

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова

Навчально-науковий інститут комп'ютерних наук та управління проектами

Кафедра програмного забезпечення автоматизованих систем

Спеціальність 121 «Інженерія програмного забезпечення»

Освітня програма «Інженерія програмного забезпечення»

«Допущений до захисту»

Завідувач кафедри

_____ проф. Приходько С.Б.

«___» _____ 2024 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття ступеня вищої освіти «магістр»

на тему: Нелінійна регресійна модель для оцінювання тривалості розробки програмного забезпечення інтернет-магазинів, та розробка програми для її реалізації

Виконав: студент групи 6151мз

_____ Ніколаєв А.К.

(підпис, ПІБ)

Керівник роботи:

доцент кафедри ПЗАС, к.т.н, доцент.

(посада, науковий ступень вчене звання)

_____ Макарова Л.М.

(підпис, ПІБ)

Миколаїв – 2024 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет кораблебудування імені адмірала Макарова

Навчально-науковий інститут комп'ютерних наук та управління проектами
Кафедра програмного забезпечення автоматизованих систем
Спеціальність 121 «Інженерія програмного забезпечення»
Освітня програма «Інженерія програмного забезпечення»

«ЗАТВЕРДЖУЮ»

Гарант освітньої програми

_____ проф. С.Б.Приходько

(підпис)

« 14 » жовтня 2024 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ
на здобуття ступеня вищої освіти «магістр»

Студенту Ніколаєву Артему Костянтиновичу

(Прізвище, ім'я, по батькові)

1. Тема роботи: Нелінійна регресійна модель для оцінювання тривалості розробки програмного забезпечення інтернет-магазинів, та розробка програми для її реалізації.

Керівник роботи доцент кафедри ПЗАС, к.т.н, доцент Макарова Л.М.

Затверджені наказом ректора № 1070-УЧ від « 11 » 10 2024 р.

2. Термін подання роботи: 09.12.2024 р.

3. Вихідні дані по роботі: _____

4. Перелік питань, що належать до розробки (найменування розділів) _____

Вступ (Актуальність теми. Зв'язок роботи з науковими програмами, планами, темами. Мета і завдання дослідження. Об'єкт дослідження. Предмет дослідження. Методи дослідження. Наукова новизна одержаних результатів. Практичне значення одержаних результатів. Особистий внесок здобувача. Апробація результатів досліджень. Публікації.); Огляд літератури та обґрунтування необхідності проведення досліджень за обраною темою; Викладення результатів власних досліджень з висвітленням того нового, що пропонується; Розділ з розробки програмного забезпечення; Організаційно-економічний розділ; Розділи з охорони праці та охорони навколишнього середовища; Висновки; Список використаних джерел; Додатки (технічне завдання, текст програми, опис програми, інструкція користувача, програма і методика випробувань програмного забезпечення)

5. Перелік презентаційних матеріалів Титульний слайд, Мета та завдання дослідження, Об'єкт та предмет дослідження, Наукова новизна та Практичне значення одержаних результатів, Апробація результатів досліджень та Публікації, Аналіз існуючих методів та моделей для оцінювання тривалості розробки програмного забезпечення, Обґрунтування актуальності подальшого дослідження, Побудова нелінійної регресійної моделі для оцінювання тривалості розробки програмного забезпечення інтернет-магазинів, Постановка задачі на розробку програми для оцінювання тривалості розробки програмного забезпечення інтернет-магазинів, Діаграма варіантів використання, Технічний проєкт програмного забезпечення, Результати розробки, Висновки.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання 14.10.2024 р.

КАЛЕНДАРНИЙ ПЛАН

Номер	Назва етапів роботи	Терміни виконання	Примітка
1.	Підготовка розділу ВСТУП	16.10.2024	НС*
2.	Підготовка розділу з огляду літератури та обґрунтування необхідності проведення досліджень за обраною темою	18.10.2024	НС*
3.	Підготовка розділу (ів) за результатами власних досліджень	21.10.2024	НС*
4.	Підготовка розділу з розробки програмного забезпечення	27.11.2024	
5.	Підготовка організаційно-економічного розділу	29.11.2024	
6.	Підготовка розділу з охорони праці	02.12.2024	
7.	Підготовка розділу з охорони навколишнього середовища	04.12.2024	
8.	Підготовка розділу ВИСНОВКИ	05.12.2024	
9.	Оформлення списку використаних джерел та додатків	06.12.2024	
10.	Подання на кафедрі ПЗАС тексту остаточного варіанту роботи, підписаного її керівником, у роздрукованому та електронному форматі разом із заявами щодо самостійності виконання роботи та ідентичності друкованої та електронної версії роботи (Додатки 1 і 2 «Порядку здійснення заходів з перевірки робіт на наявність текстових збігів/ідентичності/схожості із використанням програмно-технічних засобів», який введений в дію наказом ректора НУК за №20 від 20.01.2020 р.)	09.12.2024	не пізніше, ніж за 14 діб до захисту (згідно п.4.1 зазначеного Порядку)
11.	Підготовка презентації та доповіді	13.12.2024	
12.	Попередній захист роботи на засіданні кафедри ПЗАС	16.12.2024	
13.	Подання на кафедрі ПЗАС електронних версії наступних документів у форматі pdf: кваліфікаційної роботи; файлу-опису кваліфікаційної роботи (згідно Додатку до наказу ректора НУК за №287-уч від 19.05.2020 р.); презентації доповіді	19.12.2024	

* - за результатами наукового стажування (НС), яке було з 02.09.2024 по 13.10.2024 р.

Студент



Ніколаєв А.К.

(ПІБ)

Керівник роботи



Макарова Л.М.

(ПІБ)

РЕФЕРАТ

Ніколаєв Артем Костянтинович

«Нелінійна регресійна модель для оцінювання тривалості розробки програмного забезпечення інтернет-магазинів та розробка програми для її реалізації»

Кваліфікаційна робота на здобуття освітнього рівня магістра зі спеціальності 121 «Інженерія програмного забезпечення». Національний університет кораблебудування імені адмірала Макарова. Миколаїв, 2024 р.

Обсяг роботи: 93 стор., 6 табл., 11 рис., 26 використаних джерел, 5 додатків.

Актуальність теми: Достовірність оцінювання витрат, необхідних на розробку ПЗ, залишається однією з важливих проблем при розробці програмних проєктів, при чому потрібно враховувати різні аспекти, як то трудомісткість розробки, кількість та кваліфікацію розробників, складність задачі і таке інше. Все це впливає в кінцевому підсумку на тривалість розробки конкретного програмного проєкту. Таким чином, побудова нелінійної регресійної моделі для оцінювання тривалості розробки програмного забезпечення інтернет-магазинів є актуальним завданням. Це дозволить підвищити достовірність оцінювання тривалості проєктів з розробки програмного забезпечення інтернет-магазинів.

Мета та завдання дослідження: підвищення достовірності оцінювання тривалості розробки програмного забезпечення інтернет-магазинів за рахунок побудови нелінійної регресійної моделі. Завдання: проаналізувати особливості розробки програмного забезпечення інтернет-магазинів, проаналізувати існуючі методи та моделі для оцінювання тривалості розробки програмного забезпечення, побудувати однофакторну нелінійну регресійну модель для оцінювання тривалості розробки програмного забезпечення інтернет-магазинів, розробити програму для оцінювання тривалості розробки програмного

забезпечення інтернет-магазинів, використовуючи побудовану однофакторну нелінійну регресійну модель.

Об'єкт дослідження: процес оцінювання тривалості розробки програмного забезпечення інтернет-магазинів.

Предмет дослідження: однофакторна нелінійна регресійна модель для оцінювання тривалості розробки програмного забезпечення інтернет-магазинів.

Методи дослідження: методи теорії ймовірностей, математичної статистики, математичного моделювання, регресійного аналізу, об'єктно-орієнтованого програмування.

Наукова новизна: удосконалено однофакторну нелінійну регресійну модель для оцінювання тривалості розробки програмного забезпечення інтернет-магазинів з використанням нормалізуючого перетворення у вигляді десяткового логарифму та врахуванням викидів у даних, що дозволило підвищити достовірність оцінювання тривалості розробки програмного забезпечення інтернет-магазинів в порівнянні з існуючими моделями СОСОМО та ISBSG.

Практичне значення одержаних результатів: розробка програми для оцінювання тривалості розробки програмного забезпечення інтернет-магазинів.

Апробація результатів дослідження: основні положення і результати досліджень, викладені у кваліфікаційній роботі, пройшли апробацію на VII Всеукраїнській науково-практичній інтернет-конференції молодих вчених та студентів «Сучасні інформаційні системи та технології» (м. Херсон, 29.11.2024).

Публікації: основні результати кваліфікаційної роботи викладено у 1 науковій праці – матеріалах конференції.

Ключові слова: програмне забезпечення, оцінювання тривалості розробки, достовірність оцінювання, нелінійна регресійна модель, інтернет-магазин.

ABSTRACT

Nikolaiev Artem Kostiantynovych

«A regression model for estimating the duration of online stores software development and developing the software for its implementation»

Qualification work for obtaining a master's degree in the speciality 121 «Software Engineering». Admiral Makarov National University of Shipbuilding. Mykolaiv, 2024.

Volume of work: 93 pages, 6 tables, 11 figures, 26 references, 5 appendices.

Relevance of the topic: The reliability of estimating the costs required for software development remains one of the important problems in the development of software projects, which must take into account various aspects, such as the complexity of development, the number and qualifications of developers, the complexity of the task, and so on. All this ultimately affects the duration of the development of a particular software project. Thus, building a nonlinear regression model to estimate the duration of software development of online stores is an urgent task. This will increase the reliability of assessing the duration of software development projects for online stores.

Purpose and objectives of the study: increase reliability of estimating duration of developing software of online stores by constructing a non-linear regression model. Task: analyze the features of software development of online stores, analyze existing methods and models for estimating the duration of software development, build a single-factor non-linear regression model for estimating the duration of software development of online stores, develop a software to estimate the duration of software development of online stores using a constructed single-factor non-linear regression model.

The object of the study: the process of estimating the duration of software development of online stores.

Subject of the study: single-factor nonlinear regression model to estimate the duration of online stores software development.

Methods of the study: methods of probability theory, mathematical statistics, mathematical modeling, regression analysis, object-oriented programming.

Scientific novelty: improved single-factor nonlinear regression model for estimation of duration of software development of online stores using normalizing transformation in the form of decimal logarithm and taking into account outliers in data, which allowed to increase reliability of estimation of duration of software development of online stores in comparison with existing COCOMO and ISBSG models.

Practical significance of the obtained results: development of a software for assessing the duration of software development of online stores.

Testing of the research results: the main provisions and research results set forth in the qualification work were tested at the VII All-Ukrainian Scientific and Practical Internet Conference of Young Scientists and Students "Modern Information Systems and Technologies" (Kherson, 29.11.2024).

Publications: the main results of qualification work are set forth in 1 scientific work - conference materials.

Keywords: software, estimation of development duration, reliability of estimation, nonlinear regression model, online store.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ.....	9
ВСТУП.....	10
1 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ТА МОДЕЛЕЙ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ІНТЕРНЕТ-МАГАЗИНІВ	14
1.1 Особливості розробки ПЗ інтернет-магазинів.....	14
1.2 Аналіз існуючих методів та моделей для оцінювання тривалості розробки ПЗ.....	17
1.3 Обґрунтування актуальності подальшого дослідження	19
2 НЕЛІНІЙНА РЕГРЕСІЙНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ІНТЕРНЕТ-МАГАЗИНІВ	22
2.1 Методи побудови нелінійних регресійних моделей для оцінювання тривалості розробки ПЗ	22
2.2 Попередня обробка даних для побудови нелінійної регресійної моделі для оцінювання тривалості розробки ПЗ	24
2.3 Побудова нелінійної регресійної моделі для оцінювання тривалості розробки ПЗ інтернет-магазинів.....	27
3 РОЗРОБКА ПРОЕКТУ ПРОГРАМИ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ІНТЕРНЕТ-МАГАЗИНІВ	32
3.1 Постановка задачі на розробку програми для оцінювання тривалості розробки ПЗ інтернет-магазинів.....	32
3.2 Розробка ескізного проекту програми для оцінювання тривалості розробки ПЗ інтернет- магазинів.....	34
3.3 Розробка технічного проекту програми для оцінювання тривалості розробки ПЗ інтернет-магазинів.....	38
3.4 Розробка робочого проекту програми для оцінювання тривалості розробки ПЗ інтернет- магазинів.....	40
4 РОЗРАХУНОК ЕКОНОМІЧНОЇ ЕФЕКТИВНОСТІ ВІД РОЗРОБКИ І ВПРОВАДЖЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ПЗ ІНТЕРНЕТ-МАГАЗИНІВ	44
4.1 Розрахунок витрат на створення й експлуатацію ПЗ.....	44
4.2 Економічна ефективність розробки та впровадження ПЗ	47
5 ОХОРОНА ПРАЦІ	50
5.1 Загальні питання охорони праці.....	50
5.2 Аналіз шкідливих та небезпечних факторів при роботі з комп'ютером.....	52
5.3 Розробка заходів щодо зменшення впливу шкідливих та небезпечних факторів при роботі з комп'ютером.....	54

	8
6 ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА	58
6.1 Загальні питання охорони навколишнього середовища	58
6.2 Забруднення навколишнього середовища	60
6.3 Розробка заходів щодо зменшення забруднення навколишнього середовища.....	61
ВИСНОВКИ.....	63
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	65
ДОДАТОК А – ТЕХНІЧНЕ ЗАВДАННЯ НА РОЗРОБКУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ПЗ ІНТЕРНЕТ- МАГАЗИНІВ.....	69
ДОДАТОК Б – КОД ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ПЗ ІНТЕРНЕТ-МАГАЗИНІВ	75
ДОДАТОК В – ОПИС ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ПЗ ІНТЕРНЕТ-МАГАЗИНІВ	85
ДОДАТОК Г – ІНСТРУКЦІЯ КОРИСТУВАЧА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ПЗ ІНТЕРНЕТ-МАГАЗИНІВ.....	87
ДОДАТОК Д – ПРОГРАМА ТА МЕТОДИКА ВИПРОБУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ПЗ ІНТЕРНЕТ- МАГАЗИНІВ.....	91

**ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ,
СКОРОЧЕНЬ І ТЕРМІНІВ**

ЕД	емпіричні дані
ПЗ	програмне забезпечення
COCOMO	Constructive Cost Model
CMS	Content Management System
ISBSG	International Software Benchmarking Standards Group
HTML	HyperText Transfer Protocol
MMRE	Mean of Magnitude of Relative Errors
PRED	Percentage of Prediction
SaaS	Software as a Service
SEO	Search Engine Optimization

ВСТУП

Актуальність теми. Достовірність оцінювання витрат, необхідних на розробку ПЗ, залишається однією з важливих проблем при розробці програмних проєктів, при чому потрібно враховувати різні аспекти, як то трудомісткість розробки, кількість та кваліфікацію розробників, складність задачі і таке інше. Все це впливає в кінцевому підсумку на тривалість розробки конкретного програмного проєкту.

Для розробки інтернет-магазинів використовують різні підходи та технології в залежності від задачі. Це можуть бути CMS, наприклад, Wordpressразом з Woocommerce, Joomla!, Opencart, Magento, Shopify, Horoshop або Mercurio [1]. Або використання онлайн-конструктору, наприклад, Tilda, Wix, Squarespace, що є найпростішим варіантом, адже для цього не потрібно досконало знати мови програмування або веб-дизайн [2]. Більшість розробників пропонують рішення на одній із CMS.

Оцінювання тривалості розробки ПЗ інтернет-магазинів на початкових стадіях розробки може бути складною задачею, оскільки отримати необхідну інформацію буває важко. Особливо це характерно для невеликих компаній-розробників.

Оцінювання тривалості розробки програмних проєктів можна виконати, використовуючи моделі, в яких за основу взято трудомісткість робіт, наприклад, COCOMO [3] та ISBSG [4]. Однак ці нелінійні регресійні моделі, побудовані для різних умов та проєктів, не завжди адекватно відповідають конкретним умовам, що виникають у процесі роботи різних компаній-розробників ПЗ.

Маючи конкретні дані по своїм попереднім розробкам, компанія-розробник ПЗ може побудувати власну модель для оцінювання та прогнозування тривалості розробки проєктів. Оскільки розподіл тривалості та

трудомісткості розробки програмних проєктів зазвичай не підпорядковується нормальному закону розподілу [5 – 7], для адекватності моделі бажано побудувати нелінійну регресійну модель залежності тривалості розробки від її трудомісткості. Враховуючи, що моделі COCOMO та ISBSG були побудовані з використанням логарифмічного перетворення, також можна використати логарифмічне перетворення для побудови моделі.

Таким чином, побудова нелінійної моделі регресії для оцінювання тривалості розробки ПЗ інтернет-магазинів є актуальним завданням, вирішення якого підвищить достовірність оцінювання тривалості проєктів з розробки ПЗ інтернет-магазинів.

Мета дослідження. Метою роботи є підвищення достовірності оцінювання тривалості розробки ПЗ інтернет-магазинів за рахунок побудови нелінійної регресійної моделі.

Завдання дослідження. Для досягнення сформульованої мети дослідження необхідно вирішити наступні завдання:

- проаналізувати особливості розробки ПЗ інтернет-магазинів;
- проаналізувати існуючі методи та моделі для оцінювання тривалості розробки ПЗ;
- обґрунтувати актуальність подальшого дослідження;
- розглянути методи побудови нелінійних регресійних моделей для оцінювання тривалості розробки ПЗ;
- виконати попередню обробку даних для побудови нелінійної регресійної моделі для оцінювання тривалості розробки ПЗ інтернет-магазинів;
- побудувати однофакторну нелінійну регресійну модель для оцінювання тривалості розробки ПЗ інтернет-магазинів;
- оцінити якість побудованої однофакторної нелінійної регресійної моделі;

- порівняти побудовану однофакторну нелінійну регресійну модель з існуючими моделями COCOMO та ISBSG;
- розробити програму для оцінювання тривалості розробки ПЗ інтернет-магазинів, використовуючи побудовану однофакторну модель нелінійної регресії.

Об'єктом дослідження є процес оцінювання тривалості розробки ПЗ інтернет-магазинів.

Предметом дослідження є однофакторна нелінійна регресійна модель для оцінювання тривалості розробки ПЗ інтернет-магазинів.

Методи дослідження. Для рішення завдань дослідження були застосовані методи теорії ймовірності та математичної статистики, математичного моделювання, регресійного аналізу, об'єктно-орієнтованого програмування.

Наукова новизна отриманих результатів полягає в удосконаленні однофакторної нелінійної регресійної моделі для оцінювання тривалості розробки ПЗ інтернет-магазинів з використанням нормалізуючого перетворення у вигляді десяткового логарифму та врахуванням викидів у даних, що дозволило підвищити достовірність оцінювання тривалості розробки ПЗ інтернет-магазинів в порівнянні з існуючими моделями COCOMO та ISBSG.

Практичне значення одержаних результатів полягає у розробці програми для оцінювання тривалості розробки ПЗ інтернет-магазинів з використанням побудованої однофакторної нелінійної регресійної моделі.

Особистий внесок здобувача. Кваліфікаційна робота є самостійно виконаною працею. Усі результати, викладені у роботі, отримані автором особисто. У праці, яка опублікована у співавторстві [8], здобувачеві належить: побудова однофакторної нелінійної регресійної моделі для оцінювання тривалості розробки ПЗ інтернет-магазинів.

Апробація результатів роботи. Основні положення і результати досліджень, викладені у кваліфікаційній роботі, пройшли апробацію на VII

Всеукраїнській науково-практичній інтернет-конференції молодих вчених та студентів «Сучасні інформаційні системи та технології» (м. Херсон, 29.11.2024).

Публікації. Основні результати кваліфікаційної роботи викладено у 1 науковій праці – матеріалах конференції.

1 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ТА МОДЕЛЕЙ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ІНТЕРНЕТ-МАГАЗИНІВ

1.1 Особливості розробки ПЗ інтернет-магазинів

Зараз торгівля дедалі більше йде у мережевий простір – Інтернет. Покупці замовляють в інтернет-магазинах майже все: електронні пристрої, одяг, товари для дому та побуту, їжу. У такої торгівлі майже немає обмежень, інтернет-магазини працюють в режимі 24/7 без врахування кордонів та часових поясів.

Сучасний інтернет-магазин дозволяє користувачам сформувати замовлення, обрати спосіб оплати та доставки через браузер або мобільний застосунок. Також інтернет-магазин може пропонувати додаткові функції: особистий кабінет, історія покупок, список вподобань и таке інше. Головна мета будь-якого інтернет-магазину – надати можливість споживачам здійснювати покупки онлайн [9].

Відповідно, ПЗ інтернет-магазину – це ПЗ, значна частина якого знаходиться на сервері, а призначений для користувача інтерфейс сприймається або в браузері як веб-сторінки, або в мобільному застосунку.

Розробка ПЗ інтернет-магазину – важливий та відповідальний етап. Незрозумілий дизайн, неправильний шаблон, структура, в якій покупцям буде важко орієнтуватися – все це може відштовхнути клієнтів, обмежити відвідування магазину і, загалом, знизити продажі. Тому всім вказаним питанням потрібно приділити увагу при розробці ПЗ інтернет-магазину та створити простий та зручний для користувача інтернет-магазин.

Крім того, потрібно також зосередитися на просуванні інтернет-магазину та його рекламі. Для цього можна використовувати кілька стратегій:

- пошукову оптимізацію (SEO);
- рекламу в соціальних мережах;
- контекстну рекламу.

Існує багато рішень для створення інтернет-магазинів, зокрема:

- конструктори;
- CMS з відкритим вихідним кодом;
- CMS із коробки;
- SaaS рішення.

Всі вони мають подібні функціональні можливості, але відрізняються інструментами.

Конструктори – найпростіше рішення для створення інтернет-магазинів. Для роботи з конструкторами не потрібні знання мов програмування та спеціальні навички. Після придбання послуги хостингу деякі провайдери пропонують конструктори сайтів, серед яких можна обрати і шаблони для створення інтернет-магазинів. За допомогою конструктора та шаблону можна обрати оформлення сайту, сформувати його структуру. Додавання, редагування та зміна наповнення у розділах інтернет-магазину відбувається за допомогою Wysiwyg-редактора, подібного до звичайного офісного пакету програм, наприклад, Microsoft Office. Як приклад конструкторів, можна привести Tilda, Wix, Squarespace [2].

CMS або системи керування контентом дають гнучкість для швидкого ефективного створення, наповнення та управління інтернет-магазином. Працювати з ними також можуть користувачі без знань у галузі розробки ПЗ. Однак у даному випадку важливо, щоб обрана CMS була простою у використанні та надавала ефективні інструменти для управління інтернет-магазином.

До CMS з відкритим вихідним кодом можна віднести:

- WordPress з плагіном Woocommerce – одне з найбільш ефективних рішень для створення та управління інтернет-магазином [10, 11];
- OpenCart – просту, гнучку та багатофункціональну CMS для електронної комерції [12];
- Magento – платформу для електронної комерції, написану на PHP, яку використовують багато великих брендів та яка має три продукти для проектів різного масштабу [13].

Серед CMS із коробки варто відмітити Mercurio CMS – движок від українських розробників ПЗ, призначений саме для інтернет-магазинів. Mercurio CMS створена на основі фреймворків Laravel, Elasticsearch, Vue.JS. Вона дає розробникам ПЗ інтернет-магазинів широкі можливості для налаштування, зокрема вона легко інтегрується з популярними службами доставки Нова Пошта, Укрпошта, Meest, Justin [14].

Серед SaaS-рішень можна відмітити Shopify – хостингове рішення, яке просте у використанні, багатофункціональне, але має тарифні плани з помісячною оплатою. Також пропонує як платні, так і безкоштовні шаблони, гнучкі налаштування та підтримку [15].

Також в цій категорії можна відмітити Хорошоп – український продукт в сегменті SaaS-рішень. Його перевагою є інтеграція з багатьма популярними українськими маркетплейсами та агрегаторами, зокрема: Rozetka, Prom, Hotline, Price.ua [16].

Але яке б рішення не було обрано для створення та управління інтернет-магазином, все рівно потрібно вирішувати питання інтеграції зі сторонніми сервісами, масштабування, SEO оптимізації, підтримки актуальності контенту.

Крім того, одна з головних вимог сьогодні – адаптивність, тому що значна кількість користувачів інтернет-магазинів оформлюють замовлення через мобільні пристрої, в першу чергу смартфони, тому сайт має однаково коректно працювати як на комп'ютерах, так і на мобільних пристроях.

1.2 Аналіз існуючих методів та моделей для оцінювання тривалості розробки ПЗ

Найбільше використовуваними моделями для оцінювання тривалості розробки ПЗ сьогодні є COCOMO та ISBSG – параметричні нелінійні моделі регресії. Незалежною змінною в цих моделях виступає трудомісткість робіт, виражена у людино-годинах, залежною змінною виступає тривалість робіт, виражена у місяцях.

Модель COCOMO побудована для трьох типів проєктів (в залежності від кількості рядків програмного коду): органічного (2 000 – 50 000 рядків коду), напів-розділеного (50 000 – 300 000 рядків коду), вбудованого (більше 300 000 рядків коду) та складається з трьох рівнів: базового (двопараметрична модель, враховує тип проєкту та кількість рядків коду), середнього (уточнення моделі за рахунок 15 атрибутів вартості, які згруповані за 4 категоріями) та детального (підвищення точності досягається за рахунок ієрархічної декомпозиції ПЗ, а також обліку вартісних факторів на кожному рівні ієрархії та фазах робіт).

До органічного типу ПЗ відносять програми, які призначені для проміжних розрахунків, для наукових обчислень, для внутрішніх потреб розробників. До напів-розділеного типу ПЗ можна віднести, наприклад, прикладні системи або компілятори. До вбудованого типу ПЗ можна віднести мережі АТМ, військові системи, системи контролю повітряного руху [3].

Якщо прийняти, що один людино-місяць містить 152 людино-години, можна записати такі рівняння для попереднього оцінювання [17]:

– для органічного типу ПЗ:

$$D = 0,371 * E^{0,38}; \quad (1.1)$$

– для напів-розділеного типу ПЗ:

$$D = 0,431 * E^{0,35};$$

– для вбудованого типу ПЗ:

$$D = 0,501 * E^{0,32}.$$

У приведених формулах:

E – трудомісткість проекту в людино-годинах;

D – тривалість проекту в місяцях.

Модель ISBSG, розробник якої – International Software Benchmarking Standards Group [4], можна записати за допомогою такого рівняння для попереднього оцінювання [17]:

$$D = 0,662 * E^{0,328}, \quad (1.2)$$

де E – трудомісткість проекту в людино-годинах;

D – тривалість проекту в місяцях.

При всій своїй простоті використання для оцінювання тривалості розробки ПЗ, наведені моделі не дозволяють виконувати оцінювання довірчих інтервалів тривалості робіт.

1.3 Обґрунтування актуальності подальшого дослідження

Щоб перевірити якість регресійних моделей та рівнянь регресії, а також порівняти їх між собою, можна використати наведені нижче параметри якості.

Коефіцієнт детермінації R^2 розраховується за формулою [18]:

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2},$$

де y_i – емпіричне значення y ,

$\hat{y}$ – значення y , розраховане за рівнянням регресії,

$\bar{y}$ – середнє значення y , яке розраховується за формулою:

$$\bar{y} = \frac{1}{n} \sum_{i=1}^n y_i,$$

n – кількість значень y у вибірці.

Середнє значення відносної похибки $MMRE$ розраховується за формулою [19]:

$$MMRE = \frac{1}{n} \sum_{i=1}^N MRE_i, \quad (1.3)$$

де MRE – величина відносної похибки, яке розраховується за формулою:

$$MRE_i = \left| \frac{y_i - \hat{y}_i}{y_i} \right|.$$

Рівень прогнозування $Pred(0,25)$ розраховується за формулою [19]:

$$Pred(0,25) = \frac{k}{n}, \quad (1.4)$$

де k – кількість значень з $MRE \leq 0,25$.

Оцінемо тривалість розробки ПЗ інтернет-магазинів за даними, взятими з відповідних проектів, виконаних компанією – розробником ПЗ. В розрахунку розглядатимемо 62 проекти. Для побудови за розглянутими вище моделями COCOMO та ISBSG представимо дані у такому вигляді: Y – тривалість проекту – у місяцях, X – трудомісткість проекту – у людино-годинах. Фрагмент даних наведено у таблиці 1.1.

Таблиця 1.1 – Фрагмент вихідні дані для розрахунку

№ проекту	Y	X	№ проекту	Y	X
1	5,83	400	11	5,83	400
2	5,03	355	12	9,17	405
3	10,04	455	13	10,08	505
4	8,33	405	14	6,67	500
5	7,5	405	15	9,17	350
6	13,33	650	16	10,07	455
7	5,09	950	17	13,33	455
8	4,17	450	18	4,17	350
9	7,5	350	19	3,33	400
10	4,11	350	20	20,06	905

Усі розглянуті проекти мають невелику кількість рядків коду, тому можна віднести їх до органічного типу проектів моделі COCOMO та використати рівняння (1.1) для розрахунку. При використанні моделі ISBSG можна використати рівняння (1.2) для розрахунку. Також додатково можна виконати розрахунок залежності тривалості проекту від його трудомісткості за рівнянням, наведеним у роботі [7]:

$$\hat{Y} = 10^{-2,233} * X^{1,001}. \quad (1.5)$$

Параметри якості побудованих моделей за формулами (1.3), (1.4) наведено у таблиці 1.2.

Таблиця 1.2 – Параметри якості побудованих моделей

Параметр	Модель COCOMO	Модель ISBSG	Рівняння (1.5)
<i>MMRE</i>	0,4472	0,3337	0,6057
<i>Pred(0,25)</i>	0,1452	0,4032	0,0323

Проаналізувавши дані, наведені у таблиці 1.2, можна зробити висновок, що жодна із застосованих моделей не може бути використана для прогнозування залежності тривалості розробки ПЗ інтернет-магазинів, оскільки вони не мають задовільних значень по жодному із параметрів, а саме середня величина відносної похибки *MMRE* перевищує прийнятне значення 0,25 та рівень прогнозування *Pred(0,25)* значно нижче, ніж 0,75.

Провівши наведені попередні розрахунки, можемо зробити висновок про актуальності подальшого дослідження та необхідність побудови однофакторної моделі нелінійної регресії для оцінювання тривалості розробки ПЗ інтернет-магазинів з використанням нормалізуючого перетворення та врахуванням викидів у даних, що дозволить підвищити достовірність оцінювання тривалості розробки ПЗ інтернет-магазинів в порівнянні з існуючими моделями COCOMO та ISBSG.

2 НЕЛІНІЙНА РЕГРЕСІЙНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ІНТЕРНЕТ-МАГАЗИНІВ

2.1 Методи побудови нелінійних регресійних моделей для оцінювання тривалості розробки ПЗ

Регресійну модель для оцінювання тривалості розробки ПЗ можна записати таким чином [19]:

$$y = \hat{y} + \varepsilon = f(x) + \varepsilon, \quad (2.1)$$

де: y – залежна змінна,

$\hat{y}$ – розраховане значення залежної змінної y ,

ε – випадкова помилка, що розподілена згідно нормального закону розподілу,

$f(x)$ – лінійна або нелінійна функція, яка визначає вид регресійної моделі,

x – незалежна змінна.

У разі, коли функція $f(x)$ нелінійна, маємо модель нелінійної регресії. Якщо розподіл випадкових величин, які використовуються для побудови моделі для оцінювання тривалості розробки ПЗ, не є нормальним, потрібно використовувати нормалізовані значення тривалості та трудомісткості проектів. Отримати нормалізовані значення випадкових величин можна за допомогою нормалізуючих перетворень. В даній роботі буде використано нормалізуюче перетворення на основі десяткового логарифму, яке застосуємо до обох даних, залежної та незалежної змінних [19]:

$$z_x = \log_{10} x, \quad (2.2)$$

$$z_y = \log_{10} y, \quad (2.3)$$

де: z_x, z_y – нормально розподілені дані,

x, y – величини, які нормалізуються.

Нормалізуючим перетворенням (2.2) – (2.3) відповідають зворотні перетворення [19]:

$$x = 10^{z_x}, \quad (2.4)$$

$$y = 10^{z_y}. \quad (2.5)$$

В разі відповідності випадкових величин нормальному закону розподілу, будемо модель лінійної регресії таким чином [19]:

$$z_y = \hat{z}_y + \varepsilon = \hat{b}_0 + \hat{b}_1 z_x + \varepsilon \quad (2.6)$$

де: z_y – залежна змінна,

$\hat{z}_y$ – розраховане значення залежної змінної z_y ,

ε – нормально розподілена випадкова величина,

$\hat{b}_0, \hat{b}_1$ – оцінки коефіцієнтів лінійної регресії,

z_x – незалежна змінна.

Оцінки коефіцієнтів лінійної регресії $\hat{b}_0, \hat{b}_1$, можна знайти, використовуючи метод найменших квадратів [19]:

$$b_1 = \frac{n \sum_{i=1}^n z_{xi} z_{yi} - \sum_{i=1}^n z_{xi} \cdot \sum_{i=1}^n z_{yi}}{n \sum_{i=1}^n z_{xi}^2 - \left(\sum_{i=1}^n z_{xi} \right)^2}, \quad (2.7)$$

$$b_0 = \frac{\sum_{i=1}^n z_{yi} - b_1 \sum_{i=1}^n z_{xi}}{n}. \quad (2.8)$$

Якщо нормалізуючим перетворенням випадкових величин використовувати десятковий логарифм згідно з рівняннями (2.2), (2.3), тоді потрібну модель нелінійної регресії для оцінювання тривалості розробки ПЗ з урахуванням зворотних перетворень (2.4), (2.5), та (2.1) можна записати таким чином [19]:

$$\hat{y} = 10^{\varepsilon + \hat{b}_0} x^{\hat{b}_1}. \quad (2.9)$$

Використовуючи метод нормалізуючих перетворень для рішення задачі побудови моделі нелінійної регресії для оцінювання тривалості розробки ПЗ, можна отримати прийнятні результати якості побудованої моделі.

2.2 Попередня обробка даних для побудови нелінійної регресійної моделі для оцінювання тривалості розробки ПЗ

Для задачі побудови нелінійної регресійної моделі для оцінювання тривалості розробки ПЗ маємо двовимірні дані тривалості проєкту Y та трудомісткості X . Тому у цьому випадку попередня обробка даних включатиме в себе перевірку вихідних даних на відповідність багатовимірному

нормальному закону розподілу та перевірку вихідних даних на багатовимірні викиди.

Для перевірки вихідних даних на відповідність багатовимірному нормальному закону розподілу можна використати багатовимірний ексцес β_2 , значення якого можна записати таким чином [20]:

$$\beta_2 = \frac{1}{n} \sum_{i=1}^n \{(Z_i - \bar{Z})^T S_n^{-1} (Z_i - \bar{Z})\}^2, \quad (2.10)$$

де: Z_i – i -ий компонент даних, що має координати Z_{x_i}, Z_{y_i} відповідно,

$\bar{Z}$ – вектор вибірових середніх,

n – кількість елементів у вибірці,

S_n – коваріаційна матриця, яку можна записати таким чином:

$$S_n = \frac{1}{n} \sum_{i=1}^n (Z_i - \bar{Z}) (Z_i - \bar{Z})^T. \quad (2.11)$$

Для багатовимірного нормального розподілу даних значення β_2 можна записати таким чином [20]:

$$\beta_2 = m(m+2), \quad (2.12)$$

де m – кількість параметрів у наборі даних.

Якщо значення багатовимірного ексцесу β_2 , розраховане за формулою (2.10) з урахуванням (2.11), перевищує значення, розраховане за формулою (2.12), можна зробити висновок, що вихідні дані не відповідають багатовимірному нормальному закону розподілу.

Також про невідповідність вихідних даних багатовимірному нормальному закону розподілу може свідчити наявність викидів у вихідному наборі даних.

Для перевірки багатовимірних даних на викиди можна використати квадрат відстані Махаланобіса [21]:

$$d_i^2 = (Z_i - \bar{Z})^T S_n^{-1} (Z_i - \bar{Z}), \quad (2.13)$$

де: Z_i – i -ий компонент даних, що має координати Z_{x_i}, Z_{y_i} відповідно,

$\bar{Z}$ – вектор вибірових середніх,

n – кількість елементів у вибірці,

S_n – коваріаційна матриця, яку можна знайти за формулою (2.11).

За значенням d_i^2 , розрахованим за формулою (2.13), можливо зробити висновок, чи є точка даних викидом, використовуючи критерій Пірсона χ^2 . Значення d_i^2 порівнюється з квантилем розподілу Пірсона χ^2 . Рівень значущості α для розрахунку приймається рівним 0,005. Точка даних, для якої значення d_i^2 , розраховане за формулою (2.13), перевищує квантиль розподілу Пірсона χ^2 , є викидом.

Наведений алгоритм був представлений у роботах [22, 23]. Даний алгоритм є ітераційним і виконується доти, доки є точки даних, для яких d_i^2 перевищує квантиль розподілу Пірсона χ^2 .

Для одновимірних даних перевірку на відповідність нормальному закону розподілу можна виконати за допомогою критерію згоди χ^2 Пірсона [19]:

$$\chi^2 = \sum_{i=1}^m \frac{(n_i - np_i)^2}{np_i}, \quad (2.14)$$

де m – кількість підінтервалів,

n – кількість елементів у вибірці,

де n_i – кількість значень, які потрапили у i -й підінтервал,

p_i – ймовірність потрапляння у i -й підінтервал.

Кількість підінтервалів можна розрахувати за допомогою формули Старджеса [19]:

$$m = 3,31 \cdot \lg n + 1, \quad (2.15)$$

задавши рівень значущості α та кількість ступенів вільності ν :

$$\nu = m - k - 1,$$

де k – кількість параметрів, від яких залежить закон розподілу, за таблицею верхніх $100\alpha\%$ точок розподілу χ^2 можна знайти значення $\chi^2_{кр}$. Якщо розраховане значення критерію згоди χ^2 Пірсона не перевищує критичне, з довірчою ймовірністю $1-\alpha$ можна стверджувати, що закон розподілу відповідає нормальному. Для нормального закону розподілу $k=2$. Значення α приймемо 0,05.

2.3 Побудова нелінійної регресійної моделі для оцінювання тривалості розробки ПЗ інтернет-магазинів

Однофакторна модель нелінійної регресії для оцінювання тривалості розробки ПЗ інтернет-магазинів буде побудована з використанням нормалізуючого перетворення у вигляді десяткового логарифму згідно з перетвореннями (2.2) та (2.3) та враховуванням пошуку викидів у багатовимірних даних з використанням квадрату відстані Махаланобіса згідно з формулою (2.13).

Для побудови однофакторної нелінійної регресійної моделі використовувалися дані, взяті з відповідних проектів, виконаних компанією –

розробником ПЗ. Для побудови використовується 62 проекти, де Y – тривалість проекту у місяцях, X – трудомісткість проекту у людино-годинах. Фрагмент даних наведено у таблиці 1.1.

Перевіримо вихідні дані на відповідність багатовимірному нормальному закону розподілу з використанням багатовимірного ексцесу β_2 , використовуючи для цього формули (2.10) – (2.12).

Значення багатовимірного ексцесу β_2 , розраховане за формулою (2.10), дорівнює 18,02, тоді як значення багатовимірного ексцесу β_2 , розраховане за формулою (2.12), для двовимірного нормального дорівнює 8. Вихідні дані не відповідають багатовимірному нормальному закону розподілу.

Подальша обробка вихідних даних виконувалася згідно з методикою, наведеною в роботі [24]. Спершу вихідні дані були нормалізовані за допомогою десятичного логарифму згідно з (2.2) та (2.3). Далі за допомогою квадрату відстані Махаланобіса було виконано визначення викидів у двовимірних даних згідно з (2.13).

Фрагмент вихідних даних, нормалізованих даних та значення квадрату відстані Махаланобіса наведені в таблиці 2.1.

Таблиця 2.1 – Фрагмент вихідних даних, нормалізованих даних та значення квадрату відстані Махаланобіса

№ проекту	Y	X	Z_y	Z_x	SMD
1	2	3	4	5	6
1	5,83	400	0,7657	2,6021	0,27
2	5,03	355	0,7016	2,5502	0,99
3	10,04	455	1,0017	2,6580	1,05
4	8,33	405	0,9206	2,6075	0,71
5	7,5	405	0,8751	2,6075	0,34
6	13,33	650	1,1248	2,8129	3,01
7	5,09	950	0,7067	2,9777	16,23
8	4,17	450	0,6201	2,6532	2,53

Продовження таблиці 2.1

1	2	3	4	5	6
9	7,5	350	0,8751	2,5441	1,52
10	4,11	350	0,6138	2,5441	1,92
11	5,83	400	0,7657	2,6021	0,27
12	9,17	405	0,9624	2,6075	1,23
13	10,08	505	1,0035	2,7033	0,81
14	6,67	500	0,8241	2,6990	0,37
15	9,17	350	0,9624	2,5441	2,84
16	10,07	455	1,0030	2,6580	1,07
17	13,33	455	1,1248	2,6580	3,52
18	4,17	350	0,6201	2,5441	1,83
19	3,33	400	0,5224	2,6021	4,09
20	20,06	905	1,3023	2,9566	9,19

Обчислимо квантиль розподілу Пірсона χ^2 при рівні значущості $\alpha=0,005$, $\chi^2=10,60$. Тобто точка даних, для якої значення d_i^2 , розраховане за формулою (2.13), перевищує $\chi^2=10,60$, є викидом. На першій ітерації було знайдено одну точку даних, для якої $d_i^2=16,23$. Вважаємо її викидом та видаляємо з подальших розрахунків.

Далі будуємо модель лінійної регресії для нормалізованих даних згідно з рівнянням (2.6). Коефіцієнти рівняння лінійної регресії знаходимо згідно з (2.7), (2.8). $\hat{b}_0=-1,8346$, $\hat{b}_1=1,0148$. Лінійна регресійна модель має вигляд:

$$z_y = -1,8346 + 1,0148z_x + \varepsilon.$$

Перевіримо випадкову величину ε , чи розподілена вона згідно нормального закону розподілу за допомогою критерію згоди χ^2 Пірсона за формулою (2.14) з урахуванням (2.15). Для довірчої ймовірності 0,95 розрахункове значення критерію згоди χ^2 дорівнює 8,57, критичне значення

критерію згоди χ^2 дорівнює 9,48. Отже, можемо зробити висновок, що розподіл випадкової величини ε відповідає нормальному закону розподілу і побудована лінійна регресійна модель може бути використана.

Наступним кроком будуємо модель нелінійної регресії для вихідних даних без викидів згідно з рівнянням (2.9). Нелінійна регресійна модель для оцінювання тривалості розробки ПЗ інтернет-магазинів має вигляд:

$$\hat{Y} = 10^{\varepsilon-1,8346} X^{1,0148}.$$

Виконаємо оцінювання якості побудованої нелінійної регресійної моделі за допомогою таких параметрів: середньої величини відносної похибки *MMRE* згідно з формулою (1.3) та рівня прогнозування *Pred(0,25)* згідно з формулою (1.4): *MMRE*=0,2686, *Pred(0,25)*=0,6066.

Порівняємо якість побудованої нелінійної регресійної моделі для оцінювання тривалості розробки ПЗ інтернет-магазинів з існуючими моделями COCOMO, рівняння (1.1), ISBSG, рівняння (1.2) та моделлю (1.5). Дані показники наведені в таблиці 1.2.

Після аналізу даних можемо бачити, що модель нелінійної регресії для оцінювання тривалості розробки ПЗ інтернет-магазинів, яка побудована у даній роботі, має кращі параметри якості у порівнянні з усіма наведеними для порівняння моделями.

Однак, а ні за параметром *MMRE*, а ні за параметром *Pred(0,25)* побудована модель не відповідає необхідним умовам: *MMRE*=0,2686 трохи вище за 0,25, *Pred(0,25)*=0,6066 нижче, ніж 0,75.

Можемо зробити наступний висновок. Однофакторна нелінійна регресійна модель для оцінювання тривалості розробки ПЗ інтернет-магазинів, яка побудована з використанням нормалізуючого перетворення десяткового логарифму та врахуванням викидів у вихідних даних, дозволила підвищити

достовірність оцінювання тривалості розробки відповідного ПЗ в порівнянні з існуючими моделями COCOMO та ISBSG. Однак у подальшій роботі потрібно використовувати інші нормалізуючі перетворення, наприклад, перетворення Бокса-Кокса або перетворення Джонсона, щоб побудована модель відповідала всім вимогам, зокрема вимогам до параметрів якості моделі.

З РОЗРОБКА ПРОЕКТУ ПРОГРАМИ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ІНТЕРНЕТ-МАГАЗИНІВ

3.1 Постановка задачі на розробку програми для оцінювання тривалості розробки ПЗ інтернет-магазинів

Першим етапом розробки проекту програми для оцінювання тривалості розробки ПЗ інтернет-магазинів є постановка задачі на розробку. Будемо використовувати побудовану раніше модель нелінійної регресії для оцінювання тривалості розробки ПЗ інтернет-магазинів.

На основі аналізу існуючих моделей, що використовуються для оцінювання тривалості розробки ПЗ, сформулюємо постановку задачі на розробку програми.

Програма, що розробляється, повинна задовольняти таким вимогам.

Вимоги до складу виконуваних функцій

Програма повинна давати можливість виконувати такі функції:

- Введення даних про параметри регресійної моделі (значення коефіцієнтів рівняння регресії).
- Введення значення довірчої ймовірності.
- Введення назви проекту розробки інтернет-магазину та значення трудомісткості його розробки для подальшого розрахунку.
- Оцінювання трудомісткості розробки інтернет-магазину.
- Збереження результатів оцінювання трудомісткості розробки інтернет-магазину в базу даних.

Вимоги до організації вхідних та вихідних даних

Вимоги до вхідних даних

Вхідні дані вводяться у відповідні поля форми відповідно до допустимих типів (числові або текстові).

До вхідних даних належать: параметри регресійної моделі b_0 , b_1 , значення довірчої ймовірності, назва проєкту, значення трудомісткості розробки.

Вимоги до вихідних даних

Вихідні дані виводяться у відповідні поля форми та зберігаються в базі даних.

Вихідними даними є прогнозована тривалість розробки ПЗ інтернет-магазинів.

Вимоги до складу та параметрів технічних засобів

Для роботи з програмним продуктом необхідно, щоб технічні засоби відповідали наступним мінімальним вимогам:

- CPU: Core i3;
- RAM: 1 Гб;
- HDD/SDD 500 Мб;
- пристрої вводу-виводу: монітор роздільною здатністю не менше ніж 1600x900 пікселів, клавіатура, маніпулятор типу «миша».

Вимоги до інформаційної та програмної сумісності

Програма повинна працювати в операційній системі Windows 10 або новіших версіях.

Детальна постановка задачі наведена у технічному завданні у Додатку А.

3.2 Розробка ескізного проекту програми для оцінювання тривалості розробки ПЗ інтернет-магазинів

Розробку проекту програми для оцінювання тривалості розробки ПЗ інтернет-магазинів починаємо з ескізного проекту, в рамках якого були визначені актори, варіанти використання програми, розроблені відповідні специфікації та сценарії варіантів використання та прототип користувацького інтерфейсу програми.

Ця програма передбачає наявність тільки одного актора – користувача програми, який буде використовувати всі функціональні можливості програми.

Виявлені варіанти використання програми наведені у таблиці 3.1.

Таблиця 3.1 – Варіанти використання програми для оцінювання тривалості розробки ПЗ інтернет-магазинів

Варіант використання	Короткий опис
Введення параметрів регресійної моделі	Призначений для введення параметрів розробленої моделі: значень коефіцієнтів рівняння регресії.
Введення довірчої ймовірності	Призначений для введення значення довірчої ймовірності для розрахунку тривалості.
Введення даних про проєкт	Призначений для введення назви проєкту та значення трудомісткості його розробки.
Оцінювання трудомісткості розробки	Призначений для оцінювання тривалості розробки проєкту в залежності від заданої трудомісткості.
Збереження результатів оцінювання	Призначений для збереження результатів оцінювання трудомісткості розробки інтернет-магазину в файл бази даних.

Діаграма варіантів використання програми для оцінювання тривалості розробки ПЗ інтернет-магазинів наведена на рисунку 3.1.

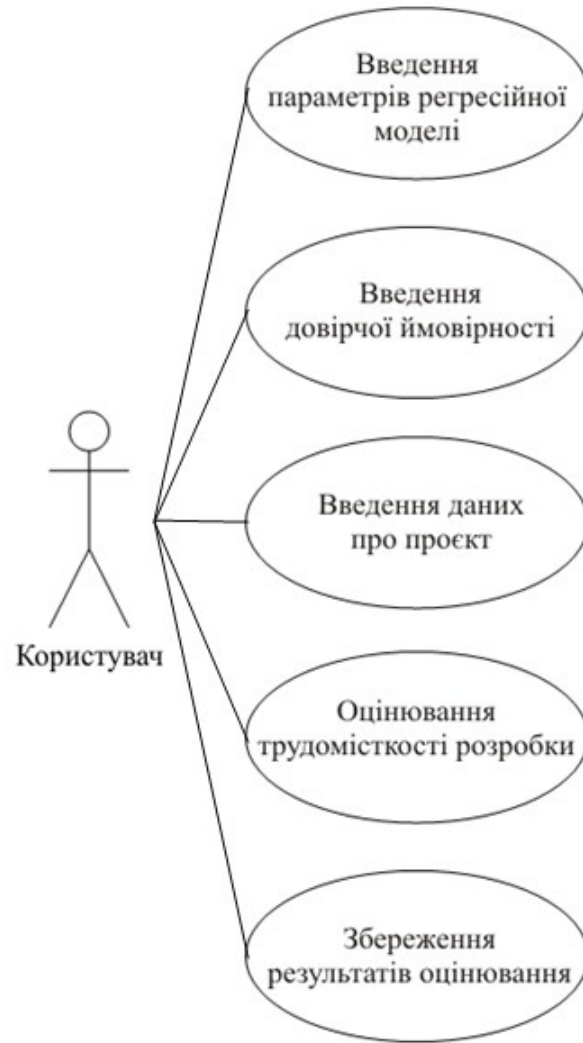


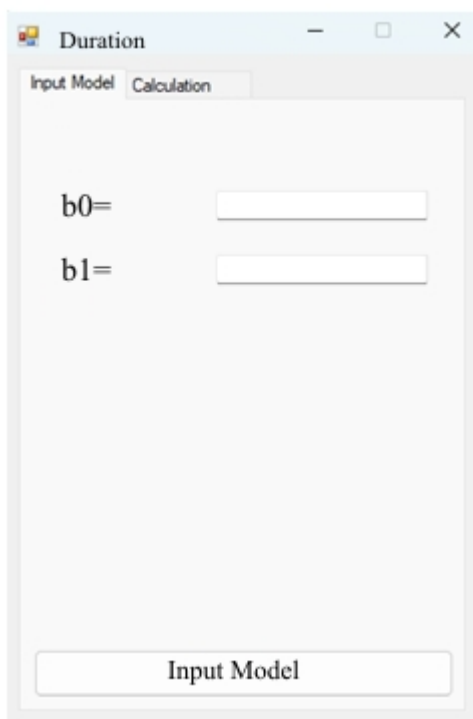
Рисунок 3.1 – Діаграма варіантів використання програми для оцінювання тривалості розробки ПЗ інтернет-магазинів

Сценарії варіантів використання програми для оцінювання тривалості розробки ПЗ інтернет-магазинів відображені у вигляді діаграми діяльності на рисунку 3.2.



Рисунок 3.2 – Діаграма діяльності програми для оцінювання тривалості розробки ПЗ інтернет-магазинів

Прототип інтерфейсу користувача програми для оцінювання тривалості розробки ПЗ інтернет-магазинів представлено на рисунках 3.3, 3.4.



The screenshot shows a window titled "Duration" with two tabs: "Input Model" (selected) and "Calculation". The "Input Model" tab contains two input fields labeled "b0=" and "b1=" with empty text boxes next to them. At the bottom of the window is a button labeled "Input Model".

Рисунок 3.3 – Прототип вкладки введення даних регресійної моделі



The screenshot shows the same "Duration" window, but with the "Calculation" tab selected. The "Input Model" tab is now disabled. The "Calculation" tab contains a section labeled "Input for calculation" with three input fields labeled "Project", "X=", and " α =" with empty text boxes next to them. Below this is a section labeled "Result" with a text box labeled "Duration:". At the bottom of the window are two buttons: "Calculate" and "Save".

Рисунок 3.4 – Прототип вкладки оцінювання трудомісткості розробки

На цьому розробку ескізного проєкту можна вважати завершеною і переходити до наступного кроку проєктування.

3.3 Розробка технічного проєкту програми для оцінювання тривалості розробки ПЗ інтернет-магазинів

Наступним кроком проєктування була розробка технічного проєкту, в рамках якого було розроблено діаграму класів програми для оцінювання тривалості розробки ПЗ інтернет-магазинів та динамічну модель.

Діаграму класів програми для оцінювання тривалості розробки ПЗ інтернет-магазинів представлено на рисунку 3.5.

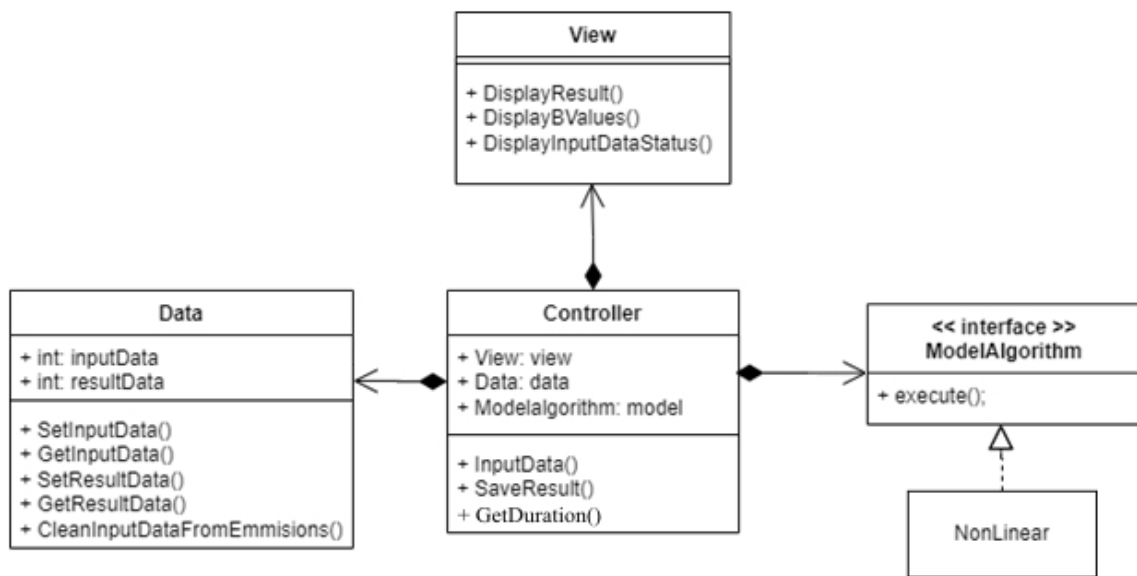


Рисунок 3.5 – Діаграма класів програми для оцінювання тривалості розробки ПЗ інтернет-магазинів

Динамічна модель програми для оцінювання тривалості розробки ПЗ інтернет-магазинів була відображена у вигляді діаграми послідовностей на рисунках 3.6 - 3.7.

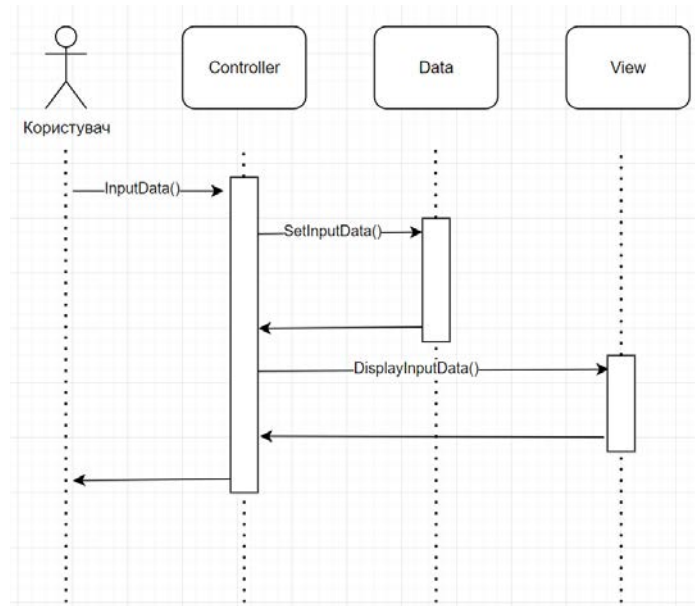


Рисунок 3.6 – Діаграма послідовностей для варіанту використання «Введення параметрів регресійної моделі»

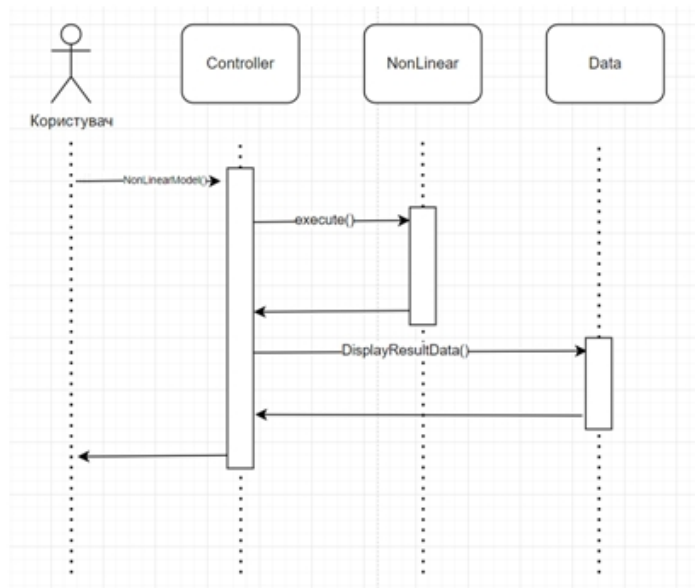


Рисунок 3.7 – Діаграма послідовностей для варіанту використання «Оцінювання трудомісткості розробки»

Після розробки технічного проекту можна приступати до наступного етапу проектування - розробки робочого проекту.

3.4 Розробка робочого проекту програми для оцінювання тривалості розробки ПЗ інтернет-магазинів

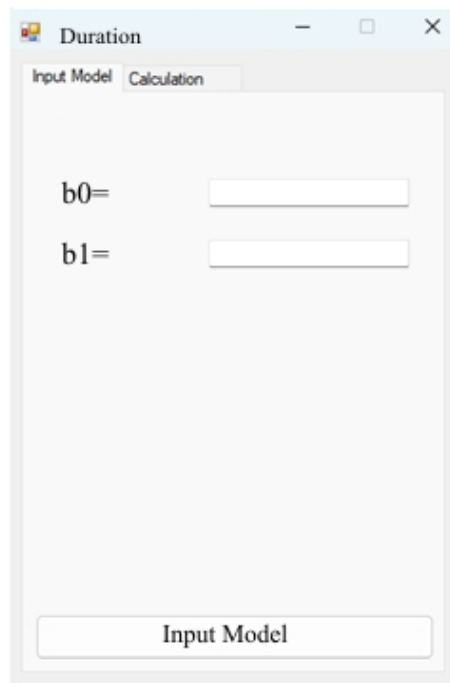
При розробці робочого проекту було виконано вибір мови та засобів розробки, кодування, тестування та випробування програми для оцінювання тривалості розробки ПЗ інтернет-магазинів, а також розробка усієї необхідної програмної документації.

Програма була розроблена мовою програмування C# з використанням середовища розробки MS Visual Studio. Код програми наведений у додатку Б.

Тестування розробленої програми було проведено методом "чорної скриньки". Для перевірки було використано дані з таблиці 1.1.

Випробування програми для оцінювання тривалості розробки ПЗ інтернет-магазинів було виконано згідно м програмою і методикою випробування, наведеною у Додатку Д.

Зовнішній вигляд та результати випробування наведені на рисунках 3.8 - 3.11.



Duration

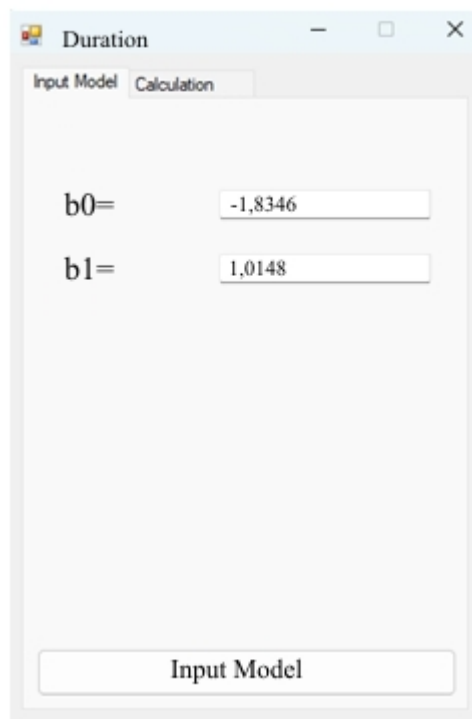
Input Model Calculation

b0=

b1=

Input Model

Рисунок 3.8 – Основне вікно програми для оцінювання тривалості розробки ПЗ інтернет-магазинів



Duration

Input Model Calculation

b0=

b1=

Input Model

Рисунок 3.9 – Ввод даних про параметри регресійної моделі



Duration

Input Model: Calculation

Input for calculation

Project:

X=:

α =:

Result

Duration:

Calculate

Save

Рисунок 3.10 – Зовнішній вигляд вкладки "Calculation"



Duration

Input Model: Calculation

Input for calculation

Project:

X=:

α =:

Result

Duration:

Calculate

Save

Рисунок 3.11 – Розрахунок тривалості розробки

Випробування показало повну працездатність розробленої програми для оцінювання тривалості розробки ПЗ інтернет-магазинів.

Також були розроблені: Опис програми, наведений у Додатку В та Інструкція користувача, наведена у Додатку Г.

4 РОЗРАХУНОК ЕКОНОМІЧНОЇ ЕФЕКТИВНОСТІ ВІД РОЗРОБКИ І ВПРОВАДЖЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ПЗ ІНТЕРНЕТ-МАГАЗИНІВ

Дана кваліфікаційна робота присвячена розробці програми для оцінювання тривалості розробки ПЗ інтернет-магазинів.

Створення програмного продукту вимагає одноразових витрат на розробку, придбання необхідних технічних засобів і поточних витрат на функціонування продукту. Економія від функціонування ПЗ визначається з урахуванням витрат на його експлуатацію. Відношення цієї економії до витрат на створення програмного продукту характеризує економічну ефективність капітальних вкладень. Економічні показники визначаються за діючими на момент розрахунку оптовими цінами, тарифами і ставками заробітної плати.

У даному підрозділі розраховується собівартість розробки програмного продукту, а також розрахунки економічної ефективності та строку окупності.

4.1 Розрахунок витрат на створення й експлуатацію ПЗ

Витрати на розробку ПЗ складаються з витрат на заробітну платню розробника; на амортизацію комп'ютера, на якому виконується розробка; на експлуатацію комп'ютера; на засоби розробки; на матеріали і комплектуючі.

Розробка програмного продукту виконується програмістом, місячна заробітна плата якого складає 30000 грн. Вартість комп'ютера складає 40000 грн. Витрати на допоміжні матеріали наведені в таблиці 4.1.

Таблиця 4.1 – Витрати на допоміжні матеріали

№	Пункти витрат	Сума, грн.
1	Доступ до мережі Internet	350
2	Папір А4	180
3	Заправка картриджа до принтера	180
4	Непередбачувані витрати	300
	Разом	1010

Вартість розробки ПЗ знаходиться за формулою:

$$K_{\text{пр}} = (B_{\text{зп}} + B_{\text{есв}} + B_{\text{зг}} + B_{\text{е}}) * T + B_{\text{дм}}, \quad (4.1)$$

де: $B_{\text{зп}}$ – витрати на оплату праці, встановлені зарплатнею працівника, грн.,

$B_{\text{есв}}$ – єдиний соціальний внесок, що складає 38% від витрат на заробітну плату, грн.,

$B_{\text{зг}}$ – загальногосподарські витрати, що складають 10% від витрат на оплату праці без урахування податкового навантаження, грн.;

$B_{\text{е}}$ – витрати за електроенергію, грн.;

T – тривалість розробки, міс.;

$B_{\text{дм}}$ – витрати на допоміжні матеріали, грн.

Розрахунок вартості розробки ПЗ виконується при вартості кіловат-години електроенергії 4,32 грн. За умови споживання потужності 0,75 кВт та тривалості розробки на місяць, а саме $22 \times 8 = 176$ годин розрахуємо значення $З_{\text{е}}$:

$$B_e = 176 \cdot 0,75 \cdot 4,32 = 570,24 \text{ грн.}$$

Витрати на розробку ПЗ наведені в таблиці 4.2.

Таблиця 4.2 – Витрати на розробку ПЗ

№	Найменування витрат	Одиниця	Кількість
1	Тривалість розробки	міс.	1
2	Основна і додаткова заробітна плата	грн.	36000
3	Відрахування в соціальні фонди	грн.	13680
4	Загальногосподарські витрати	грн.	3600
5	Витрати на допоміжні матеріали	грн.	1010
6	Витрати на електроенергію	грн.	570,24

Оскільки за методом експертної оцінки на основі розробки аналогічних систем визначено, що загальна тривалість розробки T складає 1 місяць, а кількість необхідних спеціалістів дорівнює 1 людині, то вартість розробки системи розраховуємо за формулою (4.1) і вона складає:

$$K_{\text{пр}} = (36000 + 13680 + 3600 + 570,24) \cdot 1 + 1010 = 54860,24 \text{ грн.}$$

Амортизаційні відрахування на устаткування складають 60% балансової вартості в рік:

$$A_{\text{уст}} = 40000 \cdot 0,6 = 24000 \text{ грн.}$$

У масштабах підприємства річні витрати на основні і допоміжні матеріали визначаються у розмірі 5% вартості основного устаткування:

$$B_m = 40000 \cdot 0,05 = 2000 \text{ грн.}$$

Оскільки в середньому устаткування споживає 0,2 кВт електроенергії і навантажується близько 6,5 годин на день, а в році маємо 256 робочих днів, можемо розрахувати річне споживання електроенергії устаткуванням:

$$B_e = 256 \cdot 6,5 \cdot 4,32 \cdot 0,2 = 1437,70 \text{ грн.}$$

Експлуатаційні витрати на устаткування на рік складуть:

$$B_{уст} = 24000 + 2000 + 1437,70 = 27437,70 \text{ грн.}$$

Отже, у перший рік витрати на створення й експлуатацію програмного продукту складуть:

$$B_p = 54860,24 + 27437,70 = 82297,94 \text{ грн.}$$

4.2 Економічна ефективність розробки та впровадження ПЗ

Основним показником економічної ефективності системи є підвищення ефективності керування інформацією у вигляді зниження витрат на керування при одночасному збільшенні швидкості і якості одержання потрібного результату.

Крім багатьох інших негативних ефектів, ручна обробка інформації спричиняє наступні негативні економічні ефекти: високі витрати на складування паперових документів (сейфи, шафи, папки й ін.); підвищені

витрати на канцтовари; витрати, пов'язані з роботою виявлення раніше допущених помилок

До числа основних факторів, що визначають приріст прибутку в зв'язку з упровадженням ПЗ, відносяться: підвищення продуктивності праці; вивільнення робочого часу.

Крім того, не піддається прямій грошовій оцінці підвищення оперативності керування, якості одержуваних результатів, поліпшення організації праці і т.д. Обов'язковою умовою визначення економічної ефективності ПЗ є порівнянність усіх показників у часі, за цінами й іншими нормами, використання для визначення показників, за змістом і колом елементів витрат.

Визначення прямої економічної ефективності ґрунтується на тому, що впровадження ПЗ вивільняє 1 працівника (за експертною оцінкою фахівців підприємства).

Зарплата 1 працівника в рік складає:

$$(30000,00 \cdot 12) \cdot 1 = 360000,00 \text{ грн.}$$

Річний економічний ефект розраховується за наступною формулою:

$$E_{\text{год}} = \Delta C_n - E_n \cdot k, \quad (4.2)$$

де ΔC_n – вивільнені кошти після впровадження ПЗ (360000,00 грн) мінус експлуатаційні витрати (27437,70) = 332562,30 грн.;

E_n – коефіцієнт ефективності (дорівнює коефіцієнту амортизації (0,6));

k – одноразові витрати на впровадження продукту (54860,24).

$$E_{\text{год}} = 332562,30 - 0,6 \cdot 54860,24 = 299646,16 \text{ грн.}$$

Строк окупності системи розраховується по формулі:

$$T = k/\Delta C_n = 54860,24/332562,30 \approx 0,16 \text{ (року)}$$

Отже, строк окупності ПЗ складає приблизно 2 місяці.

5 ОХОРОНА ПРАЦІ

5.1 Загальні питання охорони праці

Охорона праці – це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності [25].

Метою політики охорони праці є зведення до мінімуму показників виробничого травматизму та професійних захворювань. Ця мета набула нових форм у ЄС протягом останніх років і поширилася сьогодні до пропаганди "добробуту на роботі", що означає моральний, фізичний та соціальний добробут, а не лише відсутність нещасних випадків та професійних захворювань.

Крім того, необхідно також досягти низки допоміжних цілей:

- профілактика соціальних ризиків (стресів, домагань на робочому місці, депресій та роздратування, а також ризиків, які пов'язані з алкогольною, наркотичною залежністю);
- аналіз ризиків, пов'язаних із роботою, а також ергономічні, психологічні та соціальні ризики;
- урахування змін у формах зайнятості, організації роботи та робочого часу працівників з нестандартною та тимчасовою зайнятістю;
- урахування розмірів підприємства (конкретні заходи щодо інформування, підвищення рівня обізнаності, програм попередження ризиків на малих та середніх підприємствах, приватних підприємців, домашньої обслуги тощо);
- інтенсивна профілактика професійних захворювань (спричинених

азбестом, втрата слуху, проблеми опорно-рухового апарату);

- урахування демографічних змін.

Основними завданнями управління охороною праці є:

1) Опрацювання заходів щодо здійснення державної політики з охорони праці на регіональному та галузевому рівнях;

2) підготовка, прийняття та реалізація заходів, спрямованих на забезпечення:

- належних, безпечних і здорових умов праці;
- утримання в належному стані виробничого устаткування, будівель і споруд, інженерних мереж, безпечного ведення технологічних процесів;
- необхідних засобів індивідуального захисту для працівників;
- організації і проведення навчання працівників з питань охорони праці;
- пропаганди охорони праці;
- обліку, аналізу та оцінки стану умов і безпеки праці;
- професійного добору працівників окремих спеціальностей;
- страхування працівників від нещасного випадку на виробництві та профзахворювань;
- організаційно-методичне керівництво на регіональному та галузевому рівнях;
- стимулювання інтеграції управління охороною праці в єдину систему загального управління організацією виробництва;
- широке впровадження позитивного досвіду у сфері охорони праці.

За умови економічної, екологічної та демографічної кризи в Україні, склалася надзвичайна ситуація з безпекою та умовами праці на більшості підприємств, особливо середнього і малого бізнесу.

5.2 Аналіз шкідливих та небезпечних факторів при роботі з комп'ютером

До фізичних шкідливих і небезпечних чинників відносяться:

- підвищені рівні електромагнітного, рентгенівського, ультрафіолетового і інфрачервоного випромінювання;
- підвищений рівень статичної електрики і запилене повітря робочої зони;
- підвищений вміст позитивних іонів і понижений вміст негативних іонів в повітрі робочої зони;
- підвищений рівень яскравості;
- нерівномірність розподілу яскравості в полі зору;
- підвищене значення напруги в електричному ланцюзі, замикання якого може статися через тіло людини.

Хімічні шкідливі і небезпечні чинники наступні: підвищений вміст в повітрі робочої зони двоокису вуглецю, озону, аміаку, фенолу і формальдегіду.

Психофізіологічні шкідливі і небезпечні фактори:

- напруга зору і уваги;
- інтелектуальні, емоційні і тривалі статичні навантаження;
- монотонність праці;
- великий обсяг інформації, що обробляється в одиницю часу;
- нераціональна організація робочого місця.

Типовими відчуттями, які відчувають до кінця робочого дня оператори за комп'ютерами, є: перевтома очей, головний біль, тягучий біль в м'язах шиї, рук і спини, зниження концентрації уваги.

Уже в перші роки комп'ютеризації було відзначено специфічне зорове стомлення у користувачів дисплеїв, що отримало загальну назву «комп'ютерний зоровий синдром». Однією з причин є те, що сформована за мільйони років

еволюції зорова система людини пристосована для сприйняття об'єктів у відбитому світлі (друковані тексти, малюнки тощо), а не для роботи за дисплеєм. Зображення на дисплеї принципово відрізняється від звичних оку об'єктів спостереження – воно світиться, мерехтить, складається з дискретних точок, а кольорове комп'ютерне зображення не відповідає природним кольорам. Але не тільки особливості зображення на екрані викликають зорове стомлення.

Велике навантаження орган зору відчуває при введенні інформації, так як користувач змушений часто переводити погляд з екрану на текст і клавіатуру, що знаходяться на різній відстані і по-різному освітлені. Зорове стомлення проявляється скаргами на затуманення зору, труднощі при перенесенні погляду з ближніх предметів на дальні і з далеких на ближні, що здаються зміни забарвлення предметів, їх двоїння, відчуття печіння, «піску» в очах, почервоніння повік, болі при руханні очима.

Тривала і інтенсивна робота на комп'ютері може стати джерелом важких професійних захворювань, таких, як травма повторюваних навантажень (ТПН), що представляє собою поступове накопичення нездужань, котрі переходять у захворювання нервів, м'язів і сухожиль руки.

До професійних захворювань, пов'язаних з ТПН, відносяться:

- тендовагініт – запалення сухожиль кисті, зап'ястя, плеча;
- тендосиновіт – запалення синовіальної оболонки сухожильного основи кисті і зап'ястя;
- синдром зап'ястного каналу (СЗК) – викликається утиском серединного нерва в зап'ястному каналі. Травма викликає утворення продуктів розпаду в області зап'ястного каналу, в результаті чого спочатку виникає набряк, а потім СЗК.

З'являються скарги на пекучий біль і поколювання в зап'ясті, долоні, а також пальцях, крім мізинця. Спостерігається хворобливість і оніміння, ослаблення м'язів, що забезпечують рух великого пальця.

Ці захворювання зазвичай настають в результаті безперервної роботи на неправильно організованому робочому місці.

Механізм порушень, що відбуваються в організмі під впливом електромагнітних полів, обумовлений їх специфічною (нетепловою) і тепловою дією.

5.3 Розробка заходів щодо зменшення впливу шкідливих та небезпечних факторів при роботі з комп'ютером

Після проведення аналізу робочого місця інженера-програміста було з'ясовано, що воно відповідає встановленим вимогам. Але, в результаті аналізу були виявлені порушення в організації безпосередньо самого робочого місця програміста. У зв'язку з цим було запропоновано організувати робоче місце програміста наступним способом:

- висота над рівнем підлоги робочої поверхні, на якій працює програміст, повинна складати 720 мм. Бажано, щоб робочий стіл при необхідності можна було регулювати по висоті в межах 680 – 780 мм;
- оптимальний розмір поверхні столу 1600 x 1000 мм². Під столом повинен бути простір для ніг з розмірами по глибині 650 мм. Робочий стіл оператора повинен також мати підставку для ніг, розташовану під кутом 15 град. до поверхні столу. Довжина підставки – 400 мм, ширина – 350 мм. Відстань клавіатури від краю столу повинна бути не більш 300 мм, що забезпечить програмісту зручну опору для передпліч. Відстань між очима й екраном монітору повинне складати 40 – 80 см;
- робочий стілець інженера-програміста повинен бути оснащений підйомно-поворотним механізмом. Висота сидіння повинна регулюватися в межах 400 – 500 мм. Глибина сидіння повинна складати не менш 380 мм, а

ширина – не менш 400 мм. Висота опорної поверхні спинки не менш 300 мм, ширина – не менш 380 мм. Кут нахилу спинки стільця до площини сидіння повинен змінюватися в межах 90 – 110°.

Виходячи з результатів аналізу важкості та напруженості праці було запропоновано скоротити час роботи за комп'ютером, робити перерви сумарний час яких повинен складати 50 хвилин при 8-ми годинній зміні.

Під час аналізу освітлення на робочому місці інженера-програміста було встановлено, що воно відповідає встановленим нормам. Для підтримки запроектованого освітлення у чистому виді необхідно скласти графік, де передбачити очищення віконних блоків і світильників не менше 2 разів на рік.

Як міри по зниженню шуму можна запропонувати:

- облицювання стелі і стін звукопоглинаючим матеріалом (знижують шум на 6 – 8 дБ);
- екранування робочого місця (постановкою перегородок, діафрагм);
- установка в комп'ютерних приміщеннях устаткування, що робить мінімальний шум;
- раціональне планування приміщення.

Також для зменшення шуму було запропоновано використовувати замість струменевого принтера, що створює багато шуму, більш тихий – лазерний принтер.

Електробезпеку у приміщенні лабораторії було запропоновано забезпечити наступними технічними способами і засобами захисту:

- для зменшення накопичення статичної електрики застосовувати зволожувачі і нейтралізатори, антистатичне покриття підлоги;
- забезпечити приєднання металевих корпусів устаткування до жили, що заземлює. Заземлення корпуса ПК забезпечити підведенням жили, що заземлює, до розеток. Опір заземлення 4 Ом, згідно (ПУЕ) для електроустановок з напругою до 1000 В.;

- для захисту приладів від перенавантажень та коротких замикань використовується запобіжник.

А також організаційними заходами:

- своєчасне проведення інструктажів з охорони праці;
- заборона на використання непередбачених електричних приладів, таких як електричні чайники, обігрівачі.

Також важливою складовою охорони праці є пожежна безпека. Пожежа може привести до дуже несприятливих наслідків (втрата кошовної інформації, псування майна, загибель людей і т.д.), тому необхідно: виявити й усунути всі причини виникнення пожежі; розробити план заходів для ліквідації пожежі в будинку.

Причинами виникнення пожежі можуть бути:

- несправності електропроводки, розеток і вимикачів які можуть привести до короткого замикання або пробією ізоляції;
- використання ушкоджених (несправних) електроприладів;
- використання в приміщенні електронагрівальних приладів з відкритими нагрівальними елементами;
- виникнення пожежі внаслідок влучення блискавки в будинок;
- загоряння будинку внаслідок зовнішніх впливів;
- неакуратне поводження з вогнем і недотримання мір пожежної безпеки.

Для профілактики пожежі надзвичайно важлива правильна оцінка пожежонебезпеки будинку, визначення небезпечних факторів і обґрунтування способів і засобів попередження пожеж і захисту.

Одне з умов забезпечення безпеки від пожежі – ліквідація можливих джерел запалення.

Джерелами запалення можуть бути:

- несправне електроустаткування, несправності в електропроводці,

електричних розетках і вимикачах. Для виключення виникнення пожежі з цих причин необхідно вчасно виявляти й усувати несправності, проводити плановий огляд і вчасно усувати всі несправності;

- несправні електроприлади. Необхідні міри для виключення пожежі містять у собі своєчасний ремонт електроприладів, якісне виправлення поломок, не використання несправних електроприладів;

- обігрівання приміщення електронагрівальними приладами з відкритими нагрівальними елементами. Відкриті нагрівальні поверхні можуть спричинити пожежу, тому що в приміщенні знаходяться паперові документи і довідкова література у виді книг, посібників, а папір – легкозаймистий предмет. З метою профілактики пожежі пропоную не використовувати відкриті обігрівальні прилади в приміщенні лабораторії;

- коротке замикання в електропроводці. З метою зменшення імовірності виникнення пожежі внаслідок короткого замикання необхідно, щоб електропроводка була схованою;

- влучення в будинок блискавки. У літній період під час грози можливе влучення блискавки внаслідок чого можливий пожежа. Щоб уникнути цього я рекомендую установити на даху будинку блискавковідвід;

- недотримання мір пожежної безпеки і паління в приміщенні також може спричинити пожежу. Для усунення загоряння в результаті паління в приміщенні лабораторії пропоную категорично заборонити паління, а дозволити тільки в строго відведеному для цього місці.

Таким чином був здійснений аналіз небезпечних і шкідливих факторів, що впливають на інженера-програміста при розробці відповідного ПЗ. Були визначені умови, що визначають оптимальну організацію робочого місця, дозволяють зберегти хорошу працездатність протягом всього робочого дня, підвищать як в кількісному, так і в якісному продуктивність праці інженера-програміста.

6 ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА

6.1 Загальні питання охорони навколишнього середовища

Охорона навколишнього природного середовища, раціональне використання природних ресурсів, забезпечення екологічної безпеки життєдіяльності людини – невід'ємна умова сталого економічного та соціального розвитку України.

Відносини у галузі охорони навколишнього природного середовища в Україні регулюються законом України "Про охорону навколишнього природного середовища", а також розроблюваними відповідно до нього земельним, водним, лісовим законодавством, законодавством про надра, про охорону атмосферного повітря, про охорону і використання рослинного і тваринного світу та іншим спеціальним законодавством (стаття 2 Закону).

Основними принципами охорони навколишнього природного середовища є (стаття 3 Закону):

- пріоритетність вимог екологічної безпеки, обов'язковість додержання екологічних стандартів, нормативів та лімітів використання природних ресурсів при здійсненні господарської, управлінської та іншої діяльності;
- гарантування екологічно безпечного середовища для життя і здоров'я людей;
- запобіжний характер заходів щодо охорони навколишнього природного середовища;
- екологізація матеріального виробництва на основі комплексності рішень у питаннях охорони навколишнього природного середовища, використання та відтворення відновлюваних природних ресурсів, широкого

впровадження новітніх технологій;

- обов'язковість екологічної експертизи;
- гласність і демократизм при прийнятті рішень, реалізація яких впливає на стан навколишнього природного середовища, формування у населення екологічного світогляду;
- науково обґрунтоване нормування впливу господарської та іншої діяльності на навколишнє природне середовище;
- компенсація шкоди, заподіяної порушенням законодавства про охорону навколишнього природного середовища;
- встановлення екологічного податку, збору за спеціальне використання води, збору за спеціальне використання лісових ресурсів, плати за користування надрами відповідно до Податкового кодексу України.

У процесі виробничої діяльності підприємства людей екологічно убезпечують, раціонально використовують природні ресурси, дотримують нормативів шкідливих впливів на навколишнє природне середовище відповідно до вимог природоохоронного законодавства України.

На підприємствах діють такі заходи з охорони навколишнього природного середовища:

- розроблено стандарт якості з екологічної безпеки підприємства відповідно до основних положень міжнародних стандартів серії ISO 14000 щодо керування якістю навколишнього середовища;
- здійснюють взаємодію з організаціями Міністерства екології та природних ресурсів, Міністерства охорони здоров'я, прокуратури, державних організацій з питань охорони навколишнього природного середовища;
- оформлено необхідні дозвільні документи (дозволи на викиди шкідливих речовин, інвентаризацію викидів зі стаціонарних джерел, інвентаризацію промислових і побутових відходів підприємства, паспорти й реєстраційні картки на кожен вид відходу);

- здійснюють контроль за обсягами і складом забруднювальних речовин, що викидаються в атмосферу, та рівнями енергетичних викидів (шуму, вібрації, теплового й електромагнітного проміння) та їх постійний облік. Забезпечено дотримання санітарно-гігієнічних норм щодо зазначених впливів;
- використовують організовані джерела викидів (з газоочисними пристроями, якщо є потреба), які забезпечують допустимий рівень впливу на навколишнє середовище;
- здійснюють платежі за забруднення навколишнього природного середовища;
- організовано облік, збір і безпечне зберігання промислових відходів у спеціально відведених і обладнаних місцях;
- визначено склад і властивості утворюваних відходів, а також ступінь їх небезпеки для навколишнього природного середовища та здоров'я людини;
- організовано передачу промислових відходів іншим підприємствам за договорами (актами) з мінімальним розміщенням відходів у навколишньому природному середовищі [26].

6.2 Забруднення навколишнього середовища

Охорона навколишнього середовища являється формою відносин між суспільством і природою. На даний час мають місце різні засоби, за допомогою яких вона може здійснюватись. А саме, серед них: правові, біологічні, економічні, гігієнічні, науково-технічні тощо.

У процесі охорони навколишнього середовища вирішуються дві головні проблеми – це забезпечення чистоти, її підтримання та організація раціонального природокористування.

Будь-яка діяльність, в тому числі економічна, забруднює природне середовище. В результаті отримуємо дефіцит та забрудненні повітряні ресурси, території, вода.

За порушення у сфері охорони природи, незаконне використання природних ресурсів законодавством України передбачена адміністративна (статті 52-92 Кодексу про адміністративні правопорушення) та кримінальна (статті 236-254 Кримінального кодексу) відповідальність.

Утилізація – будь-які технологічні операції, пов'язані зі зміною фізичних, хімічних або біологічних властивостей відходів, з метою підготовки їх до екологічно безпечного зберігання, перевезення, утилізації чи видалення.

Установи, що займаються розробкою програмних продуктів використовують у своїй діяльності персональні комп'ютери, ноутбуки, монітори, клавіатури, маніпулятори «миша», техніка для створення копій, інші електронні компоненти, а також різноманітне канцелярське приладдя (папер, тонер для принтерів, батарейки і т.п.). Все це потрібно правильно утилізувати, щоб мінімізувати шкоду для навколишнього середовища.

6.3 Розробка заходів щодо зменшення забруднення навколишнього середовища

Оскільки комп'ютерна техніка має свій час, через який вона виходить зі строю, постає питання про правильну утилізацію даної техніки. Вона потребує утилізації, адже у її складі знаходиться майже вся періодична таблиця хімічних елементів. Також окрім металів, вона має у своєму складі матеріали на основі полівінілхлориду. Вони не представляють загрози в процесі роботи, однак картина змінюється коли закінчується життєвий цикл приладу і він потрапляє на звалище.

Також слід зазначити, що утилізація комп'ютерів – це обов'язкова процедура для підприємств і організацій різної форми власності. Неналежне виконання цієї процедури призводить до податкової та адміністративної відповідальності.

Оскільки на метали що містяться в електронних компонентах впливає волога, то вони переходять в розчинні сполуки, що являють собою сильні отрути. Нагальною екологічною проблемою є і утилізація пластиків, які містять в собі хлорні сполуки, а також ароматичні вуглеводні.

Акумуляторні батареї, що були використані, збирають для утилізації у спеціально-розроблений контейнер. Після того як контейнер буде наповнений він здається до відповідних установ, що спеціалізуються на переробці акумуляторних батарей. Акумуляторні батареї не можна викидати на звичайні звалища, адже вони містять у своєму складі велику кількість токсичних складових, наприклад такі як літій, марганець, свинець, цинк. Якщо їх викинути на звичайне звалище – то токсичні складові почнуть отруювати ґрунт та воду, а як наслідок отруювати навколишню флору і фаун, в тому числі людський організм.

Використовуючи комплексний підхід до проблеми утилізації оргтехніки кількість відходів, що не переробляються зводиться до мінімуму. Також рідкісні метали, що містяться у складі комп'ютерної техніки (чорні і кольорові метали, люмінофор) вилучаються та здаються до Державного фонду чи залучаються до повторного використання.

Отже, як висновок, слід зазначити, що виконувати утилізацію комп'ютерної техніки потрібно як з економічних міркувань так і з екологічних.

ВИСНОВКИ

В результаті виконання кваліфікаційної роботи було вирішено актуальне завдання - підвищення достовірності оцінювання тривалості розробки ПЗ інтернет-магазинів.

Удосконалено однофакторну нелінійну регресійну модель для оцінювання тривалості розробки ПЗ інтернет-магазинів з використанням нормалізуючого перетворення у вигляді десяткового логарифму та врахуванням викидів у даних, що дозволило підвищити достовірність оцінювання тривалості розробки ПЗ інтернет-магазинів в порівнянні з існуючими моделями COCOMO та ISBSG.

В результаті роботи було вирішено такі завдання:

- проаналізовано особливості розробки ПЗ інтернет-магазинів;
- проаналізовано існуючі методи та моделі для оцінювання тривалості розробки ПЗ;
- обґрунтовано актуальність подальшого дослідження;
- розглянуто методи побудови нелінійних регресійних моделей для оцінювання тривалості розробки ПЗ;
- виконано попередню обробку даних для побудови нелінійної регресійної моделі для оцінювання тривалості розробки ПЗ інтернет-магазинів;
- побудовано однофакторну нелінійну регресійну модель для оцінювання тривалості розробки ПЗ інтернет-магазинів;
- оцінено якість побудованої однофакторної нелінійної регресійної моделі;
- порівняно побудовану однофакторну нелінійну регресійну модель з існуючими моделями COCOMO та ISBSG;

– розроблено програму для оцінювання тривалості розробки ПЗ інтернет-магазинів, використовуючи побудовану однофакторну нелінійну регресійну модель.

В кваліфікаційній роботі також були виконані спеціальні розділи з розрахунку економічної ефективності від розробки і впровадження ПЗ для оцінювання тривалості розробки ПЗ інтернет-магазинів, охорони праці, охорони навколишнього середовища.

Мета роботи досягнута, усі завдання виконані в повному обсязі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Кращий движок для інтернет-магазину. Огляд кращих платформ для інтернет-магазину. URL: <https://tuthost.ua/uk/blog/strong-luchshij-dvizhok-dlya-internet-magazina-obzor-luchshih-dvizhkov-dlya-internet-magazina-strong/> (дата звернення: 26.09.2024)
2. Як створити і відкрити свій інтернет-магазин. URL: <https://tuthost.ua/uk/blog/yak-stvoriti-i-vidkriti-svij-internet-magazin/> (дата звернення: 26.09.2024)
3. Boehm B.W. Software engineering economics. Englewood Cliffs, NJ: Prentice Hall, 1981. 768 p.
4. International Software Benchmarking Standards Group. URL: <https://www.isbsg.org/> (дата звернення: 26.09.2024).
5. Приходько С.Б., Пухалевич А.В. Розробка нелінійних регресійних моделей трудомісткості програмних проєктів на основі перетворення Джонсона. *Збірник наукових праць НУК*. Миколаїв: НУК, 2014. №2 (2014). С.76-80.
6. Макарова Л.М., Литвин М.В. Математична модель для оцінювання тривалості розробки інтернет-магазинів, створених за допомогою платформи Shopify. *Матеріали XIII-ої Міжнародної науково-практичної конференції «Free and Open Source Software»*. (Харків, 16-18 листопада 2021 р.). Харків: Харківський національний університет будівництва та архітектури, 2021. С.22-23.
7. Макарова Л.М., Каіров В.О., Поліщук І.О. Нелінійна регресійна модель для оцінювання тривалості розробки веб frontend-застосунків. *Матеріали VIII Міжнародної науково-технічної конференції "Комп'ютерне моделювання та оптимізація складних систем" (КМОСС-2023)* (1-3 листопада 2023 року, м. Дніпро, Україна). С. 57.

8. Ніколаєв А.К., Макарова Л.М. Математична модель для оцінювання тривалості розробки програмного забезпечення інтернет-магазинів. *Матеріали VII Всеукраїнської науково-практичної інтернет-конференції студентів, аспірантів та молодих вчених «Сучасні інформаційні системи та технології»* (29 листопада 2024 року, м. Херсон, м. Хмельницький). За ред. А.А. Григорової. Херсон: Книжкове видавництво ФОП Вишемирський В.С., 2024. С. 193-194.

9. Як відкрити інтернет-магазин: покрокова інструкція створення бізнесу з нуля. URL: <https://fondy.ua/uk/knowledge/online-store-creation/> (дата звернення: 26.09.2024)

10. Meet WordPress. URL: <https://wordpress.org/> (дата звернення: 26.09.2024)

11. WooCommerce is the platform that grows with you. URL: <https://woocommerce.com/> (дата звернення: 26.09.2024)

12. The best FREE and open-source eCommerce platform. URL: <https://www.opencart.com/> (дата звернення: 26.09.2024)

13. Adobe commerce (Magento). URL: <https://business.adobe.com/products/magento/magento-commerce.html> (дата звернення: 26.09.2024)

14. Готовий інтернет-магазин з сучасною CMS. URL: <https://mercurio-cms.com/> (дата звернення: 26.09.2024)

15. The one commerce platform behind it all. URL: <https://www.shopify.com/> (дата звернення: 26.09.2024)

16. Створення інтернет-магазинів. URL: <https://horoshop.ua/ua/> (дата звернення: 26.09.2024)

17. Пухалевич А.В. Моделі та інформаційна технологія переробки інформації для оцінювання тривалості проектів з розробки програмного забезпечення : дис. ... к-та тех. наук : 05.13.06 / НУК ім. адм. Макарова. Миколаїв, 2017. 179 с.

18. Коефіцієнт детермінації. URL: https://studwood.net/2219196/ekonomika/koeffitsiyent_determinatsiyi_poznachayetsya_--_kvadrat_--_statistichniy_pokaznik_vikoristovuyetsya_statistichnih (дата звернення: 01.10.2024).
19. Приходько С.Б., Макарова Л.М., Приходько Н.В., Пухалевич А.В. Методичні вказівки та завдання до виконання лабораторних робіт з дисципліни "Емпіричні методи програмної інженерії". Миколаїв: НУК, 2023. 56 с.
20. Mardia K.V. Measures of multivariate skewness and kurtosis with applications. *Biometrika*. 57. 1970. P. 519–530. DOI: 10.1093/biomet/57.3.519
21. Aggarwal C.C. Outlier analysis. 2nd ed. New York : Springer International Publishing AG, 2017. 481 p.
22. Prykhodko, S., Prykhodko, N., Makarova, L., Pukhalevych, A. Application of the Squared Mahalanobis Distance for Detecting Outliers in Multivariate NonGaussian Data. *Proceedings of 14th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET)* (Lviv-Slavske, Ukraine, February 20-24, 2018). P.962-965. DOI: 10.1109/TCSET.2018.8336353
23. Prykhodko S., Prykhodko N., Makarova L., Pukhalevych A. Outlier Detection in NonLinear Regression Analysis Based on the Normalizing Transformations. *Proceedings of the 15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET)*, IEEE. (Lviv-Slavske, 2020). P.407-410.
24. Prykhodko N.V., Prykhodko S.B. Constructing the Nonlinear Regression Models on the Basis of Multivariate Normalizing Transformations. *Electronic Modeling*. 2018. V.40. №6. P.99-108.
25. Про охорону праці: Закон України від 15.11.2024 № 2694-XII. URL: <https://zakon.rada.gov.ua/laws/show/2694-12#Text> (дата звернення: 10.11.2024 р.).

26. Про охорону навколишнього природного середовища: Закон України від 15.11.2024 № 1264-XII. URL: <https://zakon.rada.gov.ua/laws/show/1264-12#Text> (дата звернення: 12.11.2024 р.).

ДОДАТОК А – ТЕХНІЧНЕ ЗАВДАННЯ НА РОЗРОБКУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ПЗ ІНТЕРНЕТ-МАГАЗИНІВ

Вступ

Повна назва розроблюваного проекту: Програма для оцінювання тривалості розробки ПЗ інтернет-магазинів.

Коротка назва: програма.

Галузь застосування: програма призначена для застосування в компаніях та командах розробників, що займаються розробки ПЗ інтернет-магазинів.

1 Підстави для розробки

Підставою для розробки є завдання на кваліфікаційну роботу, видане кафедрою ПЗАС НУК ім. адмірала Макарова.

2 Призначення розробки

2.1 Функціональне призначення програми

Дана програма призначена для оцінювання тривалості розробки ПЗ інтернет-магазинів.

2.2 Експлуатаційне призначення програми

Програма може використовуватися для оцінювання тривалості розробки ПЗ інтернет-магазинів в компаніях та командах розробників, що займаються розробкою відповідного ПЗ.

3 Вимоги до програмного забезпечення

3.1 Вимоги до функціональних характеристик

3.1.1 Вимоги до складу виконуваних функцій

Програма повинна забезпечувати можливість виконання наведених нижче функцій:

- Введення даних про параметри регресійної моделі (значення коефіцієнтів рівняння регресії).
- Введення значення довірчої ймовірності.
- Введення назви проєкту розробки інтернет-магазину та значення трудомісткості його розробки для подальшого розрахунку.
- Оцінювання трудомісткості розробки інтернет-магазину.
- Збереження результатів оцінювання трудомісткості розробки інтернет-магазину в базу даних.

3.1.2 Вимоги до організації вхідних та вихідних даних

Вимоги до організації вхідних даних

Вхідні дані вводяться у відповідні поля форми відповідно до допустимих типів (числові або текстові).

До вхідних даних належать: параметри регресійної моделі b_0 , b_1 , значення довірчої ймовірності, назва проєкту, значення трудомісткості розробки.

Вимоги до організації вихідних даних

Вихідні дані виводяться у відповідні поля форми та зберігаються в базі даних.

Вихідними даними є прогнозована тривалість розробки ПЗ інтернет-магазинів.

3.2 Вимоги до надійності

Вимоги до забезпечення надійного функціонування ПЗ має бути забезпеченим виконанням сукупності організаційно-технічних заходів, перелік яких наведений нижче:

- організацією безперебійного живлення ПК або сервера;
- безперебійним каналом зв'язку.

Час відновлення після відмови

Час відновлення після відмови, викликаній проблемами з електроживленням або каналом зв'язку, не фатальним збоєм (не крахом) операційної системи, не перевищує 3 хвилин, які необхідні для перезавантаження ПК або відновлення каналу зв'язку.

3.3 Вимоги до умов експлуатації

3.3.1 Кліматичні умови експлуатації

Кліматичні умови експлуатації, при яких повинні забезпечуватися задані характеристики, повинні задовольняти вимогам, пропонованим до технічних засобів у частині умов їхньої експлуатації.

3.3.2 Вимоги до видів обслуговування

Програмний продукт не вимагає проведення будь-яких видів обслуговування.

3.3.3 Вимоги до чисельності та кваліфікації персоналу

З програмою працює один користувач. Необхідний рівень підготовки користувачів: кінцевий користувач повинен мати практичні навички роботи з комп'ютером та розумітися на предметній області.

3.4 Вимоги до складу та параметрів технічних засобів

Для роботи з програмним продуктом необхідний наступний склад технічних засобів з мінімальними параметрами:

- CPU: Core i3;
- RAM: 1 Гб;
- HDD/SDD 500 Мб;
- пристрої вводу-виводу: монітор з екраном роздільної здатності не менше ніж 1600x900 пікселів, клавіатура, маніпулятор типу «миша».

3.5 Вимоги до інформаційної та програмної сумісності

Для використання даної програми потрібно мати операційну систему Windows 10 або вище.

3.6 Вимоги до маркування та пакування

Вимоги до маркування та пакування не пред'являються.

3.7 Вимоги до транспортування та зберігання

Вимоги до транспортування та зберігання не пред'являються.

4 Вимоги до програмної документації

Програмна документація повинна містити:

- технічне завдання;
- код програми;
- опис програми;
- інструкцію користувача;
- програму та методику випробувань ПЗ.

5 Техніко-економічні показники

Економічна ефективність впровадження ПЗ розрахована в розділі 4. Згідно з отриманими результатами впровадження даного ПЗ є доцільним.

6 Стадії та етапи розробки

Стадії та етапи розробки програми для оцінювання розміру веб-застосунків, що створюються мовою Python, представлені в таблиці А.1.

Таблиця А.1 – Стадії та етапи розробки ПЗ

Стадії розробки	Етапи розробки	Термін виконання
1. Технічне завдання	Розробка та затвердження технічного завдання	21.09.2024
2. Ескізний проект	Розробка ескізного проекту	18.10.2024
	Затвердження ескізного проекту	21.10.2024
3. Технічний проект	Розробка технічного проекту	03.11.2024
	Затвердження технічного проекту	06.11.2024
4. Робочий проект	Розробка програми	16.11.2024
	Розробка програмної документації	26.11.2024
	Випробування програми	01.12.2024

7 Порядок контролю та приймання

Контроль за аналізом та проектуванням кожної окремої частини ПЗ на кожному етапі з урахуванням вимог, визначених у технічному завданні. Кожна стадія розробки повинна бути представлена в зазначені строки та узгоджена із замовником.

Приймання проводяться відповідно до програми і методики випробувань і документують за допомогою протоколу проведення випробувань. У разі

знаходження помилок підчас прийому програмного виробу складається акт про знайдені помилки, який підписуються представниками замовника і розробника і затверджуються керівником організації – замовника та організації – розробника. Розробник повинен протягом не більше ніж двох тижнів виправити зазначені помилки і оповістити замовника про повторне проведення перевірки.

ДОДАТОК Б – КОД ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ПЗ ІНТЕРНЕТ-МАГАЗИНІВ

MainView.cs

```
using System;
using System.Windows.Forms;

namespace Metrics_Program
{
    public partial class MainView : Form
    {
        public Action OnInputModelButtonClick;
        public Action OnCalculateButtonClick;
        public Action OnSaveButtonClick;

        public MainView()
        {
            InitializeComponent();
        }

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

        public void DisplayResult(float resultDuration)
        {
            ResultLabel.Text = "Duration:" + resultDuration;
        }

        public void DisplayBValues(float b0, float b1)
        {
            B0Label.Text = "b0=" + b0;
            B1Label.Text = "b1=" + b1;
        }

        private void InputModelButtonClick(object sender, EventArgs
e)
        {
            OnInputModelButtonClick?.Invoke();
        }
    }
}
```

```

e) private void CalculateButtonClick(object sender, EventArgs
    {
        OnCalculateButtonClick?.Invoke();
    }

    private void SaveClick(object sender, EventArgs e)
    {
        OnSaveButtonClicked?.Invoke();
    }

namespace Metrics_Program
{
    partial class MainView
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        /// <param name="disposing">true if managed resources
should be disposed; otherwise, false.</param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Windows Form Designer generated code

        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        private void InitializeComponent()
        {
            this.InputModelButton = new
System.Windows.Forms.Button();
            this.CalculateButton = new
System.Windows.Forms.Button();
            this.InputGroup = new System.Windows.Forms.GroupBox();
            this.ResultGroup = new System.Windows.Forms.GroupBox();
            this.ResultLabel = new System.Windows.Forms.Label();

```

```

        this.FileIsReadyLabel = new
System.Windows.Forms.Label();
        this.tabControl1 = new
System.Windows.Forms.TabControl();
        this.ModelTab = new System.Windows.Forms.TabPage();
        this.B0Label = new System.Windows.Forms.Label();
        this.B1Label = new System.Windows.Forms.Label();
        this.ModelLabel = new System.Windows.Forms.Label();
        this.InputModel = new System.Windows.Forms.Button();
        this.Save = new System.Windows.Forms.Button();
        this.CalculationTab = new
System.Windows.Forms.TabPage();
        this.InputGroup.SuspendLayout();
        this.ResultGroup.SuspendLayout();
        this.tabControl1.SuspendLayout();
        this.ModelTab.SuspendLayout();
        this.CalculationTab.SuspendLayout();
        this.SuspendLayout();
        //
        // InputModelButton
        //
        this.InputModelButton.Location = new
System.Drawing.Point(6, 327);
        this.InputModelButton.Name = "InputModelButton";
        this.InputModelButton.Size = new
System.Drawing.Size(279, 32);
        this.InputModelButton.TabIndex = 0;
        this.InputModelButton.Text = "Input Model";
        this.InputModelButton.UseVisualStyleBackColor = true;
        this.InputModelButton.Click += new
System.EventHandler(this.InputModelButtonClick);
        //
        // CalculateButton
        //
        this.CalculateButton.Location = new
System.Drawing.Point(6, 365);
        this.CalculateButton.Name = "CalculateButton";
        this.CalculateButton.Size = new
System.Drawing.Size(279, 32);
        this.CalculateButton.TabIndex = 1;
        this.CalculateButton.Text = "Calculate";
        this.CalculateButton.UseVisualStyleBackColor = true;
        this.CalculateButton.Click += new
System.EventHandler(this.CalculateButtonClick);
        //
        // InputGroup
        //
        this.InputGroup.Location = new System.Drawing.Point(6,
6);

```

```

        this.InputGroup.Name = "InputGroup";
        this.InputGroup.Size = new System.Drawing.Size(279,
170);
        this.InputGroup.TabIndex = 4;
        this.InputGroup.TabStop = false;
        this.InputGroup.Text = "Input for calculation";
        //
        // Project
        //
        this.Project.AutoSize = true;
        this.Project.Font = new System.Drawing.Font("Microsoft
Sans Serif", 10.25F);
        this.Project.Location = new System.Drawing.Point(6,
95);
        this.Project.Name = "Project";
        this.Project.Size = new System.Drawing.Size(100, 17);
        this.Project.TabIndex = 7;
        this.Project.Text = "Project";
        //
        // Project_field
        //
        this.Project_field.Location = new
System.Drawing.Point(131, 92);
        this.Project_field.Name = "Project_field";
        this.Project_field.Size = new System.Drawing.Size(142,
20);
        this.Project_field.TabIndex = 6;
        //
        // X
        //
        this.X.AutoSize = true;
        this.X.Font = new System.Drawing.Font("Microsoft Sans
Serif", 10.25F);
        this.X.Location = new System.Drawing.Point(6, 50);
        this.X.Name = "X";
        this.X.Size = new System.Drawing.Size(112, 17);
        this.X.TabIndex = 5;
        this.X.Text = "X=";
        //
        // X_field
        //
        this.x_field.Location = new System.Drawing.Point(131,
50);
        this.X_field.Name = "X_field";
        this.X_field.Size = new System.Drawing.Size(142, 20);
        this.X_field.TabIndex = 4;
        //
        // ResultGroup
        //

```

```

        this.ResultGroup.Controls.Add(this.ResultLabel);
        this.ResultGroup.Location = new System.Drawing.Point(6,
182);
        this.ResultGroup.Name = "ResultGroup";
        this.ResultGroup.Size = new System.Drawing.Size(279,
111);
        this.ResultGroup.TabIndex = 8;
        this.ResultGroup.TabStop = false;
        this.ResultGroup.Text = "Result";
        //
        // ResultLabel
        //
        this.ResultLabel.AutoSize = true;
        this.ResultLabel.Font = new
System.Drawing.Font("Microsoft Sans Serif", 20.25F);
        this.ResultLabel.Location = new
System.Drawing.Point(17, 45);
        this.ResultLabel.Name = "ResultLabel";
        this.ResultLabel.Size = new System.Drawing.Size(177,
31);
        this.ResultLabel.TabIndex = 0;
        this.ResultLabel.Text = "Duration:";
        //
        // tabControl1
        //
        this.tabControl1.Controls.Add(this.ModelTab);
        this.tabControl1.Controls.Add(this.CalculationTab);
        this.tabControl1.Location = new
System.Drawing.Point(12, 9);
        this.tabControl1.Name = "tabControl1";
        this.tabControl1.SelectedIndex = 0;
        this.tabControl1.Size = new System.Drawing.Size(299,
429);
        this.tabControl1.TabIndex = 0;
        //
        // ModelTab
        //
        this.ModelTab.Controls.Add(this.B0Label);
        this.ModelTab.Controls.Add(this.B1Label);
        this.ModelTab.Controls.Add(this.InputModel1Label);
        this.ModelTab.Controls.Add(this.Calculate);
        this.ModelTab.Controls.Add(this.Save);
        this.ModelTab.Location = new System.Drawing.Point(4,
22);
        this.ModelTab.Name = "ModelTab";
        this.ModelTab.Padding = new
System.Windows.Forms.Padding(3);
        this.ModelTab.Size = new System.Drawing.Size(291, 403);
        this.ModelTab.TabIndex = 0;

```

```

        this.ModelTab.Text = "Input Model";
        this.ModelTab.UseVisualStyleBackColor = true;
        //
        // B0Label
        //
        this.B0Label.AutoSize = true;
        this.B0Label.Font = new System.Drawing.Font("Microsoft
Sans Serif", 15.25F);
        this.B0Label.Location = new System.Drawing.Point(17,
59);

        this.B0Label.Name = "B0Label";
        this.B0Label.Size = new System.Drawing.Size(123, 25);
        this.B0Label.TabIndex = 5;
        this.B0Label.Text = "b0=";
        //
        // B1Label
        //
        this.B1Label.AutoSize = true;
        this.B1Label.Font = new System.Drawing.Font("Microsoft
Sans Serif", 15.25F);
        this.B1Label.Location = new System.Drawing.Point(17,
100);

        this.B1Label.Name = "B1Label";
        this.B1Label.Size = new System.Drawing.Size(116, 25);
        this.B1Label.TabIndex = 6;
        this.B1Label.Text = "b1=";
        //
        // InputModel
        //
        this.InputModel.Location = new System.Drawing.Point(6,
327);

        this.InputModel.Name = "InputModel";
        this.InputModel.Size = new System.Drawing.Size(279,
32);

        this.InputModel.TabIndex = 3;
        this.InputModel.Text = "Input Model";
        this.InputModel.UseVisualStyleBackColor = true;
        this.InputModel.Click += new
System.EventHandler(this.InputModelForRegresionClick);
        //
        // Save
        //
        this.Save.Location = new System.Drawing.Point(6, 365);
        this.Save.Name = "Save";
        this.Save.Size = new System.Drawing.Size(279, 32);
        this.Save.TabIndex = 8;
        this.Save.Text = "Save";
        this.Save.UseVisualStyleBackColor = true;

```

```

        this.Save.Click += new
System.EventHandler(this.SaveClick);
        //
        // CalculationTab
        //
        this.CalculationTab.Controls.Add(this.InputGroup);
        this.CalculationTab.Controls.Add(this.ResultGroup);
        this.CalculationTab.Controls.Add(this.CalculateButton);

this.CalculationTab.Controls.Add(this.InputModelButton);

this.CalculationTab.Controls.Add(this.FileIsReadyLabel);
        this.CalculationTab.Location = new
System.Drawing.Point(4, 22);
        this.CalculationTab.Name = "CalculationTab";
        this.CalculationTab.Padding = new
System.Windows.Forms.Padding(3);
        this.CalculationTab.Size = new System.Drawing.Size(291,
403);
        this.CalculationTab.TabIndex = 1;
        this.CalculationTab.Text = "Calculation";
        this.CalculationTab.UseVisualStyleBackColor = true;
        //
        // MainView
        //
        this.AutoScaleDimensions = new System.Drawing.SizeF(6F,
13F);
        this.AutoScaleMode =
System.Windows.Forms.AutoScaleMode.Font;
        this.ClientSize = new System.Drawing.Size(323, 450);
        this.Controls.Add(this.tabControl1);
        this.FormBorderStyle =
System.Windows.Forms.FormBorderStyle.FixedSingle;
        this.MaximizeBox = false;
        this.Name = "MainView";
        this.Text = "Duration";
        this.Load += new
System.EventHandler(this.MainView_Load);
        this.InputGroup.ResumeLayout(false);
        this.InputGroup.PerformLayout();
        this.ResultGroup.ResumeLayout(false);
        this.ResultGroup.PerformLayout();
        this.tabControl1.ResumeLayout(false);
        this.ModelTab.ResumeLayout(false);
        this.ModelTab.PerformLayout();
        this.CalculationTab.ResumeLayout(false);
        this.CalculationTab.PerformLayout();
        this.ResumeLayout(false);

```

```

    }

    #endregion
    private System.Windows.Forms.Button InputModelButton;
    private System.Windows.Forms.Button CalculateButton;
    private System.Windows.Forms.GroupBox InputGroup;
    private System.Windows.Forms.GroupBox ResultGroup;
    private System.Windows.Forms.Label FileIsReadyLabel;
    private System.Windows.Forms.TabControl tabControl1;
    private System.Windows.Forms.TabPage ModelTab;
    private System.Windows.Forms.TabPage CalculationTab;
    private System.Windows.Forms.Label ResultLabel;
    private System.Windows.Forms.Button InputModel;
    private System.Windows.Forms.Label B0Label;
    private System.Windows.Forms.Label ModelLabel;
    private System.Windows.Forms.Label B1Label;
    private System.Windows.Forms.Button Save;
}
}

```

Controller.cs

```

using System;
using System.Collections.Generic;
using System.IO;
using Newtonsoft.Json;

namespace Metrics_Program
{
    public class Controller
    {
        private MainView mainView;
        private Data data;

        private List<DataPoint> inputData;

        private string dataPath = "./data.txt";

        public Controller()
        {
            this.data = new Data();
        }

        public void SaveResult()
        {
            float resultData = Calculate();
            data.SetResultData(resultData);
        }
    }
}

```

```

    }

    public void CleanInputDataFromEmmissions()
    {
        data.CleanInputDataFromEmmissions();
    }

    public void SaveResult(string jsonFilePath, object model)
    {
        string json = JsonConvert.SerializeObject(model);
        File.WriteAllText(jsonFilePath, json);
    }

    public void CalculateCoefficients(List<DataPoint>
inputData)
    {
        Matrix<double> inputMatrix =
BuildInputMatrix(inputData);
        Vector<double> outputVector =
BuildOutputVector(inputData);

        RegressionValue value = new RegressionValue();
        value.B0 = inputMatrix.QR().Solve(outputVector)[0];
        value.B1 = inputMatrix.QR().Solve(outputVector)[1];

        data.SetResultData(value);
    }

    private Matrix<double> BuildInputMatrix(List<DataPoint>
inputData)
    {
        int rows = inputData.Count;
        int cols = 2;
        double[,] inputArray = new double[rows, cols];
        for (int i = 0; i < rows; i++)
        {
            inputArray[i, 0] = 1.0;
inputArray[i, 1] = inputData[i].Features[0];
        }

        return Matrix<double>.Build.DenseOfArray(inputArray);
    }

    private Vector<double> BuildOutputVector(List<DataPoint>
inputData)
    {
        double[] outputArray = new double[inputData.Count];
        for (int i = 0; i < inputData.Count; i++)
        {

```

```

        outputArray[i] = inputData[i].Result;
    }

    return Vector<double>.Build.DenseOfArray(outputArray);
}

public float GetDuration(float x1, RegressionValue value)
{
    if (inputData == null || inputData.Count == 0)
    {
        return 1f;
    }

    NonlinearModel nonlinearModel = new NonlinearModel();
    return nonlinearModel.Execute(x1, value);
}

public class DataPoint
{
    public double X1 { get; set; }
    public double Y { get; set; }

    public double[] Features { get; set; }
}

public class RegressionValue
{
    public double B0 { get; set; }
    public double B1 { get; set; }
}

public interface IModelAlgorithm
{
    float Execute(float x1, RegressionValue value);
}

public class NonlinearModel : IModelAlgorithm
{
    public float Execute(float x1, RegressionValue value)
    {
        float baseValue = (float)Math.Pow(10, value.B0);
        float x1Normalized = (float)Math.Pow(x1, value.B1);

        float duration = baseValue + x1Normalized;
        return duration;
    }
}
}

```

ДОДАТОК В – ОПИС ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ПЗ ІНТЕРНЕТ-МАГАЗИНІВ

1 Загальні відомості

Повна назва розроблюваного проекту: Програма для оцінювання тривалості розробки ПЗ інтернет-магазинів.

Коротка назва: програма.

Галузь застосування: програма призначена для застосування в компаніях та командах розробників, що займаються розробки ПЗ інтернет-магазинів.

2 Експлуатаційне призначення

Програма може використовуватися для оцінювання тривалості розробки ПЗ інтернет-магазинів в компаніях та командах розробників, що займаються розробкою відповідного ПЗ.

3 Функціональне призначення

Дана програма призначена для оцінювання тривалості розробки ПЗ інтернет-магазинів.

Програма повинна забезпечувати можливість виконання наведених нижче функцій:

- Введення даних про параметри регресійної моделі (значення коефіцієнтів рівняння регресії).
- Введення значення довірчої ймовірності.
- Введення назви проєкту розробки інтернет-магазину та значення трудомісткості його розробки для подальшого розрахунку.
- Оцінювання трудомісткості розробки інтернет-магазину.

– Збереження результатів оцінювання трудомісткості розробки інтернет-магазину в базу даних.

4 Технічні засоби, що використовуються

Для роботи з програмним продуктом необхідний наступний склад технічних засобів з мінімальними параметрами:

- CPU: Core i3;
- RAM: 1 Гб;
- HDD/SDD 500 Мб;
- пристрої вводу-виводу: монітор з екраном роздільної здатності не менше ніж 1600x900 пікселів, клавіатура, маніпулятор типу «миша».

5 Програмні засоби, що використовуються

Для використання даної програми потрібно мати операційну систему Windows 10 або вище.

6 Вхідні дані

Вхідні дані вводяться у відповідні поля форми відповідно до допустимих типів (числові або текстові).

До вхідних даних належать: параметри регресійної моделі b_0 , b_1 , значення довірчої ймовірності, назва проєкту, значення трудомісткості розробки.

7 Вихідні дані

Вихідні дані виводяться у відповідні поля форми та зберігаються в базі даних.

Вихідними даними є прогнозована тривалість розробки ПЗ інтернет-магазинів.

ДОДАТОК Г – ІНСТРУКЦІЯ КОРИСТУВАЧА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ПЗ ІНТЕРНЕТ-МАГАЗИНІВ

Програма для оцінювання тривалості розробки ПЗ інтернет-магазинів встановлюється безпосередньо на комп'ютер користувача. Програма запускається або за допомогою ярлика, який знаходиться на робочому столі, або за допомогою вибору відповідного пункту меню «Пуск» – «Програми». Програма має один режим використання.

Після запуску програми з'являється основне вікно програми, яке має дві вкладки: вкладка введення даних про параметри регресійної моделі "Input Model" та вкладка прогнозу розміру "Calculation". Основне вікно програми для оцінювання тривалості розробки ПЗ інтернет-магазинів представлено на рисунку Г.1.

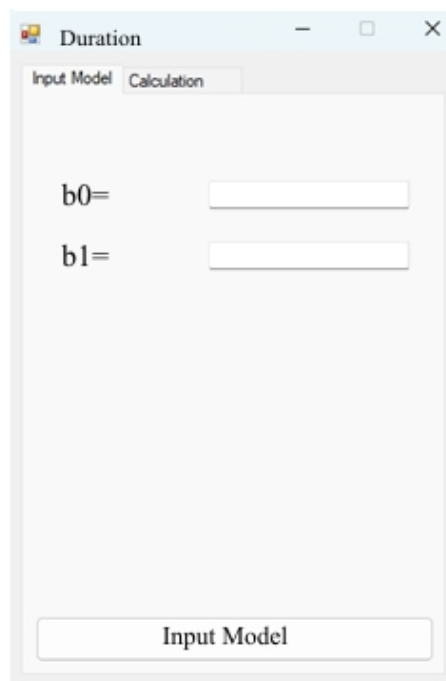
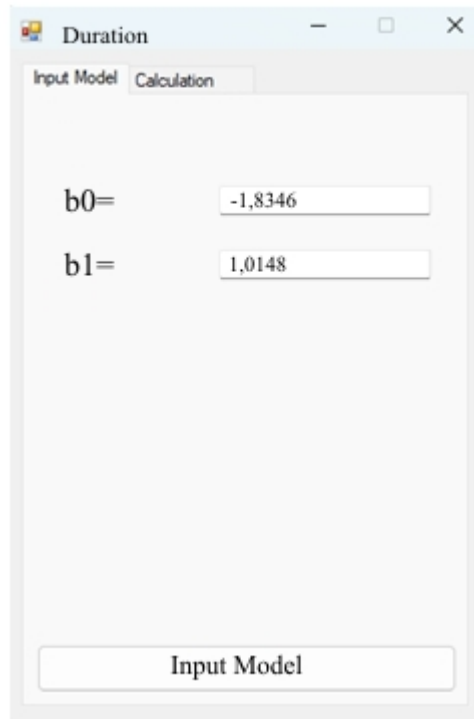


Рисунок Г.1 – Основне вікно програми для оцінювання тривалості розробки ПЗ інтернет-магазинів

Для початку роботи треба ввести дані про параметри регресійної моделі b_0 , b_1 та натиснути кнопку "Input Model" (див. рис. Г.2).



The screenshot shows a software window titled "Duration". It contains two tabs: "Input Model" (which is active) and "Calculation". In the "Input Model" tab, there are two input fields. The first is labeled "b0=" and contains the value "-1,8346". The second is labeled "b1=" and contains the value "1,0148". At the bottom of the window, there is a button labeled "Input Model".

Рисунок Г.2 – Ввод даних про параметри регресійної моделі

Для того щоб виконати оцінювання тривалості розробки ПЗ інтернет-магазинів потрібно перейти на вкладку "Calculation", зовнішній вигляд якої представлено на рисунку Г.3.



The image shows a software window titled "Duration". Inside the window, there is a tab labeled "Calculation". Below the tab, there is a section titled "Input for calculation" containing three input fields: "Project", "X=", and " α =". Below this section is a "Result" section with a label "Duration:" and a corresponding output field. At the bottom of the window, there are two buttons: "Calculate" and "Save".

Рисунок Г.3 – Зовнішній вигляд вкладки "Calculation"

Для оцінювання тривалості розробки ПЗ інтернет-магазину треба ввести дані проєкту та значення довірчої ймовірності і натиснути кнопку "Calculate". Розраховане значення тривалості розробки з'явиться у відповідному полі "Duration" (див. рис. Г.4)

The image shows a software window titled "Duration". Inside the window, there is a tab labeled "Input Model" with a sub-tab "Calculation" selected. Below this, there is a section titled "Input for calculation" containing three input fields: "Project" with the value "Test 1", "X=" with the value "750", and " α =" with the value "0,05". Below the input fields is a section titled "Result" containing a single line "Duration: 12,11". At the bottom of the window, there are two buttons: "Calculate" and "Save".

Рисунок Г.4 – Розрахунок тривалості розробки

Для збереження результатів оцінювання в базу даних треба натиснути кнопку "Save".

ДОДАТОК Д – ПРОГРАМА ТА МЕТОДИКА ВИПРОБУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ПЗ ІНТЕРНЕТ-МАГАЗИНІВ

1 Об'єкт випробовування

Об'єкт випробувань: : Програма для оцінювання тривалості розробки ПЗ інтернет-магазинів.

2 Мета випробовування

Перевірка функціональних можливостей ПЗ для оцінювання тривалості розробки ПЗ інтернет-магазинів на відповідність вимогам технічного завдання.

3 Вимоги до програми

При проведенні випробувань, функціональні характеристики ПЗ підлягають перевірці на відповідність вимогам, викладеним у пункті «Вимоги до функціональних характеристик» технічного завдання.

Для проведення випробувань підлягають перевірці наступні функції:

- Введення даних про параметри регресійної моделі (значення коефіцієнтів рівняння регресії).
- Введення значення довірчої ймовірності.
- Введення назви проєкту розробки інтернет-магазину та значення трудомісткості його розробки для подальшого розрахунку.
- Оцінювання трудомісткості розробки інтернет-магазину.
- Збереження результатів оцінювання трудомісткості розробки інтернет-магазину в базу даних.

4 Вимоги до програмної документації

Програмна документація повинна містити:

- технічне завдання;
- текст програми;
- опис програми;
- інструкцію користувача;
- програму та методику випробування ПЗ.

5 Засоби та порядок випробувань

Для роботи із програмою необхідний наступний склад технічних засобів з мінімальними параметрами:

- CPU: Core i3;
- RAM: 1 Гб;
- HDD/SDD 500 Мб;
- пристрої вводу-виводу: монітор з екраном роздільної здатності не менше ніж 1600x900 пікселів, клавіатура, маніпулятор типу «миша».

Програмні засоби, що повинні використовуватися під час випробувань

Для використання даної програми потрібно мати операційну систему Windows 10 або вище.

Випробування ПЗ повинні виконуватися в наступному порядку:

- виконуються інструкції до кожної функції;
- робиться аналіз отриманих результатів і встановлюється відповідність програмного продукту вимогам і системним документам.

При виявленні помилок у роботі програмного продукту складається їх перелік і обговорюється термін їх виправлення розробником. Після цього замовник проводить повторне тестування в повному обсязі (можливе використання нових чи додаткових тестів).

6 Методи випробувань

Випробування будемо проводити за стратегією «чорного ящика». Всі функції програми будуть проходити випробовування. В таблиці Д.1 наведено перелік випробувань, що пропонуються до виконання:

Таблиця Д.1 – Методика випробувань

№	Дія	Очікуваний результат
1	Введення даних про параметри регресійної моделі (b_0 , b_1)	Відображення результату введення даних про параметри регресійної моделі
2	Введення значення довірчої ймовірності	Відображення результату введення значення довірчої ймовірності
3	Введення назви проєкту розробки інтернет-магазину та значення трудомісткості його розробки	Відображення результату введення даних проєкту: його назви та значення трудомісткості його розробки
4	Оцінювання трудомісткості розробки інтернет-магазину	Відображення результату оцінювання трудомісткості розробки інтернет-магазину (прогнозне значення трудомісткості)
5	Збереження результатів оцінювання трудомісткості в базу даних	Запис в базу даних вхідних даних та результату оцінювання