

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ЕЛЕКТРОНІКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК
СЕКЦІЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ПРОЕКТУВАННЯ

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

**на тему: «Віртуальна лабораторна робота
з визначення величини зерна металу»**

**за спеціальністю 122 «Комп'ютерні науки»,
освітньо-професійна програма «Інформаційні технології проектування»**

Виконавець роботи: студент групи ІТ-71 Малиновський Богдан Юрійович

**Кваліфікаційна робота бакалавра
захищена на засіданні ЕК
з оцінкою**

_____ «__» _____ 2021 р.

Науковий керівник

(підпис)

к.т.н., доц., Ващенко С. М.

(науковий ступінь, вчене звання, прізвище та ініціали)

Голова комісії

(підпис)

Шифрін Д. М.

(науковий ступінь, вчене звання, прізвище та ініціали)

Засвідчую, що у цій дипломній роботі немає
запозичень з праць інших авторів
без відповідних посилань.

Студент _____
(підпис)

Сумський державний університет
Факультет електроніки та інформаційних технологій
Кафедра комп'ютерних наук
Секція інформаційних технологій проектування
Спеціальність 122 «Комп'ютерні науки»
Освітньо-професійна програма «Інформаційні технології проектування»

ЗАТВЕРДЖУЮ

Зав. секцією ІТП

_____ В. В. Шендрик
«__» _____ 2021 р.

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА СТУДЕНТУ

Малиновський Богдан Юрійович

1 Тема роботи Віртуальна лабораторна робота з визначення величини зерна металу

керівник роботи Ващенко Світлана Михайлівна, к.т.н., доцент,

затверджені наказом по університету від «14» квітня 2021 р. №0181-VI

2 Строк подання студентом роботи «7» червня 2021 р.

3 Вхідні дані до роботи технічне завдання на розробку інформаційної системи «Віртуальна лабораторна робота з визначення величини зерна металу»

4 Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) аналіз програмних продуктів – аналогів, моделювання «Віртуальної лабораторної роботи з визначення величини зерна металу», моделювання «Віртуальної лабораторної роботи з визначення величини зерна металу»

5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) актуальність роботи, постановка задачі, аналіз програмних продуктів – аналогів, IDEF0-модель, діаграма варіантів використання, демонстрація програми

6. Консультанти розділів роботи:

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання _____ 01.10.2020 _____

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Огляд останніх досліджень	01.11.2020 – 30.11.2020	
2	Вибір засобів реалізації	01.12.2020 – 15.12.2020	
3	Постановка цілей необхідних для досягнення певного результату	16.12.2020 – 31.12.2020	
4	Складання технічного завдання	01.01.2021 – 17.01.2021	
5	Підготовка прототипу	18.01.2021 – 31.01.2021	
6	Створення макету дизайну додатку	01.02.2021 – 15.02.2021	
7	Реалізація логіки додатку	16.02.2021 – 15.04.2021	
	Перевірка працездатності додатку	16.04.2021 – 25.04.2021	
	Оформлення документації	26.04.2021 – 30.04.2021	

Студент

_____ (підпис)

Малиновський Б.Ю.

Керівник роботи

_____ (підпис)

к.т.н., доц. Ващенко С.М.

РЕФЕРАТ

Тема кваліфікаційної роботи бакалавра «Віртуальна лабораторна робота з визначення величини зерна металу».

Пояснювальна записка складається зі вступу, 3 розділів, висновків, списку використаних джерел із 25 найменувань, додатків. Загальний обсяг роботи – 81 сторінка, у тому числі 34 сторінки основного тексту, 3 сторінки списку використаних джерел, 43 сторінки додатків.

Кваліфікаційну роботу бакалавра присвячено розробці віртуальної лабораторної роботи з визначення величини зерна металу.

В першому розділі наведено огляд останніх досліджень за темою роботи та проведено аналіз програмних продуктів - аналогів.

В другому розділі проведено моделювання «Віртуальної лабораторної роботи з визначення величини зерна металу», змодельовано контекстну діаграму, IDEF0 та діаграму варіантів використання. Проведено проектування моделі інформаційної бази.

В третьому розділі детально описано етапи практичної реалізації проекту та наведено приклади використання програмного додатку, що демонструють його працездатність.

Ключові слова: ВІРТУАЛЬНА ЛАБОРАТОРНА РОБОТА, ДИСТАНЦІЙНЕ НАВЧАННЯ, БАЛ ЗЕРНА МЕТАЛУ, АНАЛІЗ МЕТАЛІВ, C#, XML .

ЗМІСТ

ВСТУП.....	6
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	7
1.1 Огляд останніх досліджень і публікацій.....	7
1.2 Аналіз програмних продуктів - аналогів.....	8
1.3 Постановка задачі.....	9
2 МОДЕЛЮВАННЯ ТА ПРОЕКТУВАННЯ.....	11
2.1 Моделювання «Віртуальної лабораторної роботи з визначення величини зерна металу».....	11
2.2 Проектування моделі інформаційної бази.....	19
3 РОЗРОБКА ПРОГРАМНОГО ПРОДУКТУ.....	21
3.1 Програмна реалізація.....	21
3.2 Використання програмного продукту.....	31
ВИСНОВКИ.....	34
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	35
ДОДАТОК А.....	38
ДОДАТОК Б.....	49
ДОДАТОК В.....	61

ВСТУП

В сучасному світі все частіше впроваджуються інформаційні технології в освітній процес. Це дозволяє більш зручно та легко організовувати навчальний процес. Фактично в сучасних умовах стає обов'язковим використовувати веб-ресурси, різні додатки та інші привілеї сучасного інформаційного світу як для тестування знань студентів, так і для практичних занять. Звісно, процес навчання стає більш зручним, коли застосовуються матеріали та лабораторні роботи в електронному вигляді. Це пришвидшує процес підготовки та виконання завдання з точки зору студента, та спрощує процес перевірки для викладача.

Проте студенти технічних спеціальностей, що вивчають дисципліни, пов'язані з дослідженням металів, все ще використовують мікроскоп та фотокартки мікроструктур в процесі виконання лабораторних робіт, де вони мають визначати величину зерна металу. Більше того, вони вручну підставляють у відповідні формули значення, отримані під час дослідження, та обчислюють бал зерна. Тому було прийнято рішення про розробку програми для виконання лабораторних робіт з визначення величини зерна для більш зручного розрахунку балу зерна та уникнення випадкових помилок при математичних обчисленнях.

Тому мета роботи - розробити програму для виконання віртуальних лабораторних робіт з визначення величини зерна.

Для досягнення мети проекту необхідно виконати перелік таких робіт:

- дослідити програмні продукти-аналоги;
- скласти сценарій роботи програми-віртуальної лабораторної роботи та алгоритм відповідних розрахунків;
- провести моделювання роботи програмного забезпечення;
- виконати програмну реалізацію.

За результатами проведеної роботи опубліковано тези доповіді на конференції «ІМА 2021» [23].

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Огляд останніх досліджень і публікацій

З кожним роком навчальні заклади все більше впроваджують інформаційні технології в освітній процес. На сьогоднішній день лекція без мультимедійної презентації або лабораторні роботи без використання комп'ютера стають рідкістю. Переважна більшість викладачів намагаються спростити процес навчання студентів, звільняючи їх від непотрібної роботи, яку за них може зробити комп'ютер, та зменшуючи вірогідність допущення помилки. Особливо актуальним питання забезпечення такими навчальними матеріалами стало в епоху пандемії, що зараз переживає світ.

У публікації «Інформаційно-комунікаційні технології навчання як засоби реалізації віртуальних лабораторних робіт з фізики у майбутніх учителів хімії і біології» [24] аналізуються та конкретизуються підходи щодо впровадження інформаційно-комунікаційних технологій під час проведення лабораторних занять з фізики. Також наводяться приклади віртуальних лабораторних робіт з фізики, яку вже використовуються викладачами у навчальному процесі.

Іншим гарним прикладом є стаття «Використання інформаційних технологій у навчальному процесі та створення електронного підручника» [25], яка наголошує на тому, що інформація повинна подаватися із застосуванням необхідної наочності, що збуджує механізм сприйняття, інтерес до даної галузі знань і до навчання взагалі. Також дана стаття описує мета створення електронного підручника, що являє собою програмно-методичний комплекс, що забезпечує можливість студентам, під керівництвом викладача або самостійно, засвоювати навчальний курс, згідно із програмами дисципліни, з використанням можливостей комп'ютерних технологій.

Як можна побачити з огляду даних двох статей, процес залучення інформаційних технологій у навчальний процес поширюється на найрізноманітніші

галузі, що зайвий раз підтверджує актуальність розробки віртуальної лабораторної роботи з дослідження балу зерна металу.

1.2 Аналіз програмних продуктів - аналогів

На сьогоднішній день усе частіше використовують різноманітні програмні продукти для аналізу структури зерна металу. Основна їх частина представляє собою програми для обробки зображень, що мають спеціалізовані металографічні модулі для визначення балу зерна згідно з відповідним ГОСТ. Прикладами таких програм є «IMAGE-SP», Видео-ТесТ, Autoscan, SIAMS. Усі вони мають приблизно однакові принципи побудови, які описані у публікації «Определение балла зерна стали компьютерными методами» [21].

Програма «IMAGE-SP» призначена для реєстрації зображень структури за допомогою цифрових камер, а також подальшої обробки отриманих зображень і їх серій. Важливою особливістю програми «IMAGE-SP» є можливість роботи з зображеннями з високою роздільною здатністю. Результат обробленого зображення виводиться у вигляді таблиці для кожного виділеного зерна окремо і дозволяє оцінити геометричні параметри і орієнтацію кожного зерна або фази. Виділені зерна відповідно їх розмірам фарбуються в різні кольори.

Іншим додатком-аналогом є AxioVision - програмне забезпечення для досліджень і розробок в галузі матеріалознавства, яке призначене для отримання, обробки та аналізу зображень, і управління моторизованими частинами мікроскопа AxioVision від Carl Zeiss. Для простоти експлуатації дана програма пропонує добре адаптований графічний інтерфейс зі структурованим і легким в освоєнні робочим процесом, адаптованим до промислових вимог. Це програмне забезпечення дозволяє легко дослідити взаємозв'язок між структурами різних матеріалів і аналізувати їх властивості.

Сучасні програми по обробці зображень представляють значну складність для користувачів. Як правило, такі програми створюються для комплексної обробки і містять безліч модулів і функцій для аналізу об'єктів різного роду в області біології, медицини і т.д. Що стосується металографії, вони містять досить широкий спектр різних параметрів структури. Тому часто користувач не знає, як саме вирішити ту чи іншу задачу комп'ютерними методами і які параметри для цього використовувати. При цьому можливі помилки методичного плану, які в результаті можуть бути наслідком неправильного металознавчого висновку. Також важливе володіння комп'ютерною технікою на досить високому рівні, що не завжди має місце, особливо на виробництві. Тому дії оператора з комп'ютерного аналізу структури повинні бути строго регламентовані.

1.3 Постановка задачі

Метою роботи є розробка інтерактивної системи для визначення величини зерна металу. Розроблена система буде мати попит на кафедрах з дослідження металу у вищих навчальних закладах. Використовуючи систему викладачі зможуть зменшити обсяг витраченого часу та людських ресурсів на дослідження балу зерна металу.

Розроблена інтерактивна система повинна містити наступні функції:

- визначення балу зерна металу методом підрахунку;
- визначення балу зерна металу методом підрахунку при пересіченні;
- визначення балу зерна металу методом порівняння;
- створення звіту з лабораторної роботи у вигляді документа MS Office Word.

Оскільки існують різні методи дослідження балу зерна металу і кожен з них не дає точного результату, при дослідженні балу зерна використовують водночас різні методи для отримання більш наближеного результату.

Для досягнення мети проекту потрібно виконати такий перелік робіт:

- пошук аналогових продуктів;
- пошук документації по дослідженню металів під час лабораторних робіт у навчальних закладах;
- пошук бібліотек для роботи з графічними зображеннями;
- розробка алгоритму для отримання результатів дослідження зерна металу та їх використання у обчисленні балу зерна.

Для реалізації додатку необхідно обрати мову програмування придатну для розробки за технологією Windows Forms. Мова програмування C# є вибором більшості розробників, працюючих з додатками даного типу.

Серед середовищ розробки було обрано Visual Studio, що вважається одним з найкращих серед конкуруючих аналогів, що використовуються для розробки додатків за даною технологією.

Сценарій роботи студента з програмою є досить простим для розуміння. На початковій сторінці студент бачить основну інформацію про лабораторну роботу. Після натискання кнопки «Почати» він переходить на сторінку з методом підрахунку, де він також бачить зображення отриманої мікроструктури. Після виконання завдання за допомогою першого методу студент переходить на сторінку з методом підрахунку при пересіченні, після виконання якого на сторінку з методом порівняння. Після виконання завдання за допомогою усіх трьох методів, лабораторна робота може бути завершена та передана на перевірку викладачу.

Повне технічне завдання на розробку наведено в додатку А.

Планування робіт представлено у додатку Б.

2 МОДЕЛЮВАННЯ ТА ПРОЕКТУВАННЯ

2.1 Моделювання «Віртуальної лабораторної роботи з визначення величини зерна металу»

Після проведення аналізу предметної області, визначення потреби системи та актуальності, переходимо до етапу моделювання та проектування віртуальної лабораторної роботи. Було виконано розробку діаграми нотації IDEF0, де описано послідовності усіх процесів реалізації проекту, та діаграми Use-Case, в яких було продемонстровано всі можливі функції, що користувач може здійснювати, користуючись системою.

2.1.1 Діаграми нотації IDEF0

Нотація IDEF0 описує бізнес-процеси. Щоб легше представити бізнес-процес проект представлено у вигляді прямокутника. До його сторін під'єднанні керуючі стрілки. Кожна стрілка відповідає за різний тип даних:

- ліва стрілка – це вхідні дані системи;
- права стрілка – це вихідні дані системи;
- верхня стрілка – це дані керування: документ, фіксує як система повинна працювати та бути реалізована;
- нижня стрілка – це дані механізму: програмне забезпечення або персонал, що залучаються до реалізації системи.

Процес «Виконання віртуальної лабораторної роботи»:

- вхідні дані – фотографія мікроструктури та стандартні зображення мікроструктур.
- вихідні дані – визначений бал зерна металу, звіт з лабораторної роботи.
- управління – методики обчислення балу зерна за методом підрахунку, методом підрахунку при пересіченні та методом порівняння.

- механізми – додаток «Віртуальна лабораторна робота з визначення величина зерна металу», виконавець та технічне забезпечення.

На рис. 2.2 представлена контекстна діаграма процесу виконання віртуальної лабораторної роботи.

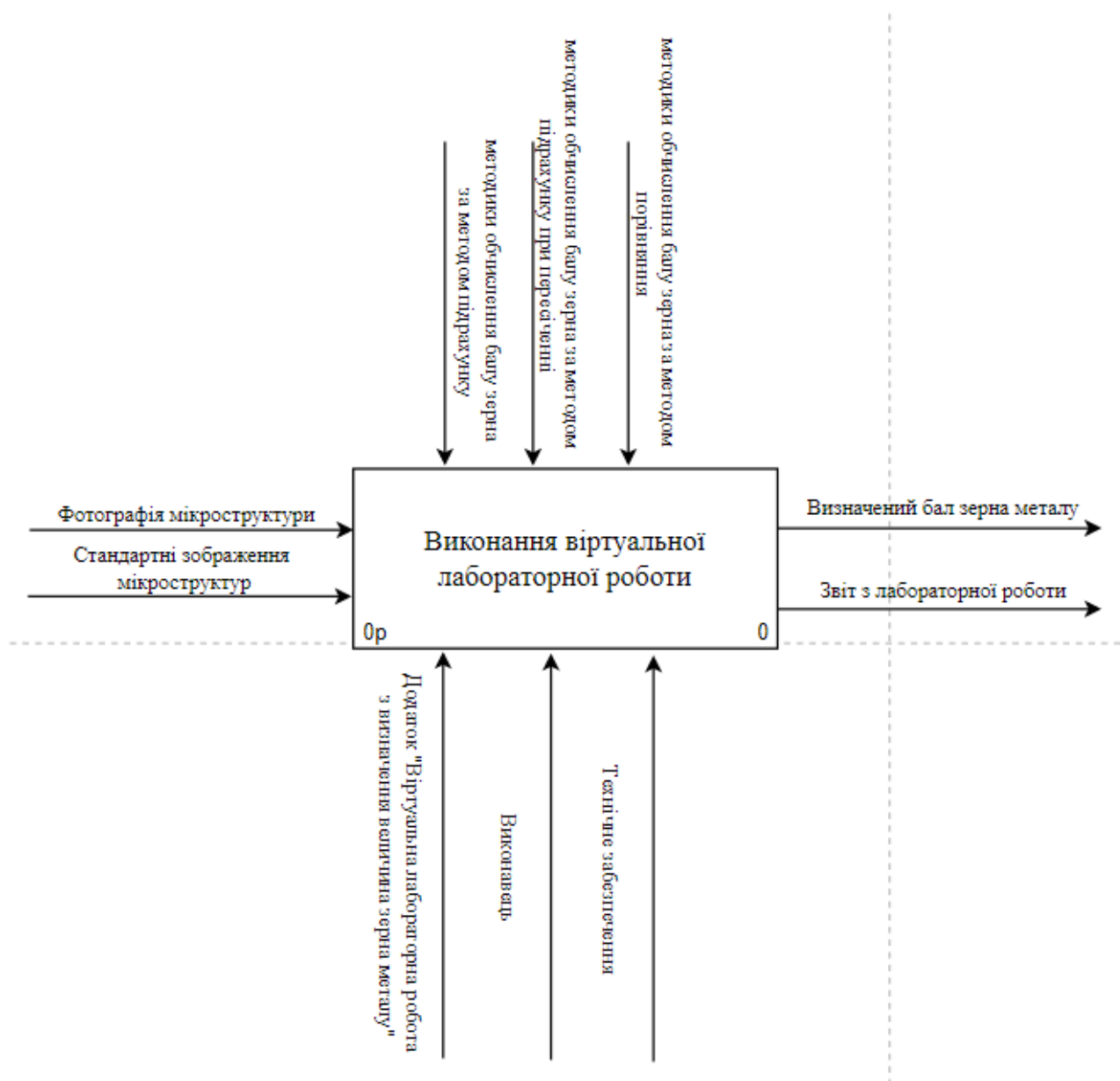


Рисунок 2.2 – Контекстна діаграма нотації IDEF0

Наступним кроком є декомпозиція реалізації системи на три головних процеси:

- визначення балу зерна методом підрахунку;
- визначення балу зерна методом підрахунку при пересіченні;
- визначення балу зерна методом порівняння.

Після декомпозиції, було розроблено діаграму першого рівня. Нижче представлено дані для діаграми:

- вхідні дані – фотографія мікроструктури та стандартні зображення мікроструктур.
- вихідні дані – визначений бал зерна металу, звіт з лабораторної роботи.
- управління – методики обчислення балу зерна за методом підрахунку, методом підрахунку при пересіченні та методом порівняння.
- механізми – додаток «Віртуальна лабораторна робота з визначення величина зерна металу», виконавець та технічне забезпечення.

На рис. 2.3 представлена діаграма першого рівня.

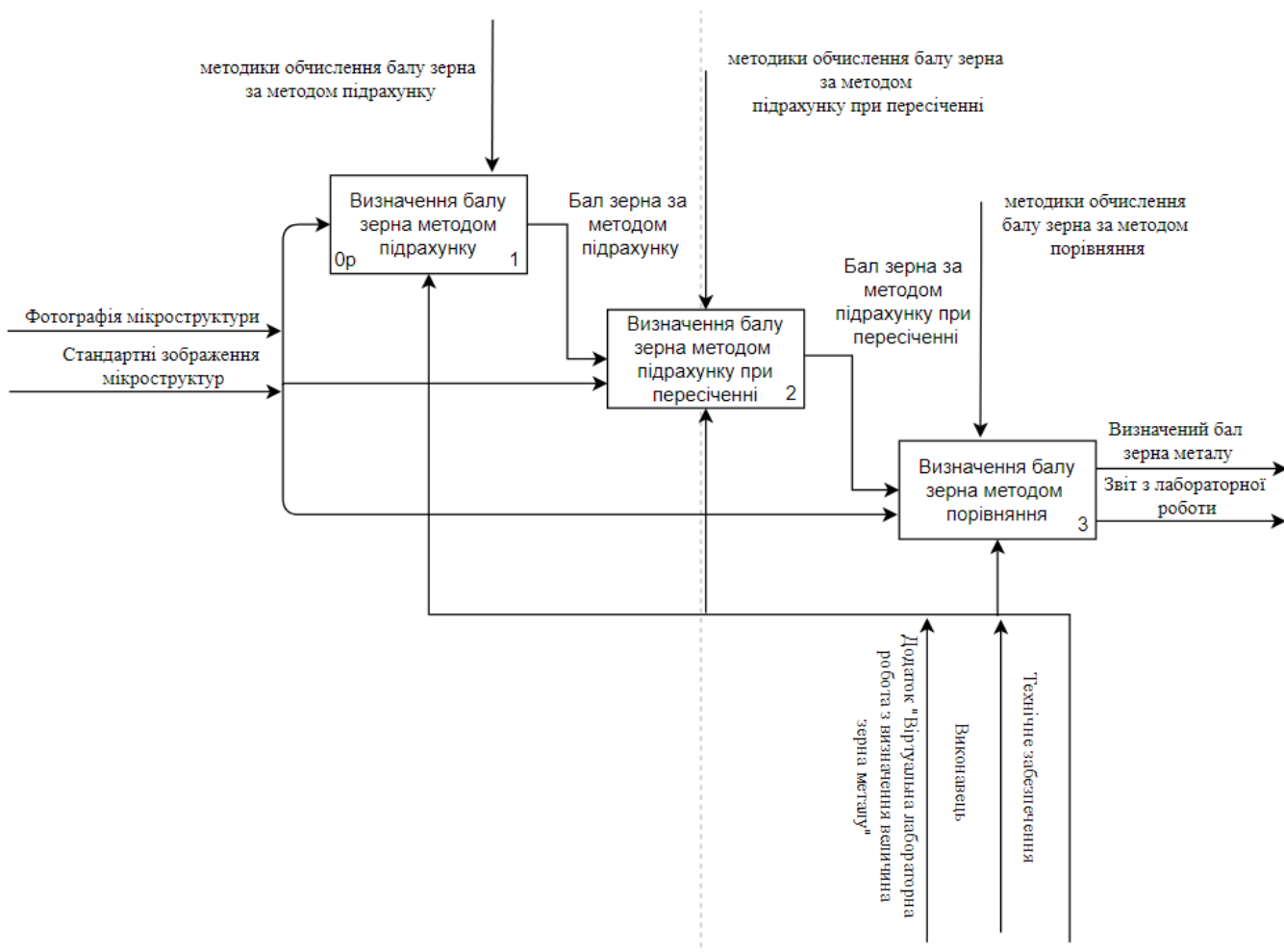


Рисунок 2.3 – Діаграма першого рівня

Далі виконується декомпозиція усіх головних процесів на основні етапи. Декомпозиція процесу «Визначення балу зерна методом підрахунку» визначає 3 основні етапи:

- визначення зерен усередині окружності;
- визначення зерен пересічених окружністю;
- визначення балу зерна методом підрахунку.

Дані діаграми декомпозиції:

- вхідні дані – фотографія мікроструктури.
- вихідні дані – визначений бал зерна металу.
- управління – методики обчислення балу зерна за методом підрахунку.

- механізми – додаток «Віртуальна лабораторна робота з визначення величина зерна металу», виконавець та технічне забезпечення.

Результат процесу «Визначення балу зерна методом підрахунку» - бал зерна металу за методом підрахунку отриманий в результаті обчислення за формулою, яка потребує таких значень, як кількість зерен усередині окружності та кількість зерен пересічених окружністю. На рис. 2.4 представлена діаграма декомпозиції процесу «Визначення балу зерна методом підрахунку».

Декомпозиція процесу «Визначення балу зерна методом підрахунку при пересіченні» розбиває його на 3 етапи:

- нанесення довільних відрізків на малюнок;
- обчислення потрібних значень;
- визначення балу зерна методом підрахунку при пересіченні.

Дані діаграми декомпозиції:

- вхідні дані – фотографія мікроструктури.
- вихідні дані – визначений бал зерна металу.
- управління – методики обчислення балу зерна за методом підрахунку при пересіченні.
- механізми – додаток «Віртуальна лабораторна робота з визначення величина зерна металу», виконавець та технічне забезпечення.

Результатом процесу «Визначення балу зерна методом підрахунку при пересіченні» є бал зерна металу за методом підрахунку при пересіченні отриманий в результаті обчислення за формулою, яка потребує таких значень, як сумарна довжина відрізків та загальне число перетинів меж зерен. На рис. 2.5 представлена діаграма декомпозиції процесу «Визначення балу зерна методом підрахунку при пересіченні».

Процес «Визначення балу зерна методом порівняння» складається з 2 етапів:

- порівняння мікроструктури з аналогами з ГОСТ;
- визначення балу зерна методом порівняння.

Дані діаграми декомпозиції:

- вхідні дані – фотографія мікроструктури та стандартні зображення мікроструктур.
- вихідні дані – визначений бал зерна металу та звіт з лабораторної роботи.
- управління – методики обчислення балу зерна за методом порівняння.
- механізми – додаток «Віртуальна лабораторна робота з визначення величина зерна металу», виконавець та технічне забезпечення.

Результатом процесу «Визначення балу зерна методом порівняння» є бал зерна металу за методом порівняння отриманий в результаті порівняння користувачем зображення даної мікроструктури з еталонними зображеннями, наведеними відповідно до ГОСТ, та отриманий звіт з лабораторної роботи з результатами трьох методів. На рис. 2.6 представлена діаграма декомпозиції процесу «Визначення балу зерна методом порівняння».

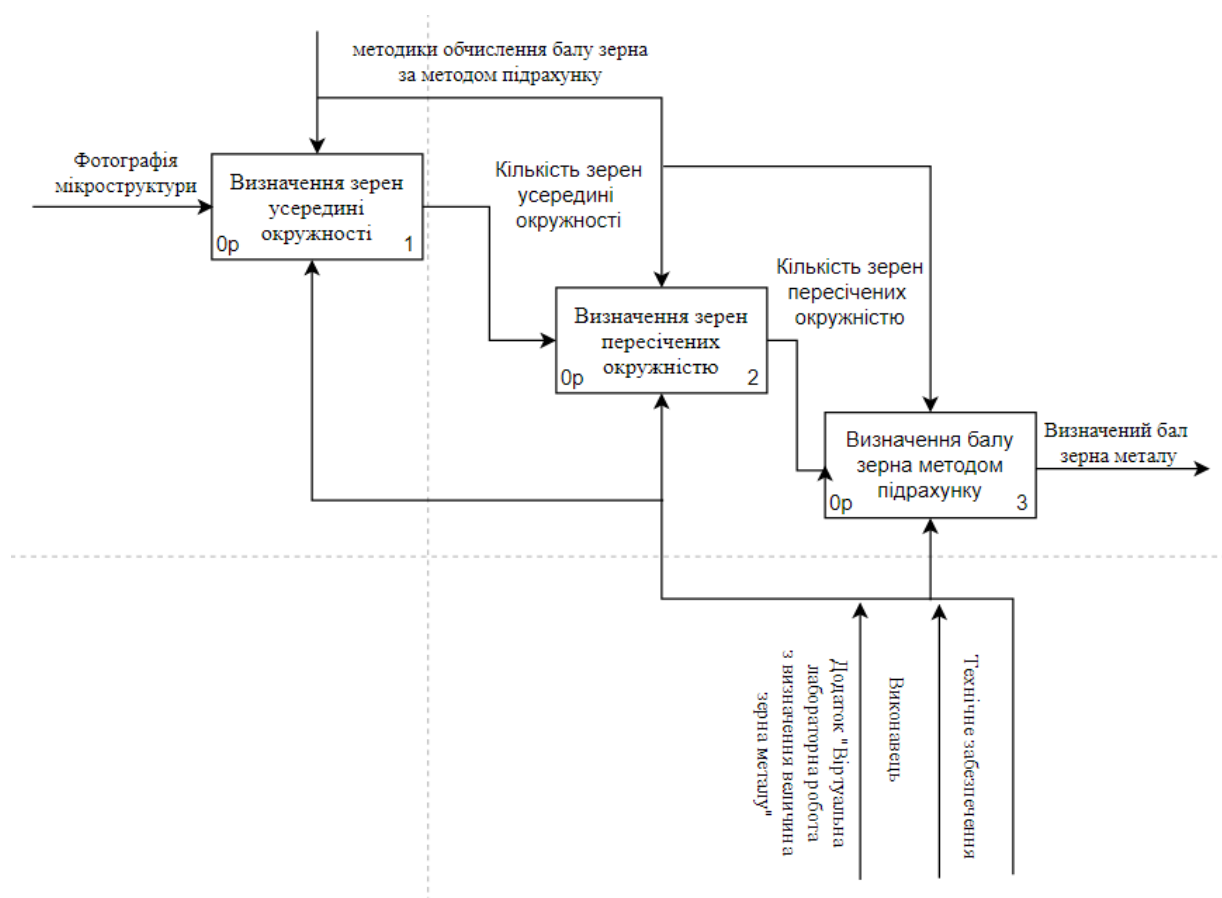


Рисунок 2.4 – Декомпозиція процесу «Визначення балу зерна методом підрахунку»

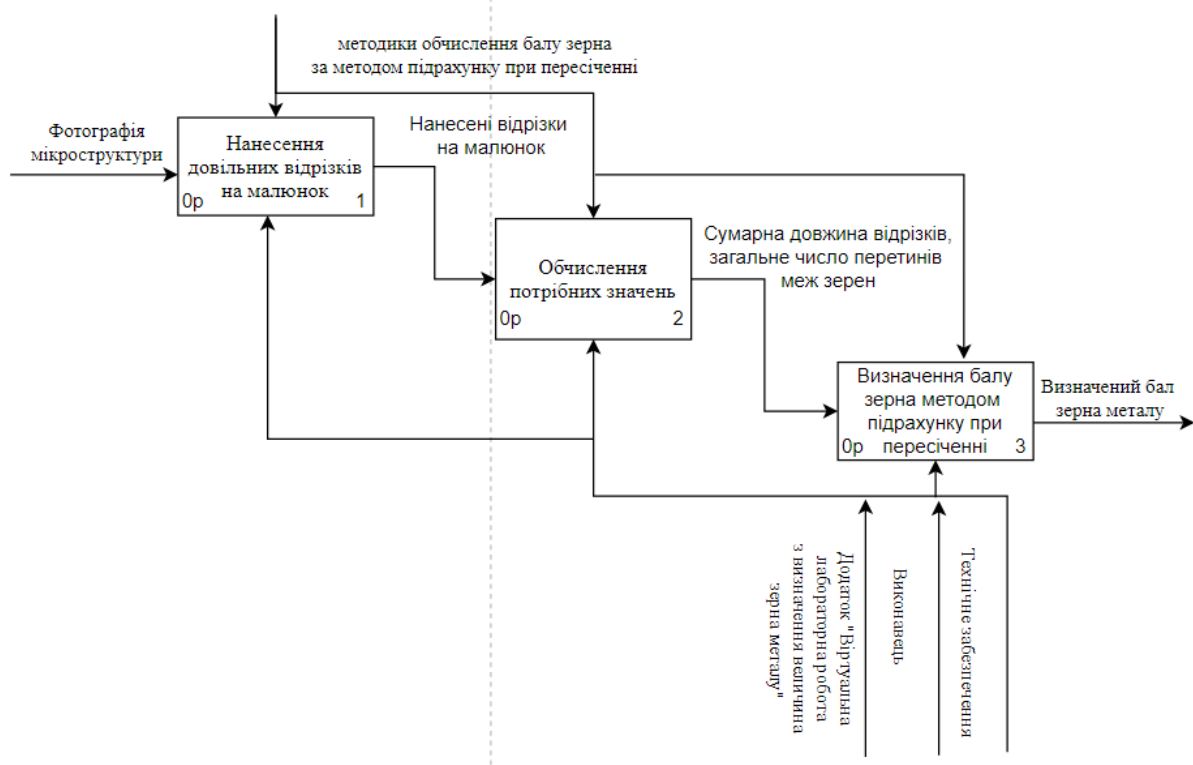


Рисунок 2.5 – Декомпозиція процесу «Визначення балу зерна методом підрахунку при пересіченні»

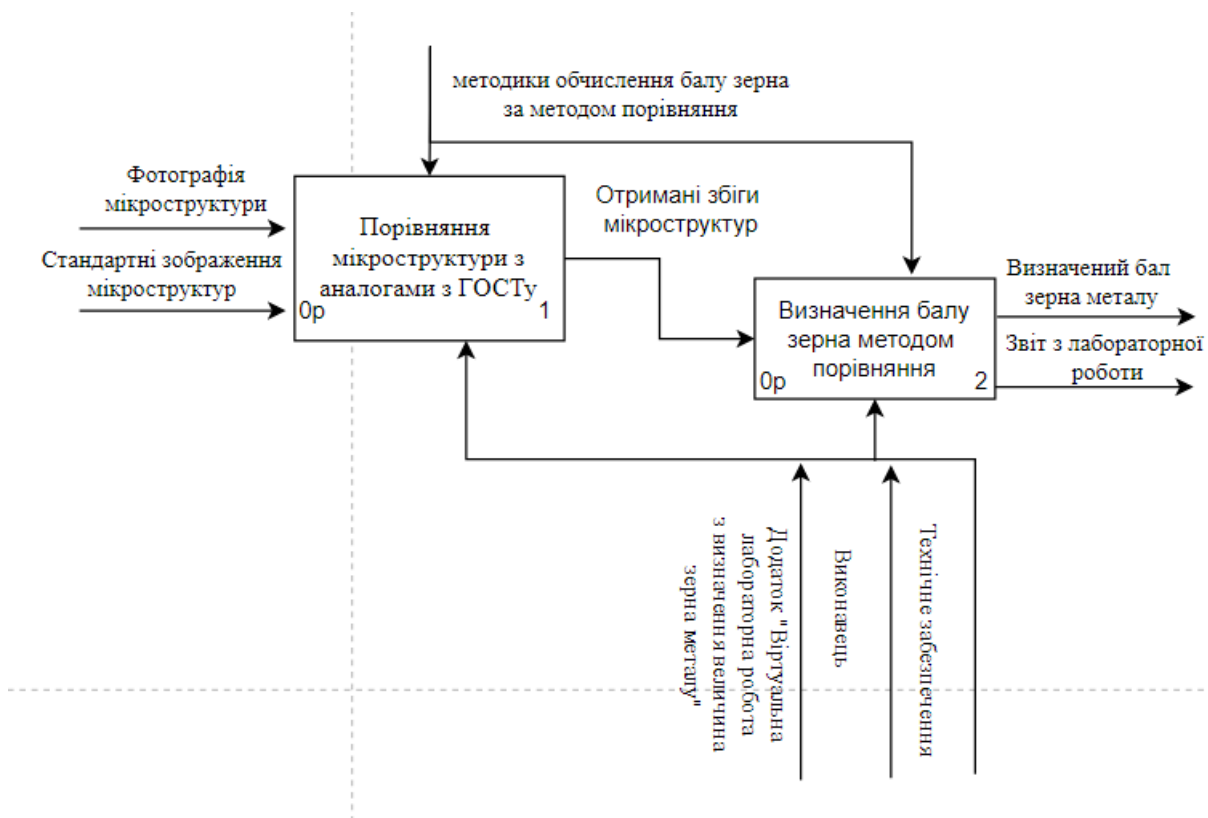


Рисунок 2.6 – Декомпозиція процесу «Визначення балу зерна методом порівняння»

2.1.2 Діаграма Use Case

Діаграма Use Case описує взаємодію користувача з системою. Для даного додатку було визначено одного актора: користувач. Нижче перелічено сценарії використання користувачем системи:

- обчислити методом підрахунку;
- обчислити методом підрахунку при пересіченні;
- обчислити методом порівняння;
- переглянути теорію;
- записати у текстовий документ.

Також було визначено артефакти у вигляді інформаційної бази та звітнього документу.

Було розроблено діаграму Use Case на основі даних сценаріїв, представлену на рис. 2.1.

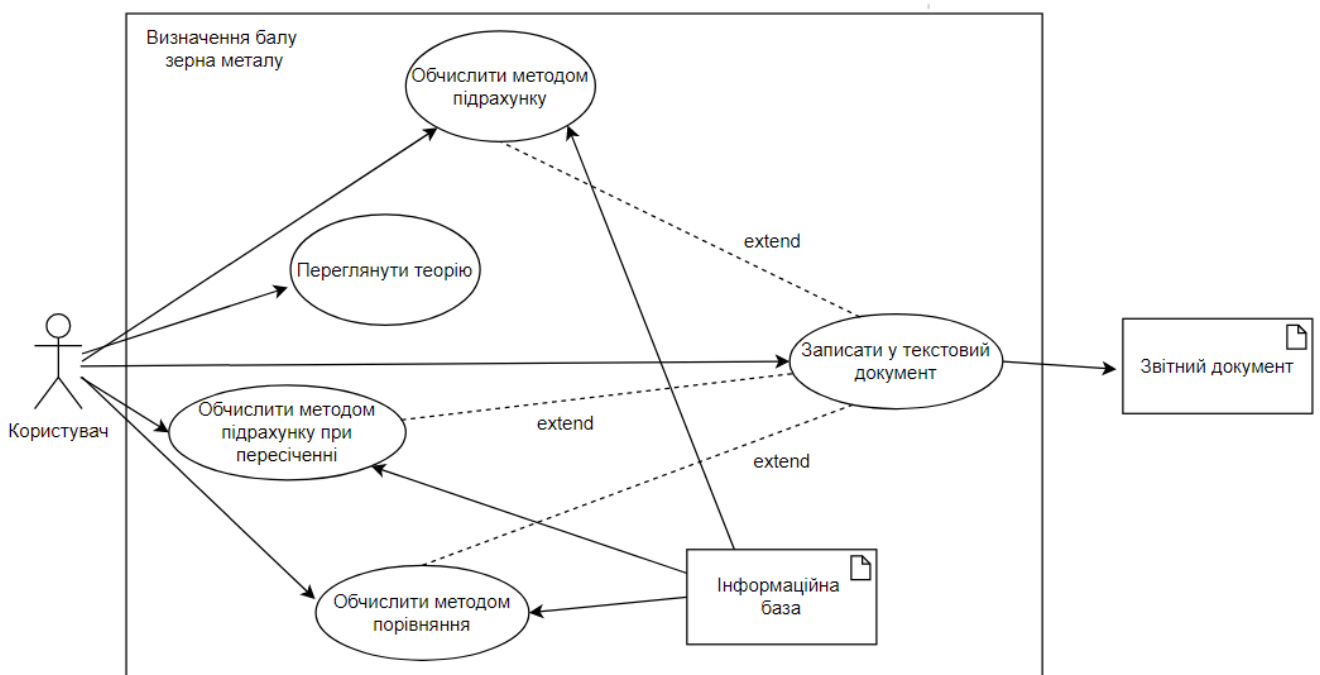


Рисунок 2.1 – Діаграма Use Case

2.2 Проектування моделі інформаційної бази

Для будь-якого продукту, який має дані для зберігання, важливо спроектувати модель бази даних. Даний додаток має такі дані для зберігання, як шляхи до файлів мікроструктур, що зберігатимуться на локальній машині користувача. Тому було вирішено замість звичайної бази даних використовувати інформаційну базу у вигляді XML файлу, який зберігатиметься локально та надаватиме додатку шляхи до фотографій мікроструктур.

Аналіз предметної області інформаційної бази:

- об'єкт – фотографія мікроструктури
- атрибути об'єкта: порядковий номер мікроструктури, збільшення мікроструктури, шлях до файлу зображення.

На рисунку 2.7 представлена ER-діаграма інформаційної бази.

Microstructure_photo	
PK	<u>photo_id</u>
	microstructure_id
	increase
	file_path

Рисунок 2.7 – Структура інформації інформаційної бази

Приклад опису наведено на рис. 2.8.

```
MicrostructuresInfoBase.xml  X
1  <?xml version="1.0" encoding="utf-8" ?>
2  <root>
3
4  <Microstructure>
5      <Name>_1</Name>
6      <Type>GOST_WITHOUT_POINT</Type>
7      <Increase>100</Increase>
8  </Microstructure>
9
10 <Microstructure>
11     <Name>_2</Name>
12     <Type>GOST_WITHOUT_POINT</Type>
13     <Increase>100</Increase>
14 </Microstructure>
15
16 <Microstructure>
17     <Name>_3</Name>
18     <Type>GOST_WITHOUT_POINT</Type>
19     <Increase>100</Increase>
20 </Microstructure>
21
22 <Microstructure>
23     <Name>_4</Name>
24     <Type>GOST_WITHOUT_POINT</Type>
25     <Increase>100</Increase>
26 </Microstructure>
27
28 <Microstructure>
29     <Name>_5</Name>
30     <Type>GOST_WITHOUT_POINT</Type>
31     <Increase>100</Increase>
32 </Microstructure>
33
```

Рисунок 2.8 – Приклад опису об'єкту структури

3 РОЗРОБКА ПРОГРАМНОГО ПРОДУКТУ

3.1 Програмна реалізація

Практична реалізація роботи розділена на 3 основні етапи: реалізація методу підрахунку, реалізація методу підрахунку при пересіченні та реалізація методу порівняння. Кожен метод реалізований як окремий UserControl.

3.1.1 Реалізація методу підрахунку

Під час виконання завдання методом підрахунку користувач робить позначення у вигляді символів плюс на фотографії мікроструктури для позначення зерен. Дана функція була реалізована за допомогою бібліотеки Drawing та методу `public void OnPictureBoxClicked(object sender, MouseEventArgs e, Func<FirstMethodDrawing, bool> pointSaver)`, який малював позначення у координатах натискання користувача на PictureBox.

Можливість змінювати розмір та колір позначення для більш кращого їх відображення на зображенні була реалізована за допомогою елемента ComboBox у вигляді випадаючого списку та відповідного обробника подій (наприклад `public void grainInsideColorComboBox_SelectedIndexChanged(object sender, EventArgs e)`), який визивався після кожної зміни обраного елемента списку та змінював значення кольору тексту, що відображається на зображенні структури. Також була реалізована можливість очищення останніх позначень у разі помилки. Значення координат кожного позначення на структурі зберігалися у масив, а при натисканні кнопки «Очистити останнє позначення» останній елемент масиву видалявся та усі інші позначення перемальовувалися. Програма самостійно підраховує кількість зроблених позначень кожного типу для подальшого використання цих значень у підрахунку балу зерна. Реалізація першого методу наведена на рисунку 3.1.

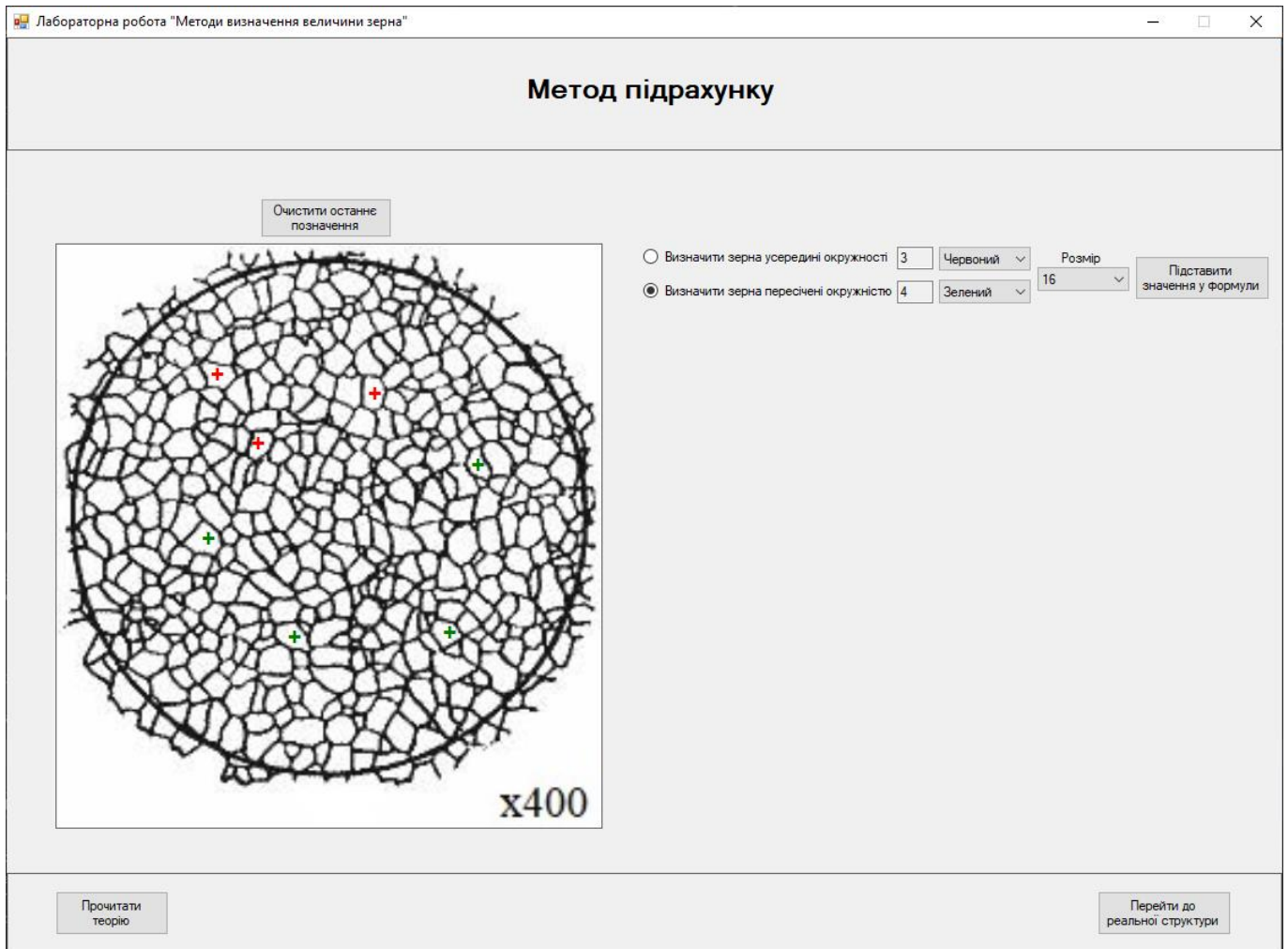


Рисунок 3.1 – Реалізація першого методу

Для підстановки значень у формули та отримання балу зерна було створено окремий UserControl, який опрацьовує введені користувачем дані у відповідні елементи TextBox та обчислює за допомогою них бал зерна металу після натискання користувачем кнопки «Обчислити бал зерна» та виклику методу `private void countGrainPointButton_Click(object sender, EventArgs e)`, який перевіряв чи усі потрібні значення встановлені у елементи TextBox.

Реалізація першого методу та заповнення формул наведена на рисунку 3.2.

Лабораторна робота "Методи визначення величини зерна"

Лабораторна робота

Очистити останнє позначення

Визначити зерна усередині окружності Червоний Розмір Підставити значення у формули

Визначити зерна пересічені окружністю Зелений

$m_{100} = m + 0,5 m_1$

m = Обчислити

m1 =

$S_{cp} = \frac{1}{M_{cp}}$

Mcp = Обчислити

$M = 2m_{100}$

m100 = Обчислити

$d_{cp} = \frac{1}{\sqrt{M_{cp}}}$

Mcp = Обчислити

Обчислити бал зерна

Прочитати теорію

Почати

Рисунок 3.2 – Реалізація першого методу та заповнення формул

Було реалізовано перехід від роботи зі структурою з ГОСТу до роботи з реальною мікроструктурою. У PictureBox встановлювалося нове зображення, а усі позначення очищалися за допомогою методу `private void clearMicrostructureDrawingSurface()`. Реалізація першого методу для реальної структури наведена на рисунку 3.3.

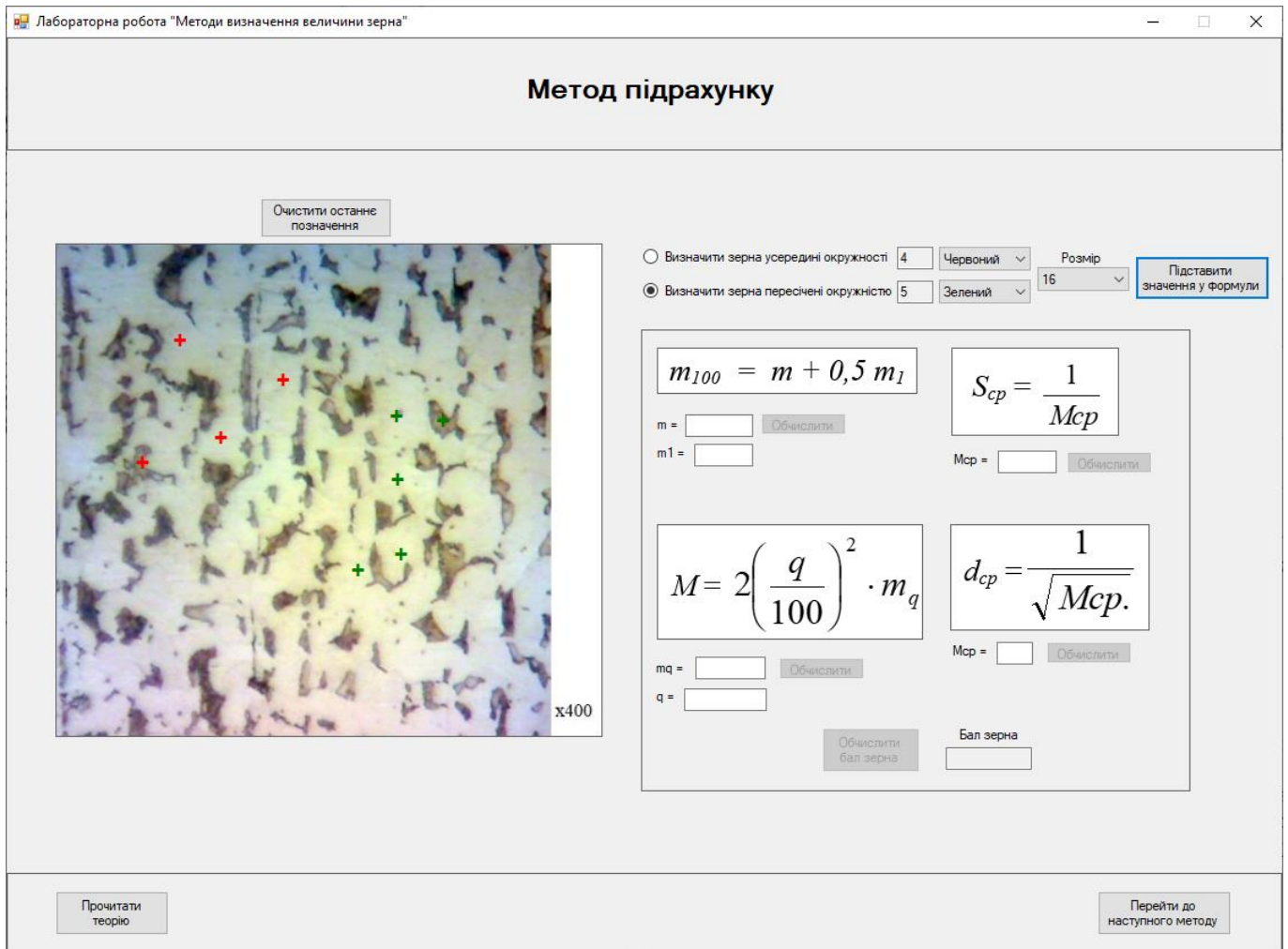


Рисунок 3.3 – Реалізація першого методу для реальної структури

3.1.2 Реалізація методу підрахунку при пересіченні

Під час виконання завдання методом підрахунку при пересіченні користувач робить позначення на фотографії мікроструктури у вигляді відрізків. Дана функція була реалізована за допомогою бібліотеки Drawing та методу `public void OnPictureBoxClicked(object sender, MouseEventArgs e, Func<Line, bool> lineSaver)`, який малює відрізок за координатами натискання користувача на PictureBox. Була реалізована можливість очищення останніх позначень у разі помилки. Програма самостійно підраховує загальну довжину відрізків, що потрібно для подальшого підрахунку балу зерна, та виводить обраховане значення у TextBox після малювання кожного нового відрізка. Реалізація другого методу наведена на рисунку 3.4.

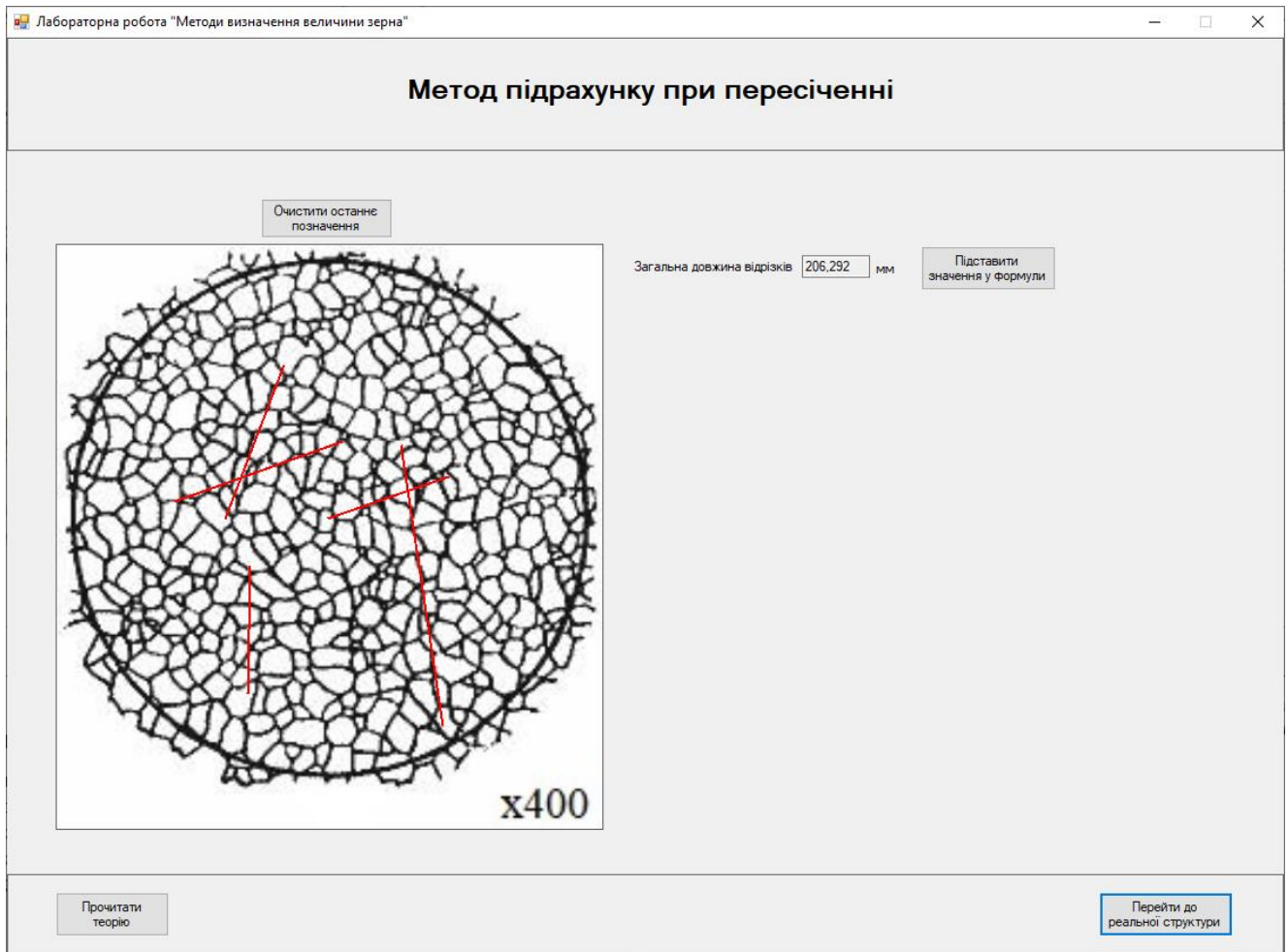


Рисунок 3.4 – Реалізація другого методу

Реалізація переходу до зображення реальної мікроструктури та виконання обчислень реалізовані аналогічно до попереднього варіанту виконання роботи.

Реалізація другого методу та заповнення формул наведена на рисунку 3.5.

The screenshot shows a software window titled "Лабораторна робота 'Методи визначення величини зерна'" (Laboratory work 'Methods of grain size determination'). The main title is "Лабораторна робота".

On the left side, there is a large empty rectangular area and a button labeled "Очистити останнє позначення" (Clear last marking).

At the top right, there is a label "Загальна довжина відрізків" (Total length of segments) with a text input field containing "0" and the unit "мм" (mm), and a button "Підставити значення у формули" (Insert values into formulas).

In the center, a box contains the formula
$$d_l = \frac{\sum l}{\sum n}$$

Below the formula, there are two input fields: $\sum l =$ [] and $\sum n =$ []. To the right of the $\sum l$ field is a button "Обчислити" (Calculate).

Below these fields, there is a button "Обчислити бал зерна" (Calculate grain score) and a text input field labeled "Бал зерна" (Grain score).

At the bottom left, there is a button "Прочитати теорію" (Read theory). At the bottom right, there is a button "Почати" (Start).

Рисунок 3.5 – Реалізація другого методу та заповнення формул

Було реалізовано перехід від роботи зі структурою з ГОСТу до роботи з реальною мікроструктурою. Реалізація другого методу для реальної структури наведена на рисунку 3.6.

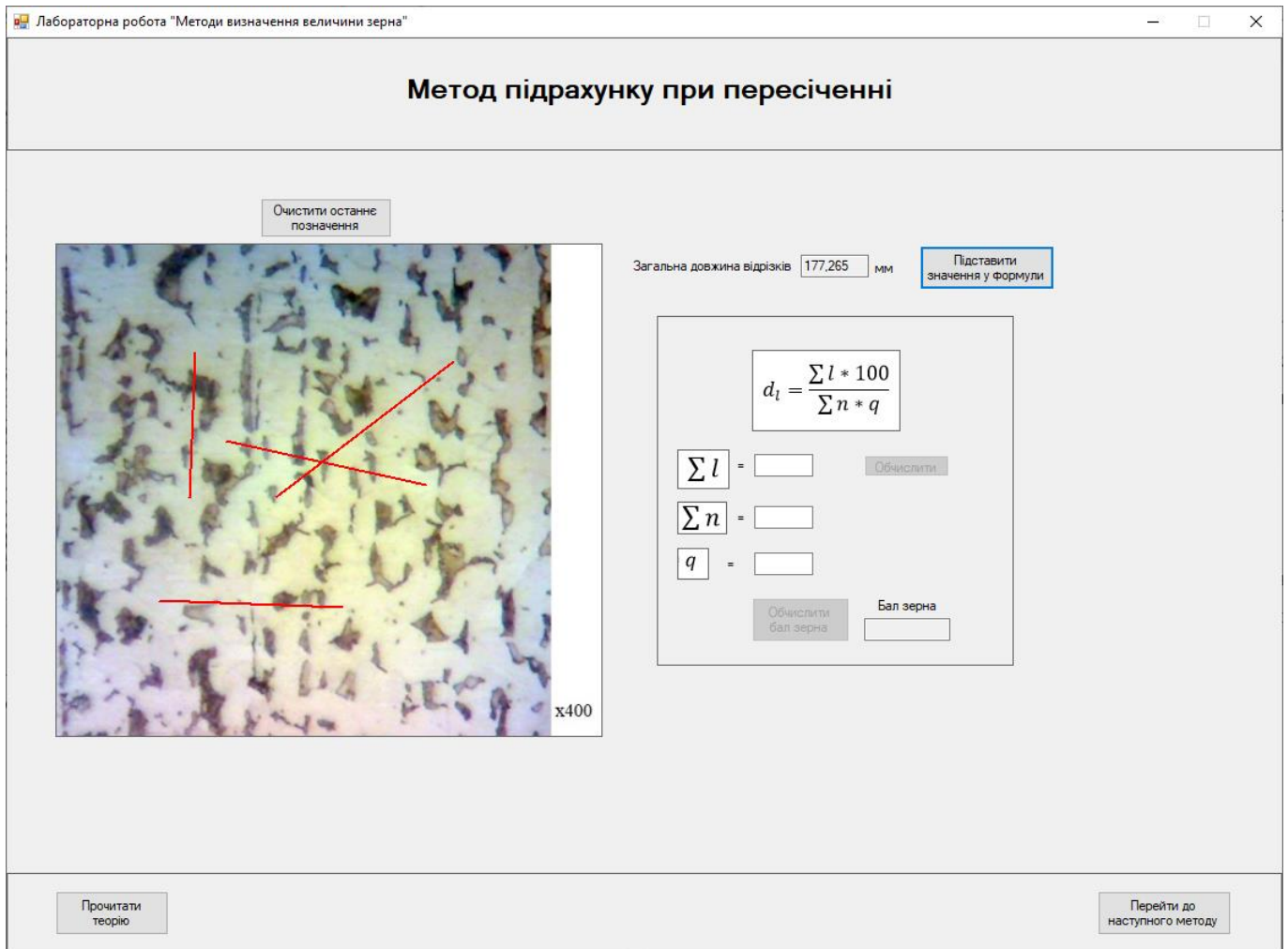


Рисунок 3.6 – Реалізація другого методу для реальної структури

3.1.3 Реалізація методу порівняння

Під час виконання завдання методом порівняння користувач порівнює мікроструктуру, з якою він працював, з мікроструктурами з ГОСТу. Реалізація даного методу полягає у відображенні мікроструктур у іншому PictureBox по черзі після натискання користувачем кнопок «Попередня» / «Наступна» та виклику методів `private void previousStructureButtonClick(object sender, EventArgs e)` та `private void nextStructureButtonClick(object sender, EventArgs e)`. У даному методі користувач самостійно визначає бал зерна вводючи його значення у відповідний TextBox. Реалізація третього методу наведена на рисунку 3.7, 3.8.

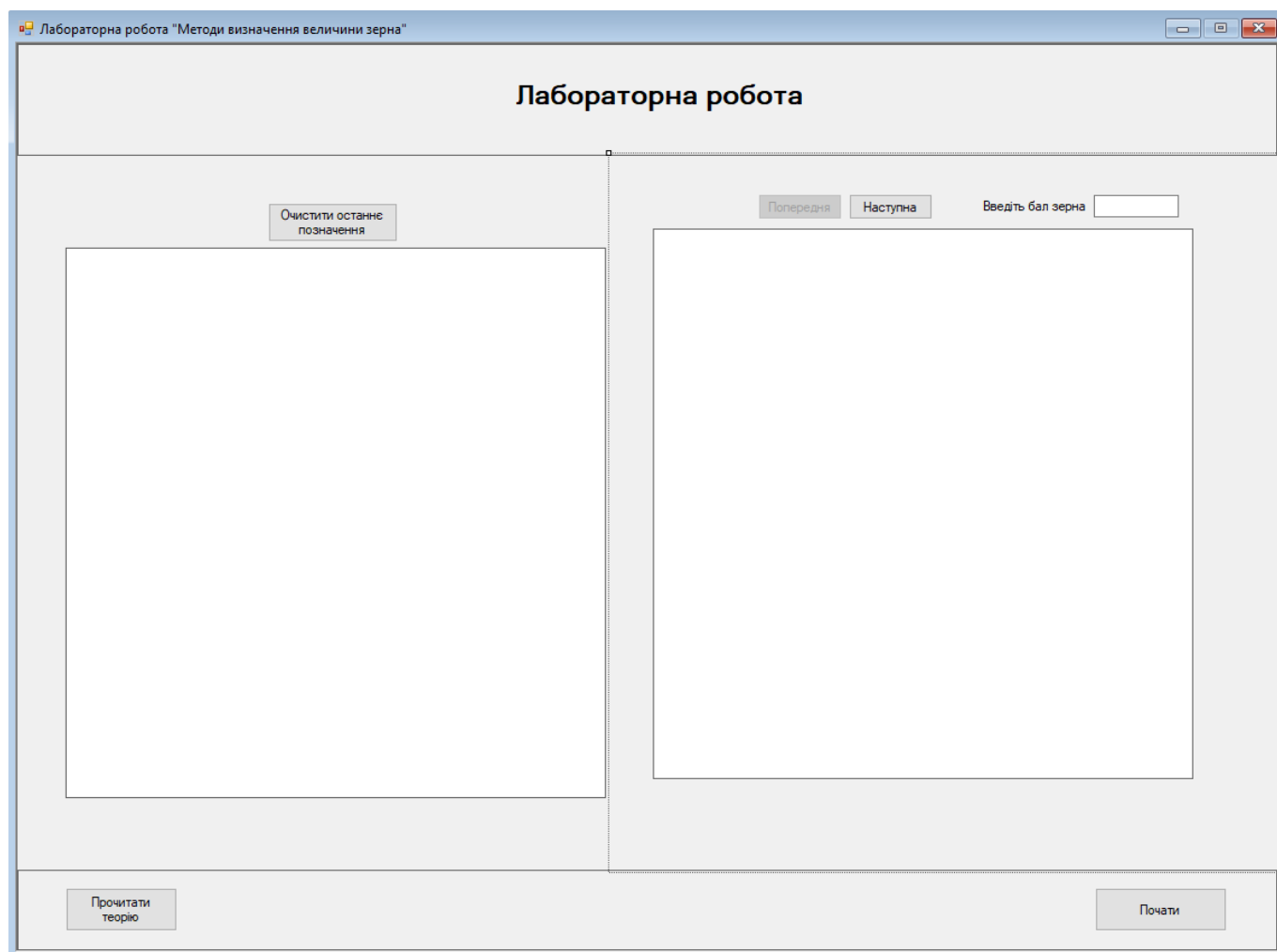


Рисунок 3.7 – Реалізація третього методу

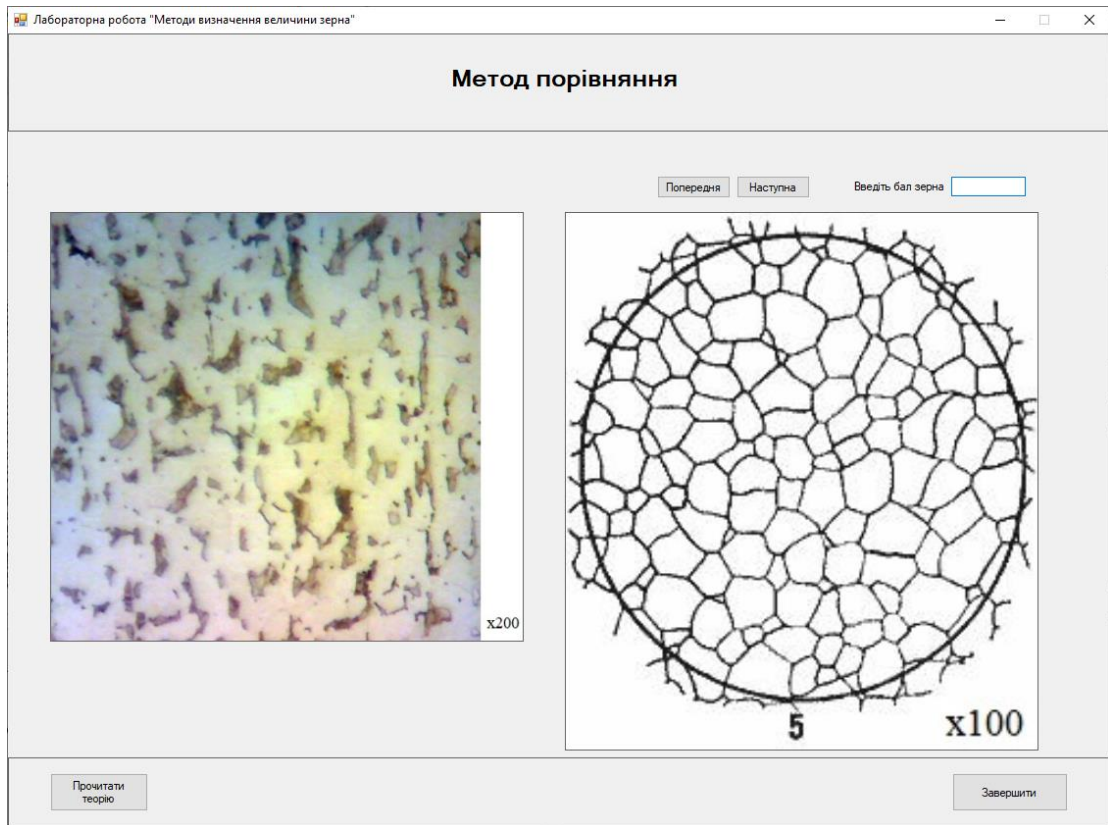


Рисунок 3.8 – Реалізація третього методу для реальної структури

Було реалізовано збереження усіх отриманих значень у документ Word шаблону звіту з лабораторної роботи, що відбувається після натискання на кнопку «Завершити». Запис у документ реалізовано за допомогою знаходження певного рядку, у вигляді індексу з символом «@» на початку, у шаблоні звіту документу Word, який замінюється на потрібне значення за допомогою функції Find.Execute. Для створення звіту було створено окремий клас ReportDocHelper, який зберігає значення усіх індексів для пошуку у шаблоні звіту у вигляді констант та усі значення для заміни, які зберігаються у масив після закінчення обчислення кожним із методів. У методі `public static bool createReport()` відбувається заміна значень у документі, потім викликається метод `private static bool saveDoc(Document doc)`, який викликає `SaveFileDialog`, який отримує значення шляху файлу звіту, обраного користувачем, і зберігає документ у створену додаткову папку за цим шляхом, у яку також зберігаються зображення мікроструктур. Реалізація виведення значень у текстовий документ наведена на рисунку 3.9.

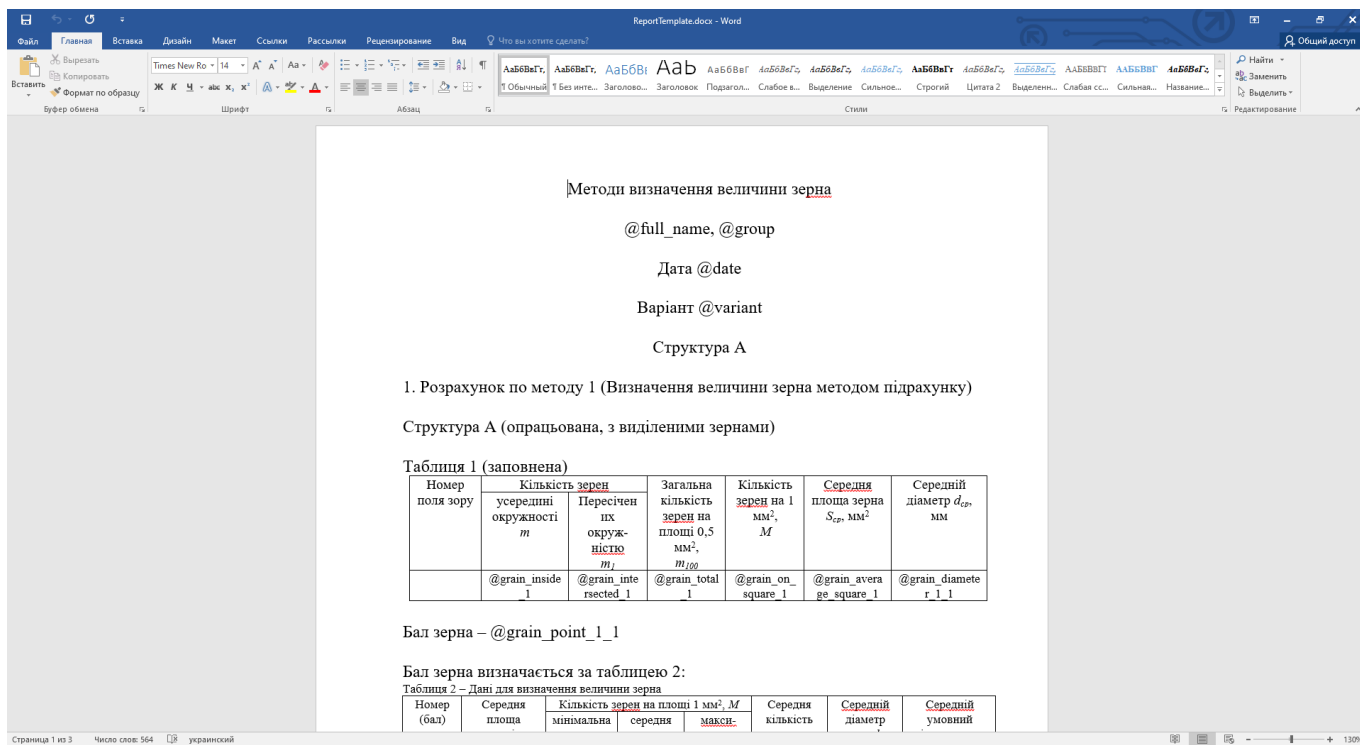
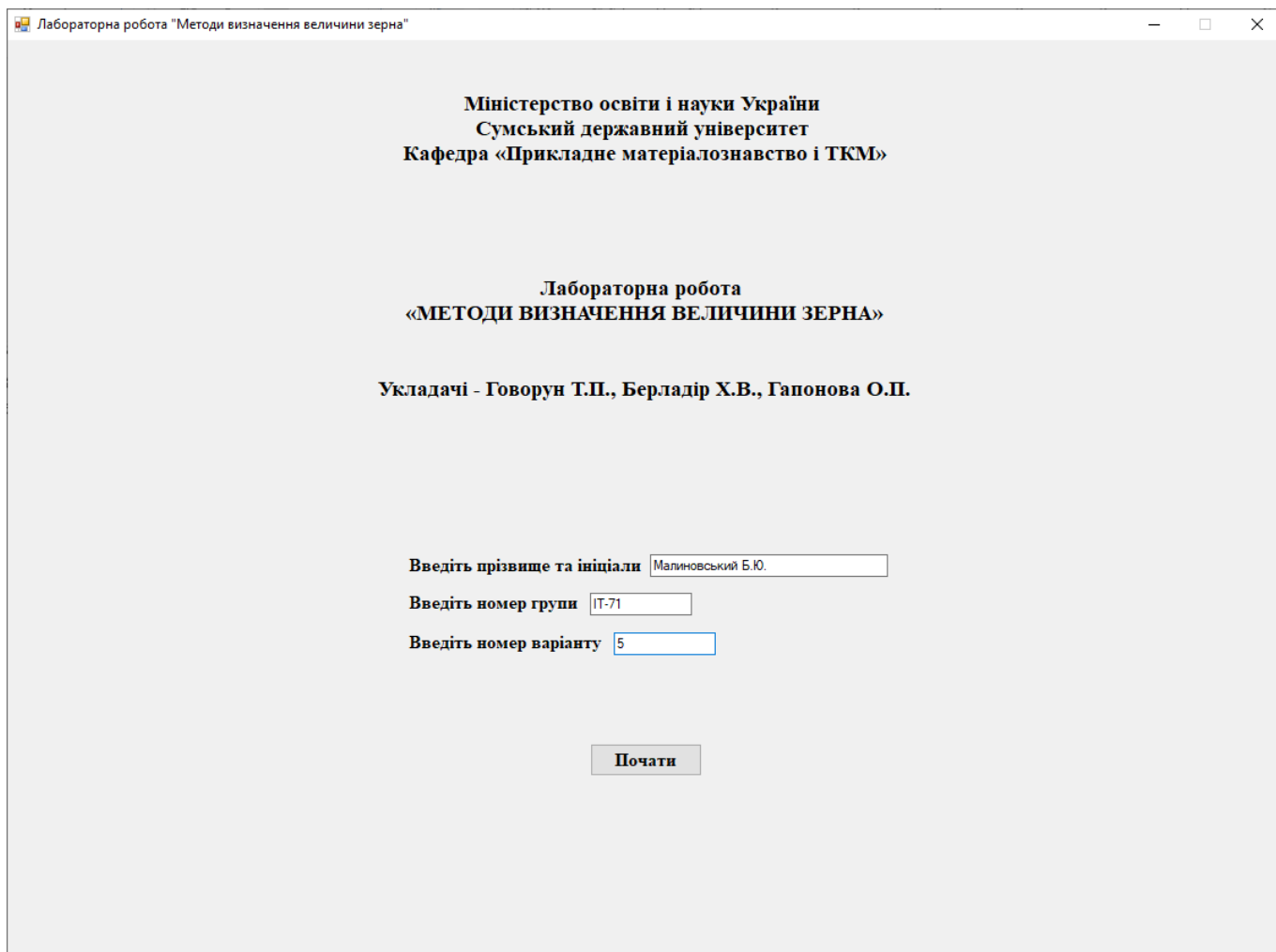


Рисунок 3.9 – Реалізація виведення значень у текстовий документ

Також було реалізовано початкову сторінку для виведення інформації щодо мети та теми лабораторної роботи, та отримання основної інформації про користувача. Кнопка «Почати» є неактивною до моменту поки усі три елементи TextBox не будуть заповнені. Реалізація початкової сторінки наведена на рисунку 3.10.



Лабораторна робота "Методи визначення величини зерна"

Міністерство освіти і науки України
Сумський державний університет
Кафедра «Прикладне матеріалознавство і ТКМ»

Лабораторна робота
«МЕТОДИ ВИЗНАЧЕННЯ ВЕЛИЧИНИ ЗЕРНА»

Укладачі - Говорун Т.П., Берладір Х.В., Гапонова О.П.

Введіть прізвище та ініціали

Введіть номер групи

Введіть номер варіанту

Рисунок 3.10 – Реалізація початкової сторінки

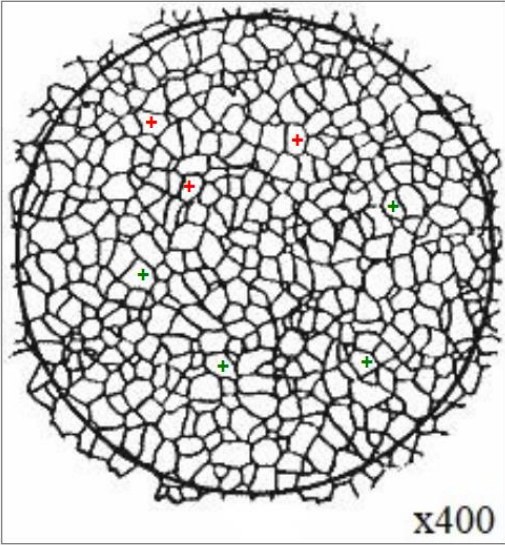
3.2 Використання програмного продукту

Після завершення реалізації програмного продукту було створено інсталятор для зручного завантаження програми на комп'ютер користувача. Приклад роботи програми наведено на рисунках 3.11-13.

Лабораторна робота "Методи визначення величини зерна"

Метод підрахунку

Очистити останнє позначення



x400

Визначити зерна усереднені окружності 3 Червоний Розмір Підставити значення у формули
 Визначити зерна пересічені окружністю 4 Зелений 16

$$m_{100} = m + 0,5 m_1$$

m = 5 Обчислити
 m1 = 3 m100 = 6,5

$$S_{cp} = \frac{1}{M_{cp}}$$

M_{cp} = 208 Обчислити
 S_{cp} = 0,005

$$M = 2 \left(\frac{q}{100} \right)^2 \cdot m_q$$

m_q = 6,5 Обчислити
 q = 400 M = 208

$$d_{cp} = \frac{1}{\sqrt{M_{cp}}}$$

M_{cp} = 0,005 Обчислити
 d_{cp} = 14,142

Обчислити бал зерна Бал зерна 0

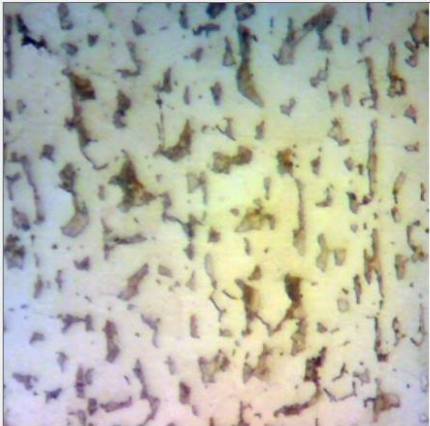
Прочитати теорію Перейти до реальної структури

Рисунок 3.11 – Приклад роботи першого методу

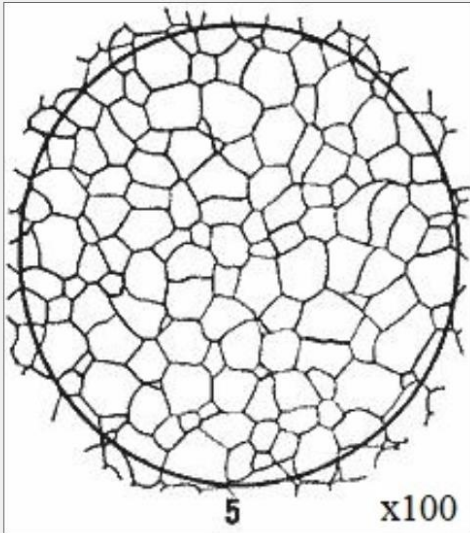
Лабораторна робота "Методи визначення величини зерна"

Метод порівняння

Попередня Наступна Введіть бал зерна



x200



5 x100

Прочитати теорію Закрити

Рисунок 3.12 – Приклад роботи третього методу

Методи визначення величини зерна
Малшовський Б.Ю., П-71
Дата 04.06.2021
Варіант 5
Структура А

1. Розрахунок по методу 1 (Визначення величини зерна методом підрозунок)
Структура А (опрацьована, з виділеними зернами)

Таблиця 1 (заповнена)

Номер поля зору	Кількість зерен у середній частині	Пересічені зерні	Загальна кількість зерен на площі 0,5 м ²	Кількість зерен на 1 м ² , М	Середній площі зерна S _{ср} , м ²	Середній діаметр d _{ср} , мм
4	5	6,5	14	0,071	3,753	

Бал зерна – 0

Бал зерна визначається за таблицею 2:

Таблиця 2 – Дані для визначення величини зерна

Номер (Бал) зерна	Середня площа зерну середньої частини S _{ср} , м ²	Кількість зерен на площі 1 м ² , М	Середня кількість зерен у 1 м ² , М _{ср}	Середній діаметр зерна d _{ср} , мм	Середній уловний діаметр зерна d _у , мм		
-3	1	0,75	1	1,5	1,0	0,875	
-2	0,5	1,5	2	3	2,8	0,707	0,650
-1	0,35	3	4	6	8	6,5	0,484
0	0,125	6	8	12	22,6	0,353	0,333
1	0,0625	12	16	24	64	0,250	0,229
2	0,0312	24	32	48	181	0,177	0,157
3	0,0156	48	64	96	512	0,125	0,111
4	0,00781	96	128	192	1448	0,088	0,0783
5	0,00390	192	256	384	4096	0,062	0,0551
6	0,00195	384	512	768	11584	0,044	0,0381

Структура А (з відрізком)

Таблиця 3 (заповнена)

Суцільна довжина відрізків у мм (на структурі на екрані)	Перерахована довжина відрізків у мм натуральної величини (з урахуванням збільшення)*, %	Число перетинів між зернами на кожному відрізку					Загальне число перетинів між зернами (Σn)	Середній умовний діаметр d _{ср} , мм	Номер зерна за шкалою G
		1	2	3	4	5			
100	100						140	0,714	-2

Висновок про середній бал зерна (одержані дані з таблиці 3 порівнюються з таблицею 2, про що і робиться висновок).

3. Метод 3 – Визначення величини зерна методом порівняння
Структура А
Структура з ГОСТа (зіставити їх або вказати відповідну шкалу)
Бал зерна – 5

Висновок про середній бал зерна.

Структура Б (аналогічно до структури А)

1. Розрахунок по методу 1 (Визначення величини зерна методом підрозунок)
Структура Б (опрацьована, з виділеними зернами)

Таблиця 1 (заповнена)

Номер поля зору	Кількість зерен у середній частині	Пересічені зерні	Загальна кількість зерен на площі 0,5 м ²	Кількість зерен на 1 м ² , М	Середній площі зерна S _{ср} , м ²	Середній діаметр d _{ср} , мм
4	5	6,5	40	0,025	6,332	

Рисунок 3.13 – Приклад створення звіту

Програмний продукт коректно працює після завантаження.

ВИСНОВКИ

У наш час зручніше навчатися використовуючи лабораторні роботи та матеріали, представлені в електронному вигляді. Це дозволяє зменшити час на виконання завдань та перевірки завдання викладачем. Не дивлячись на це студенти технічних спеціальностей, які вивчають аналіз структури металів, навіть на сьогоднішній день використовують стандартне фізичне обладнання для виконання потрібних дослідів. Це ж стосується і виконання такої роботи, як визначення величини зерна металу. Тому обрана тема кваліфікаційної роботи є актуальною.

У ході роботи над проектом було проведено аналіз предметної області та огляд існуючих альтернатив. Було створено перелік вимог до майбутнього програмного продукту.

За вимогами стандарту IDEF0 створено контекстну діаграму і діаграму першого рівня декомпозиції моделі, яка описує структурні етапи процесу виконання віртуальної лабораторної роботи та взаємозв'язки між ними. Можливі варіанти роботи з програмним додатком описано у вигляді створеної UML діаграми варіантів використання. Для створення бази допустимих фотознімків, що використовуються в ході виконання роботи, використовується xml-файл.

Програмна реалізація виконано мовою програмування C# за технологією .NET Framework. Створена програма повністю реалізує всі визначені у технічному завданні вимоги. Розроблене програмне забезпечення може замінити собою виконання реальної фізичної лабораторної роботи, а також спростить організацію навчального процесу за дистанційною формою навчання.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Диаграмма вариантов использования (use case diagram) [Электронный ресурс] – Режим доступу до ресурсу: https://sites.google.com/site/anisimovkhv/learning/pris/lecture/tema12/tema12_2
2. IDEF0 [Электронный ресурс] – Режим доступу до ресурсу: <https://sites.google.com/site/anisimovkhv/learning/pris/lecture/tema6#p62>
3. Розробка додатків за технологією Windows Forms [Электронный ресурс] – Режим доступу до ресурсу: <https://docs.microsoft.com/en-us/visualstudio/ide/create-csharp-winform-visual-studio?view=vs-2019>
4. Создание и чтение XML-файла C# [Электронный ресурс] - Режим доступу: https://vscode.ru/prog-lessons/sozдание_chtenie-xml-c-sharp.html
5. XML файлы C#: создание, загрузка, редактирование, удаление [Электронный ресурс] - Режим доступу: <https://vscode.ru/prog-lessons/sozдание-redaktirovanie-i-udalenie-dannyih-xml-fayla-c-sharp.html>
6. WPF Controls. Export to DOCX [Электронный ресурс] - Режим доступу: <https://docs.devexpress.com/WPF/118865/controls-and-libraries/printing-exporting/concepts/exporting/export-to-docx>
7. Грекул В. И. Методические основы управления ИТ-проектами / В. И. Грекул, Н. Л. Коровкина, Ю. В. Куприянов. – Москва: Интернет-Университет Информационных Технологий: БИНОМ. Лаборатория знаний., 2010. – 391 с.
8. Гультияев, А.К. MS Project 20010 Professional. Управление проектами: Самоучитель / А.К. Гультияев. – СПб.: Корона Принт, 2009. – 512 с.
9. Джалота, П. Управление проектами в области информационных технологий / П. Джалота. - М.: Лори, 2014. – 224 с.
10. Организационная структура предприятия: виды и схемы [Электронный ресурс] – режим доступу: <https://kontur.ru/articles/4197>

11. Организационная структура предприятия [Электронный ресурс] – режим доступа: <https://www.inventech.ru/lib/predpr/predpr0015/>
12. Структура декомпозиции работ или WBS иерархия проекта [Электронный ресурс] – режим доступа: <https://finswin.com/projects/instrumenty/wbs.html>
13. Что такое WBS проекта, и зачем она нужна [Электронный ресурс] – режим доступа: <http://upravlenie-proektami.ru/chto-takoe-wbs-proekta-i-zachem-ona-nuzhna>
14. RACI: распределение ролей и зон ответственности в проекте [Электронный ресурс] – режим доступа: http://smartsourcing.ru/blogs/upravlenie_proektami/633
15. Механические свойства, влияние размера зерна [Электронный ресурс] - Режим доступа: <https://mash-xxl.info/info/329429/>
16. Мікроаналіз металів і сплавів [Электронный ресурс] - Режим доступа: https://web.posibnyky.vntu.edu.ua/fmbt/shapovalova_materialoznavstvo/2.htm
17. Влияние величины зерна на свойства стали [Электронный ресурс] - Режим доступа: <https://studfile.net/preview/5433250/page:6/>
18. Макро- и микроанализ металлов и сплавов [Электронный ресурс] - Режим доступа: <http://lib.madi.ru/fel/fel1/fel18E474.pdf>
19. How to Adapt Grain Size Analysis of Metallic Alloys to Your Needs [Электронный ресурс] - Режим доступа: <https://www.leica-microsystems.com/science-lab/how-to-adapt-grain-size-analysis-of-metallic-alloys-to-your-needs/>
20. Grain Size Analysis in Metals and Alloys [Электронный ресурс] - Режим доступа: <https://www.olympus-ims.com/en/applications/grain-size-analysis/>
21. Анисович А. Г. Определение балла зерна стали компьютерными методами / А. Г. Анисович, И. Н. Румянцева, Л. В. Бислюк. // Литье и металлургия. – 2010. – №3. – С. 100.

22. Малиновський Б.Ю. Віртуальна лабораторна робота з визначення величини зерна металу. / Б.Ю. Малиновський, С.М. Ващенко // Інформатика, математика, автоматика: матеріали та програма науково-технічної конференції, м. Суми, 19 – 20 квітня 2021 р. – Суми: Сумський державний університет, 2021. – с.95.
23. AxioVision. Microscopy from Carl Zeiss [Електронний ресурс] - Режим доступу: https://mikroskop.com.pl/pdf/Zeiss_AxioVision_Brochure.pdf
24. Сільвейстр А.М. Інформаційно-комунікаційні технології навчання як засоби реалізації віртуальних лабораторних робіт з фізики у майбутніх учителів хімії і біології / А.М. Сільвейстр // Науковий журнал Вінницького державного педагогічного університету ім. М. Коцюбинського. – 2015. – 3(6). – с. 85-96.
25. Швидкий О.В. Використання інформаційних технологій у навчальному процесі та створення електронного підручника. / О.В. Швидкий, О.А. Хадикіна, В.С. //Медична освіта.- 2012. - №4. - с.144-146.

ДОДАТОК А

ТЕХНІЧНЕ ЗАВДАННЯ **на розробку інформаційної системи** **«Віртуальна лабораторна робота** **з визначення величини зерна металу»**

ПОГОДЖЕНО:

Доцент кафедри комп'ютерних наук

_____ Ващенко С.М.

Студент групи ІТ-71

_____ Малиновський Б.Ю.

Суми 2021

1. Призначення й мета створення додатку

Призначення додатку

Інформаційна система має надавати користувачу можливість отримати практичні навички проведення розрахунків по визначенню величину зерна металу.

1.2 Мета створення додатку

Спрощення процесу проведення лабораторних робіт в умовах дистанційного навчання.

1.3 Цільова аудиторія

До цільової аудиторії додатку можна студентів та викладачів, які проводять дослідження металу.

2 Вимоги до додатку

2.1 Вимоги до додатку в цілому

2.1.1 Системні вимоги

Операційна система пристрою користувача повинна бути однієї з версій Windows та мати .Net Framework не нижче 4.0 версії.

2.1.2 Вимоги до структури й функціонування додатку

Продукт повинен бути реалізований у вигляді одновіконного додатку Windows Forms.

2.1.3 Вимоги до функціональній можливостей

Додаток повинен надавати користувачу фотографію мікроструктури, з якою він буде працювати протягом усієї лабораторної роботи, отримувати дані з мікроструктури за методами підрахунку, підрахунку при пересіченні та порівняння,

обчислювати бал зерна металу за відповідними методами, виводити результат у документ MS Office Word.

2.2 Структура додатку

2.2.1 Загальна інформація про структуру додатку

Структура додатку являє собою набір сторінок, на які користувач переходить послідовно:

Початкова сторінка – призначена надання користувачу основної інформації про лабораторну роботу, такої як мета, завдання.

Сторінка 1 методу – визначення величини зерна методом підрахунку.

Сторінка 2 методу – визначення величини зерна методом підрахунку при пересіченні зерен.

Сторінка 3 методу – визначення величини зерна методом порівняння.

2.2.2 Метод підрахунку

Студент послідовно підраховує:

- кількість зерен, що повністю потрапили всередину кола (m).
- кількість зерен, що частково потрапили всередину кола, тобто пересічених нею, (параметр m_1).

Ці дані використовуються для отримання таких величин:

Ці дані використовуються для отримання таких величин:

Загальна кількість зерен на площі $0,5 \text{ мм}^2$ (m_{100})

$$m_{100} = m + 0,5 * m_1 \quad (2.1)$$

Кількість зерен на 1 мм^2 (M):

$$M = 2 * m_{100} \quad (2.2)$$

Середня площа зерна S , мм^2 :

$$S = 1/M \quad (2.3)$$

Середній діаметр d , мм:

$$d = 1/\sqrt{M} \quad (2.4)$$

Отримані результати заносяться у таблицю, структура якої відповідає табл. 2.1 (позначкою «*» відмічено дані, які заносяться в ході виконання лабораторної роботи).

Таблиця 2.1 – Таблиця основних даних методу 1

Номер поля зору	Кількість зерен		Загальна кількість зерен на площі 0,5 мм ² , m_{100}	Кількість зерен на 1 мм ² , M	Середня площа зерна S_{cp} , мм ²	Середній діаметр d_{cp} , мм
	усередині окружності m	пересічених окружностей m_l				
*	*	*	*	*	*	*

Далі студент має порівняти одержані дані з таблиці 2.1 з таблицею 2.2, та зробити висновок про середній бал зерна.

Бал зерна визначається за таблицею 2:

Таблиця 2.2 – Дані для визначення величини зерна

Номер (бал) зерна	Середня площа перерізу зерна, S_{cp} , мм ²	Кількість зерен на площі 1 мм ² , M			Середня кількість зерен у 1 мм ² , M_{cp}	Середній діаметр зерна d_{cp} , мм	Середній умовний діаметр зерна d_l , мм
		мінімальна	середня	максимальна			
-3	1	0,75	1	1,5	1	1,0	0,875
-2	0,5	1,5	2	3	2,8	0,707	0,650
-1	0,25	3	4	6	8	0,5	0,444
0	0,125	6	8	12	22,6	0,353	0,333
1	0,0625	12	16	24	64	0,250	0,222

*	*	*	*	*	*	*	*	*	*
---	---	---	---	---	---	---	---	---	---

Далі студент має порівняти одержані дані з таблиці 2.3 з таблицею 2.2, та зробити висновок про середній бал зерна.

За результатами виконання роботи формується звіт у форматі документа MS Word.

2.2.4 Метод порівняння

За допомогою цього методу бал зерна визначають під мікроскопом при збільшенні 100x шляхом перегляду всієї площі шліфа й порівняння видимих на шліфі зерен з еталонними зображеннями, наведеними відповідно до ГОСТ 5639-82.

2.2.5 Дизайн та структура додатку

Стиль додатку має бути простим, стандартним для програм створених за допомогою Windows Forms.

Додаток має бути інтуїтивно зрозумілим для використання.

Розташування елементів додатку схематично показано на рисунках

2.1-2.6

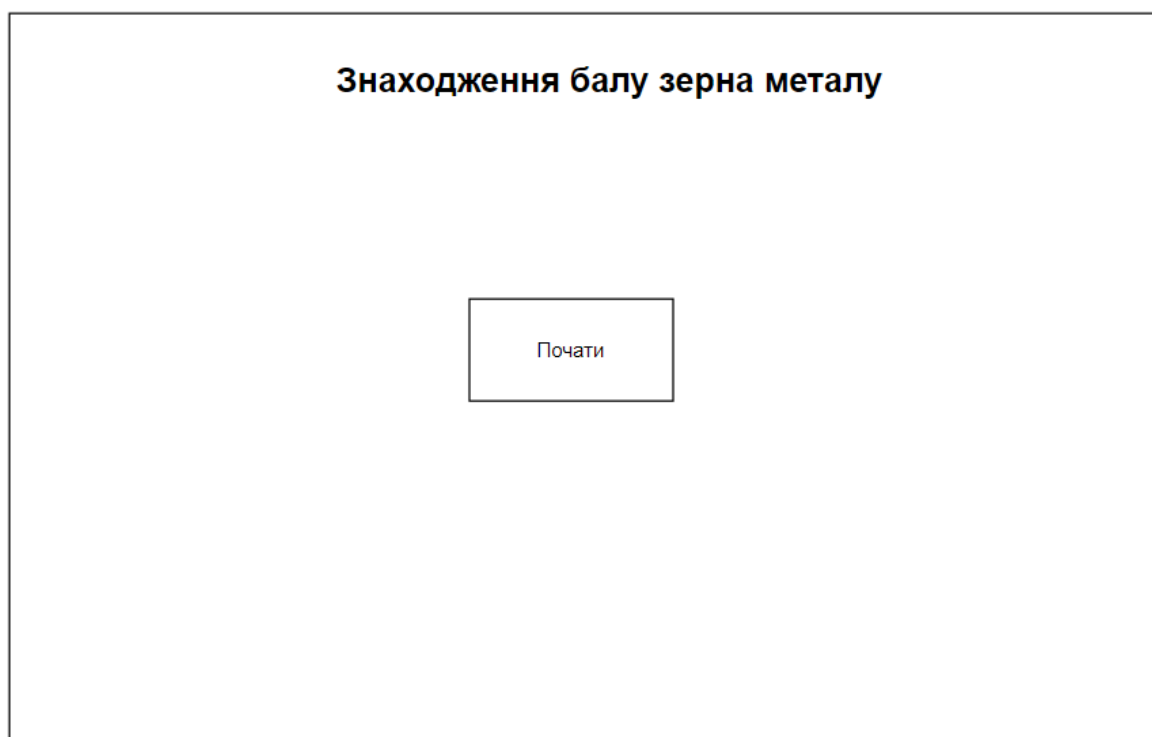


Рисунок 2.1 – Схема початкової сторінки

Метод 1

Фотографія
мікроструктури

- Визначити зерна всередині окружності
- Визначити зерна пересічені окружністю

Підставити значення у формули

Бал зерна

Вивести результати у документ

Прочитати теорію

Перейти до наступного методу

Рисунок 2.2 – Схема сторінки першого методу

Метод 2

Фотографія
мікроструктури

Сумарна довжина відрізків

Загальне число перетинів меж зерен

Підставити значення у формули

Бал зерна

Вивести результати у документ

Прочитати теорію

Перейти до наступного методу

Рисунок 2.3 – Схема сторінки другого методу

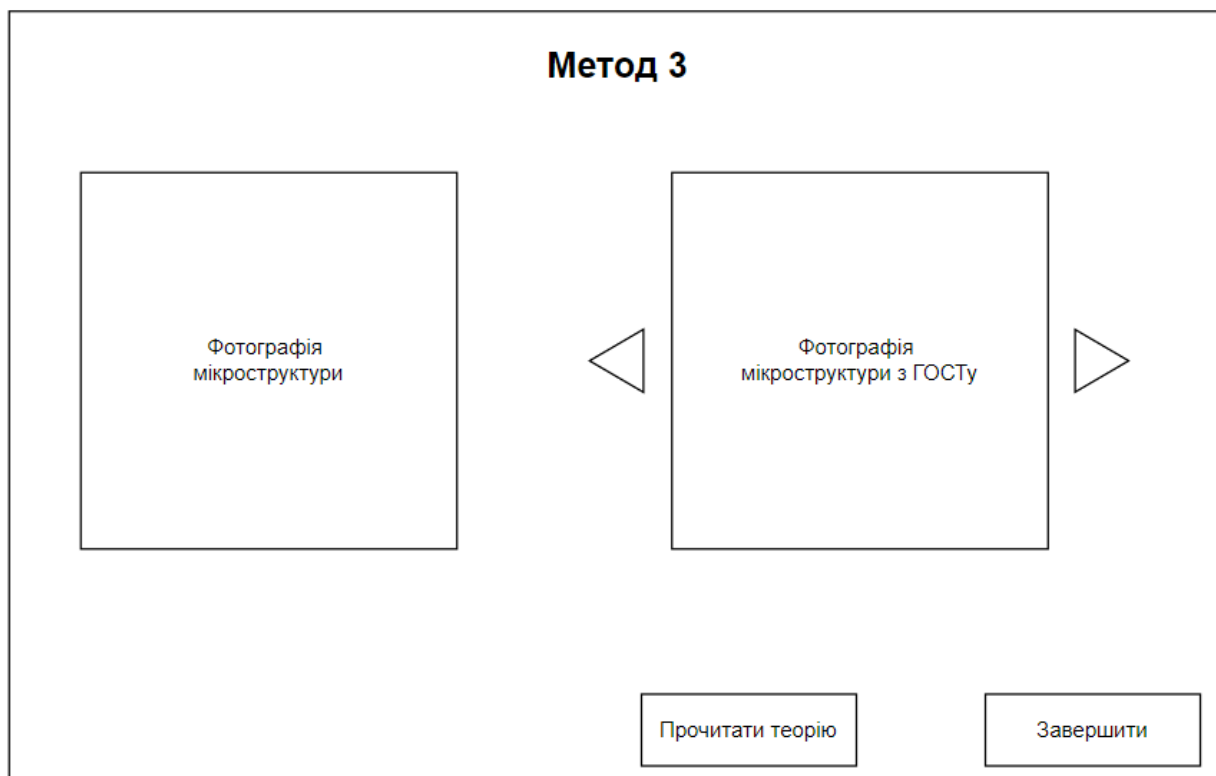
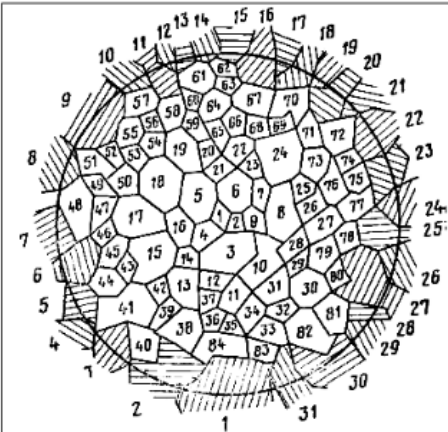


Рисунок 2.4 – Схема сторінки третього методу

Теорія



При збільшенні $\times 100$:

$$m_{100} = m + 0,5 m_1$$

Кількість зерен, що припадає на 1 мм^2 площі шліфа M , знаходять за формулою

$$M = 2m_{100}$$

При інших збільшеннях, відмінних від $100\times$, M буде дорівнювати

$$M = 2 \left(\frac{q}{100} \right)^2 \cdot m_q$$

де q - застосоване збільшення, m_q - кількість зерен при збільшенні q .

S_{cp} - середнє значення площі зерна, що знаходиться за такою формулою:

$$S_{cp} = \frac{1}{M_{cp}}$$

Діаметр зерна (d_{cp}) обчислюють за формулою

$$d_{cp} = \frac{1}{\sqrt{M_{cp}}}$$

де M_{cp} - середня арифметична кількість зерен, яку знаходять за оцінками не менше трьох найбільш характерних для даного шліфа полів зору (згідно ГОСТ 5639-82).

В лабораторній роботі розрахунок проводимо для одного поля зору, тому приймаємо

$$M_{cp} = M$$

Відповідно $d_{cp} = d$.

Рисунок 2.5 – Схема сторінки з теоретичним матеріалом

$$m_{100} = \boxed{} + 0,5 * \boxed{}$$

$$M = 2 * \boxed{}$$

Обрахувати бал
зерна

Рисунок 2.6 – Схема вікна для введення елементів виразів

2.3 Вимоги до видів забезпечення

2.3.1 Вимоги до програмного забезпечення

Для реалізації додатку необхідно обрати мову програмування придатну для розробки за технологією Windows Forms. Мова програмування С# є вибором більшості розробників, працюючих з додатками даного типу.

Серед середовищ розробки було обрано Visual Studio, що вважається одним з найкращих серед конкуруючих аналогів, що використовуються для розробки додатків за даною технологією.

2.3.2 Вимоги до лінгвістичного забезпечення

Додаток має бути виконаний українською мовою.

3 Склад і зміст робіт зі створення додатку

Докладний опис етапів роботи зі створення додатку наведено в табл. 3.1

Таблиця А.3 – Етапи створення додатку

№	Склад і зміст робіт	Строк розробки (у робочих днях)
1	Постановка цілей необхідних для досягнення певного результату	1 день
2	Складання технічного завдання	3 дні
3	Підготовка прототипу	2 дні
4	Створення макету дизайну додатку	3 дні
5	Реалізація логіки додатку	5 дні
6	Перевірка працездатності додатку	1 день
7	Завершення роботи	1 день
	Загальна тривалість робіт	16 днів

4 Вимоги до складу й змісту робіт із введення додатку в експлуатацію

Для того, щоб додатком міг користуватися замовник та уся цільова аудиторія потрібно створити інсталятор, за допомогою якого користувач зможе встановити програму на ПК. Для коректного користування додатком необхідно, щоб параметри програмного забезпечення користувача відповідали вимогам, зазначеним у ТЗ.

ДОДАТОК Б

Планування робіт

Деталізація мети проекту методом SMART. Продуктом дипломного проекту є програма для визначення величини зерна металу, призначена для кафедр з дослідження металу у вищих навчальних закладах.

Результати деталізації методом SMART розміщені у табл. Б.1.

Таблиця Б.1 – Деталізація мети методом SMART

Specific (конкретна)	Створити програму для визначення величини зерна металу.
Measurable (вимірювана)	Результатом роботи проекту є оцінка замовника.
Achievable (досяжна)	Реалізація додатку здійснюється за допомогою середовища розробки Visual Studio за технологією Windows Forms з використанням мови програмування C#.
Relevant (реалістична)	У наявності є всі необхідні програмні засоби: Visual Studio. Розробник достатньо кваліфікований для виконання поставлених задач.
Time-framed (обмежена у часі)	Ціль має часове обмеження. Робота повинна бути виконана у терміни, що були оговорені замовником проекту. Проект повинен бути виконаний згідно з календарним планом.

Планування змісту структури робіт. Основним інструментом для планування змісту структури робіт служить WBS(Work Break Structure) - представлення проекту, виконане у вигляді ієрархічної структури робіт, що досягається за допомогою послідовної декомпозиції. Інструмент спрямований на детальне планування, оцінку вартості, визначення та розподіл персональної

відповідальності виконавців та інші - тобто, на основні роботи і результати, що визначають зміст проекту.

Як правило, на верхньому рівні вказується сам проект, під ним (на першому рівні) - основні результати, кожен з яких, в свою чергу, деталізується, тобто наступний рівень завжди менше попереднього за обсягом робіт і, як правило, включає 2 і більше пакетів робіт. При цьому в різних гілках WBS може бути різна кількість рівнів в залежності від потрібного ступеня деталізації.

В ході виконання даного проекту перший рівень структури називається «Віртуальна лабораторна робота з визначення величини зерна», який в свою чергу декомпозиується на чотири рівні: ініціалізація, планування, реалізація та завершення.

Ініціалізація додатку складається з таких етапів: ознайомлення з предметною областю, визначення в потребі додатку та ідентифікація ідеї проекту.

Наступний рівень – планування, який розбивається на два рівні:

1. аналіз документації:
2. визначення вимог:
 - визначення інструментарію;
 - планування WBS;
 - планування OBS;
 - складання календарного плану;
 - визначення бюджету;
 - визначення ризиків;

Третій рівень діаграми – реалізація, який розбивається на чотири рівні:

1. моделювання роботи додатку;
 2. розробка макету додатку;
 3. розробка функціоналу додатку;
- тестування:
- тестування розробником;
 - тестування незалежною особою;

Останній етап створення проекту є завершення, що містить в собі такі процеси: оформлення документації, архівація та експлуатація продукту.

На рисунку Б.2 приведена WBS-структура даного проекту.

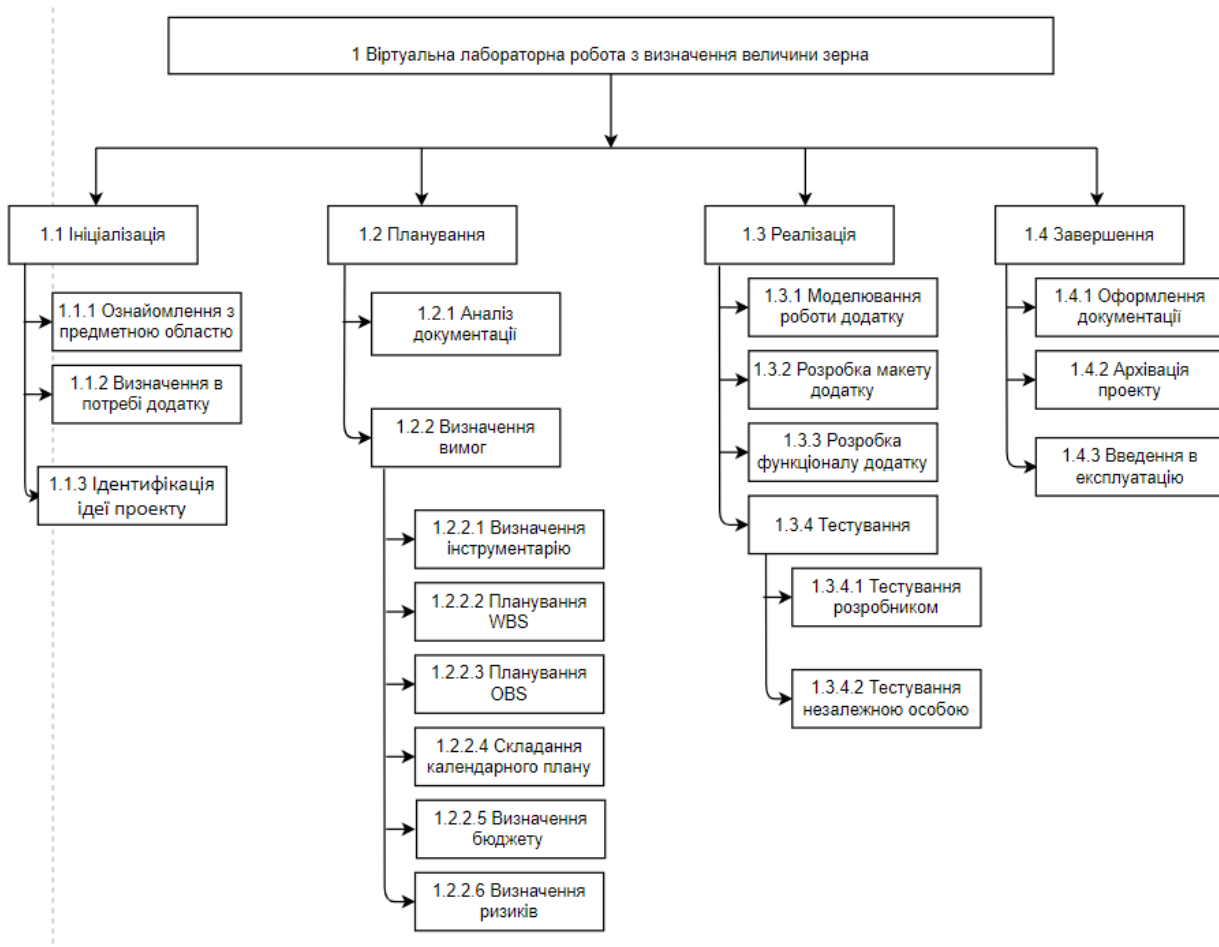


Рисунок Б.2 – WBS-структура проекту

Планування структури організації, для впровадження готового проекту (OBS). Після того, як була побудована WBS структура проекту наступним етапом є розроблення OBS (Organization Break structure) - склад, підпорядкованість, взаємодія і розподіл робіт по підрозділах і органам управління, між якими встановлюються певні відносини з приводу реалізації владних повноважень, потоків команд і інформації. Організаційна структура проекту стосується тільки внутрішньої організаційної структури проекту і не стосується відносин проектних груп чи учасників з батьківськими організаціями. Список виконавців, що функціонують в проекті представлений в таблиці Б.2. Організаційна структура проекту зображена на рисунку Б.3.

Таблиця Б.2 – Виконавці проекту

Роль	Ім'я	Проектна роль
Розробник	Малиновський Б.Ю.	Виконує розробку основного функціоналу проекту.
Менеджер проекту	Ващенко С.М.	Відповідає за виконання термінів, виконує збір та аналіз даних.
Тестувальник	Замовник	Відповідає за тестування функціоналу проекту

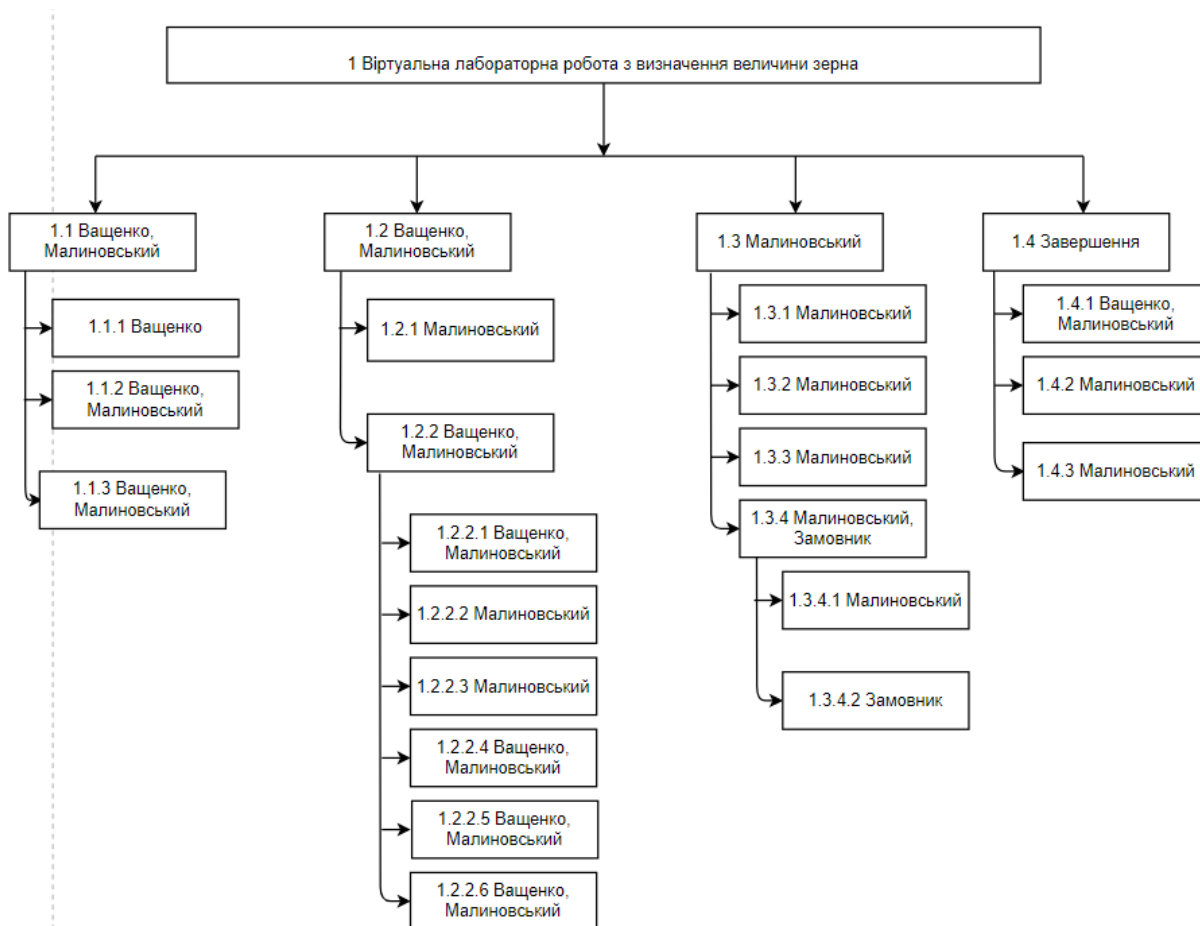


Рисунок Б.3 – OBS – структура проекту

Діаграма Ганта. Далі побудуємо календарний план виконання дипломного проекту. Найпоширеніший формат графіка в будь-якій галузі — діаграма Ганта. Управління проектами з діаграмами Ганта засноване на форматі гістограм. Це допомагає відслідковувати відсоток робіт, виконаних по кожному завданню. Керівникам проектів дуже важливо правильно розподілити завдання і бути впевненими в тому, що проект буде завершений вчасно. Основна увага діаграм Ганта зосереджено на процентному завершенні кожного завдання. Крім того, діаграми Ганта краще для проектів з невеликою кількістю взаємопов'язаних завдань. Завдяки засобам програмного продукту MS Project була розроблена діаграма Ганта, яка у вигляді гістограми відображає тривалість кожного процесу, що був визначений на етапі формування WBS. Діаграма Ганта представлена на рисунку Б.4.

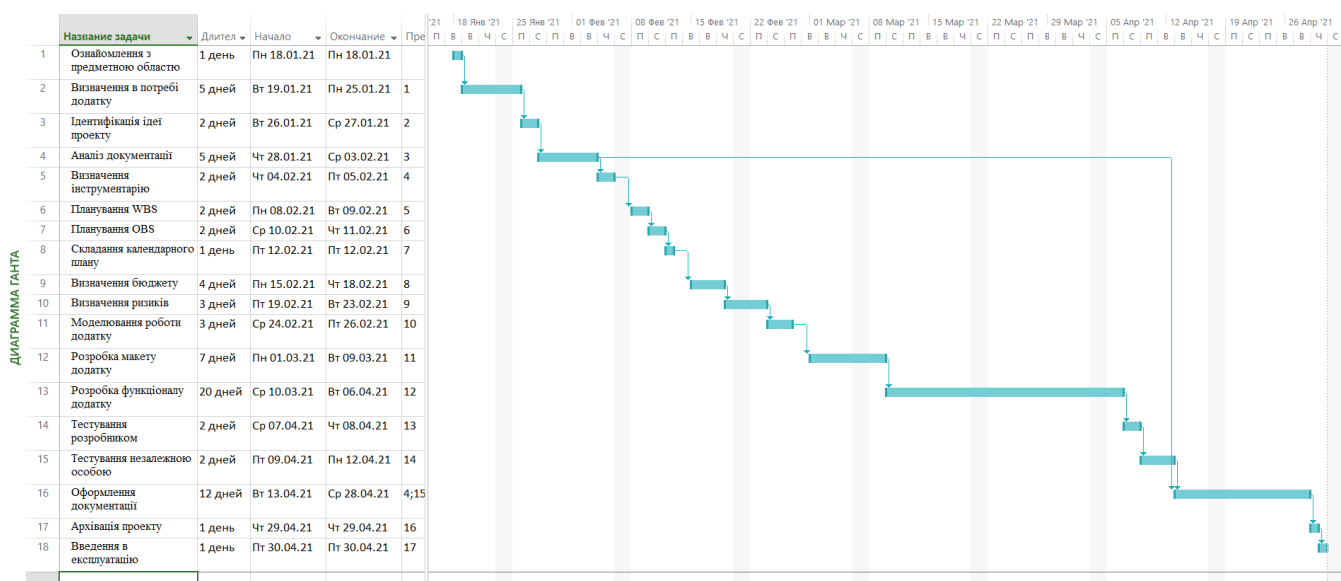


Рисунок Б.4 – Діаграма Ганта проекту




















		Режим задачи ▾	Название задачи ▾	Длитель ▾	Начало ▾	Окончание ▾	Предшественни ▾
1			Ознайомлення з предметною областю	1 день	Пн 18.01.21	Пн 18.01.21	
2			Визначення в потребі додатку	5 дней	Вт 19.01.21	Пн 25.01.21	1
3			Ідентифікація ідеї проекту	2 дней	Вт 26.01.21	Ср 27.01.21	2
4			Аналіз документації	5 дней	Чт 28.01.21	Ср 03.02.21	3
5			Визначення інструментарію	2 дней	Чт 04.02.21	Пт 05.02.21	4
6			Планування WBS	2 дней	Пн 08.02.21	Вт 09.02.21	5
7			Планування OBS	2 дней	Ср 10.02.21	Чт 11.02.21	6
8			Складання календарного плану	1 день	Пт 12.02.21	Пт 12.02.21	7
9			Визначення бюджету	4 дней	Пн 15.02.21	Чт 18.02.21	8
10			Визначення ризиків	3 дней	Пт 19.02.21	Вт 23.02.21	9
11			Моделювання роботи додатку	3 дней	Ср 24.02.21	Пт 26.02.21	10
12			Розробка макету додатку	7 дней	Пн 01.03.21	Вт 09.03.21	11
13			Розробка функціоналу додатку	20 дней	Ср 10.03.21	Вт 06.04.21	12
14			Тестування розробником	2 дней	Ср 07.04.21	Чт 08.04.21	13
15			Тестування незалежною особою	2 дней	Пт 09.04.21	Пн 12.04.21	14
16			Оформлення документації	12 дней	Вт 13.04.21	Ср 28.04.21	4;15
17			Архівація проекту	1 день	Чт 29.04.21	Чт 29.04.21	16
18			Введення в експлуатацію	1 день	Пт 30.04.21	Пт 30.04.21	17

ДИАГРАММА ГАНТА

Рисунок Б.5 – Список робіт для побудови діаграми Ганта

Аналіз ризиків. Ризик – ймовірнісна подія, яка може позитивно чи негативно вплинути на проект. Причиною виникнення ризиків є невизначеності, існуючі в кожному проекті. Ризики можуть бути «відомі» - ті, які визначені, оцінені, для яких можливе планування. Ризики «невідомі» - ті, які не ідентифіковані і не можуть бути прогнозовані. Хоча специфічні ризики і умови їх виникнення не визначені, але більшу частину ризиків можна передбачити.

Ідентифікація ризиків - визначення ризиків, здатних вплинути на проект, і документування їх характеристик.

Ідентифікація ризиків визначає, які ризики здатні вплинути на проект, і документує характеристики цих ризиків. Ідентифікація ризиків не буде ефективною, якщо вона не буде проводитися регулярно протягом реалізації проекту.

Ідентифікація ризиків повинна залучати якомога більше учасників: менеджерів проекту, користувачів, незалежних фахівців.

Класифікація ризиків:

1. За імовірністю виникнення:

- слабо ймовірнісні;
- мало ймовірнісні;
- імовірні;
- досить імовірні;
- майже імовірні.

2. За величиною втрат:

- мінімальна;
- низька;
- середня;
- висока;
- максимальна.

На основі цих даних була проведена класифікація ризиків для даного проекту, що наведена в таблиці Б.3.

Таблиця Б.3 – Класифікація ризиків

№	Назва ризику	Ймовірність	Величина втрат
1	Некоректно складене ТЗ	2	4
2	Недотримання календарного плану	1	3
3	Некоректна робота програмного забезпечення	4	4
4	Некоректна робота апаратного забезпечення	4	4
5	Хвороба розробника	2	2
6	Некоректне тестування	2	1

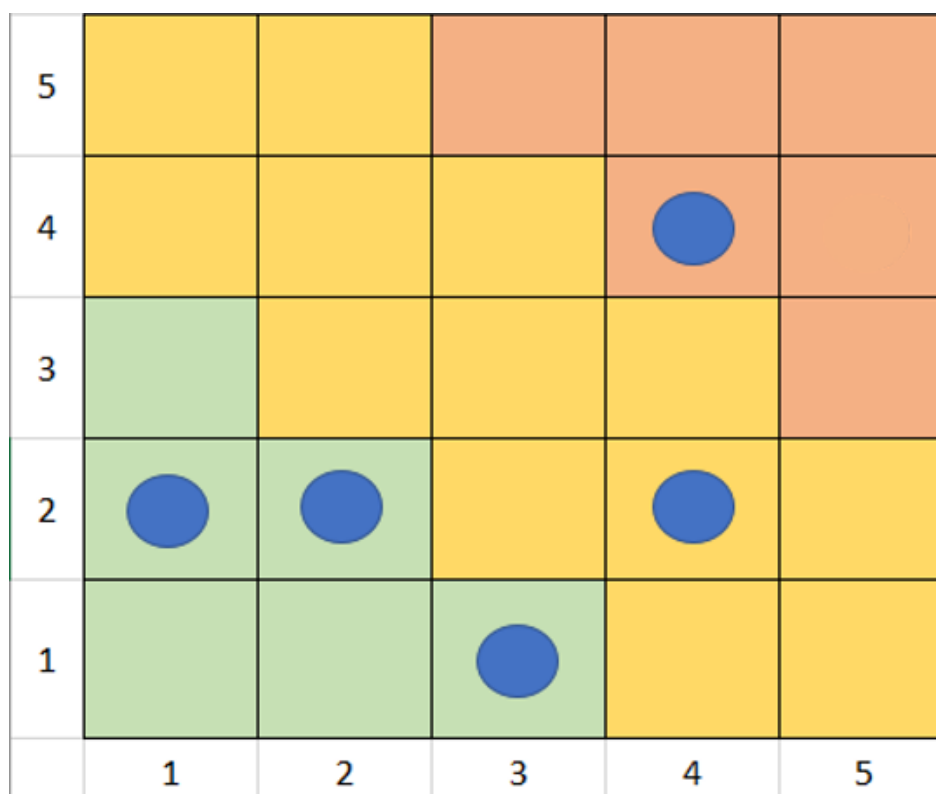


Рисунок Б.6 – Матриця ризиків

Далі визначаємо рівні ризиків та ступінь їх дії.

Рівні можуть бути:

- допустимі $1 < R < 4$;
- оправдані $5 < R < 10$;
- недопустимі $13 < R < 25$.

Ступінь дії ризиків:

- ті, що можна проігнорувати $1 < R < 4$;
- незначні $5 < R < 8$;
- помірні $9 < R < 10$;
- істотні $11 < R < 16$;
- критичні $17 < R < 25$.

На основі цих даних була виконана оцінка ступенів та рівнів для кожного ризику в проєкті. Результати роботи представлені в таблиці Б.3.

Таблиця Б.4 – Визначення ступенів та рівнів ризиків

№	Назва ризику	Ймовірність ризику	Величина втрат	Рівень ризику	Ступінь дії
1	Некоректно складене ТЗ	2	4	Оправданий	Незначний
2	Недотримання календарного плану	1	3	Допустимий	Проігнорувати
3	Некоректна робота програмного забезпечення	4	4	Недопустимий	Істотний

Продовження таблиці Б.4

№	Назва ризику	Ймовірність ризику	Величина втрат	Рівень ризику	Ступінь дії
4	Некоректна робота апаратного забезпечення	4	4	Недопустимий	Істотний
5	Хвороба розробника	2	2	Допустимий	Проігнорувати
6	Некоректне тестування	2	1	Допустимий	Проігнорувати

Після виконання прогнозування виникнення ризиків та їх ступеню впливу на результат реалізації проекту, були розроблені варіанти запобігання та реакції на кожний із них. Результати даного етапу представлені в таблиці Б.4.

Таблиця Б.5 – Варіанти запобігання та реакції на ризики

Ризики проекту	План запобігання ризику	План реакції на ризик
Некоректно складене ТЗ	Замовник повинен скласти детальне ТЗ, дотримуючись затвердженого плану: словник термінів, мета проекту, усі види вимог, терміни. Замовник та розробник повинні обговорити та затвердити його.	Уважно та чітко окреслити те, що було виконано невірно (після розмови із замовником) та зробити правки.

Продовження таблиці Б.5

Ризики проекту	План запобігання ризику	План реакції на ризик
Недотримання календарного плану	Створення плану реалізації проекту на основі ретельного аналізу списку всіх робіт. Затвердження зазначених термінів із замовником. Командна робота над планом термінів виконання. (Можливість внесення правок перед затвердженням усіма членами команди).	1. Обговорення варіантів внесення правок до термінів реалізації із керівником та замовником. 2. Домовитися про умови зміни термінів із замовником. Якщо це недопустимо, тоді переорганізувати роботу таким чином, щоб в результаті терміни виконувалися.
Некоректна робота програмного забезпечення	1. Встановлення ліцензійного програмного забезпечення з перевірених джерел перед початком роботи. 2. Забезпечити наявність антивірусного програмного забезпечення.	Перезапуск або переустановлення програми, яка дала збій.
Некоректна робота апаратного забезпечення	1. Раз на 4-6 місяців виконувати перевірку працездатності апаратного забезпечення	Виконати ремонт апаратного забезпечення, якщо терміни виконання завдань не дозволяють чекати, знайти тимчасову заміну.

Продовження таблиці Б.5

Хвороба розробника	Виконувати певну частину роботи в команді для того, щоб члени проекту змогли замінити один одного при необхідності. При плануванні термінів залишити декілька резервних днів для таких випадків.	Передати повноваження робітника іншому члену команди, якщо цього вимагають терміни виконання.
Некоректне тестування	Виконати пошук кваліфікованого тестувальника в даній предметній області.	Передати проект на додаткове тестування кваліфікованому спеціалісту.

ДОДАТОК В
Коди основних модулів
Файл FirstMethodUserControl.cs

```
using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Drawing;

using System.Data;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace MetalGrains

{

    public partial class FirstMethodUserControl : UserControl

    {

        public FirstMethodUserControl()

        {

            InitializeComponent();

            setupComboboxes();

        }

        public Graphics draw { get; private set; }

        private Font font = new Font("Arial", 10, FontStyle.Bold);

        public Brush insideColorBrush = Brushes.Red;

        public Brush intersectedColorBrush = Brushes.Green;

        private GrainType grainType = GrainType.InsideCircle;

        private int wholeGrainAmount = 0;

        public int WholeGrainAmount

        {

            get { return wholeGrainAmount; }

        }

    }

}
```

```
        set
        {
            wholeGrainAmount = value;
            wholeGrainAmountTextBox.Text = Convert.ToString(value);
        }
    }

private int partialGrainAmount = 0;
public int PartialGrainAmount
{
    get { return partialGrainAmount; }
    set
    {
        partialGrainAmount = value;
        partialGrainAmountTextBox.Text = Convert.ToString(value);
    }
}

private void radioButton_wholeGrain_CheckedChanged(object sender, EventArgs e)
{
    grainType = GrainType.InsideCircle;
}

private void radioButton_partialGrain_CheckedChanged(object sender, EventArgs e)
{
    grainType = GrainType.IntersectedCircle;
}

public void OnPictureBoxClicked(object sender, MouseEventArgs e,
Func<FirstMethodDrawing, bool> pointSaver)
{
    PictureBox pictureBox = (PictureBox)sender;
    draw = Graphics.FromImage(pictureBox.Image);
    Point point = new Point(e.Location.X - 5, e.Location.Y - 5);
    if (grainType == GrainType.InsideCircle)
```

```

    {
        draw.DrawString("+", font, insideColorBrush, point.X, point.Y);
        WholeGrainAmount = ++WholeGrainAmount;
    }
    else
    {
        draw.DrawString("+", font, intersectedColorBrush, point.X, point.Y);
        PartialGrainAmount = ++PartialGrainAmount;
    }
    FirstMethodDrawing firstMethodDrawing = new FirstMethodDrawing(point, grainType,
font);
    pointSaver(firstMethodDrawing);
    pictureBox.Refresh();
}

public void drawAllPoints(Stack<FirstMethodDrawing> pointsList, Graphics draw)
{
    foreach(FirstMethodDrawing drawing in pointsList)
    {
        if (drawing.grainType == GrainType.InsideCircle)
        {
            draw.DrawString("+", drawing.font, insideColorBrush, drawing.point.X,
drawing.point.Y);
        }
        else
        {
            draw.DrawString("+", drawing.font, intersectedColorBrush, drawing.point.X,
drawing.point.Y);
        }
    }
}

public void addNewFirstMethodFormulasUserControl()
{

```

```

firstMethodFormulasUserControl = new FirstMethodFormulasUserControl();

this.firstMethodFormulasUserControl.BorderStyle =
System.Windows.Forms.BorderStyle.FixedSingle;

this.firstMethodFormulasUserControl.Increase = MetalGrains.Increase.X_100;
this.firstMethodFormulasUserControl.Location = new System.Drawing.Point(29, 87);
this.firstMethodFormulasUserControl.Margin = new System.Windows.Forms.Padding(2);
this.firstMethodFormulasUserControl.Name = "firstMethodFormulasUserControl";
this.firstMethodFormulasUserControl.Size = new System.Drawing.Size(485, 408);
this.firstMethodFormulasUserControl.TabIndex = 13;
this.firstMethodFormulasUserControl.Visible = false;
this.Controls.Add(this.firstMethodFormulasUserControl);

firstMethodFormulasUserControl.Top = grainTypeChoosingPanel.Bottom + 20;
}

private void setupComboboxes()
{
    grainInsideColorComboBox.Items.AddRange(new string[] { "Червоний", "Жовтий",
"Зелений", "Чорний" });
    grainIntersectedColorComboBox.Items.AddRange(new string[] { "Червоний", "Жовтий",
"Зелений", "Чорний" });
    plusSizeComboBox.Items.AddRange(new string[] { "10", "12", "14", "16", "18", "20"
});

    grainInsideColorComboBox.SelectedIndex = 0;
    grainIntersectedColorComboBox.SelectedIndex = 2;
    plusSizeComboBox.SelectedIndex = 0;
}

private void plusSizeComboBox_SelectedIndexChanged(object sender, EventArgs e)
{
    int size;

    string selected = ((ComboBox)sender).SelectedItem.ToString();
    switch (selected)

```



```
{  
    case "10":  
        default:  
            size = 10;  
            break;  
    case "12":  
        size = 12;  
        break;  
    case "14":  
        size = 14;  
        break;  
    case "16":  
        size = 16;  
        break;  
    case "18":  
        size = 18;  
        break;  
    case "20":  
        size = 20;  
        break;  
}  
  
font = new Font("Arial", size, FontStyle.Bold);  
}  
}
```

Файл FirstMethodFormulasUserControl.cs

```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Drawing;  
using System.Data;
```

```

using System.Linq;
using System.Text;
using System.Windows.Forms;
using Word = Microsoft.Office.Interop.Word;

namespace MetalGrains
{
    public partial class FirstMethodFormulasUserControl : UserControl
    {
        private const string WRONG_FORMAT_MESSAGE = "Невірний формат";

        public int grainPoint { get; private set; }
        public int wholeGrainAmount { get; private set; }
        public int partialGrainAmount { get; private set; }
        public double totalGrainsNumber { get; private set; }
        public double grainNumberPer1mm2 { get; private set; }
        public double averageGrainSquare { get; private set; }
        public double averageGrainDiameter { get; private set; }

        private Increase increase = Increase.X_100;
        public Increase Increase
        {
            get
            {
                return increase;
            }
            set
            {
                increase = value;
                if(increase != Increase.X_100)
                {
                    setupGrainNumberPer1mm2FormulaWithIncrease();
                }
            }
        }
    }
}

```

```

    }
}

public FirstMethodFormulasUserControl()
{
    InitializeComponent();
}

private void FirstMethodFormulasUserControl_Load(object sender, EventArgs e)
{
    setTextBoxesKeyPressListeners();
}

private void countGrainPointButtonFirstMethod_Click(object sender, EventArgs e)
{
    grainPoint =
GrainPointCalculationHelper.getGrainPointByFirstMethod(totalGrainsNumber);

    grainPointTextBox.Text = Convert.ToString(grainPoint);
}

private void setTextBoxesKeyPressListeners()
{
    insideCircleGrainsNumberTextBox.KeyPress += textBox_KeyPress;
    intersectedGrainsNumberTextBox.KeyPress += textBox_KeyPress;
    totalGrainNumberFormulaElementTextBox.KeyPress += textBox_KeyPress;
    increaseTextBox.KeyPress += textBox_KeyPress;
    grainNumberPer1mm2FormulaElementTextBox.KeyPress += textBox_KeyPress;
    grainNumberPer1mm2FormulaElementTextBox2.KeyPress += textBox_KeyPress;

    insideCircleGrainsNumberTextBox.KeyUp += textBox_KeyUp;
    intersectedGrainsNumberTextBox.KeyUp += textBox_KeyUp;
    totalGrainNumberFormulaElementTextBox.KeyUp += textBox_KeyUp;
    increaseTextBox.KeyUp += textBox_KeyUp;
    grainNumberPer1mm2FormulaElementTextBox.KeyUp += textBox_KeyUp;
}

```

```
        grainNumberPer1mm2FormulaElementTextBox2.KeyUp += textBox_KeyUp;
    }

    private void textBox_KeyPress(object sender, KeyPressEventArgs e)
    {
        e.Handled = !Validator.IsNumericTextBoxInputValid(e.KeyChar);
    }

    private void textBox_KeyUp(object sender, KeyEventArgs e)
    {
        makeCountValueButtonEnabled(sender);
    }

    private void countTotalGrainsNumberButton_Click(object sender, EventArgs e)
    {
        try
        {
            wholeGrainAmount = Convert.ToInt16(insideCircleGrainsNumberTextBox.Text);
            partialGrainAmount = Convert.ToInt16(intersectedGrainsNumberTextBox.Text);

            totalGrainsNumber = wholeGrainAmount + 0.5 * partialGrainAmount;
            totalGrainsNumberFormulaResultTextBox.Text = totalGrainsNumber.ToString();

            totalGrainNumberLabel.Show();
            totalGrainsNumberFormulaResultTextBox.Show();

            makeCountGrainPointButtonEnabled();
        }
        catch (Exception ex)
        {
            MessageBox.Show(WRONG_FORMAT_MESSAGE);
        }
    }
}
```

```

private void countGrainNumberPer1mm2Button_Click(object sender, EventArgs e)
{
    try
    {
        double enteredTotalGrainNumber =
Convert.ToDouble(totalGrainNumberFormulaElementTextBox.Text);

        if (Increase == Increase.X_100)
        {
            grainNumberPer1mm2 = 2 * enteredTotalGrainNumber;
        }
        else
        {
            int enteredIncrease = Convert.ToInt16(increaseTextBox.Text);

            grainNumberPer1mm2 = 2 * Math.Pow(enteredIncrease / 100, 2) *
enteredTotalGrainNumber;
        }

        grainNumberPer1mm2FormulaResultTextBox.Text = Math.Round(grainNumberPer1mm2,
3).ToString();

        grainNumberPer1mm2Label.Show();
        grainNumberPer1mm2FormulaResultTextBox.Show();

        makeCountGrainPointButtonEnabled();
    }
    catch (Exception ex )
    {
        MessageBox.Show(WRONG_FORMAT_MESSAGE);
    }
}

private void countAverageGrainSquareButton_Click(object sender, EventArgs e)
{
    try
    {

```

```

        double enteredGrainNumberPer1mm2 =
Convert.ToDouble(grainNumberPer1mm2FormulaElementTextBox.Text);

        averageGrainSquare = 1 / enteredGrainNumberPer1mm2;

        averageGrainSquareFormulaResultTextBox.Text = Math.Round(averageGrainSquare,
3).ToString();

        averageGrainSquareLabel.Show();

        averageGrainSquareFormulaResultTextBox.Show();

        makeCountGrainPointButtonEnabled();
    }

    catch (Exception ex)
    {
        MessageBox.Show(WRONG_FORMAT_MESSAGE);
    }
}

private void countAverageGrainDiameterButton_Click(object sender, EventArgs e)
{
    try
    {
        double enteredGrainNumberPer1mm2 =
Convert.ToDouble(grainNumberPer1mm2FormulaElementTextBox2.Text);

        averageGrainDiameter = 1 / Math.Sqrt(enteredGrainNumberPer1mm2);

        averageGrainDiameterFormulaResultTextBox.Text =
Math.Round(averageGrainDiameter, 3).ToString();

        averageGrainDiameterLabel.Show();

        averageGrainDiameterFormulaResultTextBox.Show();

        makeCountGrainPointButtonEnabled();
    }

    catch (Exception ex)
    {
        MessageBox.Show(WRONG_FORMAT_MESSAGE);
    }
}

```

```

    }
}

private void makeCountGrainPointButtonEnabled()
{
    bool areAllValuesCounted =
        totalGrainsNumberFormulaResultTextBox.TextLength > 0 &&
        grainNumberPer1mm2FormulaResultTextBox.TextLength > 0 &&
        averageGrainSquareFormulaResultTextBox.TextLength > 0 &&
        averageGrainDiameterFormulaResultTextBox.TextLength > 0;
    if (areAllValuesCounted)
    {
        countGrainPointButton.Enabled = true;
    }
}

private void makeCountValueButtonEnabled(Object sender)
{
    TextBox textBox = (TextBox)sender;
    switch (textBox.Name)
    {
        case "insideCircleGrainsNumberTextBox":
        case "intersectedGrainsNumberTextBox":
            if (insideCircleGrainsNumberTextBox.TextLength > 0 &&
                intersectedGrainsNumberTextBox.TextLength > 0)
            {
                countTotalGrainsNumberButton.Enabled = true;
            }
            break;
        case "totalGrainNumberFormulaElementTextBox":
        case "increaseTextBox":
            if (totalGrainNumberFormulaElementTextBox.TextLength > 0 &&
                (increaseTextBox.TextLength > 0 || increaseTextBox.Visible == false))
            {

```

```

        countGrainNumberPer1mm2Button.Enabled = true;
    }
    break;
case "grainNumberPer1mm2FormulaElementTextBox":
    if (grainNumberPer1mm2FormulaElementTextBox.TextLength > 0)
    {
        countAverageGrainSquareButton.Enabled = true;
    }
    break;
case "grainNumberPer1mm2FormulaElementTextBox2":
    if (grainNumberPer1mm2FormulaElementTextBox2.TextLength > 0)
    {
        countAverageGrainDiameterButton.Enabled = true;
    }
    break;
default:
    break;
}
}

private void setupGrainNumberPer1mm2FormulaWithIncrease()
{
    grainNumberPer1mm2PictureBox.Visible = false;
    grainNumberPer1mm2Label.Top += 60;
    totalGrainNumberLabel2.Text = "mq = ";
    totalGrainNumberLabel2.Top += 60;
    totalGrainNumberFormulaElementTextBox.Top += 60;
    totalGrainNumberFormulaElementTextBox.Left -= 10;
    totalGrainNumberFormulaElementTextBox.Width += 10;
    grainNumberPer1mm2FormulaResultTextBox.Top += 60;
    grainNumberPer1mm2SecondFormulaPictureBox.Visible = true;
    increaseLabel.Visible = true;
    increaseLabel.Top += 60;
    increaseTextBox.Visible = true;
}

```



```

        increaseTextBox.Top += 60;

        countGrainNumberPer1mm2Button.Top += 60;
    }

    public void saveGostStructureValues ()
    {
        ReportDocHelper.putValue (ReportDocHelper.GRAIN_INSIDE_1,
insideCircleGrainsNumberTextBox.Text);

        ReportDocHelper.putValue (ReportDocHelper.GRAIN_INTERSECTED_1 ,
intersectedGrainsNumberTextBox.Text);

        ReportDocHelper.putValue (ReportDocHelper.GRAIN_TOTAL_1,
totalGrainsNumberFormulaResultTextBox.Text);

        ReportDocHelper.putValue (ReportDocHelper.GRAIN_ON_SQUARE_1,
grainNumberPer1mm2FormulaResultTextBox.Text);

        ReportDocHelper.putValue (ReportDocHelper.GRAIN_AVERAGE_SQUARE_1,
averageGrainSquareFormulaResultTextBox.Text);

        ReportDocHelper.putValue (ReportDocHelper.GRAIN_DIAMETER_1_1,
averageGrainDiameterFormulaResultTextBox.Text);

        ReportDocHelper.putValue (ReportDocHelper.GRAIN_POINT_1_1, grainPointTextBox.Text);
    }

    public void saveRealStructureValues ()
    {
        ReportDocHelper.putValue (ReportDocHelper.GRAIN_INSIDE_2,
insideCircleGrainsNumberTextBox.Text);

        ReportDocHelper.putValue (ReportDocHelper.GRAIN_INTERSECTED_2,
intersectedGrainsNumberTextBox.Text);

        ReportDocHelper.putValue (ReportDocHelper.GRAIN_TOTAL_2,
totalGrainsNumberFormulaResultTextBox.Text);

        ReportDocHelper.putValue (ReportDocHelper.GRAIN_ON_SQUARE_2,
grainNumberPer1mm2FormulaResultTextBox.Text);

        ReportDocHelper.putValue (ReportDocHelper.GRAIN_AVERAGE_SQUARE_2,
averageGrainSquareFormulaResultTextBox.Text);

        ReportDocHelper.putValue (ReportDocHelper.GRAIN_DIAMETER_1_2,
averageGrainDiameterFormulaResultTextBox.Text);

        ReportDocHelper.putValue (ReportDocHelper.GRAIN_POINT_1_2, grainPointTextBox.Text);
    }

```

```
    }  
}
```

Файл SecondMethodUserControl.cs

```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Drawing;  
using System.Data;  
using System.Linq;  
using System.Text;  
using System.Threading.Tasks;  
using System.Windows.Forms;  
  
namespace MetalGrains  
{  
    public partial class SecondMethodUserControl : UserControl  
    {  
        private const double pixelsInOneMillimeter = 3.78;  
  
        private int point = 1;  
        private double allLinesLength = 0;  
        public double AllLinesLength  
        {  
            get { return allLinesLength; }  
            set  
            {  
                allLinesLength = value;  
                allLinesLengthTextBox.Text = Convert.ToString(Math.Round(value, 3));  
            }  
        }  
    }  
}
```

```

private Point p1 = new Point(0, 0);
private Point p2 = new Point(0, 0);
public Graphics draw { get; private set; }
Pen pen = new Pen(Brushes.Red, 2.0F);

public SecondMethodUserControl()
{
    InitializeComponent();
}

public void OnPictureBoxClicked(object sender, MouseEventArgs e, Func<Line, bool>
lineSaver)
{
    PictureBox pictureBox = (PictureBox)sender;
    draw = Graphics.FromImage(pictureBox.Image);
    Line line = new Line(new Point(p1.X, p1.Y), new Point(e.Location.X,
e.Location.Y));
    if (point == 1)
    {
        p1 = e.Location;
    }
    if (point == 2)
    {
        p2 = e.Location;
        draw.DrawLine(pen, line.p1.X, line.p1.Y, line.p2.X, line.p2.Y);
        point = 1;
        AllLinesLength = AllLinesLength += getLineLength(line);
        lineSaver(line);
        pictureBox.Refresh();
    }
    else
    {
        point = 2;
    }
}

```

```

}

public void drawAllLines(Stack<Line> lineList, Graphics draw)
{
    foreach(Line line in lineList)
    {
        draw.DrawLine(pen, line.p1.X, line.p1.Y, line.p2.X, line.p2.Y);
    }
}

public double getLineLength(Line line)
{
    return Math.Sqrt(Math.Pow(line.p2.X - line.p1.X, 2) + Math.Pow(line.p2.Y -
line.p1.Y, 2)) / pixelsInOneMillimeter;
}

public void addNewSecondMethodFormulasUserControl()
{
    secondMethodFormulasUserControl = new
MetalGrains.SecondMethodFormulasUserControl();

    secondMethodFormulasUserControl.BorderStyle =
System.Windows.Forms.BorderStyle.FixedSingle;

    secondMethodFormulasUserControl.Increase = MetalGrains.Increase.X_100;
    secondMethodFormulasUserControl.Location = new System.Drawing.Point(43, 91);
    secondMethodFormulasUserControl.Margin = new System.Windows.Forms.Padding(2);
    secondMethodFormulasUserControl.Name = "secondMethodFormulasUserControl";
    secondMethodFormulasUserControl.Size = new System.Drawing.Size(315, 309);
    secondMethodFormulasUserControl.TabIndex = 2;
    secondMethodFormulasUserControl.Visible = false;
    Controls.Add(this.secondMethodFormulasUserControl);

    secondMethodFormulasUserControl.Top = totalLengthPanel.Bottom + 20;
}
}
}

```

Файл SecondMethodFormulasUserControl.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Drawing;
using System.Data;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace MetalGrains
{
    public partial class SecondMethodFormulasUserControl : UserControl
    {
        private const string WRONG_FORMAT_MESSAGE = "Невірний формат";

        public int grainPoint { get; private set; }
        public double allLinesLength { get; private set; }
        public int totalIntersectionNumber { get; private set; }
        public double averageGrainDiameter { get; private set; }

        private Increase increase = Increase.X_100;
        public Increase Increase
        {
            get
            {
                return increase;
            }
            set
            {
                increase = value;
                if (increase != Increase.X_100)

```

```
        {
            setupAverageGrainDiameterFormulaWithIncrease();
        }
    }
}

public SecondMethodFormulasUserControl()
{
    InitializeComponent();
}

private void SecondMethodFormulasUserControl_Load(object sender, EventArgs e)
{
    setTextBoxesKeyPressListeners();
}

private void setTextBoxesKeyPressListeners()
{
    allLinesLengthTextBox.KeyPress += textBox_KeyPress;
    totalIntersectionNumberTextBox.KeyPress += textBox_KeyPress;
    increaseSecondMethodFormulaTextBox.KeyPress += textBox_KeyPress;

    allLinesLengthTextBox.KeyUp += textBox_KeyUp;
    totalIntersectionNumberTextBox.KeyUp += textBox_KeyUp;
    increaseSecondMethodFormulaTextBox.KeyUp += textBox_KeyUp;
}

private void textBox_KeyPress(object sender, KeyPressEventArgs e)
{
    e.Handled = !Validator.IsNumericTextBoxInputValid(e.KeyChar);
}

private void textBox_KeyUp(object sender, KeyEventArgs e)
{

```

```
        makeCountValueButtonEnabled();
    }

private void countAverageDiameterButton_Click(object sender, EventArgs e)
{
    try
    {
        allLinesLength = Convert.ToDouble(allLinesLengthTextBox.Text);

        totalIntersectionNumber =
Convert.ToInt16(totalIntersectionNumberTextBox.Text);

        if(increase == Increase.X_100)
        {
            averageGrainDiameter = allLinesLength / totalIntersectionNumber;
        }
        else
        {
            int enteredIncrease =
Convert.ToInt16(increaseSecondMethodFormulaTextBox.Text);

            averageGrainDiameter = allLinesLength * 100 / totalIntersectionNumber /
enteredIncrease;
        }

        averageDiameterFormulaResultTextBox.Text = Math.Round(averageGrainDiameter,
3).ToString();

        averageDiameterFormulaResultPictureBox.Show();
        averageDiameterFormulaResultTextBox.Show();
        equalSignLabel14.Show();

        makeCountGrainPointButtonEnabled();
    }
    catch (Exception ex)
    {
        MessageBox.Show(WRONG_FORMAT_MESSAGE);
    }
}
```

```
    }  
}  
  
private void countGrainPointButton_Click(object sender, EventArgs e)  
{  
    grainPoint =  
GrainPointCalculationHelper.getGrainPointBySecondMethod(averageGrainDiameter);  
    grainPointTextBox.Text = grainPoint.ToString();  
}  
  
private void makeCountValueButtonEnabled()  
{  
    bool areAllValuesEntered =  
        allLinesLengthTextBox.TextLength > 0 &&  
        totalIntersectionNumberTextBox.TextLength > 0 &&  
        (increaseSecondMethodFormulaTextBox.TextLength > 0 ||  
increaseSecondMethodFormulaTextBox.Visible == false);  
  
    if (areAllValuesEntered)  
    {  
        countAverageDiameterButton.Enabled = true;  
    }  
}  
  
private void makeCountGrainPointButtonEnabled()  
{  
    if (averageDiameterFormulaResultTextBox.TextLength > 0)  
    {  
        countGrainPointButton.Enabled = true;  
    }  
}  
  
private void setupAverageGrainDiameterFormulaWithIncrease()  
{
```



```

averageDiameterFormulaPictureBox.Hide();

averageDiameterFormulaWithIncreasePictureBox.Show();

increaseSecondMethodFormulaPictureBox.Show();

equalSignLabel5.Show();

increaseSecondMethodFormulaTextBox.Show();

countGrainPointButton.Top += 40;

grainPointLabel.Top += 40;

grainPointTextBox.Top += 40;
}

public void saveGostStructureValues()
{
    ReportDocHelper.putValue(ReportDocHelper.TOTAL_LENGTH_1,
allLinesLengthTextBox.Text);

    ReportDocHelper.putValue(ReportDocHelper.TOTAL_INTERSACTION_AMOUNT_1,
totalIntersectionNumberTextBox.Text);

    ReportDocHelper.putValue(ReportDocHelper.GRAIN_DIAMETER_2_1,
averageDiameterFormulaResultTextBox.Text);

    ReportDocHelper.putValue(ReportDocHelper.GRAIN_POINT_2_1, grainPointTextBox.Text);
}

public void saveRealStructureValues()
{
    ReportDocHelper.putValue(ReportDocHelper.TOTAL_LENGTH_2,
allLinesLengthTextBox.Text);

    ReportDocHelper.putValue(ReportDocHelper.TOTAL_INTERSACTION_AMOUNT_2,
totalIntersectionNumberTextBox.Text);

    ReportDocHelper.putValue(ReportDocHelper.GRAIN_DIAMETER_2_2,
averageDiameterFormulaResultTextBox.Text);

    ReportDocHelper.putValue(ReportDocHelper.GRAIN_POINT_2_2, grainPointTextBox.Text);
}
}
}

```