

# Uzdevumu žurnāls

Martins Teteris 12.IND

Rīgas tālmācības vidusskola

Skolotājas: Aija Lagzdiņa, Inta Znotiņa

Priekš, 2023

# Satura rādītājs

<b>Problēmas izpēte un analīze</b>	<b>3</b>
<b>Programmatūras prasību specifikācija</b>	<b>4</b>
<b>Programmatūras izstrādes plāns</b>	<b>7</b>
<b>Atklūdošanas un akcepttestēšanas pārskats</b>	<b>8</b>
<b>Lietotāja ceļvedis</b>	<b>9</b>
<b>Piemērotās licences pamatojums</b>	<b>11</b>
<b>Projekta failu struktūra</b>	<b>12</b>
dbInit.sql	13
StaticDetails.cs	14
frmAddAssignment.cs	15
frmAssignments.cs	16
frmLogin.cs	17

## Problēmas izpēte un analīze

Mērķauditorija šai programmai ir skolnieki, un ar šo programmu viņi var žurnālot savus skolas uzdevumus / projektus / ieskaites / esejas, un labāk pārzināt, kuri uzdevumi ir svarīgāki un kurus vajag sākt pildīt. Man bieži ir bijusi vēlme kautkur visu, ko man vajag darīt mēnesī ielikt vienā vietā, lai es varu labāk zināt, uz ko fokusēt. Šīs programmas mērķis ir nodrošināt platformu, lai lietotāji varētu iesniegt un skatīt savus uzdevumus. Lai to panāktu, mums ir jārisina šādas problēmas:

- Lietotāja autentifikācija: programmai ir jānodrošina, ka uzdevumiem var piekļūt tikai autentificēti lietotāji. Tam nepieciešama droša login sistēma, kas pārbauda lietotāja identitāti, pirms tiek piešķirta piekļuve uzdevumiem.
- Uzdevuma pievienošana: programmai ir jāļauj lietotājiem pievienot jaunus uzdevumus. Tam ir nepieciešama veidlapa, kurā tiek apkopota informācija par uzdevumu, piemēram, nosaukums, steidzamības līmenis un izpildes datums. Turklāt programmai ir jānodrošina, lai katram lietotājam ir uzdevumi, ko iesniedza paši
- Uzdevumu skatīšana: programmai ir jāļauj lietotājiem skatīt savus iesniegtos uzdevumus. Tam ir nepieciešama lapa, kurā tiek parādīti visi lietotāja iesniegtie uzdevumi.
- Datu bāzes integrācija: mprogrammai ir jā saglabā uzdevuma informācija SQL datu bāzē. Šim nolūkam programmai ir jāizveido savienojums ar datu bāzi un jāizpilda SQL vaicājumi, lai lasītu un saglabātu datus.
- Lietotāja pieredze: programmai ir jānodrošina laba lietotāja pieredze. Šim nolūkam programmai ir jābūt lietotājam draudzīgam interfeisam ar skaidriem norādījumiem un izpalīdzīgiem ziņojumiem.

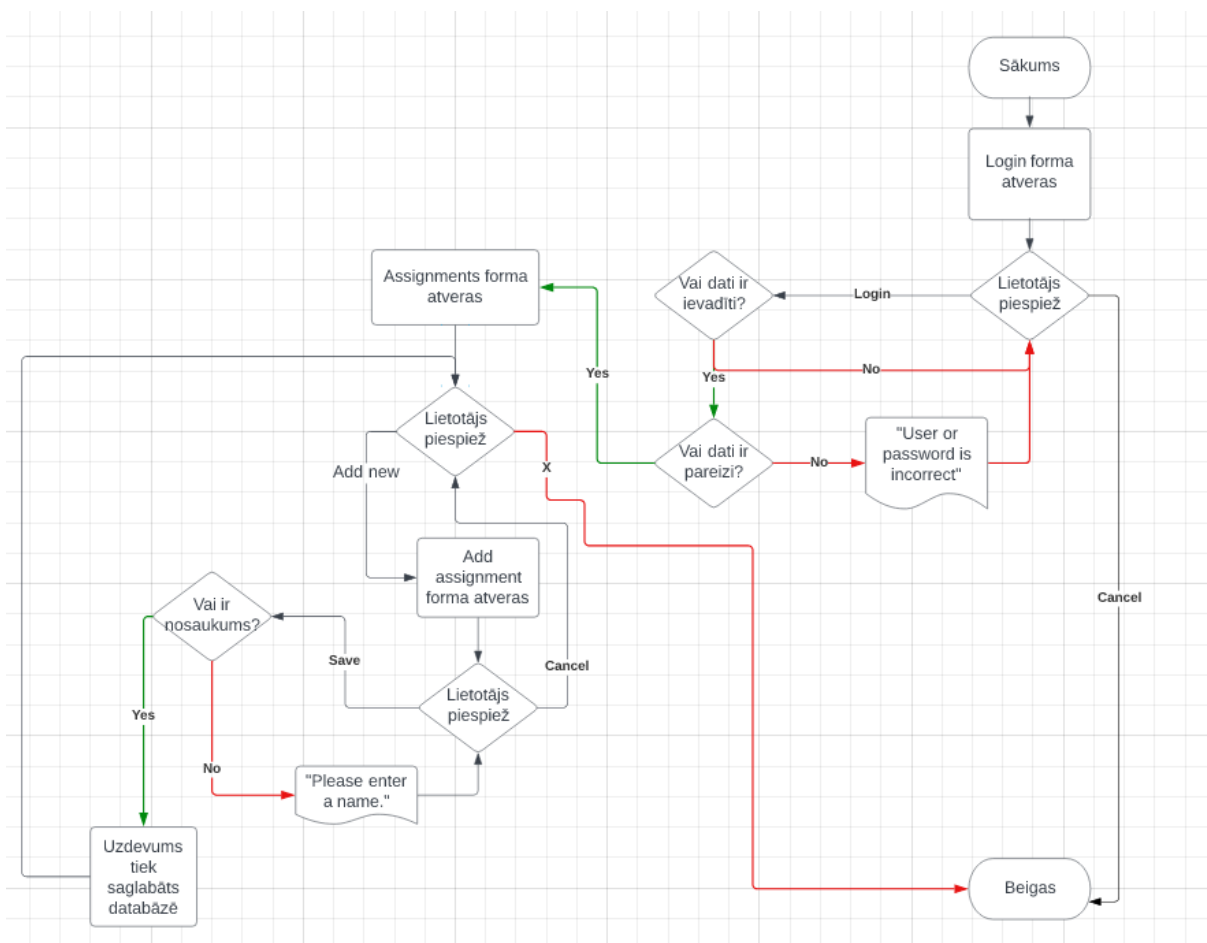
# Programmatūras prasību specifikācija

## Funkcionālās prasības:

- Lietotāja autentifikācija:
  - Sistēma autentificē lietotājus, pamatojoties uz viņu datiem (lietotājevārdu un paroli).
- Uzdevuma iesniegšana:
  - Sistēma ļauj lietotājiem iesniegt jaunus uzdevumus.
  - Sistēmai ir jāiesniedz uzdevuma nosaukums, steidzamības līmenis un izpildes datums.
  - Sistēma nodrošina, ka katrs uzdevums ir unikāls lietotājam, kurš to iesniedzis.
- Uzdevuma apskate:
  - Sistēma ļauj lietotājiem skatīt visus savus iesniegtos uzdevumus.
  - Sistēma parāda katra uzdevuma nosaukumu, steidzamības līmeni un izpildes datumu.
- Datu bāzes integrācija:
  - Sistēma jāintegrē ar MySQL datu bāzi, lai saglabātu un lasītu piešķiršanas informāciju.
- Lietotāja pieredze:
  - Sistēma nodrošina lietotājam draudzīgu saskarni, kurā ir viegli orientēties.
  - Sistēma lietotājiem parāda skaidrus norādījumus un atgriezeniskās saites ziņojumus.
  - Sistēma nodrošina intuitīvu veidlapu uzdevumu iesniegšanai.

## **Lietotāja saskarnes:**

- **Pieteikšanās forma:**
  - Pieteikšanās formai ir jābūt laukiem lietotāja lietotājvārdam un parolei.
  - Pieteikšanās formai ir jābūt pogai "Login", lai autentificētu lietotāju.
  - Pieteikšanās formai ir jābūt pogai "Quit", lai var viegli beigt programmu
- **Uzdevuma iesniegšanas forma:**
  - Uzdevuma iesniegšanas formai ir lauki uzdevuma nosaukumam, steidzamības pakāpei un izpildes datumam.
  - Uzdevuma iesniegšanas veidlapā ir poga "Save", lai iesniegtu uzdevumu.
- **Uzdevuma skatīšanas forma:**
  - Uzdevumu skatīšanas lapā tiek parādīti visi lietotāja iesniegtie uzdevumi.
  - Uzdevuma skatīšanas lapā ir jāparāda uzdevuma nosaukums, steidzamības līmenis un izpildes datums katram uzdevumam.



# Programmatūras izstrādes plāns

Programmai sekoju ūdenskrituma izstrādes modeli.

## Plānošanas posms:

- Projekta prasību un apjoma noteikšana
- Projekta plāna un grafika sastādīšana

## Izstrādes fāze:

- Pieteikšanās formas un autentifikācijas sistēmas izstrāde
- Uzdevuma iesniegšanas formas izstrāde un integrācija ar MySQL datu bāzi
- Uzdevumu apskates lapas izstrāde
- Lietotāja interfeisa dizaina ieviešana
- Koda pārskatīšanas un testēšanas veikšana

## Pārbaudes fāze:

- Funkcionālās pārbaudes veikšana
- Sistēmas testēšanas veikšana
- Testa laikā atklāto defektu vai problēmu novēršana
- Programmas pabeigšana

## Atklūdošanas un akcepttestēšanas pārskats

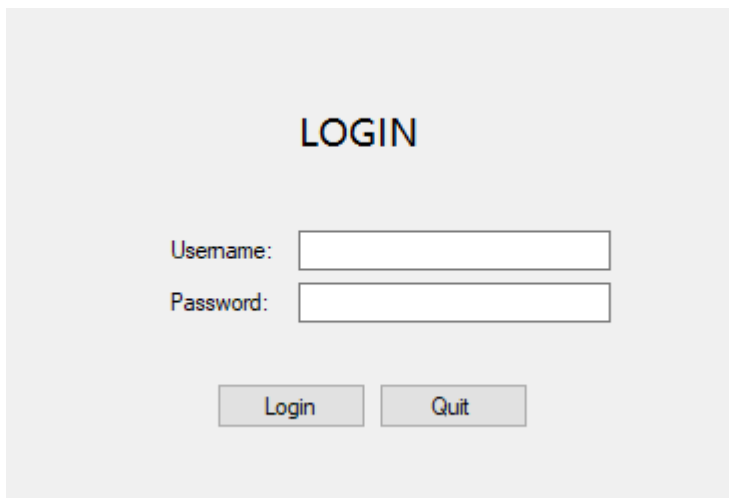
Pēc pārbažu veikšanas es identificēju vairākus defektus:

- Uzdevuma iesniegšanas kļūda: es atklāju problēmu, kuras dēļ uzdevuma iesniegšanas veidlapa nedarbojās pareizi, un lietotāji nevarēja iesniegt uzdevumus. Es novērsu šo problēmu, atjauninot kodu, kas apstrādā veidlapas iesniegšanu.
- Login kļūda: es atklāju problēmu, kuras dēļ login forma nedarbojās pareizi, un lietotāji nevarēja pieslēgties. Es novērsu šo problēmu, atjauninot kodu, kas apstrādā autentifikācijas procesu.
- Databāzes savienošana: es atklāju problēmu, kuras dēļ, nevar pieslēgt databāzi pie vs projekta. Šo problēmu, manliekas, var tikai risināt visu atinstalējot un instalējot atpakaļ + pārtaisīt projektu.



# Lietotāja ceļvedis

1. Ieslēdzot programmu ir divas opcijas:
  - a. Login - Ievadot savus piekļuves datus un piespiežot tiks tālāk uz "Assignments"
  - b. Quit - Aiztaisīs aplikāciju



A screenshot of a login interface. At the top, the word "LOGIN" is centered in a bold, black, sans-serif font. Below it, there are two input fields. The first is labeled "Username:" and the second is labeled "Password:". Both labels are in a small, black, sans-serif font. Below the input fields, there are two buttons: "Login" and "Quit". Both buttons are rectangular with a light gray background and a thin black border. The "Login" button is on the left and the "Quit" button is on the right.

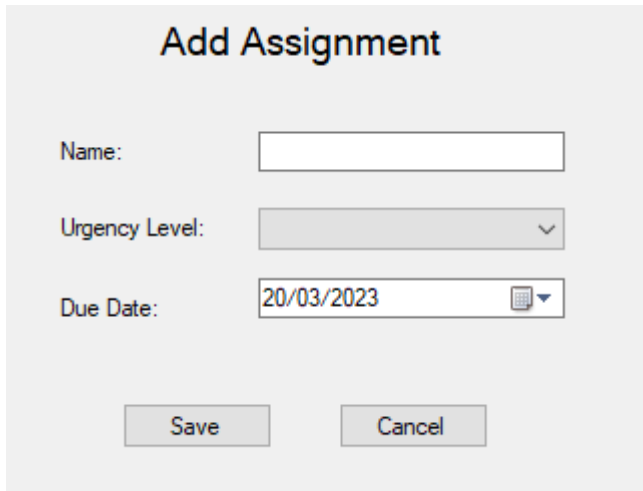
2. "Assignments" formā var redzēt visus savus uzdevumus, un ir poga "Add new"



A screenshot of an "Assignments" interface. The word "Assignments" is at the top left in a bold, black, sans-serif font. In the top right corner, there is a button labeled "Add New". The button is rectangular with a light gray background and a thin black border, and it is highlighted with a red rectangular border. Below the button, there is a large, empty gray rectangular area.

3. Piespiežot "Add new" atveras "Add assignment" forma, kur var ievadīt uzdevuma nosaukumu, līdz kuram jāiesniedz un cik svarīgi tas uzdevums ir.

4. "Add assignment" formā ir divas pogas
- a. Save - Saglabās uzdevumu
  - b. Cancel - Aiztaisīs "Add assignment" formu



**Add Assignment**

Name:

Urgency Level:

Due Date:

# Piemērotās licences pamatojums

Šī programma ir Brīvā programmatūra. Programmu var lietot, izplatīt un pat mainīt, bet mainīšanas un izplatīšanas gadījumā, tiek saglabātas norādes uz manu vārdu un oriģinālo programmu.

## **MySQL:**

Es izmantoju MySQL kā savu datu bāzes pārvaldības sistēmu. MySQL ir licencēta saskaņā ar GNU vispārējo publisko licenci (GPL), kas ir bezmaksas, copyleft licence, kas nodrošina, ka programmatūra paliek atvērtā pirmkoda un brīvi pieejama modificēšanai un izplatīšanai.

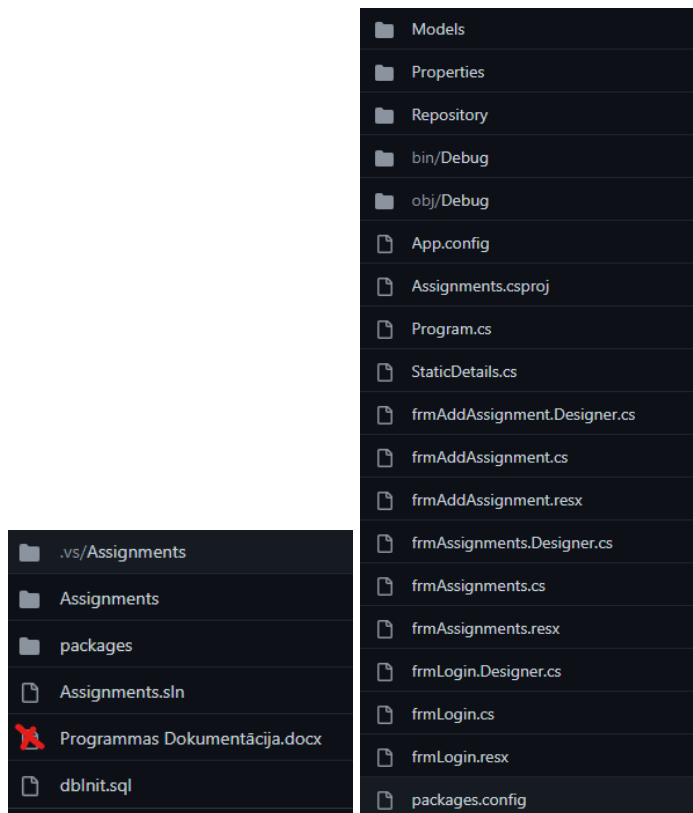
## **C# Visual Studio:**

Kā izstrādes vidi izmantoju C# Visual Studio. C# Visual Studio ir licencēta saskaņā ar Microsoft programmatūras licences noteikumiem, kas ir patentēta licence, kas ļauj lietotājiem izmantot programmatūru personiskiem vai komerciāliem mērķiem, ja vien tie atbilst licences noteikumiem.

## **Bibliotēkas:**

Savā projektā izmantoju MySql.Data bibliotēku, lai strādātu ar MySQL datu bāzi. MySql.Data ir licencēta saskaņā ar GNU vispārējo publisko licenci (GPL)

# Projekta failu struktūra



dbInit.sql - Fails ko var palaist, lai databāzē tiek uztaisītas vajadzīgās tabulas prieks programmas. Iekļauj arī dažus pagaidu lietotājus, kuriem var mainīt datus, lai vieglāk uztaisītu savu user.

StaticDetails.cs - nosaka databāzes savienojuma virkni un uztaisa int ar lietotāja ID (kurš ir ielogojies), priekš pārējām formām.

frmAddAssignment.cs - Add assignment formas kods, kas ļauj lietotājam pievienot uzdevumus databāzei, vai aiztaisīt formu.

frmAssignments.cs - Assignments formas kods, kas parāda visus lietotāja uzdevumus, kas ir saglabāti databāzē un ļauj atvērt Add Assignment formu.

frmLogin.cs - Login formas kods, kas ļauj lietotājam pieslēgties pie databāzes lietojot savu lietotājevārdu un paroli, kā arī ļauj izslēgt aplikāciju ar "Cancel pogu".

## dbInit.sql

```
CREATE TABLE users (  
  id INT NOT NULL AUTO_INCREMENT,  
  username VARCHAR(50) NOT NULL,  
  password VARCHAR(255) NOT NULL,  
  email VARCHAR(100) NOT NULL,  
  PRIMARY KEY (id)  
);
```

```
CREATE TABLE assignments (  
  id INT NOT NULL AUTO_INCREMENT,  
  name VARCHAR(255) NOT NULL,  
  urgency_level varchar(255) NOT NULL,  
  due_date DATE NOT NULL,  
  user_id INT NOT NULL,  
  PRIMARY KEY (id),  
  FOREIGN KEY (user_id) REFERENCES users(id)  
);
```

```
INSERT INTO users (username, password, email) VALUES ('john',  
'password', 'john_doe@example.com');
```

```
INSERT INTO users (username, password, email) VALUES ('jane',  
'password', 'jane_smith@example.com');
```

```
INSERT INTO users (username, password, email) VALUES ('bob',  
'password', 'bob\_smith@example.com');
```

## StaticDetails.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Assignments
{
    public class StaticDetails
    {
        public static int LoggedInUserId { get; set; }

        public static string ConnectionString {
            get
            {
                return
                "Server=localhost;Database=assignmentsDb;Uid=root;Pwd=admin;";
            }
        }
    }
}
```

## frmAddAssignment.cs

```
using Assignments.Models;
using Assignments.Repository;
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Xml.Linq;

namespace Assignments
{
    public partial class frmAddAssignment : Form
    {
        private readonly AssignmentsRepository _repository;
        public frmAddAssignment()
        {
            InitializeComponent();
            _repository = new AssignmentsRepository();
            cbUrgencyLevel.SelectedIndex = 0;
        }

        private void btnCancel_Click(object sender, EventArgs e)
        {
            DialogResult = DialogResult.Cancel;
        }

        private void btnSave_Click(object sender, EventArgs e)
        {
            if (string.IsNullOrEmpty(tbName.Text))
            {
                MessageBox.Show("Please enter a name.");
                return;
            }

            Assignment a = new Assignment();
            a.UserId = StaticDetails.LoggedInUserId;
            a.Name = tbName.Text;
            a.UrgencyLevel = cbUrgencyLevel.Text;
            a.DueDate = dtpDueDate.Value;

            bool added = _repository.AddAssignment(a);
            if (!added)
            {
                MessageBox.Show(_repository.ErrorMessage);
            }
            else
            {
                DialogResult = DialogResult.OK;
            }
        }
    }
}
```

## frmAssignments.cs

```
using Assignments.Repository;
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Assignments
{
    public partial class frmAssignments : Form
    {
        private readonly AssignmentsRepository _repository;
        public frmAssignments()
        {
            InitializeComponent();
            _repository = new AssignmentsRepository();
        }

        private void frmAssignments_Load(object sender, EventArgs e)
        {
            LoadData();
        }

        private void LoadData()
        {
            var data = _repository.GetByUserId(StaticDetails.LoggedInUserId);
            if (data == null)
            {
                MessageBox.Show(_repository.ErrorMessage);
            }
            else
            {
                dgvData.DataSource = data;
            }
        }

        private void frmAssignments_FormClosing(object sender, FormClosingEventArgs e)
        {
            Application.Exit();
        }

        private void btnAddNew_Click(object sender, EventArgs e)
        {
            frmAddAssignment frm = new frmAddAssignment();

            if (frm.ShowDialog(this) == DialogResult.OK)
            {
                LoadData();
            }
        }
    }
}
```



## frmLogin.cs

```
using Assignments.Repository;
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Assignments
{
    public partial class frmLogin : Form
    {
        private readonly UsersRepository _usersRepository;
        public frmLogin()
        {
            InitializeComponent();
            _usersRepository = new UsersRepository();
        }

        private void btnQuit_Click(object sender, EventArgs e)
        {
            Application.Exit();
        }

        private void frmLogin_Load(object sender, EventArgs e)
        {
        }

        private void btnLogin_Click(object sender, EventArgs e)
        {
            int loggedInUserId = _usersRepository.Login(tbUsername.Text, mtbPassword.Text);
            if (loggedInUserId == 0)
            {
                MessageBox.Show("User or password is incorrect", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error);
            }
            else if (loggedInUserId == -1)
            {
                // exception case
                MessageBox.Show(_usersRepository.ErrorMessage);
            }
            else
            {
                StaticDetails.LoggedInUserId = loggedInUserId;

                frmAssignments frm = new frmAssignments();
                frm.Show();
                this.Hide();
            }
        }
    }
}
```