МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВОЛОДИМИРА ДАЛЯ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ЕЛЕКТРОНІКИ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК ТА ІНЖЕНЕРІЇ

УДК 004.032.2

До захисту допускається
В.о. завідувача кафедри
комп'ютерних наук та інженерії
д.т.н., проф. Рязанцев О. І.
_____
«____»_____ 2021 р.

МАГІСТЕРСЬКА РОБОТА

НА ТЕМУ:

«Комплексна тема. Технологія об'єднання даних для систем віддаленого біометричного моніторингу»

Освітньо-кваліфікаційний рівень «Магістр»

Спеціальність 123 – «Комп'ютерна інженерія»

Науковий керівник роботи:	(підпис)	Білобородова Т.О. (ініціали, прізвище)
Консультант з охорони праці:	(підпис)	Критська Я.О. (ініціали, прізвище)
Студент:	(підпис)	Бережний О.В. (ініціали, прізвище)
Група:		КІ-19дм

Сєвєродонецьк 2021

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Факультет Інформаційних технологій та електроніки
Кафедра Комп'ютерних наук та інженерії
Освітньо-кваліфікаційний рівень Магістр

Спеціальність 123 – «Комп'ютерна інженерія»
(шифр і назва)

«ЗАТВЕРДЖУЮ»

Т.в.о. завідувача кафедри
комп'ютерних наук та інженерії
к.т.н., доц. Кардашук В. С.

"____" _____ 2020 року

З А В Д А Н Н Я
НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ

Бережному Олександру Віталійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи «Комплексна тема. Технологія об'єднання даних для систем віддаленого біометричного моніторингу»
керівник проекту (роботи) к.т.н. Білобородова Тетяна Олександрівна
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
затверджені наказом вищого навчального закладу від «05» жовтня 2020р. № 140/15.15
2. Строк подання магістром роботи 11.01.2021 р.
3. Вихідні дані до роботи Матеріали науково-дослідної практики

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) а) аналіз існуючих технологій, методів та підходів об'єднання даних, особливості їх застосування для систем віддаленого біометричного моніторингу;
б) дослідження сучасних методів, технологій, засобів отримання, обробки, аналізу та об'єднання даних в системах віддаленого біометричного моніторингу;
в) розробка технології об'єднання даних моніторингу біометричних параметрів людини;
г) практична реалізація запропонованого методу для визначення стану людини.
д) охорона праці в галузі.
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень):
електронні плакати.

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	ст. викл. Критська Я.О.		

7. Дата видачі завдання 05 жовтня 2020 р.

Керівник _____
(підпис)

Завдання прийняв до виконання _____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітка
1	Огляд та аналіз вимог до роботи.	01.09.20-19.09.20	
2	Дослідження та аналіз сучасних методів, технологій, засобів отримання, обробки, аналізу та об'єднання даних в системах біометричного моніторингу	22.09.20-10.10.20	
3	Розробка архітектури системи віддаленого біометричного моніторингу та технології об'єднання даних	11.10.20-01.11.20	
4	Програмна реалізація структурних елементів системи віддаленого біометричного моніторингу	02.11.20-16.11.20	
5	Практична реалізація технології об'єднання даних	17.11.20-09.12.20	
6	Розробка заходів з охороні праці.	10.12.20-15.12.20	
7	Оформлення пояснювальної записки.	16.12.20-31.12.20	
8	Підготовка та подання магістерської роботи до захисту.	01.01.21-10.01.21	

Магістр _____ Бережний О.В.
(підпис) (прізвище та ініціали)

Науковий керівник _____ Білобородова Т. О.
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Бережний О.В. Комплексна тема. Технологія об’єднання даних для систем віддаленого біометричного моніторингу.

В роботі представлено вирішення задачі підвищення ефективності технології об’єднання даних в системах віддаленого біометричного моніторингу, за рахунок удосконалення оцінки ступеня тяжкості симптомів довготривалих патологічних станів здоров’я людини, створення та тестування технології об’єднання даних для системи віддаленого біометричного моніторингу, дослідження методів класифікації спрямованих на інтелектуальну обробку даних. Проведено аналіз методів і технологій об’єднання даних для систем віддаленого біометричного моніторингу. Досліджено процес біометричного моніторингу. Представлена практична реалізація наступних етапів: аналіз структури системи віддаленого біометричного моніторингу, методів, технологій, засобів отримання, обробки, аналізу та об’єднання даних в системах біометричного моніторингу, проектування та розробка архітектури системи віддаленого біометричного моніторингу, розробка та тестування технології об’єднання даних для системи віддаленого біометричного моніторингу, програмна реалізація структурних елементів системи віддаленого біометричного моніторингу: Виконано розробку та тестування мобільного додатку, додатку Windows та серверної частини. Ефективність методу об’єднання даних для систем віддаленого біометричного моніторингу визначена з використанням технології об’єднання даних датчиків смартфону з використанням класифікації та голосування на підставі репутації.

Рис.: 25 Табл. 8 Бібліогр.: 46.

Ключові слова: система віддаленого біометричного моніторингу, моніторинг симптомів хвороби Паркінсона, технології об’єднання даних.

АННОТАЦИЯ

Бережной А.В. Комплексная тема. Технология слияния данных для систем удаленного биометрического мониторинга.

В работе, представлено решение задачи повышения эффективности технологии объединения данных в системах удаленного биометрического мониторинга, для совершенствования оценки степени тяжести симптомов продолжительности

патологических состояний здоровья человека, создание и изучение технологий объединения данных для систем удаленного биометрического мониторинга, исследования методов классификация направленных на интеллектуальную обработку данных. Проведен анализ методов и технологий объединения данных для системы удаленного биометрического мониторинга. Исследован процесс биометрического мониторинга. Представлена практическая реализация следующих этапов: анализ структуры систем удаленного биометрического мониторинга, методов, технологий, средств получения, обработки, анализа и объединения данных в системах биометрического мониторинга, проектирование и разработка архитектурных систем удаленного биометрического мониторинга, разработка и тестирование технологий объединения данных для систем удаленного биометрического мониторинга, программная реализации структурных элементов системы удаленного биометрического мониторинга: Выполнено разработка и тестирование мобильного приложения, приложения Windows и серверной части. Эффективность метода объединения данных для системы удаленного биометрического мониторинга определена с использованием технологий объединения данных датчиков смартфона с использованием классификации и голосования на основании репутации.

Рис.: 25. Табл.: 8. Библиогр.: 46.

Ключевые слова: система удаленного биометрического мониторинга, мониторинг симптомов болезни Паркинсона, технологии объединения данных.

ABSTRACT

Berezhnyi Oleksandr. Data fusion techique for remote biomedical monitoring system.

The paper presents a solution to the problem of increasing the efficiency of data fusion technology in remote biometric monitoring systems, to improve the assessment of the severity of symptoms of the duration of pathological conditions of human health, the creation and study of data fusion technologies for remote biometric monitoring systems, the study of classification methods aimed at intelligent data processing. The analysis of methods and technologies for combining data for a remote biometric monitoring system. The process of biometric monitoring is investigated. The practical implementation of the following stages is presented: analysis of the structure of remote biometric monitoring systems, methods, technologies, means of obtaining, processing, analyzing and combining data in biometric monitoring systems, design and development of architectural systems for remote biometric monitoring, development and testing

of data fusion technologies for remote biometric systems. monitoring, software implementation of the structural elements of the remote biometric monitoring system: The development and testing of the mobile application, Windows application and server part was completed. The effectiveness of the data fusion method for a remote biometric monitoring system was determined using combining smartphone sensor data using classification and reputation-based voting.

Fig .: 25. Table: 8. Bibliography: 46.

Keywords: Remote biomedical monitoring system, monitoring of Parkinson's disease symptoms data fusion technologies.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧОК І СКОРОЧЕНЬ 10
ВСТУП 11
РОЗДІЛ 1 ТЕХНОЛОГІЇ ОБ'ЄДНАННЯ ДАНИХ В СИСТЕМАХ БІОМЕТРИЧНОГО МОНІТОРИНГУ 14
1.1 Актуальність систем моніторингу біометричних даних 14
1.2 Аналіз засобів отримання та збору біометричних даних 15
1.3 Аналіз систем моніторингу біометричних даних 18
1.3.1 Архітектура систем моніторингу біометричних даних 19
1.3.2 Система моніторингу біометричних параметрів при хворобі Паркінсона 20
1.4 Інфраструктура системи віддаленого моніторингу біометричних даних 20
1.5 Технології об'єднання багатопотокових даних 22
1.5.1 Таксономія об’єднання багатопотокових даних 22
1.5.2 Параметри роботи систем біометричного моніторингу 24
1.5.2.1 Режими роботи систем віддаленого біометричного моніторингу 25
1.5.2.2 Типи систем віддаленого моніторингу біометричних даних 25
1.6 Постановка наукової задачі та обґрунтування методики досліджень 27
1.7 Висновки до першого розділу 28
1.8 Перелік джерел посилань до першого розділу 28
РОЗДІЛ 2 РОЗРОБКА ТЕХНОЛОГІЇ ОБ'ЄДНАННЯ ДАНИХ В СИСТЕМІ ВІДДАЛЕНОГО БІОМЕТРИЧНОГО МОНІТОРИНГУ 31
2.1 Архітектура системи віддаленого моніторингу симптомів хвороби Паркінсона 31
2.2 Технологія об’єднання даних в системі віддаленого біометричного моніторингу 32
2.2.1 Класифікація даних 32
2.2.2 Голосування на основі репутації 33
2.2.3 Методи на основі репутації 33
2.2.4 Голосування більшістю 34
2.3 Моделі технології об’єднання даних 35
2.3.1 Локальна модель класифікації 35
2.3.2 Глобальна модель об’єднання даних 35

2.4 Розрахунок обчислювальної складності технології на основі об’єднання і голосування більшістю 36
2.5 Висновки до другого розділу 37
2.6 Перелік посилань до другого розділу 37
РОЗДІЛ 3. ПРОГРАМНА РЕАЛІЗЦІЯ СТРУКТУРНИХ ЕЛЕМЕНТІВ СИСТЕМИ ВІДДАЛЕНОГО БІОМЕТРИЧНОГО МОНІТОРИНГУ ТА ТЕХНОЛОГІЇ ОБ’ЄДНАННЯ ДАНИХ 38
3.1 Розробка мобільного додатку 38
3.2 Розробка додатку Windows 41
3.3 Реалізація серверної частини 43
3.3.1 Хмарне сховищ 43
3.3.2 Сервер 45
3.4 Реалізація технології злиття даних 48
3.4.1 Опис даних 48
3.4.2 Навчання локальної моделі класифікації 48
3.4.3 Глобальна модель об’єднання даних на основі репутації 51
3.4.4 Розрахунок обчислювальної складності моделі на основі об’єднання і голосування більшістю 55
3.5 Висновок до третього розділу 55
3.6 Перелік посилань до третього розділу 56
РОЗДІЛ 4 ОХОРОНА ПРАЦІ 57
4.1 Загальні питання з охорони праці 57
4.1.1 Правові та організаційні основи охорони праці 58
4.1.2 Організаційно-технічні заходи з безпеки праці 59
4.2 Аналіз стану умов праці 60
4.2.1 Вимоги до приміщень 60
4.2.2 Вимоги до організації робочого місця 61
4.2.3 Навантаження та напруженість процесу праці 62
4.3 Виробнича санітарія 63
4.3.1 Аналіз небезпечних та шкідливих факторів при виробництві (експлуатації) виробу 63
4.3.2 Пожежна безпека 64
4.4 Гігієнічні вимоги до параметрів виробничого середовища 65
4.4.1 Мікроклімат 65

4.4.2 Освітлення робочого місця.....65
4.5 Шум, вібрація та електромагнітне випромінювання 67
4.6 Розрахунок захисного заземлення 67
4.7 Екологія 70
4.8 Висновки до четвертого розділу 71
4.9 Перелік посилань до четвертого розділу.....71
ВИСНОВКИ.....73
ДОДАТОК А ЕЛЕКТРОННА ПРЕЗЕНТАЦІЯ 74
ДОДАТОК Б ПРОГРАМНА РЕАЛІЗАЦІЯ WINDOWS ДОДАТКУ.....83
ДОДАТОК В ПРОГРАМНА РЕАЛІЗАЦІЯ ДЕРЕВА РІШЕНЬ.....89

ПЕРЕЛІК УМОВНИХ ПОЗНАЧОК І СКОРОЧЕНЬ

ЕЕГ – Електроенцефалограма

ЕКГ – Електроміографія

ЕЕГ – Електроміографія

MEC – Mobile Edge Computing (мережі мобільного зв'язку)

LRM – Logistic Regression Method (метод логістичної регресії)

BCM – Bord Calculation Method (метод підрахунку Борди)

SVM – Support Vector Algorithm (алгоритм опорних векторів)

DT – Decision Tree (дерева рішень)

MVC – Model View Controller – (Контролер перегляду моделі)

ВСТУП

Застосування систем моніторингу біометричних параметрів в галузі охорони здоров’я дозволяє здійснювати моніторинг показників здоров’я, автоматичне управління лікуванням, відстеження даних про стан людини в режимі реального часу. Моніторинг біометричних даних проводиться з використанням великої кількості різноманітних датчиків та носимих пристроїв, що обумовлює різноманітність методів аналізу таких даних для виявлення корисних знань. Об’єднання даних датчиків передбачає поєднання даних від різних датчиків та джерел, так що отримана інформація має меншу невизначеність, ніж це було б можливо, коли б ці джерела використовувались окремо. Об’єднання даних датчиків включає технології, методи та підходи, метою яких є поєднання даних, інформації та знань з метою покращення якості даних, зменшення невизначеності, вилучення / видобування нових функцій, надання статистики / аналітики тощо. Аналіз технологій та структури систем віддаленого моніторингу багатопотокових біометричних даних показав, що існуючі системи все ще мають обмеження. Для більш точного та якісного контроля симптомів довготривалого захворювання система віддаленого моніторингу багатопотокових біометричних даних вимагає удосконалених технології, методів та алгоритмів об’єднання даних.

Контроль біометричних показників організму, зокрема, тремору, поза медичними закладами у пацієнтів з хворобою Паркінсона - поширена проблема віддаленого біометричного моніторингу. Точна класифікація і розпізнавання ступеня тяжкості тремору - один з перших кроків до своєчасного виявлення прогресування симптомів хвороби Паркінсона. Сучасний розвиток інформаційних технологій дозволяє здійснювати віддалений моніторинг фізичного та фізіологічного стану пацієнтів. Інтеграція цієї можливості моніторингу з розпізнаванням і класифікацією ступеня тяжкості тремору може прискорити процес його виявлення і забезпечити своєчасну зворотний зв'язок і допомогу. Для отримання загальної оцінки стану пацієнта необхідне використання декількох тестів для перевірки рухового і когнітивного стану пацієнта. Важливо використовувати ефективні технології для аналізу даних, одержуваних в результаті віддаленого біометричного моніторингу від різних датчиків, джерел. Для цього важливо мати повне уявлення про можливості та ефективності методів класифікації. Також, для розпізнавання активності в медичній сфері потрібні недорогі і прості у використанні апаратні компоненти і алгоритми, які не потребують великих обчислювальних витрат.

Незважаючи на вагомий розвиток технології біометричного моніторингу, все ще не вистачає всебічного підходу ключових конструкцій, фундаментальних принципів та методів об’єднання даних в системах віддаленого біометричного моніторингу.

Тому обґрунтованою є тема магістерської роботи, у якій вирішується **науково-прикладне завдання** розробка технології об’єднання даних в системах віддаленого біометричного моніторингу.

Об'єкт дослідження: системи віддаленого біометричного моніторингу.

Предмет дослідження: технології об’єднання даних для оцінки фізіологічних параметрів стану людини на підставі біометричних даних.

Мета і завдання дослідження. Метою дослідження є підвищення точності оцінки фізіологічних параметрів стану людини на підставі біометричних даних за рахунок розроблення технології об’єднання даних в системах віддаленого біометричного моніторингу.

Для досягнення мети дослідження необхідно вирішити такі **завдання**:

– аналіз структури системи віддаленого біометричного моніторингу, методів та технологій об’єднання даних в системах біометричного моніторингу;

– проектування та розробка архітектури системи віддаленого біометричного моніторингу;

– розробка та тестування технології об’єднання даних для системи віддаленого біометричного моніторингу;

– програмна реалізація структурних елементів системи віддаленого біометричного моніторингу: мобільний додаток, додатку Windows, серверної частини;

– практична реалізація запропонованої технології об’єднання даних;

– оцінка ефективності запропонованої технології об’єднання даних.

Методи дослідження. Проведені в роботі дослідження основані на методах інтелектуального аналізу даних, машинного навчання. Перевірка результатів дослідження ґрунтувалась на методах експерименту та порівняння, які використовувались при розробленні практичної частини дипломного проекту.

Особистий внесок здобувача полягає у розробці технології об’єднання даних в системі віддаленого біометричного моніторингу ступеня тяжкості симптомів хвороби Паркінсона, розробка архітектури системи, програмній реалізації структурних елементів системи віддаленого біометричного моніторингу: мобільний додаток, додатку Windows,

серверної частини, що дозволяють вирішити поставлені задачі. Усі основні результати отримані автором особисто.

Апробація матеріалів магістерської роботи. Основні положення, ідеї, та висновки магістерської роботи доповідалися та обговорювалися на 2020 International Conference on Information Technology & Systems (ICITS'20) (Богота, Колумбія 5-7 лютого 2020), V регіональному форумі IT-Ідея (Сєвєродонецьк, Україна, 7 грудня 2019), XXVII Міжнародній науково-практичній конференції «Інформаційні технології: наука, техніка, технологія, освіта, здоров'я» (MicroCAD) (Харків, Україна, 15-17 травня 2019 р.); III Міжнародній науково-практичній конференції «Теоретичні і прикладні аспекти комп'ютерних наук та інформаційних технологій» (TACSIT-2019) (Сєвєродонецьк, Україна, 8-9 травня 2019), III Міжнародній науково-практичній конференції «Сучасні технології в науці та освіті» (Сєвєродонецьк, Україна, 27-28 лютого 2020 р.)

Практичне значення отриманих результатів.

– запропоновано технологію об'єднання даних в системі віддаленого біометричного моніторингу для визначення ступеня тяжкості симптомів хвороби Паркінсона;

– програмна реалізація структурних елементів системи віддаленого біометричного моніторингу пройшла тестування та готова до подальшого використання;

– матеріали досліджень використано при розробці навчального курсу «Internet of Things for Healthcare Systems: Trainings» в межах міжнародного проекту «Internet of Things: Emerging Curriculum for Industry and Human Applications (ALIOT)», що фінансується за програмою ERASMUS+ (реєстраційний номер 573818-EPP-1-2016-1-UK-EPPKA2-CBHE-JP).

Публікації. За темою магістерської роботи з викладенням її результатів опубліковано наступні наукові праці: одна стаття в іноземному фаховому виданні, внесеному до наукометричної бази Scopus, одна стаття в українському науковому фаховому виданні, три тези доповіді, матеріали досліджень увійшли до навчального посібнику.

Структура та обсяг магістерської роботи. Кваліфікаційна магістерська робота складається із вступу, чотирьох розділів, висновків, переліку посилань. Загальний обсяг складає 91 сторінку, з яких основний текст на 73 сторінках, список використаних джерел із 48 найменувань та 2 додатків на 18 сторінках. Робота містить 8 таблиць, 25 рисунки.

РОЗДІЛ 1 ТЕХНОЛОГІЇ ОБ'ЄДНАННЯ ДАНИХ В СИСТЕМАХ БІОМЕТРИЧНОГО МОНІТОРИНГУ

Розвиток сучасних інформаційних технологій дозволяє використовувати сучасні досягнення в галузі охорони здоров’я для більш якісного надання медичної допомоги. Сучасні системи моніторингу біометричних параметрів стану людину зазвичай включають датчики, пристрої, програмне та апаратне забезпечення, мережеве підключення та обмін даними. Застосування новітніх досягнень інформаційних технологій в галузі охорони здоров’я забезпечує створення нових бізнес-моделей та внесення змін у робочі процеси, покращення якості медичної допомоги, профілактики та діагностики захворювань.

Моніторинг біометричних параметрів людини є важливою частиною інтегрованої системи віддаленого контролю та оцінки симптомів довготривалих захворювань, зокрема, симптомів хвороби Паркінсона. Моніторинг здоров'я і пов'язані з ним функції були в центрі уваги великої кількості досліджень за останнє десятиліття, в першу чергу через значні змін в основних принципах, процедурах реалізації, технологічних розробках і методах моделювання.

За останні кілька років в сенсорних технологіях відбулися значні зміни. Для модернізації існуючих засобів біометричного моніторингу розробляється архітектура віддалених бездротових датчиків, яка може підтримувати безліч традиційних типів датчиків. Для підтримки зростаючого числа різноманітних датчиків потрібні нові системи даних і архітектура обробки, які можуть інтегрувати різноманітні датчики.

1.1 Актуальність систем моніторингу біометричних даних

Застосування систем моніторингу біометричних параметрів в галузі охорони здоров’я дозволяє здійснювати моніторинг показників здоров’я, автоматичне управління лікуванням, відстеження даних про стан людини в режимі реального часу. Сьогодні носимі пристрої відстежують фізичний стан, фізичну активність, патологічні відхилення та координують дії користувачів. Покращення актуальності інтерпретації даних зменшує час, витрачений кінцевими користувачами на збір даних. Володіння навичками та вміннями інтерпретувати дані про стан здоров’я та стан пацієнта допоможе уникнути ускладнень хронічних захворювань, поліпшити когнітивні функції організму людини, поліпшити фізичний стан тощо.

Аналіз біометричних даних ускладняється різноманіттям засобів, що використовується для отримання та збору біометричних даних.

1.2 Аналіз засобів отримання та збору біометричних даних

Датчики є ключовими компонентами всіх приладів та вимірювальних систем. Вони широко використовуються для моніторингу фізичних параметрів людини в медицині, охороні здоров'я та інших суміжних областях. Відповідно до основних принципів роботи, біометричні датчики можна розділити на фізичні, хімічні та біологічні [1]. Класифікація датчиків за цим принципом представлена на рис.1.1.

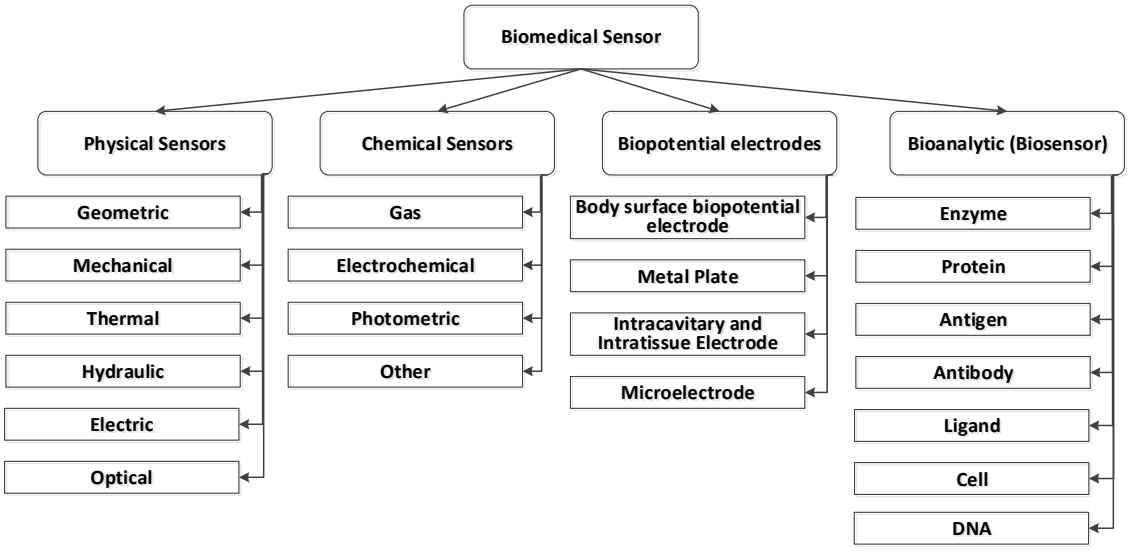


Рисунок 1.1 – Класифікація біометричних датчиків

Фізичні датчики використовуються для вимірювання артеріального тиску, температури тіла, струму та в'язкості крові, біологічного магнітного поля тощо [2]. Хімічні датчики використовуються для визначення складових елементів і концентрації рідин в організмі, наприклад, для визначення pH, концентрації глюкози в крові. Біологічні сенсори використовуються для визначення ферментів, антигену, антитіл, гормонів, ДНК, РНК, мікроорганізмів. Фізичні датчики вимірюють геометричні, механічні, теплові та гідравлічні змінні. У біометрицині використання даного типу датчика дозволяє вимірювати такі параметри, як рух м'язів, кров'яний тиск, температура тіла, кровотік тощо. Особливу роль у використанні датчиків з діагностичною метою займають датчики, що реєструють електричні явища в корпус, який зазвичай називають електродами. Найпоширенішими з

них є датчики ЕЕГ, ЕКГ, ЕМГ. Іншим прикладом фізичного датчика є оптичний датчик, який використовує світло для збору інформації.

Хімічні датчики діють на хімічні кількості, такі як ідентифікація хімічного складу, визначення концентрації різних хімічних речовин та контроль хімічної активності в організмі людини. Прикладом хімічного датчика є електрохімічні датчики, які використовуються для вимірювання концентрації хімічних речовин. Також сюди належать фотометричні датчики, які є оптичними приладами для виявлення хімічних концентрацій речовин на основі змін світлопропускання, відбиття або кольору.

Біоаналітичні датчики по суті є хімічними датчиками, але їх часто класифікують як окрему категорію датчиків. У біоаналітичних датчиках фізико-хімічні зміни біологічно активного матеріалу, що виникають внаслідок взаємодії з аналітом, перетворюються у вихідний електричний сигнал за допомогою відповідного перетворювача. Чутливі компоненти біоаналітичного датчика можуть містити ферменти, клітини, антитіла, ДНК, РНК та інші біологічно активні речовини. Відповідно до біологічно чутливого компонента біоаналітичні датчики можна розділити на п’ять класів: ферментні, мікробні, клітинні, тканинні та імунні. Основне застосування біоаналітичних датчиків у біометрицині полягає у виявленні даних клінічних та хімічних досліджень та моніторингу, моніторингу та корекції стану певних фізіологічних параметрів людини.

Існує ще одна цікава таксономія біометричних датчиків, запропонована в [3], де автори виділяють 4 типи датчиків, вони є фізіологічними датчиками, датчиками активності, що носяться, людськими датчиками та контекстними датчиками.

Фізіологічні датчики вимірюють фізіологічну статистику пацієнта або життєві показники. (наприклад, програми для iPhone для контролю пульсу).

Носимі датчики активності вимірюють атрибути активності користувачів, що відрізняються від вузьконаправлених датчиків параметрів фізичної активності людини (наприклад, акселерометри, що використовуються для контролю ходи. Виробники спортивного взуття виробляють кросівки з датчиками, здатними відстежувати ходьбу або пробіжки).

Людські датчики: означає, що ми можемо використовувати людей як інтеграторів результатів зондування. (наприклад, люди використовують веб-пошук та Twitter для створення звітів про важливі події, пов’язані зі своїм здоров’ям).

Контекстуальні датчики вбудовані в середовище навколо користувача для вимірювання різних контекстних властивостей. Прикладами є датчики виявлення руху, датчики температури, аудіо- та відеодатчики, датчики погоди тощо.

Крім того, ще однією існуючою класифікацією датчиків є їх поділ на активні та пасивні.

Активні датчики - це ті датчики, які потребують зовнішнього джерела живлення для перетворення вхідного сигналу у вихідний сигнал, тоді як пасивні датчики - це ті, які за своєю суттю забезпечують власну енергію або отримують енергію від вимірюваного явища в корисний електричний потенціал або струм.

Залежно від передбачуваного використання, біометричні датчики можуть оцінюватися за такими ключовими параметрами.

Діапазон вимірювання: Діапазон датчика відповідає мінімальним і максимальним робочим межам, які, як очікується, будуть точно виміряні датчиком.

Чутливість: Під чутливістю розуміється відношення зміни вихідного сигналу для даної зміни вхідного сигналу. Інший спосіб визначити чутливість - це знайти нахил калібрувальної лінії, що з'єднує вхідний сигнал з виходом. Висока чутливість означає, що незначна зміна вхідних даних спричиняє значну зміну виходу.

Точність (правильне вимірювання): Точність відноситься до різниці між дійсним значенням та фактичним значенням, виміряним датчиком. Класично точність виражається як відношення між попередньою різницею і справжнім значенням; він визначається як відсоток для всіх показань.

Точність (точність вимірювання): Точність відноситься до відтворюваності вимірювань за однакових умов. Високо відтворювані показники свідчать про високу точність. Точність не слід плутати з правильністю. Наприклад, вимірювання може бути дуже точним, але не обов'язково правильним.

Розподільна здатність: розподільна здатність характеризується мінімальною зміною виміряного значення, яке датчик може відчути.

Відтворюваність: відтворюваність - це здатність датчика давати однакові результати при дотриманні однакових умов. Вона визначається максимальною різницею вихідних значень датчика, отриманою двома вимірами параметрів. Зазвичай це виражається у відсотках від максимального значення вхідного сигналу. При невеликому діапазоні вимірювання відтворюваність дуже висока. Відтворюваність може залежати від діапазону вимірювань.

1.3 Аналіз систем моніторингу біометричних даних

Галузі використання вбудованих та носимих систем для біометричних застосувань можна розділити наступним чином, як показано на рис. 1.2.

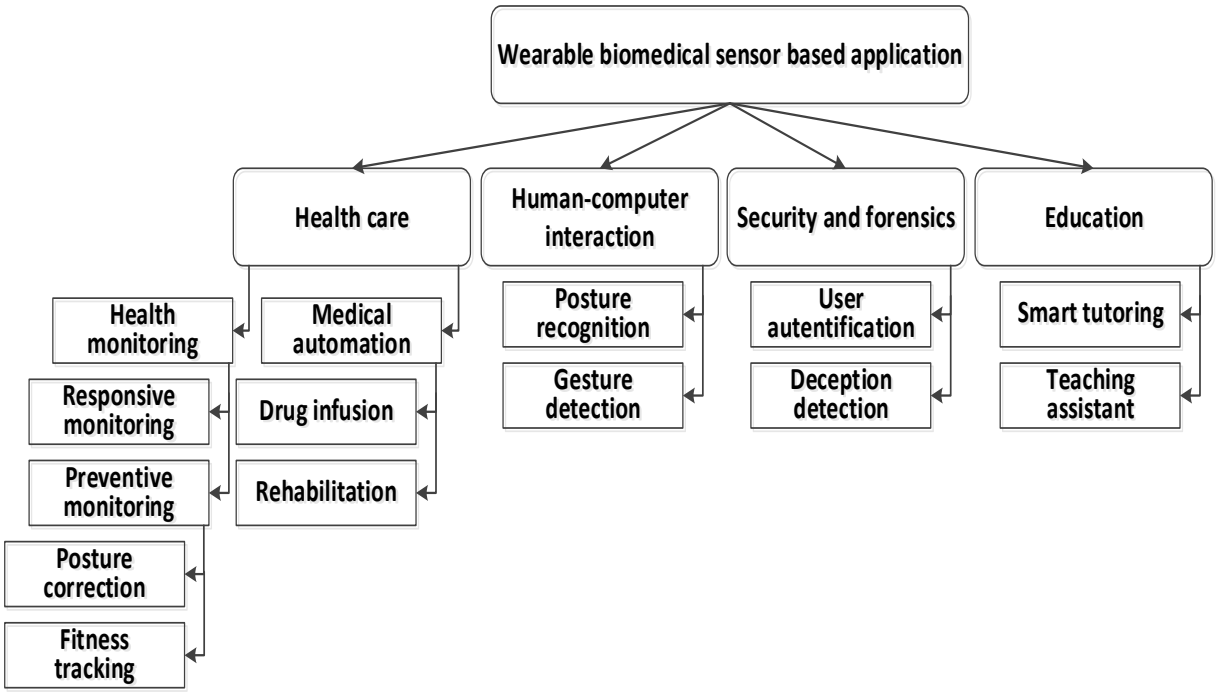


Рисунок 1.2 – Сфери застосування біометричних датчиків [4]

Біометричні системи моніторингу забезпечують моніторинг фізичного стану пацієнта для профілактики патологічних станів до появи будь-яких симптомів, які намагаються виявити патологічні зміни стану здоров'я на ранній стадії.

Системи, що намагаються виявити патологічні зміни стану здоров'я на ранніх стадіях, спрямовані на виявлення захворювань на ранній стадії шляхом моніторингу та аналізу різних біометричних параметрів, наприклад, таких як частота серцевих скорочень, рівень глюкози в крові, рівень цукру в крові, електроенцефалограма (ЕЕГ) та електрокардіограма (ЕКГ) протягом тривалого періоду часу.

Системи людського комп'ютерного інтерфейсу [5] використовуються для вилучення контексту з неявної інформації, яку отримують, збирають, обробляють носимі пристрої, що характеризує стан людини або місця, пов'язаного з мовленням, допомагає передавати ідеї один одному та реагувати відповідно [6].

Крім того, системи, розроблені в галузі інформаційної безпеки для виявлення брехні [7, 8] та автентифікації особи, можуть бути віднесені до вбудованих та носимих систем на основі датчиків для біометричних застосувань.

Ще однією областю застосування вбудованих та носимих систем на основі біометричних датчиків є розумне навчання: навчальні системи, які вибирають найкращий план навчання відповідно до відповідей студентів.

1.3.1 Архітектура систем моніторингу біометричних даних

Архітектура системи віддаленого моніторингу біометричної інформації представлена трьома основними компонентами: носимі пристрої або датчики, пристрої зберігання, обробки та передачі даних, хмарні сховища або сервери, аналітичні панелі.

Перший необхідний компонент - це різні типи носимих пристроїв та датчиків, які сприймають фізичні сигнали людського тіла, які є безпосередньо носимими пристроями.

Пристрої зберігання, обробки та передачі даних. Дані можуть передаватися безпосередньо від датчиків або за допомогою зовнішніх пристроїв з більшою обчислювальною потужністю, наприклад, смартфона. Зовнішні пристрої зазвичай виконують первинну обробку сигналів і передають дані на наступний рівень (хмарне сховище, сервери) для подальшого аналізу та довгострокового зберігання.

Смартфони нині є обов'язковою частиною повсякденного життя. Використання вбудованих сенсорів смартфона дозволить проводити моніторинг і диференціацію тремору в будь-який час в будь-якому місці з можливістю віддаленого доступу до даних моніторингу.

Третій компонент представлений хмарними сховищами, серверами. Оскільки носимі пристрої моніторингу біометричної інформації та базові станції мають обмежені ресурси, отримані дані зазвичай надсилаються на сервери, хмарне сховище для обробки та довготривалого зберігання.

Аналітичні панелі представлені додатками для персонального комп’ютеру та надають можливість віддаленого доступу до біометричних даних з метою перегляду, контролю, аналізу тощо.

1.3.2 Система моніторингу біометричних параметрів при хворобі Паркінсона

Одним з актуальних напрямів застосування систем моніторингу біометричних даних є контроль та виявлення симптомів хвороби Паркінсона. Хвороба Паркінсона - це повільно прогресуюче нейродегенеративне захворювання. Це захворювання протікає безсимптомно протягом тривалого часу. Рання діагностика дозволяє зафіксувати стан пацієнта на початкових стадіях і запобігти важким когнітивним розладам. Постійний моніторинг певних фізіологічних параметрів дозволяє відстежувати симптоми захворювання на ранній стадії. Це допомагає своєчасно вживати заходів для подовження початкової стадії та підтримки якості життя. Симптоми хвороби Паркінсона зазвичай поділяють на когнітивні та моторні розлади.

Сучасний розвиток інформаційних технологій відкриває нові можливості в біометричному моніторингу. Широкий спектр біометричних датчиків використовується для отримання різних фізіологічних сигналів людини, які можуть свідчити про наявність симптомів хвороби Паркінсона. Використання даних з носимих пристроїв, таких як сенсори смартфонів, робить можливим раннє виявлення симптомів захворювань.

Велика розмаїтість біометричних датчиків використовується для отримання даних про наявність або відсутність симптомів хвороби, ступеня їх тяжкості, типу. Акселерометр, електроміографія, гіроскоп, відео, детектор руху можуть використовуватися для вимірювання і аналізу симптомів. Сигнали обробляються і аналізуються з використанням різних методів оцінки стану здоров'я людини. Для цього використовуються вейвлет-перетворення на підставі аналізу з використанням методу ковзного вікна, аналіз головних компонентів, швидкі перетворення Фур'є, спектральний аналіз, методи машинного навчання.

1.4 Інфраструктура системи віддаленого моніторингу біометричних даних

З точки зору інфраструктури, система моніторингу біометричних даних складається з безлічі взаємопов’язаних периферійних пристроїв, мережевих вузлів та віртуальних / фізичних серверів, що відрізняються своїми можливостями. За систематизацією [9] визначено наступні параметри, що характеризують вузли: отримання / спрацьовування, інтернет-можливості, пам'ять / зберігання, переадресація та / або маршрутизація, послуги / ресурси.

Параметр отримання / спрацьовування стосується наявності датчиків і виконавчих механізмів, що забезпечують відповідні можливості. Основні функції підключення до мережі називаються інтернет-можливостями. Інші функції пов'язані з обробкою та можливостями пам'яті / зберігання. Покращені функції зв’язку стосуються функцій переадресації та / або маршрутизації, а також підтримки програмно-визначених та віртуалізованих мереж. Нарешті, можливості надання послуг / ресурсів дозволяють надавати ресурси та послуги третім особам на вимогу, орієнтовані на послуги.

Виходячи з цього набору функцій, пристрій можна розглядати як простий вузол з можливостями отримання та спрацьовування, але без будь-яких мережевих засобів, що вимагає фізичного підключення до точки доступу / шлюзу для передачі отриманих даних. Він може мати деякі (обмежені) можливості зберігання, щоб об'єднати кілька вимірювань у партії перед відправкою для мінімізації споживання енергії (що важливо для пристроїв, що живляться від акумуляторів). Прикладом можуть бути будь-які біометричні датчики та виконавчі механізми, а також сенсорні плати / екрани.

Шлюз базового зв'язку - це вузол, що відповідає за передачу даних від пристроїв до Інтернету (тобто збір значень датчика), або навпаки (тобто надсилання команд активації). Він може бути обладнаний власними датчиками та виконавчими механізмами, або їх можна підключити через специфічний (дротовий) інтерфейс до його контактів (наприклад, контактів GPIO). Сховища можуть бути присутніми на шлюзі, який, як правило, має обмежені можливості або взагалі не має можливостей обробки на борту. Прикладами можуть бути плати, оснащені лише мікроконтролером, такі як Arduino Uno та подібні.

Розумний об'єкт і розумний шлюз - це більш потужні блоки, оснащені засобами обробки, що дозволяє вбудовану (попередню) обробку даних, що надходять від власних або підключених датчиками. Прикладом можуть слугувати розумні плати та одноплатні комп’ютери з мікропроцесорами, такі як Raspberry Pi та Arduino Yun.

Блоки зв'язку та обробки - це мережеві вузли та сервери з розширеними можливостями зв’язку та обробки, але, як правило, без будь-яких засобів зондування / спрацьовування. Вони здатні підтримувати завдання обробки даних (фільтрування, аналітика, злиття даних тощо), завантажених розумними об'єктами та шлюзами, згідно з принципами обчислення Fog. Прикладами є маршрутизатори, сервери MEC (Mobile Edge Computing), базові станції та хмарні програми.

Центри обробки даних забезпечують обробку, зберігання та мережеві ресурси для підтримки систем моніторингу стану здоров’я пацієнтів, можливо, за допомогою хмарної моделі надання послуг.

Згідно з цією категоризацією, обробка та аналіз даних в системах віддаленого моніторингу може виконуватися на трьох різних рівнях: вбудованому (edge обчислення), на серверах зв'язку та обробки та / або у віддалених центрах обробки даних (хмарні обчислення).

1.5 Технології об'єднання багатопотокових даних

Об'єднання даних датчиків передбачає поєднання даних від різних датчиків та джерел, так що отримана інформація має меншу невизначеність, ніж це було б можливо, коли б ці джерела використовувались окремо [9]. Термін «зменшення невизначеності» в цьому випадку може означати більш точний, більш повний або більш надійний, або посилатися на результат нової точки зору на основі об'єднаної інформації. Об'єднання даних датчиків завжди відігравало ключову роль у системах віддаленого моніторингу.

1.5.1 Таксономія об'єднання багатопотокових даних

Об'єднання даних - це широкий загальний термін для декількох методів та підходів, метою якого є поєднання даних, інформації та знань з метою покращення якості даних, зменшення невизначеності, вилучення / видобування нових функцій, надання статистики / аналітики тощо. Визначено кілька різних моделей об'єднання даних на основі кількох параметрів. З точки зору обробки даних, визначена класифікація підходів до об'єднання даних [10] представлена у відповідності до таксономії, представленої в табл. 1.1.

З таксономії в табл. 1.1 можна виявити чіткий багатошаровий архітектурний шаблон, де окремі рівні ієрархічно будуються один на іншому.

Таблиця 1.1 - Таксономія підходів об'єднання даних

Категорія	Таксономія		
	Низький рівень	Середній рівень	Високий рівень
Стадія обробки [11]	Низька	Проміжня	Висока
Рівень абстракції [12]	Низький	Середній	Високий
Детальність даних [13]	Рівень даних	Рівень ознак	Рівень прийняття рішень
Операційні домени [14]	Часові	Просторові	Часові та просторові
Семантика	Виявлення знань	Виявлення ознак	Висновок
Відносини джерел [15]	Доповнення	Надлишковість	Співробітництво
Вимоги користувача [14]	Локальні/одиничні вузли	Регіональні	Глобальні

Концептуальну трирівневу модель об’єднання даних можна узагальнити наступним чином.

Низький рівень включає об’єднання даних, які зазвичай застосовуються до необроблених даних, що надходять із джерел, що здійснюють перший етап обробки або на низькому рівні абстракції, виконуючи локальні операції, такі як операції у часовому домені, спрямовані на виявлення знань або співпрацю з іншими вузлами. Структура об’єднання даних на низькому рівні представлена на рис. 1.3.

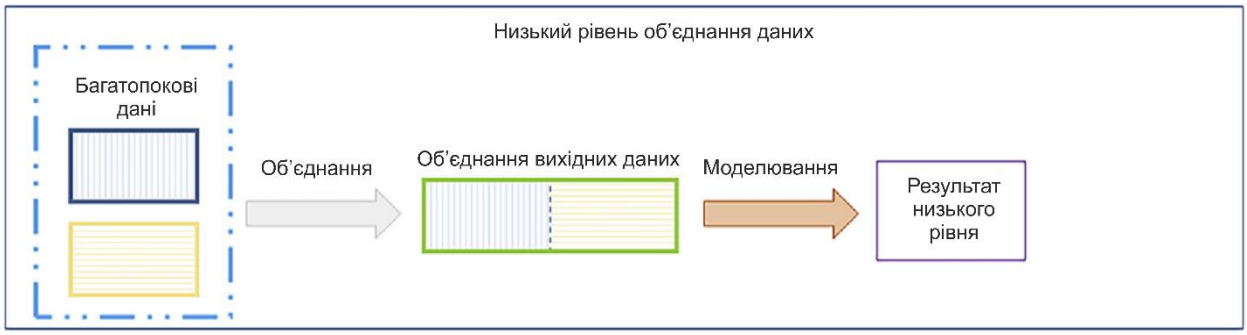


Рисунок 1.3 – Об’єднання даних на низькому рівні

Середній рівень включає функції більш високого рівня, які можуть бути виконані з попередньо обробленою інформацією, тобто результатами попереднього етапу обробки, локальним обчисленням або рівнем абстракції, наприклад для отримання оцінок просторових параметрів на даній області або надлишкові обчислення. Структура об’єднання даних на середньому рівні представлена на рис. 1.4.

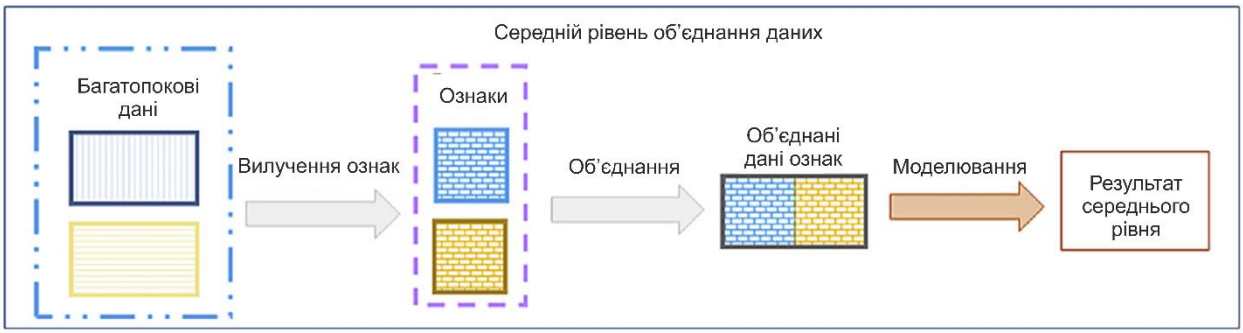


Рисунок 1.4 – Об’єднання даних на середньому рівні

Високий рівень реалізує останній етап обробки на найвищому рівні абстракції в глобальному домені, виконуючи умовивід або складні міркування та прийняття рішень щодо даних, що надходять з нижніх шарів, або здійснюючи часово-просторове об’єднання,

наприклад, використовуючи співпрацюючі моделі. Структура об’єднання даних на високому рівні представлена на рис. 1.5.

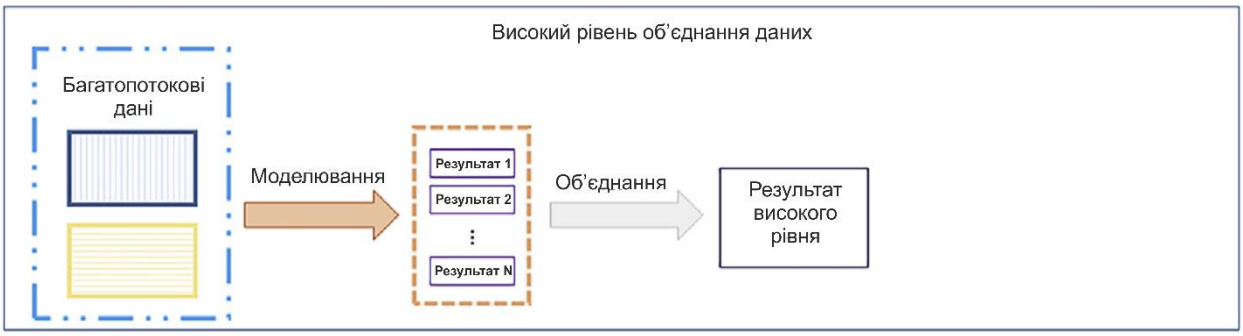


Рисунок 1.5 – Об’єднання даних на високому рівні

Підходи низького рівня приносять кілька переваг [16]. Наприклад, ними легко користуватися, оскільки вони не вимагають жодного етапу вилучення особливостей на початку створення моделі, але вони, очевидно, також мають деякі недоліки. Насправді, при застосуванні цих методологій необхідний крок масштабування блоків, спрямований на те, щоб уникнути того, що блок даних, що представляє більшу дисперсію (просто тому, що він має більшу кількість змінних), ніж інші набори. У будь-якому випадку, хоча вихідні змінні використовуються для створення моделі, ймовірно, що алгоритм передбачає процедуру вибору ознаки / змінної. Коли це управляється методами, заснованими на обчисленні прихованих змінних, може виникнути додаткова проблема. Насправді, оскільки вилучення інформації відбувається на об'єднаній глобальній матриці, низькорівневі підходи не дозволяють визначення різної кількості компонентів, які слід витягти з різних блоків даних; отже, це може бути підходящим для деяких із доступних блоків, але, ймовірно, це може відповідати завищенню / заниженню складності.

Методології середнього рівня представляють крок вилучення ознак як вихідну точку. Отже, окремі блоки даних обробляються незалежно, і інформація витягується у формі прихованих змінних або вибраних ознак, а потім об'єднується в унікальну матрицю, що використовується для подальшого мультимодального моделювання даних.

1.5.2 Параметри роботи систем біометричного моніторингу

Багатопотокові системи віддаленого моніторингу можуть працювати синхронно та асинхронно. У синхронній системі більше ніж одна біометрична техніка працює як одна

система і виконується в одному процесі авторизації. В асинхронному режимі система працює послідовно, тобто послідовно виконується більше одного біометричного методу (один за іншим) [17].

1.5.2.1 Режими роботи систем віддаленого біометричного моніторингу

З трьох рівнів об’єднання даних виходять режими роботи багатопотокових систем віддаленого біометричного моніторингу: послідовний, паралельний та ієрархічний. Режими обговорюються далі.

Послідовний режим (каскадний режим). Перед дослідженням наступної модальності в цьому режимі перевіряється кожна використовувана модальність. Від імені цього режиму можна зменшити загальну тривалість розпізнавання, перед використанням наступного способу загальна кількість можливих ідентифікаційних даних - може бути зменшена.

Паралельний режим. Цей режим використовує декілька модальностей, і дані беруться з цих модальностей і виконуються одночасно для виконання розпізнавання. Тоді для остаточного рішення ці результати є результатом, об'єднаним в єдиний результат.

Ієрархічний режим. Цей режим працює на окремих класифікаторах, і всі дані об’єднуються у формі ієрархії, як деревоподібна структура. Ієрархічний режим використовується переважно з великою кількістю класифікаторів.

1.5.2.2 Типи систем віддаленого моніторингу біометричних даних

Робота мультимодальних систем біометричного моніторингу складається з фази реєстрації та фази аутентифікації. Різні типи біометричного моніторингу класифікуються наступним чином.

– Багатоалгоритмічна система: цей тип системи працює на одному біометричному зразку, і цей зразок надходить від одного датчика, а потім обробляє цей зразок, використовуючи 2 або більше різних алгоритмів.

– Багатовимірна система: Ця біометрична система використовує 1 або більше датчиків для захоплення 2 або більше різних зразків однакових біометричних ознак.

– Багатосенсорна система: цей тип біометричної системи використовує кілька датчиків для фіксації одного і того ж екземпляра біометричної ознаки.

– Біометрична система з кількома параметрами: у цій системі на етапі зарахування збираються множинні параметри тієї самої біометричної системи.

Проектування, реалізація та підтримка систем віддаленого біометричного моніторингу багатопотокових даних вирішуються шляхом об’єднання даних. Якість проектування, реалізації та підтримки системи можуть бути визначені за допомогою визначення наступних критеріїв: точність, ефективність, надійність, застосовність, універсальність.

Застосування класифікації даних у відповідності до структур об’єднання можна визначити наступним чином.

1) Об’єднання до узгодження: означає, що перед узгодженням об’єднання відбувається на двох рівнях:

а) Рівень датчика: при цьому типі об’єднання дані, які повинні бути сумісними, або дані, які можна ідентифікувати датчиками, отримуються на цьому рівні. Але це не є обов'язковим або можливим, що ми будемо отримувати дані постійно. Система біометричного моніторингу з декількома датчиками, відбирає одні й ті ж самі зразки з різними датчиками, а потім за допомогою одного або декількох алгоритмів обробляються зразки. Видиме світло та ІЧ-камера обидві камери можуть використовуватися для розпізнавання обличчя. Це приклад системи з декількома датчиками.

б) Рівень ознак: При цьому типі об’єднання проводиться класифікація. Комбіновані вектори об’єктів, отримані за допомогою декількох датчиків та різних алгоритмів вилучення об’єктів. На цьому рівні сигнал від декількох датчиків, які є особливостями будь-якої людини, обробляється, а потім класифікується ознаками шляхом вилучення ознак із групи ознак або різних ознак.

Для зменшення небажаних даних слід отримувати різні методи, щоб отримувати лише корисні дані. Тепер різні ознаки поєднуються, щоб отримати вектор об’єкта, який буде корисним у класифікації. На цьому рівні більша точність досягається, коли різні способи функцій добре підходять один одному [18, 19].

2. Об’єднання після узгодження: означає, що після етапу узгодження об’єднання відбувається на різних рівнях:

а) Динамічний вибір класифікатора;

б) Об’єднання на рівні прийняття рішень: після вилучення функцій, виконання узгодження та розпізнавання підсистеми відбуваються різні результати і стратегії

прийняття рішень використовуються для прийняття рішення. Ці стратегії використовують булеві функції, а (AND / OR) - для остаточного рішення;

в) Об’єднання на рівні ранжування: коли результати надходять у порядку зменшення достовірності, то застосовується злиття рівня рангу. Кілька рейтингових списків подаються за біометричними даними, і ці рейтинги поєднуються, і відбувається остаточне виведення рангу. Для об’єднання на рівні рангу для кінцевого результату використовуються три методи: 1) метод логістичної регресії (LRM), 2) метод підрахунку Борди (BCM) та 3) Марковські моделі.

г) Об’єднання на рівні оцінки узгодження: На цьому рівні застосовуються різні види методів, це можуть бути (правило Байєса, LRM, вищий ранг, зважений продукт, метод підрахунку Борди та зважена сума, лінійний дискримінантний аналіз, k-найближчих сусідів та Марковські моделі тощо) [20, 21].

Об’єднання на рівні оцінки узгодження також класифікується на три різні типи: схеми, засновані на щільності (роботи над моделлю Гаусова суміші та перевірка співвідношення правдоподібності); схеми, засновані на перетворенні (працює за правилом суми); алгоритм опорних векторів (SVM).

3. Комбіноване об’єднання. На цьому рівні об’єднання застосовується до або після узгодження, тобто поєднує перший та другий типи об’єднання.

1.6 Постановка наукової задачі та обґрунтування методики досліджень

Моніторинг біометричних даних проводиться з використанням великої кількості різноманітних датчиків та носимих пристроїв, що обумовлює різноманітність методів аналізу таких даних для виявлення корисних знань. Аналіз технологій та структури систем віддаленого моніторингу багатопотокових біометричних даних показав, що існуючі системи все ще має обмеження, що визначають наступним чином. Для більш точного та якісного контроля симптомів довготривалого захворювання система віддаленого моніторингу багатопотокових біометричних даних вимагає удосконалених технології, методів та алгоритмів об’єднання даних.

В даному контексті слід визначити наступні задачі магістерської роботи:

1) аналіз структури систем віддаленого біометричного моніторингу, методів та технологій об’єднання даних в системах біометричного моніторингу

2) проектування та розробка архітектури системи віддаленого біометричного моніторингу;

3) розробка технології об’єднання даних для системи віддаленого біометричного моніторингу;

4) програмна реалізація структурних елементів системи віддаленого біометричного моніторингу: мобільний додаток, додатку Windows, серверної частини;

5) практична реалізація запропонованої технології об’єднання даних;

6) оцінка ефективності запропонованої технології об’єднання даних.

Для проведення досліджень, на підставі властивостей даних, доцільно використовувати технологію об’єднання на високому рівні об’єднання та об’єднання на рівні оцінки узгодження моделювання.

1.7 Висновки до першого розділу

Аналіз досліджень галузі об’єднання даних для систем віддаленого біометричного моніторингу дозволив сформулювати наступні висновки.

1. Для підтримки зростаючого числа різноманітних датчиків потрібні нові системи даних і архітектура обробки, які можуть інтегрувати різноманітні датчики;
2. Одним з актуальних напрямів застосування систем моніторингу біометричних даних є контроль та виявлення симптомів хвороби Паркінсона;
3. Ключову роль у системі віддаленого моніторингу відіграє об’єднання даних від різних датчиків;
4. Для проведення досліджень, на підставі властивостей даних, доцільно використовувати технологію об’єднання на високому рівні об’єднання та об’єднання на рівні оцінки узгодження моделювання;
5. Сформульована загальна науково-технічна задача дослідження, як задача об’єднання даних для систем віддаленого біометричного моніторингу.

1.8 Перелік джерел посилань до першого розділу

1. Zhou G., Wang Y., Cui L. Biomedical sensor, device and measurement systems // Advances in Bioengineering. 2015.

2. Terrance D., McGrath M. Wireless sensor networks for healthcare applications // Artech House. 2010.

3. Reddy C. K., Aggrawal C.C. Healthcare Data Analytics // Chapman and Hall/CRC. 2015.

4. Arsalan M. Addressing Security and Privacy Challenges in Internet of Things // arXiv preprint arXiv:1807.06724. 2018.

5. Dey A. K., Salber D., Abowd G. D., Futakawa M. The conference assistant: Combining context-awareness with wearable computing // IEEE Int. Symp. Wearable Computers, 1999. – P. 21–28.

6. Keyes E., Johnson M.P., Starner T. Magnetometer-based gesture sensing with a wearable device. U.S. Patent 9,141,194. Google LLC, 2015.

7. Li X., Hu B., Shen J., Xu T., Retcliffe M. Mild depression detection of college students: An EEG-based solution with free viewing tasks // J. Medical Systems, 2015. – Vol. 39 (12). – P. 1–6.

8. Meservy T. O., Jensen M. L., Kruse J., Burgoon J. K., Nunamaker J. F., Twitchell D. P., Tsechpenakis G., Metaxas D. N. Deception detection through automatic, unobtrusive analysis of nonverbal behavior // IEEE Intelligent Systems, 2005. – Vol. 20(5). – P. 36–43.

9. Dautov R., Distefano S., Buyya, R. Hierarchical data fusion for Smart Healthcare // Journal of Big Data, 2019. – Vol. 6(1). – P.19.

10. Almasri M., Elleithy K. Data fusion in WSNs: architecture, taxonomy, evaluation of techniques, and challenges // Int J Sci Eng Res, 2015. – Vol. 6(4). – P. 1620–36.

11. Klein LA. Sensor and data fusion: a tool for information assessment and decision making // Vol. 138. SPIE press, 2004.

12. Nakamura E.F., Loureiro A.A.F., Frery A.C. Information fusion for wireless sensor networks: methods, models, and classifcations // ACM Comput Surv, 2007. – Vol. 39(3).

13. Neves P.A., Rodrigues J.J., Lin K. Data fusion on wireless sensor and actuator networks powered by the ZenSens system // IET Commun, 2011. – Vol. 5(12). – P. 1661–1668.

14. Zhao C., Wang Y.A. New classifcation method on information fusion of wireless sensor networks // In: IEEE 2008 international conference on embedded software and systems symposia, 2008. – P. 231–36.

15. Durrant-Whyte H.F. Sensor models and multisensor integration // Int J Robot Res, 1988. – Vol. 7(6). – P. 97–113.

16. Biancolillo A., Boque R., Cocchi M., Marini, F. Data fusion strategies in food analysis // In Data Handling in Science and Technology, 2019. – Vol. 31. – P. 271-310.

17. Singh A. et al. Review on multibiometrics: classifications, normalization and fusion levels // 2018 International Conference on Advances in Computing and Communication Engineering (ICACCE). – IEEE, 2018. – C. 159-162.

18. Geethanjali N., Thamaraiselvi K. Feature Level Fusion of Multimodal Biometrics and Two Tier Security // In ATM System, International Journal of Computer Applications, 2013. – Vol. 70. – P. 0975 – 8888.

19. Hoda M. Onsi, Gouda I. Salama, Tarek A. Mahmoud, Ola M. Aly A Multimodal Biometric Recognition system using feature fusion based on PSO // International Journal of Advanced Research in Computer and Communication Engineering, 2013. – Vol. 2(11).

20. Gandhimathi Amirthalingam, Radhamani. G. A Multimodal Approach for Face and Ear Biometric System // IJCSI International Journal of Computer Science Issues, 2013. – Vol. 10(5).

21. Nurul Hayati Che Rani, Khairul Anuar Ishak Dzati Athiar Ramli Performance of Weighted Sum Rule Fusion Scheme in Multi Instance And Multi-Modal biometric System // World Applied Sciences Journal, 2011.

РОЗДІЛ 2 РОЗРОБКА ТЕХНОЛОГІЇ ОБ'ЄДНАННЯ ДАНИХ В СИСТЕМІ ВІДДАЛЕНОГО БІОМЕТРИЧНОГО МОНІТОРИНГУ

Технологія аналізу біометричних даних в системі моніторингу симптомів хвороби Паркінсона включає наступні етапи: отримання та збір даних, попередня обробка, вилучення ознак, моделювання та оцінка якості аналізу біометричних датчиків. Технологія застосовуються для визначення ступеня тяжкості стану пацієнта з хворобою Паркінсона на підставі комплексного аналізу проведених тестів та отриманих результатів.

2.1 Архітектура системи віддаленого моніторингу симптомів хвороби Паркінсона

Концептуальна архітектура системи віддаленого моніторингу симптомів хвороби Паркінсона показана на рис. 2.1. Концептуальна архітектура включає вибір тестів та датчиків смартфону, моніторинг стану, оцінку стану, передачу даних на сервер, перегляд даних моніторингу з використанням додатку Windows.

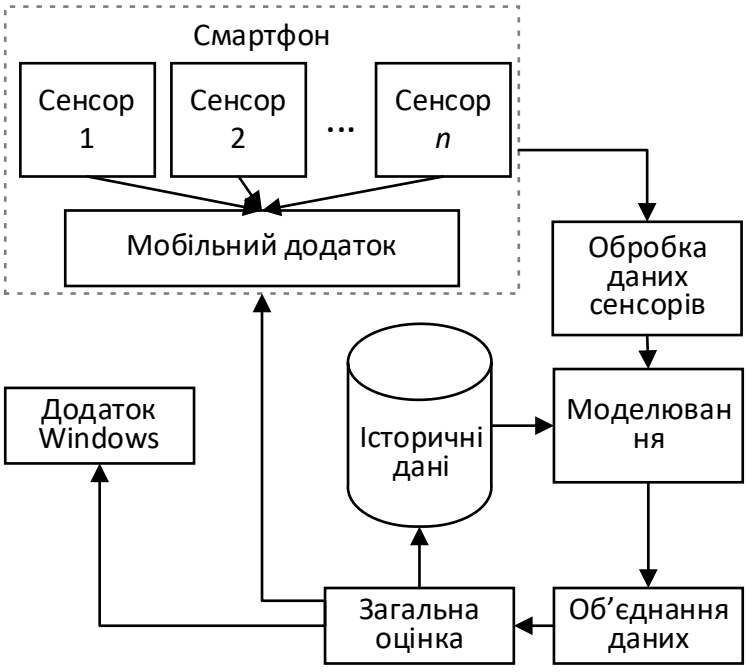


Рисунок 2.1 – Концептуальна архітектура системи віддаленого моніторингу симптомів хвороби Паркінсона

Сигнал, отриманий з сенсорів смартфону при проведенні обраного тесту на наявність симптомів хвороби Паркінсона обробляється з використанням методів статистичного аналізу для створення попереднього первинного висновку про наявність та тяжкість симптомів. Далі дані передаються на сервер для подальшої обробки, моделювання, оцінки ступеня тяжкості симптомів та зберігання. Модуль оцінки симптомів, наприклад, рухового розладу, отримує та об’єднує дані декількох датчиків, тестів та історичних даних бази даних, та може проводити загальну оцінку стану пацієнта.

2.2 Технологія об’єднання даних в системі віддаленого біометричного моніторингу

Запропонована технологія об’єднання даних в системі віддаленого біометричного моніторингу включає застосування локальної моделі класифікації та глобальної моделі об’єднання даних. Глобальна модель об’єднання даних використовує голосування на основі репутації та голосування більшістю.

Складові етапи запропонованої технології описано далі.

2.2.1 Класифікація даних

Для розпізнання й класифікації активності запропоновано використання дерева рішень (Decision Tree).

Дерево рішень - це алгоритм навчання, який використовує деревовидні графи для моделювання та оцінки дискретних функцій [1]. Побудова дерева рішень для цілей класифікації вимагає етапу навчання [2], на якому використовується набір даних і алгоритм навчання для пошуку дерева рішень з мінімальною глибиною. Дерево має містити мінімально необхідні вузли (або мінімальну глибину) для зменшення часу аналізу даних та обчислювального навантаження. Отже, алгоритм навчання зазвичай є жадібним алгоритмом локального пошуку для пошуку оптимального дерева рішень. Як тільки дерево рішень створено, дерево можна використовувати, оцінюючи вузли від кореня до листя. Останній лист містить значення, яке показує результат класифікації. Після того, як дерево рішень побудовано на етапі навчання, його можна обрізати для зменшення обчислювального навантаження. Обрізка необхідна тільки для великих дерев, щоб полегшити обчислювальну складність.

2.2.2 Голосування на основі репутації

Голосування на основі репутації - це метод об’єднання даних, що застосовується після класифікації. Коли кожна спостереження, що класифікується приймає індивідуальне рішення про приналежність екземпляра до класу, необхідно досягти консенсусу серед спостережень, що класифікують. Підходи до голосування на основі репутації засновані на з'ясуванні репутації окремого спостереження, що класифікується, та виборі рішення, що приймається на підставі класифікованих спостережень, що мають найвищу репутацію. Припускаючи, що класифіковані спостереження класифіковані правильно, необхідно оцінити, наскільки точно класифіковані інші спостереження. Для винесення судження кожне спостереження спочатку відправляє свій результат класифікації, званий значенням ваги для всіх інших спостережень в своєму оточенні.

Значення ваги визначаються для всіх сусідніх спостережень і зберігаються в таблиці значення ваг сусідніх спостережень. На наступному етапі кожне спостережень повинне судити про своє сусіднє спостережень, розглядаючи себе як еталон. Судження досягається шляхом порівняння різниці між значенням ваги самого спостереження і значеннями ваг інших сусідніх спостережень що класифікують. Якщо різниця менше порогового значення θ (яке вибирається на основі контексту), спостережень що оцінює дає позитивний голос ($V_{new} = V_{old} + 1$) іншим спостереженням.

В іншому випадку спостереження, що оцінюється отримує негативний голос ($V_{new} = V_{old} -1$). Отримані дані зберігаються в таблиці для досягнення консенсусу між різними думками. Складна частина голосування на основі репутації полягає в тому, як привласнити значення глобальної репутації кожного спостереження, щоб вибрати сутність з високою репутацією і результат її класифікації. Це можна зробити різними способами, два з яких описані далі.

2.2.3 Методи на основі репутації

Запропоновано два методи об’єднання на основі репутації для глобальної моделі об’єднання даних на основі репутації.

Перший метод на основі репутації перевіряє локальну репутацію кожного окремого спостереження з точки зору інших сутностей. Значення локальної репутації отримується на основі середнього значення V_i (позитивних або негативних голосів, відданих іншими спостереженнями) для кожного класифікованого спостереження.

Потім середня локальна репутація множиться на вагу вузлів, розраховану з використанням рівняння 2.1, для присвоєння значень глобальної репутації. Клас з найвищою вагою репутації (*W*) є результатом процедури голосування. Рівняння 2.1 показує, як розраховуються ваги.

$$W_i = R_i \text{ x } Acc_i \tag{2.1}$$

де W_i - значення репутації, відповідно класифікованого спостереження *i*, R_i - значення місцевої репутації класифікованого спостереження *i* з точки зору інших спостереження, а Acc_i - вага класифікованого спостереження *i* [3].

В другому методі на основі репутації використовуються два порогових значення θ_1, θ_2. Порівняння значення місцевої репутації (R_i) с θ_1 и θ_2 дає уявлення про те, наскільки добре виконується класифікація. Якщо ($R_i \geq \theta_1$), то класифіковані спостереження прийняли ідеальні рішення, якщо ($\theta_1 > R_i \geq \theta_2$), то класифіковані спостереження прийняли правильні рішення, і якщо ($\theta_2 \geq R_i$), тоді класифіковані спостереження прийняли невірні рішення. Визначено порогові значення наступним чином: 0,5 балів за низьку ефективність класифікації, 1 за нормальну ефективність класифікації і 2 за ідеальну ефективність класифікації. Ґрунтуючись на цих значеннях, цей метод репутації використовує рівняння 2.2 для присвоєння репутації кожному класифіковані спостереженню.

$$W_i = S_i \text{ x } Acc_i \tag{2.2}$$

де W_i - значення репутації, відповідне класифікованому спостереженню *i*, S_i виходить з рівняння 2.3, а Acc_i - вага класифікованого спостереження.

$$S_i = \begin{cases} 2 & \text{якщо } (R_i \geq \theta_1) \\ 1 & \text{якщо } (\theta_1 > R_i \geq \theta_2) \\ 0.5 & \text{якщо } (\theta_2 \geq R_i) \end{cases} \tag{2.3}$$

де θ_1 и θ_2 залежать від додатку.

2.2.4 Голосування більшістю

Голосування більшістю - це простий метод голосування, при якому вибирається результат класифікації, отриманий від спостереження, що має найбільшу репутацію.

2.3 Моделі технології об’єднання даних

2.3.1 Локальна модель класифікації

Локальна модель передбачає, що кожні дані класифікуються індивідуально, не взаємодіючі з іншими. На рис. 2.2 показана локальна модель класифікації, яка складається з (1) датчику смартфону, який надає вхідні дані для класифікатора, (2) класифікатора, який відповідає за розпізнавання ступеня тяжкості симптому і визначення приналежності кожного спостереження до класу ступеня тяжкості, і (3) заключний висновок класифікації - оцінка тяжкості симптому. Слід зазначити, що не всі сенсорні вузли повинні мати однакові класифікатори.

Рисунок 2.2 - Локальна модель класифікації

2.3.2 Глобальна модель об’єднання даних

Глобальна модель об'єднання допускає не використання даних окремих датчиків і вузлів датчиків і включає в себе більше одного вузла датчиків в процесі класифікації. Тим самим вона гарантує, що завжди є кілька вузлів датчиків, що беруть участь в процесі класифікації, та компенсує помилки. Глобальна модель об'єднання використовує основні результати локального підходу і дозволяє окремим вузлам датчиків спочатку класифікувати і виявляти дії самостійно.

Потім всі результати класифікації відправляються на вузол об'єднання для досягнення консенсусу. На рис. 2.3 показана запропонована технологія об’єднання даних з використанням локальних моделей класифікації та глобальної моделі об’єднання даних.

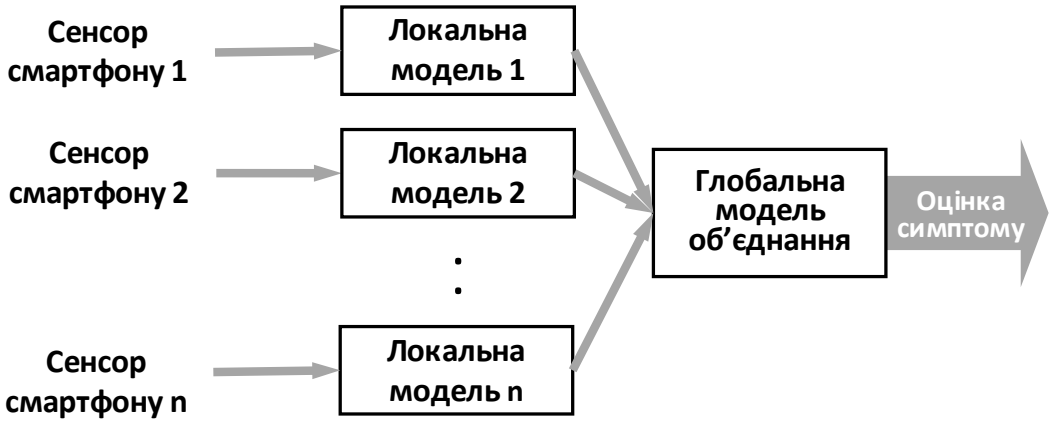


Рисунок 2.3 – Технологія об’єднання даних з використанням локальних моделей класифікації та глобальної моделі об’єднання даних

Вузли датчиків (включаючи вузол об'єднання) можуть мати різні класифікатори.

Запропонована технологія об’єднання даних, що включає локальну модель класифікації та глобальну модель об1єднання, в системі віддаленого біометричного моніторингу використана для оцінки ступеня тяжкості тремору.

2.4 Розрахунок обчислювальної складності технології на основі об’єднання і голосування більшістю

Складність процесу технології на основі злиття з використанням дерев рішень становить $O(m_1 + m_2 + \cdots + m_n) = O(m)$; де n - кількість вузлів, що беруть участь в класифікації, а m - глибина дерева рішень. Оскільки голосування не залежить від класифікації, його тимчасова складність додається до часу класифікації, як показано в рівняннях 2.4, 2.5, 2.6. Рівняння 2.7 представляє остаточну складність технології об'єднання з використанням дерев рішень і голосування більшістю.

$$OO\ (\textit{Розподілений підхід з використанням голосування більшістю}) = O([m]) + O([\textit{голосування більшістю 1}]) = \textit{Максимум}\ [(m), \textit{голосування більшістю}] \quad (2.4)$$

$$O(\textit{Розподілений підхід з використанням голосування більшістю}) = O(\textit{Максимальне отримане}) + O(\mathrm{m}) \quad (2.5)$$

$$OO\ (\textit{Розподілений підхід з використанням голосування більшістю}) = \quad (2.6)$$

$$O(\max \textit{отримане}) = O(\mathrm{c})$$

$$O\ (O(\textit{Розподілений підхід з використанням голосування більшістю}) = O(\mathrm{c}) + O(\mathrm{m})) = O(m) \quad (2.7)$$

де m - глубина дерева рішень, а n - кількість вузлів.

2.5 Висновки до другого розділу

У розділі розглянуто розробку та проектування архітектури системи віддаленого біометричного моніторингу симптомів хвороби Паркінсона.

Представлено запропоновану технологію об’єднання даних датчиків смартфону з використанням локальної моделі класифікації та глобальної моделі об’єднання даних на основі репутації та голосування більшістю. Локальна модель класифікації розробляється з використанням дерева рішень. Глобальну модель об'єднання на основі репутації включає методи визначення репутації кожного спостереження.

2.6 Перелік посилань до другого розділу

1. Russell, S. J. and P. Norvig (2003). Artificial Intelligence: A Modern Approach, 2 edition, Pearson Education, Alpaydin, E. (2004). Introduction to Machine Learning, The MIT Press.

2. Alpaydin, E. (2004). Introduction to Machine Learning, The MIT Press.

3. Bahrepour, M., N. Meratnia, et al. (2010). Fast and Accurate Residential Fire Detection Using Wireless Sensor Networks. Environmental Engineering and Management Journal9(2): 215-221.

4. Bahrepour, M., N. Meratnia, et al. (2009). Sensor Fusion-based Event Detection in Wireless Sensor Networks. SensorFusion’09. IEEE. Toronto, Canada

РОЗДІЛ 3. ПРОГРАМНА РЕАЛІЗЦІЯ СТРУКТУРНИХ ЕЛЕМЕНТІВ СИСТЕМИ ВІДДАЛЕНОГО БІОМЕТРИЧНОГО МОНІТОРИНГУ ТА ТЕХНОЛОГІЇ ОБ'ЄДНАННЯ ДАНИХ

Програмна реалізація системи віддаленого біометричного моніторингу та технології об'єднання даних включає розробку наступних структурних компонентів системи:

– мобільний додаток для операційної системи Android, який включає розробку функціональних тестів на перевірку розладів рухової активності при хворобі Паркінсона та анкету для опитування користувача;

– додаток Windows;

– серверна частина;

– практичну реалізацію об'єднання даних датчиків смартфону для отримання оцінки про ступінь тремору.

Практичну реалізацію цих елементів розглянуто далі.

3.1 Розробка мобільного додатку

Мобільний додаток розроблено для операційної системи Android з використанням середи розробки Android Studio на мові програмування Java.

Розробка мобільного додатку включає наступні дії.

Запускаємо Android Studio, й створюємо новий проект (Basic Activity) та створюємо у папці app->res->layout файли.

```
activity_accelerometer
activity_login
content_accelerometer
content_login
```

Натискаємо правою кнопкою миші на файл, «layout»->New->Layout resource file (див. рис. 3.1).

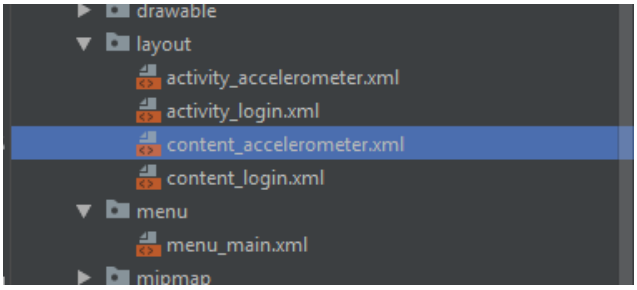


Рисунок 3.1 – layout файли.

У файлах activity_accelerometer та activity_login створено toolbar (меню palette->container->AppBarLayout), як це представлено на рис. 3.2.

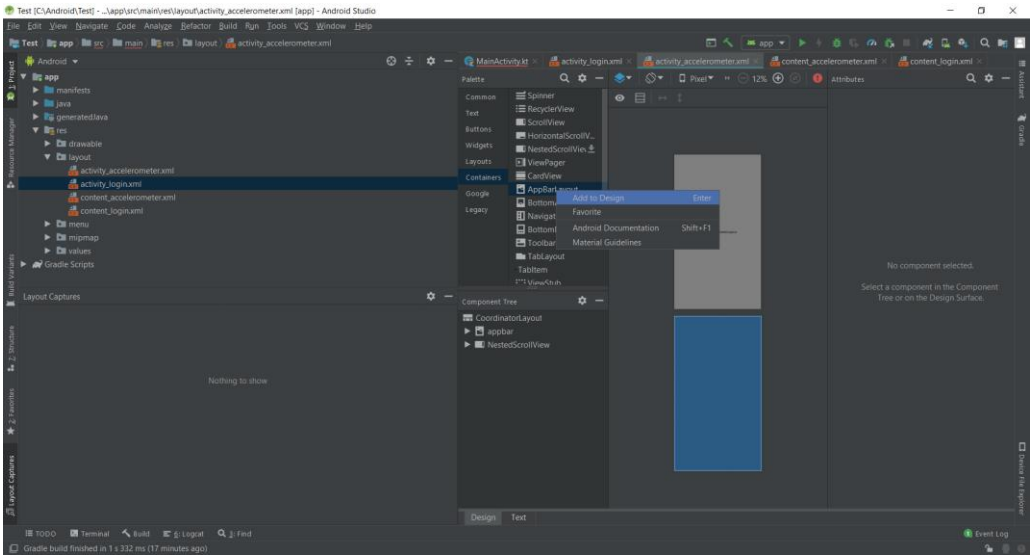


Рисунок 3.2 – Додали AppBarLayout

Розміщено на екрані елементи з palette а в поле атрибутів ID задаємо те, що в дужках.

```
TableLayout
TableRow
textView  (txtXLabel)
TableRow
Button (read_btn)
TableRow
LinearLayout (Layout_Graph_Container)
```

В результаті отримано наступне(рис. 3.3., 3.4).

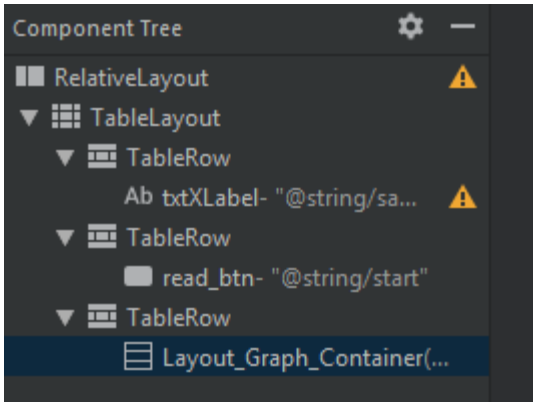


Рисунок 3.3 – додали розмітку екрану.

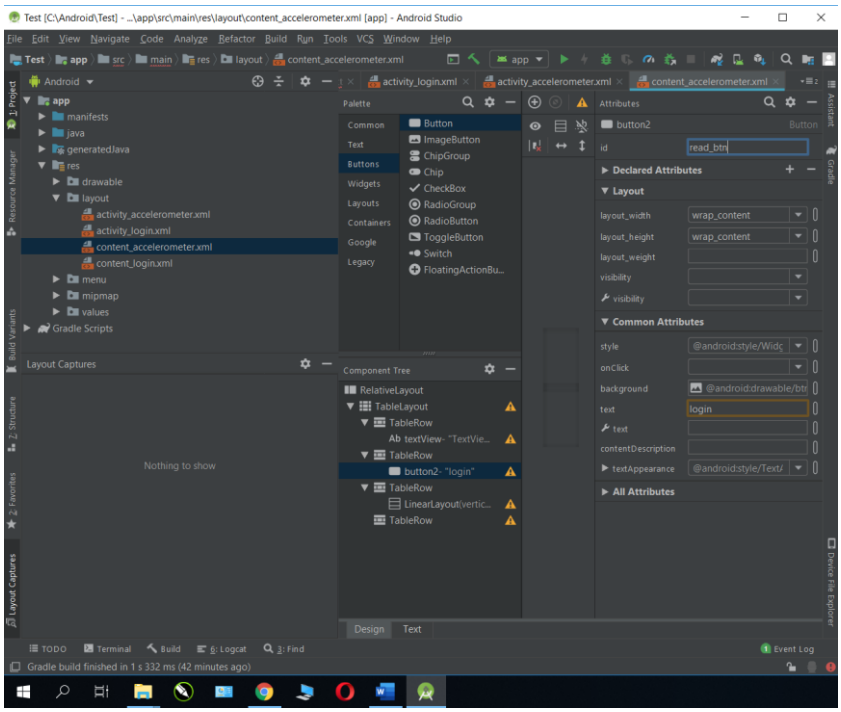


Рисунок 3.4 – підписали кнопки

У content_login розміщено одну єдину кнопку (елемент button), а в поле id записано btn_login. Далі у menu файлі menu_menu створюємо елемент item з id action_logout.

До файлу res->values->strings.xml копіюємо код з однойменного txt файлу, як це представлено на рис. 3.5.

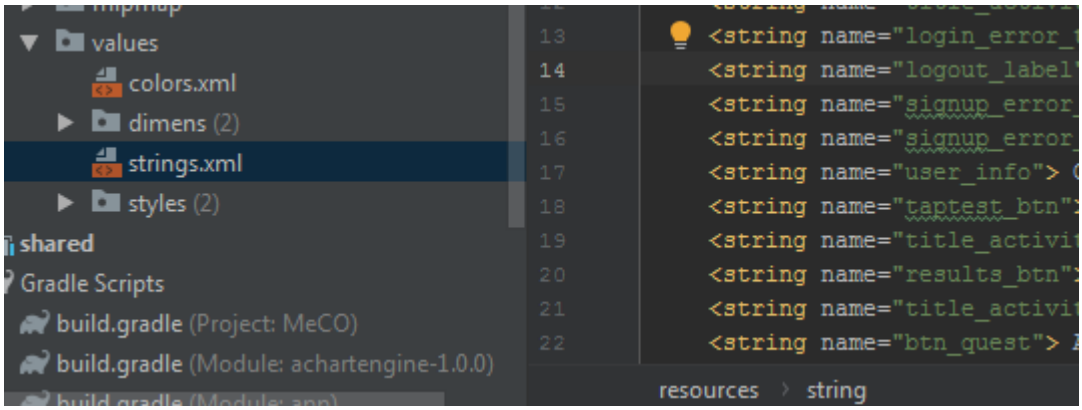


Рисунок 3.5 – файл «string.xml». у якому зберігаються строкові ресурси

Приклади інтерфейсу розробленого мобільного додатку представлено на рис. 3.6.

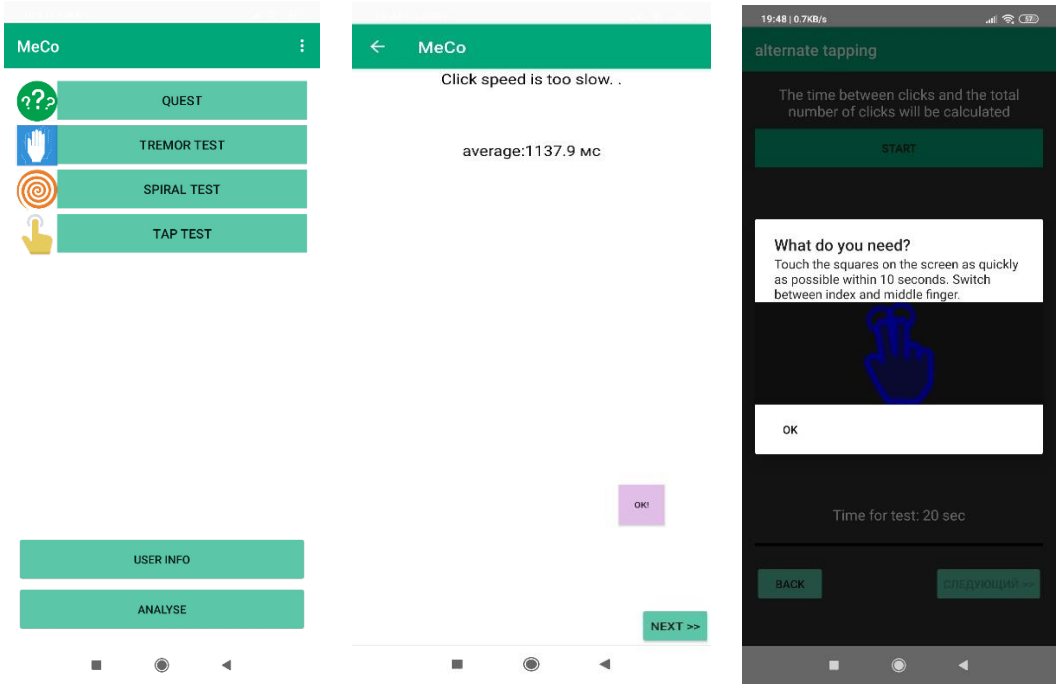


Рисунок 3.6 – Інтерфейс розробленого мобільного додатку

3.2 Розробка додатку Windows

Додаток Windows розроблено з використанням середи розробки Visual Studio 2019 на мові програмування C#. Код програмної реалізації додатку Windows наведено в Додатку Б.

При запуску додатку Windows, з'являється вікно, в яке необхідно ввести дані вже зареєстрованого користувача (рис. 3.7) і натиснути кнопку «Log in», або створити нового користувача, якщо необхідно, натиснувши кнопку «Create new account».

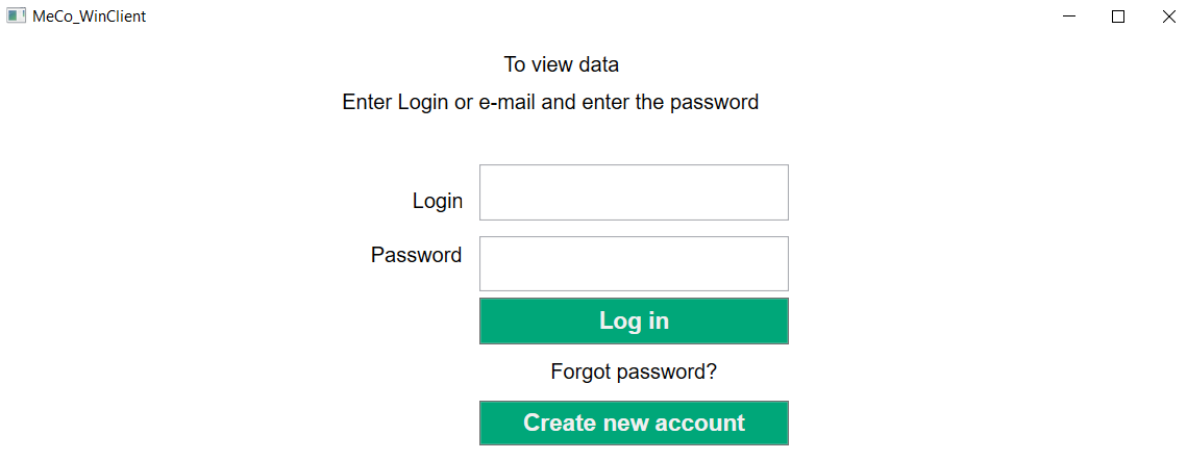


Рисунок 3.7 - Окно входа розробленого додатку Windows

При створенні нового користувача з'явиться вікно введення даних нового користувача (рис. 3.8), де потрібно вказати ім'я, пароль, email, вік, стать, та обрати лівша чи правша.

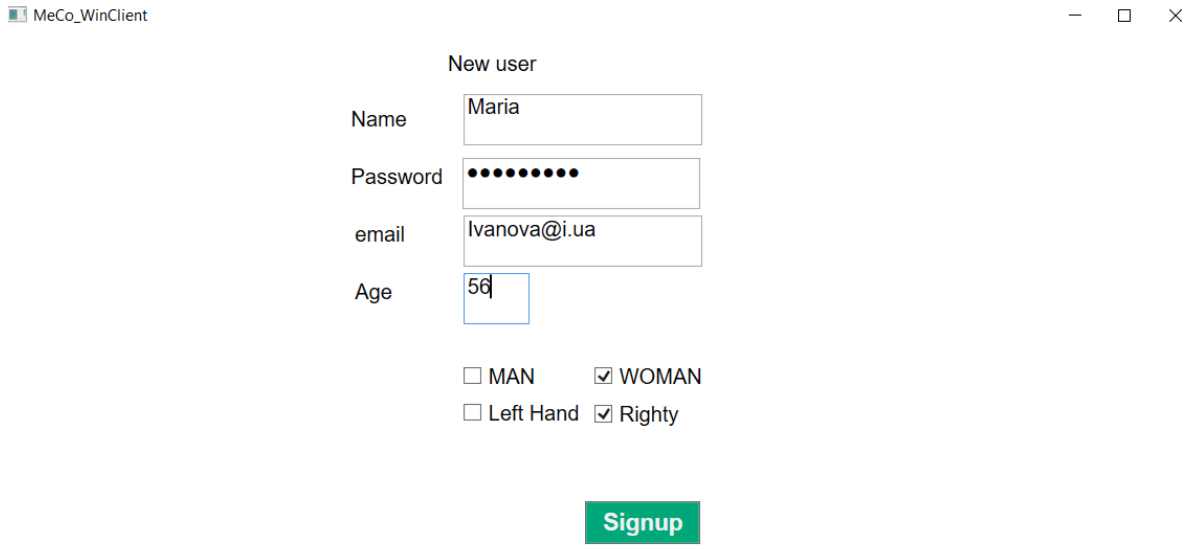


Рисунок 3.8 - Вікно реєстрації нового користувача.

Після успішного входу або реєстрації нового користувача, з'являється головне вікно програми, що представлено на рис. 3.9, де можна переглянути або змінити наступні дані, виконати наступні дії, помічені на рис. 3.9 цифрами та стрілками:

1. Ім'я користувача.
2. Його ID в системі.
3. Вік, стать, email.
4. Дані останнього тесту.
5. Редагувати інформацію про користувача.
6. Запит на оновлення даних про проходження тестів.
7. Запит на формування замітки.
8. Запит на отримання останньої замітки.

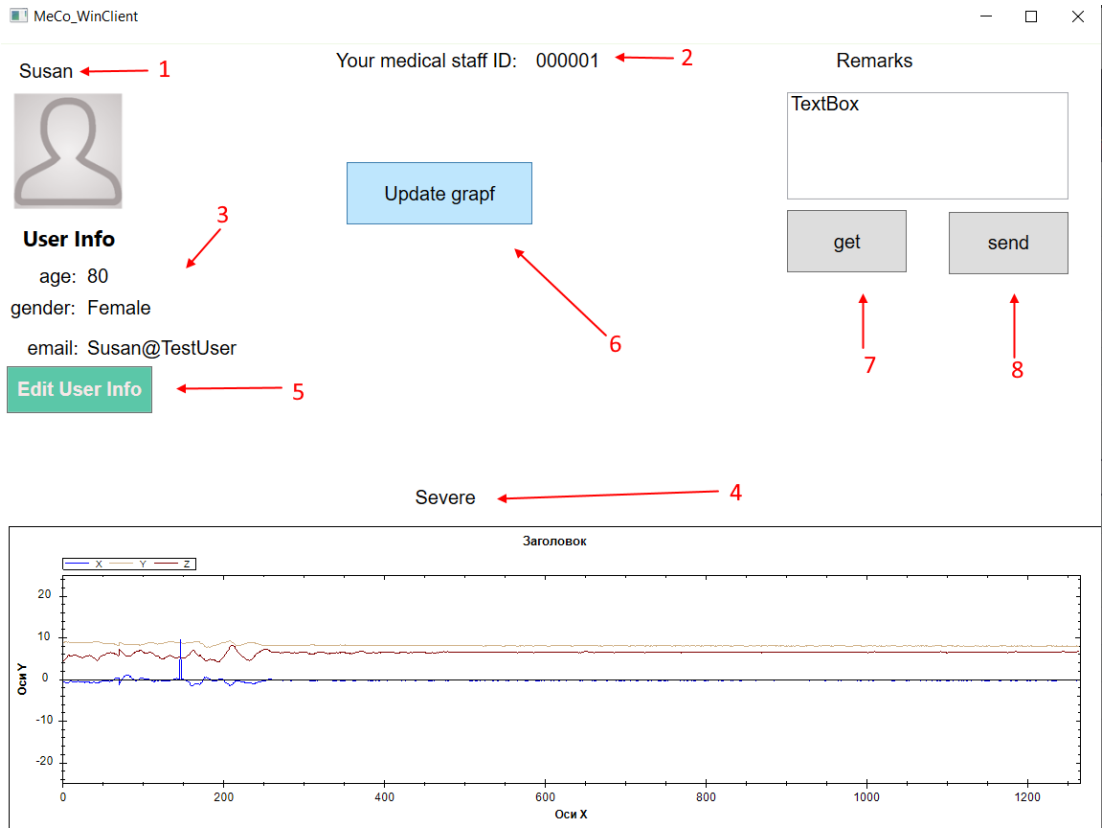


Рисунок 3.9 - Головне вікно програми

3.3 Реалізація серверної частини

Розроблено два варіанта віддаленого сховища. Перший варіант включає використання хмарного ресурсу Back4App [1].

3.3.1 Хмарне сховищ

Для доступу до смартфону, інтернету, і можливості синхронізації з хмарним сховищем необхідно задати дозволи у файлі app->manifest->AndroidManifest.xml Для цього додамо наступний рядок в даний файл

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.myapplication">
......
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission android:name="android.permission.GET_ACCOUNTS" />
    <uses-permission android:name="android.permission.READ_PROFILE" />
    <uses-permission android:name="android.permission.READ_CONTACTS" />
    <uses-permission android:name="android.permission.VIBRATE" />
      ........
```

Для доступу до хмарного сховища створюємо файли наступні файли:

```
AccelAnalysis
AccelData
Login
MainActivity
ParseApp
ParseFunctions
Statistics
```

Копіюємо в них код з відповідних .txt файлів, за винятком першого рядка.

Для підключення додатку до сервера в файл ParseApp додаємо інформацію для підключення.

Для з'єднання з сервером, і правильного відображення графіків бібліотеки, що є в архіві libs, розпаковано в папку app-> libs.(рис. 3.10)

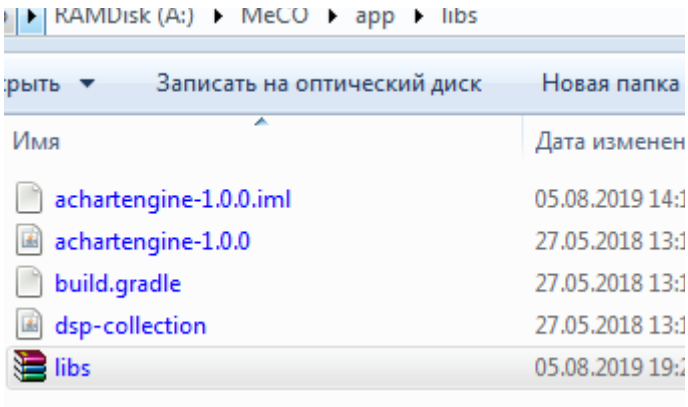


Рисунок 3.10 розпаковані бібліотеки.

Вводимо в файл build.gradle(Project: appname)

```
allprojects {
repositories {
...
maven { url "https://jitpack.io" }
}
}
```

Після цього вводимо в кінець файлу build.gradle (Module: app) наступний код

```
dependencies {
    implementation 'com.github.parse-community.Parse-SDK-
Android:parse:1.18.5'
}
```

3.3.2 Сервер

Для ефективного використання серверу, розгорнуто декілька віртуальних машин. Консоль управління віртуальними машинами, на якому розміщено віртуальні машини з серверним програмним забезпеченням та роутерами, корисне навантаження – для кращої масштабованості та більш ефективного використання ресурсів серверу, як це представлено на рис. 3.11.

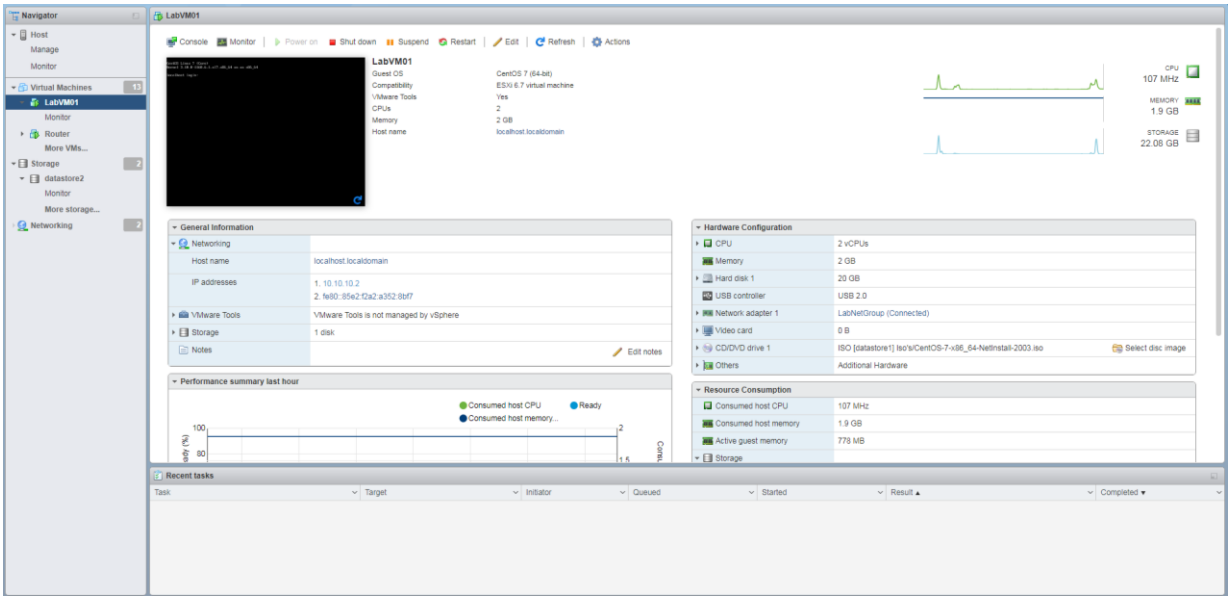


Рисунок 3.11 – Ресурси серверу

Веб-інструмент Webmin [2] використано для налаштування та управління Unix системи серверу. Webmin використовується для керування та моніторингу конкретною одиницею серверу. За допомогою Webmin налаштовані внутрішні компоненти операційної системи, визначені користувачі. На рис. 3.12 представлено перегляд інформації про статус системи.

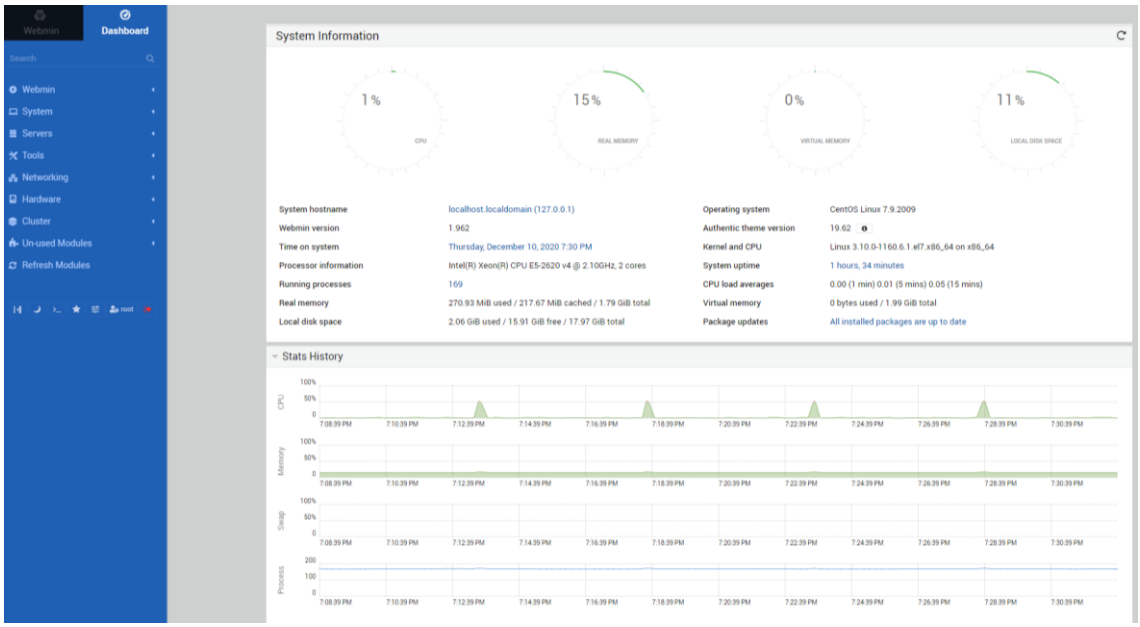


Рисунок 3.12 – Перегляд інформації про статус системи

Налаштування маршрутизаторів проведено з використанням програмного забезпечення WinBox [3]. Додаток WinBox, по суті, замінює собою стандартну консоль управління маршрутизатором, та використовується для налаштування підключення, віддаленого доступу, маршрутизації між серверами та мережевим обладнанням, а також, для захисту мережі та серверу від несанкціонованого доступу.

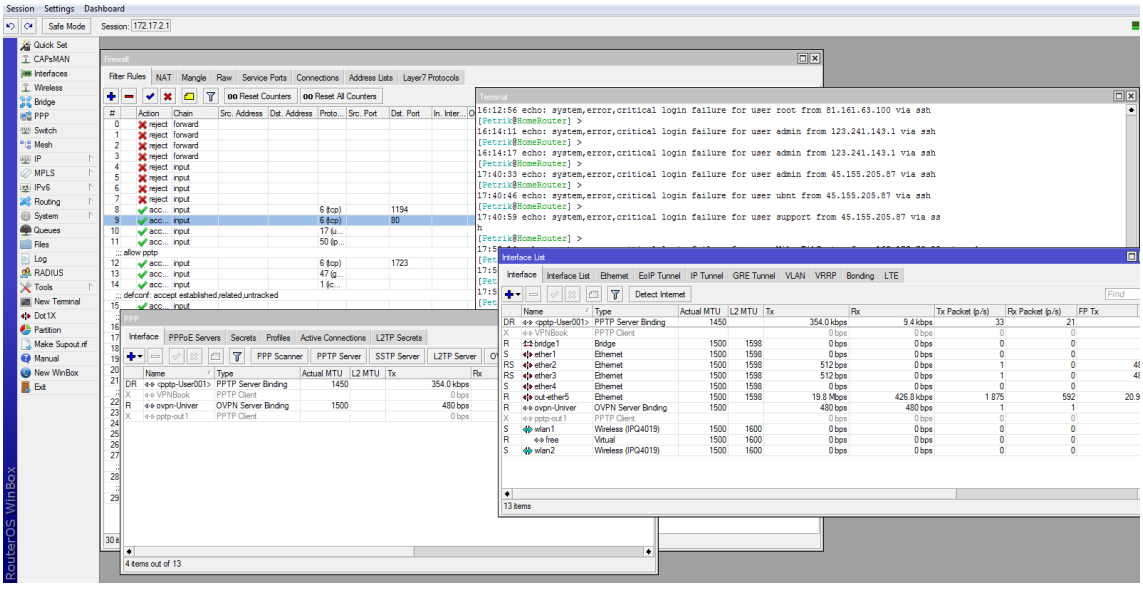


Рисунок 3.13 - Налаштування маршрутизаторів

Для створення та адміністрування бази даних користувачів розробленого мобільного додатку моніторингу симптомів хвороби Паркінсона підключено Laravel до серверу.

Laravel [4] - безкоштовний веб-фреймворк з відкритим кодом, призначений для розробки з використанням архітектурної моделі MVC (англ. Model View Controller - модель-уявлення-контролер). На рис. 3.14 представлено інтерфейс Laravel.

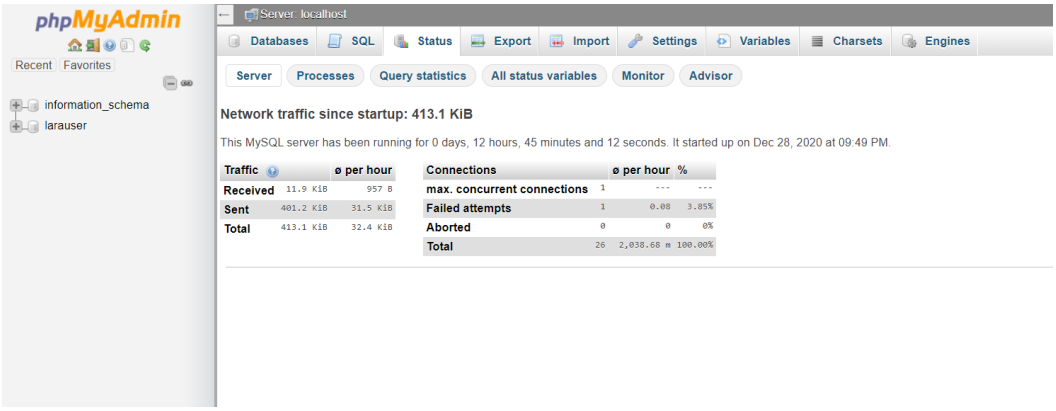


Рисунок 3.14 - Інтерфейс Laravel

На рис. 3.15 представлено один з модулів веб-фреймворку, що забезпечує роботу посилань.

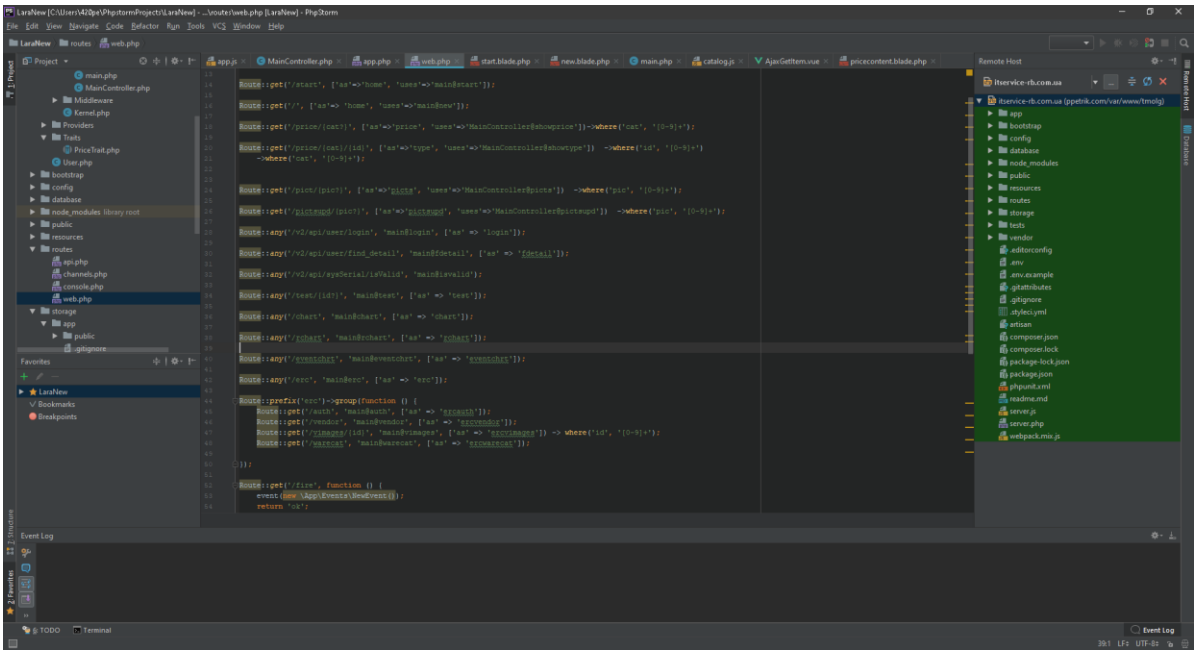


Рисунок 3.15 - Веб-фреймворк

Для доступу до бази даних з використанням Laravel використовується виділене доменне ім’я vodarb.pp.ua.

3.4 Реалізація технології злиття даних

3.4.1 Опис даних

Практична реалізація технології злиття даних проведена з використанням даних датчиків смартфону, отриманих з використанням розробленого мобільного додатку для смартфону [5], тесту на визначення ступеня тяжкості тремору, проведеного з використанням вбудованого акселерометра смартфона. Дані отримані відповідно до шкали MDS-UPDRS [6] для оцінки постурального тремору. Ця шкала має наступні ступені тяжкості тремору і відповідну їм амплітуду:

Tr0: відсутність тремору.

Tr1: дуже легкий: тремор присутній, але його амплітуда менше 1 см.

Tr2: легкий: тремор присутній, його амплітуда як мінімум 1 см, але менш 3 см.

Tr3: помірний: тремор присутній, його амплітуда як мінімум 3 см, але менш 10 см.

Tr4: важкий: тремор з амплітудою 10 см і більше.

Набір даних складається з даних 12250 часових кроків з 10 тестів із використанням акселерометра. Дані оброблено так, як це описано вище. Оброблений набір даних для навчання локальної моделі складається зі 100 сегментів вилучених ознак. Мітки класів розподілені рівномірно: по 20 випадків кожного класу.

Отримані дані сегментовані в непересічні односекундні сегменти. Вилучення ознак проведено з використанням наступних критеріїв: середнє, стандартне відхилення, енергія послідовності, кореляція Пірсона. Вилучені ознаки составили новий набір даних, який містить наступні змінні: F1 meanX, F2 meanY, F3 meanZ, F4 stdX, F5 stdY, F6 stdZ, F7 SI X, F8 SI Y, F9 SI Z, F10 PC XY, F11 PC YZ, F12 PC XZ.

3.4.2 Навчання локальної моделі класифікації

Отримані дані вилучених з даних тестів на тремор ознак класифіковані з використанням 10-кратної перехресної валідації. Класифікація здійснюється з використанням алгоритму дерева рішень (Decision Tree). Повний код програмної реалізації моделі представлено в Додатку Б.

В результаті навчання моделі локальної класифікації на підставі вилучених ознак отримана наступна вирішальна функція та коефіцієнти.

```
F4 stdX <= 1.290633
|   F2 meanY <= 3.716607: Tr0 (20.0)
|   F2 meanY > 3.716607: Tr3 (8.0)
F4 stdX > 1.290633
|   F6 stdZ <= 1.838338
|   |   F5 stdY <= 1.891277
|   |   |   F2 meanY <= 2.438984: Tr4 (4.0/1.0)
|   |   |   F2 meanY > 2.438984: Tr1 (20.0)
|   |   F5 stdY > 1.891277: Tr2 (19.0)
|   F6 stdZ > 1.838338
|   |   F4 stdX <= 7.64353
|   |   |   F5 stdY <= 1.891277: Tr4 (3.0)
|   |   |   F5 stdY > 1.891277: Tr3 (14.0/2.0)
|   |   F4 stdX > 7.64353: Tr4 (12.0)
```

Нові спостереження класифікуються з використанням цієї функції та коефіцієнтів. Навчена локальна модель дерева рішень використовує цю функцію та коефіцієнти для розрахунку критеріїв точності моделі як відповідність значень спостережень функції та спостереженням.

Отримана локальна модель класифікації з використанням алгоритму дерева рішень має глибину дерева рішень, що дорівнює 15, та має 8 вузлів. Отримане дерево рішень з коефіцієнтами для кожної гілки та змінними в якості вузлів представлено на рис. 3.16.

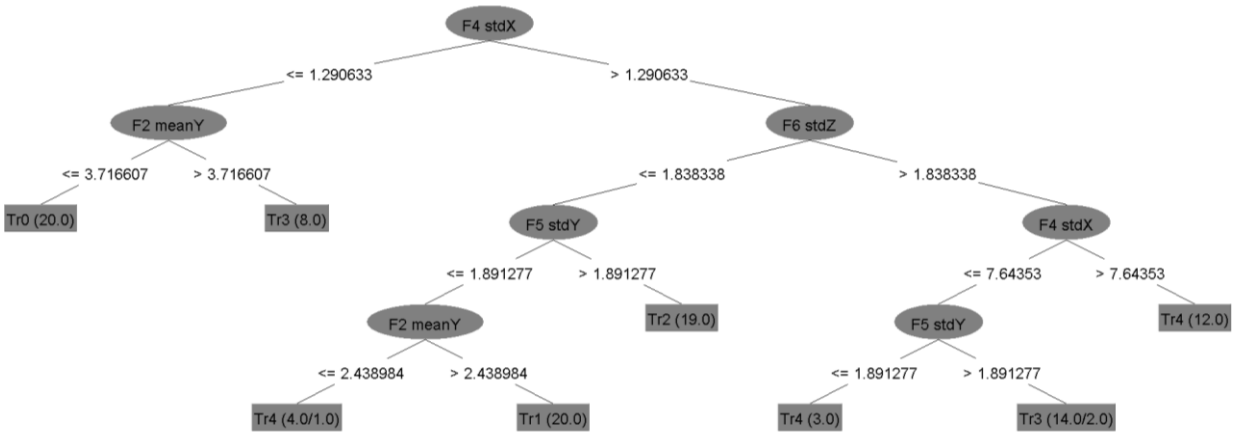


Рисунок 3.16 – Отримане дерево рішень

Для визначення якості локальної моделі класифікації використано наступні критерії: середня абсолютна помилка, середньоквадратична помилка, що становлять 0.0635 та 0.2332 відповідно. Отримано наступні результати хибнопозитивних (FP), хибнонегативних (FN), істинопозитивних (TP) та істинонегативних (TN) результатів класифікації, що представлені в матриці невідповідності наступним чином.

```
  a  b  c  d  e   <-- classified as
 18  1  0  0  1 |  a = Tr0
  0 19  0  1  0 |  b = Tr1
  0  0 19  0  1 |  c = Tr2
  1  0  0 14  5 |  d = Tr3
  1  0  0  4 15 |  e = Tr4
```

На підставі матриці невідповідності розраховано наступні критеріїв визначення якості отриманої локальної моделі класифікації: відношення TP результатів, відношення FP результатів, що використані для розрахунку площі під ROC кривою (ROC Area). Також розраховано наступні критерії: діагностична чутливість (*Recall*), діагностична специфічність (*Precision*), міра F1 (*F-score*), коефіцієнт кореляції Метьюза (*MCC*), що розраховуються з використанням хибнопозитивних (FP), хибнонегативних (FN), істинопозитивних (TP) та істинонегативних (TN) результатів, як це представлено у (3.1) – (3.4).

$$Recall = \frac{\mathrm{TP}}{\mathrm{TP+FN}} \tag{3.1}$$

$$Precision = \frac{\mathrm{TP}}{\mathrm{TP+FP}} \tag{3.2}$$

$$F-score = \frac{2 * Precision * Recall}{(Precision + Recall)} \tag{3.3}$$

$$MCC = \frac{\mathrm{TP} * \mathrm{TN} - \mathrm{FP} * \mathrm{FN}}{\sqrt{(\mathrm{TP+FP})(\mathrm{TP+FN})(\mathrm{TN+FP})(\mathrm{TN+FN})}} \tag{3.4}$$

Отримані результати критеріїв для кожного класу та узагальнена оцінка представлені наступним чином.

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	Class
	0,900	0,025	0,900	0,900	0,900	0,875	0,937	Tr0
	0,950	0,013	0,950	0,950	0,950	0,938	0,967	Tr1
	0,950	0,000	1,000	0,950	0,974	0,969	0,973	Tr2
	0,700	0,063	0,737	0,700	0,718	0,650	0,843	Tr3
	0,750	0,088	0,682	0,750	0,714	0,640	0,856	Tr4
Середня оцінка	0,850	0,038	0,854	0,850	0,851	0,814	0,790	

Навчена локальна модель використовується для моделювання процесу класифікації і тестування кожного датчика окремо або окремих вузлів датчиків.

3.4.3 Глобальна модель об’єднання даних на основі репутації

На етапі навчання спочатку навчаються локальні моделі класифікації, що працюють для кожного датчика окремо або для окремих вузлів датчиків. Потім, згідно з навченими локальними моделями класифікації на окремих датчиках або окремих вузлів датчиків, навчається глобальна модель об’єднання даних. На етапі тестування екземпляр даних передається всім окремим датчикам або окремим вузлам датчиків, а потім їх ваги, отримані в результаті застосування локальної моделі класифікації, об'єднуються з використанням методу голосування. Фазу навчання проведено з використанням 83% даних, а тестування - з використанням 17% даних. Результат класифікації тестового набору даних не є достатньо точним, що поясняється малим об’ємом даних тестів на тремор, що використовуються для навчання локальної моделі і наразі є завданням майбутніх досліджень. В даному дослідженні дані локальної моделі використано для демонстрації технології злиття даних тестів на тремор правої та лівої рук для винесення остаточної загальної оцінки про ступінь тремору користувача.

Продемонстровано стадію злиття даних на прикладі виявлення ступеня тремтіння правої та лівої рук для отримання загальної оцінки ступеня тремору. Дані правої руки були отримані з амплітудою тремору від 1 до 3 см, що відповідає класу Tr2 = легкий. Дані лівої руки були отримані з амплітудою тремтіння від 3 до 10 см, що відповідає класу Tr3 = помірний. Окремі дані правої та лівої рук отримані, оброблені та класифіковані з використанням локальної моделі, як описано вище. Результати, отримані для даних тесту, класу та ваги класу для правої та лівої рук, представлені в табл. 3.1.

Таблиця 3.1 – Результат моделювання локальної моделі

Тест правої руки			Тест лівої руки		
ID спостереження	Вага Acc_i	Прогнозований клас	ID спостереження	Вага Acc_i	Прогнозований клас
1	1	Tr2	1	1	Tr3
2	0.75	Tr2	2	0.857	Tr1
3	0.75	Tr2	3	1	Tr3
4	0.75	Tr2	4	1	Tr3
5	0.75	Tr2	5	1	Tr3
6	0.75	Tr2	6	1	Tr3
7	0.75	Tr2	7	1	Tr3
8	0.75	Tr2	8	1	Tr3
9	0.75	Tr2	9	0.857	Tr1
10	0.75	Tr2	10	0.857	Tr1

Голосування на підставі репутації проводиться наступним чином. Кожне спостереження, починаючи з першого, судить про своє сусіднє спостережень, розглядаючи себе як еталон. Судження досягається шляхом порівняння різниці між значенням ваги самого спостереження і значеннями ваг інших сусідніх класифікованих спостережень. Порогового значення обране на основі контексту θ = 0.2. Якщо різниця між значеннями ваг спостережень менше 0.2, спостереження, що оцінюється дає позитивний голос $V_{new} = V_{old} + 1$ іншим спостереженням. В іншому випадку спостереження, що оцінюється, отримує негативний голос $V_{new} = V_{old} - 1$. Сума негативних та позитивних голосів V_{new} спостереження складає його репутацію R_i. Таким чином, отримаємо наступні дані, що використовуються далі для досягнення консенсусу, представлені в табл. 3.2.

Таблиця 3.2 – Дані отримані в результаті класифікації та голосування

Тест правої руки				Тест лівої руки			
ID	Вага Acc_i	R_i	Прогнозований клас	ID	Вага Acc_i	R_i	Прогнозований клас
1	1	8	Tr2	1	1	11	Tr3
2	0.75	7.75	Tr2	2	0.857	10.857	Tr1
3	0.75	7.75	Tr2	3	1	11	Tr3
4	0.75	7.75	Tr2	4	1	11	Tr3
5	0.75	7.75	Tr2	5	1	11	Tr3
6	0.75	7.75	Tr2	6	1	11	Tr3
7	0.75	7.75	Tr2	7	1	11	Tr3
8	0.75	7.75	Tr2	8	1	11	Tr3
9	0.75	7.75	Tr2	9	0.857	10.857	Tr1
10	0.75	7.75	Tr2	10	0.857	10.857	Tr1

За першим запропонованим методом об’єднанням на основі репутації вихідна вага класифікованого класу тесту визначається на підставі даних табл. 3.2. Таким чином, визначено клас з найвищою репутацією *W*. Для 10 спостережень тесту на тремор правої руки він розраховується наступним чином за формулою (2.1).

$W_{r1} = 8 \times 1 = 8$

$W_{r2} = 7.75 \times 0.75 = 5.8125$

$W_{r3} = 7.75 \times 0.75 = 5.8125$

$W_{r4} = 7.75 \times 0.75 = 5.8125$

$W_{r5} = 7.75 \times 0.75 = 5.8125$

$W_{r6} = 7.75 \times 0.75 = 5.8125$

$W_{r7} = 7.75 \times 0.75 = 5.8125$

W_{r8} = 7.75 x 0.75 = 5.8125

W_{r9} = 7.75 x 0.75 = 5.8125

W_{r10} = 7.75 x 0.75 = 5.8125

Для 10 спостережень тесту на тремор лівої руки він розраховується наступним чином.

W_{l1} = 11 x 1 = 11

W_{l2} = 10.857 x 0.857= 9.3044

W_{l3} = 11 x 1 = 11

W_{l4} = 11 x 1 = 11

W_{l5} = 11 x 1 = 11

W_{l6} = 11 x 1 = 11

W_{l7} = 11 x 1 = 11

W_{l8} = 11 x 1 = 11

W_{l9} = 10.857 x 0.857= 9.3044

W_{l10} = 10.857 x 0.857= 9.3044

Таким чином, визначено клас з найвищим значенням репутації для правої руки W_{r1}=Tr2, що дорівнює 8 для правої руки, та клас з найвищим значенням репутації для лівої руки W_{l1}= W_{l3}= W_{l4}= W_{l5}= W_{l6}= W_{l7}= W_{l8}=Tr3, що дорівнює 11.

За другим запропонованим методом об'єднання на основі репутації відбувається наступним чином. Визначено перше порогове значення θ_1 = 10.5 та друге порогове значення θ_2 = 7.8. Якщо $R_i \geq \theta_1$, то класифіковані спостереження прийняли ідеальні рішення, якщо $\theta_1 > R_i \geq \theta_2$, то класифіковані спостереження прийняли правильні рішення, і якщо $\theta_2 \geq R_i$, тоді класифіковані спостереження прийняли невірні рішення. Визначено порогові значення наступним чином: 0,5 балів за низьку ефективність класифікації, 1 за нормальну ефективність класифікації і 2 за ідеальну ефективність класифікації. За формулою (2.3) розраховано S_i для кожного спостереження, результат розрахунків представлено в табл. 3.3.

Таблиця 3.3 – Результат розрахунків S_i

Тест правої руки					Тест лівої руки				
ID	Вага Acc_i	R_i	S_i	Прогнозований клас	ID	Вага Acc_i	R_i	S_i	Прогнозований клас
1	1	8	1	Tr2	1	1	11	2	Tr3
2	0.75	7.75	0.5	Tr2	2	0.857	10.857	2	Tr1
3	0.75	7.75	0.5	Tr2	3	1	11	2	Tr3
4	0.75	7.75	0.5	Tr2	4	1	11	2	Tr3

5	0.75	7.75	0.5	Tr2	5	1	11	2	Tr3
6	0.75	7.75	0.5	Tr2	6	1	11	2	Tr3
7	0.75	7.75	0.5	Tr2	7	1	11	2	Tr3
8	0.75	7.75	0.5	Tr2	8	1	11	2	Tr3
9	0.75	7.75	0.5	Tr2	9	0.857	10.857	2	Tr1
10	0.75	7.75	0.5	Tr2	10	0.857	10.857	2	Tr1

Далі розраховано значення репутації для кожного спостереження з використанням формули (2.2). Таким чином, визначено клас з найвищою репутацією *W*. Для 10 спостережень тесту на тремор правої руки він розраховується наступним чином.

W_{r1} = 1 x 1 = 1

W_{r2} = 0.5 x 0.75 = 0.375

W_{r3} = 0.5 x 0.75 = 0.375

W_{r4} = 0.5 x 0.75 = 0.375

W_{r5} = 0.5 x 0.75 = 0.375

W_{r6} = 0.5 x 0.75 = 0.375

W_{r7} = 0.5 x 0.75 = 0.375

W_{r8} = 0.5 x 0.75 = 0.375

W_{r9} = 0.5 x 0.75 = 0.375

W_{r10} = 0.5 x 0.75 = 0.375

Для 10 спостережень тесту на тремор лівої руки він розраховується наступним чином.

W_{l1} = 2 x 1 = 2

W_{l2} = 2 x 0.857= 1.714

W_{l3} = 2 x 1 = 2

W_{l4} = 2 x 1 = 2

W_{l5} = 2 x 1 = 2

W_{l6} = 2 x 1 = 2

W_{l7} = 2 x 1 = 2

W_{l8} = 2 x 1 = 2

W_{l9} = 2 x 0.857= 1.714

W_{l10} = 2 x 0.857= 1.714

Таким чином, визначено клас з найвищим значенням репутації для правої руки W_{r1}=Tr2, що дорівнює 1 для правої руки, та клас з найвищим значенням репутації для лівої руки W_{l1}= W_{l3}= W_{l4}= W_{l5}= W_{l6}= W_{l7}= W_{l8}=Tr3, що дорівнює 2.

Як можна побачити, за визначених порогових значень обидва запропоновані методи показали однакові класи з найвищою репутацією.

Таким чином, використовуючи голосування більшістю, для першого та другого методу об’єднання на підставі репутації, визначено загальну оцінку ступеню тяжкості тремору, яку визначає клас з найвищим значенням репутації для обох рук. Загальна оцінка ступеня тяжкості тремору визначено як клас Tr3, тобто помірний тремор.

3.4.4 Розрахунок обчислювальної складності моделі на основі об’єднання і голосування більшістю

Складність процесу класифікації моделі на основі злиття з використанням дерев рішень становить $O(m_1 + m_2 + \cdots + m_n) = O(m)$; де n - кількість вузлів, що беруть участь в класифікації, а m - глибина дерева рішень. Розглянемо голосування на підставі репутації за другим методом, що використовує два порогові значення для визначення репутації для результатів лівої руки. Оскільки голосування не залежить від класифікації, його тимчасова складність додається до часу класифікації, як показано в рівняннях 2.4, 2.5, 2.6. Рівняння 2.7 представляє остаточну складність моделі, заснованої на об'єднанні, з використанням дерев рішень і голосування більшістю.

OO (Розподілений підхід з використанням голосування більшістю) = O(max отримане) = O(c) =
= 7.64353
O (*O*(*Розподілений підхід з використанням голосування більшістю*)= 7.64353 + *O*([1.290633+3.716607+3.716607+1.290633+1.838338+1.891277+2.438984+2.438984+1.891277+1.838338+7.64353+1.891277+1.891277+7.64353])) = 49.064822

Таким чином, розрахована обчислювальна складність запропонованої технології дорівнює 49.064822.

3.5 Висновок до третього розділу

В третьому розділі представлена програмна реалізація системи біометричного моніторингу та практична реалізація технології об'єднання даних на основі репутацій. Запропонована технологія має наступні переваги: розподілена обробка та міркування, які скорочують час обробки даних і забезпечують швидке виявлення тремору, стійкість до збоїв датчиків і точне виявлення за рахунок використання об'єднання даних сенсорів, які використовуються для проведення тестів на виявлення симптомів.

Представлена програмна реалізація розробленої системи віддаленого біометричного моніторингу та її структурних елементів: мобільний додаток, додаток Windows, серверна частина.

Проведено експеримент з даними, отриманими після проведення тестів на перевірку ступеня тяжкості тремору. Дані отрімані відповідно до шкали MDS-UPDRS для оцінки постурального тремору. Ця шкала має п'ять ступенів тяжкості тремору и відповідну Їм амплітуду. Виконання автономного аналізу показує високу складність набору даних, оскільки дані, що представляють п'ять класів, мають високу ступінь перекриття. Результати оцінки показують, що, з огляду на високу складність набору даних, запропонована технологія об'єднання даних, яка включає локальну модель дерева рішень і глобальну модель об'єднання на основі репутації, яка включає методи визначення репутації кожного спостереження.

Таким чином, вирішено поставлене завдання - розробка технології об’єднання даних в системах віддаленого біометричного моніторингу.

3.6 Перелік посилань до третього розділу

1. Back4app. Low-code backend to build modern apps. Back4app.com. URL: https://www.back4app.com/ (дата звернення 20.11.2020)

2. Webmin. Webmin.com. URL: https://www.webmin.com/ (дата звернення 20.11.2020)

3. Mikrotik. Mikrotik.com. URL: https://mikrotik.com/download (дата звернення 20.11.2020)

4. Laravel. The PHP Framework for Web Artisans. Laravel.com URL: https://laravel.com/ (дата звернення 20.11.2020)

5. MeCo. Meco.turion.info. URL: http://meco.turion.info/ (дата звернення 20.11.2020)

6. Goetz C.G. et al. Movement Disorder Society-Sponsored Revision of the Unified Parkinson’s Disease Rating Scale (MDS-UPDRS): Scale Presentation and Clinimetric Testing Results // Movement Disorders, 2008. – Vol. 23(15). – P. 2129–2170.

РОЗДІЛ 4 ОХОРОНА ПРАЦІ

4.1 Загальні питання з охорони праці

Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці. В законі України «Про охорону праці» [1] визначається, що охорона праці - це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності.

При роботі з обчислювальною технікою змінюються фізичні і хімічні фактори навколишнього середовища: виникає статична електрика, електромагнітне випромінювання, змінюється температура і вологість, рівень вміст кисню і озону в повітрі. Повітря забруднюється шкідливими хімічними речовинами антропогенного походження за рахунок деструкції полімерних матеріалів, які використовуються для обробки приміщень та обладнання. Неправильна організація робочого місця сприяє загальному і локальній напрузі м'язів шиї, тулуба, верхніх кінцівок, викривлення хребта і розвитку остеохондрозу. На всіх підприємствах, в установах, організаціях повинні створюватися безпечні і нешкідливі умови праці. Забезпечення цих умов покладається на власника або уповноважений ним орган (далі роботодавець). Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці. Роботодавець повинен впроваджувати сучасні засоби техніки безпеки, які запобігають виробничому травматизмові, і забезпечувати санітарно-гігієнічні умови, що запобігають виникненню професійних захворювань працівників. Він не має права вимагати від працівника виконання роботи, поєднаної з явною небезпекою для життя, а також в умовах, що не відповідають законодавству про охорону праці. Працівник має право відмовитися від дорученої роботи, якщо створилася виробнича ситуація, небезпечна для його життя чи здоров'я або людей, які його оточують, і навколишнього середовища.

4.1.1 Правові та організаційні основи охорони праці

Основним організаційним напрямом у здійсненні управління в сфері охорони праці є усвідомлення пріоритету безпеки праці і підвищення соціальної відповідальності держави, і особистої відповідальності працівників.

Державна політика в галузі охорони праці визначається відповідно до Конституції України Верховною Радою України і спрямована на створення належних, безпечних і здорових умов праці, запобігання нещасним випадкам та професійним захворюванням. Відповідно до статті 3 Закону України «Про охорону праці» [1] (далі – Закону) законодавство про охорону праці складається з Закону, Кодексу законів про працю України [2], Закону України "Про загальнообов'язкове державне соціальне страхування від нещасного випадку на виробництві та професійного захворювання, які спричинили втрату працездатності" [3] та прийнятих відповідно до них нормативно-правових актів, норм міжнародного договору (ратифіковані Конвенції і Рекомендації МОТ, директиви Європейської Ради).

На законодавчому рівні визначено такі пріоритетні напрямки з безпеки праці:

– кожен працівник несе безпосередню відповідальність за порушення зазначених Законом, нормами і правилами вимог;

– напрямки реалізації конституційного права громадян на їх життя і здоров’я в процесі трудової діяльності:

– пріоритет життя і здоров’я працівників по відношенню до результатів виробничої діяльності підприємства;

– повна відповідальність роботодавця за створення належних – безпечних і здорових умов праці;

– соціальний захист працівників, повне відшкодування збитків особам, які потерпіли від нещасних випадків на виробництві та професійних захворювань;

– комплексне розв’язання завдань охорони праці;

– підвищення рівня промислової безпеки шляхом забезпечення суцільного технічного контролю за станом виробництв, технологій та продукції, а також сприяння підприємствам у створенні безпечних та нешкідливих умов праці;

– соціальний захист працівників, повне відшкодування збитків особам, які потерпіли від нещасних випадків на виробництві та професійних захворювань;

– використання економічних методів управління охороною праці, участь держави у фінансуванні заходів щодо охорони праці;

– використання світового досвіду організації роботи щодо поліпшення умов і підвищення безпеки праці на основі міжнародної співпраці.

Користувачі персональних комп'ютерів, для яких ця робота є головною, підлягають медичним оглядам: попереднім — під час влаштування на роботу і періодичним — протягом професійної діяльності раз на два роки. Жінок з часу встановлення вагітності та в період годування дитини грудьми до роботи з ПК не допускають.

Наявні трудові відносини між працівниками і роботодавцями в Україні за темою дипломного проекту регулюються Кодексом законів про працю (КЗпП) України, відповідно до якого права працюючої людини на охорону праці охороняються всебічно та норми охорони праці неухильно інтегровані до правил внутрішнього розпорядку організації/підприємства.

4.1.2 Організаційно-технічні заходи з безпеки праці

В організації/підприємстві проводиться навчання і перевірка знань з питань охорони праці відповідно до вимог Типового положення про порядок проведення навчання і перевірки знань з питань охорони праці, затвердженого наказом Держнаглядохоронпраці України від 26.01.2005 N 15, зареєстрованого в Міністерстві юстиції України 15.02.2005 за N 231/10511 [4].

Обов'язковими вимогами враховане наступне:

– ознайомлення з правилами безпеки праці, одержання відповідних інструктажів засвідчується у журналі інструктажів.

– перед допуском до самостійної роботи кожен працівник має право на навчання з питань охорони праці і роботодавець зобов'язаний, і проводить таке навчання у вигляді двох інструктажів з питань охорони праці:

1) вступного, який проводять працівники служби охорони праці об'єкта господарювання з усіма працівниками, яких приймають на роботу незалежно від їхньої освіти та стажу роботи за програмою, в якій подають загальні питання охорони праці із врахуванням її особливостей на об'єкті господарювання;

2) первинного, який проводять керівники структурних підрозділів на місці праці з кожним працівником до початку їхньої роботи на цьому робочому місці.

Проходження працівником цих інструктажів з питань охорони праці підтверджується записами у відповідних журналах обліку інструктажів і скріплюється підписами осіб, які проводили інструктажі та осіб, які отримали інструктажі.

3) Повторний (не рідше одного разу в 6 місяців);

4) Позаплановий (при зміні правил охорони праці);

5) Поточний (проводять з працівниками перед виконанням робіт, на яких оформляється наряд-допуск)

– обов’язкові організаційні заходи перед початком, під час і після завершення роботи повинні включати перевірку (візуально) наявності і справності електрообладнання та його заземлення, а під час виконання роботи вимогу «не залишати без нагляду обладнання, яке працює». Після закінчення роботи - вимагається прибирання робочого місця, відключення всіх електроприладів від електромережі.

4.2 Аналіз стану умов праці

Робота над дослідженням методів та алгоритмів аналізу даних моніторингу біофізичних показників проходила в приміщенні будинку проспект Центральний, 5. Для даної роботи достатньо однієї людини, для якої надано робоче місце зі стаціонарним комп’ютером.

4.2.1 Вимоги до приміщень

Геометричні розміри приміщення зазначені в таблиці 4.1.

Таблиця 4.1 – Розміри приміщення

Найменування	**Значення**
Довжина, м	5
Ширина, м	5
Висота, м	3
Площа, $м^2$	25
Об’єм, $м^3$	75

Згідно з вимогами Державних санітарних правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин (ДСанПІН 3.3.2.007-98) [5] розмір площі для одного робочого місця оператора персонального комп’ютера має бути не менше 6 кв. м, а об’єм — не менше 20 куб. м. Отже, дане приміщення цілком відповідає зазначеним нормам.

Для забезпечення потрібного рівного освітленості кімната має вікна та систему загального рівномірного освітлення, що встановлена на стелі. Для дотримання вимог пожежної безпеки встановлено порошковий вогнегасник та систему автоматичної пожежної сигналізації.

4.2.2 Вимоги до організації робочого місця

При порівнянні відповідності характеристик робочого місця нормативним основні вимоги до організації робочого місця за ДСанПІН 3.3.2.007-98 [5] і відповідними фактичними значеннями для робочого місця за ДСН 3.3.6.042-99 [6], констатуємо повну відповідність.

Таблиця 4.2 - Характеристики робочого місця

Найменування параметра	**Фактичне значення**	**Нормативне значення**
Висота робочої поверхні, мм	750	680 ÷800
Висота простору для ніг, мм	700	не менше 600
Ширина простору для ніг, мм	700	не менше 500
Глибина простору для ніг, мм	750	не менше 650
Висота поверхні сидіння, мм	450	400 ÷500
Ширина сидіння, мм	400	не менше 400
Глибина сидіння, мм	400	не менше 400
Висота поверхні спинки, мм	500	не менше 300
Ширина опорної поверхні спинки, мм	400	не менше 380
Радіус кривини спинки в горизонтальній площині, мм	400	400
Відстань від очей до екрану дисплея, мм	800	700 ÷800

Приміщення кабінету має об’єм 75 м^3, площу - 25 м^2.

Температура в приміщенні протягом року коливається у межах 16–30°С, відносна вологість - близько 50%. Система вентилювання приміщення - природна неорганізована, а опалення - централізоване.

Розміщення вікон забезпечує природне освітлення з коефіцієнтом природного освітлення не менше 1,5%, а загальне штучне освітлення, яке здійснюється за допомогою двох люмінесцентних ламп, забезпечує рівень освітленості не менше 200 Лк.

За ступенем пожежної безпеки приміщення належить до категорії В.

4.2.3 Навантаження та напруженість процесу праці

За фізичним навантаженням робота відноситься до категорії легкі роботи (Іа), її виконують сидячи з періодичним ходінням. Щодо характеру організування виконання дипломної роботи, то він підпадає під нав'язаний режим, оскільки певні розділи роботи необхідно виконати у встановлені конкретні терміни. За ступенем нервово-психічної напруги виконання роботи можна віднести до II – III ступеня і кваліфікувати як помірно напружений – напружений за умови успішного виконання поставлених завдань.

Під час виконання робіт використовують ПК та периферійні пристрої (лазерні та струменеві), що призводить до навантаження на окремі системи організму. Такі перекоси у напруженні різних систем організму, що трапляються під час роботи з ПК, зокрема, значна напруженість зорового аналізатора і довготривале малорухоме положення перед екраном, не тільки не зменшують загального напруження, а навпаки, призводять до його посилення і появи стресових реакцій.

Наявні психофізіологічні небезпечні та шкідливі фактори:

а) фізичного перевантаження:

- статичного;
- динамічного;

б) нервово-психічного перевантаження:

- розумового перенапруження;
- монотонності праці;
- перенапруження аналізаторів;
- емоційних перевантажень.

Роботу за дипломним проектом визнано, такою, що займає 50% часу робочого дня та за восьмигодинної робочої зміни рекомендовано встановити додаткові регламентовані перерви - для розробників програм тривалістю 15 хв через кожну годину роботи.

4.3 Виробнича санітарія

На підставі аналізу небезпечних та шкідливих факторів при виробництві (експлуатації), пожежної безпеки можуть бути надалі вирішені питання необхідності забезпечення працюючих достатньою кількістю освітлення, вентиляції повітря, організації заземлення, тощо.

4.3.1 Аналіз небезпечних та шкідливих факторів при виробництві (експлуатації) виробу

Роботу, пов'язану з електронно-обчислювальними машинами (далі - ЕОМ) з відео дисплейними терміналами (далі - ВДТ), у тому числі на тих, які мають робочі місця, обладнані ЕОМ з ВДТ і ПП, виконують із забезпеченням виконання НПАОП 0.00-7.15-18 [9] «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями», які встановлюють вимоги безпеки до обладнання робочих місць, до роботи із застосуванням ЕОМ з ВДТ і ПП. Переважно роботи за проектами виконують у кабінетах чи інших приміщеннях, де використовують різноманітне електрообладнання, зокрема персональні комп’ютери (ПК) та периферійні пристрої.

Основними робочими характеристиками персонального комп'ютера є наступні:

–робоча напруга U = +220В ± 5%;

–робочий струм I = 2А;

–споживана потужність P = 500 Вт.

Робоче місце має відповідати вимогам Державних санітарних правил і норм роботи з візуальними дисплейними терміналами електронно-обчислювальних машин, затверджених постановою Головного державного санітарного лікаря України від 10.12.98 N 7 [5].

Аналіз небезпечних та шкідливих виробничих факторів виконується у табличній формі (табл. 4.3).

Таблиця 4.3 – Аналіз небезпечних і шкідливих виробничих факторів

Небезпечні і шкідливі виробничі фактори	Джерела факторів (види робіт)	Кількісна оцінка	Нормативні документи
1	2	3	4
Фізичні			
- підвищений рівень напруги електричної мережі, замикання якої може відбутися через тіло людини	-//-	4	[7]
- недостатність природного світла	порушення умов праці (вимог до приміщень)	2	[8]
- недостатнє освітлення робочої зони	порушення гігієнічних параметрів виробничого середовища	3	[8]
Психофізіологічні:			
- нервово-психічна перевантаження (розумове, перенапруження аналізаторів-зорових)	- пошук інформації для постановки теми; - пошук та аналіз аналогів і літератури; - пошук наявних технологій; - виконання роботи за темою диплома, тестування; - оформлення роботи	4	[9] [5]
- фізичні (статичне – сидіння)	порушення умов праці (організації місця праці- сидіння користувача,) та організації робочого часу - безпервна робота)	2	[9] [5]

4.3.2 Пожежна безпека

Пожежна безпека при застосуванні ПК забезпечується: системою запобігання пожежі, системою протипожежного захисту, організаційно-технічними заходами.

Згідно ДСТУ Б В.1.1-36:2016 «Визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою» [10] таке приміщення, площею 25 м^2, відноситься до категорії "В" (пожеженебезпечної) та для протипожежного захисту в ньому проектом передбачено устаткування автоматичною пожежною сигналізацією із застосуванням датчиків-сповіщувачів РІД-1 (сповіщувач димовий ізоляційний) в кількості 1 шт., і застосуванням первинних засобів пожежогасіння.

Продуктами згорання, що виділяються на пожежі, є: окис вуглецю; сірчистий газ; окис азоту; синильна кислота; акромін; фосген; хлор і ін. При горінні пластмас, окрім

звичних продуктів згорання, виділяються різні продукти термічного розкладання: хлорангідридні кислоти, формальдегіди, хлористий водень, фосген, синильна кислота, аміак, фенол, ацетон, стирол.

Для захисту персоналу від дії небезпечних і шкідливих чинників пожежі проектом передбачається застосування промислового протигаза.

4.4 Гігієнічні вимоги до параметрів виробничого середовища

4.4.1 Мікроклімат

Мікроклімат робочих приміщень – це клімат внутрішнього середовища цих приміщень, що визначається діючим на організм людини з'єднанням температури, вологості, швидкості переміщення повітря. Оптимальні значення для температури, відносної вологості й рухливості повітря для робочого місця відповідають ДСН 3.3.6.042-99 [6] і наведені в таблиці 4.4.

Таблиця 4.4 – Норми мікроклімату робочої зони об’єкту

Період року	Категорія робіт	Температура C^0	Відносна вологість %	Швидкість руху повітря, м/с
Холодний	легка-1 а	22 - 24	40 – 60	0,1
Теплий	легка-1 а	23 - 25	40 – 60	0,1

Для забезпечення оптимальних параметрів мікроклімату в приміщенні проводяться перерви в роботі користувача, з метою його провітрювання.

4.4.2 Освітлення робочого місця

Світло є природною умовою існування людини. Освітленість приміщення має велике значення при роботі на ППК. Основний потік природного світла повинен бути зліва. Не допускається спрямування основного світлового потоку природного світла праворуч, ззаду і спереду працівника на ППК.

У приміщенні, де розташовані ПК передбачається природне бічне освітлення, рівень якого відповідає. ДБН В.2.5-28:2018 «Природнє і штучне освітлення» [8]. Регулярно повинен проводитися контроль освітленості, який підтверджує, що рівень освітленості задовольняє СНіП і для даного приміщення в світлий час доби достатньо природного освітлення.

Розрахунок освітлення.

Для будівель виробництв світловий коефіцієнт приймається 1/6 - 1/10:

$$\sqrt{a^2 + b^2} \cdot S_b = (1/8 \div 1/10) \cdot S_n \quad (4.1)$$

де S_b – площа віконних прорізів, м^2;

S_n – площа підлоги, м^2.

$S_n = a \cdot b = 5 \cdot 5 = 25$ м^2

$S_{вік} = 1/8 \cdot 25 = 3{,}125$ м^2

Приймаємо 2 вікна площею $S = 1{,}6$ м^2 кожне.

Розрахунок штучного освітлення виробляється по коефіцієнтах використання світлового потоку, яким визначається потік, необхідний для створення заданої освітленості при загальному рівномірному освітленні. Розрахунок кількості світильників N виробляється по формулі (4.2):

$$n = \frac{E \cdot S \cdot Z \cdot K}{F \cdot U \cdot M} \quad (4.2)$$

де E – нормована освітленість робочої поверхні, за нормами – 300 лк;

S – освітлювана площа, м2; S = 25 м2;

Z – поправочний коефіцієнт світильника (для стандартних світильників Z = 1.1 - 1.3) приймаємо рівним 1,1;

K – коефіцієнт запасу, що враховує зниження освітленості в процесі експлуатації – 1,5;

U – коефіцієнт використання, залежний від типу світильника, показника індексу приміщення і т.п. – 0,575

M – число люмінесцентних ламп в світильнику – 2;

F – світловий потік лампи – 5400лм.

Підставивши числові значення у формулу (4.2), отримуємо:

$$n = \frac{300 \cdot 25 \cdot 1{,}1 \cdot 1{,}5}{5400 \cdot 0{,}575 \cdot 2} \approx 1{,}99$$

Встановити 1,99 світильники ми не можемо, тому округляємо це значення до 2 і приймаємо освітлювальну установку, яка складається з 2-х світильників, які складаються з двох люмінесцентних ламп загальною потужністю 160 Вт, напругою – 220 В.

Потужність електроосвітлювальної установки з урахуванням місцевого освітлення визначається за формулою:

$$N = \frac{n \cdot W + (0,1 \div 0,2) \cdot n \cdot W}{1000}, \kappa Вт \quad (4.3)$$

де n– розрахункова кількість ламп для освітлення даного приміщення;

W – потужність однієї лампи, Вт;

(0,1ч0,2) – додаткова потужність для ламп місцевого освітлення, Вт

$$N = \frac{3 \cdot 160 + 0,2 \cdot 3 \cdot 160}{1000} = 0,576 \ \kappa Вт$$

4.5 Шум, вібрація та електромагнітне випромінювання

Рівень шуму, що супроводжує роботу користувачів персональних комп'ютерів, а також зовнішніми чинниками, коливається у межах 50–65 дБА (ДСН 3.3.6.037-99) [14]. У залах опрацювання інформації та комп'ютерного набору рівні шуму не повинні перевищувати 65 дБА.

Вібрація на робочому місці в приміщенні, що розглядається, відповідає нормам ДСН 3.3.6.037-99 [14].

4.6 Розрахунок захисного заземлення

Згідно з класифікацією приміщень за ступенем небезпеки ураження електричним струмом НПАОП 40.1-1.01-97 [45], приміщення в якому проводяться всі роботи відноситься до першого класу (без підвищеної небезпеки). Під час роботи використовуються електроустановки з напругою живлення 36 В, 220 В. Опір контуру заземлення повинен мати не більше 4 Ом.

Розрахунок проводять за допомогою методу коефіцієнта використання (екранування) електродів. Коефіцієнт використання групового заземлювача з – це відношення діючої провідності цього заземлювача до найбільш можливої його провідності за нескінченно великих відстаней між його електродами. Коефіцієнт використання вертикальних заземлювачів в залежності від розміщення заземлювачів та їх кількості знаходиться в межах 0,4…0,99. Взаємну екрануючу дію горизонтального заземлювача (з'єднувальної смуги) враховують за допомогою коефіцієнта використання горизонтального заземлювача зс.

Визначимо необхідний опір штучних заземлювачів Rшт.з.:

$$R_{шт.з.} = \frac{R_д \cdot R_{пр.з.}}{R_{пр.з.} - R_д}, \tag{4.4}$$

де Rпр.з. – опір природних заземлювачів; Rд – допустимий опір заземлення. Якщо природні заземлювачі відсутні, то Rшт.з.=Rд.

Підставивши числові значення у формулу (4.4), отримуємо:

$$R_{шт.з.} = \frac{4 \cdot 40}{40 - 4} \approx 4 \text{ Ом}$$

Опір заземлення в значній мірі залежить від питомого опору ґрунту с, Ом·м. Приблизне значення питомого опору глини приймаємо с=40 Ом•м (табличне значення).

Розрахунковий питомий опір ґрунту, срозр, Ом·м, визначається для заземлювачів срозр.в, срозр.г, Ом·м за формулою:

$$\rho_{розр..} = \psi \cdot \rho, \tag{4.5}$$

де ш – коефіцієнт сезонності для вертикальних заземлювачів I кліматичної зони з нормальною вологістю землі, приймається для вертикальних заземлювачів срозр.в=1,7 і горизонтальних срозр.г=5,5 Ом·м.

$$\rho_{розр.в} = 1,7 \cdot 40 = 68 \text{ Ом•м}$$

$$\rho_{розр.г} = 5,5 \cdot 40 = 220 \text{ Ом•м}$$

Розрахуємо опір розтікання струму вертикального заземлювача Rв, Ом

$$R_в = \frac{\rho_{розр.в}}{2 \cdot \pi \cdot l_в} \cdot \left(\ln \frac{2 \cdot l_в}{d_{ст}} + \frac{1}{2} \cdot \ln \frac{4 \cdot t + l_в}{4 \cdot t - l_в} \right), \quad (4.6)$$

де lв – довжина вертикального заземлювача (для труб - 2−3 м; lв=3 м);

dст – діаметр стержня (для труб - 0,03−0,05 м; dст=0,05 м);

t – відстань від поверхні землі до середини заземлювача, яка визначається за ф. (4.7):

$$t = h_в + \frac{l_в}{2}, \quad (4.7)$$

де hв – глибина закладання вертикальних заземлювачів (0,8 м); тоді

$$t = 0,8 + \frac{3}{2} = 2,3$$

$$R_в = \frac{68}{2 \cdot \pi \cdot 3} \cdot \left(\ln \frac{2 \cdot 3}{0,05} + \frac{1}{2} \cdot \ln \frac{4 \cdot 2,3 + 3}{4 \cdot 2,3 - 3} \right) = 18,5 \text{ Ом}$$

Визначаємо теоретичну кількість вертикальних заземлювачів n штук, без урахування коефіцієнта використання зв:

$$n = \frac{2 \cdot R_в}{R_д} = \frac{2 \cdot 18,5}{4} = 9,25 \quad (4.8)$$

Визначаємо коефіцієнт використання вертикальних електродів групового заземлювача без врахування впливу з'єднувальної стрічки зв =0,57 (табличне значення).

Визначаємо необхідну кількість вертикальних заземлювачів з урахуванням коефіцієнта використання nв, шт:

$$n_в = \frac{2 \cdot R_в}{R_д \cdot \eta_в} = \frac{2 \cdot 18,5}{4 \cdot 0,57} = 16,2 \approx 16 \quad (4.9)$$

Визначаємо довжину з'єднувальної стрічки горизонтального заземлювача lс, м:

$$l_с = 1,05 \cdot L_в \cdot (n_в - 1), \quad (4.10)$$

де Lв – відстань між вертикальними заземлювачами, (прийняти Lв = 3м);

nв – необхідна кількість вертикальних заземлювачів.

$$l_c = 1,05 \cdot 3 \cdot (16-1) \approx 48_{\text{м}}$$

Визначаємо опір розтікання струму горизонтального заземлювача (з'єднувальної стрічки) Rг, Ом:

$$R_г = \frac{\rho_{розр.г}}{2 \cdot \pi \cdot l_c} \cdot \ln \frac{2 \cdot l_c^{\,2}}{d_{см} \cdot h_г}, \qquad (4.11)$$

де dсм – еквівалентний діаметр смуги шириною b, dсм =0,95b, b = 0,15 м;

hг – глибина закладання горизантальних заземлювачів (0,5 м);

l_c - довжина з'єднувальної стрічки горизонтального заземлювача lс, м

$$R_г = \frac{220}{2 \cdot \pi \cdot 48} \cdot \ln \frac{2 \cdot 48^2}{0,95 \cdot 0,15 \cdot 0,5} = 8,1 \text{ Ом}$$

Визначаємо коефіцієнт використання горизонтального заземлювача зс. відповідно до необхідної кількості вертикальних заземлювачів nв.

Коефіцієнт використання з'єднувальної смуги зс=0,3 (табличне значення).

Розраховуємо результуючий опір заземлювального електроду з урахуванням з'єднувальної смуги:

$$R_{заг} = \frac{R_в \cdot R_г}{R_в \cdot \eta_c + R_г \cdot n_в \cdot \eta_в} \le R_д. \qquad (4.12)$$

Висновок: дане захисне заземлення буде забезпечувати електробезпеку будівлі, так як виконується умова: Rзаг <4 Ом, а саме:

$$R_{заг} = \frac{18,5 \cdot 8,1}{18,5 \cdot 0,3 + 8,1 \cdot 16 \cdot 0,57} = 1,9 \le R_д$$

4.7 Екологія

Діяльність за темою магістерської роботи, «Методи та алгоритми аналізу даних моніторингу біофізичних показників» впливає на навколишнє природнє середовище і регламентується нормами діючого законодавства: Законом України «Про охорону навколишнього природного середовища» [11], Законом України «Про забезпечення

санітарного та епідемічного благополуччя населення» [12], Законом України «Про відходи» [13].

В процесі діяльності з виконанням дипломного проектуванням виникають процеси поводження з відходами ІТ галузі. Нижче надано перелік відходів, що утворюються в процесі роботи:

- Відпрацьовані люмінесцентні лампи - I клас небезпеки.
- Змінні носії інформації - IV клас небезпеки.
- Відпрацьовані вогнегасники - IV клас небезпеки.
- Макулатура - IV клас небезпеки.

4.8 Висновки до четвертого розділу

У даному розділі проведено аналіз умов праці, шкідливих та небезпечних чинників. Визначено параметри і характеристики приміщення для роботи над магістерською роботою «Методи та алгоритми аналізу даних моніторингу біофізичних показників», заходи, які потрібно зробити для того, щоб дане приміщення відповідало необхідним нормам і було комфортним і безпечним для роботи. Приведені рекомендації щодо організації робочого місця, електробезпеки та пожежної безпеки. Наведені розміри приміщення та значення температури, вологості й рухливості повітря, необхідна кількість і потужність ламп та інші параметри, значення яких впливає на умови праці, рекомендації з охорони праці, техніки безпеки при роботі на комп’ютері.

4.9 Перелік посилань до четвертого розділу

1. Закон України "Про охорону праці". Вводиться в дію Постановою ВР № 2695-XII від 14.10.92, ВВР, 1992, № 49, ст.669. - Режим доступу: www. URL: https://zakon.rada.gov.ua/laws/show/2694-12

2. Кодекс законів про працю України. Затверджується Законом № 322-VIII від 10.12.71 ВВР, 1971. Режим доступу: www. URL: https://zakon.rada.gov.ua/laws/show/322-08

3. Закон України "Про загальнообов'язкове державне соціальне страхування від нещасного випадку на виробництві та професійного захворювання, які спричинили втрату працездатності". Наказ від 21 грудня 2000 року N 2180-III. Режим доступу: www. URL: https://dnaop.com/html/2065/doc-zakon-ukrajini-pro-zagalynoobovjazkove-derzhavne-socialyne-strahuvannya-vid-neshhasnogo-vipadku-na-virobnictvi-ta-profesijnogo-z

4. Про затвердження Типового положення про порядок проведення навчання і перевірки знань з питань охорони праці НПАОП 0.00-4.12-05. Наказ від 26.01.2005 №15. Режим доступу: www. URL: https://zakon.rada.gov.ua/laws/show/z0231-05

5. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин ДСанПІН 3.3.2.007-98. Затверджено Постановою Головного державного санітарного лікаря України 10 грудня 1998 р. N 7. Режим доступу: www. URL: https://zakon.rada.gov.ua/rada/show/v0007282-98

6. Санітарні норми мікроклімату виробничих приміщень ДСН 3.3.6.042-99. Постанова N 42 від 01.12.99. Режим доступу: www. URL: https://zakon.rada.gov.ua/rada/show/va042282-99

7. Електробезпека в будівлях і спорудах. Вимоги до захисних заходів від ураження електричним струмом. Наказ від 1 липня 2016 року N 204. Режим доступу: www. URL: http://epicentre.co.ua/dstu/doc28522.html

8. ДБН В.2.5-28:2018 «Природне і штучне освітлення». Режим доступу: www. URL: http://www.minregion.gov.ua/wp-content/uploads/2018/12/V2528-1.pdf

9.НПАОП 0.00-7.15-18 «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями». Зареєстровано в Міністерстві юстиції України 25 квітня 2018 р. за № 508/31960. Режим доступу: www. URL: https://zakon.rada.gov.ua/laws/show/z0508-18

10. ДСТУ Б В.1.1-36:2016 «Визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою». Наказ від 15.06.2016 №158. Режим доступу: www. URL: https://zakon.rada.gov.ua/rada/show/v0158858-16

11. Закон України «Про охорону навколишнього природного середовища» . Вводиться в дію Постановою ВР № 1268-XII від 26.06.91, ВВР, 1991, № 41, ст.547. Режим доступу: www. URL: https://zakon.rada.gov.ua/laws/show/1264-12

12. Закони України «Про охорону навколишнього природного середовища». Вводиться в дію Постановою ВР № 4005-XII від 24.02.94, ВВР, 1994, № 27, ст.219. Режим доступу: www. URL: https://zakon.rada.gov.ua/laws/show/4004-12

13. Закон України «Про відходи». Відомості Верховної Ради України (ВВР), 1998, № 36-37, ст.242. Режим доступу: www. URL: https://zakon.rada.gov.ua/laws/show/187/98-вр

14. ДСН 3.3.6.037-99 «Державні санітарні норми виробничого шуму, ультразвуку та інфразвуку» Режим доступу: https://zakon.rada.gov.ua/go/va037282-99- 31.05.2019 р.

15. НПАОП 40.1-1.01-97 Правила безопасной эксплуатации электроустановок Режим доступу: http://zakon.rada.gov.ua/laws/show/z0011-98- 31.05.2019 р.

ВИСНОВКИ

Метою дипломної роботи визначена розробка технології об’єднання даних в системах віддаленого біометричного моніторингу.

В ході дослідницької частини роботи були отримані наступні результати:

1. Проведено аналіз структури системи віддаленого біометричного моніторингу, методів, технологій, засобів отримання, обробки, аналізу та об’єднання даних в системах біометричного моніторингу;
2. Спроектовано та розроблено архітектуру системи віддаленого біометричного моніторингу;
3. Розроблено технологію об’єднання даних для системи віддаленого біометричного моніторингу.

В ході практичної частини роботи були отримані наступні результати:

1. Програмно реалізовано структурні елементи системи віддаленого біометричного моніторингу: мобільний додаток, додатку Windows, серверної частини;
2. Практично реалізовано технологію об’єднання даних для системи віддаленого біометричного моніторингу;
3. Визначено ефективність запропонованої технології об’єднання даних.

Таким чином, вирішено поставлене завдання - підвищення точності оцінки фізіологічних параметрів стану людини на підставі біометричних даних за рахунок розроблення технології об’єднання даних в системах віддаленого біометричного моніторингу.

ДОДАТОК А ЕЛЕКТРОННА ПРЕЗЕНТАЦІЯ

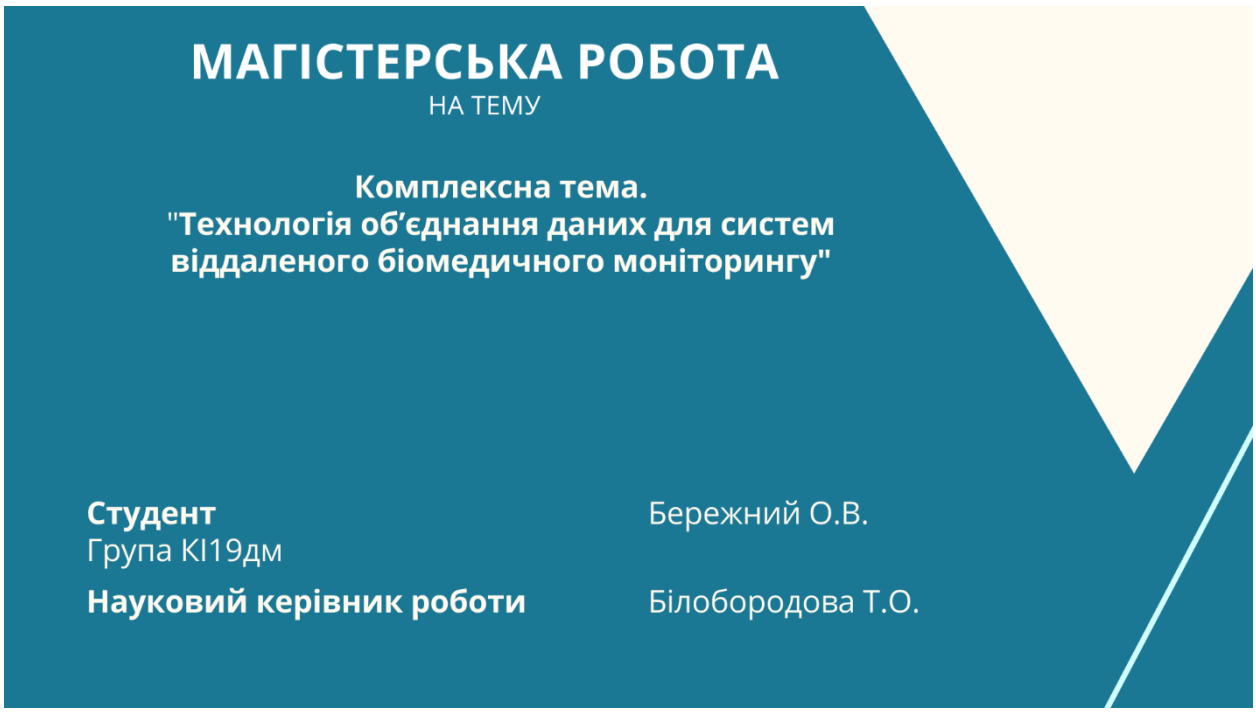


Рисунок А.1 – Слайд 1

Рисунок А.2 – Слайд 2

Постановка завдання

Об'єкт дослідження: системи віддаленого біометричного моніторингу.

Предмет дослідження: технології об’єднання даних для оцінки фізіологічних параметрів стану людини на підставі біометричних даних.

Мета і завдання дослідження. Метою дослідження є підвищення точності оцінки фізіологічних параметрів стану людини на підставі біометричних даних за рахунок розроблення технології об’єднання даних в системах віддаленого біометричного моніторингу.

Для досягнення мети дослідження необхідно вирішити такі **завдання**:

– аналіз структури системи віддаленого біометричного моніторингу, методів та технологій об’єднання даних в системах біометричного моніторингу;
– проектування та розробка архітектури системи віддаленого біометричного моніторингу;
– розробка та тестування технології об’єднання даних для системи віддаленого біометричного моніторингу;
– програмна реалізація структурних елементів системи віддаленого біометричного моніторингу: мобільний додаток, додатку Windows, серверної частини ;
– практична реалізація запропонованої технології об’єднання даних;

оцінка ефективності запропонованої технології об’єднання даних.

Рисунок А.3 – Слайд 3

Архітектура системи віддаленого моніторингу симптомів хвороби Паркінсона

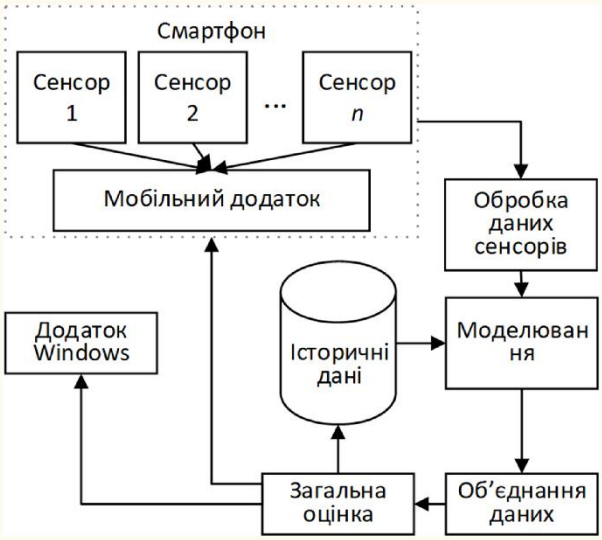


Рисунок А.4 – Слайд 4

Технологія об'єднання даних в системі віддаленого біометричного моніторингу

- ***локальна модель класифікації***
- ***глобальна модель об'єднання даних***
  - голосування на основі репутації
  - голосування більшістю

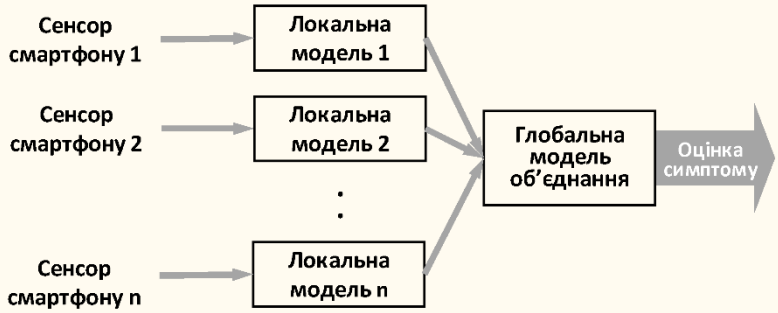


Рисунок А.5 – Слайд 5

Технологія об'єднання даних в системі віддаленого біометричного моніторингу

- ***локальна модель класифікації (Decision Tree)***

Рисунок А.6 – Слайд 6

Технологія об'єднання даних в системі віддаленого біометричного моніторингу

- ***глобальна модель об'єднання даних***
  - ***голосування на основі репутації***

порогове значення θ
позитивний голос $V_{new} = V_{old} + 1$
негативний голос $V_{new} = V_{old} - 1$

Рисунок А.7 – Слайд 7

Технологія об'єднання даних в системі віддаленого біометричного моніторингу

- ***Метод 1***

$$W_i = R_i \text{ x } Acc_i \qquad (1)$$

W_i - значення репутації, відповідно класифікованого спостереження i,
R_i - значення місцевої репутації класифікованого спостереження i з точки зору інших спостереження
Acc_i - вага класифікованого спостереження i

- ***Метод 2***

$$W_i = S_i \text{ x } Acc_i \qquad (2)$$

$$S_i = \begin{cases} 2 & \text{якщо } (R_i \geq \theta_1) \\ 1 & \text{якщо } (\theta_1 > R_i \geq \theta_2) \\ 0.5 & \text{якщо } (\theta_2 \geq R_i) \end{cases} \qquad (3)$$

Рисунок А.8 – Слайд 8

Програмна реалізація системи віддаленого біометричного моніторингу даних включає розробку наступних структурних компонентів:

Мобільний додаток для операційної системи Android

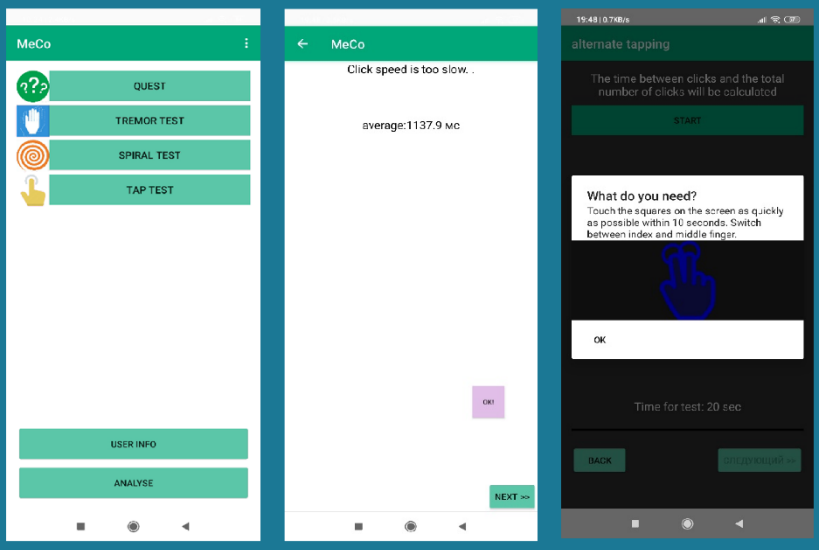
Додаток Windows

Серверна частина

Практична реалізація технології об'єднання даних датчиків смартфону для отримання оцінки ступеня тремору

Рисунок А.9 – Слайд 9

Розробка мобільного додатку



Інтерфейс розробленого мобільного додатку

Рисунок А.10 – Слайд 10

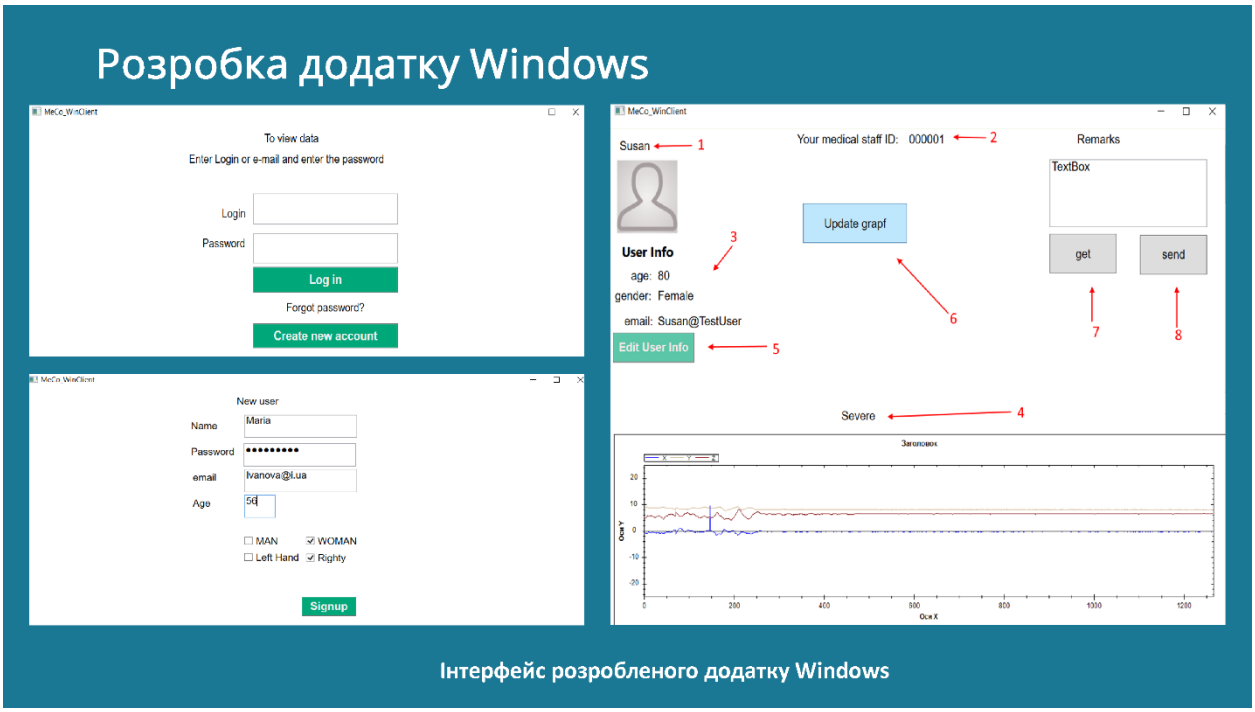


Рисунок А.11 – Слайд 11

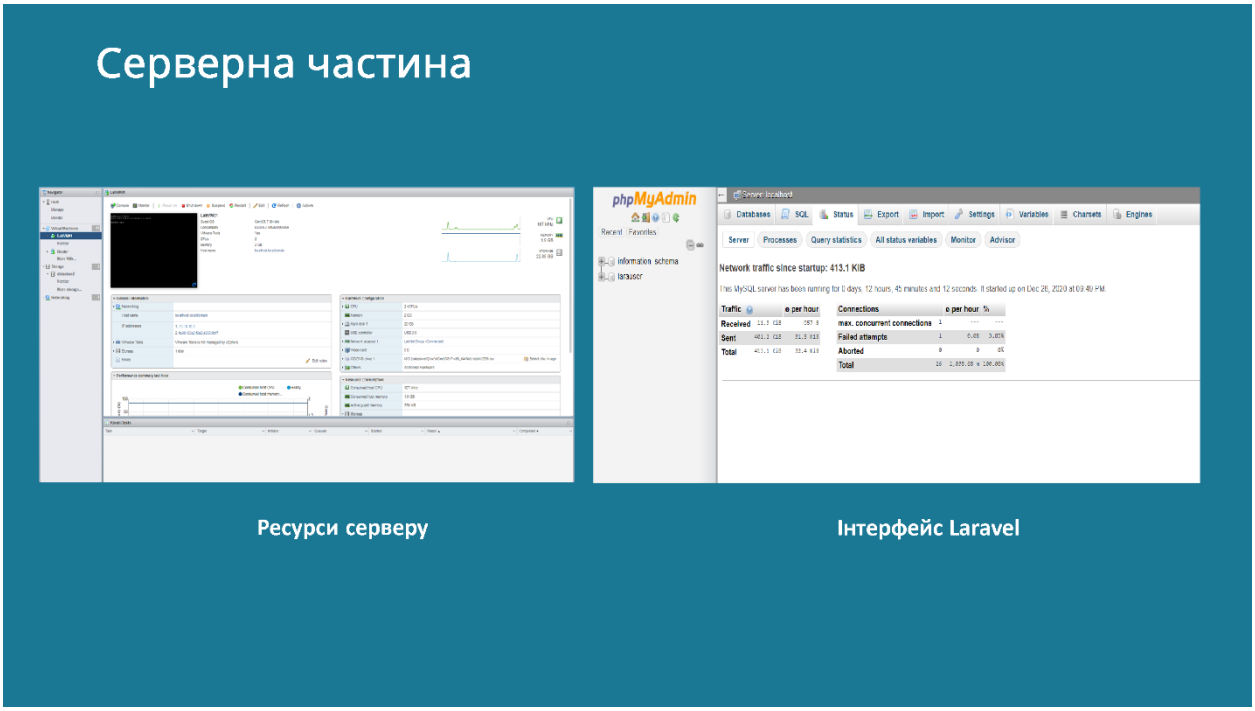
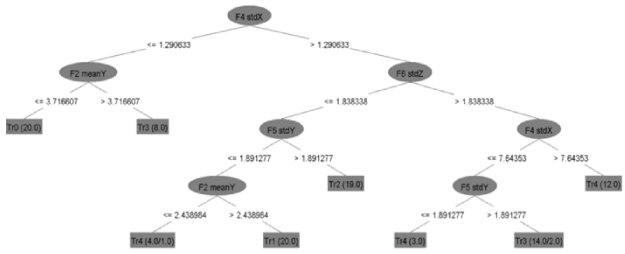


Рисунок А.12 – Слайд 12

Технологія об'єднання даних

Локальна модель класифікації



Локальна модель класифікації

Тест правої руки			Тест лівої руки		
ID спостереження	Вага Acc_i	Прогнозований клас	ID спостереження	Вага Acc_i	Прогнозований клас
1	1	Tr2	1	1	Tr3
2	0.75	Tr2	2	0.857	Tr1
3	0.75	Tr2	3	1	Tr3
4	0.75	Tr2	4	1	Tr3
5	0.75	Tr2	5	1	Tr3
6	0.75	Tr2	6	1	Tr3
7	0.75	Tr2	7	1	Tr3
8	0.75	Tr2	8	1	Tr3
9	0.75	Tr2	9	0.857	Tr1
10	0.75	Tr2	10	0.857	Tr1

Результат моделювання локальної моделі на тестах на тремор

Рисунок А.13 – Слайд 13

Технологія об'єднання даних

Голосування на підставі репутації – Метод 1

$\theta = 0.2$

$V_{new} = V_{old} + 1$

$V_{new} = V_{old} - 1$

Клас з найвищим значенням репутації для правої руки Wr1=Tr2, що дорівнює 8

Клас з найвищим значенням репутації для лівої руки Wl1= Wl3= Wl4= Wl5= Wl6= Wl7= Wl8=Tr3, що дорівнює 11

Тест правої руки				Тест лівої руки			
ID	Вага Acc_i	R_i	Прогнозований клас	ID	Вага Acc_i	R_i	Прогнозований клас
1	1	8	Tr2	1	1	11	Tr3
2	0.75	7.75	Tr2	2	0.857	10.857	Tr1
3	0.75	7.75	Tr2	3	1	11	Tr3
4	0.75	7.75	Tr2	4	1	11	Tr3
5	0.75	7.75	Tr2	5	1	11	Tr3
6	0.75	7.75	Tr2	6	1	11	Tr3
7	0.75	7.75	Tr2	7	1	11	Tr3
8	0.75	7.75	Tr2	8	1	11	Tr3
9	0.75	7.75	Tr2	9	0.857	10.857	Tr1
10	0.75	7.75	Tr2	10	0.857	10.857	Tr1

Дані отримані в результаті локальної моделі класифікації та голосування

Рисунок А.14 – Слайд 14

Технологія об’єднання даних

Голосування на підставі репутації – Метод 2

$\theta_1 = 10.5$
$\theta_2 = 7.8$

Клас з найвищим значенням репутації для правої руки W_{r1}=Tr2, що дорівнює 1

Клас з найвищим значенням репутації для лівої руки W_{l1}= W_{l3}= W_{l4}= W_{l5}= W_{l6}= W_{l7}= W_{l8}=Tr3, що дорівнює 2.

Тест правої руки					Тест лівої руки				
ID	Вага Acc_i	R_i	S_i	Прогнозований клас	ID	Вага Acc_i	R_i	S_i	Прогнозований клас
1	1	8	1	Tr2	1	1	11	2	Tr3
2	0.75	7.75	0.5	Tr2	2	0.857	10.857	2	Tr1
3	0.75	7.75	0.5	Tr2	3	1	11	2	Tr3
4	0.75	7.75	0.5	Tr2	4	1	11	2	Tr3
5	0.75	7.75	0.5	Tr2	5	1	11	2	Tr3
6	0.75	7.75	0.5	Tr2	6	1	11	2	Tr3
7	0.75	7.75	0.5	Tr2	7	1	11	2	Tr3
8	0.75	7.75	0.5	Tr2	8	1	11	2	Tr3
9	0.75	7.75	0.5	Tr2	9	0.857	10.857	2	Tr1
10	0.75	7.75	0.5	Tr2	10	0.857	10.857	2	Tr1

Результат розрахунків S_i

Рисунок А.15 – Слайд 15

ВИСНОВОК

В ході дослідницької частини роботи були отримані наступні результати:

1. Проведено аналіз структури системи віддаленого біомедичного моніторингу, методів, технологій, засобів отримання, обробки, аналізу та об’єднання даних;

2. Спроектовано та розроблено архітектури системи;

3. Розроблено технологію об’єднання даних для системи.

В ході практичної частини роботи були отримані наступні результати:

1. Програмно реалізовано: мобільний додаток, додаток Windows, серверну частину;

2. Практично реалізовано технологію об’єднання даних;

3. Визначено ефективність запропонованої технології об’єднання даних.

Рисунок А.16 – Слайд 16

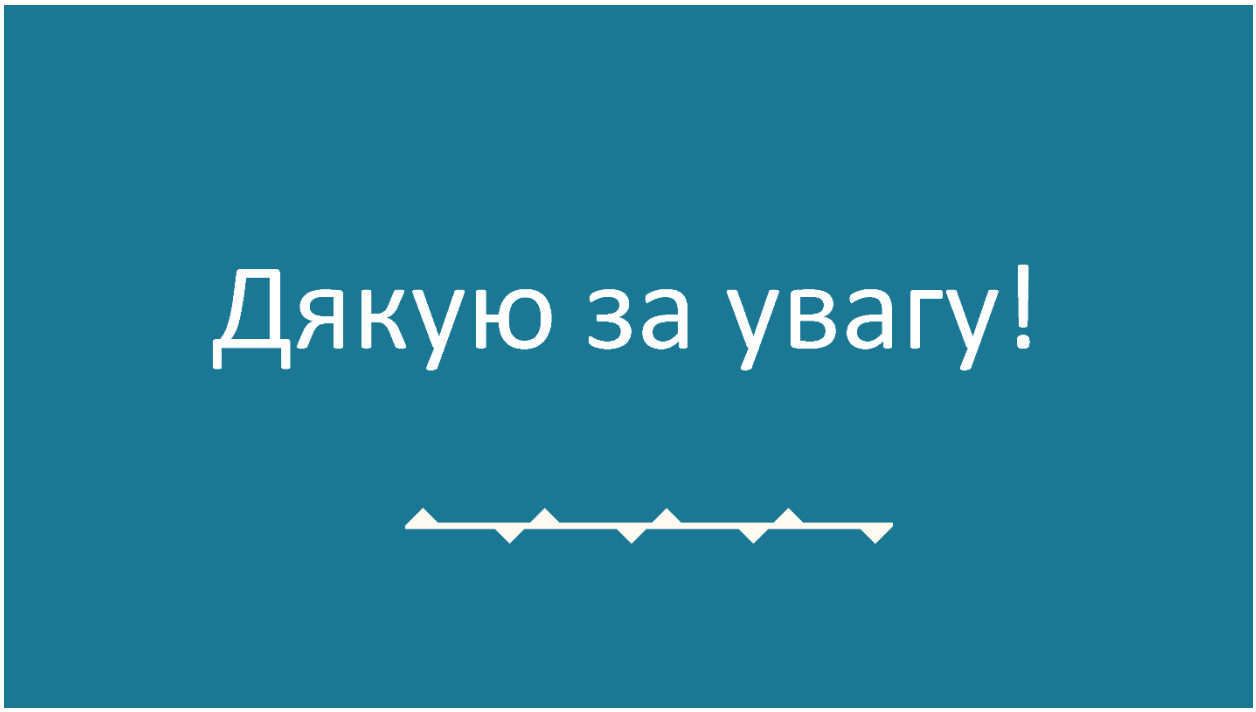


Рисунок А.17 – Слайд 17

ДОДАТОК Б ПРОГРАМНА РЕАЛІЗАЦІЯ WINDOWS ДОДАТКУ

Головний функціонал

```
1.  public partial class MainWindow : Window
2.  {
3.  public MainWindow()
4.  {
5.  InitializeComponent();
6.  ParseClient.Initialize(new ParseClient.Configuration
7.  {
8.  ApplicationId = "5n02mAVr7gQcALJY9vAxbm5n0ChxDlSEbByvnRG8",
9.  Server = "https://parseapi.back4app.com/",                // Parse
10. WindowsKey = "qiGe9ytQvuukKOCZqmLS39lJ1WvGjNm62zWu082p"
11. });

12. ParseUser currentUser = null;
13. try
14. {
15. currentUser = ParseUser.CurrentUser;
16. }
17. catch { }
18. Main_rebuild.Visibility = Visibility.Hidden;
19. New_User.Visibility = Visibility.Hidden;
20. {
21. Thickness margin = LogIN.Margin;
22. margin.Left = 250;
23. LogIN.Margin = margin;
24. }
25. }
26. private void LogIN_Click(object sender, RoutedEventArgs e)
27. {
28. CheckINAsync();//Проверить логин и пароль
29. Main_rebuild.Visibility = Visibility.Visible;
30. try
31. {
32. ParseUser currentUser =ParseUser.CurrentUser;
33. }
34. catch { }
35. if (currentUser != null)
36. try
37. {
38. UserName.Content = ParseUser.CurrentUser.Get<string>("username");
39. email.Content = currentUser.Email;
40. gender.Content = currentUser.Get<string>("Gender");
41. dob.Content = currentUser.Get<string>("DOB");
42. }
43. catch { }
44. System.Threading.Thread.Sleep(2500);
45. BTN_GET_click(sender, null) ;
46. }
47. public async Task CheckINAsync()
48. {
49. try
50. {   //если пользователь существует то войти
51. await ParseUser.LogInAsync(User_TXT.Text, Login_PASS.Password);
52. LogIN.Visibility = Visibility.Collapsed;
53. Grafik_Tremx.Visibility = Visibility.Visible;
54. if (currentUser != null)
55. try if (UserName.Content.ToString() != ParseUser.CurrentUser.Get<string>("username"))
56. {
57. UserName.Content = ParseUser.CurrentUser.Get<string>("username");
58. email.Content = currentUser.Email;
59. gender.Content = currentUser.Get<string>("Gender");
60. dob.Content = currentUser.Get<string>("DOB");
61. }
62. GetAccelDataAsync();
63. }
64. catch { }
65. }
66. catch
67. {   //если нет, то выдать ошибку
68. LogIN_Label1.Content = "Ошибка! Пользователь не найден либо не верный пароль";
69. LogIN_Label1.Background = System.Windows.Media.Brushes.Red;
70. Grafik_Tremx.Visibility = Visibility.Hidden;
```

```
71. }
72. }

73. private void New_usr_btn_Click(object sender, RoutedEventArgs e)
74. {
75. {
76. Thickness margin = New_User.Margin;
77. margin.Left = 250;
78. New_User.Margin = margin;
79. }
80. LogIN.Visibility = Visibility.Hidden;
81. New_User.Visibility = Visibility.Visible;

82. BTN_GET_hlpclk();
83. }
84. }
```

Функції SendNote_Click, Get_Note_Click, SignUP_click

```
85. private void SendNote_Click(object sender, RoutedEventArgs e)
86. {
87. ParseObject Remarks = new ParseObject("Remarks")
88. {
89. ["username"] = ParseUser.CurrentUser.Username.ToString(),
90. ["email"] = ParseUser.CurrentUser.Email.ToString(),
91. ["Notes"] = TXT1.Text
92. };
93. Remarks.SaveAsync();
94. }
95. private void Get_Note_Click(object sender, RoutedEventArgs e)
96. {
97. Get_NoteAsync();
98. }
99. public async Task Get_NoteAsync (){
100.   string username = ParseUser.CurrentUser.Username;
101.   var query = ParseObject.GetQuery("Remarks")
102.   .WhereEqualTo("username", username);
103.   //query = query.Limit(5);
104.   IEnumerable<ParseObject> results = await query.FindAsync();
105.   ParseObject LastData = results.Last();
106.   string Note_string = LastData.Get<string>("Notes");
107.   TXT1.Text = Note_string;
108.   }
109.   private void SignUP_click(object sender, RoutedEventArgs e)
110.   {
111.   Main_rebuild.Visibility = Visibility.Visible;
112.   New_User.Visibility = Visibility.Hidden;
113.   var user = new ParseUser()
114.   {
115.   Username = SignUp_NAME.Text,
116.   Password = SignUp_PASS.Password,
117.   Email = SignUp_MAIL.Text
118.   };
119.   user["DOB"] = SignUp_AGE.Text;
120.   user["DOB"] = CHK_LeftHand.IsChecked==true? "Righty" : CHK_Righty.IsChecked == true ?
    "Lefty" : "ERR1";
121.   user["DOB"] = CHK_MAN.IsChecked == true ? "Male" : CHK_WOMAN.IsChecked == true ?
    "Female" : "ERR2";
122.   user.SignUpAsync();
123.   BTN_GET_hlpclk();
124.   }
125.   }
```

Функція CreateGrapf

```
126.   public void CreateGrapf(int max)
127.   {
128.   // Получим панель для рисования
129.   GraphPane pane = graph.GraphPane;
130.   // Очистим список кривых на тот случай, если до этого сигналы уже были нарисованы
131.   pane.CurveList.Clear();
132.   // Создадим список точек
133.   PointPairList listX = new PointPairList();
134.   PointPairList listY = new PointPairList();
135.   PointPairList listZ = new PointPairList();
136.   int xmin = 0;
137.   int xmax = max;
138.   // Устанавливаем интересующий нас интервал по оси X
139.   pane.XAxis.Scale.Min = 0;
140.   pane.XAxis.Scale.Max = max;
```

```
141.   // Устанавливаем интересующий нас интервал по оси Y
142.   pane.YAxis.Scale.Min = -25;
143.   pane.YAxis.Scale.Max = 25;
144.   // Заполняем список точек
145.   listX.Clear();
146.   listY.Clear();
147.   listZ.Clear();
148.   for (int x = xmin; x < xmax; x ++)
149.   {   // добавим в список точкі
150.   listX.Add(x, X_Stamp[x]);
151.   listY.Add(x, Y_Stamp[x]);
152.   listZ.Add(x, Z_Stamp[x]);
153.   }
154.   // Создадим кривую с названием "NAME", которая будет рисоваться голубым цветом
155.   LineItem myCurveX = pane.AddCurve("X", listX, System.Drawing.Color.Blue,
    SymbolType.None);
156.   LineItem myCurveY = pane.AddCurve("Y", listY, System.Drawing.Color.Tan,
    SymbolType.None);
157.   LineItem myCurveZ = pane.AddCurve("Z", listZ, System.Drawing.Color.Maroon,
    SymbolType.None);
158.   // Вызываем метод AxisChange (), чтобы обновить данные об осях.
159.   graph.AxisChange();
160.   // Обновляем график
161.   graph.Invalidate();
162.   }
```

Функції NormalizeArrayZ, getVarCoeff, getMean, Statistics, getVariance.

```
163.   private double[] NormalizeArrayZ(double[] Arr, double M)
164.   {
165.   double[] ArrZ_d = new double[Arr.Length];
166.   for (int i = 0; i < Arr.Length; i++)
167.   {
168.   ArrZ_d[i] = (Arr[i]) / M; //get Z-axis only
169.   }
170.   return ArrZ_d;
171.   }
172.   double[] ArrZ_double = new double[0];
173.   private double getVarCoeff() //returns coefficient of variation
174.   {
175.   double Mean = getMean();
176.   double[] ArrZ_Normalized = new double[Z_Stamp.Count];
177.   ArrZ_Normalized = NormalizeArrayZ(ArrZ_double, Mean);

178.   Statistics s = new Statistics((ArrZ_Normalized));
179.   return s.getVariance();
180.   }
181.   private double getMean()
182.   {
183.   ArrZ_double = new double[Z_Stamp.Count];
184.   for (int a=0;a< Z_Stamp.Count;a++)
185.   {
186.   ArrZ_double[a] = Convert.ToDouble(Z_Stamp[a]);
187.   }
188.   Statistics s = new Statistics((ArrZ_double));
189.   return s.getMean();
190.   }
191.   double[] data;
192.   int size;

193.   public Statistics(double[] data)
194.   {
195.   this.data = data;
196.   size = data.Length;
197.   }
198.   public double getMean()
199.   {
200.   double sum = 0.0;
201.   for (int k = 0; k < data.Length; k++)
202.   {
203.   double a = data[k];
204.   sum += a;
205.   }

206.   return sum / size;
207.   }
208.   public double getVariance()
209.   {
210.   double mean = getMean();
```

```
211.   double temp = 0;
212.   for (int k = 0; k < data.Length; k++)
213.   {
214.   double a = data[k];
215.   temp += (mean - a) * (mean - a);
216.   }
217.   return temp / size;
218.   }
219.   double getStdDev()
220.   {
221.   return Math.Sqrt(getVariance());
222.   }
```

Функції Get_UserListAsync BTN_GET, BTN_GET_hlpclk, GetAccelDataAsync, AnalyseAccelDataAsync.

```
223.   public async Task Get_UserListAsync()
224.   {
225.   if (!IsUserGetted)
226.   {
227.   User_List.Items.Clear();
228.   try
229.   {
230.   var userObjects = await ParseUser.Query.FindAsync();
231.   for (int i = 0; i < userObjects.Count(); i++)
232.   {
233.   string UserLST =
    userObjects.ElementAtOrDefault(i).Username;//ElementAt(i).Email.ToString();
234.   User_List.Items.Add(UserLST);
235.   }
236.   User_List.SelectedIndex = 0;
237.   }
238.   catch
239.   {
240.   User_List.Items.Add("ERROR03");
241.   }
242.   }
243.   IsUserGetted = true;
244.   }
245.   private void BTN_GET_click(object sender, RoutedEventArgs e)
246.   {
247.   BTN_GET_hlpclk();
248.   }
249.   public async void BTN_GET_hlpclk()
250.   {
251.   try
252.   {
253.   currentUser = ParseUser.CurrentUser;
254.   }
255.   catch { }
256.   if (currentUser != null)
257.   try
258.   {
259.   if (UserName.Content.ToString() != ParseUser.CurrentUser.Get<string>("username"))
260.   {
261.   UserName.Content = ParseUser.CurrentUser.Get<string>("username");
262.   email.Content = currentUser.Email;
263.   gender.Content = currentUser.Get<string>("Gender");
264.   dob.Content = currentUser.Get<string>("DOB");
265.   }
266.   await GetAccelDataAsync();
267.   }
268.   catch { }
269.   }
270.   public async Task GetAccelDataAsync()
271.   {
272.   X_Stamp = new List<double>();
273.   Y_Stamp = new List<double>();
274.   Z_Stamp = new List<double>();
275.   //Grafik_Tremx.Visibility = Visibility.Visible;
276.   string username = ParseUser.CurrentUser.Username;
277.   List<double> time = new List<double>();
278.   var query = ParseObject.GetQuery("AccelData")
279.   .WhereEqualTo("username", username);
280.   IEnumerable<ParseObject> results = await query.FindAsync();
281.   ParseObject LastData = results.Last();
282.   string Data = LastData.Get<string>("ArrayList");
283.   for (int X = 5; X != Data.Length; X++)
284.   {
```

```
285.  if (Data[X] == ':' & Data[X - 2] == 'p' & Data[X - 3] == 'm')
286.  {
287.  for (int I = 8; I < 13; I++)
288.  {
289.  timestamp += Data[X + I].ToString();
290.  }
291.  try
292.  {
293.  time.Add(Convert.ToDouble(timestamp));
294.  }
295.  catch
296.  {
297.  time.Add(0);
298.  }
299.  timestamp = "";
300.  }
301.  if (Data[X] == ':' & Data[X - 2] == 'x')
302.  {
303.  for (int I = 1; I < 5; I++)
304.  {
305.  if (Data[X + I].ToString() == ".") Xval += ',';
306.  else Xval += Data[X + I].ToString();
307.  }
308.  try
309.  {
310.  X_Stamp.Add(Convert.ToDouble(Xval));
311.  }
312.  catch
313.  {
314.  X_Stamp.Add(0);
315.  }
316.  }
317.  if (Data[X] == ':' & Data[X - 2] == 'y')
318.  {
319.  for (int I = 1; I < 5; I++)
320.  {
321.  if (Data[X + I].ToString() == ".") Yval += ',';
322.  else Yval += Data[X + I].ToString();
323.  }
324.  try
325.  {
326.  Y_Stamp.Add(Convert.ToDouble(Yval));
327.  }
328.  catch
329.  {
330.  Y_Stamp.Add(0);
331.  }
332.  Yval = "";
333.  }
334.  if (Data[X] == ':' & Data[X - 2] == 'z')
335.  {
336.  for (int I = 1; I < 5; I++)
337.  {
338.  if (Data[X + I].ToString() == ".") Zval += ',';
339.  else Zval += Data[X + I].ToString();
340.  }
341.  try
342.  {
343.  Z_Stamp.Add(Convert.ToDouble(Zval));
344.  }
345.  catch
346.  {
347.  Z_Stamp.Add(0);
348.  }
349.  Zval = "";
350.  }
351.  }
352.  CreateGrapf(time.Count);
353.  AnalyseAccelDataAsync();
354.  }
355.  public int AnalyseAccelDataAsync()
356.  {

357.  int resscale;
358.  double Mean = getMean();
359.  double VC = getVarCoeff();
360.  if (VC >= 1.58E-04 && VC < 3.26E-04
361.  {
362.  resscale = 1;
```

```
363.   Level_trem.Content = "Slight";
364.   }
365.   else if (VC >= 3.26E-04 && VC < 8.16E-04
366.   {
367.   resscale = 2;
368.   Level_trem.Content = "Mild";//+
369.   }
370.   else if (VC >= 8.16E-04 && VC < 5.31E-03
371.   {
372.   resscale = 3;
373.   Level_trem.Content = "Moderate";
374.   }
375.   else if (VC >= 5.31E-03
376.   {
377.   resscale = 4;
378.   Level_trem.Content = "Severe";
379.   }
380.   else
381.   {
382.   resscale = 0;
383.   Level_trem.Content = "Normal";
384.   }
385.   return resscale;
386.   }
```

ДОДАТОК В ПРОГРАМНА РЕАЛІЗАЦІЯ ДЕРЕВА РІШЕНЬ

```
1.      import weka.core.Attribute;
2.      import weka.core.Capabilities;
3.      import weka.core.Capabilities.Capability;
4.      import weka.core.Instance;
5.      import weka.core.Instances;
6.      import weka.core.RevisionUtils;
7.      import weka.classifiers.Classifier;
8.      import weka.classifiers.AbstractClassifier;
9.      public class WekaWrapper
10.       extends AbstractClassifier {
11.       /**
12.        * Returns only the toString() method.
13.        *
14.        * @return a string describing the classifier
15.        */
16.       public String globalInfo() {
17.         return toString();
18.       }
19.
20.       /**
21.        * Returns the capabilities of this classifier.
22.        *
23.        * @return the capabilities
24.        */
25.       public Capabilities getCapabilities() {
26.         weka.core.Capabilities result = new weka.core.Capabilities(this);
27.
28.         result.enable(weka.core.Capabilities.Capability.NOMINAL_ATTRIBUTES);
29.         result.enable(weka.core.Capabilities.Capability.NUMERIC_ATTRIBUTES);
30.         result.enable(weka.core.Capabilities.Capability.DATE_ATTRIBUTES);
31.         result.enable(weka.core.Capabilities.Capability.MISSING_VALUES);
32.         result.enable(weka.core.Capabilities.Capability.NOMINAL_CLASS);
33.         result.enable(weka.core.Capabilities.Capability.MISSING_CLASS_VALUES);
34.         result.setMinimumNumberInstances(0);
35.         return result;
36.       }
37.
38.       /**
39.        * only checks the data against its capabilities.
40.        *
41.        * @param i the training data
42.        */
43.       public void buildClassifier(Instances i) throws Exception {
44.         // can classifier handle the data?
45.         getCapabilities().testWithFail(i);
46.       }
47.
48.       /**
49.        * Classifies the given instance.
50.        *
51.        * @param i the instance to classify
52.        * @return the classification result
53.        */
54.       public double classifyInstance(Instance i) throws Exception {
55.         Object[] s = new Object[i.numAttributes()];
56.
57.         for (int j = 0; j < s.length; j++) {
58.           if (!i.isMissing(j)) {
59.             if (i.attribute(j).isNominal())
60.               s[j] = new String(i.stringValue(j));
61.             else if (i.attribute(j).isNumeric())
62.               s[j] = new Double(i.value(j));
63.           }
64.         }
65.
66.         // set class value to missing
67.         s[i.classIndex()] = null;
68.
69.         return WekaClassifier.classify(s);
70.       }
71.
72.       /**
73.        * Returns the revision string.
74.        *
```

```
75.         * @return        the revision
76.         */
77.        public String getRevision() {
78.          return RevisionUtils.extract("1.0");
79.        }
80.
81.        /**
82.         * Returns only the classnames and what classifier it is based on.
83.         *
84.         * @return a short description
85.         */
86.        public String toString() {
87.          return "Auto-generated classifier wrapper, based on weka.classifiers.trees.J48
       (generated with Weka 3.9.1).\n" + this.getClass().getName() + "/WekaClassifier";
88.        }
89.
90.        /**
91.         * Runs the classfier from commandline.
92.         *
93.         * @param args the commandline arguments
94.         */
95.        public static void main(String args[]) {
96.          runClassifier(new WekaWrapper(), args);
97.        }
98.      }
99.
100.     class WekaClassifier {
101.
102.       public static double classify(Object[] i)
103.         throws Exception {
104.
105.         double p = Double.NaN;
106.         p = WekaClassifier.N59f991590(i);
107.         return p;
108.       }
109.       static double N59f991590(Object []i) {
110.         double p = Double.NaN;
111.         if (i[3] == null) {
112.           p = 0;
113.         } else if (((Double) i[3]).doubleValue() <= 1.2906328370036) {
114.         p = WekaClassifier.N44402dab1(i);
115.         } else if (((Double) i[3]).doubleValue() > 1.2906328370036) {
116.         p = WekaClassifier.N2d088d992(i);
117.         }
118.         return p;
119.       }
120.       static double N44402dab1(Object []i) {
121.         double p = Double.NaN;
122.         if (i[1] == null) {
123.           p = 0;
124.         } else if (((Double) i[1]).doubleValue() <= 3.7166071428571) {
125.           p = 0;
126.         } else if (((Double) i[1]).doubleValue() > 3.7166071428571) {
127.           p = 3;
128.         }
129.         return p;
130.       }
131.       static double N2d088d992(Object []i) {
132.         double p = Double.NaN;
133.         if (i[5] == null) {
134.           p = 1;
135.         } else if (((Double) i[5]).doubleValue() <= 1.8383383548776) {
136.         p = WekaClassifier.N3983391f3(i);
137.         } else if (((Double) i[5]).doubleValue() > 1.8383383548776) {
138.         p = WekaClassifier.N690d597d5(i);
139.         }
140.         return p;
141.       }
142.       static double N3983391f3(Object []i) {
143.         double p = Double.NaN;
144.         if (i[4] == null) {
145.           p = 1;
146.         } else if (((Double) i[4]).doubleValue() <= 1.891276673105) {
147.         p = WekaClassifier.N2c65042f4(i);
148.         } else if (((Double) i[4]).doubleValue() > 1.891276673105) {
149.           p = 2;
150.         }
151.         return p;
152.       }
```

```
153.       static double N2c65042f4(Object []i) {
154.         double p = Double.NaN;
155.         if (i[1] == null) {
156.           p = 4;
157.         } else if (((Double) i[1]).doubleValue() <= 2.4389842519685) {
158.           p = 4;
159.         } else if (((Double) i[1]).doubleValue() > 2.4389842519685) {
160.           p = 1;
161.         }
162.         return p;
163.       }
164.       static double N690d597d5(Object []i) {
165.         double p = Double.NaN;
166.         if (i[3] == null) {
167.           p = 3;
168.         } else if (((Double) i[3]).doubleValue() <= 7.6435302658132) {
169.         p = WekaClassifier.N8075806(i);
170.         } else if (((Double) i[3]).doubleValue() > 7.6435302658132) {
171.           p = 4;
172.         }
173.         return p;
174.       }
175.       static double N8075806(Object []i) {
176.         double p = Double.NaN;
177.         if (i[4] == null) {
178.           p = 4;
179.         } else if (((Double) i[4]).doubleValue() <= 1.891276673105) {
180.           p = 4;
181.         } else if (((Double) i[4]).doubleValue() > 1.891276673105) {
182.           p = 3;
183.         }
184.         return p;
185.       }
186.     }
```