

Центральноукраїнський національний технічний університет
Механіко-технологічний факультет
Кафедра кібербезпеки та програмного забезпечення

”Допущено до захисту”
Завідувач кафедри кібербезпеки
та програмного забезпечення
д.т.н., професор
_____ Олексій СМІРНОВ
« ____ » _____ 2026 р.

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
за першим (бакалаврським) рівнем вищої освіти
на тему
“Програмне забезпечення системи обробки даних двовимірних
матричних штрихкодів типу Data MatrixECC 200”

Виконав здобувач вищої освіти
IV курсу, групи КН-22
ОПП «Комп’ютерні науки»
спеціальності 122 «Комп’ютерні науки»
_____ Осадчий В.К.
« ____ » _____ 2026 р.

Керівник проекту
кандидат технічних наук, доцент
_____ Коваленко А.С.
« ____ » _____ 2026 р.
Рецензент _____

Центральноукраїнський національний технічний університет
Факультет Механіко-технологічний
Кафедра Кібербезпеки та програмного забезпечення
Освітній ступінь бакалавр
Галузь знань . 12 “Інформаційні технології”
Спеціальність 122 “Комп’ютерні науки”
Освітньо-професійна (освітньо-наукова) програма “Комп’ютерні науки”

ЗАТВЕРДЖУЮ

Завідувач кафедри

д.т.н., проф.

Олексій СМІРНОВ

« 06 » лютого 2026 року

ЗАВДАННЯ НА ВИПУСКНУ КВАЛІФІКАЦІЙНУ РОБОТУ ЗА ПЕРШИМ (БАКАЛАВРСЬКИМ) РІВНЕМ ВИЩОЇ ОСВІТИ ЗДОБУВАЧА ВИЩОЇ ОСВІТИ

Осадчому Владиславу Костянтиновичу

(прізвище, ім’я, по батькові)

1. Тема роботи *Програмне забезпечення системи обробки даних
двовимірних матричних штрихкодів типу Data
MatrixECC 200*

2. Керівник роботи *Коваленко Анна Степанівна, канд. техн. наук, доцент*

(прізвище, ім’я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу № 116-02 від 06.02.2026 року

3. Строк подання студентом роботи до захисту *23.05.2026 р.*

4. Мета та завдання випускної кваліфікаційної роботи: *Метою роботи є розробка
програмного забезпечення системи обробки даних двовимірних матричних
штрихкодів типу Data MatrixECC 200*

5. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно
розробити)

1. Призначення та область використання.

2. Перегляд аналогічних існуючих систем.

3. Опис і обґрунтування проектних рішень.

4. Етапи програмування системи.

5. Впровадження системи в промислову експлуатацію.

6. Висновки

6. Перелік графічного матеріалу (з точним зазначенням обов’язкових креслень)

Структурна схема системи *1 аркуш*

Функціональна схема системи *1 аркуш*

Діаграма процесів *1 аркуш*

Блок-схема алгоритму роботи додатку *2 аркуша*

7. Дата видачі завдання « 06 » лютого 2026 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти	Строк виконання етапів випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти	Примітка
1.	Аналіз існуючих систем	10.03.2026 р.	
2.	Постановка задачі, оформлення ТЗ	15.03.2026 р.	
3.	Розробка моделі компонента	20.03.2026 р.	
4.	Розробка структур даних	25.03.2026 р.	
5.	Розробка алгоритмів зв'язку та відображення	30.03.2026 р.	
6.	Програмування алгоритмів	10.04.2026 р.	
7.	Оформлення ПЗ	17.04.2026 р.	
8.	Попередній захист роботи	23.05.2026 р.	

Дата видачі завдання
« 06 » лютого 2026 р.

Підпис керівника

Коваленко А.С.
(прізвище та ініціали)

Завдання прийнято до виконання
« 06 » лютого 2026 р.

Підпис здобувача

Осадчий В.К.
(прізвище та ініціали)

АНОТАЦІЯ

Осадчий В.К. Програмне забезпечення системи обробки даних двовимірних матричних штрихкодів типу Data MatrixECC 200. 122 Комп'ютерні науки. Центральнотукаїнський національний технічний університет. Кропивницький. 2026.

В даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи обробки даних двовимірних матричних штрихкодів типу Data MatrixECC 200.

Метою розробки є програмне забезпечення системи обробки даних двовимірних матричних штрихкодів типу Data MatrixECC 200.

Результат роботи – програмна реалізація системи обробки даних двовимірних матричних штрихкодів типу Data MatrixECC 200.

В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

Розроблено зручний інтерфейс користувача. Наведені інструкції по роботі з програмними засобами.

Програма може використовуватися на ПЕОМ з ОС Windows 10/11.

Програму розроблено в середовищі Python.

Ключові слова: комп'ютерні науки, обробка даних, двовимірні матричні штрихкоди

ABSTRACT

Osadchyi V.K. Software for the data processing system of two-dimensional matrix barcodes of the Data MatrixECC 200 type. 122 Computer Science. Central Ukrainian National Technical University. Kropyvnytskyi. 2026.

In this final qualification work for the first (bachelor's) level of higher education, software has been developed, which is intended for the data processing system of two-dimensional matrix barcodes of the Data MatrixECC 200 type.

The purpose of the development is the software for the data processing system of two-dimensional matrix barcodes of the Data MatrixECC 200 type.

The result of the work is the software implementation of the data processing system of two-dimensional matrix barcodes of the Data MatrixECC 200 type.

In the process of working on the software model, an analysis of existing hardware and software was performed. All components of the developed software are fully described.

A convenient user interface has been developed. Instructions for working with software are provided.

The program can be used on PCs with Windows 10/11.

The program is developed in the Python environment.

Keywords: computer science, data processing, two-dimensional matrix barcodes

3MIST

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ	2
ВСТУП.....	3
1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ	6
1.1 Призначення системи.....	6
1.2 Область застосування.....	11
2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ	14
2.1 Огляд існуючих систем, технологій, архітектур та програмних рішень за профілем теми випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти.....	14
2.2 Обґрунтування вибору засобів для побудови системи та мови програмування.....	20
2.3 Розгорнута постановка завдання	25
3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ	27
3.1 Опис функціонування системи	27
3.2 Розробка структурної схеми.....	29
3.3 Розробка функціональної схеми	34
3.4 Розробка діаграми процесів.....	36
4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ.....	38
4.1 Розробка блок-схем та опис алгоритмів функціонування системи.....	38
4.2 Захист розробленого програмного забезпечення.....	53
5 ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ	55
6 ОСНОВНІ ВИСНОВКИ.....	61
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	63

					ВКРБ-122.26.0022.00.00.ПЗ				
Вим	Арк.	№ докум.	Підп.	Дата	Програмне забезпечення системи обробки даних двовимірних матричних штрихкодів типу Data MatrixECC 200	Лім.	Аркуш	Аркушів	
Розроб.	Осадчий В.К.					Б		1	70
Перев.	Коваленко А.С.								
Н.контр.	Коваленко А.С.								
Затв.	Смірнов О.А.					ЦНТУ КН-22			

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

OCR	—	Optical Character Recognition
DataMatrixECC200	—	матричний штрих код
КПК	—	кишеньковий персональний комп'ютер
ПЗ	—	програмне забезпечення

К6ПЗ_2026

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		2

ВСТУП

Актуальність теми. До маркування на товарах у вигляді штрихових кодів звикли багато покупців. Вони зустрічаються на продуктах харчування або на упакуванні побутової техніки, на буклетах, вуличних вивісках, квитанціях, пачках сигарет, медикаментах і пляшках з алкоголем. Не всі знають про те, що існують лінійні й двовимірні штрихкоди. Прийнятий для позначення DataMatrix code відрізняється від QR-code. Крім цього, існують нюанси, пов'язані з розшифровкою й декодуванням Data Matrix кодів і використанням касових апаратів для сканування маркірованих товарів. У всьому цьому постараємося докладно розібратися в нашій роботі.

Чим відрізняється QR-код від DataMatrix

Перед тим, як розібратися, чим відрізняються QR-код і DataMatrix, з'ясуємо, що собою представляє штрихкод і яким образом його наносять на поверхню.

Штрихкод – послідовне нанесення смуг білий і чорний кольори. У них зашифрована інформація про товар, продукт або послугу. Її зчитують спеціально призначені для цього технічні пристрої. Нерідко поруч зі знаком ця ж інформація дублюється для споживача у вигляді друкованого тексту.

Існують лінійні й двомірні штрихкоди.

Вони використовуються:

- у торгівлі;
- у роботі систем охоронної сигналізації;
- у роботі бібліотек;
- при здійсненні операцій по складському обліку;
- у процесі обробки поштових документів;
- при здійсненні виробничих техпроцесів.

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		3

Винайшли цю корисну для людей систему кодування в 1938 році три молоді американські вчені Сільвер, Вудланд і Джохансон. Прототип штрихкоду Вудланд винайшов на підставі принципу функціонування абетки Морзе. Йому вдалося подовжити звичні всім крапки й тире й додатково використовувати спосіб оптичного прочитання закодованих даних. Патент на свій винахід учені одержали в 1952 році.

Спочатку створили лінійне кодування. Принцип його роботи полягає в тому, що інформація зчитується в горизонтальному напрямку з використанням спеціального сканера. До розповсюдженого лінійного кодування ставиться EAN-символіка.

Перевага цього виду маркування – простота. Для зчитування необхідної інформації не потрібно дороге устаткування. Створений код у вигляді числової послідовності вміщає в себе в цьому випадку не більше 30 символів.

Двовимірне кодування містить у собі чотири види:

- PDF417;
- Aztec Code;
- DataMatrix;
- QR-код.

Мета й завдання дослідження. Метою роботи є програмне забезпечення системи обробки даних двовимірних матричних штрихкодів типу Data MatrixECC 200.

Для досягнення поставленої мети визначена програма дослідження, що складається з наступних завдань:

- Огляд існуючих систем обробки даних двовимірних матричних штрихкодів типу Data MatrixECC 200.
- Дослідження системи обробки даних двовимірних матричних штрихкодів типу Data MatrixECC 200.
- Програмна реалізація системи обробки даних двовимірних матричних штрихкодів типу Data MatrixECC 200.

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		4

Практична цінність отриманих результатів полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі обробки даних двовимірних матричних штрихкодів типу Data MatrixECC 200.

Таким чином, виходячи з вищеперерахованого, програмне забезпечення системи обробки даних двовимірних матричних штрихкодів типу Data MatrixECC 200, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти.

КБПЗ_2026

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		5

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Особливості й переваги друку Data Matrix code

Те, що друк DataMatrix code – це та ж саме, що друк QR-code – омана. QR – це самостійний вид двомірного кодування.

DataMatrix код – двомірна матриця, створена співробітниками компанії International Data Matrix. В 2005 році винахід викупила Siemens. Щоб ним сьогодні скористатися, не треба одержувати ліцензію й платити за це гроші.

У горизонтальному й у вертикальному положенні розміщена парна кількість модулів різного ступеня яскравості на білому тлі. У кожному із цих блоків дві або більше пересічні лінії, що зображують букви L.

Якщо отримані дані ушкоджені, то до 30% з них відновлюється за допомогою алгоритму корекції Ріда-Соломона, що використовується в багатьох системах. За допомогою декодера Ріда-Соломона обробляється окремо кожний блок, після чого відбувається виправлення помилок і відновлення вихідних даних.

Переваги Matrix Data code

До головної переваги ставиться мінімальний розмір DataMatrix. На площі 2 на 2 мм шифрується до 50 символів. Інформація буде відрізнятися високим ступенем яскравості й контрастності.

Цей вид коду наноситься на поверхню для наступного зчитування за допомогою сканерів мобільного або стаціонарного типу, що забезпечує високий ступінь схоронності закодованих даних. Сканер здатний прочитати інформацію при наявності дефектів і втраті зображення до 62%.

Використання формату DataMatrix заощаджує в цілому від 30 до 60% місця для нанесення інформації з порівняння з QR-кодом.

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		6

До переваг матричного штрихкоду цього типу відноситься можливість розміщення:

- текстової й цифрової інформації;
- номерів телефону;
- e-mail;
- інтернет-посилань і url-адрес;
- будь-якої іншої потрібної інформації.

У цілому щільність розміщення даних становить до 500 млн. символів на один дюйм.

Версії DataMatrix символики

Використовуються дві версії символики:

– ECC 000-140. Цю символіку застосовують винятково в замкнених прикладних системах.

– ECC 200. У ній утримується парне число елементів, розміщених по кожній зі сторін. Ємність цього символу, що складає з 144 модулів, містить у собі максимум 3116 цифр і не більше 2335 букв.

DataMatrix ECC 200 по ISO/IEC 16022 – єдина версія, що підтримує структури даних GS1 системи разом із застосуванням функціонального знака FNC 1.

Це означає, що крім кодування ідентифікаційних ключів GS1, розпізнаються додаткові дані (вага, дата випуску, номер і серія партії, строк придатності).

GS1 – це 2D-символіка DataMatrix маркування, що включає в себе білі й чорні модулі, які потім упорядковуються в матрицю прямокутної або квадратної форми.

Вона застосовується в сфері виробництва фармацевтичної продукції й медичних виробів завдяки можливості:

– прямого маркування хірургічних інструментів або малих виробів і упакувань для медицини, на яких не можна розмістити окрему етикетку;

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		7

- маркування на невеликій площі величезного обсягу інформації;
- виявлення й виправлення помилки в процесі експлуатації, що дасть можливість, незважаючи на ушкодження впакування, забезпечити зчитуваність символіки.

Алгоритм розпізнавання DataMatrix с 2D-технологією дає можливість закодувати в цілому до 3116 цифр і 2335 латинських букв, які вміщують у себе 1556 байт інформації.

Способи й системи нанесення DataMatrix коду

Крапки в DataMatrix не тільки квадратні, але й круглі. Круглі наносяться за допомогою крапле-струминного друку або голко-ударного маркування. Стандарти дозволяють перевищувати максимальний ідеальний розмір крапок на 105%, а їхній мінімальний розмір не може виходити за рамки 60% від ідеального.

Цей вид двомірного маркування наноситься на продукти й товари (упакування, таблички, етикетки й інші поверхні) типографським способом або шляхом виготовлення на особливих принтерах спеціальних наклейок.

Для цієї мети застосовується метод гравірування, струминна й лазерна друк. До затребуваного ставляться ударно-крапкові маркировочні системи й лазерні маркировочні машини.

Ударно-крапкові маркировочні системи

У порівнянні з маркіраторами лазерного типу, саме ударно-крапкове гравіровочне устаткування відрізняється меншими розмірами. Така виробнича лінія буде дешевою й компактною.

Цей варіант нанесення інформації використовується в тих галузях, у яких лазерна обробка неприпустима. Наприклад, це ставиться до аерокосмічного устаткування. Деталі й вузли для цієї галузі не можна маркірувати за допомогою лазера, тому що теплове випромінювання впливає на властивості матеріалу, образуючи в ньому мікротріщини.

У результаті нанесення кодової інформації за допомогою голкоударного маркіратора в матеріалі з усіх боків утворюються поглиблення й своєрідні вигини.

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		8

Це дозволить охоронити деталь від концентрації напруги. Отримані штрихкоди менш стираєми в порівнянні з тими, які нанесені лазерно-маркировочним устаткуванням.

Лазерна маркірувальна машина

З її допомогою DataMatrix-коди створюються на будь-якому матеріалі, у т.ч. на металі й пластику. Якщо розміщення зашифрованих даних треба зробити швидко й при цьому нанести зображення мінімального розміру, то для цього оптимальними будуть лазерні машини.

У порівнянні з ударно-крапковим, таке зображення буде більше точним. Ідентифікаційні дані будуть зчитуватися швидше й краще. За допомогою лазерного апарата на поверхню наносяться не круглі, як від ударно-крапкового устаткування, а прямокутні або квадратні фігури. При цьому код виходить не тільки контрастним, але й легким для зчитування й ідеально чітким.

Друк двомірного QR-code

Це матричний код двомірного типу на початку 90-х років минулого століття винайшли інженери японської компанії Denso-Wave. У цій країні він був дуже популярний через розпізнавання символів кани (японських ієрогліфів). В 1994 році винахід презентували широкої громадськості. Тепер його використовують по усьому світі біля 7 абонентів мобільного зв'язку. Це штрихкод без офіційного стандарту, яким вільно й безкоштовно користуються не тільки різні компанії, але й звичайні люди різних країн і континентів. Розміри QR-code можуть становити від 21 на 21 до 177 на 177 модулів (версії від 1 до 40. Для однієї сторони крок дорівнює чотирьом модулям). вільний простір, Що Перебуває навколо коду, залежить від версії й містить у собі від двох до чотирьох модулів.

Цей варіант широко застосовується, у тому числі в якості двомірного штрихкоду на податковій декларації відповідного формату.

Використання цього винаходу пов'язане з декількома видами діяльності:

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		9

– З рекламним бізнесом і маркетингом. Скориставшись зображенням на рекламному носії, користувач переглядає промо-ролик або відвідує віртуальний магазин замовника.

– З туризмом, виставочними заходами й музеями. QR-code часто розміщується на визначних пам'ятках, що користуються популярністю. З його допомогою можна одержати коротку довідку, відкрити сторінку в «Вікіпедії», довідатися багато інших новин у світі туризму.

– З оформленням квитків. При придбанні квитка в онлайн-режимі користувач одержує QR-code для збереження на мобільному пристрої. Після цього QR-code досить прикласти до того, що знаходиться в аеропорті, автобусному, залізничному вокзалі до зчитувача.

– З візитками. У коді зашифровуються всі реквізити про її власника.

– Із квитанціями по оплаті ЖКП. Розшифровується інформація за допомогою смартфона зі спеціальною програмою, що зчитує QR-code.

Загальна ємність QR-code і корекції помилок

Якщо порівнювати з Data Matrix кодом, то може здатися, що QR через менші можливості зберігає в собі в закодованому виді тільки URL або електронні адреси. Насправді загальний обсяг QR-code у цифровому вираженні дорівнює 7 089, а в цифрах і плюсах у латинських буквах 4 296 символам. У цілому кодується близько 2 Кб. Це розширює спектр використання такого варіанта розміщення даних з обліком не тільки оптимальної швидкості, але й зручності доставки до кінцевого користувача.

Чотирьохрівнева система виправлення помилок відбувається за допомогою того ж алгоритму Ріда-Соломона. Це дозволяє без особливої праці зчитувати ушкоджену, затерту, перекреслену інформацію.

1.2 Область застосування

Криптозахист лікарських засобів по системі маркування «Чесний знак». Інтеграція з DataMatrix

З лютого 2017 року почав працювати пілотний проект, пов'язаний з маркуванням фармацевтичної продукції. З початку 2020 року всі лікарські засоби й препарати підлягають обов'язковому маркуванню. Сьогодні виробники на добровільних початках розміщують на пакувальних матеріалах DataMatrix код, у якому перебуває докладна інформація про лікарські засоби. З листопада 2018 року система маркування ліків попадає під дію всеукраїнської системи маркування товарів «Чесний Знак». До самого DataMatrix коду повинна додатися криптозахист. Для того щоб використовувати крипто захищений DataMatrix, всім виробникам медичної продукції треба буде придбати відповідне устаткування.

Продаж сигарет і іншої табачної продукції із цифровим кодом Data Matrix

З 15 січня 2018 року почався експеримент, відповідно до якого пачки й блоки сигарет маркуються за допомогою двовимірного кодування DataMatrix. Виробник сигарет запитує в інформаційній системі маркування й оборту тютюнової продукції відповідний код. У ньому буде присутній інформація про марку виробу, місці й даті його виготовлення. На бічну частину блоку й на торець пачки сигарет наноситься унікальний 2D-ідентифікатор. Завдяки йому відбувається захист паперових DataMatrix від повторного копіювання. Отриманий код не скопіюєш і не підробиш. Інформація, що втримується в коді, направляється держорганам у момент продажу.

Процес реалізації тютюнової продукції відбувається в такий спосіб. На першому етапі дистриб'ютор закупає сигарети. Для наступного продажу дистриб'ютор одержує від виробника УПД-накладну за допомогою системи ЕДО (електронного документообігу). Обліку підлягає кожний код, що перебуває на пачці. Потім дистриб'ютором перевіряються всі марки сигарет і їхньої партії.

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		11

Після прийняття УПД-накладної, потрібні дані інформація через ЕДО йде в інформаційну систему маркування й оборту тютюнової продукції. Після проведення всіх маніпуляцій партія сигарет передається дистриб'ютором для продажу в торговельну точку. Потім всі дані за допомогою системи електронного документообігу надходить в ІС МОТП. У результаті на кожному з етапів продажу сигарет відбувається контроль продукції. У момент продажу пачки або блоку сигарет продавець-касир за допомогою спеціального оператора фіскальних даних направляє інформацію про проданий товар.

Яку онлайн-касу придбати для розпізнавання маркування у форматі DataMatrix

Щоб зчитувати передбачену законом обов'язкове маркування, варто придбати онлайн-касу, до якої згодом можна підключити 2D-сканер. Крім того, касовий апарат повинен уміти розпізнавати маркування товару й мати можливість роботи з оператором фіскальних даних.

Сучасні виробники ККТ пропонують широкий вибір таких кас, які здатні розпізнавати обов'язкове маркування. Для правильного вибору варто проконсультуватися з фахівцями, які підберуть потрібний апарат і сканер штрих-кодів.

ККТ, що розпізнає штрих-коди, роблять Атол, Евотор, Дримкас, Штрих-М. Прийом або відпустка товару здійснюється після його сканування за допомогою системи електронного документообігу, інтегрованої із програмним забезпеченням комплексу. Продаж продукції фіксується в касовому чеку. Касир усього лише зчитує сканером код, що перебуває на впакуванні. Перевагою пропонованих онлайн-кас є їхній великий вибір і доступна для представників бізнесу вартість. При бажанні підприємець може замовити індивідуальний комплект необхідного для роботи устаткування.

Резюме

Розміщення двомірних Data Matrix або QR code на товарах, продуктах харчування, квитках, квитанціях по оплаті комунальних послуг дозволяє

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		12

заощадити місце й час для інформування продавців і споживачів про всі властивості продукції. Ці два види кодування наносяться за допомогою спеціального устаткування й відрізняються друг від друга кількістю розташовуваних символів. У результаті покупець одержує криптозахист від підробки, держава – інструмент контролю продаваної продукції, а бізнесмен має додаткові витрати на придбання відповідного устаткування.

Таким чином, виходячи з вищеперерахованого, програмне забезпечення системи обробки даних двовимірних матричних штрихкодів типу Data MatrixECC 200, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти.

К6ПЗ_2026

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		13

2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ

2.1 Огляд існуючих систем, технологій, архітектур, програмних рішень за профілем теми випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти

У 2026 році програми для зчитування QR-кодів значно еволюціонували. Тепер вони стали інтуїтивно зрозумілішими та швидшими. Користувачі цінують зручність сканування QR-кодів для різних цілей. Багато функцій покращують взаємодію з користувачем. До них належать журнали історії, параметри налаштування та інтеграція з платіжними системами.

Багато людей вважають програми для зчитування QR-кодів життєво важливими для щоденних завдань. Сканування кодів продуктів для порівняння цін або доступу до квитків на події ще ніколи не було таким простим. Однак деякі користувачі повідомляють про проблеми з несподіваним збоєм певних програм. Це дратує їх у вирішальні моменти. Плавна робота є важливою, особливо коли потрібна швидкість.

Поради: Переконайтеся, що програмне забезпечення вашого пристрою оновлено. Це може вирішити проблеми з продуктивністю. Завжди перевіряйте відгуки про програми, щоб отримати корисну інформацію. Добре оцінений зчитувач QR-кодів може заощадити час і уникнути клопоту. Пам'ятайте, що не кожна програма працює однаково добре. Вибирайте з розумом, щоб зробити сканування приємним.

Основні характеристики, які слід шукати в програмах для зчитування QR-кодів

Розглядаючи найкращі програми для зчитування QR-кодів на 2026 рік, виділяється кілька ключових особливостей. Зручний інтерфейс є важливим. Користувачі повинні мати можливість швидко сканувати коди без складних

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		14

налаштувань. Згідно з нещодавнім опитуванням, 78% користувачів віддають перевагу програмам, які вимагають мінімальних кроків для успішного сканування. Це підтверджує ідею про те, що простота приваблює більше користувачів.

Ще однією важливою особливістю є здатність програми обробляти різні типи QR-кодів. Деякі програми можуть зчитувати лише стандартні QR-коди, нехтуючи розширеними типами. Звіти показують, що програми, які можуть керувати різними форматами, утримують на 40% більше активних користувачів. Функції безпеки також важливі. Користувачі більше усвідомлюють потенційні ризики від шкідливих кодів. Близько 60% респондентів надають пріоритет програмам, які пропонують певну форму сканування безпеки.

Нарешті, інтеграція з іншими сервісами є важливим аспектом. Наприклад, програми, які підключаються до соціальних мереж або платіжних систем, покращують взаємодію з користувачем. Однак не всі програми для зчитування QR-кодів пропонують таку інтеграцію, що призводить до невдоволення користувачів. Фактично, огляд показав, що 55% користувачів видалили б програми, яким бракує безперебійної функціональності. Таким чином, балансування функціональності, зручності використання та безпеки залишається життєво важливим для розробників у ландшафті Qr Reader та Qr Reader Factorys.

Найкращі програми для зчитування QR-кодів 2026 року: порівняльний аналіз

У 2026 році програми для зчитування QR-кодів є важливими інструментами для швидкого доступу до інформації. Багато програм можуть похвалитися зручними інтерфейсами з можливостями швидкого сканування. Ці функції значно покращують взаємодію з користувачем. Однак не всі програми працюють однаково добре. Деяким бракує розширених функцій, що може бути неприємно.

Порівнюючи програми для зчитування QR-кодів, враховуйте такі характеристики: швидкість, точність та дизайн. Програма може перевершувати в

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		15

одній області, але поступатися в іншій. Наприклад, швидкість може бути вражаючою, але точність може розчаровувати. Цю невідповідність важливо враховувати під час вибору програми.

Поради щодо вибору зчитувача QR-кодів включають перевірку відгуків користувачів. Дізнайтеся, що інші говорять про надійність та функції. Добре збалансована програма також повинна пропонувати варіанти налаштування. Також майте на увазі, що деякі програми можуть бути перевантажені надмірною кількістю реклами. Чистий інтерфейс зазвичай призводить до кращої навігації. Завжди переконайтеся, що обрана вами програма відповідає вашим потребам сканування, залишаючись ефективною.

Відгуки користувачів розкривають важливі аспекти роботи програм для зчитування QR-кодів у 2026 році. Багато користувачів цінують простоту використання. Нещодавнє опитування показує, що 72% користувачів надають пріоритет простоті. Оптимізований інтерфейс часто призводить до більшої задоволеності. Однак деякі користувачі висловлюють розчарування через часті рекламні оголошення, які можуть переривати процес сканування. Балансування реклами та функціональності залишається проблемою для розробників.

Ще одним помітним аспектом є точність сканування. Відгуки користувачів показують, що 68% користувачів стикаються з випадковими помилками. Деяким програмам важко читати складні коди. Крім того, швидкість сканування значно варіюється залежно від програми. Швидший сканер, як правило, покращує утримання користувачів. Відгуки підкреслюють важливість надійності, особливо для бізнес-цілей. Деякі користувачі повідомляли про втрату часу через повільну реакцію.

Функції безпеки також залишаються гарячою темою. У багатьох оглядах згадуються занепокоєння щодо конфіденційності даних. Звіт показує, що 55% користувачів хочуть кращих заходів безпеки. Вони віддають перевагу додаткам, які пропонують зашифровані опції сканування. Користувачі прагнуть впевненості в обміні особистою інформацією. Вирішення цих проблем є життєво важливим

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		16

для розробників додатків, які стикаються з конкурентним ринком. Загалом, відгуки користувачів стимулюють еволюцію зчитувачів QR-кодів.

Показники продуктивності: швидкість, точність та простота використання

Продуктивність програм для зчитування QR-кодів зосереджена на трьох критичних показниках: швидкості, точності та простоті використання. Дослідження показують, що користувачі віддають перевагу програмам, які розшифровують QR-коди протягом однієї секунди, оскільки 75% користувачів відмовляються від повільних програм. Швидкість має вирішальне значення, особливо в місцях з високим трафіком. Точність також має значення; звіт показав, що 90% користувачів помічають помилки в сканованих даних. Забезпечення надійного зчитувача може суттєво вплинути на задоволення користувачів.

Якщо розглядати простоту використання, інтуїтивно зрозумілі інтерфейси отримують найвищі оцінки користувачів. Перевага надається програмам, які вимагають мінімальних кроків для сканування. Однак деякі користувачі повідомляють про розчарування щодо програм, яким бракує чітких інструкцій. Дослідження зазначило, що 40% користувачів видалили б програму, якби їм було важко в ній орієнтуватися. Відгуки користувачів підкреслюють цінність простого дизайну: менше кнопок і простіші інструкції створюють кращий досвід.

Поради: Вибирайте програми з можливостями швидкого сканування. Шукайте відгуки користувачів, які згадують високу точність і легкість навігації. Оптовий зчитувач QR-кодів може пропонувати унікальні функції, але не може компенсувати поганий користувацький досвід. Завжди тестуйте різні варіанти, щоб знайти найкраще рішення для ваших потреб. Пам'ятайте, що ефективний зчитувач QR-кодів може значно оптимізувати процеси.

Міркування безпеки для програм для сканування QR-кодів

Зростання популярності програм для сканування QR-кодів приносить зручність, проте виникають серйозні проблеми безпеки. Дослідження показують,

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		17

що понад 60% користувачів не перевіряють URL-адреси, що стоять за сканами. Кіберзлочинці користуються цим незнанням. Вони часто вбудовують шкідливі посилання в, здавалося б, нешкідливі QR-коди.

У звіті експертів з кібербезпеки зазначається, що 43% підприємств минулого року зазнали фішингових атак через QR-коди. Ця статистика підкреслює потенційні ризики. Користувачі можуть ненавмисно розкрити свої дані або стати жертвами шахрайства. Постійне навчання безпечним методам сканування є надзвичайно важливим.

Деякі зчитувачі QR-кодів містять вбудовані функції безпеки, такі як попередній перегляд URL-адрес. Однак багатьом з них бракує цих захистів. Різниця в якості програм викликає питання. Користувачі повинні бути пильними. Вони повинні шукати програми, які надають пріоритет безпеці. Тільки тоді переваги QR-кодів можна повністю реалізувати, не ставлячи під загрозу особисту інформацію.

Майбутні тенденції в технології та застосуванні зчитувачів QR-кодів

Майбутнє технології зчитування QR-кодів стрімко розвивається. Оскільки все більше підприємств впроваджують безконтактні рішення, зростає попит на інноваційні функції. Розширені можливості розпізнавання зображень є надзвичайно важливими. Користувачі хочуть швидкого сканування навіть в умовах низької освітленості. Покращене оптичне розпізнавання символів (OCR) також може змінити спосіб вилучення інформації зі сканованих кодів.

Окрім функціональності, важливим залишається і зручність користування. Інтерфейс має бути інтуїтивно зрозумілим та простим у навігації. Багато людей шукають налаштовувані параметри, що дозволяють користувачам змінювати теми та налаштування. Також зростають занепокоєння щодо конфіденційності. Користувачі очікують прозорості щодо використання даних. Мобільні додатки повинні відображати ці зміни та забезпечувати довіру користувачів.

Зрештою, хоча майбутнє виглядає світлим, проблеми залишаються. Зчитувачі QR-кодів повинні постійно оновлюватися, щоб йти в ногу з

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		18

тенденціями. Розробники повинні зосередитися на безперешкодній інтеграції з іншими технологіями. Це підвищить їхню загальну цінність. Регулярний зворотний зв'язок від користувачів може спрямовувати ці вдосконалення, забезпечуючи розвиток додатка відповідно до змінних потреб. Досягти правильного балансу між інноваціями та задоволенням користувачів буде непросто, але це важливо.

Основні характеристики та переваги зчитувачів QR-кодів для систем Android, Windows та Mac OS

Зчитувачі QR-кодів стали важливими інструментами для різних платформ, включаючи системи Android, Windows та Mac OS. Їхня універсальність підвищується такими функціями, як підтримка кількох варіантів підключення, таких як Bluetooth та бездротовий зв'язок 2.4G, що робить їх адаптованими до різних середовищ та уподобань користувачів. Звіт Міжнародної корпорації даних (IDC) показує, що впровадження технології мобільного сканування зросло більш ніж на 30% за останні кілька років, що підкреслює зростаючий попит на ефективні та надійні рішення для збору даних як у комерційних, так і в особистих умовах. Однією з видатних особливостей сучасних зчитувачів QR-кодів є їхній тривалий час роботи від акумулятора, який підтримується вбудованим акумулятором ємністю 2600 мАг, що забезпечує надзвичайно тривалий час роботи в режимі очікування. Крім того, ці пристрої підтримують різні режими сканування, включаючи ручний, безперервний та автоматичний індукційний, задовольняючи потреби різних користувачів та підвищуючи ефективність роботи. Найновіша технологія CMOS з глобальною експозицією затвора дозволяє користувачам декодувати різноманітні штрих-коди, включаючи пошкоджені, розмиті або відбиваючі – критичний аспект, зазначений у звіті MarketsandMarkets, який прогнозує, що ринок сканерів штрих-кодів досягне 3,5 мільярда доларів до 2027 року.

Крім того, ці пристрої розроблені зі зручними функціями, такими як індикатор заряду батареї для зменшення тривоги щодо низького заряду батареї та

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		19

кнопка живлення з однією клавішею для зручної роботи. Вони підтримують кілька мов клавіатури та режимів підключення, що підвищує зручність їх використання в різних географічних регіонах. Завдяки доступним опціям налаштування користувачі можуть налаштувати пристрій відповідно до своїх конкретних вимог до робочого процесу, що робить зчитувачі QR-кодів незамінними в сучасному швидкозмінному цифровому середовищі.

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

Python – динамічна інтерпретована об'єктно-орієнтована скриптова мова програмування із строгою динамічною типізацією. Офіційний сайт мови програмування Python <https://www.python.org/>. Python – багатоцільова мова програмування, яка дозволяє писати код, що добре читається. Відносний лаконізм мови Python дозволяє створити програму, яка буде набагато коротше свого аналога, написаного на іншій мові. Python – багатоплатформова мова програмування. Це означає, що програми на Python можна запускати в різних операційних системах без будь-яких змін.

Ще однією перевагою Python є його стандартна бібліотека, яка встановлюється разом з Python і містить готові інструменти для роботи з операційною системою, веб-сторінками, базами даних, різними форматами даних, для побудови графічного інтерфейсу програм тощо. Програми, написані на мові програмування Python, можуть бути як невеликими скриптами, так і складними системами. Python абсолютно безкоштовний.

Швидкість виконання коду Python

Один з можливих недоліків Python – швидкість виконання коду. Python не є компільованою мовою. Код на Python спочатку компілюється у внутрішній байт-код, який потім виконується інтерпретатором Python. У більшості випадків

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		20

при використанні Python виходять програми повільніші в порівнянні з такими мовами, як С.

Втім, сучасні комп'ютери мають таку обчислювальну потужність, що для більшості застосунків швидкість розробки важливіша швидкості виконання, а програми на Python зазвичай пишуться набагато швидше.

Окрім того, Python легко розширюється модулями, написаними на С або С++. Такі модулі можуть використовуватися для виконання частин програми, що створюють інтенсивне навантаження на процесор.

Використання Python

Python використовується для різних цілей: для створення ігор і веб-застосунків, розробки внутрішніх інструментів для різноманітних проектів. Мова також широко застосовується в науковій області для досліджень і розв'язування прикладних завдань.

Застосування мови програмування Python:

1. BitTorrent – протокол для обміну даними.
2. Ubuntu Software Center – вільне програмне забезпечення для пошуку, установки і видалення пакунків в системі Ubuntu Linux.
3. Blender – програма для створення тривимірної комп'ютерної графіки, що включає засоби моделювання, анімації, вимальовування, пост-обробки відео, а також створення відеоігор.
4. GIMP – растровий графічний редактор, із підтримкою векторної графіки.
5. World of Tanks.
6. Вільна енциклопедія Вікіпедія.
7. Пошукова система Google.
8. DropBox – файловий хостинг, що включає персональне хмарне сховище, синхронізацію файлів і програму-клієнт.
9. YouTube – популярне відеосховище.

Версії Python

Мови програмування з часом змінюються – розробники додають в них нові можливості, а також виправляють помилки. Так з'являються різні версії мови. Наприклад, код написаний на Python 2 у більшості випадків не буде працювати у версії Python 3 без внесення додаткових змін.

Процесор є найважливішим компонентом в комп'ютері. Одна з основних функцій процесора – це обробка даних згідно комп'ютерної програми, яка є списком інструкцій, шляхом виконання арифметичних і логічних операцій над фрагментами даних.

Кожна інструкція в програмі – це команда, яка «повідомляє» процесору, яку операцію він повинен виконати. Процесор комп'ютера може розуміти лише ті інструкції, які написані на машинній мові. Машинна мова – це штучна мова, створена для передачі команд комп'ютеру. За допомогою машинної мови створюються ефективні програми, оскільки розробник отримує доступ до всіх можливостей процесора. Машинна мова – мова низького рівня.

Інструкція машинної мови існує для кожної операції, яку процесор здатний виконати – є інструкція для додавання чисел, є інструкція для віднімання чисел і т.д. Увесь набір інструкцій, який центральний процесор може виконати, відомий як набір інструкцій процесора.

Наприклад, у вас є певна програма, яка зберігається на диску вашого комп'ютера. Для виконання програми, ви здійснюєте подвійний клік на значку програми. Це змушує програму копіюватися з диска в оперативну пам'ять, після чого процесор комп'ютера виконує копію програми, яка знаходиться в оперативній пам'яті.

Коли процесор виконує інструкції програми, він бере участь у процесі, який є відомим як цикл `fetch – decode – execute` (отримати – декодувати – виконати). Цей цикл виконується для кожної інструкції у програмі і складається з трьох кроків:

Отримати

Програма – це послідовність інструкцій на машинній мові. Першим кроком циклу є завантаження (отримання) наступної інструкції з пам'яті в процесор.

Декодувати

Інструкція машинної мови – це двійкове число, яке представляє команду, що повідомляє процесору виконати певну операцію. На цьому кроці процесор декодує інструкцію, яку було «витягнуто» з пам'яті, для визначення того, яка операція повинна виконуватись.

Виконати

Останній крок циклу – виконати операцію.

Хоча процесор комп'ютера розуміє тільки машинну мову, людині непрактично писати програми на машинній мові. Така програма може мати тисячі або навіть мільйони бінарних інструкцій, і написання такої програми буде дуже обтяжливим процесом.

З цієї причини була створена мова асемблера як альтернатива машинній мові. Замість використання двійкових чисел для написання інструкцій, мова асемблера використовує короткі слова, відомі як мнемокоди.

Незважаючи на те, що мова асемблера не вимагає двійкових інструкцій, як у випадку машинної мови, проте вона вимагає високих знань про процесор. Використовуючи мову асемблера, навіть для найпростішої програми, необхідно написати велику кількість інструкцій.

Мова програмування високого рівня дозволяє створювати складні програми, не знаючи, як працює процесор, і не записуючи великої кількості інструкцій низького рівня. Крім того, більшість мов програмування високого рівня використовують слова, які легко зрозуміти.

Python – одна із популярних сучасних мов програмування високого рівня. Python – інтерпретована мова програмування. Python – це високорівнева інтерпретована мова програмування, на відміну від C++, яка є прикладом

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		23

компільованої мови програмування. Назва Python відноситься як до мови програмування, так і до інтерпретатора – комп'ютерної програми, яка зчитує початковий код (написаний на Python) і виконує інструкції (команди).

Для перекладу мови високого рівня на машинну мову доступні два типи програм:

1. Компілятор.
2. Інтерпретатор.

Завантаження Python

Версії інтерпретатора Python для різних операційних систем доступні для безкоштовного завантаження за адресою <https://www.python.org/downloads>.

Середовище програмування для Python

Для написання програм використовують текстові редактори або інтегровані середовища розробки, які включають в себе різні інструменти для роботи з кодом: засіб для написання коду (текстовий редактор), інтерактивний інтерпретатор, відлагоджувач тощо.

Текстові редактори та інтегровані середовища програмування для Python:

- IDLE – стандартний редактор Python. Встановлюється разом з Python для користувачів Windows, окремим пакунком для користувачів Linux.
- Notepad++ – безкоштовний текстовий редактор початкового коду, який підтримує велику кількість мов, в тому числі і Python. Лише для користувачів Windows.
- Visual Studio Code – це легкий, але потужний редактор початкового коду, який розповсюджується безкоштовно і доступний у версіях для платформ Linux, Windows і macOS.
- PyScripter – інтегроване середовище розробки для мови програмування Python. Для користувачів Windows. Поширюється безкоштовно.
- Wing IDE 101 – вільне інтегроване середовище для Python, розроблене для навчання програмістів-початківців. Для користувачів Linux, Windows і macOS. Поширюється безкоштовно.

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		24

– Geany – вільний текстовий редактор з базовими елементами інтегрованого середовища розробки, доступний для операційних систем Linux, Windows і macOS.

– PyCharm – інтегроване середовище розробки для мови програмування Python. PyCharm є власницьким програмним забезпеченням. Наявна безкоштовна версія Community з усіченим набором можливостей. Для користувачів Linux, Windows і macOS.

– Thonny – IDE для вивчення програмування мовою Python. Для користувачів Linux, Windows і macOS.

– Mu – редактор коду Python для програмістів-початківців. Для користувачів Linux, Windows і macOS.

2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випускню кваліфікаційну роботу за першим (бакалаврським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи обробки даних двовимірних матричних штрихкодів типу Data Matrix ECC 200.

В процесі розробки випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти необхідно виконати наступний обсяг роботи:

а) провести аналіз існуючих систем-аналогів для виявлення їх позитивних і негативних якостей. Результати аналізу врахувати в подальших розробках;

б) вибрати та обґрунтувати методику побудови системи контролю роботи технологічного обладнання на виробництві в автоматизованому режимі. Розробити функціональну та структурну схеми системи;

в) розробити програмне забезпечення системи, що дозволить реалізувати поставлену технічним завданням задачу. Побудувати блок-схеми алгоритмів програми та підпрограми;

г) організувати інтерфейс користувача з метою формування та виводу на

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		25

екран ЕОМ повідомлень про некоректні дії користувача та нестандартні ситуації в роботі технологічного обладнання;

д) розробити рекомендації по організаційних та методичних заходах, які забезпечать впровадження системи в промислову експлуатацію та її подальшу успішну експлуатацію;

е) провести розрахунки по визначенню економічної ефективності розробленої системи;

ж) розробити заходи по охороні праці при впровадженні та експлуатації системи, а також розробити заходи з цивільного захисту;

з) сформулювати висновки про виконаний обсяг робіт та одержані результати.

КБПЗ-2026

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		26

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

Що таке DataMatrix

DataMatrix – двовимірний матричний ідентифікатор у вигляді квадрата або прямокутника, призначений для кодування інформації про состав продукції, виробнику й іншому. Його наносять на папір, блістери (для лікарських препаратів), тверді поверхні.

Штрихкод утворений парною кількістю рядів і стовпців, що складаються із квадратів: білих і чорним або одноколірних з різним ступенем яскравості. Елементи можуть бути й круглими – у стандарті ISO не зазначена конкретна форма.

Звичайні розміри квадратного DataMatrix 10х 10-144х144 модулів, а прямокутного – 8х 18-16х48 модулів.

DataMatrix ECC200 і система GS1

ECC200 – новітня версія штрихкоду DataMatrix. Тільки вона працює в GS1 – міжнародній системі штрихового кодування й ідентифікації. Після підписання квітневих урядових розпоряджень генерацією ідентифікаторів відає Центр розвитку перспективних технологій (далі ЦРПТ, Центр або Оператор). GS1 привласнює товару обліковий номер – номенклатуру для певної товарної групи.

Розшифровка DataMatrix ECC200 виконується навіть при 30-процентному ушкодженні елементів. Це можливо завдяки системі Ріда-Соломона, що дозволяє відновити стерті відомості й знижує ймовірність появи помилки до 1 на 10 млн сканованих символів.

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		27

Чим відрізняється QR-код від DataMatrix

В електроніці, розважальній і рекламній сферах, промисловості для шифрування інформації про вироби застосовують Data Matrix або QR-код. Це 2 основних шифри, у яких утримуються відомості про марку, дату виготовлення, E-mail-Адреса й інші дані. QR-code – обов'язковий реквізит чеків, які друкують сучасні онлайн-каси.

DataMatrix і QR-code – двовимірні штрихкоди, які складаються із чорно-білих квадратних (в DataMatrix можливо круглих) елементів. Вони дозволяють зашифрувати до 3 Кбайт інформації. Зчитуються навіть при 30-процентному ушкодженні пікселів. Обидва штрихкоди генеруються безкоштовно по стандартах ISO, відкритим для вільного користування. Однак безкоштовного посібника з кодування не існує. Документ можна придбати на офіційному ресурсі організації ISO.

Основні характеристики ідентифікаторів збігаються. Але в пробних проектах для маркування тютюнових виробів, лікарських препаратів і взуття виготовлювачі використовують DataMatrix. Чим же він відрізняється від QR-коду?

Таблиця 3.1 – Порівняння кодів

Характеристика	Data Matrix	QR-код
Розмір (пікселі)	Мінімальний – 10x10; Максимальний – 144x144	Мінімальний – 21x21; Максимальний – 177x177
Форма	Квадратна, прямокутна	Квадратна
Кількість стовпців і рядів	Парне	Непарне
Стандарт	ISO/IEC16022	ISO/IEC18004
Розпізнавання в просторі	Під довільним кутом	Під довільним кутом, у дзеркальному відбитті

Основна перевага DataMatrix, як обов'язкового товарного знака – компактний розмір. Штрихкод DataMatrix міститься на тонких пачках сигарет шириною близько 14 мм.

3.2 Розробка структурної схеми

На рисунку 3.1 зображено розроблену структурну схему системи обробки даних двовимірних матричних штрихкодів типу Data MatrixECC 200.

Розглянемо схему згори до низу. Спочатку проходить введення даних користувача, а саме даних які будуть вбудовані у двовимірний матричний штрихкод типу Data MatrixECC 200. При необхідності проводиться зміна налаштувань генерації двовимірного матричного штрихкоду типу Data MatrixECC 200 по замовчанню.

Далі проводиться кодування даних користувача з використання існуючих налаштувань кодування. Після цього проводиться формування кодів корегування тобто дублювання даних для поміхо стійкості з використанням модулю генерації кодів корегування.

Після цих дій проводиться застосування маски та додавання версії штрих-коду. Версії штрих-коду дозволяє при розпізнаванні двовимірного матричного штрихкоду типу Data MatrixECC 200 користувачем більш точно провести декодування.

Та на останньому етапі проводиться генерація двовимірного матричного штрихкоду типу Data MatrixECC 200 з використанням модуля створення JPG файлів, що у кінцевому результаті дає повноцінний згенерований двовимірний матричний штрихкод типу Data MatrixECC 200.

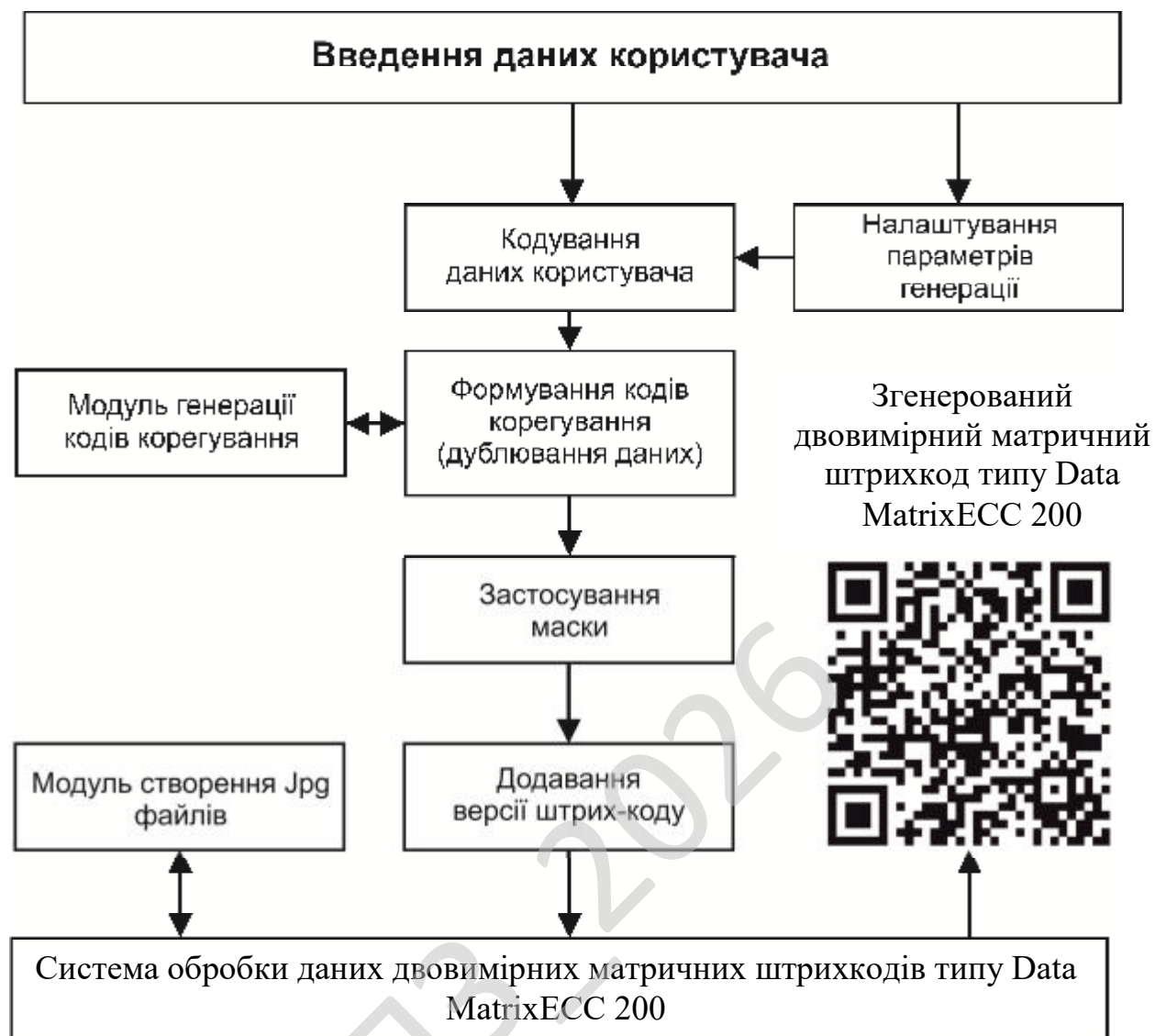


Рисунок 3.1 – Структурна схема системи

Товари, що підлягають маркуванню DataMatrix

В 2017-2018 роках почалося експериментальне маркування лікарських препаратів, виробів з тютюну, коштовностей і взуття. Ці проекти – підготовка до кодування даних про товарні групи, що почнеться в 2020 році. Обов'язкове маркування – друк на впакуванні унікального 2D-коди у форматі DataMatrix. Ідентифікатор дозволить відстежити шлях товару від виготовлювача до кінцевого покупця.

Як працює маркування штрихкодом DataMatrix

Відповідно до нормативного акту, маркировочні номери створюються централізовано. Права для генерації ідентифікаторів передані ЦРПТ.

Для кодування використовують криптозахищені коди DataMatrix. Тобто штрихкод утвориться із двох компонентів:

- ідентифікатора, що внесений у перелік продукції й повідомляє про дату виготовлення, строку придатності, найменуванні й іншому (сполучення номера товару по GS1 і номера ідентифікації);
- перевірного коду, сформованого за допомогою криптотехнологій і який забезпечує захист паперових або нанесених на інші поверхні DataMatrix від повторного копіювання.

Генерація 2D-коду – платна послуга. У вартість входить створення захищеного шифру й реєстрація даних на інформаційному ресурсі з моменту випуску продукту до його реалізації кінцевому споживачеві.

Основні етапи функціонування системи кодування:

- компанія-виробник направляє Операторові запит про створення ідентифікатора;
- оператор генерує унікальний криптозахищений штрихкод і передає його організації-виробникові;
- при виготовленні продукції зашифровану інформацію наносять на впакування, ярлик або тару;
- 2D-код сканують при імпорті, оптовому продажі, роздрібній реалізації, переміщенні, списанні;
- відомості про рух одиниці відбиваються в спеціальній інформаційній базі до моменту списання або реалізації такий кінцевому покупцеві, після чого шифри назавжди вилучають із оберту – штрихкоди не повторюються.

Дані про кінцевий продаж попадають із онлайн-каси до оператора фіскальних даних, що передає їх у єдину систему.

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		31

ЦРПТ і Чесний знак: інтеграція даних DataMatrix

Оператором маркування призначений Центр розвитку перспективних технологій:

Оператор відповідає за:

- генерацію, криптозахист, облік і перевірку кодів;
- зберігання й обробку відомостей, надаваних виробниками й іншими учасниками товарообігу;
- створення й ведення федерального каталогу маркірованих товарних груп;
- створення загальної інформаційної системи.

Для інтеграції відомостей про одиниці, маркірованих DataMatrix, ЦРПТ створив Чесний знак. Це система для відстеження руху товарних позицій зі штрихкодами, що дозволить:

- рядовому громадянину здобувати якісну, легальну продукцію;
- виробникові захистити бізнес і бренд від підробок, збільшити товарообіг і виторг;
- державі контролювати виробничі процеси, повноту й правильність нарахування, сплати податків і зборів.

Споживачі зможуть перевіряти дійсність здобувається продукції, що, за допомогою мобільного застосунку «Чесний ЗНАК». Для цього необхідно:

- скачати утиліту в App Store і Google Play і встановити її на мобільний гаджет;
- знайти на впакуванні або тарі цифровий код;
- відсканувати знак за допомогою камери телефону;
- перевірити дані через застосунок.

Якщо на екрані гаджета після перевірки з'явиться повідомлення про те, що інформація коректна, значить товар виготовлений легально, а його якість відповідає заявленому підприємством-виробником.

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		32

Якщо утиліта повідомить про помилку – одиниця контрафактна. У цьому випадку покупець може повідомити про підробку в режимі онлайн, використовуючи можливості застосунку.

Одна із ключових складових Чесного знака – постійно оновлюваний перелік товарів. Каталог доступний у режимі онлайн. Виробники й імпортери наповнюють перелік добровільно. Внесення в документ нових відомостей безкоштовно.

Тютюнова продукція із цифровим кодом Data Matrix

На початку 2019 року стартував пілотний проект кодування інформації про вироби з тютюну, що протриває до кінця року. За даними Центра в експерименті взяли участь 256 компаній-виробників сигарет, касової техніки, програмного забезпечення для організації електронного документообігу, операторів фіскальних даних.

Перша тютюнова продукція із цифровим кодом Data Matrix на впакуванні випущене в лютому 2019 р. Для правильного зчитування й розшифровки ідентифікатора виконана модернізація торговельної техніки. Приклади DataMatrix code можна побачити на продукції Imperial Tobacco, Philip Morris International і інших виготовлювачів.

Вимоги до коду

Унікальний шифр наносить організація-виробник або імпортер цигарок, сигарет. Код друкують на пачках, блоках, ящиках, контейнерах. Законодавство не обмежує розміри впакування.

Складові маркування:

- міжнародний товарний шифр GTIN;
- номер, що дозволяє ідентифікувати позицію в єдиному каталозі;
- перевірочний код, сформований за допомогою криптографічних технологій.

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		33

DataMatrix code на одязі й взутті

Обов'язкове маркування взуття за допомогою DataMatrix почнуть із грудня 2026 р. Із цього ж моменту почнуть використовувати й ідентифікатори для одягу. За станом на грудень 2025 р. їхній формат не визначений: можливо це будуть чипи, можливо матричні штрихкоди.

У червні 2025 р. запущена пілотна програма по нанесенню унікальних шифрів на взуття. Проект виконується по загальному механізмі: присвоєння ідентифікатора, внесення інформації в онлайн-каталог, відстеження руху продукції в Чесному знаку.

Експеримент із взуттям завершиться наприкінці другого кварталу 2026 року.

Устаткування торговельної точки для обробки кодів Data Matrix

До 2027 року маркування стане обов'язковим для всіх товарних груп: продуктів, текстилю, предметів для дітей і іншого.

Для обробки й друку матричних кодів на чеках необхідний ручний, бездротовий, стаціонарний або інший 2D-сканер і сучасна онлайн-каса, програмне забезпечення якої підтримує:

- формат фіскальних документів;
- роботу із сервісом Центра.

Інтеграція онлайн-каси з товарообліковою програмою, наприклад, «Мій склад», дозволить налаштувати електронний документообіг і полегшить роботу з маркірованими одиницями.

3.3 Розробка функціональної схеми

На рисунку 3.2 зображена функціональна схема системи. На схемі детально розглянуто результат роботи розробленої програми – згенерований двовимірний матричний штрихкоду типу Data MatrixECC 200.

На наведеному прикладі закодовано назву учбового закладу – текст

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		34

«Центральноукраїнський національний технічний університет» (докладніше у розділі 5).

Для розпізнавання створеного двовимірного матричного штрихкоду типу Data MatrixECC 200 існує велика кількість сторонніх програм (розглянуті у розділі 2). При отриманні цього коду вони шукають створені області детектування та центрування.

Після цього зчитують системну інформацію. Яка представляє собою графічно представлений бітовий масив.

Та усіваючи все зайве за допомогою маски отримують кінцеві дані – текст «Центральноукраїнський національний технічний університет».



Рисунок 3.2 – Функціональна схема системи

Розглянувши усі блоки функціональної схеми перейдемо до розгляду діаграми взаємодії процесів, які відбуваються у системі.

3.4 Розробка діаграми процесів

Розглянемо розроблену діаграму процесів яка зображена на рисунку 3.3. Основна будова діаграми процесів полягає у графічному представленні складу сукупностей даних, що характеризуються як співвідношення різних частин кожної з сукупностей.

Склад статистичної сукупності графічно може бути представлений як за допомогою абсолютних, так і відносних показників.

Графічне зображення складу сукупності по абсолютними і відносними показниками сприяє проведенню більш глибокого аналізу і дозволяє проводити аналіз системи.

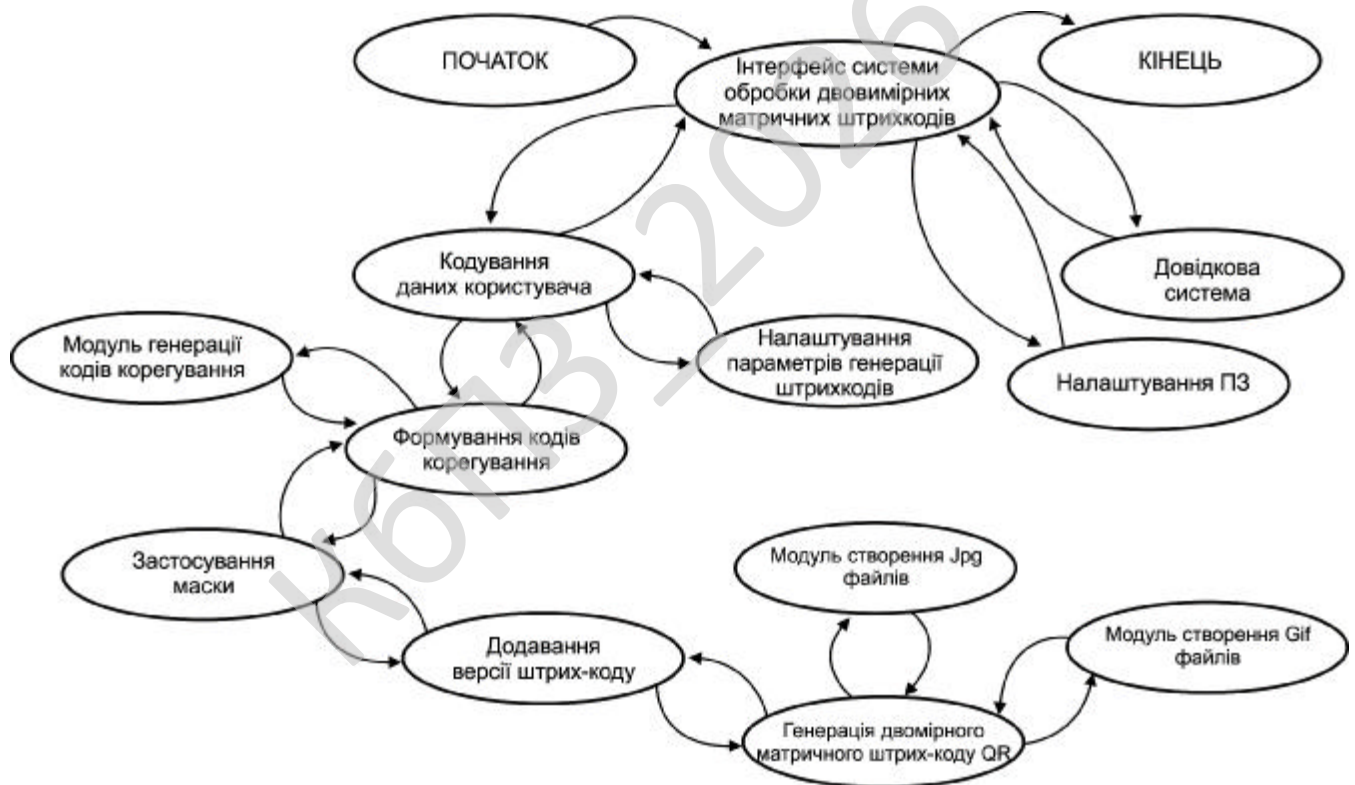


Рисунок 3.3 – Діаграма взаємодії процесів

Діаграма взаємодії процесів використовується для візуалізації процесів обробки даних (структурне проектування).

Для розробника вважається звичним спочатку креслити діаграму взаємодії процесів даних рівня контексту, завдяки чому буде показано взаємодію системи.

Ця діаграма в подальшому підлягає уточненню шляхом деталізації процесів та потоків даних з метою показати систему що розробляється.

При детальному її розгляді можна побачити як саме проходить взаємодія у розробленій системі.

Використовується модель проектування, графічне представлення «потоків» даних в інформаційній системі.

Діаграми потоків даних містять чотири типи елементів:

- Зовнішні по відношенню до системи сутності.
- Потоки даних між елементами трьох попередніх типів.
- Процеси які являють собою трансформацію даних в рамках описуваної системи.
- Сховища даних (репозиторії).

Таким чином, розглянувши опис системи, структурну, функціональну схеми системи, та діаграму взаємодії процесів перейдемо до опису блок-схем основної програми, та підпрограм, які використовуються, для реалізації системи.

4 РЕАЛІЗАЦІЯ ПРОЕКТУ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ПРАВИЛЬНІСТЬ ПРОЕКТНИХ РІШЕНЬ

4.1 Блок-схеми та опис алгоритмів функціонування системи

Розглянемо реалізацію бакалаврської дипломної роботи. Були проведені розрахунки і підібрані набори тестових даних для перевірки правильності реалізації проектних рішень.

Було створено блок-схеми роботи системи. Перед їх розглядом необхідно провести роз'яснення який саме тип блок-схем використовується.

Блок-схема це представлення задачі для її аналізу або розв'язування за допомогою спеціальних символів (геометричних образів), які позначають такі елементи, як операції, потік, дані тощо.

Блок вхідних та вихідних даних прийнято позначати паралелограмом, блок обчислень (обробки) даних – прямокутником, блок прийняття рішень – ромбом, еліпсом – початок та кінець алгоритму.

У інформаційних технологіях функціональна схема складається з функціональних блоків, які являють собою конструктивно відособлені частини (елементи або пристрої) автоматичних систем, які виконують певні функції. Функціональні блоки на схемі позначають прямокутниками, всередині яких надписують їх найменування відповідно до функцій, що виконуються.

Зв'язки між функціональними блоками (внутрішні впливи) позначаються лініями зі стрілками, які вказують напрям впливів.

Функціональні схеми можуть виконуватися в укрупненому і розгорненому вигляді. У першому випадку на схемі зображають найважливіші блоки системи і зв'язки між ними.

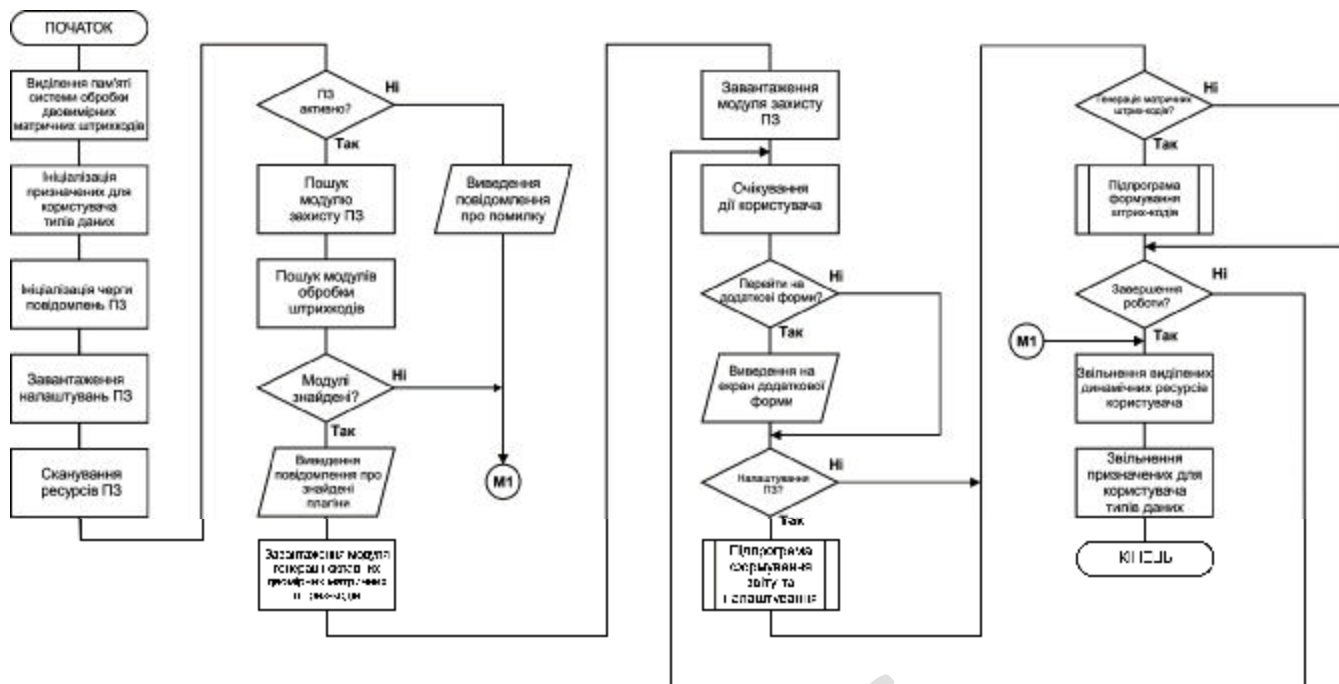


Рисунок 4.1 – Блок-схема основної програми

У другому варіанті схема відображається більш детально, що полегшує її читання та ілюструє принцип роботи.

Основні елементи схем алгоритму це термінатор, процес, рішення, зумовлений процес (підпрограма), дані та з'єднувач. Термінатор це елемент відображає вхід із зовнішнього середовища або вихід з неї (найчастіше застосування – початок і кінець програми). В середині фігури записується відповідна дія.

Процес це виконання однієї або кількох операцій, обробка даних будь-якого виду (зміна значення даних, форми подання, розташування). В середині фігури записують безпосередньо самі операції. Рішення це показує рішення або функцію перемикального типу з одним входом і двома або більше альтернативними виходами, з яких тільки один може бути обраний після обчислення умов, визначених всередині цього елементу.

Вхід в елемент позначається лінією, що входить зазвичай у верхню вершину елементу. Якщо виходів два чи три то зазвичай кожен вихід позначається лінією, що виходить з решти вершин (бічних і нижній).

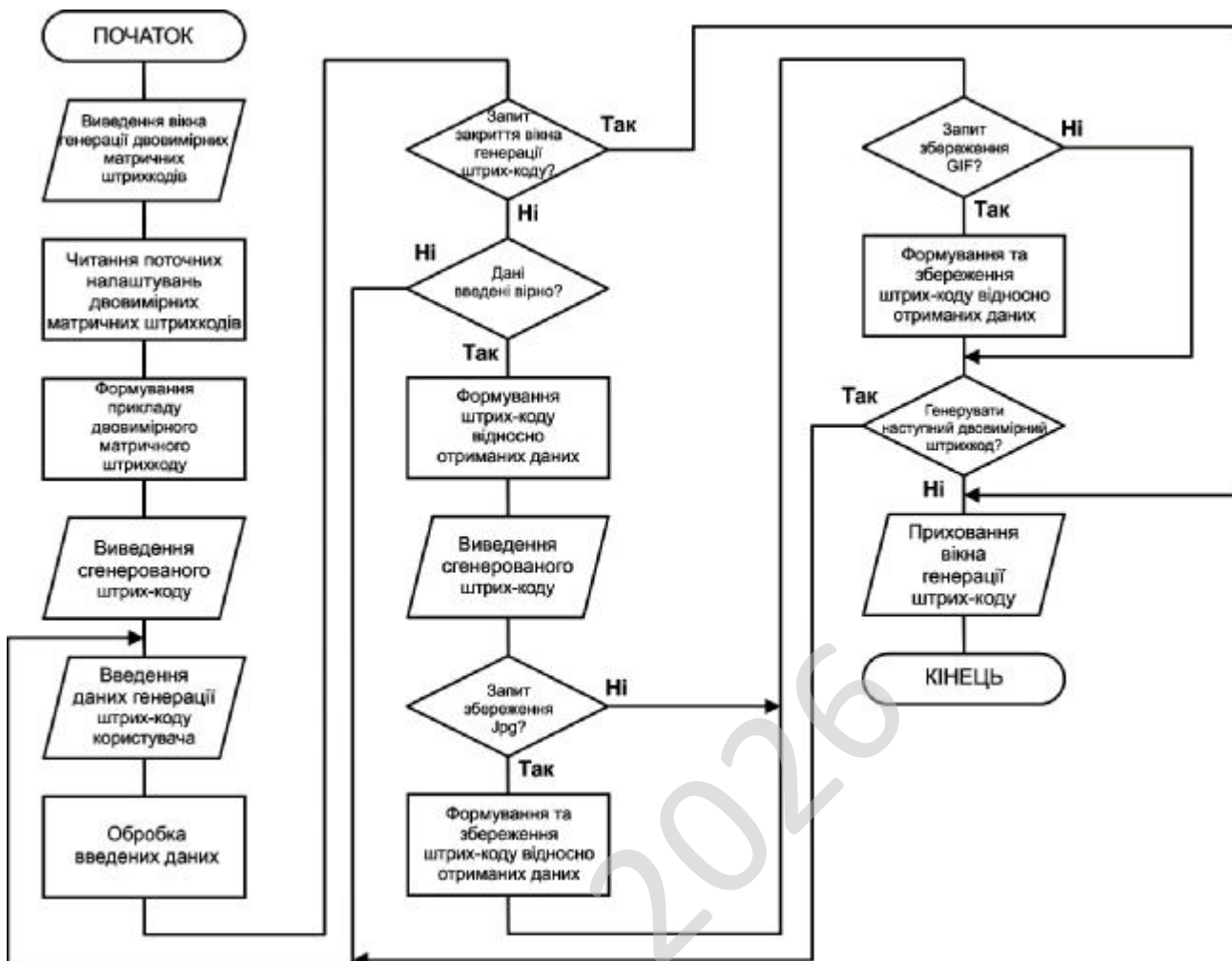


Рисунок 4.2 – Блок-схема роботи підпрограми

Якщо виходів більше трьох, то їх слід показувати однією лінією, що виходить з вершини (частіше нижній) елемента, яка потім розгалужується.

Відповідні результати обчислень можуть записуватися поруч з лініями, що відображають ці шляхи. Зумовлений процес (підпрограма) це символ відображає виконання процесу, що складається з однієї або кількох операцій, що визначені в іншому місці програми (у підпрограмі, модулі).

Всередині символу записується назва процесу і передані в нього дані. Дані це перетворення у форму, придатну для обробки (введення) або відображення результатів обробки (виведення). Цей символ не визначає носія даних (для вказівки типу носія даних використовуються специфічні символи).

З'єднувач це символ відображає вихід в частину схеми і вхід з іншої частини цієї схеми.

Використовується для обриву лінії та продовження її в іншому місці (приклад: поділ блок-схеми, що не поміщається на листі). Відповідні сполучні символи повинні мати одне (при тому унікальне) позначення.

Блок-схеми показують весь процес роботи системи з підсистемами та частково доказують правильність вибраних проектних рішень. Тому від точності і детальної блок-схеми залежить результат всієї програми.

При виборі початкової точки відліку при побудові схем було враховано, що виходячи з вибору мови програмування і інших технічних засобів, програма буде об'єктно-орієнтована що вимагає високого рівня декомпозиції задач на класи.

На рисунку 4.1 зображена основна блок-схема програми, на рисунку 4.2 зображено роботу підпрограми.

З яких видно що робота основної програми складається з початкових етапів ініціалізації ПЗ, перевірки наявності ресурсів системи, блоку початку основного циклу з чеканням запиту від користувача в якому відбувається виклик підсистеми та останньої стадії – перевірка поточного стану з завершенням роботи розробленого ПЗ.

При роботі підпрограми виконується основний функціонал системи з циклічними послідовностями, перевіркою поточного стану та поверненням в основну програму прапорів стану виконання.

Redmine – вільне серверне ПЗ для управління проектами та відстежування помилок. До системи входить календар-планувальник та діаграми Ганта для візуального представлення ходу робіт за проектом та строків виконання.

Redmine написано на мові Ruby і є ПЗ розробленим з використанням відомого веб-фреймворку Ruby on Rails, що означає легкість в розгортанні системи та її адаптації під конкретні вимоги. Для кожного проекту можна вести свої вікі та форуми.

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		41

Функціональні можливості:

- Ведення декількох проектів.
- Гнучка система доступу з використанням ролей.
- Система відстеження помилок.
- Діаграми Ганта та календар.
- Ведення новин проекту, документів та управління файлами.
- Сповіщення про зміни за допомогою RSS-потоків та електронної пошти.
- Власна Wiki для кожного проекту.
- Форуми для кожного проекту.
- Облік часових витрат.
- Налаштування власних (custom) полів для задач, затрат часу, проектів та користувачів.

– Легка інтеграція із системами керування версіями (SVN, CVS, Git, Mercurial, Bazaar и Darcs).

- Створення записів про помилки на основі отриманих листів
- Підтримка LDAP автентифікації.
- Можливість самореєстрації нових користувачів.
- Багатомовний інтерфейс (у тому числі українська мова).
- Підтримка СКБД: MySQL, PostgreSQL, SQLite.

Діаграма Ганта (Gantt chart, також стрічкова діаграма, графік Ганта) – це популярний тип діаграм, який використовується для ілюстрації плану, графіка робіт за будь-яким проектом. Є одним з методів планування та управління проектами.

Діаграма Ганта являє собою відрізки (графічні плашки), розміщені на горизонтальній шкалі часу. Кожен відрізок відповідає окремому завданню або підзадачі. Завдання і підзадачі, складові плану, розміщуються по вертикалі. Початок, кінець і довжина відрізка на шкалі часу відповідають початку, кінцю і тривалості завдання. На деяких діаграмах Ганта також показується залежність між завданнями.

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		42

Діаграма може використовуватися для представлення поточного стану виконання робіт: частина прямокутника, що відповідає завданню, заштриховується, відзначаючи відсоток виконання завдання; показується вертикальна лінія, що відповідає моменту «сьогодні».

Часто діаграма Ганта використовується спільно з таблицею зі списком робіт, рядки якої відповідають окремо взятій задачі, зображеній на діаграмі, а стовпці містять додаткову інформацію про задачу.

Система відстеження помилок Багтрекер – прикладна програма для допомоги розробникам програмного забезпечення (програмістам, тестувальникам тощо) враховувати і контролювати помилки, знайдені у програмах, питання щодо функціональності, рішення та оновлення, побажання користувачів, а також стежити за процесом їх виконання.

Кожному, хто розробляв програмні продукти, добре знайоме співвідношення «20/80» – останні 20 % роботи тривають 80 % часу.

Як це не парадоксально, але нічого дивного в цій пропорції немає, адже саме на завершальній стадії починається тестування проекту, коли виявляються помилки, і що більший проект, то більше буде знайдено помилок.

Водночас досить часто виявляється, що більшість цих помилок були відомі та могли бути виправлені з меншими витратами на попередніх стадіях роботи, але не були вчасно описані, а потім загубилися серед інших важливих завдань.

Отже, система відстеження помилок у найпростішому варіанті – це процес, що включає в себе виявлення помилки, її опис, виправлення і перевірку цього виправлення, тобто процес «стеження» за багом протягом всього як його життєвого циклу, так і життєвого циклу розробки в цілому.

Сукупність інформації про дефект. Головний компонент такої системи – база даних, що містить відомості про виявлені дефекти. Ці відомості можуть включати в себе:

- номер (ідентифікатор) дефекту;
- хто повідомив про дефект;
- дата і час виявлення дефекту;
- версія продукту, в якій виявлено дефект;
- серйозність (критичність) дефекту та пріоритет рішення;
- опис кроків для відтворення дефекту (неправильної поведінки програми);
- відповідальний за усунення дефекту;
- обговорення можливих рішень та їх наслідків;
- поточний стан виправлення дефекту;
- версії продукту, в якій дефект виправлений.

Крім того, розвинені системи надають можливість прикріплювати файли, які допомагають описати проблему, наприклад, дамп пам'яті або скріншот.

Використання. Основна перевага систем відстеження помилок полягає в забезпеченні чітких централізованих оглядів, запитів на розробку (включаючи помилки і виправлення) та їх стан. У корпоративному середовищі, системи відстеження помилок можуть бути використані для генерації звітів по продуктивності програмістів виправлення помилок. Однак, це може іноді приводити до неточних результатів, тому що різні помилки можуть мати різні ступені пріоритету та серйозності, що пов'язано з складністю їх фіксації.

Життєвий цикл дефекту. Як правило, система відстеження помилок використовує той чи інший варіант «життєвого циклу» помилки, стадія якого визначається поточним станом помилки.

Типовий життєвий цикл дефекту:

1. Новий – дефект зареєстрований тестувальником.
2. Призначений – призначений відповідальний за виправлення дефекту.

3. Дозволений – дефект переходить назад у сферу відповідальності тестувальника. Як правило, супроводжується резолюцією, наприклад:

- Виправлено (виправлення включені у версію таку-то).
- Дубль (повторює дефект, що вже знаходиться в роботі).
- Не виправлено (працює відповідно до специфікації, має занадто низький пріоритет, виправлення відкладено до наступної версії тощо).
- «В мене все працює» (запит додаткової інформації про умови, в яких дефект проявляється).

4. Далі тестувальник проводить перевірку виправлення, залежно від чого дефект або знову переходить у стан «Призначений» (якщо він описаний як виправлений, але не виправлений), або у стан «Закрито».

5. Відкрито повторно – дефект знайдено знову в іншій версії.

Система може надавати адміністраторові можливість налаштування користувачі, які можуть переглядати і редагувати помилки залежно від їх стану, переводити їх в інший стан або видаляти.

У корпоративному середовищі, система відстеження помилок може використовуватися для отримання звітів, що показують продуктивність програмістів при виправленні помилок. Однак, часто такий підхід не дає достатньо точних результатів через те, що різні помилки мають різну ступінь серйозності та складності. При цьому серйозність проблеми прямо не стосується складності її усунення.

Було використано метод розробки динамічних систем (Dynamic Systems Development Method, DSDM) – це головним чином методика розробки програмного забезпечення, що базується на концепції швидкої розробки застосунків (Rapid Application Development, RAD). У 2007 році DSDM став основним підходом до управління проектом і розробки застосунків. DSDM – це ітеративний і інкрементний підхід, який надає особливого значення тривалого участі в процесі користувача/споживача.

Мета методу – здати готовий проект вчасно і вкластися в бюджет, але в той же час регулюючи зміни вимог до проекту під час його розробки. DSDM входить в сімейство гнучкої методології розробки програмного забезпечення, а також розробок, що не входять у сферу інформаційних технологій.

Остання версія DSDM називається DSDM Atern. Назва Atern – це скорочення від Arctic Tern (пер. Полярна крачка).

Полярна крачка – птах, яка може подорожувати на великі відстані. Вона символізує безліч аспектів методу, наприклад визначення пріоритету або кооперування, які є природним способом ведення робочого процесу.

Попередня версія DSDM (випущена в травні 2003 року), яка все ще діє і широко використовується, – це DSDM 4.2, є дещо розширеною версією DSDM 4. Розширена версія містить настанови щодо того, як використовувати DSDM спільно з Екстремальним програмуванням (Extreme Programming).

Як розширення концепції швидкої розробки застосунків, DSDM фокусується на проектах інформаційних систем, що характеризуються стислими термінами і бюджетами. У DSDM присутні вказівки на типові помилки проектів інформаційних систем, таких як перевищення бюджету, запізнення з термінами здачі (виконання), недостатнє залучення користувачів і топ-менеджерів в роботу над проектом. DSDM складається з:

- трьох стадій: передпроектна стадія життєвого циклу проекту і постпроектна стадія;

- стадія життя проекту складається з 5 етапів: дослідження реалізованості, дослідження економічної доцільності, створення функціональної моделі, проектування і розробка, етап реалізації.

При деяких умовах існує можливість включення в DSDM частин інших методик, таких як Rational Unified Process (RUP), Екстремальне програмування, PRINCE2. Інший гнучкий метод, схожий на DSDM по процесу і концепції – Scrum.

Метод DSDM був розроблений у Великобританії в 1990-х Консорціумом DSDM. Консорціум DSDM – це асоціація розробників та експертів в області програмного забезпечення, створена з метою «спільної розробки і просування незалежного фреймворку RAD» комбінуванням кращого практичного досвіду учасників асоціації. Консорціум DSDM – це некомерційна організація незалежних розробників, які володіють та управляють фреймворком DSDM. Перша версія фреймворку була завершена в січні 1995 року і опублікована в лютому 1995 року. У липні 2006 року була представлена відкрита версія DSDM 4.2, яка стала доступна приватним особам для перегляду і використання. Тим не менш, всі, хто поширює DSDM, повинні бути членами цього некомерційного консорціуму.

На початку 1990-х в індустрії інформаційних технологій став поширюватись новий термін – швидка розробка застосунків (Rapid Application Development, RAD). Інтерфейси прикладних програм еволюціонували від старих зелених екранів до графічних інтерфейсів користувача, які використовуються і зараз. На ринок почали виходити нові інструменти для створення застосунків, наприклад PowerBuilder. Вони дозволили розробникам простіше ділитися планованими розробками з покупцями – з'явилося прототипування і почалося руйнування класичних, послідовних (каскадних) методів розробки.

Тим не менш, новий рух RAD було дуже неструктурованим: не існувало узгодженого опису цього методу і у багатьох організацій були створені власні опису і підходи до нього. Безліч великих корпорацій були зацікавлені в перспективах, що надаються методом, але вони також були стурбовані тим, щоб не знизився рівень якості їх продукції в кінцевому результаті.

Консорціум DSDM був утворений в 1994 році, коли група людей зустрілася на заході, організованому Butler Group в Лондоні. Всі, хто прийшов на цей захід, працювали у великих організаціях, таких як British Airways, American Express, Oracle and Logica (такі компанії як Data Sciences і Allied Domecq з тих пір були поглинені іншими організаціями).

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		47

На цьому зібранні було вирішено, що Дженніфер Степлтон, тоді представляла компанію Logica, розробить архітектуру комплексного, орієнтованого на користувача методу з хорошим контролем якості для ітеративної і інкрементній розробки. Підсумкова архітектура була спроектована так, щоб бути повністю сумісна зі стандартом ISO 9000 і PRINCE2.

Коли архітектура була готова (через місяць після зборів), Консорціум сформував групи для її поширення у всіх областях розробки програмного забезпечення, які включали в себе: методи та засоби управління проектом, контроль якості та тестування, методи і засоби розробки. Контрольна група, очолювана творцем архітектури і складається з глав цих груп, повинна була забезпечити розуміння методу так, як він спочатку замислювався.

Незважаючи на те, що багато членів Консорціуму були прямими конкурентами, вони вільно ділилися тим, як вони вирішують проблеми, що виникають. Практика показала, що найкращий результат може бути досягнутий тільки працюючи як одне ціле.

Консорціум збільшився за перший рік від декількох організацій до шістдесяти; опис методу ставало все більш і більш повним. Версія 1 була сформована в грудні 1994 року і опублікована в лютому 1995 року. Результатом став універсальний метод, що охоплює людей, процеси та інструменти. Він сформувався на основі досвіду організацій, різних за родом своєї діяльності і розмірами.

Метод DSDM – принципи. Існує 9 принципів, що складаються з 4 основних та 5 початкових точок.

1. Залучення користувача – це основа ведення ефективного проекту, де розробники ділять з користувачами робочий простір і тому прийняті рішення будуть більш точними.

2. Команда повинна бути уповноважена приймати важливі для проекту рішення без узгодження з начальством.

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		48

3. Часта поставка версій результату, з урахуванням такого правила, що «поставити щось хороше раніше – це завжди краще, ніж поставити все ідеально зроблена в кінці». Аналіз поставок версій з попередньої ітерації враховується на наступній.

4. Головний критерій – як можна більш швидка поставка програмного забезпечення, яке відповідає поточним потребам ринку. Але в той же час постачання продукту, який задовольняє потребам ринку, не менш важлива, ніж вирішення критичних проблем у функціоналі продукту.

5. Розробка – ітеративна та інкрементна. Вона ґрунтується на зворотного зв'язку з користувачем, щоб досягти оптимальної з економічної точки зору рішення.

6. Будь-які зміни під час розробки – оборотні.

7. Вимоги встановлюються на високому рівні перш, ніж почнеться проект.

8. Тестування інтегровано в життєвий цикл розробки.

9. Взаємодія і співпраця між усіма учасниками необхідно для його ефективності.

Передумови для використання DSDM.

Щоб успішно використовувати DSDM, необхідно щоб був виконаний ряд передумов. По-перше, необхідно організувати взаємодію між проектною командою, майбутніми користувачами і вищим керівництвом. По-друге, повинна бути можливість поділу проекту на менші частини, що дозволить використовувати ітеративний підхід.

Два приклади проектів, для яких DSDM не дуже підходить:

1. Проекти, критичні безпеки розширене тестування та затвердження в таких проектах конфліктують з метою методу DSDM укластися в терміни і бюджет.

2. Проекти, чия мета зробити компоненти багаторазового використання – вимоги в таких проектах занадто високі і не вкладаються в принцип 80 %/20%.

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		49

Життєвий цикл проекту. Фреймворк DSDM складається з трьох послідовних стадій, які називаються передпроектна стадія, стадія життєвого циклу проекту і постпроектна стадія. Стадія життєвого циклу проекту – сама продумана і детально розроблена з усіх інших. Вона складається з п'яти етапів, які формують ітеративний, інкрементний підхід до розробки інформаційних систем.

Ці три фази і відповідні етапи будуть більш детально описані в наступних розділах.

Для кожної стадії або етапи будуть розглянуті найважливіші функції і будуть представлені результати.

Стадія 1 – Передпроектна.

На цій стадії визначаються ймовірні проекти, відбувається виділення коштів та визначення проектною команди. Рішення задач на цій стадії допоможе уникнути проблем на більш пізніх стадіях проекту.

Стадія 2 – Життєвий цикл проекту.

На рисунку зображена дана стадія. На ньому показано 5 етапів, які потрібно пройти проекту, щоб стати інформаційною системою. Перші два етапи, дослідження реалізованості та дослідження економічної доцільності, йдуть послідовно і доповнюють один одного. Після завершення цих етапів, відбувається ітеративна та інкрементна розробка системи в етапах: створення функціональної моделі, проектування і розробка, етап реалізації. Ітеративна та інкрементна природа DSDM буде описана далі.

Стадія 3 – Постпроектна.

На цій стадії забезпечується ефективна робота системи. Це досягається за рахунок підтримки проекту, його покращення та виправлення помилок згідно з принципами DSDM. Підтримка проекту здійснюється продовженням розробки, заснованої на ітеративної і інкрементній природі DSDM. Замість того, щоб закінчити проект за один цикл, зазвичай повертаються до попередніх стадій або етапів, щоб поліпшити продукт.

Нижче на діаграмі представлений весь життєвий цикл проекту. Ця діаграма описує ітеративну розробку DSDM. Опис кожного етапу буде представлено нижче.

Чотири етапи стадії життєвого циклу проекту

Дослідження:

1. Дослідження реалізованості. На даному етапі визначається – потрапляє проект під рамки DSDM. Розглядаючи тип проекту, організаційні і кадрові питання, виноситься рішення – використовувати метод DSDM чи ні. Таким чином буде отримано звіт про застосовність, допустимий прототип і приблизний глобальний план проекту, який включає в себе план розробки і протокол можливих ризиків.

2. Дослідження економічної доцільності. На даному етапі аналізуються основні економічні і технологічні характеристики. Відбувається нарада експертів, на якій обговорюються найбільш важливі сторони системи і приймається рішення про пріоритети у розробці. На цьому етапі розробляються список основних вимог, опис сфери комерційної діяльності, опис архітектури системи і приблизний план створення прототипів.

Створення функціональної моделі:

1. Визначення функціонального прототипу. Визначення функціоналу, який буде закладено в прототипі на даному етапі. На цьому підетапі розробляється функціональна модель згідно до результатів, отриманих на стадії дослідження економічної доцільності.

2. Узгодження планів. Відбувається узгодження як і в які терміни має бути розроблена функціональність прототипу.

3. Створення функціонального прототипу. Розробка функціонального прототипу, згідно з планами та функціональної моделі.

4. Аналіз функціонального прототипу. Перевірка справності розробленого прототипу. Це може бути досягнуто тестуванням прототипу кінцевим

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		51

користувачем і/або переглядом документації. Результат – документ, що містить огляд функціонального прототипу.

Проектування і розробка:

1. Визначення конструктивного прототипу. Визначення функціональних і нефункціональних вимог системи. На основі цих вимог повинна бути створена стратегія реалізації. Якщо існує запис про тестування з попередньої ітерації, тоді він теж буде використаний для створення стратегії реалізації.

2. Узгодження планів. Відбувається узгодження як і в які терміни повинні бути реалізовані поставлені вимоги.

3. Створення конструктивного прототипу. Створення системи (конструктивного прототипу), яку можна віддати користувачам для тестування.

4. Аналіз конструктивного прототипу. Перевірити справність спроектованої системи. На цьому підетапі застосовується тестування та перегляд результатів. Створення користувальницької документації та протоколу випробування.

Реалізація:

1. Затвердження системи користувачем. Кінцеві користувачі стверджують протестовану систему для подальшої реалізації і створення довідника користувача.

2. Навчання користувачів. Навчання майбутнього користувача роботі з системою. Результат підетапу – контингент підготовлених користувачів.

3. Реалізація. Реалізація протестованої системи серед користувачів.

4. Аналіз ринку системи. Аналіз впливу випущеної системи на ринок. Головне питання – чи досягнута мета, поставлена при проектуванні системи. Ґрунтуючись на цьому проект переходить на наступну стадію (постпроектную) або повертається на попередню для доопрацювання. Аналіз буде представлений в документі аналізу проекту.

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		52

4.2 Захист розробленого програмного забезпечення

Для захисту розробленого програмного забезпечення запропоновано використовувати алгоритм Lucifer. Алгоритм Lucifer являє собою мережу перестановок і підстановок, його основні блоки нагадують блоки алгоритму DES. В DES результат функції f складається операцією XOR із входом попереднього раунду, утворюючи вхід наступного раунду. В S-блоках алгоритму Lucifer 4-бітові входи й виходи, вхід S-блоків являє собою перетасований вихід S-блоків попереднього раунду, входом S-блоків першого раунду служить відкритий текст. Для вибору використовуваного S-блоку із двох можливих використовується біт ключа. (Lucifer реалізує все це в єдиному T-блоці з 9 бітами на вході й 8 бітами на виході). На відміну від алгоритму DES, половини блоку між раундами не переставляються, та й саме поняття половини блоку в алгоритмі Lucifer не використовується. У цього алгоритму 16 раундів, 128-бітові блоки й більше проста, чим в DES, схема розгорнення ключа.

Блок тексту розглядається як ненегативне ціле число, або як кілька незалежних ненегативних цілих чисел. Довжина блоку завжди вибирається рівною ступеню двійки. У алгоритмі Lucifer використовуються наступні типи операцій:

- Таблична підстановка, при якій група біт відображається в іншу групу біт. Це так звані S-box.
- Переміщення, за допомогою якого біти повідомлення переупорядковуються.
- Операція додавання по модулю 2, позначувана XOR або $\oplus$.
- Операція додавання по модулю 2^{32} або по модулю 2^{16} .
- Циклічне зрушення на деяке число біт.

Ці операції циклічно повторюються в алгоритмі, створюючи так звані раунди. Входом кожного раунду є вихід попереднього раунду й ключ, що отриманий по певному алгоритму із ключа шифрування K . Ключ раунду

називається підключем. Алгоритм шифрування може бути представлений у такий спосіб:

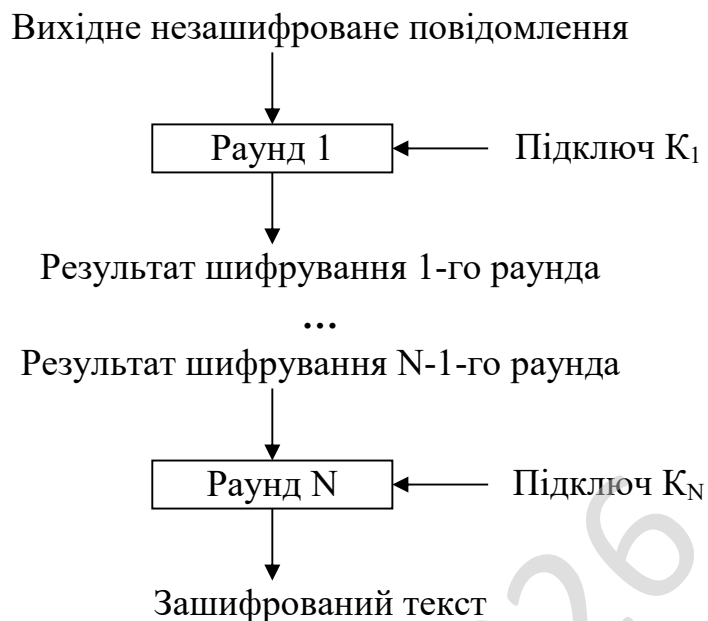


Рисунок 4.3 – Структура алгоритму алгоритмі Lucifer

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

На рисунку 5.1 зображено розроблене у бакалаврській роботі дослідження та програмна реалізація системи обробки даних двовимірних матричних штрихкодів типу Data MatrixECC 200. З рисунку можна побачити що інтерфейс головного вікна розподілено на наступні функціональні розділи:

- Навігаційного меню яке викликається натисканням правої клавіші маніпулятора миші.
- Розділу обрання групи.
- Розділу виведення результату роботи системи.
- Функції представлені у графічному вигляді (іконки).

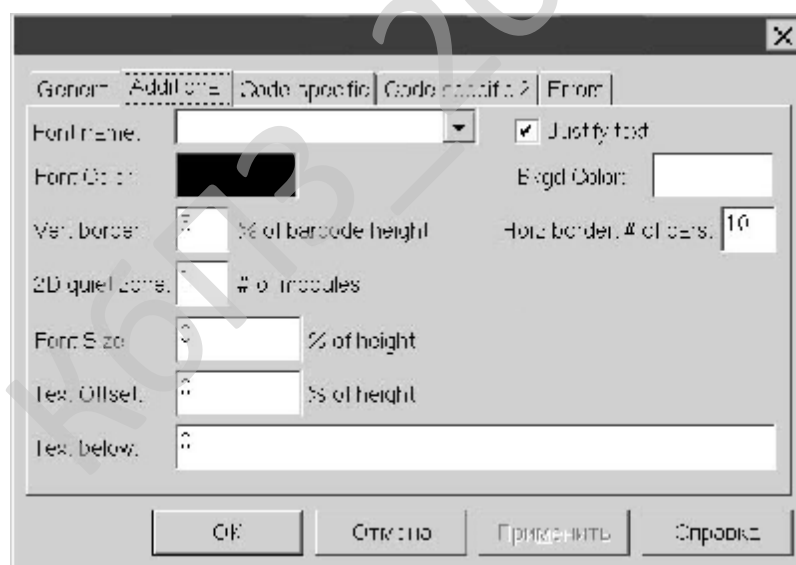


Рисунок 5.1 – Головне вікно ПЗ

Розроблена програма має дуже простий і зрозумілий інтерфейс з користувачем. Кожен, хто в достатньому обсязі володіє операційним середовищем Windows без особливих складностей освоїть і цю програму,

оскільки її інтерфейс інтуїтивно зрозумілий. Якщо програма не видала ніяких помилок, і працює, то можна використовувати, інакше слід слідувати інструкціям, які пропонує програма.

На рисунку 5.2 зображено авторські дані розробленого програмного забезпечення.

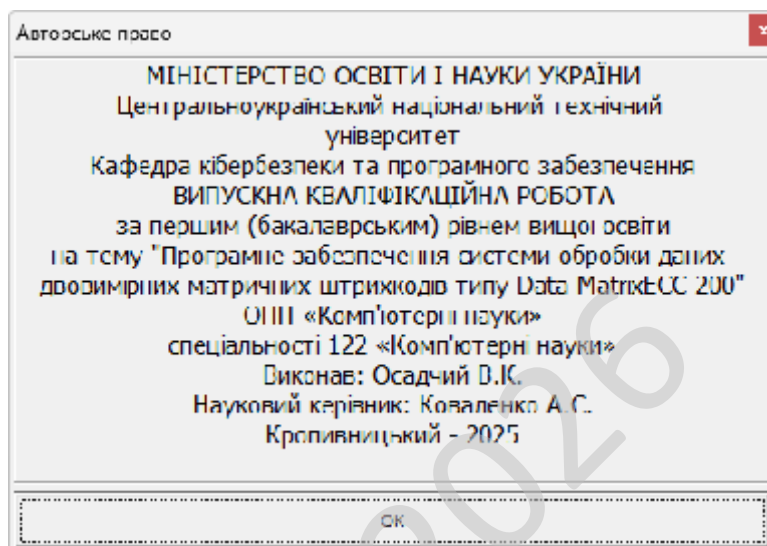


Рисунок 5.2 – Авторське право

Розглянемо процес впровадження програмного забезпечення, це процес налаштування програмного забезпечення під певні умови використання, а також навчання користувачів роботі з програмним продуктом. Впровадження програмного забезпечення це усі дії, що роблять розроблену програмну систему готовою до використання. Даний процес є частинною життєвого циклу програмного забезпечення.

Загалом процес розгортання складається з кількох взаємопов'язаних дій із можливими переходами між ними. Ця активність може відбуватися як з боку виробника так і з боку споживача. Оскільки кожна програмна система є унікальною, то усі процеси та процедури під час розгортання важко передбачити. Тому, "розгортання" можна трактувати як загальний процес відповідно до певних

вимог та характеристик. Розгортання може здійснюватись програмістом і в процесі розробки програмного забезпечення.

До діяльностей пов'язаних із розгортанням програмного забезпечення відносять:

- Випуск.
- Встановлення та активація.
- Деактивація.
- Адаптація.
- Оновлення.
- Вмонтування.
- Відстежування версій.
- Видалення.
- Вилучення з обігу.

При впровадженні програмного забезпечення потрібно урахувати наступні дії:

– Виділення критичних, з точки зору загального результату, процедур в діяльності організації. Коли набір таких процедур визначений, необхідно в першу чергу використовувати ІТ рішення для автоматизації операцій усередині саме цих процедур. Таким чином, розроблене ІТ рішення автоматично стає життєво важливим і затребуваним для організації, а також буде забезпечена публічність процесу впровадження;

– Розширення нормативної бази організації шляхом включення до неї регламентів, що описують порядок виконання процедур автоматизованих процесів. В іншому випадку є небезпека виникнення неузгодженості між автоматизованими процедурами та іншими процесами організації.

– Виконання робіт з загальної стандартизації існуючої діяльності організації, коли виділяються кращі практики виконання процедур і включаються в ІТ рішення за принципом найбільшої корисності для більшості учасників. Відсоток таких процедур щодо загального обсягу автоматизації може бути

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		57

невеликий, але це надає процесу побудови рішення вагу в організації за рахунок збільшення його необхідності.

Під час роботи над програмою було проведено тестування програмного забезпечення, тобто технічне дослідження, призначене для виявлення інформації про якість продукту відносно контексту, в якому воно має використовуватись.

Тестування включає як процес пошуку помилок або інших дефектів, так і випробування програмних складових з метою їх оцінки.

Проводилась оцінка:

- відповідності поставленим вимогам;
- правильна відповідь для усіх можливих вхідних даних;
- виконання функцій за прийнятний час;
- практичність;
- сумісність з ОС та стороннім ПЗ.

Оскільки число можливих тестів для програмних компонент практично нескінченне, тому стратегія тестування полягала в тому, щоб провести всі можливі тести з урахуванням наявного часу та ресурсів.

Як результат ПЗ тестувалось стандартним виконанням програми з метою виявлення помилок або інших дефектів.

Проводилось тестування чорної скриньки. Основне місце програми тестів «чорної скриньки» – інтерфейс ПЗ. Відомі: функції програми. Досліджується: робота кожної функції на всій області визначення.

Ці тести демонструють:

- Як виконуються функції програми.
- Як приймаються вихідні дані.
- Як виробляються результати.
- Як зберігається цілісність зовнішньої інформації.

При тестуванні «чорної скриньки» розглядаються системні характеристики програм, ігнорується їхня внутрішня логічна структура. Вичерпне тестування, як правило, неможливе.

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		58

Наприклад, якщо в програмі 10 вхідних величин і кожна приймає по 10 значень, то кількість тестових варіантів становитиме 10^{10} . Тестування «чорної скриньки» не реагує на багато особливостей програмних помилок.

Тестування «чорної скриньки» (функціональне тестування) дозволяє отримати комбінації вхідних даних, які забезпечують повну перевірку всіх функціональних вимог до програми.

Програмний виріб тут розглядається як «чорна скринька», чію поведінку можна визначити тільки дослідженням його входів та відповідних виходів. При такому підході бажано мати:

- Набір, утворений такими вхідними даними, які призводять до аномалій у поведінці програми (назвемо його ІТс).

- Набір, утворений такими вхідними даними, які демонструють дефекти програми (назвемо його ОТ).

Будь-який спосіб тестування «чорної скриньки» повинен:

- Виявити такі вхідні дані, які з високою ймовірністю належать набору ІТс.
- Сформулювати такі очікувані результати, які з високою ймовірністю є елементами набору ОТ.

Принцип «чорної скриньки» не альтернативний принципу «білої скриньки». Скоріше це доповнює підхід, який виявляє інший клас помилок.

Тестування «чорної скриньки» забезпечує пошук наступних категорій помилок:

- Некоректних чи відсутніх функцій.
- Помилки інтерфейсу.
- Помилки у зовнішніх структурах даних або в доступі до зовнішньої бази даних.
- Помилки характеристик (необхідна ємність пам'яті і т.д.).
- Помилки ініціалізації та завершення.

Обрано умови розповсюдження – Shareware. Під умовно-безплатним програмним забезпеченням можна розуміти спосіб або метод розповсюдження

комерційного ПЗ на ринку (тобто на шляху до кінцевого користувача), при якому випробувачеві пропонується обмежена за можливостями (не повнофункціональна або демонстраційна версія), терміном дії (тріал версія) або версія з вбудованим набридливим нагадуванням про необхідність оплати використання програми.

В угоді про використання (ліцензії для кінцевого користувача, EULA) також може бути обумовлена заборона на комерційне або професійне (не тестове) її використання.

Основний принцип умовно-безплатного ПЗ – «спробуй, перш ніж купити» (try before you buy). ПЗ що поширюється як умовно-безплатний, надається користувачам безоплатно. Звичайно користувач платить тільки за час завантаження файлів через Інтернет або за носій (CD диск, флешку, ключ). Протягом певного терміну, що становить зазвичай тридцять днів, він може користуватися програмою, тестувати її, освоювати її можливості.

Якщо після закінчення цього терміну користувач вирішить продовжити використання ПЗ, він зобов'язаний купити його (zareestruvatisya), zaplativshi avtorovi певну суму.

В іншому випадку користувач повинен припинити використання ПЗ та видалити його зі свого комп'ютера.

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		60

6 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти, призначено для системи обробки даних двовимірних матричних штрихкодів типу Data MatrixECC 200.

В межах України в недостатній мірі представлені вітчизняні розробки в цій області.

Рішення завдання полягало у вирішенні наступних задач:

- Був проведений огляд існуючих систем обробки даних двовимірних матричних штрихкодів типу Data MatrixECC 200.
- Досліджена система обробки даних двовимірних матричних штрихкодів типу Data MatrixECC 200.
- На основі отриманих результатів досліджень створена програмна реалізація системи обробки даних двовимірних матричних штрихкодів типу Data MatrixECC 200.

Розроблені під час виконання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання обробки даних двовимірних матричних штрихкодів типу Data MatrixECC 200.

Розроблене програмне забезпечення має простий, дружній та зручний інтерфейс користувача, що забезпечує легкість у освоєнні роботи програмного продукту, зручність у використанні, і не потребує особливих спеціальних знань.

При створенні програмного забезпечення було використано об'єктно-орієнтований підхід, що відповідає сучасним тенденціям у галузі розробки комерційних програмних систем.

Програма реалізована на мові високого рівня Python. Дана мова програмування дозволяє найбільш ефективно обробляти дані призначені для

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		61

системи обробки даних двовимірних матричних штрихкодів типу Data MatrixECC 200. Це дозволило мінімізувати строк розробки програмного забезпечення, і, як слід, зменшити витрати на його розробку. Запропоноване програмне забезпечення ділиться на загальне програмне забезпечення, що поставляється із засобами обчислювальної техніки й спеціальне програмне забезпечення, що спеціально розроблене для даної конкретної системи й включає програми, що реалізують її функції.

Програма призначена для виконання під управлінням багатозадачної операційної системи Windows 10/11.

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

Для підвищення рівня безпеки запропоновано застосовувати алгоритм Lucifer.

В цілому створене програмне забезпечення підтверджує правильність використаних проектних рішень та повністю відповідає вимогам технічного завдання. Створене програмне забезпечення має потенційну можливість для подальшого вдосконалення і застосування у різних галузях.

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		62

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Мелешко Є.В., Якименко М.С., Поліщук Л.І. Алгоритми та структури даних: Навчальний посібник для студентів технічних спеціальностей денної та заочної форми навчання. – Кропивницький: Видавець – Лисенко В.Ф., 2019. – 156 с.
2. Knuth D. The Art of Computer Programming, Vol. 1: Fundamental Algorithms, 3rd Edition 3rd Edition. – Addison-Wesley Professional, 2019. – 672 p.
3. Knuth D. Art of Computer Programming, Vol. 2: Seminumerical Algorithms 3rd Edition, Kindle Edition. – Addison-Wesley Professional, 2019. – 672 p.
4. Knuth D. The Art of Computer Programming: Vol. 3: Sorting and Searching 2nd Edition, Kindle Edition. – Addison-Wesley Professional, 2019. – 800 p.
5. Cormen T.H., Leiserson C.E., Rivest R.L., Stein C. Introduction to Algorithms, 3rd Edition (The MIT Press) 3rd Edition – The MIT Press, 2019. – 1292 p.
6. Вінтенко Б.Ю., Миронець І.В., Смірнов О.А. «Модель комп'ютерно-орієнтованої процедури системи підтримки оперативного персоналу АЕС». *IV Міжнародна науково-практична Інтернет-конференція «Інновації та перспективні шляхи розвитку інформаційних технологій (ППШРІТ-2025)»* м.Черкаси 25 листопада 2025 року – Черкаси: ЧДТУ.– 2025. – С.101-103.
7. Kuznetsov O., Frontoni E., Kuznetsova Y., Chevardin V., Smirnov O. «Architectural foundations for adaptive security in edge computing systems». *Cybersecurity Defensive Walls in Edge Computing*, 2025. pp. 21-61.
8. Вінтенко, Б.Ю., Миронець, І.В., Смірнов, О.А., Коваленко, О.В., Усік, П.С., Буравченко, К.О., Лисенко, І.А. «Логіко-структурна модель комп'ютерно-орієнтованої процедури системи підтримки оперативного персоналу АЕС». *Кібербезпека: освіта, наука, техніка*. 2025. Том 2 № 30. С. 413-427, 2025.

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		63

9. Смірнова, Т.В. «Дослідження методів, моделей та сучасних ІТ-рішень для підтримки технологічних процесів у критичній інфраструктурі держави». *Кібербезпека: освіта, наука, техніка*. 2025. Том 2 № 30. С.195-208, 2025.

10. Вінтенко Б., Смірнов О., Миронець І., Смірнова Т., Смірнов С. «Імітаційна модель шляхів вхідних даних комп'ютерної інтелектуальної системи підтримки оператора енергоблоку АЕС». *Комбінаторні конфігурації та їхні застосування: Матеріали XXVII Міжнародного науково-практичного семінару, присвяченого 125-річчю Національного університету «Запорізька політехніка» (Запоріжжя-Кропивницький-Київ, 4-6 червня 2025 р.)*. Запоріжжя: НУ «Запорізька політехніка», 2025. С.82-91.

11. Al-Azzeh, J., Ayyoub, B., Mesleh, A., Smirnova, T., Gnatyuk, S., Driev, O., Smirnov, O., Dorenskyi, O. «Cloud-Based Information System for Evaluating Caverns in the Process of Blasting Metal Surfaces of Details». *International Review on Modelling and Simulations* 18 (1), 2025. pp. 32-42.

12. Вінтенко Б.Ю., Смірнов О.А., Миронець І.В., Смірнова Т.В., Коваленко О.В., Мацуй А.М. «Модель шляхів отримання вхідних даних комп'ютерної інтелектуальної системи підтримки оперативного персоналу АЕС». *Центральноукраїнський науковий вісник. Технічні науки*. 2025. Вип. 11(42), ч. II. С.52-62.

13. Вінтенко Б.Ю., Смірнов О.А., Миронець І.В., Смірнова Т.В. «Методи забезпечення відмовостійкості інтелектуальних систем підтримки оператора». *VIII міжнародна науково-практична конференція “Інформаційна безпека та комп'ютерні технології”*, м. Кропивницький. 24-25 квітня 2025 р. – Кропивницький: ЦНТУ. – 2025. – С. 44-46.

14. Смірнов, О.А., Константинова, Л.В., Коноплицька-Слободенюк, О.К., Козірова, Н.В, Якименко, Н.М., Доренський, О.П., Буравченко, К.О. «Дослідження інструментів штучного інтелекту для роботи з базами даних та аналізу даних». *Кібербезпека: освіта, наука, техніка*. 2025. №3(27), С. 429–448.)

15. Smirnov O., Fedorov E., Neskorodieva A., Neskorodieva T. «Intellectual Classification method of Gymnastic Elements Based on Combinations of Descriptive and Generative Approache». *CEUR Workshop Proceedings* Volume 3664, 2024, Pages 11-23.

16. Вінтенко, Б., Миронець, І., Смірнов, О., Коваленко, А., Коноплицька-Слободенюк, О., Смірнова, Т., Константинова, Л. «Дослідження застосування систем підтримки оперативного персоналу об'єкту критичної інфраструктури при керуванні енергоблоком АЕС з реактором типу ВВЕР-1000». *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка»*, 2024. № 2(26), С. 6-26.

17. Вінтенко, Б., Миронець, І., Смірнов, О., Кравчук, О., Козірова, Н., Савеленко, Г., Коваленко, А. «Дослідження вимог та аналіз кібербезпеки програмного забезпечення інформаційно-керуючих систем АЕС, важливих для безпеки». *Кібербезпека: освіта, наука, техніка*. 2024. №3(23), С. 111-131.

18. Kuznetsov O., Ilchenko O., Kryvinska N., Buravchenko K., Smirnov O., Savchenko Iu. «An Empirical Assessment of Leading Blockchain Financial Services». *2023 IEEE 1st Ukrainian Distributed Ledger Technology Forum (UADLTF)*, Kyiv, Ukraine, 2023, pp. 1-6,

19. Kuznetsov, O., Kryvinska, N., Ilchenko, O., Smirnova, T., Ulianovska, Y. «Comparative Analysis of Cryptocurrency Trading Platforms Using the Analytic Hierarchy Process». *CEUR Workshop Proceedings*, 2023, 3628, pp. 106-115.

20. Malyukov V., Bebashko B., Lakhno V., Smirnov O., Malyukova I., Mohylnyi H. «Managing the Purchase-Sale Process of Digital Currencies Under Fuzzy Conditions». *Lecture Notes in Networks and Systems*, 2023, 729 LNNS, pp. 104–112.

21. Al-Mudhafar Aqeel, A.M., Smirnova, T., Buravchenko, K., Smirnov, O. «The method of assessing and improving the user experience of subscribers in software-configured networks based on the use of machine learning». *Advanced Information Systems*, 2023, 7(2), pp. 49-56.

22. Smirnov, O., Sydorenko, V., Aleksander, M., Zhyharevych, O., Yenchев, S. «Simulation of the cloud IoT-based monitoring system for critical infrastructures». *CEUR Workshop Proceedings*, Volume 3530, 2023, pp. 256-265.

23. Smirnov, O., Odarchenko, R., Smirnova, T., Bondar, S., Volosheniuk, D. «Optimal Structure Construction of Private 5G Network for the Needs of Enterprises». *Lecture Notes on Data Engineering and Communications Technologies*, 2023, 178, pp. 208–223.

24. Вінтенко Б.Ю., Смірнов О.А., Коваленко А.С., Смірнов С.А., Буравченко К.О. «Дослідження вимог міжнародних стандартів ІЕС60880 та ІЕС62138 з розробки програмного забезпечення інформаційно-керуючих систем АЕС, важливих для безпеки». *Системи управління, навігації та зв'язку*, 2023, вип. 3(73), С. 155-166.

25. Вінтенко, Б., Миронець, І., Смірнов, О., Кравчук, О., Козірова, Н., Савеленко, Г., Коваленко, А. «Дослідження вимог та аналіз кібербезпеки програмного забезпечення інформаційно-керуючих систем АЕС, важливих для безпеки». *Кібербезпека: освіта, наука, техніка*. 2024. №3(23), С. 111-131.

26. Вінтенко Б.Ю., Смірнов О.А., Коваленко О.В., Смірнов С.А., Коваленко А.С. «Дослідження нормативних документів та галузевих стандартів розробки програмного забезпечення комп'ютерних систем управління АЕС, важливих для безпеки». *Системи управління, навігації та зв'язку*, 2023, вип. 2(72), С. 170-178.

27. Аль-Мудхафар Акіл Абдулхуссейн М., Смірнова Т.В., Буравченко К.О., Смірнов О.А. «Метод оцінки та підвищення користувальницького досвіду абонентів в програмно-конфігурованих мережах на основі використання машинного навчання». *Сучасні інформаційні системи*, 2023, том 7, № 2, С. 49-56.

28. Вінтенко Б.Ю., Смірнов О.А., Коваленко О.В., Смірнов С.А. «Дослідження нормативної документації та стандартів розробки програмного забезпечення комп'ютерних систем управління АЕС, важливих для безпеки». VI міжнародна науково-практична конференція “Інформаційна безпека та

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		66

комп'ютерні технології", м. Кропивницький. 20-21 квітня 2023 р. – Кропивницький: ЦНТУ. – 2023. – С. 35-36.

29. Smirnov, O., Karapetyan, A., Fedorov, E., «Creating Neural Network and Single Solution Human-Based Metaheuristic Methods of Solving the Traveling Salesman Problem». CEUR Workshop Proceedings, Volume 3312, 2022, pp. 47-58.

30. Smirnov O., Kuznetsov A., Kryvinska N., Kiian A., Kuznetsova K. «Full Non-Binary Constant-Weight Codes». SN Computer Science, Vol 2, 337, 2021. <https://doi.org/10.1007/s42979-021-00739-w>.

31. Smirnov O., Kovalenko O., Kovalenko A., Kavun S. «Quantitative Risk Assessment Method Development in the Context of the SDLC-model». 2021 IEEE 8th International Conference on Problems of Infocommunications, Science and Technology (PIC S&T), 2021, pp. 203-208, doi: 10.1109/PICST54195.2021.9772143

32. Smirnova T., Gnatyuk S., Berdibayev R., Avkurova Zh., Iavich M. «Cloud-Based Cyber Incidents Response System and Software Tools». Communications in Computer and Information Science, 2021, vol 1486. Springer, Cham. pp 169-184.

33. Smirnov, O., Kuznetsov, A., Potii, O., Poluyanenko, N., Stelnyk, I., Mialkovsky, D. «Combining and filtering functions in the framework of nonlinear-feedback shift register». International Journal of Computing; 2020, Volume 19, Issue 2 – Research Institute for Intelligent Computer Systems – 2020. – P. 247-256.

34. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova T. «Non-binary constant weight coding technique». CEUR Workshop Proceedings. Volume 2740, 2020, Pages 102-114.

35. Smirnov O., Kuznetsov A., Kiian A., Cherep A., Kanabekova M., Chepurko I. «Testing of code-based pseudorandom number generators for post-quantum application». 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT), Ukraine, Kyiv, May 14-18. 2020. P. 172-177.

36. Smirnov, O., Shekhanin, K., Kuznetsov, A., Krasnobayev, V. «Detecting Hidden Information in FAT». International Journal of Computer Network and Information Security (IJCNIS). Vol. 12, No. 3, 2020. PP.33-43.
37. Smirnov, O., Drieieva, H., Drieiev, O., Simakhin, V., Bondar, S., Odarchenko, R. «Managing multifractal properties of the binary sequence generated with the Markov chains», CEUR Workshop Proceedings Volume 2608, 2020, Pages 633-645.
38. Smirnov O. Kuznetsov A., Zaichenko Yu., Pastukhov M., Oleshko O., Kuznetsova K., «Formation of Discrete Signals with Special Correlation Properties». International Conference on Information and Telecommunication Technologies and Radio Electronics, UkrMiCo 2019; Odessa; Ukraine; 9-13 September 2019. P.22-28.
39. Smirnov, O., Kuznetsov, A., Kolovanova, I., Kuznetsova, T., «Noise immunity of the algebraic geometric codes». International Journal of Computing; 2019, Volume 18, Issue 4 – Research Institute for Intelligent Computer Systems – 2019. – P. 393-407.
40. Smirnov, O., Kuznetsov, A., Reshetniak, O., Ivko, N., Katkova, T., Kuznetsova, T., «Generators of Pseudorandom Sequence with Multilevel Function of Correlation». 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T), Kyiv, Ukraine, 8 – 11 October 2019 . P.517-522.
41. Smirnov, O., Krasnobayev, V., Yanko, A., Kuznetsova, T. «Methods of nulling numbers in the system of residual classes». CEUR Workshop Proceedings, Vol 2588, P. 90-106, 2019.
42. Kuznetsova, T., «Code-Based Schemes for Post-Quantum Digital Signatures», 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2019; Metz; France; 18-21 September 2019. P. 707-712.
43. Smirnov, O., Kuznetsov, A., Stefanovych, O., Gorbenko, Y., Krasnobaev, V., Kuznetsova K. «Information Hiding Using 3D-Printing Technology»,

10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2019; Metz; France; 18-21 September 2019. P.701-706.

44. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Averchev, A., Pastukhov, M., Kuznetsova, K., «Formation of Pseudorandom Sequences with Special Correlation Properties», 2019 3rd International Conference on Advanced Information and Communications Technologies, AICT-2019/ Lviv, Ukraine, 2-6 July, 2019, P. 395-399.

45. Смірнов, О.А., Усік П.С., Полігенько О.О., Одарченко Р.С., Терещенко Л.Ю. «Інформаційна технологія та програмне забезпечення для підвищення ефективності планування підсистеми базових станцій стільникового зв'язку». Проблеми телекомунікацій. № 1(26). С. 83-96. 2020.

46. Смірнов О.А., Усік П.С., Миронець І.В., Буравченко К.О., Якименко Н.М. «Метод підвищення ефективності розподіленої обробки даних у комп'ютерних системах операторів стільникового зв'язку» Вісник Черкаського державного технологічного університету. Технічні науки. №4. С. 103-110. 2020.

47. О.А.Смірнов, Т.В.Смірнова, Л.І. Поліщук, К.О. Буравченко, А.О.Макевнін, «Дослідження хмарних технологій як сервісів», Кібербезпека: освіта, наука, техніка. № 3(7). С. 43-62. 2020.

48. Смірнов О.А., Дреєва Г.М., Дреєв О.М., Смірнова Т.В. «Фрактальний аналіз генератора самоподібного трафіку на основі ланцюга Маркова». Центральнотраїнський науковий вісник. Технічні науки. № 2(33). с. 161-172, 2019.

49. Smirnov, O., Kuznetsov, A., Kuznetsova., K. Synthesis of Discrete Signals with Improved Correlation Properties. Монографія: In.: ISCI'2019: Information Security in Critical Infrastructures. Collective monograph. Edited by Ivan D. Gorbenko and Alexandr A. Kuznetsov, ASC Academic Publishing, USA, 2019, pp. 281-299. – ISBN: 978-0-9989826-8-7 (Hardback), ISBN: 978-0-9989826-9-4 (Ebook).

50. Смірнова Т.В., Солових Є.К., Смірнов О.А., Дреєв О.М. Побудова хмарних інформаційних технологій оптимізації технологічного процесу відновлення та зміцнення поверхонь деталей. Центральнотукраїнський науковий вісник. Технічні науки. № 1(32). с. 184-194, 2019.

51. Смірнов О.А., Котелянець В.В. Стійкі до колізій стохастичні моделі функціонування безпроводових сенсорних мереж. Вісник інженерної академії України, №3, с. 145-152, 2018

КБПЗ_2026

					ВКРБ-122.26.0022.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		70