

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

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

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
за першим (бакалаврським) рівнем вищої освіти
на тему
“Програмне забезпечення системи сервісів визначення місця
розташування OmniAccess Stellar”

Виконав здобувач вищої освіти
IV курсу, групи КІ-22-2
ОПП «Комп’ютерна інженерія»
спеціальності 123 «Комп’ютерна інженерія»
_____ Жупанов Н.О.
«___» _____ 2026 р.

Керівник проекту
кандидат технічних наук, доцент
_____ Буравченко К.О.
«___» _____ 2026 р.
Рецензент _____

Центральноукраїнський національний технічний університет
Факультет Механіко-технологічний
Кафедра Кібербезпеки та програмного забезпечення
Освітній ступінь бакалавр
Галузь знань 12 "Інформаційні технології"
Спеціальність 123 "Комп'ютерна інженерія"
Освітньо-професійна (освітньо-наукова) програма "Комп'ютерна інженерія"

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

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

Жупанову Назару Олександровичу

(прізвище, ім'я, по батькові)

1. Тема роботи Програмне забезпечення системи сервісів визначення
місця розташування OmniAccess Stellar

2. Керівник роботи Буравченко Костянтин Олегович, канд. техн. наук, доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу № 136-02 від 27.02.2026 року

3. Строк подання студентом роботи до захисту 23.05.2026 р.

4. Мета та завдання випускної кваліфікаційної роботи: Метою роботи є розробка
програмного забезпечення системи сервісів визначення місця розташування
OmniAccess Stellar

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

1. Призначення та область використання.

2. Перегляд аналогічних існуючих систем.

3. Опис і обґрунтування проектних рішень.

4. Етапи програмування системи.

5. Впровадження системи в промислову експлуатацію.

6. Висновки

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

Структурна схема системи 1 аркуш

Функціональна схема системи 1 аркуш

Діаграма процесів 1 аркуш

Блок-схема алгоритму роботи додатку 2 аркуша

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

КАЛЕНДАРНИЙ ПЛАН

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

Дата видачі завдання
« 27 » лютого 2026 р.

Підпис керівника

Буравченко К.О.
(прізвище та ініціали)

Завдання прийнято до виконання
« 27 » лютого 2026 р.

Підпис здобувача

Жупанов Н.О.
(прізвище та ініціали)

АНОТАЦІЯ

Жупанов Н.О. Програмне забезпечення системи сервісів визначення місця розташування OmniAccess Stellar. 123 Комп'ютерна інженерія. Центральноукраїнський національний технічний університет. Кропивницький. 2026.

В даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи сервісів визначення місця розташування OmniAccess Stellar.

Метою розробки є програмне забезпечення системи сервісів визначення місця розташування OmniAccess Stellar.

Результат роботи – програмна реалізація системи сервісів визначення місця розташування OmniAccess Stellar.

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

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

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

Програму розроблено в середовищі Python.

Ключові слова: комп'ютерна інженерія, OmniAccess Stellar

ABSTRACT

Zhupanov N.O. Software for the OmniAccess Stellar location services system. 123 Computer Engineering. Central Ukrainian National Technical University. Kropyvnytskyi. 2026.

In this final qualification work for the first (bachelor's) level of higher education, software has been developed that is intended for the OmniAccess Stellar location services system.

The purpose of the development is the software for the OmniAccess Stellar location services system.

The result of the work is the software implementation of the OmniAccess Stellar location services system.

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

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

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

The program was developed in the Python environment.

Keywords: computer engineering, OmniAccess Stellar

3MIST

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	13
2.1 Огляд існуючих систем, технологій, архітектур та програмних рішень за профілем теми випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти.....	13
2.2 Обґрунтування вибору засобів для побудови системи та мови програмування.....	19
2.3 Розгорнута постановка завдання	24
3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ	26
3.1 Опис функціонування системи	26
3.2 Розробка структурної схеми.....	30
3.3 Розробка функціональної схеми	35
3.4 Розробка діаграми процесів.....	38
4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ.....	40
4.1 Розробка блок-схем та опис алгоритмів функціонування системи.....	40
4.2 Захист розробленого програмного забезпечення.....	70
5 ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ	72
6 ОСНОВНІ ВИСНОВКИ.....	78
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	80

					ВКРБ-123.26.0043.00.00.ПЗ				
Вим	Арк.	№ докум.	Підп.	Дата	Програмне забезпечення системи сервісів визначення місця розташування OmniAccess Stellar	Літ.	Аркуш	Аркушів	
Розроб.		Жупанов Н.О.				Б		1	87
Перев.		Буравченко К.О.							
Н.контр.		Коваленко А.С.							
Затв.		Смірнов О.А.							

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

КПК – кишеньковий персональний комп'ютер

РМД – робоче місце диспетчера

СЕР – Circular Error of Probability – окружність можливої помилки

GPS – глобальна система місцевизначення.

GSM – система цифрового стільникового телефонного зв'язку

IOD – issue of data – часовий ідентифікатор

LAASS – навігаційний комплекс

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

WAASS – система супутників і наземних станцій, що робить коректування сигналів

GPS

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		2

ВСТУП

Актуальність теми. Компанія Alcatel-Lucent Enterprise доповнила своє рішення для бездротових мереж компонентом OmniAccess Stellar LBS. Технологічні нововведення допоможуть розширити цифрову взаємодію, поліпшити якість обслуговування клієнтів і підвищити їхню лояльність за рахунок сервісів, що враховують місце розташування користувача.

Прогнозується, що число людей, що використовують смартфони, до 2019 року перевищить 5 мільярдів. Аналогічне дослідження демонструє, що споживачі частіше роблять покупки, коли реклама має адресний підхід і враховує навколишнє оточення. Технологія LBS, що дозволяє використовувати місце розташування клієнтів, дає можливість для збільшення виторгу й поліпшує враження від використання сервісу.

Рішення Alcatel-Lucent OmniAccess Stellar LBS допомагає підвищити якість взаємодії зі споживачами за допомогою різних персоналізованих сервісів. За допомогою LBS компанії можуть оцінити й проаналізувати, як використовуються їхні площі, і з урахуванням отриманої інформації збільшити виторг, підвищити ефективність співробітників і скоротити витрати. Рішення задіє відразу дві технології – Bluetooth Low Energy (BLE) для користувачів мобільних телефонів і Stellar Wi-Fi для сервісів, що використовують місце розташування.

Рішення OmniAccess Stellar LBS оптимізоване для широкого кола завдань і галузей, у тому числі транспорту, охорони здоров'я й утворення. Його алгоритм задіє убудовані в смартфон сенсори: Bluetooth, Wi-Fi, GPS і датчик прискорення для надання інформації про точне місцезнаходження й мінімізує використання батареї телефону. Stellar LBS адаптується до навколишніх умов і забезпечує безперервний сервіс при переміщенні з будинків на відкрите повітря. Зокрема, нова функція Smart Park допомагає знайти місце, де припаркований автомобіль.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		3

Мета й завдання дослідження. Метою роботи є програмне забезпечення системи сервісів визначення місця розташування OmniAccess Stellar.

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

- Огляд існуючих систем сервісів визначення місця розташування OmniAccess Stellar.
- Дослідження системи сервісів визначення місця розташування OmniAccess Stellar.
- Програмна реалізація системи сервісів визначення місця розташування OmniAccess Stellar.

Практична цінність отриманих результатів полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі сервісів визначення місця розташування OmniAccess Stellar.

Таким чином, виходячи з вищеперерахованого, програмне забезпечення системи сервісів визначення місця розташування OmniAccess Stellar, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		4

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Stellar LBS надає користувачам просту у використанні навігацію по будинках, аеропортах, госпіталях і університетським кампусам. Технологія дозволяє підсилити систему маркетингового інформування й допомагає доставляти вигідні пропозиції про нові продукти й знижки з урахуванням точного місцезнаходження клієнта. Також Stellar LBS збирає інформацію про користувальницькі й поведінкові моделі – їхній аналіз дозволить бізнесу підготувати індивідуальні пропозиції й залучити нових клієнтів.

Радіопередавачі Stellar LBS Bluetooth Low Energy (BLE) використовують акумулятори військового класу й гарантують роботу рішення протягом 5-10 років, що у два рази більше в порівнянні з терміном дії більшості аналогів. Тривала експлуатація радіомаяків значно скорочує операційні витрати й забезпечує справну роботу геолокаційних сервісів протягом тривалого періоду.

Рішення оснащено хмарним механізмом керування. Це спрощує впровадження, роботу й підтримку внутрішніх геолокаційних сервісів, що створює високопродуктивну й просту у використанні BLE-інфраструктуру.

Сучасні технології допомагають бізнесу зрозуміти поведінку своїх споживачів на більше глибокому рівні, оптимізувати робітниче середовище й підвищити якість обслуговування. Рішення ALE LBS дозволить компаніям поліпшити взаємодію з кінцевими користувачами й співробітниками – від створення простої навігації в готелях до забезпечення безпеки в освітніх установах.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		5

1.2 Область застосування

Зовсім недавно соціальні мережі виступали як новий етап в еволюції людської комунікації, коли основне спілкування між людьми перемістилося в онлайн-середу. Зараз ми настільки щільно «підсіли» на «соціалки», що вони з розряду соціологічних феноменів перетворилися в складний інструмент самоорганізації суспільства, що робить вплив навіть на реальну політику. Соціальні мережі будуть розвиватися, трансформуватися, але вектор їхнього розвитку вже заданий, ніша сформована. Поговоримо про те, що буде на порядку денному в соціальних комунікаціях завтра, куди нас заведе технічний прогрес.

Тут однієї із самих перспективних технологій представляються геолокаційні сервіси (Location-based services-LBS), що дозволяють визначити місце розташування користувача/мобільного абонента.

Можна виділити кілька рівнів використання геолокації:

- Інформаційний – надання користувачеві інформації про найближчі до нього об'єктах або об'єктах певної локації.
- Соціальний – можливість довідатися місцезнаходження інших користувачів.
- Комунікаційний – можливість зареєструвати місце на карті, що стане доступним всім користувачам, залишити відгук про відвідування місця або прочитати відкликання інших, прив'язати власний контент до певного місця на карті (наприклад, фото або текстове повідомлення).

Через зростаючу популярність геолокаційних сервісів серед користувачів можна говорити й ще про один рівень – маркетинговому. Геолокація стала інструментом таргетування стимулюючих маркетингових акцій і основою для розробок нових механік взаємодії зі споживачем. Про цьому й поговоримо.

На найпростішому (і затребуваному) рівні орієнтування на місцевості геолокація відома нам по сервісах Google.Maps. Найвідоміші сервіси геолокації, зав'язані на «соціальній» функції, – All4geo, AlterGeo, FourSquare, Loopt.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		6

Крім цього, опції геолокації інтегруються у вже працюючі складні продукти, наприклад, в Facebook (Places), Google (Latitude і Places), Twitter.

Великі гравці індустрії намагаються осідлати хвилю інтересу користувачів до LBS. Останні покупки Facebook тому доказ – спочатку в грудні минулого року гігант придбала «туристичну» Gowalla (правда, досить швидко її й зактив), недавно було анонсоване поглинання геолокаційного застосунка Glancee. Цей крок – пряма відповідь на посилення позиції LBS-сервісу Foursquare, що у перспективі може стати прямим конкурентом Facebook.

Росте вплив LBS-сервісів і в знижковому бізнесі. Популярний знижковий агрегатор Groupon рухається в напрямку «локалізації» своїх користувачів і, за деякими даними, прощупує ґрунт у питанні придбання Foursquare.

Які фактори впливають на те, що геолокаційні сервіси так стрімко впроваджуються в споживчий бізнес і соціальні комунікації? Тут можна виділити дві групи інтересів, які, перетинаючись, утворюють резонанс: інтереси користувача й інтереси бізнесу.

Інтерес споживача:

– Геолокаційний сервіс розширює границі «соціальності». Раніше індивід міг демонструвати навколишніх свою identity через сторінку в соціальній мережі, де значилися його: інтереси, улюблена музика, місця, які йому подобаються, фотографії. Це була «статична» інформація, як зліпок з портрета користувача. Застиглі факти, відбиті моменти минулого. Цю картинку розроблювачі «пожвавили» сервісом мікроблогів і живою стрічкою новин. Користувач став «актуальнішим». Впровадження сервісу геолокацій стало логічним продовженням цієї «актуалізації» – користувач знайшов «динамічність». Тепер він може розповісти більше про свій спосіб життя, розпорядку дня, соціальному рівні. Як будь-який споживач бренду, він переймає частину декларуємого брендами успіху. У цьому випадку – брендів закладів, сфери послуг. У цьому змісті успіх геолокаційних сервісів можна пояснити тим, що вони фактично заповнили незайняту нішу «споживання брендів»: якщо раніше всі демонстрували свій

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		7

статус брендами-продуктами, то тепер з'явилася можливість додати до цього часом більше показові бренди сфери обслуговування.

– Добре продумана соціальна механіка, що забезпечує високу залученість користувачів у процес рекомендацій, оцінок, ранжирування. Практично всі успішні соціальні геолокаційні сервіси тією чи іншою мірою використовують ігрову форму взаємодії з користувачем, щоб зберегти його мотивацію для самостійної генерації контенту. У цьому велике досягнення сервісів геолокації – вони не зав'язані на модерації контенту адміністраторами й не обтяжені необхідністю наповнення об'ємного масиву даних по локаціям. Користувачі відзначаються й створюють нові мітки, щоб одержати ексклюзивний статус, набрати більше балів, ніж інші; щоб знаходити й мати можливість бути знайденими своїми друзями в певнім місці. Це механізм із дуже потужним віральним потенціалом, що зараз далеко не повністю розкритий. Правда, цей же механізм має зворотну сторону. Роблячи інформацію про себе публічною, користувач ризикує стати об'єктом неправомірних дій, що зачіпають його приватну. Так, у березні цього року Apple заблокував мобільний додаток з мовець назвою Girls Around Me, що показувало чоловікам, де поблизу перебувають жінки. При цьому самі жінки не догадувалися, що інформація про їхнє місце розташування, який вони ділилися зі своїми друзями, використовується кимсь ще. Щоб уникнути так званого «сталкінгу» (небажаної уваги до іншої особи і її нав'язливого переслідування), LBS-сервіси, швидше за все, будуть еволюціонувати у бік строго закритих груп друзів, яким ви персонально можете надати доступ до інформації про ваше місцезнаходження (приклад – додаток «Знайти моїх друзів» від Apple). Наслідку від необдуманого надання інформації про своє місцезнаходження можуть бути й куди більше серйозними – американським військовим в Іраку рекомендовано відключати геолокаційні опції в смартфонах щоб уникнути цільових мінометних атак супротивника.

– Комерційна вигода. Великий плюс LBS-сервісів для рядового споживача – це можливість одержати рекомендації, ради й підказки від інших

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		8

користувачів, які вже скористалися тією або іншою точкою обслуговування. Достоїнство це почасти пов'язане з високим ступенем довіри до персоналізованих рекомендацій і поки невисоким ступенем їх комерційного, «проплаченого» розміщення. Більші перспективи обіцяє синтез геолокацій з купонними сервісами. Приклад: найбільший український LBS-сервіс Altergeo пропонує знайти найближчі до вас закладу, де сьогодні діє знижка (звичайно 10-20%). Зареєструвавши купон прямо з застосунка, ви одержуєте ідентифікатор, пред'явлення якого в закладі дасть вам знижку. Це дорога, по якій уже щосили йде Groupon. Foursquare з липня також приєднається до купонної лихоманки. Таким чином, LBS-сервіси розширюють сферу свого впливу: від простого інформування, що перебуває навколо користувача, вони стають інструментом discount hunting'a (полювання за знижками), що дозволяє реально заощаджувати в ореолі крокової доступності, а не в масштабі величезного мегаполіса.

Як бізнес може використовувати той величезний потенціал, що мають LBS-сервіси:

1. Уже зараз впроваджується синхронізація між LBS-застосунками й системами компаній. Так, можна довідатися ціну бензину на найближчих АЗС, подивитися розклад кінотеатру, меню ресторану й кафе, забронювати столик, подивитися наявність вільних місць у готелях, забронювати номер. Звичайно, в авангарді нововведень – США, але, здається, найближчим часом ця хвиля дійде й до нашої країни.

З огляду на статистику використання LBS-сервісів, у найближчі місяці й роки цю технологію очікує лавиноподібний ріст. Якщо бізнес хоче одержати конкурентну перевагу, потрібно вже зараз включати роботу з аудиторією LBS-сервісів у маркетинговий план.

Про те, що цей інструмент уже працює й приносить прибуток бізнес-одиницям, говорить недавнє рішення Foursquare стягувати символічну плату в \$10 за верифікацію точок компаній на карті. Раніше це здійснювалося безкоштовно, за запитом в адміністрацію сервісу. З огляду на масштаб бази

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		9

компаній, які вже зареєстровані на Foursquare, компанія буде одержувати досить непоганий дохід.

Які конкретні методи маркетингового стимулювання споживачів можна використовувати, задіючи LBS:

- пропозиція купонної знижки LBS-користувачам з короткостроковим періодом дії (обмежуємо дискаунт хантерів) (наприклад, Altergeo).

- знижки за багаторазовий чек-ін/нагромадження балів (статус «міра» в Foursquare).

- спеціальні умови для тих, хто здійснює чек-ін (наприклад, популярний американський сервіс Shopkick нараховує бали за відвідування магазинів, які потім можна реалізувати в реальній знижці або безкоштовному продукті).

- групова знижка за одночасний чек-ін.

З огляду на вірусний характер поширення інформації в колі «мобільної» молоді й високий рівень лояльності до LBS-сервісів, активність у цьому напрямку може виявитися досить продуктивною. Хоча вірусний ефект прогнозувати досить складно.

- ігрові моделі використання чек-інів

Наприклад, компанія може запропонувати користувачам пройти «квест» – зареєструватися в декількох точках мережі своїх закладів, щоб одержати певний приз або виграти конкурс. Це може бути й послідовне проходження певного «маршруту» у заданий період часу. До цього «маршруту» можна прив'язати акції маркетингового стимулювання – ті ж самі знижки за чек-іни.

2. Системи геоконтекстної реклами (LBA – local-based advertising), по суті, є симбіозом LBS-сервісів і контекстної реклами. Система автоматично визначає місце розташування людини по GPS-координатах і використовує цей параметр для таргетування рекламних оголошень. Площадки для такої реклами – автомобільні навігатори, карти в мобільних телефонах і на веб-сайтах.

Геоконтекстна реклама може бути реалізована й у вигляді push-повідомлень від мобільних додатків. Людина йде по вулиці й одержує

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		10

повідомлення про те, що в магазині поруч іде розпродаж або в кав'язі через дорогу діє знижка на бізнес-ланчи – хто відмовить собі в задоволенні одержати миттєву, а не відкладену знижку, «по шляху»? Ця маркетингова можливість прямо пов'язана з питаннями безпеки даних про місцезнаходження користувача й регламентом їхнього комерційного використання. Зрозуміло, трекинг повинен здійснюватися тільки за умови згоди користувача на це.

3. Наступна можливість, що бізнес повинен (і буде) використовувати – це обробка інформації про користувача, що він уже залишив про себе в Мережі.

Реєструючись у певних місцях, у певний час, користувач малює широкими мазками розроблювачам свій автопортрет. Кожне місце транслює певний «меседж» зовні, що різними соціальними групами декодується по-своєму. З'єднуючи «пазл» з різних місць, які регулярно відвідує користувач, можна з великим ступенем точності визначити сферу його інтересів і коло спілкування.

Приналежність до певного соціального кластера – це не звична будь-якому маркетингологів сегментація. Люди різного соціального рівня й віку, які не перетинаються по стандартних соціально-демографічних параметрах, можуть бути прихильниками здорового способу життя, вегетаріанства, однієї конфесії, можуть займатися йогою, танцями, єдиноборствами. Аналіз їхніх маршрутів може допомогти більш правильно таргетувати спрямовану на них рекламу. І навіть міняти її зміст.

Так, можна робити індивідуальні пропозиції для тих користувачів, хто вже замічений як клієнт прямих конкурентів або є частим відвідувачем подібного роду закладів («активний клієнт»). Робити знижки на бізнес-ланчи для тих, хто відзначається в даній локації вдень, і на романтичні вечери – для тих, хто робить це ввечері.

Для користувач із більшою кількістю соціальних зв'язків у Мережі будуть актуальні групові програми. Колективні знижки й спеціальні пропозиції будуть стимулювати інтерес відвідати точку не поодиночі, а цілою компанією.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		11

Історія пошуку користувача також може вважатися їжею для міркувань. Пропонуючи менш відому, але аналогічну по суті або більше вигідну альтернативу, можна чекати позитивний відгук. Це актуально для нових, ще не розкритих місць.

4. Мабуть, саме вражаюче на сьогоднішній день використання можливостей геолокації – її спільне використання з доповненою реальністю. Суть її полягає в тому, що на об'єкти реального світу накладається віртуальні елементи, у розважальних або інформаційних цілях. Ви наводите смартфон на будь-яку точку в місті, і на екран виводиться інформація про неї, будь те вулиця, будинок, цікавий об'єкт і так далі.

Продовжуючи розвивати цю тему, можна буде «побачити» друзів, що сидять усередині кафе або були тут раніше, наявність вільних місць, ціновий рівень ресторану й так далі. Те, що ми ще 20 назад бачили у фантастичних фільмах, ми незабаром побачимо своїми очами на екранах наших незамінних гаджетів.

Таким чином, виходячи з вищеперерахованого, програмне забезпечення системи сервісів визначення місця розташування OmniAccess Stellar, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за першим (бакалаврським) рівнем вищої освіти.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		12

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

LocationSmart

LocationSmart – найбільший у світі постачальник систем функціонального доповнення з локальною зоною дії (as-a-Service, LaaS), він пропонує визначення місця розташування пристроїв для корпоративних і споживчих додатків.

Геолокаційні сервіси можуть стати ключовим фактором, що визначає індивідуальний підхід і скорочення витрат. Хмарне розміщення LocationSmart і використання платформи обміну повідомленнями дозволяють надати контакт-центрам можливість автоматичного визначення місцезнаходження клієнта, що подзвонив, тим самим сприяючи скороченню витрат, підвищенню ефективності роботи й першокласному обслуговуванню клієнтів.

Продукти Genesys і LocationSmart можна інтегрувати як взаємодоповнюючі технології, що дозволить створити потужну комбінацію для рішення безлічі проблем, з якими зіштовхуються контакт-центри.

Автоматизація визначення місця розташування клієнта в багатьох змістах спрощує роботу контакт-центра, що дозволяє скоротити витрати:

- зменшується вплив людського фактора, оскільки виключається необхідність введення адреси вручну;
- з'являється можливість визначення місця розташування клієнта, навіть якщо він сам не знає, де перебуває;
- за рахунок автоматизації неефективних випадків взаємодії скорочуються тривалість дзвінків і кількість пропущених викликів.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		13

Інтеграція платформи LocationSmart з вашим контакт-центром дозволить об'єднати витратні за часом процеси, що забезпечить максимальне збільшення продуктивності й можливість сконцентруватися на інших найважливіших функціях.

Геолокаційні сервіси можна використовувати для рішення наступних завдань:

– Маршрутизація дзвінків. Перенапрямок дзвінків на підходящий регіональний або місцевий пункт зв'язку для оптимізації якості обслуговування.

– Організація черги дзвінків із пріоритетом. Переміщення пріоритетних дзвінків в екстреної або потребує багаторічної обробки випадках у початок черги залежно від географічного розташування.

– Допомога у використанні. Застосування інформації про місце розташування мобільного пристрою для підвищення якості взаємодії й спрощення розподілу в підходящі пункти зв'язку.

– Пошук об'єктів поблизу. Можливість включити пошук торговельних об'єктів на підставі дзвінків, що надійшли, і виконати пошук найближчих пунктів обслуговування, щоб позбавити клієнтів від необхідності введення поштових індексів.

Платформа LocationSmart являє собою хмарну технологію, що не передбачає завантаження програмного забезпечення або додатків. Така технологія легко вбудовується в існуючі програми голосового зв'язку, а крім того, можливо її впровадження на початкових стадіях розвитку. LocationSmart пропонує контакт-центрам послуги визначення місця розташування по сигналах базових станцій всіх операторів мобільного зв'язку, визначення місця розташування по IP-адресі, а також сервіси для обміну повідомленнями.

Life360 – вся родина разом

Для сучасних родин, всі члени яких мають власні смартфони, одним із самих оптимальних рішень для визначення місця розташування один одного стане додаток Life360.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		14

В Life360 можна не тільки побачити місце розташування рідних або друзів, але й листуватися з ними в загальному чаті.

Після установки застосунку користувачі можуть створювати в ньому закриті приватні групи для своїх рідних або друзів, положення всіх учасників яких відображається на убудованій карті. Ця інформація може обновлятися в режимі реального часу на основі даних GPS-модуля або по чек-інам користувачів. Геолокаційні можливості Life360 також включають автоматичне відправлення оповіщень у моменти, коли окремі користувачі з'являються в обраних геозонах або залишають їх. За допомогою цієї функції, наприклад, можна одержувати повідомлення, коли дитина буде виходити зі школи.

Крім цього, члени однієї групи можуть листуватися в загальному чаті й посилати іншим користувачам приватні повідомлення, що перетворює Life360 у справжній сімейний центр. У застосунку доступна й кнопка «паніки», що відсилає повідомлення з координатами й проханням передзвонити всім учасникам групи.

Програма не обмежує користувачів вибором певних пристроїв – сервіс працює на смартфонах під керуванням Android, iOS і Windows Phone. Life360 вимагає платної підписки в розмірі п'яти доларів на місяць або 50 у рік, але новим користувачам безкоштовно надається 30 днів для знайомства з додатком.

«Локатор», «Маячок», «Дитина під доглядом» – сервіси від операторів

Користуватися всією родиною одним додатком і постійно тримати включеним GPS – не єдиний спосіб одержувати дані про місце розташування своїх близьких. У деяких ситуаціях більше простим рішенням можуть стати геолокаційні сервіси, пропоновані операторами стільникового зв'язка.

Послуги з визначення координат телефону пропонують всі українські оператори.

Головною їхньою перевагою є те, що вони підключаються безпосередньо до телефонного номера й можуть працювати з усіма пристроями, а не тільки зі смартфонами. Таким чином, навіть якщо дитина або літні батьки користуються

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		15

звичайним мобільним телефоном, сервіси від операторів дозволяють одержувати інформацію про їхнє місце розташування. При цьому приналежність номера не має значення – відслідковувати можна телефони всіх операторів, а не тільки ті, які належать одній мережі. Рішення, пропоновані операторами «великої трійки», по своїй функціональності не сильно відрізняються друг від друга. Всі вони використовують дані від базових станцій стільникового зв'язку, тому в певних умовах можуть давати досить більшу погрішність – від декількох сотень метрів до одного кілометра. Керування послугою у всіх сервісах реалізовано як за допомогою команд, що відправляються по SMS, так і через спеціальний мобільний додаток або веб-сайт. За запитом користувач просто одержує координати, які відображаються у вигляді точки на карті.

Підключення геолокаційної послуги на телефоні, що буде використовуватися для контролю, проходить як і у всіх інших мобільних сервісів – через особистий кабінет на сайті оператора або за допомогою SMS. А от підключення номерів, що відслідковуються, вимагає підтвердження від їхнього власника – на номер приходить SMS-повідомлення, на яке обов'язково потрібно відповісти. Як правило, оператори надають можливість стежити за декількома абонентами – можна підключити від одного до п'яти телефонних номерів. Також обмежується число запитів, які можна відправити за один день, але на практиці навряд чи може знадобитися дізнаватися координати людини більше 100 разів у добу.

Набори додаткових послуг, надаваних разом із сервісом геолокації, у кожного оператора свої. Vodafone, наприклад, у рамках послуги «Дитина під доглядом» пропонує можливість створення окремих геозон, що дозволяють одержувати SMS-повідомлення при всіх переміщеннях власника номера.

Amber Alert GPS – трекер для дитини

Визначення місця розташування за допомогою телефону має свої недоліки, особливо якщо необхідно стежити за переміщеннями дитини. Багато хто воліють не купувати дітям смартфони, а у звичайних мобільних телефонів

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		16

точність геолокаційних сервісів із прив'язкою до базових станцій часто залишає бажати кращого. До того ж у пристрою може сісти батарея або його можна просто виключити. Розібратися з подібними проблемами допоможе окремий GPS-трекер, наприклад спеціально призначений для дітей Amber Alert GPS.

Дитина може подзвонити батькам з Amber Alert GPS натисканням однієї кнопки.

Це пристрій фактично являє собою мініатюрний телефон з постійно включеним GPS і можливістю набору певного номера натисканням однієї кнопки. Його можна налаштувати на відправлення координат з певною періодичністю – від декількох хвилин до годин. По завіреннях виробників, час автономної роботи Amber Alert GPS навіть при максимально частоту відновленні координат становить близько 40 годин.

Батьки можуть переглядати всю інформацію про місцезнаходження дитини за допомогою веб-сервісу або в спеціальному мобільному застосунку. За допомогою керуючої програми можна також налаштувати до 20 геозон, при перетинанні границь яких буде відправлятися повідомлення.

PocketFinder – універсальний маячок

Втім, спеціальну GPS-мітку можна використовувати не тільки для того, щоб стежити за переміщеннями своєї дитини. За допомогою подібного пристрою можна контролювати й місце розташування свого автомобіля, домашнього вихованця або багажу. Одним з найпростіших, але функціональних гаджетів, призначених для подібної геолокації, є PocketFinder.

PocketFinder можна використовувати як GPS-трекер для автомобіля, собаки або багажу.

Цей мініатюрний пристрій має досить більший набір функцій для геотрекінгу. Його можна налаштувати на відправлення координат з певною періодичністю й одержувати повідомлення на смартфон або електронну пошту, коли датчик буде перетинати границі певних зон. Якщо використовувати PocketFinder для відстеження місця розташування автомобіля, то користувачеві

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		17

доступна функція повідомлень про перевищення швидкості й перегляд 60-денної історії поїздок.

Пристрій має запалі- і вологозахищений корпус, так що здатний пережити навіть раптове купання, якщо воно використовується в якості трекера для собаки. Відправлення даних здійснюється через стільниковий зв'язок – для роботи гаджета потрібно встановити в нього SIM-карту. Управляти PocketFinder можна через мобільні додатки для операційних систем Android і iOS.

«Кнопка життя» – техніка приходить на допомогу

Пристрою з геолокацією незамінні й для літніх користувачів, особливо якщо вони об'єднані з медичними сервісами. Зокрема, в Україні подібні послуги пропонує спеціалізована служба «Кнопка життя».

Пристрою із сервісом «Кнопка життя» здатні самостійно відправити тривожний сигнал лікарям при настанні екстреної ситуації.

Це не просто виробник електронної техніки. Насамперед «Кнопка життя» – це окремий український Центр допомоги, у якому оператори обробляють сигнали, що надходять із пристроїв користувачів. При цьому сама техніка може бути дуже різної – підтримка сервісу є як у спеціальних телефонах для літніх, так і в традиційних GPS-трекерах і пристроях, що носяться навіть, з можливістю зворотного зв'язку. Одним із самих функціональних варіантів, доступних для покупки на сайті сервісу, є тривожний браслет S911, що фактично представляє собою «розумні» годинники.

Він оснащений GPS-модулем і SIM-картою, завдяки чому підтримує не тільки відправлення координат, але й голосовий зв'язок – на браслет і з його можна подзвонити. Таким чином, родичі в будь-який момент можуть зв'язатися із власником пристрою або довідатися, де він перебуває, скориставшись геолокаційною службою. Мобільний зв'язок для пристроїв з «Кнопкою життя» надає Vodafone – спеціальний тариф для сервісу відрізняється відсутністю абонентської плати й низкою вартістю дзвінків.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		18

Браслет також оснащений тривожною кнопкою, при натисканні якої встановлюється зв'язок з оператором центра допомоги. Дзвінок приймає черговий лікар, що у момент виклику відразу ж одержує поточні координати власника й інформацію з його медичної карти, включаючи дані про хвороби й алергійні реакції на медичні препарати. На додаток до цього в браслеті є окремий датчик падінь. При його спрацьовуванні, навіть якщо власник в екстреній ситуації не може викликати допомогу самостійно, тривожний сигнал надходить операторові, що зв'язується із самим користувачем, а при відсутності відповіді – з його родичами, довіреними особами й швидкою допомогою.

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

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

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

Швидкість виконання коду Python

Один з можливих недоліків Python – швидкість виконання коду. Python не є компільованою мовою. Код на Python спочатку компілюється у внутрішній байт-код, який потім виконується інтерпретатором Python. У більшості випадків при використанні Python виходять програми повільніші в порівнянні з такими мовами, як C.

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

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

Використання Python

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

Застосування мови програмування Python:

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

8. DropBox – файловий хостинг, що включає персональне хмарне сховище, синхронізацію файлів і програму-клієнт.

9. YouTube – популярне відеосховище.

Версії Python

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

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

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

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

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

Коли процесор виконує інструкції програми, він бере участь у процесі, який є відомим як цикл `fetch – decode – execute` (отримати – декодувати –

виконати). Цей цикл виконується для кожної інструкції у програмі і складається з трьох кроків:

Отримати

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

Декодувати

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

Виконати

Останній крок циклу – виконати операцію.

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

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

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

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

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		22

Python – одна із популярних сучасних мов програмування високого рівня. Python – інтерпретована мова програмування. Python – це високорівнева інтерпретована мова програмування, на відміну від C++, яка є прикладом компільованої мови програмування. Назва Python відноситься як до мови програмування, так і до інтерпретатора – комп'ютерної програми, яка зчитує початковий код (написаний на Python) і виконує інструкції (команди).

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

1. Компілятор.
2. Інтерпретатор.

Завантаження Python

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

Середовище програмування для Python

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

Текстові редактори та інтегровані середовища програмування для Python:

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

– Wing IDE 101 – вільне інтегроване середовище для Python, розроблене для навчання програмістів-початківців. Для користувачів Linux, Windows і macOS. Поширюється безкоштовно.

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

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

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

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

2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випускню кваліфікаційну роботу за першим (бакалаврським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи сервісів визначення місця розташування OmniAccess Stellar.

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

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

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

в) розробити програмне забезпечення системи, що дозволить реалізувати

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		24

поставлену технічним завданням задачу. Побудувати блок-схеми алгоритмів програми та підпрограми;

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

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

е) провести розрахунки по визначенню економічної ефективності розробленої системи;

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

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

КБПЗ-2026

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		25

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

Набір OmniAccess Stellar LBS пропонує три функції зміни місця розташування для розширення можливостей вашої компанії, розширення ваших послуг, отримання глибокої аналітики та допомоги для прийняття кращих рішень.

OmniAccess Stellar LBS забезпечує більше, ніж просто навігацію в аеропортах, лікарнях, готелях та університетських містечках. Це основа безлічі інноваційних послуг, таких як:

- Близький маркетинг для акцій ресторану та роздрібної торгівлі.
- Автоматизовані інформаційні повідомлення та надзвичайні сповіщення.
- Розумна паркова.
- Обмін місцезнаходженням.
- Пошук визначних місць із покроковими вказівками.
- Мобілізація та координація персоналу.

Крім того, OmniAccess Stellar LBS пропонує поглиблену аналітику місцеположення, щоб краще зрозуміти потоки людей, час перебування, використання площі та концентрацію. Поєднання цих ексклюзивних функцій може допомогти організаціям покращити процеси, оптимізувати залучення клієнтів та налагодити надання послуг.

Позиціонування в приміщенні

Позиціонування в приміщенні є ключовою особливістю, яка може перетворити смартфон у внутрішній GPS-пристрій. Це створює цінні послуги у вашому мобільному застосунку, такі як націлювання маршрутів, покрокова навігація, пошук друзів, запам'ятовують моє місце для паркування, діляться цікавими місцями тощо.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		26

Технологія позиціонування ALE оптимізована для широкого спектру випадків використання. Він використовує унікальну гібридну технологію 'fusion engine', інтелектуально поєднуючи дані з навколишнього середовища та вбудовані датчики смартфона: Bluetooth, Wi-Fi, GPS, акселерометр, магнітометр, гіроскоп, барометр, карта локації ...

Забезпечивши безпрецедентну точність і безперебійний перехід від внутрішнього до зовнішнього місця (важливо у великих кампусах), ця технологія також мінімізує використання батареї смартфона.

Геофанування

Геофанінг надсилає сповіщення мобільному застосунку залежно від місцезнаходження користувача. Сповіщення про географічне обмеження можуть бути спровоковані різними подіями, наприклад, коли користувач входить, виходить або залишається у визначеній зоні. За допомогою OmniAccess Stellar LBS Suite області географічного простору просто малюються на карті з платформи Cloud Manager, потім синхронізуються та виконуються SDK Stellar LBS всередині мобільного застосунку.

Деякі типові приклади служб геофіксування включають:

- Близький маркетинг: за допомогою push-сповіщень на мобільному телефоні ви можете легко спілкуватися з відвідувачами. Відвідувач просто вирішує скористатися будь-якими купонами чи завантаженнями пропозицій програми.

- Подання інформації: Коли користувач наближається до цікавої точки, продукту чи витвору мистецтва, служба геофіксації дає змогу спілкуватися з ними через контекстуалізовані, націлені повідомлення.

- Вхід та виявлення зон: Потужний двигун локації, вбудований у SDK у поєднанні з маяками, розміщеними на стратегічних місцях, дозволяє дуже точно визначити вхід та вихід відвідувачів із віртуальних зон. Цей тип послуг дозволяє краще керувати рухами та оптимізувати управління трафіком. Часто впроваджується на корпоративних майданчиках для контролю за розміщеністю

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		27

приміщень, виявленню входу та виїзду пропонує високу додану вартість.

Аналітика

Усі дані про місцезнаходження в приміщенні – це ваше використання, і ви можете переглянути їх на платформі Cloud Manager. Зведена тут аналітика надає різноманітні статистичні дані (кількість, тривалість, частота відвідувань ...), щоб ви могли:

- Проаналізуйте поведінку своїх користувачів – оцінюючи трафік, слідкуючи за рухами, вивчаючи потоки
- Керуйте потоками в закритому просторі, наприклад, офісі або конференц-залі та його використання для оптимізації та адаптації ваших продуктів та послуг
- Виміряйте вплив та рентабельність інвестицій у ваші мобільні маркетингові кампанії
- Оптимізуйте та адаптуйте свій простір як частина стратегії постійного вдосконалення

Інструменти управління OmniAccess Stellar LBS

Два потужних інструменти управління для спрощення впровадження, експлуатації та обслуговування служб локації в приміщенні.

OmniAccess Stellar LBS Cloud Manager

Хмарний менеджер – це всебічна платформа, яка надає вам повний доступ до всіх доступних ресурсів, будь то для одного місця, великого кампусу чи цілого розумного міста з кількома будівлями. OmniAccess Stellar LBS Cloud Manager забезпечує безпрецедентну масштабованість та просте управління.

За допомогою Cloud Manager ви можете віддалено визначати потреби в приміщенні та керувати ними.

OmniAccess Stellar LBS Installer

Цей простий додаток для смартфонів є важливим інструментом, який допоможе вам:

- Відкалібруйте службу локації в приміщенні.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		28

- Оцініть точність розташування.
- Випробування геонотифікацій.
- Вкажіть місця маяка.
- Перевіряйте стан життя маяка в режимі реального часу.
- (Re) налаштування маяків.

Маяки BLE OmniAccess

Прості в управлінні акумуляторні сигнали Bluetooth, що працюють на акумуляторі, за допомогою технології iBeacon, забезпечують особистий досвід на основі місцезнаходження.

Рішення OmniAccess Stellar LBS використовує маячки Bluetooth (фізичні пристрої) з низьким рівнем енергії (фізичні пристрої) для надання статичної інформації про місце розташування смартфона, встановленому за допомогою OmniAccess Stellar LBS SDK, коли він знаходиться в безпосередній близькості від місця, де знаходиться маяк.

Ці маяки BLE можуть надавати статичну інформацію про місцезнаходження, стан живлення акумулятора (для автономних маяків), ідентифікатор маяка, силу сигналу живлення, частоту сигналу маяка та стан підключення

Набір для розробки програмного забезпечення OmniAccess Stellar LBS

Розробіть рішення для вирішення будь-яких вимог до місця розташування, від найпростішого до найскладнішого – все, забезпечуючи при цьому найвищу якість та надійність.

OmniAccess Stellar LBS SDK – це комплект для розробки програмного забезпечення: набір засобів для розробки програмного забезпечення, що дозволяє створювати послуги на основі розташування на різних мобільних пристроях та операційних системах. За допомогою смартфона та маяків OmniAccess Stellar BLE або OmniAccess Stellar 1230AP, він фіксує сигнали місцезнаходження від маяків і переводить їх у інформацію про місцезнаходження в реальному часі.

Дані, зібрані з SDK, передаються через OmniAccess Stellar LBS Cloud Manager, програму хмарного управління, через Wi-Fi або 3G / 4G / LTE.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		29

3.2 Розробка структурної схеми

Картографія перетворилася в чудо – з безумно великих аркушів паперу в те, що вміщається на долоні. На сьогодні, карти використовуються не тільки для пошуку культурного центра в Парижу, але щоб мати можливість знайти все:

- найближчий готель;
- аптеку або лікарню;
- продуктовий магазин;
- університет або школу;
- місце проведення концерту Ореста Лютного у Кропивницькому.

Трансформація технології на цьому не зупиниться. Це стане краще й точніше, як ніколи.

В 1989 році в США був запущений найперший супутник GPS для комерційного використання. Розвиток цієї технології став швидким із цього моменту. Усього за 10 років Benefon Inc. стала першою компанією, що представила перший комерційно доступний мобільний телефон з убудованим GPS. Це 2020 рік і ми пройшли довгий шлях із часів Бенефона.

Ви можете знайти загублених свійських тварина з нашийниками, дітей, батьків, які мають убудований GPS у розумні пристрої.

Зараз можна знайти місце розташування вашого улюбленого гамбургера і його доставку, не встаючи зі свого дивана.

От, як ми використовуємо цю технологію індивідуально в повсякденному житті.

Геолокація використовується в набагато більших масштабах – підприємствами для відстеження й націлювання на потрібних користувачів, щоб збільшити продажі й зробити охоплення потрібної аудиторії клієнтів.

Отже, перший порядок бізнесу й основне правило нашого часу: розкусити цей термін на дійсно потрібні й потребуючі визначення.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		30

Геолокація – це ідентифікація або оцінка реального географічного місця розташування об'єкта, такого як радарне джерело, мобільний телефон, базова станція або будь-який девайс підключений до Інтернету. У своїй найпростішій формі геолокація включає генерацію набору географічних координат і тісно пов'язана з використанням систем позиціонування, але її корисність підвищується завдяки використанню цих координат для визначення значимого місця розташування, такого як адреса вулиці, поштового індексу.

Інакше кажучи, геолокація – це Вашу фізичну мережну адресу на планеті.

IPv4 і IPv6 адреси

IP є аббревіатурою від інтернет-протоколу. IP-адреса – це унікально ідентифікуєма мітка для кожного пристрою, підключеного до комп'ютерної мережі й яка має розподіл по місцю реєстрації для континентів (реєстратори). Або, простіше говорячи, IP – це ваша віртуальна адреса.

Де б у світі ви не користувалися Інтернетом, Вам буде привласнена IP-адреса.

IP-адреси формуються у власні діапазони, які закріплені регіональними інтернет-реєстраторами для різних провайдерів і власників мереж, регульовані автономною системою (ASN):

- American Registry for Internet Numbers (ARIN) – для Північної Америки.
- RIPE Network Coordination Centre (RIPE NCC) – для Європи, Близького Сходу й Центральної Азії.
- Asia-Pacific Network Information Centre (APNIC) – для Азії й Тихоокеанського регіону.
- Latin American and Caribbean Network Information Centre (LACNIC) – для Латинської Америки й Карибського регіону.
- African Network Information Centre (AfriNIC) – для Африки й регіону Індійського океану.

При роботі з IP-адресами Ви побачите два типи IP-адрес:

0.0.0.0 – це адреса IPv4,

а

2001:db8:85a3::8a2e:370:7334 – приклад адреси IPv6.

API – це скорочення від інтерфейсу прикладного програмування.

Це набір функцій, що дозволяє додаткам спілкуватися один з одним. Ви бачили сайти, які дозволяють Вам зареєструватися через Facebook або Twitter (або будь-яку зовнішню платформу). Ці відносини між цим сайтом і Facebook створюються за допомогою API.

Отже, тепер, коли ми знаємо, що таке геолокація, IP і API, я представляю Вам IP Geolocation API.

API геолокації на ділі

IP Geolocation API – це невеликий фрагмент коду (API), що допомагає визначити приблизне фізичне місце розташування користувача за допомогою його віртуальної адреси (IP). IP Geolocation – це метод визначення приблизного фізичного місця розташування підключеного до Інтернету пристрою через адресу, а IP Geolocation API – це доступність цього методу у вигляді API.

Варіанти використання IP Geolocation

Якщо Ви маркетолог, то можете використовувати інформацію із цього API і направити свої маркетингові зусилля в правильному й коректному напрямку, щоб одержати гідний результат при взаємодії.

Інтернет-продавці можуть використовувати IP-геолокацію для відстеження будь-якого потенційного шахрайства із кредитними картами. Це робиться шляхом зіставлення IP-адреси покупця з адресою доставки/виставлення рахунку, що він увів.

Наприклад, якщо уведена адреса доставки перебуває в Індії, але IP-адреса вказує нам місцезнаходження відвідувача в Австралії, він може бути позначений як шахрайство.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		32

Правоохоронні органи використовують геолокацію для відстеження злочинної діяльності в Інтернеті й запобігання майбутніх правопорушень.

Геозони – це «огороження» навколо контенту, що дозволений для показу в певних регіонах. Можливо, Ви зіштовхувалися із сайтами, на яких написано: «Цей контент заблокований для перегляду у Вашій країні».

Так, це неприємно й створює труднощі при доступі до певних сервісів.

Наприклад, Вконтакте заблокований в Україні, останні відео Джиммі Кіммела заблоковані в Південно-східній Азії, а Pokemon Go споконвічно дозволено відтворювати в США (але воно було випущено більш ніж в 60 країнах).

Багато великих продавців електронної комерції успішно використовують цю технологію й одержують дійсно приголомшливі результати.

Ми склали список деяких речей, щоб показати, як вони відмінно справляються з маркетингом за допомогою IP-геолокації.

Інтегруючи засобу геолокаційних сервісів і комбінуючи їх із усілякими SDK, плагінами, створюється ситуація коли великий пріоритет спостерігається саме до геолокації й для неї властиво одержувати більшу кількість параметрів.

Функціонал і можливості надання послуги в сервісах просто потрясає, для IP-адреси Ви можете одержати: хостнейм, ім'я й код континенту, ім'я й код країни у двох форматах, прапор країни, столицю, назви країни, область, район, місто, поштовий індекс, координати (довгота й широта), телефонний код країни, TLD, список мов країни, тип з'єднання, ім'я організації й провайдеру, валюту і її символ, годинну зону і поточний час – цей багаж даних дає грандіозну можливість для впровадження подібної інформації про IP-адреси Ваших же клієнтів. Це велике надбання для дійсно нужденних у найбільшій деталізації користувальницьких даних проектів і сервісів, одержуючи спектр і зведення в такому масштабі.

Така кількість надаваної інформації для адреси в мережі допоможе вдосконалити аналітичні, статистичні й моніторингові дані про клієнта, щоб поліпшити таргетинг свого продукту або бізнесу.

Якщо Ви хочете одержати набагато більше й створити власний пакет метаданих при використанні гео-сервісів, це правильний варіант щоб почати застосовувати аналітичну статистику й аналіз, а так-же просто комбінувати допоміжні інструменти для зручностей використання, визначення й виводу інформації на рівні геолокації.

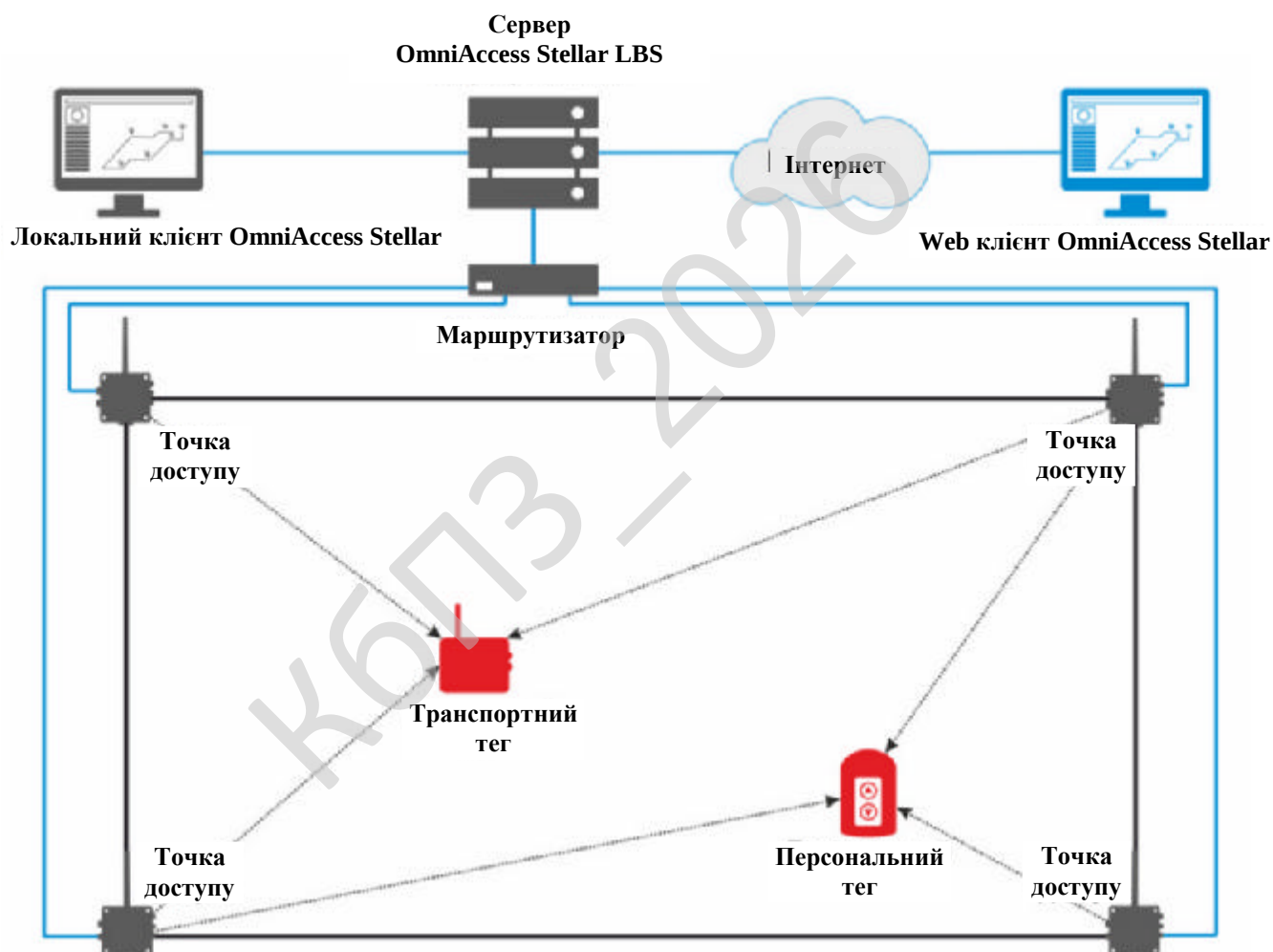


Рисунок 3.1 – Структурна схема роботи

3.3 Розробка функціональної схеми

Реальні магазини, що використовують традиційні інструменти аналітики місця розташування, наприклад дверні лічильники й камери, не здатні сегментувати аудиторію й проводити таргетинг так само ефективно, як їхні інтернет-конкуренти. Крім того, ці організації ризикують втратити прибуток через неправильне розміщення персоналу й неоптимального оформлення вітрин. Завдяки сервісам визначення місця розташування OmniAccess Stellar LBS можна поліпшити взаємодію із клієнтами на роздрібних підприємствах, стадіонах і в транспортних вузлах, використовуючи дані про точне місце розташування клієнтів.

OmniAccess Stellar LBS формує актуальні із практичної точки зору дані, які допомагають організаціям досягти детального розуміння свого об'єкта й підвищити загальну ефективність. OmniAccess Stellar LBS розгортається в мережах WLAN на основі пристроїв Ruckus, а тому має гнучкі можливості розгортання, не вимагає якого-небудь додаткового встаткування й має необмежену масштабованість у хмарі.

Розглянемо функціональні блоки

Блок одержання цінної бізнес-інформації й детальних даних про місце розташування

Постійно зіштовхуєтеся з низькими обсягами продажів у магазині або зазнаєте труднощів при забезпеченні видимості кампаній серед цільової аудиторії? Ядро визначення місця розташування Ruckus OmniAccess Stellar LBS формує корисну інформацію про місце розташування, що може допомогти в прийнятті стратегічних рішень для підвищення операційної ефективності й ефективності рекламних кампаній. Збирайте кошовну інформацію про ключових клієнтів і розробляйте моделі взаємодії, використовуючи дані аналітики відвідуваності без установки або завантаження додаткового ПЗ на стороні користувача. Завдяки хмарній архітектурі OmniAccess Stellar LBS підтримує

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		35

необмежена кількість пристроїв і об'єктів, а також забезпечує конфіденційність даних. OmniAccess Stellar LBS дозволяє:

- відстеження місця розташування споживачів у режимі реального часу для більше ефективного планування схеми об'єкта й розташування вітрин;
- оптимізування кількості персоналу на підставі припливу клієнтів за допомогою даних про відвідуваність;
- виключення місцевих пристроїв Wi-Fi з аналітики місця розташування для формування даних з підвищеною точністю;

Блок підвищення прибутку й залучення клієнтів за допомогою екосистемних рішень

Останнім часом завдання по таргетуванню й утриманню клієнтів стали для реальних магазинів складними як ніколи. OmniAccess Stellar LBS пропонує безпечні інтерфейси API, які можуть використовуватися партнерами по екосистемі для розробки галузевих сервісів на основі геолокації (LBS). Завдяки LBS ви зможете надавати клієнтам магазину унікальне обслуговування, а також:

- визначення демографії клієнтів;
- підвищення залучення користувачів за допомогою таргетованих рекламних кампаній, знижок і програм лояльності;
- пропозиція таргетованих програм для особливо кошовних клієнтів;
- аналіз ефективності маркетингових кампаній.

Блок відстеження кошовного майна для мінімізації втрат

Організації, що розташовують кошовним майном, часто його втрачають або зіштовхуються з його крадіжкою. Завдяки функції відстеження кошовного майна з підтримкою Wi-Fi сервіс OmniAccess Stellar LBS знімає тягар відповідальності з IT-відділів. OmniAccess Stellar LBS дозволяє відслідковувати місцезнаходження попередньо заданого майна в режимі реального часу, а також пропонує історичні дані про його місцезнаходження. Щоб знайти пристрій, просто введіть його унікальну MAC-адресу в панелі відстеження пристроїв.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		36

Блок одержання цінної бізнес-інформації й детальних даних про місце розташування

- відстеження місця розташування споживачів у режимі реального часу для більше ефективного планування схеми об'єкта й розташування вітрин;
- оптимізування кількості персоналу на підставі припливу клієнтів за допомогою даних про відвідуваність;
- виключення місцевих пристроїв Wi-Fi з аналітики місця розташування для формування даних з підвищеною точністю.

Блок відстеження коштовного майна для мінімізації втрат

- миттєве визначення місцезнаходження будь-якого пристрою в режимі реального часу й одержати доступ до історичних даних про його місцезнаходження за останні 7 днів;
- перегляд історії переміщення пристрою у вигляді теплових карт із вказівкою періоду часу для кожного місця;
- налаштування попереджень, видаваних при несанкціонованому переносі пристрою, – для запобігання крадіжки або втрати;
- визначення місцезнаходження користувачів у випадку надзвичайної ситуації.

Програмне забезпечення системи сервісів визначення місця розташування
OmniAccess Stellar

Блок підвищення прибутку й залучення клієнтів за допомогою екосистемних рішень

- визначення демографії клієнтів;
- підвищення залучення користувачів за допомогою таргетованих рекламних кампаній, знижок і програм лояльності;
- пропозиція таргетованих програм для особливо коштовних клієнтів;
- аналіз ефективності маркетингових кампаній.

Рисунок 3.2 – Функціональна схема системи

Організації, які використовують відслідковування пристроїв за допомогою сервісу OmniAccess Stellar LBS, можуть реалізувати наступні дії:

- миттєве визначення місцезнаходження будь-якого пристрою в режимі реального часу й одержати доступ до історичних даних про його місцезнаходження за останні 7 днів;

- перегляд історії переміщення пристрою у вигляді теплових карт із вказівкою періоду часу для кожного місця;
- налаштування попереджень, видаваних при несанкціонованому переносі пристрою, – для запобігання крадіжки або втрати;
- визначення місцезнаходження користувачів у випадку надзвичайної ситуації.

3.4 Розробка діаграми процесів

Розглянемо розроблену діаграму процесів яка зображена на рисунку 3.3. Основна будова діаграми процесів полягає у графічному представленні складу сукупностей даних, що характеризуються як співвідношення різних частин кожної з сукупностей. Склад статистичної сукупності графічно може бути представлений як за допомогою абсолютних, так і відносних показників. Графічне зображення складу сукупності по абсолютними і відносними показниками сприяє проведенню більш глибокого аналізу і дозволяє проводити аналіз системи.

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

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

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

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

Діаграми потоків даних містять чотири типи елементів:

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

- Процеси які являють собою трансформацію даних в рамках описуваної системи.
- Сховища даних (репозиторії).

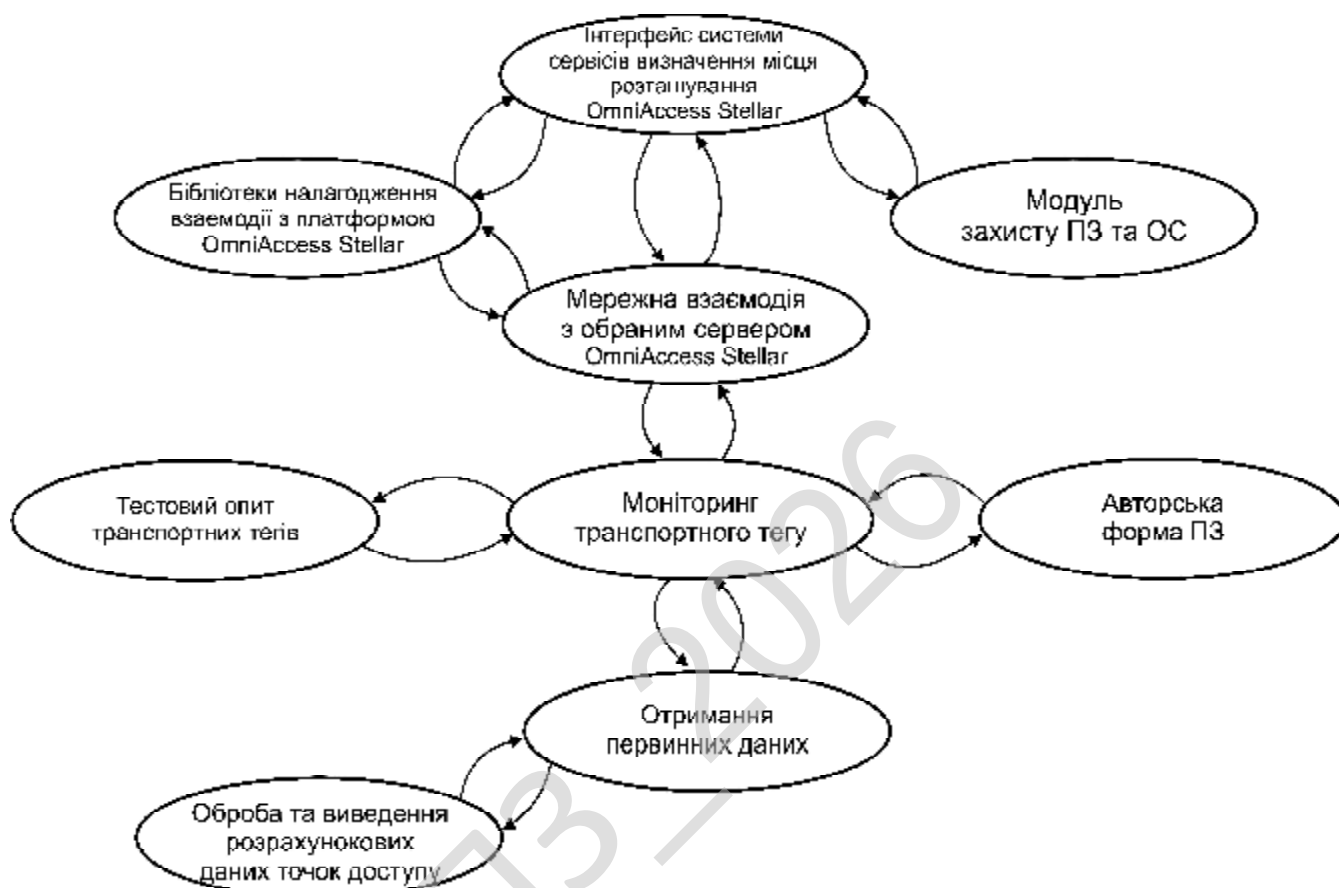


Рисунок 3.3 – Діаграма взаємодії процесів

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

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

4.1 Блок-схеми та опис алгоритмів функціонування системи

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

Було створено блок-схеми роботи системи. Перед їх розглядом необхідно провести роз'яснення який саме тип блок-схем використовується. Блок-схеми показують весь процес роботи системи з підсистемами та частково доказують правильність вибраних проектних рішень. Тому від точності і детальної блок-схеми залежить результат всієї програми.

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

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

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

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

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		40

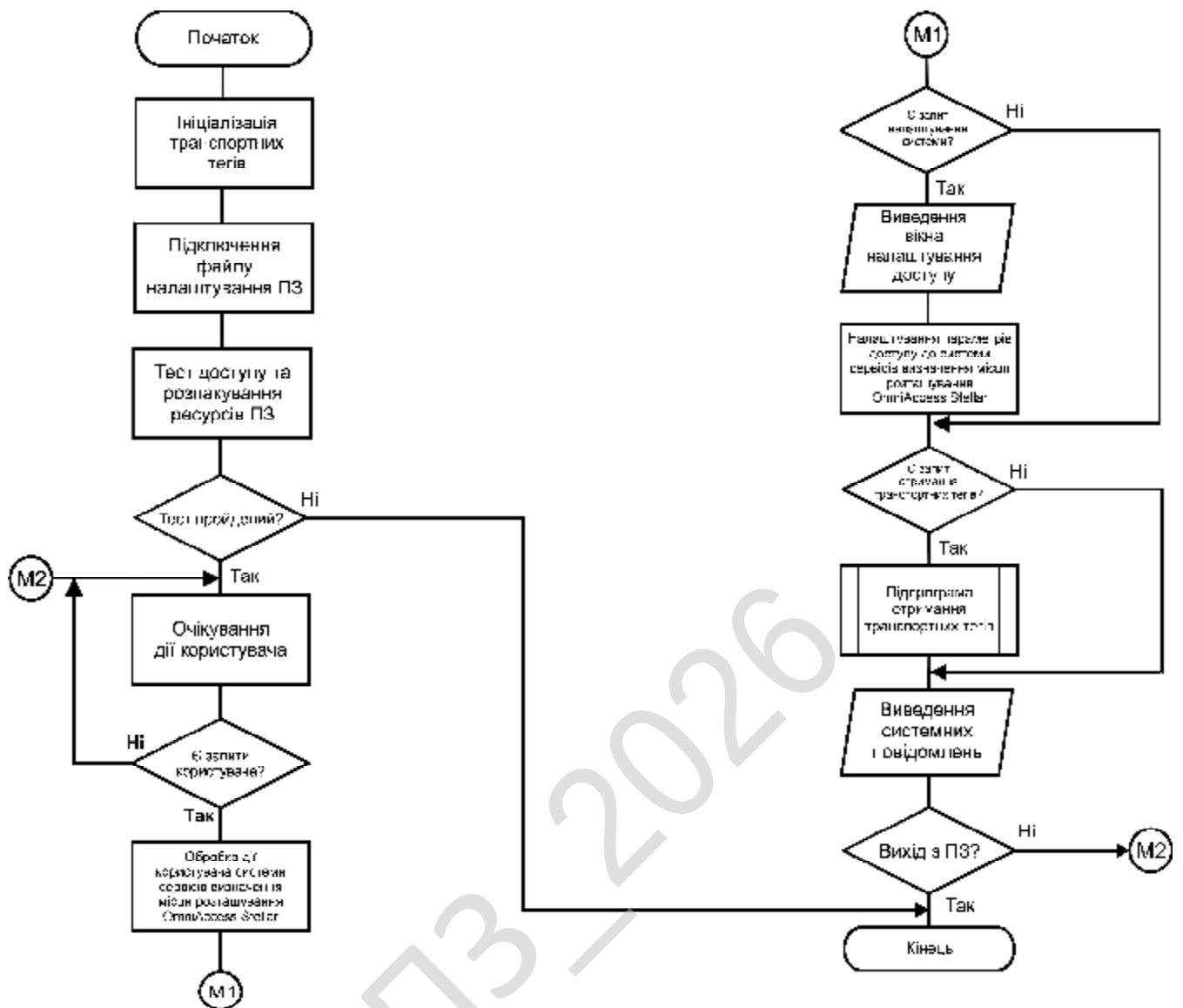


Рисунок 4.1 – Блок-схема основної програми

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

UML не є мовою програмування, але в засобах виконання UML-моделей як інтерпретованого коду можлива кодогенерація.

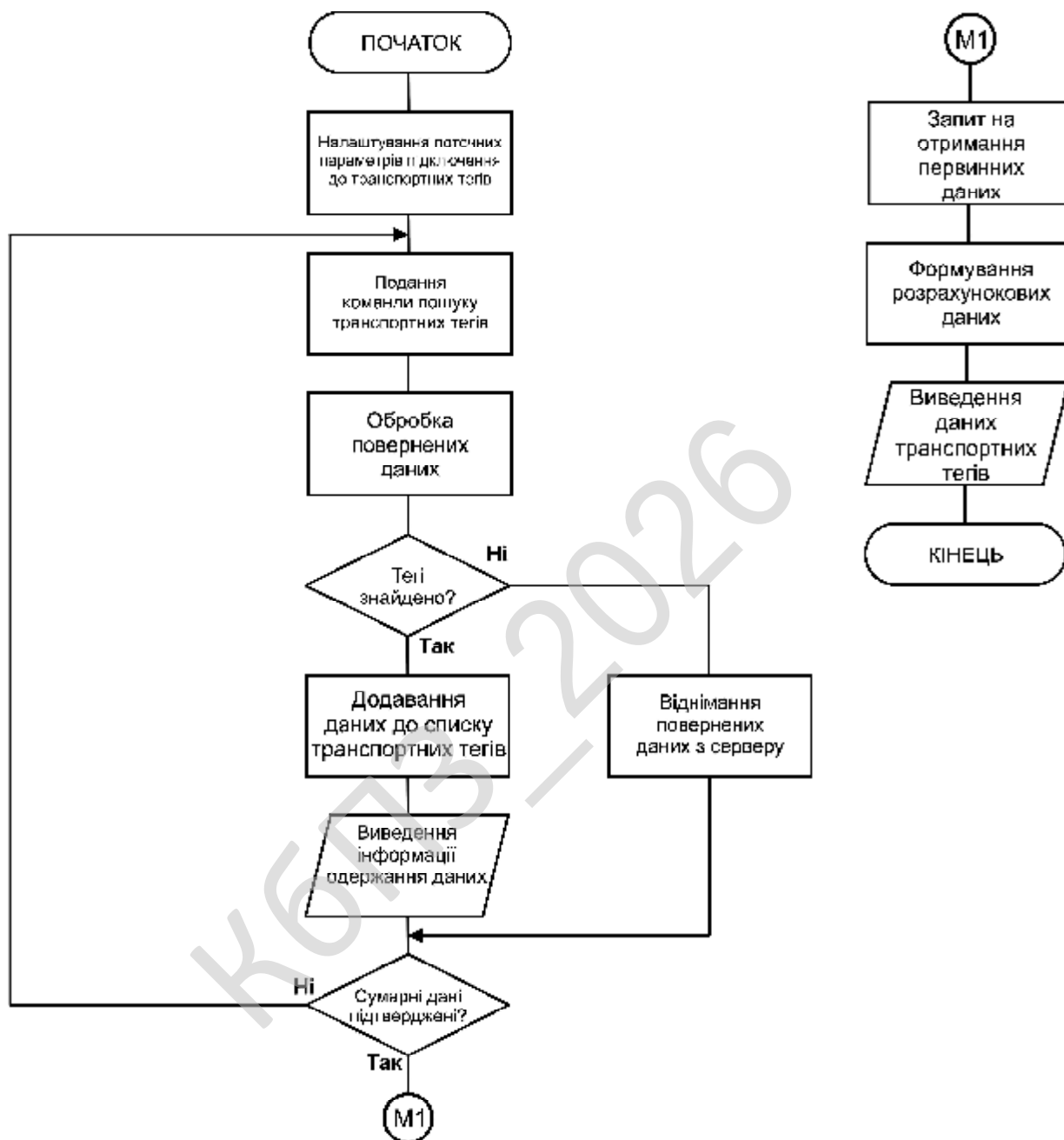


Рисунок 4.2 – Блок-схема роботи підпрограми

Було використано наступні підходи UML: діаграма діяльності (діаграми поведінки типу); діаграма прецедентів (діаграми поведінки типу); Діаграма класів; Діаграма компонент; Діаграма об'єктів; Діаграма розгортання.

Діаграма діяльності. Це візуальне представлення графу діяльностей. Граф діяльностей є різновидом графу станів скінченного автомату, вершинами якого є певні дії, а переходи відбуваються по завершенню дій. Дія є фундаментальною одиницею визначення поведінки в специфікації. Дія отримує множину вхідних сигналів, та перетворює їх на множину вихідних сигналів.

Одна із цих множин, або обидві водночас, можуть бути порожніми. Виконання дії відповідає виконанню окремої дії. Подібно до цього, виконання діяльності є виконанням окремої діяльності, буквально, включно із виконанням тих дій, що містяться в діяльності. Кожна дія в діяльності може виконуватись один, два, або більше разів під час одного виконання діяльності. Щонайменше, дії мають отримувати дані, перетворювати їх та тестувати, деякі дії можуть вимагати певної послідовності.

Специфікація діяльності (на вищих рівнях сумісності) може дозволяти виконання декількох (логічних) потоків, та існування механізмів синхронізації для гарантування виконання дій у правильному порядку.

Діаграма прецедентів це діаграма, на якій зображено відношення між акторами та прецедентами в системі. Також, перекладається як діаграма варіантів використання.

Діаграма прецедентів є графом, що складається з множини акторів, прецедентів (варіантів використання) обмежених границею системи (прямокутник), асоціацій між акторами та прецедентами, відношень серед прецедентів, та відношень узагальнення між акторами. Діаграми прецедентів відображають елементи моделі варіантів використання.

Суть даної діаграми полягає в наступному: проєктована система представляється у вигляді безлічі сутностей чи акторів, що взаємодіють із системою за допомогою так званих варіантів використання. Варіант використання (use case) використовують для описання послуг, які система надає актору. Іншими словами, кожен варіант використання визначає деякий набір дій, який виконує система при діалозі з актором.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		43

При цьому нічого не говориться про те, яким чином буде реалізована взаємодія акторів із системою.

У мові UML є кілька стандартних видів відношень між акторами і варіантами використання:

- асоціації (association relationship);
- включення (include relationship);
- розширення (extend relationship);
- узагальнення (generalization relationship).

При цьому загальні властивості варіантів використання можуть бути представлені трьома різними способами, а саме – за допомогою відношень включення, розширення і узагальнення.

Відношення асоціації – одне з фундаментальних понять у мові UML і в тій чи іншій мірі використовується при побудові всіх графічних моделей систем у формі канонічних діаграм.

Включення (include) у мові UML – це різновид відношення залежності між базовим варіантом використання і його спеціальним випадком. При цьому відношенням залежності (dependency) є таке відношення між двома елементами моделі, при якому зміна одного елемента (незалежного) приводить до зміни іншого елемента (залежного).

Відношення розширення (extend) визначає взаємозв'язок базового варіанта використання з іншим варіантом використання, функціональна поведінка якого задіюється базовим не завжди, а тільки при виконанні додаткових умов.

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

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

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

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

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

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

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

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

Вказуючи кратність на одному кінці асоціації, ви тим самим говорите, що на цьому кінці саме стільки об'єктів повинно відповідати кожному об'єкту на протилежному кінці. Кратність можна задати рівною одиниці (1), можна вказати

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		45

діапазон: "нуль або одиниця" (0..1), "багато" (0 .. *), "одиниця або більше" (1 .. *). Дозволяється також вказувати певне число (наприклад, 3). За допомогою списку можна задати і більш складні кратності, наприклад 0. . 1, 3..4, 6 .. *, що означає "будь-яке число об'єктів, крім 2 і 5".

Агрегація це проста асоціація між двома класами відображає структурний відношення між рівноправними сутностями, коли обидва класу знаходяться на одному концептуальному рівні і ні один не є більш важливим, ніж інший. Але іноді доводиться моделювати відношення типу «частина/ціле», в якому один з класів має більш високий ранг (ціле) і складається з декількох менших за рангом (частин).

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

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

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

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

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

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

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		46

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

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

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

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

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

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

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

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

Діаграма розгортання (deployment diagram) це діаграма в UML, на якій відображаються обчислювальні вузли під час роботи програми, компоненти, та

об'єкти, що виконуються на цих вузлах. Компоненти відповідають представленню робочих екземплярів одиниць коду. Компоненти, що не мають представлення під час роботи програми на таких діаграмах не відображаються; натомість, їх можна відобразити на діаграмах компонент. Діаграма розгортання відображає робочі екземпляри компонент, а діаграма компонент, натомість, відображає зв'язки між типами компонент.

Система керування базами даних (СКБД, Database Management System, DBMS) – набір взаємопов'язаних даних (база даних) і програм для доступу до цих даних. Надає можливості створення, збереження, оновлення та пошуку інформації в базах даних з контролем доступу до даних.

Першим поколінням СКБД прийнято вважати ієрархічні й мережеві системи. Ці системи отримали широке поширення в 1970-х роках, а першою комерційною системою цього типу була система IMS компанії IBM.

У 1980-х роках ці системи були витіснені системами другого покоління – повсюдно використовуваними і донині реляційними СКБД. У цих системах використовувалися непроцедурні мови управління даними (SQL) і передбачався значний ступінь незалежності даних. Реляційні системи внесли значні удосконалення в управління даними: графічний користувацький інтерфейс (GUI), клієнт-серверні застосунки, розподілені бази даних, паралельний пошук даних та інтелектуальний аналіз даних.

Але вже до кінця 1980-х років існуюча тоді реляційна модель перестала задовольняти розробників в силу низки обмежень. Відповіддю на зростаючу складність програм баз даних стали два нових напрямки розвитку СКБД: об'єктно-орієнтовані СКБД і об'єктно-реляційні СКБД.

У 1991 був утворений консорціум ODMG, основною метою якого стало вироблення промислового стандарту об'єктно-орієнтованих баз даних. Між 1993 та 2001 роками ODMG опублікувала п'ять ревізій своїх специфікацій. Остання версія стандарту має індекс 3.0, після чого група розпустилася. До кінця 1990-х існувало близько десяти компаній, що виробляли комерційні продукти, що

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		48

позиціонуються на ринку як ООСКБД. Найбільш відомими системами даного класу стали Objectivity, Versant виробництва однойменних компаній, а також СКБД Jasmine, випущена компанією CA. Незважаючи на переваги, що дозволяють ефективніше вирішувати певний ряд завдань, об'єктно-орієнтовані системи так і не змогли завоювати значущу частку ринку СКБД, залишившись «нішевим» продуктом.

Постачальниками традиційних реляційних СКБД також була проведена значна робота з об'єднання об'єктно-орієнтованих і реляційних систем. Розробники постаралися розширити мову SQL, щоб включити в неї концепції об'єктно-орієнтованого підходу, зберігаючи переваги реляційної моделі (об'єктні розширення мови SQL були зафіксовані в стандарті SQL:1999).

Основний принцип – це еволюційний розвиток можливостей СКБД без поломки попередніх підходів та зі збереженням наступності з системами попереднього покоління.

Поняття СКБД третього покоління, якими, власне кажучи, і є об'єктно-реляційні СКБД, з'явилося після опублікування групою відомих фахівців в області баз даних «Маніфесту систем баз даних третього покоління». Основні принципи СКБД третього покоління, позначені в маніфесті:

Крім традиційних послуг з управління даними, СКБД третього покоління повинні забезпечити підтримку розвиненіших структур об'єктів і правил. Розвинутіша структура об'єктів характеризує засоби, необхідні для зберігання і маніпулювання нетрадиційними елементами даних (тексти, просторові дані, мультимедіа).

СКБД третього покоління повинні включити в себе СКБД другого покоління. Системи другого покоління внесли вирішальний вклад у двох областях – непроцедурний доступ за допомогою мови запитів SQL і незалежність даних. Ці досягнення обов'язково повинні враховуватися в системах третього покоління.

СКБД третього покоління повинні бути відкриті для інших підсистем. Це включає оснащення різноманітними інструментами підтримки прийняття рішень, доступом з багатьох мов програмування, інтерфейсами до існуючих популярних систем і бізнес-застосунків, можливістю запуску програм з бази даних на іншій машині і розподілені СКБД. Весь набір інструментів і СКБД має ефективно функціонувати на різноманітних апаратних платформах з різними операційними системами.

Крім того, СКБД, що розраховує на широку сферу застосування, повинна бути оснащена мовою четвертого покоління (4GL).

У середині 1990 років було лише кілька дослідних прототипів СКБД, які поєднали найкращі риси реляційних і об'єктно-орієнтованих СКБД. Першим комерційним продуктом, якому були властиві об'єктно-реляційні риси, став Universal Server компанії Informix(згодом була поглинена IBM). В даний час більшість цих ідей вже втілено в реальних комерційних рішеннях, в тому числі і в продуктах основних постачальників СКБД (Oracle Database і IBM DB2).

Розвиток індустрії систем керування базами даних базується на значних фундаментальних наукових дослідженнях. Найчастіше, між самими дослідженнями та їхньою конкретною реалізацією в прикладних рішеннях минають роки, а іноді й десятиліття. Роботу в області управління даними проводять як університетські дослідницькі групи (MIT, Berkeley), так і центри розробок основних постачальників СКБД (Oracle, IBM, Microsoft). Інвестування в управління даними – це довгострокове, і разом з тим, вигідне вкладення коштів. В даний час дослідники мають у своєму розпорядженні засоби, що дозволяють ефективно реалізувати найскладніші запити, що маніпулюють терабайтами й петабайтами різних даних.

Основними тенденціями, які дали привід для проведення різних масштабних досліджень в області баз даних стали:

Експонентний ріст даних. Обсяг даних, у тому числі синтетичних, що генеруються автоматизованими системами, значно зріс. Збільшилося і число

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		50

прикладних областей, в яких вимагається обробка великих обсягів даних. До таких областей тепер відносяться не тільки традиційні корпоративні програми, пошук у веб, але також і наукові дослідження, обробка природних мов, аналіз соціальних мереж тощо.

Значне ускладнення структур використовуваних даних. Прості види даних у вигляді чисел і символьних рядків стали доповнятися численною мультимедійною інформацією, просторовими, процедурними даними та великою кількістю інших складних форматів.

Широке поширення дешевих високопродуктивних апаратних засобів. Щорічно ми спостерігаємо зростання обчислювальних можливостей мікропроцесорів, збільшення ємності і зниження вартості доступних і зручних в експлуатації пристроїв дискової оперативної пам'яті.

Активний розвиток засобів комунікації та «всесвітньої павутини» World Wide Web. WWW стає єдиним інформаційним середовищем, що пронизує весь світ і об'єднує величезне число користувачів та електронних пристроїв.

Поява нових важливих областей застосування СКБД. У першу чергу, це пов'язано з інтелектуальним аналізом даних, сховищами даних, а останнім часом – з паралельними обчисленнями і хмарними технологіями.

Основні характеристики СКБД

1. Контроль за надлишковістю даних.
2. Несуперечливість даних.
3. Підтримка цілісності бази даних (коректність та несуперечливість).
4. Цілісність описується за допомогою обмежень.
5. Незалежність прикладних програм від даних.
6. Спільне використання даних.
7. Підвищений рівень безпеки.

Можливості СКБД

1. Дозволяється створювати БД (здійснюється за допомогою мови визначення даних DDL (Data Definition Language)).

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		51

2. Дозволяється додавання, оновлення, видалення та читання інформації з БД (за допомогою мови маніпулювання даними DML, яку часто називають мовою запитів)

3. Можна надавати контрольований доступ до БД за допомогою: Системи забезпечення захисту, яка запобігає несанкціонованому доступу до БД; Системи керування паралельною роботою прикладних програм, яка контролює процеси спільного доступу до БД; Система відновлення – дозволяє відновлювати БД до попереднього несуперечливого стану, що був порушений в результаті збою апаратного або програмного забезпечення.

Основні компоненти середовища СКБД

1. Апаратне забезпечення.
2. Програмне забезпечення.
3. Дані.
4. Процедури – інструкції та правила, які повинні враховуватись при проектуванні та використанні БД.
5. Користувачі: адміністратори даних (керування даними, проектування БД, розробка алгоритмів, процедур) та БД (фізичне проектування, відповідальність за безпеку та цілісність даних); розробники БД; прикладні програмісти; кінцеві користувачі.

Архітектура СКБД

Існує трирівнева система організації СКБД ANSI-SPARC, при якій існує незалежний рівень для ізоляції програми від особливостей представлення даних на нижчому рівні.

Рівні:

1. Зовнішній – представлення БД з точки зору користувача.
2. Концептуальний – узагальнене представлення БД, описує які дані зберігаються в БД і зв'язки між ними. Підтримує зовнішні представлення, підтримується внутрішнім рівнем.
3. Внутрішній – фізичне представлення БД в комп'ютері.

Логічна незалежність – повна захищеність зовнішніх моделей від змін, що вносяться в концептуальну модель.

Фізична незалежність – захищеність концептуальної моделі від змін, які вносяться у внутрішню модель.

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

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

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

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

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

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

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

Кілька процесів або потоків можуть одночасно без жодних проблем читати дані з однієї бази. Запис в базу можна здійснити тільки в тому випадку, коли

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		53

жодних інших запитів у цей час не обслуговується; інакше спроба запису закінчується невдачею, і в програму повертається код помилки. Іншим варіантом розвитку подій є автоматичне повторення спроб запису протягом заданого інтервалу часу.

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

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

Особливості SQLite:

1. Транзакції атомарні, послідовні, ізольовані, і міцні (ACID) навіть після збоїв системи і збоїв живлення.
2. Встановлення без конфігурації – не потребує ані установки, ані адміністрування.
3. Реалізує значну частину стандарту SQL92.
4. База даних зберігається в одному крос-платформовому файлі на диску.
5. Підтримка терабайтних розмірів баз даних і гігабайтного розміру рядків і BLOBів.
6. Малий розмір коду: менше ніж 350KB повністю налаштований, і менш 200KB з опущеними додатковими функціями.
7. Швидший за популярні рушії клієнт-серверних баз даних для найпоширеніших операцій.
8. Простий, легкий у використанні API.
9. Написана в ANSI C, включена прив'язка до TCL; доступні також прив'язки для десятків інших мов.
10. Добре прокоментований сирцевий код зі 100% тестовий покриттям гілок.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		54

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

12. Автономність: немає зовнішніх залежностей.

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

14. Сирці перебувають в суспільному надбанні.

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

Інструменти створення та обслуговування БД

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

Технології, що підтримують SQLite та мови програмування

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

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

Web-інструментарії

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

Багато програм підтримують SQLite як формат зберігання даних, зокрема:

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

2. Gajim – SQLite використовується для зберігання історії контактів.

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

4. Banshee.

5. F-Spot.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		55

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

7. Google Chrome.

8. Google Gears.

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

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

Redmine – вільне серверне ПЗ для управління проектами та відстежування помилок. До системи входить календар-планувальник та діаграми Ганта для візуального представлення ходу робіт за проектом та строків виконання. Redmine написано на мові Ruby і є ПЗ розробленим з використанням відомого веб-фреймворку Ruby on Rails, що означає легкість в розгортанні системи та її адаптації під конкретні вимоги. Для кожного проекту можна вести свої вікі та форуми.

Функціональні можливості:

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

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		56

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

– Створення записів про помилки на основі отриманих листів

– Підтримка LDAP автентифікації;

– Можливість самореєстрації нових користувачів;

– Багатомовний інтерфейс (у тому числі українська мова);

– Підтримка СКБД: MySQL, PostgreSQL, SQLite.

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

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

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

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

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

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		57

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

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

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

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

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

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

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

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

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

Типовий життєвий цикл дефекту:

1. Новий – дефект зареєстрований тестувальником.
2. Призначений – призначений відповідальний за виправлення дефекту.
3. Дозволений – дефект переходить назад у сферу відповідальності тестувальника. Як правило, супроводжується резолюцією, наприклад:
 - Виправлено (виправлення включені у версію таку-то).
 - Дубль (повторює дефект, що вже знаходиться в роботі).
 - Не виправлено (працює відповідно до специфікації, має занадто низький пріоритет, виправлення відкладено до наступної версії тощо).
 - «В мене все працює» (запит додаткової інформації про умови, в яких дефект проявляється).
4. Далі тестувальник проводить перевірку виправлення, залежно від чого дефект або знову переходить у стан «Призначений» (якщо він описаний як виправлений, але не виправлений), або у стан «Закрито».
5. Відкрито повторно – дефект знайдено знову в іншій версії.

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

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		59

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

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

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

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

Більшість гнучких методологій націлені на мінімізацію ризиків, шляхом зведення розробки до серії коротких циклів, що мають назву ітерацій, які зазвичай тривають один-два тижні. Кожна ітерація сама по собі виглядає як програмний проект в мініатюрі, і включає всі завдання, необхідні для видачі мінімального приросту за функціональністю: планування, аналіз вимог, проектування, кодування, тестування і документування. Хоча окрема ітерація, як правило, недостатня для випуску нової версії продукту, мається на увазі те, що гнучкий програмний проект готовий до випуску наприкінці кожної ітерації. Після закінчення кожної ітерації, команда виконує переоцінку пріоритетів розробки.

Agile акцентує увагу на безпосередньому спілкуванні «віч-на-віч». Більшість agile команд розташовані в одному офісі, його іноді називають bullpen. Як мінімум вона включає і «замовників» (замовники, які визначають продукт, також це можуть бути менеджери продукту, бізнес аналітики або клієнти). Офіс може також включати тестувальників, дизайнерів інтерфейсу, технічних авторів і менеджерів.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		60

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

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

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

Основні ідеї:

- Особистості та їхні взаємодії важливіші, ніж процеси та інструменти;
- Робоче програмне забезпечення важливіше, ніж повна документація;
- Співпраця із замовником важливіша, ніж контрактні зобов'язання;
- Реакція на зміни важливіша, ніж дотримання плану.

Принципи, які роз'яснює Agile Manifesto:

- задоволення клієнта за рахунок ранньої та безперебійної поставки коштовного програмного забезпечення;
- вітання змін вимог навіть наприкінці розробки (це може підвищити конкурентоспроможність отриманого продукту);
- часта поставка робочого програмного забезпечення (кожен місяць або тиждень або ще частіше);
- тісне, щоденне спілкування замовника з розробниками впродовж всього проекту;
- проектом займаються мотивовані особистості, які забезпечені потрібними умовами роботи, підтримкою і довірою;

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		61

- рекомендований метод передачі інформації – особиста розмова;
- робоче програмне забезпечення – найкращий вимірник прогресу;
- спонсори, розробники та користувачі повинні мати можливість підтримувати постійний темп на невизначений термін;
- постійну увагу поліпшенню технічної майстерності та зручному дизайну;
- простота – мистецтво не робити зайвої роботи;
- найкращі технічні вимоги, дизайн та архітектура виходять у самоорганізованої команди;
- постійна адаптація до мінливих обставин.

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

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

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

Один з повторюваних пунктів критики: при agile-підході часто нехтують створенням «дорожньої карти» розвитку продукту, так само як і управлінням вимогами, в процесі якого і формується така «карта». Гнучкий підхід до управління вимогами не має на увазі далекосяжних планів (по суті, управління вимогами просто не існує в даній методології), а має на увазі можливість замовника раптом і несподівано наприкінці кожної ітерації виставляти нові вимоги, що часто суперечать архітектурі вже створеного і поставленого продукту. Таке іноді призводить до катастрофічних «авралів» з масовим рефакторингом і переробками практично на кожній черговій ітерації.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		62

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

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

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

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

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

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

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

5. Essential Unified Process (EssUP).

6. Екстремальне програмування (Extreme programming, XP).

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		63

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

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

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

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

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

Розглянемо формат що використовується – **JSON** (JavaScript Object Notation, укр. запис об'єктів JavaScript, вимовляється джейсон) – це текстовий формат обміну даними між комп'ютерами.

JSON базується на тексті, може бути прочитаним людиною. Формат дозволяє описувати об'єкти та інші структури даних. Цей формат головним чином використовується для передачі структурованої інформації через мережу (завдяки процесу, що називають серіалізацією). Розробив і популяризував формат Дуглас Крокфорд.

JSON знайшов своє головне призначення у написанні веб-програм, а саме при використанні технології AJAX. JSON, що використовується в AJAX, виступає як заміна XML (використовується в AJAX) під час асинхронної передачі структурованої інформації між клієнтом та сервером.

При цьому перевагою JSON перед XML є те, що він дозволяє складні структури в атрибутах, займає менше місця і прямо інтерпретується за допомогою JavaScript в об'єкти.

JSON з'явився через необхідність обміну даними з сервером у реальному часі без використання плагінів для браузерів, flash-додатків або Java-апплетів, які використовувались скрізь на початку 2000-х років. Дуглас Крокфорд був тим, хто активно просував новий на той час формат.

Він з колегами хотів створити технологію, яка використовувала б можливості звичайного браузера давала б веб-розробникам можливість створювати веб-додатки із постійним двостороннім зв'язком із веб-сервером.

JSON вперше був використаний в проекті в Communities.com для Cartoon Network, він дозволяв обмінюватись повідомленнями і одночасно маніпулювати DHTML-елементами.

Веб-сайт JSON.org було запущено 2002 року. У грудні 2005-го року Yahoo! почав переводити деякі зі своїх веб-сервісів на роботу з JSON. Google взялася до роботи з технологією у своєму веб-протоколі GData у грудні 2006-го року.

Використання

За рахунок своєї лаконічності в порівнянні з XML, формат JSON може бути більш придатним для серіалізації складних структур.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		65

Якщо говорити про веб-застосунки, в такому ключі він доречний в задачах обміну даними як між браузером і сервером (AJAX), так і між самими серверами (програмні HTTP-інтерфейси).

Формат JSON так само добре підходить для зберігання складних динамічних структур в реляційних базах даних або файловому кеші.

Синтаксис

JSON будується на двох структурах:

1. Набір пар назва/значення. У різних мовах це реалізовано як об'єкт, запис, структура, словник, хеш-таблиця, список з ключем або асоціативним масивом.

2. Впорядкований список значень. У багатьох мовах це реалізовано як масив, вектор, список, або послідовність.

Це універсальні структури даних. Теоретично всі сучасні мови програмування підтримують їх у тій чи іншій формі. Оскільки JSON використовується для обміну даними між різними мовами програмування, то є сенс будувати його на цих структурах.

У JSON використовуються такі їхні форми:

1. Об'єкт – це послідовність пар назва/значення. Об'єкт починається з символу { і закінчується символом }. Кожне значення слідує за : і пари назва/значення відділяються комами.

2. Масив – це послідовність значень. Масив починається символом [і закінчується символом]. Значення відділяються комами.

3. Значення може бути рядком в подвійних лапках, або числом, або логічними true чи false, або null, або об'єктом, або масивом. Ці структури можуть бути вкладені одна в одну.

4. Рядок – це послідовність з нуля або більше символів юнікода, обмежена подвійними лапками, з використанням ескапе-послідовностей, що починаються зі зворотної косої риски \. Символи представляються простим рядком. Тип Рядок (String) дуже схожий на String в мовах C і Java. Число теж дуже схоже на C- або

Java-число, за винятком того, що вісімкові та шістнадцяткові формати не використовуються. Пропуски можуть бути вставлені між будь-якими двома лексемами.

Використання JSON в AJAX

Наступний фрагмент коду JavaScript показує, як клієнт може використати XMLHttpRequest для запиту об'єктів в форматі JSON з серверу. (Серверна частина коду пропущена, вона просто повертає на запит URL рядок у JSON форматі).

Треба відзначити, що тут використання XMLHttpRequest не є кросс-браузерним (за деталями звертайтеся на сторінку XMLHttpRequest), і код зазнаватиме незначних модифікацій в різних версіях оглядачів Internet Explorer, Opera, Safari або Mozilla. Використання запиту XMLHttpRequest обмежено правилом одного джерела (same origin policy): URL, що відповідає на запит, має посилатися на той же сайт, що обслуговує сторінку, що ініціювала запит.

Оглядачі можуть також використовувати тег <iframe> для асинхронного запиту JSON-даних в кросс-браузерному варіанті, або використати просте перенаправлення `<form action="url_to_cgi_script" target="name_of_hidden_iframe">`. Такий підхід був поширений до приходу популярного нині запиту XMLHttpRequest.

Динамічний тег <script> також можна використати для підвантаження JSON-даних. Ця техніка можлива, щоб обійти надто суворе правило одного джерела, але вона не є безпечною. Запит JSONRequest пропонується, як безпечніша альтернатива.

Питання безпеки

Хоча JSON призначений для передачі даних в серіалізованому вигляді, його синтаксис відповідає синтаксису JavaScript і це створює низку проблем безпеки.

Часто для обробки даних, отриманих від зовнішнього джерела у форматі JSON, до них застосовується функція eval() без якої-небудь попередньої перевірки.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		67

Підробка крос-доменного запиту

Непродумане використання JSON робить сайти вразливими до підробки міжсайтових запитів (CSRF або XSRF). Оскільки тег `<script>` допускає використання джерела, що не належить до того ж домену, що і використовуваний ресурс, це дозволяє виконувати код даних, представлених у форматі JSON, в контексті довільної сторінки, що робить можливою компрометацію паролів або іншої конфіденційної інформації користувачів, що пройшли авторизацію на іншому сайті.

Це є проблемою тільки у разі вмісту в JSON-даних конфіденційної інформації, яка може бути компрометована третьою стороною і якщо сервер розраховує на політику одного джерела, блокуючи доступ до даних при виявленні зовнішнього запиту. Це не є проблемою, якщо сервер визначає допустимість запиту, надаючи дані тільки у разі його коректності. HTTP cookie не можна використовувати для визначення цього. Виключне використання HTTP cookie використовується підробкою міжсайтових запитів.

Розширення, JSONP

JSONP або «JSON з підкладкою» є розширенням JSON, коли назва функції зворотного виклику вказується як вхідний аргумент. Спочатку ідея була запропонована в блозі MacPython в 2005 році, і в наш час використовується багатьма Web 2.0 застосунками, такими, як Dojo Toolkit Applications, Google Toolkit Applications і zanox Web Services.

Подальші розширення цього протоколу були запропоновані з урахуванням введення додаткових аргументів, як, наприклад, у разі JSONPP за підтримки S3DB вебсервісів.

Оскільки JSONP використовує скрипт-теги, виклики по суті відкриті світові. З цієї причини JSONP може бути недоречними для зберігання конфіденційних даних.

Включення скриптових тегів від віддалених сайтів дозволяє їм передати будь-який контент на сайті. Якщо віддалений сайт має вразливості, які

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		69

дозволяють виконати ін'єкції JavaScript, то початковий сайт також може бути ними зачеплений.

Розширення, BSON

BSON – це бінарна форма представлення простих структур даних і асоціативних масивів (які називають об'єктами або документами). Назва «BSON» заснована на визначенні JSON і неофіційно означає «Binary JSON» (бінарний JSON).

JSON Reference

Стандарт JSON не описує посилання на інші об'єкти або частини, але існує чернетка стандарту IETF для посилань на об'єкти на основі JSON. Посилання дозволяють здійснювати трансклюзію – вставляти одні документи в інші.

JSON Reference – це JSON-об'єкт з ключем "\$ref" (всі інші ключі ігноруються) і значенням стрічкового типу що містить URI, наприклад:

```
{ "$ref": "http://example.com/example.json#/foo/bar" }
```

Якщо URI містить ідентифікатор фрагмента (в прикладі вище "/foo/bar"), він інтерпретується як JSON Pointer.

Модуль dojo.json.ref в Dojo toolkit, забезпечує підтримку декількох форм JSON Reference.

4.2 Захист розробленого програмного забезпечення

Для захисту розробленого програмного забезпечення запропоновано використовувати алгоритм Khufu. Khufu – це 64-бітовий блоковий шифр. 64-бітовий відкритий тест спочатку розщеплюється на дві 32-бітові половини, L і R . Над обома половинами й певними частинами ключа виконується операція XOR. Потім, аналогічно DES, результати проходять деяку послідовність раундів. У кожному раунді молодший значущий байт L використовується як вхід S-блоку. У кожного S-блоку 8 вхідних біт і 32 вихідних біта. Далі обраний в S-блоці 32-бітовий елемент піддається операції XOR з R . Потім L циклічно зрушується на число, кратним восьми бітам, L і R міняються місцями, і раунд завершується.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		70

Сам S-блок не статичний, він міняється кожні вісім раундів. Нарешті, по закінченні останнього раунду, над L і R виконується операція XOR з іншими частинами ключа, і половини поєднуються, утворюючи блок шифртексту.

Хоча частини ключа використовуються для операції XOR із блоком шифрування на початку й кінці виконання алгоритму, головне призначення ключа – генерація S-блоків. Ці S-блоки секретні, по суті, це частина ключа. Повний розмір ключа алгоритму Khufu дорівнює 512 біт (64 байт), алгоритм надає спосіб генерації S-блоків по ключу.

К6ПЗ_2026

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		71

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

На рисунку 5.1 зображено розроблене у бакалаврської дипломної роботі система сервісів визначення місця розташування OmniAccess Stellar. З рисунку можна побачити що інтерфейс головного вікна розподілено на наступні функціональні розділи: Функціональних кнопок ПЗ; Навігаційного меню яке викликається натисканням правої клавіші маніпулятора миші; Верхнього меню; Розділу обрання групи; Розділу виведення результату роботи системи.

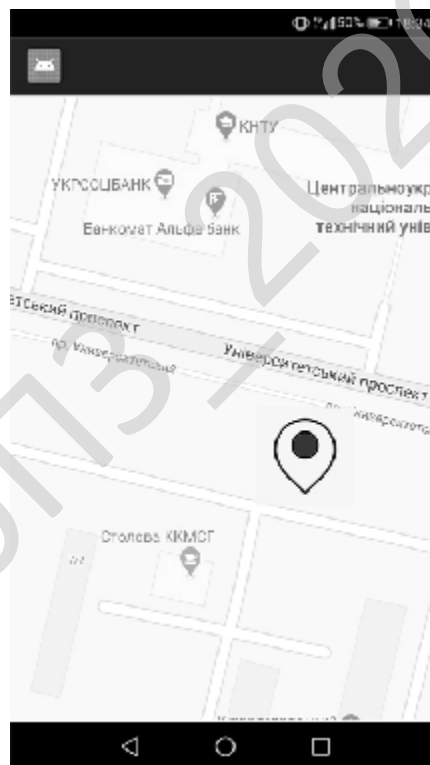


Рисунок 5.1 – Головне вікно ПЗ

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

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		72

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

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

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

До діяльностей пов'язаних із розгортанням програмного забезпечення відносять:

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

При впровадженні програмного забезпечення потрібно урахувати наступні дії:

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		73

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

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

– Виконання робіт з загальної стандартизації існуючої діяльності організації, коли виділяються кращі практики виконання процедур і включаються в ІТ рішення за принципом найбільшої корисності для більшості учасників. Відсоток таких процедур щодо загального обсягу автоматизації може бути невеликий, але це надає процесу побудови рішення вагу в організації за рахунок збільшення його необхідності.

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

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

Проводилась оцінка:

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

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

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

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

Ці тести демонструють:

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

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

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

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

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

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

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		75

Будь-який спосіб тестування «чорної скриньки» повинен:

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

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

Тестування «чорної скриньки» забезпечує пошук наступних категорій помилок:

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

Обрано умови розповсюдження – Shareware.

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

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

Основний принцип умовно-безплатного ПЗ – «спробуй, перш ніж купити» (try before you buy). ПЗ що поширюється як умовно-безплатний, надається користувачам безоплатно. Звичайно користувач платить тільки за час завантаження файлів через Інтернет або за носій (CD диск, флешку, ключ).

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		76

Протягом певного терміну, що становить зазвичай тридцять днів, він може користуватися програмою, тестувати її, освоювати її можливості.

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

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

К6ПЗ_2026

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		77

6 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти, призначено для системи сервісів визначення місця розташування OmniAccess Stellar.

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

Рішення завдання полягало у вирішенні наступних задач:

- Був проведений огляд існуючих систем сервісів визначення місця розташування OmniAccess Stellar.

- Досліджена система сервісів визначення місця розташування OmniAccess Stellar.

- На основі отриманих результатів досліджень створена програмна реалізація системи сервісів визначення місця розташування OmniAccess Stellar.

Розроблені під час виконання випускної кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання сервісів визначення місця розташування OmniAccess Stellar.

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

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

Програма реалізована на мові високого рівня Python. Дана мова програмування дозволяє найбільш ефективно обробляти дані призначені для системи сервісів визначення місця розташування OmniAccess Stellar. Це дозволило мінімізувати строк розробки програмного забезпечення, і, як слід,

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		78

зменшити витрати на його розробку. Запропоноване програмне забезпечення ділиться на загальне програмне забезпечення, що поставляється із засобами обчислювальної техніки й спеціальне програмне забезпечення, що спеціально розроблене для даної конкретної системи й включає програми, що реалізують її функції.

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

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

Для підвищення рівня безпеки запропоновано застосовувати алгоритм Khufu.

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

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		79

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. JJ Geewax. API Design Patterns. Manning Publications Co. 2021. 481 p.
2. Prateek Prasad. App Design Apprentice. Razeware LLC. 2020. 272 p.
3. Dawn Griffiths, David Griffiths. Head First Android Development. O'Reilly Media, Inc. 2021. 1414 p.
4. Nathan Metzler. Kotlin Programming for Beginners. Independently published. 2021. 158 p.
5. Aaron Torres. Go Programming Cookbook Second Edition. Packt Publishing Ltd. 2019. 427 p.
6. Мелешко Є.В., Якименко М.С., Полішук Л.І. Алгоритми та структури даних: Навчальний посібник для студентів технічних спеціальностей денної та заочної форми навчання. – Кропивницький: Видавець – Лисенко В.Ф., 2019. – 156 с.
7. Knuth D. The Art of Computer Programming, Vol. 1: Fundamental Algorithms, 3rd Edition 3rd Edition. – Addison-Wesley Professional, 2019. – 672 p.
8. Knuth D. Art of Computer Programming, Vol. 2: Seminumerical Algorithms 3rd Edition, Kindle Edition. – Addison-Wesley Professional, 2019. – 672 p.
9. Knuth D. The Art of Computer Programming: Vol. 3: Sorting and Searching 2nd Edition, Kindle Edition. – Addison-Wesley Professional, 2019. – 800 p.
10. Cormen T.H., Leiserson C.E., Rivest R.L., Stein C. Introduction to Algorithms, 3rd Edition (The MIT Press) 3rd Edition – The MIT Press, 2019. – 1292 p.
11. Вінтенко Б.Ю., Миронець І.В., Смірнов О.А. «Модель комп'ютерно-орієнтованої процедури системи підтримки оперативного персоналу АЕС». IV Міжнародна науково-практична Інтернет-конференція «Інновації та перспективні шляхи розвитку інформаційних технологій (ІПШРІТ-2025)» м.Черкаси 25 листопада 2025 року – Черкаси: ЧДТУ.– 2025. – С.101-103.

12. Kuznetsov O., Frontoni E., Kuznetsova Y., Chevardin V., Smirnov O. «Architectural foundations for adaptive security in edge computing systems». *Cybersecurity Defensive Walls in Edge Computing*, 2025. pp. 21-61.

13. Вінтенко, Б.Ю., Миронець, І.В., Смірнов, О.А., Коваленко, О.В., Усік, П.С., Буравченко, К.О., Лисенко, І.А. «Логіко-структурна модель комп'ютерно-орієнтованої процедури системи підтримки оперативного персоналу АЕС». *Кібербезпека: освіта, наука, техніка*. 2025. Том 2 № 30. С. 413-427, 2025.

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

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

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

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

18. Вінтенко Б.Ю., Смірнов О.А., Миронець І.В., Смірнова Т.В. «Методи забезпечення відмовостійкості інтелектуальних систем підтримки

оператора». *VIII міжнародна науково-практична конференція “Інформаційна безпека та комп’ютерні технології”*, м. Кропивницький. 24-25 квітня 2025 р. – Кропивницький: ЦНТУ. – 2025. – С. 44-46.

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

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

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

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

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

24. 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.

					ВКРБ-123.26.0043.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		82

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

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

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

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

29. Вінтенко Б.Ю., Смірнов О.А., Коваленко А.С., Смірнов С.А., Буравченко К.О. «Дослідження вимог міжнародних стандартів IEC60880 та IEC62138 з розробки програмного забезпечення інформаційно-керуючих систем АЕС, важливих для безпеки». *Системи управління, навігації та зв'язку*, 2023, вип. 3(73), С. 155-166.

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

31. 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.

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

33. Вінтенко Б.Ю., Смірнов О.А., Коваленко О.В., Смірнов С.А. «Дослідження нормативної документації та стандартів розробки програмного забезпечення комп'ютерних систем управління АЕС, важливих для безпеки». VI міжнародна науково-практична конференція “Інформаційна безпека та комп'ютерні технології”, м. Кропивницький. 20-21 квітня 2023 р. – Кропивницький: ЦНТУ. – 2023. – С. 35-36.

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

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

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

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

38. 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.

39. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova T. «Non-binary constant weight coding technique». CEUR Workshop Proceedings. Volume 2740, 2020, Pages 102-114.
40. 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.
41. 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.
42. Smirnov, O., Drieieva, H., Drieiev, O., Simakhin, V., Bondar, S., Odarchenko, R. «Managing multifractal properties of the binary sequence generated with the Markov chains», CEUR Workshop Proceedings Volume 2608, 2020, Pages 633-645.
43. Smirnov O. Kuznetsov A., Zaichenko Yu., Pastukhov M., Oleshko O., Kuznetsova K., «Formation of Discrete Signals with Special Correlation Properties». International Conference on Information and Telecommunication Technologies and Radio Electronics, UkrMiCo 2019; Odessa; Ukraine; 9-13 September 2019. P.22-28.
44. 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.
45. 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.

46. 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.

47. 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.

48. 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.

49. 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.

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

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

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

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

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

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

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

57. O. Smirnov, O. Kovalenko, A. Kovalenko, S. Smirnov, V. Vialkova. The mathematical model of the testing technology for DOM XSS vulnerabilities. Scientific & practical cyber security journal (SPCSJ) Vol 2 Issue 1, 22-28 pp. [Электронный Журнал]. Georgia. Tbilisi: SCSA – 2018.

58. Oleksii Smirnov, Oleksandr Kovalenko, Jamil Al-Azzeh, Anna Kovalenko, Serhii Smirnov. Qualitative risk analysis of software development. Asian Journal of Information Technology. – Volume 17(3). – Medwell Journals. – 2018. – P. 218-230.

59. Смірнов О.А., Коваленко О.В., Коваленко А.С., Смірнов С.А. Розробка методу передтестової компіляції й розподілу доступу. Збірник наукових праць ІІ міжнародної науково-практичної конференції “Інформаційна безпека та комп’ютерні технології”, м. Кропивницький. 19-20 квітня 2018р. – Кропивницький: ЦНТУ. – 2018. – С. 214-215