МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

Кафедра автоматизації проектування енергетичних процесів і систем

Лабораторна робота №1

з дисципліни «Системи з розподіленими даними»

**Виконав:**

Студенти 5-го курсу

ТЕФ, гр. ТР-02мп

Круглий Д.В.

**Перевірив:**

Варава І.А.

**Київ-2021**

**Тема:** розробка клієнтського додатка Dropbox для Windows.

**Мета:** навчитись розробляти додатки для управління даними у хмарному сховищі Dropbox.

**Представлення коду програми**

using Dropbox.Api;

using Dropbox.Api.Files;

using Microsoft.Win32;

using System;

using System.Collections.Generic;

using System.IO;

using System.Linq;

using System.Text;

using System.Threading;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace MyDropboxApp

{

/// <summary>

/// Interaction logic for MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

private DropboxClient dbx = new DropboxClient("sl.AtK4klp0AEy0ALYfO3yBIdy1Z-5F8rT9fHJn5QFlj5KjgfF4YBG69a\_RVmfvAPY5yXeJ8qyzfyaogH9udumC7iiWpfPa54drE-cEqIGOgJOf81V497cqvyx73ZEa4pgo2KurXdKnI0Y");

public string NewFolderName { get; set; } = "Test";

public MainWindow()

{

InitializeComponent();

}

private void BtnGetStorage\_Click(object sender, RoutedEventArgs e)

{

tvFolders.Items.Clear();

var task = Task.Run(ListRootFolder);

task.Wait();

}

async Task ListRootFolder()

{

var list = await dbx.Files.ListFolderAsync(string.Empty);

foreach (var item in list.Entries.Where(i => i.IsFolder))

{

tvFolders.Dispatcher.BeginInvoke(System.Windows.Threading.DispatcherPriority.Normal, (ThreadStart)delegate

{

TreeViewItem t = new TreeViewItem();

t.Header = item.Name;

tvFolders.Items.Add(t);

});

}

foreach (var item in list.Entries.Where(i => i.IsFile))

{

tvFolders.Dispatcher.BeginInvoke(System.Windows.Threading.DispatcherPriority.Normal, (ThreadStart)delegate

{

TreeViewItem t = new TreeViewItem();

t.Header = item.Name;

tvFolders.Items.Add(t);

});

}

}

private void BtnUploadFile\_Click(object sender, RoutedEventArgs e)

{

OpenFileDialog openFileDialog = new OpenFileDialog();

if (openFileDialog.ShowDialog() == true)

{

string fullPath = openFileDialog.FileName;

string fileName = openFileDialog.SafeFileName;

var task = Task.Run(() => Upload(fullPath, fileName));

task.Wait();

}

}

async Task Upload(string fullPath, string fileName)

{

var mem = new MemoryStream(File.ReadAllBytes(fullPath));

try

{

var updated = await dbx.Files.UploadAsync("/" + fileName, WriteMode.Overwrite.Instance, body: mem);

}

catch (Exception e)

{

MessageBox.Show(e.ToString());

}

}

private void BtnCreateFolder\_Click(object sender, RoutedEventArgs e)

{

try

{

var task = Task.Run(() => CreateNewFolder("/" + NewFolderName));

task.Wait();

}

catch (AggregateException ex)

{

MessageBox.Show("Folder with the same name already exist");

}

}

async Task CreateNewFolder(string folderName) => await dbx.Files.CreateFolderV2Async(folderName);

private void BtnDeleteItem\_Click(object sender, RoutedEventArgs e)

{

try

{

string itemName = ((TreeViewItem)tvFolders.SelectedItem).Header.ToString();

var task = Task.Run(() => DeleteItem("/" + itemName));

task.Wait();

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

}

}

async Task DeleteItem(string itemName) => await dbx.Files.DeleteV2Async(itemName);

private void MiRename\_Click(object sender, RoutedEventArgs e)

{

try

{

TreeViewItem selectedItem = tvFolders.SelectedItem as TreeViewItem;

if (selectedItem != null)

{

string oldItemName = selectedItem.Header.ToString();

RenameDialog dialog = new RenameDialog();

dialog.NewName = oldItemName;

if (dialog.ShowDialog() == true)

{

string newItemName = dialog.NewName;

var task = Task.Run(() => RenameItem("/" + oldItemName, "/" + newItemName));

task.Wait();

}

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

}

}

async Task RenameItem(string oldItemName, string newItemName) => await dbx.Files.MoveAsync(oldItemName, newItemName);

}

}

**Приклад роботи програми**

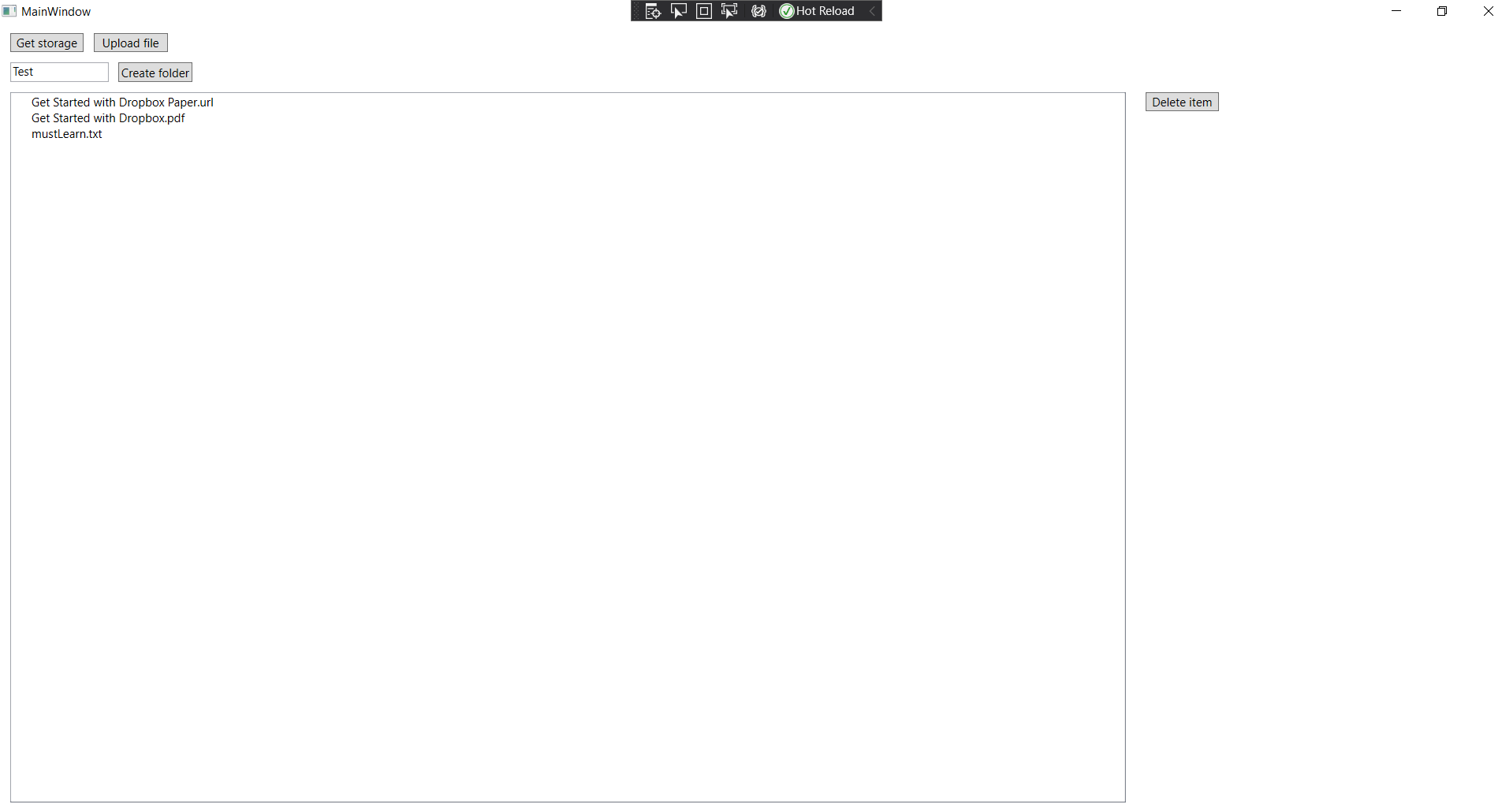


Рис. 1 – Головне вікно програми, де відображено каталог файлів

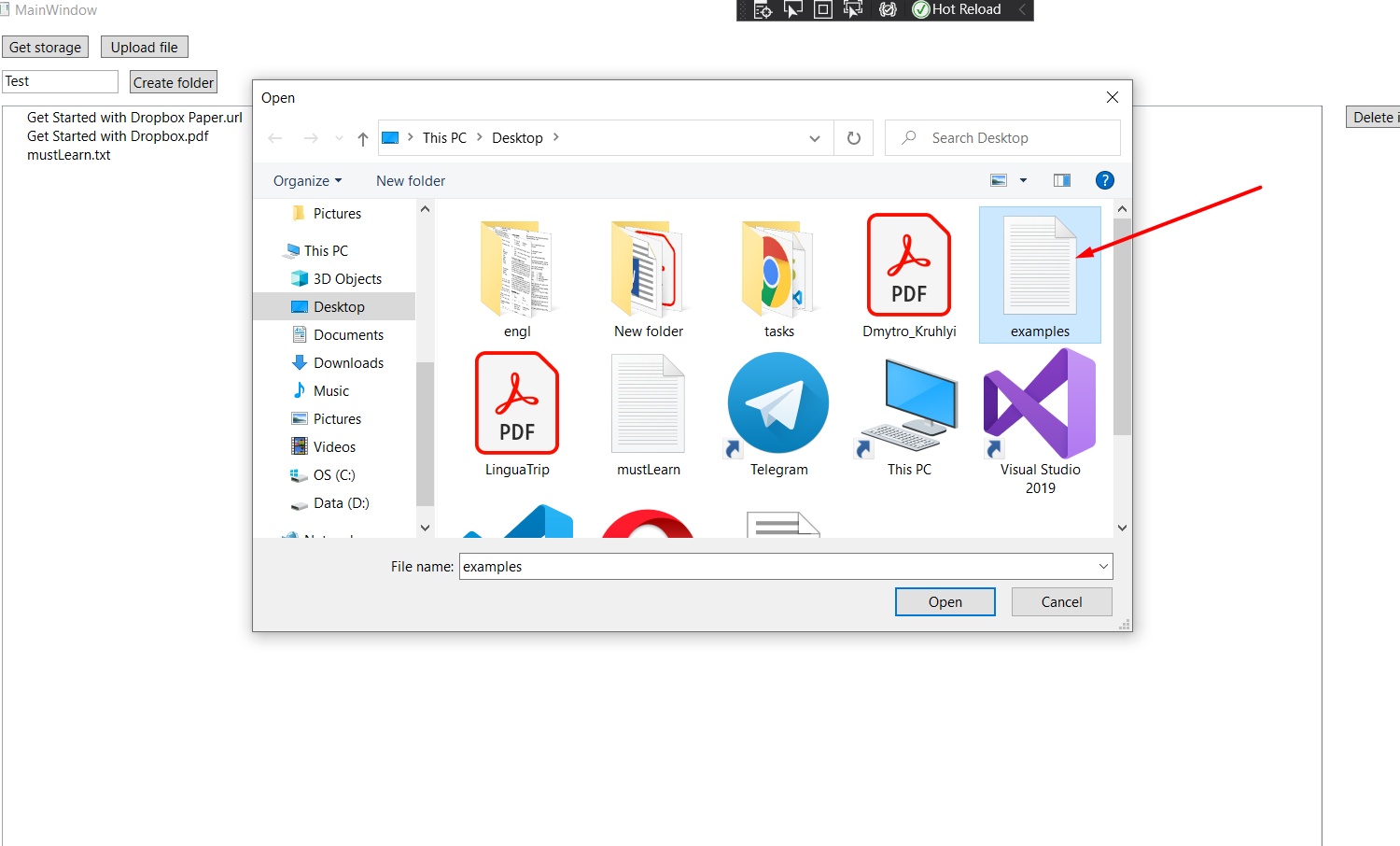


Рис. 2 – Завантаження файлу

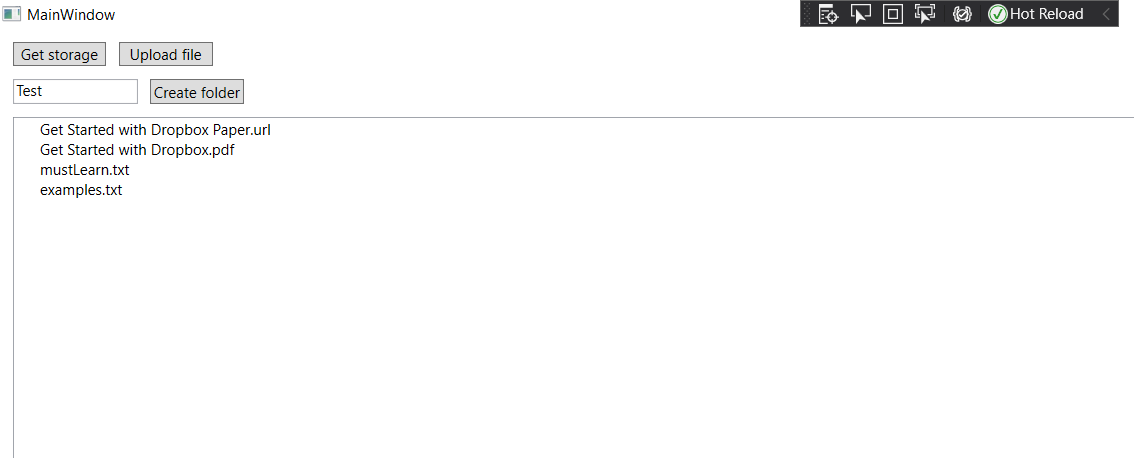


Рис. 3 – Перегляд завантаженого файлу

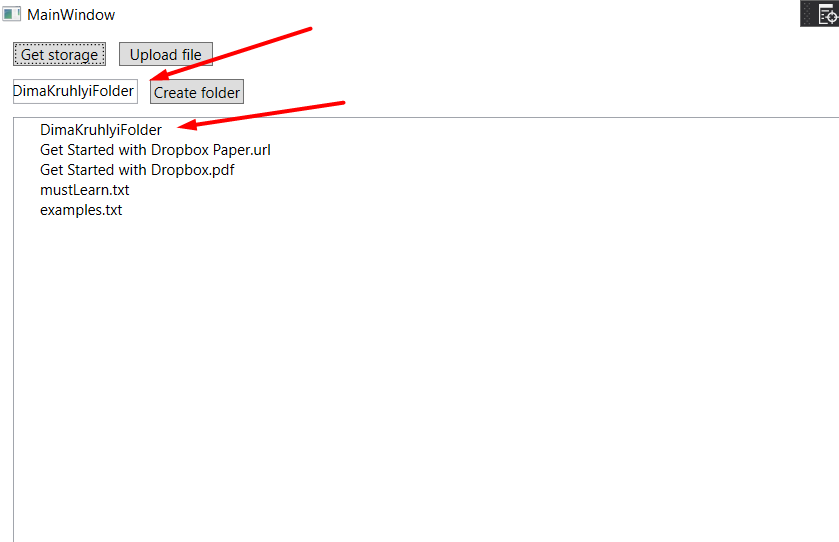
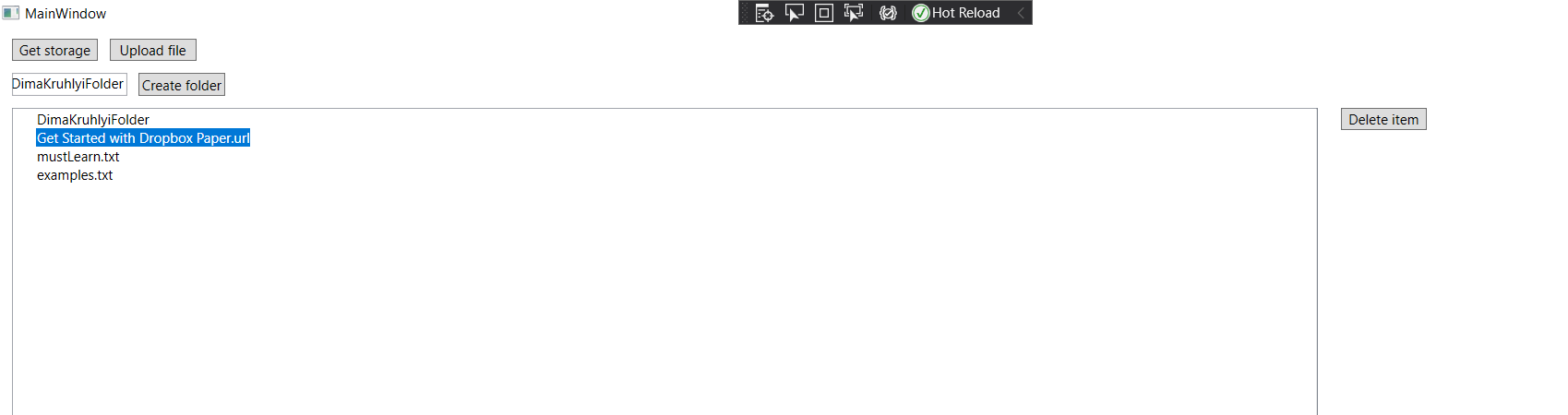


Рис. 4 – Створення і відображення директорії



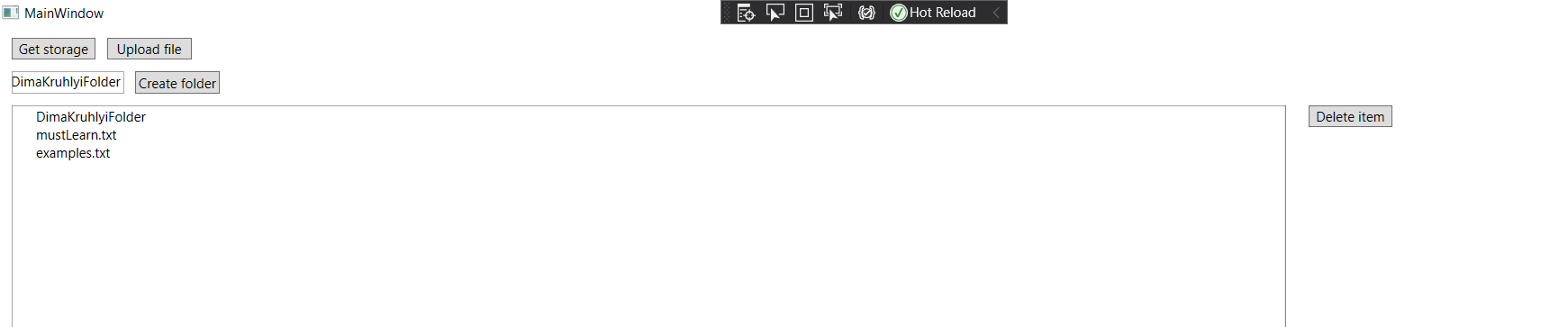


Рис. 5 – Видалення файлу

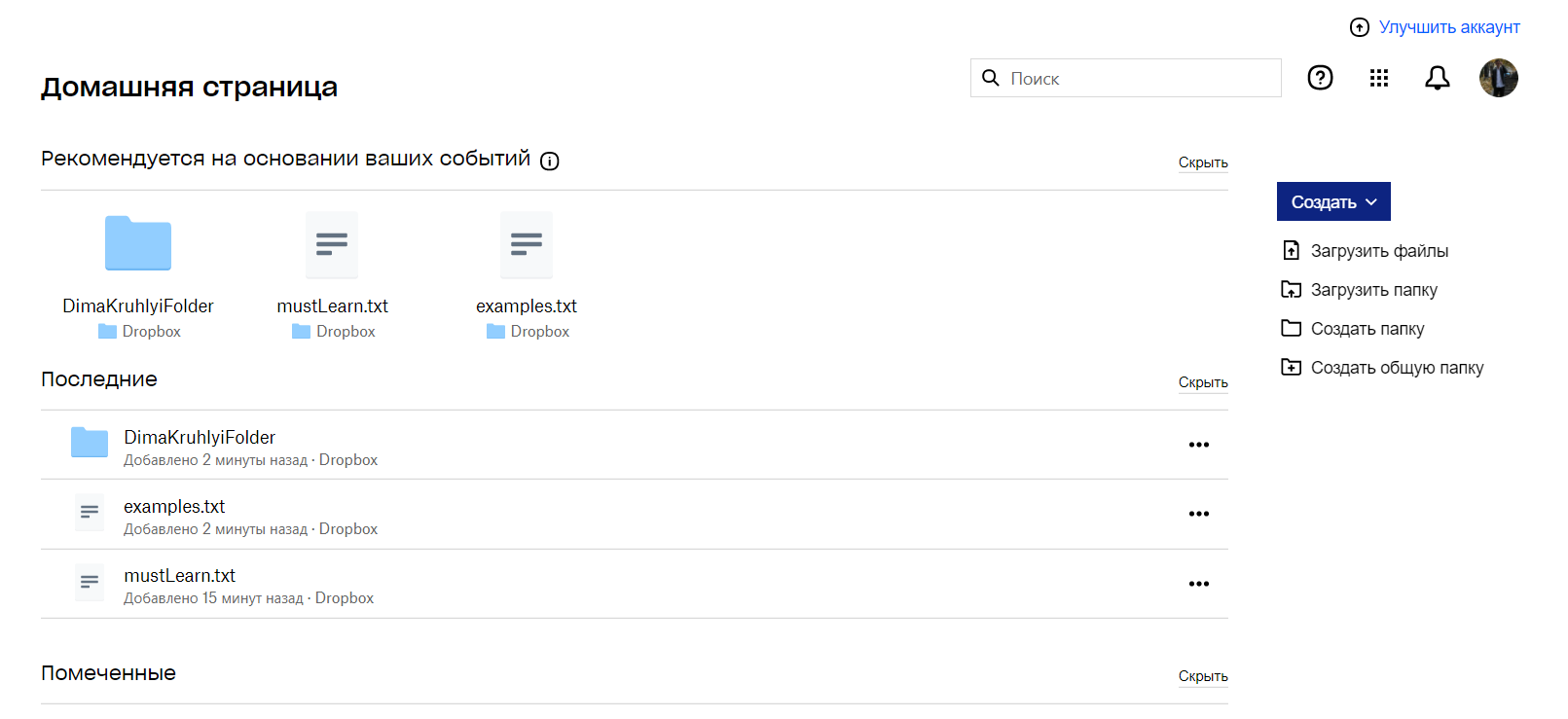


Рис. 6 – Перевірка наявності даних файлів

**Висновок:** виконавши дану лабораторну роботу, я навчився розробляти додатки для управління даними у хмарному сховищі Dropbox.