

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 інтелектуальної обробки
відеоінформації для huddle rooms”

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

Керівник проекту
кандидат технічних наук
_____ Лисенко І.А.
« ____ » _____ 2026 р.
Рецензент _____

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

ЗАТВЕРДЖУЮ

Завідувач кафедри

д.т.н., проф.

Олексій СМІРНОВ

« 15 » січня 2026 року

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

Горячківському Кирилу Ігоровичу

(прізвище, ім'я, по батькові)

1. Тема роботи *Програмне забезпечення системи інтелектуальної
обробки відеоінформації для huddle rooms*

2. Керівник роботи *Лисенко Ірина Анатоліївна, канд. техн. наук*

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

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

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

4. Мета та завдання випускної кваліфікаційної роботи: *Метою роботи є розробка
програмного забезпечення системи інтелектуальної обробки відеоінформації для
huddle rooms*

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 « 15 » січня 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
« 15 » січня 2026 р.

Підпис керівника

Лисенко І.А.
(прізвище та ініціали)

Завдання прийнято до виконання
« 15 » січня 2026 р.

Підпис здобувача

Горячківський К.І.
(прізвище та ініціали)

АНОТАЦІЯ

Горячківський К.І. Програмне забезпечення системи інтелектуальної обробки відеоінформації для huddle rooms. 122 Комп'ютерні науки. Центральноукраїнський національний технічний університет. Кропивницький. 2026.

В даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи інтелектуальної обробки відеоінформації для huddle rooms.

Метою розробки є програмне забезпечення системи інтелектуальної обробки відеоінформації для huddle rooms.

Результат роботи – програмна реалізація системи інтелектуальної обробки відеоінформації для huddle rooms.

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 Visual C#.

Ключові слова: комп'ютерні науки, інтелектуальна обробка відеоінформації, huddle rooms

ABSTRACT

Horiachkivskyi K.I. Software for the intelligent video information processing system for huddle rooms. 122 Computer Science. Central Ukrainian National Technical University. Kropyvnytskyi. 2026.

In this final qualification work for the first (bachelor's) level of higher education, software has been developed, which is intended for the intelligent video information processing system for huddle rooms.

The purpose of the development is the software for the intelligent video information processing system for huddle rooms.

The result of the work is the software implementation of the intelligent video information processing system for huddle rooms.

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 OS Windows 10/11.

The program was developed in the Visual C# environment.

Keywords: computer science, video processing, huddle rooms

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ	2
ВСТУП.....	3
1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ	5
1.1 Призначення системи.....	5
1.2 Область застосування.....	6
2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ	11
2.1 Огляд існуючих систем, технологій, архітектур та програмних рішень за профілем теми випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти.....	11
2.2 Обґрунтування вибору засобів для побудови системи та мови програмування.....	15
2.3 Розгорнута постановка завдання	18
3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ	20
3.1 Опис функціонування системи	20
3.2 Розробка структурної схеми.....	28
3.3 Розробка функціональної схеми	31
3.4 Розробка діаграми процесів.....	34
4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ.....	37
4.1 Розробка блок-схем та опис алгоритмів функціонування системи.....	37
4.2 Захист розробленого програмного забезпечення.....	52
5 ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ	58
6 ОСНОВНІ ВИСНОВКИ.....	63
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	65

					ВКРБ-122.26.0009.00.00.ПЗ			
Вим.	Арк.	№ докум.	Підп.	Дата	Програмне забезпечення системи інтелектуальної обробки відеоінформації для huddle rooms	Літ.	Аркуш	Аркушів
Розроб.		Горячківський К.І.				Б		71
Перев.		Лисенко І.А.					1	
Н.контр.		Коваленко А.С.				ЦНТУ КН-22		
Затв.		Смірнов О.А.						

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

ДКП	—	дискретне косинусне перетворення
ПЗ	—	програмне забезпечення
ЛОМ	—	локальна обчислювальна мережа
CCTV	—	система замкнутого телебачення
IP	—	інтернет протокол
USB	—	універсальна серійна шина

К6ПЗ_2026

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		2

ВСТУП

Актуальність теми. Популярність і масовий попит на оснащення невеликих переговорних – так звані «акваріуми» – багато в чому обумовлені їхньою невисокою вартістю. У мінімальній комплектації – без відеопанелі, але із пристроєм для спільної роботи – вартість складе порядку півтори тисяч доларів, що цілком по засобах навіть невеликої компанії. Ефективна робота сучасних компаній немислима без підтримки постійних контактів з розподіленою мережею філій і представництв, а також з безліччю партнерів і замовників. Часто ці контакти мають вигляд незапланованих нарад невеликих груп фахівців. Для спілкування один з одним всім учасникам бізнес-процесу необхідні комунікаційні засоби, такі як відео-конференц-зв'язок. При цьому нерідко виникає потреба продемонструвати дані з екранів своїх електронних пристроїв віддаленим учасникам. Для подібних зустрічей оптимальні невеликі приміщення, які в англomовному світі одержали назву huddle rooms. У перекладі «huddle» – це група, скупчення. У цього слова є ще одне значення, і воно досить точно передає призначення таких переговорних кімнат. Це нарада футболістів на поле, коли треба швидко, прямо під час гри знайти нове рішення, скорегувати план дій. У нашій країні за такими кімнатами закріпився термін «акваріуми».

Сьогодні основні замовники подібних рішень – це компанії з великим штатом, розподіленою мережею філій і зі співробітниками «у полях», що прагнуть збільшити ефективність роботи й продуктивність праці. Серед найбільш активних покупців – швидко зростаючі комерційні підприємства, де важливе інтелектуальна складова й де доводиться приймати безліч рішень і проробляти великі обсяги інформації, причому щораз це можуть бути різні групи фахівців.

Мета й завдання дослідження. Метою роботи є програмне забезпечення системи інтелектуальної обробки відеоінформації для huddle rooms.

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		3

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

- Огляд існуючих систем інтелектуальної обробки відеоінформації для huddle rooms.
- Дослідження системи інтелектуальної обробки відеоінформації для huddle rooms.
- Програмна реалізація системи інтелектуальної обробки відеоінформації для huddle rooms.

Практична цінність отриманих результатів полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі інтелектуальної обробки відеоінформації для huddle rooms.

Таким чином, виходячи з вищеперерахованого, програмне забезпечення системи інтелектуальної обробки відеоінформації для huddle rooms, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти.

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		4

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Свої споживчі переваги користувачі переносять і в середу-середовище-бізнес-середовище. Відповідно до недавнього опитування Redshift Research, 87% молодих респондентів зволіли б працювати не в традиційній офісній обстановці, а там, де активно використовуються відеотехнології. І ці очікування компанії враховують при плануванні своїх приміщень і оснащенні їхніми відеокомунікаційними засобами.

Сьогодні сам термін «відео-конференц-зв'язок» здається застарілою, оскільки припускає використання цього виду комунікації переважно для проведення зборів і нарад. Однак з поширенням практики віддаленої роботи й розширенням потреб у гнучкому формуванні проектних команд відеозв'язок стає інтегральною частиною повсякденних бізнес-процесів. До того ж розпечені технологічним прогресом користувачі хочуть мати не просто можливість взаємодії, а якісний багатофункціональний інструмент.

Це проявляється, зокрема, і в тому бумі, що ми спостерігаємо на ринку рішень для оснащення невеликих переговорних на 5–6 чоловік (huddle room). Застосування виділених відеотерміналів як і раніше залишається пріоритетним способом задоволення висунутих вимог до якості зв'язку, функціональності, сумісності, простоті й зручності використання й т.д. Необхідне встаткування можна придбати за півтори-дві тисячі доларів – за ціною гарного ноутбука.

Як наслідок, незважаючи на негативні прогнози щодо перспектив попиту на встаткування для ВКЗ на тлі поширення хмарних сервісів, що відповідає сегмент поки відчуває себе непогано (так, в Wainhouse Research заявили про річний ріст продажів). Причому це стосується не тільки нашої країни, де хмарні

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		5

сервіси ВКЗ поки не одержали визнання користувачів, але й світового ринку в цілому.

Все більшу конкуренцію терміналам, у тому числі при оснащенні малих переговорних, становлять рішення на базі ПК. Однак зробити відеозв'язок дійсно масовим й повсюдно доступним обіцяє технологія WebRTC, що нарешті-то одержала потужний імпульс завдяки підтримці з боку таких компаній, як Google, Apple, Microsoft, Firefox і Cisco. Вона інтегрована в сучасні браузерери, і для участі у відеоконференції тепер не треба завантажувати ніяких додатків – досить посилання.

Втім, зміни стосуються не тільки способів організації відеозв'язку. Технології віртуальної й доповненої реальності дозволять домогтися того, що не вдалося повною мірою реалізувати імерсивним системам, – забезпечити ефект присутності, коли всі учасники зустрічі спілкуються так, начебто перебувають в одному приміщенні. Наприклад, у рамках проекту Holoportations Microsoft представила бета-версію технології відеозв'язку, за допомогою якої зображення співрозмовників представляються у вигляді голограм. Звичайно, поки ця справа «не настільки віддаленого майбутнього», але з урахуванням бурхливого пожвавлення на ринку AR/VR воно може наступити швидше, ніж ми очікуємо.

Подальші перспективи розвитку відеотехнологій корисно оцінювати хоча б для того, щоб не помилитися при виборі рішень для впровадження.

1.2 Область застосування

Оснащення «акваріумів»

Набір типових пристроїв для встаткування «акваріумів» невеликий. Це екран, досить великий, для того щоб можна було вивести на нього обговорювану інформацію – графіки, тексти, презентації, відео; проста система ВКЗ, щоб можна було бачити віддаленого співрозмовника й спілкуватися з ним. Часто як сервер

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		6

для ВКЗ використовується ноутбук або смартфон із програмними продуктами на зразок Skype, CiscoWEBex, TrueConf і т.д.

Бажано, щоб переговорна була оснащена якісною відеосистемою Full HD. Для захвата відео буде потрібно камера – наприклад, Lumens з підключенням по USB 3.0 і HDMI-інтерфейсом для відео-конференц-зв'язку. Для аудіо – настільний USB-спікерфон на зразок Yamaha YVC-300. Обоє ці пристрої автоматично визначаються комп'ютером або ноутбуком, після чого їх просто необхідно вибрати в налаштуваннях звуку й відео.

Одне із частих вимог замовників – підтримка концепції BYOD, щоб у ході ділового заходу можна було ділитися контентом зі свого особистого портативного пристрою.

Крім того, в «акваріумах» нерідко застосовують рішення, які дозволяють записувати коментарі й зауваження або на своєму пристрої, або прямо на екрані за допомогою інтерактивних панелей. Багато виробників пропонують РК-накладки, які перетворюють існуючий монітор в інтерактивну панель.

Проектори в офісах швидко й рішуче витісняються РК-моніторами й світлодіодними стінами. Так, для якісного відтворення відео Full HD можна рекомендувати монітори Sharp з убудованим медіаплеєром і широкими кутами огляду по вертикалі й горизонталі.

Для більше вимогливих інсталяцій, де необхідно забезпечити високу якість не тільки відео, але й аудіо, незамінні такі продукти, як медіасервери Biamp і просунуті мережні мікрофонні масиви Shure Microflex Advance. Медіасервер Biamp Tesira Forte підключається до локальної мережі Ethernet і збирає аудіодані з безлічі пристроїв і з безлічі переговорних. А керування, обробка й розподіл даних здійснюються із центра моніторингу або дистанційно через програмний інтерфейс. Поява подібних мережних пристроїв відбиває ще одну тенденцію – злиття ІТ і аудіовідеотехнологій.

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		7

Це може собі дозволити кожний

Інтерес до «акваріумів» росте в усьому світі. У процес їхнього створення ввімкнулися провідні виробники аудіовідеообладнання. На минулій у лютому 2018 року виставці Integrated Systems Europe всі великі виробники представили безліч новинок саме для невеликих переговорних кімнат.

Популярність і масовий попит на «акваріуми» багато в чому обумовлені їхньою невисокою вартістю. У мінімальній комплектації, без відеопанелі, але із пристроєм для спільної роботи, вартість складе порядку півтори тисяч доларів, що цілком по засобах навіть невеликої компанії.

Крім того, з оснащенням huddle room у стані впоратися будь-який інтегратор, що спеціалізується на побудові інформаційних і мультимедійних систем. Досить навчити інженера, що буде надалі клонувати типові проекти переговорних, оскільки ці рішення прості в установці й налаштуванні й мають дружній інтерфейс.

В Україні ринок рішень для невеликих переговорних усе ще перебуває в процесі формування, масовий інтерес до цих систем тільки зароджується. Зрозуміло, про якесь насичення ринку говорити поки рано.

Сьогодні замовлення на реалізацію подібних проектів найчастіше надходять від великих комерційних компаній, мережних фірм, банків, підприємств різних галузей, страхових і торговельних компаній.

Звичайно в одному будинку обладнають трохи невеликих, добре оснащених переговорних, які поєднується в єдину мережну структуру прийняття рішень.

Державні структури в Україні також починають користуватися цими простими рішеннями. Наприклад, в «акваріумах» проходять наради по ВКЗ між керівництвом шкіл і представниками районних відділів народного утворення.

Фахівці відзначають ріст звернень клієнтів, які вже використовували прості переговорні з телевізором і фліп-чартом, і хочуть переоснастити

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		8

«акваріуми» більше сучасними комунікаційними пристроями, забезпечивши в тому числі реалізацію концепції BYOD.

Рішення для huddle room швидко завойовують ринок. Їхня популярність обумовлена тим, що бізнес став більше гнучким і динамічним. Якщо раніше співробітники неквапливо збиралися в просторах приміщеннях, для того щоб вислухати годинну доповідь, проголосувати, прийняти рішення, то зараз багато компаній намагаються обладнати кілька маленьких «акваріумів», розуміючи, що набагато ефективніше проводити зустрічі винятково в справі – для обговорення конкретного питання й оперативного реагування.

Високоякісний звук для ВКЗ

Компанія Viamp спеціально для невеликих переговорних створила рішення Devio, до складу якого входять: високоякісний мікрофон зі змінюваною діаграмою спрямованості, що активно реагує на зміну акустичної обстановки; пристрій, до якого підключаються мікрофон, комп'ютер, USB-камера; система цифрової обробки звуку (придушення зворотного зв'язка, підвищення розбірливості мови, шумозаглушення). Користувачеві не потрібно набудовувати звукову систему, вона сама після підключення автоматично підбере характеристики відповідно до акустичних параметрів приміщення.

Для переговорних компанія Shure, відомий виробник мікрофонів, випустила серію мікрофонних масивів Microflex Advance, які монтуються в стельові плити або встановлюються на столі. Це складні пристрої із цифровими процесорами, які формують вузькі діаграми спрямованості, що дозволяє точно «захопити» мову й відітнути сторонні шуми, навіть якщо людина переміщається по приміщенню, якими б при цьому ні минулому акустичні параметри «акваріума».

Мікрофонні масиви Shure легко вбудовуються в мережну інфраструктуру, їх можна без праці підключити за протоколом Dante до комп'ютерної мережі й настроїти дистанційно. Живлення на масиви подається по PoE, устаткування може бути повністю приховано в обстановці. І самому головне – кінцевому

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		9

користувачеві не прийдеться розбиратися в роботі складного встаткування: усе буде зроблено автоматично, так що йому не буде потрібно міняти гучність акустичної системи, підносити мікрофон близько до рота або натискати кнопки на пультах. Це дозволяє максимально продуктивно використовувати час ділової зустрічі.

Таким чином, виходячи з вищеперерахованого, програмне забезпечення системи інтелектуальної обробки відеоінформації для huddle rooms, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти.

К6ПЗ_2026

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		10

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

Системи відео-конференц-зв'язку

Спілкування між людьми в сучасному суспільстві стрімко переходить на новий рівень. Тепер для комунікацій використовуються новітні технології й рішення, що дозволяють створити високоякісний зв'язок між абонентами на будь-якій відстані. Яскравим прикладом є система відео-конференц-зв'язку.

Відео-конференц-зв'язок – це специфічна технологія комунікації, що забезпечує взаємодію між певним числом учасників, а також обмін у реальному часі відео- і аудіоданими.

Існує кілька основних типів систем відео-конференц-зв'язку:

- Персональні або індивідуальні системи відеоконференцій – забезпечують комунікацію між двома користувачами в режимі реального часу.
- Групові системи відео-конференц-зв'язку – забезпечують зв'язок між більшою кількістю учасників, які можуть перебувати в різних куточках миру.
- Мобільні системи відеоконференцій – специфічний тип, що забезпечує комунікацію між учасниками в нестандартних місцях. Відмітна риса – мобільність. Варто врахувати, що даний тип конференц-систем вимагає наявності спеціального дорогого устаткування. Мобільні системи знайшли широке застосування у військовій справі, медицині, телебаченні й ін.
- Галузеві системи відео-конференц-зв'язку – застосовуються в певних галузях, наприклад, медицині, будівництві, судовій практиці й т.д. Відмітною рисою даного типу систем є індивідуальність, тому що використовуються вони в специфічних цілях.

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		11

Програмні й апаратні рішення для систем відео-конференц-зв'язку

Відеоконференції – це дуже зручний вид комунікації для великої кількості учасників. Використовуючи його, можна заощадити багато часу, сил і матеріальних засобів. Наприклад, можна позбутися від нестатку переїжджати на тривалі відстані, щоб потрапити на важливу нараду й т.п. Але для створення якісної системи необхідно використовувати специфічні програмні й апаратні рішення, без яких здійснення задуманого буде неможливим.

Програмні рішення для відео-конференц-зв'язку

Програмні рішення – це обробка різної інформації усередині системи за допомогою програмного забезпечення. Наприклад, процеси кодування/декодування аудіо й відео. Можна виділити кілька типів програмних рішень: програмні сервери відео-конференц-зв'язку й клієнтські додатки. Головна перевага програмних рішень – універсальність і простота масштабування (немає необхідності в придбанні додаткового дорогого встаткування).

TrueConf Server

Яскравим прикладом програмного сервера відео-конференц-зв'язку є TrueConf Server. Дане програмне забезпечення дозволяє створювати масштабні відеоконференції (до 250 учасників) як по мережі "інтернет", так і в закритій локальній мережі. Використання даного продукту гарантовано спричинить економію матеріальних засобів і величезні можливості для різних організацій. Головна перевага – установка програмного забезпечення на власний сервер, а також можливість його докладного індивідуального налаштування.

TrueConf Online

TrueConf Online – це чудовий клієнтський додаток для здійснення відеодзвінків і проведення відеоконференцій. Зв'язок між учасниками в даному додатку здійснюється через інтернет. Високопродуктивні хмарні сервера забезпечують найвище якості передачі звукового й відео сигналів. Додаток TrueConf Online доступно для більшості сучасних платформ, а це дає можливість використовувати його як на ПК, так і на мобільному гаджеті.

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		12

Апаратні рішення для відео-конференц-зв'язку

Без надійного встаткування створення якісної відеоконференції просто неможливо, а знайти його буває нелегко. Інтернет-магазин "UnitSolutions" пропонує вам відмінні апаратні рішення, які допоможуть створити відеоконференцію високого рівня.

Рішення для відео

Для захвату й передачі відеосигналу під час конференції ми рекомендуємо два типи камер – web-камери й PTZ-камери (поворотні). Перші використовуються для передачі зображення одного або кількох людей, які містяться в кадр. Веб-камери мають невелику фокусну відстань, тому ідеально підходять для застосування в тих випадках, коли учасники перебувають поблизу від пристрою. Другі використовуються при проведенні масштабних конференцій у залах, коли необхідно перенаправляти фокус на виступаючого. Вони мають більшу фокусну відстань, що забезпечує передачу якісного зображення учасників, що перебувають удалині від пристрою. Також ми рекомендуємо задіяти пульт керування поворотними камерами, що дозволить контролювати процес зйомки більш чітко. Всі вони мають власні технічні й конструктивні особливості, що дозволить підібрати оптимальне встаткування для кожного конкретного випадку.

Якщо вас цікавить запис важливих відеоконференцій, рекомендуються карти захвату, здатні забезпечити висока якість зображення записаного відео. Вони сумісні як із ПК і ноутбуками, так і з багатьма моделями PTZ-камер. У нашій магазині ви можете відразу купити комплект, що складається з PTZ-камери й карти захвату відеосигналу.

Захват і передача голосу

При проведенні конференцій дуже важливої складової є чітка передача голосу виступаючого. Для здійснення даного завдання ми рекомендуємо мікрофони для конференцій. Представлені як вузьконаправлені моделі мікрофонів, що мають місце при передачі голосу одного учасника, так і

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		13

широконаправленні, які мають більшу чутливість і здатні захоплювати голос учасників, що перебувають удалині від пристрою. У нас ви знайдете як провідні, так і бездротові моделі. Але також є альтернативний і більше універсальний варіант комплектації апаратури. Для масштабних конференцій, що передбачають голосування, ми рекомендуємо використовувати пульт голови й пульт делегата. Ці пристрої оснащені мікрофоном (вузьконаправленим) і сумісні з певними типами систем для проведення відеоконференцій. Складові кожної з них представлені в магазині. Такий варіант є більше простим і зручним у застосуванні. Особливо, якщо це бездротова система ВКЗ. А в тому випадку, коли виступаючий перебуває в постійному русі й має потребу у вільних руках, ми рекомендуємо оптимальне рішення – бездротові гарнітури.

Відтворення й регулювання звукового сигналу

Колонки, призначені для відтворення голоси й мінімальний музичний супровід, відрізняються від спеціалізованих музичних. Вимоги до таких колонок дуже строгі, тому що відтворений звук повинен був чітким, насиченим і максимально наближеним до реальності. Рекомендується використовувати спеціальні колонки для конференцій. У нас ви знайдете як портативні, так і настінні варіанти. Колонки невеликої потужності звичайно йдуть із убудованим підсилювачем і призначені для маленьких приміщень. А якщо вам потрібно забезпечити насичений і голосний звук у великому залі, варто використовувати могутніші колонки. У такому випадку ми раді порекомендувати відмінні підсилювачі потужності мікшери (мікшерні пульти). Перші необхідні для того, щоб підсилити сигнал до потужності, відтвореної колонками. Другі – для більше ретельного налаштування звукового супроводу (підстроювання громкостей і частот кожного окремого каналу).

Універсальне встаткування

У якості універсального аудіо-рішення ми раді рекомендувати спікерфони – пристрою, які поєднують у собі динаміки відтворення й мікрофон. У нас представлені самі актуальні й популярні моделі. Чутливий мікрофон і

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		14

голосні динаміки, об'єднані в спікерфоні, здатні забезпечити високий комфорт при проведенні відеоконференцій через популярні VoIP-додатки, наприклад, Skype, TrueConf Online і ін. Так само для комфортного спілкування групи людей з віддаленим телефонним співрозмовником по голосному зв'язку використовуються конференц-телефони.

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

Програмне забезпечення написане мовою Visual C#. Ця мова обрана виходячи з наступних міркувань. Visual C# – строго типізована об'єктно-орієнтована мова, призначена для розробки різноманітних безпечних і потужних додатків, виконуваних у середовищі .NET Framework. Мовою Visual C# можна розробляти звичайні клієнтські додатки Windows, веб-служби XML, розподілені компоненти, додатки типу “ сервер-клієнт”, додатки баз даних і багато яких інших. В Visual C# є розширений редактор коду, конструктори зі зручним користувацьким інтерфейсом, вбудований відладник і багато інших засобів, покликані спростити розробку додатків мовою Visual C# версії 5.0 і .NET Framework версії 4.5.

Синтаксис Visual C# дуже виразний, але простий у вивченні. Усі, хто знаком з мовами C, C++ або Java з легкістю визнають синтаксис із фігурними дужками, характерний для мови Visual C#. Розроблювачі, що знають кожен із цих мов, як правило, зможуть домогтися ефективної роботи з мовою Visual C# за дуже короткий час. Синтаксис Visual C# робить простіше те, що було складно в C++, і забезпечує потужні можливості, такі як типи значень Nullable, перерахування, делегати, лямбда-вираження й прямий доступ до пам'яті, чого немає в Java. Visual C# підтримує універсальні методи й типи, забезпечуючи більше високий рівень безпеки й продуктивності, а також ітератори, що дозволяють при реалізації колекцій класів визначати власне поводження ітерації,

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		15

що може легко використовуватися в клієнтському коді. В Visual C# 5.0 вираження LINQ (Language-Integrated Query) роблять строго-типізований запит першокласною конструкцією мови.

Як об'єктно-орієнтована мова, Visual C# підтримує поняття інкапсуляції, спадкування й поліморфізму. Всі змінні й методи, включаючи метод **Main** – крапку входу додатка – інкапсулюється у визначення класів. Клас може успадковувати безпосередньо з одного родового класу, але може реалізовувати будь-яке число інтерфейсів. Для методів, які перевизначають віртуальні методи в батьківському класі, необхідно ключове слово **override**, щоб виключити випадкове повторне визначення. У мові Visual C# структура схожа на полегшений клас: це тип, що розподіляється по стопках, що реалізує інтерфейси, але не підтримує спадкування.

На додаток до основних описаних об'єктно-орієнтованих принципів, мова Visual C# спрощує розробку компонентів програмного забезпечення завдяки декільком інноваційним конструкціям мови, у число яких входять наступні:

- Інкапсульовані підписи методів, називані делегатами, які підтримують строго-типізовані повідомлення про події.
- Властивості, що виступають у ролі методів доступу для закритих змінних-членів.
- Атрибути з декларативними метаданими про типи під час виконання.
- Вбудовані коментарі XML-документації.
- LINQ (Language-Integrated Query), що пропонує вбудовані можливості запитів у різних джерелах даних.

Якщо буде потрібно забезпечити взаємодію з іншим програмним забезпеченням Windows, таким як об'єкти COM або власні бібліотеки DLL Win32, у мові Visual C# можна використовувати процес, що називається "Interop". Процес Interop дозволяє програмам на Visual C# виконувати практично будь-які дії, які може виконувати вихідний додаток на C++. Мова Visual C# підтримує

навіть покажчики й поняття "небезпечного" коду для тих випадків, коли прямий доступ до пам'яті має вкрай важливе значення.

Процес побудови Visual C# у порівнянні з C і C++ простий і є більше гнучким, чим в Java. Немає окремих файлів заголовка, а методи й типи не потрібно повідомляти в певному порядку. У вихідному файлі Visual C# може бути визначене будь-яке число класів, структур, інтерфейсів і подій.

Архітектура платформи .NET Framework

Програма мовою Visual C# виконується в середовищі .NET Framework – інтегрованому компоненті Windows, що містить віртуальну систему виконання (середовище CLR) і уніфікований набір бібліотек класів. Середовище CLR являє собою комерційну реалізацію корпорацією Майкрософт інфраструктури CLI, що є міжнародним стандартом, який лежить в основі створення середовищ виконання й розробки, у яких забезпечується тісна взаємодія між мовами й бібліотеками.

Вихідний код, написаний мовою Visual C#, компілюється в проміжну мову (IL) у відповідності зі специфікацією CLI. Код IL і ресурси, такі як растрові зображення й рядки, зберігаються на диску у файлі, що виконується, названому складанням, з розширенням EXE або DLL у більшості випадків. Складання містить маніфест із відомостями про типи складання, версії, мови й регіональні параметри та вимоги безпеки.

При виконанні програми на Visual C# складання завантажується в середовище CLR залежно від відомостей у маніфесті. Далі, якщо вимоги безпеки дотримані, середовище CLR виконує JIT-компіляцію для перетворення коду IL в інструкції машинного коду. Середовище CLR також надає інші служби, що відносяться до автоматичного збору сміття, обробки виключень і керуванню ресурсами. Код, виконуваний середовищем CLR, іноді називають "керованим кодом" у протиставлення "некерованому коду", що компілюється в машинний код, призначений для певної системи. Далі показані відносини під час компіляції

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		17

й час виконання між файлами з вихідним кодом Visual C#, бібліотеками класів .NET Framework, складаннями й середовищем CLR.

Взаємодія між мовами є ключовою особливістю .NET Framework. Оскільки код IL, створюваний компілятором Visual C# відповідає специфікації CTS, код IL на основі Visual C# може взаємодіяти з кодом, створюваним версіями мов Visual Basic, Visual C++, Visual J# платформи .NET Framework і ще більш ніж 20 CTS-сумісних мов. В одному складанні може бути кілька модулів, написаних на різних мовах платформи .NET Framework, і типи можуть посилатися один на одного, як якби вони були написані на одній мові.

Крім служб часу виконання, в.NET Framework також є велика бібліотека, що складається з більш ніж 4000 класів, організованих по просторах імен, які забезпечують різноманітні корисні функції для будь-яких дій, починаючи від введення й виведення файлів для керування рядками для розбивки XML, і закінчуючи елементами керування Windows Forms. У звичайному додатку мовою Visual C# бібліотека класів .NET Framework інтенсивно використовується для "устрою" коду.

2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випускню кваліфікаційну роботу за першим (бакалаврським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи інтелектуальної обробки відеоінформації для huddle rooms.

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.

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		18

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) сформулювати висновки про виконаний обсяг робіт та одержані результати.

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		19

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

Якщо раніше виробники орієнтувалися на верхній сегмент ринку, роблячи ставку на іммерсійні й кімнатні групові системи, то в останні кілька років вони усе більше уваги приділяють рішенням для оснащення невеликих переговорних.

Технології сприяють розвитку демократичних процесів у не меншому ступені, чим дії політиків: те, що колись було прерогативою високого начальства, тепер доступно й рядовим співробітникам. Подібна демократизація відбулася за останні 5-10 років у сфері відео-конференц-зв'язку: і якщо раніше виробники орієнтувалися на верхній сегмент ринку, роблячи ставку на іммерсійні й кімнатні групові системи, то в останні кілька років вони усе більше уваги приділяють рішенням для оснащення невеликих переговорних.

Аналітики прогнозують масовий попит на рішення по оснащенню системами відео-конференц-зв'язку (або, як тепер прийнято говорити, співробітництва) невеликих переговорних (huddle rooms). Потенційна ємність цього ринку набагато перевершує обсяг ринку групових систем. Таких приміщень, за різними оцінками, у п'ять-десять разів більше, ніж конференц-залів. При цьому тільки 10% традиційних переговорних оснащені засобами відеозв'язку, а про інші й говорити не доводиться – поле непахане.

На чому ж засновані очікування виробників і що собою представляють ці рішення?

Переговорна для маленької компанії

Пройшов той час, коли компанії воліли оснащувати один-два конференц-залу по останньому слову техніки. При високій орендній платі це стає недозволеною розкішшю, тим більше що такі приміщення не завжди використовуються ефективно: більшу частину часу вони або зовсім не

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		20

затребувані, або заповнюються не повністю. У результаті середній розмір приміщень для проведення нарад скорочується: тенденція розгорнулася у бік оснащення засобами для організації конференцій невеликих переговорних, тим більше що витрати на них знижуються завдяки появі цільових рішень (див. наступний розділ). Але це лише один з факторів наростаючого буму.

З поширенням методології гнучкої (agile) розробки організації перебудовують свої процеси й інфраструктуру, щоб забезпечити її підтримку. Це впливає на дизайн нових офісних просторів, де передбачається безліч окремих приміщень для невеликих (до 5-6 учасників) команд фахівців, у яких вони могли б обговорити поточні питання. В англійській пресі такі переговорні одержали назву huddle room, що можна перекласти як «кімната для летючок». Цей термін указує на неформальний характер зустрічей, порівнянний з розмовою біля кавашини й у курилці, де люди «кучкуються» (huddle).

Площа типової huddle room, по даним Magnus Commercial Real Estate, становить 10–25 м² (хоча, на нашу думку, розміри трохи перебільшені). Для створення робочої обстановки встановлюються тільки необхідні меблі – стіл і стільці, тим більше що й розгорнутися особливо ніде. Устаткування теж найпростішому – це завжди компроміс між функціональністю й ціною.

Популярність невеликих переговорних зростає завдяки гнучкості їхнього використання: при наявності декількох приміщень попереднє бронювання не потрібно, тому команда може зібратися в будь-який момент. Проста обстановка робить зустрічі менш формальними, що сприяє їхній ефективності, оскільки кожний учасник одержує можливість висловитися й утягується в обговорення.

Однак, щоб взаємодія була продуктивним, віддаленим учасникам переговорів повинні бути надані всі необхідні комунікаційні засоби й інструменти для спільної роботи, насамперед з контентом.

Щоб оснастити переговорну системою відео-конференц-зв'язку, ще кілька років назад треба було витратити кілька десятків, а те й сотню тисяч доларів (якщо мова йшла про імерсивних системи з ефектом присутності). У результаті

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		21

невеликі переговорні залишалися без відеозасобів – у найкращому разі встановлювалися спікерфон і монітор, на якому показувалися презентації.

Тепер для оснащення переговорної на п'ять-шість чоловік сучасними засобами відеозв'язку знадобиться менше 5 тис. доларів, а якщо обмежуватися необхідним мінімумом, можна обійтися навіть півтора-двоми тисячами. Багато виробників пропонують спеціальні відеотермінали для таких переговорних (наприклад, Cisco SX10, Polycom Debut, Lifesize Icon Flex і ін.) вартістю одна-дві тисячі доларів. Завантаживши додаток на планшет або смартфон і підключившись по Wi-Fi, користувачі одержать можливість обмінюватися контентом.

Масовий ринок привертає увагу безлічі виробників, тому вже пропонуються відеотермінали, ціна яких істотно менше 1000 доларів. Якщо в переговорної вужі встановлені монітор або телевізор, то цією сумою можна обмежитися – досить підключити відеотермінал до екрана. Однак для організації якісного відео-конференц-зв'язку й повноцінної командної взаємодії такого мінімального варіанта, швидше за все, виявиться недостатньо (див. далі розділ про оснащення малих переговорних).

Одним з переваг установки систем ВКЗ у малі переговорні є можливість проведення зустрічей ad hoc («з нагоди») без попереднього бронювання приміщення. Для цього відеозасобами повинне бути оснащене достатня кількість переговорних, але тоді великим організаціям, де десятки й сотні переговорних не рідкість, прийде витратити велику суму. Таким чином, ціна як і раніше залишається істотним фактором, а порядок витрат приблизно тим же, правда вигода від впровадження може виявитися помітно вище завдяки мережному ефекту.

Багатобічну конференцію сьогодні багато хто проводять взагалі без спеціального устаткування. Так, програмне забезпечення ezTalks Meeting можна використовувати для організації конференцій до 100 учасників, причому безкоштовно. Воно підтримує досить розвинені засоби спільної роботи:

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		22

демонстрацію екрана, інтерактивну панель, планування й запис зустрічей і т.д. Якщо не влаштовує якість убудованої в ноутбук або смартфон камери, можна придбати недорогу Web-камеру. Навіщо ж купувати щось ще?

Якщо користувачі готові миритися із дрібними незручностями й дрібним зображенням, а компанії не заперечують проти використання такого ПО, подібного оснащення буває досить. Однак хтось повинен виділити свій ноутбук для запуску додатка, що не завжди зручно, оскільки під час зустрічі потрібно працювати, відео може «завмирати» через виконання паралельних процесів і додатків, а на екрані зненацька з'являються приватні повідомлення із чата й т.д. При використанні виділеного встаткування подібних проблем, як правило, не виникає.

Крім названих незручностей, рішенням споживчого рівня властиві й технічні обмеження. Наприклад, кут огляду камери може виявитися недостатнім для захвата всіх учасників зустрічі, що перебувають у кімнаті. При всій важливості якості зображення ключове значення для успіху комунікації має розбірливість мови співрозмовника, а для задовільної передачі звуку, особливо в умовах стороннього шуму, необхідне застосування спеціального устаткування.

Нарешті, забезпечення взаємодії продуктів споживчого класу, що випускаються різними виробниками, може виявитися важкорозв'язним завданням, так що додаткові витрати на придбання рішення корпоративного рівня найчастіше цілком виправдані.

Збери сам

Наступний по витратах варіант – використання рішення, що складає з відеокамери, аудіосистеми й програмного забезпечення на виділеному ПК. Якщо в переговорній немає ТБ, прийде придбати ще й монітор. Цей варіант дозволяє позбутися від ряду проблем, властивих безкоштовним споживчим рішенням, але йому властиві й свої власні.

Використання окремого монітора рятує учасників зустрічі від необхідності юрбитися навколо екрана в спробі побачити віддалених учасників

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		23

конференції або розглянути презентацію. Оптимальним варіантом для малих переговорних вважається монтуємий на стіну РК- або LED-монітор або ТБ діагоналлю 40-50 дюймів (вибір залежить від розмірів приміщення й бюджету).

Якщо в переговорній вужі є ПК, то в більшості випадків його продуктивності буде досить для виконання необхідного програмного забезпечення. У багато готових комплектів, крім камери й спікерфона, входить Intel NUC – цей варіант кращий для невеликих приміщень завдяки компактним розмірам. Як на будь-який Windows-комп'ютер, на нього можна встановити будь-яке необхідне ПО для організації співробітництва.

Деякі виробники, як ми вже відзначали, пропонують готові набори: поряд з камерою, спікерфоном і програмним забезпеченням надається передплата на хмарні сервіси відеоконференцій. Всі складові комплекту можна придбати й окремо, але в кожному разі необхідно звертати увагу на вибір камери й аудіосистеми.

Хочу всіх бачити

На камері краще не заощаджувати, як і на аудіосистемі. Це зовсім не означає, що треба здобувати найдорожчу. Найчастіше досить USB-камери, але Web-камера для домашнього використання не підійде – такі пристрої розраховані на зйомку одного-двох чоловік. Щоб інші учасники не залишалися за кадром, знадобиться широкоугольна камера.

Інший варіант – установка поворотної камери (Pan, Tilt, Zoom, PTZ). Щоб правильно кадриувати зображення, потрібно вміти з нею звертатися, але, як показує практика, користувачі з небажанням роблять будь-які додаткові дії, навіть якщо необхідні навички в них є. У принципі, якщо камера дозволяє, можна задати передвстановлені налаштування, і тоді при натисканні всього однієї кнопки камера стане фіксуватися на промовці.

Щоб позбавити користувачів від зайвих дій, можна придбати камеру із системою відстеження голосу. Така камера автоматично фокусується на промовці,

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		24

але дане рішення навряд чи можна віднести до бюджетних, та й для багатьох сценаріїв використання воно надлишково.

Якщо потрібно тільки загальний план із зображенням всіх учасників, можна встановити кімнатні системи панорамування (room framing system). Вони відслідковують місцезнаходження учасників у приміщенні й автоматично підбудовують параметри камери, щоб усе перебували в кадрі. Такі камери можуть бути як фіксованими, так і поворотними.

У якості USB-Відеокамер для малих переговорних можуть використовуватися, наприклад, ClearOne Unite, Logitech MeetUp або Huddly GO. Автоматичне відстеження по голосі й панорамуванні кадру підтримують відеокамери Polycom Eagle Eye Director, Cisco Spark Room Series і Lifesize Icon 450.

Якщо відеосигнал пропаде, це викличе невдоволення, але зустріч продовжиться. Якщо ж співрозмовника не буде чути, то обговорення прийде перервати (у принципі, його можна перевести в чат, якщо він залишиться доступним, але в такому випадку основна цінність зустрічі – живе спілкування – губиться). Тривала бесіда при нерозбірливості мови здатна викликати лише розчарування, тому аудіосистема повинна бути якісною й надійною.

На ринку представлені відеокамери з убудованими мікрофоном і динаміками, але все-таки це не основна їхня функція, і для якісної передачі відтворення краще встановити окремі мікрофон і колонки. Втім, останні можуть бути інтегровані в спікерфон. У багатьох випадках спікерфон установлюється на столі для засідань. Як приклад можна привести Polycom Trio, Yealink CP960 Conference Phone або ClearOne SHAT 170. Опція, що користується популярністю, – колонки в одному витягнутому корпусі (speakers bar), які можна розмістити під монітором, щоб звук і зображення мовця не розділялися.

В офісі з відкритим плануванням необхідно мати ізольовані приміщення, де можна було б обговорити справи й попрацювати в тиші й без перешкод. Застосування професійних мікрофонів дозволяє провести зустріч із більшим

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		25

комфортом завдяки повному охопту приміщення й усуненню зовнішніх звуків. Однак і цього може виявитися недостатньо, і тоді може знадобитися шумоізоляція для захисту від проникнення шуму ззовні й, відповідно, від ризику підслуховування.

У принципі можна придбати готовий акустичний робітник відітнув (acoustic work pod), призначений для організації переговорної в умовах відкритого офісу. Але ми вже виходимо за рамки теми й передбачуваного бюджету.

Конкретне встаткування, що може знадобитися для оснащення переговорної, залежить від характеристик приміщення й розв'язуваних бізнес-завдань. На ринку є безліч пропозицій, розрахованих на різні потреби й гарантії, так що вибір «правильної» комбінації виявляється навіть більше складний, чим у випадку оснащення конференц-залу. Щоб не помилитися, треба оцінити свої потреби в засобах для групової роботи й підключення віддалених користувачів, а також, звичайно, в аудіо- і відеоустаткуванні.

Щоб скоротити число розглянутих опцій, можна скористатися одним з рішень «усе в одному». У комплекті є все необхідне: виділений кодек, дисплей, камера, інтегрований мікрофон і колонки й кабелі. При цьому користувачам не прийде турбуватися про сумісність компонентів. Крім того, сучасними оптимізованими рішеннями легко користуватися: так, якщо вірити виробникові, навіть початкове налаштування Cisco Telepresence MX300 G2 займає всього 10 хв, а керування реалізується за принципом однієї кнопки.

Простота й зручність використання мають вирішальне значення, інакше встановлені системи виявляються незатребуваними системами. Як показує опитування AVI Systems, біля двох третин опитаних скаржилися на те, що до половини конференц-колів, у яких вони брали участь, виявлялися впусту витраченим часом. Головним чином це було викликано тим, що початок конференції затягувалося через проблеми з підключенням віддалених користувачів.

Ідея надати рішення «усе в одному» для проведення неформальних конференцій знайшла своє закінчене втілення в Polycom RealPresence Centro: усе

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		26

компоненти, включаючи чотири екрани по периметрі, уже змонтовані на одній платформі. Учасники зустрічі можуть розташуватися, де їм зручно, – досить прикати цю пересувну платформу в потрібне місце. Кругова камера дозволяє бачити всіх присутніх завдяки п'яти убудованим об'єктивам, а інтегровані мікрофон і динаміки забезпечують якісний звук.

Щоб робота в невеликих віддалених групах була ефективною, обране рішення повинне забезпечувати можливість підключення з будь-якого пристрою. Але одних тільки комунікаційних інструментів недостатньо – користувачам необхідно виводити на екран різний контент і працювати над ним разом з іншими членами команди: вносити виправлення, робити анотації, зберігати створене й т.п. Багато подібних рішень підтримують відображення контенту з підключеного пристрою при провідному й/або бездротовому з'єднанні, але якої саме функції спільної роботи з контентом реалізовані, краще з'ясувати окремо.

Згідно із широко відомою оцінкою Wainhouse Research, в усьому світі налічується 30-50 млн невеликих переговорних, а також 10 млн конференц-залів і залів для засідань. Оснащення перших обходиться набагато дешевше, а використовуються вони більш інтенсивно. У результаті, за деякими оцінками, число корпоративних користувачів ВКЗ може вирости на порядок, а виходить, неминучий ріст навантаження як на інфраструктуру, так і на служби технічної підтримки.

Постачальники рішень для спільної роботи з використанням відео-конференц-зв'язку прагнуть зробити їх максимально простими й інтуїтивно зрозумілими у використанні. Вони всіляко підкреслюють інтеграцію різних компонентів для спільної роботи. Однак закон більших чисел ніхто не скасовував: навіть якщо відносна частота проблем скорочується, їхня загальна кількість може зрости.

У своєму вмінні що-небудь зламати користувачі уступають, напевно, тільки дітям. Інакше кажучи, зовнішня простота й зручність застосування не скасовують необхідності досконального вивчення ІТ-фахівцями технічних

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		27

особливостей розгорнутих рішень, щоб оперативно діагностувати виниклі проблеми й усувати їх. І тоді малі переговірні не будуть пустувати й повною мірою виправдають покладені на них надії як на інструмент підвищення ефективності командної роботи.

3.2 Розробка структурної схеми

Структурна схема розробленої системи зображена на рисунку 3.2. На ній показано структуру системи інтелектуальної обробки відеоінформації для huddle rooms, та місце у цій системі покадрового кодування та передачі відеоданих на основі ієрархічного кодування.

Як безпосередньо пристрої інтелектуальної обробки відеоінформації для huddle rooms можуть застосовуватися відеокамери, що використовують відеосервер для перетворення відеосигналу в потік даних для передачі. Основна перевага відеосерверів перед мережними відеокамерами – це можливість споживача самостійно вибирати вже зарекомендували себе телевізійні камери під необхідні умови експлуатації (значні перепади освітленості, заднє засвітлення, інтелектуальної обробки відеоінформації для huddle rooms «день-ніч»).

Традиційно системи інтелектуальної обробки відеоінформації для huddle rooms будуються на аналогових технологіях. Відеосервери оцифровують джерело аналогового відеосигналу й передають цифрове відеозображення по мережах передачі даних, тим самим перетворюючи звичайні аналогові відеокамери в мережні. Крім того, відеосервера дозволяють одержати віддалений доступ до відеокамер у режимі реального часу, використовуючи мережі. Відеосервер, підключений до мережі, пропонує широкий спектр можливостей інтелектуальної обробки відеоінформації для huddle rooms, передаючи відеозображення куди завгодно в межах цієї мережі. Незалежно від місця розташування об'єкта інтелектуальної обробки відеоінформації для huddle rooms, будь це громадське місце або виробничі приміщення, доступ до «живого» відео може бути

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		28

отриманий з будь-якої робочої станції в межах комп'ютерної мережі або через Інтернет особами, що мають на той дозвіл.

Усе, що потрібно для оцифровки джерел аналогового відеосигналу й наступної передачі інформації з комп'ютерних мереж, перебуває в одному пристрої – відеосервері. Він може передавати до 25 кадрів/с у стандарті PAL по звичайних комп'ютерних мережах Ethernet. Відеосервер містить у собі один або більше аналогових відеовходів, цифровий перетворювач зображення, мікросхему або програму, відповідальну за компресію зображення, веб-сервер, мережні й послідовні інтерфейси.

Цифрове відео може зберігатися на жорсткому диску, що дозволяє здійснювати швидкий і ефективний пошук потрібних кадрів. Щоб уникнути переповнення жорсткого диска, можна запрограмувати комп'ютер таким чином, що старі зображення будуть автоматично видалятися, звільняючи місце для нових.

Відеосервери дозволяють користувачеві перейти від існуючих аналогових систем інтелектуальної обробки відеоінформації для huddle rooms до цифрових. Один відеосервер може підключати до комп'ютерної мережі до 8 аналогових камер – вигідне рішення для передачі якісного відеозображення по мережі. По суті справи відеосервери виявляються мостом, що з'єднує аналогові й цифрові технології, і дозволяють скористатися вже наявними аналоговими камерами й уникнути проблем з коаксіальними кабелями.

На рисунку 3.1 наведена структурна схема системи інтелектуальної обробки відеоінформації для huddle rooms.

У місцях, де встановлені аналогові відеокамери, сигнал з них надходить спочатку на відеосервер, а вже потім на абонентський радіомодуль для передачі даних на базову станцію. Далі з базової станції використовуючи Bluetooth сигнал надходить на комп'ютер із установленим спеціальним програмним забезпеченням для поділу даних на кілька незалежних картинок, які можуть виводитися як на один великий монітор, так і на групу моніторів.

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		29



Рисунок 3.1 – Структурна схема системи

Як ми бачимо з відеокамери отримуються дані, які кодуються за допомогою розробленого програмного забезпечення покадрового кодування та передачі відеоданих на основі ієрархічного кодування.

Після цього закодоване зображення передається через інтерфейс на основі стандарту 802.15.1 (Bluetooth).

Це відбувається у зв'язку з тим, що камери інтелектуальної обробки відеоінформації для huddle rooms знаходяться у місцях, де під'єднується бездротова мережа.

На приймальній стороні інформація, яка поступає через Bluetooth-коннектор, або декодується й переглядається, або записується на носій інформації.

Стандарт 802.15.1 (Bluetooth) вибрано виходячи з наступних міркувань:

- Гранична швидкість, обумовлена стандартом ZigBee (250 Кбіт/с), істотно обмежує пропускну здатність і недостатня для передачі чіткого зображення.

- Швидкість Bluetooth (1 Мбіт/с) є достатньою й залишає можливість нарощування функціональності.

- У частині витрат на реалізацію стандарт Wi-Fi досить надлишковий, оскільки передбачає реалізацію мережної взаємодії по моделі ISO/OSI. Це надає зручність у роботі із пристроєм на основі стандарту Wi-Fi, але істотно збільшує його вартість.

- Стандарт Bluetooth використовує принцип стрибкоподібної перебудови частоти (FHSS), що підвищує рівень захисту від несанкціонованого доступу й забезпечує більшу завадостійкість у порівнянні з Wi-Fi.

3.3 Розробка функціональної схеми

Функціональна схема розробленої системи зображена на рисунку 3.2.

Зі схеми видно, що розроблена система складається з наступних частин:

1. Припустимі дозволи захвату відео.

– для локальних відеокамер 384x288, 480x360, 560x420, 640x480, 720x540, 768x576 (напівкадр і повний кадр).

– для мережних камер 176x144, 240x180, 320x240, 352x240, 352x288, 384x288, 480x360, 560x420, 640x480, 704x420, 704x576, 720x540, 720x576, 768x576, 1280x720, 1280x960, 1280x1024, 1600x1200.

2. Програмний детектор руху.

Програмний детектор руху працює на основі порівняння кількості змінених пікселів у сусідніх відеокадрах із заданим у налаштуваннях граничним значенням для кожної відеокамери.

Додатково включає наступні технології:

- налаштування рівня шуму;
- виділення областей аналізу накладенням маски;
- групування в тимчасові сеанси руху.

3. Режими запису відеокадрів.

- без запису на жорсткий диск,
- запис відеокадрів по спрацьовуванню детектора;
- запис по детектору + запис певної кількості відеокадрів до/після спрацьовування детектора (режими передзапису й післязапису).

Відеокадри можуть зберігатися у файлах наступного формату:

- файли у форматі покадрового кодування та передачі відеоданих на основі ієрархічного кодування;
- окремі файли JPEG;
- фільми AVI, стиск кодеком MJPEG.

Відеоархів – циклічний (кільцевий), з видаленням частини старих даних при досягненні вільного місця жорсткого диска певної величини.

Забезпечено можливість накладення (малювання) тексту й часу на збережений відеокадр.

4. Режими інтелектуальної обробки відеоінформації для huddle rooms.

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		32

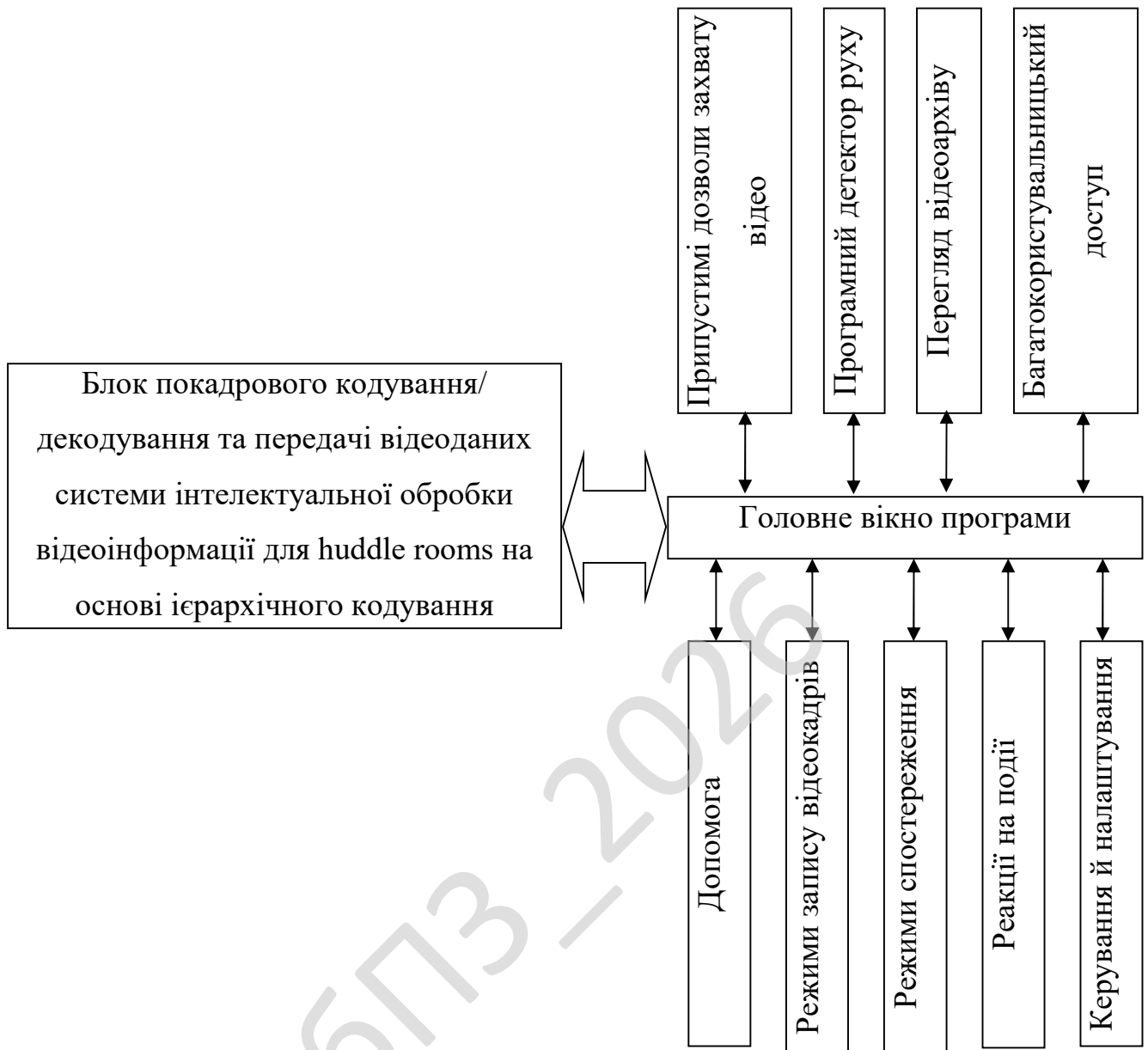


Рисунок 3.2 – Функціональна схема системи

Перегляд у режимі реального часу можливий:

– за допомогою локальної програми, що виводить зображення форматах квадратор, мультіекран, поліекран і ін.;

– за допомогою веб-браузера з дозволених віддалених комп'ютерів.

5. Керування й налаштування.

Реалізована для зареєстрованих користувачів по паролю.

6. Перегляд відеоархіву.

Реалізований для зареєстрованих користувачів по пароллю.

7. Реакції на події.

- Програвання звукового файлу на початок і закінчення сеансу руху.
- Повідомлення через e-mail по початку сеансу руху із вкладеним JPEG файлом першого кадру руху (прим: по мережі на SMTP сервер).
- Виконання користувальницьких скриптів і завдань.

8. Багатокористувальницький доступ.

У системі передбачені 4 групи користувачів:

- інсталятори (установники),
- адміністратори,
- оператори архіву,
- оператори інтелектуальної обробки відеоінформації для huddle rooms.

Кількість користувачів і віддалених робочих місць програмно не обмежено.

9. Допомога.

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.

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		34

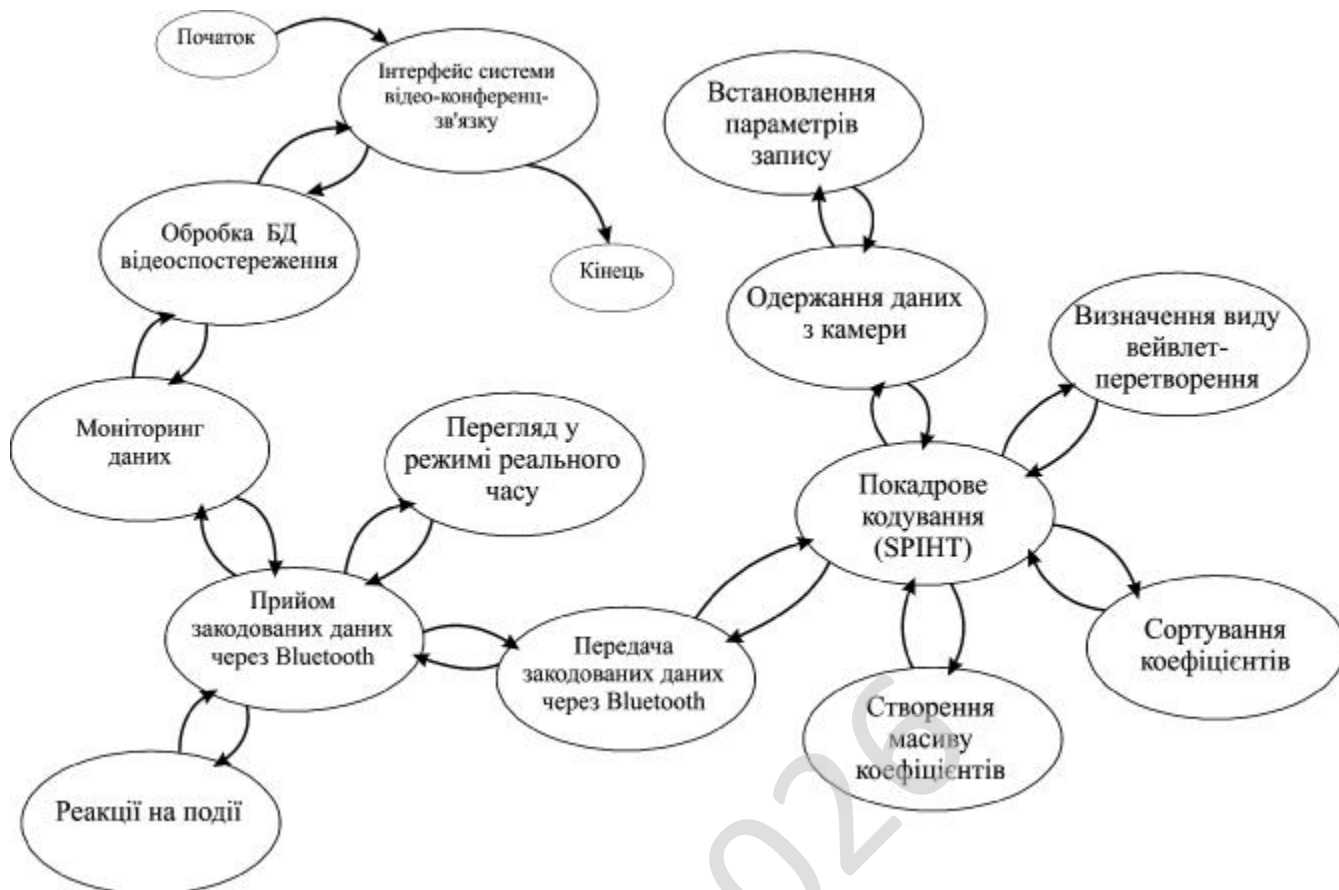


Рисунок 3.3 – Діаграма взаємодії процесів

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

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

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

При детальному її розгляді можна побачити як саме проходить взаємодія у розробленій системі. Використовується модель проектування, графічне представлення «потоків» даних в інформаційній системі.

Діаграми потоків даних містять чотири типи елементів:

- Зовнішні по відношенню до системи сутності.
- Потоки даних між елементами трьох попередніх типів.

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

– Сховища даних (репозиторії).

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

К6ПЗ_2026

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		36

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.

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.

Розглянемо код алгоритму роботи системи.

```
namespace SP
{
    using System;
    using System.Drawing;
```

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		37

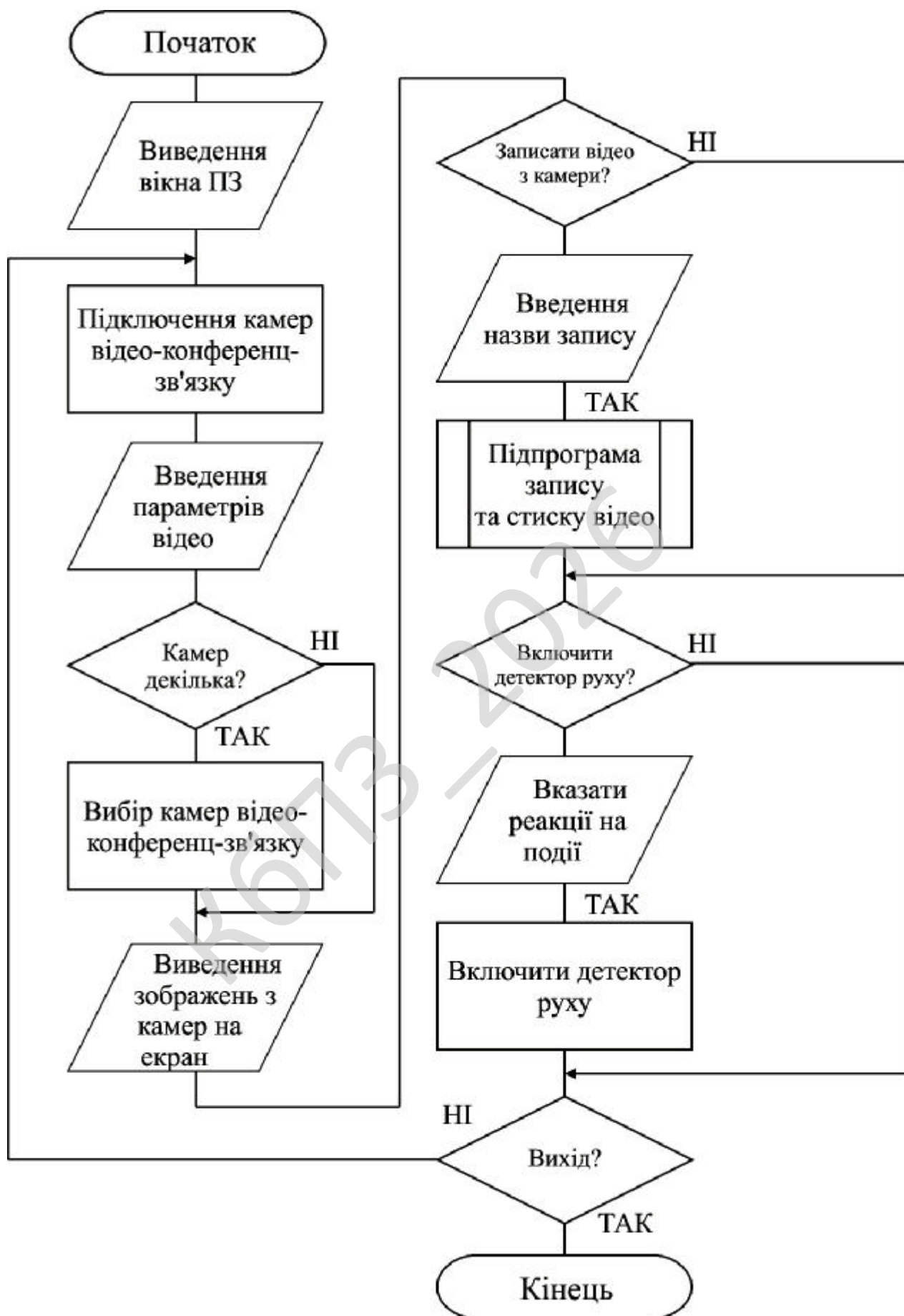


Рисунок 4.1 – Блок-схема основної програми

```

using System.IO;
using System.Threading;
using System.Net;
using videomsource;
public class SPIHTSource : IVideoSource
{
    private stringsource;
    private stringlogin = null;
    private stringpassword = null;
    private objectuserData = null;
    private int framesReceived;
    private int bytesReceived;
    private bool useSeparateConnectionGroup = false;
    private bool preventCaching = false;
    private int frameInterval = 0; // інтервал подання фреймів у
    мілісекундах
    private const int bufSize = 512 * 1024; // розмір буферу
    private const int readSize = 1024; // розмір блоку для читання
    private Threadthread = null;
    private ManualResetEvent stopEvent = null;
    // нова подія у фреймі
    public event CameraEventHandler NewFrame;
    // SeparateConnectioGroup властивості
    // індикатор відкриття WebRequest у групі
    public bool SeparateConnectionGroup
    {
        get { return useSeparateConnectionGroup; }
        set { useSeparateConnectionGroup = value; }
    }
    // PreventCaching властивості
    // Якщо властивості є правильними, то керуємо параметриу URL. Цей клієнт
    повинен бути встановлений на проксі-сервері.
    public bool PreventCaching
    {
        get { return preventCaching; }
        set { preventCaching = value; }
    }
    // FrameInterval властивості - інтервал між фреймами
    //Якщо властивості встановлені в 100, тоді джерело формує 10 фреймів
    // в секунду
    public int FrameInterval
    {

```

```

        get { return frameInterval; }
        set { frameInterval = value; }
    }
    // VideoSource властивості
    public virtual string VideoSource
    {
        get { return source; }
        set { source = value; }
    }
    // Властивості підключення
    public string Login
    {
        get { return login; }
        set { login = value; }
    }
    // Властивості паролювання
    public string Password
    {
        get { return password; }
        set { password = value; }
    }
    // FramesReceived властивості
    public int FramesReceived
    {
        get
        {
            int frames = framesReceived;
            framesReceived = 0;
            return frames;
        }
    }
    // BytesReceived властивості
    public int BytesReceived
    {
        get
        {
            int bytes = bytesReceived;
            bytesReceived = 0;
            return bytes;
        }
    }
    // UserData властивості

```

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		40

```

public object UserData
{
    get { return userData; }
    set { userData = value; }
}
// Отримуємо стан вихідного відео
public bool Running
{
    get
    {
        if (thread != null)
        {
            if (thread.Join(0) == false)
                return true;
            // Якщо стан не заданий, звільнюємо ресурси
            Free();
        }
        return false;
    }
}
// Конструктор
public SPIHTSource()
{
}
// Починаємо роботу
public void Start()
{
    if (thread == null)
    {
        framesReceived = 0;
        bytesReceived = 0;
        // Створюємо подію
        stopEvent = new ManualResetEvent(false);

        // Створюємо й стартуємо нову подію
        thread = new Thread(new ThreadStart(WorkerThread));
        thread.Name = source;
        thread.Start();
    }
}
// Сигнал події до остановки роботи
public void SignalToStop()

```

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		41

```

{
    // Остановлюємо подію
    if (thread != null)
    {
        // Сигнал остановки
        stopEvent.Set();
    }
}
// Чекаємо остановки події
public void WaitForStop()
{
    if (thread != null)
    {
        // Чекаємо остановки події
        thread.Join();
        Free();
    }
}
}
}

```

Хоча я реалізовував програму сам, було використано підходи Scrum для саморозвитку та пришвидшенню розробки, розглянемо цей метод. Scrum – підхід управління проектами для гнучкої розробки програмного забезпечення. Скрам чітко робить акцент на якісному контролі процесу розробки.

Підхід вперше описали Гіротакі Такеучі та Ікуджіро Нонака в статті The New New Product Development Game (Гарвардський Діловий Огляд, січ–лют 1986). Вони відзначили, що проекти, над якими працюють невеликі, крос–функціональні команди, зазвичай систематично продукують кращі результати, і пояснили це, як «підхід регбі». У 1991 році ДеГрейс та Шталь у книжці Злі проблеми, справедливі рішення послалися на цей підхід, як на Scrum (штовханина; сутичка навколо м'яча (у регбі)), спортивний термін, згаданий в статті Такеучі і Нонака. Кен Швабер на початку 1990–х використовував підхід який привів Scrum в його компанію.

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		42

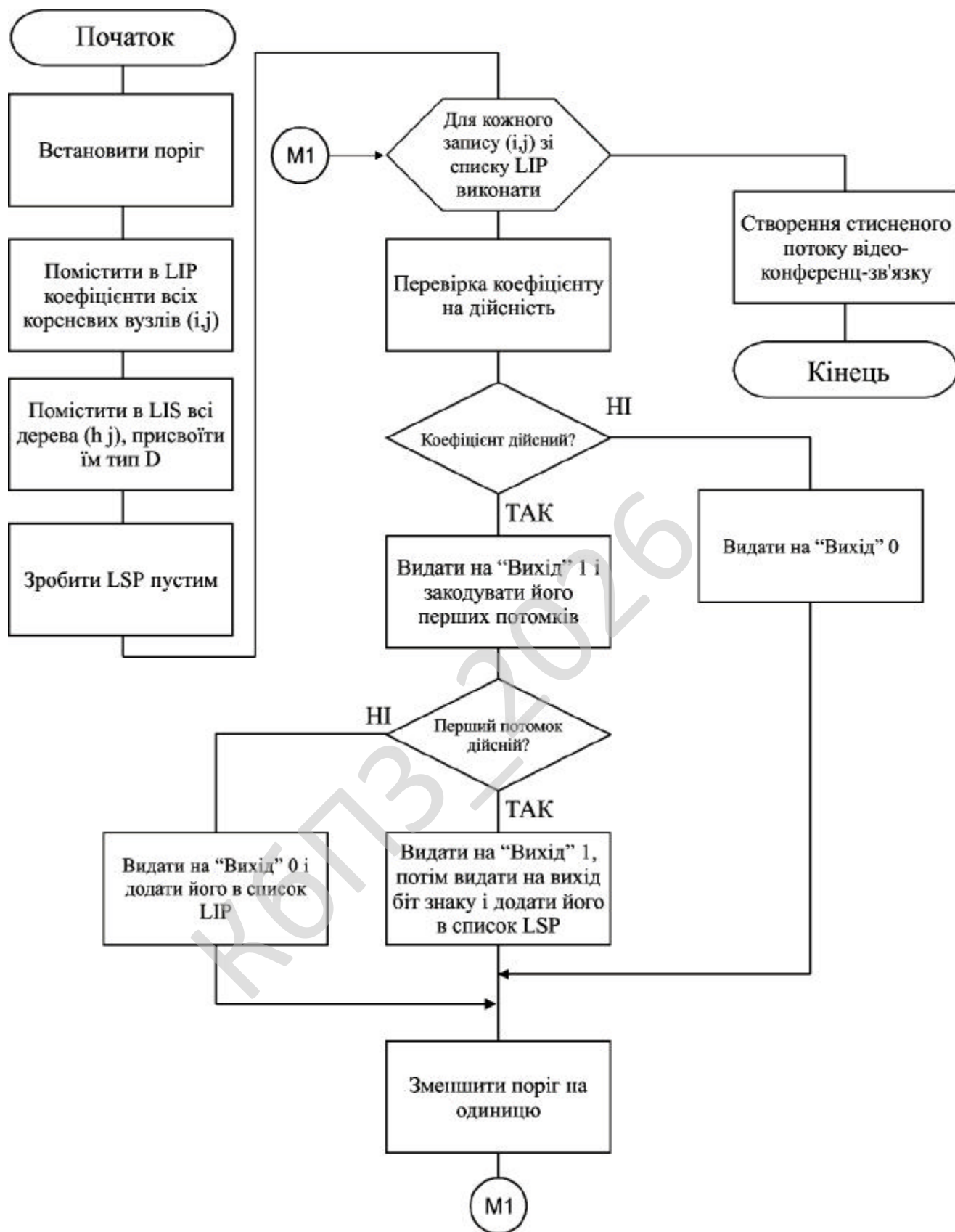


Рисунок 4.2 – Блок-схема роботи підпрограми

Вперше метод Scrum було представлено на загальний огляд задокументованим, чітко сформульованим та описаним спільно Сазерлендом та Швабером на OOPSLA'96 в Остіні. Швабер та Сазерленд протягом наступних років працювали разом щоб обробити та описати весь їхній досвід та найкращі практичні зразки для індустрії в одне ціле, в ту методологію, що відома сьогодні як Scrum. Швабер об'єднав зусилля з Майком Бідлом в 2001, щоб детально описати метод в книжці Agile Software Development with SCRUM. Не зважаючи на те, що для Scrum нарікли долю управління проектами з розробки ПЗ, він може також використовуватися в роботі команд обслуговувань програмного забезпечення (software maintenance teams), або як підхід управління розробкою і супроводом програм: Scrum of Scrums.

Scrum – це кістяк процесу, який включає набір методів і попередньо визначених ролей. Головні дійові особи – ScrumMaster, той хто опікується процесами, веде їх і працює як керівник проекту, Власник Продукту, людина, що представляє інтереси кінцевих користувачів та інших зацікавлених в продукті сторін, та Команду, яка включає розробників.

Протягом кожного спринту, 15–30 денного періоду (тривалість визначається командою), працівники створюють функціональний ріст програмного забезпечення.

Набір можливостей, які імплементуються кожного спринту, приходять з етапу, що має назву product backlog (документація запитів на виконання робіт), який має найвищу пріоритетність за рівнем вимог до роботи, що повинна бути виконана.

Запити на виконання робіт (backlog items), що визначені протягом наради з планування спринту (sprint planning meeting), переміщуються в етап спринту. Протягом цієї наради Власник Продукту інформує про завдання, які він хоче, аби були виконані. Тоді Команда визначає, скільки з бажаного вони можуть зробити, щоб завершити необхідні частини протягом наступного спринту. Протягом спринту команда виконує визначений фіксований список завдань (т.з. backlog

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		44

items). Впродовж цього періоду ніхто не має права змінювати перелік запитів на виконання робіт, що слід розуміти, як заморожування вимог (requirements) протягом спринту.

Product backlog – це документ, який має список вимог до функціональності, які упорядковані згідно зі ступенем важливості. Product backlog представляє список того, що повинно бути реалізовано. Елементи цього списку називається «історіями» (user story) або елементами backlog–у (backlog items). Product backlog відкритий для редагування усім учасникам Scrum–процесу.

Обов'язкові поля:

1. ID – унікальний ідентифікатор, порядковий номер, який використовується для ідентифікації історій у разі їх перейменування.

2. Назва (Name) – стислий опис історії. Він повинен бути однозначним, щоб і розробники і product owner могли зрозуміти, про що йдеться і відрізнити одну історію від іншої.

3. Важливість (Importance) – ступінь важливості даної історії на погляд product owner 'а. Зазвичай являє собою натуральне число, іноді для цієї цілі використовуються числа Фібоначчі. Чим більше значення, тим більше пріоритет.

4. Попередня оцінка (initial estimate) – початкова оцінка об'єму робіт, необхідного для реалізації історії порівняно з іншими історіями. Вимірюється у story point'ах. Приблизно відповідає числу «ідеальних людино–днів».

5. Як продемонструвати (how to demo) – стисле пояснення того, як завершена задача буде продемонстрована у кінці спринта. Дане поле може являти собою код автоматизованого приймального тесту.

Додаткові поля. Іноді, також, використовуються додаткові поля у product backlog, в основному для того, щоб допомогти product owner'у визначитися з його пріоритетами.

Категорія (track). Наприклад, «панель управління» чи «оптимізація». За допомогою цього поля product owner може легко вибрати усі пункти категорії «оптимізація» і задати їм низький пріоритет.

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		45

Компоненти (components) – указує, які компоненти (наприклад, база даних, сервер, клієнт) будуть зачеплені при реалізації історії. Дане поле складається з групи checkbox'ів, які відмічаються, якщо відповідні компоненти потребують змін.

Ініціатор запиту (requestor). Product owner може захотіти зберігати інформацію про усіх замовників, зацікавлених у даній задачі. Це потрібно для того, щоб тримати їх у курсі діла про хід виконання робіт.

ID у системі обліку помилок (bug tracking ID) – якщо ви використовуєте окрему систему обліку помилок, тоді у описі історії корисно зберігати посилання на всі дефекти, які до неї відносяться.

Sprint backlog – містить функціональність, обрану Product Owner із Product Backlog. Всі функції розбиті по задачах, кожна з яких оцінюється командою. Кожен день команда оцінює об'єм роботи, який необхідно провести для завершення задачі.

Burndown chart – показує, скільки вже виконано і скільки ще залишається зробити.

Планування спринта (Sprint Planning Meeting)

Проходить на початку нової ітерації Спринта:

- Із Product Backlog обираються задачі, зобов'язання по виконанню яких за спринт приймає на себе команда;
- На основі обраних задач створюється Sprint Backlog. Кожна задача оцінюється у ідеальних людино-годинах;
- Рішення задачі не повинно займати більше 12 годин або одного дня. При необхідності задача розбивається на підзадачі;
- Обговорюється та визначається, яким чином буде реалізовано цей об'єм робіт;
- Тривалість наради обмежена зверху 4–8 годинами в залежності від тривалості ітерації, досвіду команди тощо;

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		46

– (перша частина наради) Беруть участь Product Owner + Команда: обирають задачі із Product Backlog;

– (друга частина наради) Бере участь лише команда: обговорюють технічні деталі реалізації, наповнюють Sprint Backlog.

Щоденна нарада (Daily Scrum Meeting)

Відбувається кожен день протягом спринта. Є «пульсом» ходу спринта.

Нараді властиві наступні обмеження:

- починається точно вчасно;
- всі можуть спостерігати, але говорять тільки обрані;
- триває не більш ніж 15 хвилин;
- проводиться в одному і тому ж місці протягом одного спринта.

Протягом наради кожен член команди відповідає на 3 запитання:

- Що зроблено з моменту попередньої щоденної наради?;
- Що буде зроблено з моменту поточної наради до наступної?;
- Які проблеми заважають досягненню цілей спринта? (Над рішенням цих проблем працює ScrumMaster. Зазвичай це рішення проходить за рамками щоденної наради і у складі осіб, що безпосередньо займаються даною перешкодою.)

Демонстрація (Sprint Review Meeting):

- Проходить у кінці ітерації (спринта).
- Команда демонструє внесок функціональності до продукту всім зацікавленим особам.
- Залучається максимальна кількість глядачів.
- Усі члени команди беруть участь у демонстрації (одна людина на демонстрацію або кожен показує, що зробив за спринт).

– Обмежена 4-ма годинами в залежності від тривалості ітерації і змін у продукті.

Ретроспектива (Sprint Retrospective):

- Члени команди висловлюють свою думку про минулий спринт.

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		47

- Відповідають на два основних запитання: Що було зроблено добре у минулому спринті?; Що потрібно покращити в наступному?.
- Виконують покращення процесу розробки (вирішують питання та фіксують вдалі рішення).
- Обмежена 1–3ма годинами.

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 класів; Діаграма компонент; Діаграма об'єктів; Діаграма розгортання.

Діаграма класів це статичне представлення структури моделі. Відображає статичні (декларативні) елементи, такі як: класи, типи даних, їх зміст та відношення.

Діаграма класів, також, може містити позначення для пакетів та може містити позначення для вкладених пакетів. Також, діаграма класів може містити позначення деяких елементів поведінки, однак їх динаміка розкривається в інших типах діаграм.

Діаграма класів (class diagram) служить для представлення статичної структури моделі системи в термінології класів об'єктно-орієнтованого програмування. На цій діаграмі показують класи, інтерфейси, об'єкти й кооперації, а також їхні відносини.

В UML існують наступні типи зв'язків які використовуються у діаграмі класів: Асоціації; Агрегація; Композиція.

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		48

Асоціації це якщо між двома класами визначена асоціація, то можна переміщатися від об'єктів одного класу до об'єктів іншого. Цілком припустимі випадки, коли обидва кінці асоціації відносяться до одного і того ж класу. Це означає, що з об'єктом деякого класу дозволено зв'язати інші об'єкти з того ж класу. Асоціація, що зв'язує два класи, називається бінарної. Можна, хоча це рідко буває необхідним, створювати асоціації, що зв'язують відразу кілька класів. Графічно асоціація зображується у вигляді лінії, що з'єднує клас сам з собою або з іншими класами.

Асоціації може бути присвоєно ім'я, яке описує природу відносини. Зазвичай ім'я асоціації не вказується, якщо тільки ви не хочете явно задати для неї рольові імена або у вашій моделі настільки багато асоціацій, що виникає необхідність посилатися на них і відрізняти один від одного. Ім'я буде особливо корисним, якщо між одними і тими ж класами існує кілька різних асоціацій.

Клас, що бере участь в асоціації, грає в ній деяку роль. По суті, це "обличчя", яким клас, що знаходиться на одній стороні асоціації, звернений до класу з іншого її боку. Можна явно позначити роль, яку клас грає в асоціації.

Часто при моделюванні буває важливо вказати, скільки об'єктів може бути пов'язано допомогою одного примірника асоціації. Це число називається кратністю (Multiplicity) ролі асоціації та записується або як вираз, значенням якого є діапазон значень, або в явному вигляді.

Вказуючи кратність на одному кінці асоціації, ви тим самим говорите, що на цьому кінці саме стільки об'єктів повинно відповідати кожному об'єкту на протилежному кінці. Кратність можна задати рівною одиниці (1), можна вказати діапазон: "нуль або одиниця" (0..1), "багато" (0 .. *), "одиниця або більше" (1 .. *). Дозволяється також вказувати певне число (наприклад, 3). За допомогою списку можна задати і більш складні кратності, наприклад 0. . 1, 3..4, 6 .. *, що означає "будь-яке число об'єктів, крім 2 і 5".

Агрегація це проста асоціація між двома класами відображає структурний відношення між рівноправними сутностями, коли обидва класу знаходяться на

одному концептуальному рівні і ні один не є більш важливим, ніж інший. Але іноді доводиться моделювати відношення типу «частина/ціле», в якому один з класів має більш високий ранг (ціле) і складається з декількох менших за рангом (частин).

Ставлення такого типу називають агрегацією; воно зараховане до відносин типу «має» (з урахуванням того, що об'єкт-ціле має кілька об'єктів-частин). Агрегація є окремим випадком асоціації і зображується у вигляді простої асоціації з незафарбованим ромбом з боку «цілого». Графічно агрегація представляється порожнім ромбом на блоці класу, і лінією, яка від цього ромба до міститься класу.

Композиція це більш суворий варіант агрегації. Відома також як агрегація за значенням.

Композиція має жорстку залежність часу існування екземплярів класу контейнера та примірників містяться класів. Якщо контейнер буде знищений, то весь його вміст буде також знищено. Графічно представляється як і агрегація, але з зафарбованим ромбиком.

Діаграма компонент в UML це діаграма, на якій відображаються компоненти, залежності та зв'язки між ними.

Діаграма компонент відображає залежності між компонентами програмного забезпечення, включаючи компоненти вихідних кодів, бінарні компоненти, та компоненти, що можуть виконуватись.

Модуль програмного забезпечення може бути представлено в якості компоненти. Деякі компоненти існують під час компіляції, деякі – під час компонування, а деякі під час роботи програми.

Діаграма компонент відображає лише структурні характеристики, для відображення окремих екземплярів компонент слід використовувати діаграму розгортання.

Компоненти об'єднуються разом використовуючи структурні зв'язки (assembly connector) щоб об'єднати інтерфейси двох компонент. Це ілюструє зв'язок типу «клієнт-сервер».

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		50

Структурна взаємодія – «зв'язок двох компонент, який передбачає, що один з них надає послуги, потрібні іншому компоненту».

При використанні діаграми компонент щоб показати внутрішню структуру компонента, клієнтські та серверні інтерфейси можуть утворювати пряме з'єднання з внутрішніми. Таке з'єднання називається з'єднанням делегації.

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.

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		51

4.2 Захист розробленого програмного забезпечення

Дані в програмі захищаються за допомогою використання алгоритму CAST-128 (або CAST5) у криптографії, це блоковий алгоритм симетричного шифрування на основі мережі Фейстеля, який використовується в цілому ряді продуктів криптографічного захисту, зокрема деяких версіях PGP і GPG і крім того схвалений для використання Канадським урядом.

Основні відомості

Алгоритм був створений в 1996 році Карлайлом Адамсом (Carlisle Adams) і Стаффордом Таваресом (Stafford Tavares) використовуючи метод побудови шифрів CAST, який використовується також і іншим їхнім алгоритмом CAST-256 (алгоритм-кандидат AES).

CAST-128 складається з 12 або 16 раундів мережі Фейстеля з розміром блоку 64 біта й довжиною ключа від 40 до 128 біт (але тільки з інкрементацією по 8 біт). 16 раундів використовуються коли розміри ключа перевищують 80 біт. В алгоритмі використовуються 8х16 S-блоки, засновані на бент-функції, операції XOR і модулярної арифметиці (модулярне додавання й вирахування). Є три різні типи функцій раундів, але вони схожі за структурою й різняться тільки у виборі виконуваної операції (додавання, вирахування або XOR) у різних місцях.

Хоча CAST-128 захищений патентом Entrust, його можна використовувати в усьому світі для комерційних або некомерційних цілей безкоштовно.

Опис

CAST – це популярний 64-бітовий шифр, що допускає розміри ключа аж до 128 біт

Алгоритм CAST використовує 64-бітовий блок і 64-бітовий ключ. CAST стійкий до диференціального й лінійного криптоаналізу. Сила алгоритму CAST укладена в його S-блоках. В CAST немає фіксованих S-блоків і для кожного додатка вони конструюються заново. Створений для конкретної реалізації CAST S-блок уже більше ніколи не міняється. Інакше кажучи, S-блоки залежать від

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		52

реалізації, а не від ключа. Northern Telecom використовує CAST у своєму пакеті програм Entrust для комп'ютерів Macintosh, PC і робочих станцій UNIX. Обрані ними S-блоки не опубліковані, що втім не дивно.

CAST-128 належить компанії Entrust Technologies, але є безкоштовним як для комерційного, так і для некомерційного використання. CAST-256 – безкоштовне доступне розширення CAST-128, яке ухвалює розмір ключа до 256 біт і має розмір блоку 128 біт. CAST-256 був одним з первісних кандидатів на AES.

Опис алгоритму

CAST-128 заснований на мережі Фейстеля. Повний алгоритм шифрування викладений у наступних чотирьох кроках:

ВХІД: текст $m_1 \dots m_{64}$, ключ $K = k_1 \dots k_{128}$.

ВИХІД: зашифрований текст $c_1 \dots c_{64}$.

1. (розгорнення ключа) становить 16 пар підключів $\{K_{m_i}, K_{r_i}\}$ отриманих з K (див. розділи Пари раундових ключів і Неідентичні раунди).

2. $(L_0, R_0) \leftarrow (m_1 \dots m_{32}, m_{33} \dots m_{64})$. (Розділяє текст на ліву й праву 32-бітні половини $L_0 = m_1 \dots m_{32}$ і $R_0 = m_{33} \dots m_{64}$).

3. (16 раундів) for i from 1 to 16, обчислити L_i і R_i у такий спосіб: $L_i = R_{i-1}$; $R_i = L_{i-1} \oplus F(R_{i-1}, K_{m_i}, K_{r_i})$, де F визначена в розділі «Пари раундових ключів» (F має тип 1, тип 2, тип 3 або, залежно від i).

4. $c_1 \dots c_{64} \leftarrow (R_{16}, L_{16})$. (Міняємо остаточні блоки місцями L_{16}, R_{16} і поєднуємо, щоб сформувати зашифрований текст.)

Розшифруванні збігається з алгоритмом шифрування, наведеним вище, крім того, що раунди (i , отже, пари підключів), використовуються у зворотному порядку, щоб обчислити (L_0, R_0) з (R_{16}, L_{16}) .

Пари раундових ключів

CAST-128 використовує пару підключів за раунд: 32-бітні величини K_m використовується в якості "маскування" ключа й K_r використовують як "перестановки" ключа, з яких використовуються тільки початкові 5-біт.

Неідентичні раунди

Три різні типів функції використовуються в CAST-128. Типи виглядає в такий спосіб (де "D" є вхідними даними у функцію F і "Ia"- "Id" є найбільш значимий байт – найменш значимий байт I, відповідно). Зверніть увагу, що "+" і "-" додавання й вирахування по модулю $2^{**} 32$, "" є побітове XOR і "<<<" є циклічним зрушенням уліво.

Раунди $I = ((K_{m_i} + R_{i-1}) <<< K_{r_i})$

1,4,7,10,13,16 $F = ((S1[I_a] \ S2[I_b]) - (S3[I_c])) + S4[I_d]$

Раунди $I = ((K_{m_i} \wedge R_{i-1}) <<< K_{r_i})$

2,5,8,11,14 $F = ((S1[I_a] - S2[I_b]) + (S3[I_c])) \ S4[I_d]$

Раунди $I = ((K_{m_i} - R_{i-1}) <<< K_{r_i})$

3,6,9,12,15 $F = ((S1[I_a] + S2[I_b]) \ (S3[I_c])) - S4[I_d]$

Поля заміни

CAST-128 використовує вісім полів заміни: поля S1, S2, S3 і S4 раундові функції полів заміни, S5, S6, S7 і S8 є ключами розгорнення полів заміни. Незважаючи на те, що 8 полів заміни вимагають у цілому 8 Кбайт для зберігання, зверніть увагу на те, що тільки 4 Кбайта потрібні під час фактичного шифрування / дешифрування, тому що генерація підключа звичайно робиться до будь-якого введення даних.

Ключі розгорнення

Представимо 128-розрядний ключ у вигляді $x_0x_1x_2x_3x_4x_5x_6x_7x_8x_9x_{10}x_{11}x_{12}x_{13}x_{14}x_{15}x_{16}x_{17}$, де x_0 старший байт, і x_{17} молодший байт.

Представимо $z_0..z_{15}$ проміжними (тимчасовими) байтами. $Si[]$ представляє поле заміни і $i \wedge j$ представляє додавання по Xor'у.

Поля заміни формуються із ключа $x_0x_1x_2x_3x_4x_5x_6x_7x_8x_9x_{10}x_{11}x_{12}x_{13}x_{14}x_{15}x_{16}x_{17}$ у такий спосіб.

$$z_0z_1z_2z_3 = x_0x_1x_2x_3 \wedge S5[x_D] \wedge S6[x_F] \wedge S7[x_C] \wedge S8[x_E] \wedge S7[x_8]$$

$$z_4z_5z_6z_7 = x_8x_9x_{10}x_{11} \wedge S5[z_0] \wedge S6[z_2] \wedge S7[z_1] \wedge S8[z_3] \wedge S8[x_A]$$

$$z_8z_9z_{10}z_{11} = x_{12}x_{13}x_{14}x_{15} \wedge S5[z_7] \wedge S6[z_6] \wedge S7[z_5] \wedge S8[z_4] \wedge S5[x_9]$$

$zCzDzEzF = x4x5x6x7 \wedge S5[zA] \wedge S6[z9] \wedge S7[zB] \wedge S8[z8] \wedge S6[xB]$
 $K1 = S5[z8] \wedge S6[z9] \wedge S7[z7] \wedge S8[z6] \wedge S5[z2]$
 $K2 = S5[zA] \wedge S6[zB] \wedge S7[z5] \wedge S8[z4] \wedge S6[z6]$
 $K3 = S5[zC] \wedge S6[zD] \wedge S7[z3] \wedge S8[z2] \wedge S7[z9]$
 $K4 = S5[zE] \wedge S6[zF] \wedge S7[z1] \wedge S8[z0] \wedge S8[zC]$
 $x0x1x2x3 = z8z9zAzB \wedge S5[z5] \wedge S6[z7] \wedge S7[z4] \wedge S8[z6] \wedge S7[z0]$
 $x4x5x6x7 = z0z1z2z3 \wedge S5[x0] \wedge S6[x2] \wedge S7[x1] \wedge S8[x3] \wedge S8[z2]$
 $x8x9xAxB = z4z5z6z7 \wedge S5[x7] \wedge S6[x6] \wedge S7[x5] \wedge S8[x4] \wedge S5[z1]$
 $xCxDxExF = zCzDzEzF \wedge S5[xA] \wedge S6[x9] \wedge S7[xB] \wedge S8[x8] \wedge S6[z3]$
 $K5 = S5[x3] \wedge S6[x2] \wedge S7[xC] \wedge S8[xD] \wedge S5[x8]$
 $K6 = S5[x1] \wedge S6[x0] \wedge S7[xE] \wedge S8[xF] \wedge S6[xD]$
 $K7 = S5[x7] \wedge S6[x6] \wedge S7[x8] \wedge S8[x9] \wedge S7[x3]$
 $K8 = S5[x5] \wedge S6[x4] \wedge S7[xA] \wedge S8[xB] \wedge S8[x7]$
 $z0z1z2z3 = x0x1x2x3 \wedge S5[xD] \wedge S6[xF] \wedge S7[xC] \wedge S8[xE] \wedge S7[x8]$
 $z4z5z6z7 = x8x9xAxB \wedge S5[z0] \wedge S6[z2] \wedge S7[z1] \wedge S8[z3] \wedge S8[xA]$
 $z8z9zAzB = xCxDxExF \wedge S5[z7] \wedge S6[z6] \wedge S7[z5] \wedge S8[z4] \wedge S5[x9]$
 $zCzDzEzF = x4x5x6x7 \wedge S5[zA] \wedge S6[z9] \wedge S7[zB] \wedge S8[z8] \wedge S6[xB]$
 $K9 = S5[z3] \wedge S6[z2] \wedge S7[zC] \wedge S8[zD] \wedge S5[z9]$
 $K10 = S5[z1] \wedge S6[z0] \wedge S7[zE] \wedge S8[zF] \wedge S6[zC]$
 $K11 = S5[z7] \wedge S6[z6] \wedge S7[z8] \wedge S8[z9] \wedge S7[z2]$
 $K12 = S5[z5] \wedge S6[z4] \wedge S7[zA] \wedge S8[zB] \wedge S8[z6]$
 $x0x1x2x3 = z8z9zAzB \wedge S5[z5] \wedge S6[z7] \wedge S7[z4] \wedge S8[z6] \wedge S7[z0]$
 $x4x5x6x7 = z0z1z2z3 \wedge S5[x0] \wedge S6[x2] \wedge S7[x1] \wedge S8[x3] \wedge S8[z2]$
 $x8x9xAxB = z4z5z6z7 \wedge S5[x7] \wedge S6[x6] \wedge S7[x5] \wedge S8[x4] \wedge S5[z1]$
 $xCxDxExF = zCzDzEzF \wedge S5[xA] \wedge S6[x9] \wedge S7[xB] \wedge S8[x8] \wedge S6[z3]$
 $K13 = S5[x8] \wedge S6[x9] \wedge S7[x7] \wedge S8[x6] \wedge S5[x3]$
 $K14 = S5[xA] \wedge S6[xB] \wedge S7[x5] \wedge S8[x4] \wedge S6[x7]$
 $K15 = S5[xC] \wedge S6[xD] \wedge S7[x3] \wedge S8[x2] \wedge S7[x8]$
 $K16 = S5[xE] \wedge S6[xF] \wedge S7[x1] \wedge S8[x0] \wedge S8[xD]$

половина, що залишається, ідентична тому, що дане вище, продовження від останнього створило $x0..xf$, щоб генерувати ключі K17 – K32.

$$z0z1z2z3 = x0x1x2x3 \wedge S5[xD] \wedge S6[xF] \wedge S7[xC] \wedge S8[xE] \wedge S7[x8]$$

$$z4z5z6z7 = x8x9xAxB \wedge S5[z0] \wedge S6[z2] \wedge S7[z1] \wedge S8[z3] \wedge S8[xA]$$

$$z8z9zAzB = xCxDxExF \wedge S5[z7] \wedge S6[z6] \wedge S7[z5] \wedge S8[z4] \wedge S5[x9]$$

$$zCzDzEzF = x4x5x6x7 \wedge S5[zA] \wedge S6[z9] \wedge S7[zB] \wedge S8[z8] \wedge S6[xB]$$

$$K17 = S5[z8] \wedge S6[z9] \wedge S7[z7] \wedge S8[z6] \wedge S5[z2]$$

$$K18 = S5[zA] \wedge S6[zB] \wedge S7[z5] \wedge S8[z4] \wedge S6[z6]$$

$$K19 = S5[zC] \wedge S6[zD] \wedge S7[z3] \wedge S8[z2] \wedge S7[z9]$$

$$K20 = S5[zE] \wedge S6[zF] \wedge S7[z1] \wedge S8[z0] \wedge S8[zC]$$

$$x0x1x2x3 = z8z9zAzB \wedge S5[z5] \wedge S6[z7] \wedge S7[z4] \wedge S8[z6] \wedge S7[z0]$$

$$x4x5x6x7 = z0z1z2z3 \wedge S5[x0] \wedge S6[x2] \wedge S7[x1] \wedge S8[x3] \wedge S8[z2]$$

$$x8x9xAxB = z4z5z6z7 \wedge S5[x7] \wedge S6[x6] \wedge S7[x5] \wedge S8[x4] \wedge S5[z1]$$

$$xCxDxExF = zCzDzEzF \wedge S5[xA] \wedge S6[x9] \wedge S7[xB] \wedge S8[x8] \wedge S6[z3]$$

$$K21 = S5[x3] \wedge S6[x2] \wedge S7[xC] \wedge S8[xD] \wedge S5[x8]$$

$$K22 = S5[x1] \wedge S6[x0] \wedge S7[xE] \wedge S8[xF] \wedge S6[xD]$$

$$K23 = S5[x7] \wedge S6[x6] \wedge S7[x8] \wedge S8[x9] \wedge S7[x3]$$

$$K24 = S5[x5] \wedge S6[x4] \wedge S7[xA] \wedge S8[xB] \wedge S8[x7]$$

$$z0z1z2z3 = x0x1x2x3 \wedge S5[xD] \wedge S6[xF] \wedge S7[xC] \wedge S8[xE] \wedge S7[x8]$$

$$z4z5z6z7 = x8x9xAxB \wedge S5[z0] \wedge S6[z2] \wedge S7[z1] \wedge S8[z3] \wedge S8[xA]$$

$$z8z9zAzB = xCxDxExF \wedge S5[z7] \wedge S6[z6] \wedge S7[z5] \wedge S8[z4] \wedge S5[x9]$$

$$zCzDzEzF = x4x5x6x7 \wedge S5[zA] \wedge S6[z9] \wedge S7[zB] \wedge S8[z8] \wedge S6[xB]$$

$$K25 = S5[z3] \wedge S6[z2] \wedge S7[zC] \wedge S8[zD] \wedge S5[z9]$$

$$K26 = S5[z1] \wedge S6[z0] \wedge S7[zE] \wedge S8[zF] \wedge S6[zC]$$

$$K27 = S5[z7] \wedge S6[z6] \wedge S7[z8] \wedge S8[z9] \wedge S7[z2]$$

$$K28 = S5[z5] \wedge S6[z4] \wedge S7[zA] \wedge S8[zB] \wedge S8[z6]$$

$$x0x1x2x3 = z8z9zAzB \wedge S5[z5] \wedge S6[z7] \wedge S7[z4] \wedge S8[z6] \wedge S7[z0]$$

$$x4x5x6x7 = z0z1z2z3 \wedge S5[x0] \wedge S6[x2] \wedge S7[x1] \wedge S8[x3] \wedge S8[z2]$$

$$x8x9xAxB = z4z5z6z7 \wedge S5[x7] \wedge S6[x6] \wedge S7[x5] \wedge S8[x4] \wedge S5[z1]$$

$$x C x D x E x F = z C z D z E z F \wedge S 5[x A] \wedge S 6[x 9] \wedge S 7[x B] \wedge S 8[x 8] \wedge S 6[z 3]$$

$$K 29 = S 5[x 8] \wedge S 6[x 9] \wedge S 7[x 7] \wedge S 8[x 6] \wedge S 5[x 3]$$

$$K 30 = S 5[x A] \wedge S 6[x B] \wedge S 7[x 5] \wedge S 8[x 4] \wedge S 6[x 7]$$

$$K 31 = S 5[x C] \wedge S 6[x D] \wedge S 7[x 3] \wedge S 8[x 2] \wedge S 7[x 8]$$

$$K 32 = S 5[x E] \wedge S 6[x F] \wedge S 7[x 1] \wedge S 8[x 0] \wedge S 8[x D]$$

Маскування й перестановка підключів

$K m_1, \dots, K m_{16}$ 32-розрядні підключи маскування (один на раунд).

$K r_1, \dots, K r_{16}$ 32-розрядні перестановки підключів (один на раунд); тільки молодші 5 бітів використовуються в кожному раунді.

for ($i=1$; $i \leq 16$; $i++$) { $K m_i = K i$; $K r_i = K 16+i$; }

Змінний розмір ключа

CAST-128 Алгоритм шифрування був розроблений, щоб розмір ключа міг варіюватися від 40 до 128 біт, в 8-бітному кроці (тобто припустимі розміри ключа рівняються 40, 48, 56, 64..., 112, 120, і 128 бітам). Для змінної роботи розміру ключа специфікація наступні:

- 1) Для розмірів ключа до й включаючи 80 бітів (тобто, 40, 48, 56, 64, 72, і 80 бітів) алгоритм точно такої ж, але використовує 12 раундів замість 16;
- 2) Для розмірів ключа більше, чим 80 бітів, алгоритм використовує повні 16 раундів;
- 3) Для розмірів ключа менше, чим 128 бітів ключ доповнений нульовими байтами (у самих правих, або молодших, позиціях) до 128 биток (тому що розклад ключа CAST 128 ухвалює вхідний ключ 128 бітів).

Розшифрування

Розшифрування для CAST-128 відносно проста. Розшифрування працює в тому ж алгоритмічному напрямку, що й шифрування, починаючи із зашифрованого тексту як вхідних даних. При цьому підключ використовуються у зворотному напрямку.

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

На рисунку 5.1 зображено розроблене у бакалаврської дипломної роботі програмне забезпечення системи інтелектуальної обробки відеоінформації для huddle rooms.

З рисунку можна побачити що інтерфейс головного вікна розподілено на наступні функціональні розділи:

- Функціональних кнопок ПЗ.
- Навігаційного меню яке викликається натисканням правої клавіші маніпулятора миші.
- Верхнього меню: Програма – обрання типу виклику; Про автора – Довідка системи.
- Розділу обрання групи.
- 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.

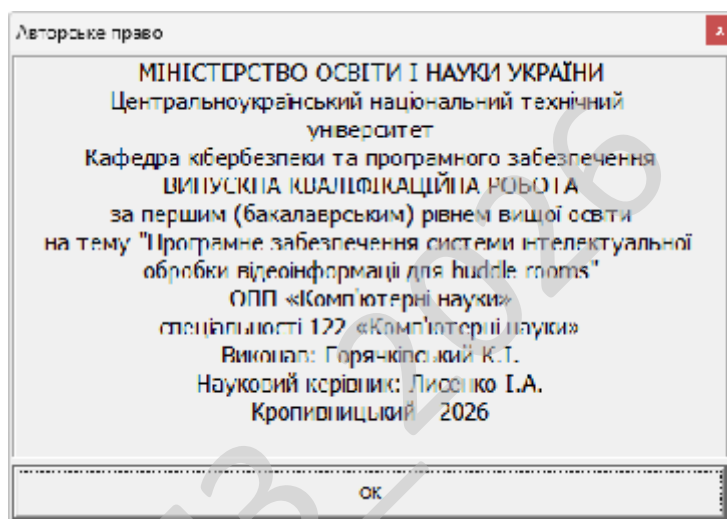


Рисунок 5.2 – Авторське право

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

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		59

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

До діяльностей пов'язаних із розгортанням програмного забезпечення відносять:

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

При впровадженні програмного забезпечення потрібно урахувати наступні дії:

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

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

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

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		60

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.

Обрано умови розповсюдження – Freeware.

Це власницьке програмне забезпечення, котре можна Безоплатно використовувати протягом необмеженого терміну без обмежень у функціональності, і поширюване без сирцевих кодів.

Автори такого програмного забезпечення, як правило, хочуть «дати щось спільноті», але хочуть також контролювати його подальшу розробку. Іноді, коли програмісти вирішують припинити розробку, вони передають сирцевий код іншим програмістам, або ж спільноті як вільне програмне забезпечення.

Дуже часто плутають поняття «безплатне програмне забезпечення» та «вільне програмне забезпечення», хоча вони суттєво відрізняються.

Безплатне програмне забезпечення можна безоплатно встановлювати та використовувати (іноді з певними обмеженнями, як, наприклад, «безплатне для домашнього або некомерційного вжитку»), в той час як вільне програмне забезпечення можна продавати за будь-яку суму, але при тому, у користувача, котрий його отримує, повинні бути права на вивчення, модифікацію та поширення сирцевих кодів одержаної програми.

КБПЗ_2026

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		62

6 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти, призначено для системи інтелектуальної обробки відеоінформації для huddle rooms.

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

Рішення завдання полягало у вирішенні наступних задач:

- Був проведений огляд існуючих систем інтелектуальної обробки відеоінформації для huddle rooms.
- Досліджена система інтелектуальної обробки відеоінформації для huddle rooms.
- На основі отриманих результатів досліджень створена програмна реалізація системи інтелектуальної обробки відеоінформації для huddle rooms.

Розроблені під час виконання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання інтелектуальної обробки відеоінформації для huddle rooms.

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

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

Програма реалізована на мові високого рівня Visual C#. Дана мова програмування дозволяє найбільш ефективно обробляти дані призначені для системи інтелектуальної обробки відеоінформації для huddle rooms. Це дозволило мінімізувати строк розробки програмного забезпечення, і, як слід, зменшити

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		63

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 CAST-128.

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

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		64

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein. Introduction to Algorithms. The MIT Press. 2022 1677 p.
2. Will Grant. 101 UX Principles. Packt Publishing. 2022. 432 p.
3. Nathan Metzler. Kotlin Programming for Beginners. Independently published. 2021. 158 p.
4. Henry Lloyd. Interactive Computer Graphics. States Academic Press. 2022. 247 p.
5. Ranjan Parekh. Fundamentals of Image, Audio, and Video Processing Using MATLAB® With Applications to Pattern Recognition. CRC Press. 2021. 406 p.
6. Alasdair McAndrew. A Computational Introduction to Digital Image Processing. Chapman & Hall. 2021. 560 p.
7. Михайло Пічугін, Іван Канкін, Володимир Воротніков Комп'ютерна графіка. Навчальний посібник / Центр навчальної літератури 346 с. 2019р.
8. Маценко В.Г. Комп'ютерна графіка: Навчальний посібник. – Чернівці: Рута, 2009 – 343 с.
9. 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.
10. 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.
11. Kopf, Johannes and Lischinski, Dani. Depixelizing Pixel Art (англ.) // ACM Trans. Graph. – 2011. – Vol. 30, no. 4. – P. 99:1--99:8.
12. Giachetti, Andrea and Asuni, Nicola. Real-Time Artifact-Free Image Upscaling (англ.) // Trans. Img. Proc.. – 2011. – Vol. 20, no. 10. – P. 2760—2768.
13. Kuznetsov, O., Frontoni, E., Kryvinska, N., Chevardin, V., Smirnov, O. «Wireless Network Encryption Stream Ciphers, Computational Modeling, and Security

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		65

Analysis». *Computational Modeling and Simulation of Advanced Wireless Communication Systems*, 2024, pp. 379–402.

14. 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

15. Смірнова Т.В., Коноплицька-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.

16. 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.

17. 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.

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

19. 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.

20. 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.

21. Smirnov, O., Neskorođieva, T., Fedorov, E., Rudakov, K., Neskorođieva, A. «Method Detection Audit Data Anomalies on Basis Restricted Cauchy Machine» CEUR Workshop Proceedings, Volume 3187, 2022,
22. 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.
23. 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>
24. 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.
25. 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.
26. 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.
27. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova T. «Non-binary constant weight coding technique». CEUR Workshop Proceedings. Volume 2740, 2020, Pages 102-114.
28. 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.

29. 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.

30. 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.

31. 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.

32. 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.

33. 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.

34. 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.

35. 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.
36. Zhurakovskiy, 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.
37. Smirnov O. Kuznetsov A., Zaichenko Yu., Pastukhov M., Oleshko O., Kuznetsova K., «Formation of Discrete Signals with Special Correlation Properties». International Conference on Information and Telecommunication Technologies and Radio Electronics, UkrMiCo 2019; Odessa; Ukraine; 9-13 September 2019. P.22-28.
38. 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.
39. 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.
40. 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.
41. 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.
42. 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.

43. 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.

44. 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.

45. 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.

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

47. 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.

48. 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.

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

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		70

50. 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.

51. Смірнов О.А., Коноплицька-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.

КБПЗ_2026

					ВКРБ-122.26.0009.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		71