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

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

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
за першим (бакалаврським) рівнем вищої освіти
на тему
“Програмне забезпечення системи мережевого відеонагляду газового родовища на основі обладнання Axis”

Виконав здобувач вищої освіти
IV курсу, групи КІ-22-1
ОПП «Комп’ютерна інженерія»
спеціальності 123 «Комп’ютерна інженерія»
_____ Бочко К.К.
«____» _____ 2026 р.

Керівник проекту
доктор філософії (PhD)
_____ Усік П.С.
«____» _____ 2026 р.
Рецензент _____
_____

м. Кропивницький

Центральноукраїнський національний технічний університет

Факультет *Механіко-технологічний*

Кафедра *Кібербезпеки та програмного забезпечення*

Освітній ступінь *бакалавр*

Галузь знань _12 _____*"Інформаційні технології"*_____

Спеціальність 123 *"Комп'ютерна інженерія"*

Освітньо-професійна (освітньо-наукова) програма _*"Комп'ютерна інженерія"*_

ЗАТВЕРДЖУЮ
Завідувач кафедри
д.т.н., проф.
_____ Олексій СМІРНОВ
« 27 » лютого 2026 року

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

Бочко Костянтину Костянтиновичу
(прізвище, ім'я, по батькові)

1. Тема роботи *Програмне забезпечення системи мережевого відеонагляду газового родовища на основі обладнання Axis*

2. Керівник роботи *Усік Павло Сергійович, доктор філософії (PhD)*
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу № 133-02 від 27.02.2026 року

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

4. Мета та завдання випускної кваліфікаційної роботи*: Метою роботи є розробка програмного забезпечення системи мережевого відеонагляду газового родовища на основі обладнання Axis*

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

1. Призначення та область використання.
2.Перегляд аналогічних існуючих систем.
3. Опис і обґрунтування проектних рішень.
4. Етапи програмування системи.
5. Впровадження системи в промислову експлуатацію.
6. Висновки

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

Структурна схема системи	*1 аркуш*
Функціональна схема системи	*1 аркуш*
Діаграма процесів	*1 аркуш*
Блок-схема алгоритму роботи додатку	*2 аркуша*

7. Дата видачі завдання « 27 » лютого 2026 р.

КАЛЕНДАРНИЙ ПЛАН

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

Дата видачі завдання
« 27 » лютого 2026 р.

Підпис керівника
_____ Усік П.С.
(прізвище та ініціали)

Завдання прийнято до виконання
« 27 » лютого 2026 р.

Підпис здобувача
_____ Бочко К.К.
(прізвище та ініціали)

АНОТАЦІЯ

Бочко К.К. Програмне забезпечення системи мережевого відеонагляду газового родовища на основі обладнання Axis. 123 Комп'ютерна інженерія. Центральноукраїнський національний технічний університет. Кропивницький. 2026.

В даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи мережевого відеонагляду газового родовища на основі обладнання Axis.

Метою розробки є програмне забезпечення системи мережевого відеонагляду газового родовища на основі обладнання Axis.

Результат роботи – програмна реалізація системи мережевого відеонагляду газового родовища на основі обладнання Axis.

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

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

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

Програму розроблено в середовищі Builder C++.

Ключові слова: комп'ютерна інженерія, відеонагляд, Axis

ABSTRACT

Bochko K.K. Software for a network video surveillance system for a gas field based on Axis equipment. 123 Computer Engineering. Central Ukrainian National Technical University. Kropyvnytskyi. 2026.

In this final qualification work for the first (bachelor's) level of higher education, software has been developed that is intended for a network video surveillance system for a gas field based on Axis equipment.

The purpose of the development is software for a network video surveillance system for a gas field based on Axis equipment.

The result of the work is a software implementation of a network video surveillance system for a gas field based on Axis equipment.

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

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

The program can be used on a PC with Windows 10/11 OS.

The program was developed in the Builder C++ environment.

Keywords: computer engineering, video surveillance, Axis

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

БПД	–	бездротова передача даних
ПЗ	–	програмне забезпечення
СПД	–	системи передачі даних
ACK	–	повідомлення підтвердження прийому
ARQ	–	протокол повторної передачі даних
BPSK	–	Binary phase-shift keying
FFD	–	повнофункціональний пристрій
GFSK	–	Gaussian frequency-shift keying
MAC	–	шар механізму доступу
NACK	–	повідомлення непідтвердження прийому
OSI	–	мережна модель
P2P	–	однорангові мережі
PAN	–	персональна мережа
PPS	–	Portable Protocol Stack
RFD	–	пристрій з полегшеними функціями
TDMA	–	часовий поділ
Wi-Fi	–	бездротова технологія

ВСТУП

Актуальність теми. Нафтогазова промисловість відіграє важливу роль у стимулюванні економічного зростання та забезпеченні енергетичних потреб сучасного суспільства. Однак ця критично важлива інфраструктура стикається з багатьма проблемами безпеки, які загрожують її діяльності, безпеці персоналу та цілісності навколишнього середовища.

Від крадіжок та вандалізму до кіберзагроз, ризики, з якими стикаються нафтогазові об'єкти, є багатогранними та вимагають проактивних заходів безпеки для ефективного їх пом'якшення.

Інфраструктура нафтогазових об'єктів – це величезна, різноманітна мережа, яка охоплює все: від бурових майданчиків та нафтопереробних заводів до трубопроводів та сховищ. Ці об'єкти часто розташовані у віддалених та важкодоступних районах, що ускладнює їхню безпеку. Кожен сектор нафтогазової промисловості має свої специфічні потреби в безпеці. Тому необхідні індивідуальні рішення для захисту від унікальних вразливостей.

Крім того, наявність вибухонебезпечних середовищ у багатьох нафтогазових середовищах створює додаткові проблеми безпеки, які вимагають спеціалізованих заходів безпеки для запобігання потенційним небезпекам. Важливість надійних процесів безпеки в бізнесі підтверджується необхідністю дотримання суворих правил та можливими втратами грошей в результаті порушення правил безпеки.

Мета й завдання дослідження. Метою роботи є програмне забезпечення системи мережевого відеонагляду газового родовища на основі обладнання Axis.

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

– Огляд існуючих систем мережевого відеонагляду газового родовища на основі обладнання Axis.

– Дослідження системи мережевого відеонагляду газового родовища на основі обладнання Axis.

– Програмна реалізація системи мережевого відеонагляду газового родовища на основі обладнання Axis.

Практична цінність отриманих результатів полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі мережевого відеонагляду газового родовища на основі обладнання Axis.

Таким чином, виходячи з вищеперерахованого, програмне забезпечення системи мережевого відеонагляду газового родовища на основі обладнання Axis, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти.

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Система призначена для мережевого відеонагляду газового родовища на основі обладнання Axis. Використовуючи IP-камери в сполученні з детекторами руху, у яких застосовується технологія радіолокації, можна створити рішення для охоронного відеоспостереження з обліком ваших конкретних вимог. Перевагами такого рішення будуть наступні особливості:

– Безпроблемна інтеграція в систему мережевих камер функції точного й надійного визначення місця, де відбулася несанкціонована дія, і наступного спостереження, що забезпечує візуальну ідентифікацію порушників.

– Автоматичне відправлення повідомлень у режимі реального часу, якщо хтось перетинає периметр зони обмеженого доступу, з візуальним підтвердженням і вказівкою місця розташування.

– Гнучкі й прості у використанні рішення із застосуванням міцних камер у вандалозахищеному виконанні й детекторів руху, призначених для роботи на вулиці.

Підвищення продуктивності праці й безпеки персоналу

Крім підтримки безпеки в приміщеннях компанії, мережеві камери також є коштовним засобом для підвищення загальної виробничої ефективності й створення безпечного робітничого середовища для співробітників. Інтеграція системи камер у виробничу систему забезпечує віддалений доступ до живих зображень мережею, а це означає появу наступних можливостей:

– Віддалений моніторинг виробничих ліній, можливість візуально перевірити й переконатися в правильності виконуваних функцій і процесів.

– Гарантія дотримання правил техніки безпеки й технологічних вимог, а також правильний обіг з інструментами й виробничим устаткуванням.

– Віддалена діагностика й усунення несправностей; допомога в обслуговуванні й підтримці.

1.2 Область застосування

Областю застосування є газові родовища України. Нафтогазові об'єкти охоплюють величезні території, включаючи бурові майданчики, нафтопереробні заводи, трубопроводи та сховища. Забезпечення безпеки цієї розгалуженої інфраструктури є значним викликом. Багато нафтогазових об'єктів розташовані у віддалених або важкодоступних районах (наприклад, на морських платформах, у пустелях або арктичних регіонах).

Для забезпечення безпеки цих ізольованих місць потрібні надійні мережі зв'язку, відеоспостереження та контроль доступу. Кожен тип об'єкта має унікальні вимоги до безпеки. Наприклад, кульові майданчики під час розвідки та видобутку корисних копалин потребують інших механізмів захисту, ніж трубопроводи для транспортування нафти та газу.

Як згадувалося раніше, нафтогазові об'єкти часто містять вибухонебезпечні гази або пари. Ці небезпечні середовища вимагають спеціалізованих рішень безпеки. Традиційні системи безпеки можуть бути непридатними через ризик займання. Іскробезпечне обладнання є важливим для запобігання іскрам або пожежам.

Будь-які перебої в нафтогазових операціях можуть мати серйозні наслідки. Порушення безпеки, навмисні чи випадкові, можуть порушити виробництво, транспортування та розподіл. Забезпечення безперебійної роботи при дотриманні надійних заходів безпеки є делікатним балансом.

Нафтогазова промисловість повинна збалансувати все це, дотримуючись суворих правил. Дотримання стандартів безпеки, охорони навколишнього середовища та охорони має вирішальне значення. Протоколи безпеки також повинні відповідати галузевим рекомендаціям та законодавчим вимогам.

Нафтогазові об'єкти, не підготовлені до інцидентів, можуть зазнати значних фінансових збитків. Крадіжка, вандалізм або саботаж можуть зупинити виробництво, що ще більше вдарить по доходах. Пом'якшення економічного

впливу передбачає проактивні заходи безпеки та швидке реагування на інциденти.

Зрештою, нафтогазові операції стають дедалі більш цифровими. Це наражає операції на значний ризик кіберзагроз, таких як кібератаки, витоки даних та атаки програм-вимагачів.

Нафтогазова промисловість стикається з багатогранними викликами безпеки, що вимагає індивідуальних рішень, що враховують унікальні характеристики цього сектору.

Таким чином, виходячи з вищеперерахованого, програмне забезпечення системи мережевого відеонагляду газового родовища на основі обладнання Axis, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти.

2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ

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

Якщо ви серйозно ставитеся до безпеки свого майна, то знаєте, що відеоспостереження – один з головних елементів для візуального контролю й захисту. Вона складається з відеокамери (або декількох), монітора, відеореєстратора, а також додаткового встаткування (ІЧ-прожектори, модулятори й т.д.). Офіси, торгові центри, приватний будинок, спостережень за територією, банки й інші об'єкти – неповний перелік об'єктів, де з успіхом застосовуються популярні системи відеоспостереження.

Але як вибрати пристрій, які системи більш вигідні або кращі? Проводячи огляд систем відеоспостереження, вийшла наступна картина.

Умовне перше місце популярності займають багатоканальні системи.

Їхні переваги:

– економія засобів – по суті, один комплект дозволяє забезпечити спостереження за всім об'єктом відразу (залежно від кількості камер);

– централізоване керування;

– всі елементи ретельно підібрані й сумісні, що гарантує надійність роботи.

Рисунок 2.1 – Багатоканальні системи

Друге місце за бездротовими системами відеоспостереження – які стають усе більше популярні.

Рисунок 2.2 – Бездротові системи відеоспостереження

Їх відрізняє:

– економія грошей на покупці кабелю, з'єднань, рознімань і т.д.;

– відсутність проводів допомагає зберегти дизайн;

– швидкий і більше простий монтаж устаткування;

– мобільність – якщо вам знадобитися закріпити камери в іншому місці, це не важко буде.

І нарешті, третє місце займає система з однією камерою:

– відмінно підійдуть для невеликих приміщень: кімната, сходова клітка, ларьок та ін.;

– ІЧ-підсвічування до 20 м.

Рисунок 2.3 – Система з однією камерою

Особняком знаходиться вибір камер відеоспостереження, адже найчастіше система підтримує різні моделі, і при бажанні ви можете їх міняти. Перше, що

потрібно знати – зовнішні камери більше вимогливі. На них впливають зовнішні фактори: зміна дня й ночі, сонячні промені, дощ, туман. Зрештою, їх можуть ушкодити хулігани.

Тому краще встановлювати камери на висоту не менш 3 м, вони повинні володіти вологостійким міцним корпусом. ІЧ-підсвічуванням (для кращої видимості вночі або при поганому освітленні), гарним розв'язною здатністю зйомки. Але це тільки деякі з основних вимог при виборі. Інші залежать від конкретних умов на місці установки й завдань.

Комплект бездротового IP-відеоспостереження Dahua KIT-IP 43-2B-W

Dahua KIT-IP43-2B-W являє собою готовий комплект бездротового IP-відеоспостереження, що включає компактний 4-канальний відеореєстратор з Wi-Fi модулем, дві корпусні Wi-Fi камери з розв'язною здатністю 3 мегапікселя для зовнішньої установки, а також джерело живлення й набір для монтажу. Разом з KIT-IP43-2B-W ви самостійно зможете створити якісну систему відеоспостереження в себе будинку, в офісі, магазині, складі або на іншому об'єкті. Для установки й настроювання системи не потрібні спеціальні знання й навички. У комплект входить все необхідне встаткування, за винятком жорсткого диска. Купуючи готовий набір устаткування ви заощаджуєте значну частину засобів.

Відеореєстратор Dahua DH-NVR4104HS-W-S2

Мережевий відеореєстратор Dahua DH-NVR4104HS-W-S2 (Wi-Fi) з убудованим Wi-Fi модулем (два діапазони: 2.4 ghz і 5 ghz), що дозволяє підключати 4 IP-камери спостереження з максимальним розв'язною здатністю до 5 мегапікселів. Дана модель прекрасно підходить для організації повноцінної й багатофункціональної системи бездротового відеоспостереження. Убудований процесор обробки здатний з високою ефективністю обробляти й відтворювати відео з розв'язною здатністю 2560 x 1920 із чотирьох камер, здійснювати архівний запис у форматі стиску H.264, а також передавати відеопотік мережею Інтернет. Ви зможете вести моніторинг у режимі реального часу через телевізор,

монітор комп'ютера або ios/Android смартфон, підключений до інтернету. Запис всієї інформації здійснюється на один жорсткий диск HDD SATA 3.5" з максимальною ємністю до 4 Тб. Завдяки підтримці стандарту сумісності ONVIF версії 2.4 ви зможете підключати відеокамери інших виробників.

Корпусна камера Dahua DH-IPC-HFW1320 S-W

IP камера відеоспостереження для зовнішнього або внутрішнього застосування. Пристрій наділений чутливою матрицею 1/3" progressive scan CMOS із прогресивною системою сканування, що здатний знімати відео з максимальним розв’язною здатністю 3 Мп (2304 x 1296 пікселів). Убудований об'єктив має фіксована фокусна відстань 3,6 мм і кут огляду 77 градусів. Гарну видимість в умовах недостатнього освітлення забезпечує інфрачервоне підсвічування з дальністю дії до 30 метрів, а також режим «День/Ніч». Завдяки підтримці мережевого доступу, ви зможете управляти камерою з комп'ютера, відео реєстратора, а також мобільних пристроїв на операційних системах ios, Android і Windows Phone. Інтернет підключення здійснюється через мережеве рознімання RJ-45 або убудований Wi-Fi (IEEE802.11b/g/n). Для зберігання записаної інформації передбачений окремий слот для установки Micro-SD карти пам'яті з максимальним обсягом до 128 Гб.

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

Оскільки потрібно розробити просту та легку у користуванні програму, яка б виконувалась під операційною системою Windows, то для її реалізації я обрав Builder C++. Існує велике число бібліотек написаних під Builder C++ , тому це одна з важливих причин вибору мови програмування. Середовище Builder C++ досить просте в користуванні, його вихідний код значно менше по об’єму в порівнянні з Delphi чи деякими іншими програмами такого типу. Досить легко

організувати взаємодію між модулями програм, об’єктно-орієнтований підхід дає можливість значно скоротити код програми, а отже і час його виконання.

На заміну старого розробленого набору елементів управління у Builder C++ інтегрована бібліотека візуальних компонентів VCL, представлених на палітрі компонентів. Після переносу на форму методом перетягування (drag-and-drop) компоненти відразу становляться діючими об’єктами вашої програми. Окрім типізованих інтерфейсних елементів Windows (кнопки, смуги прокручування, редагуємі текстові області, прості та комбіновані списки, та інше) у бібліотеку включені елементи підтримки діалогових вікон, обслуговування баз даних та багато іншого. Можливо не тільки модифікувати поведінку існуючих компонентів, але і будувати нові.

Builder C++ підтримує останні розширення стандарту мови C++ та забезпечує швидку компіляцію та складання 32-розрядних програм для Windows. Результуючі програми оптимізовані з точки зору швидкості виконання програм та затрат пам’яті. Зручний відладгоджувальник (з асемблерним вікном, можливістю крокового виконання, завдання точок зупинки, трасування та інше) повністю інтегрований у систему проектування. Дизайнер форм, редактор коду, інспектор об’єктів та інші інструменти зостаються доступними під час виконання програми, саме через це вносити зміни до коду можна прямо у процесі відлагодження.

Дизайнер форм, Інспектор об'єктів і інші засоби залишаються доступними під час роботи програми, тому вносити зміни можна в процесі відлагодження.

Builder C++ поставляється в трьох варіантах: Standard (стандартний), Professional (для професіоналів розробників, орієнтованих на мережеву архітектуру) і Client/Server Suite (для розробки систем в архітектурі клієнт/сервер). Останні два варіанти доповнюють стандартний початковими текстами візуальних компонентів, різномасштабним словником даних, новими функціями мови запитів SQL для бази даних, пакетом підтримки систем Internet, службою моніторингу програм, а також рядом інших засобів.

Builder C++ підтримує зв'язок з різними базами даних 3-х видів: dBASE і Paradox: Sybase, Oracle, InterBase і Informix; Excel, Access, FoxPro і Btrieve. Механізм BDE (Borland Database Engine) додає обслуговуванню зв'язків з базами даних дивовижну простоту і прозорість. Провідник Database Explorer дозволяє зображати зв'язки і об'єкти баз даних графічно. Використовуючи компоненти баз даних, я побудував електронний записник згідно таблиці dBASE за півгодини роботи на комп'ютері. Спадкоємство готових форм і їх "підгонка" під специфічні вимоги помітно скорочують тимчасові витрати на вирішення подібних завдань.

Довідкова служба Builder C++ надавала мені допомогу в цій і багатьох інших подібних ситуаціях. Є повний опис кожного управляємого компонента, включаючи списки властивостей і методів, а також численні приклади. Виклад матеріалу в книзі був значно покращуваний і систематизований завдяки відомостям, почерпнутим мною з довідкової служби.

Завдяки засобам управління проектами, двосторонній інтеграції додатку і синхронізації між засобами візуального і текстового редагування, а також вбудованому відладнику (з асемблерним вікном прокрутки, покрокового виконання, точок останову, трасуванням і тому подібне) – Builder C++ корпорації Borland надає собою вражаюче середовище розробки, яка, мабуть, витримає конкурентну боротьбу з такими модними продуктами як Developer Studio фірми Microsoft.

2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випускну кваліфікаційну роботу за першим (бакалаврським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи мережевого відеонагляду газового родовища на основі обладнання Axis.

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

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

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

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

г) організувати інтерфейс користувача з метою формування та виводу на екран ЕОМ повідомлень про некоректні дії користувача та нестандартні ситуації в роботі технологічного обладнання;

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

е) провести розрахунки по визначенню економічної ефективності розробленої системи;

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

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

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

Подолання проблем безпеки в нафтогазовій промисловості вимагає поєднання стратегічного планування, передових технологій та галузевої експертизи. Ось деякі підходи до вирішення цих проблем.

Оцінка та зменшення ризиків

Операції в нафтогазовій промисловості є складними та різноманітними. Проте проведення ретельної оцінки ризиків стає ключовим інструментом для виявлення вразливостей та потенційних загроз безпеці. Оцінка ризиків включає оцінку фізичного середовища, операційних процесів, технологічних систем та людського фактора.

Мета полягає у визначенні ймовірності та потенційного впливу інцидентів безпеки. Проведення комплексної оцінки ризиків допомагає організаціям розробляти індивідуальні стратегії для ефективного зниження ризиків та підвищення загальної стійкості безпеки.

Систематичне виявлення вразливостей, оцінка загроз та впровадження цільових стратегій пом'якшення наслідків можуть підвищити стійкість безпеки та мінімізувати ймовірність і вплив інцидентів безпеки. Система включає виявлення вразливостей, оцінку загроз, створення стратегій пом'якшення наслідків, а також моніторинг та перегляд на постійній основі.

Інтегровані системи безпеки

Інтегровані системи безпеки відіграють життєво важливу роль у захисті та безпеці нафтогазових об'єктів, забезпечуючи комплексну та цілісну безпеку, адаптовану до унікальних викликів, з якими стикається галузь. Ці системи поєднують різні компоненти безпеки, включаючи виявлення вторгнень по периметру, контроль доступу, відеоспостереження, дистанційний моніторинг та

керування сигналізацією, в єдину платформу.

Однією з основних переваг інтегрованої системи безпеки є її здатність забезпечувати цілісне уявлення про операції з безпеки на всьому об'єкті. Інтегруючи ці різні компоненти, інтегровані системи безпеки дозволяють здійснювати моніторинг у режимі реального часу, виявляти загрози та швидко реагувати, що може підвищити загальну безпеку нафтогазових об'єктів.

Збираючи інформацію з численних джерел, таких як відеопотоки, дані датчиків та журнали доступу, системи спостереження можуть забезпечити операторів моніторингу розумінням загроз безпеці та вразливостей у режимі реального часу. Це дозволяє операторам ефективніше виявляти та реагувати на інциденти безпеки, мінімізуючи ризик перебоїв у роботі.

Захист периметра

Периметральна безпека є важливим компонентом комплексних систем безпеки для нафтогазових об'єктів. Удосконалені системи виявлення вторгнень по периметру (PIDS), оснащені волоконно-оптичними датчиками та наземними радіолокаційними системами, забезпечують безперервний моніторинг меж об'єкта. Вони можуть виявляти будь-які спроби несанкціонованого вторгнення та негайно попереджати персонал служби безпеки.

Розгортаючи ці передові технології PIDS як частину інтегрованої системи безпеки, оператори можуть ефективно стримувати потенційних зловмисників та допомагати захищати критично важливу інфраструктуру.

Система контролю доступу

Контроль доступу – ще один ключовий аспект інтегрованих систем безпеки для нафтогазових об'єктів. Системи контролю доступу регулюють вхід до чутливих зон на об'єкті. Це гарантує, що лише уповноважений персонал має доступ до критично важливих активів та інфраструктури. Інтегруючи системи контролю доступу з іншими компонентами безпеки, такими як відеоспостереження та керування сигналізацією, оператори можуть посилити заходи безпеки та ефективніше забезпечувати дотримання протоколів безпеки.

Відеоспостереження з дистанційним моніторингом

Відеоспостереження є основою сучасних систем безпеки, особливо в середовищах з високим рівнем ризику, таких як нафтогазові об'єкти. Сучасні системи відеоспостереження пропонують можливості дистанційного моніторингу та управління. Вони надають операторам критично важливий інструмент для моніторингу, виявлення та реагування на загрози безпеці в режимі реального часу.

Оператори моніторингу можуть отримувати доступ до відео в реальному часі та керувати камерами з централізованого місця, розташованого далеко від нафтогазового об'єкта. Така гнучкість дозволяє співробітникам служби безпеки реагувати на загрози безпеці в реальному часі, незалежно від їхнього фізичного місцезнаходження.

Камери безпеки покращують ситуаційну обізнаність, оскільки оператори моніторингу та відеоаналітика можуть безперервно спостерігати за всім об'єктом. Моніторинг дозволяє швидко виявляти загрози безпеці та вживати відповідних заходів для операторів.

Камери відеоспостереження є ефективним інструментом для запобігання несанкціонованому доступу осіб до охоронюваних територій або незаконній діяльності. Потенційних зловмисників можна стримувати від спроб отримати несанкціонований доступ, крадіжки, вандалізму чи саботажу, оскільки вони знають, що за ними спостерігають. Цей проактивний засіб стримування безпеки допомагає запобігти інцидентам безпеки до їх виникнення, зменшуючи ризик перебоїв у роботі та пом'якшуючи потенційні втрати.

У разі інциденту, пов'язаного з безпекою, записи з камер відеоспостереження слугують цінним доказом для розслідування. У міру розвитку подій система записує та зберігає відеоматеріали. Їх можна передати правоохоронним органам, які можуть використовувати цю інформацію для ідентифікації винних, визначення послідовності подій та притягнення правопорушників до відповідальності.

Крім того, відеозаписи можна використовувати для судово-медичного аналізу, щоб реконструювати інцидент та виявити будь-які вразливості в протоколах безпеки, які, можливо, потребують усунення. Деякі компанії використовують відеозаписи для навчання, питань відповідальності та страхування.

Дистанційний відеомоніторинг є важливим інструментом для покращення планування дій у надзвичайних ситуаціях на нафтогазових об'єктах. Ці об'єкти повинні мати надійні плани готовності до надзвичайних ситуацій, а інтеграція передових технологій спостереження може допомогти досягти цієї мети. Завдяки обізнаності про ситуацію в режимі реального часу оператори можуть швидко реагувати на будь-які інциденти, незалежно від того, чи пов'язані вони з експлуатаційними несправностями, чи порушеннями безпеки. Крім того, дистанційний відеомоніторинг забезпечує критично важливий рівень інтелекту, який відповідає нормативним вимогам і допомагає забезпечити безпеку об'єкта.

Навчання та підвищення обізнаності персоналу

На нафтогазових об'єктах людський фактор так само важливий, як і технологічні засоби, такі як відеокамери, для забезпечення захисту об'єкта. Навчальні програми для працівників гарантують, що працівники знають про загрози безпеці та розуміють, як на них реагувати.

Навчальні програми забезпечують співробітників розумінням протоколів та процедур безпеки, що відповідають їхнім ролям та обов'язкам. Це включає в себе інструкції щодо контролю доступу, охорони периметра , реагування на надзвичайні ситуації та повідомлення про підозрілу діяльність. Ознайомлюючи співробітників із встановленими протоколами безпеки, навчальні програми допомагають підготувати персонал до належного реагування на інциденти безпеки та зменшення потенційних ризиків. Навчання повинно включати підвищення обізнаності з питань кібербезпеки, щоб навчити співробітників важливості протоколів кібербезпеки, таких як керування паролями, обізнаність про фішинг та безпечне поводження з конфіденційною інформацією.

Навчання та підвищення обізнаності персоналу – це безперервний процес, а не одноразова справа. Працівники повинні продовжувати навчання та посилення протоколів безпеки за допомогою регулярних тренінгів, курсів підвищення кваліфікації та кампаній з підвищення обізнаності.

Організації можуть створити культуру свідомості та пильності щодо безпеки, надаючи пріоритет обізнаності співробітників у сфері безпеки, зменшуючи кількість порушень безпеки та підвищуючи загальну стійкість безпеки.

Найефективніша та проактивна безпека нафтогазових об'єктів

Дистанційне відеоспостереження забезпечує цілодобове покриття всього об'єкта та навколишньої території. На відміну від традиційних систем безпеки, що реагують на зміни, дистанційне відеоспостереження – це проактивна система безпеки нафтогазових об'єктів, яка може допомогти запобігти злочинам до того, як буде завдано шкоди.

Камери безпеки можуть допомогти виявити помилки у дотриманні вимог до того, як щось станеться. Завдяки досягненням у технологіях безпеки нафтогазової промисловості, контроль доступу та дистанційне відеоспостереження можуть бути економічно ефективними рішеннями, які забезпечують швидку окупність інвестицій.

3.2 Розробка структурної схеми

Заходи безпеки на нафтогазових об'єктах та інфраструктурі повинні бути максимально ефективними на постійній основі. Причина цього полягає в тому, що сировина, що переробляється на цих об'єктах, зазвичай є легкозаймистою та вибухонебезпечною.

Проблеми забезпечення безпеки

Ланцюг видобутку нафти і газу охоплює всі напрямки морських розвідувальних та бурових платформ до наземних нафтових свердловин, від

трубопроводів до нафтопереробних заводів, і, коли йдеться про загальну безпеку, це створює багато проблем.

Як правило, об'єкти охоплюють величезні території та можуть бути розташовані у віддалених, важкодоступних місцях на суші або у відкритому морі, і часто зазнають впливу несприятливих екологічних умов. Ці об'єкти мають багато ключових зон, які можуть стати цілями для незаконного проникнення, вандалізму, крадіжки або, в найгіршому випадку, терористичної атаки.

З огляду на середовище високого ризику, виникають численні проблеми із забезпеченням захисту персоналу, дорогої інфраструктури та виробничих процесів. Однак те саме стосується суворих та ретельних стандартів, які визначають вимоги до обладнання, яке можна встановити в таких середовищах. Через те, що сировина, що переробляється в цих зонах, є надзвичайно легкозаймистою, токсичною та корозійною, міжнародні рекомендації класифікують ці об'єкти як такі, що мають «вибухонебезпечну атмосферу».

Міжнародні рекомендації

Усе електричне або механічне обладнання, встановлене в місці, яке класифіковано як потенційно вибухонебезпечне, повинно відповідати суворим проектним параметрам, проходити точні випробування на відповідність та отримувати сертифікат як «вибухобезпечний продукт» від акредитованих органів сертифікації. Насправді існує кілька типів методів захисту, щоб запобігти тому, щоб камери стали джерелом займання. Серед іншого: вибухобезпечні камери гарантують, що будь-які іскри, що виникають всередині їхніх корпусів, локалізовані, а температура зовнішньої частини пристрою, двигуна або корпусу підтримується в межах певних параметрів. Обидва ці фактори, електричні іскри та температура пристрою, є можливими джерелами займання, які можуть призвести до вибуху.

Відеоспостереження: чому і як

Відеоспостереження з використанням камер, які належним чином розміщені в критичних точках, є важливим у нафтогазовій промисловості, щоб

допомогти операторам своєчасно виявляти ризики, підозрілі події, несправності, аварії та крадіжки – все це загрожує безпеці персоналу та об'єктів.

Крім того, «віртуальне око» спеціалізованих камер здатне постійно контролювати делікатні виробничі процеси, що відбуваються в зонах, заповнених газом або токсичними випарами, або де температура занадто висока для людей.

Безсумнівно, найважливішими критеріями при проектуванні систем відеоспостереження завжди будуть надійність та якість. Також необхідно буде гарантувати довгострокову безперервність та підтримку за допомогою оновлень і нових технологічних продуктів, щойно вони з'являться на ринку.

Склад системи відеоспостереження

Системи мережевого відеоспостереження зараз є найбільш підходящою технологією для цих застосувань, що забезпечує певні переваги для нафтогазових об'єктів, такі як:

– Вища якість зображення. Більша деталізація дозволяє точніше аналізувати сцену, уникає хибних тривог та забезпечує більш своєчасне та ефективне втручання.

– Віддалений доступ для керування системою, навіть здалеку. Оператори можуть отримати доступ до потокового відео в реальному часі в будь-який час та з будь-якого авторизованого ПК, підключеного до Інтернету. Також є можливість архівування записів у віддалених місцях.

– Краща інтеграція з іншими системами. Завдяки стандартному протоколу ці пристрої можуть взаємодіяти з іншими системами активної безпеки, оптимізуючи та спрощуючи управління системою.

– Більша гнучкість та простіша модернізація системи в майбутньому. За необхідності ви можете додавати камери абсолютно без труднощів.

– Загалом нижчі витрати на встановлення та обслуговування.

Цифрові потоки даних – відео, аудіо та інші – передаються через локальну мережу (LAN) або глобальну мережу (WAN) за допомогою IP (інтернет-протоколу). Технологія кодування H.264 забезпечує високоякісне відео,

використовуючи менше місця для зберігання та пропускної здатності.

Управління IP-адресами забезпечує централізоване дистанційне керування за допомогою спеціалізованих програмних додатків, розроблених для збільшення часу реагування на проблеми та значного зменшення необхідності участі у вирішенні питань технічної допомоги, що є важливим фактором у випадку виробничих майданчиків, розташованих у важкодоступних місцях.

Крім того, цифрові дані полегшують аналіз відео, архівування та отримання записів у будь-який час і з будь-якого місця, навіть віддалено.

Який тип камери?

Для камери, яка використовуватиметься на об'єкті з ризиком вибуху, найголовнішою вимогою є її вибухобезпечність. Вона повинна гарантувати, що вона може містити будь-які іскри, джерела займання та не перегріватися.

Як результат, ці камери зазвичай мають корпус, виготовлений з такого матеріалу, як нержавіюча сталь, який достатньо міцний, щоб витримувати тиск внутрішнього вибуху. Він міцний і стійкий до корозії, спричиненої хімічними речовинами в навколишній атмосфері, зазвичай внаслідок виробничих процесів, що відбуваються, або морського туману, присутнього на об'єктах у відкритому морі або поблизу морських районів.

Щоб отримати максимальну віддачу від камери, об'єктив повинен бути оснащений вбудованим склоочисником, щоб зображення залишалося чітким без необхідності обслуговування. Також будуть необхідні спеціальні сертифіковані вибухобезпечні резервуари з насосами для утримання рідини для очищення скла.

Пости зі статичними камерами мають фіксоване поле зору та ідеально підходять у випадках, коли необхідно постійно спостерігати за певною чутливою зоною. Моторизовані PTZ-камери зазвичай використовуються для моніторингу великих зовнішніх територій. Причиною цього є можливість дистанційного керування панорамуванням, нахилом та масштабуванням, що дозволяє відстежувати рухомі цілі або виконувати цикл попередньо встановлених положень.

Коли йдеться про можливості запису, важливо вибирати високоякісні та високороздільні денні/нічні камери, щоб захопити найвищий рівень деталізації в будь-якій сцені, що спостерігається.

Камери можуть пропонувати розширені функції, такі як широкий динамічний діапазон, який дозволяє детально знімати людей або об'єкти при різному ступені освітлення, незалежно від того, чи є яскраві плями, тіньові ділянки чи сильне підсвічування. Також є функція Defog, яка використовує спеціальний алгоритм корекції зображень для створення дуже чітких кадрів крізь туман, дим або густий туман.

Технологія Full HD, яку зараз пропонують сучасні IP-камери, забезпечує виняткову якість відео та інші цікаві функції, такі як точність кольору, формат 16:9 та прогресивна розгортка, яка підтримує до 60 кадрів на секунду при повній частоті кадрів. Така висока частота кадрів означає, що детальні зображення несподіваних і раптових подій можна отримувати без проблем.

Тепловізори для моніторингу в темряві

Для найважливішого моніторингу навколишнього середовища, де необхідно гарантувати постійне візуальне покриття вдень і вночі незалежно від умов або відстані, найкращим рішенням є використання тепловізійної камери.

Тепловий датчик генерує зображення на основі тепла, що виділяється об'єктами або людьми, незалежно від того, наскільки яскраво освітлена сцена. Він неперевершений у виявленні наявності підозрілих об'єктів, незвичайних подій або підвищених температур, навіть у сліпучому світлі або повній темряві.

Тепловізори можуть допомогти виявити початок пожежі або витоки легкозаймистих речовин.

З цією метою на ринку існують PTZ-пристрої з подвійним баченням, які мають подвійні тепловізійні/оптичні камери день/ніч. Це ідеальне рішення для суворих вимог безпеки критичних середовищ.

Добре продумана система відеоспостереження, інтегрована з іншими активними та пасивними системами безпеки та розроблена з використанням

високоякісних і надійних продуктів, може знизити ризик у потенційно критичних зонах, таких як ті, що зустрічаються в нафтогазовій промисловості, сприяючи захисту здоров'я людей, активів та навколишнього середовища.

Рисунок 3.1 – Структурна схема системи

3.3 Розробка функціональної схеми

У даному пункті опишемо функціональну схему системи мережевого відеонагляду газового родовища на основі обладнання Axis. Функціональна схема зображена на рисунку 3.2. Підходи, які реалізуються в даній схемі, можуть використовуватися не тільки в газових родовищах, а й в інших сферах застосування відеоспостереження.

1. Швидкорозготаєме відеоспостереження

Швидкорозготаєме відеоспостереження оптимально підходить для тимчасових і віддалених установок. У цьому прикладі будівельний майданчик, компанія-постачальник послуг безпеки й керівник будівництва пов'язані із системами безпеки й керування доступом у реальному часі.

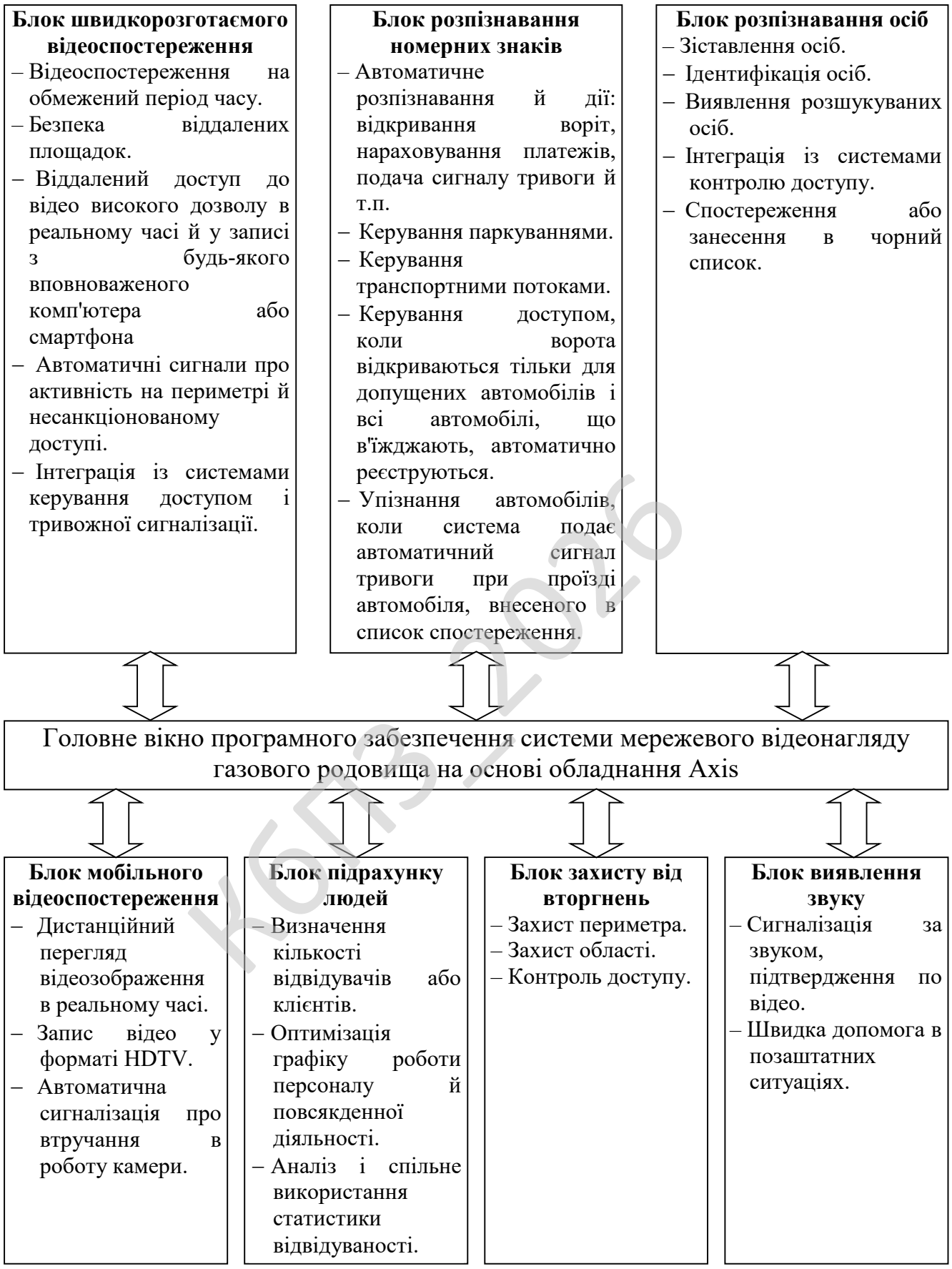

Рисунок 3.2 – Функціональна схема системи

Забезпечення безпеки в потрібнім місці й у потрібний час

Іноді охоронне відеоспостереження високого рівня потрібно лише на обмежений період часу. Також іноді виникає необхідність забезпечувати безпеку віддалених площадок. В обох випадках може бути занадто складно або дорого встановлювати спеціалізовану мережеву інфраструктуру для традиційної системи відеоспостереження.

Наше швидкорозготаєме рішення радикально міняє ситуацію. Воно швидко й легко встановлюються й при цьому надає найсучасніші можливості охоронного відеоспостереження.

Надійне відеоспостереження без розгортання фізичної мережі

Рішення складається з декількох IP-камер і/або відеодомофонів, укомплектованого заздалегідь апаратної шафи з убудованим маршрутизатором 4G/LTE і програмне забезпечення для віддаленого керування й перегляду відео. Після установки такої системи замовник одержує:

– Віддалений доступ до відео високого дозволу в реальному часі й у записі з будь-якого вповноваженого комп'ютера або смартфона

– Автоматичні сигнали про активність на периметрі й несанкціонованому доступі

– Простоту інтеграції із системами керування доступом і тривожної сигналізації

– Гнучке, масштабоване й надійне рішення для відеоспостереження, не потребуючого розгортання фізичної мережі на місці

Розумне й економічне багатоцільове рішення

Це комплексне рішення для забезпечення безпеки підходить для безлічі різних застосувань. Наприклад:

– Разові масові заходи – передвиборні зустрічі, концерти, марафонські забіги, фестивалі.

– Будівельні майданчики – для забезпечення безпеки або для спостереження за ходом будівництва через Інтернет.

– Дорожньо-будівельні проекти, у яких потрібне відстеження в реальному часі з боку влади для оптимізації керування рухом транспорту.

– Інфраструктура – для надійного моніторингу віддалених і безлюдних об'єктів.

При тимчасовому розгортанні, наприклад, на заходах або на будівельних майданчиках, те саме встаткування можна використовувати багаторазово, у результаті чого рішення стає ще економічніше.

2. Розпізнавання осіб

Як знайти одну особу з мільйона

Камера з ПЗ для розпізнавання осіб – це ефективне рішення у випадках, коли вам потрібно розпізнавати, ідентифікувати або підтверджувати особистість людей. Дана технологія допоможе вам поліпшити обслуговування клієнтів і захистити майно, визначаючи важливих клієнтів або потенційних порушників.

Функція розпізнавання осіб є відмінним доповненням для вашої системи охоронного відеоспостереження. Вона сприяє не тільки підвищенню безпеки, але й поліпшенню якості відеоспостереження в режимі реального часу.

Зіставлення осіб

Програмне забезпечення для розпізнавання осіб зіставляє особи з живого відео з базою даних або раніше збереженими фотографіями. Ви розподіляєте обличчя в базі даних по категоріях у відповідність із конкретними вимогами, такими як контроль доступу, виявлення VIP-персон або розшукуваних/небажаних людей.

Коли особа знімається камерою, у режимі реального часу проводиться зіставлення – і відповідність або невідповідність базі даних приведе до надання або заборони доступу, або до активації тривоги з повідомленням обраного персоналу відповідних служб про необхідність вживання заходів.

Численні можливості

Ми пропонуємо рішення для розпізнавання людських осіб, які можна застосовувати в самих різних ситуаціях:

– Ідентифікація осіб з категорії V.I.P. у казино, готелях або дорогих магазинах.

– Виявлення розшукуваних осіб в аеропортах.

– Інтеграція із системами контролю доступу в корпоративних і комерційних будинках.

– Спостереження або занесення в чорний список на стадіонах, у магазинах або казино.

3. Розпізнавання номерних знаків

Відеокамери з функціями LPR здатні автоматично зчитувати номерні знаки проїжджаючих автомобілів.

Ефективність, безпека й більше високий рівень обслуговування

Донедавна надійне розпізнавання номерних знаків (LPR) залишалося дорогою технологією, доступною лише для обмеженого числа додатків. Сьогодні швидкий розвиток IP-камер дозволяє застосовувати автоматичні рішення для розпізнавання номерних знаків значно ширше. Залежно від ваших потреб такі рішення можуть бути корисні для підвищення ефективності повсякденної роботи, рівня безпеки й захищеності персоналу.

Автоматичне розпізнавання й дії

Наші LPR-Рішення базуються на камерах Axis і програмному забезпеченні наших партнерів, що працює або на камері, або на сервері. Така система автоматично зчитує номерний знак у реальному часі, порівнює його або додає до заздалегідь певного списку й уживає відповідні дії – відкриває ворота, нараховує платіж, подає сигнал тривоги й т.п. Залежно від умов освітлення для оптимальної роботи рішення можуть знадобитися камери з убудованим інфрачервоним підсвічуванням або додаткові джерела світла.

Широка застосовність

Автоматичне розпізнавання номерних знаків може бути коштовним рішенням для безлічі різних завдань, включаючи наступні:

– Керування паркуваннями – для втілення й навіть автоматизації оплати, в'їзду й виїзду

– Платні дороги – для забезпечення невпинного проїзду через площадки оплати

– Керування транспортними потоками в аеропортах – для допуску тільки вповноважених транспортних засобів на смуги для таксі/суспільного транспорту

– Керування доступом, коли ворота відкриваються тільки для допущених автомобілів і всіх автомобілів, що в'їжджають, автоматично реєструються.

– Упізнання автомобілів, коли система подає автоматичний сигнал тривоги при проїзді автомобіля, внесеного в список спостереження

4. Мобільне відеоспостереження

Перевірені рішення для мобільного відеоспостереження застосовуються для захисту водіїв, пасажирів, вантажів, транспортних засобів. Вони створюються розраховуючи на тяжкі умови експлуатації на транспорті, що рухається, і передбачають ряд функцій для полегшення відеоспостереження:

– Дистанційний перегляд відеозображення в реальному часі.

– Запис відео у форматі HDTV.

– Автоматична сигналізація про втручання в роботу камери.

– Гнучкість і низька вартість установки.

Як це працює?

Наші рішення для мобільного відеоспостереження стійкі до вібрації, а дозвіл HDTV і технологія WDR дозволяють справлятися з коливаннями освітленості під час руху. Ви можете бути впевнені, що наші камери відповідають всім стандартам для бортових систем відеоспостереження.

Мережеве відеоспостереження ідеально сполучається з бездротовими технологіями 3G/4G/LTE, ніздрюватими мережами й технологіями Wi-Fi, значно полегшуючи надання дистанційної допомоги. Систему також можна інтегрувати з іншим устаткуванням – GPS, акселерометром, тривожною кнопкою водія для оперативної подачі сигналу тривоги.

Бортові системи, побудовані на основі наших рішень, можуть адаптувати якість переданого відео до доступної пропускної здатності каналу зв'язку, одночасно записуючи другий відеопотік з повною частотою кадрів і максимальною якістю відео на бортовий накопичувач. Запис і архівування можуть вироблятися як у самій камері, так і на бортовому комп'ютері автомобіля або на мережевих пристроях зберігання даних. Для мінімізації витрат і скорочення строків розслідування високоякісні записи подій можна зчитувати із транспортних засобів по бездротовому каналу.

Наші рішення для мобільного відеоспостереження рекомендуються для:

– Парків суспільного транспорту – автобусів, поїздів, трамваїв і т.д.

– Автомобілів екстрених служб – поліцейських, медичних, пожежних

– Вантажівок, розвозних фургонів і інших автомобілів, використовуваних для доставки посилок і товарів

– Інкасаторських автомобілів

– Будівельної техніки – самоскидів, бульдозерів, екскаваторів

– Таксі й маршрутних мікроавтобусів.

5. Підрахунок людей

Інтелектуальний і надійний підрахунок відвідувачів

Поняття «Великі дані» більше не є лише професійним терміном. У сучасному світі доступ до деталізованої аналітичної інформації, одержуваної в режимі реального часу, – ключова конкурентна перевага в багатьох галузях. Підрахунок кількості людей на основі відеозапису надає підприємствам роздрібної торгівлі й інших компаній сфери обслуговування спосіб одержання й аналізу даних про відвідувачів на різних об'єктах.

Наприклад, його використовують:

– роздрібні магазини для розрахунку коефіцієнтів перетворення й оцінки результатів своїх кампаній;

– музеї й компанії-організатори заходів для зручного надання статистики по відвідувачах.

Рішення Axis для підрахунку відвідувачів у роздрібній торгівлі

Якщо вам потрібно надійний і економічний спосіб виміру відвідуваності, направлені для підрахунку людей, засноване на відеотехнологіях, буде ідеальним вибором. Розроблене програмне забезпечення дозволяє з легкістю:

– Визначити кількість відвідувачів або клієнтів.

– Оцінити й підвищити ефективність маркетингової діяльності й дій по просуванню товарів.

– Оптимізувати графіки роботи персоналу й повсякденної діяльності.

– Аналізувати і спільно використовувати значиму статистику відвідуваності.

Крім того, можна вимірювати черги, одержувати демографічні показники й оцінювати розрахункове число людей у магазинах.

Чому варто вибрати відеоспостереження?

У сучасних методиках розрахунку числа людей по відео використовуються IP-камери з убудованими інструментами інтелектуального аналізу відеозображень, що забезпечують максимальну точність і надійність. Камери звичайно розміщаються над зоною, у якій ви хочете підраховувати вхідних людей. Уповноважені користувачі системи можуть переглядати статистику реального часу й історичні дані з будь-якого пристрою й у будь-якому місці. Система з легкістю додається в існуючу мережу й працює на основі IP-камер.

6. Захист від вторгнень

Забезпечте безпеку свого об'єкта – від периметра до найважливіших внутрішніх зон.

Рішення Axis для запобігання несанкціонованого доступу надійно захищає ваш об'єкт. Оператори, що чергують в апаратної, одержують відповідні сигнали від оптичних і тепловізійних камер з убудованими аналітичними додатками, робота яких доповнюється відстеженням за допомогою радара. Відбувається автоматичний запуск заздалегідь записаних попереджень, які відтворюються для

потенційних зловмисників, яких після цього можна пізнати за допомогою відеозаписів, зроблених PTZ-камерами.

Системи контролю доступу Axis не тільки запобігають несанкціонований доступ у найважливішу зону на вашім об'єкті й на навколишню її територію, але також дозволяють вам стежити за діями персоналу. Упевніться в тім, що ті, кому покладене, завжди перебувають там, де вони повинні перебувати, і роблять те, що вони повинні робити.

Головні переваги захисту від вторгнень

Захист периметра

Ви можете виявити й переконатися в наявності потенційного порушника в периметра вашого об'єкта практично з будь-якого місця в режимі реального часу. Поставте на службу собі тепловізійні або оптичні камери й радар, щоб вони виявляли, перевіряли й ідентифікували потенційних зловмисників. Наші системи охоронного відеоспостереження призначені для роботи навіть в умовах недостатньої освітленості або в повній темряві. Інтелектуальні аналітичні додатки запускають попередження й відповідні сигнали для персоналу, що заощаджує час і засоби, затрачувані на звичайне патрулювання з урахуванням помилкових тривог. Крім того, попередження можуть активувати відтворення готового оголошення, щоб змусити порушника покинути територію.

Захист області

Виявлення будь-якої небажаної активності й відстеження людей або предметів по декількох об'єктах з одержанням інформації про їхню швидкість, відстань до них і куті, під яким вони переміщаються. Використання камер, у яких передбачене керування панорамуванням, нахилом і зумом для спостереження за конкретними областями, дозволяє ідентифікувати людей на великих територіях. Для запобігання небажаних дій варто вживати відповідних заходів при одержанні сигналу тривоги, а також використовувати підготовлені звукові повідомлення. За допомогою рішення Axis все це можна робити, перебуваючи у віддаленому місці – де завгодно й у будь-який час.

Контроль доступу

Здійснюйте моніторинг входів і виходів, щоб бути впевненим у тім, що доступ на вашу територію надається тільки вповноваженому персоналу. Рішення Axis для контролю доступу забезпечують контроль на багатьох рівнях і на декількох об'єктах, незважаючи на те, що обробка даних ведеться в одному віддаленому місці. Можна охороняти весь будинок, окреме приміщення в будинку або шафа в цьому приміщенні. Наші рішення – це більше чим контроль. Вони дозволяють відслідковувати благополуччя кожної людини на вашім об'єкті й особливо тих, хто перебуває в найважливіших, чутливих або уразливих зонах.

7. Виявлення звуку

Оптимізація забезпечення безпеки з використанням інтелектуальних камер з функцією прийому звуку.

Багато інцидентів, що стосуються безпеки, випереджаються або починаються різними шумами. А якщо мова йде про можливі інциденти з агресивним поводженням, пострілами або битким склом, відеокамери з функціями виявлення звуку значно підвищують ефективність системи безпеки. Такі камери можуть автоматично сигналізувати про потенційних і інциденти, що відбуваються, у реальному часі й дозволяють швидко й адекватно реагувати на них, мінімізуючи наслідку.

Сигналізація за звуком, підтвердження по відео

Рішення складається з камери Axis і програмного забезпечення компанії-партнера для поглибленого аналізу звуку. Воно постійно відслідковує заздалегідь певні види шумів і ініціює сигнал тривоги, якщо такі шуми виникають. Це дозволяє негайно привернути увагу операторів системи безпеки, які можуть оцінити ситуацію по відеозображенню реального часу з камери, що подала сигнал, і сусідніх з нею камер. До системи також можна додати гучномовці, що дозволяють негайно звернутися до порушників, щоб запобігти збільшенню ситуації. Рішення відслідковує певні особливості звуку, а не конкретні слова, тому діє незалежно від мови й не порушує конфіденційності.

Швидка допомога в позаштатних ситуаціях

Відеокамери з функціями аналізу звуку дозволяють наділити пасивну систему відеоспостереження можливостями реагування, що попереджає. З їхньою допомогою можна оперативно реагувати в дистанційному режимі на позаштатні ситуації в самих різних сценаріях, таких як:

– Поява в суспільному транспорті п'яних і неадекватних осіб, що представляють небезпеку для пасажирів і персоналу. Припинення агресивного поводження на початкових стадіях – втручання на стадії словесних образ до переходу до фізичної агресії.

– Раннє виявлення тривожних ситуацій у в'язницях і виправних установах, де ув'язнені можуть навмисно уникати поля зору камери.

– Охорона невеликих магазинів у нічний час – реєстрація звуку дозволяє скоротити час реагування у випадку агресивного поводження або збройного нападу. Якщо магазин уночі закритий, це рішення здатне автоматично сигналізувати про звук розбитого скла для швидкого припинення злому.

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

3.4 Розробка діаграми процесів

Розглянемо розроблену діаграму процесів яка зображена на рисунку 3.3. Основна будова діаграми процесів полягає у графічному представленні складу сукупностей даних, що характеризуються як співвідношення різних частин кожної з сукупностей. Склад статистичної сукупності графічно може бути представлений як за допомогою абсолютних, так і відносних показників. Графічне зображення складу сукупності по абсолютними і відносними показниками сприяє проведенню більш глибокого аналізу і дозволяє проводити аналіз системи. Діаграма взаємодії процесів використовується для візуалізації процесів обробки даних (структурне проектування). Для розробника вважається звичним спочатку креслити діаграму взаємодії процесів даних рівня контексту, завдяки чому буде

показано взаємодію системи. Ця діаграма в подальшому підлягає уточненню шляхом деталізації процесів та потоків даних з метою показати систему що розробляється. При детальному її розгляді можна побачити як саме проходе взаємодія у розробленій системі. Використовується модель проектування, графічне представлення «потоків» даних в інформаційній системі.

Діаграми потоків даних містять чотири типи елементів:

– Зовнішні по відношенню до системи сутності.

– Потоки даних між елементами трьох попередніх типів.

– Процеси які являють собою трансформацію даних в рамках описуваної системи.

– Сховища даних (репозиторії).

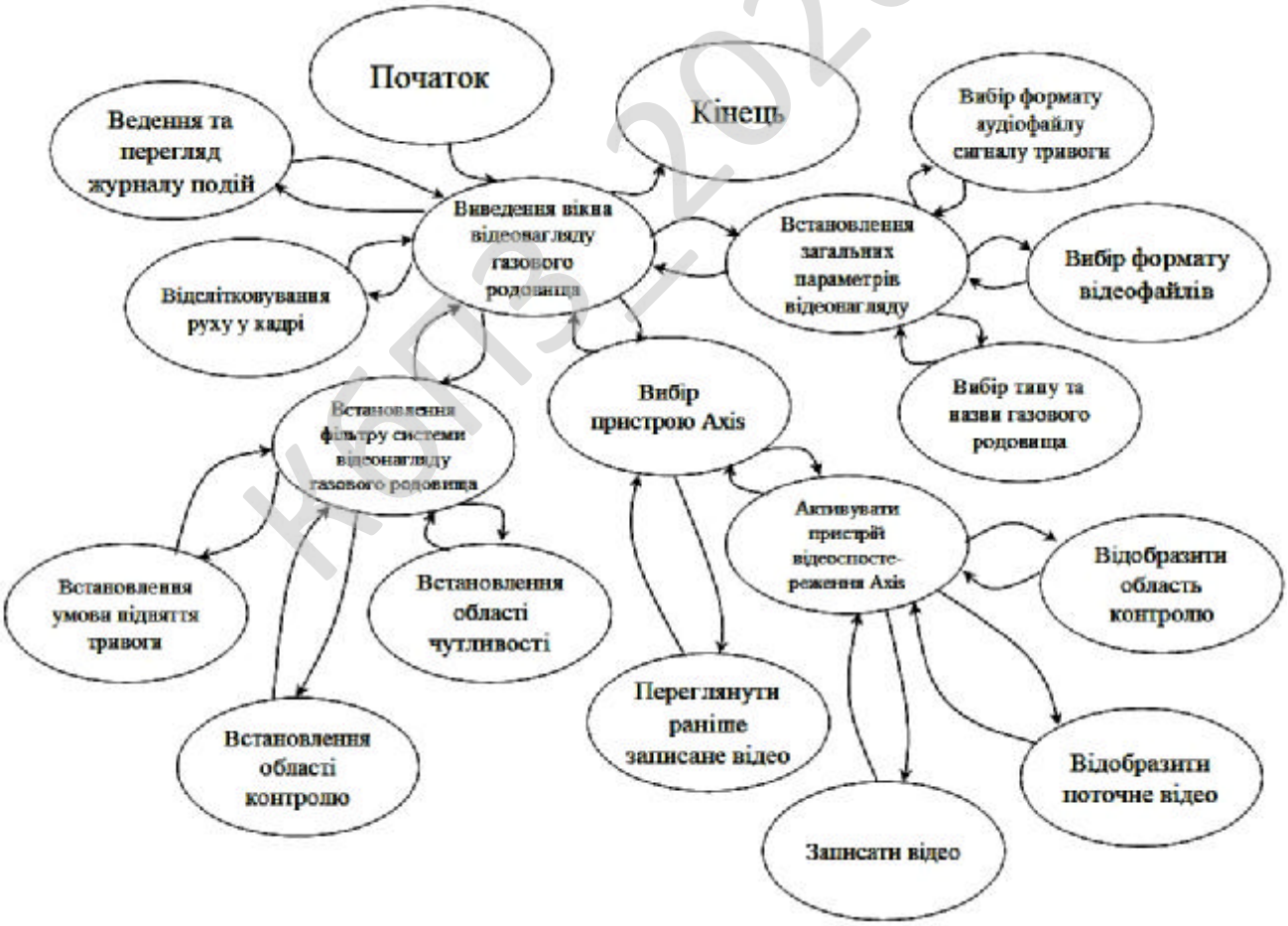

Рисунок 3.3 – Діаграма взаємодії процесів

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

4.1 Блок-схеми та опис алгоритмів функціонування системи

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

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

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

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

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

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

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

Незважаючи на те що я працював над ПЗ один в реалізації програми я використовував підходи пришвидшення розробки на основі методологій Agile.

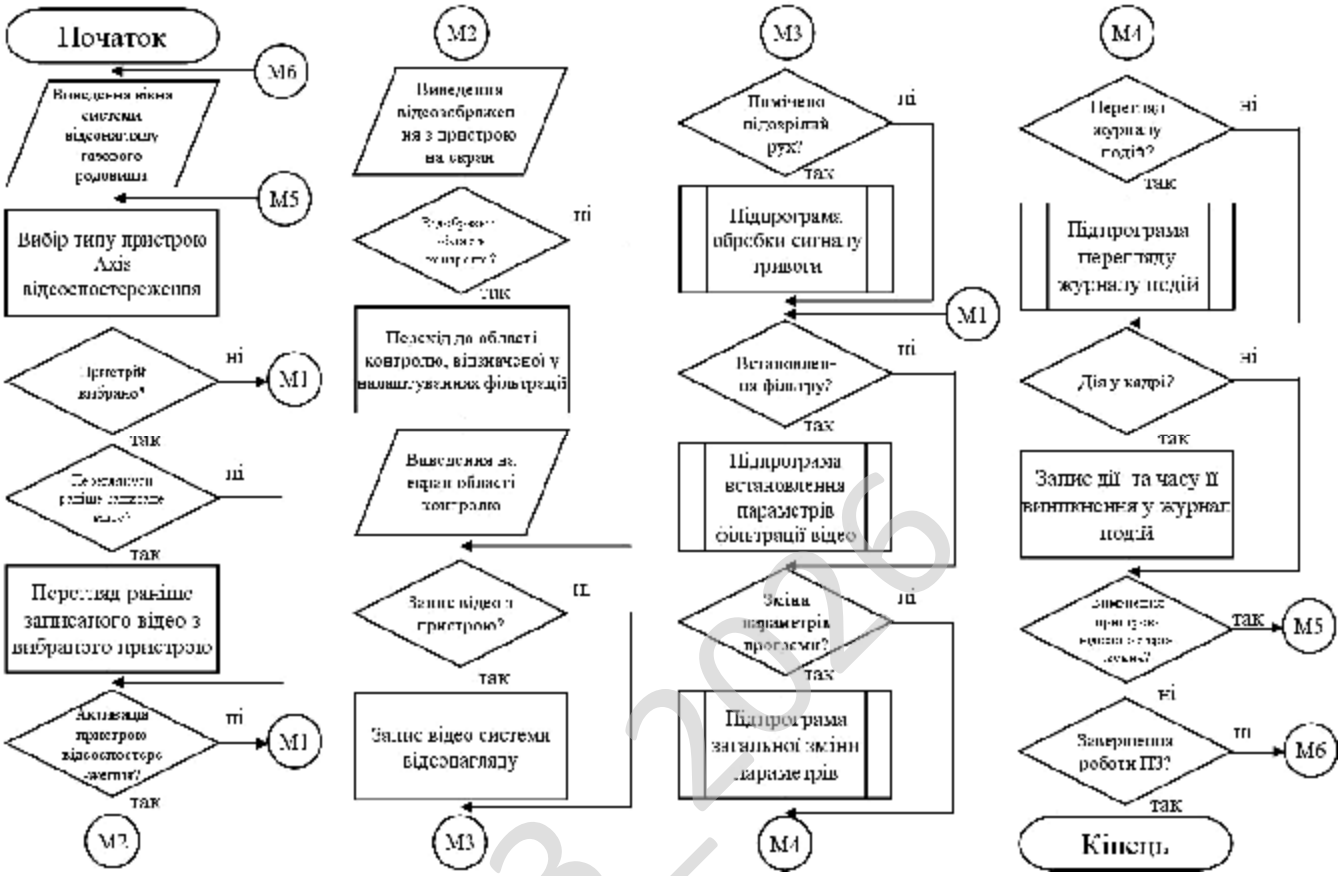

Рисунок 4.1 – Блок-схема основної програми

Гнучка́ розробка програмного забезпечення (Agile software development, agile-методи) – клас методологій розробки програмного забезпечення, що базується на ітеративній розробці, в якій вимоги та розв'язки еволюціонують через співпрацю між самоорганізовуваними багатофункціональними командами.

Гнучка розробка – найкращий засіб для підвищення продуктивності розробників програмного забезпечення.

Більшість гнучких методологій націлені на мінімізацію ризиків, шляхом зведення розробки до серії коротких циклів, що мають назву ітерацій, які зазвичай тривають один-два тижні.

Кожна ітерація сама по собі виглядає як програмний проект в мініатюрі, і включає всі завдання, необхідні для видачі мінімального приросту за функціональністю: планування, аналіз вимог, проектування, кодування, тестування і документування.

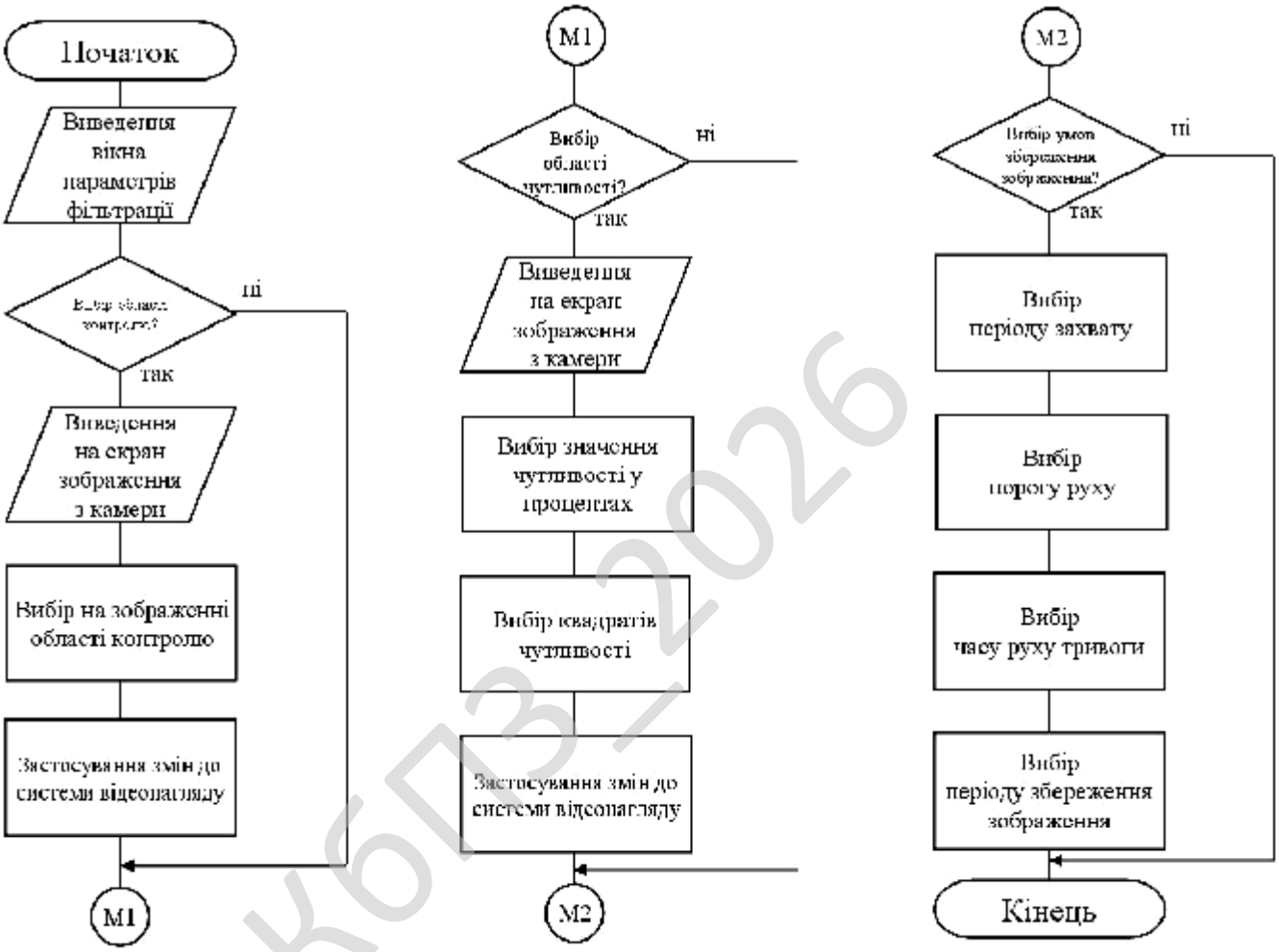

Рисунок 4.2 – Блок-схема роботи підпрограми

Хоча окрема ітерація, як правило, недостатня для випуску нової версії продукту, мається на увазі те, що гнучкий програмний проект готовий до випуску наприкінці кожної ітерації. Після закінчення кожної ітерації, команда виконує переоцінку пріоритетів розробки.

Agile акцентує увагу на безпосередньому спілкуванні «віч-на-віч». Більшість agile команд розташовані в одному офісі, його іноді називають bullpen.

Як мінімум вона включає і «замовників» (замовники, які визначають продукт, також це можуть бути менеджери продукту, бізнес аналітики або клієнти). Офіс може також включати тестувальників, дизайнерів інтерфейсу, технічних авторів і менеджерів.

Основною метрикою agile методів є робочий продукт. Віддаючи перевагу безпосередньому спілкуванню, agile-методи зменшують обсяг письмової документації в порівнянні з іншими методами. Це привело до критики цих методів як недисциплінованих.

Agile – родина процесів розробки, а не єдиний підхід в розробці програмного забезпечення, і визначається Agile Manifesto. Agile не включає практик, а визначає цінності та принципи, якими керуються успішні команди.

Agile Manifesto розроблений і прийнятий 17 розробниками 11-13 лютого 2001 року на лижному курорті The Lodge at Snowbird в горах Юти. Маніфест підписали представники наступних методологій Extreme programming, Scrum, DSDM, Adaptive software development, Crystal Clear, Feature driven development, Pragmatic Programming. Agile Manifesto містить 4 основні ідеї та 12 принципів. Примітно, що Agile Manifesto не містить практичних порад.

Основні ідеї:

– Особистості та їхні взаємодії важливіші, ніж процеси та інструменти;

– Робоче програмне забезпечення важливіше, ніж повна документація;

– Співпраця із замовником важливіша, ніж контрактні зобов'язання;

– Реакція на зміни важливіша, ніж дотримання плану.

Принципи, які роз'яснює Agile Manifesto:

– задоволення клієнта за рахунок ранньої та безперебійної поставки коштовного програмного забезпечення;

– вітання змін вимог навіть наприкінці розробки (це може підвищити конкурентоспроможність отриманого продукту);

– часта поставка робочого програмного забезпечення (кожен місяць або тиждень або ще частіше);

– тісне, щоденне спілкування замовника з розробниками впродовж всього проекту;

– проектом займаються мотивовані особистості, які забезпечені потрібними умовами роботи, підтримкою і довірою;

– рекомендований метод передачі інформації – особиста розмова (віч-на-віч);

– робоче програмне забезпечення – найкращий вимірювач прогресу;

– спонсори, розробники та користувачі повинні мати можливість підтримувати постійний темп на невизначений термін;

– постійну увагу поліпшенню технічної майстерності та зручному дизайну;

– простота – мистецтво не робити зайвої роботи;

– найкращі технічні вимоги, дизайн та архітектура виходять у самоорганізованої команди;

– постійна адаптація до мінливих обставин.

Маніфест та Принципи гнучкої розробки містять високорівневі ідеї щодо того, як потрібно вибудовувати процес розробки програмного забезпечення, щоб успішно завершувати проекти й створювати команди, в яких приємно та цікаво працювати.

Документи визначають, що потрібно для цього зробити, але не говорять, як це зробити. По-іншому й не могло бути, оскільки Маніфест та Принципи народилися внаслідок консенсусу представників різних (хоча й споріднених) напрямів, які могли знайти спільну основу лише на рівні базових цінностей та принципів.

Критика. Багато керівників проектів, що працюють у традиційних методологіях на кшталт «водоспаду», критикують agile-методи.

Один з повторюваних пунктів критики: при agile-підході часто нехтують створенням «дорожньої карти» розвитку продукту, так само як і управлінням вимогами, в процесі якого і формується така «карта». Гнучкий підхід до управління вимогами не має на увазі далекосяжних планів (по суті, управління

вимогами просто не існує в даній методології), а має на увазі можливість замовника раптом і несподівано наприкінці кожної ітерації виставляти нові вимоги, що часто суперечать архітектурі вже створеного і поставленого продукту. Таке іноді призводить до катастрофічних «авралів» з масовим рефакторингом і переробками практично на кожній черговій ітерації.

Крім того вважається, що робота в agile мотивує розробників вирішувати всі прибулі завдання найпростішим і найшвидшим можливим способом, при цьому часто не звертаючи уваги на коректність коду з точки зору вимог базової платформи (підхід «працює, та й добре», при цьому не враховується, що може перестати працювати при найменшій зміні або ж породити важкі до відтворення дефекти після реального розгортання у клієнта). Це призводить до зниження якості продукту і накопиченню дефектів.

Методології. Існують методології, які дотримуються цінностей і принципів заявлених в Agile Manifesto, деякі з них:

1. Agile Modeling – набір понять, принципів і прийомів (практик), що дозволяють швидко і просто виконувати моделювання і документування в проектах розробки програмного забезпечення. Не включає в себе детальну інструкцію з проектування, не містить описів, як будувати діаграми на UML.

Основна мета – ефективне моделювання і документування; але не охоплює програмування та тестування, не включає питання управління проектом, розгортання і супроводу системи. Однак включає в себе перевірку моделі кодом.

2. Agile Unified Process (AUP) спрощена версія IBM Rational Unified Process (RUP), розроблена Скоттом Амблером, яка описує просте і зрозуміле наближення (модель) для створення програмного забезпечення для бізнес-додатків.

3 Agile Data Method – група ітеративних методів розробки програмного забезпечення, в яких вимоги та рішення досягаються в рамках співпраці різних крос-функціональних команд.

4. DSDM заснований на концепції швидкої розробки додатків (Rapid Application Development, RAD). Являє собою ітеративний і інкрементний підхід, який надає особливого значення тривалій участі в процесі користувача/споживача.

5. Essential Unified Process (EssUP).

6. Екстремальне програмування (Extreme programming, XP).

7. Feature driven development (FDD) – функціонально-орієнтована розробка. Використовуване в FDD поняття функції або властивості (feature) Системи досить близько до поняття прецеденту використання, використовуваному в RUP, істотна відмінність – це додаткове обмеження: «кожна функція повинна допускати реалізацію не більше, ніж за два тижні». Тобто якщо сценарій використання досить малий, його можна вважати функцією. Якщо ж великий, то його треба розбити на декілька відносно незалежних функцій.

8. Getting Real – ітераційний підхід без функціональних специфікацій, що використовується для веб-додатків. У даному методі спершу розробляється інтерфейс програми, а потім її функціональна частина.

9. OpenUP – це ітераційно-інкрементний метод розробки програмного забезпечення. Позиціюється, як легкий і гнучкий варіант RUP. OpenUP ділить життєвий цикл проекту на чотири фази: початкова фаза, фази уточнення, конструювання та передачі. Життєвий цикл проекту забезпечує надання зацікавленим особам та членам колективу точок ознайомлення і прийняття рішень впродовж усього проекту. Це дозволяє ефективно контролювати ситуацію і вчасно приймати рішення про задовільність результатів. План проекту визначає життєвий цикл, а кінцевим результатом є остаточний додаток.

10. Scrum встановлює правила керування процесом розробки та дозволяє використовувати вже існуючі практики кодування, коректуючи вимоги або вносячи тактичні зміни. Використання цієї методології дає можливість виявляти і усувати відхилення від бажаного результату на більш ранніх етапах розробки програмного продукту.

11. Бережлива розробка програмного забезпечення (lean software development). Використовує підходи з концепції бережливого виробництва.

Було використано підходи з використанням UML, це уніфікована мова моделювання, використовується у парадигмі об'єктно-орієнтованого програмування. Є невід'ємною частиною уніфікованого процесу розробки програмного забезпечення. UML є мовою широкого профілю, це відкритий стандарт, що використовує графічні позначення для створення абстрактної моделі системи, називаної UML-моделлю. UML був створений для визначення, візуалізації, проектування й документування в основному програмних систем. UML не є мовою програмування, але в засобах виконання UML-моделей як інтерпретованого коду можлива кодогенерація.

Було використано наступні підходи UML: Діаграма об'єктів; Діаграма розгортання.

Діаграма об'єктів в UML це діаграма, що відображає об'єкти та їх зв'язки в певний момент часу. Діаграма об'єктів може розглядатись як окремий випадок діаграми класів, на якій можуть бути представлені як класи, так і екземпляри (об'єкти) класів. Схожою за змістом є діаграма взаємодії (collaboration diagram).

Діаграми об'єктів не мають власної нотації. Оскільки діаграми класів можуть відображати об'єкти, то діаграма класів, на якій відображено лише об'єкти, та не відображено класи, може вважатись діаграмою об'єктів.

Діаграма об'єктів відображає об'єкти та зв'язки в певний момент роботи програми. Об'єкти можуть містити інформацію про власні значення а не про описання. Для відображення загальних шаблонів об'єктів та зв'язків, що можуть багаторазово створюватись під час роботи програми, слід використовувати діаграму взаємодії, яка може відображати характеристики об'єктів та зв'язків. Екземпляр діаграми взаємодії створює діаграму об'єктів.

Діаграма об'єктів не відображає еволюцію системи під час роботи. Натомість, слід використовувати діаграми взаємодії з повідомленнями, або діаграми послідовності.

Діаграма розгортання (deployment diagram) це діаграма в UML, на якій відображаються обчислювальні вузли під час роботи програми, компоненти, та об'єкти, що виконуються на цих вузлах. Компоненти відповідають представленню робочих екземплярів одиниць коду. Компоненти, що не мають представлення під час роботи програми на таких діаграмах не відображаються; натомість, їх можна відобразити на діаграмах компонент. Діаграма розгортання відображає робочі екземпляри компонент, а діаграма компонент, натомість, відображає зв'язки між типами компонент.

SQLite – полегшена реляційна система керування базами даних. Втілена у вигляді бібліотеки, де реалізовано багато зі стандарту SQL-92.

Сирцевий код SQLite поширюється як суспільне надбання (public domain), тобто може використовуватися без обмежень та безоплатно з будь-якою метою. Фінансову підтримку розробників SQLite здійснює спеціально створений консорціум, до якого входять такі компанії, як Adobe, Oracle, Mozilla, Nokia, Bentley і Bloomberg.

З 2018р SQLite, як й JSON та CSV, рекомендований Бібліотекою Конгресу США формат зберігання структурованого набору даних. У 2005 році проект отримав нагороду Google-O'Reilly Open Source Awards.

Особливістю SQLite є те, що вона не використовує парадигму клієнт-сервер, тобто рушій SQLite не є окремим процесом, з яким взаємодіє застосунок, а надає бібліотеку, з якою програма компілюється і рушій стає складовою частиною програми.

Таким чином, як протокол обміну використовуються виклики функцій (API) бібліотеки SQLite. Такий підхід зменшує накладні витрати, час відгуку і спрощує програму.

SQLite зберігає всю базу даних (включаючи визначення, таблиці, індекси і дані) в єдиному стандартному файлі на тому комп'ютері, на якому виконується застосунок.

Простота реалізації досягається за рахунок того, що перед початком виконання транзакції весь файл, що зберігає базу даних, блокується; ACID-функції досягаються зокрема за рахунок створення файлу-журналу.

Кілька процесів або потоків можуть одночасно без жодних проблем читати дані з однієї бази. Запис в базу можна здійснити тільки в тому випадку, коли жодних інших запитів у цей час не обслуговується; інакше спроба запису закінчується невдачею, і в програму повертається код помилки. Іншим варіантом розвитку подій є автоматичне повторення спроб запису протягом заданого інтервалу часу.

У комплекті постачання йде також функціональна клієнтська частина у вигляді виконуваного файлу sqlite3, за допомогою якого демонструється реалізація функцій основної бібліотеки. Клієнтська частина працює з командного рядка, і дозволяє звертатися до файлу БД на основі типових функцій ОС.

Завдяки архітектурі рушія можливо використовувати SQlite як на вбудовуваних (embedded) системах, так і на виділених машинах з гігабайтними масивами даних.

Особливості SQLite:

1. Транзакції атомарні, послідовні, ізольовані, і міцні (ACID) навіть після збоїв системи і збоїв живлення.

2. Встановлення без конфігурації – не потребує ані установки, ані адміністрування.

3. Реалізує значну частину стандарту SQL92.

4. База даних зберігається в одному крос-платформовому файлі на диску.

5. Підтримка терабайтних розмірів баз даних і гігабайтного розміру рядків і BLOBів.

6. Малий розмір коду: менше ніж 350KB повністю налаштований, і менш 200KB з опущеними додатковими функціями.

7. Швидший за популярні рушії клієнт-серверних баз даних для найпоширеніших операцій.

8. Простий, легкий у використанні API.

9. Написана в ANSI C, включена прив'язка до TCL; доступні також прив'язки для десятків інших мов.

10. Добре прокоментований сирцевий код зі 100% тестовий покриттям гілок.

11. Доступний як єдиний файл сирцевого коду на ANSI C, який можна легко вставити в інший проект.

12. Автономність: немає зовнішніх залежностей.

13. Крос-платформовість: з коробки підтримується Unix (Linux і Mac OS X), OS/2, Windows (Win32 і WinCE). Легко переноситься на інші системи.

14. Сирці перебувають в суспільному надбанні.

15. Поставляється з автономним клієнтом інтерфейсу командного рядка, який може бути використаний для управління базами даних SQLite.

Інструменти створення та обслуговування БД

Створення та обслуговування БД можуть здійснюватись через текстову консоль SQL-командами або через спеціальні інструменти, у тому числі – з графічним інтерфейсом користувача.

Технології, що підтримують SQLite та мови програмування

Сама бібліотека SQLite написана мовою С. Проте є реалізація бібліотеки на JavaScript sql.js, яка дозволяє обробляти файли БД безпосередньо в браузері.

Для інших мов програмування розроблено механізм підключення й роботи з БД через цю бібліотеку: C++, Java, Python, Perl, PHP, Ruby, Haskell, Scheme, Smalltalk, Lua тощо. Засоби для роботи з Tcl включені в комплект постачання SQLite. Повний список наявних засобів можна знайти на сторінці проекту.

Web-інструментарії

У ряді інструментаріїв присутня можливість використання SQLite як бази даних, наприклад: Django; Java; PHP; Ruby on Rails; Trac; Symfony; Parser.

Багато програм підтримують SQLite як формат зберігання даних, зокрема:

1. Amarok – може використовувати бази даних SQLite як сховище музичної колекції.

2. Gajim – SQLite використовується для зберігання історії контактів.

3. Songbird (як застосунок, заснований на XULRunner 1.9)

4. Banshee.

5. F-Spot.

6. Платформа XUL на рушії Gecko 1.9, XULRunner 1.9 і, потенційно, всі застосунки, засновані на цій платформі, у тому числі й Firefox починаючи з версії 3.0.

7. Google Chrome.

8. Google Gears.

9. Mendeley – менеджер pdf-документів, академічний засіб для дослідження (реалізується desktop & web).

10. Zotero – менеджер інформації, бібліографічний менеджер, додаток Firefox.

Наведемо частину коду підпрограми для фільтрації відеозображення:

```
TSetFilterForm *SetFilterForm;
int SRX1, SRX2, SRY1, SRY2;
int CurSens = -1;
//---------------------------------------------------------------------------
__fastcall TSetFilterForm::TSetFilterForm(TComponent* Owner)
        : TForm(Owner)
{}
//---------------------------------------------------------------------------
void __fastcall TSetFilterForm::OkBitBtnClick(TObject *Sender)
{
   DeleteFile("prev12.cfg");
   Close();
}
//---------------------------------------------------------------------------
void __fastcall TSetFilterForm::CancelBitBtnClick(TObject *Sender)
{
   Sets.LoadFromFile("prev12.cfg");
   DeleteFile("prev12.cfg");
   Close();
```

```
}
//---------------------------------------------------------------------------
void __fastcall TSetFilterForm::SRPBPaint(TObject *Sender)
{
   DrawSR();
}
//---------------------------------------------------------------------------
void __fastcall TSetFilterForm::SRPBMouseMove(TObject *Sender,
      TShiftState Shift, int X, int Y)
{
  if(DrawRectButton->Down){
     SRPB->Cursor = crCross;
     if(Shift.Contains(ssLeft)){
//натиснута ліва клавіша миші
        SRX2 = X;
        SRY2 = Y;
        DrawSR();
     }
  }
  else{
     SRPB->Cursor = crDefault;
  }
}
//---------------------------------------------------------------------------
void __fastcall TSetFilterForm::SRPBMouseDown(TObject *Sender,
      TMouseButton Button, TShiftState Shift, int X, int Y)
{
   if(Button == mbLeft && DrawRectButton->Down){
      SRX1 = SRX2 = X;
      SRY1 = SRY2 = Y;
   }
}
//---------------------------------------------------------------------------
void TSetFilterForm::DrawSR()
{
    TPicture *tPic = new TPicture;
    tPic->Bitmap->Width = SRPB->Width;
    tPic->Bitmap->Height =  SRPB->Height;
    tPic->Bitmap->Canvas->Brush->Color = (TColor)RGB(100,100,100);
    tPic->Bitmap->Canvas->FillRect(tPic->Bitmap->Canvas->ClipRect);
    tPic->Bitmap->Canvas->Brush->Color = clWhite;
    tPic->Bitmap->Canvas->FillRect(Rect(SRX1, SRY1, SRX2, SRY2));
```

```
    tPic->Bitmap->Canvas->CopyMode = cmSrcAnd;
    tPic->Bitmap->Canvas->StretchDraw(tPic->Bitmap->Canvas->ClipRect,
Sets.FramePic->Bitmap);
    SRPB->Canvas->Draw(0, 0, tPic->Bitmap);
    delete tPic;
}
void __fastcall TSetFilterForm::FormShow(TObject *Sender)
{
   //зберегти поточні налаштування
   Sets.SaveToFile("prev.cfg");
   SRPB->Width = Sets.pW;
   SRPB->Height = Sets.pH;
   SensPB->Width = Sets.pW;
   SensPB->Height = Sets.pH;
   if(Sets.ClipRect.bottom > 0){
      SRX1 = Sets.ClipRect.left;
      SRX2 = Sets.ClipRect.right;
      SRY1 = Sets.ClipRect.top;
      SRY2 = Sets.ClipRect.bottom;
   }else{
      SRX1 = SRY1 = 0;
      SRX2 = Sets.pW;
      SRY2 = Sets.pH;
   }
   ActivePointsEdit->Text = IntToStr(Sets.MaxActivePoints);
   AlarmTimeEdit->Text = IntToStr(Sets.AlarmMotionTime);
   CycleTimeEdit->Text = IntToStr(Sets.CycleTime);
   ForceSaveEdit->Text = IntToStr(Sets.ForceSaveTime);
}
//---------------------------------------------------------------------------
void __fastcall TSetFilterForm::RectSaveButtonClick(TObject *Sender)
{
   int t;
   if(SRX1 > SRX2){
      t = SRX1; SRX1 = SRX2; SRX2 = t;
   }
   if(SRY1 > SRY2){
      t = SRY1; SRY1 = SRY2; SRY2 = t;
   }
   Sets.ClipRect.left = SRX1;
   Sets.ClipRect.right = SRX2;
   Sets.ClipRect.top = SRY1;
```

```
   Sets.ClipRect.bottom = SRY2;
   if(Sets.bConnected) capGrabFrame(Sets.hCaptureW);
}
//---------------------------------------------------------------------------
void __fastcall TSetFilterForm::SensPBPaint(TObject *Sender)
{
   DrawSensRegions();
}
//---------------------------------------------------------------------------
void TSetFilterForm::DrawSensRegions()
{
   int i;
   SensRegion *sR;
   TPicture *tPic = new TPicture;
   tPic->Bitmap->Width = SensPB->Width;
   tPic->Bitmap->Height =  SensPB->Height;
   for(i = 0; i < Sets.SensList->Count; i++){
      sR = (SensRegion*)Sets.SensList->Items[i];
      int cl = sR->sens * 255.0 / 100.0;
      tPic->Bitmap->Canvas->Brush->Color = (TColor)RGB(cl, cl, cl);
      tPic->Bitmap->Canvas->FillRect(*sR);
   }
   tPic->Bitmap->Canvas->CopyMode = cmSrcAnd;
   tPic->Bitmap->Canvas->Draw(0, 0, Sets.GrayPic->Bitmap);
   tPic->Bitmap->Canvas->Font->Color = clRed;
   tPic->Bitmap->Canvas->Pen->Width = 1;
   for(i = 0; i < Sets.SensList->Count; i++){
      sR = (SensRegion*)Sets.SensList->Items[i];
      tPic->Bitmap->Canvas->Brush->Style = bsSolid;
      tPic->Bitmap->Canvas->Brush->Color = clRed;
      tPic->Bitmap->Canvas->FrameRect(*sR);
      tPic->Bitmap->Canvas->Brush->Style = bsClear;
      tPic->Bitmap->Canvas->TextOut((sR->left+sR->right)*0.5-10, (sR->top+sR-
>bottom)*0.5-10, IntToStr(sR->sens));
   }
   SensPB->Canvas->Draw(0, 0, tPic->Bitmap);
   delete tPic;
  SensPB->Canvas->Draw(0, 0, Sets.GrayPic->Bitmap);
}
void __fastcall TSetFilterForm::SensSpeedButtonClick(TObject *Sender)
{
   if(!((TSpeedButton*)Sender)->Down){
```

```
      CurSens = -1;
   }else{
      CurSens = StrToInt(((TSpeedButton*)Sender)->Caption);
   }
}
//---------------------------------------------------------------------------
void __fastcall TSetFilterForm::SensPBMouseUp(TObject *Sender,
      TMouseButton Button, TShiftState Shift, int X, int Y)
{
   int i;
   SensRegion *sR;
   if(CurSens != -1 && Button == mbLeft){
      for(i = 0; i < Sets.SensList->Count; i++){
         sR = (SensRegion*)Sets.SensList->Items[i];
         if(sR->left < X && sR->right > X && sR->top < Y && sR->bottom > Y){
            sR->sens = CurSens;
         }
      }
      DrawSensRegions();
   }
}
//---------------------------------------------------------------------------
void __fastcall TSetFilterForm::SensPBMouseMove(TObject *Sender,
      TShiftState Shift, int X, int Y)
{
   if(CurSens != -1){
      SensPB->Cursor = crHandPoint;
   }else{
      SensPB->Cursor = crDefault;
   }
}
//---------------------------------------------------------------------------
void __fastcall TSetFilterForm::SensClearButtonClick(TObject *Sender)
{
   int i;
   SensRegion *sR;
   for(i = 0; i < Sets.SensList->Count; i++){
      sR = (SensRegion*)Sets.SensList->Items[i];
      sR->sens = 90;
   }
   DrawSensRegions();
}
```

```
//---------------------------------------------------------------------------
void __fastcall TSetFilterForm::AlarmApplyBitBtnClick(TObject *Sender)
{
   try{
      Sets.MaxActivePoints = StrToInt(ActivePointsEdit->Text.Trim());
      Sets.AlarmMotionTime = StrToInt(AlarmTimeEdit->Text.Trim());
      Sets.CycleTime = StrToInt(CycleTimeEdit->Text.Trim());
      Sets.ForceSaveTime = StrToInt(ForceSaveEdit->Text.Trim());
   }catch(...){
      ShowMessage("Введене число не входить в допустимий діапазон значень!");
   }
}
```

4.2 Захист розробленого програмного забезпечення

Дані в програмі захищаються за допомогою використання алгоритму SHA-3 (Keccak) – алгоритм гешування змінної розрядності, розроблений групою авторів на чолі з Йоаном Дайменом, співавтором Rijndael, автором шифрів MMB, SHARK, Noekeon, SQUARE і BaseKing. 2 жовтня 2012 року Keccak став переможцем конкурсу криптографічних алгоритмів, проведеним Національним інститутом стандартів і технологій США. 5 серпня 2015 року алгоритм затверджено та опубліковано в якості стандарту FIPS 202[1]. У програмній реалізації автори заявляють про 12,5 циклах на байт при виконанні на ПК з процесором Intel Core 2. Проте в апаратних реалізаціях Keccak виявився набагато швидшим, ніж всі інші фіналісти.

Конструкція функції губки, використана в геш-функції. P_i – вхідні блоки, Z_j – вихід алгоритму. Невикористаний для виведення набір бітів c («capacity») повинен мати значний розмір для досягнення стійкості до атак.

Алгоритм SHA-3 побудований за принципом криптографічної губки (дана структура криптографічних алгоритмів була запропонована авторами алгоритму Keccak раніше).

Геш-функції сімейства SHA-3 побудовані на основі конструкції криптографічної губки, в якій дані спочатку «вбираються» в губку, при якому

початкове повідомлення M піддається багатораундовим перестановкам f, потім результат Z «віджимається» з губки. На етапі «вбирання» блоки повідомлення додаються за модулем 2 з підмножиною стану, який потім перетвориться з допомогою функції перестановки f. На етапі «віджимання» вихідні блоки зчитуються з одного і того ж підмножинного стану, зміненого функцією перестановок f. Розмір частини стану, який записується і зчитується, називається «швидкістю» (англ. rate) і позначається r, а розмір частки, яка незаймана введенням / виведенням, називається «ємністю» (англ. capacity) і позначається c.

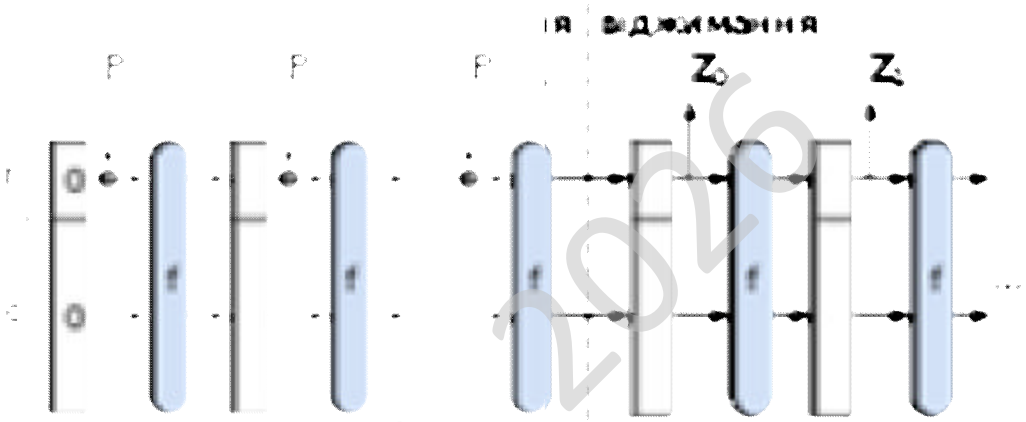

Рисунок 4.3 – Конструкція функції губки, використана в геш-функції

Алгоритм отримання значення хеш-функції можна розділити на кілька етапів:

– Вихідне повідомлення M додається до рядка P довжини, кратній r, за допомогою функції доповнення (pad-функції).

– Рядок P ділиться на n блоків довжини r: P_0, P_1,..., P_{n-1}

– «Всмоктування»: кожен блок P_i доповнюється нулями до рядка довжини b біт і підсумовується по модулю 2 з рядком стану S, де S – рядок довжини b біт (b = r + c). Перед використанням цієї функції всі елементи S дорівнюють нулю. Для кожного наступного блоку стан – рядок, отриманий застосуванням функції перестановок f до результату попереднього кроку.

– «Віджимання»: поки довжина Z менша d (d – кількість біт в результаті геш-функції), до Z додається r перших біт стану S, після кожного додавання до S, застосовується функція перестановок f. Потім S обрізається до довжини d біт

– Рядок Z довжини d біт повертається в якості результату

Завдяки тому, що стан містить c додаткових біт, алгоритм стійкий до атаки подовженням повідомлення, до якої прийняті алгоритми SHA-1 і SHA-2.

У SHA-3 стан S – це масив 5×5 слів довжиною $w = 64$ біта, всього $5 \times 5 \times 64 = 1600$ біт. Також в Keccak можуть використовуватися довжини w, рівні меншим ступенями 2 (від $w = 1$ до $w = 32$).

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

На рисунку зображено розроблене у бакалаврської дипломної роботі програмне забезпечення системи мережевого відеонагляду газового родовища на основі обладнання Axis. З рисунку можна побачити що інтерфейс головного вікна розподілено на наступні функціональні розділи:

– Навігаційного меню яке викликається натисканням правої клавіші маніпулятора миші.

– Верхнього меню: Дії; Вид; Довідка.

– Розділу виведення результату роботи системи.

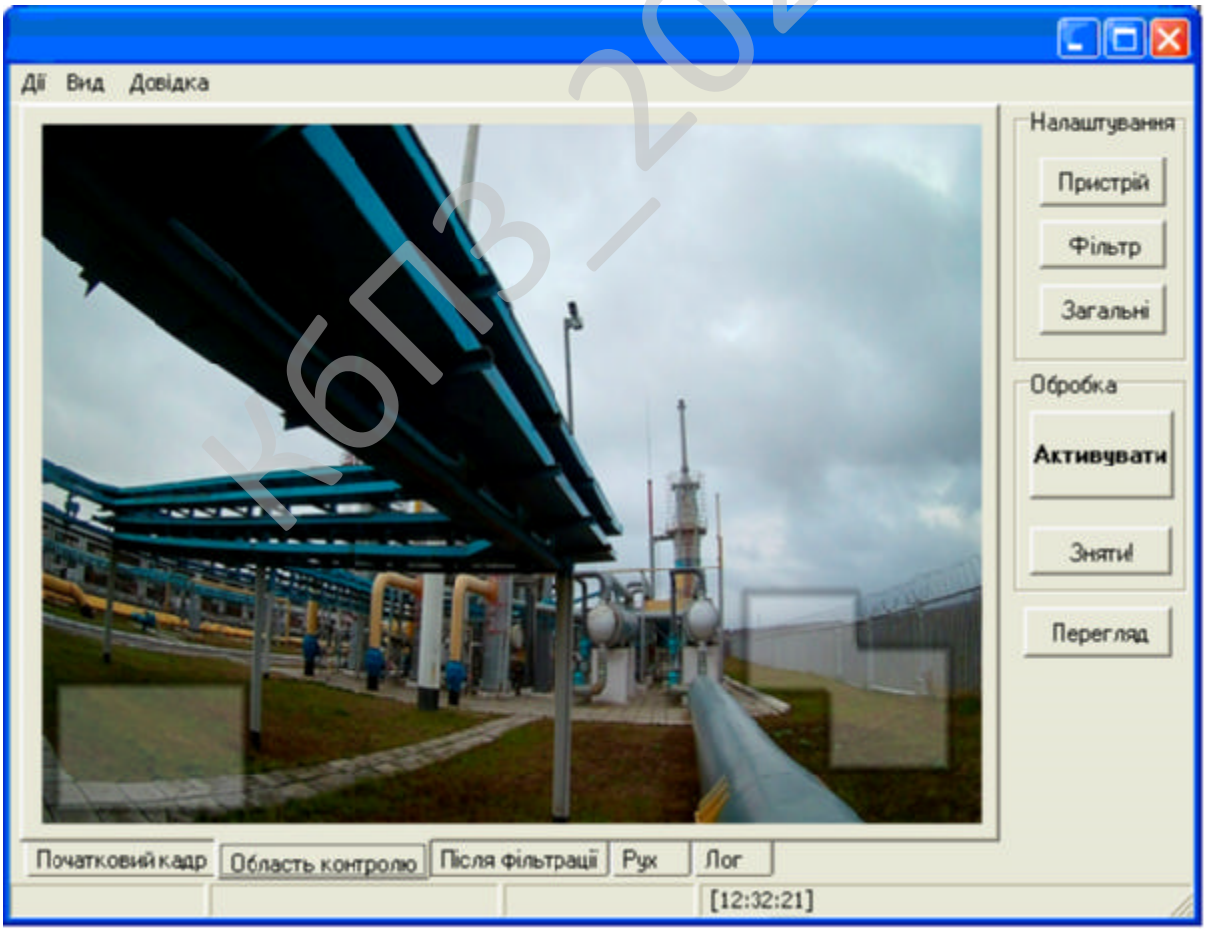

Рисунок 5.1 – Головне вікно ПЗ

Розроблена програма має дуже простий і зрозумілий інтерфейс з користувачем. Кожен, хто в достатньому обсязі володіє операційним середовищем Windows без особливих складностей освоїть і цю програму, оскільки її інтерфейс інтуїтивно зрозумілий. Якщо програма не видала ніяких помилок, і працює, то можна використовувати, інакше слід слідувати інструкціям, які пропонує програма.

На рисунку 5.2 зображено авторські дані розробленого програмного забезпечення.

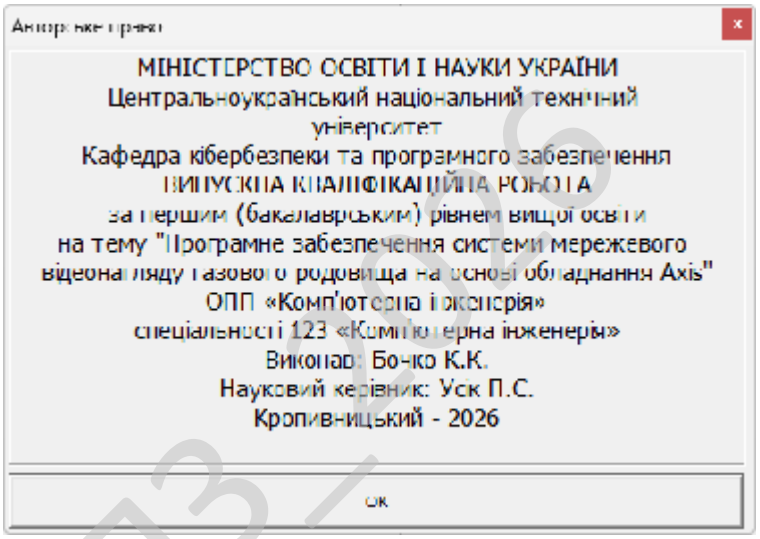

Рисунок 5.2 – Авторське право

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

Загалом процес розгортання складається з кількох взаємопов'язаних дій із можливими переходами між ними. Ця активність може відбуватися як з боку виробника так і з боку споживача. Оскільки кожна програмна система є

унікальною, то усі процеси та процедури під час розгортання важко передбачити. Тому, "розгортання" можна трактувати як загальний процес відповідно до певних вимог та характеристик. Розгортання може здійснюватись програмістом і в процесі розробки програмного забезпечення.

До діяльностей пов'язаних із розгортанням програмного забезпечення відносять:

– Випуск.

– Встановлення та активація.

– Деактивація.

– Адаптація.

– Обновлення.

– Вмонтування.

– Відстежування версій.

– Видалення.

– Вилучення з обігу.

При впровадженні програмного забезпечення потрібно урахувати наступні дії:

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

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

– Виконання робіт з загальної стандартизації існуючої діяльності організації, коли виділяються кращі практики виконання процедур і включаються

в ІТ рішення за принципом найбільшої корисності для більшості учасників. Відсоток таких процедур щодо загального обсягу автоматизації може бути невеликий, але це надає процесу побудови рішення вагу в організації за рахунок збільшення його необхідності.

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

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

Проводилась оцінка:

– відповідності поставленим вимогам;

– правильна відповідь для усіх можливих вхідних даних;

– виконання функцій за прийнятний час;

– практичність;

– сумісність з ОС та стороннім ПЗ.

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

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

Проводилось тестування форматом білої скриньки засноване на аналізі керуючої структури програми. Програма вважається повністю перевіреною, якщо проведено вичерпне тестування маршрутів (шляхів) її графа управління.

У цьому випадку формуються тестові варіанти, в яких:

– Гарантується перевірка всіх незалежних маршрутів програми.

– Знаходяться гілки True, False для всіх логічних рішень.

– Виконуються всі цикли (у межах їхніх кордонів та діапазонів).

– Аналізується правильність внутрішніх структур даних.

Недоліки тестування "білої скриньки":

– Кількість незалежних маршрутів може бути дуже велика.

– Повне тестування маршрутів не гарантує відповідності програми вихідним вимогам до неї.

– У програмі можуть бути пропущені деякі маршрути.

– Не можна виявити помилки, поява яких залежить від даних.

Переваги тестування "білої скриньки" пов'язані з тим, що принцип «білої скриньки» дозволяє врахувати особливості програмних помилок:

– Кількість помилок мінімально в «центрі» і максимально на «периферії» програми.

– Попередні припущення про ймовірність потоку керування або даних у програмі часто бувають некоректними. У результаті типовим може стати маршрут, модель обчислень за яким опрацьована слабо.

– При записі алгоритму програмного забезпечення у вигляді тексту на мові програмування можливе внесення типових помилок трансляції (синтаксичних та семантичних).

– Деякі результати в програмі залежать не від вихідних даних, а від внутрішніх станів програми.

Проводилось тестування чорної скриньки.

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

Ці тести демонструють:

– Як виконуються функції програми.

– Як приймаються вихідні дані.

– Як виробляються результати.

– Як зберігається цілісність зовнішньої інформації.

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

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

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

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

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

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

Будь-який спосіб тестування «чорної скриньки» повинен:

– Виявити такі вхідні дані, які з високою ймовірністю належать набору ITc;

– Сформулювати такі очікувані результати, які з високою імовірністю є елементами набору ОТ.

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

Тестування «чорної скриньки» забезпечує пошук наступних категорій помилок:

– Некоректних чи відсутніх функцій;

– Помилок інтерфейсу;

– Помилок у зовнішніх структурах даних або в доступі до зовнішньої бази даних;

– Помилок характеристик (необхідна ємність пам'яті і т.д.);

– Помилок ініціалізації та завершення.

Обрано умови розповсюдження – Shareware.

Під умовно-безплатним програмним забезпеченням можна розуміти спосіб або метод розповсюдження комерційного ПЗ на ринку (тобто на шляху до кінцевого користувача), при якому випробувачеві пропонується обмежена за можливостями (не повнофункціональна або демонстраційна версія), терміном дії (тріал версія) або версія з вбудованим набридливим нагадуванням про необхідність оплати використання програми.

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

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

Якщо після закінчення цього терміну користувач вирішить продовжити використання ПЗ, він зобов'язаний купити його (зареєструватися), заплативши авторові певну суму.

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

6 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти, призначено для системи мережевого відеонагляду газового родовища на основі обладнання Axis.

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

Рішення завдання полягало у вирішенні наступних задач:

– Був проведений огляд існуючих систем мережевого відеонагляду газового родовища на основі обладнання Axis.

– Досліджена система мережевого відеонагляду газового родовища на основі обладнання Axis.

– На основі отриманих результатів досліджень створена програмна реалізація системи мережевого відеонагляду газового родовища на основі обладнання Axis.

Розроблені під час виконання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання мережевого відеонагляду газового родовища на основі обладнання Axis.

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

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

Програма реалізована на мові високого рівня Builder C++. Дана мова програмування дозволяє найбільш ефективно обробляти дані призначені для системи мережевого відеонагляду газового родовища на основі обладнання Axis.

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

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

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

Для підвищення рівня безпеки запропоновано застосовувати алгоритм SHA-3.

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

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Михайло Пічугін, Іван Канкін, Володимир Воротніков Комп'ютерна графіка. Навчальний посібник / Центр навчальної літератури 346 с. 2019р.

2. Маценко В.Г. Комп’ютерна графіка: Навчальний посібник. – Чернівці: Рута, 2009 – 343 с.

3. Інженерна комп’ютерна графіка: підручник / В.В. Проців [та ін.] / М-во освіти і науки України, Нац. гірн. унт-т. – Дніпро: НГУ, 2017. – 247 с.

4. Проців В.В. Прикладна комп'ютерна графіка [Текст]: Навч. посібник / В.В. Проців, К.А. Зіборов, К.М. Бас, Г.К. Ванжа; М-во освіти і наук, Нац. гірн. ун-т. - Д.: НГУ, 2016. - 187 с.

5. Kopf, Johannes and Lischinski, Dani. Depixelizing Pixel Art (англ.) // ACM Trans. Graph. – 2011. – Vol. 30, no. 4. – P. 99:1--99:8.

6. Giachetti, Andrea and Asuni, Nicola. Real-Time Artifact-Free Image Upscaling (англ.) // Trans. Img. Proc.. – 2011. – Vol. 20, no. 10. – P. 2760—2768.

7. Kuznetsov, O., Frontoni, E., Kryvinska, N., Chevardin, V., Smirnov, O. «Wireless Network Encryption Stream Ciphers, Computational Modeling, and Security Analysis». *Computational Modeling and Simulation of Advanced Wireless Communication Systems*, 2024, pp. 379–402.

8. Kuznetsov, O., Frontoni, E., Kryvinska, N., Smirnov, O., Imoize, G.L. «Computational Modeling of Enhanced Spread Spectrum Codes for Asynchronous Wireless Communication». *Computational Modeling and Simulation of Advanced Wireless Communication Systems*, 2024, pp. 403–447

9. Смірнова Т.В., Конопліцька-Слободенюк О.К., Буравченко К.О., Смірнов С.А., Кравчук О.В., Козірова Н.Л., Смірнов О.А. «Дослідження технологій забезпечення кібербезпеки хмарних сервісів IaaS, PaaS та SaaS». *Кібербезпека: освіта, наука, техніка.* 2024. №4(24), С. 6-27.

10. Батрак О., Смірнова Т., Гнатюк В., Одарченко Р., Смірнов О. «Дослідження показників ефективності функціонування та перспектив розвитку систем IP-телефонії». *Підводні технології,* 2024, № 13, с. 28-35.

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

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

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

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

15. Smirnov, O., Neskorodieva, T., Fedorov, E., Rudakov, K., Neskorodieva, A. «Method Detection Audit Data Anomalies on Basis Restricted Cauchy Machine» CEUR Workshop Proceedings, Volume 3187, 2022,

16. Smirnov O., Smirnova T., Anas M. Al-Oraiqat, Drieiev O., Polishchuk L., Sheroz Khan, Yassin M. Y. Hasan, Aladdein M. Amro, Hazim S. AlRawashdeh «Method for Determining Treated Metal Surface Quality Using Computer Vision Technology». Sensors (Basel, Switzerland) Volume 22, Issue 16, 6223, 2022.

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

18. Smirnov O., Kuznetsov A., Zhora V., Onikiychuk A., Pieshkova O. «Hiding Messages in Audio Files Using Direct Spread Spectrum». 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2021, Cracow, Poland, 22-25 September 2021. P. 414-418.

19. Smirnov O., Kuznetsov A., Lokotkova I., Kuznetsova T., Florov S., Lebid O. «Using Orthogonal Signals to Hide Information in Images». 4 IEEE International Conference on Advanced Information and Communication Technologies (AICT) - 2021, Lviv, Ukraine, September 21-25, 2021. P. 255-260.

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

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

22. Smirnov O., Alimseitova Zh., Adranova A., Akhmetov B., Lakhno V., Zhilkishbayeva G. «Models and algorithms for ensuring functional stability and cybersecurity of virtual cloud resources». Journal of theoretical and applied information technology Vol.98. No 21, 2020, P. 3334-3346.

23. Smirnov O., Kuznetsov A., Kovalchuk D., Kuznetsova T. «New technique for data hiding in cover images using adaptively generated pseudorandom sequences». CEUR Workshop Proceedings Volume 2654, 2020, Pages 1-14.

24. Smirnov O., Kuznetsov A., Onikiychuk A., Makushenko T., Anisimova O., Arischenko A. «Adaptive pseudo-random sequence generation for spread spectrum image steganography». 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT), Ukraine, Kyiv, May 14-18. 2020. P. 161-165.

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

26. Smirnov O., Kuznetsov A., Pushkar'ov A., Serhiienko R., Babenko V., Kuznetsova T., «Representation of Cascade Codes in the Frequency Domain». In: Radivilova T., Ageyev D., Kryvinska N. (eds) Data-Centric Business and Applications. Lecture Notes on Data Engineering and Communications Technologies, vol 48. Springer, Cham. 2021. pp 557-587.

27. Smirnov, O., Drieieva, H., Drieiev, O., Polishchuk, Y., Brzhanov, R., Aleksander, M. «Method of fractal traffic generation by a model of generator on the graph». CEUR Workshop Proceedings Volume 2616, 2020, Pages 366-379.

28. Smirnov, O., Drieieva, H., Drieiev, O., Simakhin, V., Bondar, S., Odarchenko, R. «Managing multifractal properties of the binary sequence generated with the Markov chains», CEUR Workshop Proceedings Volume 2608, 2020, Pages 633-645.

29. Smirnov, O., Kuznetsov, A., Gorbacheva, L., Babenko, V., «Hiding data in images using a pseudo-random sequence», CEUR Workshop Proceedings Volume 2608, 2020, Pages 646-660.

30. Zhurakovskyi, B., Tsopa, N., Batrak, Y., Odarchenko, R., Smirnova, T «Comparative analysis of modern formats of lossy audio compression». Workshop Proceedings, 2020, 2654, стр. 315-327.

31. Smirnov O. Kuznetsov A., Zaicenko Yu., Pastukhov M., Oleshko O., Kuznetsova K., «Formation of Discrete Signals with Special Correlation Properties». International Conference on Information and Telecommunication Technologies and Radio Electronics, UkrMiCo 2019; Odessa; Ukraine; 9-13 September 2019. P.22-28.

32. Smirnov, O., Kuznetsov, A., Kolovanova, I., Kuznetsova, T., «Noise immunity of the algebraic geometric codes». International Journal of Computing; 2019,

Volume 18, Issue 4 – Research Institute for Intelligent Computer Systems – 2019. – P. 393-407.

33. Smirnov, O., Kuznetsov, A., Reshetniak, O., Ivko, N., Katkova, T., Kuznetsova, T., «Generators of Pseudorandom Sequence with Multilevel Function of Correlation». 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T), Kyiv, Ukraine, 8 – 11 October 2019 . P.517-522.

34. Smirnov, O., Krasnobayev, V., Yanko, A., Kuznetsova, T. «Methods of nulling numbers in the system of residual classes». CEUR Workshop Proceedings, Vol 2588, P. 90-106, 2019.

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

36. Smirnov, O., Kuznetsov, A., Kavun, S., Babenko, B., Nakisko, O., Kuznetsova, K., «Malware Correlation Monitoring in Computer Networks of Promising Smart Grids», 2019 IEEE 6th International Conference On Energy Smart Systems (2019 IEEE ESS), Kyiv, Ukraine April 17-19, 2019 P. 347-352.

37. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Pastukhov, M., Kuznetsova, K., Prokopovych-Tkachenko, D., «Discrete Signals with Special Correlation Properties», CEUR Workshop Proceedings Volume 2353, CEUR Workshop Proceedings 2019, Pages 618-629.

38. Smirnov A.A., Kuznetsov A.A., Danilenko D.A., Berezovsky A., «The statistical analysis of a network traffic for the intrusion detection and prevention systems», Telecommunications and Radio Engineering. – Volume 74, Issue 1. – Begel House Inc. – 2015. – P. 61-78.

39. Smirnov O., Kuznetsov A., Kovalchuk D., Kuznetsova T. «New Technique for Hiding Data in Cover Images Using Adaptively Generated

Pseudorandom Sequences». CEUR Workshop Proceedings Volume 2732, 2020, Pages 214-227.

40. Т.В. Смірнова, О.М. Дрєєв, О.А. Смірнов «Хмарна інформаційна система оцінювання шорсткості з використанням дискретного частотного аналізу макрофотографій». IV міжнародна науково-практична конференція "Інформаційна безпека та комп'ютерні технології", м. Кропивницький. 15-16 квітня 2021р. – Кропивницький: ЦНТУ. – 2021. – С. 30.

41. О.А. Смірнов, П.С. Усік, «Дослідження перспектив використання технологічних рішень в мережах 5G» у Кібербезпека та інформаційні технології: монографія. – Х. : ТОВ «ДІСА ПЛЮС», 2020.С. 122-135.

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

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

44. О. Смірнов, Є. Деменко, О. Онікійчук, А. Арищенко, Л. Горбачова, «Формування псевдовипадкових послідовностей для приховування даних в зображеннях» Комп'ютерні науки та кібербезпека. № 4. С. 30-37. 2019.

45. Смірнов О.А., Конопліцька-Слободенюк О.К., Смірнов С.А., Буравченко К.О., Смірнова Т.В. Поліщук Л.І. Проектування комп'ютерних систем та мереж. Навчальний посібник – Кропивницький: вид. Лисенко В.Ф. 2019. – 264 с.

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

47. Смірнов О.А., Дрєєва Г.М. Метод генерування фрактального трафіку за допомогою моделі генератора на графі. Монографія: Інформаційна безпека та інформаційні технології : монографія / за заг. ред. В. С. Пономаренка. – Х. : Вид. Рожко С.Г. 2019. С. 123-139

48. Дрєєва Г.М., Смірнов О.А., Дрєєв О.М. Метод генерування фрактальноподібної числової послідовності на основі скінченного автомату для моделювання трафіку у мережі. Центральноукраїнський науковий вісник. Технічні науки. № 1(32). с. 173-183, 2019.

49. Смірнов О.А., Кавун С.В., Коваленко О.В., Дрєєв О.М. Мережні інформаційні технології. Навчальний посібник – Кіровоград: РВЛ КНТУ, 2016. – 159 с.

50. Смірнов О.А., Смірнов С.А. Дідик А.К., Дрєєв О.М. Моделі системи нейромережевих експертів безпечної маршрутизації у хмарних антивірусних системах. Збірник наукових праць "Системи обробки інформації". - Випуск 3 (140). - Х.: ХУПС - 2016. - С. 36-39.

51. Смірнов О.А., Кавун С.В., Коваленко О.В., Доренський О.П., Дрєєв О.М., Вялкова В.І. Комп'ютерні мережі. Навчальний посібник – Кіровоград: РВЛ КНТУ, 2016. – 233 с.

52. Смірнов О.А., Дрєєв О.М. Порівняння бітових щільностей при використанні різних методів кодування інформації. Збірник наукових праць "Системи обробки інформації". - Випуск 2 (118). т.2. - Х.: ХУПС - 2014. - С. 64-67

53. Смірнов О.А., Дрєєв О.М. Порівняння бітових щільностей при використанні різних методів кодування інформації. Збірник тез VI міжнародної науково-практичної конференції "Проблеми та перспективи розвитку ІТ-індустрії". м. Харків. 17-18 квітня 2014р. – Харків: ХНЄУ. - 2014. - С. 240.