Вінницький національний технічний університет

Факультет менеджменту та інформаційної безпеки

Кафедра менеджменту та безпеки інформаційних систем

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

Покращення захисту інформаційних систем з використанням захищеного флеш-носія, двофакторної ідентифікації на основі аналізу клавіатурного почерку та оптимізацією звернень до бази даних на основі генетичного алгоритму

Виконав: ст. 2-го курсу, групи КІТС-22мз
спеціальності 125– Кібербезпека
Освітня програма – Кібербезпека
інформаційних технологій та систем
Каращенко В. С.
Керівник: к. т. н., проф., професор каф. МБІС
Азарова А. О.
«____» _____ 2024 р.
Опонент: к.т.н., доц., доцент каф. ОТ
Крупельницький Л.В.
«____» _____ 2024 р.

Допущено до захисту
Голова секції УБ кафедри МБІС

_____Юрій ЯРЕМЧУК
“_____”_____2024 р.

Вінницький національний технічний університет
Факультет менеджменту та інформаційної безпеки
Кафедра менеджменту та безпеки інформаційних систем

Рівень вищої освіти ІІ-й (магістерський)
Галузь знань 12 – Інформаційні технології
Спеціальність 125 – Кібербезпека
Освітньо-професійна програма – Кібербезпека інформаційних технологій та систем

ЗАТВЕРДЖУЮ
Голова секції УБ, кафедра МБІС

_____ **Юрій ЯРЕМЧУК**

“_____”_____2024 р.

З А В Д А Н Н Я на магістерську кваліфікаційну роботу студенту

Каращенку Вячеславу Сергійовичу

1. Тема роботи «Покращення захисту інформаційних систем з використанням захищеного флеш-носія, двофакторної ідентифікації на основі аналізу клавіатурного почерку та оптимізацією звернень до бази даних на основі генетичного алгоритму» Керівник роботи Азарова А.О. проф., затверджені наказом вищого навчального закладу від «11» березня 2024 року № 81
2. Строк подання студентом роботи за тиждень до захисту
3. Вихідні дані до роботи: нормативно-правова база, монографії та сучасні наукові статті по темі, Інтернет-ресурси, стандарти, існуюче ПЗ.
4. Зміст текстової частини: в першому розділі описано необхідність забезпечення захищеного доступу до ІС, здійснено аналіз існуючих методів контролю доступу до ІС; в другому розділі здійснено вдосконалення методу, описано особливості реалізації клавіатурного почерку, перевірки флеш-носія та оптимізації звернень до БД; в третьому розділі описано програмну реалізацію розробки на основі удосконаленого методу та проведено тестування; в четвертому розділі проаналізовано економічну ефективність розробленого програмного забезпечення на основі удосконаленого методу.

5. Перелік ілюстративного матеріалу (з точним зазначенням обов`язкових креслень):

– у першому розділі наведено 10 рис., 2 табл.;

– у другому розділі наведено 10 рис.;

– у третьому розділі наведено 18 рис., 2 табл.; – у четвертому розділі наведено 9 табл.

6. Консультанти розділів магістерської кваліфікаційної роботи

Розділ	Прізвище, ініціали та посада Консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	проф. каф. МБІС Азарова А.О.		
2	проф. каф. МБІС Азарова А.О.		
3	проф. каф. МБІС Азарова А.О.		
4	к.е.н., доц. каф. ЕПВМ, Причепа І.В.		

7.Дата видачі завдання 12.03.2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів МКР	Строк виконання етапів МКР	Примітка
1.	Визначення напрямку МКР, формулювання теми	12.03.2024	15.03.2024
2.	Аналіз предметної області обраної теми	16.03.2024	27.03.2024
3.	Розробка алгоритму роботи	28.03.2024	05.04.2024
4.	Написання МКР на основі розробленої теми	06.04.2024	20.04.2024
5.	Розробка економічної частини	21.04.2024	25.04.2024
6.	Попередній захист МКР	26.04.2024	20.05.2024
7.	Виправлення, уточнення, коригування роботи	21.05.2024	31.05.2024
8.	Захист МКР	10.06.2024	11.06.2024

Студент _____ Каращенко В.С.

Керівник МКР _____ Азарова А.О.

АНОТАЦІЯ

УДК: 004.77

Каращенко В.С. Покращення захисту інформаційних систем з використанням захищеного флеш-носія, двофакторної ідентифікації на основі аналізу клавіатурного почерку та оптимізацією звернень до бази даних на основі генетичного алгоритму. Магістерська кваліфікаційна робота зі спеціальності 125 – «Кібербезпека», освітня програма «Кібербезпека інформаційних технологій та систем». Вінниця: ВНТУ, 2024. 128 с.

Укр. мовою бібліогр.: 50 назв; рис.: 38; табл. 13.

У магістерській кваліфікаційній роботі розроблено метод покращення захисту ІС з використанням захищеного флеш-носія, двофакторної ідентифікації на основі аналізу клавіатурного почерку і оптимізацією звернень до бази даних на основі генетичного алгоритму, а також ПЗ на основі даного методу.

У першому розділі роботи проведено аналіз теоретичного матеріалу обраної галузі знань: поняття захищеності ІС, застосування флеш-носіїв для контролю доступу; недоліки і переваги методів автентифікації користувачів, що уможливило обґрунтування застосування поведінкової біометричної характеристики для удосконалення існуючого підходу.

У другому розділі роботи описано вдосконалення методу для покращення захисту інформаційних систем із використанням захищеного флеш-носія, двофакторної ідентифікації на основі аналізу клавіатурного почерку та оптимізацією звернень до бази даних на основі генетичного алгоритму.

У третьому розділі роботи здійснено практичну реалізацію програмного продукту для захисту ІС від НСД на основі вдосконаленого методу.

У четвертому розділі роботи доведено економічну доцільність розробленого програмного засобу, що свідчить про її комерційний потенціал та потребу подальшого впровадження.

Ключові слова: ІС, несанкціонований доступ, флеш-носій, біометрична автентифікація, клавіатурний почерк, контроль доступу.

ABSTRACT

Karashchenko V.S. Improving the protection of information systems using secure flash media, two-factor identification based on the analysis of keyboard handwriting and optimization of database accesses based on a genetic algorithm. Master's qualification work in specialty 125 – «Cyber Security», Education Program «Information Security Management». Vinnytsa: VNTU, 2024. 128 p.

In Ukrainian. Bibliography: 50 titles; Figures: 38; Table 13.

The master's thesis presents a method for improving the security of information systems (hereinafter referred to as IS) using a secure flash drive, two-factor identification based on keyboard handwriting analysis and optimization of database accesses based on a genetic algorithm, as well as a software tool based on this method.

The first section of the paper analyzes the theoretical material of the chosen field of knowledge: the concept of IP security, the use of flash media for access control, and a method of user authentication based on behavioral biometric characteristics.

The second section of the paper describes the improvement of the method for improving the protection of information systems (hereinafter referred to as IS) using a secure flash drive, two-factor identification based on the analysis of keyboard handwriting and optimization of database accesses based on a genetic algorithm.

In the third section of the paper, the practical implementation of a software product for protecting information system from unauthorized access based on the improved method is carried out.

In the fourth section of the paper, the economic feasibility of the developed software tool is proved, which indicates its commercial potential and the need for further implementation.

Keywords: information system, unauthorized access, flash drive, biometric authentication, keyboard handwriting, access control.

ЗМІСТ

ВСТУП

Актуальність. Сьогодні інформація є цінним ресурсом, а отже потребує якісного захисту, що має постійно удосконалюватися. Для окремих даних та інформаційних систем загалом існує низка загроз, що є наслідком діяльності хакерів, які намагаються зламати наявні системи захисту. Однією із таких загроз є несанкціонований доступ до інформаційних систем, що призводить до витоку даних, фінансових та репутаційних збитків, а також збоїв у роботі обладнання.

Опрацьовано значний теоретичний доробок у галузі досліджуваних питань, варто відзначити роботи провідних закордонних та вітчизняних науковців, таких, як Б. Шнайер, К. Клар, Рафаель Янь, Ю.Яремчук, С. Петренко, В.Матвієнко [1 – 15]. Разом із тим, переважна більшість підходів є ресурсовитратною і дорогою, орієнтуються на застосування саме спеціалістами у галузі кібербезпеки. Отже, виникає нагальна потреба в розробленні таких засобів захисту інформаційних систем від несанкціонованого доступу, що є ефективними і, водночас, позбавленими вище наведених недоліків, що є запорукою забезпечення захищеності інформації та контролю доступу до неї.

Захист ІС – це не одноразова задача, а постійний процес, який потребує постійного моніторингу та оновлення. Організації та підприємства повинні регулярно оцінювати ризики та вразливості своїх ІС та вживати заходів для їх усунення. Враховуючи даний фактор, приватні і державні організації, науковці займаються дослідженням проблеми захисту інформації та впроваджують на практиці різні програмні розробки.

Для захисту від НСД застосовують технічні, організаційні та правові засоби. Зокрема, серед технічних засобів захисту все більшої популярності набувають системи доступу, що забезпечують не лише високий ступінь захисту інформації, але і зручність для користувача. Прикладами таких програмних рішень є розробки, які не потребують від користувачів особливих зусиль для проходження перевірки, зокрема це стосується біометричних даних та

електронних ключів. Такі характеристики не потребують постійних фінансових витрат та запам’ятовування складних паролів, а тому є зручними у користуванні.

Варто зауважити, що застосування електронних ключів у сфері кібербезпеки є постійно зростаючим, особливо для контролю доступу, шифрування і підпису даних, що забезпечує конфіденційність, цілісність і автентичність інформації.

Отже, можна узагальнити, що на сьогодні питання захисту інформаційних систем від несанкціонованого доступу залишається доволі актуальним, а ефективні та зручні засоби контролю доступу до таких систем користуються популярністю. Саме тому, у даній роботі здійснено та описано вдосконалення методу захисту ІС від НСД шляхом поєднання вказаних засобів захисту.

Мета і задачі дослідження. Метою роботи є вдосконалення методу захисту інформаційних систем з використанням захищеного флеш-носія, двофакторної ідентифікації на основі аналізу клавіатурного почерку та оптимізацією звернень до бази даних на основі генетичного алгоритму.

Для досягнення такої мети було поставлено та вирішено такі **задачі:**

– проаналізовано існуючі інструменти захисту інформаційних систем від несанкціонованого доступу та досліджено можливість їх застосування для захисту ІС;

– удосконалено метод захисту інформаційних систем від НСД шляхом застосування двофакторної ідентифікації;

– створено дуальний підхід до перевірки унікальних біометричних даних користувача на основі клавіатурного почерку;

– побудовано підхід до використання зашифрованого флеш-носія для реалізації захищеного доступу;

– розроблено БД для збереження еталонних даних користувачів та оптимізовано процес звернення до неї на основі генетичних алгоритмів;

– на базі складеного підходу запропоновано алгоритм роботи відповідного програмного засобу та здійснено його реалізацію;

– протестовано роботу ПЗ та проаналізовано отримані результати;

– обґрунтовано економічну доцільність впровадження запропонованої розробки.

Об’єкт дослідження – процес захищеності інформаційної системи засобами флеш-носія та біометричної ідентифікації.

Предмет дослідження – удосконалення методу захисту ІС засобами флеш-носія та біометричної ідентифікації.

Наукова новизна: удосконалено метод захисту інформаційних систем із використанням зашифрованого флеш-носія, двофакторної ідентифікації на основі аналізу клавіатурного почерку та оптимізації звернень до бази даних на основі генетичного алгоритму, що, на відміну від існуючих підходів, дозволяє суттєво покращити рівень захисту автентифікації контролю доступу користувачів до ІС.

Практична цінність: розроблено програмний засіб для захисту доступу до інформаційних систем із використанням захищеного флеш-носія, біометричного ідентифікатора та оптимізації звернень до бази даних на основі генетичного алгоритму.

Публікації. Результати роботи було захищено шляхом отримання свідоцтва на реєстрацію авторського права на твір [16].

1 ТЕОРЕТИЧНІ ЗАСАДИ ПРОЦЕСУ ЗАХИСТУ ІНФОРМАЦІЙНОЇ СИСТЕМИ ВІД НЕСАНКЦІОНОВАНОГО ДОСТУПУ

У даному розділі здійснено аналіз існуючих джерел у галузі захисту інформаційної системи від несанкціонованого доступу, визначено переваги та недоліки існуючих методів захисту інформаційних систем від НСД із метою обґрунтування базового серед них для подальших досліджень.

Під час написання розділу проаналізовано роль та значення захищеного доступу до інформаційної системи, метод контролю доступу на основі флеш-носіїв, а також процес автентифікації користувачів на основі біометричних поведінкових характеристик.

1.1 Роль та значення захищеного доступу до ІС

ІС – це сукупність організаційних та технічних засобів для збереження та оброблення інформації з метою забезпечення інформаційних потреб користувачів. Функції, які виконує інформаційна система наведено на рис. 1.1.

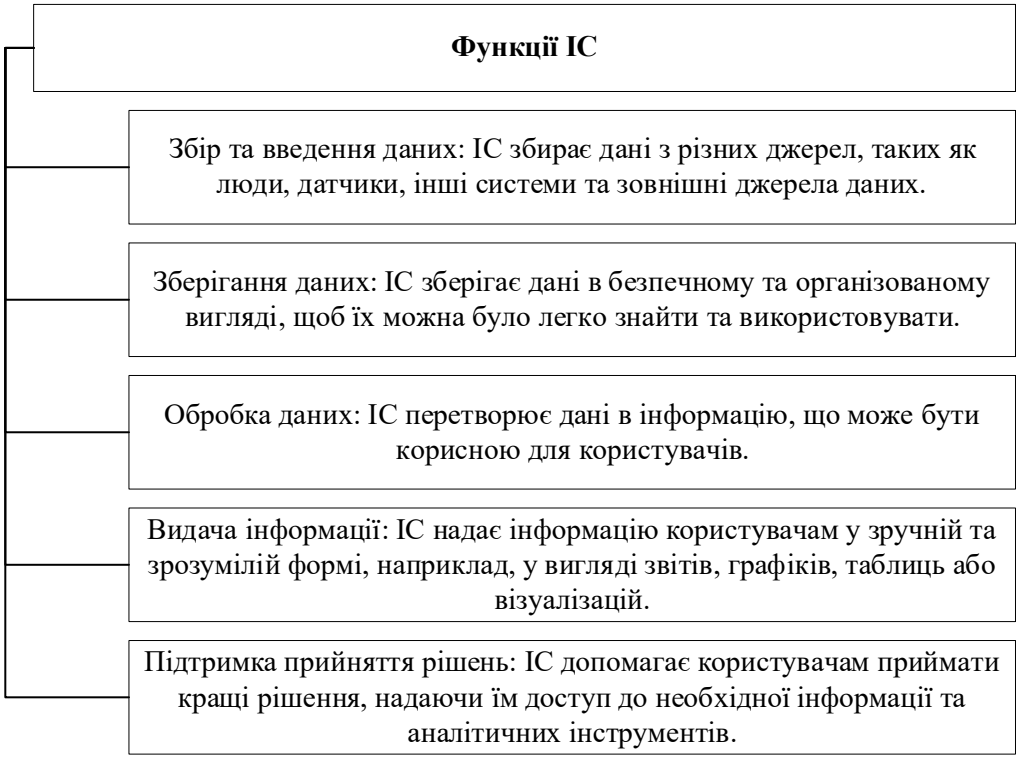


Рисунок 1.1 – Функції інформаційної системи [17]

Враховуючи можливості та функції інформаційної системи, актуальним питанням є її захист, зокрема, від несанкціонованого доступу сторонніх користувачів.

Під поняттям несанкціонованого доступу до ІС слід розуміти отримання доступу до даних, програмного забезпечення або апаратних компонентів ІС без дозволу або права власника [17].

Це може бути випадковий або навмисний акт, що може призвести до серйозних наслідків, які можуть бути представлені у вигляді крадіжки даних, порушенні конфіденційності, пошкоджені системи та як наслідок фінансових та репутаційних збитків.

Комп'ютери проникли і продовжують проникати в багато галузей. Багато заводів і великих компаній впроваджують у себе комп'ютерну техніку і створюють інформаційну інфраструктуру. На навчання персоналу і на процес переходу від паперового документообігу до електронного витрачаються великі кошти. Контроль доступу до інформаційних ресурсів стає складним завданням.

Сьогодні складно назвати точну кількість сумарних втрат від комп'ютерних злочинів, пов'язаних із НСД до захищуваної інформації. Насамперед, це пояснюють небажанням компаній, що постраждали оприлюднювати інформацію про свої втрати, а також тим, що не завжди можна провести оцінку втрат від розкрадання інформації в грошовому еквіваленті [18].

Причин поширення комп'ютерних злочинів та, відповідно, пов'язаних із цим фінансових втрат досить багато, значущими серед них є: перехід від традиційної «паперової» технології зберігання та передавання відомостей до електронної та недостатній при цьому розвиток технології захисту інформації в таких технологіях; об'єднання обчислювальних систем, створення глобальних мереж та розширення доступу до інформаційних ресурсів; збільшення складності програмних засобів та пов'язане з цим зменшення їхньої надійності та збільшенням кількості вразливостей.

Будь-яке сучасне підприємство незалежно від виду діяльності та форми власності не в змозі успішно розвиватися і вести господарську діяльність без

створення на ньому умов для надійного функціонування системи захисту власної інформації.

Відсутність у багатьох керівників підприємств і компаній чіткого уявлення з питань захисту інформації призводить до того, що їм складно повною мірою оцінити необхідність створення надійної системи захисту інформації на своєму підприємстві, а тим більше складно буває визначити конкретні дії, необхідні для захисту тих чи інших конфіденційних відомостей.

Побудова надійного захисту включає оцінку інформації, що циркулює в комп'ютерній системі, з метою уточнення ступеня її конфіденційності, аналізу потенційних загроз її безпеці та встановлення необхідного режиму її захисту.

Нині відсутня будь-яка універсальна методика, що дає змогу чітко співвідносити ту чи іншу інформацію до категорії комерційної таємниці. Можна лише порадити виходити з принципу економічної вигоди та безпеки підприємства − надмірна «засекреченість» призводить до необґрунтованого удорожчання необхідних заходів із захисту інформації та не сприяє розвитку бізнесу, коли як широка відкритість може призвести до більших фінансових втрат або розголошення таємниці.

Визначившись із необхідністю захисту інформації, безпосередньо приступають до проектування системи захисту інформації.

Конкретний зміст зазначених заходів для кожного окремо взятого підприємства може бути різним за масштабами і формами. Це залежить насамперед від виробничих, фінансових та інших можливостей підприємства, від обсягів конфіденційної інформації та ступеня її значущості. Суттєвим є те, що весь перелік зазначених заходів обов'язково має плануватися і використовуватися з урахуванням особливостей функціонування інформаційної системи підприємства.

Встановлення особливого режиму конфіденційності спрямоване на створення умов для забезпечення фізичного захисту носіїв конфіденційної інформації. Обмеження доступу до конфіденційної інформації сприяє створенню найбільш ефективних умов збереження конфіденційної інформації.

Необхідно чітко визначати коло співробітників, допущених до конфіденційної інформації, до яких конкретно відомостей їм дозволено доступ і повноваження співробітників щодо доступу до конфіденційної інформації. Як показує практика роботи, для розроблення необхідного комплексу заходів щодо захисту інформації бажано залучення кваліфікованих експертів у сфері захисту інформації. Ефективність захисту інформації досягається не кількістю грошей, витрачених на її організацію, а здатністю її адекватно реагувати на всі спроби несанкціонованого доступу до інформації; заходи щодо захисту інформації від несанкціонованого доступу повинні носити комплексний характер, тобто об'єднувати різнорідні заходи протидії загрозам (правові, організаційні, програмно-технічні); основна загроза інформаційній безпеці комп'ютерних систем походить безпосередньо від співробітників [8].

З огляду на це необхідно максимально обмежувати як коло співробітників, що допускаються до конфіденційної інформації, так і коло інформації, до якої вони допускаються (зокрема, і до інформації за системою захисту). При цьому кожен співробітник повинен мати мінімум повноважень щодо доступу до конфіденційної інформації.

Таким чином, засоби захисту від несанкціонованого доступу можна поділити на три категорії (рис. 1.2).

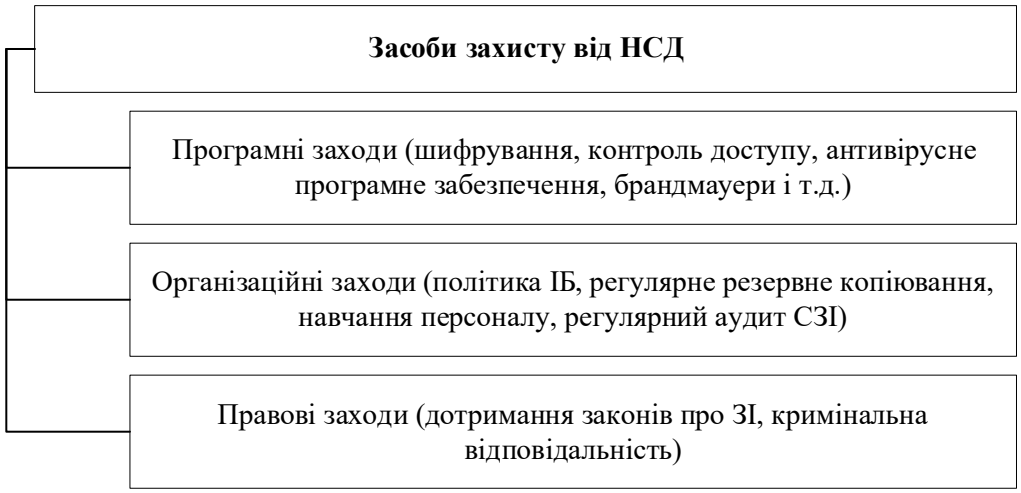


Рисунок 1.2 – Засоби захисту від несанкціонованого доступу [19]

Виходячи із поставлених задач роботи, далі проаналізуємо саме

програмні засоби захисту інформаційних систем із використанням технологій двофакторної авторизації та рольової моделі розмежування доступу.

1.2 Аналіз переваг та недоліків існуючих методів захисту ІС від несанкціонованого доступу

Комерційна таємниця, фінансові транзакції, кадрові документи можуть становити чималий інтерес для конкурентів, зловмисників.

Захист інформації від несанкціонованого доступу допомагає уникнути її нецільового використання та заподіяння шкоди власнику. Найчастіше для цього вдаються до контролю повноважень і доступу співробітників компанії до інформаційних систем.

Несанкціонований доступ (НСД) – це протиправні дії, що порушують внутрішній регламент роботи компанії або законодавство, під час яких особа, не наділена правами доступу, отримує потрібну їй інформацію [20].

Це можуть бути випадки як перевищення посадових повноважень і отримання розширеного обсягу даних, так і спроби отримати інформацію зовсім закритого типу. Будь-який подібний випадок несе загрозу розкриття конфіденційності інформації, використання її проти власника та його інтересів.

Можливі шляхи здійснення несанкціонованого доступу сторонніми особами до інформаційних систем наведено на рис. 1.3.

Найбільшу небезпеку для конфіденційності інформації несуть саме внутрішні загрози безпеці. Вони мають прихований характер і можуть не виявлятися роками.

Виходячи із наведених загроз та методів реалізації НСД, існують різні методи захисту (рис. 1.4) [21 – 22].

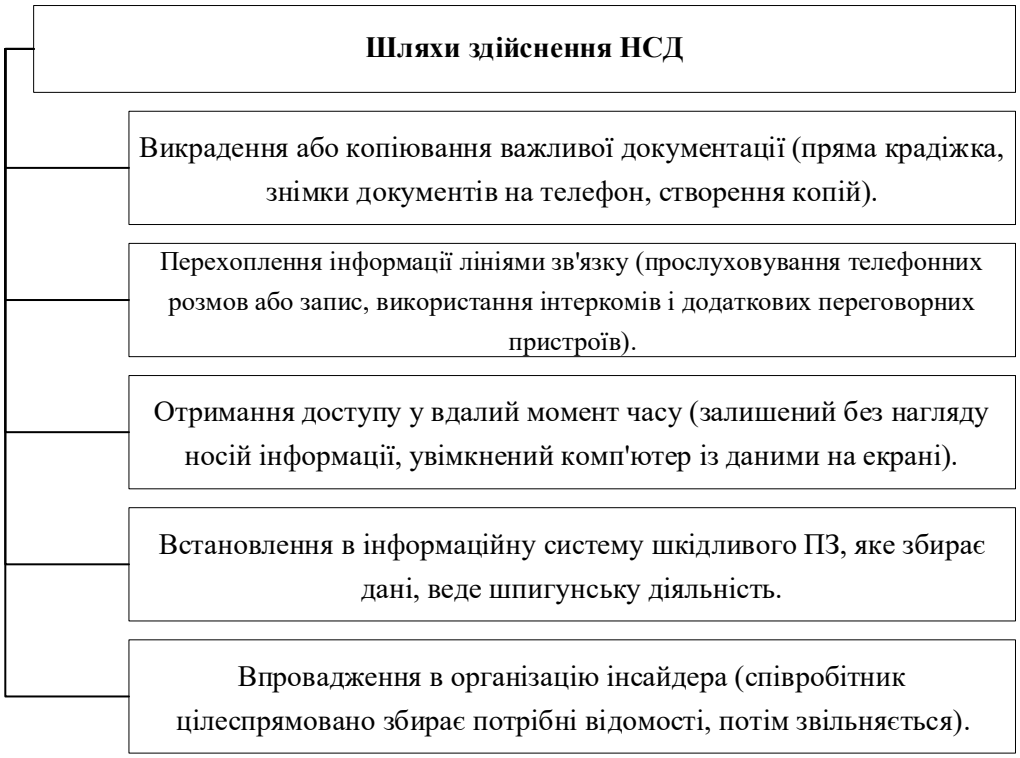


Рисунок 1.3 – Шляхи здійснення несанкціонованого доступу

Рисунок 1.4 – Розповсюджені методи захисту від несанкціонованого доступу

Розглянемо їх детальніше.

Існує кілька груп методів захисту, зокрема перешкоджання передбачуваному порушнику за допомогою фізичних і програмних засобів, управління або вплив на елементи системи, що захищається, маскування або перетворення даних із застосуванням криптографічних методів, регулювання

або розроблення законодавства та комплексу заходів, спрямованих на стимулювання правильної поведінки користувачів, що працюють із базами даних, примус або створення умов, за яких користувач буде змушений дотримуватися правил роботи з даними.

Перш ніж забезпечувати захист інформації від НСД, потрібно провести підготовчі заходи. По-перше, класифікувати дані за ступенем важливості та рівнем критичності. По-друге, провести оцінку потенційних внутрішніх загроз, наприклад, вірогідність передачі інформації власними співробітниками один одному або їхнє винесення за межі компанії.

Захист інформації від несанкціонованого доступу варто розділити на кілька основних напрямів [23]:

– запобігання (заходи профілактичного типу, пов'язані з обмеженням доступу для небажаного кола осіб);

– виявлення (заходи, спрямовані на знаходження інцидентів);

– обмеження (заходи для зниження збитків, якщо зловмисник зумів обійти захист);

– відновлення (заходи, спрямовані на реконструкцію інформації згідно зі встановленими методиками).

Проаналізуємо більш детальніше основні технічні методи захисту він несанкціонованого доступу, зокрема, що полягають у процесі шифрування, автентифікації та забезпеченні контролю доступу користувачів.

Шифрування мережевого трафіку гарантує, що дані не можуть бути перехоплені зловмисником, який підглядає за мережевим трафіком. Широко поширеним є використання шифрування для захисту мережевого трафіку, що проходить через відкритий Інтернет, зазвичай у вигляді з'єднань SSL/TLS. Однак усередині центрів обробки даних з'єднання між серверами часто не шифруються [24].

Зловмисник, який отримав доступ до такої мережі, навіть не маючи доступу до самих серверів, на яких зберігаються дані, однаково може перехопити захищені конфіденційністю дані, що передаються між серверами в

багатомашинному кластері. Крім того, організації все частіше записують і аналізують власний мережевий трафік для виявлення вторгнень у мережу. У результаті, повні копії мережевого трафіку можуть зберігатися в таких системах моніторингу протягом тривалих періодів часу. Це може призвести до ненавмисного витоку захищених даних у систему, в якій не реалізовано такі самі рівні контролю та нагляду.

Двофакторна аутентифікація вимагає пред'явлення користувачем двох частин інформації перед наданням доступу до системи. Таким чином, скомпрометувати обліковий запис легітимного користувача набагато складніше, ніж просто дізнатися його пароль [25].

Якщо першим фактором зазвичай є пароль, то другий фактор – це одноразовий код доступу, що надається спеціалізованим обладнанням або програмним забезпеченням, таким як токен або застосунок для смартфона, що виконує ту саму функцію (наприклад, Google Authenticator). Це також може бути одноразовий код, що надається через механізм зв'язку, наприклад, SMS-повідомлення або автоматичний телефонний дзвінок.

Хоча це робить аутентифікацію тривалою, це змушує будь-якого потенційного зловмисника не тільки скомпрометувати пароль користувача, а й фізичний пристрій, що перебуває під його контролем.

Для ще більшої безпеки трьохфакторна автентифікація може супроводжувати кожен успішний вхід до системи електронним або SMS-повідомленням, щоб користувач міг швидко отримати попередження про несанкціонований доступ до облікового запису та негайно повідомити про це.

Захист конфіденційності – це насамперед регулювання санкціонованого доступу до даних та їх використання [26]. Інформаційна безпека, яка насамперед спрямована на запобігання несанкціонованому доступу до інформації, – це те, що робить захист конфіденційності можливим. Без контролю несанкціонованого доступу створення режиму захисту конфіденційності для авторизованих користувачів не має сенсу, оскільки будь-який захист, який можна легко обійти, не є справжнім захистом взагалі.

Таким чином, захист інформаційних систем від несанкціонованого доступу є критично важливим завданням для забезпечення конфіденційності, цілісності та доступності даних. Існує безліч методів захисту ІС про які було згадано вище, кожен з яких має свої переваги та недоліки. Узагальнимо та порівняємо наведені методи у табл. 1.1.

Таблиця 1.1 – Переваги та недоліки методів захисту від НСД [27 – 28]

Тип методу	Метод	Переваги	Недоліки
Програмні	Шифрування	Високий рівень безпеки	Потребує багато ресурсів Складний у реалізації
	Брандмауери	Захист від зовнішніх загроз	Не захищають від внутрішніх загроз
	Антивірусне ПЗ	Виявляє та видаляє шкідливе ПЗ	Не може виявити всі типи загроз
	IDS/IPS	Виявляє та блокує підозрілу активність	Можуть генерувати багато помилок
Організаційний	Контроль доступу	Обмежує доступ до ресурсів	Може бути складним для адміністрування
	Політика безпеки інформації	Визначає правила та процедури	Не гарантує дотримання правил
	Навчання персоналу	Підвищує обізнаність про ризики	Не гарантує безпечної поведінки
	Резервне копіювання та відновлення	Відновлення даних у разі збою	Не захищає від НСД
	Регулярний аудит	Виявлення вразливостей	Не гарантує постійний захист

Вибір методу захисту від несанкціонованого доступу залежить також від поставлених задач. Метою даної роботи є забезпечення саме контролю доступу користувачів до інформаційної системи та забезпечення якісної процедури ідентифікації.

Виходячи із поставлених цілей доцільно обрати для роботи один із існуючих засобів контролю доступу у вигляді флеш-носія, а також біометричний метод ідентифікації, як один із найбільш ефективних.

1.3 Засоби застосування ключів у системах безпеки

Аналізуючи ситуації із забезпечення захисту інформаційних систем, часто трапляються ситуації, коли працівники записують паролі на папірцях, які лежать на робочих столах або приклеєні до моніторів.

Це підвищує ймовірність крадіжки конфіденційної інформації, або створює умови для порушення доступу до важливих даних.

У такому випадку витік і слабкість паролів стає серйозною проблемою для мережевих адміністраторів і відповідальних за інформаційну безпеку.

Компанії намагаються розв'язувати дану проблему і одним із способів вирішення поставленої задачі може бути застосування електронних ключів – USB брелків, смарт-карт та інших апаратних аутентифікаторів [29].

Дане рішення є доцільним та практично значущим за таких умов:

– процес переходу зі звичайного методу аутентифікації (за паролями) на двофакторний метод (за допомогою USB-ключів) чітко спланований;

– у компанії є кваліфікований персонал для обслуговування такої системи;

– забезпечення всебічної підтримки з боку виробника.

USB-ключі можуть бути ефективним інструментом для контролю доступу до комп'ютерів, мереж та інших ресурсів, а головною умовою їх використання повинна бути готовність компанії до цього, яка характеризується наведеними вище умовами [20].

Переваги використання ключів для використання контролю доступу наведено на рис. 1.5.

Порівняємо між собою два найбільш популярні типи ключів: у вигляді смарт-карти та флеш-носія. Для цього більш детальніше розглянемо принцип їх роботи.

Смарт-карта – це пластикова картка з вбудованим мікрочипом (мікропроцесор і пам'ять), на зразок банківської або телефонної картки (рис. 1.6) [30].

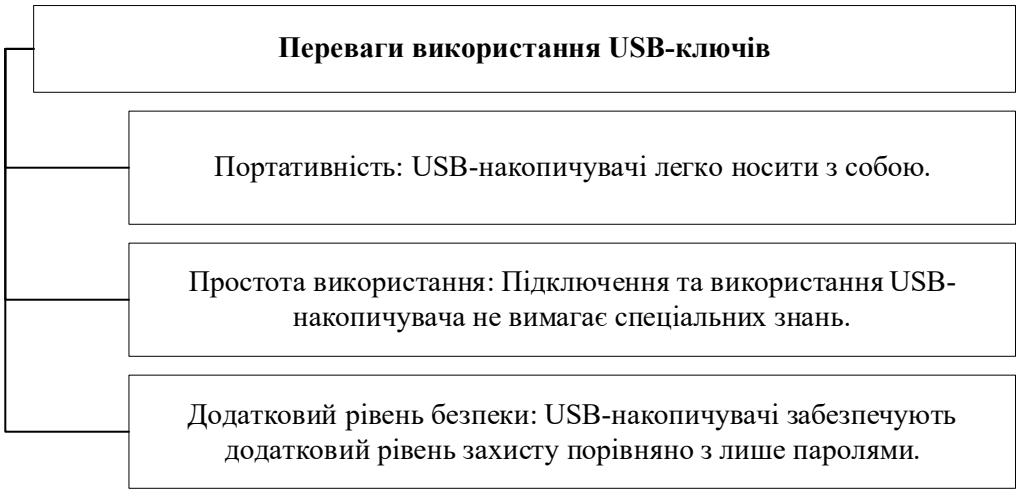


Рисунок 1.5 – Переваги використання USB-ключів

Для роботи зі смарт-картками потрібен спеціальний пристрій – смарт-карт зчитувач, який працює із використанням відповідних драйверів. Смарт-картку необхідно підготувати (відформатувати\персоналізувати) перед першим використанням за допомогою спеціальної програми.

Рисунок 1.6 – Зовнішній вигляд смарт-картки [30]

Розмір пам'яті смарт-картки дуже малий (кілька кілобайт), оскільки пам'ять у смарт-картках призначена тільки для зберігання малих об'ємів інформації (цифрових сертифікатів, ключів шифрування, паролів). Тому перед використанням смарт-карти та її залученням у системи безпеки варто переконатися, що розмір пам'яті підійде для поставлених цілей. Оскільки можливі варіанти, коли деяким працівникам необхідно мати доступ до багатьох

систем, тому їхні смарт-картки повинні містити більше інформації.

Програми, які працюють зі смарт-картками зазвичай підтримують саме необхідний тип смарт-карток (або кілька), тому необхідно переконатися що для смарт-карток, які у є в розпорядженні, можна знайти програмне забезпечення для ваших потреб.

USB брелок – це симбіоз смарт-карт зчитувача і смарт-карти. У USB брелок (або токен) впаяний мікрочіп від смарт-карти. Для використання USB брелока необхідно інсталювати драйвера (вставляння / витягування аналогічно вставлянню / витягування картки в зчитувач) [31].

Рисунок 1.7 – Зовнішній вигляд USB-ключа [24]

Щоб почати використовувати USB-носій у застосунках, його теж необхідно підготувати (відформатувати) спеціальною утилітою. Пам'ять USB-ключа також мала і призначена тільки для зберігання даних малого розміру.

Порівняємо два наведених типи електронних ключів на практичних прикладах. Наприклад, чи може зловмисник зробити дублікат смарт-картки або USB-носія, якщо він опиниться у нього в руках?

Для смарт-картки, якщо на ній встановлено PIN код, то дублікат зробити неможливо. Якщо PIN коду немає, то можливо, але за допомогою спеціальної утиліти. Усе залежить від конкретного типу смарт-картки. У деяких випадках виконати повноцінний дублікат неможливо.

Для USB-носія це неможливо, оскільки за замовчуванням login-профайл «прив'язаний» до серійного номера USB накопичувача. А якщо є PIN код, то це

ще більше перешкоджає скористатися цим ключем, і дізнатися пароль зловмисник не зможе. А в деяких випадках і відсутність PIN коду не дає змоги дізнатися пароль, зокрема, якщо здійснена прив'язка USB-носія до комп'ютера.

Далі розглянемо таку ситуацію: якщо використовувати USB-носій або смарт-картку для авторизації в системі, то чи можуть співробітники використати їх також для виносу конфіденційної інформації за межі компанії або навпаки принести в компанію вірус.

Для смарт-картки дана ситуація безпечна, оскільки копіювати файли на смарт-картку не можливо.

Для USB-носія також дану ситуацію можна вважати безпечною, оскільки заборонити доступ до накопичувача можна ще на початковій стадії розробки, що відповідно унеможливить запис на нього будь-яких даних.

Таким чином, якщо йдеться про безпеку у великій компанії, яка готова витратити гроші на спеціалізовані пристрої, якими є смарт-картки та USB-носії, то саме вони будуть найкращим вибором. Якщо поставлена задача обмежити доступ до інформації на особистому комп'ютері або підвищити надійність паролів в офісі, то найкращим варіантом є USB накопичувач.

Отже, виходячи із універсальності застосування USB-носіїв, саме вони будуть обрані для подальшої роботи в якості одного із факторів забезпечення захищеного доступу до інформаційної системи з метою запобігання НСД.

1.4 Вивчення можливостей автентифікації користувачів на основі їх біометричних даних

Біометрія – це вимірювання та аналіз унікальних фізичних і поведінкових характеристик людей. Біометрична технологія в основному використовується для підтвердження особи людини. Основна передумова, пов'язана з цим, у тому, що кожну людину можна точно ідентифікувати за її фізичними або поведінковими ознаками [32].

Біометрична автентифікація – це процес забезпечення безпеки, який заснований на технології розпізнавання особи за її біологічними

характеристиками. Унікальність цих характеристик дає змогу підтвердити, чи є людина саме тим користувачем, за якого себе видає. Біометричні дані людини завантажуються в базу даних і є еталонними. А далі під час запиту доступу до інформаційного ресурсу системи біометричної автентифікації забирають введені дані користувача і порівнюють їх з еталонними, що зберігаються в базі. Якщо дані збігаються, то аутентифікація вважається успішною, підтвердженою і користувач отримує доступ до ресурсу.

Біометричну автентифікацію можна використовувати для управління як фізичним доступом на об'єкти (будівлі або окремі приміщення), так і під час допуску до різних інформаційних ресурсів (систем, додатків, баз даних тощо).

Звісно, біометрична автентифікація підвищує безпеку, бо складно підробити індивідуальні біологічні параметри людини, але вона ще й дуже зручна, бо неможливо забути чи загубити свою біометрію.

Аутентифікація з використанням біометрії стала дуже поширеною в корпоративних середовищах безпеки. Але крім безпеки драйвером розвитку такої технології є зручність для користувачів. У багатьох випадках це рішення дає змогу зовсім відмовитися від паролів (не потрібно їх запам'ятовувати і десь зберігати), а також деякі біометричні методи працюють без прямого контакту з людиною, наприклад, такі як зчитування обличчя або ходи людини [33].

Біометрична технологія складається з таких компонентів (рис. 1.8).

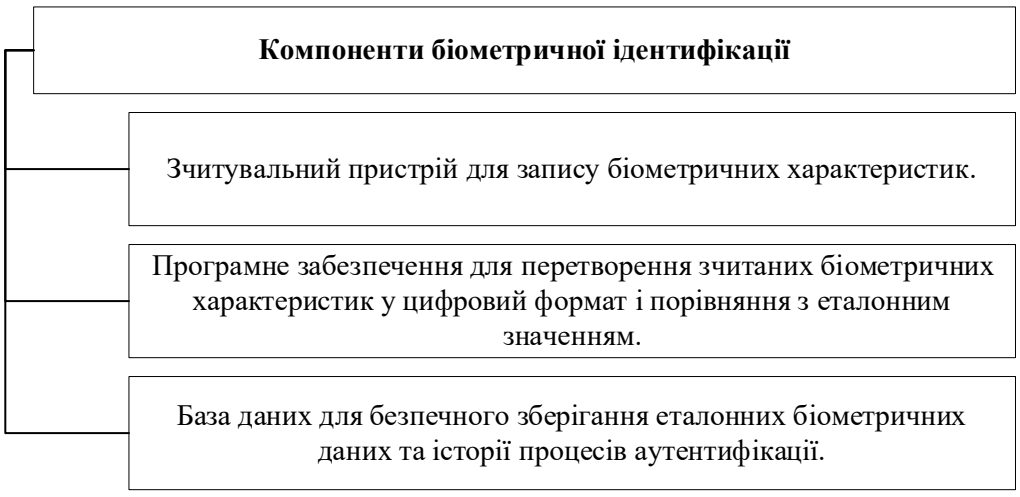


Рисунок 1.8 – Складові компоненти біометричної ідентифікації [27]

Біометрію можна умовно розділити на дві частини: дані, які залежать від фізіології, і дані, які залежать від поведінкових характеристик. Види біометричної ідентифікації наведемо у табл. 1.2.

Таблиця 1.2 – Види біометричної ідентифікації [34 – 35]

Фізіологічні характеристики (постійні)	Відбиток пальця Малюнок сітківки ока Риси обличчя Звук голосу Малюнок вен долоні Малюнок райдужної оболонки ока
Поведінкові характеристики (динамічні)	Хода Манера набору тексту (швидкість, розташування рук) Сила натискання клавіш Жести

Наведені види біометричної ідентифікації та біометрія вцілому набула досить широкого поширення в різних сферах. Це і фінансові послуги, і охорона здоров'я, і правоохоронна система, і освіта, та й багато інших галузей і напрямів діяльності людини. Аутентифікація за допомогою біометрії, безумовно, підвищує безпеку і захищає активи. Актуальність застосування даної технології зумовлюється рядом переваг (рис. 1.9).

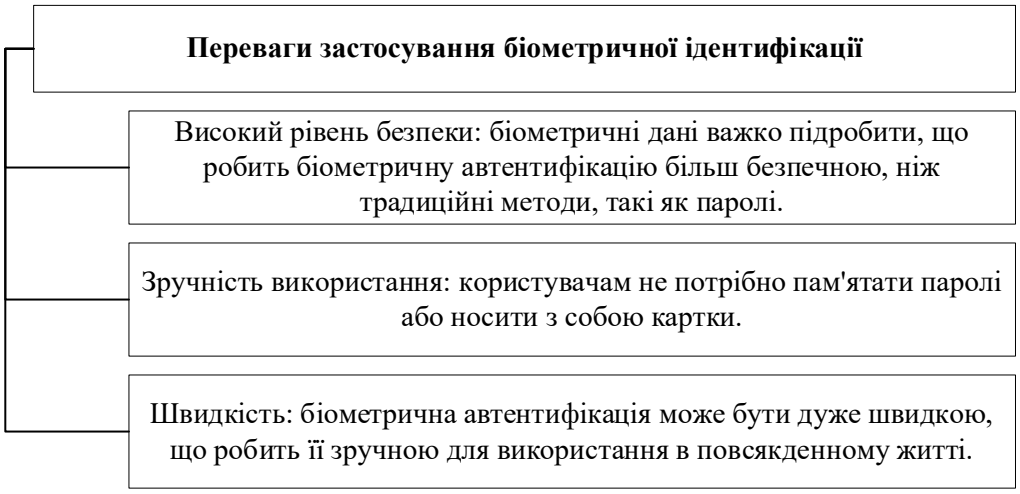


Рисунок 1.9 – Переваги застосування біометричної ідентифікації [36]

Далі розглянемо системи ідентифікації особи на основі біометричних даних.

Біометричні системи аутентифікації працюють у двох режимах [37 – 38]:

1. Режим реєстрації. У даному режимі реалізується отримання відповідних даних про користувача. Даний процес виконується за допомогою певних типів біометричних зчитувальних пристроїв (наприклад, сканери). Далі уся отримана інформація зберігається в базі даних разом з персональними даними користувача (ім'я, електронна пошта тощо) для полегшення подальшого процесу аутентифікації.

2. Режим автентифікації. Отримані під час процесу реєстрації. біометричні дані використовуються захищуваною системною для перевірки того факту, чи є користувач власне тією особою за яку він себе видає, за кого він себе видає. Дана функція реалізується за допомогою порівняння біометричних даних користувача з еталонними, що містяться у базі даних. Якщо заявлені дані користувача присутні у базі даних, то такий користувач може прийматися за істинного і навпаки.

Загалом, будь-яка біометрична система складається з таких компонент (рис. 1.10).

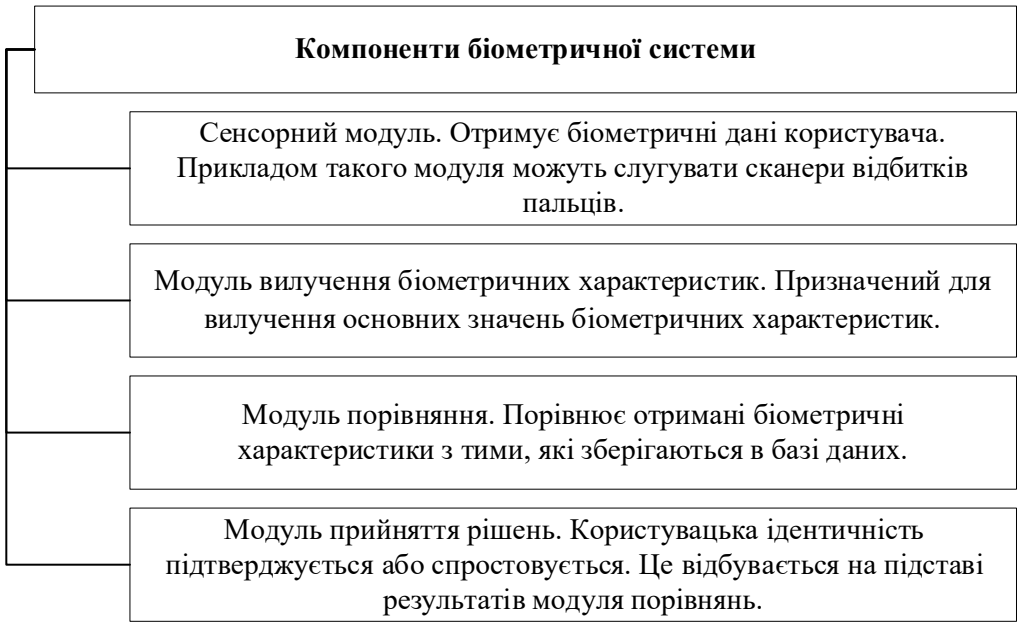


Рисунок 1.10 – Складові компоненти біометричної системи

Оцінювання систем біометричної автентифікації здійснюється на підставі помилок першого (FRR) і другого (FAR) роду. Помилка першого роду (FRR − False Rejection Rate) − це ймовірність помилкової відмови в доступі клієнту, який має право доступу [38]. Помилка другого роду (FAR − False Acceptance Rate) − це ймовірність помилкового доступу, коли система помилково впізнає чужого як свого. Обчислення їх проводиться за формулами [38]:

$$FRR = \frac{\text{загальне число помилкових відмов у доступі}}{\text{загальне число тестованих об}'\text{єктів}},$$

$$FAR = \frac{\text{загальне число помилкових доступів}}{\text{загальне число тестованих об}'\text{єктів}}.$$

Показник істинного прийняття (Genuine Accept Rate – GAR) – це ймовірність істинного прийняття користувача, який має право доступу:

$$GAR = 1 - FRR$$

В ідеальному варіанті система біометричної аутентифікації мала б такі показники:

$$FAR = 0 \text{ і } FRR = 0.$$

Однак, такі значення в реальності є складно досяжними. Значення будь-якої з двох помилок можливо зменшити. Недолік цього в тому, що друге значення зросте. Зазвичай системні параметри налаштовують так, щоб домогтися необхідного коефіцієнта помилкових підтверджень, що визначає відповідний коефіцієнт помилкових відмов.

Таким чином, можна підсумувати, що застосування біометричних даних є досить ефективним та надійним методом для застосування у системах забезпечення захисту інформаційних систем від несанкціонованого доступу. Наведені переваги та методи застосування біометричної ідентифікації уможлювлюють її практичне використання у роботі.

1.5 Висновки та постановка задач

У даному розділі було проаналізовано роль та значення захищеного доступу до ІС, наведено функції ІС, досліджено засоби захисту від несанкціоно-

ваного доступу, що містять організаційні, правові та програмні заходи.

Здійснено аналіз застосування електронних ключів у системах безпеки, проведено порівняння смарт-карток та флеш-носіїв, вивчено можливості автентифікації користувача на основі біометричних характеристик, зокрема, клавіатурного почерку.

Проаналізовано переваги та недоліки існуючих методів захисту ІС від НСД, підібрано оптимальні засоби захисту для розроблюваної системи на основі удосконаленого методу.

У результаті проведеного аналізу та, виходячи з мети магістерської роботи, необхідно у наступних розділах виконати такі задачі:

– удосконалити метод захисту інформаційних систем від НСД шляхом застосування двофакторної ідентифікації;

– створити дуальний підхід до перевірки унікальних біометричних даних користувача на основі клавіатурного почерку;

– побудувати підхід до використання зашифрованого флеш-носія для реалізації захищеного доступу;

– розробити БД для збереження еталонних даних користувачів та оптимізовано процес звернення неї на основі генетичних алгоритмів;

– на базі складеного підходу запропонувати алгоритм роботи відповідного програмного засобу та здійснити його реалізацію;

– протестувати роботу ПЗ та проаналізувати отримані результати;

– обґрунтувати економічну доцільність впровадження запропонованої розробки.

Виконання поставлених завдань роботи дозволяє покращити захищеність інформаційної системи за рахунок використання контролю доступу та біометричної ідентифікації, а також оптимізувати звернення до бази даних за рахунок використання генетичного алгоритму.

Подальше дослідження передбачає розроблення та тестування програмного засобу для захисту інформаційної системи на основі удосконаленого методу.

2 УДОСКОНАЛЕННЯ ЗАХИСТУ ІС НА ОСНОВІ ЗАХИЩЕНОГО ФЛЕШ-НОСІЯ, БІОМЕТРИЧНИХ ДАНИХ ТА ОПТИМІЗАЦІЯ ЗВЕРНЕНЬ ДО БД

У даному розділі описано удосконалення методу захищеного доступу до інформаційної системи на основі флеш-носія та біометричних даних. Використовуваний флеш-носій пропонується захищати на основі криптографічного алгоритму AES та паролювання. Біометричні дані користувача представлено у вигляді клавіатурного почерку, що зчитується у випадку відомої та випадкової фрази для користувача. Описано процес оптимізації звернень до бази даних для забезпечення швидкодії роботи програмного модуля із використанням генетичного алгоритму.

2.1 Удосконалення методу захищеного доступу до ІС та розроблення загальної структурної моделі такого процесу

Розглянемо розроблення методу захищеного доступу до ІС, що уможливлює удосконалення такого процесу з використанням захищеного флеш-носія, двофакторної ідентифікації на основі аналізу клавіатурного почерку.

Суть вдосконалення полягає у комплексному підході до процесу авторизації користувача з використанням зовнішнього носія інформації та унікальних біометричних даних.

Застосування даного підходу захищеного доступу до інформаційних систем має низку переваг, розглянутих на рис. 2.1.

Слід звернути увагу на можливість захисту інформаційних систем від брут-форс атак та зручності використання користувачами. Біометричні дані користувачів для їх автентифікації в системі є на сьогодні одним із найпопулярніших та затребуваних методів. Зокрема перевірка клавіатурного почерку не вимагає дороговартісного додаткового обладнання, має допустимі межі у складності реалізації та проста на практиці для користувачів.

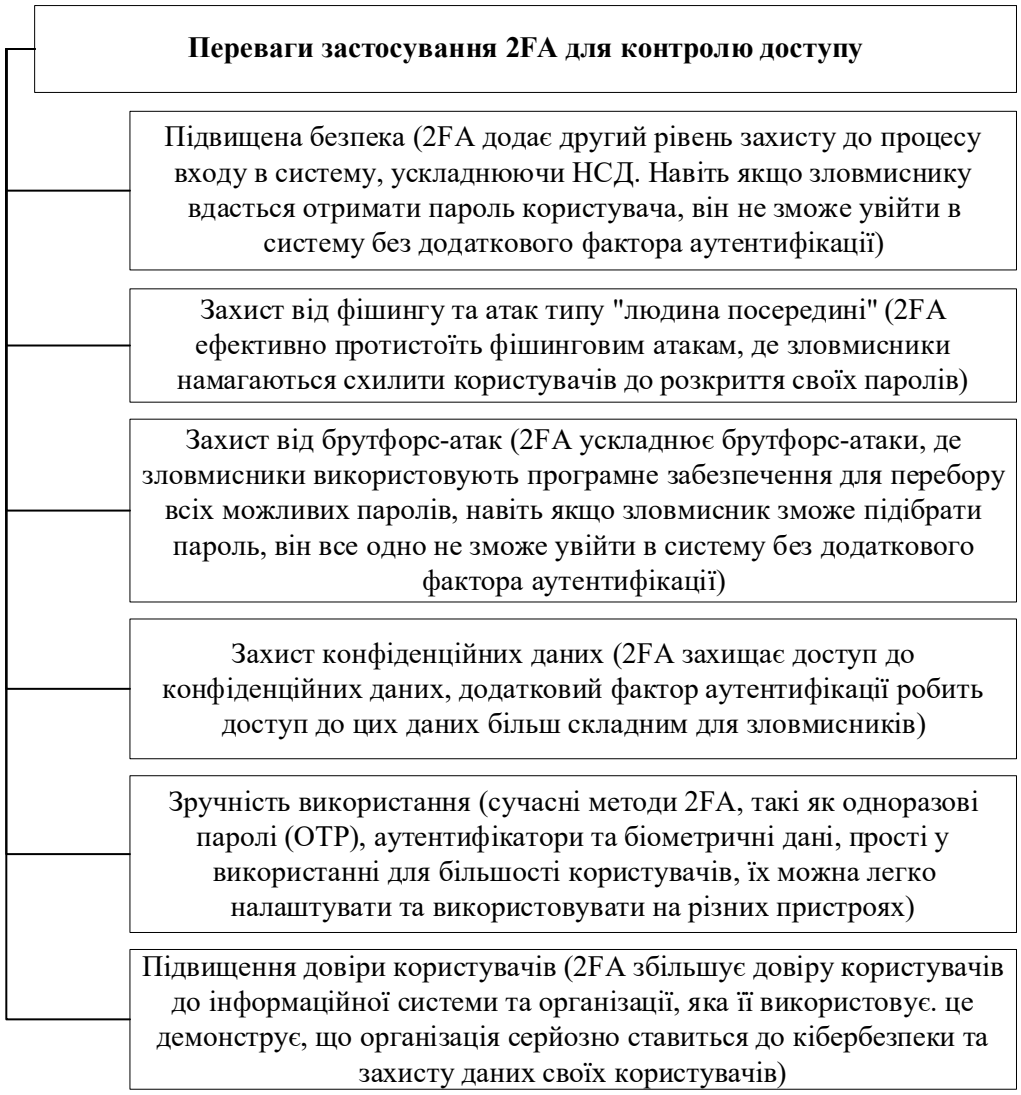


Рисунок 2.1 – Переваги використання вдосконаленого методу двохфакторної автентифікації (2FA) для захищеного доступу до ІС

Структурну модель процесу удосконалення контролю доступу та ідентифікації користувача ІС представимо на рис. 2.2.

Розглянемо більш детально кожен із поданих на рис. 2.2 етапів.

Етап 1. Введення логіну користувачем. Логін користувача є його ідентифікатором, який буде використовуватися для входу в систему, мінімальна вимога – довжина не менше 5 символів.

Етап 2. Введення паролю користувачем. Пароль слугує ідентифікатором, що використовується для ідентифікації за клавіатурним почерком за першим методом – введення заздалегідь відомої фрази користувачем. Мінімальна вимога до паролю – довжина не менше 8 символів.

Рисунок 2.2 – Структурна модель удосконаленого методу

Етап 3. Перший крок перевірки клавіатурного почерку на основі введеного користувачем паролю.

Етап 4. Передавання даних про результати першого кроку до бази даних. Отримані в результаті введення паролю дані про клавіатурний почерк порівнюються з еталонними у базі даних.

Етап 5. Завантаження даних флеш-носія користувачем. Відбувається користувачем шляхом вибору шляху до носія.

Етап 6. Зчитування флеш-носія доданого користувачем. На носії містяться певні зашифровані дані, які слугують другим етапом авторизації користувача.

Етап 7. Другий крок введення клавіатурного почерку на основі введеної користувачем відповідні на випадкове питання. Відповідь користувача є його ідентифікатором, який буде щоразу різного змісту, мінімальна вимога –

довжина не менше 20 символів.

Етап 8. Перевірка другого етапу клавіатурного почерку на основі введеної користувачем відповіді на випадкове питання. Дана відповідь слугує ідентифікатором, що використовується для ідентифікації за клавіатурним почерком за другим методом – введення заздалегідь невідомої фрази користувачем.

Етап 9. Процес перевірки даних користувача. Комплексна перевірка отриманих даних після двох етапів зчитування клавіатурного почерку та даних з флеш-носія.

9.1. З використанням генетичного алгоритму – звернення до бази даних додатку для перевірки даних користувача.

9.2. Отримання відповіді про результати перевірки користувача.

Етап 10. Прийняття рішення про надання доступу користувачеві до системи.

10.1. У випадку, якщо усі дані надані користувачем відповідають еталонам у базі даних – користувачеві надається доступ до інформаційної системи.

10.2. У випадку, якщо дані надані користувачем не відповідають хоча б одному еталонному зразку у базі даних – користувачеві відмовляється у доступі до інформаційної системи.

Етап 11. Надання прав доступу користувачеві, що надається відповідною роллю користувача системі (наприклад, робітник 1 чи робітник 2).

Таким чином, удосконалений метод комплексної перевірки користувачів дозволяє забезпечити якісний процес авторизації. Недоліком клавіатурного почерку може бути зміна поведінки користувача залежно від того відома чи невідома йому фраза, зміна типу клавіатури і т.п. Проте, за рахунок того, що в базі даних зберігається кілька еталонних зразків почерку користувача, а особливості його друку перевіряються як для відомих, так і невідомих заздалегідь фраз, забезпечується максимальна точність аналізу.

Авторизація за фактом перевірки флеш-носія не лише дозволяє забезпечити фізичний контроль доступу до системи певним користувачем, але і контролювати його під час роботи в системі. Оскільки перевірка наявності флеш-носія у USB-порті відбувається регулярно через малі проміжки часу, скористатись носієм одноразово тільки для отримання доступу у користувача не можливо.

2.2 Побудова підходу до використання флеш-носія для реалізації захищеного доступу

Використання флеш-носіїв для реалізації захищеного доступу до комп'ютерів та інших систем стає дедалі популярнішим. Цей метод пропонує ряд переваг, таких як портативність, зручність використання та порівняно невисока вартість. Однак важливо правильно реалізувати цю систему, щоб гарантувати її безпеку.

Робота із флеш-носієм для надання доступу користувачеві до інформаційної системи представлено послідовністю таких процедур (рис. 2.3).

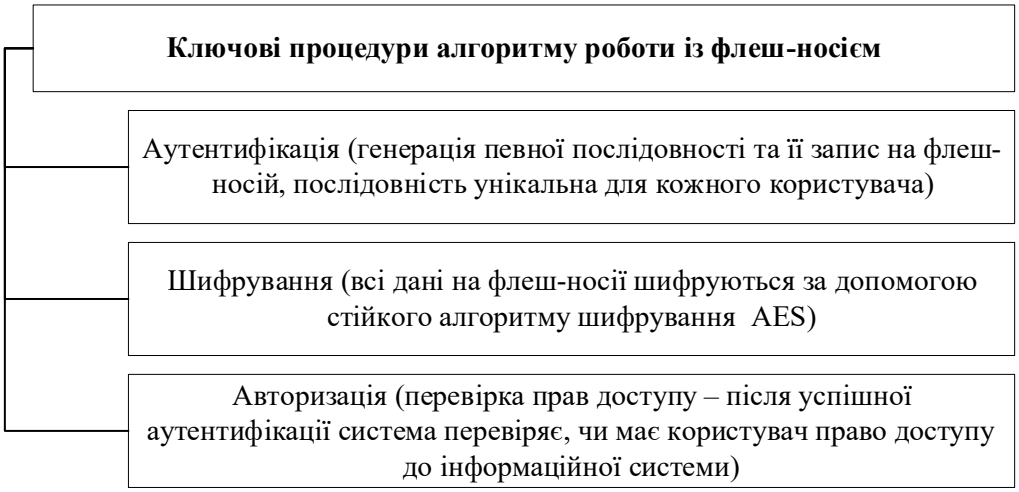


Рисунок 2.3 – Ключові процедури роботи із флеш-носієм

Розглянемо наведений підхід детальніше.

1. Пароль користувача для підключеного флеш-носія. Пароль вводиться користувачем на етапі його реєстрації в системі. Валідність паролю

визначається кількістю символів (не менше восьми), а також наявністю хоча б однієї літери верхнього регістру та одного символу. Пароль вводиться користувачем двічі та в подальшому може бути змінений лише за погодженням з адміністратором захищуваної системи. Пароль користувача є частиною інформації на флеш-носії у зашифрованому вигляді.

2. Шифрування даних на флеш-носії. На знімному пристрої, що використовується у вигляді ключа для доступу до системи знаходиться прихований шифрований файл із даними згенерованими на основі відомостей при реєстрації. Усі дані оброблені та зашифровані криптографічним алгоритмом AES (Advanced Encryption Standard) – це симетричний алгоритм блочного шифрування, який використовується для криптографічного захисту даних [39].

Застосування такого криптографічного алгоритму зумовлено низкою його переваг, а саме:

– AES стійкий до відомих атак, практика підтверджує, що він забезпечує високий рівень безпеки;

– AES є відносно швидким алгоритмом, що робить його придатним для використання в різних застосуваннях, без уповільнення роботи програмних розробок;

– AES може використовуватися для шифрування даних різного типу, таких як текстові файли, зображення та відео, що не викликає проблем із обробкою даних будь-якого формату;

– AES є стандартизованим алгоритмом, що робить його широко сумісним з різними програмними та апаратними засобами.

Далі розглянемо особливості даного алгоритму шифрування, що важливі для подальшої практичного розроблення програмного додатку захищеного доступу до ІС:

– розмір блоку даних для оброблення за алгоритмам AES 128 біт;

– для шифрування даних використовується ключ довжиною 128 біт.

– алгоритм проходить 11 раундів, під час кожного з яких відбувається

оброблення захищуваних даних, такими операціями є заміна байтів, перестановка байтів та додавання ключів;

– результатом проходження всіх раундів є зашифрований блок даних розміром 128 біт.

Блок-схему алгоритму роботи AES наведено на рис. 2.4.

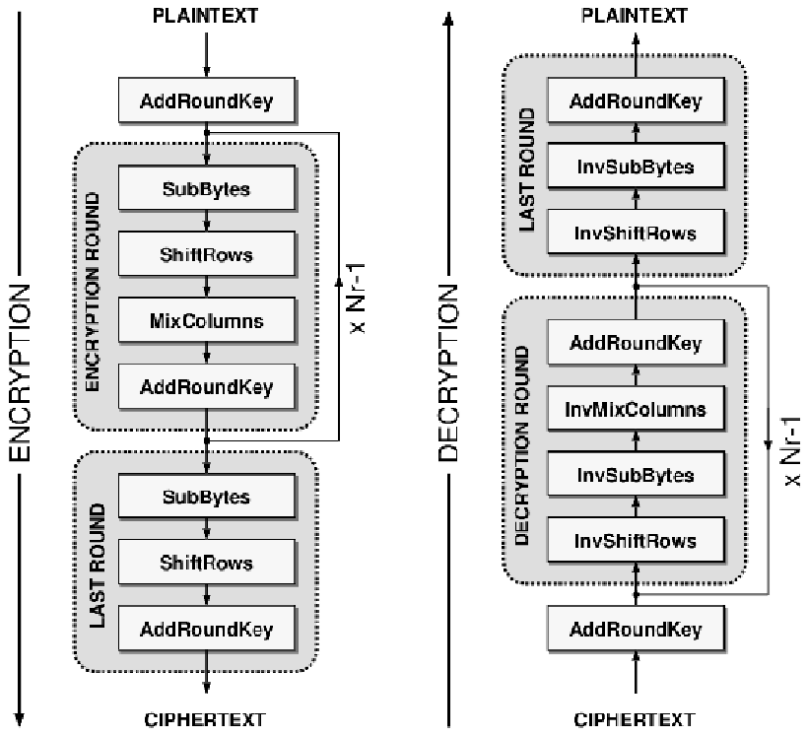


Рисунок 2.4 – Блок-схема алгоритму роботи AES [40]

3. Авторизація користувача. Після того, як програмним модулем отримано дані користувача у вигляді флеш-носія, паролю та відповіді на рандомне запитання – відбувається повторне звернення до бази даних. Якщо отримані дані відповідають наявним у базі даних та біометрична перевірка пройдена успішно, дозволяється авторизація користувача в системі із подальшим наданням йому прав доступу відповідно до ролі, визначеної адміністратором.

Таким чином, процес оброблення флеш-носія під час автентифікації та авторизації користувача у захищеній інформаційній системі є одним із ключових етапів та потребує детальної програмної реалізації із дотриманням усіх вимог описаного алгоритму.

2.3 Побудова дуального підходу до перевірки унікальних біометричних даних користувача

Як унікальний біометричний ідентифікатор для ідентифікації користувача в системі обрано поведінкову характеристику особи – клавіатурний почерк.

Клавіатурний почерк [41] – це унікальний патерн набору тексту людиною, який можна використовувати для її ідентифікації. Ідея методу полягає в тому, що кожна людина має унікальний спосіб набору тексту, який характеризується такими параметрами як:

– швидкість набору (час, який витрачається на введення кожної клавіші);

– час утримування клавіш (час, протягом якого клавіша натиснута);

– інтервали між натисканнями (час між натисканням клавіш);

– послідовність натискання клавіш (порядок натискання клавіш для введення слів або фраз);

– використання клавіш-модифікаторів (частота використання клавіш Ctrl, Shift, Alt та інших клавіш-модифікаторів).

Усі алгоритми розпізнавання користувача за клавіатурним почерком можна розділити на ті, де ідентифікація відбувається за набором довільного або ті, що мають фіксований (заздалегідь заданий) текст, наприклад, короткого пароля. Алгоритми першого типу дають змогу відстежувати несанкціоновану підміну оператора, у той час як алгоритми другого типу простіші в реалізації.

З метою підвищення достовірності та надійності результатів біометричної ідентифікації, в даній роботі застосовується дуальний метод перевірки клавіатурного почерку, що відповідно поєднує ці два види.

У будь-якому разі система включає два режими роботи: навчання та ідентифікація [42].

Алгоритм навчання застосовується, коли новий користувач «реєструється» в системі, тобто вводить свій логін і залишає зразок свого клавіатурного почерку. Усі зареєстровані системою натискання клавіш аналізуються. Для зменшення статистичної похибки пропонується зчитувати

відомості про мінімум 10 натискань.

Для кожної клавіші підраховується час утримання ($T_{\text{ут.}}$), вимірюваний у мілісекундах:

$$T_{\text{ут.}} = t_{up} - t_{down}$$

Для кожної літери українського та англійського алфавіту, натиснутої користувачем, обчислюється середній час утримання:

$$\overline{T_{\text{ут.}}} = \frac{\sum_{i=1}^{n} T_{\text{ут.}(i)}}{n}$$

де n – загальна кількість натискань на клавішу і, здійснених користувачем.

Середній $T_{\text{ут.}}$ розраховується окремо для кожної клавіші з урахуванням наявності або відсутності накладень.

Далі відбувається розрахунок пауз між натисканнями:

$$pause = t_{down} - t_{up}$$

Якщо значення паузи становить понад 1000 мс, це значення відфільтровується як таке, що не підходить для аналізу. Це відбувається тому, що тривала пауза між натисканнями свідчить уже не про динаміку друку, а про відволікання користувача від роботи з клавіатурою. Можливо, людина задумалася або відволіклася, відійшла від комп'ютера тощо.

Після цього розраховується середній час пауз для зразка клавіатурного почерку:

$$\overline{pause} = \frac{\sum_{i-1}^{n} pause_i}{n},$$

де n – загальна кількість пауз між натисканнями, за винятком відфільтрованих.

Кількість помилок введення приймаємо рівною кількості натискань на клавіші Delete і Backspace:

$$arrors = delete + backspece.$$

Нехай:

P – міра несхожості;

A – значення ВУК-1 у мс;

B – значення $T_{\text{ут.}}$ у мс;

N – кількість елементів в аналізованій вибірці;

S – матриця коваріації.

На основі вказаних параметрів, розрахуємо наступні значення.

Евклідова міра обчислюється за формулою:

$$P = \sqrt{\sum_{i=1}^{N} (A_i - B_i)^2},$$

Мангеттенська відстань обчислюється таким чином:

$$P = \sum_{i=1}^{N} |A_i - B_i|,$$

відстань Махаланобіса:

$$P = \sqrt{(\vec{A} - \vec{B})^T S^{-1} (\vec{A} - \vec{B})}.$$

Отримані дані записуються в базу даних як зразок клавіатурного почерку користувача. Усього пропонується зберігати до 10 зразків почерку для кожного користувача. Це дасть змогу передбачити можливість зміни клавіатурного почерку користувача залежно від часу доби та використовуваної клавіатури.

В алгоритмі розпізнавання зчитуються відомості про 8 натискань (для першого типу) та 20 натискань (для другого типу) на клавіатуру і надсилаються на сервер для аналізу. Для отриманого зразка клавіатурного почерку розраховуються час утримання клавіш, тривалість пауз, кількість помилок введення, визначається наявність накладень.

Після цього отриманий зразок по черзі порівнюється з кожним зразком клавіатурного почерку (еталоном), наявним у базі. За результатами порівняння приймається відповідне рішення. Алгоритм процесу порівняння початково записаного у базі еталонного зразку почерку та отриманого у ході авторизації користувача у захищуваній системі наведено на рис. 2.5 [43].

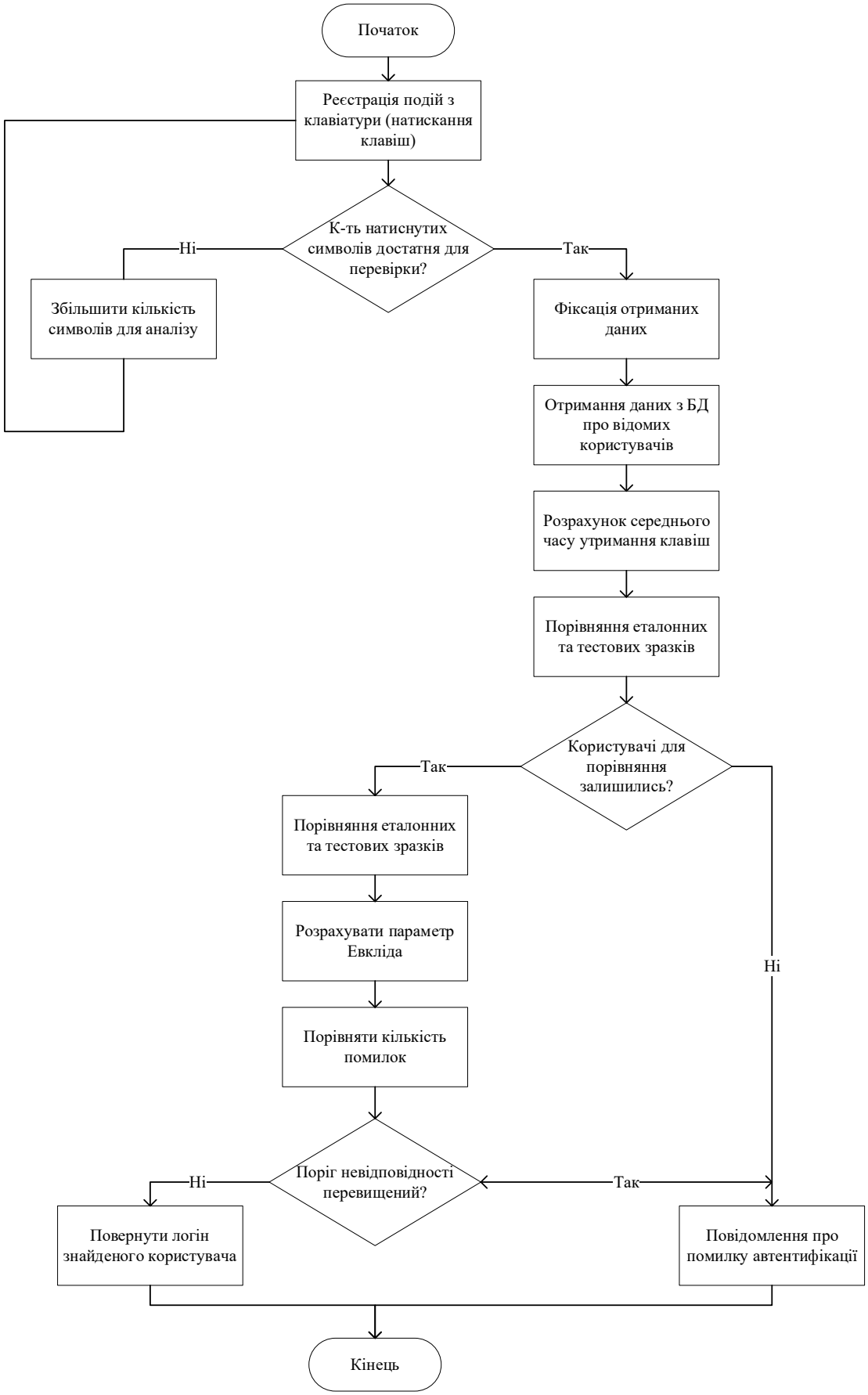


Рисунок 2.5 – Алгоритм розпізнавання користувача за клавіатурним почерком

Таким чином, обраний для роботи етап ідентифікації на основі біометричних даних у вигляді клавіатурного почерку є досить надійним підходом, що підсилений у роботі за рахунок дуального використання. Враховуючи те, що біометричні характеристики є унікальними для кожного користувача та дають можливість з мінімальними похибками здійснити ідентифікацію, дуальність такого рішення спрямована на покращення якості та результативності роботи системи захищеного доступу до ІС.

2.4 Оптимізація процесу звернення до БД засобами ГА

Генетичні алгоритми (ГА) – це обчислювальний підхід, що за своїми принципами близький до процесу природного відбору і генетики [44]. Вони широко використовуються в різних галузях, включно з оптимізацією звернень до баз даних. ГА пропонують унікальний і потужний спосіб розв'язання складних завдань оптимізації, імітуючи принципи еволюції.

У генетичному алгоритмі сукупність потенційних рішень розвивається протягом поколінь, щоб знайти найкраще можливе рішення. Процес полягає у реалізації таких ключових етапів (рис. 2.6).

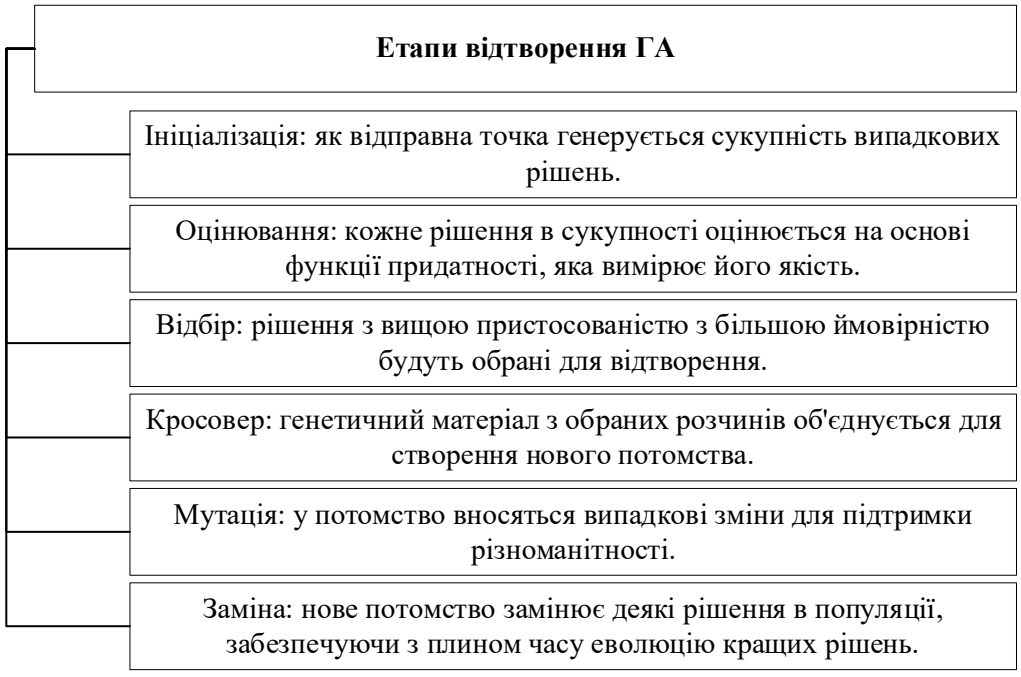


Рисунок 2.6 – Етапи відтворення типового генетичного алгоритму

Далі опишемо детальніше процес роботи генетичного алгоритму, що реалізовуватиметься в роботі для оптимізації звернень до бази даних. Реалізація генетичного алгоритму базується на операторі «схрещування», що рекомбінує рішення-кандидати, які представляються у вигляді масиву бітів. Зручність такого генетичного представлення полягає у легкому вирівнюванні всіх частин завдяки фіксованому розміру, що зумовлює полегшення виконання простих операцій кросовера.

Схематично алгоритму роботи генетичного алгоритму на базі оператора «схрещування» [44] представимо на рис. 2.7.

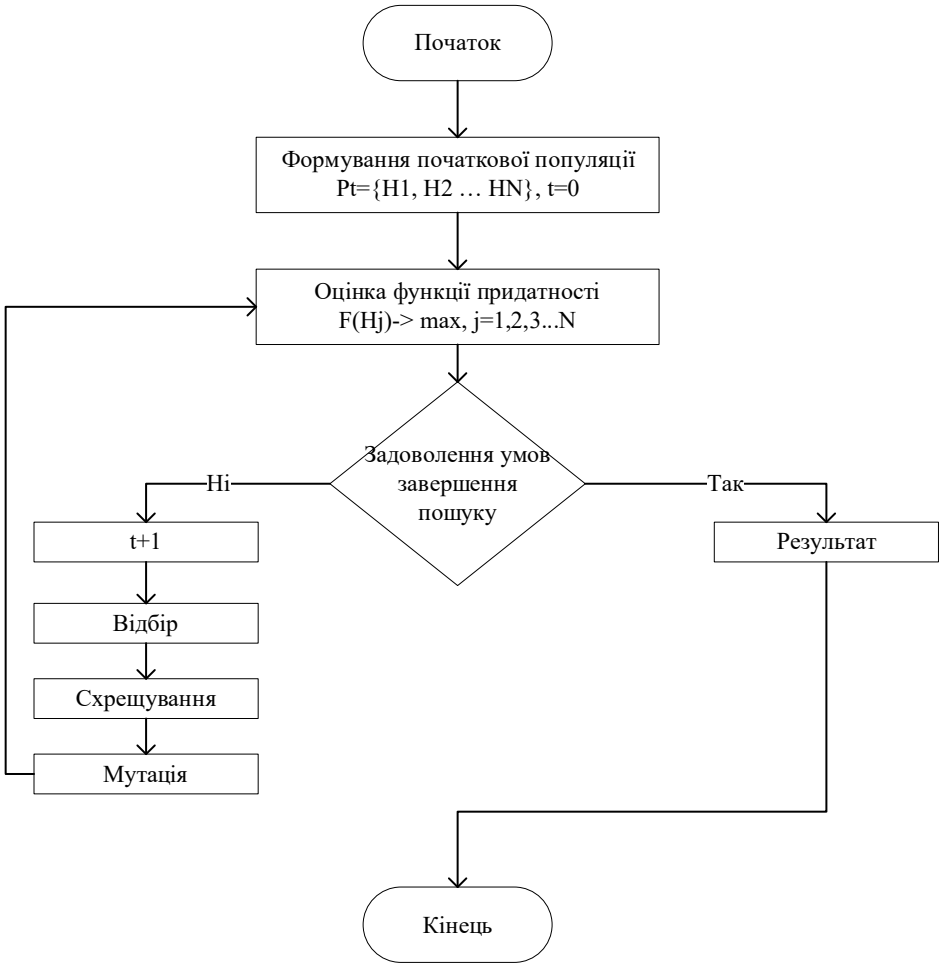


Рисунок 2.7 – Схема роботи генетичного алгоритму на основі оператора «схрещування» [44]

Для оптимізації звернень до бази даних в роботі застосовується генетичний алгоритм, що працює з популяцією хромосом, які можуть бути окремим вирішенням задачі.

Міра пристосовуваності $F(i)$ або фітнес-функція обчислюється для кожної хромосоми i у відповідній популяції. Для того, щоб стати батьками наступного покоління, в залежності від фітнес-функції відбувається відбір хромосом в популяції. Процес є циклічним. У використовуваній в роботі моделі чисельність населення N є фіксованою.

Кодування хромосом відбувається на основі таблиць, які беруть участь в запиті. Для реалізації кодування комівояжера створюється відповідний список відношень, які містяться у виразці. Список відношень L є впорядкований і формується наступним чином:

$$L = [1, 2, 3\ , 4, 5]$$

Вище згадана фітнес-функція вказує на степінь перевагами конкретного індивідуума (відповідно кращим вважається той, який показує менший час для виконання певного запиту).

Фітнес-функція може бути представлена у такому вигляді:

$$F = \frac{1}{\sum_{fj=1}^{n} w_j},$$

де $\mathrm{w}-$ час, що вказує на час реалізації запиту, що закодований у хромосому.

Етап відбору батьків відбувається за принципом інбридингу, коли вибір першого батька здійснюється випадковим чином, а вибір другого – має бути найбільш схожий на першого.

Етап селекції відбувається за принципом еліптизму, для якого характерна відносно швидка збіжність, він означає, що сортування особин в популяції відбувається в порядку зменшення їх фітнес-функції. Після такого сортування N особин буде зберігатись для наступного покоління. На етапі кросоверу використовується кросовер послідовного обміну два, оскільки саме від доцільний для хромосом, кодування яких відбувалось методом комівояжера. Згідно даного методу, відбувається обмін між батьками для подальшого формування нащадків. Послідовності генів для цього обираються випадковим чином. Схематично даний етап відобразимо на рис. 2.8.

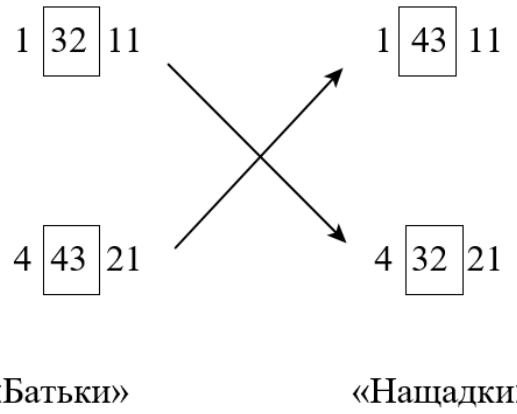


Рисунок 2.8 – Схема етапу кросоверу [44]

На етапі мутації взаємного обміну гени, що знаходяться на позиціях (i) і $[(i+1)\ mod\ N]$, де $N-$ довжина хромосоми, здійснюють обмін позиціями.

Критерієм зупинки алгоритму є рівень «плато», який означає, що протягом N поколінь визначений генотип популяції не зазнає змін достатнім чином.

Наведений генетичний алгоритм передбачається застосовувати для бази даних, що буде реалізована для збереження та оброблення даних користувачів. База даних для демонстраційної версії програми містить сутності користувача, ролі, лог-файлу, ключа доступу та клавіатурного почерку із відповідним набором властивостей. Діаграма [45] такої бази даних наведена на рис. 2.9.

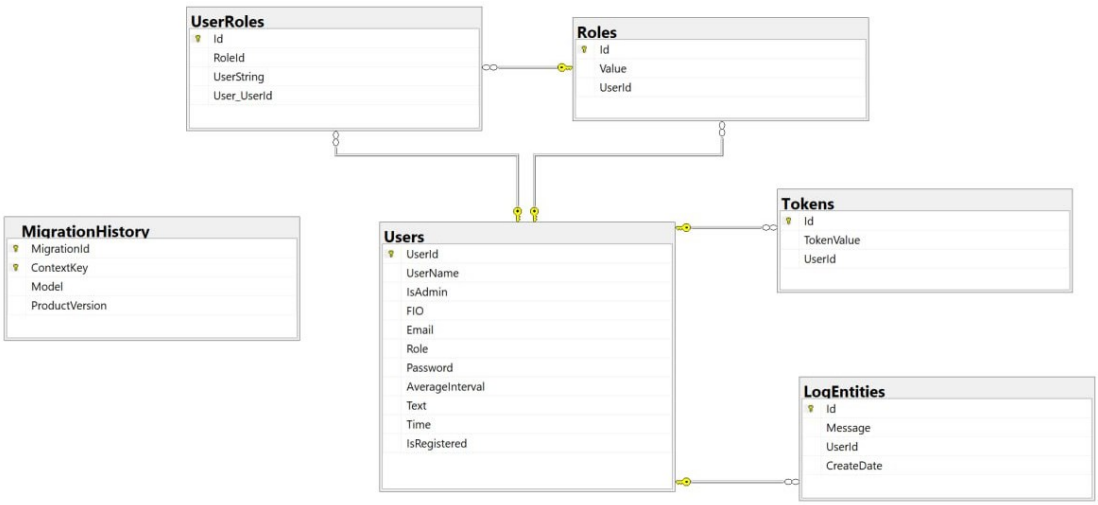


Рисунок 2.9 – Діаграма БД для програмного додатку

Варто зауважити, що для тестового варіанту програмного додатку, звернення до такої бази даних можуть опрацьовувати вбудовані засоби керування БД, наприклад, MS SQL. Проте, під час застосування даної розробки на практиці, із врахуванням значного обсягу та вмісту даних, впровадження генетичного алгоритму буде доцільним рішенням для оптимізації звернень та швидкої роботи захищуваної ІС.

2.5 Обґрунтування засобів програмування для автоматизації удосконаленого методу захисту

Виходячи із поставленої мети роботи та проаналізувавши особливості подальшої програмної реалізації додатку на основі удосконаленого методу захищеного доступу на основі двофакторної авторизації з використанням клавіатурного почерку, флеш-носія та оптимізації звернень до бази даних, було обрано відповідні програмні засоби (рис. 2.10).

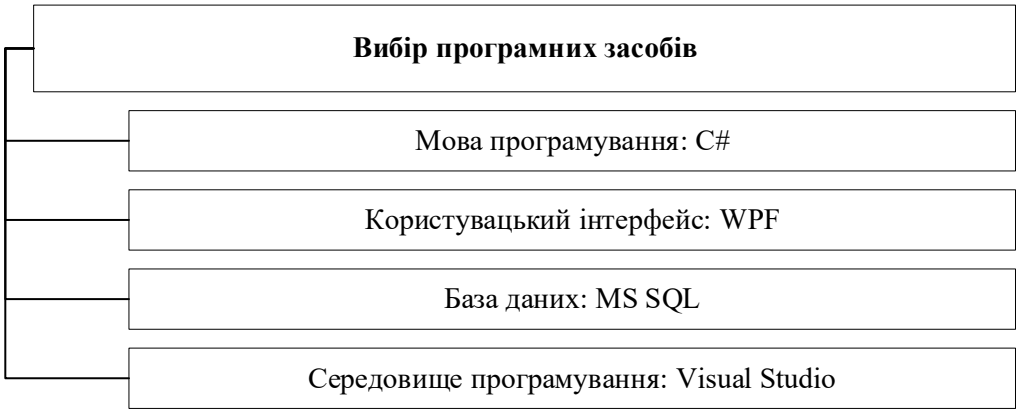


Рисунок 2.10 – Програмні засоби для практичної реалізації удосконаленого методу

Розглянемо деякі ключові особливості наведених програмних інструментів, які зумовили їх вибір для подальшої практичної розробки.

C# – це сучасна, загального призначення, об'єктно-орієнтована мова програмування, розроблена компанією Microsoft [46]. Вона широко використовується для розробки різноманітних програмних рішень.

Використання C# для реалізації програми захищеного доступу до інформаційної системи на основі флеш-носія та клавіатурного почерку має ряд переваг, зокрема, це компільована мова програмування, що робить її більш стійкою до атак, ніж інтерпретовані мови. C# має вбудовані функції безпеки, такі як перевірка типів та управління пам'яттю, які допомагають запобігти поширеним помилкам програмування, що можуть призвести до порушення безпеки. Виконуючи додатки даною мовою, можна використовувати додаткові бібліотеки та фреймворки безпеки, такі як .NET Framework Security, для посилення безпеки програми.

Наступна важлива особливість, що впливає на вибір – це продуктивність, адже C# компілюється в машинний код, що робить програми C# швидкими та ефективними, а також дана мова підтримує кешування та інші методи оптимізації, які покращують продуктивність програми.

Окрім того, C# підтримує різні типи даних, алгоритми та структури даних, що робить її гнучким інструментом для реалізації складних програм. C# можна використовувати для розробки програм для різних платформ, включаючи Windows, macOS, Linux. Це робить C# зручним вибором для розробки програми, яка має працювати на різних комп'ютерах.

Опираючись на особливості мети роботи, слід також зазначити, що існує ряд бібліотек та фреймворків C#, які можна використовувати для роботи з флеш-носіями та клавіатурним почерком. Це робить C# зручним вибором для реалізації програми, яка використовує ці методи аутентифікації..

Загалом, C# – це потужна, універсальна та зручна мова програмування, яка добре підходить для розробки широкого спектру програмних рішень.

Windows Presentation Foundation (WPF) – це фреймворк для побудови користувальницьких інтерфейсів у програмах .NET [47]. Він пропонує декларативний підхід до розробки інтерфейсу, ґрунтуючись на XAML (Extensible Markup Language), мові розмітки, схожій на HTML. WPF використовується для створення багатих, динамічних і візуально привабливих КІ для десктопних, веб- і мобільних програм.

Обраний фреймворк має зручний функціонал та широкі можливості для забезпечення максимально ефективної взаємодії користувача з додатком.

MS SQL – це реляційна система керування базами даних (СКБД), розроблена компанією Microsoft [48]. Вона широко використовується в корпоративних середовищах для зберігання та керування великими обсягами даних. Це потужна та універсальна СКБД, яка може допомогти покращити продуктивність, знизити витрати, приймати кращі рішення, підвищити конкурентну перевагу та забезпечити відповідність.

Що стосується технічних характеристик, то варто зазначити, що MS SQL має вбудовані функції безпеки, які допомагають захистити дані від несанкціонованого доступу; може масштабуватися від невеликих робочих груп до великих корпоративних середовищ з тисячами користувачів; сумісна з широким спектром інструментів та програмного забезпечення; має зручний для користувача графічний інтерфейс користувача (GUI) та потужну мову Transact-SQL (T-SQL) для керування даними.

Visual Studio (VS) – це інтегроване середовище розробки (IDE) від Microsoft, яке використовується для розробки широкого спектру програмного забезпечення, включаючи веб-додатки, десктопні програми, ігри, мобільні програми, хмарні програми та системне програмування [49].

Ключовими особливостями даного середовища є те, що VS підтримує широкий спектр мов програмування, включаючи C#, підтримує редактор коду, налагоджувач, профілювальник, інструменти керування версіями та багато іншого. Також VS можна розширювати за допомогою плагінів та інструментів, що робить його гнучким та адаптивним до потреб розробника. Середовище можна використовувати для розробки програм для різних платформ, включаючи Windows, macOS, Linux, Android та iOS, виходячи з потреб користувачів.

Отже, обрані та проаналізовані програмні продукти слугуватимуть програмними засоби для розробки додатку на основі удосконаленого методу, що уможливить реалізувати захищений доступ до ІС.

2.6 Висновки до розділу 2

Отже, в даному розділі описано удосконалення захисту ІС на основі захищеного флеш-носія, біометричних даних та оптимізацією звернень до БД. Під час виконання роботи обґрунтовано доцільність та практичність використання двофакторної авторизації для контрольованого доступу користувачів.

Обраний для роботи підхід ідентифікації на основі біометричних даних у вигляді клавіатурного почерку є досить надійним, оскільки підсилений за рахунок дуального використання та дає можливість із мінімальними похибками здійснити ідентифікацію та передбачити можливість зміни клавіатурного почерку користувача залежно від часу доби та використовуваної клавіатури.

Використання флеш-носія дає можливість не лише забезпечити один із факторів авторизації користувача, але і організувати постійний контроль під час роботи в системі. Шифрування алгоритмом AES даних на носії забезпечує захищеність унікальних ідентифікаційних даних.

Застосування генетичного алгоритму дозволить оптимізувати звернення до бази даних, там самими забезпечити швидку та ефективне оброблення даних.

Обрані засоби програмування, такі як мова C#, фреймворк WPF, СУБД MS SQL та середовище Visual Studio, дозволяють у повній мірі реалізувати поставлені практичні задачі.

3 ПРАКТИЧНА РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ КОМПЛЕКСНОГО МЕТОДУ ЗАХИЩЕНОГО ДОСТУПУ ДО ІС

У даному розділі описано практичну реалізацію удосконаленого комплексного методу для захищеного доступу до інформаційної системи на основі флеш-носія, біометричної автентифікації та із використанням генетичного алгоритму для оптимізації звернень до бази даних.

Розроблюваний програмний додаток на основі удосконаленого методу реалізований на мові програмування C# із використанням графічної підсистеми WPF у середовищі програмування Visual Studio. База даних для десктопного програмного додатку розроблена засобами MS SQL.

3.1 Алгоритм роботи програмного застосунку, що автоматизує розроблений підхід для захищеного доступу до ІС

Виходячи із поставленої мети роботи та на основі удосконаленого методу, сформулюємо алгоритм роботи програмного засобу, що автоматизує розроблений автором магістерської роботи підхід.

Крок 1. Запуск виконуваного файлу захищуваної інформаційної системи.

Крок 2. Автоматичне відкриття програми авторизації користувача для входу в систему.

Крок 3. Вибір користувачем форми роботи з додатком (авторизація / реєстрація).

Крок 4. Для авторизації в системі користувачеві у відповідному полі необхідно ввести логін.

Крок 5. Наступним етапом є підключення флеш-носія у USB-порт пристрою.

Крок 6. Користувач, переконавшись у коректному введенні логіну та підключенні флеш-носія, повинен натиснути кнопку «Перевірити дані».

Крок 6.1 У випадку, якщо логін користувача відомий системі – перевіряється підключений флеш-носій та пароль до нього. Перехід до кроку 7.

Крок 6.2 У випадку, якщо логін користувача невідомий системі, користувач отримує повідомлення про відмову доступу до системи.

Крок 7. Пароль користувач вводить у два етапи.

Крок 7.1. Введення паролю, що на постійній основі встановлений для даного носія.

Крок 7.2. Вибір випадковим чином сформованого питання із системи та введення користувачем відповіді на нього.

Крок 8. Перевірка змісту паролей та клавіатурного почерку користувача за введеними фразами.

Крок 8.1 У випадку, якщо звернення до бази даних показало, що вказаний користувач відомий системі, підключений флеш-носій та паролі відповідають вимогам, відбувається перехід до кроку 9.

Крок 8.2 У випадку, якщо звернення до бази даних показало, що вказаний користувач представив хибні дані авторизації – у доступі до системи відмовлено.

Крок 9. Виведення повідомлення користувачеві про успішний процес авторизації в ІС.

Крок 10. Реєстрація факту входу користувача в систему.

Крок 11. Перевірка підключеного флеш-носія до пристрою протягом усього часу роботи користувача в системі.

Крок 11.1 У випадку, якщо флеш-пристрій постійно підключений та перевірки успішні – користувач має змогу не відволікаючись на додаткові дії працювати у системі.

Крок 11.2 У випадку, якщо в один із моментів перевірки флеш-носій відсутній або його параметри не відповідають вимогам, користувач отримує повідомлення про негайне припинення роботи в програмі, доступ автоматично блокується. Перевірка флеш-носія відбувається через довільні проміжки часу, але не більші, ніж 60с.

Описаний алгоритм розроблюваного програмного додатку на основі удосконаленого методу представимо у вигляді блок-схеми на рис. 3.1.

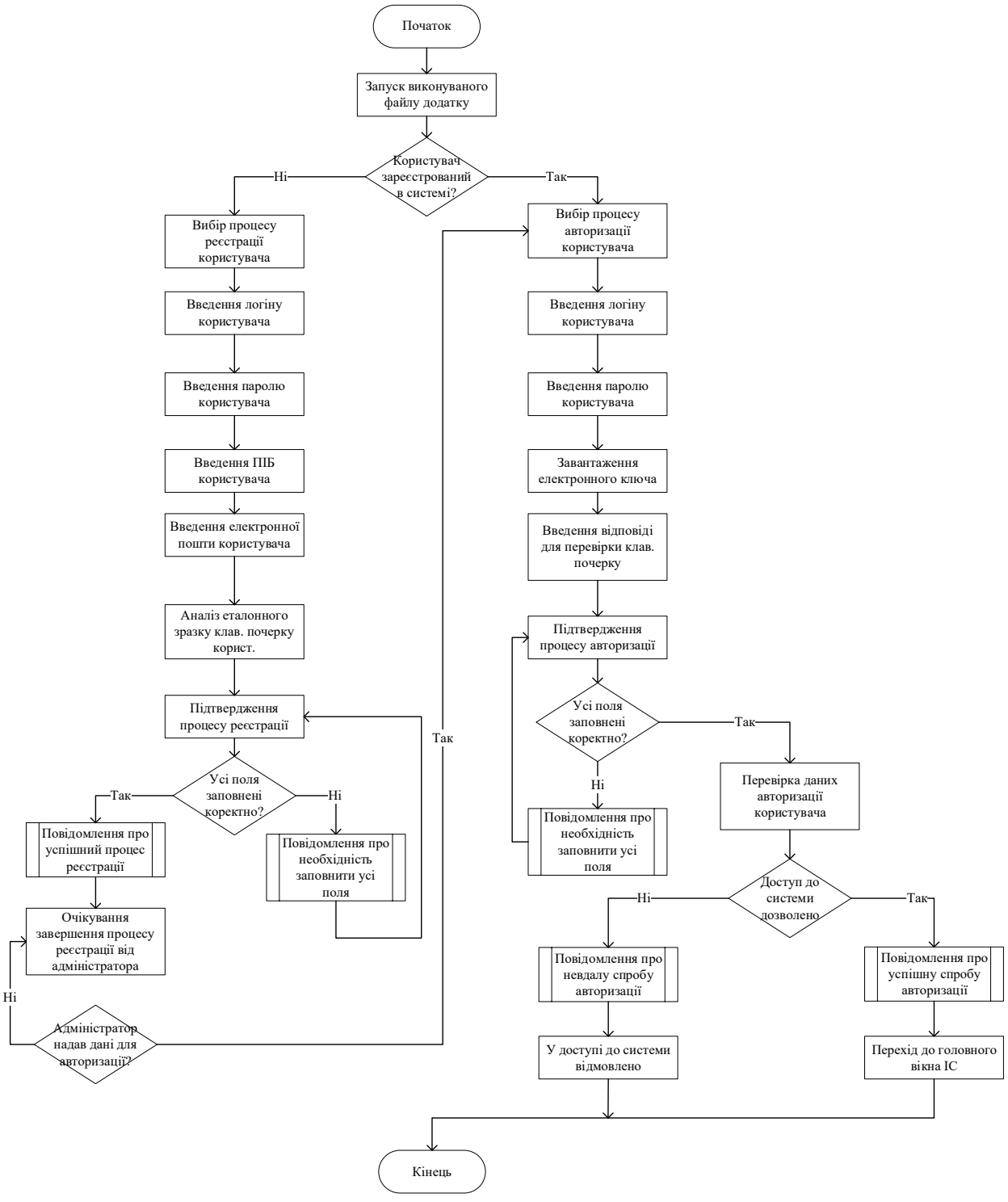


Рисунок 3.1 – Алгоритм роботи програмного додатку на основі удосконаленого методу для захищеного доступу до ІС

База даних користувачів формується при їх первинній реєстрації. Для цього можливому користувачеві інформаційною системою необхідно заповнити форму реєстрації.

У формі реєстрації користувач повинен вказати своє прізвище та ім’я, займану посаду, вказати електронну пошту, яка слугуватиме логіном для входу в систему, вказати пароль, який використовуватиметься для перевірки та підтвердження підключеного флеш-носія.

Другою частиною форми реєстрації є заповнення бази питань, які мають загальний характер, не маючи змістовності для анкети користувача, а використовуються виключно для зчитування біометричних даних у вигляді клавіатурного почерку користувача.

Після завершення реєстрації користувачем у системі, перевірка даних відбувається адміністратором системи, який призначає роль користувача та видає флеш-носій, який використовуватиметься користувачем для входу в систему.

Робота в системі організовується на основі рольової моделі розмежування доступу. Такий підхід надає можливість ще з моменту входу користувача в систему контролювати його дії, обмежуючи доступ до файлів, що не передбачені для його ролі.

Для демонстраційного варіанту програмного додатку, реалізовано три ролі в системі:

– адміністратор – користувач системи, що має доступ до бази даних та здійснює перевірку та підтвердження реєстрації користувача;

– користувач з високим рівнем доступу (робітник 1) – має доступ до файлів та функціоналу системи на рівні керуючої посади;

– користувач з низьким рівнем доступу (робітник 2) – має доступ до файлів загального доступу з можливістю лише їх перегляду.

Оскільки, процес реєстрації та авторизації користувача ґрунтується на тривалій обробці інформації із бази даних, для оптимізації звернень до БД та швидкої роботи програмного модуля захищеного доступу, використаємо генетичний алгоритм.

Проаналізувавши розроблений алгоритм роботи програмного додатку, спроектуємо орієнтовний вигляд вікон програми, щоб забезпечити в подальшому зручний та зрозумілий інтерфейс для користувача.

Передбачається, що після запуску виконуваного файлу додатку, відкривається вікно програми, що пропонує користувачеві здійснити один із процесів: авторизацію або реєстрацію. Спроєктований вигляд даного вікна представлено на рис. 3.2.

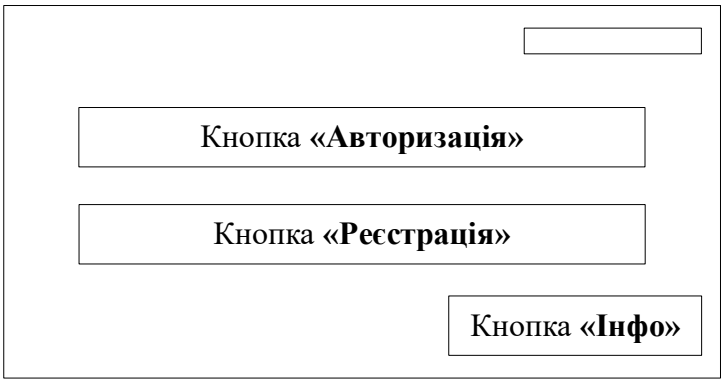


Рисунок 3.2 – Проектування вікна додатку для вибору опції роботи з додатком

Наступне вікно, яке доцільно спроєктувати перед початком роботи – це вікно авторизації користувача в системі, щоб прописати усі значущі функціональні кнопки та поля.

Передбачається, що дане вікно повинне містити:

– поле для введення логіну та паролю;

– кнопку завантаження даних про флеш-носій для його перевірки;

– кнопку старту перевірки клавіатурного почерку;

– поле для введення тексту для аналізу;

– кнопку завершення аналізу;

– кнопку підтвердження процесу авторизації.

Наведемо орієнтований проект даного вікна на рис. 3.3.

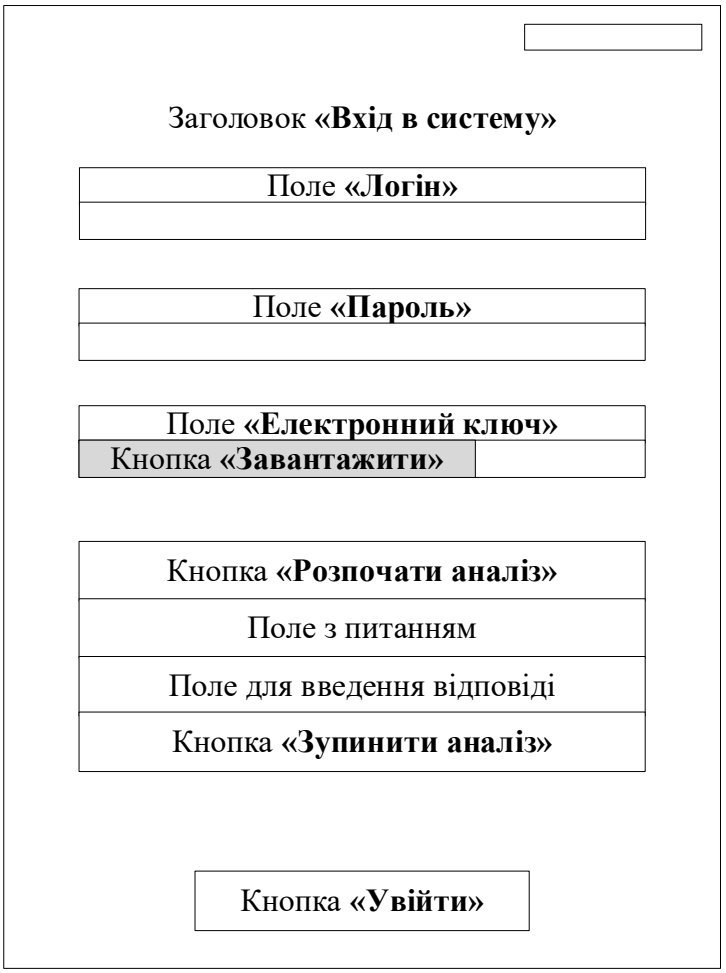


Рисунок 3.3 – Проєктування вікна додатку для авторизації користувача

Також в роботі реалізовано вікна для реєстрації користувача, вікна з функціоналом для адміністратора системи, що дозволяють генерувати ключі, призначати ролі.

Варто зазначити, що програмний додаток орієнтований на забезпечення захищеного доступу до інформаційної системи довільної організації, а тому потенційні користувачі не є фахівцями з кібербезпеки. З огляду на це, інтерфейс додатку повинен бути простим та зрозумілим, щоб не викликати труднощів при його використанні.

Особливості роботи та реалізації наведених засобів авторизації, що слугують складовими частинами вдосконаленого методу захищеного доступу до ІС, було описано у попередньому розділі.

3.2 Програмна реалізація додатку, що автоматизує запропонований комплексний метод

Із використанням програмних засобів, вибір яких обґрунтовано у підрозділі 2.5, було реалізовано практично програмний додаток на основі запропонованого комплексного методу. Розглянемо основні фрагменти коду, що були написані.

Для роботи додатку були підключені необхідні інструкції (using), зокрема, для імпорту простору імен структурних програмних частин, таких як класи та функціональні можливості для роботи з файлами та директоріями, методи і т.д.:

```
using Newtonsoft.Json;
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Configuration;
using System.Diagnostics;
using System.IO;
using System.Linq;
using System.Text;
using System.Timers;
using System.Windows;
using usbkeyword.Context;
using usbkeyword.Models;
using Path = System.IO.Path;
```

Для перевірки флеш-носія реалізовано метод StartFileCheck, що звертається до USB-порти та «шукає» необхідний ключ:

```
private void StartFileCheck(string filePath)
    {
      _cancellationTokenSource = new CancellationTokenSource();
      Task.Run(() => CheckFileExistence(filePath, _cancellationTokenSource.Token));
    }
```

Як згадувалося раніше, алгоритмом передбачено перевірку наявності флеш-носія у портові ПК під час роботи користувача у системі. Перевірка відбувається через певні проміжки часу без будь-якого попередження чи повідомлення після, тому користувачеві ці перевірки невідомі. Проте, якщо система захисту під час такої перевірки не виявить відповідний носій, робота

користувача блокується. Реалізовані дані перевірки за допомогою прописаного методу CheckFileExistence:

```
private async Task CheckFileExistence(string filePath, CancellationToken token)
    {
      while (!token.IsCancellationRequested)
      {
        if (!File.Exists(filePath))
        {
          Application.Current.Dispatcher.Invoke(() =>
          {
            Application.Current.Shutdown();
          });
          break;
        }
        await Task.Delay(5000, token);
      }
    }
```

Інформація, що використовується для перевірки флеш-носія та міститься на ньому у зашифрованому вигляді захищається з використанням криптостійкого алгоритму AES. Фрагмент реалізації наведемо далі:

```
private static byte[] AES_Decrypt(byte[] bytesToBeDecrypted, byte[] passwordBytes)
    {
      byte[] decryptedBytes = null;
      byte[] saltBytes = new byte[] { 2, 1, 7, 3, 6, 4, 8, 5 };
      using (MemoryStream ms = new MemoryStream())
      {
        using (RijndaelManaged AES = new RijndaelManaged())
        {
          AES.KeySize = 256;
          AES.BlockSize = 128;
          var key = new Rfc2898DeriveBytes(passwordBytes, saltBytes, 1000);
          AES.Key = key.GetBytes(AES.KeySize / 8);
          AES.IV = key.GetBytes(AES.BlockSize / 8);
          AES.Mode = CipherMode.CBC;
          using (var cs = new CryptoStream(ms, AES.CreateDecryptor(), 
CryptoStreamMode.Write))
          {
            cs.Write(bytesToBeDecrypted, 0, bytesToBeDecrypted.Length);
            cs.Close();
          }
          decryptedBytes = ms.ToArray();
        }
      }
      return decryptedBytes;
    }
```

Другим фактором авторизації користувача є клавіатурний почерк, який перевіряється у два етапи.

На першому етапі перевіряється клавіатурний почерк користувача, що зчитується під час введення паролю:

```
private bool AnalyzeButton_Click(User user)
    {
      double averageInterval = 0;
      if (intervals.Count > 0)
      {
        averageInterval = CalculateAverage(intervals);
      }
      if (user.AverageInterval - 300 < averageInterval && averageInterval < user.AverageInterval + 300)
        return true;

      return false;
    }
```

Другий етап перевірки користувача відбувається під час введення відповіді на випадкове питання:

```
private double CalculateAverage(List<double> values)
    {
      double sum = 0;
      foreach (var value in values)
      {
        sum += value;
      }
      return sum / values.Count;
    }
```

У формі авторизації користувача реалізована валідація полів. У випадку, якщо користувач натисне кнопку входу, проте певні поля не будуть заповненні, він отримає відповідне повідомлення (наприклад, «Введіть логін», «Введіть ключ», «Введіть пароль»):

```
private bool VerifyData()
    {
      if (String.IsNullOrEmpty(UserName.Text))
      {
        MessageBox.Show("Введіть логін");
        return false;
      }
      if (String.IsNullOrEmpty(FileName.Text))
      {
```

```
            MessageBox.Show("Введіть ключ");
            return false;
        }
        if (String.IsNullOrEmpty(Password.Text))
        {
            MessageBox.Show("Введіть пароль");
            return false;
        }
        return true;
    }
```

У випадку, якщо користувач, що не має облікового запису, йому слід спочатку зареєструватись. Натиснення кнопки «Реєстрація» передбачає відкриття відповідної форми:

```
private void newAccount_Click(object sender, RoutedEventArgs e)
    {
        active = false;
        var w = Application.Current.Windows[0];
        this.Hide();
        Registration wfAbout = new Registration();
        wfAbout.ShowDialog();

        this.Close();
    }
```

Для перевірки користувачів та звернень до системи для отримання інформації про них, прописано метод GetUser:

```
    private List<User> GetUsers()
    {
        var useFile = Convert.ToBoolean(ConfigurationManager.AppSettings["UsFile"]);
        var users = new List<User>();
        if (useFile == true)
        {                                   var                                 path                                =
$"{Path.GetDirectoryName(AppDomain.CurrentDomain.BaseDirectory)}/ResourceFile.json";
            var exists = File.Exists(path);
            if (!exists)
                File.Create(path);
            using (var r = new StreamReader(path))
            {
                var json = r.ReadToEnd();
                users = JsonConvert.DeserializeObject<List<User>>(json);
            }
        }         else
        {             users = _context.Users.ToList();
        }         return users;
    }
```

Наступним кроком продемонструємо фрагмент коду, що призначений для реалізації функціоналу зчитування клавіатурного почерку користувача у випадку невідомої заздалегідь текстової послідовності. В даному випадку користувачеві слід натиснути кнопку, яка запускає початок перевірки, далі ввести безпосередньо саму відповідь на надане питання, потім натиснути кнопку для зупинки аналізу:

```
private void Button_Click_2(object sender, RoutedEventArgs e)
    {
      if (!drawEnable)
      {
        text_keyword.Text = "";
        drawEnable = true;
        StaerRecog.Content = "Аналіз розпочато, зупинити?";
        stopwatch.Start();
        text_keyword.IsEnabled = true;
        lable_text.Text = "Розробка програмного забезпечення вимагає ретельного планування та тестування на кожному етапі";
      }
      else
      {
        stopwatch.Stop();
        drawEnable = false;
        text_keyword.IsEnabled = true;
        Time = stopwatch.ElapsedMilliseconds;
        Text = text_keyword.Text;
        StaerRecog.Content = "Розпочати знову?";
        MessageBox.Show("Аналіз завершено та дані збережено!");
      }
    }
```

Фрагмент коду для метод text_keyword_TextChanged, що перевіряє коректність введених даних користувачем при другому етапі перевірки клавіатурного почерку:

```
private void text_keyword_TextChanged(object sender, System.Windows.Controls.TextChangedEventArgs e)
    {
      if (!stopwatch.IsRunning)
      {
        stopwatch.Start();
        lastTime = stopwatch.ElapsedMilliseconds;
      }
      else
      {
```

```
            double currentTime = stopwatch.ElapsedMilliseconds;
            double interval = currentTime - lastTime;
            if (interval > 0)
            {
               intervals.Add(interval);
            }
            lastTime = currentTime;
         }
      }
```

Реалізовано також діалогові вікна в додатку, для того, щоб взаємодіяти з користувачем, наприклад, повідомляти про успішну / неуспішну спробу входу, про те, що користувача не знайдено і необхідно зареєструватись і т.д. Якщо один із параметрів невірний, користувач отримує також про це відповідне повідомлення.

Далі наведемо відповідні фрагменти коду:

– якщо користувач не існує:

```
private void Button_Click_1(object sender, RoutedEventArgs e)
      {
         if (!VerifyData())
            return;
         var res = GetUsers();
         var eror = "";
         if (res == null)
         {
            eror = "\"Користувача не знайдено\"";
         }
         else
         {
            var f = lblFaceName.Content.ToString().ToLower();
            var user = res.FirstOrDefault(x => x.UserName == UserName.Text);
            if (user == null)
            {
               eror = "\"Користувача не знайдено\"";
            }
            else
            {
               if (user.IsAdmin || (user.UserId == Decript(FileName.Text) && user.Password ==
Password.Text) && AnalyzeButton_Click(user))
               {
                  this.Hide();
                  Helper.AddLog("User is logged in", user.UserName);
                  CommonPage commonpage = new CommonPage();
                  commonpage.AuthUserName.Content = user.UserName;
                  if (user.IsAdmin || user.Role == "Адміністратор")
```

```
                {
                    commonpage.UserTable.Visibility = Visibility.Visible;
                }
                else
                {
                    commonpage.UserTable.Visibility = Visibility.Hidden;
                }
                StartFileCheck(FileName.Text);

                commonpage.Registration.Visibility = commonpage.Login.Visibility =
Visibility.Hidden;
                commonpage.Exit.Visibility = Visibility.Visible;
                commonpage.UserInfo.Visibility = Visibility.Visible;
                commonpage.history.Visibility = Visibility.Visible;
                enableAutoNavigate = false;
                MessageBox.Show("Спроба входу була вдалою!");
                commonpage.ShowDialog();
                this.Close();
            }
```

– якщо частина даних введена некоректно:

```
else
            {
                if (user.UserId != Decript(FileName.Text))
                {
                    eror = $"Невірний ключ";
                }
                else if (user.Password != Password.Text)
                {
                    eror = $"Невірний пароль";
                }
                else
                {
                    eror = $"Помилка входу";
                }
            }
        }
```

– якщо спроба входу невдала:

```
        if (!string.IsNullOrEmpty(eror))
        {
            --failCount;
            MessageBox.Show($"Невдала спроба ідентифікації. {eror}. Залишилося спроб -
{failCount}");
            if (failCount == 0)
            {
```

```
            this.Hide();
            Environment.Exit(1);
          }
          else
          {
            return;
          }
        }
      }
```

Для здійснення усіх облікових дій в системі відповідно відбувається взаємодія з реалізованою базою даних, що складається з попередньо спроєктованих сутностей та їх властивостей. Фрагмент взаємодії з базою даних наведемо далі:

```
public class RecognizeContext : DbContext
  {
    public RecognizeContext() : base("DefaultConnection")
    {
    }
    public DbSet<User> Users { get; set; }
    public DbSet<Token> Tokens { get; set; }
    public DbSet<LogEntity> LogEntities { get; set; }
    public DbSet<Role> Roles { get; set; }
    public DbSet<UserRole> UserRoles { get; set; }
    protected override void OnModelCreating(DbModelBuilder modelBuilder)
    {

    }
  }
```

Для оптимізації звернень до бази даних та ефективної роботи з нею, реалізовано генетичний алгоритм. Фрагмент даної реалізації наведемо далі:

```
public void Run()
  {
    for (int generation = 0; generation < MaxGenerations; generation++)
    {
      EvaluateFitness();
      List<Chromosome> parents = new List<Chromosome>();
      for (int i = 0; i < PopulationSize * 2; i++)
      {
        parents.Add(SelectParents());
      }
      List<Chromosome> offspring = new List<Chromosome>();
      for (int i = 0; i < PopulationSize; i++)
      {
        Chromosome parent1 = parents[i * 2];
```

```
                Chromosome parent2 = parents[i * 2 + 1];
                Chromosome child = Crossover(parent1, parent2);
                MutateChromosome(child);
                offspring.Add(child);
            }
            Population = offspring;
        }
        Chromosome bestChromosome = Population.OrderByDescending(c => c.Fitness).First();
        Console.WriteLine("Best chromosome: " + bestChromosome.Genes.ToTrimmedString());
    }
}
GeneticAlgorithm ga = new GeneticAlgorithm(100, 32, 0.
```

Отже, наведені фрагменти коду є основними функціональними частинами програмного проекту, який є практичною реалізацією удосконаленого методу, який має на меті забезпечити захищений контрольований доступ дод інформаційної системи за рахунок двофакторної авторизації та оптимізувати процес звернень до бази даних за рахунок використання генетичного алгоритму.

3.3 Інструкція користувача для роботи з програмним додатком

Опишемо інструкцію користувача для роботи із реалізованим програмним додатком на основі комплексного методу. Виходячи з мети роботи було удосконалено метод захищеного доступу до інформаційної системи, що передбачає двофакторну авторизацію користувача на основі клавіатурного почерку та використання флеш-носія.

Програмний додаток працює на операційній системі Windows 10 та 11, не потребує встановлення додаткових утиліт, окрім налаштування бази даних при встановленні, власне, всієї системи.

Для початку роботи користувачеві необхідно запустити виконуваний файл програми, після чого відкривається діалогове вікно (рис. 3.4).

Якщо користувач вже має обліковий запис в системі, йому необхідно скористатися кнопкою «Авторизація» та продовжити свою роботу. Якщо ж користувач використовує додаток вперше, обов’язковим є процес реєстрації.

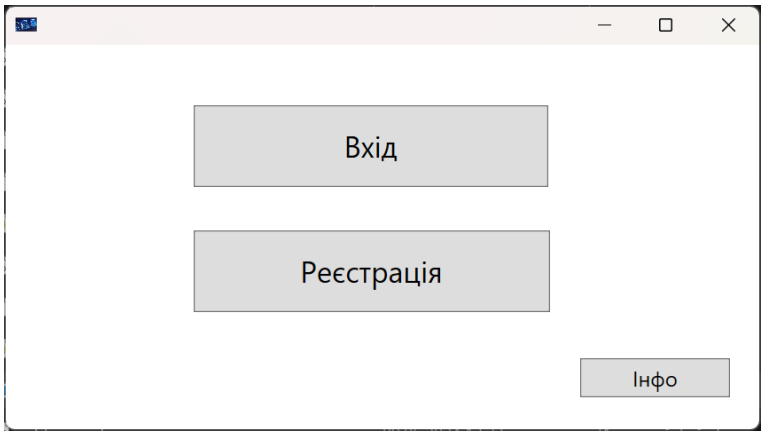


Рисунок 3.4 – Вигляд початкового вікна програмного додатку

Розглянемо процес реєстрації.

Вигляд відповідного вікна наведено на рис. 3.5.

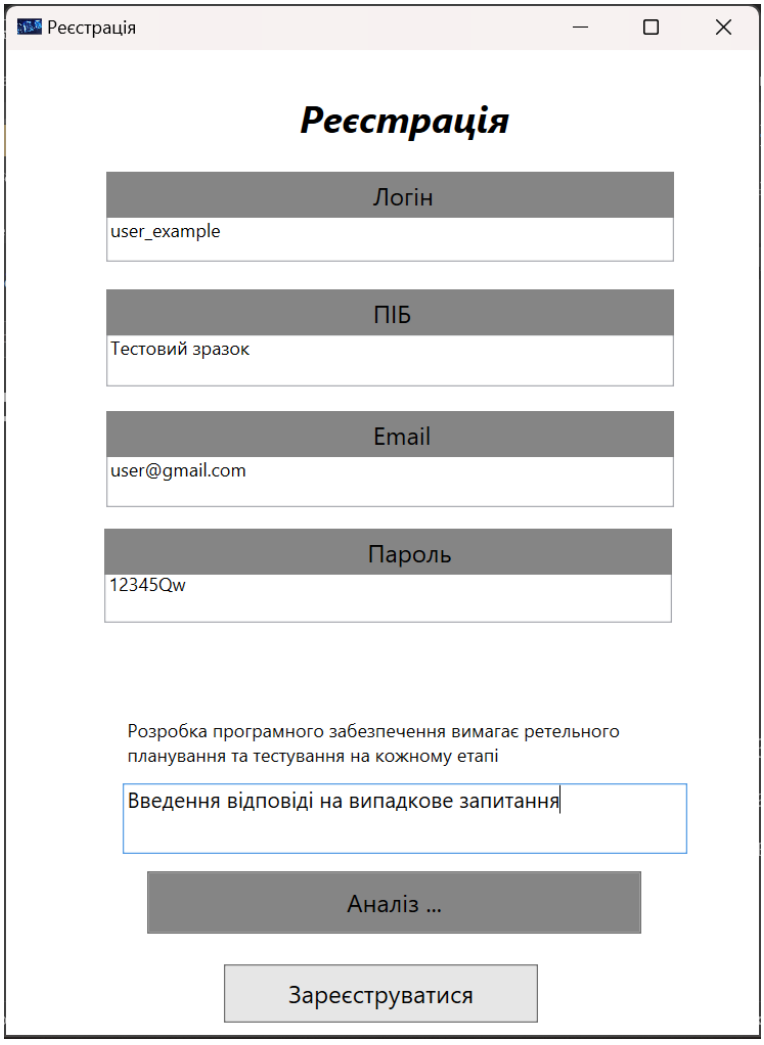


Рисунок 3.5 – Вигляд вікна форми «Реєстрація»

У даному вікні користувачеві необхідно вказати свій логін, ПІБ, електронну адресу, пароль. Далі необхідно натиснути кнопку «Розпочати аналіз клавіатурного почерку». Після натиснення даної кнопки зявиться питання. Дані питання не мають змістовного навантаження та є випадковими. Їх призначення – це перевірити клавіатурний почерк користувача при введенні заздалегідь невідомої комбінації. Після того, як користувач завершить введення відповіді, йому слід припинити функцію виконання аналізу.

Якщо користувач виконав усі дії вірно і аналіз був проведено успішно, буде виведено відповідне повідомлення (рис. 3.6).

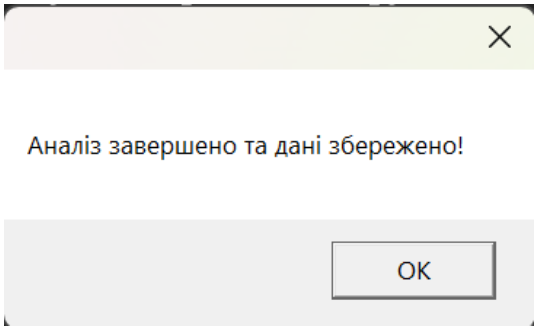


Рисунок 3.6 – Вигляд вікна з повідомленням про успішний аналіз почерку

Якщо аналіз не був здійсненний або виникли помилки, користувачеві слід скористатись кнопкою «Розпочати аналіз знову».

Далі користувач завершує свій процес реєстрації, підтверджуючи свої дії натисненням кнопки «Зареєструватись». Про успішну реєстрацію буде повідомлено відповідним сповіщенням (рис. 3.7).

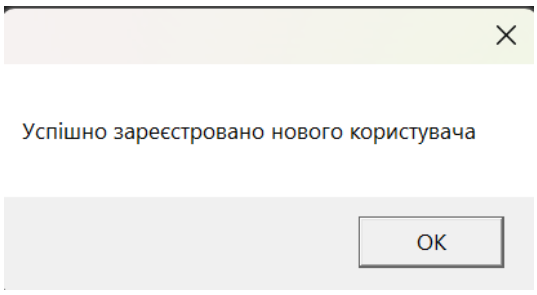


Рисунок 3.7 – Вигляд вікна з повідомленням про успішну реєстрацію

Після виконання даних функцій користувачеві слід очікувати відповідь від адміністратора системи, який підтвердить його реєстрацію, визначить

рівень доступу та надасть зашифрований флеш-носій.

Розглянемо дії адміністратора. У своєму обліковому записі адміністратор може переглядати та керувати обліковими записами зареєстрованих користувачів. Для цього адміністратор натискає кнопку «Адміністрування користувачів» на головному вікні системи (рис. 3.8).

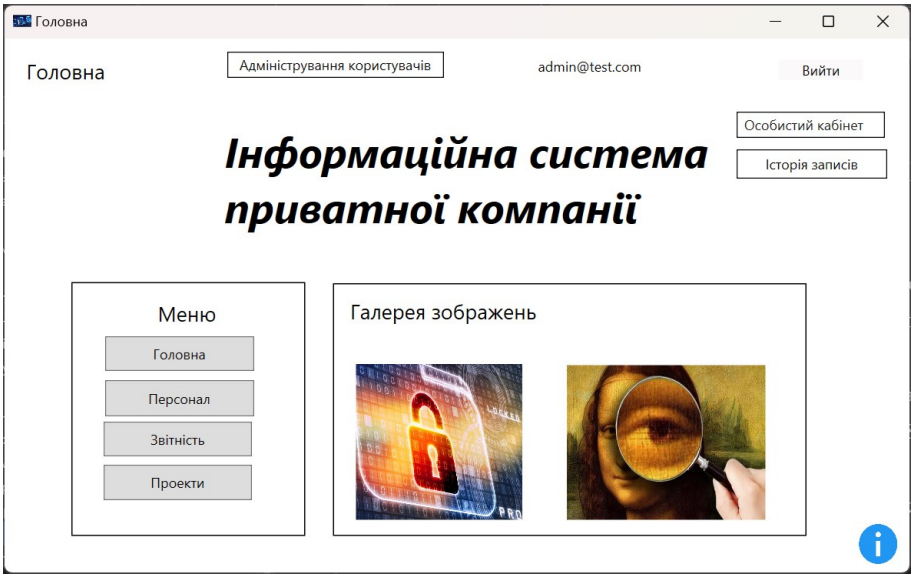


Рисунок 3.8 – Вигляд головного вікна системи для облікового запису адміністратора

На сторінці адміністрування користувачів (рис. 3.9) адміністратору доступна таблиця користувачів із функціональними кнопками «Перегляд» та «Видалити».

Кнопка «Перегляд» дозволяє адміністратору:

– перевірити вказані дані користувача;

– надати роль (для демонстрації запропоновано тестові ролі);

– згенерувати файл для подальшого створення захищеного флеш-носія, що слугуватиме одним із факторів авторизації.

Вигляд вікна редагування даних користувача наведено на рис. 3.10.

Кнопка «Видалити» відповідно видаляє користувача із системи, натиснувши її, адміністратор отримує відповідне сповіщення (рис. 3.11).

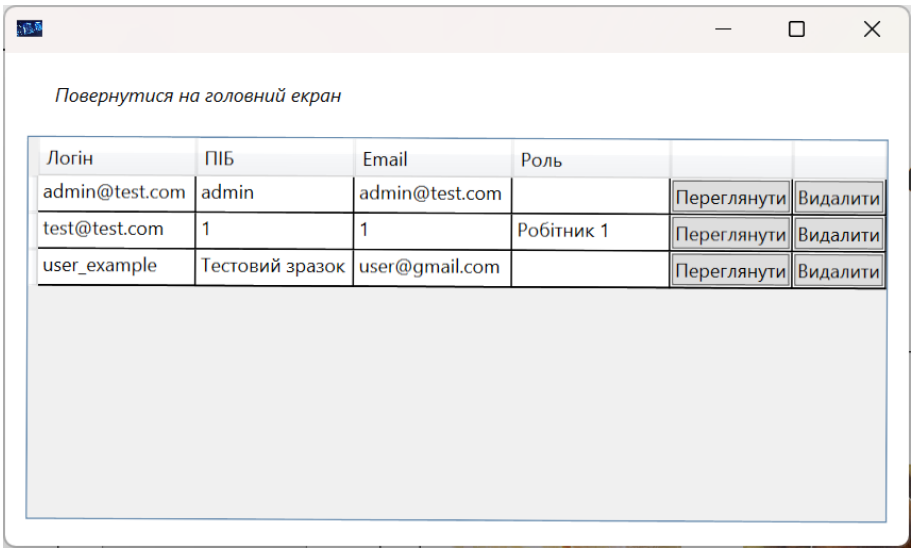


Рисунок 3.9 – Вигляд вікна адміністрування користувачів

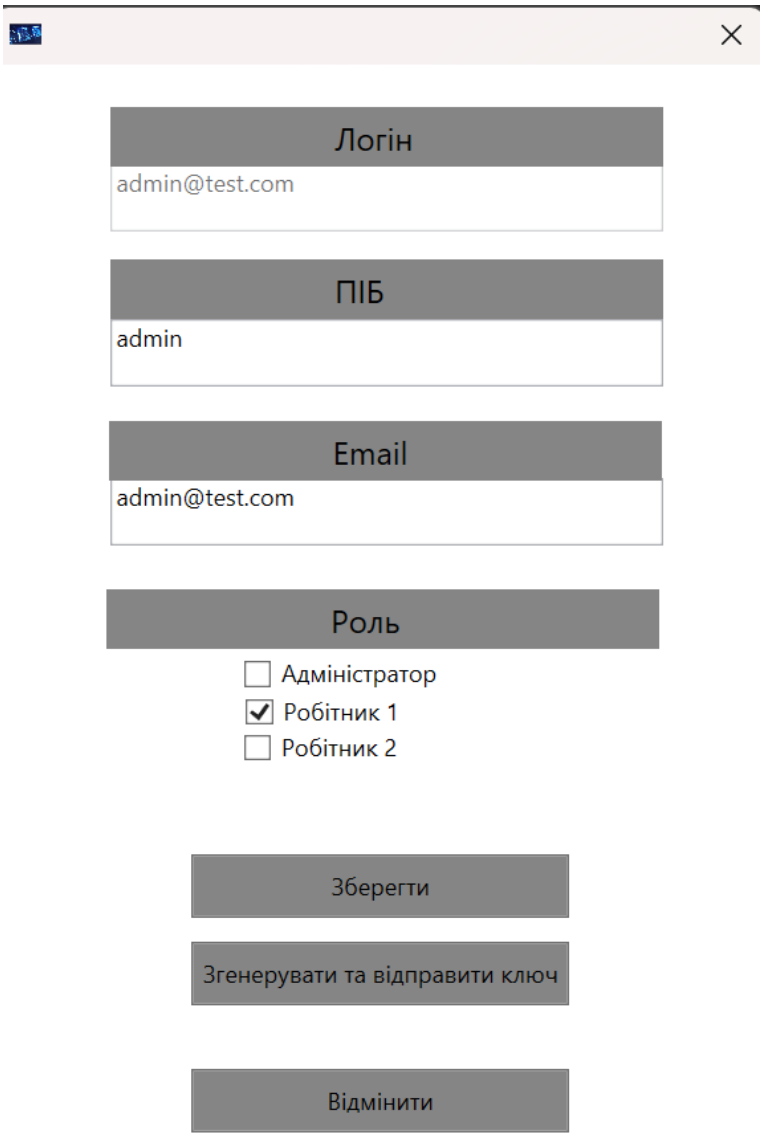


Рисунок 3.10 – Вигляд вікна редагування даних користувача

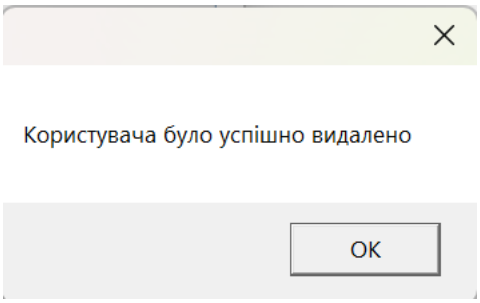


Рисунок 3.11 – Вигляд вікна із сповіщенням про успішне видалення користувача

Для створення ключа, адміністратору слід натиснути кнопку «Згенерувати та відправити ключ». Після успішної генерації ключа буде відповідне сповіщення від системи (рис. 3.12).

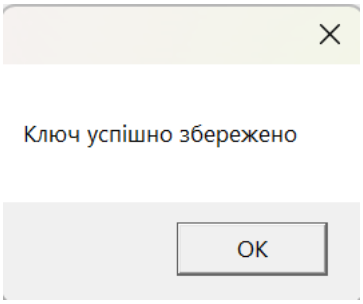


Рисунок 3.12 – Вигляд вікна із сповіщенням про успішну генерацію ключа

Після виконання усіх необхідних дій по реєстрації користувача, адмінстратору слід зберегти змін натисненням кнопки «Зберегти». Після цього у вікні «Адміністрування користувачів» у відповідних колонках будуть відображатись зміни, наприклад, про зміну ролі користувача.

На основі згенерованого файлу із зашифрований ключем, адмінстратор записує захищений флеш-носій та надає його відповідному користувачеві, після чого той може здійснити процес авторизації.

Для цього, обравши у першому вікні (рис. 3.4) функціональну кнопку вхід, користувач здійснює процес авторизації. Заповнює поля «Логін» та «Пароль», обирає свій електронний ключ, здійснює введення відповіді на випадкове питання для аналізу клавіатурного почерку та підтверджує авторизацію в системі (рис. 3.13).

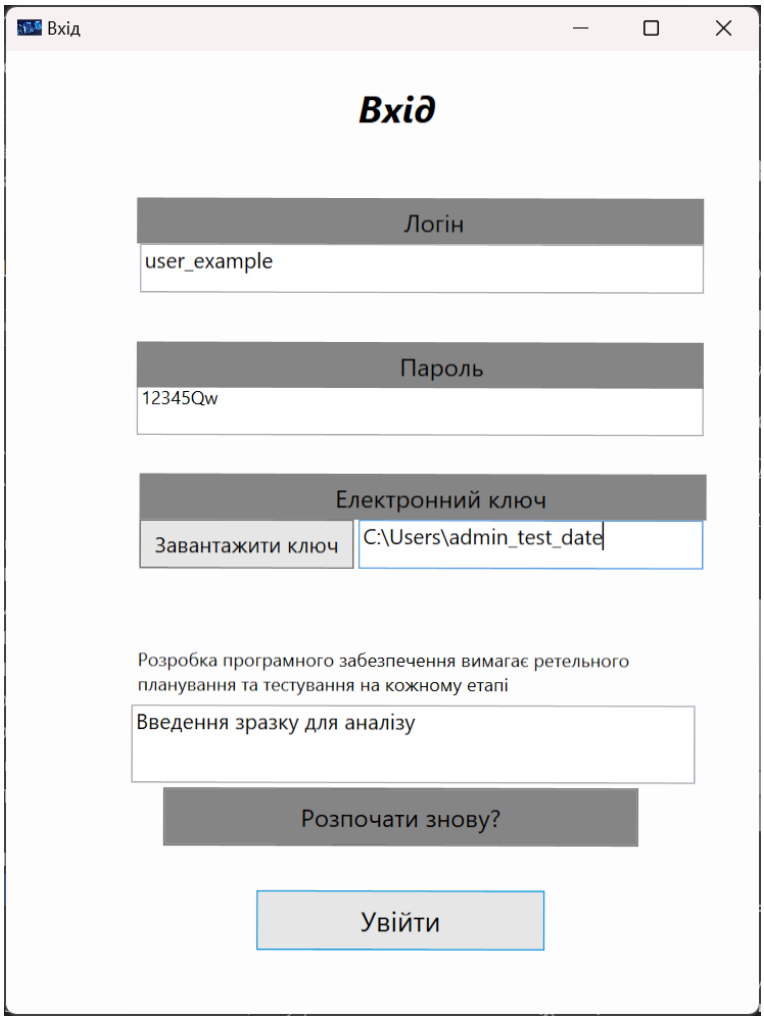


Рисунок 3.12 – Вигляд вікна форми авторизації користувача

У випадку, якщо користувач виконав усі умови вірно, його спроба авторизації буде успішною, програма про це повідомить відповідним сповіщенням (рис. 3.13).

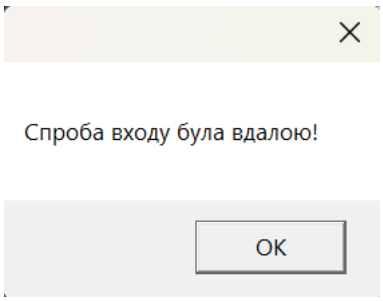


Рисунок 3.13 – Вигляд вікна із сповіщенням про успішну авторизацію

У випадку, якщо користувач ввів некоректні дані, клвіатурний почерк на певному етапі не відповідає еталонним зразкам або флеш-носій не є ключем для вказаного користувача, програма про це сповіщає та вказує на обмежену

кількість спроб авторизації (рис. 3.14).

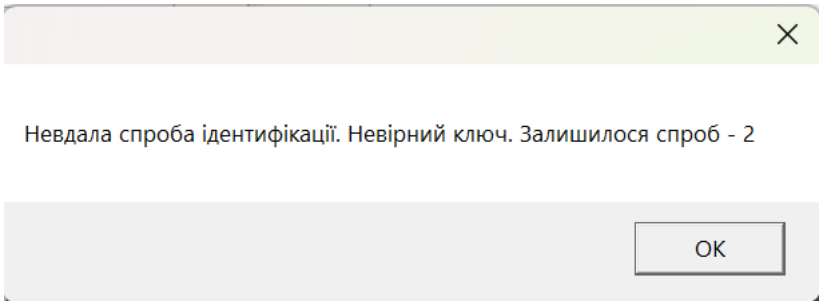


Рисунок 3.14 – Вигляд вікна із сповіщенням про невдалу спробу авторизації

У випадку, якщо користувач намагатиметься увійти в систему, але при цьому не заповнить усі необхідні поля, програма повідомлятиме про це сповіщеннями із відповідним текстом (рис. 3.15).

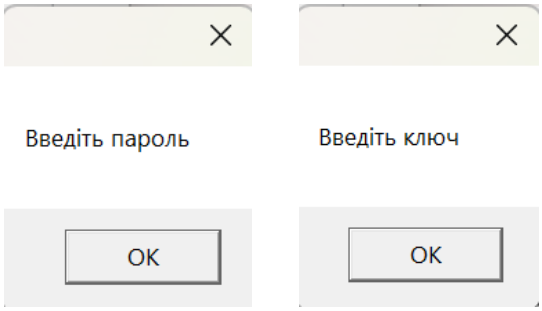


Рисунок 3.15 – Вигляд вікна із сповіщенням про необхідність введення відповідних даних

Після здійснення успішного процесу реєстрації, користувачеві відкривається головне вікно інформаційної системи, де він може продовжити свою роботу (рис. 3.16).

Натиснувши кнопку «Особистий кабінет», користувач може переглянути свої облікові відомості (рис. 3.17). Користувачеві доступна можливість змінити ПІБ та електронну пошту, проте поля логіну та ролі залишають заблокованими, оскільки такі дані може змінювати лише адміністратор. При цьому дані про пароль та флеш-носій не відображаються взагалі, оскільки вони використовуються при авторизації в системі та є еталонними зразками, а отже не можуть бути змінені чи відредаговані ні за бажанням користувача, ні адміністратора. Такі заходи впроваджуються в цілях безпеки.

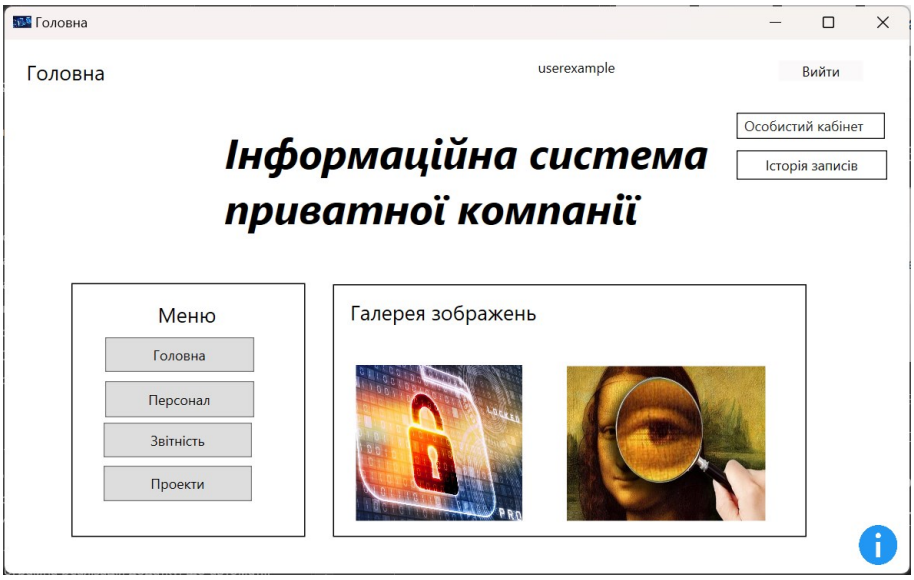


Рисунок 3.16 – Вигляд головного вікна системи для облікового запису користувача

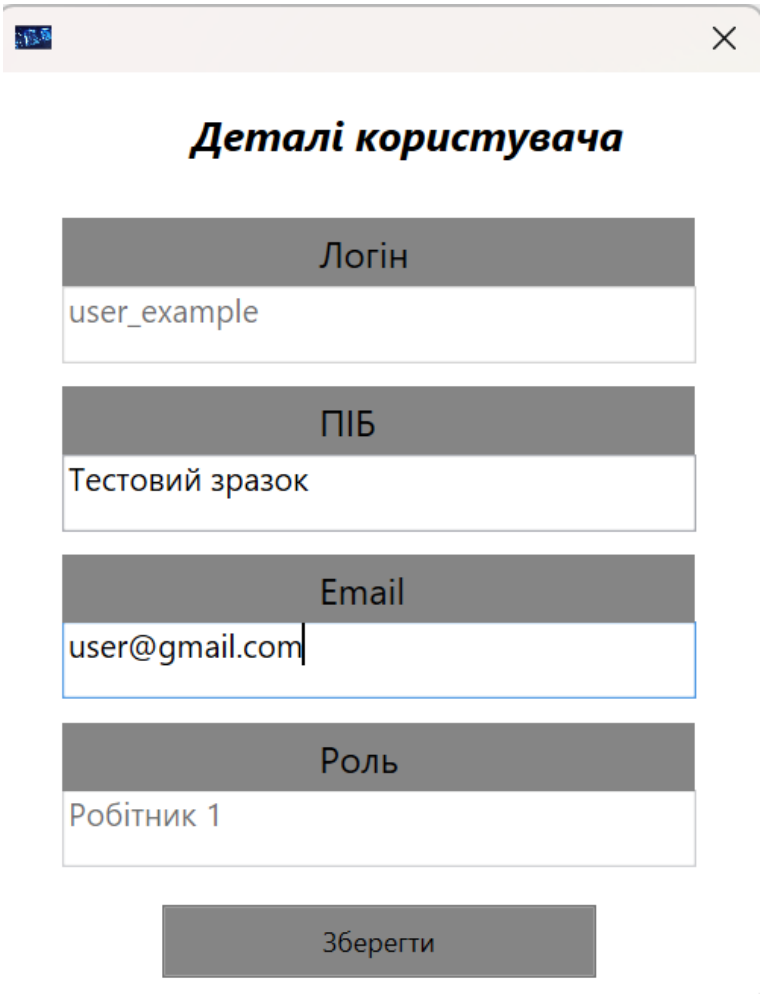


Рисунок 3.17 – Вигляд вікна «Особистий кабінет» для облікового запису користувача

Отже, в даному підрозділі була наведена інструктція користувача для роботи із розробленим програмним додатком для адміністратора та користувача. Варто зауважити, що з метою демонстрації були наданні тестові ролі та сторінки інформаційної системи. При застосуванні розробки на практиці, програмну розробку можна доповнювати різноманітними функціями розширюючи коло ролей та їх функціональних можливостей.

3.4 Аналіз тестування ПЗ та коригування підходу до захисту ІС

Для оцінки ефективності удосконаленого методу проведемо його тестування на основі розробленого програмного застосунку.

На першому етапі дослідимо значення показників помилки, які використовуються для оцінювання якості біометричної ідентифікації.

Такими показниками є чисельне значення ймовірностей FAR (ймовірність помилкового доступу) і FRR (ймовірність помилкової відмови в доступі).

$$\mathrm{FAR} = \frac{\mathrm{NFA}}{\mathrm{NIVA}} \cdot 100\%,$$

де FAR – ймовірність помилкового доступу;

NFA – кількість фактів помилкового доступу;

NIVA – загальна кількість звернень до системи.

$$\mathrm{FRR} = \frac{\mathrm{NFF}}{\mathrm{NEVA}} \cdot 100\%\,,$$

де FRR – ймовірність помилкової відмови в доступі;

NFF – кількість фактів відмови в доступі;

NEVA – загальна кількість звернень до системи.

Здійснимо аналіз біометричної ідентифікації для реалізованого дуального методу клавіатурного почерку, клавіатурного почерку за зразком паролю та клавіатурного почерку за завчасно невідомою користувачеві фразою.

В дослідженні приймало участь 10 користувачів, кожен здійснював по 5 спроб входу. Наведемо середні значення показників для кожного виду дослідження клавіатурного почерку користувачів.

Отримані результати тестування наведено у табл. 3.1 та 3.2

Таблиця 3.1 – Порівняння показників FAR і FRR (для користувачів 1 – 5)

Метод	Корист. 1		Корист. 2		Корист. 3		Корист. 4		Корист. 5	
	FAR, %	FRR, %	FAR, %	FRR, %	FAR, %	FRR, %	FAR, %	FRR, %	FAR, %	FRR, %
Дуальний метод	0	20	0	0	0	20	0	0	0	0
На основі відомого паролю	20	20	0	20	0	0	0	0	40	0
На основі невідомої фрази	0	20	20	0	0	0	0	0	20	20

Таблиця 3.2 – Порівняння показників FAR і FRR, (для корист. 6 – 10)

Метод	Корист. 6		Корист. 7		Корист. 7		Корист. 9		Корист. 10	
	FAR, %	FRR, %	FAR, %	FRR, %	FAR, %	FRR, %	FAR, %	FRR, %	FAR, %	FRR, %
Дуальний метод	0	20	0	0	0	20	0	0	0	0
На основі відомого паролю	0	60	0	40	20	20	0	20	20	60
На основі невідомої фрази	0	40	20	0	20	0	0	0	0	20

Наведені в табл. 3.1 та 3.2 результати тестування дуального методу клавіатурного почерку демонструють низькі показники ймовірності відмови в доступі зареєстрованому користувачеві і виключають можливість доступу не відомому системі користувача, що свідчить про ефективність розробки та її доцільність використання на практиці. Такі показники порівняно з іншими методами досягнуті за рахунок підвищення точності та якості ідентифікації, що дозволяє врахувати більшість факторів, що впливають на введення даних з клавіатури (час доби, емоційний стан, підготовленість до вводу конкретних даних і т.п.).

Наступним етапом перевірки здійснимо дослідження можливості підробки результату. Для цього протестуємо зразки п’ятьох користувачів та порівняємо з еталонним. Показник за яким буде здійснюватися порівняння – це час утримання клавіші клавіатури. Тест проводився на текстовій послідовності в 200 символів, спроб для кожного користувача по 5.

Результати досліду представлено на рис. 3.18.

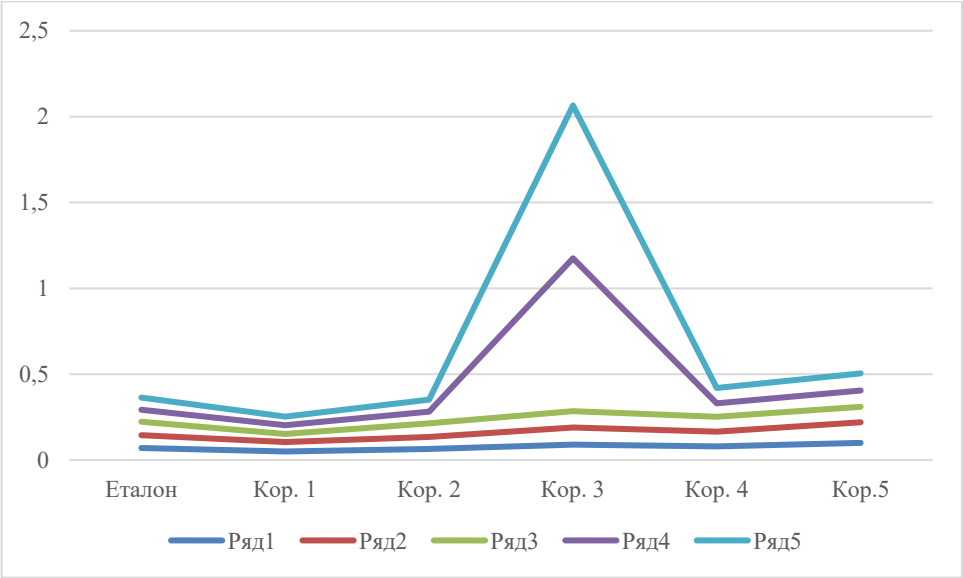


Рисунок 3.18 – Результати тестування користувачів за клавіатурним почерком

Як можна побачити з наведеного рис. 3.18, навіть за однакових умов та фрагментів друкованого тексту – результати клавіатурного почерку цілковито різняться. Це доводить їх унікальність та ефективність у застосуванні у цілях ідентифікації користувача.

Для перевірки практичності застосування флеш-носія з метою контролю доступу до ІС було проведено експериментальне дослідження: створено десять користувачів, згенеровано для них ключі та зафіксовано еталони клавіатурного почерку. У тестовому режимі здійснювалися спроби авторизації користувачів за наявними в системі зразками клавіатурного почерку та згенерованими ключами. Підбір флеш-носія відбувався таким чином, що він не належав користувачеві, який вводив свій зразок клавіатурного почерку. Результатом дослідження є 100% відмова в доступі усім користувачам з переплутаними даними, проте наявними в системі. Такий експеримент свідчить про високу якість програмної розробки, що в повній мірі підтримує та реалізує запланований функціонал системи.

Отже, виходячи із отриманих результатів тестування, можна зробити висновок, що удосконалений метод на основі дуальної перевірки клавіатурного почерку та захищеного флеш-носія працює коректно та має високі якісні показники.

3.5 Висновки до розділу 3

У даному розділі було здійснено практичну реалізацію програмного продукту на основі удосконаленого методу для забезпечення захищеного доступу до інформаційної системи.

Під час написання розділу було наведено покроковий алгоритм роботи програмного застосунку, що автоматизує розроблений підхід для захищеного доступу до ІС, описано особливості програмної реалізації та наведено відповідні фрагменти коду на мові C#, описано інструкцію користувача для роботи з програмою із наведеними діалоговими вікнами розробки.

Для визначення практичної цінності та якості розробки на основі удосконаленого методу було проведено її тестування. Перевірка додатку відбувалася на основі дослідження показників FAR (ймовірність помилкового доступу) та FRR (ймовірність помилкової відмови в доступі), визначення можливості підробки клавіатурного почерку та експерименту щодо ефективності роботи системи за умови переплутаних (між користувачами) факторів авторизації. Всі етапи тестування засвідчили позитивні показники та високу якість розробки на основі удосконаленого методу.

4 ЕКОНОМІЧНА ЧАСТИНА

Метою даного розділу є проведення дослідження економічного потенціалу розробки, зокрема, оцінювання комерційного потенціалу; прогнозування витрат на виконання наукової роботи та впровадження її результатів; прогнозування комерційних ефектів від реалізації результатів розробки та розрахунок ефективності вкладених інвестицій та періоду їх окупності.

Результати даного дослідження уможливлюють прийняття рішення про економічну доцільність розробки програмного засобу з використанням захищеного флеш-носія, двофакторної ідентифікації на основі аналізу клавіатурного почерку та оптимізацією звернень до бази даних на основі генетичного алгоритму.

4.1 Оцінювання комерційного потенціалу розробки програмного забезпечення

Метою проведення технологічного аудиту є оцінювання комерційного потенціалу розробки, створеної в результаті науково-технічної діяльності [50].

Результатом магістерської кваліфікаційної роботи є розробка програмного засобу з використанням захищеного флеш-носія, двофакторної ідентифікації на основі аналізу клавіатурного почерку та оптимізацією звернень до бази даних на основі генетичного алгоритму.

Для проведення технологічного аудиту залучено трьох незалежних експертів.

У межах даної роботи такими експертами є викладачі кафедри МБІС: Карпінець В. В. (к.т.н., доцент каф. МБІС ВНТУ); Салієва О.В. (д.ф., викл. каф. МБІС ВНТУ); Грицак А. В. (доц., викл. каф. МБІС ВНТУ).

Оцінювання комерційного потенціалу буде здійснено за критеріями, що наведені в таблиці 4.1

Таблиця 4.1 – Критерії оцінювання комерційного потенціалу розробки бальна оцінка

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
Кри-терій	0	1	2	3	4
Технічна здійсненність концепції:					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено роботоздатність продукту в реальних умовах
Ринкові переваги (недоліки):					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів
Критерії оцінювання та бали (за 5-ти бальною шкалою)					
Крите-р.	0	1	2	3	4
4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів
5	Експлуатацій ні витрати значно вищі, ніж в аналогів	Експлуатацій ні витрати дещо вищі, ніж в аналогів	Експлуатацій ні витрати на рівні експл. витрат аналогів	Експлуатацій ні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
Ринкові перспективи					
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкурентів немає

Продовження таблиці 4.1

Практична здійсненність					
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витрачати значні кошти та час на навч. наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої к-ті дозвільних документів на вир-во та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

Результати оцінювання комерційного потенціалу експертами розробки зведено в таблицю 4.2.

Таблиця 4.2 – Результати оцінювання комерційного потенціалу розробки

Критерії	Прізвище, ініціали, посада експерта		
	1 – Карпінець В.В.	2 – Салієва О.В.	3 – Грицак А.В.
1	3	3	3
Ринкові переваги (недоліки):			
2	4	4	4
3	3	3	4
4	3	4	4
5	3	4	4
Ринкові перспективи			
6	3	4	3
7	3	3	3
Практична здійсненність			
8	4	4	4
9	3	3	3
10	4	4	4
11	4	4	4
12	3	3	3
Сума балів	$СБ_1 = 40$	$СБ_1 = 43$	$СБ_1 = 43$
Середньоарифметична сума балів $\overline{СБ}$	$\overline{СБ} = 42$		

За даними таблиці 4.2 можна зробити висновок, щодо рівня комерційного потенціалу розробки. Врахуємо на результат й порівняємо його з рівнями комерційного потенціалу розробки, що представлено в таблиці 4.3.

Таблиця 4.3 – Рівні комерційного потенціалу розробки

Середньоарифметична сума балів $\overline{СБ}$, розрахована на основі висновків експертів	Рівень комерційного потенціалу розробки
0 – 10	Низький
11 – 20	Нижче середнього
21 – 30	Середній
31 – 40	Вище середнього
41 – 48	Високий

Рівень комерційного потенціалу розробки, становить 42 бали, що відповідає рівню «високий».

Наступним кроком здійснимо аналіз технічної задачі та дослідимо аналоги. Наукова новизна розробки полягає в покращенні захисту інформаційних систем з використанням захищеного флеш-носія, двофакторної ідентифікації на основі аналізу клавіатурного почерку та оптимізацією звернень до бази даних на основі генетичного алгоритму

Розробка програмного засобу має на меті практичну реалізацію алгоритму на основі удосконаленого методу з подальшим практичним тестуванням.

Виходячи із отриманих результатів дослідження розробленого програмного продукту на основі вдосконаленого методу, можна вважати, що розроблений алгоритм надає можливість реалізації та впровадження ефективної системи контролю доступу на основі біометричних даних користувача, а саме клавіатурного почерку та апаратного рішення у вигляді флеш-носія, що не потребує значних фінансових затрат та є доступним для широкого загалу користувачів.

Такі результати досягнуто за рахунок здійсненого в роботі удосконаленого методу, а саме комплексного поєднання засобів контролю доступу та авторизації користувача у зручному та доступному додатку.

Враховуючи виявлені переваги розробленого методу, порівняємо його з аналогами. У таблиці 4.4 наведені основні технічні показники аналога і нового програмного продукту.

Таблиця 4.4 – Основні технічні показники аналога і нового програмного продукту

Показники, %	Аналог	Нова розробка	Відношення параметрів нової розробки до параметрів аналога
Функціональність	80	100	1,25
Надійність	75	100	1,33
Сумісність	60	100	1,67
Супровід	90	100	1,11
Економія ресурсів і часу	70	100	1,42
Простота використання	80	100	1,25

Порівняно з аналогами, розроблений програмний додаток на основі удосконаленого методу, є універсальним та доступним засобом для здійснення контролю доступу до інформаційної системи. Особливо високі результати мають показники сумісності та економії часу та ресурсів, оскільки

розроблюваний програмний продукт не є «вибагливим» до системних вимог, а його встановлення та навчання персоналу не потребують багато часу.

Функціональність, стійкість, ефективність та точність програмної розробки була перевірена практично під час етапу тестування.

Нове технічне рішення, яке пропонується для розробки, матиме високі показники, порівняно з аналогами та більшою мірою задовольнить потреби споживачів. Тому його розробка та впровадження є актуальним та доцільними.

Програмна розробка на основі обраного та вдосконаленого алгоритму сьогодні є актуальною для використання в корпоративних інформаційних системах..

Для розповсюдження розробки можуть бути використанні торгові та рекламні майданчики інтернет-магазинів з продажу програмного забезпечення та/або офіційний сайт додатку із встановленою середньою ціною на продукт.

Оскільки, на сьогодні питання інформаційних систем та контролю доступу до них є актуальним, а технології, що реалізують дані можливості успішно використовуються для вирішення таких завдань – цілком ймовірний високий попит на запропонований програмний продукт.

4.2 Прогнозування витрат на виконання наукової роботи та впровадження її результатів

Прогнозування витрат на виконання науково-дослідної, дослідно-конструкторської та конструкторсько-технологічної роботи складається з таких етапів:

1-й етап: розрахунок витрат, які безпосередньо стосуються виконавців даного розділу роботи;

2-й етап: розрахунок загальних витрат на виконання даної роботи;

3-й етап: прогнозування загальних витрат на виконання та впровадження результатів даної роботи.

Виконаємо розрахунок витрат, які безпосередньо стосуються виконавців даного розділу роботи, за такими статтями та формулами, приймаючи до уваги

те, що для розробки інформаційної технології було залучено одного розробника програмного забезпечення.

1. Основна заробітна $З_о$:

$$З_о = \frac{М}{Т_р} \cdot t, \text{грн.} \quad (4.1)$$

де М – місячний посадовий оклад 42 000 грн.(для розробника),35 000 грн (для керівника проекту), 10 500 (для менеджера проекту);

$Т_р$ – число робочих днів в місяці; приблизно $Т_р = 21$ день;

t – число робочих днів для всієї роботи – 30 днів.

Таким чином:

$$З_1 = \frac{42\,000}{21} \cdot 30 = 60\,000 \text{ (грн.)} - \text{для розробника}$$

$$З_1 = \frac{35\,000}{21} \cdot 30 = 50\,000 \text{ (грн.)} - \text{для керівника}$$

$$З_1 = \frac{10\,500}{21} \cdot 30 = 15\,000 \text{ (грн.)} - \text{для розробника}$$

Таблиця 4.5 – Витрати по заробітній платі

Найменування посади	Місячний посадовий оклад, грн.	Оплата за робочий день, грн.	Число днів роботи	Витрати на заробітну плату
Керівник проекту	35 000	1 666	30	50 000
Розробник	42 000	2 000	30	60 000
Менеджер	10 500	500	30	15 000
Всього				125 000

$$З_о = 125\,000 \text{ (грн.)}$$

Витрати на основну заробітну плату робітників ($З_р$) за відповідними найменуваннями робіт розраховують за формулою:

$$З_р = \sum_{i=1}^{n} C_i \cdot t_i, \quad (4.2)$$

де $C_i -$ погодинна тарифна ставка робітника відповідного розряду, за виконану відповідну роботу, грн/год;

$t_i -$ час роботи робітника на виконання певної роботи, год.

Погодинну тарифну ставку робітника відповідного розряду C_i можна визначити за формулою:

$$C_i = \frac{M_M \cdot K_i \cdot K_C,}{T_p \cdot t_{\text{зм}}} \quad (4.3)$$

де $M_M -$ розмір прожиткового мінімуму працездатної особи або мінімальної місячної заробітної плати (залежно від діючого законодавства), грн; $K_i -$ коефіцієнт міжкваліфікаційного співвідношення для встановлення тарифної ставки робітнику відповідного розряду; $K_C -$ мінімальний коефіцієнт співвідношень місячних тарифних ставок робітників першого розряду з нормальними умовами праці виробничих об'єднань і підприємств до законодавчо встановленого розміру мінімальної заробітної плати.

$T_p -$ середня кількість робочих днів в місяці, приблизно $T_p = 21…23$ дні;

$t_{\text{зм}} -$ тривалість зміни, год.

$$C_1 = \frac{8000 * 1{,}1 * 1{,}65}{21 * 8} = 82{,}5 \text{ (грн.)}$$

$$З_{р1} = 82{,}5 \cdot 4 = 330 \text{ (грн.)}$$

Проведені розрахунки потрібно звести до таблиці.

Таблиця 4.6 – Величина витрат на основну заробітну плату робітників

Найменування робіт	Тривалість роботи, год	Розряд роботи	Тарифний коефіцієнт	Погодинна тарифна ставка, грн	Величина оплати на робітника грн
Налаштування комп'ютерного обладнання	4	2	1,1	82,5	330
Організація робочого простору	3	2	1,1	82,5	247,5
Розробка ПЗ	3	5	1,7	127,5	385,5
Тестування ПЗ	2	2	1,1	82,5	160
				Всього	1123,0

$$З_р = 1123{,}0 \text{ (грн.)}$$

2. Додаткова заробітна плата $З_д$ працівників розраховується як 12% від основної заробітної плати:

$$З_д = 0{,}12 \cdot (125\,000 + 1123{,}0) = 15\,134{,}8\ (\text{грн.})$$

3. Нарахування на заробітну плату дослідників та робітників розраховується як 22% від суми основної та додаткової заробітної плати дослідників і робітників за формулою:

$$З_з = (З_о + З_р + З_{дод}) \cdot \frac{Н_{зп}}{100\%} \quad (4.4)$$

де $Н_{зп}$ – норма нарахування на заробітну плату.

$$З_н = (125\,000 + 1123{,}0 + 15\,134{,}8) \cdot \frac{22}{100} = 31\,076{,}7\ (\text{грн.})$$

4. Амортизація обладнання, комп'ютерів та приміщень, які використовувались під час виконання даного етапу роботи розраховуємо за формулою:

$$А = \frac{Ц \cdot Т}{12 \cdot Т_В} \quad (4.5)$$

де Ц – загальна балансова вартість обладнання, приміщення тощо, грн.;

Т – фактична тривалість використання, міс;

$Т_В$ – термін використання обладнання, приміщень тощо, роки.

Розробка програмного забезпечення ведеться орієнтовно 1,3 місяці.

Для офісного приміщення $А = \frac{50\,000 \cdot 1{,}3}{12 \cdot 20} = 270{,}8 \text{грн.}$; для комп'ютера $А = \frac{47\,000 \cdot 1{,}3}{12 \cdot 2} = 2\,545{,}8$ грн.; для монітора $А = \frac{14\,000 \cdot 1{,}3}{12 \cdot 2} = 758 \text{грн.}$

Розрахунки зведено до таблиці 4.7:

Таблиця 4.7 – Амортизаційні відрахування

Наймену-вання	Балансова вартість (грн.)	Термін використання (років)	Фактична тривалість в-ня, (міс.)	Величина ам.. відрахувань, (грн..)
Офісне приміщення	50 000	20	1,3	270,8
Комп'ютер	47 000	2	1,3	2 545,8
Монітор	14 000	2	1,3	758
			Всього	3 574,6

5. Витрати на матеріали M, що були використані під час виконання даного етапу роботи, розраховуються за формулою:

$$M = \sum_{1}^{n} \text{Н}_j \cdot \text{Ц}_j \cdot K_j - \sum_{j=1}^{n} B_j \cdot \text{Ц}_{\text{в}j} \qquad (4.8)$$

де Н_j – норма витрат матеріалу j-го найменування, кг;

n – кількість видів матеріалів;

Ц_j – вартість матеріалу j-го найменування, грн/кг;

K_j – коефіцієнт транспортних витрат, (Кj = 1,1 … 1,15);

B_j – маса відходів j-го найменування, кг;

$\text{Ц}_{\text{в}j}$ – вартість відходів j-го найменування, грн/кг.

Проведені розрахунки потрібно звести до таблиці.

Таблиця 4.8 – Витрати на матеріали

Найменування комплектувальних	Кількість	Ціна за штуку, грн.	Сума, грн.	Примітка
Клавіатура	3	1200 грн.	3 600 грн.	
Комп'ютерна мишка	1	800 грн.	800 грн.	
Флеш-носії	5	500 грн.	2 500 грн.	
Всього:			$M_{\text{і}} = 1{,}1$	7 590 грн.

6. Витрати на силову електроенергію $\text{В}_\text{е}$ розраховуються за формулою:

$$\text{В}_\text{е} = \sum_{i=1}^{n} \frac{W_{yi} \cdot t_i \cdot \text{Ц}_\text{е} \cdot K_{\text{впі}}}{\eta_i} \qquad (4.9)$$

де W_{yi} – встановлена потужність обладнання на певному етапі розробки, кВт;

t_i – тривалість роботи обладнання на етапі дослідження, год;

Ц$_е$- вартість 1 кВт-години електроенергії, грн; (вартість електроенергії визначається за даними енергопостачальної компанії);

$K_{впі}$ – коефіцієнт, що враховує використання потужності, К$_{впі}$ <1;

η_i – коефіцієнт корисної дії обладнання, η_i<1.

Проведені розрахунки необхідно звести до таблиці.

$$В_е = \frac{0{,}6 \cdot 240 \cdot 7{,}50 \cdot 0{,}95}{0{,}97} = 1\,057{,}7\ (грн.)$$

$$В_е = \frac{0{,}35 \cdot 240 \cdot 7{,}50 \cdot 0{,}95}{0{,}97} = 617\ (грн.)$$

$$В_е = \frac{0{,}2 \cdot 240 \cdot 7{,}50 \cdot 0{,}95}{0{,}97} = 352{,}6\ (грн.)$$

Таблиця 4.9 – Витрати на електроенергію

Найменування обладнання	Встановлена потужність, кВт	Тривалість роботи, год	Сума, грн
Комп’ютер	0,6	240	1 057,7
Монітор	0,35	240	617
Офісне приміщення	0,2	240	352,6
		Всього	2 027,3

7. Інші витрати В$_{ін}$ охоплюють:

– витрати на управління організацією;

– оплату службових відряджень;

– витрати на утримання, ремонт та експлуатацію основних засобів;

– витрати на опалення, освітлення, водопостачання, охорону праці тощо.

Інші витрати $В_{ін}$ можна прийняти як 100% від суми основної заробітної плати усіх працівників:

$$В_{ін} = 125\,000 \cdot 1 = 60\,000 \text{ (грн)}$$

Послуги Інтернету – 400 грн., канцтовари – 200 грн., інші офісні потреби – 200 грн. Загальна вартість становить:

$$400 + 200 + 200 = 800 \text{ (грн.)}$$

8. Сума всіх попередніх статей витрат дає витрати на виконання даної частини роботи – В.

$$В = 125\,000 + 1123 + 15\,134{,}8 + 31\,076{,}7 + 3\,574{,}6 + 7590 + 2027{,}3 + 125\,000 + 800 = 279\,481{,}5 \text{ (грн.)}$$

9. Проведемо прогнозування загальних витрат ЗВ на виконання та впровадження результатів виконаної наукової роботи. Прогнозування здійснюється за формулою:

$$ЗВ = \frac{В_{заг}}{\beta}, \text{грн.} \qquad (4.10)$$

де β – коефіцієнт, який характеризує етап (стадію) виконання даної роботи.

Так, якщо розробка знаходиться: на стадії науково-дослідних робіт, то $\beta \approx 0{,}1$; на стадії технічного проектування, то $\beta \approx 0{,}2$; на стадії розробки конструкторської документації, то $\beta \approx 0{,}3$; на стадії розробки технологій, то $\beta \approx 0{,}4$; на стадії розробки дослідного зразка, то $\beta \approx 0{,}5$; на стадії розробки промислового зразка, $\beta \approx 0{,}7$; на стадії впровадження, то $\beta \approx 0{,}9$.

$В_{заг}$ – загальна вартість всієї наукової роботи.

$$В = 279\,481{,}5 \text{ (грн.)}$$

$$ЗВ = \frac{279\,481{,}5}{0{,}7} = 399\,259{,}3 \text{ (грн.)}$$

Отже, прогноз загальних витрат ЗВ на виконання та впровадження результатів виконаної наукової роботи складає $399\,259{,}3$ (грн.)

4.3 Прогнозування комерційних ефектів від реалізації результатів розробки

У даному підрозділі проведемо кількісне прогнозування, яку вигоду можна отримати у майбутньому від впровадження результатів виконаної наукової роботи.

В умовах ринку узагальнюючим позитивним результатом, що його отримує підприємство від впровадження результатів тієї чи іншої розробки, є збільшення чистого прибутку підприємства. Зростання чистого прибутку можна оцінити у теперішній вартості грошей.

Зростання чистого прибутку забезпечить інвестору надходження додаткових коштів, які дозволять покращити фінансові результати діяльності.

Виконання даної наукової роботи та впровадження її результатів складає приблизно 1 рік. Позитивні результати від впровадження розробки очікуються вже в перші місяці після впровадження.

Проведемо детальне прогнозування позитивних результатів та кількісне їх оцінювання по роках.

Майбутній економічний ефект буде базуватися на таких даних:

$\Delta N -$ збільшення кількості споживачів продукту у періоди, що аналізуються, завдяки покращенню його характеристик:

Перший рік – 450 користувачів;

Другий рік – 750 користувачів;

Третій рік – 900 користувачів.

N – кількість споживачів, які використовували аналогічний продукт до впровадження результатів нової науково-технічної розробки, становила 300 користувачів.

$Ц_б -$ вартість програмного продукту до впровадження результатів розробки складала 10 000 грн.

$\pm \Delta Ц_0$ – зміна вартості програмного продукту після впровадження результатів науково-технічної розробки складе 1500 грн.

Таким чином, запровадження інноваційної розробки призведе до збільшення кількості користувачів та вартості продукту, що позитивно вплине на чистий прибуток потенційного інвестора.

Можливе збільшення чистого прибутку у потенційного інвестора для кожного з трьох років, протягом яких очікується отримання позитивних результатів від впровадження та комерціалізації науково-технічної розробки, розраховується за формулою:

$$\Delta П_i = (\pm \Delta Ц_0 \cdot N + Ц_0 \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot \left(1 - \frac{\vartheta}{100}\right), \quad (4.11)$$

де $\lambda -$ коефіцієнт, який враховує сплату потенційним інвестором податку на додану вартість. У 2024 році ставка податку на додану вартість складає 20%, а коефіцієнт $\lambda = 0{,}8333$;

$\rho -$ коефіцієнт, який враховує рентабельність інноваційного продукту. Приймемо $\rho = 30\%$;

$\vartheta -$ ставка податку на прибуток, який має сплачувати потенційний інвестор, у 2024 році $\vartheta = 18\%$.

Збільшення чистого прибутку 1-го року:

$$\Delta П_1 = (1500 \times 300 + 10\,000 \times 450) \times 0{,}83 \times 0{,}3 \times \left(1 - \frac{0{,}18}{100}\right)$$
$$= 1\,230\,331\ (\text{грн.})$$

Збільшення чистого прибутку 2-го року:

$$\Delta П_2 = (1500 \times 300 + 10\,000 \times (450 + 750)) \times 0{,}83 \times 0{,}3 \times \left(1 - \frac{0{,}18}{100}\right)$$
$$= 3\,094\,470\ (\text{грн.})$$

Збільшення чистого прибутку 3-го року:

$$\Delta П_3 = (1500 \times 300 + 10\,000 \times (450 + 750 + 900)) \times 0{,}83 \times 0{,}3 \times \left(1 - \frac{0{,}18}{100}\right)$$
$$= 5\,331\,436\ (\text{грн.})$$

Отже, розрахунки показують, що відповідно прогнозуванню комерційний ефект від впровадження розробки виражається у значному збільшенні чистого прибутку підприємства.

4.4 Розрахунок ефективності вкладених інвестицій та періоду їх окупності

Основними показниками, які визначають доцільність фінансування наукової розробки певним інвестором, є абсолютна і відносна ефективність вкладених інвестицій та термін їх окупності.

Розрахунок ефективності вкладених інвестицій передбачає:

1-й крок. Розрахунок теперішньої вартості інвестицій PV, що вкладаються в наукову розробку. Такою вартістю ми можемо вважати прогнозовану величину загальних витрат ЗВ на виконання помножену на коефіцієнт:

$$PV = k_{\text{інв}} \cdot \text{ЗВ}, \tag{4.12}$$

де $k_{\text{інв}}$ –– коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію. Це можуть бути витрати на підготовку приміщень, розробку технологій, навчання персоналу, маркетингові заходи тощо; зазвичай інв k =2...5, але може бути і більшим;

ЗВ – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, грн.

$$PV = 399\,259{,}3 \cdot 3 = 1\,197\,777{,}9\ (\text{грн.})$$

2-й крок. Розрахуємо очікуване збільшення прибутку $\Delta\text{П}_\text{і}$, що його отримає підприємство (організація) від впровадження результатів наукової розробки, для кожного із років, починаючи з першого року впровадження.

Таке збільшення прибутку також було розраховане нами раніше та становить:

$$\Delta\text{П}_1 = 1\,230\,331\ \ (\text{грн}), \Delta\text{П}_2 = 3\,094\,470\ (\text{грн}), \Delta\text{П}_3 = 5\,331\,436\ (\text{грн}).$$

3-й крок. Розрахуємо абсолютну ефективність вкладених інвестицій $\text{Е}_{\text{абс}}$ за формулою:

$$\text{Е}_{\text{абс}} = (\text{ПП} - PV), (\text{грн.}) \tag{4.8}$$

де ПП – приведена вартість всіх чистих прибутків, що їх отримає підприємство (організація) від реалізації результатів наукової розробки, грн.;

Приведена вартість всіх чистих прибутків ПП розраховується за формулою:

$$ПП = \sum_{1}^{T} \frac{\Delta П_i}{(1+\tau)^t}, (грн) \quad (4.9)$$

де $\Delta П_i$ – збільшення чистого прибутку у кожному із років, протягом яких виявляються результати виконаної та впровадженої НДДКР, грн.;

Т – період часу, протягом якого виявляються результати впровадженої НДДКР, роки;

τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні – 0,1;

t – період часу (в роках) від моменту отримання чистого прибутку до точки «0»;

$$ПП = \frac{1\ 230\ 331}{(1+0{,}1)^1} + \frac{3\ 094\ 470}{(1+0{,}1)^2} + \frac{5\ 331\ 436}{(1+0{,}1)^3} = 7\ 681\ 483\ (грн.)$$

$$Е_{абс} = 7\ 681\ 483 - 1\ 197\ 777{,}9 = 6\ 483\ 705\ (грн.)$$

Оскільки $Е_{абс} > 0$, результат від проведення наукових досліджень щодо розробки програмного продукту та їх впровадження принесе прибуток, тобто є доцільним, проте це ще не свідчить про те, що інвестор буде зацікавлений у фінансуванні даної програми.

4-й крок. Розрахуємо відносну (щорічну) ефективність вкладених в наукову розробку інвестицій $Е_в$ за формулою:

$$Е_в = \sqrt[Т_ж]{1 + \frac{Е_{абс}}{PV} - 1} \quad (4.10)$$

де $Е_{абс}$ – абсолютна ефективність вкладених інвестицій, грн.;

PV – теперішня вартість інвестицій $PV = ЗВ, грн.$

$Т_ж$ – життєвий цикл наукової розробки, роки.

$$E_в = \sqrt[3]{1 + \frac{6\ 483\ 705}{1\ 197\ 777{,}9}} - 1 = \sqrt[3]{6{,}4} - 1 = 0{,}85 \text{ або } 85\%$$

Порівняємо $E_в$ з мінімальною (бар'єрною) ставкою дисконтування τ_{min}, яка визначає ту мінімальну дохідність, нижче за яку інвестиції вкладатися не

будуть.

Спрогнозуємо величину τ_{min}.

У загальному вигляді мінімальна (бар'єрна) ставка дисконтування τ_{min}визначається за формулою:

$$\tau_{min} = d + f \quad (4.11)$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; $d = 0{,}2$;

f – показник, що характеризує ризикованість вкладень;

величина $f = 0{,}5$.

$$\tau_{min} = 0{,}2 + 0{,}5 = 0{,}7$$

Оскільки $E_в = 85\% > \tau_{min} = 70\%$, то у інвестора є потенційна зацікавленість у фінансуванні даної наукової розробки.

6-й крок. Розрахуємо термін окупності вкладених у реалізацію наукового проекту інвестицій $Т_{ок}$ за формулою:

$$Т_{ок} = \frac{1}{Е_в}, \text{рік} \quad (4.12)$$

$$Т_{ок} = \frac{1}{0{,}85} = 1{,}2 \text{ (року)}$$

Оскільки термін окупності вкладених у реалізацію наукового проекту інвестицій менше трьох років ($Т_{ок} < 3$ років), то фінансування нової розробки є доцільним.

4.5 Висновки до розділу 4

В даному розділі було виконано оцінювання комерційного потенціалу розробки програмного засобу на основі вдосконаленого методу для покращення захисту інформаційних систем з використанням захищеного флеш-носія, двофакторної ідентифікації на основі аналізу клавіатурного почерку та оптимізацією звернень до бази даних на основі генетичного алгоритму.

Проведено технологічний аудит з залученням трьох незалежних експертів. Визначено, що рівень комерційного потенціалу розробки вище середнього. Проведено порівняння з аналогами.

Згідно з проведеним оцінюванням нова розробка є якісною та конкурентоспроможною. Рівень комерційного потенціалу розробки, становить 42 бали, що відповідає рівню «високий».

Згідно із розрахунками всіх статей витрат на виконання науково-дослідної, дослідно-конструкторської та конструкторсько-технологічної роботи загальні витрати на розробку складають 399 259,3 (грн.). Розрахована абсолютна ефективність вкладених інвестицій в сумі 6 483 705 (грн.) свідчить про отримання прибутку інвестором від комерціалізації програмного продукту.

Щорічна ефективність вкладених в наукову розробку інвестицій складає 85%, що вище за мінімальну бар'єрну ставку дисконтування, яка складає 70%. Це означає потенційну зацікавленість інвесторів у фінансуванні розробки.

Термін окупності вкладених у реалізацію проекту інвестицій становить 1,2 (року), що також свідчить про доцільність фінансування нової розробки.

Отже, в результаті аналізу отриманих економічних показників, можна вважати, що запропонована розробка програмного засобу на основі вдосконаленого методу для покращення захисту ІС з використанням захищеного флеш-носія, двофакторної ідентифікації на основі аналізу клавіатурного почерку та оптимізацією звернень до бази даних має високий комерційний потенціал, а тому є доцільною для подальшого впровадження.

ВИСНОВОК

У магістерській кваліфікаційній роботі здійснюється покращення захисту інформаційних систем з використанням захищеного флеш-носія, двофакторної ідентифікації на основі аналізу клавіатурного почерку та оптимізацією звернень до бази даних на основі генетичного алгоритму.

У першому розділі було здійснено аналіз наукової літератури по даній темі, а саме, досліджено необхідність та цілі захищеного доступу до ІС, проведено аналіз переваг та недоліків існуючих методів захисту від несанкціонованого доступу до ІС, зокрема, з використанням методів авторизації, шифрування, правових та соціальних аспектів. Значна увага під час написання розділу приділялася саме обраним методам контролю доступу до ІС на основі біометричних даних та флеш-носіїв. Було проаналізовано існуючі види біометричної ідентифікації, її особливості, переваги та недоліки. Під час аналізу захищеного доступу на основі електронних ключів було досліджено ключі у вигляді USB-носіїв та смарт-карток, визначено їх сильні та слабкі аспекти, рекомендовано галузі застосування.

У другому розділі роботи проводилося удосконалення захисту ІС на основі захищеного флеш-носія, біометричних даних та оптимізації звернень до БД. Особливістю запропонованого підходу є використання дуального аналізу клавіатурного почерку як унікального біометричного ідентифікатора. Дане рішення було зумовлено необхідністю зменшити можливі неточні результати ідентифікації користувачів під час перевірки такого біометричного параметру через змінену психологічно-емоційного стану користувача та особливості пристроїв, за допомогою яких здійснюється зчитування почерку.

Із метою постійного контролю доступу до ІС автором було запропоновано використання захищеного з використанням алгоритму AES флеш-носія, що використовується як для надання доступу до системи користувачеві, так і для постійного моніторингу роботи користувача в такій ІС.

Для збереження усіх даних та їх надійного зберігання, реалізовано базу

даних. Враховуючи, що при застосування на практиці даного методу в ІС циркулюватимуть значні обсяги інформації, а звернення до БД буде практично безперервним, для оптимізації звернень до БД було запропоновано використання генетичного алгоритму, що реалізується за рахунок виконання таких основних послідовних кроків: ініціалізації, мутації, відбору та заміни.

На основі удосконаленого методу, розроблено алгоритм роботи програмного додатку та здійснено його подальшу програмну реалізацію на мові програмування C# із використанням графічної підсистеми WPF у середовищі програмування Visual Studio. База даних для десктопного програмного додатку розроблена засобами MS SQL.

Для визначення практичної цінності та якості розробки на основі удосконаленого методу було проведено тестування. Перевірка додатку відбувалась на основі дослідження показників FAR (ймовірність помилкового доступу) та FRR (ймовірність помилкової відмови в доступі), перевірці можливості підробки клавіатурного почерку та проводився експеримент ефективності роботи системи при переплутаних між користувачами факторах авторизації. Всі етапи тестування показали позитивні показники та високу якість розробки на основі удосконаленого методу.

Отримані результати економічних показників свідчать, що запропонована розробка програмного засобу на основі вдосконаленого методу для покращення захисту ІС з використанням захищеного флеш-носія, двофакторної ідентифікації на основі аналізу клавіатурного почерку та оптимізацією звернень до бази даних має високий комерційний потенціал, а тому є доцільною для подальшого впровадження.

Таким чином, проаналізувавши отримані результати магістерської кваліфікаційної роботи, можна вважати, що поставлені завдання було виконано в повному обсязі, що уможливило досягнення окресленої мети дослідження.

ПЕРЕЛІК ПОСИЛАНЬ

1. Schneier on Security: : Applied Cryptography. Schneier on Security. URL: https://www.schneier.com/books/applied-cryptography/ (дата звернення: 11.06.2024).

2. Комплексні системи захисту інформації : [навчальний посібник] / Ю. Є. Яремчук, П. В. Павловський, В. С. Катаєв, В. В. Сінюгін. – Вінниця : ВНТУ, 2018. – 118 с.

3. Кібербезпека : сучасні технології захисту. Навчальний посібник для студентів вищих навчальних закладів. / С. Е. Остапов, С. П. Євсеєв, О.Г. Король. – Львів: «Новий Світ- 2000», 2020 . – 678 с.

4. В. В. Карпінець, О. І. Костюченко, П. В. Павловський, А. В. Приймак, С. В. Юхименко. Забезпечення захищеності користувача від несанкціонованого доступу до інформації засобами гібридного гіпервізора *: Оптико-електронні інформаційно-енергетичні технології,* 2018. № 2. С. 34-43. URL: http://nbuv.gov.ua/UJRN/oeiet_2017_2_6 (дата звернення: 11.05.2024).

5. Герт Крюгер. Стандарти інформаційної безпеки – огляд. Dqsglobal. веб-сайт. URL: https://www.dqsglobal.com/uk-ua/navchajtesya/blog/standarti-informacijnoyi-bezpeki-oglyad (дата звернення: 11.05.2024).

6. Кавун С. В. Інформаційна безпека. Навчальний посібник. Ч. 2 / С. В. Кавун, В. В. Носов, О. В. Манжай. – Харків: ХНЕУ, 2008. – 196 с

7. Остапов С. Е. Технології захисту інформації: навчальний посібник / С. Е. Остапов, С. П. Євсеєв, О. Г. Король. – Харків: ХНЕУ, 2013. – 476 с

8. Кузьменко Б. В. Захист інформації / Б. В. Кузьменко, О. А. Чайковська. – Київ: Видавничий відділ КНУКіМ, 2019. – 69 с.

9. Богуш В.М., Богуш В.В., Бровко В.Д., Настрадін В.П. Основи кіберпростору, кібербезпеки та кіберзахисту. Ліра-К – 2021. – 554с.

10. Genetic Algorithm: An Approach on Optimization. IEEE Xplore. URL: https://ieeexplore.ieee.org/document/9002372 (дата звернення: 11.05.2024).

11. Data secrecy and security in computer network based on genetic algorithm. IEEE Xplore. URL: https://ieeexplore.ieee.org/document/9898725 (дата звернення: 11.05.2024).

12. Захист інформації в комп'ютерних системах та мережах : навч. посіб. / С.Г.Семенов, А.О.Подорожняк, О.І.Баленко, С.Ю.Гавриленко – Х.: НТУ «ХПІ», 2014.– 251 с.

13. Захист інформації в комп'ютерних системах: підручник. – Ніжин: ФОП Лук'яненко В.В., ТПК «Орхідея», 2020. – 236с.

14. Хорошко В.О. Комп'ютерна стеганографія / В.О. Хорошко, Ю.Є. Яремчук, В.В. Карпінець – Вінниця: ВНТУ, 2014. – 155 с.

15. Качинський А.Б. Безпека складних систем. К.: ТОВ «Видавництво «Юстон», 2017. 498 с.

16. Азарова А. О., Каращенко В.С., Комп'ютерна програма «Програмний додаток для захищеного доступу до ІС з використанням захищеного флеш-носія, двофакторної ідентифікації на основі аналізу клавіатурного почерку» Свідоцтво про реєстрацію авторського права на твір № 586476. Дата реєстрації 11.06.2024 р. Заявка с67858687.

17. Розроблення пристрою для захисту від несанкціонованого доступу на основі трифакторної ідентифікації та аутентифікації користувачів / А. О. Азарова та ін. *Реєстрація, зберігання і обробка даних*. 2021. Т. 23, № 2. С. 72–80. URL: https://doi.org/10.35681/1560-9189.2021.23.2.239250 (дата звернення: 10.05.2024).

18. What is symmetric encryption – Venafi Blog : веб-сайт. URL: https://www.venafi.com/blog/whatsymmetric-encryption (дата звернення: 11.05.2024).

19. Вимоги із захисту конфіденційної інформації від несанкціонованого доступу під час оброблення в автоматизованих системах класу 2 - *Нормативний документ системи технічного захисту інформації :* веб-сайт. URL: http://www.dsszzi.gov.ua/dsszzi/control/uk/publish/article;jsessionid=

15FDA2B2745B1390AC937214804F2E76?showHidden=1&art_id=101885&cat_id=89734&ctime=1344501165427 (дата звернення: 04.05.2020).

20. What is Two-Factor Authentication (2FA) and How Does It Work? : веб-сайт. URL: https://searchsecurity.techtarget.com/definition/two-factor-authentication (дата звернення: 11.05.2024).

21. All About Passwords – *PBCIT :* веб-сайт. URL: https://www.pbcit.com/all-about-passwords (дата звернення: 11.05.2024).

22. What is Two-Factor Authentication *(2FA) and How Does It Work? :* веб-сайт. URL: https://searchsecurity.techtarget.com/definition/two-factor-authentication (дата звернення: 11.05.2024).

23. Біометричні системи безпеки. Навчальні матеріали: веб-сайт. URL: https://uadoc.zavantag.com/ (дата звернення 11.05.2024).

24. Ivancsy, Renata, and Sandor Juhasz. "Analysis of web user identification methods." International Journal of Computer and Information Engineering 1.10. – 2007. - 3049-3056 p.

25. Біометричні технології в XXI столітті та їх використання правоохоронними органами: посібник / В. П. Захаров, В. І. Рудешко; Львів. держ. ун-т внутр. справ. – 2-ге вид., допов. – Львів: ЛьвДУВС, 2015. – 491 с.

26. Біометричні технології ідентифікації особистості - їх значення і переваги. *Worldvision – інтернет магазин систем безпеки :* веб-сайт. URL: https://worldvision.com.ua/articles/biometricheskie-tehnologii-identifikatsii-lichnosti-ih-znachenie-i-preimushchestva (дата звернення: 11.05.2024).

27. Біометрична ідентифікація – тренд кібербезпеки. *Mirobase: employee management and perfomance solution - Mirobase: employee management and perfomance solution :* веб-сайт. URL: https://mirobase.com/face-id-trend/ (дата звернення: 11.05.2024).

28. Recent Advances in Biometric Systems: A Signal Processing Perspective. *SpringerOpen :* веб-сайт. URL: https://www.springeropen.com/collections/biosys (дата звернення: 11.05.2024).

29. ДСТУ ISO 5807:2016 Обробляння інформації. Символи та угоди щодо документації стосовно даних, програм та системних блок-схем, схем мережевих програм та схем системних ресурсів (ISO 5807:1985).

30. Смарт-картка CryptoCard-338 від ТОВ “Автор”. Портфель. URL: https://portfel.ua/smart-kartka-cryptocard-338-vid-tov-avtor/ (дата звернення: 11.05.2024).

31. USB-ключ Автентифікація без паролів – чи це можливо і як: Google про свої нові проєкти. Highload.today - медіа для розробників. URL: https://highload.today/uk/avtentifikatsiya-bez-paroliv-chi-tse-mozhlivo-i-yak-google-pro-svoyi-novi-proyekti/ (дата звернення: 11.05.2024).

32. Біометричні технології в XXI столітті та їх використання правоохоронними органами: посібник. – 2-ге вид., доп. / В. П. Захаров, В. І. Рудешко. – Львів: ЛьвДУВС, 2015. – 492 с.

33. Jain A.K. 50 Years of Biometric Research: Accom-plishments, Challenges and Opportunities [Text] / Jain A.K., Nandakumar K., Ross A.// Pattern Recognition Letters. – 2016. – №79. – P. 80-105

34. Control Systems - Block Diagrams - Tutorialspoint : веб-сайт. URL: https://www.tutorialspoint.com/control_systems/control_systems_block_diagrams.htm (дата звернення: 11.05.2024).

35. ISO/IEC 19785-1:2006 Information technology. – Common Biometric Exchange Formats Framework – Part 1: Data element specification.

36. Скорик Юлія, Безрук Валерій. Вибір переважного методу біометричної автентифікації. International Science Journal of Engineering & Agriculture Vol. 2, No. 4, 2023, pp. 28-34.

37. Ted Dunstone, Neil Yager. Biometric System and Data Analysis: Design, Evaluation / Dunstone Ted. – Springer, 2008. – 263 p.

38. FAR and FRR: security level versus user convenience. *Recogtech:* веб-сайт. URL: https://www.recogtech.com/en/knowledge-base/security-level-versus-user-convenience (дата звернення 11.05.2024)

39. Системи захисту інформації *(Криптографія) :* веб-сайт. URL: http://pm.fmi.org.ua/files/5b16d283d4fe75.50671571.pdf (дата звернення: 11.05.2024).

40. Еволюція криптографічних алгоритмів шифрування *Isearch:* веб-сайт. URL: http://isearch.kiev.ua/uk/searchpractice/ (дата звернення: 11.05.2024).

41. What is Two-Factor Authentication *(2FA) and How Does It Work?* : веб-сайт. URL: https://searchsecurity.techtarget.com/definition/two-factor-authentication (дата звернення: 11.05.2024).

42. Two-tier & Three-tier Architecture - *Flevian Kanaiza - :* веб-сайт. URL: https://medium.com/@fleviankanaiza/two-tier-three-tier-architecture-8b02536d3482 (дата звернення: 11.05.2024).

43. Станістав М. Two-factor authentication. Ely, Cambridgeshire, United Kingdom : It Governance Publishing, 2015. 114 с.

44. Безклубенко І.С., Гетун Г.В., Баліна О.І., Буценко Ю.П. (2022) Дослідження властивостей множини ефективних значень критеріїв в задачі оптимізації інженерної мережі. Управління розвитком складних систем. № 51, С. 81-86.

45. What is a block diagram? – *Knolwledge Base - microTOOL :* веб-сайт. URL: https://www.microtool.de/en/knowledge-base/what-is-a-block-diagram/ (дата звернення: 11.05.2024).

46. Мова програмування C# Навчальний портал: веб-сайт. URL: http://www.znannya.org/?view=csharp (дата звернення: 11.05.2024).

47. Мовний пакет Microsoft .NET Framework 4.8 (автономний інсталятор) для Windows - Підтримка від Microsoft. Microsoft Support. URL: https://support.microsoft.com/uk-ua/topic/мовний-пакет-microsoft-net-framework-4-8-автономний-інсталятор-для-windows-4bee217e-4096-6922-eba9-3e3c27342ff6 (дата звернення: 11.05.2024).

48. Create a new app with Visual Studio tutorial - WPF .NET. Microsoft Learn: Build skills that open doors in your career. URL:

https://learn.microsoft.com/en-us/dotnet/desktop/wpf/get-started/create-app-visual-studio?view=netdesktop-8.0 (дата звернення: 11.05.2024).

49. Visual Studio *Microsoft:* веб-сайт. URL: https://visualstudio.microsoft.com (дата звернення: 11.05.2024).

50. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт / Уклад. : В. О. Козловський, О. Й. Лесько, В. В. Кавецький. – Вінниця : ВНТУ, 2021. – 42 с.

ДОДАТКИ

Вінницький національний технічний університет
Факультет менеджменту та інформаційної безпеки
Кафедра менеджменту та безпеки інформаційних систем

ЗАТВЕРДЖУЮ
Голова секції «Управління інформаційною безпекою» кафедри МБІС
д.т.н., професор
_____ Юрій ЯРЕМЧУК
«12» березня 2024 р.

ТЕХНІЧНЕ ЗАВДАННЯ

до магістерської кваліфікаційної роботи на тему:

Покращення захисту інформаційних систем з використанням захищеного флеш-носія, двофакторної ідентифікації на основі аналізу клавіатурного почерку та оптимізацією звернень до бази даних на основі генетичного алгоритму
08-72.МКР.005.00.004.ТЗ

Керівник магістерської кваліфікаційної роботи
к.т.н., проф. Азарова А.О. _____

Вінниця – 2024 р.

1. Найменування та область застосування

Покращення захисту інформаційних систем з використанням захищеного флеш-носія, двофакторної ідентифікації на основі аналізу клавіатурного почерку та оптимізацією звернень до бази даних на основі ГА.

2. Підстава для розробки

Розробка виконується на основі наказу ректора ВНТУ №81 від 11.03.2024р.

3. Мета та призначення розробки

3.1 Мета розробки: вдосконалення методу захисту інформаційних систем з використанням захищеного флеш-носія, 2FA на основі аналізу клавіатурного почерку та оптимізацією звернень до бази даних на основі ГА.

3.2 Призначення: розроблений програмний засіб виконує захист інформаційної системи від НСД на основі вдосконаленого методу.

4. Джерела розробки

4.1. Schneier on Security: : Applied Cryptography. Schneier on Security. URL: https://www.schneier.com/books/cryptography/ (дата звернення: 11.06.2024).

4.2. Комплексні системи захисту інформації : [навчальний посібник] / Ю. Є. Яремчук, П. В. Павловський, В. С. Катаєв, В. В. Сінюгін. – Вінниця : ВНТУ, 2018. – 118 с.

4.3. Кібербезпека : сучасні технології захисту. Навчальний посібник для студентів вищих навчальних закладів. / С. Е. Остапов, С. П. Євсеєв, О.Г. Король. – Львів: «Новий Світ- 2000», 2020 . – 678 с.

4.4. Азарова А. О., Каращенко В.С., Комп’ютерна програма «Програмний додаток для захищеного доступу до ІС з використанням захищеного флеш-носія, двофакторної ідентифікації на основі аналізу клавіатурного почерку» Свідоцтво про реєстрацію авторського права на твір № 586476. Дата реєстрації 11.06.2024 р. Заявка с67858687

4.5. Розроблення пристрою для захисту від НСД на основі трифакторної ідентифікації та аутентифікації користувачів / А. О. Азарова та ін. Реєстрація, зберігання і обробка даних. 2021. Т. 23, № 2. С. 72–80.

5. Вимоги до програми

5.1 Вимоги до функціональних характеристик:

5.1.1 Програмний засіб повинен мати зручний, легкий у використанні інтерфейс користувача;

5.1.2 Реалізація методу не повинна вимагати спеціальних ліцензійних програмних додатків;

5.1.3 Програмний засіб повинен виконувати процес автентифікації користувачів у системі.

5.2 Вимоги до надійності:

5.2.1 Програмний засіб повинен працювати без помилок, у випадку виникнення критичних ситуацій необхідно передбачити виведення відповідних повідомлень;

5.2.2 Бази даних повинні бути налаштовані на автоматичне створення резервних копій;

5.2.3 Програмний засіб повинен виконувати свої функції.

5.3 Вимоги до складу і параметрів технічних засобів: процесор – Pentium 1500 МГц і подібні до них; оперативна пам’ять – не менше 512 Mb; середовище функціонування – ОС Windows; вимоги до техніки безпеки при роботі з програмою повинні відповідати існуючим вимогам та стандартам з техніки безпеки при користуванні комп’ютерною технікою.

6. Вимоги до програмної документації

6.1 Обов’язкова поетапна інструкція для майбутніх користувачів, наведена у пункті 3.3.

7. Вимоги до технічного захисту інформації

7.1 Необхідно забезпечити захист відеофайлів від несанкціонованого копіювання.

7.2 Неможливість отримання доступу незареєстрованих користувачів до онлайн-сервісу.

8. Техніко-економічні показники

8.1 Цінність результатів використання даного проекту повинна перевищувати витрати на його реалізацію.

8.2 Має бути реалізований таким чином, щоб підходити для використання широкого загалу.

9. Стадії та етапи розробки

№ з/п	Назва етапів магістерської кваліфікаційної роботи	Початок	Закінчення
1	Визначення напрямку магістерської роботи, формулювання теми	12.03.2024	15.03.2024
2	Аналіз предметної області обраної теми	16.03.2024	27.03.2024
3	Апробація отриманих результатів	28.03.2024	05.04.2024
4	Розробка алгоритму роботи	06.04.2024	20.04.2024
5	Написання магістерської роботи на основі розробленої теми	21.04.2024	25.04.2024
6	Розробка економічної частини	26.04.2024	20.05.2024
7	Передзахист магістерської кваліфікаційної роботи	21.05.2024	31.05.2024
8	Виправлення, уточнення, корегування магістерської кваліфікаційної роботи	10.06.2024	11.06.2024
9	Захист магістерської кваліфікаційної роботи	10.06.2024	11.06.2024

10. Порядок контролю та прийому

10.1 До приймання магістерської кваліфікаційної роботи надається:

- ПЗ до магістерської кваліфікаційної роботи;
- програмний додаток;
- презентація;
- відзив керівника роботи;
- відзив опонента

Технічне завдання до виконання прийняв _____ Карапценко В.С.

Додаток Б. Лістинг Login.xaml

```
using Newtonsoft.Json;
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Configuration;
using System.Diagnostics;
using System.IO;
using System.Linq;
using System.Security.Cryptography;
using System.Text;
using System.Threading;
using System.Threading.Tasks;
using System.Windows;
using usbkeyword.Context;
using Application = System.Windows.Application;
using MessageBox = System.Windows.MessageBox;
using OpenFileDialog = Microsoft.Win32.OpenFileDialog;
using Path = System.IO.Path;
using Timer = System.Timers.Timer;
using User = usbkeyword.Models.User;
namespace usbkeyword
{
   public partial class Login : Window, INotifyPropertyChanged
   {
      private long Time = 0;
      private string Text = "";
      private bool drawEnable = false;
      private RecognizeContext _context = new RecognizeContext();
      private List<double> intervals;
      private double lastTime;
      private CancellationTokenSource _cancellationTokenSource;
      private string _filePath;
      public Login()
      {
         InitializeComponent();
         stopwatch = new Stopwatch();
         intervals = new List<double>();
              }
      private int failCount = 3;
      public event PropertyChangedEventHandler PropertyChanged;
      private bool active = true;
      private bool enableAutoNavigate = true;
      private Timer captureTimer;
      Stopwatch stopwatch = new Stopwatch();
      public void OnLoad()
      {
```

```
        }
        private void Window_Loaded(object sender, RoutedEventArgs e)
        {
            OnLoad();
        }
        private void Button_Click_1(object sender, RoutedEventArgs e)
        {
            if (!VerifyData())
                return;
            var res = GetUsers();
            var eror = "";
            if (res == null)
            {
                eror = "\"Користувача не знайдено\"";
            }
            else
            {
                var f = lblFaceName.Content.ToString().ToLower();

                var user = res.FirstOrDefault(x => x.UserName == UserName.Text);
                if (user == null)
                {
                    eror = "\"Користувача не знайдено\"";
                }
                else
                {
                    if (user.IsAdmin || (user.UserId == Decript(FileName.Text) && user.Password ==
Password.Text) && AnalyzeButton_Click(user))
                    {
                        this.Hide();
                        Helper.AddLog("User is logged in", user.UserName);
                        CommonPage commonpage = new CommonPage();
                        commonpage.AuthUserName.Content = user.UserName;
                        if (user.IsAdmin || user.Role == "Адміністратор")
                        {
                            commonpage.UserTable.Visibility = Visibility.Visible;
                        }
                        else
                        {
                            commonpage.UserTable.Visibility = Visibility.Hidden;
                        }
                        StartFileCheck(FileName.Text);
                        commonpage.Registration.Visibility = commonpage.Login.Visibility =
Visibility.Hidden;
                        commonpage.Exit.Visibility = Visibility.Visible;
                        commonpage.UserInfo.Visibility = Visibility.Visible;
                        commonpage.history.Visibility = Visibility.Visible;
                        enableAutoNavigate = false;
                        MessageBox.Show("Спроба входу була вдалою!");
```

```
                    commonpage.ShowDialog();
                    this.Close();
                }
                else
                {
                    if (user.UserId != Decript(FileName.Text))
                    {
                        eror = $"Невірний ключ";
                    }
                    else if (user.Password != Password.Text)
                    {
                        eror = $"Невірний пароль";
                    }
                    else
                    {
                        eror = $"Помилка входу";
                    }
                }
            }
            if (!string.IsNullOrEmpty(eror))
            {
                --failCount;
                MessageBox.Show($"Невдала спроба ідентифікації. {eror}. Залишилося спроб -
{failCount}");

                if (failCount == 0)
                {
                    this.Hide();
                    Environment.Exit(1);
                }
                else
                {
                    return;
                }
            }
        }

        private void StartFileCheck(string filePath)
        {
            _cancellationTokenSource = new CancellationTokenSource();
            Task.Run(() => CheckFileExistence(filePath, _cancellationTokenSource.Token));
        }
        private async Task CheckFileExistence(string filePath, CancellationToken token)
        {
            while (!token.IsCancellationRequested)
            {
                if (!File.Exists(filePath))
                {
```

```
                    Application.Current.Dispatcher.Invoke(() =>
                    {
                        Application.Current.Shutdown();
                    });
                    break;
                }
                await Task.Delay(5000, token);
            }
        }
        private bool AnalyzeButton_Click(User user)
        {
            double averageInterval = 0;
            if (intervals.Count > 0)
            {
                averageInterval = CalculateAverage(intervals);
            }
            if (user.AverageInterval - 300 < averageInterval && averageInterval < user.AverageInterval + 300)
                return true;
            return false;
        }
        private double CalculateAverage(List<double> values) // біометрія
        {
            double sum = 0;
            foreach (var value in values)
            {
                sum += value;
            }
            return sum / values.Count;
        }
        private static string Decript(string path)
        {
            string encryptedResult = null;
            using (StreamReader r = new StreamReader(path))
            {
                encryptedResult = r.ReadToEnd();
            }
            byte[] bytesToBeDecrypted = Convert.FromBase64String(encryptedResult);
            byte[] passwordBytesdecrypt = Encoding.UTF8.GetBytes("Password");
            passwordBytesdecrypt = SHA256.Create().ComputeHash(passwordBytesdecrypt);
            byte[] bytesDecrypted = AES_Decrypt(bytesToBeDecrypted, passwordBytesdecrypt);
            return Encoding.UTF8.GetString(bytesDecrypted);
        }
        private static byte[] AES_Decrypt(byte[] bytesToBeDecrypted, byte[] passwordBytes)
        {
            byte[] decryptedBytes = null;
            byte[] saltBytes = new byte[] { 2, 1, 7, 3, 6, 4, 8, 5 };
            using (MemoryStream ms = new MemoryStream())
            {
```

```
                using (RijndaelManaged AES = new RijndaelManaged())
                {
                    AES.KeySize = 256;
                    AES.BlockSize = 128;
                    var key = new Rfc2898DeriveBytes(passwordBytes, saltBytes, 1000);
                    AES.Key = key.GetBytes(AES.KeySize / 8);
                    AES.IV = key.GetBytes(AES.BlockSize / 8);
                    AES.Mode = CipherMode.CBC;
                    using (var cs = new CryptoStream(ms, AES.CreateDecryptor(),
CryptoStreamMode.Write))
                    {
                        cs.Write(bytesToBeDecrypted, 0, bytesToBeDecrypted.Length);
                        cs.Close();
                    }
                    decryptedBytes = ms.ToArray();
                }
            }
            return decryptedBytes;
        }
        private bool VerifyData()
        {
            if (String.IsNullOrEmpty(UserName.Text))
            {
                MessageBox.Show("Введіть логін");
                return false;
            }
            if (String.IsNullOrEmpty(FileName.Text))
            {
                MessageBox.Show("Введіть ключ");
                return false;
            }
            if (String.IsNullOrEmpty(Password.Text))
            {
                MessageBox.Show("Введіть пароль");
                return false;
            }
            return true;
        }
        private void newAccount_Click(object sender, RoutedEventArgs e)
        {
            active = false;
            var w = Application.Current.Windows[0];
            this.Hide();
            Registration wfAbout = new Registration();
            wfAbout.ShowDialog();
            this.Close();
        }
        private List<User> GetUsers()
        {
```

```
            var useFile = Convert.ToBoolean(ConfigurationManager.AppSettings["UsFile"]);
            var users = new List<User>();
            if (useFile == true)
            {
                var path = $"{Path.GetDirectoryName(AppDomain.CurrentDomain.BaseDirectory)}/ResourceFile.json";
                var exists = File.Exists(path);
                if (!exists)
                    File.Create(path);
                using (var r = new StreamReader(path))
                {
                    var json = r.ReadToEnd();
                    users = JsonConvert.DeserializeObject<List<User>>(json);
                }
            }
            else
            {
                users = _context.Users.ToList();
            }
            return users;
        }
        private void Login_OnClosed(object sender, EventArgs e)
        {
            captureTimer = null;
            active = false;
            if (enableAutoNavigate)
            {
                var w = Application.Current.Windows[0];
                this.Hide();
                CommonPage wfAbout = new CommonPage();
                wfAbout.ShowDialog();
                this.Close();
            }
        }
        private void Button_Click(object sender, RoutedEventArgs e)
        {
            OpenFileDialog dlg = new OpenFileDialog();
            dlg.RestoreDirectory = true;

            if (dlg.ShowDialog() == true)
            {
                string selectedFileName = dlg.FileName;
                FileName.Text = selectedFileName;
            }
        }
        private void Button_Click_2(object sender, RoutedEventArgs e)
        {
            if (!drawEnable)
            {
```

```
                text_keyword.Text = "";
                drawEnable = true;
                StaerRecog.Content = "Аналіз розпочато, зупинити?";
                stopwatch.Start();
                text_keyword.IsEnabled = true;
                lable_text.Text = "Розробка програмного забезпечення вимагає ретельного планування та тестування на кожному етапі";
            }
            else
            {
                stopwatch.Stop();
                drawEnable = false;
                text_keyword.IsEnabled = true;
                Time = stopwatch.ElapsedMilliseconds;
                Text = text_keyword.Text;
                StaerRecog.Content = "Розпочати знову?";
                MessageBox.Show("Аналіз завершено та дані збережено!");
            }
        }
        private void text_keyword_TextChanged(object sender, System.Windows.Controls.TextChangedEventArgs e)
        {
            if (!stopwatch.IsRunning)
            {
                stopwatch.Start();
                lastTime = stopwatch.ElapsedMilliseconds;
            }
            else
            {
                double currentTime = stopwatch.ElapsedMilliseconds;
                double interval = currentTime - lastTime;
                if (interval > 0) // Додаємо лише валідні інтервали
                {
                    intervals.Add(interval);
                }
                lastTime = currentTime;
            }
        }
    }
}
```

Додаток В. Лістинг UserTable.xaml

```
using Newtonsoft.Json;
using System;
using System.Collections.Generic;
using System.Configuration;
using System.IO;
using System.Linq;
using System.Text;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Media;
using usbkeyword.Context;
using usbkeyword.Models;

namespace usbkeyword
{
   public partial class UserTable : Window
   {
      private RecognizeContext _context = new RecognizeContext();
      public UserTable()
      {
         InitializeComponent();
      }

      private void Window_Loaded(object sender, RoutedEventArgs e)
      {
         ShowUser();
      }
      public void ShowUser()
      {
         var users = GetUsers();
         if (users == null)
            return;
         var userList = new List<UserForTable>();
         var roles = GetRoless()
            ;
         foreach (var user in users)
         {
            dataGridView.Items.Add(new UserForTable()
            {
               Name = user.UserName,
               Text = user.FIO,
               Email = user.Email,
               Role = user.Role
            });
         }
      }
```

```
        private List<Role> GetRoless()
        {
            var useFile = Convert.ToBoolean(ConfigurationManager.AppSettings["UsFile"]);
            var roles = new List<Role>();
            if (useFile == false)
            {
                roles = _context.Roles.ToList();
            }
            return roles;
        }
        void ShowHideDetails(object sender, RoutedEventArgs e)
        {
            EditUser wfAbout = new EditUser();
            for (var vis = sender as Visual; vis != null; vis = VisualTreeHelper.GetParent(vis) as
Visual)
            {
                if (vis is DataGridRow)
                {
                    var row = (DataGridRow)vis;
                    if (row != null)
                    {
                        var uiUser = row.DataContext;
                        var users = GetUsers();
                        if (users == null)
                            return;
                        var name = uiUser.GetType().GetProperty("Name").GetValue(uiUser, null);
                        var user = users.Find(x => x.UserName == name.ToString());
                        wfAbout.Email.Text = user.Email;
                        wfAbout.UserName.Text = user.UserName;
                        wfAbout.FIO.Text = user.FIO;
                    }
                    wfAbout.ShowDialog();
                    this.Close();
                }
                this.Close();
            }
            this.Close();
        }
        void DeleteUser(object sender, RoutedEventArgs e)
        {
            for (var vis = sender as Visual; vis != null; vis = VisualTreeHelper.GetParent(vis) as
Visual)
                if (vis is DataGridRow)
                {
                    var row = (DataGridRow)vis;


                    if (row != null)
```

```
                {
                    var uiUser = row.DataContext;
                    var users = GetUsers();
                    if (users == null)
                        return;
                    var name = uiUser.GetType().GetProperty("Name").GetValue(uiUser, null);
                    var deleteUser = users.Find(x => x.UserName == name.ToString());
                    users.Remove(deleteUser);
                    string json = JsonConvert.SerializeObject(users);
                    File.WriteAllText(

$"{Path.GetDirectoryName(System.AppDomain.CurrentDomain.BaseDirectory)}/ResourceFile.json
",
                        json, Encoding.UTF8);
                }
                else
                {
                    return;
                }
                dataGridView.Items.Clear();
            }
            ShowUser();
            MessageBox.Show("Користувача було успішно видалено");
        }
        public class UserForTable
        {
            public string Name { get; set; }
            public string Text { get; set; }
            public string Email { get; set; }
            public string Role { get; set; }
        }
        private List<User> GetUsers()
        {
            var useFile = Convert.ToBoolean(ConfigurationManager.AppSettings["UsFile"]);
            var users = new List<User>();
            if (useFile == true)
            {
                var                                    path                                    =
$"{Path.GetDirectoryName(AppDomain.CurrentDomain.BaseDirectory)}/ResourceFile.json";
                var exists = File.Exists(path);
                if (!exists)
                    File.Create(path);
                using (var r = new StreamReader(path))
                {
                    var json = r.ReadToEnd();
                    users = JsonConvert.DeserializeObject<List<User>>(json);
                }
            }
            else
```

```
            {
                users = _context.Users.ToList();
            }
            return users;
        }
        private void Button_Click(object sender, RoutedEventArgs e)
        {
            EditUser commonpage = new EditUser();
            commonpage.ShowDialog();
        }
        private void Button_Click_1(object sender, RoutedEventArgs e)
        {
            this.Close();
        }
        private void dataGridView_SelectionChanged(object sender,
SelectionChangedEventArgs e)
        {
        }
    }
}
```

Додаток Г. Інтерфейс програмного додатку

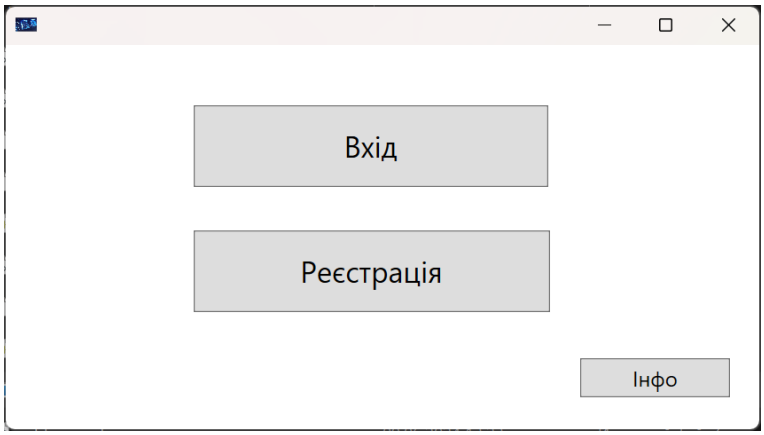


Рисунок 1 – Вигляд початкового вікна програмного додатку

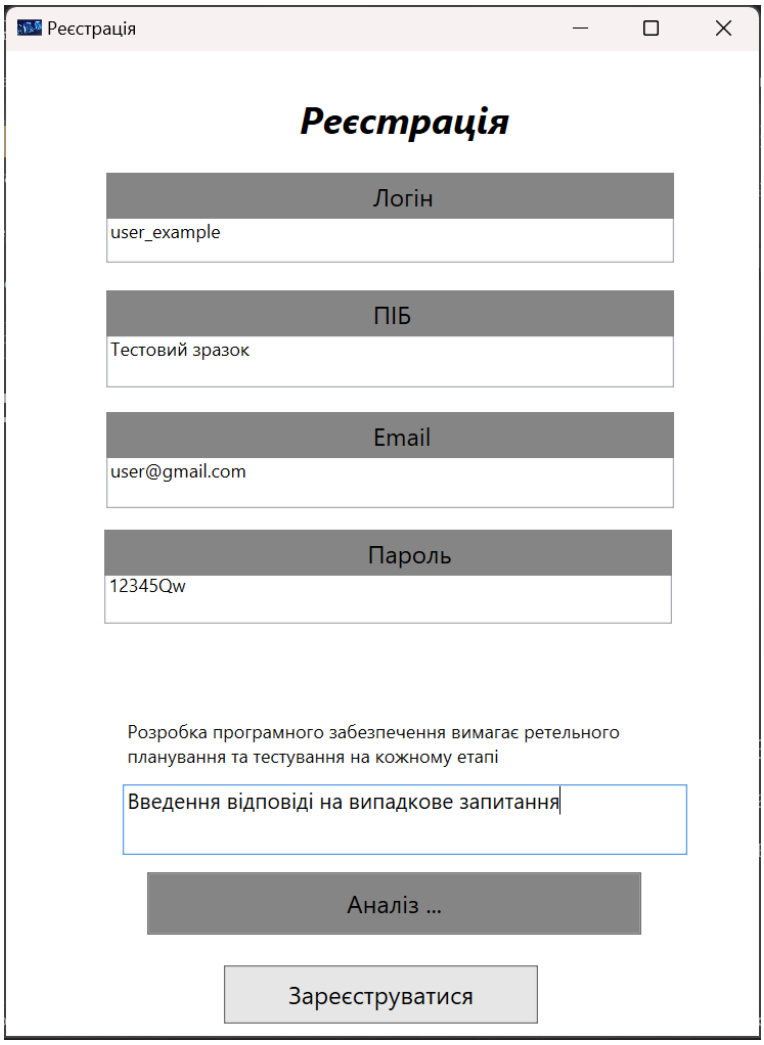


Рисунок 2 – Вигляд вікна форми «Реєстрація»

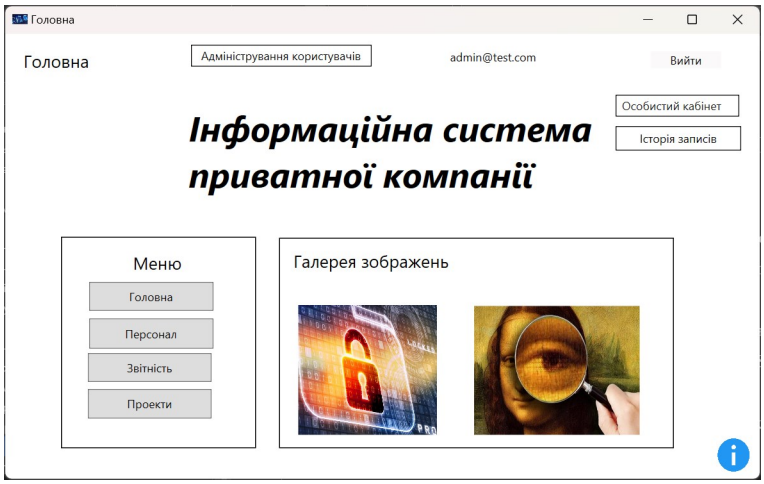


Рисунок 3 – Вигляд головного вікна системи для облікового запису адміністратора

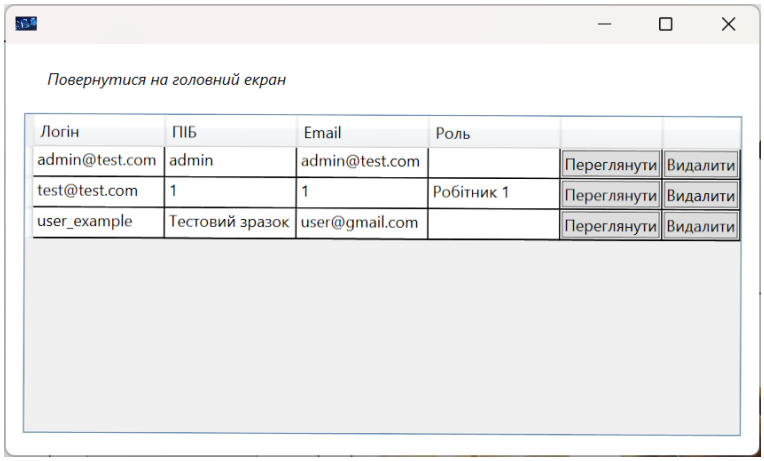


Рисунок 4 – Вигляд вікна адміністрування користувачів

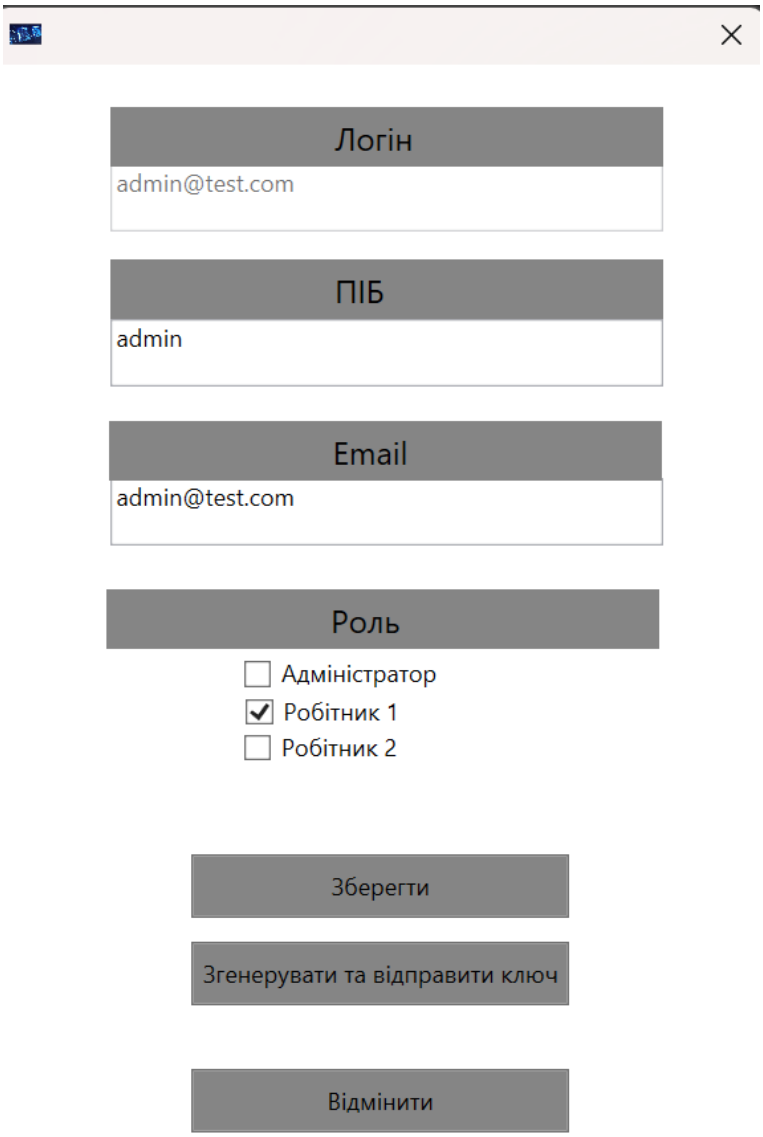


Рисунок 5 – Вигляд вікна редагування даних користувача

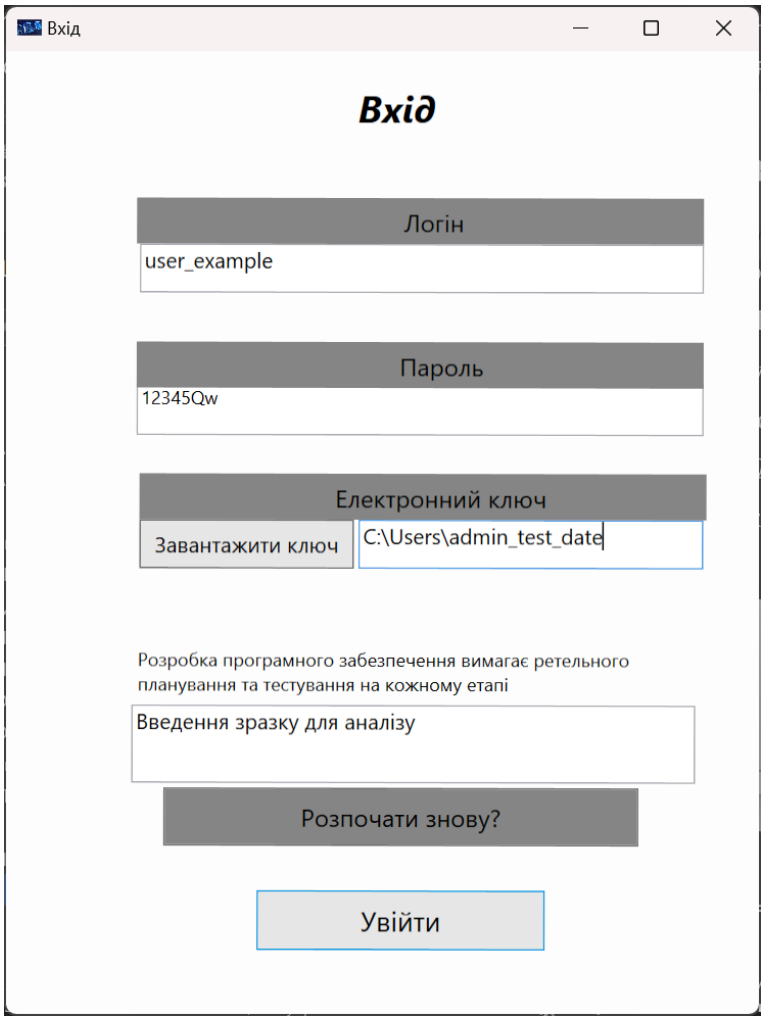


Рисунок 6 – Вигляд вікна форми авторизації користувача

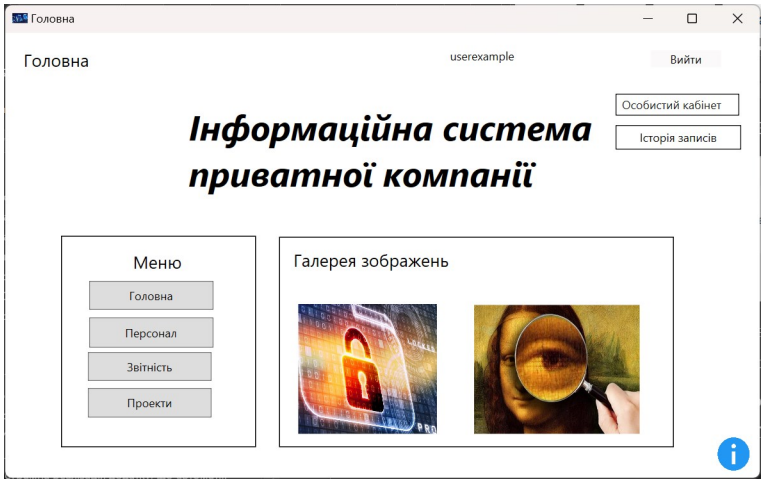


Рисунок 7 – Вигляд головного вікна системи для облікового запису користувача

Додаток Д. Ілюстративний матеріал

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА
на тему:

Покращення захисту інформаційних систем з використанням захищеного флеш-носія, двофакторної ідентифікації на основі аналізу клавіатурного почерку та оптимізацією звернень до бази даних на основі генетичного алгоритму

Виконав: ст. 2-го курсу, групи КІТС-22мз Каращенко В. С.

Керівник: к. т. н., проф.; професор каф. МБІС Азарова А. О.

Мета, задачі та актуальність роботи

Метою роботи є вдосконалення методу захисту інформаційних систем з використанням захищеного флеш-носія, двофакторної ідентифікації на основі аналізу клавіатурного почерку та оптимізацією звернень до бази даних на основі генетичного алгоритму.

Для досягнення такої мети було поставлено та вирішено такі задачі:

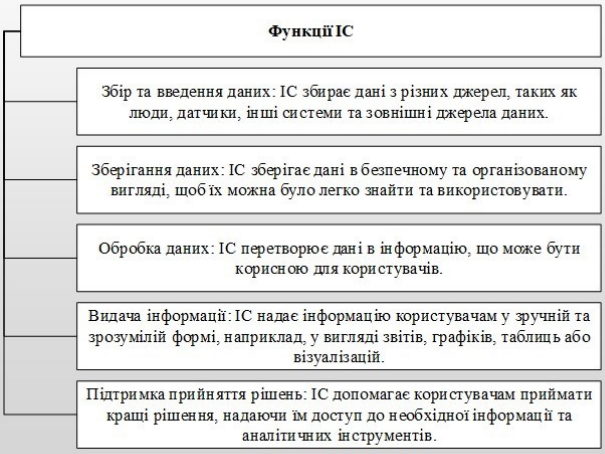
- проаналізовано існуючі інструменти захисту ІС від НСД та досліджено можливість їх застосування для захисту ІС;
- удосконалено метод захисту ІСвід НСД шляхом застосування двофакторної ідентифікації;
- на базі складеного підходу запропоновано алгоритм роботи відповідного програмного засобу та здійснено його реалізацію;
- протестовано роботу ПЗ та проаналізовано отримані результати;
- обґрунтовано економічну доцільність впровадження запропонованої розробки.

Практична цінність: розроблено програмний засіб для захисту доступу до інформаційних систем із використанням захищеного флеш-носія, біометричного ідентифікатора та оптимізації звернень до бази даних на основі генетичного алгоритму.

Публікації. Результати роботи було захищено шляхом отримання свідоцтва на реєстрацію авторського права на твір.

Роль та значення захищеного доступу до ІС

- **Інформаційна система** – це сукупність організаційних та технічних засобів для збереження та оброблення інформації з метою забезпечення інформаційних потреб користувачів.
- **Функції,** які виконує інформаційна система наведено на рисунку.
- Під поняттям **несанкціонованого доступу до ІС** слід розуміти отримання доступу до даних, програмного забезпечення або апаратних компонентів ІС без дозволу або права власника [5].



Аналіз переваг та недоліків існуючих методів захисту ІС від НСД

Тип методу	Метод	Переваги	Недоліки
Програмні	Шифрування	Високий рівень безпеки	Потребує багато ресурсів Складний у реалізації
	Брандмауери	Захист від зовнішніх загроз	Не захищають від внутрішніх загроз
	Антивірусне ПЗ	Виявляє та видаляє шкідливе ПЗ	Не може виявити всі типи загроз
	IDS/IPS	Виявляє та блокує підозрілу активність	Можуть генерувати багато помилок
Організаційний	Контроль доступу	Обмежує доступ до ресурсів	Може бути складним для адміністрування
	Політика безпеки інформації	Визначає правила та процедури	Не гарантує дотримання правил
	Навчання персоналу	Підвищує обізнаність про ризики	Не гарантує безпечної поведінки
	Резервне копіювання та відновлення	Відновлення даних у разі збою	Не захищає від НСД
	Регулярний аудит	Виявлення вразливостей	Не гарантує постійний захист

Удосконалення методу захищеного доступу до ІС

Суть вдосконалення полягає у комплексному підході до процесу авторизації користувача з використанням зовнішнього носія інформації та унікальних біометричних даних.

- Застосування даного підходу захищеного доступу до інформаційних систем **має низку переваг.**

Переваги застосування 2FA для контролю доступу

- Підвищена безпека (2FA додає другий рівень захисту до процесу входу в систему, ускладнюючи НСД. Навіть якщо зловмиснику вдасться отримати пароль користувача, він не зможе увійти в систему без додаткового фактора аутентифікації)
- Захист від фішингу та атак типу "людина посередині" (2FA ефективно протистоїть фішинговим атакам, де зловмисники намагаються схилити користувачів до розкриття своїх паролів)
- Захист від брутфорс-атак (2FA ускладнює брутфорс-атаки, де зловмисники використовують програмне забезпечення для перебору всіх можливих паролів, навіть якщо зловмисник зможе підібрати пароль, він все одно не зможе увійти в систему без додаткового фактора аутентифікації)
- Захист конфіденційних даних (2FA захищає доступ до конфіденційних даних, додатковий фактор аутентифікації робить доступ до цих даних більш складним для зловмисників)
- Зручність використання (сучасні методи 2FA, такі як одноразові паролі (OTP), аутентифікатори та біометричні дані, прості у використанні для більшості користувачів, їх можна легко налаштувати та використовувати на різних пристроях)
- Підвищення довіри користувачів (2FA збільшує довіру користувачів до інформаційної системи та організації, яка її використовує. це демонструє, що організація серйозно ставиться до кібербезпеки та захисту даних своїх користувачів)

Структурна модель удосконаленого методу

Удосконалений метод комплексної перевірки користувачів дозволяє забезпечити якісний процес авторизації. Недоліком клавіатурного почерку може бути зміна поведінки користувача залежно від того відома чи невідома йому фраза, зміна типу клавіатури і т.д.

- Проте, за рахунок того, що в базі даних зберігається декілька еталонних зразків почерку користувача, а особливості його друку перевіряються як для відомих так і не відомих заздалегідь фраз, **забезпечується максимальна точність аналізу.**

Побудова підходу до використання флеш-носія для реалізації захищеного доступу

Ключові процедури алгоритму роботи із флеш-носієм

- Аутентифікація (генерація певної послідовності та її запис на флеш-носій, послідовність унікальна для кожного користувача)
- Шифрування (всі дані на флеш-носії шифруються за допомогою стійкого алгоритму шифрування AES)
- Авторизація (перевірка прав доступу – після успішної аутентифікації система перевіряє, чи має користувач право доступу до інформаційної системи)

○ **Ключові процедури роботи із флеш-носієм**

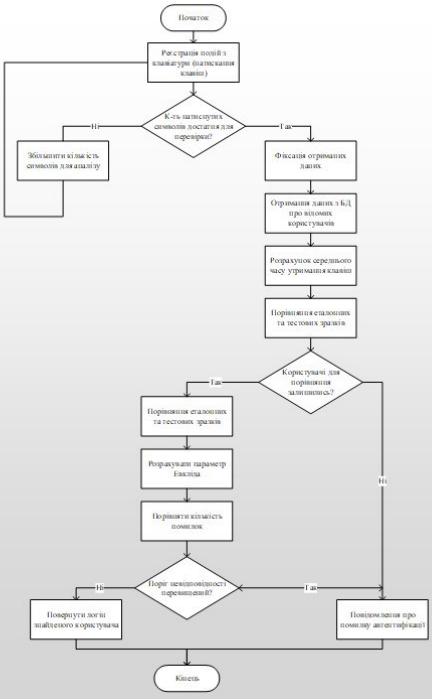
Побудова дуального підходу до перевірки унікальних біометричних даних користувача

$$\overline{pause} = \frac{\sum_{i-1}^{n} pause_i}{n} \qquad errors = delete + backspece.$$

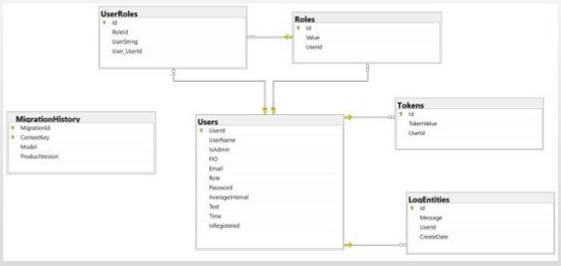
$$P = \sqrt{\sum_{i=1}^{N} (A_i - B_i)^2} \qquad P = \sum_{i=1}^{N} |A_i - B_i|$$

$$P = \sqrt{(\vec{A} - \vec{B})^T S^{-1} (\vec{A} - \vec{B})}$$

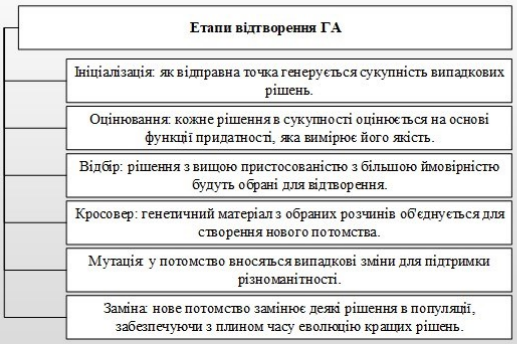
Алгоритм розпізнавання користувача за клавіатурним почерком



Оптимізація процесу звернення до БД засобами генетичного алгоритму

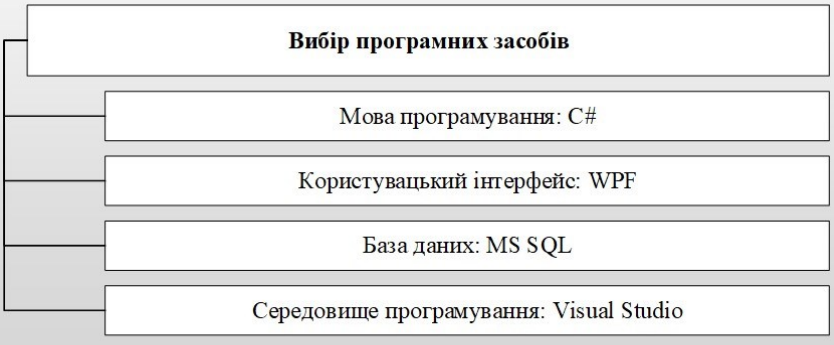


Діаграма БД для програмного додатку



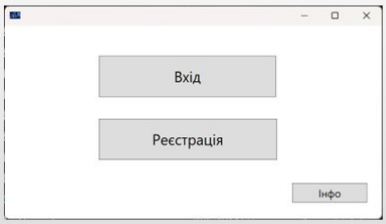
Етапи відтворення типового генетичного алгоритму

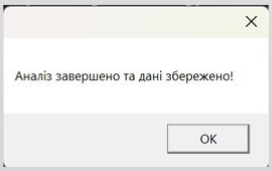
Обґрунтування засобів програмування для автоматизації удосконаленого методу захисту

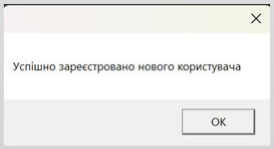


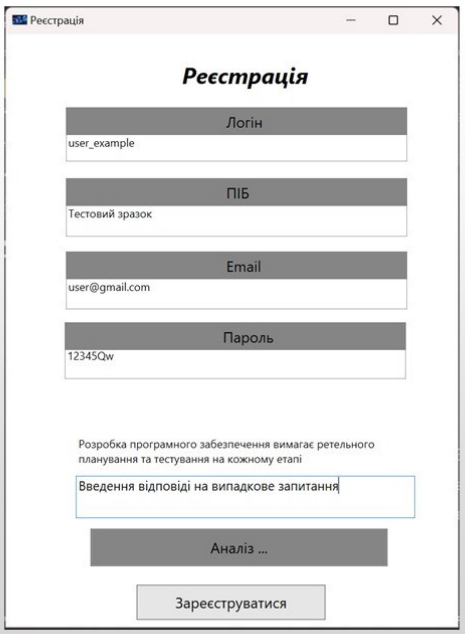
Програмна реалізація

- Процес реєстрації нового користувача



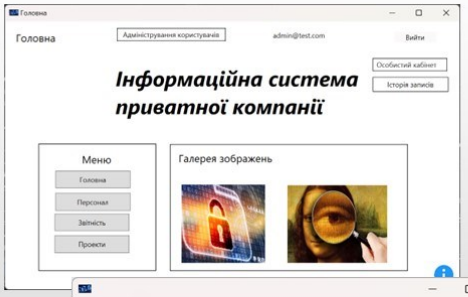


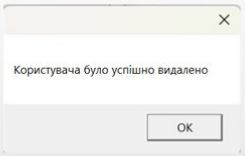


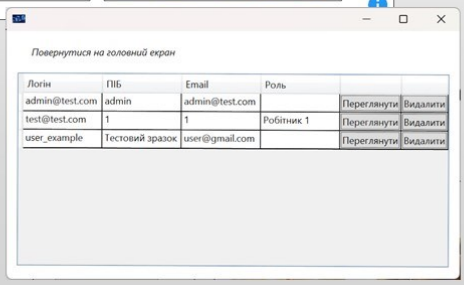


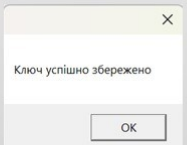
Програмна реалізація

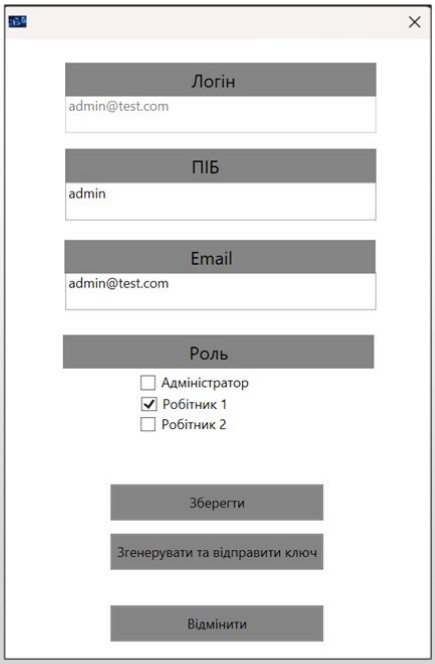
- Обліковий запис адміністратора





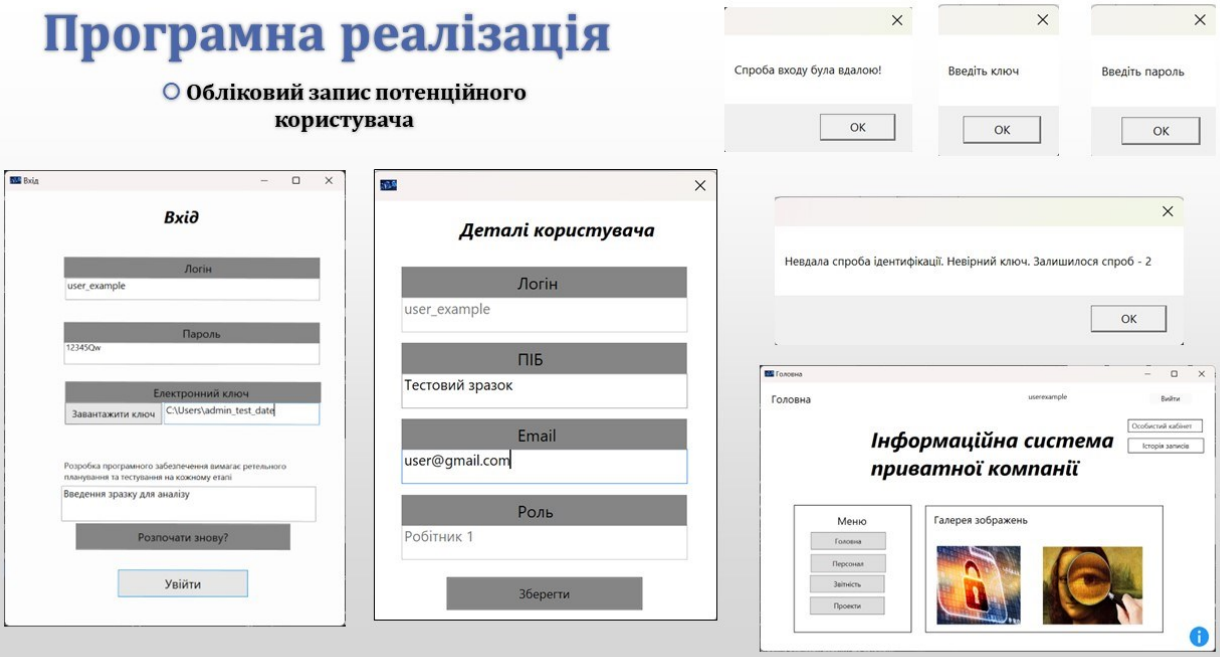






Програмна реалізація

- Обліковий запис потенційного користувача



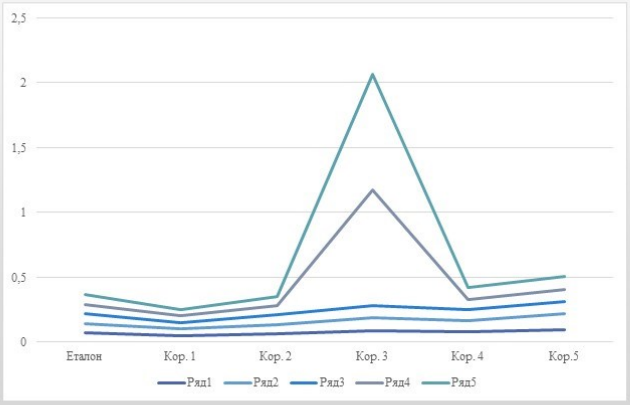
Аналіз тестування ПЗ та коригування підходу до захисту інформаційної системи

$$FAR = \frac{NFA}{NIVA} \cdot 100\% \qquad FRR = \frac{NFF}{NEVA} \cdot 100\%$$

Метод	Корист. 1		Корист. 2		Корист. 3		Корист. 4		Корист. 5	
	FAR, %	FRR, %	FAR, %	FRR, %	FAR, %	FRR, %	FAR, %	FRR, %	FAR, %	FRR, %
Дуальний метод	0	20	0	0	0	20	0	0	0	0
На основі відомого паролю	20	20	0	20	0	0	0	0	40	0
На основі невідомої фрази	0	20	20	0	0	0	0	0	20	20

Метод	Корист. 6		Корист. 7		Корист. 7		Корист. 9		Корист. 10	
	FAR, %	FRR, %	FAR, %	FRR, %	FAR, %	FRR, %	FAR, %	FRR, %	FAR, %	FRR, %
Дуальний метод	0	20	0	0	0	20	0	0	0	0
На основі відомого паролю	0	60	0	40	20	20	0	20	20	60
На основі невідомої фрази	0	40	20	0	20	0	0	0	0	20

Аналіз тестування ПЗ та коригування підходу до захисту інформаційної системи



Графік для тесту 2

Тест 3. Для перевірки практичності застосування флеш-носія з метою контролю доступу до ІС було **проведено експериментальне дослідження:** створено десять користувачів, згенеровано для них ключі та зафіксовано еталони клавіатурного почерку. У тестовому режимі **здійснювалися спроби авторизації користувачів за наявними в системі зразками клавіатурного почерку та згенерованими ключами.** Результатом дослідження є **100% відмова в доступі усім користувачам з переплутаними даними,** проте наявними в системі.

Висновки

Дякую за увагу!

- У МКР здійснюється **покращення захисту ІС з використанням захищеного флеш-носія, двофакторної ідентифікації на основі аналізу клавіатурного почерку та оптимізацією звернень до бази даних на основі генетичного алгоритму.**
- Особливістю запропонованого підходу **є використання дуального аналізу клавіатурного почерку як унікального біометричного ідентифікатора.** Дане рішення було зумовлено необхідністю **зменшити можливі неточні результати ідентифікації користувачів під час перевірки** такого біометричного параметру через змінену психологічно-емоційного стану користувача та особливості пристроїв, за допомогою яких здійснюється зчитування почерку.
- Для збереження усіх даних та їх надійного зберігання, реалізовано базу даних. Враховуючи, що при застосування на практиці даного методу в ІС циркулюватимуть значні обсяги інформації, а звернення до БД буде практично безперервним, для оптимізації звернень до БД було **запропоновано використання генетичного алгоритму.**
- Для визначення практичної цінності та якості розробки на основі удосконаленого методу було проведено тестування. Всі етапи тестування **показали позитивні показники та високу якість розробки** на основі удосконаленого методу.
- Отримані результати економічних показників свідчать, що **запропонована розробка програмного засобу має високий комерційний потенціал, а тому є доцільною для подальшого впровадження.**

Додаток Е. Протокол перевірки на антиплагіат

ПРОТОКОЛ ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ НА НАЯВНІСТЬ ТЕКСТОВИХ ЗАПОЗИЧЕНЬ

Назва роботи:
Покращення захисту інформаційних систем з використанням захищеного флешносія, двофакторної ідентифікації на основі аналізу клавіатурного почерку та оптимізацією звернень до бази даних на основі генетичного алгоритму
Тип роботи: магістерська кваліфікаційна робота
(БДР, МКР)

Підрозділ: Кафедра менеджменту та безпеки інформаційних систем
Факультет менеджменту та інформаційної безпеки
(кафедра, факультет)

Показники звіту подібності Unicheck

Оригінальність 93% Схожість 7%

Аналіз звіту подібності (відмітити потрібне):

1. **Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.**
2. Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її виконання автором. Роботу направити на розгляд експертної комісії кафедри.
3. Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Особа, відповідальна за перевірку _____ Коваль Н.П.
(підпис) (прізвище, ініціали)

Ознайомлені з повним звітом подібності, який був згенерований системою Unicheck щодо роботи.

Автор роботи _____ Каращенко В.С.
(підпис) (прізвище, ініціали)

Керівник роботи _____ Азарова А.О.
(підпис) (прізвище, ініціали)