

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
« ____ » _____ 2024 р.

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
за першим (бакалаврським) рівнем вищої освіти
на тему
“Програмне забезпечення системи кібербезпеки для стеження
за мобільним пристроєм з використанням GPS”

Виконав здобувач вищої освіти
IV курсу, групи КБ-21-ЗСК
ОПП «Кібербезпека»
спеціальності 125 «Кібербезпека»
_____ Дубіцька М.М.
« ____ » _____ 2024 р.

Керівник проекту
кандидат технічних наук, доцент
_____ Смірнов С.А.
« ____ » _____ 2024 р.
Рецензент _____

Центральноукраїнський національний технічний університет

Факультет *Механіко-технологічний*

Кафедра *Кібербезпеки та програмного забезпечення*

Освітній ступінь *бакалавр*

Галузь знань . 12 *“Інформаційні технології”*

Спеціальність *125 “Кібербезпека”*

Освітньо-професійна (освітньо-наукова) програма *“Кібербезпека”*

ЗАТВЕРДЖУЮ

Завідувач кафедри

д.т.н., проф.

Олексій СМІРНОВ

« 17 » січня 2024 року

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

Дубіцькій Марині Миколаївні

(прізвище, ім'я, по батькові)

1. Тема роботи

Програмне забезпечення системи кібербезпеки для стеження за мобільним пристроєм з використанням GPS

2. Керівник роботи

Смірнов Сергій Анатолійович, канд. техн. наук, доцент

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

затверджені наказом вищого навчального закладу № 136-02 від 01.04.2024 року

3. Строк подання студентом роботи до захисту

23.05.2024 р.

4. Мета та завдання випускної кваліфікаційної роботи: *Метою роботи є розробка програмного забезпечення системи кібербезпеки для стеження за мобільним пристроєм з використанням GPS*

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 кібербезпеки в промислову експлуатацію.

6. Висновки

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

Структурна схема системи кібербезпеки

1 аркуш

Функціональна схема системи кібербезпеки

1 аркуш

Діаграма процесів

1 аркуш

Блок-схема алгоритму роботи додатку

2 аркуша

7. Дата видачі завдання « 17 » січня 2024 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	10.03.2024 р.	
2.	Постановка задачі, оформлення ТЗ	15.03.2024 р.	
3.	Розробка моделі компонента	20.03.2024 р.	
4.	Розробка структур даних	25.03.2024 р.	
5.	Розробка алгоритмів зв'язку та відображення	30.03.2024 р.	
6.	Програмування алгоритмів	10.04.2024 р.	
7.	Оформлення ПЗ	17.04.2024 р.	
8.	Попередній захист роботи	23.05.2024 р.	

Дата видачі завдання
« 17 » січня 2024 р.

Підпис керівника

Смірнов С.А.
(прізвище та ініціали)

Завдання прийнято до виконання
« 17 » січня 2024 р.

Підпис здобувача

Дубіцька М.М.
(прізвище та ініціали)

АНОТАЦІЯ

Дубіцька М.М. Програмне забезпечення системи кібербезпеки для стеження за мобільним пристроєм з використанням GPS. 125 Кібербезпека. Центральноукраїнський національний технічний університет. Кропивницький. 2024.

В даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи кібербезпеки для стеження за мобільним пристроєм з використанням GPS.

Метою розробки є програмне забезпечення системи кібербезпеки для стеження за мобільним пристроєм з використанням GPS.

Результат роботи – програмна реалізація системи кібербезпеки для стеження за мобільним пристроєм з використанням GPS.

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 мобільному пристрої під керівництвом ОС Android.

Програму розроблено в середовищі Delphi 10.4.

Ключові слова: кібербезпека, GPS, Android

ABSTRACT

Dubitska M.M. Cyber security system software for tracking a mobile device using GPS. 125 Cyber security. Central Ukrainian National Technical University. Kropyvnytskyi. 2024.

In this graduation thesis for the first (bachelor) level of higher education, software is developed, which is intended for a cyber security system for tracking a mobile device using GPS.

The goal of the development is a cyber security system software for tracking a mobile device using GPS.

The result of the work is a software implementation of a cyber security system for tracking a mobile device using GPS.

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 tools are provided.

The program can be used on a mobile device running the Android OS.

The program was developed in the Delphi 10.4 environment.

Keywords: cyber security, GPS, Android

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 кібербезпеки та мови програмування.....	25
2.3 Розгорнута постановка завдання	31
3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ	33
3.1 Опис функціонування системи	33
3.2 Розробка структурної схеми.....	38
3.3 Розробка функціональної схеми	43
3.4 Розробка діаграми процесів.....	50
4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ.....	52
4.1 Розробка блок-схем та опис алгоритмів функціонування системи.....	52
4.2 Захист розробленого програмного забезпечення.....	61
5 ВПРОВАДЖЕННЯ СИСТЕМИ КІБЕРБЕЗПЕКИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ	63
6 ОСНОВНІ ВИСНОВКИ.....	65
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	67

						ВКРБ-125.24.0040.00.00.ПЗ		
Вим.	Арк.	№ докум.	Підп.	Дата				
Розроб.	Дубіцька М.М.				Програмне забезпечення системи кібербезпеки для стеження за мобільним пристроєм з використанням GPS	Літ.	Аркуш	Аркушів
Перев.	Смірнов С.А.					Б	1	73
Н.контр.	Коваленко А.С.				ЦНТУ КБ-21-3СК			
Затв.	Смірнов О.А.							

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

LAAS – навігаційний комплекс

GPS – глобальна система місцевизначення.

GSM – система цифрового стільникового телефонного зв'язку

LAT – широта

LON – довгота

MFC – Microsoft Foundation Class library – бібліотека класів

OLE – технологія зв'язування й вбудовування об'єктів

POI – point of interest – вбудована база «цікавих» точок

SA – режим обмеженого доступу

WAAS – система супутників і наземних станцій, що робить коректування сигналів
GPS

КПК – кишеньковий персональний комп'ютер

РМД – робоче місце диспетчера

СЕР – Circular Error of Probability – окружність можливої помилки

IOD – issue of data – часовий ідентифікатор

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		2

ВСТУП

Актуальність теми. Більшість людей знайомі з фразою GPS-пристрій відстеження та його головним призначенням. Однак мало хто знає, в чому полягають основні принципи роботи GPS-трекера. Тому ми надаємо вам короткий вступ до основних принципів пристрою GPS-трекінгу.

GPS-модуль використовується для точного визначення місця розташування пристрою, а також для синхронізації часу пристрою. Якщо час пристрою відхиляється від часу GPS більш ніж на 3 секунди, пристрій повторно синхронізується з ним. Це дозволяє бути впевненим, що записи пристрою на сервері представляють події, які відбувалися саме в той час. Модулі GPS постійно виводять інформаційні пакети на мікроконтролер, що містять таку інформацію, як швидкість, висота, довгота, широта та якість сигналу GPS.

Модуль GSM – це зв'язок, що з'єднує GPS-трекінг із світом. Для цього користувач повинен вставити в пристрій робочу SIM-карту. Комунікація складається з двох основних функцій:

1. GSM зв'язок. GSM-з'єднання встановлюється, коли вставлено робочу SIM-карту та доступний оператор, до якого можна підключитися.
2. GPRS зв'язок. GPRS-з'єднання використовується для зв'язку пристрою з сервером.

Перш ніж кожен запис надсилається на сервер, він зберігається у флеш-пам'яті пристрою (або SD-карті, якщо вставлено). У більшості випадків пристрій надсилає свої записи на сервер, отримує підтвердження про отримання записів, а потім видаляє їх.

Акcesуари – це периферійні пристрої, які використовуються для розширення функціональності пристрою відстеження GPS. Наприклад, за допомогою різних акcesуарів ви можете контролювати температуру в різних відсіках причепа, реєструвати та ідентифікувати водіїв, стежити за рівнем палива

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		3

безпосередньо з паливного баку, відстежувати відкриття та закриття дверей причепа та багато іншого.

Мета й завдання дослідження. Метою роботи є програмне забезпечення системи кібербезпеки для стеження за мобільним пристроєм з використанням GPS.

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

- Огляд існуючих систем для стеження за мобільним пристроєм з використанням GPS.
- Дослідження системи кібербезпеки для стеження за мобільним пристроєм з використанням GPS.
- Програмна реалізація системи кібербезпеки для стеження за мобільним пристроєм з використанням GPS.

Практична цінність отриманих результатів полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі для стеження за мобільним пристроєм з використанням GPS.

Таким чином, виходячи з вищеперерахованого, програмне забезпечення системи кібербезпеки для стеження за мобільним пристроєм з використанням GPS, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти.

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		4

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Система призначена для стеження за мобільним пристроєм з використанням GPS. Для того, щоб використовувати програмне забезпечення GPS спостереження для роботи, потрібно спочатку купити GPS пристрій, що може бути розміщене на автомобілі або на будь-якому іншому рухливому об'єкті, або мобільний пристрій з наявністю GPS-модуля. Хоча й немає необхідності вкладати в них занадто багато, треба мати на увазі, що якщо вони перестануть працювати або втратять сигнал, програма GPS спостереження більше не будуть мати можливість стежити за ними. Більшість сайтів, які продають пристрої GPS спостереження, пропонують почитати відгуки клієнтів про товар. Відгуки є важливим джерелом інформації про те, будуть чи ні ці пристрої працювати усередині будинків і є чи території, у яких прийом сигналу залишає бажати кращого. Більшість програм GPS моніторингу будуть працювати практично з будь-яким пристроєм, але краще, звичайно, перевірити на сумісність перед покупкою. Деякі мобільні пристрої поставляються з безкоштовним програмним забезпеченням. У більшості випадків, це програмне забезпечення відразу готове до роботи, без особливої необхідності в налаштуванні. Цивільне програмне забезпечення, що призначене для відстеження тільки декількох машин або стільникових телефонів, як правило, просте у використанні й займає не більше декількох минут установки. Поки комп'ютер або мобільний телефон підключені до Інтернету, розташування об'єкта спостереження можна побачити за допомогою простого натискання однієї кнопки. На додаток до моніторингу в режимі реального часу, більшість програм, які в цей час присутні на ринку, дозволяють записувати активність пристрою спостереження, інакше кажучи, ви зможете побачити пройдений об'єктом спостереження маршрут на карті міста.

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		5

Для підприємств програми GPS спостереження за автотранспортними засобами мають неоціненне значення. Вони дозволяють роботодавцям відслідковувати автомобілі компанії й переконатися, що їх використовують безпечно й правильно. Система може гарантувати, що мобільні співробітники використовують службові автомобілі належним чином. Багато будівельних і транспортних компаній оснащують програмою GPS моніторингу кожний автомобіль у своєму парку. У більшості випадків, ці транспортні засоби представляють найбільші цінності компанії. Коли відбувається крадіжка, транспортні засоби можуть бути швидко й легко знайдені за допомогою програм GPS спостереження за транспортними засобами. Налаштування програми GPS моніторингу звичайно справа проста. Крім усього іншого, існує ряд корисних опцій, які можуть скоротити час, затрачений на моніторинг вашого об'єкта GPS спостереження. Ви можете настроїти систему оповіщень, що повідомляє вам місцезнаходження автомобіля, що відслідковується, у випадку, якщо воно виїхало за межі маршруту або яка подає сигнал при перевищенні швидкості спостережуваного автомобіля. GPS спостереження може бути використане для моніторингу всього, що ви вважаєте коштовним. Батьки можуть використовувати його на автомобілі своєї дитини або на його мобільному телефоні. Підприємства можуть відслідковувати кожний свій транспортний засіб або безпосередньо співробітника, перевіряючи, як він виконує доручене йому завдання. Програми GPS спостереження – відмінне рішення, що полегшує життя.

1.2 Область застосування

Смартфони містять GPS-приймачі та передавачі, які дозволяють їм, як телефонам iOS, так і Android, отримувати доступ до навігації в реальному часі та інших служб на основі визначення місцезнаходження. У разі втрати або крадіжки мобільного телефону ви також можете віддалено активувати GPS-трекер, щоб активно відстежувати місцезнаходження мобільного телефону. Це означає, що ви

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		6

можете будь-коли перевірити цільовий пристрій, чи він живиться та підключений до бездротової мережі. Кілька облікових записів також можна об'єднати, щоб друзі та родина могли стежити один за одним.

Очевидно, що служба відстеження місцезнаходження GPS не зможе працювати, якщо акумулятор повністю розряджено, але служба може зареєструвати останнє відоме місцезнаходження пристрою. Служби відстеження мобільних телефонів додано до телефонів iOS і Android, але функції дещо відрізняються на кожному з них. Давайте розберемося, **як відстежувати телефон** або планшет на обох платформах.

Відстеження iPhone

Відстеження Apple увімкнено за допомогою програми «Find My», яку можна поєднати з пристроями iOS, такими як годинники Apple, комп'ютери Mac, AirPods і AirTags. Для iOS доступні такі функції відстеження:

- Переглянути розташування.
- Відтворення звуку.
- Позначити як втрачене (режим втрати).
- Віддалене стирання.
- Сповістити, коли знайдено.
- Сповістити, коли залишиться позаду.

Щоб використовувати ці функції, вам знадобиться другий пристрій, прив'язаний до вашого облікового запису Apple.

Налаштування Find My на iOS і Mac за допомогою служби підтримки Apple:

1. Відкрийте програму Налаштування.
2. Торкніться свого імені, а потім торкніть «Знайти».
3. Якщо ви хочете, щоб друзі та родина знали ваше місцезнаходження мобільного телефону, увімкніть параметр «Надіслати моє місцезнаходження».
4. Торкніться «Знайти мій [пристрій]», а потім увімкніть «Знайти мій [пристрій]».

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		7

5. Щоб бачити свій пристрій навіть у режимі офлайн, увімкніть функцію «Знайти мою мережу». (Мережа «Знайти мене» – це зашифрована анонімна мережа сотень мільйонів пристроїв Apple, яка може допомогти вам знайти свій пристрій.)

6. Щоб дані про місцезнаходження вашого пристрою надсилалися в Apple, коли заряд акумулятора низький, увімкніть параметр «Надіслати останнє місцезнаходження».

Якщо ви хочете знайти свій втрачений пристрій на карті, переконайтеся, що функцію локації ввімкнено. Для цього перейдіть у «Налаштування» > «Конфіденційність і безпека» > «Служби локації» та ввімкніть «Служби локації».

Якщо ви користуєтеся програмою Find My, ви не зможете побачити місцезнаходження свого пристрою, якщо минуло більше семи днів з моменту останнього надсилання його місцезнаходження в Apple.

Якщо iPhone нелегко відновити, ви також можете виконати дистанційне стирання:

1. Переконайтеся, що ви виконали наведені вище дії, щоб відновити втрачений пристрій – і зауважте, що цю дію неможливо скасувати.
2. Відкрийте програму «Знайти мене» та виберіть вкладку «Пристрої».
3. Виберіть пристрій, який потрібно віддалено стерти.
4. Прокрутіть униз і виберіть «Стерти цей пристрій».
5. Виберіть Видалити цей [пристрій].

Якщо пристрій працює під керуванням iOS 15 або новішої версії, ви все одно зможете ідентифікувати та відстежувати його в програмі Find My після видалення його вмісту.

Відстеження телефону Android

Функція Android «Знайти мій пристрій» дозволяє користувачам відстежувати смартфон або планшет, блокувати його віддалено або стерти дані з пристрою. Все це можна зробити в Інтернеті за допомогою облікового запису

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		8

Google, підключеного до цільового пристрою. Служба підтримки Google описує кроки, які необхідно виконати, щоб відстежувати пристрої Android.

Щоб знайти пристрій Android, переконайтеся, що він:

- увійшов в обліковий запис Google;
- має ввімкнене місцезнаходження;
- має функцію «Знайти мій пристрій»;
- має силу;
- підключено до мобільного передавання даних або WiFi;
- відображається в Google Play.

Якщо ваш пристрій вимкнено або не під'єднано до мобільного передавання даних чи Wi-Fi, ви все одно можете знайти його останнє місцезнаходження, якщо ввімкнули «Зберігати останнє місцезнаходження».

Переконавшись, що всі ці налаштування правильні, можна розпочати процес відстеження:

1. У браузері перейдіть на android.com/find або завантажте програму «Знайти мій пристрій» на другий телефон із магазину Google Play.

2. Увійдіть у свій обліковий запис Google.

– Місцезнаходження приблизне і може бути неточним.

– Якщо ваш пристрій не вдається знайти, ви можете знайти його останнє відоме місцезнаходження, якщо воно доступне.

далі:

1. Загублений телефон Android отримує сповіщення.

2. На карті ви отримаєте інформацію про те, де знаходиться пристрій.

– Якщо у вас більше одного пристрою: у верхній частині бічної панелі виберіть втрачений пристрій.

– Якщо на вашому втраченому пристрої є кілька профілів користувача, увійдіть за допомогою облікового запису Google, який є в основному або особистому профілі.

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		9

Після цього виберіть план дій, які ви хочете зробити:

– Якщо з'явиться запит, торкніться значка «Увімкнути блокування та стирання» .

– Відтворити звук: ваш пристрій дзвонить на повну гучність протягом п'яти хвилин, навіть якщо на ньому встановлено беззвучний режим або вібрацію.

– Захистіть пристрій: блокує ваш пристрій за допомогою PIN-коду або пароля. Якщо у вас немає замка, ви можете встановити його. Щоб допомогти комусь повернути вам ваш пристрій, ви можете додати повідомлення або номер пристрою на екран блокування.

– Очистити пристрій: назавжди видаляє всі дані на вашому пристрої, але може не видаляти SD-карти. Після стирання функція «Знайти пристрій» не працюватиме на пристрої.

Таким чином, виходячи з вищеперерахованого, програмне забезпечення системи кібербезпеки для стеження за мобільним пристроєм з використанням GPS, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти.

КБПЗ-2024

					ВКРБ-125.24.0040.00.00.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

Датчик GPS у мобільних пристроях можна використовувати не тільки по прямому призначенню (для навігації або визначення координат), але й у ряді інших, часто неочевидних на перший погляд, завдань.

Оскільки в багатьох випадках відразу кілька програм уміють виконувати схожі завдання, то докладний опис давався лише для однієї-двох, а інші перераховувалися в «Альтернативах».

Передача координат

Прості, але корисні утиліти для швидкої передачі своїх координат іншим людям.

Here I Am 2

Утиліта припускає такий сценарій використання: на телефоні включається GPS, пристрій визначає координати користувача й відображає його місце розташування на карті. Отриману інформацію, а також посилання на Google Maps можна відправити через SMS, пошту або інші сервіси друзям, знайомим, родичам.

GPS координати виду 54.5235, 46.2335 розпізнаються смартфонами навіть в SMS-повідомленнях, після чого перетворюються в посилання, які можна відкрити прямо в браузері або на карті, тому відправлення посилання на Google Maps у багатьох випадках навіть не буде потрібно.

Альтернативи: GPS Share, Wizi SMS Location.

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		11

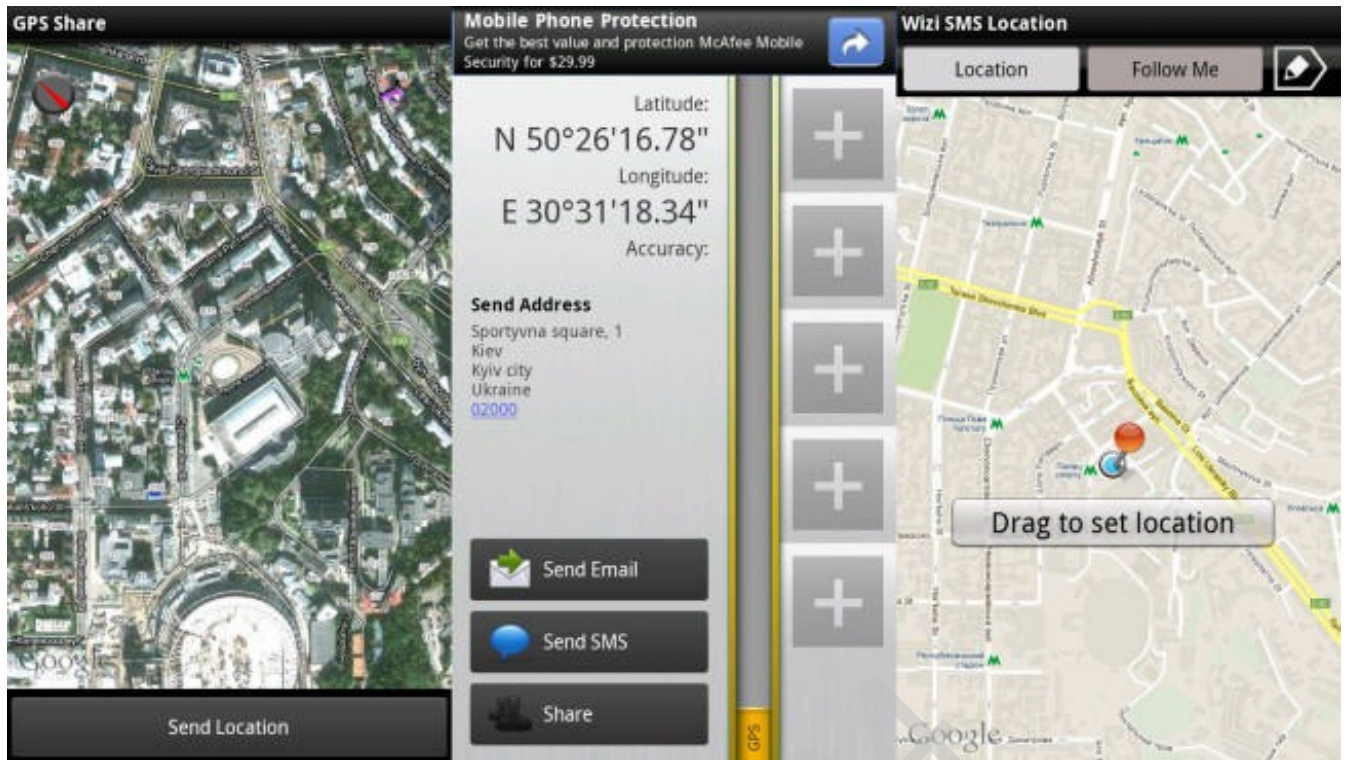


Рисунок 2.1 – Інтерфейси користувача програм передачі координат

Батьківський контроль / координати знайомих

GPS Tracking

Програма, встановлена на телефон дитини (родича, знайомого) дозволяє в будь-який момент довідатися, де він зараз перебуває. Найбільш затребувана утиліта, природно, буде в молодих батьків, які хочуть бути впевненими в тому, що їхні діти не втрапилися.

З огляду на, що в деяких місцях GPS покриття працює навіть при відсутності мобільного зв'язку, така програма буде корисна й у тих випадках, коли на телефон не виходить додзвонитися.

GPS Tracker

GPS Tracker автоматично відправляє SMS-повідомлення або електронний лист батькам, коли їхня дитина відвідує або залишає певні місця (наприклад, школу). Таким чином, необов'язково вручну займатися моніторингом дітей.

Альтернатива: Family Locator.

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		12

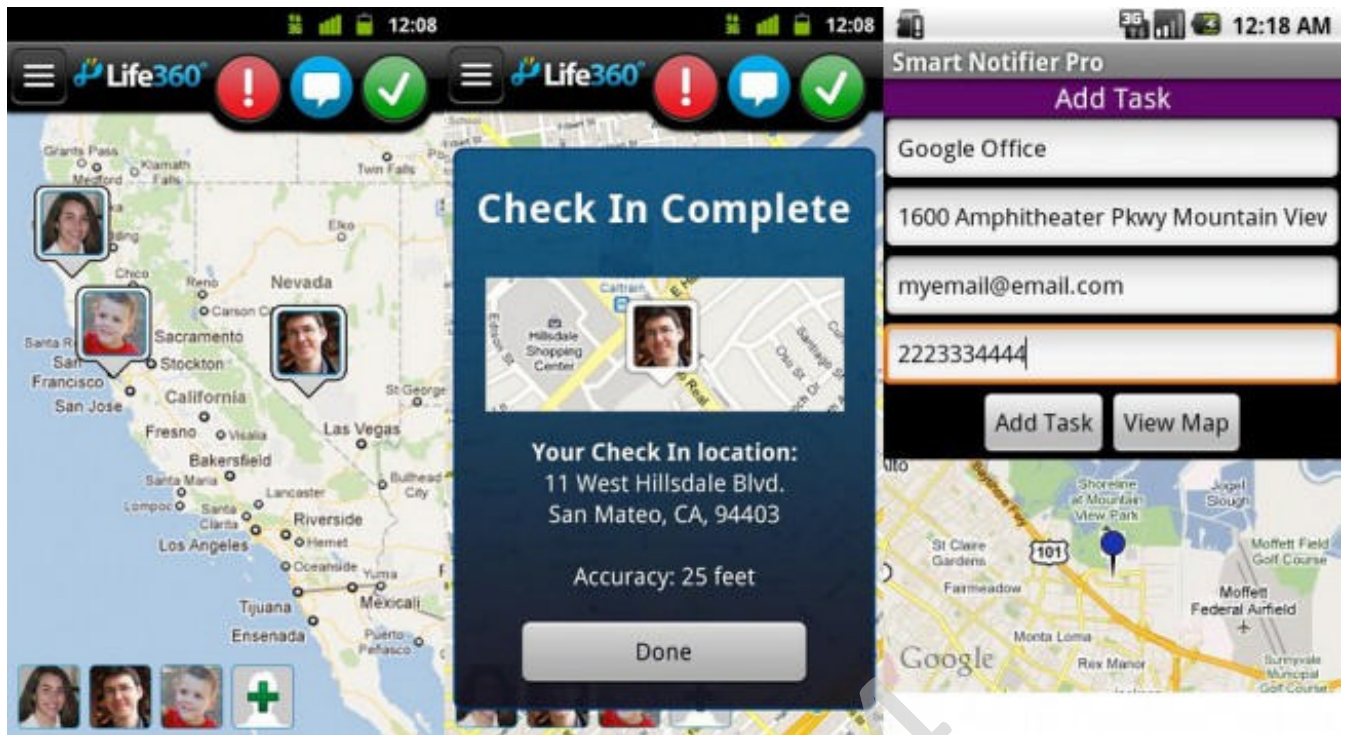


Рисунок 2.2 – Інтерфейси користувача програм батьківського контролю / координат знайомих

Підключення GPS

bluegps4droid

Утиліта надається в тому випадку, якщо в телефоні немає GPS-приймача, але є присутнім Bluetooth, за допомогою якого пристрій і можна перетворити в навігатор. Bluegps4droid може працювати й при наявності внутрішнього GPS, продовжуючи одержувати всі дані із зовнішнього приймача. Це може пригодиться в тих випадках, коли зовнішній GPS має кращу чутливість або працює від звичайних батарейок.

Bluetooth GPS Provider

Практично аналог bluegps4droid, але з невеликою перевагою – програму можна встановити прямо з Android Market. Як і в попередній утиліті, розроблювач попереджає, що деякі додатки можуть працювати некоректно із зовнішнім GPS-датчиком, але це вже проблема самих програм.

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		13

GPS over BT

Утиліта що дозволяє приєднувати смартфон до ПК по Bluetooth і використовувати зовнішній GPS датчик у таких додатках як Google Earth, 2Gis, SasPlanet і багатьох інших.

ShareGPS Beta

ShareGPS Beta також підтримує підключення до ПК по Bluetooth, але крім цього дозволяє працювати й по USB.



Рисунок 2.3 – Інтерфейси користувача програм підключення GPS

Виконання різних завдань залежно від координат

Ллама – Location Profiles

Для тих, у кого телефон часто голосно дзвонить у самий невідповідний момент, випущена програма Ллама, що вміє управляти параметрами дзвінка залежно від того, де перебуває користувач. У поліклініці, бібліотеці, на нараді або в будь-якому іншому зазначеному місці телефон може автоматично переходити в

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		14

тихий, безшумний або вібро-режим залежно від налаштувань, і його власникові більше не прийде турбуватися про те, щоб щораз відключати звук самостійно.

Можливості програми виходять за рамки керування гучністю телефону в певних місцях. Тригером для будь-яких передбачених дій може служити день тижня, наявність сигналу від певної Wi-Fi мережі, час доби (телефон може автоматично переходити в безшумний режим уночі, при цьому програму можна настроїти таким чином, щоб дзвінки від родичів і родини однаково були голосними).

Особливістю Llama є те, що вона не використовує GPS для визначення координат, а покладається на стільникові вишки. Проте, вирішено було включити неї в огляд, як альтернативу GEO-Tasker.

GEO-Tasker

GEO-Tasker виконує в цілому ту ж роботу, що й Llama, але при цьому використовує GPS-датчик, а для користувача із цього випливає як більше точне визначення координат, так і принципова можливість роботи вдалині від населених пунктів.

Утиліту можна використовувати для регулювання гучності дзвінка, відправлення SMS, запуску обраної програми, виводу нагадувань, автоматичного включення Wi-Fi датчика в місцях з безкоштовним бездротовим Інтернетом, активації будильника при під'їзді до своєї зупинки/станції.

GPS Alarm

GPS Alarm має тільки функцію будильника, але за рахунок цього програму легко набудувати. В утиліті є й ще одна незаперечна перевага: для визначення координат за замовчуванням використовуються тільки стільникові вишки, і лише після того як відстань до пункту призначення скорочується до 5 км, у телефоні активується GPS, що дозволяє заощаджувати батарею. Для роботи GPS Alarm не потрібне інтернет-з'єднання.

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		15



Рисунок 2.4 – Інтерфейси користувача програм виконання різних завдань залежно від координат

Утиліти / різне

GpsFix

Іноді при запуску додатків, що використовують GPS-датчик, час визначення координат може становити кілька хвилин. Щоб уникнути написів «waiting for location», рекомендується попередньо запускати утиліту GpsFix, що встановлює точні координати користувача й «фіксує» їх для інших програм.

Альтернатива: GPS Test.

GPS Status & Toolbox

Утиліта дозволяє одержувати прямий доступ до наступної GPS даних: точні координати, швидкість, прискорення, потужність сигналу, точність визначення, градус нахилу телефону. За допомогою убудованого електронного компаса також можна довідатися, де перебуває магнітний або справжній північний полюс.

Аналоги: Ulysse Gizmos, GPS Essentials.

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		16

Geotag Photos Lite

Далеко не всі фотоапарати мають убудований GPS-датчик, тому в знімках відсутня інформація, що дозволяє пізніше прив'язати їх до конкретних місць на карті (в Picasa, iPhoto або Aperture). За допомогою Geotag Photos Lite координати можна записувати окремо в смартфоні, і потім синхронізувати їх (за часом) з фотографіями, зробленими будь-якою фотокамерою. Програма підтримує всі розповсюджені формати зображень, включаючи RAW, і має кілька режимів роботи GPS, що дозволяє при незначному зниженні точності продовжити час автономної роботи.

GPSies

Клієнт GPSies дозволяє використовувати більшість можливостей сайту (наприклад, шукати цікаві маршрути в окрузі) безпосередньо на телефоні.

SpeedView

SpeedView показує не тільки швидкість переміщення, що вмiють робити й багато інших програм, але також відображає максимальну й середню швидкість, загальну дистанцію й час у дорозі. Додатково програма вмiє подавати сигнал у випадку перевищення одного із трьох установлених лімітів швидкості.

Альтернатива: Ulysse Speedometer.

Fake GPS location

Якщо потрібно протестувати яку-небудь програму з довільними GPS-Координатами, то це можна зробити за допомогою Fake GPS location. Виберіть будь-яку точку на карті миру – і вона буде вважати, що користувач дійсно перебуває там.

Щоб деякі програми не змогли правильно визначати координати по Wi-Fi або стільникових вишках, у системних налаштуваннях телефону можна відключити цю функцію або просто перевести його в Airplane mode.

Where I Am Widget Demo

Віджет, що відображає карту й положення користувача прямо на домашньому екрані телефону. Where I Am Widget Demo підтримує карти Google Maps, Bing і Yahoo.

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		17

Back2It

Гарна програма для того, щоб без проблем повернутися у вихідну точку в незнайомому місті. В Back2It, наприклад, можна внести координати припаркованої машини й потім швидко знайти її.

Альтернатива: MyCar Locator Free.

Location Cache

Напевно, усі пам'ятають скандал, що виник після звісток про те, що iPhone і Android зберігають дані про переміщення користувача в пам'яті телефоні. Щоб подивитися, яка інформація зберігається у вашій пристрої, необхідно запустити Location Cache на смартфоні з root-доступом, і програма відобразить всі доступні дані як цифровому виді, так і безпосередньо на карті. Збережений кеш можна очистити й навіть заблокувати його створення в майбутньому.

Альтернатива: Android Location Cache Viewer.

RunKeeper

Програма для тих, хто як активний вид спорту вибрав біг або їзду на велосипеді. RunKeeper веде облік перебореної дистанції, швидкості, витраченого часу, витрачених калорій, а також уміє аналізувати пульс при використанні зовнішніх датчиків і записує маршрут переміщення. Додаток можна настроїти таким чином, щоб через кожні кілька хвилин або кілометрів користувач чув у навушниках голосове повідомлення. Якщо пройти необов'язкову реєстрацію, то інформацію про кожну пробіжку можна буде завантажувати на сайт для більше детального аналізу й порівняння в майбутньому зі своїми новими результатами.

Альтернативи: iMapMyRUN Run, AllSport GPS FREE.

Share Address

Якщо ви перебуваєте в будинку й не знаєте його адреси, то за допомогою Share Address можна одержати всю необхідну інформацію у звичному виді (назва вулиці, номер будинку, область, місто, країна, індекс). Дані потім можна додати в будь-який контакт із адресної книги, передати за допомогою будь-якого системного сервісу (пошта, SMS, Facebook і т.д.) іншим користувачам, додатково

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		18

додавши до вже перерахованих даних посилання на Google Maps і точні GPS-Координати. Точно так само можна відправити й сам знімок карти у форматі jpg, попередньо вибравши звичайний або супутниковий режим відображення.

Share Address може відправляти дані періодично в автоматичному режимі, і при виборі одного з більш ніж 50 підтримуваних мов адреса буде відображатися в зазначеному регіональному форматі.

kdANDo

Зручна програма для того, щоб зустрітися із друзями в певнім місці. Указавши будь-яку точку на карті (при необхідності утиліта автоматично визначає адресу найближчого будинку), можна вибрати час зустрічі й відправити через SMS, пошту (насправді в цьому пункті доступні й деякі додаткові сервіси, наприклад, Evernote і VKontakte), Facebook, або Twitter запрошення тим, хто повинен прийти. У повідомленні буде зазначена адреса, посилання на Google Maps і ще одне посилання на Google Directions, що прокладе на карті шлях від початкового до кінцевого пункту.



Рисунок 2.5 – Інтерфейси користувача програм утиліт / різне

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		19

Запис маршрутів

Нагадаємо, що всі необхідні опції для запису треків доступні в більшості карт, розглянутих у цьому матеріалі.

My Tracks

Проста утиліта, призначена винятково для запису маршруту користувача. Треки можна експортувати у формати GPX, KML, CSV, TC або завантажувати прямо в Google Maps, де їх потім можна подивитися на будь-якому комп'ютері після авторизації під тим же обліковим записом Google. Як звичайно, файли можна прямо відправляти іншим користувачам за допомогою будь-якого системного сервісу передачі даних.

Open GPS Tracker

Ця утиліта в цілому повторює можливості My Tracks, але додатково дозволяє перемикатися між різними картами (супутникова, Mapnik, CloudMade, OpenCycleMap), і відображає деяку додаткову інформацію (швидкість, висоту, пробки, компас). Окремий екран статистики показує всю необхідну інформацію про маршрут (графік швидкості, час переміщення, дистанцію, перепади висот і т.д.). Open GPS Tracker підтримує інтеграцію з www.gobreadcrumbs.com, куди програма завантажує треки, а також залишені на карті замітки з аудіо, відео, фото й текстом.

Live GPS Tracker

Відрізняється від інших програм тим, що маршрут можна завантажувати на сайт <http://livegpstracks.com> в автоматичному режимі ще в міру його проходження, а не вручну після закінчення запису. При цьому Live GPS Tracker стискає проміжні дані для економії трафіку й автоматично активує/деактивує з'єднання з Інтернетом під час передачі. Треки можуть бути приватними (доступними тільки користувачеві), для друзів або повністю відкритими. Програма має налаштування енергозбереження, а також опції, що уточнюють, як часто необхідно завантажувати файл на сервер (через певний час, після подолання заданої дистанції або вручну).

Альтернативи: GPSTracker Lite, AndAndo.

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		20

Plan B

Якщо попередні утиліти необхідно було встановлювати до втрати телефону, то Plan B можна використовувати й після того, як це трапилось. Відкривши Android Market на ПК, необхідно натиснути кнопку Install на сторінці Plan B, і якщо на телефоні активований Wi-Fi або можлива установка програм через 2G/3G-з'єднання, то вона почне інстальоватися автоматично. Це справедливо й для інших програм в Android Market, але на відміну від них, Plan B запускається відразу після установки й протягом наступних 10 хвилин відсилає координати пристрою й відстань до телефону на пошту користувача. Навіть якщо новий «власник» телефону зрозуміє, чим займається утиліта, він однаково не встигне перервати як мінімум першу передачу даних. Якщо ж користувач просто не розбирається в мобільних телефонах, то після закриття вікна вітання він навіть не буде знати про те, що програма продовжує працювати, оскільки в статус-барі її іконка не відображається.



Рисунок 2.7 – Інтерфейси користувача програм пошуку загубленого телефону

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		22

Щоб повторити 10-хвилинний сеанс передачі координат, необхідно відправити на телефон спеціальне слово в SMS-повідомленні. В Plan B воно вже задано за замовчуванням, і цим програма відрізняється від своїх альтернатив.

Plan B уміє автоматично активувати GPS-датчик на деяких телефонах, однак якщо це неможливо, те визначення координат буде відбуватися по стільникових вишках.

Альтернативи: Lost Phone, Lookout Security & Antivirus, GPS Tracker, GPS Tracker.

Спостереження

Цей розділ виділений для утиліт, що дозволяють у реальному часі обновляти інформацію про своє місце розташування й надавати доступ до неї іншим людям. Схожі функції, звичайно, можна одержати й від утиліт для пошуку загубленого телефону або додатків батьківського контролю, однак вони спеціалізовані для інших завдань, і тому не настільки зручні.

Glympse

Glympse генерує спеціальне посилання на сайт glympse.com (реєстрація не потрібно), яку можна відправити комусь по SMS, пошті або за допомогою будь-якого іншого сервісу, встановленого на Android. Лінк відкриває в браузері карту, де в режимі реального часу відображається положення відправника.

Передача координат за замовчуванням працює протягом 30 хвилин (тривалість налаштовується), після чого сторінка перестає обновлятися. Користувач також може вказати на карті кінцевий пункт свого маршруту, щоб його можна було перехопити десь по шляху.

Glympse – одна із самих корисних і цікавих програм в огляді, тому не дивно, що вона відзначена значком EDITORS' CHOICE в Android Market.

BattleTac

Утиліта розроблена спеціально для аматорів ейрсофту й пейнтболу. BattleTac дозволяє бачити на екрані положення всіх гравців своєї команди й передавати швидкі повідомлення й вказівки для координації дій. Бета-версія програми поширюється безкоштовно.

Альтернатива: Real Time GPS Tracker

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		23

Розширена реальність

Wikitude

Wikitude відображає віртуальні значки з підписами прямо на екрані смартфона, накладаючи їх на картинку, отриману з убудованої камери. Сервіс працює й у наших реаліях і дозволяє шукати певні послуги й сервіси (банкомати, магазини, ресторани) або просто розглядати все, що перебуває навколо.

Google Sky Map (Star Chart)

Програма дозволяє вивчати зоряне небо, відображаючи на екрані сузір'я й планети. Деякі небесні об'єкти можна знайти по ім'ю через пошук, а в галереї доступні відомі «визначні пам'ятки» (наприклад, туманність орла або галактика Андромеда), положення яких на небі програма виділить стрілками.

Дуже хотілося розповісти й про програму Star Chart, однак, на відміну від Google Sky Map, вона платна.

Альтернатива: Layar.

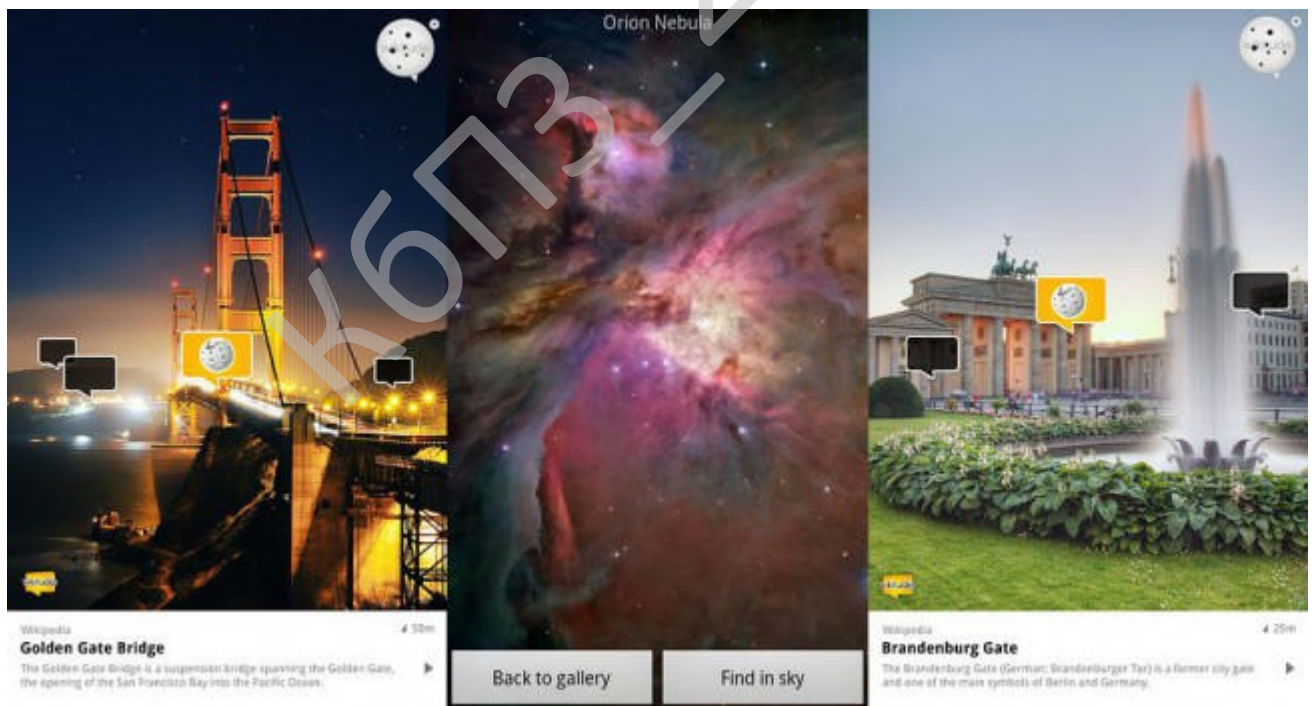


Рисунок 2.8 – Інтерфейси користувача програм розширеної реальності

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		24

Якщо програми довго визначають координати пристрою на відкритому просторі, рекомендуємо попередньо запускати утиліти GPS Fix або GPS Test, про які можна почитати вище, або активувати можливість передачі даних по GPRS/2G/3G каналу. Повірте, це допомагає.

Під час запису маршрутів удалині від цивілізації є зміст переводити телефон у режим Airplane mode для відключення модуля стільникового зв'язку. Мобільний зв'язок у таким місцях, як правило, і так практично недоступний, а от зекономленого заряду батареї цілком може вистачити для безперервного запису треку протягом цілого світлового дня.

Якщо доступ в Інтернет з мобільного пристрою відсутній, але хочеться в реальному часі бачити своє переміщення на карті, можна використовувати одну із програм, що підтримують оффлайн-карти (наприклад, RMaps або Maverick).

2.2 Обґрунтування вибору засобів для побудови системи кібербезпеки та мови програмування

Embarcadero Delphi, раніше Borland Delphi і Codegear Delphi, – інтегроване середовище розробки ПЗ для Microsoft Windows, Mac OS, iOS і Android мовою Delphi (що раніше носила назву Object Pascal), створена спочатку фірмою Borland і на даний момент приналежна й розроблювальна Embarcadero Technologies. Embarcadero Delphi є частиною пакета Embarcadero RAD Studio і поставляється в чотирьох редакціях: Community (поширюється безкоштовно й має обмежену ліцензію на використання в комерційних цілях), Professional, Enterprise і Architect.

Delphi 10.4 Sydney

Випущено 26 травня 2020 року. RAD Studio Delphi 10.4 забезпечує значно поліпшену високопродуктивну нативну підтримку Windows, кращу продуктивність розробки, миттєві підказки code completion, прискорення виконання коду із синтаксисом керованих записів, поліпшення виконання паралельних завдань на сучасних багатоядерних CPU, а також містить більш 1000

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		25

виправлень багів, поліпшення продуктивності середовища й бібліотек і багато чого крім того.

Основні можливості Delphi 10.4.1:

– Істотні розширення для Windows: поліпшення для застосунків на моніторах 4K High DPI, інтеграція з новим WebView2 на базі Chromium, використання розширених title bars, таких же, як в Office, Explorer, Google Chrome.

– Керування пам'яттю в Delphi тепер стандартизоване на всіх підтримуваних платформах – мобільних, настільних і серверних – використовувачи класичну реалізацію керування пам'яттю об'єктів.

– Істотне поліпшення Delphi Code Insight (без можливого блокування IDE – в окремому процесі), що допоможе при роботі з великими проектами.

– Тип даних Delphi «record» тепер підтримуватимуть довільні ініціалізацію, фіналізацію й операції копіювання.

– Розширена підтримка бібліотек C++: ZeroMQ, SDL2, SOCI, libSIMDpp і Nematode.

– Відладник Win 64 (на LLDB) і збирач для C++.

– Поліпшення для C++: Включена велика кількість поліпшень STL з Dinkumware.

– Підтримка Metal Driver GPU для macOS і iOS.

– Вбудований Fmxlinux.

– Компонент Twebbrowser для iOS тепер реалізований на Wkwebview API.

Реалізація компонента Media Player для macOS тепер використовує Avfoundation.

Реалізований заново стилізуємий FMX компонент TМемо на платформі Windows значно поліпшений і тепер має відмінну підтримку IME.

– Численні поліпшення швидкості й стабільності роботи нашої бібліотеки The Parallel Programming Library (PPL).

– Додані оновлені драйвери для FireBird, PostgreSQL і SQLite.

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		26

– Клієнтські бібліотеки HTTP і REST Client розширені застосунковими можливостями роботи з HTTPS. Також були розширені можливості підтримки Amazon AWS services

– У технологію Visual LiveBindings внесена безліч поліпшень, у тому числі швидкодії, що стосуються, застосунків на VCL і FireMonkey

RAD Studio 10.4 Короткий огляд:

– Істотні розширення для Windows. Створення застосунків, що чудово виглядають, із чіткими елементами інтерфейсу на 4k моніторах High DPI за допомогою нової гнучкої підтримки стилів елементів керування на екрані. Інтеграція із сучасними, безпечними web-технологіями від Microsoft – новим WebView2 на базі Chromium. Використання сучасних розширених title bars, таких же, як в Office, Explorer, Google Chrome, у своїх проектах. Істотні поліпшення надійності налагодження в новому відладнику для C++ Windows 64-bit.

– Зросла продуктивність розробки. Ріст продуктивності за рахунок миттєвої реакції підказок code completion у середовищі IDE. Краща сумісність із уже наявною кодовою базою, і спрощення програмування за рахунок уніфікованої архітектури керування пам'яттю. Швидке зв'язування даних і візуальних елементів за допомогою розширеної технології Visual LiveBindings з підвищеною швидкістю. Просте використання розповсюджених бібліотек C++, наприклад, ZeroMQ, SDL2, SOCI, libSIMDpp і Nematode. Оновлена підтримка Amazon AWS cloud.

– Поліпшення швидкодії і якості. Більш 1000 поліпшень швидкодії і якості. Краща ефективність коду за допомогою нового синтаксису custom managed records. Більш швидке виконання паралельних завдань на сучасних багатоядерних CPU. Переконаєтеся в прискоренні відображення на екрані з підтримкою Metal API на macOS і iOS. Краща сумісність із уже наявною кодовою базою й спрощення програмування за рахунок уніфікованої архітектури керування пам'яттю.

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		27

Істотне поліпшення Delphi Code Insight

Як найбільше й головне поліпшення інструментів програмування Delphi за багато років, в 10.4 Delphi Code Insight реалізований через Language Server Protocol (LSP). LSP – це технологія генерації результатів для code completion, навігації й інших сервісів в окремому процесі. Це значить, що code completion і Code Insight одержать більш точні результати без блокування IDE. 10.4 забезпечує набагато більш високу продуктивність розроблювачів, які працюють із більшими проектами, що містять мільйони рядків коду.

Delphi Custom Managed Records

Ключове розширення мови Delphi: тип даних Delphi «record» тепер підтримуть довільні ініціалізацію, фіналізацію й операції копіювання. Управляйте тем, як ці структури створюються, копіюються й звільнюються з допомогу вашого коду, який буде виконуватися у відповідний момент.

Це розширює потужність конструкцій records в Delphi, які використовуються щоб одержати більшу ефективність у порівнянні із класами.

Єдине керування пам'яттю

Керування пам'яттю в Delphi тепер стандартизоване на всіх підтримуваних платформах – мобільних, настільних і серверних – використовувачи класичну реалізацію керування пам'яттю об'єктів.

У порівнянні з Automatic Reference Counting (ARC), це дає кращу сумісність із існуючим кодом і спрощує написання компонентів, бібліотек і застосунків.

ARC модель керування пам'яттю model залишилася для керування рядками й посиланнями на тип інтерфейсу на всіх платформах. Для C++ це означає, що при створенні й звільненні Delphi-style класів в C++ використовується звичайне керування пам'яттю, як у будь-якого heap-allocated класу C++, що значно знижує складність коду.

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		28

Розширена підтримка бібліотек C++

В 10.4 ми портували багато популярних бібліотек C++ у C++Builder.

Забезпечивши оптимізовану підтримку бібліотек ZeroMQ, SDL2, SOCL, libSIMDpp і Nematode, поряд із уже підтримуваними Boost і Eigen, які можуть бути додані за допомогою менеджера пакетів Getit.

Win 64-відладник і збирач для C++

В 10.4 з'явився новий відладник C++ для Windows 64-bit. Відладник заснований на LLDB і показує значне збільшення стабільності при налагодженні 64-bit застосунків поряд з новими відладочними можливостями, такими як перегляд і інспекція типів начебто рядків C++ і Delphi, а також колекцій STL, включаючи std::vector, std::map і інших. Крім того, згенерована для застосунку відладочна інформація має інший внутрішній формат, сприяючи більш стабільному й багатому на можливості процесу налагодження, більш докладним перегляду й інспекції в debug-time.

Підвищення якості й швидкодії інструментів

- Велика кількість поліпшень STL від Dinkumware.
- Поліпшені деякі найважливіші методи й області RTL, на базі поліпшень сумісності з популярними бібліотеками C++.
- Поліпшена підтримка Cmake.
- Велика кількість виправлень для підвищення стабільності і якості.
- Відновлення Windows API – Обновлено й додали безліч декларацій API щоб добитися ще більшої інтеграції із платформою Windows.
- Загальні вдосконалення в бібліотеці доступу до БД FireDAC, включаючи оновлені драйвера для FireBird, PostgreSQL і SQLite. Вибір статичного або динамічного підключення SQLite до застосунку.

Змінені стилі VCL для High DPI

В 10.4, архітектура стилізації VCL була суттєво розширена для підтримки High DPI і 4K моніторів. Тепер усі елементи UI на формі VCL автоматично

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		29

масштабується під відповідне до монітора дозвіл для показу форми. Був оновлений API стилізації для підтримки стилів high DPI.

Кожний графічний елемент UI може бути обраний з наборів різних масштабів і масштабований до потрібного DPI, що дає чітке зображення елементів UI на всіх моніторах.

Нові High DPI стилі й стилізація окремих VCL компонент

Обновлено велике число вбудованих і преміальних VCL стилів для підтримки нового режиму стилізації High-dpi. Це дозволяє вам створювати застосунку з відмінним дизайном для всіх моніторів.

Розроблювачі VCL застосунків тепер можуть використовувати трохи VCL стилів на різних формах в одному застосунку або в різних компонентах на одній формі. Це також включає стилізацію компонентів загальною темою для платформи. Крім застосункової гнучкості використання стилів, це дозволяє використовувати нестилізовані компоненти із зовнішніх бібліотек в VCL застосунках, що використовують стиль.

Поліпшена кроссплатформеність

- Додана підтримка Metal Driver GPU для macOS і iOS.
- Крім підтримки останнього iOS SDK, в RAD Studio 10.4 розроблювачі можуть задовольнити нові вимоги Apple до набору стартових екранів.
- Реалізований заново стилізуємий FMX компонент TMemo на платформі Windows значно поліпшений і тепер має відмінну підтримку IME.
- Користувачам редакцій Enterprise або Architect доступна повна інтеграція Fmxlinux з IDE для створення клієнтських застосунків Linux з GUI.
- Компонент Twebbrowser для iOS тепер реалізований на Wkwebview API.
- Реалізація компонента Media Player для macOS тепер використовує Avfoundation.

Оновлений менеджер пакетів Getit

Менеджер пакетів Getit в IDE був значно вдосконалений.

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		30

Дати випуску релізів пакетів тепер видні, і можливе сортування списку по цих датах; відбір тільки встановлених пакетів, контенту, доступного тільки при наявності підписки, багато чого іншого.

Універсальний інсталятор для установки Online і Offline

В 10.4 включений новий універсальний інсталятор, який використовує технологію на базі Getit. Цей інсталятор підтримує як online, так і offline (з ISO) варіанти установки.

Тепер обоє варіанта установки дозволяють вам указати початковий набір можливостей RAD Studio для установки, наприклад, свою комбінацію мов програмування й цільових платформ, мов інтерфейсу, і додавати до нього або видаляти непотрібне в будь-який момент.

2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випуск кваліфікаційну роботу за першим (бакалаврським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи кібербезпеки для стеження за мобільним пристроєм з використанням GPS.

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 кібербезпеки контролю роботи технологічного обладнання на виробництві в автоматизованому режимі. Розробити функціональну та структурну схеми системи;

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

г) організувати інтерфейс користувача з метою формування та виводу на

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		31

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 кібербезпеки в промислову експлуатацію та її подальшу успішну експлуатацію;

е) провести розрахунки по визначенню економічної ефективності розробленої системи;

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

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

КБПЗ_2024

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		32

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

Система, яка розробляється під час виконання бакалаврського проектування, призначена для стеження за мобільним пристроєм з використанням GPS. Крім цього система дозволяє будувати карти руху пристрою у заданий час. Захищати мобільний ПК при викрадені чи втраті.

Фізично, існує складність при прийманні сигналу викликани головним чином тим, що радіохвилі на своєму шляху переборюють іоносферу Землі, що являє собою плазмену хмару, утворену сонячним вітром. Її границі простираються від 70 до 1300 км над поверхнею Землі, і при проходженні через іоносферу радіосигнали послабляються й спотворюються. У нічний час, коли іоносфера перебуває в стані спокою, затримка передачі сигналу становить 1 м, а вдень, коли активність плазми висока, – більше 10 м. Для того щоб мінімізувати вплив іоносфери, використовують диференційований D-GPS.

У такій схемі використовуються два приймачі: один мобільний, а другий перебуває в точці з відомими координатами. Дані, що надходять із цих GPS, рівняються й обробляються, після чого відбувається коректування показань мобільного приймача. Чим ближче вони перебувають, тим точніше визначаються координати. Починаючи з 2005 р. супутники передають додаткові сигнали, які допоможуть виключити перешкоди від іоносфери.

По два сигнали додадуться до військових L1 і L2 і один – до цивільних L1, а існуючі нині сигнали не перетерплять яких-небудь змін. Наступний етап удосконалювання системи почався в 2008 р. Супутники будуть передавати ще один цивільний сигнал -5, що буде в 5 разів могутнішим, чим зараз. Здвоєний сигнал дозволить мінімізувати вплив іоносфери. GPS-приймачі майбутнього

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		33

зможуть порівнювати перекручування двох сигналів, вносячи необхідні корективи в розрахунки.

Оператори, що використовують D-GPS-приймачі, також виявляться у виграші. Нагадаємо, що точність роботи D-GPS-системи знижується в міру того, як збільшується відстань між фіксованим приймачем і мобільним GPS. Це пов'язане з тим що на приймачі попадають сигнали від супутників, що пройшли через різні шари іоносфери. При роботі із двома сигналами мобільний GPS здатний оцінити вплив іоносфери, а дані від фіксованого приймача допоможуть звести до мінімуму інші погрішності, які можуть становити від 30 до 50 див.

Щоб одержати точність позиціонування в межах сантиметрів або навіть міліметрів, користувачі можуть скористатися D-GPS-приймачами. Їхні сучасні моделі, маючи зв'язок зі стаціонарною станцією по радіоканалу, передають відомості про своє місцезнаходження й одержують відкоректовані дані. Довжина хвилі, на якій ведеться передача сигналу із супутника, становить 19 см. Приймач може виміряти час одержання сигналу з точністю до 1%. В абсолютному вираженні ця величина складе кілька міліметрів.

Для проведення більш точних вимірів приймач повинен ідентифікувати хвилю сигналу із супутника. Сучасні GPS зіставляють сигнали від супутників по каналах L1 і L2. У системі GPS довжини хвиль відрізняються на 85 см, що дозволяє проводити виміру з точністю до 8 мм. Надійність такої системи виміру в сотні разів більше, ніж у систем, що працюють із PRN-кодами. Їхня межа – 50 см. D-GPS приймачі, що працюють із одним каналом L1, забезпечують точність виміру до 19 см. Дорогі моделі GPS мають можливість підвищити точність виміру за допомогою зіставлення частот сигналів, що надходять по каналах L1 і L2. З початком передачі додаткових сигналів із супутників істотно зростає точність і надійність роботи GPS-приймачів.

Перший сигнал несе інформацію про місцезнаходження супутника й час передачі сигналу. Він приймається стаціонарними наземними станціями,

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		34

обробляється й відправляється на супутник, що передає його всім користувачам системи.

Другий сигнал це код, необхідний для визначення часу передачі сигналу. Творці системи називають його псевдовипадковим шумом. Щоб перебороти відстань в 20 тис. км, сигналу потрібен час. Якщо користувач зможе за допомогою свого приймача, у який закладений код, визначити час його відправлення, то нескладно буде зафіксувати час його проходження й, помноживши отримані дані на швидкість поширення, розрахувати відстань до супутника.

Встановлений час на мобільному ПК має першочергове значення, якщо в GPS-приймачі встановити годинники, то одержавши видалення від трьох супутників, користувач зможе визначити широту, довготу й висоту свого місцезнаходження.

Сигнал, що йде від супутників, нагадує три сфери, що перетинаються в різний час у різних точках. Для користувача, що перебуває на Землі, існує тільки один момент їхнього зіткнення в даний проміжок часу.

Для більше злагодженої синхронізації сигналу на супутниках установлені атомний годинник, що забезпечують точність ходу до однієї мільярдної.

У більшості GPS-приймачів вони можуть відставати на одну або більше секунд у день. Можна підрахувати, що помилка всього в одну секунду змінить відстань від супутника до користувача на 300 тис. км. Інженери називають процес виміру відстані між супутником і користувачем псевдовиміром. Справа в тому, що погрішність присутня й у сигналах від чотирьох супутників, у результаті чого ми одержуємо чотири рівняння із чотирма невідомими.

Сучасні GPS-приймачі здатні враховувати доплерівський ефект у випадку, якщо виміри проводяться в русі. При переміщенні приймача в бік поширення хвилі її довжина стає більше, а при зустрічному ході – менше. З огляду на даний ефект, можна одержати швидкість руху GPS-приймача.

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		35

Таким чином отримуються первинні дані з GPS модуляє. Такий метод виміру швидкості дуже точний. Таким чином, GPS-приймачі визначають три координати й три вектори швидкості, а також роблять синхронізацію часу через мережу. При цьому самі приймачі не передають сигналів в ефір.

Супутники GPS-системи передають сигнал, що володіє класичною синусоїдальною формою, на звичайній радіочастоті. Зараз на мікрохвильовій частоті передаються два сигнали їх назва L1, L2.

Канал L1 доступний для всіх. Вважається, що він призначений для цивільних користувачів, хоча й військові про нього не забувають.

Канал L2 призначений для військовослужбовців. Цивільні користувачі приймають на свої GPS-приймачі цей канал, але в силу того, що вони не мають доступу до PRN-коду, виникає помилка в позиціонуванні. Тільки дорогі приймачі дозволяють цивільним користувачам працювати в діапазоні L2.

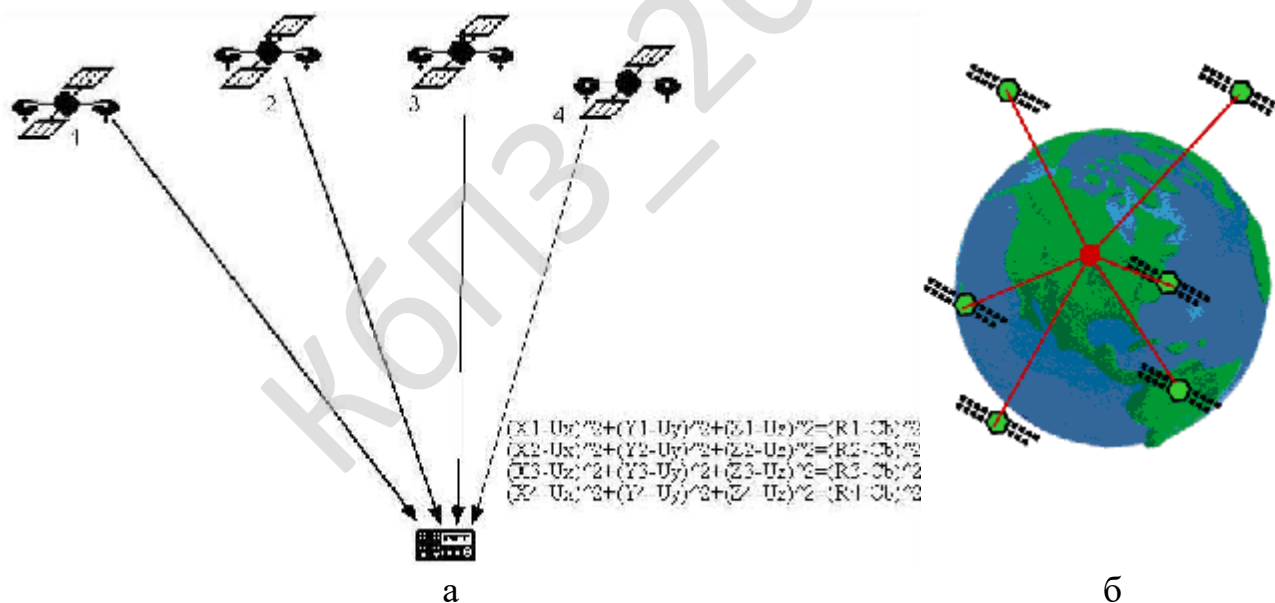


Рисунок 3.1 – Принцип дії GPS:

а – розрахунок положення відносно даних з супутників;

б – позиція відносно супутників

Система GPS складається з трьох основних компонентів: супутників, наземних станцій керування та приймачів GPS. Супутники, якими керує уряд США, обертаються навколо Землі на висоті приблизно 12 550 миль. На даний момент на орбіті знаходиться 31 робочий супутник GPS, для глобального покриття потрібно мінімум 24.

Наземні станції управління по всьому світу відповідають за моніторинг і контроль супутників. Вони гарантують, що супутники знаходяться на правильній орбіті та передають правильні сигнали.

GPS-приймач – це пристрій, який ви використовуєте для визначення свого місцезнаходження. Його можна знайти в різноманітних пристроях, включаючи смартфони, автомобілі та навіть човни та літаки. Приймач використовує сигнали супутників для обчислення відстані до кожного супутника. Вимірюючи відстань від кількох супутників, приймач може триангулювати своє положення та визначити своє точне місцезнаходження.

Приймач GPS використовує процес, який називається трилатерація, щоб визначити своє положення. Трилатерація – це метод визначення розташування точки шляхом вимірювання відстані до неї від двох або більше інших точок.

Щоб GPS-приймач міг визначити своє положення, він повинен мати можливість отримувати сигнали принаймні від чотирьох супутників. Приймач порівнює час передачі сигналу супутником із часом його прийому. Знаючи швидкість світла та різницю в часі, приймач може розрахувати відстань до супутника.

Одним із ключових факторів, що визначає точність GPS, є кількість супутників, з яких пристрій може приймати сигнали. Смартфони зазвичай мають кращий прийом GPS, ніж розумні годинники, оскільки вони мають більші антени та можуть приймати сигнали від більшої кількості супутників.

Ще один важливий фактор – потужність сигналу. Смартфони часто мають потужніші радіостанції та можуть отримувати сигнали від супутників, які розташовані далі, що підвищує їх точність. З іншого боку, розумні годинники

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		37

мають менші радіостанції та більш обмежені можливості прийому сигналів від віддалених супутників.

Смартфони мають інші технологічні переваги, як-от Wi-Fi та стільникові дані, які можуть підвищити точність визначення їхнього місцезнаходження. Розумні годинники зазвичай залежать лише від GPS.

Підсумовуючи, ключові причини, чому смартфони мають кращий прийом GPS, ніж розумні годинники:

- Смартфони мають більші антени і можуть приймати сигнали з більшої кількості супутників.
- Смартфони мають потужніші радіостанції та можуть приймати сигнали з великої відстані.
- Смартфони можуть використовувати інші технології, такі як Wi-Fi і стільникові дані, щоб підвищити точність свого місцезнаходження.

Хоча це правда, що розумні годинники стають все більш досконалішими, а їхні можливості GPS покращуються, малоімовірно, що вони коли-небудь зможуть зрівнятися з точністю смартфонів. Це просто питання фізичних обмежень пристроїв.

3.2 Розробка структурної схеми

Структурна схема розробленої системи зображена на рисунку 3.2. Для того, щоб система працювала необхідно з'єднатися зі супутниками. В залежності від поточного положення мобільного телефону з GPS модулем на небосхилі може бути від 3 до 13 супутників системи супутникового місце визначення. Вони передають своє поточне положення та через розрахунки видають положення користувача мобільного пристрою.

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		38

інформацію про місцезнаходження, щоб служити цілям програм. Хоча для деяких застосувань може бути достатньо приблизної оцінки, для інших потрібна більш точна інформація [25]. Наприклад, за допомогою програми Foursquare люди можуть повідомляти своє поточне місцезнаходження та свою історію місцезнаходжень у термінах «ділові приміщення», «ресторани», «бари» тощо. Деталізація необхідних даних досить низька, як і необхідна частота для оновлення інформації про локалізацію. Навпаки, інші додатки, наприклад для автомобільної навігації, вимагають високої точності та частого оновлення місцезнаходження, щоб забезпечити надійне визначення маршруту. У таких подіях, як спортивні змагання, необхідна дуже точна та точна інформація про позиціонування, яка також вимагає частого оновлення [7]. Для ілюстрації Добсон [5] надає неповний список із 18 різних способів збору інформації про місцезнаходження з різним рівнем точності, залежно від потреби та мети локалізації.

Три основні технології локації інтегровані як стандартні компоненти в сучасні смартфони: Cell ID, WLAN і GPS [13]. Також зауважте, що інші датчики можуть бути креативно використані для визначення місцезнаходження, виходячи за рамки їх основних цілей визначення – наприклад, використовуючи мікрофон смартфона для позиціонування в приміщенні [наприклад, 20].

Локалізація на основі Cell ID працює, доки є покриття мобільної мережі, до якої смартфон може підключитися. Положення пристрою виводиться з координат обслуговуючої базової станції [26]. Він точний лише в межах сотень метрів від відхилення. Інформацію про стільниковий ідентифікатор можна об'єднати з приблизною оцінкою часу проходження між пристроєм і базовою станцією (так зване «значення випередження часу»), з якого можна визначити діапазон між ними. За допомогою цієї комбінації можна досягти більш високої точності визначення положення [13].

Локалізація за допомогою WLAN заснована на вимірюванні близькості мобільного пристрою до точок бездротового доступу через інтенсивність отриманого сигналу (потужність отриманого сигналу; RSS). Шаблони RSS,

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		40

отримані в кількох точках доступу, порівнюються з таблицею заздалегідь визначених шаблонів RSS, зібраних у різних позиціях; розташування в порівнянні, яке найкраще підходить, приймається як положення пристрою [13, 26]. На основі цього механізму позиціонування на основі WLAN є точним у межах відхилення від 30 до 50 метрів і працює як у приміщенні, так і на вулиці [26]. Однак, оскільки ця технологія вимагає наявності бездротової точки доступу для реєстрації, вона обмежена густонаселеними районами [13, 26].

GPS – це супутникова навігаційна система, розроблена компанією Міністерство оборони США для військових цілей [6]. Це дозволяє визначати місцезнаходження мобільних пристроїв у будь-якому місці та в будь-який час на Землі за допомогою трилатерації з вимірюванням дальності між спостерігачем і кількома видимими супутниками [6]. Для обчислення фіксованого положення потрібні щонайменше чотири незалежні вимірювання (супутники) [10]. Хоча військова версія GPS здатна надавати більш точні дані позиціонування, її публічна версія обмежена точністю до 5-10 метрів [8, 13]. Такого рівня точності також можна досягти за допомогою недорогих GPS-приймачів, таких як інтегровані в смартфони, коли прив'язано до мінімальної кількості супутників [10]. Функціональність GPS, однак, обмежена позиціонуванням поза приміщенням, оскільки необхідна лінія прямої видимості супутників [13]. Незважаючи на те, що поширеність GPS-приймачів різко знизилась, розмір і вимоги до потужності [6], визначення GPS є досить ресурсомістким у порівнянні з іншими технологіями визначення місцезнаходження [3, 9].

Дослідження відстеження смартфонів

Точність вимірювань за допомогою GPS, вбудованої в смартфони, була предметом багатьох досліджень. Менард, Міллер, Новак і Норріс [14] проаналізували різницю в потужності позиціонування на основі GPS у смартфонах (з використанням трьох різних смартфонів: Samsung Galaxy, Motorola Droid і iPhone) порівняно з трекером автомобіля. Вони дійшли висновку, що ці три смартфони є прийнятною альтернативою іншим пристроям відстеження в

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		41

автомобілях. Вони показали, що кожен смартфон був точним в межах 10 метрів приблизно в 95 відсотків часу.

Відмінності в точності позиціонування між різними пристроями Apple (iPhone, iPod Touch і iPad) досліджували von Watzdorf і Michahelles [25]. Вони виявили значні відмінності в точності. Однак різні досліджувані пристрої також використовували різні технології визначення місцезнаходження (WLAN проти комбінації WLAN, GPS і Cell ID). Навпаки, Zandbergen [26] оцінив відмінності між позиціонуванням на основі GPS, WLAN і Cell ID, використовуючи лише один пристрій (iPhone 3G) і одну програму. Він виявив, що метод WLAN має потенціал для позиціонування в приміщенні, але на вулиці він відстає порівняно з локалізацією на основі GPS.

Використовуючи власно розроблену програму LBS, Hess, Farahani, Tschirschnitz і von Reischach [7] порівнювали різні смартфони з різними операційними системами з різними чіпсетамі GPS. Вони дійшли висновку, що точність вимірювань GPS сильно залежить від відповідного смартфона.

Kost і Brčić [11] проаналізували помилки позиціонування GPS на основі погодних умов на двох різних пристроях смартфонів для однієї фіксованої позиції, дійшовши висновку, що дослідницьке співтовариство має враховувати, що точність позиціонування залежить від різних факторів, таких як компоненти помилок, пов'язані з космічною погодою.

Мок, Ретшер і Вен [16] досліджували, чи може комбінація різних датчиків, вбудованих у смартфони, сприяти позиціонуванню GPS для підвищення рівня точності. Вони виявили, що за допомогою акселерометра та цифрового компаса можна підвищити точність позиціонування GPS.

Однак, наскільки нам відомо, різні програми відстеження, які працюють на одному пристрої (і мають однакові налаштування для кожної програми), ще не досліджувалися.

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		42

3.3 Розробка функціональної схеми

На рисунку 3.3 зображена функціональна схема системи. Нижче розглянемо її більш докладно. Складна криміногенна обстановка в нашій країні змушує уважніше стежити не тільки за своїми матеріальними цінностями й майном, але, насамперед, за безпекою дітей і людей похилого віку. Саме тому системи спостереження за людьми за допомогою GPS сьогодні знаходять таке широке застосування в нашому суспільстві.

У даному бакалаврському проекті реалізована система GPS спостереження за людьми – простий і ефективний інструмент спостереження за рухливими об'єктами в режимі реального часу.

Основні функціональні можливості системи спостереження за мобільними пристроями:

- Відстеження місця розташування людей на карті міста.
- Планування маршрутів мобільних співробітників під час дня.
- Призначення співробітникам завдання в режимі реального часу й контролювати їхнє виконання протягом дня.

Розроблена GPS система спостереження за людьми дозволяє відслідковувати маршрут спостережуваного об'єкта, оснащеного мобільним пристроєм з GPS модулем. Об'єкту спостереження можна призначати завдання, відзначаючи необхідні до відвідування точки на карті, а об'єкт за допомогою нашого сервісу побачить своє місце розташування на карті щодо потрібної точки.

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		43

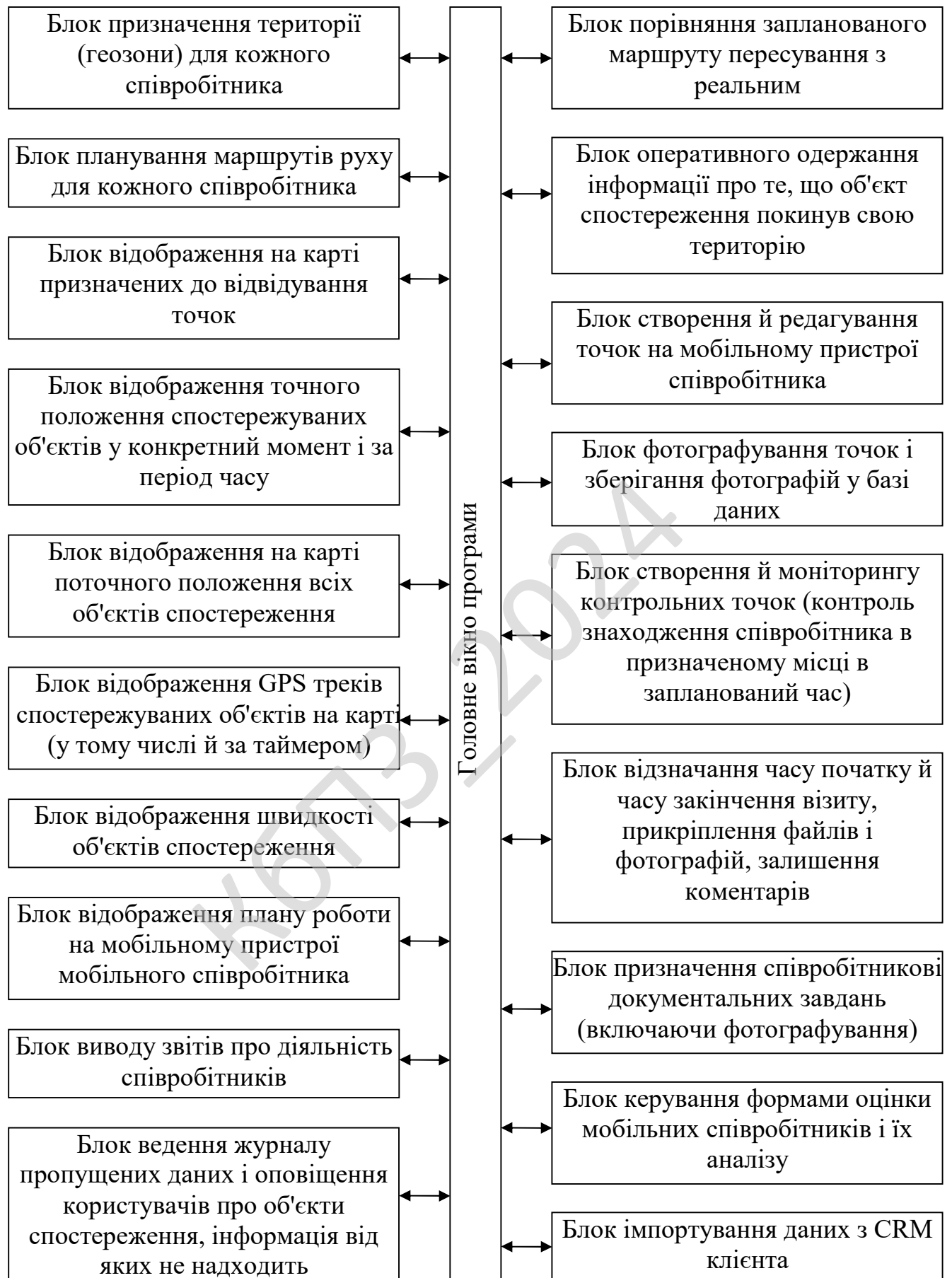


Рисунок 3.3 – Функціональна схема системи

Використання розробленого сервісу спостереження за людиною не вимагає використання яких-небудь спеціальних знань. Як стежить, так і об'єкт спостереження можуть користуватися системою, не володіючи якими-небудь спеціальними навичками – наша система діє за принципом «включив і працюй».

Функціональне призначення розробленого сервісу – керування мобільними співробітниками й контроль за їхньою діяльністю.

За допомогою сервісу можливо планувати роботу пересувних співробітників, давати їм завдання й контролювати виконання. До складу сервісу входять інструменти для зберігання клієнтської бази й планування роботи на будь-яку перспективу. Співробітник, оснащений мобільним пристроєм, не тільки бачить список призначених до відвідування точок і завдання для них, але так само має можливість побачити на карті своє місце розташування стосовно точок безпосередньо на мобільному пристрої. Робота із сервісом не вимагає спеціальних знань і навичок, збір GPS даних організований за принципом «включив і пішов».

- Блок призначення території (геозони) для кожного співробітника.
- Блок планування маршрутів руху для кожного співробітника.
- Блок відображення на карті призначених до відвідування точок.
- Блок відображення точного положення спостережуваних об'єктів у конкретний момент і за період часу.
- Блок відображення на карті поточного положення всіх об'єктів спостереження.
- Блок відображення GPS треків спостережуваних об'єктів на карті (у тому числі й за таймером).
- Блок відображення швидкості об'єктів спостереження.
- Блок відображення плану роботи на мобільному пристрої мобільного співробітника.
- Блок виводу звітів про діяльність співробітників.
- Блок порівняння запланованого маршруту пересування з реальним.

– Блок оперативного одержання інформації про те, що об'єкт спостереження покинув свою територію.

– Блок створення й редагування точок на мобільному пристрої співробітника.

– Блок фотографування точок і зберігання фотографій у базі даних.

– Блок створення й моніторингу контрольних точок (контроль знаходження співробітника в призначеному місці в запланований час).

– Блок відзначання часу початку й часу закінчення візиту, прикріплення файлів і фотографій, залишення коментарів.

– Блок призначення співробітникові документальних завдань (включаючи фотографування).

– Блок керування формами оцінки мобільних співробітників і аналізу їх.

– Блок імпортування даних з CRM клієнта.

– Блок ведення журналу пропущених даних і оповіщення користувачів про об'єкти спостереження, інформація від яких не надходить.

Візити й відвідування

Будучи відбиті на карті маршруту, а так само у звітах, точки діляться на три категорії:

– Заплановані, але не відвідані.

– Заплановані відвідані.

– Відвідані позапланово.

Відвідування точки мобільним співробітником може бути зафіксоване автоматично (по даним GPS) – власно відвідування – і вручну, коли співробітник фіксує початок і закінчення візиту вручну засобами Універсального інструмента мобільного співробітника – візит.

На відміну від автоматично розрахованого відвідування візит гарантує присутність співробітника в точці, тому що для створення візиту потрібно вручну зафіксувати час початку й закінчення візиту, при цьому система не дозволить зафіксувати візит, якщо співробітник фізично перебуває на видаленні від точки,

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		46

порівнюючи поточне географічне положення співробітника зі збереженим географічним положенням точки.

Крім того, у рамках візиту співробітник має можливість зберігати довільні файли, фотографії й коментарі.

У звітах при наявності візиту в точку час початку й закінчення відвідування відображається за результатами візиту, при цьому фіксується, що це саме візит.

Планування документальних завдань

Поряд з можливістю текстових завдань у точці є можливість призначати об'єкту спостереження завдання у вигляді шаблонів документів. Алгоритм роботи наступний:

- Створюється набір шаблонів (типів) документів.
- Шаблон або шаблони додаються в план для точки на обрану дату.
- За допомогою універсального інструмента об'єкт спостереження наповнює шаблон даними й зберігає документ.

Шаблон

Шаблон (тип) документа являє собою заголовок (назва) документа й наповнення (табличну частину). Рядка табличної частини можуть містити наступні стандартні типи даних:

- Ціле число.
- Десятковий дріб із двома знаками після коми.
- Логічний тип (ТАК/НЕСАЄ).
- Дата.
- Рядок.
- Фотографія.

Форми оцінки мобільних співробітників

Форми оцінки мобільних співробітників – це інструмент виміру й аналізу професійного розвитку мобільного співробітника.

Форми оцінки мобільних співробітників дозволяють:

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		47

– Планомірно оцінювати рівень розвитку навичок для кожного мобільного співробітника й команди в цілому.

– Визначати навички, що вимагають розвитку для кожного мобільного співробітника й всієї команди.

– Забезпечувати безперервність і послідовність процесу розвитку навичок мобільного співробітника.

– Підвищувати якість роботи команди й кожного мобільного співробітника, ґрунтуючись на розвитку конкретних навичок.

Форми оцінки мобільних співробітників являють собою електронні бланки документів.

Відсутність даних від трекерів

У деяких випадках від об'єктів спостереження можуть не приходити дані.

Це може відбуватися по наступних причинах:

– виключено пристрій (смартфон, планшет, GPS-трекер);

– відсутній зв'язок GSM;

– відсутній зв'язок із супутниками GPS або неможливо визначити місце розташування по мережах GSM/WiFi;

– на пристрої виключений GPS або неможливо визначити місце розташування по мережах GSM/WiFi;

– на пристрої виключені програмні трекери.

Переглянути список об'єктів спостереження, від яких не надходили дані й відповідні записи логів можна за допомогою журналу.

Оповіщення по електронній пошті

Кожні 60 минут протягом доби сервіс опитує базу даних про "загублені" об'єкти спостереження. Отримана інформація може бути вислана по електронній пошті. Необхідною умовою для цього є наявність адреси електронної пошти в картці користувача, а так само налаштування аккаунта на відправлення повідомлень.

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		48

Робота із сервісом з мобільних пристроїв можлива тільки в режимі моніторингу.

Робота із сервісом здійснюється від імені аккаунта – організації або фізичної особи – назва якого не обов'язково повинне збігатися з назвою організації або ім'ям людини. Оскільки ідентифікація в системі виробляється по аккаунтам, не допускається створення декількох аккаунтів з однаковою назвою – всі аккаунти повинні бути унікальними.

У рамках одного аккаунта може бути створене як завгодно багато користувачів (логінів) – операторів, що здійснюють контроль за об'єктами спостереження, що створюють базу точок, що формують плани й здійснюють сервісні функції. Користувачі аккаунта розділені на дві групи:

- адміністратори аккаунта (можуть створювати інших користувачів і перепризначувати їхнього права, здійснювати оплату, змінювати тип аккаунта, створювати об'єкти спостереження, а так само використовувати всі функції, доступні звичайним користувачам;

- користувачі зі звичайним набором прав – можуть створювати точки, заводити плани й давати завдання для об'єктів спостереження, здійснювати контроль за об'єктами спостереження й одержувати звіти.

Інші учасники сервісу – об'єкти спостереження – мобільні співробітники або автомобілі, за яких ведеться спостереження і яким даються завдання для виконання. Об'єкти спостереження перебувають «у полях» і оснащені мобільними пристроями з GPS приймачами або спеціалізованими GPS трекерами.

Тарифікація аккаунтів ведеться по кількості об'єктів спостереження, закріплених за даним аккаунтом.

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

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		49

Далі через запит даних з мобільного пристрою проходить отримання GPS даних та обробка розрахункових даних та виведення розрахункових даних.

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.

КБПЗ_2024

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		51

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

4.1 Блок-схеми та опис алгоритмів функціонування системи

На рисунку 4.1 наведено блок-схема клієнтського ПЗ. Її робота складається з виконання наступних кроків:

- Ініціалізація ресурсів ПЗ.
- Ініціалізація GPS модулю.
- Підключення файлу налаштування ПЗ.
- Підключення додаткових модулів ПЗ.
- Сканування ресурсів ПЗ на відповідність версій.
- Запит – перевірка пройдена?
- Виведення інтерфейсу ПЗ.
- Очікування запиту користувача.
- Запит користувача?
- Обробка дії користувача.
- Запит налаштування GPS модуля?
- Виведення вікна налаштування GPS.
- Отримання даних та налаштування параметрів доступу до GPS.
- Запит отримання геопараметричних даних?
- Підпрограма отримання даних геопозиціонування.
- Виведення системної інформації модуля.
- Запит – передача даних до ПК?
- Підпрограма передачі даних до ПК.
- Прийом даних з ПК?
- Підпрограма прийому даних з ПК.

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		52

- Запит WM_CLOSE?
- Запит OnError?
- Виведення повідомлення тіла помилки.
- Звільнення виділених ресурсів ПЗ.

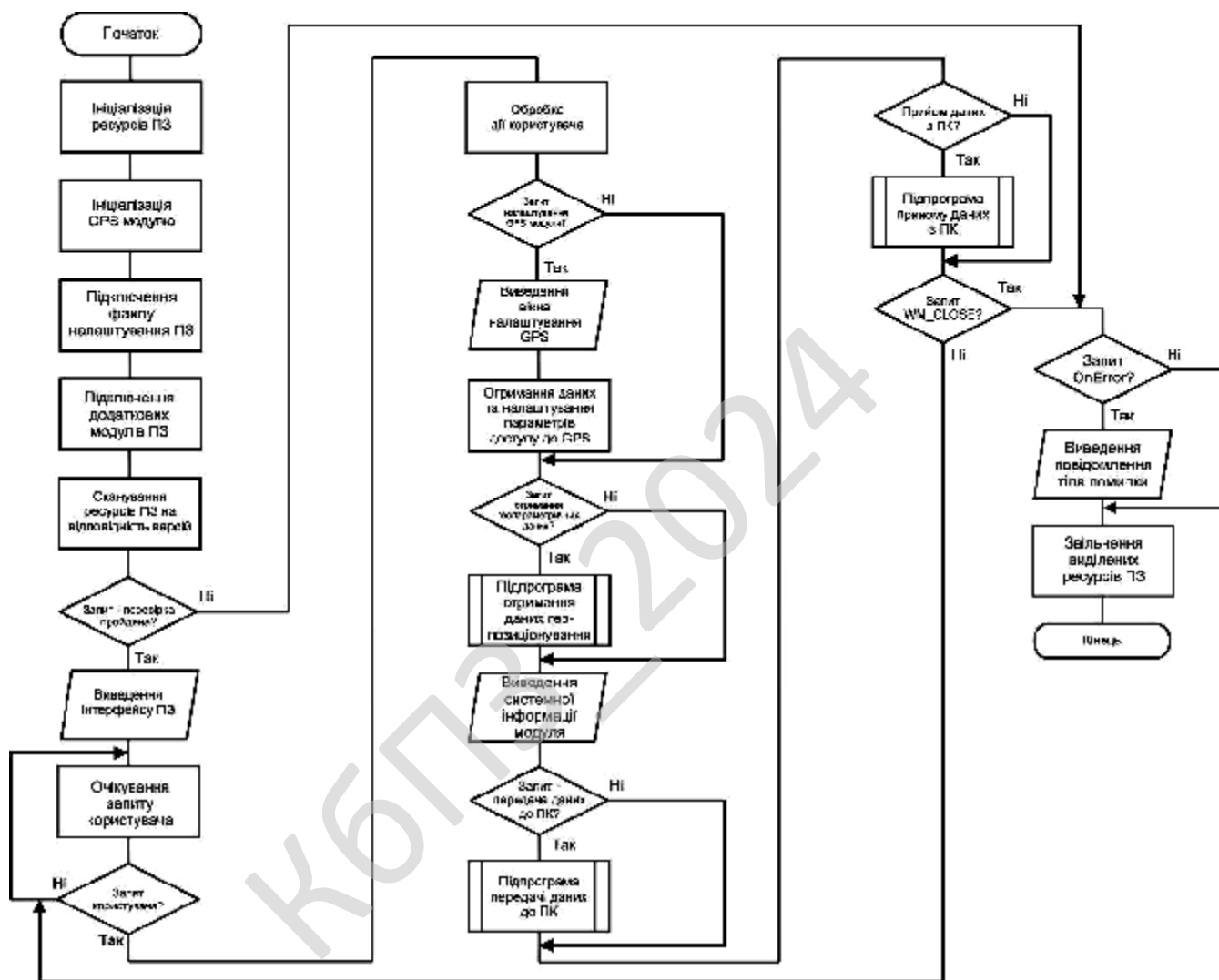


Рисунок 4.1 – Блок-схема клієнтського ПЗ

На рисунку 4.2 наведено блок-схема серверного ПЗ. Її робота складається з виконання наступних кроків:

- Ініціалізація ресурсів ПЗ та API Google maps.
- Підключення модулів взаємодії з ОС WM.
- Підключення модуля GPS.

- Перевірка цілісності ПЗ.
- Запит – перевірка пройдена?
- Очікування запиту початку роботи ПЗ.
- Запит отримано?
- Перехід до режиму моніторингу.
- Підпрограма відправлення запиту GPS даних.
- Запит є вхідні дані?
- Підпрограма отримання GPS даних.
- Продовжити обробку запитів?
- Звільнення ресурсів ПЗ.

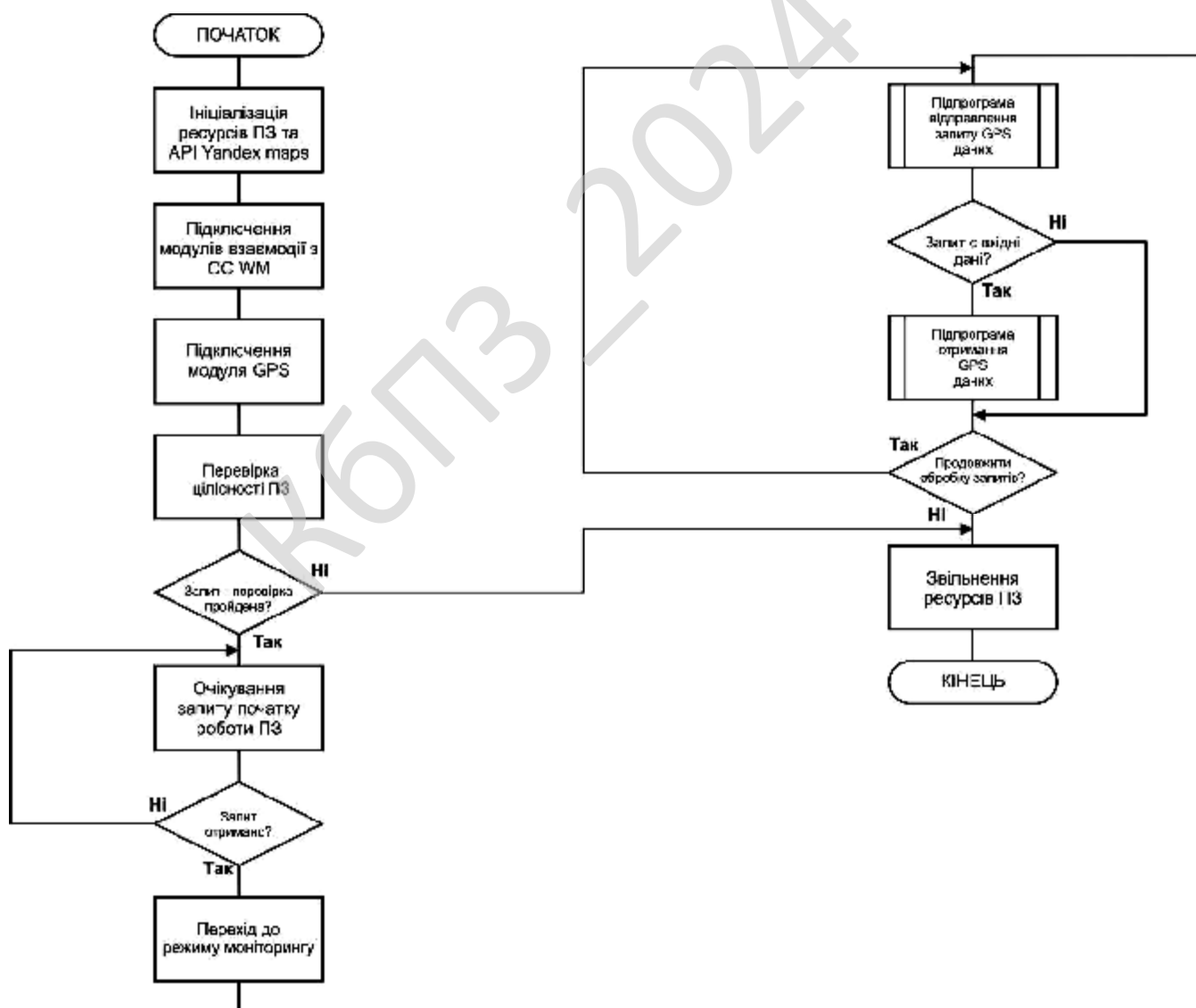


Рисунок 4.2 – Блок-схема серверного ПЗ

Розглянемо реалізацію алгоритмів та методів бакалаврського проекту. При написанні ПЗ використовувалось JavaScript API для взаємодії з Google Maps. Служба «API Google Maps» має програмний інтерфейс (API) на Javascript, який дозволяє інтегрувати інтерактивні карти у веб-сторінки.

Об'єкт Ymaps.Geocoder підтримує зворотне геокодування, за допомогою якого можна перетворювати координати об'єкта в поштову адресу.

Для зворотного геокодування у параметрі Request потрібно передати замість адреси геоточки (Ymaps.Geopoint).

```
// Запускає процес геокодування
var geocoder = new Ymaps.Geocoder(new Ymaps.Geopoint(37.588395, 55.762718),
{results: 1});
// Оброблювач успішного завершення процесу геокодування
Ymaps.Events.observe(geocoder, geocoder.Events.Load, function () {
    if (this.length()) {
        map.addoverlay(this.get(0));
        map.panto(this.get(0).getgeopoint())
    }else {
        alert("не знайдене")
    }
});
// Оброблювач невдалого завершення геокодування
Ymaps.Events.observe(geocoder, geocoder.Events.Fault,
function (geocoder, error) {
    alert("Відбулася помилка: " + error.message)
});
```

Результати пошуку, як і у випадку із прямим геокодуванням, вертаються у вигляді групи міток класу Ymaps.GeoCoderResult, які відразу можна нанести на карту.

Об'єкт Ymaps.YMapSml дозволяє відобразити YMapSml документ на карті, використовуючи Javascript.

Щоб відобразити документ необхідно передати на вхід конструктора Ymaps.YMapSml URL документ, а потім додати створений об'єкт оверлей на карту.

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		55

Що завантажується YMapSml повинен бути доступний привселюдно по протоколу http.

```
var ml = new Ymaps.YMapSml("http://api.google.ru/maps/YMapSml/examples  
/xml/demonstration.xml");  
map.addoverlay(ml);
```

Тому що завантаження YMapSml документа – асинхронний процес, закінчення завантаження необхідно очікувати за допомогою події Load.

Щоб одержати можливість переглядати помилки при завантаженні YMapSml документа, необхідно прикріпити оброблювач до події Fault.

```
Ymaps.Events.observe(ml, ml.Events.Fault, function (ml, error)  
{  
alert("Відбулася помилка: " + error);  
});
```

Тому що створений об'єкт оверлей належить класу Maps.Geoobjectcollection, з ним можна звертатися як із групою. Для пошуку об'єкта по якому-небудь параметру використовується метод групи filter().

```
var ml = new  
Ymaps.YMapSml("http://api.google.ru/maps/YMapSml/examples/xml/mapparams.xml");  
map.addoverlay(ml);  
Ymaps.Events.observe(ml, ml.Events.Load, function () {  
var obj = this.filter(function (obj) {  
return obj.name == 'Ім'я об'єкта'  
});  
obj[0].openballoon();  
});
```

API ключ – унікальна цифро-буквена комбінація, по якій сервіс «API Google Maps» ідентифікує ПЗ, що звертається до API.

Щоб одержати API ключ, слід зареєструватися на Яндексі, у спеціальній формі одержання ключа ознайомитися й прийняти умови користувацької угоди, а також указати програму, на яким буде використовуватися API.

Ключ можна закріпити за будь-якою програмою. Для однієї програми може бути отримано кілька ключів, причому не обов'язково, щоб усі ключі були зареєстровані на один обліковий запис.

За допомогою Javascript API можливо:

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		56

Взаємодія з картою здійснюється за допомогою елементів керування, які дозволяють користувачеві змінювати більшість параметрів карти.

Об'єкти-оверлеї (мітки) додаються на карту поверх усіх шарів. Об'єкти прикріплюються до певного місця на карті й зберігають цю позицію при зміні масштабу й переміщенні карти.

Для навігації й для вказівки місця розташування об'єктів на карті використовується відповідна система координат.

Для географічних карт – це геодезична система координат WGS-84, де для вказівки положення об'єкта на карту досить указати його довготу й широту.

Для навігації по кресленнях і планам місцевості також може використовуватися декартова система координат.

Можливості карти також можуть бути розширені за допомогою сервісів геокодування, прокладки маршруту й візуалізації YMapSml.

YMapSml (Google Maps Markup Language) – XML мова, призначена для опису географічних даних на картах Яндекс.

За допомогою YMapSml можливо в текстовому виді задати ділянку карти з нанесеними мітками й іншими об'єктами, а потім відобразити його в браузері засобами JS API Google Maps.

Сервіс геокодування служби API Google Maps використовує YMapSml для опису знайдених по запиті географічних об'єктів.

Для візуалізації YMapSml документів засобами JS API Google Maps потрібно вказувати Арі ключ.

YMapSml використовується як формат передачі даних між сервісами Google Maps і сторонніми програмними засобами.

Дані у форматі YMapSml можуть бути відображені на Google Maps, у цьому ж форматі повертає інформацію про місце розташування об'єктів сервіс геокодування.

Розглянемо як було реалізоване створення карти та пошук КПК у магістерському ПЗ. Щоб додати карту на сторінку, створювався оброблювач події

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		58

закінчення завантаження сторінки onload якій розмішався у тегу head після скрипта завантаження API.

```
<script type="text/javascript">
// Створює оброблювач події window.onload
    Ymaps.jquery(function () {
// Створює екземпляр карти й прив'язує його до
// створеного контейнера
        var map = new
            Ymaps.Мap(Ymaps.jquery("#ymapsid") [0]);
// Установлює початкові параметри відображення
// карти: центр карти
// і коефіцієнт масштабування
        map.setcenter(new Ymaps.Geopoint(37.64, 55.76), 10);
    }
</script>
```

Клас Ymaps.Geocoder дозволяє відправляти запити геокодеру, одержувати інформацію про статус і результат процесу геокодування, а також одержувати результати геокодування у вигляді міток.

Конструктор класу приймає два параметри.

1. request – рядок з адресою, яку потрібно геокодувати (для зворотного геокодування – точка з географічними координатами);

2. options – параметри:

– boundedby – область на карті, у якій здійснюється пошук об'єкта;

– strictbounds – прапор, що вказує, що шукати впливає тільки усередині області, заданою опцією boundedby;

– results – необхідна кількість результатів пошуку (не більш);

– skip – вказівка пропустити перші n результатів у відповіді;

– preflang – мова відповіді.

Щоб визначити координати об'єкта, створювався екземпляр класу Ymaps.Geocoder і передавався йому в якості параметра рядок з адресою об'єкта. У прикладі нижче проводиться пошук усіх об'єктів з назвою "Кіровоград":

```
var geocoder = new Ymaps.Geocoder("Кіровоград");
map.addoverlay(geocoder);
```

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		59

У результаті виконання прикладу, змінна `geocoder` буде містити результати пошуку у вигляді групи спеціальних міток класу `Ymaps.Geocoderresult`. Ці мітки відрізняються від звичайних міток класу `Ymaps.Placemark` наявністю чотирьох додаткових полів, у яких передаються додаткові відомості про знайдений об'єкт:

- `kind` – вид топоніма (вулиця, будинок, місто і т.д.);
- `precision` – передбачувана точність відповіді;
- `text` – адреса одним рядком;
- `Addressdetails` – структуроване подання адреси об'єкта у відповідності зі стандартом `hxl`.

Клас `Ymaps.Geocoderresult` реалізує інтерфейс `Ymaps.Overlay`, тому мітки `Ymaps.Geocoderresult` можна додавати на карту точно так само, як і звичайні.

Для обробки результатів геокодування необхідно враховувати, що геокодеру потрібен час на обробку запиту й відповідь. Тому рекомендується завжди використовувати оброблювачі подій для одержання інформації про закінчення геокодування й про його поточний статус.

У класі `Ymaps.Geocoder` передбачено дві події: `Load` (геокодування пройшло без помилок) і `Fault` (з помилками).

В оброблювачі події `Load` можна оперувати результатами геокодування як групою.

Груповання об'єктів. Приклад нижче ілюструє обробку подій геокодера реалізованого у магістерській роботі. У випадку успішного геокодування користувачеві карти буде показане повідомлення про кількість знайдених об'єктів, а самий релевантний (перший) з них буде показаний на карті.

```
Ymaps.Events.observe(geocoder, geocoder.Events.Load, function () {
    if (this.length()) {
        alert("Знайдене:" + this.length());
        map.addoverlay(this.get(0));
        map.panto(this.get(0).getgeopoint())
    }else {
        alert("Нічого не знайдено")
    }
});
```

```
Ymaps.Events.observe(geocoder, geocoder.Events.Fault,  
function (error) { alert("Відбулася помилка: " + error.message) });
```

Параметр `preflang` геокодера дає можливість контролювати мову результатів пошуку.

Цей параметр дозволяє задати мову, на якій буде проводитися видача результатів геокодування. Код мови вказується у відповідності зі стандартом ISO 639.

Список можливих значень:

- ru Росія;
- uk Україна.

Якщо встановити параметр `preflang` у значення “uk”, то в результаті пошуку адреси «Кіровоград, вул. Шеченко» буде отриманий результат – «Кіровоград вулиця Шеченко».

```
var geocoder = new Ymaps.Geocoder("Кіровоград",  
{ preflang: "uk" } );  
Ymaps.Events.observe(geocoder, geocoder.Events.Load, function (geocoder) {  
    map.addoverlay(geocoder.get(0));  
    map.setbounds(geocoder.get(0).getbounds());  
});
```

4.2 Захист розробленого програмного забезпечення

Захист розробленого програмного забезпечення буде відбуватися за допомогою алгоритму FEAL – блоковий шифр, запропонований Акіхіро Симідзу і Седзі Міягуті.

У ньому використовуються 64-бітовий блок і 64-бітовий ключ. Його ідея полягає і в тому, щоб створити алгоритм, подібний DES, але з більш сильною функцією етапу. Використовуючи менше етапів, цей алгоритм міг би працювати швидше. На жаль, дійсність виявилася далекою від цілей проекту.

Як вхід процесу шифрування використовується 64-бітовий блок відкритого тексту. Спочатку блок даних підлягає операції XOR з 64 бітами ключа. Потім блок даних розщеплюється на ліву і праву половини. Об'єднання

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		61

лівої і правої половин за допомогою XOR утворює нову праву половину. Ліва половина і нова права половина проходять через N етапів (спочатку 4). На кожному етапі половина об'єднується за допомогою функції F[1] з 16 бітами ключа і за допомогою XOR – з лівою половиною, створюючи нову праву половину. Вихідна права половина (на початок етапу) стає новою лівою половиною. Після N етапів (ліва і права половини не переставляти після N-го етапу) ліва половина знову об'єднується з допомогою XOR з правою половиною, утворюючи нову праву половину, потім ліва і права об'єднуються разом в 64-бітове ціле. Блок даних об'єднується за допомогою XOR з іншими 64 бітами ключа і алгоритм завершується.

КБПЗ – 2024

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		62

На рисунку 5.2 зображено форму авторського права. Було обрано Shareware умову розповсюдження.

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

В угоді про використання (ліцензії для кінцевого користувача, EULA) також може бути обумовлена заборона на комерційне або професійне (не тестове) її використання. Основний принцип умовно-безплатного ПЗ – «спробуй, перш ніж купити» (try before you buy). ПЗ що поширюється як умовно-безплатний, надається користувачам безоплатно.

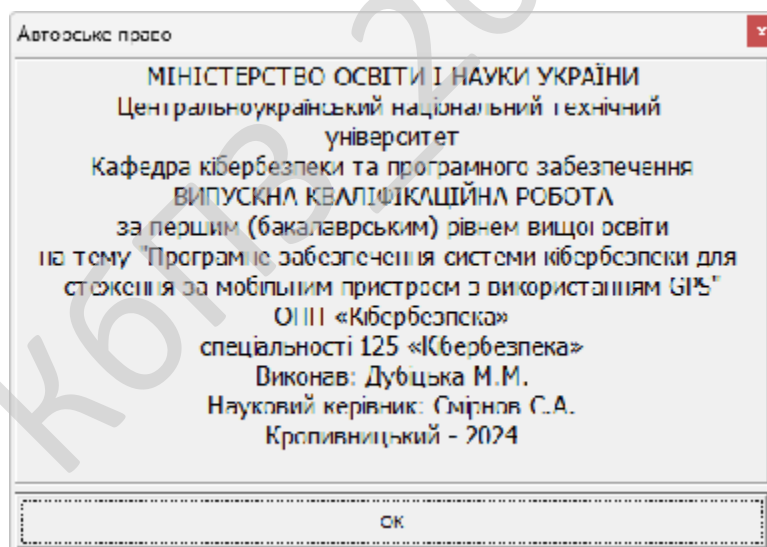


Рисунок 5.2 – Довідка

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		64

6 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти, призначено для системи кібербезпеки для стеження за мобільним пристроєм з використанням GPS.

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

Рішення завдання полягало у вирішенні наступних задач:

– Був проведений огляд існуючих систем для стеження за мобільним пристроєм з використанням GPS.

– Досліджена система для стеження за мобільним пристроєм з використанням GPS.

– На основі отриманих результатів досліджень створена програмна реалізація системи кібербезпеки для стеження за мобільним пристроєм з використанням GPS.

Розроблені під час виконання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання для стеження за мобільним пристроєм з використанням GPS.

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 Delphi 10.4. Дана мова програмування дозволяє найбільш ефективно обробляти дані призначені для системи кібербезпеки для стеження за мобільним пристроєм з використанням

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		65

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

Програма призначена для виконання під управлінням багатозадачної операційної системи Android.

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

Для підвищення рівня безпеки запропоновано застосовувати алгоритм FEAL.

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.

КБПЗ-2024

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		66

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Loren Kohnfelder. Designing Secure Software. No Starch Press. 2022. 332 p.
2. Samir Kumar Rakshit. Ethical Hacker's Penetration Testing Guide. BPB Online. 2022. 509 p.
3. Corey J. Ball. Hacking APIs. No Starch Press. 2022. 353 p.
4. Kevin Beaver. Hacking for Dummies. John Wiley & Sons. 2022. 419 p.
5. Mark S. Merkow. Practical Security for Agile and DevOps. CRC Press. 2022. 236 p.
6. Derek Fisher. Application Security Program Handbook. Manning Publications. 2021. 155 p.
7. Cameron Wyatt PH.D. Kali Linux Tutorial. Independently published. 2021. 60 p.
8. Alex Matrosov, Eugene Rodionov, Sergey Bratus. Rootkits and Bootkits. No Starch Press. 2019. 450 p.
9. Kuznetsov, O., Kryvinska, N., Ilchenko, O., Smirnova, T., Ulianovska, Y. «Comparative Analysis of Cryptocurrency Trading Platforms Using the Analytic Hierarchy Process». *CEUR Workshop Proceedings*, 2023, 3628, pp. 106-115.
10. Smirnov, O., Sydorenko, V., Aleksander, M., Zhyharevych, O., Yanchev, S. «Simulation of the cloud IoT-based monitoring system for critical infrastructures». *CEUR Workshop Proceedings*, Volume 3530, 2023, pp. 256-265.
11. Kuznetsov, O., Kandiy, S., Frontoni, E., Smirnov, O. «Trade-offs in Post-Quantum Cryptography: A Comparative Assessment of BIKE, HQC, and Classic McEliece». *CEUR Workshop Proceedings*, Volume 3504, 2023, pp. 1-11.
12. Smirnov, O., Neskorodieva, T., Fedorov, E., Rudakov, K., Neskorodieva, A. «Method Detection Audit Data Anomalies on Basis Restricted Cauchy Machine» *CEUR Workshop Proceedings*, Volume 3187, 2022,

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		67

13. Smirnov, O., Lakhno, V., Akhmetov, B., Chubaievskiy, V., Khorolska, K., Bebesko, B. «Selection of a Rational Composition of Information Protection Means Using a Genetic Algorithm». In: Rajakumar, G., Du, KL., Vuppalapati, C., Beligiannis, G.N. (eds) *Intelligent Communication Technologies and Virtual Mobile Networks. Lecture Notes on Data Engineering and Communications Technologies*, vol 131. 2023. Springer, Singapore. pp. 21-34.

14. Smirnov O.A., Al-Oraiqat A.M., Ulichev O.S., Meleshko Ye.V., Al-Rawashdeh H.S., Polishchuk L.I. «Modeling strategies for information influence dissemination in social networks». *Journal of Ambient Intelligence and Humanized Computing* Volume 13, Issue 5. Springer, Cham. 2022, pp. 2463-2477.

15. 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

16. 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.

17. Smirnov O., Kuznetsov A., Girzheva O., Kiian A., Nakisko O., Kuznetsova T. «Advanced Code-Based Electronic Digital Signature Scheme». *2020 IEEE International Conference on Problems of Infocommunications Science and Technology, PIC S and T 2020*, Kharkiv, 6 October 2020-9 October 2020, P. 358-362.

18. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova K. «Data hiding scheme based on spread sequence addressing». *CEUR Workshop Proceedings* Volume 2805, 2020, Pages 44-58.

19. 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.

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

21. 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.

22. Smirnov O., Kuznetsov A., Arischenko A., Chepurko I., Onikiychuk A., Kuznetsova T. «Pseudorandom sequences for spread spectrum image steganography». *CEUR Workshop Proceedings* Volume 2654, 2020, Pages 122-131.

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

24. Smirnov O., Lutsenko M., Kuznetsov A., Kiian A., Kuznetsova T., «Biometric cryptosystems: overview, state-of-the-art and perspective directions». *Lecture Notes in Networks and Systems*, vol 152. Springer, Cham. 2021, pp 66-84.

25. 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.

26. Smirnov O., Kuznetsov A., Kiian A., Babenko V., Perevozova I., Chepurko I. «New Approach to the Implementation of Post-Quantum Digital Signature Scheme». *2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT)*, Ukraine, Kyiv, May 14-18. 2020. P. 166-171.

27. 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.

28. Smirnov O., Kuznetsov A., Pushkar'ov A., Serhienko 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.

29. Smirnov, O., Markovets, O. Vovk, N., Turchyn, Y., «Model of informational support for social network administrators' content creation». *CEUR Workshop Proceedings Volume 2616*, 2020, Pages 125-136.

30. Smirnov, O., Shekhanin, K., Kuznetsov, A., Krasnobayev, V. «Detecting Hidden Information in FAT». *International Journal of Computer Network and Information Security (IJCNIS)*. Vol. 12, No. 3, 2020. PP.33-43.

31. 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.

32. Smirnov, O., Kuznetsov, A., Kolovanova, I., Kuznetsova, T., «Noise immunity of the algebraic geometric codes». *International Journal of Computing*; 2019, Volume 18, Issue 4 – Research Institute for Intelligent Computer Systems – 2019. – P. 393-407.

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

34. Smirnov, O., Ulichev, O., Meleshko, Y., Khokh, V., Goncharenko, I. «Method of Choosing Objects for Informational Influence in Social Networks during

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		70

Information Campaign Based on the Analytic Hierarchy Process». *CEUR Workshop Proceedings*, Vol 2588, P. 215-227, 2019.

35. 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.

36. Smirnov, O., Kuznetsov, A., Kiian, A., Gorbenko, Y., Cherep, O., Bexhter L. «Code-based Pseudorandom Generator for the Post-Quantum Period», *2019 IEEE International Conference on Advanced Trends in Information Theory (IEEE ATIT 2019)*. 18.12.19-20.12.19 Kyiv Ukraine. P. 204 – 209.

37. Smirnov, O., Kuznetsov, A., Nariezhnii, O., Stelnyk, S., Kokhanovska, T., Kuznetsova T., «Side Channel Attack on a Quantum Random Number Generator», *10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2019*; Metz; France; 18 - 21 September 2019. P.713-718.

38. Kuznetsova, T., «Code-Based Schemes for Post-Quantum Digital Signatures», *10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2019*; Metz; France; 18-21 September 2019. P. 707-712.

39. Smirnov, O., Kuznetsov, A., Stefanovych, O., Gorbenko, Y., Krasnobaev, V., Kuznetsova K. «Information Hiding Using 3D-Printing Technology», *10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2019*; Metz; France; 18-21 September 2019. P.701-706.

40. Smirnov, O., Hu, Z., Vasiliu, Y., Sydorenko, V., Polishchuk, Y., «Abstract Model of Eavesdropper and Overview on Attacks in Quantum Cryptography Systems», *10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2019*; Metz; France; 18-21 September 2019. P.399-405.

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		71

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., Kiian, A., Babenko, B., Zhosan, H., Prokopovych-Tkachenko, D., «Soft Decoding Method for Turbo-Productive Codes», *2019 3rd International Conference on Advanced Information and Communications Technologies, AICT 2019, Lviv, Ukraine, 2-6 July, 2019, P. 129-134.*

43. Smirnov, O., Kuznetsov, A., Kiian, A., Zamula, A., Rudenko, S., Hryhorenko, V., «Variance Analysis of Networks Traffic for Intrusion Detection in Smart Grids», *2019 IEEE 6th International Conference On Energy Smart Systems (2019 IEEE ESS), Kyiv, Ukraine April 17-19, 2019 P. 353-358.*

44. 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.*

45. 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.*

46. Smirnov, O., Kuznetsov, A., Kiian, A., Kuznetsova, K., Ivko, T., Prokopovych-Tkachenko, D., «Soft Decoding Based on Ordered Subsets of Verification Equations of Turbo-Productive Codes», *CEUR Workshop Proceedings Volume 2353, CEUR Workshop Proceedings 2019, Pages 873-884.*

47. 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.*

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		72

48. 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.

49. Смірнов О.А. Козлов Я.О., Смірнова Т.В. «Дослідження застосування SIEM-систем для забезпечення кібербезпеки та захисту інформації». *II Міжнародна науково-практична Інтернет-конференція «Інновації та перспективні шляхи розвитку інформаційних технологій (ПШПІТ-2023)»* м.Черкаси 6 грудня 2023 року – Черкаси: ЧДТУ.– 2023. – С.251-252.

50. Козлов Я.О., Смірнова Т.В., Смірнов О.А. «Дослідження SIEM-систем для забезпечення кібербезпеки». *VII міжнародна науково-практична конференція “Інформаційна безпека та комп’ютерні технології” до 30-ти річчя кафедри кібербезпеки та програмного забезпечення*, м. Кропивницький. 1 листопада 2023 р. – Кропивницький: ЦНТУ. – 2023. – С. 26.

51. Козлов Я.О., Козірова Н.Л., Смірнов О.А. «Дослідження структури та принципу роботи SIEM-системи». *VII міжнародна науково-практична конференція “Інформаційна безпека та комп’ютерні технології” до 30-ти річчя кафедри кібербезпеки та програмного забезпечення*, м. Кропивницький. 1 листопада 2023 р. – Кропивницький: ЦНТУ. – 2023. – С. 59.

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

					ВКРБ-125.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		73

Додаток А
(обов'язковий)

Технічне завдання

Зміст

1 Найменування та область застосування.....	2
2 Підстава для розробки.....	2
3 Мета та призначення розробки.....	2
4 Джерела розробки.....	2
5 Технічні вимоги.....	2
5.1 Вміст проекту.....	2
5.2 Показники призначення.....	3
5.3 Вимоги до функціональних характеристик.....	3
5.4 Вимоги до архітектури.....	3
5.5 Вимоги до надійності.....	3
5.6 Умови експлуатації.....	4
5.7 Вимоги до складу та параметрів технічних засобів.....	4
5.8 Вимоги до інформаційної і програмної сумісності.....	4
5.8.1 Обладнання.....	4
5.8.2 Мова програмування.....	4
5.8.3 Вхідні дані.....	5
5.8.4 Вихідні дані.....	5
6 Вимоги до програмної документації.....	5
7 Перелік документів, що розробляються.....	5
8 Етапи розробки.....	6
9 Порядок контролю та приймання.....	6

					ВКРБ-125.24.0040.00.00.ТЗ			
<i>Вим.</i>	<i>Арк.</i>	<i>№ документа</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розробив</i>	<i>Дубіцька М.М.</i>				<i>Програмне забезпечення системи кібербезпеки для стеження за мобільним пристроєм з використанням GPS</i>	<i>Літ.</i>	<i>Аркуш</i>	<i>Аркушів</i>
<i>Перевірів</i>	<i>Смірнов С.А.</i>					<i>Б</i>	<i>1</i>	<i>6</i>
<i>Н. Контр.</i>	<i>Коваленко А.С</i>				<i>ЦНТУ КБ-21-3СК</i>			
<i>Затв.</i>	<i>Смірнов О.А.</i>							

1 Найменування та область застосування

Це технічне завдання розповсюджується на розробку системи кібербезпеки для стеження за мобільним пристроєм з використанням GPS.

2 Підстава для розробки

Підставою для розробки служить завдання на випускню кваліфікаційну роботу за першим (бакалаврським) рівнем вищої освіти, видане на кафедрі кібербезпеки та програмного забезпечення (нак. № 136-02 від 01.04.2024 року).

3 Мета та призначення розробки

Метою випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти є розробка програмного забезпечення системи кібербезпеки для стеження за мобільним пристроєм з використанням GPS.

4 Джерела розробки

Джерелом цієї випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти є стосовна до теми література і існуючі аналоги.

5 Технічні вимоги

5.1 Склад продукції

Складниками розробки є:

- вибір і обґрунтування методів реалізації проекту;

					ВКРБ-125.24.0040.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		2

- розробка програмної частин системи, а також розробка взаємодії системи кібербезпеки з ОС та з користувачем;
- розробка програми, що реалізує спроектовані алгоритми роботи системи.

5.2 Показники призначення

Система повинна забезпечувати:

- системи кібербезпеки для стеження за мобільним пристроєм з використанням GPS;
- цілісність даних у процесі роботи та при зберіганні;
- простий, інтуїтивно зрозумілий інтерфейс.

5.3 Вимоги до функціональних характеристик

Розроблене програмне забезпечення не повинно мати обмежень на версію драйверів та операційної системи.

5.4 Вимоги до архітектури

Компонент, що розробляється повинен використовувати системні засоби та апаратні засоби, що на даному етапі розвитку обчислювальної техніки найбільше поширені.

5.5 Вимоги до надійності

Програмні модулі написані по всім правилам, які стосуються стандартних викликів процедур, функцій, методів і форм, визначених технічною документацією на середовище розробки.

					ВКРБ-125.24.0040.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		3

5.6 Умови експлуатації

Робочі місця користувачів ПЗ повинні задовольняти наступним умовам експлуатації:

- температура повітря: 19-20 град. по Цельсію;
- відносна вологість повітря до 80%;
- атмосферний тиск 107 кПа.

5.7 Вимоги до складу та параметрів технічних засобів

Програмне забезпечення повинно бути реалізоване на мобільному пристрої під керівництвом ОС Android і з сумісними з цією платформою пристроями і прикладним програмним забезпеченням.

5.8 Вимоги до інформаційної і програмної сумісності

Переносність програмного забезпечення повинна бути забезпечена за рахунок його реалізації стандартного інтерфейсу взаємодії з ОС, що працюють під управлінням ОС Android.

5.8.1 Обладнання

Комп'ютер Intel® Celeron/8 Mb/1.2 Gb/SVGA 14" 1Mb або сумісні з ним.

5.8.2 Мова програмування

Середовище Delphi 10.4.

					ВКРБ-125.24.0040.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		2

5.8.3 Вхідні дані

Опис алгоритму роботи запропонованої системи.

5.8.4 Вихідні дані

Робоча програма.

6 Вимоги до програмної документації

Програмна продукція повинна бути представлена у виді опису структури даних, схем та опису алгоритму, а також текстів вихідних модулів програмного забезпечення згідно ЄСПД .

7 Перелік документів, що розробляються

- Структурна схема системи – 1 аркуш.
- Функціональна схема системи – 1 аркуш.
- Діаграма процесів – 1 аркуш.
- Блок-схема алгоритму роботи програми – 2 аркуша.
- Пояснювальна записка – 73 аркуші.

8 Етапи розробки

8.1 Збір і обробка інформації по темі випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти. Постановка задачі на виконання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти (складання ТЗ).

					ВКРБ-125.24.0040.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		5

8.2 Проведення досліджень або експериментальних робіт для уточнення основних положень випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти.

8.3 Розробка функціональних схем, блок схем алгоритмів роботи програмного забезпечення.

8.4 Побудова схем взаємодії даних.

8.5 Створення прототипу ПЗ.

8.6 Віднаходження ПЗ, аналіз отриманих результатів.

8.7 Оформлення пояснювальної записки і виконання робіт по графічній частині.

9 Порядок контролю та приймання

9.1 Подання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти на попередній захист 23.05.2024 р.

9.2 Подання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти на захист 14.06.2024 р.

					ВКРБ-125.24.0040.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		6

Додаток Б
(обов'язковий)

Міністерство освіти і науки України
Центральноукраїнський національний технічний університет

ЗАТВЕРДЖУЮ

Керівник випускної кваліфікаційної роботи за
першим (бакалаврським) рівнем вищої освіти
_____ Смірнов С.А.

*Програмне забезпечення системи кібербезпеки для стеження за мобільним
пристроєм з використанням GPS*

Лістинг програми

Код документу 12

Носій: CD/DVD-диск / USB-флеш-накопичувач

Загальна кількість аркушів: 40

Літера: РП

Кропивницький – 2024 року

НАЛАШТУВАННЯ ПАРАМЕТРІВ GPS.PAS

```

unit GPS; // Форма отримання геопараметричних даних
{
Програмне забезпечення системи кібербезпеки для стеження за мобільним пристроєм
з використанням GPS
Дубіцька Марина Миколаївна
гр. КВ-21-ЗСК, 2024
}
Interface // Інтерфейсна частина - об'ява даних
// об'ява бібліотек
SysUtils, WinTypes, WinProcs, Messages, Classes, Graphics, Controls,
Forms, Dialogs, ExtCtrls, Buttons, Grids, StdCtrls, Clipbrd, Printers;

Type // об'ява власних типів даних
TMouseEvent = procedure (Sender: TObject; Shift: TShiftState; Points: TRect)
of object;
TGPSDataArray=array[0..16]of double;

TGPS = class(TPanel) // розроблений клас від TPanel
private
MAXELNUM:integer; //об'ява змінних класу
npolypoints:word;
MapLimits:TRect;
Titulo: string;
Nbloqs:word;
viewpct,viewext,viewbmp:TRect;
FLegendChartPos,FLegendBulletsPos,FLegendShadowsPos:TRect;
FZoomBox:TRect;
arrfname:string;
bitmap:tbitmap;
backbitmap:tbitmap;
logobitmap:tbitmap;
tometa:boolean;
bmpfname,polfname:string;
Fready:boolean;
sel:TSELARRAY;
BB_anchorx,BB_anchory,BB_curx,BB_cury:integer;
BB_drawing:boolean;
FPaintMode:integer;
BackLink:array[1..5] of string;
nBacs:integer;
fgrid:integer;
paleta:hpalette;
FOnMouseDown:TMouseEvent;
FOnMouseUp:TMouseEvent;
FOnMouseMove:TMouseMoveEvent;
FOnMouseLine:TMouseEvent;
// Обробка сповіщень WM_...
Procedure WMRButtonDown(var msg:TWMRButtonDown); message WM_RBUTTONDOWN;
Procedure WMLButtonUp(var msg:TWMLButtonUp); message WM_LBUTTONUP;
Procedure WMRButtonUp(var msg:TWMRButtonUp); message WM_RBUTTONUP;
Procedure WMMouseMove(var msg:TWMMouseMove); message WM_MOUSEMOVE;
FOnMouseBox:TMouseEvent;
procedure time(sender:tobject);
procedure replaceglis;
Procedure CalcChartRatios;
Procedure SetPBGGPS(k:integer;c:tGPS);
function GetPBGGPS(k:integer):tGPS;
Procedure SetPFfGPS(k:integer;c:tGPS);
function GetPFfGPS(k:integer):tGPS;
Procedure SetBitMap(fname:string);
Procedure DrawBitMap(h:hdc);
Procedure CleanHDC(h:hdc;col:tGPS;box:TRect);
Procedure Meridians(h:hdc;box:TRect);
Procedure SetHDCFrame(h:hdc;r:TRect);
Procedure SetChartValue(element,item:word;value:real);

```

```

Function  GetChartValue(element,item:word):real;
Procedure WMF_clip2file(wnd:HWND;fname:string);
Procedure SetPPattern(k:integer;c:byte);
function  GetPPattern(k:integer):byte;
Procedure SetChartLegend(k:word;s:string);
function  GetChartLegend(k:word):string;
procedure ClpWMF2Text(fname:string;textfname:string);
Function  ClpWMFRect(fname:string):trect;
Procedure ScaleClpWMF2Text(fname,textfname:string;iso:boolean);
Procedure RectToPolyLine(r2:trect;var cuadro:array of tpoint);
procedure adaptlegrect(r,r1:trect;var r2:trect);
Procedure SetFont(i:integer;f:tfont);
Procedure MyMouseMove(Sender:TObject; Shift:TShiftState; X, Y: Integer);
Procedure WMLButtonDown(var msg:TWMLButtonDown); message WM_LBUTTONDOWN;
Procedure Loaded;Override;

public
drw:tdraws; //об'ява змінних класу
np:TSPOLARRAY;
Arr:TSArrange;
InfoPanel,
CLegPanel,
BLegPanel,
SLegPanel,
ChartPanel:TGMovPan;
UserLine:record
    p:array[1..300] of tpoint;
    np:integer;
    drawing:boolean;
    dist:double;
end;
MyVersion:string;
gli:TGlyphList;
timer:ttimer;
Procedure LoadGlyphsBitmap(fname:string);
procedure addgli(p:tpoint);
procedure placegli(ix:integer;p:tpoint);
function  GlyphCount:integer;
Procedure GlyphSettings(ixglyph:integer;imageix:integer;
    animate:boolean;enable:boolean);
constructor create(sender:tcomponent);override;
destructor  destroy;override;
Procedure  paint;override;
Procedure  CreatePanels;
Procedure  FromFile(fname:string);
Procedure  ToFile(fname:string);
Procedure GetPoint(elem,ix:integer;var c:tcitypoint);
Function  RemovePoint(elem,ix:integer):boolean;
Function  GetPointCount(elem:integer):integer;
Procedure ClearPoints(elem:integer);
Procedure SetSymbol(f:tfont;c:char);
Procedure ForwardLink(ix:integer);
Procedure BackwardLink;
Procedure LoadPaletteFromFile(filename:string);
Procedure LoadMosaicBitmap(fname:string);
Procedure SetMapOptions(var op:tmapop);
Procedure SetChartOptions(var op:tchop);
Procedure SetBulletOptions(var op:BullOptions);
Procedure SetShadowOptions(var op:tshop);
Procedure SetThemeShadowOptions(var op:TSHThem);
Procedure SetBackgroundOptions(var op:tbkop);
Procedure SetLayerOptions(var op:TLAYOP);
Procedure GetMapOptions(var op:tmapop);
Procedure GetChartOptions(var op:tchop);
Procedure GetBulletOptions(var op:BullOptions);
Procedure GetShadowOptions(var op:tshop);
Procedure GetThemeShadowOptions(var op:TSHThem);
Procedure GetBackgroundOptions(var op:tbkop);
Procedure GetLayerOptions(var op:TLAYOP);
Procedure Merge(fname:string;layeroffset:integer; x,y:integer);

```

```

Procedure Flip(horz,vert:boolean);
Procedure ScaleTo(r:trect);
Procedure AddPoly(closed:boolean);
Procedure DeleteElements;
Procedure JoinElements(orig,dest:integer);
Procedure ToFront(elem:integer);
Procedure ToBack(elem:integer);
Procedure ReMap(var m:array of integer);
// завдання властивості наступним типам даних
Property Ready:boolean read Fready write fready;
Property NumBox:boolean read GetNumBox write SetNumBox;
Property Selected[i:word]:boolean read GetElSelected write SetElSelected;
Property Pol_Layer[i:word;j:word]:byte read GetPolLayer write SetPolLayer;
Property PolCount[i:word]:byte read GetPolCount;
Property PolNames[i:word;j:word]:string read GetPolName write SetPolName;
Property Links[i:word]:string read GetElmLink write SetElmLink;
Property Fonts[i:integer]:Tfont read GetFont write SetFont;
Property canvas;
Property PaintMode:integer read FPaintMode write FPaintMode;
Property ElementCount:word read npolypoints;
Property ElementName[i:word]:string read getElmName write SetElmName;
Property ZoomBox:TREct read FZoomBox write FZoomBox;
property PBgGPS[i:integer]:tGPS read GetPBgGPS write SetPBgGPS;
property PFgGPS[i:integer]:tGPS read GetPFgGPS write SetPFgGPS;

Implementation // Реалізація об'явлених даних у інтерфейсній частині

// конструктор класу TGPS
constructor TGPS_MOBILE.create(sender:tcomponent);
begin //початок
inherited create(sender);
ready:=false;
fgrid:=8;
MaxElNum:=TGPS_MAXELEMENTS;
MyVersion:=TGPS_Version;
BackLink[1]:='';
nbacks:=0;

FPaintMode:=0;
UserLine.drawing:=false;

npolypoints:=0;
Arr:=nil;
bitmap:=nil;
backbitmap:=tbitmap.create;
logobitmap:=tbitmap.create;

drw:=tdraws.create;
tometa:=false;
Align:=AlClient;

paleta:=GetDefaultPalette;
FLegendChartPos:=rect(1,70,30,99);
FLegendBulletsPos:=rect(31,70,60,99);
FLegendShadowsPos:=rect(61,70,98,99);
bmpfname:='';
arrfname:='';
polfname:='';
zoombox:=rect(1,1,100,100);
viewpct:=rect(1,1,99,99);
end;
// завантаження
Procedure TGPS_MOBILE.Loaded;
Begin //початок
  Inherited Loaded;
  CreatePanels;
end;

procedure createmasks(bm:tbitmap);

```

```

var t:tGPS;
    x,y,h:integer;
begin
    h:=bm.height;
    bm.height:=h*2;
    t:=bm.canvas.pixels[0,0];
    for x:=0 to bm.width-1 do for y:=0 to h-1 do
        if bm.canvas.pixels[x,y]=t then bm.canvas.pixels[x,y+h]:=$00FFFFFF
        else bm.canvas.pixels[x,y+h]:=$00000000
    end;
end;

```

```

begin
if not (ComponentState=[]) then exit;
gli:=TGlyphList.create(canvas);
timer:=ttimer.create(self);
timer.interval:=500;
timer.ontimer:=time;
timer.enabled:=false;
InfoPanel:=TGMovPan.create(self);
infopanel.Left:= 32;    infopanel.width:= 64;
infopanel.top:=8;    infopanel.height:= 20;
infopanel.Align:= alnone;    infopanel.marge:=3;
infopanel.borderwidth:=0;    infopanel.bevelwidth:=1;
infopanel.BevelOuter:= bvLowered;    infopanel.BevelInner:= bvNone;
infopanel.Caption:= 'GPS 2.0';    infopanel.visible:=true;
InfoPanel.parent:=self;
if bitmap<>nil then SetBitmap(bmpfname);
if polfname<>' then Fromfile(polfname);
end;

```

```

// деструктор класу TGPS
destructor TGPS_MOBILE.destroy;
Var //розділ об'яви змінних
i:integer;
begin
gli.Free;
for i:=1 to npolypoints do np[i].free;
npolypoints:=0;
arr.free;
bitmap.free;
backbitmap.free;
logobitmap.free;
drw.free;
DeleteObject(paleta);
timer.free;

```

```

InfoPanel.free;
CLegPanel.free;
BLegPanel.free;
SLegPanel.free;
ChartPanel.free;
inherited destroy;
end;

```

```

//отримання даних з файлу
Procedure TGPS_MOBILE.FromFile(fname:string);
var f:file;
i,j:integer;
ax:shortstring;
bloque:record
codigo:word;
nombre:string[50];
numpolis:word;
nummarcos:word;
end;
elemento:record
fillGPS,borderGPS:longint;
numpuntos:word;

```

```

layer:word;
other:longint;
nombre:str50;
end;
buffer: ^SPltSmallPoint;

tfn:tfontname;
rt:trect;
nupu,ipu:word;
c:tsapoint;

begin //початок
if not fileexists(fname) then exit;
polfname:=fname;
if not (ComponentState=[]) then exit;

new(buffer);
assignfile(f,fname); // підключення файлу
reset(f,1); // встановлення типу роботи з файлом

blockread(f,ax,100); // читання даних з фалу
if ax<>MyVersion then begin
closefile(f); // закриття файлу
dispose(buffer);
exit;
end;
screen.cursor:=crHourGlass;
for i:=1 to npolypoints do np[i].free;
npolypoints:=0;
ready:=false;
c:=tsapoint.create;
try // спроба з можливим викликом виключення
if nbloqs> MaxElNum then nbloqs:=maxelnum;
npolypoints:=nbloqs;
for i:=1 to npolypoints do begin
blockread(f,bloque,sizeof(bloque)); // читання даних з фалу
np[i]:=TSPolyPol.create(bloque.nummarcos,bloque.numpolis,bloque.nombre);
for j:=1 to bloque.nummarcos do begin
blockread(f,elemento,sizeof(elemento)); // читання даних з фалу
blockread(f,buffer^,sizeof(tSmallPoint)*elemento.numpuntos);
np[i].SetMValsSmall(elemento.fillGPS,elemento.borderGPS,j,elemento.numpuntos,buffer^);
np[i].m[j].drawer:=drw;
np[i].m[j].layer:=elemento.layer;
np[i].m[j].nombre:=elemento.nombre;
end;

for j:=1 to bloque.numpolis do begin
blockread(f,elemento,sizeof(elemento)); // читання даних з фалу
blockread(f,buffer^,sizeof(tSmallPoint)*elemento.numpuntos);
np[i].SetPValsSmall(elemento.fillGPS,elemento.borderGPS,j,elemento.numpuntos,buffer^);
np[i].p[j].drawer:=drw;
np[i].p[j].layer:=elemento.layer;
np[i].p[j].nombre:=elemento.nombre;
end;
blockread(f,nupu,sizeof(nupu)); // читання даних з фалу
np[i].npoints:=nupu;
for ipu:=1 to nupu do begin
blockread(f,c.c,sizeof(c.c)); // читання даних з фалу
np[i].points[ipu]:=c;
end;
blockread(f,np[i].eo,sizeof(np[i].eo)); // читання даних з фалу
blockread(f,np[i].link,sizeof(np[i].link)); // читання даних з фалу
blockread(f,np[i].values,sizeof(np[i].values)); // читання даних з фалу
np[i].CalcGPS;
np[i].drawer:=drw;
end;
finally // виконати у будь якому випадку навідь при появи виключення

```

```

closefile(f); // закриття файлу
dispose(buffer);
c.free;
drw.setfonts;
arr.free;
arr:=TSArrange.create(2,npolypoints,'Normal');
if bmpfname<>' ' then setbitmap(bmpfname);
if arrfname<>' ' then ArrFromFile(arrfname);
CalcGPS;
ready:=true;
zoombox:=MapLimits;
infopanel.caption:=title;
adaptlegreect(rect(left,top,left+width,top+height),FLegendChartPos,rt);
ClegPanel.Left:= rt.left;    ClegPanel.width:= rt.right-rt.left;
ClegPanel.top:=rt.top;    ClegPanel.height:= rt.bottom-rt.top;
adaptlegreect(rect(left,top,left+width,top+height),FLegendBulletsPos,rt);
BlegPanel.Left:= rt.left;    BlegPanel.width:= rt.right-rt.left;
BlegPanel.top:=rt.top;    BlegPanel.height:= rt.bottom-rt.top;
adaptlegreect(rect(left,top,left+width,top+height),FLegendShadowsPos,rt);
SlegPanel.Left:= rt.left;    SlegPanel.width:= rt.right-rt.left;
SlegPanel.top:=rt.top;    SlegPanel.height:= rt.bottom-rt.top;
infopanel.Left:= left+4;
infopanel.top:=top+4;
repaint;
screen.cursor:=crdefault;
end;
end;

//збереження даних до вайлу
Procedure  TGPS_MOBILE.ToFile(fname:string);
var f:file;
i,j:integer;
ax:string;
bloque:record
codigo:word;
nombre:string[50];
numpolis:word;
nummarcos:word;
end;
elemento:record
fillGPS,borderGPS:longint;
numpuntos:word;
layer:word;
other:longint;
nombre:str50;
end;
buffer:^SPltSmallPoint;
nupu,ipu:word;
c:tsapoint;
begin //початок
new(buffer);
c:=tsapoint.create;
assignfile(f,fname); // підключення файлу
rewrite(f,1); // встановлення типу роботи з файлом
fillchar(ax,100,20);
ax:=MyVersion;
i:=0;
nbloqs:=npolypoints;
blockwrite(f,nbloqs,sizeof(nbloqs));
for i:=1 to npolypoints do begin
bloque.nummarcos:=np[i].nmarcos;
bloque.numpolis:=np[i].npolis;
bloque.nombre:=np[i].nombre;
bloque.codigo:=i;
blockwrite(f,bloque,sizeof(bloque));
for j:=1 to bloque.nummarcos do begin
elemento.fillGPS:=np[i].m[j].bkcol;
elemento.borderGPS:=np[i].m[j].frcol;
elemento.numpuntos:=np[i].m[j].n;

```

```

elemento.layer:=np[i].m[j].layer;
elemento.other:=0;
elemento.nombre:=np[i].m[j].nombre;
blockwrite(f,elemento,sizeof(elemento));
Copy32To16(np[i].m[j].pt^,buffer^,elemento.numpuntos);
blockwrite(f,buffer^,sizeof(TSmallPoint)*elemento.numpuntos);
end;
for j:=1 to bloque.numpolis do begin
elemento.fillGPS:=np[i].p[j].bkcol;
elemento.borderGPS:=np[i].p[j].frcol;
elemento.numpuntos:=np[i].p[j].n;
elemento.layer:=np[i].p[j].layer;
elemento.other:=0;
elemento.nombre:=np[i].p[j].nombre;
blockwrite(f,elemento,sizeof(elemento));
Copy32To16(np[i].p[j].pt^,buffer^,elemento.numpuntos);
move( np[i].p[j].pt^,buffer^,elemento.numpuntos*sizeof(tpoint));
blockwrite(f,buffer^,sizeof(tsmallpoint)*elemento.numpuntos);
end;
nupu:=np[i].npoints;
blockwrite(f,nupu,sizeof(nupu));
for ipu:=1 to nupu do begin
c:=np[i].points[ipu];
blockwrite(f,c.c,sizeof(c.c));
end;
  blockwrite(f,np[i].eo,sizeof(np[i].eo));
  blockwrite(f,np[i].link,sizeof(np[i].link));
  blockwrite(f,np[i].values,sizeof(np[i].values));
end;

closefile(f); // закриття файлу
dispose(buffer);
c.free;
end;

//читання даних
Procedure TGPS_MOBILE.ReadData(fname:string);
var f:file of TGPSTDataArray; TGPSTDataArray=array[0..16]of double;
ad:TGPSTDataArray;
i,j:integer;
begin
assignfile(f,fname); // підключення файлу
reset(f); // встановлення типу роботи з файлом
i:=1;
try // спроба з можливим викликом виключення
while ((not eof(f)) and (i<elementcount)) do begin
read(f,ad);
for j:=0 to 16 do np[i].values[j]:=ad[j];
inc(i);
end;
finally // виконати у будь якому випадку навідь при появі виключення
closefile(f); // закриття файлу
end;
end;

//зберігання даних
Procedure TGPS_MOBILE.SaveData(fname:string);
var f:file of TGPSTDataArray; TGPSTDataArray=array[0..16]of double;
ad:TGPSTDataArray;
i,j:integer;
begin
assignfile(f,fname); // підключення файлу
rewrite(f); // встановлення типу роботи з файлом
try // спроба з можливим викликом виключення
for i:=1 to elementcount do begin
for j:=0 to 16 do ad[j]:=np[i].values[j];
write(f,ad);
end;
finally // виконати у будь якому випадку навідь при появі виключення

```

```

closefile(f); // закриття файлу
end;
end;
// отримання даних форми - шрифту
Function TGPS_MOBILE.GetFont(i:integer):tfont;
Begin //початок
case i of
1:GetFont:=drw.font1;
3:GetFont:=drw.font3;
4:GetFont:=drw.font4;
5:GetFont:=drw.font5;
else GetFont:=drw.font2;
end;
end;

// реалізація внутрішньої процедури класу TGPS_MOBILE ArrCalcBoxes
Procedure TGPS_MOBILE.ArrCalcBoxes;
var i,j:integer;
begin
for i:=1 to arr.n do begin
prepect(arr.caja[i]);
for j:=1 to arr.t do if arr.matr.bit[i,j] then begin
comprect(np[j].caja,arr.caja[i]);
end;end;end;

// реалізація внутрішньої процедури класу TGPS_MOBILE ArrFromFile
Procedure TGPS_MOBILE.ArrFromFile(fname:string);
Begin
arrfname:=fname;
if not fileexists(fname) then exit;
arr.fromfile(fname);
arrcalcboxes;
end;

// реалізація внутрішньої процедури класу TGPS_MOBILE getElmName
Function TGPS_MOBILE.getElmName(i:word):string;
begin
if ((i>0) and (i<=npolypoints)) then GetElmName:=np[i].nombre
else GetElmName:='';
end;

// реалізація внутрішньої процедури класу TGPS_MOBILE SetElmName
Procedure TGPS_MOBILE.SetElmName(i:word;s:string);
begin
if ((i>0) and (i<=npolypoints)) then np[i].nombre:=s;
end;

// реалізація внутрішньої процедури класу TGPS_MOBILE SetSel
Procedure TGPS_MOBILE.SetSel(lb:tlistbox);
var i:integer;
begin
for i:=1 to lb.items.count do sel[i]:=(lb.selected[i-1]);
end;
// реалізація внутрішньої процедури класу TGPS_MOBILE IsArrSelected
Function TGPS_MOBILE.IsArrSelected(father:word):boolean;
var i:integer;
begin //початок
IsArrSelected:=true;
for i:=1 to npolypoints do if sel[i] then
if arr.Esta(father,i) then exit;
IsArrSelected:=false;
end;

// реалізація внутрішньої процедури класу TGPS_MOBILE DoChart
Procedure DoChart(h:hdc;r:trect);
var kx,ky,i,j:integer;
ox,oy:integer;
v:integer;
x0,y0:integer;

```

```

bola:integer;
ax:string;
mx:integer;
maximovalor,total:real;
caxa:trect;
pu:tpoint;
suma:array[0..16] of integer;
ds:array[1..16] of double;
dd:double;
a:array[0..255] of char;
centro:tpoint;
begin
ChartPanel.font:=font;
chartpanel.GPS:=drw.co.boxbackGPS;
centro.x:=(r.left+r.right) div 2;
centro.y:=(r.top+r.bottom) div 2;
selectobject(h,getstockobject(BLACK_PEN));
for i:=1 to 16 do ds[i]:=0;
maximovalor:=0;
total:=0;
for i:=drw.co.nvalues downto 1 do
for j:=1 to elementcount do if sel[j] then begin
ds[i]:=ds[i]+np[j].values[i];
if ds[i]>maximovalor then maximovalor:=ds[i];
end;
if maximovalor=0 then maximovalor:=1;
for i:=1 to drw.co.nvalues do total:=total+ds[i];

SetBkMode(h,1);
setTextAlign(h,TA_CENTER+TA_TOP);
str(total:3:drw.co.ndecimals,ax);
ax:='Всего: '+ax;
strpcopy(a,ax);
textout(h,centro.x,r.top+2,a,length(ax));
caxa:=r;
ox:=(caxa.right-caxa.left) div 5;
oy:=(caxa.bottom-caxa.top) div 5;
kx:=(caxa.right-caxa.left-2*ox) div drw.co.nvalues;
ky:=3*oy;
bola:=4;
mx:=kx div 2;
if ((ox<5) or (oy<5)) then exit;

if drw.co.style<>4 then begin
movetoex(h,caxa.left+ox,caxa.bottom-oy,nil);
lineto(h,caxa.left+ox+drw.co.nvalues*kx,caxa.bottom-oy);
MoveToEx(h,caxa.left+ox,caxa.bottom-oy,nil);
lineto(h,caxa.left+ox,caxa.bottom-oy - ky);
end;

case drw.co.style of
0:begin
for i:=1 to drw.co.nvalues do begin
drw.getbrocha(h,drw.co.patt[i],drw.co.lineGPS,drw.co.GPSs[i]);
v:=trunc(ds[i]*ky/maximovalor);
rectangle(h,
caxa.left+ox+(i-1)*kx,
caxa.bottom-oy,
caxa.left+ox+i*kx,
caxa.bottom-oy - v);
drw.releasebrocha;
end;
end;
3:begin
for i:=1 to drw.co.nvalues do begin
v:=trunc(ds[i]*ky/maximovalor);
drw.bar3d(h,
caxa.left+ox+(i-1)*kx,
caxa.bottom-oy,

```

```

                caxa.left+ox+i*kx,
                caxa.bottom-oy - v,i);
end;
end;

1,2:begin
  drw.getpen(h,3,drw.co.GPSs[1]);
  x0:=caxa.left+ox+mx;y0:=caxa.bottom-oy -trunc(ds[1]*ky/maximovalor);
  MoveToEx(h,x0,y0,nil);
  for i:=1 to drw.co.nvalues do begin
    v:=trunc(ds[i]*ky/maximovalor);
    lineto(h,caxa.left+ox+i*kx-mx,caxa.bottom-oy - v);
  end;
  drw.releasepen;

  if drw.co.style=2 then begin
    drw.getbrocha(h,drw.co.patt[2],drw.co.LineGPS,drw.co.GPSs[2]);
    for i:=1 to drw.co.nvalues do begin
      v:=trunc(ds[i]*ky/maximovalor);
    end;
    drw.releasebrocha;
  end;
end;

4:begin
  pu:=centro;
  dd:=0; for i:=1 to drw.co.nvalues do begin
    dd:=dd+ds[i];
    ds[i]:=dd;
  end;
  suma[0]:=0; v:=0;
  for i:=1 to drw.co.nvalues do begin
    v:=v+trunc(ds[i]*360/dd);
    suma[i]:=trunc(ds[i]*360/dd);
  end;
  drw.Queso(h,pu.x,pu.y,drw.co.PieSize,0,360,true,
    drw.co.GPSs[1],drw.co.lineGPS,drw.co.GPSs[1]);
  for i:=1 to drw.co.nvalues do
    drw.Queso(h,pu.x,pu.y,drw.co.PieSize,suma[i-1],suma[i],false,
    drw.co.GPSs[1],drw.co.GPSs[i],drw.co.lineGPS);
  end;
end;

if drw.co.style<>4 then begin
  ox:=ox- kx div 2;
  for i:=1 to drw.co.nvalues do begin
    if drw.co.ndecimals=0 then ax:=inttostr(trunc(ds[i]))
    else str(ds[i]:3:drw.co.ndecimals,ax);
    v:=trunc(ds[i]*ky/maximovalor);
    drw.ContrastText(h,caxa.left+ox+i*kx ,
      caxa.bottom-oy - v,ax,3,false);
  end;
end;

end;
// реалізація внутрішньої процедури класу TGPS_MOBILE SelObjects
procedure SelObjects(c:tGPS;f:tfont);
begin
  oldf:=selectobject(h,f.handle);
  setttextGPS(h,f.GPS);
  drw.getbrocha(h,0,f.GPS,c);
  rectangle(h,rto.left,rto.top,rto.right,rto.bottom);
  drw.releasebrocha;
end;
Var //розділ об'яви змінних
  mm,ww:double;
begin
  chartpanel.visible:=true;
  if not tometa then begin
    h:=ChartPanel.canvas.handle;
    rto:=rect(1,1,ChartPanel.width,ChartPanel.height);

```

```

SelObjects (ChartPanel.GPS, drw.font4);
Dochart (h, rto);
end
else begin
mm:=(r.right-r.left)/width;
ww:=(r.bottom-r.top)/height;
rto.left:=trunc (r.left+ChartPanel.left*mm);
rto.top:=trunc (r.top+ChartPanel.top*ww);
rto.right:=trunc (r.left+(ChartPanel.left+ChartPanel.width)*mm);
rto.bottom:=trunc (r.top+(ChartPanel.top+ChartPanel.height)*ww);
DoChart (h, rto);
end;
end;

// реалізація внутрішньої процедури класу TGPS_MOBILE ChartLegend
Procedure ChartLegend (h:hdc; r:trect);
var i:integer;
wy, wb, wm:integer;
a:array[0..255] of char;
begin //початок
wy:=(r.bottom-r.top) div (drw.co.nvalues+1);
wb:=trunc (0.60*wy);
wm:=trunc (0.20*wy);
if abs(font.size) >wy then font.size:=wy;
SetBkMode (h, 1);
setTextAlign (h, TA_CENTER+TA_TOP);
strcpy (a, drw.co.title);
textout (h, (r.left+r.right) div 2, r.top+2, a, length (drw.co.title));
setTextAlign (h, TA_LEFT+TA_BOTTOM);
for i:=1 to drw.co.nvalues do begin
Bar (h, r.left+wb, r.top+wy*i+wm,
r.left+wb+wb, r.top+wy*i+wm+wb, i);
strcpy (a, drw.co.LegTexts[i]);
textout (h, r.left+3*wb, r.top+wy*i+wm+wb, a, length (drw.co.LegTexts[i]));
end;
end;
Procedure BulletLegend (h:hdc; rl:trect);
var i, j, k, k0, k2:integer;
wy, wm, wb:integer;
a:array[0..255] of char;
q:real;
ax:string;
pc:pchar;
oldf:hfont;
h1, h2, h3:integer;
f:tfont;
fh1:integer;
begin
case drw.bo.applyAs of
0, 2:begin
k0:=trunc (drw.bo.size);
wb:=k0 div drw.bo.legendpoints;
q:=drw.maxval0 / drw.bo.legendpoints;
wy:=(rl.bottom-rl.top) div (drw.bo.legendpoints+1);
wm:=trunc (0.20*wy);
if abs(font.size) >wy then font.size:=wy;
SetBkMode (h, 1);
setTextAlign (h, TA_CENTER+TA_TOP);
strcpy (a, drw.co.LegTexts[drw.bo.BulletSource]);
textout (h, (rl.left+rl.right) div
2, rl.top+2, a, length (drw.co.LegTexts[drw.bo.BulletSource]));
setTextAlign (h, TA_LEFT+TA_BOTTOM);
for i:=1 to drw.bo.legendpoints do begin
j:=drw.bo.legendpoints-i+1;
drw.fontsymbol.size:=k*2;
k2:=k;
pc:=@drw.bo.thesymbol;
settextGPS (h, drw.bo.lineGPS);
setbkmode (h, 1);

```

```

        oldf:=selectobject(h,drw.fontsymbol.handle);
        fh1:=abs(drw.fontsymbol.height) div 2 ;
        setTextAlign(h,TA_LEFT+TA_TOP);
        textout(h,rl.left+wm,rl.top+wy*i+wm-fh1,pc,1);
        selectobject(h,oldf);
        str(q*j:3:drw.mo.decimals,ax);
        strpcopy(a,ax);
        textout(h,rl.left+2*k0+wm,rl.top+wy*i+wm,a,length(ax));
        if drw.bo.applyAs=0 then begin
            movetoex(h,rl.left+wm+k0 ,rl.top + wy * i ,nil);
            lineto(h,rl.left+2*k0+wm,rl.top + wy * i);
        end;
    end;
end;
1:begin
    wy:=(rl.bottom-rl.top) div (drw.bo.legendpoints+1);
    wb:=trunc(0.80*wy);
    wm:=trunc(0.10*wy);

    k:=100 div drw.bo.legendpoints;
    q:=drw.maxval0 / drw.bo.legendpoints;

    if abs(font.size) >wy then font.size:=wy;
    SetBkMode(h,1);
    setTextAlign(h,TA_CENTER+TA_TOP);
    ax:=drw.co.LegTexts[drw.bo.BulletSource];
    strpcopy(a,ax);
    textout(h,(rl.left+rl.right) div 2,rl.top+2,a,length(ax));
    setTextAlign(h,TA_LEFT+TA_BOTTOM);

    for i:=1 to drw.bo.legendpoints do begin
        drw.getbrocha(h,0,drw.bo.shadows[k*i],drw.bo.shadows[k*i]);
        ellipse(h,rl.left+wb,rl.top+wy*i+wm,rl.left+wb+wb,rl.top+wy*i+wm+wb);
        drw.releasebrocha;
        str(q*i:3:drw.mo.decimals,ax); strpcopy(a,ax);
        textout(h,rl.left+3*wb,rl.top+wy*i+wm+wb,a,length(ax));
    end;

end;
5:begin
    rectangle(h,rl.left,rl.top,rl.right,rl.bottom);
    h1:=rl.top+font.size*3;
    h2:=rl.top+rl.bottom -h1;
    h2:=h2 div drw.bo.custom.nranges;

    setTextAlign(h,TA_CENTER+TA_TOP);
    strpcopy(a,drw.co.LegTexts[drw.bo.BulletSource]);
    textout(h,(rl.left+rl.right) div
2,rl.top+2,a,length(drw.co.LegTexts[drw.bo.BulletSource]));
    setTextAlign(h,TA_LEFT+TA_TOP);

    for i:=1 to drw.bo.custom.nranges do begin
        strpcopy(a,drw.bo.custom.ranges[i].text);
        textout(h,44,h1+(i-1)*h2 + (drw.bo.custom.ranges[i].size div
2),a,length(drw.bo.custom.ranges[i].text));
    end;
    settextGPS(h,drw.bo.lineGPS);
    f:=tfont.create;
    f.name:=drw.bo.custom.fontname;
    for i:=1 to drw.bo.custom.nranges do begin
        f.size:=drw.bo.custom.ranges[i].size;
        oldf:=selectobject(h,f.handle);
        pc:=@drw.bo.custom.ranges[i].symbol;
        textout(h,16,h1+(i-1)*h2 ,pc,1);
        selectobject(h,oldf);
    end;
    f.free;
end;
end;
end;

```

```

end;
// реалізація внутрішньої процедури класу TGPS_MOBILE CalcChartRatios
Procedure TGPS_MOBILE.CalcChartRatios;
var i,j:integer;
mx,mx0,mx1:real;
begin //початок
mx:=0;mx0:=0;mx1:=0;
for i:=1 to npolypoints do if sel[i] then begin
if np[i].values[drw.bo.bulletsource]>mx0 then
mx0:=np[i].values[drw.bo.bulletsource];
if np[i].values[drw.so.source]>mx1 then mx1:=np[i].values[drw.so.source];
for j:=1 to drw.co.nvalues do
if np[i].values[j]>mx then mx:=np[i].values[j];
end;
drw.maxval0:=mx0;
drw.maxvall:=mx1;
drw.maxvals:=mx;
end;

// реалізація внутрішньої процедури класу TGPS_MOBILE SetHDCFrame
Procedure TGPS_MOBILE.SetHDCFrame(h:hdc;r:trect);
begin
viewext:=MapLimits;
drw._view:=fzoombox;
drw._back:=r;
adaptlegrect(r,viewpct,drw._frame);
if drw.mo.aspect=1 then begin
if drw._view.right-drw._view.left<drw._view.bottom-drw._view.top
then begin drw._view.right:=drw._view.left+drw._view.bottom-drw._view.top;end
else begin drw._view.bottom:=drw._view.top+drw._view.right-drw._view.left;end;
end;
if (drw._view.right-drw._view.left) >0 then
drw.Rx:=(drw._frame.right-drw._frame.left)/(drw._view.right-drw._view.left)
else drw.Rx:=1;
if (drw._view.bottom-drw._view.top)>0 then
drw.Ry:=(drw._frame.bottom-drw._frame.top)/(drw._view.bottom-drw._view.top)
else drw.Ry:=1;
if drw.rx=0 then drw.rx:=1;
if drw.ry=0 then drw.ry:=1;
end;

Procedure TGPS_MOBILE.FillNames(lb:tlistbox);
var i,j,k:integer;
begin //початок
lb.clear;
for i:=1 to npolypoints do lb.items.add(inttostr(i)+' ':'+np[i].nombre);
end;

Procedure TGPS_MOBILE.FillArranges(lb:tlistbox);
begin
if not ready then exit;
arr.alltolistbox(lb);
end;

procedure TGPS_MOBILE.ArrSet(father,son:word;ok:boolean);
begin //початок
if not ready then exit;
arr.Poner(father,son,ok);
end;

Procedure TGPS_MOBILE.ArrFromListBox(lb:tlistbox;father:word);
begin
if not ready then exit;
arr.FromListBox(lb,father);
ArrCalcBoxes;
end;

Procedure TGPS_MOBILE.ArrToListBox(lb:tlistbox;father:word);
begin

```

```

if not ready then exit;
arr.ToListBox(lb, father);
end;

// реалізація внутрішньої процедури класу TGPS_MOBILE ArrSetName
Procedure TGPS_MOBILE.ArrSetName(name:string; father:word);
begin
if not ready then exit;
if father>arr.n then exit;
arr.titulos[father]^:=name;
end;

Procedure TGPS_MOBILE.ArrSetNumber(n:word);
begin
arr.setnumber(n);
end;

Function TGPS_MOBILE.ArrGetNumber:word;
begin
ArrGetNumber:=arr.n;
end;

Procedure TGPS_MOBILE.SelArrGroup(father:integer; ok:boolean);
var i:integer;
begin
if father>0 then
if father<=arr.n then
for i:=1 to arr.t do if arr.Matr.Bit[father,i] then sel[i]:=ok;
end;

// реалізація внутрішньої процедури класу TGPS_MOBILE GetElSelected
Function TGPS_MOBILE.GetElSelected(i:word):boolean;
begin
GetElSelected:=false;
if i>0 then if i<=npolypoints then GetElSelected:=Sel[i];
end;

Procedure TGPS_MOBILE.SetElSelected(i:word; ok:boolean);
begin
if i>0 then if i <=npolypoints then Sel[i]:=ok;
end;

Procedure TGPS_MOBILE.SetAllSelections(ok:boolean);
var i:integer;
begin
for i:=1 to elementcount do selected[i]:= ok;
end;

// реалізація внутрішньої процедури класу TGPS_MOBILE SetBitMap
Procedure TGPS_MOBILE.SetBitMap(fname:string);
begin
bmpfname:=fname;
bitmap.free;
bitmap:=nil;
if fileexists(fname) then begin
bitmap:=tbitmap.create;
bitmap.loadfromfile(fname);
drw.mo.mapimage:=2;
end;
end;

// реалізація внутрішньої процедури класу TGPS_MOBILE DrawBitMap
Procedure TGPS_MOBILE.DrawBitMap(h:hdc);
var ew,ow:longint;
w:trect;
vex,vey,vox,voy:integer;
rwb:record x,y:real;end;
wb:trect;

```

```

bm:tbitmap;
i,j:integer;
wx,wy:double;
wf:trect;

begin //початок
if bitmap=nil then exit;
if drw.mo.mapimage=0 then exit;
if drw.mo.mapimage=2 then begin
  wx:=Bitmap.width/(viewext.right-viewext.left);
  wy:=Bitmap.height/(viewext.bottom-viewext.top);

  w:=drw._view;
  if w.left<viewext.left then w.left:=viewext.left;
  if w.top<viewext.top then w.top:=viewext.top;
  if w.right>viewext.right then w.right:=viewext.right;
  if w.bottom>viewext.bottom then w.bottom:=viewext.bottom;
  wb.left:=Trunc((w.left-viewext.left)*wx);
  wb.right:=Trunc((w.right-viewext.left)*wx);
  wb.top:=Trunc((w.top-viewext.top)*wy);
  wb.bottom:=Trunc((w.bottom-viewext.top)*wy);
  drw.ConvertaRect(w,wf);
  if tometa then begin
    bm:=tbitmap.create;
    bm.assign(bitmap);
    StretchBlt(bm.canvas.handle,0,0,wb.right-wb.left,wb.bottom-wb.top,
      bitmap.canvas.handle,wb.left,wb.top,wb.right-wb.left,wb.bottom-
wb.top,SRCCOPY);

    bm.width:=wb.right-wb.left;bm.height:=wb.bottom-wb.top;
    StretchBlt(h,wf.left,wf.top,wf.right-wf.left,wf.bottom-wf.top,
      bm.canvas.handle,0,0,bm.width,bm.height,SRCCOPY);
    bm.destroy;
  end
  else
  with wf do
    StretchBlt(h,left,top,right-left,bottom-top,
      bitmap.canvas.handle,wb.left,wb.top,wb.right-wb.left,wb.bottom-
wb.top,SRCCOPY);
  end
  else begin
  with drw._frame do
    StretchBlt(h,left,top,right-left,bottom-top,
      bitmap.canvas.handle,0,0,bitmap.width,bitmap.height,SRCCOPY);
  end;
end;

Procedure TGPS_MOBILE.CleanHDC(h:hdc;col:tGPS;box:trect);
var b,oldb:hbrush;
p,oldp:hpen;
i,j,k,wy:integer;
gr:TGrArray;
w,hh,x,y:integer;
bm:tbitmap;
begin //початок
CalcColGradients(drw.bko.minGPS,drw.bko.maxGPS,drw.bko.GradItems,gr);
if not tometa then box:=drw._back;
w:=(box.right-box.left) div drw.bko.GradItems;
hh:=(box.bottom-box.top) div drw.bko.GradItems;
b:=drw.GetSolidBrush(gr[drw.bko.GradItems]);
oldb:=selectobject(h,b);

fillrect(h,rect(box.left,box.top,box.right,box.bottom),b);
selectobject(h,oldb);deleteobject(b);

case drw.bko.effect of
1:for i:=1 to drw.bko.GradItems do begin

```

```

b:=drw.GetSolidBrush(gr[i]);
selectobject(h,b);
fillrect(h,rect(box.left,box.top+(i-1)*hh,box.right,box.top+i*hh),b);
selectobject(h,oldb);deleteobject(b);
end;
2:for i:=1 to drw.bko.GradItems do begin
b:=drw.GetSolidBrush(gr[i]);
selectobject(h,b);
fillrect(h,rect(box.left+(i-1)*w,box.top,box.left+i*w,box.bottom),b);
selectobject(h,oldb);deleteobject(b);
end;
3:begin
x:=(box.right-box.left) div 2;
y:=(box.bottom-box.top) div 2;
w:=x div 20;hh:=y div 20;
for i:=drw.bko.GradItems downto 1 do begin
b:=drw.GetSolidBrush(gr[i]);
selectobject(h,b);
p:=createpen(0,0,gr[i]);
oldp:=selectobject(h,p);
ellipse(h,box.left+x-w*i,box.top+y-hh*i,box.left+x+w*i,box.top+y+hh*i);
selectobject(h,oldp);deleteobject(p);
selectobject(h,oldb);deleteobject(b);
end;
end;
4:begin
x:=(box.right-box.left) div 2;
y:=(box.bottom-box.top) div 2;
w:=x div drw.bko.GradItems;hh:=y div drw.bko.GradItems;
for i:=drw.bko.GradItems downto 1 do begin
b:=drw.GetSolidBrush(gr[i]);
selectobject(h,b);
fillrect(h,rect(box.left+x-w*i,box.top+y-hh*i,box.left+x+w*i,box.top+y+hh*i),b);
selectobject(h,oldb);deleteobject(b);
end;
end;
5:begin
if not BackBitmap.empty then begin
x:=(box.right-box.left) div BackBitmap.width;
y:=(box.bottom-box.top) div BackBitmap.height;
for i:=0 to x do for j:=0 to y do
BitBlt(h,box.left+i*BackBitmap.width,box.top+j*BackBitmap.height,BackBitmap.widt
h,BackBitmap.height,BackBitmap.canvas.handle,0,0 ,SRCCOPY);
end;
end;
end;
end;

```

```

Procedure TGPS_MOBILE.Meridians(h:hdc;box:trect);
var i,j,w,hh,x,y:integer;
p,oldp:hpen;
pix:tGPS;
bm:tbitmap;
begin
if drw.bko.lines then begin
if not tometa then box:=drw._back;
w:=drw.bko.pcth;w:=trunc(w*(box.right-box.left)/100.0);
hh:=drw.bko.pctv;hh:=trunc(hh*(box.bottom-box.top)/100.0);
x:=(box.right-box.left) div w; y:=(box.bottom-box.top) div hh;
p:=createpen(0,0,drw.bko.lineGPS);
oldp:=selectobject(h,p);
for i:=1 to x do begin
movetoex(h,box.left+i*w,box.top,nil);
lineto(h,box.left+i*w,box.bottom);
end;
for i:=1 to y do begin
movetoex(h,box.left,box.top+i*hh,nil);
lineto(h,box.right,box.top+i*hh);
end;
end;

```

```

end;
selectobject(h,oldp);
deleteobject(p);
end;
if drw.bko.logoon then begin
i:=LogoBitmap.width div 2;
j:=LogoBitmap.height div 2;
box.left:=box.left+1;box.right:=box.right-
1;box.top:=box.top+1;box.bottom:=box.bottom-1;
case drw.bko.Logopos of
  1:begin x:=box.left;y:=box.top;end;
  2:begin x:=(box.left+box.right) div 2 - i;y:=box.top;end;
  3:begin x:=box.right-LogoBitmap.width;y:=box.top;end;
  4:begin x:=box.left;y:=(box.top+box.bottom) div 2 -j;end;
  5:begin x:=(box.left+box.right) div 2 - i;y:=(box.top+box.bottom) div 2 -
j;end;
  6:begin x:=box.right-LogoBitmap.width;y:=(box.top+box.bottom) div 2 -j;end;
  7:begin x:=box.left;y:=box.bottom-LogoBitmap.height;end;
  8:begin x:=(box.left+box.right) div 2 - i;y:=box.bottom-LogoBitmap.height;end;
  9:begin x:=box.right-LogoBitmap.width;y:=box.bottom-LogoBitmap.height;end;
  else begin x:=box.left;y:=box.top;end;
end;
end;
end;

Procedure TGPS_MOBILE.SetGPSs(t:byte;f,b:tGPS;ix:word);
begin
np[ix].setGPSs(t,f,b);
end;

Procedure TGPS_MOBILE.SetPBGGPS(k:integer;c:tGPS);
var i:integer;
begin
for i:=1 to np[k].npolis do np[k].p[i].bkcol:=c;
for i:=1 to np[k].nmarcos do np[k].m[i].bkcol:=c;
end;

function TGPS_MOBILE.GetPBGGPS(k:integer):tGPS;
begin
if np[k].npolis>0 then
GetPBGGPS:=np[k].p[1].bkcol
else if np[k].nmarcos>0 then
GetPBGGPS:=np[k].m[1].bkcol
else GetPBGGPS:=clwhite;
end;
// реалізація внутрішньої процедури класу TGPS_MOBILE SetPFgGPS
Procedure TGPS_MOBILE.SetPFgGPS(k:integer;c:tGPS);
var i:integer;
begin
for i:=1 to np[k].npolis do np[k].p[i].frcol:=c;
for i:=1 to np[k].nmarcos do np[k].m[i].frcol:=c;
end;

function TGPS_MOBILE.GetPFgGPS(k:integer):tGPS;
begin //початок
if np[k].npolis>0 then
GetPFgGPS:=np[k].p[1].FRcol
else if np[k].nmarcos>0 then
GetPFgGPS:=np[k].m[1].FRcol
else GetPFgGPS:=clBLACK;
end;

Procedure TGPS_MOBILE.SetPPattern(k:integer;c:byte);
var i:integer;
begin
for i:=1 to np[k].npolis do np[k].p[i].trama:=c;
for i:=1 to np[k].nmarcos do np[k].m[i].trama:=c;
end;

```

```

function  TGPS_MOBILE.GetPPattern(k:integer):byte;
begin
if np[k].npolis>0 then
GetPPattern:=np[k].p[1].trama
else if np[k].nmarcos>0 then
GetPPattern:=np[k].m[1].trama
else GetPPattern:=0;
end;

Procedure TGPS_MOBILE.SetChartLegend(k:word;s:string);
begin
if k>16 then exit;
drw.co.LegTexts[k]:=s;
end;

function  TGPS_MOBILE.GetChartLegend(k:word):string;
begin //початок
if k>16 then exit;
GetChartLegend:=drw.co.LegTexts[k];
end;

Procedure TGPS_MOBILE.SetShadows(a,b:tGPS);
begin
drw.calcshadows(a,b);
end;

Procedure TGPS_MOBILE.Applyshadows;
var i,j:integer;
aux:double;
c:integer;
pcts:array[0..250] of record
    f,s:double;
    v:double;
end;

Procedure CalcFrecs;
var i,j,k:integer;
d,m:double;
s:double;
nump:integer;
begin //початок
d:=drw.maxvall / 250;
fillchar(pcts,sizeof(pcts),0);
nump:=0;
for i:=1 to 250 do pcts[i].v:=i*drw.maxvall/250;
for i:=1 to npolypoints do if sel[i] then begin
k:=trunc(np[i].values[drw.so.source] *250/drw.maxvall);
if drw.sot.calc=1 then
pcts[k].f:=pcts[k].f+1
else pcts[k].f:=pcts[k].f+np[i].values[drw.so.source];
inc(nump);
end;
s:=0;
for i:=1 to 250 do begin
s:=s+pcts[i].f;
pcts[i].s:=s;
end;
if drw.sot.calc=1 then m:=nump / drw.so.legendpoints
else m:=s / drw.so.legendpoints;
for i:=1 to drw.so.legendpoints do
for j:=1 to 250 do
if pcts[j].s>= m*i then begin
drw.brkshadow[i]:=pcts[j].v;
break;
end;
drw.brkshadow[drw.so.legendpoints]:=drw.maxvall;
for i:=1 to npolypoints do if sel[i] then begin
m:=np[i].values[drw.so.source];
for j:=1 to drw.so.legendpoints do

```

```

if m<=drw.brkshadow[j] then begin
  np[i].SetGPSs(0,0,drw.sot.colsf[j]);
  break;
begin
calcchartrratios;
drw.brkshadow[drw.so.legendpoints]:=drw.maxvall;
case drw.sot.kind of
0:begin
drw.calcshadows(drw.so.minGPS,drw.so.maxGPS);
for i:=1 to drw.so.legendpoints do
drw.brkshadow[i]:=(drw.maxvall/drw.so.legendpoints)*i;
drw.brkshadow[drw.so.legendpoints]:=drw.maxvall;
for i:=1 to npolypoints do if sel[i] then begin
c:=trunc(np[i].values[drw.so.source]*100/drw.maxvall);
if c<1 then c:=1;if c>100 then c:=100;
np[i].SetGPSs(0,0,drw.shadow[c]);
end;
end;
1:begin
if drw.sot.calc>0 then calcfrecs
else begin
for i:=1 to drw.so.legendpoints do
drw.brkshadow[i]:=(drw.maxvall/drw.so.legendpoints)*i;
drw.brkshadow[drw.so.legendpoints]:=drw.maxvall;
for i:=1 to npolypoints do if sel[i] then begin
c:=trunc(np[i].values[drw.so.source]*drw.so.legendpoints/drw.maxvall);
if c<1 then c:=1;if c>drw.so.legendpoints then c:=drw.so.legendpoints;
np[i].SetGPSs(0,0,drw.sot.colsf[c]);
end;
end;
end;
2:begin
for i:=1 to npolypoints do if sel[i] then begin
aux:=np[i].values[drw.so.source];
for j:=1 to drw.so.legendpoints do
if ((aux>=drw.sot.vals[j].min) and (aux<=drw.sot.vals[j].max)) then begin
  np[i].SetGPSs(0,0,drw.sot.colsf[j]);
  break;
end;
end;

Procedure TGPS_MOBILE.RectToPolyLine(r2:trect;var cuadro:array of tpoint);
Begin //початок
cuadro[0].x:=r2.left;cuadro[0].y:=r2.top;
cuadro[1].x:=r2.right;cuadro[1].y:=r2.top;
cuadro[2].x:=r2.right;cuadro[2].y:=r2.bottom;
cuadro[3].x:=r2.left;cuadro[3].y:=r2.bottom;
cuadro[4].x:=r2.left;cuadro[4].y:=r2.top;
end;

procedure TGPS_MOBILE.adaptlegrect(r,r1:trect;var r2:trect);
begin //початок
r2.left:=trunc((r1.left/100)*(r.right-r.left)+r.left);
r2.top:=trunc((r1.top/100)*(r.bottom-r.top)+r.top);
r2.right:=trunc((r1.right/100)*(r.right-r.left)+r.left);
r2.bottom:=trunc((r1.bottom/100)*(r.bottom-r.top)+r.top);
end;

Procedure TGPS_MOBILE.ClipWMF;
Var //розділ ОБ'ЯВИ змінних
hdcmeta:hdc;
hmeta:hmetafile;
r:trect;
twmf:tmetafile;
begin
tometa:=true;
hdcmeta:=createmetatile(nil);
r:=rect(0,0,width,height);

```

```

SetWindowExtEx(hdcmeta,width,height,nil);
SetWindowOrgEx(hdcmeta,0,0,nil);
SetViewportOrgEx(hdcmeta,0,0,nil);
SetViewportExtEx(hdcmeta,width,height,nil);
Draw(hdcmeta,r);
hmeta:=closemetafile(hdcmeta);
twmf:=tmetafile.create;
twmf.width:=width;
twmf.height:=height;
with TMetafileCanvas.Create(twmf, 0) do
try // спроба з можливим викликом виключення
PlayMetaFile(handle,hmeta);
finally // виконати у будь якому випадку навідь при появі виключення
free;
end;
twmf.enhanced:=false;
clipboard.assign(twmf);
twmf.free;
tometa:=false;
end;
Procedure TGPS_MOBILE.WMF_clip2file(wnd:HWND;fname:string);
Var //розділ об'яви змінних
hGMem:longint;
lpmfp:^tMETAFILEPICT ;
f:file;
a:array[0..250] of char;
begin
OpenClipboard(Wnd);
hGMem:= GetClipboardData(CF_METAFILEPICT) ;
if (hGMem<>0) then begin
lpmfp:= GlobalLock(hGMem);
strcpy(a,fname);
copymetafile(lpmfp^.hmf,a);
GlobalUnlock(hGMem);
end;
CloseClipboard;
end;

procedure TGPS_MOBILE.ClpWMF2Text(fname:string;textfname:string);
var tm:tmetaheader;
f:file;
ft:textfile;
rs:longint;
cf:word;
i,r,j:word;
s:longint;
tlf:tlogfont;
n,np:word;
ltp:^SPltSmallPoint;
begin
new(ltp);
assignfile(f,fname); // підключення файлу
reset(f,1); // встановлення типу роботи з файлом

blockread(f,tm,sizeof(tm)); // читання даних з фалу
n:=0;
while not eof(f) do begin
blockread(f,rs,4); // читання даних з фалу
blockread(f,cf,2); // читання даних з фалу
if cf=$0324 then inc(n);
for j:=1 to rs-3 do blockread(f,cf,2);
end;
closefile(f); // закриття файлу
if n=0 then exit;

assignfile(f,fname); // підключення файлу
reset(f,1); // встановлення типу роботи з файлом

assignfile(ft,textfname); // підключення файлу

```

```

rewrite(ft); // встановлення типу роботи з файлом

blockread(f,tm,sizeof(tm)); // читання даних з фалу
I:=1;

while not eof(f) do begin
blockread(f,rs,4); // читання даних з фалу
blockread(f,cf,2); // читання даних з фалу
if ((cf=$0324) and (i<maxelnum)) then begin
blockread(f,np,2); // читання даних з фалу
blockread(f,ltp^,np*sizeof(TSmallPoint)); // читання даних з фалу
writeln(ft,':Element '+inttostr(i));
writeln(ft,'+P1');
for j:=1 to np do
writeln(ft,inttostr(ltp^[j].x)+', '+inttostr(ltp^[j].y));
writeln(ft);
inc(i);
end
else begin
for j:=1 to rs-3 do if not eof(f) then blockread(f,cf,2);
end;
end;
closefile(f); // закриття файлу
closefile(ft); // закриття файлу
dispose(ltp);
end;

Function TGPMS_MOBILE.ClpWMFRect(fname:string):trect;
var tm:tmetaheader;
f:file;
rs:longint;
cf:word;
i,r,j:word;
s:longint;
tlf:tlogfont;
n,np:word;
ltp:^SPltSmallPoint;
re:trect;
begin //початок
new(ltp);
assignfile(f,fname); // підключення файлу
reset(f,1); // встановлення типу роботи з файлом
blockread(f,tm,sizeof(tm)); // читання даних з фалу
n:=0;
re.left:=32000;re.top:=32000;re.bottom:=-32000;re.right:=-32000;

while not eof(f) do begin
blockread(f,rs,4); // читання даних з фалу
blockread(f,cf,2); // читання даних з фалу
if cf=$0324 then inc(n);
for j:=1 to rs-3 do if not eof(f) then blockread(f,cf,2);
end;
closefile(f); // закриття файлу
if n=0 then exit;

assignfile(f,fname); // підключення файлу
reset(f,1); // встановлення типу роботи з файлом
blockread(f,tm,sizeof(tm)); // читання даних з фалу
I:=1;

while not eof(f) do begin
blockread(f,rs,4); // читання даних з фалу
blockread(f,cf,2); // читання даних з фалу
if ((cf=$0324) and (i<maxelnum)) then begin
blockread(f,np,2); // читання даних з фалу
blockread(f,ltp^,np*sizeof(TSmallPoint));
for j:=1 to np do begin
if ltp^[j].x<re.left then re.left:=ltp^[j].x;
if ltp^[j].y<re.top then re.top:=ltp^[j].y;

```

```

if ltp^[j].x>re.right then re.right:=ltp^[j].x;
if ltp^[j].y>re.bottom then re.bottom:=ltp^[j].y;
end;
inc(i);
end
else begin
    for j:=1 to rs-3 do if not eof(f) then blockread(f,cf,2);
end;
end;
closefile(f); // закриття файлу
dispose(ltp);
ClpWMFRect:=re;
end;
// реалізація внутрішньої процедури класу TGPS_MOBILE ScaleClpWMF2Text
Procedure TGPS_MOBILE.ScaleClpWMF2Text (fname,textfname:string;iso:boolean);
var tm:tmetaheader;
f:file;
ft:textfile;
rs:longint;
cf:word;
i,r,j:word;
s:longint;
tlf:tlogfont;
n,np:word;
ltp:^SPltSmallPoint;
e, re:trect;
sx,sy:real;
lt:integer;
puntos:integer;
a:array[0..255] of char;
ax:string;
begin
e:=rect(0,0,10000,10000);
re:=ClpWMFRect(fname);

if (re.right-re.left)>(re.bottom-re.top) then begin
sx:=(e.right-e.left)/(re.right-re.left);
sy:=(e.bottom-e.top)/(re.right-re.left);
end
else begin
sx:=(e.right-e.left)/(re.bottom-re.top);
sy:=(e.bottom-e.top)/(re.bottom-re.top);
end;

new(ltp);
assignfile(f,fname); // підключення файлу
reset(f,1); // встановлення типу роботи з файлом
blockread(f,tm,sizeof(tm)); // читання даних з фалу
n:=0;
while not eof(f) do begin
blockread(f,rs,4); // читання даних з фалу
blockread(f,cf,2); // читання даних з фалу
if cf=$0324 then inc(n);
if cf=$0325 then inc(n);
for j:=1 to rs-3 do if not eof(f) then blockread(f,cf,2);
end;
closefile(f); // закриття файлу
if n=0 then exit;

assignfile(f,fname); // підключення файлу
reset(f,1); // встановлення типу роботи з файлом

assignfile(ft,textfname); // підключення файлу
rewrite(ft);

blockread(f,tm,sizeof(tm)); // читання даних з фалу
I:=1;
puntos:=0;
while not eof(f) do begin

```

```

blockread(f,rs,4); // читання даних з фалу
blockread(f,cf,2); // читання даних з фалу
if ((cf=$0324) and (i<maxelnum)) then begin
blockread(f,np,2);
blockread(f,ltp^,np*sizeof(TSmallPoint));
writeln(ft,':Element '+inttostr(i));
writeln(ft,'P'+inttostr(i)+' #2');
for j:=1 to np do begin
  ltp^[j].x:=trunc((ltp^[j].x - re.left)*sx + e.left);
  ltp^[j].y:=trunc((ltp^[j].y - re.top)*sy + e.top);
  writeln(ft,inttostr(ltp^[j].x)+', '+inttostr(ltp^[j].y));
end;
writeln(ft);
inc(i);
puntos:=1;
end
else if ((cf=$0325) and (i<maxelnum)) then begin
blockread(f,np,2); // читання даних з фалу
blockread(f,ltp^,np*sizeof(TSmallPoint)); // читання даних з фалу
writeln(ft,':Element '+inttostr(i));
writeln(ft,'_L'+inttostr(i)+' #2');
for j:=1 to np do begin
  ltp^[j].x:=trunc((ltp^[j].x - re.left)*sx + e.left);
  ltp^[j].y:=trunc((ltp^[j].y - re.top)*sy + e.top);
  writeln(ft,inttostr(ltp^[j].x)+', '+inttostr(ltp^[j].y));
end;
writeln(ft);
inc(i);
end

else if ((cf=$0521) and (i<100)) then begin
blockread(f,lt,2); // читання даних з фалу
blockread(f,a,lt); a[lt+1]:=#0; ax:=strpas(a);
writeln(ft,'T'+inttostr(puntos)+' #3');
blockread(f,lt,2);
write(ft,inttostr(lt)+' , ');
blockread(f,lt,2); // читання даних з фалу
writeln(ft,inttostr(lt)+' , '+ax);
inc(puntos);
end
else begin
  for j:=1 to rs-3 do if not eof(f) then blockread(f,cf,2);
end;
end;
closefile(f); // закриття файлу
closefile(ft); // закриття файлу
dispose(ltp);
end;

Procedure TGPS_MOBILE.FromWMF(fname:string);
var tmf:tmetafile;
a:array[0..255] of char;
ax,bx:string;
begin //початок
GetWindowsDirectory(a,200);
ax:=strpas(a)+'\gps_temp.clp';
bx:=strpas(a)+'\gps_temp.txt';
tmf:=tmetafile.create;
tmf.loadfromfile(fname);
clipboard.assign(tmf);
WMF_clip2file(parent.handle,ax);
tmf.destroy;
ScaleClpWMF2Text(ax,bx,true);
FromTextFile(bx);
end;

// реалізація внутрішньої процедури класу TGPS_MOBILE FromClipWMF
Procedure TGPS_MOBILE.FromClipWMF;
var a:array[0..255] of char;

```

```

ax,bx:string;
begin //початок
if not clipboard.hasformat(CF_metafilepict) then exit;
GetWindowsDirectory(a,200);
ax:=strpas(a)+'\gps_temp.clp';
bx:=strpas(a)+'\gps_temp.txt';
WMF_clip2file(parent.handle,ax);
ScaleClpWMF2Text(ax,bx,true);
FromTextFile(bx);
end;

Procedure TGPS_MOBILE.SaveWMF(fname:string);
var tmf:tmetafile;
begin
ClipWMF;
tmf:=tmetafile.create;
tmf.assign(clipboard);
tmf.savetofile(fname);
tmf.destroy;
end;

Procedure TGPS_MOBILE.ClipBmp;
Var //розділ об'яви змінних
r:trect;
bm01:tbitmap;
begin
if not ready then exit;
bm01:=tbitmap.create;
try // спроба з можливим викликом виключення
r:=rect(0,0,width,height);
bm01.width:=(r.right-r.left)+1;
bm01.height:=(r.bottom-r.top)+1;
bm01.canvas.copyrect(r,canvas,r);
clipboard.assign(bm01);
finally // виконати у будь якому випадку навідь при появі виключення
bm01.free;
end;
end;

Procedure TGPS_MOBILE.SaveBmp(fname:string);
var bm:tbitmap;
r:trect;
begin
if not ready then exit;
bm:=tbitmap.create;
r:=rect(0,0,width,height);
bm.width:=(r.right-r.left)+1;
bm.height:=(r.bottom-r.top)+1;
bm.canvas.copyrect(r,canvas,r);
bm.SaveToFile(fname);
bm.free;
end;

Procedure TGPS_MOBILE.SnapShot(destinationcanvas:tcanvas;
                                destrect:trect;maprect:trect);
var r:trect;
begin //початок
if not ready then exit;
drw.convertarect(maprect,r);
StretchBlt(destinationcanvas.handle,
            destrect.left,
            destrect.top,
            destrect.right-destrect.left+1,
            destrect.bottom-destrect.top+1,
            canvas.handle,
            r.left,
            r.top,
            r.right-r.left+1,

```

```

    r.bottom-r.top+1,
    SRCCOPY);

end;

Procedure TGPS_MOBILE.SetChartValue(element,item:word;value:real);
Begin //початок
if element>npolypoints then exit;
if element<1 then exit;
if item>drw.co.nvalues then exit;
np[element].values[item]:=value;
end;

Function  TGPS_MOBILE.GetChartValue(element,item:word):real;
begin
GetChartValue:=0;
if element>npolypoints then exit;
if element<1 then exit;
if item>drw.co.nvalues then exit;
GetChartValue:=np[element].values[item];
end;

// реалізація внутрішньої процедури класу TGPS_MOBILE ValuesToGrid
Procedure  TGPS_MOBILE.ValuesToGrid(grid:tstringgrid);
var i,j:integer;
ax:string;
begin
with grid do begin
colcount:=drw.co.nvalues+2;
rowcount:=npolypoints+1;
fixedcols:=0;
fixedrows:=0;
cells[0,0]:='';
cells[1,0]:='Bullet';
for i:=1 to drw.co.nvalues do cells[1+i,0]:=drw.co.LegTexts[i];
for i:=1 to npolypoints do cells[0,i]:=np[i].nombre;
for i:=1 to npolypoints do
cells[1,i]:=fcadd(np[i].values[0],3,drw.co.ndecimals);
for i:=1 to drw.co.nvalues do
for j:=1 to npolypoints do
cells[i+1,j]:=fcadd(np[j].values[i],3,drw.co.ndecimals);
end;
end;

Procedure  TGPS_MOBILE.ValuesFromGrid(grid:tstringgrid);
var i,j:integer;
ax:string;
begin
with grid do begin
drw.co.nvalues:=colcount-2;
for i:=0 to drw.co.nvalues do drw.co.LegTexts[i]:=cells[1+i,0];
for i:=1 to rowcount-1 do np[i].nombre:=cells[0,i];
for i:=1 to rowcount-1 do np[i].values[0]:=fnum(cells[1,i]);
for i:=1 to drw.co.nvalues do
for j:=1 to rowcount-1 do
np[j].values[i]:=fnum(cells[i+1,j]);
end;
end;
CalcChartRatios;
end;

// реалізація внутрішньої процедури класу TGPS_MOBILE ImportDataFromGrid
Procedure  TGPS_MOBILE.ImportDataFromGrid(grid:tstringgrid);
var i,j,k,code:integer;
r:real;
ax:string;
begin
if grid.colcount>18 then grid.colcount:=18;
drw.co.nvalues:=grid.colcount-2;
for i:=1 to grid.rowcount-1 do begin

```

```

ax:=uppercase(grid.cells[0,i]);
for j:=1 to ElementCount do
if uppercase(ElementName[j])=ax then
for k:=1 to grid.colcount-1 do begin
val(grid.cells[k,i],r,code);
chartvalue[j,k-1]:=r;
end;
end;
for i:=1 to grid.colcount-1 do chartlegend[i-1]:=grid.cells[i,0];
end;

Function  TGPS_MOBILE.GetPatternBits(i:word):string;
Begin //початок
if i>8 then exit;
if i=0 then GetPatternBits:='000000'
else GetPatternBits:=trama[i];
end;

Procedure TGPS_MOBILE.SetPatternBits(i:word;s:string);
begin
if ((i<1) or (i>8)) then exit;
trama[i]:=copy(s,1,64);
end;

Procedure TGPS_MOBILE.Bar(h:hdc;x1,y1,x2,y2:integer;pat:word);
begin
if pat>16 then exit;if pat<1 then exit;
drw.GetBrocha(h,drw.co.patt[pat],drw.co.lineGPS,drw.co.GPSs[pat]);
rectangle(h,x1,y1,x2,y2);
drw.ReleaseBrocha;
end;

Function TGPS_MOBILE.GetPolLayer(element,pol:word):byte;
Begin //початок
GetPolLayer:=1;
if element=0 then exit; if pol=0 then exit;
if (element>npolypoints) then exit;
if (pol>np[element].npolis) then exit;
GetPolLayer:=np[element].p[pol].layer;
end;

Procedure TGPS_MOBILE.SetPolLayer(element,pol:word;l:byte);
begin
if element=0 then exit; if pol=0 then exit;
if (element>npolypoints) then exit;
if (pol>np[element].npolis) then exit;
np[element].p[pol].layer:=l;
end;

Function TGPS_MOBILE.GetPolCount(element:word):byte;
begin
GetPolCount:=0;
if element=0 then getpolcount:=npolypoints;
if (element>npolypoints) then exit;
GetPolCount:=np[element].npolis;
end;

Function TGPS_MOBILE.GetPolName(element,pol:word):string;
begin
GetPolName:='';
if element=0 then exit; if pol=0 then exit;
if (element>npolypoints) then exit;
if (pol>np[element].npolis) then exit;
GetPolName:=np[element].p[pol].nombre;
end;

Procedure TGPS_MOBILE.SetPolName(element,pol:word;s:string);
begin
if element=0 then exit; if pol=0 then exit;

```

```

if (element>npolypoints) then exit;
if (pol>np[element].npolis) then exit;
np[element].p[pol].nombre:=s;
end;

Procedure TGPS_MOBILE.ShowPolNames (b:boolean);
begin
drw.showpols:=b;
end;

Function TGPS_MOBILE.GetDistance (p1,p2:tpoint):double;
begin
GetDistance:=distancia (p1.x,p1.y,p2.x,p2.y)*drw.mo.scale;
end;

Procedure TGPS_MOBILE.Setnumbox (b:boolean);
begin
drw.numbox:=b;
end;
Function TGPS_MOBILE.GetNumBox:boolean;
begin
GetNumBox:=drw.numbox;
end;

Function TGPS_MOBILE.ElementbyName (s:string):integer;
var i,k:integer;
begin
k:=1;
for i:=1 to elementcount do
if pos (uppercase (s),uppercase (elementname[i]))>0 then begin
k:=i;
break;
end;
ElementbyName:=k;
end;

// реалізація внутрішньої процедури класу TGPS_MOBILE Nearest
Function TGPS_MOBILE.Nearest (p:tpoint):integer;
var i,j,k:integer;
mybox:trect;
begin //початок
k:=0;
Nearest:=0;
for i:=npolypoints downto 1 do
if np[i].npolis>0 then begin
if ((p.x>np[i].caja.left) and (p.x<np[i].caja.right)
and (p.y>np[i].caja.top) and (p.y<np[i].caja.bottom))
then begin Nearest:=i;exit;end;
end;
Nearest:=k;
end;

Function TGPS_MOBILE.NearestElement (p:tpoint):integer;
var i,j,k:integer;
mybox:trect;
begin
k:=0;
NearestElement:=0;
for i:=npolypoints downto 1 do if sel[i] then
if np[i].npolis>0 then begin
if ((p.x>np[i].caja.left) and (p.x<np[i].caja.right)
and (p.y>np[i].caja.top) and (p.y<np[i].caja.bottom))
then begin NearestElement:=i;exit;end;
end;
NearestElement:=k;
end;

procedure muerte;

```

```

begin
showmessage('Fatal error reading '+fname);
halt;
end;
var i,j,k,im,ip,ipu:integer;
ax,bx,nom:string;
fin:boolean;
p:tpoint;
f:textfile;
nsp:^tnsp;
n:word;
buffer:^SPltpoint;
maxcaja:trect;
c:char;
thelayer:integer;
lm:longint;
begin
reset(f); // встановлення типу роботи з файлом
ax:='';
for i:=1 to npolypoints do begin
while ((ax[1]<>':') and (not eof(f))) do begin
readln(f,ax);
ax:=ltrim(ax);
end;
if eof(f) then muerte;
infopanel.caption:=inttostr(i)+' of '+ inttostr(npolypoints);
infopanel.repaint;
ax:=rltrim(derc(ax,':'));
np[i]:=TSPolyPol.create(nsp^[i].m,nsp^[i].p,ax);
np[i].npoints:=nsp^[i].pu;
im:=0;ip:=0; ipu:=0;
for j:=1 to nsp^[i].m + nsp^[i].p + nsp^[i].pu do begin
repeat
readln(f,ax);
ax:=ltrim(ax);
until ((ax[1] in ['_','+', '/' ]) or eof(F));
if eof(f) then muerte;
c:=ax[1];
nom:=rltrim(izqc(derc(ax,c), '#'));
ax:=rltrim(derc(ax, '#'));
if ax='' then ax:='1';
thelayer:=num(ax);
if c='_ ' then begin
inc(im);
np[i].SetMVals(clwhite,clblack,im,k,buffer^);
np[i].m[im].layer:=thelayer;
np[i].m[im].drawer:=drw;
np[i].m[im].nombre:=nom;
end
else
if c='+' then begin
inc(ip);
np[i].SetPVals(clwhite,clblack,ip,k,buffer^);
np[i].p[ip].layer:=thelayer;
np[i].p[ip].drawer:=drw;
np[i].p[ip].nombre:=nom;
end;
end;
np[i].CalcGPS;
comprect(np[i].caja,maxcaja);
np[i].drawer:=drw;
end;
closefile(f); // закриття файлу
dispose(nsp);
dispose(buffer);
for i:=1 to npolypoints do sel[i]:=true;
if arr<>nil then arr.destroy;
arr:=TSArrange.create(2, npolypoints, 'Normal');
ready:=true;

```

```

Titulo:=fname;
Nbloqs:=npolypoints;
MapLimits:=MaxCaja;
end;

// додавання точки на карті
Function TGPS_MOBILE.AddPoint(elem:integer;c:tcitypoint):integer;
var a:tsapoint;
begin //початок
AddPoint:=0;
if elem>npolypoints then exit; if elem<1 then exit;
if np[elem].npoints>15 then exit;
a:=tsapoint.create;
a.c:=c;
np[elem].npoints:=np[elem].npoints+1;
np[elem].points[np[elem].npoints]:=a;
AddPoint:=np[elem].npoints;
a.free;
end;

Procedure TGPS_MOBILE.GetPoint(elem,ix:integer;var c:tcitypoint);
begin
if elem>npolypoints then exit; if elem<1 then exit;
if ix<1 then exit;if ix> np[elem].npoints then exit;
c:=np[elem].points[ix].c;
end;

Function TGPS_MOBILE.RemovePoint(elem,ix:integer):boolean;
var i:integer;
begin
removepoint:=false;
if elem>npolypoints then exit; if elem<1 then exit;
if np[elem].npoints<1 then exit;
if ix<1 then exit;if ix> np[elem].npoints then exit;
for i:=ix to np[elem].npoints-1 do np[elem].points[i]:=np[elem].points[i+1];
np[elem].npoints:=np[elem].npoints-1;
Removepoint:=true;
end;

Function TGPS_MOBILE.GetPointCount(elem:integer):integer;
Begin //початок
GetPointCount:=0;
if elem>npolypoints then exit; if elem<1 then exit;
if np[elem].npoints<1 then exit;
GetPointCount:=np[elem].npoints;
end;

Procedure TGPS_MOBILE.ClearPoints(elem:integer);
begin
if elem>npolypoints then exit; if elem<1 then exit;
np[elem].npoints:=0;
end;

Procedure TGPS_MOBILE.SetSymbol(f:tfont;c:char);
begin
drw.fontsymbol.assign(f);
drw.asymbol:=c;
end;

procedure TGPS_MOBILE.time(sender:tobject);
begin
replaceglis;
gli.paint(canvas);
end;

Procedure TGPS_MOBILE.MyMouseUp(Sender: TObject; Button: TMouseButton;Shift:
TShiftState; X, Y: Integer);
var p:tpoint;
i,k:integer;

```

```

pu:tpoint;
pr:trect;
begin //початок
case drw.mo.mousemode of
8:begin
    p.x:=x;p.y:=y;
    p:=PointInMap(p);
    if button=mbright then begin
        p.x:=(gpsbox.right+gpsbox.left) div 2;
        p.y:=(gpsbox.top+gpsbox.bottom) div 2;
    end;
    k:=(zoombox.right-zoombox.left) div 2;
    i:=(zoombox.bottom-zoombox.top) div 2;
    fzoombox.left:=p.x-k; fzoombox.right:=p.x+k;
    fzoombox.top:=p.y-i; fzoombox.bottom:=p.y+i;
    repaint;
end;

10:if bb_drawing then begin
    bb_drawing:=false;
    DrawFocusRect(canvas.handle,rect(bb_anchorx,bb_anchory,bb_curx,bb_cury));
    p:=PointInMap(point(bb_anchorx,bb_anchory));
    if button=mbright then SelFromBox(gpsbox,true)
    else
        if ((bb_curx-bb_anchorx>10) and (bb_cury-bb_anchory>10)) then
            begin
                pr.left:=p.x;pr.top:=p.y;
                p:=PointInMap(point(bb_curx,bb_cury));
                pr.right:=p.x;pr.bottom:=p.y;
                if ssCtrl in Shift then SelFromBox(pr,false)
                else SelFromBox(pr,true);
            end
            else begin
                k:=Nearest(p);
                if k>0 then begin
                    if ssCtrl in Shift then selected[k]:=true
                    else selected[k]:= not selected[k];
                end;
            end;
            invalidate;
        end;

1,9,12,15:if bb_drawing then begin
    bb_drawing:=false;
    DrawFocusRect(canvas.handle,rect(bb_anchorx,bb_anchory,bb_curx,bb_cury));

    if drw.mo.mousemode =9 then begin
        pr.left:=trunc((bb_anchorx-left)/(drw._back.right-drw._back.left)*100);
        pr.right:=trunc((bb_curx-left)/(drw._back.right-drw._back.left)*100);
        pr.top:=trunc((bb_anchory-top)/(drw._back.bottom-drw._back.top)*100);
        pr.bottom:=trunc((bb_cury-top)/(drw._back.bottom-drw._back.top)*100);
    end;

    case drw.mo.mousemode of
    9:
        if ((bb_curx-bb_anchorx>10) and (bb_cury-bb_anchory>10)) then
            begin
                viewpct:=pr;
                infopanel.caption:=inttostr(viewpct.left)+' ','+
                inttostr(viewpct.top)+' - '+
                inttostr(viewpct.right)+' ','+
                inttostr(viewpct.bottom);

                end
            else viewpct:=rect(1,1,99,99);

    1:
        if ((bb_curx-bb_anchorx>10) and (bb_cury-bb_anchory>10)) then
            begin

```

```

    p:=PointInMap(point(bb_anchorx,bb_anchory));
    fzoombox.left:=p.x;fzoombox.top:=p.y;
    p:=PointInMap(point(bb_curx,bb_cury));
    fzoombox.right:=p.x;fzoombox.bottom:=p.y;
end
else

begin          //початок
p.x:=x;p.y:=y;
p:=PointInMap(p);
k:=(zoombox.right-zoombox.left) div 2;
i:=(zoombox.bottom-zoombox.top) div 2;
fzoombox.left:=p.x-k; fzoombox.right:=p.x+k;
fzoombox.top:=p.y-i; fzoombox.bottom:=p.y+i;
end;

12:if Assigned(FOnMouseBox) then begin
    if ((bb_curx-bb_anchorx>10) and (bb_cury-bb_anchory>10)) then
        begin
            p:=PointInMap(point(bb_anchorx,bb_anchory));
            pr.left:=p.x;pr.top:=p.y;
            p:=PointInMap(point(bb_curx,bb_cury));
            pr.right:=p.x;pr.bottom:=p.y;
            end
        else pr:=gpsbox;
            OnMouseBox(self,shift,pr);
        end;
15:begin
    if ((bb_curx-bb_anchorx>20) and (bb_cury-bb_anchory>20)) then
        begin
            p:=PointInMap(point(bb_anchorx,bb_anchory));
            pr.left:=p.x;pr.top:=p.y;
            p:=PointInMap(point(bb_curx,bb_cury));
            pr.right:=p.x;pr.bottom:=p.y;
            end
        else pr:=gpsbox;
            scaleto(pr);
        end;
    end;
    invalidate;
end;
3:begin
    bb_drawing:=false;
    invalidate;
end;
4,11:if bb_drawing then begin
    bb_drawing:=false;
    SetRop2(canvas.handle,r2_not);
    movetoex(canvas.handle,drw.cvx(bb_anchorx),drw.cvy(bb_anchory),nil);
    lineto(canvas.handle,drw.cvx(bb_curx),drw.cvy(bb_cury));
    if drw.mo.mousemode=11 then if assigned(FOnMouseLine)
        then OnMouseLine(self,shift,rect(bb_anchorx,bb_anchory,bb_curx,bb_cury));
    end;
5:if bb_drawing then begin
    bb_drawing:=false;
    SetRop2(canvas.handle,r2_not);
    p.x:=abs(bb_curx-bb_anchorx);
    p.y:=p.x;

    k:=abs(drw.cvx(bb_curx)-drw.cvx(bb_anchorx));
    pu.x:=drw.cvx(bb_anchorx);
    pu.y:=drw.cvy(bb_anchory);
    Arc(canvas.handle,pu.x-k,pu.y-k,
        pu.x+k,pu.y+k,
        pu.x,pu.y,pu.x,pu.y);
end;
16: if not userline.drawing then begin
    if button=mbright then invalidate;

```

```

    end;
end;
end;

// реалізація внутрішньої процедури класу TGPS_MOBILE WMLButtonDown
Procedure TGPS_MOBILE.WMLButtonDown(var msg:TWMLButtonDown);
Begin //початок
if not ready then exit;
MyMouseDown(self,mbleft,KeysToShiftState(msg.keys),msg.xpos,msg.ypos);
If Assigned(FOnMouseDown) then
OnMouseDown(self,mbleft,KeysToShiftState(msg.keys),msg.xpos,msg.ypos);
end;

Procedure TGPS_MOBILE.WMRButtonDown(var msg:TWMRButtonDown);
begin
if ready then begin
MyMouseDown(self,mbright,KeysToShiftState(msg.keys),msg.xpos,msg.ypos);
If Assigned(FOnMouseDown) then
OnMouseDown(self,mbright,KeysToShiftState(msg.keys),msg.xpos,msg.ypos)
else if drw.mo.mousemode=0 then parent.dispatch(msg);
end
else inherited;
end;

Procedure TGPS_MOBILE.DeleteElements;
var i,j,k:integer;
ts:tspolypol;
begin
k:=npolypoints;
for j:=k downto 1 do if selected[j] then begin
if j<npolypoints then begin
ts:=np[j];
np[j]:=np[npolypoints];
selected[j]:=false;
ts.free;
end
else begin
np[npolypoints].free;
end;
npolypoints:=npolypoints -1;
end;
end;

Procedure TGPS_MOBILE.JoinElements(orig,dest:integer);
var i,j,k:integer;
ts:tspolypol;
begin
if orig<npolypoints then begin
ts:=np[orig];
np[orig]:=np[npolypoints];
ts.free;
end
else begin
np[npolypoints].free;
end;
npolypoints:=npolypoints -1;

end;

// реалізація внутрішньої процедури класу TGPS_MOBILE posterxN
procedure TGPS_MOBILE.posterxN(Nx,Ny:integer);
VAR //РОЗДІЛ ОБ'ЯВИ ЗМІННИХ
P: TPosterPrint;
begin
if not ready then exit;
if (( nx<1) or (ny<1) or (nx>10) or (ny>10)) then exit;
screen.cursor:=crhourglass;
p:=TPosterPrint.create(self);
try // спроба з можливим викликом виключення

```

```

with p do begin
    HorzPages:=nx;
    VertPages:=ny;
    HorzMarge:=100;
    VertMarge:=100;
    Image:='gps_temp.wmf';
    LabelPages:=true;
end;
SaveWMF('gps_temp.wmf');
p.doprint;
finally // виконати у будь якому випадку навідь при появі виключення
p.free;
screen.cursor:=crdefault;
end;
end;

Function TGPS_MOBILE.GetArea(elem:integer):longint;
begin
GetArea:=trunc(np[elem].GetArea(drw.mo.scale,drw.mo.scale));
end;

Procedure TGPS_MOBILE.LoadPaletteFromFile(filename:string);
begin
deleteobject(paleta);
paleta:=CreatePaletteFromFile(filename);
end;

Procedure TGPS_MOBILE.LoadMosaicBitmap(fname:string);
begin
drw.mosaicbmp.loadfromfile(fname);
end;
// реалізація внутрішньої процедури класу TGPS_MOBILE placegli
procedure TGPS_MOBILE.placegli(ix:integer;p:tpoint);
var pp:tpoint;
begin
if ((ix>0) and (ix<=gli.count)) then begin
gli.glyph[ix].mappos:=p;
drw.convertapoint(p,pp);
gli.glyph[ix].pos:=pp;
gli.glyph[ix].paint(canvas);
end;
end;

procedure TGPS_MOBILE.addgli(p:tpoint);
begin
gli.add(0);
placegli(gli.count,p);
end;

procedure TGPS_MOBILE.replaceglis;
var i:integer;
p,pp:tpoint;
begin
for i:=gli.count downto 1 do begin
drw.convertapoint(gli.glyph[i].mappos,p);
gli.glyph[i].pos:=p;
end;
end;
// реалізація внутрішньої процедури класу TGPS_MOBILE GlyphCount
function TGPS_MOBILE.GlyphCount:integer;
begin
GlyphCount:=gli.count;
end;
// реалізація внутрішньої процедури класу TGPS_MOBILE GlyphSettings
Procedure TGPS_MOBILE.GlyphSettings(ixglyph:integer;imageix:integer;
animate:boolean;enable:boolean);
begin
if gli.glyph[ixglyph]<>nil then with gli.glyph[ixglyph] do begin
if imageix<gli.gn then index:=imageix;

```

```

animated:=animate;
enabled:=enable;
end;
end;
// реалізація внутрішньої процедури класу TGPS_MOBILE LoadGlyphsBitmap
Procedure TGPS_MOBILE.LoadGlyphsBitmap (fname:string);
begin
gli.Loadfromfile (fname);
end;

Procedure TGPS_MOBILE.WMLButtonUp (var msg:TWMLButtonUp);
begin
if not ready then exit;
MyMouseUp (self,mbleft,KeysToShiftState (msg.keys),msg.xpos,msg.ypos);
if ready then If Assigned (fOnMouseUp) then
OnMouseUp (self,mbleft,KeysToShiftState (msg.keys),msg.xpos,msg.ypos);
inherited;
end;

Procedure TGPS_MOBILE.WMRButtonUp (var msg:TWMRButtonUp);
begin
if ready then begin
MyMouseUp (self,mbright,KeysToShiftState (msg.keys),msg.xpos,msg.ypos);
if ready then If Assigned (fOnMouseUp) then
OnMouseUp (self,mbright,KeysToShiftState (msg.keys),msg.xpos,msg.ypos);
end;
end;

Procedure TGPS_MOBILE.WMMouseMove (var msg:TWMMouseMove);
begin
if not ready then exit;
MyMouseMove (self,KeysToShiftState (msg.keys),msg.xpos,msg.ypos);
if ready then if Assigned (fOnMouseMove) then
OnMouseMove (self,KeysToShiftState (msg.keys),msg.xpos,msg.ypos);
end;

Procedure TGPS_MOBILE.ToFront (elem:integer);
var auxp:TSPolyPol;
i:integer;
auxs:boolean;
begin
if ((elem<1) or (elem>npolypoints)) then exit;
auxp:=np[elem];
auxs:=selected[elem];
for i:=elem to npolypoints-1 do begin
np[i]:=np[i+1];
selected[i]:=selected[i+1];
end;
np[npolypoints]:=auxp;
selected[npolypoints]:=auxs;
end;

// реалізація внутрішньої процедури класу TGPS_MOBILE ToBack
Procedure TGPS_MOBILE.ToBack (elem:integer);
var auxp:TSPolyPol;
i:integer;
auxs:boolean;
begin
if ((elem<1) or (elem>npolypoints)) then exit;
auxp:=np[elem];
auxs:=selected[elem];
for i:=elem downto 2 do begin
np[i]:=np[i-1];
selected[i]:=selected[i-1];
end;
np[1]:=auxp;
selected[1]:=auxs;
end;

```

```

Procedure TGPS_MOBILE.ReMap(var m:array of integer);
var sp:^TSPOLARRAY;
i:integer;
begin
new(sp);
for i:=1 to npolypoints do sp^[i]:=np[m[i]];
for i:=1 to npolypoints do np[i]:=sp^[i];
dispose(sp);
end;

Function  TGPS_MOBILE.IsInBox(ix:integer;r:trect):boolean;
var rr:trect;
begin
IntersectRect(rr,r,np[ix].caja);
IsInBox:=((rr.left<>0) or (rr.right<>0) or (rr.bottom<>0) or (rr.top<>0));
end;

// реалізація внутрішньої процедури класу TGPS_MOBILE SelFromBox
Procedure TGPS_MOBILE.SelFromBox(r:trect;cleanup:boolean);
var i:integer;
begin
if cleanup then begin
for i:=1 to elementcount do selected[i]:= IsInBox(i,r);
end
else
for i:=1 to elementcount do if IsInBox(i,r) then selected[i]:= true;
end;

Procedure TGPS_MOBILE.ForwardLink(ix:integer);
var ax:string;
begin
if ix>0 then
if np[ix].link<>' ' then
if fileexists(np[ix].link) then begin
if nbacks<5 then inc(nbacks);
BackLink[nbacks]:=polfname;
ax:=np[ix].link;
FileName:=ax;
drw.mo.mousemode:=14;
end;
end;

// реалізація внутрішньої процедури класу TGPS_MOBILE BackwardLink
Procedure TGPS_MOBILE.BackwardLink;
var ax:string;
begin
if nbacks>0 then begin
if fileexists(BackLink[nbacks]) then begin
ax:=BackLink[nbacks];
FileName:=ax;
drw.mo.mousemode:=14;
end;
end;
dec(nbacks);
end;
end;

Procedure TGPS_MOBILE.SetBackgroundOptions(var op:tbkop);
begin
drw.bko:=op;
if drw.bko.tilefname<>' ' then if fileexists(drw.bko.tilefname)
then backbitmap.loadfromfile(drw.bko.tilefname);
if drw.bko.logofname<>' ' then if fileexists(drw.bko.logofname)
then logobitmap.loadfromfile(drw.bko.logofname);
end;

Procedure TGPS_MOBILE.CalcGPS;
var i:integer;
begin;
prepect(MapLimits);

```

```

for i:=1 to npolypoints do comprect(np[i].caja,MapLimits);
end;

// реалізація внутрішньої процедури класу TGPS_MOBILE Merge
Procedure TGPS_MOBILE.Merge(fname:string;layeroffset:integer; x,y:integer);
var agps:TGPS;
i,j,k:integer;
begin
if not fileexists(fname) then exit;
ready:=false;
agps:=TGPS_MOBILE.create(self);
agps.filename:=fname;
for i:=1 to agps.elementcount do
np[elementcount+i]:=agps.np[i];
npolypoints:=npolypoints+agps.elementcount;
agps.npolypoints:=0;
agps.free;
CalcGPS;
ready:=true;
zoombox:=MapLimits;
end;
Procedure TGPS_MOBILE.ScaleTo(r:trect);
var sb,aux:trect;
kx,ky:double;
x0,y0,x1,y1:integer;
i,j:integer;
function newx(x:integer):integer;
begin
newx:=trunc(x1+(x-x0)*kx);
end;
function newy(y:integer):integer;
begin
newy:=trunc(y1+(y-y0)*ky);
end;
function newrect(t:trect):trect;
var tt:trect;
begin
tt.left:=newx(t.left);
tt.right:=newx(t.right);
tt.top:=newy(t.top);
tt.bottom:=newy(t.bottom);
newrect:=tt;
end;
begin
for i:=1 to elementcount do if selected[i] then
np[i].scaleto(newrect(np[i].caja));
end;
// реалізація внутрішньої процедури класу TGPS_MOBILE AddPoly
Procedure TGPS_MOBILE.AddPoly(closed:boolean);
var i,j:integer;
var slt:^SPltpoint;
begin
if userline.np<2 then exit;
npolypoints:=npolypoints+1;
np[npolypoints]:=TSPolyPol.create(1,0,'New');
slt:=addr(userline.p);
with np[npolypoints] do begin
SetMVals(clred,clblue,1,userline.np,slt^);
m[1].drawer:=drw;
m[1].layer:=2;
m[1].nombre:='Line 1';
CalcGPS;
drawer:=drw;
end;
end;
CalcGPS;
selected[npolypoints]:=true;
end;
end.

```

ОСНОВНА ПРОГРАМА, GPS_1.DPR ОСНОВНОЇ ПРОГРАМИ

```

program GPS_1;
{
  Програмне забезпечення системи кібербезпеки для стеження за мобільним
  пристроєм з використанням GPS
  Дубіцька Марина Миколаївна
  гр. КБ-21-ЗСК, 2024
}
Uses // об'ява бібліотек
  Forms, // бібліотека форм ОС Windows
  GPS_About in 'GPS_About.pas' {F2}, // Форма авторського права
  GPS_MainForm in 'GPS_MainForm.pas' {F1}, // Основна форма
  GPS in 'GPS.pas'; // Отримання даних
{$R *.RES}

// процедура отримання піддиректорій
procedure GetSubDirs (Folder: string; sList: TStringList);
  Var //розділ об'яви змінних
    sr: TSearchRec; // структура
  begin //початок
    if FindFirst (Folder + '.*', faDirectory, sr) = 0 then
      try // спроба з можливим викликом виключення
        repeat
          if (sr.Attr and faDirectory) = faDirectory then
            sList.Add (sr.Name);
          until FindNext(sr) 0;
        finally // виконати у будь якому випадку навідь при появі виключення
          FindClose(sr);
        end;
      end;
    end;
end;

//пошук файлів
function GetAllFiles (Filter, Folder: string):TFilesList;
  Var //розділ об'яви змінних
    sr: TSearchRec;
    sDirList, _FilesList, _LST: TStringList;
    i, j: Integer;
  begin
    j:=0;
    _FilesList:= TStringList.create;
    _FilesList.Clear;
    if FindFirst (Folder + Filter, faAnyFile , sr) = 0 then
      repeat
        j:=j+1;
        if j 0;
        FindClose(sr);
        sDirList := TStringList.Create;
        try // спроба з можливим викликом виключення
          GetSubDirs (Folder, sDirList);
          for i := 0 to sDirList.Count - 1 do
            if (sDirList[i] '.') and (sDirList[i] '..') then
begin _LST:=GetAllFiles(Filter, IncludeTrailingPathDelimiter(
  Folder + sDirList[i]));
          for j:=0 to _LST.Count-1 do
            _FilesList.Add(_LST.Strings[j]);
            _LST.free;
          end;
        finally // виконати у будь якому випадку навідь при появі виключення
          sDirList.Free;
        end;
      end;
    Result:= _FilesList;
  end;

function DoFolderAction(FolderPath, ArchivePath:string;
  ActionFunction :TActionFuntion):boolean;
  Var //розділ об'яви змінних
    ArchiveFile, CurrFile :THandle;

```

```

Value,i,CurrFileSize,_readed,_writed:DWORD;
_Files:TStringList;
path_in_archive_file :string;
pBuff:Pointer;
begin
Result:=false;
if FolderPath[Length(FolderPath)]!='\' then
FolderPath:=FolderPath+'\' ;
_Files:=GetAllFiles('*.*',FolderPath);
_ArchiveFile := CreateFile(pchar(ArchivePath),
GENERIC_READ+GENERIC_WRITE,FILE_SHARE_READ,0,
CREATE_ALWAYS,0,0);
Value:=ArchiveSignature;
WriteFile(ArchiveFile,value,4,_writed,0);
Value:=_Files.Count;
WriteFile(ArchiveFile,value,4,_writed,0);
for i:=0 to value-1 do
begin
if not ActionFunction(_Files.Strings[i],bkpFile) then
begin
CloseHandle(ArchiveFile);
exit;
end;
CurrFile := CreateFile(pchar(bkpFile),GENERIC_READ,
FILE_SHARE_READ,0,OPEN_EXISTING,0,0);
CurrFileSize := GetFileSize(CurrFile,nil);
if path_in_archive_file[1] = '\' then
Delete(path_in_archive_file,1,1);
Write_LS(ArchiveFile,path_in_archive_file);
WriteFile(ArchiveFile,currFileSize,4,_writed,0);
pBuff:=VirtualAlloc(0,MainBufferSize,
MEM_COMMIT+MEM_RESERVE,PAGE_READWRITE);
repeat
ReadFile(CurrFile,pBuff^,MainBufferSize,_readed,0);
WriteFile(ArchiveFile,pBuff^,_readed,_writed,0);
until _writed < MainBufferSize;
VirtualFree(pBuff,MEM_RELEASE,0);
CloseHandle(CurrFile);
DeleteFile(bkpFile);
end;
CloseHandle(ArchiveFile);
Result:=True;
end;

begin // Початок основного потоку ПЗ
Application.Initialize; // Ініціалізація
Application.CreateForm(TF1, F1); // Створення форми
Application.Run;//Запуск ПЗ
end.

```

ФАЙЛ АВТОРСЬКОГО ПРАВА ABOUT.PAS

```
unit About; // Форма авторського права
{
  Програмне забезпечення системи кібербезпеки для стеження за мобільним
  пристроєм з використанням GPS
  Дубіцька Марина Миколаївна
  гр. КБ-21-ЗСК, 2024
}
Interface // Інтерфейсна частина - об'ява даних

// об'ява бібліотек
WinTypes, WinProcs, Classes, Graphics, Forms, Controls, StdCtrls,
  Buttons, ExtCtrls;

Type // об'ява власних типів даних
TAboutBox = class(TForm) // розроблений клас
  Panel1: TPanel;
  OKButton: TBitBtn;
  ProgramIcon: TImage;
  ProductName: TLabel;
  Version: TLabel;
  Copyright: TLabel;
  Label1: TLabel;
  procedure OKButtonClick(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

Var //розділ об'яви змінних
  AboutBox: TAboutBox;

Implementation // Реалізація об'явлених даних у інтерфейсній частині

{$R *.DFM}

procedure TAboutBox.OKButtonClick(Sender: TObject);
begin
  close;
end;

end.
```