

RĪGAS 64. VIDUSSKOLA

# **LAIKA PLĀNOŠANAS UN IERADUMU SEKOŠANAS RĪKS**

**Programmatūras prasību specifikācija**

Darba autors:  
**Eduards Žuga**

Darba vadītājs:  
**Edvards Bukovskis**

RĪGA 2023

# Saturs

<b>Definīcijas, akronīmi un saīsinājumi .....</b>	<b>3</b>
<b>1. Ievads.....</b>	<b>4</b>
1.1. Nolūks.....	4
1.2. Darbības sfēra .....	4
1.3. Saistība ar citiem dokumentiem.....	4
1.4. Pārskats .....	4
<b>2. Problēmas izpēte un analīze .....</b>	<b>5</b>
2.1. Aktualitāte.....	5
2.2. Pētāmā problēma un mērķauditorija .....	5
<b>3. Programmatūras prasību specifikācija .....</b>	<b>6</b>
3.1. Funkcionālās prasības.....	6
3.2. Nefunkcionālās prasības .....	10
3.2.1. Uzturamība .....	10
3.2.2. Lietojamība .....	10
3.2.3. Datu glabāšana .....	10
3.2.4. Drošība .....	10
3.2.5. Veiktspēja.....	11
3.2.6. Datu bāze.....	11
<b>4. Programmatūras izstrādes plāns .....</b>	<b>12</b>
4.1. Detalizēti izstrādes posmi .....	12
<b>5. Atklādošanas un akcepttestēšanas pārskats .....</b>	<b>13</b>
<b>6. Lietotāja ceļvedis .....</b>	<b>14</b>
6.1. Vienkāršots programmatūras darbības plāns .....	14
6.2. Programmatūras ieviešanas plāns .....	15
<b>7. Piemērotās licences pamatojums .....</b>	<b>16</b>
<b>Pielikumi .....</b>	<b>17</b>

## Definīcijas, akronīmi un saīsinājumi

Saīsinājums, termins	Skaidrojums
LPISR	Laika plānošanas un ieradumu sekošanas rīks
AL	Augstākā līmeņa
Lietotājs	Persona, kas izmanto programmatūru
m.g.	Mācību gads
Saskarne	Funkciju kopa, kas nodrošina datu apmaiņu starp lietotāju un programmatūru.
UI	“User Interface” - Saskarne
OS	Operētājsistēma

# **1. Ievads**

## **1.1. Nolūks**

Mūsdienās ikdienas dzīve ir sasteigta, un šķiet, ka visu laiku pietrūkst laika, un šajā ikdienas steigā ir viegli piemirst rūpēties par sevi, par saviem ieradumiem, kā arī nodarboties ar saviem hobijiem un interesēm.

Dokuments ir sagatavots “Skola2030” projekta ietvaros kā AL programmēšanas eksāmena piekļuves materiāls. Dokumentā tiek aprakstīta LPISR struktūra un funkcionalitāte, kā arī, dokuments ietver gatavā produkta redzējumu un tā daļēju tehnisko izpildījumu, kas turpmāk būs kā pamats pilnas programmatūras izveidei.

## **1.2. Darbības sfēra**

LPISR ir paredzēts kā publiski pieejama aplikācija, kuru var lietot ikviens, kuram ir interese vai nepieciešamība pēc funkcijām, ko šī programmatūra nodrošina. Šī programmatūra pārveido ikdienas un laika plānošanu par interaktīvu spēli, padarot sekošanu saviem ieradumiem un uzdevumiem mazāk garlaicīgu un motivējošāku.

## **1.3. Saistība ar citiem dokumentiem**

1. Dokumenta PPS noformēšanā ievērotas standarta LVS 68:1996 prasības.
2. Dokumenta PPS izveidē ievēroti 2022./2023. m.g. centralizētā eksāmena programmas kritēriji augstākajam mācību satura apguves līmenim programmēšanā.
3. Dokumenta PPS kalpo kā atsauce turpmākiem sistēmas uzlabojumiem

## **1.4 Pārskats**

Pirmajā nodaļā ir ievadinformācija un vispārējs apraksts, kas satur dokumenta nolūku, definīciju skaidrojumu, saistību ar citiem dokumentiem un sistēmas darbības sfēru.

Otrajā nodaļā tiek veikta problēmas izpēte un analīze.

Trešajā nodaļā tiek aprakstītas programmatūras funkcionālās un nefunkcionālās prasības

Ceturtā nodaļa sniedz programmatūras izveides plānu.

Piektajā nodaļā tiek veikta programmatūras atklūdošana un testēšana.

Sestajā nodaļā ir izveidots lietotāja ceļvedis.

Septītajā nodaļā ir aprakstīts piemērojamās licences pamatojums.

## 2. Problēmas izpēte un analīze

### 2.1. Aktualitāte

Mūsdienās ikdienas dzīve ir sasteigta, un šķiet, ka visu laiku pietrūkst laika, un šajā ikdienas steigā ir viegli piemirst rūpēties par sevi, par saviem ieradumiem, kā arī nodarboties ar saviem hobijiem un interesēm. Savukārt modernajā pasaulē trūkst disciplīnas, jo uzmanību novērsošs saturs ir pieejams visapkārt, un katru brīdi, kā kādam paliek garlaicīgi, uzreiz visa uzmanība tiek novirzīta uz sociālajiem medijiem, spēlēm un citiem dopamīna izraisošiem avotiem.

### 2.2. Pētāmā problēma un mērķauditorija

Konkrēta risināmā problēma ir garlaicība sava laika plānošanā, kā arī uzskati, ka laika plānošana ir laikietilpīgs process, ka labāk un vienkāršāk ir dzīvot “plūsmā”, kā rezultātā var rasties negatīvi, bet dopamīnu izraisoši ieradumi un pazust pozitīvi veselību uzlabojoši un pašapziņas veidojoši ieradumi. Stresam arī ir liela nozīme laika izpratnē, jo stresojot var šķist, ka laiks pietrūkst, bet realitātē laiks ir pietiekami.

Mērķauditorija veidotajai programmatūrai ir:

- Skolēni un studenti, jo šī vecuma cilvēkiem ir visvieglāk paļauties dopamīna izraisošiem avotiem, šīm personām laika plānošanā un ieradumu kontrolēšana var šķist nesvarīgs un garlaicīgs process, pietrūkst motivācija;
- Ikviens strādājošs cilvēks, it īpaši strādājoši cilvēki ar nepilngadīgiem bērniem, jo var būt grūti apvienot darbu ar dažādām interesēm, it īpaši vēl kad ir jāaudzina bērns, var pietrūkt īstais rīks, kas palīdzētu risināt šo problēmu.

Lai izveidotu produktu, kas pēc iespējas vairāk līdzinātos minētajās mērķauditorijas redzējumam un vajadzībām, nepieciešam aptauja, ar mērķi noskaidrot tipiskākās problēmas vai neērtības šajā jomā, kas programmatūrai ir jāpanāk, lai atrisinātu šo problēmu vai apmierinātu lietotājus, piemēram aptaujā tiktu iekļauti šādi jautājumi:

- “Kāds ir tavš vecums, kāda ir tava nodarbošanās?”
- “Vai tev ir kāds hobijs, ar kuru vēlies nodarboties, bet aizmirsti vai pietrūkst laiks?”
- “Vai ir kādi ieradumi, kurus vēlies veidot/atmest, un kāpēc neesi attiecīgos ieradumus izveidojis/atmetis?”
- “Vai tu plāno savu laiku, ja nē, kāpēc?”
- “Vai tev motivācija ir noteicošs faktors mērķu sasniegšanā un lietu darīšanā?”

Anketēšanas metode ir vispiemērotākā šīs problēmas izpētē, jo ar anketēšanu ir visvienkāršāk iegūt masveidā datus, kas nepieciešami konkrētās problēmas risināšanā, un LPISR mērķis ir atvieglot ikdienu lietotājam, tāpēc ir nepieciešams ievākt tieši lietotāju vēlmes un analizēt viņu domas.

### 3. Programmatūras prasību specifikācija

Programmatūras produkta mērķauditorijai ir nepieciešams rīks, kas palīdz un motivē plānot laiku, atgādināt par izpildāmajiem uzdevumiem, piemēram, nodarboties ar savu hobiju, turpināt vai atmest ieradumu utt. Šim rīkam ir jānodrošina interaktīva vide ar apbalvošanas un spēļu elementiem, padarot laika plānošanu interesantu, motivējot lietotāju pildīt izveidotos uzdevumus.

LPISR ir kalendāra veida aplikācija, kas pārvērs laika plānošanu un sekošanu saviem ieradumiem spēlē. Lietotājs reģistrē ieradumus, kurus vēlas izveidot vai atmest, un attiecīgi par to izpildi iegūst punktus un krāj līmeņus, pārveidojot ikdienas dzīvi par spēli.

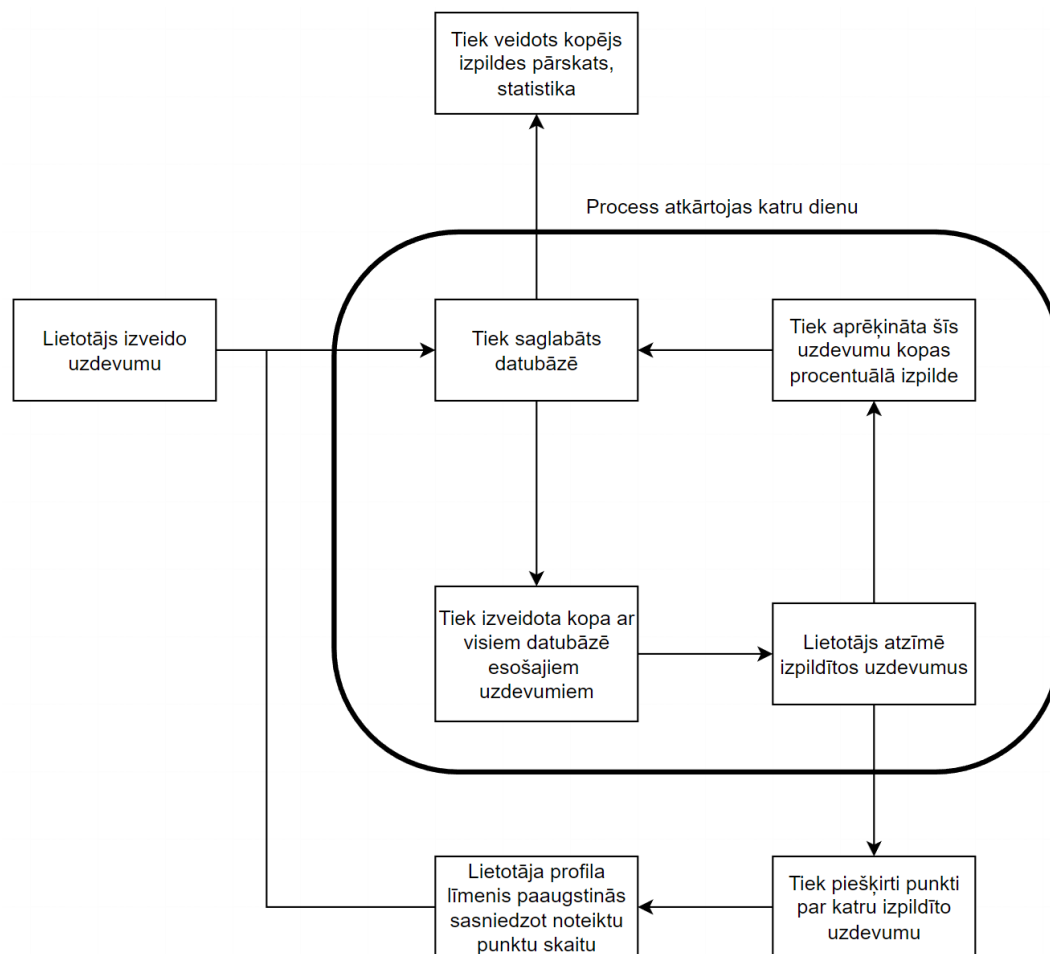
LPISR tiks izstrādāts uz “Flutter” bāzes. Flutter ir bezmaksas un atvērta pirmkoda lietotāja saskarnes programmatūras izstrādes komplekts, kas sastāv no Dart, C, C++ programmēšanas valodām. Flutter programmatūrai ir laba pārnesamība - kods strādā uz visām galvenajām OS (Windows, MacOS, Linux, Android, iOS, strādā arī pārlūkprogrammatūrās) ar ļoti vienkāršu pielāgošanu. Tā kā izstrādājamā programmatūras produkta funkcijas ir vienkāršas un lielākā funkcionalitāte balstās ir tā UI, Flutter ir vispiemērotākā programmatūra šī produkta izstrādei.

Flutter arī ir pieejamas dažādas papildus bibliotēkas, kas nodrošina specifiskas funkcijas. LPISR ir nepieciešama datu glabāšana, kas tiek nodrošināta ar “hive” bibliotēku, kā arī izmantota “flutter\_heat\_map” bibliotēka statistikas attēlošanā.

#### 3.1. Funkcionālās prasības

LPISR pamatfunkcija ir uzdevumu izveide un to izpildes atzīmēšana, kā arī to saglabāšana. (skatīt 1. attēlu). Visas pārējās funkcijas, kā statistika, kalendārs, lietotāja līmenis, ir sistēmas papildinājumi, kas balstās uz pamatfunkcijas.

LPISR galvenās funkcijas ir uzdevumu izveides funkcija (skatīt 1. tabulu), uzdevumu rediģēšanas funkcija (skatīt 2.tabulu), uzdevumu izpildes funkcija (skatīt 3. tabulu), statistikas veidošanas funkcija (skatīt 4. tabulu), lietotāja līmeņa rēķināšanas funkcija (skatīt 5. tabulu) un kalendāra funkcija (skatīt 6. tabulu).



Attēls 1. LPISR galvenās funkcijas darbības modelis

### Uzdevuma izveides funkcija

<b>Mērķis:</b>
Ļauj lietotājam izveidot uzdevumu, iestatīt tā kategoriju, nepieciešamo laika izpildi, uzdevuma vērtību, tiek saglabāts datubāzē
<b>Ievade:</b>
Lietotāja izdomāts uzdevuma nosaukums (teksts), izvēlētā kategorija no iepriekš sagatavota kategoriju saraksta, uzdevuma vērtība (vērtību skala no 1 - 10 veseli skaitļi), plānotais uzdevuma izpildes laiks (minūtes)
<b>Apstrāde:</b>
Uzdevums tiek saglabāts.
<b>Izvade:</b>
Saraksts ar lietotāja izveidotajiem uzdevumiem, saraksts, kas ir sakārtots pa uzdevumu kategorijām, ir redzams UI.

Tabula 1.

### Uzdevuma rediģēšanas funkcija

<b>Mērķis:</b>
Ļauj lietotājam rediģēt vai dzēst iepriekš izveidoto uzdevumu datus (nosaukumu, kategorija, uzdevuma vērtību, izpildes laiku)
<b>Ievade:</b>
Rediģējot: lietotāja izdomāts uzdevuma nosaukums (teksts), izvēlētā kategorija no iepriekš sagatavota kategoriju saraksta, uzdevuma vērtība (vērtību skala 1 - 10, veseli skaitļi), plānotais uzdevuma izpildes laiks (minūtes).
<b>Apstrāde:</b>
Uzdevuma dati tiek izmainīti un saglabāti uzdevumu tabulā, dzēšot uzdevuma dati tiek izdzēsti.
<b>Izvade:</b>
Saraksts ar lietotāja izveidotajiem un rediģētajiem uzdevumiem, saraksts, kas ir sakārtots pa uzdevumu kategorijām, ir redzams UI.

Tabula 2.

### Uzdevuma izpildes funkcija

<b>Mērķis:</b>
Ļauj lietotājam atzīmēt izpildītos uzdevumus.
<b>Ievade:</b>
Uzdevums tiek atzīmēts kā izpildīts (true or false).
<b>Apstrāde:</b>
Uzdevums tiek saglabāts kā izpildīts, tiek aprēķināta visu izpildīto un visu uzdevumu attiecība konkrētajā dienā, lietotājam pieskaita punktus (punktu daudzums atkarīgs no uzdevuma vērtības) pie kopējā punktu skaita, dati tiek saglabāti.
<b>Izvade:</b>
Izpildīto un visu uzdevumu attiecība konkrētajā dienā, kopējais punktu skaits pēc uzdevuma izpildes/punktu piešķiršanas.

Tabula 3.



### Statistikas veidošanas funkcija

<b>Mērķis:</b>
Iegūt grafisku uzdevumu izpildes attēlojumu.
<b>Ievade:</b>
Izpildīto un visu uzdevumu attiecība konkrētajā dienā.
<b>Apstrāde:</b>
Izpildīto un visu uzdevumu attiecība konkrētajā dienā tiek attēlota grafiski (siltuma karte).
<b>Izvade:</b>
Grafisks izpildīto un visu uzdevumu attiecības attēlojums iekš UI.

Tabula 4.

### Lietotāja līmeņa aprēķināšanas funkcija

<b>Mērķis:</b>
Lietotāja līmeņa aprēķināšana atkarībā no punktu skaita.
<b>Ievade:</b>
Kopējais punktu skaits.
<b>Apstrāde:</b>
Kopējais punktu skaits tiek ievadīts vienādojumā, caur kuru tiek aprēķināts lietotāja līmenis.
<b>Izvade:</b>
Lietotāja līmenis, redzams saskarnē.

Tabula 5.

## Kalendāra funkcija

<b>Mērķis:</b>
Iespēja lietotājam plānot laiku un notikumus kalendārā. Iespēja rediģēt vai izdzēst notikumu.
<b>Ievade:</b>
Notikums, tā datums un laiks.
<b>Apstrāde:</b>
Notikums tiek saglabāts kalendārā specificētajā datumā un laikā. Ja notikums tiek dzēsts, tas kalendārā vairs neparādās.
<b>Izvade:</b>
Kalendārs ar pievienoto vai izdzēsto notikumu, redzams saskarnē

Tabula 6.

## 3.2. Nefunkcionālās prasības

### 3.2.1. Uzturamība

Programmatūras kodam ir jābūt labi strukturētam, komentētam un viegli uztveramam, jāievēro labs programmēšanas stils.

### 3.2.2. Lietojamība

Programmatūrai jābūt brīvi pieejamai jebkuram lietotājam, kā arī ērti lietojamai - visa sniegtā informācija saskarnē ir skaidra un lietotājam viegli uztverama, kā arī intuitīvi lietojamai. Programmatūrai jābūt lietojamai uz populārākajām OS (Windows, MacOS, Linux, Android, iOS).

### 3.2.3. Datu glabāšana

Ievadītie dati jā saglabā uz sistēmas vietējās krātuves, uz kuras tiek darbināta programmatūra, lai katru reizi no jauna atverot programmatūru, jāielādē iepriekšējā lietošanas reizē saglabātie dati.

### 3.2.4. Drošība

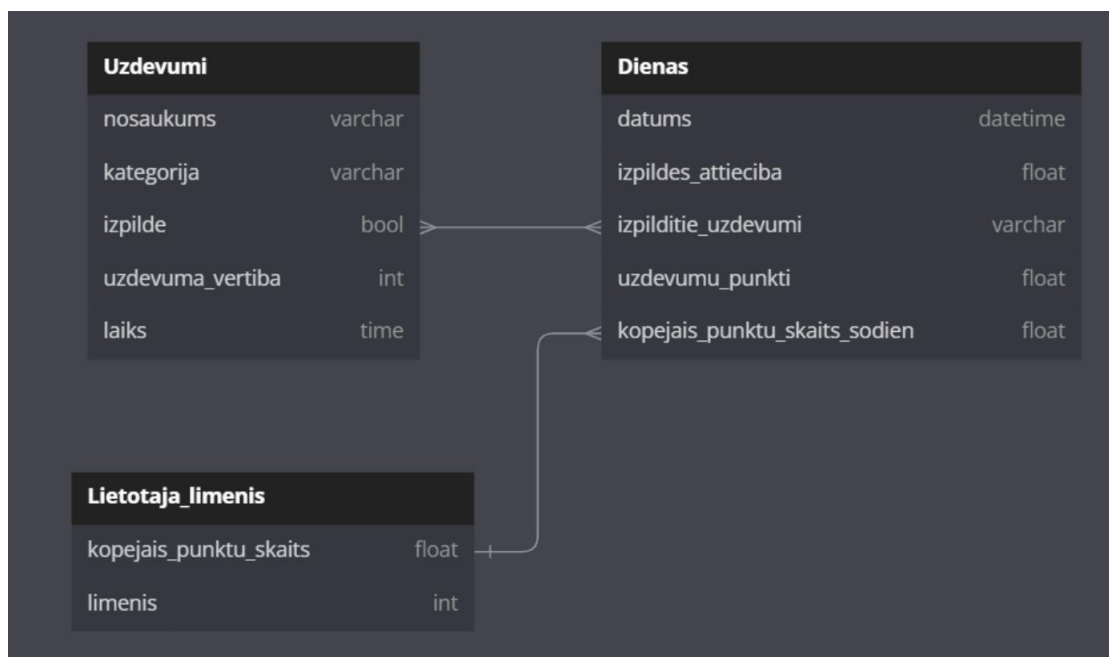
Programmatūrā ievadītajiem datiem jābūt droši glabātiem, lai tiem nav iespējams piekļūt no ārējām sistēmām. Lietotāja ievadītie dati tiek šifrēti, lai nodrošinātu lielāku datu aizsardzību un netiktu pārkāpta privātuma politika. Piekļuve lietotāja ievadītajiem datiem programmatūras izveidē iesaistītajām pusēm un trešajām personām netiek nodrošināta.

### 3.2.5. Veiktspēja

Programmatūrai ir jādarbojas ar augstu precizitāti, lai ievadītie un apstrādātie dati tiktu uzreiz attēloti un redzami UI, lai izvairītos no dublikātu veidošanas un citām cilvēka kļūdām, kas radušās neprecīzas vai novēlotas informācijas attēlošanas rezultātā.

### 3.2.6. Datu bāze

LPISR gala publicējamajā versijā tiks iesaistīta datubāze, lai uzglabātu nepieciešamo informāciju un nodrošinātu vienkāršāku un ātrāku tās apmaiņu programmatūras sistēmā (skatīt 2. attēlu). Datu bāze tiks saglabāta izmantojot Flutter “hive” bibliotēku, jo tā spēj uzglabāt lielu datu daudzumu ar augstu drošību un šifrēšanu un labu veiktspēju.

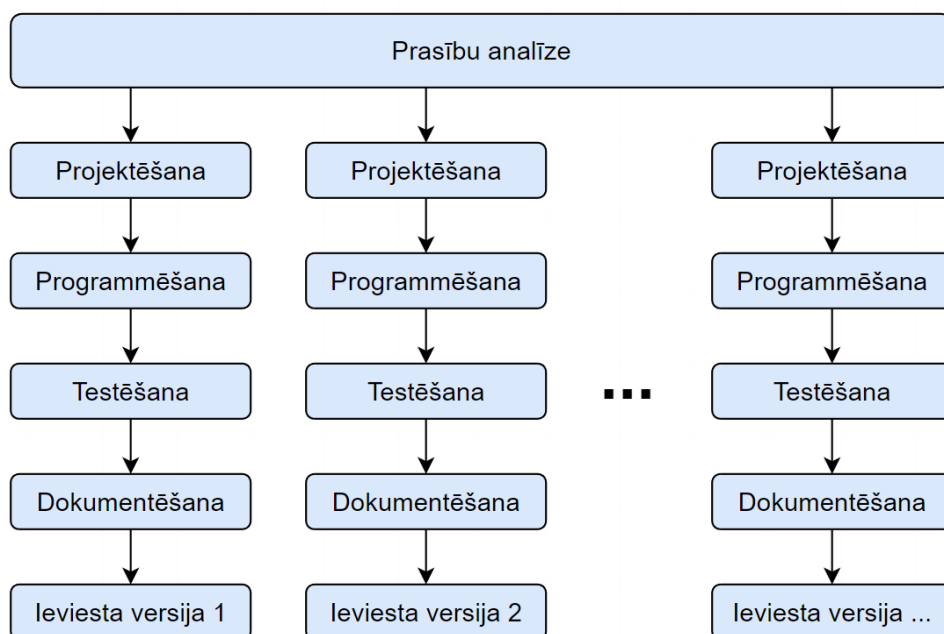


2. attēls: LPISR datu bāzes relāciju modelis

## 4. Programmatūras izstrādes plāns

LPISR izstrādes modelim, galvenokārt, jābalstās uz izstrādājamās programmatūras dokumentācijas, taču tajā pašā laikā tam jāļauj veikt programmatūras paplašinājumus un uzlabojumus, lai LPISR no vienkāršota produkta, kas satur pašas galvenās funkcijas, pakāpeniski attīstītos un kļūtu par pilnvērtīgu produktu. Programmatūras izstrādē arī var tikt iesaistīts pats lietotājs, ievācot tā atsauksmes un attiecīgi uzlabojot produktu, tādēļ LPISR izstrādes modulim jābūt iteratīvam, lai spētu galvenās prasības papildināt ar dažādiem uzlabojumiem vai atjauninājumiem.

LPISR tiks izstrādāts pēc inkrementālā modeļa (skatīt 2. attēlu).



2. attēls: LPISR izstrādes modelis

### 4.1. Detalizēti izstrādes posmi

1. Prasību analīze - tiktu veikta aptauja laika menedžēšanas problēmām, kādi ir iespējamie risinājumi vai ietekmējošie faktori
2. Projektēšana - tiek izstrādātas programmatūras funkcijas balstoties uz 1. punktā veiktās aptaujas rezultātiem.
3. Programmēšana - iepriekšējā posmā izstrādāto programmatūras funkciju realizēšana “Flutter” platformā.
4. Testēšana - programmas funkcionalitāte tiek pārbaudīta.
5. Dokumentēšana - dokumentācija tiek papildināta ar testēšanas rezultātiem un citu informāciju
6. Versijas ieviešana - programmatūras versija tiek palaista publiskai piekļuvei. Katram lietotājam ir iespēja veikt ieteikumus jauna produkta versijai, kuri var tikt ņemti vērā jaunas versijas projektēšanā un realizācijā.

## 5. Atklādošanas un akcepttestēšanas pārskats

LPISR izstrādē tika veikti testi, lai pārbaudītu sistēmas darbību. Testi tika veikti funkcijās ar lielāko kļūdu iespējamību, kuras prasa lietotāja ievadītus datus.

ID	Ievade	Sagaidāmais rezultāts	Reālais rezultāts
saveNewHabit	Nosaukuma ievadē ir jebkāda simbolu virkne ("Meditate")	Funkcija saglabā un atgriež sarakstu ar ievadīto nosaukumu	Izpildās
Apraksts	Kategorijas izvēle (dota izvēlne ar izvēles opcijām)	Funkcija saglabā un atgriež sarakstu ar izvēlēto kategoriju	Izpildās
Saglabā datubāzē jaunu ievadīto uzdevumu			

ID	Ievade	Sagaidāmais rezultāts	Reālais rezultāts
editHabit	Nosaukuma ievadē ir jebkāda simbolu virkne ("Meditate")	Funkcija saglabā un atgriež sarakstu ar ievadīto nosaukumu	Izpildās
Apraksts	Kategorijas izvēle (dota izvēlne ar izvēles opcijām)	Funkcija saglabā un atgriež sarakstu ar izvēlēto kategoriju	Izpildās
Rediģē datubāzē esošu uzdevumu			

ID	Ievade	Sagaidāmais rezultāts	Reālais rezultāts
habitCompletionPercentage	Uzdevums tiek atzīmēts kā izpildīts	Funkcija saglabā un atgriež izpildīto uzdevumu un visu uzdevumu attiecību	Izpildās
Apraksts	Uzdevums tiek atzīmēts kā neizpildīts	Funkcija saglabā un atgriež sarakstu ar izvēlēto kategoriju	Izpildās
Aprēķina izpildīto un visu uzdevumu attiecību			

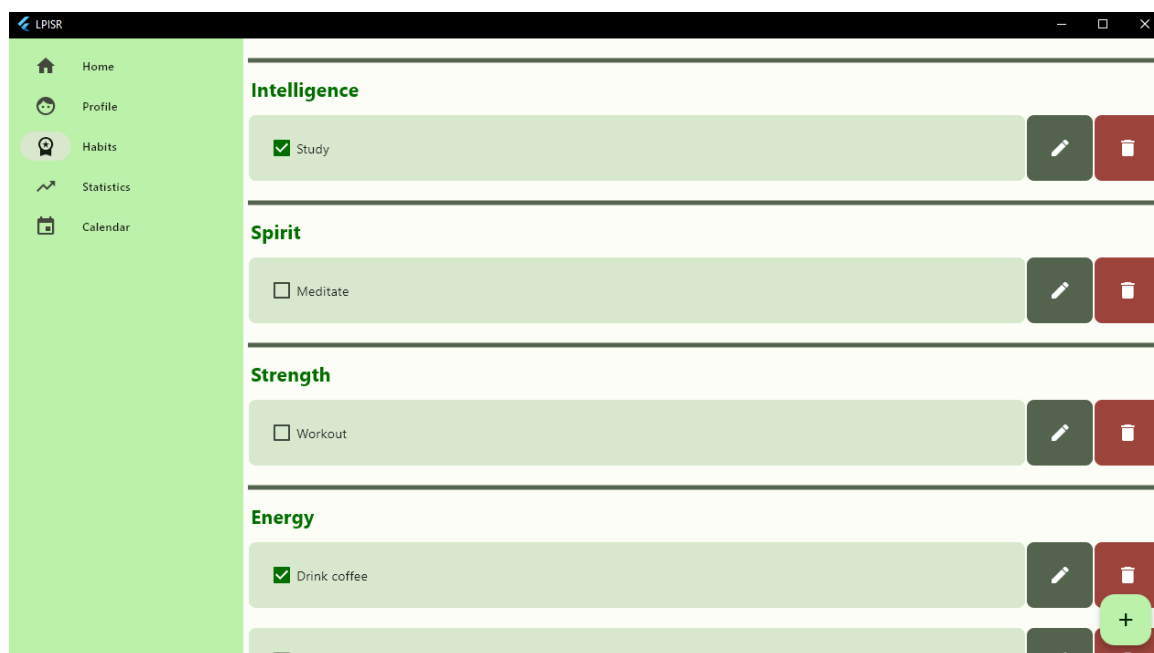
## 6. Lietotāja ceļvedis

LPISR ir paredzēts jebkuram lietotājam, kurš vēlas to lietot vai izmantot tā funkcijas. LPISR atbalsta Windows, MacOS, Linux, Android, iOS operētājsistēmas.

Šis LPISR ceļvedis ir paredzēts jebkuram LPISR lietotājam.

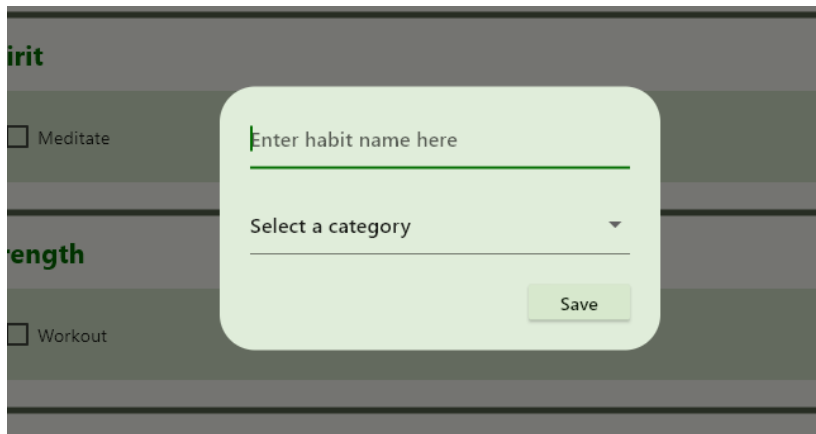
### 6.1. Vienkāršots programmatūras darbības plāns

1. Atverot LPISR UI, ir redzama sāukmlapa ar pogām kreisajā malā, kas ļauj pārslēgties starp lapām (skatīt 3. attēlu). “Home” lapā ir vispārīga informācija par šodienas kalendāra notikumiem, izpildāmajiem uzdevumiem u.c. “Profile” lapā ir informācija par lietotāja sakrātajiem punktiem, līmeni un punktu iegūšanas un tērēšanas vēsturi. “Habits” lapā var redzēt visus uzdevumus un ir iespējams izveidot, rediģēt un dzēst uzdevumus. “Statistics” lapā ir redzama kopējā statistika par izpildītajiem uzdevumiem. “Calendar” lapā ir pieejams kalendārs, kurā var plānot laiku, veidojot notikumus.
2. Atverot “Habits” lapu ir redzams saraksts ar jau pievienotajiem uzdevumiem, iespēju pievienot jaunu, rediģēt vai dzēst jau esošu uzdevumu (skatīt 3. attēlu).



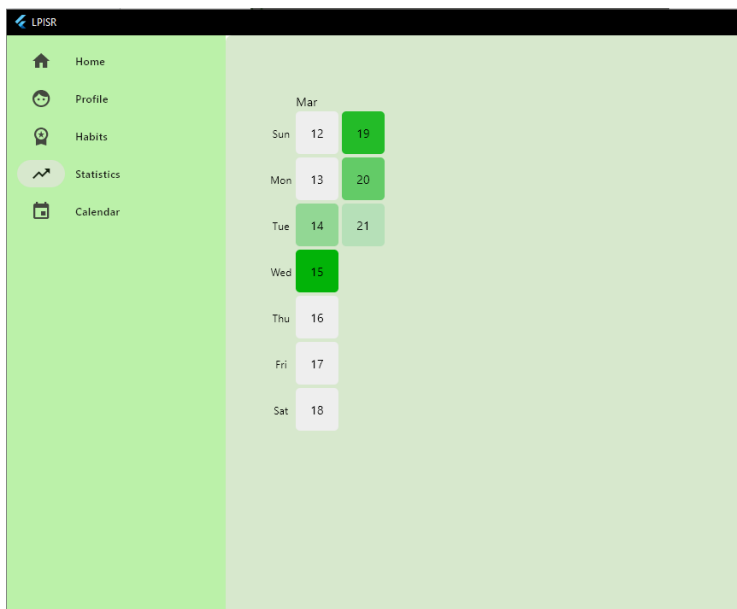
3. attēls: ekrānuzņēmums ar LPISR “Habits” lapu

3. Nospiežot uz jauna uzdevuma pievienošanas pogas (“+”) vai esošā uzdevuma rediģēšanas pogas (apzīmēta tumši zaļa ar zīmuli) atveras logs, kurā šobrīd LPISR esošajā versijā var ievadīt uzdevuma nosaukumu un izvēlēties kategoriju (skatīt 4. attēlu). Nospiežot uz pogas “Save” uzdevums vai izmaiņas tiks saglabātas. Ja darbību vēlas atcelt, uz klaviatūras jānospiež “Esc” poga. Nospiežot uz uzdevuma dzēšanas pogas (apzīmēta sarkana ar atkritumu tvertni), uzdevums tiks izdzēsts.



4. attēls: Uznirstošais logs, kur jāievada uzdevuma nosaukums un kategorija

- Atzīmējot izpildītos uzdevumus, “Statistics” lapā attiecīgā datuma lauciņš paliks tumšāks atkarībā no uzdevumu izpildes tajā dienā (skatīt 5. attēlu) - tumšāks lauciņš, vairāk uzdevumu izpildīts (skatīt 6. attēlu).



5.. attēls: “Statistics” lapa



6. attēls: Lauciņu krāsu maiņa

## 6.2. Programmatūras ieviešanas plāns

Pēc veiksmīgas produkta izstrādes un vēlamā gala rezultāta ieguves, pirmo versiju padarīt brīvi pieejamu publikai internetā. Katram programmatūras lietotājam ir iespēja sniegt kritiku un ieteikumus par produkta uzlabošanu un funkciju pievienošanu, kas var tikt ņemti vērā nākamās versijas izstrādē.

## 7. Piemērotās licences pamatojums

LPISR tiks piešķirta GNU Vispārējā Publiskā Licence (GPLv3).

Programmatūras mērķis ir uzlabot un atvieglot sabiedrības ikdienas dzīvi, bet to var panākt tikai ar lietotāja personīgu iesaisti un veltījumu programmatūras lietošanā, kā arī LPISR ir domāts būt pieejamam visai publikai. Katra cilvēka motivējošie vai garlaicību nosakošie faktori var atšķirties, tāpēc programmatūrai jābūt modificējamai pēc katra lietotāja vajadzībām. GPLv3 licence ļauj lietotājiem iesaistīties programmatūras attīstībā, veicot dažādus ieteikumus, pat papildinot un izveidojot jaunas funkcijas vai veicot citas modifikācijas, tādējādi nodrošinot pielāgojamību katram indivīdam, kā arī programmatūras turpmāko attīstību ārpus pirmkoda veidotāju veiktajiem uzlabojumiem. LPISR nav domāta kā komerciāla aplikācija, bet gan tēmēta uz kopējās publikas ieguvumu.



## **Pielikumi**

1. pielikums  
LPISR pirmkods

Pieejams GitHub repozitorijā: <https://github.com/Edzonsxd/LPISR>