now is empty.	Correct.
profile_change_password general_password_strength	I am a user , located on the profile page.	I enter “12345678” in password field and the password “abcdefgh” in the “repeat password” field and submit.	An error message is displayed.	Correct.
profile_change_password general_password_strength	I am a user , located on the profile page.	I enter “pa45word#\$%” and the same in the “repeat password” field and submit.	A success message is displayed. My password was changed.	Correct.
general_mobile_usability	I am a user located on the profile page. I am using a		The top navigation is collapsed and can be expanded by clicking on	Correct.

Saistītais(-ie) lietotājs(-i)	Priekšnosacījumi	Testpiemērs	Sagaidāmie rezultāti	Statuss
	smart phone.		it. The input fields are big enough to click them.	

3.3.3.3. Modulis “Kontakti”

Saistītais(-ie) lietotājs(-i)	Priekšnosacījumi	Testpiemērs	Sagaidāmie rezultāti	Statuss
contact_send_message	I am signed in as a user or admin.	I open the contact page.	The contact form “name” field has my name, and the “email” field has my email address.	Correct.
contact_send_message	I am a guest , located on the contact page.	I enter a message “ab” in the message field, that is too short and submit.	An error message is displayed.	Correct.
contact_send_message contact_forward_to_email	I am a guest located on the contact page.	I enter “John Doe” in the “name” field, “example@taskgps.com” in the email field and “Hello!” in the “message” field.	A success message is displayed. An email is sent to “contact@taskgps.com” containing the name, email address and message.	Correct.
general_mobile_usability	I am a user located on the contact page. I am using a smart phone.		The top navigation is collapsed and can be expanded by clicking on it. The input fields are big enough to click them.	Correct.

3.3.3.4. Modulis “Administrācija”

Saistītais(-ie) lietotājs(-i)	Priekšnosacījumi	Testpiemērs	Sagaidāmie rezultāti	Statuss
admin_users	I am an admin , and admin user list is open.	I click on the column “name”.	All users are displayed ordered by their name alphabetically.	Correct.
admin_users	I am an admin , and admin user list is open.	I type a search query “example_user@” and submit the query.	Only one user should be displayed.	Correct.
admin_users general_access_control	I am a user .	I try to open the user list.	I am redirected to the index page.	Correct.
admin_contact_message	I am an admin .	I click on the “name” column.	All messages are displayed ordered by the senders name alphabetically.	Correct.
admin_contact_message general_access_control	I am a user .	I try to open the contact message list.	I am redirected to the index page.	Correct.

3.3.3.5. Modulis “Uzdevumi”

Saistītais(-ie) lietotājs(-i)	Priekšnosacījumi	Testpiemērs	Sagaidāmie rezultāti	Statuss
task_new	I am signed in as a user X and I am located at new task form.	I enter the email address of user Y in the “to” field; “Shopping” in the title field; “Buy bread.” in description; add tags	A new task is created. It is visible in the task list.	Correct.

Saistītais(-ie) lietotājs(tāsts(-i))	Priekšnosacījumi	Testpiemērs	Sagaidāmie rezultāti	Statuss
		“shopping” and “family” and set the deadline to today at 23:55 and submit.		
task_new		Same as above, but enter invalid email “abcd@example”, too short title “a” and type “xx” in the hour input field.	Task is not created; three errors are displayed.	Correct.
task_edit_creat or		I change the title to “Shopping (edited)” and submit.	A success message is displayed. The task title is edited.	Correct.
task_edit_creat or	I am signed in as a user X and I am located at task editing form for a task, that I have created.	I click the “mark as done” button.	A success message is displayed. The task is now considered as done. It is displayed with a green background in the task list.	Correct.
task_edit_creat or		I click the “delete” button. I answer “yes” to the confirmation.	The task is deleted. List of all tasks is displayed.	Correct.
task_edit_doer		I try to change the value of “to” field.	The “to” field is read- only, therefore can't be changed.	Correct.
task_edit_doer	I am signed in as a user Y , and have opened a task editing form, that has been assigned to me.	I edit the task description to “Buy bread and milk.” and submit.	A success message is displayed. The task description is edited.	Correct.
task_edit_doer		I click the “mark as done” button.	A success message is displayed. The task is now considered as done.	Correct.

Saistītais(-ie) lietotājs(-i)	Priekšnosacījumi	Testpiemērs	Sagaidāmie rezultāti	Statuss
			It is displayed with a green background in the task list.	
task_edit_doer		I click the “delete” button. I answer “yes” to the confirmation.	The task is deleted. List of all tasks is displayed.	Correct.
task_list	I am signed in as a user X and have opened the task list.	I enter “example task” in the search box and submit.	Only tasks matching the search query are displayed.	Correct.
task_list		I click on the “owner” link in table header.	All tasks are sorted in an alphabetical order.	Correct.
task_view	I am user , located at a task view.		I can see all the information about the task: “title”, “description”, “date created”, “deadline”, “date done at”, “owner”, “doer” and all comments.	Correct.
task_add_comment		I enter “It will be done today.” in the comment text area and submit.	A success message is displayed. The comment is added and can be seen in comment list.	Correct.
task_add_comment		I leave the comment box empty and submit.	An error message is displayed.	Correct.
general_access_control	I am a guest .	I try to open a task view, that I am not the owner of, nor the doer of (by manually editing the URL in navigation bar).	I am redirected back to index page.	Correct.

Saistītais(-ie) lietotājstāsts(-i)	Priekšnosacījumi	Testpiemērs	Sagaidāmie rezultāti	Statuss
general_access_control		I try to open a task edit, that I am not the owner of, nor the doer of (by manually editing the URL in navigation bar).	I am redirected back to index page.	
general_mobile_usability	I am a user located on the task list. I am using a smart phone.		Some fields of the table are hidden, so that I don't have to scroll horizontally.	Correct.
general_mobile_usability	I am a user located on the new task page. I am using a smart phone.		The input fields are big enough to click them.	Correct.

4. PROJEKTA ORGANIZĀCIJA

Sistēmas izstrāde notika pēc spējās izstrādes metodes. Sāukumā sistēmas prasības tika apkopotas vispārīgos lietotārstāstos (*epics [1]*), kuri pirms katras iterācijas tika sadalīti vairākos mazākos lietotārstāstos un tika aprakstīti sīkāk, tika definēti akceptēšanas kritēriji. Lietotāju stāstus rakstīja kvalifikācijas darba autors, bet tie tika saskaņoti ar pasūtītāju.

Sistēma tika izstrādāta trijās iterācijās. Katras iterācijas sākumā tika izvēlēti izpildāmie vispārīgie lietotārstāsti, tie tika aprakstīti sīkāk, tika definēti akceptēšanas kritēriji, tika uzrakstīts programmkods, programmkoda vienībtesti, akcepttestu scenāriji. Iterācijas beigās visiem akcepttestiem bija jāizpildās, lai iterācija tiktu pabeigta. Katrā iterācijā tika radīta notestēta, sistēmas versija, kas teorētiski bija gatava izmitināšanai produkcijas vidē, bet praktiski funkcionalitāte nebija pietiekami plaša, lai to darītu. Piemēram, pirmās iterācijas rezultātā bija radīta visa funkcionalitāte, kas saistīta ar pieteikšanos sistēmā, taču sistēmai, kurai piemīt tikai šāda funkcionalitāte, nav praktiska pielietojuma.

Iterācijas netika plānotas tā, lai tās būtu vienādas pēc laiktelpības, bet gan tā, lai katras iterācijas galā būtu pabeigta loģiska funkciju kopa (piemēram, viss, kas saistīts ar pieteikšanos sistēmā).

Tabula 4.1: sistēmas izstrādes iterācijas

Iterācija	Paveiktais
1	Modulis “Pieteikšanās, atteikšanās”.
2	Modulis “Profils”, “Administrācija”, “Kontakti”. <i>Piezīme: tika realizēti trīs moduļi, jo tie bija neliela apjoma.</i>
3	Modulis “Uzdevumi”.

5. KVALITĀTES NODROŠINĀŠANA

Katras iterācijas sākumā autors apkopoja prasības lietotājistāstos, kuri tika pārrunāti un apstiprināti no pasūtītāja puses, lai pārliecinātos, ka prasības apkopotas pareizi.

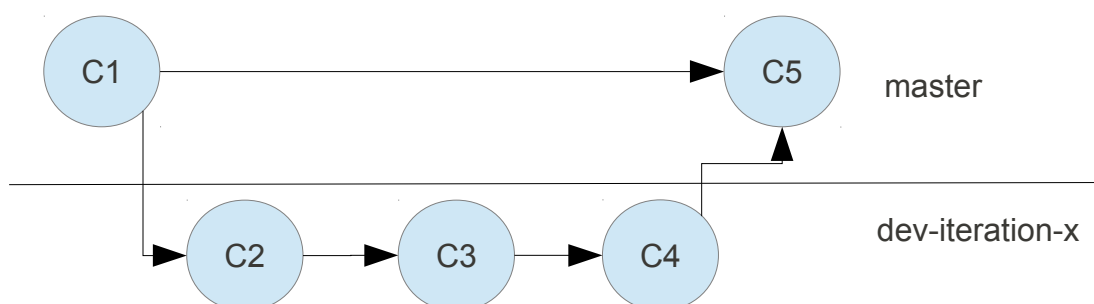
Programmkods valodā *PHP* tika rakstīts saskaņā ar izmantotā ietvara *FuelPHP* noteiktajam koda stila un noformējuma standartu [2]. Papildus šiem nosacījumiem, tika noteikts, ka katrā *PHP* faila sākumā komentāru blokā tiek minēts programkoda mērķis un tā autors (*docblock* birka *@author*). Katrai funkcijai un klases metodei tika noteikts minimālais komentāru komplekts – ir jāuzskaita visi parametri (*docblock* birka *@param*) ar atbilstošajiem datu tipiem, atgrieztā vērtība (*docblock* birka *@return*) un īss funkcijas mērķis. Katram klases mainīgajam tika noteikts minimālais komentāru komplekts – mainīgā tips (*docblock* birka *@var*) un īss mainīgā apraksts. Valodā *JavaScript* tika izmantots līdzīgs stils un komentāru komplekts.

Lielākajai daļai rakstītā programmkoda tika rakstīti automātiski vienībtesti ar testēšanas ietvaru *PHPUnit*. Pēc *PHPUnit* (izmantotā versija 3.7.14) koda pārklājuma ar vienībtestiem analīzes [3], pēc pēdējās iterācijas koda pārklājums bija aptuveni 70%. Papildus automātiskajiem koda vienībtestiem testiem, tika veikta arī sistēmas testēšana pēc akcepttestiem.

Pirms katras iterācijas beigām sistēma tika atrādīta pasūtītājam, lai pārliecinātos, ka realizēta funkcionalitāte atbilst pasūtītāja vēlmēm. Ja tā nebija, tad pirms iterācija tika pabeigta, tika veikti labojumi.

6. KONFIGURĀCIJU PĀRVALDĪBA

Izstrādātās sistēmas pirmkods tika glabāts versiju kontroles sistēmā *git*. Katrai iterācijai bija savs atzars (*branch*), kas tika nosaukts pēc principa *dev-iteration-X*. Katrs iterācijas atzars sastāv no vairākiem izmaiņu kopumiem (*commits*), katram no tiem ir aprakstošs ziņojums par to, kas tika izmainīts. Zarā (*branch*) *master* pēc katras iterācijas pabeigšanas tika iepludināta (*merged*) jaunākā, pabeigtā versija.



Attēls 6.1: pirmkoda organizēšana git repozitorijā

Datubāzes struktūra tika definēta migrāciju failos. Katrai migrācijas fails ir *PHP* skripti, kas izveido jaunās datubāzes tabulas. Katrai migrācijai ir arī apgrieztā funkcionalitāte, kas jaunās tabulas izdzēš. Šādā veidā tika iegūti skripti, ar kuriem var viegli kontrolēt datubāzes stāvokli gan izstrādes, gan produkcijas vidēs. Migrāciju failos tika definēti arī sistēmas darbībai nepieciešamie dati.

Izmantojot *git* un aprakstītos migrāciju failus, tika panākts, ka visas programmatūras komponentes bija saskaņotos stāvokļos. *git* repozitorijam uz izstrādes datora tika veidotas rezerves kopijas uz attālināta servera pēc katra izmaiņu kopuma (*commit*) vai arī darba dienas beigās.

7. DARBIETILPĪBAS NOVĒRTĒJUMS

7.1. Darbietilpības novērtējums pirms katras iterācijas

Katras iterācijas sākumā pēc lietotājstāstu apkopošanas, tika veikts to darbietilpības novērtējums, balstoties uz autora iepriekšēju pieredzi. Cieši saistītiem lietotājstāstiem tika dots kopīgs novērtējums. Tā kā programmējot viena lietotājstāsta prasības tika radīts kods, ko varēja lietot arī citu izpildē, tad patiesā darbietilpība katram lietotājstāstam atsevišķi nav uzrādīta, bet iterācijas beigās konstatēta kopējā darbietilpība.

Iterācija	Lietotājstāsta identifikators	Prognose (darba dienas)	Reāli (darba dienas)
1	auth_providers_sign_in_first	10	
	auth_providers_sign_in_again		
	auth_twitter_first,	5	
	auth_twitter_again		
	auth_native_sign_up_step1,	5	
	auth_native_sign_up_step2		
	auth_native_sign_in	3	
	auth_recovery	3	
	auth_native_auto_recovery	1	
2	auth_welcome_email	1	
	auth_sign_out	1	
	KOPĀ	29	35
	profile_change_name	1	
	profile_change_password	1	
	contact_send_message	1	
	contact_forward_to_email	1	
	admin_users	4	
	admin_contact_messages	4	
3	KOPĀ	12	16
	task_new	5	
	task_edit_creator	2	

Iterācija	Lietotājistāsta identifikators	Prognose (darba dienas)	Reāli (darba dienas)
	task_edit_doer		
	task_list	3	
	task_view	3	
	task_add_comment	2	
	KOPĀ	15	10.5
KOPĀ (visas iterācijas)		56	61.5

Kā redzams, kopējais prognozētais ir nedaudz mazāks par patieso (aptuveni 8% starpība). Tas visticamāk izskaidrojams ar to, ka autoram izstrādes gaitā nācās iepazīties ar jaunām tehnoloģijām un rīkiem vairāk nekā tas bija paredzējis sākumā.

7.2. Darbietilpības novērtējums pēc izstrādātā koda daudzuma ar COCOMO metodi

Pēc pēdējās iterācijas tika saskaitīts uzrakstīto programmkoda rindiņu skaits, izmantojot rīku *CLOC* [4] (versija 1.58).

Valoda	Koda rindiņu skaits
<i>PHP</i> (kods)	3946
<i>PHP</i> (vienībtesti)	1708
<i>HTML</i> (galvenokārt)	689
<i>JavaScript</i>	380
<i>CSS</i>	152

Tabula 7.1: Programmkoda rindiņu skaits

Atmetot vienībtestu kodu (jo, pēc autora domām, šīm koda rindiņām patērēto laiku var ieskaitīt pieskaitīt pie testējamā koda rakstīšanas laika) un atmetot *HTML* un *CSS* (kuru rakstīšana neaizņemt tik daudz laika), iegūstam $3946 + 380 = 4326$ koda rindiņas.

Izmantojot *COCOMO Basic* metodi darbietilpības novērtēšanai ar *Organic mode* koeficientiem, tiek iegūts 10.1 personmēneša darbietilpības novērtējums, kas ir krietni vairāk, nekā realitātē. Tas

skaidrojams ar to, ka *COCOMO* darbietilpības novērtējums ir uzticams apjomīgākiem projektiem, kur koda apjoms pārsniedz 10'000 rindiņu.

NOBEIGUMS UN SECINĀJUMI

Kvalifikācijas darba ietvaros autors guva praktisku pieredzi programmatūras izstrādē ar spējās izstrādes pieeju, prasību formulēšanu lietotājistāstos, nostiprināja zināšanas par *PHP*, *HTML*, *CSS*, *JavaScript*.

Autors apguva automātisku testu rakstīšanu *PHP* vidē, izmantojot *PHPUnit* testēšanas ietvaru. Tā kā visus vienībtestus varēja izpildīt dažu sekunžu laikā, tos varēja izpildīt ļoti bieži. Tādā veidā tie palīdzēja pārliecināties, ka jaunākās izmaiņas programmkodā nerada traucējumus jau esošajā funkcionalitātē. Kā papildus plusu automātiskajiem vienībtestiem jāmin to, ka tie mudina rakstīt programmas vienības (funkcijas, klases), kas ir autonomas un nav atkarīgas no globāliem stāvokļiem, jo šādu kodu ir vieglāk testēt. Šādas vienības ir arī vieglāk pielietot un tās nerada negaidītus blakusefektus.

Autors guva iemaņas lietotāja saskarnes veidošanā, kas spēj pielāgoties dažādām ekrāna izšķirtspējām. Tā kā izstrādātās sistēmas saskarnes bija salīdzinoši vienkāršas, tad šāda pieeja lietojamības uzlabošanai uz dažādām ierīcēm bija veiksmīga. Taču gadījumā, ja saskarnes būtu sarežģītākas, visticamāk, ka būtu nepieciešams veidot vairākas atsevišķas saskarnes (vietnes *mobilo* un *parasto* versiju).

Kopumā kvalifikācijas darba ietvaros tika izveidots funkcionējoša tīmekļa vietne ar svarīgākajām funkcijām, kādas nepieciešamas uzdevumu pārraudzības sistēmai. Darbs pie šīs sistēmas tiks turpināts arī pēc kvalifikācijas darba beigām, veidojot plašāku funkcionalitāti.

PATEICĪBAS

Pateicība prakses vadītājam Andrim Alnim par sniegtajiem padomiem un konsultācijām sistēmas izstrādes gaitā.

Pateicība darba vadītājam Dr. dat. Dainim Dosbergam par atbildēm uz jautājumiem un padomiem kvalifikācijas darba rakstīšanas gaitā.

IZMANTOTĀ LITERATŪRA UN AVOTI

- [1] “The Difference Between Agile Themes, Epics and User Stories | Themes, Stories, Epics.” . Pieejams: <http://agile101.net/2009/08/10/the-difference-between-agile-themes-epics-and-user-stories/>. Pēdējā piekļuve: 28. Maijs 2013. gads.
- [2] “Coding Standards - General - FuelPHP Documentation.”. Pieejams: http://fuelphp.com/docs/general/coding_standards.html. Pēdējā piekļuve: 28. Maijs 2013. gads.
- [3] “Chapter 13. Code-Coverage Analysis.”. Pieejams: <http://phpunit.de/manual/3.0/en/code-coverage-analysis.html>. [Pēdējā piekļuve: 28. Maijs 2013. gads.
- [4] “CLOC -- Count Lines of Code.”. Pieejams: <http://cloc.sourceforge.net/>. Pēdējā piekļuve: 28. Maijs 2013. gads.

PIELIKUMI

Kontrolieris “Profile” (classes/controller/profile.php)

```
<?php

namespace TaskGPS\Controller;

use \Response;
use \Router;
use \Lang;
use \Input;
use \Security;
use \TaskGPS\Model\Service\ProfileForm;

/**
 * Users can change their own password and name via the profile page.
 * @author peteris
 */
class Profile extends Base
{
    /**
     * Force the user to be authenticated to use this controller.
     * @return void
     */
    public function before()
    {
        parent::before();
        if (!$this->auth->is_signed_in())
        {
            Response::redirect(Router::get('frontpage'));
        }
    }
}
```



```

        Lang::load('profile', 'profile');
    }

/**
 * Create and handle profile form.
 * @return void
 */
public function action_form()
{
    $errors = array();
    $saved = false;
    $user = $this->auth->get_signed_in_user();
    $name = $user->name;

    if (Input::post() and Security::check_signature((string)Input::post('csrf_token')))
    {
        $service = new ProfileForm();
        $errors = $service->update_profile(
            $this->auth->get_signed_in_user(),
            (string)Input::post('name'),
            (string)Input::post('password'),
            (string)Input::post('password_again')
        );

        if ($errors)
        {
            $name = (string)Input::post('name');
        }
        else
        {
            $name = $user->name;
        }
    }
}

```

```

        $saved = true;
    }
}

$this->template->title = __('profile.form.html_title');
$this->load_content(
    'profile/form',
    array(
        'errors' => $errors,
        'saved' => $saved,
        'name' => $name,
    )
);
}
}

```

Modela serviss “ProfileForm” (classes/model/service/profileform.php)

```
<?php
```

```
namespace TaskGPS\Model\Service;
```

```
use \TaskGPS\Model\Entity\User;
```

```
class ProfileForm
```

```
{
```

```
    /**
```

```
     * Update user profile - name and password.
```

```
     *
```

```
     * @param User $user user who's data is being changed.
```

```
     * @param string $name new user name
```

```
     * @param string $password new password (blank if not being changed)
```

```
     * @param string $password_again must match $password (blank if not being changed)
```

```
     * @return array of errors or empty array on success
```

```
    */
```

```
public function update_profile(User $user, $name, $password, $password_again)
```

```
{
```

```
    $errors = array();
```

```
    $name_response = User::validate_name($name);
```

```
    $errors = array_merge($errors, $name_response['errors']);
```

```
    $name = $name_response['name'];
```

```
    if ($password or $password_again)
```

```
    {
```

```
        $password_response = User::validate_password($password, $password_again);
```

```
        $errors = array_merge($errors, $password_response['errors']);
```

```
        $password = $password_response['password'];
```

```
    }
```

```

else
{
    $password = $password_again = null;
}

if ($errors)
{
    return $errors;
}

$user->set_name($name);
$password and $user->set_password_hash($password);
$user->save();

return array();
}
}

```

Modeļa servisa “ProfileForm” vienībtesti
(tests/model/service/test_profileform.php)

```
<?php
```

```
namespace TaskGPS\Tests;
```

```
use \TaskGPS\Model\Service\ProfileForm;
```

```
use \TaskGPS\Model\Service\Mailer;
```

```
use \TaskGPS\Model\Service\User\Basic;
```

```
use \TaskGPS\Model\Entity\User;
```

```
/**
```

```
 * Tests for ProfileForm service (app/classes/model/service/profileform.php).
```

```
 *
```

```
 * @author peteris
```

```
 * @group taskgps
```

```
 * @group taskgps_unit
```

```
 */
```

```
class Test_ProfileForm extends UnitTestCase
```

```
{
```

```
    /**
```

```
     * @var User
```

```
    */
```

```
    private $user;
```

```
    /**
```

```
     * @var ProfileForm
```

```
    */
```

```
    private $profile;
```

```
    public function setUp()
```

```

{
    $basic = new Basic(Mailer::forge_dummy());
    $this->user = $basic->sign_up(
        self::NEW_USER_EMAIL,
        self::NEW_USER_PASSWORD
    );
    $this->profile = new ProfileForm();
}

public function tearDown()
{
    $this->user->hard_delete();
}

public function update_profile_data_provider()
{
    return array(
        // Passwords don't match
        array(
            'New Name 1', 'password1', 'password2',
            array('auth.passwords_dont_match')
        ),

        // OK with password change
        array('New Name 2', '~9F$7=)U0@@@', '~9F$7=)U0@@@', array()),

        // OK without password change
        array('New name 3', "", "", array()),
    );
}

/**
 * @dataProvider update_profile_data_provider

```

```

*/
public function test_update_profile($name, $password, $password_again, $expected_errors)
{
    $errors = $this->profile->update_profile(
        $this->user,
        $name,
        $password,
        $password_again
    );
    $this->assertLanguageStringArray($expected_errors, $errors);

    if (!$errors)
    {
        $this->assertEquals($this->user->name, $name);
    }
}
}

```

Kvalifikācijas darbs „*Uzdevumu pārraudzības tīmekļa vietnes izstrāde*” 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: *Pēteris Rudzusīks* _____ .06.2013.

Rekomendēju darbu aizstāvēšanai

Darba vadītājs: **Dr. dat. Dainis Dosbergs** _____ .06.2013.

Recenzents: **Dr. dat. , Zane Bičevska**

Darbs iesniegts 03.06.2013.

Kvalifikācijas darbu pārbaudījumu komisijas sekretārs: *Imants Gorbāns* _____

Darbs aizstāvēts kvalifikācijas darbu pārbaudījuma komisijas sēdē

____.06.2013. prot. Nr. _____

Komisijas sekretārs(-e): _____